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 Design Overview**

**6.0 Improvement Summary**

**7.0** **Expected Development**

**8.0 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. We also have designed some textures by ourselves by using editor tools like Adobe Photoshop and Illustrator. All these used textures and tools can be found on the reference page. GameObjectHandler is also a new addition made to handle all game objects like bullets, characters and etc. Comparing with first iteration, we have made significant additions and optimizations in the second iteration. For example, the second iteration version of the game can be played by 2 players simultaneously. Furthermore, in the second iteration we have implemented the SurpriseBox class that includes either health or lives for the player in order to make the game play more enjoyable. Apart from these, we have implemented Boss classes that contain the enemy bosses for the end of each level to make the game more challenging.

**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 by choosing either “Play Single Player” or “Play Multiplayer” buttons. After choosing their options they will be directed to game screen in which the game will start according to their chosen option.

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.

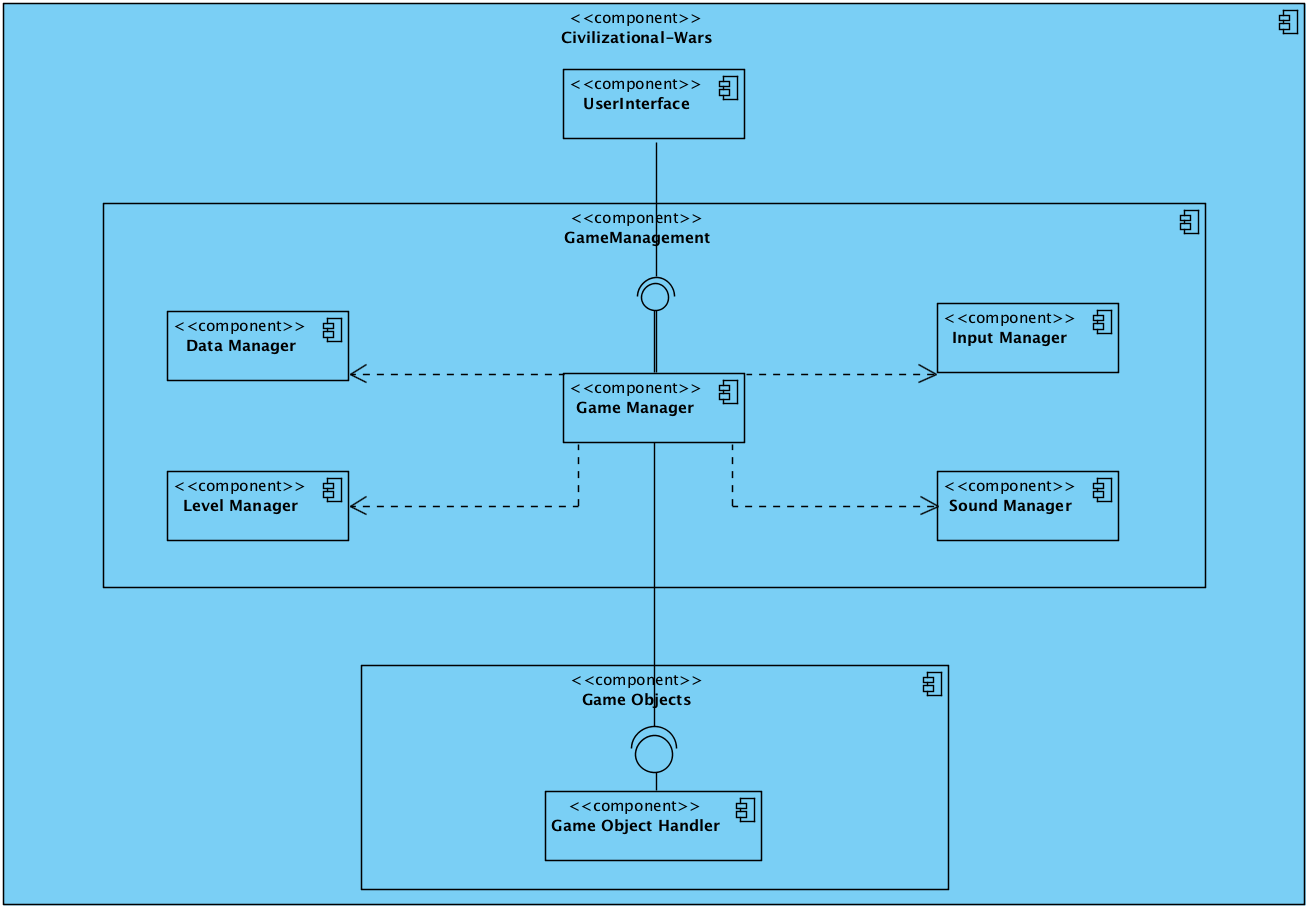
Following is the “Help” option which brings up a text based tutorial of the game, informing player of the game mechanics and the buttons for default control.

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.

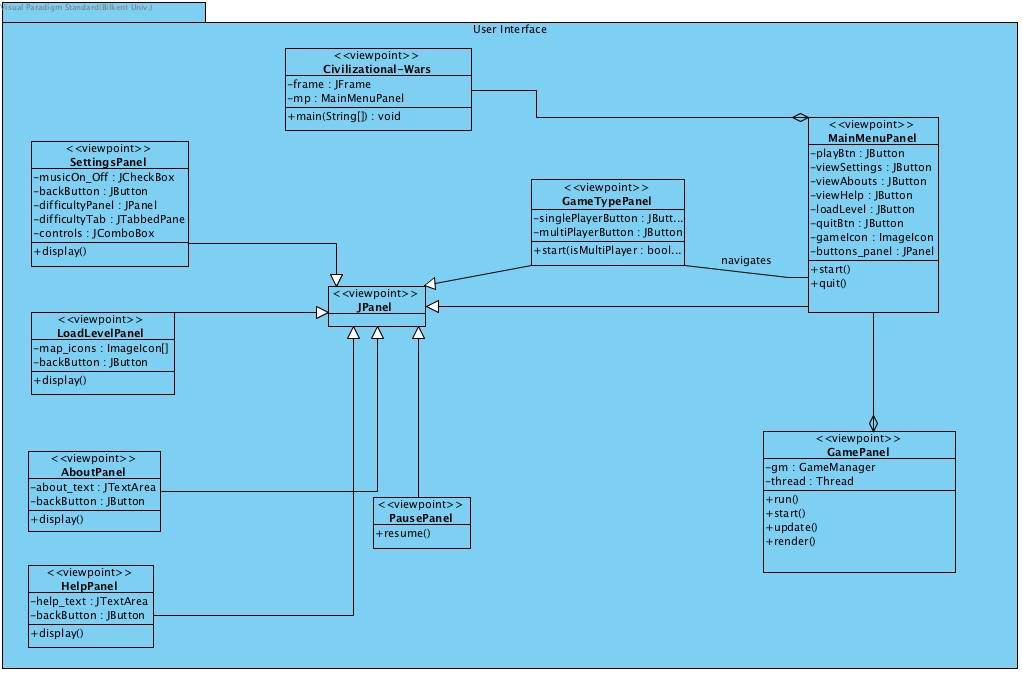
**5.0** Design Overview

We have made use of MVC design pattern during our implementation of the project.

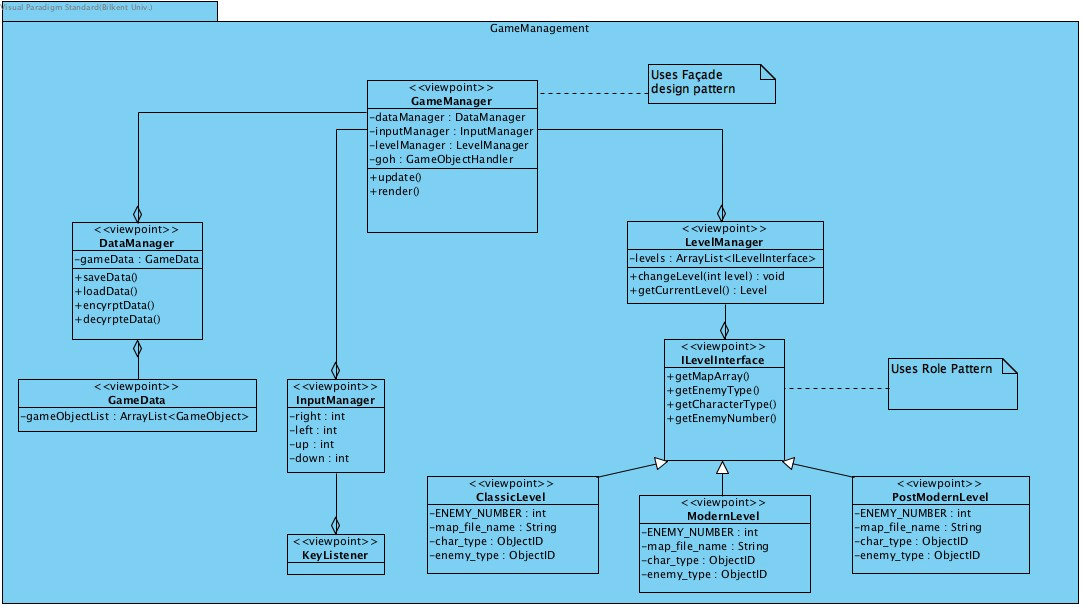
*General view of subsystems:*



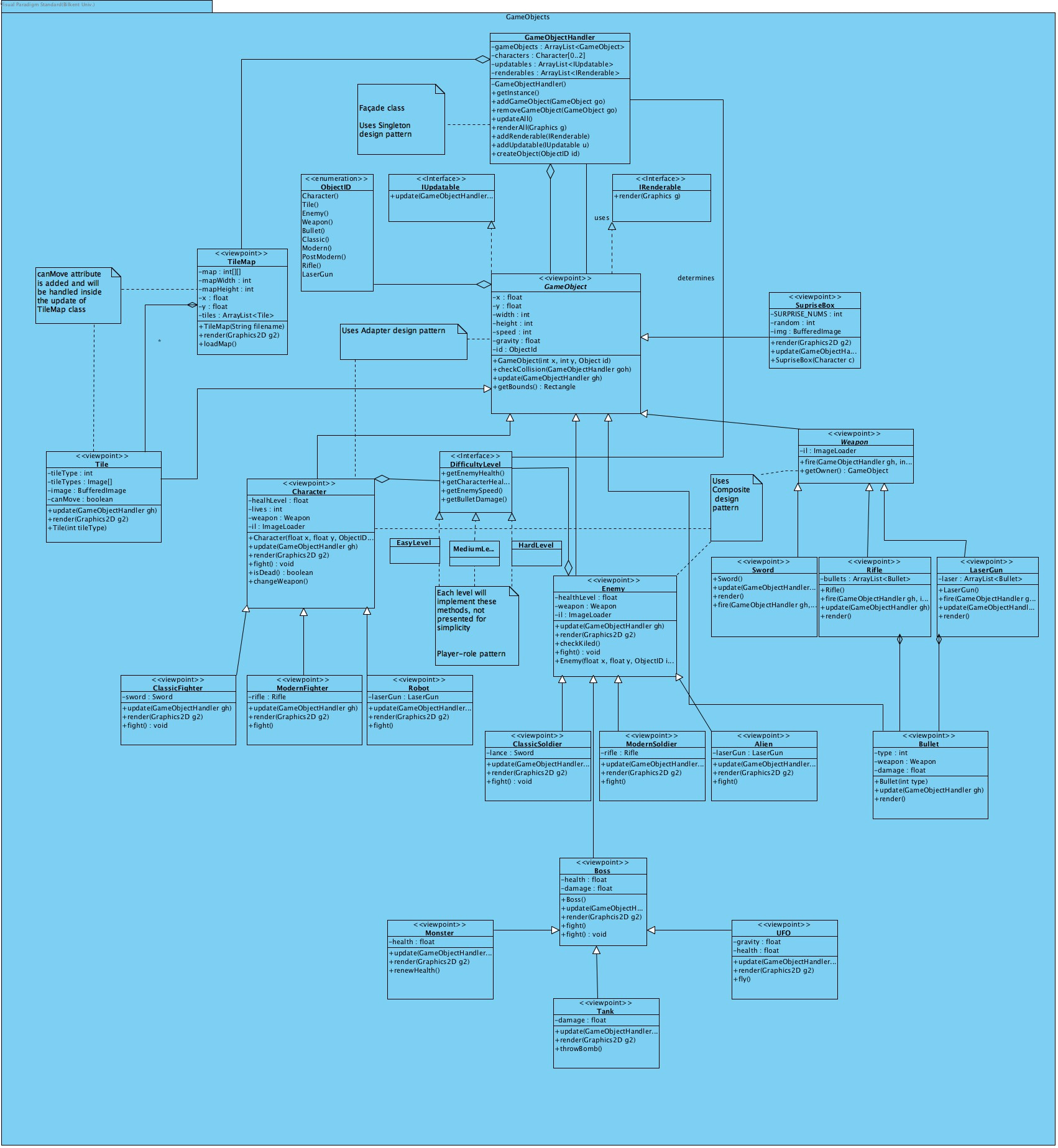
*UI subsystem:*



*GameManagement subsystem:*



*GameObjects subsystem:*



**6.0** Improvement Summary

Unlike the first iteration of the game, we now have additional classes and functionalities in our implementation. Additions include SurpriseBox (Screenshot 1) class that provides player with box which includes either an extra live or a full health bar, DataManager class that saves the game and Boss (Screenshot 2) classes which contain the boss enemies for the end of each level. Furthermore, our game now can be played by two users simultaneously (Screenshot 3). We also added some additional UI components for the background of each level. Apart from these, we diversified the map designs for each level. Moreover, player now can change the difficulty level of the game by choosing from given three difficulties that include easy, medium, hard.

*Screenshot 1:*



*Screenshot 2:*



*Screenshot 3:*



**7.0** Expected Development

Since it was a project that needed to be implemented in a limited time period, we mainly focused on completing the functionalities that we have promised at the begging of the semester. As a future development project can be extended by adding new levels, characters and different fighting techniques. Furthermore, a server can be made in order to make multiplayer version accessible to the people who aren’t behind the same computer. Apart from these, a simple chat can be added to the game in order to create a communication between the players who aren’t in the same location and want to play the multiplayer version.

**8.0** References

-Bosses, ClassicSoldier

[www.opengameart.org](http://www.opengameart.org)

- 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>