Contents

[LIST OF FIGURES 2](#_Toc204248304)

[1. INTRODUCTION 4](#_Toc204248305)

[2. Functional Requirements 4](#_Toc204248306)

[2.1 User Management 4](#_Toc204248307)

[2.2 Tour Packages Management 4](#_Toc204248308)

[2.3 Booking System 4](#_Toc204248309)

[2.4 Payment Integration 4](#_Toc204248310)

[2.5 Reviews and Feedback 5](#_Toc204248311)

[2.6 Contact and Inquiry 5](#_Toc204248312)

[2.7 Admin Dashboard 5](#_Toc204248313)

[2.8 Notifications 5](#_Toc204248314)

[3. Non-Functional Requirements 5](#_Toc204248315)

[3.1 Performance 5](#_Toc204248316)

[3.2 Security 5](#_Toc204248317)

[3.3 Usability 5](#_Toc204248318)

[3.4 Availability 6](#_Toc204248319)

[3.5 Maintainability 6](#_Toc204248320)

[3.6 Scalability 6](#_Toc204248321)

[3.7 Interoperability 6](#_Toc204248322)

[4. DOCUMENTATION OF THE WEB APP SYSTEM 6](#_Toc204248323)

[4.1. User management 6](#_Toc204248324)

[4.2. Tour Packages 6](#_Toc204248325)

[4.3. Booking System 6](#_Toc204248326)

[4.4. Payment integration 6](#_Toc204248327)

[4.5. Reviews and Ratings 6](#_Toc204248328)

[4.6. Admin Dashboard 7](#_Toc204248329)

[4.7. Security and Quality 7](#_Toc204248330)

[5. SCREENSHOTS OF THE PROJECT 7](#_Toc204248331)

[6. SOURCE CODE 11](#_Toc204248332)

# LIST OF FIGURES

[Figure 1: User registration and login 7](#_Toc204247466)

[Figure 2: Dashboard home for Admins 8](#_Toc204247467)

[Figure 3: Tour package with search filter 8](#_Toc204247468)

[Figure 4: Tour package booking form 9](#_Toc204247469)

[Figure 5: User booking history 9](#_Toc204247470)

[Figure 6: Contact details 10](#_Toc204247471)

[Figure 7: Reviews and Ratings 10](#_Toc204247472)

PROJECT TITLE: SUPERIORS TOURS AND TRAVEL SYSTEM.

TECHNOLOGY STACK: Python, Flask, PostgreSQL, Bootstrap, JavaScript, Chart.js

|  |  |  |
| --- | --- | --- |
| NAME | REGISTRATION NUMBER | STUDENT NUMBER |
| ARUHO PRISCILLAH | 23/U/24960 | 2300724960 |
| ANDIMASHIMWE RHODA | 23/X/25067/PS | 2300725067 |
| BAHATI BRENDA KIZITO | 23/U/07321/EVE | 2300707321 |
| BUYINZA SAMUEL | 23/U/07787/PS | 2300707787 |
| NUWAHEREZA DENNISE | 23/U/16238/PS | 2300716238 |

# INTRODUCTION

Superiors Tours and Travel system is a web-based travel management system designed to simplify the booking of travel packages for tourists while offering an administrative dashboard for tour operators. It allows users to register, browse and book tour packages, make secure payments, submit feedback, and interact through inquiries, all in a user-friendly and responsive interface.

# Functional Requirements

## 2.1 User Management

* FR1: Users shall be able to register for an account.
* FR2: Users shall be able to log in and log out securely.
* FR3: Users shall be able to manage and update their profile.
* FR4: Admins shall be able to manage (create, update, delete) user accounts.

## 2.2 Tour Packages Management

* FR5: Admins shall be able to create, edit, and remove tour packages.
* FR6: Users shall be able to view all available tour packages.
* FR7: Users shall be able to search/filter tour packages by date, destination, or price.

## 2.3 Booking System

* FR8: Users shall be able to book tour packages.
* FR9: The system shall generate booking details and a reference number.
* FR10: The system shall send email confirmations for successful bookings.
* FR11: Users shall be able to view and manage their bookings.

## 2.4 Payment Integration

* FR12: The system shall integrate with a secure payment gateway (e.g., PayPal/Stripe).
* FR13: The system shall confirm transactions and update booking statuses accordingly.

## 2.5 Reviews and Feedback

* FR14: Users shall be able to submit reviews and ratings after completing tours.
* FR15: The system shall calculate and display average ratings for tour packages.

## 2.6 Contact and Inquiry

* FR16: Users shall be able to submit inquiries via a contact form.
* FR17: Admins shall receive email notifications for new inquiries.

## 2.7 Admin Dashboard

* FR18: Admins shall be able to manage users, tours, and bookings.
* FR19: Admins shall be able to view reports, revenue, and popular tours.

## 2.8 Notifications

* FR20: The system shall send notifications for booking confirmations, cancellations, and payment updates.

# Non-Functional Requirements

## 3.1 Performance

* NFR1: The system shall respond to user requests within 2 seconds 90% of the time.
* NFR2: The system shall process bookings and payments within 5 seconds.

## 3.2 Security

* NFR3: Passwords shall be stored using secure hashing.
* NFR4: The system shall use HTTPS for all data transmissions.
* NFR5: Input validation shall prevent SQL injection and XSS attacks.
* NFR6: Role-based access shall be enforced for users and admins.

## 3.3 Usability

* NFR7: The UI shall be intuitive, clean, and responsive across all devices.
* NFR8: The system shall display clear error messages and form validation.

## 3.4 Availability

* NFR9: The system shall maintain 99% uptime on production servers.
* NFR10: System errors and exceptions shall be logged for maintenance.

## 3.5 Maintainability

* NFR11: Code shall follow Flask and PEP 8 standards.
* NFR12: Code shall be modular and reusable via Flask Blueprints.

## 3.6 Scalability

* NFR13: The system shall support up to 500 concurrent users.
* NFR14: The system shall allow scalable infrastructure for future growth.

## 3.7 Interoperability

* NFR15: The system shall provide REST APIs for mobile or third-party integrations.

# DOCUMENTATION OF THE WEB APP SYSTEM

## 4.1. User management

* + - * + Register, login, logout securely
        + User profile management

## 4.2. Tour Packages

* + - * + Administrators can create, update, and delete packages
        + Users can view, search, and filter packages

## 4.3. Booking System

* + - * + Users book tours and get confirmation via email
        + Booking history view and management

## 4.4. Payment integration

* + - * + Secure simulated payments
        + Transaction confirmation and updates

## 4.5. Reviews and Ratings

* + - * + Users submit feedback on tours
        + Admin can view customer feedback and ratings

## 4.6. Admin Dashboard

* + - * + Manage users, bookings, and destinations
        + View analytics via chart.js

## 4.7. Security and Quality

* + - * + Implements secure password hashing and role-based access
        + Input validation, error handling, and HTTPS support
        + Built following PEP 8, Flask conventions, and modular structure

# SCREENSHOTS OF THE PROJECT

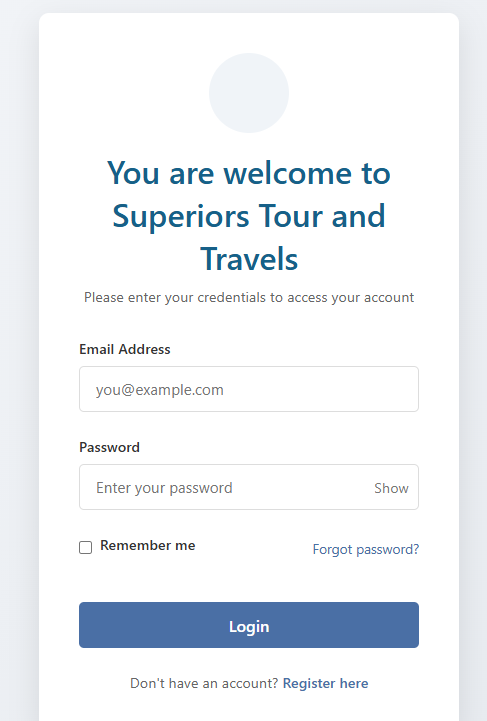


Figure 1: User registration and login

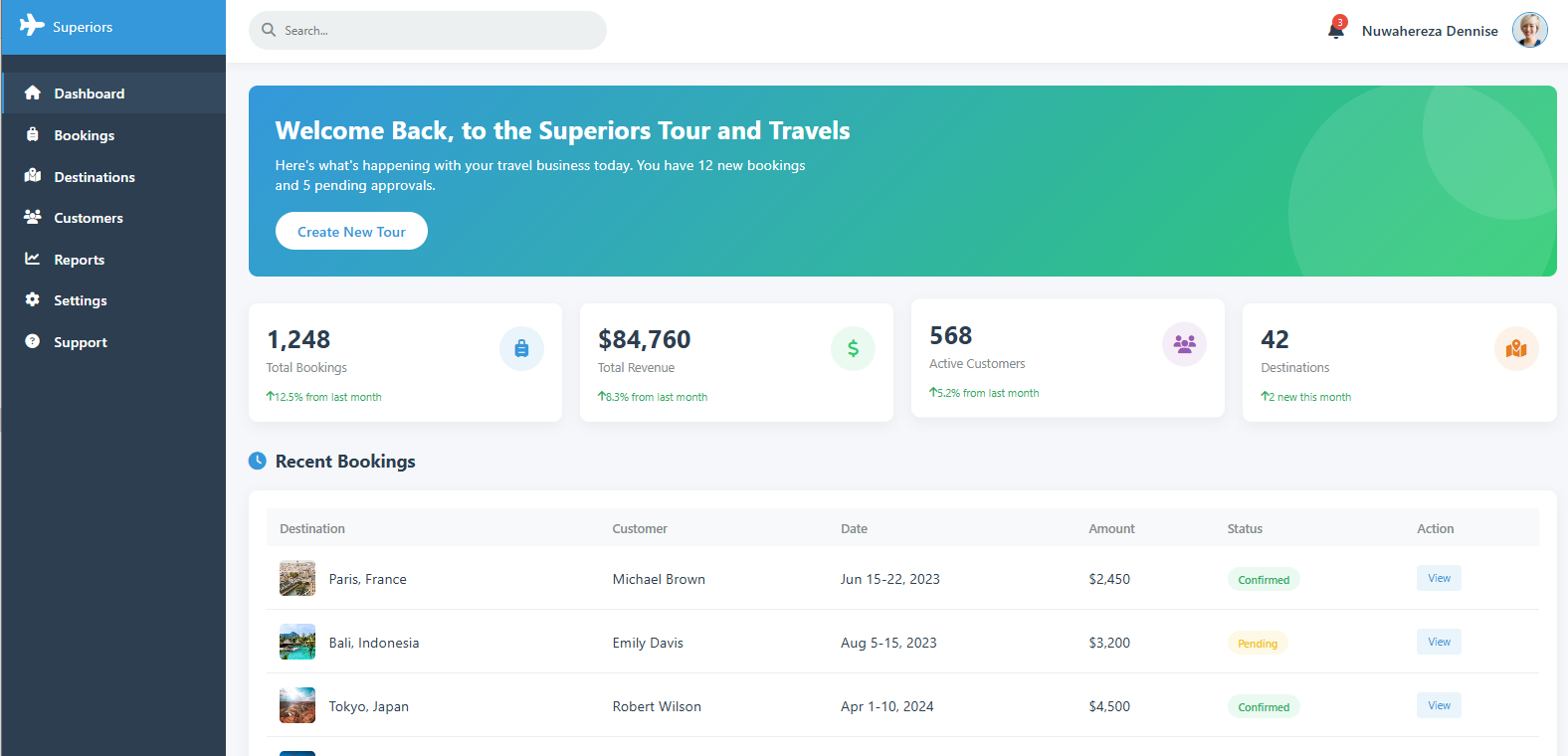


Figure 2: Dashboard home for Admins

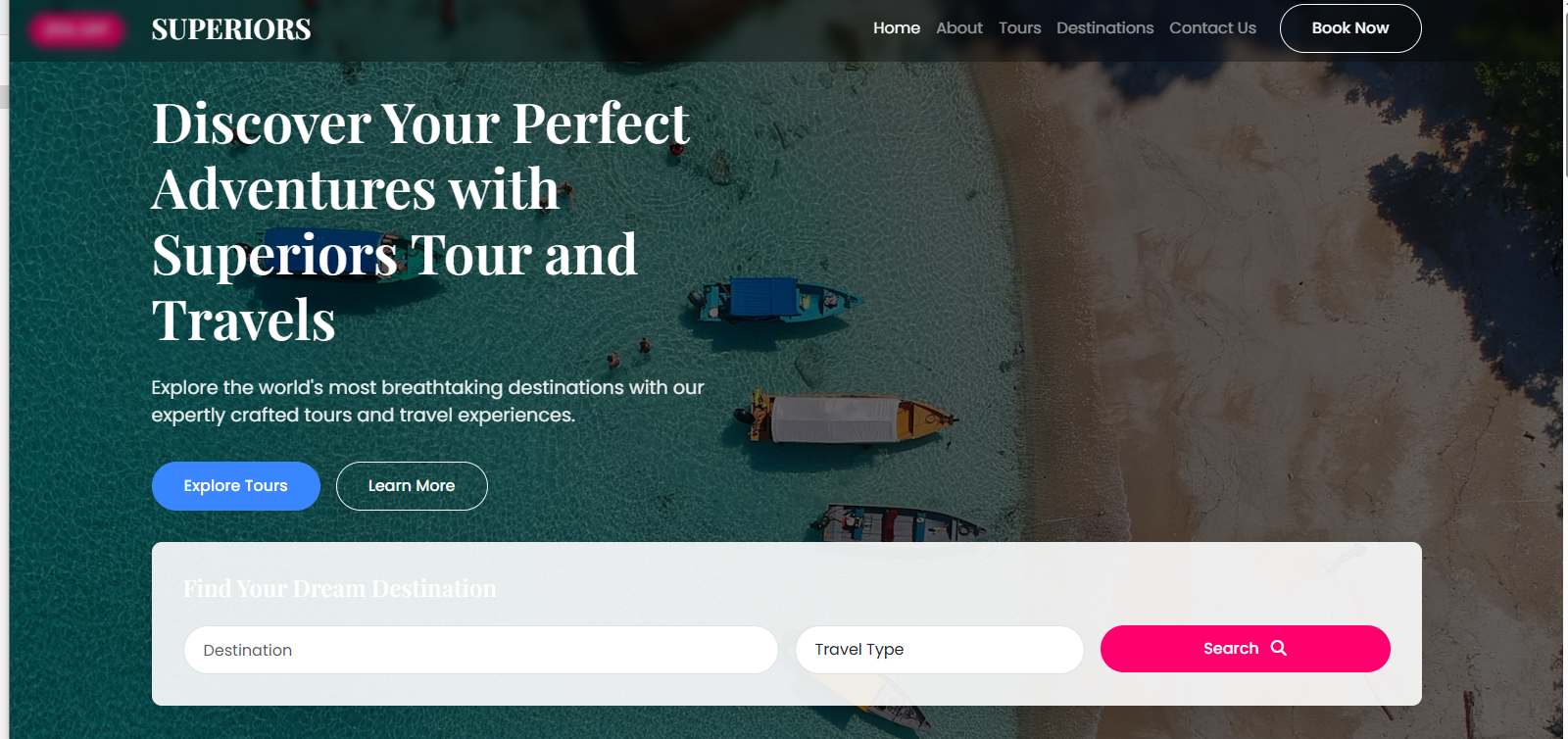


Figure 3: Tour package with search filter

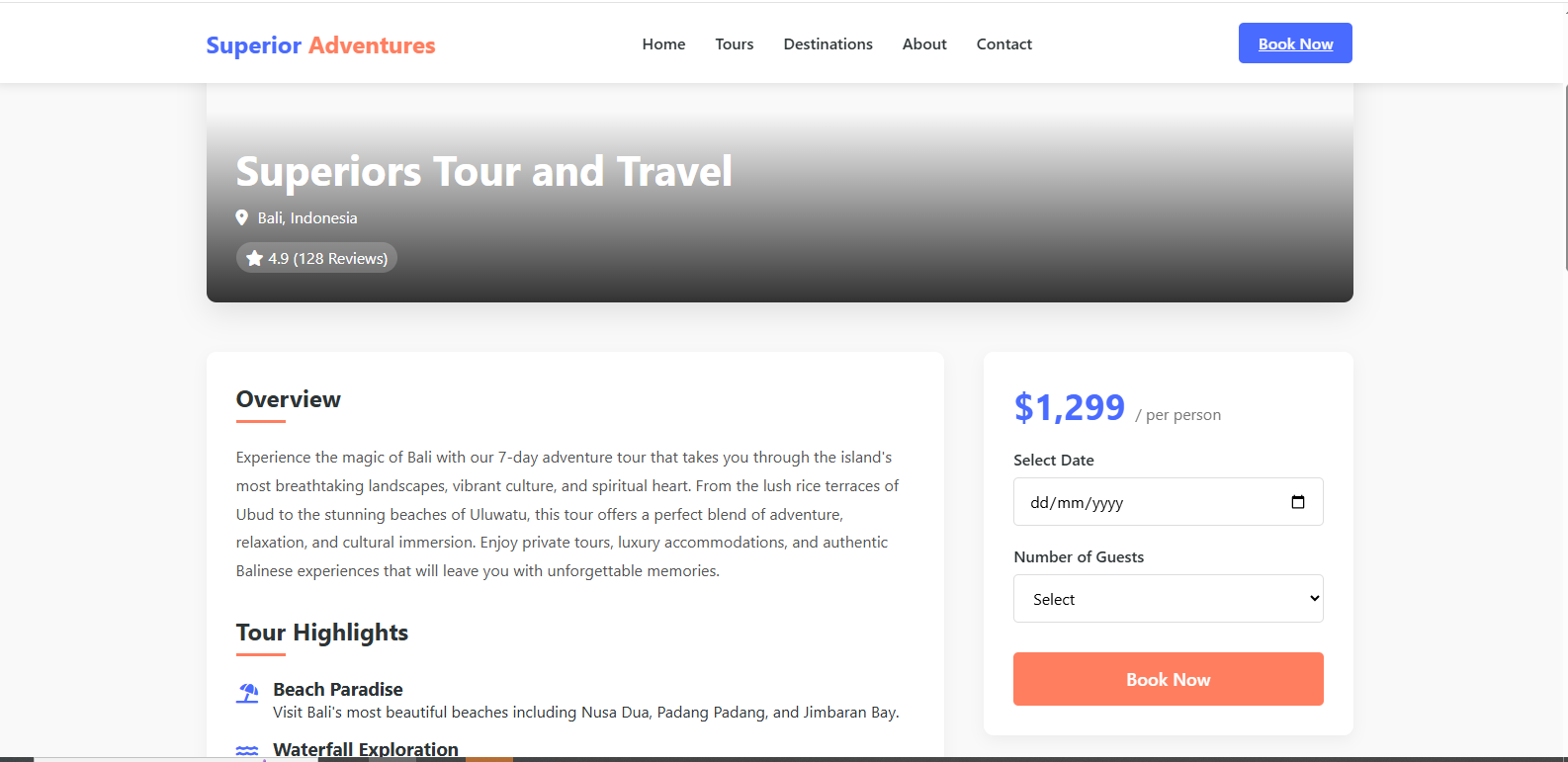


Figure 4: Tour package booking form

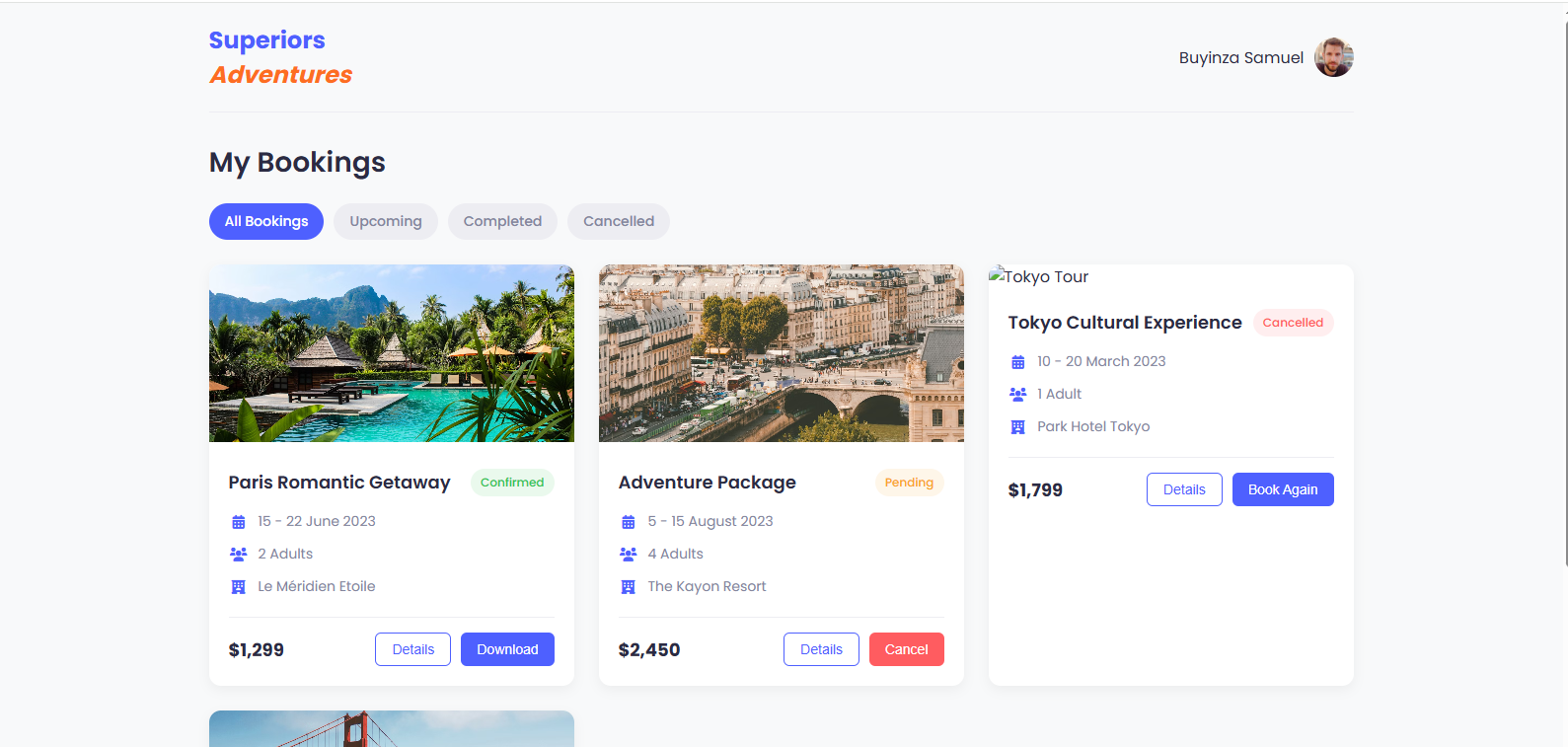


Figure 5: User booking history

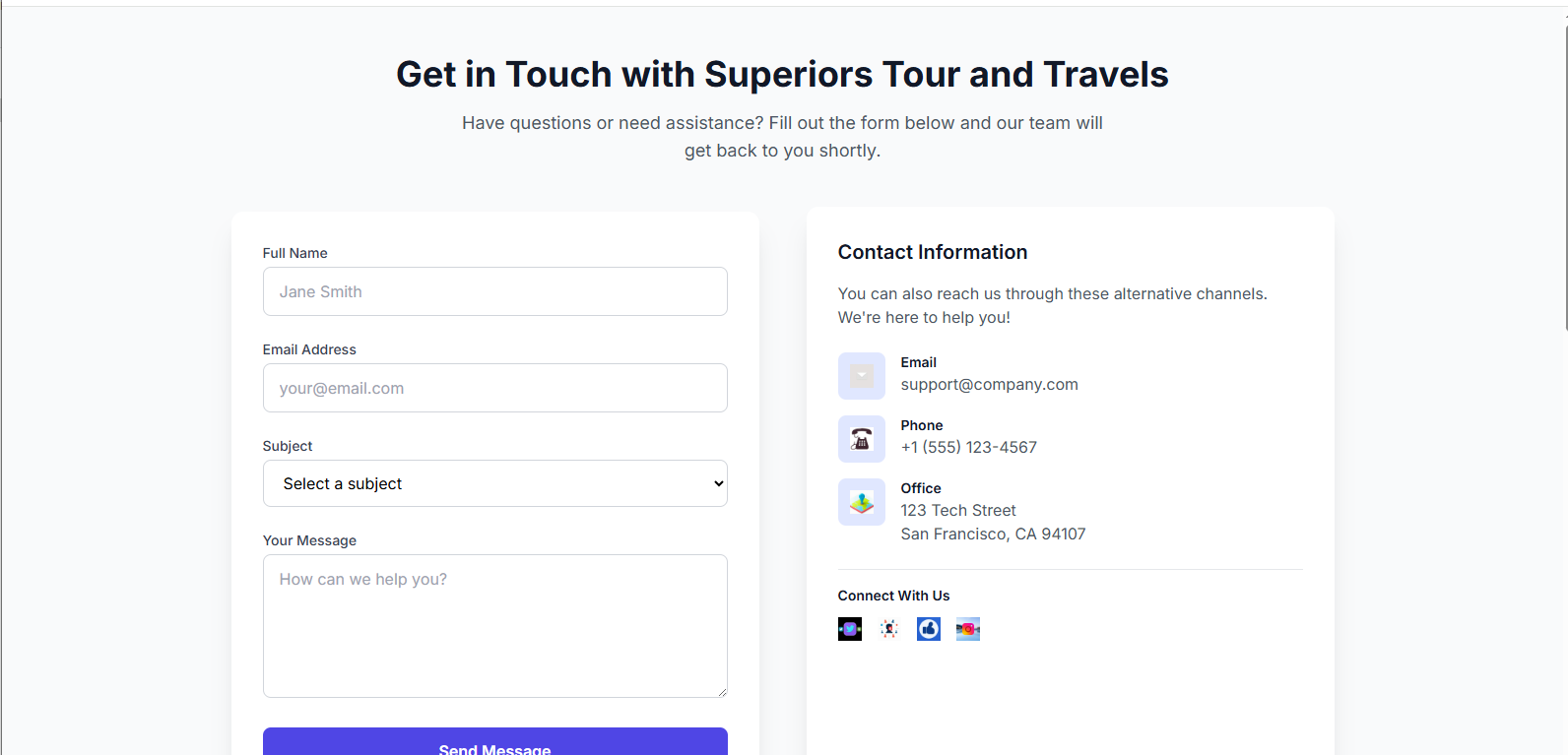


Figure 6: Contact details

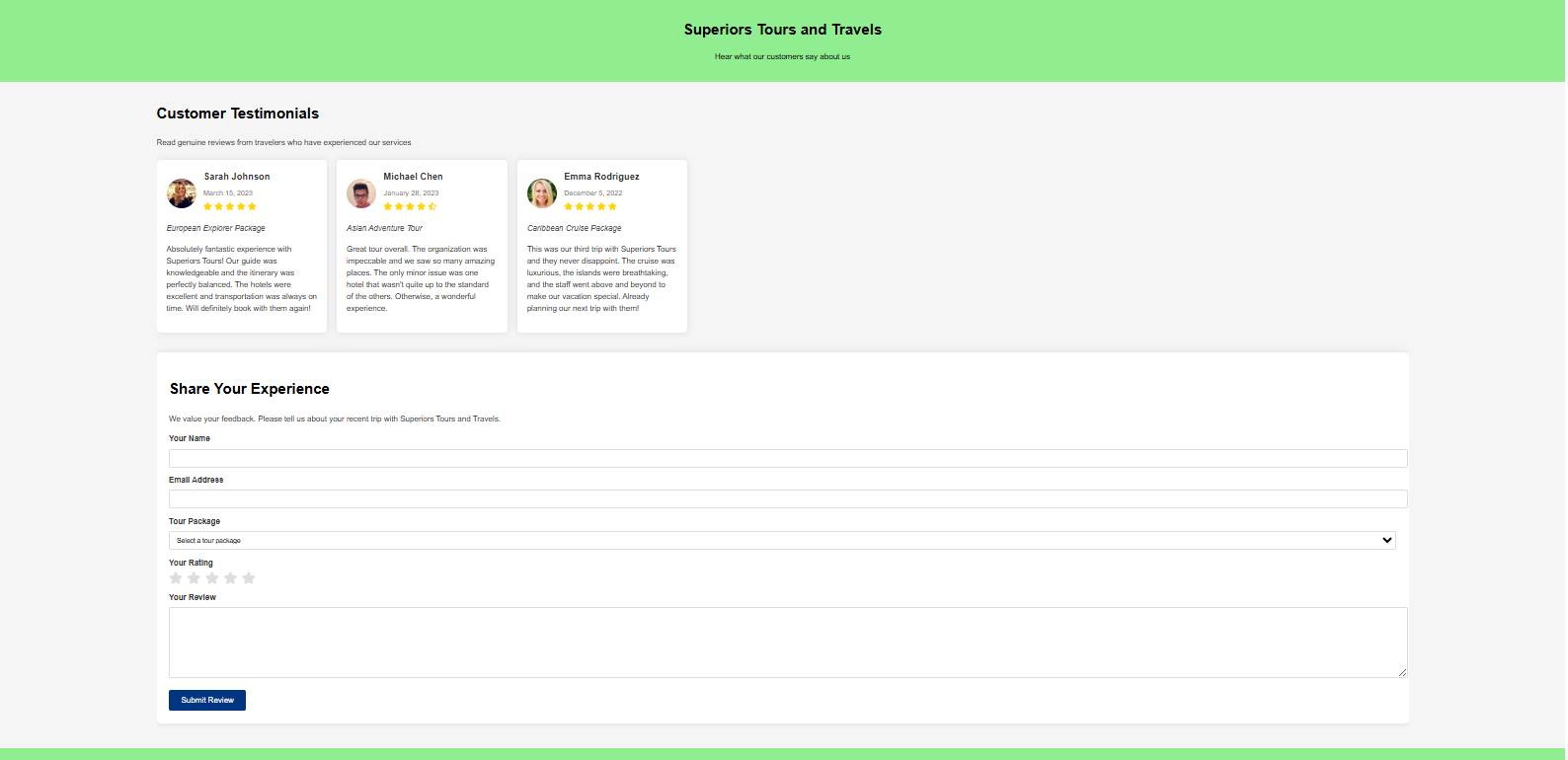


Figure 7: Reviews and Ratings

# SOURCE CODE

https://github.com/buyinzasamuel/GROUP-F-SUPERIOR.git