I. Cover Page

Group Members:

Richard Park

YongHoon Nathan Kwon

Matthew Van Gorden

Title:

BU Invasion

Course:

CS110, Summer 2016

II. Top Level Design

Classes:

Title class **[View]**

Image, Music

Background class **[View]**

Image

Enemy class **[Model]**

Image, Speed, Position

reset_pos( ), update( )

Rocket class **[Model]**

Image, Position

update( )

Bullet class **[Model]**

Image, Speed, Position

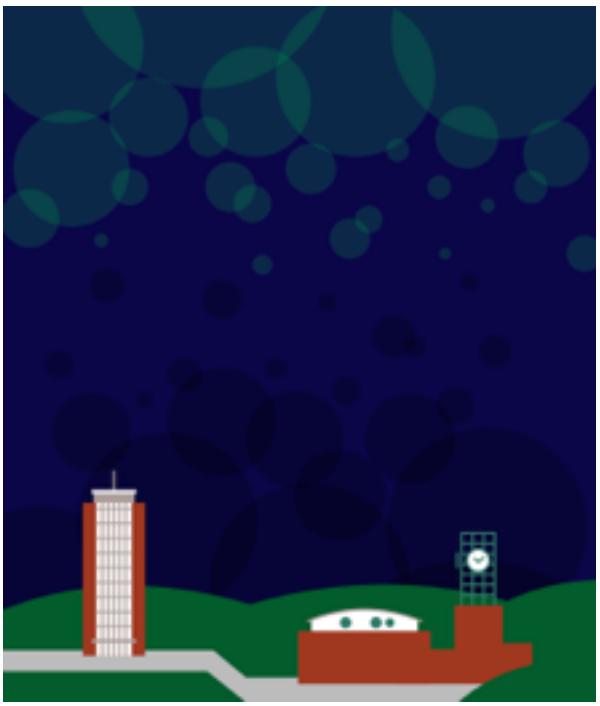
update( )

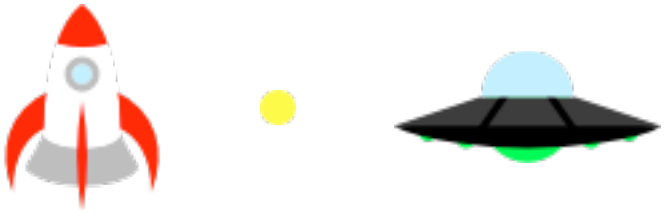
Game class **[Controller]**

Sprite groups, Score, Game over, Screen, Database, Event screen, Score counter, Enemy spawn rate, Enemy accelerator, Firing rate

enemy_respawn( ), handle_event( ), scoreCount( ), run_logic( ), display_frame( )

III. User Interface Design





User Guide:

BU Invasion is a vertically oriented shoot ‘em up game for Python. You play as a rocket and protect Binghamton University from an impending alien invasion. The object of the game is to achieve the highest score possible by destroying the enemies as they descend.

I. Welcome Screen

The game initiates with a welcome screen with “BU Invasion” displayed. When the player clicks with the mouse, the game itself starts.

II. Game

While the game is in play, the rocket moves as the mouse cursor. By clicking on either the mouse or the SPACEBAR, the rocket shoots a bullet straight up. Swarms of enemies will randomly appear at the top of the screen. Shoot the enemies one at a time to increase your score, which is displayed at the bottom of the screen. Be careful not to touch any enemies with the rocket or let any enemies touch the university, or else it’s Game Over. As your score increases, the enemies will gradually speed up. Destroy as many as you can until you lose. You can quit the game at any time by clicking the ESC key or the X in the window.

III. Game Over

Once you get a Game Over, the game will stop and a message will display: “Game Over, click to restart”. The top five scores in the database will also display above. If your score is greater than any of the high scores, it will be added to the list and displayed. Once the player clicks with the mouse, the game will restart.

IV. Tasks and Responsibilities

Brainstorming game logic **[Collaborative]**

Title and Background classes **[Matthew]**

Design images **[Matthew]**

Enemy class **[Richard]**

Find music and SFX **[Richard]**

Rocket, Bullet, and Game classes **[YongHoon]**

Functionality for firing rate, enemy accelerator, score counter, database display **[Collaborative]**

General troubleshooting **[Collaborative]**

V. Testing

Tested GUI for title screen and then initiating the game

Tested functionality for moving rocket with keyboard, but changed to mouse due to reaction speed

Tested collisions

Tested functionality of Game Over screen

Tested firing rate for bullet

Tested random generation for each enemy, tried to increase volume of enemies spawned, but changed to increase speed of enemies due to error in logic

Tested the endpoint for enemies that causes a Game Over (when a UFO touches the university, then the game ends)

Tested the score counter display

Tested the high score display and whole database functionality