**StartUp**

**Veb aplikacija za nalaženje studentskih praksi**

**Detaljni arhitekurni projekat**

**Verzija 1.0**

Pregled izmena

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autor** |
| 30.11.2022. | 1.0 | Inicijalna verzija | Halida, Aldina |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Sadržaj**

[1. Cilj dokumenta 5](#_Toc127198785)

[2. Opseg dokumenta 5](#_Toc127198786)

[3. Reference 5](#_Toc127198787)

[4. Predstavljanje arhitekture 5](#_Toc127198788)

[5. Ciljevi i ograničenja arhitekture 5](#_Toc127198789)

[6. Pogled na slučajeve korišćenja 6](#_Toc127198790)

[6.1 Dijagram slučajeva korišćenja 6](#_Toc127198791)

[6.2 Kratak opis slučajeva korišćenja 8](#_Toc127198792)

[6.2.1 Pregled osnovnih podataka 9](#_Toc127198793)

[6.2.2 Pregled informacija o praksama 9](#_Toc127198794)

[6.2.3 Pregled informacija o kompanijama 9](#_Toc127198795)

[6.2.4 Filtriranje podataka prema lokaciji 9](#_Toc127198796)

[6.2.5 Filtriranje podataka prema poslu 9](#_Toc127198797)

[6.2.6 Prijavljivanje 9](#_Toc127198798)

[6.2.7 Prijavljivanje na oglas 9](#_Toc127198799)

[6.2.8 Ocenivanje kompanija 9](#_Toc127198800)

[6.2.9 Kontaktiranje kompanija 10](#_Toc127198801)

[6.2.10 Dodavanje oglasa 10](#_Toc127198802)

[6.2.11 Brisanje oglasa 10](#_Toc127198803)

[6.2.12 Kreiranje naloga 10](#_Toc127198804)

[6.2.13 Brisanje naloga 10](#_Toc127198805)

[6.2.14 Ažuriranje naloga 10](#_Toc127198806)

[6.2.15 Ažuriranje oglasa 11](#_Toc127198807)

[6.2.16 Pretraživanje oglasa 11](#_Toc127198808)

[6.2.17 Dodavanje u listu omiljenih 11](#_Toc127198809)

[7. Pogled na logičku arhitekturu sistema 11](#_Toc127198810)

[7. 1 Pregled arhitekture – organizacija paketa i podsistema u slojeve 12](#_Toc127198811)

[7.1.1 Korisnički interfejs 12](#_Toc127198812)

[7.1.2 Aplikaciona logika 13](#_Toc127198813)

[7.1.3 Pristup podacima 13](#_Toc127198814)

[7.1.4 HTML 13](#_Toc127198815)

[7.1.5 ASP.NET 13](#_Toc127198816)

[7.1.6 SQL 13](#_Toc127198817)

[8. Pogled na procese 13](#_Toc127198818)

[8.1 Procesi 14](#_Toc127198819)

[8.1.1 Web čitač 14](#_Toc127198820)

[8.1.2 Web server 14](#_Toc127198821)

[8.1.3 MySQL Server 15](#_Toc127198822)

[9. Pogled na raspoređivanje sistema 15](#_Toc127198823)

[9.1 Klijent 15](#_Toc127198824)

[9.2 Veb server 15](#_Toc127198825)

[9.3 DBMS server 15](#_Toc127198826)

[10. Pogled na implementaciju sistema 16](#_Toc127198827)

[10.1 Model domena 16](#_Toc127198828)

[10.2 Šema baze podataka 16](#_Toc127198829)

[10.3 Komponente sistema 17](#_Toc127198830)

[10.3.1. Komponente korisničkog interfejsa 17](#_Toc127198831)

[10.3.2 Komponente aplikacione logike 17](#_Toc127198832)

[10.3.3 Komponente za pristup podacima 18](#_Toc127198833)

[11. Performanse 18](#_Toc127198834)

[12. Kvalitet 19](#_Toc127198835)

Detaljni arhitekturni projekat

# 1. Cilj dokumenta

Cilj ovog dokumenta je detaljni opis arhitekture StartUp portala.

# 2. Opseg dokumenta

Dokument se odnosi na StartUp portal koji će biti razvijen od strane Beta-e. StartUp predstavlja skraćenicu za Veb aplikaciju za nalaženje praksi. Namena sistema je efikasno prezentovanje, kreiranje i održavanje sadržaja vezanih za prakse.

# 3. Reference

Spisak korišćene literature:

1. StartUp – Predlog projekta, Beta-StartUp-01, V1.0, 2022, Beta.
2. StartUp – Planirani raspored aktivnosti na projektu, V1.0, 2022, Beta.
3. StartUp – Plan realizacije projekta, V1.0, 2022, Beta.
4. StartUp – Vizija sistema, V1.0, 2022, Beta.
5. StartUp – Specifikacija zahteva, V1.0, 2022, Beta.

# 4. Predstavljanje arhitekture

Arhitektura sistema u dokumentu je prikazana kao serija pogleda na sistem: pogled na slučajeve korišćenja, pogled na procese, pogled na razmeštaj komponenti sistema i pogled na implementaciju. Ovi pogledi su predstavljeni odgovarajućim UML dijagramima.

# 5. Ciljevi i ograničenja arhitekture

Ključni zahtevi i sistemska ograničenja koja imaju značajan uticaj na izbor arhitekture i projektovanje sistema su:

1. StartUp portal će biti implementiran kao Web aplikacija zasnovana na ASP.NET skripting jeziku i SQL bazi podataka [4].
2. Klijentski deo StartUp portala će biti optimizovan za sledeće Web čitače: Internet Explorer 6.0 i noviji, Opera 8.0 i noviji, kao i Firefox (Mozilla) [4].
3. Svi zahtevi u pogledu performansi dati u [5] moraju biti uzeti u obzir pri izboru arhitekture i razvoju sistema.
4. Klasifikaciju publikacija treba standardizovati sa postojećom tipologijom propisanom od strane Ministarstva za nauku.

# 6. Pogled na slučajeve korišćenja

U ovom odeljku je dat pogled na slučajeve korišćenja definisane u specifikaciji zahteva [5].

Slučajevi korišćenja StartUp portala su:

* Pregled informacija
  + Pregled informacija o praksama
  + Pregled informacija o kompanijama
  + Pretražvianje praksi
  + Filtriranje prema lokaciji
  + Filtriranje prema oblasti rada
* Prijavljivanje
* Ažuriranje podataka
* Dodavanje oglasa
  + Ažuriranje podataka o praksama
  + Brisanje oglasa
* Kreiranje, brisanje i ažuriranje naloga
  + Kreiranje novog studenta ili kompanije
  + Brisanje studenta ili kompanije
  + Ažuriranje studenta ili kompanije
* Ocenivanje kompanije
* Kontaktiranje kompanije
* Prijavljivanje na praksu

Ove slučajeve korišćenja mogu da iniciraju posetilac, student, kompanija ili administrator.

## 6.1 Dijagram slučajeva korišćenja

Osnovni UML dijagram koji prikazuje korisnike i slučajeve korišćenja StartUp portala prikazan je na sledećoj slici:

Diagram

Description automatically generated

Slučajevi korišćenja *pregled informacija* i *kreiranje, brisanje i arhiviranje članova* i *ažuriranje podataka o praksama* obuhvataju složenije radnje koje se mogu razložiti dalje razložiti na pojedinačne slučajeve korišćenja.

Detaljni UML dijagram za slučaj korišćenja *pregled informacija* je prikazan na sledećoj slici:

Diagram

Description automatically generated

Detaljni UML dijagram za slučaj korišćenja *kreiranje, brisanje i arhiviranje članova* je prikazan na sledećoj slici:

Diagram

Description automatically generated

Detaljni UML dijagram za slučaj korišćenja *ažuriranje podataka o oglasima* je prikazan na sledećoj slici:

Diagram

Description automatically generated with medium confidence

## 6.2 Kratak opis slučajeva korišćenja

### 6.2.1 Pregled osnovnih podataka

Kratak opis: Prikaz stranice portala sa osnovnim podacima.

Akteri: Posetilac portala, Student, Kompanija, Administrator.

### 6.2.2 Pregled informacija o praksama

Kratak opis: Prikaz starnice sa svim oglasima.

Akteri: Posetilac portala, Student, Kompanija, Administrator.

### 6.2.3 Pregled informacija o kompanijama

Kratak opis: Prikaz starnice sa kompanijama koje učestvuju u postavljanju oglasa.

Akteri: Posetilac portala, Student, Kompanija, Administrator.

### 6.2.4 Filtriranje podataka prema lokaciji

Kratak opis**:** Izborom lokacije izlistavaju sve svi oglasi praksi koje se nalaze na toj lokaciji.

Akteri: Posetilac portala, Student, Kompanija, Administrator.

### 6.2.5 Filtriranje podataka prema poslu

Kratak opis: Izborom lokacije izlistavaju sve svi oglasi za određeni posao.

Akteri: Posetilac portala, Student, Kompanija, Administrator.

### 6.2.6 Prijavljivanje

Kratak opis: Prijavljivanje korisnika na veb aplikaciju u cilju pristupa specifičnim funkcijama koje zahtevaju autorizaciju.

Akteri: Student, Kompanija, Administrator.

### 6.2.7 Prijavljivanje na oglas

Kratak opis: Student se prijavljuje za odabranu praksu.

Akteri: Student.

### 6.2.8 Ocenivanje kompanija

Kratak opis: Student može oceniti neku od kompanija.

Akteri: Student.

### 6.2.9 Kontaktiranje kompanija

Kratak opis: Student može kontaktirati kompaniju.

Akteri: Student.

### 6.2.10 Dodavanje oglasa

Kratak opis: Dodavanje novog oglasa od strane kompanije.

Akteri**:** Kompanija, Administrator

### 6.2.11 Brisanje oglasa

Kratak opis: Brisanje oglasa od strane kompanije.

Akteri: Kompanija, Administrator.

### 6.2.12 Kreiranje naloga

Kratak opis: Kreiranje korisničkog naloga za novog člstudenta ili kompaniju.

Akteri: Student, Kompanija, Administrator.

### 6.2.13 Brisanje naloga

Kratak opis: Brisanje korisničkog naloga i podataka za postojećeg korisnika.

Akteri: Kompanija, Student, Administrator.

### 6.2.14 Ažuriranje naloga

Kratak opis: Ažuriranje podataka o sebi od strane prijavljenogstudenta ili kompanije.

Akteri: Student, Kompanija, Administrator.

### 6.2.15 Ažuriranje oglasa

Kratak opis: Ažuriranje podataka o oglasu od strane prijavljenogstudenta ili kompanije.

Akteri: Kompanija, Administrator.

### 6.2.16 Pretraživanje oglasa

**Kratak opis:**

Ažuriranje podataka o oglasu od strane prijavljenogstudenta ili kompanije.

**Akteri:**

Student, Posetilac, Kompanija, Administrator.

### 6.2.17 Dodavanje u listu omiljenih

**Kratak opis:**

Student dodaje praksu u listu omiljenih.

**Akteri:**

Student.

# 7. Pogled na logičku arhitekturu sistema

U ovom odeljku je dat pregled logičke arhitekture sistema. Ovaj pogled sadrži opis najznačajnijih klasa, njihove organizacije u pakete i podsisteme, i organizacija podsistema u slojeve. U cilju opisivanja dinamičkih aspekata arhitekture, ovaj odeljak može da uključi opise realizacije najznačajnijih slučajeva korišćenja. Da bi se ilustrovala veza između arhitekturno značajnih klasa, podsistema, paketa ili slojeva moguće je uključiti i odgovarajuće dijagrame klasa.

Logički pogled na StartUp portal obuhvata 3 glavna paketa: Korisnički interfejs, Aplikaciona logika, Pristup podacima.

Paket *Korisnički interfejs* sadrži Web stranice, ASP.NET skripte i multimedijalni sadržaj koji realizuju grafički dizajn i forme preko kojih korisnici sistema komuniciraju sa sistemom.

Paket *Aplikaciona logika* predstavlja srednji sloj sistema koji sadrži ASP.NET skripte zadužene za realizaciju funkcionalnosti specifičnih za domen sistema koji se razvija.

Paket *Pristup podacima* sadrži ASP.NET skripte koje predstavljaju interfejs za pristup, dodavanje i ažuriranje podataka koji se čuvaju u bazi podataka.

## 7. 1 Pregled arhitekture – organizacija paketa i podsistema u slojeve

**Diagram

Description automatically generated**

### 7.1.1 Korisnički interfejs

sloj

Ovaj sloj realizuje korisnički interfejs portala. U njemu su sadržane sve HTML, multimedijalni sadržaji i PHP skripte koje generišu HTML stranice preko kojih korisnici komuniciraju sa sistemom.

Sloj korisničkog interfejsa zavisi od sloja aplikacione logike, kao i paketa React.js i ASP.NET.

### 7.1.2 Aplikaciona logika

sloj

Sloj aplikacione logike je srednji sloj u troslojnoj arhitekturi StartUp portala. Sadrži ASP.NET skripte koje realizuju funkcionalnost karakterističnu za domen primene portala i uspostavljaju vezu između korisničkog interfejsa i sloja za pristup podacima.

Ovaj sloj zavisi od sloja za pristup podacima i ASP.NET paketa.

### 7.1.3 Pristup podacima

sloj

Sloj za pristup podacima se nalazi na dnu troslojne arhitekture i sadrži ASP.NET skripte zadužene za pribavljanje, dodavanje i ažuriranje podataka koji se čuvaju u SQL bazi podataka.

Ovaj sloj ne zavisi od drugih slojeva, ali je zavisan od paketa ASP.NET i SQL baza podataka.

### 7.1.4 HTML

tehnologija

Tehnologija HTML-a definiše gradivne elemente stranica koje se prikazuju u Web čitaču i koje omogućavaju prikaz formatiranih informacija i realizaciju formi za unos i ažuriranje podataka.

### 7.1.5 ASP.NET

tehnologija

Tehnologija ASP.NET-a obezbeđuje mehanizam za pisanje i izvršavanje skripti na strani servera. Ove skripte mogu da generišu HTML kod koji realizuje korisnički interfejs i pristupaju bazi podataka u cilju pribavljanja, unosa i ažuriranja podataka.

### 7.1.6 SQL

DBMS

SQL predstavlja sistem za upravljanje bazama podataka koji će se koristiti za realizaciju StartUp portala.

# 8. Pogled na procese

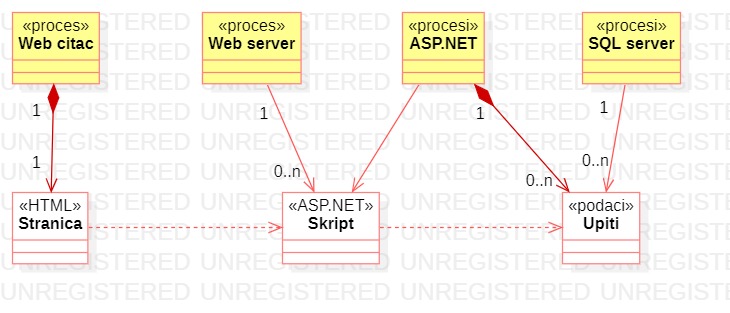
U ovom odeljku je sadržan pogled na procesnu arhitekturu sistema. Ovaj opis treba da sadrži specifikaciju različitih zadataka (procesa i niti) uključenih u rad sistema. Takođe je potrebno dati dijagrame koji pokazuju njihovu interakciju i konfiguraciju. Dodela objekata i klasa na određene zadatke takođe spada u opis procesne arhitekture.

Web aplikacije zasnovane na ASP.NET-u imaju relativno jednostavan procesni model koji je u potpunosti pod kontrolom Web servera. Sa stanovišta projektanta ASP.NET Web aplikacije nije potrebno voditi računa o načinu rada Web servera i načinu izvršavanja skripti.

Ilustracije radi u nastavku je dat opis procesa uključenih u izvršenje StartUp portala kao Web aplikacije.

## 8.1 Procesi

Na sledećem UML dijagramu klasa prikazani su procesi koji učestvuju u izvršenju StartUp portala. Dijagram je opšteg tipa i može se primeniti na bilo koju Web aplikaciju zasnovanu na ASP.NET-u i SQL bazi podataka.



### 8.1.1 Web čitač

Web čitač je proces koji izvršava funkcionalnost aplikacije za prikaz HTML stranica dobijenih od nekog Web servera. U najopštijem slučaju Web čitač u jednom trenutku može da prikazuje samo jednu HTML stranicu.

Web čitač zavisi od Web servera koji generiše i vraća odgovarajuću HTML stranicu na zahtev.

### 8.1.2 Web server

Web server je proces koji izvršava funkcionalnost opsluživanja zahteva prispelih sa više Web čitača

### 8.1.3 MySQL Server

SQL Server je proces koji izvršava funkcionalnost SQL sistema za upravljanje bazama podataka. Ovaj proces može konkurentno da prihvati određen broj upita, izvrši ih nad bazom podataka i vrati rezultate procesu koji je upite postavio.

# 9. Pogled na raspoređivanje sistema

Pogled na raspoređivanje sistema prikazuje različite fizičke čvorove za najopštiju konfiguraciju sistema. Fizičkim čvorovima koji predstavljaju procesore vrši se dodeljivanje identifikovanih procesa.

Na sledećoj slici dat je UML dijagram raspoređivanja StartUp veb aplikacije.

Letter

Description automatically generated with low confidence

## 9.1 Klijent

Pristup StartUp veb aplikaciji se obavlja preko klijentskih računara na kojima se izvršava Veb čitač. Za povezivanje između klijenta i Veb servera koristi se Internet infrastruktura tako da nema ograničenja u pogledu lokacije klijenta.

## 9.2 Veb server

Računar na kome se izvršava Veb server opslužuje više klijenata koji pristupaju preko Interneta.U najopštioj konfiguraciji DBMS se izvršava na posebnoj mašini koja je sa Veb serverom u lokalnoj mreži (LAN).

## 9.3 DBMS server

DBMS server je računar na kome se izvršava SQL server proces koji realizuje funkcionalnost sistema za upravljanje bazama podataka. Zbog sigurnosti podataka koji se na ovom računaru čuvaju pristup bazi je ograničen samo na računare iz lokalne mreže (LAN).

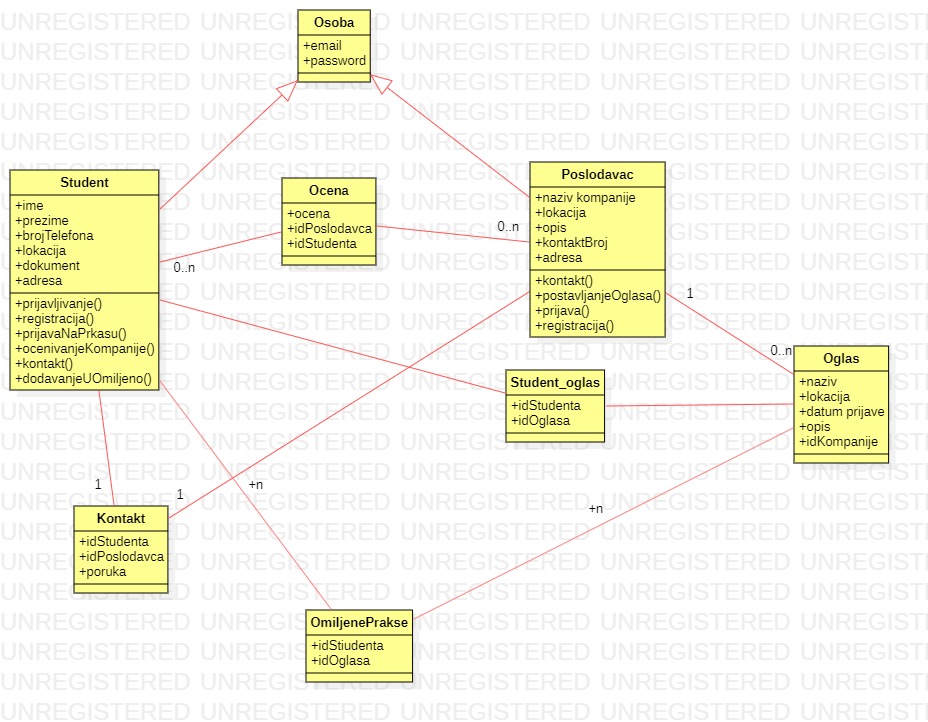
# 10. Pogled na implementaciju sistema

Pogled na implementaciju prikazuje različite aspekte bitne za implementaciju sistema. U slučaju Bookshelf veb aplikacije ovaj odeljak sadrži model domena, šemu baze podataka i prikaz komponenti sistema razvrstanih u ranije identifikovane pakete.

## 10.1 Model domena

Model domena za koji se StartUp veb aplikacija projektuje je ilustrovan UML dijagramom klasa. U njemu su prikazane domenske klase, neki od njihovih atributa, kao i veze koje se mogu identifikovati između njih.

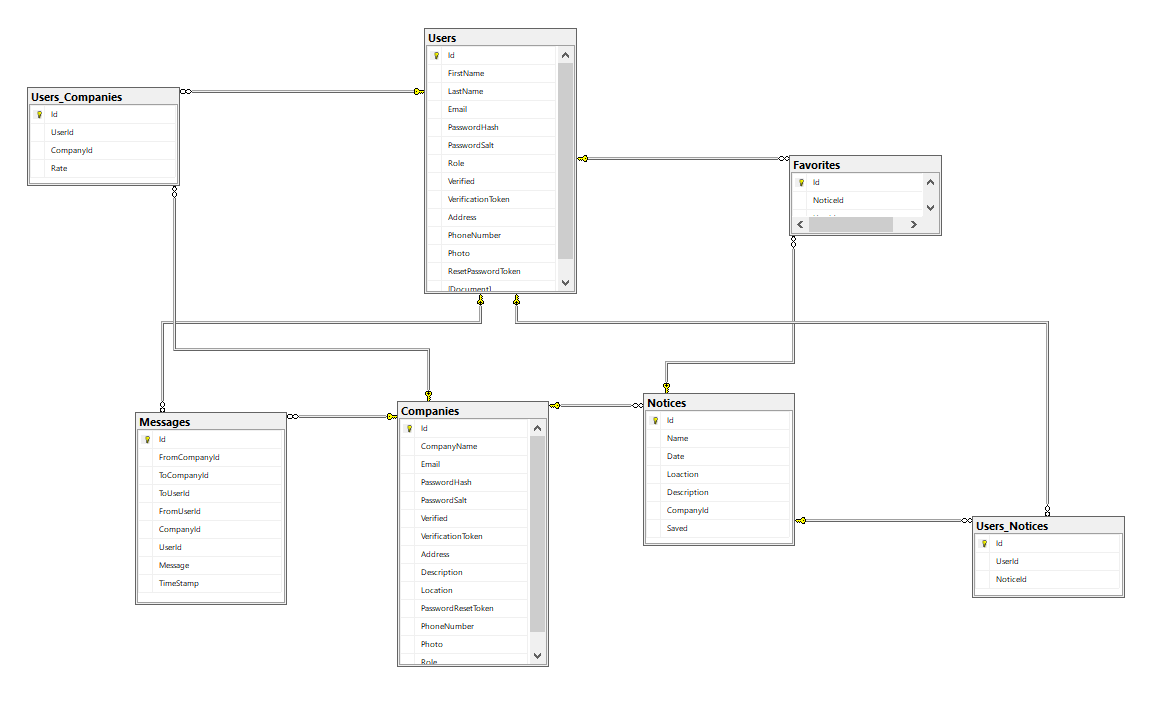
Model domena predstavlja osnovu za projektovanje baze podataka, ali i identifikaciju nekih od komponenti (ASP.NET kontrolera) koje će biti implementirane.



## 

## 10.2 Šema baze podataka

Detaljna šema SQL baze podataka je prikazana na sledećem dijagramu. Dijagram je kreiran korišćenjem aplikacije Microsoft SQL Server.



## 10.3 Komponente sistema

Komponente sistema StartUp veb aplikacije su React komponente čiji će pregled biti dat po arhitekturnim slojevima. Za ilustraciju će biti korišćeni UML dijagrami komponenti, ali i dijagrami klasa. U slučajevima gde je komponenta prikazana kao klasa atributi predstavljaju ulazne podatke koji se uzimaju iz GET ili POST dela HTTP poruke, dok metodi predstavljaju funkcije definisane u okviru skripta.

### 10.3.1. Komponente korisničkog interfejsa

Dizajn korisničkog interfejsa je obuhvaćen dvema komponentama:

A picture containing text

Description automatically generated

Komponenta **index.jsx** je implementira stranicu aplikacije čiji sadržaj može da varira od parametara koji joj se proslede u zahtevu.

Komponenta **style.css** predstavlja opis stilova za pojedine HTML elemente koji se javljaju na različitim stranicama.

### 10.3.2 Komponente aplikacione logike

Komponente koje realizuju domen problema se uključuju isključivo preko **index.jsx** komponente korisničkog interfejsa. Na taj način zadržavaju sva podešavanja stila definisana u ovom skriptu. Na sledećem dijagramu su prikazane komponente ovog sloja i njihove međusobne zavisnosti:

Diagram

Description automatically generated

### 10.3.3 Komponente za pristup podacima

Pristup bazi podataka je u potpunosti zatvoren u funkcije koje su definisane u okviru ASP.NET skripta **backend.cs**. Pomenuti skript se uključuje na početku index.jsx-a, tako da su sve funkcije za pristup podacima dostupne svim komponentama.

A picture containing text

Description automatically generated

# 11. Performanse

Izabrana arhitektura softvera podržava zahteve u pogledu broja korisnika koji mogu simultano pristupati sistemu i vremena odziva za pristup bazi podataka specificirane u zahtevima u pogledu performansi [5]:

1. Sistem će da podrži do 100 simultanih pristupa korisnika veb aplikacije.
2. Vreme potrebno za pristupanje bazi podataka u cilju izvršenje nekog upita ne sme da bude veće od 5 sekundi.

Zahtevane performanse su zadovoljene izborom tehnologija na kojima će sistem biti razvijen i definisane hardverske platforme [5].

# 12. Kvalitet

Izabrana arhitektura softvera podržava zahteve u pogledu dostupnosti i srednjeg vremena između otkaza specificirane u zahtevima u pogledu pouzdanosti [5]:

1. Bookshelf veb aplikacija će biti dostupna 24 časa dnevno, 7 dana u nedelji. Vreme kada veb aplikacija nije dostupna ne sme da pređe 10%.
2. Srednje vreme između dva sukcesivna otkaza ne sme da padne ispod 120 sati.