Contents

1. Introduction
- **1.1 Purpose**
- **1.2 Acronyms, and Abbreviations**
- **1.3 Definitions**
- **1.4 Contents**
- **1.5 Project Scope**

2. General Description
- **2.1 Overview**
- **2.2 Project Perspective**
- **2.3 Objectives**
- **2.4 Overview of Data Requirements**

3. Functional Requirements
- **3.1 User database**
- **3.2 Administrator database**
- **3.3 Performance Requirements**

4. Non Functional Requirements
- **4.1 Software Interface:**
- **4.2 System Interfaces**
- **4.3 User Interfaces**
- **4.4 Hardware Interfaces**

5. Diagrams and Flow charts
- **5.1 Use Case Diagrams**
- **5.2 Data Flow Diagrams**
- **5.3 ER Diagram**

6. Source Code
- **6.1 Login/Signup Page**
- **6.2 Home Page**
- **6.3 MoveTheBox Game**
- **6.4 Snake Xenzia**
- **6.5 Quiz game**
  - **6.5.1 Quiz.html**
  - **6.5.2 style.css**
  - **6.5.3 Javascript.js**
  - **6.6 Psychic Game**

7. Screenshots

8. Conclusion

1. INTRODUCTION

1.1 Purpose of this Document

This SRS describes the function and performance requirements of our class project. These include an overview of the project description, functional requirements of systems the project will run on, and characteristics of target users.

1.2 Acronyms, and Abbreviations

Project Gamezy : The code name for our project.

1.3 Definitions

The **online Gaming portal** aims toward building online games for the players who like challenges. The product is independent and mostly self-contained. All the internal designs belongs to our project group. The purpose of the document is to collect and analyze all assorted ideas that we have come up to define this gaming system, its requirements with respect to players.

The effort here is made to build interactive and interesting game namely,

1. **Move the box**: In this game basically there is an object i.e. ‘red’ box which has to be passed through the obstacles to gain a high score. The main objective here is to save the box from getting smashed from the obstacles. The box can be moved with the help of the arrow key to 4 directions i.e. up,down,left and right. The more time a user can avoid the obstacles and moves ahead, the more score it makes.
2. **Snake Xenzia**: In this game the snake is to move around and to increase the score,the snake must take in the food and should avoid hitting itself and the snake grows in size each time it eats a food. Each time the snake eats, the score gets increased by 1.
3. **Quiz Time**: It tests the knowledge of the player by seeking some basic questions and scores are given as per the number of correct answers.
4. **Psychic game**: In this game the computer generates a random alphabet which has to be guessed by the user to win the game and 8 chances are to be offered each time. If the user gets the correct answer within 8 chances offered to him, they wins. Otherwise, They can try again with their luck.

1.4 Contents

This project is having

- Home Pages
- Sign Up form
- A Login Page
- 4 Gaming Pages

1.5 Project Scope

The website gives the complete access to the games depending on their class type and .It will require a good internet connection to play games.

The gamer may purchase many games to play in offline mode and also, they can go for free offline games which won't need to be purchased.

The website will send a gamer card to the user after verifying all his personal details and payment details. The company allows you to see the profiles of other gamer, make teams, join tournaments, see gaming videos, put reviews, etc. The achievements of the gamer must be properly recorded so that gamer must not require to play a big game again from the beginning.

2. General Description

2.1 Overview

Back in the days before Graphical User Interfaces became popular, a number of entirely text-based games were written, one game for a user was provided. There in the game, the user had to install a lot of interconnected games by conducting a textual dialogue with the machine.

Moreover in those games, the machines would little describe of what the user could "see", and the user typed in various commands which were a simple subset of English language sentences. Each command contained one of a limited number of verbs (e.g. go, open, hit, look, grasp, throw), together with an object and/or direction. Also if during the game, if the user would like to switch between the games then the machine had to be restarted again to do so.

Also the difficulty of the game arose from remembering what your scores were, and where you had been, as the computer offered no scores or records. Designing

your own score helped, but was not easy, as the inter connectivity of the games was often a little complicated.

2.2 Project Perspective

This game will display the game to screen and enable the player to interact with the game through mouse and keyboard.

2.3 Objectives

So here our objectives are

1. Maintaining and regularly updating the details of games that were available to user.
2. List of all the games at your home page.
3. Displaying the scores continuously and showing the levels.
4. Maintaining database of user login details.

2.4 Overview of Data Requirements

Other than input with Mouse and Keyboard there are no other data requirements in the game.

3. FUNCTIONAL REQUIREMENTS:-

Functional requirements define the specific functions that the system performs, along with the data operated on by the functions. The functional requirements are presented in scenarios that depict an operational system from the perspective of its end users.

A user should be able to **register and login** to the system through this application, mention his/her required **username -Password**& he should get their information which can be viewed and update it whenever required.

R.1: User database

Description: User first have to register and then they can login into their account. There they can view different games after a correct login.

R.1.1: Registration

Input- User provide necessary details like name, email-id, password etc.

Output- Now user can able to login their id.

R.1.2: Login

Input- User provide their email-id and password.

Output- User able to login and check on various games.

R.1.3: Featured games

Input- Click on the icon of any game of their interest.

Output- Open page successfully and can start to play

R.2: Administrator database

Description: Admin first have to login their account. He manage game details, maintain availability & user database.

R.2.1: Login-

Input- Admin provide their email-id & password.

Output- Login successful. Admin able to check, update & maintain game details.

R.2.2: Game database-

Input- Admin upload different game details for users and can update the game and can include new games also.

Output- Database stored successfully.

3.3 Performance Requirements

- 56k and above internet connection
- high response time (low latency)
- runs at 30 FPS

4. NON-FUNCTIONAL REQUIREMENTS:-

Non-functional requirements address aspects of the system other than the specific functions it performs. These aspects include system performance, costs, and such general system characteristics as reliability, security, and portability. The non-functional requirements also address aspects of the system development process and operational personnel. It includes the following:

- ✓ The system shall be user friendly and consistent.
- ✓ We get the response within seconds.
- ✓ The system shall provide attractive graphical interface for the user.
- ✓ The system shall allow developer access to installed environment.
- ✓ The system shall be for the target customer base i.e. for the players.

Software Interface:

The system is a web based application, in the front end HTML, CSS and JavaScript are used.

System Interfaces:

The web application developed is a standalone engine and all the components can execute on Windows, Linux and Mac OS.

User Interfaces:

The user interface of the game is going to be achieved through the web browsers like Chrome,Firefox or Microsoft Edge with JavaScript enabled. Interface is user friendly and very easy to get used-to. Game screen will be embedded into web browser. There will be score box, timer, information table and some buttons to move and manipulate agents and to start the game.

Hardware Interfaces :

There is no constraint on which kind of hardware must be used. There are common hardware devices that are enough to interact with the game. These are

- **Monitor screen**: Screen provides visual information to user.
- **Keyboard**: Keyboard provides user to communicate with other users and to manipulate the agents inside the game.
- **Processor** capable enough to run a supported web browser.
- **Mouse** : Provides interface for the user and the web portal.

• **Internet Connection :** A good working internet connection is required.

5. Diagrams and Flow Charts

5.1. Use Case Diagrams :

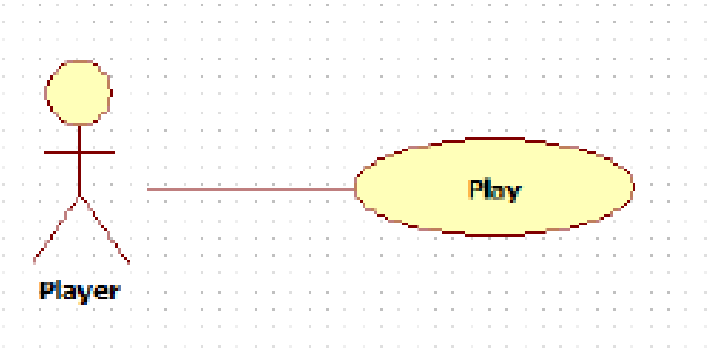


Figure 1 – Play Use Case

Used in ingame screen, this movement triggered by touch movements.

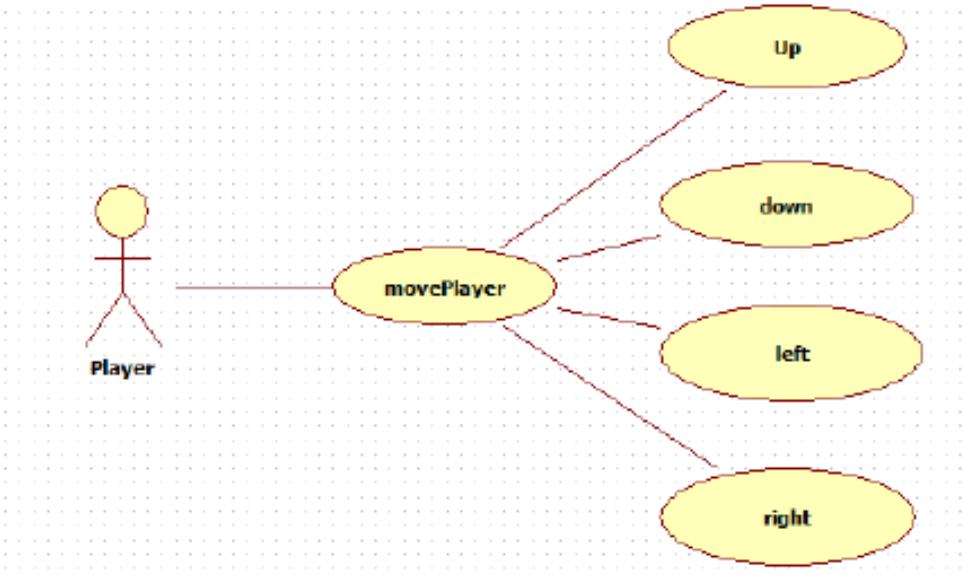


Figure 2 – movePlayer Use Case

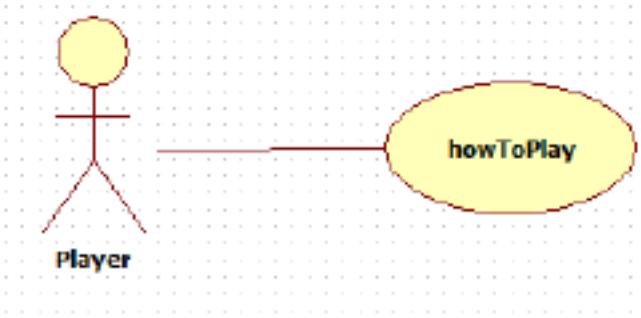


Figure 3 : How to play use case

Figure 4 : Exit use case

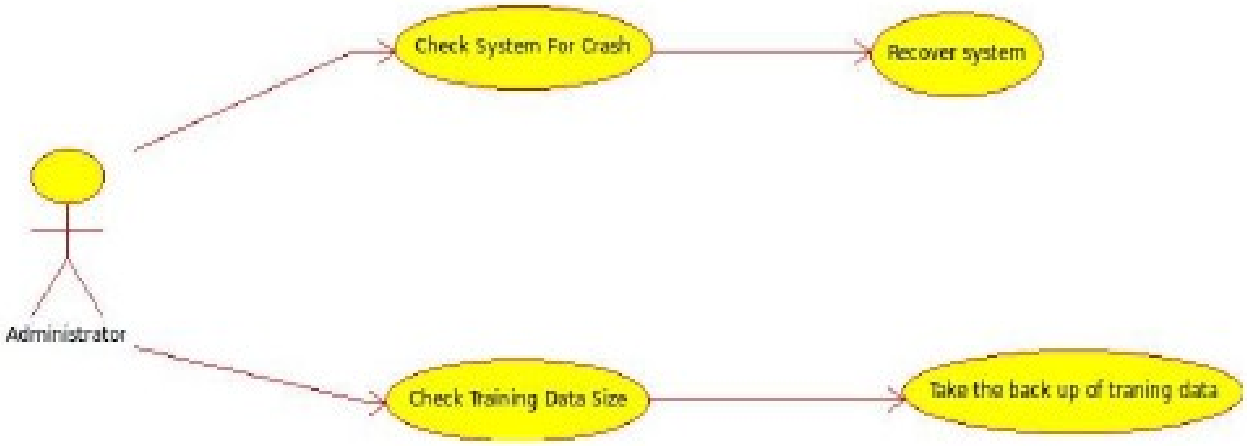
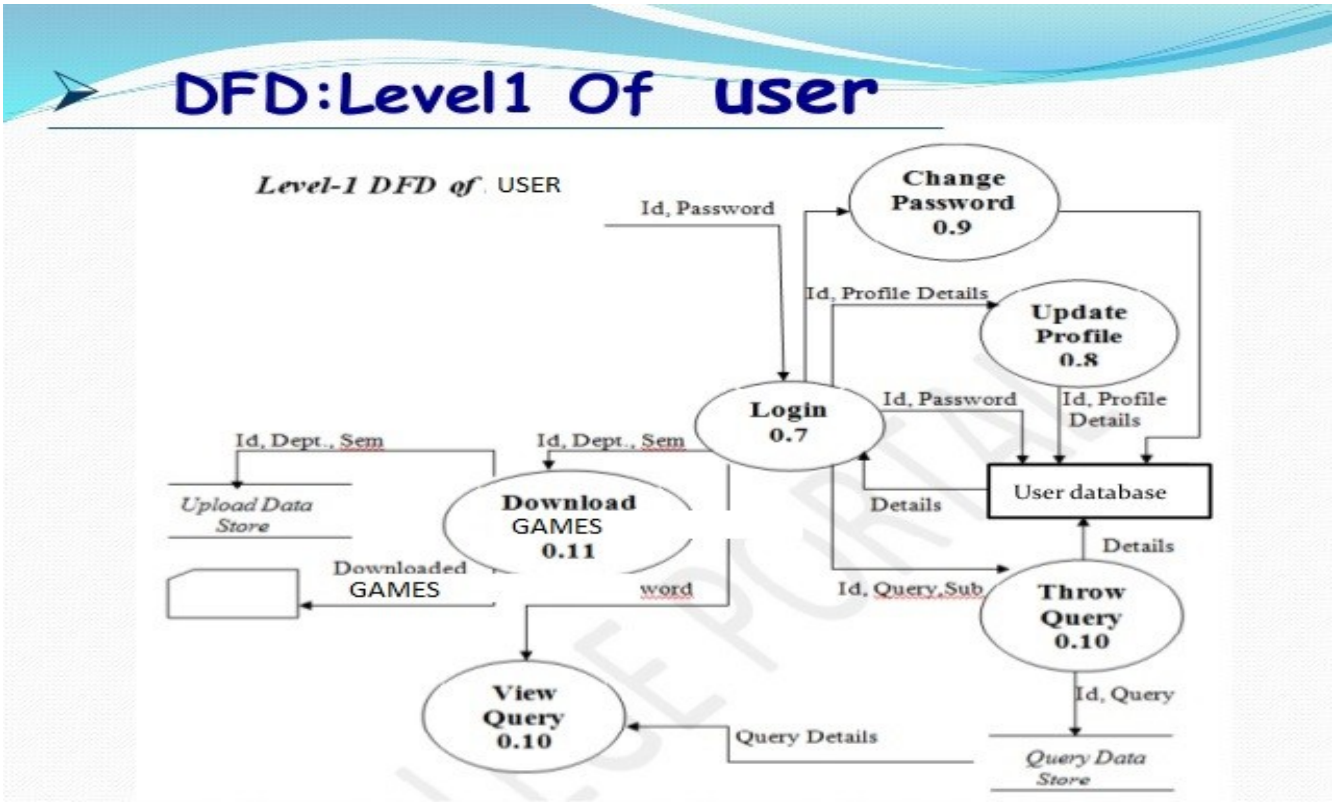


Figure 5 : Administrator use case

5.2 Data Flow Diagram : Level 1



5.3 Entity relation Diagram