PROJECT PLAN:
WEB-BASED MULTIPLAYER GAME

Supervisor: Dr. Vincent Lau

Takumi Takeshige and Hyunse Yoon
3035374242 & 3035343281

Table of Contents

1. Introduction

1.1 Background

Ever since the foundation of the Internet, the number of Internet users has been increasing exponentially. The number of Internet users on June 30th, 2019 was found to be around 4.3 billion, which is more than half of the world population at that time [1, 2]. Simultaneously, video games have become a fast growing industry in this era that has risen to popularity. In 2019, around a third of the population are considered video game players [3]. When merging these two technologies, the development of web-based multiplayer game becomes possible. Web-based multiplayer game is a game in which two or more players play the same game at the same time to cooperate or compete on a web browser. In this project, a web-based multiplayer game influenced from two existing board games, chess and Dungeons & Dragons, will be developed, and it will provide a fair competition between players via a match-making system.

1.2 Objective

The first main objective of this project is to create a web-based multiplayer competitive game where a system is implemented so that players are matched with other players of similar proficiency in the game. This system will be dubbed “match-making system (MMS)”. This allows for players to have a wholesome experience playing the game as the opponent will pose enough of a challenge while at the same time not dominate the player.

The second objective is to create a game that will enhance the players’ creativity and tactical decision-making skills. Selection of characters before each game to suit the randomised tiles facilitates players to adapt their strategies uniquely to each game, stimulating them with creativity and theory-crafting further than chess would, where the beginning of each game is standardised and unchanging.

2. Deliverables

This project will essentially have three deliverables: a web-based multiplayer game, an integration of an MMS and a login system on the said game.

One of the products provided upon completion of this project will be a web-based multiplayer game that allows users to play on a browser using their personal computers or touch-based devices. The game will be a competitive turn-based strategy board game for two people, much like chess, except the pieces played, called characters in this game, and what these characters can do on the board, namely the move set of a character, can be modified by the user.

In addition to a web-based multiplayer game, an MMS will be implemented to provide fair competition to people. The server will collect and analyze the data of users and it will use these data to search for pairs of users queuing for the next match with similar skill in the game as possible.

In order to deliver an MMS, a login system will be required to uniquely identify each user. User accounts will be stored in the server database for the use of the MMS. In addition, each user will be able to save their “team compositions”, the sets of characters with their selected choice of move sets as to alleviate the need to compose the team of characters from scratch each game, or a “friend list”, a list of other accounts that the player wishes to remember and compete with in the future, in their account.

3. Employed Technologies

The implementation of the proposed web application will be accomplished using the MERN stack. To facilitate the creation of the video game, Java will be used.

3.1 The MERN Stack

The acronym stands for MongoDB, Express, React.js and Node.js. A point to note is that all components of the MERN stack is built upon JavaScript, Therefore, MERN stack is a crucial technology for this project.

3.1.1 MongoDB

In this project, MongoDB, which is a NoSQL database [4], will be used. Using MongoDB will be more suitable than using SQL-based databases because MongoDB has greater flexibility with regard to storing data. SQL databases is capable of storing data only in tables, but MongoDB can store data in tables, key-value based storage, document-based storage, and more [4]. MongoDB has lack of strongly-defined schema, which will allow more flexibility of design and more scalability of database.

3.1.2 Express.js and Node.js

Express is a framework for the development of web applications, such as web-based multiplayer game, for Node.js [5]. Node.js is the JavaScript-based asynchronous runtime environment, which will be used in the development of the back-end of the game [6].

3.1.3 React

React is a JavaScript-based library that is used to create fast and lightweight user interfaces [7]. In terms of Model-View-Controller model, which is a web application design model that categorizes 3 fundamental components of web application, React integrates the “View” component. Using React is more suitable than using Angular for this project because React uses Virtual Document Object Model, which is a extremely abstracted model of the structure of an HTML document [8].

3.2 Java

Java is a general-purpose programming language that is supported by a wide range of extensions and libraries [9]. In addition to its vast libraries and extensions, it supports object-oriented programming, which is a language model that will be used in this project. Furthermore, because Java uses Just-In-Time (JIT) compiler, high-performance can be expected from the proposed game [10].

4. Feasibility assessment

4.1 Scope of work

As the project focuses on the MMS and the enhancement of players' creativity, the video game need not be "flashy" for the sake of the project, so long as the core element of it is there such that the game is at least playable. As such, visual graphics will be limited, and there will be no audial implementation. Also, as the project focuses on creating a web-based multiplayer game, the game's accessibility will be limited to the web-browser alone, though during development, we intend to first implement the game in Java for a smoother development of the whole project.

On the backend, as previously mentioned, there will be an MMS. In order for that to be implemented, there will also be a login system to identify each player for the purpose of the MMS. On top of that, the project will include a direct matching system to give players a choice on who they want to play against instead of letting the MMS decide.

4.2 Risks and Mitigation

A common risk that many games with an MMS face is a potential exploitation by the players to skew the MMR from where they should actually place to be able to face easier or harder opponents than they should be. Once a player finds out how they can influence their MMR, it is possible for players to play the game with the intention of altering their MMR, resulting in a misrepresentation of their skill and therefore tricking the MMS into placing them into unfair games, whether to the player abusing it or the unfortunate opponent. A mitigation to this is to conceal the MMR from the player, presenting a difficulty in finding out what exactly influences their MMR and therefore exploiting the MMS.

Another potential problem to consider involves the hosting of the game online. An approach that would avoid expending personal or provided resources is the use of game hosting services, i.e. itch.io. Itch.io is a web platform for individual game developers to handily upload their work without any financial repercussions [11]. This will be the temporary solution to where the game will be hosted.

5. Methodology

This project will be carried out in two steps in broad terms: game development on localhost, and then implementation of the server with an MMS from the backend.

5.1 Game development on localhost

Development will take a top-down approach, starting with a basic infrastructure of the game and then fine-tuning the smaller aspects. For example, the creation of the framework of the database and objects required will be the foundation of this game's development. After designing the framework, implementation of game logic will be built based on this framework. Then, the integration of graphical assets into the game logic will be applied. This development will initially take place on Java. After the completion of game design, JavaScript is used to implement the client and server and display the game on a localhost website.

5.2 Implementation of the server with an MMS

Once the game development is over, the implementation of the server, alongside everything relevant to the MMS, which is handled in the backend by the server, will commence, starting with the transference of the backend from localhost to the server. Following that will be the implementation of a login system with its database of each player and their in-game statistics to preemptively have an array of users with their statistics for use of the MMS as soon as it is deployed.

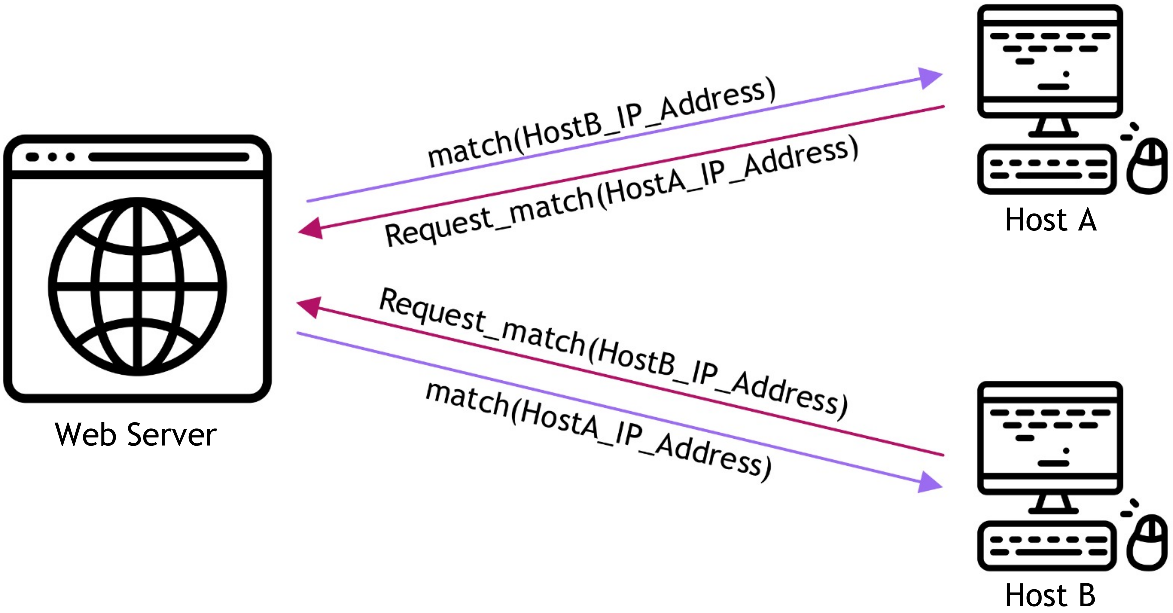

Figure 1: Server-client interaction for MMS

After the login system will be the MMS. As seen in Figure 1, players will request to the server to queue for a match, and once the MMS finds two suitable players it will respond to them and the match will begin. It will make use of the match-making ranking (MMR), a number assigned to each player that determines their level in the game. The higher it is, the better that player is in the game, and vice versa. The

MMS can then find two players with as little difference in MMR as possible as a way to fairly match two players. The MMR of a player will be determined by a number of factors derived from their statistics of previously played games.

6. Project schedule

Period	Objective and Milestone
Sep. 20 ~ Sep. 30	● Obtain relevant resources
Oct. 1 ~ Nov. 21	● Game development in Java
Nov. 22 ~ Dec. 22	● Translation to web browser on local host
Dec. 23 ~ Jan. 10	● Game testing & balancing ● First presentation
Jan. 11 ~ Feb. 11	● 1st Interim report (due Feb. 1) ● Translation to server ● Login system ● Direct matching system
Feb. 12 ~ Mar. 11	● Match-making system
Mar. 12 ~ Apr. 19	● Final testing & debugging ● Final project report ● Final presentation

7. References

[1] "World Internet Users Statistics and 2019 World Population Stats", Internetworldstats.com, 2019. [Online]. Available: https://www.internetworldstats.com/stats.htm. [Accessed: 24 Sep. 2019].

[2] "World Population Clock: 7.7 Billion People (2019) - Worldometers", Worldometers.info, 2019. [Online]. Available: https://www.worldometers.info/world-population/ [Accessed 24 Sep. 2019].

[3] T. Wijman, “The Global Games Market Will Generate $152.1 Billion in 2019 as the U.S. Overtakes China as the Biggest Market,” 18 Jun 2019. [Online]. Available: https://newzoo.com/insights/articles/the-global-games-market-will-generate-152-1-billion-in-2019-as-the-u-s-overtakes-china-as-the-biggest-market/. [Accessed 22 Sep. 2019].

[4] "What Is MongoDB?", MongoDB, 2019. [Online]. Available: https://www.mongodb.com/what-is-mongodb. [Accessed: 24 Sep. 2019].

[5] Node.js Foundation, "About | Node.js", Node.js, 2019. [Online]. Available: https://nodejs.org/en/about/. [Accessed: 24 Sep. 2019].

[6] "Express - Node.js web application framework", Expressjs.com, 2019. [Online]. Available: https://expressjs.com/. [Accessed: 24 Sep. 2019].

[7] "React – A JavaScript library for building user interfaces", Reactjs.org, 2019. [Online]. Available: https://reactjs.org/. [Accessed: 24 Sep. 2019].

[8] "MVC: Model, View, Controller | Codecademy", Codecademy, 2019. [Online]. Available: https://www.codecademy.com/articles/mvc. [Accessed: 24 Sep. 2019].

[9] "Java Resources for Students, Hobbyists and More | go.Java | Oracle", Go.java, 2019. [Online]. Available: https://go.java/index.html. [Accessed: 24 Sep. 2019].

[10] "IBM Knowledge Center", Ibm.com, 2019. [Online]. Available: https://www.ibm.com/support/knowledgecenter/en/SSYKE2_8.0.0/com.ibm.java.vm.80.doc/docs/jit_overview.html. [Accessed: 24 Sep. 2019].

[11] "About itch.io", itch.io, 2019. [Online]. Available: https://itch.io/docs/general/about. [Accessed: 26 Sep. 2019].