Software Requirements Specification

for

Mumbai

Version 1.0 approved

Prepared by Saideep Dicholkar

Adil Khatri

Chandan Choudhary

<organization>

<date created>

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

3. External Interface Requirements 3

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

4. System Features 4

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

5. Other Nonfunctional Requirements 4

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

6. Other Requirements 5

Appendix A: Glossary 5

Appendix B: Analysis Models 5

Appendix C: To Be Determined List 6

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

Our product satisfies the needs of Mumbai tourism. It will be helpful for tourists visiting Mumbai.

## Document Conventions

The document is developed based on the IEEE SRS standard.

## Intended Audience and Reading Suggestions

It is developed for tourists visiting Mumbai or people who want to know more about Mumbai. Also, students and developers can take advantage of this website and start web design and developing new websites.

## Product Scope

Our product satisfies the needs of Mumbai tourism. It will be helpful for tourists visiting Mumbai. This website will provide tourists with best hotels to stay as well as places for shopping, places to visit and provides some history about Mumbai.

## References

This System Requirement Specifications document is based on the guidelines of IEEE SRS format and refers to the guidelines set by IEEE.

# Overall Description

## Product Perspective

There are few existing web sites about Mumbai tourism but our web site provides short and crisp information. It provides maximum information in few clicks.

## Product Functions

It provides following details:

1) HOTELS TO STAY

2) PLACES TO VISIT

3) SHOPPING PLACES

4) HISTORY OF MUMBAI

5) PHOTOS OF MUMBAI

6) MAP

## User Classes and Characteristics

It can be used by following users:

* Visitors

Users visiting website to find information about Mumbai

* Students

School students who are learning web development

* Developers

Open source for learning and doing further modifications as desired.

## Operating Environment

All Operating Systems.

## Design and Implementation Constraints

Firewall should not block social media sites as links are provided for Instagram for owners of respective photographs. There are no such hardware requirements except for given minimum requirements. It works best on Chrome Browser with minimum resolution of 1024 X 768.The website will be maintained by Developers themselves.

## User Documentation

User Documentations such as on-line help, documentation, manual will be provided along with the website for easy use by user and troubleshooting any issues.

## Assumptions and Dependencies

We are assuming that user will have an internet connection along with a PC / Laptop with the required minimum specifications. Also, if any firewall used by user shouldn’t block social media sites which might lead to user not able to view the original photographer of photograph used in website. User’s browser should allow running scripts as the website uses JavaScript and having Allow Script option turned off might lead to issues and not proper functioning of some functionalities. In future, we are thinking of using database to store and retrieve data for the website.

# External Interface Requirements

## User Interfaces

The User Interface is inspired by and based on Google’s Material Design Standards. The Material Design Standards states that the components are like layers of papers. They are flat as paper and have depth (shadow) depending on their height. Whenever a cursor is hovered on some component it’s depth increases giving a visual look of component being raised high. All this leads to a visually appealing website for users. The material design can be applied to buttons, images to look like cards and floating buttons and to other components as needed. **<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>**

## Hardware Interfaces

The website can work on and be viewed on all Desktops based systems and on Mobile devices. The data required can be fetched from remote or local servers. **<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware, and communication protocols to be used.>**

## Software Interfaces

The website uses PHP for connecting to Database. Database used is MySQL. Both PHP and MySQL are used to store the feedback taken from user and to send reply to the user accordingly. The feedback is taken from a Feedback form which includes user’s Name, E-mail and Message (Feedback). These entities get stored in our database.

For Maps we have used Google Maps API from Google for displaying Map on Website. This Map contains the API key which calls the Map API from Google Cloud Platform to display the Google Maps Data. **<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.>**

## Communications Interfaces

The website will use Database Server for storing the data and PHP for connecting the Database Server with Feedback form. HTTPS protocol will be used by website for communication and security purpose. **<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.>**

# System Features

<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

## System Feature 1

<Don’t really say “System Feature 1.” State the feature name in just a few words.>

4.1.1 Description and Priority

<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, pen

alty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>

4.1.2 Stimulus/Response Sequences

<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>

4.1.3 Functional Requirements

<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>

REQ-1:

REQ-2:

## System Feature 2 (and so on)

# Other Nonfunctional Requirements

## Performance Requirements

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.>

## Safety Requirements

<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied.>

## Security Requirements

<Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. Refer to any external policies or regulations containing security issues that affect the product. Define any security or privacy certifications that must be satisfied.>

## Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.>

## Business Rules

<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>

# Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

Appendix C: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>