Bilkent University

Object Oriented Software Engineering Project

CS 319 Project: Civilizational Wars



**Final Report**

Project members:

• Fuad Aghazada \_ 21503691

• Seyfullah Yamanoğlu \_ 21400697

• Bayram Muradov \_ 21503664

• Berk Erzin \_ 21201516

Supervisor: Uğur Doğrusöz

Contents:

**1.0** **Introduction**

**2.0** **System Requirements and installation**

**3.0** **Implementation of Project**

**4.0** **User’s Guide**

**5.0** **Expected Development**

**6.0 References**

# **1.0 Introduction**

“Civilizational Wars” has been designed as a 2D sidescroller shooter at its core. 2D side scrollers games are one of the older classic types of gaming genres, incorporating two dimensional gameplay with simple shooter mechanics. In “CW”(abbreviation of “Civilizational Wars”) player takes control of a character fighting through time against a force bent on rewriting the history, along with creatures feeding on the time flux whom time travelling characters generate, one of which is the player character. As player progresses through levels they will also progress through time periods, gaining different weapons and equipment to use against different kinds of enemies each time period offers.

Game interface will be easy to adapt for the players within all groups of ages. The game controls will not only be easy to use as default, but also will be modifiable in the settings of the game. In terms of the interests of the players, the game will have 3 different difficulty types: easy, medium, hard - which could help the player to test his/her gaming skills in different types of difficulties. Another purpose behind the designing this kind of game is also experience the players, especially the ones with same generation with us, the nostalgic atari games like Contra (in terms of map and fighting).

2.0 System Requirements and Installation

In order to play the game, users must have java sdk 1.8 installed on their personal computers. After the installation, user can download the jar file of the game provided in Github repository. After completing both of these steps, user will not face any problem in the means of running the game on their personal computers.

3.0 Implementation of Project

During the implementation of the project we made use of Java swing library and IntelliJ integrated development environment. We also made use of some additional textures that were available on the internet as open source files in order to have a UI, since intended textures and character models couldn’t be prepared in time due to complications. All these used textures can be found on the reference. GameObjectHandler is also a new addition made to handle all game objects like bullets, characters and etc. Many kinds of levels, weapons and enemies are yet to be implemented due to time constraints, which are left to second iteration.

4.0 User’s Guide

When the game opens main menu will be the first thing that greets the players. From here players can start a new game using the “Play Game” option, which starts a new instance of the game from the first level.

Below that is “Load Game” option which allows player load a previously saved files. Loading a saved file starts the game from the start of the level that players saved their game in.

After these options come the “Settings” option which directs user to the settings menu. Settings menu allows a player to modify the sound volume of the game and the difficulty level. It also allows player to redefine key bindings for ingame controls. Quitting this menu returns player back to their previous menu.

Following is the “Help” option which brings up a text based tutorial of the game, informing player of the game mechanics and the buttons for default control.

Last two options on this menu are “About” and “Exit”. “About” choice brings up information about the game and developers, similar to a credits. “Exit” option closes the game, returning the player back to desktop.

5.0 Expected Development

For the first iteration we mainly focused on the core functionalities of our project such as creating the TileMap and giving basic functionalities to the character and enemy such as firing a weapon, moving around the TileMap and getting killed after certain amount of damage. In the current implementation, player will not be able to save or load the game since we did not have enough time to implement these functionalities. Furthermore, since the game is not in its final stage yet, player won’t be able to make use of settings panel. Apart from the functionality part of our project, some UI components such as different Tiles, ModernFighter, Robot, ClassicSoldier, ModernSoldier are missing since we couldn’t prepare different textures because of the time limit as such currently using placeholders until better replacements can be drawn/found. As expected development, we are as a group planning to completely implement all functionalities and also complete the missing UI components in order to have more diverse system.

6.0 References

- Soldier,Tanks\_pack, Aliens\_pack, Letter\_tiles, platformer-pack-redux-360-assets

<http://kenney.nl/assets?q=2d>

-Laser Gun

<http://fallout.wikia.com/wiki/Pulse_gun_(Fallout:_New_Vegas)?file=Pulse_gun.png>

-Rifle

<http://www.imfdb.org/wiki/M16_rifle_series>