# SE 3XA3: Software Requirements Specification
# Wordle 2.0

Team 8, Goufs
Richard Fan, fanr13
Noel Zacharia, zacharin
Biranugan Pirabaharan, pirabahb

April 11, 2022

# Contents

# List of Tables

# List of Figures



Table 1: **Revision History**

| Date | Version | Notes |
|---|---|---|
| Feb 7, 2022 | 1.0 | Initial Document |
| Feb 9, 2022 | 1.1 | More sections added |
| Feb 10, 2022 | 1.2 | Use case diagram and final sections added |
| Feb 10, 2022 | 1.3 | Final revision and check |
| April 09, 2022 | 1.4 | Updates for Rev1 |

This document describes the requirements for the game Wordle 2.0. The template for this Software Requirements Specification is a subset of the Volere template[1].

# 1 Project Drivers

## 1.1 The Purpose of the Project

The purpose of this project is to recreate the functionality of the web game Wordle. Additionally, we aim to modernize the UI and add new features such as a dark mode and more levels.

## 1.2 The Stakeholders

### 1.2.1 The Client

The client is the professor for this course, Dr. Asghar Bokahri, and the teaching assistants, Veerash Palanichamy, Oluwaseun Owojaiye, and Abdul Rab Mohammed. The clients will provide us with deadlines on deliverables, and will have the final decision on the project's acceptance and adherence to the SRS.

### 1.2.2 The Customers

In this case, the customers are all people who will play the Wordle 2.0 game. This game targets people of a demographics and will have minimal hardware requirements to decrease the barriers to access.

### 1.2.3 Other Stakeholders

The developers of this project are also stakeholders. Without them, there would be no Wordle 2.0. The skills of individual team members, both technical and soft skills, will be put to use to complete this project. They will be responsible for the implementation, documentation, and testing of the product. Additionally, the original developers and contributors to the Wordle Clone repository will be stakeholders in this project as they are constantly updating the respository and wish to improve the game.

## 1.3 Mandated Constraints

### 1.3.1 Solution Constraints

**Description:** Wordle must be able to run on any modern browser or laptop.
**Rationale:** The potential users of Wordle will need to have the listed browsers.
**Fit Criterion:** Wordle will run on any latest version of node.

### 1.3.2 Implementation Environment of the Current System

*N/A*

### 1.3.3 Partner or Collaborative Applications

*N/A*

### 1.3.4 Off-the-Shelf Software

*N/A*

### 1.3.5 Anticipated Workplace Environment

*N/A*

### 1.3.6 Schedule Constraints

**Description:** The Project must follow the project schedule shown in the Development Plan
**Rationale:** The Project needs to follow a predefined plan to ensure the completion of the milestones by their respective due dates.
**Fit Criterion:** The Project will be completed with all milestones submitted on time.

### 1.3.7 Budget Constraints

*N/A*

## 1.4 Naming Conventions and Terminology

Table 2: Naming Conventions and Terminology

| Naming Conventions and Terminology | |
|---|---|
| **Term** | **Definition** |
| JavaScript(JS) | The programming language used in this project. |
| Player/User | The individual playing the game. |
| TypeScript | The programming language used in Not Wordle. |
| System | The software behind the game. |
| SRS | Acronym meaning Software Requirments Specification. This document describes what the software system is to do and its expected performance. |
| Web Browser | The platform on which the game will run. |
| User Interface | The point at which the user interacts with the system. |
| ~~React~~ | ~~The JS library that is used for building the user interface.~~ |

## 1.5 Relevant Facts and Assumptions

Facts:
The original repository is still being updated constantly. However, we are implementing and improving the version of the repository we found at the start of the term. This is to avoid having the original repository grow too large for the scope of our project.
Originally the repository did not have a dark theme, various word length game modes, or hard mode where all guesses must contain correct letters from previous guesses.
Assumptions:
Users will use a system with a web browser and have internet access.
Users will know how to operate a computer or phone.
Users have a moderate proficiency in English, at least a middle school level.

# 2 Functional Requirements

## 2.1 The Scope of the Work and the Product

### 2.1.1 The Context of the Work

Wordle 2.0 is a standalone application that doesn't interact with other folders or systems. It is meant to be played by one person at a time.

### 2.1.2 Work Partitioning

Table 3: Work Partitioning Events

| Event Number | Event Name | Input | Output |
|---|---|---|---|
| 1 | Starting a new game | Mouse | Final Score |
| 2 | Reading the instructions | Mouse | Instructions shown |
| 3 | Opening the settings menu | Mouse | Settings shown |
| 4 | Viewing about | Mouse | About |

Table 4: Work Partitioning Summaries

| Event Number | Summary |
|---|---|
| 1 | The user will use the mouse to start a new game. When the game ends, the user will see the hidden word and a summary of their guesses. |
| 2 | The user using the mouse, decides to read the instructions of Wordle 2.0. |
| 3 | During the game, the user can use the mouse to open the settings menu. |
| 4 | The user, through mouse input, views the about page. |

### 2.1.3 Individual Product Use Cases

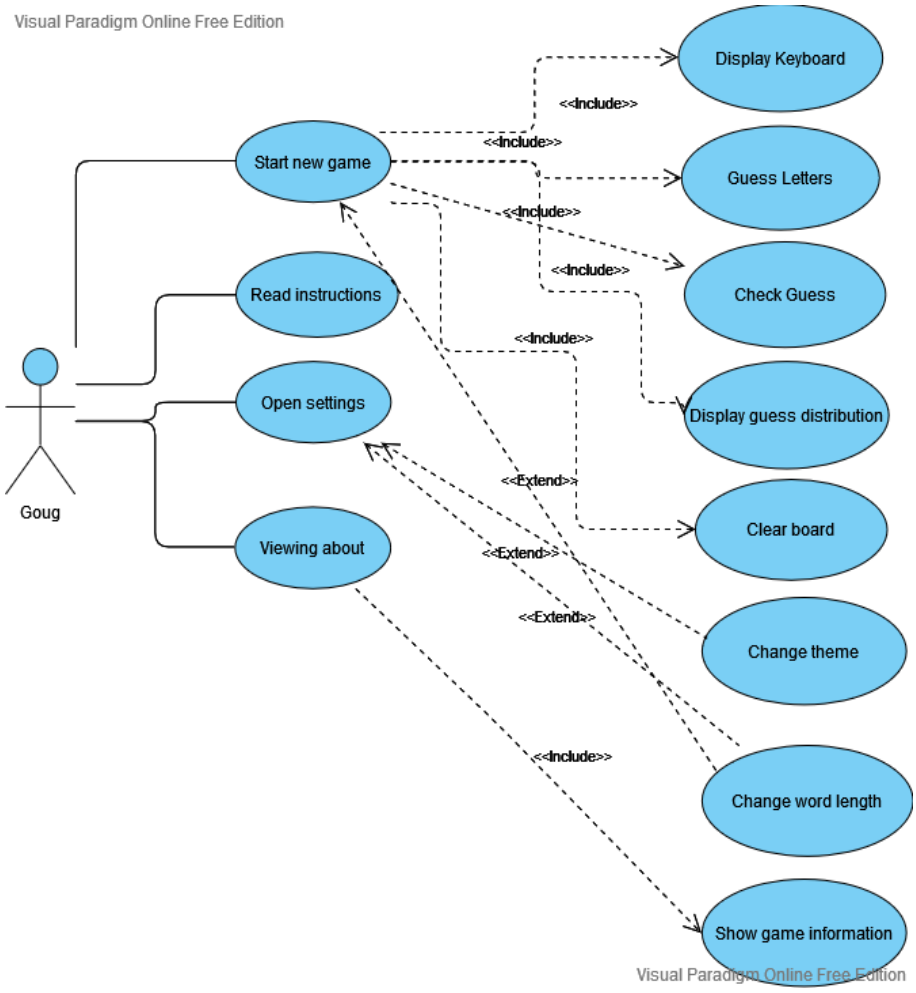


Figure 1: Use case diagram that displays the main functionalities of the application.

## 2.2 Functional Requirements

FR1. The system must display the letters in a keyboard layout, along with the enter and delete options.
FR2. The system must display the 6 rows for attempts, each having 5 boxes for the letters.
FR3. The system must have an option to change the theme of the page.
FR4. The system must have an option to display the rules of the game.
FR5. The system must have an option to display the player's statistics.
FR6. The system must display the player's total games played.
FR7. The system must display the success rate of the player.
FR8. The system must display the current streak of correct guesses made.
FR9. The system must display the player's best streak of correct guesses.
FR10. The system must display the player's guess distribution (distribution of the number of attempts a correct guess took).
FR11. The system must have a share option (copies the result of their last game to their clipboard as emojis)
FR12. The system must allow the user to make a guess
FR13. The system must display if a letter is in the correct position in the target word.
FR14. The system must display if a letter is in the target word but not in the correct position.
FR15. The system must display if a letter is not in the target word.
FR16. The system must alert the player if their guess is not a valid word.
FR17. The system must alert the player if their guess does not contain enough letters.
FR18. The system must update the displayed keyboard to reflect the accuracy of the letters guessed.
FR19. The system must prohibit the user from changing their guess after they submit it.
FR20. The system must have an option to the game mode (word length to either 4 or 6).
FR21. The system must update the display to reflect the game mode.
FR22. The system must give an option to play again, giving the player a new target word and resetting the display to its initial state.

# 3 Non-functional Requirements

## 3.1 Look and Feel Requirements

### 3.1.1 Appearance Requirements

LF1 The UI must be easy to understand and use.

### 3.1.2 Style Requirements

LF2 The UI theme must be inspired by wordle's UI.

## 3.2 Usability and Humanity Requirements

UH1 The game must be understood by users between the ages of $MIN\_AGE$ and $MAX\_AGE$.

UH2 The game must be playable with the use of only one hand.

UH3 The game must provide themes to best suit the user's environment.

UH4 The game must be playable with no background knowledge.

UH5 The game must have two methods of input(keyboard/mouse) to best suit the user's preferences.

## 3.3 Performance Requirements

P1 The system must update the display quickly (within $MIN\_TIME$ seconds) whenever the user has input.

P2 The system must load all visuals on startup within $MIN\_TIME$ seconds.

P3 The system must update the statistics within $MIN\_TIME$ seconds after the last game has ended.

## 3.4 Operational and Environmental Requirements

### 3.4.1 Expected Physical Environment

OR1 The system must not require an internet connection nor should it require high-performance graphic cards.

### 3.4.2 Requirements for Interfacing with Adjacent Systems

OR2 The program should not access and make changes to any file outside its own folder

### 3.4.3 Productization Requirements

~~OR3 The game must be distributed as a zip folder and its total size must be below~~ $MAX\_STORAGE$.

### 3.4.4 Release Requirements

OR4 The product will have a final release date corresponding to the last day of classes.

## 3.5 Maintainability and Support Requirements

M1 The code must be documented through the use of comments and

M2 The code must follow the style agreed upon in the Development Plan.

M3 Any changes will be documented.

## 3.6 Security Requirements

SR1. The user must not be able to view the statistics of other players.

## 3.7 Cultural Requirements

### 3.7.1 Cultural Requirements

CR1 The game can't suggest offensive words or non-English words.

### 3.7.2 Political Requirements

CR 2 The game can't suggest politically charged words.

## 3.8 Legal Requirements

LR1 The program must not break any digital privacy laws within the countries of Canada and the United States of America.

## 3.9 Health and Safety Requirements

HS1 The project will not include any form of high-frequency flashes, contrast alteration, or brightness variation which may cause epileptic seizures for some users.

HS2 ~~To prevent repetitive stress injury and addiction, a warning will be displayed by the system encouraging users to take regular breaks.~~

# 4 Project Issues

## 4.1 Open Issues

We have not decided on a testing framework as no Javascript framework is considered the best solution according to the programming community. Currently leaning towards using JEST or Cypress, but still exploring the limitations of each.
We are also not proficient with TypeScript, thus it is uncertain how difficult it will be to modularize the source code as it is written in TypeScript.

## 4.2 Off-the-Shelf Solutions

Wordle is the original version of this game. However, its source code is not published. There are few public solutions specific to the creation of Wordle 2.0; any code adapted for the project will come from Not Wordle.

### 4.2.1 Ready-Made Products

Wordle Clones exist in the market. One such example is the source repository.

### 4.2.2 Reusable Components

The source repository has resusable components that can be replicated on a need-to-need basis.

### 4.2.3 Products That Can Be Copied

The source repository has an MIT licence and thus redistribution or replication of the product is allowed.

## 4.3 New Problems

## 4.4 Tasks

Tasks are assigned and specified using a Gantt Chart. The chart is maintained and updated through the lifetime of the product's development.

## 4.5 Migration to the New Product

N/A. This product will remain independent of the source code.

## 4.6 Risks

There is little to no risk associated with this project. The program is designed to run within a web browser which has very little interaction with other processes running within the user's system. Any risks stemming from the program itself may occur in the form of crashes and ppoor performance. Test driven development will be done to minimize these risks.

## 4.7 Costs

The development of the project will have no cost. Under the MIT licence, the source project is allowed to be modified and distributed freely. The applications used for development are also free. An estimated time cost for development is 6 hours a week for all developers.

## 4.8 User Documentation and Training

### 4.8.1 User Documentation Requirements

Instructions will be included as an option in the game's main menu. The game's directory will contain a README file providing installation instructions.

### 4.8.2 Training Requirements

No specific training is necessary to play Wordle 2.0. Controls should be intuitive.

## 4.9 Waiting Room

Should there be additional time, the following low priority features will be considered and implemented:

1. Different world lengths. The user would be able to change the gameplay mode to include words that are not 5 letters in length
2. Allow for multiplayer functionality.

## 4.10 Ideas for Solutions

N/A.

# References

[1] J. Robertson and S. Robertson, *Volere Requirements Specification Template*, 16 ed., 2012.

# 5 Appendix

## 5.1 Symbolic Parameters

$MIN\_AGE = 10$
$MAX\_AGE = 90$
$MAX\_STORAGE = 20$
$MIN\_TIME = 0.25$