OOAD-Project-6-7
CSCI-5448.
Tim Coleman
Curtis Covington

This is a progress report for Project 6 for CSCI-5448 Su22.
Please see the main README.md doc for the the latest updates as this pdf might go stale

A new Flutter project. It's a game loosely based on the original Zelda 2D NES except this game doesn't have the map to move link “the character” around.

This repo contains Flutter/Dart code to run a Zelda like game as defined in our project proposal outlined in the project_Design_5.pdf

The main work that has been done thus far is described below in the matrix table. You can find the current UML class diagram here that Project_6_Current_UML_Class_Diagram.pdf that has been implemented.

Feature	Done
General Flutter Widget Structure	Yes
Pattern for observer (using flutter based NotificationListener)	Yes
Pattern for factory	Yes
Pattern for singleton	TODO
Pattern for command	TODO
UI Store	Yes
UI Items	Yes
UI Battle	Yes
Store Logic (Multiple classes)	Partial
Battle Logic (Multiple classes)	TODO
Event Logic (Multiple classes)	Partial
Testing	TODO

Below is a high level breakdown of the classes and subclasses.

* GameState (Main): Contains the main game state for the duration of the game
* BattleScreen - UI widget to display Battle Logic
* EventScreen - UI widget to display Event Logic
* Item (Abstract)
    * Boomerang
    * Potion
    * Wand
    * Claw
* Battle - Logic for Battle. UI references
* Event - Logic for Event. UI references
* Store - Logic for Store. UI reference

Code compiled and tested with Flutter/Dart
Dart SDK version: 2.17.5 (stable)
Tools • Dart 2.17.5 • DevTools 2.12.2
Tested on Android Nexus 6 Emulator

Overall Comments

Curtis Covington previously had some experience with Flutter and was able to lay down some structure to help get the project moving, which is wonderful to have his experience. I (Tim Coleman) am coming into Flutter/Dart completely fresh so learning a little about the framework and widget structure was confusing at first but is making more sense now. Part of the goal of this project was to apply what we learned but also take advantage of a different language, which we are doing but it also probably slowed us down a little vs using a language such as Java or Python where the spin-up time would be less. Nonetheless I feel we are making good progress with our game.