Title : Tourism Webapplicaton

Software Requirements Specification(SRS)

24-04-2024

GOPI KUMAR

Github Link :- [Git1811/INT-219-Project (github.com)](https://github.com/Git1811/INT-219-Project)

Prepared for

Continuous Assessment 3

2024

Table of Contents

1. Introduction 4

1.1 Purpose 4

1.2 Scope 4

1.3 Definitions, Acronyms, and Abbreviations 5

1.4 Overview 5

2. General Description 6

2.1 Product Functions 7

2.2 User Characteristics 8

2.3 General Constraints 8

2.4 Assumptions and Dependencies 8

3. Specific Requirements 9

3.1 Functional Requirements 9

3.2 Non-Functional Requirements 10

3.3 Design Constraints 10

4. Analysis Models 11

4.1 Screenshots…………………………………………………………………………………………………………………………………………..11

4.2 Data Flow Diagrams (DFD) 13

5. Github link…………………………………………………………………………………………………….15

A. Appendices 15

A.1 Appendix 1 15

Table of Figures

|  |  |
| --- | --- |
| Fig No.: Detail | Page No. |
| Fig.1: Home Page | 7 |
| Fig.2: Packages | 8 |
| Fig.3: Best International’s | 9 |
| Fig.4: Jackpot Deals | 10 |
| Fig.5: Explore The Unexlored | 11 |

Introduction

EXPLORRRR is an innovative tourism website designed to revolutionize the way people plan and book their travel experiences. It offers a seamless platform for users to book plane tickets, arrange comprehensive tourism packages, secure hotel accommodations, and access food catering services—all with significant discounts. The website is geared towards making travel planning more convenient and cost-effective for users, whether they are booking a quick domestic flight or a detailed international vacation. By integrating multiple travel-related services, EXPLORRRR aims to be a one-stop solution for travelers, providing them with a user-friendly interface, robust customer support, and reliable payment processing, all within a secure and scalable web platform.

1.1 Purpose

EXPLORRRR is designed to be a comprehensive tourism platform that simplifies the travel planning process for users. Its primary purpose is to provide a one-stop solution for booking plane tickets, arranging tourism packages, securing hotel accommodations, and accessing food catering services—all at competitive prices. By integrating these services into a single user-friendly website, EXPLORRRR aims to streamline the travel experience, offering customers convenience, affordability, and a reliable resource for both national and international travel needs.

1.2 Scope

EXPLORRRR is a web-based platform designed for tourists to book plane tickets, purchase tourism packages, and reserve hotel accommodations. The website targets national and international travelers, providing a comprehensive and user-friendly booking experience. The scope includes all features, functionalities, system architecture, and technical constraints.

## Definitions, Acronyms, and Abbreviation

* Definitions Acronyms and Abbreviations  
  **SRS :** Software Requirements Specification - This document outlining the requirements and specifications of the Student College Account Management System.  
  **HTML**  Hypertext Markup Language - The standard language for creating web pages and applications.  
  **CSS :** Cascading Style Sheets - Used to style the visual presentation of HTML elements.  
  **JAVASCRIPT :** A programming language used to add interactivity and dynamic behavior to web pages.  
  **Angular :** A JavaScript framework for building web applications, including our Student College Account Management System.  
  **UI :** User Interface - The visual elements and layout of our system that users interact with.  
  Bootstrap: A front-end framework for developing responsive and mobile-first websites and web applications.  
  **PDF:** Portable Document Format - A file format used for documents, such as notes and other downloadable resources within our system.  
  **URL:** Uniform Resource Locator - Web address used to access specific resources within our system.  
  **HTTP**: Hypertext Transfer Protocol - The protocol used for transferring data over the web, such as fetching notes or accessing announcements.  
  **API:** Application Programming Interface - Defines interactions between our Angular application and other software components, such as retrieving data from the database.  
  **UX:** User Experience - The overall experience and satisfaction a user has when interacting with our system, influenced by its design and functionality.

1.4 Overview

This SRS document outlines the product features, user characteristics, and specific requirements for EXPLORRRR. It aims to guide the development team through the project implementation and ensure all stakeholders have a clear understanding of the system's objectives and constraints.

1. General Description
   1. Product Functions

EXPLORRRR provides the following core functions:

**Flight Booking:** Book national and international plane tickets with discounted prices.

**Tourism Packages:** Purchase comprehensive packages including flights, hotels, and catering services.

**Hotel Booking:** Reserve hotel accommodations either as part of a package or individually.

**Customer Support:**

Contact support for booking assistance or queries.

**Payment Processing:**

Secure payment gateway for booking and package purchases.

2.2 User Characteristics

* Customers:

Users who book flights, tourism packages, and hotels. They may have varying technical skills and require a user-friendly interface. Administrators: Website administrators responsible for managing content, pricing, and customer support.

* Third-party Services:

External services providing flight and hotel booking APIs and other necessary integrations.

2.3 General Constraints

* Platform Constraints: The website must be compatible with modern web browsers (e.g., Chrome, Firefox, Safari, Edge).
* Security Constraints: Compliance with industry standards for data protection and payment processing.
* Scalability Constraints: The system must support high user traffic and simultaneous bookings.

2.4 Assumptions and Dependencies

The website will be developed using standard web technologies such as HTML, CSS, and JavaScript.

External APIs for flight and hotel bookings will be accessible and stable.

Hosting infrastructure will be reliable and provide redundancy for high availability.

User data will be protected according to industry best practices.

3. Specific Requirements

3.1 Functional Requirements

* User Registration and Authentication:

Users must be able to create accounts and log in securely.

Password reset functionality must be available.

* Flight Booking:

Users can search for flights by destination, date, and other parameters.

Users can book flights and receive booking confirmations.

* Tourism Packages:

Users can view available tourism packages, including details of flights, hotels, catering services.

Users can book packages and receive booking confirmations.

* Hotel Booking:

Users can search for and reserve hotel rooms.

* Payment Processing:

Secure payment processing with credit/debit cards or electronic methods.

Compliance with PCI DSS standards.

* Customer Support:

Contact forms and other communication methods for customer support.

Frequently Asked Questions (FAQ) section for common questions.

3.2 Non-Functional Requirements

* Performance:

The website should load quickly and respond to user interactions within acceptable timeframes.

* Security:

All user data should be encrypted in transit and at rest.

Compliance with security standards for data protection and payment .

* Scalability:

The website must be able to handle high user traffic and multiple

* Usability:

The user interface should be intuitive and easy to use for a wide range of .

* Reliability:

The website should have minimal downtime and reliable backups to prevent .

3.3 Design Constraints

* Technology Stack:

The frontend should be developed using HTML, CSS, and JavaScript.

The backend should use a scalable architecture, integrating with external APIs for flight and hotel bookings.

* Hosting:

The website must be hosted on a secure platform with redundancy and backup mechanisms in place.

* Compliance:

The website must comply with relevant laws and industry standards for tourism, e-commerce, and data protection.

4. Analysis Models

4.1 Screenshots

Fig1: Home Page



Fig2: Packages



Fig3: Best International’s

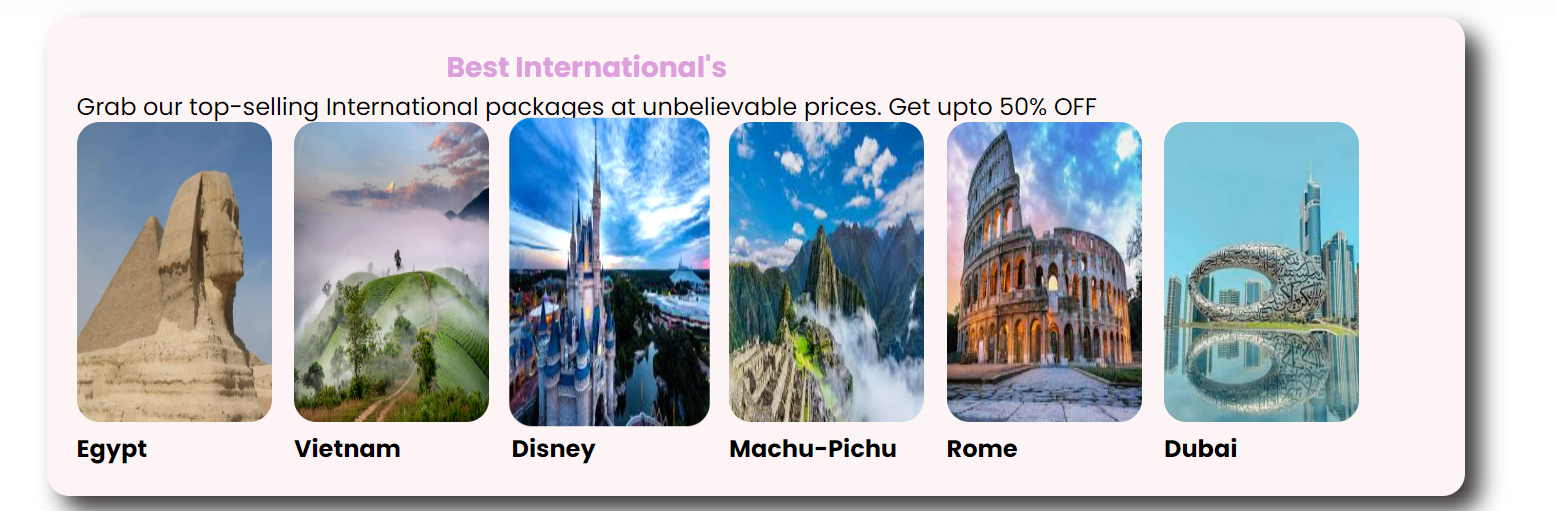
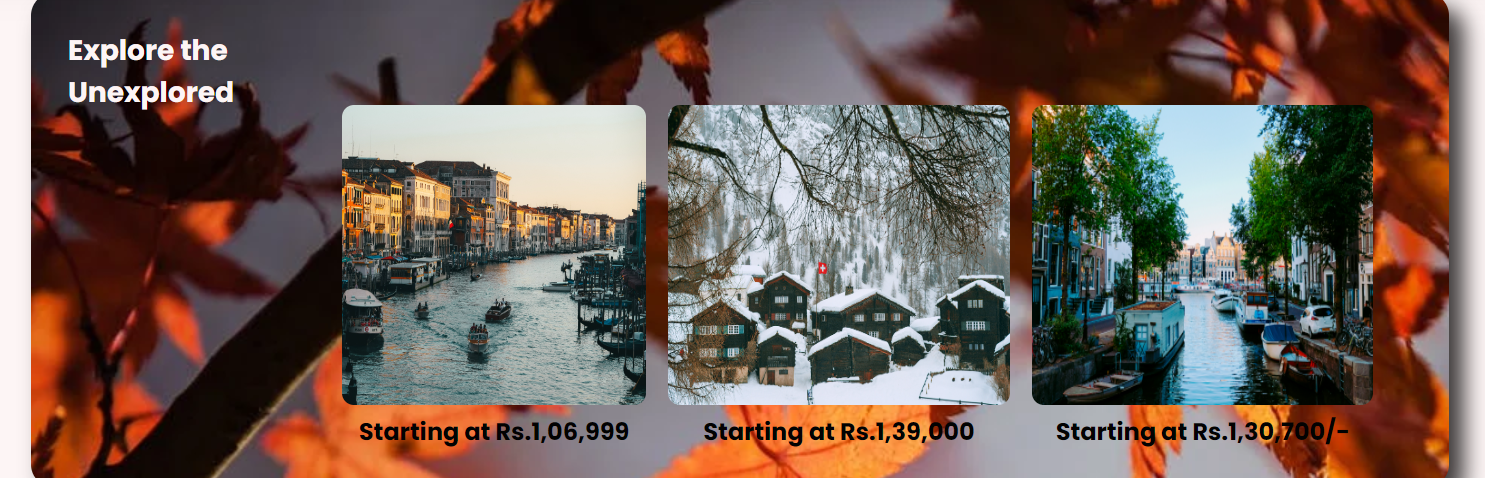


Fig4: Jackpot Deals



Fig5: Explore the Unexplored



4.2 Data Flow Diagrams (DFD)

Introduction:

Data Flow Diagrams (DFDs) are graphical representations of the flow of data within a system. They illustrate how data moves through various processes and interactions within the system. In the context of the Student College Account Management System, DFDs are used to visualize the flow of information related to attendance tracking, exam schedules, announcements, timetables, and other functionalities.

In my project there is no backend till now in it but I can tell how DFD looks like if I added in it.

Narrative Description:

Level 0 DFD (Context Diagram)

This high-level diagram shows the system as a single process interacting with external entities such as users (students, teachers, administrators), external databases, and the college infrastructure.

It provides an overview of the entire system and its interactions with external entities, without going into detail about internal processes.

Level 1 DFDs

Attendance Tracking (DFD)

This DFD details the flow of data related to attendance tracking.

Processes include viewing attendance records, updating attendance, and generating attendance reports.

External entities are students and teachers interacting with the system.

Exam Schedule Management (DFD)

This DFD illustrates the flow of data for managing exam schedules.

Processes include uploading exam schedules, viewing schedules, and sending notifications.

External entities are students and administrators.

Announcements Display (DFD)

This DFD outlines the flow of data for displaying college announcements.

Processes include posting announcements, viewing announcements, and dismissing read announcements.

External entities are all users of the system.

Timetable Creation (DFD)

This DFD shows the flow of data for creating and managing personal timetables.

Processes include selecting courses, generating timetables, and printing or downloading timetables.

External entities are students.

Teacher Leave Updates (DFD)

This DFD illustrates the flow of data for updating teacher leave status.

Processes include updating leave, notifying students, and displaying leave information.

External entities are teachers and administrators.

Notes Downloading (DFD)

This DFD details the flow of data for accessing downloadable course materials.

Processes include uploading notes, downloading notes, and organizing materials.

External entities are students and teachers.

Community Group Feature (DFD)

This DFD outlines the flow of data for managing community groups and discussions.

Processes include creating groups, joining groups, posting messages, and receiving notifications.

External entities are users participating in group activities.

User Authentication (DFD)

This DFD shows the flow of data for user authentication and account management.

Processes include logging in, resetting passwords, and managing user profiles.

External entities are all users interacting with the system.

Traceability to SRS Requirements:

Each DFD model is directly linked to specific requirements outlined in the SRS.

For example, the DFD for "Attendance Tracking" is directly linked to the specific requirements related to attendance tracking functionality.

These DFDs provide a visual representation of how data flows through the system to fulfill the defined requirements.

1. Github Link: [Git1811/INT-219-Project (github.com)](https://github.com/Git1811/INT-219-Project)

A. Appendices

Appendices may provide additional information related to the project. If present, the SRS will explicitly state whether the information in the appendices is considered part of the overall set of requirements.

A.1 Appendix 1

This appendix may include initial conceptual documents for the software project, such as early design sketches, wireframes, or prototypes.

These documents can provide context and background information for the development process.