SHINY APP FOR WEATHER TRENDS

A PROJECT REPORT

Submitted by

TAMILDEEPAA A(2303811724322116)

in partial fulfillment of requirements for the award of the course

AGI1252 - FUNDAMENTALS OF DATA SCIENCE USING R

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

JUNE- 2025

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY (AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report on **"SHINY APP FOR WEATHER TRENDS"** is the bonafide work of **TAMILDEEPAA A(2303811724322116)** who carried out the project work during the academic year 2024 - 2025 under my supervision.

SIGNATURE

Dr.T. AVUDAIAPPAN, M.E.,Ph.D.,

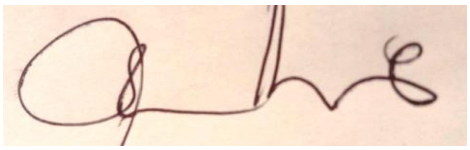
HEAD OF THE DEPARTMENT

PROFESSOR

Department of Artificial Intelligence

K.Ramakrishnan College of Technology (Autonomous)

Samayapuram–621112.



SIGNATURE

Ms.S.Murugavalli., M.E.,(Ph.D).,

SUPERVISOR

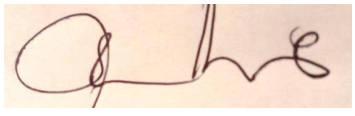
ASSISTANT PROFESSOR

Department of Artificial Intelligence

K.Ramakrishnan College of Technology (Autonomous)

Samayapuram–621112.

Submitted for the viva-voce examination held on ……**02.06.2025**.



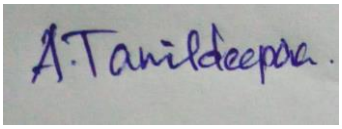
INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARATION

I declare that the project report on **"SHINY APP FOR WEATHER TRENDS"** is the result of original work done by me and best of our knowledge, similar work has not been submitted to **"ANNA UNIVERSITY CHENNAI"** for the requirement of Degree of **BACHELOR OF TECHNOLOGY**. This project report is submitted on the partial fulfilment of the requirement of the completion of the course **AGI1252 - FUNDAMENTALS OF DATA SCIENCE USING R**

.



TAMILDEEPAA A

Place: Samayapuram

Date:02.06.2025

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and in-debt to our institution "**K.Ramakrishnan College of Technology (Autonomous)**", for providing us with the opportunity to do this project.

I glad to credit honourable chairman **Dr. K. RAMAKRISHNAN**, **B.E.,** for having provided for the facilities during the course of our study in college.

I would like to express my sincere thanks to our beloved Executive Director **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding to my project and offering adequate duration in completing our project.

I would like to thank **Dr. N. VASUDEVAN, M.Tech., Ph.D.,** Principal, who gave opportunity to frame the project the full satisfaction.

I whole heartily thanks to **Dr. T. AVUDAIAPPAN, M.E.,Ph.D.,** Head of the department, **ARTIFICIAL INTELLIGENCE** for providing his encourage pursuing this project.

I express our deep expression and sincere gratitude to my project supervisor **Ms.S.Murugavalli., M.E.,(Ph.D).,** Department of **ARTIFICIAL INTELLIGENCE,** for her incalculable suggestions, creativity, assistance and patience which motivated me to carry out this project.

I render my sincere thanks to Course Coordinator and other staff members for providing valuable information during the course.

I wish to express my special thanks to the officials and Lab Technicians of my departments who rendered their help during the period of the work progress.

INSTITUTE

Vision:

- To serve the society by offering top-notch technical education on par with global standards.

Mission:

- Be a center of excellence for technical education in emerging technologies by exceeding the needs of industry and society.
- Be an institute with world class research facilities.
- Be an institute nurturing talent and enhancing competency of students to transform them as all – round personalities respecting moral and ethical values.

DEPARTMENT

Vision:

- To excel in education, innovation, and research in Artificial Intelligence and Data Science to fulfil industrial demands and societal expectations.

Mission

- To educate future engineers with solid fundamentals, continually improving teaching methods using modern tools.
- To collaborate with industry and offer top-notch facilities in a conducive learning environment.
- To foster skilled engineers and ethical innovation in AI and Data Science for global recognition and impactful research.
- To tackle the societal challenge of producing capable professionals by instilling employability skills and human values.

PROGRAM EDUCATIONAL OBJECTIVES (PEO)

- **PEO1:** Compete on a global scale for a professional career in Artificial Intelligence and Data Science.
- **PEO2:** Provide industry-specific solutions for the society with effective communication and ethics.
- **PEO3** Enhance their professional skills through research and lifelong learning initiatives.

PROGRAM SPECIFIC OUTCOMES (PSOs)

- **PSO1:** Capable of finding the important factors in large datasets, simplify the data, and improve predictive model accuracy.
- **PSO2:** Capable of analyzing and providing a solution to a given real-world problem by designing an effective program.

PROGRAM OUTCOMES (POs)

Engineering students will be able to:

1. **Engineering knowledge:** Apply knowledge of mathematics, natural science, computing, engineering fundamentals, and an engineering specialization to develop solutions to complex engineering problems.

2. **Problem analysis:** Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development.

3. **Design/development of solutions:** Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required.

4. **Conduct investigations of complex problems:** Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions.

5. **Engineering Tool Usage:** Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems.

6. **The Engineer and The World:** Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment.

7. **Ethics:** Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws.

8. **Individual and Collaborative Team work:** Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

9. **Communication:** Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences.

10. **Project management and finance:** Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

11. **Life-long learning:** Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change.

ABSTRACT

The Shiny App For Weather Trends involves the development of a modular Shiny web application to visualize weather trends, specifically temperature and humidity, for ten globally recognized cities using simulated data. The application is structured to follow best practices in modular programming by separating key components—such as UI layout, server logic, and plotting functionality—into independent, reusable modules. This enhances code clarity, simplifies maintenance, and facilitates scalability for future extensions. The weather data is generated programmatically to simulate realistic conditions, allowing the app to function entirely offline without the need for external APIs or internet connectivity. Users can interactively select a city and instantly view line plots of temperature and humidity trends over a 60-hour period.

ABSTRACT WITH POs AND PSOs MAPPING

CO 5 : BUILD DATA SCIENCE USING R PROGRAMMING FOR SOLVING REAL-TIME PROBLEMS.

ABSTRACT	POs MAPPED	PSOs MAPPED
This project presents a modular Shiny web application developed using R to visualize weather trends-temperature and humidity-across 10 global cities. By programmatically generating realistic datasets, the application operates offline and offers interactive visualization features. The code is structured using modular components for plotting and data processing, promoting scalability, maintainability, and reusability. This design helps students understand how to handle data-driven real-time problems and build functional, database-like applications in a simulated environment. The project maps directly to CO5 by demonstrating how to organize, manipulate, and visualize structured datasets interactively.	**PO1 -3** **PO2 -3** **PO3 -3** **PO5 -3** **PO9 -3**	**PSO1 -3** **PSO2 -3**

Note: 1- Low, 2-Medium, 3- High

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 Introduction To Project

With the growing importance of data-driven decision-making, the ability to build interactive and modular applications has become a critical skill in solving real-time problems. This project introduces a Shiny-based web application developed in R for visualizing weather trends such as temperature and humidity across multiple global cities. By leveraging simulated datasets, the application demonstrates how to organize, process, and display time-series weather data interactively.

1.2 Objective

The objective of this project is to develop a modular Shiny web application that simulates and visualizes weather trends such as temperature and humidity for multiple global cities. The application is designed to function offline using synthetically generated data, mimicking real-time weather datasets. It aims to demonstrate how structured data can be managed, processed, and displayed in an interactive web environment. By employing modular programming practices, the project ensures improved code organization, scalability, and reusability. Through this implementation, the project supports the understanding and application of database concepts and reactive programming techniques in solving real-time problems.

1.3 Overview

- To develop a modular Shiny web application that visualizes weather trends (temperature and humidity) for 10 cities using simulated data.
- To demonstrate structured data handling and real-time visualization using R and reactive programming.
- To apply modular programming principles for enhanced code clarity, reusability, and scalability.
- To simulate a database-driven solution for solving real-time problems, aligning with CO5 learning outcomes.

1.4 Data Science related concepts

1. **Data Simulation**

- Since real-time API data is not used, the project simulates realistic temperature and humidity datasets. This teaches the importance of generating synthetic data for testing and prototyping models or applications.

2. **Time Series Data**

- The weather data includes timestamps, allowing users to analyze trends over time. Handling and visualizing time-series data is a critical concept in data science, especially in forecasting and trend analysis.

3. **Data Preprocessing**

- The app processes simulated data by filtering it based on city selection. This mimics real-world scenarios where raw data must be cleaned and subsetted before use.

4. **Data Visualization**

- Visualization is a core part of data science. This project uses ggplot2 to create clear, interactive line charts to help users explore trends and patterns effectively.

5. **Reactive Programming**

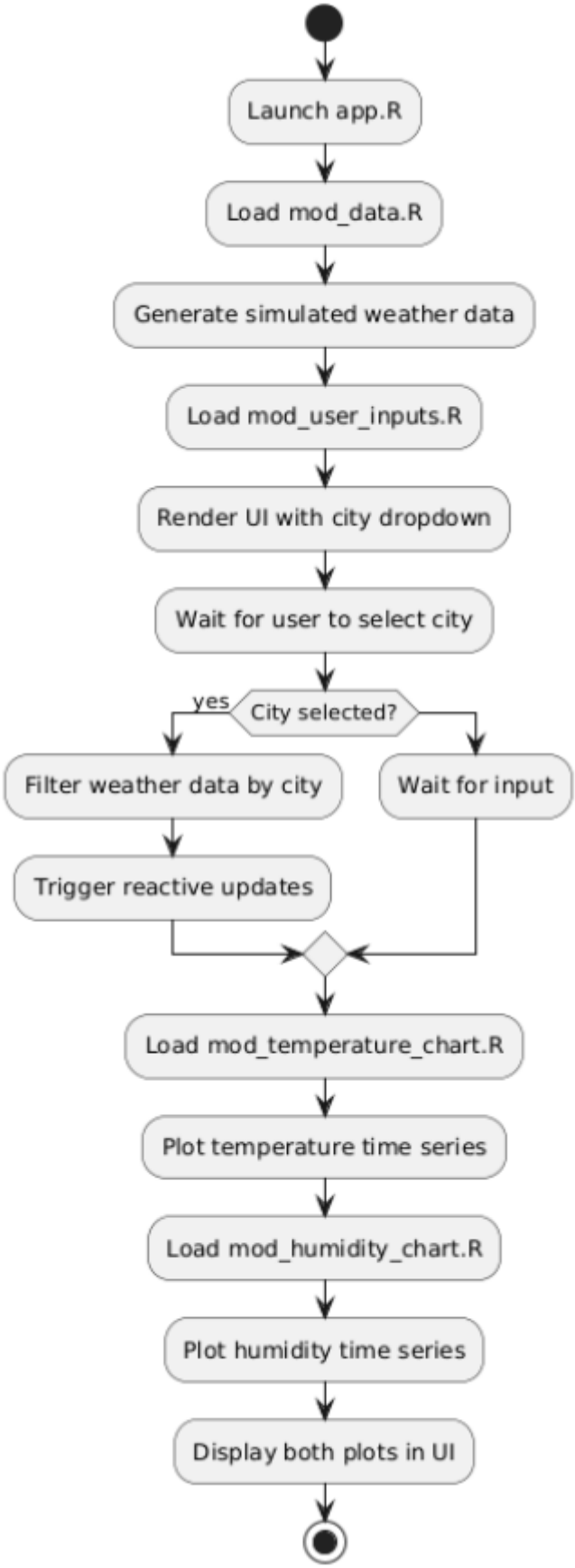
- Shiny's reactive model allows automatic updates to plots and outputs based on user inputs, demonstrating how interactivity can enhance data applications.

CHAPTER 2
PROJECT METHODOLOGY

2.1 Proposed Work

The proposed work involves the design and development of a modular Shiny web application that simulates and visualizes weather trends, specifically temperature and humidity, across ten global cities. The application will use R for data generation, processing, and visualization. The project will be structured using modular programming techniques, separating the user interface, server logic, and plotting components into distinct modules to ensure maintainability and scalability. The application will allow users to select a city and view corresponding weather trends in the form of line plots. Since the data is synthetically generated, the app will be capable of running offline without dependence on external APIs. The use of reactive programming will enable real-time interactivity in data display. The proposed solution serves as a demonstration of how simulated datasets can be treated like real-time databases to build meaningful and interactive applications for data analysis and problem-solving.

2.2 BLOCK DIAGRAM



CHAPTER 3

MODULE DESCRIPTION

3.1 Data Generation and Management Module

This module is responsible for generating the simulated weather dataset used in the application. It creates time-series data for 10 major cities, including temperature, humidity, datetime, and weather conditions. The data mimics real-time sensor input using random values and serves as the app's core database. This module ensures the app can function offline while still reflecting realistic weather behavior. The dataset is also filtered here based on user selection, ensuring efficiency in rendering outputs.

3.2 User Input Control Module

This module defines the user interface elements used to interact with the application. It includes components like city selection dropdowns and action buttons that trigger updates in the visualizations. Structuring input controls as a module promotes reusability and makes it easier to maintain or extend the UI—for example, adding more filters or input types in the future.

3.3 Temperature Chart Module

This module generates an interactive time-series plot for temperature trends. It receives filtered weather data and uses ggplot2 to render a line chart that reflects the temperature changes for the selected city over a given period. By encapsulating this logic in a module, the temperature chart becomes independent, reusable, and easier to modify or replace.

3.4 Humidity Chart Module

This module functions similarly to the temperature chart module but focuses on displaying humidity trends. It takes the same filtered dataset and renders a humidity-specific plot, enabling users to compare changes in humidity over time. Keeping it as a separate module ensures better code separation and allows for independent improvements (e.g., adding average humidity lines or thresholds).

3.5 Main Application Controller

This is the main application file that ties everything together. It sources all the modules and integrates them into a cohesive UI and server structure. The **app.R** script coordinates data flow between user input and chart modules, handles the layout using fluidPage, and runs the application using shinyApp(ui, server). It serves as the backbone of the project and is essential for launching the full application.

CHAPTER 4

CONCLUSION

This Shiny App for Weather Trends successfully demonstrates the development of a modular Shiny web application that visualizes simulated weather data for ten global cities. By using R programming and the Shiny framework, the application showcases real-time trends in temperature and humidity through interactive charts. The use of modular design separating data generation, user input, and plotting logic ensures better code readability, reusability, and scalability. The application offers an interactive user interface that allows users to select a city and view corresponding weather trends through dynamic plots. It demonstrates effective use of reactive programming and structured data handling within a web-based framework. By simulating real-world weather conditions and displaying them through interactive graphics, the project provides a practical example of how databases and data visualization can be applied to solve real-time problems.

FUTURE SCOPE

The current version of the project demonstrates weather trend visualization using simulated data in a modular Shiny application. However, several enhancements can be made to extend its functionality and real-world applicability. In future iterations, the application can be integrated with live weather APIs (such as OpenWeatherMap or Weatherstack) to fetch real-time data instead of using simulated values. This would make the app suitable for practical use in forecasting and monitoring systems.Additional weather parameters like wind speed, rainfall, and air quality index can also be included to provide more comprehensive environmental insights. Support for storing historical data in databases (e.g., SQLite or PostgreSQL) would enable long-term trend analysis.

APPENDICES

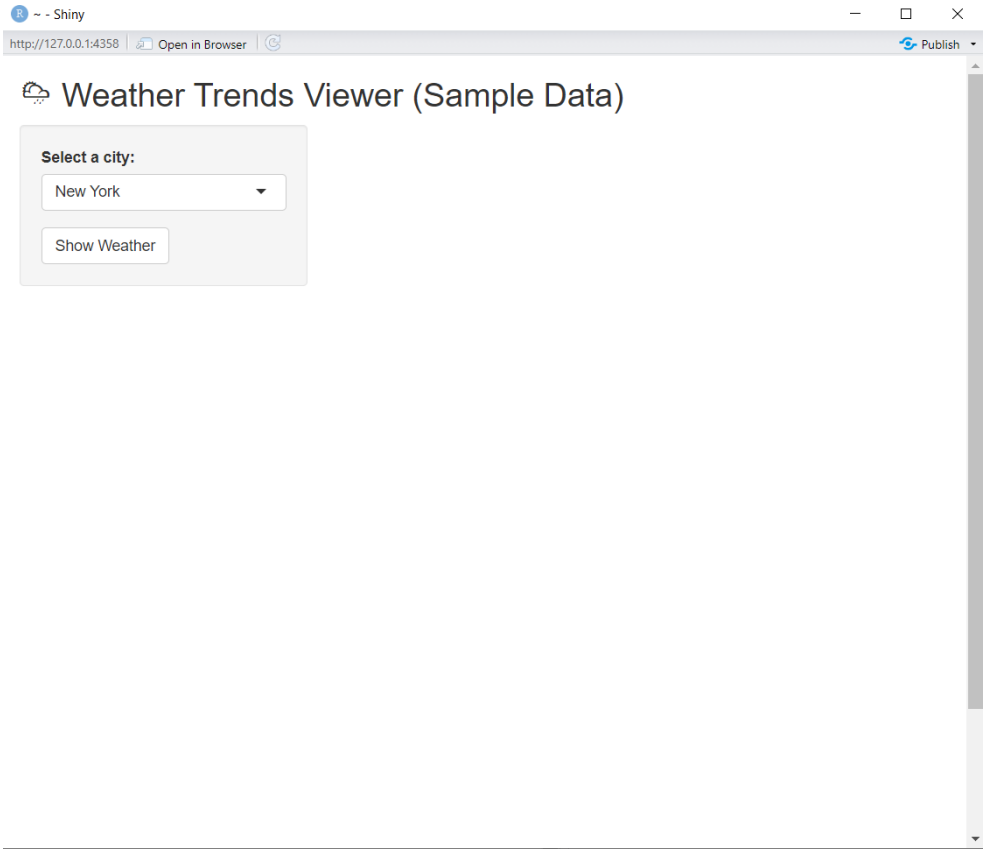
APPENDIX A–SOURCE CODE

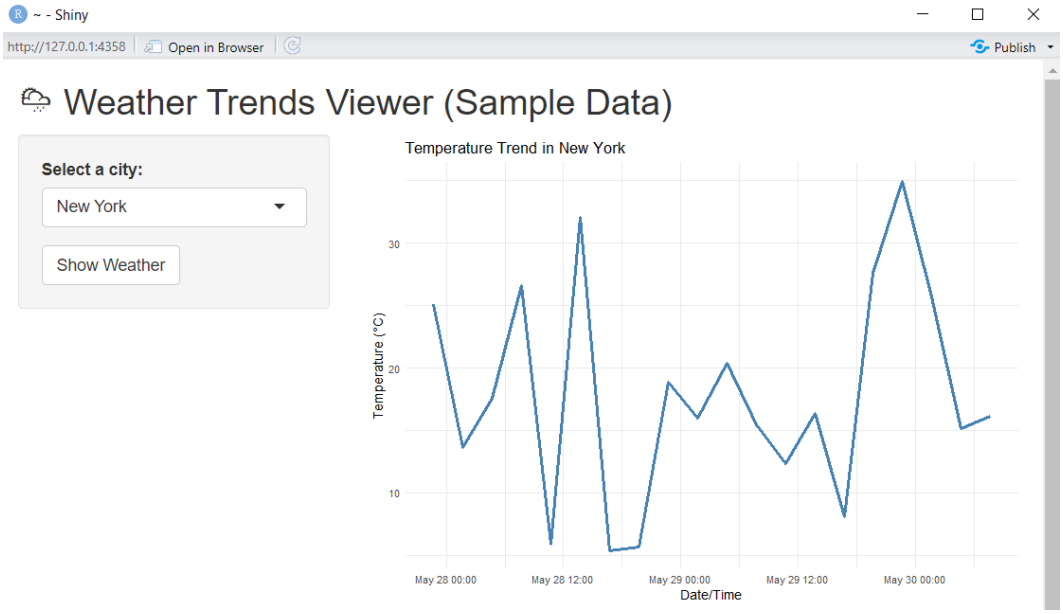
```
library(shiny)
library(ggplot2)
library(lubridate)
library(dplyr)

# □ List of 10 sample cities
cities <- c("New York", "London", "Tokyo", "Paris", "Sydney",
            "Cairo", "Moscow", "Rio de Janeiro", "Toronto", "Mumbai")

# □ Generate mock weather data for each city
generate_city_data <- function(city_name) {
  data.frame(
    city = city_name,
    datetime = seq.POSIXt(from = Sys.time(), by = "3 hours", length.out =
20),
    temp = runif(20, min = 5, max = 35),
    humidity = runif(20, min = 30, max = 90),
    weather = sample(c("Clear", "Rain", "Clouds", "Sunny", "Snow"), 20,
replace = TRUE)
  )
}

# □ Full dataset for all cities
weather_dataset <- bind_rows(lapply(cities, generate_city_data))

# □ UI
```

```
ui <- fluidPage(
  titlePanel("□ Weather Trends Viewer (Sample Data)"),
  sidebarLayout(
    sidebarPanel(
      selectInput("city", "Select a city:", choices = cities),
      actionButton("goButton", "Show Weather")
    ),
    mainPanel(
      plotOutput("tempPlot"),
      plotOutput("humidityPlot")
    )
  )
)

# □ Server
server <- function(input, output) {
  weather_data <- eventReactive(input$goButton, {
    filter(weather_dataset, city == input$city)
  })

  output$tempPlot <- renderPlot({
    df <- weather_data()
    ggplot(df, aes(x = datetime, y = temp)) +
      geom_line(color = "steelblue", size = 1.2) +
      labs(title = paste("Temperature Trend in", input$city),
           x = "Date/Time", y = "Temperature (°C)") +
      theme_minimal()
  })

  output$humidityPlot <- renderPlot({
```

```
    df <- weather_data()
    ggplot(df, aes(x = datetime, y = humidity)) +
      geom_line(color = "darkgreen", size = 1.2) +
      labs(title = paste("Humidity Trend in", input$city),
           x = "Date/Time", y = "Humidity (%)") +
      theme_minimal()
  })
}

# □ Run App
shinyApp(ui = ui, server = server)
```

APPENDIX B-SCREENSHOTS





REFERENCES

1. Chang, W., Cheng, J., Allaire, J., Xie, Y., & McPherson, J. (2023). *Shiny: Web Application Framework for R* (Version 1.8.0) [R package]. Retrieved from https://shiny.rstudio.com
2. Grolemund, G., & Wickham, H. (2016). *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*. O'Reilly Media.
3. Open Weather Map API. (n.d.). Retrieved from https://openweathermap.org/api
4. R Core Team. (2023). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Retrieved from https://www.r-project.org
5. Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York.