Project Plan

VocabVersus[[1]](#footnote-1)

Thomas van der Molen

|  |  |
| --- | --- |
| **Project Information** | |
| Project members | Thomas van der Molen |
| Project Name | VocabVersus |
| Version | 1.4 |

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Change** |
| 1.0 | 12-02-2023 | Created Document |
| 1.1 | 16-02-2023 | Changed Project |
| 1.2 | 02-03-2023 | Added Architecture |
| 1.3 | 03-03-2023 | Added Dashboard Technology |
| 1.4 | 16-03-2023 | Changed Document Ordering |
| 1.5 | 05-04-2023 | Moved architecture to separate document |

Table of Contents

[Prerequisite 3](#_Toc131610369)

[Functional Learning Outcomes 3](#_Toc131610370)

[Introduction 4](#_Toc131610371)

[Similar Contexts 4](#_Toc131610372)

[Target Audience 4](#_Toc131610373)

[Functional Components 4](#_Toc131610374)

[User Stories 4](#_Toc131610375)

# Prerequisite

While the project is done for educational purposes, it is also meant to prove the learning outcomes set by Fontys for Advanced Software semester 6.

The setup and execution of this project will be heavily driven by the learning outcomes set by Fontys.

## Functional Learning Outcomes

For Advanced Software semester 6, there are 9 Learning Outcomes, below is a list of the learning outcomes that can be functionally proven in the process of this project:

**Learning outcome 1 – Future-oriented Organization**

Develop and deploy scalable software in accordance to the project goals, and design such solution with the ability for future further development.

**Learning outcome 2 – Investigative Problem-Solving**

Deliver professional products based on a structured problem-solving and methodical planning in a critical/professional manner.

**Learning outcome 5 – Scalable Architectures**

Develop architecture of scalable software, considering attributes related to enterprise contexts with high volume data and/or events.

The architecture should also be future expandable and allow for independent monitoring and deployment.

**Learning outcome 6 – Development and Operations (DevOps)**

Set up software development environments allowing for as much automation as possible, enabling short release times and high software quality.

**Learning outcome 7 – Cloud Services**

Integrate cloud services and deploy (parts of) an application to a cloud platform.

**Learning outcome 8 – Security by Design**

Incorporate best practices and minimize security risks.

**Learning outcome 9 – Distributed Data**

Consider legal and ethical issues alongside specific data requirements for enterprise systems.

# Introduction

VocabVersus is a web-based multiplayer vocabulary game, in this game multiple players will compete to think of a word containing given letters. Points will be given based on player’s speed and complexity of the word given.

User’s will have the ability to play in a competitive environment to try and become the top player on the leaderboards, or create their own game types (word lists, game rules) and play for fun.

The game’s point distribution will change dynamically based on the habits of players, for example: Words used very often will become worth less points.

## Similar Contexts

Similar games based on creating/guessing words already exist such as [Scrabble](https://playscrabble.com/) or [Wordle](https://www.nytimes.com/games/wordle/index.html) and games such as [scribble.io](https://skribbl.io/) is similar in its functionality/behavior.

## Target Audience

The target audience will be expected to be people playing for a short period (~1 game) or a group of friends wanting to play a simple party game.

To make the game as approachable as possible for these two major groups, the design of the web-application should make it deliberately easy to join a game or to create your own and invite people, along with supporting many different screen-sizes (e.g. phone, tablet, monitor) .

## Functional Components

|  |
| --- |
| **Component** |
| Webpage(s) allowing users to navigate to the different parts of the game |
| Webpage used by a player during the game |
| Game system for receiving game input and returning game information |
| Player connection system handling the communication between players and game instances |
| Word system for handling logic related to handling the letter-word functionality |
| Account system for allowing users to create and authenticate player accounts |
| Deployment pipeline for automated CI/CD |

## User Stories

|  |
| --- |
| **Story** |
| As a player, I want to be able to start a game quickly and easily. |
| As a player, I want to be able to easily join a created game via a link. |
| As a player, I want to customize my game with different word-sets. |
| As a player, I want to be able to create my own word-sets to use. |
| As a player, I want the ability to change game rules when creating a game. |
| As a competitive player, I want the game to be balanced as to reward the use of complex or less common words. |
| As a player group, we want to be able to manage who joins our game. |
| As a player, I want to be able to play the game from many different devices (pc, phone, tablet). |
| As a player, I want to see my score compared to the other players in the game. |
| As a player, I want to have visual feedback on the current state of a game. |
| As a player, I want to clearly see the requirements of for the word to submit. |
| As a player, I want to easily submit word attempts. |
| As a player, I want visual feedback why a word is incorrect |

1. Subject to change [↑](#footnote-ref-1)