Bilkent University

Department of Computer Engineering

CS 319 - Object-Oriented Software Engineering

Design Report

**Man of War**

**Group 1-D**

* Berat Tuna Karlı
* Ceren Uysal
* Serdar Erkal
* Selim Fırat Yılmaz

Contents

[1. Introduction 3](#_Toc496354683)

[1.1 Purpose of the System 3](#_Toc496354684)

[1.2 Design Goals 3](#_Toc496354685)

[User Criteria: 4](#_Toc496354686)

[Maintenance Criteria: 4](#_Toc496354687)

[Performance Criteria: 5](#_Toc496354688)

[Trade Off: 5](#_Toc496354689)

[1.3 Definitions, acronyms, and abbreviations 6](#_Toc496354690)

[1.4 References 6](#_Toc496354691)

[1.5 Overview 7](#_Toc496354692)

# Introduction

## Purpose of the System

Man of War is a 2d quick time event1 based adventure game. The reason why we chose this project is because we think that we could reflect the basic principle of OOP structure better and this is a great chance to improve our skills of how to use and design object-oriented programming idea. In addition to this, we would like to study about the process of planning a Greenfield project rather than simply creating a game. We will develop this project in Java because we all have experience in Java language, hence we believe that we could use Java more efficiently. Development platform of this project would be IntelliJ IDE. We are planning to deploy the Final project to .JAR through Maven2.

## Design Goals

We give priority to design goals before implementation, because all of the implementation is based on decided contents from the design report. We did not directly start writing codes of the game, because if we encounter any problems related with coding we would stuck at that point and we could not progress for the next steps of the project. Therefore we decided to reveal all of our specialties clearly first. It will be given brief information about each of the elements below.

### User Criteria:

Ease of Use: The game has known logic with many old games (ex. Pokemon FireRed). For this reason, our game appeals to any player with every age and gender. There will be 3 different levels of the game, this will provide various maps to avoid boredom and to increase entertainment. Player will only use arrow keys which will make game play easier.

Ease of learning: There will be an option in main menu in order to provide a tutorial for the players who want to learn how to play the game before they play. The logic behind the game is not only easy to learn but also familiar for everybody.

### Maintenance Criteria:

Extendibility: It is essential to make updates to this types of adventure game to capture players’ attention. This feature increases the chance of appealing other players desire to play who have not played our game before. There could be new features like different types of potions, enemies and characters. Also, player might be able to change the background sound of the game. There would be some additional new levels which could make the game more challenging for the expert players. Moreover, we might have multiplayer feature. That feature would also increase entertainment of the game. In this way, the game would become more challenging and the competition between the players would increase.

Portability: Our game will be implemented in Java so that we might use the opportunity of JVM which will provide us to play the game on different platforms. This feature supports freedom for both players and developers.

Understandability-User friendliness: The game has designed to have a simple underlying

the logic. In addition to this, there is a tutorial in the main menu for the players who will just start playing the game. The background sound and game sound are optional since player make changes with respect to their request.

### Performance Criteria:

Efficiency: The efficiency of our game will be based on the efficiency of our codes which will build up the game itself. We are planning to use optimized number of components to prevent extra cost of memory for computer and extra time for developer. Also we prefer to abstain from using CPU more than it is needed.

### Trade Off:

Performance vs Memory: We are planning to make our game as smooth as possible since we would like to make the game more attractive and interesting but we try to avoid using memory too much, because this would occupy the memory more than it is needed which could make the game a lot slower. Thus, this might cause the game becoming less interesting and challenging. Also, several players might stop playing the game which will probably cause loosing more potential players. In addition, we chose pixel graphics to occupy less space in the memory, however nowadays pixel graphics are very popular and widely used in many games. Players prefer the games with pixel graphics because it reminds them the past which is somehow nostalgic and familiar.

User-friendliness vs Functionality: Our game is designed to involve many useful functionalities but we also want our game to be user-friendly. Hence we try to make optimal choices for both sections with respect to the player and his/ her desires. This is why, we give importance to both functionality and user-friendliness equally. For instance, there are different types of enemies and different levels with different difficulties that also make the game more entertaining, however we try to explain all of these functionalities with the help of the tutorial feature. There is only one type of main attack and defence function, we choose QTE because it is simple, effective and different from other usual games. Also, there is one main warrior character but later we might have some additional characters.

## Definitions, acronyms, and abbreviations

Maven [1] : A software project management and comprehension tool. Based on the concept of a project object model (POM).

MVC [2] : Model View Controller

JDK [3] : Java Development Kit

QTE [4] : Quick Time Event. A method of [context-sensitive](https://en.0wikipedia.org/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvQ29udGV4dC1zZW5zaXRpdmVfdXNlcl9pbnRlcmZhY2U) [gameplay](https://en.0wikipedia.org/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvR2FtZXBsYXk) in which the player performs actions on the control device shortly after the appearance of an on-screen instruction.

JRE [5] : Java Runtime Environment

Green Field Project [6] : Basically, not taking previous work into account when starting a new project. Having Analysis and Design parts completed before starting the implementation of the project.

## References

[1] <https://maven.apache.org/>

[2] *Object-Oriented Software Engineering, Using UML, Patterns, and Java, 3rd Edition*, by Bernd Bruegge and Allen H. Dutoit, Prentice-Hall, 2010, ISBN-10: 0136066836.

[3] https://www.techopedia.com/definition/5594/java-development-kit-jdk

[4] <https://www.giantbomb.com/quick-time-event/3015-6/>

[5] <https://techterms.com/definition/jre>

[6]<https://workplace.stackexchange.com/questions/17293/what-is-a-greenfield-project-in-it>

## Overview

In the Introduction section, the identification of the main goals of our game is explained. The goals of the game includes using OPP well, learning how to design well, before starting a new project. We learned that the documentation of a project is essential for the people who could possibly join later to the project. Thus, analysis and design stages of the project are more important than simply coding. Also, we would like to create interesting game for players. We also mention the design goals of the game. Such as ease of use, ease of learning, extendibility, portability, understandability-user friendliness, efficiency and so forth. The trade offs are also discussed to compare between the contrasting issues about the games like performance vs memory and user-friendliness vs functionality. We try to have equal and balanced distinction between the comparisons to obtain the optimal results.