BILKENT UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING

CS 319 - Object Oriented Software Engineering

Term Project Iteration 1

FIGHT OR FLIGHT

Final Report

17/03/2018

Group 1D

Mihri Nur Ceren

Adahan Yalçınkaya

Berk Erzin

Emre Sülün

Table of Contents

1. Introduction

In Fight or Flight, there are two stages which are “Flight Stage” and “Fight Stage”. When the game is started, player first encounters “Flight Stage”. This stage is a running part of the game.Throughout the level, player will run. As time goes by, player’s speed will increase. Also, there will be some obstacles. In order to avoid these obstacles, player will control his character by using keyboard. If player dies at some point in this level, the game will end. Player who does not die at “Flight Stage” can move to “Fight Stage”. Instead of running, player stands where he is and encounters waves of orcs, undead and bosses with each wave. At the end of each wave player has two choices that are going back “Flight Stage” or continue fighting. To gain more points, player should fight against waves more.

2. Design changes

Instead of having a MenuManager and an ObjectManager separately as described in our design report, we now have a GameStateManager which is the main controller object as it handles different states of the game. For instance, it handles the transition between the Menu and the Play state of the game.

3. Lessons learnt

During the implementation we learnt the importance of scheduling because we generally could not manage the time successfully. We wrote the big part of the code only in a single day.

4. User’s guide

4.1. System requirements & installation

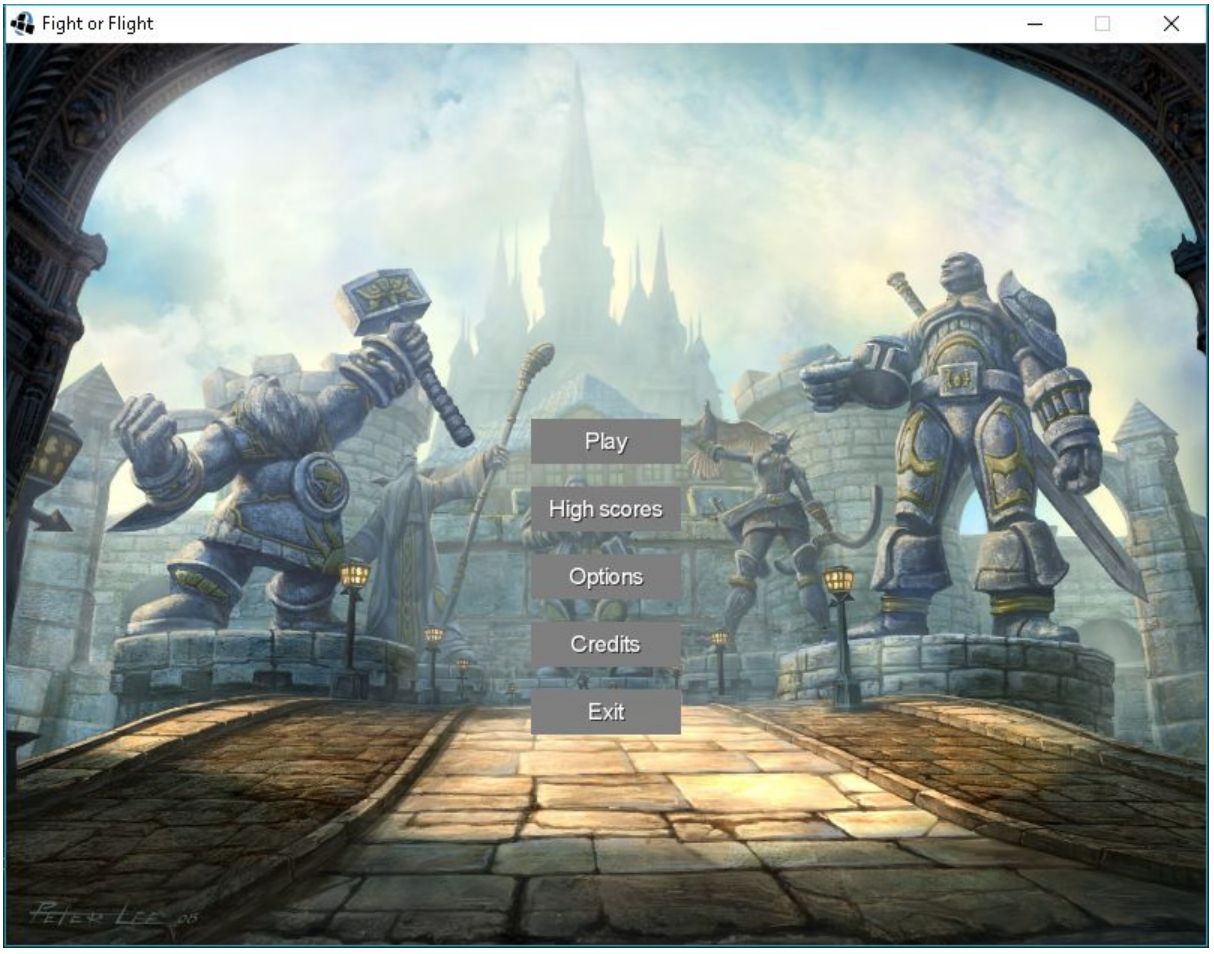
Fight or Flight is a Java based game and as such it needs a JRE(Java Runtime Environment) to be installed in the user’s device. Game is in .jar

format so user only need to download it. Nothing else is needed for an installation and user can run the game as soon as download finishes.

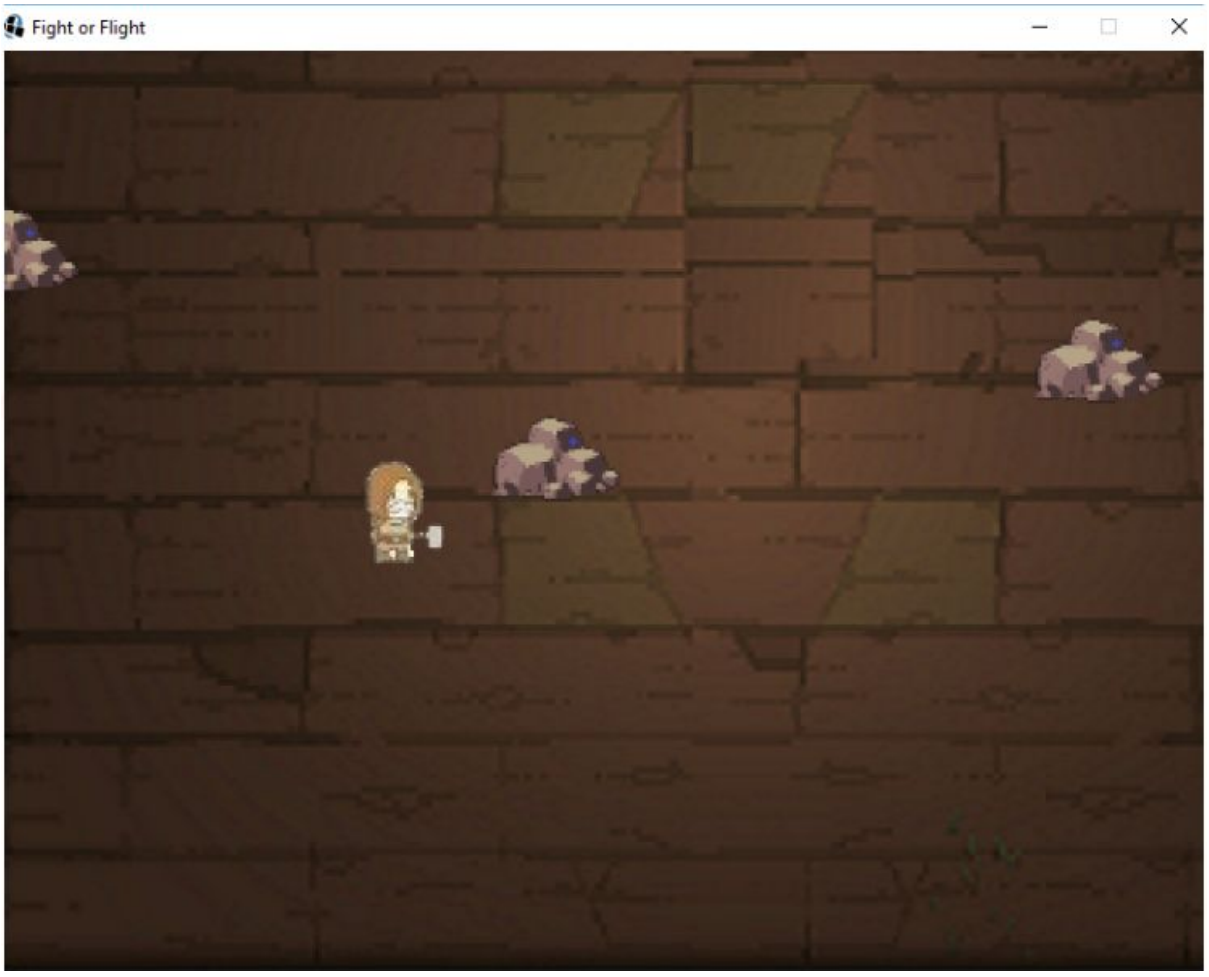
4.2. How to use

Software itself is pretty simple and user-friendly. Menus inside the game can be operated through simple point and click actions issued through a mouse. Game controls require a keyboard to be connected to users device. “W” and “S” keys are bound to player character’s movements. “W” key moves character upwards while “S” key moves it downward.

5. Screenshots



Main menu



Game screen

Credits screen

6. References

Bruegge, B, & Dutoit, A 2014, Object-Oriented Software Engineering : Using UML, Patterns, And Java, n.p.: Boston : Prentice Hall, 2014., Bilkent University Library Catalog (BULC), EBSCOhost, viewed 16 February 2018.

LibGdx tutorial:
http://www.kilobolt.com/zombie-bird-tutorial-flappy-bird-remake.html

Video tutorial:
https://www.youtube.com/playlist?list=PLZm85UZQLd2TPXpUJfDEdWTSgszionbJy