



Hospital Management System

Name:Moldovan
Claudia
Group:30234

Table of Contents

1. Project Specification	3
2. Functional Requirements	3
3. Use Case	4
4. Supplementary Specification	7
Non-functional Requirements	7
Design Constraints	7
5. Glossary	7
5. Domain Model	8
6. Architectural Design	8
7. Package Design	9
8. Component and Deployment Diagram	9
Deployment Diagram	9
Component Diagram	10
Deliverable 3	10
Design Model	10
Bibliography	11

1.Project Specification

Aplicatia este un site web ce are in vedere implementarea unui sistem de management a activitatii in spital. Prin aceasta aplicatie se pot realiza operatiile generale pe care le face un medic si receptionist in mod digital pentru o mai buna comunicare si digitalizare a datelor.

2.Functional Requirements

Cateva dintre functional requirements sunt urmatoarele:

Orice tip de user: logare in cont

Receptioner/Admin: -adaugarea unei noi programari

- stergere a unei programari

- modificarea unei programari

- adaugarea unui nou pacient

- stergere a unui pacient

- afisare consultatii pentru medici

- afisarea fisei facute de medic dupa consult

Admin-modificare/stergere/adaugare angajat

Medic: -adaugarea unui consult/diagnostic

- afisarea consultatiilor

- afisarea pacientilor

Adaugare in baza de date a fiecarei noi programari, fiecarui nou pacient, medic sau receptionist.

3. Use Case



Use Case Model 1 Use Cases Identification

Use-Case:adaugare programare

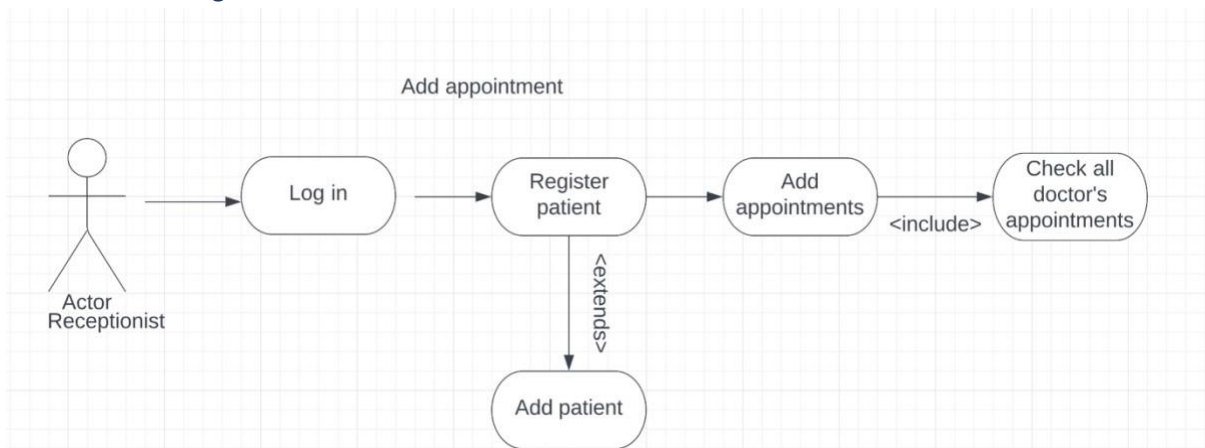
Level: user-goal level

Primary Actor:receptionist

Main success scenario:Se locheaza, inregistreaza pacientul pentru programare, iar in cazul in care acesta nu exista il v-a adauga in baza de date. Mai departe pentru a adauga programarea este obligatoriu sa se verifice ce doctori au liber in program.

Extensions:adaugarea pacientului

UML Use Case Diagrams



Use Case Model 2 Use Cases Identification

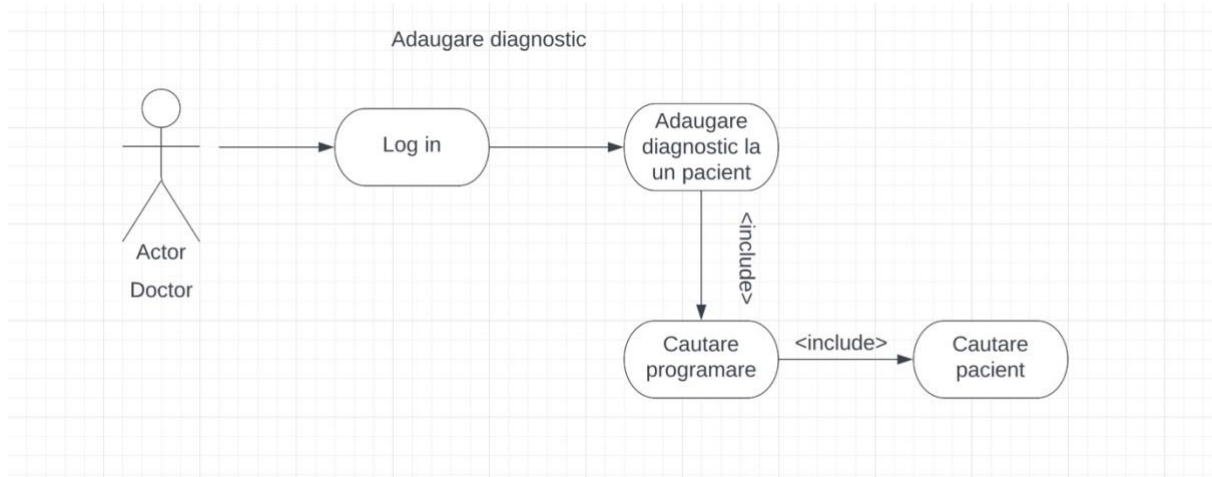
Use-Case: adaugare diagnostic

Level: user-goal level

Primary Actor:doctor

Main success scenario:Se logheza si doreste adaugarea diagnosticului. Pentru aceasta operatiune trebuie sa caute programare in functie de pacient. Extensions:-

UML Use Case Diagram



Use Case Model 3 Use Cases Identification

Use-Case:adaugare un nou receptioner

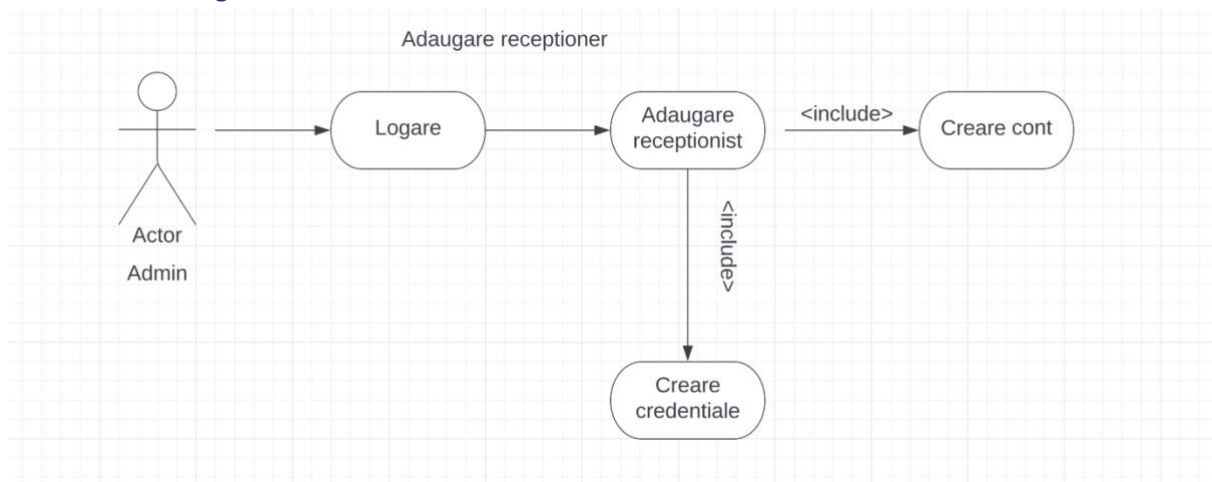
Level: user-goal level

Primary Actor:admin

Main success scenario:Se locheaza si adauga receptionistul. Pentru aceasta trebuie sa-i creeze un cont si sa-i asigneze niste credentiale de logare.

Extensions:-

UML Use Case Diagrams



4. Supplementary Specification

Non-functional Requirements

1. Verificare la logare daca userul este autorizat pentru actiunile ce doreste sa le realizeze
2. SQL injection
3. Portabilitate
4. Securitate-pentru conectare necesita username si parola
5. Disponibilitate-site ul este disponibil oricand

Design Constraints

Aplicatia foloseste ca limbaj de programare Java Spring. Pentru mentinerea datelor ii este asociata baza de date SQL. FrontEndul aplicatiei este implementat in Angular.

5. Glossary

UML- Unified Modeling Language este un limbaj standard pentru descrierea de modele și specificații pentru software.

SQL injection- Acest tip de atac constă în inserarea unui script de tip SQL (Structured Query Language) într-o anumită aplicație prin intermediul unei căsuțe de input. Mai simplu, dacă un hacker găsește pe un site un formular de înscriere el va încerca să introducă acolo o comandă SQL care poate citi informații din baza de date.

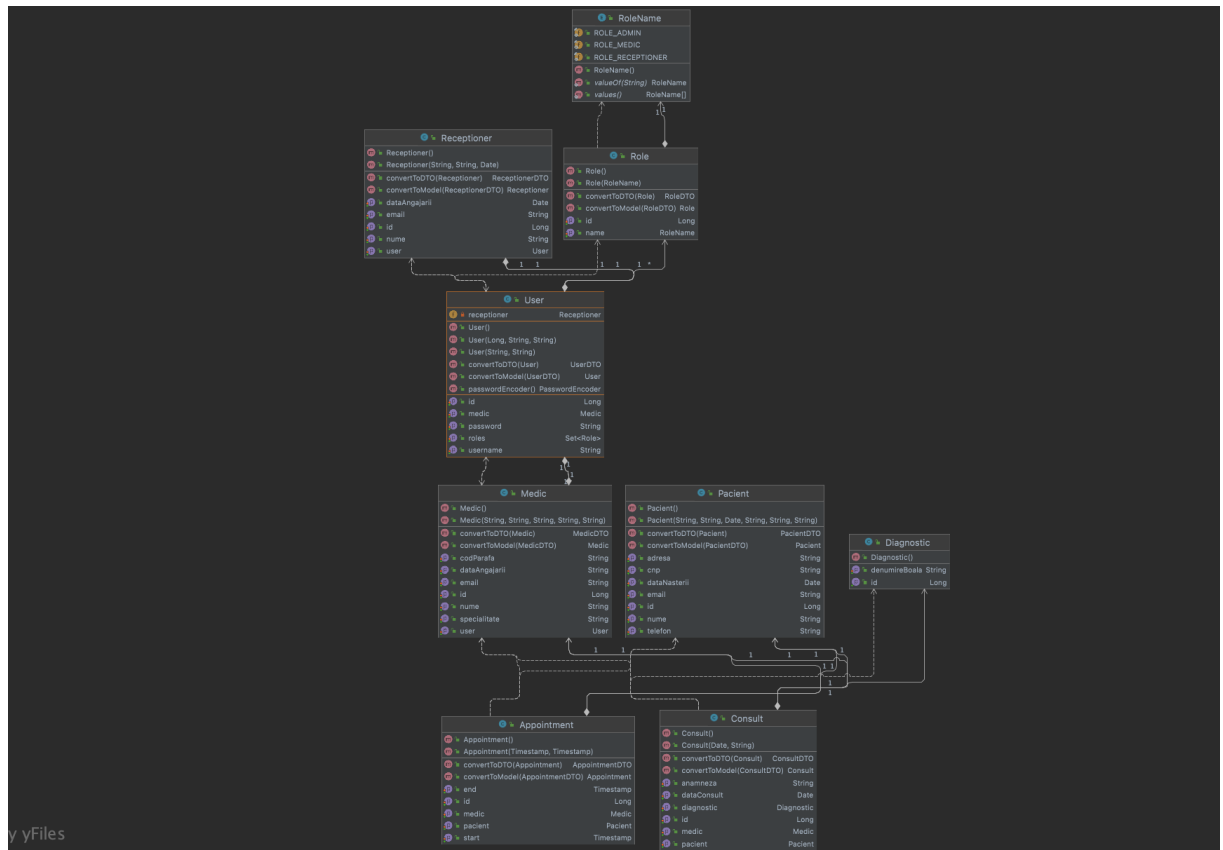
SQL -(de la numele englez Structured Query Language - limbaj de interogare structurat - care se pronunță este un limbaj de programare specific pentru manipularea datelor în sistemele de manipulare a bazelor de date relaționale (RDBMS), iar la origine este un limbaj bazat pe algebra relațională.

Java Spring - este o platformă Java open source. Platforma ce oferă un suport vast pentru crearea unei infrastructuri de dezvoltare a aplicațiilor Java. Deasemenea Java Spring Boot structurează programul și legăturile între entități, ușurând lucrul programatorului.

Angular-este o platformă de dezvoltare web cu sursă deschisă bazată pe limbajul TypeScript.

TypeScript- este un limbaj de programare open source dezvoltat și menținut de Microsoft. Este un superset sintactic al limbajului JavaScript și asigură un sistem de tipuri opțional. TypeScript este proiectat pentru dezvoltarea de aplicații de mari dimensiuni și se compilează în JavaScript.

5. Domain Model



6. Architectural Design

Layer architecture se referă la organizarea software-ului sau a sistemelor în straturi distincte sau niveluri de complexitate. Fiecare strat este responsabil pentru un set specific de funcționalități și interacționează cu straturile adiacente pentru a crea o aplicație sau un sistem complex.

Layer architecture este o metodă comună de proiectare a sistemelor de software, deoarece oferă o structură clară și modulară pentru dezvoltarea și mentenanța aplicațiilor. Aceasta poate fi utilizată într-o gamă largă de aplicații și sisteme, inclusiv în aplicații web, baze de date și rețele de calculatoare.

O asemenea arhiectura tipică poate include mai multe straturi, cum ar fi:

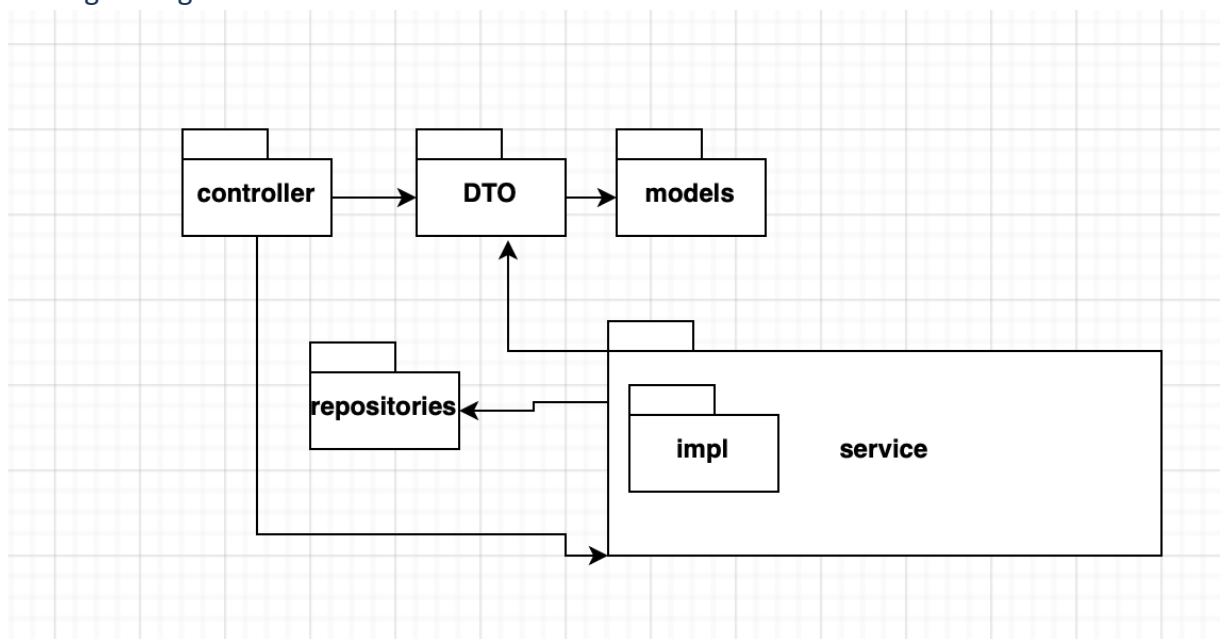
Straturile de interfață utilizator (UI) - care se ocupă de interacțiunea cu utilizatorul.

Straturile de logică de afaceri - care conțin codul care manipulează datele și implementează logica de afaceri.

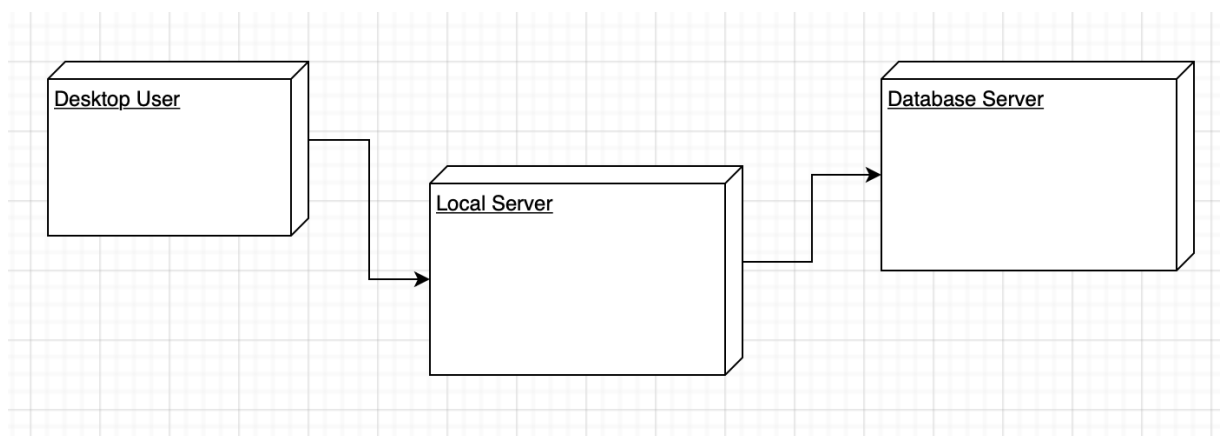
Straturile de acces la date - care se ocupă cu interacțiunea cu bazele de date sau alte surse de date externe.

Folosind această arhitectură, fiecare strat poate fi dezvoltat și testat separat, ceea ce face mai ușor dezvoltarea, testarea și mentenanța sistemelor complexe.

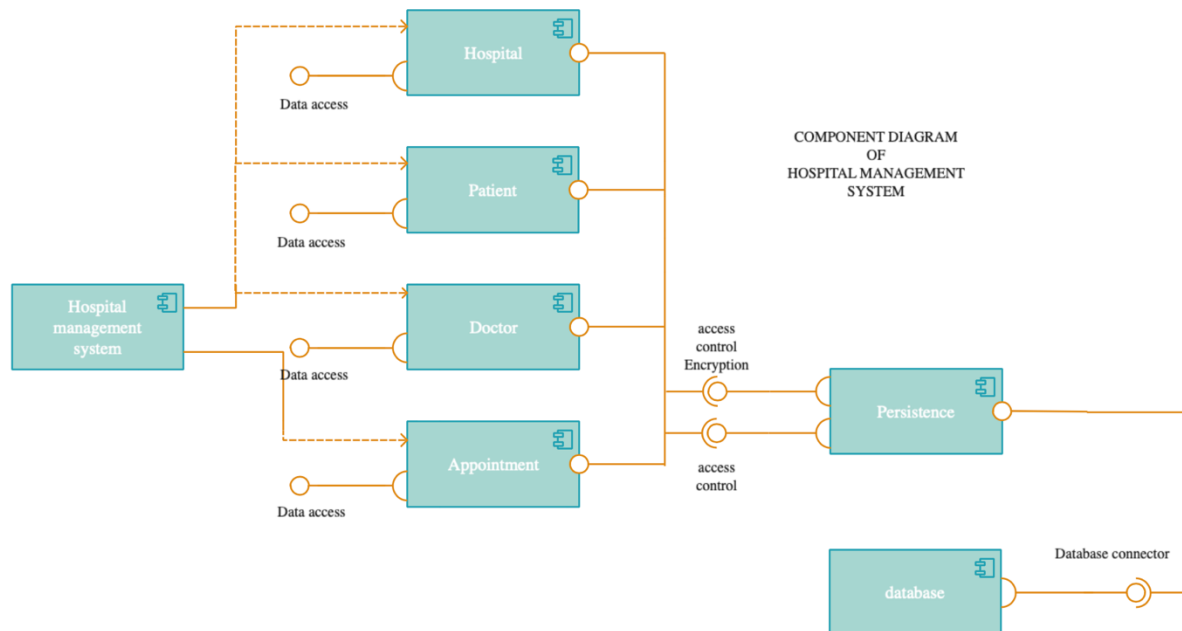
7. Package Design



8. Component and Deployment Diagram Deployment Diagram



Component Diagram



Deliverable 3

Design Model

Dynamic Behavior

[Create the interaction diagrams (1 sequence, 1 communication diagrams) for 2 relevant scenarios]

Class Diagram

[Create the UML class diagram; apply GoF patterns and motivate your choice]

Data Model

[Create the data model for the system.]

System Testing

[Describe the testing methods and some test cases.]

Future Improvements

[Present some features that apply to the application scope.]

Conclusion

Bibliography