# CMSC 447

### Software Design and Development

# Fall 2022

# Wordle Coach System Requirements Specification

Wordle Coach

System Requirements Specification

#### Table of Contents

Page

## 1. Introduction

* 1. Purpose of This Document 3
  2. References 3
  3. Purpose of the Product 4
  4. Product Scope 4

2. Functional Requirements 5

3. Non-Functional Requirements 9

3.1 Customer Constraints 9

3.2 External Interfaces 9

3.3 Other 9

4. Deliverables 10

5. Open Issues 10

## Appendix A – Agreement Between Customer and Contractor 11

Appendix B – Team Review Sign-off 12

Appendix C – Document Contributions 13

## Document Versioning Control

|  |  |  |
| --- | --- | --- |
| **Version Number** | **Date** | **Changes from Previous Version** |
| 1.0 | 11/11/2022 | N/A |

## 1. Introduction

1.1 Purpose of This Document

This document will discuss the requirements for the Wordle Coach application and is intended for the development team of this product. This document will outline the requirements that the development team will follow during development of the app. The intended reader for this document is the dev team, team leadership and management, and technical developer in the future looking to update, improve, or revise the app.

On top of listing the specified requirements given by both the customer and the development team this document will provide other important information. This document will discuss the purpose of this product, the scope of this project, as well as constraints. The constraints are given by the customer as well as other external ones. There will be references for any documents relating to constraints or requirements given by the customer. A section of this document will then discuss the functional requirements.

1.2 References

Provide a list of all applicable and referenced documents and other media (e.g., the Sommerville text, UML references, documents provided by the customer, documents provided by your instructor, websites) that were used in the creation of this document. See the Writing Resources on Blackboard for the appropriate formats for references.

Wardle, Josh. "Wordle - A Daily Word Game." The New York Times - Breaking News, US News, World News and Videos, New York Times, 2022, www.nytimes.com/games/wordle/index.html. Accessed 20 Sept. 2022.

dracos. “valid-wordle-words.txt.” *GitHub*, 4 Jan. 2022, <https://gist.github.com/dracos/dd0668f281e685bad51479e5acaadb93>. Accessed 11 Nov. 2022.

1.3 Purpose of the Product

This Wordle Coach project is designed to give assistance to a user playing Wordle with the intent for the user to become better at the game. At the time of writing this, Wordle is a very popular game that people of all ages play. Seeing that this game can be difficult for some users, this application will aid those players. If a user is stuck while playing Wordle, they will be able to type in their game board into this app. The application will then give the user suggested words and the probabilities of those words being correct. Considering that the correct answer to the Wordle of the day is not known to the public, this application will not be able to provide the certain correct answer and not allow the user to cheat but rather help the user make a more educated guess. With these suggestions, the user will be able to learn techniques and see similarities in words in hopes of performing better on the Wordle by themselves.

* 1. Product Scope

Android Phone/Tablet App

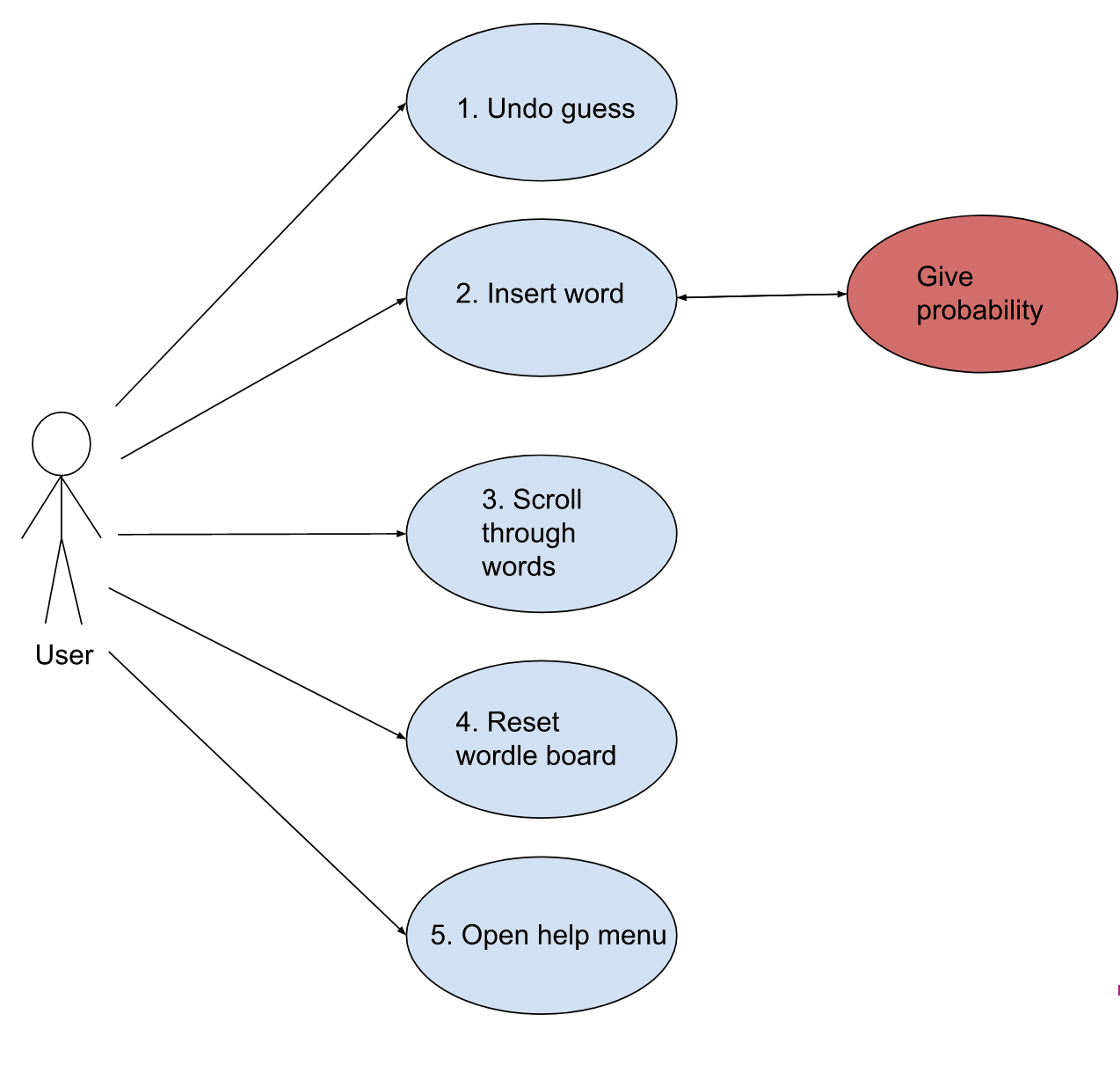


Figure 1: This is the use case diagram for the Wordle Coach which shows the functions and the primary user’s relationship.

The five main functions that a user can perform are to undo a guess, insert a word, scroll through the word list, reset the wordle board, and open the help menu. Only inserting a word will communicate with the back-end system. The back end of this system refers to leaving the front-end user interface portion of the code and calling the library containing the wordle coach algorithm. The back end and front end will communicate bidirectionally with the back end receiving a word. The back end will then send back words and their associated probabilities. The only input to this system is from a user, there will be no admin roles. There will also not be any external systems in the use case.

2. **Functional Requirements**

The following functional requirements are derived from Figure 1

* The app shall take in a five-character word guess and allow the user to specify the color of each character (green, yellow, gray) and then display wordle words with their associated of probability of being the wordle.
* The app shall alert the user if they have not entered 5 characters for their guess and shall return to a state as if the invalid guess had not been made.
* The app shall allow the user to start over and reset their progress in the current game, and will prompt the user to confirm the action before starting over
* The app shall allow the user to undo their previous guess.
* The app shall display a help screen containing information on how to use the app.

|  |  |  |
| --- | --- | --- |
| **Number** | 1 | |
| Name | Enter Word | |
| **Summary** | Enter a word and its associated letter colors to get wordle words likely to be the wordle of the day | |
| **Priority** | 5 | |
| **Preconditions** | The app is running, there is not a popup present on the screen and less than 6 guesses have been made | |
| **Postconditions** | A word has been entered, the number of previous guess have been incremented, and likely wordle words have been presented | |
| **Primary Actor(s)** | User/Player | |
| **Secondary Actor(s)** | N/A | |
| **Trigger** | The app is running, and the user enters a guess then clicks enter guess | |
| **Play Wordle** | **Step** | **Action** |
|  | 1 | The user puts a letter in the first character box |
|  | 2 | The user selects the correct color for the first character |
|  | 3 | The user puts a letter in the second character box |
|  | 4 | The user selects the correct color for the second character |
|  | 5 | The user puts a letter in the third character box |
|  | 6 | The user selects the correct color for the third character |
|  | 7 | The user puts a letter in the fourth character box |
|  | 8 | The user selects the correct color for the fourth character |
|  | 9 | The user puts a letter in the fifth character box |
|  | 10 | The user selects the correct color for the fifth character |
|  | 11 | The user clicks enter guess |
|  | 12 | The app displays the words most likely to be the wordle of the day based off the previous guesses |

|  |  |  |
| --- | --- | --- |
| **Number** | 2 | |
| Name | Reject guess | |
| **Summary** | The app rejects the users guess because they did not enter enough characters | |
| **Priority** | 4 | |
| **Preconditions** | The app is running and less then 6 guesses have been made | |
| **Postconditions** | The bad guess is discarded and the app returns to a state as if it had not been made | |
| **Primary Actor(s)** | User Player | |
| **Secondary Actor(s)** | N/A | |
| **Trigger** | The user has entered less than 5 characters and clicked enter guess | |
| **Main Scenario** | **Step** | **Action** |
|  | 1 | The user enters less than 5 characters could or could not have selected the colors for those letters |
|  | 2 | The user clicks enter guess |
|  | 3 | The app displays an error message telling the user to enter guesses with five characters |
|  | 4 | The user dismisses the message |
|  | 5 | The app returns to a state as if the bad guess had not been entered |

|  |  |  |
| --- | --- | --- |
| **Number** | 3 | |
| Name | Start Over | |
| **Summary** | The app will simply restart and allow the user to start all over again after the user accepts the confirmation prompt | |
| **Priority** | 4 | |
| **Preconditions** | The app is running | |
| **Postconditions** | The app clears the entire board, and it will run like the user is attempting to solve the wordle for the day | |
| **Primary Actor(s)** | User/Player | |
| **Secondary Actor(s)** | N/A | |
| **Trigger** | The user has clicked on the restart button and confirm the prompt asking about restarting the wordle | |
| Main Scenario | Step | Action |
|  | 1 | The user has clicked on the start over button |
| 2 | The app will prompt the user confirming that they want to proceed with this action |
| 3 | The user accepts the confirmation message |
|  | 4 | The wordle board becomes empty and the user will have all their guesses available |
| Extensions | Step | Branching Action |
|  | 1 | If the user simply denies the prompt, nothing will happen |

|  |  |  |
| --- | --- | --- |
| Number | 4 | |
| Name | Undo | |
| Summary | The app will simply delete the user’s last guess and let them attempt that guess again after the user accepts the confirmation prompt | |
| Priority | 4 | |
| Preconditions | The app is running and at least one guess has been made | |
| Postconditions | The app simply clears the last guess and lets the user attempt that guess again | |
| Primary Actor(s) | User/Player | |
| Secondary Actor(s) | N/A | |
| Trigger | The user has clicked on the undo last guess button while the app is running after at least one guess is made | |
| Main Scenario | Step | Action |
|  | 1 | The user has entered at least one guess for the wordle |
|  | 2 | The user has clicked on the undo last guess button |
|  | 3 | The app will prompt the user confirming that they want to proceed with this action |
|  | 4 | The user accepts the confirmation message |
|  | 5 | The wordle board will delete the last guess entered and restart from that point |
| Extensions | Step | Branching action |
|  | 1 | If the user simply denies the prompt, nothing will happen |

|  |  |  |
| --- | --- | --- |
| Number | 5 | |
| Name | Open Help Menu | |
| Summary | The app will open a pop up window that tells the user how the Wordle Coach app works. | |
| Priority | 3 | |
| Preconditions | The app is running, likely without any guesses having been made yet | |
| Postconditions | The app is running, at any amount of guesses having been made | |
| Primary Actor(s) | User/Player | |
| Secondary Actor(s) | N/A | |
| Trigger | The user has clicked on the help button while the app is running | |
| Main Scenario | Step | Action |
|  | 1 | The user presses the help button |
|  | 2 | The app will open up a pop up window that explains how the app works |
|  | 3 | The user will read the instructions to understand how the app works |
|  | 4 | The user will then close the pop up window when they are finished reading |
|  | 5 | The app will return to the screen that it was on before the help button was pressed |

(This template was adapted from *Basic Use Case Template***,** byAlistair Cockburn, http://members.aol.com/acockburn/papers/uctempla.htm, accessed 1/17/08.)

3. Non**-Functional Requirements**

* NFR 1 (Priority 4) - Making sure that there are no copyright issues with Wordle since they do have the copyright to the name. The priority for this will be a 4 since the New York Times can take our product, causing us to lose the right to the product and possibly be sued for it, which can cost a lot.
* NFR 2 (Priority 3) - Help button functionality so that the user can learn how to use the app and effectively get better at playing Wordle when adapting to the choices that the app will provide along with tips and tricks. The priority for this will be a 3 since this function will be helpful for the user and it makes sure that the user is able to use the app without any confusion.

3.1 Customer Constraints

The system shall be an Android application. The system shall run on all Android devices that can run currently supported operating systems (OS 10-13). The customer has specified no constraints with respect to their own devices.

3.2 External Interfaces

We read from one data file – the Wordle acceptable word list, which is a .txt file. Each word in the file is on its own line and is a string of length 5 that we parse through in our algorithm to generate the list of probable next guesses.

3.3 Other

We do not have any other non-functional requirements.

4. **Deliverables**

|  |  |  |
| --- | --- | --- |
| Deliverable | Format | Delivery Date |
| Source Code | .zip file and github repo | 12/15/2022 |
| Executable | .apk file | 12/15/2022 |
| App System Requirement Specification Document | .docx or .doc file | 12/15/2022 |
| App Design Document | .docx or .doc file | 12/15/2022 |
| UI Design Document | .docx or .doc file | 12/15/2022 |
| Development Schedule | .pdf and Google Calendar | 12/15/2022 |
| README.md | Included in source .zip |  |

5. **Open Issues**

11/3: Opening the android keyboard irreversibly glitches the UI layout and the app must be restarted. Resolution target: 11/7

**Appendix A – Agreement Between Customer and Contractor**

When the customer and development team sign this document, they assert that the contents of this document have been agreed upon. When the customer signs they acknowledge that the requirements contained herein satisfy their business requirements and agree to pay for the software developed in accordance with these requirements. When the development team signs this document they agree to develop software that satisfies these requirements. They also agree to develop and deliver all deliverables. If the document needs to be changed, the members of the development team involved in the changes must add a version number, the current date, and a description of their changes to the version control table. They must also inform all other development team members of their changes.

**Appendix B – Team Review Sign-off**

The below signatures certify that all group members have reviewed this document and agree on the contents and the format contained herein. Note, there are not any points of disagreement.

Signature: Date:

Zan Wills 11/18/2022

Jamie Kirk 11/17/2022

Dennis Mayher 11/15/2022

Parth Patel 11/7/2022

Nathan Hoernlein 11/7/2022

**Appendix C – Document Contributions**

|  |  |
| --- | --- |
| Team Member | Role |
| Nathan Hoernlein | Worked on functional requirements, customer constraints, external interfaces, other |
| Zan Wills | Worked on functional requirements, use case specifications, document agreement, document purpose and intended readership |
| Parth Patel | Worked on non-functional requirements and references |
| Dennis Mayher | Worked on non-functional requirements, use case specifications, top level use case diagram |
| Jamie Kirk | Worked on external interfaces, references |