Sean Kunz

George Akgul

Dillon Feinman

**Lab 8 Proposal: Kangaroo Run**



https://github.com/skunz42/Kangaroo-Run

CS 110

Fall 2016

II. Public Interface Design:

CS 110 Final Project User Guide

Kangaroo Run

Sean Kunz, George Akgul, Dillon Feinman

Section A0

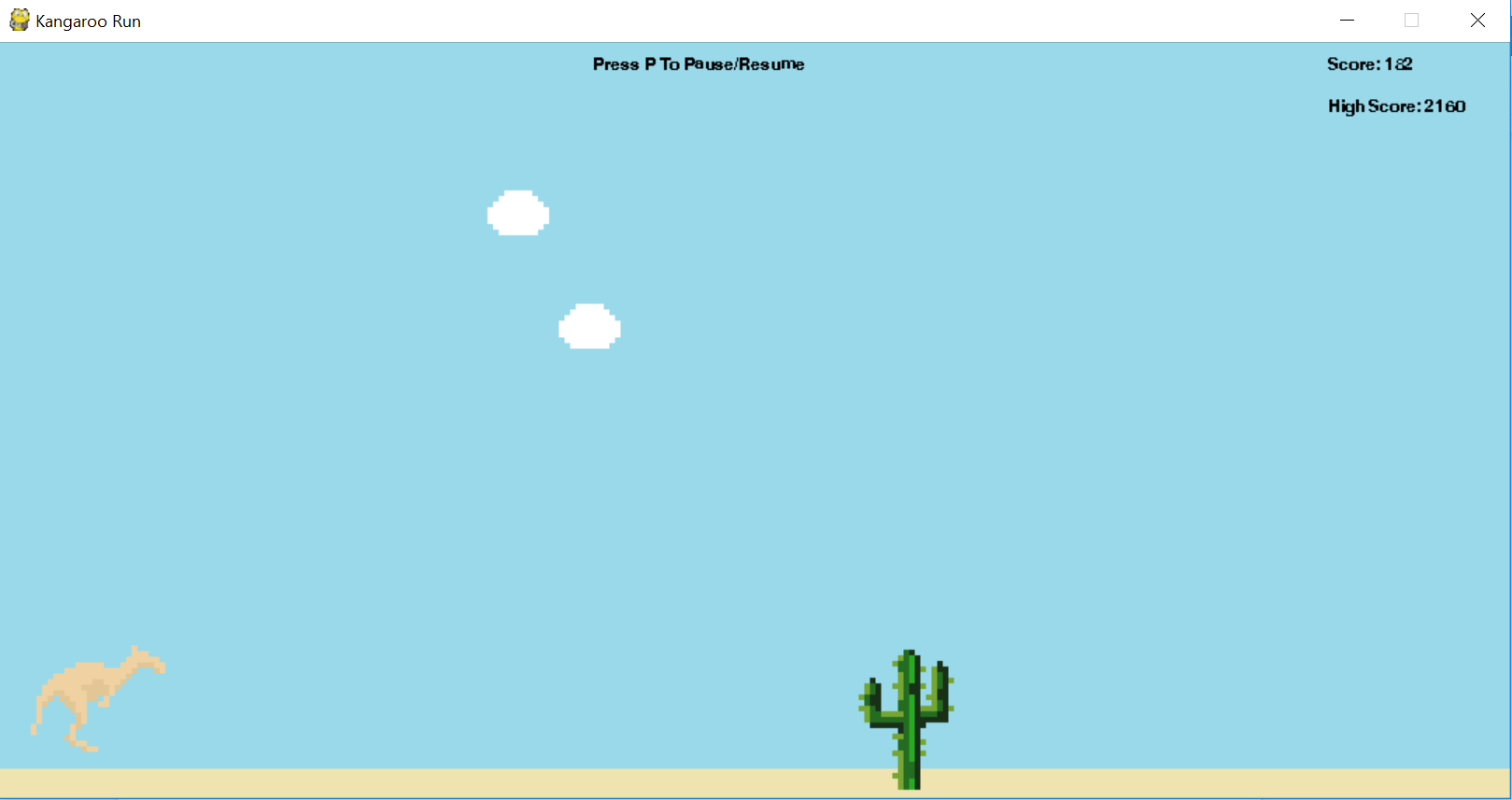
1. Start screen:

The start screen displays the title of the game, Kangaroo Run. Pressing ‘S’ will start the actual game. Controls are also listed.



II. The Game:

After pressing ‘S’, the game is started. The goal of the game is to last as long as possible without colliding with a cactus. Pressing the up arrow key will allow the Kangaroo to jump. Pressing ‘P’ will pause the game, pressing ‘R’ will restart the game, and pressing ‘Q’ will quit the game. The score and high score are listed in the top right corner. If your score is greater than the high score, your score will be recorded.



III. Top Level Design:

Libraries used: Pygame

Classes:

\_\_init\_\_

* Executable
* Imports background, kangaroo, cactus and cloud classes
* Methods:
  + \_\_init\_\_(self) - Initializes screen
  + main() - Executable

Background

* Loads background
* Imports utilities
* Methods:
  + \_\_init\_\_(self) - Returns image and image rect, throws exception if there is an error

Kangaroo

* Loads the kangaroo and its functions
* Imports utilities
* Methods:
  + \_\_init\_\_(self) - Initializes kangaroo
  + reinit(self) - Re-draws kangaroo
  + update(self) - Cycles through images of kangaroo
  + freezeKang(self) - Freezes kangaroo
  + jump(self) - Sets jump equals to true
  + collide(self) - Sets colliding to true
  + draw(self) - Draws kangaroo and actions on screen

Cactus

* Loads a cactus and its functions
* Imports utilities
* Methods:
  + \_\_init\_\_(self) - Initializes cactus
  + reinit(self) - Re-draws cactus
  + freezeCact(self) - Freezes cactus
  + resumeCact(self) - Resumes cactus activity
  + draw(self) - Draws cactus and actions

Cloud

* Loads a cloud and its functions
* Imports utilities
* Methods:
  + \_\_init\_\_(self) - Initializes cloud
  + reinit(self) - Re-draws cloud
  + freezeCloud(self) - Freezes cloud
  + draw(self) - Draws cloud and cloud actions

Utilities

* Image loading method
* Methods:
  + load\_png(self) - Loads images

IV. Tasks and Responsibilities:

GUI:

* Screen setup: Sean
* Background generation: Sean
* Text display: Sean
* Artwork and image loading: Sean
* Music and mp3 loading: George

Backend:

* Kangaroo class: Sean
* Cactus class: George
* Cloud class: George, Sean
* Score calculation: Dillon
* High score database: Sean and Dillon
* Menus: Sean and George
* Controls: Sean

Project Organization:

* Class structure: Sean
* Bug testing: Sean, George and Dillon
* Compiling of various files: Sean

V. Testing:

Testing Process: Try/Except statements were used to test errors in code. When syntax and runtime errors were fixed, playtesting began. All of the controls were tested to make sure no crashes occurred.