

Project Report

On

“Journey-Genie”

SUBMITTED TO

OURKELA INSTITUTE OF MANAGEMENT STUDIES

(As a Partial fulfilment of the requirement for the award of degree)

FOR

“MASTER IN COMPUTER APPLICATION “

(2023-25)

SUBMITTED BY

POOJA PRIYADARSHINI DAS

Registration Roll No:- 2305260014

MCA 4th SEMESTER

OURKELA INSTITUTE OF MANAGEMENT STUDIES

(Affiliated to Biju Patnaik University of Technology, Odisha)

Rourkela – 769015

Rourkela Institute of Management Studies

Rourkela

Department of Computer Science

Rourkela Institute of Management Studies

Chhend, Rourkela-15, Odisha

Phone:0661 2480482

Fax:91-0661-1480665

Mail: rkl_rimsgrol@sancharnet.in

Visit: www.rims-edu.com

CERTIFICATE OF EXAMINATION

This is to certify that this project report entitled "**Journey-Ginie**" submitted by **Pooja Priyadarshini Das** of 4th Semester, **Rourkela Institute of Management Studies, Rourkela**, is accepted as partial fulfillment of requirements for the degree in **Master in Computer Applications**, under **Biju Pattnaik University of Technology, Rourkela**, this has been verified by us and found be original up to our satisfaction.

External Examiner

Rourkela Institute of Management Studies

Rourkela

Department of Computer Science

Rourkela Institute of Management Studies

Chhend, Rourkela-15, Odisha

Phone:0661 2480482

Fax:91-0661-1480665

Mail: rkl_rimsgrol@sancharnet.in

Visit: www.rims-edu.com

CERTIFICATE OF EXAMINATION

This is to certify that this project report entitled "**Journey-Genie**" submitted by **Pooja Priyadarshini Das** of 4th Semester, **Rourkela Institute of Management Studies, Rourkela**, is accepted as partial fulfillment of requirements for the degree in **Master in Computer Applications**, under **Biju Pattnaik University of Technology, Rourkela**, this has been verified by us and found be original up to our satisfaction.

Internal Examiner

Rourkela Institute of Management Studies

Rourkela

Department of Computer Science

Rourkela Institute of Management Studies

Chhend, Rourkela-15, Odisha

Phone:0661 2480482

Fax:91-0661-1480665

Mail: rkl_rimsgrl@sancharnet.in

Visit: www.rims-edu.com

CERTIFICATE

This is to certify that this project entitled "**Journey-Genie**" has been and submitted by **Pooja Priyadarshini Das, M.C.A 2023-2025, Rourkela Institute of Management Studies, Rourkela**, has been examined by us. She is found fit and approved for the award of "**Master in Computer Application** "Degree.

To the best my knowledge this work has not been submitted for the award of any other degree.

I wish all success in her life.

DEAN ACADEMIC RIMS, ROURKELA



Prof. Bibhudendu Panda Head of The
Department, MCA
Rourkela Institute of Management Studies, Rourkela

CERTIFICATE

This is to certify that **Pooja Priyadarshini Das** student of **M.C.A,**
Rourkela Institute of Management Studies, Rourkela, Odisha of
Session 2023-2025 has completed the project successfully.

I wish all success in her life.

(Prof. Bibhudendu Panda)



DECLARATION

I, **Pooja Priyadarshini Das**, hereby declare that the project report Entitled "**Journey-Genie**" is of my work. The above work I submitted to "**Biju Patnaik University of Technology, Rourkela**" for the award of "**Master in Computer Applications**" Degree.

To the best of my knowledge, this work has not been submitted or published anywhere for the award of any degree.

Pooja Priyadarshini Das

ACKNOWLEDGEMENT

I am deeply indebted to **Rourkela Institute of Management Studies, Chhend, Rourkela**, for providing me an opportunity to undertake a project work entitled "**Journey-Ginie**".

I am grateful to my project guide **Prof. Bibhudendu Panda** without his guidance it would not have been possible on my part to complete the project. I acknowledge the help and co-operation received from all my team members in making this project.

I consider myself fortunate that I have successfully completed this project; I acknowledge my sincere gratitude to all those works and ideas that had helped me in writing this project.

Pooja Priyadarshini Das

University Roll No: 2305260014

MCA (2023-2025)

Rourkela Institute of Management Studies,
Rourkela.

Abstraction

Journey Ginie is a comprehensive travel and tour website designed to streamline the trip-planning experience for travelers worldwide. With an intuitive user interface, vibrant visuals, and interactive features, the platform allows users to effortlessly explore and book destinations, accommodations, and transportation. The website incorporates a seamless trip booking system, enabling users to select single or multiple destinations, compare prices, and customize their itineraries. With a secure registration process, budget-friendly options, and an easy-to-navigate dashboard, Journey Ginnie ensures a hassle-free and personalized travel experience. Its responsive design, secure payment gateway, and well-organized admin panel make it a reliable and efficient solution for both travelers and travel service providers.

The website goes beyond basic trip bookings by offering an accommodation section that covers flights, trains, buses, cabs, hotels, restaurants, spas, and malls, providing users with a comprehensive travel planning solution. Additionally, its robust admin dashboard allows travel service providers to manage destinations, bookings, payments, and user reviews efficiently. Journey Ginnie's modern UI, featuring a transparent navbar, dynamic widgets, and an elegant ombre light blue theme, enhances the overall user experience. Designed with scalability and security in mind, the platform caters to solo travelers, families, and business professionals, making travel planning effortless, secure, and enjoyable.

Table of Contents

| | | |
|-----|---------------------------------|-------|
| 1. | TITLE PAGE | 1 |
| 2. | CERTIFICATE OF EXAMINATION..... | 2 |
| 3. | CERTIFICATE OF EXAMINATION..... | 3 |
| 4. | CERTIFICATE..... | 4 |
| 5. | CERTIFICATE..... | 5 |
| 6. | DECLARATION..... | 6 |
| 7. | ACKNOWLEDGEMENT..... | 7 |
| 8. | ABSTRACT..... | 8 |
| 9. | TABLE OF CONTENTS..... | 10 |
| 10. | CHAPTER ONE:..... | 11-16 |
| • | INTRODUCTION | |
| • | Background | |
| • | Significance | |
| • | Purpose | |
| • | Motivation | |
| 11. | CHAPTER TWO : | 17-20 |
| • | LITRATURE REVIEW | |
| • | Challenges | |
| 12. | CHAPTER THREE : | 20-25 |
| • | SYSTEM REQUIREMENTS | |
| • | HARDWARE REQUIREMENTS | |

| | |
|---|-------|
| • SOFTWARE REQUIREMENTS | |
| 13. CHAPTER FOUR : | 24-33 |
| • TOOLS AND TECHNOLOGIES | |
| • HTML | |
| • CSS | |
| • SPRINGBOOT | |
| • JAVA | |
| • JAVASCRIPT | |
| • MYSQL | |
| 14. CHAPTER FIVE : | 33-41 |
| • SYSTEM DESIGN | |
| • Use Case Diagram | |
| • ER Diagram | |
| 15. CHAPTER SIX : | 41-45 |
| • SYSTEM ANALYSIS | |
| • Proposed System | |
| • Web Application | |
| • User Requirement | |
| • Admin | |
| • Inventory Control | |
| 16. CHAPTER SEVEN : | 46-49 |
| • IMPLEMENTATION | |
| • Primary Implementation Exploring Database | |
| 17. CHAPTER EIGHT : | 50-79 |

| | |
|------------------------------|-------------------|
| • Coding Section | |
| 18. CHAPTER NINE : |80-82 |
| • TESTING | |
| • Introduction | |
| • Unit Testing | |
| • Testing Scenarios | |
| 19. CHAPTER TEN : |83-85 |
| • CONCLUSION | |
| • Conclusion | |
| • Project Limitations | |
| 20. CHAPTER ELEVEN : |85-88 |
| • FUTURE ENHANCEMENTS | |
| • REFERENCES | |

Chapter 1:

1.1 INTRODUCTION

Background:

In an era where travel has become an integral part of personal and professional life, the need for a seamless, efficient, and reliable travel planning platform has never been greater. Journey Ginie was developed to bridge the gap

between complex travel arrangements and user-friendly digital solutions, offering a one-stop platform for booking trips, accommodations, and transportation. Traditional travel planning often involves navigating multiple websites, comparing prices across different providers, and managing scattered reservations. Journey Ginie eliminates these challenges by integrating all essential travel services into a single, intuitive platform, ensuring a hassle-free and time-saving experience for users. Whether for solo travelers, families, or business professionals, the website provides a structured and convenient way to explore destinations, customize itineraries, and make secure bookings, all while maintaining a visually engaging and interactive design.

One of the key advantages of Journey Ginie is its comprehensive and flexible trip booking system, which allows users to select single or multiple destinations, compare real-time pricing, and tailor their travel plans to fit their schedules and budgets. The Aadhaar-based registration system ensures authenticity and security while offering users a personalized experience with saved preferences and budget-friendly recommendations. Unlike traditional travel websites, Journey Ginie combines convenience with customization, enabling travelers to modify their itineraries, choose from a wide range of accommodations, and access additional services such as flights, trains, buses, cabs, hotels, restaurants, spas, and malls, all in one place. This comprehensive approach enhances user satisfaction by eliminating the need for third-party services and creating a streamlined, all-in-one travel ecosystem.

Beyond enhancing the user experience, Journey Ginie provides robust support for travel service providers and administrators through its advanced admin dashboard. This feature allows for efficient management of destinations, bookings, payments, customer reviews, and settings, ensuring that travel agencies and service providers can seamlessly operate and optimize their offerings. Additionally, the platform incorporates secure payment gateways, visually appealing UI elements, interactive widgets, and an ombre light blue theme, making it not only functionally superior but also aesthetically modern. By integrating cutting-edge technology, security features, and an intuitive interface, Journey Ginie stands as a future-ready, scalable, and user-centric travel solution that simplifies travel planning while ensuring a safe, efficient, and enjoyable experience for every traveler.

1.2 MOTIVATION

The inspiration behind Journey Ginnie stems from the growing demand for a seamless, efficient, and user-friendly travel planning platform that caters to modern travelers. With the rapid evolution of the tourism industry and the increasing reliance on digital solutions, travelers often find themselves overwhelmed by scattered information, multiple booking platforms, and complex planning processes. Journey Ginnie was developed to bridge this gap by offering an all-in-one, intuitive, and visually engaging solution where users can explore destinations, customize itineraries, book accommodations, and manage travel expenses effortlessly. By addressing the common pain points faced by travelers—

such as time-consuming research, lack of price transparency, and difficulty in managing multiple bookings—Journey Ginnie simplifies the journey from planning to execution, ensuring a stress-free and enjoyable travel experience.

Another key motivation behind Journey Ginnie is the desire to provide a personalized and secure travel experience. Many existing platforms lack an efficient system for user authentication and tailored recommendations. By incorporating Aadhaar-based registration, budget preferences, and flexible booking options, Journey Ginnie enhances both security and customization, allowing users to plan trips based on their unique needs and financial considerations. Additionally, the platform integrates a comprehensive accommodation section, covering flights, trains, buses, cabs, hotels, restaurants, spas, and malls, eliminating the need for travelers to visit multiple websites for different services. This streamlined approach saves time and enhances convenience, making travel planning more accessible, transparent, and hassle-free.

Beyond individual travelers, Journey Ginnie is also motivated by the need to empower travel agencies and service providers with a robust and efficient management system. The platform includes an advanced admin dashboard, enabling service providers to manage destinations, track bookings, handle payments, and monitor customer reviews efficiently. Additionally, with its secure payment gateway, modern UI elements, interactive widgets, and responsive design, Journey Ginnie stands as a future-ready, scalable platform that evolves with the changing needs of the travel industry. By integrating innovation,

security, and user-centric design, Journey Ginnie is not just a travel website—it is a revolutionary tool that transforms the way people explore the world.

1.3 PURPOSE

- Streamlining Travel Planning – Provide a seamless, intuitive, and efficient platform for booking trips, accommodations, and transportation, eliminating the complexities of traditional travel arrangements.
- Comprehensive Travel Solutions – Integrate a wide range of travel services, including flights, trains, buses, cabs, hotels, restaurants, spas, and malls, within a single platform to enhance user convenience.
- Optimized User Experience – Facilitate effortless destination selection, real-time price comparison, and itinerary customization, ensuring a personalized and hassle-free booking process.
- Enhanced Security & Transaction Integrity – Implement a robust and encrypted payment gateway, ensuring secure financial transactions and protecting user data from potential threats.
- Personalization & User Authentication – Incorporate Aadhaar-based registration, budget planning tools, and

preference-based recommendations to deliver a tailored travel experience.

- Empowering Travel Service Providers – Offer an advanced admin dashboard that enables seamless management of destinations, bookings, payments, customer reviews, and operational settings for travel agencies and service providers.
- Transparency & Accessibility – Provide real-time pricing updates, detailed package information, and a user-centric interface, ensuring clarity and ease of navigation for travelers.
- Engaging & Aesthetic User Interface – Utilize a visually appealing design, transparent navigation bar, dynamic grid layouts, and an ombre light blue theme, enhancing user engagement and overall experience.
- Efficiency & Time Optimization – Eliminate the need for third-party platforms by offering a fully integrated, all-in-one travel ecosystem, saving users significant time and effort.
- Scalability & Future-Readiness – Develop a versatile, adaptive, and technologically advanced platform that evolves with industry trends, ensuring long-term sustainability and continuous innovation in the travel sector.

Chapter 2:

2.1 LITRATURE REVIEW

The development of digital travel platforms has transformed the tourism industry, enabling seamless trip planning and booking experiences. Numerous studies highlight the significance of user-friendly interfaces, personalized travel recommendations, and secure payment systems in enhancing customer satisfaction. According to research on digital tourism, modern travelers seek convenience, transparency, and customization, leading to the rise of AI-driven and data-centric platforms that offer real-time pricing, itinerary suggestions, and tailored experiences (Buhalis & Law, 2008).

Existing travel websites, such as Booking.com, Expedia, and TripAdvisor, have successfully incorporated features like dynamic search filters, customer reviews, and multi-destination trip planning. However, gaps remain in terms of integrated services, personalized budget planning, and a centralized booking system. Journey Ginnie aims to address these gaps by offering a comprehensive, all-in-one travel ecosystem that combines destination booking, accommodation selection, transportation services, and budget management within a single platform. Furthermore, the platform integrates an Aadhaar-based registration system, ensuring authenticity and security, which is not widely available in existing travel platforms.

Additionally, the literature emphasizes the growing role of mobile responsiveness, engaging UI/UX design, and AI-powered recommendations in modern travel platforms. Journey Ginnie aligns with these trends by incorporating a transparent and visually appealing interface, interactive widgets, and an ombre light blue theme to enhance user engagement. While digital travel platforms continue to evolve, Journey Ginnie positions itself as a scalable, future-ready solution that leverages industry best practices while addressing key challenges in online trip booking and management.

2.2 Challenges

Despite its innovative approach, Journey Ginnie faces several challenges that impact its implementation and operational efficiency:

1. **User Trust & Data Security** – With growing concerns about data breaches and online fraud, ensuring secure payment transactions and Aadhaar-based authentication without compromising user privacy remains a challenge.
2. **Competitive Market Landscape** – The travel industry is dominated by established giants like Expedia, MakeMyTrip, and Booking.com, making it difficult for new platforms to gain user trust and establish a unique value proposition.

3. Seamless Integration of Multiple Services – Offering a unified travel experience requires seamless coordination between flight operators, hotels, transport services, and travel agencies, which can be complex due to differing API standards and booking systems.
4. Real-Time Price Updates – Maintaining accurate and dynamic pricing for multiple travel services requires robust backend infrastructure and real-time synchronization with third-party data providers.
5. User Experience & Load Performance – With multiple interactive elements, ensuring fast-loading pages, smooth animations, and mobile responsiveness can be challenging, especially for users accessing the website from low-bandwidth regions.
6. Regulatory & Compliance Challenges – The platform must adhere to international travel regulations, data protection laws, and financial transaction policies, which vary across different countries and regions.
7. Scalability & Future Expansion – As the platform grows, handling an increasing number of users, transactions,

and destinations will require robust cloud-based infrastructure and AI-driven optimizations.

8. Customer Support & Dispute Resolution – Addressing issues related to booking cancellations, refunds, and service complaints requires an efficient customer support framework, which can be resource-intensive.

9. Marketing & User Adoption – Attracting and retaining users in a competitive market requires aggressive digital marketing, partnerships, and loyalty programs, adding to operational costs.

10. Localization & Language Barriers – Expanding to international markets necessitates multilingual support and localized content, ensuring accessibility for a diverse global audience.

Chapter : 3

Hardware and Software Requirement

3.1 : Introduction

In this chapter, we outline the essential hardware and software requirements necessary to develop, run, and maintain the Journey-Genie e-commerce application. This includes the development environment, server specifications, and software dependencies used to build and deploy the system efficiently.

3.2 : System Environment

The system environment refers to the technical infrastructure in which the Journey-Genie application is developed and executed. This includes the operating system, backend server, and the architecture used for communication.

- **Operating System:**
 - The application is compatible with Windows environments.
 - During development, it was tested on Windows 10/11.
- **Backend Server:**
 - The project uses Spring Boot's embedded Apache Tomcat server to run the backend services.
 - No external Tomcat installation is required during local execution.
 - The server handles HTTP requests and responses through RESTful APIs.
- **API Architecture:**
 - Spring Boot RESTful APIs are used for communication between the frontend and backend.
 - These APIs handle operations like fetching products, managing orders, and processing payments.
- **Database:**

- MySQL is used as the relational database to store product details, customer information, order history, and transaction data.

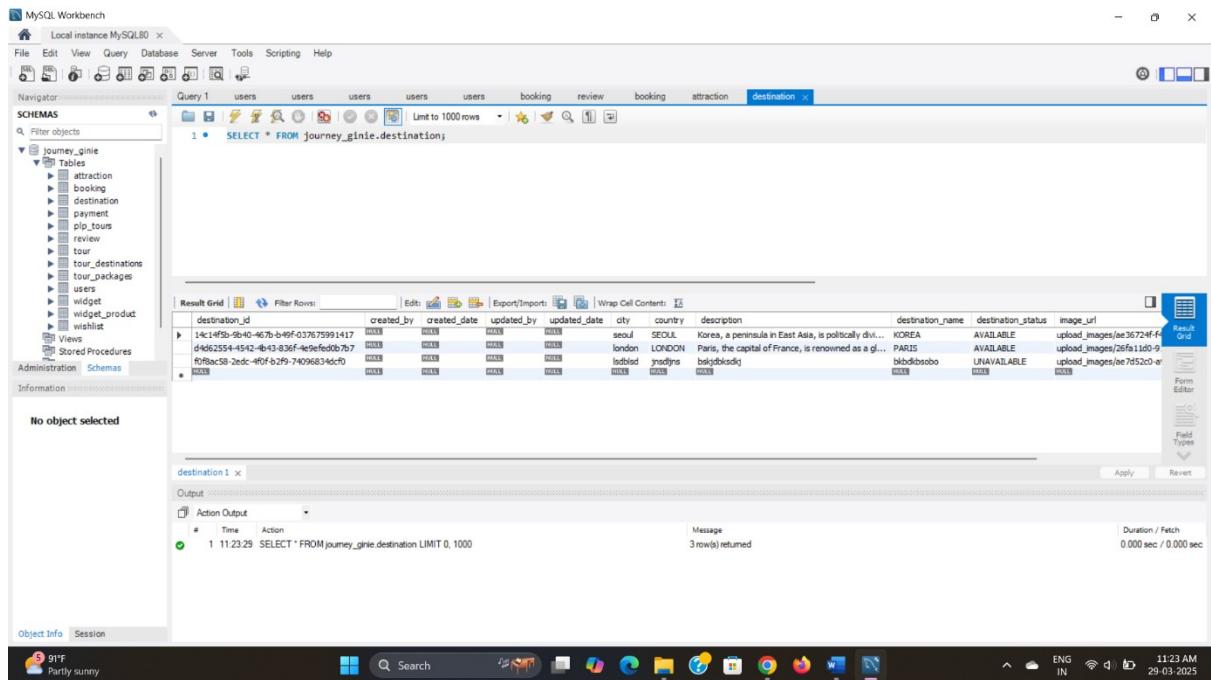
3.3 : Software Requirement

To build and run the Flavour Hub application, the following software components are required:

- Backend Technologies:
 - Java(jdk.17)
 - JavaScript
 - HTML,CSS
 - Spring Boot
 - Thymeleaf
 - Microsoft Visual Studio Code
 - Git and GitHub
 - Spring Boot Dependencies:
 1. Spring Web
 2. Spring JPA
 3. Spring Security
 4. Spring Dev Tool
 5. Validation
 6. Thymeleaf
 7. My Sql Driver
 8. Lombok

The screenshot shows the Spring Initializr interface. On the left, there are sections for 'Project' (Gradle or Maven selected), 'Language' (Java selected), and 'Spring Boot' (version 3.4.4 selected). Under 'Project Metadata', fields are filled with Group: 'rims-edu', Artifact: 'flavour-hub', Name: 'flavour-hub', and Description: 'Demo project for Spring Boot'. Below these are buttons for 'GENERATE' (CTRL + D) and 'EXPLORE' (CTRL + SPACE). A progress bar at the bottom indicates the generation process is 17% complete. On the right, a sidebar lists various dependencies: 'Spring Boot DevTools' (Developer Tools), 'Lombok' (Developer Tools), 'Thymeleaf' (Template Engines), 'Validation' (I/O), 'MySQL Driver' (SQL), 'Spring Security' (Security), and 'Spring Data JPA' (SQL). Each dependency has a brief description below it.

- **Database:**
 - MySQL Server : For managing and storing application data.
 - MySQL Workbench : For database administration and visualization.



(Journey-Ginie table data)

- **Development Tools :**
 - IDE: Visual Studio Code (VS Code) or IntelliJ IDEA.
 - Build Tool: Maven :To manage dependencies and build the project.
 - Version Control: Git : For tracking changes and collaborating.

3.4 : Hardware Requirement

The following hardware specifications are recommended for developing and running the Journey-Ginie application:

- For Development Environment:
 - Processor: Intel Core i5.
 - RAM: 8 GB for smoother multitasking.

- Storage: SSD with 256 GB for faster build times.
- Graphics: Integrated graphics (sufficient for development).
- Network: Stable internet connection for accessing external dependencies and APIs.

Chapter : 4

4.1 TOOLS AND TECHNOLOGIES

Journey Ginnie - Tools & Technologies Breakdown

Journey Ginnie is a travel and tour website designed with a combination of frontend, backend, and database technologies to ensure smooth user experience, efficient booking management, and secure transactions. Below is a detailed breakdown of the tools and technologies used:

4.1.1. Frontend Development (User Interface & Experience)

This part focuses on how users interact with the website, including trip selection, booking forms, animations, and interactive elements.

Languages & Frameworks:

- **HTML5** – The structural foundation of the website. It defines elements like buttons, forms, images, grids, and navigation bars.
- **CSS3** – Used for styling, themes, and layout design. Key features include:
 - **Blue & White Theme** – Ensures a professional, travel-friendly look.
 - **CSS Grid & Flexbox** – Used for responsive and adjustable grids.
 - **Ombre Light Blue Theme** – Applied for widgets and visual appeal.
- **JavaScript** – Enhances interactivity, animations, and form validations. Examples include:
 - Animations on login and registration pages.
 - Destination selection interactions.
 - Payment page dynamic updates.
- **CSS Frameworks:**
 - **CSS** – Helps in creating mobile-friendly and structured layouts quickly.

4.2 Backend Development (Java & MySQL)

The backend will be responsible for user authentication, trip bookings, admin management, and payments using Spring Boot and MySQL.

Technologies Used:

- Java (Spring Boot Framework) – Manages API endpoints and business logic.
- Spring Security – Ensures authentication and user role management.
- Spring Data JPA – Simplifies interaction with MySQL.
- Hibernate ORM – Manages database operations efficiently.
- MySQL – Stores user, booking, and payment data.
- JWT (JSON Web Token) – Secures login sessions.
- Swagger API Docs – Provides API documentation for developers.

4.3 Backend Functionalities

4.3.1. User Authentication & Registration

- Users can register, log in, and manage their profiles.
- Aadhaar card validation for new users.
- Passwords are hashed using BCrypt.

- Role-based authentication (User/Admin).

User Table (MySQL Schema)

sql

CopyEdit

```
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    email VARCHAR(100) UNIQUE,
    password VARCHAR(255),
    aadhaar_number VARCHAR(12) UNIQUE,
    role ENUM('USER', 'ADMIN')
);
```

User Authentication Flow

1. User Registers → Stores data in MySQL.
 2. Spring Security + JWT generates a secure token.
 3. User logs in → Token is verified for further API requests.
-

2. Destination Management (Admin Panel)

- Admins can add, update, or delete destinations.
- Destinations include name, description, country, city, image URL.

Destination Table (MySQL Schema)

sql

CopyEdit

```
CREATE TABLE destinations (
    id INT AUTO_INCREMENT PRIMARY KEY,
    destination_name VARCHAR(100),
    description TEXT,
    country VARCHAR(50),
    city VARCHAR(50),
    image_url VARCHAR(255)
);
```

Spring Boot API for Destination Management

java

CopyEdit

```
@RestController
```

```
@RequestMapping("/destinations")
```

```
public class DestinationController {
```

```
    @Autowired
```

```
    private DestinationService destinationService;
```

```
    @PostMapping("/add")
```

```
    public ResponseEntity<Destination>
```

```
    addDestination(@RequestBody Destination destination) {
```

```
        return
```

```
        ResponseEntity.ok(destinationService.addDestination(desti
nation));
```

```
    }
```

```

    @GetMapping("/all")
    public ResponseEntity<List<Destination>>
    getAllDestinations() {
        return
    ResponseEntity.ok(destinationService.getAllDestinations());
    }
}

```

3. Trip Booking System

- Users select destinations and confirm bookings.
- Multiple transport options (Flights, Trains, Buses, Cabs).
- Dynamic pricing calculation.

Booking Table (MySQL Schema)

sql

[CopyEdit](#)

```

CREATE TABLE bookings (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT,
    destination_id INT,
    transport_mode ENUM('Flight', 'Train', 'Bus', 'Cab'),
    total_price DECIMAL(10,2),
)

```

```
    booking_date TIMESTAMP DEFAULT
CURRENT_TIMESTAMP,
    FOREIGN KEY (user_id) REFERENCES users(id),
    FOREIGN KEY (destination_id) REFERENCES destinations(id)
);
```

Spring Boot API for Booking

```
java
```

```
CopyEdit
```

```
@RestController
```

```
@RequestMapping("/bookings")
```

```
public class BookingController {
```

```
    @Autowired
```

```
    private BookingService bookingService;
```

```
    @PostMapping("/create")
```

```
    public ResponseEntity<Booking>
createBooking(@RequestBody Booking booking) {
```

```
        return
```

```
        ResponseEntity.ok(bookingService.createBooking(booking));
```

```
    }
```

```
    @GetMapping("/user/{userId}")
```

```
    public ResponseEntity<List<Booking>>
getUserBookings(@PathVariable int userId) {
```

```
        return
```

```
        ResponseEntity.ok(bookingService.getUserBookings(userId));
```

```
    }  
}  
  
}
```

4. Payment Processing

- Uses Razorpay/Stripe/PayPal API integration.
- Transaction status (Success, Pending, Failed) stored in MySQL.

Payment Table (MySQL Schema)

sql

CopyEdit

```
CREATE TABLE payments (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    user_id INT,  
    booking_id INT,  
    amount DECIMAL(10,2),  
    payment_status ENUM('Success', 'Pending', 'Failed'),  
    transaction_id VARCHAR(100),  
    payment_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    FOREIGN KEY (user_id) REFERENCES users(id),  
    FOREIGN KEY (booking_id) REFERENCES bookings(id)  
);
```

Spring Boot API for Payment

java

CopyEdit

@RestController

@RequestMapping("/payments")

```
public class PaymentController {  
    @Autowired  
    private PaymentService paymentService;  
  
    @PostMapping("/process")  
    public ResponseEntity<Payment> processPayment(@RequestBody  
    Payment payment) {  
        return  
    ResponseEntity.ok(paymentService.processPayment(payment));  
    }  
}
```

Chapter :5

System Design

5.1: Introduction

The system design of Journey-Genie, a travelling e-commerce platform, defines the architectural blueprint of the application. It outlines how different components, such as the frontend, backend, and database, interact to deliver a seamless experience. The design process begins with a conceptual overview, highlighting the core actors and processes, followed by detailed data flow diagrams (DFD) to represent the movement of information between various modules. The ER diagram illustrates the relationships between key entities like users, products, orders, and payments. The logical design describes the data model, API endpoints, and business logic, while the physical design

focuses on the technologies used, including Spring Boot for backend development, MySQL for data storage, and HTML, CSS, and JavaScript for the frontend. This chapter serves as a technical foundation, ensuring the system is scalable, secure, and efficient in managing customer orders, product catalogs, and payments.

5.2: Conceptual Design

The conceptual design provides a high-level representation of the system, defining its main components and their interactions without technical implementation details.

Actors:

- Travelers (Customers):
 - Browse destinations and tour packages.
 - Book trips and manage reservations.
 - Make secure online payments.
- Admins:
 - Manage tour packages, destinations, and bookings.
 - Monitor payment transactions.
 - View reports and user activity.

Processes:

- Destination & Tour Package Management:
 - Admins add, update, and delete travel destinations and tour packages.
- Booking Management:
 - Travelers book packages via the website.

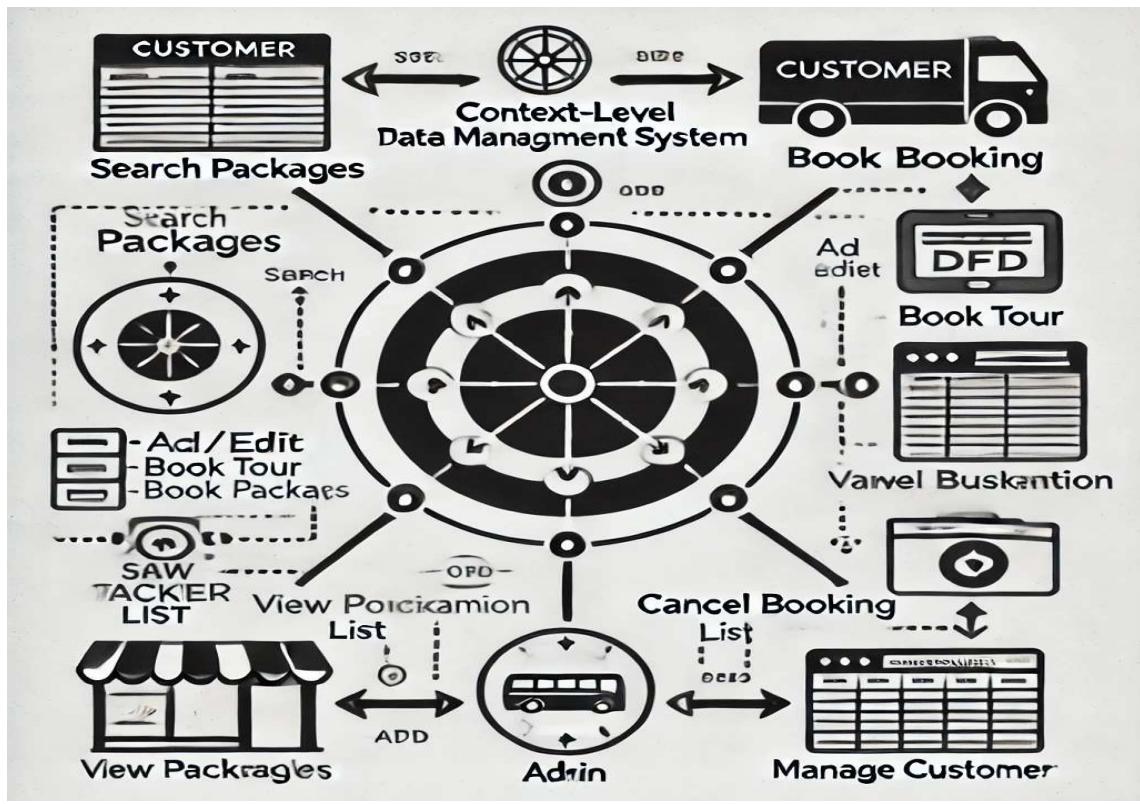
- Admins track and manage bookings from the dashboard.
- **Payment Processing:**
 - Travelers complete payments through integrated payment gateways.
 - Payment status is recorded and verified.

Data Flow:

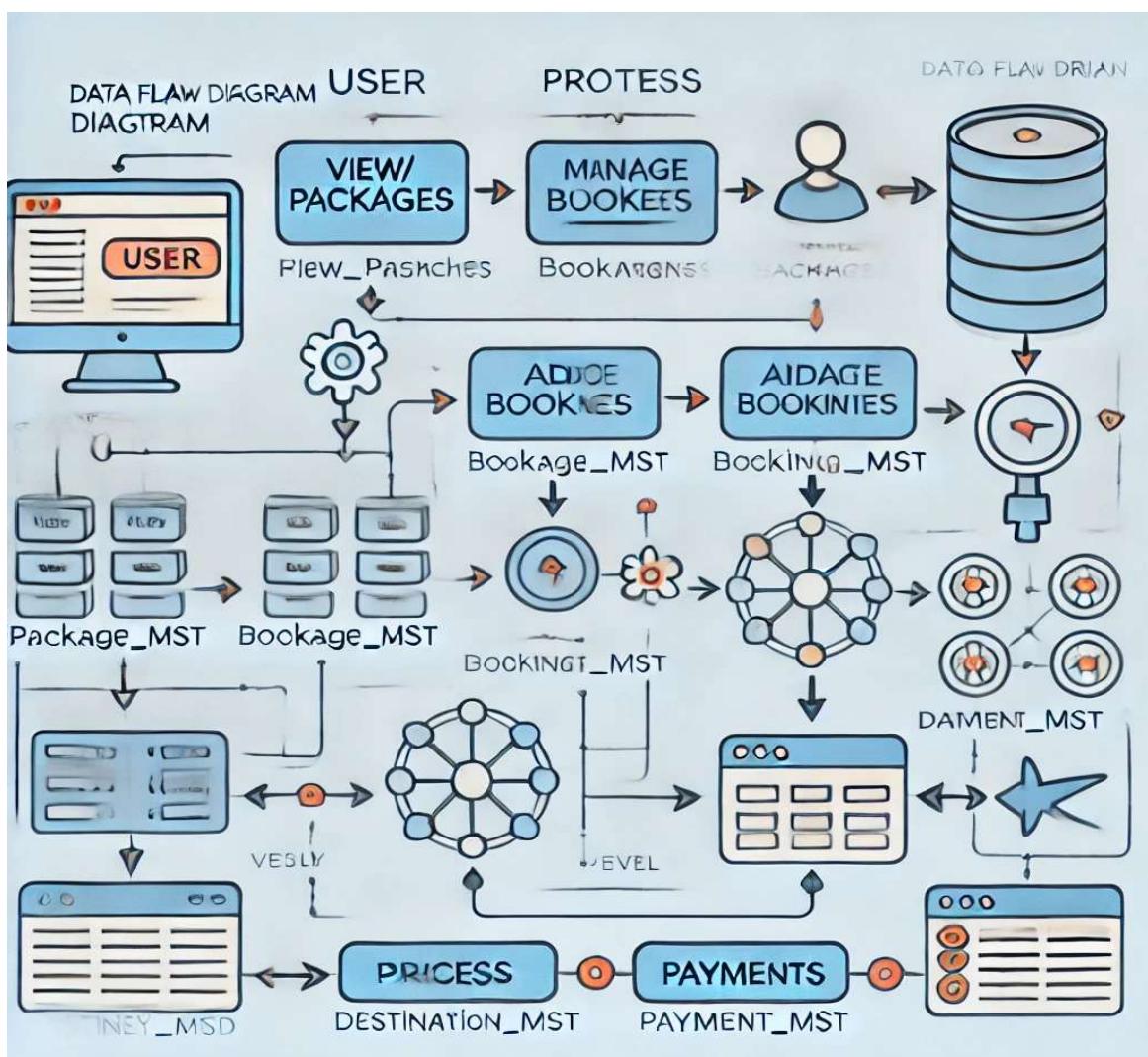
1. Users interact with the frontend (HTML, CSS, JavaScript).
2. Backend (Spring Boot) processes requests and communicates with the MySQL database.
3. Responses are sent back to the frontend to update booking status, package availability, and transaction details.

5.3 : Data Flow Diagram (DFD) of “JOURNEY-GINIE Website”

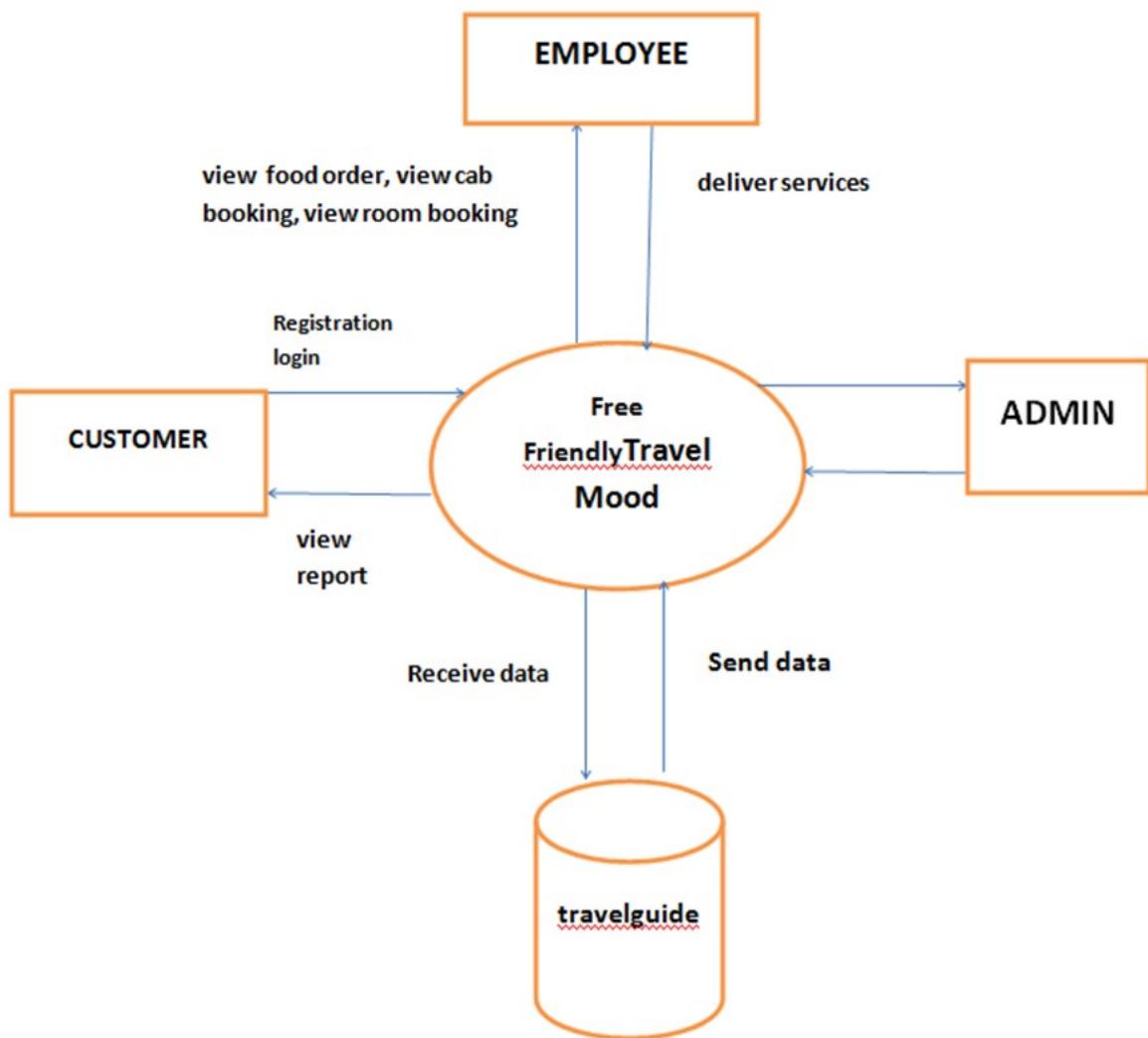
Context level DFD for Online travel booking system



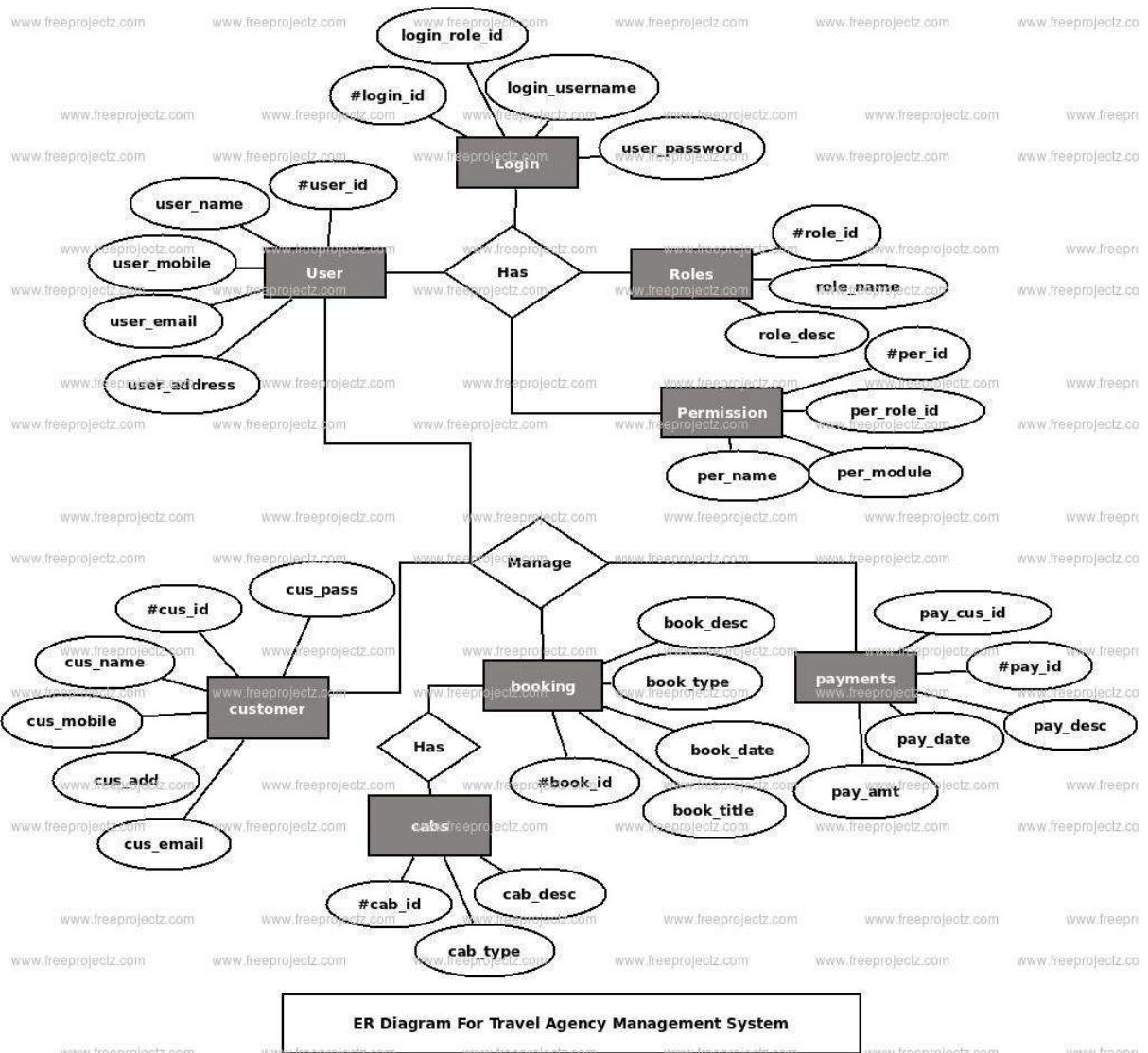
DFD for user



Admin side DFD for online travelling system



5.4 : E-R Diagram



5.5 : Logical Design

The logical design defines the data model and application logic without considering the physical implementation. It focuses on:

- **Database Schema:**
 - Tables: User, Destination, Tour, Booking, Packages, Payment.
 - Relationships: Primary and foreign key constraints.
- **Business Logic:**

- Authentication:
 - Spring Security handles user authentication.
- **Booking Management:**
 - Customers book destinations/tours, and the backend handles booking processing.
- **Package Management:**
 - Admins manage travel packages through CRUD operations.
- **API Endpoints:**
 - GET /destinations → Retrieve destination list.
 - POST /destination /add → Add destinations to the cart.
 - POST /booking/place → Place booking.
 - GET /admin/bookings → View all bookings.

5.6 : Physical Design

The physical design details the actual implementation and technology stack used in your project. It specifies how the system is built and deployed.

➤ Backend (Server-side)

- Language: Java (Spring Boot)
- Framework: Spring Boot
- Server: Embedded Apache Tomcat
- RESTful APIs: For data exchange

➤ Frontend (Client-side)

- Languages: HTML, CSS, JavaScript
- Framework: Bootstrap for styling
- Templates:
 - home.html, customer.html, admin.order.html
 - Static templates for UI rendering

➤ Database (Data Storage)

- Database: MySQL
- Schema:
 - Tables: User, Destination, Tour, Booking, Packages, Payment
- **JPA and Hibernate:**
 - Used for ORM (Object-Relational Mapping) to interact with the database.

➤ Hosting & Deployment:

- Local Deployment: Using Spring Boot's embedded Tomcat server for local testing.
- Deployment Tools:
 - Build tool: Maven (`pom.xml` configuration).
 - IDE: Visual Studio Code.

Chapter :6

System Analysis

6.1: Purpose

The purpose of the Journey-Genie project is to develop a robust admin dashboard for managing tour packages, destinations, and customer bookings. This platform enables travel administrators to efficiently organize travel-related data, monitor bookings, and streamline business operations.

By integrating secure authentication and a user-friendly interface, the system enhances operational efficiency, scalability, and customer satisfaction.

6.2: Project Scope

The scope of the Journey-Genie project defines its features, functionalities, and limitations:

User Roles:

- Admins: Manage tour packages, destinations, and customer bookings.
- Customers: Browse packages and book tours (handled via a separate frontend).

Core Features:

- Tour & Destination Management:
 - Admins can add, update, and delete travel packages and destinations.
- Booking Management:
 - Admins can view and manage customer bookings.
- Authentication & Security:
 - Spring Security ensures secure user authentication and access control.
- Dashboard & Data Visualization:
 - Admins get an overview of bookings, tours, and destinations via a structured dashboard.

Technological Scope:

- Backend: Spring Boot REST APIs with embedded Tomcat.

- Database: MySQL for storing tour, destination, and booking data.
- Frontend: HTML, CSS, JavaScript, and Font Awesome icons (No Bootstrap).

Limitations:

- The system does not support real-time booking updates yet.
 - Currently, it is an admin-only panel; customer interactions are handled separately.
-

6.3: Existing System

The existing system relies on manual processes for managing travel bookings, leading to inefficiencies:

Manual Booking Management:

- Customers book tours via calls or emails, leading to delays and mismanagement.
- Confirmation and tracking are not automated, increasing chances of errors.

Limited Tour & Destination Management:

- Admins store package details in spreadsheets or paper records.
- No centralized database, making it difficult to track available tours.

Payment & Security Issues:

- Manual payments may cause delays and discrepancies.
- Lack of role-based access control, increasing security risks.

6.4: Proposed System

The Journey-Genie Admin Dashboard introduces an automated and efficient tour booking management solution.

Key Features:

User-Friendly Admin Panel:

- Fixed left sidebar and top bar for easy navigation.
- Admins can manage packages, bookings, and destinations from a scrollable content area.

Efficient Booking Management:

- Bookings are processed and managed using RESTful APIs.

Secure & Centralized Data Handling:

- MySQL database ensures structured storage of tours, destinations, and bookings.

Authentication & Role-Based Access:

- Spring Security ensures only authorized admins can access the system.

6.5: System Description

The Journey-Genie platform consists of the following key components:

Frontend:

- Developed with HTML, CSS, and JavaScript for a clean and responsive admin panel.

- The admin dashboard provides a structured data table with an add button for managing records.

Backend:

- Built using Spring Boot, handling:
 - User authentication
 - Tour and destination management
 - Booking operations
- Uses RESTful APIs for seamless data exchange between the frontend and backend.

Database:

- MySQL stores:
 - Tour details
 - Destination information
 - Customer bookings securely

Server:

- The platform runs on Spring Boot's embedded Tomcat server, ensuring efficient request handling.

Security:

- Spring Security ensures authentication and authorization for admins only.

Chapter 7:

Implementation Issues

7.1 HTML

HTML forms the backbone of the Journey-Genie an online travel booking website. The implementation challenges faced while working with HTML include:

- **Semantic Structuring:** Ensuring proper use of semantic elements for better SEO(Search Engine Optimization) and accessibility.
- **Responsive Design Issues:** Difficulty in achieving a consistent layout across different devices.
- **Cross-Browser Compatibility:** Variations in how browsers render HTML elements, leading to inconsistencies.
- **Form Validation Constraints:** Client-side validation sometimes fails to work efficiently, requiring additional JavaScript support.

7.2 CSS

CSS is used to style the Journey-Genie website and provide a visually appealing interface. Some challenges encountered include:

- **Flexbox and Grid Layout Adjustments:** Achieving a fully responsive design using CSS Grid and Flexbox required multiple iterations.

- **Styling Consistency Across Browsers:** Differences in how browsers interpret CSS properties, requiring extensive testing.
- **Managing Large Stylesheets:** As the project grew, maintaining and organizing CSS files became challenging, leading to the adoption of SCSS for better modularity.
- **Animations and Transitions:** Ensuring smooth animations without affecting page performance was a challenge.

7.3 JavaScript

JavaScript is used to handle interactive components of the Journey-Genie website. The primary challenges faced include:

- **DOM Manipulation Performance:** Frequent updates to the DOM affected performance, requiring optimization techniques like event delegation.
- **Asynchronous Operations:** Managing asynchronous data fetching from APIs without blocking UI updates.
- **Cross-Browser Issues:** Some JavaScript features were not supported across all browsers, necessitating polyfills.
- **Security Concerns:** Preventing XSS (Cross-Site Scripting) and CSRF (Cross-Site Request Forgery) attacks.

7.4 Java

Java is used in the backend of the Flavour Hub system. Some challenges faced in its implementation include:

- **Performance Optimization:** Handling multiple requests simultaneously without slowing down the server.
- **Memory Management:** Ensuring efficient garbage collection and preventing memory leaks.
- **Integration with Databases:** Managing connections efficiently to prevent bottlenecks in data retrieval.
- **Error Handling:** Implementing a robust exception-handling mechanism to avoid system crashes.

7.5 Spring Boot

Spring Boot is used to develop the backend of the Flavour Hub website. The key implementation issues encountered include:

- **Configuration Management:** Properly setting up properties files for different environments (development, testing, production).
- **Dependency Injection Issues:** Managing dependency injections efficiently without causing bean conflicts.
- **Security Configuration:** Setting up authentication and authorization mechanisms to prevent unauthorized access.
- **REST API Performance:** Optimizing RESTful APIs to handle multiple concurrent user requests efficiently.

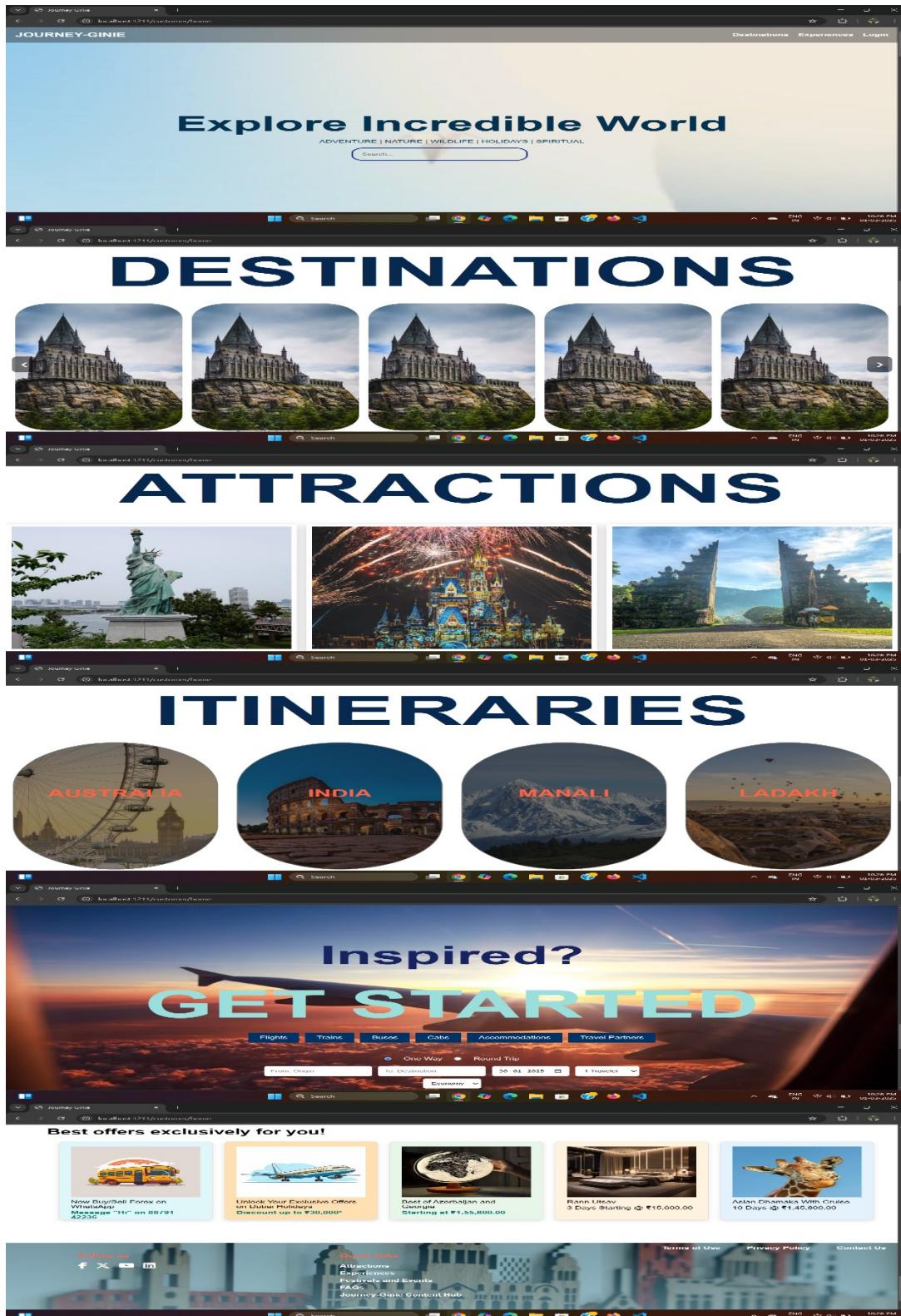
7.6 MySQL

MySQL is used as the database for Flavour Hub. The implementation issues faced include:

- **Database Schema Design:** Structuring tables efficiently to handle complex queries and relationships.
- **Query Optimization:** Ensuring SQL queries run efficiently to prevent slow page loads.
- **Data Consistency Issues:** Handling transactions properly to prevent data inconsistencies.
- **Scaling Challenges:** Ensuring the database can handle increased traffic without performance degradation.

Chapter 8.

User's Screen and Coding



```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Journey Ginie</title>
    <link rel="stylesheet" href="/style/home1.css">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.7.2/css/all.min.css"
        integrity="sha512-Evv84Mr4kqVGRNSgIGL/F/aIDqQb7xQ2vcrdIwxfjThSH8CSR7PBEakCr51Ck+w+/U6swU2Im1vVX0SVk9ABhg=="
        crossorigin="anonymous" referrerpolicy="no-referrer" />
</head>

<body>
    <header>
        <nav class="flex flex-between">
            <h1>JOURNEY-GINIE</h1>
            <ul class="flex">
                <li><a href="/customer/destination/destination?destination=6f435336-8f31-44d3-9d16-d343c0e708bc">Destinations</a>
                </li>
                <li><a href="/customer/tour/tour?tour=9f6987bc-35eb-499e-a8a7-255935daff10">Tours</a></li>
                <li><a href="/review">Reviews</a></li>
                <li><a href="/wishlist">Wishlist</a></li>
                <li><a href="/customer/login">Login</a></li>
            </ul>
        </nav>
    </header>

```

```
</ul>

</nav>

<div class="video">
  <video src="/video/travel.mp4" autoplay muted loop></video>
</div>

<div class="overlay">
  <div>
    <h1>Explore Incredible World</h1>
    <p>ADVENTURE | NATURE | WILDLIFE | HOLIDAYS | SPIRITUAL</p>
    <form action="/destination/search">
      <div class="search-bar">
        <!-- <div class="header"> -->
        <input name="search" placeholder="Search..." class="search-input">
      </div>
    </form>
  </div>
</div>

<div class="categories">
  <div class="category">
    
    <p>Bicycle</p>
  </div>
  <div class="category">
    
    <p>Beach</p>
  </div>
  <div class="category">
    
    <p>Adventure</p>
  </div>
  <div class="category">
    
```

```
<p>Safari</p>
</div>

<div class="category">

<p>River Cruise</p>
</div>

<div class="category">

<p>Hill Stations</p>
</div>

<div class="category">

<p>Trekking</p>
</div>

<div class="category">

<p>River Cruise</p>
</div>

<div class="category">

<p>Hill Stations</p>
</div>

<div class="category">

<p>Hiking</p>
</div>

<div class="category">

<p>River Cruise</p>
</div>

<div class="category">

<p>Hill Stations</p>
```

```

        </div>
    </div>

</header>

<section class="destinations">
    <h2 class="title">DESTINATIONS</h2>
    <div class="destination-slider">
        <div class="destination-grid flex pl-1">
            <div class="destination" data-th-each="destination : ${destinations}">
                <a data-th-href="/customer/destination?destination=${destination.destinationId}">
                    <div class="destination-image">
                        
                        <h4 class="img-name"></h4>
                    </div>
                </a>
                <div class="img-name" data-th-text="${destination.destinationName}"></div>
            </div>
        </div>
        <button class="slider-btn prev" onclick="moveSlide(-1)">←</button>
        <button class="slider-btn next" onclick="moveSlide(1)">→</button>
        <!-- <div class="destination">
            
            <h4 class="img-name">AMERICA</h4>
        </div>
        <div class="destination">
            
            <h4 class="img-name">PARIS</h4>
        </div>
        <div class="destination">
            
            <h4 class="img-name">KOREA</h4>
        </div>-->
    </div>

```

```

</div>
</section>
<script src="/script/home.js"></script>

<section class="attraction">
  <h2 class="title">ATTRACTIOnS</h2>
  <div class="image-wrapper flex">
    <div class="box" data-th-each="attraction : ${attractions}">
      <a data-th-href="/customer/attraction?attraction=${attraction.attractionId}">
        
      </div>
    </a>
    <!-- <div class="box"></div>
    <div class="box"></div> -->
  </div>
  <div class="button-container">
    <button class="discover-btn">Discover More</button>
  </div>
</section>

<section class="itineraries">
  <h2 class="title">POPULAR TOURS</h2>
  <div class="itinerary-grid flex flex-center gap-2">
    <div class="itinerary" data-th-each="tour : ${tours}">
      <a data-th-href="/customer/tour?tour=${tour.id}">
        <div class="itinerary">
          
          <!--  -->
        </div>
      </a>
      <div class="intro">

```

```
<div>
    <h3 data-th-text="${tour.title}"></h3>
    <p>20+ wildlife packages</p>
</div>
</div>

<!-- <div class="itinerary">
    
    <div class="intro">
        <div>
            <h3>ADVENTURE</h3>
            <p>20+ adventure packages</p>
        </div>
    </div>
</div>

<div class="itinerary">
    
    <div class="intro">
        <div>
            <h3>HONEYMOON</h3>
            <p>18+ honeymoon packages</p>
        </div>
    </div>
</div>

<div class="itinerary">
    
    <div class="intro">
        <div>
            <h3>FREEDOM</h3>
            <p>25+ freedom packages</p>
        </div>
    </div>
</div>
```

```
</div>
</div> -->
</section>

<div class="about-container">
<h1>Why Journey-Ginie ?</h1>
<div class="about-grid">
<div class="about-box box1">
<h2> Explore the World</h2>
<p>Discover breathtaking destinations with our hand-picked travel packages, curated just for you!</p>
</div>
<div class="about-box box2">
<h2> Best Travel Deals</h2>
<p>We bring you exclusive discounts on flights, hotels, and holiday packages to make your trip budget-friendly.</p>
</p>
</div>
<div class="about-box box3">
<h2> Hassle-Free Bookings</h2>
<p>Enjoy seamless online reservations with our easy-to-use booking system, making your travel stress-free.</p>
</div>
<div class="about-box box4">
<h2> 24/7 Customer Support</h2>
<p>Our travel experts are available anytime, anywhere, ensuring a smooth journey from start to finish.</p>
</div>
<div class="about-box box5">
<h2> Personalized Itineraries</h2>
<p>Get tailor-made travel plans designed to match your style, interests, and budget.</p>
</div>
<div class="about-box box6">
<h2> Cruise & Luxury Travel</h2>
```

<p>Indulge in premium travel experiences, including luxury cruises and VIP holiday packages.</p>

</div>

</div>

</div>

<th:block th:each="widget : \${widgets}" th:switch="\${widget.widgetName}"></th:block>

<h1>Best offers exclusively for you!</h1>

<div class="container">

<div class="grid">

<div class="offer-card whatsapp">

<p>Now Buy/Sell Forex on WhatsApp</p>

<p class="message">Message "Hi" on 88791 42236</p>

</div>

<div class="offer-card dubai">

<p>Unlock Your Exclusive Offers on Dubai Holidays</p>

<p class="discount">Discount up to ₹30,000*</p>

</div>

<div class="offer-card georgia">

<p>Best of Azerbaijan and Georgia</p>

<p class="offer">Starting at ₹1,55,600.00</p>

</div>

<div class="offer-card rann-utsav">

<p>Rann Utsav</p>

<p>3 Days Starting @ ₹15,600.00</p>

</div>

<div class="offer-card cruise">

<p>Asian Dhamaka With Cruise</p>

<p>10 Days @ ₹1,45,800.00</p>

```

        </div>
        </div>
        </div>
        </div>

<section class="hope"
    style="background-image: url('/img/aero2.jpg'); background-size: cover; background-position: center;">
    <h1>Inspired?</h1>
    <h2>GET STARTED</h2>
    <nav class="tabs">
        <a href="/customer/inspiration">
            <button class="tab active">Flights</button>
            <button class="tab">Trains</button>
            <button class="tab">Buses</button>
            <button class="tab">Cabs</button>
            <button class="tab">Accommodations</button>
            <button class="tab">Travel Partners</button>
        </a>
    </nav>
    <!-- </section> -->

<div class="booking-form">
    <div class="trip-type">
        <input type="radio" id="one-way" name="trip" checked />
        <label for="one-way">One Way</label>

        <input type="radio" id="round-trip" name="trip" />
        <label for="round-trip">Round Trip</label>
    </div>

    <div class="inputs">
        <input type="text" placeholder="From: Origin" aria-label="Departure location" />
        <input type="text" placeholder="To: Destination" aria-label="Arrival location" />
    </div>

```

```

<input type="date" value="2025-01-30" />

<select aria-label="Number of travelers">
  <option value="1">1 Traveler</option>
  <option value="2">2 Travelers</option>
  <option value="3">3 Travelers</option>
</select>

<select aria-label="Travel class">
  <option value="economy">Economy</option>
  <option value="business">Business</option>
</select>

</div>

<div class="fare-type">
  <label><input type="radio" name="fare" checked /> Regular Fares</label>
  <label><input type="radio" name="fare" /> Armed Forces Fares</label>
  <label><input type="radio" name="fare" /> Student Fares</label>
  <label><input type="radio" name="fare" /> Senior Citizen Fares</label>
  <label><input type="radio" name="fare" /> Doctors & Nurses Fares</label>
</div>

<div class="btn-container">
  <a href="/customer/booking">
    <button class="booking-btn" onclick="alert('Book Now')">BOOK YOUR TRIP</button>
  </a>
</div>

</section>

<section class="reviews-section">
  <h2>🌟 What Our Travelers Say</h2>

```

```
<div class="reviews-container">
  <div class="review-box">
    <h3>Emily Johnson</h3>
    <div class="rating">★ ★ ★ ★</div>
    <p>"Journey-Ginie made my dream vacation a reality! Everything was seamless and unforgettable."</p>
  </div>

  <div class="review-box">
    <h3>Michael Smith</h3>
    <div class="rating">★ ★ ★ ★</div>
    <p>"Great service and excellent tour guides! Highly recommended for hassle-free trips."</p>
  </div>

  <div class="review-box">
    <h3>Sophia Martinez</h3>
    <div class="rating">★ ★ ★ ★ ★</div>
    <p>"Loved the adventure package! The team ensured every moment was enjoyable and safe."</p>
  </div>

  <div class="review-box">
    <h3>Daniel Wilson</h3>
    <div class="rating">★ ★ ★ ★ ★</div>
    <p>"Professional, friendly, and well-organized. Journey-Ginie exceeded my expectations!"</p>
  </div>

</div>
<script src="/script/review.js"></script>

<!-- Add a Review Section --&gt;
&lt;div class="add-review"&gt;
  &lt;h3&gt;○ Share Your Experience&lt;/h3&gt;</pre>
```

```

<input type="text" placeholder="Your Name" id="reviewer-name">
<textarea placeholder="Write your review..." id="review-text"></textarea>
<button onclick="submitReview()">Submit Review</button>
<button class="delete-review" onclick="deleteReview(this)">X Delete</button>
</div>
</section>

<section class="ftr"
  style="background-image: url('/img/ft.jpg'); background-size: cover; background-position: center;">
  <footer>
    <div class="footer-container">
      <div class="footer-section">
        <h3>Follow us</h3>
        <div class="social-icons">
          <a href="#"><i class="fab fa-facebook-f"></i></a>
          <a href="#"><i class="fab fa-x-twitter"></i></a>
          <a href="#"><i class="fab fa-youtube"></i></a>
          <a href="#"><i class="fab fa-linkedin"></i></a>
        </div>
      </div>
      <div class="footer-section">
        <h3><b>Quick links</b></h3>
        <ul>
          <li><a href="#"><b>Attractions</b></a></li>
          <li><a href="#"><b>Experiences</b></a></li>
          <li><a href="#"><b>Festivals and Events</b></a></li>
          <li><a href="#"><b>FAQs</b></a></li>
          <li><a href="#"><b>Journey-Ginie Content Hub</b></a></li>
        </ul>
      </div>
      <div class="footer-section">
        <h3><b>Newsletter</b></h3>
        <p><b>Sign up for exciting news, events, and great travel ideas.</b></p>
      </div>
    </div>
  </section>

```

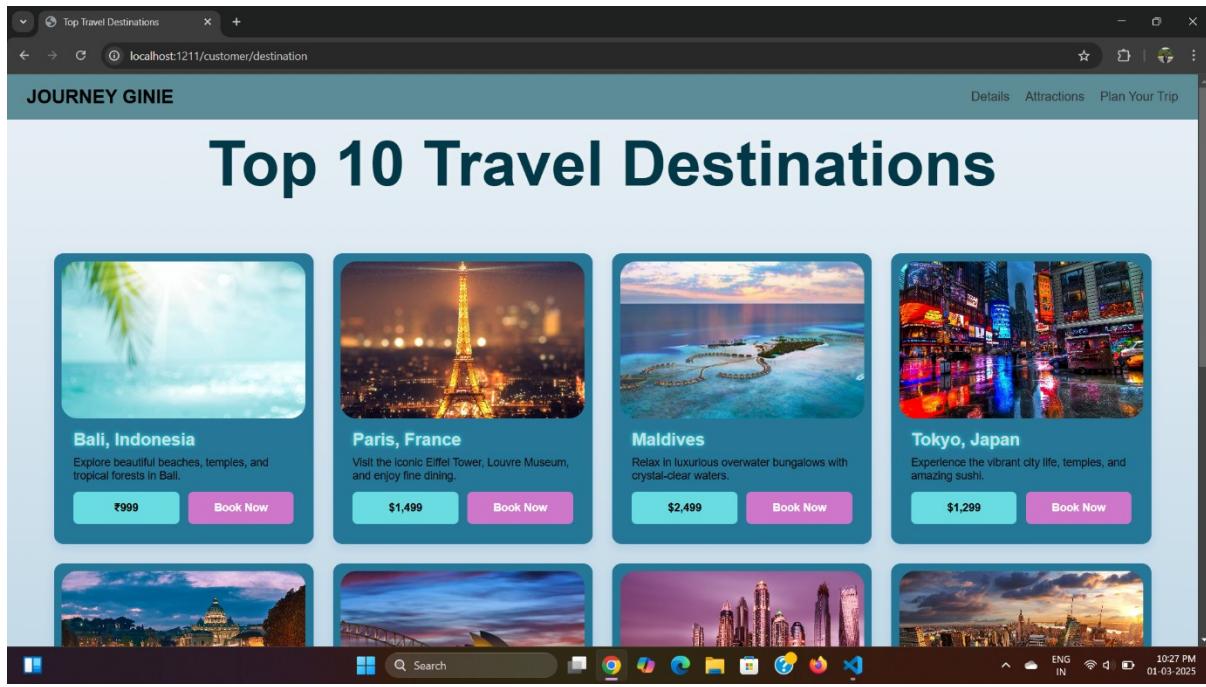
```
<button class="newsletter">Subscribe now</button>
</div>

<!-- <div class="footer-section qr-section">
<h3><b>Scan to chat</b></h3>
<b></b>
</div> -->
</div>

<div class="footer-links">
<a href="#"><b>Terms of Use</b></a>
<a href="#"><b>Privacy Policy</b></a>
<a href="#"><b>Contact Us</b></a>
</div>
</footer>
</section>
</body>

</html>
```

Product listing page



```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Top Travel Destinations</title>
    <link rel="stylesheet" href="/style/destination.css">
</head>

<body>
    <nav class="navbar">
        <div class="logo">JOURNEY GINIE</div>
        <ul class="nav-links">
            <li><a href="/customer/pdp">Details</a></li>
            <li><a href="#">Attractions</a></li>
            <li><a href="#">Plan Your Trip</a></li>
        </ul>
    </nav>

```

```

<h1 class="title">Top 10 Travel Destinations</h1>
<div class="destinations-container">

<div class="destination-card" data-th-each="destination : ${destinations}">
    
    <div class="destination-info">
        <h2 data-th-text="${destination.destinationName}"></h2>
        <article class="destination">
            <a data-th-
                href="|/customer/destination/pdp?destination=${destination.destinationId} | "
                style="all: unset;">
                <p data-th-text="${destination.description}"></p>
                <!-- <h4 data-th-text="${destination.destinationName}"></h4> -->
                <!-- <p data-th-text="| &#8377; ${destination.price} | "></p> -->
                <div class="buttons">
                    <p data-th-text="| &#8377; ${destination.price} | "></p>
                    <!-- <p class="place-price">&#8377;999</p> -->
                    <button class="book-btn">View Details</button>
                </div>
            </a>
        </article>
    </div>
</div>

<!-- <div class="destination-card">
    
    <div class="destination-info">
        <h2>Paris, France</h2>
        <p>Visit the iconic Eiffel Tower, Louvre Museum, and enjoy fine dining.</p>
        <div class="buttons">
            <button class="price-btn">$1,499</button>
            <button class="book-btn">Book Now</button>
        </div>
</div>

```

```
</div>
</div>

<div class="destination-card">
  
  <div class="destination-info">
    <h2>Maldives</h2>
    <p>Relax in luxurious overwater bungalows with crystal-clear waters.</p>
    <div class="buttons">
      <button class="price-btn">$2,499</button>
      <button class="book-btn">Book Now</button>
    </div>
  </div>
</div>

<div class="destination-card">
  
  <div class="destination-info">
    <h2>Tokyo, Japan</h2>
    <p>Experience the vibrant city life, temples, and amazing sushi.</p>
    <div class="buttons">
      <button class="price-btn">$1,299</button>
      <button class="book-btn">Book Now</button>
    </div>
  </div>
</div>

<div class="destination-card">
  
  <div class="destination-info">
    <h2>Rome, Italy</h2>
    <p>Walk through ancient history with the Colosseum and Vatican City.</p>
  </div>
</div>
```

```

<div class="buttons">
    <button class="price-btn">$1,199</button>
    <button class="book-btn">Book Now</button>
</div>
</div>

<div class="destination-card">
    
    <div class="destination-info">
        <h2>Sydney, Australia</h2>
        <p>Enjoy the Opera House, Harbour Bridge, and Bondi Beach.</p>
        <div class="buttons">
            <button class="price-btn">$1,599</button>
            <button class="book-btn">Book Now</button>
        </div>
    </div>
</div>

<div class="destination-card">
    
    <div class="destination-info">
        <h2>Dubai, UAE</h2>
        <p>Visit the luxurious city with skyscrapers and desert adventures.</p>
        <div class="buttons">
            <button class="price-btn">$1,799</button>
            <button class="book-btn">Book Now</button>
        </div>
    </div>

```

```

</div>

<div class="destination-card">
  
  <div class="destination-info">
    <h2>New York, USA</h2>
    <p>Experience Times Square, Broadway, and Central Park.</p>
    <div class="buttons">
      <button class="price-btn">$1,399</button>
      <button class="book-btn">Book Now</button>
    </div>
  </div>
</div>

<div class="destination-card">
  
  <div class="destination-info">
    <h2>Cape Town, South Africa</h2>
    <p>Explore Table Mountain, beaches, and rich cultural heritage.</p>
    <div class="buttons">
      <button class="price-btn">$1,299</button>
      <button class="book-btn">Book Now</button>
    </div>
  </div>
</div>

<div class="destination-card">
  
  <div class="destination-info">
    <h2>Santorini, Greece</h2>

```

```

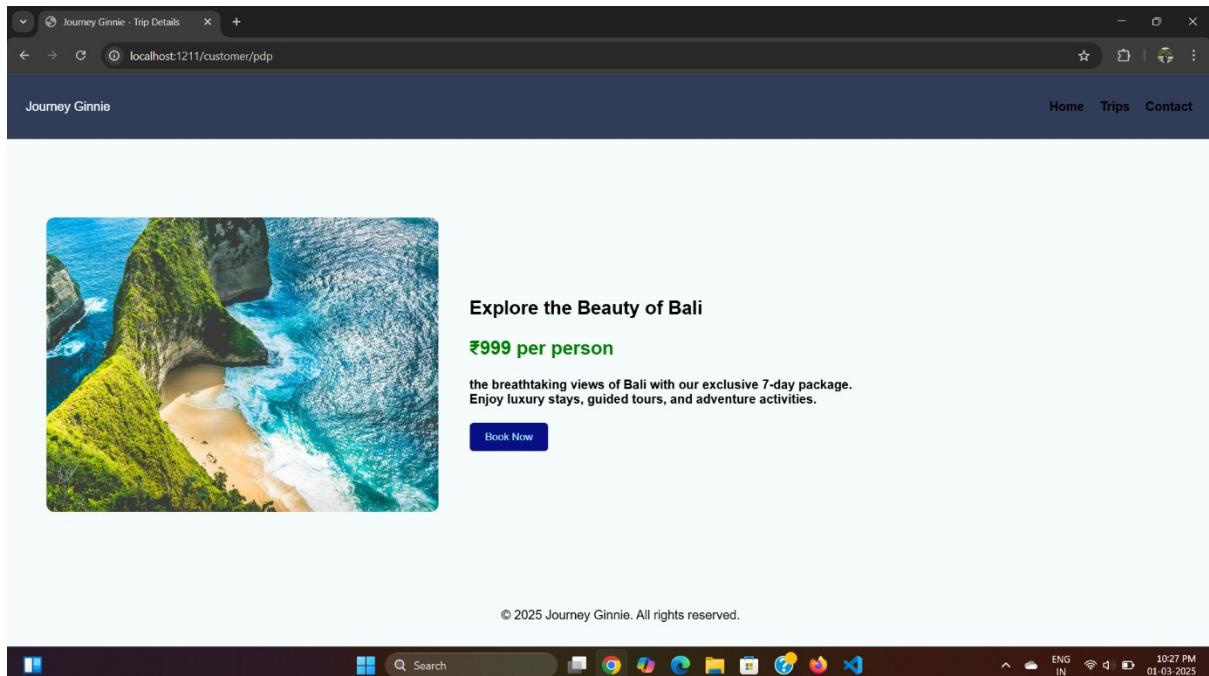
<p>Marvel at the white-washed buildings and stunning sunsets.</p>
<div class="buttons">
    <button class="price-btn">$1,699</button>
    <button class="book-btn">Book Now</button>
</div>
</div> -->
</div>

</body>

</html>

```

Product Description page



```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Journey Ginnie - Trip Details</title>
    <link rel="stylesheet" href="/style/pdp.css">
</head>
<body>
    <header>
        <nav class="navbar">
            <div class="logo">Journey Ginnie</div>
            <ul class="nav-links">
                <li><a href="/customer/home">Home</a></li>
                <li><a href="#trips">Trips</a></li>
                <li><a href="#contact">Contact</a></li>
            </ul>
        </nav>
    </header>
    <div data-th-if="${destination != null}">
        <section class="trip-details">
            <div class="trip-image">
                
            </div>
            <div class="trip-info">
                <h1 data-th-text="${destination.description}"></h1>
                <h4 data-th-text="${destination.destinationName}"></h4>
                <p class="price" data-th-text="| &#8377; ${destination.price} per person | "></p>
                <a href="/customer/booking">
                    <button class="book-now">Book Now</button>
                </a>
            </div>
        </section>
    </div>
</body>

```

```

</div>

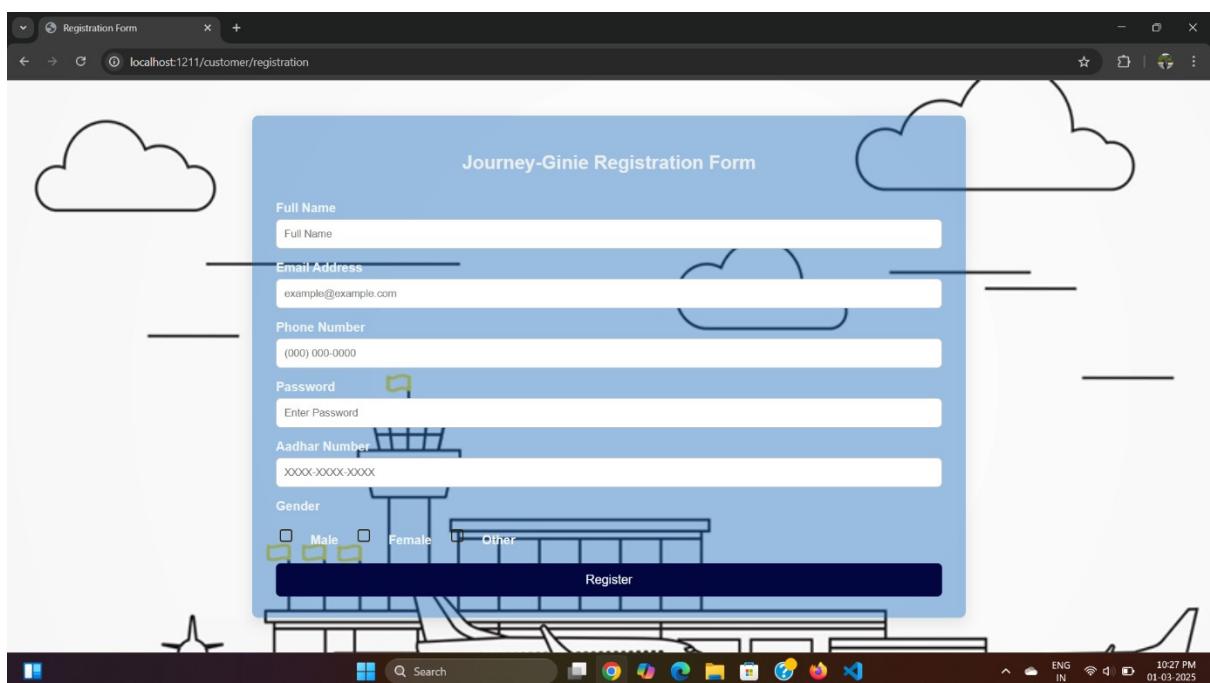
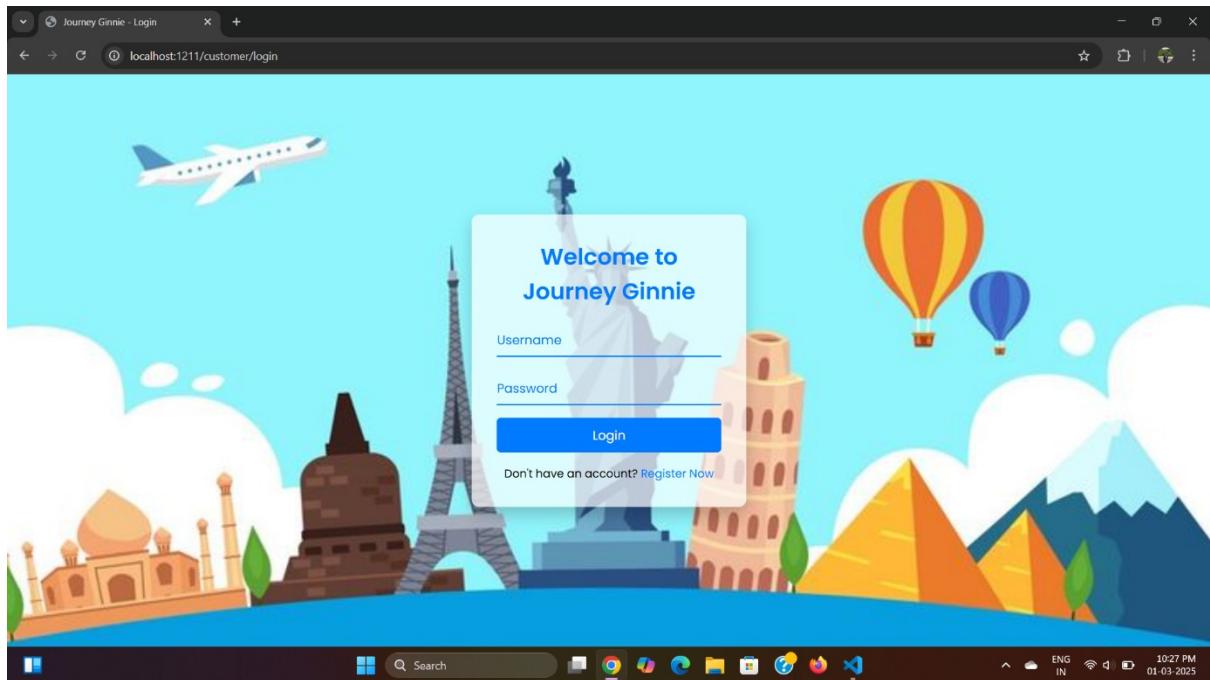
</section>
</div>
<!-- <div data-th-if="${tourPackage != null}">
<section class="trip-details">
<div class="trip-image">

</div>
<div class="trip-info">
<h1 data-th-text="${tourPackage.packageTitle}"></h1> -->
<!-- <p class="price" data-th-text=" | &#8377; ${destination_place.price} per
person | "></p> -->
<!-- <h4 data-th-text="${tourPackage.packageDescription}"></h4>
<a class="book-now">Book Now</a>
</div>
</section>
</div> -->
<footer class="footer">
<p>&copy; 2025 Journey Ginnie. All rights reserved.</p>
</footer>

</body>
</html>

```

User Login



Login code

```
<div id="loginForm">  
    <h2>Login to Your Account</h2>  
    <form class="form" method="post" action="/customer/login">  
        <input type="text" name="username" id="loginEmail" placeholder="Enter your email" required>  
        <input type="password" name="password" id="loginPassword" placeholder="Enter your password" required>  
        <input type="hidden" name="_csrf" data-th-value="${_csrf.token}">
```

```

<button type="submit">Sign In</button>
</form>
<p class="toggle-text">Don't have an account? <a href="#" onclick="toggleForms()">Sign Up</a></p>
</div>

Signup

<div id="signupForm" class="hidden">
    <h2>Create an Account</h2>
    <form class="form" method="post" action="/customer/sign-up">
        <div class="name">
            <input type="text" id="signupName" name="userName" placeholder="Full Name" required>
            <input type="email" id="signupEmail" name="userEmail" placeholder="Email" required>
            <input type="tel" id="userPhone" name="userPhone" placeholder="Phone Number" required>
            <input type="password" id="signupPassword" name="userPassword" placeholder="Password" required>
            <input type="text" id="street" name="userAddress" placeholder="Street Address" required>
            <input type="text" id="city" placeholder="City" required>
            <input type="text" id="state" placeholder="State" required>
            <input type="text" id="zip" placeholder="Pin Code" required>
        </div>
        <input type="hidden" name="_csrf" data-th-value="${_csrf.token}">
        <button type="submit">Sign Up</button> </form>
        <p class="toggle-text">Already have an account? <a href="#" onclick="toggleForms()">Sign In</a></p>
    </div>
</div>

```

Destination Controller Code

```
package edu.rims.Journey_Genie.controller;
```

```
import java.io.FileInputStream;
```

```

import java.io.FileOutputStream;
import java.io.IOException;
import java.util.List;
// import java.util.Locale.Destination;
import java.util.UUID;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.ResponseBody;
import org.springframework.web.multipart.MultipartFile;

import edu.rims.Journey_Genie.entity.Destination;
import edu.rims.Journey_Genie.repository.DestinationRepository;

@Controller
public class DestinationController {
    @Autowired
    private DestinationRepository destinationRepository;

    @GetMapping("/customer/destination")
    String getDestinationByDestinationId(Model model) {
        model.addAttribute("destinations", destinationRepository.findAll());
        return "customer/destination";
    }

    @PostMapping("/admin/destinations")
    public String destinationAdd(@ModelAttribute Destination destination,

```

```

    @RequestParam("destinationImageFile") MultipartFile file, @RequestParam String
destImage) throws IOException {

    String destinationId = destination.getDestinationId();

    destination.setDestinationId(destinationId == null || destinationId.isEmpty() ? null :
destinationId);

    if (!file.isEmpty()) {

        String originalFilename = file.getOriginalFilename();

        String extName = originalFilename.substring(originalFilename.lastIndexOf("."));

        String fileName = "upload_images/" + UUID.randomUUID().toString() + extName;

        FileOutputStream fileOutputStream = new FileOutputStream(fileName);

        fileOutputStream.write(file.getBytes());

        fileOutputStream.close();

        destination.setImageUrl(fileName);

    }else if(!destImage.isEmpty()){

        destination.setImageUrl(destImage);

    }

    destinationRepository.save(destination);

    return "redirect:/admin/destinations";

}

@GetMapping("/destination/images/{destinationId}")

@ResponseBody

public byte[] getDestinationImage(@PathVariable String destinationId) throws IOException {

    Destination destination = destinationRepository.findById(destinationId).orElseThrow();

    String imageName = destination.getImageUrl();

    if (imageName == null || imageName.startsWith("http")) {

        return null;

    }

    FileInputStream fileInputStream = new FileInputStream(imageName);

    return fileInputStream.readAllBytes();

}

```

```

@GetMapping("/destination/search")
String searchDestination(@RequestParam("search") String destinationName, Model model) {
    List<Destination> destinations =
destinationRepository.findByDestinationNameContainingIgnoreCase(destinationName);
    model.addAttribute("destinations", destinations);
    return "customer/destination";
}

@GetMapping("/customer/destination/pdp")
String getDestinationBydestinationId(@RequestParam("destination") String destinationId,
Model model) {
    Destination destination = destinationRepository.findById(destinationId).orElseThrow();
    model.addAttribute("destination", destination);
    return "customer/pdp";
}

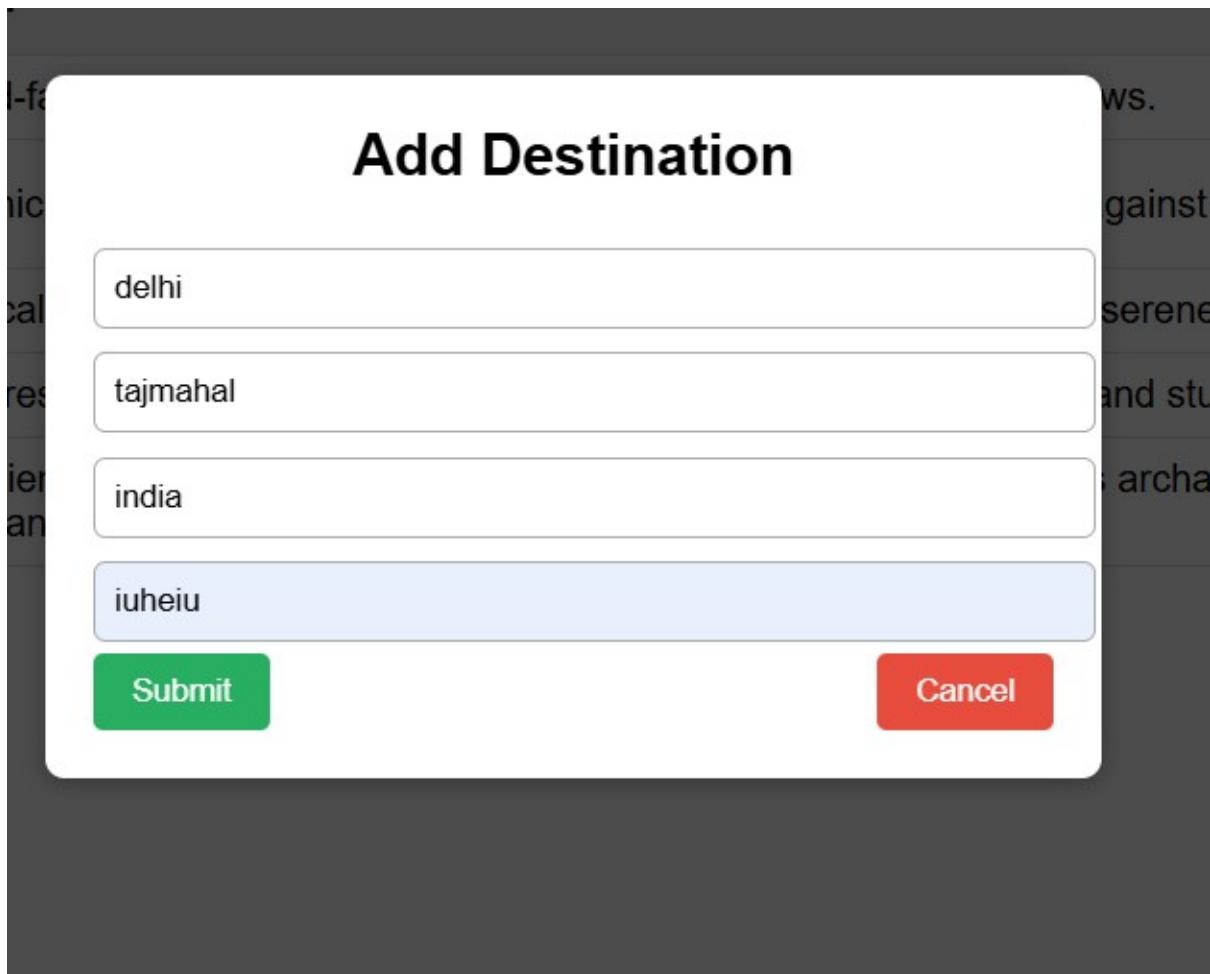
```

Admin Screen and Coding



The screenshot shows the Journey-Genie admin dashboard. On the left, a dark sidebar menu lists: Dashboard, Bookings, Customers, Destinations (which is the active tab), Tours, Settings, and Logout. The main content area has a header with a search bar and an 'Add Destination' button. Below is a table listing two destinations:

| Destination Name | Description | Country | City | Image URL |
|------------------|--|---------|-------|---|
| Paris | The city of lights, known for the Eiffel Tower. | France | Paris | https://example.com/paris.jpg |
| Tokyo | A bustling metropolis blending tradition and technology. | Japan | Tokyo | https://example.com/tokyo.jpg |



Admin Dashboard

```
<div class="sales-boxes">  
    <div class="recent-sale-box">  
        <div class="title">Recent Sales</div>  
        <div class="sales-details">  
            <table>  
                <thead>  
                    <tr> <th>Customer</th>  
                        <th>Address</th>  
                        <th>Destinations </th>  
                        <th>Country </th>  
                        <th>Status</th>  
                        <th>Action </th>  
                </tr>  
            </thead>
```

```

        </thead>
        <tbody>
            <tr data-th-each="order : ${recentBookings}">
                <td data-th-text="${order.user.userName}"></td>
                <td data-th-text="${order.user.userAddress}"></td>
                <td>
                    <ul>
                        <li data-th-each="item : ${booking.bookingItems}">
                            <span data-th-text="${item.destination.destinationName}"></span>
                        </li>
                    </ul>
                </td>
                <td>
                    <ul>
                        <li data-th-each="item : ${booking.bookingItems}">
                            <span data-th-text="${destination.bookingItemQuantity}"></span>
                        </li>
                    </ul>
                </td>
                <td>
                    <p class="status"
                        data-th-class="status ' +
                        ${booking.bookingStatus.name().toLowerCase()}">
                    </p>
                    <span data-th-text="${booking.bookingStatus}"></span>
                </td>
                <td><strong data-th-text="₹' + ${booking.totalPrice}"></strong></td>
            </tr>
        </tbody>
    </table>
</div>
<div class="button">
    <a href="/admin/booking">See all</a>
</div>

```

Chapter 9: Testing

9.1 Software Testing Life Cycle (STLC)

The Software Testing Life Cycle (STLC) is a structured process followed during the testing phase of the Journey-Genie travel website. It includes the following stages:

- **Requirement Analysis:** Understanding functional and non-functional requirements to determine test objectives.
- **Test Planning:** Creating a strategy for testing, including scope, resources, schedule, and risks.
- **Test Case Development:** Designing and documenting test cases based on requirements.
- **Environment Setup:** Preparing the necessary hardware, software, and configurations for testing.
- **Test Execution:** Running test cases and identifying defects.
- **Test Closure:** Evaluating the testing process, documenting learnings, and finalizing reports.

9.2 Test Plan Creation

A test plan is developed to define the testing strategy for Journey-Genie. Key elements include:

- **Test Objectives:** Ensuring the website is fully functional, secure, and user-friendly.

- **Scope of Testing:** Covering functional, performance, security, and usability testing.
- **Test Deliverables:** Test cases, reports, defect logs, and final test summary.
- **Testing Tools:** Identifying tools like Selenium, JUnit, or Postman for automation and API testing.
- **Roles and Responsibilities:** Assigning team members to execute test cases and report issues.
- **Risk Analysis:** Identifying potential risks and mitigation strategies.

9.3 Environment Setup

The testing environment for Journey-Genie is configured to simulate real-world conditions. This includes:

- **Server Configuration:** Setting up development, testing, and production servers.
- **Database Setup:** Preparing a test database with sample data to validate CRUD operations.
- **Network & Security Configurations:** Ensuring firewalls and access controls are in place.
- **Browser & Device Compatibility:** Testing across multiple browsers (Chrome, Firefox, Edge) and devices (mobile, tablet, desktop).

9.4 Defect Logging

Defect logging ensures that issues found during testing are documented and tracked for resolution. The process includes:

- **Identifying Defects:** Noting issues related to UI, functionality, performance, and security.

- **Logging in a Bug Tracking Tool:** Using Jira, Bugzilla, or similar tools to track defects.
- **Prioritizing & Assigning Defects:** Categorizing defects based on severity (Critical, Major, Minor) and assigning them to developers.
- **Re-Testing & Verification:** Ensuring fixes are applied and validated.

9.5 Test Cycle Closure

The test cycle closure marks the completion of testing for Journey-Genie. It involves:

- **Final Test Report:** Summarizing test results, defect trends, and overall software quality.
- **Lessons Learned:** Documenting insights for improving future testing cycles.
- **Sign-Off:** Obtaining approval from stakeholders to proceed with deployment.
- **Post-Production Monitoring:** Ensuring a smooth transition from testing to live deployment.

Conclusion

A structured testing approach ensures the Journey-Genie website meets high-quality standards, providing a seamless user experience. Each phase of testing plays a vital role in identifying and resolving issues before deployment.

Chapter 10:Advantages and Limitations of Journey-Genie

Advantages of Journey-Genie

1. Convenience for Customers

- Users can book tours and destinations online without visiting a travel agency.
- Provides real-time availability of packages, prices, and offers.

2. Operational Efficiency

- Reduces the need for physical offices, lowering operational costs.
- Automates booking, payments, and customer management, reducing manual workload.

3. Scalability

- Can expand to multiple regions without significant infrastructure investment.
- Easily integrates with third-party travel service providers and booking platforms.

4. Enhanced Customer Engagement

- Personalized recommendations based on past bookings and preferences.
- Loyalty programs, discounts, and promotional offers to retain customers.

5. Data-Driven Insights

- Analytics on customer travel preferences, peak seasons, and popular destinations.
- Helps in strategic planning and optimizing tour offerings.

6. Cost-Effective Marketing

- Digital marketing via social media, email campaigns, and targeted ads.
- Lower marketing costs compared to traditional travel agencies.

7. Seamless Booking & Payment Options

- Supports various payment methods, including credit/debit cards, UPI, and digital wallets.
- Reduces cash handling and enhances transaction security.

Limitations of Journey-Genie

1. Dependence on Internet & Technology

- Requires stable internet for customers and admin operations.
- Technical issues or downtime can disrupt bookings and transactions.

2. Logistics and Travel Uncertainties

- Unforeseen issues like flight delays, bad weather, or cancellations can affect itineraries.
- Dependence on third-party travel partners may impact service quality.

3. High Competition

- Competing with established platforms like MakeMyTrip, Expedia, and Airbnb.

- Requires strong marketing and unique selling points to attract users.

4. Initial Development Costs

- Investment needed for website, admin dashboard, and security features.
- Ongoing maintenance costs for updates and bug fixes.

5. Customer Trust & Retention

- Gaining credibility in the travel industry takes time.
- Negative reviews or bad experiences can harm brand reputation.

6. Regulatory and Compliance Issues

- Needs to comply with tourism regulations, travel insurance, and data protection laws.
- Managing taxes, refunds, and legal formalities for different destinations.

7. Limited Personal Interaction

- Lack of face-to-face assistance compared to traditional travel agencies.
- Difficult to provide personalized customer service without direct interaction.

Chapter 11: Future Scope of Journey-Ginie

Future Scope of Journey-Ginie Expansion to Multiple Locations Scaling the Journey-Ginie platform to cater to different cities and international destinations.

- ❖ Partnering with local travel agencies and tour operators for a wider market reach.
- ❖ Mobile App Development Launching a dedicated mobile app for seamless tour browsing and bookings. Integration of push notifications for personalized travel deals and trip reminders.
- ❖ AI-Powered Travel Recommendations Implementing machine learning algorithms to suggest travel packages based on user preferences. AI-driven chatbots for instant customer support, itinerary planning, and booking assistance.
- ❖ Integration with Smart Travel Technologies Augmenting the platform with IoT-enabled smart travel assistance (e.g., smart luggage tracking). Real-time monitoring of flight status, weather updates, and local attractions.
- ❖ Enhanced Booking & Payment System Developing a flexible payment system with EMI options for expensive travel packages.
- ❖ Integration with blockchain for secure and transparent travel transactions. Subscription-Based Travel Plans Offering membership plans for frequent travelers with exclusive discounts and perks.
- ❖ Personalized travel subscription services, including annual vacation planning and concierge services. Sustainability Initiatives Promoting eco-friendly travel options like carbon-neutral trips and sustainable accommodations.
- ❖ Partnering with local eco-tourism initiatives to support responsible travel. Collaboration with Corporate &

Events Providing corporate travel solutions for business trips, conferences, and retreats.

- ❖ Partnering with event organizers for curated travel experiences and destination weddings. Integration with Virtual Reality (VR) & Augmented Reality (AR)
Offering virtual tours of destinations to help users make informed booking decisions.
- ❖ AR-based interactive travel guides for real-time navigation and cultural insights. Loyalty & Referral Programs Introducing a reward system where users earn points for bookings and reviews. Referral programs allowing users to invite friends and earn travel credits.

Conclusion:

Journey-Genie aims to revolutionize the travel industry by leveraging digital solutions, automation, and personalized experiences. By integrating AI-powered recommendations, a seamless mobile experience, and sustainable tourism initiatives, it ensures convenience for travelers while optimizing business efficiency. The development process involves tackling technical challenges, including backend infrastructure, database management, and security protocols, while ensuring a smooth user experience. Through rigorous testing and continuous improvements, the platform enhances reliability and performance. The platform presents numerous advantages, such as cost-effective marketing, personalized recommendations, and an efficient booking system. However, challenges like high competition, logistics, and compliance with travel regulations require ongoing enhancements. Looking ahead, Journey-Genie has immense growth potential. Future advancements, such as blockchain-based payments, smart travel integrations, and VR-enabled

trip planning, will strengthen its position in the travel-tech industry. By continuously innovating and improving user engagement, Journey-Genie can evolve into a leading digital travel solution, catering to modern travelers' needs while maintaining a scalable and sustainable business model.

REFERENCE

- Geeks for geeks-<https://www.geeksforgeeks.org/introduction-to-spring-boot/>
- Kesari Tours- <https://www.kesari.in/>

- font awesome -<link rel="stylesheet" href="<https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.7.2/css/all.min.css>" integrity="sha512-Evv84Mr4kqVGRNSgIGL/F/aIDqQb7xQ2vcrdlwxfjThSH8CSR7PB EakCr51Ck+w+/U6swU2Im1vVX0SVk9ABhg==" crossorigin="anonymous" referrerpolicy="no-referrer" />
- MakeMyTrip- <https://www.makemytrip.com/>
- MySql- <https://dev.mysql.com/doc/workbench/en/>
- ClearTrip-
[Cleartrip: #1 Site for Booking Flights, Hotels & Buses Online](#)

- Chatgpt - <https://chatgpt.com/c/67dc205a-f7f8-800b-aebe-1544680f9dab>
- Google
- Incredible India:
<https://www.incredibleindia.gov.in/en/attractions>
- Spring-<https://spring.io/projects/spring-authorization-server>

