Version

1.3

ONLINE GAMe Player Matching

KSU SWE6813 Team 3 Spring 2022 – Project Proposal

**Online Player Matching**

SWE 6813: Webservice ENGINEERING SPRING 2022 Group 2

Online Gaming Project Proposal

Prepared by: KSU SWE6813 Software Service Group 2  
Project Sponsor: Dr. Nasrin Dehbozorgi

Project Team:

Jerry Cowell (Team Leader)

Steven Carver

Takyi Obeng

Kyla Okobah

Frank Wear

Project Owner: Takyi Obeng

Jerry Cowell: Scrum Master

**Introduction**

This project provides solutions to issues of online matchmaking gaming. For our project, the team aims at developing an effective application which will provide personal connections for players to be matched with similar skill caliber players together. The application will take into consideration various constraints like personal level interaction, accessibility of data for matchmaking, and the possibility of players engaging in an online playground where there is reduced hostile and volatile communications.

**2. Project Overview**

**2.1 Document Overview/Addressing Comments about the Proposal**

This document is intended to outline the scheduled project plan along with the proposal of the project, the list of primary deliverables, as well as project work and product estimates. It contains all project resources’ requirements, project dependencies and constraints, project risks, a description of the project development process, and the project quality and measurement plans. The purpose of this document is to guide the members of the project team during the phases of designing, developing, and testing the system and involving the product owner at every stage of the product development.

**2.2 Project Goal**

The purpose of this project is to design and build an application that will resolve existing matchmaking issues by matching players on criteria beyond simple skill. The application will enable users to accomplish this through a process of collecting more information about players, parameters related to playstyle, preferences, and attitude. The App will reintroduce components to online gaming, which will help teammates to brighten their horizon with improved gaming experience. And it will match players from other games by providing connectivity between players of different teams. The application will be a Web App with web-based technology, which will be dynamic and uses web services and visualization.

The project is identified as a Multiplayer Matchmaking Application, with a design framework, which will center on the architectural requirements specified.

**2.3 Requirements**

* The application will be a Web App with web-based technology, which will be dynamic and uses web services and visualization.
* Provide an easy way to reintroduce human elements to online teammates and to increase the range of their knowledge, understanding, and experience.
* Provide means to obtain data from in-game sources and cross it with players and user generated behavior profiles.
* Provide an easy way to match players with teammates who suit both their skill and behavior.
* Provide an easy way to collect every necessary information about a gamer including measures related to playstyle, preferences, and attitude.
* Provide an application which enables users to find players from other games that have

both their skill and behavior. Provide an uncomplicated way for gamers to make friends

and form groups with like-minded players.

**2.4.1 Project Platform**

* Azure DevOps

Link:

* PowerApp

Link:

* Github

Link:

**2.4.2 Identification**

Title: Multiplayer Matchmaking Application

Version: 1.1

Release Date: 02.20.22

**2.5 System Overview**

Takyi Obeng, the project owner, wants a system where users can create and send a request to a web server over the internet through the application's user interface. The application, as a web server, must send the request to the web application server. Also, the owner wants the web application server to execute the request task and generate the results of the required data. The web application server must send the requested information or processed data of the requests to the web server. The web server must carry the requested information to the client, and the required information must appear on the user’s display or screen.

**2.6 Background and Objectives**

Currently, online gaming is a huge industry which is growing at a higher rate with equal expansion of its audiences. Unfortunately, the growth in online gaming does not have equal technology which matches the players meaningfully. Because the technology is lacking behind, there is a need for an updated system to replace the system where game metric and statistics service as means for matching players of similar skill caliber to play together. And because the technology is lacking behind, players are often matched using usernames which lack personal connectivity.

**2.7 Assumptions and Constraints**

Constraints which apply to the current system are that the current system falls short of the expectations of the online players. It does not have the functionality for adding personal level to the game, and it has occurrence of hostile and volatile communication. Consequently, players use third-party applications to match players across multiple games.

**2.7.1 Users or Involved Personnel**

The current users of online gaming applications are online gamers who only will need the application to match players of other online games. Users will then have full access to all functionalities for selecting players of similar skills. The project owner, through the system, will create and send a request to a web server over the Internet through the application's user interface.

**2. 7.2 Target Customers**

The target audience for this project is gamers interested in multi-player online gaming experiences. A study conducted in 2020 by Entertainment Software Association (ESA) found that more than 214 million Americans play at least one hour of videos games per week. 65% of these players are playing with other players.

**Graphical user interface

Description automatically generated**

                        Diagram #1: The ESA Data, retrieved from www.theesa.com

The ESA diagram shows players spend 6.6 hours a week playing with other players online. 17% of the players play with people they met online. This means that our target audience will fall within this range, although the audience is potentially much more.

**3. Development Background/Approach**

The team will be developing the new web application using JavaScript and NodeJS. NodeJS will be used for the frontend, and () for the backend.  We will integrate the UI (User Interface) with database functionalities. This will allow users to access the players’ database. Some members of the team are familiar with NodeJS and () and will be able to guide the rest of the team through the design and implementation of the system.

**3.1 High Level Estimates**

It is estimated that First Sprint will take the team 20 hours to complete by February 24. During this phase, our major project deliverable will be our Project Plan and proposal. Other deliverables will involve second, third, and final Sprints, and project schedule. We will be conducting two meetings per week, ranging from 30 min to 1 hour per meeting. We will also be conducting emergency meetings as the workload demands. The remaining time will be spent on

individual research and work.

**3.3 Key Contacts and Stakeholders**

**Name**: Takyi Obeng

**Role:** Product Owner and Subject Matter Expert

**Description:**

This stakeholder has the greatest interest in the success of the web application as the system directly impacts the number of gamers, and he will patronize the system. Mr. Takyi Obeng, in his capacity as a developer himself, can provide us with domain knowledge to make the application the best it can be.

**Name**: Jerry Cowell

**Role:** Scrum Master

**Description:**

This stakeholder has the greatest interest in the success of the web application ……

**Name**: Dr. Nasrin Dehbozogi

**Role:** Project Patron

**Description:**

This stakeholder is a professor at Kennesaw State University who is responsible for tutoringWebservice Engineering. And she is the one who will provide the team withspecific details about the entire process of running this course, including tasks and responsibilities of the team**.**

**Email:** Dnasrin@kennesaw.edu

**Role:** Online gamers

**Description:**

These are the users who will patronize the Web Application and whose feedback will impact online gaming business operations. If they decide to patronize, the system has been successful.

**4. FEATURES**

**4.1 Features List**

* Provide an effortless way to reintroduce human elements to online teammates and to

increase the range of their knowledge, understanding, and experience.

* Provide means to obtain data from in-game sources and cross it with players and user

generated behavior profiles.

* Provide an effortless way to match players with teammates who suit both their skill and

behavior.

* Provide an uncomplicated way to collect every necessary information about a gamer

including measures related to playstyle, preferences, and attitude.

* Provide an application which enables users to find players from other games that have

both their skill and behavior. Provide a straightforward way for gamers to make friends

and form groups with like-minded players

**5. Project Schedule**

|  |  |  |  |
| --- | --- | --- | --- |
| Task Name | Duration | Start | Finish |
| **Project Planning** | **14 days** | 02.08.22 | 02.22.22 |
| Role/Skill Planning | **6 days** | 02.22.22 | 02.28.22 |
| Project Document Template | **1 day** | 03.01.22 | 03.01.22 |
| Project Schedule | **1 day** | 03.02.22 | 03.02.22 |
| Define Project Tasks | **2 days** | 03.02.22 | 03.04.22 |
| ***PROJECT PLAN DUE DATE*** |  | ??? | |
| **Requirement Analysis** | **7 days** | 03.05.22 | 03.12.22 |
| Process Overview Analysis | **2 days** | 03.12.22 | 02.14.22 |
| Define Functional Requirements | **2 days** | 03.14.21 | 03.16.22 |
| Create Use Case Diagram | **3 days** | 03.16.21 | 03.19.22 |
| **Define Non-Functional Requirements** | **2 days** | 03.19.22 | 03.23.21 |
| Risk Analysis | **1 day** | 03.22.21 | 03.22.22 |
| Other Requirements | **1 day** | 03.23.22 | 02.24.22 |
| **Design** | **13 days** | 03.24.23 | 04.07.22 |
| Define Architecture | **3 days** | 04.07.22 | 04.10.22 |
| Define Technical Specifications | **2 days** | 04.10.22 | 04.12.22 |
| ***MIDTERM DELIVERABLE DUE DATE*** |  | ??? | |
| **UML Diagrams** | **2 days** | 04.12.22 | 04.14.22 |
| UML Class Diagram | **2 days** | 04.14.22 | 04.16.22 |
| UML Sequence/Activity Diagrams | **1 days** | 04.05.22 | 04.06.22 |
| **Database Design** | **2 days** | 04.06.22 | 04.07.22 |
| ERD Diagram | **1 days** | 04.07.22 | 04.08.22 |
| **UI Design** | **2 days** | 04.08.22 | 04.10.22 |
| Screen Design | **2 days** | 04.10.22 | 04.12.22 |
| Screens Wireframing | **1 days** | 04.12.22 | 04.13.22 |
| **Quality Assurance Planning** | **3 days** | 04.13.22 | 04.16.22 |
| Create Test Plan | **2 days** | 04.16.22 | 04.18.22 |
| ***DESIGN DOCUMENT DUE DATE*** |  | ???? | |
| Create User Manual | **2 days** | 04.18.22 | 04.20.22 |
| **Development** | **10 days** | 04.20.22 | 04.30.22 |
| Create Source Code Repositories | **3 days** | 04.30.22 | 05.03.22 |
| Database Configuration | **2 day** | 05.03.22 | 05.5.22 |
| Requirement Implementation | **6 days** | 05.05.22 | 05.11.22 |
| **Testing** | **4 days** | 05.11.22 | 05.15.22 |
| Follow Test Plan | **4 days** | 05.15.22 | 05.19.22 |
| **Deployment** | **4 days** | 05.19.22 | 05.23.22 |
| Deploy Application Version | **2 days** | 05.23.22 | 05.25.22 |
| ***TEST DOCUMENT DUE DATE*** |  | ????? | |
| Integration Testing | **4 days** | 05.25.21 | 05.29.22 |
| ***FINAL SOFTWARE PRODUCT DUE DATE*** |  | ???? | |
|  | | | |
| **WEEKENDS ARE EXCLUDED** | | | |

**Work Breakdown Structure:**

* Project Planning - 14 days (about 2 weeks)
* Requirement Analysis - 7 days
* Design (UML diagrams, UI, and database) - 13 days (about 2 weeks)
* Development (Repositories, Database configuration, and Requirement

implementation) - 10 days (about 1 and a half weeks)

* Testing of application - 8 days
* Deployment of application - 6 days
* Final Launch

**5.2 Project Status Tracking & Working Meeting Minutes**

The team will track, on a regular basis, project progress and the minutes of scheduled meetings, which include attendees, discussion of items, risks, deliverable status, as well as the reports of tasks assigned to all the team members.

**6. Project Estimate**

**6.1 Estimate Summary**

The following tables summarize the product size and effort estimates:

|  |  |  |  |
| --- | --- | --- | --- |
| **Project** | **Estimate Attributes** | | |
| **Size** | |  |
| **WBS areas** | **Unit of Size** | **Size** | **Effort** |
| **Total Requirements Effort**  (Includes feature-related and "other" (non-feature) Requirements work) | 1-10 | 6 | 5 |
| **Feature Related Requirements Size and Effort Totals** | PAGES | 10 | 4 |
| **Total Development Effort**  (Includes feature-related and "other" (non-feature) Development work) | 1-10 | 6 | 6 |
| **Feature Related Development Coding Size and Effort Totals** | LOC | 2000 | 6 |
| **Feature Related Development Documentation Size and Effort** | PAGES | 20 | 5 |
| **Total Testing Effort**  (Includes feature-related and "other" (non-feature) Testing work) | 1-10 | 5 | 4 |
| **Feature Related Testing Size and Effort Totals** | TEST CASES | 9 | 4 |
| **Feature Level Effort Total**  (From Feature Estimate Worksheet) | 1-10 | **5** | 5 |
| **Development Effort Total**  (Includes Feature Level and project level overhead for Requirements, Development, and Testing) | 1-10 | **8** | 7 |
| **Project Level Effort Total**  (From Project Level Effort Estimates worksheet, excluding requirements, development, and testing) | 1-10 | **6** | 6 |
| **Project Total Effort** (Project Totals + Feature Totals) | 1-10 | **6** | **7** |

**7. Project Resource Requirements**

**7.1 Staffing/ Skill Requirements**

**Name:**Jerry Cowell Jr

**Role:** Team Leader / Scrum Master

**Critical Skills: UX Frameworks, UI, Marketing,**

Technical writer, Java,,,,, Team Management

**Skill Gaps: IDE,GIT**

**Name:**Steven Carver

**Role:** Technical Lead

**Critical Skills:**    
Front and backend development experience, Java, ,,,,,,,,

**Skill Gaps:**    
??

**Name:**Frank Wear

**Role:** Systems Architect / Developer

**Critical Skills:** Javascript, PHP, Dynamic Web Design, Java, HTML, CSS

**Skill Gaps:** IDE, GIT, Azure/PowerApps operations and resources

**Name:**Kyla Okobah

**Role:** Technical Lead

**Critical Skills:** Unit Testing, Creating Test Cases, Java, HTML, Node.JS

**Skill Gaps:**

**Name:** Takyi Obeng

**Role:** Product Owner, Quality Manager

**Critical Skills:**

Unit Testing, Writing Test Cases, Java, CSS, HTML, Database design, Angular, React

**Skill Gaps:**

My programming experience is solely academic.

**7.2 Plan to Fill Skill Gaps**

Of the team members, there may be some skill gaps where other members are experienced. Team members who are more experienced in coding will provide relevant resources to assist other members to understand and contribute to the progress of the work. All team members are expected to dedicate their personal time and make the necessary efforts outside of group meetings to increase their knowledge of relevant materials to be an asset to the team.

**8.** **Dependencies/Constraints**

The software system will be developed and implemented using JavaScript and Node. Both frontend and backend interfaces will be user-friendly which can be updated. The system will be designed, developed, tested, and delivered, along with all appropriate documentation by May (…), 2022.

**9. Risk Management**

**9.1 Risk Management Strategy**

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Potential Impact** | **Mitigation Plan** | **Priority**  **(1- highest priority)** |
| Misunderstanding of requirements | Delayed project deliverables  Deliverables not what expected.  Reprogramming costs | Periodic meetings and affirmation of requirements with customers | 1 |
| Constant change of requirements | Delayed project deliverables  Additional training costs  Additional planning and programming | Go over plan with customer for completeness. | 2 |
| Misunderstanding of new methodology | Delayed project deliverables  Additional training costs | Invest in training before the project starts | 3 |
| Schedule overruns | Delayed project deliverables | Periodic status meeting to make sure deliverables are on schedule | 4 |

**10. Project Process**

**10.1 Software Life Cycle Model**

The team is developing this project using the Agile methodology; we are using GitHub, an open-source tool, to manage our product backlog. The requirement elicitation phase is scheduled to be completed by February 24, 2022. The team has divided the tasks into separate phases, projecting to begin the design phase soon after the elicitation phase and follow up with implementation, testing, installation, troubleshooting, and ends up at operation and maintenance.