Project Report

Project Team Members:

- Nikolay Sizov
- Elizabeth Johney
- Nicanor Sanderson

CMPCS 463 Final Project
Date: 12/02/2023

Project: Personal Achievement as a Game

1. Project Goals:

The primary goal of the Personal Goal Game is to provide users with a gamified approach to achieving personal goals. The project aims to allow users to define goals, break them down into tasks, and visualize their progress through an interactive user interface. By incorporating game elements, the project intends to make the process of goal-setting more engaging and enjoyable.

2. Significance of the Project:

The project is significant as it aligns with the concept of Safe Space, providing users with a tool that encourages personal growth and self-improvement. By combining goal-setting with a game-like experience, the project offers a unique approach to fostering motivation and discipline. Users can track their progress, visualize task dependencies, and celebrate achievements within a supportive digital environment.

3. Installation and Usage Instructions:

Installation:

- Ensure you have Python installed on your system.
- Clone the project repository from GitHub.
- Install the required dependencies using `pip install -r requirements.txt`.

Usage:

1. Run the main application file (`game.py`) using a Python interpreter.
2. Use the provided GUI to add goals, tasks, and visualize progress.

3. Save and load goals as needed.
4. Play the game by starting the Pygame interface.

4. Code Structure:

The code follows a modular structure, comprising multiple Python files for better organization. Here is an overview of the code structure:

- `game.py`: Main entry point for the application. Defines the `Goal` and `Task` classes
- `game_ui.py`: Contains the Pygame-based game interface.
- `data`: Directory stores saved goal objects as JSON files
- `sounds`: Directory stores royalty free sounds for the gameplay

5. List of Functionalities and Test Results:

Functionalities:

1. **Add Goal:** Users can add goals, specifying the goal name and reward.
2. **Add Task:** Tasks can be added to goals, with points and optional dependencies.
3. **Show Progress:** Visualize progress through completed and pending tasks.
4. **Calculate Path:** Determine the optimal path through tasks using Dijkstra's algorithm.
5. **Show Tasks Graph:** Shows available tasks and connection, based on directed graph
6. **Save/Load Goal:** Allows saving and loading previously saved goals containing tasks
7. **Edit Task Dependency:** Modify task dependencies for sequential or parallel execution.
8. **Play Game:** Starts the game when goal is configured or loaded from file

Test Results:

- User acceptance testing was performed to validate the user interface's effectiveness.

6. Discussion and Conclusions:

Project Issues:

- One limitation is the complexity of task dependency management for larger goals.
- The Pygame interface may require additional features for enhanced user engagement.

Application of Course Learnings:

The project applied concepts learned in the course, such as object-oriented programming, graphical user interface design, file I/O, and algorithm implementation. The use of Pygame demonstrated practical application in creating interactive game elements.

Conclusion:

The Personal Goal Game offers a creative and interactive solution for goal-setting. While there are areas for improvement, the project successfully integrates course concepts into a meaningful application that aligns with the Safe Space philosophy. The combination of a Tkinter-based interface and Pygame elements provides users with an engaging tool for personal development.