

SOFTWARE REQUIREMENTS SPECIFICATION

ON

PUBG GAMING APP

Submitted by
PRATHAM THAKUR
12114778



LOVELY
PROFESSIONAL
UNIVERSITY

School of Computer Science

LOVELY PROFESSIONAL UNIVERSITY

Phagwara Punjab.

1. INTRODUCTION

- 1.1 Purpose
- 1.2 Scope
- 1.3 Abbreviations
- 1.4 Goals and challenges
- 1.5 Overview

2. OVERALL DESCRIPTIONS

- 2.1 Product Perspective
- 2.2 Product functions
- 2.3 User characteristics
- 2.4 Constraints

3. SPECIFIC REQUIREMENTS

3.1 DETAILED REQUIREMENTS

- 3.1.1 System interfaces
- 3.1.2 User interfaces
- 3.1.3 Hardware interfaces
- 3.1.4 Software interfaces
- 3.1.5 Communications interfaces
- 3.1.6 Operations

3.2 FUNCTIONAL REQUIREMENTS

- 3.2.2 Menu Requirements
- 3.2.3 Game Flow Requirements

3.3 PERFORMANCE REQUIREMENTS

- 3.3.1 Static requirements
- 3.3.2 Dynamic requirements

3.4 ATTRIBUTES

3.5 OTHER REQUIREMENTS

4. USE CASES

- 4.1. Game play use case
- 4.2. Login use case
- 4.3. Chat use case

1. INTRODUCTION

1.1 PURPOSE

Player Unknown Battle Grounds will be an online multi-player battle royal game, up to one hundred users parachute onto island and scavenge for weapons and equipment to kill others while avoiding getting killed themselves. The available safe area of the game decreases in size over time, directing surviving users into tighter areas to force encounters. The last player or team standing wins the round.

1.2 SCOPE

This specification establishes the functional, performance and development requirements for release of PUBG Mobile. This document will be used by the end-user, developer and tester of the game.

1.3 ABBREVIATIONS

DB: Database

AI: Artificial Intelligence

GUI: Graphical User Interface

Http: Hypertext Transfer Protocol

API: Application programming interface

1.4 GOALS AND CHALLENGES

- Maintaining the game-state consistent
- Ordering the events
- Increasing the interactive responsiveness
- Providing realistic rendering 3D Components
- Integrating artificial intelligence into the game
- Displaying animations
- Scheduling computation across users
- Providing authentication / authorization mechanisms
- Implementing a cheat-proof design.

1.5 OVERVIEW

Section 1 Identifies the scope of this document, the purpose of the game and list of the definitions.

Section 2 describes the overall description of the project.

Section 3 identifies specific and detailed requirements of the game. There will be more specific details and information about the project context, general requirements and environment.

2. OVERALL DESCRIPTION

2.1. PRODUCT PERSPECTIVE

PUGB will be a player versus player shooter game in which up to one hundred users fight in a battle royal, a type of large scale last man standing death match where users fight to remain last alive. Users can choose to enter the match solo, duo or with a small team of up to four people either with random matchmaking or with the friends. The last person or team alive wins the match. Each match shall start with users parachuting from a plane onto map (Island), with areas of approximately 8x8 kilometers (5.0 x 5.0 mi). The plane's flight path across the map shall vary with each round, requiring users to quickly determine the best time to eject and parachute to the ground. Users start with no gear beyond customized clothing selections which do not affect game play.

Once they land, users can search buildings, towns and other sites to find weapons, vehicles, armor and other equipment. These items will be procedurally distributed throughout the map at the start of a match. Killed users can be looted to acquire their gear as well. Every few minutes, the playable area of the map will begin to shrink down towards a random location, with any player caught outside the safe zone taking damage incrementally, and eventually being eliminated if the safe zone

is not entered in time; in game, the user can see the boundary as a shimmering blue wall that contract over time. On average, a full round takes no more than 30 minutes.

2.2. PRODUCT FUNCTIONS

Register: The user must register once in order to get access to the game; after registering, a unique username and password will be provided and will allow the logging in of the registered user.

Log-in: A registered user has to enter his/her unique username and password combination in order access the game.

Log-Out: User logs out in order to let another user to register or log-in.

Settings: An option allowing the user to modify the settings for the controls and graphics.

Exit: User exits the program.

2.3. USER CHARACTERISTICS

- User must be above 14 years.
- All the users must have minimal technical expertise (need to know how to use mobile and game control).

- User needs to be very familiar and comprehend the game rules.

2.4. CONSTRAINTS

TIME CONSTRAINTS: There exist strict deadlines for each phase of the process, so the constraints of the meeting the deadlines is of utmost concern.

ATTRIBUTE CONSTRAINTS: As a common denominator in the game programming, it is best to use C++ as the main programming language as the server side component. The components to be integrated, hence the overall design is considered around the core of C++ component.

TOOL CONSTRAINTS: Using Graphics and AI engines leads to limitations within the corresponding domain according to the implantation of functionalities. While engines reduce the burden of low level programming, they introduce constraints to the capabilities of the project.

NETWORK CONNECTIONS: Since the aim is to be able to support a great number of users, the effective transmission of game data is exposed to network connection speed constraints, which greatly affects the project design.

GRAPHICS CONSTRAINTS: A good frame rate is to be guaranteed in order to have a decent display and game play. This places constraints of to what extent the graphics module complexity can be pushed.

3. SPECIFIC REQUIREMENTS

3. DETAILED REQUIREMENTS

3.1.1 SYSTEM INTERFACE

The game integrates two internal systems to provide functionality.

CLIENT: The game software has an interface to user's client to receive user input and process the selection for the game.

SERVER: The game software has an interface to the network in order to transmit information.

3.1.2 USER INTERFACE

It includes an interface resembling a common dashboard of the player.

3.1.3 HARDWARE INTERFACE

The system has no hardware interface requirements.

3.1.4 SOFTWARE INTERFACE

The required software products for PUBG Mobile are

* Game engine IDE platform called Unreal Engine.

* Java for the use of OpenGL for graphics makes the game realistic along with good performance.

* Python to make the game more secure, hack proof and to provide a lot of built in functions for gaming and also making the game more adaptable and stable.

3.1.5 COMMUNICATION INTERFACE

Communication b/w the clients is facilitated by common network protocols using TCP/IP.

3.1.6 OPERATIONS

- The system should allow users to login.
- The system should allow users to log out.
- The system should allow users to register account.
- The system should allow the administrator to ban account.
- The system should allow users to play the game.
- The system should allow users to exit the game at any point.
- The system should allow users to view tutorial of how to play game.
- The system should allow the users to practice playing the game.
- The system should allow users to view their performance history.
- The system should allow users to view modify their game settings.
- The system should allow the administrator

to add, modify and delete levels.

3.2. FUNCTIONAL REQUIREMENTS

3.2.2 MENU REQUIREMENTS

These are the requirements that are related to the menu that is displayed in the game. This part is divided into two sections; one is about the general properties and usage of the items that can appear in this menu.

a) GENERAL REQUIREMENTS

- These are / is menu displayed from two contexts: first one is the entrance of the game (Main Access Menu) and the latter one is anytime whenever requested by the player during the playing of the game (Paused Access Menu).
- Menu is composed of items which are included in the menu according to the context that the menu is displayed from.

b) MENU ITEMS REQUIREMENTS

VIEW PROFILE

- This item is included in Main Access Menu
- The player can view his profile by this.
- The information about the player career results, statistics and achievements.

HELP

- This item is included in both Main Access Menu and Paused Access.
- Help section is displayed which includes general information about game contents, playing of the game and game controls.

SHOP

- This item is included in main menu access.
- The player can buy outfits, weapon's finishes and used to redeem items.

SETTING

- This consist of various settings like basic setting, graphics setting, controls setting, sensitivity setting, pick up setting, scope setting, audio setting, language setting and other settings.

MAIL

- The mail box is to receive game messages, updates and rewards.

INVENTORY

- This consists of the outfit, weapon finish and other items owned by player.

START BUTTON

- To start the game

FRIEND'S LIST

- It consists of in game friends list and also acts as communication channel between two players.

MISSION

- This consists of the mission that is to be accomplished by the player

3.2.3 GAME FLOW REQUIREMENTS

These requirements cover the period that the player that the player is in the game actively. It includes environment, main character abilities, interaction between player and game and overall game logic during the game.

GAME LOGIC REQUIREMENTS

- The game has a homogenous characteristic.
- The game is played in map, which consists of various locations.
- The player can change his location from one place to another.
- Most of the places will have buildings and monuments
- Every location has a different view, which means a different appearance.

HUMAN PLAYER: It has abilities like

- Walk
- Sprint
- Aim
- Jump
- Heal
- Crouch
- Drive
- Turn
- Reload
- Climb
- Sweep weapon
- Prone
- Peek
- Look around
- Pick up items
- Fire
- Use items
- Drop items
- Speak, chat with other players
- Get wounded, every player will have specific health
- Die

PLAYER-GAME INTERACTION REQUIREMENTS

- Player uses fingers on touch display to supply input to the game.
- Game uses display and sound devices to supply output to the player
- During the game the player can talk to each other, this talk can be private or public.
- Player manages the character's inventory items like selecting or dropping one by a menu.
- Player views the game from the first person's perspective and third person's perspective.

- Player can view the health and inventory status during the game.
- Player can escape to the menu during the game.

3.3

PERFORMANCE REQUIREMENTS

This section describes the expected performance requirements. This is an estimation of the system, and all the numerical values may vary depends on how large the final application is.

3.3.1

STATIC REQUIREMENTS

1. The system shall support only one terminal.
2. The system shall support only one simultaneous user on each device; however, it shall support multiple users to create personal accounts and access the system on the same machine at different times.
3. The system shall run on both iOS 8 and Android version above 5.1. Mobile devices with at least 3GB of memory.

3.3.2

DYNAMIC REQUIREMENTS

1. The system shall be loaded and functioning within 10 seconds 95% of the time after starting the game.

2. Each account shall be stored and activated within 5 seconds after creation.
3. Each user input during the session shall be responded to within 3 seconds 98% of the time.
4. Each session grade report shall be generated within 5 seconds at the end of each session 95% of the time.

3.4 ATTRIBUTES

RELIABILITY

The system should never crash or hang, other than as the result of an Internet connection error (Network Error). If there is a network error the system will ask the user to make sure that the device has active internet connection.

AVAILABILITY

The game has the risk of server failure because of excessive number of users. Important precautions will be taken to avoid the circumstance such as setting up checkpoints and some regular interval backups. So application will always have a stable state which we offer to users.

SAFETY

The system should warn the user to take a break after every two hours continuous play to prevent eyestrain and repetitive strain injury. No other safety requirements have been identified.

PORABILITY

The system must be portable to any Android System version above 5.1, including iOS 9. No other specific portability requirements have been identified.

SECURITY

Security will be guaranteed to all users. No personal information will be shared with or sold to any other third party companies.

MAINTAINABILITY

The design will be flexible. Whenever a new functionality is needed for game, it will be easy to integrate because the design is going to have a layered structure.

3.5

OTHER REQUIREMENTS

There are no other requirements.

4. USE CASES

The use of case diagrams provided along with, go in direct correspondence with each of the functional requirements item discussed in the Project Requirements section of the report.

The reader is encouraged to revisit that section, if necessary.

4.1. GAME PLAY USE CASE

The use case displays the actions that the player can do in game play without interaction with other players. We grouped these actions into four: Movement, Camera, Items and Menu Entrance. Player can walk, jump or crouch in the game. Those movements can be executed forward, backward left or right. Player can change the camera view in the game. Items are an important issue in the game. There are two types of items

in the game : Hall items are non-moveable items and can only be used , Inventory items are moveable . An inventory item can be picked up , dropped , equipped or unequipped . The last action is the Menu entrance in the game , player can enter the menu whenever he wants .

4.2. LOGIN USE CASE

This use case displays the player's login to the game . Authentication server checks player's username and password and according to the check it either allows or disallows player's entrance to the game .

4.3. CHAT USE CASE

This use case displays the chat usage in the game . Player can chat with other players in the game . This chat can be private , which can only be done by two players . Player can also interact with AI players in the game ; this interaction is of course limited .





