



SOFTWARE REQUIREMENTS SPECIFICATION

Virtual Tour Based Game (Inbound Contingency)



FEBRUARY 13, 2018

GROUP - 20

Inderpreet Singh Chera – 160101035

Shubhendu Patidar – 160101068

Siddharth Sharma - 160101071

Table of Contents	2
1. Introduction	3
1.1 Purpose	3
1.2 Document Conventions	3
1.3 Intended Audience and Reading Suggestions	3
1.4 References	3
1.5 Product Scope	3
2. Overall Description	4
2.1 Product Perspective	4
2.2 Product Functions	4
2.3 User Classes and Characteristics	4
2.4 Operating Environment	4
2.5 Assumptions and Dependencies	4
2.6 Apportioning of Requirements	5
3. External Interface Requirements	6
3.1 User Interfaces	6
3.2 Hardware Interfaces	6
3.3 Software Interfaces	6
3.4 Communications Interfaces	6
4. Functional Requirements	7
4.1 Game-start menu	7
4.2 In-game start-menu	7
4.3 Start training mode	7
4.4 Shortest path to checkpoint	7
4.5 Show labels	8
4.6 Start new game	8
4.7 Maneuvering the player	9
4.8 Asking Questions	9
4.9 Score Calculator	9
4.10 Providing user preferences	10
4.11 Sharing	11
5. Other Nonfunctional Requirements	12
5.1 Performance Requirements	12
5.2 Security Requirements	12
5.3 Availability	12
5.4 Maintainability	12
5.5 Portability	12
5.6 Reliability	12
5.7 Usability	13
5.7.1 Contextual Inquiry	13
6. Other Requirements	16
Appendix A: Abbreviations	16
Appendix B: Tools used	16

1.Introduction

The Software Requirements Specification lays the foundation for the systematic approach towards the development of this software. It provides a general overview of the system. It helps to translate the ideas in the minds of the clients. It includes understanding of the problem domain, establishing an architectural framework, identifying all external entities that will interact with the system, the users and defining their functions, the constraints that would exist and the overall functioning of the system.

1.1 Purpose

In this Document, we describe the software requirements for a yet unnamed virtual reality game, which will be henceforth called as “Inbound contingency(IC)”, which familiarizes the end-user to the layout of Computer Science and Engineering, IITG. The purpose of this document is to give a detailed description of the requirements for this game. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with the user. This SRS covers the entire application from front end graphical user interface to the back end of the product.

1.2 Document Conventions

The format specified by IEEE was followed while creating this document. The Font is ‘Times New Roman’. The headings of each topic are bigger in size. Numbered points are used as and when required.

1.3 Intended Audience and Reading Suggestions

The intended audience of this SRS document is our overseeing project professor, Dr. Samit Bhattacharya. The document is meant as a means to inspect the students in and refer to their overall product goals and commitments. It also serves an ancillary purpose of allows the project team to make sure that intended goals and functionalities of the product are being met.

1.4 References

IEEE, IEEE Standard 830-1998 IEEE Recommended Practice for Software Requirements Specifications, IEEE Computer Society, 1998.

1.5 Product Scope

The purpose of this project is to design and develop a game which helps the user to become acquainted with the CSE department of IITG. We do this by providing an immersive virtual tour followed by some fun trivia which encourages the user to get familiar with the place.

Inbound Contingency is thus meant to serve as a newcomer’s guide for familiarizing the user to the map and layout of the CSE department, IITG. Our goal is to develop a user friendly interface which promotes the memorization of all the pathways of CSE department in a fun way. The user can visit CSE department, IITG even if they are not physically available in IITG. Even physically challenged person who are unable to go on 1st and 2nd floor due to lack of lifts in the department can visit through the game.

2.Overall Description

2.1 Product Perspective

Our vision of this project is that this is available for all departments of IITG, and then later probably get adopted by even more Institutes. We want this to develop into a cloud based integration which provides the map of most of the institutes via the cloud. This allows the user to install the app with virtually no storage requirement.

2.2 Product Functions

Functions included in the final product will be as follows:

1. Game-start menu
2. In-game start-menu
3. Start training mode
4. Shortest path to checkpoint
5. Show labels
6. Start new game
7. Maneuvering the player
8. Asking Questions
9. Score Calculator
10. Providing user preferences
11. Sharing

2.3 Users classes and characteristics

Our game is targeted for 4 major groups of people. These include:

1. The people who just want to play the game for fun and enjoyment.
2. The people who want to explore the CSE department of IITG.
3. New comers who want to take tour of the department so as to get acquainted with it.
4. Physically challenged person who are unable to walk.

The intended users for the product will have the following characteristics:

1. Able to afford and use a Windows device following the minimum hardware and software requirements.
2. Able to understand the functioning and operation of the software on a basic level.
3. Able to understand the English Language to operate the application.
4. The users of all ages can use our game.

2.4 Operating Environment

The game works on universal windows platform. It will work on Anniversary update 1607 or higher.

2.5 Assumptions and Dependencies

It depends on resolution of images used to create views for the game. It also depends on the features provided by Unity and Autodesk 3ds Max SDK.

Visual Studio and Unity are free of cost, so cost of game is also negligible. Performance of game will depend on hardware configuration of device and operating system in use.

2.6 Apportioning of Requirements

The requirement for modification of gaming environment (weather, etc.), and integration with social media platforms may be avoided in the first version of the design and thus might not be available after the product is finished. Since these requirements are secondary they can be avoided without affecting overall functioning of the application.

3.External Interface Requirements

3.1 User Interfaces

The application would have a user friendly interface. The screen displays the first person view of the CSE department, IITG. Dedicated overhead text will show the score and queries as well as the current location and destination of the player.

There would be a “Training” section describing controls and providing a tour of the CSE department, IITG so that the user can familiarize and memorize the major pathways and locations.

3.2 Hardware Interfaces

Minimum System Requirements-:

1. Processor: Intel Core series and Intel M series.
2. RAM: 2 GB
3. Video card: Intel HD Graphics 6000
4. Peripherals: Keyboard and mouse.
5. Hard Disk Space: 1 GB

3.3 Software Interfaces

The game will run on all windows 10 devices (version 1607 and up).

3.4 Communication Interfaces

Only communication interface is the internet. It allows the user to share his/her score on social media.

4.Functional Requirements

4.1 Game start Menu

Input: Game is launched

Output: Main menu appears

Description: When the user starts the game, main menu appears on the screen having options like start game, quit, choose a level, settings, etc.

4.2 In-game Menu

Input: Escape key is pressed while game is running.

Output: In-game menu appears.

Description: An in-game menu appears with options like Resume, Exit to main menu, settings, etc.

4.3 Start Training mode

Input: Training option is selected.

Output: Training mode begins.

Description: The training mode starts and the user can free roam the whole CSE department and explore all the shortest path from one location to another.

4.3.1 Map Loader

Input: Training option is selected.

Output: Map is loaded and shown.

Description: Map is loaded and player is placed at a fixed starting point.

4.3.1.1. User positioning

Input: Training option is selected.

Output: User at Starting point location.

Description: User is positioned at a particular pre-fixed location.

4.3.2 Show overhead controls

Input: Training option is selected.

Output: User movement controls are displayed.

Description: The pre-defined controls are displayed on the starting of the training mode.

4.3.3 Show Miniature Map

Input: Training option is selected.

Output: Miniature map is displayed.

Description: A 2-D miniature map of the whole department with current location and checkpoints is displayed on the screen.

4.4 Shortest path to Checkpoint

Input: User's current location.

Output: Directions to the selected location are given.

Description: A user will select from the list of the available checkpoints and user will get the directions along the shortest path to the selected checkpoint.

4.4.1 List checkpoints

Input: User clicks the checkpoints button on the game screen

Output: A list of checkpoints is displayed

Description: There is a dedicated checkpoint button on the game screen. As soon as the user clicks on the button. A list of checkpoints is displayed from which the user can select a particular one as destination.

4.4.2 Display path

Input: A checkpoint is selected.

Output: Directions are displayed.

Description: After selecting the destination checkpoint from the checkpoint list, the shortest path to the checkpoint from user's current location is displayed.

4.5 Show Labels

Input: User's Current Location.

Output: Overhead labels are displayed.

Description: Adds labels like Name of Room, label to any specific object (e.g.: computer, etc.), etc.

4.6 Start New Game

Input: Game option is selected.

Output: Game mode starts.

Description: The game mode starts and the user is placed at a random position.

4.6.1 Map Loader

Input: Game option is selected.

Output: Map is loaded and shown.

Description: Map is loaded and player is placed at a random starting point.

4.6.1.1. Randomized initial position

Input: Game option is selected.

Output: Player at random starting point location.

Description: User is positioned at a random initial location.

4.6.1.2. Randomized final position

Input: Game option is selected.

Output: Destination is displayed

Description: A random destination is chosen from the list of checkpoints and displayed.

4.6.2 Show objective

Input: Game option is selected.

Output: Objective is displayed.

Description: Objective for the game is displayed i.e. to find shortest path from starting point to destination and etc.

4.6.3 Show Miniature Map

Input: Game option is selected.

Output: Miniature map is displayed.

Description: A 2-D miniature map of the whole department with current location and destination are displayed on the screen.

4.7 Maneuvering the player

Input: A specific key is pressed (or the mouse is moved).

Output: Frame of reference of the player changes.

Description: Depending on the key pressed (and the movement of the mouse), the player either moves or changes orientation.

4.7.1 Location Change

Input: A key mapped to user movement is pressed.

Output: Player moves.

Description: As soon as the key is pressed, the player moves in the specified location

4.7.2 Orientation Change

Input: A key mapped to user orientation is pressed or the mouse moves.

Output: The orientation of the player changes.

Description: Depending on the key pressed (and the movement of the mouse), the player either moves or changes directions.

4.8 Asking Questions

Input: Player reaches the destination coordinates.

Output: Questions in the form of overhead text displayed.

Description: After the user completes the objective and reaches the destination, he/she is asked a set of questions testing his/her knowledge about the map.

4.9 Score Calculator

Input: Objective is completed.

Output: Final score is displayed.

Description: After the user answers the questions the final score is calculated taking into account the path, and time the user took to reach the destination along with the performance in the question/answers.

4.9.1 Calculate Score on the basis of path

Input: User reaches destination.

Output: Returns the Score of the player.

Description: This function returns the score of the player taking into account the distance moved by the user and the time taken by the user.

4.9.1.1 Calculate Score according to time taken

Input: User reaches destination.

Output: Returns the time taken by the player.

Description: This function returns time taken by the player to reach the destination.

4.9.1.2 Calculate Score according to Distance moved by the player

Input: User reaches destination.

Output: Returns the distance covered by the player.

Description: This function returns the distance covered by the player to reach the destination.

4.9.2 Calculate score on the basis of Question-Answers

Input: Objective is completed.

Output: Returns the Score of the player.

Description: After the player answers the questions, the score based on his/her performance is calculated.

4.9.3 Calculate final score

Input: Objective is completed.

Output: Calculates and displays the final score on the screen.

Description: This uses the output of the above two functions to calculate the final score of the user

4.9.4 Storing High Score

Input: A specific key is pressed.

Output: Frame of reference of the player changes.

Description: Depending on the key pressed (and the movement of the mouse), the player either moves or changes directions.

4.10 Providing user preferences

Input: "Preferences" option is selected.

Output: The list of modifications is shown.

Description: Allows the users to customize the gaming experience according to their own needs.

4.10.1 Modifying sound settings

Input: sound settings are selected.

Output: Sound settings are changed.

Description: Allows the user to change the background music, or the volume during the game.

4.10.2 Modifying controls

Input: controls settings are selected.

Output: Sound settings are changed.

Description: Allows the user to change the controls of the game.

4.10.2.1 Movement controls

Input: A movement control is selected.

Output: Control for the movement is changed.

Description: Allows the user to customize movement controls.

4.10.2.2 Camera controls

Input: A movement control is selected.

Output: Control for the camera is changed.

Description: Allows the user to customize camera controls.

4.10.3 Modifying weather and time of day

Input: An environment modification is selected.

Output: The gaming environment changes.

Description: Allows the user to customize the environment in which they are playing, like changing the time of day, or the general weather etc.

4.11 Sharing

Input: The Score is calculated.

Output: User's score is shared via Social Media.

Description: Allows the user to share his/her score via Social Media.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The performance of the game depends upon-:

5.1.1 GPU and Video card: Better GPU will result in no frame-drops and no lags while maneuvering.

5.1.2 RAM: Better and more RAM will help in preloading next frames instantly.

5.1.3 CPU configuration: Loading the game requires high CPU usage, so much better processor will reduce its loading time and is thus preferred.

5.1.4 Desktop is preferred in place of hybrid devices (Ex Surface Pro) because of dedicated GPUs.

5.2 Availability

The system will be available for use whenever the user deems necessary 24/7. The system shall allow users to restart the application after failure with the loss of at most the last running instance of the game.

5.3 Maintainability

The system will be updatable from software patches available through the Windows Store. Any discrepancies will be addressable by any developer as the coding will be done according to the coding standards of IEEE.

5.4 Portability

Since the game is developed using the universal windows platform, it will run on any device which works on windows 10(including tablets, laptops and hybrids).

5.5 Reliability

5.5.1 The software will be able to run 99% of the time when launched.

5.5.2 There is a potential for errors relating to the state of the operating system that could prevent the game from launching (for example not enough resources available, etc.). The chance of such an occurrence is at most 1%.

5.5.3 Online backup of high score will be provided by the system

5.5.4 The software will employ mechanisms to ensure that there is no data loss if the game crashes.

5.5.5 The system will not be prone to errors caused by unexpected input.

5.5.6 The software will be able to handle all sorts of input and be immune to side effects cause by undesirable inputs (such as buffer overflow), which could potentially create security holes in the system.

5.6 Usability

The system has been envisioned according to the User Center Design model(UCD). The usability of the system has been described based on the factors given below:-

5.6.1 Contextual Inquiry (CI)

Contextual Inquiry is a user centered design research method which is usually structured as an approximately two hour one on one interaction in which the researcher watches the user in the course of user's normal activities and discusses those activities with the user.

It can be performed in following two ways -:

5.6.1.1. Active CI - In this type of CI the user is actively included in development team.

5.6.1.2. Passive CI - In this type of CI the user merely provides his/her observations and experiences to be recorded by the developers.

While developing our project we adopted passive type of contextual inquiry.

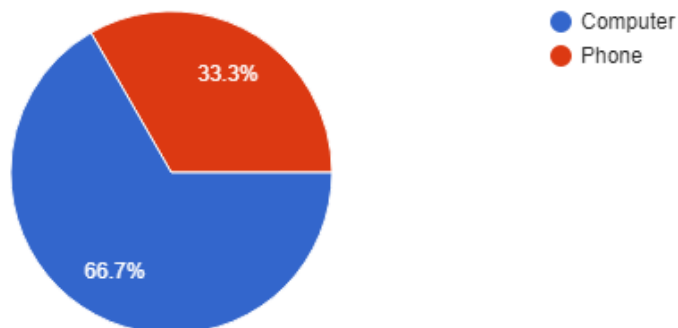
We wanted to know the opinion of users before we finalize the features of the game. We personally asked people some questions.

We have also conducted a survey and assembled its result. The link to result of the survey is: <https://goo.gl/forms/c2eBlb9ln0u16Wi42>

Here are the results:

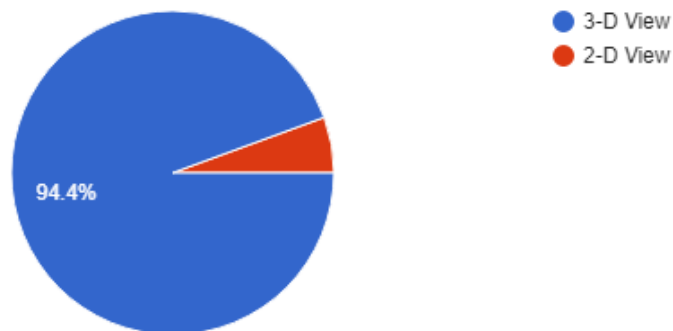
Do you like to play games on computer or phone?

9 responses



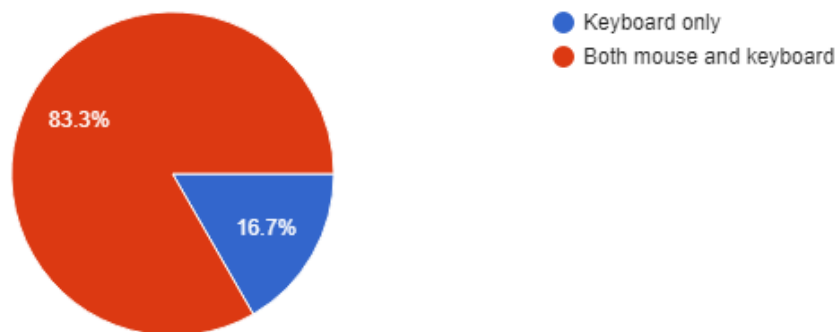
Would you like 3-D view of map or 2-D view of map?

18 responses



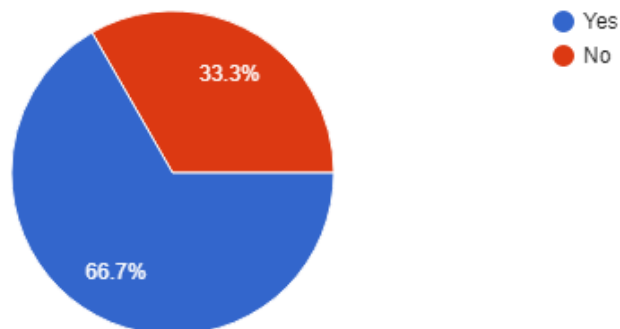
Do you prefer to play with keyboard only or with both mouse and keyboard?

18 responses



Would you like to share your high score on social media?

18 responses



Ques. What type of background music you would like in the game (or not want any music)?

Ans.

- User 1: No music, just maybe the sound of the footsteps.
- User 2: Option to add custom music.

- User 3: Any music would do.

Ques. What type of ques would you like to be asked while playing the game?

Ans.

- User 1: About the history of the department or various parts of the department.
- User 2: Related to the path used, for e.g., the shortest distance from one path to another.
- User 3: Name of the professor, name of the lab, way back to entrance.

Ques. Would you like to give any other suggestions for the game?

Ans.

- User 1: No, it's seem pretty good.
- User 2: Extend the map to other departments or may be for whole IITG.
- User 3: Allow us to change external environment like weather.

Seeing the results of the survey, we decided to add few features to the game.

The end-users that participated in the research were informed about it and its purpose. Users were informed about the purpose of the study and were asked to participate in the research at the beginning of the observation session. It was ensured that the users were not affected by the presence of the observer in an unobstructed way, since the survey was send individually to all the users.

6. Other Requirements

Appendix A: Abbreviations

IEEE: Institute of Electronics and Electrical Engineer
IITG: Indian Institute of Technology Guwahati
CSE: Computer Science and engineering
IC: Inbound Contingency (name of the game)
GPU: Graphics Processing Unit
GB: Gigabytes
SDK: Software Development Kit
RAM: Random Access Memory
2D, 3D: Two dimensional, Three Dimensional
CPU: Central Processing Unit
CI: Contextual Inquiry

Appendix B: Tools Used

Visual Studio: Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as web sites, web apps, web services and mobile apps.

Autodesk 3ds Max: Autodesk 3ds Max is a professional 3D computer graphics program for making 3D animations, models, games and images.

Unity (Game Engine): Unity is a cross-platform game engine developed by Unity Technologies, which is primarily used to develop both three-dimensional and two-dimensional video games and simulations for computers, consoles, and mobile devices