|  |
| --- |
| **How to Use the SDP Notebook Template**   1. Students will use this template throughout the SDP to document their project development. 2. A screenshot of a computer     Description automatically generatedFollow the instructions in the comments by enabling “All Markup” in the “Review” tab on the top ribbon of the Microsoft Word document. 3. Delete all **comments** and this **front page** before submitting the final version of this Notebook |

Group E

Matt Ruth

Albert Liu

Matthew Zhao

Vik Yadav

ENGR 1181.01 - Fund Engr 1 (6449)

Elizabeth Jurgens

12/4/24

# Table of Contents

[Table of Contents 3](#_Toc180580151)

[Executive Summary 4](#_Toc180580152)

[Project Management Documentation 5](#_Toc180580153)

[1. Team Working Agreement 5](#_Toc180580154)

[2. Individual Responsibility Agreement 5](#_Toc180580155)

[Programming Responsibilities/ Roles 5](#_Toc180580156)

[3. Project Schedule 5](#_Toc180580157)

[4. Meeting Notes 5](#_Toc180580158)

[a. Meeting 1 5](#_Toc180580159)

[b. Meeting 2 5](#_Toc180580160)

[c. Meeting 3 5](#_Toc180580161)

[Business Plan 6](#_Toc180580162)

[1. User Identification and Interviews 6](#_Toc180580163)

[a. User Interview 1 6](#_Toc180580164)

[b. User Interview 2 6](#_Toc180580165)

[2. Electronic Advertisement 6](#_Toc180580166)

[3. Pitch Video Link 6](#_Toc180580167)

[Software Documentation 7](#_Toc180580168)

[1. Introduction 7](#_Toc180580169)

[2. User Manual 7](#_Toc180580170)

[3. Program Description for Developers 7](#_Toc180580171)

[4. Final Algorithm or Flowchart or Pseudocode 7](#_Toc180580172)

[5. Final Program with Comments 7](#_Toc180580173)

[6. Brief Discussion 7](#_Toc180580174)

[7. Conclusion and Recommendation 7](#_Toc180580175)

[References 8](#_Toc180580176)

# Executive Summary

In this paper is the documentation to the creation of Wordle in MATLAB. This paper walks through the process of developing Wordle with 4 different team members. First the group members are assigned different parts of the project to work on. Then the project schedule is created, detailing the deadline and progress of the project. A detailed log of 3 assigned meetings is also included, where the team members discuss their progress and potential adjustments to the code. A business plan is included to market the game, consisting of interviews from a test audience, a promotional flyer, and a pitch video. Next is the documentation of creating the software. This documentation includes a user manual to help code the game as well as obstacles that the team encountered while creating Wordle. Lastly, there is a recommendation for future additions and suggestions to the game that could be implemented if given more time on the project.

# Project Management Documentation

## Team Working Agreement

**Team Working Agreement**

Term (Example: **Autumn 2024**)

**Creation 08/21/2024**

### 1) Group Identification

Lab section # (see Carmen) - AU24 ENGR 1181.01 (6449)

Table Letter - E

Instructor (GTA) - Ulfat Tahsin (tahsin.4@osu.edu)

Team member info:

|  |  |  |  |
| --- | --- | --- | --- |
| **NAME:** | **EMAIL:** | **PHONE:** | **OTHER:** |
| Albert Liu | liu.11591@osu.edu | (207) 344-7180 |  |
| Matthew | zhao.4474@osu.edu | (380) 249-5353 |  |
| Matt Ruth | ruth.166@osu.edu | (630) 427-5760 |  |
| Vik | yadav.187@osu.edu | (513) 307-2421 |  |
|  |  |  |  |

### 2) Primary Means of Communication and Expectations

State your team’s agreed upon means of communication and expectations for response.

The group chat software uses “Snapchat”, the meeting is mainly held through online Zoom, and if offline work is needed, the meeting time is communicated through Snapchat. In most cases, all group members can reply as soon as they see the message, and they must check the group chat once every two days to make sure the message is not missed.

### 3) Scheduling of Meetings

**Day and time of regular meeting:** **Choose half an hour every Sunday between 4pm and 7pm.**

**Location/format of meeting: Zoom or In-person**

**Agreed upon means of scheduling other meetings: “Snapchat”**

(Example: Group will meet every Thursday from 6:30 to 7:00 over Zoom. Before leaving at the end of each class, the team will agree to set any meetings times and agenda needed before the next class and beyond if possible. The Coordinator for that week will send out a reminder of the meeting with an agenda within eight hours after the class.) Example Agenda Format:

Team Name:

Meeting Schedule:

|  |  |  |
| --- | --- | --- |
| Date: 09/03/24 | Time: 7 pm | Location: Zoom |

Participating members (If not all.):

Agenda: (items in bulleted/numbered list)

### 4) General Responsibilities for All Team Members

This element of the team working agreement is the list of rules/agreements or the contract that all members agree to abide by.

Everyone should do their own work, upload their own work files to Snapchat in time to facilitate other team members to complete their homework. Actively communicate and seek solutions to any problems or situations.

### 5) Specific Team Member Roles

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Lab** | **Coordinator** | **Recorder** | **Checker** | **Monitor** |
| Team Working  Agreement | Math | Albert | Matthew | Vik |
| Technical  Communications Lab | Maintained  (tentative) | Maintained  (tentative) | Maintained  (tentative) | Maintained  (tentative) |
| Application 1 | Maintained  (tentative) | Maintained  (tentative) | Maintained  (tentative) | Maintained  (tentative) |
| Beam Bending Lab | Maintained  (tentative) | Maintained  (tentative) | Maintained  (tentative) | Maintained  (tentative) |
| Humanitarian Relief  Lab | Maintained  (tentative) | Maintained  (tentative) | Maintained  (tentative) | Maintained  (tentative) |
| Wind Turbine Lab | Maintained  (tentative) | Maintained  (tentative) | Maintained  (tentative) | Maintained  (tentative) |

**Note: A student may have more than one role, or two students may have the same role depending on the size of the team**

### 6) Conflict Resolution

Each team should have a pre-agreed approach to addressing issues that may arise.

Example statement: When there are problems within our group pertaining to the general responsibilities or specific responsibilities, the following steps will be taken in this order until a resolution is found.

1. Enter conflict resolution assuming good intentions from teammates.
2. Discuss problems within the group to come to a working solution, trying to make sure all members have opportunity to participate in the discussion.
3. Hold a team discussion of the problem with a GTA or Professor to find a solution.
4. If 2 and 3 fail, ask a GTA or Professor to assist (arbitrate).

### 7) Expectations of Faculty and GTA’s Suggested Statement:

If a team member fails to live up to this agreement, the situation may be reported to the staff, but the team will still be responsible for submitting a completed assignment. Staff will be available to meet with teams to resolve issues.

### 8) Team Signatures

Signature for each student is to be included.

Example: A close up of a letter

Description automatically generated

Brutus Buckeye

Albert Liu: A close up of a name

Description automatically generated

Matthew: A close up of a paper

Description automatically generated

Matt: A close up of a note

Description automatically generated

Vik:A close up of a black background

Description automatically generated

## Individual Responsibility Agreement

### Programming Responsibilities/ Roles

|  |  |  |
| --- | --- | --- |
| **Team Member** | **Initial Assigned Tasks** | **End Result** |
| Albert Liu | Coding, Parts of documents edit | Finished |
| Matthew Zhao | Create sprites, debug word detection system, create numerical system to assign numbers to letters | Finished |

Table 1. Programming Responsibilities/ Roles

**Documentation Responsibilities/ Roles**

|  |  |  |
| --- | --- | --- |
| **Team Member** | **Initial Assigned Tasks** | **End Result** |
| Matt Ruth | Document meetings, import sprites, create starting screen | Finished |
| Vik Yadav | Debug code, import word list, check if each individual letter is correct and whole word | Finished |

Table 2. Documentation Responsibilities/ Roles

## Project Schedule

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Task** | **Start Date** | **Finish Date** | **Due Date** | **Primary Person** | **Secondary Person** | **Est. Hours** | **% Completed** |
| **Word and Letter Detection System** | **11/11** | **11/20** | **11/20** | **Albert Liu** | **Matt**  **Ruth** | **12** | **100** |
| **Debugging** | **11/11** | **11/20** | **11/20** | **Matthew**  **Zhao** | **Vik Yadav** | **5** | **100** |
| **Sprite Creation and Importation** | **11/4** | **11/11** | **11/20** | **Matt Ruth** | **Matthew**  **Zhao** | **3** | **100** |
| **Word List Importation** | **11/4** | **11/11** | **11/20** | **Vik Yadav** | **Albert Liu** | **2** | **100** |

Table 3. Project Schedule

## Meeting Notes

### Meeting 1

Date/Time: 11/08/24 10:30 am

Members Present: Matthew Zhao, Albert Liu, Matt Ruth, Vik Yadav

Topics/ Agenda: Establishing an understanding of what needs to be done, assigning roles, start working on SDP assignment

### Meeting 2

Date/Time: 11/13/24 10:30 am

Members Present: Matthew Zhao, Albert Liu, Matt Ruth, Vik Yadav

Topics/Agenda: Reviewing each other’s work, getting recommendations from other team members, integrating each other’s work into one script file, improving code, implementing new ideas that were previously forgotten

### Meeting 3

Date/Time: 11/20/24 10:30 am

Members Present: Matthew Zhao, Albert Liu, Matt Ruth, Vik Yadav

Topics/ Agenda: Reviewing final code, testing code to make sure it runs smooth and seamlessly

# Business Plan

## User Identification and Interviews

Intended Audience: Students

### Sam Gordon

Sample set of questions:

1. What are some games you have played recently?

Monster Hunter Rise, ICBM Escalation, Europa Universalis 4, Bloons TD 6

1. What are some games you might enjoy playing in MATLAB?

Clash Royale, Temple Run, Subway Surfers

1. In your opinion, what is the optimal length a game should take to play?

3-5 minutes

1. What aspects of a game make it enjoyable to play?

Rewarding gameplay loop, the ability to progress, good visuals

1. Based on playing this game, what changes would you recommend?

Make is so that the game only accepts words in the word list instead of any combination of five letters

### Will Barnett

1. What are some games you have played recently?

Ncaa 25, rim world, online euchre, DayZ, pool, poker, basketball,

1. What are some games you might enjoy playing in MATLAB?

amaze, chess, battleship, checkers, pool, any game pigeon game

1. In your opinion, what is the optimal length a game should take to play?

depends on the game, under ten minutes

1. When playing these games, which aspects did you enjoy?

Replayability, difficulty, easy to navigate,

1. Based on playing this game, what changes would you recommend?

Add a delete button

## Electronic Advertisement

[Turquoise and Orange Retro Illustration Now Hiring Flyer.pdf](file:///C:\Users\mattc\Downloads\Turquoise%20and%20Orange%20Retro%20Illustration%20Now%20Hiring%20Flyer.pdf)

A poster with text and pictures on it

Description automatically generated

## Pitch Video Link

# Software Documentation

## Introduction Welcome to Wordle! Here we will show you how we coded this game!

## User Manual ~Simple Wordle Game---Enter 1 to start game, 0 to finish game~

## Program Description for Developers

1.Game Initialization: The game uses the “simpleGameEngine” function to render the graphical interface. Color offsets are defined to represent correct letters (green), misplaced letters (yellow), incorrect letters (red), and default blank spaces (gray).

2. User Interaction: Players interact with the game via keyboard inputs: ‘1’ to state the game or play again, ‘0’ quit the game, ‘2’ delete the last entered letter during guessing. The game provides immediate feedback for each guess and prompts for replay upon completion.

3. Core Logic

Word Selection: A random word is chosen from an external file (words.txt) containing valid 5-letter words. Guess Evaluation: Each guessed word is compared with the target word: Correct letters in the correct position are marked green; Correct letters in the wrong position are marked yellow; Incorrect letters are marked red. Validation: The program checks if the guessed word exists in the database before processing.

4. Graphical Interface

The game board dynamically updates to reflect the player's guesses and the corresponding color-coded feedback. Custom messages are displayed to guide the user through gameplay, errors, and results.

5. Game End and Replay

If the player guesses correctly or exhausts their attempts, the game concludes with appropriate feedback. A replay prompt allows users to start a new game or exit.

6. End Menu

A "Thank You for Playing" message is displayed when the user quits the game

## Final Algorithm or Flowchart or Pseudocode

A diagram of a flowchart

Description automatically generated

## Final Program with Comments

[Wordle.pdf](file:///C:\Users\mattc\OneDrive\Engineering%201181.01\Wordle.pdf)

A screenshot of a computer program

Description automatically generatedA screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generatedA screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generatedA screenshot of a computer program

Description automatically generatedA screenshot of a computer program

Description automatically generated

## Brief Discussion

This MATLAB Wordle game is a creative implementation of a classic word-guessing puzzle that leverages MATLAB's graphical capabilities to provide an interactive user experience. The code demonstrates effective use of modular programming principles, clear logic flow, and robust input validation. The graphical feedback mechanism enhances the gameplay, providing intuitive color-coded hints that align with traditional Wordle rules. However, there are a few limitations:

1. The word database only dependency on words.txt, the game's functionality is reliant on the external word database file. If the file is missing or contains inappropriate words, the game will not function correctly.
2. Limited customization: The game is fixed to 5-letter words and a maximum of six guesses, with no options to adjust difficulty levels or word lengths.
3. Error Handling: The code could be improved by handling edge cases, such as an empty or corrupted word file, to make it more robust.

## Conclusion and Recommendation

Conclusion: The MATLAB Wordle game is a well-designed project that effectively demonstrates the use of graphical and logical programming in MATLAB. It combines user-friendly features, such as keyboard interactivity and dynamic feedback, with the challenge of solving a word puzzle. The modular structure ensures maintainability, while the straightforward gameplay guarantees accessibility for all users.

Recommendations to enhance the game's functionality and appeal: improve error handling, add customization options, enhance graphics, add scoring system (with infinitive mode), word verification, add multiplayer mode (Racing mode).

# References

**Wordle game in web:** [**https://powerlanguage-wordle.github.io/**](https://powerlanguage-wordle.github.io/)

**Wordle’s background:** [**https://en.wikipedia.org/wiki/Wordle**](https://en.wikipedia.org/wiki/Wordle)