Gamification in a Web-based platform for teaching robotics engineering

David Rodríguez-Rives[1[0009−0008−6777−4285]],
Raúl Fernández-Ruiz[1[0000−0001−8325−5372]],
Daniel Palacios-Alonso[1[0000−0001−6063−4898]],
Nikola Hristov-Kalamov[1[0000−0002−2194−1112]], and
José M. Cañas[1[0000−0003−4179−2211]]

Universidad Rey Juan Carlos, Spain

Abstract. An effective approach to enhance the educational experience is gamification, a methodology that incorporates game-like elements and mechanics into the learning process to boost motivation and engagement. This paper describes the integration of gamification into the Unibotics open web platform, which is dedicated to teaching robotics to university engineering students. First, the infrastructure necessary to support robotics tournaments based on a collection of robotics exercises was designed and programmed. It includes an automated asynchronous execution of the code from each participant and an *automated evaluation system* capable of ranking participants based on their performance in these exercises. It also automatically generates video summaries of the execution of the student's code. This innovation not only enhances the visibility of participants' achievements but also fosters a dynamic and interactive learning environment. It uses the latest web technologies such as Django, React and mainly Selenium, which leverages the task automation due to its versatility. Second, a pilot experience was implemented with 26 engineering real students at a Spanish university showing good results. This work highlights the potential of combining gamification, robotics and web-based platforms to improve the learning experience, particularly in higher education, by fostering engagement, competition, and practical skill development.

Keywords: Distance learning · Engineering education · Web-based and remote robotics · simulation.

1 Introduction

The rapid advancements in robotics and technology have underscored the need for innovative and effective methods to teach robotics engineering in schools and colleges. These developments also bring significant benefits to *STEM* disciplines — science, technology, engineering, and mathematics — by providing students with interactive and engaging learning experiences that foster critical skills. Within this context, robotics competitions have gained significant popularity, offering young learners dynamic challenges that stimulate interest and creativity.

One promising approach to enhance robotics education is *gamification* [4], which seeks to replicate the engaging and motivating aspects of games to boost students' participation and learning outcomes. Gamification, grounded in the Mechanics -Dynamics - Aesthetics (MDA) framework [13], provides a systematic way to design and evaluate game-like educational experiences. This framework encompasses three interconnected components:

– **Mechanics**, which define the rules, controls, and actions available to participants, shaping the structure of the activity. These may include systems such as player rankings or interactive elements, such as buttons, that facilitate actions like saving progress or transmitting the code being developed.
– **Dynamics**, which represent the interactions between participants and the system, highlighting the emergent behavior, motivation, and engagement during the learning process. Such as the motivation and competition students may experience watching the ranking.
– **Aesthetics**, which emerge from the interplay of Mechanics and Dynamics, reflecting the emotional responses and enjoyment experienced by participants. For example, a well-designed ranking system may create excitement among students as they work further and learn new techniques to improve their position.

Implementing gamification effectively in educational contexts requires a phased and strategic approach [3,9]. The following strategy was proposed by Deterding et. al in September 2011 [8]. It is proposed as follows:

1. **Objectives and Design**: Clearly defining the desired learning outcomes and planning activities and resources to support them.
2. **Content and Resource Creation**: Developing educational materials, interactive games, and feedback mechanisms to facilitate engagement.
3. **Gamification Integration**: Adapting existing resources and embedding gamified activities into the educational framework.
4. **Implementation and Monitoring**: Testing the activities and observing their impact on students.
5. **Evaluation and Adjustment**: Analyzing results and refining the approach based on feedback and performance data.

Building on this structured strategy, the *Unibotics* team has developed and implemented an asynchronous tournament framework as part of their efforts to create a comprehensive robotics academy for university students. The tournament system, integrated into the *Unibotics* web platform, incorporates automated evaluation mechanisms to rank participants based on their performance in robotics exercises. These are thirteen robotics engineering exercises which include engaging challenges such as a "cat and mouse" game with drones and "follow-line" competition using Formula 1-style circuits and vehicles.

The objective of this paper is to show the impact of this web-based platform on the teaching of robotics engineering, using gamification techniques grounded in the MDA framework. *Unibotics* [19] is a well-established platform used in several universities in Spain, Germany and Colombia, designed to enhance students' learning experiences through practical robotics exercises and innovative engagement strategies.

The remainder of this article is structured as follows: Section 2 reviews related work in the fields of robotics and gamification in educational contexts. Section 3 outlines the Unibotics web-based plataform. Section 4 describes the design and implementation of the asynchronous tournaments, along with the technologies and strategies employed. It also describes the automatic evaluation system, ranking mechanism, and video generation features. Section 5 presents the results of the tournament with real engineering students, analyzing its impact on their engagement and its overall effectiveness as an educational tool.

2 Related Works

In the last years many companies and organizations use elements and mechanics from games to encourage competition or learning, such as web rankings or championships [7].

2.1 Robotics competitions

Some communities create robotics competitions annually. The one of most popular is *For Inspiration and Recognition of Science and Technology (FIRST®) Robotics Competitions* which students varying from four to eighteen years compete in a series of challenges [17]. According to [1], the use of robotics as an educational tool, such as Lego Mindstorms, has a significant impact on learning outcomes by providing students with hands-on experiences that promote active participation and collaboration. The versatility and accessibility of platforms like Lego Mindstorms make them particularly well-suited for robotics competitions, where students not only learn to design and program robots but also develop essential skills such as critical thinking, creativity, and teamwork. These competitions, rooted in experiential learning, serve as a dynamic environment for students to bridge theoretical knowledge with practical applications, thus fostering a deeper understanding of STEM disciplines and preparing them for real-world challenges.

Likewise, since 1997, the *RoboCup Federation* organizes soccer leagues for robots all around the world. Its goal is to win a soccer game against the most recent World Cup leaders by the official rules of FIFA [14]. Within this context, the developments presented by the 2019 RoboCup AdultSize winner, NimbRo [18], stand out for their innovations in deep learning-based perception, in-walk kicking capabilities, push recovery mechanisms, and team play strategies. These contributions, which integrate advanced artificial intelligence techniques and robust system designs, not only highlight the potential of robotics to address complex challenges but also establish a strong connection with academic events such as the International Conference on Robotics and Automation (ICRA). ICRA, a key gathering for researchers and professionals in robotics, provides a platform to delve deeper into the advancements demonstrated in RoboCup, particularly in areas like perception and multi-robot coordination. This synergy between RoboCup and ICRA underscores how practical competitions and academic research complement each other to drive innovation in autonomous robotics.

Cyberbotics company also created a few years ago Robotbenchmark[1], a cloud-based 3D robot simulation platform, which offered a series of robot programming challenges including performance evaluation metrics and competition. Those challenges address various topics across a wide range of difficulty levels, from middle school to PhD.

2.2 Gamification in schools

Gamification has emerged as a powerful tool for enhancing student engagement and motivation in educational settings, with various platforms successfully integrating game-based elements into learning processes. One of the most widely adopted platforms in schools is *Kahoot!*, a game-based learning application that allows teachers to effortlessly create, share, and play interactive learning games or trivia quizzes. *Kahoot!* has become a cornerstone for engaging students in collaborative and competitive learning activities across diverse subjects [2].

Another notable example is *Minecraft Education Edition*, an initiative developed by Mojang and Microsoft that leverages the popular *Minecraft* game to teach STEM subjects to students aged three to fourteen [15]. By incorporating block-based programming and a highly immersive virtual environment, this platform facilitates the development of computational thinking, problem-solving, and creativity while maintaining a high level of engagement.

Additionally, *Scratch*, a coding platform and community available in over 70 languages and used by children from more than 200 countries, has made significant contributions to gamified learning. *Scratch* allows students to create and share digital stories, games, and animations, fostering critical skills such as logical reasoning, creativity, and collaboration. Its wide accessibility and versatility make it a valuable tool for introducing programming concepts and promoting digital literacy among young learners [11].

2.3 Web platforms to program robots

Web-based platforms have revolutionized the field of robotics by enabling developers to work online without the need to install specialized software or manage complex local infrastructures. These platforms provide powerful tools for simulation, testing, and programming, making robotics development more accessible and scalable.

One of the most prominent examples is *AWS RoboMaker*, a cloud-based simulation service provided by Amazon Web Services. *AWS RoboMaker* allows robotics developers to run, scale, and automate simulations while eliminating the need for managing physical infrastructure. This platform is particularly valued

[1] https://cyberbotics.com/doc/blog/robotbenchmark

for its seamless integration with the Robot Operating System (ROS), as well as for its ability to support large-scale simulations in cloud environments.

Another notable platform is *Asimovo*, which serves as a research-oriented resource for robotics developers. *Asimovo* offers an extensive collection of robots, environments, and open-source tools based on ROS, allowing users to simulate and test robotic systems online. This platform stands out for its comprehensive resources, which are particularly useful for research and prototyping in academic and industrial contexts.

For a more education and entertainment-focused approach, *Riders* offers a web-based platform that combines robotic programming with engaging simulations. *Riders* provides users with a wide variety of scenes and robot models, enabling students and hobbyists to learn robotics in an interactive and enjoyable manner. Additionally, the platform fosters global collaboration and competition through initiatives such as the *Riders Robotics League*, an international league designed to inspire innovation and community engagement in the field of robotics.

These platforms exemplify the diverse ways in which web-based tools are transforming robotics education and development, offering accessible, scalable, and versatile environments for developers, students, and researchers alike.

3 Unibotics

The design of the Unibotics platform has been thoroughly detailed in [19], with its architecture depicted in Figure 1. At runtime, the system is composed of two main components: the web browser and a Docker container. The web browser functions as the graphical user interface (GUI) of the application, enabling users to select an exercise in which they must solve a specific task by programming a robot. Upon selecting an exercise, the corresponding GUI is displayed, as illustrated in Figure 1. This web interface allows users to edit the source code of their robotics applications and observe the execution of the code. Multiple widgets are provided, including those for visualizing the simulated robotic environment, monitoring the robot in action, displaying a text console, and offering auxiliary debugging tools. Additional widgets tailored to each specific exercise are also available.

The execution of the robotics simulator and the user's robotics application occurs within a Docker container, referred to as the Robotics Backend (RB). Users are required to download this container from a public repository and launch it prior to using Unibotics. The Robotics Backend serves as the core processing unit of the system, with all necessary robotics dependencies pre-installed. This eliminates the need for users to install dependencies locally, significantly reducing the time required to begin programming robots.

Within the RB, a key component is the Robot Application Manager (RAM), which serves as a bridge between the browser and the robotics applications running in the backend. When a user runs their code via the browser, it is transmitted to the RAM, which forwards it to the robotics application template.

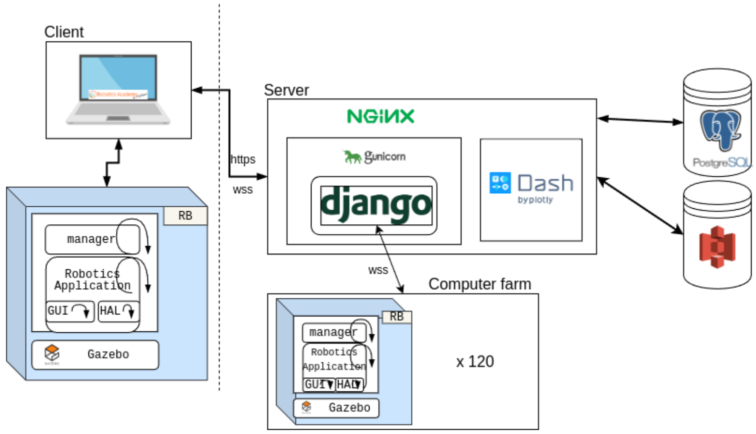


Fig. 1. Architecture of the Unibotics platform.

The robotics application executes the code and returns the results (e.g., images, data, or simulation outcomes) back to the RAM, which then delivers them to the browser for display.

Recent improvements to the platform, as detailed in [12], include the migration of the system's frontend to React for enhanced interactivity and performance. Additionally, the robotics application has been restructured into a single process that integrates user-generated code with two Python modules: `GUI.py` and `HAL.py`. The RAM now operates in four distinct steps to streamline communication between the browser and the backend, ensuring efficient execution.

The platform offers a wide range of robotics content, covering various domains:

- **Mobile Robotics exercises**, as introduced in [6], which focus on the fundamentals of programming autonomous ground vehicles.
- **Service Robotics exercises**, designed to simulate practical applications such as object manipulation and navigation in domestic or industrial settings.
- **Drone programming exercises**, as detailed in [5], which explore aerial robotics and autonomous flight control.
- **Control engineering exercises**, presented in [16], which provide users with an understanding of control systems and their application to robotic platforms.

Unibotics thus combines advanced technology and user-friendly tools to offer a comprehensive learning environment for robotics education and programming, making it a valuable resource for both academic and professional applications.

4 Automatic competitions in Unibotics

This section describes the framework and technologies used and the steps in the automatization processes, management and the infrastructure of these tournaments. This paper follows the work developed by Fernández-Ruiz et al [10], basically integrating the Automatic Competitions into the new Unibotics realease, as a single web-app into that platform, and refactoring it to work with REACT as major frontend technology.

4.1 Infrastructure

A crucial aspect of the infrastructure is the differentiation between the types of ranked tournaments implemented on the platform. On one hand, occasional events—such as evaluations—are tailored for scenarios where students must submit their solutions within a constrained time frame. On the other hand, asynchronous tournaments without strict time limits are designed to foster a more exploratory and iterative learning process, allowing students to experiment, refine their strategies, and improve progressively with each evaluation cycle.

Another vital component is the implementation of a dynamic ranking system aimed at motivating students to enhance their performance while promoting competition-based-learning. Tournaments are broadcasted live, allowing all users to access and follow the latest evaluations and rankings in real time, ensuring seamless functionality and an optimal user experience (see Figure 5). Furthermore, the platform incorporates challenges inspired by the original Unibotics exercises, but specifically adapted to enable more precise and targeted evaluations. These adaptations not only provide an opportunity to intensify competition but also ensure a robust and fair assessment process.

In the technical side, *React* components structure the tournament rankings and challenges, allowing them to function independently from one another and enabling development by different teams. The use of *PostgreSQL* database to storage all the tournaments, challenges and users evaluation information provide great flexibility in future updates and support.

4.2 Automation

This section explores the automation processes implemented inside Unibotics and highlights their advantages over traditional, manual methods for running a robotics competition.

An auxiliary user, named "Ceremony Master", has been created and automatically enters into the web plataform to execute the source code of each competition participant in order to evaluate it. *Selenium* has been utilized as the primary tool to automate interactions of this auxiliary user with the web browser, such as entering text or simulating button clicks for running the user source code. This technology facilitated the development of scripts capable of evaluating students' submissions (all of them in the same computer) and recording the evaluation process autonomously. By minimizing manual intervention, these automated solutions significantly enhance efficiency and consistency.

Challenge Deployment The process of deploying challenges is limited to specific user roles, namely the Ceremony Master and Unibotics administrators, who operate exclusively via the platform's management web page. The Ceremony Master role stands out for its access to a private portal, where challenges can be uploaded using a dynamic web template. This template is designed to simplify deployment by requiring only the retrieval of users' code and the delivery of evaluation results, thus avoiding the need for custom web pages for individual challenges. This streamlined approach ensures scalability, consistency, and reduced development overhead.

Using *Selenium*, a Python script automates the process of loading and evaluating challenges. First, logging into the Unibotics platform with Ceremony Master credentials, uploading the challenges included in the tournaments, retrieving participants' usernames and codes and submitting their evaluation scores.

Evaluation. Each challenge has its own unique method of evaluation. For example, the time spent to complete a track or the distance traveled in the Follow-Line challenge. The Selenium script will request each participants codes sequentially and initiate the evaluation process.

Recording. To document the evaluation process, the script integrates with *OBS (Open Broadcaster Software)* through its WebSocket service. After deploying the challenge, the script starts the recording process. Upon completion of the evaluation, the video file is saved and named after the corresponding participant.

For automated video uploads, the script uses the Google Developers Console and the YouTube Data API (Figure 2). OAuth 2.0 authentication is performed, which enables the script to upload the video directly to YouTube. After uploading, the script retrieves the unique YouTube video ID and updates the challenge's web page, ensuring that the video, along with the user's name and score, is properly stored in the database. The entire process is available on YouTube[2].

4.3 Management

A dedicated web portal consolidates all tournament information and provides tools for efficient management (see Figure 3). This portal simplifies the creation, editing, and deletion of tournaments, eliminating the need for direct access to the Django admin interface. To create a tournament, an administrator specifies key details such as the challenge to be loaded, the registration deadline, whether it is a dynamic ranking (without a specific date) or an occasional event, and the event's scheduled date. Administrators can modify these details at any time using the tournament editing tool. In addition, tournaments can be deleted simply by referencing their unique ID. The Ceremony Master also has the ability to manage participant data within each tournament via the private portal. This includes correcting scores or videos, or marking participants as unevaluated if necessary, ensuring accurate and up-to-date tournament records.

[2] https://youtu.be/M34SF9HevGI

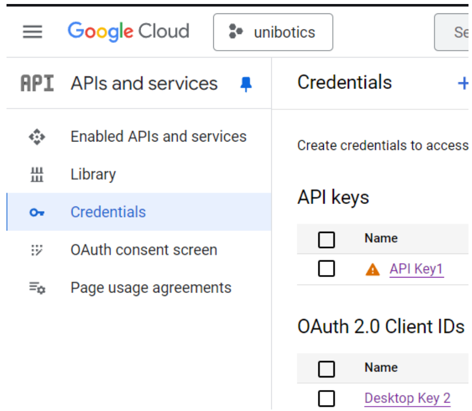


Fig. 2. Google Developers Console used for video uploads.

5 Results and Evaluation

To validate the functionality and robustness of the tournament system at Unibotics 26 robotics students from the *Robotics Software Engineering Degree at Rey Juan Carlos University (URJC)* were invited to participate in a specially designed occasional event aimed at testing the platform's capabilities in a controlled, competitive environment.

The engineering challenge proposed to the students was the "Follow Line" task, based on visual PID control, using a Formula1 car in a simulated circuit in Gazebo simulator (Figure 4). The evaluation metrics was based on the percentage of the track completed within a fixed time limit of 90 seconds, providing an objective measure of performance. During the course of two weeks, participants submitted their code, which was automatically evaluated by the platform and uploaded to YouTube. The results were displayed in real time on the dynamic ranking system of the platform [3], allowing participants to monitor their performance and progress relative to their peers. This feature fostered a competitive yet engaging atmosphere, enhancing the students' motivation and experience.

The successful execution of this event demonstrated the operational readiness of the system, validating its capacity to handle real-world scenarios involving multiple participants, automated evaluation, and data visualization. The final ranking of the tournament, as shown in Figure 5, highlights the ability of the platform to deliver reliable and transparent results in a user-friendly format.

6 Conclusions

This paper presents the design and implementation of an automated competition and gamification system inside a web-based platform for teaching robotics

[3] https://www.youtube.com/watch?v=P3YunnYXw3I

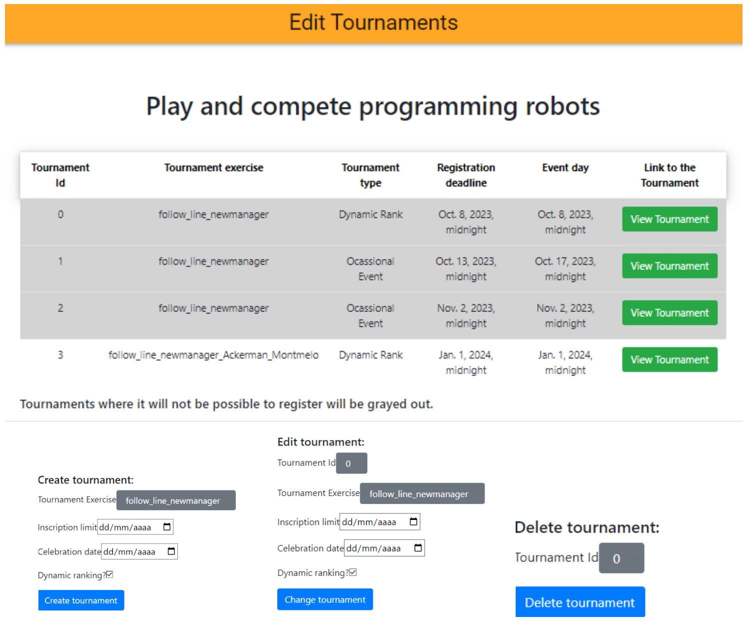


Fig. 3. Tournaments management web page

engineering. We have implemented a system in that encourages students to play in tournaments designed to propose challenges to enhance their learning and understanding in robotics engineering. Student submissions are stored in a database and automatically evaluated by the same computer. These evaluations are recorded and automatically uploaded to YouTube making them accessible to all participants. A ranking was implemented to display the students' score and evaluations,, allowing them to compare their performance to others and promote a sense of competition. The system also offers a simple way to start new tournaments, modify them or even delete past tournaments. This technology was tested in a competition with real university students and has proven to be effective to encourage them to further develop their learning skills.

Acknowledgments

This research was partially funded by the UNIBOTICS-GAM project Ref. TED2021-132632B-I00, from Proyectos de Transición Ecológica y Transición Digital 2021 call, Spain. Authors also appreciate the help of Google for improving Robotics Academy through the Google Summer of Code program since 2017.

References

1. Afari, E., Khine, M.S.: Robotics as an educational tool: Impact of lego mindstorms. International Journal of Information and Education Technology **7**(6), 437–

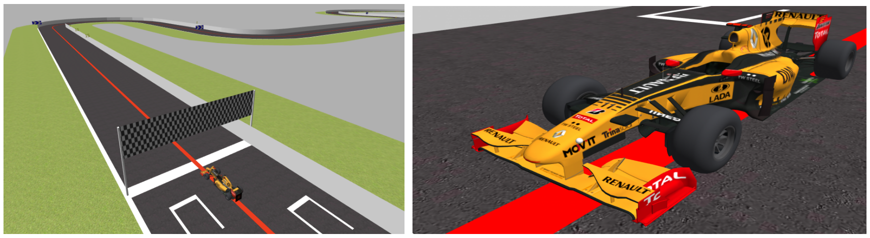
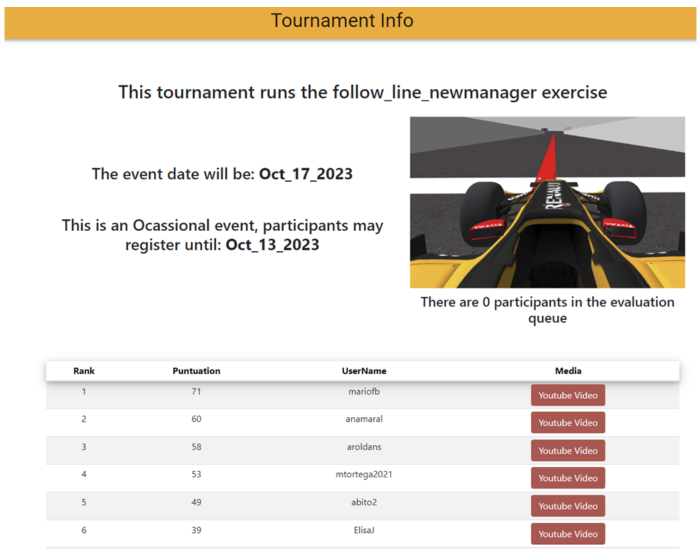


Fig. 4. Follow-Line challenge based on a Formula One track.



Rank	Puntuation	UserName	Media
1	71	mariofb	Youtube Video
2	60	anamaral	Youtube Video
3	58	aroldans	Youtube Video
4	53	mtortega2021	Youtube Video
5	49	abito2	Youtube Video
6	39	ElisaJ	Youtube Video

Fig. 5. Official Tournament Ranking.

442 (2017)

2. Aibar-Almazán, A., Castellote-Caballero, Y., Carcelén-Fraile, M.d.C., Rivas-Campo, Y., González-Martín, A.M.: Gamification in the classroom: Kahoot! as a tool for university teaching innovation. Frontiers in Psychology **15**, 1370084 (2024)
3. Baptista, G., Oliveira, T.: Gamification and serious games: A literature meta-analysis and integrative model. Computers in Human Behavior **92**, 306–315 (2019). https://doi.org/https://doi.org/10.1016/j.chb.2018.11.030, `https://www.sciencedirect.com/science/article/pii/S074756321830565X`
4. Buckley, P., Doyle, E.: Gamification and student motivation. Interactive learning environments **24**(6), 1162–1175 (2016)
5. Cañas, J.M., Martín-Martín, D., Arias, P., Vega, J., Roldán-Álvarez, D., García-Pérez, L., Fernández-Conde, J.: Open-source drone programming course for distance engineering education. Electronics **9**(12), 2163 (2020)
6. Cañas, J.M., Perdices, E., García-Pérez, L., Fernández-Conde, J.: A ROS-based open tool for intelligent robotics education. Applied Sciences **10**(21), 7419 (2020)

7. Chen, T.I., Lin, S.K., Chung, H.C.: Gamified educational robots lead an increase in motivation and creativity in stem education. Journal of Baltic Science Education **22**(3), 427–438 (2023)
8. Deterding, S., Dixon, D., Khaled, R., Nacke, L.: From game design elements to gamefulness: Defining "gamification". Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments pp. 9–15 (2011). https://doi.org/10.1145/2181037.2181040
9. Donnermann, M., Lein, M., Messingschlager, T., Riedmann, A., Schaper, P., Steinhaeusser, S., Lugrin, B.: Social robots and gamification for technology supported learning: An empirical study on engagement and motivation. Computers in Human Behavior **121**, 106792 (2021)
10. Fernández-Ruiz, R., Palacios-Alonso, D., Cañas-Plaza, J.M., Roldán-Álvarez, D.: Automatic competitions in the unibotics open online robot programming web. In: Tardioli, D., Matellán, V., Heredia, G., Silva, M.F., Marques, L. (eds.) ROBOT2022: Fifth Iberian Robotics Conference. pp. 463–474. Springer International Publishing, Cham (2023)
11. Fonseca, I., Caviedes, M., Chantré, J., Bernate, J.: Gamification and game-based learning as cooperative learning tools: A systematic review. International Journal of Emerging Technologies in Learning (iJET) **18**(21), 4–23 (2023)
12. García-Pérez, L., Roldán, D., Cervera, E., Wadhwani, P., Cañas, J.M.: Improving usability of a web-based platform for teaching robotics engineering. In: International Conference on Robotics in Education (RiE). pp. 313–324. Springer (2024)
13. Hunicke, R., LeBlanc, M., Zubek, R.: Mda: A formal approach to game design and game research. Proceedings of the Challenges in Games AI Workshop, Nineteenth National Conference on Artificial Intelligence (2004), `https://www.researchgate.net/publication/228884866_MDA_A_Formal_Approach_to_Game_Design_and_Game_Research`
14. Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., Osawa, E., Matsubara, H.: Robocup: A challenge problem for ai. AI magazine **18**(1), 73–73 (1997)
15. Ming, G.: The use of minecraft education edition as a gamification approach in teaching and learning mathematics among year five students. International Research Journal of Education and Sciences **4**(2), 14–17 (2020)
16. Pérez, L.G., Martín-Martín, D., Plaza, J.M.C., Sombría, J.C., Álvarez, D.R.: Introduction to control education with the unibotics web framework. Jornadas de Automática (45) (2024)
17. Rocker Yoel, S., Shwartz Asher, D., Schohet, M., Dori, Y.J.: The effect of the first robotics program on its graduates. Robotics **9**(4), 84 (2020)
18. Rodriguez, D., Farazi, H., Ficht, G., Pavlichenko, D., Brandenburger, A., Hosseini, M., Kosenko, O., Schreiber, M., Missura, M., Behnke, S.: Robocup 2019 adultsize winner nimbro: Deep learning perception, in-walk kick, push recovery, and team play capabilities. In: RoboCup 2019: Robot World Cup XXIII 23. pp. 631–645. Springer (2019)
19. Roldán-Álvarez, D., Cañas, J.M., Valladares, D., Arias-Perez, P., Mahna, S.: Unibotics: open ros-based online framework for practical learning of robotics in higher education. Multimedia Tools and Applications **83**(17), 52841–52866 (2024)