Software Implementation and Testing Document

For

Korona Kingdom

Version 1.0

Authors:

Karl Cooley

Alexander Jordan

Ryan Goldberg

Anthony Micciche

Alexander Kostandarithes

1. Programming Languages (5 points)

- Our project is implemented mainly with Python. The majority of us had Python experience and wanted to make a game so Alex recommended we use the Python arcade library. It has to be Python 3.6 or greater because the arcade library does not support the older versions of Python.
- We are also using Extensible Markup Language (XML) for map creation. This was the result of us wanting to use a tiled map editor recommended by arcade. The output of using this editor is a file in the translation memory exchange(.tmx) format.

2. Platforms, APIs, Databases, and other technologies used (5 points)

Inside Project:

- Our project is implemented using the Python arcade library (https://arcade.academy/) for source code.
- The Tiled map editor (https://www.mapeditor.org/) for map generation,

Project Tools (Each of us have our own set-up):

- Editor: Pycharm
- Version Control: Github
- Image Manipulation: GIMP, Microsoft Paint, Pinta (https://pinta-project.com/pintaproject/pinta/)

3. Execution-based Functional Testing (10 points)

Karl Cooley

Inventory

- Tested than when the player hits [I] the inventory pops up / exits. There is currently no doubling of panels. (1/2)
- Tested that the info shows up, however, it is just static information as there is no player class so its incorrect information. (3)
- Tested that correct equipped item shows. However, this was tested with only 2 different images so further tested is needed when new items are added. (4)
- Tested that when the user opens the inventory the tab selected is always in the top left and that when the user scrolls left/right to change the component type it doesn't go out of bounds. However, at this time we don't have any text to show what each of these tabs should contain. (5/6/7)
- Tested that when the user opens the inventory all items they have should appear (even if they have nothing). If a list contains more than one item they can scroll up/down without going out of bounds. (8/10pt1)
- Tested that the selected panel is highlighted in blue, and if they have more items than slots given, they can scroll past 8 slots and all items will just shift upward once. Not possible to go out of bounds. (10pt2)
- Tested that the inventory and encounter system should never appear on screen at the same time. Tried to movement keys while in inventory and using [I] while the encounter screen was up. Current behavior is correct. (13/14)

- Tested that when the player hits [Enter]/[BackSpace] items are equipped and unequipped. Checked that arrays showed this data and made checks to make sure 'None' could never try to be accessed (15/16)

Overlay

Player Info

- Tested that when the user walks around the overworld the user's info, user's energy, and player's image is shown and static. However, this information is incorrect. (1/2/3)

Dialogue Box

- Tested that the dialogue box shows who is speaking, shows an image of the speaker, and shows what is being said. However, a problem has come up from a member using a Linux machine causing the text to be a different font and go out of the correct printed area. (2/3/4)
- Dialogue Box is only on your screen when it needs to be now (Events/Encounter) (1/5)

Menu Box

- Tested that when the user walks around the overworld the menu boxes are shown and static. The only way to access menu options is by keyboard shortcuts (which will be tested in their associated sections). (1)

Event

- Tested that when a player walks into a dialogue event the correct event shows up and the player is unable to move (1/3)
- Tested that when Inventory is open in an event they can interact with their inventory. However, pressing [Enter] while this is open will continue the dialogue event that should not be able to be modified. (4)
- Tested that multiple dialogue speakers works and that pressing [Enter] makes the conversation continue (5/6)
- Tested that once an event is over the dialogue box disappears and the event also disappears from the map. Even if the user switches maps the dialogue event will stay gone forever. (2/7)
    - Ex. An event you saw on the overworld shouldn't happen again if you return from a dungeon

Inventory (Alexander Kostandarithes):

- Tested that items added do appear in the player items list
- Tested that correct stats are being displayed
- Tested that appending items to certain spots in the player list add where they need to go

Encounter system(Alexander Jordan):

- Anthony and I agreed that the old trigger for the encounter system was not adequate (it only caused encounters on key press in the old version. So if the player just continued moving in one direction without changing it was too easy for them to avoid encounters). We decided to change the trigger to be based on whether the player was in motion at all. Upon testing this I had to greatly reduce the probability that an encounter would pop up compared to the old version because the encounter system was now being triggered far too often.
- An additional bug with the encounter system that I discovered was that it was repeatedly calling the encounter.handle_selection() function everytime the game loop updated until the encounter ended rather than only calling the function one time when the user actually makes a selection. I rearranged the code that called this function in order to ensure it is only called once per user selection.
- Upon Anthony and I integrating the encounter and combat systems together, we ran into some bugs that needed fixing.
    - Originally the encounter system worked as expected for the first two encounters, then upon the third encounter combat did not seem to be occurring and the only output to the game showing was “None”.
    - After some time spent reviewing the code Anthony and I were able to determine that the health of the enemies within the encounter was not being reset which caused the bug.
    - We made some adjustments and now it works as expected.

Combat System (Anthony Miciche):

- Tested functionality of data structures that enable the combat to work
    - Dictionary of string function pairs for various attacks
    - Dictionary of string object pairs for enemies
- Tested that the various stats of players and enemies are being entered in the correct order and processed correctly.
- Tested that a random enemy is called from the getEnem(loc) function, provided that a string is provided to select what area of the game the player is in.
- Tested that enemy AI is working correctly
- Tested that one random encounter does not affect the next, aside from your HP being potentially lower

Animations (Ryan Goldberg)

- Tested the walk animation with a higher refresh rate but reverted to a lower rate as it caused game lag

4. Execution-based Non-Functional Testing (10 points)

Karl Cooley

Inventory

1. Tested that the inventory system took less than a second to load.

Overlay

2. Nobody has reported performance problems of the Overlay being slow.

Alexander Kostandarithes

Inventory

- The Inventory can managed all items that a player holds at a given time

Map/dungeons(Alexander Jordan)

- I created more graphics/art for the dollar store dungeon, and ran the game to ensure that this art was pleasing enough during gameplay.

Encounter system(Alexander Jordan):

- Upon testing the encounter system after integrating the combat system only some of the relevant output was being displayed to the user. Anthony and I tracked down the issue and fixed this. Now the relevant information about the user's attacks as well as the enemy's attacks are displayed in the dialogue box during the encounter.

Animations (Ryan Goldberg)

- Reverted to only using two images for the animation and updated the array accordingly. Still easily conveys walking to the user.

5. Non-Execution-based Testing (10 points)

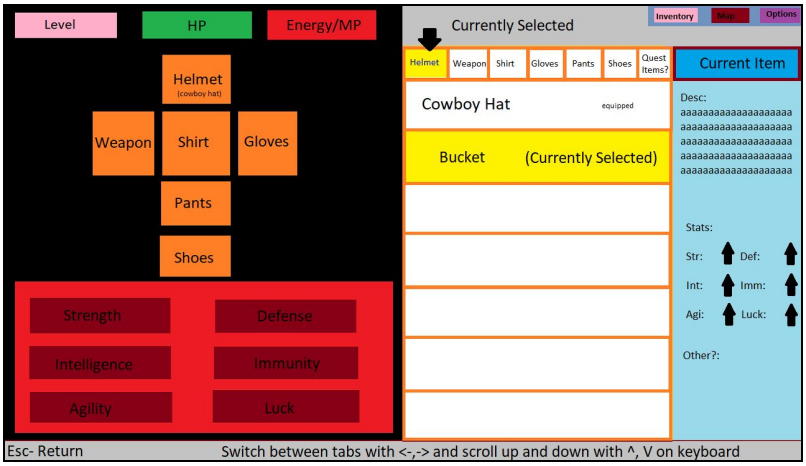
Everyone:

We hold meetings every MWF, where we discuss what we have been working on, and often show our code to each other for approval and remarks.

Karl Cooley

Inventory

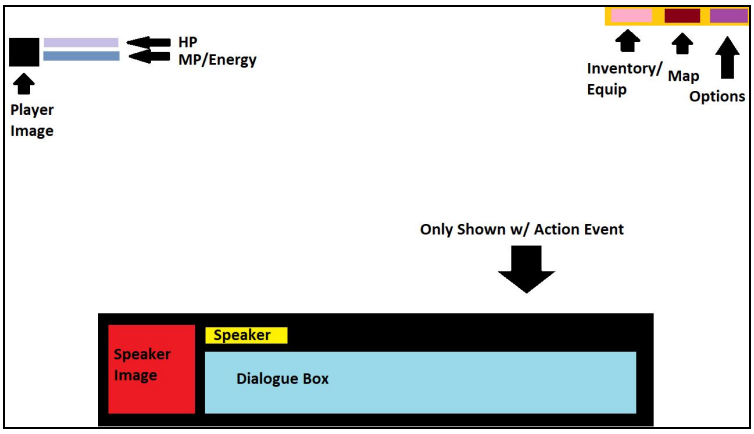
1. Asked members if they approved of the inventory template. Created a mock-up of inventory before code implementation.



2. Code reviewed Alexander's Entity class and gave suggestions on how to implement it into the current Inventory system.

Overlay

1. Asked members if they approved of the overlay template. Created a mock-up of overlay before code implementation.



Alexander Kostandarithes

Inventory:

By hard-coding items into the inventory, testing that items appeared and displayed their stats was possible.