

**Signage Navigation System
Bar Genish - 313174583**

**Software Requirements Specification
Document**

Version: (2)

Date: (01/18/2020)

Table of Contents

1. Introduction	4
1.1 Purpose	4
1.2 Scope	4
1.3 Definitions, Acronyms, and Abbreviations	4
1.4 References	4
1.5 Overview	4
2. The Overall Description	5
2.1 Product Perspective	5
2.1.1 System Interfaces	5
2.1.2 Interfaces	5
2.1.3 Hardware Interfaces	6
2.1.4 Software Interfaces	6
2.1.5 Communications Interfaces	6
2.1.6 Memory Constraints	6
2.1.7 Operations	6
2.1.8 Site Adaptation Requirements	6
2.2 Product Functions	6
2.3 User Characteristics	7
2.4 Constraints	7
2.5 Assumptions and Dependencies	7
2.6 Apportioning of Requirements	7
3. Specific Requirements	7
3.1 External interfaces	7
3.2 Functions	7
3.3 Performance Requirements	8
3.4 Logical Database Requirements	8
3.5 Design Constraints	8
3.6 Software System Attributes	8
3.6.1 Reliability	8
3.6.2 Availability	8
3.6.3 Security	8
3.6.4 Maintainability	8
3.6.5 Portability	9
3.7 Organizing the Specific Requirements	9
3.7.1 System Mode	9

3.7.2 User Class	9
3.7.3 Objects	9
3.7.4 Feature	10
3.7.5 Stimulus	10
3.7.6 Response	10
3.7.7 Functional Hierarchy	10

3.8 Additional Comments

4. Change Management Process

5. Document Approvals

1. Introduction

1.1 Purpose

The purpose of this SRS is to define the project in a convenient and orderly manner.
Our intended audience is : customers and the developers.

1.2 Scope

Signag navigation system,

The project will include a VR game(tour) where the player has a mission to get to a specific checkpoint.

The game will collect statistic information about the user's path.

The goal is to find how much does signboards help people operate in certain situations.

1.3 Definitions, Acronyms, and Abbreviations.

Scene –a picture that every possible cross-section is an new scene.

Signboard - kind of clue that should help the player reach his destination.

Vr glasses - Virtual reality glasses that match the game for the user, which will be charged for play

User – the player that use the system.

1.4 References

<https://docs.google.com/document/d/1I08qX76DgSFyN1ByIGtPuqXh7bVKraHcNIA25tpAzE/edit>

<https://firebase.google.com/docs>

1.5 Overview

the rest of the srs contains :

user requiremets and details in section 2

developer requierment and details in section 3

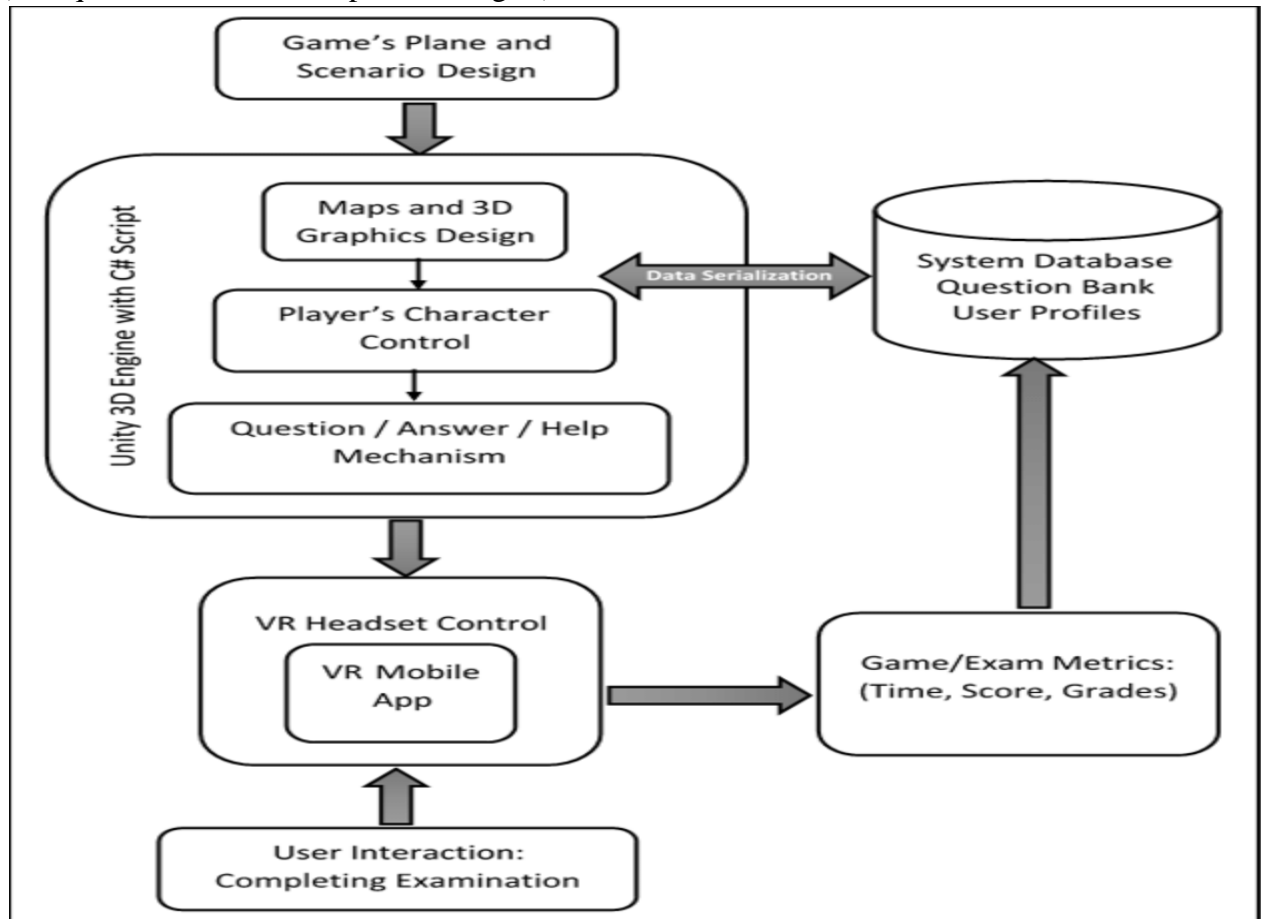
2. The Overall Description

2.1 Product Perspective

The product related to google cardboard vr glasses , so it should support and run on the glasses.

The product needs to get the data (picture) and build the scenes in vr such that it would work on the vr glasses without problems.

(The question bank are the possible edges.)



2.1.1 System Interfaces

As I said the system should run with google cardboard on mobile phone.

So it should work with phone with logical storage, speed and power.

The project will be written in c# on unity, so its should interact with unity platform.

2.1.2 Interfaces

The system will interact with the users in real time (vr). The system will show the user hotspots that represent choices to move around. The user will be able to travel between scenes by looking on the hotspots for a specific amount of time.
the connection between the user and the system should be easy , reliable (To allow for a proper transition between scenes) and fast .

2.1.3 Hardware Interfaces

The system will run on all android devices with API level 16(Jelly Bean or later) that support unity engine and google cardboard.

2.1.4 Software Interfaces

Firebase db – we need to save the user's output(statistics).
Support unity version 2018.3.7f1

2.1.5 Communications Interfaces

The system will make use of a web-based database, so it will require web services to save the user's output(statistics).

2.1.6 Memory Constraints

In order to run the system, the cellphone will need to have enough space to run the game with all of his data (pictures).

2.1.7 Operations

The normal and special operations required by the user are :
insert correct data, play in a spacious place to stay protected .

2.1.8 Site Adaptation Requirements

To install the system and use it, the user must have a "google cardboard" VR-headset and an android device. The user will need to enter certain information .

2.2 Product Functions

The major functions are:

1. Create the scenes by the input
2. Set automatic hot-spots using the input data
3. Enable movement between scenes by looking at hot spot.

4. Save the statistic information on the firebase.

2.3 User Characteristics

Our intended users are everyone, we don't require any specific attribute.

2.4 Constraints

- The system will run fast
- No buttons
- Should run on a mobile phone processor
- Support google cardboard
- Deadline is 9.9.20

2.5 Assumptions and Dependencies

- Finding a more convenient db.
- Finding a more convenient vr engine.
- Supporting different version for cell phone.

2.6 Apportioning of Requirements.

- Which statistic information is interesting us.
- How to design the system.
- How should the system show messages and information to the user.

3. Specific Requirements

3.1 External Interfaces

3.2 Functions

- 3.2.1 The system will get input that contains :
 - Photos of the scenes (a 360 photo)
 - Possible passage between all scenes
 - The azimuth of every possible passage between all scenes
 - Id of every photo
- 3.2.2 The system will take statistic information from the game and upload it to the db.
- 3.2.3 The system may set automatic hot-spots using the input data.
- 3.2.4 The system will build the game generically according to the given pictures and data.

- 3.2.5 The system will allowed the user to move from specific scene only to optional scene using the passages.
- 3.2.6 The system will inform the user when he gets to the destination.
- 3.2.7 The system will measure times for the user from the beginning of the game untill the destination.
- 3.2.8 The system may receive 360 images and transfer them to VR.
- 3.2.9 The system will ask the customer to be in a protected environment while using it.
- 3.2.10 The system will save the user path for analyzing it.
- 3.2.11 The system will have a starting point and a destination point that the user will know and will need to reach.
- 3.2.12 The system will build a graph based on the input data that it received ,where every image is a scene(node) and every cross-section is a legitimate option(edge).
- 3.2.13 When the player goes from one scene to another we want him to look in the next scene to the same azimuth he was looking before the change.
- 3.2.14 The system will exit if the input not valid.- optional
- 3.2.15 Each image will be transformed into a scene.
- 3.2.16 The program will put pressure on the user to check their results under pressure – optional.

3.3 Performance Requirements

3.4 Logical Database Requirements

We will save the statistic information on the firebase , we will create a class - game_stats that will contain the time,score, name and all the information that we need, and we will insert it to the db.

The data will be inserted automatically after the player will finish the game.

We will tell the user that we are going to use this data.

3.5 Design Constraints

The system will run on a mobile device operating system. Need to be supported by devices with limited memory and CPU power.

3.6 Software System Attributes

3.6.1 Reliability

3.6.2 Availability

3.6.3 Security

- Check data integrity for critical variables and input data
- Restrict communications between some areas of the program – connect with db only after the game is over and prevent the user from taking control of the db.

3.6.4 Maintainability

3.6.5 Portability

3.7 Organizing the Specific Requirements

3.7.1 System Mode

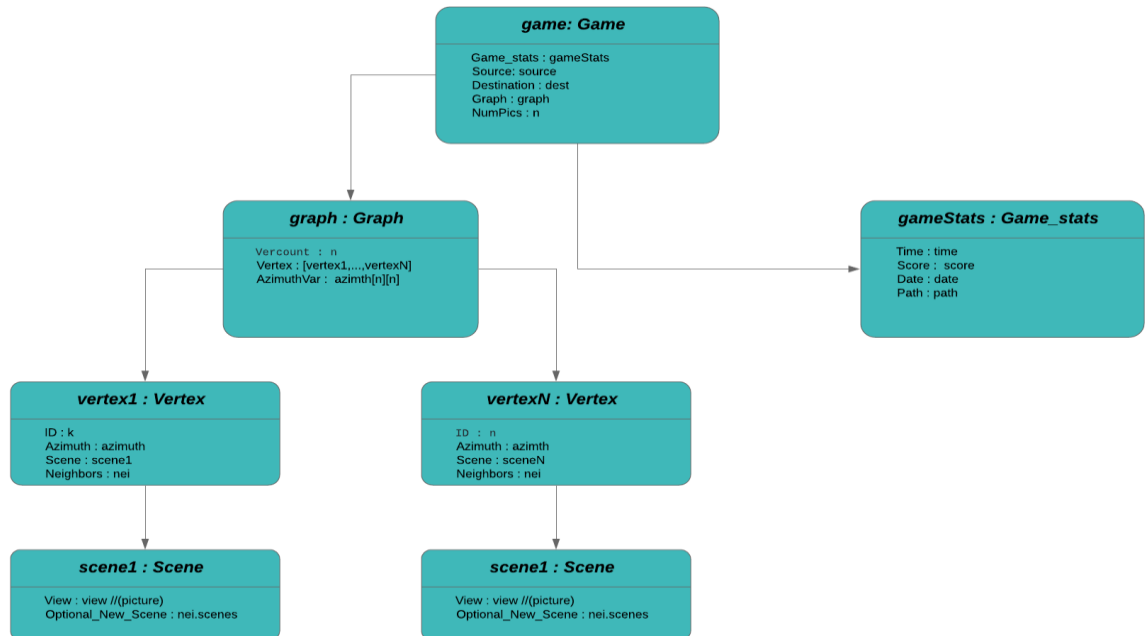
The system contain debug mode and production mode. The first will show relevant data for debugging and operating the program on a low level.

3.7.2 User Class

Same sets of functions for all users.

3.7.3 Objects

- Vertex – represent a picture and all its neighbors are possible new scene.
 - Graph – contains all images and allows the transfer of an image to its neighbors.
 - Game_stats – an object that will keep all the statistics that interest us and will keep in the db.
 - Game – gets data as input, graphs for this data , analyzes the data of interest and saves it in Game_stats.
 - Scene – a certain view that the user sees and can move to a new scene (each Vertex is a scene)
- Object diagram of a game :



3.7.4 Feature

3.7.5 Stimulus

3.7.6 Response

3.7.7 Functional Hierarchy

3.8 Additional Comments

4. Change Management Process

5. Document Approvals [YOUR SUPERVISOR]

Identify the approvers of the SRS document. Approver name, signature, and date should be used