A Project Report

on

Weather Forecasting App

Submitted in partial fulfillment of the requirement

for the award of the degree of

Bachelor of Technology in Computer Science and Engineering

Under The Supervision of
Mr. E. Gautham
Assistant Professor
Department of Computer Science and Engineering

Submitted By

19SCSE1010706 – HRITIK JAISWAL

19SCSE1010041 – KAUSHAL PANDEY

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GALGOTIAS UNIVERSITY, GREATER NOIDA, INDIA

DECEMBER - 2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the project, entitled **"weather forecasting app"** in partial fulfillment of the requirements for the award of the **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING**
submitted in the **School of Computing Science and Engineering** of Galgotias University, Greater Noida, is an original work carried out during the period of **JULY-2021 to DECEMBER-2021**, under the supervision of **Mr.E.Gautham, Assistant Professor**, **Department of Computer Science and Engineering** of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the project has not been submitted by me/us for the award of any other degree of this or any other places.

19SCSE1010706 – HRITIK JAISWAL
19SCSE1010041 – KAUSHAL PANDEY

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Supervisor

(Mr.E.Gautham, Assistant Professor)

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of **19SCSE1010706 – HRITIK JAISWAL, 19SCSE1010041 – KAUSHAL PANDEY** has been held on _____and his/her work is recommended for the award of **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING**.

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date:

Place:

TABLE OF CONTENTS

ABSTRACT

Weather forecasting is the application of science and technology to predict the state of the atmosphere for a given location . Ancient weather forecasting methods usually relied on observed patterns of events, also termed pattern recognition. For example, it might be observed that if the sunset was particularly red, the following day often brought fair weather . However, not all of these predictions prove reliable.

Here this system will predict weather based on parameters such as temperature, humidity and wind. User will enter current temperature; humidity and wind, System will take this parameter and will predict weather(rainfall in inches) from previous data in database(dataset). The role of the admin is to add previous weather data in database, so that system will calculate weather(estimated rainfall in inches) based on these data. Weather forecasting system takes parameters such as temperature, humidity, and wind and will forecast weather based on previous record therefore this prediction will prove reliable. This system can be used in Air Traffic, Marine, Agriculture, Forestry, Military, and Navy etc.

Table of content

1.INTRODUCTION

- **Data Warehousing**

Data Warehouse is electronic storage of a large amount of information by a business which is designed for query and analysis instead of transaction processing. It is a process of transforming data into information and making it available to users for analysis.

- **Data Mining**

Data mining is looking for hidden, valid, and potentially useful patterns in huge data sets. Data Mining is all about discovering unsuspected/ previously unknown relationships amongst the data.It is a multi-disciplinary skill that uses machine learning, statistics, AI and database technology.

1.1. Introduction

As PC innovation and high velocity scattering frameworks developed (e.g.Internet), National Weather Service (NWS) clients/accomplices were demanding detailed conjectures in gridded, computerized and realistic organizations. Customer NWS text forecast items limit the measure of extra data that can be conveyed to the client local area. The idea of computerized information base estimating gives the capability to meet client/accomplice requests for more precise, detailed hydrometeorological gauge.

1.1 Literature survey

Weather forecasting has been one of the most challenging difficulties around the world because of both its practical value in popular scope for scientific study and meteorology. Weather is a continuous, dynamic ,multi-dimensional chaotic process, and data-intensive and these properties make weather forecasting a stimulating challenge. It is one of the most imperious and demanding operational responsibilities that must be carried out by many meteorological services all over the globe. Various organizations / workers in India and abroad have done demonstrating using supported time series data manipulation.

1.2 Problem Definition

They may be caused by the use of different data sources ... quality control of the data, space and time scale for which the information is valid, rendering of the information [or] interpretation by the user of the information."

Problem Solution:-

The application is reduced as much as possible to avoid errors while entering the data. It will also provide error message while entering invalid data. As described above it can lead to error free, secure and fast management system We will build full secure system without any trespass of any data.

1.2 **Scope**

It tells about the current weather with the help of api.

1.4 Purpose

There are several reasons why weather forecasts are important. They would certainly be missed if they were not there. It is a product of science that impacts the lives of many people. The following is a list of various reasons why weather forecasts are important:

1. Helps people prepare for how to dress (i.e. warm weather, cold weather, windy weather, rainy weather)
2. Helps businesses and people plan for power production and how much power to use (i.e. power companies, where to set thermostat)
3. Helps people prepare if they need to take extra gear to prepare for the weather (i.e. umbrella, rain coat, sun screen)
4. Helps people plan outdoor activities (i.e. to see if rain/storms/cold weather will impact outdoor event)
5. Helps curious people to know what sort of weather can be expected (i.e. a snow on the way, severe storms)
6. Helps businesses plan for transportation hazards that can result from the weather (i.e. fog, snow, ice, storms, clouds as it relates to driving and flying for example)
7. Helps people with health related issues to plan the day (i.e. allergies, asthma, heat stress)
8. Helps businesses and people plan for severe weather and other weather hazards (lightning, hail, tornadoes, hurricanes, ice storms)
9. Helps farmers and gardeners plan for crop irrigation and protection (irrigation scheduling, freeze protection)

1.5 Problem and Existing Technology

The traditional forecast process employed by most NMHSs involves forecasters producing text-based, sensible, weather-element forecast products (e.g. maximum/minimum temperature, cloud cover) using numerical weather prediction (NWP) output as guidance. The process is typically schedule-driven, product-oriented and labour-intensive. Over the last decade, technological advances and scientific breakthroughs have allowed NMHSs' hydrometeorological forecasts and warnings to become much more specific and accurate.

As computer technology and high-speed dissemination systems evolved (e.g. Internet), National Weather Service (NWS) customers/partners were demanding detailed forecasts in gridded, digital and graphic formats. Traditional NWS text forecast products limit the amount of additional information that can be conveyed to the user community. The concept of digital database forecasting provides the capability to meet customer/partner demands for more accurate, detailed hydrometeorological forecasts. Digital database forecasting also offers one of the most exciting opportunities to integrate PWS forecast dissemination and service delivery, which most effectively serves the user community.

1.6 Proposed System

User will enter current temperature; humidity and wind, System will take this parameter and will predict weather from previous data in database. The role of the admin is to add previous weather data in database, so that system will calculate weather based on these data. Weather forecasting system takes parameters such as temperature, humidity, and wind and will forecast weather based on previous record therefore this prediction will prove reliable.

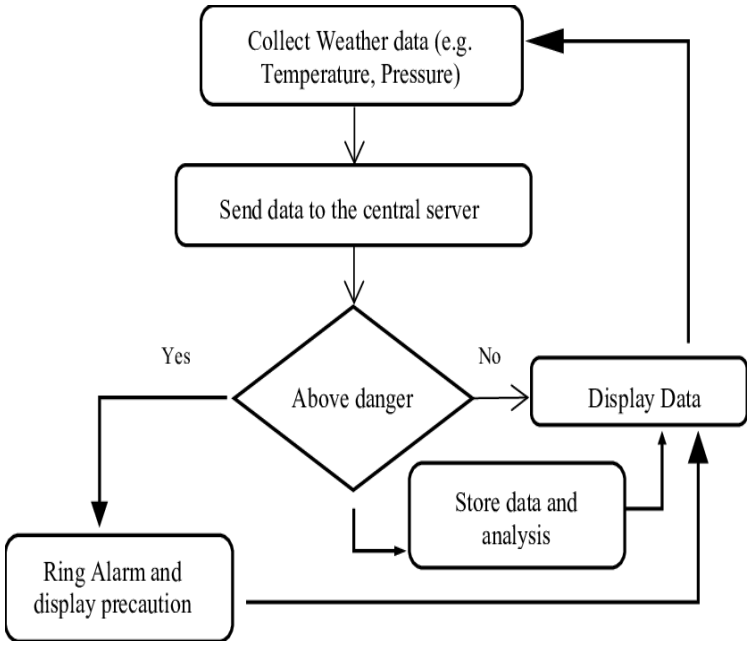
2.REQUIREMENTS

2.1. Platform Requirements

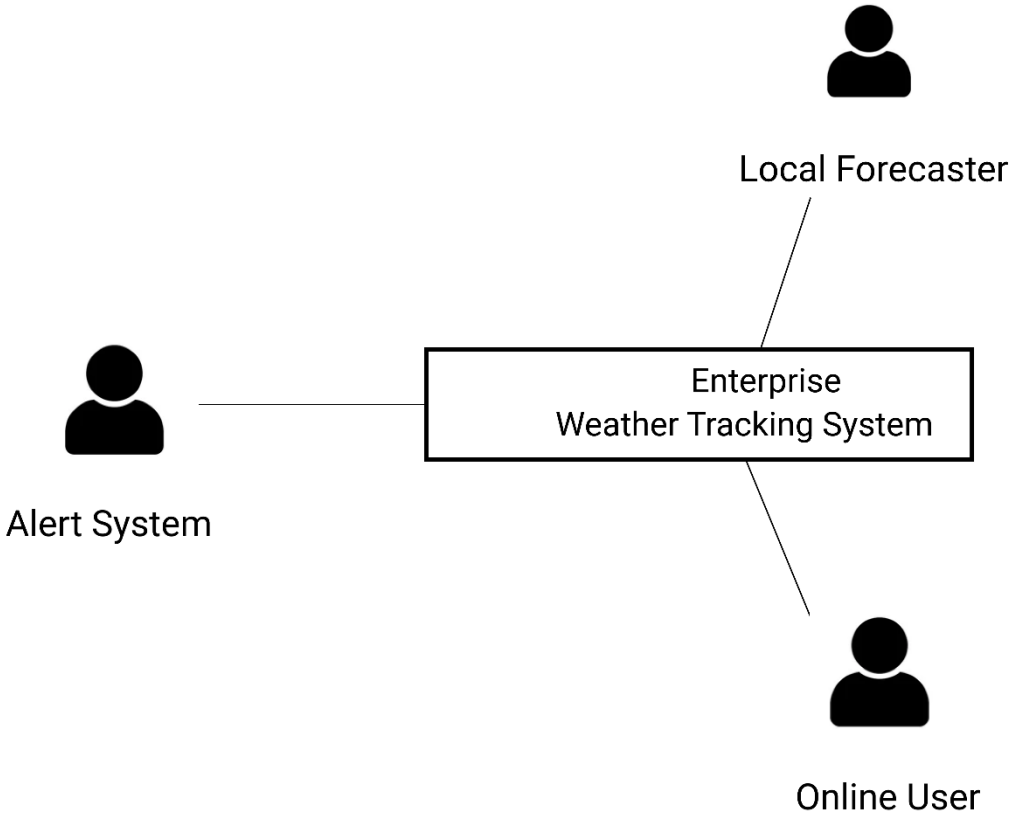
Hardware/Software	Hardware / Software element	Specification /version
Hardware	Processor	i5
	RAM	8GB
	ssd	512GB
Software	OS	Windows,Linux.
	Python IDE	Jupyter NoteBook. Python 3.

CHAPTER 3: DESIGN & IMPLEMENTATION

3.1 DFD DIAGRAM :



3.2 CONTEXT DIAGRAM



3.3 CODE :

```
# import all functions from the tkinter
from tkinter import *
from tkinter import messagebox

# function to find weather details
# of any city using openweathermap api
def tell_weather() :

        # import required modules
        import requests, json

        city_name = city_field.get()

        complete_url =
"http://api.openweathermap.org/data/2.5/weather?q="+city_name+"&appid=1798b68d151252eee4293b935
85943ee"
        # get method of requests module
        # return response object
        response = requests.get(complete_url)

        # json method of response object convert
        # json format data into python format data
        x = response.json()

        # now x contains list of nested dictionaries
        # we know dictionary contains key value pair
        # check the value of "cod" key is equal to "404"
        # or not if not that means city is found
        # otherwise city is not found
        if x["cod"] != "404" :

                # store the value of "main" key in variable y
                y = x["main"]
```

```
            # store the value corresponding to the "temp" key of y
            current_temperature = y["temp"]

            # store the value corresponding to the "pressure" key of y
            current_pressure = y["pressure"]

            # store the value corresponding to the "humidity" key of y
            current_humidity = y["humidity"]

            # store the value of "weather" key in variable z
            z = x["weather"]

            # store the value corresponding to the "description" key
            # at the 0th index of z
            weather_description = z[0]["description"]

            # insert method inserting the
            # value in the text entry box.
            temp_field.insert(15, str(current_temperature) + " Kelvin")
            atm_field.insert(10, str(current_pressure) + " hPa")
            humid_field.insert(15, str(current_humidity) + " %")
            desc_field.insert(10, str(weather_description) )

    # if city is not found
    else :

            # message dialog box appear which
            # shows given Error message
            messagebox.showerror("Error", "City Not Found \n"
                                          "Please enter valid city name")

            # clear the content of city_field entry box
            city_field.delete(0, END)


# Function for clearing the
```

```
# contents of all text entry boxes
def clear_all() :
        city_field.delete(0, END)
        temp_field.delete(0, END)
        atm_field.delete(0, END)
        humid_field.delete(0, END)
        desc_field.delete(0, END)

        # set focus on the city_field entry box
        city_field.focus_set()



# Driver code
if _name_ == "_main_" :

        # Create a GUI window
        root = Tk()

        # set the name of tkinter GUI window
        root.title("Gui Application")

        # Set the background colour of GUI window
        root.configure(background = "light green")

        # Set the configuration of GUI window
        root.geometry("425x175")

        # Create a Weather Gui Application label
        headlabel = Label(root, text = "Weather Gui Application",
                                        fg = 'black', bg = 'red')

        # Create a City name : label
        label1 = Label(root, text = "City name : ",
                                fg = 'black', bg = 'dark green')

        # Create a City name : label
        label2 = Label(root, text = "Temperature :",
```

```
                        fg = 'black', bg = 'dark green')

# Create a atm pressure : label
label3 = Label(root, text = "atm pressure :",
                        fg = 'black', bg = 'dark green')

# Create a humidity : label
label4 = Label(root, text = "humidity :",
                        fg = 'black', bg = 'dark green')

# Create a description :label
label5 = Label(root, text = "description :",
                        fg = 'black', bg = 'dark green')


# grid method is used for placing
# the widgets at respective positions
# in table like structure .
headlabel.grid(row = 0, column = 1)
label1.grid(row = 1, column = 0, sticky ="E")
label2.grid(row = 3, column = 0, sticky ="E")
label3.grid(row = 4, column = 0, sticky ="E")
label4.grid(row = 5, column = 0, sticky ="E")
label5.grid(row = 6, column = 0, sticky ="E")


# Create a text entry box
# for filling or typing the information.
city_field = Entry(root)
temp_field = Entry(root)
atm_field = Entry(root)
humid_field = Entry(root)
desc_field = Entry(root)

# grid method is used for placing
# the widgets at respective positions
# in table like structure .
```

```
# ipadx keyword argument set width of entry space .
city_field.grid(row = 1, column = 1, ipadx ="100")
temp_field.grid(row = 3, column = 1, ipadx ="100")
atm_field.grid(row = 4, column = 1, ipadx ="100")
humid_field.grid(row = 5, column = 1, ipadx ="100")
desc_field.grid(row = 6, column = 1, ipadx ="100")

# Create a Submit Button and attached
# to tell_weather function
button1 = Button(root, text = "Submit", bg = "red",
                                    fg = "black", command = tell_weather)

# Create a Clear Button and attached
# to clear_all function
button2 = Button(root, text = "Clear", bg = "red",
                                    fg = "black", command = clear_all)

# grid method is used for placing
# the widgets at respective positions
# in table like structure .
button1.grid(row = 2, column = 1)
button2.grid(row = 7, column = 1)

# Start the GUI
root.mainloop()
```

OUTPUT :


Gui Application
Weather Gui Application
City name :
Dehradun
Submit
Temperature :
273.778 Kelvin
atm pressure :
867.05 hPa
humidity :
87 %
description :
clear sky
Clear

Tools and Technology:-

Techs used are Python Ide ,API , Tkinter module , & request module.

Result

It will be a complete working and fully established and developed.

Python Ide-

Python… the most popular and favorite programming language in the world for programmers of all age groups. If you are a beginner this language is strongly recommended to learn first. Well, In programming, we all know the importance of code editor and IDEs to write our program and to execute it but choosing the best code editor or IDE is always a confusing question. Understand that the best code editor or IDE depends on so many things such as programming language, project type, project size, OS support and considering a lot of other features. If we talk about Python so this language is also not an exception.

List of Best Python IDE

1. PyCharm

In industries most of the professional developers use PyCharm and it has been considered the best IDE for python developers. It was developed by the Czech company JetBrains and it's a cross-platform IDE. It gives daily tips to improve your knowledge of how you can use it more efficiently which is a very good feature. It comes in two versions community version and a professional version where community version is free but the professional version is paid. Below are some other features of this IDE.

- It is considered as an intelligent code editor, fast and safe refactoring, and smart code.
- Features for debugging, profiling, remote development, testing the code, auto code completion, quick fixing, error detection and tools of the database.
- Support for Popular web technologies, web frameworks, scientific libraries and version control.

2. Spyder

Spyder is another good open-source and cross-platform IDE written in Python. It is also called Scientific Python Development IDE and it is the most lightweight IDE for Python. It is mainly used by data scientists who can integrate with Matplotlib, SciPy, NumPy, Pandas, Cython, IPython, SymPy, and other open-source software. It comes with the Anaconda package manager distribution and it has some good advanced features such as edit, debug, and data exploration. Below are some other features of this IDE.

- Auto code completion and syntax highlighting.
- Ability to search and edit the variables from the graphical user interface itself.
- Static code analysis
- It is very efficient in tracing each step of the script execution by a powerful debugger.

3. Eclipse PyDev

Eclipse is one of the most popular IDE among developers which is written in Java but you can install Pydev plugin in eclipse and use it for Python as well. The primary focus of this IDE is the analysis of code, debugging in the graphical pattern, refactoring of python code, etc. Eclipse PyDev is stable and provides good performance for most of the python project life cycle. Below are some other features of this IDE.

- Pydev supports Django integration, Unittest integration, PyLint integration
- Code folding and code completion with auto import
- Good syntax high lighting and remote debugger

- Interactive console
- Allows you to create a Google App Engine (GAE) Python project

4. IDLE

IDLE is a cross-platform open-source IDE that comes by default with Python so you don't need to worry about the installation or setup. IDLE is written in Python and this IDE is suitable for beginner level developers who want to practice on python development. IDLE is lightweight and simple to use so you can build simple projects such as web browser game automation, basic web scraping applications, and office automation. This IDE is not good for larger projects so move to some advance IDEs after learning the basics from IDLE.

- Python shell with syntax highlighting
- Call stack's clear visibility
- Multi-window code editor that allows features like smart indentation, autocomplete, etc
- It has an interactive interpreter with colorizing of input, output, and error messages.
- Program animation or stepping.

5. Wing

Wing IDE is created by Wingware and it is faster, stable and extremely lightweight cross-platform Python IDE. It comes in three editions:

- **Wing Pro (Free Trial):** A full-featured commercial version, for professional programmers.
- **Wing Personal (Paid):** Free version that omits some features, for students and hobbyists.
- **Wing 101 (Paid):** A very simplified free version, for beginners in programming.

This IDE comes with a strong debugger and smart editor that makes the interactive Python development speed, accurate and fun to perform. Some of its main features are given below…

- Automatic multi-process, child process, remote debug process and refactoring, etc.
- Test-driven development with various frameworks like the unittest, pytest, nose, doctest, and Django testing.
- It also has auto code completion in which the error is displayed in a feasible manner and line editing is also possible.
- Remote development support

API –

API is the acronym for Application Programming Interface, which is a software intermediary that allows two applications to talk to each other. Each time you use an app like Facebook, send an instant message, or check the weather on your phone, you're using an API.

What Is an Example of an API?

When you use an application on your mobile phone, the application connects to the Internet and sends data to a server. The server then retrieves that data, interprets it, performs the necessary actions and sends it back to your phone. The application then interprets that data and presents you with the information you wanted in a readable way. This is what an API is - all of this happens via API.

To explain this better, let us take a familiar example.

Imagine you're sitting at a table in a restaurant with a menu of choices to order from. The kitchen is the part of the "system" that will prepare your order. What is missing is the critical link to communicate your order to the kitchen and deliver your food back to your table. That's where the waiter or API comes in. The waiter is the messenger – or API – that takes your request or order and tells the kitchen – the system – what to do. Then the waiter delivers the response back to you; in this case, it is the food.

Tkinter –

Tkinter is the Python interface to the Tk GUI toolkit shipped with Python.

Tkinter Programming

Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

Creating a GUI application using Tkinter is an easy task. All you need to do is perform the following steps −

- Import the *Tkinter* module.
- Create the GUI application main window.
- Add one or more of the above-mentioned widgets to the GUI application.
- Enter the main event loop to take action against each event triggered by the user.

Request module-

Requests library is one of the integral part of Python for making HTTP requests to a specified URL. Whether it be REST APIs or Web Scrapping, requests is must to be learned for proceeding further with these technologies. When one makes a request to a URI, it returns a response. Python requests provides inbuilt functionalities for managing both the request and response.

5.CONCLUSION

Weather plays a major role in our daily life, and without the meteorologist and forecaster we would have difficulty planning our daily activities. As we can see, the weather is not a simple subject like we may have been thinking. The study of weather phenomenon requires the use of science, math, and different types of equipment and technology and data. Even with all these equipment, data, and observation tools, the weather continues to be a topic to study because it is constantly changing. Meteorologist and forecasters predict the weather and its possible changes, but in reality, weather is still unpredictable.

We effectively anticipated the precipitation utilizing the straight relapse yet here this is not exceptionally exact just a few times any way it relies on the environment changes to season to season. Here we are taking just summer season climate informational collection it only useful to fore see precipitation in summer season

6.REFERENCES

Textbooks:-

1. Data Mining: The Textbook 2015 Edition, Kindle Editionby **Charu C. Aggarwal .**

2. Data Mining: Concepts and TechniquesBy **Jiawei Han, Jian Pei, Micheline Kamber.**

Weblinks:-

1) https://towardsdatascience.com/introduction-to-machine-learning-algorithms-linear-regression-14c4e325882a

2) https://www.kaggle.com/grubenm/austin-weather