



**CS319 Object Oriented Software Engineering
Final Report**

Lords in Halls

Group Members

1. Şamil İraz
2. Ateş Bilgin
3. Enes Emre Erdem
4. Gülnihal Muslu
5. Can Ozan Kaş

Supervisor: Eray Tüzün

Teaching Assistant: Gülden Olgun

Table of Contents

1. Implementation Process
2. Changes in Design
 - 2.1) Subsystem, Models, View, Controller
 - 2.2) Wall Placement
 - 2.3) Hint
 - 2.4) Creator Mode
3. Work Allocation
4. Users Guide
 - 4.1) System Requirements
 - 4.2) How to Use

1. Implementation Process

After we delivered the first design report, we started the implementation of the project. We divided the jobs among the developers. We use MVC design pattern in the design process of our game, therefore, it was easy to split tasks among participants. One group is responsible for the class Models of the game, another group was responsible for the design of the server and database, and another group was responsible for the design of the views for User Interfaces. After we divide each role, the developers that are responsible for each role worked parallel, which makes the implementation process faster. We made that decision because the different project parts are easy to merge.

The tasks were divided among the participants in the following way: Enes installed and set up the database and designed the required interface for the database usage. Şamil developed the model classes required for the game logic. After the installment of database and game models, Enes and Şamil merged the database and game models in the Controller part and designed the web server. On the other hand, in the parallel process, Ateş Gülnihal and Ozan developed the User Interfaces of the game and level maps as a view part. Before the first iteration, the different parts were combined together, and we completed the design of the game with some of the functional requirements. After the first iteration, we implemented the additional features and commit them to the project.

On the implementation process of the different parts, we kept in contact with each other, via using discord, to inform each other about the details of the implementation. We use IntelliJ development environment in the implementation process, since it provides good

development environment and easy to commit changes into github. Lastly, we use Java spring framework in order to design and install the server side.

In conclusion, we completed most of the functional requirements, promised on the analysis report. We completed the four required game mode. Moreover, we added some more functionalities to the game, such as get hint, and Creator mod. On the other hand, there left some of the functional requirements didn't completed, such as map creator algorithm, or drag-and- drop wall controls.

2. Changes in Design

2.1 Subsystems, Models, Views, Controller

In our project, we did not make any changes in the subsystem decompositions or in the design patterns, as it was MVC. Some of the page views were not implemented in the first iteration, we add them in the second iteration.

2.2 Wall Placement

In the first iteration, we were thinking to implement the wall controls of user with mouse drag and drop methods. However, we decided to implement wall placements with entering coordinates on the map, because it is easier and faster to implement.

2.3 Hint

After the first iteration, we decided to add new features to the game to make it more fun and interesting to play. Since the game gets more and more harder, we add hint button to the game to help players in harder levels. Hint button shows players the correct place of one wall.

2.4 Creator Mode

We keep thinking about adding new features to the game after first iteration, and come up with a new idea. In create level mode, users can create their own levels. Created maps are saved in the database, and can be added to the one of the game modes later.

3. Work Allocation

- Ateş Bilgin

Reports, Presentation, Front end: HTML pages, design, Javascript.

- Can Ozan Kaş

Reports, Presentation, Front end: HTML pages, design, Javascript.

- Enes Emre Erdem

Reports, Front end: Javascript, Back end: spring boot.

- Gülnihal Muslu

Reports, Presentation, Front end: HTML pages, design, Javascript.

- Şamil İraz

Reports, Back end: file system, spring boot.

4. User's Guide

4.1 System requirements

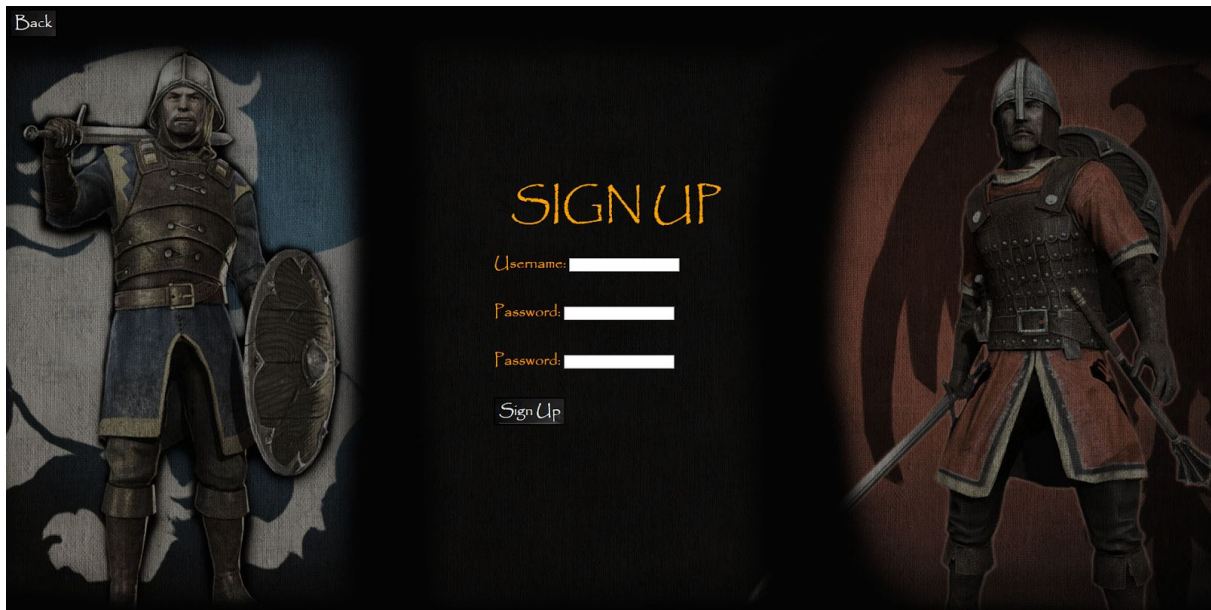
System requires internet connection in order to play the game. The game works in web browser, so user need to enter the game using a web browser. It does not require installation process, users can play it directly from the browser.

To run the application, you first need to download the project from github. Then open in with IntelliJ IDE. Build the project in IntelliJ, then go to the URL: "localhost:8080". This URL will redirect you to the initial page of the application. From now on, you can use links to navigate through the application.

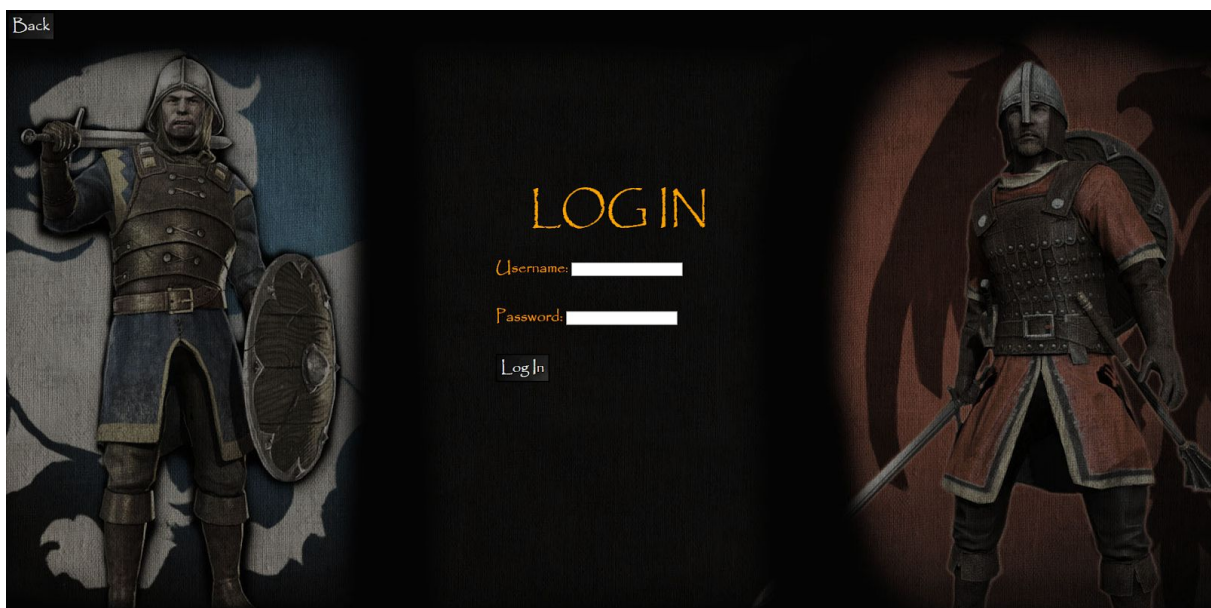
4.2 How to use



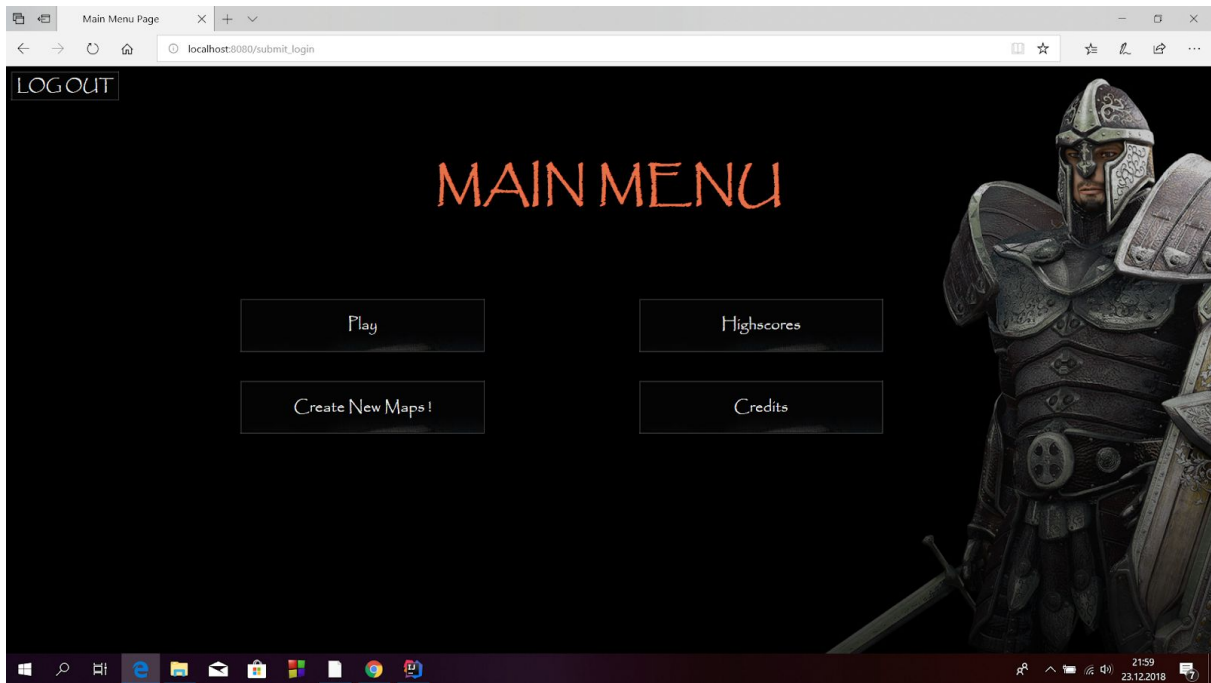
When user enters the game, an initial page will confront the user. User can open an account with sign up, or enter the game with existing account with login.



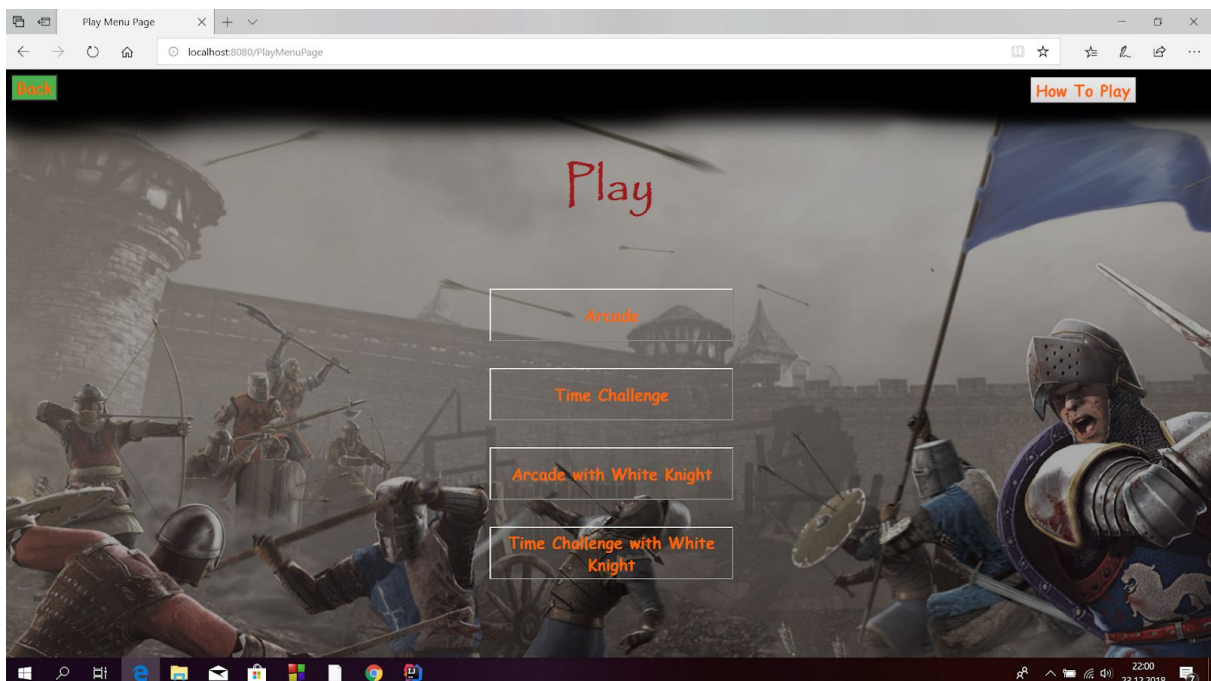
User can create new account in sign up page, with entering desired username, password and entering the password second time in order to verify.



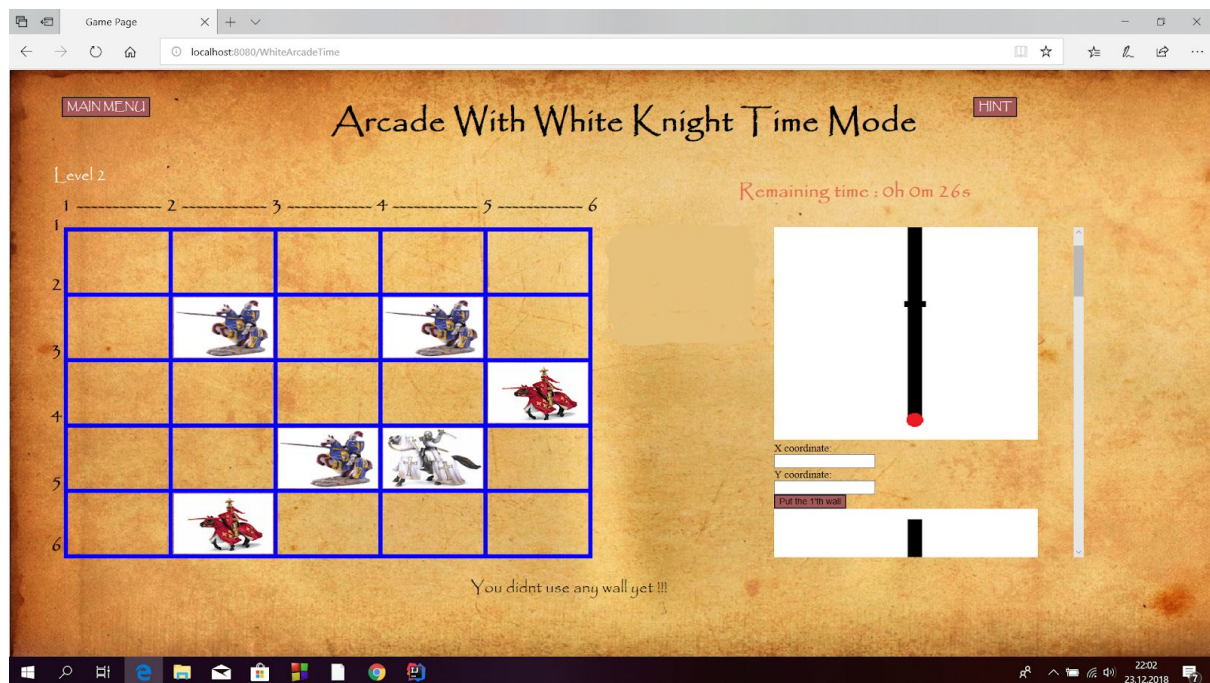
In login page, user enters the username and password to enter the game.



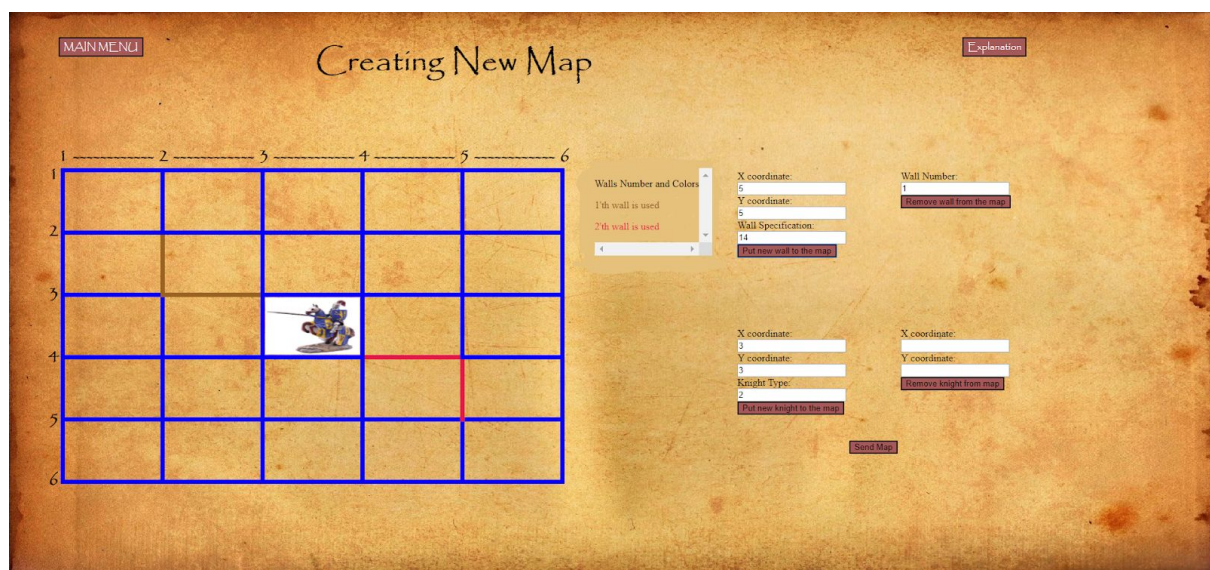
When login is verified, user is directed to the main menu. User can click play to proceed to choose the game mode. Also user can go to the create new maps to design his/her custom map, or highscores to see global highscores, and his/her rank.



In the play menu, user can choose which game mode he/she want to play to start game. Also can take information about the game with clicking How to Play button.



In the game, user can place the walls by providing coordinates. He/she can go back to the main menu with the button at the left top. After placing all walls, if solution is correct, user will be redirected to the next challenge. Otherwise, he/she will be asked to revise the solution. Users are also able to get some hints about the challenge by clicking “HINT”. In time modes, user has to solve the challenge in a given time. If time is up, it is possible to get extra time by consuming hints.



In map creation page, users can create their own maps. To create a map, user has to place red knights, blue knights, and optionally white knights with correct wall specification. If

the created map is applicable, it will be submitted for evaluation. Later on, it might be added to the game challenges.

Game Page
localhost:8080/Highscores

MAIN MENU

HIGH SCORES

List	User	Score
1	gulnihal	8
2	ates	6
3	ozan	5
4	enes	3
5	samil	1
11	admin	0

List	User	Score
1	enes	3
2	ozan	3
3	samil	2
4	gulnihal	2
5	ates	1
11	admin	0

List	User	Score
1	ates	8
2	samil	3
3	gulnihal	3
4	enes	1
5	ozan	1
11	admin	0

List	User	Score
1	ates	9
2	ozan	8
3	gulnihal	7
4	enes	5
5	samil	4
6	admin	2

In highscores page, top 5 scorers of each game mode are listed in corresponding tables of game modes. Below the tables, user can see his/her own score and their own rank.