FACULTY OF ENGENEERING

Degree in Computer Engineering

Fantasy

Course 2018-2019

Luis Gutiérrez Flores
Nicolás Ruiz Requejo
Jesús Rodríguez Heras
Arantzazu Otal Alberro
Alejandro Segovia Gallardo
Alejandro José Caraballo García
Gabriel Fernando Sánchez Reina

Puerto Real, May 31, 2019

FACULTY OF ENGENEERING

Degree in Computer Engineering

Fantasy

DEPARTMENT: Computer Engineering.
PROJECT DIRECTOR: Alecia Adelaide May Reid.
PROJECT AUTHOR: Team Fantasy.

Puerto Real, May 31, 2019

Signed: Team Fantasy

Summary

Web application to promote learning through the imagination and creativity of children between 10 and 13 years old in scientific-technological subjects in collaboration with the European project STIMEY.

As a game, children can create interactive stories and teachers can evaluate them.

Keywords: Fantasy, learning, development, illusion, entertainment, creativity, questionnaire, evaluation, teaching, science, European Union.

Contents

Part I

Prolegomenon

Chapter 1

Introduction

1.1 Motivation

It is a work of the subject “Proyectos Informáticos” which, at a professional level, helps us to gain work experience and face real situations facing a demanding clientele.

1.2 Description of the current system

Initially, our client had an application that showed information about a topic on a page and the students did not focus on learning, but went directly to the final questionnaire in order to finish earlier. This means that students did not learn properly or encourage their imagination or creativity.

1.3 Objectives and scope of the project

1.3.1 Objetives

Motivation of creativity and promotion of imagination in children.

To fulfill the general objective, we will have to cover the following points:

- Interactive learning resources.
- Can be evaluated by a teacher.
- You can share fantasies between users.
- It is simple and manageable by primary school students.
- Encourage STEM teaching skills (science, technology, engeneering and maths).

1.3.2 Scope

The students will be able to create fantasies, share them and they will be able to be evaluated by the professors, who will be able to send as a task the making of fantasies.

1.4 Organization of the document

This document is organized according to the specifications set out for the presentation of an end-of-degree project following the following sections:

1. Introduction.
2. Project plan.
3. Analysis of requirements.
4. System design.
5. Implementation of the system.
6. System tests.
7. User manual.
8. Installation manual.
9. Conclusions.

In addition to this document, we also have an appendix where the user manual is narrated, step by step.

Chapter 2

Planning

This chapter includes the planning, the approach and the begining of a project that we have named "**Fantasy**", a web portal where teachers can make a series of tasks (called fantasies) with the objective that students can play and learn in a creative way.

The students will also have the possibility of creating the fantasies that the teacher sends them as project and then they will be evaluated by said professor.

2.1 Development methodology

The methodology used will be **Scrum**: Agile development method characterized by having an incremental development and basing the quality of the result on knowledge rather than on the processes used.

2.2 Project planning

The project will last three months and weekly meetings with the client will be held for a maximum of one hour.

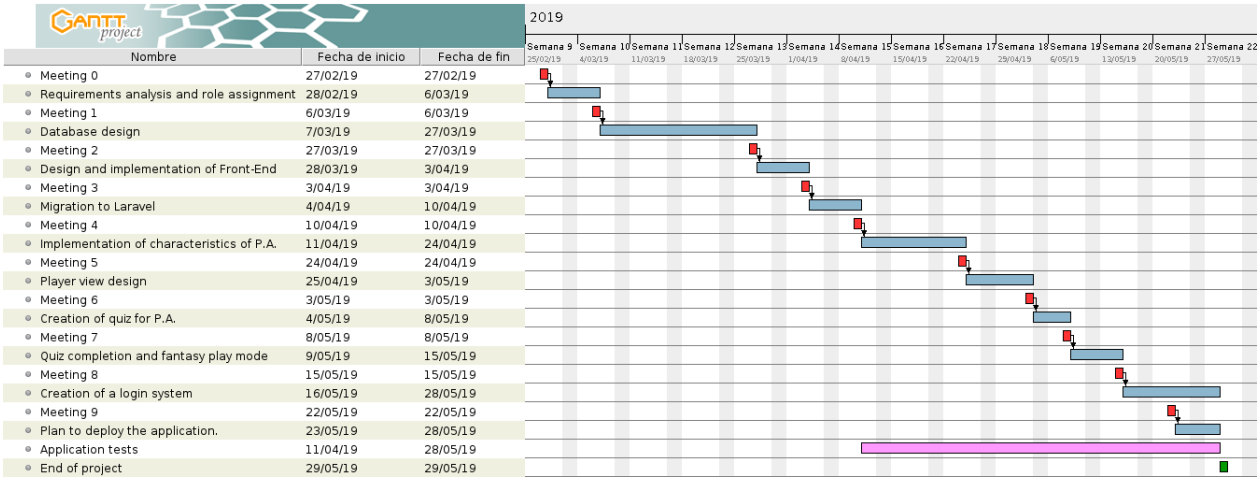


Figure 2.1: Gantt diagram

2.3 Hits

2.3.1 Sprint 0 (27^{th} February to 6^{th} March)

1. Creation of work platforms and version control (GitHub).
2. Creation of the requirements sketch.
3. Creation of use cases and their descriptions.

2.3.2 Sprint 1 (6^{th} March to 27^{th} March)

1. Creation of platform mockups.
2. Implementation of the database.

2.3.3 Sprint 2 (27^{th} March to 3^{rd} April)

1. Implementation of front-end.
2. Migrations of the database.
3. Adaptation of the project to the Laravel framework.
4. Creation of fantasies.

2.3.4 Sprint 3 (3^{rd} April to 10^{th} April)

1. End of front-end.
2. Final migrations of the database.
3. End of fantasy creation.

2.3.5 Sprint 4 (10^{th} April to 24^{th} April)

1. End of database migrations.
2. Creation of active points with their basic characteristics.

2.3.6 Sprint 5 (24^{th} April to 1^{st} May)

1. Creation of active points with all their characteristics.
2. Start of the view to be able to play the fantasy.

2.3.7 Sprint 6 (3^{rd} May to 8^{th} May)

1. Creation of the quiz for the active points.
2. Continuation of the view to be able to play the fantasy.

2.3.8 Sprint 7 (8^{th} May to 15^{th} May)

1. Finalization of active points with all their specified characteristics.
2. Finalization of the view to be able to play the fantasy.

2.3.9 Sprint 8 (15^{th} May to 22^{nd} May)

1. Creation of a login system.

2.3.10 Sprint 9 (22^{nd} May to 29^{th} May)

1. Plan to deploy the application.

2.4 Meetings

2.4.1 Meeting 0 (27^{th} February)

1. Analysis of system requirements.
2. Creation of use cases.
3. Proposal of the system database.
4. Distribution of sprint tasks 0.

2.4.2 Meeting 1 (6^{th} March)

1. End of sprint 0.
2. Distribution of information to search.
3. Start of sprint 1.

2.4.3 Meeting 2 (27^{th} March)

1. End of sprint 1.
2. Start of sprint 2.

2.4.4 Meeting 3 (3^{rd} April)

1. End of sprint 2.
2. Start of sprint 3.

2.4.5 Meeting 4 (10^{th} April)

1. End of sprint 3.
2. Start of sprint 4.

2.4.6 Meeting 5 (24^{th} April)

1. End of sprint 4.
2. Start of sprint 5.

2.4.7 Meeting 6 (3^{rd} May)

1. End of sprint 5.
2. Start of sprint 6.

2.4.8 Meeting 7 (8^{th} May)

1. End of sprint 6.
2. Start of sprint 7.

2.4.9 Meeting 8 (15^{th} May)

1. End of sprint 7.
2. Start of sprint 8.

2.4.10 Meeting 9 (22^{nd} May)

1. End of sprint 8.
2. Start of sprint 9.

2.4.11 Meeting 9 (29^{th} May)

1. End of sprint 9.

2.5 Team policy

The team has decided to hold weekly meetings with the client, throughout the week, the team members will try to establish meetings between them with the necessary duration to continue advancing in the project (estimated time: one hour).

2.6 Organization

2.6.1 Roles

- **Administrator:** Luis Gutiérrez Flores.
- **Analysts:** Jesús Rodríguez Heras and Nicolás Ruiz Requejo.
- **Designers:** Arantzazu Otal Alberro, Gabriel Fernando Sánchez Reina and Nicolás Ruiz Requejo.
- **Developers:** Luis Gutiérrez Flores, Alejandro Segovia Gallardo and Alejandro José Caraballo García.
- **Test Engineers:** Jesús Rodríguez Heras and Luis Gutiérrez Flores.

2.6.2 Hardware and software resources

As hardware resources we have the laptops of the 7 members of the group and the STIMEY server.

As software resources we have the Laravel framework, Atom, Visual Studio Code, TeXStudio, phpMyAdmin, MySQL, GitHub.

2.7 Costs

2.7.1 Human costs

- Hours in the learning of Laravel.
- PHP and MySQL training hours.
- GitHub training hours.
- Documentation hours.

2.7.2 Material costs

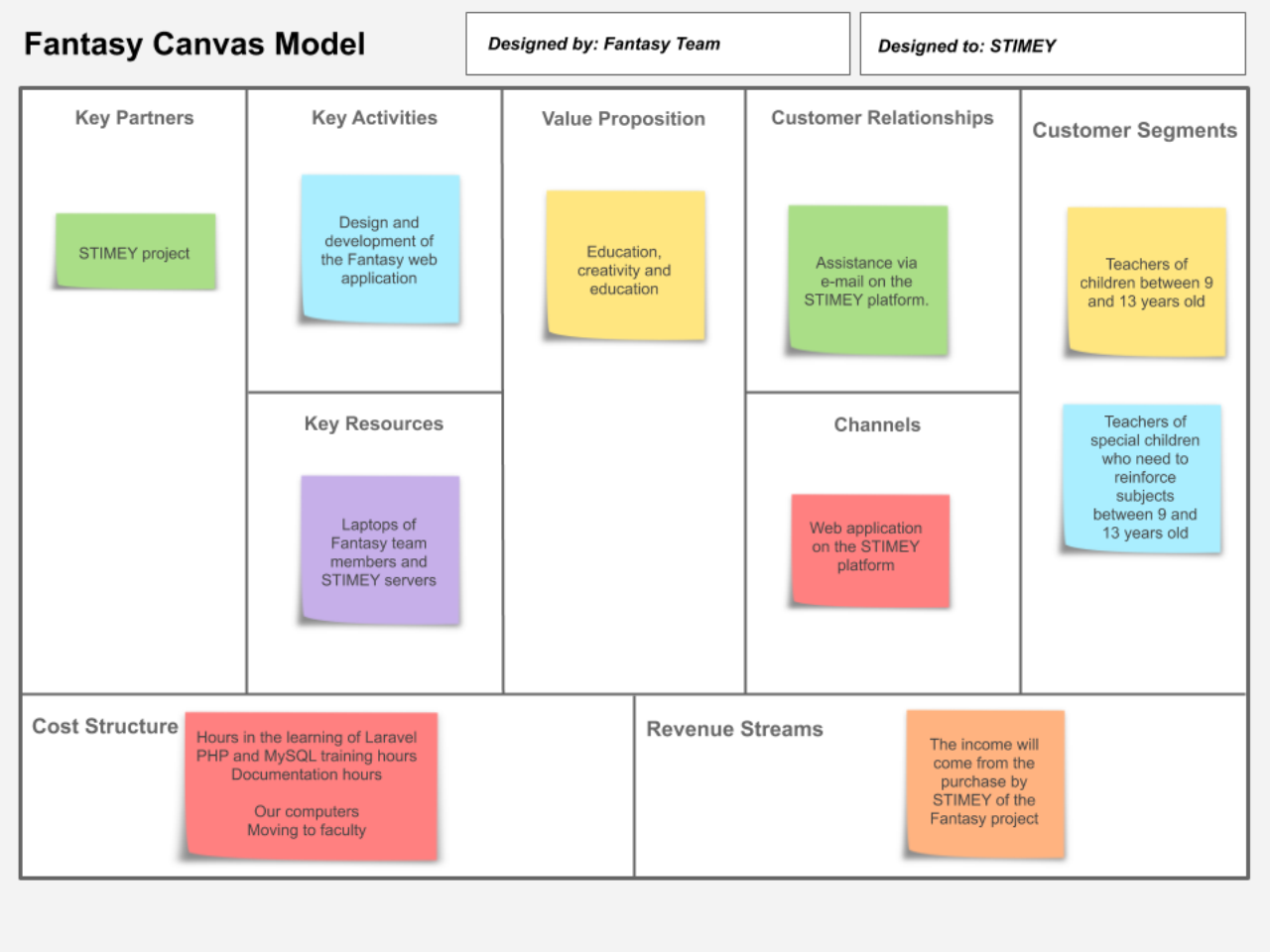
- Our computers.
- Moving to faculty.
- STIMEY server expenses.

2.8 Risk management

- Not meeting deadlines for trying to cover too much and leaving incomplete functionalities.

2.9 Canvas model

The canvas model of the Fantasy project is the following:



Part II

Developing

Chapter 3

Analisys of requirements

3.1 Agents catalogue

Fantasy project agents catalogue will be the students who interact with de webpage.

These agents can be sorted in two groups: students and teachers.

3.1.1 Teachers

- **Background:** They can open a window where you can select an image from the Internet, the computer or an image previously used. This image will cover all the workspace.
- **Active Point:** They can establish active points in the workspace and modify them conveniently by adding images, text, video, audio, etc. They can also establish a score for each active point of the fantasy to evaluate the students.

3.1.2 Students

- **Background:** They can do the same as the teachers.
- **Active Point:** They can establish active points in the workspace and modify them conveniently by adding images, text, video, audio, etc. You can not set a score for active points.

3.2 Functional requiriments

Characteristics of fantasies

- At the end of all active points there will be a button at the bottom right of "**more information**" and in the center there will be a new quiz that will be the final exam. This exam will have an independent score to all active points and will not have the statistical summary. If this quiz is repeated, the score would be updated with a percentage of the new score, plus the previous score with the objective that a student who repeats a quiz can not get the best grade due to repetition of it.
- The teachers can ask the students to make fantasies to learn as homework. These tasks can be done in groups of students based on two ideas:

1. **Compulsory:** A student make fantasy and the rest search for additional information.
2. **Optional:** Concurrent fantasy edition among all the members of the group.

- Each fantasy will have a code to be shared.
- We will have two types of permissions in the fantasies: "**see**" and "**see and edit**".
- The platform will notify the teacher when the students have finished their respective jobs.
- The fantasies may be private, shared or public. By default, they will always be public and can be accesible by everyone who uses the platform.
- Shared fantasies can be accesible by other people with a password.
- Fantasies can be cloned.

Characteristics of active points

- It will be possible to move it within the background and modify its contents.
- If an image is added to the active point, that point is adapted to the shape of the image.
- You can also assign a video, which will open a window to play it, or an audio. In case there is no audio or video, the respective button will not be displayed.
- Active points may have background music that will be muted if the audio or video playback assigned to that point by the teacher begins. The music will be restored when the corresponding audio or video ends.
- The active points can be reorganized by the teachers and students so that they emerge in the desired order.
- Through the realization (not creation) of a fantasy, a student can not continue with the next active point without finishing the current one.
- The active points quiz should be fun and intuitive.
- There should be only three questions in each quiz of each active point and they should not be too difficult (multiple answer, simple answer, write a word).
- The quiz of the active point will appear on the screen when that active point is closed.
- Once the quiz is finished, the next active point appears in the order established by the teachers/students in the workspace.
- Each active point will have a score to add (to all) a maximum of 100 points.
- When assigning a score to an active point, it will be subtracted from the total (maximum 100 points). If an active point is eliminated, the general counter recovers the score assigned to that active point.
- The student does not know the total number of active points in the fantasy.
- When the student obtains a score by completing an active point, this amount is added to the global counter.
- Finally, we can have a statistical summary with the right/wrong questions of each active point.
- Only the score obtained will be saved the first time a quiz is done, then it can be done more times, but the note will not be recorded in the system.
- The student will have the option to save his progress with a save button manually or through the auto-save option.

All the use cases described then have the following implicit precondition to be able to use said use cases in the final application:

- The user (teachers/students) must have an account on the STIMEY platform and have logged in with that account.

The proposed use cases are the following:

3.2.1 CRUD fantasy

Use case: Create fantasy

- **Description:** Create a new fantasy.
- **Agents:** Creator-editor (user).
- **Preconditions:** The user must have permission to create a new fantasy.
- **Postconditions:** Fantasy is stored in the system.
- **Main stage:**
  1. The user selects the option “Create new fantasy”.
  2. The system requests the name of the fantasy.
  3. The user enters the name of the fantasy.
  4. The system requests the fantasy code.
  5. The user enters the fantasy code.
  6. The system gives you the option to choose if the fantasy will be public (by default), shared or private.
  7. The user selects “ Public ”.
  8. The user creates the fantasy.
  9. The fantasy is stored in the system.
- **Extensions:**
  7. a) The user selects that the fantasy will be shared.
     1. The system allows to insert in a list the identifiers of other users with which the fantasy will be shared.
     2. The user enters the identifiers of the users who will share the fantasy.
     3. Step 8.

  7. b) The user selects that the fantasy will be private.
     1. The system marks the fantasy as private for that user without giving the possibility of sharing.
     2. Step 8.

  *a) At any time, the user can go back to the main menu.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Visualize fantasy

- **Description:** Read an existing fantasy.
- **Agents:** Creator-editor (user).
- **Preconditions:** The fantasy must exist in the system and the user must have the modification permissions.
- **Postconditions:** There are no changes in the fantasy.
- **Main stage:**
  1. The user selects the option “My fantasies”
  2. The system displays a list of fantasies accessible by the user.
  3. The user selects the fantasy that he wants to visualize.
  4. The system displays a pop-up window with the fantasy information and its options.
  5. The user selects the option “Visualize fantasy”.
  6. The system shows the fantasy.
  7. The user reads the fantasy without making any changes and, when it is over, closes the fantasy.
  8. The fantasy remains unchanged.
- **Extensions:**
  *a) At any time, the user can go back to the main menu.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Update fantasy

- **Description:** Modify an existing fantasy.
- **Agents:** Creator-editor (user).
- **Preconditions:** The fantasy must exist in the system and the user must have the modification permissions.
- **Postconditions:** The fantasy is modified.
- **Main stage:**
  1. The user selects the option “My fantasies”.
  2. The system displays a list of fantasies accessible by the user.
  3. The user selects the fantasy that he wants to modify.
  4. The system displays a pop-up window with the fantasy information and its options.
  5. The user selects the option “Modify fantasy”.
  6. The system displays the fantasy creation screen for modification.
- **Extensions:**
  *a) At any time, the user can go back to the main menu.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Delete fantasía

- **Description:** Erase an existing fantasy.
- **Agents:** Creator-editor (user).
- **Preconditions:** The fantasy must exist in the system and the user must have the removal permissions.
- **Postconditions:** Fantasy is eliminated from the system.
- **Main stage:**
  1. The user selects the option “My fantasies”.
  2. The system displays a list of fantasies accessible by the user.
  3. The user selects the fantasy that he wants to modify.
  4. The system displays a pop-up window with the fantasy information and its options.
  5. The user selects the option “Clear fantasy”.
  6. The system displays a confirmation message.
  7. The user selects “Accept”.
  8. The system erases the fantasy.
- **Extensions:**
  7. a) The user selects “Cancel”.
     1. The system closes the pop-up window.
     2. Step 1.

  *a) At any time, the user can go back to the main menu.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

3.2.2 Use case: Copy fantasy

- **Description:** Clone a fantasy.
- **Agents:** Creator-editor (user).
- **Preconditions:** The fantasy must exist in the system and the user must have the modification permissions.
- **Postconditions:** Create a copy of the selected fantasy.
- **Main stage:**
  1. The user selects the option “My fantasies”.
  2. The system displays a list of fantasies accessible by the user.
  3. The user selects the fantasy that he wants to copy.
  4. The system displays a pop-up window with the fantasy information and its options.
  5. The user selects the option “Copy fantasy”.
  6. The system creates a copy of the selected fantasy.
- **Extensions:**
  *a) At any time, the user can go back to the main menu.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

3.2.3 CRUD background

- **Description:** Allows selecting, modifying and deleting the background.
- **Agents:** Creator-editor (user).
- **Preconditions:** The fantasy must exist in the system and the user must have the modification permissions.
- **Postconditions:** The background that the user has chosen is established.
- **Main stage:**
  1. The user selects the "Background" option.
  2. The system displays a window to add a background to the workspace.
  3. The user selects an image.
  4. The system sets the background selected by the user.
- **Extensions:**
  *a) At any time, the user can go back to the main menu.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

3.2.4 CRUD active point

Use case: Create active point

- **Description:** Create a new active point.
- **Agents:** Creator-editor (user).
- **Preconditions:** The fantasy must exist in the system and the user must have the modification permissions.
- **Postconditions:** An empty active point is created in the workspace.
- **Main stage:**
  1. The user selects the option "New active point".
  2. The system creates a new active point in the workspace.
  3. The user can move the active point to the area of the workspace that he wants.
  4. The system will save the active point in the fantasy.
- **Extensiones:**
  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Visualize active point

- **Description:** Shows an existing active point for reading.
- **Agents:** Creator-editor (user).
- **Preconditions:** Fantasy must exist in the system and the active point must exist in fantasy. In addition, the user must have the modification permissions.
- **Postconditions:** The active point is shown for reading.
- **Main stage:**
  1. The user selects the active point that he wants to view.
  2. The system displays a window with the information of the active point and its options.
  3. The user selects the option “Visualize”.
  4. The system displays a window with the summary of that active point.
- **Extensions:**
  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Update active point

- **Description:** Modifies an existing active point.
- **Agents:** Creator-editor (user).
- **Preconditions:** Fantasy must exist in the system and the active point must exist in fantasy. In addition, the user must have the modification permissions.
- **Postconditions:** Modifies the selected active point.
- **Main stage:**
  1. The user selects the active point that he wants to modify.
  2. The system displays a window with the information of the active point and its options.
  3. The user selects the option “Modify”.
  4. The system displays the creation window of the active point.
- **Extensions:**
  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Delete active point

- **Description:** Deletes an existing active point.
- **Agents:** Creator-editor (user).
- **Preconditions:** Fantasy must exist in the system and the active point must exist in fantasy. In addition, the user must have the modification permissions.
- **Postconditions:** Deletes the selected active point.
- **Main stage:**
  1. The user selects the active point that he wants to delete.
  2. The system displays a window with the information of the active point and its options.
  3. The user selects the option “Delete”.
  4. The system displays a confirmation message.
  5. The user selects “Accept”.
  6. The system deletes the active point.
- **Extensions:**
  5. a) The user selects “Cancel”.
     1. The system closes the pop-up window.
     2. Step 1.

  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

3.2.5 CRUD image

Use case: Create image

- **Description:** Insert an image in an active point.
- **Agents:** Creator-editor (user).
- **Preconditions:** The corresponding active point must exist and the fantasy must be being edited.
- **Postconditions:** Insert an image in the selected active point.
- **Main stage:**
  1. The user selects the corresponding active point within the fantasy.
  2. The system displays a pop-up window with the information of the active point.
  3. The user selects the option “Insert image”.
  4. The system shows a window in which the user chooses from where they can insert the image (Internet, local, image already used in fantasy).
  5. The user chooses the option “Internet” to include an image of the Internet.
  6. The system asks the user for the url of the image.

7. The user inserts the correct url of the image.
8. The active point takes the shape of the image.

- **Extensions:**
  5. a) The user chooses the "Local" option to include an image from his computer.
     1. The system opens a file browser window.
     2. The user selects the image and press "Accept".
     3. The system closes the file browser window.
     4. Step 8.
  5. b) The user chooses the option "Image previously used" to include an image previously used.
     1. The system opens a window with the images previously used.
     2. The user selects the image and press "Accept".
     3. The system closes the pop-up window.
     4. Step 8.
  7. a) The url is not correct.
     1. The system displays an error message.
     2. Step 6.

  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Update image

- **Description:** Update an image.
- **Agents:** Creator-editor (user).
- **Preconditions:** There must exists the corresponding active point, you must be editing the fantasy and there must be an image.
- **Postconditions:** The image is modified.
- **Main stage:**
  1. The user selects the corresponding active point within the fantasy.
  2. The system opens a popup window with the information of the active point.
  3. Step 4 of Create image.
- **Extensiones:**
  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Delete image

- **Description:** Delete an image of an active point.
- **Agents:** Creator-editor (user).
- **Preconditions:** There must be the corresponding active point, you must be editing the fantasy and there must be an image.
- **Postconditions:** Delete the image and leave the active point in its default state.
- **Main stage:**
  1. The user selects the corresponding active point within the fantasy.
  2. The system opens a popup window with the information of the active point.
  3. The user selects the image and presses the "Delete" button.
  4. The system displays a confirmation message.
  5. The user selects "Accept".
  6. The system deletes the image of the active point.
- **Extensions:**
  5. a) The user selects "Cancel".
     1. The system closes the pop-up window.
     2. Step 1.

  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

3.2.6 CRUD video

Use case: Create video

- **Description:** Insert a video in an active point.
- **Agents:** Creator-editor (user).
- **Preconditions:** The corresponding active point must exist and the fantasy must be being edited.
- **Postconditions:** Insert a video in the selected active point.
- **Main stage:**
  1. The user selects the corresponding active point in the fantasy.
  2. The system displays a pop-up window with the information of the active point.
  3. The user selects the option "Insert video".
  4. The system shows a window in which the user chooses where to choose the image (Internet, local, video already used in fantasy).
  5. The user chooses the option "Internet" to include an Internet video.
  6. The system asks the user for the url of the video.

7. The user enters the correct url of the video.
8. The system saves the video in the active point.

- **Extensions:**
  5. a) The user chooses the "Local" option to include a video from his computer.
     1. The system shows a file browser sale.
     2. The user selects the image and press "Accept".
     3. The system closes the file browser window.
     4. Step 8.
  5. b) The user chooses the option "Video previously used" to include a video previously used.
     1. The system opens a window with the videos previously used.
     2. The user selects the video and press "Accept".
     3. The system closes the pop-up window.
     4. Step 4.
  7. a) The url is not correct.
     1. The system displays an error message.
     2. Step 6.

  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Update video

- **Description:** Update a video.
- **Agents:** Creator-editor (user).
- **Preconditions:** There must be the corresponding active point, you must be editing the fantasy and there must be a video.
- **Postconditions:** The video is modified.
- **Main stage:**
  1. The user selects the corresponding active point in the fantasy.
  2. The system opens a pop-up window with the information of the active point.
  3. Step 4 of Create video.
- **Extensions:**
  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Delete video

- **Description:** Delete a video of an active point.
- **Agents:** Creator-editor (user).
- **Preconditions:** There must be the corresponding active point, you must be editing the fantasy and there must be a video.
- **Postconditions:** Delete the video of the active point.
- **Main stage:**
  1. The user selects the corresponding active point within the fantasy.
  2. The system opens a popup window with the information of the active point.
  3. The user selects the video and presses the "Delete" button.
  4. The system displays a confirmation message.
  5. The user selects "Accept".
  6. The system deletes the video from the active point.
- **Extensions:**
  5. a) The user selects "Cancel".
     1. The system closes the pop-up window.
     2. Step 1.

  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

3.2.7 CRUD text

- **Description:** Insert a text in an active point.
- **Agents:** Creator-editor (user).
- **Preconditions:** The corresponding active point must exist and the fantasy must be being edited.
- **Postconditions:** Insert a text in the selected active point.
- **Main stage:**
  1. The user selects the active point to which he wants to add-edit the text.
  2. The system displays a pop-up window with the information of the active point.
  3. The user selects enter the text in the "Text" field with the formatting options that you want.
  4. The user clicks on the "Accept" button.
  5. The system saves the text in the corresponding active point.
- **Extensions:**
  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

3.2.8 CRUD quiz

Use case: Create quiz

- **Description:** Create a small questionnaire about the theme of the active point.
- **Agents:** Creator-editor (user).
- **Preconditions:** The corresponding active point must exist and the fantasy must be being edited.
- **Postconditions:** Create a small questionnaire in relation to the corresponding active point.
- **Main stage:**
  1. The user selects the corresponding active point.
  2. The system displays a pop-up window with the information of the active point.
  3. The user selects the option “Create quiz”.
  4. The system shows the possible options.
  5. The user selects “Simple answer”.
  6. The system displays a pop-up window to create the question with its possible answers.
  7. The user populates the pop-up window with the question and the appropriate answers and press “Accept” when it finishes.
  8. The system closes the pop-up window.
  9. The questionnaire is registered in the selected active point.
- **Extensions:**
  3. a) The user chooses the option “Word”.
     1. The system opens a pop-up window to create the question and its answer.
     2. The user fill in the pop-up window with the question and the appropriate answer and press “Accept” when it finishes.
     3. Step 8.

  3. b) The user chooses the option “Quiz with images”.
     1. The system opens a pop-up window to create the question with the image and its response.
     2. The user fills in the pop-up window with the question, the image and the appropriate answer, and press “Accept” when it finishes.
     3. Step 8.

  3. c) The user chooses the “Join” option.
     1. The system opens a pop-up window to create the join quiz.
     2. The user populates the pop-up window with the possible answers and their correct answer and press “Accept” when it finishes.
     3. Step 8.

  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Visualize quiz

- **Description:** Shows the status of quiz.
- **Agents:** Creator-editor (user).
- **Preconditions:** The corresponding active point must exist, the fantasy must be edited and there must be a quiz.
- **Postconditions:** Displays the status of quiz at the corresponding active point.
- **Main stage:**
  1. The user selects the corresponding active point.
  2. The system displays a pop-up window with the information of the active point.
  3. The user selects the option “Read quiz”.
  4. The system displays a pop-up window with the final view of quiz.
- **Extensions:**
  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Update quiz

- **Description:** Update the quiz.
- **Agents:** Creator-editor (user).
- **Preconditions:** The corresponding active point must exist, the fantasy must be edited and there must be a quiz.
- **Postconditions:** Modify the quiz of an active point.
- **Main stage:**
  1. The user selects the corresponding active point.
  2. The system displays a pop-up window with the information of the active point.
  3. The user selects the option “Modify quiz”.
  4. Step 4 of Create quiz
- **Extensions:**
  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Delete quiz

- **Description:** Delete the quiz of the selected active point.
- **Agents:** Creator-editor (user).
- **Preconditions:** The corresponding active point must exist, the fantasy must be edited and there must be a quiz.
- **Postconditions:** Delete the quiz of the selected active point.
- **Main stage:**
  1. The user selects the corresponding active point.
  2. The system displays a pop-up window with the information of the active point.
  3. The user selects the option “Delete quiz”.
  4. The system displays a confirmation message.
  5. The user selects “Accept”.
  6. The system deletes the quiz of the active point.
- **Extensions:**
  5. a) The user selects “Cancel”.
     1. The system closes the pop-up window.
     2. Step 1.

  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

3.2.9 CRUD audio effect

Use case: Create audio effect

- **Description:** Set a background audio effect on the active point.
- **Agents:** Creator-editor (user).
- **Preconditions:** The corresponding active point must exist and the fantasy must be being edited.
- **Postconditions:** Set the background audio effect.
- **Main stage:**
  1. The user selects the corresponding active point.
  2. The system displays a pop-up window with the information of the active point.
  3. The user selects the option “Add audio effect”.
  4. The system displays a pop-up window in which you select the user from where you want to select the audio (Internet, local, audio already used in the fantasy).
  5. The user chooses the option “Internet” to include an Internet audio.
  6. The system asks the user for the audio url.
  7. The user inserts the audio url.

8. The system saves the audio in the active point.

- **Extensions:**
  5. a) The user chooses the option “Local” to include an audio from his computer.
     1. The system opens a file browser window.
     2. The user selects the audio and press “Accept”.
     3. The system closes the file browser window.
     4. Step 8.
  5. b) The user chooses the “Audio previously used” option to include an previously used audio.
     1. The system opens a window with the previously used audios.
     2. The user selects the audio and press accept.
     3. The system closes the pop-up window.
     4. Step 8.
  7. a) The url is not correct.
     1. The system displays an error message.
     2. Step 6.

  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Update audio effect

- **Description:** Update an audio effect.
- **Agents:** Creator-editor (user).
- **Preconditions:** There must exist the corresponding active point, you must be editing the fantasy and there must be an audio.
- **Postconditions:** Update the audio effect.
- **Main stage:**
  1. The user selects the active point.
  2. The system opens a pop-up window with the information of the active point.
  3. Step 4 of Create audio effect
- **Extensions:**
  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Delete audio effect

- **Description:** Delete an audio effect from an active point.
- **Agents:** Creador-editor (usuario).
- **Preconditions:** There must exist the corresponding active point, you must be editing the fantasy and there must be an audio.
- **Postconditions:** Deletes an audio effect from an active point.
- **Main stage:**
  1. The user selects the active point.
  2. The system opens a pop-up window with the information of the active point.
  3. The user selects the audio and presses the "Delete" button.
  4. The system displays a confirmation message.
  5. The user selects "Accept".
  6. The system deletes the audio from the active point.
- **Extensions:**
  5. a) The user selects "Cancel".
     1. The system closes the pop-up window.
     2. Step 1.

  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

3.2.10 CRUD additional information

- **Description:** Insert a text as additional fantasy information.
- **Agents:** Creator-editor (user).
- **Preconditions:** You must be editing the corresponding fantasy.
- **Postconditions:** Insert a text as additional fantasy information.
- **Main stage:**
  1. The user selects the option "Additional information".
  2. The system displays a pop-up window with a text box.
  3. The user enters the desired text in the text box with the desired formatting options.
  4. The user clicks on the "Accept" button.
  5. The system saves the additional information in the corresponding fantasy.
- **Extensions:**
  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

3.2.11 Use case: Organize active points

- **Description:** Organise the appearance of active points.
- **Agents:** Creator-editor (user).
- **Preconditions:** The fantasy must be created.
- **Postconditions:** It establishes the order of appearance of the active points of the fantasy.
- **Main stage:**
  1. The user selects the corresponding fantasy.
  2. The system displays a window with the fantasy information and available options.
  3. The user selects the option “Organize active points”.
  4. The system displays a pop-up window with the name of the existing active points in the fantasy and a box to establish the order of appearance.
  5. The user establishes the order of appearance in the boxes next to the name of the active points of the fantasy.
  6. The user presses “Accept” to save the changes made.
  7. The system keeps the order of appearance of the active points.
- **Extensions:**
  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

3.2.12 CRUD final quiz

Use case: Create final quiz

- **Description:** Create a questionnaire about the theme of fantasy.
- **Agents:** Creator-editor (user).
- **Preconditions:** There must exist and they must be editing the fantasy.
- **Postconditions:** Create a questionnaire related to the theme of fantasy.
- **Main stage:**
  1. The user presses the “Final questionnaire” button.
  2. The system shows the possible options.
  3. The user selects “Create new quiz”.
  4. The system shows the possible creation options.
  5. The user selects “Simple answer”.
  6. The system displays a pop-up window to create the question with its possible answers.
  7. The user populates the pop-up window with the question and the appropriate answers and press “Accept” when it finishes.
  8. The system closes the pop-up window.

9. The questionnaire is recorded in the fantasy.

- **Extensions:**
  5. a) The user chooses the option "Word".
     1. The system opens a pop-up window to create the question and its answer.
     2. The user populates the pop-up window with the question and the appropriate answer and press "Accept" when it finishes.
     3. Step 8.
  5. b) The user chooses the option "Quiz with images".
     1. The system opens a pop-up window to create the question with the image and its response.
     2. The user fills in the pop-up window with the question, the image and the appropriate answer, and press "Accept" when it finishes.
     3. Step 8.
  5. c) The user chooses the "Join" option.
     1. The system opens a pop-up window to create the join quiz.
     2. The user populates the pop-up window with the possible answers and their correct answer and press "Accept" when it finishes.
     3. Step 8.

  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Visualize final quiz

- **Description:** Show the status of the final quiz.
- **Agents:** Creator-editor (user).
- **Preconditions:** The fantasy must exist and they must be being edited, and the final quiz must exist.
- **Postconditions:** Show the status of the final quiz.
- **Main stage:**
  1. The user selects the "Final questionnaire" button.
  2. The system shows the possible options.
  3. The user selects the option "Read quiz final".
  4. The system displays a pop-up window with the final view of quiz.
- **Extensions:**
  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Update final quiz

- **Description:** Update the final quiz.
- **Agents:** Creator-editor (user).
- **Preconditions:** The fantasy must exist and they must be being edited, and the final quiz must exist.
- **Postconditions:** Modify the final quiz of the fantasy.
- **Main stage:**
  1. The user presses the "Final questionnaire" button.
  2. The system shows the possible options.
  3. The user selects the option "Modify quiz final".
  4. Use case Create final quiz
- **Extensions:**
  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

Use case: Delete final quiz

- **Description:** Delete the final quiz of the fantasy.
- **Agents:** Creator-editor (user).
- **Preconditions:** The fantasy must exist and must be being edited, and the final quiz must exist.
- **Postconditions:** Delete the final quiz of the fantasy.
- **Main stage:**
  1. The user presses the "Final questionnaire" button.
  2. The system shows the possible options.
  3. The user selects the final "Delete quiz" option.
  4. The system displays a confirmation message.
  5. The user selects "Accept".
  6. The system deletes the final quiz of the fantasy.
- **Extensions:**
  5. a) The user selects "Cancel".
     1. The system closes the pop-up window.
     2. Step 1.

  *a) At any time, the user can go back.
- **Variations:** None.
- **Not-functional:** None.
- **Issues:** None.

3.3 Information requirements

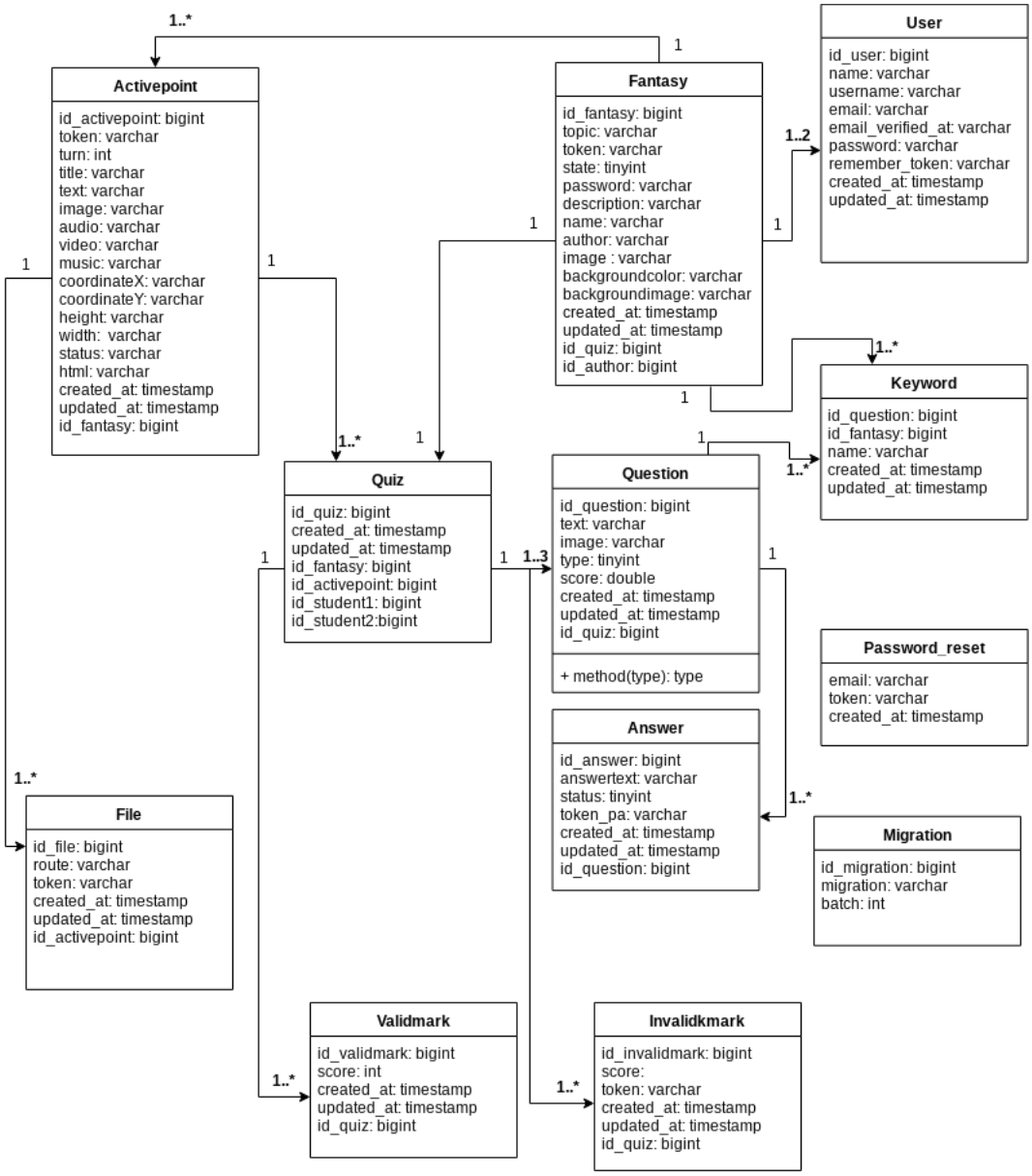


Figure 3.1: Stimey UML.

3.4 Non-functional requirements

3.4.1 Efficiency

- All system functionality and business transaction must respond to the user in less than 5 seconds.
- The system is able to operate properly with several users with concurrent sessions.
- The modified data in the database must be updated for all users who access in less than 2 seconds.

3.4.2 Logical and data security

- The access permissions to the system may only be changed by the data access administrator.
- The passwords of the users for their login are encrypted.

3.4.3 Usability

- The learning time of the system by a user is less than 2 hours.
- The system has properly structured user manuals.
- The web application has a "Responsive" design to ensure proper viewing on multiple personal computers, tablet devices and smartphones.
- The system has well-formed graphical interfaces.

3.4.4 Dependibility

- The system must have an availability of 99.99% of the time a user tries to access it once it is launched on the STIMEY servers.
- The time to start or restart the system is less than 2 minutes.
- The probability of failure of the system can not be less than 0.05%.

3.5 Business rules

Not applicable.

3.6 Study of technological alternatives

As a technological alternative, the web application could be proposed as a REST server that is responsible for receiving the requests and resolving them concurrently, which would give greater modularity to the system.

In addition, in case of failure, it would be known where the fault is most easily due to the REST system.

3.7 GAP analysis

Not applicable.

Chapter 4

System design

4.1 Architecture design

4.1.1 Phisical architecture

The phisical architecture used for the project development has been the computers from the team Fantasy members, which have worked as users as well as server.

Once the application is deployed and integrated with STIMEY project, the phisical architecture will be STIMEY servers.

4.1.2 Logical architecture

For logical architecture, we have used libraries and functionalities provided by Laravel framework and Javascript.

4.1.3 Design architecture

Both architectures previously mentioned join together in the view-controller model used in project Fantasy development. This provide many views from the clients side with their respective controller.

This makes that the application can be implemented litle by litle and with guarantee of functionality in every one of their steps.

4.2 User interface desing

The user interface design used in project Fantasy follows the rules of design of STIMEY both in the creation view and in the playing view of the fantasy.

We have reached an agreement with STIMEY project members so they give us the rules of stile and so we can make them resemble at the most, because the finality of the project is to integrate it in their plataform.

The performance of this interface will be regulated by the view-controller model used in their development.

4.3 Data design

Regarding data design, we have proposed the following diagram for the database of Fantasy project.

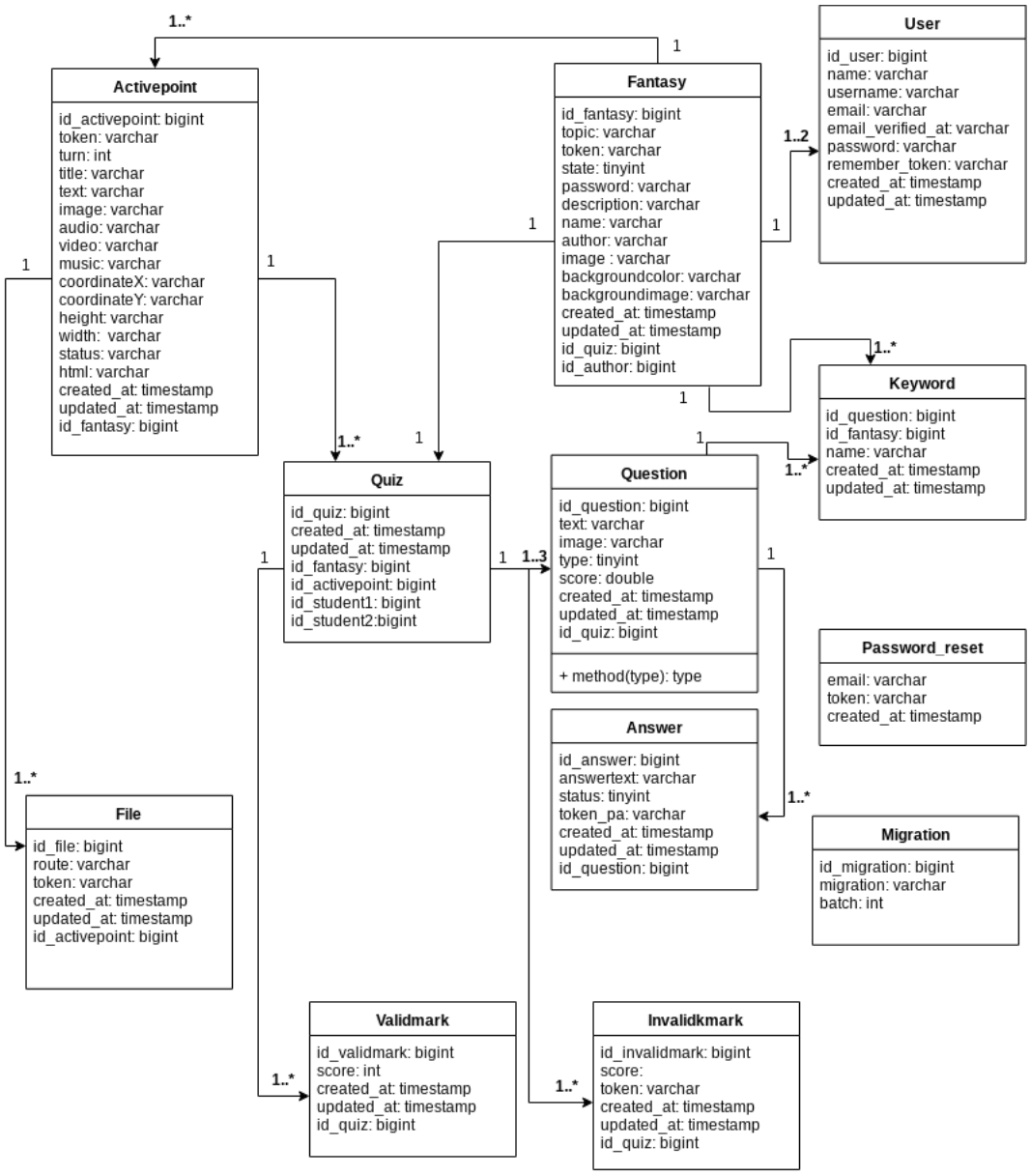


Figure 4.1: Stimey database.

4.4 Design of components

The components used in the development of Fantasy project are: Laravel framework, MySQL and phpMyAdmin.

Thanks to this elements, we could incorporate the necesary views and controllers for the application functioning.

4.5 Parameterization of the base software

Not applicable.

Chapter 5

Implementation of the system

5.1 Technological environment

The technological environment of programming used in Fantasy project has been the framework Laravel, which has an integration with PHP and MySQL.

Thanks to PHP, we have made that the Fantasy project webpage was the most similar to STIMEY's for its following integration in the plataform.

And, with MySQL, we manage the database of the fantasies which students and teachers make, being in control of the property and of the access to the fantasies.

5.2 Source code

Source code of application is in the compressed file that we attached to the task.

5.3 Code quality

To see the quality of the code we used the SonarQube tool.

This tool allows you to check the quality of the code hosted on GitHub online.

The results have been the following:

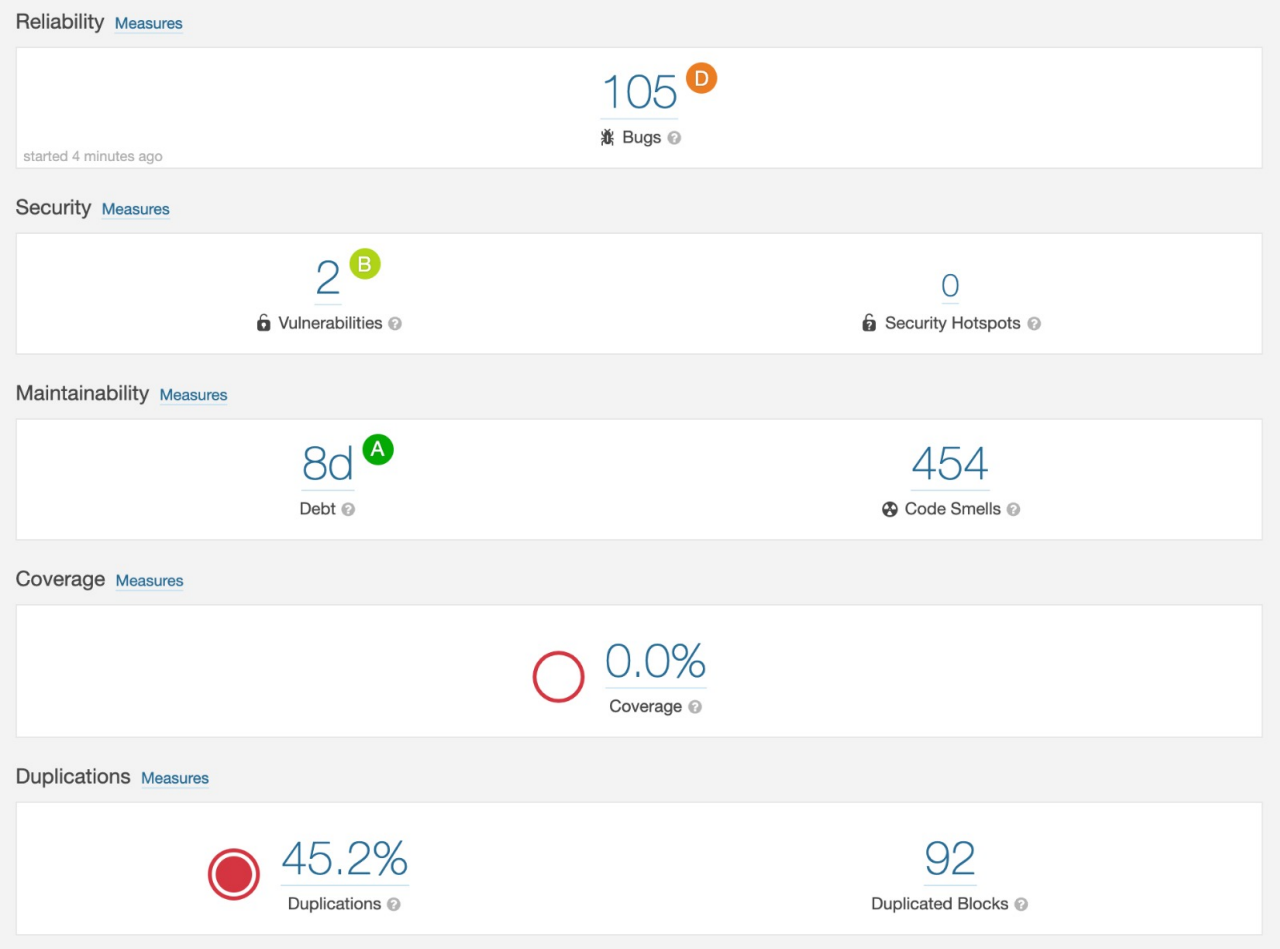


Figure 5.1: SonarQube results.

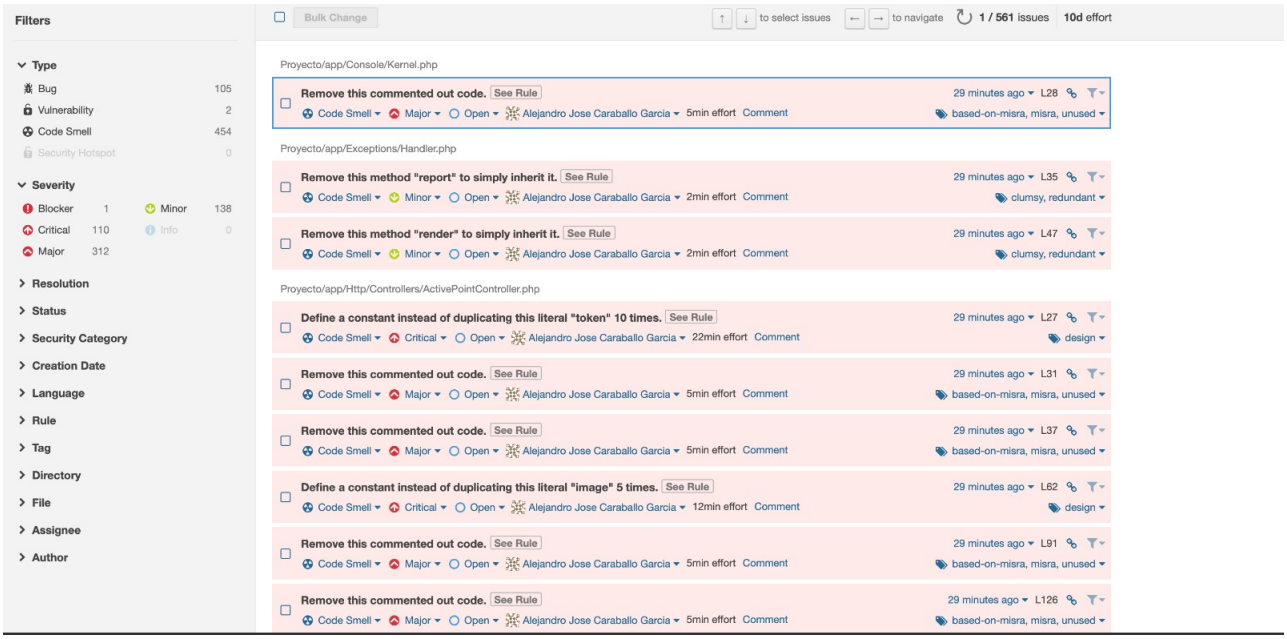


Figure 5.2: SonarQube rules.

Chapter 6

System tests

6.1 Unit tests

For the unit tests we have used the Laravel framework that allows the automation of tests through code.

This improves the confidence of the code and helps the application to be as secure as possible.

6.2 Integration testing

In the integration tests we have used the seeders, which mimic the behavior of real objects in a controlled manner. These objects are also useful when real objects have not yet been developed, are very expensive to instantiate or are not available.

We will use the Laravel framework just like in the automated unit tests.

6.3 System tests

The system tests have been checked thanks to the testing tools provided by Laravel, which, like the unit tests, have been program to be linked automatically.

6.3.1 Functional testing

We will perform functional tests in order to check whether the system performs correctly the functionality described in the requirements. Black box tests will be carried out.

Throughout the development of the project, manual tests will be carried out to verify the proper functioning of each of the scenarios of each established use case and the next milestone will not be advanced until it has been verified that everything is working correctly.

6.3.2 Non-functional tests

The purpose of the non-functional tests is to check whether the system (integrated and complete) meets the non-functional requirements previously established in the requirements analysis. For this, they will carry out the following types of tests:

- Efficiency: Monitoring of resource consumption, load and stress tests on the server and web performance tests will be carried out.
- Logical an data security: To guarantee the authenticity, confidentiality and integrity of the data, as well as responsibility and non-repudiation, we will carry out an exhaustive check of the read and write permissions on the system data and operations execution.
- Usability: We will carry out an analysis of the web through heuristic rules, we will study the results working with real users, through field observations, interviews and questionnaires. They will also be tested for use by specific browsers or by manipulating their properties and automatic validations will be made.
- Dependibility:There will be a static analysis of the code, detection of bugs, security vulnerabilities, code smells, complexity metrics, size, test coverage, etc.

6.4 Acceptance Tests

We have been making the acceptance tests with our client of this project in each of the meetings, on which they argumented the changes they wanted and accepted or rejected our proposals.

Part III

Epilogue

Chapter 7

User manual

7.1 Introduction

This is the manual that the users may follow in order to use the application[1].

Having in mind that this will be a web application, the first step will be going to the web link where the service is set[2].

7.2 Features

The Fantasy application allows the user to create fantasies so the students learn in a more creative and funny way, with active points, quizzes associated to each active point and a final quiz associated to each fantasy.

This score will be send to STIMEY's platform in order to be stored in each student profile.

The students can also create fantasies which are asked by their teacher as a task, this task could be in couples or individually and it can be evaluated afterwards by their teacher.

7.3 Previous requirements

The previous requirements when using the Fantasy project application are to enter in the weblink where the service is located and to register with the account of the user who is going to use the application.

By that means, it is not necessary to install anything in the user's computers because the application is located in a server.

7.4 Utilization

When using the application, and having accessed with our user and password to the platform, we will see a main screen where we could see our fantasies (the ones that we have created) and the fantasies that we have marked as favourites to play again.

[1] For more detail about the user manual, see the appendix.

[2] If it has been integrated in STIMEY's platform, it will be in laboratory zone.

7.4.1 Create fantasy

If we want to create a fantasy, we click the "Create a new fantasy" icon.

Next, we fill in the necessary fields and the fantasy will be created.

Once we have finished with the creation of the fantasy, we could go back and see it in the section reserved for our fantasies.

7.4.2 Modify fantasy

If we want to modify a fantasy, we click the "Edit fantasy" icon.

Next, we modify the fields that we wish to change.

7.4.3 Delete fantasy

If we want to delete a fantasy, we click the "Delete fantasy" icon.

7.4.4 Create active point

If we want to create an active point, we click the "Create active point" icon.

Next, we fill in the necessary field and we will have created the first active point.

We can create a maximum of ten active points following the steps mentioned above for each active point.

7.4.5 Modify active point

In active points we will have the possibility of moving and resize the active point as we like.

Moreover, if we double click on them, we will have the possibility of changing the content of its fields.

7.4.6 Search fantasy

We can look for fantasies according to their theme and difficulty.

7.4.7 Play fantasy

If we want to play fantasy (created by ourselves or by another user) we click the "Play fantasy" icon and we could play the fantasy[3].

[3]We need to have in mind that for the students, we will only save the first score they get, although they can repeat the fantasy as many times as they wish.

Chapter 8

Installation and operation manual

8.1 Introduction

Due to the fact that Fantasy project is not an application that can be installed in a personal computer, but it is developed as a web application, it will have to be installed in a server.

8.2 Previous requirements

In order to install the application in the server, we need the Laravel framework, PHP and MySQL. Once those elements are installed, we will only have to launch the application from the directory of the project.

8.3 Inventory of components

The necesary components to launch Fantasy application would be:

- Laravel framework.
- PHP.
- MySQL.
- phpMyAdmin (optional).

8.4 Installation procedures

In the Laravel installation itself, we will be installing both PHP and MySQL. For that reason, we will only need to follow the following tutorial.

8.5 Implantation tests

The implementation tests have been tested personally by the Fantasy team, verifying the possible configurations that could lead to an error, and they have been resolved mostly.

Those tests have been made in the Fantasy team's laptops.

Chapter 9

Conclusions

9.1 Objetives

Fantasy project objetives are to make the students learn in a more creative and funny way through the creation of fantasies, and dinamically through the making of these fantasies and their active points.

9.2 Lessons learned

Regarding the lessons learned, all the group members have worked hard in the making of Fantasy project, focusing in the programation of the project for many hours. This has make us learn that a good initial organization is fundamental when covering a project of these dimenssions and with a client from the laboral world which distance us from the educative environment that the most of us are familiarized with.

Because of that, time management and task delegation have been essential during this project.

9.3 Future work

As future work shall remain the implementation of Fantasy web application in STIMEY's platform, so that way it will belong to their laboratories and it could be used by all users (both teaches and students) who wish to learn about a theme in particular or to used in class to show their students in a more interactive, creative and funny way.

Chapter 10

Bibliography

- Laravel documentation.
- Javascript documentation.
- LaTeX documentation.
- PHP documentation.
- SonarQube tool.