TRIBHUVAN UNIVERSITY

Faculty of Humanities and Social Sciences

PROJECT REPORT ON

Blissful Holiday: Tour and Travel System

In partial fulfillment of the requirements for the Bachelors in Computer Application

Department of Computer Application

Vedas College

Jawalakhel, Lalitpur

Submitted by:

Abhaya Dhakal (6-2-149-55-2020)

Manish KC (6-2-503-29-2020)

Under the Supervision of

Harendra Raj Bist

Submitted to:

Tribhuban University

Faculty of Humanities and Social Sciences

Date of Submission:

November 26, 2024

VEDAS COLLLEGE

Affilated to Tribhuvan University

SUPERVISOR'S RECOMMENDATION

I hereby recommend that this project prepared under my supervision by Abhaya Dhakal and Manish KC entitled "**Tour and Travel System (Blissful Holiday)"** in partial fulfillmentof the requirements for the degree of Bachelor of Computer Application is recommended for the final evaluation.

……

Harendra Raj Bist

Project Supervisor

Vedas College

Jawalakhel, Lalitpur

VEDAS COLLLEGE

Affilated to Tribhuvan University

LETTER OF APPROVAL

This is to certify that this project prepared by Abhaya Dhakal and Manish KC entitled "**Tour and Travel System"** in partial fulfillment of the requirements for the degree of Bachelor in Computer Application has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

SIGNATURE of Supervisor …… MR. Harendra Raj BistaVedas College Jawalakhel, Lalitpur	**SIGNATURE of HOD/ Coordinator** …… MR. Harendra Raj Bist Vedas College Jawalakhel, Lalitpur
SIGNATURE of External Examiner …… **Internal Examiner**	**SIGNATURE of External Examiner** …… **External Examiner**

ACKNOWLEDGEMENT

Without the kind support and assistance of many individuals and institute this project would not have been success. We are very thankful and blessed to have got all this support and guidance for the completion of the project. We would like to extend our sincere gratitude to each and every one who have been directly and indirectly involved in this project.

We would like to thank to our project supervisor **Mr. Harendra Bist** who has always guided us throughout the project. We are very thankful for his idea, experience, information and knowledge for developing a functional web application. We would also like to thank him for his encouragement and motivation for making this report standard as per the university rule.

Vedas College is an unmissable institution that deserve our huge gratitude. We are very thankful for the constant guidance and supervision also providing necessary resources, information and friendly environment for the successful completion of the project. The effort and dedication provided by the director **Mr. Prashant Thakur** is also highly acknowledged, without his guidance the project would have been far from completion.

All teaching staff and lecturers who helped us throughout the project, we would like to thank them as well. All theirs help and support is highly appreciated. The is our fortune that we got such supportive teaching staff and lecturers. They also deserve a huge credit for the completion of the project.

Our thanks and gratitude are also for each and every one of our friends and colleagues for their continuous support and motivation for the completion of the project. Team members of Blissful Holiday.

Abhaya Dhakal

Manish Kc

ABSTRACT

Blissful Holiday is a project initiated by the TEAM which indicates “A good facility to tourists and improve tourism of our country Nepal”. In this project, we will integrate a recommendation feature through which a person who wants to visit in different places can easily find their destination, which will be recommended by the system according to the user preferences. In this project, we will develop a web application through which a person who wants to visit in Nepal can have a good idea of visiting a place. Likewise, we are going to provide international tours for Nepali people at very less cost in countries like India, Tibet, Bhutan.

Many tourists are facing difficulty while traveling in Nepal as well as Nepalese people who travel within the country and outside the country. All kinds of tourists will feel free to travel and enjoy their trip with the packages provided by us.

Keywords: Blissful Holiday, Tourism, Nepal, Web application, Visitor, Place, Home stay, Hospitality, International tourist, Local tourists, international tours, Traveling, Packages, Trip

Table of Contents

LIST OF ABBREVIATIONS

Css: Cascading Style Sheet

Html: Hyper Text Markup Language

Er Diagram: Entityrelationship Diagram

Sdlc: System Development Life Cycle

Dfd: Data Flow Diagram

Ide: Integrated Development Environment

LIST OF FIGURES

LIST OF TABLES

Chapter 1: Introduction

1.1 Introduction

Tourism industry is one of the most significant industries throughout the world and its importance is growing rapidly in recent years especially in Nepal. Tourism has become an important sector of the global economy, with millions of people traveling each year to destinations around the world. It contributes significantly to the gross domestic product (GDP) of many countries and provides employment opportunities for millions of people worldwide.

Project Blissful Holiday is based upon tour and travel where we provide different facilities to the tourists. Through this system tourists from different places can easily book their packages at very reasonable prices and this system will let users know more about the place by which they can decide whether they want to visit that place or not. We are going to create a platform where tourists can easily book their trip according to them and make them enjoy their trip freely without any tension.

Also, the new rule has been made by our Nepal Government for the tourists to take a guide with them compulsorily for their safety. Through this system tourists can book their personal well-trained and highly rated guide who will help them throughout their trip so that they will not get distracted and can successfully reach their destination. Packages that are provided by our system will be very budget friendly as well as trustworthy. Although the price rate of packages will be low, we will give the best facility on a low budget. [1]

1.2 Problem Statement

Problem Definition:

Many tourists who visit Nepal for the first time must face difficulty while visiting Nepal. Tourists cannot decide which place to visit, and they also cannot assume the price of the product that is found here where they get cheated by the locals. Also because of the lack of genuine guidance many tourists get offended and never come back again to visit Nepal. Likewise, there are many tourists who cannot select a proper place to visit in Nepal. According to the news report many tourists who are visiting in Nepal are getting frauded by the tourist guide and by locals.

Many tourists visiting Nepal for the first time encounter challenges, such as selecting destinations and budgeting accurately, which often leads to overspending or instances of

being overcharged by locals. Additionally, without a reliable guide, tourists can feel lost and disappointed with their experience. These negative encounters discourage return visits and harm Nepal's reputation as a travel destination. Project Blissful Holiday seeks to address these issues by providing a recommendation-based travel booking system that connects tourists with genuine, reliable guidance.

1.3 Objectives

Objectives of the Project:

- To provide a better facility for the tourists (guests) through our system.
- To prevent tourists from getting scammed by guides and locals.
- To provide an enhanced facility for tourists through a recommendation-based travel system.
- To prevent tourists from being confused to choose a best destination.

1.4 Scope and limitation

1.4.1 Scope

Online Booking: You can offer a platform for travelers to search, compare, and book flights, hotels, tours, and other travel-related services. This can make travel planning more convenient for your users.

Destination Information: Provide detailed information about various travel destinations, including attractions, weather, culture, and local tips to help users plan their trips.

Customization: Offer tools for users to create personalized itineraries and travel packages based on their preferences and budget.

Mobile Compatibility: Ensure that your website is responsive and mobile-friendly, as many travelers use smartphones for booking and planning on the go.

Destination Recommendation: Incorporate social media features to allow users to share their travel plans and experiences, which can help in word-of-mouth marketing.

Online Payment: Consider offering content in multiple languages to cater to a diverse audience of travelers.

1.4.2 Limitations:

Competition: The travel industry is highly competitive, with many established players. Gaining visibility and trust can be challenging.

Data Accuracy: Ensuring that the information on your website, such as prices and availability, is up to date can be a constant challenge.

Security Concerns: Dealing with user data and payment information requires stringent security measures to protect against data breaches and fraud.

Licensing and Regulations: The travel industry is subject to various regulations and licenses, depending on your location and the services you provide. You need to comply with these regulations.

Technical Challenges: Maintaining a website that can handle high traffic, especially during peak travel seasons, can be technically challenging.

Customer Support: Providing responsive customer support for booking issues, cancellations, and other inquiries is crucial but can be resource intensive.

Dependence on Third Parties: You may rely on third-party providers for services like booking systems and payment processing, which can introduce vulnerabilities and limitations beyond your control.

Marketing Costs: Attracting and retaining customers may require significant marketing expenditures, including SEO, advertising, and promotions.

Global Reach: Expanding your services to international markets can be complex due to currency conversions, language barriers, and cultural differences.

User Trust: Building trust among users is vital, as travelers often rely on reviews and recommendations. Negative experiences or reviews can harm your reputation.

1.5 Methodology

The Agile model is a popular and widely adopted software development life cycle (SDLC) methodology. It emphasizes flexibility, collaboration, and iterative development. Here are some key characteristics of the Agile model because of which we are using Agile methodology:

Iterative and Incremental: Agile projects are divided into short iterations and sprints, typically lasting one to four weeks. Each sprint focuses on delivering a working increment of the software. The product evolves through successive iterations, with frequent releases of new features.

Customer Collaboration: Agile methodologies emphasize active involvement and collaboration with customers or stakeholders. Regular feedback from customers is crucial for understanding their needs and incorporating changes throughout the development process.

Adaptive Planning: Rather than detailed upfront planning, Agile projects rely on adaptive planning. Requirements are captured in a prioritized backlog, and the development team selects a set of high-priority items to work on.

Figure 1.1 Agile Mode

1.6 Report Organization

This is the final report of our system creation, and we created the organization of the report according to the guidance of the TU (Tribhuvan University). According to the guidance of TU we divided our report into six parts which are explained below.

i. Introduction

Chapter one, includes the overall view of the project i.e., the basic problem definition and the general overview of the problem which describes the problem in layman terms. It also specifies the software used and the proposed solution strategy.

ii. Background Study and Literature Review

In chapter two, it is further divided into two parts i.e. Background Study and Literature Review. The background study consists of a review of the area being researched, current information surrounding the issues, previous studies on the issue, and relevant history on the issue. Ideally, the study should effectively set forth the history and background information on your thesis problem. It is a written overview of major writing and other sources on a selected topic. Sources covered in the review may include government reports, Web sites,etc.

iii. System Analysis and System Design

Chapter 3, consists of two major parts System Analysis and System Design. The system analysis in this report for the project's major stakeholders, the client, the academic supervisors and the development team. The purpose of this section is to identify and describe the system requirements and constraints on the system. Such as requirement analysis feasibility study etc.

This part contains the design of the system we are building. There are many ways we can show the design of the system such as through data modelling, process modelling. Interface design etc. We use a structure approach to design our system. We use ER diagram, form and report to show the overview of the system.

iv. Implementations and Testing

Chapter 4, consists of two more major part Implementation and Testing. This part of the report contains the process of the software development life cycle which describes the process how the system is build such as which tool we used to manage the process of SDLC, which languages we use and which IDE used to code the program and many more.

v. Conclusion and Future Recommendation

Chapter five is Conclusion and Future Recommendation part. The conclusion part represents the overall overview of the system we build. What are the consequences occurring during the project life cycle and future recommendations for the system I.e., what type of future update can be added to the system? In which field can thissystem be used.

Chapter 2: Background Study and Literature Review

2.1 Background study:

Our project is based on the tourism sector, as we all know that our country's one of the main sources of income is the tourism sector. There is no proper hospitality to the tourist. Although there are many businesses that are based on tourism but also there are so many cases where tourists return to their places leaving bad reviews. So, to provide proper hospitality for the tourists and increase tourism in our country our team have planned to build a tour and travel system.

i. Current Status:

Many tourists are visiting Nepal at very high rate. To make them feel comfortable with the environment of Nepal online service for tourists is in very high need. As the tourists do not feel easy to visit in Nepal. So, to provide proper guidance to the tourists our online system for tours and travel is very important.

ii. Currently running system:

Very few tours and travel system are there in Nepal. Although there are many offline businesses going on in Nepal but in context of online servicing very few systems are there like Luxury tours and travel, Blue Mountain tours and travel etc.

2.2 Literature Review

2.2.1 Study of Tourism in Nepal

Tourism is at the forefront of these developments: as a driver of ICT introductions, an arena for testing & trialing, and a global market. This paper critically examines these developments and their linkages to tourism and sustainability goals, concluding that existing academic assessments are optimistic, simplistic and monocausal, with a focus on business and marketing opportunities. Tourism appears to have developed through four stages of ICT adoption - opportunity, disruption, immersion and usurpation -, which reflect new opportunities and risks, and the need for more critical evaluations of the implications of the ICT economy. [3]

2.2.2 Study of digitalization of Tourism in Nepal

Digital technologies have become important in tourism business and their development is reshaping this industry. This paper discusses the role of emerging technologies on the digital tourism business ecosystems model for Nepal. It highlights the role of emerging technology and the value they add to the current digital business systems. The study employs design science research methodology and considers the current designs, current implementation scenarios and the current technological status of Nepal to elaborate the role and application of emerging technologies in tourism business. This work holds a great significance as it brings out the knowledge on use of emerging technologies in tourism business of Nepal. This will help the government, business houses and supporting business entities to identify new business opportunities and build new innovative business models for the tourism domain. The work uses extensive figures and illustrations to explain the role of emerging technologies in tourism business of Nepal. [4]

2.2.3 Study on A strategic website evaluation of online travel agencies

Online travel Web sites have been the most frequently visited online information facilities by travelers. To evaluate the effectiveness of a travel Web site, the Web site manager should regularly check whether or not it is fulfilling the objectives that were established for it. This research uses a strategic Web site evaluation framework to introduce a five-stage process for examining the consistency of Web site's presence and its intended strategies. Two leading online travel agencies with different business strategies are selected to demonstrate methods of implementing a strategic evaluation framework and to compare the evaluation results. A hierarchical evaluation structure is introduced to explicitly delineate the two Web sites' different strategy intentions and related evaluation criteria. Results show that an individual Web site's strategy-inconsistent criterion can be easily identified through a gap analysis and criteria performance matrix. A strategy-inconsistent dimension can be discovered through a radar chart analysis of the 4PsC (Product, Promotion, Price, Place, and Customer Relationship) dimensions and a transaction phases analysis. [5]

Chapter 3: System Analysis And design

3.1 System analysis

3.1.1 Requirement Analysis

i. Functional Requirement

Functional requirements for a tour and travel system define the specific features and capabilities the system must have to meet its intended goals and provide value to users. Here are some key functional requirements for a tour and travel system:

User Registration and Authentication:

Users should be able to create accounts and log in securely. Password reset and recovery options should be available.

Booking: Users should be able to search tours and other travel services.. Booking and reservation functionality with confirmation emails.

User Profile Management: Users should be able to update their personal information, preferences.

Travel Information: Detailed information about destinations, including attractions, weather, local culture, and visa requirements. User-generated content such as reviews and ratings.

Itinerary Planning: Tools for users to create and customize travel itineraries. Suggestions for activities, tours, and accommodations based on user preferences.

Booking Management: Users should be able to view and manage their bookings.

Destination Recommendation: System should recommend user according to their preferences.

Customer Support: Contact forms or chat support for users to get assistance with booking or travel-related queries. FAQ section for common questions.

Admin Panel: A back-end system for administrators to manage user accounts, bookings, and content. Tools for content management, including adding, editing, and deleting travel information.

.

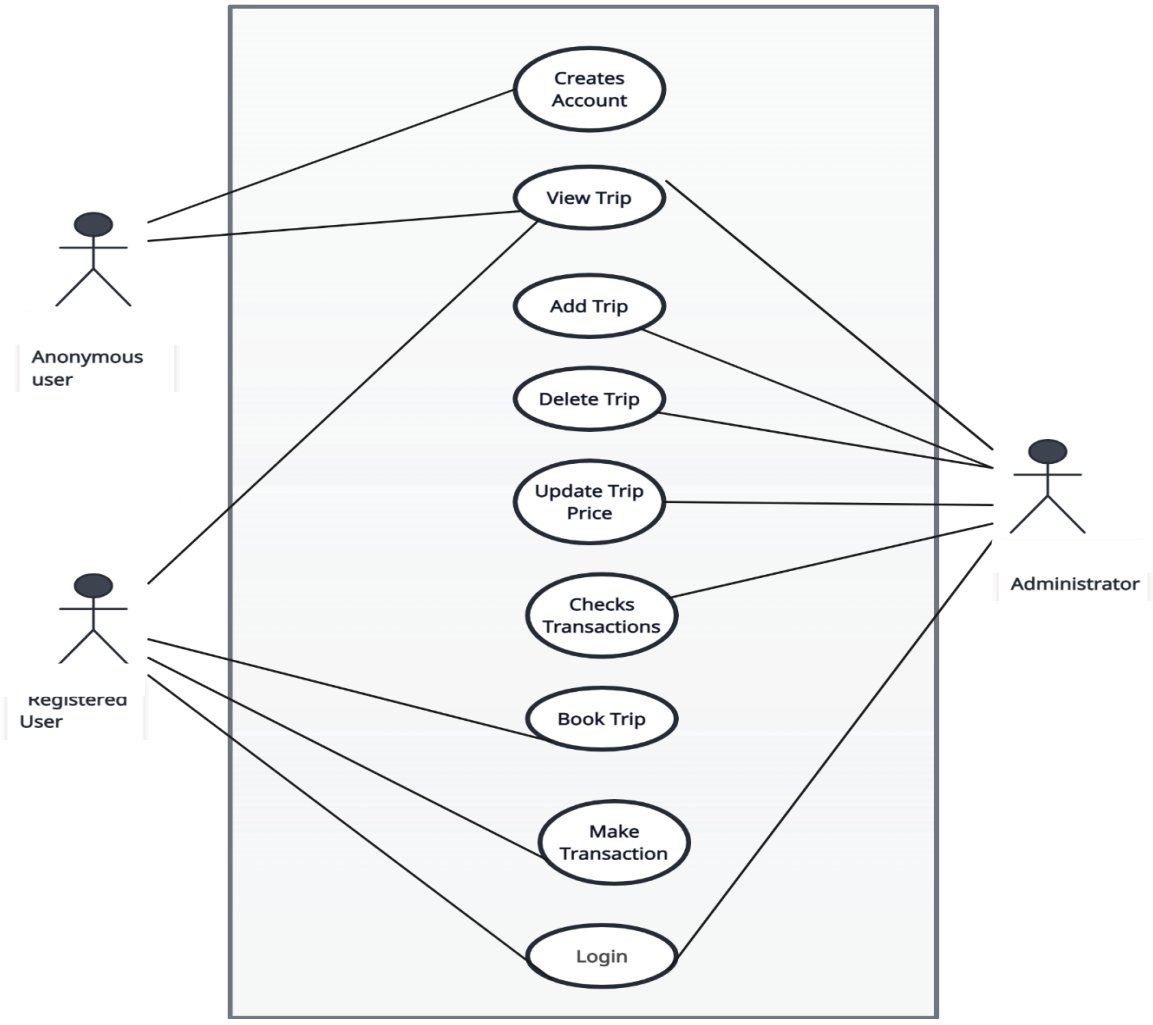


Figure 3.1 Use Case Diagram

ii. Non-functional requirements

Non-functional requirements for a tour and travel system define the qualities, constraints, and performance characteristics that the system must possess. These requirements are just as crucial as functional requirements for ensuring the overall effectiveness and user satisfaction with the system. Here are some non-functional requirements for a tour and travel system:

Performance:

Response Time: The system should provide quick responses to user requests, such as search queries and booking processing.

Scalability: The system must handle an increasing number of users and transactions, especially during peak travel seasons.

Load Handling: It should be capable of handling heavy loads without crashes or slowdowns.

Reliability: Availability: The system should be available 24/7, with minimal downtime for maintenance or updates.

Fault Tolerance: It should continue functioning even in the presence of hardware or software failures.

Security:

Data Security: Protect user data, including personal information and payment details, from unauthorized access and breaches.

Authentication and Authorization: Implement robust authentication and authorization mechanisms to ensure that only authorized users can access certain features and data.

Usability and User Experience:

User-Friendly Interface: The system should have an intuitive and easy-to-navigate user interface.

Consistency: Maintain a consistent look and feel throughout the website.

Compatibility:

Device Compatibility: Ensure compatibility with different devices, including desktops, laptops, tablets, and smartphones.

Scalability:

Database Scalability: The database should scale to accommodate a growing volume of data.

Infrastructure Scalability: The system should be able to scale horizontally or vertically to meet increased demand.

Data Backup and Recovery: Regularly backup user and system data to prevent data loss in case of unforeseen events. Implement robust data recovery procedures to restore the system to its last known good state.

3.1.2 Feasibility Analysis

A feasibility study for a tours and travel system is a critical initial step in evaluating whether it's viable to develop and implement such a system. It involves assessing various aspects, including technical, operational, economic, legal, and scheduling, to determine if the project is worth pursuing. Here's a breakdown of the key components of a feasibility study for a tours and travel system:

i. Technical Feasibility:

Hardware and Software Requirements:

Determine the technical infrastructure needed for the system, including servers, databases, and software.

Development Skills:

Assess whether your team or potential developers possess the necessary technical expertise to build and maintain the system.

Integration:

Analyze how the system will integrate with existing systems or third-party services, such as payment gateways and booking engines.

ii. Operational Feasibility:

User Acceptance: Gauge the willingness of employees and users to adopt the new system.

Training Requirements: Determine the need for training and support during system implementation and afterward.

Change Management: Identify potential challenges related to organizational changes and how they will be addressed.

iii. Economic Feasibility:

Cost-Benefit Analysis: Calculate the total cost of developing, implementing, and maintaining the system versus the expected benefits, including revenue, cost savings, and improved efficiency.

Return on Investment (ROI):

Assess the projected ROI and payback period. Funding Sources: Determine how the project will be funded, whether through internal resources, loans, or external investors.

3.1.3 Data Modeling (ER diagram)

ER diagram

The ER diagram we created using the tools called Lucid Chart which is online tools to create UML diagram and many software engineering related tools. We put only few of the attributes of the entity as we required large space for the Complete ER diagram.

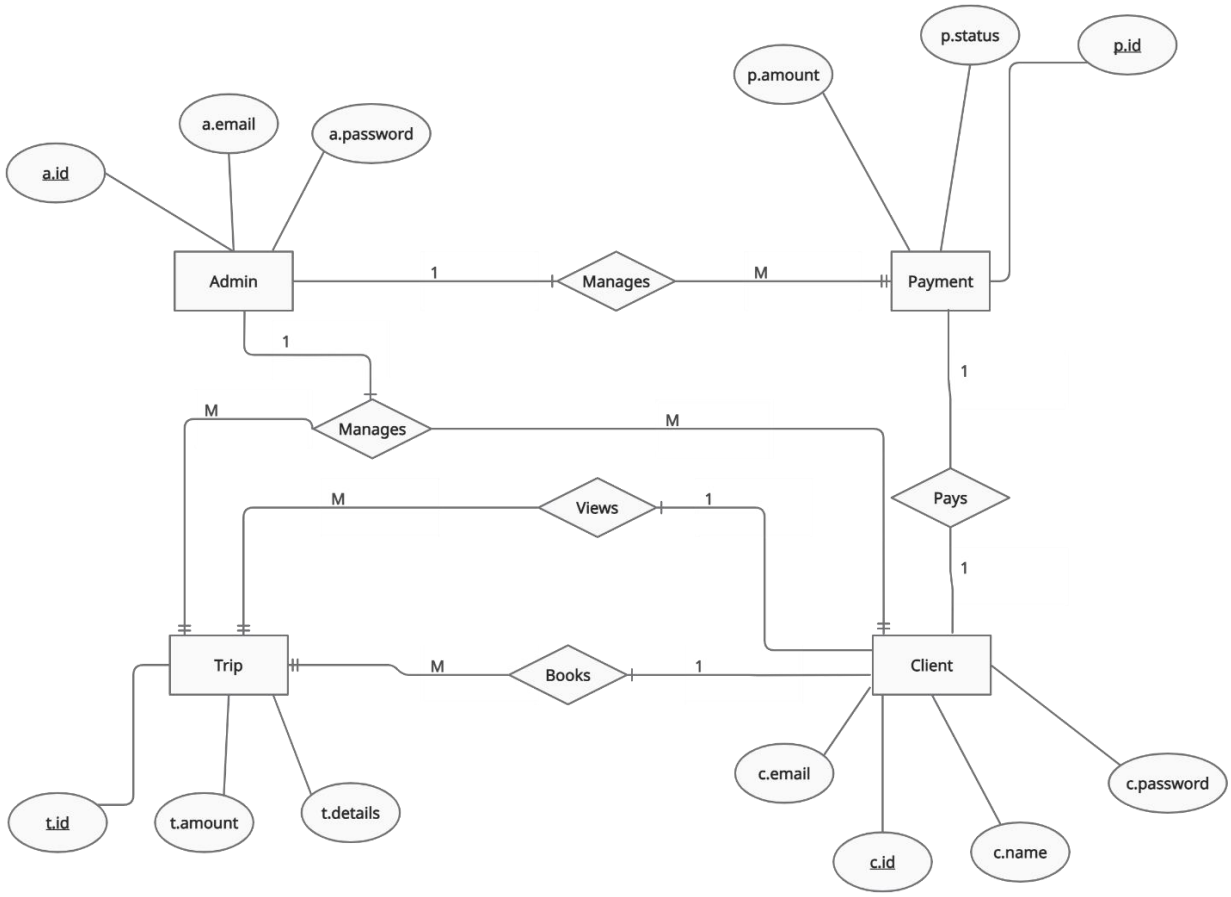


Figure 3.2: ER Diagram

As we can see the figure 3.2 the client's views trip with its details, amount. Customer can also book the trip accordingly. Admin can manage trip clients and payment done by the client.

3.1.4 Data Flow Diagram

We have created the DFD diagram for our system according to the use case diagram. DFD for tour and travels, manages trips, manages booking and many more. DFD will show how our system taking data from user and how it is handling it.

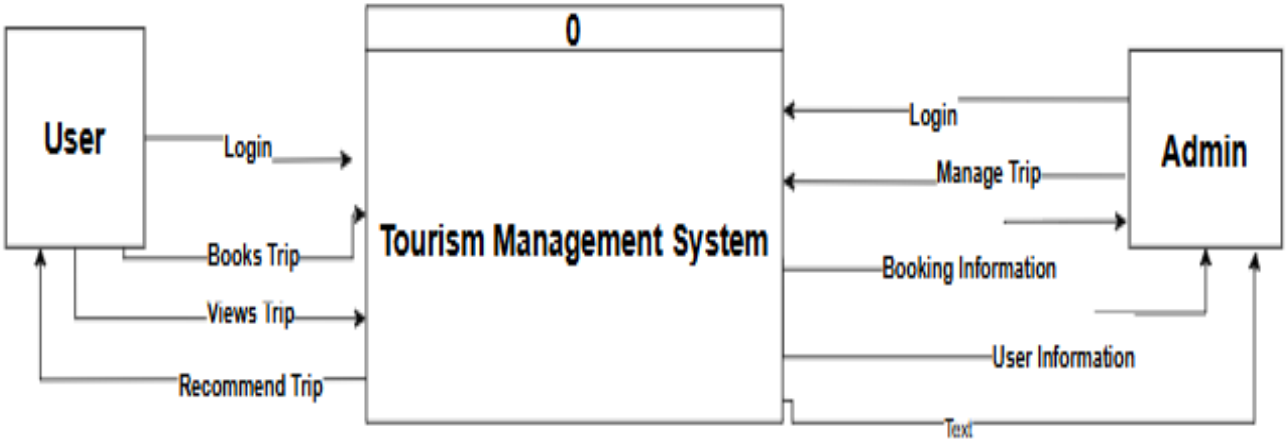


Figure 3.3: Level 0 Data Flow Diagram

Contest level DFD describes the whole system. The level 0 DFD descries all user modules who operates the system. In our system user can views detailed information about trip, book the trip and admin can manage trip, manages user's booking. Lastly there is another entity in our DFD i.e. payment which is responsible for sending the payment notification and manages the payment details done by the users.

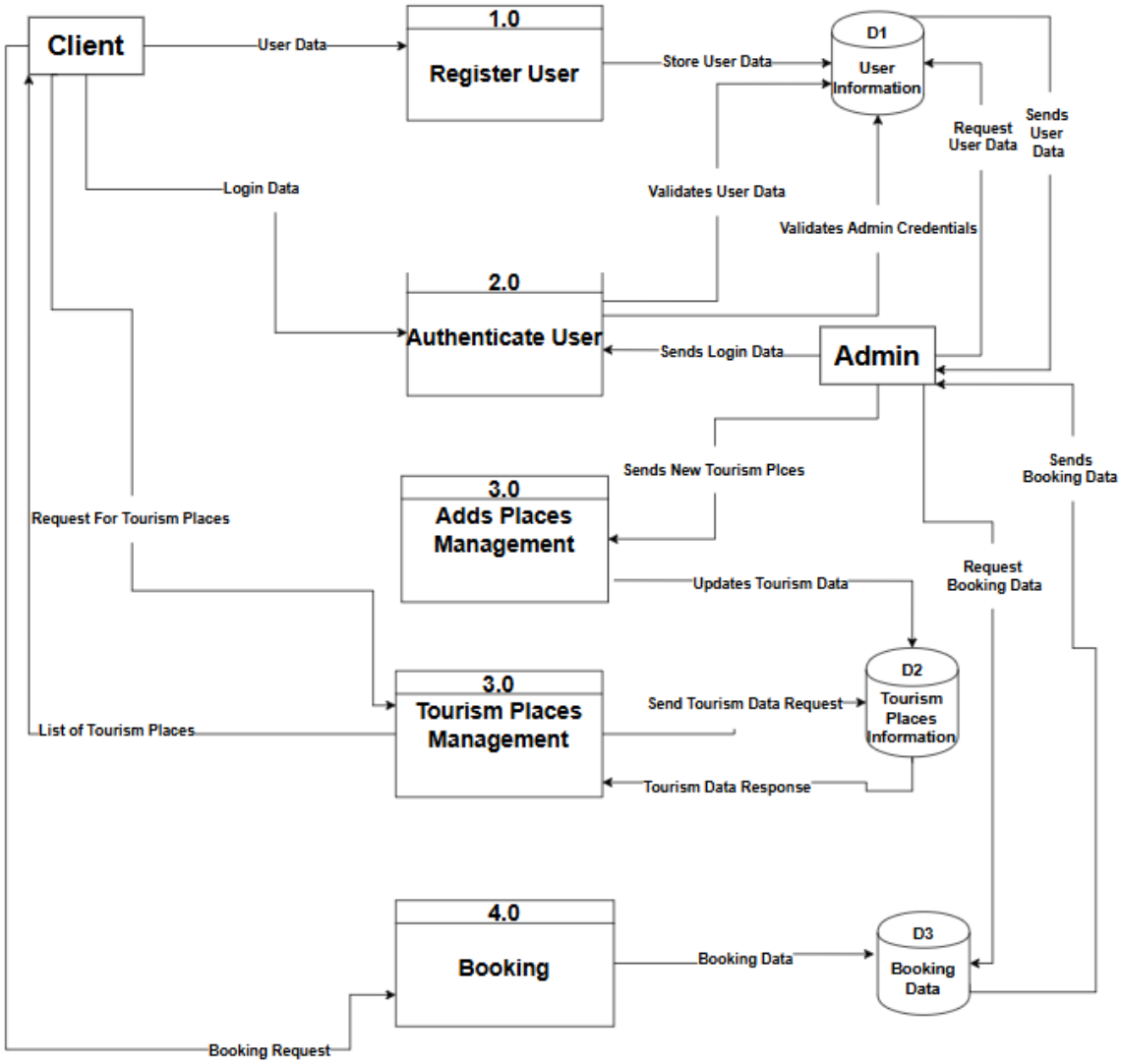


Figure 3.4: Level 1 Data Flow Diagram

The above DFD is Level 1 DFD of our system. In the above DFD there are six processes they are Registration, check for login, Tourism Places, Feedback and communication, Booking, Make payment. Here user first create account if it is not registered. After that only user will be able to login the system. If user is already registered then they simply get logged in to the system. Same process goes to the admin as well. After user login user can now check different visiting places book those places, make payment and give feedback about the trip. Admin can add different visiting places and also manages user's account. Likewise, there are three different databases we have used in our DFD i.e. booking data, payment data, user data. Where all the booked records are stored in booking data, Payment data are stored in database payment data and user's information are stored in user's data.

3.2 System Design

3.2.1 Design

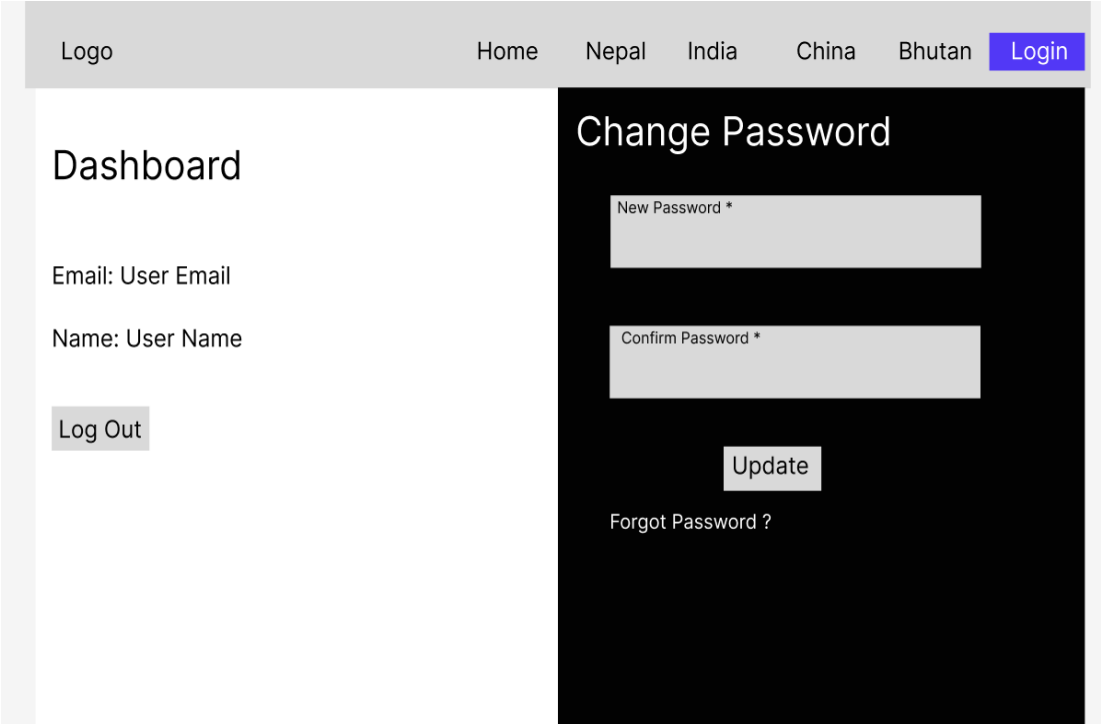
i. Database Design

Figure 3.5: Database Design

Customer entity has attributes C_id, C_name, C_email, C_password. Admin entity has A_id, A_email, A_password. Payment entity has attributes P_id, P_amount, P_status. Trip entity has attributes T_id, T_amount, T_details.

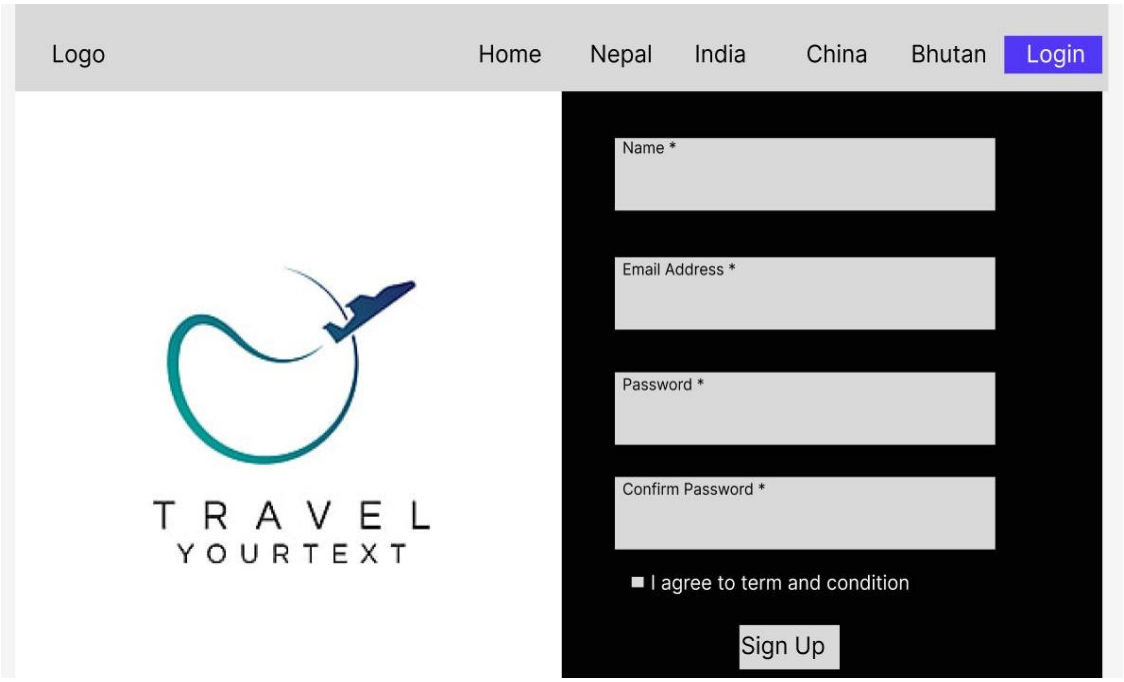
ii. Form and Report Design

Figure 3.6: User Login Form



User can login into the system using valid login credentials if the users are registered.

Figure 3.7: User Registration Form



For registeration on the system database, user can sign up by giving Name, email address, password and confirmation password. If the user has already account, then he/she can directly sign in.

iii. Interface and Dialogue

UI design, also known as user interface design, focuss almost exclusively on the look and feel of the product. UI Designers are focused on creating a cohesive, delighful product completed with animations, responsiveness and a consistent style for the product to follow.

Page

Figure 3.8: User Interface

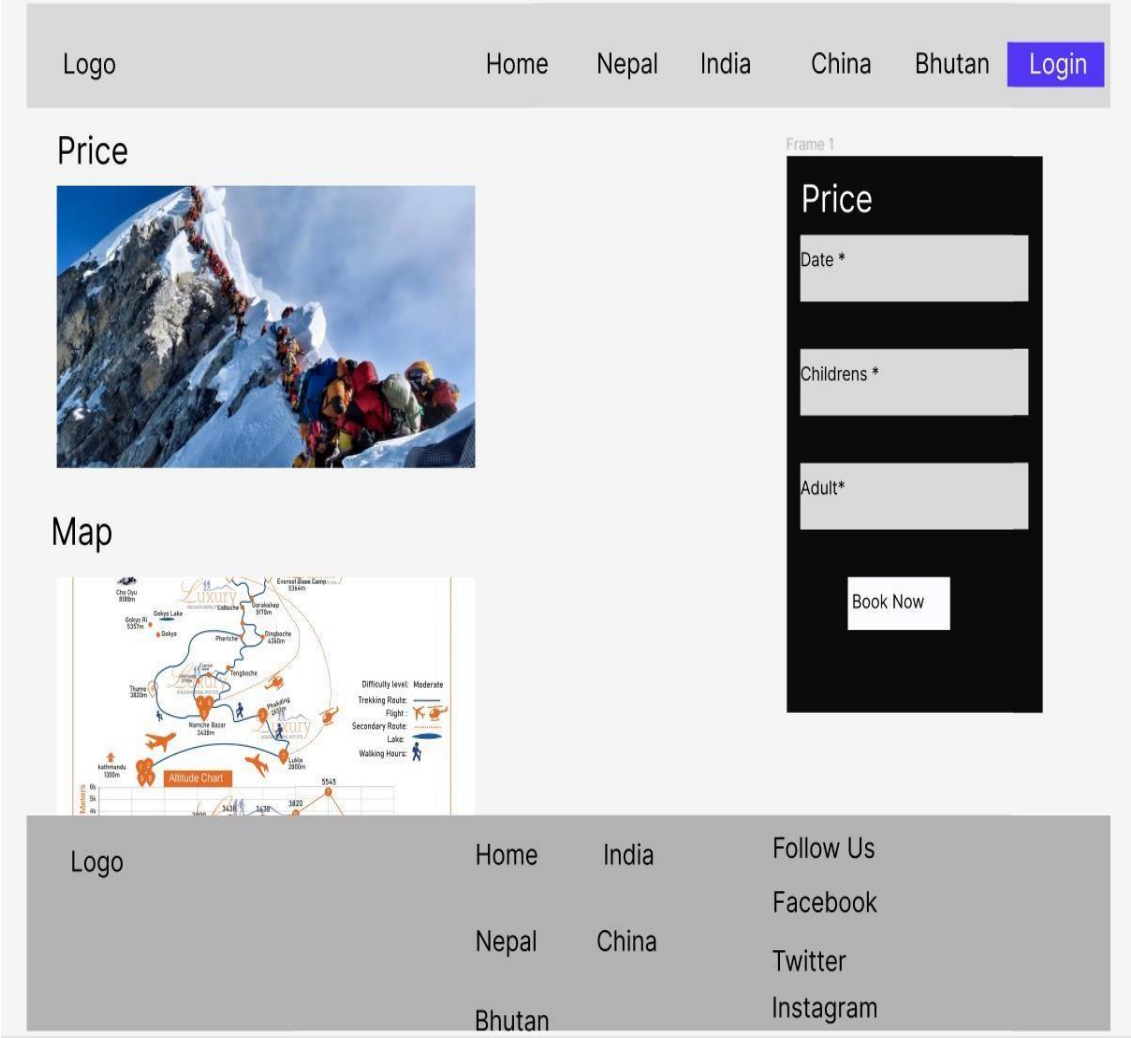
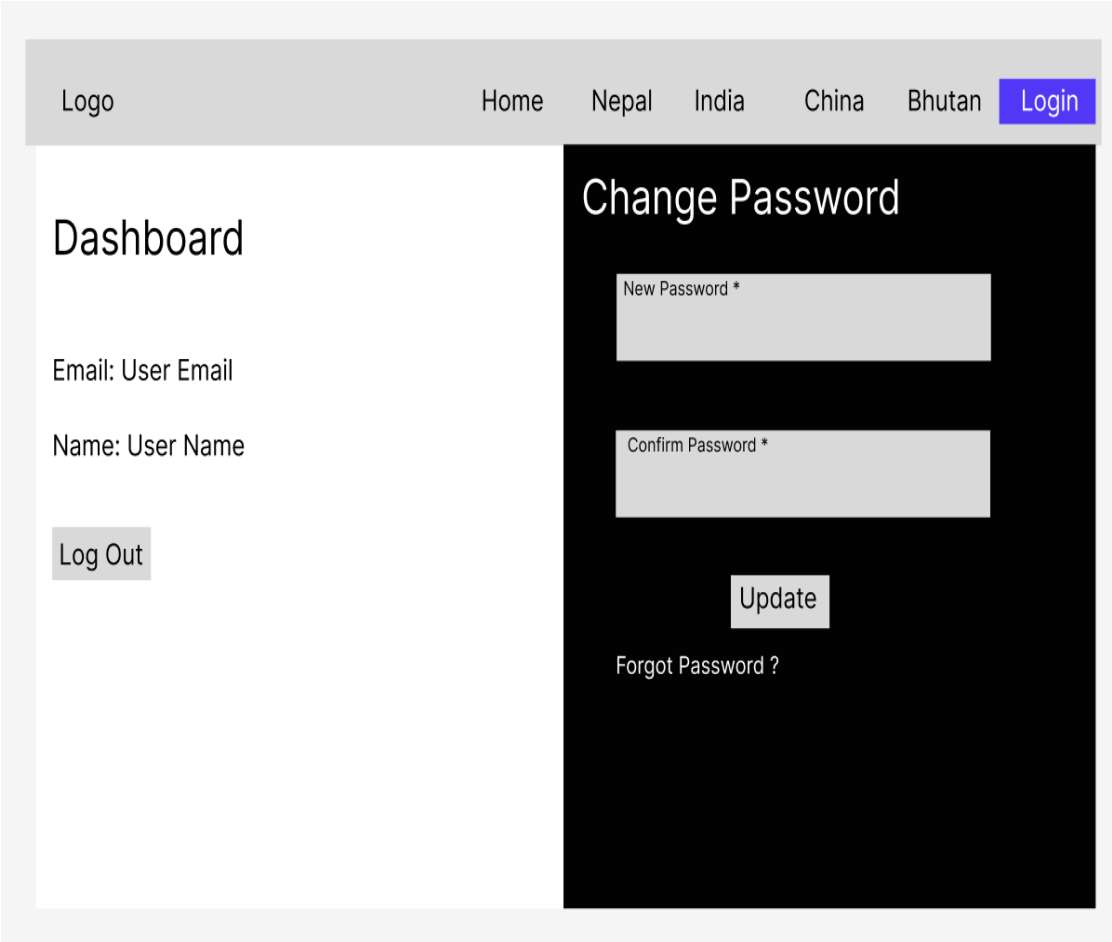


Figure 3.9: User Dashboard



3.3 Algorithm Details

FUNCTION recommend_destinations(user_preferences):
PRINT "User Preferences:" + user_preferences

# Retrieve the dataset of destinations
destination_data = get_destination_data()
PRINT "Destination Data:" + destination_data

Prepare the data for analysis
FILL missing values in 'activities' column of destination_data with an empty string

Initialize TF-IDF Vectorizer to analyze text in 'activities' column
INITIALIZE tfidf_vectorizer with stop_words='english'
CREATE tfidf_matrix by fitting and transforming the 'activities' column using tfidf_vectorizer

# Compute cosine similarity between all destinations
cosine_sim = COMPUTE cosine similarity of tfidf_matrix

# Extract user preferences
preferred_activities = user_preferences['preferred_activities']
preferred_dest_types = user_preferences['preferred_destination_types']

```
    # Filter destinations based on user preferences
    CREATE mask WHERE:
        destination_data['destination_type'] IS IN preferred_dest_types OR
        destination_data['activities'] CONTAINS ANY keyword in preferred_activities

    PRINT "Mask:" + mask

    # Get the indices of destinations that match the mask
    recommended_indices = LIST OF indices WHERE mask is True

    # Filter and sort recommended destinations by their rating in descending order
    recommended_destinations = FILTER destination_data USING recommended_indices
    SORT recommended_destinations by 'rating' in descending order

    PRINT "Recommended Destinations:" + recommended_destinations

    # Return the top recommended destinations with selected columns
    RETURN columns ['name', 'country', 'activities', 'budget_range', 'rating'] FROM
recommended_destinations.
```

Chapter 4: Implementation and Testing

4.1 Implementation

4.1.1 Tools Used

i. CASE Tools

Case tools, stands for Computer-Aided Software Engineering tools, are software applications or platforms designed to assist in various aspects of software development and engineering. These tools are used by software developers, analysts, and project managers to streamline and automate various stages of the software development life cycle (SDLC). Case tools provide support for tasks like modeling, design, documentation, and project management. Here are some common CASE tools that we have used in our project:

Design Tools

We have designed ER diagram, Data Flow Diagram, UML diagram and Flow Chart of our project. To design all these designs, we have used Creately.

Programming Tools

To do programming of our project we have to use programming tools like VS code. Which is most popular around the world and supports multiple languages. This tool is very easy to use and makes code manageable as well.

ii. Programming Language

- HTML

- HTML is the foundational language for creating web pages. It's used to structure the content of a web page using elements like headings, paragraphs, images, links, forms, and more.
- HTML provides the structure and semantics of a web page, defining the layout and hierarchy of elements.

- CSS

- CSS is used to style and format the content created with HTML. It defines how HTML elements should be displayed in terms of layout, colors, fonts, and more.
- CSS allows for the separation of content (HTML) and presentation (styling), making it easier to create visually appealing and responsive web designs.

- Bootstrap
- Bootstrap is a popular front-end framework that provides pre-designed, responsive components and CSS styles to streamline web development.
- It simplifies the process of creating consistent and mobile-friendly web pages by offering a grid system, typography, navigation bars, modals, and other UI elements
- React JS
- React is a JavaScript library for building user interfaces. It's often used to create dynamic and interactive web applications.
- React uses a component-based architecture, allowing developers to build reusable UI components that update efficiently when data changes. It's commonly used in single-page applications (SPAs).
- Django
- Django is a high-level Python web framework for building robust, secure, and maintainable web applications.
- It follows the Model-View-Controller (MVC) architectural pattern (or Model-View-Template, in Django's case) and includes features for handling databases, authentication, routing, and more.

iii. Database Platform

For the database here we have gone through Django Rest framework. Django is not a database platform itself, but rather a high-level web framework that includes a powerful Object-Relational Mapping (ORM) system. Django provides support for multiple database management systems, allowing you to choose the database platform that best suits your project's needs. So, we have chosen SQLite for our project.

SQLite: SQLite is often the default choice for small to medium-sized projects, development, and testing. It's a serverless, embedded database that is simple to set up and use.

Here is a simple demo:

```
DATABASES = {
   'default': {
      'ENGINE': 'django.db.backends.sqlite3',
```

```
        'NAME': BASE_DIR / 'db.sqlite3',
    }
}
REST_FRAMEWORK = {
    'DEFAULT_AUTHENTICATION_CLASSES': (
        'rest_framework_simplejwt.authentication.JWTAuthentication',
    )
}
```

Implementation Detail of Modules

Modules for user account creation or registration

Models

```
class UserManager(BaseUserManager):
  def create_user(self, email, name, tc, password=None, password2=None):
      """
      Creates and saves a User with the given email, name, tc and password.
      """
      if not email:
          raise ValueError('User must have an email address')
      user = self.model(
          email=self.normalize_email(email),
          name=name,
          tc=tc,
      )
      user.set_password(password)
      user.save(using=self._db)
      return user
```

```
def create_superuser(self, email, name, tc, password=None):
    """
    Creates and saves a superuser with the given email, name, tc and password.
    """
    user = self.create_user(
        email,
        password=password,
        name=name,
        tc=tc,
    )
    user.is_admin = True
    user.save(using=self._db)
    return user
```

Model refers to the representation of your data structure in your database. Models in Django are used to define the structure of database tables, including fields and their types. Models are part of Django's Object-Relational Mapping (ORM) system, which abstracts the database interaction and allows you to work with databases using Python code rather than SQL queries.

Admin

```
class UserModelAdmin(BaseUserAdmin):
  # The fields to be used in displaying the User model.
  # These override the definitions on the base UserModelAdmin
  # that reference specific fields on auth.User.
  list_display = ('id', 'email', 'name', 'tc', 'is_admin')
  list_filter = ('is_admin',)
```

```
  fieldsets = (
      ('User Credentials', {'fields': ('email', 'password')}),
      ('Personal info', {'fields': ('name', 'tc')}),
      ('Permissions', {'fields': ('is_admin',)}),
  )
  # add_fieldsets is not a standard ModelAdmin attribute. UserModelAdmin
  # overrides get_fieldsets to use this attribute when creating a user.
  add_fieldsets = (
      (None, {
          'classes': ('wide',),
          'fields': ('email', 'name', 'tc', 'password1', 'password2'),
      }),
  )
  search_fields = ('email',)
  ordering = ('email', 'id')
  filter_horizontal = ()
```

The admin interface is a powerful tool that allows you to manage and interact with your application's data without writing custom views or forms. The admin interface is automatically generated based on your defined models.

Serealizer

```
class UserRegistrationSerializer(serializers.ModelSerializer):
  # We are writing this becoz we need confirm password field in our Registratin Request
  password2 = serializers.CharField(style={'input_type':'password'}, write_only=True)
  class Meta:
```

```
    model = User
    fields=['email', 'name', 'password', 'password2', 'tc']
    extra_kwargs={
      'password':{'write_only':True}
    }
  # Validating Password and Confirm Password while Registration
  def validate(self, attrs):
    password = attrs.get('password')
    password2 = attrs.get('password2')
    if password != password2:
      raise serializers.ValidationError("Password and Confirm Password doesn't match")
    return attrs
  def create(self, validate_data):
    return User.objects.create_user(**validate_data)
```

Serializers are used to convert complex data types, such as query set and model instances, into native Python data types that can be easily rendered into JSON or other content types. Serializers play a crucial role in handling data representation and parsing in Django REST Framework (DRF), which is commonly used for building RESTful APIs.

Views

```
class UserRegistrationView(APIView):
  renderer_classes =  [UserRenderer]
  def post(self, request, format=None):
    serializer = UserRegistrationSerializer(data=request.data)
    serializer.is_valid(raise_exception=True)
    user = serializer.save()
```

token = get_tokens_for_user(user)

return Response({'token':token, 'msg':'Registration Successful'}, status=status.HTTP_201_CREATED)

Views play a crucial role in handling HTTP requests and generating HTTP responses. Views are responsible for processing user requests, interacting with the database or other data sources, and rendering the appropriate response.

Urls

rom django.urls import path

from account.views import SendPasswordResetEmailView, UserChangePasswordView, UserLoginView, UserProfileView, UserRegistrationView, UserPasswordResetView,bookingList

urlpatterns = [

path('register/', UserRegistrationView.as_view(), name='register'),

The url.py file is used to define URL patterns for your web application. It serves as a routing mechanism that maps URLs to corresponding views, allowing Django to determine which view should handle a particular request.

From Frontend

Connecting to base URL and endpoints

baseQuery: fetchBaseQuery({ baseUrl: 'http://127.0.0.1:8000/api/user/' }),

endpoints: (builder) => ({

registerUser: builder.mutation({

query: (user) => {

return {

url: 'register/',

method: 'POST',

body: user,

```
        headers: {
            'Content-type': 'application/json',
        }
    }),
```

4.2 Testing

4.2.1 Test case for unit testing

i. Unit testing for backend

Table 4.1 Test Cases for backend

Test ID	Test Scenario Description	Test Case Description	Test Steps	Expected Result	Actual result	Remarks	Status
1	Admin authentication with invalid credentials	Should not allow user to authenticate with invalid details	Details enter: Username: admin@example.com Password: 123	Should show error message	Incorrect details	We cannot login to our system	Pass
2	Admin authentication with invalid credentials	Should allow user to authenticate with invalid details	Details enter: Username: admin@example.com Password: 12345	Should show error message	Correct details	Successfully logged in to our system	Pass

ii. Unit testing for frontend

Table 4.2 Test Cases for frontend

Test ID	Test Scenario Description	Test Case Description	Test Steps	Expected Result	Actual result	Remarks	Status
1	Registration testing	Creating a new user	Create user with email: abhaydhkal@gmail.com password: 12345 Confirmation password: 12345	Should create a new user	New user created	Successfully created new user	Pass
2	Login testing	Login using stored username and password	Details enter: Username: abhaydhkal@gmail.com Password: 12345	Allow user to login	Successfully logs in	User successfully logs in	Pass
3	Logout Testing	Logout using logout button	Clicking logout button	Tokens get deleted and user logged out	Tokens get deleted and user logged out	Stored token in the local storage and tokens are deleted and user logged out	Pass

3	Recommen-dation check when no user preference given	Should not give any recommen-dation	Data entered: No data given	Should give no recemmendation available message	No data	No recommendation available	Pass
4	Recommen-dation check when user preference given	Should now give any recommen-dation	Data entered: User Preferences like destination type and many more	Should give recemmendation	Correct Data	Should recommend data	Pass

4.2.2 Test case for system testing

We will use it to test the correct functioning of the systems. Suppose a new user is created with the required credentials, then the user should be able to book the trip along with viewing the trip. If the user is not registered then user cannot view the trip as well as cannot book the trip as well. Likewise, user can change password also from their dashboard. Similarly, admin can change the content, manage user's account after login. This makes the whole system dynamic.

For recommendation system testing we will not give any user preferences where the system will not recommend any destinations. When we give the user preferences to the system the system will recommend the system according to the user preferences.

Chapter 5: Conclusion and Future Recommendation

5.1 Conclusion

The system, Blissful Holiday – is web portal was successfully built with implementation of Django and reactjs. Using this system, client would be able to explore different places to visit. They can also see the whole information of the trip including the price of the trip. Similarly, client can also book their trip according to their requirement.

Key achievements include a secure authentication system. However, the integration complexity between Django and React presented challenges, demanding meticulous attention to data synchronization and communication.

Looking forward, potential enhancements could include the development of a mobile application for added convenience and the incorporation of advanced analytics to glean insights into user preferences and travel trends. Despite challenges, the Tours and Travel Management System represents a successful synergy of technologies, a testament to the effectiveness of the Django and React stack in meeting the demands of the modern traveler and laying a solid foundation for future growth and innovation.

5.2 Recommendation

A lot of features can be added in our system as well. The User Interface can be made more user friendly in the first place. Our system can provide more features like video of the particular place. Booking system can be more precise and more functionality can be added in booking system. More precise responsiveness can be done. Similarly, the admin panel can also be made more professional.

Different algorithm can be used for the better version of the system example searching algorithm. System can be made more dynamic. The system allows user to add and delete the content about the trip. But it can be made more dynamic by allowing admin to add and delete the trips according to their need.

The authentication, and authorization process of the admin panel is not very accurate. As the system allow to create the superuser from the terminal except from the admin login page. So, the system can be improved in case of admin registration process. Which will make the system more secure and safe.

References

[1] S. K. Nepal, "Tourism in protected areas: the Nepalese Himalaya," *Annals of Tourism Research 27.3 (2000)*.

[2] "krasmo," [Online]. Available: https://www.krasamo.com/agile-development-process/.

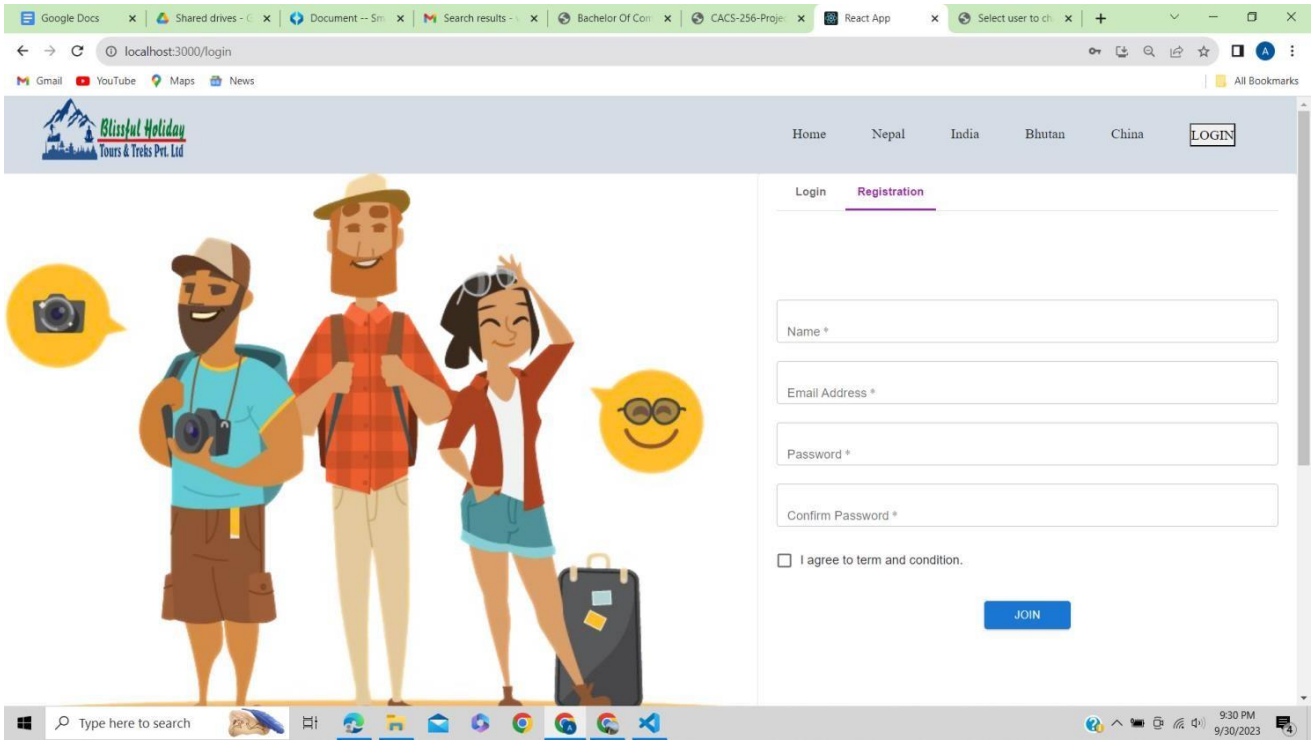
[3] T. Wenan, . D. Shrestha, D. Shrestha, N. Rajkarnikar and S. R. Jeong, "The Role of EmergingTechnologies in Digital Tourism Business Ecosystem Model for Nepal," *Cham: Springer International Publishing,* vol. 569, pp. 1123-1137, 2022.

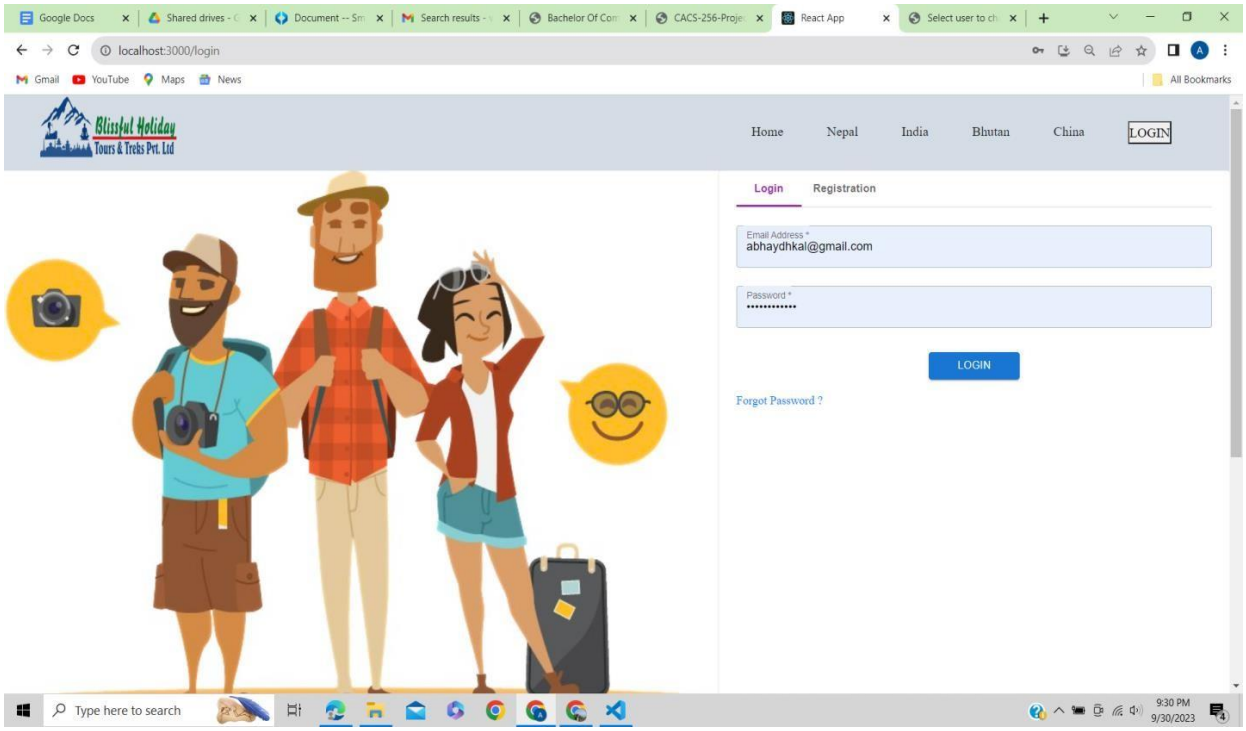
[4] H. P. Shrestha and P. Shrestha, "Tourism in Nepal: A Historical Perspective and Present Trend of Development.," *Himalayan Journal of Sociology & Anthropology,* vol. 5, no. 1, pp.54-75, 2012.

[5] W.-C. Chiou, . L. Chin-Chao and C. Perng, "A strategic website evaluation of online travelagencies," *Tourism Management 32.6,* no. 3, pp. 1463-1473., 2011.

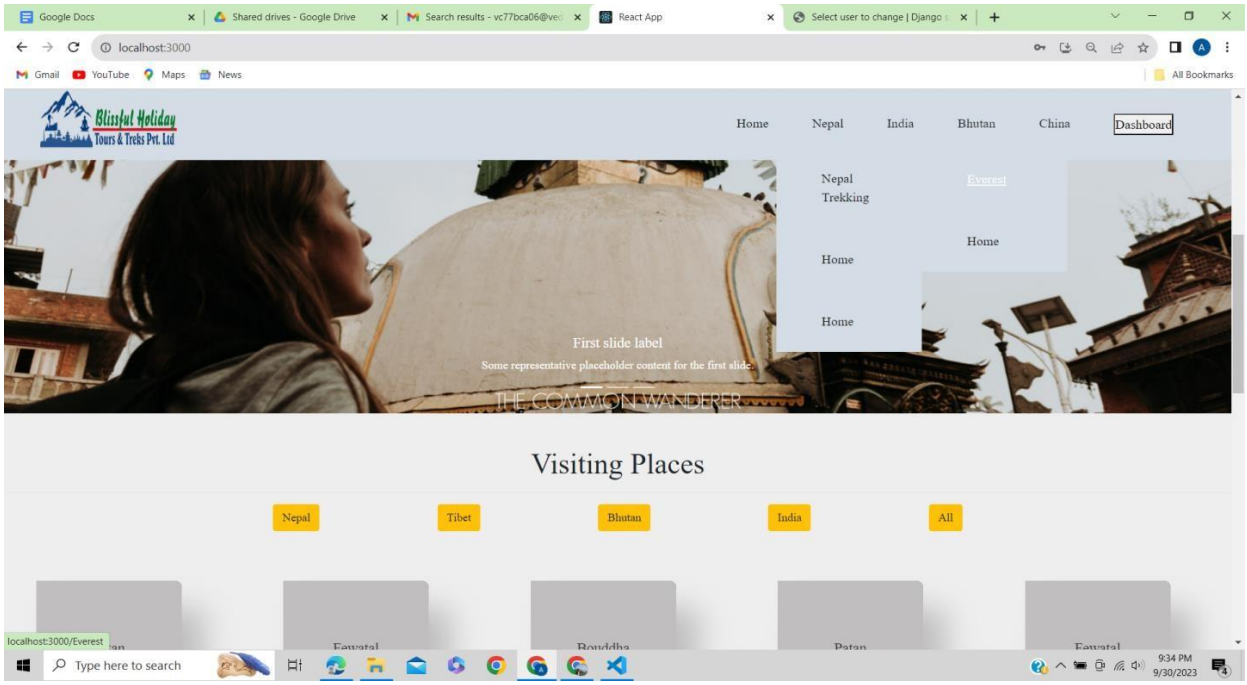
Appendix

From and Registration design

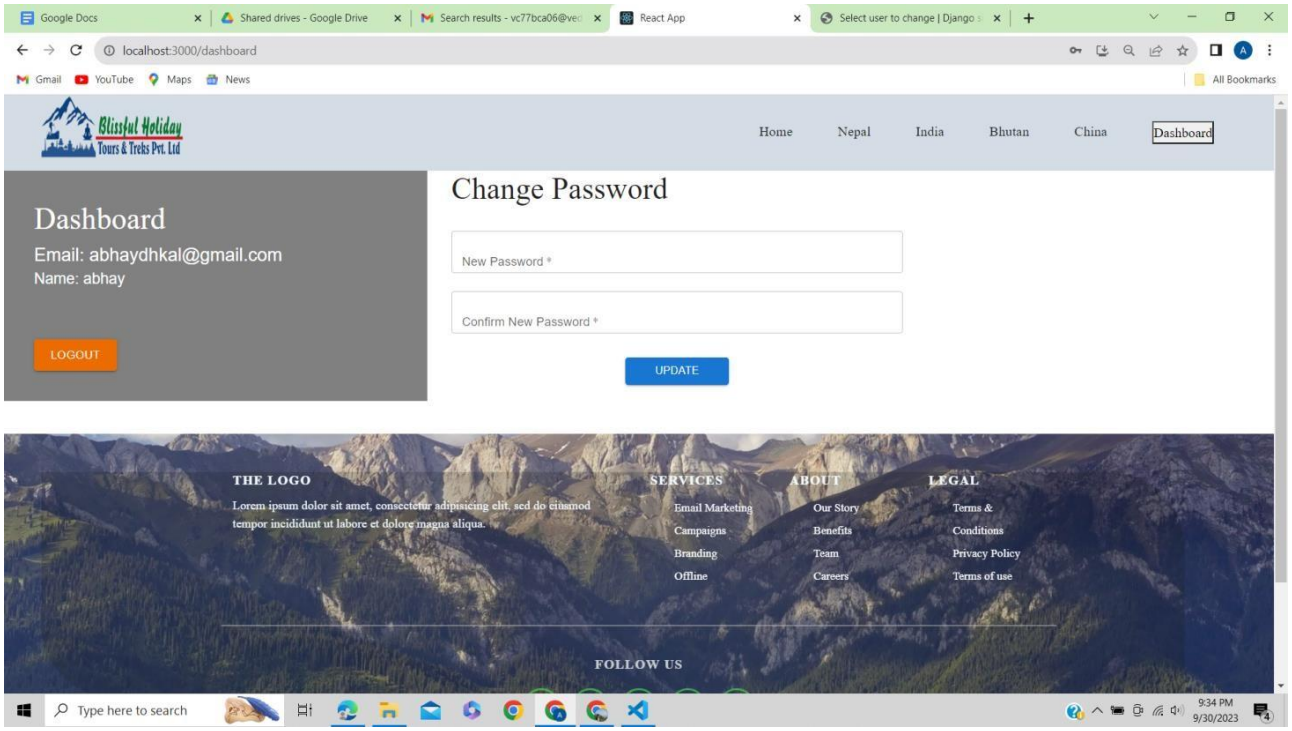




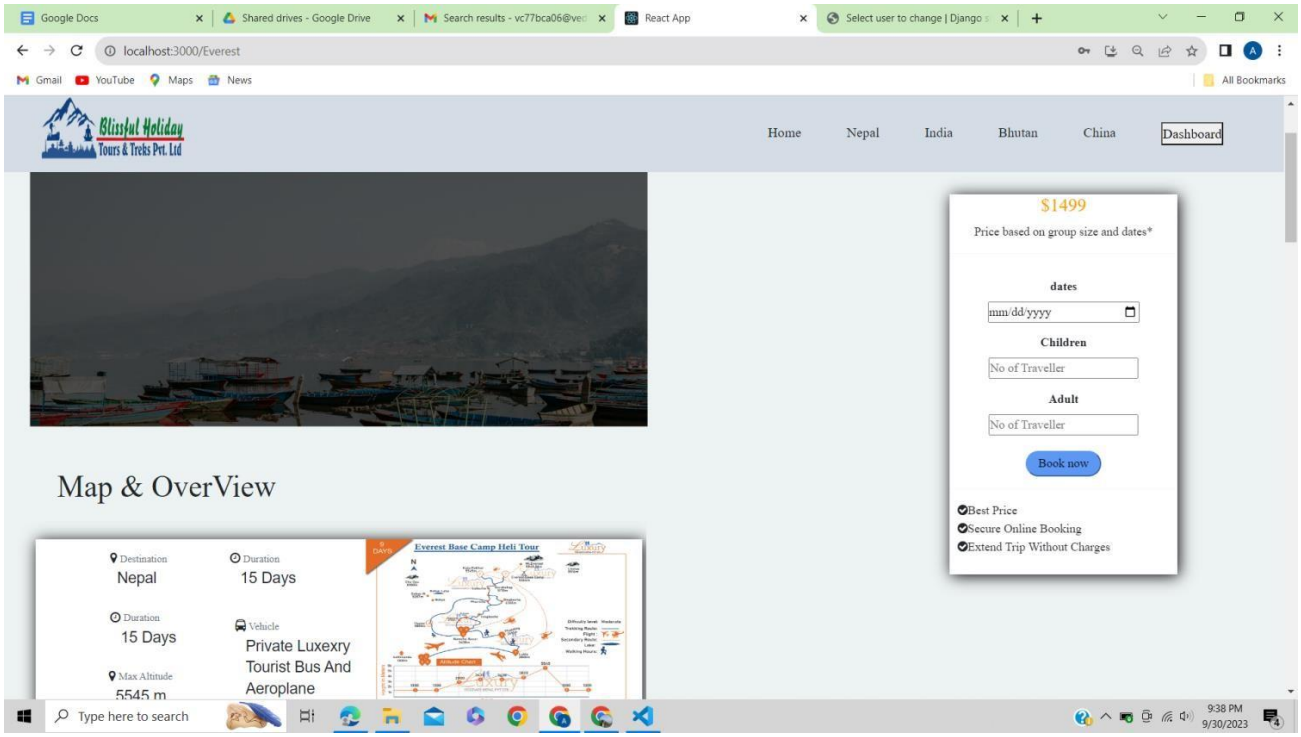
Home Page


Google Docs
Shared drives - Google Drive
Search results - vc77bca06@ve
React App
Select user to change | Django
localhost:3000
Gmail
YouTube
Maps
News
All Bookmarks
Blissful Holiday
Tours & Treks Pvt. Ltd
Home
Nepal
India
Bhutan
China
Dashboard
Nepal Trekking
Everest
Home
Home
Home
First slide label
Some representative placeholder content for the first slide.
THE COMMON WANDERER
Visiting Places
Nepal
Tibet
Bhutan
India
All
Patan
localhost:3000/Everest
Type here to search
9:34 PM
9/30/2023

User Dashboard



Sub Page



Admin Dashboard