A

Project Report

on

ECOWHEELS

Submitted in fulfillment of the HACKWARTS hackathon conducted by

Shree L.R TIWARI COLLEGE OF ENGINEERING, MIRA ROAD

By

Dhruv Mahyavanshi

Mayank Mishra

Sanket Kurle

Harshit Shah

Abhishek Kumbhar

Academic Year: 2024-25

TABLE OF CONTENTS

Chapter 1

Introduction

1.1 Purpose

The primary goal of EcoWheels is to simplify sustainable transportation. With rising environmental concerns and congestion issues in cities, this platform serves as a comprehensive solution for users looking to access, book, and manage eco-friendly transportation options, promoting a greener alternative for urban commuting

1.2 Problem Statement

Modern cities face challenges related to pollution and traffic congestion. While eco-friendly transportation solutions exist, their adoption is limited due to fragmented systems that make it difficult for users to access and book these services in one place. **EcoWheels** addresses this gap by integrating multiple eco-transport options into a single platform, providing users with real-time access to sustainable transportation.

1.3 Objectives

The main objectives of the EcoWheels platform are:

- To provide a unified platform for booking eco-friendly transportation.
- To create an intuitive user interface that works across devices.
- To offer real-time data on transportation availability and user bookings.
- To reduce the carbon footprint by promoting greener transportation options.

1.4 Scope

The main objectives of the EcoWheels platform are:

- Can provide a unified platform for booking eco-friendly transportation.
- Can create an intuitive user interface that works across devices.
- Can offer real-time data on transportation availability and user bookings.
- Can reduce the carbon footprint by promoting greener transportation options.

Chapter 2

Proposed System

2.1 User Authentication and Management

EcoWheels employs a Firebase-based authentication system for secure user logins and registrations. The platform ensures that user information is stored securely and allows users to manage their profiles, bookings, and payment history in a safe environment.

2.2 Admin Panel

The admin panel provides system administrators with the ability to manage transport listings, review user data, monitor bookings, and track system performance. It ensures the seamless functioning of the platform by providing powerful tools for administrative control.

2.3 Real-time Transportation Listings

Users can browse through various eco-friendly transportation options (electric bikes, car-sharing, etc.). Real-time availability data ensures that users can make instant bookings, reducing delays and improving the overall efficiency of the platform. Users can filter transportation options based on proximity, type of service, and vehicle specifications.

2.4 Interactive Dashboard

The dashboard provides users with a bird's-eye view of their activities on the platform. It includes information about past bookings, transportation usage statistics, and environmental impact. The dashboard allows both users and admins to monitor transportation trends, thus helping to promote sustainable transportation decisions.

2.5 Carbon Footprint Tracker

A core feature of the platform is its carbon footprint tracker, which provides users with detailed insights into how their transportation choices impact the environment. By calculating emissions saved through the use of eco-friendly options, the system fosters greater environmental awareness among users.

2.6 Booking System

The platform’s real-time booking system allows users to reserve eco-friendly vehicles instantly. The system integrates with GPS APIs to provide real-time availability, ensuring users can book the closest and most suitable transport options. Booking information is securely stored, and users receive notifications about their reservations.

2.2 System Architecture

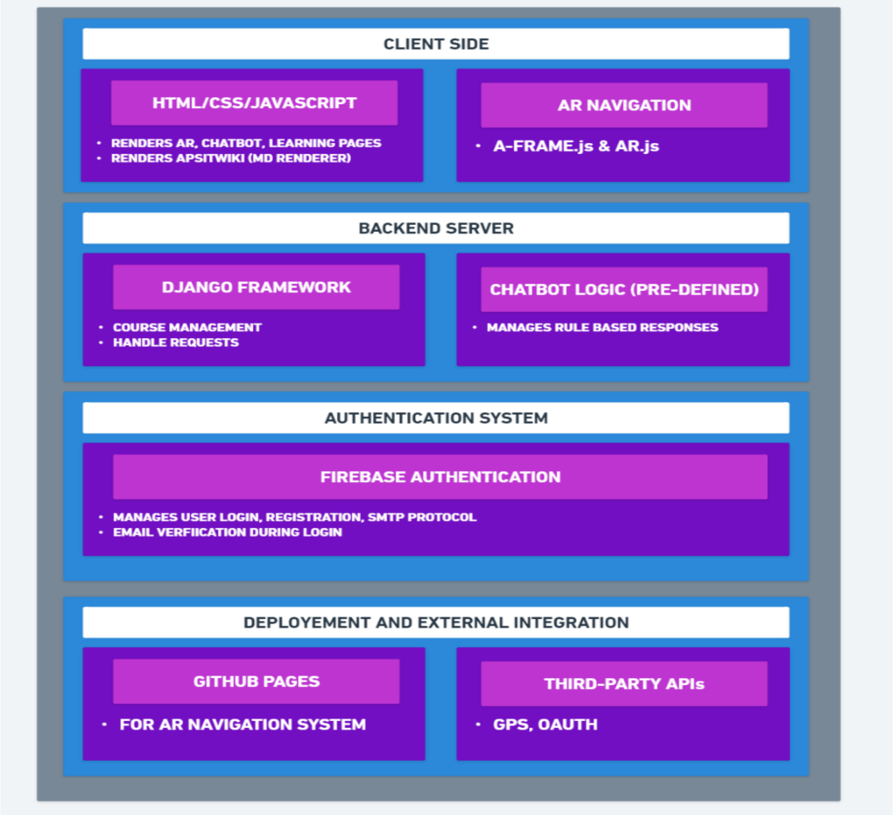
CLIENT SIDE
HTML/CSS/JAVASCRIPT
• RENDERS AR, CHATBOT, LEARNING PAGES
• RENDERS APSITWIKI (MD RENDERER)
AR NAVIGATION
• A-FRAME.js & AR.js
BACKEND SERVER
DJANGO FRAMEWORK
• COURSE MANAGEMENT
• HANDLE REQUESTS
CHATBOT LOGIC (PRE-DEFINED)
• MANAGES RULE BASED RESPONSES
AUTHENTICATION SYSTEM
FIREBASE AUTHENTICATION
• MANAGES USER LOGIN, REGISTRATION, SMTP PROTOCOL
• EMAIL VERFIICATION DURING LOGIN
DEPLOYEMENT AND EXTERNAL INTEGRATION
GITHUB PAGES
• FOR AR NAVIGATION SYSTEM
THIRD-PARTY APIs
• GPS, OAUTH

Fig 2.1: System Architecture

The system architecture can be explained as follows:

2.2.1 Client-Side

The frontend of the system is developed using HTML/CSS/JavaScript, ensuring responsive user interactions. The interface is intuitive and designed to provide seamless experiences across devices, particularly smartphones and tablets, allowing users to interact with the platform on the go.

2.2.2 Backend Server

The backend is powered by Django, which handles the server-side logic, data processing, and communication with the frontend. SQLite is used as the database to store user data, booking information, and transportation listings.

2.2.4 GPS and API Integration

EcoWheels integrates with GPS APIs to provide real-time location data for users. This helps in identifying available transportation options nearby and in offering precise navigation directions to the user's destination.

Chapter 4

Technical Specification

1.**Python Language:**

Python is a high-level, interpreted programming language known for its easy-to-read syntax and dynamic semantics. It's widely used for Rapid Application Development and as a scripting language to connect existing components. Here are some key points:

- Interpreted Language: Python code is executed line by line, making debugging easier and eliminating the need for a separate compilation step.
- Dynamic Typing: Variables in Python can change type dynamically, which adds flexibility to the language.
- Built-in Data Structures: Python comes with ready-to-use data structures, which simplifies code and enhances productivity.
- Extensive Libraries: A vast standard library and modules support various programming tasks, from web development to data analysis.

2.**Django Framework to develop GUI:**

Django is our framework of choice to build the GUI with since it offers a diverse range of built-in methods and parameters which expertly advance our user interface in usability.Django is a high-level Python web framework that empowers developers to build robust and maintainable web applications efficiently. Here are the key points about Django:

- Philosophy and Purpose:
  - Django simplifies web development by handling repetitive tasks, allowing developers to focus on writing their applications.
  - It adheres to the DRY (Don't Repeat Yourself) principle, emphasizing automation and code reusability.

- Features and Benefits:
  - Rapid Development: Django accelerates project implementation, reducing the time needed to build web applications.
  - Secure: Built by experienced developers, Django incorporates security best practices.
  - Scalable: It handles scalability challenges, making it suitable for both small projects and large-scale applications.
  - Fully Loaded: Django includes built-in features like authentication, admin interface, and database management.

    - Versatile: Developers can create various types of applications, from content management systems to social networks.
    - Open Source: Django is free and open source, with an active community contributing to its growth.
- MVT Design Pattern:
    - Django follows the MVT (Model-View-Template) design pattern:
        - Model: Represents the data structure (database schema).
        - View: Handles data processing and interacts with templates.
        - Template: Defines how data is presented in HTML.
- History and Popularity:
    - Django was publicly released in 2005 and has since gained widespread adoption.
    - Notable sites like Instagram, Mozilla, and Pinterest use Django for their web applications.

In summary, Django streamlines web development, provides security, and encourages clean, pragmatic design. Developers can build powerful applications while relying on Django's robust features

3. **Visual Studio Code(VCS) as Integrated Development Environment (IDE):**

Visual Studio Code (VCS) is a very specific pick for our project as it provides a very speculative look into our code along with quick and responsive info which assisted us in launching forward with a headstart on the frontend creation. Visual Studio is a powerful Integrated Development Environment (IDE) developed by Microsoft. It's designed for building a wide range of applications, including desktop, web, mobile, and cloud services. Here's a brief overview:

- **Multi-Language Support**: Visual Studio supports multiple programming languages like C#, C++, VB, Python, and JavaScript.

- **Development Platforms**: It uses Microsoft software development platforms like Windows API, Windows Forms, and Windows Presentation Foundation (WPF).

- **Editions**: Available in Community, Professional, and Enterprise editions, with the Community version being free for individual developers and open-source projects.

- **Features**: Includes a code editor with IntelliSense, code refactoring, debugging, and many other tools for software development.

4. **HTML:**

HTML, or HyperText Markup Language, is the standard markup language for creating web pages. It structures web content and is often assisted by CSS (Cascading Style Sheets) and JavaScript. Here’s a brief overview:

- Markup Language: HTML uses tags to define the structure and formatting of web content.
- Elements and Tags: Elements are the building blocks of HTML pages, enclosed by tags like <html>, <head>, and <body>.
- Web Browsers: Browsers interpret HTML documents to render multimedia web pages.
- HTML5: The latest version, HTML5, supports multimedia elements like audio and video.

5. **CSS:**

CSS, or Cascading Style Sheets, is a style sheet language used to describe the presentation of a document written in HTML or XML. It controls how web pages are displayed in browsers, allowing you to style elements using selectors and declarations. Here’s a brief overview:

- Presentation Control: CSS manages layout, colors, fonts, and overall look of web content.
- Separation of Content: It separates content from presentation, improving accessibility and maintainability.
- Flexibility: CSS enables different styles for various devices and screen sizes.
- Versions: The latest version, CSS3, introduced features like animations, transitions, and grid layouts.

6. **Markdown:**

Markdown is a lightweight markup language created by John Gruber in 2004. It’s designed to be easy to write and read, with a simple syntax that can be converted to HTML. Here’s a brief overview:

- Plain Text Formatting: Markdown allows you to format text using simple characters like asterisks for bold or underscores for italics.
- Extensibility: It supports additional features like tables, code blocks, and lists, making it versatile for various documentation needs.
- Widely Used: Popular on platforms like GitHub, Stack Overflow, and many forums for its readability and ease of use.
- File Extension: Markdown files typically have the extension .md or .markdown.

Chapter 4

Project Design

4.1 Use Case diagram

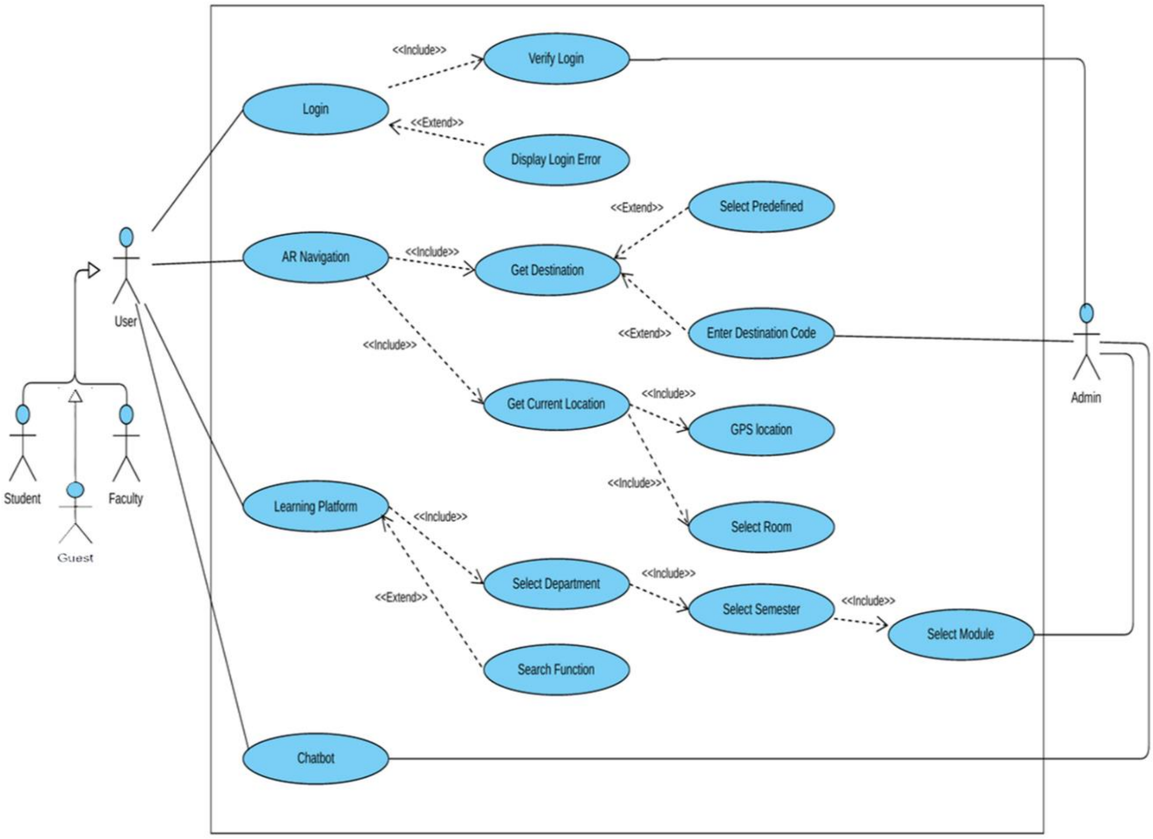

Fig 4.1 : Use Case Diagram

A detailed **use case diagram** outlines the interaction between various users (admin, user, guest) and the system. Key functionalities include user registration, login, booking transport, and managing transport listings through the admin panel.

4.2 DFD (Data Flow Diagram)

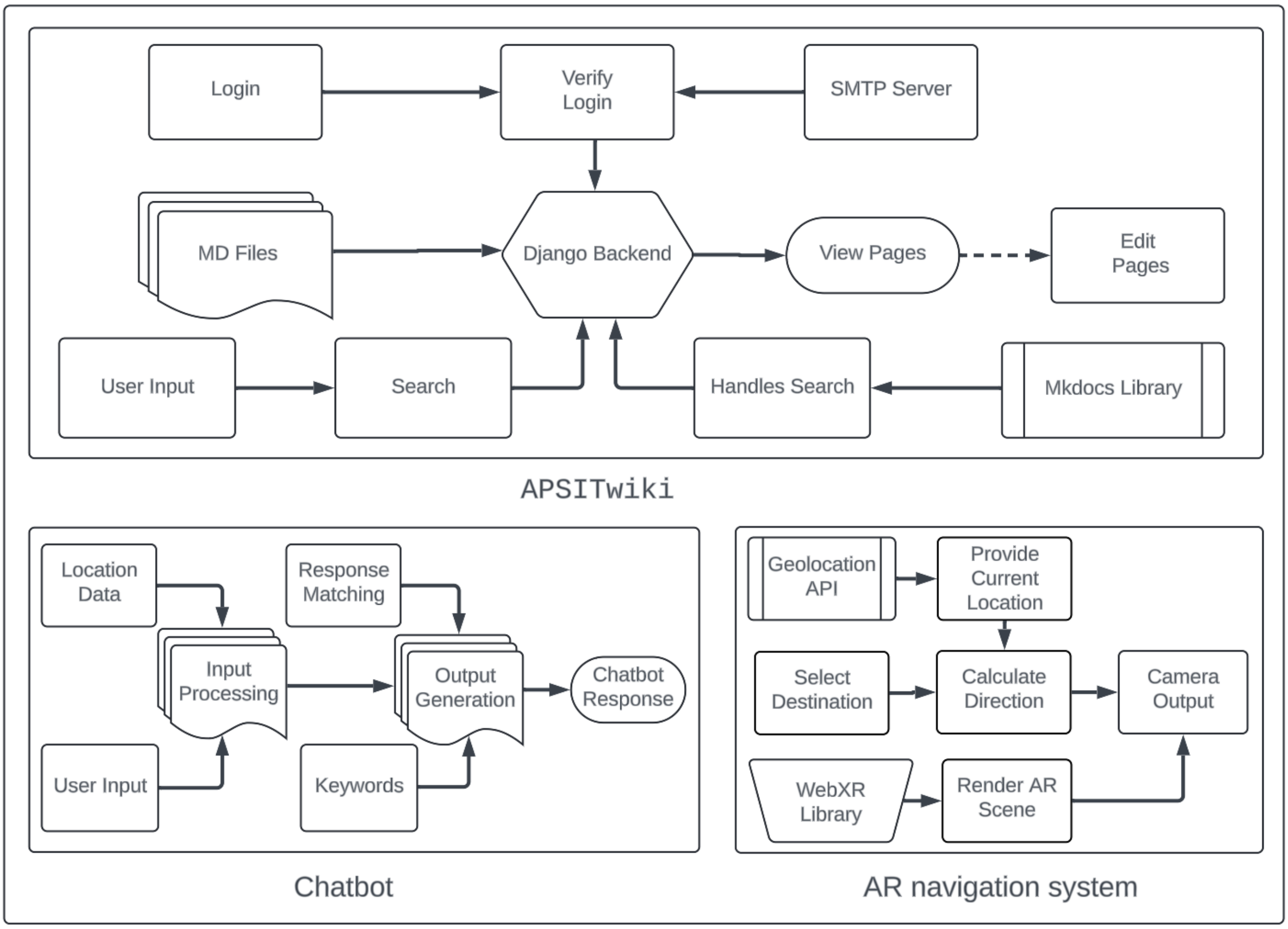

Fig 4.2: Data Flow Diagram

The data flow diagram (DFD) illustrates how information moves within the system. The user inputs data through the frontend, which is processed by the Django backend. The system interacts with external services like the GPS API and Firebase Authentication to deliver results, which are then presented to the user via the frontend.

Chapter 5

Results

- User Authentication: Successfully implemented using Firebase Authentication.
- Transport Booking System: Real-time booking is functional and offers users the ability to book eco-friendly transportation instantly.
- Admin Panel: Allows administrators to manage transport listings, track users, and manage system settings.
- Environmental Impact Tracking: The carbon footprint tracker provides detailed insights into the emissions saved by users through the platform.

Chapter 6

Conclusion & Future Scope

EcoWheels is a significant step toward promoting sustainable transportation by centralizing the booking and management of eco-friendly transport options. With additional features such as AR-based navigation and an enhanced chatbot system in future versions, the platform has the potential to revolutionize urban commuting and further reduce carbon footprints.

Future enhancements may include:

- **Wearable Integration**: For users to access the platform on devices like smartwatches or AR glasses.
- **Expanded Booking Options**: Integrating more transportation modes (electric buses, etc.) to broaden the scope of sustainable commuting options

References

[1] APSIT, APSIT moodle server, Thane: APSIT, 2016.

[2] Daniel Roy-Greenfeld and Audrey Roy-Greenfeld, "Django content Management," Django, NY, 2018.

[3] Adithela, Stephen Paul, et al. , ""Django content management system evaluation and integration" Proceedings of the Practice and Experience on Advanced Research Computing.," ACM, NJ, 2018.

[4] Fabrizio Romano, Gaston C. Hillar, Arun Ravindran, Learn Web Development with Python, LA: Packt Publishing Ltd, 2018.

[5] X. Qiao, P. Ren, S. Dustdar, L. Liu, H. Ma and J. Chen, Web AR: A Promising Future for Mobile Augmented Reality—State of the Art, Challenges, and Insights, DN: Proceedings of the IEEE, vol. 107, no. 4, pp. 651-666, 2019.

[6] F. L. M. K. K. P. N. A. M. B. A. D. M. K. a. E. T. Kostas Cheliotis, "A systematic review of application development in augmented reality navigation research," ResearchGate, SE, 2023.

[7] P. A. E. a. P. K. I. Satya Kiranmai Tadepalli, "Indoor Navigation Using Augmented Reality," ResearchGate, HYD, 2021.