Assignment: Python Programming for GUI Development

Name : Prakalya p v

Register Number :192311418

Department :C.S.E

Date of Submission :26-08-2024

Problem 1: Real-Time Weather Monitoring System

Scenario:

You are developing a real-time weather monitoring system for a weather forecasting company. The system needs to fetch and display weather data for a specified location.

Tasks:

1. **Model the data flow for fetching weather information from an external API and displaying it to the user.**
2. **Implement a Python application that integrates with a weather API (e.g., OpenWeatherMap) to fetch real-time weather data.**
3. **Display the current weather information, including temperature, weather conditions, humidity, and wind speed.**
4. **Allow users to input the location (city name or coordinates) and display the corresponding weather data.**

Deliverables:

- Data flow diagram illustrating the interaction between the application and the API.
- Pseudocode and implementation of the weather monitoring system.
- Documentation of the API integration and the methods used to fetch and display weather data.
- Explanation of any assumptions made and potential improvements.

Introduction:

. Describe the purpose of the real-time monitoring system.

. Explain the problems it aims to solve or the improvements it provides over existing solutions.

Components:

. data sources ,data ingestion ,processing Engine ,storage ,User Interface

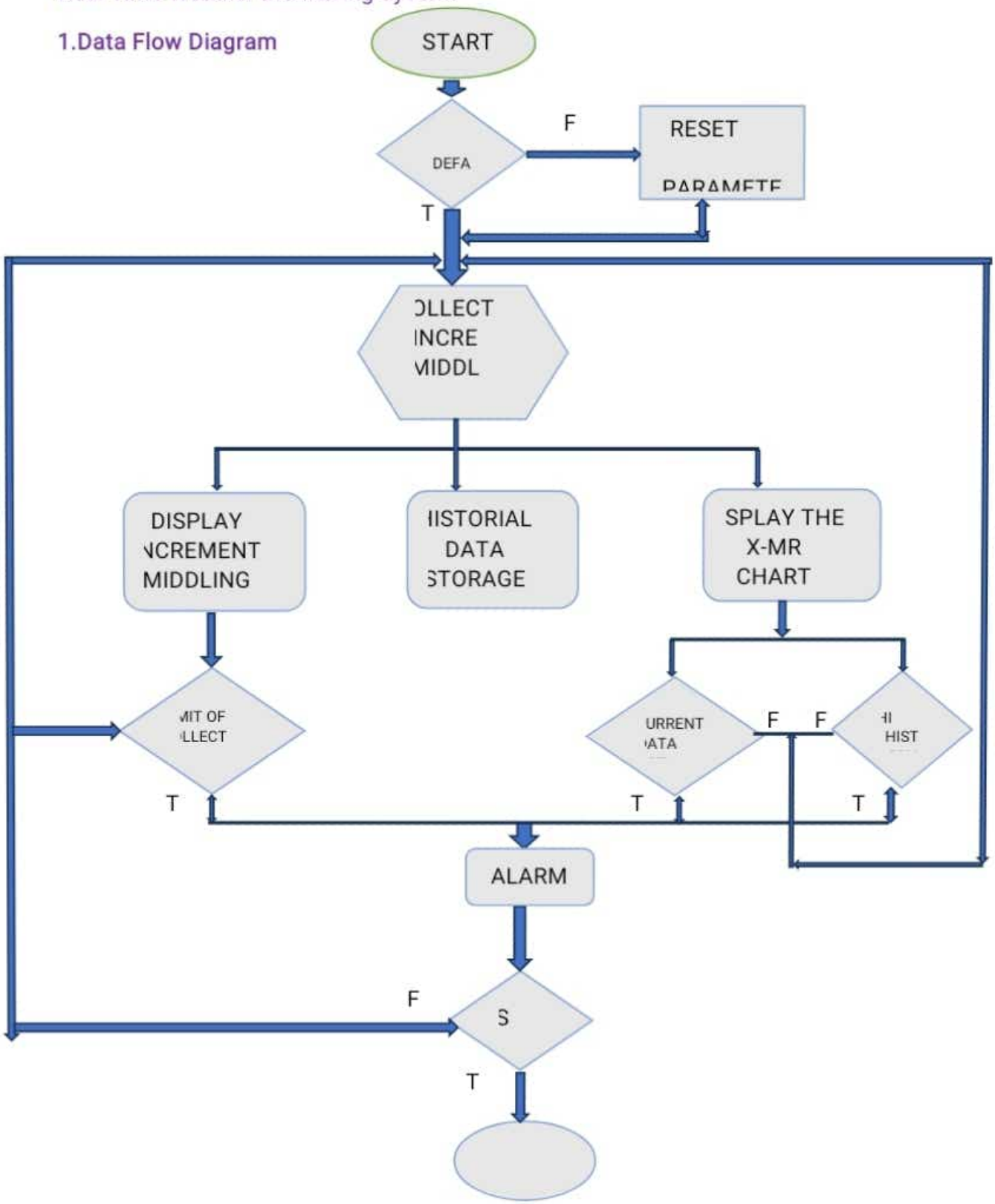
Maintenance:

. Outline regular maintenances tasks and schdules.

. Provide guidelines for updating and upgrading the system

Real-Time Weather Monitoring System

1.Data Flow Diagram



2. Implementation

```
   import requests
api_key="0b563bad1d7970f0a7d1a2d766f968f2"
user_input=input("enter the city name:")
weather_data=requests.get(

 f"https://api.openweathermap.org/data/2.5/weather?q={user_input}&units=imperial&AP
PID={api_key}")
if weather_data.json()["cod"]=="404":
   print("city not found")
else:
   #print(weather_data.json())
   weather=weather_data.json()["weather"][0]["main"]
   temp=round(weather_data.json()["main"]["temp"])
   humidity=weather_data.json()["main"]["humidity"]
   windspeed=weather_data.json()["wind"]["speed"]
   country=weather_data.json()["sys"]["country"]
   print(f"the weather in {user_input} is {weather}")
   print(f"the temperture in {user_input} is {temp}F")
   print(f"the humidity in {user_input} is {humidity}")
   print(f"the windspeed in {user_input} is {windspeed}kmph")
   print(f"the {user_input} is in {country}")
```

3.Display the Current weather information

```
enter the city: Kurnool
Temperature (in kelvin unit) = 304.42
atmospheric pressure (in hPa unit) = 1002
humidity (in percentage) = 53
description = overcast clouds
```

4.User Input

```
import requests
api_key="0b563bad1d7970f0a7d1a2d766f968f2"
user_input=input("enter the city name:")
weather_data=requests.get(
   f"https://api.openweathermap.org/data/2.5/weather?q={user_input}&units=imperial&APPID={api_key}")
if weather_data.json()["cod"]=="404":
    print("city not found")
else:
    #print(weather_data.json())
    weather=weather_data.json()["weather"][0]["main"]
    temp=round(weather_data.json()["main"]["temp"])
    humidity=weather_data.json()["main"]["humidity"]
    windspeed=weather_data.json()["wind"]["speed"]
    country=weather_data.json()["sys"]["country"]
    print(f"the weather in {user_input} is {weather}")
    print(f"the temperture in {user_input} is {temp}F")
    print(f"the humidity in {user_input} is {humidity}")
    print(f"the windspeed in {user_input} is {windspeed}kmph")
    print(f"the {user_input} is in {country}")
```

```
enter the city name:kurnool
the weather in kurnool is Clouds
the temperture in kurnool is 87F
the humidity in kurnool is 55
the windspeed in kurnool is 20.62kmph
the kurnool is in IN
```

5.Documentation

API Overview

- Describe the purpose of the API and its main functionalities.

Endpoints

- List and describe the available API endpoints.
- Include request and response formats, parameters, and example requests.

Authentication

- Explain the authentication methods used for accessing the API.

Appendices

Glossary

- Define key terms and acronyms used in the documentation.

References

- List any references or additional resources related to the system.

Change Log

- Maintain a record of changes and updates to the documentation.

Contact Information