Documentatie Proiect Python

**Tema proiectului:**

Site pentru toate armele Legendare din Borderlands 2

**Tip de proiect:**

Pentru realizarea acestui proiect am randat pagini Web cu ajutorul modulului Flask si am create o baza de date prin modulul FireBase.

**Rulare program:**

Trebuie creata o instanta a clasei flask pentru a putea randa pagini Web si o instanta a clasei firebase prin care vom prelua informatiile din baza de date. Vom instala modulul “Flask” si “firebase”.  Pentru a fi instalate se introduc urmatoarele comenzi in terminal:

***python -m pip install flask***

***python -m pip install firebase***

In fisierul app.py trebuie sa cream o instanta a clasei Flask ( app=Flask(\_\_name\_\_ )) pentru a putea crea & rula site-ul pe care l-am facut. Pentru a crea o pagina Web (sau pentru a o randa) am folosit urmatoarea sintaxa:

**@app.route(”url-ul paginii respective”)**

**def NumelePaginiCreate(argumente daca este nevoie):**

**return ”Ce vrem sa scrie avem in pagina respectiva”**

De exemplu pentru prima pagina Web (a.k.a. Home) functia arata in felul urmator:

**@app.route(”/”)**

**def Home():**

**return render\_template(”home.html”)** ””” functia render\_template(”numePaginaHTML.html”) este o functie din Flask prin care randam prima pagina Web prin ”importarea” unei pagini web facute in HTML. ”””

In prima pagina (home.html) am creat diferite butoane pentru diferite tipuri de arme din jocul Borderlands 2 pentru a ne redirectiona catre o lista tipul armei respectiv. Pentru acest lucru ne-am folosit de o baza de date, firebase, pentru a crea template-uri HTML pentru fiecare tip de arma in parte, iar apoi pentru fiecare tip de arma am creat cate o pagina pentru fiecare pistol, assault riffle,etc. Baza de date sa gaseste in fisierul proiectului ca ” catalinz-default-rtdb-export.json” care trebuie uploadat in cloud intr-un proiect FireBase la adresa: <https://console.firebase.google.com/>

In fisierul app.py am creat o instanta a clasei FireBase unde am conectat baza de date din cloud in codul programului cu ajutorul unui link:

**firebase=firebase.FirebaseApplication(’** [**https://catalinz-default-rtdb.europe-west1.firebasedatabase.app/**](https://console.firebase.google.com/project/catalinz/database/catalinz-default-rtdb/data/~2F)**’, None)**

Pentru pagina Web a fiecarui tip de arma am creat un template in html (”weapond\_name\_list.html”) si cate o functie separata in Flask cu numele tipului de arma respectiv unde am preluat din baza de date numele armei respective si o poza cu aceasta arma. Exemplu:

**@app.rout(”/smgs”)** # reprezinta url-ul paginii pistol

**def SMGs():**

**data = get\_weapon\_data(’smgs’)** ””” cu aceasta functie am preluat informatiile necesare din baza de date pe care le-am folosit in template-ul nostru (”weapond\_name\_list.html”) pentru a crea pagina Web de smg-uri”””

**return render\_template(”weapon\_name\_list.html”,name=’smgs’, data=data)** ”””fata de pagina Home in interiorul functiei ”render\_template” am introdus doua variabile (name si data) prin care transmitem in interirul template-ul ”weapond\_name\_list.html” informatiile necesare pe care vrem sa le folosim in aceasta pagina”””

In fuctia ”get\_weapon\_data(name)” am prealuat informatiile necesare din baza de date pe care am vrut sa le afisam in template-ul nostru. Comentarii functie:

**def get\_weapon\_data(name):** # name e tipul armei de unde luam date

**weapon\_data=firebase.get(’/’+name, None)** ”””prin intermediul acestei functii luam datele necesare din baza de date”””

**return weapon\_data** # returnam datele respective

Pentru a crea cate o pagina Web pentru fiecare arma din joc am folosit un alt template html (”weapons\_template.html”) unde am afisat diverse informatii despre o anumita arma. Pentru preluarea de informatiilor necesare din baza de date crearii acestui template am creat o alta functie prin care preluam informatiile din baza de date:

**def get\_weapon\_data2(type,id):** # type=tipul armei(ex: smg), id=numele armei

**weapon\_data=firebase.get(’/’+type+’/’+id, None)** #ca la functia anterioara

**return weapon\_data**

Fata de functiile create pentru fiecare tip de arma in Flask, pentru randarea unei pagini Web a fiecarui tip de pistol spre exemplu, am creat o functie tip template unde prin intermediul functiei ”get\_weapon\_data2(type,id)” am preluat informatiile necesare din baza de date.

Comentarii functie template Flask pentru fiecare tip de pistol, smg, etc.:

**@app.route(”/<type>/<id>”)** ””” folosim <> pentru a „templatiza” url-ul pe baza type=tipului de arma(ex: pistol) si id=nume arma(ex: Harold) ”””

**def weapon\_details(type, id):** # folosim aceleasi variabile

**data=get\_weapon\_data2(type, id)** # prin data luam info din baza de date

**return render\_template(’weapons\_template.html’, id=id, type=type, data=data)**

**Repartizarea sarcinilor de lucru:**

Zarcan Catalin-Ionut-Antonel: design site, functii flask, randare pagini site, completare baza de date (culori, fonturi, aranjarea in pagina etc.)

Alex: creare baza de date cu FireBase si implementarea pe template in html

**Research:**

https://www.youtube.com/watch?v=sugvnHA7ElY

https://www.youtube.com/watch?v=mqhxxeeTbu0&list=PLzMcBGfZo4-n4vJJybUVV3Un\_NFS5EOgX

https://code.visualstudio.com/docs/python/tutorial-flask

https://www.youtube.com/watch?v=o4XveLyI6YU

https://chat.openai.com/share/8b47009d-1693-4c8d-b329-228915549b14

https://www.lootlemon.com/db/borderlands-2/weapons

https://chat.openai.com/share/7a9da753-d9b5-4820-af4d-811080263159

**GitHub:**

https://github.com/qCatalinZ/Proiect-Python/tree/main