

Front End Engineering-II

Project Report

Semester-IV (Batch-2022)

Weather-App



Supervised By:

Raveesh Samkaria

Submitted By:

Shiven Narang

Roll Number: -2210990835

Group - 13

**Department of Computer Science and Engineering
Chitkara University Institute of Engineering & Technology,
Chitkara University, Punjab**

Abstract

This project encapsulates the creation of a weather app website utilizing HTML, Bootstrap, and JavaScript to deliver real-time weather information. The website integrates with a weather API to dynamically retrieve and display current weather conditions, forecasts, and related data for specified locations.

Built with a responsive design using Bootstrap, the weather app website ensures optimal viewing across various devices and screen sizes. The user interface is crafted to enhance user experience, featuring intuitive navigation and interactive elements that allow users to seamlessly explore weather details.

Key components of the weather app website include:

Weather Information: Display of current weather conditions, temperature, humidity, wind speed, and other relevant meteorological data.

Forecast: Presentation of future weather forecasts for upcoming days or hours, enabling users to plan ahead.

Location Search: Integration of a search function that allows users to enter locations and retrieve corresponding weather information.

User Interaction: Implementation of user-friendly features such as clickable elements, tooltips, or animations to enrich the user experience.

The website's development emphasizes accessibility features to ensure inclusivity and usability for all users, aligning with modern web standards. By leveraging Bootstrap for responsive design and JavaScript for dynamic content updates through API fetch calls, this weather app website exemplifies a blend of functionality and user-centric design principles.

The overarching goal of this project is to provide users with a seamless and informative digital experience for accessing weather data, reflecting a commitment to delivering high-quality web solutions tailored to practical user needs.

Table of Contents

Sr.no	Section	Page No.
1	Introduction	4
2	Problem Statement	5
3	Technical Details	6
5	File Structure	7
6	Result	8-25
7	References	26

Introduction

In the current digital age, a compelling online presence plays a crucial role in effectively presenting services and connecting with target audiences. This is especially relevant for weather app developers aiming to provide accessible and real-time meteorological information to users. This report details the creation and deployment of a front-end website for a weather app, utilizing HTML, Bootstrap, and JavaScript.

The website functions as a virtual platform for the weather app, offering users comprehensive weather data, forecasts, and interactive features. Through thoughtful design choices and an intuitive user interface, the website aims to engage visitors and provide them with valuable weather insights.

This introduction sets the context for the project's objectives, emphasizing the importance of a well-designed website in the context of digital tools for weather information. Subsequent sections will delve into technical aspects of website development, outlining key features, design considerations, and implementation strategies tailored for a weather app. Furthermore, the report will address any challenges encountered during development and the corresponding solutions devised, offering valuable insights for future web-based weather applications. Ultimately, this project showcases the synergy between technology and practical user needs, highlighting the app's commitment to delivering accurate and accessible weather information to its users.

Problem Statement

In the realm of weather applications, establishing a compelling online presence is essential for attracting users and showcasing the app's unique features and commitment to providing accurate meteorological data. However, developing a front-end website that effectively communicates the app's functionality, weather insights, and user interaction while ensuring optimal performance across devices presents several challenges.

Firstly, weather app websites must strike a balance between visual appeal and functionality. It's crucial to convey the app's data-driven approach through engaging designs while maintaining intuitive navigation and delivering essential weather information to users seeking forecasts and real-time updates.

Secondly, creating a responsive website that seamlessly adapts to various devices and screen sizes is critical. With the increasing use of smartphones and tablets for accessing weather information, users expect a consistent and user-friendly browsing experience regardless of the device used. Failing to meet these expectations could lead to user frustration and a diminished user base.

Moreover, ensuring accessibility for all users, including those with disabilities, is paramount. Weather app websites should adhere to accessibility standards to accommodate individuals with diverse needs, such as implementing keyboard navigation and providing alternative text for visual elements.

In summary, the challenge lies in developing a front-end website for a weather app that effectively integrates creativity with functionality, ensures seamless responsiveness across devices, adheres to accessibility guidelines, and optimizes performance to deliver an exceptional user experience. Overcoming these challenges is crucial for achieving the project's objectives and enhancing the app's visibility and usability among weather enthusiasts and casual users alike.

Technical Details

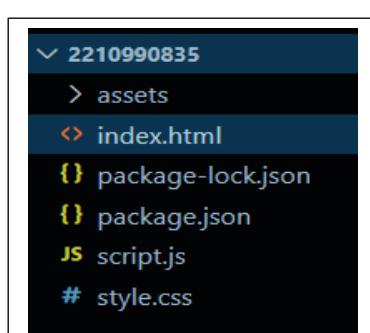
The front-end website for the weather app was developed with a streamlined approach, utilizing key technologies to deliver a responsive and efficient user interface. Here are the essential technical aspects of the project:

- **HTML Structure:** The website's structure was crafted using semantic HTML to enhance accessibility and readability. Each page was organized with appropriate tags to facilitate navigation and present weather data in a clear and concise manner.
- **Bootstrap Framework:** Bootstrap was leveraged to implement responsive design principles seamlessly. Utilizing Bootstrap's grid system and components ensured that the website layout adapts fluidly to different screen sizes, enhancing user experience across devices.
- **CSS Styling:** Cascading Style Sheets (CSS) were utilized to style the website's layout, typography, and color palette. Custom CSS rules were applied to maintain a cohesive visual identity while optimizing page load times and ensuring code simplicity.
- **JavaScript Integration:** JavaScript was employed to enhance the website's functionality and interactivity. Key features included dynamic data retrieval using API fetch calls to display real-time weather information. Additionally, JavaScript was used to implement user-friendly interactions such as location-based weather updates and interactive elements for seamless navigation.
- **API Integration:** The website integrated with a weather API to retrieve and display current weather conditions, forecasts, and related data. API fetch calls were implemented in JavaScript to fetch data asynchronously and update the user interface dynamically based on user inputs.

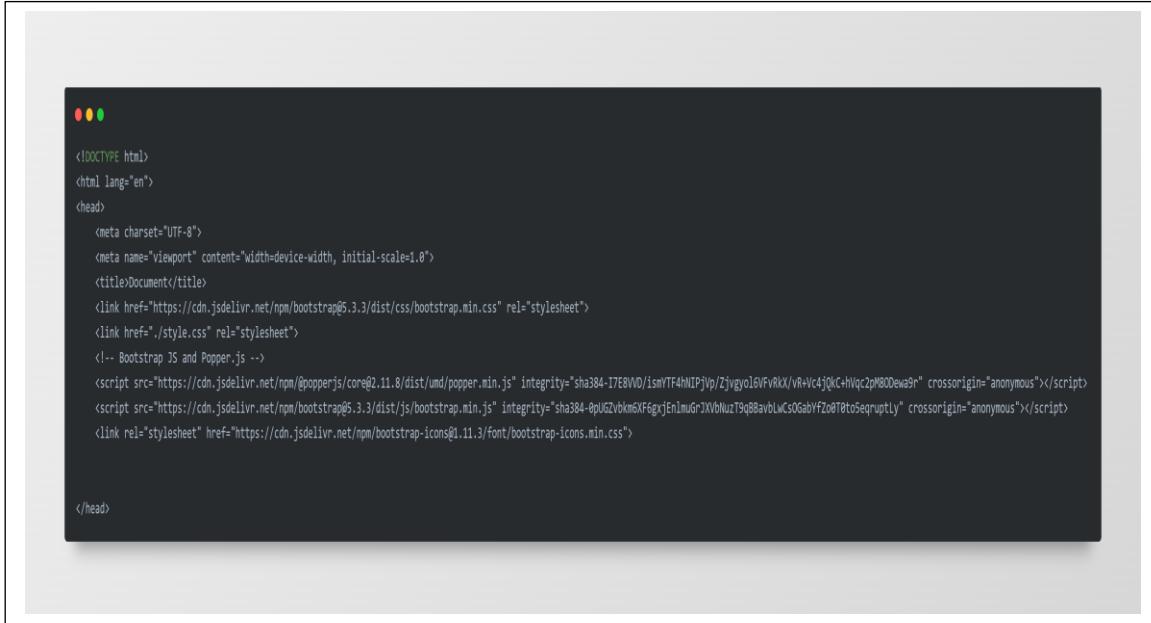
By leveraging these technologies, the weather app website achieved a balance between functionality and performance, delivering a responsive and engaging experience for users seeking accurate and accessible weather information across various devices and platforms.

File Structure: -

- **index.html:** This file serves as the central hub for the website, housing the HTML markup that defines the structure and content of the web pages. Within this file, sections such as the header, navigation bar, main content area, footer, and any additional elements are meticulously crafted to ensure a cohesive user experience.
- **index.css:** The CSS file is responsible for orchestrating the visual presentation of the website. It contains a meticulous set of styling rules governing colors, fonts, layout properties, and other aesthetic elements. Selectors within this file meticulously target specific HTML elements or classes to apply the desired styles, harmonizing the overall look and feel of the website.
- **index.js:** This JavaScript file holds the key to enhancing user interactivity on the website. In this minimalist setup, the primary functionalities may include features like smooth scrolling to different sections of the page, elegantly facilitated by the navigation bar.
- **assets folder:** This directory serves as the repository for all image assets utilized throughout the website. From logos to background images, each visual element is carefully curated to enrich the design and enhance the presentation of content. By organizing images methodically within this folder, accessibility and clarity are maintained, ensuring seamless integration into the website's layout.
- **PSD file:** This PSD (Photoshop Document) file encapsulates the initial design mockup crafted using Adobe Photoshop. While not directly integrated into the website's development process, it serves as a guiding beacon for design elements, layout, and visual styling, facilitating consistency and fidelity between the envisioned design and the final product.

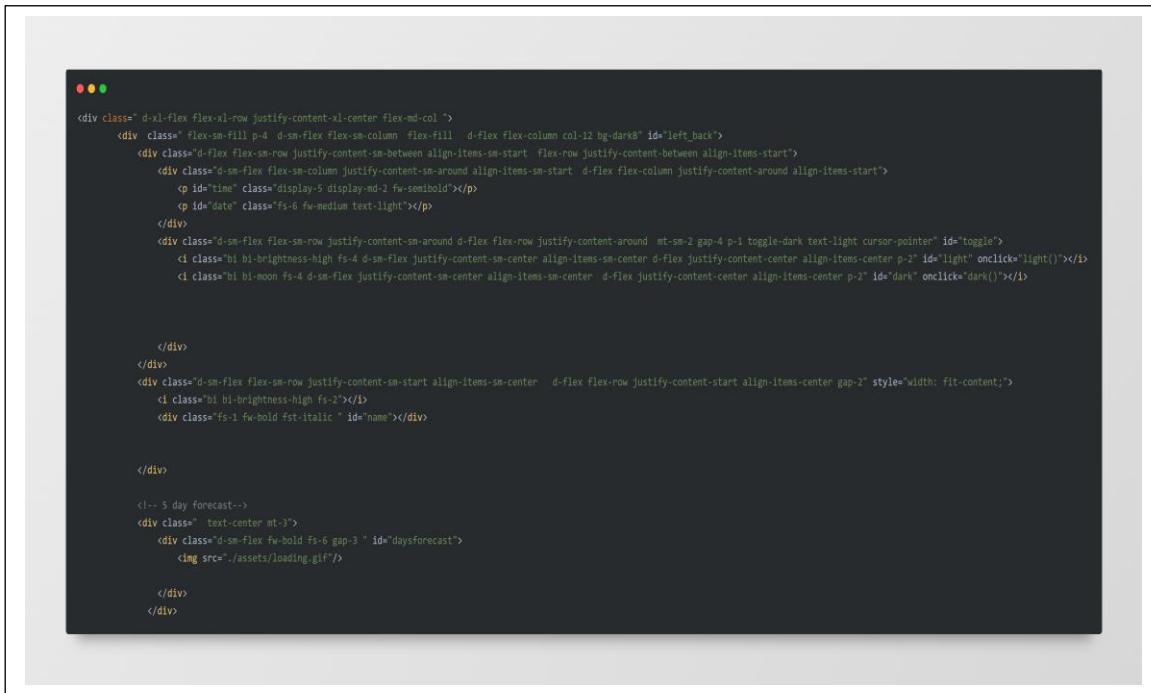


Result



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">
    <link href="style.css" rel="stylesheet">
    <!-- Bootstrap JS and Popper.js -->
    <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.8/dist/umd/popper.min.js" integrity="sha384-I7E8VVD/VMoWDFFZQ6FxqpkavAYtdz8jV1xWV1fkn8EUVq+J6jZyF3QFhHJX" crossorigin="anonymous"></script>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.min.js" integrity="sha384-O8tR+rWY5q9KsfC7VZJzq5OBQvTnqD7P7kW3oqjP3B1IqkqjZaVqcpM80Dewa9n" crossorigin="anonymous"></script>
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.3/font/bootstrap-icons.min.css">
</head>
```

Figure 1 Head Tag



```
<div class="d-xl-flex flex-xl-row justify-content-xl-center flex-md-col">
    <div class="flex-sm-fill p-4 d-sm-flex flex-sm-column flex-fill d-flex flex-column col-12 bg-dark" id="left_back">
        <div class="d-flex flex-sm-row justify-content-sm-between align-items-sm-start flex-row justify-content-between align-items-start">
            <div class="d-sm-flex flex-sm-column justify-content-sm-around align-items-sm-start d-flex flex-column justify-content-around align-items-start">
                <p id="time" class="display-5 display-md-2 fw-semibold"></p>
                <p id="date" class="fs-6 fw-medium text-light"></p>
            </div>
            <div class="d-sm-flex flex-sm-row justify-content-sm-around d-flex flex-row justify-content-around mt-sm-2 gap-4 p-1 toggle-dark text-light cursor-pointer" id="toggle">
                <i class="bi bi-brightness-high fs-4 d-sm-flex justify-content-sm-center align-items-sm-center d-flex justify-content-center align-items-center p-2" id="light" onclick="light()"></i>
                <i class="bi bi-moon fs-4 d-sm-flex justify-content-sm-center align-items-sm-center d-flex justify-content-center align-items-center p-2" id="dark" onclick="dark()"></i>
            </div>
        </div>
    </div>
    <div class="d-sm-flex flex-sm-row justify-content-sm-start align-items-sm-center d-flex flex-row justify-content-start align-items-center gap-2" style="width: fit-content;">
        <i class="bi bi-brightness-high fs-2" id="name"></i>
        <div class="fs-1 fw-bold fst-italic" id="name"></div>
    </div>
</div>

<!-- 5 day forecast-->
<div class="text-center mt-3">
    <div class="d-sm-flex fw-bold fs-6 gap-3" id="daysforecast">
        
    </div>
</div>
```

Figure 2 HTML- 5-Day Forecast Section

```
<div class="d-md-flex flex-md-row justify-content-md-start mt-5 align-items-md-start gap-md-3">
    <div class="flex-fill d-sm-flex flex-sm-column d-flex flex-column gap-3">
        <div class="d-sm-flex flex-sm-column d-flex flex-column round p-4 air-dark" id="air">
            <p class="fs-4 fw-bold text-light" id="air_quality">Air Quality Index</p>
            <div class="d-sm-flex justify-content-sm-between align-items-sm-center d-flex flex-row justify-content-between align-items-center gap-5">
                <i class="bi bi-geo-alt fs-5"></i>
                <p id="country_name" class="fs-5 fw-light text-light"></p>
            </div>
        </div>
        <div class="d-sm-flex flex-sm-row justify-content-sm-between align-items-sm-center d-flex flex-row justify-content-between align-items-center">
            <div class="d-sm-flex flex-sm-row justify-content-sm-around align-items-sm-center d-flex flex-row justify-content-around align-items-center gap-3">
                <i class="bi bi-cloud-fog fs-custom primary"></i>
                <div class="d-sm-flex flex-sm-column justify-content-sm-around align-items-sm-start d-flex flex-column justify-content-around align-items-start">
                    <p id="air_type" class="fs-2 fw-medium primary"></p>
                    <p id="air_info" class="fs-sm-3 fs-5 fw-light text-light"></p>
                </div>
            </div>
            <button class="btn btn-primary" onclick="airpollutionByCountry()">Refresh</button>
        </div>
        <div class="d-sm-flex flex-sm-row justify-content-sm-around align-items-sm-center d-flex flex-row justify-content-around align-items-center flex-fill div-effect-dark text-light" id="air_all">
            <div class="d-sm-flex flex-sm-column justify-content-sm-around align-items-sm-center d-flex flex-column justify-content-around align-items-center rounded p-2 col-1 fw-medium">
                <p id="pm2_value">N/A</p>
                <p>PM2</p>
            </div>
            <div class="d-sm-flex flex-sm-column justify-content-sm-around align-items-sm-center d-flex flex-column justify-content-around align-items-center rounded p-2 col-1 fw-medium">
                <p id="pm10_value">N/A </p>
                <p>PM10</p>
            </div>
            <div class="d-sm-flex flex-sm-column justify-content-sm-around align-items-sm-center d-flex flex-column justify-content-around align-items-center rounded p-2 col-1 fw-medium">
                <p id="so2_value"> N/A </p>
                <p>SO2</p>
            </div>
            <div class="d-sm-flex flex-sm-column justify-content-sm-around align-items-sm-center d-flex flex-column justify-content-around align-items-center rounded p-2 col-1 fw-medium">
                <p id="no2_value"> N/A </p>
                <p>NO2</p>
            </div>
            <div class="d-sm-flex flex-sm-column justify-content-sm-around align-items-sm-center d-flex flex-column justify-content-around align-items-center rounded p-2 col-1 fw-medium">
                <p id="o3_value"> N/A </p>
                <p>O3</p>
            </div>
            <div class="d-sm-flex flex-sm-column justify-content-sm-around align-items-sm-center d-flex flex-column justify-content-around align-items-center rounded p-2 col-1 fw-medium">
                <p id="co_value"> N/A </p>
                <p>CO</p>
            </div>
        </div>
    </div>
</div>
```

Figure 3 HTML – Air Quality Index Section

```
● ● ●
<div class="front3" id="about">
  <div class="front3title">
    Title Here
  </div>
  <div class="front3text1">
    >
      Lorem Ipsum available, but the majority have suffered alteration in some
      form.
    </div>
    <div class="doctors">
      <div class="front3img"
        data-aos="fade-right"
        data-aos-delay="50"
        data-aos-duration="1000"
        data-aos-easing="ease-in-out"
        data-aos-once="false"
      >
        
        <p class="front3imghead">
          Veterinarian
        </p>
        <p class="front3imgtext">
          Lorem Ipsum available, but the majority have suffered alteration in
          some.
        </p>
      </div>

      <div class="front3img"
        data-aos="fade-up"
        data-aos-delay="50"
        data-aos-duration="1000"
        data-aos-easing="ease-in-out"
        data-aos-once="false"
      >
        
        <p class="front3imghead">
          Vaccination Care
        </p>
        <p class="front3imgtext">
          Lorem Ipsum available, but the majority have suffered alteration in
          some.
        </p>
      </div>

      <div class="front3img"
        data-aos="fade-left"
        data-aos-delay="50"
        data-aos-duration="1000"
        data-aos-easing="ease-in-out"
        data-aos-once="false"
      >
        
        <p class="front3imghead">
          Dental Care
        </p>
        <p class="front3imgtext">
          Lorem Ipsum available, but the majority have suffered alteration in
          some.
        </p>
      </div>
    </div>
  </div>
</div>
```

Figure 4 HTML - About Section



```
<div>
  <iframe width="100%" height="300" src="https://api.maptiler.com/maps/satellite/?key=ezju5DPQNa2QzAI9qk5#0.7/6.37398/8.11378" style="border-radius: 1rem;"></iframe>
</div>
<div class="flex-fill p-3 d-sm-flex flex-sm-column justify-content-sm-between flex-fill p-3 d-flex flex-column justify-content-between gap-4 aiz-dark mt-sm-3 mt-xl-0 mt-3" id="sunrise_sunset">
  <div class="d-sm-flex flex-sm-row justify-content-sm-between align-items-sm-start d-flex flex-row justify-content-between align-items-start">
    <p class="fs-3 text-light" id="sunrise_text">Sunrise & Sunset</p>
    <i class="bi bi-plus-circle fs-2 cursor-pointer" onclick="addMore()"></i>
  </div>
</div>
```

Figure 5 HTML – Map Section

```

    <div class="d-sm-flex flex-sm-column justify-content-sm-around d-flex flex-column justify-content-around container-sm p-3 sunrise-dark" id="sunrise_inside" >
      <div class="d-sm-flex flex-sm-row justify-content-sm-start d-flex flex-row justify-content-start gap-1">
        <i class="bi bi-geo-alt fs-5 text-info"></i>
        <p id="country_name1" class="fs-5 fw-light"></p>
      </div>

      <div class="d-sm-flex flex-sm-row justify-content-sm-around align-items-sm-center d-flex flex-row justify-content-around align-items-center gap-2 ">
        <div class="d-sm-flex flex-sm-row justify-content-sm-around align-items-sm-center d-flex flex-row justify-content-around align-items-center gap-3">
          <i class="bi bi-brightness-high fs-1 text-warning"></i>
          <div class="d-sm-flex flex-sm-column justify-content-sm-start align-items-sm-center d-flex flex-column justify-content-start align-items-center">
            <p>Sunrise</p>
            <p id="sunrise_time" class="text-info"></p>
          </div>
        </div>
      </div>

      <div class="d-sm-flex flex-sm-row justify-content-sm-around align-items-sm-center d-flex flex-row justify-content-around align-items-center gap-3">
        <i class="bi bi-moon-stars fs-1 text-warning"></i>
        <div class="d-sm-flex flex-sm-column justify-content-sm-start align-items-sm-center d-flex flex-column justify-content-start align-items-center">
          <p>Sunset</p>
          <p id="sunset_time" class="text-info"></p>
        </div>
      </div>

    </div>
    <!--random sunrise-->

    <div class="d-sm-flex flex-sm-column justify-content-sm-around d-flex flex-column justify-content-around container-sm p-3 sunrise-dark" id="sunrise_inside1" >
      <div class="d-sm-flex flex-sm-row justify-content-sm-start d-flex flex-row justify-content-start gap-1">
        <i class="bi bi-geo-alt fs-5 text-info"></i>
        <p id="country_namerandom" class="fs-5 fw-light"></p>
      </div>

      <div class="d-sm-flex flex-sm-row justify-content-sm-around align-items-sm-center d-flex flex-row justify-content-around align-items-center gap-2 ">
        <div class="d-sm-flex flex-sm-row justify-content-sm-around align-items-sm-center d-flex flex-row justify-content-around align-items-center gap-3">
          <i class="bi bi-brightness-high fs-1 text-warning"></i>
          <div class="d-sm-flex flex-sm-column justify-content-sm-start align-items-sm-center d-flex flex-column justify-content-start align-items-center">
            <p>Sunrise</p>
            <p id="sunrise_timerandom" class="text-info"></p>
          </div>
        </div>
      </div>

      <div class="d-sm-flex flex-sm-row justify-content-sm-around align-items-sm-center d-flex flex-row justify-content-around align-items-center gap-3">
        <i class="bi bi-moon-stars fs-1 text-warning"></i>
        <div class="d-sm-flex flex-sm-column justify-content-sm-start align-items-sm-center d-flex flex-column justify-content-start align-items-center">
          <p>Sunset</p>
          <p id="sunset_timerandom" class="text-info"></p>
        </div>
      </div>

    </div>
    <!--random sunset-->

    <div id="addmore" >
      </div>
    </div>

    </div>
  </div>
</div>

```

Figure 6 HTML – Sunrise and Sunset Section

```


<form onsubmit="performSearch(event)" class="mt-3">
  <div class="d-sm-flex flex-sm-row justify-content-sm-start align-items-sm-center gap-2 p-1 mx-auto search-dark" id="search_bar">
    <i class="bi bi-search fs-4"></i>
    <input type="search" id="search" oninput="searchHandler(event)" class="search-dark-text" placeholder="Search..."/>
  </div>
</form>

<!--button onclick="performSearch()">search</button-->
<div class="d-sm-flex flex-sm-column d-flex flex-column p-3 rounded" id="rootdiv" style="background-image: url('./assets/all.gif');">
  <div class="d-sm-flex flex-sm-row d-flex flex-row gap-2">
    <i class="bi bi-geo-alt fs-3"></i>
    <p id="location_name" class="fs-4">N/A</p>
  </div>
  <div id="icon" class="d-sm-flex flex-sm-column justify-content-sm-center align-items-sm-center d-flex flex-column justify-content-center align-items-center"><img src='./assets/error_weather.webp' width='30%' /></div>
  <div id="today_date" class="text-center fs-5">N/A</div>
  <div id="current_temp" class="text-center fs-0 fw-bold">N/A</div>
  <div id="temp_name" class="text-center fs-4">N/A</div>
  <div class="d-sm-flex flex-sm-row justify-content-sm-start align-items-sm-center d-flex flex-row justify-content-start align-items-center gap-3 text-center mx-auto col-7 me-5">
    <div class="d-sm-flex flex-sm-row justify-content-sm-start align-items-sm-center d-flex flex-row justify-content-start align-items-center gap-2">
      <i class="bi bi-wind"></i>
      <div>Wind</div>
    </div>
    <div class="fs-3 " ></div>
    <div id="windSpeed">N/A</div>
  </div>


```

Figure 7 HTML – Current Weather Section

```
</div>
<div class="d-sm-flex flex-sm-column justify-content-sm-around d-flex flex-column justify-content-around gap-4 d-sm-none d-xl-inline mt-3 mt-sm-0" >
    <div id="add_more_data" class="text-light">
        </div>

    </div>
    <div class="d-sm-flex flex-sm-column justify-content-sm-around align-items-sm-center bg-success p-3 rounded gap-2 fixed-size-div mt-3 mt-sm-0">
        <div id="news_title" class="fs-5 fw-bold">N/A</div>
        <hr/>
        <div id="news_description" class="fw-bold">N/A</div>
    </div>
</div>
</div>
```

Figure 8 HTML – News Section

```
@import url("https://fonts.googleapis.com/css2?family=Quicksand:wght@300..700&display=swap");
.back {
    background-repeat: no-repeat;
    font-family: "Quicksand", sans-serif;
}
.togglebtn {
    background-color: #5a96ff;
    border-radius: 50%;
    transition: background-color ease-in 0.5s;
}
.d-flex-1 {
    display: flex;
    gap: 3%;
    justify-content: space-around;
    flex-wrap: wrap;
}

.card-effect-light {
    background-image: url("./assets/back.gif");
    background-size: cover;
    background-position-x: center;
    backdrop-filter: blur(20px);

    color: white;
    border-radius: 1rem;
    padding: 3%;
    width: fit-content;
    font-family: "Quicksand", sans-serif;
}
```

Figure 8 CSS



```


.div-effect-dark {
        background: rgba(255, 255, 255, 0.25);
        box-shadow: 0 8px 32px 0 rgba(31, 38, 135, 0.37);
        backdrop-filter: blur(4px);
        -webkit-backdrop-filter: blur(4px);
        border-radius: 10px;
        border: 1px solid rgba(255, 255, 255, 0.18);
    }

    .div-effect-light {
        background-color: #e8ffff;
        border-radius: 1rem;
    }

    .search-dark-text {
        background-color: transparent;
        border: none;
        outline: none;
        color: white;
    }

    .search-dark-text::placeholder {
        color: white;
        background-color: transparent;
    }

    .search-light-text {
        background-color: transparent;
        border: none;
        outline: none;
        color: gray;
    }

    .search-light-text::placeholder {
        color: gray;
        background-color: transparent;
    }

    .round {
        border-radius: 2rem;
    }
}



.fs-custom {
        font-size: 4rem;
    }

    .Primary {
        color: #e0e0e0;
    }

    .cursor-pointer {
        cursor: pointer;
    }

    .fs-0 {
        font-size: 0.5rem;
    }

    .card-hover {
        cursor: pointer;
    }

    .card-hover:hover {
        transform: scale(1.1);
        transition: all 0.5s;
    }

    .fixed-size-div {
        width: 100%;
        min-height: 17rem;
        border: 1px solid #ccc;
        overflow: hidden;
        background-image: url("./assets/news.gif");
        color: black;
    }

    .bg-darkB {
        background-color: #3c305e;
    }

    .bg-lightB {
        background-color: #f0f5fd;
    }

    .air-dark {
        background-color: #483b70;
        border-radius: 2rem;
    }

    .air-light {
        background-color: white;
        box-shadow: 5px 5px 10px gray;
        border-radius: 2rem;
    }

    .sunrise-dark {
        background-color: #3c305e;
        border-radius: 1rem;
        height: 100%;
    }

    .sunrise-light {
        background-color: #fff8ec;
        border-radius: 1rem;
        height: 100%;
    }
}



.right-back-dark {
        background-color: #342b52;
        max-width: fit-content;
    }

    .right-back-light {
        background-color: #fffffc;
        max-width: fit-content;
    }

    .search-dark {
        background-color: #40305d;
        border-radius: 2rem;
        max-width: 80%;
    }

    .search-light {
        background-color: #f5f8fd;
        border-radius: 2rem;
        max-width: 80%;
    }

    .toggle-dark {
        background-color: #4d3b6f;
        border-radius: 2rem;
        height: fit-content;
    }

    .toggle-light {
        background-color: #fffffc;
        border-radius: 2rem;
        height: fit-content;
    }
}



body,
    #left_back,
    #date,
    #air,
    #air_quality,
    #country_name,
    #air_info,
    #air_all,
    #sunrise_sunset,
    #sunrise_text,
    #sunrise_inside,
    #sunrise_inside1,
    #right,
    #search_bar,
    #search,
    #toggle {
        transition: color 0.5s, background-color 0.5s;
    }

    #left_back,
    #right,
    #search_bar {
        transition: background-color 0.5s;
    }

    #air_all {
        transition: color 0.5s, background-color 0.5s, box-shadow 0.5s;
    }

    input[type="search"]::-webkit-search-cancel-button,
    input[type="search"]::-webkit-search-clear-button {
        display: none;
    }
}


```

Figure 9 CSS

```

● ● ●

function getTime() {
    let currDate = new Date();
    let hrs = currDate.getHours();
    let mins = currDate.getMinutes();

    let ampm = hrs >= 12 ? "PM" : "AM";
    hrs = hrs % 12;
    hrs = hrs ? hrs : 12; //midnight case
    mins = mins < 10 ? "0" + mins : mins;

    document.getElementById("time").innerHTML = `${hrs} : ${mins} ${ampm}`;
}
window.onload = getTime();
setInterval(() => {
    getTime();
}, 1000);

(function () {
    function formatDate() {
        const currdate = new Date();
        const weekdays = [
            "Sunday",
            "Monday",
            "Tuesday",
            "Wednesday",
            "Thursday",
            "Friday",
            "Saturday",
        ];
        const months = [
            "January",
            "February",
            "March",
            "April",
            "May",
            "June",
            "July",
            "August",
            "September",
            "October",
            "November",
            "December",
        ];

        const weekday = weekdays[currdate.getDay()];
        const day = currdate.getDate();
        const month = months[currdate.getMonth()];
        const year = currdate.getFullYear();

        document.getElementById(
            "date"
        ).innerHTML = `${weekday}, ${day} ${month}, ${year}`;

        document.getElementById("today_date").innerHTML = `Today, ${day} ${month}`;
    }
    formatDate();
})

```

Figure 10 JS

Figure 11 JS

Figure 12 JS

```

var search = "";

function searchHandler(event) {
    search = event.target.value;
    console.log(search);
}

var forecast_search = [];
var forecastElement="";

async function get5DayForecastByCountry() {

    try {
        const data = await fetch(
            `https://api.openweathermap.org/data/2.5/forecast?q=${search}&appid=11d420a5264d4e5a6b6f930c3c88284d`
        );
    }

    if (!data.ok) {
        throw new Error("Error fetching forecast data: " + data.statusText);
    }

    const resp = await data.json();

    forecast_search = resp.list.filter(weather => {
        return weather.dt_txt.split(" ")[1] === "00:00:00";
    });

    console.log(forecast_search);

    const forecastContainer = document.getElementById("daysforecast");
    forecastContainer.innerHTML = "";
    forecast_search.forEach((item, index) => {
        const day = convert_to_day(item.dt_txt);
        const temp = FahrenheitToCelsius(item.main.temp);

        let weatherIconSrc;

        if (item.weather[0].main === "Clouds") {
            weatherIconSrc = "./assets/cloudy.png";
        } else if (item.weather[0].main === "Rain") {
            weatherIconSrc = "./assets/rainy.png";
        } else if (item.weather[0].main === "Snow") {
            weatherIconSrc = "./assets/snowy.png";
        } else if (item.weather[0].main === "Fog") {
            weatherIconSrc = "./assets/fog.png";
        } else if (item.weather[0].main === "Thunderstrom") {
            weatherIconSrc = "./assets/thunderstrom.png";
        } else if (item.weather[0].main === "Haze") {
            weatherIconSrc = "./assets/haze.png";
        } else if (item.weather[0].main === "Mist") {
            weatherIconSrc = "./assets/mist.png";
        } else {
            weatherIconSrc = "./assets/sunny.png";
        }

        forecastElement = document.createElement("div");
        forecastElement.classList.add("card-effect-light");
        forecastElement.classList.add("col");
        forecastElement.classList.add("fw-bold");

        forecastElement.classList.add("card-hover");

        forecastElement.innerHTML =
            `
            <div class="mt-2 ${day}</div>
            <div>${temp}<sup>c</sup></div>
            >`;
        forecastContainer.appendChild(forecastElement);
    });
}

catch (error) {
    console.error("Error:", error.message);
    const errorGif = document.createElement("div");
    errorGif.classList.add("d-flex");
    errorGif.classList.add("justify-content-around");
    errorGif.classList.add("align-items-center");
    errorGif.classList.add("fs-1");
    errorGif.classList.add("fw-bold");
    errorGif.innerHTML = '<p>404 | Not Found</p>
    ';
    forecastContainer = document.getElementById("daysforecast");
    forecastContainer.innerHTML = "" // Clear the container
    forecastContainer.appendChild(errorGif); // Append the GIF
}
}

```

Figure 13 Js

```

function performSearch(event) {
    event.preventDefault();

    const forecastContainer = document.getElementById("daysforecast");
    forecastContainer.innerHTML = "";
    get5DayForecastByCountry();
    console.log("running next api");
    currentWeatherByCountry();
}

async function airpollution() {
    const resp = await fetch(
        `http://api.openweathermap.org/data/2.5/air_pollution?lat=${lat}&lon=${long}&appid=11d420a5264d4e5a6b6f930c3c88284d`
    );
    const res = await resp.json();
    console.log(res);
    console.log(res.list[0].main.aqi);
    const aqi = res.list[0].main.aqi;
    document.getElementById("country_name").innerHTML = `${cityname}`;

    if (aqi >= 0 && aqi <= 50) {
        document.getElementById("air_type").innerHTML = "good";
        document.getElementById("air_info").innerHTML = "Clear skies, stride-wise";
    } else if (aqi > 50 && aqi <= 100) {
        document.getElementById("air_type").innerHTML = "Moderate";
        document.getElementById("air_info")
            .innerHTML = "Balanced air, walk with care";
    } else if (aqi > 100 && aqi <= 150) {
        document.getElementById("air_type").innerHTML = "Unhealthy";
        document.getElementById("air_info")
            .innerHTML = "Caution in air, tread with care";
    } else {
        document.getElementById("air_type").innerHTML = "Hazardous";
        document.getElementById("air_info")
            .innerHTML = "Air's a concern, indoor's your turn.";
    }

    const pm2 = res.list[0].components.pm2_5;
    document.getElementById("pm2_value").innerHTML = `${pm2}`;

    const pm10 = res.list[0].components.pm10;
    document.getElementById("pm10_value").innerHTML = `${pm10}`;

    const so2 = res.list[0].components.so2;
    document.getElementById("so2_value").innerHTML = `${so2}`;

    const no2 = res.list[0].components.no2;
    document.getElementById("no2_value").innerHTML = `${no2}`;

    const o3 = res.list[0].components.o3;
    document.getElementById("o3_value").innerHTML = `${o3}`;

    const co = res.list[0].components.co;
    document.getElementById("co_value").innerHTML = `${co}`;

    Function convert() {
        const militaryTime = "18:51";
        // Extract hours and minutes from military time
        const [hours, minutes] = militaryTime.split(":").map(Number);

        // Convert military time to 12-hour format
        let period = "PM";
        let twelveHour = hours;

        if (twelveHour === 0) {
            twelveHour = 12;
        } else if (twelveHour > 12) {
            twelveHour -= 12;
        }

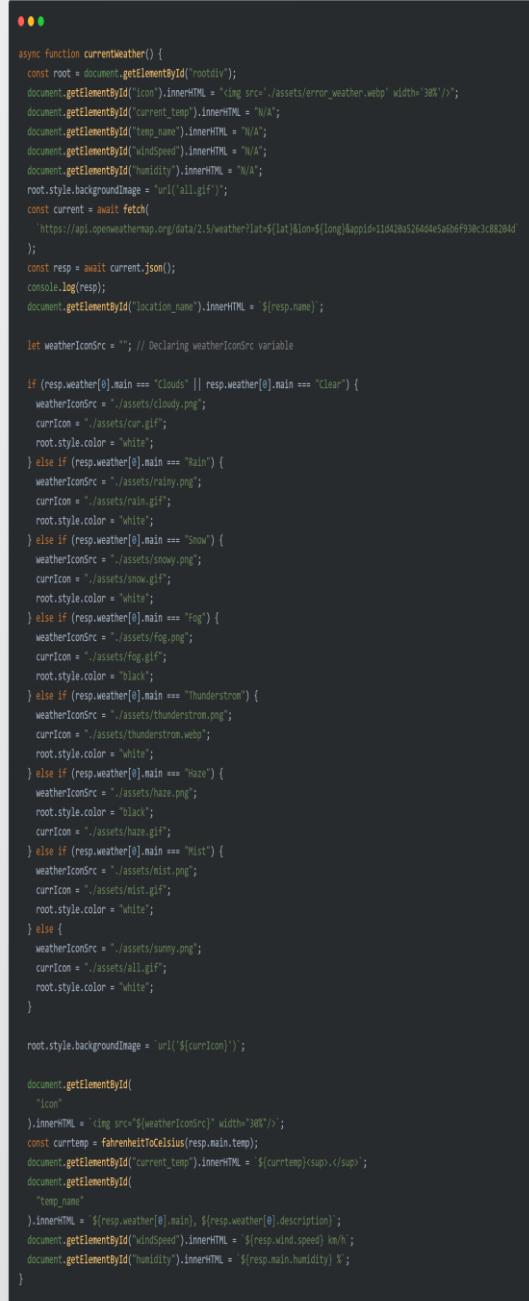
        const twelveHourTime = `${twelveHour}:${minutes}
        .toString()
        .padStart(2, '0') ${period}`;
        return twelveHourTime;
    }
}

```

Figure 14 JS

Figure 15 JS

Figure 16 JS



```

async function currentWeather() {
    const root = document.getElementById("rootdiv");
    document.getElementById("icon").innerHTML = "<img src='./assets/error_weather.webp' width='30%'>";
    document.getElementById("current_temp").innerHTML = "N/A";
    document.getElementById("temp_name").innerHTML = "N/A";
    document.getElementById("windSpeed").innerHTML = "N/A";
    document.getElementById("humidity").innerHTML = "N/A";
    root.style.backgroundImage = "url('all.gif')";
    const current = await fetch(
        "https://api.openweathermap.org/data/2.5/weather?lat=${lat}&lon=${long}&appid=1642b5264de5a0bf930c3c88284d"
    );
    const resp = await current.json();
    console.log(resp);
    document.getElementById("location_name").innerHTML = `${resp.name}`;

    let weatherIconSrc = ""; // Declaring weatherIconSrc variable

    if (resp.weather[0].main === "Clouds" || resp.weather[0].main === "Clear") {
        weatherIconSrc = "./assets/cloudy.png";
        currIcon = "./assets/cntr.gif";
        root.style.color = "white";
    } else if (resp.weather[0].main === "Rain") {
        weatherIconSrc = "./assets/rainy.png";
        currIcon = "./assets/rain.gif";
        root.style.color = "white";
    } else if (resp.weather[0].main === "Snow") {
        weatherIconSrc = "./assets/snowy.png";
        currIcon = "./assets/snow.gif";
        root.style.color = "white";
    } else if (resp.weather[0].main === "Fog") {
        weatherIconSrc = "./assets/fog.png";
        currIcon = "./assets/fog.gif";
        root.style.color = "black";
    } else if (resp.weather[0].main === "Thunderstron") {
        weatherIconSrc = "./assets/thunderstron.png";
        currIcon = "./assets/thunderstron.webp";
        root.style.color = "white";
    } else if (resp.weather[0].main === "Haze") {
        weatherIconSrc = "./assets/haze.png";
        currIcon = "./assets/haze.gif";
        root.style.color = "black";
    } else if (resp.weather[0].main === "Mist") {
        weatherIconSrc = "./assets/mist.png";
        currIcon = "./assets/mist.gif";
        root.style.color = "white";
    } else {
        weatherIconSrc = "./assets/sunny.png";
        currIcon = "./assets/all.gif";
        root.style.color = "white";
    }

    root.style.backgroundImage = `url('${currIcon}')`;

    document.getElementById(
        "icon"
    ).innerHTML = "<img src='${weatherIconSrc}' width='30%'>";
    const currtemp = FahrenheitToCelsius(resp.main.temp);
    document.getElementById("current_temp").innerHTML = `${currtemp}<sup>°C</sup>`;
    document.getElementById(
        "temp_name"
    ).innerHTML = `${resp.weather[0].main}, ${resp.weather[0].description}`;
    document.getElementById("windSpeed").innerHTML = `${resp.wind.speed} km/h`;
    document.getElementById("humidity").innerHTML = `${resp.main.humidity} %`;
}

```

Figure 17 JS

FEE-II, 22CS0014



```

.front4 {
    display: flex;
    background-color: white;
    max-width: 100vw;
    justify-content: end;
    align-items: center;
    font-family: "Poppins", sans-serif;
    gap: 50px;
    padding: 5%;
    box-sizing: border-box;
    flex-direction: column;
    overflow-x: hidden;
}

.shape2 {
    width: 286px;
    height: 260px;
    background-color: rgba(218, 172, 0, 0.5686274509803921);
    z-index: 10;
    background-image: url("assets/docear.svg");
    position: relative;
    display: flex;
    justify-content: center;
    align-items: center;
    background-size: 100%;
}

.front4right {
    display: flex;
    flex-direction: column;
    justify-content: end;
    align-items: end;
    text-align: right;
    max-width: 100%;
    gap: 10px;
}

.front4head{
    font-size: 200.5%; font-weight: 600
}

.front4text{
    font-size: 121.25%; font-weight: 400
}

```

Figure 18 CSS- Advertisement Section

```

var lat_1="";  
var lon_1="";  
var air_pollution_city="";  
  
async function currentWeatherByCountry() {  
    const root = document.getElementById("rootdiv");  
    document.getElementById("icon").innerHTML = <img src='./assets/error_weather.webp' width="300"/>;  
    document.getElementById("location_name").innerHTML= "N/A";  
    document.getElementById("current_temp").innerHTML = "N/A";  
    document.getElementById("temp_name").innerHTML = "N/A";  
    document.getElementById("windSpeed").innerHTML = "N/A";  
    document.getElementById("humidity").innerHTML = "N/A";  
  
    try {  
        const current = await fetch(  
            "https://api.openweathermap.org/data/2.5/weather?q=$search&appid=11d4205264de5a6b6f930c3c88284d"  
        );  
  
        if (!current.ok) {  
            throw new Error('Network response was not ok');  
        }  
  
        const resp = await current.json();  
        console.log('current athen');  
        console.log(resp);  
        lat_1=resp.coord.lat;  
        lon_1=resp.coord.lon  
        air_pollution_city=resp.name;  
        document.getElementById("location_name").innerHTML = $(resp.name);  
  
        let weatherIconSrc = "";  
        let currIcon = "url('./all.gif')";  
        if (resp.weather[0].main === "Clouds" || resp.weather[0].main === "Clear") {  
            weatherIcons = "./assets/cloudy.png";  
            currIcon = "./assets/cu.gif";  
            root.style.color = "white";  
        } else if (resp.weather[0].main === "Rain") {  
            weatherIconSrc = "./assets/rainy.png";  
            currIcon = "./assets/rain.gif";  
            root.style.color = "white";  
        } else if (resp.weather[0].main === "Snow") {  
            weatherIconSrc = "./assets/snowy.png";  
            currIcon = "./assets/snow.gif";  
            root.style.color = "white";  
        } else if (resp.weather[0].main === "Fog") {  
            weatherIconSrc = "./assets/rog.png";  
            currIcon = "./assets/fog.gif";  
            root.style.color = "black";  
        } else if (resp.weather[0].main === "Thunderstrom") {  
            weatherIconSrc = "./assets/thunderstrom.png";  
            currIcon = "./assets/thunderstrom.webp";  
            root.style.color = "white";  
        } else if (resp.weather[0].main === "Haze") {  
            weatherIconSrc = "./assets/haze.png";  
            root.style.color = "black";  
            currIcon = "./assets/haze.gif";  
        } else if (resp.weather[0].main === "Mist") {  
            weatherIconSrc = "./assets/mist.png";  
            currIcon = "./assets/mist.gif";  
            root.style.color = "white";  
        } else {  
            weatherIconSrc = "./assets/sunny.png";  
            root.style.color = "white";  
        }  
  
        root.style.backgroundImage = `url('${currIcon}')`;  
  
        document.getElementById(  
            "icon"  
        ).innerHTML = ;  
        const currtemp = FahrenheitToCelsius(resp.main.temp);  
        document.getElementById("current_temp").innerHTML = `${currtemp}°C`;  
        document.getElementById(  
            "temp_name"  
        ).innerHTML = `${resp.weather[0].main}, ${resp.weather[0].description}`;  
        document.getElementById("windSpeed").innerHTML = `${resp.wind.speed} km/h`;  
        document.getElementById("humidity").innerHTML = `${resp.main.humidity} %`;  
    } catch (error) {  
        root.style.color="#FADECE";  
        root.style.backdropFilter = "blur(10px)";  
        root.style.backgroundImage = `url('./assets/all.webp')`;  
    }  
  
    airpollutionByCountry();  
    sunriseAndsunsetByCountry();  
}

```

Figure 19 JS

```

async function airpollutionByCountry() {  
    const resp = await fetch(  
        "http://api.openweathermap.org/data/2.5/air_pollution?lat=$[lat_1]&lon=$[lon_1]&appid=11d4205264de5a6b6f930c3c88284d"  
    );  
  
    const res = await resp.json();  
    console.log("airpollutant")  
    console.log(res);  
  
    console.log(res.list[0].main.aqi);  
    const aqi = res.list[0].main.aqi;  
    document.getElementById("country_name").innerHTML = $(air_pollution_city);  
  
    if (aqi >= 0 && aqi <= 50) {  
        document.getElementById("air_type").innerHTML = "good";  
        document.getElementById("air_info").innerHTML = "Clear skies, stride with care";  
    } else if (aqi > 50 && aqi <= 100) {  
        document.getElementById("air_type").innerHTML = "moderate";  
        document.getElementById("air_info").innerHTML = "Balanced air, walk with care";  
    } else if (aqi > 100 && aqi <= 150) {  
        document.getElementById("air_type").innerHTML = "Unhealthy";  
        document.getElementById("air_info").innerHTML = "Caution in air, tread with care";  
    } else {  
        document.getElementById("air_type").innerHTML = "hazardous";  
        document.getElementById("air_info").innerHTML = "Air's a concern, indoor's your turn.";  
    }  
  
    const pm2 = res.list[0].components.pm2.5;  
    document.getElementById("pm2_value").innerHTML = `${pm2}`;  
  
    const pm10 = res.list[0].components.pm10;  
    document.getElementById("pm10_value").innerHTML = `${pm10}`;  
  
    const no2 = res.list[0].components.no2;  
    document.getElementById("no2_value").innerHTML = `${no2}`;  
  
    const o3 = res.list[0].components.o3;  
    document.getElementById("o3_value").innerHTML = `${o3}`;  
  
    const co = res.list[0].components.co;  
    document.getElementById("co_value").innerHTML = `${co}`;  
  
    const sunrise = timeConverter(res.sys.sunrise);  
    document.getElementById("sunrise_time").innerHTML = `${sunrise} AM`;  
  
    const sunset = timeConverter(res.sys.sunset);  
    document.getElementById("sunset_time").innerHTML = `${sunset} PM`;  
  
    var count = 0;  
  
    async function getnews() {  
        const res = await fetch("https://newsapi.org/v2/top-headlines?country=id&apiKey=c13088074647de5e28187e608821");  
        const news = await res.json();  
        return news.articles;  
    }  
  
    async function displayNews() {  
        const title = document.getElementById("news_title"); // Get reference to title element  
        const des = document.getElementById("news_description"); // Get reference to description element  
        const newsCount = news.length;  
  
        const carouseInterval = setInterval(() => {  
            title.innerHTML = `${news[count].title}`;  
            if(news[count].description!=null){  
                des.innerHTML = `...`;  
            } else{  
                des.innerHTML = `${news[count].description}`;  
            }  
  
            count++;  
            if(count == newsCount){  
                clearInterval(carouseInterval);  
                count = 0;  
            }  
        }, 5000);  
    }  
  
    displayNews();

```

Figure 20 JS

DESKTOP VIEW OUTPUT :-

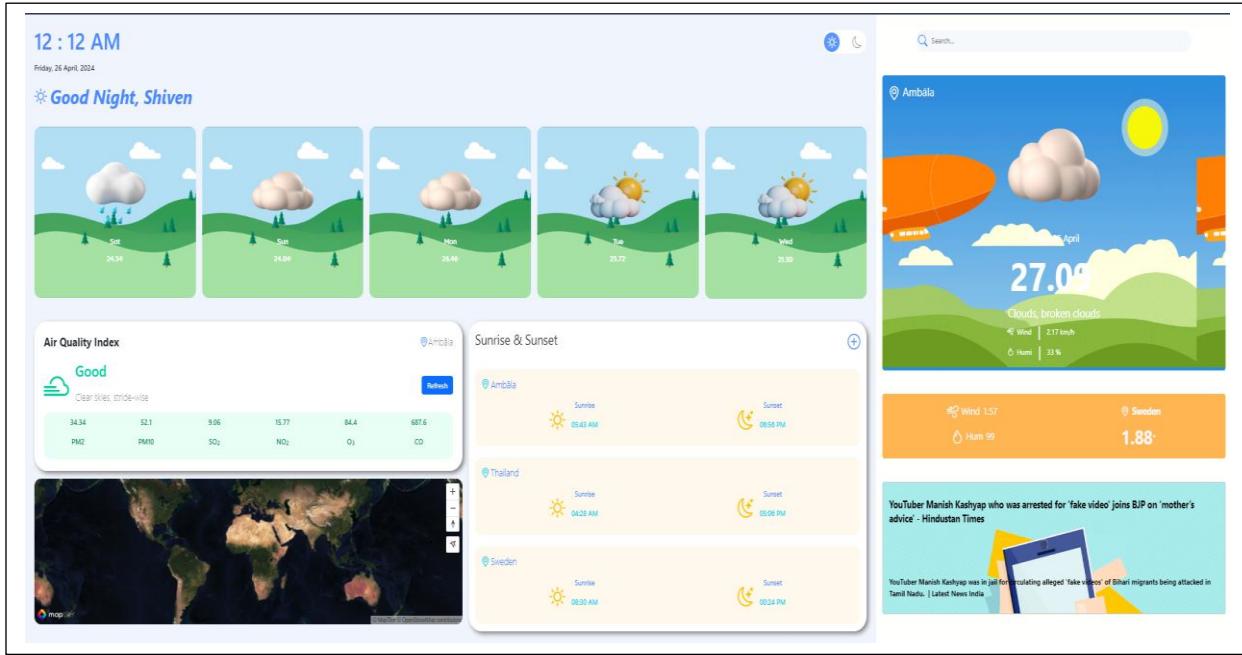


FIGURE 21 OUTPUT-HOME SECTION

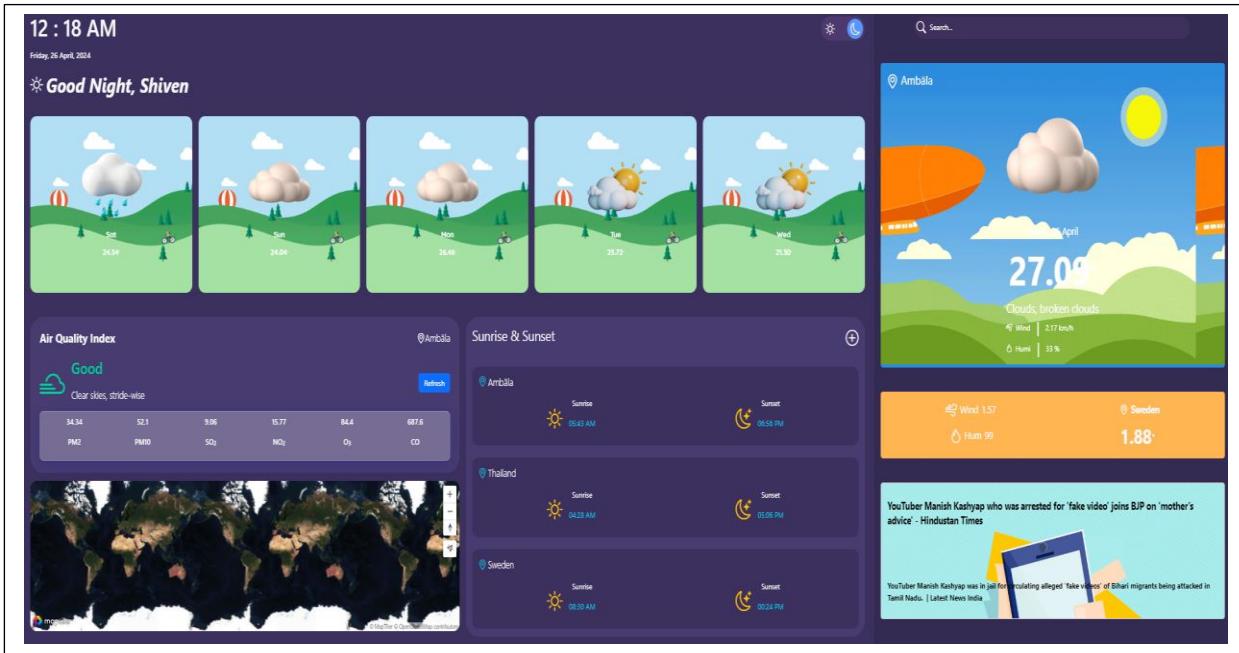


FIGURE 21.A OUTPUT-HOME SECTION(DARK-MODE)

MOBILE VIEW OUTPUT :-

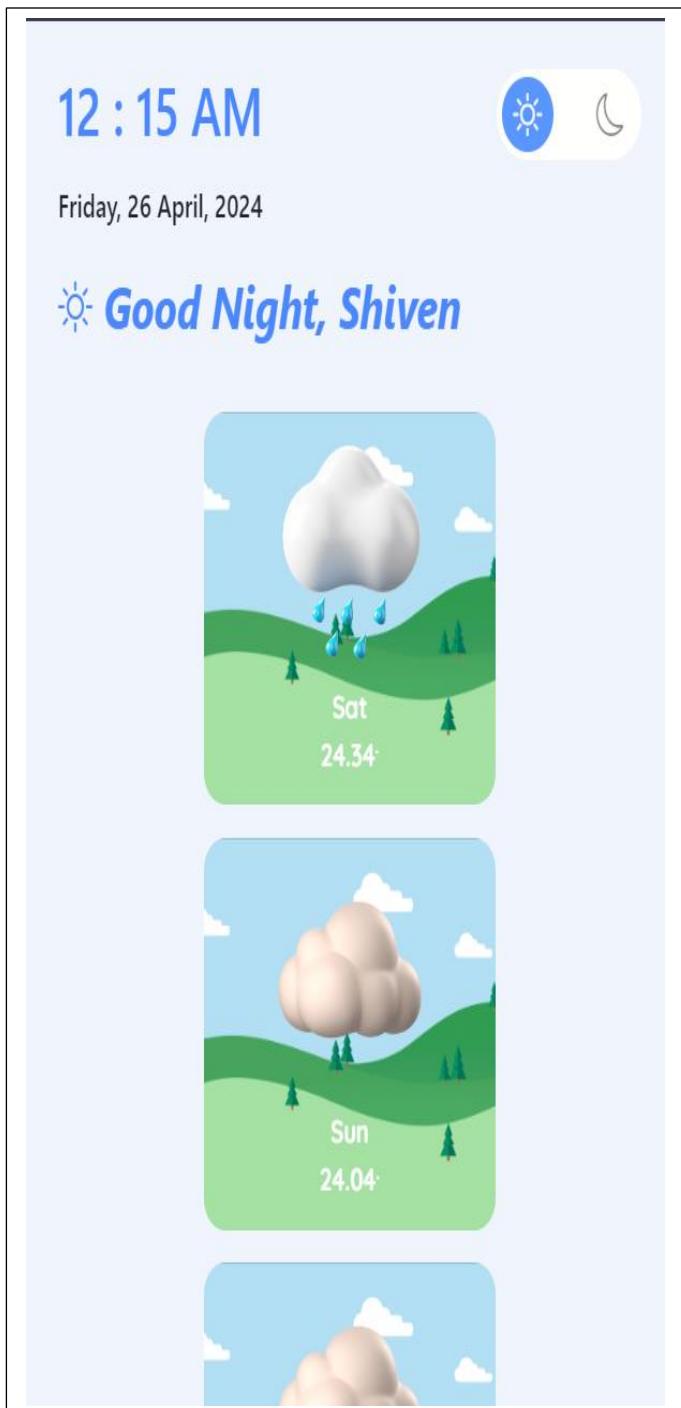


FIGURE 22 OUTPUT-HOME SECTION

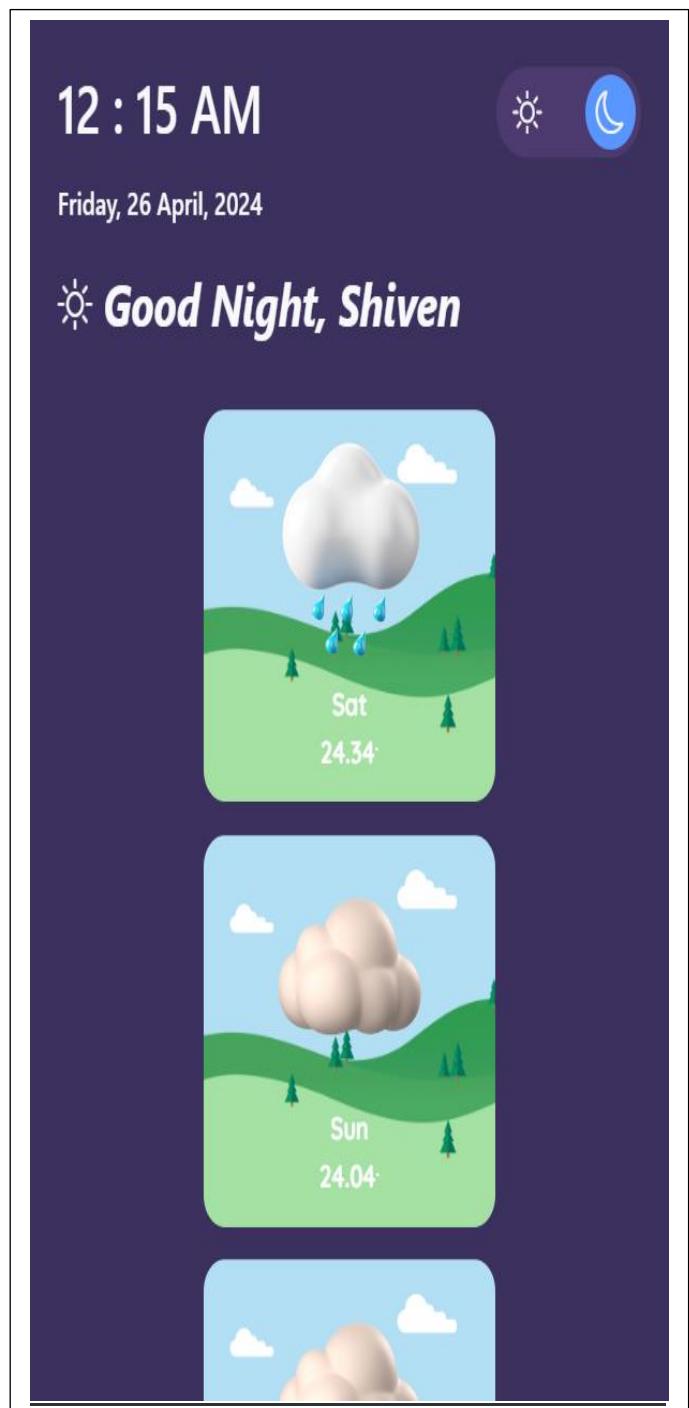


FIGURE 22.A OUTPUT-HOME SECTION(DARK)

MOBILE VIEW OUTPUT :-

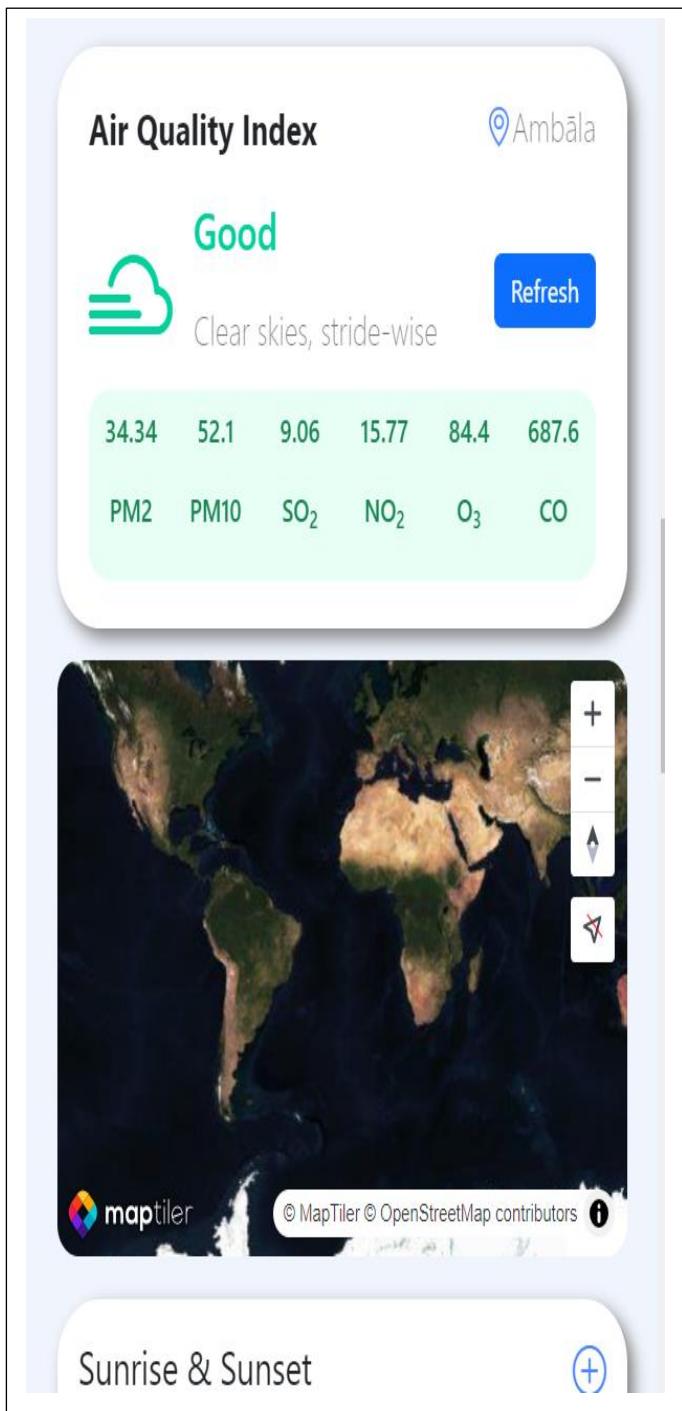


FIGURE 23 OUTPUT-HOME SECTION

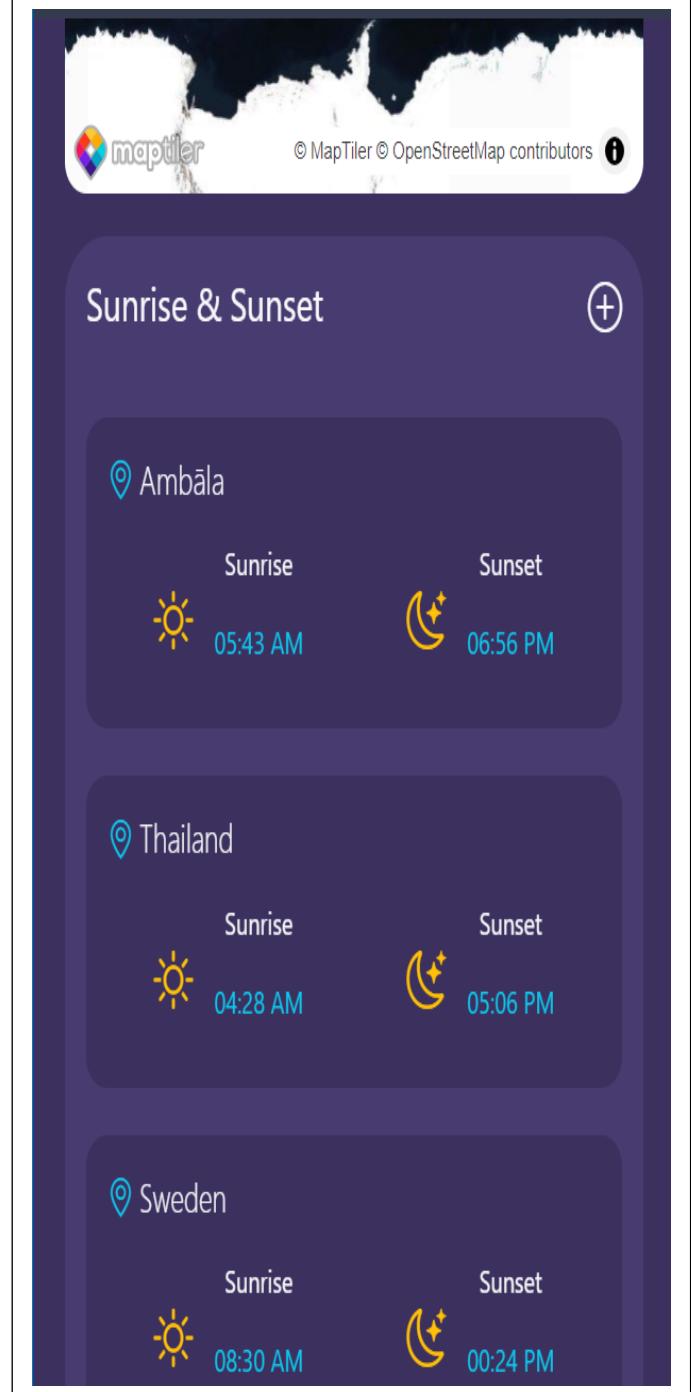


FIGURE 23.A OUTPUT-HOME SECTION(DARK)

MOBILE VIEW OUTPUT :-

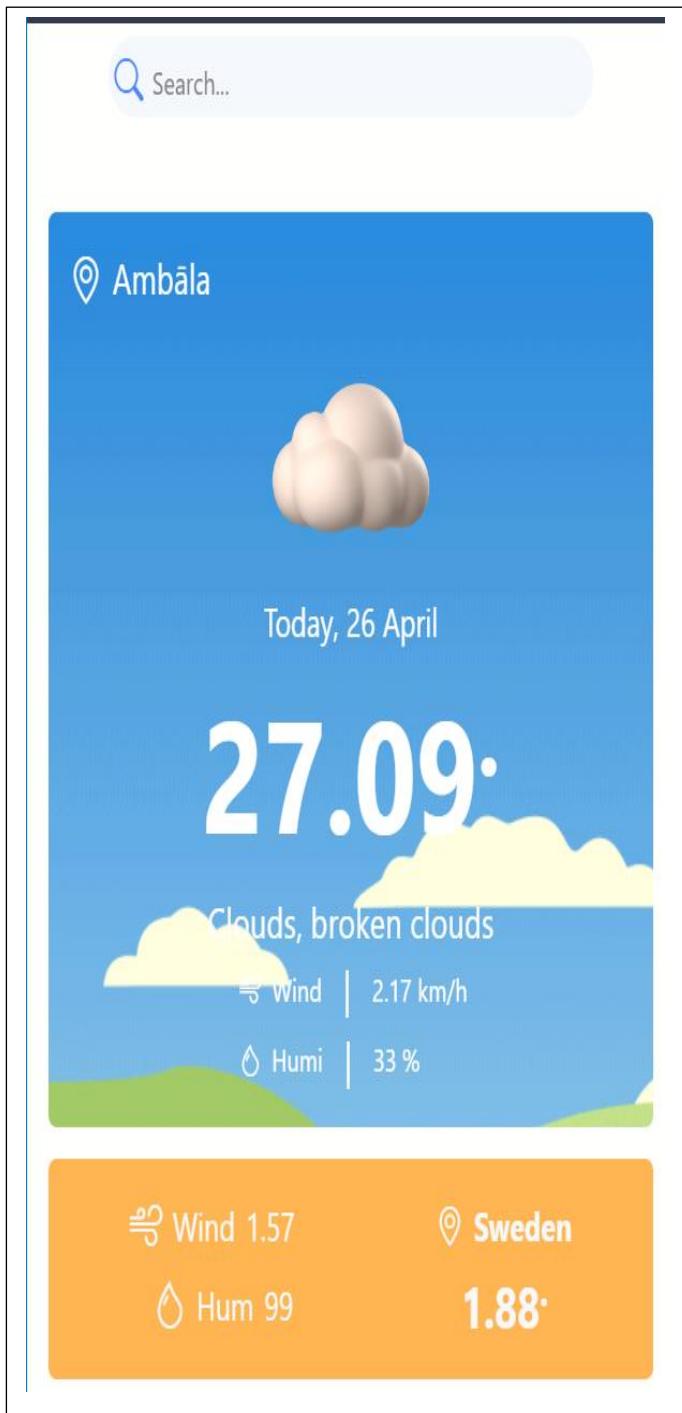


FIGURE 24 OUTPUT-HOME SECTION

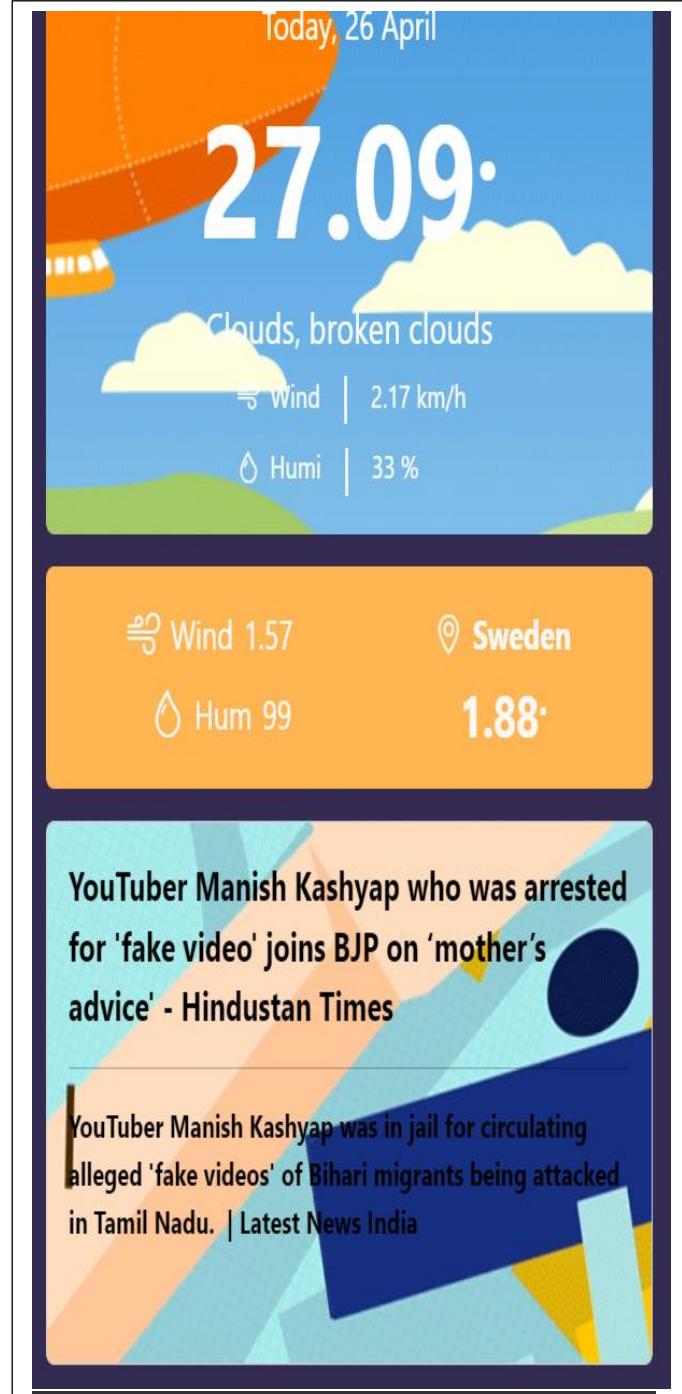


FIGURE 24.A OUTPUT-HOME SECTION(DARK)

References

1. OpenWeatherMap API
2. Bootstrap Documentation
3. Dribble

