Project 7 Update

Project Title: Orchard Odyssey
Team Members: Ben Lipman, Jaden Snell, Will Dravenstott

Video:

https://youtu.be/TBMXjyVuAWA

Work Done:

The current build of Orchard Odyssey features a pop-up screen featuring a main player image with working jump mechanics, moving “enemies” with working collision detection from the main player, and images that the player can collect. Also featured in this build are a point system that keeps track of a player’s score, and a setup for implementing a background and custom sprites in the future.

In terms of who was working on what, Jaden worked on implementing the score, keeping track of collisions, and the game ending condition. Will worked on the jump mechanics for the player, taking in key input, and the setup and plan for the project, and Ben implemented the test cases for all classes to ensure functionality.

Changes/Issues:

In terms of changes, there hasn’t been an obstacle too tough to overcome that was written out in our plan. Changes might have to appear, however, in terms of the decoration of the game, as it is looking to be somewhat challenging to have smooth image generation and endless scroll without hitching.

The issues that appeared so far mostly included working with foreign libraries, most notably the Java swingx library. None of us as group members have ever worked with those libraries before, and there was necessary exploration before incorporation.

Patterns:

Currently, we have been working to get a minimum viable product out for this first release, which entails just getting a framework to elaborate on. Our work in using the design patterns has been delayed due to the high-level nature of the libraries we are using. The patterns that we’re planning on using are a factory pattern to create different types of enemy objects with scroll. The different objects will be those that are in the air, those that are on the ground, and those that are in the middle, allowing the player to either jump or crouch.

Another pattern we will incorporate will be the singleton, as there should only be one instance of the GameScreen class present in the game. This will allow for a consistent and easy-to-use system in updating the object.

The final pattern we will incorporate into the game will be the observer pattern, which will observe game-starting and game-ending scenarios, as well as sensing and storing a new high

score.

BDD Scenarios:

- Feature: Colliding with an obstacle
    - As a User
    - When the player image collides with an obstacle
    - Then the game should end
    - And the score should be shown
    - And a screen should show stating that the game has ended
- Feature: Collectable
    - As a user
    - When the player collides with an image of a fruit
    - Then the score should increase
    - And the fruit should disappear
- Feature: Jump
    - As a user
    - When I use either the W or UP key on my keyboard
    - Then the player image should jump
    - With gravitational acceleration simulated
    - And the hitbox of the player should change based on its location
- Feature: Crouch
    - As a user playing the game
    - When I use either the down arrow or the s key on my keyboard
    - Then the player image should shrink to half its height
    - And the hitbox should be updated to remain with the image

Plan for next iteration:

From now until the time that the final deliverable is submitted, several smaller changes will be implemented. First, looking at the patterns, all three of the design patterns explained above must be implemented and tested in their totality.

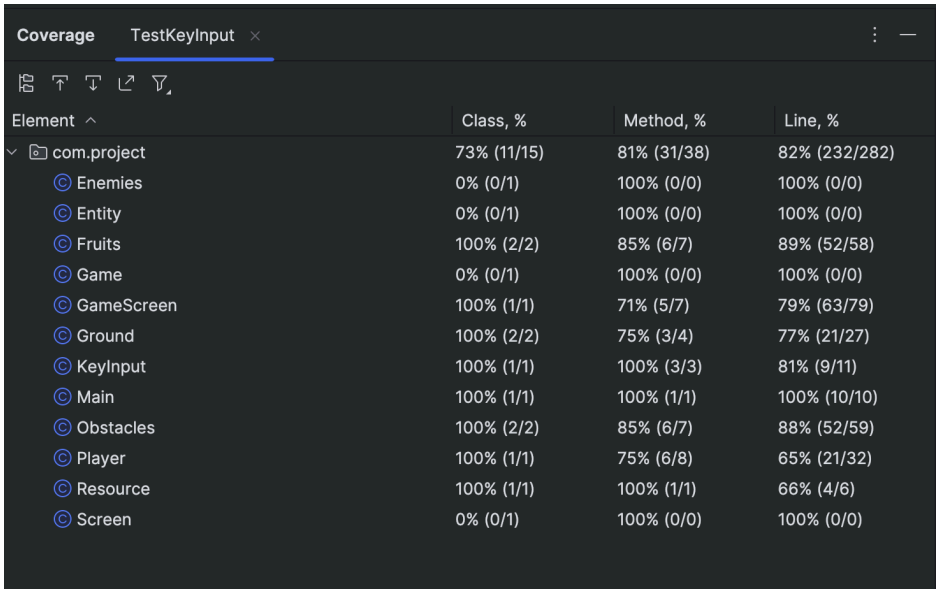
In terms of display, the goal is to have both a title and end game screen that will be displayed as the user reaches a condition for either of them. These screens will feature custom images developed by our group. In addition to this, sprites will be made for the main player, the background, obstacles, and collectable fruits, and will be implemented into the game to make it fully in-house.

In terms of gameplay, the crouch move will need to be implemented, and shown on screen the shrinking of the hitbox.

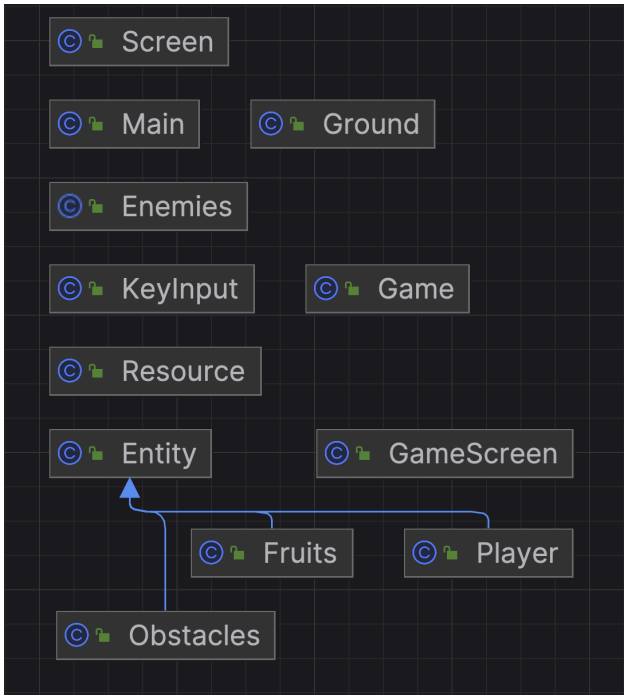
In terms of functionality, the score calculation will be tweaked to increase more the further a player has gotten into the game, and the number of collectable items appearing will

also increase as the game goes on.

Test Coverage:



UML Diagram:



Sources:
https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.hiclipart.com%2Ffree-transparent-background-png-clipart-ydqaa&psig=AOvVaw183YFxhdtAh8mH6aCf_zvg&ust=1712961909671000&source=images&cd=vfe&opi=89978449&ved=0CBIQjRxqFwoTCKDeosKeu4UDFQAAAAAdAAAAABAJ

https://www.google.com/imgres?q=plum%20clipart&imgurl=https%3A%2F%2Fpng.pngtree.com%2Fpng-clipart%2F20210226%2Fourmid%2Fpngtree-translucent-plum-clip-art-png-image_2958872.jpg&imgrefurl=https%3A%2F%2Fpngtree.com%2Fso%2Fplum-clipart&docid=CxtzVw0vBh5yfM&tbnid=zo0zg1p9IhWEZM&vet=12ahUKEwiDwMbVnruFAxXqFTQIHUOeCH8QM3oECEwQAA..i&w=360&h=360&hcb=2&ved=2ahUKEwiDwMbVnruFAxXqFTQIHUOeCH8QM3oECEwQAA

https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcRtmpRu2m7cD_LgC8E35wH5CtDsYFEMTRn1BeZlxAiHxg&s

https://www.google.com/url?sa=i&url=https%3A%2F%2F3docean.net%2Fitem%2Flittle-characters%2F29800510&psig=AOvVaw3Ze1W7hjD-hHo_GLjVl-Mw&ust=1712962210833000&source=images&cd=vfe&opi=89978449&ved=0CBIQjRxqFwoTCODp-tGfu4UDFQAAAAAdAAAAABAE