GOI ESKOLA POLITEKNIKOA

ESCUELA POLITÉCNICA SUPERIOR

MONDRAGON UNIBERTSITATEA

LOGLAS, WEB BASED MULTIPLAYER GAMES

DEGREE IN COMPUTER ENGINEERING

AUTHORS: Loredi Altzibar, Nahia Li Gómara, Aitor Landa, Xabier Landa, Ander Olaso, Ane Sajeras

Tutor: Goiuria Sagardui Mendieta

Arrasate, January 22nd, 2020

ABSTRACT

Nowadays, the video game industry is constantly growing, and more and more people enjoy this hobby. For this reason, “Loglas”, a web-based multiplayer gaming platform, has been developed. This project combines different types of games including true/false questions and multiple-choice questions, which are interactively designed. The content of these questions is didactic so that new knowledge on different topics can be acquired.

This project has used programming knowledge based on different languages and the techniques and tools of software engineering and systematic project management. This whole process has been developed using Scrum a method of managing tasks and workflows in order to obtain the best possible result of our project.

RESUMEN

Actualmente, la industria del videojuego está en constante crecimiento y son cada vez más las personas que disfrutan de esta afición. Por ello, se ha desarrollado “Loglas”, una plataforma de juegos multijugador basado en web. Este proyecto combina diferentes tipos de juegos entre ellos preguntas verdadero/falso y de múltiples opciones, las cuales están diseñadas de forma interactiva. El contenido de dichas preguntas es didáctico por lo que se podrán adquirir nuevos conocimientos sobre diferentes temáticas.

En este proyecto se han utilizado conocimientos de programación basada en diferentes lenguajes y las técnicas y herramientas de la ingeniería de software y una gestión del proyecto sistemática. Todo este proceso ha sido desarrollado utilizando Scrum un método de administración de tareas y flujos de trabajos con el fin de obtener el mejor resultado posible de nuestro proyecto.

LABURPENA

Gaur egun, bideo-jokoaren industria etengabe hazten ari da, eta gero eta gehiago dira zaletasun horretaz gozatzen duten pertsonak. Hori dela eta, "Loglas" garatu dugu, webgunean oinarritutako jokalari anitzeko joko plataforma. Proiektu honek hainbat joko mota konbinatzen ditu, horien artean egia/gezurra galderak eta aukera anitzekoak, eta horiek modu interaktiboan diseinatuta daude. Galdera horien edukia didaktikoa denez, hainbat gairi buruzko ezagutza berriak eskuratu ahal izango dira.

Proiektu honetan, lengoaia ezberdinetan oinarritutako programazioko ezagutzak eta software-ingeniaritzaren teknikak eta tresnak eta proiektuaren kudeaketa sistematikoa erabili dira. Prozesu hori guztia Scrum bidez garatu da, zereginak eta lan-fluxuak administratzeko metodo bat erabiliz, gure proiektutik ahalik eta emaitzarik onena lortzeko.

INDEX

IMAGES TABLE

1. INTRODUCTION

Nowadays, the video game industry is increasingly in demand due to the large number of people who enjoy this hobby. According to the Global Games Market there are more than 2.5 billion gamers in the world that generate US$152.1 billion during 2019, 9.6% more than in 2018. Due to this great demand, a web-based multiplayer gaming platform has been developed to play different types of online games in order to gain more knowledge about different topics.

This platform allows users (clients and administrators), to connect using their web browser, although only administrators will have the option to manage the questions. On the other hand, this platform provides the necessary infrastructure to be able to create/read/update/delete questions.

This project combines different types of games as true/false, multiple options, mixed games, drag and drop and Tetris all designed interactively. The content of the games questions is didactic and varied designed for people of different ages. Regarding Tetris, it helps brain efficiency.

This project has used programming knowledge based on different languages and the techniques and tools of software engineering. Moreover, it has been used a systematic project management. This whole process has been developed using Scrum a method of managing tasks and workflows in order to obtain the best possible result of our project.

This memory is divided into 8 parts. After the introduction, the project objective is explained. In the second section there are the related documents, that is, the reference to the annex with a brief description. Then there is the state of the art and after it, a short explanation of the problem to be solved. After it, there is the solution to the problem with the development of the different parts. In the next part, there are the conclusions and the future lines. At the end of the document there’s the bibliography.

The following is the procedure that has been followed from the approach of the problem to the design of the solution for further development and result.

2. RELATED DOCUMENTS

Throughout the document, the different tasks performed to achieve the final product are briefly explained. This information has been necessary to obtain greater knowledge of the subject and for its further development. The tasks carried out are briefly explained throughout the document. In order to obtain more detailed information about the different tasks that have been carried out, different annexes have been generated. The following information is included in the annexes:

1. Project management
   1.1. MEETING_MINUTES + RETROSPECTIVE + REVIEW
   1.2. PROJECT MANAGEMENT DOCUMENTATION
2. Interface development
   2.1. AXURE
   2.2. UCD
   2.3. Usability test AXURE
   2.4. Grafana DASHBOARDS
   2.5. Usability test LOGLAS
3. Software development
   3.1. FINAL HANDBOOK – SOFTWARE ENGINEERING
4. Operating systems
   4.1. OPERATING_SYSTEMS_DOCUMENTATION
5. Web engineering
   5.1. W3C

3. ECONOMIC AND SOCIAL IMPACT

Have you ever stopped to think about the great impact video games have on users?

In the last decade the video game industry has had a great evolution and impact on society, economically and culturally. Nowadays, video games have a great impact on children and young people, as it is a way of entertaining and working with various devices: computers, consoles, mobile phones...

According to various sources, this industry is expected to continue growing so that it could become a $300 billion industry by 2025. The market shows high growth in the future also due the increasing usage of smartphones and consoles along with cloud penetration.

VIDEO GAMES EVOLUTION[1]

This web platform is a tool for both personal and professional progress and development. It makes it possible to increase knowledge on various subjects or to strengthen the knowledge acquired. It also allows you to spend a pleasant time with your opponents. Improves learning, memory and social skills of different users.

According to the economic impact, it is necessary to talk about the growth of the gaming industry over the last decade. In relation to Loglas this game is expected to become viral and one of the most played games, because is free and offers more options than its close competitors.

In the following table there is a comparison between several competitors emphasizing different characteristic of each of them.

CHARACTERISTICS	LOGLAS	PREGUNTADOS	KAHOOT
You can play different types of games in the same game	 You can play in the same game to T/F or multiple-choice questions.	 You only have multiple choice questions.	 You only have multiple choice questions.
Chat option	 You can chat with different people.	 You can chat with different people.	 You do not have a chat.
You must wait until the other opponents answer the question.	 All the players answer to the question in the same instance.	 You need to wait until the other opponent answers her question.	 All the players answer to the question in the same instance.
All the opponents have different question	 All the users have the same questions.	 Each user has different questions and about different category (science, sport…).	 All the users have the same questions.
You need internet connection	 You need to be connected to the Internet.	 You need to be connected to the Internet.	 You need to be connected to the Internet.

4. APPROACH TO THE PROBLEM

This section details the problem and mentions the points needed to reach the target.

4.1. EXAMINATION OF THE PROBLEM

The video game industry is currently increasingly in demand and it is expected to continue growing in the coming years. According to some sources the video games market could become a $300 billion industry by 2025, with the growth of mobile gaming and innovative offerings, like cloud gaming. Concretely in 2018, the video games market generated $131 billion, with mobile gaming outpacing revenue made by pc and console gaming. Looking ahead, the biggest revenue drivers anticipated continue to be mobile gaming, as well as innovative technology like cloud gaming and VR gaming.

Nowadays, there are many online games about general culture that allow us to acquire new knowledge about different topics, although not all the games offer the same options. While some have a chat to communicate with other users, others have a multilingual platform or others have the option to choose the topic of the questions. But what application offers all these options?

4.1.1 TECHNICAL PROBLEMS

One of the main problems encountered during this project is the **coordination between the teammates** in order to develop the java web application. If there are many developers working at the same time, it might be messing to unify all the code in a final project. Furthermore, it could incur in the communication and work of the team.

On the one hand, problems are usually the **way it is programmed** and if the appropriate techniques are being used. This happens because sometimes there is a tendency to write many comments due to the complexity of the code itself. This often causes problems when looking for information or modifying it. In addition, it is necessary to carry out tests to verify the correct operation, since both tests can be carried out with unitary tests of code and a simulation. Performing the test between the different pages and seeing how the interaction is performed.

The main cause of been a lot of project not going in progress is due to **the poor planification**. This is because many times the projects are not organized properly. Starting from the beginning, it could be noted that many projects start with good planning, but unfortunately, they do not end up fulfilling it.

On the other hand, sometimes some tasks take much more time than the one that it was expected, but by the time the team realizes about this situation, it is usually too late. This really happens due to the lack of planning.

One of the most common problems is usually to worry about **changes in the database**, wasting too much time in that process. In addition, a solution to the problem of the difference between the two coexisting data models in an application must be sought: the

one used in the computer memory (object orientation) and the one used in the databases (relational model).

Another problem has been detected that is related to **the control of actions that are done simultaneously**, since they usually generate conflicts. These mainly appear when we talk about multi-user systems like the one that is going to be developed. In the process of managing simultaneous operations in the database, problems are often encountered because they can often interfere with each other, and it is very important to maintain the integrity of the data. Therefore, interference should be avoided when two or more users access the database simultaneously and at least one is updating data.

Finally, nowadays, it is lived surrounded by big data, where small environments also located a lot of data. Once collected the following is to analyze them, the problem comes where they can analyze millions of different ways. On the other hand, it is also true that **data visualization** is considered good practice today. It helps teams to monitor their environment, detect patterns and take measures identifying anomalous behaviors. It even sometimes provides the visibility required to understand what happened at a given time.

In the following table there are the problems that we have found during the development of the project with the corresponding solution.

PROBLEMS	SOLUTIONS
Coordination between the teammates	Specify programmer style and use version control tool.
Poor planification	Select the best methodology according to the project type.
Way it is programmed	Unitary test, mocks...
Changes in the database	Tool to provide this change.
The control of actions that are done simultaneously	Concurrency control.
Data visualization	Data visualization tools and appropriate graphics for each display

4.2. SPECIFICATIONS OF THE OBJECTIVES

The main objective of the project has been to develop a Web based multiplayer game using a development process, concretely the software engineering techniques and tools. The project management has been systematic.

According to the operational objectives have been the following ones:

1. Create an intuitive and user-friendly interface, designed for people of any age. Furthermore, Loglas has been designed to be an efficient, complete and an accessible application.
2. Be able to develop a multithread application in Java to give solution to the concurrency issues such as the score table, lobby players or accessing lobbies. These problems are explained later, concretely in the 6.5 section.
3. Be able to plan the project nimble and follow up the different tasks to achieve the project objectives. Moreover, Scrum helps completing complex project that previously could not be done.
4. Use software engineering tools, static analysis, automatic builds and continuous integration. The static analysis must be done during all the project so that it maintains a good quality of the code. According to the automatic builds they are necessary to automate and make the dependency management and build and deployment process easier. Finally, there is the continuous integration, concretely Jenkins.
5. Design a web architecture that provides the needed infrastructure and functionalities to Create/Read/Update/Delete questions. The platform offers at least two different types of games: multiple choice answers or true/false answers. The platform has two different clients: users and administrators and they will be connected using their web browser. Customers will have access to the data in a useful and concrete way and will have the option to communicate in real time with their opponents.

4.3. PLANNING

In order to carry out this project, the tasks have been divided by people into a specific order as some tasks depend on others.

Before starting the application, the meetings with teachers have been very important and of great help, as they advise on the tasks carried out with an external and experienced vision.

To manage all these tasks, we have used Scrum a method of managing tasks and workflows in order to obtain the best possible result of our project.

5. DESIGN OF THE SOLUTION

This project is based on a web-based multiplayer gaming platform. Five types of games are offered: true or false, multiple options, a mixture of both, drag and drop and Tetris. The content of these questions is interactive and designed to gain more knowledge about different topics, as it is designed for people of all ages. On the other hand, there are two user types: administrator and user. The administrator will be able to load different questions to the application or to see the different statistics. According to the users will have the opportunity to log in, create a game room, or join to a game room as a viewer or as a player.

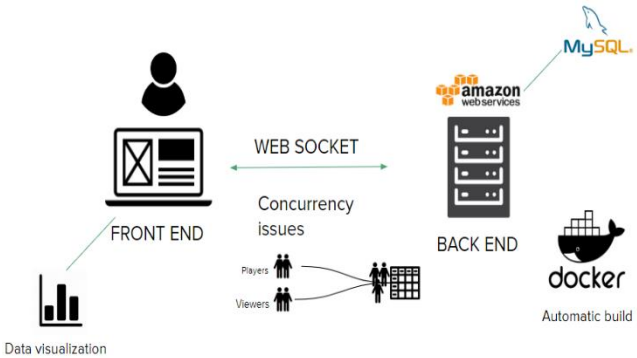
During the development process, Software engineering techniques and tools and systematic project management have been used. Moreover, automatic builds have been done from the beginning of the project such as test cases. Git+branch strategy has been used.

On the one hand, a multithreaded application in java has been developed.

On the other hand, different libraries, frameworks and tools have been used to develop new interfaces. Several graphs have been implemented to represent the statistics of the different data obtained, such as the line diagram to represent the points obtained per day, the pie graph to represent the most played games, gauge to see the number of players that are in the games... These graphics will be visible from the application or even from Grafana where it will be the chance to see the users entering and leaving the application in real time.

All of this has been implemented using Scrum, for a better organization and planning of the project. With this method we have achieved visibility of the workflow and improved decision-making. The Scrum method has provided us with productivity and quality, as well as flexibility and adaptation to customer needs.

In order to have a more detailed information about the design of the solution, it has been created the following UML diagram where there are the different components of the designed solution. These components will be explained throughout the following sections.



UML DIAGRAM[2]

6. DEVELOPMENT AND RESULT

The work has been divided into several tasks and then, these have joined to achieve the final product. This section mentions and explains how these tasks have been carried out.

6.1. PROJECT MANAGEMENT

We are going to use “Scrum” starting from the practical test run and our basic knowledge. Our goal is to improve and polish our knowledge on the methodology applying it on this project. We have selected scrum methodology because its objective is to control and plan projects with a large volume of last-minute changes, where uncertainty is high.

6.1.1 MEETINGS

Project planning meeting: this is the first meeting of the project management. In this meeting we defined into how many sprints we should separate the project and the project tasks with their level of difficulty and duration, product backlog. This product backlog can be modified in any moment, can add new tasks and modify task difficulty or duration.

This project process will be composed of 3 sprints (regarding the final sprint). Each 2 weeks long. The Sprints are:

1 Sprint 1 until 13 December

2. Sprint 2 until 10 January

3. Sprint 3 until 22 January

Sprint meeting: This meeting must be done before any sprint. In this meeting it needs to be decided what to do in each sprint. Moreover, it will be considered the priority of each activity.

Daily meeting: In this meeting it must be spoken about what to do each day and which are the activities that were done the day before. Moreover, there can be seen the issues or the problems that have been.

Review meeting: In this meeting the project status will be analyzed with the product owner, who can share a review with all the team. That review will be used to have a global vision of the product.

Retrospective meeting: it will be used to analyze how we did the process, i.e. how we went in the sprint as a group, improving coordination, and seeing what failures we had, being able to improve them as a group for next time.

6.1.2 ROLES

At scrum methodology there are different roles:

- Scrum Master: It is a person who makes sure that the methodology is being carried out correctly. Nahia Li Gomara.
- Product Owner: It is responsible for the management of the product related to a defined product vision, the prioritization of work, the financing of the project. Xavier Landa.

Team: takes care of the technical development of the project. All us.

6.1.3 SCRUM BOARD

This tool is used to visualize the flow of tasks. This board is separated in 5 different columns and all the task must get through these columns.

- Sprint Backlog Item: where this sprint task start.
- Refine: where the task is refined in sub task.
- To Do: where task and subtask are as they develop.
- Test: where task and subtask are as they test.
- Done: where task and subtask end the flow.

If any task gets stuck it stays between columns until it is repaired.

6.1.4 BURNDOWN CHART

This graph shows how the number of tasks to be done decreases each day.

6.2. INTERFACE DEVELOPMENT

Loglas has used a user-centered design methodology for the interface development and code them with web technologies. Axure has been used to plan and prototype the first version of Loglas, while the real implementation has been done using a web platform.

More details about UCD process will be provided in UCD attached file.

6.2.1 DATA VISUALIZATION

According to the data visualization, different types of graphs have been defined considering the data that we wanted to visualize. So, in order to do it the graphs have been divided into users and administrations where each of it could see different information. While users can see the points obtain per day or the percentage of correct answers to questions, administrators can see the number of players that are playing, or which is the most played game.

Grafana, an analytics and monitoring solution for the database, has been used to see the statistics in real time as for example the number of players that join or go out of the

game. Moreover, data visualization libraries, frameworks and tools have been used to develop the dashboards in the web platform.

6.2.2 UCD DOCUMENT

On the other hand, a UCD document has been done, where there are Loglas requirements such as business, user, functional or game and the usability testing for the interface evaluation.

- Business requirements:
  - App will let users create a new game room.
  - App will let users join to a created game as players or viewers.
  - App will let users join to T/F or multiple-choice games.

- User requirements:
  - Users will have two different buttons to select the kind of game (T/F, multiple choice or mixed games).
  - Registered users will see the games that are already created.

- Functional requirements
  - The application will redirect to a list of games if the login is correct.
  - The application will set a timer for the duration of each question.

- Game requirements
  - If the game room is full the game automatically starts.
  - For each correct answer you win 10 points but for each bad answer you lose 3 points.

In this way we can check that the program worked as expected. Moreover, we could notice about errors that we didn't know that they were, so we could solve them and make some improvements for making the application better. Clicks have been the most used iteration types.

6.3. PROFESSIONAL SOFTWARE DEVELOPMENT

For the software development we have set a Linux server with AWS (Amazon Web Services), in which we have installed all the tools that we needed in order to develop and deploy the code.

We have used docker in order to install all the software more easily and to keep them isolated from the main server not to have problems with the dependencies.

More details about software engineering will be provided in the Software Engineering Handbook attached file.

6.3.1 DOCUMENTATION

In order to maintain the code readable and understandable we set some rules to follow when developing the code of the application. One of the rules is to document all the classes and functions that are developed so that the documentation can be generated.

6.3.2 VERSION CONTROL

The main goal of version controlling is to have trackable code during all the process of the development, to achieve that we have used GitLab, which is a remote repository where the code is stored.

We have used different branches to develop, such as to develop new features or to make fixes when needed. A main branch, the master branch is used for the working versions of the project and a development branch to make and test all the changes of the features before merging to the master and tag them with the new version.

6.3.3 STATIC ANALYSIS

A static analysis has been made during the project so that it maintains a good quality of the code, each time a change is made to the development branch an analysis is made to check if the quality of the project is good enough to deploy it.

To achieve this goal, we have used SonarQube, which is installed in the server so that all the developers can access it whenever they wanted or needed.

For the testing of the code we have used JUnit to test the functionalities using the white box testing method and pass it to the SonarQube server using Jacoco. On the other hand, when we refer to the black box testing we use Katalon, which is installed on the local machines of the developers, it is used to test the navigation of the application and checks whenever the application can be used correctly.

6.3.4 AUTOMATIC BUILDS

We have set automatic builds with Maven to automate and make the dependency management and build and deployment process easier. All the dependencies needed for the project are defined in an XML file so that Maven can install all the dependencies it needs despite of the machine it is running on.

6.3.5 CONTINUOUS INTEGRATION

To make the deployment process automatic we have used continuous integration software, more concretely Jenkins, which is also installed in the server inside a docker machine, in it we have defined a pipeline so that it is triggered by a webhook each time a change is made into the development branch, when that happens Jenkins downloads the latest version of the code, it builds the project, makes the static analysis and the testing and if it succeeds it deploys the application so that it can be used.

6.4. WEB DEVELOPMENT

In order to achieve an online game site, this software is developed in a web environment. To achieve this web environment, a Java Web Server was thought the most convenient, which is designed for medium and large business applications. This supports Java Servlets and Java Server Pages.

This development is based on html code to describe browser pages for the client alongside JavaScript for the logic. On the other hand, Java programming language is used for the server-side logic.

Through the process, some new tools were used to facilitate the code development and to avoid various problems that often arise during programming.

6.4.1 SPRING FRAMEWORK

Spring is an open source framework for creating Java applications. It has a modular structure, that is, it is oriented to be able to have different modules that group diverse functionalities. It also has great flexibility, allowing us to develop all types of applications, desktop applications, applications that access databases via SQL directly, small, medium, large etc. As our application was increasingly growing, the framework was needed to give a modular structure and facilitate programming and understanding of the code for the developers.

6.4.2 HIBERNATE

Hibernate is an object-relational mapping tool. It provides a framework for mapping an object-oriented domain model to a relational database. Hibernate handles object-relational impedance mismatch problems by replacing direct, persistent database accesses with high-level object handling functions.

Hibernates primary feature is mapping from Java classes to database tables, and mapping from Java data types to SQL data types. Hibernate also provides data query and retrieval facilities. It generates SQL calls and relieves the developer from the manual handling and object conversion of the result set.

Since or software is tightly related with Java language, hibernate (JPA) helps the developers with Java object persistence's. Furthermore, this avoids the problems of defining and sharing a database with all developers when testing and developing locally.

6.4.3 TILES

Apache tiles is a templating framework that allows to simplify the development of user interfaces. Tiles allows us to develop different fragments of our interfaces and to assembly them as needed.

Tiles contributes with modularity, which is important for scalable project like this one. Apart from having to write less code using this framework allowed us to maintain the same design in the whole page making the page look more professional and more appealing for the user.

6.4.4 WEB SOCKET

WebSocket is a computer communications protocol, providing full-duplex communication channels over a single TCP connection.

The WebSocket protocol enables interaction between a web browser (or other client application) and a web server with lower overhead than other alternatives, facilitating real-time data transfer from and to the server. WebSocket provide a standardized way for the server to send content to the client without being first requested by the client and allowing messages to be passed back and forth while keeping the connection open.

For an active web games application like this project, it is important that user actions are fetched and sent at life (streaming) and have a tight communication between client and server.

Most browsers support this protocol, including Google Chrome, Microsoft Edge, Internet Explorer, Firefox, Safari and Opera. This way, we are covering many potential clients.

6.4.5 GSON FOR JSON MAPPINGS

As life communication is very important and as Java and JavaScript language have no communication, we need to satisfy a need for information encapsulation which can be achieved with JSON objects. In order to work with JSON, Gson api is used to map Java object in order to send through web-socket into the client and vice-versa.

Gson is an open-source Java library to serialize and deserialize Java objects to JSON. This allows us an easy communication between client - server to send “encapsulated” information.

6.4.6 JAVAX.MAIL

Javax.mail is an open-source Java library to send mail from the code. Is important to configure the sender's Gmail first. This is used to recover the user’s password.

6.5. CONCURRENT APPLICATIONS

A concurrency issue is given when two or more application threads are sharing a common resource and they want to access it at the same time. A concurrent application must manage this issue in order to handle the common resource access.

Regarding the concurrency issues which have been detected in the application, there are three different main issues that have been solved. Although, there are more issues that can be solve with the same solution, for that issues there are not developed any solution because the solution proposed for the main issues can be adapted to the other issues.

More details about solutions will be provided in OSS attached file.

6.5.1 SCORE TABLE

The application has a score table to record players score in each round. This table is going to be accessed by all players of a round because they always will have a score. With the purpose of control this table and avoid the modifications between players, the players will access to the score between individually. A monitor is going to be used to control this access.

Moreover, users with viewer role can see this table whenever they want while the round is going on. So, it can be the case that there are multiple viewers viewing the scoreboard and at the same time players who are playing and changing the score table.

To solve this concurrency problem, it has been decided to implement it based on the problem of readers/writers giving priority to writers, because it is wanted to keep the scoreboard as up to date as possible.

6.5.2 LOBBY PLAYERS

The lobby management requires to consider the players which are inside each lobby, there is a counter which increase if a player join in a lobby and decrease if a player leave a lobby.

It can be a possibility that two or more players join in at the same time, so to avoid the unneeded changes this variable will be protected with a monitor.

6.5.3 WEBSOCKETS USAGE

Websockets can only be used by one player. Therefore, it is necessary to protect these items so that 2 or more players cannot access them at the same time.

Websockets establish a communication, when this communication channel is created, the variable will be protected, by means of a monitor, and until it is finished, the variable will not be released.

7. CONCLUSIONS AND FUTURE LINES

Loglas, a web based multiplayer game platform has been develop with the following characteristics:

- There have been develop 5 types of games: True/False, multiple options, mixed games, drag and drop games and Tetris. These web games are running on server.
- Data stats visualization. Grafana and the web platform has been used to develop the different graphs.
- Different user interactions have been used.
- There is support for users and administrators, each of it with different options. Users can log in, create a lobby and join to it, whereas administrators can load questions or see the different statistics.
- Thread safe application, that is, there is a multiple user accessing resource.

All the above problems have been resolved, although in the future it would be appropriate to use another planning method instead of Scrum. Although Scrum offers many advantages such as the fact that each stakeholder is assigned a role or that allows us to save time to achieve the most complex goals, it often requires a lot of meetings for little progress. This can become exhausting and stressful as it involves meeting several times with the group to talk about the same subject and in the end could happened that some members of the team lose interest in the project. In addition, Scrum does not allow changes to be made during Sprints when they are necessary to achieve the final objective, nor does it necessarily require each element to be addressed in order of priority.

In order to solve the problem mentioned above, it would be useful to use the XP methodology. It is a supplementary of Scrum that allows us adaptability. XP teams can incorporate new work items to an iteration and change items of equivalent size (if they have not been started) if the customer decides a new priority.

The following table provides more clarity for the most significant differences in each method.

SCRUM	XP
• Changes in sprint are not allowed • Once tasks for a certain sprint are set, the team determines the sequence in which they will develop the backlog items • The Scrum Master is responsible for what is done in the sprint, including the code that is written.	• If the team hasn't started working on a particular feature, a new feature, of equivalent size can be swapped into the interaction in exchange for an un-started feature. • Tasks are taken in a strict priority order

• The validation of the software is completed at the end of each sprint, at Sprint Review.	• Developers can modify or refactor parts of code as the need arises • The software needs to be validated at all time, to the extent that tests are written prior to the actual software.

The second impact is closely related to what previously was stated, as during many projects the code cannot be edited in a shared way, as its subsequent unification became complicated and inefficient. In this project, thanks to the Git version control tool, both modifications and stable versions of the project have been made, affecting the rest minimally. One of the features that should be highlighted is that you could go back to the previous version and compare the changes made in each of them.

In relation to the multilanguage in the future, new languages could be added so that the application is accessible to more people. Moreover, the code could be modified to make it more readable, or even implement a page reader to reproduce the content of the page by voice.

On the one hand, new games could be added, and the interface improved to make it more attractive. Furthermore, the security of the website could be improved. Another aspect of improvement would be the automatic destruction of the lobbies after certain time. This action would be executed if the room doesn't have the required minimum number of players. On the other hand, in case all players answer the question, they would automatically move on to the next question, without having to wait for the time to run out.

Finally, it should be mentioned that the application is modular and scalable, so it allows us to adapt to future needs with ease. Therefore, it adds flexibility to the possible changes that may occur in our environment.

8. BIBLIOGRAPHY

[1] **"BAELDUNG"** [Accessed in January 2020] Accessible from https://www.baeldung.com

[2] **"TUTOR DE PROGRAMACIÓN"** [Accessed in January 2020] Accessible from http://acodigo.blogspot.com/2017/04/spring-mvc-vistas-json.htm

[3] **"CÓMO LEER Y ESCRIBIR FICHEROS JSON EN JAVA"** [Accessed in January 2020] Accessible from https://blog.openalfa.com/como-leer-y-escribir-ficheros-json-en-java

[4] **"AXURE"** [Accessed in December 2019] Accessible from https://docs.axure.com/axure-rp/installing-activating/activating-rp/

[5] **"MASTERING SPRING BOOT 2.0"** [Accessed in December 2019] Accessible from https://www.packtpub.com/eu/application-development/mastering-spring-boot-20

[6] **"SPRING 5 RECIPES"** [Accessed in December 2019] Accessible from https://www.apress.com/la/book/9781484227893

[7] **"CÓMO ENVIAR CORREO ELECTRÓNICO CON JAVA A TRAVÉS DE GMAIL"** [Accessed in January 2020] Accessible from https://www.campusmvp.es/recursos/post/como-enviar-correo-electronico-con-java-a-traves-de-gmail.aspx

[8] **"CREATE A MULTI LANGUAGE WEB APPLICATION WITH SPRING BOOT"** [Accessed in January 2020] Accessible from https://o7planning.org/en/11691/create-a-multi-language-web-application-with-spring-boot

[9] **"SPRING"** [Accessed in December 2019] Accessible from https://spring.io/

[10] **"CONCURRENCY"** [Accessed in January 2020] Accessible from https://docs.oracle.com/javase/tutorial/essential/concurrency/

[11] **"GUIDE TO java.util.concurrent.Locks"** [Accessed in January 2020] Accessible from https://www.baeldung.com/java-concurrent-locks

[12] **"SEMAPHORES IN JAVA"** [Accessed in January 2020] Accessible from https://www.baeldung.com/java-semaphore