SOFTWARE REQUIREMENTS SPECIFICATION

for

Majuli River Island virtual tour (interactive)

Version 1.1
February 6, 2023

Prepared by
1. Pruthvi Raj G N (200101088)
2. Gandrathi Srija (200101030)
3. Yakkala Hema Venkat Rao (200101111)
4. Advaita Mallik (200101010)
5. Kanchumarthi H Pravallika (200101049)

IIT Guwahati

supervised by Prof. Samit Bhattacharya

Contents

1 Introduction

1.1 Purpose

The purpose of this document is to give a detailed description of the requirements for the Majuli River Island virtual tour (interactive). It will illustrate the purpose and complete declaration for the development of the system along with the system constraints. This document is primarily intended to be proposed to a customer for their approval and as a reference for developing the first version of the system for the development team.

1.2 Document Conventions

Term	Definition
Device	An electronic device using which the user does the tour
DESC	Description
TourItem	The place where we are visiting in the village
TourItemPage	The page corresponding to TourItem
VR	Virtual Reality

1.3 Project Scope

This software is meant to be deployed as a real-world application where the visitor can conveniently tour the Majuli River Island. This software shall allow the visitor to witness the village culture, tradition, and daily life activities through a lightweight virtual reality system.

2 Overall Description

2.1 Product Perspective

The Majuli River Island virtual tour is a Google Cardboard-enabled virtual reality
system, used for the following two reasons:
1. To promote tourism
2. To preserve the cultural heritage and traditions of the Jangraimukh tribal
(Mishing) village.

2.2 Product Functions

The virtual tour begins by moving the user's perspective from the globe to the India map, to Majuli River Island. On entering the island, the virtual tour uses multimedia, including images, audio and video to give a comprehensive idea of the island, showing the important landmarks, geographical relief, ecology and the daily lives of the locals. Ambient noise is used to increase immersion.

The virtual reality tour allows easy traversal through the Jangraimukh village allowing teleportation to any location of their choice, skipping and revisiting locations. On exit, the state of the user is saved for future use. Feedback is taken at the end of the tour to improve the system.

The application also provides an audio-visual experience with background sounds relevant to the user's current location.

2.3 User Classes and Characteristics

"Majuli River Island virtual tour" has one type of user.

- Visitor

2.4 Operating Environment

The environment is a Google Cardboard-enabled VR system, expected to be run on a smartphone that supports audio/video.

2.5 Design and Implementation Constraints

The beginning of the tour consists of changing perspectives from the globe to the river island. This is an unskippable animation that will play every time a user visits.
Since Google Cardboard is a mobile VR headset, the VR system should be optimized for mobile devices to ensure a smooth and responsive experience. Most videos are played in standard definition, and 3D objects are less detailed and more limited than in Oculus versions of the software.
Latency might be an issue in VR systems, and it is important to ensure that VR systems provide low-latency experiences to prevent motion sickness. This may require optimizing the graphics engine and reducing the computationally intensive tasks.
The VR system should be compatible with popular mobile operating systems, including Android and iOS. This might require testing the VR system on multiple devices and operating systems to ensure compatibility.
The VR system should be compatible with Google Cardboard API, which requires following Google's guidelines for VR development.

2.6 Assumptions and Dependencies

One assumption about the product is that it will always be used on mobile phones that have enough performance. If the phone does not have enough hardware resources available for the application, for example the users might have allocated them with other applications, there may be scenarios where the application does not work as intended or even at all.

3 External Interface Requirements

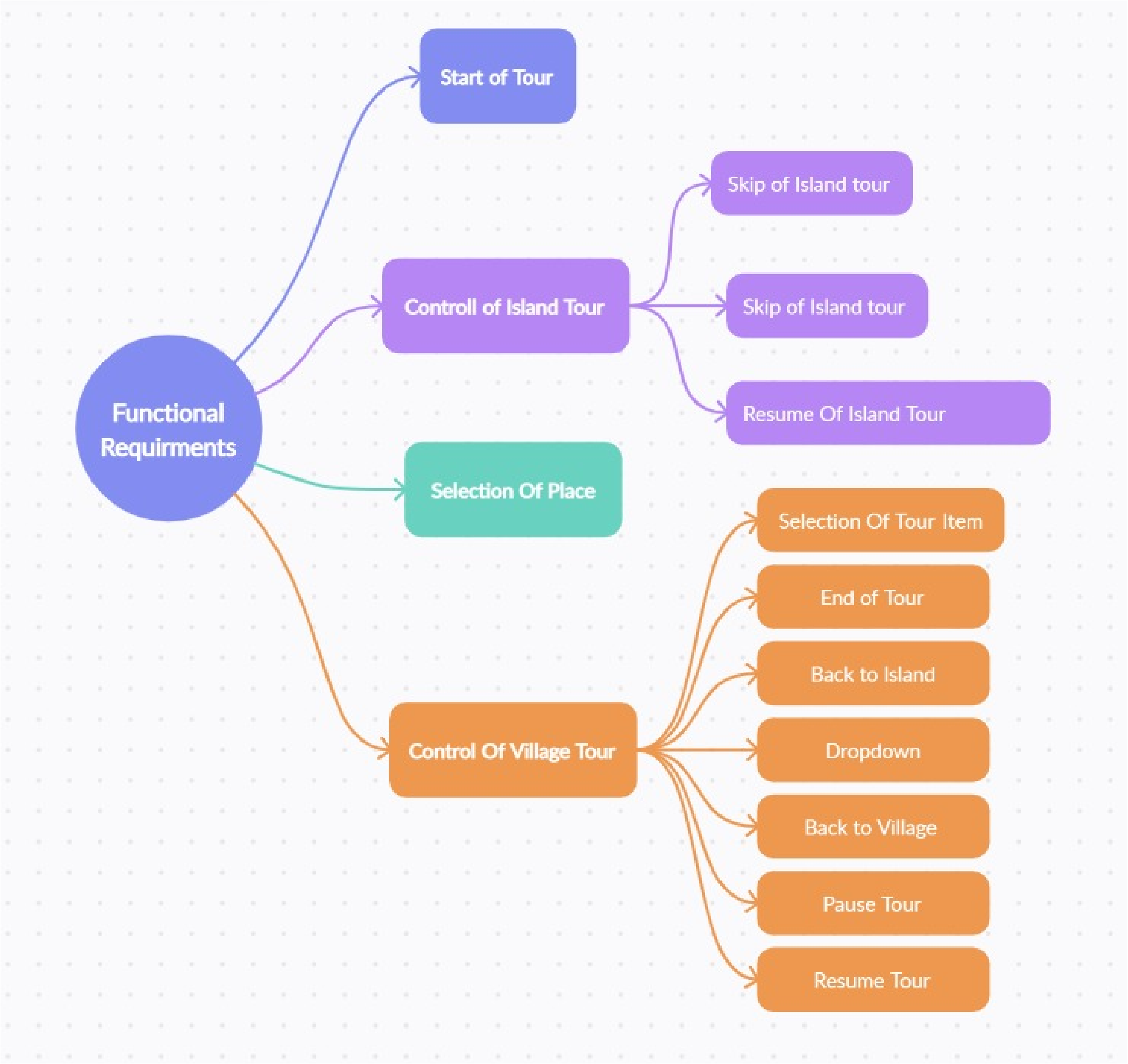
3.1 Software Interfaces

The user's smartphone should have enough performance and storage to be able to run the software.
The smartphone should support Google Cardboard. Alternatively, it should support software that supports Google Cardboard.

3.2 Hardware Interfaces

The VR system should be compatible with a wide range of mobile devices, including smartphones and tablets. The VR system should support popular mobile operating systems, such as Android and iOS, and be optimized for different screen sizes and resolutions.

4 Functional Hierarchy



5 Functional Requirements

5.1 StartOfTour

ID: FR_1
INPUT: "Start Tour" button
OUTPUT: Animation of perspective change from the globe to island map
DESC: When the "Start Tour" button is clicked, the user's perspective is moved from the globe to the island map using an animation.

5.2 ControlOfIslandTour

1 SkipOfIslandTour

ID: FR_2.1
INPUT: "Skip" button
OUTPUT: "Selection Of Place" page
DESC: When the "Skip" button is clicked, the user is navigated to the "Selection Of Place" page.

2 PauseOfIslandTour

ID: FR_2.2
INPUT: "Pause" button
OUTPUT: Pause of Island Tour
DESC: When the "Pause" button is clicked, the video tour is paused

and the "Pause" button changes to the "Resume" button.

3 ResumeOfIslandTour

ID: FR_2.3
INPUT: "Resume" button
OUTPUT: Resume of Island Tour
DESC: When the "Resume" button is clicked, the paused video tour is resumed and the "Resume" button reverts to the "Pause" button.

5.3 SelectionOfPlace

ID: FR_3
INPUT: "Jaingraimukh Village" icon
OUTPUT: Homepage of Village Tour
DESC: When the "Jaingraimukh Village" icon is selected, the user is navigated to the homepage of Village Tour.

5.4 ControlOfVillageTour

1 SelectionOfTourItem

ID: FR_4.1
INPUT: Tour item
OUTPUT: Tour item page
DESC: Selecting any item in the table will teleport the user to the selected location in the village.

2 EndOfTour

ID: FR_4.2
INPUT: "End Tour" button
OUTPUT: "Thank You and Feedback" page
DESC: When the "End Tour" button is pressed, the user is navigated to the "Thank You and Feedback" page where the user is requested to provide feedback about their experience.

3 BackToIsland

ID: FR_4.3 INPUT: "Back to Island" button
OUTPUT: Homepage of "Island Tour"
DESC: When the "Back to Island" button is pressed, the user is navigated back to the Island Tour homepage (4.2).

4 Dropdown

ID: FR_4.4
INPUT: Tour item
OUTPUT: Tour item page
DESC: Selecting any item in the dropdown will teleport the user to the selected location in the village.

5 BackToVillage

ID: FR_4.5
INPUT: "Back to Village" button
OUTPUT: Homepage of "Village Tour"
DESC: When the "Back to Village" button is pressed, the user is navigated back to the Village Tour homepage (4.4.1).

6 PauseTour

ID: FR_4.6
INPUT: ”Pause” button
OUTPUT: Pause of Tour
DESC: When the ”Pause” button is clicked, the video tour is paused and the ”Pause” button changes to the ”Resume” button.

7 ResumeTour

ID: FR_4.7
INPUT: ”Resume” button
OUTPUT: Resume of Tour
DESC: When the ”Resume” button is clicked, the paused video tour is resumed and the ”Resume” button reverts to the ”Pause” button.

6 Other Nonfunctional Requirements

6.1 Performance Requirements

- The VR system should render high-quality graphics in real-time, with support for various graphics modes, including low, medium, and high-quality graphics. The VR system should be able to run on a wide range of mobile devices, including both high-end and entry-level devices.
- The VR system should provide high-quality audio, with support for background music and sound effects. The audio should be adjustable by users, with options for turning off music or sound effects as desired.
- The VR system should respond quickly to user inputs, with minimal latency and lag. This is important to ensure a smooth and immersive virtual reality experience.
- The VR system's text material should be legible and easily understandable. This is important to ensure smooth user-interface communication.

6.2 Security Requirements

As no user data is being requested by the application, no security precautions or encryption of data is necessary.

6.3 Sofware Quality Attributes

- Code should be modular and well-written, so as to facilitate further addition and implementation of new features and improvements.
- The system should be robust enough not to fail during the tour.

6.4 Other Requirements

As system software keeps on updating, our virtual tour app needs maintenance on a regular basis. It will need re-factoring and further, the requirements can be changed as the field is changing frequently.

7 Usability Requirements to Functional Requirements

Usability requirements include pausing the tour and sometimes skipping sections of the tour. For that, we have made the PauseTour function. Skipping has been implemented with SkipOfIslandTour for skipping different parts of the Island Tour.

We are allowing an easy exit from the application with the EndTour function.

To allow the user to skip, start from or teleport to any Tour Item, we offer the Dropdown function that allows the user to select their place of interest.

If a user wishes to view other parts of the Island after already entering the Village homepage, we offer a function BackToIsland to allow the user to return to the Island homepage in one step.

Similar to the BackToIsland function is the BackToVillage function that allows a user to move from any Tour Item's page to the Village homepage.