Type Defender

1. Project Overview

The Type Defender game is a fast-paced action-based typing game. In this game, players defend their base by typing words that appear as enemies. The challenge increases over time, with enemies becoming stronger (longer words) and faster. The game tracks typing speed, accuracy, and mistakes to calculate the player's performance.

2. Project Review

A similar game to Monkeytype, Type Defender adds an element of defense gameplay where players must quickly type words to destroy incoming enemies before they reach the base. The core concept of tracking typing speed and accuracy remains the same, but Type Defender introduces an engaging action sequence where the typing accuracy directly impacts the survival of the player's base.

3. Programming Development

3.1 Game Concept

Players are given a text passage from the dataset(csv file) to type.Speed, accuracy, and mistakes are measured, damage are taken once the enemy reach the base, the word are chosen at random from the dataset

key features asd

- Players type words to destroy incoming enemies(non duplicate).
- Difficulty increases over time with stronger(Longer word), faster enemies.
- The game tracks key statistics like typing speed (WPM), accuracy, and error rates.

3.2 Object-Oriented Programming Implementation

The game will include the following classes:

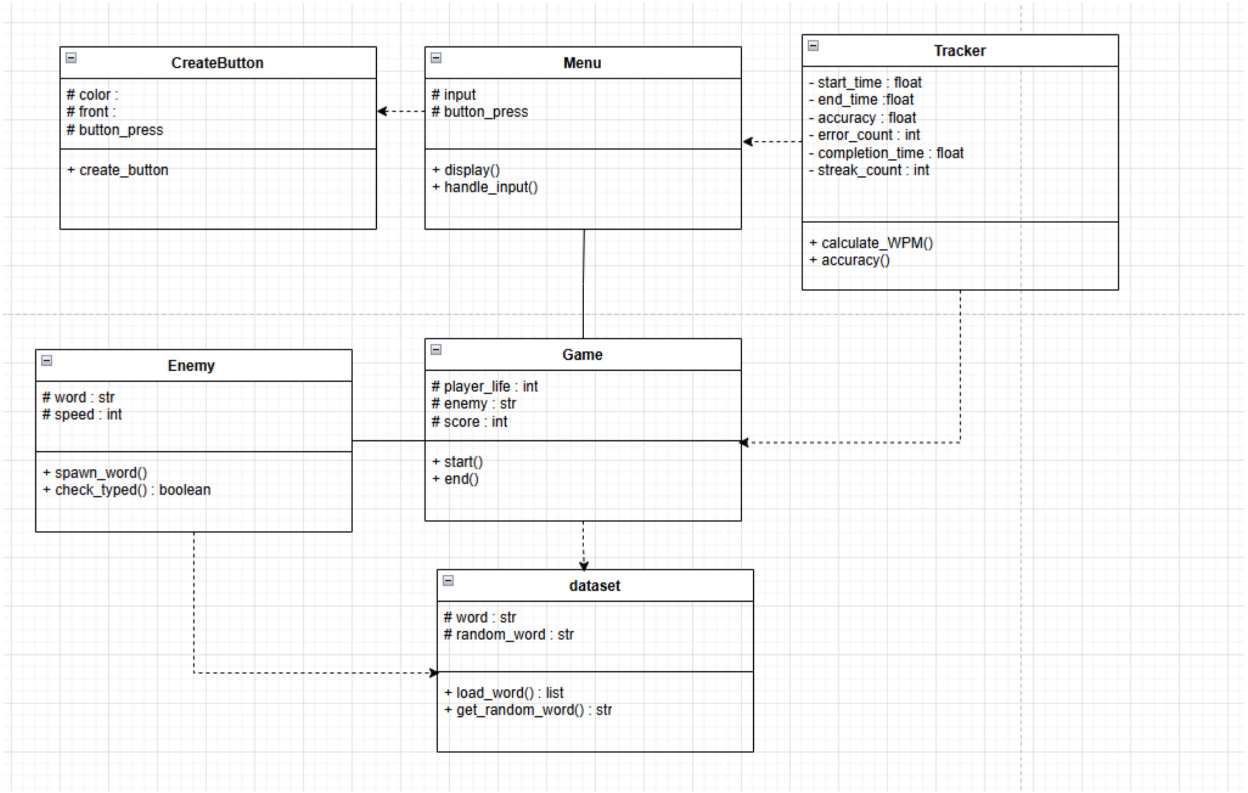
- **Game** – Runs the entire typing game, starting and ending the game sessions.
  key attributes: player,enemy,score
  key Methods : start(), end()

- **Tracker** – Tracks and calculates typing performance, including words per minute (WPM) and accuracy.
  key attributes: start_time,end_time
  key Methods : calculate_WPM(), accuracy()

- **Enemy** – Creates words to type and checks if the typed word matches the enemy's word.
  key attributes: word,speed
  key Methods : spawn_word(), check_typed()

- **Dataset** – Manages the word dataset(csv file), random words to spawn for the player.
  key attributes: words,random_word
  key Methods : load_words(), get_random_word()

- **Menu** –Handles the user interface (UI), displaying options and handling player input.
  key attributes: input,button_press
  key Methods : display(), handle_input()



More classes may be added later.

3.3 Algorithms Involved

Typing Speed Calculation: Measures how many words the player can type per minute (WPM), directly impacting the score.

Detection: Ensures that correct typing destroys the right enemy, even when multiple enemies are present.

Scoring System: The score is based on a combination of speed (WPM), accuracy, and time taken to destroy each enemy.

4. Statistical Data (Prop Stats)

4.1 Data Features

At the end or each sessions a scoring system will track player score and
The game will track the following statistics for analysis:

- **Typing Speed (WPM)** - Measures words per minute.
- **Typing Accuracy** - Tracks percentage of correct keystrokes.
- **Error Rate** - Counts incorrect characters and their positions.
- **Time Taken** - Records the time to complete a word.
- **Keystrokes per Second** - Monitors typing consistency

Additionally, at the end of each session, a scoring system will track the player's score then compare with error rate, accuracy rate, and longest streak

	Reason	Data Collection Method	Class / Variable	Display
Typing Speed (WPM)	To improve the player's typing speed over time, providing feedback on their progress.	Compute WPM every session	Class: Tracker Variable: Start_time,end_time, count_key_press	Graph : line chart
Typing Accuracy	To track how accurately The player types and identify areas of improvement	Compare typed world with the actual words	Class: Tracker Variable: accuracy	Statistic : Mean,Min,Max
Error Rate	To identify the frequency of mistakes and adjust difficulty accordingly	Track error per session	Class: Tracker Variable: error_count	Graph : line chart
Word Completion Time	To measure how quickly players can type individual words, which helps in reaction time and efficiency	Track time taken to complete a single word.	Class: Tracker Variable: completion_time	Statistic : Mean,Median
Number of Mistyped	To give insight into the	Tracks player errors at	Class: Tracker	Graph : Histogram

Words	types of words or characters the player struggles with	typing word	Variable: mistyped_words	
Longest Streak	To motivate players by tracking their consistency, enhancing player engagement	Track how many consecutive words were typed correctly	Class: Tracker Variable: streak_count	Statistic : Mean,Min,Max

Feature name	Graph objective	Graph type	x-axis	y-axis
Typing Speed	To see player typing speed	Line Chart	Time	Words Per Minute
Error Rate	To identify the frequency of mistakes	Bar Chart	Sessions	Error Rate (%)
Number of Mistyped Words	o give insight into the types of words or characters the player struggles with	Pie Chart	Word Categories(length)	Mistyped Words (%)
Typing Accuracy	To track how accurately The player types	Area Chart	Time	Accuracy (%)
Word Completion Time	To measure how quickly players can type individual words	Scatter Plot	Word Length	Completion Time

4.2 Data Recording Method

Data will be stored in a CSV file

4.3 Data Analysis Report

Outline how you will analyze the recorded data. What statistical measures will you use? How will the analysis be presented (e.g., graphs, tables, charts)?

- **Performance table**– Tracks WPM, accuracy, and consistency over multiple sessions.
- **Graphical report**– Visualizes WPM, accuracy trends, and keystroke timing with line graphs.

5. Project Timeline

Week	Task
1 (10 March)	Proposal submission / Project initiation
2 (17 March)	Full proposal submission
3 (24 March)	Finalize game design and core functionality
4 (31 March)	Implement core gameplay mechanics
5 (7 April)	Begin developing Tracker class
6 (14 April)	Submission week (Draft)
7 (23 April)	Finalize gameplay features and UI
8 (11 may)	project completion

6. Document version

Version: 4.0
Date: 31 March 2025

Date	Name	Description of Revision, Feedback, Comments

14/3	Pattapon	The idea is good overall :), but some parts need more details as I mentioned in my comments.
15/3	Phiranath	The idea is interesting, some parts need clarification. Don’t forget to add project timeline and reformat topic number.
29/6	Phiranath	Project timeline and version are still incomplete.