

## 4.1 SDP Plan Introduction

This Software Development Plan provides the information regarding the schedule/plan for the development of Keyboard Warrior, which provides an application to help users improve their typing skills in the form of a video game.

Keyboard Warrior is a typing learning program designed as a roguelike video game for Windows computers. Inspired by popular typing games and websites, such as Typing of the Dead, TypeRacer, and MonkeyType; Keyboard Warrior combines elements from these sources into an engaging and interactive application focused on improving users' typing skills. The program's key features include dynamic difficulty scaling, ensuring progressively challenging scenarios, and a unique typing-based combat system that incorporates offensive and defensive maneuvers. The game aims to cater to two main audiences: enthusiasts of roguelike games seeking a fresh experience and computer users looking to enhance their typing proficiency. With a strong emphasis on both entertainment and skill development, Keyboard Warrior makes the process of typing longer and more intricate words quickly and accurately an enjoyable and achievable challenge.

In the development phase, the project will focus on creating the Minimum Viable Product (MVP). The typing-based combat system and per-game character progression system will be the primary focus initially, with room for additional features if time allows. The development activities will involve conceptualizing and implementing game mechanics, testing and refining gameplay, and ensuring a user-friendly UI. The project's milestones include completing the typing mechanics, implementing the typing combat system, implementing abilities and items, introducing progression, and finalizing UI and the theme of the game. Beyond the program itself, other milestones include a final presentation and poster

### 4.1.1 Project Deliverables

- Project Proposal Document - Week 2
  - Document containing a description of the project and the justification for its creation
- Software Requirements Specification - Week 5
  - Document specifying what exactly is being built for the project
- Software Development Plan Document - Week 7
  - Document describing the process that will be used during the projects period of development
- Written Status Reports - Every 2 weeks
  - Biweekly reports on the projects development, specifying tasks currently being worked on, future tasks, and current troubles
- Design Mockup and Idea Document - Week 3
  - Document containing notes and ideas for concepts, mechanics, and implementation
  - Also includes sketches for UI
- Base Prototype - Week 4

- Unity project file containing basic typing mechanic required for additional features
- Milestone Builds - Every 2 weeks
  - Iterative versions of the Unity project with incremental features and improvements
  - Will include builds that focus on the implementation of the mechanics referenced in the introduction
    - These include the following:
      - Typing mechanics
      - Combat system
      - Abilities and Items
      - Game progression
      - UI and theme refinement
- Final Project State - Week 17
  - The final version of the project before final presentations

## 4.2 Project Resources

- Sebastian Cruz
  - Responsible for all planning and development

### 4.2.1 Hardware Resources

- Windows laptop

### 4.2.2 Software Resources

- Unity game engine software

## 4.3 Project Organization

The project is divided into 2 major functions: typing and combat

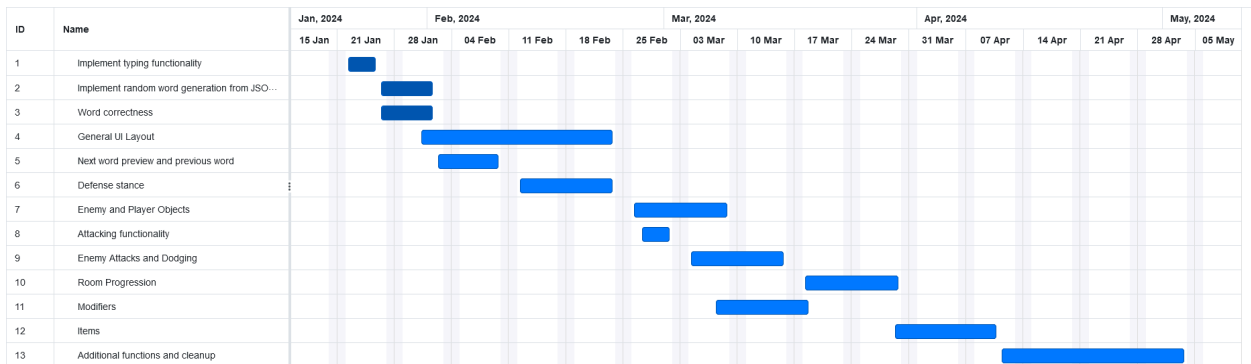
The typing functionality is responsible for allowing the user to type words generated from a json file and receive feedback on their accuracy and speed of typing. The development plan will begin with setting up player typing and the generation of random words from a list, checking correctness of user input, calculation of WPM, chances of generating words from more difficult lists.

Combat functionality is responsible for using the typing functionality to apply damage to either the player or the enemy. It is also responsible for creating an attack pattern for the enemy that the player must react to by switching between attacking and defending. The development plan will begin with setting up player stance changes, player and enemy object creation, damage applied from correct typing, enemy attacks, dodge notifications.

## 4.4 Project Schedule

The following section provides scheduled information for Keyboard Warrior.

4.4.1 GANTT Chart



4.4.2 Task / Resource Table

- Because this is an individual project, all tasks will be completed by Sebastian Cruz
- Because the project is developed solely through Unity, all tasks will require Unity game engine software and a device with Windows
- 

Task	To	Hardware Resource	Software Resource
Typing functionality	Sebastian	Windows device	Unity game engine
Random word generation from JSON file	Sebastian	Windows device	Unity game engine
Word correctness	Sebastian	Windows device	Unity game engine
General UI Layout	Sebastian	Windows device	Unity game engine
Next word preview and previous word	Sebastian	Windows device	Unity game engine
Defense stance	Sebastian	Windows device	Unity game engine
Enemy and Player Objects	Sebastian	Windows device	Unity game engine
Attacking functionality	Sebastian	Windows device	Unity game engine
Enemy attacks and dodging	Sebastian	Windows device	Unity game engine
Room progression	Sebastian	Windows device	Unity game engine
Modifiers	Sebastian	Windows device	Unity game engine

Items	Sebastian	Windows device	Unity game engine
Additional functions and cleanup	Sebastian	Windows device	Unity game engine