

Team: Phil Leonowens, Jake Mitchell, Ryan Craig

Title: Hacker Defense

Project Summary: Hacker Defense is a tower defense game. There is a playable area that will have a track laid out on it for the enemies to go through. The User is allowed to use the rest of the grid to place towers down that will try and stop the enemies. We will also have a settings screen where the User can adjust settings. Lastly there will be a database where for local and global scores.

Project Requirements

Business Requirements

ID	Requirements	Success Metric	Priority
0	Create a desirable game that people will play and that is engaging	Successful feedback from user testing	Medium
1	The game is an appropriate difficulty	Our user testers are able to beat the game, but not all of them win	Medium

User Requirements

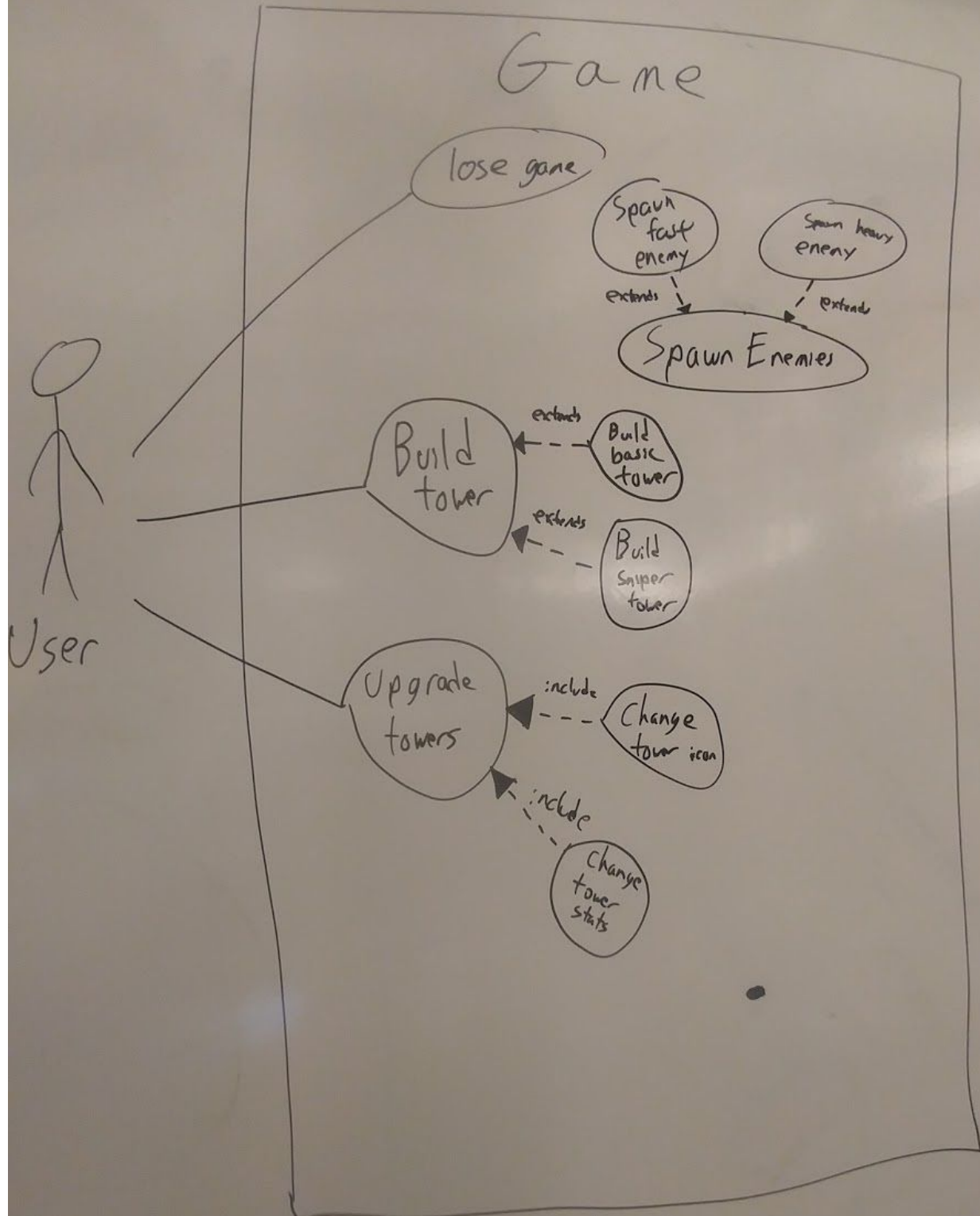
ID	Requirements	Success Metric	Priority
0	A variety of towers needs to be able to be constructed by a player	We have at least four different varieties of towers	Critical
1	A variety of enemies need to spawn with a number of different characteristics abilities and properties	Five different kinds of enemies, with varying speeds and skills exist	Critical
2	The game can be lost	There are losing conditions, whether it's a base HP, or a certain number of creeps reaching base	Critical
3	Towers can be upgraded	¾ of the towers have at least one upgrade path	low
4	Users must be able to see the projectiles in flight	Projectiles fly smoothly towards their targets and	High

		perform an animation on hit	
5	Creeps can be damaged and killed by projectiles	Projectiles can interact with creeps and deal damage to them	High

Non Functional Requirements

ID	Requirements	Success Metric	Priority
0	Deliverable will be small enough to distribute	Can be hosted online successfully in a manner that the average user can download within 120 seconds	
1	Resolutions up to 720p are supported	Adjustable resolutions up to 720p work and render well and legibly	
2	Runs at a steady 30fps	Our computers are able to run the game at 30 frames per second without many lag spikes	

Use Cases

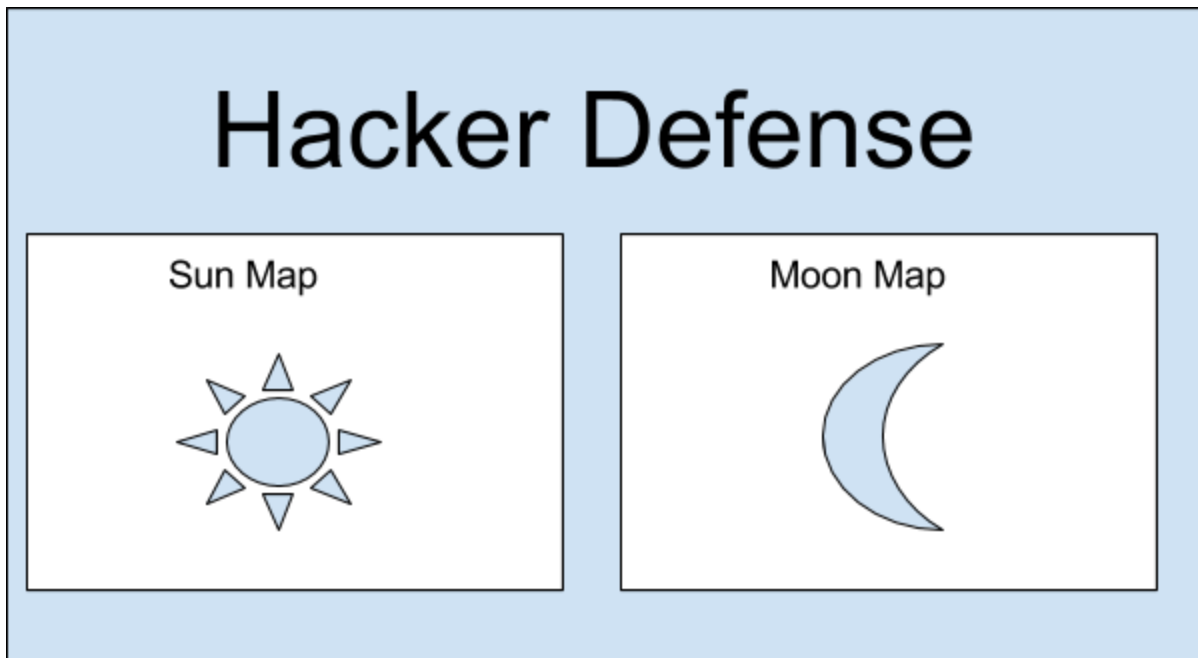


intel

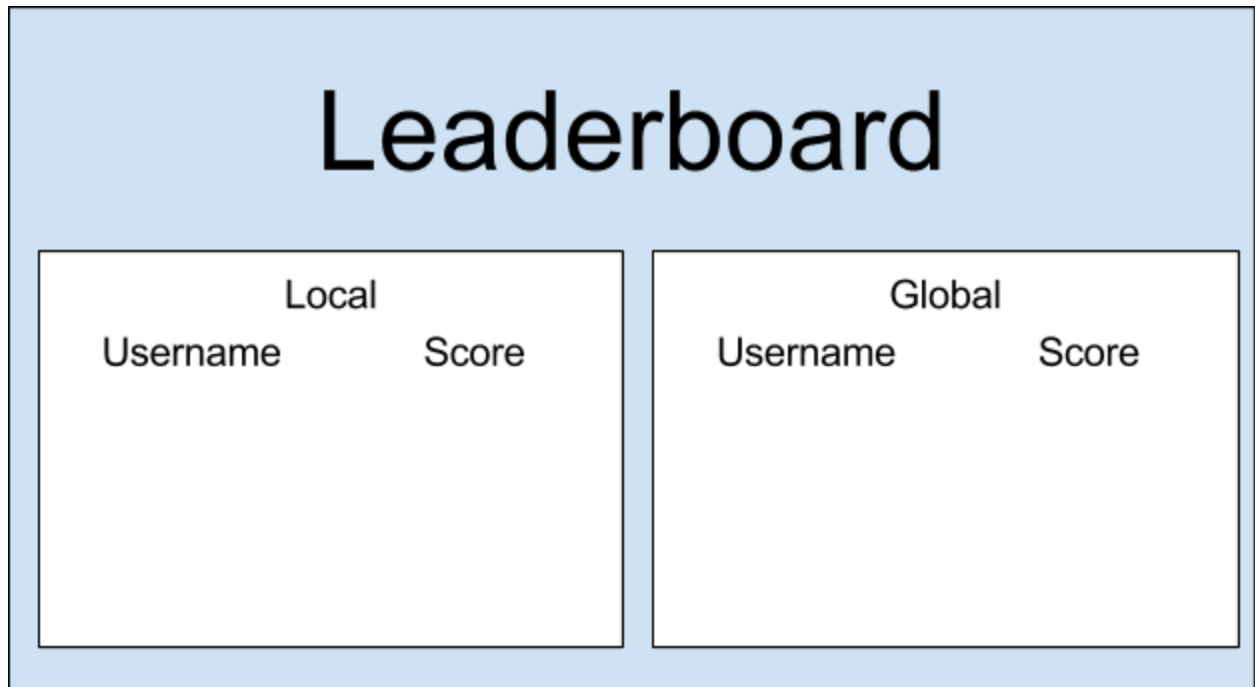
Document how the system will support each task via a use case. For the Project Part 2 GROUP, provide the use case diagram (the use case documents will be in your individual submission). • Use Case Overview: Create a single overview use case diagram depicting the main use cases the actors interact with. Note that these should map back to your user requirements. • Sub-diagrams: Create any necessary sub-diagrams to show more in depth the details of the use cases including any <<includes>> and/or <<extends>>. Be sure to label which use case you are describing in more detail. • UI Mockups: Create screen mockups for the user interface of

UI Mockups

This first image is of map selection screen. The different map options will be displayed and greyed out if the user has no unlocked them yet. The blue background will have some sort of picture or animation related to the game.

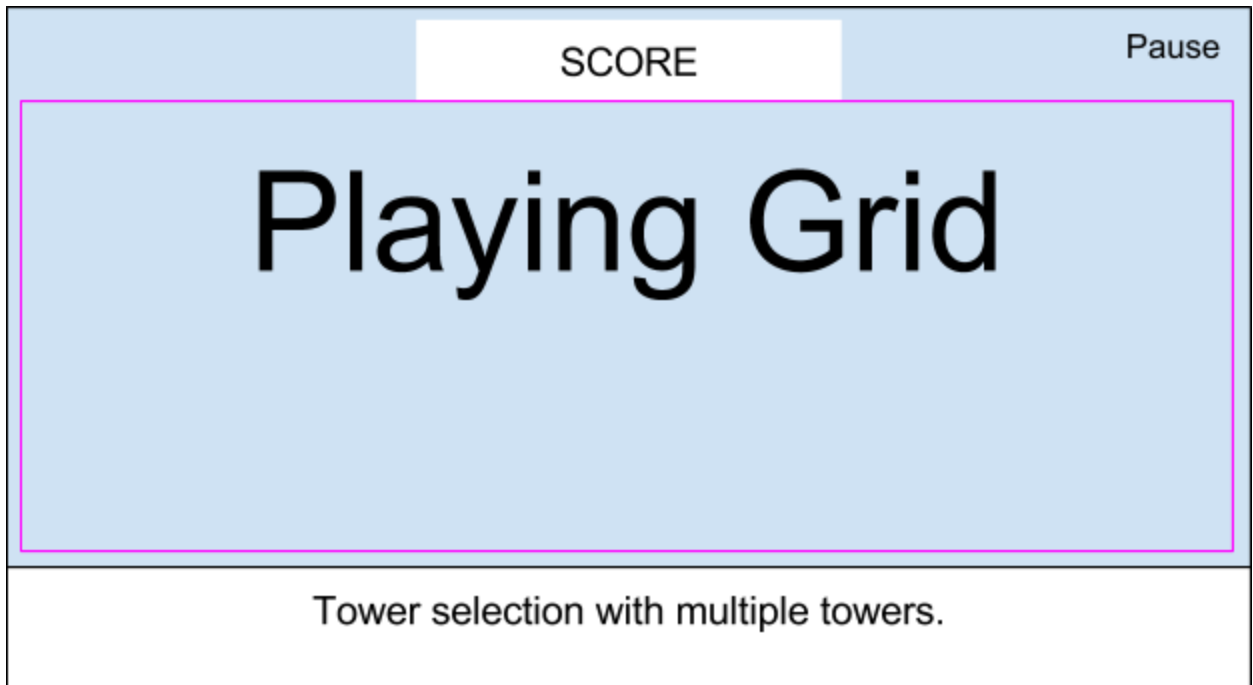
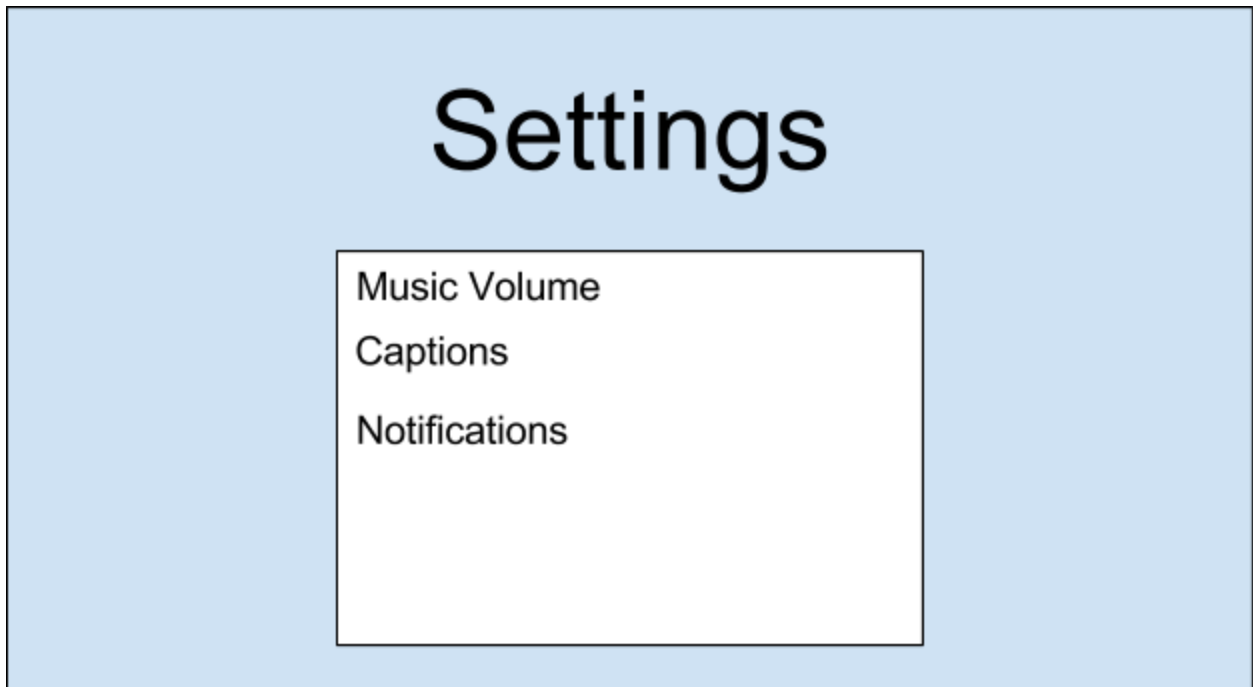


The second image is the main menu. It will have the ability to start a game, check leader boards, and change the settings. The blue background like above will have some sort of image or background related to the game.



With the Leaderboard we will have two different displays. A local score which will either be region based score or possibly highscores set on the local machine. The Global section will populate with the top players of the world.

The settings page will have different options for changeable elements. Such as music volume, captions, notifications, etc.



This last image is the actual game screen. It will consist of the score at the top and a pause option. Below that is the largest area, it is a large grid that will have different tracks for the enemies. Where there is no track the user will be able to place some of the towers.

DB System

For the database we are going to use MySQL.

As of right now we are thinking of having 4 tables (Users, Passwords, and Local Scores and Global scores). The tables and what's planned to be stored is listed below.

- Users
 - Email
 - Username
 - User Account Number
- Passwords
 - User Account Number
 - Encrypted Password
- Local Score (Possibly just a local file)
 - Username
 - Score
 - Time it was set
- Global Scores
 - Username
 - Score
 - Time it was set

Class Diagram

