**ROMÂNIA**

**MINISTERUL APĂRĂRII NAȚIONALE**

**ACADEMIA TEHNICĂ MILITARĂ “FERDINAND I”**

**FACULTATEA DE SISTEME INFORMATICE ȘI SECURITATE CIBERNETICĂ**

**Specializarea: Calculatoare şi sisteme informatice pentru apărare şi** **securitate naţională**



**Clothing Selector**

**Proiect Realizat de:**

**Cujbă Mihai**

**Florea Vlad**

**Nișulescu Alexandru**

**BUCUREȘTI**

**2020**

**Introducere**

Proiectul își propune implementarea unei aplicații web pentru alegerea hainelor corespunzatoare pentru vremea din locația dorită. Pentru partea de interfață cu utilizatorul, proiectul o să folosească html și css pentru design, iar pentru server și backend va fi folosit Phyton 3 împreună cu Flask. Pentru datele despre vreme o să folosim API-ul pus la dispoziție de OpenWeatherMap.

Obiectivul aplicației este ca în urma selectării locației dorite, aplicația să estimeze cât mai corect hainele pe care utilizatorul să le aleagă pentru a nu îi fi cald sau frig.

Pentru alegerea tinutei corespunzătoare a fost folosit algoritmul Hill Climbing.

În analiza numerică, hill climbing este o tehnică de optimizare matematică care aparține familiei căutării locale. Este un algoritm iterativ care începe cu o soluție arbitrară la o problemă, apoi încearcă să găsească o soluție mai bună făcând o schimbare incrementală a soluției. Dacă modificarea produce o soluție mai bună, se face o altă modificare incrementală a noii soluții și așa mai departe până când nu se mai pot găsi îmbunătățiri suplimentare. Simplitatea relativă a algoritmului îl face o alegere populară printre algoritmi de optimizare. Este utilizat pe scară largă în inteligența artificială, pentru atingerea unei stări de obiectiv de la un nod de pornire. Diferite opțiuni pentru nodurile următoare și nodurile de pornire sunt utilizate în algoritmi asociați.

**Descrierea problemei**

Pentru implementarea aplicației, aceasta a fost spartă în trei părți: aplicația web, algoritmul de hill climbing și scripturile pentru prelucrarea datelor despre vreme și haine. Aplicația web trebuie să trimită datele selectate de utilizator către scriptul care va face un request la serverul de la OpenWeatherMap pentru prognoza meteo. După primirea datelor de la acesta, aplicația va folosi algoritmul Hill Climbing pentru a decide care dintre obiectele vestimentare din baza noastra de date este cea mai potrivită. Odată alese, interfața din browserul web îi va arăta utilizatorului care sunt aceste haine, folosind imaginile din baza de date.

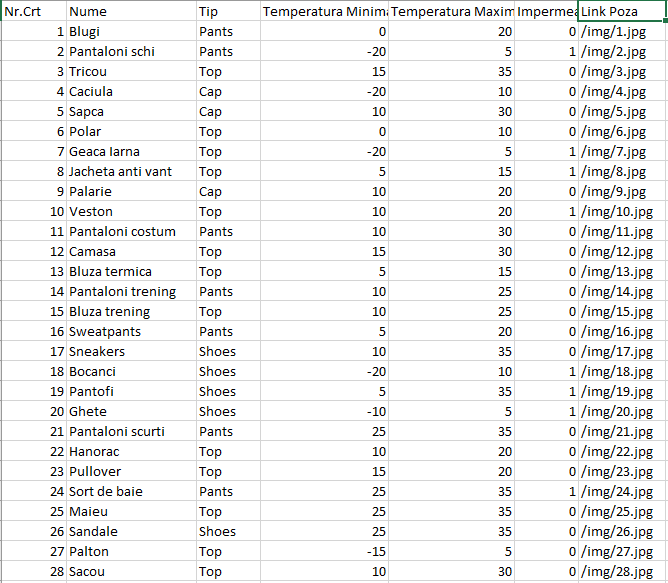
 Baza de date cu haine este sub forma unui csv. Acesta are 7 coloane, și anume: numarul articolului, denumirea acestuia, tipul, temperatura minimă la care acesta poate fi purtat, temperatura maxima, impermeabilitatea și calea către imaginea cu acesta. Algoritmul folosit trebuie să decidă pentru fiecare dintre tipurile de haine care este cea mai indicată pentru a fi purtată. Acesta va ține cont de temperaturile între care aceasta poate fi purtată, dar și de impermeabilitatea lor, în cazul în care afară sunt precipitații. În figura următoare puteți vedea formatul și intrările din fișierul csv:

Figure Fișierul CSV cu datele despre haine

**Descrierea implementarii**

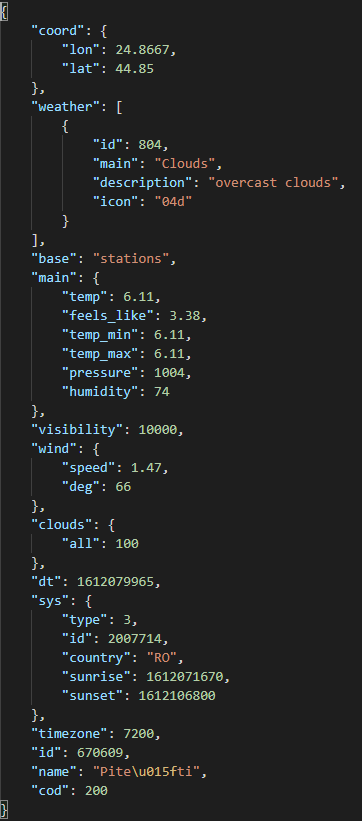
 Interfața web va prezenta utilizatorului un searchbar în care acesta va trebui să introducă numele orașului căutat, dar, mai sus acesta poate căuta numele orașului și cu ajutorul Google Maps. După ce utilizatorul apasă tasta ENTER, serverul va porni scriptul WeatherGather care va face un request la API-ul de la OpenWeatherMap, care va returna un fișier de tip JSON. Acest fisier este salvat în fișierul weather.json, pentru citirea lui ulterioară. Structura acestui fișier este următoarea:

Figure Structura fișierului JSON

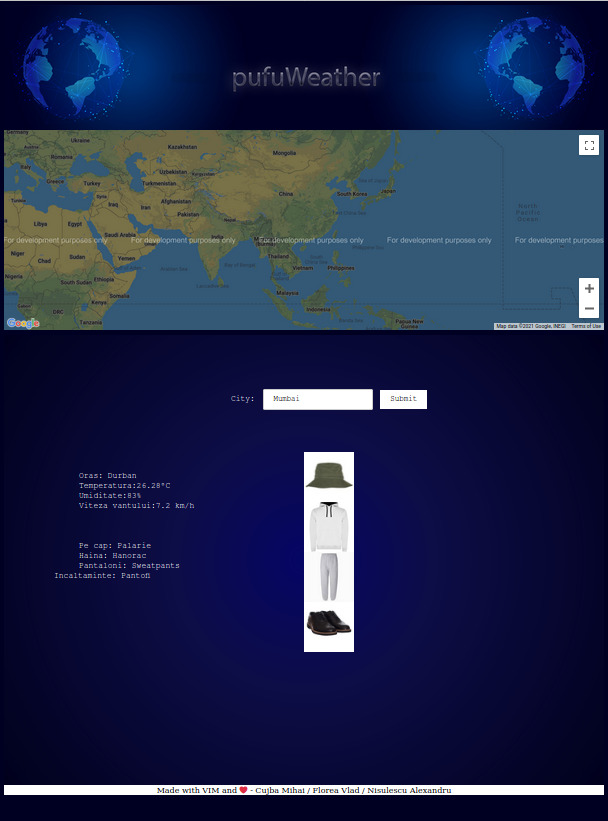
 Pentru citirea acestuia se foloseste scriptul WeatherParser, care returnează datele de pe vreme, ce urmează sa fie afișate în interfața web și mai apoi scriptul app.py o să pornească scriptul outfitFinder. Furnizându-i datele despre vreme și calea către fișierul csv cu baza de date despre haine, acesta returnează serverului pentru fiecare tip de haine (Cap, Top, Pants și Shoes) varianta pe care o consideră cea mai potrivită. După primirea datelor, serverul afișează o imagine cu fiecare dintre acestea. În figura următoare puteți observa interfața web în urma selectării orașului Durban:

Figure Selectiile de îmbracăminte

Pentru mai multe detalii de implementare, sau codul proiectului puteți vizita: <https://github.com/AlexNisulescu/CloathingSelector>.

**Algoritmul de Hill Climbing**

Cea mai importantă piesă din acest proiect o reprezintă algoritmul de Hill Climbing. Folosind datele primite, acesta selectează hainele corespunzătoare, făcând funcțională principala funcționalitate a aplicației.

O configurație în spațiul de stări al hainelor este reprezentată de o combinație între următorii 4 itemi: îmbrăcaminte cap, îmbrăcaminte top, îmbrăcaminte pantaloni, îmbrăcaminte pantofi.

Fiecare item de îmbrăcăminte este caracterizat prin următorii parametrii: temperatura minimă, temperatura maximă și impermeabilitate.

Pe baza acestor caracteristici putem să comparăm mai multe configurații raportat la vremea curentă și să determinam configurația cea mai potrivita.

Pentru acest proiect am folosit următorii parametrii caracteristici

* Temperatura
* Umiditate – procent
* Vânt – m/s

Am folosit urmatoarea formulă pentru a determina valoarea relativă a temperaturii raportată la cum se simte:

***TempRelativa= temperatura- 0.2\*vant-(temperature\*umiditate)/2***

Pentru a determina o configurație potrivită, avem nevoie de o metrică prin intemediul căreia să determinăm care item de îmbrăcăminte este mai bun.

Pentru aceasta am folosit următoarea formulă pentru a determina eroarea unui item față de temperatura curenta:

***Err = abs((tempMin+tempMax/2)-TempRelativa)*** 🡪 Eroarea fata de mijlocul intervalului de temperature

În eventualitatea în care plouă, la eroarea calculată pentru fiecare item, dacă acesta nu este impermeabil se măreste eroarea cu 20%.

Pentru fiecare configuratie se calculeaza eroarea totala:

***Eroarea=errRelativaItemCap\*ImportantaItemCap+errRelativaItemTop\*ImportantaItemTop***

Pentru folosirea algoritmului Hill Climbing se pleacă de la o configurație inițială aleasă aleator.

Un pas reprezintă o schimbare într-una din urmatoarele direcții:

* Item cap
* Item pantaloni
* Item top
* Item pantofi

Se încearcă un pas într-una din direcții, se calculează eroarea configurației și se compară cu eroarea configurației curentă. Daca noua eroare este mai mică se păstrează configurația și se repetă algoritmul.

Dacă noua eroare nu este mai mică, se încearcă un pas într-o altă direcție.

Dacă am încercat toate direcțiile și nu am găsit o nouă eroare mai mică, înseamnă că eroarea curentă este un minim local și că am găsit o configurație potrivită.

După un număr prestabilit de iterații, algoritmul se oprește și returnează cea mai mică eroare găsită până la acel moment.  
 Pentru a încerca găsirea unui minim global, față de unul local, se rulează algoritmul Hill Climbing de mai multe ori plecând din configurații initiale diferite, la final, alegându-se eroarea minimă dintre acestea.

**Concluzii**

În concluzie, după testarea lui de către membrii echipei, proiectul funcționează conform așteptărilor. Acesta furnizează articole de îmbrăcăminte corespunzătoare temperaturii și precipitațiilor de afară, pentru locația dorită.

Algoritmul Hill Climbing face o aproximare foarte bună a articolelor de îmbrăcăminte ce ar trebui purtate, conform bazei noastre de date, dar și conform intuiției umane, care ne-ar spune să purtăm același lucru pentru vremea respectivă. Această aplicație își rezolvă cu succes scopul final, și anume, acela de a elimina grija utilizatorului asupra vestimentației. Un lucru de luat în considerare, totuși, este acela că aplicația nu ține cont de ultimele tendințe în vestimentație.

**Bibliografie**

1. Cursurile și laboratoarele de Inteligență Artificială
2. <https://en.wikipedia.org/wiki/Hill_climbing>
3. <https://openweathermap.org/>
4. <https://thetravelinsider.co/th/en/travel-inspirations/travel-hacks/what-to-pack-and-wear-for-different-winter-climates-10-c-0-c-0-c>
5. <https://www.pinterest.com/pin/165225880057794781/>