DESKTOP NOTIFICATION USING PYTHON

ADVANCED PROGRAMMING PRACTICES –18CSC207J

MINI-PROJECT – CT4

Department and Branch: NWC, CSE

Register Numbers: RA2111029010009

Section: Q2

BONAFIDE CERTIFICATE

Register No. RA2111029010009 Certified to be the bonafide work done by Meenakshi Gayathri S of II Year/IV Sem B.Tech Degree Course in the Practical Advanced Programming Practices 18CSC207J in SRM INSTITUTE OF SCIENCE AND TECHNOLOGY, Kattankulathur during the academic year 2022 – 2023.

Faculty in-charge:

Mrs.Parimala.G

Assistant Professor

SRM Kattankulathur

ABSTRACT

The purpose of this project is to develop a Python script that displays desktop alerts for the most recent news items from an RSS feed. The script retrieves and parses the XML content of the RSS feed using the requests and feedparser libraries before extracting the most recent news headlines. The title and synopsis of each news story are shown in the alerts using the plyer library. The script continually checks for fresh content and loops over alerts for each new headline. This initiative offers quick and rapid access to news information for those who wish to keep current on the most recent news without continually checking news websites or applications.

DESKTOP NOTIFIER IN PYTHON

The desktop notifier in python is a simple application that gives us the notification message in the form of pop-ups on our desktops or laptops.

This project aims at producing the notifications regarding the weather or climate in our respective cities.

These notifications can be triggered by various events, such as new emails, social media updates, or weather alerts. Python is a popular programming language for building desktop notifiers due to its simplicity, readability, and extensive library support.

Python desktop notifiers typically use third-party libraries such as **plyer**, **win10toast**, **py-notifier**, or **notify2** to display notifications on the desktop. These libraries provide a unified interface to interact with the native notification systems of different operating systems, such as Windows, Linux, or macOS.

Desktop notifiers can be customized to display notifications with various options, such as the title, message, icon, duration, sound, and actions. They can also be programmed to automatically check for updates or events and display notifications in real-time, providing a convenient way to stay informed without constantly checking different apps or websites.

Python desktop notifiers can be useful for a wide range of applications, such as news alerts, weather updates, system notifications, reminder apps, social media monitors, and more. They are also easy to create and customize, making them a popular choice for beginner and advanced Python developers alike.

IMPLEMENTATION

```
import requests
from plyer import notification
import time

API_KEY = "YOUR_API_KEY"
WEATHER_ENDPOINT = "http://api.openweathermap.org/data/2.5/weather"
city_name = "New York"
def get_weather():
    # API request parameters
    params = {
        "q": city_name,
        "appid": API_KEY,
        "units": "metric"
    }

    response = requests.get(WEATHER_ENDPOINT, params=params)
   data = response.json()
    weather_description = data['weather'][0]['description']
    temperature = data['main']['temp']

    title = f"Weather update for {city_name}:"
    message = f"{weather_description}, {temperature}°C"

    # Displaying desktop notification
    notification.notify(
```

```
        title=title,
        message=message,
        timeout=10
    )

while True:
    get_weather()
    time.sleep(3600)  # 1 hour in seconds
```

SAMPLE OUTPUT

Notification:

Weather update for New York:
overcast clouds, 14.7°C

CONCLUSION

In this project, we showed how to create a weather notifier that shows real-time weather updates for a particular place on the desktop using Python and the Open WeatherMap API. This example may be tweaked and expanded to develop notifiers for a variety of applications, including news alerts, system notifications, and social media monitoring. Overall, the desktop notifier project in Python offers a practical and adaptable method to remain updated on significant events and developments, making it a useful tool for developers, organisations, and people alike.