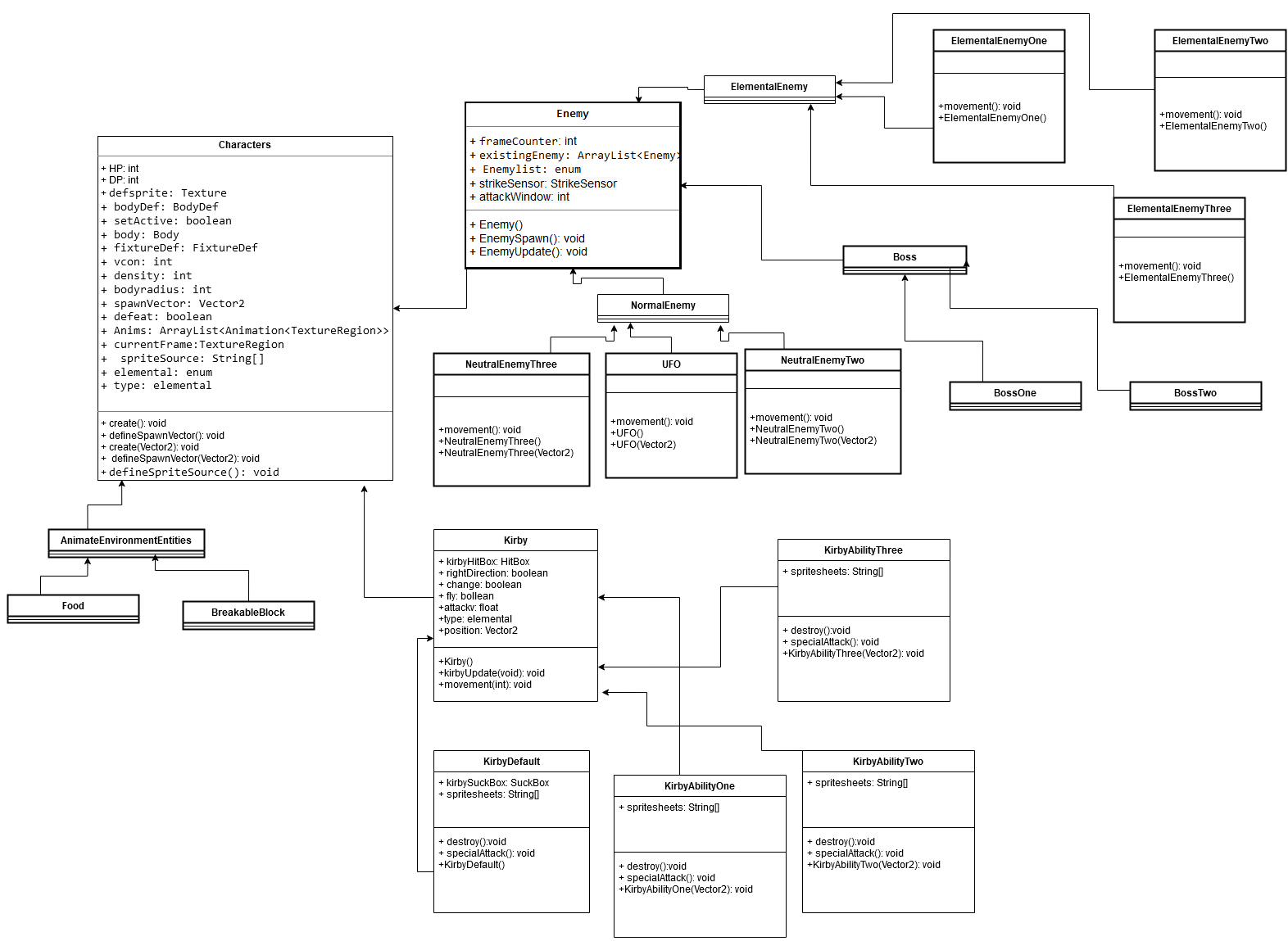
GAME DESIGN

Project Description: A 2D platformer game where players controls Kirby to go through game stages.

Characters: Kirby. Kirby can walk. Kirby can fly. Kirby can inhale.

Game Mechanics: Arrow keys to move horizontally and vertically. A neutral attack move and a special attack move, which is “inhale” for the default kirby and others for elemental kirby (kirby after inhaling an elemental enemy.)

Class Diagram:



Level Design: TBA

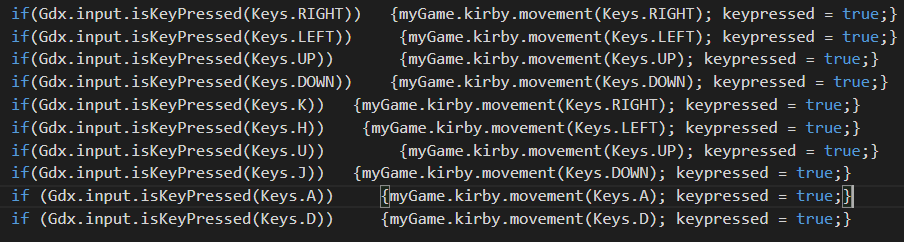
Items: TBA

HUD: TBA

Music and Sounds: TBA

User Control:

In an Open Stage, press Enter to start the game.

In a Game Stage, User can use arrow keys and back-up keys as below to move. Use A to perform special attack and D for neutral attack.

Animation:

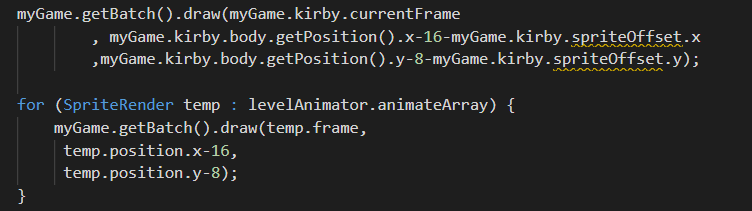
The Animation is done by clearing the previous screen and drawing a new frame on top of it

* Clear screen



* Draw on top



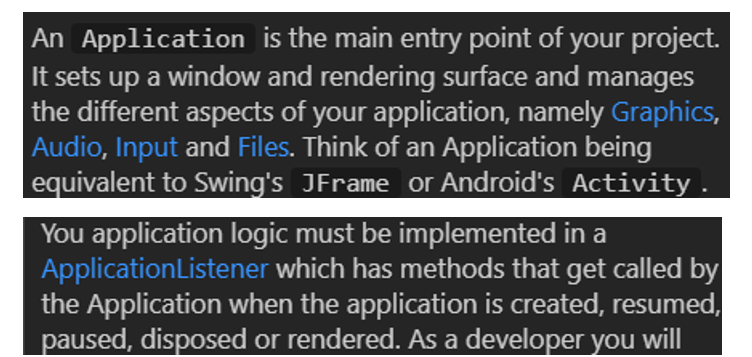


Platform: desktop

[turn to next page for coding documentation]

CODING DOCUMENTATION

The project uses the libgdx framework and creates a desktop game. The core asset and source code is placed in the ‘core’ folder and the desktop display, which contains the main method, is in ‘desktop’ folder. The main method calls the myGame constructor and configure display width and height.



The myGame class is a child class of ApplicationAdapter, a libgdx class (refer to the information provided by developers in the two images above.) In the myGame class, there are three methods: create() which creates and imports the assets used in the game, render() which is called periodically to update the graphics by batch.draw(), update the physics world by world.step(), receive user input with managerUI() and call custom update function updateEntities() to modify any of the physics bodies. In addition, world.step() also has Listener as a ContactListener.

The class hierarchy is attached as the file Kirby.html