Little Journey

Authors:
- Belciug Matei 325CA
- Bigan Radu-Cristin 325CA

1. Introduction

1.1. Description

Our project is a 2d adventure choice-based game where the player has to make decisions that affect their progress through the story. The player can collect items and store them in the inventory, can engage in fights, and can reach different endings. The gameplay consists of interacting with different characters and selecting the desired dialogue option. The target market is the mini-game enjoying demographic of players.

1.2. Functionalities:

- interface: GUI
- menu
- local save system
- inventory system
- player stats

2. System Overview

The core component of the application is the Pygame library used for implementing basic gaming functionalities. The game state will be saved using the BSON format. The user can select the save file that he wants and for loading we will use a predefined path relative to the game location in the file system. Also, for an easier implementation we will use object oriented programming paradigm.

3. Detailed Component Design

- Main Menu
  It represents the starting point of the game. The player can choose to view the credits, the help section, their saves and of course to start the game.

- Choice Making System
  Pressing “Esc” will cause a drop down settings menu to appear, otherwise the player has to select the option he prefers from the options displayed on the screen. To do so they can press the number corresponding to the choice, use the arrows to select it, or just click on it.
- Characters
  These will have different dialogue lines, different appearances, stats such as damage or hp and so on.
- Inventory
  The player will be able to store and use a limited number of items. Each item has different characteristics, some heal, some provide a temporary buff, some have special uses and some do nothing.
- Combat
  The player will sometimes engage in fights with various enemies. During these, certain effects can be applied such as freeze, poison and others, besides the classic damage dealing.
- Story Progression System
  There are some aspects that will influence story progression such as: some special dialogues, the location of the player that can be selected on a map and environmental attributes such as the likelihood of an event to take place.
- Save System
  The relevant information is saved in a BSON format and loaded when the player chooses to do so. There will also be an automatic autosave kicking in after important story progress.

4. Testing

We will use unit-testing for non-GUI parts of the game logic. This includes things like player health updates, item usage, collision detection and other core mechanics. For the GUI parts of the application we will use manually testing.

5. Conclusions

For a 2 man project we think it will meet the desired level of complexity. We belive we will encounter a workload bigger than 20 hours per person to properly implement all our desired functionalities. To be honest, we are more

excited about the implementations of the different functionalities of our project rather than actually crafting a story. We will have to learn to use Pygame which will help us develop our python skill set.