COMS 309 : VC B 4

Screen sketches assignment

Jeremy Chee, Johan Lanzrein, Alex Nicklaus, Joe Ward

Screens made by : Alex Nicklaus

General Overview of Gameplay Screens

Out project is a player versus player turret defense game where both players pit themselves against one another creating units to destroy the enemy base and turrets to defend themselves from the offending units sent by their opponent.

The game has several different displays: base, army, and enemy's base. These screens display separately and can be swapped between by the player. These different screens have different functionality assigned to them but there is common functionality shared among them as well.

Common Functionality Between Screens

Building turrets and units as well as upgrading them costs resources. The game generates resources over time for the player to spend. The player's current amount of resources is clearly displayed in the top left hand of the screen. This tab is visible on all screens.

The game operates in waves. Players will assign units to waves, these assigned units will attack the enemy when a wave start, a wave starts at the end of some fixed time.

This time is displayed in the top right hand corner of the screen. This tab is visible on all screens.

Next to the time display is a small gear icon this displays an in game menu where the player can adjust their settings or exit their current game.

There are two buttons located in the top left-of-center part of the screen. These buttons allow the player to navigate between screens these buttons change depending on what screen they are on.

1. Base

a. This screen is the first screen a player will see when starting a game

b. Here we see the main play area for this player, their base, where they can interact with the field of play to build and upgrade turrets which will protect their base from enemy units

c. At the bottom of the screen there is a Buildables Bar on it this is from where players can select the buildable they wish to construct and drag in into position. Once placed the appropriate amount of resources will be deducted from their resource pool. In the top center of this bar there is an up arrow above text which reads, "Upgrades." Tapping this arrow will bring the bar up over the screen as a semi-transparent overlay. See 2. Upgrades Overlay Menu for more on this.

2. Base Upgrades Overlay Menu

a. If a player taps on the up arrow of the Buildables Bar then this overlay will come up taking up most the screen.

b. This screen displays upgrade trees for all buildables and the associated costs with each upgrade.

c. Once paid for the upgrade is applied to all current turrets on the map and all future turrets.

d. This overlay will return to its original off screen position if the player taps the down arrow button located just above the "Upgrades" text.

3. Army Tab

a. This tab allows the player to view your army, construct units, commit troops to the next wave, and upgrade your units.

b. A player can construct units using the Units Bar located at the bottom of the screen which displays available units and their associated costs.

c. The player can commit units to the next wave by operating the up and down arrows on the menu located on the right side of the screen. When the next wave starts the units committed to the next wave will be release at the enemy's base at the players selected spawn point and will target structures according to their set priority.

4. Army Upgrades Overlay

a. If a player taps on the up arrow of the Units Bar then this overlay will come up taking up most the screen.

b. This screen displays upgrade trees for all Units and the associated costs with each upgrade.

c. Once paid for the upgrade is applied to all existing Units and all future units.

d. This overlay will return to its original off screen position if the player taps the down arrow button located just above the "Upgrades" text.

5. Enemy Base

a. Should a player navigate to this screen they will see the layout of their enemy's base.

b. They cannot however see what units their enemy has constructed or those that will be sent towards them.

c. The player can set spawn points, prioritize targets, the path their units will take, and direct digger units to construct new roads.

6. Enemy Base In Game Menu

a. Should a player tap the small gear icon located in the top right of all screens an In-Game-Menu Overlay will pop up.

b. This menu allows the player to do adjust their in game settings or exit the game.

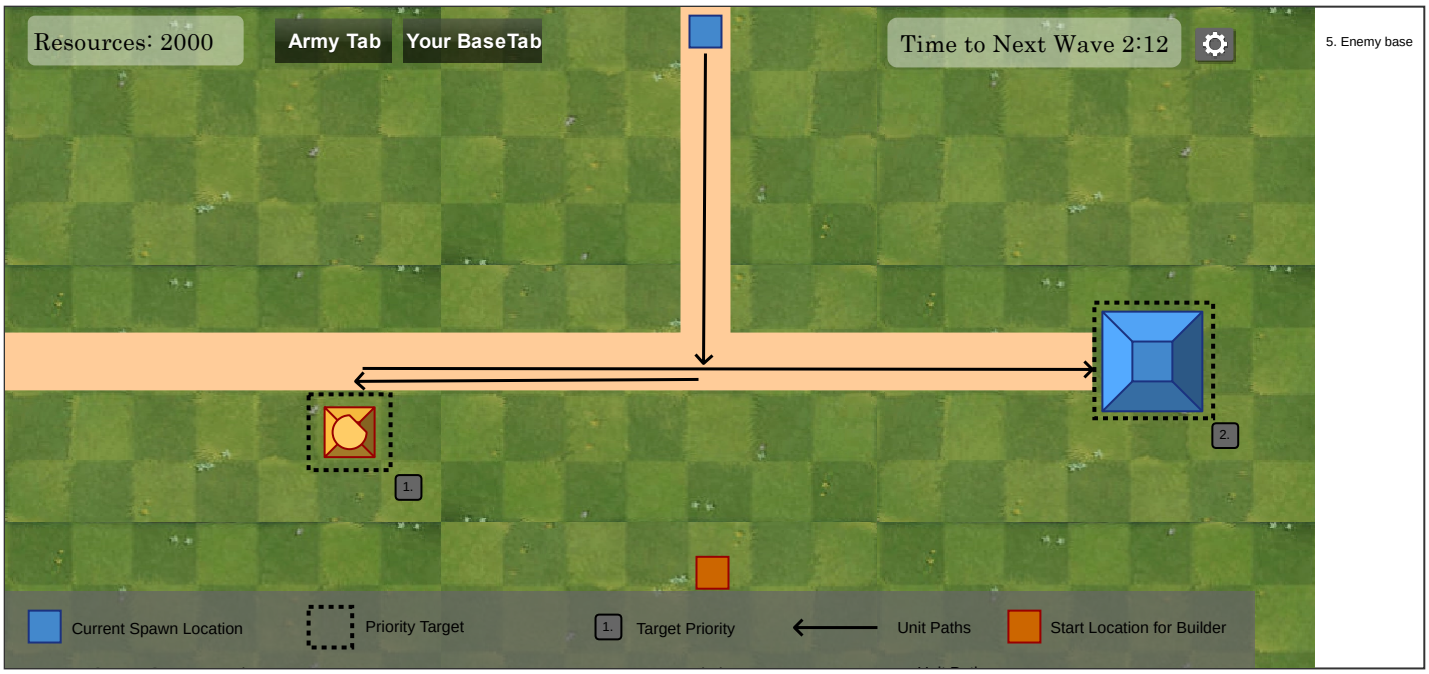
Untitled Page

Untitled Page

Untitled Page

Untitled Page

Untitled Page

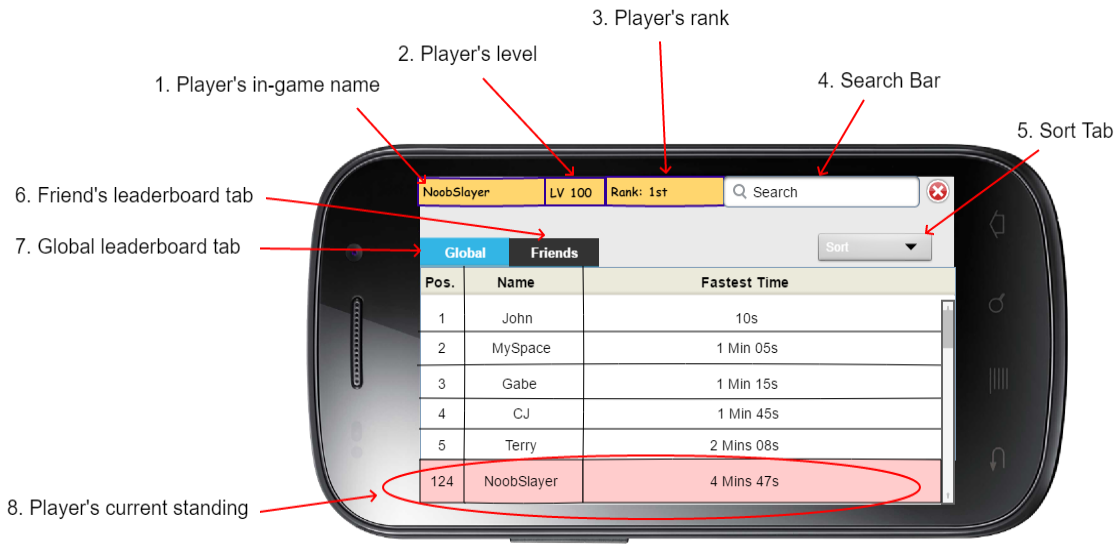


Untitled Page

6. Enemy base in game menu

Leaderboard Screen

Made by Jeremy Min Yih Chee

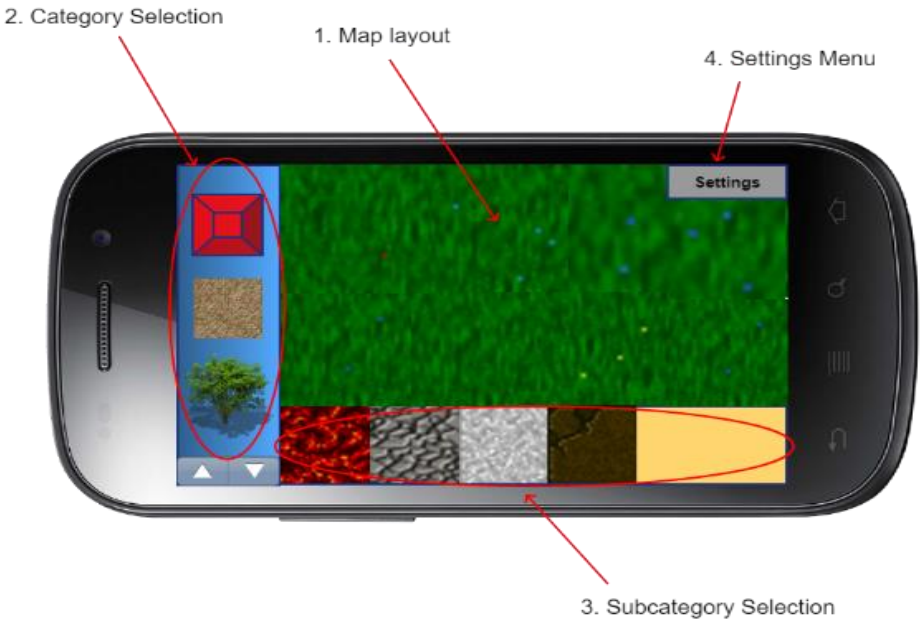


The leaderboard screen allows players to view their own and other players' position based on different conditions like fastest time to defeat an opponent and ranking. The leaderboard also ensure and maintain the competiveness among players which makes the game more appealing. Players can accessed to this screen by selecting the "Leaderboard" option in the Main Menu.

In the top part of the Leaderboard screen, the players' in-game name (1), players' level (2), and players' rank (3) will be displayed to grant flexibility for the players to view their in-game information. As for the Search Bar (4), it allows the players to search for other players' in-game information. Since, the leaderboard is sorted based on different conditions, players are able to view the leaderboard based on the conditions of their liking with the Sort Tab (5). Aside from that, the Leaderboard is divided into a friend's (6) and global (7) leaderboard tabs. In the friend's leaderboard tab (6), the players' friends are ranked among each other only whereas for the global leaderboard (7), it ranks every players in the game among themselves; both of these tabs are sorted based on the different game condition. The Leaderboard will also display the player's current standing (8). By having this particular feature (8), players are able to compare their current standings without going through the trouble of manually finding their own standings.

Map Editor

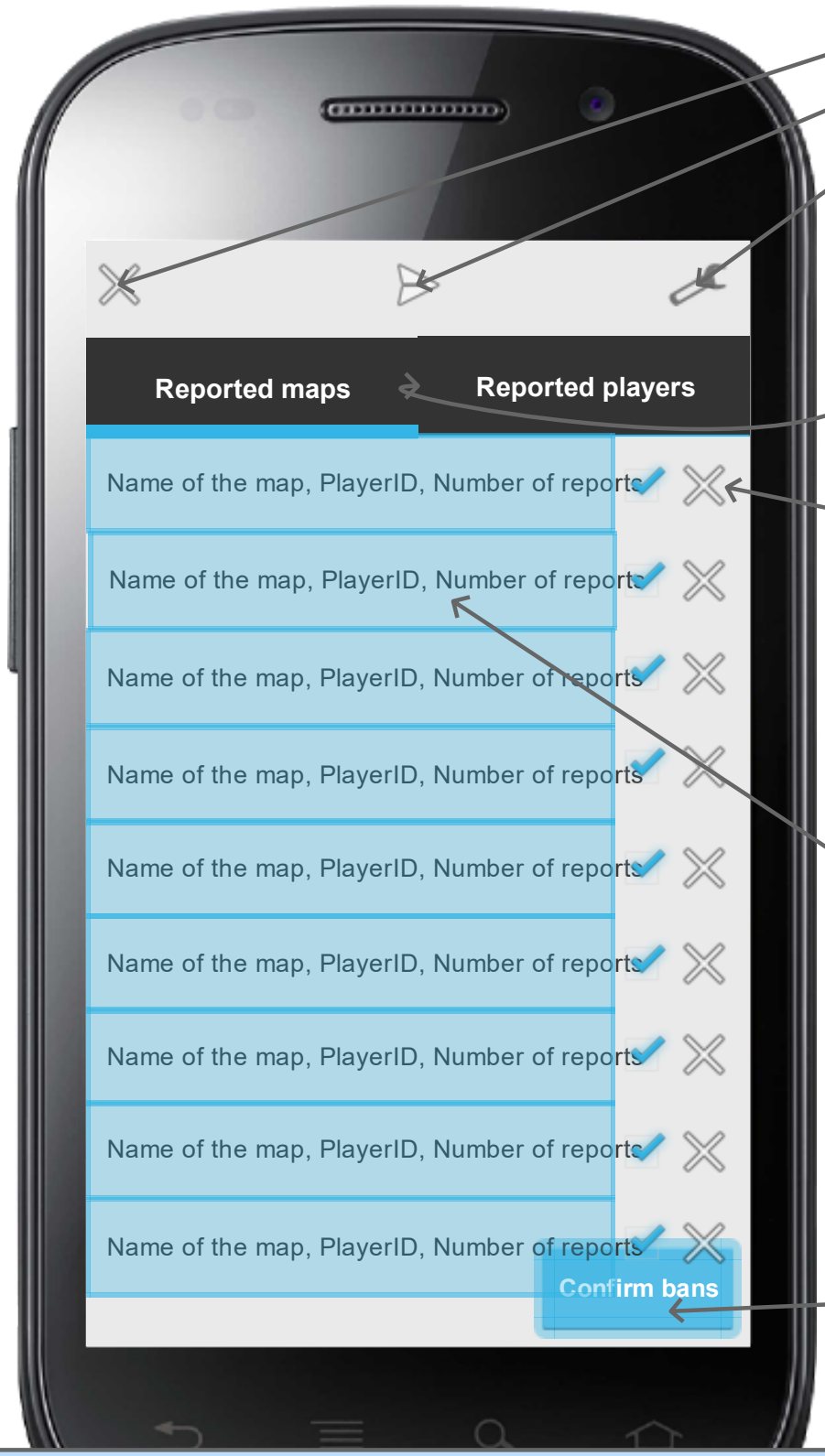
Made by Jeremy Min Yih Chee



The map editor screen allows players to create their own gameplay map with the available assets. This feature prevents the game from being stagnant as it allows players to experience different kind of gameplay. This editor will appear when the “Create Map” option in the Map Editor Menu is selected whereas the Map Editor Menu can be accessed through the Main Menu of the game. Aside from that, this editor can also be accessed through the “Edit Map” option in the Saved Map Viewer menu that can be selected in the Map Editor Menu.

In the Map Editor, players are allow to edit or customize their desired map in the Map layout (1). The Map Layout (1) is designed in a grid based manner where players are only needed to drag and drop their desired asset onto the Map layout (1). In the Category Selection (2), objects are categorized based on their functionality. This particular tab (2) allows players to navigate through the available objects more easily. Once the players selected one of the object in the Category Selection (2), the Subcategory Selection (3) will be available. In this Selection (3), a list of objects that shares the same functionality but possess different traits will be available for players to drag and drop into the Map layout (1). For example, a player can select a terrain in the Category Selection (2) and the different kinds of terrain like ice terrain or sand terrain will be displayed in the Subcategory Selection (3). If players decide to save or publish their current map design, they can select the Settings Menu (4). Other than saving and publishing, the Settings Menu (4) also allows players to quit from the Map Editor and edit their map description.

Untitled Page



3 Options :
-Go Back
-Send mail to app creator
-Settings

Admin can either moderate the reported maps or the reported players.
By switching tabs it loads.
Sorted by Number of reports

Either he rejects or approves of the report.

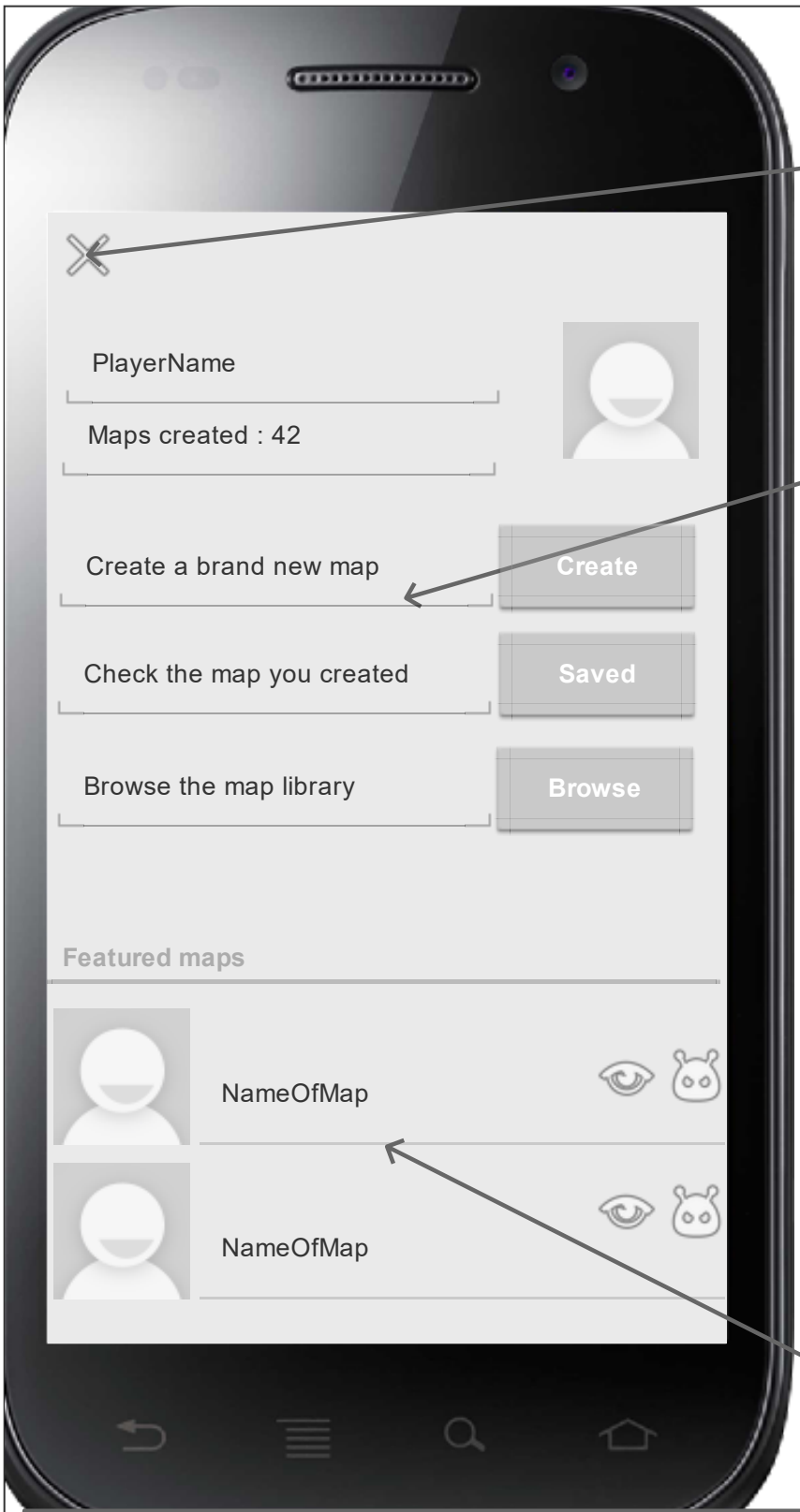
Displays relevant info about the Map/Player
For MAP:
MapID, PlayerID, Number of reports.
For PLAYER:
PlayerID, NumberOfReport, Main cause of report.

At the end. He confirms his selections. This allows to make batches of ban.

This is the admin menu. Where administrator of the game can decide whether or not to ban a player from the player/map set. This is comparable to delete player/map from the database. Therefore actions taken here will influence on the database we have.
He can also choose to write an email to the developpers of the app if an issue arises that is concerning.

Made by : Johan Lanzrein

Untitled Page



Options :
-Go back
-

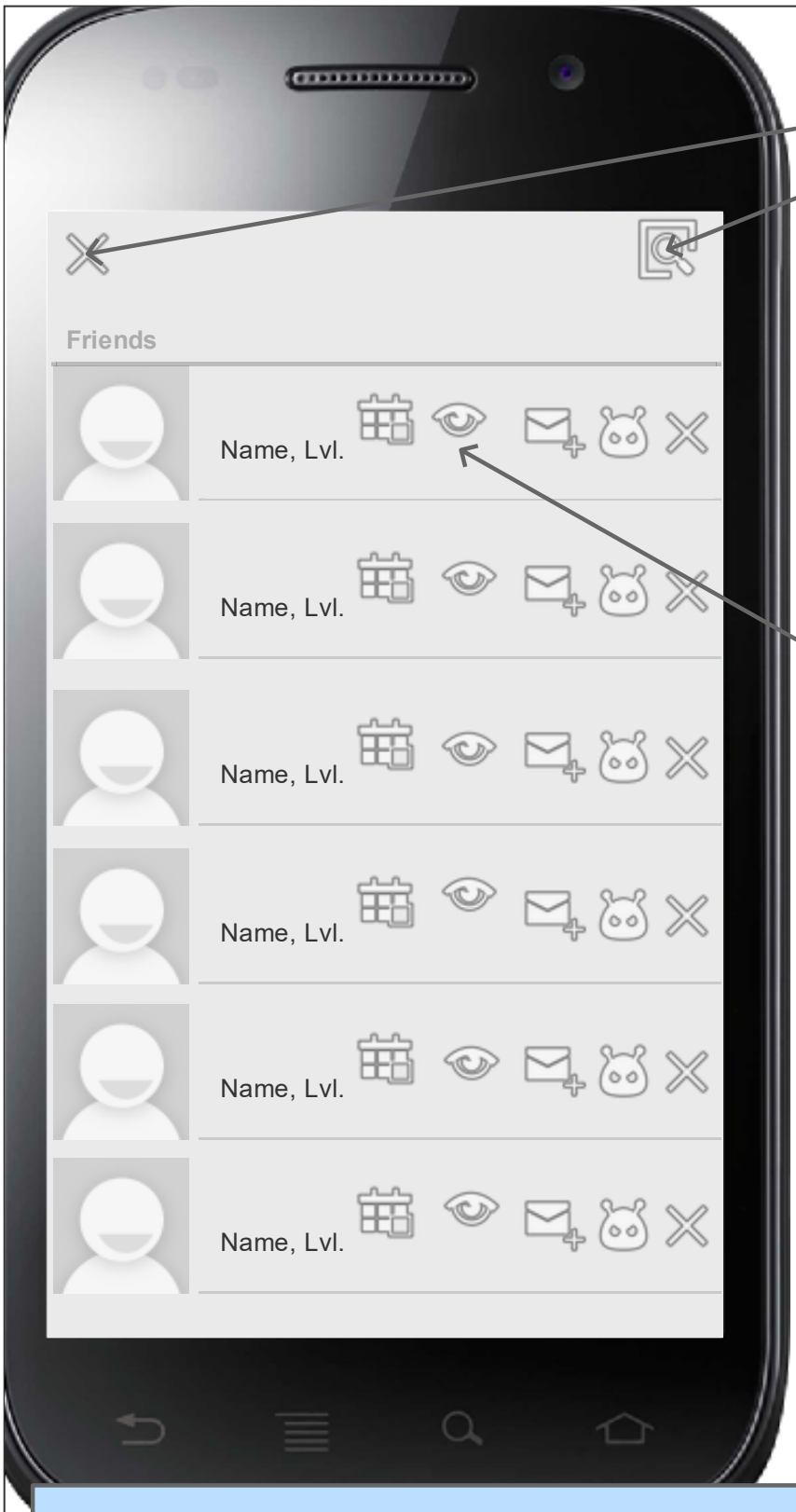
Main options. It displays the player name and the number of map he has created.
He is given 3 options :
- Create a brand new map
-Edit one of his saved maps
-Browse the list of maps.

This features 2 maps every day. So it gives exposure to lots of maps. The two maps are taken at random in the pool of exisiting maps.

Map Editing menu .
The user gets there from the main menu. Its a hub for the map editing community. They can share their maps and publish them. And then a player can retake an exisiting map save it locally and publish it again.

Made by : Johan Lanzrein

Untitled Page



Options :
-Go back
-Look for a player and add him

Display of a friend.
Shows his profile picture,
Whether or not he is online.
Gives 4 options :
- see his stats
-send msg
-duel with him
-or delete him.
NOTE : The pictograms are placeholders. We will have our own but didn't find something close enough.

Social menu.
You can here see all your friends. see who is online and from there message them, see their stats or even defy them in a duel.
Or if you happen to not be friend anymore you could delete him.
You can also add new friends by searching them from their player names.

Made by : Johan Lanzrein

This is the lobby for the 1vMany gamemode. Here the player can add or kick players from the lobby

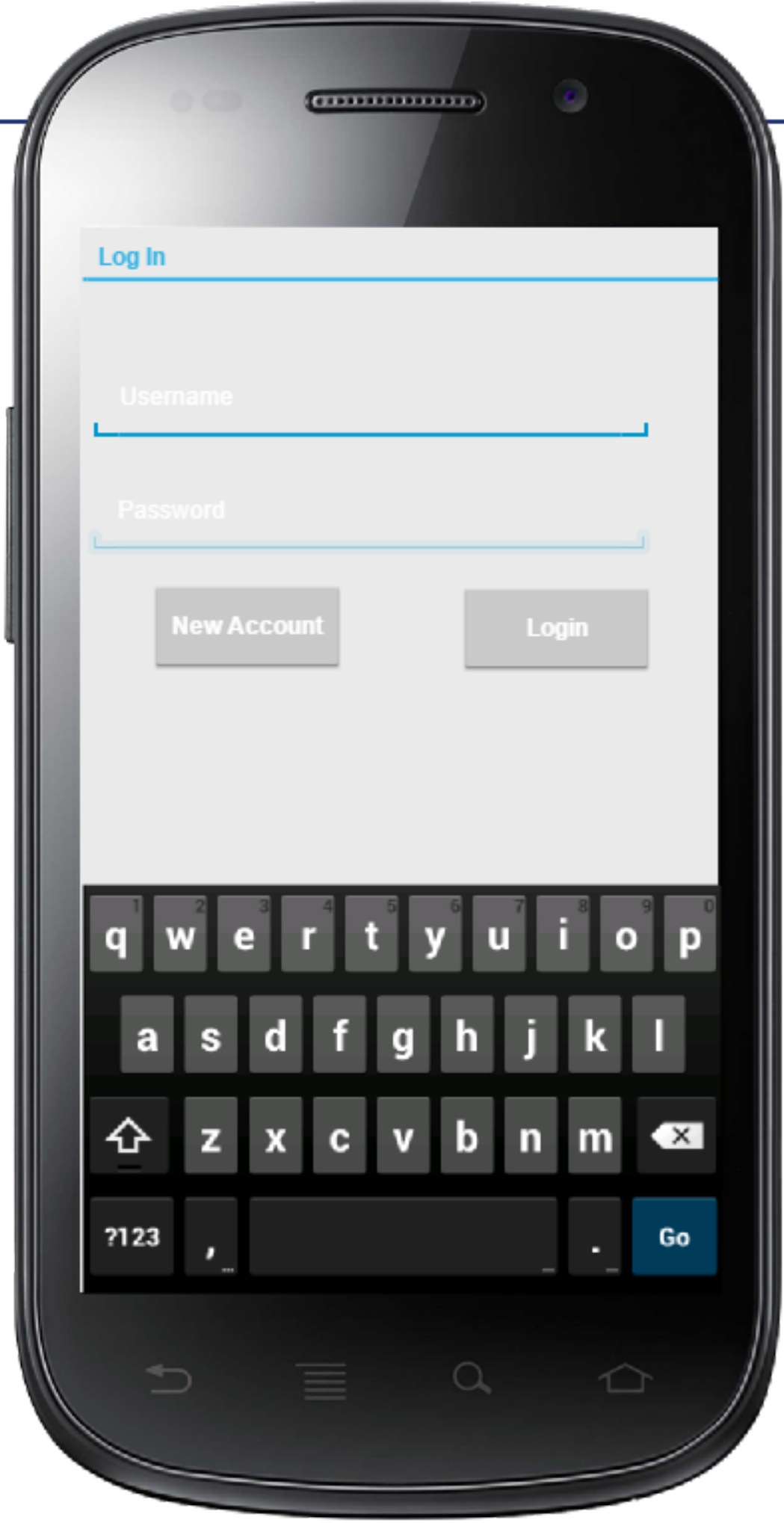
players can drag their player to any empty player slot under the attacking or defender tab

The player can select their race in the tab on the right.

The player can hit play when there is a defender and atleast one attacker, a map has been selected, and all player have selected a race.

On the bottom, the player can select the map to be played. The default maps and any saved maps will be displayed and availible for choosing.

Made by : Joe Ward

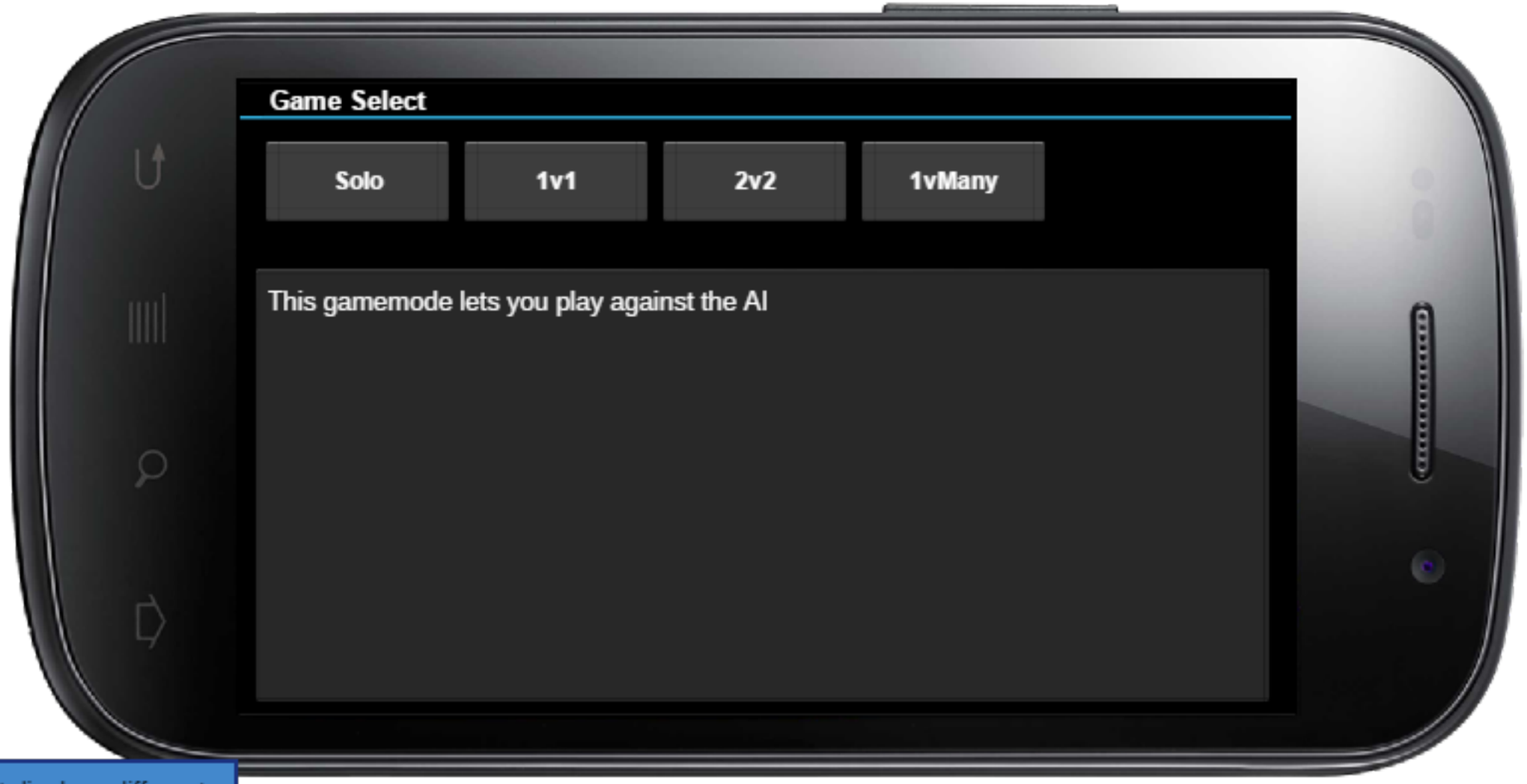


A simple login screen.
Takes you to the main menu

Made by : Joe Ward

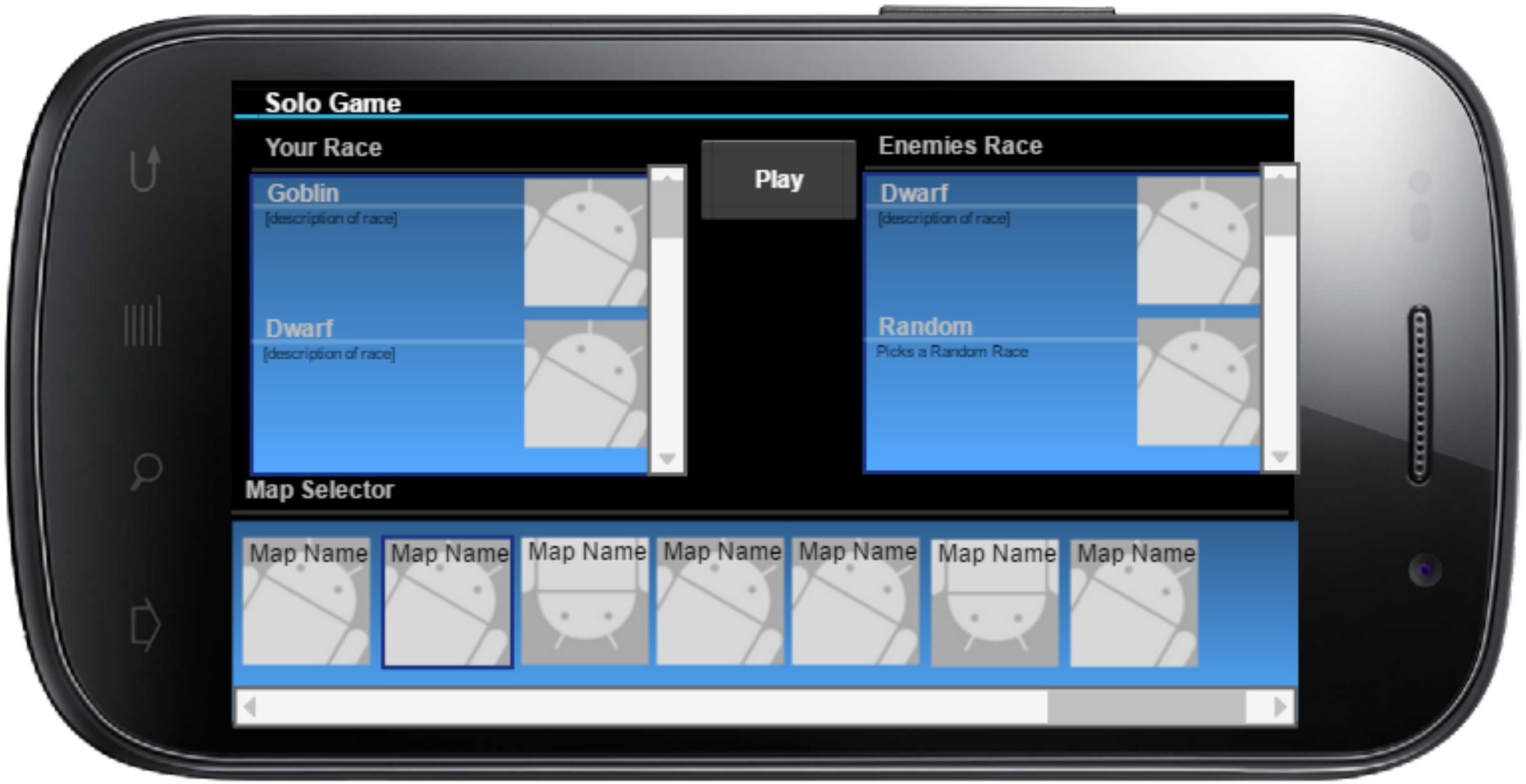
Base main menu :
From here you can access all kind of game options
You can also see what friends are online. and contact them.

Made by : Joe Ward



Screen that displays different gamemodes to play. A single click will display the gamemode description and two clicks will play the game

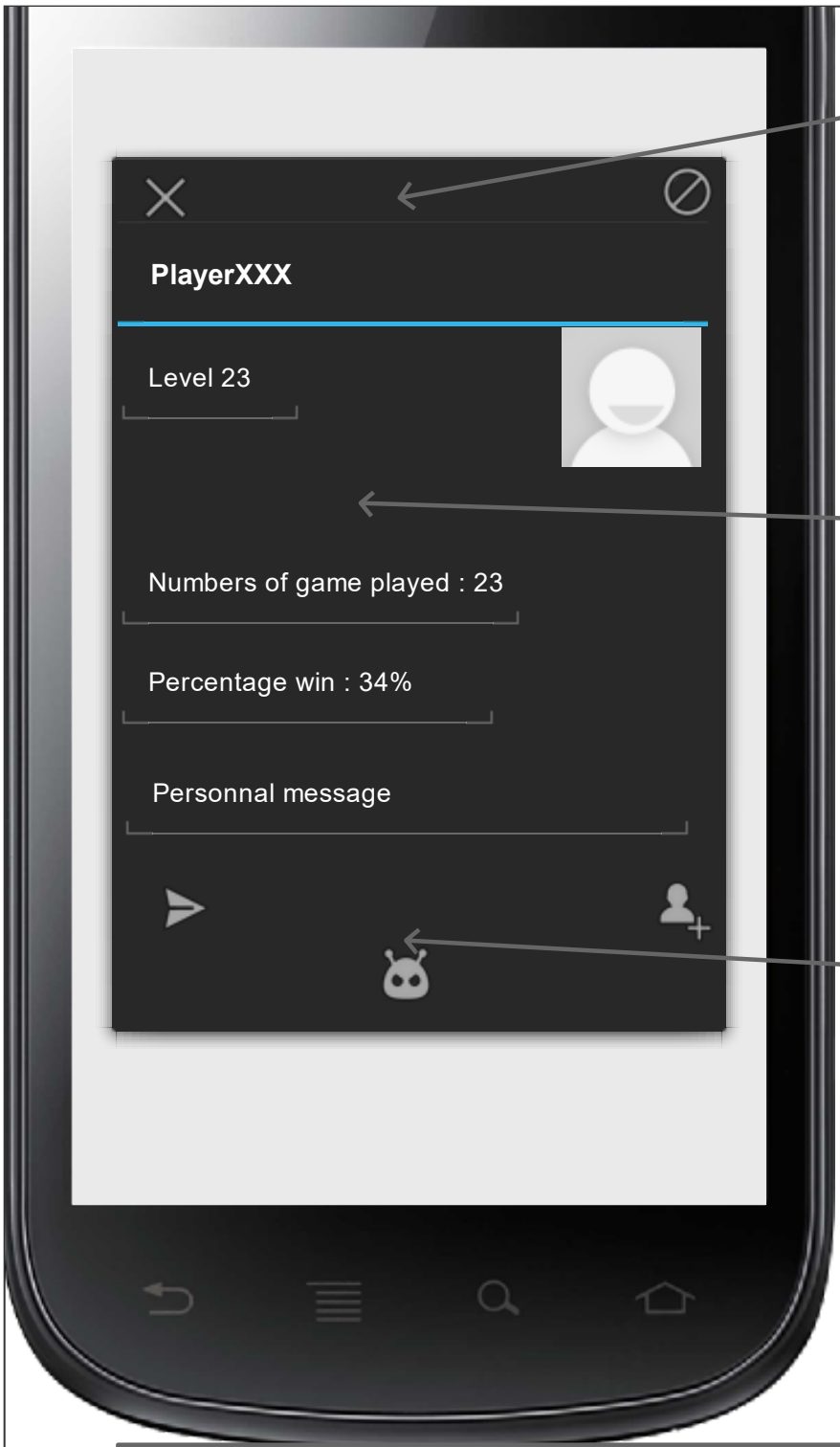
Made by : Joe Ward



This screen is for race and map selection of a solo game. Solo games are 1v1 where there is a player and an AI. The player can select the map, their race, and the AI's race.

Made by : Joe Ward

Untitled Page



Option to
- Leave the Pop-Up
.- or report the player

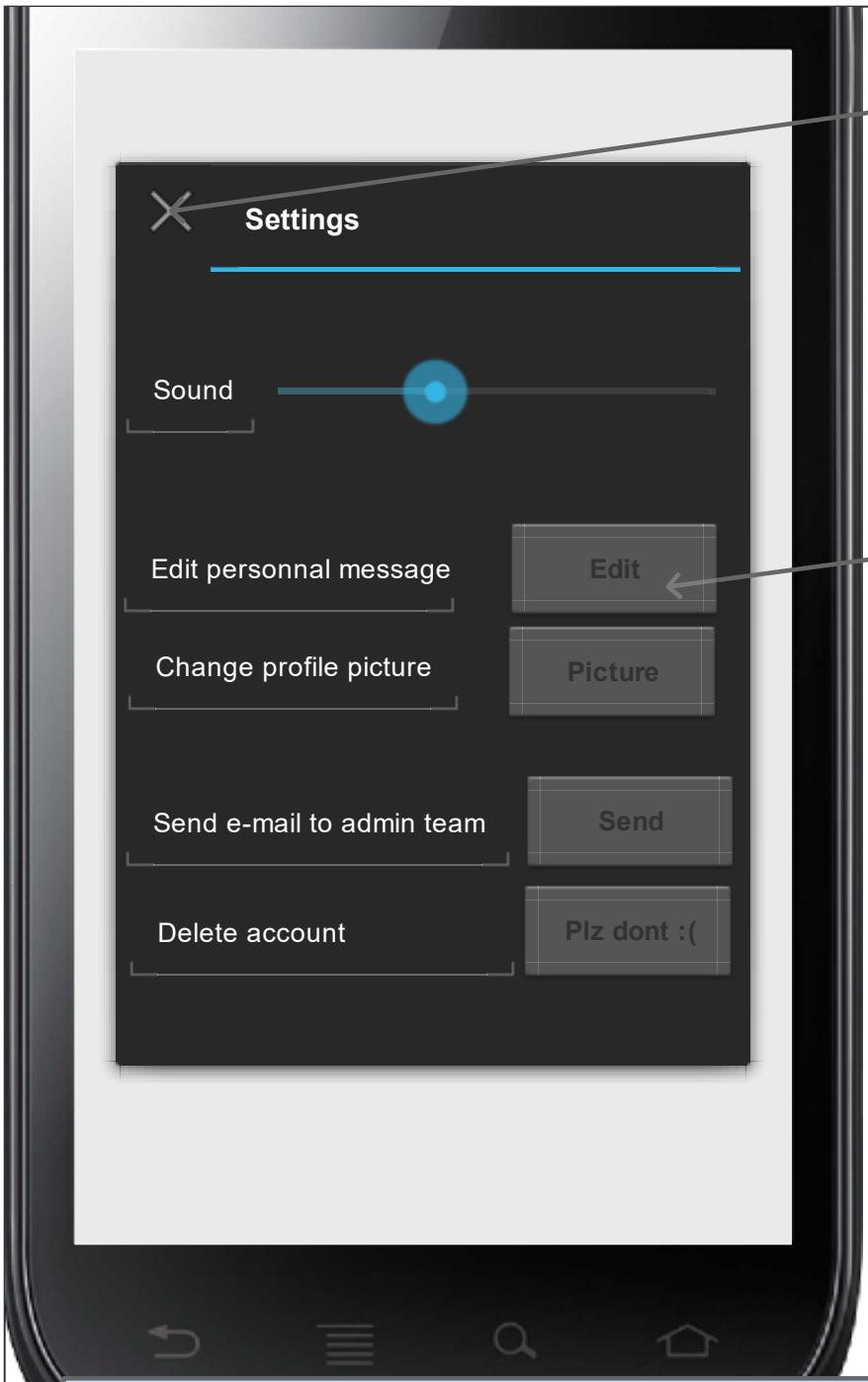
Display some stats about the player. Like lvl, status, ???, number of games playd, percentage of wins, profile picutre and profile status.

Options to :
- Message a player
- Duel him
- Add him to your friend list

Friend Pop-Up
This pop up appears when you select a player in the menu, or when you look for him from the socia menu .
This doesnt really count as a single screen but it was important to have it designed already.

Made by : Johan Lanzrein

Untitled Page

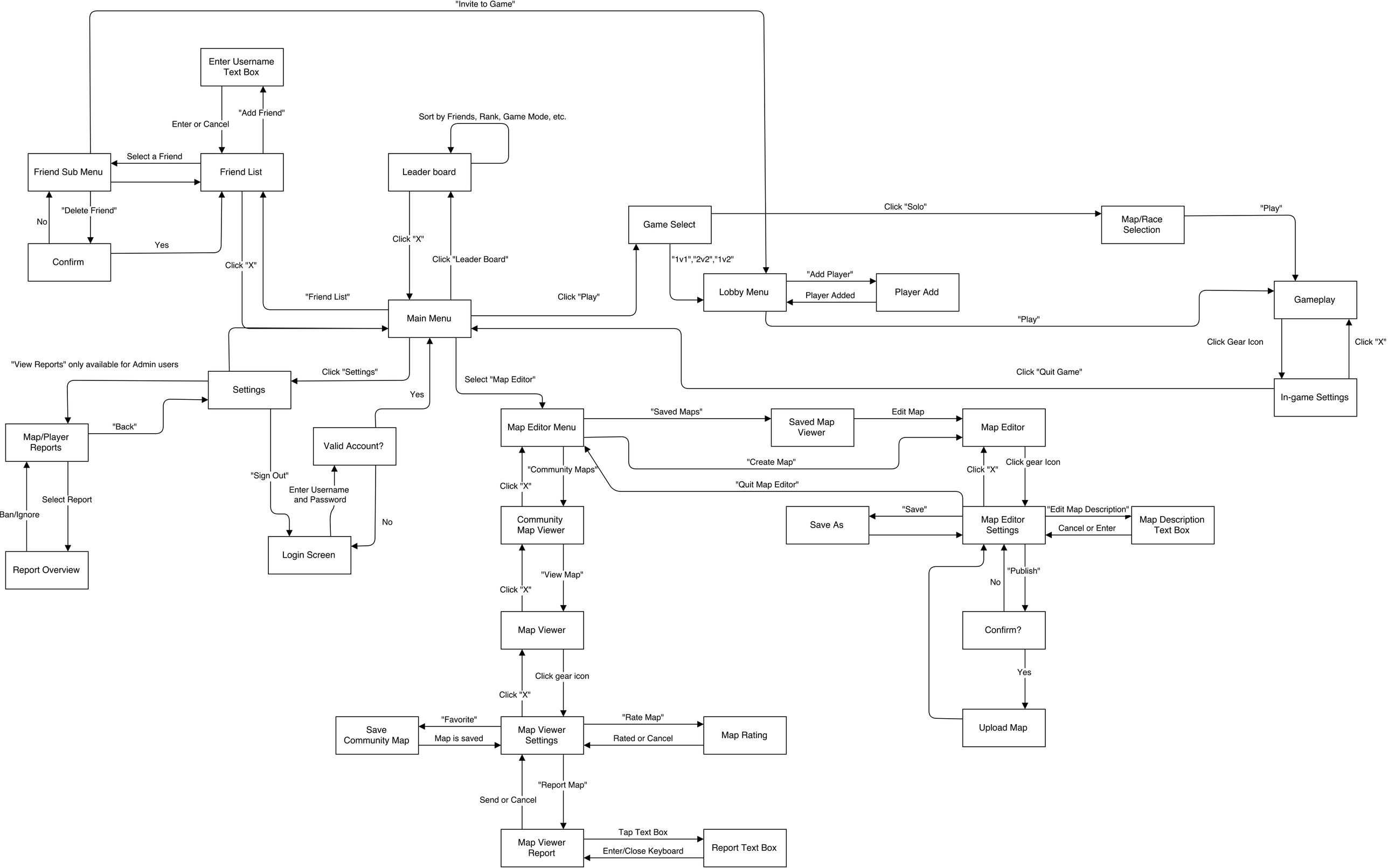


Option to
- Leave the Pop-Up
.

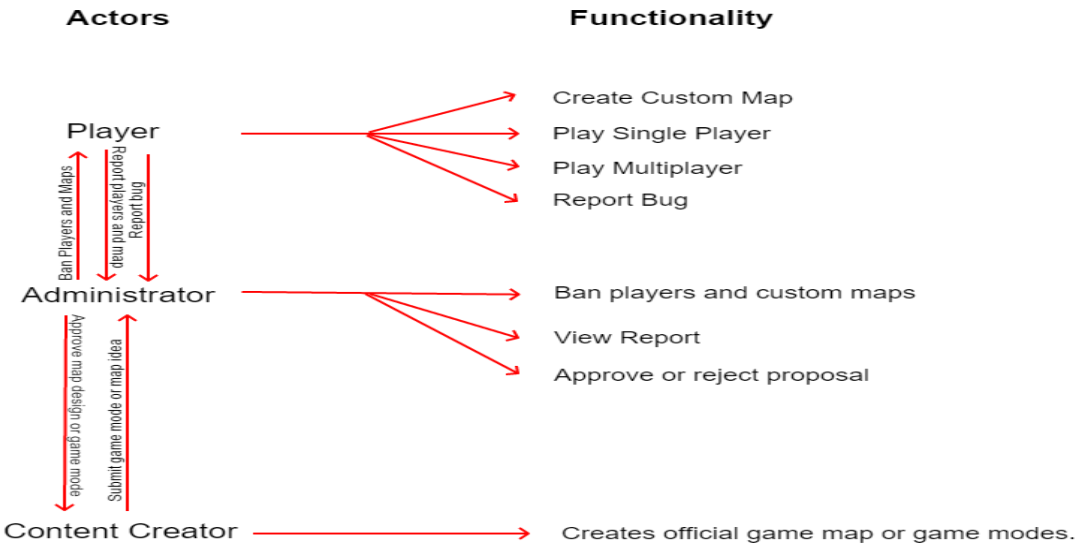
Options :
Sound, edit your personnal message, send an email to the admin team in case of bigger problems, delete all your own maps (In case you dont want people to play them.)

Settings pop-up :
This is the settings pop up from the main menu.
Nothing special about it. Just a small pop up.

Made by : Johan Lanzrein


"Invite to Game"
Enter Username Text Box
Enter or Cancel
"Add Friend"
Friend Sub Menu
Select a Friend
Friend List
Sort by Friends, Rank, Game Mode, etc.
Leader board
No
"Delete Friend"
Confirm
Yes
Click "X"
Click "X"
Click "Leader Board"
"Friend List"
Main Menu
Click "Play"
Game Select
Click "Solo"
Map/Race Selection
"Play"
"1v1","2v2","1v2"
Lobby Menu
"Add Player"
Player Added
Player Add
"Play"
Gameplay
Click Gear Icon
Click "X"
"View Reports" only available for Admin users
Settings
Click "Settings"
Yes
Select "Map Editor"
Click "Quit Game"
In-game Settings
Map/Player Reports
"Back"
Valid Account?
"Sign Out"
Enter Username and Password
No
Select Report
Ban/Ignore
Report Overview
Login Screen
Map Editor Menu
"Saved Maps"
Saved Map Viewer
Edit Map
Map Editor
"Create Map"
"Community Maps"
Click "X"
"Quit Map Editor"
Click "X"
Click gear Icon
Community Map Viewer
Save As
"Save"
Map Editor Settings
"Edit Map Description"
Cancel or Enter
Map Description Text Box
"View Map"
Click "X"
No
"Publish"
Map Viewer
Confirm?
Click gear icon
Click "X"
Yes
Save Community Map
"Favorite"
Map is saved
Map Viewer Settings
"Rate Map"
Rated or Cancel
Map Rating
Upload Map
"Report Map"
Send or Cancel
Map Viewer Report
Tap Text Box
Enter/Close Keyboard
Report Text Box

Functionality based on an online game



Functionality Description

Player

- **Create Custom Map:** Users interact with the map editing feature of the game to create a gameplay or map design of their liking.
- **Play Single Player:** Users interact with the single player mode in the main menu to play the game's story mode or to play against an enemy AI.
- **Play Multiplayer:** Users selected the multiplayer option in the main menu to play against one another or to play together in Co-op mode against enemy AI or other users.
- **Report bugs:** Players will report encountered in-game bugs to the administrator by interacting with the report bug option in the main menu.

Administrator

- **Ban players and custom maps:** The administrator will ban players or custom maps that are deemed inappropriate in the game by interacting with the admin menu that they can only accessed in the game.
- **View Report:** Administrator interact with the admin menu to view a report that users sent. The content of the report are the profile of the players' being reported, the custom map that is deemed inappropriate. The report also contained in-game bugs reported by the game master.
- **Approve or reject proposal:** Approve or reject proposal made by the game creator in the Admin Menu.

Game Creator

- **Creates official game map or game modes:** The game creator creates official game map or game modes for the game by interacting with the map editor that can be accessed from the main menu. Only game creator has accessed to additional resources in the map editor.

Non-Functional Requirements

We have decided on three non-functional requirements. They will be described in this short document by order of priority. Moreover, this document will explain how we will try to approach our code to get the requirements fulfilled.

1)Performance

The reason why we want this requirement to be on top is that in any game you need to have an optimum responsiveness. If the game lags (meaning the actions are delayed because of problems on the server or the client), the user experience decreases a lot.

To reach this requirement, we will try to have the code be optimal in the sense of having methods not running into loops too big. Or also having algorithm run in a maximum of O(nlog(n)). (This idea will also be applied to the requirement (2).

2)Scalability

We want our system to be scalable. Meaning that the server should be able to handle a big amount of user without having to invest in more servers. After looking up online the different types of scalability[1], we will mainly focus on functional scalability, and administrative scalability. Therefore, we will strive to have methods and classes able to handle new methods without much struggle for whoever implements the new functionality.

Other actions taken to insure a good scalability will be to have our algorithm run into a maximum of O(nlog(n)) if possible. And, to minimize the exchange of data between client and server so the server can handle more request at the same time.

3)Maintainability

Finally, we want our software to be maintainable. This means that if a bug arises, or an exploit was found we want whoever oversees the system. Even if it is not a member of the original development team to be able to easily navigate the code and find his way around it. Another scenario where maintainability is desirable is if someone (not necessarily from the original team) wants to modify the app to implement new features or change it. That person should be able to find his way around the code without having to spend an absurd amount of time to figure out what a method is doing. However, note that while understanding what a method does, does not mean understanding how it does it, meaning that the documentation will exists for the method, but it will not always be precisely explained how each algorithm is implemented. Although if a method is long, complicated or key to understanding the code, we will provide extra information on how the code works.

To insure this requirement, we will make sure that all classes and packages are clearly separated. That the documentation exists and is kept up to date. We will also provide a clear class diagram that explains how each class relate to each other.

[1] At : https://en.wikipedia.org/wiki/Scalability

List tables and fields

We propose to use two main databases. The first one will handle the player base. And the second one will take care of the maps since we want to give the option to edit their own map and publish them

1) Player Database

In this database, we will control all the players and their personal stats. The fields will be the following:

- PlayerID : an integer it will be the primary key of the database. This allows for different user to have the same user name
- Username : The name of the player. A string of up to 10 characters. Not null
- Experience : a float. We will use a float to allow for bigger numbers. It will be the main reason to help compute the level
- Level : An integer will determine the player level.
- Number of wins : an integer
- Number of game played : an integer
- Number of reports : an integer. This item will only be visible by admins to handle.
- Password : We will actually store a hash code of the password to insure security of it. When the user logs in we hash his password with the same hash function and try to see if it matches. We can add a salt to minimize hash collision / increase risk of attacks.

We hesitated to include the list of friend in the database. But after looking online it looks like it is not a good idea to store a list in a database.

2) Map database :

This database will store all the things related to the maps. This will allow player to fetch a particular map from the database in order to play it. It will hold the following fields :

- MapID : the ID of the map (primary key)
- MapName : The name of the map.
- CreatorID : It will reference PlayerID. (Primary key with MapID) The reasoning behind this, is that this way a Map can be taken and re-edited by a different player and he can release the map with the same ID but with his playerID on it. It allows also to find all the players who contributed on a map.
- Map: List[char] : it will be a compressed version of the map. It will also include the description of the byte. Since our map will be composed of a square of 15x15 and a description will hold up to 225 characters. This will allow a maximum of 225 bytes. Not even a kilobyte which is quite light for a map, and that is if we don't compress it. (Will be encoded with RLE).
- Description : a char string of up to 255 characters. Describes the maps briefly.

Files

The main files outside of the source codes, will be all the files related to the graphical interface. We will need sprites to represent all monsters and tower. And also resources sprites to represent the map. All those resources will be included in the app that way no graphical components will travel across the network while playing.

Source for the latex template : https://www.latextemplates.com/template/stylish-title-page