Progress Report

- Increment 2 -

Group #1 Korona Kingdom

1) Team Members

Alexander Jordan - arj14b; github: A-Jordan-95

Alexander Kostandarithes - ak17c; github: FamousStephens

Karl Cooley - kzc18; github: Kazaco

Ryan Goldberg - rag19d; github: tabularasa98

Anthony Micciche - ajm17h; github: amicciche

2) Project Title and Description

Korona Kingdom:

A turn based RPG set in a parody of the current covid-19 pandemic. Users must defeat the bosses contained within three dungeons while fighting off enemies in random encounters. If they make it to the last boss and defeat them they will win the prize of a single roll of toilet paper.

3) Accomplishments and overall project status during this increment

- Implemented an Event System
    - Events are triggered by running into “Event Tiles” on the map
    - Dialogue Box will only appear in the context it is needed now (Events/Encounters)
    - Supports multiple speakers/dialogue chains
    - Dev Team can easily add new events using a Tile ID/Tiled Editor
- Implemented new features to Inventory System
    - Scrolling through found items
    - Equipping/Unequipping items
    - Displaying item stats for a highlighted item
- Added clearer walking animations
- Edited world design for dungeons
- Changed trigger for encounter system from a movement key being pressed, to the player’s movement speed being greater than 0 (i.e if the player is in motion there will be a random chance that they will run into an encounter)

- Updated how the encounter.handle_selection function is called in order to ensure that only one call to this function is made per user selection within the encounter menu. In the previous increment this function was being called repeatedly.
- Integrated the combat system with the encounter system so that the player can now engage in a combat situation.
- Overhauled the data management of the combat system to allow for rapid expansion in the next increment.
    - Created a master move list dictionary, which uses strings (move names) for keys and functions (the function calculates how much damage the attack did) for values.
        - To add a new attack, we just need to write its damage function and add it to the dictionary
    - Created an enemy dictionary, which uses strings (enemy names) for keys and lists of objects for their values
        - By passing the name of a location to enemyMap, one gets a list of enemies that can appear in that area. One is randomly selected and a new copy is made for combat.
        - Each enemy has its own object, however they all have the same member data and functions in them.
            - Each enemy has all stats in the game, as well as a movelist, and a getDecision function to work as very simple AI. By giving each enemy its own AI, we can have different enemies behave differently.
    - Implementing a rudimentary save file system.
    - Changed return types on some functions to facilitate integration with UI and moved various bits of code around
        - For example, output is now all handled in the same function which makes printing it all to the UI a much simpler task

4) Challenges, changes in the plan and scope of the project and things that went wrong during this increment

Events Time-Wasting Annoyances

- Wrote 3 alternative ways to try and read a .tmx file because each way initially had a problem
    - 1. Parsing .tmx file to retrieve Object layer information
        - arcade supports reading Tile Layers with process_layer but not Object Layers
        - Tried to parse this file using a loop but found that (x,y) coord of Tiled objects does not match that of arcade, but could retrieve object properties
    - 2. Making Tile's have properties
        - arcade supports Tile Layers with process_layer
        - Tried this but couldn't find a way to read properties from a Tile still without parsing, no method was in their official documentation on the website. So I stopped and went to (3).
    - 3. Overriding process_layer function in arcade

- Tried rewriting process_layer from their source code to return custom sprites (Alex suggestion) with property information. However, through this process I found that they used a property call that I needed for (2) to work.

5) Team Member Contribution for this increment

Karl Cooley:

Progress Report: Contributed to Challenges (Events) and Accomplishments Section. (Events/Inventory)

RD Document: Contributed to Functional Requirements (Inventory-15/16 new and updated, Overlay - updated, Event System - new), and Non-Functional Requirements (updated Inventory/Overlay and added Event System), Use-Case Diagram (new Inventory cases and Events), updated my portions of Class Diagram (Inventory/RPG/Overlay and added Event), and modified Sequence Diagram (Added Event/Entity).

IT Document: Contributed to Execution-Based Functional Testing(Inventory - some new stuff, Overlay- updated, Event - new stuff), Execution-based Non-Functional Testing (updated), and Non-Execution-based Testing (Reviewed Code).

Source Code: Contributed to RPG.py (set-up anything related to Inventory/Event functionality in-game), all of Event.py including current events on the map, and added LHS related stuff for Inventory equipping/unequipping items(waiting on Alexander atm), and scrolling for Inventory.

Presentation: Talked about Events(event tiles,dialogue events) and Inventory RHS additions(scrolling, equipping).

Alexander Kostandarithes:

Progress Report: Contributed to sections 3,6

RD Document: Contributed to sections 2,3

IT Document: Contributed to sections 3,4,5

Source Code: Contributed to Entity.py (creating entity class and inheriting ones) and Inventory.py (ability to highlight selected items and show relevant stats

Presentation: Discussed the Inventory LHS additions (item highlighting and item stats)

Alexander Jordan:

Progress report: Contributed to sections 1,2,3,5, and 6 of the progress report.

RD document: Contributed to sections 1,2,4,5,6, and 7 of the RD document.

IT Document: Contributed to sections 1,2,3, and 4 of the IT document.

Source code: Contributed to RPG.py and Encounter.py (all sections regarding the encounter system as well as handling code for processing of the maps)

Presentation: Talked about the encounter system changes and dollar store dungeon changes.

Anthony Micciche

Progress report: Contributed to everything related to the combat system

RD document: Contributed to everything related to the combat system

IT Document: Contributed to everything related to the combat system

Source code: Wrote koronakombat.py, enemyList.py, masterMoveDict.py, and playerData.txt . Contributed to Encounter.py during the integration of combat and UI.

Presentation: Talked about combat development in the video

Ryan Goldberg

Progress Report: Contributed to sections 1,2,3,5, and 6 of the progress report.

RD document: any sections pertaining to player animations or art design

IT document: any sections pertaining to player animations or art design

Source code: made slight changes to the animation.py file and updated images within the player sprites folder

Presentation: briefly discussed the new walking animations and compiled all video clips together for the final presentation

6) Plans for the next increment

Alexander Jordan:

- For the next increment I plan to have the encounter system fully operational and integrated with Anthony's combat system without any bugs/errors.
- I also plan to have all of the dungeon maps art fully implemented.
- I also plan on implementing the boss encounter of each dungeon.

Anthony Micciche

- Will make use of the structural changes made to combat this increment in order to add many more attacks and enemies.
    - New enemies will also mean new AI methods
- Will add behavior that occurs when combat ends
    - Exp
        - Will also add leveling system of some sort
    - Enemies will drop currency and items sometimes
- Will write boss enemies

- Will implement immunity, and korona based attacks that will attack max hp and not current hp.

Ryan Goldberg

- Will update dungeon maps with tile editor
- Will add sprite animations when combat and UI systems are integrated
- Program tiles for item pick up and inventory integration
- Take over art design aspects of game in a broader scale

Karl Cooley

- Will update the UI for Inventory and Player Info Box once we have a Player class with non-static data
- Program tiles for item pick-up and dialogue events

Alexander Kostandarithes

- Work on item stats in regards to game balance
- Adding more in-game items in Entity.py

https://drive.google.com/file/d/1i903fXjcFEQw0MKuZuyHk4Ne5V08TIGD/view?usp=sharing