***WEATHER* FORECASTER**

***A***

***Mini Project Report***

*Submitted* ***in partial fulfilment of the***

***Requirements for the award of the Degree of***

**BACHELOR OF ENGINEERING**

**IN**

**INFORMATION TECHNOLOGY**

**By**

**P.SAI SRINATH 1602-20-737-159**

**K. ROHAN SAI 1602-20-737-155**

**N. ABHIRAM 1602-20-737-121**



**Department of Information Technology**

**Vasavi College of Engineering (Autonomous)**

**VASAVI COLLEGE OF ENGINEERING (AUTONOMOUS)**

**(AFFILIATED TO OSMANIA UNIVERSITY)**

**HYDERABAD - 500 030**

**Department of Information** **Technology**



**DECLARATION BY CANDIDATE**

We, **P. SAI SRINATH, K. ROHAN SAI, N. ABHIRAM** bearing hall ticket number,**1602-20-737-159,1602-20-737-155,1602-20-737-121** hereby declare that the project report entitled **“Weather Forecasting”** Department of Information Technology, Vasavi College of Engineering, Hyderabad, is submitted in partial fulfillment of the requirement for the award of the degree of **Bachelor of Engineering in Information Technology**

This is a record of Bonafide work carried out by me and the results embodied in this project report has not been submitted to any other university or institute for the award of any other degree or diploma.

**P.SAI SRINATH 1602-20-737-159**

**K. ROHAN SAI 1602-20-737-155**

**N. ABHIRAM 1602-20-737-121**

**VASAVI COLLEGE OF ENGINEERING (AUTONOMOUS)**

**(AFFILIATED TO OSMANIA UNIVERSITY)**

**HYDERABAD - 500 030**

**Department of Information** **Technology**



**BONAFIDE CERTIFICATE**

This is to certify that the project entitled **“Weather Forecasting”** being submitted by **P. SAI SRINATH, K. ROHAN SAI, N. ABHIRAM** bearing **1602-20-737-159,1602-20-737-155,1602-20-737-121** in partial fulfillment of the requirements for the completion of MINI PROJECT of Bachelor of Engineering in Information Technology is a record of bonafide work carried out by them under my guidance.

Dr. K. Ram Mohan Rao

Faculty I/C HOD, IT

**ACKNOWLEDGMENT**

We extend our sincere thanks to Dr. S. V. Ramana, Principal, Vasavi College of Engineering for his encouragement.

We express our sincere gratitude to Dr. K. Ram Mohan Rao, Professor & Head, Department of Information Technology, Vasavi College of Engineering, for introducing the Mini-Project module in our curriculum, and also for his suggestions, motivation, and co-operation for the successful completion of our Mini Project.

We also want to thank and convey our gratitude towards our mini project coordinators **Sireesha Chitteu Mam**, for guiding us in understanding the process of project development & giving us timely suggestions at every phase.

We would also like to sincerely thank the project reviewers for their valuable inputs and suggestions.

**ABSTRACT**

This is a weather forecast project which is going to be developed using python programming. This weather prediction project in python is a simple console-based application using machine learning which helps to determine if the current situation of weather is good to play or not. It is going to be done under the supervised learning in which data are given first to train the system and then result for new data will be produced. weather forecasting is a complex and challenging science that depends on efficient interplay of weather observation and data analysis. The user should enter the desired name of the place . So that our projects help in calculating the weather deatils such as temperature, humidity, wind speed for the next consecutive 3 days in 4 intervals of a day such as morning, noon, evening, and Night.By this we can find weather details of any city,district,country.

This weather predicting project have many applications in our day-to-day life. This helps us to know how good the weather is to go out and play. This also predicts the heavy rains and other natural calamities. For doing this project we gained some basic knowledge about machine learning in python. We are using requests library for this project.

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| S.No | Topic | Page No. |
| 1. | Introduction | 7 |
| 2. | Technology | 9 |
| 3. | Design | 10 |
| 4. | Implementation | 14 |
| 5. | Testing | 17 |
| 6. | Additional Knowledge Acquired | 23 |
| 7. | Conclusion and Future work | 24 |
| 8. | References | 25 |

**INTRODUCTION**

Weather forecasts are made by collecting as much data as possible about the current state of the atmosphere (particularly the temperature, humidity and wind) and using understanding of atmospheric processes (through meteorology) to determine how the atmosphere evolves in the future.However, the chaotic nature of the atmosphere and incomplete understanding of the processes mean that forecasts become less accurate as the range of the forecast increases.Traditional observations made at the surface of atmospheric pressure, temperature, wind speed, wind direction, humidity, precipitation are collected routinely from trained observers, automatic weather stations or buoys.During the data assimilation process, information gained from the observations is used in conjunction with a numerical model's most recent forecast for the time that observations were made to produce the meteorological analysis.

Numerical weather prediction models are computer simulations of the atmosphere.They take the analysis as the starting point and evolve the state of the atmosphere forward in time using understanding of physics and fluid dynamics.The complicated equations which govern how the state of a fluid changes with time require supercomputers to solve them.The output from the model provides the basis of the weather forecast.

Forecasting weather conditions is important for, e.g., operation of hydro power plants and for flood management. Mechanistic models are known to be computationally demanding. Hence, it is of interest to develop models that can predict weather conditions faster than traditional meteorological models. The field of machine learning has received much interest from the scientific community. Due to its applicability in a variety of fields, it is of interest to study whether an artificial neural network can be a good candidate for prediction of weather conditions in combination with large data sets. The availability of meteorological data from multiple online sources is an advantage. In order to simplify the retrieval of data, a Python API to read meteorological data has been developed, and ANN models have been developed using TensorFlow.

Weather prediction is a convenient case for studying machine learning. By developing APIs for accessing available data from meteorological institutes and other weather stations, this gives access to an abundance of data. Weather data is something that most people can relate to in their daily life, but is also important for energy systems, flood prediction, etc. Good physical based meteorological models are available, which makes it easy to compare the quality of machine learning models. In this paper, we have focused on a new Python API for collecting weather data, and given simple, introductory examples of how such data can be used in machine learning.

operational meteorologists, also known as forecasters:

* Collect weather data in some country, but it is mostly done by technicians elsewhere.
* Analyze data and numerical weather prediction model outputs to prepare daily weather forecasts.
* Provide weather advice and guidance to private or governmental users.
* Collaborate with the researchers for integrating science and technology into the forecast process, in particular for indices and model outputs, for weather-dependent users such as [farming](https://en.wikipedia.org/wiki/Agricultural_meteorology" \o "Agricultural meteorology), [forestry](https://en.wikipedia.org/wiki/Forest_fire_weather_index" \o "Forest fire weather index), [aviation](https://en.wikipedia.org/wiki/Aviation_meteorology" \o "Aviation meteorology), [maritime shipping and fisheries](https://en.wikipedia.org/wiki/Marine_weather_forecasting" \o "Marine weather forecasting), etc.

Python was chosen initially because it comes with many standard libraries useful in collecting, parsing, and storing data from the web. Among those particularly useful in this application were the regular expression library, the thread library, the object serialization library, and gzip data compression library. Other libraries, such as an HTTP client capable of accepting cookies ([ClientCookie](http://wwwsearch.sourceforge.net/ClientCookie/)), and an HTML table parser ([ClientTable](http://wwwsearch.sourceforge.net/ClientTable/)) were available as third party modules. These proved invaluable and were easy to use.Python is also used in the aggregation engine, which runs as a separate process to combine forecast accuracy scores into monthly and yearly slices. The aggregation process uses queries via [MySQLdb](http://sourceforge.net/projects/mysql-python) to the [MySQL](http://www.mysql.com/) database where the input modules have placed the forecast and climatological data they have harvested. Colorized maps, showing forecast accuracy by geographical area, are then generated for use on the web site and in printed reports.

**TECHNOLOGY**

**Hardware Requirements**

* Laptop
* Internet Connection

**Software Requirements**

* Windows
* PYTHON INTERPRETER

**Packages used**

**--tkinter:**

import tkinter as tk

from tkinter import \*

from tkinter import messagebox

### requests Module:

****Requests**** is a simple, yet elegant HTTP library. It allows you to send HTTP/1.1 requests extremely easily.

**URL to check weather:**

url = 'https://wttr.in/{}'

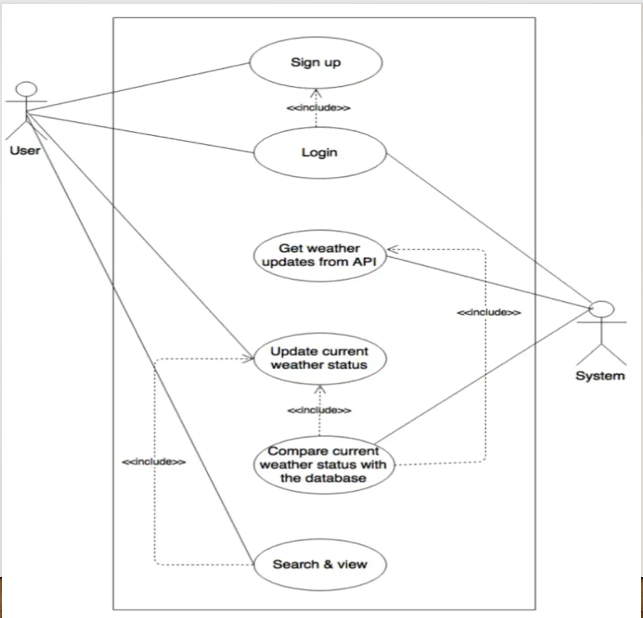
wttr.in is a console-oriented weather forecast service that supports various information representation methods like terminal-oriented ANSI-sequences for console HTTP clients (curl, httpie, or wget), HTML for web browsers, or PNG for graphical viewers.

wttr.in uses [wego](http://github.com/schachmat/wego) for visualization and various data sources for weather forecast information.

**PROPOSED WORK**

**DESIGN :**

USE CASE DIAGRAM –



**Use Case-1:**

**Name** : Login

**Actors** : User

**Description** : Allows user to login

**Pre-Conditions** : enter login details

**Post-Conditions** : login success or failed

**Main Flow:**

|  |  |
| --- | --- |
| **USER** | **SYSTEM** |
| 1.Enter username and password |  |
| 2.click login |  |
|  | 3.verifies login credentials. |
|  | 4.Takes to next page if login details are correct orelse enter  login details again. |

**Use Case-2:**

**Name** : weather details 1

**Actors** : User

**Description** : displaying weather details(temperature,wind condition)

**Pre-Conditions** : enter place name

**Post-Conditions** : display weather report for 3 days

**Main Flow:**

|  |  |
| --- | --- |
| **USER** | **SYSTEM** |
| 1.Enter place name. |  |
| 2.click “get weather details” |  |
|  | 3.checks for entered place in provided URL. |
|  | 4.displays weather details for 3 days(morning,noon,evening,night) |

**Use Case-3:**

**Name** : weather details 2

**Actors** : User

**Description** : testing for place which is not there.

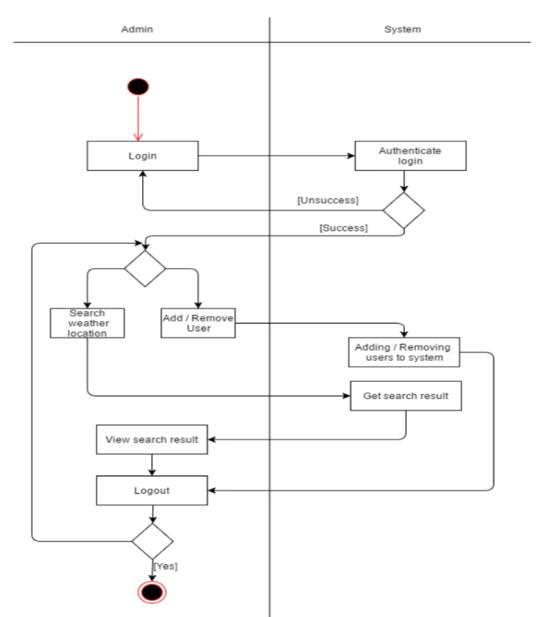
**Pre-Conditions** : enter wrong place name

**Post-Conditions** : unable to find location

**Main Flow:**

|  |  |
| --- | --- |
| **USER** | **SYSTEM** |
| 1.Enter wrong place |  |
| 2.click get weather details. |  |
|  | 3.checks for place in URL. |
|  | 4.displays “unable to find location” |

**ACTIVITY DIAGRAM**



**IMPLEMENTATION**

**Source code :**

#import the necessary package!

import tkinter as tk

import requests

from tkinter import \*

from tkinter import messagebox

#login

def Ok():

uname = e1.get()

password = e2.get()

if (uname == "" and password == ""):

messagebox.showinfo("", "Blank Not allowed")

elif (uname == "Admin" and password == "123"):

messagebox.showinfo("", "Login Success")

root.destroy()

else:

messagebox.showinfo("", "Incorrent Username and Password")

root = Tk()

root.title("Login")

root.geometry("300x200")

global e1

global e2

Label(root, text="UserName").place(x=10, y=10)

Label(root, text="Password").place(x=10, y=40)

e1 = Entry(root)

e1.place(x=140, y=10)

e2 = Entry(root)

e2.place(x=140, y=40)

e2.config(show="\*")

Button(root, text="Login", command=Ok, height=3, width=13).place(x=10, y=100)

root.mainloop()

root = tk.Tk()

canvas1 = tk.Canvas(root, width=400, height=300, relief='raised')

canvas1.pack()

#gui using tkinter

label1 = tk.Label(root, text='WEATHER FORECASTING')

label1.config(font=('helvetica', 14))

canvas1.create\_window(200, 25, window=label1)

label2 = tk.Label(root, text='ENTER PLACE NAME')

label2.config(font=('helvetica', 10))

canvas1.create\_window(200, 100, window=label2)

entry1 = tk.Entry(root)

canvas1.create\_window(200, 140, window=entry1)

#fetch the weater details

#display the result!

def weather():

url = 'https://wttr.in/{}'.format(entry1.get())

res = requests.get(url)

print(res.text)

button1 = tk.Button(text='Get WEATHER DETAILS', command=weather, bg='brown', fg='white',

font=('helvetica', 9, 'bold'))

canvas1.create\_window(200, 180, window=button1)

root.mainloop()

**Git Hub Link – https://gitlab.vce.ac.in/team4-c-web/weather-forecasting#modal-upload-blob**

**SCREENSHOTS OF APPLICATION TESTCASES**

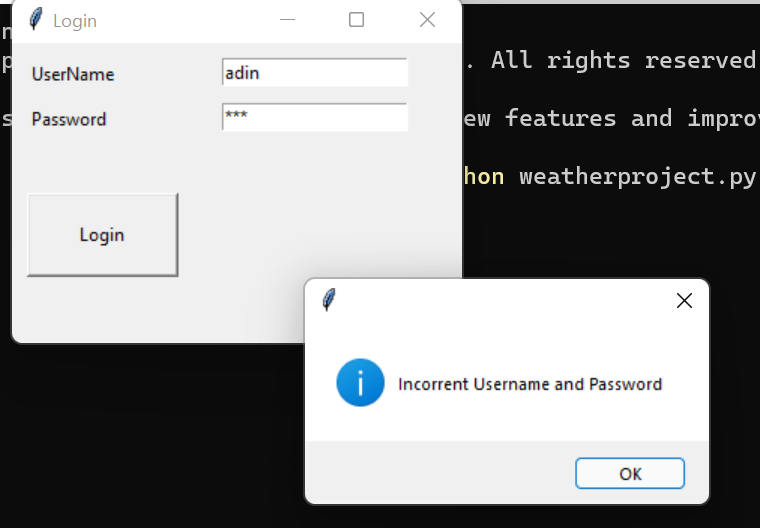
**Login page:**

Graphical user interface

Description automatically generated**­**

Graphical user interface, application

Description automatically generated



1. **GUI INTERFACE:**

Graphical user interface, text, application

Description automatically generated

1. **WEATHER RESULTS 1:**

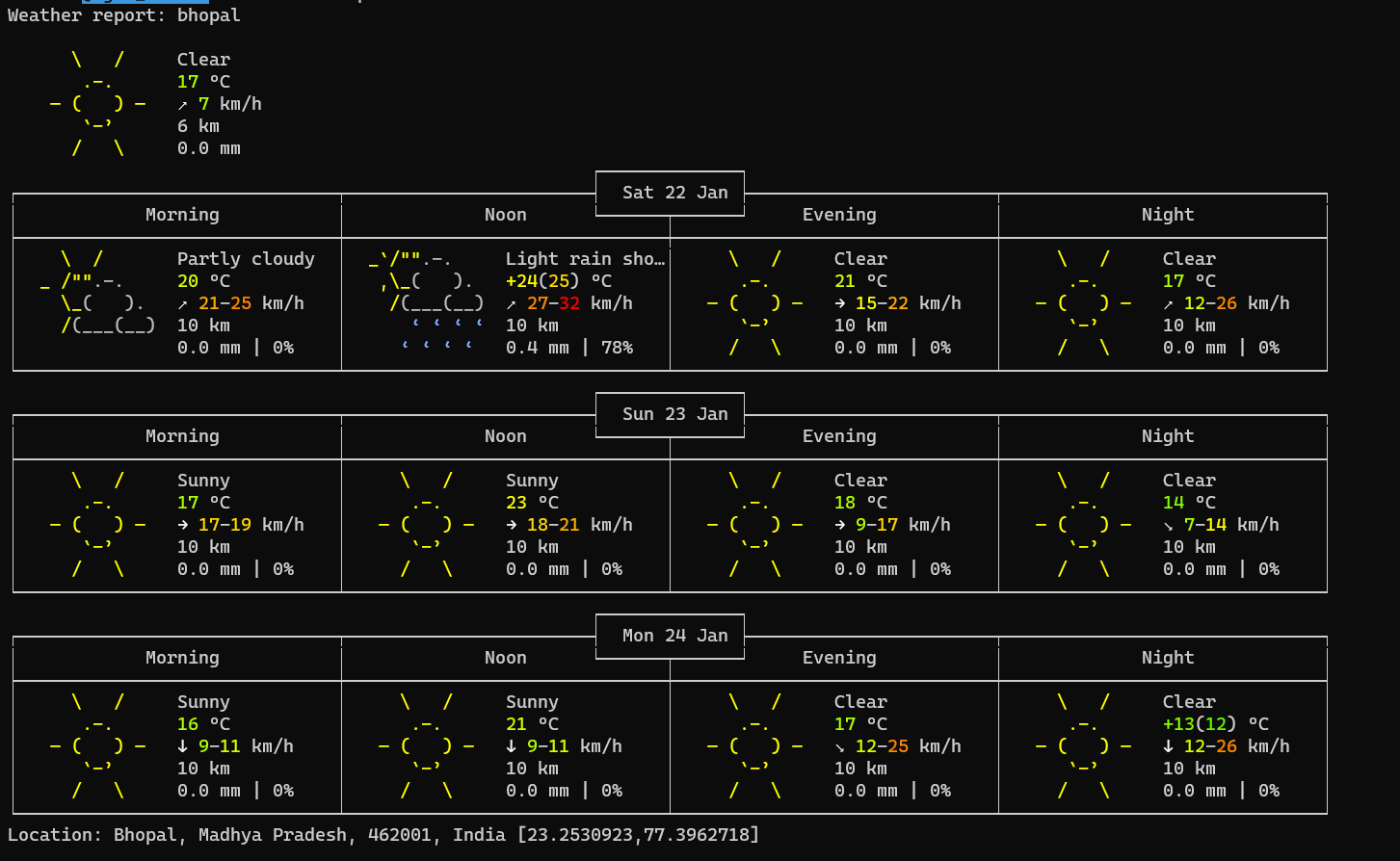
Graphical user interface

Description automatically generated

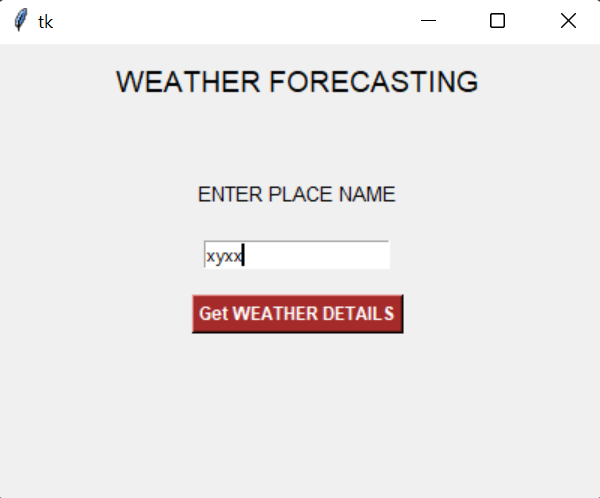
1. **WEATHER RESULTS 2:**

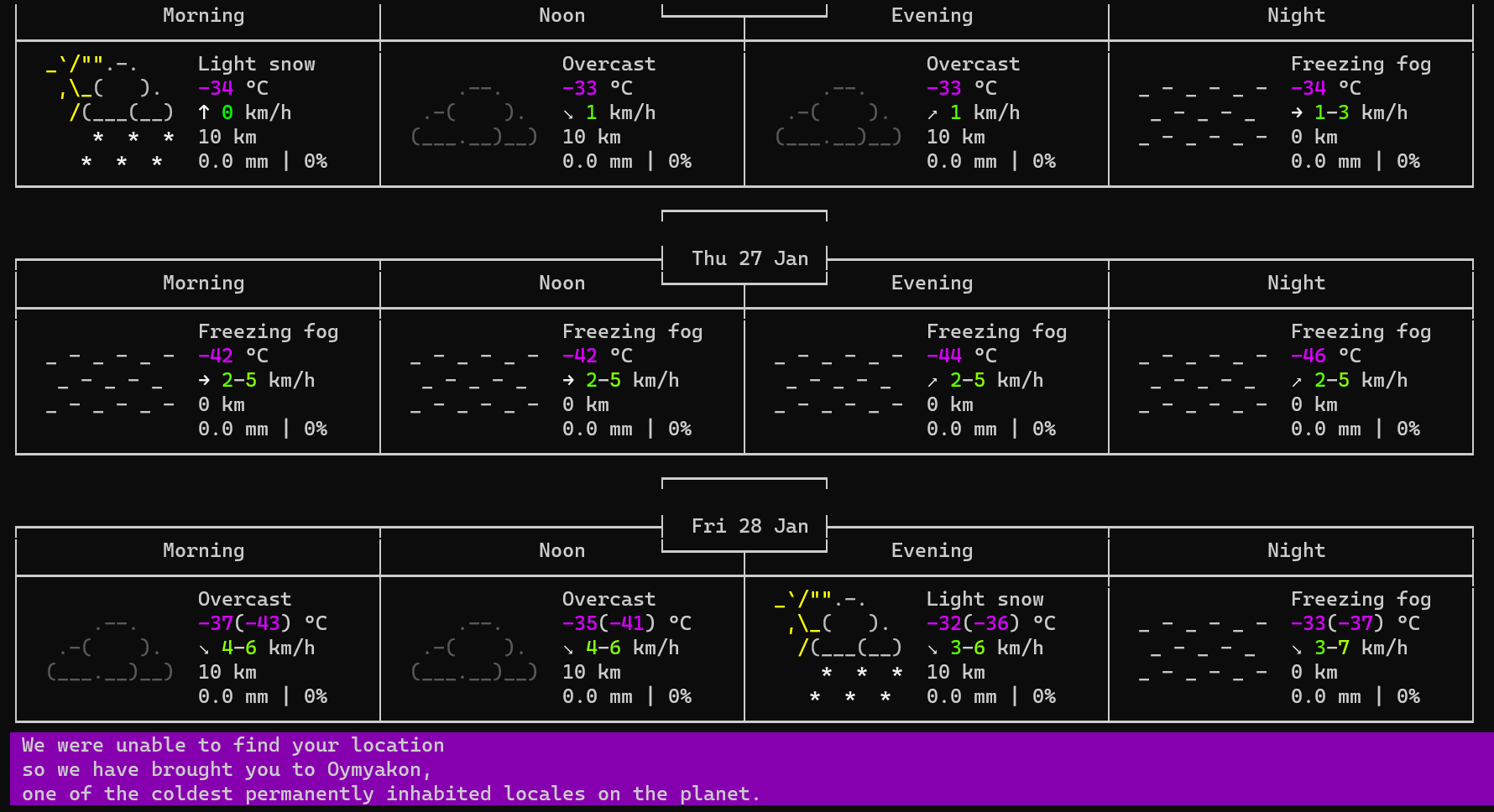
Graphical user interface, application

Description automatically generated



**5.wrong place search:**





**ADDITIONAL KNOWLEDGE ACQUIRED**

* Python GUI programming using the module tkinter.
* Using the requests module for getting details from URL.
* Use case diagrams.
* Activity diagram.

**CONCLUSION AND FUTURESCOPE:**

Finally I conclude that our project is extremely easy to adopt as they don’t require any specific computational power like Deep Learning methods (RNN, CNN … ).Nonetheless, ****predictions perfectly fit in the error range designed by the dataset itself****. It is important to consider that we only have examined monthly average values while it may be interesting to consider daily values too and have daily predictions.

One result of the unprecedentedly rapid advances in science and technology which we are experiencing at the present time is a widespread awareness of change and a preoccupation with the future.We hope our project helps the future in following ways:

1. As machine learning is used in our project ,it helps meteorological experts to give better information to predict agriculture output and natural disasters
2. The unpredictable global and micro weather patterns which are emerging can be measured by our project if we develop it in future.
3. By adding still more weather data and sensing parameters in the future we can also predict the crop health and yield on the geographical location.

4.In future government and other private agencies can also use our project if we will be able to add some more meteorological predictions.

**REFERENCES –**

Basic Python

1. Course covered during 2nd semester by Dr.Srinivas Chakravarthy
2. Ppts and handouts provided by the sir.
3. Python Programming - Using Problem Solving Approach First Edition by Reema Thareja.

tkinter Tutorial :

1. https://docs.python.org/3/library/tkinter.html
2. https://www.tutorialspoint.com/python/python\_gui\_programming.htm
3. https://www.youtube.com/watch?v=YXPyB4XeYLA&t=1s&ab\_channel=freeCodeCamp.org

requests Tutorial :

1. https://www.geeksforgeeks.org/python-requests-tutorial/#:~:text=Requests%20library%20is%20one%20of,URI%2C%20it%20returns%20a%20response.
2. https://realpython.com/python-requests/
3. https://www.w3schools.com/python/module\_requests.asp

Use cases and activity diagram

1. Hand-outs provided by SIREESHA ma’am.
2. <https://www.youtube.com/watch?v=zid-MVo7M-E>