DON BOSCO COLLEGE OF ENGINEERING

FATORDA, MARGAO, GOA - 403 602

DEPARTMENT OF COMPUTER ENGINEERING

2023 — 2024

"Weather Information System"

By

Mr. Reuben C Fernandes

Mrs. Grisellda Fernandes

Mrs. Rochelle Patil

Under the Guidance of

Prof. Amey Shet Tilve

BACHELOR OF ENGINEERING: GOA UNIVERSITY

DON BOSCO COLLEGE OF ENGINEERING

FATORDA, MARGAO, GOA - 403 602

2023 — 2024

CERTIFICATE

Second Year Project Report

"WEATHER INFROMATION SYSTEM"

submitted in partial fulfilment of the requirement for bachelors degree in computer engineering of Goa university is the bona fide work of

Mr. Reuben C Fernandes (2324006)

Mrs. Grisellda Fernandes (2214014)

Mrs. Rochelle Patil (2214045)

Prof. Amey Shet Tilve

Assistant Professor

PROJECT GUIDE

Dr. Gaurang S Patkar

HEAD OF DEPARTMENT

ABSTRACT

Clear Skies is a web-based application designed to provide users with real-time weather information, forecasts, and air quality indices across various locations globally. Leveraging the OpenWeather API, the project aims to deliver accurate and timely weather details via an intuitive interface developed using the MERN stack (MongoDB, Express.js, React, and Node.js). This service will feature capabilities for displaying current weather conditions, a five-day forecast, and the air quality index for user-specified locations. Additionally, ClearSkies will offer a unique functionality that displays a list of the most searched locations, enhancing user engagement by highlighting popular areas of interest. The application is intended for a wide range of users, from individuals planning their daily activities to travelers preparing for trips. With a focus on reliability, usability, and scalability, ClearSkies seeks to become a trusted resource for weather-related information, ensuring accessibility via various devices through a responsive web design.

ACKNOWLEDGEMENT

We would like to take this opportunity to express our sincere and heartfelt gratitude to all the stakeholders who have helped us directly or indirectly by providing us with the opportunity, the necessary expertise and guidance, motivating and mentoring us constantly. Without their valuable help this project would have always remained as a dream.We would like to thank our Director Rev.Fr.Kinley D'Cruz and our principal Dr.Neena S.P.Panandikar for providing us with the necessary facilities and the much needed motivation throughout the project. We are indebted to you.We would like to express our gratitude to our Head of the Department, Dr.Gaurang Patkar for showing confidence in us and for giving us timely suggestions and advice with his rich experience and expertise.Guides are like beacon of light showing us the right path in the phase of uncertainty and confusion during the course of the project. When the things seem to be drifting away from the goal, they put us back on the track with their constant monitoring and guidance and expertise. We are indebted to our Internal Guide Prof. Amey Shet Tilve.

TABLE OF CONTENTS

Chapter No.	Contents	Page No.
	TITLE PAGE	i
	CERTIFICATE I	ii
	CERTIFICATE II	iii
	ABSTRACT	iv
	ACKNOWLEDGEMENT	v
	TABLE OF CONTENTS	vi
	LIST OF FIGURES	ix
	LIST OF TABLES	x

Chapter 1 **INTRODUCTION** ……..01

1.1 Introduction to Project 01

1.2 Purpose of the project 01

1.3 Problem definition 01

1.3.1 Existing System.....01

1.3.2 Proposed system.....01

1.4 Future scope of the project.....02

1.5 Report organisation 02

Chapter 2 **SOFTWARE REQUIREMENT SPECIFICATION**.....03

2.1 Introduction.....03

2.2 Purpose 04

2.3 scope 04

2.4 Overall Description.....04

2.4.1 Product …….04

Chapter 3 **DESIGN** 05
3.1 Introduction 05
3.1.1 Scope and Purpose 05
3.1.2 Overall system design objective 05
3.1.3 Structure of Design Document.....05
3.2 Data Design.....05
3.3 Functional Design Description.....06
3.3.1 Data Flow Design.....06
3.4 Use Case Diagram……..06
3.5 Sequential Diagram 06

Chapter 4 **IMPLEMENTATION** 07

Chapter 5 **TESTING** 12

Chapter 6 **CONCLUSION** 14

REFERENCES 15

CHAPTER 1:
INTRODUCTION

1.1 Introduction to Project

ClearSkies is a dynamic weather information website that provides real-time weather updates, forecasts, and air quality indices. Utilising the robust capabilities of the OpenWeather API, the project is developed using the MERN stack, ensuring a responsive and scalable application. The platform is designed to cater to a wide audience range, aiming to assist individuals in daily weather planning and offering valuable insights for environmental monitoring.

1.2 Purpose of the Project

The main goal of the ClearSkies project is to provide dependable and easy-to-access weather data to people all over the world and to recognised the need for a user-friendly platform where users can quickly get the latest weather updates, forecasts, and air quality information. This kind of data is crucial for making informed choices about daily activities and travel plans.

1.3 Problem Definition

1.3.1 Existing System

The existing systems for weather forecasting are often cluttered with information and do not always prioritise user experience, leading to difficulties in quickly finding relevant weather data. Additionally, many platforms do not integrate air quality data comprehensively, which is crucial for health-conscious individuals or those in pollution-sensitive areas.

1.3.2 Proposed System

The proposed ClearSkies system will address these issues by providing a user interface that focuses on ease of use without compromising on the detail or accuracy of information. It will feature enhanced search functionality, real-time updates, and integration of air quality indices, all accessible through a single, user-friendly portal.

1.4 Future Scope of the Project

The future scope of ClearSkies includes:

- Expansion of data sources to include additional environmental parameters such as pollen counts and extreme weather alerts.
- Development of a mobile application to provide users with on-the-go access to weather forecasts.
- Integration of machine learning techniques to predict user preferences and provide personalised weather alerts and suggestions.

1.5 Report Organisation

The reports contain 6 Chapters, the chapters are organised as follows:

Chapter 1 introduces the topic and outlines the report's structure.

In Chapter 2, will discuss the project's software requirement specification, which includes the technologies we'll be using to develop the project and the various kinds of hardware and software needed for the research.

In Chapter 3, which addresses project design, we learn what kind of model we are implementing in our research and how it will benefit us.

Chapter 4 focuses entirely on the project's implementation, which essentially encapsulates the project's overall flow.

Chapter 5 discusses the various test cases used

conclusion are covered in detail in Chapter 6.

CHAPTER 2
SOFTWARE REQUIREMENT SPECIFICATIONS

2.1 Introduction

2.2 Purpose

The ClearSkies weather website is designed to provide timely and accurate weather information, including current conditions, detailed forecasts, and air quality indexes. A key feature of the system is its ability to display and store the most searched locations, enhancing user experience by highlighting popular areas of interest. This system aims to support users in making informed decisions about travel, health, and daily activities based on up-to-date weather data.

2.3 Scope

The ClearSkies platform simplifies access to comprehensive weather information through an intuitive interface that integrates with the OpenWeather API. This allows the platform to maintain current and reliable data efficiently.

Key Features:

- Search Functionality: Users can search for weather data by cities or regions. The most searched locations are displayed prominently.
- Weather Forecasts: Displays the current weather and provides a forecast chart for today as well as a six-day weather forecast.
- Air Quality Index: Shows real-time air quality data for searched locations.

Technologies Used: React (Frontend), Node.js with Express (Backend), MongoDB (Database), OpenWeather API (Data Source)

2.4 Overall Description

2.4.1 Product Perspective

The ClearSkies website serves as a comprehensive tool for global users to access essential weather data. The system is designed to be user-friendly, allowing users to quickly adapt and navigate its features. The architecture supports high availability and scalability to handle the dynamic nature of weather data and user interactions. With robust backend management, administrators can efficiently maintain system integrity and manage user data securely

CHAPTER 3
DESIGN

3.1.1 Scope and purpose

The purpose of the design phase is to develop a clear understanding of what the developer want people to gain from his/her project. As you the developer work on the project, the test for every design decision should be "Does this feature fulfil the ultimate purpose of the project?" A Purpose statement affects the design process by explaining what the developer wants the project to do, rather than describing the project itself. The Design Document will verify that the current design meets all of the explicit requirements contained in the system model as well as the implicit requirements desired by the customer.

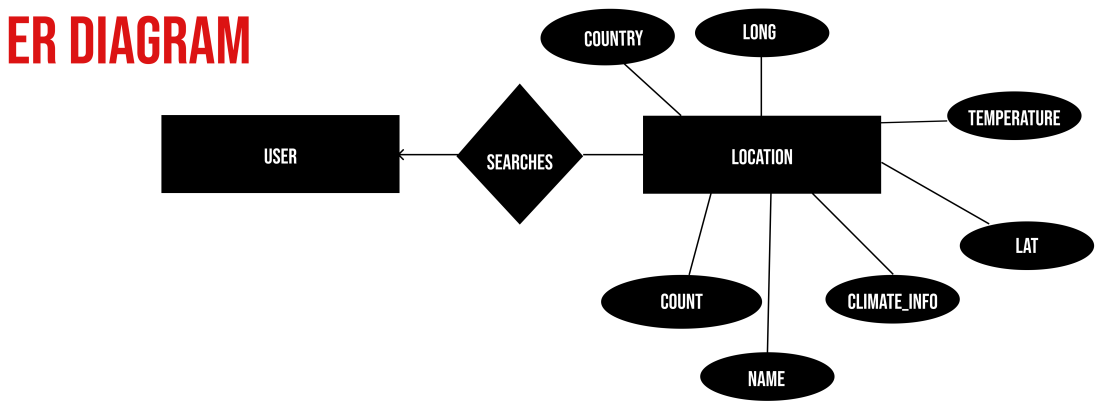
3.1.2 Overall System Design Objectives

The overall system design objective is to provide an efficient, modular design that will reduce the system's complexity, facilitate change and result in an easy implementation. This will be accomplished by designing strongly cohesion system with minimal coupling. In addition, this document will provide interface design models that are consistent user friendly and will provide straight forward transition through the various system functions.

3.1.3 Structure of Design Document

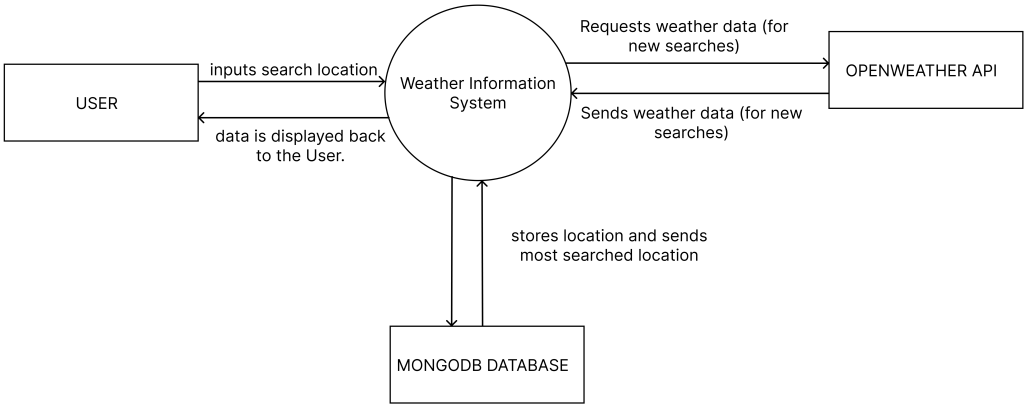
Data Design - The data Design include an ERD as well as Database design
Functional Design Description - This section has the functional partitioning from the SRS, and goes into great detail to describe each function.3.2 Data Design

3.2.1 Entity Relationship Diagram

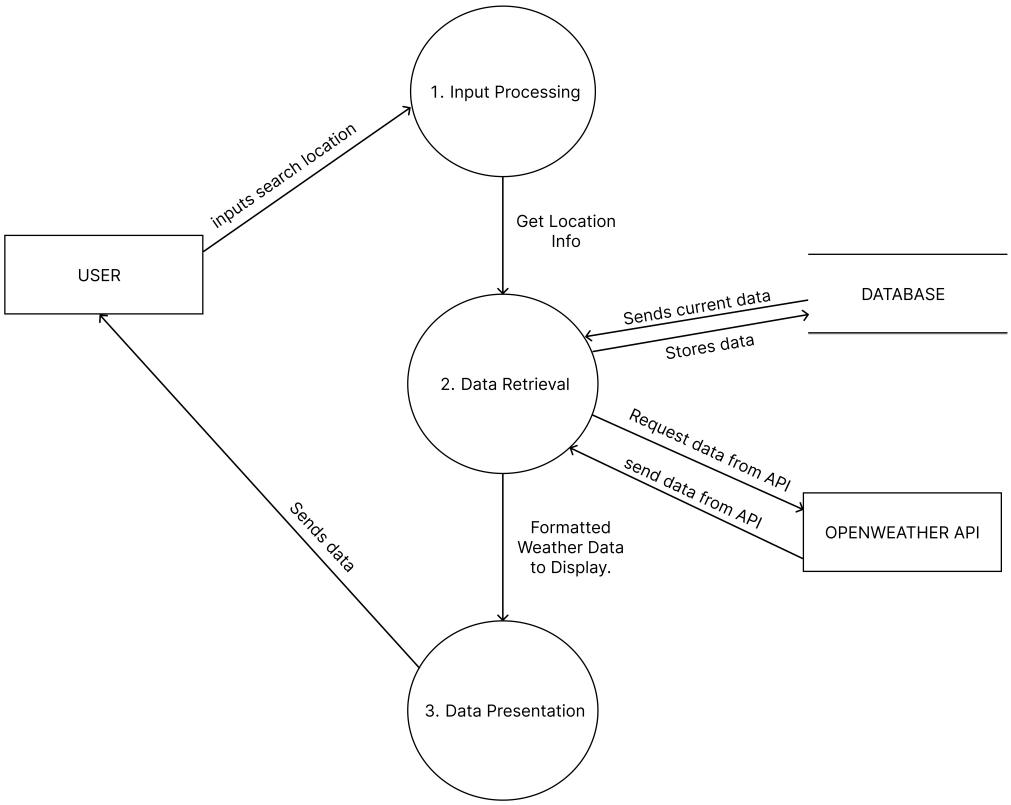


3.3.1 Data Flow Diagram:

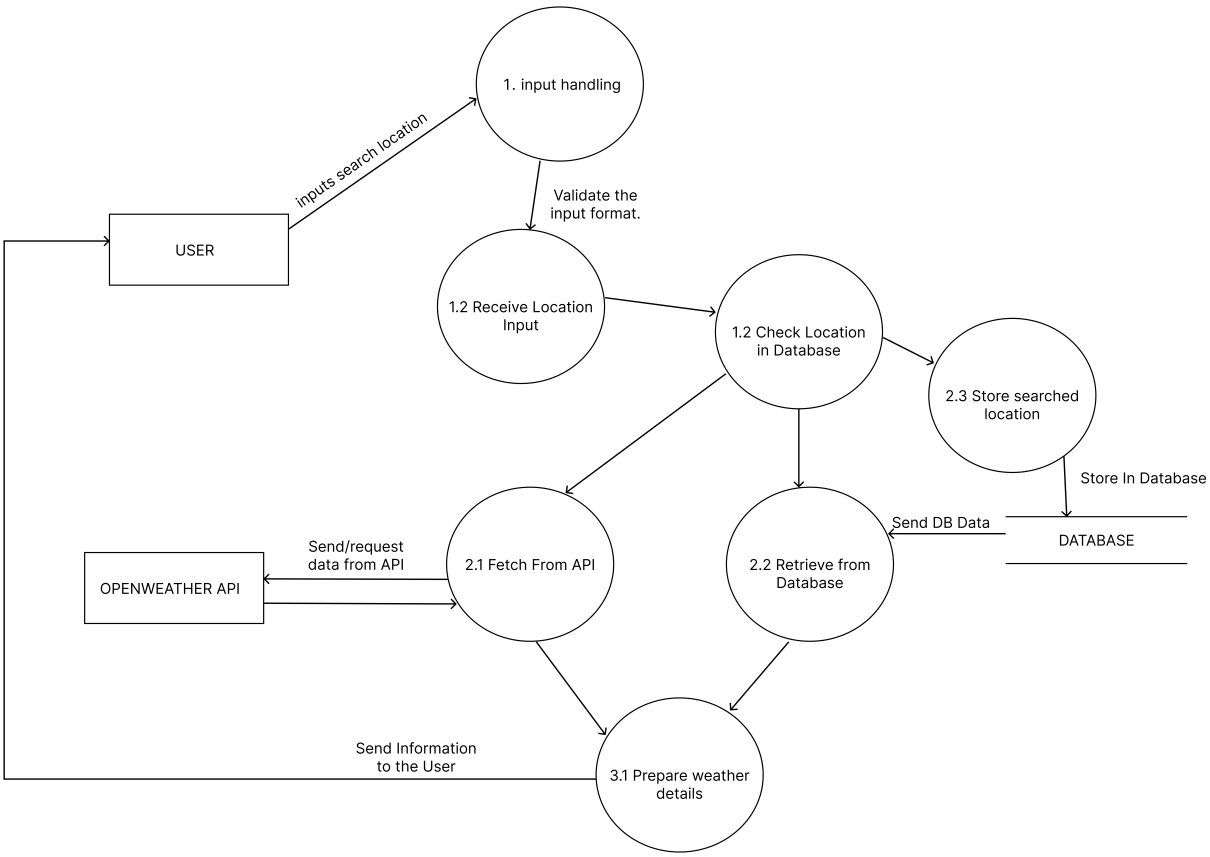
CONTEXT LEVEL DFD



LEVEL 0 DFD

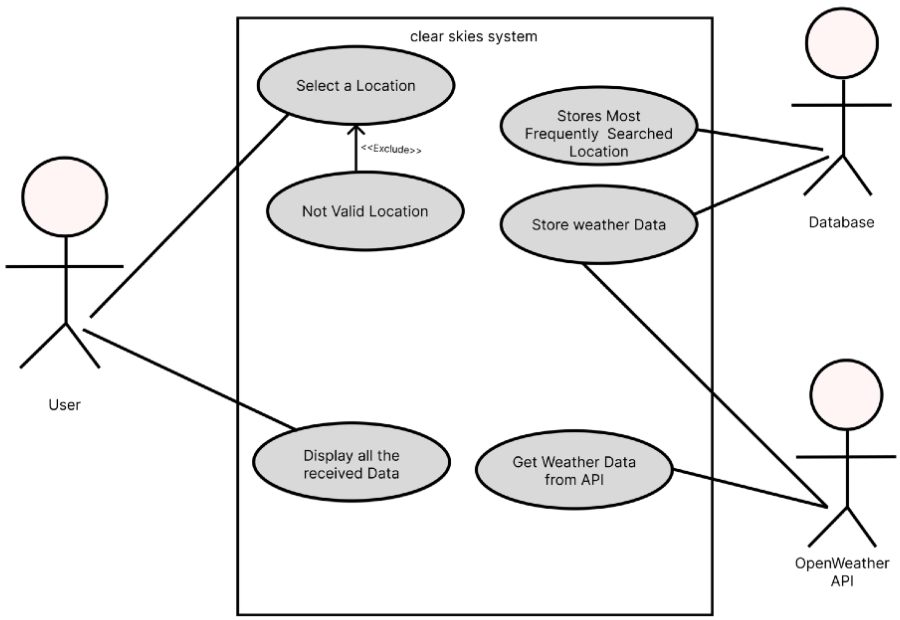


LEVEL 1 DFD

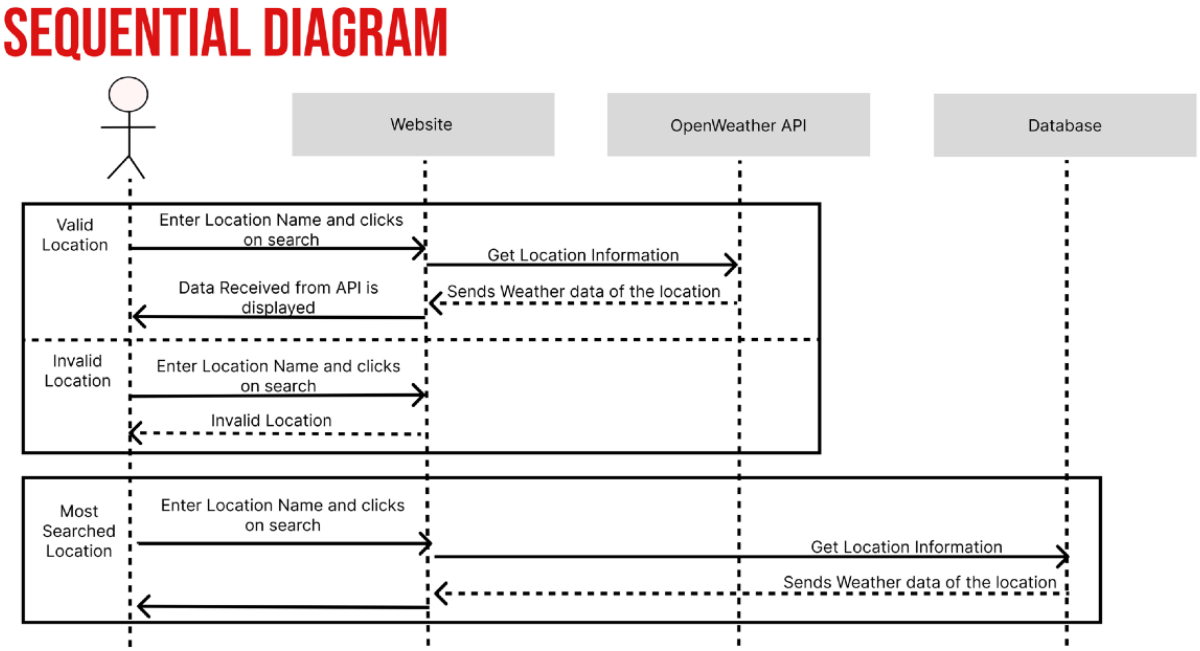


3.4 Use Case Diagram

USE CASE DIAGRAM



3.5 Sequential Diagram



CHAPTER 4
IMPLEMENTATION

Implementing the ClearSkies weather website involves a series of steps tailored to meet specific requirements for weather data management and user interaction using the MERN stack. This high-level overview will guide the development and deployment of the system, ensuring robust and reliable service.

4.1.1 Requirements Gathering:

Define Scope: Determine the precise scope of the weather data and user interaction capabilities.

Identify Features: Include features like live weather updates, forecast charts, air quality index, and tracking of most searched locations.

Data Types: Establish the data types necessary for weather conditions, user preferences, and search statistics.

4.1.2 Database Design:

Schema Design: Design a MongoDB schema to store and manage weather data and user information efficiently.

Entity Relationships: Define relationships between user data and weather records.

Attributes and Data Types: Specify attributes for each entity including location data, weather conditions, and temperature.

4.1.3 Backend Development:

Framework Selection: Utilise Node.js with Express for backend development.

API Development: Develop APIs to retrieve, update, and manage weather and user data.

CRUD Operations: Implement CRUD operations for managing weather data interactions.

4.1.4 Frontend Development:

Technology Stack: Use React for developing the frontend.

UI Development: Design and develop user interfaces that allow easy navigation and interaction with the weather data in Figma .

Data Interaction: Implement forms for search functionalities and display components for showing weather data and forecasts

4.1.5 Integration:

Connect Frontend and Backend: Ensure that the frontend and backend components work seamlessly together.

Testing Integration: Test the integration thoroughly to ensure that the API communication between client and server is flawless.

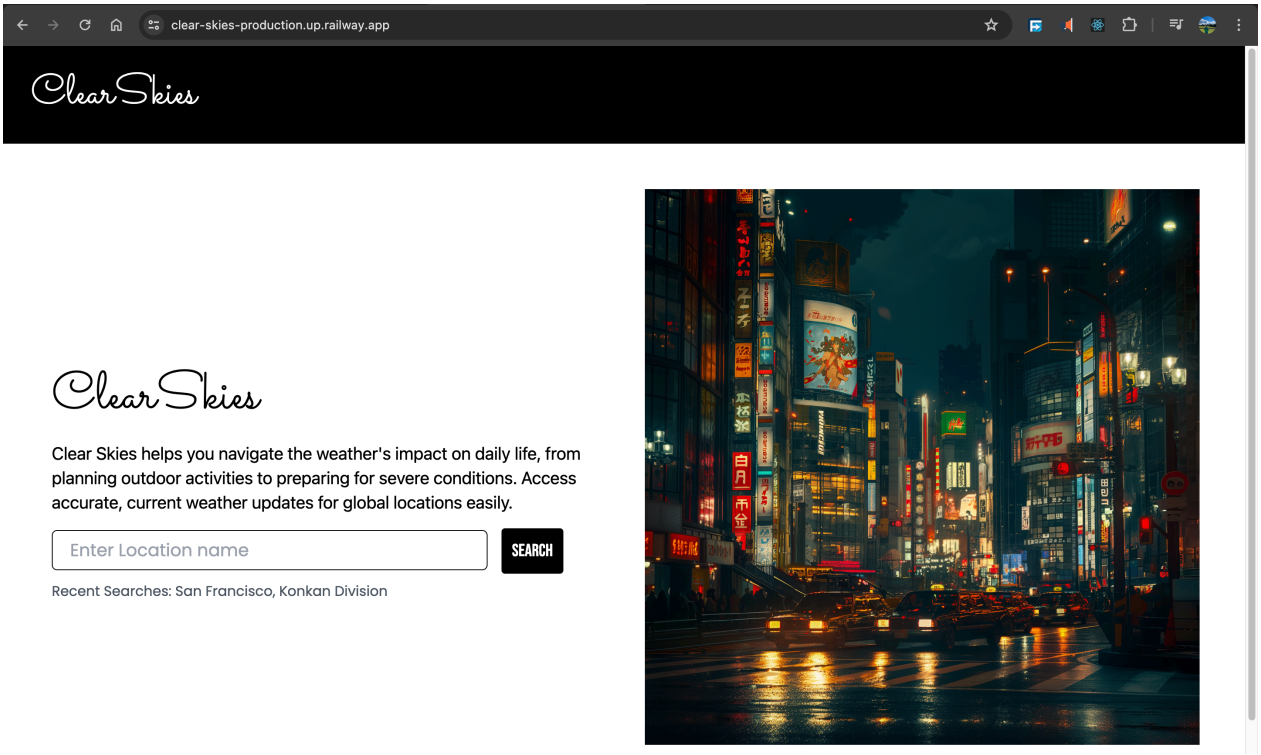
4.1.6 Deployment

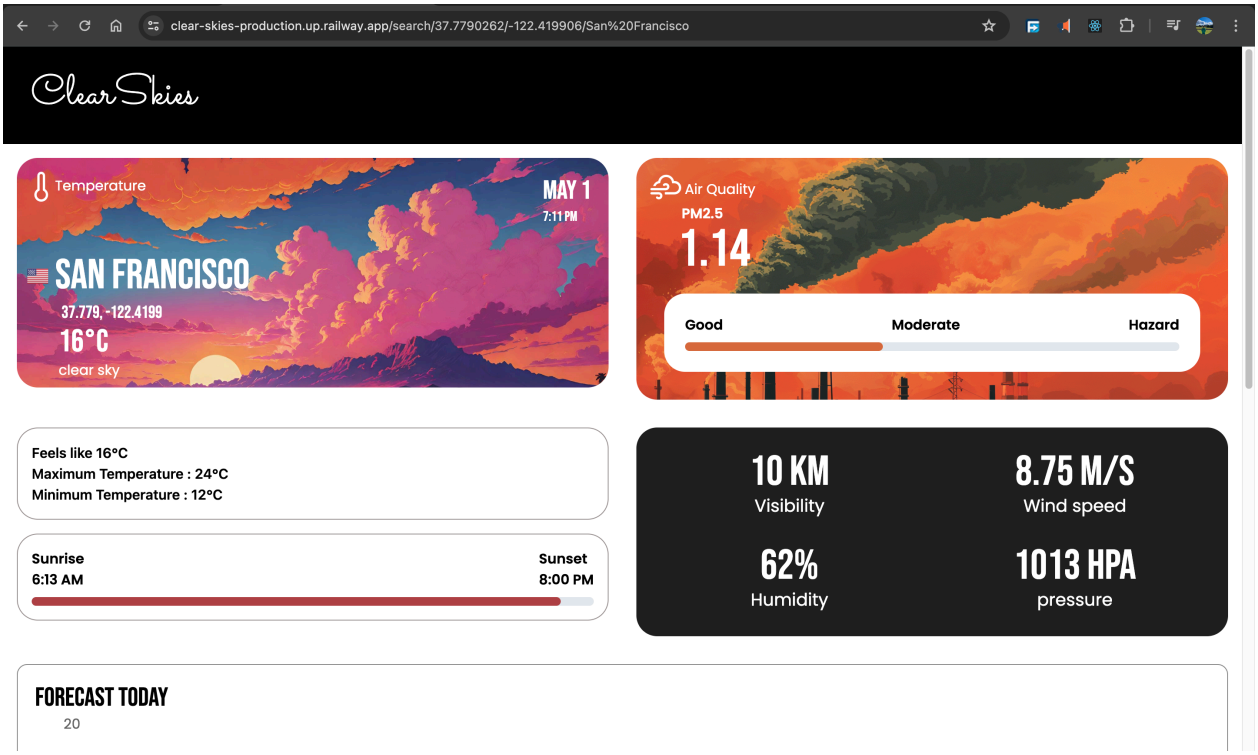
Hosting Platform: The ClearSkies weather website is hosted on Railway, providing robust, scalable, and easy-to-manage cloud hosting services. This hosting solution supports the continuous deployment and efficient scaling of web applications.

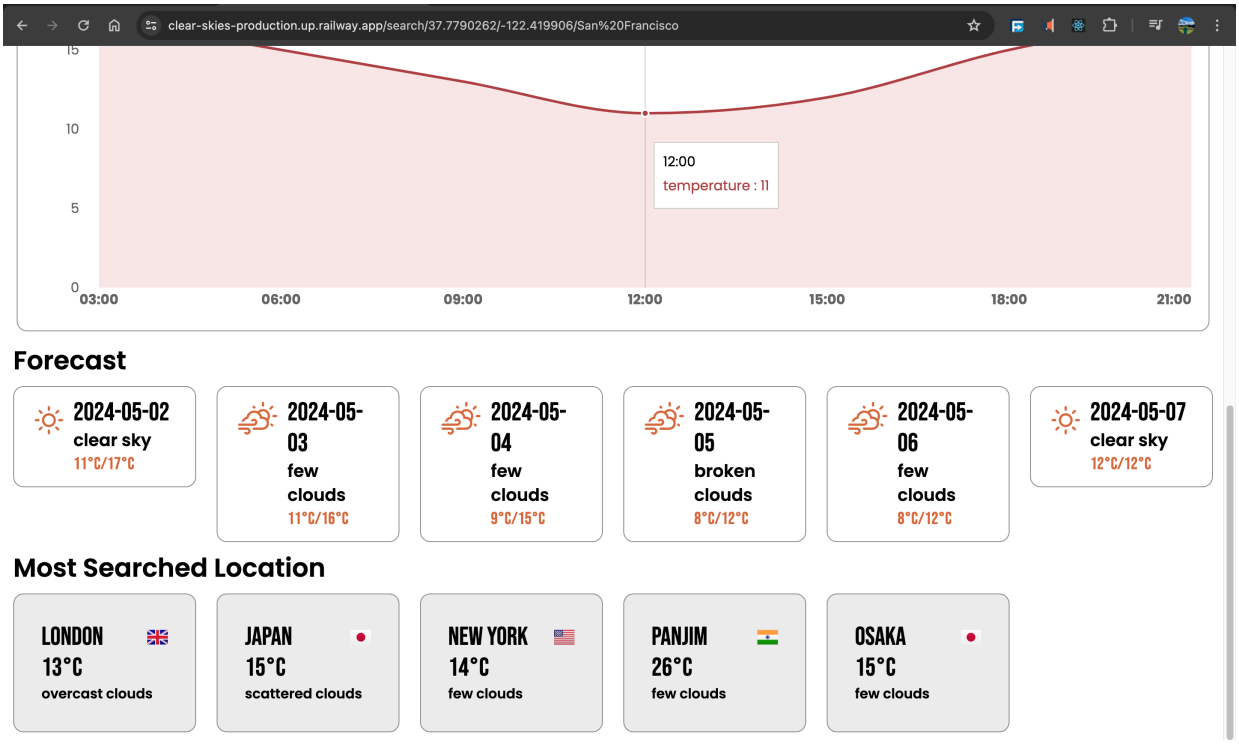
URL: clear-skies-production.up.railway.app - This is the live URL where the website can be accessed by users worldwide.

4.1.7 Testing and Quality Assurance:

Functional Testing: Conduct comprehensive tests to verify that all features work as intended.
Requirement Validation: Confirm that the system meets all outlined requirements and user needs.







Forecast

2024-05-02	2024-05-03	2024-05-04	2024-05-05	2024-05-06	2024-05-07
clear sky	few clouds	few clouds	broken clouds	few clouds	clear sky
11°C/17°C	11°C/16°C	9°C/15°C	8°C/12°C	8°C/12°C	12°C/12°C

Most Searched Location

LONDON	JAPAN	NEW YORK	PANJIM	OSAKA
13°C	15°C	14°C	26°C	15°C
overcast clouds	scattered clouds	few clouds	few clouds	few clouds

clear-skies-production.up.railway.app/NOTFOUND

OPEN WEATHER API COULDNT FIND THE GIVEN LOCATION

Go Home

clear-skies-production.up.railway.app/ghdxgszgb

404

Page Not Found

Couldnt find the page you were looking for

Go Home

CHAPTER 5
TESTING

Test Case 1: Verify Weather Search by City

Test Case ID: #CS001

Test Scenario: To verify that the user can successfully search for weather by entering a city name.

Test Steps:

1. Navigate to the ClearSkies homepage.
2. Enter "San Francisco" in the 'Search' field.
3. Click the 'Search' button.

Prerequisites: User has access to the internet.

Browser: Chrome v124. Device:intel i7 12th gen desktop.

Test Data: City name "San Francisco".

Expected/Intended Results: The webpage displays the current weather, forecast, and air quality index for San Francisco.

Actual Results: As expected.

Test Status – Pass/Fail: Pass.

Test Case 2: Validate Response to Incorrect City Name

Test Case ID: #CS002

Test Scenario: To validate that the system appropriately responds to an incorrect or misspelled city name.

Test Steps:

1. Navigate to the ClearSkies homepage.
2. Enter "San Fraciso" (misspelled) in the 'Search' field.
3. Click the 'Search' button.

Prerequisites: User has access to the internet.

Browser: Chrome v124. Device:intel i7 12th gen desktop.

Test Data: Incorrect city name "San Fraciso".

Expected/Intended Results: The webpage should display a message indicating that the city was not found or suggest corrections.

Actual Results: As expected.

Test Status – Pass/Fail: Pass.

Test Case 3: Verify Mobile Responsiveness

Test Case ID: #CS003

Test Scenario: To verify that the website is responsive and functions correctly on a mobile device.

Test Steps:

1. Open the website on a mobile browser.
2. Navigate through various pages (Main, weather)

Prerequisites: None.

Browser: Mobile chrome. Device: iPhone 11.

T**est Data:** N/A.

Expected/Intended Results: The website adjusts to fit the screen size and orientation smoothly, with all elements accessible and functional.

Actual Results: As expected.

Test Status – Pass/Fail: Pass.

Test Case 4: Validate Response to Incorrect Url

Test Case ID: #CS002

Test Scenario: To validate that the system appropriately responds to an incorrect URL

Test Steps:

1. Navigate to the ClearSkies homepage.
2. Enter an invalid URL Ex: https://clear-skies-production.up.railway.app/gsusuatsg in the 'Search' field.

Prerequisites: User has access to the internet.

Browser: Chrome v124. Device:intel i7 12th gen desktop.

Test Data: Incorrect URL.

Expected/Intended Results: The webpage should display a message indicating that the URL was not found or suggest corrections.

Actual Results: As expected.

Test Status – Pass/Fail: Pass.

The ClearSkies weather website has demonstrated its effectiveness as a robust tool in providing accurate and timely weather information to users globally. Since its implementation, the project has brought forth several benefits and yielded positive results.

CHAPTER 6
CONCLUSION

ClearSkies has significantly enhanced the efficiency of accessing weather data. By moving away from traditional, less interactive weather services to a dynamic web-based system, it allows for faster and more precise updates on weather conditions, forecasts, and air quality indexes. The automation of data fetching from the OpenWeather API has not only minimised errors but also ensured that the information provided is up-to-date, thereby reducing the time users spend searching for reliable weather information.

Additionally, the website has facilitated better user engagement through its intuitive design and responsive user interface. Users can easily navigate through the website, access various weather metrics quickly, and personalise their searches based on location. The feature to track and display the most searched locations has also enhanced the interactive aspect of the site, making it a more user-centered platform.

Furthermore, ClearSkies has improved the accessibility of weather information. With the website being available on multiple devices and platforms, users can access weather forecasts anywhere and anytime, which is particularly beneficial for those on the go or planning ahead for travel and outdoor activities.

Moreover, the implementation of the MERN stack and the continuous integration with the OpenWeather API have ensured that ClearSkies remains scalable and maintainable. This technical foundation allows the platform to handle increasing traffic and data as more users access the service, all while maintaining fast response times and reliable data accuracy.

References

1. MERN Stack Documentation:

MongoDB Manual: (https://docs.mongodb.com/manual/)

Express.js Guide: (https://expressjs.com/en/guide/)

React Documentation: (https://reactjs.org/docs/getting-started.html)

Node.js Handbook: (https://nodejs.org/en/docs/)

2. OpenWeather API:

OpenWeather API Documentation:(https://openweathermap.org/api)

3. Technical Articles and Online Resources:

Smashing Magazine: (https://www.smashingmagazine.com/category/coding/)

MDN Web Docs:(https://developer.mozilla.org/)

4. Tailwind css

Official website : (https://tailwindcss.com/docs/installation)