Requirements Analysis

Jogos Matemáticos

Course:

PI – Projeto em Informática

Students:

93221: Pedro Santos
93367: Rafael Baptista
93283: Pedro Amaral
93118: Ricardo Cruz
95278: Diogo Cunha
92969: Diogo Carvalho

Index

1 Requirements gathering

In order to fully grasp what the problem was, as well as the possible solutions to solve it, we conducted sessions of brainstorming and meetings with both our advisors, professors Nuno Lau and Diogo Gomes, as well as Professor Pedro Pombo, manager of Fábrica Centro Ciência Viva de Aveiro.

On our initial sessions, we began by trying to understand what the problem was, both by reading the available project guidelines, as well as by research what the different games were, how they were currently played, who played them and under what conditions they were played, either in a more formal or informal way.

We also discussed previous experiences with similar apps (both positive and negative) so that we could understand and aspects from the already available resources, this part was challenging at first since we encountered no apps where the games we wanted to implement could be played, and we had no access to physical representations. We adopted a new strategy which was to analyze other board games, since most features could be adapted to fit our own games.

The meetings we conducted were also very informative since we could extract knowledge from both our advisors, which were more familiar with both the project and other similar ones, and Pedro Pombo, which is responsible for the organization of math games tournaments conducted at FCCV. These meetings made us aware of who our target audience was, what concerns we should have in regard to user's data privacy, user management requirements (for example, a report/ ban system) and other aspects we should consider based on our target audience.

From these sessions we concluded a few things, first, that our target audience would be mostly young children, who usually participated in these math games tournaments, that we should we would try to present a familiar UI taking inspiration from the FCCV, that we would need to implement ways to compel users to come back to our app (through gamification aspects), that a report and ban system would be essential for our project to work and not become filled with griefers and cheaters and that we would need to comply with GDPR rules in regards to users privacy and data protection. We also established some of the technologies we would use to develop the project, mainly deciding the use of Phaser 3 to develop the game engines, and the use of electron to package our web app into a desktop app, with local modules to allow offline play storing relevant data.

2 Context

This section presents the description of how the system is expected to be used by its stakeholders.

2.1 Kids/ Teenagers

Kids and teenagers are the people that will utilize the main service our system provides which is playing games, they will be able to do so either with or without a registered account. Games can be played offline (personally with a friend or against an AI available in 3 different levels of skill) or online (against a friend through an invitation or against another random player), they will also be able to participate in available tournaments. If the user creates an account, they will have access to their match history (wins/ losses), their rating (skill level based on match history) and their account level (increased by playing games). They should also be able to customize their profile and/ or game boards through unlockable avatars and themes.

2.2 Tournament Organizers

Regarding institutions that are interested in using our application to host their own tournaments, we will present them with a solid and stable platform capable of handling multiple tournaments with a set of useful features. In this set of features, we give them the ability to invite/ expel specific users, make their tournaments private and report players that may be having a bad behavior.

2.3 Member of FCCV

Finally, when it comes to FCCV, we have a specific administrative area. This area constitutes our platform's control panel where we are able to ban/ unban players directly, watch a game's individual statistics as well as general platform statistics, which they can use to improve the application and better understand user's interests.

3 State of the Art

3.1 Digital Web Games

In order to better understand what had already been done, what worked and what didn't in terms of functionalities, we researched some of the available services which had similar characteristics to those we are going to develop for our own app. We began by taking a look at apps that members of our team already knew and used, starting with some of the popular chess websites, like chess.com and lichess.org, because even though chess is not one of the games we are going to develop, all board games have some similar characteristics, therefore we could extract some knowledge from these popular services, we also analyzed other services recommended by our advisors like Kaggle.com.

These services helped us better understand the current state of web games (useful/ interesting functionalities), the state of web tournaments, what relevant information is stored and shown relative to user profiles and what gamification methods are used.

3.2 Math Games

In order to better understand what and how currently recognized math games were played, we researched several resources available online and spoke with Professor Pedro Pombo, manager of Fábrica Centro Ciência Viva de Aveiro.

From our research we reached the conclusion that current math games like Rastros, Cães e Gatos, Yoté, and others, have not yet made the transition to the digital space, meaning they are solely played with a physical board or a paper with the board layout printed on it, this meant that there is still a lot of room for improvement. Previous tournaments were also organized in person at the FCCV which meant that during the pandemic, it was no longer viable to organize them.

3.3 Technologies Used

We also analyzed what current technologies are the most popular for developing web applications/ web games. Most resources and websites only listed HTML, CSS, and JS, but we were looking for a framework which could simplify some of the code we needed to write. With the suggestion from our advisors, we took a look at Phaser 3, a popular JS framework used in web game development. This framework turned out to be perfect for us since it had a lot of documentation, lots of game examples/ tutorials and it simplified the development of our games and their functionalities.

4 Actors

With the development of our system we are mainly targeting kids and teenagers who will make use out of the several games we are going to provide, we will be creating a simple UI to help users with less technological knowledge/ experience, without the need to create an account since they may not yet possess a personal email. From this requirement we can extract one of our actors, which is a **user without an account**. These users will be able to play matches (from all available games) with other users on the platform, they will also have access to our offline tools, which are playing vs AI and playing with a friend on the same device, their progress however, will not be stored.

We hope to capture these users' attention by providing a fun, competitive and engaging environment, which will hopefully lead to the creation of an account, generating our second actor, a **user with an account**. These users will have access to all of the functionalities stated above, while also having their progress stored so they don't risk losing it when transferring between different devices. The match history will also be available, they will have a rank (increased/ decreased by winning/ losing games), an account level (increased by playing games) and will have access to limited content unlocked by leveling up their account. They will also have the ability to participate in tournaments.

Our third actor is a **user with tournament privileges**, these users have the additional functionality of creating and managing tournaments, these privileges are granted by a system admin.

The actors stated above will have access to the offline functionalities through the web and desktop apps, online functionalities will be available through all technologies, web app, desktop app and mobile app.

We are also targeting the **FCCV staff (system admins)**, by providing a way to organize tournaments through the internet without need for the physical presence of participants, and, even if they are hosted at the FCCV, competitors could be provided with computers to play out the games. Organizers will also have the ability to manage users by banning those who are misbehaving or disrespecting rules. These users will be mostly using our web and desktop app.

5 Personas

In this section we are going to present personas based on the actors defined in the section above.

Rui is an 8 year old shy kid who loves board games, he knows how to play chess and checkers, which he mostly played with his friends at school before the pandemic started.

Motivation: Rui is looking for a website where he can play fun and interesting board games with other people, he doesn't have an email so he can't use websites that require an account. Since he is shy, he prefers to learn how to play the games before challenging other people, so he also wants to be able to train alone.

Ricardo is a very competitive teenager, he excels in most of his classes and always works hard to accomplish his goals. Lately he's been feeling like there is no competition for him at school, so he wants a different way to compete with others.

Motivation: Ricardo wants an app where he can challenge and be challenged, he wants to be able to use it anywhere so that he can always feel the thrill of competition. He wants to be able to visually see his improvement while competing against similarly matched people.

Ana is a middle school teacher who wishes to organize a fun activity with his students, she is always looking for innovative ways of teaching and entertaining the minds of her students, and she believes board games are a great way of stimulating creativity and problem solving skills.

Motivation: Ana wants to be able to create and manage multiple board game tournaments intuitively, she also wants to invite as many or as little students as she would like.

Artur is one the members of Fábrica Centro Ciência Viva de Aveiro, he was unable to host the math games tournament this year due to regulations that took place due to the pandemic.

Motivation: Rui wants to have control over an application where users can play the recognized math games, he also wishes to be able to organize official digital tournaments of those games. He also wants access to important statistics to view which games are the most popular.

Icons made by Freepik from @flaticon

6 Use Cases

In this section, we are going to present our use case diagrams of the whole system, where we can have a look into how our actors will interact with the system, we'll also give a description of each use case and its priority. We have divided the use cases in two sets of actors: Regular users (Figure 1) and Administrator users (Figure 2).

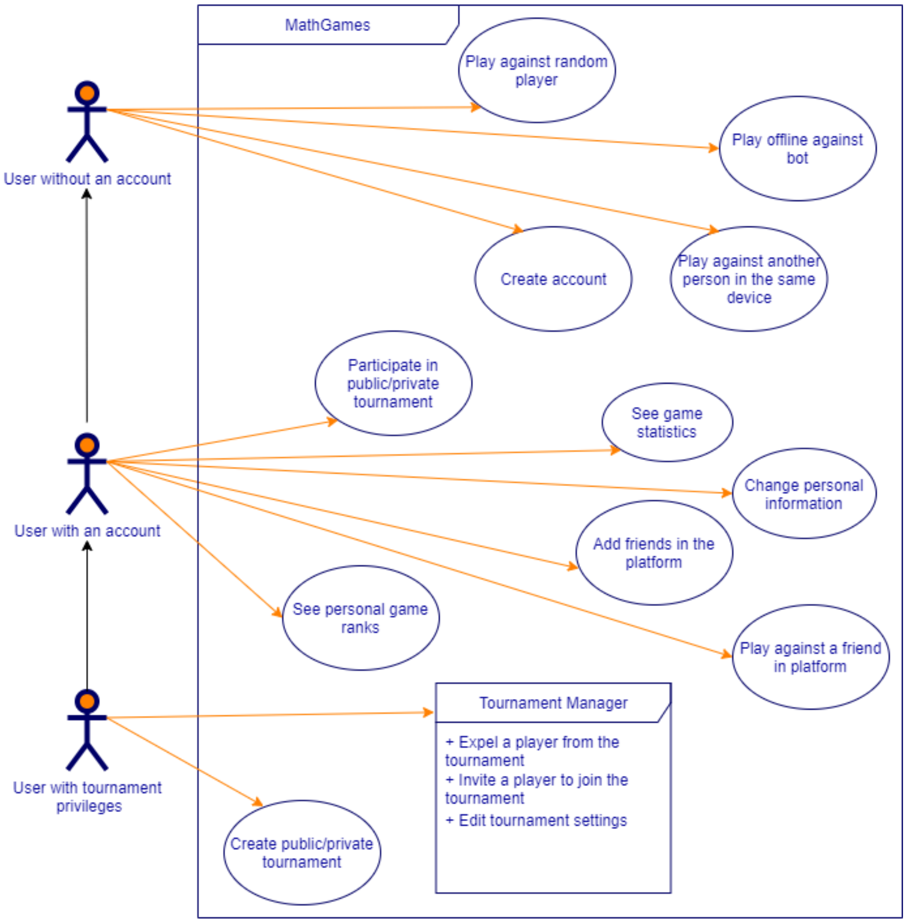
MathGames
User without an account
User with an account
User with tournament privileges
Play against random player
Play offline against bot
Create account
Play against another person in the same device
Participate in public/private tournament
See game statistics
Change personal information
Add friends in the platform
See personal game ranks
Play against a friend in platform
Tournament Manager
+ Expel a player from the tournament
+ Invite a player to join the tournament
+ Edit tournament settings
Create public/private tournament

Figure 1 - Regular Users Use Cases Diagram

- **<u>Play Against Random Player</u>**

Rui intends to play Avanço against a random person, in order to do this, he must first access our main page, he must then press a button present on that page or, through the side menu, access the games option. After reaching the games page he will be able to choose what game he wants to play, which will then show the multiple game modes for that specific game, after choosing the mode for online versus another player, the matchmaking process will begin and after a while the game will begin.

Priority: High

- **Play Offline Against AI**

Rui wants to improve his skills in Rastros by playing against the computer. he must first access our main page, he must then press a button present on that page or, through the side menu, access the games option. He can then choose the game he wants to play and, on the page of the game he chose, click on the respective mode.

Priority: High

- **Play Against Someone on the Same Device**

Rui wants to play Yoté against his brother on a computer without access to the internet. In order to do this, he must first access our main page, he must then press a button present on that page or, through the side menu, access the games option. He can then choose the game Yoté, which will show the different game modes available, after selecting the option for playing against a player on the same device, the game will start.

Priority: High

- **Play with a Friend on Different Devices**

Rui wants to challenge a friend in a game of Rastros, so he must first access our main page, he must then press a button present on that page or, through the side menu, access the games option. Then choose the specific game he wants to play and finally, on the game page, he must choose the option to invite a friend. A link will then be generated, which can be sent to another his friend. After accessing the link the game will begin.

Priority: High

- **Create an Account**

Rui wants to create an account on the platform to be able to play against his friends. If the users want to create an account on the platform, they must first access the login page, then click on the create account button, which will open a new page with the registration form, after filling it, they will be able to submit it, later receiving a confirmation e-mail.

Priority: High

- **Add a Friend**

Ricardo wants to add his real life friend on the platform in order to find out what his rank is. To add a friend, the user has to access his list of friends and click on the icon which represents adding a new friend. Then, the user is asked to enter their friend's username and press send. The friend who received the request has to access his own list of friends so he can accept it.

Priority: High

- **Challenge a Friend**

Ricardo wishes to challenge one of his friends directly through the app, to do that he can access his friend list from the navbar, he can then press the challenge button next to the friend he wants to challenge.

Priority: Medium

- **Participate in a Tournament**

Ricardo wants to compete in a large tournament created by his school's teacher. Users wishing to participate in public or private tournaments must access the tournament list page through the side menu. Then choose the tournament in which they wish to participate, the user is then redirected to the tournament page where he can choose to sign up.

Priority: High

- **View Game Statistics**

Ricardo wants to view his progress over the last few matches. To accomplish that, users can access a page to view their match history and statistics of previously played games, this page can be accessed through the side menu.

Priority: Medium

- **Change Account Settings**

Ricardo wants to change information regarding his account (password, e-mail, …). He can accomplish that by accessing the definitions page through the side menu. His information appears so that he can choose what he wants to change.

Priority: High

- **View Personal Rank of a Specific Game**

Ricardo wants to know his rank in a specific game, in order to find that out, a user should access the Games page through the side menu and then choose the game he's looking for which will redirect him to a page with the game's information and the user's rank.

Priority: Low

- **Create a Tournament**

Ana wants to create a tournament to give her students a way of having fun while still learning new skills. After requesting permission to create tournaments, she can access the tournaments page on the side menu, where she will have the option to create a new tournament, by filling the various settings that are required like max number of students and whether the tournament is public or private.

Priority: High

- **Invite/ Kick a user from a tournament**

After creating it, Ana wants to invite her students to participate in a tournament. She can accomplish that by choosing the tournaments option on the side menu, she can then select the tournament from the list of personal tournaments. From inside the tournament page, she will have an option to invite participants. If she invites someone by mistake, she will be able to kick the player from inside the same page.

Priority: Medium

- **Change Tournament Settings**

In case Ana wants to alter any information regarding the tournament's settings (description, max number of players, …) she can do so by choosing the tournaments option on the side menu, she can then select the tournament from the list of personal tournaments. From inside the tournament page, she can choose to edit the various settings.

Priority: Medium

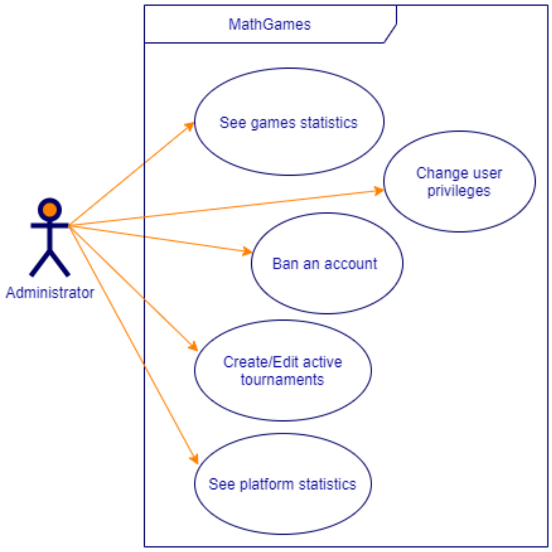

Figure 2 - Admin Use Case Diagram

- **View Statistics About the Platform**

Artur wishes to see the statistics of all the games, in order to find out what the most played games are. Administrators who want to see statistics for all games should access the statistics page through the side menu.

Priority: Medium

- **Ban a User**

Artur intends to ban an account that has been reported by multiple users of using cheats. Administrators can ban users by accessing the page with the list of all players through the side menu. Then search for the account they want to ban and click on the ban icon.

Priority: High

7 Functional Requirements

7.1 Playing Games

- Play games with a friend in the same device;
- Play games with a friend in distinct devices;
- Play games against an AI Bot;
- Allow changing AI Bot's difficulty;
- Play games against random people (with or without identical rank).

7.2 Communication

- Allow communication via chat with pre-defined sentences;
- Report users;
- Add friends;
- Remove friends;
- Invite friends to play.

7.3 Customization

- Change personal account settings;
- Change/Customize avatar.

7.4 Tournaments

- Create public or private tournaments;
- Administrate tournaments (Change configurations);
- Invite players to tournament;
- Kick players from tournament;
- Join tournaments.

7.5 Authentication

- Register a new account;

- Log-in using either a created account or through other services (Google, Facebook, …).

7.6 Admin Privileges

- Upgrade account privileges;
- Ban users;
- Watch full statistics in admins side regarding all games/users.

7.7 User Records

- Access match history;
- View ranking of a specific game;
- View account level.

8 Non-Functional Requirements

8.1 Performance

- Each page must load within 3 seconds.

8.2 Usability

- Application must be open access for different devices;
- The software should be portable. Moving from one OS to another should not create any problem.

8.3 Availability

- Application must be available all the time, except when in maintenance service, keeping maintenance time minimal.

8.4 Security and Data Integrity

- User's information should be confidential
- Passwords shall never be viewable at the point of entry or at any other time.

8.5 Regulatory

- Application must follow GDPR guidelines.

8.6 Capacity

- The app should be able to handle a high number of simultaneous users;
- Should also be able to store the data related to those users.

9 System Architecture

9.1 Architecture Diagram

Our architecture diagram aims to show the different layers of components in our project and how they are linked. Therefore, we will have a database in MySQL which will be connected to a remote nodejs server that contains an API, the business logic, and the online game engines. In the client-side, we have a desktop application implemented with electron, react and phaser. This application will contain the react website, a local nodejs server and a local database in SQLite that will be synchronized with the remote database whenever there a connection to the internet is established. Lastly, we also have a mobile application implemented in react native and phaser with only online functionalities.

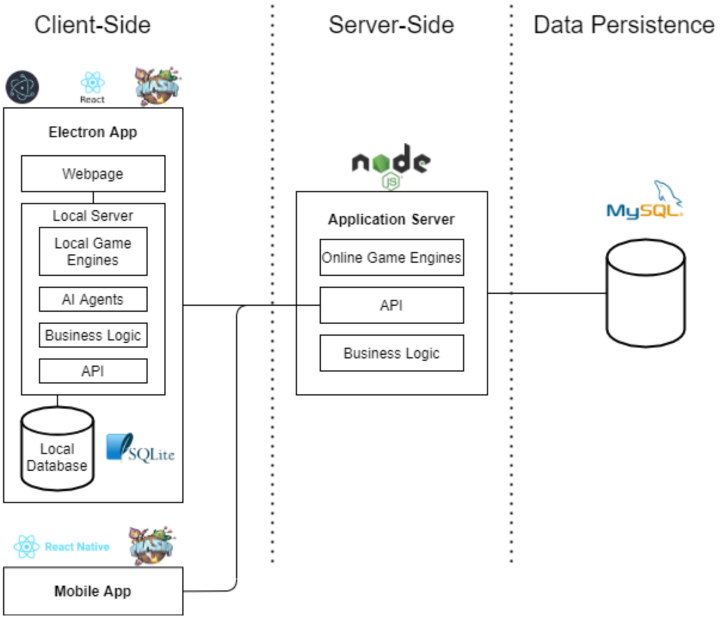

Figure 3 - Architecture Diagram

9.2 Domain Model

Our domain diagram aims to represent the data our system will contain. Therefore, it includes:

1. A user who must have an id, a username, an email, a password, an avatar, a global ranking, information if it is or is not an administrator, a rank for each game and N friends.
2. A game which must have an id, a name, a description, and a recommended age.

3. A game match which must include an id, a game, two users (player1 and player2), a winning player, the current state, a counter of moves and a game type.
4. A tournament which must have an id, a creator, a name, a game, the max number of participants, information on whether it's private or public, a password, N participating users (with information if it has already been eliminated) and a winning user.
5. A banned user who must have the reason why said user was banned.

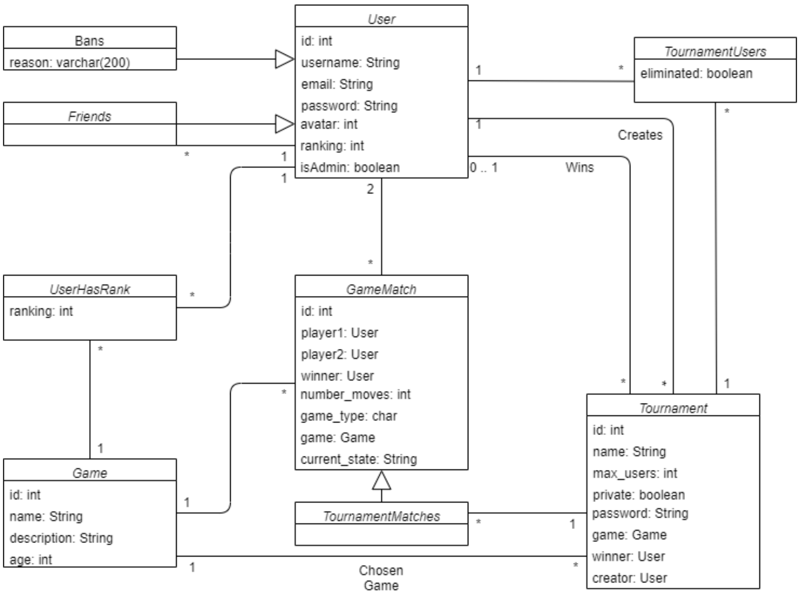

Figure 4 - Domain Model

9.3 Deployment Diagram

Our project will contain distinct components, which can all run on different machines. Our deployment diagram seeks to show how the components are connected and deployed. Essentially, we will have a MySQL relational database (deployed in a mysql8 docker container), which will be connected through nodejs's MySQL Driver to our Application Server (deployed in a node docker container). Furthermore, we will have a mobile application and a desktop application in Electron (which internally contains a web server, a client browser and a local database). These applications will be connected to the Application Server API through HTTP.

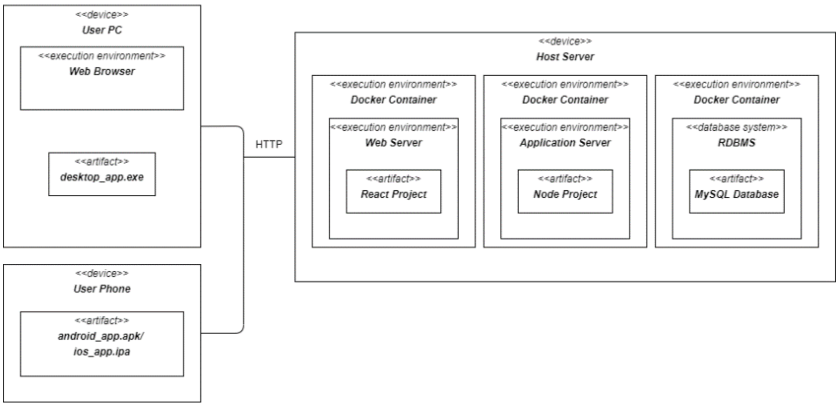

Figure 5 - Deployment Diagram