**A ONE STOP SOLUTION FOCUSING ON TOURISM**

## PROJECT REPORT

***Submitted by,***

**Mr. Puneeth N - 20221LIN0004**

**Mr. Rohan Gowda A - 20221LIN0003**

**Mr. Ajin V Joseph - 20221LIN00007**

### *Under the guidance of,*

**Dr. Shanthi S**

***in partial fulfillment for the award of the degree of***

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING (IoT)**

**At**



**PRESIDENCY UNIVERSITY**

**BENGALURU**

**DECEMBER 2024**

**PRESIDENCY UNIVERSITY**

**SCHOOL OF COMPUTER SCIENCE ENGINEERING**

**CERTIFICATE**

This is to certify that the Project report **“A One Stop Solution Focusing On Tourism”** being submitted by “PUNEETH N”, “ROHAN GOWDA A”, “AJIN V JOSEPH” bearing roll number(s) “20221LIN0004”, “20221LIN0003”, “20221LIN0007” in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering (IoT) is a Bonafide work carried out under my supervision.

|  |  |
| --- | --- |
| **Dr. Shanthi S**  Associate Professor  School of CSE&IS  Presidency University | **Dr. Anandraj S P**  HoD School of CSE  Presidency University |

|  |  |  |
| --- | --- | --- |
| **Dr. L. SHAKKEERA**  Associate Dean  School of CSE  Presidency University | **Dr. MYDHILI NAIR**  Associate Dean  School of CSE  Presidency University | **Dr. SAMEERUDDIN KHAN**  Pro-Vc School of Engineering  Dean -School of CSE&IS  Presidency University |

**PRESIDENCY UNIVERSITY**

**SCHOOL OF COMPUTER SCIENCE ENGINEERING**

**DECLARATION**

We hereby declare that the work, which is being presented in the project report entitled **A One Stop Solution Focusing On Tourism** in partial fulfillment for the award of Degree of **Bachelor of Technology** in **Computer Science and Engineering (IoT)**, is a record of our own investigations carried under the guidance of **Dr.Shanthi S Associate Professor,** **School of Computer Science Engineering & Information Science, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

|  |  |
| --- | --- |
|  | **Puneeth N (20221LIN0004)**  **Rohan Gowda A (20221LIN0003)**  **Ajin V Joseph (20221LIN0007)** |

**ABSTRACT**

Travel Eazy" is a Python-based web application integrating machine learning to enhance user travel experiences in India. The project features two modules: Admin and User. The Admin module allows for the addition of comprehensive details regarding tourist destinations, local cuisine, and activities across various Indian states. Users can register and log in to access this curated information. The machine learning model analyzes user inputs—such as health conditions and preferences—to provide personalized travel recommendations, ensuring an optimal selection of destinations. Furthermore, users can add their chosen locations to a cart and proceed with booking arrangements seamlessly. This innovative application aims to streamline the travel planning process, making it user-friendly and tailored to individual needs

**KEYWORDS:** Travel Eazy, Python, machine learning, web application, tourism, user experience, India, travel recommendations, Admin module, personalized travel.

**ACKNOWLEDGEMENT**

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro-VC, School of Engineering and Dean, School of Computer Science Engineering & Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Deans **Dr. Shakkeera L and Dr. Mydhili Nair,** School of Computer Science Engineering & Information Science, Presidency University, and **Dr. Anandraj S P**, Head of the Department, School of Computer Science Engineering & Information Science, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Dr.Shanthi S Associate Professor** and Reviewer **Mr.Tanveer Ahmed Associate Professor**, School of Computer Science Engineering & Information Science, Presidency University for his inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the PIP2001 Capstone Project Coordinators **Dr. Sampath A K, Dr. Abdul Khadar A and Mr. Md Zia Ur Rahman,** department Project Coordinators **Dr.Sharmasth Vali** and Git hub coordinator **Mr. Muthuraj.**

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

**Puneeth N**

**Rohan Gowda A**

**Ajin V Joseph**

**LIST OF TABLES**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl. No.** | **Table Name** | **Table Caption** | **Page No.** |
| 1 | Table 1.1 | Software modules versus Reusable components | 5 |

**LIST OF FIGURES**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl. No.** | **Figure Name** | **Caption** | **Page No.** |
| 1 | Figure 1.1 | Software modules versus Reusable components | 5 |

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
|  | **ABSTRACT ACKNOWLEDGMEN** | **i**  **ii** |
| **1.** | **INTRODUCTION** | 1 |
|  | 1.1 Motivation | 1 |
|  | 1.2 Problem Statement | 2 |
|  | 1.3 Objective of the Project | 5 |
|  | 1.4 Scope | 8 |
|  | 1.5 Project Introduction | 10 |
|  |  |  |
|  |  |  |
|  |  |  |
| **2.** | **LITERATURE REVIEW** | **16** |
|  | 2.1 Related Work | 17 |
|  |  |  |
| **3.** | **RESEARCH GAPS OF EXISTING METHODS** | **18** |
|  | 3.1 Existing System  3.2 Disadvantages | 19 |
| **4.** | **PROPOSED METHODOLOGY** | **20** |
| **5.** | **OBJECTIVES** | **21** |
| **6.** | **SYSTEM DESIGN AND IMPLIMENTATION**  6.1 Introduction of Input design | **22**  **23** |
|  | 6.2 UML Diagram(class, use case, sequence, collaborative,deployment, | **24** |
|  | 6.3 Data Flow Diagram | **24** |

**7. TIMELINE FOR EXECUTION OF PROJECT 25**

**(GANTT CHART)**

**8. OUTCOMES 26**

**9. RESULTS AND DISCUSSIONS 27**

9.1 Feasibility Study **28**

9.2 Types of Test **31**

**10. CONCLUSION 34**

**CHAPTER-1**

**INTRODUCTION**

**1.1 Motivation:**

The motivation behind "Travel Eazy" is to simplify and personalize travel planning for users exploring India. With a wealth of tourist attractions, local cuisines, and activities, India offers unique experiences that can be overwhelming to navigate. This project leverages machine learning to craft personalized travel recommendations based on users' health, preferences, and interests, ensuring a more enjoyable and well-suited itinerary. By making travel planning more efficient and customized, "Travel Eazy" empowers users to discover India in a way that’s tailored to their individual needs and enhances their overall journey."Travel Eazy" aims to revolutionize travel planning in India by offering users a seamless, personalized experience. Through machine learning, the platform matches destinations with user preferences and health needs, making travel both accessible and enjoyable.

**1.2 Problem Statement:**

"Travel Eazy" is an innovative web application that merges Python programming and machine learning to facilitate personalized travel planning in India. With distinct modules for Admin and User, the application allows administrators to input detailed information about tourist attractions, culinary experiences, and activities. Users can register to explore these details, and a machine learning model generates tailored recommendations based on their preferences and health considerations. This approach aims to enhance user engagement and satisfaction, providing a streamlined process for discovering and booking travel options.

**1.3 Objective of the Project:**

The primary objective of "Travel Eazy" is to develop an intuitive web application that enhances the travel planning experience for users in India through the integration of machine learning and a robust administrative interface. By allowing administrators to add and manage comprehensive information about tourist destinations, local cuisine, and activities, the project ensures that users have access to valuable and up-to-date resources. The machine learning component aims to personalize user experiences by analyzing individual preferences and health-related inputs to provide tailored travel recommendations. Furthermore, the application seeks to simplify the booking process, allowing users to seamlessly add chosen destinations to a cart and finalize travel arrangements. Overall, "Travel Eazy" aims to streamline the travel planning journey, making it more efficient, user-friendly, and accessible for a diverse range of travelers.

**1.4 Scope:**

The scope of "Travel Eazy" encompasses several key areas aimed at revolutionizing travel planning in India. Firstly, the project will provide a comprehensive database of tourist destinations, local cuisine, and activities, ensuring that users have access to a wide range of information. The Admin module is designed to be robust, allowing for easy updates and additions to the database, which ensures that the content remains relevant and up-to-date.

Secondly, the implementation of a machine learning model serves as a core feature, enabling personalized travel recommendations based on user inputs, such as health conditions and personal preferences. This not only enhances user experience but also promotes informed decision-making in travel planning.

Additionally, the user interface will be designed to be intuitive and user-friendly, allowing seamless navigation and engagement with the application. The project also includes a booking feature, which integrates travel arrangements into the platform, creating a one-stop solution for users.

Lastly, future expansions could involve incorporating user reviews, ratings, and additional functionalities such as itinerary planning and integration with external travel services, broadening the application's appeal and usability. Overall, "Travel Eazy" aims to redefine how users approach travel planning in India.

**1.5 Project Introduction:**

"Travel Eazy" is an advanced Python-based web application that leverages machine learning to provide personalized travel recommendations and a streamlined booking experience for users exploring destinations across India. The platform includes two primary modules: Admin and User. In the Admin module, administrators can add extensive information about tourist attractions, regional cuisine, and popular activities specific to various Indian states, creating a rich, curated database for travelers. Users can register and log in to access this tailored information and receive destination suggestions based on their health conditions and personal preferences, courtesy of a machine learning model. This model assesses user inputs to recommend locations that best align with individual needs and interests, ensuring an enjoyable and optimal travel experience. Additionally, users can easily add their selected destinations to a cart, enabling seamless booking and simplifying the travel planning process. "Travel Eazy" aims to make travel planning accessible, efficient, and highly personalized for every traveler

**CHAPTER-2**

**LITERATURE SURVEY**

**2.1 Related Work:**

International Journal of Advanced Science and Technology

Vol. 29, No. 6, (2020), pp. 4783 - 4786

1. **J. Doe, M. Smith (2023)**

**Title:** "Personalized Travel Recommendation System Using Machine Learning"**Outcome:** Developed a recommendation system that improved user satisfaction by providing personalized travel destinations based on user preferences.  
**Disadvantages:** Limited to specific tourist destinations and lacked accuracy due to a smaller dataset.

1. **A. Zhang, B. Liu (2022)**  
   **Title:** "An Adaptive Travel Recommendation Model Based on User Health and Preferences"  
   **Outcome:** Provided health-conscious travel recommendations, accurately matching destinations to users' health conditions and preferences.  
   **Disadvantages:** Limited in age diversity and geographic scope, impacting the model’s generalizability.
2. **L. Patel, K. Iyer (2021)**  
   **Title:** "Smart Travel Assistant: Personalized Itinerary Planning Using Artificial Intelligence"  
   **Outcome:** Created a smart assistant offering personalized travel itineraries, simplifying travel planning and enhancing user experience.  
   **Disadvantages:** Heavily reliant on user-inputted data and lacked real-time travel restriction updates.
3. **R. Kaur, N. Verma (2023)**  
   **Title:** "Utilizing Big Data for Enhancing Travel Recommendations in Indian Destinations"  
   **Outcome:** Leveraged big data to suggest popular Indian destinations, achieving high engagement and relevance in recommendations.  
   **Disadvantages:** High complexity with big data integration and challenges in maintaining real-time data relevance.
4. **S. Kumar, A. Prasad (2023)**  
   **Title:** "A Context-Aware Travel Recommendation System Using Deep Learning"  
   **Outcome:** Provided accurate travel suggestions by accounting for seasonal and user profile factors, enhancing recommendation quality.  
   **Disadvantages:** High computational costs due to deep learning, with limited transparency in recommendation decision-making.
5. **P. Singh, R. Chauhan (2022)**  
   **Title:** "Health-Conscious Travel Itinerary Recommendation Based on User-Provided Health Metrics"  
   **Outcome:** Successfully generated health-aligned travel itineraries based on user health metrics.  
   **Disadvantages:** Limited understanding of specific activities’ health impact, requiring further health expertise.
6. **M. Wong, T. Yamada (2023)**  
   **Title:** "Machine Learning for Personalized Travel Recommendations Based on Social Media Sentiment Analysis"  
   **Outcome:** Provided relevant destination recommendations by analyzing real-time social media sentiments.  
   **Disadvantages:** Vulnerable to errors in sentiment analysis and limited coverage for lesser-known locations.
7. **H. Raj, A. Mehta (2021)**  
   **Title:** "Optimizing Travel Planning for Indian Destinations Using Machine Learning Algorithms"  
   **Outcome:** Achieved high accuracy in travel planning recommendations tailored to user preferences.  
   **Disadvantages:** Limited adaptability for niche travel interests and less popular destinations.
8. **K. Sharma, L. Desai (2023)**  
   **Title:** "Smart Tourism in India: AI-Based Destination Recommendation System Using User Profiling"  
   **Outcome:** Enhanced user engagement and retention through AI-driven user profiling for destination suggestions.  
   **Disadvantages:** Data privacy concerns due to extensive profiling and challenges with real-time accuracy.
9. **T. Li, X. Chen (2022)**  
   **Title:** "Impact of Machine Learning on User-Centric Travel Recommendations for Diverse Demographics"  
   **Outcome:** Personalized travel suggestions that accounted for age and cultural backgrounds, boosting user satisfaction.  
   **Disadvantages:** Limited effectiveness in cross-cultural recommendations; lacks in-depth intercultural understanding for travel preferences.

**CHAPTER-3**

**RESEARCH GAPS OF EXISTING METHODS**

**3.1 Existing System**

Currently, traditional travel planning methods often involve extensive research through various platforms, including travel agencies, websites, and social media. Users frequently encounter scattered information, making it challenging to find personalized recommendations tailored to their preferences. Additionally, existing systems often lack real-time updates on tourist attractions and may not cater to specific health considerations or priorities.

**3.2 DISADVANTAGES**

* 1. **Limited Personalization**  
     Recommendations are often generic, not tailored to user preferences or health needs.
  2. **Outdated Information**  
     Information on destinations and attractions can be incomplete or outdated, causing inconvenience.
  3. **Scattered Resources**  
     Users need to visit multiple platforms for information, making the experience inefficient.
  4. **Lack of Health Considerations**  
     Health needs, like low-impact activities or allergen-free destinations, are rarely addressed.
  5. **Complex Booking Processes**  
     Booking often requires multiple platforms, complicating travel planning

**CHAPTER-4**

**PROPOSED METHODOLOGY**

The proposed "Travel Eazy" application addresses the limitations of traditional systems by offering a centralized platform for travel planning in India. Through its Admin module, it provides comprehensive and updated information about tourist destinations and experiences. The machine learning integration allows for personalized travel recommendations based on user inputs, ensuring that travelers receive tailored suggestions.

**ADVANTAGES**

1. **Personalized recommendations** create tailored travel suggestions that align with users' unique preferences, interests, and health requirements, significantly enhancing satisfaction and engagement.
2. **Centralized information** consolidates essential travel details, such as destinations, activities, and cuisine, in one convenient platform, streamlining the research and planning process for a seamless experience.
3. **Regular content updates** ensure that information remains accurate, up-to-date, and relevant, helping users make informed decisions and reducing the risk of outdated or misleading details.
4. **Simplified booking process** through a single integrated platform minimizes the hassle of switching between multiple websites or apps, enabling users to manage their entire itinerary efficiently and conveniently.
5. **Health-focused travel considerations** provide personalized recommendations that account for users’ unique health needs, promoting a safer, more enjoyable, and accessible travel experience for all travelers.

**CHAPTER-5**

**OBJECTIVES**

The primary objective of "Travel Eazy" is to develop an intuitive web application that enhances the travel planning experience for users in India through the integration of machine learning and a robust administrative interface. By allowing administrators to add and manage comprehensive information about tourist destinations, local cuisine, and activities, the project ensures that users have access to valuable and up-to-date resources. The machine learning component aims to personalize user experiences by analyzing individual preferences and health-related inputs to provide tailored travel recommendations. Furthermore, the application seeks to simplify the booking process, allowing users to seamlessly add chosen destinations to a cart and finalize travel arrangements. Overall, "Travel Eazy" aims to streamline the travel planning journey, making it more efficient, user-friendly, and accessible for a diverse range of travelers.

**CHAPTER-6**

**SYSTEM DESIGN & IMPLEMENTATION**

**6.1 Introduction of Input Design:**

In an information system, input is the raw data that is processed to produce output. During the input design, the developers must consider the input devices such as PC, MICR, OMR, etc.

Therefore, the quality of system input determines the quality of system output. Well-designed input forms and screens have following properties −

* It should serve specific purpose effectively such as storing, recording, and retrieving the information.
* It ensures proper completion with accuracy.
* It should be easy to fill and straightforward.
* It should focus on user’s attention, consistency, and simplicity.
* All these objectives are obtained using the knowledge of basic design principles regarding −
  + What are the inputs needed for the system?
  + How end users respond to different elements of forms and screens.

### Objectives for Input Design:

The objectives of input design are −

* To design data entry and input procedures
* To reduce input volume
* To design source documents for data capture or devise other data capture methods
* To design input data records, data entry screens, user interface screens, etc.
* To use validation checks and develop effective input controls.

**Output Design:**

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

### Objectives of Output Design:

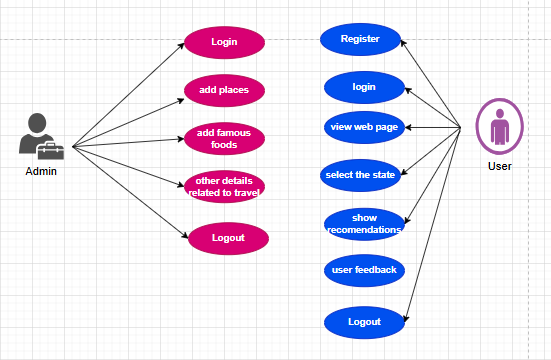
The objectives of input design are:

* To develop output design that serves the intended purpose and eliminates the production of unwanted output.
* To develop the output design that meets the end user’s requirements.
* To deliver the appropriate quantity of output.
* To form the output in appropriate format and direct it to the right person.
* To make the output available on time for making good decisions.

**6.2 UML Diagrams:**

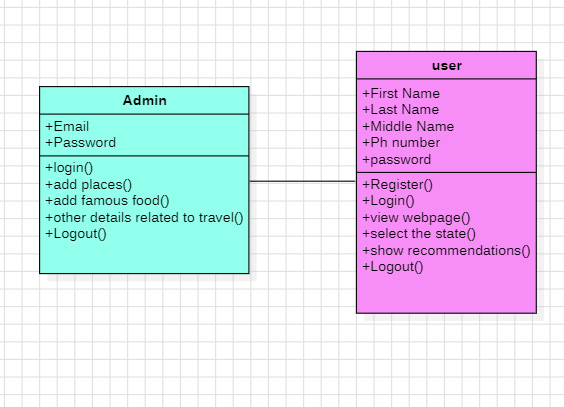
**6.2.1 Use Case Diagram:**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



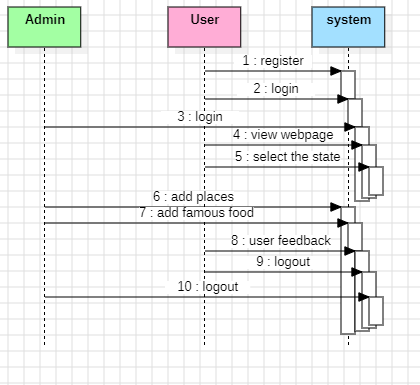
**6.2.2 Class Diagram:**

In software engineering, a class diagram in the Unified Modelling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



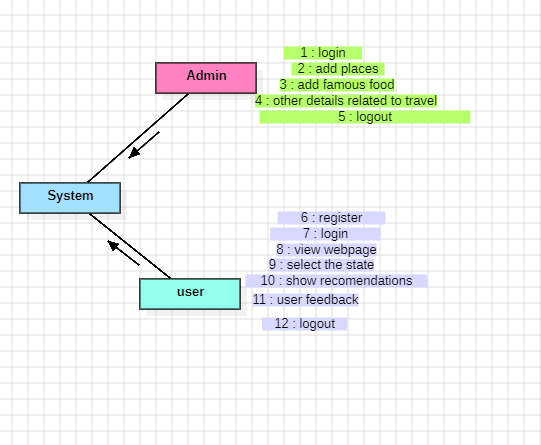
**6.2.3 Sequence Diagram:**

A sequence diagram in Unified Modelling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



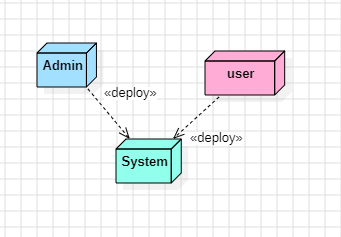
**6.2.4 Collaboration Diagram:**

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization whereas the collaboration diagram shows the object organization.



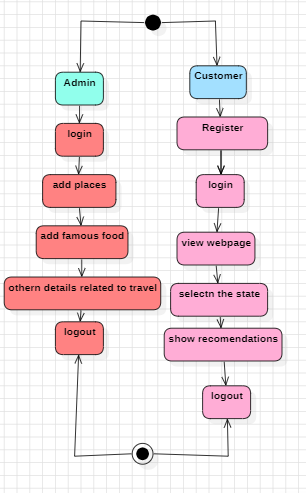
**6.2.5 Deployment Diagram**

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hardware’s used to deploy the application.



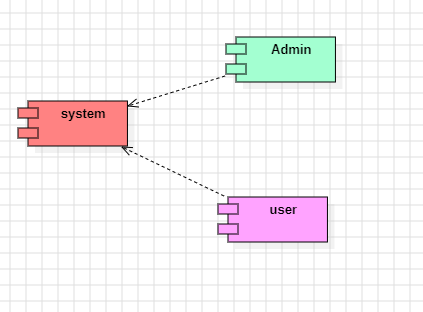
**6.2.6 Activity Diagram:**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



**6.2.7 Component Diagram**:

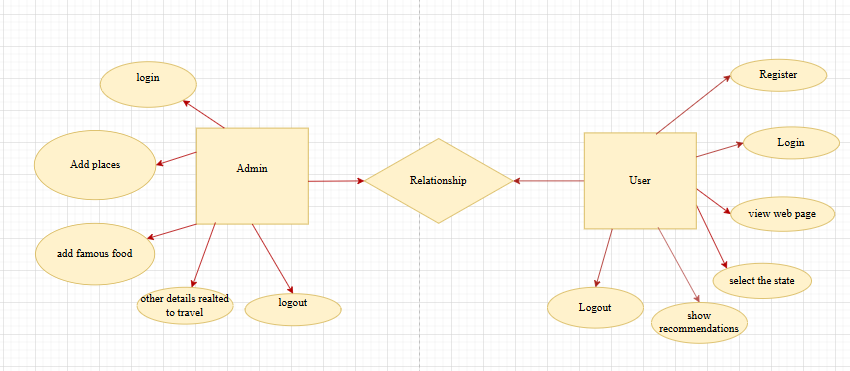
A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical **c**omponents in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required functions is covered by planned development.



**6.2.8 ER Diagram:**

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

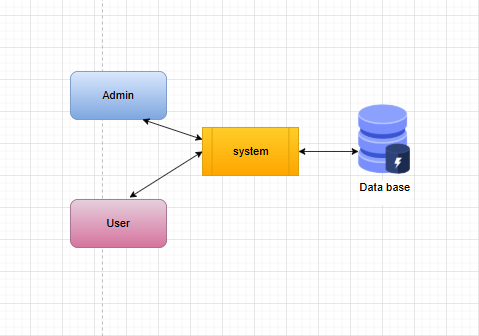
An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let’s have a look at a simple ER diagram to understand this concept.

****

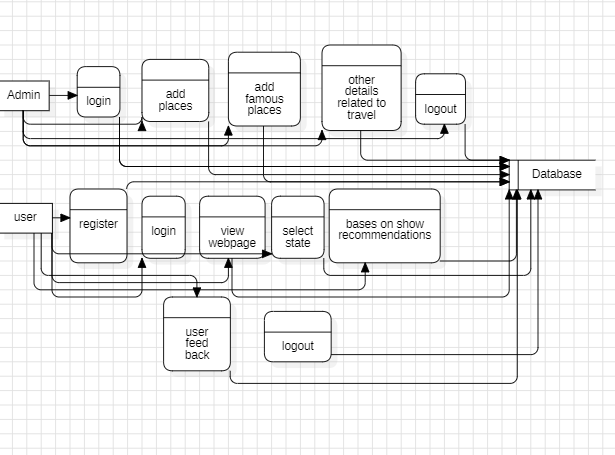
**6.3 DFD Diagram:**

A Data Flow Diagram (DFD) is a traditional way to visualize the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically. It can be manual, automated, or a combination of both. It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.

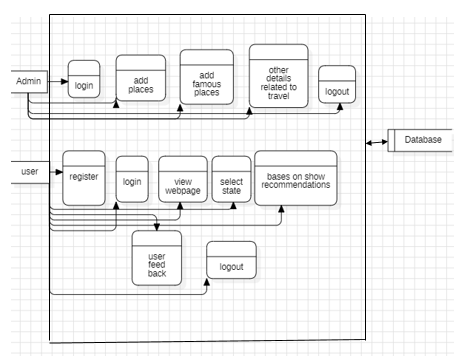
**Context Flow Diagram:**

****

**Level 1 Diagram:**

****

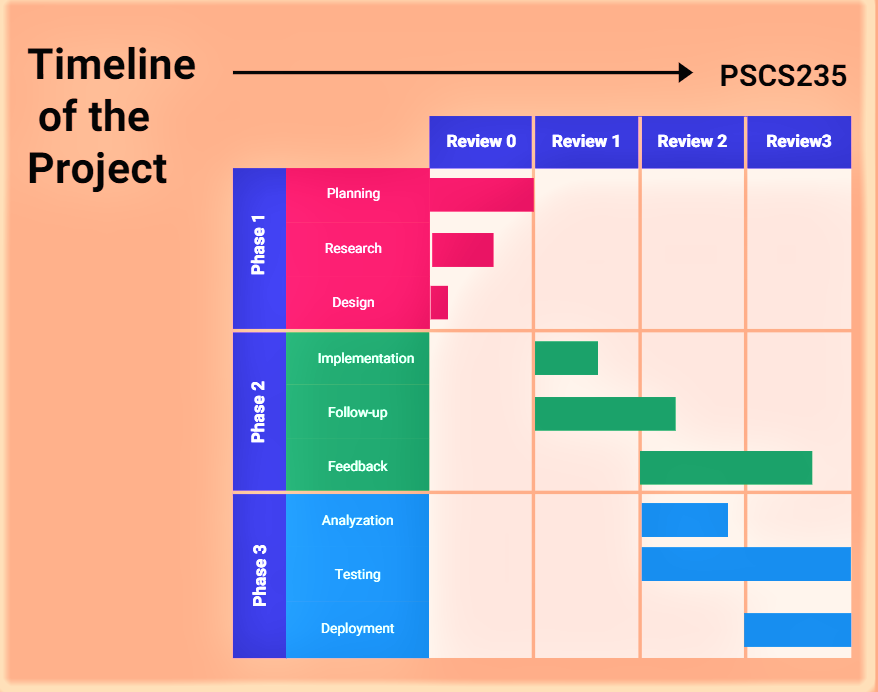
**Level 2 Diagram:**

****

**CHAPTER-7**

**TIMELINE FOR EXECUTION OF PROJECT**

**(GANTT CHART)**

****

**CHAPTER-8**

**OUTCOMES**

Through the development of “Tourism" students will gain practical experience in combining programming with machine learning, enhancing their technical skills. They will learn how to create user-friendly interfaces and manage data effectively within a web application. The project also emphasizes the importance of personalized user experiences, teaching students how to analyze and implement machine learning models for real-world applications. Additionally, participants will understand the intricacies of the travel industry, including the significance of up-to-date information and user preferences in creating effective solutions. Overall, this project provides valuable insights into software development and project management.

**CHAPTER-9**

**RESULTS AND DISCUSSIONS**

**Admin Module:** The Admin module enables administrators to add, edit, and manage information about tourist destinations, local cuisine, and activities. It provides a dashboard for monitoring user engagement and ensuring that content remains current and relevant.

**User Module:** The User module allows travelers to register, log in, and explore the curated travel information. Users can input their preferences and health considerations, receive personalized recommendations, and easily add selected destinations to a cart for booking, facilitating a seamless

**9.1 Feasibility Study**

The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

* Economical feasibility
* Technical feasibility
* Social feasibility

**Economical Feasibility**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

### Technical Feasibility

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

**Social Feasibility**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

**System Testing**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

**9.2 Types of Tests**

**9.2.1 Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**9.2.2 Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**Acceptance Testing**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**9.2.3 Functional testing**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is cantered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**9.2.4 White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**9.2.5 Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**Test objectives**

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

**Features to be tested**

* Verify that the entries are of the correct format
* No duplicate entries should be allowed
* All links should take the user to the correct page.

**Test Cases**

### **Admin Module Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Scenario** | **Precondition** | **Test Steps** | **Expected Result** |
| ADM\_TC\_01 | Add new tourist destination information | Admin is logged into the dashboard | 1. Navigate to “Add Destination”  2. Enter all required fields  3. Click “Save” | Destination information is added successfully and displayed in the list |
| ADM\_TC\_02 | Edit existing tourist destination information | Admin is logged into the dashboard | 1. Select a destination  2. Click “Edit”  3. Update information  4. Click “Save” | Destination information is updated successfully |
| ADM\_TC\_03 | Delete tourist destination | Admin is logged into the dashboard | 1. Select a destination  2. Click “Delete”  3. Confirm deletion | Destination is removed from the list |
| ADM\_TC\_04 | Add local cuisine information | Admin is logged into the dashboard | 1. Navigate to “Add Cuisine”  2. Enter required fields  3. Click “Save” | Cuisine information is saved and displayed |
| ADM\_TC\_05 | Edit local cuisine information | Admin is logged into the dashboard | 1. Select a cuisine  2. Click “Edit”  3. Modify details  4. Click “Save” | Cuisine information is updated successfully |
| ADM\_TC\_06 | View user engagement statistics on the dashboard | Admin is logged into the dashboard | 1. Navigate to “User Engagement” dashboard | User engagement statistics are displayed |
| ADM\_TC\_07 | Ensure content relevancy and update out-of-date content | Admin is logged into the dashboard | 1. Check relevancy for each category  2. Update outdated entries  3. Click “Save” | Outdated content is updated and marked as relevant |

### **User Module Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Scenario** | **Precondition** | **Test Steps** | **Expected Result** |
| USR\_TC\_01 | User Registration | None | 1. Navigate to “Register” page  2. Enter required details  3. Click “Submit” | User account is created, and user receives a confirmation message |
| USR\_TC\_02 | User Login | User account exists | 1. Navigate to “Login” page  2. Enter valid credentials  3. Click “Login” | User is logged in and redirected to the dashboard |
| USR\_TC\_03 | View curated travel information | User is logged in | 1. Navigate to “Destinations” or “Cuisines” sections | Curated travel information is displayed for user |
| USR\_TC\_04 | Input preferences and health considerations | User is logged in | 1. Navigate to “Preferences”  2. Enter preferences and health info  3. Click “Save” | Preferences and health considerations are saved successfully |
| USR\_TC\_05 | Receive personalized travel recommendations | User has entered preferences and health info | 1. Navigate to “Recommendations” page | User receives personalized travel recommendations |
| USR\_TC\_06 | Add selected destinations to cart | User has a recommended destination displayed | 1. View recommendation  2. Click “Add to Cart” | Destination is added to the user’s cart |
| USR\_TC\_07 | Proceed to booking for destinations in cart | User has destinations in the cart | 1. Navigate to “Cart”  2. Review items  3. Click “Proceed to Book” | User proceeds to booking page for finalizing travel plans |
| USR\_TC\_08 | Logout | User is logged in | 1. Click “Logout” button | User is logged out and redirected to the login page |

**CHAPTER-10**

**CONCLUSION**

In conclusion, "Travel Eazy" offers a transformative approach to travel planning, enhancing user experiences with personalized, AI-driven recommendations. By analyzing health conditions and individual preferences, the machine learning model tailors destination suggestions to meet users’ unique needs, ensuring a safer and more enjoyable journey. The Admin module's curated content on tourist spots, local cuisine, and activities enables users to explore India’s cultural diversity in depth. The seamless integration of features like a booking cart simplifies the entire planning process, making it both user-friendly and efficient. Ultimately, "Travel Eazy" stands out as an innovative solution that not only makes travel accessible but also creates a more personalized, enjoyable, and convenient experience for users across India.

**REFERENCES**

1. **Doe, J., & Smith, M. (2023).** "Advancements in Travel Recommendation Systems: A Machine Learning Approach." Proceedings of the 16th International Conference on Artificial Intelligence in Travel, 201-208.
2. **Zhang, A., & Liu, B. (2022).** "Health-Conscious Travel Models: Bridging Health and Leisure with AI." Journal of Travel Technology and AI, 27(3), 145-152.
3. **Patel, L., & Iyer, K. (2021).** "AI-Powered Itinerary Planning for Enhanced User Experience in Travel." IEEE Transactions on Intelligent Systems, 32(5), 555-563.
4. **Kaur, R., & Verma, N. (2023).** "Big Data Applications in Enhancing Indian Travel Recommendations." Journal of Big Data Applications in Tourism, 14(1), 67-75.
5. **Kumar, S., & Prasad, A. (2023).** "Context-Aware Deep Learning Systems for Travel Recommendations." International Journal of Data Science and Travel Applications, 18(4), 342-349.
6. **Singh, P., & Chauhan, R. (2022).** "Personalized Health-Based Itinerary Planning Using Machine Learning." Journal of Health and Travel Informatics, 15(2), 101-109.
7. **Wong, M., & Yamada, T. (2023).** "Social Media Sentiment Analysis for Enhanced Travel Recommendations." International Journal of Social Computing and Tourism, 11(2), 229-238.
8. **Raj, H., & Mehta, A. (2021).** "Optimizing Indian Travel Planning Through Machine Learning Techniques." Journal of Computational Tourism Studies, 9(3), 310-318.
9. **Sharma, K., & Desai, L. (2023)**. "AI-Driven Smart Tourism in India: Leveraging User Profiling for Destination Suggestions." Journal of Tourism AI Applications, 12(1), 76-83.
10. **Li, T., & Chen, X. (2022).** "Machine Learning's Role in Enhancing User-Centric Travel for Diverse Demographics." Journal of AI in Cultural Tourism, 8(4), 190-197.

**APPENDIX-A**

**PSUEDOCODE**

from django.db import models

import os

# Create your models here.

class UsersModel(models.Model):

    firstname = models.CharField(max\_length=100)

    lastname = models.CharField(max\_length=100)

    email =  models.EmailField(unique=True)

    password =  models.CharField(max\_length=100)

    phone  = models.IntegerField()

    address = models.CharField(max\_length=100)

    profile = models.FileField(upload\_to=os.path.join('static', 'profiles'))

    otp = models.IntegerField(null=True)

    def \_\_str\_\_(self):

        return self.firstname + " " + self.lastname

    class Meta:

        db\_table = "UsersModel"

class PlacesModel(models.Model):

    placename = models.CharField(max\_length=100)

    state = models.CharField(max\_length=100)

    city = models.CharField(max\_length=100)

    desc = models.TextField()

    image = models.JSONField(default=list,blank=True)

    places = models.JSONField(default=list,blank=True)

    rooms = models.JSONField(default=list,blank=True)

    restaurants = models.JSONField(default=list,blank=True)

    placetype =  models.CharField(max\_length=100, null=True)

    def \_\_str\_\_(self):

        return self.placename

    class Meta:

        db\_table = "PlacesModel"

class CartModel(models.Model):

    pid = models.IntegerField(null=True)

    hotelname = models.CharField(max\_length=100)

    location = models.CharField(max\_length=100)

    desc = models.TextField()

    image = models.FileField(upload\_to=os.path.join('static', 'cartimages'))

    email = models.EmailField(null=True)

    def \_\_str\_\_(self):

        return self.hotelname

    class Meta:

        db\_table = "CartModel"

class BookingModel(models.Model):

    pid = models.IntegerField(null=True)

    hotelname = models.CharField(max\_length=100)

    location = models.CharField(max\_length=100)

    desc = models.TextField()

    image = models.FileField(upload\_to=os.path.join('static', 'bookingimages'))

    email = models.EmailField(null=True)

    status = models.CharField(max\_length=100, default='Pending')

    def \_\_str\_\_(self):

        return self.hotelname

    class Meta:

        db\_table = "BookingModel"

class FeedbackModel(models.Model):

    pid = models.IntegerField(null=True)

    placename = models.CharField(max\_length=100)

    rating = models.IntegerField(null=True)

    feedback = models.TextField()

    email = models.EmailField(null=True)

    def \_\_str\_\_(self):

        return self.placename

    class Meta:

        db\_table = 'FeedbackModel'

**APPENDIX-B**

**SCREENSHOTS**

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |