  
  
  
  
CS 319 - Object-Oriented Software Engineering  
Design Report  
  
  
Punch For Glory  
Group 1-C

Taner Baygün

Özgür Can Erdoğan  
Saba Betigül Şahinkaya  
Ömer Berk Uçar

**1. Introduction**

**1.1 Purpose of the System**

Punch for Glory is a system aiming to entertain user with well-designed gameplay which includes difficulty levels and some power-ups. Punch for Glory looks very poor in terms of graphics. However, gameplay of Punch for Glory is designed to maximize the sense of achievement for player. Punch for Glory has a user-friendly interface which reduces the complexity of the gameplay. Opponent boxer’s levels are low at the beginning of the game. When, the player increase his position in the league, his opponents will be more difficult. The reason behind it is, increasing the user’s game experience.

**1.2 Design Goals**

* **Efficiency:** The system is going to be responsive and able to run with high performance. The game will run at least 30 fps in order to provide smoothness. This is one of the most important design goal because performance is highly related with the user’s gameplay experience. In order to get optimum performance, memory usage of the game will be minimized, and also, each objects gets allocated memory in order do their own tasks. This will boost the performance of the game significantly because it will decrease the workload.
* **Reliability:** The game will be bug-free and stable in the boundary conditions. The game should not crash at any time. To get this goal, the testing procedures will involve. The testing procedure will continue each stage of the development of the game. Therefore, boundary conditions will be tested carefully in order to avoid unexpected game crashes.
* **Adaptability:** Java is one of the most well-known programing language which provides cross-platform portability. Our game can work all JRE installed platforms, therefore, user should not worry about the operating system requirements. Using Java will cause to sacrifice performance but its adaptability obscures the performance problems.
* **Usability:** One of the most important thing about our game is, it should be easy to play. Therefore, it affects our design goals too. This makes the game more user-friendly. Therefore, the game will be designed such that user can easily understand the concept of the game and user can easily interact with it. However, user-friendly interface doesn’t mean that the gameplay easier. If user-friendly interface and gameplay is easier, the player can be bored immediately.
* **Extensibility:** Object oriented architecture of the game enables system changes without causing any bugs or harming other classes. Modifications are easier by using object oriented architecture. For example, we can add new opponents or change the opponents without having to modify anything in other classes. Therefore, object oriented architecture minimizes the possibility to cause some malfunctions in other classes.

**Tradeoffs**

* **Efficiency – Reusability:** Reusability is not the main concern for our game. Because, our system is not planned to integrate to other systems. Therefore, the classes are designed specifically for our game. Also, it prevents us to being too complex in class implementation. The main design approach is being efficient in our game.
* **Functionality – Usability:** The most important thing for a game, its usability. Because, costumers are our main targets and they should understand the game easily. Therefore, functionality of the game should be basic.We focused on usability rather than functionality. Because our purpose is entertaining the people. The game has simple user interface and easy gameplay. Thus, they can enjoy while playing game rather than suffering.
* **Space – Speed:** The implementation details of our game is planned to be fast. We don’t merge any object to earn memory space. Reason behind it is, making game fast as possible. If we consider our game, there will be plenty of objects that use memory. However, we try to keep our game simple as possible, therefore, memory problem is not the most important one especially in this era. However, if the speed of our game is not enough to provide 30 fps, users don’t like the gameplay. Therefore it should be fast.

**1.3 Definitions, acronyms, and abbreviations**

**Cross-platform:** Cross-platform refers to ability of software to run in same way on different platforms such as Microsoft Windows, Linux and Mac OS X.

**JRE:** Java execution environment is termed as the Java Runtime Environment (JRE). All systems need Java Runtime Environment (JRE) in order to execute the projects which are developed in Java.

**Fps:** Abbreviation of “frames per second”. Fps represents the number of graphical layouts can be prepared by the system each second.

**Boundary conditions:** Conditions of the system which may generate run-time errors. They are exceptional cases according to the normal flow of the program. These conditions must be handled carefully for robustness of the system.