**FYP SRS Document Template**

Final Year Project

Software Requirement Specification

For

XPK Application

BSCS

By

| S# | **Name** | **Registration #/Roll #/Section** | **Mobile #** | **E-Mail** |
| --- | --- | --- | --- | --- |
| 1. | Abdulmanan Nazir | 17277/FA-2021/BSCS/287#/F | 03174689617 | mrmanan143@gmail.com |
| 2. | Iqra Fateh | 17294/FA-2021/BSCS/304#/F | 03097002398 | iqra133660@gmail.com |

**Supervised by:**

Teacher Name **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (**Signature**)**



Department of Computer Science

Lahore Garrison University

Lahore

Table of Contents

1. Introduction........................................................................................................... Page#

1.1 Purpose ........................................................................................................................

1.2 Document Conventions ................................................................................................

1.3 Intended Audience and Reading Suggestions ..............................................................

1.4 Product Scope ..............................................................................................................

2. Overall Description ...............................................................................................

2.1 Product Perspective......................................................................................................

2.2 Product Functions ......................................................................................................

2.3 User Classes and Characteristics ...............................................................................

2.4 Operating Environment ..............................................................................................

2.5 Design and Implementation Constraints ....................................................................

2.6 User Documentation ..................................................................................................

2.7 Assumptions and Dependencies ................................................................................

3. External Interface Requirements .......................................................................

3.1 User Interfaces ...........................................................................................................

3.2 Hardware Interfaces ...................................................................................................

3.3 Software Interfaces ....................................................................................................

3.4 Communications Interfaces .......................................................................................

4. System Features ..................................................................................................

4.1 System Feature 1 ........................................................................................................

4.2 System Feature 2 (and so on) .....................................................................................

5. Other Nonfunctional Requirements ...................................................................

5.1 Performance Requirements ........................................................................................

5.2 Safety Requirements ..................................................................................................

5.3 Security Requirements ...............................................................................................

5.4 Software Quality Attributes .......................................................................................

5.5 Business Rules ...........................................................................................................

References ....................................................................................................................

# **1. Introduction**

## 1.1 **Purpose**

The purpose of this document is to define the software requirements for **Explore Pakistan (XPK)**, a mobile application designed to facilitate tourism in Pakistan. The application enables users to explore cities, plan trips based on their available time, upload vlogs/blogs, and discover nearby attractions. The scope includes creating a user-friendly platform with features tailored for tourists, including trip recommendations, location-based services, and social sharing functionalities.

## **1.2 Document Conventions**

* Headings are numbered for hierarchy.
* Bold text represents important terms.
* Functional requirements are labeled as REQ-1, REQ-2, etc.
* Placeholder text, where applicable, is marked as TBD (To Be Determined).

## **1.3 Intended Audience and Reading Suggestions**

This document is intended for:

* **Developers**: For understanding functional and non-functional requirements.
* **Project Managers**: To track deliverables and scope.
* **Testers**: To develop test cases based on specified requirements.
* **End Users**: To gain insights into the features and functionalities of the application.
* **Stakeholders**: To ensure alignment with business goals.

It is recommended to start with the **Introduction** section and then move to **System Features** for detailed functionalities.

## **1.4 Product Scope**

Explore Pakistan (XPK) aims to:

* Provide an all-in-one platform for tourists to explore Pakistan.
* Offer tailored recommendations based on user preferences, such as budget, trip duration, and location.
* Enable social interaction by allowing users to upload and share vlogs/blogs.
* Enhance user experience through AI-powered trip planning and location-based services.

# **2. Overall Description**

## **2.1 Product Perspective**

The product is a standalone mobile application designed for both local and international tourists. It integrates:

* **Google Maps API**: For location-based services.
* **Firebase**: For database, authentication, and storage.
* **AI Features**: For personalized recommendations.

## **2.2 Product Functions**

* Search and view tourist spots by city.
* Plan trips based on time and budget.
* Upload vlogs/blogs with multimedia.
* View itineraries and nearby attractions.
* User authentication and profile management

## **2.3 User Classes and Characteristics**

* **Tourists**: Casual users looking for travel guidance.
* **Content Creators**: Users uploading vlogs/blogs

## **2.4 Operating Environment**

* **Mobile Platforms**: Android and iOS.
* **Backend**: Node.js with MongoDB.
* **APIs**: Google Maps, Firebase, ZEGOCLOUD for communication.

## **2.5 Design and Implementation Constraints**

<Describe any items or issues that will limit the options available to the developers. These might include corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer’s organization will be responsible for maintaining the delivered software).>

## **2.6 User Documentation**

* User guide for app navigation.
* FAQs and troubleshooting section.

## **2.7 Assumptions and Dependencies**

* Users will have active internet connections.
* Third-party APIs (e.g., Google Maps) will remain operational.
* Smartphones with GPS support are used.

# **3. External Interface Requirements**

## **3.1 User Interfaces**

* Intuitive and responsive UI/UX.
* Screens for trip planning, profile management, and vlog/blog uploads etc.

## **3.2 Hardware Interfaces**

Smartphones with minimum 2GB RAM and GPS.

laptop for development minimum 256 SSD 12 RAM

## **3.3 Software Interfaces**

* **Google Maps API**: For location services.
* **Firebase**: For backend services.
* **ZEGOCLOUD API**: For chat and video call functionalities.

## **3.4 Communications Interfaces**

* Secure communication via HTTP.
* Real-time messaging

# **4. System Features**

### 4.1 Trip Planning

#### 4.1.1 Description and Priority

Allow users to plan trips based on available time, budget, and preferences. **Priority: High**.

#### 4.1.2 Stimulus/Response Sequences

* User selects city and inputs preferences.
* System generates trip recommendations.

#### 4.1.3 Functional Requirements

* REQ-1: Provide trip suggestions based on user inputs.

# **5. Other Nonfunctional Requirements**

## **5.1 Performance Requirements**

* Response time should be under 2 to 3 seconds for all operations.
* Support 1,000 concurrent users.

## **5.2 Safety Requirements**

* Data backups every 24 hours.
* Compliance with GDPR for user data.

## **5.3 Security Requirements**

* Two-factor authentication.
* End-to-end encryption for communications.

## **5.4 Software Quality Attributes**

* **Usability**: Easy navigation and user-friendly interface.
* **Reliability**: 99.9% uptime.
* **Maintainability**: Modular code for easy updates.

## **5.5 Business Rules**

* Content moderation by admins.
* User roles and permissions: Tourists, Content Creators, Admins.

# **References**

* Google Maps API Documentation.
* Firebase Documentation.
* ZEGOCLOUD API Documentation.
* Node Documentation
* mongo DB Documentation