Çankaya University

Department of Computer Engineering

Software Requirements Specifications

for

CAMPUS KART RACING GAME

(CKRG)

Version 1.0

Prepared By:

Esra ŞAHİN

Ozan ÇETİNER

Buğra DOĞAN

N. Cem ALTUNBULDUK

Due Date: 04 December 2020

Change History

This document is the first version Software Requirements Specification document of the project named Campus Kart Racing Game. It was created on December 3, 2020.

Table of Contents

1. INTRODUCTION

The document is a review of the Software Requirements Specification (SRS) document of the CAMPUS KART RACING GAME project.

1.1 Purpose

This document provides information about the technical details of the CAMPUS KART RACING GAME (CKRG) project.

It also includes definitions of all functions that run on CKRG, specific requirements, limitations, user characteristics.

1.2 Scope of Project

CKRG is a cross-platform racing game with single player and multiplayer modes.The aim of the project is to provide a competitive racing experience among players. Players can join the game on mobile and computer. Also players can compete against artificial intelligence in single player mode or they can compete with other players in multiplayer mode if they wish.

1.3 Glossary

Expression	**Meaning**
CKRG	Campus Kart Racing Game
SRS	Software Requirements Specifications
SP	Single player mode is a game mode played by only one player.
MP	Multiplayer mode is a game mode in which multiple players can play together at the same time.

AI	Artificial Intelligence
CPG	Cross Platform Games
GE	Game Engine
Use Case Diagram	A type of diagram in UML that represents the user's interaction with the system.
IEEE	Institute of Electrical and Electronics Engineers
SDD	Software Design Document
NPC	Non Playable Characters
OS	Operating System
PC	Personal Computer
RAM	Random Access Memory

1.4 Overview of Document

This document gives a general description of CKRG. Also, this document is prepared in accordance with the IEEE Std. 830-1998, IEEE Recommended Practice for Software Requirements Specifications. [1]

This SRS document of our project consists of 7 sections. In the first part, there is general introduction information of the project. Overall description section is the 2nd section. Detailed explanations about artificial intelligence can be found in section 3. In 4 chapters, the requirements required for this project are explained clearly. Project planning and references information is at the end of the document.

2. OVERALL DESCRIPTION

This part defines the general factors that affect the CKRG and its requirements. To simply make it more understandable.

2.1 Product Perspective

CKRG is a cross-platform racing game developed with the Unity game engine. The game consists of two modes. These are the single player mode that competes against artificial intelligence. The second mode is the multiplayer racing mode where you can compete with real players. The user uses kart type vehicles. And it can use customizations such as adding new parts and changing colors as desired.

2.2 Memory Constraints

Devices must have and have at least 512MB of RAM.

2.3 Product Functions

There are many options available to the player in the product. These options are explained in detail with diagrams.

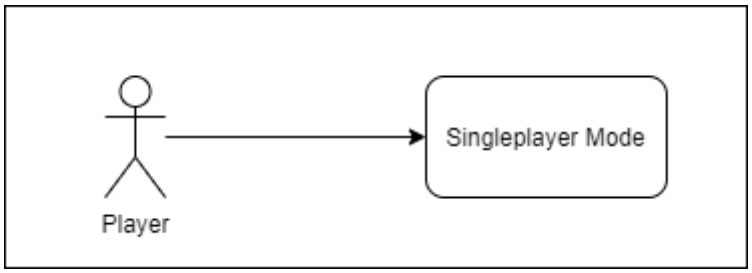

Figure 1: Singleplayer Mode.

Singleplayer Mode: It is the game mode in which the player will compete against artificial intelligence.

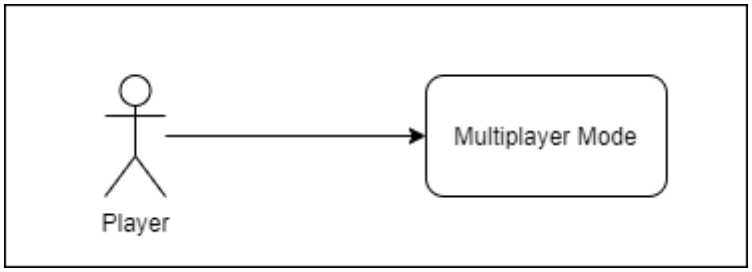

Figure 2: Multiplayer Mode.

Multiplayer Mode: It is the game mode in which the player can compete with other real players.

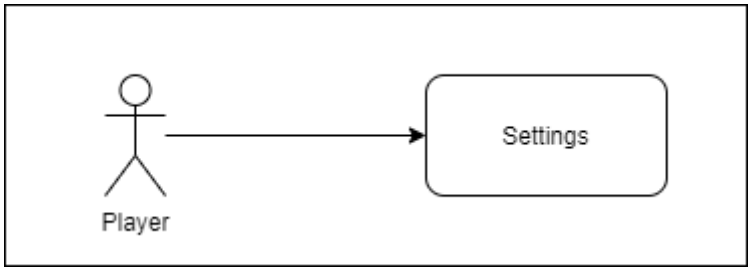

Figure 3: Settings that the user can set.

Settings: It is the menu where the user can make settings such as texture quality, loudness, resolution.

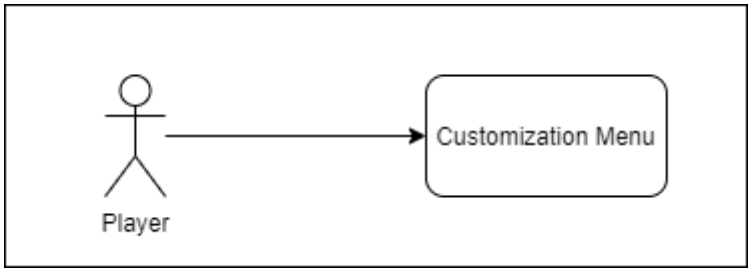

Figure 4: Customization Menu.

Customization Menu: It is the menu where the player can customize their vehicle. It can use customizations such as color change, add parts, etc.

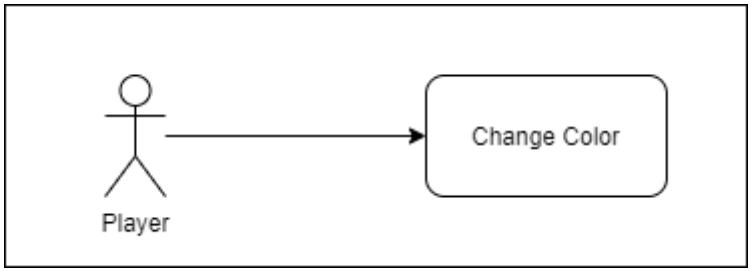

Figure 5: Car Color Changing.

Change color: Players can change the car color.

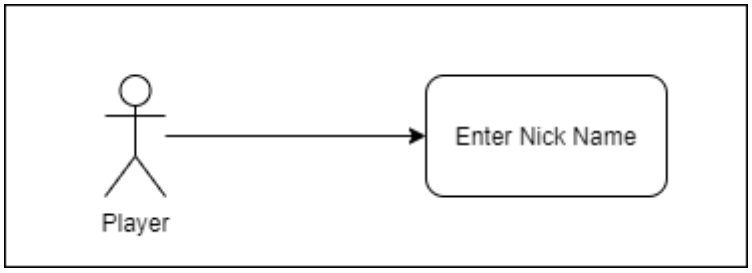

Figure 6: Entering Nick Name.

Enter Nick Name: The player can set the nickname that will appear in the game.

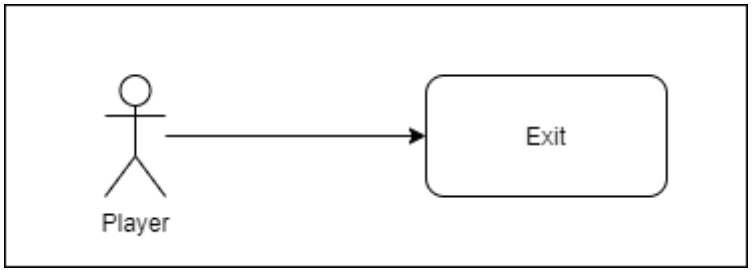

Figure 7: Exiting the Game.

Exit: The player can exit the game.

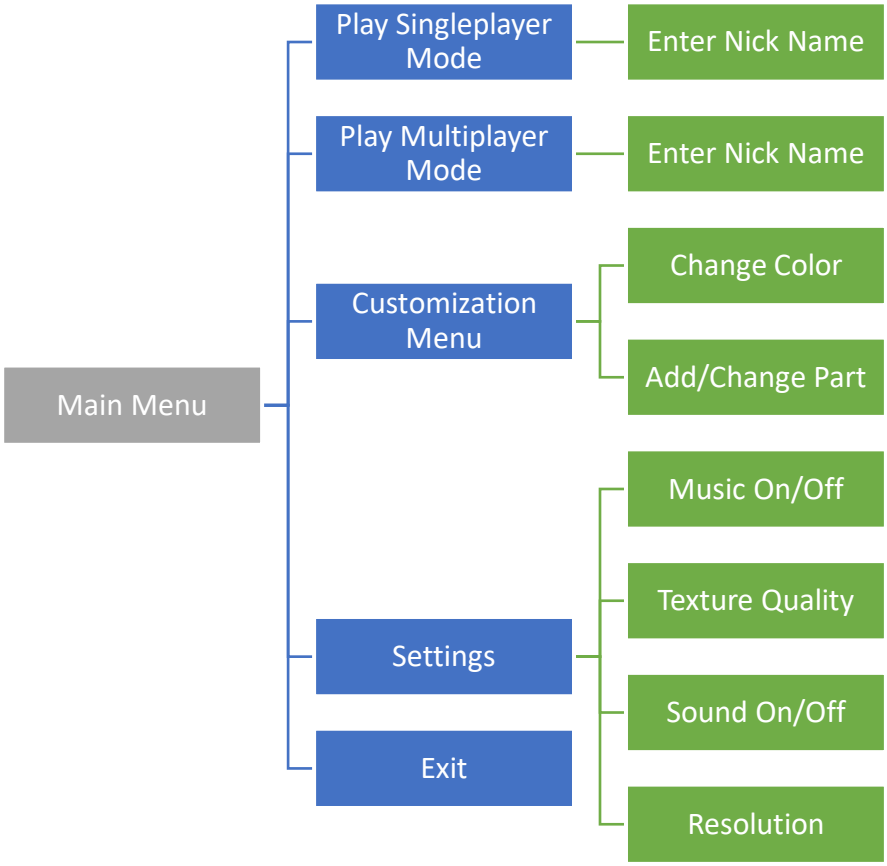

Figure 8: CKRG Functions

2.4 User Characteristics

The user who has knowledge about mobile or computer games should also know the English language.

2.5 Constraints

In this project, requires minimum Android 4.4+ for mobile devices, Windows 7 and higher for personal computers.Also, an internet connection is required for multiplayer mode.

2.6 Assumptions and Dependencies

We assume that the game server can handle at least 1.000 simultaneous connections for multi-user game.

3. ARTIFICIAL INTELLIGENCE

3.1 Introduction

In this part of the document, you will see how artificial intelligence and NPCs work and how they interact with each other. Artificial Intelligence is getting more and more talked about both academically and technologically. Game artificial intelligence continues to develop depending on all these other developments.

Looking at today, many progress has been made in artificial intelligence. Health sector, voice assistants, e-commerce, autonomous vehicles and communication are among the areas of use. In our age, there is no place where artificial intelligence is hardly used. In digital games, the use of deep learning model artificial intelligence and rule-based artificial intelligence, which are among the artificial intelligence techniques, has become widespread.

To understand how artificial intelligence will become more intertwined with games, it is necessary to look at the history of the artificial intelligence and the game industry. From the earliest days of the game industry, developers have been striving for artificial intelligence to act like a human and create a game world from scratch without the need for a real person. Because it is thought that an active learning artificial intelligence will create games of superhuman beauty. For example, in a war game, enemy soldiers who are constantly hit on the head see this situation and come wearing more robust helmets in the following sections. In another example, the game automatically increases the values of the players of the opponent team and offers you a more challenging game according to your playing well in the football match game. Another contribution of artificial intelligence to the game industry is in graphic design. It can analyze the pictures recorded on the computer and process them on the game and thus reveal more realistic games.

To explain the artificial intelligence technology in games in a simpler form, all the events that occur outside of you while you are playing are made by this technology. For example; When playing games, a character passes by us or enemy soldiers see and shoot you in a war game.

3.2 Non Playable Characters

NPCs will be active in single player mode and will be managed by artificial intelligence. In online mode, real-time users will compete instead of NPC. So no matter which mode you enter the game, you will be able to find a character that can compete.

3.3 NPC Interaction with Players

It is the name given to the characters that we have seen in online games, moving from time to time and standing still from time to time. "NPC" are "BOT" services that are used for item purchases and placed by game developers so that you can receive the rewards offered by the task and the game.

In single player mode, the user will compete against these NPCs. With the help of AI, NPCs will detect when a car arrives near them and act accordingly. For example, when the user's tool NPC gets too close to the vehicle, the NPC may understand this and try to move away.

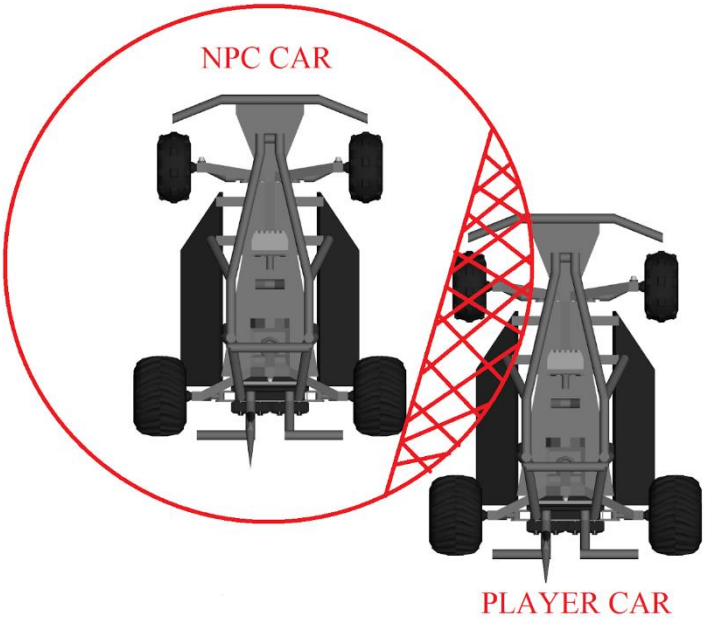

Figure 9:Non Playable Character Car vs Player Car.

3.4 Pathfinding

The Pathfinding method is the method that you can move freely between two points. It gives you more options for determining the destination and feels freer.

When the race starts, each NPC car will first calculate the path to a destination and then follow it. The first part of the algorithm is called 'pathfinding', the second part is called 'path tracking'. The scope of our changes was mostly limited to the latter and we avoided making changes (other than minor adjustments) to the pathfinding algorithm. This is because these methods are shared among all the different object classes in the game.

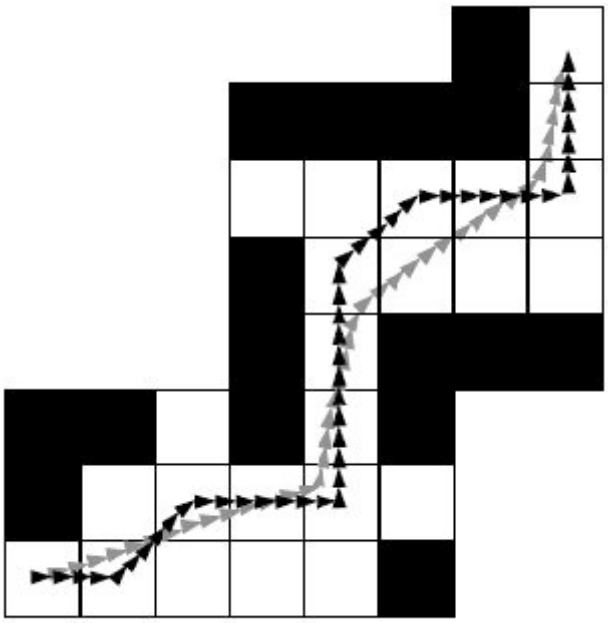

Figure10:Pathfinding in grid system.

3.5 Curvatures and Bumps

Curved areas and realistic bumps in car racing, along with bends, maximize the gaming experience. If the control is not maintained correctly when driving at high speed, the vehicle may become unusable. For this reason, the curvatures and bumps on the racing fields are minimized. While doing this, real go-kart tracks have been researched and examined in detail.

3.6 Overtaking the Car

One of the important factors to win in racing games is successful overtaking. For this, we organize systems that will create overtaking situations for the player from the race track to the structure of their cars. In the next stage, the places of the checkpoint and final lines are chosen properly for overtaking. It is referred to as "the Field of Opponents", one of the subfields of artificial intelligence.

3.7 Driving the Car without Going off the Road

To drive in the track direction without any collision with the track edges, we must have repelling as well as attractive forces spread all over the track to keep the car on track. The repelling forces are concentrated on the edges of the track while the attractive forces are spread along the preferred navigational path. For example if our controller wants to follow the middle of the track throughout the race, the attractive forces are placed in the middle of the track along the length of the whole track.

3.8 Improvements and Additional Details

- A menu that will launch the game quickly. Two simple options: Single Mode and Multiplayer mode. After the mode is selected, the countdown for the race starts.

- Sound effects for acceleration, braking, crash, race start countdown and finish.

- One way to keep number of laps and show them during the race so that we know how many laps passed while the race was going on and when it ended.

- A timer that shows how many minutes and seconds have passed since the race started.

- Win / Loss screen and ranking chart after the race is over, based on what the player or computer has won.

These features must be available in the final publishable version of the game.

4. SPECIFICATION OF REQUIREMENTS

4.1 External Interface Requirements

4.1.1 User Interfaces

The user interface will be worked on Windows and Android operating systems.

4.1.2 Hardware Interfaces

Minimum system requirements that can run the game:

1. Graphic processor supporting minimum OPENGL ES 2.0 or higher for mobile devices.
2. Graphics processor that supports minimum DirectX 9.0 and higher for personal computers.

4.1.3 Software Interfaces

Since this application is a cross platform application, it will need an Android version 4.4 or higher or Windows 7 or higher version.

4.1.4 Communications Interfaces

Wired or wireless internet connection is required for multiplayer game mode.

4.1.5 Performance Requirements

At least 30 FPS is aimed for a smooth gaming experience. Also, for a competitive multiplayer mode, the internet ping values should be below 100ms.

4.2 Software System Attributes

4.2.1 Portability

- The game has been developed for Windows and Android operating systems.
- The game supports both 32 and 64 bit processor architecture.

4.2.2 Performance

- The game frame rate must be 30 frames per second.
- Instead of real time lighting system, baked lighting system should be used.
- Objects outside the camera angle should not be rendered.
- Level of detail of objects should be changed according to distance between object and the participant.

4.2.3 Usability

When a player will open the game,he/she will see a simple interface.The game is easy-to-use.

4.2.4 Adaptability

The game can be played on computers and smartphones only.

4.2.5 Safety Requirements

Playing games for a long time (looking at the screen) can cause illness such as eye strain.

5. PLANNING

The following sections give information about this project planning.

5.1 Team Structure

The four members of the CKRG team are Buğra Doğan, Esra Şahin, Ozan Çetiner and N. Cem Altunbulduk.

TASK	**MEMBERS**
3D Modelling	Team
Game Logic	Buğra DOĞAN, Cem ALTUNBULDUK
Game Music	Esra ŞAHİN
Development of Artificial Intelligence	Buğra DOĞAN, Cem ALTUNBULDUK
Game Animations	Ozan ÇETİNER
Lighting	Buğra DOĞAN, Esra ŞAHİN
Sound Effect	Team
Menu Design	Buğra DOĞAN, Ozan ÇETİNER

5.2 Estimation

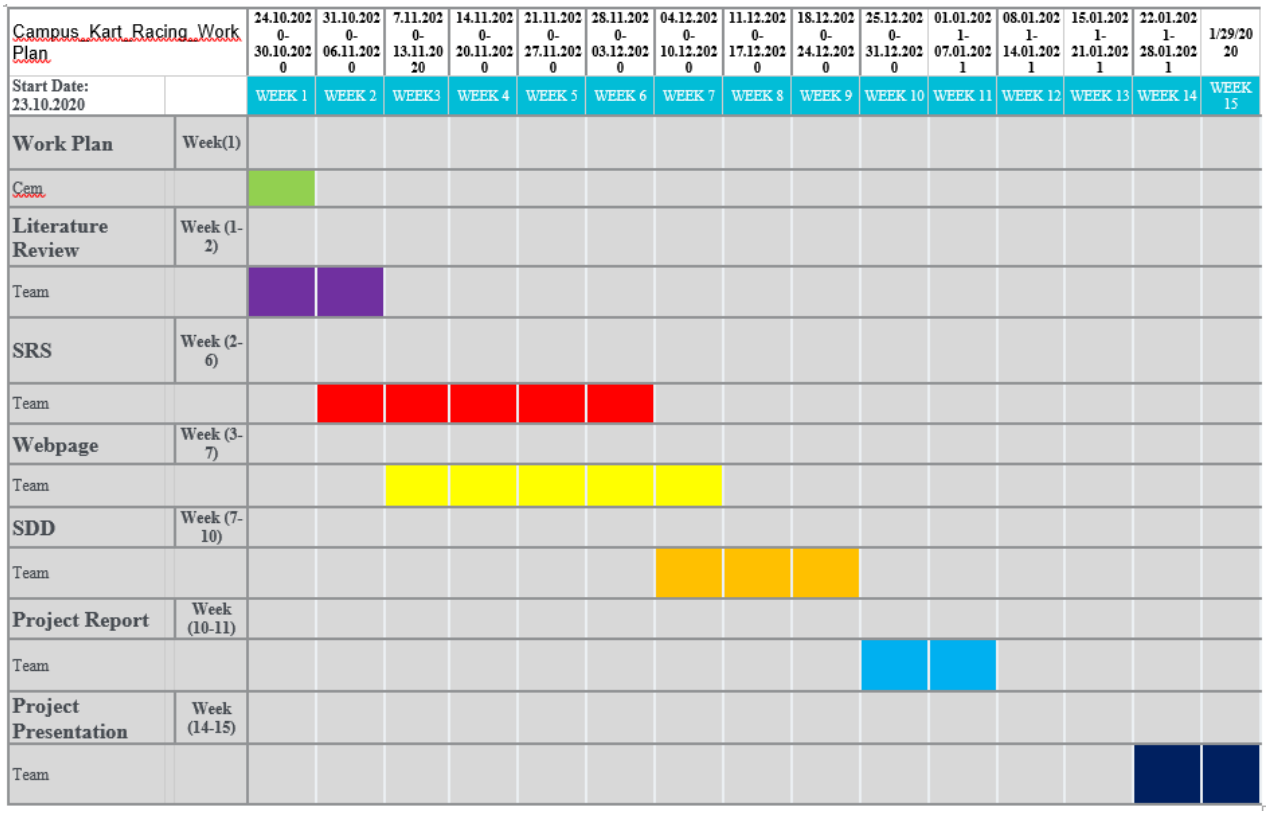

Figure 11: Project Work Plan.

5.3 Process Model

We chose to work with the agile method which is one of the project management methodologies for our CKRG game project. The necessary elements for our CKRG project (regular adaptation to changing conditions and even late changes in requirements are welcomed), the most convenient method is the agile method.

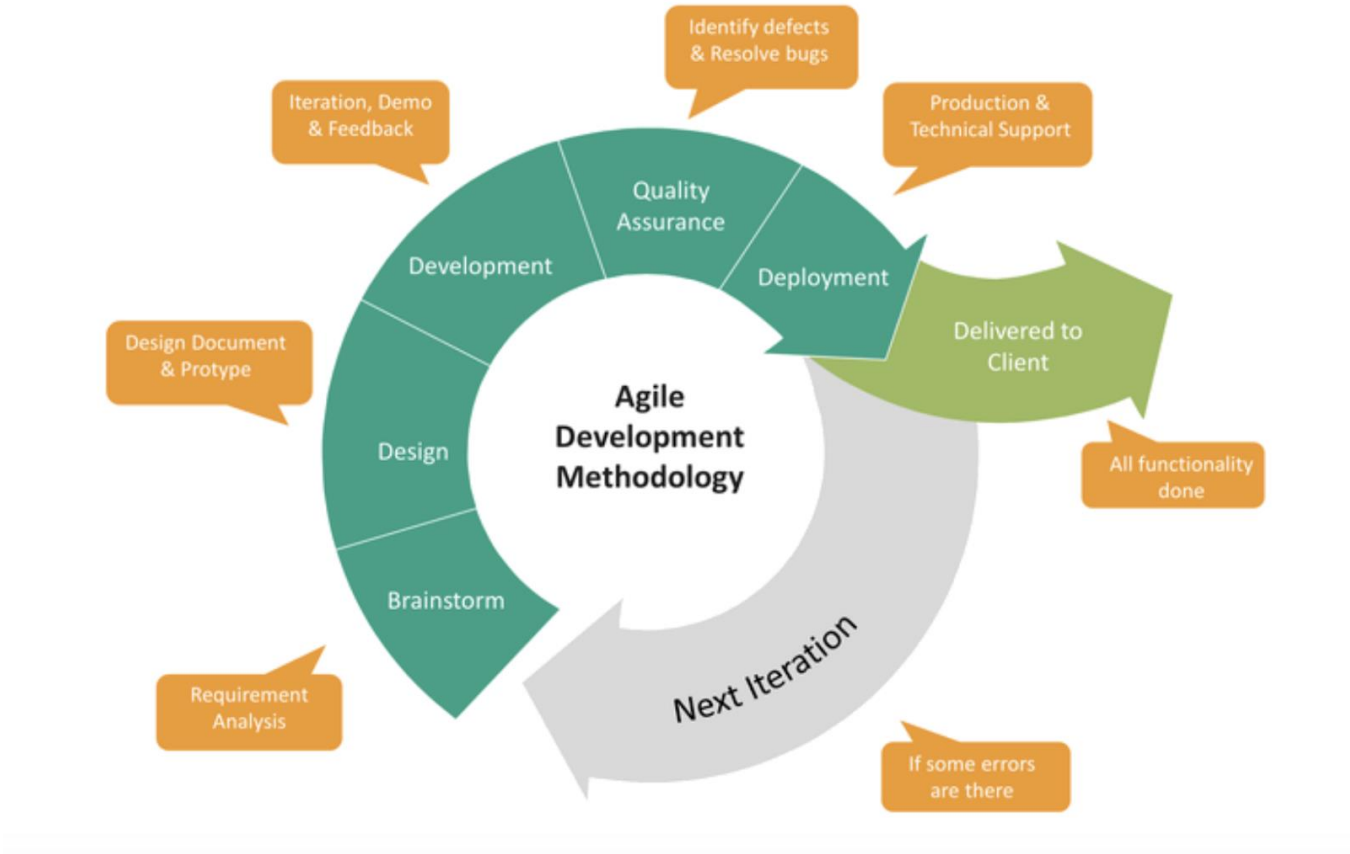

Figure 12: Agile Development Methodology.

Agile method is the general name for the iterative software development method which anticipates continuous development and testing activities throughout the software life cycle. These iterative steps are plan, design, develop, test and release.

6. CONCLUSION

In this software requirement specification document, all the required software requirement analysis, used systems and details of the project named CKRG are explained. Detailed functional and non-functional requirements, system, user, software and hardware interfaces, artificial intelligence are explained in detail. This SRS document will be a guide for the SDD document to be developed.

7. REFERENCES

[1] IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications.