

Universidad Rey Juan Carlos

Departamento de Teoría de la Señal y Comunicaciones  
y Sistemas Telemáticos y Computación



Concurso al Cuerpo de Catedráticos de Universidad

Área de Conocimiento: Ingeniería Telemática

Plaza: Y158/DF005206/16-10-2017

**Contexto Académico,  
Proyecto Docente,  
Proyecto Investigador y  
Curriculum Vitae**

Jesús M. González Barahona

Noviembre 2017



Dedicado a todas las personas  
que habéis hecho posible  
que yo haya llegado hasta aquí,  
que sois muchas.



# Índice general

<b>Enlaces relevantes</b>	<b>1</b>
<b>1. Introducción</b>	<b>5</b>
<b>I Contexto académico</b>	<b>7</b>
<b>2. Contexto general</b>	<b>11</b>
2.1. Marco legal . . . . .	11
2.2. La población Universitaria . . . . .	12
2.3. Presupuestos de las Universidades Públicas . . . . .	14
<b>3. La Universidad Rey Juan Carlos</b>	<b>17</b>
<b>4. La Escuela y el Departamento</b>	<b>23</b>
4.1. La Escuela Técnica Superior de Ingeniería de Telecomunicación . . . . .	23
4.2. El Departamento de Teoría de la Señal y Comunicaciones y Sistemas Telemáticos y Computación . . . . .	24
<b>Bibliografía del contexto académico</b>	<b>27</b>
<b>II Proyecto docente</b>	<b>29</b>
<b>5. Los estudios de Servicios y Aplicaciones Telemáticas</b>	<b>33</b>
5.1. Curricula relacionados con la informática . . . . .	33
5.1.1. SE2014 . . . . .	33
5.1.2. IS2010 . . . . .	34
5.1.3. ANECA Ingeniería Informática . . . . .	35
5.2. Curricula relacionados con las telecomunicaciones . . . . .	36
5.3. Las titulaciones de Ingeniería de Telecomunicación en la URJC . . . . .	37
5.3.1. Servicios y Aplicaciones Telemáticas dentro de sus itinerario formativo . . . . .	38
5.3.2. Aplicaciones Telemáticas dentro de sus itinerario formativo . . . . .	42
<b>6. Propuesta de método didáctico</b>	<b>45</b>
6.1. Objetivos docentes . . . . .	45
6.1.1. El proceso de definición de los objetivos . . . . .	46
6.1.2. Objetivos generales . . . . .	47
6.2. Método docente . . . . .	47
6.2.1. Aprendizaje basado en problemas . . . . .	48
6.2.2. Introducción de conceptos teóricos . . . . .	49
6.2.3. Otros recursos de ayuda a la docencia . . . . .	51
6.3. Evaluación . . . . .	53
6.3.1. Evaluación del alumno . . . . .	53
6.3.2. Evaluación de la asignatura y del profesor . . . . .	55

6.4.	Medios necesarios . . . . .	56
6.4.1.	Laboratorio docente . . . . .	56
6.4.2.	Técnico de apoyo . . . . .	58
6.4.3.	Forja de desarrollo y sistemas anejos . . . . .	58
6.4.4.	Campus virtual y sitio de recursos abiertos . . . . .	58
<b>7.</b>	<b>Asignatura: Servicios y Aplicaciones Telemáticas</b>	<b>61</b>
7.1.	Datos generales . . . . .	61
7.2.	Objetivos . . . . .	61
7.3.	Metodología . . . . .	62
7.4.	Evaluación . . . . .	62
7.5.	Programa detallado . . . . .	63
7.5.1.	00 - Presentación . . . . .	63
7.5.2.	01 - Conceptos básicos de aplicaciones web . . . . .	63
7.5.3.	02 - Servicios web que interoperan . . . . .	65
7.5.4.	03 - Modelo-vista-controlador . . . . .	65
7.5.5.	04 - Introducción a XML . . . . .	65
7.5.6.	05 - Hojas de estilo CSS . . . . .	66
7.5.7.	06 - Ajax . . . . .	66
7.5.8.	P1 - Introducción a Python . . . . .	67
7.5.9.	P2 - Aplicaciones web simples . . . . .	67
7.5.10.	P3 - Servidores simples de contenidos . . . . .	68
7.5.11.	P4 - Introducción a Django . . . . .	68
<b>8.</b>	<b>Asignatura: Aplicaciones Telemáticas</b>	<b>71</b>
8.1.	Datos generales . . . . .	71
8.2.	Objetivos . . . . .	71
8.3.	Metodología . . . . .	72
8.4.	Evaluación . . . . .	72
8.5.	Programa detallado . . . . .	73
8.5.1.	Presentación . . . . .	73
8.5.2.	Introducción a HTML . . . . .	73
8.5.3.	Introducción a CSS . . . . .	74
8.5.4.	Introducción a Bootstrap . . . . .	74
8.5.5.	Introducción a JavaScript . . . . .	75
8.5.6.	Introducción a jQuery . . . . .	76
8.5.7.	Introducción a HTML5 . . . . .	77
8.5.8.	Otras bibliotecas JavaScript . . . . .	78
8.5.9.	APIs JavaScript . . . . .	78
8.5.10.	APIs de Google . . . . .	79
8.5.11.	Presentación de la Práctica final . . . . .	79
<b>9.</b>	<b>Ejemplo de transparencias</b>	<b>81</b>
<b>10.</b>	<b>Sitio web de las asignaturas</b>	<b>85</b>
<b>11.</b>	<b>Ejemplos de ejercicios</b>	<b>87</b>
11.0.1.	Calculadora . . . . .	87
11.0.2.	Sumador simple versión REST . . . . .	87
11.0.3.	Calculadora simple versión REST . . . . .	87
11.0.4.	Django calc . . . . .	88

<b>12.Ejemplos de prácticas</b>	<b>89</b>
12.1. Socios . . . . .	89
12.2. Aparcamientos para residentes en Madrid (mayo 2017, borrador) . . . . .	90
12.2.1. Enunciado . . . . .	90
12.2.2. Funcionalidad optativa . . . . .	92
12.2.3. Entrega de la práctica . . . . .	93
12.3. Entrega de prácticas . . . . .	94
12.3.1. Uso de la rama gh-pages . . . . .	94
12.3.2. Uso de la rama gh-pages con dos clones . . . . .	96
12.3.3. Más información sobre GitHub Pages . . . . .	96
<b>13.Ejemplo de examen</b>	<b>97</b>
13.1. Enunciado . . . . .	97
13.2. Soluciones . . . . .	99
<b>14.Ejemplo de encuesta a los alumnos</b>	<b>103</b>
<b>Bibliografía del proyecto docente</b>	<b>105</b>
<b>III Proyecto de investigación</b>	<b>107</b>
<b>15.Introducción al proyecto de investigación</b>	<b>111</b>
15.1. Resumen en español . . . . .	111
15.2. Summary in English . . . . .	112
<b>16.Scientific proposal</b>	<b>113</b>
16.1. State of the art . . . . .	114
16.2. Previous experience and relationship with other research groups . . . . .	115
16.3. Research and technical objectives . . . . .	116
16.4. Specific goals . . . . .	117
<b>17.Realization</b>	<b>119</b>
17.1. Methodology . . . . .	119
17.2. Infrastructure and equipment . . . . .	120
17.3. Schedule . . . . .	120
17.4. Deliverables . . . . .	121
17.5. Personnel . . . . .	121
<b>18.Impact and training</b>	<b>123</b>
18.1. Transfer of results . . . . .	123
18.2. Dissemination plan . . . . .	124
18.3. Training capabilities and program . . . . .	124
<b>Research Project References</b>	<b>127</b>
<b>IV Curriculum vitae</b>	<b>129</b>
<b>19.Datos generales</b>	<b>133</b>
<b>20.Indicios de calidad</b>	<b>135</b>
20.1. Indicios generales . . . . .	135
20.2. Periodo 2011-2016 . . . . .	135
20.3. Periodo 2005-2010 . . . . .	136
<b>21.Participación en proyectos de I+D</b>	<b>139</b>

<b>22. Contratos de I+D con empresas y administraciones</b>	<b>147</b>
<b>23. Publicaciones o Documentos Científico-Técnicos</b>	<b>153</b>
<b>24. Contribuciones a congresos</b>	<b>161</b>
<b>25. Tesis doctorales dirigidas</b>	<b>183</b>
<b>26. Participación en comités y grupos técnicos</b>	<b>185</b>
<b>27. Estancias en Centros extranjeros</b>	<b>187</b>
<b>28. Experiencia en organización de actividades de I+D</b>	<b>189</b>
<b>29. Otros méritos</b>	<b>193</b>
29.1. Tesis doctoral . . . . .	193
29.2. Publicaciones docentes . . . . .	193
29.3. Actividades relacionadas con la docencia . . . . .	193
29.4. Participación en el diseño de planes de estudios . . . . .	193
29.5. Enseñanza reglada impartida . . . . .	194
29.6. Grupos de trabajo . . . . .	199
29.7. Becas y Premios . . . . .	199
29.8. Periodos de docencia e investigación reconocidos . . . . .	200
29.9. Comités de publicaciones científicas . . . . .	200
29.10. Sociedades Científicas y Profesionales . . . . .	200
29.11. Labores académicas de gestión . . . . .	200
29.12. Actividades de transferencia de resultados de I+D . . . . .	201
29.13. Otros . . . . .	201



# Enlaces relevantes

Se incluyen a continuación algunos enlaces que pueden ser interesantes para completar los contenidos de esta memoria:

- Información complementaria a esta memoria:  
<http://gsyc.es/~jgb/plaza-cu>
- Sitio web de la Universidad Rey Juan Carlos:  
<http://urjc.es>
- Sitio “Cursos web”, con recursos abiertos para las dos asignaturas presentadas en el proyecto docente:  
<http://cursosweb.github.io/>
- Lista de publicaciones del candidato en Google Scholar:  
<https://scholar.google.es/citations?user=vYIPWBOAAAAJ>
- Lista de publicaciones del candidato en DBLP:  
[http://dblp.dagstuhl.de/pers/hy/g/Gonz=acute=lez=Barahona:Jes=uacute=s\\_M=.html](http://dblp.dagstuhl.de/pers/hy/g/Gonz=acute=lez=Barahona:Jes=uacute=s_M=.html)
- GrimoireLab (conjunto de herramientas para la analítica del desarrollo de software, en cuya concepción e implementación participa el ponente).  
<http://grimoirelab.github.io>
- Bitergia (empresa de base tecnológica de la que el ponente es co-fundador):  
<http://bitergia.com>



A veces me pregunto  
que hago yo aquí

---

José Antonio Labordeta



# Capítulo 1

## Introducción

Esta memoria se presenta como parte de la documentación elaborada para concursar a la plaza Y158/DF005206/16-10-2017 del cuerpo de Catedráticos de Universidad del área de conocimiento Ingeniería Telemática de la Universidad Rey Juan Carlos.

Está compuesta por las siguientes partes:

- Contexto Académico (parte I). Presentará el contexto académico en que se enmarca la plaza: el contexto universitario español, la Universidad Rey Juan Carlos, su Escuela Superior de Ingeniería de Telecomunicación y su Departamento de Teoría de la Señal y Comunicaciones y Sistemas Telemáticos y Computación.
- Proyecto Docente (parte II). Presentará el proyecto docente para las dos asignaturas especificadas en el perfil de la plaza.
- Proyecto Investigador (parte III). Presentará el proyecto investigador que se propone para concursar, basado en un proyecto real elaborado por el candidato.
- Curriculum Vitae (parte IV). Presentará los principales datos del curriculum vitae del candidato.

Además, el tribunal podrá consultar información complementaria (incluyendo una versión digital de esta memoria) en:

<http://gsyc.es/~jgb/plaza-cu>



**Parte I**

**Contexto académico**





*Artículo 1. Funciones de la Universidad.*

- 1. La Universidad realiza el servicio público de la educación superior mediante la investigación, la docencia y el estudio.*
- 2. Son funciones de la Universidad al servicio de la sociedad:*
  - a) La creación, desarrollo, transmisión y crítica de la ciencia, de la técnica y de la cultura.*
  - b) La preparación para el ejercicio de actividades profesionales que exijan la aplicación de conocimientos y métodos científicos y para la creación artística.*
  - c) La difusión, la valorización y la transferencia del conocimiento al servicio de la cultura, de la calidad de la vida, y del desarrollo económico.*
  - d) La difusión del conocimiento y la cultura a través de la extensión universitaria y la formación a lo largo de toda la vida.*



## Capítulo 2

# Contexto general

Desde el punto de vista de organización, la plaza objeto del concurso que nos ocupa se inscribe dentro del Departamento de Teoría de la Señal y Comunicaciones y Sistemas Telemáticos y Computación (DTSCySTC) de la Escuela Técnica Superior de Ingeniería de Telecomunicación (ETSIT) de la Universidad Rey Juan Carlos (URJC). Este capítulo se dedicará a describir la situación actual de dichos Departamento y Escuela, y de la Universidad en la que se enmarcan, dado que influirán en el entorno docente. Desde una perspectiva más general también se introducirán algunos apuntes sobre la situación actual de la Universidad en España, que sin duda mediatiza y condiciona las situaciones particulares de cualquier actividad docente realizada en este país.

### 2.1. Marco legal

La base del ordenamiento jurídico por el que rige actualmente la Universidad española es la Ley Orgánica de Universidades (LOU), 6/2001 [1], modificada por la Ley Orgánica 4/2007 [2], y con última modificación del 28 de junio de 2017. La Ley Orgánica 4/2007 fue la primera en asumir los acuerdos internacionales relacionados con el Espacio Europeo de Educación Superior, por lo que a partir de ella se han desarrollado con plenitud las titulaciones que respetan, y están acogidas a estos acuerdos. Este ordenamiento jurídico ha mantenido la transferencia de amplias cuotas de autonomía desde el Estado Central a las Comunidades Autónomas y sobre todo a las propias Universidades que realizó la Ley Orgánica de Reforma Universitaria (L.R.U.) de 25 de agosto de 1983. La Comunidad Autónoma de Madrid asumió a finales de la década de 1990 la totalidad de las competencias en Educación que le corresponden, y con ellas la responsabilidad sobre las Universidades madrileñas. La Universidad Rey Juan Carlos fue la primera (y por ahora única) Universidad pública madrileña que ha sido creada por la propia Comunidad Autónoma de Madrid, que también la gestionó durante sus primeros años, hasta que empezó a ejercer su propia autonomía universitaria en 2004.

El artículo 1 de la LOU configura la Universidad como una institución dedicada a la investigación, a la docencia y al estudio. Cualquier Universidad tiene que preocuparse de su potencial docente y de desarrollarse como parte de la comunidad investigadora. Y también ha de mantener una relación fluida con el tejido productivo, realizando la necesaria transferencia tecnológica a la sociedad. Por este motivo los profesionales que trabajamos en ella hemos de ocuparnos, cada uno en la medida de sus funciones y capacidades, de que se desarrollen todas estas dimensiones. Sin descuidar la excelencia docente, hemos de ser capaces de aportar novedades al desarrollo científico y cultural, y ser capaz de transferir los resultados de nuestra labor al conjunto de la sociedad.

Ante esta situación ideal, las Universidades reales, y los profesionales que trabajamos en ellas, nos encontramos con numerosos problemas. Entre ellos cabe destacar la insuficiencia presupuestaria, la masificación, y las trabas administrativas. Además, las Universidades jóvenes, como la URJC, tienen sus propios problemas, derivados habitualmente de un rápido crecimiento, y de una cierta falta de experiencia de todas las estructuras Universitarias. Pero también tienen sus propias potencialidades, derivadas de la falta de historia y bagajes del pasado, que leídos en positivo, implican que muchas ventanas de oportunidad que en otras Universidades más antiguas quizás se hayan cerrado, estén aún abiertas.

Es en este territorio situado entre los deseos, las limitaciones y las oportunidades, en el que se ha

movido la URJC desde su creación. Y naturalmente, esto ha propiciado una casuística muy particular, en gran medida derivada de la inmadurez de los servicios de apoyo, y de la rápida evolución a la que se ven obligados todas las estructuras universitarias, incluido, claro está, el cuerpo de profesores. Poco a poco, según la URJC va alcanzando la madurez, estas circunstancias particulares van disminuyendo, y la situación se va asimilando con la de cualquier otra Universidad de nuestro entorno socio-económico.

En cualquier caso, en este entorno particular hay que situar las dos actividades fundamentales del profesorado en la Universidad: la búsqueda de nuevos conocimientos (investigación) y la transmisión de estos conocimientos a la sociedad (docencia y transferencia de tecnología). A ellas hay que unir una tercera actividad que, aunque podría pensarse como marginal, en la práctica requiere de una gran cantidad de tiempo y recursos: la gestión.

Los tres aspectos se verán reflejados en el curriculum vitae del autor de esta memoria, y los tres aparecerán también, en distinta medida, en los proyectos docente e investigador que se incluyen más adelante. Todos ellos se han visto conformados por la situación de la Universidad Rey Juan Carlos (donde ha desarrollado su carrera como Titular de Universidad), y la de la Universidad Carlos III, donde se desarrollaron los primeros años de su carrera profesional, en una época en que también era una Universidad en formación.

## 2.2. La población Universitaria

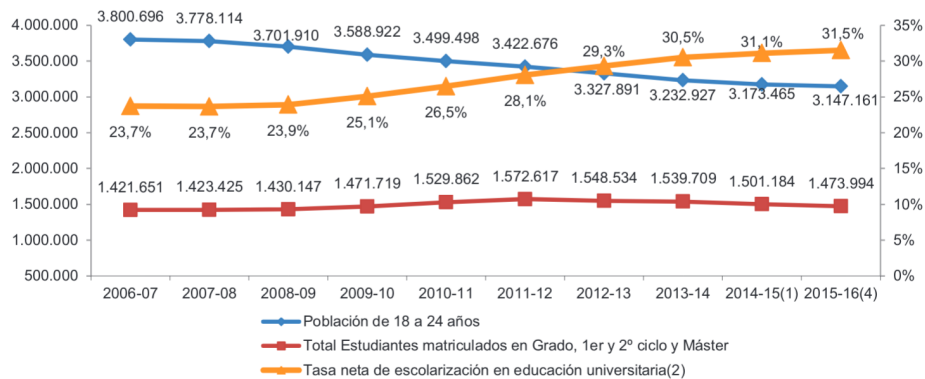


Figura 2.1: Evolución de la población de 18 a 24 años y de los matriculados en estudios universitarios. Fuente: Datos y cifras del sistema universitario español [3].

Durante los años 2006 a 2016 la población universitaria española ha pasado de ser un 23,7% de toda la población entre 18 y 24 años, a ser un 31,5%, casi una tercera parte (figura 2.1). Esta evolución es debida en gran medida a la demografía, dado que el número de personas entre 18 y 24 años ha disminuido desde algo más de 3.800.000 hasta 3.147.000. Por ello el aumento relativo no se ha reflejado en una mayor población de alumnos matriculados, que ha permanecido estable en torno a un millón y medio de personas. La “presión social cuantitativa”, por tanto, sobre la Universidad española, se ha mantenido también bastante constante durante estos años.

Este ratio de población matriculada en la Universidad no es constante entre las Comunidades Autónomas. En el caso de Madrid, sube desde el mencionado 31,5% hasta un 46,8% en 2016, con un total de 265.000 alumnos matriculados. Esto es debido en parte a la recepción de alumnos de otras comunidades y extranjeros, pero también a una mayor tasa de matriculación sobre la población de la Comunidad. Esto supone un hecho diferencial madrileño, pues ninguna otra comunidad llega al 38%, que se refleja también en un mayor número de Universidades (datos recogidos de [3]).

Durante estos 10 últimos años, mientras la población matriculada evolucionaba como se ha descrito, ha tenido lugar la sustitución de los títulos anteriores al modelo actual de grados (modelo “Bolonia”) por los grados (figura 2.2). Lo que sin duda ha supuesto un estrés extra para todas las estructuras Universitarias.

Pero este no ha sido el único gran cambio que ha experimentado la Universidad durante esta última década: uno quizás menos conocido, pero seguramente más relevante, es el desplazamiento de alumnos hacia la rama de Ciencias de la Salud desde prácticamente todas las demás ramas (figura 2.3). Este cambio

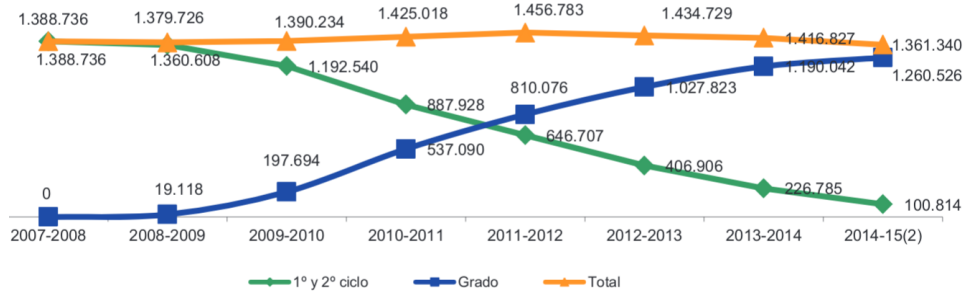


Figura 2.2: Evolución de alumnos matriculados en títulos anteriores al modelo actual de grados, y de matriculados en grados. Fuente: [3].

es particularmente intenso en cuanto a los alumnos matriculados en Ingeniería y Arquitectura, que han pasado de ser un 26% del total en 2005 a ser un 20% en 2015 (aproximadamente un 20% de descenso). En otras palabras, los estudios de Ingeniería y Arquitectura, donde se inscriben las plazas que nos ocupan en esta memoria, son percibidos como menos atractivos por los alumnos que eligen titulación.

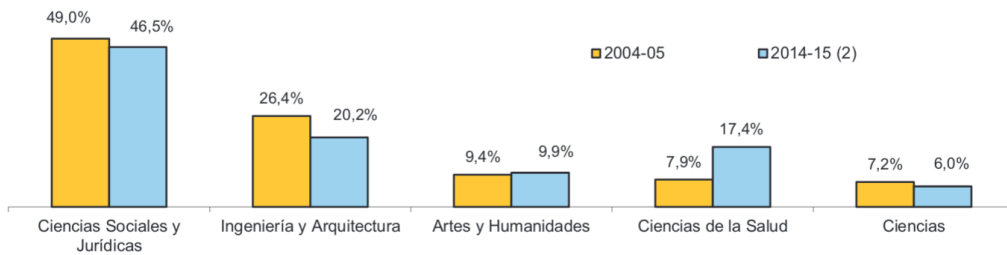


Figura 2.3: Alumnos matriculados por rama de enseñanza (2005 y 2015). Fuente: [3].

Si atendemos a la demografía de los estudiantes actualmente en la Universidad, uno de los hechos que desataca es su distribución por género. En líneas generales, la paridad entre los alumnos está bien consolidada, con un porcentaje de mujeres mayor que el de hombres, tanto en alumnos matriculados como egresados. Sin embargo, el área de estudios que nos ocupa en esta memoria, Ingeniería y Arquitectura, está aún lejos de esos números. En ella, el número de mujeres está sólo en torno a una cuarta parte del total (figura 2.4). Los números son tan claros que no es preciso resaltarlos. Es preciso entender primero, y actuar después, si queremos que las mujeres se incorporen en mayor número a los estudios técnicos.

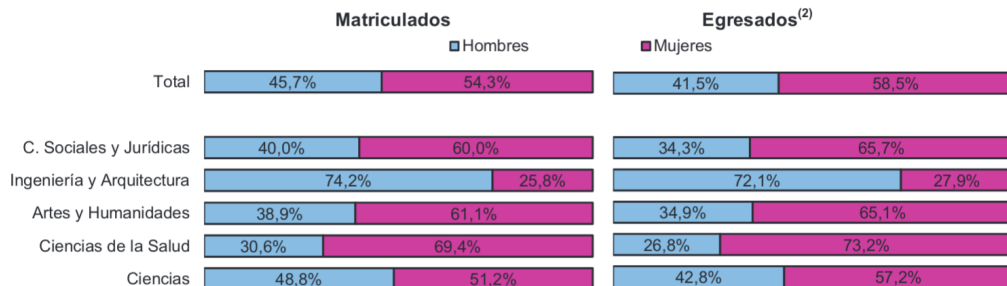


Figura 2.4: Alumnos matriculados y egresados por género (curso 2014-2015). Fuente: [3].

Otro dato que es preciso tener en cuenta es la composición por edad de los alumnos. Si atendemos a las edades de los egresados (figura 2.5) se observa cómo una gran parte de los alumnos que egresan de la Universidad son mayores de 25 años: bastantes más de la mitad, en el caso de Ingeniería y Arquitectura.

Entre los estudiantes de esta rama, más de un 21 %, de hecho, tienen más de 30 años. Si consideramos que los alumnos que se matriculasen al terminar el Bachillerato podrían haber egresado con unos 22 o 23 años, esto supone que muchos alumnos, o bien tardan mucho más de lo planificado en completar sus grados, o bien se incorporan a la Universidad mucho más tarde. Habitualmente estos alumnos tienen necesidades específicas que habrá que considerar.

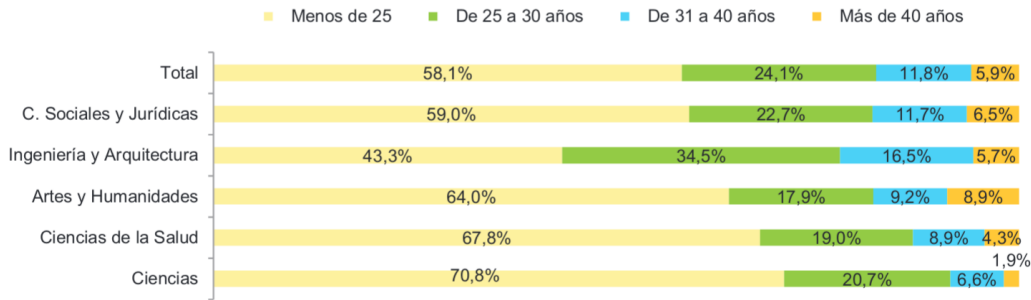


Figura 2.5: Alumnos egresados por edad (2015). Fuente: [3].

Si nos centramos por un momento en las calificaciones que obtienen los alumnos mientras están en la Universidad, la rama de Ingeniería y Arquitectura presenta también características claramente particulares. Como se muestra en la figura 2.6, las notas medias de los egresados de esta rama son sensiblemente inferiores: más de un 60 % reciben menos de un 7 de calificación, frente a un 44 % en Ciencias, la siguiente rama con menos nota.

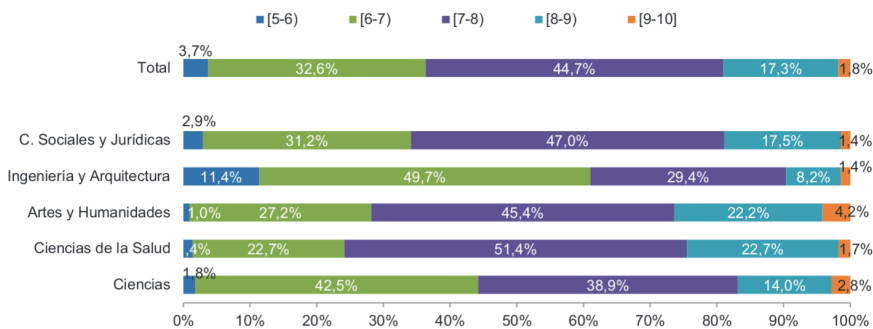


Figura 2.6: Alumnos egresados por nota media de su grado (egresados en 2015). Fuente: [3].

Aún así, esta situación no causa que los alumnos abandonen su carrera en mayor medida que otros: en Ingeniería y Arquitectura abandonaron en primer año un 24,5 % de los alumnos, muy cerca del 22,5 % de media de todos los alumnos. Tampoco supone un periodo más largo hasta completar los estudios: en Ingeniería y Arquitectura los alumnos tardan en terminar los grados de 4 años una media de 4,27 años, frente a una media de 4,20 para todos los alumnos.

Por lo tanto, podría decirse que las ingenierías parecen carreras donde en general se evalúa a los alumnos con notas más bajas, aunque esto no supone que se desanimen y dejen la carrera en mayor medida, ni que tarden más en terminarla.

### 2.3. Presupuestos de las Universidades Públicas

Desde el punto presupuestario, las cuentas mostradas en la tabla 2.3 son un buen referente del descenso presupuestario que ha afectado a las Universidades Públicas durante los últimos años. De hecho, es muy posible que los datos entre 2009 y 2016 sean aún muchos más claros, pues esas fechas cubrirían mejor el periodo de estricta reducción presupuestaria. Atendiendo sólo a los totales de ingresos y gastos, puede verse

Capítulo contable	2010	2013
1. Gastos de personal	5.962	5.625
2. Gastos en bienes y servicios	1.445	1.265
6. Inversiones reales	2.046	1.383
Total gastos	10.208	8.998
31. Precios públicos	1.751	2.078
44. Transf. corrientes AGE	208	164
45. Transf. corrientes Com. Autónomas	6.171	5.172
70. Transf. capital AGE	396	228
75. Transf. capital Com. Autónomas	675	715
Total ingresos	10.465	9.121

Tabla 2.1: Ejecución del presupuesto de ingresos y gastos de Universidades Públicas, en 2010 y 2013, en millones de euros. Fuente: [3].

una reducción en torno al 10 %, y esto en un periodo en que, como ya hemos visto, el número de alumnos atendidos ha permanecido bastante constante. Puede verse también la transferencia de ingresos desde los recibidos de las Administraciones Públicas, y sobre de las Comunidades Autónomas, a los precios públicos (fundamentalmente, ingresos por matrícula). Estos últimos se han incrementado en más de un 50 % (250 millones de euros), y aún así este incremento no ha equilibrado la reducción de ingresos corrientes de Comunidades Autónomas, de casi un 20 % y mil millones de euros en estos cuatro años.

En el caso de Madrid, el peso de la financiación por matrícula es en general mayor que en otras Comunidades Autónomas, por el mayor precio del crédito con respecto a la mayoría de las demás comunidades, y especialmente en segunda y tercera matrícula, como muestra la figura 2.7.

Por lo tanto, se puede decir que durante los últimos años las Universidades han estado sometidas a una gran tensión presupuestaria, con reducciones generalizadas que se han reflejado en todas las partidas de gastos. Y que durante este mismo periodo se ha producido una transferencia muy clara de ingresos recibidos de los presupuestos públicos generales a ingresos debidos a matrícula (esto es, pagados directamente por los estudiantes).

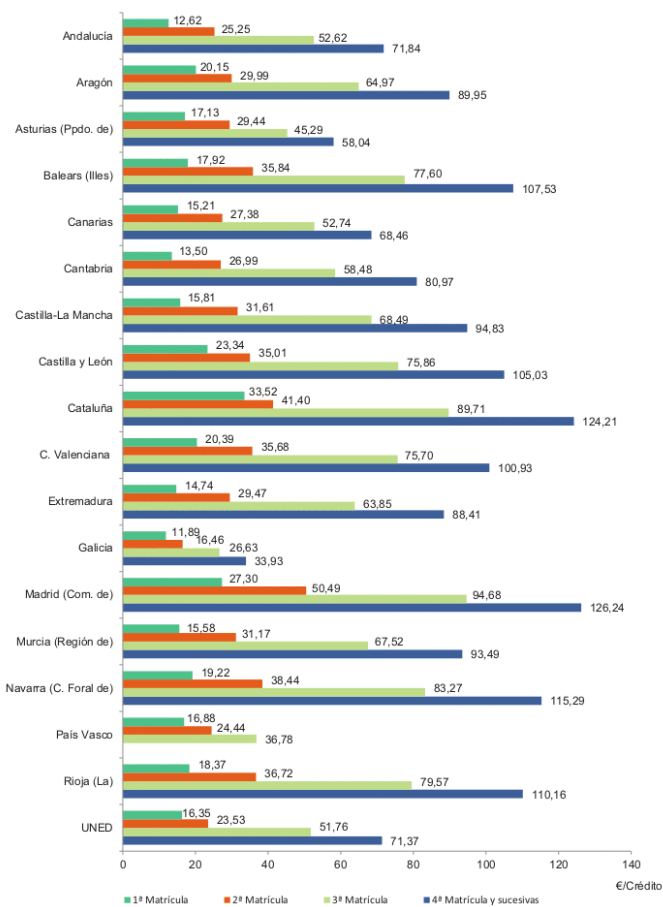


Figura 2.7: Precios público del crédito de grado, por Comunidad Autónoma. Fuente: [3].



## Capítulo 3

# La Universidad Rey Juan Carlos

La Universidad Rey Juan Carlos fue creada en el año 1996 por la Ley 7/1996 de 8 de julio, por iniciativa de la Comunidad Autónoma de Madrid (Boletín Oficial de la Comunidad de Madrid, número 169, de 17 de julio de 1996). En el artículo segundo de dicha ley se establece que la Universidad tendrá una estructura departamental y que constaría inicialmente de tres centros: La Facultad de Ciencias Jurídicas y Sociales, la Escuela Superior de Ciencias Experimentales e Ingeniería y la Facultad de Ciencias de la Salud. La disposición transitoria cuarta estableció así mismo, que el Consejo de Gobierno de la Comunidad autónoma adscribiría el Centro de Estudios Ramón Carande a la Universidad Rey Juan Carlos mediante un convenio.

Con el tiempo, la estructura de la Universidad ha variado (quizás demasiado, teniendo en cuenta su corta historia), hasta estar compuesta hoy por seis centros de enseñanza de grado y dos de postgrado, y varios centros adscritos o asociados. Físicamente, la URJC cuenta con cinco campus (Alcorcón, Aranjuez, Fuenlabrada, Madrid y Móstoles), todos ellos en la Comunidad de Madrid, al que se podría unir como un sexto el campus virtual<sup>1</sup>. Móstoles, Fuenlabrada, Aranjuez y Alcorcón son ayuntamientos situados al sur del área metropolitana de Madrid, y el campus de Madrid se sitúa fundamentalmente en Vicálvaro un distrito situado al oeste de la capital. La distancia geográfica entre los campus más alejados está en torno a los 40 o 50 kilómetros (entre Aranjuez y el campus de Madrid o Alcorcón, por ejemplo). El rectorado de la URJC se encuentra en el campus de Móstoles.

Los centros de enseñanza de la son<sup>2</sup>:

- Escuela Superior de Ciencias Experimentales y Tecnología (ESCET), con sede en el campus de Móstoles. En ella se imparten grados técnicos y experimentales, como Ciencias e Ingeniería Ambiental, Biología, y Ciencias Experimentales, y los que se podrían considerar como del ámbito de la Ingeniería Industrial, como Ingeniería de la Energía y de Materiales, de Tecnologías y de Organización Industrial, de Ingeniería Mecánica y Química.
- Facultad de Ciencias de la Salud (FCS), con sede en el campus de Alcorcón. En ella se imparten grados del área de salud, como Medicina, Enfermería, Fisioterapia, Terapia Ocupacional, Odontología y Psicología.
- Facultad de Ciencias Jurídicas y Sociales (FCJS), con sede en el campus de Madrid. Forma parte de la Universidad Rey Juan Carlos desde el curso 1998-99, aunque su integración se realizó gradualmente y terminó completamente hacia 2005. En este centro se imparten titulaciones jurídicas y económicas como Administración y Dirección de Empresas, Derecho, Economía, Turismo, Relaciones Laborales y Recursos Humanos, Contabilidad y Finanzas, Ciencia Política y Gestión Pública, y Criminología.
- Facultad de Ciencias de la Información (FCCOM), con sede en el campus de Fuenlabrada. En ella se imparten grados de Periodismo, Comunicación Audiovisual y Publicidad y Relaciones Públicas.
- Escuela Técnica Superior de Ingeniería Informática (ETSII), con sede en el campus de Móstoles. En ella se imparten grados como Ingeniería Informática, del Software, de Computadores, Diseño y

---

<sup>1</sup>Estructura en campus de la URJC: <https://www.urjc.es/universidad/campus>

<sup>2</sup>Centros de la URJC: <https://www.urjc.es/universidad/facultades>

Desarrollo de Videojuegos y Matemáticas. En su día esta escuela se creó a partir de titulaciones anteriormente encuadradas en la ESCET.

- Escuela Técnica Superior de Ingeniería de Telecomunicación (ETSIT), con sede en el campus de Fuenlabrada. En ella se imparten grados que se podrían considerar del ámbito de la Ingeniería de Telecomunicación, como Tecnologías de la Telecomunicación, Telemática, Sistemas de Telecomunicación o Sistemas Audiovisuales y Multimedia, además de Ingeniería Aeroespacial y Biomédica.
- Escuela Internacional de Doctorado, con sede en Móstoles.
- Escuela de Másteres Oficiales, con sede también en Móstoles.

La mayoría de los centros están encargados de la organización de títulos en otros campus, además de aquel en el que tienen la sede, lo que complica la gestión, pero permite también una mayor diversidad geográfica de la oferta docente de la Universidad.

Grados	61
Grados en inglés	9
Grados semipresenciales	7
Grados a distancia	2
Dobles grados	81
Másteres oficiales	76
Másteres de títulos propios	109
Total de titulaciones	345

Tabla 3.1: Títulos ofrecidos por la URJC (2017).

La tabla 3.1 muestra algunos números que dan idea de la dimensión docente de la URJC. En ella se puede apreciar la importancia que tienen los dobles grados en esta Universidad, y la gran oferta de grados que la convierten en una Universidad generalista.

INDICADOR	2013-14	2014-15	2015-16	2016-17	% Variación
Oferta	8307	9024	8881	9398	5.82%
Preinscritos 1ª Opción	10601	10223	10892	12272	12.67%
Tasa de demanda (Preinscritos 1ª Opción/Oferta)	1.28	1.13	1.23	1.31	6.47%
Matriculados Nuevo Ingreso	8441	8238	8139	8970	10.21%
Matriculados 1ª Opción	3912	4302	4246	4358	2.64%
Matriculados 1ª Opción/Nuevo Ingreso	46.35%	52.22%	52.17%	48.58%	-6.87%
Matriculados nuevo ingreso nacionalidad extranjera	753	992	796	885	11.18%
Matriculados nuevo ingreso nacionalidad extranjera (%)	8.92%	12.04%	9.78%	9.87%	0.88%

Tabla 3.2: Oferta, demanda y matrícula en grados de la URJC.

La URJC se encuentra aún en una fase de franco crecimiento. Tanto la oferta como la demanda y los alumnos matriculados están creciendo (tabla 3.2). Esto augura que durante los próximos años, si no reduce su oferta, la Universidad continuará creciendo en alumnos, aunque no se abrieran nuevos títulos. Viendo la tabla 3.3 puede entenderse que este aumento se debe sobre todo a las matrículas de grado, que han crecido en más de 6,000 durante los últimos cuatro años, una vez se descuenta la disminución lógica en los títulos antiguos, que están en proceso de extinción. La matrícula de postgrado también ha aumentado de forma muy importante: el aumento conjunto de doctorado, máster y títulos propios aportan unas 3,000 nuevas matrículas.

Total de Matriculados	2012-13	2013-14	2014-15	2015-16
Grado	23.876	28.691	31.624	34.427
Primer y Segundo Ciclo	5.396	2.992	1.445	1.081
Máster	2.980	2.861	3.206	4.460
Doctorado	1.021	1.258	1.509	1.694
Títulos Propios	2.128	2.599	2.668	3.208
Universidad de Mayores	567	597	542	588

Tabla 3.3: Matrícula por tipo de título (URJC).

La actividad docente que da servicio a todos estos alumnos, más la correspondiente actividad investigadora y de otros tipos, se mantiene con la siguiente estructura de personal de personal que se ve en las figuras 3.1 (personal de administración y servicios) y 3.2 (personal docente e investigador).

Estas figuras muestran un ratio alumnos / PDI / PAS, para el año 2016, de 34.427/1.569/635. Esto es, el ratio alumnos/PDI es de aproximadamente 22, el de alumnos/PAS de 54, y el de PDI/PAS de 2,5. Puede decirse con un algo grado de seguridad que los ratios de alumnos a PDI y PAS son extraordinariamente altos. Como referencia, la ratio alumnos/PDI para todas las Universidades Públicas presenciales españolas era en 2014 de 14,1, y la de PDI/PAS de 1,5, y de 14,1 y 1,4 para la Comunidad de Madrid (datos ofrecidos por el Ministerio de Educación, Cultura y Deporte en el Anuario de indicadores universitarios 2015).

Esta discrepancia entre los ratios en la URJC y los datos a nivel nacional o de Comunidad Autónoma es debido a que el gran crecimiento en alumnos que ha experimentado durante los últimos años no se ha correspondido con un crecimiento en personal, debido entre otras causas a la congelación de la oferta de empleo público durante este periodo.

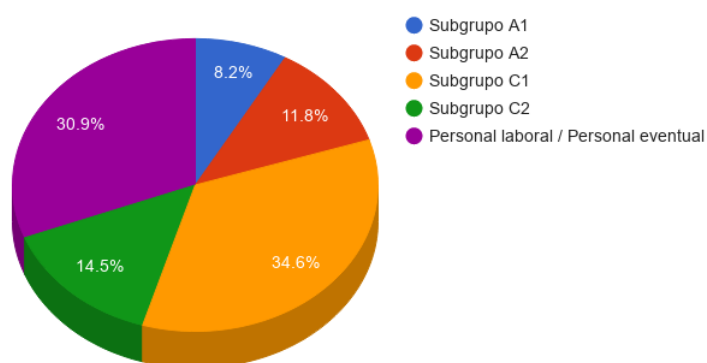
Por otro lado, centrándonos en el profesorado, un número que habitualmente se considera un buen predictor de la experiencia investigadora de este personal es el número de sexenios (periodos de investigación reconocidos) acumulado. El número de profesores con al menos un sexenio se considera, por otro lado, un buen indicador con respecto de la madurez investigadora del personal. Como se puede ver en la tabla 3.4 ambos indicadores han ido creciendo de forma continua durante los últimos 12 años. En este periodo ambos se han multiplicado aproximadamente por cuatro, lo que es consistente con la maduración investigadora de una gran parte del profesorado de la Universidad, que por edad, simplemente no habían tenido tiempo de conseguir sus sexenios. De esta forma, la investigación ha pasado de ser una actividad relativamente minoritaria, si se comparaba con otras Universidades Públicas, en algo mucho más frecuente, ya bastante en línea con otras Universidades de su entorno. Aunque también hay que tener en cuenta que el ratio de sexenios/profesores es todavía demasiado bajo.

La evolución de los proyectos internacionales (tabla 3.5), sin embargo, no se corresponde con este crecimiento. Tanto por volumen facturado por este concepto, como por número de proyectos, la URJC está aún muy por detrás del lugar en que podría estar.

Nº sexenios	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016
Total profesores con sexenios	158	200	230	270	308	368	384	425	448	469	504	558
Nº total sexenios	271	334	384	457	515	620	667	760	817	899	991	1.125

Tabla 3.4: Periodos de investigación reconocidos (sexenios) del PDI (URJC).

Por último, los presupuestos de la URJC se muestran en las tablas 3.6 (ingresos) y 3.7 (gastos). Los números anteriormente mostrados sobre el personal contratado en la Universidad explican las relativamente bajas partidas en el capítulo de gastos de personal, y el alto número de matrículas, los relativamente



CATEGORIA	Nº	Funcionario de Carrera	Funcionario interino	Funcionario Eventual	Ratio Mujer:Hombre
Subgrupo A1	52	49		3	60:40
Subgrupo A2	75	74		1	56:44
Subgrupo C1	220	212	8		61:39
Subgrupo C2	92	64	27	1	58:42
Personal laboral / Personal eventual	196				55:45
<b>TOTAL</b>	<b>635</b>	<b>399</b>	<b>35</b>	<b>5</b>	<b>58:42</b>

Figura 3.1: Estructura del personal de administración y servicios (URJC, diciembre de 2016).

altos capítulos de tasas y precios públicos. En líneas generales, la comparación de estos números con los de otras Universidades de su entorno muestra cómo la URJC tiene un presupuesto bastante menor de lo que le correspondería por su número de alumnos. De nuevo, esta situación es fruto del crecimiento en alumnos que ha experimentado durante los últimos años, mientras los presupuestos no crecían a la misma velocidad.

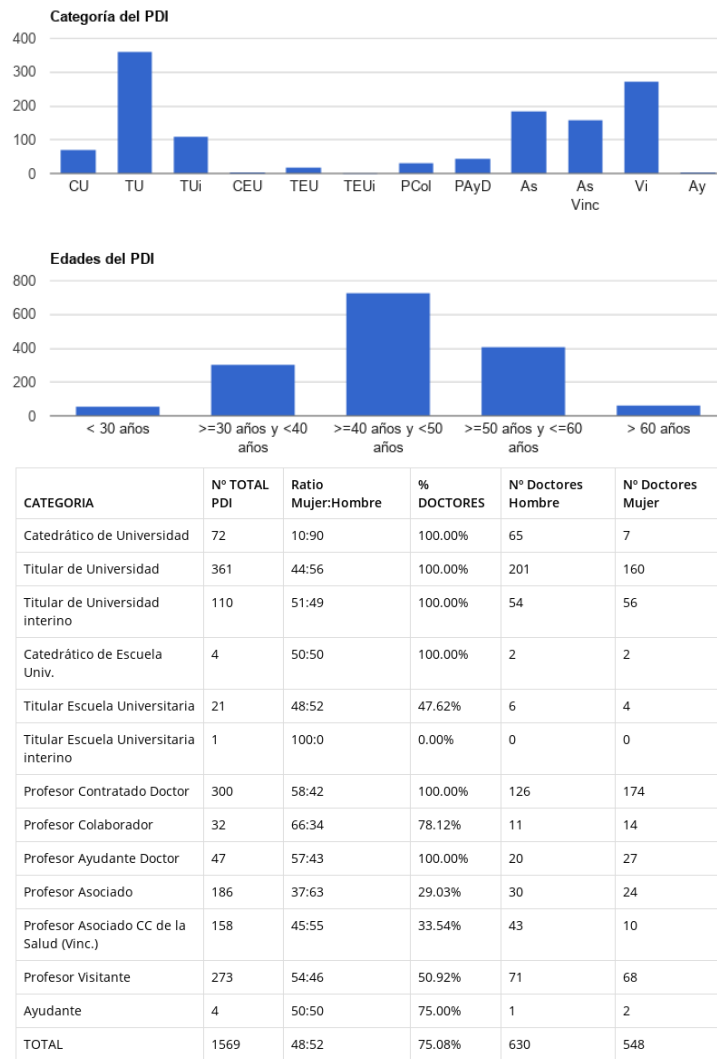


Figura 3.2: Estructura del personal docente y de investigación (URJC, diciembre de 2016).

Proyectos Internacionales	2012	2013	2014	2015
Proyectos 7PM	4	8	4	1
Importe	2.134.634€	2.471.381€	1.670.880€	32.000€
Proyectos H2020	-	-	1	4
Importe	-	-	108.462€	719.720€
Otros proyectos internacionales	1	2	2	3
Importe	89.736€	100.419€	218.554€	37.498€
<b>Total proyectos internacionales</b>	<b>5</b>	<b>10</b>	<b>7</b>	<b>8</b>
<b>Importe total de proyectos internacionales</b>	<b>2.224.370€</b>	<b>2.571.800€</b>	<b>1.997.896€</b>	<b>789.218€</b>

Tabla 3.5: Proyectos internacionales (URJC).

<b>CAPÍTULO</b>	<b>2016</b>	<b>2017</b>	<b>Incremento sobre 2016</b>
<b>III Tasas, precios públicos y otros</b>	50.802.551	47.239.914	-7,01%
<b>IV Transferencias corrientes</b>	69.143.708	77.911.454	12,68%
<b>V Ingresos patrimoniales</b>	420.000	375.000	-10,71%
<b>VII Transferencias de capital</b>	1.850.000	2.793.632	51,01%
<b>TOTAL</b>	<b>122.216.259</b>	<b>128.320.000</b>	<b>4,99%</b>

Tabla 3.6: Presupuesto de ingresos (URJC, ejercicio 2017).

<b>CAPÍTULO</b>	<b>2016</b>	<b>2017</b>	<b>Incremento sobre 2016</b>
<b>I Gastos de personal</b>	87.209.232	88.000.000	0,91%
<b>II Gastos corrientes en bienes y servicios</b>	21.213.806	24.464.206	15,32%
<b>III Gastos financieros</b>	100.000	100.000	0,00%
<b>IV Transferencias corrientes</b>	4.277.750	4.642.708	8,53%
<b>VI Inversiones reales</b>	9.043.240	10.740.855	18,77%
<b>VIII Activos financieros</b>	210.000	210.000	0,00%
<b>IX Pasivos financieros</b>	162.231	162.231	0,00%
<b>TOTAL</b>	<b>122.216.259</b>	<b>128.320.000</b>	<b>4,99%</b>

Tabla 3.7: Presupuesto de gastos (URJC, ejercicio 2017).

## Capítulo 4

# La Escuela y el Departamento

Después de describir la Universidad en la que se inscribe la plaza objeto de esta memoria, vamos a describir brevemente su contexto cercano: La Escuela Técnica Superior de Ingeniería de Telecomunicación y el Departamento de Teoría de la Señal y Comunicaciones y Sistemas Telemáticos y Computación.

### 4.1. La Escuela Técnica Superior de Ingeniería de Telecomunicación

La plaza objeto de concurso está inscrita en el Departamento de Teoría de la Señal Comunicaciones y Sistemas Telemáticos y Computación, que es el único departamento adscrito a la Escuela Técnica Superior de Ingeniería de Telecomunicación (ETSIT), con sede en el Campus de Fuenlabrada de la URJC. La ETSIT se encarga de gestionar la docencia en los grados y títulos de postgrado que tiene adscritos. Además gestiona, en colaboración con la Gerencia del Campus de Fuenlabrada varios laboratorios docentes y de investigación, y algunas instalaciones auxiliares.

En la ETSIT se imparten en la actualidad los siguientes programas de grado:

- Ingeniería Aeroespacial en Aeronavegación
- Ingeniería Aeroespacial en Vehículos Aeroespaciales
- Ingeniería Biomédica
- Ingeniería en Sistemas Audiovisuales y Multimedia
- Ingeniería en Sistemas de Telecomunicación
- Ingeniería en Tecnologías de la Telecomunicación



Figura 4.1: Plano del campus de Fuenlabrada, enero de 2017.

- Ingeniería en Telemática
- Doble Grado en Ingeniería en Sistemas de Telecomunicación y Administración y Dirección de Empresas
- Doble Grado en Ingeniería en Tecnologías de la Telecomunicación e Ingeniería Aeroespacial en Aeronavegación

Aunque formalmente es organizado por la Escuela de Postgrado, los profesores de la ETSIT imparten también el Máster en Ingeniería de Telecomunicación, y participan en el Programa de Doctorado en Tecnologías de la Información y las Comunicaciones organizado por la Escuela de Doctorado.

En septiembre de 2017, los alumnos matriculados en la ETSIT eran 1012. La oferta actual de la ETSIT para alumnos de nuevo ingreso es de 392. Por lo tanto, es de prever que, si se mantiene esta oferta durante los próximos cursos, la matrícula suba hasta unos 1.100 o 1.200 alumnos. Aún así, la ETSIT es una escuela pequeña, por número de matrículas, en el contexto de la URJC. Sin embargo, su actividad investigadora es muy grande para su tamaño.

La mayoría de los números de la ETSIT relacionados con PDI son muy similares a los del DTSCySTC, que se ofrecen a continuación.

## 4.2. El Departamento de Teoría de la Señal y Comunicaciones y Sistemas Telemáticos y Computación

En la actualidad, la mayoría de los profesores que realizan tareas en la ETSIT pertenecen al único Departamento adscrito a esta Escuela, el Departamento de Teoría de la Señal y Comunicaciones y Sistemas Telemáticos y Computación (DTSCySTC). En consecuencia, este departamento integra profesores de muchos de los campos necesarios para impartir los grados que se ofrecen en la ETSIT, por lo que sus profesores están adscritos a diversas áreas de conocimiento (ver tabla 4.1). El departamento cuenta en la actualidad con un personal docente formado por 73 personas, de las cuales 45 son doctores. Por dedicación, 54 tienen dedicación a tiempo completo, y 19 a tiempo parcial.

Área de conocimiento	Número
Teoría de la Señal y Comunicaciones	31
Ingeniería Telemática	19
Ingeniería Aeroespacial	7
Lengua Española	5
Estadística e Investigación Operativa	3
Electrónica	3
Literatura Española	2
Ciencias de la Computación e Inteligencia Artificial	2
Arquitectura y Tecnología de Computadores	1
<b>Total</b>	<b>73</b>

Tabla 4.1: Personal docente del DTSCySTC por adscripción a áreas de conocimiento. Datos de noviembre de 2017.

La composición por categorías profesionales muestra también una gran variedad, aunque sobre todo se ve reflejada la corta historia del departamento por el número relativamente pequeño de profesores funcionarios con respecto al total (ver tabla 4.1).

Como ya se ha comentado, el personal docente del DTSCySTC imparte la mayor parte de la docencia en las titulaciones de la ETSIT, aunque hay algunas asignaturas impartidas por otros departamentos, especialmente en los grados de Ingeniería Aeroespacial y Biomédica. También imparte docencia en algunas titulaciones de otras Facultades y Escuelas, aunque pequeña medida.

Como resumen, puede decirse que el Departamento coincide, a día de hoy, en gran medida con la Escuela, y que ambos centran su actividad en el campus de Fuenlabrada. Ambos (Departamento y Escuela) tienen una estructura de personal que muestra su relativa juventud, que está empezando a madurar



#### 4.2. EL DEPARTAMENTO DE TEORÍA DE LA SEÑAL Y COMUNICACIONES Y SISTEMAS TELEMÁTICOS

Categoría	Número
Catedrático de Universidad	3
Titular de Universidad	12
Titular de Universidad interino	7
Contratado Doctor	13
Ayudante Doctor	8
Visitante	12
Asociado	18
<b>Total</b>	<b>73</b>

Tabla 4.2: Personal docente del DTSCySTC por categoría profesional. Datos de noviembre de 2017.

durante los últimos años. Y ambos están centrados en la ingenierías de los ámbitos de telecomunicación y aeroespacial, fundamentalmente.



# Bibliografía del contexto académico

- [1] Ley Orgánica de Universidades (LOU), 6/2001, de 21 de diciembre. Publicada en el Boletín Oficial del estado el 24 de diciembre, December 2001.  
<https://www.boe.es/buscar/act.php?id=BOE-A-2001-24515>.
- [2] Ley Orgánica 4/2007, de 12 de abril, por la que se modifica la Ley Orgánica de Universidades 6/2001, de 21 de diciembre. Publicada en el Boletín Oficial del estado el 13 de abril, April 2007.  
<https://www.boe.es/buscar/doc.php?id=BOE-A-2007-7786>.
- [3] Ministerio de Educación, Cultura y Deporte. Datos y cifras del sistema universitario español, 2016.  
<http://www.mecd.gob.es/dms/mecd/servicios-al-ciudadano-mecd/estadisticas/educacion/universitaria/datos-cifras/datos-y-cifras-SUE-2015-16-web-.pdf> (consultado el 23 de noviembre de 2017).



**Parte II**

**Proyecto docente**



“when we solve the many problems we face everyday,  
learning occurs”

---

Barrows, H. S. and Tamblyn, R. M.  
Problem-Based Learning: An Approach to Medical Education  
1980





## Capítulo 5

# Los estudios de Servicios y Aplicaciones Telemáticas

Las asignaturas especificadas en el perfil de la plaza que nos ocupa son Servicios y Aplicaciones Telemáticas (Grado en Ingeniería en Tecnologías de la Telecomunicación) y Aplicaciones Telemáticas (Grado en Ingeniería Telemática). Aunque cada una tiene su especificidad, para definir su contexto las vamos a considerar, por ahora, como “estudios de Servicios y Aplicaciones Telemáticas”. Más adelante explicaremos en qué se diferencian, y cómo se complementan.

Los servicios y las aplicaciones telemáticas se encuentran en la confluencia entre los grados relacionados con la Telecomunicación, y los relacionados con la Informática. De hecho, es posible una aproximación desde el mundo del diseño y la construcción de aplicaciones, o desde el mundo de las comunicaciones. La primera aproximación entiende que los servicios y las aplicaciones telemáticas son básicamente programas que “usan intensivamente las comunicaciones”. La segunda, que son la parte final de la pila de comunicaciones, donde los protocolos definidos se usan para dar servicios a los usuarios o a otras aplicaciones. Dado que las asignaturas que nos ocupan están en el ámbito de las Telecomunicaciones, se seguirá fundamentalmente la segunda aproximación, aunque nunca se perderá de vista la primera. De hecho, en ambas asignaturas, al estar situadas ya en cursos altos del grado, se tratará de ofrecer una visión holística, que permita al alumno “encajar” ambas visiones, aprovechándose de lo mejor de cada una de ellas.

Siguiendo esta idea, en este capítulo vamos a repasar primer brevemente el encaje de los conocimientos y capacidades relacionados con estas asignaturas en recomendaciones internacionales relacionadas con la informática, para a continuación centrarnos en los más propios de Telecomunicación a nivel nacional, terminando con los planes de estudios concretos de los grados que nos ocupan.

### 5.1. Curricula relacionados con la informática

Entre los curricula internacionalmente reconocidos relacionados con la informática, se han elegido los dos siguientes, por su especial impacto global:

- Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering (SE2014) [24], elaborado conjuntamente por la Association for Computing Machinery y la IEEE Computer Society.
- Curriculum Guidelines for Undergraduate Degree Programs in Information Systems (IS2010) [14], elaborado conjuntamente por la Association for Computing Machinery y la Association for Information Systems

#### 5.1.1. SE2014

SE2014 es el último producto de una ya larga historia de recomendaciones sobre planes de estudios relacionados con la informática. Ya en los años 1968 y 1979 la ACM publicó varios informes [22, 23] cuyo propósito básico era efectuar recomendaciones y establecer criterios en el diseño de planes de estudios para la formación de graduados en Informática. En el año 1983, la Computer Society del IEEE hizo público un informe con características y propósitos similares [6]. Todas estas recomendaciones se dirijan

fundamentalmente a las universidades estadounidenses que, ajustándose a ellas, buscaban la acreditación de sus programas. En 1991 ambas asociaciones publicaron ya un informe conjunto, que fue continuado con otro en 2001, y otros a partir de 2010.

El primer informe conjunto fue realizado entre 1988 y 1991 por la *Joint Curriculum Task Force*, formada por la ACM y la IEEE-CS, modelo que se ha seguido para ediciones posteriores. Este grupo de trabajo partió de las anteriores recomendaciones de ambas asociaciones, y tuvo en cuenta otros documentos que proponían cambios a los cursos introductorios propuestos en ellas [19, 20], que definían líneas para la acreditación de los programas [15, 5, 21], que proponían modelos de plan de estudios alternativos [17] o que presentaban una nueva visión de la disciplina de Informática [13]. Como resultado, este grupo presentó en el año 1991 el informe *Computing Curricula 1991* [16].

SE2014 es la última actualización de estos documentos. En esta edición se incluye una recomendación general de contenidos relevante para nuestras asignaturas:

- Software Design. Menciona entre otros muchos temas “Service-Oriented Architectures”, diseño de bases de datos, diseño de aplicaciones en red y móviles,

Sin embargo, sin duda donde mejor encajan estas asignaturas con las propuestas de SE2014 es cuando menciona (Curriculum Guideline 6) que una institución podría ofrecer cursos como “Telecommunications Software Engineering”, donde se integrasen los contenidos específicos (en este caso de aplicaciones de telecomunicación) con los genéricos de ingeniería del software que se especifican en todo el plan de estudios. Este es precisamente el enfoque que se seguirá, más adelante, cuando se diseñen las dos asignaturas. También se especifica en la “Curriculum Guideline 12” que los alumnos “deben conseguir experiencia trabajando herramientas apropiadas y actualizadas, incluso si los detalles de las herramientas no son el centro del aprendizaje”, y en la “Curriculum Guideline 14” que “el curriculum debe basarse en gran medida en el mundo real”. La aproximación práctica, basada en sistemas de uso real y actual, será exactamente la que seguirá en nuestro diseño curricular.

En cuanto a la forma de enfocar las clases, cabe destacar el seguimiento en el diseño de nuestras asignaturas de la “Curriculum Guideline 18”, que aconseja trascender el modelo de clase magistral, y considerar varias aproximaciones al proceso de aprendizaje, entre las que menciona el aprendizaje basado en problemas, el aprendizaje justo a tiempo (“just-in-time”), el aprendizaje mediante los fallos, y el aprendizaje apoyado por la tecnología (mencionando por ejemplo los recursos educativos abiertos, y los sistemas que permiten el trabajo en colaboración).

En general, se puede decir que en el diseño de ambas asignaturas se han tenido en cuenta casi todas las recomendaciones del SE2014, siguiendo la aproximación de “demos la oportunidad a los alumnos de experimentar con buenas prácticas del mundo del desarrollo software”.

### 5.1.2. IS2010

En IS2010, los siguientes temas están directamente relacionados con las asignaturas que nos ocupan:

- IS 2010.1: Foundations of Information Systems. Tema “The Internet and WWW”, que incluye entre otros asuntos Internet e intranet (como conceptos), y sobre todo tecnologías utilizadas (con mención de wikis, tags, blogs, netcasts, self-publishing) y “nuevas formas de colaboración” (redes sociales, etc.). También son relevantes los temas “Business Intelligence” (como “minería de datos, texto y web) y “Application Systems”.
- IS 2010.2: Data and Information Management. Fundamentalmente en lo que tiene que ver con la organización de bases de datos, uso de bases de datos desde aplicaciones y arquitectura de datos e información.
- IS 2010.3: Enterprise Architecture. Se recomiendan aspectos como arquitectura de datos e información, gestión de contenidos, computación distribuida, middleware. Se menciona específicamente que esta asignatura puede usarse para introducir tecnologías como Web 2.0, SOA o SaaS.
- IS 2010.4: IT Infrastructure. Se explica cómo el curso ha de estar fuertemente basado en soluciones basadas en Internet, y se menciona explícitamente el uso de ejemplos prácticos para explicar cómo los protocolos, el uso de virtualización y computación en la nube y de la computación como servicio.

Otros temas de IS2010 tocan temas también relevantes para nuestras asignaturas, aunque bien de forma más general (como seguridad, o gestión de proyectos software y metodologías de desarrollo).

Pero donde más reflejo se puede encontrar de las asignaturas que nos ocupan es en una asignatura que se propone como optativa: “IS Innovation and New Technologies”. En ella se propone directamente el estudio de las tecnologías que permiten construir nuevos sistemas innovadores, como comunidades digitales, incluyendo cómo la arquitectura web puede usarse para mejorar la compartición de información y la funcionalidad. Entre sus temas se mencionan de forma prominente:

- Importancia estratégica del web como una plataforma: servicios web, redes sociales, etc.
- Herramientas Web 2.0 (RSS, wikis, blogs, mashups)
- Organización de la información (taxonomías, etiquetado)

Como se verá más adelante, muchos de estos conceptos constituyen el núcleo de nuestras asignaturas. Aún así, hay que tener en cuenta que el IS2010, probablemente debido a la fecha en que fue propuesto, utiliza un vocabulario e incluso algunos conceptos que necesariamente han de ser actualizados.

El enfoque que se propone para el curso resuena también muy bien con la propuesta que se ofrece más adelante para las asignaturas objeto de esta memoria: “se seleccionan temas que puedan maximizar el interés de los alumnos por los sistemas de información”, “es de importancia crítica exponer a los alumnos cómo los sistemas de información están impactando en el mundo de los negocios”, y “el curso debería incluir demostraciones prácticas y proyectos que permitan a los alumnos manejar estas herramientas digitales.

### 5.1.3. ANECA Ingeniería Informática

A nivel nacional, el documento de referencia para los estudios de informática es el Libro Blanco de ANECA sobre el Título de Grado en Ingeniería Informática [12]. En esta recomendación se mencionan, entre los contenidos específicos, los siguientes relacionados con las asignaturas que nos ocupan:

- Subcategoría 2.3: Sistemas Distribuidos.
- Subcategoría 2.3: Arquitectura e Infraestructura de Redes y Servicios Telemáticos.

Aunque ambos tienen que ver con nuestras asignaturas, sin embargo no corresponden con sus contenidos fundamentales. La primera, porque aunque las aplicaciones en las que nos centraremos son distribuidas por naturaleza, no están directamente relacionadas con muchos de los conceptos incluidos en un curso típico de Sistemas Distribuidos. La segunda, porque en líneas generales se refiere a las tecnologías necesarias para construir servicios y aplicaciones telemáticas, pero no tanto a los servicios y aplicaciones en sí mismos.

La situación cambia cuando consideramos las materias discrecionales, entre las que se encuentran:

- Bloque temático 4, “Redes y Sistemas Operativos”, en el que encontramos contenidos como Tecnologías de Red y Sistemas de Telecomunicación que, en principio, podrían incluir contenidos relacionados con nuestras asignaturas.
- Bloque temático 5, “Sistemas de Información”, con contenidos como Servicios para Web, ERP, CRM, o Comercio Electrónico, que son mucho más cercanos a las asignaturas que nos ocupan (en particular el primero).

Hay que reconocer que, en ambos casos, los nombres de los contenidos son lo suficientemente amplios para dar cabida a gran parte de los programas que se propondrán más adelante, pero también a otros.

En resumen, puede decirse, como ya se introducía anteriormente, que los contenidos de las asignaturas a las que se dedica esta memoria están abundantemente contemplados en las recomendaciones relacionadas con las titulaciones informáticas. Sin embargo, también hay que reconocer que en muchos casos esto ocurre cuando se mencionan contenidos que podrían tener otras interpretaciones, que tocan colateralmente los contenidos que propondremos, o que están propuestos como materiales optativos.

## 5.2. Currícula relacionados con las telecomunicaciones

Las principales recomendaciones consideradas con respecto a los grados relacionados con las telecomunicaciones han sido:

- Curriculum Development Guidelines. New ICT Curricula for the 21st Century: Designing Tomorrow's Education [7].
- Libro Blanco: Título de Grado en Ingeniería de Telecomunicación [10], elaborado por ANECA.
- Libro Blanco: Anexo de Estudios de Ingeniería Telemática [11].
- Orden CIN/355/2009, de 9 de febrero [9], por la que se establecen los requisitos para la verificación de los títulos universitarios oficiales que habiliten para el ejercicio de la profesión de Ingeniero de Telecomunicación.
- Orden CIN/352/2009, de 9 de febrero [8], por la que se establecen los requisitos para la verificación de los títulos universitarios oficiales que habiliten para el ejercicio de la profesión de Ingeniero Técnico de Telecomunicación.

De estos tres, el primero incluye fundamentalmente recomendaciones genéricas, realizadas por un consorcio de grandes empresas europeas, de cara al diseño de los nuevos títulos que siguieran las recomendaciones de Bolonia. Por lo tanto, tiene poco que decir sobre los contenidos específicos de asignaturas como las que nos ocupan. Aún así, se han tenido en cuenta algunos de estos aspectos genéricos, como la introducción de tecnologías con interés industrial, y el fuerte contenido práctico de los contenidos tecnológicos.

La segunda recomendación es la principal para los estudios de telecomunicación. Y entre ellos, las dos titulaciones que nos ocupan. Ingeniería de Tecnologías de la Telecomunicación sigue las recomendaciones generales para Ingeniería de Telecomunicación de esta guía. Ingeniería Telemática sigue las recomendaciones específicas para Ingeniería Telemática, así como las del anexo específico (tercera recomendación del listado anterior). Veamos cómo están representadas nuestras asignaturas en ambas.

En el Libro Blanco sobre Ingeniería de Telecomunicación se incluyen algunos contenidos relacionados con las asignaturas objeto de este proyecto docente. Entre los contenidos tecnológicos básicos:

- Comunicaciones (total de 20 ECTS). Se mencionan las Redes de Ordenadores, que son la base para nuestras asignaturas, aunque, en general, no se puede entender que cubren sus contenidos.
- Computación y Algorítmica (total de 20 ECTS). Se mencionan las Bases de Datos, que está entre los contenidos parcialmente cubiertos por nuestras asignaturas, aunque el enfoque es algo diferente.

Son mucho más aplicables los descriptores que se encuentran para una de las líneas de “Aplicaciones y metodologías de solución de sistemas”, denominada Servicios y Tecnologías de la Información (total de 6 ECTS). Menciona los Servicios Internet y los Sistemas Distribuidos. Como ya se ha comentado, los contenidos clásicos en Sistemas Distribuidos no están más que muy parcialmente cubiertos por nuestras asignaturas, pero “Servicios Internet” puede describir muy bien gran parte de sus contenidos. Además, se explica cómo según evolucionen estas tecnologías, los contenidos precisos pueden ir cambiando, siendo lo fundamental “la capacidad de relacionar tecnologías y bloques constitutivos con aplicaciones y servicios, y las habilidades relacionadas con el diseño, construcción, gestión y explotación de sistemas”. Como se verá, esto cuadra perfectamente con el espíritu de ambas asignaturas.

Más adelante las recomendaciones sobre Ingeniería de Telecomunicación introducen los contenidos de varias “Especialidades regladas”, entre ellas la Especialidad Telemática. En la tercera de sus áreas de contenidos, “Aplicaciones y Metodologías de Solución en Sistemas” (total de 42 ECTS) incluye ya varios descriptores que cubren muy claramente las asignaturas objeto de este proyecto docente: Ingeniería de Servicios de Telecomunicación, Tecnologías, Dominios de Aplicación. Como se verá más adelante, en gran medida son estos contenidos los que cubren estas asignaturas. Además, se deja la opción de que las universidades amplíen aún más los créditos para estas áreas de contenidos, si la cuenta total de créditos lo permite.

Como podría ser de esperar dadas las asignaturas a que nos referimos, el Libro Blanco sobre Ingeniería Telemática, que cubre el Grado en Ingeniería Telemática, es aún más claro en cuanto a este tipo de

contenidos. En el apartado “Tecnologías Avanzadas” tenemos el epígrafe Aplicaciones Telemáticas (total de entre 18 y 22 créditos), que incluye entre otros los siguientes descriptores: Protocolos de nivel de aplicación, Servicios de información, Web, Web semántica, Servicios Web, Modelos de interacción entre sistemas abiertos, Aplicaciones distribuidas en Internet, Tecnologías XML. Prácticamente todos estos descriptores están contemplados en el diseño de las asignaturas que presentaremos más adelante.

Por lo tanto, y como resumen, se puede ver cómo ambas asignaturas están perfectamente contempladas entre las recomendaciones de sus títulos respectivos. En lo que afecta específicamente al Grado en Ingeniería Telemática, hay un espacio amplísimo para sus contenidos, mucho más allá de los 6 créditos que tiene la asignatura “Aplicaciones Telemáticas”, por lo que está podrá ir acompañada por otras que la completarán en cuanto a contenidos (como se mostrará cuando se detallen los contenidos en asignaturas relacionadas, más adelante). En el caso de la asignatura “Servicios y Aplicaciones Telemáticas” también ocurrirá algo similar, aunque en este caso, por estar en una carrera más generalista, con menos amplitud.

Los dos últimos documentos mencionados al principio de este apartado, CIN/355/2009 y CIN/352/2009, son las Órdenes Ministeriales que detallan los requisitos que ha de cumplir un plan de estudios para habilitar para ejercer las competencias específicas de los Ingenieros de Telecomunicación y de los Ingenieros Técnicos de Telecomunicación, respectivamente. Su misión es acomodar a los nuevos planes de estudios las competencias que estaban descritas en función de los títulos antiguos (de cinco y tres años de duración, respectivamente). Más adelante se explicará cómo los títulos de la URJC en los que se enmarcan nuestras asignaturas cumplen con esta normativa.

### 5.3. Las titulaciones de Ingeniería de Telecomunicación en la URJC

En esta sección vamos a revisar brevemente las titulaciones “descendientes” de la Ingeniería de Telecomunicación que ofrece la URJC, ya que son el contexto cercano de las asignaturas que nos ocupan. Para cada una de ellas, dedicaremos especial atención a las asignaturas similares a ellas.

En la actualidad, la ETSIT de la URJC ofrece las siguientes titulaciones que podrían considerarse descendientes de la Ingeniería de Telecomunicación:

- Grado en Ingeniería en Tecnologías de la Telecomunicación (GITT). Es el grado más generalista, y el que se podría considerar hasta cierto punto heredero el antiguo título de Ingeniería Superior en Telecomunicación (si bien tiene menos contenidos, y ha de ser completado con un máster para llegar a las competencias profesionales que otorgaba éste).
- Grado en Ingeniería Telemática (GIT). Puede considerarse una evolución de la antigua Ingeniería Técnica Telemática, o de la especialidad en Telemática de la Ingeniería Superior de Telecomunicación.
- Grado en Ingeniería en Sistemas de Telecomunicación (GIST). Puede considerarse una evolución de la antigua Ingeniería Técnica en Sistemas de Telecomunicación, o de la especialidad en Sistemas de Telecomunicación de la Ingeniería Superior de Telecomunicación.
- Grado en Ingeniería en Sistemas Audiovisuales y Multimedia (GISAM). Hasta cierto punto heredero de la Ingeniería Técnica de Imagen y Sonido, o la especialidad en Imagen y Sonido de la Ingeniería Superior de Telecomunicación.

Todas ellas tienen una duración de 240 créditos ECTS (4 cursos académicos completos), y están diseñados de forma que cumplen con la Orden Ministerial CIN/352/2009, y por tanto habilitan como Ingeniero Técnico de Telecomunicación. Además, el Grado en Ingeniería en Tecnologías de la Telecomunicación está diseñado de forma que cumple las condiciones especificadas en el apartado 4.2.2 de la Orden Ministerial CIN/355/2009. Esto permite que, cursando un máster adicional de 90 ECTS, adecuadamente diseñado, se cumplan las condiciones establecidas por la mencionada CIN/355/2009 que habilitan como Ingeniero de Telecomunicación. La URJC ofrece un máster que cumple esta característica: el Máster en Ingeniería de Telecomunicación.

Adicionalmente, los demás grados (GIT, GIST y GISAM) podrán matricularse en másteres de ese tipo si se cursan previamente ciertos complementos de formación (entre 12 y 30 ECTS, según el título), y al terminarlo cumplir también los requisitos de habilitación como Ingeniero de Telecomunicación especificados en CIN/355/2009.

Todos los grados mencionados comparten una cierta estructura común. Aunque los nombres pueden variar, sus contenidos son muy similares, y en general son convalidables entre las cuatro titulaciones:

- Formación básica común (24 ECTS). Cuatro asignaturas, con temario similar en todos los grados: Informática (normalmente introducción a la programación), Expresión Oral y Escrita, Normativa y Deontología en Telecomunicaciones, e Idioma Moderno (inglés).
- Formación básica de rama (48 ECTS). Ocho asignaturas, con temario similar en todos los grados: Álgebra, Cálculo, Estadística, Física, Organización Empresarial, Señales y Sistemas, Campos Electromagnéticos, Programación.
- Prácticas Externas (12 ECTS o 18 ECTS).
- Trabajo Fin de Grado (12 ECTS).

En la URJC todos los títulos tienen créditos de formación básica común (FBC), para todos los títulos de la Universidad, y créditos de formación básica de rama (FBR), en el caso que nos ocupa comunes para todas las ingenierías. Por lo tanto, los créditos FBC y FBR en GITT, GIT y GIST son los mismos. Estos créditos se imparten durante primer y segundo curso, sumando 72 de los 120 ECTS de esos dos cursos. El resto de estos dos cursos está compuesto por créditos de asignaturas obligatorias (48 ECTS) que son iguales en los tres grados. Esto permite compartir grupos docentes, durante estos dos primeros cursos.

El resto de las asignaturas corresponden a créditos obligatorios y optativos aunque, en la versión actual del itinerario formativo que se está impartiendo, cada titulación sólo proporciona una opción para cada asignatura optativa, lo que las hace también obligatorias, de facto.

Por lo tanto, los programas de las dos titulaciones que nos interesarán directamente, por ser nuestras asignaturas parte de ellas (GITT y GIT) serán bastante similares, como se ve en las tablas 5.1, 5.2, 5.3 y 5.4.

Asignatura	Semestre	Créditos	GITT	GIT
Cálculo y Análisis Diferencial	1	6	FBR	FBR
Álgebra Lineal y Matemática Discreta	1	6	FBR	FBR
Fundamentos de la Programación	1	6	FBC	FBC
Organización Empresarial y Recursos Humanos	1	6	FBR	FBR
Expresión Oral y Escrita y Búsqueda de Información	1	6	FBC	FBC
Señales y Sistemas	2	6	FBR	FBR
Fundamentos Físicos de la Telecomunicación	2	6	FBR	FBR
Arquitectura de Redes de Ordenadores	2	6	OB	OB
Sistemas y Circuitos	2	6	OB	OB
Normativa y Deontología en Telecomunicaciones	2	6	FBC	FBC

Tabla 5.1: Asignaturas de primer curso de las titulaciones GITT y GIT. FBR: formación básica de rama; FBC: formación básica común; OB: obligatoria

### 5.3.1. Servicios y Aplicaciones Telemáticas dentro de sus itinerario formativo

En el caso de la asignatura **Servicios y Aplicaciones Telemáticas**, del GITT, las asignaturas previas que más influyen en ella son (véase la figura 5.1 para una representación gráfica de las relaciones entre las asignaturas):

- **Fundamentos de Programación** (semestre 1). Conceptos principales relacionados: principios de programación. Competencias específicas relacionadas:
  - CE02. Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

Asignatura	Semestre	Créditos	GITT	GIT
Estadística	3	6	FBR	FBR
Campos Electromagnéticos	3	6	FBR	FBR
Programación de Sistemas de Telecomunicación	3	6	FBR	FBR
Ampliación de Señales y Sistemas	3	6	OB	OB
Electrónica	3	6	OB	OB
Radiación y Propagación	4	6	OB	OB
Sistemas Telemáticos	4	6	OB	OB
Teoría de la Comunicación	4	6	OB	OB
Sistemas Digitales	4	6	OB	OB
Idioma Moderno	3-4	6	FBC	FBC

Tabla 5.2: Asignaturas de segundo curso de las titulaciones GITT y GIT. FBR: formación básica de rama; FBC: formación básica común; OB: obligatoria

Asignatura	Semestre	Créditos	GITT	GIT
Sistemas Operativos	5	6	OB	OB
Arquitectura de Computadores	5	6	OB	OB
Sistemas de Telecomunicación	5	6	OB	
Arquitectura de Sistemas de Telecomunicación	5	6		OB
Comunicaciones Digitales	5	6	OB	
Transmisión Digital	5	6		OB
Procesamiento Digital de la Información	5	6	OB	
Tratamiento de Información Multimedia	5	6		OB
<b>Servicios y Aplicaciones Telemáticas</b>	6	6	OB	
Servicios Telemáticos	6	6		OB
Investigación Operativa	6	6	OB	OB
Ingeniería de Sistemas Telemáticos	6	6	OB	
Ampliación de Sistemas Telemáticos	6	6		OB
Control de Calidad	6	6	OB	
Introducción a la Bioingeniería	6	6	OP	
Seguridad en Redes de Ordenadores	6	6		OP
Robótica	6	6		OP

Tabla 5.3: Asignaturas de tercer curso de las titulaciones GITT y GIT. OB: obligatoria, OP: optativa

Ingeniería de Sistemas de Información	7	6	OB	
Comunicaciones Inalámbricas	7	6	OB	
Procesamiento digital avanzado en Comunicación	7	6	OP	
Sistemas Distribuidos	7	6		OB
Laboratorio de Administración y Gestión de Redes y Sistemas	7	6		OP
Reconocimiento Académico de Créditos	7	6	OP	O
Terminales de Comunicaciones	8	6	OB	
Desarrollo de Aplicaciones Telemáticas	8	6	OP	
<b>Aplicaciones Telemáticas</b>	8	6		OB
Laboratorio de Sistemas Móviles y Ubicuos	8	6		OP
Prácticas Externas	7-8	12	OB	
Prácticas Externas	7-8	18		OB
Trabajo Fin de Grado	7-8	12	OB	OB

Tabla 5.4: Asignaturas de cuarto curso de las titulaciones GITT y GIT. OB: obligatoria, OP: optativa

- **Arquitectura de Redes de Ordenadores** (semestre 2). Conceptos principales relacionados: arquitecturas de protocolos, introducción a la arquitectura TCP/IP. Competencias específicas relacionadas:
  - CE06. Capacidad para aprender de manera autónoma nuevos conocimientos y técnicas adecuados para la concepción, el desarrollo o la explotación de sistemas y servicios de telecomunicación.
  - CE11. Capacidad de concebir, desplegar, organizar y gestionar redes, sistemas, servicios e infraestructuras de telecomunicación en contextos residenciales (hogar, ciudad y comunidades digitales), empresariales o institucionales responsabilizándose de su puesta en marcha y mejora continua, así como conocer su impacto económico y social.
  - CE17. Conocimiento y utilización de los conceptos de arquitectura de red, protocolos e interfaces de comunicaciones.
  
- **Programación de Sistemas de Telecomunicación** (semestre 3). Conceptos principales relacionados: conceptos avanzados de programación, tipos de datos, programación concurrente, programación con APIs de comunicación. Competencias específicas relacionadas:
  - CE12. Conocimiento y utilización de los fundamentos de la programación en redes, sistemas y servicios de telecomunicación.
  
- **Sistemas Telemáticos** (semestre 4). Conceptos principales relacionados: nivel de red (IP, enrutamiento, interconexión de redes), nivel de transporte (TCP, congestión), nivel de aplicación (HTTP), seguridad en redes. Competencias específicas relacionadas:
  - CE18. Capacidad de diferenciar los conceptos de redes de acceso y transporte, redes de conmutación de circuitos y de paquetes, redes fijas y móviles, así como los sistemas y aplicaciones de red distribuidos, servicios de voz, datos, audio, vídeo y servicios interactivos y multimedia.
  - CE19. Conocimiento de los métodos de interconexión de redes y enrutamiento, así como los fundamentos de la planificación, dimensionado de redes en función de parámetros de tráfico.
  
- **Sistemas Operativos** (semestre 5). Conceptos principales relacionados: conceptos de sistemas operativos (procesos, ficheros, concurrencia), intérpretes de comandos. Competencias específicas relacionadas:
  - CE34. Capacidad de programación de servicios y aplicaciones telemáticas, en red y distribuidas.
  - CE41. Capacidad de comprender, utilizar y diseñar sistemas operativos, bases de datos y sistemas de información en el ámbito de los sistemas y servicios de la telecomunicación.
  
- **Arquitectura de Computadores** (semestre 5). Conceptos principales relacionados: procesador, jerarquía de memoria, entrada/salida. Competencias específicas relacionadas:
  - CE14. Capacidad de análisis y diseño de circuitos combinacionales y secuenciales, síncronos y asíncronos, y de utilización de microprocesadores y circuitos integrados.

Hay una asignatura relacionada que se imparte concurrentemente:

- **Ingeniería de Sistemas Telemáticos** (semestre 6). Conceptos principales relacionados: tecnologías middleware, invocación de objetos remotos, sistemas peer-to-peer. Competencias específicas relacionadas:
  - CE34. Capacidad de programación de servicios y aplicaciones telemáticas, en red y distribuidas.
  - CE35. Capacidad de construir, explotar y gestionar servicios y aplicaciones de telecomunicaciones, entendidas éstas como sistemas de captación, tratamiento analógico y digital, codificación, transporte, representación, procesado, almacenamiento, reproducción, gestión y presentación de servicios audiovisuales e información multimedia.

Y hay varias asignaturas que dependen, en parte, de los contenidos de nuestra asignatura:



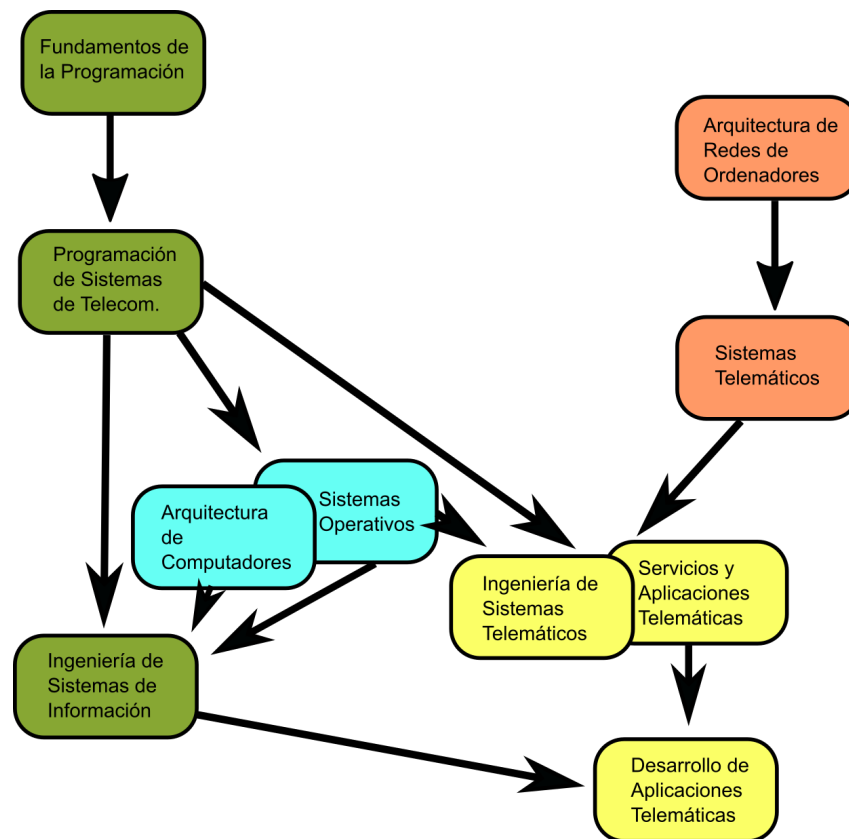


Figura 5.1: Asignaturas del plan de estudios del GITT directamente relacionadas con Servicios y Aplicaciones Telemáticas. La altura en la gráfica representa el semestre en que se imparten (arriba, primer semestre, abajo, octavo semestre). Las flechas tratan de mostrar las principales relaciones entre ellas.

- **Ingeniería de Sistemas de Información** (semestre 7). Conceptos principales relacionados: conceptos de desarrollo de software (ciclos de vida, metodologías), requisitos, diseño, pruebas, configuración. Competencias específicas relacionadas:
  - CE34. Capacidad de programación de servicios y aplicaciones telemáticas, en red y distribuidas.
  - CE41. Capacidad de comprender, utilizar y diseñar sistemas operativos, bases de datos y sistemas de información en el ámbito de los sistemas y servicios de la telecomunicación.
- **Desarrollo de Aplicaciones Telemáticas** (semestre 8). Conceptos principales relacionados: aplicaciones en el navegador, HTML5, APIs de servicios. Competencias específicas relacionadas:
  - CE34. Capacidad de programación de servicios y aplicaciones telemáticas, en red y distribuidas.
  - CE33. Capacidad de diseñar arquitecturas de redes y servicios telemáticos.

Por lo tanto la asignatura Servicios y Aplicaciones Telemáticas sirve en gran medida, en el Grado en Ingeniería en Tecnologías de la Telecomunicación, de unión de los itinerarios formativos relacionados con el desarrollo de aplicaciones (programación), sistemas (sistemas operativos, arquitectura de ordenadores) y comunicaciones. A su vez dará paso a la asignatura Desarrollo de Aplicaciones Telemáticas. Entre ambas, forman una especie de cierre de la mayoría de las asignaturas del ámbito telemático. Por este motivo, y dado el fuerte componente práctico de ambas, será convenientes utilizarlas para mostrar a los alumnos cómo las capacidades que han ido adquiriendo pueden utilizarse para realizar sistemas reales, que les permitan obtener una visión global de todo el itinerario.

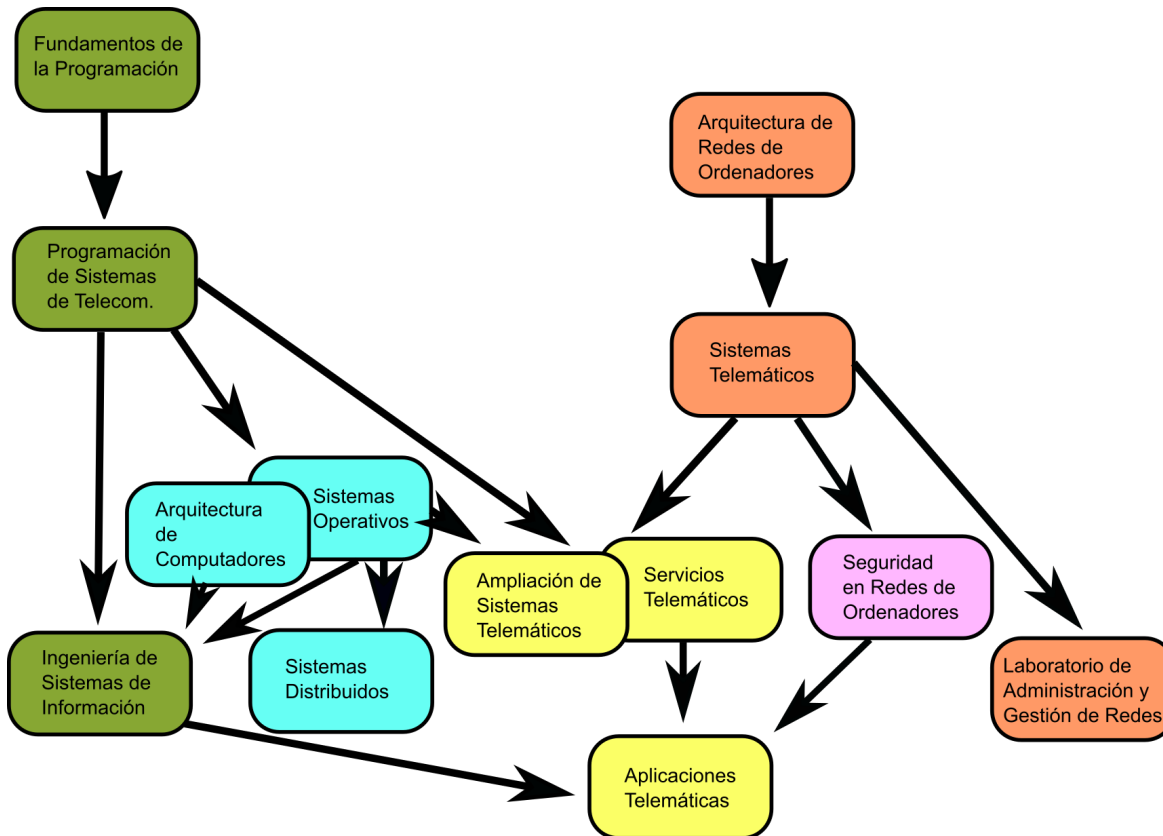


Figura 5.2: Asignaturas del plan de estudios del GIT relacionadas con Aplicaciones Telemáticas. La altura en la gráfica representa el semestre en que se imparten (arriba, primer semestre, abajo, octavo semestre). Las flechas tratan de mostrar las principales relaciones entre ellas.

### 5.3.2. Aplicaciones Telemáticas dentro de su itinerario formativo

En el caso de la asignatura **Aplicaciones Telemáticas**, del GIT, las asignaturas previas que tienen cierta influencia en ella son las mismas que en el caso de Servicios y Aplicaciones Telemáticas, del GITT, por el motivo ya explicado de que las asignaturas de primero y segundo curso son las mismas, y que Sistemas Operativos y Arquitectura de Computadores son obligatorias en ambas titulaciones. A partir de ahí, tenemos diferentes asignaturas relacionadas:

- **Servicios Telemáticos** (semestre 6). Asignatura similar a Servicios y Aplicaciones Telemáticas del GITT. Conceptos principales relacionados: aplicaciones web en el lado del servidor, APIs REST, XML, acceso a bases de datos. Competencias específicas relacionadas:
  - CE22. Capacidad para aplicar las técnicas en que se basan las redes, servicios y aplicaciones telemáticas, tales como sistemas de gestión, señalización y conmutación, encaminamiento y encaminamiento, seguridad (protocolos criptográficos, tunelado, cortafuegos, mecanismos de cobro, de autenticación y de protección de contenidos), ingeniería de tráfico (teoría de grafos, teoría de colas y teletráfico) tarificación y fiabilidad y calidad de servicio, tanto en entornos fijos, móviles, personales, locales o a gran distancia, con diferentes anchos de banda, incluyendo telefonía y datos.
  - CE24. Capacidad de describir, programar, validar y optimizar protocolos e interfaces de comunicación en los diferentes niveles de una arquitectura de redes.
- **Ampliación de Sistemas Telemáticos** (semestre 6). Asignatura similar a Ingeniería de Sistemas Telemáticos del GITT. Conceptos principales relacionados: tecnologías middleware, invocación de objetos remotos.

- **Seguridad en Redes de Ordenadores** (semestre 6). Conceptos principales relacionados: protocolos seguros, criptografía. Competencias específicas relacionadas:
  - CE22. Capacidad para aplicar las técnicas en que se basan las redes, servicios y aplicaciones telemáticas, tales como sistemas de gestión, señalización y conmutación, encaminamiento, seguridad (protocolos criptográficos, tunelado, cortafuegos, mecanismos de cobro, de autenticación y de protección de contenidos), ingeniería de tráfico (teoría de grafos, teoría de colas y teletráfico) tarificación y fiabilidad y calidad de servicio, tanto en entornos fijos, móviles, personales, locales o a gran distancia, con diferentes anchos de banda, incluyendo telefonía y datos.
- **Ingeniería de Sistemas de Información** (semestre 7). Asignatura similar a la del mismo nombre del GITT. Conceptos principales relacionados: conceptos de desarrollo de software (ciclos de vida, metodologías), requisitos, diseño, pruebas, configuración.
- **Sistemas Distribuidos** (semestre 7). Conceptos principales relacionados: sistemas operativos y algoritmos distribuidos, concurrencia distribuida, tolerancia a fallos. Competencias específicas relacionadas:
  - CE27. Capacidad de programación de servicios y aplicaciones telemáticas, en red y distribuidas.
- **Laboratorio de Administración y Gestión de Redes y Sistemas** (semestre 7). Conceptos principales relacionados: administración de redes y sistemas. Competencias específicas relacionadas:
  - CE21. Capacidad de construir, explotar y gestionar las redes, servicios, procesos y aplicaciones de telecomunicaciones, entendidas éstas como sistemas de captación, transporte, representación, procesado, almacenamiento, gestión y presentación de información multimedia, desde el punto de vista de los servicios telemáticos.
  - CE23. Capacidad de construir, explotar y gestionar servicios telemáticos utilizando herramientas analíticas de planificación, de dimensionado y de análisis.
- **Laboratorio de Sistemas Móviles y Ubicuos** (semestre 8). Dado que es concurrente a la asignatura que nos ocupa, tiene poca influencia en ella. Conceptos principales relacionados: aplicaciones para sistemas móviles y ubicuos. Competencias específicas relacionadas:
  - CE22. Capacidad para aplicar las técnicas en que se basan las redes, servicios y aplicaciones telemáticas, tales como sistemas de gestión, señalización y conmutación, encaminamiento, seguridad (protocolos criptográficos, tunelado, cortafuegos, mecanismos de cobro, de autenticación y de protección de contenidos), ingeniería de tráfico (teoría de grafos, teoría de colas y teletráfico) tarificación y fiabilidad y calidad de servicio, tanto en entornos fijos, móviles, personales, locales o a gran distancia, con diferentes anchos de banda, incluyendo telefonía y datos.
  - CE27. Capacidad de programación de servicios y aplicaciones telemáticas, en red y distribuidas.

Puede verse un gráfico mostrando las principales relaciones entre las asignaturas relacionadas del itinerario formativo del GIT en la figura 5.2.

En resumen, puede decirse que la asignatura Aplicaciones Telemáticas del GIT es una de las principales asignaturas “terminales” del itinerario formativo más directamente relacionado con la telemática (desarrollo de software, comunicaciones, sistemas). Siendo además una asignatura eminentemente práctica, será conveniente aprovechar para construir en ella sistemas que permitan al alumno poner en valor conocimientos y capacidades adquiridas durante toda la carrera, ayudándole a ver “cómo encajan las piezas”.



## Capítulo 6

# Propuesta de método didáctico

Entre las tareas del profesor, la que quizás sea más difícil pero también más apasionante, es apoyar al alumno en su proceso de aprendizaje. O, afinando un poco, ayudarlo a que los adquiera con menos esfuerzo, más aprovechamiento y mejores resultados cualitativos que si lo hiciera únicamente con sus propios medios, sin ayuda del profesor. La razón última de la utilidad del profesor en un modelo educativo cualquiera reside precisamente en este punto. Y también es el que más marca la diferencia entre un “buen profesor” y otro que no lo es tanto. En un extremo, el profesor se convierte en un elemento decorativo si no aporta nada a lo que el alumno puede realizar por su cuenta. En el otro, todo alumno desearía disponer de un profesor capaz de conseguir que aprenda sin ningún esfuerzo por su parte (idealmente, sin siquiera dedicarle unos minutos al proceso de aprendizaje). Entre estos dos extremos se sitúa siempre la labor docente. Y el trabajo del profesor consiste en con lograr que su aportación esté lo más cercana posible al segundo caso, naturalmente manteniendo los objetivos de aprendizaje que se estimen convenientes para una asignatura dada.

Teniendo esto en cuenta, en este capítulo se expone una forma de enfocar la tarea pedagógica, y un método<sup>1</sup> de enseñanza que es general, pero que aquí se centrará en su aplicación a las asignaturas de Servicios y Aplicaciones Telemáticas y Aplicaciones telemáticas. Entre otros, se abordan aspectos como el diseño general de los objetivos docentes, la forma de enfocar las clases teóricas y prácticas, los mecanismos de evaluación y la motivación y seguimiento del alumno.

Tratando de sistematizar un poco la exposición, se ha dividido el capítulo en cuatro apartados principales:

- Objetivos docentes (apartado 6.1), donde se tratará sobre el problema de decidir qué conocimientos y habilidades, tanto concretas dentro del ámbito de la asignatura, como generales, deberían adquirir los alumnos.
- Metodología docente (apartado 6.2), donde se explicará cómo se puede enfocar la labor de enseñar en los distintos tipos de conceptos (teóricos y prácticos) que componen una asignatura como la que nos ocupa.
- Evaluación (apartado 6.3), necesaria no sólo para decidir en qué medida los alumnos han conseguido los objetivos fijados previamente, sino también para conocer el impacto de la metodología docente en la consecución de estos objetivos. A partir de este conocimiento se podrán tomar decisiones sobre cambios en la metodología, e incluso sobre los propios objetivos.
- Medios necesarios (apartado 6.4), que detallará los medios necesarios para poder utilizar el modelo de apoyo al aprendizaje descrito.

### 6.1. Objetivos docentes

Las materias que han de tratarse en una asignatura vienen marcados por sus descriptores en el plan de estudios. Sin embargo, los temas concretos, su enfoque, y sobre todo su nivel de detalle y profundidad

---

<sup>1</sup> Probablemente hablar de “método” en este caso es un tanto inexacto. Más que método, lo que se expone es un conjunto de experiencias y formas depuradas a lo largo de varios años de enseñanza. Algunas tienen un fundamento pedagógico, pero otras son sólo el fruto de ensayos y errores.

vienen marcados, hoy por hoy, por el profesor. Igualmente, es el profesor el que decide, al final del periodo lectivo, qué alumnos tienen conocimientos y capacidades suficientes, y en qué grado. Para poder realizar esta evaluación es preciso haber definido a priori unos objetivos docentes, que servirán para determinar quién los ha conseguido. Estos objetivos serán útiles también para evaluar el propio desarrollo de la asignatura, y el conjunto profesor-alumnos-entorno. Al fin y al cabo, este conjunto es el que determinará completamente los resultados de cualquier proyecto docente.

Dicho de otra forma, el profesor elaborará los objetivos formativos, tanto los más generales como los de aplicación a una asignatura concreta. A partir de estos objetivos se elaborará el plan formativo que trate de conseguirlos. Como parte de este plan se seleccionarán los contenidos, el orden de presentación de los mismos, la metodología de su exposición, los medios auxiliares que se utilicen, las actividades recomendadas a los alumnos, etc.

De la comparación entre la realidad conseguida con el desarrollo de este plan y los objetivos docentes surgirá la evaluación, tanto del alumno como del conjunto profesor-alumnos-entorno. En el primer caso el resultado de la evaluación será la calificación que obtenga el alumno. En el segundo, el resultado fundamental debería ser el análisis del proceso y la identificación de las mejoras y modificaciones que se puedan hacer al plan docente para acercarse lo más posible a los objetivos, en un proceso necesariamente iterativo y, deseablemente, convergente.

En el resto de este apartado analizaremos cómo se puede desarrollar el proceso de definición de los objetivos docentes, y qué factores condicionantes hay que tener en cuenta en él. También se propondrán algunos objetivos generales deseables en las enseñanzas tecnológicas. Los objetivos concretos para una asignatura de Redes se propondrán en los capítulos 7 y 8, dedicados específicamente al plan para las asignaturas.

### 6.1.1. El proceso de definición de los objetivos

La identificación de los objetivos docentes ha de tener en cuenta al menos dos condicionantes:

- Por un lado, el profesor ha de determinar con la mayor exactitud posible los conocimientos, habilidades y experiencias que un alumno debe tener al final del proceso docente para la asignatura en cuestión. Estos vendrán dados por los descriptores del plan de estudios de la titulación para esa asignatura, matizado por lo que comúnmente sea aceptado como el corpus de la asignatura y por la evolución de la tecnología y de los conocimientos en el campo.
- Por otro, habrá que tener en cuenta el entorno, y especialmente el tipo de estudiantes, sus intereses, su formación, los medios a disposición del profesor, el resto de asignaturas que están cursando, los requisitos demandados por el mercado de trabajo, etc. Esto no necesariamente ha de tener incidencia en el nivel de los objetivos, pero quizás sí en su enfoque concreto, y en la importancia que se de a cada uno de ellos.

Normalmente, una asignatura no es impartida en solitario por un profesor, sino que son varios los que colaboran en ella (o quizás en un conjunto de asignaturas relacionadas). En el proceso de elaboración de los objetivos es conveniente que participe el grupo completo de profesores implicados, y que se tenga en consideración la opinión de otros profesores que en el pasado se hayan encargado de esa docencia. El intercambio de opiniones entre el profesorado relacionado con una asignatura proporciona una visión de conjunto enriquecedora para todos. Por otro lado, este proceso de discusión puede servir como estímulo para cada profesor si le proporciona una visión en perspectiva de su propia labor. Este trabajo de coordinación, que no está exento de dificultades, tiene gran importancia, y por encima de otras ventajas permite ofrecer al alumno una imagen coherente de la asignatura.

En cuanto al entorno y las condiciones en las que se desarrollará la asignatura, el número de alumnos por grupo de teoría y prácticas será fundamental. Claramente no es lo mismo trabajar en un entorno masificado que con grupos pequeños en los que es posible un seguimiento personalizado y continuo. Otro factor muy importante será la disposición de recursos como una buena biblioteca que ponga a disposición de los alumnos y profesores material específico y actualizado para cada área temática. En el caso que nos ocupa, es preciso recordar que nos encontramos en un área de conocimiento en continua evolución, que requiere un continuo reciclaje y actualización de los conocimientos. Para la identificación de objetivos prácticos será fundamental la disponibilidad de laboratorios adecuados en los que el alumno pueda disponer del tiempo necesario para adquirir las habilidades necesarias.

### 6.1.2. Objetivos generales

Independientemente de las condiciones concretas en las que se desarrolle la enseñanza, hay unos cuantos objetivos que pueden considerarse como generales, para cualquier estudio tecnológico, y en particular, para las titulaciones de ingeniería de telecomunicación. Todos ellos son habilidades, capacidades, conocimientos y experiencias que sería deseable que adquiriese un alumno:

- Desarrollo de habilidades para el estudio continuado y la adquisición de los conocimientos necesarios para afrontar problemas nuevos. Entre estas habilidades será especialmente importante proporcionar al alumno capacidades de autoformación, partiendo de una documentación básica que se completará, si es preciso, con el resultado de una búsqueda de documentación.
- Capacidades de abstracción, y una cierta dosis de intuición, para abordar problemas a los que el alumno no se ha enfrentado antes. De forma más general, el alumno debería ser capaz de identificar los elementos que es preciso considerar para resolver un problema.
- Sólida base teórica, especialmente en los conocimientos necesario para entender las tecnologías que componen el núcleo de su formación. Esta base deberá permitir al alumno evolucionar en la comprensión de nuevos sistemas y tecnologías que irán apareciendo durante el desarrollo de su labor profesional.
- Conocimiento detallado de las tecnologías importantes en el ámbito de la titulación, de su situación actual y de su previsible evolución futura, de sus condiciones de entorno y de su influencia en otros campos de aplicación.
- Desarrollo de capacidades para proponer soluciones y tomar decisiones económicamente óptimas en un plazo prefijado, utilizando las opciones tecnológicas disponibles, y teniendo en cuenta las condiciones de entorno.
- Conseguir el desarrollo de la capacidad crítica. El alumno ha de ser capaz de realizar estudios de propuestas de solución de problemas, y de diseños y planes de ejecución, y juzgar sobre su viabilidad y potencialidad, hasta el punto de poder recomendar, basado en sus conocimientos y experiencia, una opción sobre otras similares pero no tan ventajosas. También habrá de ser capaz de argumentar esa decisión.
- Ser capaz de expresarse de forma oral y escrita para transmitir sus conocimientos, diseños y soluciones a problemas. Esto no sólo es importante por si mismo, sino que además fuerza al alumno a poner en orden sus propios conocimientos, y a racionalizar su proceso de toma de decisiones.

Todos estos objetivos no se circunscriben a una asignatura concreta, sino que forman parte de los que habrán de conseguirse en toda la titulación. Podría decirse que forman parte de lo que se considera como perfil general de un profesional formado en una titulación técnica. Sin embargo, los objetivos concretos de cualquier asignatura deberán tenerlos en cuenta, y reforzarlos allá donde sea posible.

En el contexto que nos ocupa, estos objetivos están reflejados en las capacidades generales que se especifican en el plan de estudios, y en particular en las que se detallan para la asignatura. Además, el plan de estudios especifica también objetivos específicos para la asignatura, que el profesor deberá detallar y matizar, teniendo en cuenta los aspectos mencionados, para cada edición de la asignatura.

## 6.2. Método docente

Una vez definidos los objetivos docentes de una asignatura habrá que decidir la metodología (o más bien el método, o el modelo de método) que se emplea para tratar de conseguirlos. En el sistema de definición de los planes de estudio actuales, los objetivos y capacidades que se esperan de los estudiantes están definidos con bastante detalle, pero la metodología docente típicamente queda a la elección del profesor. En este apartado se propone una entre otras posibles. La experiencia de unos cuantos años de docencia, más el intercambio de experiencias con otros profesores son su principal aval.

La principal premisa de este método docente es que el aprendizaje mediante la práctica es, para las asignaturas que nos ocupan, más eficiente, y obtiene mejores resultados, que otros más teóricos. El

método está basado fuertemente en el aprendizaje basado en problemas (*problem-based learning*) [18] [4], con algunos aspectos de aprendizaje invertido (*flipped learning*) [26] [25]. Sin embargo, no se trata de ser purista en su aplicación, y se añaden a estas metodologías algunas técnicas más tradicionales, especialmente en las ingenierías, como el aprendizaje mediante ejemplos (incluyendo la resolución de ejercicios, de forma colectiva, en el aula), las prácticas individuales y en grupo, e incluso en ciertos momentos, las clases magistrales. Todo ello estará apoyado, en la medida de lo posible, por recursos educativos abiertos (en gran parte desarrollados por los propios profesores), y complementado por prácticas de colaboración basadas en las herramientas de apoyo al desarrollo de software que son hoy comunes en muchas empresas.

Es preciso mencionar que, para que el método sea posible tal y como se describe aquí, las clases han de tener lugar en el laboratorio, o al menos, que cada alumno tenga la posibilidad de resolver los problemas en un ordenador, de forma individual o en grupos muy pequeños, durante la propia clase (ver sección 6.4 sobre medios necesarios).

### 6.2.1. Aprendizaje basado en problemas

La mayor parte de las clases están estructuradas alrededor de la resolución de problemas. En algunos casos, si el tema así lo requiere, pueden presentarse algunos conceptos teóricos útiles para la resolución antes de plantear los problemas. Pero normalmente, se empezará precisamente por el planteamiento de problemas, pasando después a la discusión de posibles soluciones, y a la prueba de alguna o varias de ellas. Este último paso requerirá normalmente desarrollar un pequeño programa, que los alumnos tendrán que hacer por sí mismos. Finalmente, se explicará una solución elaborada por el profesor, o una elegida por los alumnos, sacando las conclusiones oportunas para reforzar el proceso de aprendizaje. Cuando sea conveniente, se introducirán, tras la resolución del problema, conceptos teóricos que ayuden a generalizar la solución, a formalizarla, y a acercar a los alumnos al vocabulario y a la estructura de pensamiento común en el campo.

Detallemos un poco el esquema habitual de estas sesiones de aprendizaje basado en problemas:

- Opcionalmente: introducción de algunos conceptos teóricos. Normalmente esto ocurrirá cuando se está comenzando un tema y es preciso introducir un cierto vocabulario, ciertos conceptos que enfoquen la resolución de los problemas a una determinada estrategia, o un enlace con problemas de temas anteriores.
- Planteamiento del problema. El problema se introducirá informalmente, pero se dispondrá también de un enunciado lo más exacto posible. La introducción informal servirá para ponerlo en contexto, y para iniciar la siguiente fase. La descripción escrita exacta servirá de especificación para la implementación de la solución que se realizará posteriormente. Los problemas estarán cuidadosamente elegidos de forma que introduzcan gradualmente los aspectos que se quieran trabajar, y en general tendrán una dificultad que permita a los alumnos llegar a soluciones por sí mismos, con muy poco apoyo del profesor. Tendrán una dificultad adecuada como para poder discutirlos e implementar su solución en periodos cortos, entre 30 y 90 minutos (dependiendo de los conceptos a trabajar).
- Discusión del problema y sus posibles soluciones. Esta discusión se realizará entre los alumnos, en grupos pequeños, o de forma abierta entre todos los alumnos y el profesor. La discusión debe estar dirigida de forma que el alumno tenga oportunidad de ver el problema desde distintos puntos de vista, y considerar varias soluciones posibles, y los problemas y ventajas que pueden plantear.
- Implementación de una solución al problema. Para fijar la solución, y permitir a los alumnos que trabajen más con ella para comprenderla completamente, los alumnos la implementarán en trabajo individual o en grupos pequeños. Luego compartirán estas soluciones, incluso si no han funcionado completamente.
- Discusión de una o varias soluciones. Bien presencialmente en el aula, bien asincrónicamente en el campus virtual, se discutirá al menos una solución. Puede ser la solución de referencia que proporcione el profesor (en general tendrá una para cada problema), alguna de las que han realizado los alumnos, o ambas. Esta fase servirá para considerar en detalle la estrategia o estrategias seguidas en la resolución, y analizar los pros y los contras de las opciones elegidas.



- Opcionalmente: introducción de algunos conceptos teóricos. Siempre que sea conveniente, el profesor aprovechará esta oportunidad para formalizar las soluciones aportadas, y para introducir los conceptos adecuados al tema que se esté tratando.

Este esquema se repetirá sucesivamente, en una o varias clases. Los problemas introducidos en esta sucesión estarán diseñados, siempre que sea posible, para ir construyendo un hilo argumental que lleva hacia problemas reales. Habitualmente se comenzará con una versión muy simplificada de un problema real, al que en pasos sucesivos se irán incluyendo cada vez más detalles del problema completo, si es posible llegando al final a una implementación cercana, en su estructura, a la que podría resolver el problema real de forma completa.

Para evitar que los alumnos se queden descolgados en este proceso, el profesor ofrecerá siempre implementaciones de referencia. Estas servirán por un lado para que los alumnos que no hayan podido realizar implementaciones completas de su solución puedan aún así comprender una solución completa. Por otro, los que las hayan terminado, podrán comparar la suya con la de referencia, quizás aprendiendo otras estrategias. Y por último, tanto unos como otros dispondrá de un punto de referencia para continuar con los siguientes problemas de la cadena, utilizando esta práctica de referencia para completarla con las soluciones a los nuevos problemas que se vayan planteando, si así lo desean.

De esta forma, por un lado el alumno irá tomando responsabilidad en la resolución de problemas pequeños, pero cercanos a los que ese encontrará más adelante en su vida profesional, e irá desarrollando estrategias que le ayuden en esta tarea. Por otro, irá conociendo paso a paso los problemas y las soluciones relevantes en los temas incluidos en el plan de estudios.

Además, la resolución de estos problemas ayudarán también al alumno (y al propio profesor) a evaluar su grado de entendimiento de los temas que se estén trabajando. Por supuesto, también servirán para desarrollar la capacidad creativa y de razonamiento de los alumnos, para que adquieran habilidades metodológicas de resolución, y para aplicar a casos prácticos los conocimientos teóricos.

### 6.2.2. Introducción de conceptos teóricos

Como ya se ha comentado, en ciertos momentos será conveniente introducir conceptos teóricos. Para ello se expondrán de forma sistemática y ordenada los conceptos e ideas pertinentes. La exposición que se realice en cada caso debe destacar su relación con otros temas, y con los problemas resueltos o por resolver. Así mismo es muy importante que los alumnos perciban siempre que sea posible la importancia de cada uno de los temas, su uso en el mundo real, y el tipo de problemas que se presentan en su ámbito.

Es muy deseable que antes de asistir a estas sesiones sobre conceptos teóricos los alumnos hayan tenido ya una exposición, mediante lecturas o visionado de vídeos, a los principales conceptos que se vayan a introducir. De esta forma el alumno puede madurar y reflexionar los conocimientos que ya ha adquirido en parte, y sobre todo realizar las preguntas pertinentes para que el profesor pueda aclararle los puntos que no haya comprendido. Ello implica que el profesor debería informar con anterioridad a los alumnos de qué tema va a tratarse en cada sesión teórica, y que estos deberían tener a su disposición documentación relativa a ese tema con antelación suficiente.

Tradicionalmente, las clases de teoría se podrían clasificar en dos tipos: las clases magistrales (las más clásicas) y las clases abiertas. Las primeras son realmente conferencias desarrolladas por el profesor mientras que las segundas pretenden la participación directa de los alumnos para repasar los conceptos introducidos, discutir los temas referentes a la bibliografía de consulta, etc.

Sin embargo, más que dos tipos de clases de teoría en realidad tenemos un continuo de tipos de clases, de los que esos dos serían probablemente sus extremos. Para la introducción de los conceptos teóricos se propone un modelo que está en algún lugar de ese continuo, y cuyas características se detallan a continuación:

- De las clases magistrales se han tomado fundamentalmente la exposición oral por parte del profesor del tema objeto de la clase. Esta exposición suministrará datos, explicación de conceptos, relacionará ideas, y por supuesto proporcionará razonamientos que justifiquen todo lo anterior. Naturalmente con esta exposición el profesor tratará de transmitir entusiasmo por la materia, crear interés por los problemas y soluciones que se proponen, y despertar en el alumno su capacidad de razonamiento para anticiparse, siempre que sea posible, a las conclusiones de cada punto tratado en clase.

- De las clases abiertas se ha tomado, como ya se ha comentado, la resolución de problemas en colaboración entre alumnos y profesor, la realización de discusiones con los alumnos y la respuesta a cualquier tipo de dudas que pueda surgir al alumno durante la exposición. Además, se intenta realizar la exposición dando pie a (e incluso motivando directamente) la participación del alumno con preguntas y comentarios.
- Además, se integrarán en la clase teórica la resolución de problemas, y la descripción de sistemas reales. Por un lado, esta integración permite que la clase sea más dinámica e interactiva, y rompa un poco la inevitable monotonía de las exposiciones puramente teóricas. También ayudará a fijar las ideas aportando casos y ejemplo concretos de aplicación.
- Como apoyo a la exposición se utilizarán en algunos casos transparencias. Por un lado, estas transparencias servirán para centrar la exposición y uniformar en lo posible el tratamiento de los temas cuando varios profesores imparten la misma asignatura a varios grupos. Por otros, proporcionará a los alumnos de un esquema con los puntos más importantes de cada tema. Es muy importante explicar a los alumnos que las transparencias no son más que eso, un apoyo a la exposición de los temas teóricos, y en ningún caso sustituyen a unos apuntes y mucho menos a un libro. En el capítulo 9 se muestra un ejemplo del tipo de transparencias que se proponen.

De esta manera, se intenta añadir interactividad a la clase magistral. Como ya se ha dicho, sería conveniente que los alumnos hubieran tomado ya contacto antes de la clase con los temas tratados en ello, con lo que la exposición magistral sería prácticamente innecesaria. Desgraciadamente, esto no es habitual en el entorno que nos ocupa. Por ello, la componente “magistral” de la clase suple la lectura y estudio previos que no suele realizar el estudiante, lo que constituye un complemento de las indudables virtudes de la clase abierta.

### Prácticas constructivas

Además de la resolución de problemas mediante construcción de programas, se plantará a los alumnos la realización de varias prácticas constructivas de más entidad. Si se consiguen diseñar bien, este tipo de prácticas puede motivar mucho al alumno, especialmente si su resultado es un sistema que pueda utilizarse para realizar alguna tarea reconocida como “interesante” por el alumno.

Para diseñar este tipo de prácticas, se tendrán en cuenta los siguientes aspectos:

- **Duración.** Para que la práctica tenga una cierta entidad, es conveniente que se dedique a ella una parte sustancial del tiempo dedicado a prácticas en la asignatura. Esto permitirá realizar un sistema más completo, y que el alumno se acerque lo más posible a sistemas reales.
- **Recursos** a disposición del alumno, incluyendo no sólo los disponibles en el laboratorio, sino también teniendo en cuenta la posibilidad de que el alumno haga partes de la práctica en otros lugares (en su hogar, en el trabajo, etc.).
- **Diseño** del sistema a implementar. En general, es bueno que el alumno no reciba directamente un diseño que sólo tiene que implementar. En la mayoría de los casos, el diseño de una solución es tan importante o más que su implementación. Por ello, en lo posible, será bueno que el alumno participe en el diseño del sistema a construir.
- **Desarrollo.** Es importante mantener la motivación y la asistencia de alumnos a las sesiones prácticas durante todo el tiempo que durará el desarrollo. Para esto se ha observado que da un resultado muy bueno utilizar variantes del modelo de desarrollo en espiral. El desarrollo de la práctica pasa por varias fases, cada una de ellas más completa, más detallada y más compleja, aunque todas ellas funcionales (naturalmente, en distinta medida). Cada fase tendrá sus propias especificaciones, su propio diseño (basado en el de la fase anterior), y su propia implementación. Su duración será relativamente corta, de forma que se puedan realizar tres o cuatro fases durante el periodo de duración de la práctica.
- **Enfoque** en los aspectos relacionados con la asignatura. Es conveniente procurar que el alumno pueda ser creativo en temas relacionados con la asignatura, y no se vea dificultado por otros ajenos a ella, como problemas con el lenguaje o con el entorno de programación. Además de un cuidadoso

diseño de la práctica, se pueden tomar medidas dirigidas especialmente a evitar a los alumnos trabajos y complicaciones no relacionados con la asignatura. Por ejemplo, se les puede proporcionar bibliotecas especialmente orientadas al objetivo de la práctica, implementaciones de ejemplo para alguna de las fases, documentos de preguntas más frecuentes para tareas a realizar en la práctica, etc.

- **Niveles de trabajo** diferentes para alumnos con diferentes intereses y niveles de motivación. Para esto se ofrecerán partes obligatorias y optativas. Las primeras tendrán que realizarlas todos los alumnos. Las segundas estarán destinadas a que los alumnos más motivados puedan mejorar su nota.
- **Trabajo en solitario o en grupo.** ¿Es mejor que los alumnos las aborden en solitario o en grupos de distintos tamaños? Ambos casos tienen sus ventajas, según prime el esfuerzo personal y la comprensión de todos los aspectos de la práctica o la colaboración en la resolución de problemas. En este proyecto se prefiere un enfoque mixto, donde parte de las prácticas (o fases de las prácticas) sean realizadas en grupo, aunque luego haya un trabajo personal que permita al alumno enfrentarse en solitario con los problemas, y que proporcione un trabajo evaluable de forma individual.
- **Motivación** del alumno. Hay que procurar que la práctica proporcione al alumno “algo que enseñar”, desde las primeras fases. En general, esto le motivará ante sus compañeros y ante sí mismo. De hecho, si la práctica permite realizar algún tipo de actividad, es más que posible que el alumno dedique una cantidad no despreciable de tiempo a realizar esa actividad, con lo que en general se familiarizará más con el objetivo de la práctica, y estará motivado para mejorarla. Algunos ejemplos de prácticas de este tipo son las implementaciones de juegos, de programas que interaccionan con sistemas reales (como servidores web) o de sistemas que realizan alguna función “útil” (como construir una red social).
- **Evaluación.** No hay que olvidar que la práctica ha de ser evaluable. Normalmente, pueden mantenerse las primeras versiones como no evaluables, pero al menos la última ha de estar diseñada de forma que tenga elementos suficientes variados como para poder evaluar el trabajo de cada uno de los alumnos, la comprensión del dominio de conocimiento en que se desarrolla, lo adecuado de las soluciones a los problemas que plantean las especificaciones y la calidad de la implementación realizada.

En el capítulo 12 se muestra un enunciado de práctica constructiva que se ajusta bastante a las ideas expuestas en este apartado.

### 6.2.3. Otros recursos de ayuda a la docencia

La labor docente no se acaba en las clases de teoría y de prácticas. Para ayudar a los alumnos en su aprendizaje, pueden usarse también otros muchos medios, que en unos casos servirán para profundizar en los temas, en otros completarán las enseñanzas de clase, y en otros proporcionarán nuevas perspectivas y posibilidades docentes.

#### Tutorías

Entre las obligaciones docentes del profesor está el atender a las consultas de los alumnos en las llamadas horas de tutoría. Estas consultas constituyen un complemento fundamental de las clases, ya que permiten un trato mucho más directo e individualizado con los alumnos. También son un buen momento para discutir con los alumnos las soluciones a los problemas propuestos, o las implementaciones que estén realizando en las clases prácticas.

Sin embargo, desgraciadamente no es habitual que los alumnos utilicen muy a menudo esta posibilidad. Por ello, es bueno fomentarlas por todos los medios posibles. Entre ellos, cabe destacar el dejar claramente publicados los horarios y lugares de tutoría. Y aún mejor, siempre que sea posible, limitar lo menos posible en el tiempo y en el espacio las oportunidades de tutoría. Si el profesor puede atender a un alumno, cualquier momento y lugar puede ser bueno para una tutoría.

Otra forma de fomentar las tutorías, especialmente para los alumnos con problemas de horario, o incluso para los que no acuden frecuentemente a clase, es usar tutorías telemáticas.

### Servicios digitales de apoyo

Entre otros, se proponen los siguientes servicios y medios de información:

- **Documentos fundamentales.** Se ofrecerán a los alumnos dos documentos fundamentales, que expondrán con el mayor detalle los objetivos, la planificación, al evaluación, y otros aspectos relevantes de la asignatura. Estos documentos son la Guía Docente de la asignatura, que está disponible en el sitio web de la Universidad para todas las asignaturas de todas las titulaciones, y el llamado Programa de la asignatura, que es en realidad un documento más completo, en parte parecido al Syllabus utilizado en las universidades anglosajonas. En particular, éste último incluye la colección completa de problemas que se realizarán durante el transcurso de la asignatura, junto con enlaces y referencias a las soluciones de referencia.
- **Sitio web de la asignatura.** Los alumnos encontrarán muy conveniente que se les deje toda la información posible en un sitio web (el sitio de la asignatura). Esa información debería incluir por lo menos información general de la asignatura (incluyendo temarios, planificación en el tiempo, bibliografía básica, etc.), copia de las transparencias y apuntes, copia de los guiones de prácticas y material de apoyo, colecciones de problemas (incluyendo exámenes de otros años), implementaciones de referencia a los problemas planteados, etc. En el caso de las asignaturas que nos ocupa este será un recurso fundamental, que se gestiona como un recurso abierto (ver apartado 10).
- **Campus virtual.** Permite el seguimiento de la asignatura cuando se puede asistir poco a las clases presenciales, pero además sustituye con ventaja a las listas de correo electrónico para muchos tipos de avisos y discusiones. Se pueden abrir foros para proporcionar con facilidad punteros e información complementaria sobre cada tema teórico, o noticias de actualidad relacionadas con la asignatura.
- **Forja de desarrollo,** para su uso en las implementaciones de las clases de prácticas. Pueden servir también para que unos alumnos puedan ver el trabajo de otros en las fases de trabajo en grupo de las prácticas constructivas.
- **Recursos disponibles en Internet,** desde páginas de preguntas más frecuentes y manuales hasta vídeos. Gran parte de estos recursos pueden usarse como material de apoyo, o como referencias para que el alumno pueda profundizar en los conceptos vistos en clase. Por otro lado, teniendo en cuenta la asignatura que nos ocupa, también hay muchos recursos que pueden usarse como ilustración de los servicios de comunicación tratados en clase.
- **Archivos de clases magistrales,** grabadas en vídeo y digitalizadas para que puedan ser accedidas en línea. El equipo necesario para hacer estas grabaciones no es muy caro (cámara de vídeo, micrófono inalámbrico), y pueden aprovecharse ocasiones en las que un buen ponente venga a la Universidad a dar una charla para grabarle. Posteriormente, la grabación puede digitalizarse a formato MPEG o similar, y colocarla en un archivo de grabaciones que los alumnos puedan usar desde el laboratorio (donde cada puesto estaría dotado de salida de sonido y auriculares).

### Cursos, seminarios y conferencias

Fuera del temario quedan muchos asuntos interesantes relacionados con la asignatura. Por un lado están las novedades tecnológicas y los temas de actualidad, por otro el tratamiento detallado de temas que sólo se ven de pasada, por otro aún el análisis de casos reales de uso de tecnologías relacionadas con la asignatura. Si esto es cierto para cualquier asignatura, y en especial para las relacionadas directamente con la tecnología, lo es aún más para una como la que nos ocupa, dado el rápido cambio de las tecnologías de comunicación, y su profunda influencia en todas las actividades humanas. Los cursos, seminarios y conferencias son métodos que se han utilizado tradicionalmente para rellenar este hueco. Entre todos ellos podemos destacar los siguientes formatos:

- Seminarios internos del equipo docente. Sirven sobre todo como formación permanente para el equipo de profesores. Pueden invitarse a ellos a alumnos, especialmente los más interesados en la asignatura. Son simples de preparar y se pueden realizar con una frecuencia relativamente alta.

- Cursos y conferencias realizados por los propios profesores. Pueden ser un buen complemento específico para los temas tratados en la asignatura. Por ejemplo, los profesores pueden trasladar con ellos a los alumnos asuntos que les interesan especialmente, o los que están tratando en sus líneas de investigación.
- Cursos y conferencias realizados por ponentes invitados. Pueden servir para dar una visión complementaria a la del equipo docente al que pertenece el profesor de la asignatura. Será especialmente interesante cuando el ponente es experto en el tema que trata. Si además viene del mundo empresarial, podrá proporcionar la perspectiva de la industria sobre el tema que trate.

### Otras actividades

Otras actividades complementarias interesantes pueden ser las siguientes:

1. **Demostraciones.** Pueden presentarse a los alumnos equipos y servicios reales relacionados con la asignatura. La presentación puede hacer de forma que fomente la capacidad crítica del alumno, tratando de exponer los puntos buenos y malos del producto en cuestión. Un ejemplo simple pero que suele resultar muy adecuado es una exposición de los diversos equipos que componen la red de la Universidad, y de los servicios que funcionan en ella.
2. **Colaboraciones** en proyectos de investigación. Los alumnos más interesados pueden colaborar en los proyectos de investigación del grupo, desarrollando trabajos parciales que podrían servir también de preparación para su proyecto fin de carrera.

## 6.3. Evaluación

La evaluación es un aspecto muy importante del proceso formativo. Por un lado, la sociedad concede a la Universidad capacidades de certificación del conocimiento que suelen estar delegadas en el profesor de cada asignatura. Por ello, cuando acaba el periodo docente, el profesor ha de calificar a todos sus alumnos, decidiendo cuáles tienen conocimientos y habilidades suficientes, cuáles no, y en ambos casos transformar esta decisión en un número. Por otro lado, el propio profesor y el desarrollo general de la asignatura también han de ser evaluados para detectar problemas y sobre todo para identificar aspectos susceptibles de mejora.

En este apartado se proponen mecanismos de evaluación tanto para el alumno como para el conjunto alumno-profesor-entorno.

### 6.3.1. Evaluación del alumno

El final del proceso docente en toda asignatura supone la evaluación de los conocimientos adquiridos por el alumno. Dado que las enseñanzas impartidas son tanto teóricas como prácticas, es conveniente realizar una evaluación por separado de ambas partes, aunque quizás con pesos diferentes, según la importancia que hay tenido cada una en el desarrollo de la asignatura.

#### Evaluación de los conceptos teóricos

Aunque en general es deseable la evaluación continua, raramente puede llevarse a efecto en los grupos numerosos que son comunes en la Universidad española. Por ello, es preciso recurrir a los exámenes. Para la parte teórica, se proponen exámenes escritos con las siguientes características:

- El examen estará compuesto por varias partes, que idealmente deberían cubrir todo el temario.
- Se evitará preguntar por conocimientos meramente memorísticos. De hecho, el examen podrá realizarse con libros y apuntes, primando las preguntas que obliguen a aplicar y relacionar los conocimientos adquiridos en la asignatura.
- Se incluirán problemas cuya resolución implique no sólo el conocimiento de las materias estudiadas en la asignatura, sino una cierta capacidad de razonamiento para llegar paso a paso hasta la solución final.

- El examen estará diseñado de forma que se puedan detectar varios niveles de conocimiento de la asignatura y de habilidad en el manejo de los conceptos tratados en ella.
- Cada pregunta o problema estará diseñada de forma que el alumno tenga que organizar sus conocimientos antes de responder. Naturalmente, la duración del examen estará ajustada para que el alumno tenga tiempo suficiente para responder con precisión.

Este tipo de examen presenta algunas ventajas frente a otros:

- Frente a los exámenes orales, el alumno tiene tiempo para meditar y estructurar sus conocimientos antes de responder.
- Frente a los exámenes de preguntas muy cortas, o tipo test, permite al profesor un mayor grado de discriminación sobre los conocimientos reales del alumno.

Naturalmente, también hay que estar atento en el momento de diseñar el examen, para evitar los siguientes problemas:

- El alumno puede orientar su estudio únicamente a superar la prueba, centrándose en los temas y problemas que piensa que pueden ser incluidos en él. Para evitarlo, habrá que cambiar continuamente la forma del examen, y ser lo más creativo posible en la elección de preguntas y problemas.
- Necesariamente el examen abarcará sólo una parte del temario. Es preciso asegurarse de que el conjunto de preguntas y problemas cubren razonablemente al menos la parte más importante de la asignatura. Siempre que sea posible, se tratará de que cada parte del examen precise del conocimiento de muchos temas para contestarla correctamente.
- Es difícil corregir estos ejercicios con objetividad. Por ello se hace preciso la elaboración previa a la corrección de una criterios claros de corrección y calificación que se puedan aplicar luego en cada caso.

De hecho, en muchos de los exámenes en las asignaturas que nos ocupan se ha probado con un modelo basado en describir un sistema que se quiere construir, con detalle razonable, y luego plantear varias preguntas sobre cómo resolver problemas que aparecerán en esa construcción, o explicar en cierto detalle como deberían funcionar algunos aspectos. Este tipo de examen se asemeja, hasta cierto punto, a la realización de una “práctica teórica”, y en general ofrece un buen equilibrio entre las ventajas y los riesgos descritos anteriormente.

En el capítulo 13 se muestra un ejemplo de examen diseñado según las directrices que se proponen en este apartado.

Como complemento a la evaluación teórica, si el tamaño del grupo y la planificación temporal lo permiten, se pueden permitir a los alumnos preparar temas relacionados con la asignatura, y exponerlos ante la clase. Esta exposición puede llevar también aparejada una calificación que matice la obtenida en el examen teórico escrito. Con ella se podrán medir especialmente las capacidades expresivas y el grado de conocimiento específico que un alumno dado puede lograr por su cuenta, utilizando la documentación señalada por el profesor.

### **Evaluación práctica**

Una parte importante de la evaluación de la asignatura se realizará sobre la realización de prácticas. Esta evaluación será de dos tipos:

- Evaluación del trabajo continuo de realización de las implementaciones fruto del método de aprendizaje basado en problemas, y de prácticas constructivas de más entidad.
- Evaluación de una práctica constructiva que permita evaluar todas (o la mayor parte) de las capacidades adquiridas durante la asignatura.

La primera se realizará dando pequeños incentivos a los alumnos que realicen satisfactoriamente esas implementaciones, aunque en general estarán diseñadas más para favorecer el proceso de aprendizaje que para ser evaluadas. La evaluación supondrá más una motivación adicional para el alumno que una verdadera evaluación de capacidades, pues se espera que todos los alumnos sean capaces de realizarlas sin problemas especiales.

Para la evaluación de la práctica constructiva final, se propone un método basado en los siguientes elementos:

- Realización individual por parte de cada alumno de una parte de las prácticas.
- Entrega por parte de cada alumno de un elemento (normalmente un programa de ordenador) que será fruto de ese trabajo individual.
- Examen presencial para cualquier alumno que plantee dudas al profesor sobre cuál debería ser su calificación, más un grupo aleatorio de alumnos, combinado con la comprobación de la funcionalidad del elemento de evaluación.

Desde el punto que se expone en esta memoria, este método de evaluación presenta varias características que lo hacen ventajoso:

- No se evalúa todo el proceso de las prácticas, sino sólo los resultados finales de algunas de ellas (normalmente de las últimas fases de las prácticas constructivas). Eso permite que otras partes de las prácticas puedan realizarse como trabajos en grupo, si se estima que es más conveniente para su proceso de aprendizaje.
- Se puede evaluar con relativa facilidad del conocimiento del alumno de los problemas y soluciones relacionados con la realización de las prácticas, y no sólo el grado de cumplimiento de especificaciones.
- El alumno tiene la oportunidad de explicar los detalles que estime conveniente sobre su práctica, lo que en general es conveniente por el grado de complejidad que puede alcanzar.
- Será difícil presentar para su evaluación trabajos no realizados personalmente por el alumno, y en caso de que se haga, al menos el alumno deberá estar muy familiarizado con ellos.

### 6.3.2. Evaluación de la asignatura y del profesor

Esta evaluación debe, en primer lugar, permitir comprobar si se han conseguido o no los objetivos propuestos y en qué grado. Esto permitirá decidir si estos eran o no realistas, y por tanto si en el futuro deben modificarse. Además debe aportar los datos necesarios para poder tomar decisiones sobre métodos, programas y orientaciones y evaluar los propios métodos didácticos.

#### Evaluación final

Es conveniente que, al finalizar cada curso, se realice una encuesta entre los alumnos para que estos expresen su opinión sobre la marcha de la asignatura y sobre el equipo de profesores encargados de la docencia de la mismas.

La utilidad de la encuesta es doble. Por una parte, para la institución docente es un dato más para evaluar la actividad desarrollado por los profesores. Por otra, sirve al profesor de la asignatura como indicador de la opinión mayoritaria del alumnado sobre su propia actuación. Así mismo, permite identificar los aspectos que, según el alumno, no han funcionado adecuadamente durante el curso.

En la actualidad la Universidad Rey Juan Carlos realiza encuestas en forma de cuestionario de respuesta cerrada múltiple sobre diversos aspectos de la docencia y los servicios. En ellas el alumno puede elegir una respuesta entre las 5 propuestas (generalmente graduaciones del tipo: Mucho, Bastante, Normal, Poco, Muy Poco).

Según la opinión del candidato este tipo de encuestas aporta poca información. Aunque puede ayudar a detectar problemas, difícilmente servirá para identificar sus causas. Por ello se propone realizar cuando esté finalizado el periodo docente una encuesta de formato abierto, donde los alumnos puedan comentar anónimamente sobre los diversos aspectos de la asignatura. En el capítulo 14 se muestra una de estas encuestas, que fue pasado a varios grupos de alumnos en la Universidad Rey Juan Carlos.

### Círculos de calidad

La realización de “encuestas finales” tiene el problema de que los problemas se detectarán, en el mejor caso, cuando ya ha terminado el periodo docente. Además, la necesidad de realizarlas anónimamente hace imposible mantener una fase de discusión que aclare la opinión de las partes sobre el desarrollo del curso. Un mecanismo que puede complementar estas encuestas, aportando soluciones para estos problemas, es la formación de círculos de calidad para cada asignatura.

Dicho en pocas palabras, un círculo de calidad estará compuesto por el profesor o profesores de una asignatura y un grupo de alumnos voluntarios. Su misión será identificar al principio del periodo docente de un grupo de parámetros que interesen a todas las partes para medir el desarrollo de la asignatura, y seguir la evolución de estos parámetros hasta que termine este periodo. Naturalmente, si los parámetros se desvían de lo que sea considerado deseable, se buscarán los motivos y se tratará de tomar medidas que permitan volver a la situación deseable.

El tipo de parámetros que se usen puede ser muy variado, pero a modo de ejemplo se van a citar unos cuantos. El profesor podría proponer entre otros los siguientes:

- Número de alumnos que asisten a clases de teoría y prácticas.
- Desviación del desarrollo de los temas frente a la planificación temporal.
- Número de preguntas por clase.
- Número de alumnos que realizan alguno de los ejercicios complementarios propuestos.

Los alumnos podrían proponer, entre otros, los siguientes:

- Retraso o adelanto de los momentos de comienzo y terminación de cada clase.
- Disposición del material de cada tema antes de que comience a exponerse.
- Cantidad de referencias accesibles para ellos para cada tema de la asignatura.

El círculo de calidad se reunirá periódicamente, levantando acta de las reuniones (para que quede un registro histórico del desarrollo de la asignatura). En cada reunión repasará la evolución de los parámetros que se vigilan, si es caso propondrá otros nuevos, y estudiará las desviaciones. Al terminar la asignatura tendrá acceso a las encuestas realizadas por el grupo, y tras leerlas y estudiar la evolución de los parámetros observados elaborará un breve informe donde se trate de identificar los mayores problemas que ha tenido el desarrollo de la asignatura, y donde se propongan soluciones para ellos, susceptibles de ser puestas en funcionamiento el curso siguiente.

Desde luego, no es fácil hacer funcionar un círculo de calidad si no hay una labor de mentalización previa, tanto por parte del alumno como del profesor. Es especialmente importante que el círculo no se convierta en un lugar para las quejas, y también que sus participantes no caigan en la percepción de que nada se puede arreglar y por lo tanto su trabajo no sirva para nada. Si se superan estos problemas, y se enfoca adecuadamente su trabajo, el círculo de calidad es una herramienta muy potente para la detección temprana de problemas, y su reencaminamiento hacia lo deseable sin tener que esperar a un nuevo curso.

## 6.4. Medios necesarios

Para poder llevar a cabo los planes y objetivos mencionados anteriormente, es preciso disponer de unos ciertos medios, que se detallan a continuación.

### 6.4.1. Laboratorio docente

El elemento fundamental, dada la metodología basada en la resolución de problemas, es el laboratorio en que tendrán lugar las clases. Aunque algunos problemas tendrán un planteamiento y resolución que podríamos llamar teórico, la mayoría de ellos serán fundamentalmente prácticos. Lo que en una asignaturas como las que nos ocupan supondrá la realización y prueba de programas de ordenador. Por lo tanto, es fundamental disponer de esos ordenadores en el aula. Como los problemas se resolverán durante las clases, no es suficiente que el alumno tenga acceso a los ordenadores antes o después de éstas.





Figura 6.1: Vista parcial de laboratorio docente de la URJC (Laboratorio 3.210 de Laboratorios III, campus de Fuenlabrada).

Las ventajas fundamentales del laboratorio frente a otras opciones (como que el alumno lleve al aula su propio ordenador portátil) son:

- Todos los alumnos disponen de un entorno adecuado para la realización de los ejercicios, que no dependa de sus medios, ni de sus conocimientos previos en cuanto a instalación, configuración y mantenimiento de sistemas.
- Los ordenadores pueden ser instalados y configurados adecuadamente antes de que comiencen las clases, con lo que los alumnos no han de dedicar tiempo a estas tareas: se encuentran directamente con un entorno de trabajo completo, listo para iniciar sus actividades.
- El entorno de trabajo de profesores y alumnos, y de estos entre sí, es muy uniforme, con lo que la resolución de cualquier problema, o la compartición de cualquier solución, será mucho más fácil.
- El laboratorio proporciona un sistema “canónico”, que puede utilizarse como sistema de referencia por ejemplo para definir el entorno en que deberán funcionar los programas realizados, incluyendo los proyectos de más envergadura que tengan que realizar los alumnos.

En caso de que no se pueda disponer de un laboratorio adecuadamente organizado, se podría seguir el método propuesto si cada alumno puede utilizar durante la clase su propio ordenador, y éste ha sido adecuadamente configurado antes del comienzo de las clases. Según la tecnología va avanzando, el coste de los ordenadores se reduce, y las distribuciones basadas en software libre se hacen más maduras, la solución de emplear una combinación de ordenadores de laboratorio con ordenadores personales de los alumnos se va haciendo cada vez más atractiva. En algún momento es concebible pensar en un modelo basado en ordenadores de los alumnos, complementado con algunos ordenadores de laboratorio para los que no dispongan de él, o no hayan podido configurarlo adecuadamente.

Si cada alumno ha de usar sus propios medios, es preciso proporcionales instrucciones detalladas de instalación, y distribuciones específicas preparadas para cubrir las necesidades de la asignatura.

Aunque se podrían concebir modelos diferentes, el método propuesto está basado que en laboratorio (o los ordenadores de los alumnos) están instalados con software libre. Esto permite poder compartir del entorno de desarrollo usado en clase con los alumnos sin ninguna limitación, y en particular que puedan reproducirlo en sus propios ordenadores si así lo desean. Permite también un uso efectivo de los recursos disponibles, al no requerir gasto en licencias ni en la negociación de las mismas, y lo que es más importante, no depender de que estas licencias se renueven o se actualicen en el futuro. Y por último, pero no menos importante, hoy día este entorno permite la utilización de todas las tecnologías necesarias para las asignaturas que nos ocupan, de una forma muy similar a como se haría en cualquier entorno profesional.

Es necesario que los ordenadores que se usen estén disponibles también fuera de los horarios de clases, porque el alumno necesitará consolidar lo aprendido, y explorar nuevos temas, con actividades fuera del aula. También necesitará estos ordenadores para la realización de los proyectos de mayor envergadura que son básicos en el método propuesto. En estos casos será deseable que el alumno pueda usar a su

conveniencia los ordenadores del laboratorio fuera de su horario de clases, o su propio ordenador, de forma lo más indistinta posible.

En el caso de las impartición de las asignaturas que nos ocupan en la URJC, se dispone de varios laboratorios que permiten realizar las clases, con un ordenador por alumno en la mayoría de los grupos<sup>2</sup>.

#### 6.4.2. Técnico de apoyo

La disposición de personal de apoyo técnico de laboratorios es fundamental en estas asignaturas. En el caso de utilizar un laboratorio para las clases, como se ha comentado en el apartado anterior, es clara la necesidad para instalar, mantener y actualizar periódicamente toda la infraestructura. Esta es una labor altamente especializada, que requiere mucha dedicación, y sobre todo precisa esfuerzo de forma urgente en momentos muy precisos. Es muy difícil que el profesor de la asignatura, o el equipo de profesores que se encarguen de ellas, puedan realizar estas tareas sin desatender las que les son propias. Incluso con un esfuerzo de buena voluntad, habrá momentos donde la no disposición del técnico suponga que las clases no puedan tener lugar adecuadamente, o de que los alumnos experimenten problemas serios para realizar sus trabajos.

Pero incluso si no se usa el laboratorio, habrá mucho trabajo que realizar en la preparación de distribuciones que puedan usar los alumnos, en pruebas de aplicaciones y de las propias distribuciones, y en resolución de problemas relacionados con su instalación en los ordenadores de los alumno. De nuevo, aunque estos trabajos los pueden realizar los profesores implicados, es muy difícil que puedan hacerlo de forma adecuada, y con el grado de especialización necesario.

Por último, como se comentará en los apartados siguientes, hay otra infraestructura que mantener, que será necesaria para la asignatura. En algunos casos, puede que esta infraestructura sea ya ofrecida por la Universidad (como es el caso del campus virtual), o puede que se puedan usar servicios gratuitos o de bajo coste proporcionados por terceros (como es el caso de la forja de desarrollo). Pero si no es así, hará falta también personal técnico de apoyo que se encarguen de su instalación, mantenimiento y actualización.

En el caso de las impartición de las asignaturas que nos ocupan en la URJC, se dispone de una persona altamente especializada en el mantenimiento de los laboratorios y las herramientas y sistemas necesarios.

#### 6.4.3. Forja de desarrollo y sistemas anejos

Dado que la asignatura pretende utilizar herramientas modernas de desarrollo de software, y de hecho las utiliza para aspectos concretos como la entrega de prácticas, es preciso disponer de una forja de desarrollo. En el pasado, las asignaturas que nos ocupan han utilizado GitHub para estos fines, ya que es una plataforma ampliamente utilizada (bien en su versión pública, bien en su versión GitHub Enterprise) por la industria, que cubre todas las necesidades de la asignatura.

El curso pasado se desplegó en la URJC una instancia de GitLab, una plataforma de funcionalidades similares, también ampliamente utilizada (aunque no tanto como GitHub) en la industria. Esta instancia podría tener como principal ventaja el control completo sobre ella, sin coste por el servicio o la licencia (es software libre, y el técnico de laboratorio de que se dispone se hace cargo de su mantenimiento). En la actualidad está en evaluación el utilizarla en próximas ediciones de las asignaturas.

Además de la forja, es conveniente disponer de sistemas anejos, como alguno que permita la automatización de pruebas, o el despliegue de herramientas en entornos de producción. Hasta el momento se han utilizado o evaluado sistemas de amplio uso industrial que permiten su uso gratuito (en ciertas circunstancias) como TravisCI (para las pruebas), Heroku o Azure (para el despliegue). Según va evolucionando la tecnología, es de prever que estos sistemas vayan siendo cada vez más importantes para las asignaturas.

#### 6.4.4. Campus virtual y sitio de recursos abiertos

La relación asíncrona con los alumnos, y de ellos entre si, y gran parte del intercambio de información con ellos, necesitará sistemas de soporte específicos. Ambos casos son muy intensos en las asignaturas que nos ocupan, dado el diseño que se les ha dado. Por lo tanto es conveniente que estos sistemas cumplan bien estos objetivos de permitir la relación asíncrona y la transmisión de información.

---

<sup>2</sup><https://labs.gsync.urjc.es/>

En la URJC se usa una instancia de Moodle como campus virtual<sup>3</sup>, en la que se montan las asignaturas, usándose fundamentalmente sus capacidades para compartir documentos, para intercambiar información en foros, para entregar y corregir ejercicios, y para recibir realimentación mediante encuestas.

También se quiere mantener información pública con gran parte de los contenidos de las asignaturas, lo que supone poder montar un sitio web de libre acceso público. Para ello se usa usando un sitio web en GitHub Pages<sup>4</sup>, que proporciona toda la funcionalidad necesaria.

---

<sup>3</sup><https://aulavirtual.urjc.es>

<sup>4</sup><http://cursosweb.github.io/>



## Capítulo 7

# Asignatura: Servicios y Aplicaciones Telemáticas

A continuación se presenta un plan detallado de la asignatura, muy similar a como se entrega a los alumnos, y basado en lo que se impartió en ella durante el curso 2016-2017.

### 7.1. Datos generales

<b>Título:</b>	Servicios y aplicaciones telemáticas
<b>Titulación:</b>	Grado en Ingeniería en Tecnologías de Telecomunicación
<b>Cuatrimestre:</b>	Tercer curso, segundo cuatrimestre
<b>Créditos:</b>	6 (3 teóricos, 3 prácticos)
<b>Horas lectivas:</b>	4 horas semanales
<b>Horario:</b>	martes, 11:00–13:00 jueves, 11:00–13:00
<b>Sedes telemáticas:</b>	<a href="http://aulavirtual.urjc.es/">http://aulavirtual.urjc.es/</a> <a href="http://cursosweb.github.io">http://cursosweb.github.io</a>
<b>Aulas:</b>	Laboratorio 209, Edif. Laboratorios III Aula 125, Edif. Departamental I (sólo presentación)
<b>Número de alumnos:</b>	Habitualmente 30-50

### 7.2. Objetivos

En esta asignatura se pretende que el alumno obtenga conocimientos detallados sobre los servicios y aplicaciones comunes en las redes de ordenadores, y en particular en Internet. Se pretende especialmente que conozcan las tecnologías básicas que los hacen posibles.

Para ello, se ha centrado la asignatura en el estudio de las aplicaciones web que corren en el lado del servidor (*back-end web development*), y típicamente ofrecen una API HTTP, que puede servir páginas HTML o en otros formatos, para consumo bien de personas, bien de otros programas. Se utilizará Python, funcionando en el servidor, para construir estas aplicaciones. De esta forma, esta asignatura complementa a asignaturas previas que describen los protocolos que serán necesarios (fundamentalmente HTTP y todos los que lo apoyan en capas inferiores de la pila de comunicaciones), que muestran cómo programar, y que describen los componentes principales de un ordenador y un sistema operativo. Apoyándose en estos conceptos, la asignatura abrirá el camino a asignaturas posteriores que tratarán el desarrollo de aplicaciones web en el navegador (*front-end web development*).

Las capacidades que se pretenden lograr con esta asignatura están detalladas formalmente en su Guía Docente, pero de forma más práctica, mencionemos aquí que sus objetivos concretos, son conseguir que el alumno consiga:

- Capacidad de comprender el uso del protocolo HTTP para construir aplicaciones web en el lado del servidor.

- Capacidad de comprender cómo se usan mecanismos de HTTP, como las cookies, para resolver los problemas relacionados con el mantenimiento de sesión en aplicaciones web, y de construir software que los use.
- Capacidad de comprender los problemas de tolerancia a fallos y de escalado de las aplicaciones web, y de aplicar las técnicas necesarias para resolverlos.
- Capacidad de comprender y construir APIs HTTP REST, y de modelar con ellas la solución a problemas reales.
- Capacidad de comprender lenguajes marcado de datos como HTML, XML y JSON, y de utilizarlos para resolver problemas reales.
- Comprensión del modelo MVC, y su aplicación a la construcción de aplicaciones web en el lado del servidor. Capacidad de construir aplicaciones reales basadas en él.
- Comprensión de los detalles necesarios para la construcción de aplicaciones web en el lado del servidor, como el despacho según el recurso invocado, el uso de plantillas, el modelado y el acceso a datos persistentes, etc.
- Capacidad de resolver todos los problemas necesarios para construir una aplicación web que resuelva un problema razonablemente real.

### 7.3. Metodología

La asignatura tiene un enfoque eminentemente práctico. Por ello se realizará en la medida de lo posible en el laboratorio, y las prácticas realizadas (incluyendo especialmente la práctica final) tendrán gran importancia en la evaluación de la asignatura. Los conocimientos teóricos necesarios se intercalarán con los prácticos, en gran medida mediante metodologías apoyadas en la resolución de problemas. En las clases teóricas se utilizan, en algunos casos, transparencias que sirven de guión. En todos los casos se recomendarán referencias (usualmente documentos disponibles en Internet) para profundizar conocimientos, y complementarias de los detalles necesarios para la resolución de los problemas prácticos. En el desarrollo diario, las sesiones docentes incluirán habitualmente tanto aspectos teóricos como prácticos.

Se utiliza un sistema de apoyo telemático a la docencia (aula virtual de la URJC) para realizar actividades complementarias a las presenciales, y para organizar parte de la documentación ofrecida a los alumnos. La mayoría de los contenidos utilizados en la asignatura están disponibles o enlazados desde el sitio web CursosWeb. Asimismo, se utiliza el servicio GitHub como repositorio, tanto de los materiales de la asignatura, como para entregar las prácticas por parte de los alumnos.

### 7.4. Evaluación

Parámetros generales:

- Teoría (obligatorio): 0 a 5.
- Microprácticas diarias: 0 a 1
- Miniprácticas preparatorias: 0 a 1
- Práctica final (obligatorio): 0 a 2.
- Opciones y mejoras de la práctica final: 0 a 3
- Nota final: Suma de notas, moderada por la interpretación del profesor
- Mínimo para aprobar:
  - aprobado en teoría (2) y práctica final (1)
  - 5 puntos de nota final

Evaluación teoría: prueba escrita

Evaluación microprácticas diarias (evaluación continua):

- entre 0 y 1
- preguntas y ejercicios en foro y entregados en GitHub
- es muy recomendable hacerlas

Evaluación práctica final:

- posibilidad de examen presencial para práctica final
- tiene que funcionar en el laboratorio
- enunciado mínimo obligatorio supone 1, se llega a 2 sólo con calidad y cuidado en los detalles
- realización individual de la práctica

Opciones y mejoras práctica final:

- permiten subir la nota mucho

Evaluación extraordinaria:

- prueba escrita (si no se aprobó la ordinaria)
- nueva práctica final (si no se aprobó la ordinaria)
- entrega de ejercicios de evaluación continua (con penalización)

## 7.5. Programa detallado

### 7.5.1. 00 - Presentación

Sesión del 17 de enero (2 horas)

- **Presentación:** Presentación de la asignatura. Breve introducción y motivación de las aplicaciones web.
- **Material:** Transparencias, tema “Presentación”.
- **Ejercicio propuesto (entrega en el foro):** “Web 2.0”  
Entrega recomendada: antes del 24 de enero.

Sesión del 24 de enero (0.5 horas)

- **Presentación:** Introducción a la entrega de prácticas en GitHub.

### 7.5.2. 01 - Conceptos básicos de aplicaciones web

Sesión, mantenimiento de estado, persistencia.

Sesión del 19 de enero (2 horas)

- **Ejercicio propuesto (discusión en clase):** “Última búsqueda”  
Se introduce el problema, y se pide que trabajen sobre las interacciones HTTP involucradas. Para la próxima sesión se deja la discusión sobre el uso de cookies (u otros mecanismos) para conseguir la funcionalidad requerida.

**Sesión del 26 de enero (2 horas)**

Páginas dinámicas (diferentes según cómo y cuándo se invocan). Cómo realizar sesiones en HTTP. Profundización en el concepto de sesión, y técnicas para conseguirla, incluyendo cookies y otros mecanismos.

- **Discusión de ejercicio (entrega en el foro):** “Última búsqueda”  
Almacenamiento en el lado del servidor y en el lado del cliente. Relación entre peticiones HTTP. Cookies como herramienta para ambas situaciones.  
Entrega recomendada: antes del 2 de febrero.
- **Ejercicio:** “Espía a tu navegador (Firefox Developer Tools)”
- **Discusión de ejercicio:** “Servicio horario”.

**Sesión del 8 de febrero (2 horas)**

Datos persistentes entre operaciones HTTP diferentes. Concepto de estado persistente frente a caídas del servidor.

- **Presentación:** Cookies
- **Material:** Transparencias, tema “Cookies”
- **Discusión:** Usos de las cookies.  
Uso de las cookies para identificación de visitantes, para autenticación (interacción de autenticación y cookie de sesión posterior), para almacenamiento. Implicaciones de trasladar una cookie de identificación o de sesión de un ordenador a otro. Implicaciones de almacenar datos en el lado del navegador.
- **Ejercicio propuesto:** “Cookies en tu navegador”
- **Ejercicio propuesto:** “Cookies en tu navegador avanzado”
- **Discusión de ejercicio (entrega en el foro):** “Última búsqueda: números aleatorios o consecutivos”  
Entrega recomendada: antes del 15 de febrero.
- **Discusión:** Tres mecanismos básicos para mantenimiento de sesión (cookies, reescritura de URLs, campos ocultos en formularios)
- **Discusión:** Medición de audiencias y visitas únicas por un sitio web.

**Sesión del 9 de febrero (2 horas)**

- **Discusión de ejercicio (entrega en el foro):** “Traza de historiales de navegación por terceras partes”  
Entrega recomendada: antes del 16 de febrero.
- **Ejercicio (discusión en clase, entrega en el foro):** “Trackers en páginas web”  
Entrega recomendada: antes del 16 de febrero.
- **Discusión:** Introducción al problema de los rearranques, y operaciones idempotentes.

**Sesión del 21 de febrero (1 hora)**

- **Discusión de ejercicio:** “Sumador simple con varios navegadores intercalados”  
Trabajo en grupo. Sólo se pretende llegar al pseudocódigo (lo más “estilo Python” que se pueda).
- **Discusión de ejercicio (entrega en el foro):** “Sumador simple con rearranques”.  
Entrega recomendada: antes del 28 de febrero.



### 7.5.3. 02 - Servicios web que interoperan

Invocaciones a aplicaciones web desde aplicaciones web. Servicios web como un conjunto de aplicaciones que interoperan.

#### Sesión del 23 de febrero (2 horas)

- **Presentación:** Introducción breve a la arquitectura REST
- **Material:** Transparencias, tema “REST”
- **Ejercicio (discusión en clase, entrega en el foro):** “Listado de lo que tengo en la nevera”. Trabajo en grupos y discusión de los detalles del ejercicio.  
Entrega recomendada: antes del 2 de marzo.

#### Sesión del 2 de marzo (2 horas)

- **Presentación:** Arquitectura REST
- **Material:** Transparencias, tema “REST”
- **Ejercicio (discusión en clase, entrega en el foro):** “Aplicación redirectora”.  
Entrega recomendada: antes del 9 de marzo.  
Repo GitHub: <https://github.com/CursosWeb/X-Serv-App-Redirectora>

#### Sesión del 9 de marzo (2 horas)

- **Ejercicio (discusión en clase, entrega en el foro):** “Calculadora simple versión REST”.  
Entrega recomendada: antes del 16 de marzo.  
Repo GitHub: <https://github.com/CursosWeb/X-Serv-App-Calculadora-REST>
- **Discusión de ejercicio:** “Cache de contenidos”.  
Trabajo en grupos y discusión de los detalles del ejercicio.

#### Sesión del 16 de marzo (2 horas)

- **Ejercicio (entrega en el foro):** “Cache de contenidos anotado”.  
Entrega recomendada: antes del 23 de marzo.  
Repo GitHub: <https://github.com/CursosWeb/X-Serv-App-Cache-Anotada>

### 7.5.4. 03 - Modelo-vista-controlador

Explicación del patrón de diseño “modelo-vista-controlador”.

#### Sesión del 23 de marzo (2 horas)

- **Presentación:** “Modelo-vista-controlador”.
- **Material:** Transparencias “Modelo-vista-controlador”.
- **Presentación:** “Componentes de aplicaciones Django y MVC”. Repaso de los componentes principales de una aplicación Django y su relación con el patrón modelo-vista-controlador.

### 7.5.5. 04 - Introducción a XML

Uso de XML en aplicaciones web.

**Sesión del 30 de marzo (2 horas)**

- **Presentación:** “Introducción a XML”.  
Introducción a XML, sintaxis básica, formas de especificar gramáticas, reconocedores SAX y DOM.
- **Presentación:** “Introducción a XML”.  
Usos de XML, canales semánticos usando RSS y similares. Complementos de XML, JSON.
- **Material:** Transparencias “Introducción a XML”.
- **Ejercicio (discusión en clase):** “Chistes XML”.
- **Ejercicio (entrega en GitHub):** “Titulares de BarraPunto”.  
Entrega recomendada: antes del 6 de abril.  
Repo GitHub: <https://github.com/CursosWeb/X-Serv-XML-Barrapunto>

**Sesión del 6 de abril (2 horas)**

- **Ejercicio (discusión en clase):** “Gestor de contenidos con titulares de BarraPunto”.
- **Ejercicio (entrega en GitHub):** “Gestor de contenidos con titulares de BarraPunto versión Django”.  
Entrega recomendada: antes del 13 de abril.  
Repo GitHub: <https://github.com/CursosWeb/X-Serv-XML-ContentApp-Barrapunto>

**7.5.6. 05 - Hojas de estilo CSS**

Hojas de estilo CSS, separación entre contenido y presentación.

**Sesión del 20 de abril (2 horas)**

Hojas de estilo CSS, y su uso para manejar la apariencia de las páginas HTML.

- **Presentación:** “Hojas de estilo CSS”. Introducción a CSS. Principales elementos.
- **Material:** Transparencias, tema “CSS”.
- **Demo:** Inspección de datos de aspecto y hojas CSS con Firebug en Firefox.
- **Ejercicio (entrega en el foro):** “Django cms\_css simple”.  
Entrega recomendada: antes del 27 de abril.

**Sesión del 27 de abril (2 horas)**

Hojas de estilo CSS, y su uso para manejar la apariencia de las páginas HTML.

- **Presentación:** “Hojas de estilo CSS”. Otra información relacionada con CSS.
- **Material:** Transparencias, tema “CSS”.
- **Ejercicio (entrega en GitHub):** “Django cms\_css elaborado”.  
Entrega recomendada: antes del 4 de mayo.  
Repo GitHub: <https://github.com/CursosWeb/X-Serv-CSS-Elaborado>

**7.5.7. 06 - Ajax**

Introducción a Ajax, mashups y otros tipos de aplicaciones web con código en el lado del cliente.

**Sesión del 29 de abril (2 horas)**

- **Presentación:** Aplicaciones web con código en el lado del cliente. DHTML, SPA, AJAX
- **Material:** Transparencias de la asignatura, tema “AJAX”.
- **Ejercicio (discusión en clase):** “SPA Sentences generator”.
- **Ejercicio (discusión en clase):** “Ajax Sentences generator”.

**7.5.8. P1 - Introducción a Python**

Introducción al lenguaje de programación Python, que se utilizará para la realización de las prácticas de la asignatura.

**Sesión del 24 de enero (1.5 horas)**

- **Presentación:** “Introducción a Python” (introducción, entorno de ejecución, características básicas del lenguaje, ejecución en el intérprete, strings, listas, estructuras condicionales (if else), bucles for).
- **Material:** Transparencias “Introducción a Python”
- **Material:** Ejercicios “Uso interactivo del intérprete de Python” y “Haz un programa en Python”.
- **Ejercicio propuesto (entrega en foro):** “Tablas de multiplicar”. Entrega recomendada: antes del 31 de enero.

**Sesión del 31 de enero (2 horas)**

- **Presentación:** “Introducción a Python”.
- **Material:** Transparencias “Introducción a Python”
- **Material:** Ejercicio “Ficheros y listas”.
- **Ejercicio propuesto (entrega en GitHub):** “Ficheros, diccionarios y excepciones”. Entrega recomendada: antes del 7 de febrero.

**Sesión del 7 de febrero (1 hora)**

- **Presentación:** “Introducción a Python”.
- **Material:** Transparencias “Introducción a Python”
- **Ejercicio propuesto (discusión en clase):** “Calculadora”.

**7.5.9. P2 - Aplicaciones web simples**

Construcción de aplicaciones web mínimas sobre la biblioteca Sockets de Python.

**Sesión del 7 de febrero (1 hora)**

- **Ejercicio:** “Aplicación web hola mundo”  
Se muestra la solución del ejercicio, y se comenta en clase. Se pide a los alumnos que lo ejecute, lo modifiquen y se fijen en las cabeceras HTTP enviadas por el cliente y que el servidor muestra en pantalla (pero no hay entrega específica).
- **Ejercicio propuesto (entrega en GitHub):** “Aplicación web generadora de URLs aleatorias”  
Entrega recomendada: antes del 14 de febrero.

**Sesión del 14 de febrero (2 horas)**

- **Explicación de ejercicio:** “Aplicación web generadora de URLs aleatorias”
- **Ejercicio:** “Sumador simple”  
Entrega recomendada: antes del 28 de febrero.

**Sesión del 16 de febrero (2 horas)**

- **Trabajo y explicación del ejercicio:** “Clase servidor de aplicaciones”.  
Explicación de la estructura general que tienen las aplicaciones web, y fundamentos de cómo esta estructura se puede encapsular en una clase.
- **Ejercicio propuesto:** “Clase servidor de aplicaciones, sumador”.  
Entrega recomendada: antes del 28 de febrero.

**7.5.10. P3 - Servidores simples de contenidos**

Construcción de algunos servidores de contenidos que permitan comprender la estructura básica de una aplicación web, y de cómo implementarlos aprovechando algunas características de Python.

**Sesión del 28 de febrero (2 horas)**

- **Ejercicio propuesto (entrega en GitHub):** “Clase servidor de aplicaciones, generador de URLs aleatorias”.
- **Ejercicio propuesto:** “Clase servidor varias aplicaciones”  
Explicación de la estructura principal de una clase que gestiona varias aplicaciones (o varios recursos, cada uno manejado por una aplicación)
- **Ejercicio propuesto:** “Clase servidor, cuatro aplis”.
- Presentación de la primera práctica de entrega voluntaria. Entrega en GitHub. Fecha de entrega: antes del 7 de marzo.

**Sesión del 7 de marzo (2 horas)**

- **Ejercicio:** “Clase contentApp”  
Explicación de la estructura principal de una aplicación que sirve contenidos previamente almacenados.
- **Ejercicio:** “Instalación y prueba de Poster”.
- **Ejercicio:** “Clase contentPutApp”.
- **Ejercicio:** “Herramientas de Web Developer”.
- **Ejercicio:** “Clase contentPostApp”.

**7.5.11. P4 - Introducción a Django****Sesión del 14 de marzo (2 horas)**

Presentación de Django como sistema de construcción de aplicaciones web.

- **Presentación:** Introducción a Django (primera parte)
- **Ejercicio:** “Instalación de Django”.
- **Ejercicio:** “Django Intro”.
- **Material:** Transparencias “Introducción a Django”

- **Ejercicio (discusión en clase:** “Django Primera Aplicación”.
- **Ejercicio (entrega en GitHub:** “Django calc”.  
Entrega recomendada: antes del 21 de marzo.

#### Sesión del 21 de marzo (2 horas)

Primeros ejercicios con base de datos.

- **Presentación:** Introducción a Django (segunda parte)
- **Material:** Transparencias “Introducción a Django”
- **Ejercicio:** “Django cms”.
- **Ejercicio (entrega en GitHub):** “Django cms\_put”.  
Entrega recomendada: antes del 28 de marzo.

#### Sesión del 28 de marzo (2 horas)

Usuarios, administración y autenticación con Django.

- **Presentación:** Introducción a Django (tercera parte)
- **Material:** Transparencias “Introducción a Django”
- **Ejercicio (discusión en clase):** “Django cms\_put”.  
Entrega recomendada: antes del 4 de abril.

#### Sesión del 4 de abril (2 horas)

- **Presentación:** Introducción a Django (cuarta parte)
- **Material:** Transparencias “Introducción a Django”
- **Ejercicio (discusión en GitHub):** “Django cms\_users”.
- **Ejercicio (entrega en GitHub):** “Django cms\_users\_put”.
- **Ejercicio (entrega en GitHub):** “Django cms\_templates”.  
Entrega recomendada: antes del 18 de abril.

#### Sesión del 18 de abril (2 horas)

Fin de introducción a Django, incluyendo formularios (templates).

- **Presentación:** Introducción a Django (cuarta parte)
- **Material:** Transparencias “Introducción a Django”
- **Ejercicio:** “Django cms\_post”.
- Presentación de la **Práctica 2**  
Entrega recomendada: antes del 25 de abril.

#### Sesión del 25 de abril (2 horas)

Uso de módulos externos con Django

- **Presentación:** Posibilidades de feedparser.py
- **Presentación:** Posibilidades de BeautifulSoup.py
- **Ejercicio (entrega en GitHub):** “Django feed\_expander”.  
Entrega recomendada: antes del 3 de mayo.
- **Presentación:** Práctica final.)



## Capítulo 8

# Asignatura: Aplicaciones Telemáticas

A continuación se presenta un plan detallado de la asignatura, muy similar a como se entrega a los alumnos, y basado en lo que se impartió en ella durante el curso 2016-2017.

### 8.1. Datos generales

<b>Título:</b>	Aplicaciones Telemáticas
<b>Titulación:</b>	Grado en Ingeniería Telemática
<b>Cuatrimestre:</b>	Cuarto curso, segundo cuatrimestre
<b>Créditos:</b>	6 (3 teóricos, 3 prácticos)
<b>Horas lectivas:</b>	4 horas semanales
<b>Horario:</b>	miércoles, 13:00–15:00 jueves, 13:00–15:00
<b>Sedes telemáticas:</b>	<a href="http://aulavirtual.urjc.es/">http://aulavirtual.urjc.es/</a> <a href="http://cursosweb.github.io">http://cursosweb.github.io</a>
<b>Aulas:</b>	Laboratorio 209, Edif. Laboratorios III Aula 044, Edif. Departamental I (sólo presentación)
<b>Número de alumnos:</b>	Habitualmente 25-40

### 8.2. Objetivos

En esta asignatura se pretende que el alumno obtenga conocimientos detallados sobre los servicios y aplicaciones comunes en las redes de ordenadores, y en particular en Internet. Se pretende especialmente que conozcan las tecnologías básicas que los hacen posibles.

Para ello, se ha centrado la asignatura en el estudio de las aplicaciones web que corren en el navegador (*front-end web development*), y típicamente utilizan APIs HTTP proporcionadas por servicios web. Se utilizará JavaScript funcionando en el navegador para construir las aplicaciones. De esta forma, esta asignatura complementa a asignaturas previas que describen los protocolos que serán necesarios (fundamentalmente HTTP y todos los que lo apoyan en capas inferiores de la pila de comunicaciones), que muestran cómo programar, y que describen los componentes principales de un ordenador y un sistema operativo. Apoyándose en estos conceptos, y en asignaturas previas que introducen el desarrollo de aplicaciones web en el servidor (*back-end web development*), se hará un esfuerzo por “encajar todas las piezas del puzzle”, y construir aplicaciones muy similares a las reales que se puede encontrar cualquier usuario web, si bien simplificadas.

Las capacidades que se pretenden lograr con esta asignatura están detalladas formalmente en su Guía Docente, pero de forma más práctica, mencionemos aquí que sus objetivos concretos, son conseguir que el alumno consiga:

- Capacidad de comprender los módulos principales de un navegador web moderno (máquina JavaScript, módulo HTTP, módulo de presentación para renderizar HTTP y CSS, modelo de eventos, etc.).

- Capacidad de construir aplicaciones en el navegador, utilizando la funcionalidad proporcionada por la JSAPI.
- Capacidad de construir aplicaciones en el navegador, utilizando la funcionalidad proporcionada por bibliotecas que abstraen la JSAPI.
- Capacidad de comprender los módulos principales que proporciona el navegador para la construcción de aplicaciones, de acuerdo al estándar HTML5, y de utilizarlas para resolver problemas reales.
- Capacidad de comprender la funcionalidad que proporcionan las versiones modernas de CSS, y de utilizarlas para diseñar la presentación de aplicaciones.
- Capacidad de comprender cómo usar API de servicios, de descubrir su documentación asociada, de probarlas y de aprovecharlas en la construcción de aplicaciones.
- Capacidad de utilizar los mecanismos habituales hoy día en aplicaciones web para resolver problemas como los de autenticación, mantenimiento de sesión, incorporación de datos de sitios terceros, etc, y construir aplicaciones que utilicen estos mecanismos.
- Capacidad de utilizar bibliotecas de alto nivel para la construcción de aplicaciones web complejas.
- Capacidad de resolver todos los problemas necesarios para construir una aplicación web que resuelva un problema razonablemente real.

### 8.3. Metodología

La asignatura tiene un enfoque eminentemente práctico. Por ello se realizará en la medida de lo posible en el laboratorio, y las prácticas realizadas (incluyendo especialmente la práctica final) tendrán gran importancia en la evaluación de la asignatura. Los conocimientos teóricos necesarios se intercalarán con los prácticos, en gran medida mediante metodologías apoyadas en la resolución de problemas. En las clases teóricas se utilizan, en algunos casos, transparencias que sirven de guión. En todos los casos se recomendarán referencias (usualmente documentos disponibles en Internet) para profundizar conocimientos, y complementarias de los detalles necesarios para la resolución de los problemas prácticos. En el desarrollo diario, las sesiones docentes incluirán habitualmente tanto aspectos teóricos como prácticos.

Se usa un sistema de apoyo telemático a la docencia (aula virtual de la URJC) para realizar actividades complementarias a las presenciales, y para organizar parte de la documentación ofrecida a los alumnos. La mayoría de los contenidos utilizados en la asignatura están disponibles o enlazados desde el sitio web CursosWeb. Asimismo, se utiliza el servicio GitHub como repositorio, tanto de los materiales de la asignatura, como para entregar las prácticas por parte de los alumnos.

### 8.4. Evaluación

#### Parámetros generales:

- Microprácticas diarias: 0 a 1
  - Entrega en GitHub o en foro del aula virtual
  - Evaluación continua
- Miniprácticas preparatorias: 0 a 2
  - Evaluación: funcionamiento
  - Consideración especial de las extensiones
- Práctica final (obligatorio): 0 a 5.
  - Práctica básica: 0 a 2
  - Opciones y mejoras: 0 a 3



- Evaluación: funcionamiento e implementación
- Entrega telemática de materiales solicitados
- Puede haber presentación presencial
- Teoría (obligatorio): 0 a 4.
  - Evaluación: prueba escrita
- Nota final: Suma de notas, moderada por la interpretación del profesor
- Mínimo para aprobar:
  - aprobado en teoría (2.5) y práctica final (1), y
  - 5 puntos de nota final

**Consideraciones sobre la práctica final:**

- posibilidad de examen presencial para práctica final
- tiene que funcionar en el laboratorio
- enunciado mínimo obligatorio supone 1, se llega a 2 sólo con calidad y cuidado en los detalles
- realización individual de la práctica
- las opciones y mejoras práctica final permiten subir la nota mucho

**Evaluación extraordinaria:**

- prueba escrita (si no se aprobó la ordinaria)
- nueva práctica final (si no se aprobó la ordinaria)

## 8.5. Programa detallado

### 8.5.1. Presentación

**Sesión del 16 de enero (2 horas)**

- **Presentación:** Presentación de la asignatura. Breve introducción y motivación de las aplicaciones web.
- **Material:** Transparencias, tema “Presentación”.

**Sesión del 17 de enero (0.5 horas)**

- **Presentación:** Introducción a la entrega de prácticas en GitHub.

### 8.5.2. Introducción a HTML

Introducción a algunos conceptos de HTML y tecnologías relacionadas.

**Sesión del 23 de enero (2 horas)**

- **Ejercicio (discusión en clase):** “Página HTML simple”.
- **Presentación:** Introducción a HTML
- **Material:** Transparencias, tema “HTML”.
- **Ejercicio libre:** Practica con HTML. Elige una página web y modificala (trata de hacerlo para todos los elementos que puedas entre los vistos en la presentación).
- **Ejercicio (discusión en clase):** “Manipulación de HTML desde Firebug”.

### 8.5.3. Introducción a CSS

Introducción a algunos conceptos de CSS.

#### Sesión del 17 de enero (1.5 horas)

- **Presentación:** Introducción a CSS
- **Material:** Transparencias, tema “CSS”.
- **Ejercicio (discusión en clase):** “Añadir selectores”.
- **Ejercicio (en clase):** “Tipografía”.
- **Ejercicio (entrega en el foro):** “Una caja CSS2”.

#### Sesión del 24 de enero (2 horas)

- **Presentación:** CSS3
- **Material:** Transparencias, tema “CSS3”.
- **Ejercicio (discusión en clase):** “Una caja CSS2”.
- **Ejercicio (discusión en clase):** “Bordes redondeados”.
- **Ejercicio (discusión en clase):** “Sombra de texto”.
- **Ejercicio (discusión en clase):** “Sombra de borde”.
- **Ejercicio (discusión en clase):** “Fondo semitransparente”.
- **Ejercicio (discusión en clase):** “Fondo en gradiente”.
- **Ejercicio (entrega en el foro):** “Alpha en los bordes”.
- **Ejercicio (discusión en clase):** “Alpha en los bordes”.
- **Ejercicio (discusión en clase):** “Rotación”.
- **Ejercicio (discusión en clase):** “Escalado”.
- **Ejercicio (discusión en clase):** “Rotación en el eje Y”.
- **Ejercicio (discusión en clase):** “Animación”.
- **Ejercicio (entrega en el foro):** “Transiciones”.

### 8.5.4. Introducción a Bootstrap

#### Sesión del 31 de enero (2 horas)

- **Presentación:** Bootstrap
- **Material:** Transparencias, tema “Bootstrap”.
- **Ejercicio (discusión en clase):** “Inspeccionando Bootstrap”.
- **Ejercicio (discusión en clase):** “Una sencilla página con Bootstrap”.
- **Ejercicio (entrega en GitHub):** “Utilizando el Carousel de Bootstrap”.

**Sesión del 7 de febrero (2 horas)**

- **Presentación:** Bootstrap
- **Material:** Transparencias, tema “Bootstrap”.
- **Ejercicio:** “El *grid* de Bootstrap”.
- **Ejercicio:** “Bootstrap responsivo”.
- **Ejercicio (entrega en GitHub):** Concurso: “Tu diseño Bootstrap”.

**8.5.5. Introducción a JavaScript****Sesión del 30 de enero (2 horas)**

- **Presentación:** JavaScript: objetos, tipos, strings, números, booleanos, vectores (arrays), variables.
- **Material:** Transparencias, tema “JavaScript”
- **Ejercicios:** Ejercicios varios ejecutados en la consola de Firebug. Exploración de las opciones de depuración de Firebug para JavaScript.

**Sesión del 6 de febrero (2 horas)**

- **Presentación:** JavaScript: funciones.
- **Material:** Transparencias, tema “JavaScript”
- **Ejercicio (discusión en clase y entrega en el foro):** “Página HTML con JavaScript”.
- **Ejercicio (discusión en clase y entrega en el foro):** “Iteración sobre un objeto”.
- **Ejercicio (discusión en clase):** “Función que cambia un elemento HTML”.
- **Ejercicio (discusión en clase):** “Vacía página”.

**Sesión del 20 de febrero (2 horas)**

- **Presentación:** JavaScript: closures.
- **Material:** Transparencias, tema “JavaScript”
- **Ejercicio (discusión en clase):** “De lista a lista ordenada”.
- **Ejercicio (discusión en clase):** “Sumador JavaScript muy simple”.
- **Ejercicio (discusión en clase, entrega en GitHub):** “Sumador JavaScript muy simple con sumas aleatorias”.  
Entrega recomendada: antes del 27 de febrero.  
Repo GitHub: <https://github.com/CursosWeb/X-Nav-JS-Sumador>
- **Ejercicio (discusión en clase):** “JSFIDDLE”.

**Sesión del 21 de febrero (2 horas)**

- **Ejercicio (discusión en clase, entrega en GitHub):** “Mostrador aleatorio de imágenes”.  
Entrega recomendada: antes del 28 de febrero  
Repo GitHub: <https://github.com/CursosWeb/X-Nav-JS-Fotos>
- **Ejercicio (discusión en clase, entrega en GitHub):** “Calculadora binaria simple”.  
Entrega recomendada: antes del 28 de febrero  
Repo GitHub: <https://github.com/CursosWeb/X-Nav-JS-Calculadora>
- **Ejercicio (discusión en clase):** “Greasemonkey”.

**Sesión del 27 de febrero (2 horas)**

- **Ejercicio (discusión en clase):** “Prueba de addEventListener para leer contenidos de formularios”.
- **Ejercicio (discusión en clase, entrega en GitHub):** “Colores con addEventListener”.  
Entrega recomendada: antes del 6 de marzo  
Repo GitHub: <https://github.com/CursosWeb/X-Nav-JS-Event>

**8.5.6. Introducción a jQuery****Sesión del 6 de marzo (2 horas)**

- **Presentación:** JQuery: introducción
- **Material:** Transparencias, tema “jQuery”
- **Ejercicio (discusión en clase):** “Uso de jQuery”.
- **Ejercicio (discusión en clase, entrega en Git Hub):** “Cambio de colores con jQuery”.  
Entrega recomendada: antes del 13 de marzo.  
Repo GitHub: <https://github.com/CursosWeb/X-Nav-JQ-Colores>

**Sesión del 13 de marzo (2 horas)**

- **Presentación:** jQuery: continuamos
- **Material:** Transparencias, tema “jQuery”
- **Ejercicio (discusión en clase):** “Texto con jQuery”.
- **Ejercicio (discusión en clase):** “Difuminado (fading) con jQuery”.
- **Presentación de minipráctica preparatoria:** “Calculadora SPA”.  
Entrega recomendada: antes del 24 de marzo.  
Repo GitHub: <https://github.com/CursosWeb/X-Nav-Practica-Calculadora>

**Sesión del 27 de marzo (2 horas)**

- **Presentación:** jQuery: AJAX. Historia, motivación, el objeto XMLHttpRequest. Uso de AJAX desde jQuery.
- **Presentación:** JSON, AJAX con JSON y uso de AJAX con JSON desde jQuery.
- **Material:** Transparencias, tema “jQuery”

- **Ejercicio (discusión en clase):** “Ejemplos simples con Ajax”.  
Entrega recomendada: antes del 3 de abril.  
Repo GitHub: <https://github.com/CursosWeb/X-Nav-JQ-Ajax>
- **Ejercicio (discusión en clase):** “Ejemplos simples con Ajax y JSON”.
- **Ejercicio (discusión en clase):** “Generador de frases aleatorias”.

#### Sesión del 3 de abril (1 hora)

- **Presentación de minipráctica preparatoria:** “Socios”.  
Entrega recomendada: antes del 17 de abril.  
Repo GitHub: <https://github.com/CursosWeb/X-Nav-Practica-Socios>
- **Material:** Transparencias, tema “jQuery”
- **Ejercicio (entrega en el foro):** “Utilización de JSONP”.  
Entrega recomendada: antes del 10 de abril.  
Repo GitHub: <https://github.com/CursosWeb/X-Nav-JQ-Flickr>

### 8.5.7. Introducción a HTML5

Introducción a algunos conceptos de HTML5.

#### Sesión del 13 de febrero (2 horas)

- **Presentación:** HTML5: introducción
- **Material:** Transparencias, tema “HTML5”
- **Ejercicio (discusión en clase):** “Un sencillo Paint”.
- **Ejercicio (discusión en clase):** “Un sencillo Paint con brocha”.  
Entrega recomendada: antes del 20 de febrero.

#### Sesión del 14 de febrero (2 horas)

- **Presentación:** HTML5: Guardar en local; aplicaciones sin conexión
- **Material:** Transparencias, tema “HTML5”
- **Ejercicio (discusión en clase):** “Un sencillo juego con canvas”.
- **Ejercicio (entrega en GitHub):** “Mejora el juego con canvas”.
- **Ejercicio (discusión en GitHub):** “Juego con estado”.  
Entrega recomendada: antes del 21 de febrero.

#### Sesión del 28 de febrero (2 horas)

- **Presentación:** HTML5: Otras cuestiones HTML5
- **Material:** Transparencias, tema “HTML5”
- **Ejercicio:** “Modernizr: Comprobación de funcionalidad HTML5”.
- **Ejercicio (entrega en GitHub):** “Geolocalización con HTML5”.  
Entrega recomendada: antes del 7 de marzo.

**Sesión del 7 de marzo (2 horas)**

- **Presentación:** HTML5: Web Workers
- **Material:** Transparencias, tema “HTML5”
- **Ejercicio (entrega en GitHub):** “Cálculo de números primos con Web Workers”.  
Entrega recomendada: antes del 14 de marzo.

**Sesión del 14 de marzo (2 horas)**

- **Presentación:** HTML5: History API
- **Ejercicio (entrega en GitHub):** “History API - Cambiando la historia con HTML5”.  
Entrega recomendada: antes del 21 de marzo.

**Sesión del 21 de marzo (2 horas)**

- **Presentación:** HTML5: WebSocket
- **Material:** Transparencias, tema “HTML5”
- **Ejercicio:** “Cliente de eco con WebSocket”.
- **Ejercicio:** “Cliente y servidor de eco con WebSocket”.
- **Ejercicio:** “Cliente y servidor de chat con WebSocket”.
- **Ejercicio (entrega en GitHub):** “Canal con información horaria”.  
Entrega recomendada: antes del 28 de marzo.

**8.5.8. Otras bibliotecas JavaScript****Sesión del 3 de abril (1 hora)**

- **Presentación:** JQueryUI: introducción
- **Ejercicio (discusión en clase):** “JQueryUI: Instalación y prueba”.
- **Ejercicio (discusión en clase):** “JQueryUI: Uso básico”.
- **Ejercicio (opcional, sin entrega):** “Elige un plugin de jQuery”.
- **Ejercicio (entrega en el foro):** “JQueryUI: Juega con JQueryUI”.  
Entrega recomendada: antes del 10 de abril.  
Repo GitHub: <https://github.com/CursosWeb/X-Nav-JSLibs-JQueryUI>
- **Ejercicio (optativo):** “JQueryUI: Clon de 2048”  
Entrega recomendada: antes del 10 de abril.  
Repo GitHub: <https://github.com/CursosWeb/X-Nav-JSLibs-2048>

**8.5.9. APIs JavaScript****Sesión del 17 de abril (2 horas)**

- **Presentación:** Leaflet: introducción
- **Ejercicio (discusión en clase):** “Leaflet: Instalación y prueba”.
- **Ejercicio (discusión en clase):** “Leaflet: Coordenadas”.
- **Ejercicio (entrega en el foro):** “Leaflet: Aplicación móvil”.  
Entrega recomendada: antes del 25 de abril. Repo GitHub: <https://github.com/CursosWeb/X-Nav-APIs-Leaflet>

**Sesión del 24 de abril (2 horas)**

- **Ejercicio (discusión en clase):** “Leaflet: Coordenadas y búsqueda de direcciones”.
- **Ejercicio (entrega en GitHub):** “Leaflet: Fotos de Flickr”.  
Entrega recomendada: antes del 1 de mayo.  
Repo GitHub: <https://github.com/CursosWeb/X-Nav-APIs-Map-Flickr>
- **Ejercicio (discusión en clase):** “GitHub.js: Datos de un repositorio”.
- **Ejercicio (entrega en GitHub):** “GitHub.js: Crea un fichero”.  
Entrega recomendada: antes del 1 de mayo.  
Repo GitHub: <https://github.com/CursosWeb/X-Nav-APIs-GitHub-Fichero>

**8.5.10. APIs de Google****Sesión del 28 de marzo (2 horas)**

- **Presentación:** La API de los servicios de Google
- **Material:** Transparencias, tema “APIs de Google”
- **Ejercicio:** “Conociendo la Google API Console”.
- **Ejercicio:** “Tu Perfil vía la API de Google+”.
- **Ejercicio (entrega en GitHub):** “Tomando datos de la API de Google+”.  
Entrega recomendada: antes del 4 de abril.

**8.5.11. Presentación de la Práctica final**

- **Presentación de proyecto final:** “Wifi abierta en instalaciones municipales en Madrid”.






## Capítulo 9

# Ejemplo de transparencias

En este anexo se muestra un ejemplo de transparencias para un tema teórico. Concretamente, se han elegido las transparencias que se han usado en la asignatura de Redes de la titulación de Ingeniería Técnica en Informática de Sistemas de la URJC en el curso 2001-2002.

<p>Cookies HTTP</p>  <p><b>Cookies HTTP</b></p>	<p>Presentación de la asignatura</p> <h2>Cookies</h2> <ul style="list-style-type: none"> <li>• Forma de mantener estado en un protocolo sin estado (HTTP)</li> <li>• Forma de almacenar datos en el lado del cliente</li> <li>• Dos versiones principales:             <ul style="list-style-type: none"> <li>• Versión 0 (Netscape)</li> <li>• Version 1 (RFC 2109 / RFC 2965)</li> </ul> </li> </ul>
<p>GSyC (<a href="http://cursosweb.github.io/">http://cursosweb.github.io/</a>) Servicios y Aplicaciones Telemáticas 2015-16 19 de enero de 2016 21 / 324</p> <p>Presentación de la asignatura</p>	<p>GSyC (<a href="http://cursosweb.github.io/">http://cursosweb.github.io/</a>) Servicios y Aplicaciones Telemáticas 2015-16 19 de enero de 2016 22 / 324</p> <p>Presentación de la asignatura</p>
<h3>Cabeceras HTTP para cookies (version 0)</h3> <ul style="list-style-type: none"> <li>• Set-Cookie: De servidor a navegador             <pre>Set-Cookie: statusmessages="deleted"; Path=;/ Expires=Wed, 31-Dec-97 23:59:59 GMT</pre> <ul style="list-style-type: none"> <li>• Nombre de la cookie: statusmessages</li> <li>• Valor: "deleted"</li> <li>• Path: / (todo el sitio del que se recibió)</li> <li>• Expira: 31-Dec-97 23:59:59 GMT</li> </ul> </li> <li>• Cookie: De navegador a servidor             <pre>Cookie: statusmessages="deleted"</pre> <ul style="list-style-type: none"> <li>• Nombre de la cookie: statusmessages</li> <li>• Valor: "deleted"</li> </ul> </li> </ul> <p>RFC 2965 (version 1) tiene: "Cookie", "Cookie2" y "Set-Cookie2"</p>	<h3>Estructura de Set-Cookie</h3> <ul style="list-style-type: none"> <li>• Nombre y valor (obligatorios)             <ul style="list-style-type: none"> <li>• Uso normal: "nombre=valor"</li> <li>• También valor nulo: "nombre="</li> </ul> </li> <li>• Fecha de expiración             <ul style="list-style-type: none"> <li>• "Expires=fecha"</li> <li>• Si no tiene, cookie no persistente (sólo en memoria)</li> </ul> </li> <li>• Path: camino para el que es válida             <ul style="list-style-type: none"> <li>• "Path=/camino"</li> <li>• Prefijo de caminos válidos</li> </ul> </li> <li>• Domain: dominio para el que es válida             <ul style="list-style-type: none"> <li>• El servidor ha de estar en el dominio indicado</li> <li>• Si no se indica, servidor que sirve la cookie</li> </ul> </li> <li>• Seguridad: se necesita SSL para enviar la cookie             <ul style="list-style-type: none"> <li>• Campo "Secure"</li> </ul> </li> <li>• Campos separados por ","</li> </ul>
<p>GSyC (<a href="http://cursosweb.github.io/">http://cursosweb.github.io/</a>) Servicios y Aplicaciones Telemáticas 2015-16 19 de enero de 2016 23 / 324</p>	<p>GSyC (<a href="http://cursosweb.github.io/">http://cursosweb.github.io/</a>) Servicios y Aplicaciones Telemáticas 2015-16 19 de enero de 2016 24 / 324</p>

<p style="text-align: right; font-size: small;">Presentación de la asignatura</p> <h3 style="text-align: left; background-color: #1a3a7a; color: white; padding: 2px;">Estructura de Cookie</h3>	<p style="text-align: right; font-size: small;">Presentación de la asignatura</p> <h3 style="text-align: left; background-color: #1a3a7a; color: white; padding: 2px;">Límites para las cookies</h3>
<ul style="list-style-type: none"> <li>• Lista de pares nombre - valor</li> <li>• Cada par corresponde a un "Set-Cookie"</li> <li>• Se envían las cookies válidas para el dominio y el path de la petición HTTP</li> <li>• Si no se especificó dominio en "Set-Cookie", el del servidor</li> <li>• Si no se especificó camino en "Set-Cookie", todo</li> </ul> <p>Cookie: user=jgb; last=5; edited=False</p>	<ul style="list-style-type: none"> <li>• Originalmente: 20 cookies del mismo dominio</li> <li>• La mayoría de los navegadores: 30 o 50 cookies del mismo dominio</li> <li>• Cada cookie: como mucho 4 Kbytes</li> </ul>
<p style="font-size: x-small;">GSyC (<a href="http://cursosweb.github.io/">http://cursosweb.github.io/</a>) Servicios y Aplicaciones Telemáticas 2015-16 19 de enero de 2016 25 / 324</p> <p style="text-align: right; font-size: x-small;">Presentación de la asignatura</p>	<p style="font-size: x-small;">GSyC (<a href="http://cursosweb.github.io/">http://cursosweb.github.io/</a>) Servicios y Aplicaciones Telemáticas 2015-16 19 de enero de 2016 26 / 324</p> <p style="text-align: right; font-size: x-small;">Presentación de la asignatura</p>
<h3 style="text-align: left; background-color: #1a3a7a; color: white; padding: 2px;">Gestión de sesión en HTTP</h3>	<h3 style="text-align: left; background-color: #1a3a7a; color: white; padding: 2px;">Referencias</h3>
<ul style="list-style-type: none"> <li>• Mediante cookies: normalmente, identificador de sesión en la cookie Set-Cookie: session=ab34cd-34fd3a-ef2365</li> <li>• Reescritura de urls: se añade identificador a la url http://sitio.com/path;session=ab34cd-34fd3a-ef2365</li> <li>• Campos escondidos en formularios HTML</li> </ul> <pre style="font-family: monospace; font-size: small;"> &lt;form method="post" action="http://sitio.com/path"&gt;   &lt;input type="hidden" name="session" value="ab34cd-34fd3a-ef2365"&gt;   ...   &lt;input type="submit"&gt; &lt;/form&gt;</pre>	<ul style="list-style-type: none"> <li>• Persistent Client State HTTP Cookies (especificación original de Netscape) <a href="http://curl.haxx.se/rfc/cookie_spec.html">http://curl.haxx.se/rfc/cookie_spec.html</a></li> <li>• RFC 2109: HTTP State Management Mechanism <a href="http://tools.ietf.org/html/rfc2109">http://tools.ietf.org/html/rfc2109</a></li> <li>• RFC 2965: HTTP State Management Mechanism <a href="http://tools.ietf.org/html/rfc2965">http://tools.ietf.org/html/rfc2965</a></li> </ul>
<p style="font-size: x-small;">GSyC (<a href="http://cursosweb.github.io/">http://cursosweb.github.io/</a>) Servicios y Aplicaciones Telemáticas 2015-16 19 de enero de 2016 27 / 324</p>	<p style="font-size: x-small;">GSyC (<a href="http://cursosweb.github.io/">http://cursosweb.github.io/</a>) Servicios y Aplicaciones Telemáticas 2015-16 19 de enero de 2016 28 / 324</p>



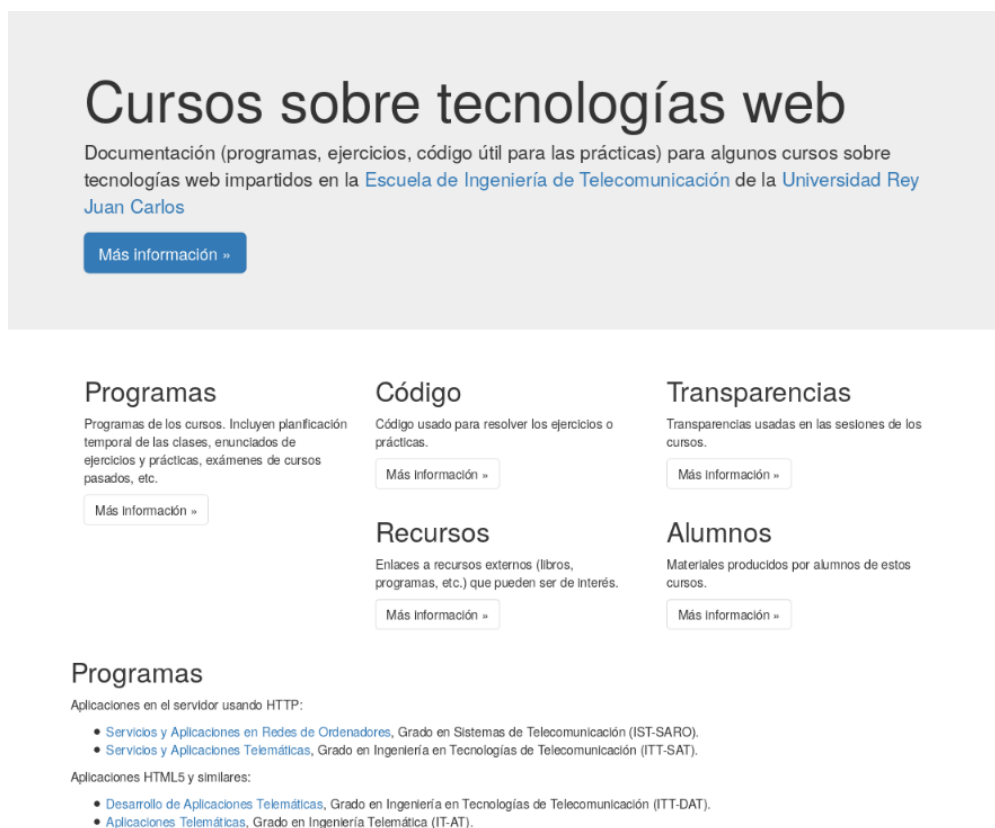
## Capítulo 10

# Sitio web de las asignaturas

En este anexo se muestran algunas capturas de pantalla del sitio web Cursos Sobre Tecnologías Web, que se utiliza para proporcionar como recursos educativos abiertos prácticamente todos los materiales utilizados en las asignaturas que nos ocupan. Las capturas se han realizado durante noviembre de 2017.

El sitio puede accederse en la siguiente dirección:

<http://cursosweb.github.io/>



### Cursos sobre tecnologías web

Documentación (programas, ejercicios, código útil para las prácticas) para algunos cursos sobre tecnologías web impartidos en la [Escuela de Ingeniería de Telecomunicación](#) de la [Universidad Rey Juan Carlos](#)

[Más información »](#)

#### Programas

Programas de los cursos. Incluyen planificación temporal de las clases, enunciados de ejercicios y prácticas, exámenes de cursos pasados, etc.

[Más información »](#)

#### Código

Código usado para resolver los ejercicios o prácticas.

[Más información »](#)

#### Transparencias

Transparencias usadas en las sesiones de los cursos.

[Más información »](#)

#### Recursos

Enlaces a recursos externos (libros, programas, etc.) que pueden ser de interés.

[Más información »](#)

#### Alumnos

Materiales producidos por alumnos de estos cursos.

[Más información »](#)

### Programas

Aplicaciones en el servidor usando HTTP:

- [Servicios y Aplicaciones en Redes de Ordenadores](#), Grado en Sistemas de Telecomunicación (IST-SARO).
- [Servicios y Aplicaciones Telemáticas](#), Grado en Ingeniería en Tecnologías de Telecomunicación (ITT-SAT).

Aplicaciones HTML5 y similares:

- [Desarrollo de Aplicaciones Telemáticas](#), Grado en Ingeniería en Tecnologías de Telecomunicación (ITT-DAT).
- [Aplicaciones Telemáticas](#), Grado en Ingeniería Telemática (IT-AT).

## Código

Todo el código está en el repositorio [CursosWeb/Code](#).

Código utilizado en IST-SARO e ITT-SAT:

- Servidores web simples en Python
- Servidores web que Interoperan (simples, en Python)
- XML y Python
- CSS
- Ajax (demos online)

Código utilizado en ITT-DAT e IT-AT:

- Introducción a HTML
- Introducción a CSS
- Introducción a JavaScript
- Introducción a jQuery (demos online)
- Introducción a HTML5
- Algunas bibliotecas JavaScript (demos online relacionadas con jQuery-UI)
- APIs de servicio usadas desde JavaScript (demos online relacionadas con mapas, demos online relacionadas con GitHub)
- OAuth (demos online)

Prácticas incrementales, entre los repositorios de [CursosWeb](#) en [GitHub](#) (ver detalles en el programa de cada asignatura):

- ITT-DAT e IT-AT: Los repositorios que comienzan por X-Nav.
- IST-SARO e ITT-SAT: Los repositorios que comienzan por X-Serv.

Recuerda que para entregar tus prácticas tendrás que "hacer un fork" de los repositorios correspondientes, y registrar tu nombre de cuenta en [Git Hub](#) en la actividad abierta para ello en el sitio de tu asignatura en el campus virtual.

Atención: iremos poniendo más código relacionado con las prácticas durante los próximos días.

## Transparencias

Aplicaciones en el servidor usando HTTP:

- [Servicios y Aplicaciones en Redes de Ordenadores](#), Grado en Sistemas de Telecomunicación (IST-SARO).
- [Servicios y Aplicaciones Telemáticas](#), Grado en Ingeniería en Tecnologías de Telecomunicación (ITT-SAT).

Aplicaciones HTML5 y similares:

- [Desarrollo de Aplicaciones Telemáticas](#), Grado en Ingeniería en Tecnologías de Telecomunicación (ITT-DAT).
- [Aplicaciones Telemáticas](#), Grado en Ingeniería Telemática (IT-AT).

Otros juegos de transparencias:

- [Frikiminutos \(serie A, para IST-SARO e ITT-SAT\)](#)
- [Frikiminutos \(serie B, para ITT-DAT\)](#)
- [Frikiminutos \(serie C, para IT-AT\)](#)
- [Frikiminutos \(curso 2014-2015\)](#)
- [Introducción práctica a las tecnologías web](#)

Atención: iremos actualizando y completando las transparencias según avanzan las asignaturas.

## Recursos

General:

- Materiales varios: [MDN Web Technologies for Developers](#). Documentación y materiales de muchos tipos para desarrolladores interesados en tecnologías web
- Materiales varios: [Firefox Developer Tools](#). Documentación sobre las herramientas de desarrollador incluidas en Firefox (consola, Inspector de HTML, depuradores de JavaScript, de red, etc.)

JavaScript:

- Transparencias: [Concise JavaScript](#). Breve introducción al lenguaje para quien ya sabe programar.
- Transparencias: [Learning jQuery in 30 minutes, jQuery Essentials](#). Introducciones a jQuery.
- Guía: [A re-introduction to JavaScript](#). Guía práctica para quien cree que ya sabe algo de JavaScript. Muy recomendable.
- Guía: [JavaScript Garden](#). Documentación sobre los detalles más peculiares de JavaScript, los que suelen dar más problemas.
- Libro: [Eloquent JavaScript](#). Puede ayudarte mucho si ya sabes programar y estás empezando a trabajar con JavaScript.
- Materiales varios: [MDN JavaScript](#). Muchos materiales para aprender JavaScript, producidos y compilados por el proyecto Mozilla.

Python:

- Guía: [The Hitchhiker's Guide to Python](#). Manual de buenas prácticas sobre la instalación, configuración y uso de Python.
- Libro: [Dive Into Python](#). Tal y como la propia introducción del libro dice: «un libro sobre Python para programadores».

Django:

- Libro: [Django Girls Tutorial](#). Tutorial sobre Django, para chicas y chicos.

Git:

- Libro: [Pro Git](#). Libro sobre git, que cubre desde el uso básico hasta varios detalles avanzados. Incluyendo el uso con GitHub.

Otros:

- Libro: [Designing for Performance](#). Optimización de aplicaciones web, teniendo en cuenta también aspectos de diseño.

## Alumnos

Materiales producidos por alumnos

- [Propuestas participantes en el concurso de Bootstrap](#), alumnos de ITT-DAT e IT-AT (curso 2016-2017).
- [Propuestas participantes en el concurso de Bootstrap](#), alumnos de ITT-DAT e IT-AT (curso 2015-2016).
- [Propuestas participantes en el concurso de Bootstrap](#), alumnos de ITT-DAT e IT-AT (curso 2014-2015).
- [Juego con Canvas](#), de Víctor Carrera.

# Capítulo 11

## Ejemplos de ejercicios

A continuación, se incluyen algunos ejemplos de ejercicios, que se utilizan en clase según los principios del aprendizaje basado en problemas. En este caso, constituyen un camino incremental en el que el alumno comienza ejercitándose en Python construyendo una calculadora simple, luego reflexiona sobre interfaces REST implementando sumadores y calculadoras como aplicaciones web, y por último realiza una implementación más completa basada en el framework Django. Estos son ejemplos reales, utilizados en la asignatura Servicios y Aplicaciones Telemáticos.

### 11.0.1. Calculadora

**Enunciado:**

Crea un programa que sirva de calculadora y que incluya las funciones básicas (sumar, restar, multiplicar y dividir). El programa ha de poder ejecutarse desde la línea de comandos de la siguiente manera: `python calculadora.py función operando1 operando2`. No olvides capturar las excepciones.

Parte del repositorio en GitHub <https://github.com/CursosWeb/X-Serv-13.6-Calculadora>

### 11.0.2. Sumador simple versión REST

**Enunciado:**

Desarrollar una versión RESTful de “Sumador simple”. ¿Plantea problemas si se usa simultáneamente desde varios navegadores? ¿Plantea problemas si se cae el servidor entre dos invocaciones por parte del mismo navegador?

**Comentarios:**

Hay varias formas de hacer el diseño, pero por ejemplo, cada sumando podría ser un recurso, y el resultado obtenerse en un tercero (o bien como respuesta al actualizar el segundo sumando). Cada suma podría también realizarse en un espacio de nombres de recurso distinto (con su propio primer sumando, segundo sumando y resultado).

### 11.0.3. Calculadora simple versión REST

**Enunciado:**

Realizar una calculadora de las cuatro operaciones aritméticas básicas (suma, resta, multiplicación y división), siguiendo los principios REST, a la manera del sumador simple versión REST (ejercicio 11.0.2).

**Comentarios:**

Este ejercicio, más que para proponer una solución concreta, está diseñado para debatir sobre las posibles soluciones que se le podrían dar. Por ejemplo, tenemos primero la versiones donde se supone un único usuario:

- Versión con un recurso por tipo de operación (“/suma”, “resta”, etc.). Se actualiza con PUT, que envía los operandos (ej: 4,5), se consulta con GET, que devuelve el resultado (ej: 4+5=9).
- Versión con un único recurso, “/operacion”. Se actualiza con PUT, que envía en el cuerpo la operación (ej: 4+5), se consulta con GET, que devuelve el resultado (ej: 4+5=7).

- Versión actualizando por separado los elementos de la operación, con un único recurso “/operacion”. PUT podrá llevar en el cuerpo “Primero: 4” o “Segundo: 5”, o “Op: +”. Cada uno de ellos actualiza el elemento correspondiente de la operación. GET de ese recurso, devuelve el resultado de la operación con los elementos que tiene en este momento.
- Versión actualizando por separado los elementos de la operación, con un único recurso “/operación”. PUT podrá llevar en el cuerpo un número si es la primera o segunda vez que se invoca, un símbolo de operación si es la tercera. GET dará el resultado si se han especificado todos los elementos de la operación, error si no. Es “menos REST”, en el sentido que guarda más estado en el lado del servidor. Pero cumple los requisitos generales de REST si consideramos que el cliente es responsable de mantener su estado y saber en qué fase de la operación está en cada momento.
- Versión donde cada elemento se envía con un PUT a un recurso (“/operacion/primeroperando”, “/operacion/segundooperando”, “/operacion/signo”), y el resultado se obtiene con “GET /operacion/resultado”. No es REST, porque el estado del recurso “/operacion/resultado” depende del estado de los otros recursos .

También podemos extender el diseño a versiones con varios usuarios:

- Podría tenerse un identificador para cada operación. “POST /operaciones” podría devolver el enlace a una nueva operación creada, como “/operaciones/2af434ad3”. Cada una de estas sumas se comportaría como las “sumas con un usuario” que se han comentado antes. “DELETE /operaciones/2af434ad3” destruiría una operación.

#### Material:

- `simplecalc.py`: Programa con una posible solución a este ejercicio. Proporciona cuatro recursos “calculadora”, uno para cada operación matemática (suma, resta, multiplicación, división). Cada calculadora mantiene un estado (operación matemática) que se actualiza con PUT y se consulta con GET.
- Vídeo que muestra el funcionamiento de `simplecalc.py`  
<http://vimeo.com/31427714>
- Vídeo que describe el programa `simplecalc.py`  
<http://vimeo.com/31430208>
- `multicalc.py`: Programa con otra posible solución a este ejercicio. Proporciona un recurso para crear calculadoras (mediante POST). Al crear una calculadora se especifica de qué tipo (operación) es. Cada calculadora mantiene un estado (operación matemática) que se actualiza con PUT y se consulta con GET. Se apoya en las clases definidas en `simplecalc.py` para implementar las calculadoras.
- `webappmulti.py`: Clase que proporciona la estructura básica para los dos programas anteriores (clase raíz de servicio web, de *aplis*, etc.)

### 11.0.4. Django calc

#### Enunciado:

Realiza una calculadora con Django. Esta calculadora responderá a URLs de la forma “/num1+num2”, “/num1\*num2”, “/num1-num2”, “/num1/num2”, realizando las operaciones correspondientes, y devolviendo error “Not Found” para las demás.

Parte del repositorio en GitHub: <https://github.com/CursosWeb/X-Serv-15.4-Django-calc>. El proyecto Django se llama `project` y la aplicación `calc`. Recuerda que sólo tendrás que modificar los siguientes ficheros: `settings.py`, `urls.py` y `views.py`.

#### Material:

- `calc.tar.gz`: Ejemplo de solución del ejercicio “Django calc”



# Capítulo 12

## Ejemplos de prácticas

A continuación se incluye el enunciado de algunas prácticas que fueron propuestas a alumnos de las asignaturas objeto de esta memoria:

- Ejemplo de proyecto práctico de preparación (apartado 12.1). Propuesto en la asignatura Aplicaciones Telemáticas el curso 2016-2017.
- Ejemplo de práctica final (apartado 12.2). Propuesto en la asignatura Aplicaciones Telemáticas en mayo de 2017.

### 12.1. Socios

#### **Enunciado:**

Vamos a construir parte del interfaz de usuario de la aplicación Socios, una nueva red social. En particular, vamos a representar en el navegador la información que nos va a llegar en varios documentos JSON. Para simular lo suficiente para poder realizar la interfaz de usuario, estos documentos JSON serán ficheros estáticos que se servirán al navegador con el resto de la aplicación, que estará compuesta por un fichero HTML, otro JavaScript y otro CSS.

Los documentos JSON mencionados son los siguientes:

- `timeline.json`: Mensajes de los socios del usuario, en modo resumen (ver detalle más abajo).
- `update.json`: Mensajes de los socios que aún no se ha mostrado en el timeline.
- `myline.json`: Mensajes del usuario, puestos en el pasado.

Para cada mensaje, los documentos JSON tendrán al menos la siguiente información:

- **Autor**: nombre del autor del mensaje.
- **Avatar**: url del avatar (imagen) del autor del mensaje.
- **Título**: título del mensaje.
- **Contenido**: contenido del mensaje.
- **Fecha**: fecha en que fue escrito el mensaje.

Opcionalmente, para cada mensaje se podrá ofrecer otra información adicional, como coordenadas de geolocalización, urls de anexos (attachements), etc.

Además de estos documentos JSON con la información de mensajes, se servirán vía HTTP las imágenes (avatares) que se citen en ellos, y los tres documentos básicos de la aplicación: uno HTML, otro CSS y otro JavaScript.

La aplicación mostrará en pestañas (tabs) diferentes la siguiente información:

- Timeline del usuario: mensajes de sus socios, según listado en `timeline.json`. Además, una vez mostrados estos mensajes, se buscará `update.json`. Si tiene alguna noticia, se mostrará una nota al principio del timeline indicando el número de mensajes pendientes. Cuando se pulse en esa nota, se desplegarán los mensajes pendientes que estaban en `update.json`.
- Mensajes enviados por el usuario, según listado en `myline.json`

En principio, de cada mensaje se mostrará sólo el nombre del autor, su avatar, y el título del mensaje. Se ofrecerá un botón para desplegar todo el mensaje: si se pulsa, se desplegará el resto de la información. Se podrán realizar otras mejoras a este comportamiento básico.

#### Entrega:

La práctica se entregará según se describe en el apartado 12.3, utilizando la rama `gh-pages` para que sea visible directamente como sitio web (ver detalles en dicho apartado).

En ambas ramas (`master` y `gh-pages`) del repositorio de entrega habrá al menos:

- Un fichero `README.md` que resuma las mejoras, si las hay, y explique cualquier peculiaridad de la entrega.
- Los tres ficheros mencionados (HTML, CSS, JavaScript).
- Los ficheros JSON especificados.
- Los ficheros de avatar (imágenes) necesarios.
- Cualquier biblioteca JavaScript que pueda hacer falta (normalmente, sólo jQuery y jQueryUI) para que la aplicación funcione.

El fichero HTML se llamará “`index.html`”, y todo el directorio (repositorio) estará construido de tal forma que bastará con servirlo mediante un servidor HTTP, y cargar en un navegador este fichero HTML, para que la vista de nuestros socios (y todo el interfaz de usuario) funcione. Igualmente, deberá funcionar si se carga el repositorio desde GitHub (que mostrará lo que haya en la rama `gh-pages`).

## 12.2. Aparcamientos para residentes en Madrid (mayo 2017, borrador)

La práctica consiste en la creación de una aplicación HTML5 que permita explorar los lugares en el municipio de Madrid que cuentan con aparcamientos para residentes, a partir de ahora *instalaciones*, construida a partir de los datos proporcionados por el propio ayuntamiento. Se trata de mostrar los datos de la forma más atractiva posible, incluyendo el nombre del aparcamiento (campo `title`), la dirección (calle, código postal), la descripción (campo `organization-desc`), fotos de la zona, localización en un mapa, etc. También se permitirá construir colecciones personales de aparcamientos, y almacenarlas de forma persistente.

### 12.2.1. Enunciado

Concretamente, la aplicación mostrará al arrancar un *banner* (imagen) con el nombre y/o logotipo del sitio y un panel con tres pestañas: la principal, la de gestión de colecciones y la de gestión de instalaciones.

La pestaña principal contará con las siguientes *zonas*:

1. Una zona con un mapa, donde se mostrará la localización de las instalaciones seleccionadas. Cuando se seleccione una instalación, se mostrará un marcador sobre el mapa con su nombre. Cada marcador tendrá una opción para quitarlo.
2. Una zona donde se mostrará una lista con todas las instalaciones. Cuando arranque la aplicación, no habrá ninguna sino que en su lugar habrá un botón para cargar el fichero JSON con todas las instalaciones (ver más adelante). Como sin instalaciones la aplicación no puede hacer gran cosa, hasta que la lista no haya sido cargada, las demás zonas de información de las instalaciones estarán desactivadas. Una vez cargado el fichero JSON, como son muchas instalaciones, se mostrarán en una tabla con scroll (barra de desplazamiento) o similar.

3. Una zona donde se mostrará la información sobre la instalación seleccionada (al menos su nombre, su dirección, y un carrusel con sus fotos, si las hay).
4. Una zona donde se mostrará las instalaciones de la colección seleccionada en la pestaña de colecciones (ver más abajo).

Las instalaciones podrán seleccionarse bien picando sobre ellas en la lista de instalaciones, en la lista de instalaciones de una colección, o sobre su marcador en el mapa.

En la pestaña de gestión de colecciones se mostrará la siguiente información:

1. Un listado de todas las instalaciones, una vez han sido cargadas (ver pestaña principal), de forma similar a como se ven en la pestaña principal.
2. Un listado de las colecciones que se hayan creado, y un formulario para crear una nueva colección indicando su nombre. Sobre el listado se podrá seleccionar una colección picando sobre ella.
3. Un listado de las instalaciones en la colección seleccionada. Cuando haya una colección seleccionada y se visite la pestaña principal, se mostrarán las instalaciones de esa colección en la zona creada para tal efecto.

Para añadir instalaciones a una colección, se podrá “arrastrar” una instalación desde el listado de instalaciones al listado de la colección seleccionada.

En la pestaña de gestión de instalaciones, se mostrará:

1. El nombre, la dirección, la descripción de la instalación seleccionada - la instalación que se muestra es la que se ha seleccionado en la pestaña principal.
2. La lista de los usuarios de Google+ asignados a esa instalación. Para añadir usuarios a una instalación, se podrá “arrastrar” un usuario desde el listado de usuarios al listado de la instalación seleccionada.
3. Una lista con usuarios de Google+ disponibles. Al principio se mostrará un botón para incluir usuarios de Google+. Al presionar el botón se abrirá una conexión con un servidor WebSocket que irá enviando identificadores de usuarios de Google+, que serán incluidos en esta zona según se vayan recibiendo. No deben aparecer usuarios repetidos.
4. El código del servidor WebSocket se podrá descargar de <https://github.com/CursosWeb/PFinalWebSocketServer> y se ejecutarán de la siguiente manera desde la línea de comandos:

```
$ python PFinalWebSocketServer.py --port 12345
```

donde el puerto podría ser cualquier otro. De hecho, depende de vuestra configuración, puede que tengáis que poner como puerto el 80, por lo que debéis lanzar el servidor con *sudo*.

En todas las pestañas, además, se verán dos botones, con la siguiente funcionalidad:

- Un botón para guardar las colecciones y las asignaciones de usuarios de Google+ a instalaciones. Cuando se pulse, aparecerá un formulario para rellenar un token de GitHub y el nombre de un repositorio y un fichero, y se almacenará en él toda la información, como un documento JSON.
- Un botón para cargar la información desde un fichero en GitHub, que cargará el contenido de un fichero JSON con colecciones y asignaciones de usuarios de Google+ a hoteles, y las mostrará en esta pestaña.

Para la maquetación de la aplicación se utilizará Bootstrap, haciendo lo posible para que la aplicación sea usable tanto en un navegador de escritorio (con una ventana utilizable grande) como en un móvil (con una pantalla utilizable pequeña, en la que no se podrán ver todos los elementos de la aplicación a la vez). Se utilizará, en la medida de lo razonable, CSS3 para todo lo relacionado con el aspecto de la aplicación.

Se usará Leaflet para mostrar los mapas. Se podrán utilizar otras bibliotecas JavaScript en lo que pueda ser conveniente.

El fichero JSON con la lista de instalaciones que se usará será el que proporciona el Ayuntamiento de Madrid. Puede verse este fichero y una aplicación de prueba de concepto muy simple de algunos aspectos de la práctica en:

- Fichero con los datos abiertos de aparcamientos para residentes proporcionado por el Ayuntamiento de Madrid:  
<http://datos.madrid.es/egob/catalogo/202584-0-aparcamientos-residentes.json>
- Descripción del fichero:  
<http://datos.madrid.es/portal/site/egob/menuitem.c05c1f754a33a9fbe4b2e4b284f1a5a0/?vgnnextoid=e84276ac109d3410VgnVCM200000c205a0aRCRD&vgnnextchannel=374512b9ace9f310VgnVCM100000171f5a0aRCRD&vgnnextfmt=default>
- Copia del fichero anterior en el repositorio CursosWeb/Code de GitHub:

Para obtener las fotos cercanas a una localización (latitud y longitud), se podrán utilizar las que proporciona Wikimedia Commons, utilizando urls como la siguiente:

- Formato JSON:  
<https://commons.wikimedia.org/w/api.php?format=json&action=query&generator=geosearch&ggsprimary=all&ggsnamespace=6&ggsradius=500&ggscoord=40.426687899755734|-3.6996173701222492&ggslimit=10&prop=imageinfo&iilimit=1&iiprop=url&iiurlwidth=200&iiurlheight=200>
- Formato JSONP:  
<https://commons.wikimedia.org/w/api.php?format=json&action=query&generator=geosearch&ggsprimary=all&ggsnamespace=6&ggsradius=500&ggscoord=40.426687899755734|-3.6996173701222492&ggslimit=10&prop=imageinfo&iilimit=1&iiprop=url&iiurlwidth=200&iiurlheight=200&callback=?>
- Explicación en StackOverflow:  
<http://stackoverflow.com/questions/23990161/wikimedia-commons-api-search-images-by-latitude-longitud>

### 12.2.2. Funcionalidad optativa

En general, se podrá añadir cualquier funcionalidad a la aplicación, mientras no perjudique o impida la funcionalidad básica descrita en el apartado anterior. A modo de ejemplos, se proponen las siguientes ideas:

- Utilización de otras APIs
- Uso de LocalStorage
- Funcionalidad para que funcione off-line
- Realizar la autenticación en GitHub dinámicamente, usando OAuth2 de forma que el usuario se autentique con su cuenta de GitHub, y obtenga el token de esa forma.
- Optimizar la aplicación para que cargue más rápido.
- Utilización de Flickr y otros servicios como fuente de fotos.

Todas las opciones que necesitan escribir en repositorios GitHub, requerirían autenticación previa por parte del usuario.

### 12.2.3. Entrega de la práctica

- **Fecha límite de entrega de la práctica:** miércoles, 24 de mayo de 2017 a las 03:00 (hora española peninsular)<sup>1</sup>
- **Fecha de publicación de notas:** sábado, 27 de mayo de 2017, en la plataforma Moodle.
- **Fecha de revisión:** lunes, 29 de mayo de 2017 a las 13:30.

La entrega de la práctica consiste en rellenar un formulario (enlazado en el Moodle de la asignatura) y en seguir las instrucciones que se describen a continuación.

1. El repositorio contendrá todos los ficheros necesarios para que funcione la aplicación (ver detalle más abajo), con la rama *master* configurada de tal manera que la aplicación se pueda probar accediendo al servidor web que monta GitHub. En el formulario de entrega se pedirá la URL del repositorio GitHub utilizado por el alumno para la práctica.

Los alumnos que entreguen la práctica podrán ser llamados a realizar una entrega presencial, que tendrá lugar en la fecha y hora de la revisión. Esta entrega presencial podrá incluir una conversación con el profesor sobre cualquier aspecto de la realización de la práctica.

2. Un vídeo de demostración de la parte obligatoria, y otro vídeo de demostración de la parte opcional, si se han realizado opciones avanzadas. Los vídeos serán de una duración máxima de 3 minutos (cada uno), y consistirán en una captura de pantalla de un navegador web utilizando la aplicación, y mostrando lo mejor posible la funcionalidad correspondiente (básica u opcional). Siempre que sea posible, el alumno comentará en el audio del vídeo lo que vaya ocurriendo en la captura. Los vídeos se colocarán en algún servicio de subida de vídeos en Internet (por ejemplo, Vimeo o YouTube).

Hay muchas herramientas que permiten realizar la captura de pantalla. Por ejemplo, en GNU/Linux puede usarse *Gtk-RecordMyDesktop* o *Istanbul* (ambas disponibles en Ubuntu). Es importante que la captura sea realizada de forma que se distinga razonablemente lo que se grabe en el vídeo.

En caso de que convenga editar el vídeo resultante (por ejemplo, para eliminar tiempos de espera) puede usarse un editor de vídeo, pero siempre deberá ser indicado que se ha hecho tal cosa con un comentario en el audio, o un texto en el vídeo. Hay muchas herramientas que permiten realizar esta edición. Por ejemplo, en GNU/Linux puede usarse *OpenShot* o *PiTiVi*.

3. Se han de entregar los siguientes ficheros:
  - Un fichero *README.md* que resuma las mejoras, si las hay, y explique cualquier peculiaridad de la entrega (ver siguiente punto).
  - Los ficheros de la práctica (HTML, CSS, JavaScript). El fichero HTML principal se llamará "index.html", y estará construido de forma que bastará con cargarlo en el navegador para que funcione el programa.
  - Cualquier biblioteca JavaScript que pueda hacer falta para que la aplicación funcione, junto con los ficheros auxiliares que utilice, si es que los utiliza.
4. Se incluirán en el fichero *README.md* los siguientes datos (la mayoría de estos datos se piden también en el formulario que se ha de rellenar para entregar la práctica - se recomienda hacer un corta y pega de estos datos en el formulario):

- Nombre y titulación.
- Nombre de su cuenta en el laboratorio del alumno.
- Nombre de usuario en GitHub.
- Resumen de las peculiaridades que se quieran mencionar sobre lo implementado en la parte obligatoria.
- Lista de funcionalidades opcionales que se hayan implementado, y breve descripción de cada una.

---

<sup>1</sup>Entiéndase la hora como miércoles por la noche, ya entrado el jueves.

- URL del vídeo demostración de la funcionalidad básica
- URL del vídeo demostración de la funcionalidad optativa, si se ha realizado funcionalidad optativa

Asegúrate de que las URLs incluidas en este fichero están adecuadamente escritas en Markdown, de forma que la versión HTML que genera GitHub los incluya como enlaces “pinchables”.

### 12.3. Entrega de prácticas

Para la entrega de prácticas se utilizarán repositorios git públicos alojados en GitHub. Para cada práctica entregable los profesores abrirán un repositorio público en el proyecto CursosWeb <sup>2</sup>, con un nombre que comenzará por “X-Nav-”, seguirá con el nombre del tema en el que se inscribe la práctica (por ejemplo, “JS” para el tema de introducción a JavaScript) y el identificador del ejercicio (por ejemplo, “Sumador”). Este repositorio incluirá un fichero README.md, con el enunciado de la práctica, y cualquier otro material que los profesores estimen conveniente.

Cada alumno dispondrá de una cuenta en GitHub, que usará a efectos de entrega de prácticas. Esta cuenta deberá ser apuntada en una lista, en el sitio de la asignatura en el campus virtual, cuando los profesores se lo soliciten. Si el alumno desea que no sea fácil trazar su identidad a partir de esta cuenta, puede elegir abrir una cuenta no ligada a sus datos personales: a efectos de valoración, los profesores utilizará la lista anterior. Si el alumno lo desea, puede usar la misma cuenta en GitHub para otros fines, además de para la entrega de prácticas.

Para trabajar en una práctica, los alumnos comenzarán por realizar una copia (fork) de cada uno de estos repositorios. Esto se realiza en GitHub, visitando (tras haberse autenticado con su cuenta de usuario de GitHub para entrega de prácticas) el repositorio con la práctica, y pulsando sobre la opción de realizar un fork. Una vez esto se haya hecho, el alumno tendrá un fork del repositorio en su cuenta, con los mismos contenidos que el repositorio original de la práctica. Visitando este nuevo repositorio, el alumno podrá conocer la url para clonarlo, con lo que podrá realizar su clon (copia) local, usando la orden `git clone`.

A partir de este momento, el alumno creará los ficheros que necesite en su copia local, los irá marcando como cambios con `git commit` (usando previamente `git add`, si es preciso, para añadirlos a los ficheros considerados por git), y cuando lo estime conveniente, los subirá a su repositorio en GitHub usando `git push`.

Por lo tanto, el flujo normal de trabajo de un alumno con una nueva práctica será:

[En GitHub: visita el repositorio de la práctica en CursosWeb,  
y le hace un fork, creando su propia copia del repositorio]

```
git clone url_copia_propia
```

[Se crea el directorio copia\_propia, copia local del repositorio propio]

```
cd copia_propia
git add ... [ficheros de la práctica]
git commit .
git push
```

Conviene visitar el repositorio propio en GitHub, para comprobar que efectivamente los cambios realizados en la copia local se han propagado adecuadamente a él, tras haber invocado `git push`.

#### 12.3.1. Uso de la rama gh-pages

La rama gh-pages en cualquier repositorio de GitHub tiene un significado especial: GitHub entenderá que ha de servir, como sitio web (esto es, como recursos HTTP) cualquier contenido que tenga esa rama. Por ejemplo, el contenido de la rama gh-pages del repositorio prueba del usuario jgbarah se podrá consultar con la siguiente url:

---

<sup>2</sup><https://github.com/CursosWeb>

`http://jgbarah.github.io/prueba`

Es importante entender que GitHub no sirve directorios. Si en el repositorio prueba anterior no hay un fichero `index.html`, GitHub indicará que no se puede servir. Si lo hay, se mostrará su contenido al pedir el recurso `/`. En otras palabras, GitHub servirá los recursos que no terminen en `/`. En caso de que se le pida uno que termine en `/` (esto es, normalmente, los que corresponden con directorios), buscará un fichero `index.html` en él, y si lo hay lo servirá. Si no lo hay, mostrará una página de error. Por este motivo, normalmente para las prácticas entregadas, el fichero HTML principal se deberá denominar `index.html`.

Para crear la rama `gh-pages`, se pueden ejecutar las siguientes órdenes en la consola (se supone que ya se está en un repositorio de entrega de prácticas, en el que se está trabajando en la rama `master`):

```
[Primero, nos aseguramos de que estamos en la rama master,
 que apareciera con asterisco]
> git branch
* master
[Creamos la rama gh-pages, que tendrá lo mismo que hay
 ahora en master]
> git checkout -b gh-pages
[Subimos la rama a GitHub]
> git push origin gh-pages
```

A partir de este momento, deberíamos poder ver la nueva rama en GitHub, y las páginas HTML que haya en la rama `gh-pages` deberían poder verse directamente vía la url indicada más arriba.

Si a partir de este momento se quiere trabajar en `master`, primero habrá que cambiar a la rama `master` de nuevo:

```
[Volvemos a la rama master]
> git checkout master
[Ahora seguimos trabajando en la rama master]
```

Cuando queramos actualizar la rama `gh-pages` con los cambios que hemos hecho en `master`, tendremos que usar “rebase” o “merge”. Usando “merge”:

```
[Cambiamos a la rama gh-pages]
> git checkout gh-pages
[Pasamos los cambios de master a la rama actual (gh-pages)]
> git merge master
[Subimos los cambios a GitHub]
> git push origin gh-pages
[Volvemos a la rama master para seguir trabajando]
> git checkout master
```

Usando “rebase”:

```
[Cambiamos a la rama gh-pages]
> git checkout gh-pages
[Pasamos los cambios de master a la rama actual (gh-pages)]
> git rebase master
[Subimos los cambios a GitHub]
> git push origin gh-pages
[Volvemos a la rama master para seguir trabajando]
> git checkout master
```

Cuando hayas realizado la entrega de las prácticas en la rama `gh-pages` (y por favor, también en la rama `master`), no olvides comprobar en GitHub que los cambios aparecen en la rama `gh-pages`, y en el sitio web correspondiente a tu repositorio.

### 12.3.2. Uso de la rama gh-pages con dos clones

Una forma de organizarse cuando se use la rama gh-pages es usando dos clones del repositorio de entrega: uno para la rama master, y otro para la rama gh-pages. Si se decide utilizar esta forma de funcionamiento, se comenará normalmente, clonando el fork del repositorio de entrega en GitHub. Se trabajará en ese clon normalmente, en la rama master. Cuando se quiera tener una rama gh-pages, se hará lo siguiente:

```
[En el clon normal, trabajando sobre master, se envían los cambios
 a GitHub, después de haber hecho los commits convenientes.]
> git push
[Se crea un nuevo clon del mismo repositorio, en el directorio
 practica-gh-pages]
> cd ..
> git clone url_repo practica-gh-pages
[Se crea en este nuevo clon la rama gh-pages]
> cd practica-gh-pages
> git checkout -b gh-pages
[Se sube ahora la nueva rama a GitHub]
> git push origin gh-pages
```

A partir de este momento, en el repositorio en GitHub están las dos ramas (master y gh-pages), ambas con el mismo contenido. Localmente, habrán quedado dos directorios, uno con la rama master, y otro con la rama gh-pages.

Si se quiere seguir trabajando en la rama master, se hará normalmente en el directorio local correspondiente, subiendo los cambios a GitHub (con push) cuando sea conveniente. Cuando se quiera sincronizar en la rama gh-pages con el estado actual de master, se podrá hacer sobre el directorio con la rama gh-pages:

```
> cd practica-gh-pages
> git fetch
> git rebase master
> git push origin
```

Alternativamente a rebase, se puede realizar un merge, como se indicó más arriba.

### 12.3.3. Más información sobre GitHub Pages

Hay más información sobre GitHub Pages y el uso de la rama gh-pages en:

- GitHub Pages:  
<http://pages.github.com/>
- User, organization and project pages (el uso que estamos haciendo es el de “project pages”):  
<http://help.github.com/articles/user-organization-and-project-pages/>



# Capítulo 13

## Ejemplo de examen

Este examen, que se ofrece a modo de ejemplo, fue realizado para la asignatura Servicios y Aplicaciones Telemáticas, del GITT, el 10 de mayo de 2017. Se incluye el enunciado del examen, y la solución que se publicó posteriormente.

### 13.1. Enunciado

Se quiere construir un sitio web, Mensajitos, donde se pueden poner mensajes para que los vean otras personas. La funcionalidad básica del sitio es la siguiente:

1. En el sitio no hay cuentas para usuarios: toda la funcionalidad está disponible para cualquiera que lo visite.
2. De todas formas, cualquier visitante podrá reservar un nombre que no esté ya en uso para cuando suba información al sitio. Este nombre se mantendrá mientras el visitante utilice el mismo navegador. Para ello se usará el formulario que aparece en la página principal (ver a continuación).
3. La página principal del sitio mostrará a los visitantes un botón para crear un canal de mensajes, y un formulario para elegir un nombre (si se ha elegido ya, en lugar del formulario aparecerá el nombre elegido). El botón permitirá crear un nuevo canal (cada visitante puede crear tantos como quiera), según se indica más abajo. Además, en esta página principal cada visitante verá la lista de los canales que ha creado previamente. Tras crear un nuevo canal, o elegir un nombre, el visitante volverá a ver la página principal del sitio.
4. Cada canal tendrá un nombre de recurso único, que se generará aleatoriamente cuando se cree. Cualquiera que conozca el nombre de recurso de un canal, podrá leer y escribir en él, simplemente accediendo a ese recurso (lo haya creado quien lo haya creado).
5. El recurso correspondiente a cada canal mostrará una página HTML (la “página del canal”) con los 10 últimos mensajes en el canal, un formulario para poner un nuevo mensaje, y un formulario para poner la url de una imagen (que puede estar en cualquier sitio de Internet, mientras la haga visible mediante HTTP). Cada mensaje que se muestre, se mostrará con el formato:

Nombre: mensaje

Donde “Nombre” es el nombre del visitante (o “Anónimo”, si no lo ha elegido), y “mensaje” es el mensaje en cuestión.

Las urls de imágenes se considerarán también como mensajes, pero antes de mostrarlos como tales (y de almacenarlos en la base de datos), se convertirán a un elemento IMG de HTML. Por ejemplo, la imagen de url `http://fotos.com/123345.jpg` se convertirá en el HTML siguiente (que se considerará el “mensaje” en el formato descrito anteriormente):

```

```

Tras poner un nuevo mensaje (o una imagen) en un canal, el visitante vuelve a ver de nuevo la página del canal.

6. Cada canal tendrá también un recurso asociado donde se podrán descargar todos sus mensajes (incluyendo aquellos que se especificaron como imágenes) en formato XML. Este recurso aparecerá también como enlace en la página del canal. El documento XML correspondiente incluirá al menos todos los mensajes que se han escrito en el canal, el nombre del visitante que puso cada uno de ellos, la fecha en que se puso cada uno de ellos, el enlace a la página HTML del canal, la fecha en que se creó el canal, y el nombre del visitante que creó el canal (si alguno de los visitantes implicados no ha especificado un nombre, se usará “Anónimo”).
7. Todas las páginas HTML del sitio incluirán una imagen de cabecera (banner) que se alojará en el propio sitio.

Teniendo en cuenta los requisitos anteriores, se pide:

1. Diseña un esquema REST para proporcionar el servicio descrito. Se habrán de especificar los nombres de recurso empleados, y cómo reaccionará la aplicación cuando reciba los métodos POST o GET sobre esas urls (no se usarán los métodos PUT o DELETE). Coloca la información en una tabla, con las urls en una columna, los métodos en otra, y la descripción de lo que realizará la aplicación al recibirlos en la tercera. Escribe también un fichero similar al fichero urls.py de Django (aunque no es importante que se respete la sintaxis mientras se entienda y la estructura sea similar a la de Django), que refleje el esquema REST anterior (1 punto).
2. Describe el modelo de datos que necesitará esta aplicación. Define las tablas necesarias y los campos necesarios para la funcionalidad descrita. Hazlo de forma lo más similar posible a lo que tendrías que escribir en el fichero models.py en Django (aunque no es importante que se respete la sintaxis mientras se entienda el modelo de datos que propones) (1 punto).
3. Describe las interacciones HTTP que ocurrirán entre el navegador y cualquier servidor web en el siguiente escenario. El escenario comienza cuando un visitante que accede por primera vez al sitio pone en el navegador la url de la página principal del sitio. A continuación, después de ver esta página principal, rellena el formulario para elegir un nombre. En este momento, el navegador vuelve a mostrar la página principal, ya con el nombre elegido en lugar del formulario para elegirlo. A continuación el visitante crea un nuevo canal. El escenario termina cuando el visitante vuelve a ver la página principal del sitio, con el nuevo canal ya creado. (1 punto).
4. Describe las interacciones HTTP que ocurrirán entre el navegador y cualquier servidor web en el siguiente escenario. El escenario comienza con un visitante que ya ha reservado nombre y está viendo la página de un canal que aún no tiene mensajes. El visitante rellena el formulario de imagen, poniendo la url de una imagen válida. El escenario termina cuando el visitante vuelve a ver la página del canal, ya con el nuevo mensaje generado a partir de la url de la imagen (1 punto).
5. Escribe cómo podría ser el documento XML para un canal que tiene tres mensajes, uno de los cuales corresponde a la url de una imagen, para un usuario que tiene nombre (1 punto).

En todos los escenarios, ten en cuenta que tu respuesta debe considerar toda la funcionalidad que ofrece el servicio, y permitir que ésta pueda proporcionarse. Diseña la aplicación de forma que envíe cookies al navegador sólo cuando sea necesario.

En las respuestas donde describas interacciones HTTP indica para cada una de ellas claramente y en este orden:

- La primera línea de la petición HTTP
- Si lo hay, el contenido de la petición
- La primera línea de la respuesta HTTP
- Si lo hay, el contenido de la respuesta
- Una brevísima explicación de para qué se usa la interacción

- Tanto en la petición como en la respuesta, las cabeceras con cookies, si es que fueran necesarias para la funcionalidad del escenario que se está describiendo (incluyendo el aspecto que han de tener esas cabeceras). Si la cabecera con cookie va o no dependiendo de algún factor ajeno a tu aplicación, explica cuando irá y cuándo no, y cuál es ese factor.

Además, asegúrate de que describes las interacciones HTTP en el orden en que ocurrirían en el escenario.

## 13.2. Soluciones

Hay muchas soluciones posibles. A continuación, una de ellas.

### Esquema REST

Recurso	Método	Comentario
/	GET	Página principal (HTML)
/	POST	Creación de canal <code>canal=True</code> Reserva de nombre <code>nombre=nombre_visitante</code> Devolverá el mismo HTML que si se invoca con GET Este HTML incluirá ya un enlace al nuevo canal
/ {id_canal}	GET	Página del canal <code>id_canal</code> (HTML)
/ {id_canal}	POST	Subir mensaje al canal <code>id_canal</code> <code>mensaje=texto</code> Subir imagen al canal <code>id_canal</code> <code>imagen=url</code> Devolverá el mismo HTML que si se invoca con GET
/ {id_canal}.xml	GET	Página del canal <code>id_canal</code> (XML)
/banner	GET	Imagen que se usará como banner del sitio

Nota: No se indica en el enunciado, pero convendrá que la página HTML que se reciba como respuesta a un POST para crear un canal incluya, de forma prominente, un enlace a dicho canal, dado que el usuario necesita saber cuál es.

### urls.py

```
from django.conf.urls import url
from . import views

urlpatterns = [
    url(r'^$', views.pagina_principal),
    url(r'^banner$', views.banner),
    url(r'^(.+\.xml)$', views.canal_xml),
    url(r'^(.+)$', views.canal)
]
```

### models.py

Versión simplificada, que cumple el enunciado:

```
from django.db import models

class Visitante(models.Model):
    nombre = models.CharField(max_length=20, null=True)
    cookie = models.CharField(max_length=64)
```

```

class Canal(models.Model):
    recurso = models.CharField(max_length=50)
    creador = models.ForeignKey('Visitante')

class Mensaje(models.Model):
    canal = models.ForeignKey('Canal')
    autor = models.ForeignKey('Visitante')
    texto = models.TextField()
    fecha = models.DateTimeField() # Para fecha en XML

```

### Primer escenario

Todas las interacciones son entre el navegador y el sitio Mensajitos.

- Petición GET /

```

GET / HTTP/1.1
...

```

- Respuesta

```

HTTP/1.1 200 OK
Set-Cookie: ....; session=session_id

```

[Pagina principal, HTML]

session\_id es un identificador de sesión (o de visitante), que se puede enviar también más adelante. Como identificador de sesión que es, será normalmente una cadena de caracteres larga, generada aleatoriamente, y por tanto difícil de adivinar para quien no la conozca.

- Petición GET /banner (para cargar la imagen del banner)

```

GET /banner HTTP/1.1
...
Cookie: session=session_id

```

- Respuesta

```

HTTP/1.1 200 OK
...

```

[Banner]

- Petición POST / (para enviar los datos del formulario de nombre)

```

POST / HTTP/1.1
...
Cookie: session=session_id

```

```

nombre=Nombre_Usado

```

- Respuesta

```

HTTP/1.1 200 OK
...

```

[Pagina principal, ya sin formulario para elegir nombre, HTML]

La cookie que se envió anteriormente, en realidad se podría enviar aquí, pues hasta este momento no hay nada que asociar a la sesión.

- Petición POST / (para enviar los datos del botón de crear canal)

```
POST / HTTP/1.1
...
Cookie: session=session_id

canal=True
```

- Respuesta

```
HTTP/1.1 200 OK
...
```

[Pagina principal, ahora con el nuevo canal, HTML]

### Segundo escenario

A continuación, las interacciones son entre el navegador y el sitio Mensajitos. Suponemos que la imagen del banner ya está en la caché del navegador, y por tanto no se pide. Como el navegador ya ha estado visitando el sitio y tiene nombre, ha de haber recibido la cookie de sesión. Suponemos que “/2732434232” es el nombre de recurso correspondiente al canal.

- Petición POST /2732434232 (para poner el mensaje)

```
POST / HTTP/1.1
...
Cookie: session=session_id

imagen="url"
```

- Respuesta

```
HTTP/1.1 200 OK
...
```

[Pagina del canal, ahora con un nuevo mensaje con el img correspondiente, HTML]

```

```

Ahora, el navegador tendrá que pedir la imagen que se haya incluido anteriormente (url “url”). Esta interacción será por lo tanto entre el navegador y el sitio al que apunte la url de la imagen. Suponiendo que la url sear `http://sitio.com/imagen`:

- Petición GET /imagen (para cargar la imagen)

```
GET /imagen HTTP/1.1
...
```

- Respuesta

```
HTTP/1.1 200 OK
...
```

[Imagen]

**Canal XML**

```
<?xml version="1.0" encoding="UTF-8" ?>

<canal>
  <recurso>http://mensajitos.com/3443344453</recurso>
  <creado>20 de marzo de 2016 23:05:05</creado>
  <creador>Flor de Loto</creador>
  <mensaje>
    <texto>Este es un mensaje</texto>
  </mensaje>
  <texto>
    
  </texto>
  <mensaje>
  </mensaje>
  <mensaje>
    <texto>Este es el último mensaje</texto>
  </mensaje>
</canal>
```

El texto para el mensaje de la imagen tendría que ponerse “codificado” para que no se confunda con texto XML, pero esto no se ha tenido en cuenta en esta solución..

## Capítulo 14

# Ejemplo de encuesta a los alumnos

Esta encuesta es un ejemplo de las que son ofrecidas a los alumnos de las asignaturas que nos ocupan en esta memoria. En la actualidad, las encuestas suelen realizarse en un campus virtual, que dispone de funcionalidad para ello.

1. Impresión general de la asignatura.
2. Competencia de los profesores: dominio de la materia, interés mostrado.
3. Claridad de las exposiciones.
4. Fomento por parte del profesor de la participación del alumno en clase con preguntas, comentarios, etc.
5. Interés de los temas tratados. Citar el más y el menos interesante.
6. Utilidad-calidad de los medios empleados (transparencias, etc).
7. ¿Hay algún tema que hubieras suprimido?
8. ¿Qué temas crees que deberían haberse incluido y no lo están?
9. Relación de las prácticas con la parte teórica.
10. Valora la explicación de los aspectos necesarios para la realización de las prácticas.
11. Valoración del lenguaje de programación utilizado como medio para la realización de prácticas en esta asignatura.
12. Volumen de trabajo que han supuesto las prácticas: número de horas aproximado, esfuerzo en relación a otras prácticas (mayor, menor, igual).
13. Apoyo por parte de los profesores en la realización de las prácticas.
14. Utilidad de la documentación en formato electrónico y del sitio de la asignatura.
15. ¿Te hubiera gustado que se incluyeran otro tipo de prácticas? ¿Cuáles?
16. Indica tu opinión sobre el laboratorio donde se han realizado las clases.
17. Indica si has hecho una parte de las prácticas en tu casa, estimando un porcentaje del total del tiempo que has dedicado a las prácticas.
18. Si has decidido no hacer las prácticas en este cuatrimestre, por favor, indica los motivos.
19. Señala el aspecto más negativo que creas que debe solucionarse con urgencia.
20. Otros comentarios sobre la asignatura





# Bibliografía del proyecto docente

- [4] H. S. Barrows and R. M. Tamblyn. *Problem-Based Learning: An Approach to Medical Education*. Springer, 1980.
- [5] Computing Sciences Accreditation Board. Criteria for accrediting programs in computer science in the united states. Technical report, January 1987.
- [6] Educational Activities Board. The 1983 model program in computer science and engineering. Technical Report 932, Computer Society of the IEEE, December 1983.
- [7] Career Space Consortium. *Curriculum Development Guidelines. New ICT Curricula for the 21st Century: Designing Tomorrow's Education*. Office for Official Publications of the European Communities, 2001.
- [8] Ministerio de Ciencia e Innovación. Orden cin/352/2009, de 9 de febrero, por la que se establecen los requisitos para la verificación de los títulos universitarios oficiales que habiliten para el ejercicio de la profesión de ingeniero técnico de telecomunicación. BOE del 20 de febrero de 2009, Febrero 2009.
- [9] Ministerio de Ciencia e Innovación. Orden cin/355/2009, de 9 de febrero, por la que se establecen los requisitos para la verificación de los títulos universitarios oficiales que habiliten para el ejercicio de la profesión de ingeniero de telecomunicación. BOE del 20 de febrero de 2009, Febrero 2009.
- [10] Agencia Nacional de Evaluación de la Calidad y Acreditación. *Libro Blanco: Título de Grado en Ingeniería de Telecomunicación*. ANECA, 2004.
- [11] Agencia Nacional de Evaluación de la Calidad y Acreditación. *Libro Blanco: Título de Grado en Ingeniería de Telecomunicación. Anexo de Estudios de Ingeniería Telemática*. ANECA, 2004.
- [12] Agencia Nacional de Evaluación de la Calidad y Acreditación. *Libro Blanco: Título de Grado en Ingeniería Informática*. ANECA, 2005.
- [13] P.J. Denning, D.E. Comer, D. Gries, M.C. Mulder, A.B. Tucker, and A.J. Turner. Computing as a discipline. *Communications of the ACM*, January 1989.
- [14] Association for Computing Machinery and Association for Information Systems. *IS2010: Curriculum Guidelines for Undergraduate Degree Programs in Information Systems*. Computing Curricula Series. ACM, AIS, 2010.
- [15] Accreditation Board for Engineering and Inc. Technology. Criteria for accrediting programs in engineering in the united states. Technical report, December 1988.
- [16] ACM/IEEE-CS Joint Curriculum Task Force. Computing curricula 1991, December 1990. <http://www.computer.org/education/cc1991>.
- [17] Norman E. Gibbs and Allen B. Tucker. Model curriculum for a liberal arts degree in computer science. *Communications of the ACM*, 29(3):202–210, March 1986.
- [18] Woei Hung, David H. Jonassen, and Rude Liu. *Handbook of Research on Educational Communications and Technology*, chapter Problem Based Learning. Springer, 4th edition, 2014. Chapter in 3rd ed.: [http://www.aect.org/edtech/edition3/er5849x\\_c038\\_fm.pdf](http://www.aect.org/edtech/edition3/er5849x_c038_fm.pdf).

- [19] Elliot P. Koffman, Philip L. Miller, and Caroline E. Wardle. Recommended curriculum for cs1: 1984 a report of the acm curriculum task force for cs1. *Communications of the ACM*, 27(10):998–1001, oct 1984.
- [20] Elliot P. Koffman, David Stemple, and Caroline E. Wardle. Recommended curriculum for cs2, 1984: A report of the acm curriculum task force for cs2. *Communications of the ACM*, 28(8):815–818, August 1985.
- [21] Michael C. Mulder and John Dalphin. Computer science program requirements and accreditation-an interim report of the acm/ieee computer society joint task force. *Communications of the ACM*, 27(4):330–335, April 1984.
- [22] ACM Curriculum Committee on Computer Science. Curriculum 68: Recommendation for academic programs in computer science. *Communications of ACM*, 11(3):151–197, March 1968.
- [23] ACM Curriculum Committee on Computer Science. Curriculum 78: Recommendation for undergraduate program in computer science. *Communications of ACM*, 22(3):147–166, March 1979.
- [24] Joint Task Force on Computing Curricula (IEEE Computer Society and Association for Computing Machinery). *Software Engineering 2014. Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering*. Computing Curricula Series. IEEE CS, ACM, February 2015.
- [25] Amy Roehl, Shweta Linga Reddy, and Gayla Jett Shannon. The flipped classroom: An opportunity to engage millennial students through active learning. *Journal of Family and Consumer Sciences*, 105(2):44, 2013.
- [26] J. E. Strayer. *The effects of the classroom flip on the learning environment: A comparison of learning activity in a traditional classroom and a flip classroom that used an intelligent tutoring system*. PhD thesis, The Ohio State University, 2007.

## Parte III

# Proyecto de investigación



“If you can’t measure it,  
you can’t improve it”

---

Peter Drucker



## Capítulo 15

# Introducción al proyecto de investigación

Este proyecto de investigación ha sido elaborado para concursar a la plaza Y158/DF005206/16-10-2017 del cuerpo de Catedráticos de Universidad del área de conocimiento Ingeniería Telemática de la Universidad Rey Juan Carlos. Esta plaza fue convocada a concurso por resolución de dicha universidad de fecha 17 de octubre de 2017, publicada en el Boletín Oficial del Estado el 30 de octubre de 2017.

Para la elaboración de este proyecto se han usado materiales, en algún caso de forma literal, de la memoria de solicitud de financiación de un proyecto real, realizado íntegramente por el autor de esta memoria. Dado que la lengua habitual para la redacción de este tipo de proyectos, por su ámbito global, es la lengua inglesa, la memoria se ofrece redactada en inglés. Se ofrece también una traducción de su resumen en lengua española.

### 15.1. Resumen en español

Los cambios en el código han sido siempre un factor muy importante en el mantenimiento y la evolución del software. Con la llegada de la entrega y el desarrollo continuo, los cambios al código se han convertido en una parte integral de la operación de los sistemas, y por tanto están sujetos a sus limitaciones. Este hecho, combinado con la gran sensibilidad de muchos sistemas al tiempo de resolución de erratas y a la disponibilidad rápida de nuevas características, ha hecho del tiempo hasta despliegue de los cambios uno de los factores más importantes a optimizar, junto con la calidad del código. Y uno de los factores más influyentes en el tiempo hasta despliegue es el tiempo hasta que el cambio es incorporado en el código.

Por ello, conseguir un tiempo hasta incorporación lo más corto posible, mientras se mantiene un alto estándar de calidad, se ha convertido en clave en muchos proyectos software, en áreas tan diferentes como computación en la nube o los sistemas empotrados. Pero siempre hay un equilibrio: el arreglo rápido de erratas y la implementación rápida de nuevas características pueden resultar en la introducción de nuevas erratas y la producción de un código de baja calidad. El desarrollo moderno usa técnicas como la revisión de código por pares y las pruebas intensivas para mitigar este riesgo. Pero esto da lugar a un nuevo equilibrio: cuanto más esfuerzo se dedica a estas técnicas, menos esfuerzo queda disponible para el desarrollo “real”. Por lo tanto, cualquier medida fiable del proceso ha de tener en cuenta todos estos factores.

Afortunadamente, hoy en día el desarrollo de software está apoyado por herramientas que permiten el seguimiento del proceso completo que lleva a la incorporación de cambios al código fuente. Cuando el cambio pretende corregir una errata, todas las acciones desde que se informa sobre ella hasta la incorporación del cambio que la arregla en el código quedan reflejadas en el sistema de gestión de incidencias, el de revisión de código, el de apoyo a las pruebas, y el de gestión de código fuente. Cuando el cambio implementa una característica, o una refactorización, todas las acciones desde el diseño hasta la aceptación final del cambio están disponibles en los mismos sistemas.

Este proyecto usará estos datos para primero, caracterizar el proceso desde la concepción de un cambio hasta su aceptación en el código, y segundo, estudiar cómo puede mejorarse, y qué factores son clave en conseguir código de calidad a la vez que se reduce todo lo posible el tiempo hasta incorporación. Este

estudio tendrá en cuenta todos los factores mencionados anteriormente, incluyendo el hecho de que muchos cambios causan erratas que necesitan ser corregidas más tarde, lo que lleva a una menor calidad del sistema mientras tanto. En otras palabras, tendremos en cuenta la vida completa de los cambios a través de las varias “encarnaciones” necesarias para corregir erratas o terminar de implementar funcionalidad no completa.

Para satisfacer estos objetivos trabajaremos sobre los resultados del proyecto SobreVision, que estudió específicamente el proceso de revisión y el de sembrado de erratas. Usando una evolución de las metodologías utilizadas en ese proyecto pretendemos calcular el tiempo completo desde que se concibe un cambio hasta que ha sido correctamente incorporado al código, teniendo en cuenta el efecto de las erratas introducidas por los cambios, y otros efectos que den lugar a una implementación incompleta.

El proyecto contará con la colaboración de la empresa de base tecnológica Bitergia, que nació como un spin-off del grupo de investigación que lo propone. Bitergia está muy interesada en los resultados del proyecto, porque ya está comercializando soluciones muy relacionadas con él, y es muy probable que esté interesada en incorporar estos resultados a su oferta comercial.

## 15.2. Summary in English

Code changes have always been a very important factor in software maintenance, and evolution. With the arrival of continuous delivery and deployment, changes to source code have become an integral part of system operation, and therefore subject to its constraints. This fact, combined with the high sensitivity of many systems to timely bug fixes and quick availability of new features, means that shortening time-to-deployment of changes is one of the most important factors to optimize, together with code quality. And one of the most influencing factors in time-to-deployment is time-to-merge (time until the change is merged in the code base). Therefore, achieving as short as possible time-to-merge, while keeping high quality standards, has become key in many software projects, in areas as different as cloud computing or embedded systems. But there is always a trade-off: quick bug fixing and feature implementation may increase the risk of producing new bugs and low quality code. Modern software development uses techniques such as peer code review and intensive testing to mitigate this risk. But this causes a new trade-off: the more effort devoted to these techniques, the less effort available for “real” development. Therefore, any reliable measure of the process should have all these factors this into account.

Fortunately, software development is nowadays supported by tools that allow the tracking of the complete process leading to merging changes in the source code. When the change intends to fix a bug, all actions since the reporting of that bug to the actual merging of the change in the code base are tracked in the issue tracking, code review, testing support, and source code management systems. When the change implements a feature request or some refactoring, all actions since the design to the actual commit of the change are available in the same systems. This project will use all that rich data to first, characterize the process that is followed from the conception of a change to its acceptance in the code base, and second, study how it can be improved, and which factors are key in maintaining high quality standards on the code base, while reducing time-to-merge as much as possible. The study will have into account all factors mentioned above, including the fact that many changes cause bugs that need to be fixed later, leading to a lower quality of the system meanwhile. In other words, the whole life of changes will be considered, through several “incarnations”, needed to fix bugs or to complete missing functionality.

For fulfilling these aims we will build upon the results of a previous project, SobreVision, which studied specifically code review, and the process of bug seeding. With an evolution of the methodologies developed in that project, we intend to compute the complete time from the moment a change is conceived, to the moment it is correctly merged in the code base, having into account the effect of bugs introduced by changes, and other effects leading to an incomplete implementation.

The project will have Bitergia as a collaborator. Bitergia was born as a spin-off of the research team submitting this proposal. Bitergia is very interested in the results of this project, because they are already marketing solutions quite related to it, and it is very likely they will be interested in adding these results to their commercial offer.



## Capítulo 16

# Scientific proposal

Changes to source code have always been the fundamental mechanism for letting a software system evolve, adapt to new needs, and fix errors that are found. These changes, once the software had been shipped to production, were the basis of software maintenance. During this stage of software development, changes were usually grouped into new “releases” or “service packs”, which addressed a certain amount of issues, and were shipped for production from time to time. In this scenario, deadlines for the complete releases were the main timing constraint; decisions on what changes were ready to be shipped with a specific release depended on these deadlines.

Continuous deployment of software, and continuous release procedures, which has enabled it, have changed dramatically the landscape. Many companies, working in diverse areas such as cloud computing or embedded systems, are deploying every single change to the code base in production, in a continuous way. In these cases, they “release” a new version, intended to be deployed in production, for every change to the code base, or for very small sets of changes to it. For example, Google is said to release most of its systems several times a day, with each of these releases going into production afterwards. Yahoo is said to have a target of about one to two days to deploy in production every new change to the code base, and even less if the change fixes an important bug.

In these cases, software operation is now completely integrated with software development, and its procedures are increasingly becoming an integral part of the continuous deployment processes. In particular, the demanding timing of the business which these systems support are translated to software development. Now, changes are no longer influenced only by the deadline of sparse releases, but pressed for a never-ending need to shorten time to deploy. Every time a change to the source code is requested, there is a pressing need to have that change in production as soon as possible. This happens both if the change is fixing a bug (which need to be fixed as soon as possible to avoid dangerous impacts on the users or on the system itself), or if it is including a new feature or optimizing some performance aspect (which usually can be translated to increased user satisfaction or decreased cost, both important in those competitive markets).

These pressing needs to deploy changes quickly conflict with another need, this one classical in software systems: to maintain the quality of the code as high as possible. Even when you want to deploy changes in production as soon as possible, you want to deploy those changes when their quality is good enough to fulfill their requirements. For example, when the change is intended to fix a bug, it is important that the bug is indeed fixed, and that no new bug is caused. When the change is intended to add some functionality, it is important that the functionality is indeed provided, with a reasonable performance, and without causing bugs.

To ensure that code is delivered quickly, but still it maintains a certain quality level, many different approaches are being used. Some of the more popular are code review, usually performed by peers before the code is committed (pre-commit peer review) and intensive testing (including unit and integration testing), sometimes called verification. In general, the most exhaustive these two practices are, and the more effort devoted to review and test every change, the better quality can be expected from the change. But again, there is a trade-off: all efforts devoted to review and testing are not devoted to “real” development, and therefore have an impact on the rate at which bugs are fixed and new features are implemented. And the longer it takes to review and test a change, the longer it takes to merge it into the code base, and therefore to have it ready for production.

In this project we are going to deal exactly with these problems. We will focus in the area of cloud computing, which is one of those where all these practices and problems are more clear, and more important to the correct functioning of the industry. It is also one where many free, open source software (FOSS) solutions are used in production, being in many cases the leaders in their respective markets. This will ensure that the results of the project are relevant, since they will be focused in one of the industries that more desperately are needed it, and that the project is feasible, since there is plenty of public data available for study.

The main aim of the project will be, therefore, to characterize, from a quantitative point of view, continuous software development in a large collection of open source projects, in the area of cloud computing. Once the main processes are quantified and characterized, the different trade-offs will be analyzed, looking for key parameters to optimize, and to detect good practices that maximize the quality of the resulting software while minimizing time-to-deployment.

One key aspect of our project is that it will consider changes not as isolated cases. On the contrary, the fact that a change potentially influences any further change will be taken into account. The fact that changes have unintended consequences will also be considered an integral part of the model. For example, if a change fixing a bug introduces a new bug itself, the impact on quality is clear, and the complete bug-fixing cycle does not finish with that change, but with the change that fixes the new bug (but only if the new one does not introduce yet another one). Similarly, if a change produced to address a feature request does only partially implement it, or introduces some bug while implementing it, it needs to be considered as a part of a longer process that will include the changes complementing the functionality or fixing the bugs introduced. Therefore, we will address the whole process, from conception of the change, to the final, correct implementation.

Of course, a change is not known to be correct a-priori. But after some time, it can be known if more changes were needed because the feature implemented was not complete, or because the change introduced some new bug that needed to be fixed. Therefore, looking backwards to the whole history of a project and its changes, it is possible to know when a certain feature was completely implemented, maybe after several changes, and no new bugs were found to be introduced by those changes.

To fulfill these aims, the project will benefit from the rich data available from many of these FOSS projects. It is nowadays a usual practice that they publish all the detailed data tracked by their development tools, including the source code management system, the issue tracking system, the code review system, and the testing support system. The combination of all of that data allows for a very detailed tracking of all the processes that lead to the changes merged into the code base.

This project will build on the results of an earlier project, *SobreVision*, which dealt with the characterization of code review processes in FOSS cloud-related projects, and with the identification of changes that introduced bugs. This way, *SobreVision* provided two of the building blocks that make this project possible.

## 16.1. State of the art

The research proposed in *SobreTodo* is framed in the empirical software engineering field. The relevant state of the art will be based on empirical and quantitative research on the main processes of interest: code review, automated testing, and bug seeding.

Although a part of its results are expected to be extensible to other projects, *SobreTodo* will focus on analyzing some of the most popular projects related to cloud computing which are developed as FOSS. In this projects, it is common nowadays that any proposal for a change starts with opening a ticket (a bug report, the description of a feature request, a proposal for a refactoring...). After discussions around that ticket, a first proposal for the change emerges, which is subject to peer pre-commit code review. During this process, the proposal is analyzed and commented, maybe requesting new versions of the change, until it is finally accepted or declined. Usually, before a change is considered for review, and after it has been accepted, a battery of automated tests are run (usually unit and integration tests). If the tests are passed, and the code review results in acceptance of the change, it is finally merged in the code base, which is considered ready for production. This whole process is the basis for “continuous release” or “continuous delivery”, a paradigm that is taking the software industry by storm although it is not free of problems and challenges [30].

But this is not the end of the complete process: after some time, maybe one or more bugs are found,

caused by the change, which still will need to be fixed. Or the change is found to be incomplete in implementing the feature it was intended to implement. In those cases, new issues, code review and testing take place. This is also known in the area of software evolution for many years [39], but has received little attention from a quantitative point of view, and comprehensively, as a complete process. This new, more complete panoramic on the life of changes will be the basis for SobreTodo. The relevant research areas, therefore, will be analysis of code review, characterization of software testing, and identification of bug seeding.

Research of these areas have been traditionally linked to the kind of repositories where the corresponding processes happen. But the idea of linking data from several repositories to obtain a higher level and more complete characterization software development processes is not new. For example, already in 2003 [31] explored how to link data from source code management systems and issue tracking systems to learn about the release process, and in 2005 [49] presented a similar composition to study the relationship between changes and bug fixes. Around 2008, several studies started to add data from code review systems to the dataset [44] [45], with the aim of characterizing the influence of changes and bug reports on the main parameters of code review, such as number of iterations until acceptance. Although there is still little activity linking all these processes to testing, some studies [43][42] have found how both unit and integration testing have a large influence in code review, issue handling and code change acceptance. In any case, the benefits of using data from different kinds of repositories to characterize the many facets having an impact in how changes to code are conceived, decided and accepted, are clear to the research community.

Most of this research has focused on understanding how pre-commit peer code review works. This is an increasingly common practice in software development, which in many research papers is named “modern code review”, to differentiate it from “classical code review”, which was usually performed by specific review teams after the code had been merged in the code base. Now, we know that this modern code review helps to maintain long term software quality of large and long lived projects [43] [42], and increases team awareness, knowledge sharing and creation of alternative solution to problems [27] [29]. Those are some of the reasons why many researchers have attempted to understand the successes, challenges and factors that impact the code review process [46] [28] [48] [29].

But code review does not happen as an isolated project in software development. On the contrary, it is clear now that it is a part of a longer process, which starts with change conception and includes discussions, decisions, and testing. This is to such an extent, that modern platforms for supporting software development are providing facilities for managing the whole process: as GitLab names it, “from idea to production” [33].

Therefore, all the research on empirical, quantitative characterization of the bug fixing, feature requesting and implementation [50], technical discussions related to software development [29], and integrated testing [32] [41] are relevant to SobreTodo.

The main contribution of our project will be to leverage on this state of the art to consider the complete process that continuously introduces changes (that are later changed once and again), characterizing it and using this characterization to find parameters that can be used in this fragile balance between being quick in delivering, but keeping high standards of quality.

## 16.2. Previous experience and relationship with other research groups

The research team proposing SobreTodo has a large experience in the analysis of software development repositories, acquired since projects such as FLOSSMetrics, QUALOSS, QUALIPSO, ALERT (FP6/FP7, 2008-2013) and SobreSale (Plan Nacional, 2012-2014) projects in which we produced and analyzed extensive datasets from several kinds of software repositories, including source code management, issue tracking, code review and asynchronous communication systems. These projects allowed us to build some of the most widely used tools for data retrieval and analysis in those kinds of repositories, MetricsGrimoire1, and perform some of the first analysis on many of them [47] [38] [35] [40] [36] [34] [39] [37].

As a result of this expertise, some of us founded, in 2012, a spin-off company of the research group, Bitergia2. Since then the company is marketing services based on the quantitative analysis of software development. During the first years, the company used and contributed to MetricsGrimoire, but since 2015 it has developed its own technology, GrimoireLab3. The relationship between the company and the

group is excellent, maintaining a very productive collaboration. This collaboration includes the use of GrimoireLab, and the collaboration in the implementation in them of features interesting for researchers, and the availability of datasets of some Bitergia clients. Bitergia is also interested in SobreTodo, and is ready to collaborate with high quality datasets and direct access to developers, ensuring that the research we do will be firmly grounded on reality.

During the last years, we are participating in Seneca, a Marie Skłodowska-Curie Action (Innovative Training Network. ITN, in the H2020 Framework) with Bitergia and two other companies from Greece and the Netherlands, and two research institutions (AUEB in Greece and Technical University of Delft in the Netherlands), in topics quite close to those of SobreTodo. We are also finishing the SobreVision project (funded under the Retos program), which started in 2015. In many respects, it is providing us with the ground work for SobreTodo, thanks to our research in code review and detection of bug seeding.

The research group is also in direct contact with the academic and industry groups that are researching in these topics. Even when this idea of studying comprehensive processes, using data from many different sources combined, is still relatively novel, an informal network for the exchange of research experiences and knowledge is starting to emerge, and we are a part of it. Among others, it is formed by Daniel German and Peter Rigby (University of Victoria, Canada), Thomas Zimmerman and Chris Bird (Microsoft Research, EE.UU.), Michael Godfrey (University of Waterloo, Canada), Ahmed Hassan (Queen's University, Canada), Tom Mens (University of Mons, Belgium), and Diomidis Spinellis (AUEB, Greece). With some of them (Daniel German, Tom Mens, Diomidis Spinellis) we are participating in international research projects.

### 16.3. Research and technical objectives

The main hypothesis of SobreTodo can be described as follows:

- **Applicability universe:** Software development projects following continuous delivery practices, and oriented to continuous deployment. Projects may be FOSS or not, but they should follow the support practices that are becoming commonplace when these practices are used: complete traceability of changes, pre-commit code review, and automated testing.
- **Scope of interest:** When it is important to minimize time-to-deployemt and time-to-merge, while at the same time it is needed to keep high quality standards, and in particular, to minimize the rate of bug seeding, and the iterations until complete implementation of features, when fixing bugs or implementing new features.
- **Hypothesis:** Metrics about the whole process, from a proposal of a change until it is completely and properly implemented, can be obtained automatically from information in the relevant software repositories. These metrics can be useful to fine tune, and act upon, the different acts involved in the discussion, implementation and testing of proposed changes. This allows to make decisions to optimize the full project from change proposal to change implementation.

This hypothesis is based on previous work, both from our team (mainly in the context of the SobreVision and Seneca research projects) and other international teams, as has been shown in the previous sections.

Even when the hypothesis is proposed for the specified universe, SobreTodo will be focused on FOSS development projects with public information, and a clear impact on industrial systems using continuous deployment, mainly in the area of cloud computing.

The hypothesis will be validated using a constructive approach: solving specific parts of the problem by using available data, and building upon these cases a general methodology and the tools needed to implement it.

From this hypothesis, the main objectives of SobreTodo can be summarized as follows:

“Development of a methodology, with the convenient software tools, for supporting software managers, developers and FOSS development communities to make decisions and put in place policies and rules oriented to improve time-to-merge of the overall implementation cycle of a software change, from requirements to complete and correct implementation.”

This methodology will be based on the empirical validation of its performance, by using statistical analysis on past data available. For that analysis, it will be important to identify complete processes corresponding to complete changes, as relevant for this objective. That means analyzing past changes, determining their causes by using information from the code review comments and issue tickets that lead to them, and finding out when they correspond to fixes of bugs introduced by previous changes, or to incomplete earlier implementations. Thus, the build of detailed data-sets, rooted in information from issue tracking, code review, testing and source code management systems will be an important part of the project. A preliminary version of these datasets have already been produced in the context of SobreVision and Alert, for software projects such as OpenStack and ElasticSearch, although still some work is needed to complete the data, and to extend the approach to more software projects.

With these datasets ready, main parameters can be identified and measured, such as time-to-merge of complete processes leading to successful changes, impact on quality (measured for example by tests passed, or by later bugs found), and factors that could influence both, such as which parts of the code are affected, who is proposing and reviewing changes, or the test coverage of those changes.

## 16.4. Specific goals

The general objectives mentioned above can be split in the following specific goals:

- G1: Tools. Creation of a set of tools, in part by adapting already existing tools, to assist in the creation of data sets with empirical data about complete change processes to source code. These tools will be based on GrimoireLab, and in the software developed in the context of SobreVision and Seneca, where preliminary versions of the needed datasets were created. The tools will automate the process as much as possible, and will be adaptable to different kinds of projects and repositories, so that they can be applied to a variety of cases. They will support all the stages considered in SobreTodo: data retrieval and storage, data clean-up and organization, computation of metrics, and statistical and big data analysis. The tools will be FLOSS, so that other researchers and practitioners can adapt them to their needs.
- G2: Data sets. Creation of a collection of data sets with relevant information for the analysis of complete processes of change to the source code, including the many changes until the code is ready and correct, and related data which can be used to determine influencing factors. Data sets will be produced with the assistance of tools developed for G1, but will be also validated manually, to ensure that the results of the project are correct. Data sets will be designed and published with the idea of using them both for benchmarking, and for reproducible research, in addition to fulfilling the specific goals of SobreTodo.
- G3: Estimation of time-to-merge. Using the data sets produced for G2, this goal will focus on the development of a methodology, as automated as possible, to determine all change processes involved in the complete process of an intended change to source code. This will allow for the measurement of time since the conception of the change (as a result of a bug report, or a feature request, usually) to the finalization of the last change that completed its correct implementation.
- G4: Estimation of intermediate quality. While a complete change is taking place, bugs can be seeded, and implementations of features found to be incomplete. During this time, the quality of the software is less than expected, and using that code to production can be troublesome. Using the data sets produced for G2, this goal will focus on studying the results of testing (including tests that will be implemented later than the first change), future bugs fixed, and future completions of partially implemented features. As a result, metrics for the intermediate quality at any given time will be determined. In addition to this definition of quality based on partially solved problems (or newly introduced problems), some other more classical metrics for quality, such as those based on static source code analysis, could be used to complete the model.
- G5: Determination of influencing factors. Using the metrics defined for G3 and G4, and data for potentially influencing factors in the data sets produced for G2, several statistical and big data analysis will be performed to determine patterns and factors of real influence, which affect time-to-merge, changes until complete implementation, and quality.

- G6: Complete analysis methodology. Based on the results for G3, G4, a complete methodology for helping decision makers to take decisions on the balance between quality and time-to-merge, will be developed. This methodology will have into account the results obtained for G5, which will ensure that they can draw recommendations and conclusions based on the experience of influencing factors in the projects in the collection of data sets.

In addition to the more theoretical results of G3, G4 and G5, G1 and G2 ensure the production of tools and data sets that will be openly shared, and will help by themselves to advance the state of the art in this area. G6 will combine most of the results together, aiming to fulfill the general objective of the project, as stated previously.

# Capítulo 17

## Realization

### 17.1. Methodology

The structure of the project suggests an iterative methodology for its implementation, starting with data from two projects, and a partial analysis of the data in the first iteration, and later extending to a broader collection of projects, with more thoughtful and complete analysis, until a proven methodology emerges. We envision three iterations:

- The first iteration will be focused on OpenStack and Elasticsearch projects. The SobreTodo team will benefit from their deep knowledge of these projects, and the data sets that they already produced for them in SobreVision and Seneca. These data sets will be completed (currently they don't have testing data, and the information about chains of changes is still incomplete), and used to start automating the task of producing them. Then, at least basic metrics about time-to-merge for the complete change process, and quality of changes will be defined and computed.
- The second iteration will include some more important FOSS projects in the area of cloud computing, so that the data remains relevant, and to some extent, comparable. The tools produced during the first iteration will be used to ease the production of the new data sets, and those will be already be complete, with all information needed for the stated goals, including data about potentially influencing parameters. This will allow to have results targeted towards goals G3, G4 and G5. The first sketches of a general methodology will be produced as well.
- The third iteration will extend the collection of data sets even further, now including some projects in other realms. The lessons learned during the first two iterations, and the tools developed, should help to produce those with much less effort. Metrics for all factors related to goals G3, G4 and G5 will be computed, and the general methodology will be developed and validated on this extensive collection of data sets.

Each iteration will include the results of previous iterations, progressing towards more general, and valid, results. Each iteration will also be used to re-validate the results of the previous one, trying to avoid overfitting.

When possible, two iterations will happen partially in parallel. For example, while finishing the computing of the metrics for the first iteration, the data retrieval and tool enhancement, and needed for the second iteration can start. In any case, we will do a careful monitoring of progress so that this intended parallelization does not cause delays in the final stages of the earlier iteration.

The most critic tasks in SobreTodo will be the definition of the methodology (goal G6). The overall success of the project will to some extent depend on its validity and generality, at least for the projects in the universe of relevance. Therefore, in all respects, the project will be grounded on data, but also on insights provided by developers and other members of the developing community. For this, the strong relationship with Bitergia will allow for the contacts needed to validate data and results, and to dig deep in the specifics of each analyzed project. This will include also the validation of the resulting methodology with experts from Bitergia and from the analyzed projects, so that they can assess on the interest of the

results, and the usefulness of the methodology for practitioners. This validation will take place in each of the three iterations.

SobreTodo will implement a continuous risk monitoring procedure, with periodic (each two-three months) reviews of the progression, and comparison with plans. The main risks expected are that the production of data sets is more complex than expected (we have good knowledge of data sources, but in case this happens, we would limit the number of projects analyzed), that the identification and computation of metrics is more complex than expected, because of peculiarities of projects or of the available information (in this case we would simplify the model, and if needed reduce the expectations about the generality of the metrics), and that the overall model is more difficult to produce and/ or validate than expected. In fact, this latter is the most important risk we face, and it will be managed by starting early in the life of the project (during the second iteration) to develop the model and methodology, even when it will be refined later. If we fail to produce a model and methodology general enough, at least we will explore which aspects of it are general, and how it could be determined for specific sets of projects, fulfilling certain conditions.

As a final step for each iteration, a report on it will be produced, including change to the original plans, if the progress of the project makes that convenient.

## 17.2. Infrastructure and equipment

The infrastructure and equipment relevant for the execution of this project are:

- Computing nodes for the collection and analysis of data sets, and their analysis. We will use at least a set of always-on computers, with enough data storage and computing power, for the collection and analysis. Since the tools will be developed by us, based on FOSS tools and systems, no specific need of licenses for acquiring software are envisioned (see software, below). Some of these nodes could be in the cloud, and some are expected to be provided by Bitergia, under the agreement we already have with them. Initially, we are budgeting for the acquisition of four of these computing nodes, and budget for 48 months of cloud-based nodes.
- Computers for the development of supporting tools, front-end to data collection and analysis, and general work by the researchers. These will be their own work stations (usually laptops with enough resources)
- Software. All the software used will be FOSS. In particular, tools will be developed in Python, with intensive use of Pandas and other Python packages for the analysis. Elasticsearch is the platform intended for data storage.

In addition, we will need some small supporting equipment and materials, with very little impact in the total budget.

## 17.3. Schedule

The schedule of tasks to perform is offered below. The realization of the project is designed for a participation of six persons working full time in it: the main researcher (who is writing this proposal), an associate researcher, with deep experience in the field, three PhD students (two of which are funded by projects related and compatible with this one) and an IT engineer with will provide support to the team, and will collaborate mainly in software development activities (see below for details, in section 17.5).

- **T1: Project kick-off.** Acquisition plan, deployment and configuration of equipment, selection process for an IT engineer.  
Estimated duration: 3 months  
Participants: all the team
- **T2: Management.** Management of all the project, including execution of acquisition, administrative management, risk management, coordination and progress review activities, and reporting activities. This task will be active during all the project, will need effort only at specific points in



Task	Quarters										
T1	█										
T2	█	█	█	█	█	█	█	█	█	█	█
T3	█	█	█	█	█	█	█				
T4			█	█	█	█	█	█			
T5				█	█	█	█	█	█	█	█

Figura 17.1: Scheduling of tasks over time. Time in quarters, starting with the project kick-off.

time.

Estimated duration: 36 months

Participants: main researcher

- **T3: First iteration.** As defined in C.1.5.  
 Estimated duration: 12 months  
 Participants: main researcher, associate researcher, PhD students (3), IT engineer
- **T4: Second iteration.** As defined in C.1.5.  
 Estimated duration: 15 months  
 Participants: main researcher, associate researcher, PhD students (1), IT engineer
- **T5: Third iteration.** As defined in C.1.5.  
 Estimated duration: 15 months  
 Participants: main researcher, associate researcher, PhD students (1), IT engineer

Figure 17.1 shows the scheduling of these tasks over time.

## 17.4. Deliverables

Deliverables to be produced are of several types:

- Scientific reports: including theoretical and methodological details, and all the information needed to reproduce the studies.
- Management reports
- Tools: complete tool sets, including tools developed by the project and other supporting tools.
- Data sets: complete data produced by each iteration of the project, corresponding to the collection of projects considered in that iteration.

Figure 17.2 shows the different deliverables, and their relationship with the defined milestones.

## 17.5. Personnel

SobreTodo will be directed by its two main researchers, one acting as main researcher (the one writing this proposal), and the other one as associate researcher. Both will work full time in it. Both will share the management responsibilities, will produce the detailed design of the project and its activities, will review results, and in general, will ensure that the work progresses and is of the expected quality. In addition, the main researcher will be direct responsible for the tools, and the associate researcher for the datasets. Both will collaborate in the design and production of the methodology, and the design of the analysis to be performed.

Milestone	Name of milestone	Date (month)	Deliverable
M1	Kick-off activities complete	3	
M2	First iteration finished	12	D2.1: tools (1st version) D2.2: data sets (1st version) D2.3: First, very limited, methodology D2.4: First iteration report
M3	Second iteration finished	24	D3.1: tools (2nd version) D3.2: data sets (2nd version) D3.3: Second methodology D3.4: Second iteration report
M4	Third iteration, and project, finished	36	D4.1: tools (3rd version) D4.2: data sets (3rd version) D4.3: Final methodology D4.4: Final project report

Figura 17.2: Scheduling of tasks over time. Time in quarters, starting with the project kick-off.

Some other persons will collaborate in the project, working also full time (during the time specified for each of them):

- PhD Student 1. She has been included only during the first 12 months of the project, since then is when she is expected to finish her PhD. But we expect that she remains in the team later, thanks to a teaching contract with the University. She will participate in the analysis and data retrieval during the first iteration, and if she is still part of the team, in the next iterations too.
- PhD Student 2. She is currently a PhD student in the Seneca project, directed by the main researcher of this proposal. She will stay in the team until Seneca finishes, at the end of 2018. She will participate in the analysis and data retrieval during the first iteration.
- PhD Student 3 hired for working in SobreTodo, which will collaborate in the research aspects of the project, including the retrieval, and analysis of data, the computing of metrics, and the definition of the methodology and model.
- Junior IT Engineer, hired for working in SobreTodo, which will work in the deployment and maintenance of all the infrastructure, and the development and fine-tuning of all the tools produced by the project.

Of all the persons collaborating in the project, only PhD Student 3 and Junior IT Engineer will be hired with the budget of this project, since the rest of the personnel already have their own funding which is compatible with working in this project.

In addition, sporadic collaboration with Bitergia personnel is expected (Bitergia is a company with direct interest in the results of the project, as is explained elsewhere in this proposal).

## Capítulo 18

# Impact and training

SobreTodo will contribute significantly to the improvement of the development and maintenance processes of continuous delivery and continuous deployment systems, that are increasingly common in many areas, such as cloud computing. This improvement will be ensured by a new model which considers all changes to source code that are needed to implement a bug fix or a feature request, together with a methodology to support taking decisions on the balance between time-to-merge (or time-to-deployment) and quality. The model and the methodology will also have into account the factors that may impact on these two metrics, to help find solutions that help to find the sweet spot balancing them.

In addition, we will produce data sets for a collection of projects, and tools for computing the relevant metrics for projects using the same kinds of repositories considered in SobreTodo. The data sets will be fundamental for developing benchmarking schemas, that allow development teams to learn about how they behave with respect to other projects, but also for researchers, since they can concentrate in finding new impacting factors, or new ways of computing metrics (and metrics to compute), with reproducible results, by building on the work of others. The availability of tools will be relevant to the industry, who will be able of using them, or evolving them to suit their needs, since they will be FLOSS. In addition, researchers will be able of using them to create their own data sets, if they need that, or to update and complement those provided by the project.

Even when the main objective of SobreTodo is to produce results useful for FOSS projects related to cloud computing, the scope of interest is in fact broader: all software development projects using modern techniques for continuous-deployment, in which both time-to-deployment and quality are fundamental. The relationship that the research team, and Bitergia, have with FOSS projects in these areas will ensure at least some level of impact. It is expected that the universe of applicability is in fact much wider, and will expand even more in the future, since more and more sectors in industry are adopting these practices.

The impact of the project is expected to be both national and international. From a national point of view, the know-how will be developed locally, and thanks to the interaction with Bitergia, will have a clear way to the production sector (see section 18.1 below), including its application in national companies interested in these matters. This will help to increase both the scientific and industrial competitiveness of national actors.

At the international level, both the research team and the company will see their current leading positions strengthened, being reinforced as centers of know-how (scientific or industrial) in this field.

### 18.1. Transfer of results

SobreTodo is designed to favor technology transfer in two complementary ways:

- From a scientific point of view, the resulting methodology, tools and data sets will be publicly available at the end of the project. In fact, tools will be developed as FOSS tools, and will be public as they are built. Other researchers will be encouraged to reuse them, and to reproduce the results we will obtain with them. Same can be said for data sets. Methodologies will be fully public after publication in academic venues. Data sets will be for popular, widely studied projects (such as OpenStack, Elasticsearch, and many others), which should help in the interest for reusing

our results, and in converting them, in the medium term, in de-facto standards in the academic community for this field.

- From an industrial point of view, Bitergia, the company collaborating in this project, will include the results they find interesting from a commercial point of view. This will ensure a commercial offer and the adequate support for the technology transfer to the productive sector. Bitergia already has an impressive portfolio of international customers (some of which will participate indirectly in the project, with their expertise in interpreting results) which have shown interest by results in this area.

## 18.2. Dissemination plan

The dissemination plan is tightly linked to the technology transfer strategy shown above. Its main lines are as follows:

- Traditional Internet-based dissemination means: web site, social network channels, blog posts, etc. Participants will commit to contribute through these channels, so that information about the project and its progress will be widely available.
- Academic dissemination in relevant journals and conferences (according to JCR and other indexes such as the GII-GRIN-SCIEE for conferences). This will include the usual preparation of papers, but also the proposal of specific workshops on the matter, and the contribution of datasets to academic venues, such as the Mining Software Repositories Challenge. The team will also participate in Summer or Winter Practical Schools for junior researchers, showing the tools, the data sets, and the preliminary versions of the methodology.
- Dissemination to the software development communities analyzed, in their own conferences and venues. Some of the team members are already regular of these venues, where they will present the tools and intermediate results as the project progresses. Since the development communities are the subject of the analysis, the interest in the results for them is increased.
- Integration of tools in wider platforms, with a community of developers around them (such as GrimoireLab), so that there are more chances of them getting attention, being used, and being improved. This will benefit both practitioners and academics, since more mature tools means tools easier to use.
- Collaboration with Bitergia in the definition of commercial services based on the results of Sobre-*Todo*. One of the main interests of the company in this project is to identify services and products that can be built based on its results. This will help to the dissemination of results to the industry.
- Publication of data sets in appropriate venues, so that other academics can reuse and improve them.

This way, we try to convert both the tools and the data sets into conveyors for transfer of results, both to the academic community and to the industry, supported by a commercial strategy.

## 18.3. Training capabilities and program

Since most than a decade ago, the GSyC/LibreSoft team, who is proposing Sobre*Todo*, is a leading research team at the international level in the quantitative, empirical study of large FOSS development communities. The group has maintained a research team of varying sizes for almost 20 years, funded with different R&D and technology transfer projects, both result of competitive calls by public bodies (Spanish Government, European Commission) and Art. 83 contracts with industry. At the time of writing this proposal, the team is composed of 8 persons (6 FTE), of which 4 are participating in this project.

Currently, the team participates in Seneca (ITN, Innovative Training Network, Marie Skłodowska-Curie Actions, H2020 Program) funded by the European Commission which funds one full-time researcher who is earning her PhD and hosts two others from participating institutions from other countries; in Sobre*Vision* (Retos, started in 2015) funded by the Spanish Government which also funds one full-time researcher who

is earning her PhD; in an Art. 83 contract funded by Bitergia which funds two part-time master students; and in two grants by Comunidad de Madrid which fund two part-time students, and some other smaller funding.

The training program that the research team offers to its junior researchers during the period they work towards their PhD thesis includes:

- Doctorate students will enroll in the Doctorate School of Universidad Rey Juan Carlos, where they will have their personal plans, agreed with their thesis directors, and reviewed by the academic committee of the School. Those plans are updated and reviewed yearly, and have to fulfill the requirements defined by the School (such as a certain number of training credits per year, and a certain performance in publication).
- Students will participate in a regular research seminar, held weekly, with the frequent participation of invited researchers, both national and international, from the rich network of academic relationship that the team maintains.
- Students will participate in activities organized by the Doctorate School, which offers a wide set of training options. Those include both horizontal activities, aimed at improve the general profile of students as researchers, and specific activities in the research fields it covers. In this School, our students enroll in the IT topic, “Study of Software Development” research line, lead by Jesús M. González Barahona.
- Students will have stays in well-known foreign groups. Thanks to our policy of exchanges and contacts with other research teams, and to the availability of funding, all PhD students in our team that took the opportunity have performed stays for at least three months (and in many cases, six or nine) during their PhD studies, in one or more foreign institution, world-class in our field. These stays are complemented with the access to researchers visiting our group. During the last years, we have received researchers, in stays of two weeks or more from the following research institutions: University of Skövde (Sweden), INRIA (France), University of California at Santa Cruz (USA), University of Syracuse (USA), University Paris-Diderot (France), University of Victoria (Canada), Chalmers University of Gothenburg (Sweden), Oregon State University (USA), University of Mons (Belgium), University of East London (UK), Technical University of Eindhoven (Netherlands), TU Delft (Netherlands), etc.
- Students will participate in International Summer and Winter Schools in our field, and other similar activities. Current PhD students are participating at least in one of these per year. They will also be encouraged to participate in Doctoral Consortia of main conferences in our field. During 2017, for example, our group has organized in our premises an intensive workshop in the context of the Seneca project, focused on technology transfer and life after earning the PhD, and SaTTOSE, a 3-day technical conference on techniques for software engineering.
- Students will participate in formal and informal R&D networks, allowing for the virtual (and in some case, in-place) collaboration with researchers in other teams. Some of the results of this participation, in addition to the mentioned research outputs and visits, are the exchange of members of the thesis reviewing committees. This has allowed us to have at least one relevant foreign member in all our thesis committees (and usually, two of them).
- Students will receive support and incentives for open publication and FOSS distribution of software. In addition, they will receive training and help in how to publish open access, how to release FOSS, and how to produce reproducible research.



# Research Project References

- [27] Alberto Bacchelli and Christian Bird. Expectations, outcomes, and challenges of modern code review. In IEEE Press, editor, *Proceedings of the 2013 International Conference on Software Engineering*, page 712–721, 2013.
- [28] O. Baysal, O. Kononenko, R. Holmes, and M.W. Godfrey. Investigating technical and non-technical factors influencing modern code review. *Empirical Software Engineering*, 21(3):932–959, 2016.
- [29] A. Bosu, J.C. Carver, C. Bird, J. Orbeck, and C. Chockley. Process aspects and social dynamics of contemporary code review: insights from open source development and industrial practice at Microsoft. *IEEE Transactions on Software Engineering*, 43(1):56–75, 2017.
- [30] L. Chen. Continuous delivery: Huge benefits, but challenges too. *IEEE Software*, 32(2):50–54, 2015.
- [31] M. Fischer, M. Pinzger, and H. Gall. Populating a release history database from version control and bug tracking systems. In IEEE Computer Society, editor, *Proceedings of the International Conference on Software Maintenance (ICSM 03)*, 2003.
- [32] V. Garousi and M.V. Mäntylä. A systematic literature review of literature reviews in software testing. *Information and Software Technology*, (80):195–216, 2016.
- [33] GitLab. Idea to production demo. Web page (last visited: July 2017). <https://about.gitlab.com/handbook/product/i2p-demo/>.
- [34] Jesus M. Gonzalez-Barahona, Daniel Izquierdo-Cortazar, Stefano Maffulli, and Gregorio Robles. Understanding how companies interact with free software communities. *IEEE Software*, 30(5):38–45, 2013.
- [35] Jesus M. Gonzalez-Barahona, Daniel Izquierdo-Cortazar, and Megan Squire. Repositories with public data about software development. *International Journal of Open Source Software & Processes*, 2(2):1–13, 2010.
- [36] Jesus M. Gonzalez-Barahona and Gregorio Robles. On the reproducibility of empirical software engineering studies based on data retrieved from development repositories. *Empirical Software Engineering*, 17(1-2):75–89, 2012.
- [37] Jesus M. Gonzalez-Barahona, Gregorio Robles, Israel Herraiz, and Felipe Ortega. Studying the laws of software evolution in a long-lived floss project. *Journal of Software: Evolution and Process*, 26(7):589–612, 2014.
- [38] Jesus M. Gonzalez-Barahona, Gregorio Robles, Martin Michlmayr, Juan Jose Amor, and Daniel M. Germán. Macro-level software evolution: a case study of a large software compilation. *Empirical Software Engineering*, 14(3):262–285, 2009.
- [39] Israel Herraiz, Daniel Rodríguez, Gregorio Robles, and Jesus M. Gonzalez-Barahona. The evolution of the laws of software evolution: A discussion based on a systematic literature review. *ACM Computing Surveys*, 46(2):28:1–28:28, 2013.
- [40] Daniel Izquierdo-Cortazar, Andrea Capiluppi, and Jesus M. Gonzalez-Barahona. Are developers fixing their own bugs?: Tracing bug-fixing and bug-seeding committers. *International Journal of Open Source Software & Processes*, 3(2):23–42, 2011.

- [41] M. Kassab, J. DeFranco, and P. Laplante. Software testing practices in industry: The state of the practice. *IEEE Software*, 2016. Early publication: 2016 (pending paper publication).
- [42] O. Kononenko, O. Baysal, and M.W. Godfrey. Code review quality: how developers see it. In *Proceedings of the 38th International Conference on Software Engineering*, pages 1028–1038, May 2016.
- [43] Shane McIntosh, Yasutaka Kamei, Bram Adams, and Ahmed E Hassan. An empirical study of the impact of modern code review practices on software quality. *Empirical Software Engineering*, pages 1–44, 2015.
- [44] Weissgerber P, Neu D, and Diehl S. Small patches get in! In *Proceedings of the 2008 International Working Conference on Mining Software Repositories*, pages 67–76, 2008.
- [45] Rigby PC, German DM, and Storey MA. Open source software peer review practices: a case study of the Apache server. In *Proceedings of the 30th International Conference on Software Engineering*, page 541–550, 2008.
- [46] Peter Rigby, Brendan Cleary, Frederic Painchaud, Margaret-Anne Storey, and Daniel German. Contemporary peer review in action: Lessons from open source development. *IEEE Software*, 29(6):56–61, 2012.
- [47] Gregorio Robles, Jesús M. González-Barahona, and Juan Julián Merelo Guervós. Beyond source code: The importance of other artifacts in software development (a case study). *Journal of Systems and Software*, 79(9):1233–1248, 2006.
- [48] Junji Shimagaki, Yasutaka Kamei, Shane McIntosh, Ahmed E Hassan, and Naoyasu Ubayashi. A study of the quality-impacting practices of modern code review at Sony Mobile. In *Proceedings of the 38th International Conference on Software Engineering Companion*, page 212–221. ACM, 2016.
- [49] J. Sliwerski, T. Zimmermann, and A. Zeller. When do changes induce fixes? In *Working Conference on Mining Software Repositories (MSR'05)*, pages 24–28, Saint Louis, Missouri, USA, May 2005.
- [50] T. Zhang, H. Jiang, X. Luo, and A.T. Chan. A literature review of research in bug resolution: Tasks, challenges and future directions. *The Computer Journal*, 59(5):741–773, 2016.



Parte IV

Curriculum vitae



Gracias a la vida  
que me ha dado tanto  
Me ha dado la risa  
y me ha dado el llanto

---

Violeta Parra



# Capítulo 19

## Datos generales

Apellidos: **González Barahona**    Nombre: **Jesús M.**  
D.N.I.: **XXXXXXXXXX**    Fecha de nacimiento: **XX.07.1966**    Sexo: **Varón**

---

### Situación profesional actual

Organismo: **Universidad Rey Juan Carlos**  
Facultad, Escuela o Instituto: **Escuela Técnica Superior de Ingeniería de Telecomunicación**  
Depto./Secc./Unidad estr.: **Dep. Teoría de la Señal y Com. y Sist. Telemáticos y Computación**  
Dirección postal: **c/ Camino del Molino s/n  
28943 Fuenlabrada, Madrid**

Teléfono:  
Fax:  
Correo electrónico: **jgb@gsyc.es**  
Especialización (Códigos UNESCO): **330413**  
Categoría profesional: **Profesor Titular de Universidad**    Fecha de Inicio **mayo 2001**  
Situación administrativa

Plantilla     Contratado     Interino     Becario   
Otras situaciones  especificar:

Dedicación:    A tiempo completo   
                  A tiempo parcial

---

### Líneas de investigación

Estudio empírico del desarrollo de software libre  
Aplicación de TIC al proceso de formación

---

### Formación académica

<u>Doctorado</u>	<u>Centro</u>	<u>Fecha</u>
<b>Doctor Ingeniero de Telecom.</b>	<b>E.T.S.I. Telecom. (U.P.M)</b>	<b>febrero 1998</b>
<u>Titulación superior</u>	<u>Centro</u>	<u>Fecha</u>
<b>Ingeniero Superior de Telecom.</b>	<b>E.T.S.I. Telecom., U.P.M.</b>	<b>diciembre 1991</b>

---

**Actividades anteriores de carácter científico profesional**

<u>Puesto</u>	<u>Institución</u>	<u>Fechas</u>
Profesor Titular de Universidad	Universidad Rey Juan Carlos	2001.05 —
Profesor Asociado (tiempo completo)	Universidad Rey Juan Carlos	1999.10 — 2001.05
Ayudante Doctor	Universidad Carlos III de Madrid	1998.04 — 1999.09
Profesor Asociado (tiempo completo)	Universidad Carlos III de Madrid	1995.10 — 1998.03
Ayudante de Universidad	Universidad Carlos III de Madrid	1993.10 — 1995.09
Profesor asociado	Universidad Carlos III de Madrid	1992.02 — 1993.09
Administrador del Sistema Informático	Sociedad Estatal V Centenario	1991.03 — 1992.09

---

**Idiomas (R = regular, B = bien, C = correctamente)**

<u>Idioma</u>	<u>Habla</u>	<u>Lee</u>	<u>Escribe</u>
Inglés	C	C	C
Alemán	R	B	B

## Capítulo 20

# Indicios de calidad

### 20.1. Indicios generales

- DBLP (noviembre de 2017): 92 publicaciones revisadas (artículos en revistas, ponencias en congresos, etc.).  
[http://dblp.dagstuhl.de/pers/hy/g/Gonz=acute=lez=Barahona:Jes=uacute=s\\_M=.html](http://dblp.dagstuhl.de/pers/hy/g/Gonz=acute=lez=Barahona:Jes=uacute=s_M=.html)
- Google Scholar (noviembre de 2017):
  - H-index: 35
  - 3778 citas  
<https://scholar.google.es/citations?user=vYIPWBOAAAAJ>
- Co-fundador de una empresa de base tecnológica (Bitergia, 2012), con más de 5 años de funcionamiento.  
<http://bitergia.com>
- Co-autor de GrimoireLab, conjunto de herramientas para la analítica del desarrollo de software (2015-)  
<http://grimoirelab.github.io>
- Co-autor de MetricsGrimoire, conjunto de herramientas para la analítica del desarrollo de software (2008-2015)  
<http://metricsgrimoire.github.io>

### 20.2. Periodo 2011-2016

Indicios de calidad de las principales aportaciones mencionadas en este curriculum vitae, de acuerdo con los criterios especificados para el subcampo de evaluación 6.2 (Ingenierías de la Comunicación, Computación y Electrónica) en la Resolución de 26 de noviembre de 2016, de la Presidencia de la Comisión Nacional Evaluadora de la Actividad Investigadora, por la que se establecen los criterios específicos en cada uno de los campos de evaluación.

Estos indicios se refieren al periodo 2011-2016.

[Las referencias completas a los artículos y ponencias mencionados aquí están disponibles, con todo detalle, más adelante en este curriculum vitae, en los apartados correspondientes]

Publicaciones de artículos en revistas científicas listadas en JCR:

- Publicación del artículo “Studying the laws of software evolution in a long-lived FLOSS project” (2014) en la revista “Journal of Software: Evolution and Process”. Listada en el JCR 2014 (Science Edition), y otros años, en la categoría “COMPUTER SCIENCE, SOFTWARE ENGINEERING” (posición 80 de 104, Q4 en 2014). Factor de impacto: 0.62.

- Publicación del artículo “Understanding How Companies Interact with Free Software Communities” (2013) en la revista “IEEE Software”. Listada en el JCR 2013 (Science Edition), en la categoría “COMPUTER SCIENCE, SOFTWARE ENGINEERING” (posición 35 de 105, Q2 en 2013). Factor de impacto: 1.23.
- Publicación del artículo “The evolution of the laws of software evolution: A discussion based on a systematic literature review” (2013) en la revista “ACM Computing Surveys”. Listada en el JCR 2013 (Science Edition), en la categoría “COMPUTER SCIENCE, THEORY AND METHODS” (posición 3 de 102, Q1 en 2013). Factor de impacto: 4.04.
- Publicación del artículo “On the reproducibility of empirical software engineering studies based on data retrieved from development repositories” (2012) en la revista “Empirical Software Engineering”. Listada en el JCR 2012 (Science Edition), en la categoría “COMPUTER SCIENCE, SOFTWARE ENGINEERING” (posición 38 de 105, Q2 en 2012). Factor de impacto: 1.18.

Ponencias en congresos científicos listados en CORE 2014 (peer-reviewed long papers):

- “Software Engineering Artifact in Software Development Process - Linkage Between Issues and Code Review Processes”: International Conference on Intelligent Software Methodologies, Tools and Techniques 2016, CORE B
- “A Quantitative Examination of the Impact of Featured Articles in Wikipedia”. ICSOFT 2011, CORE B
- “Estimating development effort in Free/Open source software projects by mining software repositories: a case study of OpenStack”: International Conference on Mining Software Repositories 2014, CORE C.
- “Intensive metrics for the study of the evolution of open source projects: case studies from apache software foundation projects”: International Conference on Mining Software Repositories 2013, CORE C.

### 20.3. Periodo 2005-2010

Indicios de calidad de las principales aportaciones mencionadas en este curriculum vitae, de acuerdo con los criterios especificados para el subcampo de evaluación 6.2 (Ingenierías de la Comunicación, Computación y Electrónica) en la Resolución de 18 de noviembre de 2009, de la Presidencia de la Comisión Nacional Evaluadora de la Actividad Investigadora, por la que se establecen los criterios específicos en cada uno de los campos de evaluación.

Estos indicios se refieren al periodo 2004-2010.

[Las referencias completas a los artículos y ponencias mencionados aquí están disponibles, con todo detalle, más adelante en este curriculum vitae, en los apartados correspondientes]

Publicaciones de artículos en revistas científicas listadas en JCR:

- Publicación del artículo “Macro-level software evolution: a case study of a large software compilation” (2009) en la revista “Empirical Software Engineering”. Listada en el JCR 2009 (Science Edition), y otros años, en la categoría “COMPUTER SCIENCE, SOFTWARE ENGINEERING” (posición 27 de 93 en 2009). Factor de impacto: 1.612. Factor de impacto (5 años, 2009): 2.039.
- Publicación del artículo “Geographic Origin of Libre Software Developers” (2008) en la revista “Information Economics and Policy”. Listada en el JCR 2008 (Social Sciences Edition), en la categoría “ECONOMICS” (posición 82 de 209 en 2008). Factor de impacto: 0.917. Factor de impacto (5 años, 2008): 0.925.
- Publicación del artículo “Beyond source code: The importance of other artifacts in software development (a case study)” (2006) en la revista “Journal of Systems and Software”. Listada en el JCR (Science Edition) 2006, y otros años, en las categorías “COMPUTER SCIENCE, SOFTWARE ENGINEERING” y “COMPUTER SCIENCE, THEORY AND METHODS”. Factor de impacto: 0.592. Factor de impacto (5 años, 2009): 1.29.



Ponencias en congresos científicos listados en CORE 2008 y ERA 2010 (categorías A y B):

- Ponencias en HICSS. CORE 2008 y ERA 2010: categoría A (máxima).
  - “On the Analysis of Contributions from Privileged Users in Virtual Open Communities” (2009)
  - “Using Software Archaeology to Measure Knowledge Loss in Software Projects Due to Developer Turnover” (2009)
  - “On the Inequality of Contributions to Wikipedia” (2008)
- Ponencia en ICSM. CORE 2008 y ERA 2010: categoría A (máxima).
  - “On the prediction of the evolution of libre software projects” (2007)
- Ponencia en ISCC. CORE 2008 y ERA 2010: categoría B.
  - “A quantitative approach to the use of the Wikipedia” (2009)
- Ponencia en COMPSAC. CORE 2008 y ERA 2010: categoría B.
  - “Managing Libre Software Distributions under a Product Line Approach” (2008)
- Ponencia en WIKISYM. CORE 2008 y ERA 2010: categoría B.
  - “Quantitative analysis and characterization of Wikipedia requests” (2008)
  - “Quantitative analysis of the Wikipedia community of users” (2007)
- Ponencia en WCRE. CORE 2008 y ERA 2010: categoría B.
  - “A Model to Understand the Building and Running Inter-Dependencies of Software” (2007)

Artículos o ponencias con alto número de referencias según Google Scholar:

- 100 citas (dic. 2010): “Applying Social Network Analysis to the Information in CVS Repositories” (MSR 2004)
- 66 citas (dic. 2010): “Remote analysis and measurement of libre software systems by means of the CVSSAnalY tool” (RAMSS '04)
- 59 citas (dic. 2010): “Evolution and growth in large libre software projects” (IWPSE 2005)



## Capítulo 21

# Participación en Proyectos de I+D financiados en convocatorias públicas

---

Título del proyecto: **SENECA. Software Engineering in Enterprise Cloud Applications Systems**

Entidad Financiadora: *Comisión Europea, H2020 ITN-2014, 642954*

Entidades participantes: *Uni. Delft (NL), SIG (NL), URJC (ES), Bitergia (ES), Uni. AUTH (GR), SingularLogic (GR)*

Duración desde: *Enero 2015* hasta: *Diciembre 2018*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *2 (en el grupo de la URJC)*

Presupuesto (euros): *247.873 (URJC), 2.250.000 (total)*

---

Título del proyecto: **SobreSale, Análisis de patrones de introducción de errores en comunidades de desarrollo de software**

Entidad Financiadora: *Ministerio de Ciencia e Innovación, TIN2011-28110*

Entidades participantes: *URJC*

Duración desde: *01/01/2012* hasta: *31/12/2014*

Director del Proyecto: *Jesús M. González Barahona*

Número de investigadores participantes: *5*

Presupuesto (euros): *53.361*

---

Título del proyecto: **ALERT, Active support and real-time coordination based on event processing in open source software development**

Entidad Financiadora: *VII Programa Marco Comisión Europea (IST), contract number, FP7-IST-25809*

Entidades participantes: *Forschungszentrum Informatik (DE), Universidad Rey Juan Carlos (ES), Institute of Communication and Computer Systems (GR), KDE (DE), Atos Origin (ES), EBM (FR), Institut Josef Stefan (SI), CIM Group (RS), Corvinno Technology Transfer Center (HU), Linagora (FR)*

Duración desde: *01/10/2010* hasta: *31/03/2013*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *5 (en el grupo de la URJC)*

Presupuesto (euros): *260.000 (URJC), 4.000.000 (total)*

---

Título del proyecto: ***eMadrid: Investigación y Desarrollo de Tecnologías para el e-learning en la Comunidad de Madrid***

Entidad Financiadora: *Programas de I+D en Tecnologías 2009 (Comunidad de Madrid)*

Entidades participantes: *UC3M, UPM, UAM, UNED y URJC*

Duración desde: *01/01/2010* hasta: *31/12/2013*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *7 (en el grupo de la URJC)*

Presupuesto (euros): *8.740 (URJC), alrededor 380.000 (total)*

---

Título del proyecto: ***OpenSE: Open Educational Framework for Computer Science Software Engineering***

Entidad Financiadora: *Comisión Europea (Lifelong Learning Programme), 503641-LLP-1-2009-1-PT-ERASMUS-ECUE*

Entidades participantes: *Sociedade Portuguesa de Inovação (PT), Free Knowledge Institute (NL), Tampere University of Technology (FI), Open University (UK), Ecole pour l'Informatique et les Techniques Avancees (FR), Aristotle University of Thessaloniki (GR), University of Maastricht (NL), University of Oxford (UK), linkSpace Management Services (DE), Universidad Rey Juan Carlos (ES)*

Duración desde: *01/10/2009* hasta: *30/11/2011*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *4 (en el grupo de la URJC)*

Presupuesto (euros): *45.675 (URJC)*

---

Título del proyecto: ***Morfeo Formación y Certificación en Tecnologías Libres***

Entidad Financiadora: *Ministerio de Industria, Turismo y Comercio, TSI-010200-2008-310 (AVANZA)*

Entidades participantes: *Telefonica I+D, UPM, Educaterra, ESI, ITI, CESLCAM, Ikusnet, URJC*

Duración desde: *Enero 2008* hasta: *Diciembre 2009*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *3 (en el grupo de la URJC)*

Presupuesto (euros): *70.000 (URJC), 525.000 (total)*

---

Título del proyecto: ***Morfeo Open Marketplace***

Entidad Financiadora: *Ministerio de Industria, Turismo y Comercio, TSI-020302-2008-37 (AVANZA)*

Entidades participantes: *Telefonica I+D, Ikusnet, Fotón, ITI, Push the Button, URJC*

Duración desde: *01/01/2008* hasta: *31/03/2009*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *3 (en el grupo de la URJC)*

Presupuesto (euros): *47.000 (URJC), 265.000 (total)*

---

Título del proyecto: ***Acción complementaria del proyecto FLOSSInclude***

Entidad Financiadora: *Ministerio de Ciencia e Innovación, TIN2008-02260-E*

Entidades participantes: *Universidad Rey Juan Carlos*

Duración desde: *18/12/2008* hasta: *17/12/2010*

Director del Proyecto: *Jesús M. González Barahona*

Número de investigadores participantes: *4*

Presupuesto (euros): *4.000*

---

Título del proyecto: **FLOSSInclude. Free/Libre/Open Source Software: International cooperation development roadmap**

Entidad Financiadora: VII Programa Marco Comisión Europea (IST), contract number, FP7-IST-1-216214

Entidades participantes: Uni. Maastricht (NL), URJC, FUNDECYT (ES), Canonical (UK), Fundación Vía Libre (AR), UWC (ZA), AITI-KACE (GH), IT for Change (IN), Sarai (IN), Open Institute (KH), Tsinghua Univ (CN)

Duración desde: 01/02/2008 hasta: 31/01/2010

Director del Proyecto: Jesús M. González Barahona (coordinador del subproyecto en la URJC)

Número de investigadores participantes: 4 (en el grupo de la URJC)

Presupuesto (euros): 72.000 (URJC), 1.090.000 (total)

---

Título del proyecto: **VULCANO: Promoción del desarrollo del SW Libre en un entorno de calidad y confianza adaptando metodologías, procesos, modelos de negocio y últimas tecnologías**

Entidad Financiadora: Ministerio de Ciencia y Tecnología, FIT-350503-2007-7 (AVANZA)

Entidades participantes: Atos Origin, ISC, Telefonica I+D, Germinus, Andago, Yaco, CTIC, UOC, ESI, ITI, URJC, UPM

Duración desde: 01/01/2008 hasta: 31/03/2009

Director del Proyecto: Jesús M. González Barahona (coordinador del grupo de la URJC)

Número de investigadores participantes: 4 (en el grupo de la URJC)

Presupuesto (euros): 80.000 (URJC), 2.654.096 (total)

---

Título del proyecto: **Gestion 2.0: Sistema Integral de Gestión Corporativa**

Entidad Financiadora: Ministerio de Ciencia y Tecnología, FIT-350401-2007-7 (PROFIT, proyecto tractor)

Entidades participantes: Ikusnet, CEIN, Plastiasite, Germinus, ESI, ITI, UPC, URJC, etc.

Duración desde: Enero 2007 hasta: Diciembre 2008

Director del Proyecto: Jesús M. González Barahona (coordinador del grupo de la URJC)

Número de investigadores participantes: 3 (en el grupo de la URJC)

Presupuesto (euros): 58.000 (URJC), 1.366.863 (total)

---

Título del proyecto: **SobreSalto, Estudio de comunidades de desarrollo de software libre**

Entidad Financiadora: Ministerio de Educación y Ciencia, TIN2007-66172

Entidades participantes: URJC

Duración desde: Octubre 2007 hasta: Septiembre 2010

Director del Proyecto: Jesús M. González Barahona

Número de investigadores participantes: 9

Presupuesto (euros): 48.400

---

Título del proyecto: **BEinGRID**

Entidad Financiadora: VI Programa Marco Comisión Europea (IST), contract number, FP6-IST-5-034702

Entidades participantes: Atos Origin (ES), CETIC (BE), British Telecom (BT), Telefonica I+D (ES), URJC (ES), etc.

Duración desde: Junio 2007 hasta: Julio 2008 (URJC, proyecto desde 2006 hasta 2010)

Director del Proyecto: Pedro de las Heras Quirós (coordinador del grupo de la URJC)

Número de investigadores participantes: 4 (en el grupo de la URJC)

Presupuesto (euros): 55.000 (URJC), 24.800.000 (total)

---

Título del proyecto: **Morfeo: Comunidad Morfeo, acciones complementarias**

Entidad Financiadora: Ministerio de Industria, Turismo y Comercio, FIT-350400-2007-5 (PROFIT)

Entidades participantes: Telefonica I+D, URJC, UPM y otras

Duración desde: Enero 2007 hasta: Diciembre 2007

Director del Proyecto: Andrés Martínez (coordinador del grupo de la URJC)

Número de investigadores participantes: 4 (en el grupo de la URJC)

Presupuesto (euros): 19.700 (URJC), 119.791 (total)

---

Título del proyecto: **FLOSS Research: Acción integradas de investigación científica y tecnológica entre España y Austria**

Entidad Financiadora: *Ministerio de Educación y Ciencia, HU2006-0005*

Entidades participantes: *URJC (ES), Vienna University of Economics and Business Administration (AT)*

Duración desde: *01/01/2007* hasta: *31/12/2008*

Director del Proyecto: *Jesús M. González Barahona*

Número de investigadores participantes: *3 (en el grupo de la URJC)*

Presupuesto (euros): *10.400 (URJC)*

---

Título del proyecto: **QUALIPSO, Quality Platform for Open Source Software**

Entidad Financiadora: *VI Programa Marco Comisión Europea (IST), contract number, FP6-IST-5-034763*

Entidades participantes: *Ingegneria (IT), Telefónica I+D (ES), Atos Origin (ES), IRIA (FR), Mandriva (FR), Siemens (DE), Thales (FR), Uni. Bozen (IT), Fraunhofer FOKUS (DE), URJC (ES), etc.*

Duración desde: *01/11/2006* hasta: *31/10/2010*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *5 (en el grupo de la URJC)*

Presupuesto (euros): *498.036 (URJC), 16.869.353 (total)*

---

Título del proyecto: **QUALOSS, Quality of Open Source Software**

Entidad Financiadora: *VI Programa Marco Comisión Europea (IST), contract number, FP6-IST-5-033547*

Entidades participantes: *CETIC (BE), Uni. Notre Dame de la Paix FUNDP (BE), URJC (ES), Fraunhofer IESE (DE), ZEA Partners (BE), Uni. Maastricht (NL), AdaCore (FR), PEPITe (BE)*

Duración desde: *01/09/2006* hasta: *28/02/2009*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *7 (en el grupo de la URJC)*

Presupuesto (euros): *278.000 (URJC), 2.954.048 (total)*

---

Título del proyecto: **FLOSSMETRICS, Free/Libre/Open Source Software Metrics and Benchmarking Study**

Entidad Financiadora: *VI Programa Marco Comisión Europea (IST), contract number, FP6-IST-5-033982*

Entidades participantes: *URJC (ES), Uni. Maastricht (NL), Uni. Wirtschaftsuniversitaet Wien (AT), Aristotle University of Thessaloniki (GR), ZEA Partners (BE), Philips Medical Systems (NL), Conecta s.r.l. (IT)*

Duración desde: *01/09/2006* hasta: *31/10/2009*

Director del Proyecto: *Jesús M. González Barahona (coordinador de todo el proyecto)*

Número de investigadores participantes: *6 (en el grupo de la URJC)*

Presupuesto (euros): *178.700 (URJC), 585.180 (total)*

---

Título del proyecto: **VULCANO: Forja de proyectos de software de calidad**

Entidad Financiadora: *Ministerio de Ciencia y Tecnología, FIT-340503-2006-3 (AVANZA)*

Entidades participantes: *Atos Origin, Telefonica I+D, URJC, UPM, Instituto Tecnológico de Aragón, CTIC, ESI, Andago, Germinus, Yaco*

Duración desde: *Enero 2006* hasta: *Diciembre 2006*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *4 (en el grupo de la URJC)*

Presupuesto (euros): *74.000 (URJC), 1.401.243 (total)*

---

Título del proyecto: **Morfeo: Comunidad de Software Libre para el desarrollo de componentes software y plataformas**

Entidad Financiadora: *Ministerio de Industria, Turismo y Comercio, FIT-350400-2006-20 (PROFIT)*

Entidades participantes: *Telefonica I+D, URJC, UPM y otras*

Duración desde: *Enero 2006* hasta: *Diciembre 2006*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *2 (en el grupo de la URJC)*

Presupuesto (euros): *23.400 (URJC), 855.524 (total)*

---

Título del proyecto: **MobileScript**

Entidad Financiadora: *Ministerio de Ciencia y Tecnología, FIT-350400-2006-43 (PROFIT)*

Entidades participantes: *Vodafone, URJC*

Duración desde: *Enero 2006* hasta: *Diciembre 2006*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *3 (en el grupo de la URJC)*

Presupuesto (euros): *51.200 (URJC), 183.000 (total)*

---

Título del proyecto: **Investigación de la situación del software libre y de las políticas para su desarrollo en España**

Entidad Financiadora: *Ministerio de Ciencia y Tecnología, FIT-350400-2006-12 (PROFIT)*

Entidades participantes: *Andago, URJC, UPM*

Duración desde: *Enero 2006* hasta: *Diciembre 2006*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *3 (en el grupo de la URJC)*

Presupuesto (euros): *28.800 (URJC), 129.000 (total)*

---

Título del proyecto: **CompatibleLinux: Sistema de compatibilidad de dispositivos en Linux**

Entidad Financiadora: *Ministerio de Ciencia y Tecnología, FIT-150110-2005-5 (PROFIT)*

Entidades participantes: *IT Deusto, Lambdaux, URJC*

Duración desde: *01/01/2005* hasta: *31/12/2005*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *2 (en el grupo de la URJC)*

Presupuesto (euros): *27.000 (URJC), 300.000 (total)*

---

Título del proyecto: **Morfeo: Comunidad de Software Libre para el desarrollo de componentes software y plataformas**

Entidad Financiadora: *Ministerio de Ciencia y Tecnología, FIT-350110-2005-0073 (PROFIT)*

Entidades participantes: *Telefonica I+D, URJC, UPM y otras*

Duración desde: *Enero 2005* hasta: *Diciembre 2005*

Director del Proyecto: *Jesús M. González Barahona (coordinador del grupo de la URJC)*

Número de investigadores participantes: *5 (en el grupo de la URJC)*

Presupuesto (euros): *60.000 (URJC), 800.000 (total)*

---

Título del proyecto: **Economic Organization, Performance and Viability of Open Source/Free Software Development**

Entidad Financiadora: *National Science Foundation, EE.UU. (via subcontrato con Universidad de Maastricht)*

Entidades participantes: *International Institute of Infonomics - Universidad de Maastricht, Stanford Institute for Economic Policy Research - Stanford University*

Duración desde: *Enero 2005* hasta: *Diciembre 2005*

Director del Proyecto: *Paul David (Stanford), Gregorio Robles (coordinador en la URJC)*

Número de investigadores participantes: *3 (dentro del subproyecto realizado en la URJC)*

Presupuesto (euros): *5,000 (URJC)*

---

Título del proyecto: **FLOSSWorld: Free Libre and Open Source Software - Worldwide Impact Study**

Entidad Financiadora: VI Programa Marco Comisión Europea (IST), contract number 015722

Entidades participantes: URJC, University of Maastricht / MERIT (NL), University of Oxford (UK), FUNDECYT (ES), etc. (a total of 17 institutions of 11 countries)

Duración desde: 22/04/2005 hasta: 21/04/2007

Director del Proyecto: Rishab Aiyer Ghosh (University of Maastricht), Jesús M. González Barahona (coordinador en la URJC)

Número de investigadores participantes: 2 (dentro del subproyecto realizado en la URJC)

Presupuesto (euros): 31.650 (URJC), 709.858 (total)

---

Título del proyecto: **Nuevas técnicas para el análisis y planificación de redes y sistemas distribuidos**

Entidad Financiadora: Universidad Rey Juan Carlos (PPR-2004-42)

Entidades participantes: URJC

Duración desde: Enero 2005 hasta: Diciembre 2005

Director del Proyecto: Luis López Fernández

Número de investigadores participantes: 10

Presupuesto (euros): 6.000

---

Título del proyecto: **DESOBRE: Desarrollo de Software Libre (análisis y herramientas)**

Entidad Financiadora: Ministerio de Educación y Ciencia (TIN2004-07296)

Entidades participantes: URJC

Duración desde: 12/12/2004 hasta: 11/12/2005

Director del Proyecto: Jesús M. González-Barahona

Número de investigadores participantes: 10

Presupuesto (euros): 13.650

---

Título del proyecto: **CALIBRE: Coordinated Action for LIBRE software**

Entidad Financiadora: VI Programa Marco Unión Europea (IST-FP6-004337)

Entidades participantes: URJC (ES), University of Limerick (IE), Poznan University of Technology (PL), IMRI - University of Paris Dauphine (FR), University College Cork (IE), University of Lincoln (UK), University of Maastricht / MERIT (NL), University of Skövde (SW) y algunos socios industriales

Duración desde: Junio 2004 hasta: Mayo 2006

Director del Proyecto: Brian Fitzgerald (University of Limerick), Jesús M. González Barahona (coordinador en la URJC)

Número de investigadores participantes: 8 (dentro del subproyecto realizado en la URJC)

Presupuesto (euros): 129.568 (URJC), 1.651.233 (total)

---

Título del proyecto: **Edukalibre**

Entidad Financiadora: Comisión Europea (Lifelong Learning Programme), 110330-CP-1-2003-1-ES-MINERVA-M)

Entidades participantes: URJC (ES), Universidade do Porto (PT), Leeds University (UK), Karlsruhe Universitaet (DE), UTIA (CZ), Universidade di Bolzano (CH)

Duración desde: 01/12/2003 hasta: 01/01/2006

Director del Proyecto: Jesús M. González Barahona (todo el proyecto)

Número de investigadores participantes: 16 (en todo el proyecto)

Presupuesto (euros): 159.130 (subvención URJC), 658.278 (total)

---

Título del proyecto: **eMAMUTi: Extensión de MAMUT para redes inalámbricas**

Entidad Financiadora: CICYT (TIC2001-0447)

Entidades participantes: URJC

Duración desde: Enero 2002 hasta: Enero 2005

Director del Proyecto: Jesús M. González Barahona

Número de investigadores participantes: 7

Presupuesto (euros): 74.092

---



Título del proyecto: **TIER: Transmisión Inalámbrica En Robots móviles**  
 Entidad Financiadora: *Universidad Rey Juan Carlos (PIGE-02-06)*  
 Entidades participantes: *URJC*  
 Duración desde: *2002* hasta: *2002*  
 Director del Proyecto: *Vicente Matellán Olivera*  
 Número de investigadores participantes: *6*  
 Presupuesto (euros):

---

Título del proyecto: **RAIDER: Redes Autoconfigurables Inalámbricas Desplegando Robots Móviles**  
 Entidad Financiadora: *Ministerio de Ciencia y Tecnología (FIT-07000-2001-118)*  
 Entidades participantes: *URJC*  
 Duración desde: *Noviembre 2001* hasta: *Diciembre 2002*  
 Director del Proyecto: *Vicente Matellán Olivera*  
 Número de investigadores participantes: *6*  
 Presupuesto (euros):

---

Título del proyecto: **eMAMUT: Extensión de MAMUT para redes inalámbricas**  
 Entidad Financiadora: *Comunidad Autónoma de Madrid (07T/0004/2002)*  
 Entidades participantes: *URJC*  
 Duración desde: *Enero 2002* hasta: *Enero 2003*  
 Director del Proyecto: *Vicente Matellán Olivera*  
 Número de investigadores participantes: *6*  
 Presupuesto (euros):

---

Título del proyecto: **MIND**  
 Entidad Financiadora: *V Programa Marco Unión Europea (IST)*  
 Entidades participantes: *UPM, Ericsson, Nokia, Deutsche Telekom, NTT, Agora Technologies, etc.*  
 Duración desde: *Julio 2001* hasta: *Diciembre 2002*  
 Director del Proyecto: *Pedro de las Heras Quirós, José Centeno González*  
 Número de investigadores participantes: *7 (en el subproyecto de la URJC)*  
 Presupuesto (euros):

---

Título del proyecto: **VisorBase: Video Sensor Object Request Broker Architecture for Distributed Services**  
 Entidad Financiadora: *V Programa Marco Unión Europea (IST-1999-10808)*  
 Entidades participantes: *Visual Tools SA (coordinador), URJC, INRIA, UPV y otros*  
 Duración desde: *Febrero 2000* hasta: *Febrero 2002*  
 Director del Proyecto: *Enrique Cabello Pardos (de subproyecto en la URJC)*  
 Número de investigadores participantes: *28 (en todo el proyecto)*  
 Presupuesto (euros):

---

Título del proyecto: **AplicAda: Replicación distribuida en Ada 95**  
 Entidad Financiadora: *CICYT (TIC98-1032-C03-03)*  
 Entidades participantes: *UPM, URJC, ULPGC*  
 Duración desde: *septiembre 1998* hasta: *septiembre 2001*  
 Director del Proyecto: *Jesús M. González Barahona (subproyecto coordinado)*  
 Número de investigadores participantes: *21 (en todo el proyecto)*  
 Presupuesto (euros): *31.500*

---

Título del proyecto: **Mamut: Módulo Autoconfigurable para transmisión Multimedia**  
 Entidad Financiadora: *Comunidad Autónoma de Madrid (07T/0005/1998)*  
 Entidades participantes: *UCAR*  
 Duración desde: *noviembre 1998* hasta: *octubre 1999*  
 Director del Proyecto: *Jesús M. González Barahona*  
 Número de investigadores participantes: *5*  
 Presupuesto (euros): *22.500*

---

Título del proyecto: *Técnicas de Tolerancia a fallos hardware y software en sistemas distribuidos y de tiempo real*

Entidad Financiadora: *CICYT (TIC94-0162-C02-02)*

Entidades participantes: *UPM, UCAR*

Duración desde: *mayo 1994* hasta: *abril 1997*

Director del Proyecto: *Jesús M. González Barahona (subproyecto coordinado)*

Número de investigadores participantes: *10 (en todo el proyecto)*

Presupuesto (euros):

---

Título del proyecto: *Desarrollo de funciones y servicios de seguridad según las normativas X.400 y X.509 y su integración en entornos EDI*

Entidad Financiadora: *CDTI (programa PASO)*

Entidades participantes: *Penta3, Centro de Cálculo de Sabadell, CSIC, UPM, UCAR,*

Duración desde: *1993* hasta: *1995*

Director del Proyecto: *Arturo Ribagorda Garnacho (subproyecto coordinado)*

Número de investigadores participantes: *17 (en todo el proyecto)*

Presupuesto (euros):

---

## Capítulo 22

# Participación en contratos de I+D de especial relevancia con Empresas y/o Administraciones

---

Título del contrato: *Analíticas de desarrollo de software basadas en información disponible en repositorios de desarrollo*

Empresa/administración financiadora: *Bitergium*

Duración desde: 01/06/2014 hasta: 31/10/2018

Investigador responsable: *Jesús M. González-Barahona*

Presupuesto (euros): *42.000*

---

Título del contrato: *Transferencia de tecnología: Herramientas de análisis de repositorios de desarrollo de software*

Empresa/administración financiadora: *Bitergia*

Duración desde: 03/10/2012 hasta: 03/10/2015

Investigador responsable: *Jesús M. González-Barahona*

Presupuesto (euros): *2.000*

---

Título del contrato: *Quantitative Analysis of FLOSS Communities*

Empresa/administración financiadora: *Samsung Electronics*

Duración desde: 01/04/2012 hasta: 31/12/2012

Investigador responsable: *Jesús M. González-Barahona*

Presupuesto (euros): *55.000*

---

Título del contrato: *Development of a new feature for software data mining tool*

Empresa/administración financiadora: *Rackspace*

Duración desde: 13/02/2012 hasta: 31/03/2012

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *2500*

---

Título del contrato: *Asistencia CAU soporte forja virtual*

Empresa/administración financiadora: *CENATIC*

Duración desde: 01/02/2012 hasta: 31/05/2012

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *12.000*

---

Título del contrato: ***Soporte forja FI-WARE***

Empresa/administración financiadora: *Telefónica I+D (subcontratado de la Comisión Europea, FP7)*

Duración desde: 01/07/2011 hasta: 30/06/2012

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *25.000*

---

Título del contrato: ***Asesoría para el Centro de innovación a través del software libre Madrid on Rails del Ayuntamiento de Madrid***

Empresa/administración financiadora: *BABEL Sistemas de Información*

Duración desde: 29/04/2011 hasta: 31/12/2011

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *6,000*

---

Título del contrato: ***Go-my Life. Going social: my social life***

Empresa/administración financiadora: *Andago*

Duración desde: 01/12/2010 hasta: 30/11/2012

Investigador responsable: *Pedro de las Heras Quirós*

Presupuesto (euros): *90.000*

---

Título del contrato: ***Análisis comparativo de los códigos fuente de las aplicaciones gisEIEL y gvSIG-EIEL***

Empresa/administración financiadora: *Diputación de La Coruña*

Duración desde: 07/06/2011 hasta: 06/07/2011

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *3.500*

---

Título del contrato: ***Máster en economía digital e industrias creativas: módulos sobre propiedad industrial, intelectual y nuevas tecnologías***

Empresa/administración financiadora: *Fundacion EOI*

Duración desde: 01/11/2010 hasta: 30/01/2011

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *10.932*

---

Título del contrato: ***Seminario de software libre para PYMES***

Empresa/administración financiadora: *Fundacion EOI*

Duración desde: 04/10/2010 hasta: 19/10/2010

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *4.380*

---

Título del contrato: ***OSOR, Open Source Observatory and Repository (FC-II)***

Empresa/administración financiadora: *Comisión Europea, IDABC (consorcio ganador de concurso público, formado por UNISYS (BE), Eurocities (BE), MERIT (NL) y URJC (ES))*

Duración desde: 23/08/2010 hasta: 01/03/2012

Investigador responsable: *Jesús M. González Barahona (del subgrupo de la URJC)*

Presupuesto (euros): *249.000 (URJC), 1.500.000 (total)*

---

Título del contrato: ***Asesoramiento en Software Libre para Planificación Estratégica***

Empresa/administración financiadora: *Fundación EOI*

Duración desde: 15/07/2010 hasta: 31/07/2010

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *847*

---

Título del contrato: ***Study of the effects of the use of the FlaggedRevision Extension on the German language Wikipedia***

Empresa/administración financiadora: *Wikimedia Deutschland*

Duración desde: 18/01/2010 hasta: 07/05/2010

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *4.725*

---

Título del contrato: ***Estudio nacional de necesidades formativas y Plan de formación en software de fuentes abiertas***

Empresa/administración financiadora: *Novasoft*

Duración desde: 27/10/2009 hasta: 03/02/2010

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *15.500*

---

Título del contrato: ***Consulting and support Mozilla-Europe activities***

Empresa/administración financiadora: *Mozilla Europe*

Duración desde: 01/10/2009 hasta: 31/12/2009

Investigador responsable: *Gregorio Robles Martínez*

Presupuesto (euros): *6.000*

---

Título del contrato: ***Textmining y tratamiento de texto libre en encuestas***

Empresa/administración financiadora: *Telefónica Investigación y Desarrollo*

Duración desde: 22/06/2009 hasta: 22/12/2009

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *4.400*

---

Título del contrato: ***Estudio de viabilidad de programas de certificación de software de fuentes abiertas***

Empresa/administración financiadora: *CENATIC: Centro Nacional de Referencia de Aplicación de las Tecnologías de la Información y la Comunicación (TIC) basadas en fuentes abiertas (consorcio ganador de concurso público, formado por ESI, Ikusnet y URJC)*

Duración desde: 19/05/2009 hasta: 18/02/2010

Investigador responsable: *Jesús M. González-Barahona*

Presupuesto (euros): *33.000*

---

Título del contrato: ***Software de Fuentes Abiertas en Universidades y Centros de I+D españoles***

Empresa/administración financiadora: *Fundación CENATIC: Centro Nacional de Referencia de Aplicación de las Tecnologías de la Información y la Comunicación (TIC) basadas en fuentes abiertas*

Duración desde: 22/04/2009 hasta: 21/10/2009

Investigador responsable: *Jesús M. González-Barahona*

Presupuesto (euros): *72.763*

---

Título del contrato: ***Participación en módulos sobre propiedad industrial, intelectual y nuevas tecnologías***

Empresa/administración financiadora: *Fundacion EOI*

Duración desde: 01/02/2009 hasta: 31/12/2009

Investigador responsable: *Gregorio Robles Martínez*

Presupuesto (euros): *4.586*

---

Título del contrato: ***Colaboración en el desarrollo de las pruebas de concepto de EzWeb***

Empresa/administración financiadora: *Telefónica Investigación y Desarrollo*

Duración desde: 15/01/2009 hasta: 14/10/2009

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *5.375*

---

Título del contrato: ***Colaboración en el desarrollo del catálogo de la versión beta de la plataforma EzWeb***

Empresa/administración financiadora: *Telefónica Investigación y Desarrollo*

Duración desde: 15/01/2009 hasta: 14/10/2009

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *27.343*

---

Título del contrato: ***Creación del software de adaptación gráfica de una forja con la imagen corporativa de CENATIC***

Empresa/administración financiadora: *CENATIC: Centro Nacional de Referencia de Aplicación de las Tecnologías de la Información y la Comunicación (TIC) basadas en fuentes abiertas*

Duración desde: 27/10/2008 hasta: 26/01/2009

Investigador responsable: *Jesús M. González-Barahona*

Presupuesto (euros): *18.041*

---

Título del contrato: ***Estudio de la integración entre herramientas en fuente abierta para la gestión de comunidades virtuales***

Empresa/administración financiadora: *CENATIC: Centro Nacional de Referencia de Aplicación de las Tecnologías de la Información y la Comunicación (TIC) basadas en fuentes abiertas*

Duración desde: 27/10/2008 hasta: 26/01/2009

Investigador responsable: *Jesús M. González-Barahona*

Presupuesto (euros): *1.000*

---

Título del contrato: ***Creación de comunidades para los candidatos más representativos***

Empresa/administración financiadora: *Infojobs S.A.*

Duración desde: 18/02/2008 hasta: 31/08/2008

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *11.410*

---

Título del contrato: ***Creación de widgets para redes sociales***

Empresa/administración financiadora: *Infojobs S.A.*

Duración desde: 18/02/2008 hasta: 31/08/2008

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *10.586*

---

Título del contrato: ***OSOR, Open Source Observatory and Repository (FC-I)***

Empresa/administración financiadora: *Comisión Europea, IDABC (consorcio ganador de concurso público, formado por UNISYS (BE), Eurocities (BE), MERIT (NL) y URJC (ES))*

Duración desde: 25/01/2008 hasta: 19/09/2010

Investigador responsable: *Jesús M. González Barahona (del subgrupo de la URJC)*

Presupuesto (euros): *771.000 (URJC), 3.500.000 (total)*

---

Título del contrato: ***Situación del software libre en las administraciones públicas españolas***

Empresa/administración financiadora: *CENATIC (Centro Nacional de Referencia de Aplicación de las Tecnologías de la Información y la Comunicación (consorcio ganador de concurso público, formado por URJC y Telefónica I+D))*

Duración desde: 03/10/2007 hasta: 03/02/2008

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *29.000*

---

Título del contrato: ***FLOSSImpact, The impact of Free/Libre/Open Source Software on innovation and competitiveness of the European Union***

Empresa/administración financiadora: *Comisión Europea, INFISO, (consorcio ganador de concurso público, formado por MERIT (NL), Uni Limerick (IE), URJC, Soupinspace (FR))*

Duración desde: 01/06/2006 hasta: 30/11/2006

Investigador responsable: *Jesús M. González Barahona (del subgrupo de la URJC)*

Presupuesto (euros): *50.000 (URJC), 450.000 (total)*

---

Título del contrato: ***Proyecto Tree***

Empresa/administración financiadora: *Vodafone Research*

Duración desde: 01/05/2006 hasta: 31/12/2006

Investigador responsable: *Jesús M. González Barahona*

Presupuesto (euros): *55.000*

---

Título del contrato: **Proyecto Morfeo (continuación)**  
 Empresa/administración financiadora: *Telefónica Investigación y Desarrollo*  
 Duración desde: 01/12/2005 hasta: 30/06/2006  
 Investigador responsable: *Andrés L. Martínez Ortíz*  
 Presupuesto (euros): *50.000*

---

Título del contrato: **Proyecto Morfeo**  
 Empresa/administración financiadora: *Telefónica Investigación y Desarrollo*  
 Duración desde: 01/12/2004 hasta: 30/06/2005  
 Investigador responsable: *Andrés L. Martínez Ortíz*  
 Presupuesto (euros): *35.000*

---

Título del contrato: **Valdelinux: Acuerdo de colaboración para el apoyo a las actuaciones con grupos de investigación de la Universidad Rey Juan Carlos**  
 Empresa/administración financiadora: *Ayuntamiento de Valdemoro*  
 Duración desde: 01/06/2004 hasta: 31/05/2006  
 Investigador responsable: *Jesús M. González Barahona*  
 Presupuesto (euros): *3.000*

---

Título del contrato: **ORK: Open Ravenscar Kernel**  
 Empresa/administración financiadora: *ESTEC (European Space Agency), ESTEC 13863/99/NL/MV, grupo formado por URJC, UPM, Construcciones Aeronáuticas, York University (UK)*  
 Duración desde: 01/10/1999 hasta: 31/07/2000  
 Investigador responsable: *Jesús M. González Barahona (suproyecto de soporte a la depuración)*  
 Presupuesto (euros):

---

Título del contrato: **Transmisión de Información Multimedia**  
 Empresa/administración financiadora: *Visual Tools*  
 Duración desde: 01/05/2001 hasta: 30/04/2002  
 Investigador responsable: *José Centeno González*  
 Presupuesto (euros):

---

Título del contrato: **Computación Móvil y Ubicua**  
 Empresa/administración financiadora: *Virtual Office*  
 Duración desde: 01/06/2001 hasta: 31/07/2002  
 Investigador responsable: *Pedro de las Heras Quirós*  
 Presupuesto (euros):

---





## Capítulo 23

# Publicaciones o Documentos Científico-Técnicos

(Clave: L = libro completo, CL = capítulo de libro, A = artículo, R = “review”, E = editor, S = documento científico técnico restringido)

---

Autores: *Héctor J. Macho, Gregorio Robles, Jesús M. González-Barahona*  
Título: ***Evaluation of FLOSS by Analyzing Its Software Evolution: An Example using the Moodle Platform***  
Ref. revista/libro: *Journal of Information Technology Research*  
Clave: A, Volumen: *8(1)*, Páginas: *62-81*,  
Fecha: *2015*  
Editorial: *IGI Global*  
Lugar de publicación: DOI: *10.4018/JITR.2015010105*

---

Autores: *Jose Teixeira, Gregorio Robles, Jesús M. González-Barahona*  
Título: ***Lessons learned from applying social network analysis on an industrial Free/Libre/Open Source Software ecosystem***  
Ref. revista/libro: *Journal of Internet Services and Applications*  
Clave: A, Volumen: *6(1)*, Páginas: *14:1-14:27*,  
Fecha: *2015*  
Editorial: *Springer*  
Lugar de publicación: DOI: *10.1186/s13174-015-0028-2*

---

Autores: *Jesus M. Gonzalez-Barahona, Gregorio Robles, Israel Herraiz, Felipe Ortega*  
Título: ***Studying the laws of software evolution in a long-lived FLOSS project***  
Ref. revista/libro: *Journal of Software: Evolution and Process*  
Clave: A, Volumen: *26(7)*, Páginas: *589-612*,  
Fecha: *2014*  
Editorial: *Wiley*  
Lugar de publicación: DOI: *10.1002/smr.1615*

---

Autores: *Jesus M. Gonzalez-Barahona, Daniel Izquierdo-Cortazar, Gregorio Robles, Alvaro del Castillo*  
Título: ***Analyzing Gerrit Code Review Parameters with Bicho***  
Ref. revista/libro: *Electronic Communication of the European Association of Software Science and Technology*  
Clave: *65*, Volumen: *,* Páginas: *2014*,  
Fecha: *European Association of Software Science and Technology*  
Editorial: *ISSN 1863-2122*  
Lugar de publicación:

---

Autores: *Jesús M. González Barahona, Daniel Izquierdo, Steffano Maffulli, Gregorio Robles*  
 Título: ***Understanding How Companies Interact with Free Software Communities***  
 Ref. revista/libro: *IEEE Software*  
 Clave: A, Volumen: 30 (5), Páginas: 38-45,  
 Fecha: 2013  
 Editorial: *IEEE*  
 Lugar de publicación:

---

Autores: *Israel Herraiz, Daniel Rodríguez, Gregorio Robles, Jesus M. Gonzalez-Barahona*  
 Título: ***The evolution of the laws of software evolution: A discussion based on a systematic literature review***  
 Ref. revista/libro: *ACM Computing Surveys*  
 Clave: A, Volumen: 46(2), Páginas: 28:1-28:28,  
 Fecha: 2013  
 Editorial: *ACM*  
 Lugar de publicación:

---

Autores: *Jesús M. González Barahona, Gregorio Robles*  
 Título: ***Trends in Free, Libre, Open Source Software Communities: From Volunteers to Companies***  
 Ref. revista/libro: *it-Information Technology*  
 Clave: A, Volumen: 55 (5), Páginas: 173-180,  
 Fecha: 2013  
 Editorial: *De Gruyter*  
 Lugar de publicación: *ISSN print: 1611-2776, ISSN online: 2196-7032, DOI: 10.1515/itit.2013.1012*

---

Autores: *Jesús M. González Barahona, Gregorio Robles*  
 Título: ***On the reproducibility of empirical software engineering studies based on data retrieved from development repositories***  
 Ref. revista/libro: *Empirical Software Engineering*  
 Clave: A, Volumen: 17(1-2), Páginas: 75-89,  
 Fecha: 2012  
 Editorial: *Springer*  
 Lugar de publicación:

---

Autores: *Antonio Reinoso, Jesus M. Gonzalez-Barahona, Rocio Muñoz-Mansilla, Israel Herraiz*  
 Título: ***Temporal Characterization of the Requests to Wikipedia***  
 Ref. revista/libro: *New Challenges in Distributed Information Filtering and Retrieval*  
 Clave: CL, Volumen: 439, Páginas: 71-89,  
 Fecha: 2012  
 Editorial: *Springer Berlin / Heidelberg*  
 Lugar de publicación: *ISBN: 978-3-642-31545-9, DOI: 10.1007/978-3-642-31546-6\_5*

---

Autores: *Daniel Izquierdo-Cortazar, Andrea Capiluppi, Jesús M. González-Barahona*  
 Título: ***Are Developers Fixing Their Own Bugs?: Tracing Bug-Fixing and Bug-Seeding Committers***  
 Ref. revista/libro: *International Journal of Open Source Software & Processes*  
 Clave: R, Volumen: 3 - 2, Páginas: 23-42,  
 Fecha: 2011  
 Editorial: *IGI Global*  
 Lugar de publicación: *ISSN: 1942-3926, DOI: 10.4018/jossp.2011040102*

---

Autores: *Gregorio Robles, Jesús M. González-Barahona, Daniel Izquierdo-Cortazar*  
 Título: ***Tools and Datasets for Mining Libre Software Repositories***  
 Ref. revista/libro: *Multi-Disciplinary Advancement in Open Source Software and Processes*  
 Clave: CL, Volumen: 1 - 1, Páginas: 24-42,  
 Fecha: 2011  
 Editorial: *IGI Global*  
 Lugar de publicación: *ISBN: 9781609605148, DOI: 10.4018/978-1-60960-513-1.ch002*

---

Autores: *Jesús M. González-Barahona, Daniel Izquierdo-Cortazar, Megan Squire*  
 Título: ***Repositories with Public Data about Software Development***  
 Ref. revista/libro: *International Journal of Open Source Software & Processes*  
 Clave: *R*, Volumen: *2 - 2*, Páginas: *1-13*,  
 Fecha: *2010*  
 Editorial: *IGI Global*  
 Lugar de publicación: *ISSN: 1942-3926, DOI: 10.4018/jossp.2010040101*

---

Autores: *Gregorio Robles, Jesús M. González-Barahona, Daniel Izquierdo-Cortazar, Israel Herraiz*  
 Título: ***Tools for the Study of the Usual Data Sources found in Libre Software Projects***  
 Ref. revista/libro: *International Journal of Open Source Software & Processes*  
 Clave: *R*, Volumen: *1 - 1*, Páginas: *24-45*,  
 Fecha: *Jan-March 2009*  
 Editorial: *IGI Global*  
 Lugar de publicación: *ISSN: 1942-3926, DOI: 10.4018/jossp.2009010102*

---

Autores: *Luis López-Fernández, Gregorio Robles, Jesús M. González-Barahona, Israel Herraiz*  
 Título: ***Applying Social Network Analysis Techniques to Community-Driven Libre Software Projects***  
 Ref. revista/libro: *Integrated Approaches in Information Technology and Web Engineering: Advancing Organizational Knowledge Sharing*  
 Clave: *CL*, Volumen: *3*, Páginas: *28-50*,  
 Fecha: *2009*  
 Editorial: *IGI Global*  
 Lugar de publicación:

---

Autores: *Jesús M. González-Barahona, Gregorio Robles, Martin Michlmayr, Daniel M. German*  
 Título: ***Macro-level Software Evolution: A Case Study of a Large Software Compilation***  
 Ref. revista/libro: *Empirical Software Engineering. ISSN: 1382-3256 (Print) 1573-7616 (Online)*  
 Clave: *A*, Volumen: *Vol. 14, No. 3*, Páginas: *262-285*,  
 Fecha: *June 2009*  
 Editorial: *Springer*  
 Lugar de publicación:

---

Autores: *Jesus M. Gonzalez-Barahona, Gregorio Robles, Roberto Andradas-Izquierdo and Rishab A. Ghosh*  
 Título: ***Geographic Origin of Libre Software Developers***  
 Ref. revista/libro: *Information Economics and Policy*  
 Clave: *A*, Volumen: *Volume 20, Issue 4*, Páginas: *356-363*,  
 Fecha: *2008*  
 Editorial: *Elsevier*  
 Lugar de publicación:

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona and Pedro de-las-Heras-Quirós*  
 Título: ***Experiencia de uso de blogs en e-learning***  
 Ref. revista/libro: *RELADA - Revista Electrónica de ADA-Madrid*  
 Clave: *A*, Volumen: *2*, Páginas: *,*  
 Fecha: *2008*  
 Editorial:  
 Lugar de publicación: *Madrid*

---

Autores: *Jesús M. González-Barahona, Álvaro del Castillo San Félix, José Gato Luis, Felipe Ortega, etc.*

Título: ***Software de fuentes abiertas para el desarrollo de la Administración Pública Española. Una visión global***

Ref. revista/libro:

Clave: *L*, Volumen: , Páginas: *149*,

Fecha: *Septiembre 2008*

Editorial: *CENATIC*

Lugar de publicación: *España*

---

Autores: *Martin Michlmayr, Gregorio Robles, Jesus M. Gonzalez-Barahona*

Título: ***Volunteers in Large Libre Software Projects: A Quantitative Analysis Over Time***

Ref. revista/libro: *Emerging Free and Open Source Software Practices*

Clave: *CL*, Volumen: , Páginas: *1-23*,

Fecha: *2007*

Editorial: *IDEA Publishing, Hershey, PA, EE.UU.*

Lugar de publicación: *ISBN: 978-1-59904-210-7, DOI: 10.4018/978-1-60566-060-8.ch111*

---

Autores: *Luis López, Jesus M. Gonzalez-Barahona, Diego Chaparro, Teófilo Romera, Luis Cañas*

Título: ***Edukalibre: a Tool for Collaborative Creation of Educational Material***

Ref. revista/libro: *International Journal on Continuing Engineering Education and Lifelong Learning*

Clave: *A*, Volumen: *Vol. 17, No. 1*, Páginas: *15-32*,

Fecha: *January 2007*

Editorial: *Inderscience Publishers*

Lugar de publicación: *ISSN online: 1741-5055, ISSN print: 1560-4624, DOI: 10.1504/IJ-CEELL.2007.013227*

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona, Juan Julian Merelo*

Título: ***Beyond Source Code: The Importance of Other Artifacts in Software Development***

Ref. revista/libro: *Journal of Systems and Software*

Clave: *A*, Volumen: *79 (9)*, Páginas: *1233-1248*,

Fecha: *September 2006*

Editorial: *Elsevier*

Lugar de publicación:

---

Autores: *Luis López-Fernández, Gregorio Robles, Jesus M. Gonzalez-Barahona, Israel Herraiz*

Título: ***Applying Social Network Analysis Techniques to Community-Driven Libre Software Projects***

Ref. revista/libro: *International Journal of Information Technology and Web Engineering*

Clave: *A*, Volumen: *vol. 1, num. 3*, Páginas: *27-48*,

Fecha: *July-September 2006*

Editorial: *Idea Group, Hershey, PA, EE.UU.*

Lugar de publicación: *ISSN: 1554-1045, DOI: 10.4018/jitwe.2006070103*

---

Autores: *Stefan Koch, Jesus M. Gonzalez-Barahona*

Título: ***Open Source Software Engineering - The State of Research***

Ref. revista/libro: *First Monday*

Clave: *A*, Volumen: *Special Issue no. 2, Open Source*, Páginas: ,

Fecha: *3rd October 2005*

Editorial: *University of Illinois at Chicago*

Lugar de publicación: *ISSN: 1396-0466, DOI: 10.5210/fm.v0i0.1466*

---

Autores: *Jesus M. Gonzalez-Barahona, Gregorio Robles*  
 Título: ***Libre software in Europe***  
 Ref. revista/libro: *Open Sources 2.0, The Continuing Evolution. Chris DiBona, Mark Stone, Danese Cooper (eds.)*  
 Clave: *CL*, Volumen: , Páginas: ,  
 Fecha: *Octubre 2005*  
 Editorial: *O'Reilly*  
 Lugar de publicación: *ISBN: 0-596-00802-3*

---

Autores: *Juan José Amor, Jesús M. González Barahona, Gregorio Robles, Israel Herraiz*  
 Título: ***Measuring Libre Software using Debian 3.1 (Sarge) as a Case Study: preliminary results***  
 Ref. revista/libro: *Upgrade Magazine*  
 Clave: *A*, Volumen: *VI*, Páginas: *8*,  
 Fecha: *Agosto 2005*  
 Editorial:  
 Lugar de publicación: *ISSN: 1684-5285*

---

Autores: *Jesús M. González Barahona, Joaquín Seoane, Gregorio Robles*  
 Título: ***Introducción a la economía del software libre***  
 Ref. revista/libro: *Libro Blanco del Software Libre en España (II). Miguel A. Segovia Romero y Alberto Abella García ed.*  
 Clave: *CL*, Volumen: , Páginas: *23*,  
 Fecha: *Julio 2005*  
 Editorial: *Junta de Extremadura. Consej. de Infraestructuras y Desarrollo Tecnológico*  
 Lugar de publicación: *Mérida, España*

---

Autores: *Vicente Matellán Olivera, Jesús M. González Barahona, Pedro de las Heras Quirós, Gregorio Robles*  
 Título: ***Sobre Software Libre***  
 Ref. revista/libro: *ISBN: 84-9772-402-X*  
 Clave: *L*, Volumen: , Páginas: *220*,  
 Fecha: *Julio 2004*  
 Editorial: *Servicio de Publicaciones de la URJC. Editorial Dykinson*  
 Lugar de publicación: *Madrid*

---

Autores: *Jesús M. González Barahona, Gregorio Robles, Miguel Ortuño Pérez, Luis Roderó Merino, José Centeno González, Vicente Matellán Olivera, Eva Castro Barbero, Pedro de las Heras Quirós*  
 Título: ***Analyzing the Anatomy of GNU/Linux Distributions: Methodology and Case Studies (Red Hat and Debian)***  
 Ref. revista/libro: *Free/Open Source Software Development, Stefan Kock (ed.)*  
 Clave: *CL*, Volumen: , Páginas: *27-58*,  
 Fecha: *June 2004*  
 Editorial: *Idea Group, Inc., Hershey, PA, EE.UU.*  
 Lugar de publicación: *ISBN: 1-59140-369-3, DOI: 10.4018/978-1-59140-369-2.ch002*

---

Autores: *Jesús M. González Barahona, Joaquín Seoane, Gregorio Robles*  
 Título: ***Introducción al Software Libre***  
 Ref. revista/libro: *ISBN: 84-9788-028-5*  
 Clave: *L*, Volumen: , Páginas: *326*,  
 Fecha: *Noviembre 2003*  
 Editorial: *Planeta UOC*  
 Lugar de publicación: *Barcelona*

---

Autores: *Jesús M. González Barahona, Gregorio Robles Martínez*

Título: **Free Software Engineering: A Field to Explore**

Ref. revista/libro: *Upgrade Magazine*

Clave: A, Volumen: IV, 4, Páginas: 49-54,

Fecha: *August 2003*

Editorial:

Lugar de publicación: *ISSN: 1684-5285*

---

Autores: *Philippe Aigrain, Jesús M. González Barahona*

Título: **Open Knowledge**

Ref. revista/libro: *Upgrade Magazine (también publicado como número de Novática, en español, de Tecnoteca/ALSI en italiano y por APRIL en francés)*

Clave: E, Volumen: IV, 3, Páginas: ,

Fecha: *June 2003*

Editorial:

Lugar de publicación: *ISSN: 1684-5285*

---

Autores: *Jesús M. González Barahona, Pedro de las Heras Quirós*

Título: **Hosting of Libre Software Projects: A Distributed Peer-to-Peer Approach**

Ref. revista/libro: *Future Directions in Distributed Computing. Andre Schiper, Alex Shvartsman, Hakim Weatherspoon, Ben Zhao eds.*

Clave: L, Volumen: 2584, Páginas: 207-211,

Fecha: *2003*

Editorial: *Springer Verlag Lecture Notes in Computer Science series (LNCS 2584)*

Lugar de publicación:

---

Autores: *Jesús M. González Barahona, Miguel A. Ortuño Pérez, Pedro de las Heras Quirós, José Centeno González, Vicente Matellán Olivera*

Título: **Counting potatoes: The size of Debian 2.2**

Ref. revista/libro: *Upgrade Magazine*

Clave: A, Volumen: II, 6, Páginas: 60-66,

Fecha: *December 2001*

Editorial:

Lugar de publicación: *ISSN: 1684-5285*

---

Autores: *Jesús M. González Barahona, Pedro de las Heras Quirós, Joe Amann*

Título: **Free software today**

Ref. revista/libro: *Upgrade Magazine (also published in Informatik/Informatique, November 2001)*

Clave: A, Volumen: II, 6, Páginas: 4-11,

Fecha: *December 2001*

Editorial:

Lugar de publicación:

---

Autores: *Jesús M. González-Barahona, Pedro de las Heras-Quirós, José Centeno-González, Vicente Matellán-Olivera, and Francisco Ballesteros-Cámara*

Título: **Libre Software for Computer Science Classes**

Ref. revista/libro: *IEEE Software*

Clave: A, Volumen: *May/June 2000*, Páginas: 76-79,

Fecha: *May 2000*

Editorial:

Lugar de publicación:

---

Autores: *Juan A. de la Puente, José Francisco Ruiz and Jesús González-Barahona*

Título: **Real-Time Programming with GNAT: Specialised Kernels versus POSIX Threads**

Ref. revista/libro: *ACM Ada Letters*

Clave: A, Volumen: XIX, 2, Páginas: 73-77,

Fecha: *June 1999*

Editorial:

Lugar de publicación:

---

Autores: *Vicente Matellán, Camino Fernández, Pedro de las Heras, José Centeno, Ignacio Aedo, Jesús González y Francisco Ballesteros*

Título: ***Java: un lenguaje de propósito general***

Ref. revista/libro: *Cuadernos de Informática FESI, ISSN 1139-5907*

Clave: *A*, Volumen: *Número 1*, Páginas: *pag. 123-188*,

Fecha: *Abril 1999*

Editorial:

Lugar de publicación:

---

Autores: *Jesús M. González Barahona, Pedro de las Heras Quirós, José Centeno González*

Título: ***Lower Layer: A family of interfaces to transport communication protocols***

Ref. revista/libro: *Ada User Journal*

Clave: *A*, Volumen: *18*, Páginas: ,

Fecha: *Diciembre 1997*

Editorial:

Lugar de publicación: *ISSN: 1381-6551*

---

Autores: *Jesús M. González Barahona, Pedro de las Heras Quirós*

Título: ***Monografía: Software Libre***

Ref. revista/libro: *NOVATICA (Revista de la Asociación de Técnicos de Informática)*

Clave: *E*, Volumen: , Páginas: ,

Fecha: *Marzo-Abril 1997*

Editorial:

Lugar de publicación:

---

Autores: *Fernando Sáez Vacas, G. Alonso García, Jesús M. González Barahona*

Título: ***Complejidad y tecnología de la información***

Ref. revista/libro:

Clave: *L*, Volumen: , Páginas: ,

Fecha: *1992*

Editorial: *Instituto Tecnológico Bull*

Lugar de publicación: *Madrid 1992*

---





## Capítulo 24

# Contribuciones a congresos

---

Autores: *Jesús M. González-Barahona, Paul Sherwood, Gregorio Robles, Daniel Izquierdo-Cortazar*  
Título: ***Technical Lag in Software Compilations: Measuring How Outdated a Software Deployment Is***  
Tipo de participación: *Ponencia*  
Congreso: *13th IFIP WG 2.13 International Conference, OSS 2017*  
Publicación: *In proceedings (182–192)*  
Lugar de celebración: *Buenos Aires, Argentina*  
Fecha: *May 22–23, 2017*  
DOI [10.1007/978-3-319-57735-7\\_17](https://doi.org/10.1007/978-3-319-57735-7_17)

---

Autores: *Daniel Izquierdo-Cortazar, Nelson Sekitoleko, Jesús M. González-Barahona, Lars Kurth*  
Título: ***Using Metrics to Track Code Review Performance***  
Tipo de participación: *Ponencia*  
Congreso: *21st International Conference on Evaluation and Assessment in Software Engineering, EASE 2017*  
Publicación: *In proceedings (214–223)*  
Lugar de celebración: *Karlskrona, Sweden*  
Fecha: *June 15–16, 2017*  
[10.1145/3084226.3084247](https://doi.org/10.1145/3084226.3084247)

---

Autores: *Daniel Izquierdo-Cortazar, Lars Kurth, Jesús M. González-Barahona, Santiago Dueñas, Nelson Sekitoleko*  
Título: ***Characterization of the Xen project code review process: an experience report***  
Tipo de participación: *Ponencia*  
Congreso: *International Working Conference on Mining Software Repositories, MSR 2016*  
Publicación: *In proceedings (386–390)*  
Lugar de celebración: *Austin, TX, USA*  
Fecha: *May 14–15, 2016*  
DOI [10.1109/MSR.2015.68](https://doi.org/10.1109/MSR.2015.68)

---

Autores: *Gregorio Robles, Laura Arjona Reina, Jesús M. González-Barahona, Santiago Dueñas Domínguez*  
Título: ***Women in Free/Libre/Open Source Software: The Situation in the 2010s***  
Tipo de participación: *Ponencia*  
Congreso: *12th IFIP WG 2.13 International Conference on Open Source Systems, OSS 2016*  
Publicación: *In proceedings (163–173)*  
Lugar de celebración: *Gothenburg, Sweden*  
Fecha: *May 30 – Jun 2, 2016*  
[10.1007/978-3-319-39225-7\\_13](https://doi.org/10.1007/978-3-319-39225-7_13)

---

Autores: *Gema Rodríguez-Pérez, Jesús M. González-Barahona, Gregorio Robles, Dorealda Dalipaj, Nelson Sekitoleko*

Título: ***BugTracking: A Tool to Assist in the Identification of Bug Reports***

Tipo de participación: *Ponencia*

Congreso: *12th IFIP WG 2.13 International Conference on Open Source Systems, OSS 2016*

Publicación: *In proceedings (192-198)*

Lugar de celebración: *Gothenburg, Sweden*

Fecha: *May 30 – Jun 2, 2016*

DOI [10.1007/978-3-319-39225-7\\_16](https://doi.org/10.1007/978-3-319-39225-7_16)

---

Autores: *Dorealda Dalipaj, Jesús M. González-Barahona, Daniel Izquierdo-Cortazar*

Título: ***Software Engineering Artifact in Software Development Process - Linkage Between Issues and Code Review Processes***

Tipo de participación: *Ponencia*

Congreso: *15th International Conference on Intelligent Software Methodologies, Tools and Techniques, SoMeT 2016*

Publicación: *In proceedings (115-122)*

Lugar de celebración: *Larnaca, Cyprus*

Fecha: *Sep 12–14, 2016*

DOI [10.3233/978-1-61499-674-3-115](https://doi.org/10.3233/978-1-61499-674-3-115)

---

Autores: *Jesús M. González-Barahona, Gregorio Robles, Daniel Izquierdo-Cortazar*

Título: ***Determining the Geographical distribution of a Community by means of a Time-zone Analysis***

Tipo de participación: *Ponencia*

Congreso: *12th International Symposium on Open Collaboration, OpenSym 2016*

Publicación: *In proceedings (3:1-3:4)*

Lugar de celebración: *Berlin, Germany*

Fecha: *Aug 17–19, 2016*

DOI [10.1145/2957792.2957802](https://doi.org/10.1145/2957792.2957802)

---

Autores: *Jesús M. González-Barahona, Gregorio Robles, Daniel Izquierdo-Cortazar*

Título: ***The MetricsGrimoire Database Collection***

Tipo de participación: *Ponencia*

Congreso: *International Working Conference on Mining Software Repositories, MSR 2015*

Publicación: *In proceedings (478-481)*

Lugar de celebración: *Florence, Italy*

Fecha: *May 16–17, 2015*

DOI [10.1109/MSR.2015.68](https://doi.org/10.1109/MSR.2015.68)

---

Autores: *Gregorio Robles, Jesús M. Gonzalez-Barahona, Carlos Cervigón, Andrea Capiluppi, Daniel Izquierdo-Cortazar*

Título: ***Estimating development effort in Free/Open source software projects by mining software repositories: a case study of OpenStack***

Tipo de participación: *Ponencia*

Congreso: *International Working Conference on Mining Software Repositories, MSR 2014*

Publicación: *In proceedings (222-231)*

Lugar de celebración: *Hyderabad, India*

Fecha: *May 31 – June 1, 2014*

ISBN [978-1-4503-2863-0](https://www.isbn-international.org/product/978-1-4503-2863-0)

---

Autores: *Gregorio Robles, Laura Arjona Reina, Alexander Serebrenik, Bogdan Vasilescu, Jesús M. Gonzalez-Barahona*

Título: ***FLOSS 2013: a survey dataset about free software contributors: challenges for curating, sharing, and combining***

Tipo de participación: *Ponencia*

Congreso: *International Working Conference on Mining Software Repositories, MSR 2014*

Publicación: *In proceedings (396-399)*

Lugar de celebración: *Hyderabad, India*

Fecha: *May 31 – June 1, 2014*

*ISBN 978-1-4503-2863-0*

---

Autores: *Jesus M. González-Barahona, Daniel Izquierdo-Cortazar, Gregorio Robles, Mario Gallegos*

Título: ***Code Review Analytics: WebKit as Case Study***

Tipo de participación: *Ponencia*

Congreso: *10th IFIP WG 2.13 International Conference on Open Source Systems, OSS 2014*

Publicación: *In proceedings (1-10)*

Lugar de celebración: *San José, Costa Rica*

Fecha: *May 6-9, 2014*

---

Autores: *Sergio Raúl Montes León, Gregorio Robles, Jesús M González-Barahona*

Título: ***Considerations Regarding the Creation of a Post-graduate Master's Degree in Free Software***

Tipo de participación: *Ponencia*

Congreso: *10th IFIP WG 2.13 International Conference on Open Source Systems, OSS 2014*

Publicación: *In proceedings (123-132)*

Lugar de celebración: *San José, Costa Rica*

Fecha: *May 6-9, 2014*

---

Autores: *Gregorio Robles, Hugo Plaza, Jesús M. Gonzalez-Barahona*

Título: ***Free/Open Source Software projects as early MOOCs***

Tipo de participación: *Ponencia*

Congreso: *IEEE Global Engineering Education Conference, EDUCON 2014*

Publicación: *In proceedings (878-883)*

Lugar de celebración: *Istanbul, Turkey*

Fecha: *April 3-5, 2014*

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona*

Título: ***Mining student repositories to gain learning analytics: An experience report***

Tipo de participación: *Ponencia*

Congreso: *IEEE Global Engineering Education Conference, EDUCON 2013*

Publicación: *In proceedings (1249-1254)*

Lugar de celebración: *Berlin, Germany*

Fecha: *March 13-15, 2013*

---

Autores: *Laura Arjona Reina, Gregorio Robles, Jesús M. González-Barahona*

Título: ***A Preliminary Analysis of Localization in Free Software: How Translations Are Performed***

Tipo de participación: *Ponencia*

Congreso: *Open Source Software: Quality Verification*

Publicación: *In proceedings (153-167)*

Lugar de celebración: *Kapodistria, Slovenia*

Fecha: *June 2013*

---

Autores: *Santiago Gala-Pérez, Gregorio Robles, Jesús M. González-Barahona, Israel Herraiz*  
Título: ***Intensive metrics for the study of the evolution of open source projects: case studies from apache software foundation projects***  
Tipo de participación: *Ponencia*  
Congreso: *Tenth International Workshop on Mining Software Repositories*  
Publicación: *In proceedings (159-168)*  
Lugar de celebración: *San Francisco, California*  
Fecha: *May 2013*

---

Autores: *Gregorio Robles, Jesús M. González-Barahona*  
Título: ***A Comprehensive Study of Software Forks: Dates, Reasons and Outcomes***  
Tipo de participación: *Ponencia*  
Congreso: *Open Source Software: 8th International IFIP WG 2.13 Conference on Open Source Systems, OSS 2012*  
Publicación: *In proceedings - Best paper award*  
Lugar de celebración: *Hammamet, Túnez*  
Fecha: *September 2012*  
***Citas en Google Scholar: 16 (dic. 2014)***

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona, Arturo Moral*  
Título: ***A synchronous on-line competition software to improve and motivate learning: Organizing (almost fully) automated quizzbowl with FLEQ***  
Tipo de participación: *Ponencia*  
Congreso: *IEEE Education Engineering (EDUCON 2012)*  
Publicación: *In proceedings*  
Lugar de celebración: *Marrakech, Morocco*  
Fecha: *17-20 April 2012*

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona, Jorge Fernández González, Julio Ramiro*  
Título: ***Hybrid Educational Worlds: Merging virtual worlds and the real world to efficiently create serious games with smartphones***  
Tipo de participación: *Ponencia*  
Congreso: *IEEE Education Engineering (EDUCON 2012)*  
Publicación: *In proceedings*  
Lugar de celebración: *Marrakech, Morocco*  
Fecha: *17-20 April 2012*

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona, Alfredo Prieto*  
Título: ***Sobre la dedicación temporal del profesor social***  
Tipo de participación: *Ponencia*  
Congreso: *VI Jornada de innovación pedagógica ADA-Madrid*  
Publicación: *En Actas*  
Lugar de celebración: *Madrid, España*  
Fecha: *29 de abril de 2011*

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona, Jorge Fernández*  
Título: ***Implementing Gymkhanas with Android Smartphones: a Multimedia M-Learning Game***  
Tipo de participación: *Ponencia*  
Congreso: *IEEE Education Engineering (EDUCON 2011)*  
Publicación: *In proceedings*  
Lugar de celebración: *Amman, Jordan*  
Fecha: *4-6 April 2011*

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona, Jorge Fernández*  
Título: ***New Trends from Libre Software that May Change Education***  
Tipo de participación: *Ponencia*  
Congreso: *IEEE Education Engineering (EDUCON 2011)*  
Publicación: *In proceedings*  
Lugar de celebración: *Amman, Jordan*  
Fecha: *4-6 April 2011*

---

Autores: *Antonio Reinoso, Jesus M. Gonzalez-Barahona, Rocío Miñoz-Mansilla, Israel Herraiz Tabernero*  
Título: ***A quantitative examination of the impact of featured articles in Wikipedia***  
Tipo de participación: *Ponencia*  
Congreso: *International Conference on Software and Data Technologies, ICSOFT 2011*  
Publicación: *In proceedings*  
Lugar de celebración: *Sevilla*  
Fecha: *18-21 julio 2011*

---

Autores: *Daniel Izquierdo-Cortazar, Jesús M. Gonzalez-Barahona, Santiago Dueñas, Gregorio Robles*  
Título: ***Towards Automated Quality Models for Software Development Communities: The QualOSS and FLOSSMetrics Case***  
Tipo de participación: *Ponencia*  
Congreso: *Seventh International Conference on the Quality of Information and Communications Technology (QUATIC 2010)*  
Publicación: *In proceedings*  
Lugar de celebración: *Porto, Portugal*  
Fecha: *29 Sept. – 2 October, 2010*

---

Autores: *Megan Squire, Jesús M. González-Barahona, Greg Madey*  
Título: ***The Present and Future of FLOSS Data Archives***  
Tipo de participación: *Panel*  
Congreso: *International Conference on Open Source Systems (OSS 2010)*  
Publicación: *IFIP Advances in Information and Communication Technology, 2010, Volume 319/2010, Springer*  
Lugar de celebración: *Uni. Notre Dame, Indiana, USA*  
Fecha: *June 2010*

---

Autores: *Daniel Izquierdo-Cortazar, Jesús M. González-Barahona, Gregorio Robles*  
Título: ***FLOSS Communities: Analyzing Evolvability and Robustness from an Industrial Perspective***  
Tipo de participación: *Panel*  
Congreso: *International Conference on Open Source Systems (OSS 2010)*  
Publicación: *IFIP Advances in Information and Communication Technology, 2010, Volume 319/2010, Springer*  
Lugar de celebración: *Uni. Notre Dame, Indiana, USA*  
Fecha: *June 2010*

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona*

Título: ***An e-learning Role Playing Game: Trials using LMS (Moodle) and Micro-Blogging (Twitter)***

Tipo de participación: *Ponencia*

Congreso: *I Jornadas de Innovación y TIC Educativas - JITICE 2010*

Publicación: *In proceedings*

Lugar de celebración: *Móstoles, España*

Fecha: *June 2010*

---

Autores: *Jesus M. Gonzalez-Barahona, Gregorio Robles, Santiago Dueñas*

Título: ***Collecting data about FLOSS development: the FLOSSMetrics experience***

Tipo de participación: *Ponencia*

Congreso: *3rd International Workshop on Emerging Trends in Free/Libre/Open Source Software Research and Development (FLOSS 2010)*

Publicación: *In proceedings*

Lugar de celebración: *Cape Town, South Africa*

Fecha: *May 2010*

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona, Alfredo Prieto*

Título: ***Fomentando la preparación de clase por parte de los alumnos mediante el Campus Virtual***

Tipo de participación: *Ponencia*

Congreso: *V Jornada de innovación pedagógica ADA-Madrid*

Publicación: *En Actas*

Lugar de celebración: *Alcalá de Henares, España*

Fecha: *16 de abril de 2010*

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona*

Título: ***Influence of libre software in education: The blogs planet case***

Tipo de participación: *Ponencia*

Congreso: *IEEE Education Engineering (EDUCON 2010)*

Publicación: *In proceedings*

Lugar de celebración: *Madrid, Spain*

Fecha: *14-16 April 2010*

---

Autores: *Israel Herraiz, Gregorio Robles, Jesus M. Gonzalez-Barahona*

Título: ***Research friendly software repositories***

Tipo de participación: *Ponencia*

Congreso: *Joint ERCIM Workshop on Software Evolution (EVOL) and International Workshop on Principles of Software Evolution (IWPSE)*

Publicación: *In proceedings*

Lugar de celebración: *Amsterdam, Neederlands*

Fecha: *August 2009*

---

Autores: *Israel Herraiz, Daniel Izquierdo-Cortazar, Francisco Rivas-Hernández, Jesús Gonzalez-Barahona, Gregorio Robles, S Duenas-Dominguez, Carlos Garcia-Campos, Juan Francisco Gato, Lilitiana Tovar*

Título: ***Flossmetrics: Free/libre/open source software metrics***

Tipo de participación: *Ponencia*

Congreso: *13rd European Conference on Software Maintenance and Reengineering, CSMR 2009*

Publicación: *In proceedings*

Lugar de celebración: *Kaiserslautern, Alemania*

Fecha: *24 de marzo de 2009*

---

Autores: *Antonio J. Reinoso, Jesús M. González-Barahona, Gregorio Robles, Felipe Ortega*

Título: ***A Quantitative Approach to the Use of the Wikipedia***

Tipo de participación: *Ponencia*

Congreso: *IEEE Symposium on Computers and Communications (ISCC '09)*

Publicación: *In Proceedings*

Lugar de celebración: *Sousse, Tunes*

Fecha: *Julio 2009*

---

Autores: *Felipe Ortega, Isabel Lafuente, José Gato, Jesús M. González-Barahona*

Título: ***Libre Software in Spanish Public Administrations***

Tipo de participación: *Poster*

Congreso: *International Conference on Open Source Systems (OSS 2009)*

Publicación: *In Proceedings*

Lugar de celebración: *Skövde, Sweden*

Fecha: *June 2009*

---

Autores: *Daniel German, Jesús M. González-Barahona*

Título: ***An empirical study of the reuse of software licensed under the GNU General Public License***

Tipo de participación: *Ponencia*

Congreso: *International Conference on Open Source Systems (OSS 2009)*

Publicación: *In Proceedings*

Lugar de celebración: *Skövde, Sweden*

Fecha: *June 2009*

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona, Israel Herraiz*

Título: ***Evolution of the core team of developers in libre software projects***

Tipo de participación: *Ponencia*

Congreso: *6th IEEE International Working Conference on Mining Software Repositories (MSR 2009)*

Publicación: *In proceedings*

Lugar de celebración: *Vancouver, BC, Canada*

Fecha: *May 2009*

***Citas en Google Scholar: 34 (dic. 2014)***

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona*

Título: ***El juicio como un juego de roles para fomentar el debate como forma de aprendizaje en un entorno de e-learning***

Tipo de participación: *Ponencia*

Congreso: *IV Jornada de innovación pedagógica ADA-Madrid*

Publicación: *En Actas*

Lugar de celebración: *Leganés, España*

Fecha: *10 de marzo de 2009*

---

Autores: *Daniel Izquierdo-Cortázar, Felipe Ortega, Jesús M. González-Barahona, Gregorio Robles*  
Título: ***Using Software Archaeology to Measure Knowledge Loss in Software Projects Due to Developer Turnover***

Tipo de participación: *Ponencia*

Congreso: *42nd Hawaii International Conference on System Sciences (HICSS 09)*

Publicación: *In Proceedings*

Lugar de celebración: *Hawaii, USA*

Fecha: *January 2009*

***Citas en Google Scholar: 17 (dic. 2014)***

---

Autores: *Felipe Ortega, Daniel Izquierdo-Cortázar, Jesús M. González-Barahona, Gregorio Robles*  
Título: ***On the Analysis of Contributions of Privileged Users in Virtual Open Communities***

Tipo de participación: *Ponencia*

Congreso: *42nd Hawaii International Conference on System Sciences (HICSS 09)*

Publicación: *In Proceedings*

Lugar de celebración: *Hawaii, USA*

Fecha: *January 2009*

---

Autores: *Antonio J. Reinoso, Jesús M. González-Barahona, Felipe Ortega, Gregorio Robles*

Título: ***Quantitative Analysis and Characterization of Wikipedia Requests***

Tipo de participación: *Póster*

Congreso: *ACM WikiSym 2008: 4th International Symposium on Wikis*

Publicación: *In Proceedings*

Lugar de celebración: *Porto, Portugal*

Fecha: *September 2008*

---

Autores: *Juan Martínez-Romo, Gregorio Robles, Jesus M. Gonzalez-Barahona, Miguel Ortuño*

Título: ***Using Social Network Analysis Techniques to Study Collaboration between a FLOSS Community and a Company***

Tipo de participación: *Ponencia*

Congreso: *The Fourth International Conference on Open Source Systems (OSS2008)*

Publicación: *In Proceedings*

Lugar de celebración: *Milan, Italia*

Fecha: *Sep 2008*

***Citas en Google Scholar: 22 (dic. 2014)***

---

Autores: *Jesus M. Gonzalez-Barahona, Andrés Martínez, Alvaro Polo, Juan José Hierro, Marcos Reyes, Javier Soriano, Rafael Fernández*

Título: ***The networked forge: new environments for libre software development***

Tipo de participación: *Ponencia*

Congreso: *The Fourth International Conference on Open Source Systems (OSS2008)*

Publicación: *In Proceedings*

Lugar de celebración: *Milan, Italia*

Fecha: *Sep 2008*

---



Autores: *Israel Herraiz, Gregorio Robles, Rafael Capilla, Jesus M. Gonzalez-Barahona*  
 Título: ***Managing Libre Software Distributions under a Product Line Approach***  
 Tipo de participación: *Ponencia*  
 Congreso: *The IEEE International Workshop on Quality and Architectural Concerns in Open Source Software (QACOS 2008), colocated with 32nd Annual IEEE International Computer Software and Applications Conference (COMPSAC 2008)*  
 Publicación: *Proceedings of the 32nd Annual IEEE International Computer Software and Applications Conference*  
 Lugar de celebración: *Turku, Finlandia*  
 Fecha: *Jul 2008*

---

Autores: *Israel Herraiz, Daniel German, Jesus M. Gonzalez-Barahona, Gregorio Robles*  
 Título: ***Towards a simplification of the bug report form in Eclipse***  
 Tipo de participación: *Ponencia*  
 Congreso: *2008 International Working Conference on Mining Software Repositories (MSR 2008)*  
 Publicación: *In Proceedings*  
 Lugar de celebración: *Leipzig, Germany*  
 Fecha: *10-11, May 2008*  
***Citas en Google Scholar: 46 (dic. 2014)***

---

Autores: *Israel Herraiz, Jesus M. Gonzalez-Barahona, Gregorio Robles*  
 Título: ***Determinism and evolution***  
 Tipo de participación: *Ponencia*  
 Congreso: *2008 International Working Conference on Mining Software Repositories (MSR 2008)*  
 Publicación: *In Proceedings*  
 Lugar de celebración: *Leipzig, Germany*  
 Fecha: *10-11, May 2008*  
***Citas en Google Scholar: 25 (dic. 2014)***

---

Autores: *Felipe Ortega, Jesus M. Gonzalez-Barahona, Gregorio Robles*  
 Título: ***On The Inequality of Contributions to Wikipedia***  
 Tipo de participación: *Ponencia*  
 Congreso: *Hawaii International Conference on System Sciences (HICCS 41)*  
 Publicación: *In Proceedings*  
 Lugar de celebración: *Waikoloa, Hawaii, USA*  
 Fecha: *7-10 de enero de 2008*  
***Citas en Google Scholar: 104 (dic. 2014)***

---

Autores: *Daniel M. German, Jesus M. Gonzalez-Barahona, Gregorio Robles*  
 Título: ***A Model to Understand the Building and Running Inter-Dependencies of Software***  
 Tipo de participación: *Ponencia*  
 Congreso: *14th Working Conference on Reverse Engineering (WCRE 2007)*  
 Publicación: *In Proceedings*  
 Lugar de celebración: *Vancouver, Canada*  
 Fecha: *28-31 de octubre de 2007*  
***Citas en Google Scholar: 19 (dic. 2014)***

---

Autores: *Felipe Ortega, Jesus M. Gonzalez-Barahona*  
 Título: ***Quantitative Analysis of the Wikipedia Community of Users***  
 Tipo de participación: *Ponencia*  
 Congreso: *International Symposium on Wikis (WikiSym 2007)*  
 Publicación: *In Proceedings*  
 Lugar de celebración: *Montreal, Canada*  
 Fecha: *21-23 de octubre de 2007*  
***Citas en Google Scholar: 84 (dic. 2014)***

---

Autores: *Israel Herraiz, Jesus M. Gonzalez-Barahona, Gregorio Robles and Daniel M. German*

Título: ***On the prediction of the evolution of libre software projects***

Tipo de participación: *Ponencia*

Congreso: *23rd IEEE International Conference on Software Maintenance*

Publicación: *In Proceedings*

Lugar de celebración: *Paris, France*

Fecha: *2-5 octubre de 2007*

***Citas en Google Scholar: 27 (dic. 2014)***

---

Autores: *Andrea Capiluppi, Jesus M. Gonzalez-Barahona, Israel Herraiz, Gregorio Robles*

Título: ***Adapting the Staged Model for Software Evolution to Free/Libre/Open Source Software***

Tipo de participación: *Ponencia*

Congreso: *International Workshop on Principles of Software Evolution (IWPSE 2007)*

Publicación: *In Proceedings*

Lugar de celebración: *Dubrovnik, Croatia*

Fecha: *3-4 de septiembre de 2007*

***Citas en Google Scholar: 31 (dic. 2014)***

---

Autores: *Jose Felipe Ortega Soto, Jesus M. Gonzalez-Barahona, Gregorio Robles*

Título: ***The Top-Ten Wikipedias***

Tipo de participación: *Ponencia*

Congreso: *2nd International Conference on Software and Data Technologies (ICSOF 2007) (Best Student Paper Award)*

Publicación: *In Proceedings*

Lugar de celebración: *Barcelona, Spain*

Fecha: *22-25 de julio de 2007*

***Citas en Google Scholar: 20 (dic. 2014)***

---

Autores: *Daniel M. German, Jesus M. Gonzalez-Barahona, Gregorio Robles*

Título: ***In what do you trust when you trust? The importance of dependencies in trust analysis***

Tipo de participación: *Ponencia*

Congreso: *1st International Workshop on Trust in Open Source Software (TOSS)*

Publicación: *In Proceedings*

Lugar de celebración: *Limerick, Ireland*

Fecha: *14 de junio de 2007*

---

Autores: *Gregorio Robles, Santiago Dueñas, Jesus M. Gonzalez-Barahona*

Título: ***Corporate Involvement of Libre Software: Study of Presence in Debian Code over Time***

Tipo de participación: *Ponencia*

Congreso: *3rd International Conference on Open Source Systems (OSS2007)*

Publicación: *"Open Source Development, Adoption and Innovation", pp. 121-132, Springer*

Lugar de celebración: *Limerick, Ireland*

Fecha: *11-14 de junio de 2007*

***Citas en Google Scholar: 24 (dic. 2014)***

---

Autores: *Jesus M. Gonzalez-Barahona, Gregorio Robles, Israel Herraiz*

Título: ***Impact of the Creation of the Mozilla Foundation in the Activity of Developers***

Tipo de participación: *Ponencia*

Congreso: *Fourth International Workshop on Mining Software Repositories, colocated with the International Congress on Software Engineering*

Publicación: *In Proceedings*

Lugar de celebración: *Minneapolis, USA*

Fecha: *19-20 de mayo de 2007*

---

Autores: *Israel Herraiz, Jesus M. Gonzalez-Barahona, Gregorio Robles*  
 Título: ***Forecasting the number of changes in Eclipse using time series analysis***  
 Tipo de participación: *Ponencia*  
 Congreso: *Fourth International Workshop on Mining Software Repositories, colocated with the International Congress on Software Engineering*  
 Publicación: *In Proceedings*  
 Lugar de celebración: *Minneapolis, USA*  
 Fecha: *19-20 de mayo de 2007*  
***Citas en Google Scholar: 26 (dic. 2014)***

---

Autores: *Israel Herraiz, Jesus M. Gonzalez-Barahona, Gregorio Robles*  
 Título: ***Towards a theoretical model for software growth***  
 Tipo de participación: *Ponencia*  
 Congreso: *Fourth International Workshop on Mining Software Repositories, colocated with the International Congress on Software Engineering*  
 Publicación: *In Proceedings*  
 Lugar de celebración: *Minneapolis, USA*  
 Fecha: *19-20 de mayo de 2007*  
***Citas en Google Scholar: 59 (dic. 2014)***

---

Autores: *Jesus M. Gonzalez-Barahona, Vania Dimitrova, Diego Chaparro, Chris Tebb, Teófilo Romero, Luis Cañas, Julika Matravers, Styliani Kleanthous*  
 Título: ***Towards Community-Driven Development of Educational Materials: The Edukalibre Approach***  
 Tipo de participación: *Ponencia*  
 Congreso: *First European Conference on Technology Enhanced Learning*  
 Publicación: *"Innovative Approaches for Learning and Knowledge Sharing", LNCS 4227, pp. 125-139, W. Nejdl and K. Tochtermann (Eds.), Springer-Verlag Heidelberg 2006*  
 Lugar de celebración: *Creta, Grecia*  
 Fecha: *1-4 octubre de 2006*

---

Autores: *Juan José Amor, Gregorio Robles, Jesus M. Gonzalez-Barahona, Alvaro Navarro*  
 Título: ***Discriminating Development Activities in Versioning Systems: A Case Study***  
 Tipo de participación: *Ponencia*  
 Congreso: *International Workshop on Predictor Models in Software Engineering (PROMISE 2006)*  
 Publicación: *In Proceedings*  
 Lugar de celebración: *Philadelphia, USA*  
 Fecha: *September 2006*

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona*  
 Título: ***Contributor Turnover in Libre Software Projects***  
 Tipo de participación: *Ponencia*  
 Congreso: *Second International Conference on Open Source Systems (OSS2006)*  
 Publicación: *In Proceedings*  
 Lugar de celebración: *Como, Italia*  
 Fecha: *Junio 2006*  
***Citas en Google Scholar: 59 (dic. 2014)***

---

Autores: *Daniel M. German, Gregorio Robles, Jesus M. Gonzalez-Barahona*

Título: ***The Challenges of Automated Quantitative Analysis of Open Source Software Projects***

Tipo de participación: *Ponencia*

Congreso: *Workshop on Evaluation Frameworks for Open Source Software (EFOSS), colocated with the Second International Conference on Open Source Systems (OSS2006)*

Publicación: *In Proceedings*

Lugar de celebración: *Como, Italia*

Fecha: *Junio 2006*

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona, Juan Luis Prieto*

Título: ***Assessing and Evaluating Documentation in Libre Software Projects***

Tipo de participación: *Ponencia*

Congreso: *Workshop on Evaluation Frameworks for Open Source Software (EFOSS), colocated with the Second International Conference on Open Source Systems (OSS2006)*

Publicación: *In Proceedings*

Lugar de celebración: *Como, Italia*

Fecha: *Junio 2006*

---

Autores: *Israel Herraiz, Gregorio Robles, Juan J. Amor, Teófilo Romera, Jesus M. Gonzalez-Barahona*

Título: ***The Process of Joining in Global Distributed Software Projects***

Tipo de participación: *Ponencia*

Congreso: *First International Workshop on Global Software Development for the Practitioner (GSD2006), colocated with the International Congress on Software Engineering*

Publicación: *In Proceedings*

Lugar de celebración: *Shanghai, China*

Fecha: *May 2006*

***Citas en Google Scholar: 49 (dic. 2014)***

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona, Martin Michlmayr, Juan Jose Amor*

Título: ***Mining Large Software Compilations over Time: Another Perspective of Software Evolution***

Tipo de participación: *Ponencia (Best Paper Award)*

Congreso: *Third International Workshop on Mining Software Repositories, colocated with the International Congress on Software Engineering*

Publicación: *In Proceedings*

Lugar de celebración: *Shanghai, China*

Fecha: *May 2006*

***Citas en Google Scholar: 60 (dic. 2014)***

---

Autores: *Gregorio Robles, Jesus M. Gonzalez-Barahona*

Título: ***Geographic Location of Developers at SourceForge***

Tipo de participación: *Ponencia*

Congreso: *Third International Workshop on Mining Software Repositories, colocated with the International Congress on Software Engineering*

Publicación: *In Proceedings*

Lugar de celebración: *Shanghai, China*

Fecha: *May 2006*

***Citas en Google Scholar: 32 (dic. 2014)***

---

Autores: *Juan José Amor, Gregorio Robles, Jesús M. González Barahona*

Título: ***Effort estimation by characterizing developer activity***

Tipo de participación: *Ponencia*

Congreso: *The 8th International Workshop on Economics-Driven Software Engineering Research (EDSER-8), colocated with the International Congress on Software Engineering*

Publicación: *In Proceedings*

Lugar de celebración: *Shanghai, China*

Fecha: *May 2006*

***Citas en Google Scholar: 24 (dic. 2014)***

---

Autores: *Israel Herraiz, Gregorio Robles, Jesús M. González Barahona, Juan F. Ramil, Andrea Capiluppi*

Título: ***Comparison between SLOCs and number of files as size metrics for software evolution analysis***

Tipo de participación: *Ponencia*

Congreso: *10th European Conference on Software Maintenance and Reengineering*

Publicación: *In Proceedings*

Lugar de celebración: *Bari, Italia*

Fecha: *Marzo de 2006*

***Citas en Google Scholar: 43 (dic. 2014)***

---

Autores: *Gregorio Robles, Jesús M. González Barahona, Israel Herraiz*

Título: ***An Empirical Approach to Software Archaeology***

Tipo de participación: *Póster*

Congreso: *International Conference on Software Maintenance 2005*

Publicación: *ICSM 2005 Poster Proceedings, ISBN: 963-460-981-3*

Lugar de celebración: *Budapest, Hungría*

Fecha: *Septiembre de 2005*

---

Autores: *Gregorio Robles, Juan José Amor, Jesús M. González Barahona, Israel Herraiz*

Título: ***Evolution and Growth in Large Libre Software Projects***

Tipo de participación: *Full paper*

Congreso: *International Workshop on Principles of Software Evolution 2005*

Publicación: *Proceedings of the IWPSE 2005, ISBN: 0-7695-2349-8*

Lugar de celebración: *Lisboa, Portugal*

Fecha: *Septiembre de 2005*

***Citas en Google Scholar: 102 (dic. 2014)***

---

Autores: *Gregorio Robles, Jesús M. González-Barahona, Martin Michlmayr*

Título: ***Evolution of volunteer participation in libre software projects: evidence from Debian***

Tipo de participación: *Ponencia*

Congreso: *First International Conference on Open Source Systems (OSS 2005)*

Publicación: *OSS 2005 Proceedings, ISBN: 88-7544-048-4*

Lugar de celebración: *Genoa, Italy*

Fecha: *11-13 July 2005*

***Citas en Google Scholar: 58 (dic. 2014)***

---

Autores: *Jesús G. Barahona, Chris Tebb, Vania Dimitrova, Diego Chaparro, Teófilo Romera*

Título: ***Transferring Libre Software Development Practices too the Production of Educational Resources***

Tipo de participación: *Ponencia*

Congreso: *First International Conference on Open Source Systems*

Publicación: *In Proceedings*

Lugar de celebración: *Genoa, Italy*

Fecha: *11-15 July 2005*

---

Autores: *Juan José Amor, Gregorio Robles, Israel Herraiz, Jesús M. González Barahona*

Título: ***From Pigs to Stripes: A Travel Through Debian***

Tipo de participación: *Ponencia*

Congreso: *DebConf5 (Debian Annual Developers Meeting)*

Publicación: *Proceedings.*

Lugar de celebración: *Helsinki, Finlandia*

Fecha: *Julio de 2005*

Autores: *Juan José Amor Iglesias, Gregorio Robles, Jesús M. González Barahona*

Título: ***GNOME como caso de estudio de Ingeniería de Software Libre***

Tipo de participación: *Ponencia*

Congreso: *II GUADEC Hispana*

Publicación: *Actas de la II GUADEC Hispana, ISBN: 84-689-5719-4*

Lugar de celebración: *La Coruña*

Fecha: *Mayo de 2005*

Autores: *Gregorio Robles, Jesús M. González Barahona*

Título: ***Developer identification methods for integrated data from various sources***

Tipo de participación: *Ponencia*

Congreso: *2nd Workshop on Mining Software Repositories, co-located at the 26th ICSE.*

Publicación: *Proceedings of the 2nd Workshop on Mining Software Repositories, also published in ACM SIGSOFT Software Engineering Notes, 30(4)*

Lugar de celebración: *St. Louis, MI, USA*

Fecha: *May 17th 2005*

***Citas en Google Scholar: 60 (dic. 2014)***

Autores: *Israel Herraiz, Gregorio Robles, Jesús M. González-Barahona*

Título: ***Towards Predictor Models for large Libre Software Projects***

Tipo de participación: *Ponencia*

Congreso: *PROMISE 2005: International Workshop on Predictor Models in Software Engineering co-located at the 21st International Conference on Software Engineering*

Publicación: *Proceedings, ISBN: 159593-125-2, publicado también en ACM SIGSOFT Software Engineering Notes, 30(4)*

Lugar de celebración: *St. Louis, Missouri, USA*

Fecha: *May 15th 2005*

***Citas en Google Scholar: 11 (dic. 2014)***

Autores: *Gregorio Robles, Juan Julián Merelo Guervós, Jesús M. González Barahona*

Título: ***Self-organized development in libre software projects: a model based on the stigmergy concept***

Tipo de participación: *Ponencia*

Congreso: *Proceedings of the 6th International Workshop on Software Process Simulation and Modeling (ProSim 2005) (co-located at the 21st International Conference on Software Engineering)*

Publicación: *Proceedings of the ProSim 2005, ISBN: 3-8167-6761-3*

Lugar de celebración: *St. Louis, Missouri, USA*

Fecha: *14-15 May 2005*

***Citas en Google Scholar: 35 (dic. 2014)***

Autores: *Jesús G. Barahona, Luis Lopez, Diego Chaparro, Luis Cañas*

Título: ***Edukalibre: a tool for collaborative creation of edukational material***

Tipo de participación: *Ponencia*

Congreso: *Foro hispano de .LRN y software libre educativo*

Publicación: *In Proceedings*

Lugar de celebración: *Madrid, Spain*

Fecha: *10-11 May 2005*

Autores: *Diego Barceló, Gregorio Robles, Jesús M. González Barahona*  
 Título: ***Evolución de la participación de empresas, instituciones e individuos en el desarrollo de software libre***  
 Tipo de participación: *Ponencia*  
 Congreso: *V Jornadas de Software Libre en Asturias*  
 Publicación: *Actas de las V Jornadas de Software Libre en Asturias.*  
 Lugar de celebración: *Pabellón Príncipe Felipe, Oviedo, Asturias*  
 Fecha: *Marzo 2005*

---

Autores: *Bernard Lang, Jean-François Abramatic, Jesús M. González-Barahona, Piera Gómez, Mogens Kühn Pedersen*  
 Título: ***Free and Proprietary Software in COTS-Based Software Development***  
 Tipo de participación: *Panel*  
 Congreso: *COTS-Based Software Systems: 4th International Conference, ICCBSS 2005*  
 Publicación: *In Proceedings*  
 Lugar de celebración: *Bilbao*  
 Fecha: *9 de febrero de 2005*

---

Autores: *Jesús M. González Barahona, Gregorio Robles*  
 Título: ***On Libre Software Engineering***  
 Tipo de participación: *Ponencia electrónica*  
 Congreso: *V International Conference of UNIX at Uninet*  
 Publicación: *Actas electrónicas*  
 Lugar de celebración: *irc.uninet.edu*  
 Fecha: *20 de diciembre de 2004*

---

Autores: *Israel Herraiz Tabernero, Gregorio Robles Martínez, Jesús M. González Barahona*  
 Título: ***CVSAnalY: una herramienta libre para el análisis de repositorios CVS***  
 Tipo de participación: *Ponencia*  
 Congreso: *IV Jornadas Andaluzas de Software Libre*  
 Publicación: *Actas de las IV Jornadas Andaluzas de Software Libre*  
 Lugar de celebración: *Algeciras*  
 Fecha: *5 y 6 de noviembre de 2004*

---

Autores: *Jesús M. Gonzalez Barahona*  
 Título: ***Edukalibre project: exploring libre software models to produce educational materials***  
 Tipo de participación: *Ponencia invitada*  
 Congreso: *SIGOSSEE Public Seminar*  
 Publicación: *In Proceedings*  
 Lugar de celebración: *Barcelona, Spain*  
 Fecha: *November 2nd 2004*

---

Autores: *Gregorio Robles, Jesús M. González-Barahona*  
 Título: ***Executable source code and non-executable source code: analysis and relationships***  
 Tipo de participación: *Ponencia*  
 Congreso: *4th Workshop on Source Code Analysis and Manipulation. 20th International Conference on Software Maintenance*  
 Publicación: *In Proceedings*  
 Lugar de celebración: *Chicago*  
 Fecha: *15 de septiembre de 2004*

---

Autores: *Jesús M. González Barahona, Gregorio Robles*  
Título: ***Getting the global picture***  
Tipo de participación: *Ponencia*  
Congreso: *Oxford Workshop on Libre Software 2004*  
Publicación: *In Proceedings*  
Lugar de celebración: *Oxford, UK*  
Fecha: *June 25th 2004*

---

Autores: *Jesús M. González-Barahona, Luis López, Gregorio Robles*  
Título: ***Community structure of modules in the Apache project***  
Tipo de participación: *Ponencia*  
Congreso: *4th Workshop on Open Source Software Engineering. 26th International Conference on Software Engineering*  
Publicación: *In Proceedings*  
Lugar de celebración: *Edimburgo, UK*  
Fecha: *25 de mayo de 2004*  
***Citas en Google Scholar: 49 (dic. 2014)***

---

Autores: *Luis López, Jesús M. González-Barahona, Gregorio Robles*  
Título: ***Applying Social Network Analysis to the Information in CVS Repositories***  
Tipo de participación: *Ponencia*  
Congreso: *Mining Software Repositories Workshop. 26th International Conference on Software Engineering*  
Publicación: *In Proceedings*  
Lugar de celebración: *Edimburgo, UK*  
Fecha: *25 de mayo de 2004*  
***Citas en Google Scholar: 133 (dic. 2014)***

---

Autores: *Gregorio Robles, Jesús M. González-Barahona, Rishab A. Ghosh*  
Título: ***GlueTheos: Automating the Retrieval and Analysis of Data from Publicly Available Repositories***  
Tipo de participación: *Ponencia*  
Congreso: *Mining Software Repositories Workshop. 26th International Conference on Software Engineering*  
Publicación: *In Proceedings*  
Lugar de celebración: *Edimburgo, UK*  
Fecha: *25 de mayo de 2004*  
***Citas en Google Scholar: 31 (dic. 2014)***

---

Autores: *Gregorio Robles, Stefan Koch, Jesús M. González-Barahona*  
Título: ***Remote analysis and measurement of libre software systems by means of the CVSanaly tool***  
Tipo de participación: *Ponencia*  
Congreso: *2nd Workshop on Remote Analysis and Measurement of Software Systems (RAMSS '04). 26th International Conference on Software Engineering*  
Publicación: *In Proceedings*  
Lugar de celebración: *Edimburgo, UK*  
Fecha: *24 de mayo de 2004*  
***Citas en Google Scholar: 94 (dic. 2014)***

---



Autores: *Gregorio Robles, Jesús M. González Barahona*  
Título: ***Plone: experiencia docente***  
Tipo de participación: *Ponencia*  
Congreso: *III Jornadas Andaluzas de Software Libre*  
Publicación: *Actas de las III Jornadas Andaluzas de Software Libre*  
Lugar de celebración: *Granada, España*  
Fecha: *Noviembre de 2003*

---

Autores: *Eva M. Castro, Jesús González, Gregorio Robles, Tomás de Miguel*  
Título: ***Interoperabilidad de aplicaciones IPv4 e IPv6***  
Tipo de participación: *Ponencia*  
Congreso: *TELECOM I+D 2003*  
Publicación: *Actas del Congreso TELECOM I+D 2003*  
Lugar de celebración: *Madrid, España*  
Fecha: *Noviembre de 2003*

---

Autores: *Luis Rodero Merino, Miguel Ángel Ortuño Pérez, Jesús M. González Barahona, Vicente Matellán Olivera*  
Título: ***Monitorización de redes 802.11 con Linux***  
Tipo de participación: *Ponencia en acta*  
Congreso: *TELECOM I+D 2003*  
Publicación: *Actas del Congreso TELECOM I+D 2003*  
Lugar de celebración: *Madrid, España*  
Fecha: *Noviembre de 2003*

---

Autores: *Gregorio Robles Martínez, Jesús M. González Barahona, Luis Rodero Merino, Miguel A. Ortuño Pérez*  
Título: ***Historia del juguete: pasado, presente y futuro de Debian***  
Tipo de participación: *Ponencia en acta*  
Congreso: *VI Congreso HispaLinux*  
Publicación: *Actas del VI Congreso HispaLinux (publicación electrónica)*  
Lugar de celebración: *Universidad Rey Juan Carlos, Madrid*  
Fecha: *Septiembre 2003*

---

Autores: *Jesús M. González Barahona, Gregorio Robles*  
Título: ***Unmounting the “code god” assumption***  
Tipo de participación: *Ponencia*  
Congreso: *Fourth International Conference on eXtreme Programming and Agile Processes in Software Engineering*  
Publicación: *Proceedings of the Fourth International Conference on eXtreme Programming and Agile Processes in Software Engineering*  
Lugar de celebración: *Génova, Italia*  
Fecha: *Mayo 2003*  
***Citas en Google Scholar: 21 (dic. 2014)***

---

Autores: *Gregorio Robles-Martínez, Jesús M. González-Barahona, José Centeno-González, Vicente Matellán-Olivera, Luis Roderó-Merino*

Título: ***Studying the evolution of libre software projects using publicly available data***

Tipo de participación: *Ponencia*

Congreso: *3rd Workshop on Open Source Software Engineering, 25th International Conference on Software Engineering*

Publicación: *Proceedings of the 3rd Workshop on Open Source Software Engineering*

Lugar de celebración: *Portland, Oregon, EE.UU.*

Fecha: *3 de mayo 2003*

***Citas en Google Scholar: 52 (dic. 2014)***

---

Autores: *Jesús M. González Barahona*

Título: ***Cómo ofrecer servicios de alojamiento de proyectos de software libre***

Tipo de participación: *Ponencia*

Congreso: *Jornadas Técnicas de RedIRIS 2002*

Publicación: *Actas de las Jornadas Técnicas de RedIRIS 2002*

Lugar de celebración: *Salamanca, España*

Fecha: *6-8 de noviembre de 2002*

---

Autores: *Jesús M. González Barahona, Pedro de las Heras Quirós*

Título: ***Can Libre Software Hosting Services be Built as Peer-to-Peer Systems?***

Tipo de participación: *Ponencia*

Congreso: *International Workshop on Future Directions on Distributed Computing*

Publicación: *Proceedings of the International Workshop on Future Directions on Distributed Computing*

Lugar de celebración: *Bertinoro (Forli), Italia*

Fecha: *2-7 de junio de 2002*

---

Autores: *Jesús M. González Barahona*

Título: ***Uso de Ada en prácticas: sistemas basados en HTTP***

Tipo de participación: *Ponencia*

Congreso: *X Jornada Técnica AdaSpain*

Publicación: *Actas de la X Jornada Técnica AdaSpain*

Lugar de celebración: *Madrid, España*

Fecha: *12 de noviembre de 2001*

---

Autores: *Vicente Matellán, Jesús González Barahona, Pedro de las Heras, José Centeno*

Título: ***Opciones para la programación de robots LEGO Mindstorms***

Tipo de participación: *Ponencia*

Congreso: *I Workshop en agentes físicos*

Publicación: *Actas del I Workshop en agentes físicos*

Lugar de celebración: *Madrid*

Fecha: *15-16 de marzo de 2001*

---

Autores: *Pedro de las Heras Quirós, Jesús M. González, Vicente Matellán y José Centeno*

Título: ***Actualidad del software libre***

Tipo de participación: *Ponencia*

Congreso: *III Congreso de Hispalinux*

Publicación: *Actas del III Congreso de Hispalinux (publicación electrónica)*

Lugar de celebración: *Leganés, Madrid*

Fecha: *Noviembre de 2000*

---

Autores: *Vicente Matellán Olivera, Jesús María González Barahona, José Centeno González y Pedro de las Heras Quirós*

Título: ***Libre software environment for robot programming***

Tipo de participación: *Ponencia*

Congreso: *2000 IEEE Conference on Systems Man and Cybernetics*

Publicación: *Proceedings of 2000 IEEE Conference on Systems Man and Cybernetics, ISBN 0-7803-6583-6, Vol. 1, 718-723*

Lugar de celebración: *Nashville (EEUU)*

Fecha: *8-10 Octubre 2000*

---

Autores: *Vicente Matellán Olivera, Jesús María González Barahona, José Centeno González y Pedro de las Heras Quirós*

Título: ***Herramientas Libres para la Programación de Agentes Físicos***

Tipo de participación: *Ponencia*

Congreso: *Primer Workshop Hispano-Luso de Agentes Físicos*

Publicación: *Actas del Primer Workshop Hispano-Luso de Agentes Físicos*

Lugar de celebración: *Tarragona (España)*

Fecha: *13-14 de Septiembre de 2000*

---

Autores: *Jesús M. González Barahona, Vicente Matellán Olivera, Pedro de las Heras Quirós, José Centeno González*

Título: ***Soporte de depuración para Ada (GDB, DDD)***

Tipo de participación: *Ponencia*

Congreso: *IX Jornada Técnica AdaSpain*

Publicación: *Actas de la IX Jornada Técnica AdaSpain*

Lugar de celebración: *Madrid, España*

Fecha: *Junio de 2000*

---

Autores: *Vicente Matellán, Jesús M. González, Pedro de las Heras y José Centeno*

Título: ***Software libre en la publicación electrónica: El ejemplo de Open Resources***

Tipo de participación: *Ponencia*

Congreso: *Jornadas sobre Publicación Electrónica*

Publicación: *Actas de las Jornadas sobre Publicación Electrónica: Un Nuevo Espacio de Comunicación, pp. 149-160*

Lugar de celebración: *Leganés (Madrid)*

Fecha: *8-9 Julio 1999*

---

Autores: *José Francisco Ruiz Martínez and Jesús M. González-Barahona*

Título: ***Implementing a New Low level Tasking Support for the GNAT Runtime System***

Tipo de participación: *Ponencia*

Congreso: *Conference on Reliable Software Technologies AdaEurope'99*

Publicación: *Proceedings of the AdaEurope'99 Conference, Springer Lecture Notes on Computer Science*

Lugar de celebración: *Santander, España*

Fecha: *Junio 1999*

---

Autores: *Juan A. de la Puente, José Francisco Ruiz and Jesús González- Barahona*  
 Título: ***Real-Time Programming with GNAT: Specialised Kernels versus POSIX Threads***  
 Tipo de participación: *Ponencia*  
 Congreso: *9th International Real-Time Ada Workshop*  
 Publicación: *Proceedings of the 9th International Real-Time Ada Workshop*  
 Lugar de celebración: *Tallahassee, Florida, EEUU*  
 Fecha: *March 1999*

---

Autores: *Vicente Matellán Olivera, Jesús M. González Barahona, Pedro de las Heras Quirós, José Centeno González, Francisco Ballesteros Cámara*  
 Título: ***Panorama del software libre***  
 Tipo de participación: *Ponencia*  
 Congreso: *Jornadas de Informática y Sociedad*  
 Publicación: *Actas de las Segundas Jornadas de Informática y Sociedad (JIS98), pp. 83-88*  
 Lugar de celebración: *Madrid (España)*  
 Fecha: *24-26 Noviembre, 1998*

---

Autores: *Vicente Matellán Olivera, Jesús M. González Barahona, Pedro de las Heras Quirós, José Centeno González, Francisco Ballesteros Cámara*  
 Título: ***El negocio del software libre: licencias, derechos y alternativas***  
 Tipo de participación: *Ponencia*  
 Congreso: *Jornadas de Informática y Sociedad*  
 Publicación: *Actas de las Segundas Jornadas de Informática y Sociedad (JIS98), pp. 137-142*  
 Lugar de celebración: *Madrid (España)*  
 Fecha: *24-26 Noviembre, 1998*

---

Autores: *Jesús M. González Barahona, Pedro de las Heras Quirós, José Centeno González, Francisco Ballesteros*  
 Título: ***Building Modular Communication Systems in Ada: the Simple.Com approach***  
 Tipo de participación: *Ponencia*  
 Congreso: *AdaEurope Third International Conference on Reliable Software Technologies*  
 Publicación: *Proceedings of the Third International Conference on Reliable Software Technologies, Springer Lecture Notes on Computer Science 1411, ISBN 3-540-64536-5*  
 Lugar de celebración: *Uppsala, Suecia*  
 Fecha: *June 1998*

---

Autores: *Pedro de las Heras Quirós, Jesús M. González Barahona, José Centeno González*  
 Título: ***Programming Distributed Fault Tolerant Systems: The replicAda Approach***  
 Tipo de participación: *Ponencia*  
 Congreso: *TRI-Ada'97 Conference*  
 Publicación: *Proceedings of the TRI-Ada'97 Conference, pag. 21-29, ISBN 0-89791-981-5*  
 Lugar de celebración: *St. Louis, Missouri, EE.UU.*  
 Fecha: *November 1997*

---

Autores: *Jesús M. González Barahona, José Centeno González, Pedro de las Heras Quirós, Francisco J. Ballesteros-Cámara, Luis López Fernández*  
 Título: ***Teaching Network Programming with Ada and Lower\_Layer***  
 Tipo de participación: *Ponencia*  
 Congreso: *TRI-Ada'97 Conference*  
 Publicación: *Proceedings of the TRI-Ada'97 Conference, pag. 105-110, ISBN 0-89791-981-5*  
 Lugar de celebración: *St. Louis, Missouri, EE.UU.*  
 Fecha: *November 1997*

---

Autores: *Jesús M. González Barahona, Pedro de las Heras Quirós, José Centeno González, Luis López Fernández, Francisco J. Ballesteros Cámara*

Título: ***Reuse and Modularity Applied to Protocol Programming: the Simple\_Com system***

Tipo de participación: *Ponencia*

Congreso: *VIII Jornadas de Paralelismo*

Publicación: *Actas de las VIII Jornadas de Paralelismo*

Lugar de celebración: *Cáceres, España*

Fecha: *Septiembre 1997*

---

Autores: *Luis L. Fernández, Francisco J. Ballesteros, José Centeno González, Jesús M. González Barahona, Pedro de la Heras Quirós*

Título: ***Advice: A Framework For Distributed Object Management***

Tipo de participación: *Ponencia*

Congreso: *VIII Jornadas de Paralelismo*

Publicación: *Actas de las VIII Jornadas de Paralelismo*

Lugar de celebración: *Cáceres, España*

Fecha: *Septiembre 1997*

---

Autores: *Jesús M. González Barahona, José Centeno González, Pedro de las Heras Quirós*

Título: ***Overview of the Simple\_Com System***

Tipo de participación: *Ponencia*

Congreso: *VI Jornadas de Concurrencia*

Publicación: *Actas de las VI Jornadas de Concurrencia*

Lugar de celebración: *Vigo, España*

Fecha: *Junio 1997*

---

Autores: *Jesús M. González Barahona, Pedro de las Heras Quirós, José Centeno González*

Título: ***A development environment for Ada programming***

Tipo de participación: *Ponencia*

Congreso: *Sexta Jornada Técnica AdaSpain*

Publicación: *Actas de la sexta jornada técnica AdaSpain*

Lugar de celebración: *Madrid, España*

Fecha: *Febrero, 1997*

---

Autores: *Pedro de las Heras Quirós, Jesús M. González Barahona, José Centeno González*

Título: ***From Global Observations to Causal Delivery Without Clocks***

Tipo de participación: *Ponencia*

Congreso: *1st. European Research Seminar on Advances in Distributed Systems*

Publicación: *Proceedings of the 1st. European Research Seminar on Advances in Distributed Systems*

Lugar de celebración: *Alpe D'Huez, Francia*

Fecha: *1995*

---

Autores: *Jesús M. González Barahona, Pedro de las Heras Quirós, José Centeno González, Luis López Fernández*

Título: ***Using WWW and Friends to Teach Computer Science***

Tipo de participación: *Ponencia*

Congreso: *WWW National Conference*

Publicación: *Proceedings of the WWW National Conference*

Lugar de celebración: *Braga, Portugal*

Fecha: *1995*

---

Autores: *Pedro de las Heras Quirós. Jesús M. González Barahona. José Centeno González*

Título: *¿Es posible explicar la entrega causal sin relojes?*

Tipo de participación: *Ponencia*

Congreso: *IV Jornadas de Concurrency*

Publicación: *Actas de las IV Jornadas de Concurrency*

Lugar de celebración: *El Escorial*

Fecha: *Madrid Junio 1995*

---

Autores: *Jesús M. González Barahona*

Título: ***Dynamic Distributed Scheduling of Real Time Tasks***

Tipo de participación: *Exposición de trabajo en curso*

Congreso: *Fifth European Workshop on Dependable Computing*

Publicación:

Lugar de celebración: *Lisboa, Portugal*

Fecha: *1993*

---

## Capítulo 25

# Tesis doctorales dirigidas

---

Título: *Global and Geographically Distributed Work Teams: Understanding the Bug Fixing Process and Potentially Bug-prone Activity Patterns (tesis con mención de doctorado europeo)*

Doctorando: *Daniel Izquierdo Cortázar*

Universidad: *Universidad Rey Juan Carlos*

Facultad/escuela: *Escuela Técnica Superior de Ingeniería de Telecomunicación*

Fecha: *27 de abril de 2012*

---

Título: *Temporal and behavioral patterns in the use of Wikipedia*

Doctorando: *Antonio José Reinoso Peinado*

Universidad: *Universidad Rey Juan Carlos*

Facultad/escuela: *Escuela Técnica Superior de Ingeniería de Telecomunicación*

Fecha: *30 de septiembre de 2011*

---

Título: *Wikipedia: A Quantitative Analysis*

Doctorando: *Felipe Ortega*

Universidad: *Universidad Rey Juan Carlos*

Facultad/escuela: *Escuela Técnica Superior de Ingeniería de Telecomunicación*

Fecha: *1 de abril de 2009*

---

Título: *A statistical examination of the evolution and properties of libre software (tesis con mención de doctorado europeo)*

Doctorando: *Israel Herraiz*

Universidad: *Universidad Rey Juan Carlos*

Facultad/escuela: *Escuela Técnica Superior de Ingeniería de Telecomunicación*

Fecha: *24 de octubre de 2008*

---

Título: *Empirical Software Engineering Research on Libre Software: Data Sources, Methodologies and Results (tesis con mención de doctorado europeo)*

Doctorando: *Gregorio Robles*

Universidad: *Universidad Rey Juan Carlos*

Facultad/escuela: *Escuela Superior de Ciencias Experimentales y Tecnología*

Fecha: *9 de febrero de 2006*

---

Título: *Programación de sistemas distribuidos en entornos de redes fraccionables*

Doctorando: *Pedro de las Heras Quirós*

Universidad: *Universidad Politécnica de Madrid*

Facultad/escuela: *Facultad de Informática*

Fecha: *27 de septiembre de 2004*

---





## Capítulo 26

# Participación en comités y grupos técnicos

---

Título: *CHAOSS Collaborative Project*  
Entidad de la que depende: *Linux Foundation*  
Tema: *Founder and co-chair of the Governing Board*  
Fecha: *2017-*

---

Título: *MSR Steering Committee*  
Entidad de la que depende: *International Conference on Mining Software Repositories*  
Tema: *Member*  
Fecha: *2015-*

---

Título: *Patronato de IMDEA Software*  
Entidad de la que depende: *IMDEA Software*  
Tema: *Representante de la URJC*  
Fecha: *2012-*

---

Título: *Consejo asesor de la Fundación CENATIC*  
Entidad de la que depende: *CENATIC*  
Tema: *Asesoría en temas relacionados con software libre, desarrollo y Universidades*  
Fecha: *2009-2012*

---

Título: *NESSI Working Group on Open Source Software*  
Entidad de la que depende: *Plataforma europea NESSI*  
Tema: *Aspectos del software libre importantes para software y servicios*  
Fecha: *2008-2011*

---

Título: *Executive Board of the Morfeo Community*  
Entidad de la que depende: *Morfeo Community*  
Tema: *Gestión de la comunidad de desarrollo de software libre Morfeo*  
Fecha: *2008-2013*

---

Título: *Coordination cluster on FLOSS Quality*

Entidad de la que depende: *Promoted by the European Commission (IST)*

Tema: *Coordinación entre proyectos sobre calidad y software libre*

Fecha: *2006-2009*

---

Título: *Grupo de trabajo sobre archivos abiertos de publicación científica*

Entidad de la que depende: *Promovido por la Comunidad de Madrid*

Tema: *Asesoría en temas técnicos relacionados con la publicación abierta*

Fecha: *2006-2008*

---

Título: *Gretel*

Entidad de la que depende: *Asociación de Ingenieros de Telecomunicación*

Tema: *Asesoramiento en política de telecomunicaciones*

Fecha: *2003-2008*

---

Título: *Comité asesor de RedIris*

Entidad de la que depende: *RedIris*

Tema: *Asesoramiento técnico sobre cuestiones relacionadas con RedIris, la red de datos académica española*

Fecha: *2002-2004*

---

Título: *European Working Group on Free Software*

Entidad de la que depende: *DG-INFO, Comisión Europea (entidad promotora)*

Tema: *Software libre*

Fecha: *1999-2002*

---

## Capítulo 27

# Estancias en Centros extranjeros

Clave: D = doctorado, P = postdoctoral, I = invitado, C = contratado, O = otras (especificar).

---

Centro: *Computer Science Department, Cornell University*

Localidad: *Ithaca, NY* País: *EEUU* Fecha: *Verano 1998* Duración (semanas): *5*

Tema: *Sistemas de comunicación multienvío tolerante a fallos para grupos de procesos* Clave:  
*P*

---



## Capítulo 28

# Experiencia en organización de actividades de I+D

---

Título: *14th International Conference on Open Source Systems (IFIP Conference)*  
Tipo de actividad: *Program Co-chair* Ámbito: *Internacional*  
Fecha: *Athens, Greece, Junio 2018*

---

Título: *International Conference on Mining Software Repositories (MSR 2017)*  
Tipo de actividad: *General Chair* Ámbito: *Internacional*  
Fecha: *Buenos Aires, Argentina, Mayo 2017*

---

Título: *2014 Joint International Symposium on Open Collaboration (OpenSym 2014)*  
Tipo de actividad: *Free, Libre, and Open Source Research Track Co-Chair* Ámbito: *Internacional*  
Fecha: *Berlin, Alemania, Agosto 2014*

---

Título: *2013 Joint International Symposium on Open Collaboration (OpenSym 2013)*  
Tipo de actividad: *Free, Libre, and Open Source Research Track Co-Chair* Ámbito: *Internacional*  
Fecha: *Hong Kong, China, Agosto 2013*

---

Título: *Workshop on Public Data about Software Development (WoPDaSD 2010)*  
Tipo de actividad: *Co-chair of organizing and program committees* Ámbito: *Internacional*  
Fecha: *Notre Dame, IN, EE.UU., Junio 2010*

---

Título: *OSS Doctoral Symposium 2010*  
Tipo de actividad: *Member of the organizing and program committees* Ámbito: *Internacional*  
Fecha: *Notre Dame, IN, EE.UU., Junio 2010*

---

Título: *Sixth International Conference on Open Source Systems (IFIP Conference)*  
Tipo de actividad: *Co-chair* Ámbito: *Internacional*  
Fecha: *Notre Dame, IN, EE.UU., Junio 2010*

---

Título: *Mini-Conference on Mining Software Repositories (MSR 2010)*  
Tipo de actividad: *Member of the program committees* Ámbito: *Internacional*  
Fecha: *Cape Town, Suráfrica, Mayo 2010*

---

Título: *Hawaii International Conference on Computer Systems (HICCS 2010)*  
Tipo de actividad: *Member of the program committee* Ámbito: *Internacional*  
Fecha: *Hawaii, United States, January 2010*

---

Título: *Workshop on Public Data about Software Development (WoPDaSD 2008)*

Tipo de actividad: *Co-chair of organizing and program committees* Ámbito: *Internacional*

Fecha: *Skövde, Suecia, Junio 2009*

---

Título: *OSS Doctoral Symposium 2010*

Tipo de actividad: *Member of the organizing and program committees* Ámbito: *Internacional*

Fecha: *Skövde, Suecia, Junio 2009*

---

Título: *Fifth International Conference on Open Source Systems (IFIP Conference)*

Tipo de actividad: *Member of the program committee* Ámbito: *Internacional*

Fecha: *Skövde, Suecia, Junio 2009*

---

Título: *Mini-Conference on Mining Software Repositories (MSR 2009)*

Tipo de actividad: *Member of the program committees* Ámbito: *Internacional*

Fecha: *Leipzig, Alemania, Mayo 2009*

---

Título: *Hawaii International Conference on Computer Systems (HICCS 2009)*

Tipo de actividad: *Member of the program committee* Ámbito: *Internacional*

Fecha: *Hawaii, United States, January 2009*

---

Título: *Workshop on Public Data about Software Development (WoPDaSD 2008)*

Tipo de actividad: *Co-chair of organizing and program committees* Ámbito: *Internacional*

Fecha: *Milano, Italia, September 2008*

---

Título: *Fourth International Conference on Open Source Systems (IFIP Conference)*

Tipo de actividad: *Member of the program committee* Ámbito: *Internacional*

Fecha: *Milano, Italia, September 2008*

---

Título: *Research Room @ FOSDEM*

Tipo de actividad: *Co-chair of organizing and program committees* Ámbito: *Internacional*

Fecha: *Brussels, Belgium, 23-24 February 2008*

---

Título: *Hawaii International Conference on Computer Systems (HICCS 2008)*

Tipo de actividad: *Member of the program committee* Ámbito: *Internacional*

Fecha: *Hawaii, United States, January 2008*

---

Título: *Workshop on Public Data about Software Development (WoPDaSD 2007)*

Tipo de actividad: *Co-chair of organizing and program committees* Ámbito: *Internacional*

Fecha: *Limerick, Irlanda, 14 July 2007*

---

Título: *Third International Conference on Open Source Systems (IFIP Conference)*

Tipo de actividad: *Member of the program committee* Ámbito: *Internacional*

Fecha: *Limerick, Irlanda, 11-14 July 2007*

---

Título: *Intl. Conf. on Software and Data Technologies (ICSOFT 2007)*

Tipo de actividad: *Member of the program committee* Ámbito: *Internacional*

Fecha: *Barcelona, Spain, June 2007*

---

Título: *Research Room @ FOSDEM*

Tipo de actividad: *Co-chair of organizing and program committees* Ámbito: *Internacional*

Fecha: *Brussels, Belgium, 24-25 February 2007*

---

Título: *Hawaii International Conference on Computer Systems (HICCS 2007)*

Tipo de actividad: *Member of the program committee* Ámbito: *Internacional*

Fecha: *Hawaii, United States, January 2007*

---

Título: **Workshop on Public Data about Software Development (WoPDaSD 2006)**  
 Tipo de actividad: *Co-chair of organizing and program committees* Ámbito: *Internacional*  
 Fecha: *Como, Italy, 10 June 2006*

---

Título: **Workshop on Free and Open Source Learning Environments and Tools (FOSLET 2006)**  
 Tipo de actividad: *Member of the program committee* Ámbito: *Internacional*  
 Fecha: *Como, Italy, 10 June 2006*

---

Título: **First International Conference on Open Source Systems**  
 Tipo de actividad: *Chair of the "Open Educational Symposium" and program committee member*  
 Ámbito: *Internacional*  
 Fecha: *Genoa, Italy, 11-15 July 2005*

---

Título: **9th International Conference on Reliable Software Technologies**  
 Tipo de actividad: *Miembro del comité de programa* Ámbito: *Internacional*  
 Fecha: *Palma de Mallorca, España, 14-18 junio de 2004*

---

Título: **VI Congreso Hispalinux**  
 Tipo de actividad: *Miembro del comité de organización* Ámbito: *Nacional*  
 Fecha: *Universidad Rey Juan Carlos, Móstoles, septiembre 2003*

---

Título: **8th International Conference on Reliable Software Technologies**  
 Tipo de actividad: *Miembro del comité de programa* Ámbito: *Internacional*  
 Fecha: *Toulouse, Francia, 16-20 junio de 2003*

---

Título: **Jornadas sobre ingeniería del software libre de la URJC**  
 Tipo de actividad: *Coordinador del comité organizador* Ámbito: *Internacional*  
 Fecha: *2002 en adelante (periodicidad aproximada: cada cuatro meses)*

---

Título: **V Congreso Hispalinux**  
 Tipo de actividad: *Coordinador del comité de programa* Ámbito: *Nacional*  
 Fecha: *Universidad Autónoma de Madrid, Madrid, noviembre 2002*

---

Título: **7th International Conference on Reliable Software Technologies**  
 Tipo de actividad: *Miembro del comité de programa* Ámbito: *Internacional*  
 Fecha: *Viena, Austria, 17-21 junio de 2002*

---

Título: **IV Congreso Hispalinux**  
 Tipo de actividad: *Miembro del comité organizador, coordinador del comité de programa* Ámbito: *Nacional*  
 Fecha: *Universidad Politécnica de Madrid, Madrid, noviembre 2001*

---

Título: **6th International Conference on Reliable Software Technologies**  
 Tipo de actividad: *Miembro del comité de programa* Ámbito: *Internacional*  
 Fecha: *Leuven, Bélgica, 14-18 mayo de 2001*

---

Título: **DISC 2000**  
 Tipo de actividad: *Miembro del comité organizador* Ámbito: *Internacional*  
 Fecha: *Toledo, 4-6 octubre de 2000*

---

Título: **III Congreso Hispalinux**  
 Tipo de actividad: *Miembro del comité organizador, coordinador del comité de programa* Ámbito: *Nacional*  
 Fecha: *Universidad Carlos III de Madrid, Leganés, noviembre 2000*

---

Título: ***IX Jornada Técnica de Ada-Spain***

Tipo de actividad: *Miembro del comité organizador, coordinador del comité de programa* Ámbito: *Nacional*

Fecha: *Universidad Rey Juan Carlos, Móstoles, 22 de junio de 2000*

---

Título: ***II Congreso Hispalinux***

Tipo de actividad: *Miembro del comité organizador, coordinador del comité de programa* Ámbito: *Nacional*

Fecha: *Universidad Carlos III de Madrid, Leganés, noviembre 1999*

---

Título: ***Jornadas sobre Software Libre***

Tipo de actividad: *Miembro del comité organizador* Ámbito: *Nacional*

Fecha: *Universidad Carlos III de Madrid, Leganés, abril 1999*

---

Título: ***Jornadas sobre Sistemas de Tiempo Real, Sistemas Distribuidos y Tolerancia a Fallos***

Tipo de actividad: *Miembro del comité organizador* Ámbito: *Nacional*

Fecha: *Universidad Carlos III de Madrid, Leganés, 26 de junio, 10 y 20 de octubre de 1995*

---

Título: ***Jornadas sobre software de libre distribución***

Tipo de actividad: *Miembro del comité organizador* Ámbito: *Nacional*

Fecha: *Universidad Carlos III de Madrid, Leganés, 14 y 15 de septiembre de 1995*

---



## Capítulo 29

# Otros méritos o aclaraciones que se desee hacer constar

---

### 29.1. Tesis doctoral

Lectura de la tesis doctoral “Arquitectura de comunicaciones para grupos de procesos replicados”, en febrero de 1998, para la consecución del título de Doctor Ingeniero de Telecomunicación por la Universidad Politécnica de Madrid. Calificada con Apto cum Laude por unanimidad del tribunal calificador.

### 29.2. Publicaciones docentes

- “Introducción al software libre”. Jesús M. González Barahona, Joaquín Seoane Pascual, Gregorio Robles. Universitat Oberta de Catalunya. Barcelona, 2003. 180 páginas.

### 29.3. Actividades relacionadas con la docencia

- Propuesta de modelo de plan de estudios para un Master Europeo en Software Libre. Coordinador del grupo MoLOS, que lo propone, con representantes de varias Universidades europeas.
- Proyecto “Diseño de unos estudios de master sobre software libre”, financiado por la Comunidad de Madrid, dentro de su programa de ayudas para la elaboración de titulaciones de postgrado con Universidades de la Unión Europea. Coordinador del proyecto. Participación de la URJC, UPM, Universidade de Porto (PT) y École Nationale Supérieure Électronique, Informatique et Radiocommunications de Bordeaux (FR), Marzo de 2003 a Febrero de 2004.

### 29.4. Participación en el diseño de planes de estudios

- Plan de estudios de las titulaciones de grado de Ingeniería de Telecomunicación, Universidad Rey Juan Carlos. Participante en los comités de diseño inter-departamento. Redacción de planes en marcha en diciembre de 2007.
- Plan de estudios del Master sobre Software Libre, título propio de la Universidad Rey Juan Carlos. Director del programa. Puesto en marcha en 2007.
- Plan de estudios del Master Oficial en Software Libre, de la Universidad Oberta de Catalunya. Miembro del comité científico. Puesto en marcha en 2005.

- Plan de estudios de Ingeniería de Telecomunicación, Universidad Rey Juan Carlos. Miembro de los comités redactor y de diseño. Primer plan de la titulación, puesto en marcha en 2003.
- Plan de estudios de Ingeniería Informática, Universidad Rey Juan Carlos. Miembro del comité de diseño. Primer plan de la titulación, puesto en marcha en 1999.
- Plan de estudios de Ingenierías Técnicas en Informática de Gestión y de Sistemas. Participación en el proceso de diseño. Primera revisión del plan de las titulaciones. Puestos en marcha en 2001.
- Plan de estudios de Ingeniería Informática, Universidad Carlos III. Participación en el proceso de diseño. Primer plan de la titulación, puesto en marcha en 1998.
- Plan de estudios de Ingeniería Técnicas en Informática de Gestión. Participación en el proceso de diseño. Primera revisión del plan de la titulación. Puestos en marcha en 1995.

## 29.5. Enseñanza reglada impartida

- Curso 2017-2018 (URJC):
  - *Información Audiovisual en Redes de Ordenadores*, Ingeniería de Telecomunicación.
  - *Servicios y Aplicaciones en Redes de Ordenadores*, Grado en Ingeniería Telemática.
  - *Servicios y Aplicaciones Telemáticas*, Grado en Ingenierías Tecnológicas de Telecomunicación.
- Curso 2016-2017 (URJC):
  - *Información Audiovisual en Redes de Ordenadores*, Ingeniería de Telecomunicación.
  - *Aplicaciones Telemáticas*, Grado en Ingeniería Telemática.
  - *Desarrollo de Aplicaciones Telemáticas*, Grado en Ingenierías Tecnológicas de Telecomunicación.
  - *Servicios y Aplicaciones en Redes de Ordenadores*, Grado en Ingeniería Telemática.
  - *Servicios y Aplicaciones Telemáticas*, Grado en Ingenierías Tecnológicas de Telecomunicación.
- Curso 2015-2016 (URJC):
  - *Información Audiovisual en Redes de Ordenadores*, Ingeniería de Telecomunicación.
  - *Aplicaciones Telemáticas*, Grado en Ingeniería Telemática.
  - *Desarrollo de Aplicaciones Telemáticas*, Grado en Ingenierías Tecnológicas de Telecomunicación.
  - *Servicios y Aplicaciones en Redes de Ordenadores*, Grado en Ingeniería Telemática.
  - *Servicios y Aplicaciones Telemáticas*, Grado en Ingenierías Tecnológicas de Telecomunicación.
- Curso 2014-2015 (URJC):
  - *Información Audiovisual en Redes de Ordenadores*, Ingeniería de Telecomunicación.
  - *Aplicaciones Telemáticas*, Grado en Ingeniería Telemática.
  - *Desarrollo de Aplicaciones Telemáticas*, Grado en Ingenierías Tecnológicas de Telecomunicación.
  - *Servicios y Aplicaciones en Redes de Ordenadores*, Grado en Ingeniería Telemática.
  - *Servicios y Aplicaciones Telemáticas*, Grado en Ingenierías Tecnológicas de Telecomunicación.
  - *Servicios y Aplicaciones Telemáticas*, Ingeniería de Telecomunicación.
- Curso 2013-2014 (URJC):
  - *Información Audiovisual en Redes de Ordenadores*, Ingeniería de Telecomunicación.
  - *Aplicaciones Telemáticas*, Grado en Ingeniería Telemática.
  - *Desarrollo de Aplicaciones Telemáticas*, Grado en Ingenierías Tecnológicas de Telecomunicación.

- *Servicios y Aplicaciones en Redes de Ordenadores*, Grado en Ingeniería Telemática.
  - *Servicios y Aplicaciones Telemáticas*, Grado en Ingeniería Tecnologías de Telecomunicación.
  - *Servicios y Aplicaciones Telemáticas*, Ingeniería de Telecomunicación.
  - *Introducción al Software Libre*, Máster Oficial en Software Libre
  - *Economía del Software Libre*, Máster Oficial en Software Libre
  - *Software libre*, Master Oficial en Sistemas Telemáticos e Informáticos.
- Curso 2012-2013 (URJC):
    - *Información Audiovisual en Redes de Ordenadores*, Ingeniería de Telecomunicación.
    - *Aplicaciones Telemáticas*, Grado en Ingeniería Telemática.
    - *Desarrollo de Aplicaciones Telemáticas*, Grado en Ingeniería Tecnologías de Telecomunicación.
    - *Servicios y Aplicaciones en Redes de Ordenadores*, Grado en Ingeniería de Sistemas de Telecomunicación.
    - *Servicios y Aplicaciones Telemáticas*, Grado en Ingeniería Tecnologías de Telecomunicación.
    - *Servicios y Aplicaciones Telemáticas*, Ingeniería de Telecomunicación.
    - *Introducción al Software Libre*, Máster Oficial en Software Libre
    - *Economía del Software Libre*, Máster Oficial en Software Libre
    - *Software libre*, Master Oficial en Sistemas Telemáticos e Informáticos.
  - Curso 2011-2012 (URJC):
    - *Información Audiovisual en Redes de Ordenadores*, Ingeniería de Telecomunicación.
    - *Servicios y Aplicaciones Telemáticas*, Ingeniería de Telecomunicación.
    - *Servicios y Aplicaciones en Redes de Ordenadores*, Grado en Ingeniería Telemática.
    - *Introducción al Software Libre*, Máster Oficial en Software Libre
    - *Economía del Software Libre*, Máster Oficial en Software Libre
    - *Software libre*, Master Oficial en Sistemas Telemáticos e Informáticos.
  - Curso 2010-2011 (URJC):
    - *Información Audiovisual en Redes de Ordenadores*, Ingeniería de Telecomunicación.
    - *Servicios y Aplicaciones Telemáticas*, Ingeniería de Telecomunicación.
    - *Tecnologías de Servicios y Aplicaciones en Internet*, Ingeniería Informática.
    - *Protocolos de Comunicaciones y Servicios en Sistemas Móviles*, Master Oficial en Redes y Servicios de Comunicación Móvil.
    - *Software libre*, Master Oficial en Sistemas Telemáticos e Informáticos.
    - *Cultura libre*, Programa Ada-Madrid (asignaturas de libre elección comunes a todas las Universidades públicas de la Comunidad de Madrid).
  - Curso 2009-2010 (URJC):
    - *Introducción a la Ingeniería de Telecomunicación*, Ingeniería de Telecomunicación.
    - *Información Audiovisual en Redes de Ordenadores*, Ingeniería de Telecomunicación.
    - *Servicios y Aplicaciones Telemáticas*, Ingeniería de Telecomunicación.
    - *Tecnologías de Servicios y Aplicaciones en Internet*, Ingeniería Informática.
    - *Protocolos de Comunicaciones y Servicios en Sistemas Ubicuos II*, Master Oficial en Redes y Servicios de Comunicación Móvil.
    - *Software libre*, Master Oficial en Sistemas Telemáticos e Informáticos.

- *Cultura libre*, Programa Ada-Madrid (asignaturas de libre elección comunes a todas las Universidades públicas de la Comunidad de Madrid).
- Curso 2008-2009 (URJC):
    - *Introducción a la Ingeniería de Telecomunicación*, Ingeniería de Telecomunicación.
    - *Información Audiovisual en Redes de Ordenadores*, Ingeniería de Telecomunicación.
    - *Servicios y Aplicaciones Telemáticas*, Ingeniería de Telecomunicación.
    - *Tecnologías de Servicios y Aplicaciones en Internet*, Ingeniería Informática.
    - *Aplicaciones y Protocolos de Transporte sobre Redes Móviles*, Master Oficial en Redes y Servicios de Comunicación Móvil.
    - *Redes de Ordenadores sobre Redes Inalámbricas*, Master Oficial en Redes y Servicios de Comunicación Móvil.
    - *Software libre*, Master Oficial en Sistemas Telemáticos e Informáticos.
    - *Cultura libre*, Programa Ada-Madrid (asignaturas de libre elección comunes a todas las Universidades públicas de la Comunidad de Madrid).
  - Curso 2007-2008 (URJC):
    - *Introducción a la Ingeniería de Telecomunicación*, Ingeniería de Telecomunicación.
    - *Información Audiovisual en Redes de Ordenadores*, Ingeniería de Telecomunicación.
    - *Redes de Ordenadores Móviles*, Ingeniería de Telecomunicación.
    - *Servicios y Aplicaciones Telemáticas*, Ingeniería de Telecomunicación.
    - *Tecnologías de Servicios y Aplicaciones en Internet*, Ingeniería Informática.
    - *Aplicaciones y Protocolos de Transporte sobre Redes Móviles*, Master Oficial en Redes y Servicios de Comunicación Móvil.
    - *Redes de Ordenadores sobre Redes Inalámbricas*, Master Oficial en Redes y Servicios de Comunicación Móvil.
    - *Software libre*, Master Oficial en Sistemas Telemáticos e Informáticos.
    - *Cultura libre*, Programa Ada-Madrid (asignaturas de libre elección comunes a todas las Universidades públicas de la Comunidad de Madrid).
  - Curso 2006-2007 (URJC):
    - *Introducción a la Ingeniería de Telecomunicación*, Ingeniería de Telecomunicación.
    - *Información Audiovisual en Redes de Ordenadores*, Ingeniería de Telecomunicación.
    - *Redes de Ordenadores Móviles*, Ingeniería de Telecomunicación.
    - *Servicios y Aplicaciones Telemáticas*, Ingeniería de Telecomunicación.
    - *Tecnologías de Servicios y Aplicaciones en Internet*, Ingeniería Informática.
    - *Aplicaciones y Protocolos de Transporte sobre Redes Móviles*, Master Oficial en Redes y Servicios de Comunicación Móvil.
    - *Redes de Ordenadores sobre Redes Inalámbricas*, Master Oficial en Redes y Servicios de Comunicación Móvil.
    - *Software libre*, Master Oficial en Sistemas Telemáticos e Informáticos (en coordinación con el Programa de Doctorado del Departamento de Ingeniería de Sistemas Telemáticos de la UPM).
  - Curso 2005-2006 (URJC):
    - *Introducción a la Ingeniería de Telecomunicación*, Ingeniería de Telecomunicación.
    - *Arquitectura de Redes de Ordenadores*, Ingeniería de Telecomunicación.

- *Información Audiovisual en Redes de Ordenadores*, Ingeniería de Telecomunicación.
  - *Tecnologías de Servicios y Aplicaciones en Internet*, Ingeniería Informática.
  - *Algoritmos Avanzados en Redes*, Doctorado en Informática y Modelización Matemática.
  - *Software libre*, Doctorado en Informática y Modelización Matemática.
- Curso 2004-2005 (URJC):
- *Introducción a la Ingeniería de Telecomunicación*, Ingeniería de Telecomunicación.
  - *Arquitectura de Redes de Ordenadores*, Ingeniería de Telecomunicación.
  - *Tecnologías de Servicios y Aplicaciones en Internet*, Ingeniería Informática.
  - *Informática Básica*, Licenciatura en Comunicación Audiovisual.
  - *Sistemas Informáticos Ubícuos*, Doctorado en Informática y Modelización Matemática.
  - *Software libre*, Doctorado en Informática y Modelización Matemática.
- Curso 2003-2004 (URJC):
- *Introducción a la Ingeniería de Telecomunicación*, Ingeniería de Telecomunicación.
  - *Arquitectura de Redes de Ordenadores*, Ingeniería de Telecomunicación.
  - *Tecnologías de Servicios y Servicios en Internet*, Ingeniería Informática.
  - *Informática Básica*, Licenciatura en Comunicación Audiovisual.
  - *Sistemas Informáticos Ubícuos*, Doctorado en Informática y Modelización Matemática.
  - *Software libre*, Doctorado en Informática y Modelización Matemática.
- Curso 2002-2003 (URJC):
- *Redes II*, Ingeniería Informática.
  - *Arquitecturas Distribuidas en Redes Heterogéneas*, Ingeniería Informática.
  - *Tecnologías de Servicios y Aplicaciones en Internet*, Ingeniería Informática.
  - *Informática Básica*, Licenciatura en Comunicación Audiovisual.
  - *Servicios de Información en Internet*, Licenciatura en Comunicación Audiovisual.
  - *Software libre*, Doctorado en Informática y Modelización Matemática.
- Curso 2001-2002 (URJC):
- *Redes*, Ingeniería Técnica en Informática de Sistemas.
  - *Redes II*, Ingeniería Informática.
  - *Informática Básica*, Licenciatura en Comunicación Audiovisual.
  - *Software libre*, Doctorado en Informática y Modelización Matemática.
- Curso 2000-2001 (URJC):
- *Redes*, de Ingeniería Técnica en Informática de Sistemas.
  - *Redes II*, de Ingeniería Informática.
  - *Introducción a la Informática*, de la Licenciatura en Publicidad y Relaciones Públicas.
- Curso 1999-2000 (URJC):
- *Ampliación de Sistemas Operativos*, de Ingeniería Técnica en Informática de Gestión y de Sistemas.
  - *Aplicaciones informáticas*, de la Diplomatura en Fisioterapia.

- Curso 1998-1999 (UCAR):
  - *Redes de Ordenadores*, de Ingeniería Técnica en Informática de Gestión.
  - *Programación de Ordenadores*, de Ingeniería Técnica en Electricidad.
  - *Programación de Ordenadores*, de Ingeniería Técnica en Mecánica.
  - *Implicaciones económicas, legales y sociales del Software Libre*, en el programa de Cursos de Humanidades de la Universidad Carlos III.
  - *Sistemas Distribuidos*, en el programa de Doctorado en Ingeniería Informática de la Universidad Carlos III.
  - *Sistemas Distribuidos*, de Ingeniería Técnica en Informática de Gestión.
- Curso 1997-1998 (UCAR):
  - *Sistemas Operativos*, de Ingeniería Superior de Telecomunicación.
  - *Sistemas Distribuidos*, de Ingeniería Técnica en Informática de Gestión.
  - *Transmisión de Datos y Redes*, de Ingeniería Técnica en Informática de Gestión.
  - *Programación de Ordenadores*, de Ingeniería Técnica en Electricidad.
  - *Sistemas Distribuidos*, en el programa de Doctorado en Ingeniería Informática de la Universidad Carlos III.
  - *Sistemas Operativos y Transmisión de Datos*, en el programa de Doctorado en Ingeniería Informática de la Universidad Carlos III.
- Curso 1996-1997 (UCAR):
  - *Transmisión de Datos y Redes*, de Ingeniería Técnica en Informática de Gestión.
  - *Sistemas Operativos*, de Ingeniería Técnica en Informática de Gestión.
  - *Sistemas Distribuidos*, de Ingeniería Técnica en Informática de Gestión.
  - *Sistemas Tolerantes a Fallos*, de Ingeniería Técnica en Informática de Gestión.
- Curso 1995-1996 (UCAR):
  - *Transmisión de Datos y Redes*, de Ingeniería Técnica en Informática de Gestión.
  - *Sistemas Distribuidos*, de Ingeniería Técnica en Informática de Gestión.
  - *Sistemas Tolerantes a Fallos*, de Ingeniería Técnica en Informática de Gestión.
- Curso 1994-1995 (UCAR):
  - *Sistemas Operativos*, de Ingeniería Técnica en Informática de Gestión.
  - *Sistemas Distribuidos*, de Ingeniería Técnica en Informática de Gestión.
  - *Transmisión de Datos y Redes*, de Ingeniería Técnica en Informática de Gestión.
- Curso 1993-1994 (UCAR):
  - *Sistemas Operativos*, de Ingeniería Técnica en Informática de Gestión.
  - *Sistemas Distribuidos*, de Ingeniería Técnica en Informática de Gestión.
  - *Redes de Ordenadores*, de Ingeniería Técnica en Informática de Gestión.
- Curso 1992-1993 (UCAR):
  - *Sistemas Operativos*, de Ingeniería Técnica en Informática de Gestión.
  - *Sistemas Distribuidos*, de Ingeniería Técnica en Informática de Gestión.
  - *Redes de Ordenadores*, de Ingeniería Técnica en Informática de Gestión.
- Curso 1991-1992 (UCAR):
  - *Estructura de Computadores*, de Ingeniería Técnica en Informática de Gestión.

## 29.6. Grupos de trabajo

- Participación en el CHAOSS Collaborative Project, organizado por la Linux Foundation, sobre métricas de desarrollo de software, desde su fundación en 2017.
- Participación en el proyecto GrimoireLab, para la creación de un conjunto de herramientas para la analítica de desarrollo software desde su fundación en 2015.
- Participación en el grupo asesor del CENATIC sobre certificación y software libre, desde octubre de 2007.
- Coordinador del grupo sobre software libre de la plataforma tecnológica nacional INES, desde 2006.
- Miembro del Working Group 2.13 (Open Source) de IFIP (Information Federation for Information Processing), desde 2006
- Participación en el grupo técnico asesor sobre publicaciones científicas abiertas promovido por la Dirección General de Investigación, Comunidad de Madrid, desde 2006.
- Participación en el grupo de diseño del IMDEA Software (Instituto Madrileño de Estudios Avanzados) durante 2006.
- Participación en el grupo AEN CTN71/GT22 de AENOR, sobre interoperabilidad con GNU/Linux, desde 2005.
- Participación en el grupo MoLOS (Master on Libre and Open Source Software), que propone un modelo de master europeo sobre software libre.
- Participación en el Grupo Gretel, sobre política tecnológica, de la Asociación de Ingenieros de Telecomunicación, desde 2003, como experto independiente.
- Participación en el Comité Científico del Master en Software Libre de la Universidad Oberta de Catalunya desde su fundación, en 2003.
- Participación en el Comité Asesor de la Semana de la Ciencia, Dirección General de Investigación, Comunidad de Madrid, en 2004.
- Participación en el Comité Asesor de RedIris (red de comunicaciones académica española) desde 2002 hasta 2004, como experto independiente.
- Participación en el grupo de trabajo de RedIris sobre transmisión de multimedia y MBONE (MEDIA), desde 1997 hasta 2000.
- Participación en el grupo de trabajo sobre software libre promovido por la Comisión Europea (<http://eu.conecta.it>) de 1999 a 2001.
- Participación, desde su fundación (1991), hasta 2000, en el GSOD (Grupo sobre Sistemas Operativos Distribuidos), dirigido por Angel Alvarez y Sergio Arévalo en la Universidad Politécnica de Madrid.
- Participación, desde su fundación (1997), en el grupo Sobre, dedicado al estudio y promoción del software libre (y sucesor del grupo PDSOFT, con similares fines, también desde su fundación, en 1992).

## 29.7. Becas y Premios

- Mining Challenge Award, en la categoría “Change Prediction” del Mining Challenge 2007 del 4th International Workshop on Mining Software Repositories (conjunto con Israel Herraiz y Gregorio Robles)
- Best Paper Award del Mining Software Repositories Workshop, por el artículo “Mining Large Software Compilations over Time: Another Perspective of Software Evolution”, Shanghai, China, Mayo de 2006.

- Beca de la Fundación Universidad Carlos III de Madrid para la realización de estancias en centros de investigación extranjeros (Cornell University, EEUU), Julio-Agosto 1998.
- Beca de CABERNET (Europeana Network of Excellence on Distributed Systems) para la asistencia al Cabernet Radicals, Espinho, Portugal, Julio 1998.

## 29.8. Periodos de docencia e investigación reconocidos

- Reconocidos tres sexenios de investigación por la Comisión Nacional Evaluadora de la Actividad Investigadora (1995-2001, 2005-2010 y 2011-2016).
- Reconocidos cuatro quinquenios de actividad docente (1992-1999, 1999-2004, 2005-2009 y 2010-2014).

## 29.9. Comités de publicaciones científicas

- Miembro del Editorial Advisory Board del International Journal of Open Source Software and Processes, IGI Global, desde su fundación hasta 2017.

## 29.10. Sociedades Científicas y Profesionales

- Miembro del IEEE y de su Computer Society desde 1990
- Miembro del COIT-AEIT (Asociación Española de Ingenieros de Telecomunicación) desde 1991
- Miembro de ATI (Asociación de Técnicos en Informática) desde 1992
- Miembro de la asociación AdaSpain desde 1993 (desde 1998 hasta 2001, miembro de su junta directiva).
- Miembro de Hispalinux, desde 1999

## 29.11. Labores académicas de gestión

- Coordinador del Grado en Ingeniería en Tecnologías de Telecomunicación de la Escuela Técnica Superior de Ingenieros de Telecomunicación de la URJC, desde mayo de 2017.
- Subdirector de Infraestructuras Tecnológicas y Laboratorios de la Escuela Técnica Superior de Ingenieros de Telecomunicación de la URJC, desde noviembre de 2011 hasta septiembre de 2015.
- Director del Máster Oficial en Software Libre de la Universidad Rey Juan Carlos, entre septiembre de 2010 y agosto de 2011.
- Director del Máster en Software Libre, Título Propio de la Universidad Rey Juan Carlos, entre septiembre de 2007 y agosto de 2010.
- Director del Departamento de Sistemas Telemáticos y Computación de la URJC, desde noviembre de 2007 hasta octubre de 2011.
- Subdirector de Infraestructuras Tecnológicas, Laboratorios y Calidad de la Escuela Superior de Ingenieros de Telecomunicación de la URJC, desde abril de 2006 hasta octubre de 2007.
- Subdirector de Infraestructuras Informáticas de la Escuela Superior de Ciencia y Tecnología de la URJC, desde septiembre de 2003 hasta noviembre de 2005.
- Coordinador de Tecnologías de la Información de la URJC, con dependencia directa del vicerrector de Nuevas Tecnologías, durante los años 2002 y 2003.



## **29.12. Actividades de transferencia de resultados de I+D**

- Co-fundador de la empresa Bitergia, dedicada a la analítica del desarrollo de software, en 2012, y colaboración en su puesta en marcha y primeros años de funcionamiento.

## **29.13. Otros**

- Desarrollador de Debian, desde 2001 hasta 2010.
- Co-coordinador de la Sección de Software Libre de la revista NOVATICA, desde 1996.

