

**Proiect Python**

Student: Savin Diana-Ioana

Anul: III

Cuprins

[1. Descriere 3](#_Toc187688965)

[2. Caracteristici 3](#_Toc187688966)

[3. Cerințe 3](#_Toc187688967)

[4. Instalare 4](#_Toc187688968)

[5. Utilizare 4](#_Toc187688969)

[6. Fișiere 4](#_Toc187688970)

[7. Fluxul general de funcționare 5](#_Toc187688971)

[8. Gestionarea erorilor 5](#_Toc187688972)

[9. Observații 6](#_Toc187688973)

[10. Bibliografie 6](#_Toc187688974)

# Descriere

Weather App este o aplicație desktop pentru prognoză meteo. A fost constuită folosind Python și biblioteca Tkinter. Aceasta oferă date meteo, dar și ora exactă, în timp real, pentru orice oraș introdus de utilizator. Datele meteo sunt preluate de la API-ul OpenWeatherMap ce afișează date precum temperatura, viteza vântului, umiditatea și presiunea atmosferică.

# Caracteristici

Principalele caracteristici ale aplicației sunt:

* **Căutare oraș**: introduceți numele unui oraș și aplicația returnează detalii despre vremea curentă
* **Ceas în timp real**: afișează ora curentă din orașul introdus.
* **Detalii meteo**: include temperatura, condițiile meteo, viteza vântului, procentajul de umiditate și presiunea atmosferică
* **Interfața prietenoasă**: design modern și simplu cu pictograme clicabile.

# Cerințe

Pentru rularea proiectului sunt necesare:

**Bibliotecile:**

* tkinter – crearea interfeței grafice
* geopy – obtinerea locației orașului introdus
* timezonefinder – determinarea fusului orar pe baza zonei geografice
* pytz – gestionarea obiectelor datetime specifice fusului orar
* requests – efectuează solicitării API către OpenWeatherMap
* pillow – procesarea imaginilor

**Cheia API:**

Pentru a rula corespunzător, aplicația are nevoie de o cheie API de la OpenWeatherMap (<https://openweathermap.org>) din care va lua datele necesare legate de vreme.

# Instalare



În vederea realizării proiectului trebuie instalate următoarele instrumente:

**Pasul 1:** Se va instala cea mai recentă versiune a aplicației Python (Python 3.13.1 la momentul întocmirii documentului; <https://www.python.org>).

**Pasul 2:** Se deschide Command Prompt-ul pentru instalarea librăriilor. Pentru acastă operație se vor folosi următoarele comenzi:

python -m pip install tkinter – crearea interfeței grafice

python -m pip install geopy – obtinerea locației orașului introdus

python -m pip install timezonefinder – determinarea fusului orar pe baza zonei geografice

python -m pip install pytz – gestionarea obiectelor datetime specifice fusului orar

python -m pip install requests – efectuează solicitării API către OpenWeatherMap

python -m pip install pillow – procesarea imaginilor

**Pasul 3:** Se descarcă fișierele necesare (da2.png, search\_icon.png, logo.png, box1.png) în același director în care va fi realizat fișierul de cod.

# Utilizare

Pașii de utilizare ai aplicației sunt:

**Pasul 1:** Lansați aplicația.

**Pasul 2:** Introduceți numele unui oraș în bara de căutare.

**Pasul 3:** Faceți click pe butonul de căutare.

**Pasul 4:** Vizualizați detaliile meteo furnizate de aplicație.

# Fișiere

Fișierele necesare programului sunt:

* *proiect.py*: scriptul principal Python care conține codul aplicației
* *da2.png*: imaginea de fundal pentru bara de căutare
* *search\_icon.png*: pictograma pentru butonul de căutare
* *logo.png*: imaginea centrala a aplicației
* *box1.png*: imaginea de fundal pentru secțiunea inferioară care afișează detalii meteo

# Fluxul general de funcționare

1. Introducerea orașului:

* Utilizatorul introduce numele orașului în bara de căutare din interfața grafică

1. Preluarea locației:

* Funcția getWeather() este declanșată când utilizatorul apasă butonul de căutare
* Utilizând biblioteca geopy, aplicația convertește numele orașului într-o locație geografică

1. Determinarea fusului orar:

* Biblioteca timezonefinder este utilizată pentru a identifica fusul orar al locației specifice
* Biblioteca pytz determina și afișează ora locală

1. Prelucrarea datelor API:

* Datele JSON returnate de API sunt analizate pentru a extrage informații precum: temperatura, condițiile meteo, viteza vântului, procentajul de umiditate și presiunea atmosferică

1. Afișarea datelor:

* Datele extrase sunt afișate în secțiunile predestinate ale interfeței folosind widgetul Label din Tkinter

# Gestionarea erorilor

În cazul în care numele orașului este introdus greșit sau API-ul este invalid, aplicația va afișa un mesaj de eroare folosind un mesaj de tip Tkinter messagebox.

# Observații

* Fișierele *da2.png* și *box1.png* sunt create în Adobe Photoshop (<https://www.adobe.com/ro/>)
* Link-ul

# Bibliografie

<https://aiit-sa.co.za/product/programming-with-python/>

<https://www.vecteezy.com/vector-art/9652218-vector-magnifying-glass-icon-isolated-on-white-background-search-illustration> - search icon

<https://tierarzt-karlsruhe-durlach.de/en/weather-sensitivity-in-cats/> - logo