

Facultatea de Automatică și Calculatoare-Calculatoare

Inginerie Software

Hospital Management

Student: Niță Alexandru-Gabriel

Grupa 30233

Profesor coordonator: Burzo Adrian

CUPRINS

1. Introducere
2. Specificație Proiect
3. Diagrama Use Case
4. Implementarea propriu-zisă
5. Diagrame, Design Pattern
6. Limbaje de programare, frameworks, baze de date
7. Concluzii
8. Bibliografie
9. ReadMe

1. Introducere

Proiectul presupune o aplicație web de administrare online pentru un spital de stat, respectiv gestionarea operațiunilor asupra utilizatorilor în cadrul instituției medicale. Această aplicație web va facilita procesele legate de gestionarea pacienților sau programări. Scopul principal al aplicației este de a facilita procesele administrative, programările și comunicarea între personalul medical și pacienți. Proiectul oferă un sistem sigur și eficient pentru gestionarea informațiilor, precum și facilitarea comunicării în cadrul spitalului.

2. Specificație Proiect

Proiectul se axează pe funcționalități esențiale pentru administrarea eficientă a unui spital, inclusiv:

Sistem de autentificare:

- Login pentru administrator și utilizator.
- Înregistrare (Sign Up) pentru crearea unui cont de utilizator.

Comunicare:

- Implementare Chat Bot pentru întrebări frecvente și informații generale.
- Chat live pentru asistență 24/7 pentru a răspunde prompt la întrebările utilizatorilor.

Programări și Calendar:

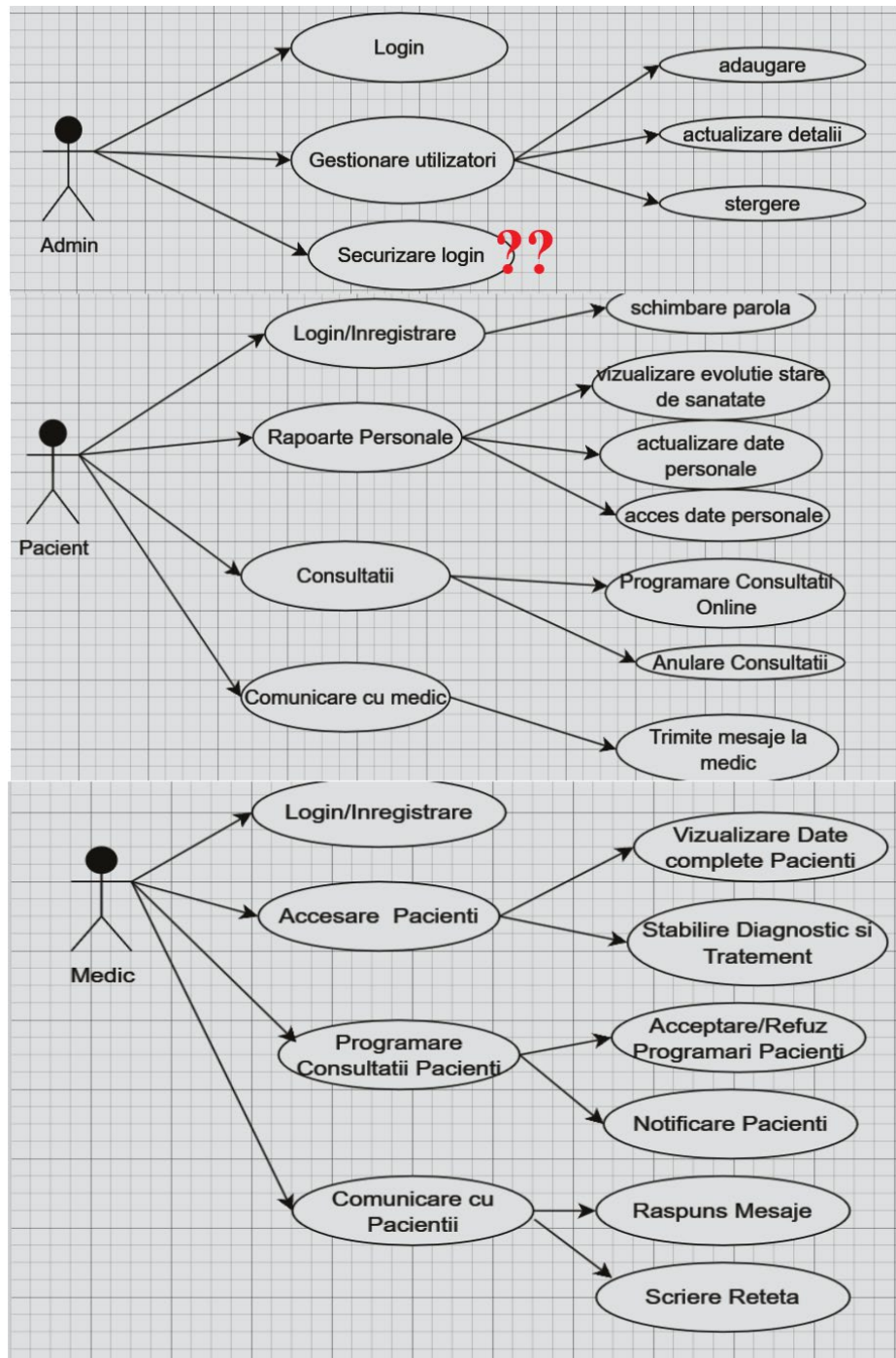
- Posibilitatea de a efectua programări online.
- Calendar interactiv pentru selectarea datelor disponibile pentru programări.

Afisare Medici:

- Afisarea listei de medici pe pagina principală pentru acces rapid.
- Informații detaliate despre fiecare medic, inclusiv specializare și program de lucru.

3. Diagrama Use Case

Diagrama Use Case evidențiază interacțiunile dintre utilizatori și sistem, ilustrând scenariile cheie.



4. Implementarea propriu-zisă

Implementarea propriu-zisă în cadrul proiectului dedicat platformei de management a spitalului a implicat utilizarea unui set divers de tehnologii și framework-uri pentru a asigura o funcționalitate robustă și o experiență utilizator de calitate. Iată detalii specifice legate de implementarea propriu-zisă a diferitelor aspecte ale proiectului:

Backend Java Spring Boot:

Pentru dezvoltarea backend-ului sistemului de gestionare a spitalului, am folosit Java Spring Boot. Aceasta a furnizat un cadru solid, iar serviciile Spring au fost utilizate pentru a organiza și separa funcționalitățile logice ale aplicației. Backend-ul gestionează operațiunile specifice unui spital, precum manipularea datelor pacienților, programărilor și informațiilor medicale.

Frontend React pentru Interfața Utilizator:

Interfața utilizator a sistemului a fost dezvoltată folosind framework-ul React. Aceasta oferă o interactivitate sporită și un management eficient al stării aplicației. Componentele React au fost implementate pentru a construi interfețe modulare și ușor de navigat, adaptate specific nevoilor unui mediu spitalicesc. Interfața utilizator oferă acces rapid la informații relevante pentru pacienți și personalul medical.

CSS și Bootstrap pentru Stilizare:

Stilizarea paginilor web s-a realizat prin intermediul CSS, asigurând un aspect modern și ergonomic, potrivit pentru mediul medical. Bootstrap a fost integrat pentru a facilita design-ul responsiv și pentru a garanta o experiență coezivă pe diferite dispozitive, inclusiv pe dispozitive mobile utilizate în cadrul spitalului. Aspectul vizual a fost adaptat pentru a oferi o experiență plăcută utilizatorilor în contextul specific al unei platforme spitalicești.

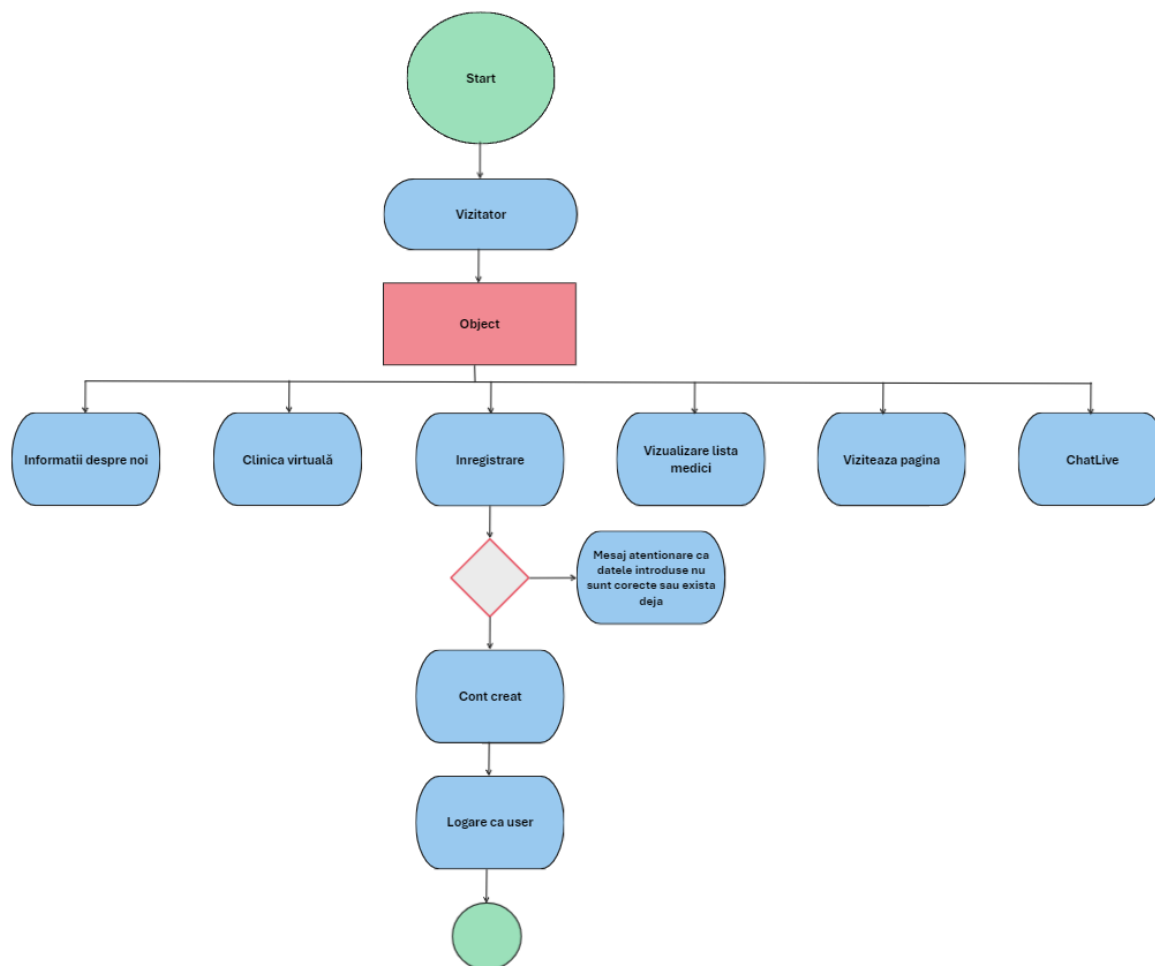
Gestionarea Bazei de Date MySQL:

MySQL a fost ales ca sistem de gestionare a bazelor de date pentru a stoca eficient informațiile legate de pacienți, programări, personal medical și alte aspecte specifice gestionării unui spital. S-au implementat interogări pentru a extrage date relevante în cadrul programărilor și a altor operațiuni necesare într-un mediu medical.

Această abordare tehnologică și implementarea detaliată au condus la crearea unei platforme solide și eficiente pentru gestionarea informațiilor și proceselor în cadrul spitalului, asigurând o experiență utilizator de calitate și funcționalități avansate pentru administrarea eficientă a resurselor medicale.

5. Diagrame, Design Pattern

Am ales diagrama de activitate:



Design Pattern->Singleton

Strategy este un pattern de proiectare comportamental care permite definirea unei familii de algoritmi, încapsularea fiecărui algoritm și făcându-i interschimbabili. Acest pattern permite clientului să aleagă algoritmul dorit dintr-o familie fără a afecta clientul însuși.

```
no usages
@Service
public class UserService {

    4 usages
    private final UserRepositoryStrategy userRepositoryStrategy;

    @Autowired
    public UserService(UserRepositoryStrategy userRepositoryStrategy) {
        this.userRepositoryStrategy = userRepositoryStrategy;
    }

    no usages
    public User getUserById(Long id) {
        return userRepositoryStrategy.findById(id);
    }

    no usages
    public List<User> getAllUsers() {
        return userRepositoryStrategy.findAll();
    }

    no usages
    public void saveUser(User user) {
        userRepositoryStrategy.save(user);
    }
}
```

6. Limbaje de programare, Frameworks, Baze de date:

Backend:

Limbaj de programare: Java

Framework: Spring Boot

Gestionare dependențe: Maven

Frontend:

Limbaj de programare: JavaScript (React pentru interfața utilizator)

Stilizare: CSS, Bootstrap pentru design responsiv

Bază de date:

Sistem de Gestionare a Bazelor de Date: MySQL

ORM (Object-Relational Mapping): Spring Data JPA

7. Bibliografie

- <https://spring.io/guides/gs/spring-boot/>
- <https://www.freecodecamp.org/news/the-best-bootstrap-examples/>
- <https://www.youtube.com/watch?v=bMknfKXIFA8>

8. Concluzii

Proiectul de Hospital Management și-a atins cu succes obiectivele propuse, oferind o soluție inovatoare și eficientă pentru administrarea online a unui spital de stat. Prin intermediul aplicației web dezvoltate, s-au facilitat procesele administrative, gestionarea pacienților și programărilor, contribuind la îmbunătățirea comunicării între personalul medical și pacienți.

9.ReadMe

- Java Development Kit (JDK)
- MySQL Workbench
- Creare bază de date

Rulare aplicație:

- Deschizi proiectul cu un IDE
- Run application
- Deschizi browserul și navighează la adresa <http://localhost:8080/>