



UNIVERSITATEA TEHNICĂ DIN CLUJ-NAPOCA

Sistem intelligent de parcare

Inginerie Software

Autori: Filip Raul si Rad Vladut

Grupa: 30235

FACULTATEA DE AUTOMATICA
SI CALCULATOARE

Cuprins

1	Introducere	2
1.1	Caracteristici principale	2
2	Diagramele proiectului	3
2.1	Diagrama cazurilor de utilizare	3
2.2	Diagrama de activitate- intrare in parcare	4
2.3	Diagrama de comunicare	5
3	Design Pattern	5
3.1	Composite	5
3.2	Observer	6
4	Limbaje de programare	6
4.0.1	Backend	6
4.0.2	Frontend	6
5	Baza de date	6
6	Github	7
7	Manual de utilizare	8

1 Introducere

Asistentul Inteligent de Parcare: Navigarea Simplificată a Parcărilor

În agitația vieții urbane, găsirea unui loc de parcare convenabil poate fi o sarcină descurajantă. Asistentul nostru Inteligent de Parcare este o soluție de ultimă generație, destinată să revoluționeze modul în care șoferii parchează în medii urbane. Acest sistem nu doar simplifică experiența de parcare, dar se integrează și în mod fără probleme cu evenimentele și reglementările orașului.

1.1 Caracteristici principale

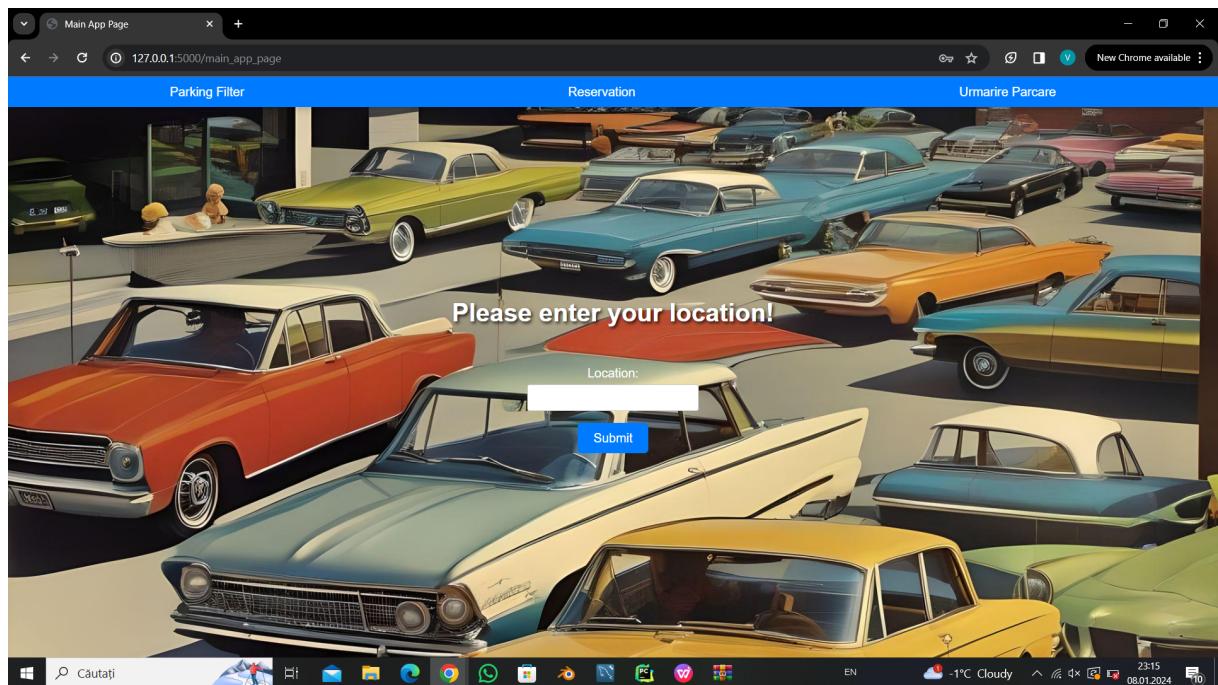
Sugestii de Parcare în Timp Real: Utilizând o hartă interactivă, sistemul oferă utilizatorilor informații în timp real despre locurile de parcare disponibile, inclusiv prețurile și disponibilitatea, pe baza locației lor sau a unui cod pin specificat.

Recomandări Bazate pe Evenimente: Sistemul sugerează în mod inteligent spații de parcare în apropierea evenimentelor la care utilizatorii intenționează să participe, evidențiind de asemenea evenimentele din apropiere care ar putea fi de interes.

Urmărire Dinamică a Locurilor Libere: Utilizatorii pot urmări dinamic locurile de parcare goale, permitându-le să-și parcheze vehiculele în cel mai apropiat slot disponibil, cu actualizări furnizate în timp real pe măsură ce locurile devin ocupate sau se eliberează.

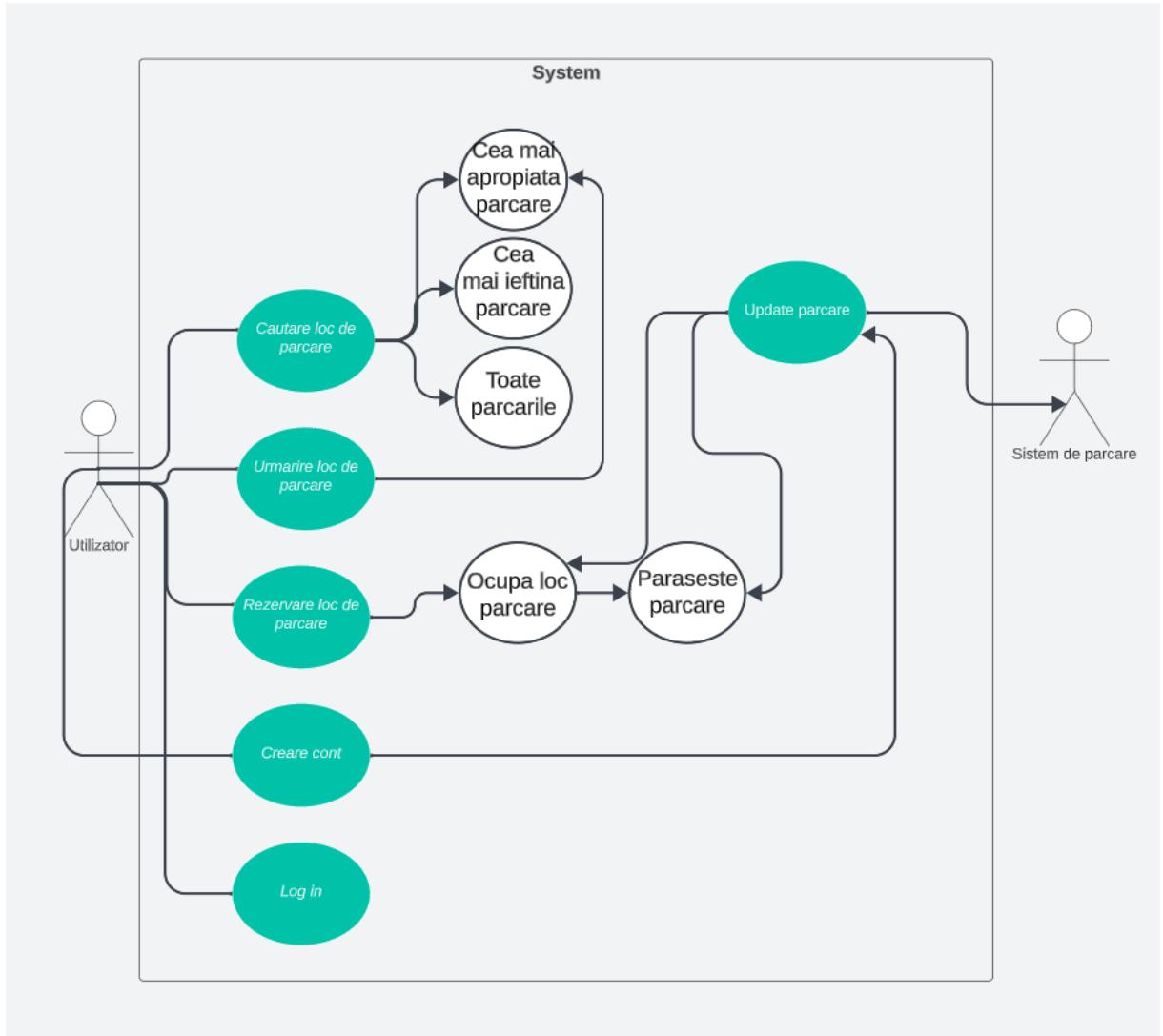
Rezervarea Locurilor de Parcare: Sistemul permite utilizatorilor să rezerve locuri de parcare înainte de a ajunge la parcarea autorizată, asigurând o experiență fără griji la sosire.

Angajamentul Comunitar: Utilizatorii pot partaja și descoperi cele mai bune locuri de parcare în cadrul comunității, promovând un mediu colaborativ.



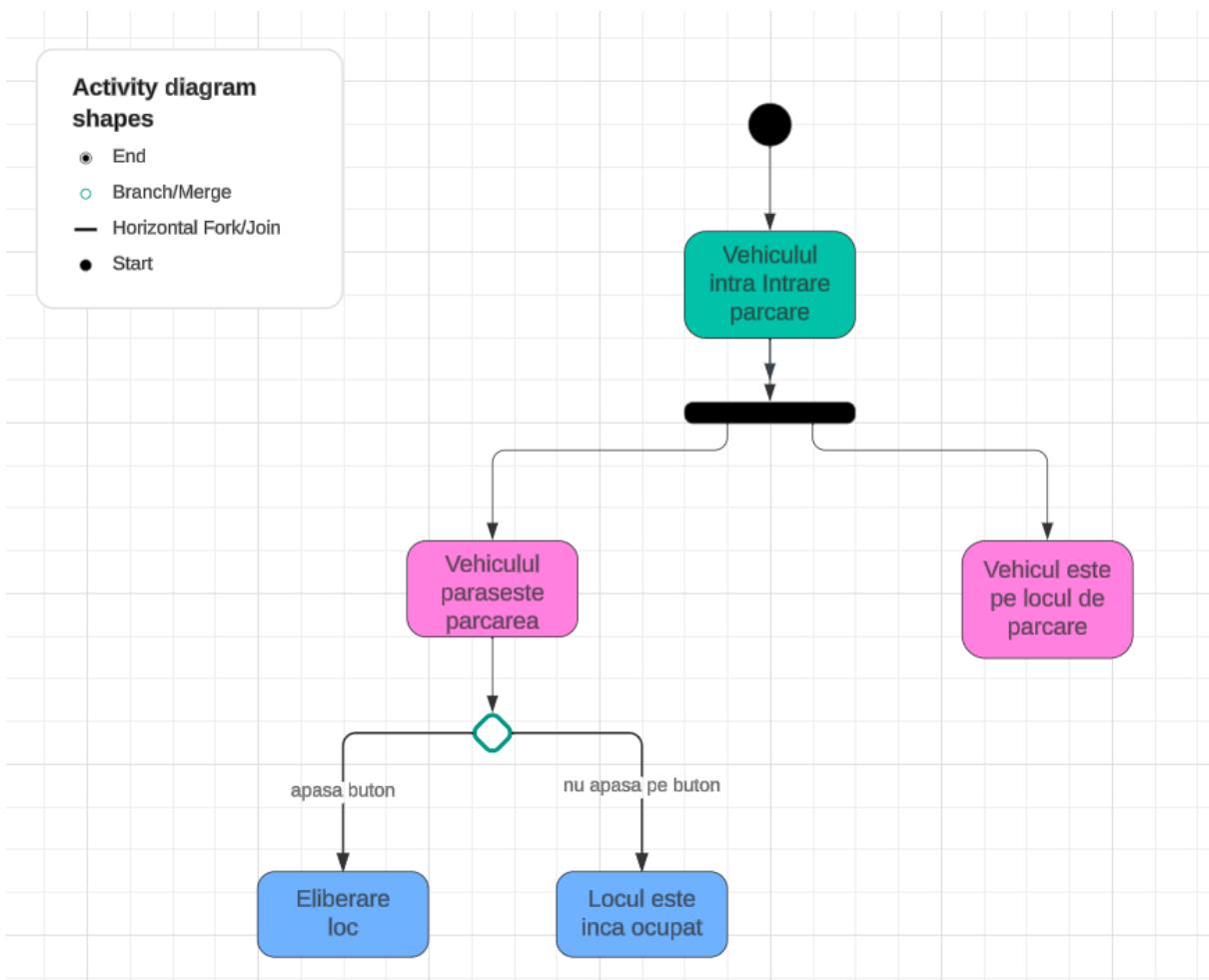
2 Diagramele proiectului

2.1 Diagrama cazurilor de utilizare



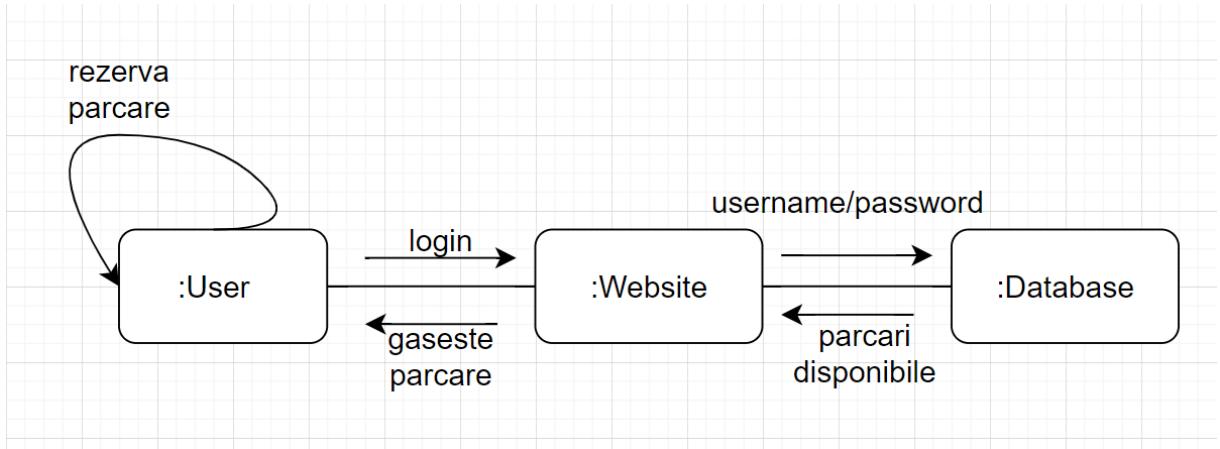
Aceasta diagrama va ajuta la înțelegerea fluxului de lucru și la identificarea tuturor funcționalităților necesare pentru sistemul de asistență la parcare

2.2 Diagrama de activitate- intrare in parcare



O diagramă de activitate bine realizată ne ajuta la vizualizarea procesului, identificând potențialele probleme sau ineficiențe și oferind o bază clară pentru dezvoltarea sistemului.

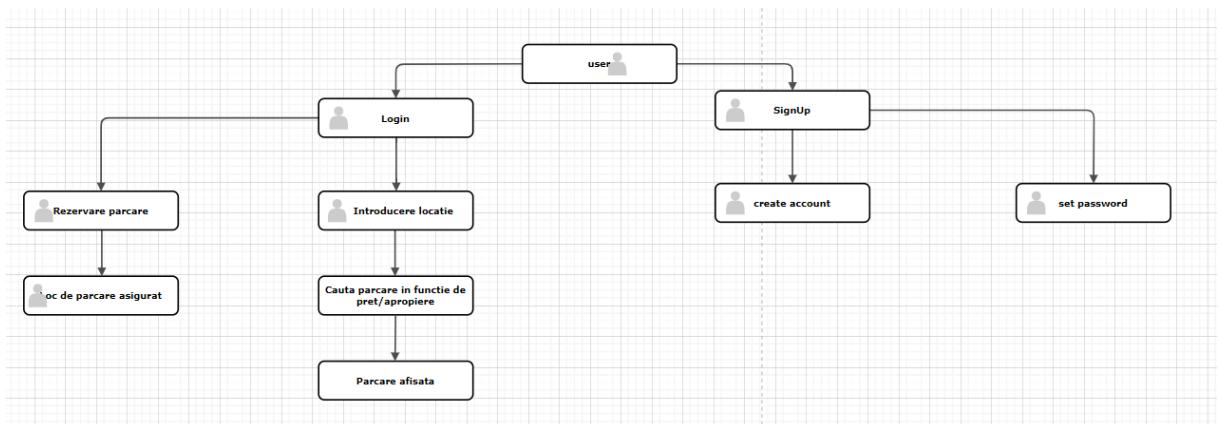
2.3 Diagrama de comunicare



Această diagramă de comunicare ilustrează modul în care diferite componente ale sistemului nostru de asistență pentru parcare interacționează între ele în timpul procesului de identificare și rezervare a unui loc de parcare.

3 Design Pattern

3.1 Composite



Utilizăm Composite Pattern pentru a reprezenta și gestiona ierarhic structura locurilor de parcare, de la nivel individual la grupuri sau zone de parcare.

3.2 Observer

În sistemul nostru de asistență pentru parcare, Observer Pattern este folosit pentru a actualiza utilizatorii cu privire la schimbările de disponibilitate ale locurilor de parcare. Observam că de fiecare dată cand un obiect se schimba, toate dependențele sale sunt notificate și actualizate automat.

4 Limbaje de programare

4.0.1 Backend

Python

Flask

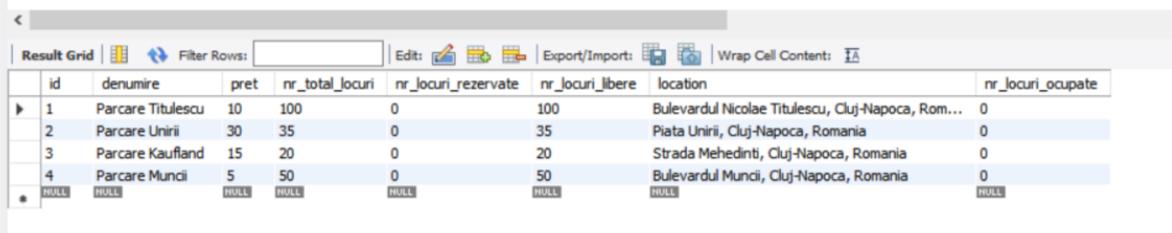
4.0.2 Frontend

HTML

JavaScript

5 Baza de date

In baza de date am introdus cateva date cum ar fi adresele de parcari din Cluj-Napoca, in functie de locatie acestea vor fi afisate dupa cerinta utilizatorului, un id unde sa verificam daca utilizatorul poate sa se logheze in aplicatie si rezervarile cu detalii specifice acestuia



A screenshot of a database grid titled "Result Grid". The grid displays information about four parking lots. The columns are: id, denumire, pret, nr_total_locuri, nr_locuri_rezervate, nr_locuri_liber, location, and nr_locuri_ocupate. The data is as follows:

	id	denumire	pret	nr_total_locuri	nr_locuri_rezervate	nr_locuri_liber	location	nr_locuri_ocupate
▶	1	Parcare Titulescu	10	100	0	100	Bulevardul Nicolae Titulescu, Cluj-Napoca, Rom...	0
▶	2	Parcare Unirii	30	35	0	35	Piata Unirii, Cluj-Napoca, Romania	0
▶	3	Parcare Kaufland	15	20	0	20	Strada Mehedinți, Cluj-Napoca, Romania	0
▶	4	Parcare Mundi	5	50	0	50	Bulevardul Mundi, Cluj-Napoca, Romania	0
*	HULL	HULL	HULL	HULL	HULL	HULL	HULL	HULL

The screenshot shows a MySQL Workbench interface with a result grid. The grid has columns labeled 'id', 'username', and 'password'. There are four rows: the first three contain data (id 1-3, username a-c, password a-c), and the fourth row is a blank header row with 'NULL' in all columns.

	id	username	password
▶	1	a	a
▶	2	b	b
▶	3	c	c
*	NULL	NULL	NULL

The screenshot shows a MySQL Workbench interface with a result grid. The grid has columns labeled 'idrezervare', 'iddient', and 'idparcare'. There are four rows: the first three contain data (id 1-3, iddent 1, idparcare 2), and the fourth row is a blank header row with 'NULL' in all columns.

	idrezervare	iddient	idparcare
▶	1	1	2
*	NULL	NULL	NULL

6 Github

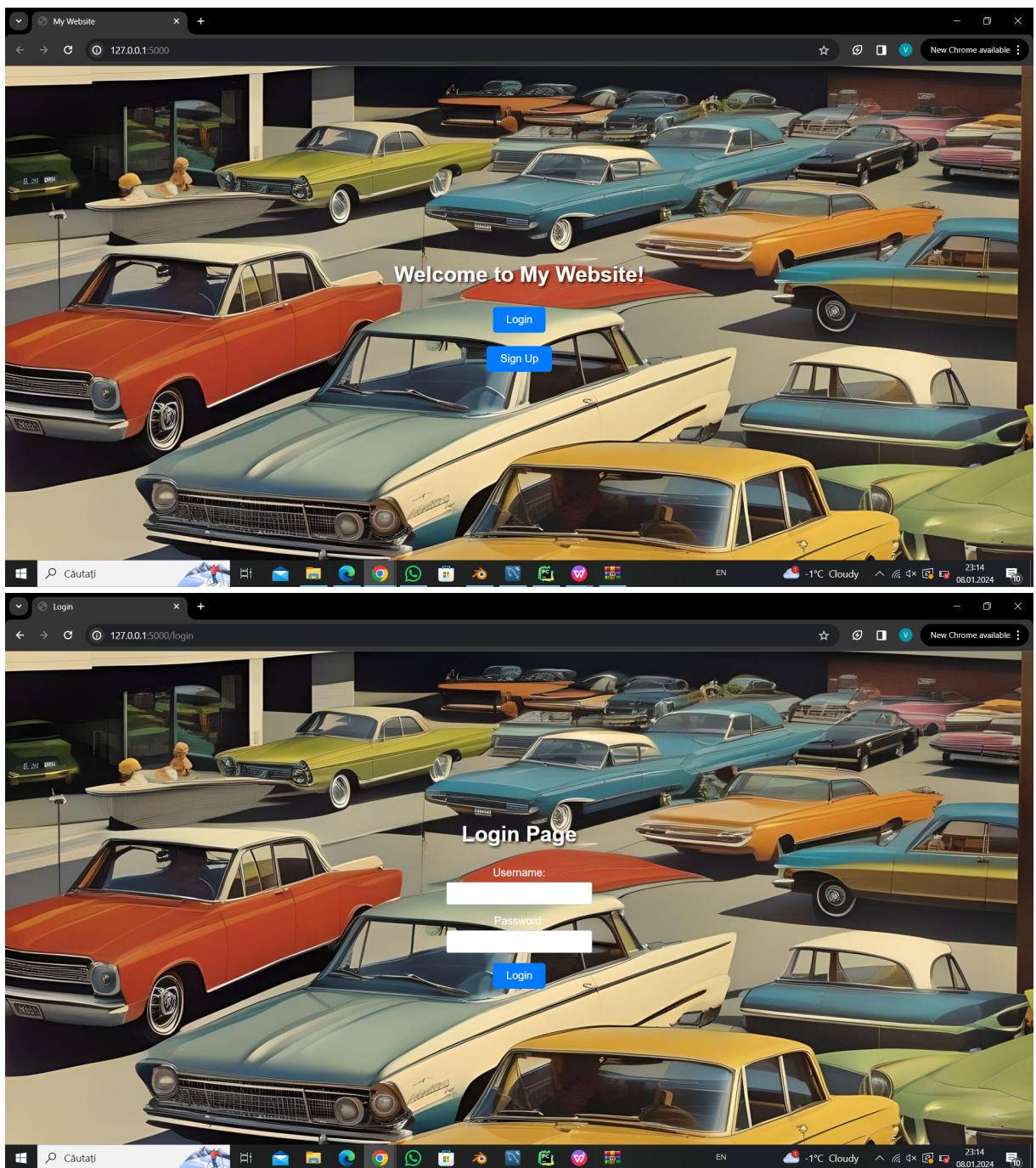
<https://github.com/raulfilip/ProjectIS>

7 Manual de utilizare

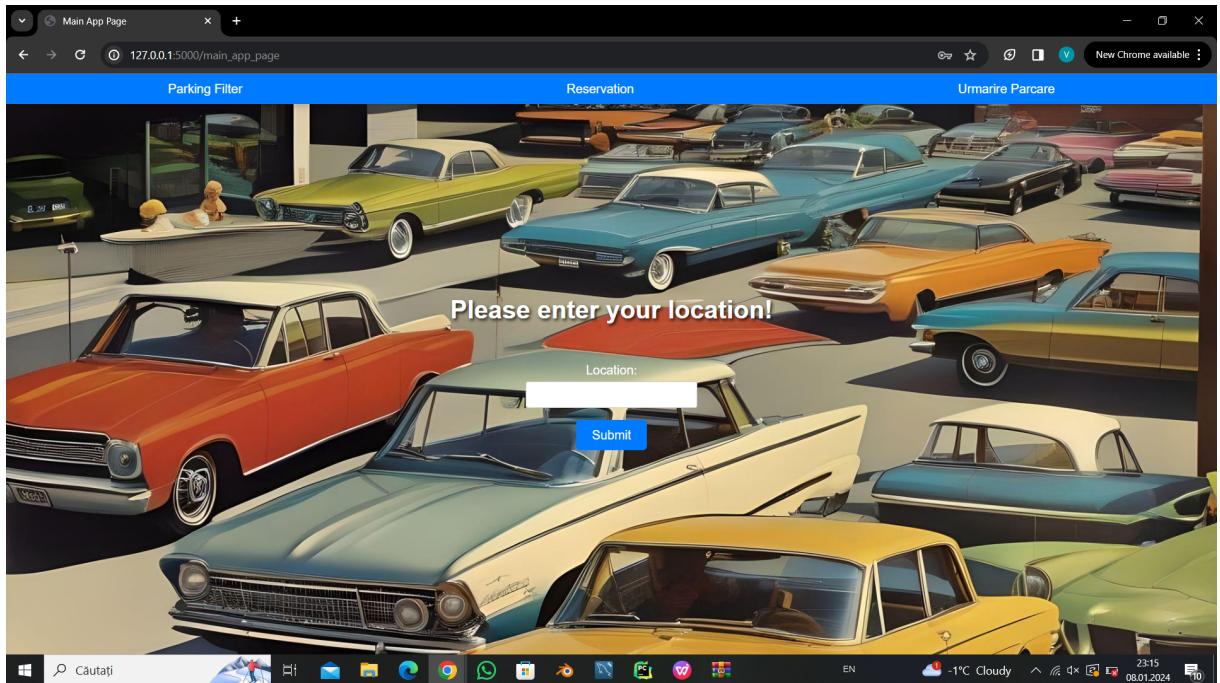
Dupa ce se ruleaza programul, este afisata o adresa care ne va trimite inspre pagina oficiala a aplicatiei.

```
* Serving Flask app 'webpage1'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
Nu s-au putut obtine rezervarile pentru clientul specificat.
```

Pagina de logare arata asa:



Dupa ce ne logam, se deschide urmatoarea pagina:



De aici putem sa introducem locatia noastra actuala si sa selectam care este cea mai apropiata parcare, sau care este cea mai ieftina, in functie de locurile disponibile din acestea.