



UNIVERSITATEA DE VEST DIN TIMIȘOARA
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ
PROGRAMUL DE STUDII DE LICENȚĂ:
Informatică

LUCRARE DE LICENȚĂ

COORDONATOR:
Lect. Dr. Sancira Monica

ABSOLVENT:
Kocsis Claudiu-Andras

TIMIȘOARA
2024

UNIVERSITATEA DE VEST DIN TIMIȘOARA
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ
PROGRAMUL DE STUDII DE LICENȚĂ:
Informatică

Sistem de recomandare pentru turism

COORDONATOR:
Lect. Dr. Sancira Monica

ABSOLVENT:
Kocsis Claudiu-Andras

TIMIȘOARA
2024

Abstract

This paper looks at the importance and impact of recommender systems in the tourism industry, with a focus on how they can enhance and personalize the traveler experience. In a world where travelers are increasingly seeking unique and personalized experiences, recommender systems are becoming an essential tool to meet these needs. Through data analytics and advanced algorithms, these systems can provide relevant recommendations for destinations, activities and tourism services tailored to travelers' individual preferences and interests.

The paper also examines the challenges and limitations associated with the implementation of these systems in the tourism industry, including issues related to the protection of personal data and user privacy. In addition, ways to manage loop filtering and ensure diversity of recommendations to avoid stagnation or repetition are discussed. Innovative approaches for the development and implementation of such systems, such as the integration of artificial intelligence and machine learning, are also explored to maximize the efficiency and relevance of the recommendations provided to users.

Through the research and implementation efforts described in the paper, a vision for the future of the tourism industry is emerging, in which the traveler's experience is enhanced through personalized and relevant recommendations. These systems can not only increase customer satisfaction and loyalty, but can also contribute to more sustainable tourism development.

Abstract

Această lucrare are în vedere importanța și impactul sistemelor de recomandare în industria turismului, cu accent pe modul în care acestea pot îmbunătăți și personaliza experiența călătorilor. Într-o lume în care călătorii caută tot mai mult experiențe unice și personalizate, sistemele de recomandare devin un instrument esențial pentru a satisface aceste nevoi. Prin analiza datelor și algoritmi avansați, aceste sisteme pot oferi recomandări relevante pentru destinații, activități și servicii turistice adaptate preferințelor și intereselor individuale ale călătorilor.

Lucrarea examinează, de asemenea, provocările și limitările asociate cu implementarea acestor sisteme în industria turismului, inclusiv aspecte legate de protecția datelor personale și confidențialitatea utilizatorilor. În plus, sunt discutate modalitățile de gestionare a filtrului buclat și de asigurare a diversității recomandărilor pentru a evita stagnarea sau repetiția. Abordările inovatoare pentru dezvoltarea și implementarea acestor sisteme, cum ar fi integrarea inteligenței artificiale și a învățării automate, sunt, de asemenea, explorate pentru a maximiza eficiența și relevanța recomandărilor oferite utilizatorilor.

Prin eforturile de cercetare și implementare descrise în lucrare, se conturează o viziune pentru viitorul industriei turismului, în care experiența călătorilor este îmbunătățită prin intermediul unor recomandări personalizate și relevante. Aceste sisteme nu numai că pot crește satisfacția și fidelitatea clienților, dar pot contribui și la dezvoltarea unui turism mai sustenabil.

Cuprins

1	Introducere	1
1.1	Generalități în domeniul sistemelor expert	1
1.2	Motivație	1
1.3	Obiective	2
1.4	Descrierea lucrării	2
2	Analiza stadiului actual în domeniul problemei	3
2.1	Generalități despre sistemele de recomandare	3
2.2	Definirea sistemelor de recomandare	3
2.3	Avantaje și dezavantaje ale sistemelor de recomandare	5
2.4	Clasificarea sistemelor de recomandare	5
2.4.1	Filtrarea colaborativă	5
2.4.2	Filtrarea bazată pe conținut	6
2.4.3	Filtrarea hibridă	7
2.5	Exemple reale de sisteme de recomandare	7
2.6	Sisteme de recomandare pentru turism	7
2.7	Principalele sisteme de recomandare pentru turism	8
2.7.1	Booking.com	8
2.7.2	TripAdvisor.com	9
2.7.3	Concluzii și comparații	10
3	Metodologia de proiectare și dezvoltare a aplicației	11
3.1	Obiectivele aplicației	11
3.2	Tehnologiile folosite	11
3.2.1	Django	11
3.2.2	Baza de date PostgreSQL	12
3.3	Beneficiile aduse în implementarea aplicației	12
3.4	Arhitectura sistemului	13
3.5	Interacțiunea componentelor	14
3.5.1	Aplicația web	14
3.5.2	Forkframe-ul Django	15
3.5.3	Baza de date PostgreSQL	15
3.5.4	Google Maps API	16
3.5.5	Fluxul de date	16
3.6	Diagrama de clase și interacțiunea dintre componente	17

3.7	Descrierea algoritmului de generare de recomandări	18
3.7.1	Tipul algoritmului de generare de recomandari	20
3.7.2	Abordarea filtrării colaborative	20
3.7.3	Interacțiunea cu baza de date	20
3.7.4	Criterii de recomandare	20
3.7.5	Detalii de implementare	21
4	Utilizarea aplicației, rezultate experimentale	22
4.1	Utilizarea aplicației	22
4.2	Rezultatele experimentale	28
4.2.1	Testarea automată	28
4.2.2	Testarea manuală	29
4.2.3	Testarea de către utilizatori reali	29
5	Concluzii, contribuții și direcții de continuarea a dezvoltării	33
5.1	Concluzii	33
5.2	Contribuții	33
5.3	Direcții de continuare a dezvoltării	34
	Bibliografie	35

Capitolul 1

Introducere

1.1 Generalități în domeniul sistemelor expert

Sistemele expert sunt programe informatice specializate în simularea procesului de luare a deciziilor umane într-un domeniu specific. Ele sunt concepute pentru a imita raționamentul uman și expertiza într-un anumit domeniu, utilizând cunoștințe și reguli predefinite pentru a rezolva probleme complexe.

Aceste sisteme sunt compuse dintr-o bază de cunoștințe care conține informații despre domeniul respectiv, și un motor de raționament care utilizează aceste informații pentru a extrage concluzii sau pentru a oferi recomandări în funcție de datele furnizate. Ele pot fi utilizate într-o varietate largă de domenii, cum ar fi medicina, ingineria, finanțele, managementul afacerilor și multe altele.

Sistemele expert sunt utile pentru a automatiza procesele de luare a deciziilor în situații în care este necesară o expertiză specializată sau în care luarea deciziilor se bazează pe reguli și criterii clare. Ele pot ajuta la accelerarea proceselor, la reducerea erorilor umane și la îmbunătățirea eficienței operaționale în diferite domenii.

1.2 Motivație

În 2019, turismul a reprezentat aproximativ 10.4 % din PIB-ul mondial, conform datelor Organizației Mondiale a Turismului (UNWTO). Acesta a avut o creștere constantă până în 2019, când apariția pandemiei COVID-19 a afectat drastic această industrie. În 2024 turismul a ajuns la 97% din impactul pe care îl avea în 2019 asupra PIB-ului mondial. În zilele noastre aplicațiile de recomandări au devenit principala sursă de marketing a orașelor, hotelurilor, restaurantelor etc. Un studiu realizat de compania McKinsey arată că digitalizarea turismului poate crește numărul de turiști cu până la 10-20% într-o anumită locație. Utilizatorii caută aplicații tot mai ușor de folosit și tot mai de încredere. Prin încredere, utilizatorii se referă la recomandări făcute pe baza recenziilor turiștilor care au trecut prin unele experiențe.

1.3 Obiective

Obiectivul principal al acestei lucrări este analiza sistemelor de recomandare pentru turism și realizarea unui astfel de sistem cât mai eficient și de încredere. Pe langa acesta, un alt obiectiv este satisfacerea experienței utilizatorului când vine vorba despre un sistem de recomandare pentru turism. Pentru aceasta am luat în considerare cel mai important aspect când vine vorba despre experiența utilizatorului, o aplicație ușor de folosit, intuitivă, care va aduce utilizatorului satisfacția de care are nevoie. Pentru ca procesul de recomandare să fie cât mai relevant pentru utilizator voi folosi o baza de date cât mai precisă și un algoritm de recomandare potrivit pentru problema discutată.

1.4 Descrierea lucrării

Această lucrare are în vedere importanța și impactul sistemelor de recomandare în industria turismului, cu accent pe modul în care acestea pot îmbunătăți și personaliza experiența călătorilor. Într-o lume în care călătorii caută tot mai mult experiențe unice și personalizate, sistemele de recomandare devin un instrument esențial pentru a satisface aceste nevoi. Prin analiza datelor și algoritmi avansați, aceste sisteme pot oferi recomandări relevante pentru destinații, activități și servicii turistice adaptate preferințelor și intereselor individuale ale călătorilor.

Lucrarea examinează, de asemenea, provocările și limitările asociate cu implementarea acestor sisteme în industria turismului, inclusiv aspecte legate de protecția datelor personale și confidențialitatea utilizatorilor. În plus, sunt discutate modalitățile de gestionare a filtrului buclat și de asigurare a diversității recomandărilor pentru a evita stagnarea sau repetiția. Abordările inovatoare pentru dezvoltarea și implementarea acestor sisteme, cum ar fi integrarea inteligenței artificiale și a învățării automate, sunt, de asemenea, explorate pentru a maximiza eficiența și relevanța recomandărilor oferite utilizatorilor.

Prin eforturile de cercetare și implementare descrise în lucrare, se conturează o viziune pentru viitorul industriei turismului, în care experiența călătorilor este îmbunătățită prin intermediul unor recomandări personalizate și relevante. Aceste sisteme nu numai că pot crește satisfacția și fidelitatea clienților, dar pot contribui și la dezvoltarea unui turism mai sustenabil.

Capitolul 2

Analiza stadiului actual în domeniul problemei

2.1 Generalități despre sistemele de recomandare

Un sistem de recomandare este o compoziție de instrumente software și tehnici de machine learning care oferă o decizie bazată pe anumiți factori când vine vorba de alegerea unor itemi sau servicii de către un utilizator. Sistemele de recomandare sunt utile în prezent atât în cercetare, cât și în domeniul comercial. Numeroase abordări au fost propuse pentru furnizarea de recomandări. Cu siguranță, sistemele de recomandare aduc utilizatorilor semne de întrebare când vine vorba despre acuratețea recomandării, siguranța datelor pe baza cărora se face recomandarea, aducerea unei satisfacții, a utilizatorului, cât mai ridicate în cadrul recomandării.

2.2 Definirea sistemelor de recomandare

”Un sistem de recomandare sau un motor de recomandare este un software de filtrare a conținutului care creează recomandări personalizate specifice utilizatorului pentru a-l ajuta în alegerile sale. Este utilizat pentru diferite produse, cum ar fi cărți, muzică, filme, videoclipuri, știri și social media.”[FR11]

Sistemele de recomandare sunt instrumente și tehnici software care oferă sugestii de itemi utili unui utilizator. Sugestiile se referă la diverse procese decizionale, cum ar fi ce articole să cumpere, ce muzică să asculte sau ce știri online să citească.

”Item” este termenul general utilizat pentru a desemna ceea ce sistemul recomandă utilizatorilor. În mod normal, un sistem de recomandare se concentrează pe un anumit tip de articol și, prin urmare, designul său, interfața grafică cu utilizatorul și tehnica principală de recomandare utilizată pentru a genera recomandările sunt toate personalizate pentru a oferi sugestii utile și eficiente pentru acel tip specific de articol.

Sistemele de recomandare se adresează în primul rând persoanelor care nu au suficientă experiență sau competență personală pentru a evalua numărul potențial copleșitor de elemente alternative pe care le poate oferi un site web, de exemplu. Un exemplu în acest sens este un sistem de recomandare a cărților care ajută utilizatorii să selecteze o carte de citit. În popularul site web, Amazon.com, site-ul utilizează un

sistem de recomandare pentru a personaliza magazinul online pentru fiecare client . Deoarece recomandările sunt de obicei personalizate, diferiți utilizatori sau grupuri de utilizatori primesc sugestii diverse.

În forma lor cea mai simplă, recomandările personalizate sunt oferite ca liste clasate de articole. În realizarea acestui clasament, sistemele de recomandare încearcă să prezică care sunt cele mai potrivite produse sau servicii, pe baza preferințelor și constrângerilor utilizatorului. Pentru a îndeplini o astfel de sarcină de calcul, sistemele de recomandare colectează de la utilizatori preferințele acestora, care sunt fie exprimate explicit, de exemplu, sub formă de evaluări pentru produse, fie deduse prin interpretarea acțiunilor utilizatorului. De exemplu, un sistem de recomandare poate considera navigarea către o anumită pagină de produse drept un semn implicit de preferință pentru articolele prezentate pe pagina respectivă.

Sistemele de recomandare joacă un rol important în multe industrii, așa că multe companii au decis să folosească acest tip de sisteme care s-au dovedit ca fiind eficiente și au adus beneficii precum:

- Creșterea vânzarilor. Acest obiectiv este atins deoarece articolele recomandate sunt de natură să satisfacă nevoile și dorințele utilizatorului.
- Vânzarea de articole diversificate. O altă funcție majoră a unui sistem de recomandare este de a permite utilizatorului să selecteze articole care ar putea fi greu de găsit fără o recomandare precisă.
- Creșterea satisfacției utilizatorului. Un sistem de recomandare bine conceput poate îmbunătăți, de asemenea, experiența utilizatorului cu site-ul sau aplicația. Utilizatorul va găsi recomandările interesante, relevante și, cu o interacțiune om-calculator bine concepută.

În Figura 2.1 este prezentată structura generală a unui sistem de recomandare. Mai sus putem observa cum Userul trimite către sistemul de recomandare diverși parametri/filtre: ratinguri, cuvinte cheie, restricții de atribute: preț, număr persoane etc. Apoi sistemul analizează datele primite și le compară cu cele pe care le-a colectat de la alți utilizatori ca mai apoi să trimită Userului un set de obiecte recomandate. Aceste obiecte fiind reprezentate de diferite locatii, restaurante, hoteluri.

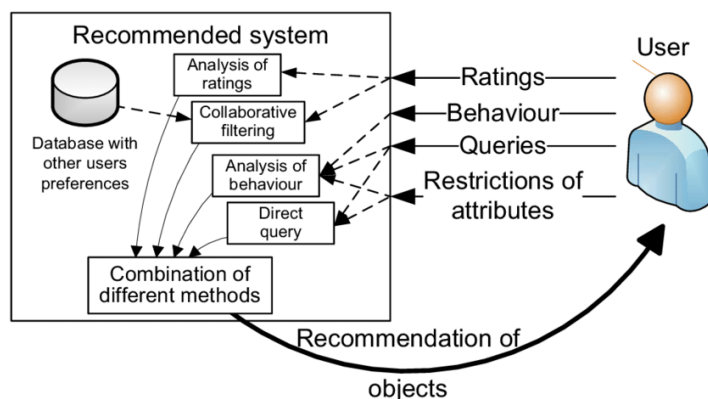


Figura 2.1: Structura unui sistem de recomandare.[Rec]

2.3 Avantaje și dezavantaje ale sistemelor de recomandare

Cu siguranță sistemele de recomandare aduc atât avantaje cât și dezavantaje.

■ Avantaje:

- Accelerează procesul luării unor decizii. Sistemele de recomandare iau decizii filtrând rapid toate atributele itemilor din baza de date.
- Aduc utilizatorului recomandări relevante pentru acesta. Acești algoritmi fac decizii pe baza acțiunilor utilizatorului care la rândul lui ia decizia finală.
- Sistemele de recomandare pot furniza informații valoroase despre preferințele și comportamentul utilizatorilor, care pot fi folosite pentru a îmbunătăți produsele sau serviciile oferite.

■ Dezavantaje:

- Prelucrarea unor date incomplete sau inexacte date poate duce la o recomandare neadecvată.
- Dacă algoritmul din spatele sistemului nu este adecvat acesta poate crea o buclă de filtrare, prezentând utilizatorilor doar produse similare cu cele vizualizate anterior limitând diversitatea și explorarea.
- Inexistența unui loc care are date complete pentru orice domeniu.
- Schimbarea continuă a datelor. Există riscul ca sistemele să fie manipulate pentru a recomanda unele iteme pe baza acțiunilor unor utilizatori, ceea ce poate afecta experiența altor utilizatori și încrederea în acel sistem.

Pentru a dezvolta și a utiliza sisteme de recomandare care să ofere beneficii utilizatorului, este important să echilibrăm aceste avantaje și dezavantaje.

2.4 Clasificarea sistemelor de recomandare

Sistemele de recomandare sunt clasificate după algoritmul din spatele acestora. La baza acestor sisteme stau 3 algoritmi: Filtrarea colaborativă, Filtrarea bazată pe conținut și Filtrarea hibridă.

2.4.1 Filtrarea colaborativă

Ideea principală a abordărilor de recomandare colaborativă este de a exploata informațiile privind comportamentul trecut sau opiniile unei comunități de utilizatori existente pentru a prezice care sunt elementele pe care utilizatorul actual al sistemului le va dori sau de care va fi interesat cel mai probabil. Aceste tipuri de sisteme sunt utilizate pe scară largă în prezent în industrie, în special ca instrument în site-urile de vânzări online pentru a personaliza conținutul în funcție de nevoile unui anumit client și pentru

a promova astfel articole suplimentare și a crește vânzările.

”Din perspectiva cercetării, aceste tipuri de sisteme au fost explorate de mulți ani, iar avantajele, performanțele și limitările lor sunt în prezent bine cunoscute. De-a lungul anilor, au fost propuși diverși algoritmi și tehnici, care au fost evaluați cu succes pe date de testare reale și artificiale.” [DJ11]

”Abordările colaborative pure iau ca unică intrare o matrice de ratinguri utilizator-element și produc, de obicei, următoarele tipuri de rezultate: (a) o predicție (numerică) care indică în ce măsură utilizatorului actual îi va plăcea sau nu un anumit element și (b) o listă de n elemente recomandate. O astfel de listă de top N nu ar trebui, desigur, să conțină articole pe care utilizatorul curent le-a cumpărat deja.” [DJ11]

2.4.2 Filtrarea bazată pe conținut

”În sistemele de recomandare bazate pe conținut, atributele descriptive ale elementelor sunt utilizate pentru a face recomandări. Termenul ”conținut” se referă la aceste descrieri. În metodele bazate pe conținut, evaluările și comportamentul de cumpărare al utilizatorilor sunt combinate cu informațiile de conținut disponibile în articole. De exemplu, să luăm în considerare o situație în care John a evaluat foarte bine filmul ”Terminator”, dar nu avem acces la evaluările altor utilizatori. Prin urmare, metodele de filtrare colaborativă sunt excluse. Cu toate acestea, descrierea articolului ”Terminator” conține cuvinte-cheie de gen similare cu cele ale altor filme science fiction, precum ”Alien and Predator”. În astfel de cazuri, aceste filme pot fi recomandate lui John.” [Agg16]

În metodele bazate pe conținut, descrierile articolelor, care sunt etichetate cu ratinguri, sunt utilizate ca date de antrenament pentru a crea o problemă de clasificare sau de modelare a regresiei specifică utilizatorului. Pentru fiecare utilizator, documentele de instruire corespund descrierilor articolelor pe care acesta le-a cumpărat sau le-a evaluat.

Metodele bazate pe conținut prezintă unele avantaje în ceea ce privește formularea de recomandări pentru articole noi, atunci când nu sunt disponibile suficiente date de evaluare pentru articolul respectiv. Acest lucru se datorează faptului că este posibil ca alte articole cu atribute similare să fi fost evaluate de utilizatorul activ. Prin urmare, modelul supravegheat va putea să utilizeze aceste evaluări împreună cu atributele articolului pentru a face recomandări chiar și atunci când nu există un istoric al evaluărilor pentru acel articol. Pe lângă aceste avantaje acest model are și dezavantaje. Această filtrare poate duce la o recomandare evidentă pe baza cuvintelor cheie. Deci, produse ce nu au cuvinte cheie asemănătoare cu iteme cu care utilizatorul a interacționat nu vor fi recomandate niciodată. Un alt dezavantaj al acestei abordări poate fi observat atunci când vine vorba despre un utilizator nou, deoarece modelul din spatele algoritmului nu are destule informații despre preferințele acestuia.

2.4.3 Filtrarea hibridă

”O abordare hibridă combină sistemul de filtrare bazat pe conținut și sistemul de filtrare colaborativă. În acest sistem este posibilă și utilizarea independentă a recomandărilor celor două tehnici de filtrare. Un sistem hibrid de recomandare este o altă categorie de sisteme de recomandare care încearcă să depășească limitele altor abordări. Deoarece combină două sau mai multe tehnici de recomandare pentru a obține o optimizare mai bună. Una dintre cele mai populare abordări hibride este cea care adaptează combinația euristică de filtrare bazată pe conținut și filtrare colaborativă.” [MA15]

2.5 Exemple reale de sisteme de recomandare

- GroupLens a fost un sistem de recomandare de top, care a fost construit ca un prototip de cercetare pentru recomandarea știrilor Usenet. Sistemul a colectat evaluări de la cititorii Usenet și le-a folosit pentru a prezice dacă altor cititori le va plăcea sau nu un articol înainte de a-l citi.
- Amazon.com a fost, de asemenea, unul dintre pionierii sistemelor de recomandare, în special în mediul comercial. Recomandările din Amazon.com sunt furnizate pe baza evaluărilor furnizate explicit, a comportamentului de cumpărare și a comportamentului de navigare.
- Netflix oferă utilizatorilor posibilitatea de a evalua filmele și emisiuni. În plus, acțiunile utilizatorilor în ceea ce privește vizionarea diferitelor elemente sunt, de asemenea, stocate de Netflix. Aceste evaluări și acțiuni sunt apoi utilizate de Netflix pentru a face recomandări.
- Facebook face recomandări de prieteni în funcție de o rețea socială. O creștere a numărului de conexiuni sociale îmbunătățește experiența unui utilizator la o rețea socială.

2.6 Sisteme de recomandare pentru turism

”Aplicațiile de turism sunt unele dintre cele mai vechi și în același timp unele din cele mai prolifice aplicații de pe internet. Diferiți termeni precum: ”eTourism”, ”Tourism Informatics” și ”Tourism ICT” sunt folosiți pentru a descrie acest tip de sisteme. Scopul acestor sisteme este de a îmbunătăți experiența de turism înainte, în timpul și după o călătorie. Turismul este în esență o activitate socială. La planificarea unei excursii participă mai mulți factori: sugestii cu privire la destinație de la prieteni/apropiați, interacțiunea cu agenții de turism pentru planificarea și rezervarea unor planuri de călătorie și cazare, găsirea diferitor activități în acele călătorii.” [Sha10]

Tehnologia sistemelor de recomandare turistică (Tourism Recommender Systems - TRS) se dezvoltă de aproximativ un deceniu. Un TRS acceptă informații de la un potențial călător și oferă recomandări pentru posibile excursii care corespund informațiilor introduse de utilizator. Un TRS simplu inițiază un dialog cu utilizatorul, punându-i diverse întrebări și apoi furnizează una sau mai multe recomandări pentru itinerarii de călătorie care corespund preferințelor utilizatorului. Un TRS de

bază furnizează recomandări sub formă de informații textuale, astfel încât utilizatorul trebuie să parcurgă mai multe resurse de informații, cum ar fi broșuri de călătorie sau pagini web referitoare la destinațiile și activitățile sugerate, pentru a vizualiza mental călătoria completă.

2.7 Principalele sisteme de recomandare pentru turism

2.7.1 Booking.com

O abordare existentă la această problema este sistemul de recomandare folosit de cei de la Booking.com[Boo]. Aceștia vin în fața utilizatorilor cu un algoritm Hybrid bazat pe conținut cât și pe baza alegerilor altor utilizatori.

Booking.com utilizează diverse sisteme de recomandare pentru a ajuta utilizatorii să descopere și să rezerve proprietăți care ar putea să le placă. Aceste recomandări sunt prezentate pe platformă în funcție de mai mulți factori relevanți, cum ar fi destinațiile populare, locuințele apreciate de alți oaspeți și proprietățile pe care alți călători cu căutări similare le-au rezervat. De exemplu, pe pagina de destinație "Sejururi" se folosesc sisteme pentru a sugera destinații și proprietăți în funcție de preferințele utilizatorilor și comportamentul altor călători.

Sistemele de recomandare se bazează pe date introduse de utilizatori în formularul de căutare (destinație, date, număr de oaspeți), istoricul interacțiunilor cu platforma și performanța proprietăților (cum ar fi rata de clic și rezervările nete și brute). De asemenea, informațiile despre disponibilitatea și scorurile evaluărilor contribuie la modul în care proprietățile sunt prezentate utilizatorilor. Importanța fiecărui factor poate varia pentru a maximiza șansele ca utilizatorii să găsească și să rezerve proprietăți potrivite.

Rezultatele căutărilor funcționează și ele ca un sistem de recomandare, fiind ordonate implicit după criteriul "Favoritele noastre" (denumit "Popularitate" în aplicație). Aceasta înseamnă că pentru a apărea în topul paginii, o cazare trebuie să performeze bine în termeni de rată de clic, rezervări brute și nete. În plus, poziția unei cazări poate fi influențată de factori precum valoarea comisionului plătit către Booking.com, participarea în programele Genius sau Partener Preferat, și informațiile colectate din interacțiunile utilizatorilor cu platforma.

Utilizatorii au la dispoziție mai multe opțiuni de sortare a rezultatelor căutării, cum ar fi locuințe și apartamente întâi, preț crescător, discount Genius întâi, și scor proprietate descrescător sau crescător. Aceste opțiuni permit utilizatorilor să personalizeze modul în care sunt afișate cazările, însă factorii din "Favoritele noastre" pot influența rezultatele chiar și atunci când sunt aplicate alte opțiuni de sortare.

Booking.com oferă și recomandări personalizate bazate pe modul în care utilizatorii interacționează cu platforma, precum și pe informațiile pe care le furnizează. Utilizatorii din Spațiul Economic European (SEE) au opțiunea de a gestiona și modifica setările pentru recomandările personalizate, fie pe website, fie în aplicație. Chiar și

atunci când recomandările personalizate sunt dezactivate, anumite informații colectate pot fi utilizate pentru a oferi o experiență mai plăcută pe platformă.

Preferințele referitoare la recomandările personalizate se aplică pe orice dispozitiv pe care utilizatorul este autentificat în contul Booking.com. Dacă utilizatorul nu este autentificat, aceste preferințe vor fi salvate în cookie-uri, care vor expira odată cu cookie-ul respectiv. Acest sistem asigură că experiența de utilizare rămâne consecventă și adaptată nevoilor și preferințelor fiecărui utilizator, contribuind astfel la o experiență optimă de căutare și rezervare a cazărilor.

2.7.2 TripAdvisor.com

Cu o altă abordare vine pe piață website-ul TripAdvisor[Tri].Sistemul acestora de recomandare, cunoscut sub numele de "Popularity Ranking", este un sistem de recomandare bazat pe filtrarea colaborativă și se bazează pe calitatea, recența și cantitatea recenziilor primite de la utilizatori, precum și pe consistența acestor recenzii în timp. Acest sistem este conceput pentru a reflecta cât mai precis experiența actuală oferită de o afacere, oferind utilizatorilor o imagine de ansamblu bazată pe feedback-ul colectiv al altor călători.

Calitatea recenziilor este determinată de evaluările sub formă de "bule" pe care utilizatorii le acordă în recenziile lor. În general, o afacere cu un număr mai mare de evaluări de cinci bule va fi clasată mai sus decât una cu evaluări mai puțin favorabile, toate celelalte aspecte fiind egale. Astfel, evaluările pozitive și de înaltă calitate joacă un rol crucial în poziționarea unei afaceri în clasament.

Recența recenziilor este un alt factor esențial, deoarece recenziile recente sunt considerate mai relevante pentru a reflecta experiența curentă la o afacere. Recenziile mai vechi, deși vizibile și utile pentru a oferi o perspectivă istorică, au o pondere mai mică în algoritmul de clasare. Această abordare asigură că ranking-ul rămâne relevant și actualizat în funcție de cele mai recente feedback-uri ale utilizatorilor.

Cantitatea recenziilor contribuie la încrederea utilizatorilor în evaluările afacerilor. Un număr mare de recenzii ajută utilizatorii să formeze o opinie echilibrată și să ia decizii mai informate. Totuși, pentru a fi clasată mai sus, o afacere nu trebuie doar să aibă multe recenzii, ci să atingă un număr suficient pentru a asigura semnificația statistică și comparabilitatea cu alte afaceri. Astfel, nu doar numărul total contează, ci și relevanța și recența acestor recenzii.

Algoritmul de Popularity Ranking integrează acești factori pentru a oferi o măsură statistică a încrederii în experiența curentă la o afacere. Pe măsură ce se acumulează mai multe recenzii, algoritmul devine mai precis în predicțiile sale. Consistența recenziilor, adică menținerea unui nivel constant de evaluări pozitive în timp, este esențială pentru o clasare superioară.

TripAdvisor continuă să optimizeze și să îmbunătățească acest algoritm pentru a reflecta cât mai exact performanța afacerilor în timp. Modificările recente au vizat promovarea unei mai mari consistențe în clasamente, astfel încât acestea să reflecte mai corect performanța relativă a afacerilor dintr-o locație.

2.7.3 Concluzii și comparații

Cele două sisteme de recomandare de la Booking.com și TripAdvisor folosesc algoritmi avansați pentru a oferi utilizatorilor sugestii relevante, însă se bazează pe factori diferiți și au abordări distincte în ceea ce privește clasarea proprietăților și a afacerilor. În Tabela 2.1 sunt comparate aceste aspecte.

Comparare:

Aspect	Booking.com	TripAdvisor
Factori de Recomandare	Utilizează date demografice și de comportament ale utilizatorilor, alături de performanța proprietăților	Se bazează pe calitatea, recența și cantitatea recenziilor
Temporalitatea	Consideră recența importantă	Pune un accent mai clar pe recenziile recente pentru a asigura relevanța experienței curente
Interacțiunea cu Utilizatorii	Oferă recomandări personalizate și permite utilizatorilor să sorteze rezultatele în funcție de preferințele lor specifice	Folosește recenziile pentru a forma un clasament care reflectă opinia colectivă a utilizatorilor
Performanța în Timp	Consistența este un factor comun: ambele platforme recompensează proprietățile și afacerile care mențin performanțe pozitive pe termen lung	
Metode de Colectare a Datelor	Se bazează pe interacțiuni directe și istoricul de căutare al utilizatorilor	Se bazează pe feedback-ul detaliat și frecvent oferit de utilizatori prin recenzii

Tabela 2.1: Compararea factorilor de recomandare între Booking.com și TripAdvisor

Deși ambele platforme folosesc algoritmi sofisticati pentru a ajuta utilizatorii să găsească opțiuni de cazare potrivite, Booking.com se axează mai mult pe comportamentul utilizatorilor și performanța proprietăților, în timp ce TripAdvisor se bazează pe analiza recenziilor pentru a oferi o evaluare echilibrată și actualizată a experiențelor la diverse afaceri.

Capitolul 3

Metodologia de proiectare și dezvoltare a aplicației

3.1 Obiectivele aplicației

Obiectivul principal al acestei aplicații este de a ajuta utilizatorii să găsească cele mai bune hoteluri adaptate preferințelor lor prin furnizarea de recomandări personalizate. Sistemul analizează recenziile și evaluările utilizatorilor pentru a le sugera hoteluri care corespund gusturilor și nevoilor lor. Luând în considerare factori precum ratingul general, numărul de recenzii și specificul localității și clasei de hoteluri, aplicația se asigură că utilizatorilor le sunt prezentate opțiuni care sunt atât bine cotate, cât și relevante pentru interesele lor. Astfel, utilizatorilor le este mai ușor să descopere noi hoteluri care ar putea să le placă, îmbunătățindu-le experiența de planificare a călătoriei.

Un alt obiectiv important al aplicației este acela de a oferi o platformă completă de căutare și evaluare a hotelurilor. Utilizatorii pot căuta hoteluri folosind diverse filtre, pot vizualiza informații detaliate despre fiecare hotel și pot citi recenzii de la alți călători. Integrarea cu Google Maps permite utilizatorilor să vadă locația exactă a hotelurilor, facilitându-le planificarea călătoriilor. Pentru utilizatorii care au analizat deja anumite hoteluri, sistemul se asigură că aceste hoteluri nu sunt recomandate din nou, concentrându-se pe opțiuni noi pe care utilizatorul nu le-a explorat încă. Această combinație de recomandări personalizate și informații detaliate despre hoteluri ajută utilizatorii să ia decizii în cunoștință de cauză și să găsească cele mai bune locuri de cazare pentru călătoriile lor.

3.2 Tehnologiile folosite

3.2.1 Django

”Django este un framework web Python sofisticat, de nivel înalt, care prioritizează dezvoltarea rapidă și principiile de proiectare pragmatice, curate. Creat de dezvoltatori experimentați, din cadrul ”Django Software Foundation” [Dja], acesta urmărește să eficientizeze procesul de dezvoltare web prin gestionarea multora dintre sarcinile repetitive implicate, permițând dezvoltatorilor să se concentreze asupra creării aplicațiilor lor.” Fiind open source, Django este disponibil gratuit pentru oricine, promovând uti-

lizarea pe scară largă și îmbunătățirile aduse de comunitate. Accentul pus pe eficiență permite dezvoltatorilor să dea viață rapid conceptelor lor, ceea ce îl face o alegere excelentă pentru proiectele care necesită termene de execuție rapide, fără a sacrifica calitatea.

Dincolo de viteza sa, Django este renumit pentru caracteristicile sale robuste de securitate, care îi ajută pe dezvoltatori să evite multe capcane comune de securitate. Cadrul include protecții integrate împotriva vulnerabilităților, cum ar fi injectia SQL, cross-site scripting și cross-site request forgery, oferind liniște atât dezvoltatorilor, cât și utilizatorilor finali. În plus, scalabilitatea Django îl face potrivit pentru a face față cerințelor site-urilor cu trafic ridicat. Multe dintre cele mai aglomerate site-uri de pe internet utilizează capacitatea lui Django de a se extinde rapid și flexibil, demonstrând capacitatea sa de a susține aplicații web dinamice la scară largă.

3.2.2 Baza de date PostgreSQL

”PostgreSQL este un sistem de baze de date puternic, open-source, de tip relațional, renumit pentru arhitectura sa robustă, fiabilitate și un set extins de caracteristici. Originar din proiectul POSTGRES de la Universitatea din California, Berkeley în 1986, PostgreSQL are peste 35 de ani de dezvoltare continuă. Această istorie a inovației și dedicarea comunității sale open-source i-au consolidat reputația de alegere de top pentru gestionarea volumelor complexe de date. PostgreSQL suportă toate sistemele de operare majore și este conform ACID din 2001, asigurând integritatea datelor.” [Pos]

Atât dezvoltatorii, cât și administratorii preferă PostgreSQL pentru extensibilitatea și caracteristicile sale cuprinzătoare care răspund diverselor nevoi de gestionare a datelor. Acesta acceptă o gamă largă de tipuri și structuri de date, de la tipuri de date primitive precum numerele întregi și șirurile de caractere până la tipuri de date personalizate complexe și documente JSON. Respectarea standardelor SQL de către PostgreSQL, combinată cu indexarea avansată, optimizarea sofisticată a interogărilor și măsurile robuste de securitate, fac din acesta o alegere ideală pentru aplicațiile care necesită performanță și securitate ridicate. Capacitatea sa de scalare eficientă îi permite să gestioneze seturi masive de date și un număr mare de utilizatori simultani, dovedindu-și scalabilitatea și fiabilitatea în medii de producție solicitante.

3.3 Beneficiile aduse în implementarea aplicației

Utilizând împreună Django și PostgreSQL în dezvoltarea acestui sistem de recomandări aduce beneficii specifice care sporesc eficiența, flexibilitatea și scalabilitatea generală a aplicației. ORM (Object-Relational Mapping) din Django se integrează perfect cu PostgreSQL, permițând dezvoltatorilor să interacționeze cu baza de date utilizând obiecte și metode Python în loc să scrie interogări SQL brute. Acest strat de abstractizare simplifică operațiunile cu baza de date și asigură că baza de cod rămâne curată și ușor de întreținut. În plus, caracteristicile integrate ale Django, cum ar fi o interfață de administrare pentru gestionarea înregistrărilor din baza de date și gestionarea formularelor pentru intrările utilizatorilor, accelerează procesul de dezvoltare. Caracteristicile de securitate ale cadrului, inclusiv protecția împotriva SQL injection, XSS (cross-site scripting) și CSRF (cross-site request forgery), asigură că aplicația

rămâne protejată împotriva vulnerabilităților web comune.

PostgreSQL, pe de altă parte, oferă capacități avansate de gestionare a datelor care sunt esențiale pentru un sistem de recomandări. Suportul său pentru interogări complexe, căutare full-text și indexare avansată permite recuperarea și prelucrarea eficientă a datelor, chiar și în cazul seturilor mari de date. Suportul PostgreSQL pentru tipurile de date JSONB permite stocarea și interogarea datelor semi-structurate, ceea ce este benefic pentru gestionarea diverselor recenzii și evaluări ale utilizatorilor. Controlul robust al simultaneității și gestionarea tranzacțiilor din baza de date asigură coerența și integritatea datelor, chiar și în scenarii cu trafic intens. Prin valorificarea caracteristicilor puternice ale PostgreSQL, sistemul de recomandări poate efectua rapid interogări analitice complexe, oferind utilizatorilor recomandări oportune și relevante. Împreună, capacitățile de dezvoltare rapidă ale Django și gestionarea avansată a bazei de date PostgreSQL creează un sistem de recomandări extrem de performant, scalabil și sigur, care răspunde atât nevoilor dezvoltatorilor, cât și ale utilizatorilor finali.

3.4 Arhitectura sistemului

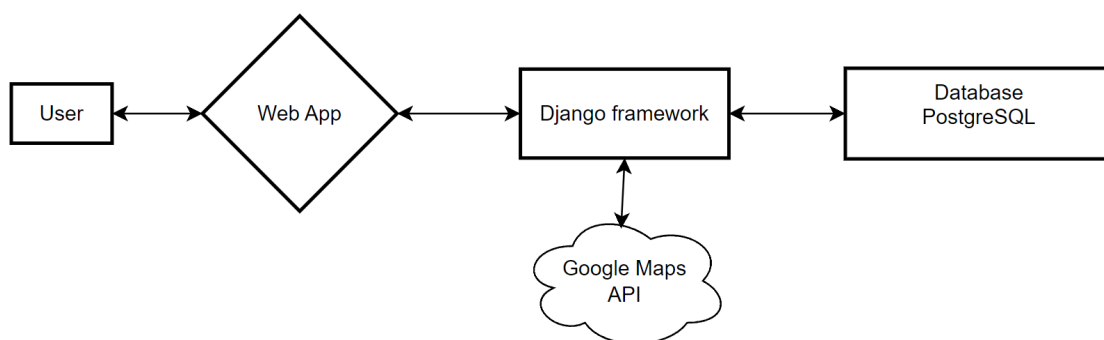


Figura 3.1: Arhitectura aplicației.

În Figura 3.1 este reprezentată arhitectura sistemului care ilustrează designul aplicației noastre web de recomandări hoteliere. Sistemul utilizează Django ca framework principal, folosind arhitectura sa robustă pentru a gestiona URL Routing, Views, Models și Templates. Utilizatorii interacționează cu aplicația web, care servește drept interfață pentru trimiterea recenziilor, vizualizarea informațiilor despre hoteluri și primirea de recomandări personalizate privind hotelurile. De asemenea, aplicația se integrează cu API-ul Google Maps pentru a furniza date geografice și a afișa vizual locațiile hotelurilor.

Backend-ul aplicației este alimentat de PostgreSQL, un sistem de baze de date fiabil și scalabil care stochează informații esențiale despre hoteluri, recenzii ale utilizatorilor și evaluări generale. Această configurație permite recuperarea eficientă a datelor și interogările complexe necesare pentru generarea de recomandări personalizate. Analizând recenziile utilizatorilor și valorificând tehnicile de filtrare colaborativă, sis-

temul recomandă hoteluri care se aliniază cu preferințele utilizatorului. În plus, arhitectura aplicației asigură interacțiuni fluide între interfața cu utilizatorul, cadrul Django, baza de date și API-urile externe, rezultând o experiență a utilizatorului fără întreruperi și receptivă.

3.5 Interacțiunea componentelor

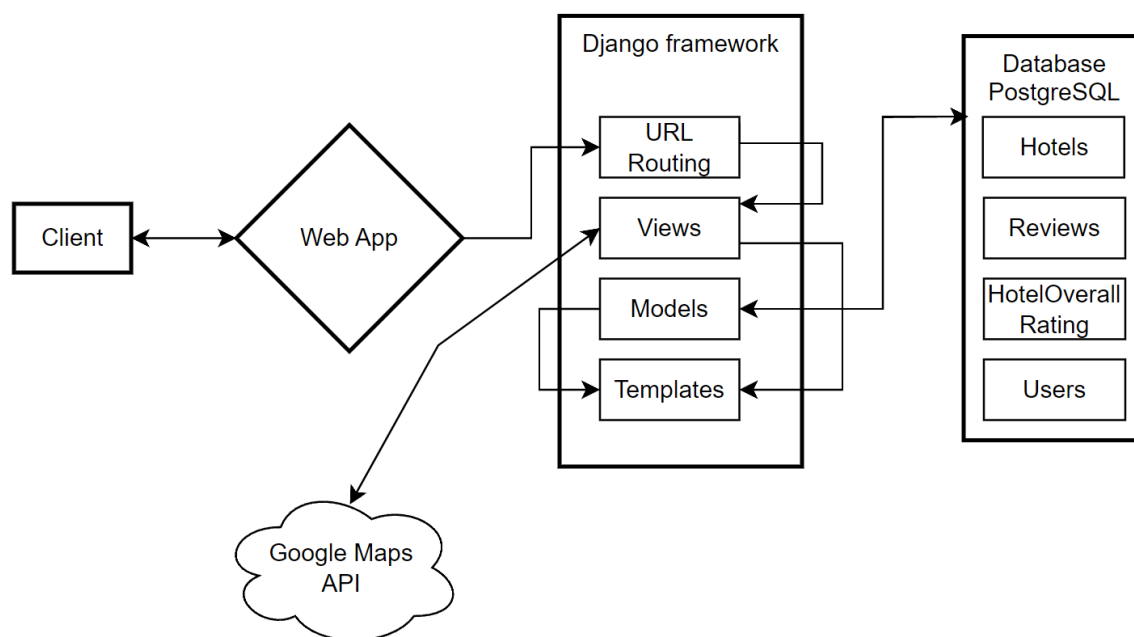


Figura 3.2: Interacțiunea dintre componentele aplicației.

Arhitectura sistemului de recomandare hotelieră este alcătuită din mai multe componente care interacționează și care lucrează împreună pentru a oferi o experiență de utilizare fără întreruperi, cum este ilustrat în Figura 3.2. În centrul acestei interacțiuni se află clientul, care accesează aplicația web prin intermediul unui browser web sau al oricărei alte interfețe client. Solicitățile clientului inițiază o serie de procese în cadrul aplicației web, care acționează ca intermediar între utilizator și componentele backend ale sistemului.

3.5.1 Aplicația web

Aplicația web servește drept interfață principală pentru interacțiunile cu utilizatorii. Ea este responsabilă de gestionarea datelor introduse de utilizator, cum ar fi interogările de căutare, criteriile de filtrare și trimerile de recenzii. Atunci când un utilizator interacționează cu aplicația web, datele de intrare sunt captate și transmise către cadrul Django pentru prelucrare ulterioară. Aplicația web se asigură că interfața cu utilizatorul este intuitivă și receptivă, oferind feedback în timp real pe baza acțiunilor utilizatorului.

3.5.2 Forkframe-ul Django

Forkframe-ul Django este coloana vertebrală a sistemului, orchestrând fluxul de date și logică între diferite componente. Acesta cuprinde patru părți principale: URL Routs, Views, Models și Templates.

- **URL Routing:** Această componentă mapează cererile utilizatorilor către vizualizările corespunzătoare pe baza modelelor URL definite în aplicație. Atunci când un utilizator trimite o cerere, cum ar fi accesarea detaliilor unui hotel sau trimiterea unei recenzii, componenta de rutare URL direcționează cererea către funcția de vizualizare corespunzătoare.
- **Views:** Componenta views conține logica de afaceri a aplicației. Aceasta procesează datele primite de la componenta de rutare URL și interacționează cu modelele pentru a prelua sau actualiza informațiile din baza de date. De exemplu, atunci când un utilizator caută hoteluri, vizualizarea preia datele relevante privind hotelurile din baza de date și le pregătește pentru afișare.
- **Models:** Modelele reprezintă structura de date a aplicației. Ele definesc schema tabelelor din baza de date și oferă un strat de abstractizare pentru interacțiunile cu baza de date. În această aplicație, modelele sunt utilizate pentru a defini entități precum hoteluri, recenzii, utilizatori și evaluări generale. Atunci când componenta views trebuie să acceseze sau să modifice date, aceasta interacționează cu modelele, care la rândul lor comunică cu baza de date PostgreSQL.
- **Templates:** Șabloanele sunt utilizate pentru redarea datelor în HTML pentru afișarea în browserul utilizatorului. Ele sunt responsabile Presentation Layer-ul aplicației, asigurându-se că datele preluate de vizualizări sunt formatate și stilizate corespunzător. Șabloanele oferă o modalitate dinamică de afișare a conținutului, permițând actualizări în timp real bazate pe interacțiunile utilizatorului.

3.5.3 Baza de date PostgreSQL

Baza de date PostgreSQL este componenta de stocare a datelor din sistem. Aceasta stochează toate informațiile referitoare la hoteluri, recenzii, evaluări generale și utilizatori. Baza de date este concepută pentru a gestiona interogări complexe și pentru a sprijini operațiunile cu utilizare intensivă de date ale aplicației. Atunci când modelele interacționează cu baza de date, acestea utilizează interogări SQL pentru a prelua, actualiza sau șterge înregistrări, după cum este necesar. De exemplu, atunci când un utilizator trimite o recenzie, vizualizarea actualizează ratingurile hotelului corespunzător în baza de date prin intermediul modelului.

Pentru popularea bazei de date am folosit un set de date de pe Kaggle.com. Acest set de date a fost colectat de utilizatorul JOAKIM ARVIDSSON de pe TripAdvisor.com, și conține 2 tabele: offerings.csv care conține date despre 4333 hoteluri și reviews.csv care conține 878561 review-uri ale hotelurilor. Cu ajutorul acestora am creat 3 tabele în baza de date: tabela hotels care conține datele din offerings.csv, tabela review care conține datele din reviews.csv și tabela hoteloverallrating în care am calculat media notelor review-urilor.

3.5.4 Google Maps API

API-ul Google Maps este integrat în sistem pentru a furniza date geografice și vizualizare. Atunci când un utilizator caută hoteluri, aplicația utilizează Google Maps API pentru a prelua detaliile locației și a afișa locațiile hotelurilor pe o hartă. Această integrare îmbunătățește experiența utilizatorului prin furnizarea unei reprezentări vizuale a locațiilor hotelurilor, ajutând utilizatorii să ia decizii informate pe baza proximității și a contextului geografic.

3.5.5 Fluxul de date

Fluxul de date în aplicație începe cu solicitarea clientului, care este direcționată prin aplicația web către cadrul Django. Componenta de rutare URL direcționează cererea către vizualizarea corespunzătoare, care procesează datele de intrare și interacționează cu modelele pentru a prelua sau actualiza datele din baza de date PostgreSQL. Rezultatele sunt apoi transmise modelelor, care redă datele în HTML pentru a fi afișate în browserul utilizatorului. Dacă sunt necesare date privind locația, componenta views interacționează, de asemenea, cu API-ul Google Maps pentru a prelua și afișa informații geografice.

Această arhitectură modulară și stratificată asigură că fiecare componentă îndeplinește un rol specific, făcând sistemul eficient, scalabil și ușor de întreținut. Separarea clară a preocupărilor permite dezvoltatorilor să se concentreze asupra componentelor individuale, asigurându-se că fiecare parte a sistemului poate fi dezvoltată, testată și optimizată independent.

3.6 Diagrama de clase și interacțiunea dintre componente

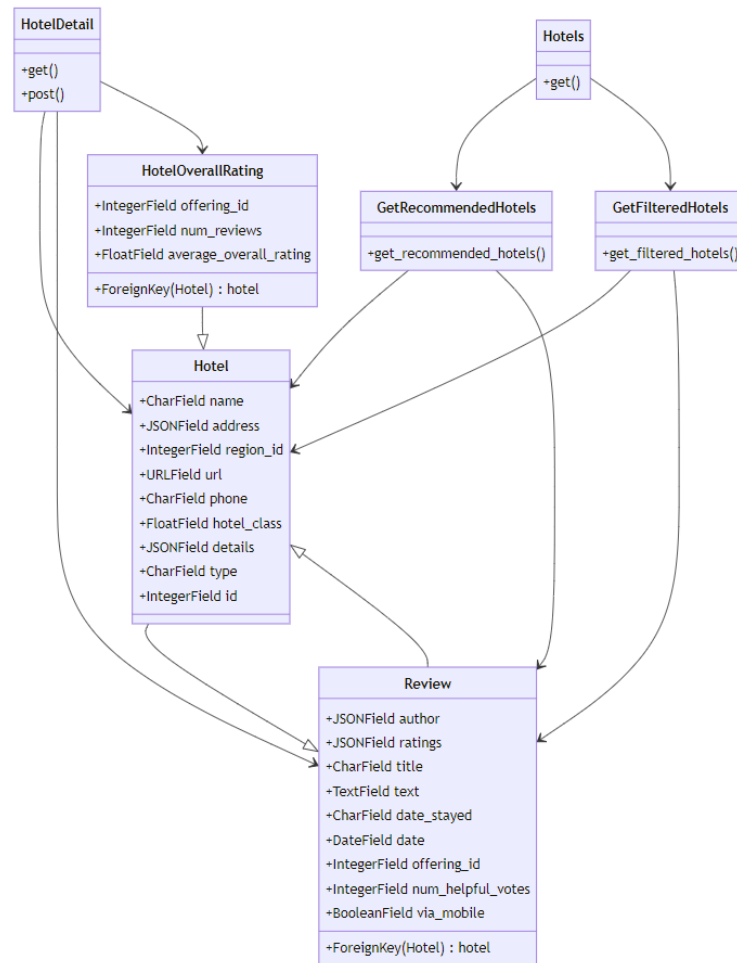


Figura 3.3: Diagrama de clase.

În diagrama de clase reprezentată în Figura 3.3, avem o reprezentare cuprinzătoare a interacțiunilor dintre diferitele componente ale aplicației noastre web Django. Modelul Hotel servește drept entitate centrală, conținând informații esențiale despre hoteluri, cum ar fi numele, adresa și clasa acestora. Fiecare hotel poate avea mai multe obiecte Review asociate, reflectând relația many-to-one, în care fiecare review conține detalii despre evaluările utilizatorilor, informații despre autor și data șederii.

Modelul HotelOverallRating agregă recenziile pentru a furniza o evaluare generală și numărul total de recenzii pentru fiecare hotel. Aceste informații sunt esențiale pentru calcularea și afișarea ratingului mediu al hotelului, la care fac referire diverse vizualizări și funcții de filtrare. Vizualizarea HotelDetail recuperează informații detaliate despre un anumit hotel, recenziile acestuia și ratingurile generale, permițând utilizatorilor să vadă informații complete despre un hotel. Aceasta interacționează direct cu modelele Hotel, Review și HotelOverallRating pentru a prezenta aceste date utilizatorului.

Vizualizarea Hoteluri este responsabilă pentru listarea hotelurilor și facilitarea proceselor de filtrare și recomandare. Aceasta apelează la funcțiile `GetRecommendedHotels` și `GetFilteredHotels` pentru a furniza utilizatorilor liste personalizate și filtrate de hoteluri. Aceste funcții execută interogări complexe pentru a identifica hotelurile care îndeplinesc criterii specifice, cum ar fi un număr minim de recenzii sau anumite praguri de rating. Interacțiunile dintre aceste componente garantează că utilizatorii primesc informații relevante și exacte despre hoteluri, îmbunătățindu-le experiența și ajutându-i să ia decizii în cunoștință de cauză.

3.7 Descrierea algoritmului de generare de recomandări

```

1 def get_recommended_hotels(user):
2     with connection.cursor() as cursor:
3         # Step 1: Find all hotels reviewed by the current user with
4         # at least 3.5 rating
5         cursor.execute("""
6             SELECT DISTINCT h.id
7             FROM hotelsapp_hotel h
8             JOIN hotelsapp_review r ON h.id = r.hotel_id
9             WHERE r.author->>'username' = %s
10             AND CAST(r.ratings->>'overall' AS NUMERIC) >= 3.5
11         """, [user.username])
12         reviewed_hotels = cursor.fetchall()
13     if reviewed_hotels:
14         reviewed_hotel_ids = [row[0] for row in reviewed_hotels]
15
16     with connection.cursor() as cursor:
17         # Step 2: Find all users who reviewed these hotels
18         cursor.execute("""
19             SELECT DISTINCT r.author->>'username'
20             FROM hotelsapp_review r
21             WHERE r.hotel_id IN %s
22             AND r.author->>'username' != %s
23         """, [tuple(reviewed_hotel_ids), user.username])
24         similar_users = cursor.fetchall()
25     if similar_users:
26         similar_usernames = [row[0] for row in similar_users]
27         # Step 3: Find all hotels reviewed by these users with at
28         # least 3.5 rating
29         with connection.cursor() as cursor:
30             cursor.execute("""
31                 SELECT h.id, h.name, h.address->>'locality' as
32                 locality, h.hotel_class,
33                 AVG(CAST(r.ratings->>'overall' AS NUMERIC)
34                 ) as avg_rating,
35                 COUNT(r.id) as review_count
36                 FROM hotelsapp_hotel h
37                 JOIN hotelsapp_review r ON h.id = r.hotel_id
38                 WHERE CAST(r.ratings->>'overall' AS NUMERIC) >=
39                 3.5
40                 AND r.author->>'username' IN %s
41                 AND h.id NOT IN (
42                     SELECT hotel_id

```



```

38         FROM hotelsapp_review
39         WHERE author->>'username' = %s
40     )
41     GROUP BY h.id
42     HAVING COUNT(r.id) >= 20
43     ORDER BY avg_rating DESC
44     LIMIT 10
45     """ , [tuple(similar_usernames), user.username])
46     rows = cursor.fetchall()
47     recommended_hotels = [
48         {'id': row[0], 'name': row[1], 'locality': row[2], '
hotel_class': row[3], 'avg_rating': row[4], 'review_count': row
[5]}
49         for row in rows
50     ]
51     else:
52         recommended_hotels = []
53     else:
54         recommended_hotels = []
55     if len(recommended_hotels) < 10:
56         additional_needed = 10 - len(recommended_hotels)
57         with connection.cursor() as cursor:
58             cursor.execute("""
59                 SELECT h.id, h.name, h.address->>'locality' as
locality, h.hotel_class,
60                     AVG(CAST(r.ratings->>'overall' AS NUMERIC)) as
avg_rating,
61                     COUNT(r.id) as review_count
62                 FROM hotelsapp_hotel h
63                 JOIN hotelsapp_review r ON h.id = r.hotel_id
64                 WHERE CAST(r.ratings->>'overall' AS NUMERIC) >= 3.5
65                 AND h.id NOT IN (
66                     SELECT hotel_id
67                     FROM hotelsapp_review
68                     WHERE author->>'username' = %s
69                 )
70                 GROUP BY h.id
71                 HAVING COUNT(r.id) >= 20
72                 ORDER BY avg_rating DESC
73                 LIMIT %s
74             """ , [user.username, additional_needed])
75         additional_rows = cursor.fetchall()
76         additional_hotels = [
77             {'id': row[0], 'name': row[1], 'locality': row[2], '
hotel_class': row[3], 'avg_rating': row[4], 'review_count': row
[5]}
78             for row in additional_rows
79         ]
80         recommended_hotels.extend(additional_hotels[:
additional_needed])
81
82     return recommended_hotels

```

Listing 3.1: Functia Get_recommended_hotels.

3.7.1 Tipul algoritmului de generare de recomandari

Funcția `get_recommended_hotels` 3.1 implementează o abordare de filtrare colaborativă pentru a genera recomandări hoteliere pentru un utilizator. Filtrarea colaborativă este o metodă utilizată de sistemele de recomandare pentru a identifica modele și preferințe din datele colectate de la mai mulți utilizatori. Această implementare specifică este o formă de filtrare colaborativă bazată pe utilizator.

3.7.2 Abordarea filtrării colaborative

Filtrare colaborativă bazată pe utilizator: Sistemul se bazează pe recenziile utilizatorilor similari pentru a oferi recomandări. Acesta caută hoteluri pe care alți utilizatori, care au preferințe similare (de exemplu, ratinguri ridicate pentru anumite hoteluri), le-au evaluat foarte bine. Acest lucru ajută la găsirea hotelurilor care ar putea plăcea și utilizatorului țintă, pe baza presupunerii că utilizatorii similari vor avea gusturi similare.

3.7.3 Interacțiunea cu baza de date

Funcția interacționează direct cu baza de date utilizând interogări SQL brute. Acest lucru asigură recuperarea eficientă a datelor, în special atunci când este vorba de seturi mari de date. Interogările SQL sunt optimizate pentru a extrage informații relevante despre hoteluri, pentru a filtra pe baza istoricului de recenzii al utilizatorului și pentru a se asigura că hotelurile recomandate îndeplinesc anumite criterii (de exemplu, au un rating mediu ridicat și un număr minim de recenzii).

3.7.4 Criterii de recomandare

- Prag de rating ridicat: Ia în considerare numai hotelurile cu un rating mediu de 3,5 sau mai mare.
- Număr minim de recenzii: Se asigură că fiecare hotel recomandat are cel puțin 20 de recenzii. Acest lucru ajută la recomandarea hotelurilor care au fost evaluate în mod constant de mulți utilizatori, reducând riscul de anomalii cauzate de puține recenzii.
- Excluderea hotelurilor deja recenzate: Hotelurile pe care utilizatorul le-a evaluat deja sunt excluse din recomandări pentru a evita redundanța și a îmbunătăți relevanța recomandărilor.
- Căutarea recomandărilor principale: Sistemul caută hoteluri foarte bine cotate de utilizatori similari, care au recenzat pozitiv, overall rating >3.5, un hotel pe care utilizatorul țintă, la rândul lui l-a recenzat pozitiv. Apoi se asigură că fiecare hotel are un număr semnificativ de recenzii și nu a fost recenzat de utilizator.
- Interogare secundară (de rezervă): În cazul în care interogarea primară nu oferă suficiente hoteluri, această interogare secundară extrage hoteluri suplimentare bazate exclusiv pe ratingurile medii ridicate și numărul de recenzii, excluzându-le pe cele deja recenzate de utilizator.

3.7.5 Detalii de implementare

Intrare: Funcția primește un obiect utilizator ca intrare.

Ieșire: Aceasta returnează o listă de dicționare, fiecare conținând detalii despre un hotel recomandat, cum ar fi ID-ul, numele, localitatea, clasa hotelului, evaluarea medie și numărul de recenzii.

Eficiență: Prin utilizarea interogărilor SQL brute și a interacțiunilor directe cu baza de date, funcția asigură recuperarea eficientă și rapidă a datelor, ceea ce este esențial pentru sistemele de recomandare în timp real.

În concluzie, funcția `get_recommended_hotels` este un exemplu robust de sistem de filtrare colaborativă bazat pe utilizator. Aceasta combină în mod eficient preferințele utilizatorilor și datele din recenzii pentru a oferi recomandări hoteliere personalizate, asigurând relevanța și calitatea prin diverse criterii și mecanisme de rezervă.

Capitolul 4

Utilizarea aplicației, rezultate experimentale

4.1 Utilizarea aplicației

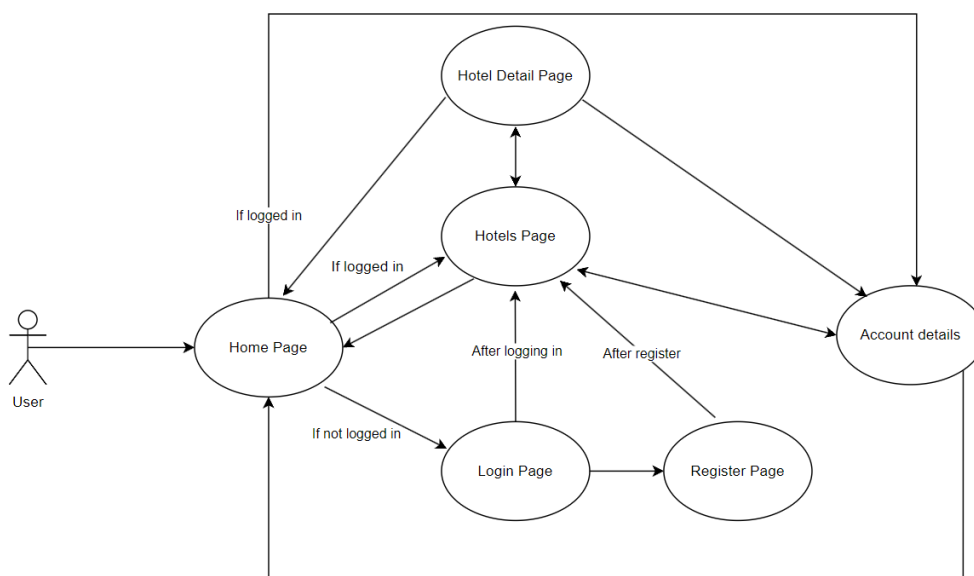


Figura 4.1: Diagrama pagini accesibile utilizatorului.

În Figura 4.1 este reprezentat flow-ul prin care utilizatorul poate accesa paginile aplicației. Din pagina Home utilizatorul, dacă nu este logat, poate ajunge doar în paginile de Login și Register, acesta fiind redirecționat către pagina de Login dacă acesta încearcă să acceseze pagina Hotels, iar acesta poate accesa paginile Hotels și Account Details dacă este logat. Din pagina Login, după ce utilizatorul s-a logat, acesta este redirecționat automat către pagina Hotels. Din pagina Register, după ce utilizatorul s-a înregistrat, acesta este redirecționat automat către pagina Hotels. Din pagina Hotels utilizatorul poate ajunge în toate paginile mai puțin cea de Login și cea de Register, deoarece utilizatorul trebuie să fie logat pentru a ajunge în pagina Hotels. Același lucru se aplica și pentru pagina Hotel Detail, care este o pagină ce deschide un url personaliat în funcție de Hotelul ales de utilizator și unde se poate ajunge doar

dacă utilizatorul a vizitat pagina Hotels. Din pagina Account details utilizatorul poate ajunge în paginile Hotels și Home.

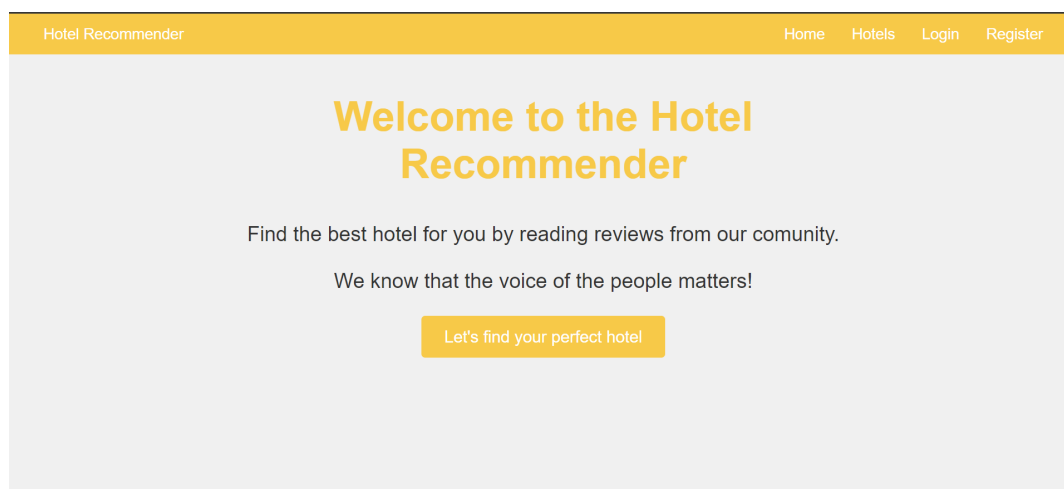


Figura 4.2: Pagina Home dacă utilizatorul nu este logat.

În Figura 4.2 este prezentată pagina Home dacă utilizatorul nu este logat. Utilizatorul este întâmpinat de un mesaj de bun venit, iar sub acesta este un buton ce te redirectioneaza catre pagina Hotels, dar deoarece utilizatorul nu e logat acesta va fi redirecționat către pagina Login.

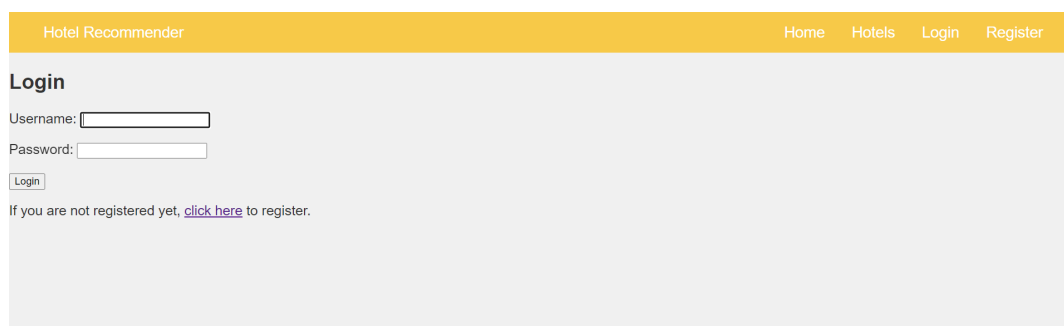


Figura 4.3: Pagina Login.

În Figura 4.3 este prezentată pagina Login care dispunde de un formular de logare. Dacă utilizatorul introduce datele corect acesta este redirecționat catre pagina Hotels. Dacă utilizatorul introduce datele incorecte acesta este avertizat că a introdus credențialele incorecte. Sub formularul de logare este un text și un buton ce te redirecționează catre pagina register.

The screenshot shows the 'Register' page of the 'Hotel Recommender' application. At the top, there is a yellow navigation bar with the text 'Hotel Recommender' on the left and links for 'Home', 'Hotels', 'Login', and 'Register' on the right. The main content area has a light gray background. The title 'Register' is in bold. Below it, there are two input fields: 'Username:' and 'Password:'. The 'Username' field has a placeholder text: 'Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.' Below the 'Password' field, there is a bulleted list of password requirements: 'Your password can't be too similar to your other personal information.', 'Your password must contain at least 8 characters.', 'Your password can't be a commonly used password.', and 'Your password can't be entirely numeric.' Below this list is a 'Password confirmation:' field with a placeholder text: 'Enter the same password as before, for verification.' At the bottom left of the form is a 'Register' button.

Figura 4.4: Pagina Register.

În Figura 4.4 este prezentată pagina Register care dispunde de un formular de înregistrare. Alături de acesta sunt explicate cerințele pentru a crea username-ul și parola: username-ul poate avea cel mult 150 caractere de tipul(litere/cifre/@/./+/-/_), parola nu poate fi similară cu username-ul, parola trebuie să conțină cel puțin 8 caractere, parola nu poate fi una uzual folosită și parola nu poate conține doar cifre. Pentru o siguranță sporită utilizatorul este rugat să reintroducă parola pentru confirmare. Dacă utilizatorul a introdus credențiale conform cerințelor acesta este logat automat și redirectionat către pagina Hotels.

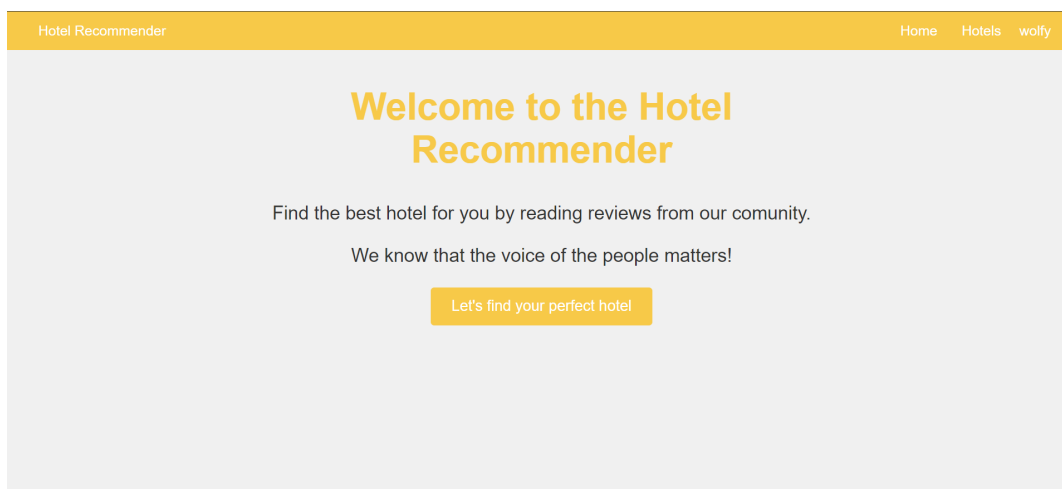


Figura 4.5: Pagina Home dacă utilizatorul este logat.

În Figura 4.5 este prezentată pagina Home dacă utilizatorul este logat. Aceasta pagină are în componență aceleași mesaje doar bara de navigare este modificată. Utilizatorul poate acum accesa pagina Hotels, iar în loc de butoanele ce duc către paginile Login și Register acesta are un buton cu numele lui de utilizator, care, dacă pune cursorul peste acesta vor apărea alte 2 butoane, unul care ne duce către pagina Account Details și unul care, odată apasat, îl deloghează pe utilizator.

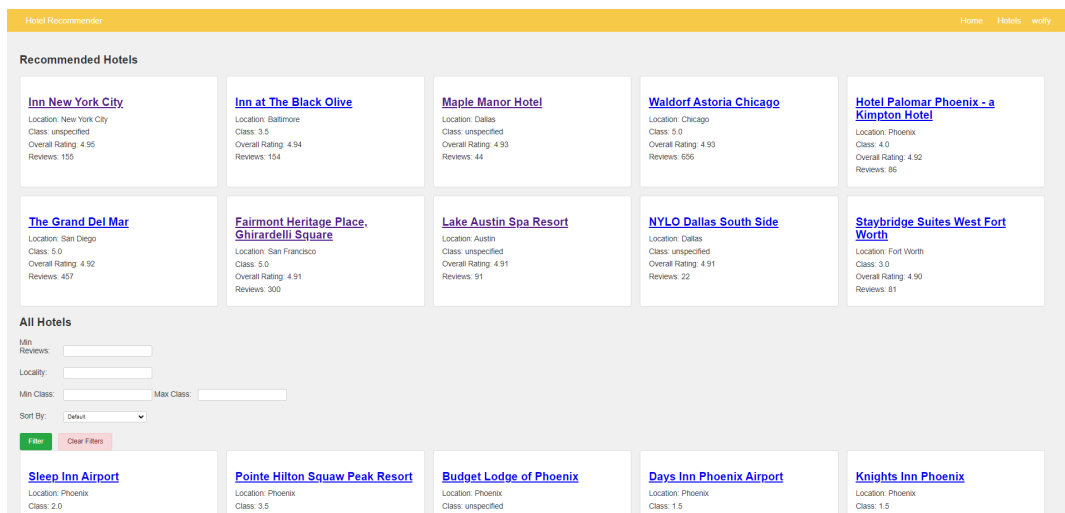


Figura 4.6: Partea de sus a paginii Hotels .

În Figura 4.6 este prezentată partea de sus a paginii Hotels. Aici putem observa bara de navigare, secțiunea de Hoteluri Recomandate unde sunt afișate 10 hoteluri recomandate de sistemul nostru. Sub aceasta, avem secțiunea unde se pot vedea toate hotelurile. Pentru o navigare mai ușoară am implementat un sistem de filtrare a hotelurilor în funcție de: numărul minim de recenzii, locația hotelului, clasa hotelului și cum poate fi vizualizată lista: ascendent sau descendent în funcție de evaluarea medie a hotelului.

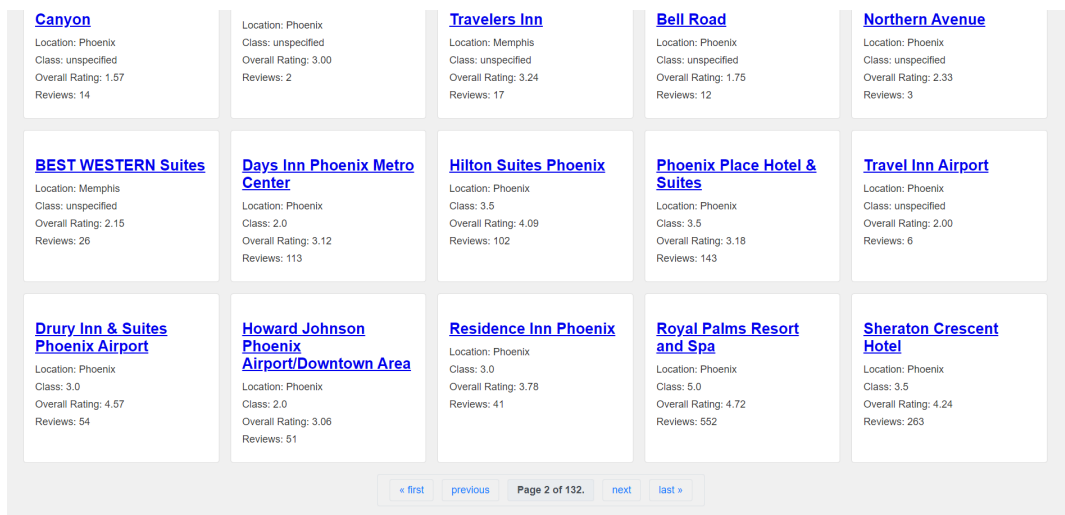


Figura 4.7: Partea de jos a paginii Hotels .

În Figura 4.7 este prezentată partea de jos a paginii Hotels. Datorită integrării unui sistem de paginare Hotelurile sunt afișate doar 30 per pagină. Jos de tot utilizatorul poate naviga printre pagini cu ajutorul butoanelor: prima pagină/pagina anterioară/pagina următoare/ultima pagină. Alături de aceste butoane utilizatorul poate vedea la ce pagina se află.

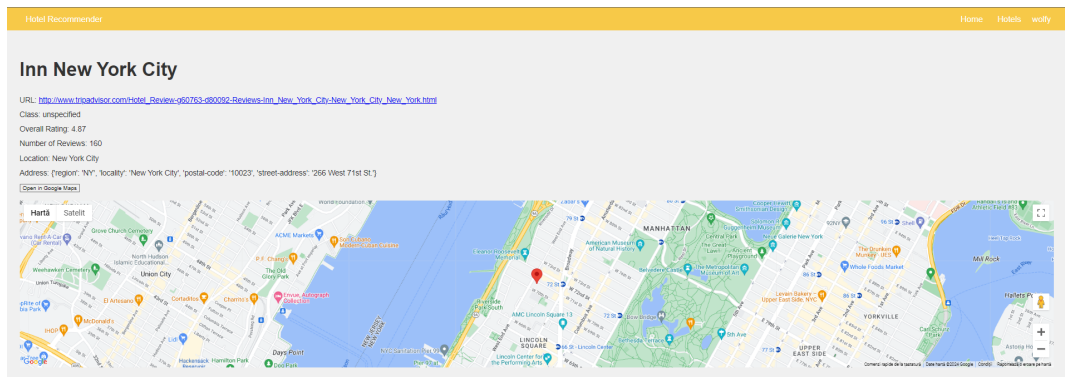


Figura 4.8: Partea de sus a paginii Hotel Detail.

În Figura 4.8 este prezentată partea de sus a paginii Hotel Detail. În această pagină avem bara de navigare. Sub aceasta avem secțiunea în care sunt prezentate datele hotelului: numele, url-ul catre pagina hotelului, clasa hotelului, evaluarea medie a hotelului, locația hotelului, adresa hotelului. Sub aceasta ne este prezentată o hartă implementată cu ajutorul Google Maps API și un buton care ne deschide o pagină nouă Google Maps cu locația hotelului.

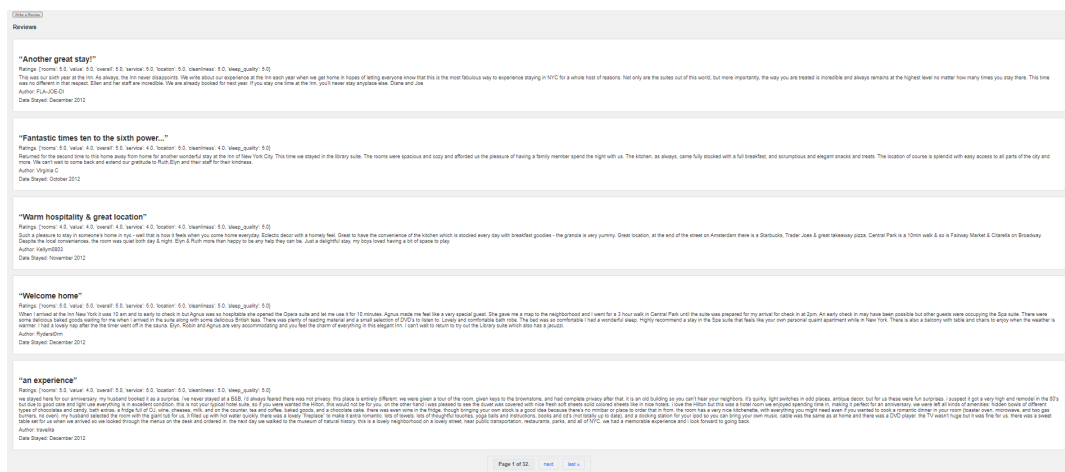


Figura 4.9: Partea de jos a paginii Hotel Detail.

În Figura 4.9 este prezentată partea de jos a paginii Hotel Detail. Aici este secțiunea dedicată recenziilor utilizatorilor care sunt afisate cate 5 per pagină cu ajutorul aceluiași sistem de paginare folosit pentru pagina Hotel.

Write a Review

Rate these fields from 1 to 5:

Service:

Cleanliness:

Overall:

Value:

Location:

Sleep Quality:

Rooms:

Enter the title of your review:

Write your detailed review:

Enter the date you stayed (e.g., June 2024):

Submit

Figura 4.10: Formularul de recenzii.

În Figura 4.10 este prezentat formularul de recenzie pe care un utilizator îl poate completa dacă acesta apasă butonul ”Write a Review”, care se află deasupra recenziilor.

Hotel Recommender Home Hotels wolffy

Account Details

Username: wolffy

Change Password

Old password:

New password:

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

New password confirmation:

Change Password

Figura 4.11: Pagina Account Details.

În Figura 4.11 este prezentată pagina account details. În această pagină utilizatorul poate să completeze un formular pentru schimbarea parolei. Acesta se realizează completând parola veche, parola nouă, care trebuie să îndeplinească aceleași condiții ca cea pe care o creezi prima dată în pagina Register și confirmarea noii parole prin rescrierea ei.

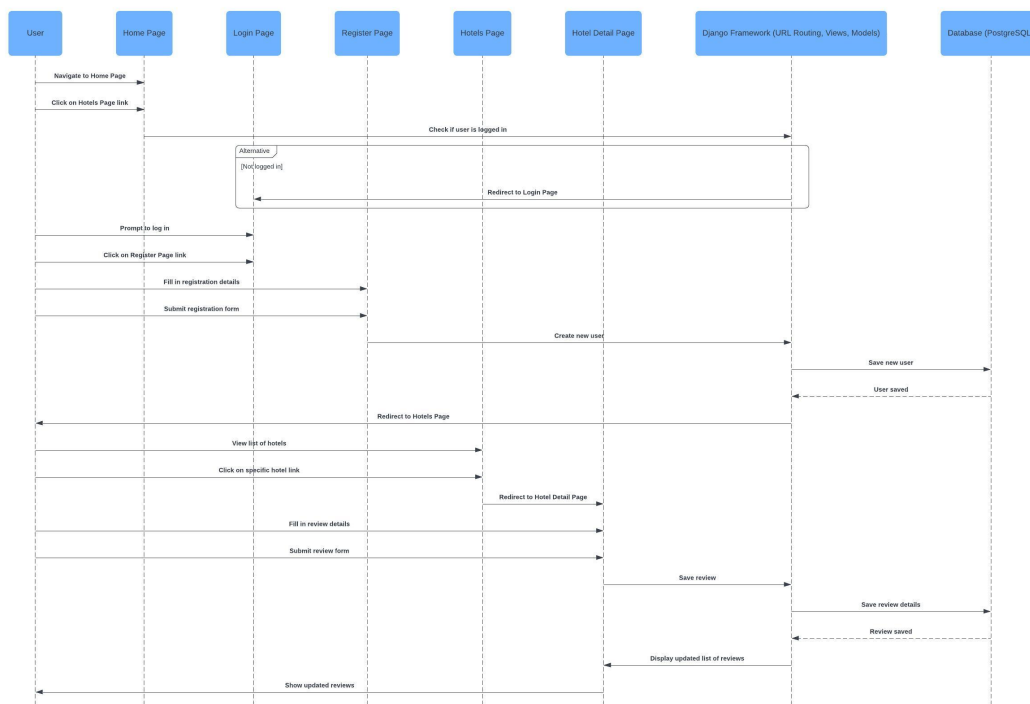


Figura 4.12: Diagrama de secvență a scenariului user recenziază un hotel.

Diagrama de secvență prezentată în Figura 4.12 ilustrează fluxul de interacțiuni între utilizator și componentele aplicației web pentru a realiza o recenzie a unui hotel. Inițial, utilizatorul navighează pe pagina principală și încearcă să acceseze pagina hotelurilor. Dacă utilizatorul nu este autentificat, este redirecționat către pagina de autentificare. De pe această pagină, utilizatorul poate accesa pagina de înregistrare pentru a crea un cont nou. După înregistrare, utilizatorul este autentificat automat și redirecționat către pagina hotelurilor. Aici, utilizatorul poate selecta un hotel specific și este redirecționat către pagina detaliilor hotelului, unde poate completa și trimite o recenzie. Diagrama evidențiază interacțiunile dintre utilizator, paginile web, Framework-ul Django (inclusiv rutarea URL-urilor, View-urile și Modelele) și baza de date PostgreSQL pentru a salva și a afișa recenziile.

4.2 Rezultatele experimentale

4.2.1 Testarea automată

Pentru testarea automată aplicației am folosit teste unitare. Testele unitare sunt o parte fundamentală a procesului de dezvoltare software, în special într-un framework precum Django. Ele ajută la asigurarea faptului că componentele individuale ale aplicației se comportă conform așteptărilor. În cazurile de testare furnizate, utilizăm cadrul de testare încorporat al Django, care se bazează pe modulul unittest al Python. Testele sunt împărțite în două categorii principale: Model tests și View tests. Model tests verifică comportamentul modelelor Django prin crearea de instanțe ale tabelelor Hotel, Review și HotelOverallRating și verificarea faptului că atributele și metodele acestora returnează rezultatele așteptate. De exemplu, testele confirmă că prin crearea unei instanțe Hotel se definesc corect numele și clasa acestuia și că reprezentarea în șir

a modelului Review corespunde titlului acestuia. Aceste teste ajută la depistarea problemelor legate de gestionarea datelor și de logica de afaceri într-un stadiu incipient.

View tests, pe de altă parte, se concentrează pe testarea diferitelor view-uri care gestionează cererile HTTP și returnează răspunsurile. Aceste teste simulează interacțiunile utilizatorului cu aplicația prin efectuarea de cereri către anumite URL-uri și verificarea codurilor de stare, a modelelor utilizate și a redirectionărilor rezultate. De exemplu, testele verifică dacă accesarea paginii principale sau a paginii cu lista hotelurilor returnează un status 200 OK și afișează șablonul corect. De asemenea, testele verifică dacă vizualizările care necesită autentificare, cum ar fi vizualizarea detaliilor hotelului, redirectionează corect utilizatorii neautentificați către pagina de autentificare. Prin includerea atât a Model tests, cât și a View tests, asigurăm acoperirea completă a funcționalității de bază a aplicației, ceea ce contribuie la menținerea calității și fiabilității codului pe parcursul dezvoltării și actualizărilor viitoare.

4.2.2 Testarea manuală

Pentru testarea manuală a aplicației am simulat singur un utilizator care încearcă toate funcționalitățile manual în toate condițiile pentru a verifica dacă aplicația funcționează corespunzător. De exemplu: am încercat înregistrarea prin a folosi o parolă care nu corespunde cerințelor și am observat dacă aplicația afișează eroarea corespunzătoare, am creat un utilizator și am făcut o recenzie apoi am verificat dacă aceasta a schimbat ceva în lista hotelurilor recomandate și dacă toate datele au fost salvate în baza de date prin comenzi SQL direct în baza de date.

4.2.3 Testarea de către utilizatori reali

Pentru acest tip de testare am pus aplicația la dispoziția a 10 utilizatori (prieteni/membri ai familiei). După ce aceștia au folosit aplicația i-am rugat să completeze un formular Google pentru a măsura unii parametri.

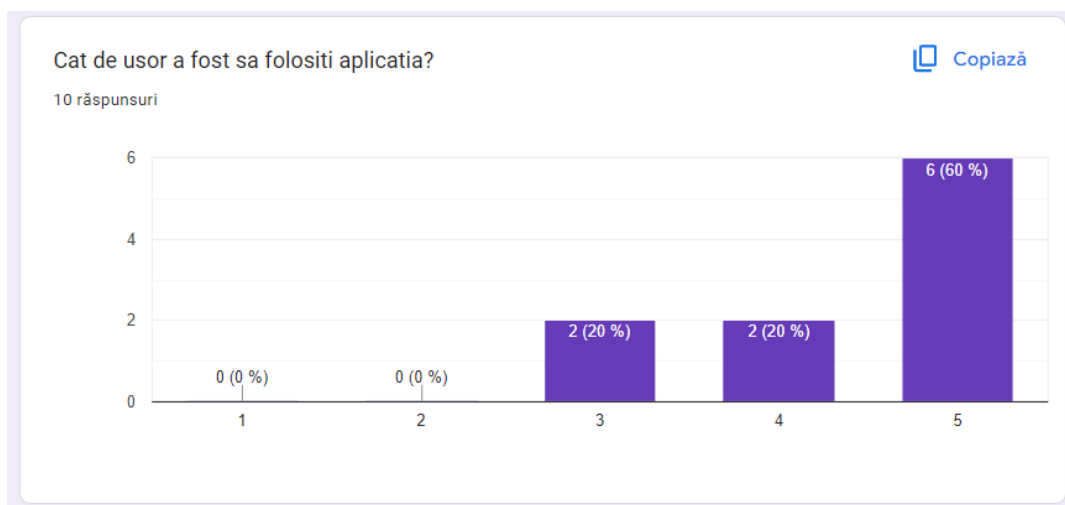


Figura 4.13: Graficul răspunsurilor la prima întrebare din formularul Google.

Rezultatele sondajului din Figura 4.13 indică faptul că majoritatea utilizatorilor au considerat aplicația ușor de utilizat. Dintre cei 10 respondenți, 60% (6 persoane) au acordat nota maximă de 5, ceea ce sugerează că experiența lor de utilizare a fost foarte pozitivă. În plus, 20% (2 persoane) au evaluat ușurința de utilizare cu nota 4, iar alte 20% (2 persoane) au dat nota 3. Niciunul dintre respondenți nu a acordat note mai mici de 3, ceea ce indică faptul că toți utilizatorii au avut o experiență de utilizare cel puțin satisfăcătoare. Aceste rezultate reflectă o percepție generală pozitivă a ușurinței de utilizare a aplicației, deși există loc pentru îmbunătățiri pentru a ajunge la un scor de 5 pentru toți utilizatorii.

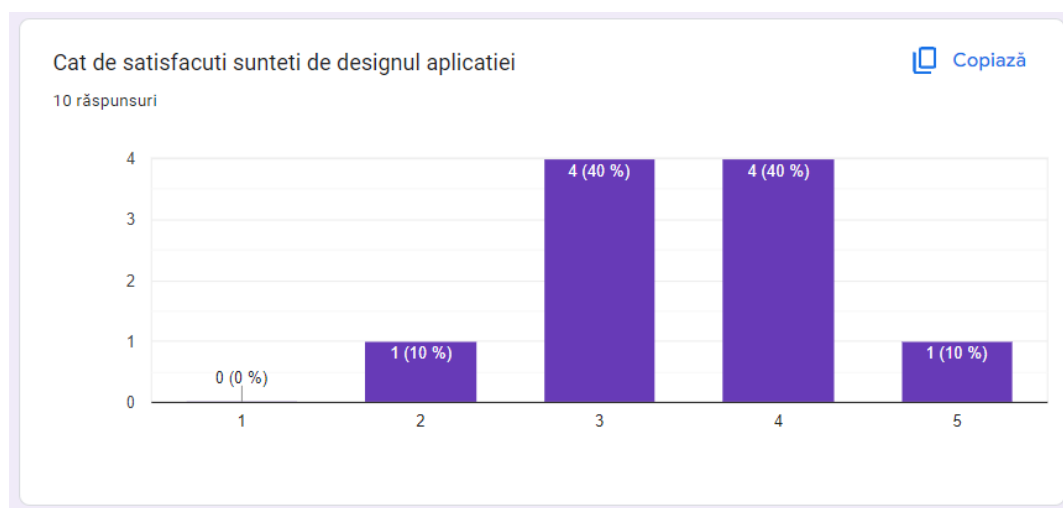


Figura 4.14: Graficul răspunsurilor la a doua întrebare din formularul Google.

Rezultatele sondajului privind satisfacția utilizatorilor față de designul aplicației, din Figura 4.14, arată o părere predominant pozitivă, dar cu o distribuție mai variată a răspunsurilor. Din cei 10 respondenți, 40% (4 persoane) au acordat nota 3 și tot 40% (4 persoane) au acordat nota 4, indicând o satisfacție moderată și, respectiv, ridicată. Doar 10% (1 persoană) au dat nota maximă de 5, ceea ce sugerează că există un mic segment de utilizatori foarte mulțumiți de design. În același timp, 10% (1 persoană) au evaluat designul cu nota 2, ceea ce indică faptul că există loc pentru îmbunătățiri pentru a satisface așteptările tuturor utilizatorilor. Nu au fost înregistrate note de 1, ceea ce sugerează că niciun utilizator nu a considerat designul complet nesatisfăcător. Per ansamblu, designul este perceput ca fiind bun, dar există oportunități pentru rafinări suplimentare pentru a crește nivelul de satisfacție generală.

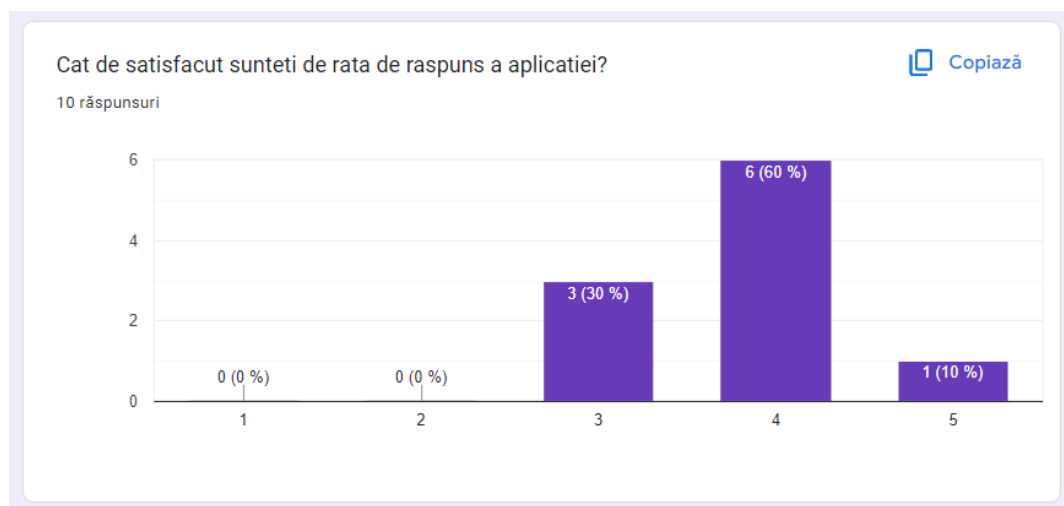


Figura 4.15: Graficul răspunsurilor la a treia întrebare din formularul Google.

Rezultatele sondajului referitoare la satisfacția utilizatorilor cu privire la rata de răspuns a aplicației, din Figura 4.15, arată că majoritatea utilizatorilor sunt mulțumiți de performanța acesteia. Din cei 10 respondenți, 60% (6 persoane) au evaluat rata de răspuns cu nota 4, indicând un nivel ridicat de satisfacție. În plus, 30% (3 persoane) au dat nota 3, ceea ce sugerează că aplicația răspunde adecvat, dar mai există loc pentru îmbunătățiri. Doar 10% (1 persoană) au acordat nota maximă de 5, ceea ce arată că foarte puțini utilizatori consideră performanța aplicației ca fiind excelentă. Niciun respondent nu a acordat note sub 3, indicând faptul că aplicația nu a fost percepută ca având probleme majore de performanță. Per ansamblu, rezultatele sugerează o percepție pozitivă asupra vitezei de răspuns, cu potențial pentru optimizări suplimentare pentru a atinge excelența în percepția utilizatorilor.

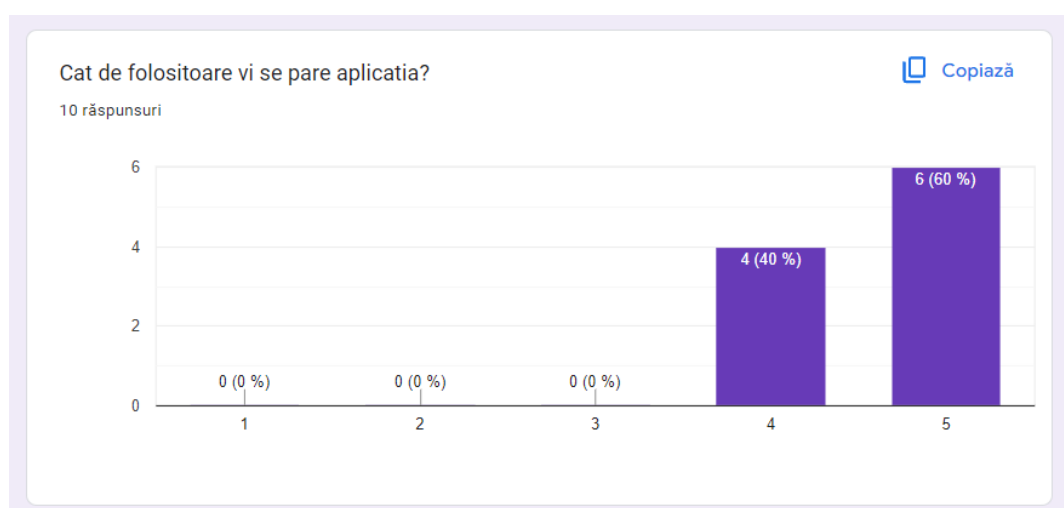


Figura 4.16: Graficul răspunsurilor la a patra întrebare din formularul Google.

Rezultatele sondajului din Figura 4.16 arată că utilizatorii consideră aplicația foarte utilă. Din cei 10 respondenți, 60% (6 persoane) au evaluat utilitatea aplicației cu nota maximă de 5, indicând un grad înalt de satisfacție. Alte 40% (4 persoane) au acordat nota 4, ceea ce arată că și acești utilizatori au găsit aplicația utilă, deși nu la cel mai înalt nivel posibil. Nu au fost acordate note sub 4, ceea ce sugerează că toți utilizatorii au perceput aplicația ca fiind cel puțin utilă. Aceste rezultate pozitive reflectă faptul că aplicația își îndeplinește bine funcția și răspunde nevoilor utilizatorilor, având un impact pozitiv în utilizarea lor cotidiană.

Capitolul 5

Concluzii, contribuții și direcții de continuarea a dezvoltării

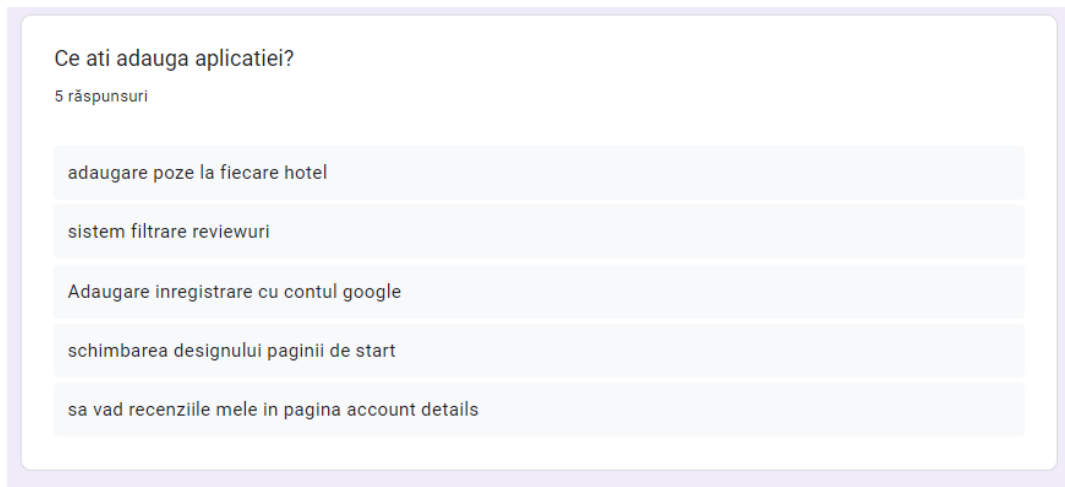
5.1 Concluzii

Această lucrare prezintă o analiză detaliată a importanței și funcționării sistemelor de recomandare, subliniind rolul lor esențial în diverse domenii, în special în turism. Sistemele de recomandare ajută la îmbunătățirea și personalizarea experienței călătorilor, oferind sugestii relevante și adaptate preferințelor individuale. În plus, aceste sisteme contribuie la creșterea satisfacției și fidelității clienților, promovând un turism mai informat și mai eficient. Pe parcursul lucrării, am comparat două dintre cele mai mari sisteme de recomandare de pe piață, Booking.com și TripAdvisor, evidențiind diferențele în abordările lor și modul în care fiecare reușește să ofere sugestii pertinente utilizatorilor săi. Această comparație a subliniat diversele strategii și tehnologii utilizate pentru a optimiza recomandările și a răspunde cât mai bine nevoilor utilizatorilor.

5.2 Contribuții

Contribuția principală a acestei lucrări constă în dezvoltarea unui sistem de recomandare avansat pentru hoteluri, care combină media recenziilor cu aprecierea similară a altor utilizatori. Acest algoritm de filtrare colaborativă oferă utilizatorilor recomandări mai precise și personalizate. Am integrat și testat cu succes acest sistem, asigurându-ne că este ușor de utilizat și eficient, conform feedback-ului primit de la utilizatori reali. În plus, am prezentat și analizat diferite tipuri de sisteme de recomandare, oferind o imagine de ansamblu asupra funcționării lor și comparându-le cu cele mai mari sisteme existente pe piață, demonstrând astfel relevanța și potențialul lor în îmbunătățirea experienței utilizatorilor. Acest proiect demonstrează eficacitatea combinării tehnologiilor Django și PostgreSQL pentru dezvoltarea de aplicații web scalabile și securizate, capabile să gestioneze și să proceseze seturi mari de date în timp real.

5.3 Direcții de continuare a dezvoltării



Ce ati adauga aplicatiei?

5 răspunsuri

- adaugare poze la fiecare hotel
- sistem filtrare reviewuri
- Adaugare inregistrare cu contul google
- schimbarea designului paginii de start
- sa vad recenzile mele in pagina account details

Figura 5.1: Răspunsurile la a cincea întrebare din formularul Google.

Pe baza feedback-ului primit de la utilizatori în formularul Google vizibil în Figura 5.1, direcțiile viitoare de dezvoltare ale aplicației includ mai multe îmbunătățiri esențiale pentru a îmbunătăți experiența utilizatorilor și a adăuga funcționalități suplimentare. Printre sugestiile utilizatorilor se numără adăugarea de poze la fiecare hotel, ceea ce ar permite o vizualizare mai detaliată și atractivă a opțiunilor disponibile. Un alt aspect important este implementarea unui sistem de filtrare a recenziilor, care ar facilita găsirea celor mai relevante opinii. De asemenea, integrarea opțiunii de înregistrare cu contul Google, prin folosirea unui API Google, ar simplifica procesul de autentificare, oferind o experiență mai fluidă. Schimbarea designului aplicației prin utilizarea unui toolkit de dezvoltare front-end, cum ar fi biblioteca React, ar putea moderniza și optimiza interfața aplicației, oferind un aspect mai atractiv și interactiv. În plus, adăugarea unei secțiuni în pagina "Account Details" unde utilizatorii să poată vedea recenziile lor anterioare ar oferi o gestionare mai bună a activităților personale. Aceste îmbunătățiri propuse reflectă angajamentul de a răspunde nevoilor utilizatorilor și de a continua evoluția aplicației pentru a asigura o experiență cât mai plăcută și eficientă.

Bibliografie

- [Agg16] Charu C. Aggarwal. *Recommender Systems, The Textbook*. Springer, 2016.
- [Boo] https://www.booking.com/content/how_we_work.ro.html.
- [DJ11] Alexander Felfernig Gerhard Friedrich Dietmar Jannach, Markus Zanker. *Recommender Systems, An Introduction*. Cambridge University Press, 2011.
- [Dja] <https://www.djangoproject.com/>.
- [FR11] Bracha Shapira Paul B. Kantor Francesco Ricci, Lior Rokach. *Recommender Systems Handbook*. Springer, 2011.
- [MA15] Mamta Bhusry Mohammad Aamir. Recommendation System: State of the Art Approach. *International Journal of Computer Applications*, 120, jun 2015.
- [Pos] <https://www.postgresql.org/about/>.
- [Rec] <https://neptune.ai/blog/recommender-systems-lessons-from-building-and-deplo>
- [Sha10] Nalin Sharda. *Tourism Informatics*. Information Science Reference, 2010.
- [Tri] <https://www.tripadvisor.com/business/insights/hotels/resources/tripadvisor-popularity-ranking>.