

**WEATHER WHIZ**

##### Mini Project - Report submitted by

|  |  |
| --- | --- |
| **NERAJAKSHARAJU N A** (4NM21CS095) |  |

6th Semester B.E.

##### Under the Guidance of

#### Ms. RAJASHREE

Assistant Professor Gd-I

*In partial fulfillment of the requirements for the award of the Degree of*

#### Bachelor of Engineering in Computer Science Engineering

*from*

##### Visvesvaraya Technological University, Belagavi

Department of Computer Science Engineering NMAM Institute of Technology, Nitte - 574110

(An Autonomous Institution affiliated to VTU, Belagavi)

### MAY 2024



### DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

CERTIFICATE

## Certified that the Mini project work entitled

*“***WEATHER WHIZ***”*

## C:\Users\venkatesh\AppData\Local\Microsoft\Windows\INetCache\Content.Word\NET Logo.jpgis a bonafide work carried out by

|  |  |
| --- | --- |
| **NERAJAKSHARAJU N A**  (4NM21CS095) |  |

*of 6th Semester B.E. in partial fulfillment of the requirements for the award of* ***Bachelor of Engineering Degree in Computer Science Engineering*** *prescribed by* ***Visvesvaraya Technological University, Belagavi*** *during the year* ***2023-2024****.*

##### Signature of the Guide Signature of the HOD

#### Viva Voce Examination

#### Name of the Examiners Signature with Date

#### 1.

#### 2.

**ACKNOWLEDGEMENT**

The success of our "Weather Whiz" project would not have been possible without the guidance and support of many individuals and entities. We extend our heartfelt gratitude to those who provided us with invaluable insights and assistance throughout the development process.

First and foremost, we express our sincere appreciation to our project guides, Ms.Rajashree, Assistant Professor Gd-I, Department of Computer Science and Engineering. Their unwavering support, technical expertise, and valuable feedback guided us in creating an efficient and user-friendly application.

We also extend our thanks to Dr. Jyothi Shetty, Head of the Department of Computer Science and Engineering, NMAM Institute of Technology, Nitte, for her continuous guidance, encouragement, and insightful suggestions that significantly enhanced our project.

A special note of appreciation goes to Dr. Niranjan N. Chiplunkar, Principal, NMAM Institute of Technology, Nitte, for his consistent support and motivation, which inspired us to strive for excellence in this project.

We are fortunate to have been surrounded by a network of knowledgeable and supportive mentors who enriched our experience throughout the development of "Note Maker." Their contributions not only helped us achieve our goals but also provided us with valuable learning opportunities.

**ABSTRACT**

This project presents a weather application that leverages the OpenWeatherMap API to provide real-time weather information based on user input of city and country. The application displays essential weather parameters such as temperature, humidity, wind speed, cloudiness, and atmospheric pressure. One notable feature is the dynamic adaptation of the user interface, with background and text colors changing according to the temperature, offering users an intuitive visual representation of weather conditions. The project demonstrates practical integration of API data retrieval, user interface design, and dynamic content presentation, contributing to a seamless and user-friendly weather forecasting experience.

**TABLE OF CONTENTS**

[INTRODUCTION 5](#_TOC_250001)

[PROBLEM STATEMENT 9](#_bookmark0)

[METHODOLOGY 10](#_bookmark1)

[IMPLEMENTATION 11](#_bookmark2)

[RESULTS 13](#_bookmark3)

[CONCLUSION 18](#_bookmark4)

[REFERENCES 19](#_TOC_250000)

# INTRODUCTION

#### Overview of the Project:

This project is a weather application designed to provide users with up-to-date weather information for any location worldwide. Utilizing the OpenWeatherMap API, the application retrieves data such as temperature, humidity, wind speed, cloudiness, and atmospheric pressure based on user input of city and country. The user interface dynamically adapts to the current weather conditions, with background and text colors changing to reflect temperature variations. This intuitive design enhances user experience by providing clear visual cues about the weather. The project encompasses features like API data retrieval, user input handling, dynamic UI updates, and error handling, demonstrating a comprehensive approach to weather forecasting.

#### Objectives of the Project:

The primary objectives of the "Weather Whiz" project include:

1. Retrieve real-time weather data from the OpenWeatherMap API.
2. Dynamically update the user interface to display weather information based on user input.
3. Implement background and text color changes to reflect temperature variations.
4. Handle user input validation and error responses gracefully.
5. Enhance user experience through intuitive design and clear presentation of weather information.

#### Importance:

The "Note Maker" project is highly relevant and valuable in today's fast-paced world, where efficient information management is crucial. The application offers several key benefits:

1. Provides users with accurate and timely weather information for any location worldwide.
2. Enhances user experience through intuitive design and dynamic UI updates.
3. Demonstrates the integration of external APIs for data retrieval in mobile application development.
4. Showcases best practices in handling user input validation and error responses.
5. Offers a practical example of how technology can be used to improve everyday tasks, such as checking the weather forecast.

# PROBLEM STATEMENT

Develop a mobile application that allows users to retrieve real-time weather information for any location globally. The application should utilize the OpenWeatherMap API to fetch weather data, including temperature, humidity, wind speed, and cloudiness. Users should be able to input the desired city and country and receive accurate weather updates displayed in a user-friendly manner. Additionally, the application should dynamically adjust the background and text color based on the temperature to enhance readability and provide a visually appealing experience. The primary challenge is to seamlessly integrate API data retrieval, user interface design, and dynamic color adjustments while ensuring responsiveness and error handling for a smooth user experience.

# METHODOLOGY

The methodology for this project adopts a structured approach aimed at delivering a robust and user-centric weather application. It involves several iterative stages, starting with comprehensive planning and requirements gathering. This phase includes analyzing user needs, defining project scope, and establishing clear objectives.

Following the planning phase, the design and prototyping stage come into play. Here, the focus is on creating intuitive user interfaces and efficient workflows. Visual designs and UI prototypes are developed, allowing stakeholders to visualize the final product and provide feedback.

Once the designs are finalized, the development phase begins. This involves implementing the application's features and functionalities according to the defined requirements. Using suitable tools and frameworks, development efforts are directed toward building a scalable and reliable solution.

Testing and quality assurance play integral roles in the development process. Rigorous testing is conducted to identify and rectify any bugs, errors, or performance issues. Various testing techniques, including unit testing, integration testing, and user acceptance testing, are employed to ensure the application meets quality standards.

Throughout the development lifecycle, an iterative refinement approach is adopted. Feedback from testing is used to make continuous improvements, ensuring the application aligns with user expectations and delivers an optimal user experience.

Finally, comprehensive documentation is prepared to assist users and developers, and the deployment process ensures smooth release of the application to production environments.

REQUIREMENTS:

**Development Environment:** Android Studio.

#### Programming Language: Java.

#### System Requirements: Minimum 8 GB of RAM.

# IMPLEMENTATION

The implementation of this project revolves around two primary aspects: user interface design and backend logic. In terms of the user interface, the goal is to create a seamless and visually appealing experience for users interacting with the weather application. This involves designing XML layouts that incorporate input fields for users to enter the desired city and country, as well as displaying the retrieved weather details in a clear and organized manner. Emphasis is placed on readability and user-friendly design principles to ensure that users can easily interpret the displayed information.

On the backend side, the application utilizes the Volley library to handle HTTP requests to the OpenWeatherMap API. These requests are made based on the user input provided through the UI, allowing the application to fetch relevant weather data for the specified location. Upon receiving the JSON response from the API, the application parses the data to extract key weather parameters such as temperature, description, humidity, and wind speed. Calculations are then performed to convert the temperature from Kelvin to Celsius, facilitating a more intuitive presentation of the weather information to the user.

A notable feature of the implementation is the dynamic adjustment of colors based on the temperature. The background color of the application layout is determined dynamically according to predefined temperature ranges, providing users with a visual representation of the current weather conditions. Additionally, the text color of the weather details is adjusted to ensure optimal readability and contrast with the background, further enhancing the overall user experience.

Throughout the implementation process, a strong emphasis is placed on error handling, testing, and documentation. Robust error handling mechanisms are implemented to gracefully handle scenarios such as invalid user input or network/API errors. Thorough testing is conducted across different devices and screen sizes to verify compatibility and responsiveness, with any encountered issues being promptly debugged and addressed. Comprehensive documentation is also provided, including code comments, explanations, and user guides, to facilitate understanding, maintenance, and effective utilization of the application.

# RESULTS

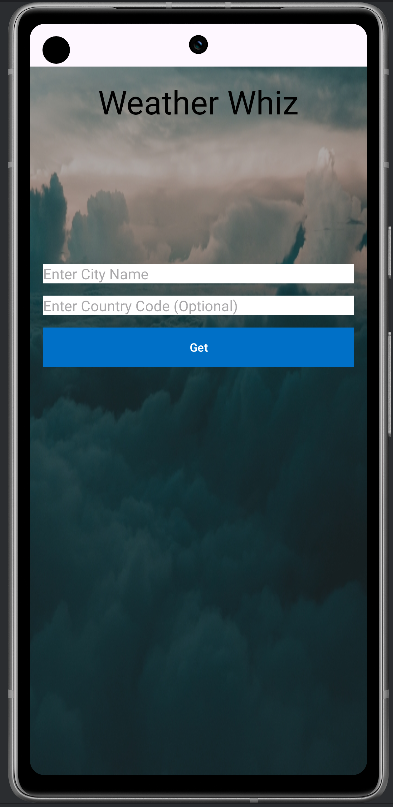
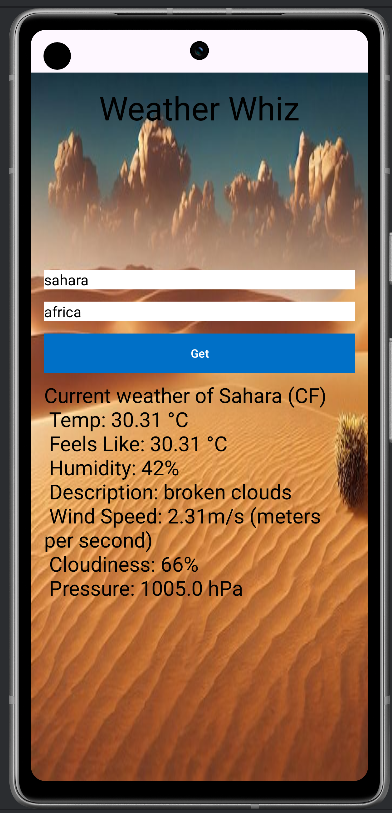
 

Fig 1:Main Page. Fig 2: Hot weather(temp>30)

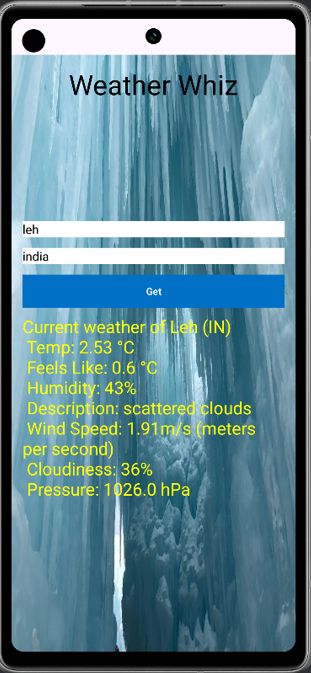
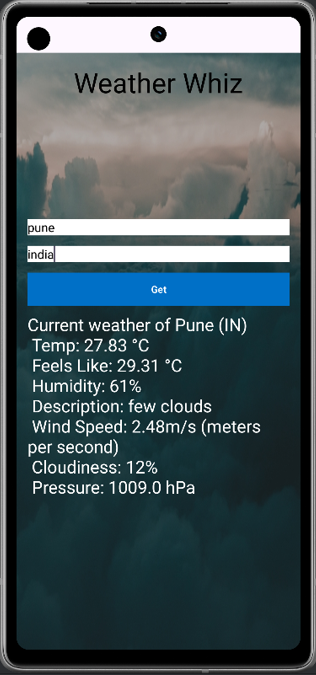
 

Fig 3 Cold weather(temp<10): Fig 4: Moderate weather(10<temp<30)

.

# CONCLUSION

The conclusion of the project underscores the successful implementation of a weather application using Android Studio and Java. Through rigorous development and testing, the application now provides accurate weather information based on user input. The project's objectives were met, offering users a seamless experience in accessing weather data. Moving forward, further enhancements and optimizations could be explored to enhance the application's functionality and user experience. Overall, the project demonstrates the effectiveness of utilizing modern development tools and methodologies to create practical and user-friendly mobile applications

# REFERENCES

1. OpenWeatherMap API Documentation
2. Android Developer Documentation
3. Volley Library Documentation
4. Stack Overflow
5. Java Documentation
6. Android Studio Documentation
7. GitHub