

Software Requirements Specification

for

CS320 Project Play Game Online Free

Version <0.1>

Prepared by

Group Name: Gamers Rise Up

|  |  |  |
| --- | --- | --- |
| Quinlan Boney | <student #> | q.boney@wsu.edu |
| Keegan Sanchez | 11641250 | [keegan.sanchez@wsu.edu](mailto:keegan.sanchez@wsu.edu) |
| Riley Moreland | 11576770 | [riley.moreland@wsu.edu](mailto:riley.moreland@wsu.edu) |

|  |  |
| --- | --- |
|  |  |
| Date: | 11/06/2020 |
|  |  |
|  |  |
|  |  |

Content

Revisions iii

1 Introduction 1

1.1 Document Purpose 1

1.2 Product Scope 1

1.3 Intended Audience and Document Overview 1

1.4 Definitions, Acronyms and Abbreviations 1

1.5 Document Conventions 1

1.6 References and Acknowledgments 2

2 Overall Description 3

2.1 Product Perspective 3

2.2 Product Functionality 3

2.3 Users and Characteristics 3

2.4 Operating Environment 3

2.5 Design and Implementation Constraints 4

2.6 User Documentation 4

2.7 Assumptions and Dependencies 4

3 Specific Requirements 5

3.1 External Interface Requirements 5

3.2 Functional Requirements 6

3.3 Behavior Requirements 6

4 Other Non-functional Requirements 7

4.1 Performance Requirements 7

4.2 Safety and Security Requirements 7

4.3 Software Quality Attributes 7

5 Other Requirements 8

Appendix A – Data Dictionary 9

Appendix B - Group Log 10

Revisions

| Version | Primary Author(s) | Description of Version | Date Completed |
| --- | --- | --- | --- |
| Rough Draft, Number 1 | Riley Moreland  Keegan Sanchez  Quinlan Boney | Initial rough draft. | 11/05/20 |
| Rough Draft, Number 1.1 | Riley Moreland  Keegan Sanchez  Quinlan Boney | Rough draft expanded upon. | 11/06/2020 |

# Introduction

This project is an online version of a board game that has an associated site with information regarding the game and players. In this section readers will find information assisting how to interpret and understand the product.

## Document Purpose

<Identify the product whose software requirements are specified in this document, including the revision or release number. Describe the scope of the product that is covered by this SRS, particularly if this SRS describes only part of the system or a single subsystem.

TO DO: Write 1-2 paragraphs describing the purpose of this document as explained above.>

Purpose: This document is intended to describe the overall system of the product created by our group. This should include the general description of the product itself, how it should behave, and what it must have.

-Revision/Release Number: 0.1

## Product Scope

<Provide a short description of the software being specified and its purpose, including relevant benefits, objectives, and goals.

TO DO: 1-2 paragraphs describing the scope of the product. Make sure to describe the benefits associated with the product.>

Scope: This software is a virtual representation of the game Checkers. The goal is to have a fully functioning site in addition to the product that provides context to the software. Our objectives are to implement leaderboards, rules, and the game itself. The benefit of this software is its ability to represent the game as well provide information for users to see.

- This software will require access to a machine capable of using the internet on a modern browser, we will be using blank to store information such as wins/losses, games played and then updating the associated pages periodically. We plan to solely have an associated username to attach stats to a user and no further data to be provided except that which we extract to create our leaderboard page. It also will be free to play. (NEED to confirm this with group)

## Intended Audience and Document Overview

<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers (In your case it would probably be the “client” and the professor). Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>

Intended Audience: Client, Professor, Testers

Describe the rest of the SRS: The rest of the document features a more in-depth description of the product in section 2. Section 3 which focuses on the necessary requirements of the product, section 4 which has listed non-requirements and section 5 which are optional requirements.

- The client will want to focus on sections 3 and 4 first as to ensure their expected or unexpected demands are met. For the professor reviewing the entire document from start to finish would be the best course of action save section 5. Testers should read the document in the order of 3,4,5 as they want to ensure the functionality of the product.

## Definitions, Acronyms and Abbreviations

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.

TO DO: Please provide a list of all abbreviations and acronyms used in this document sorted in alphabetical order.>

Acronyms(alphabetical): GRU (GamersRiseUp)

| Term | Definition |
| --- | --- |
|  |  |
| DO | Digital Ocean |
| GRU | Gamers Rise Up |
| Pages (new term needed maybe?) | The landing page that is also connected to the leaderboards and game pages. |

## Document Conventions

<TO DO: Describe any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance. Sometimes, it is useful to divide this section to several sections, e.g., Formatting Conventions, Naming Conventions, etc.>

Conventions used: In general, this document follows the IEEE formatting requirements, using Arial font size 11, or 12 throughout the document for text. It uses italics for comments. Document text is single spaced and maintains a 1” margin.

## References and Acknowledgments

<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document.

TO DO: Use the standard IEEE citation guide (attached) for this section.>

- Need to cite: DigitalOcean for the information used, other sources of information too like the WebSocket API link in discord

# Overall Description

## Product Perspective

<Describe the **context** and **origin** of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. In this part, make sure to include a simple diagram that shows the major components of the overall system, subsystem interconnections, and external interface. In this section it is crucial that you will be creative and provide as much information as possible.

TO DO: Provide at least one paragraph describing product perspective. Provide a general diagram that will illustrate how your product interacts with the environment and in what context it is being used, i.e., context diagram.>

This product is considered a follow-on member of a product family specifically of physical board games (Tic-Tac-Toe/Chess). We are taking inspiration for board games, and attempting to apply this to an online setting, to make it easier to engage with. Furthermore, during Covid-19 quarantine, face to face interaction is limited, so creating an online version of the game will allow those who are separated to still play together.

There are a few key pieces of this project

- Game (code/assets)

- Website (landing page/game page/leaderboard page)

--

- The system consists of a WebSocket and server combo that relays information regarding the game state between users. (we can try and base the diagram off this? I would need to confirm with Keegan that this statement is true.) basically diagram would show the link of the game with the WebSocket API (implement term for WebSocket in 1.4)

## Product Functionality

<Summarize the **major functions** the product must perform or must let the user perform. Details will be provided in Section 3, so only a high level summary is needed here. Organize the functions to make them understandable to any reader of the SRS. A picture of the major groups of related requirements and how they relate, such as a top level data flow diagram or object class diagram, will be effective.

TO DO:

1. Provide a bulleted list of all the major functions of the system

- Will be able to interact and use the game

- Will be able to play against an AI or another User

- Will be able to see statistics and other data regarding the game

**NEEDS FURTHER REVISION**

--

- Attach a summary explaining functions, such as below (feel free to correct/use)

- Users will be able to navigate to the landing page and then progress to the game using this page with the start menu to start a match. At this point the user will have had the option to choose to play against either another player or against an automated AI (difficulty settings needed maybe?). From the same landing page users will also have access to the other pages that provide stats and data regarding the game consisting of leaderboards and tutorial/rules. The WebSocket will maintain the system and facilitate the information needed for the game to operate.

## Users and Characteristics

<**Identify** the various **users** that you anticipate will use this product. Users may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience.

TO DO:

1. Describe the pertinent characteristics of each user. Certain requirements may pertain only to certain users.

3. Distinguish the most important users for this product from those who are less important to satisfy.>

| User | Frequency of Use | Functions Used | Technical Expertise | Access To Source | Education Level | Importance |
| --- | --- | --- | --- | --- | --- | --- |
| Tester | Each phase of development | Current features for testing | Limited/None | No | Any | 3rd |
| Professor | After completion | All | Lots | Yes | Professor | 2nd |
| Client (Player) | After completion | All | Limited/None | No | Any | 1st |

## Operating Environment

<Describe the **environment** in which the software will operate, including the **hardware** platform, **operating** **system** and versions, and any other **software** **components** or applications with which it must peacefully coexist. In this part, make sure to include a simple diagram that shows the major components of the overall system, subsystem interconnections, and external interface.

TO DO: As stated above, in at least one paragraph, describe the environment your system will have to operate in. Make sure to include the minimum platform requirements for your system. >

The web pages will be hosted on an Ubuntu Apache HTTP server, which will handle connecting clients. The pages should display and function well in most modern browsers (including Google Chrome, Microsoft Edge, and Firefox).

The game server will run in Ubuntu, and will be deployed onto a DigitalOcean server.

## Design and Implementation Constraints

<Describe any items or issues that will limit the options available to the developers. These might include: hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer’s organization will be responsible for maintaining the delivered software).

TO DO: In this section you need to consider all of the information you gathered so far, analyze it and correctly identify relevant constraints.>

The target size of the virtual machine is 1 GB of memory, 1 CPU, a 25 GB SSD for storage, and 1 TB of data transfer per month, with a target hosting cost of 5$ a month.

## User Documentation

<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.

TO DO: You will not actually develop any user-manuals, but you need to describe what kind of manuals and what kind of help is needed for the software you will be developing. One paragraph should be sufficient for this section.>

The product will have a landing page consisting of a brief tutorial on playing the game and connecting to other users.

- (Continuing off of above sentence) This brief tutorial should provide enough detail describing the rules and expectations of the game as well as how to interact with the software. It should then have another section dedicated to explaining how to interact with other players with the menus. (anything else we need to help someone with?)

## Assumptions and Dependencies

<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project.

TO DO: Provide a short list of some major assumptions that might significantly affect your design. For example, you can assume that your client will have 1, 2 or at most 50 Automated Banking Machines. Every number has a significant effect on the design of your system. >

We are assuming that we will be able to host on DigitalOcean using the 5$ tier.

# Specific Requirements

## External Interface Requirements

### User Interfaces

<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., Cancel) that will appear on every screen, error message display standards, and so on. Define the software components for which a user interface is needed.

TO DO: The least you can do for this section is to describe in words the different User Interfaces and the different screens that will be available to the user. Optional: You may also provide an initial Graphical User Interface design (does not have to be final).>

The user interface will be the website consisting of

A landing page, with links to leaderboards, and game.

A leaderboard page with a return link to the landing page.

The game, which will consist of two menus:

- The starting menu, which includes options for single or multiplayer. If multiplayer is selected, a box to enter the other clients connect code will be listed, or to host the game.

- The pause menu, which has two options, resume, or exit to landing page.

--

-

### Hardware Interfaces

<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware. You are not required to specify what protocols you will be using to communicate with the hardware, but it will be usually included in this part as well.

TO DO: Please provide a short description of the different hardware interfaces. If you will be using some special libraries to communicate with your software mention them here. In case you have more than one hardware interface divide this section into subsections.>

We will be using WebSockets to handle sending and receiving data, using the **LIBRARY** with the **PROGRAMMING LANGUAGE**.

### Software Interfaces

<Describe the connections between this product and other specific software components (name and version), including databases, operating systems (Windows? Linux? Etc…), tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.

TO DO: The previous part illustrates some of the information you would usually include in this part of the SRS document. To make things simpler, you are only required to describe the specific interface with the operating system.>

Code.

### Communications Interfaces

<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.

TO DO: Do not go into too much detail, but provide 1-2 paragraphs were you will outline the major communication standards. For example, if you decide to use encryption there is no need to specify the exact encryption standards, but rather, specify the fact that the data will be encrypted and name what standards you consider using. >

We will be using the WebSocket API, and HTTP, via the **BLANK AND BLANK LIBRARIES**. As no important user data is being transferred, we will not use any encryption.

## Functional Requirements

*< Functional requirements capture the intended behavior of the system. This behavior may be expressed as services, tasks or functions the system is required to perform. This section is the direct continuation of section 2.2 where you have specified the general functional requirements. Here, you should list in detail the different product functions with specific explanations regarding every function.*

*TO DO: Break the functional requirements to several functional areas and divide this section into subsections accordingly. Provide a detailed list of all product operations related to these functional areas.>*

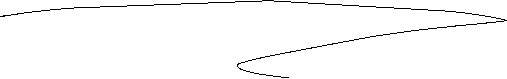
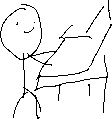
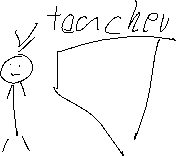
* **WEBSITE**
  + Landing Page
  + Leaderboard Page
  + Game Page
* **GAME SERVER**

## Behavior Requirements

### Use Case View

<A use case defines a goal-oriented set of interactions between external actors and the system under consideration.

TO DO: Provide a use case diagram which shows the entire system and all possible actors. Do not include detailed use case descriptions (these will be needed when you will be working on the Test Plan), but make sure to include a short description of what every use-case is, who are the actors in your diagram.>



# Other Non-functional Requirements

## Performance Requirements

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.

TODO: Provide relevant performance requirements based on the information you collected from the client. For example you can say “1. Any transaction will not take more than 10 seconds, etc…>

Applying user’s moves should take less than a second.

## Safety and Security Requirements

<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied. Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements.

TODO:

* Provide relevant safety requirements based on your interview with the client or, on your expectation for the product.
* Describe briefly what level of security is expected from this product by your client and provide a bulleted (or numbered) list of the major security requirements.>

No important user data is being transferred, so we will not be using any encryption methods.

## Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.

TODO: Use subsections (e.g., 4.3.1 Reliability, 4.3.2 Portability, etc…) provide requirements related to the different software quality attributes. Base the information you include in these subsections on the material you have learned in the class. Make sure, that you do not just write “This software shall be maintainable…” Indicate how you plan to achieve it, etc.>

### Usability

The product will adhere to performance requirements. Furthermore, the site will be easily navigable, with large icons that stand out.

### Testability

The server and client will output logs containing important information (sent/received data, current state of the game)

# Other Requirements

<This section is **Optional.** Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

Appendix A – Data Dictionary

*<Data dictionary is used to track all the different variables, states and functional requirements that you described in your document. Make sure to include the complete list of all constants, state variables (and their possible states), inputs and outputs in a table. In the table, include the description of these items as well as all related operations and requirements.>*

Appendix B - Group Log

<Please include here all the minutes from your group meetings, your group activities, and any other relevant information that will assist the Teaching Assistant to determine the effort put forth to produce this document>

3:10 – 5:15, 11/5/20

1:30 – 2:00, 11/6/20