ASSC

Aplikacija Studentskog Sportskog Centra

Arhitekturni projekat

Verzija 1.0

Pregled izmena

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verzija** | **Opis** | **Autor** |
| 18.04.2023. | 1.0 | Inicijalna verzija | Anđela, Anastasija |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sadržaj

1. Cilj dokumenta 5

2. Opseg dokumenta 5

3. Reference 5

4. Predstavljanje arhitekture 5

5. Ciljevi i ograničenja arhitekture 5

6. Pogled na slučajeve korišćenja 5

6.1 Dijagrami slučajeva korišćenja 6

6.2 Kratak opis slučajeva korišćenja 8

6.2.1 Pregled informacija o SSC-u 8

6.2.2 Ažuriranje informacija o SSC-u 8

6.2.3 Pregled informacija o sportovima 8

6.2.4 Pregled spiska takmičenja 8

6.2.5 Pregled spiska studenata 9

6.2.6 Pregled podataka o određenom studentu 9

6.2.7 Pregled spiska sportskih trenera 9

6.2.8 Pregled podataka o određenom sportskom treneru 9

6.2.9 Dodavanje novog termina treninga 9

6.2.10 Dodavanje novog takmičenja 9

6.2.11 Ažuriranje podataka o studentu 9

6.2.12 Ažuriranje podataka o sportskom treneru 9

6.2.13 Ažuriranje podataka o takmičenju 9

6.2.14 Ažuriranje termina treninga 9

6.2.15 Promena sporta 10

6.2.16 Prijavljivanje 10

6.2.17 Kreiranje korisničkog naloga 10

6.2.18 Brisanje korisničkog naloga 10

6.2.19 Evidencija o plaćenim mesečnim članarinama 10

6.2.20 Obaveštenja o kašnjenju 10

6.2.21 Zakazivanje termina treninga 10

6.2.22 Otkazivanje termina treninga 10

6.2.23 Pregled informacija o takmičenjima 11

6.2.24 Fakultet na kome se takmiče studenti 11

6.2.25 Ukupan broj takmičara 11

6.2.26 Prijavljivanje na takmičenja 11

6.2.27 Pregled informacija o timovima 11

6.2.28 Formiranje tima 11

7. Pogled na logičku arhitekturu sistema 11

7.1 Pregled arhitekture – organizacija paketa i podsistema u slojeve 12

7.1.1 Korisnički interfejs 12

7.1.2 Aplikaciona logika 12

7.1.3 Pristup podacima 12

7.1.4 React 12

7.1.5 .NET 12

7.1.6 SQL Server 13

8. Pogled na procese 13

8.1 Procesi 13

8.1.1 Web čitač 13

8.1.2 Web server 13

8.1.3 .NET 13

8.1.4 SQL Server 13

9. Pogled na raspoređivanje sistema 14

9.1 Klijent 14

9.2 Web server 14

9.3 DBMS server 14

10. Pogled na implementaciju sistema 14

10.1 Model domena 14

10.2 Šema baze podataka 15

10.3 Komponente sistema 16

10.3.1 Komponente korisničkog interfejsa 16

10.3.2 Komponente aplikacione logike na serverskoj strani 17

10.3.2 Komponente aplikacione logike na klijentskoj strani 17

10.3.3 Komponente za pristup podacima 17

11. Performanse 18

12. Kvalitet 18

Arhitekturni projekat

# Cilj dokumenta

Cilj ovog dokumenta je detaljni opis arhitekture ASSC aplikacije.

# Opseg dokumenta

Dokument se odnosi na ASSC aplikaciju koja će biti razvijena od strane A&A tima. ASSC predstavlja skraćenicu za aplikaciju Studentskog Sportskog Centra. Namena sistema je efikasno, brzo i jednostavno pronalaženje, zakazivanje, prijavljivanje i održavanje sadržaja vezanih za sportske treninge i takmičenja.

# Reference

Spisak korišćene literature:

1. ASSC – Predlog projekta, SWE-ASSC-01, V1.0, 2023, A&A.
2. ASSC – Planirani raspored aktivnosti na projektu, V1.0, 2023, A&A.
3. ASSC – Plan realizacije projekta, V1.0, 2023, A&A.
4. ASSC – Vizija sistema, V1.0, 2023, A&A.
5. ASSC – Specifikacija zahteva, V1.0, 2023, A&A.

# Predstavljanje arhitekture

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

# Ciljevi i ograničenja arhitekture

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

1. ASSC aplikacija će biti implementirana kao Web aplikacija zasnovana na .NET-u, JavaScript jeziku i SQL bazi podataka.
2. Klijentski deo ASSC aplikacije će biti optimizovan za sledeće Web čitače: Google Chrome, Microsoft Edge, Mozilla Firefox. [4].
3. Svi zahtevi u pogledu performansi dati u [5] moraju biti uzeti u obzir pri izboru arhitekture i razvoju sistema.

# Pogled na slučajeve korišćenja

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

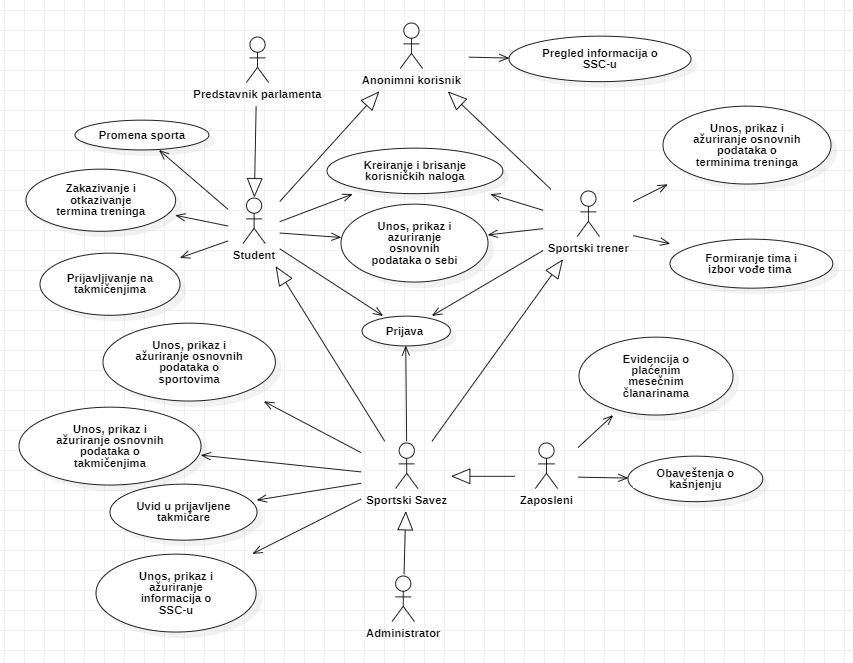
Slučajevi korišćenja ASSC aplikacije su:

* *Pregled informacija*
  + Pregled informacija o SSC-u
  + Pregled informacija o sportovima
  + Pregled spiska takmičenja
  + *Pregled informacija o studentima*
    - Pregled spiska studenata
    - Pregled podataka o određenom studentu
  + *Pregled informacija o trenerima*
    - Pregled spiska sportskih trenera
    - Pregled podataka o određenom sportskom treneru
* Ažuriranje informacija o SSC-u
* Ažuriranje podataka o studentu
  + Promena sporta
* Ažuriranje podataka o sportskom treneru
* Ažuriranje podataka o takmičenju
  + Dodavanje novog takmičenja
* Ažuriranje termina treninga
  + Dodavanje novog termina treninga
* Prijavljivanje
* *Kreiranje brisanje i arhiviranje članova*
  + Kreiranje korisničkog naloga
  + Brisanje korisničkog naloga
* Evidencija o plaćenim mesečnim članarinama
  + Obaveštenja o kašnjenju
* *Raspored treninga*
  + Zakazivanje termina treninga
  + Otkazivanje termina treninga
* Pregled informacija o takmičenjima
  + Fakultet na kome se takmiče studenti
  + Ukupan broj takmičara
  + Prijavljivanje na takmičenja
  + Pregled informacija o timovima
* Formiranje tima

Ove slučajeve korišćenja mogu da iniciraju anoniman korisnik, student, predstavnik parlamenta za sport, sportski trener, Sportski Savez, zaposleni ili administrator.

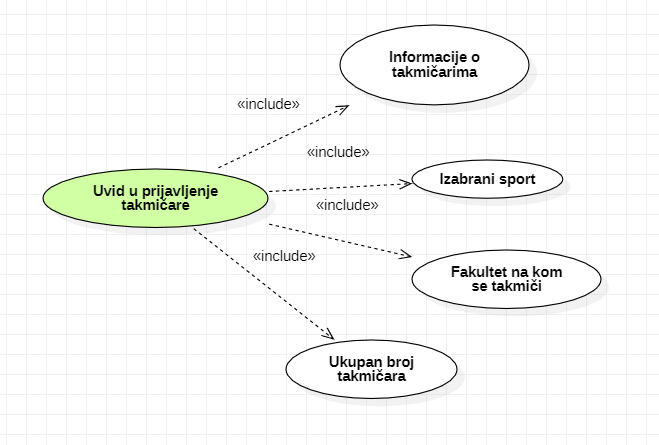
## Dijagrami slučajeva korišćenja

Osnovni UML dijagram koji prikazuje korisnike i slučajeve korišćenja ASSC aplikacije prikazan je na sledećoj slici:

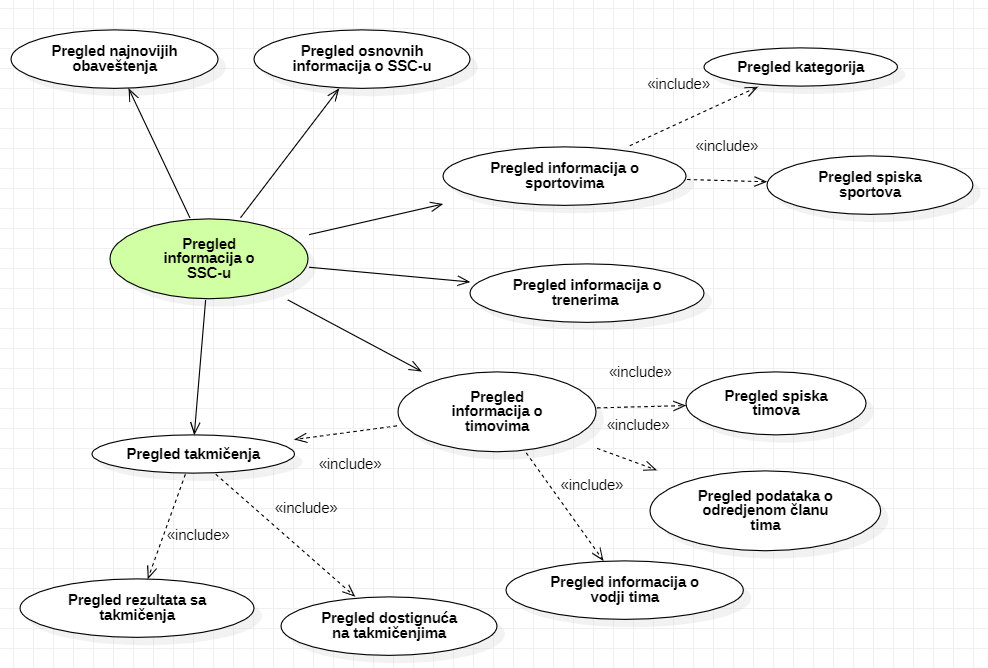


Slučajevi korišćenja *uvid u prijavljene takmičare, pregled informacija o SSC-u i kreiranje i brisanje korisničkih naloga* obuhvataju složenije radnje koje se mogu razložiti dalje na pojedinačne slučajeve korišćenja.

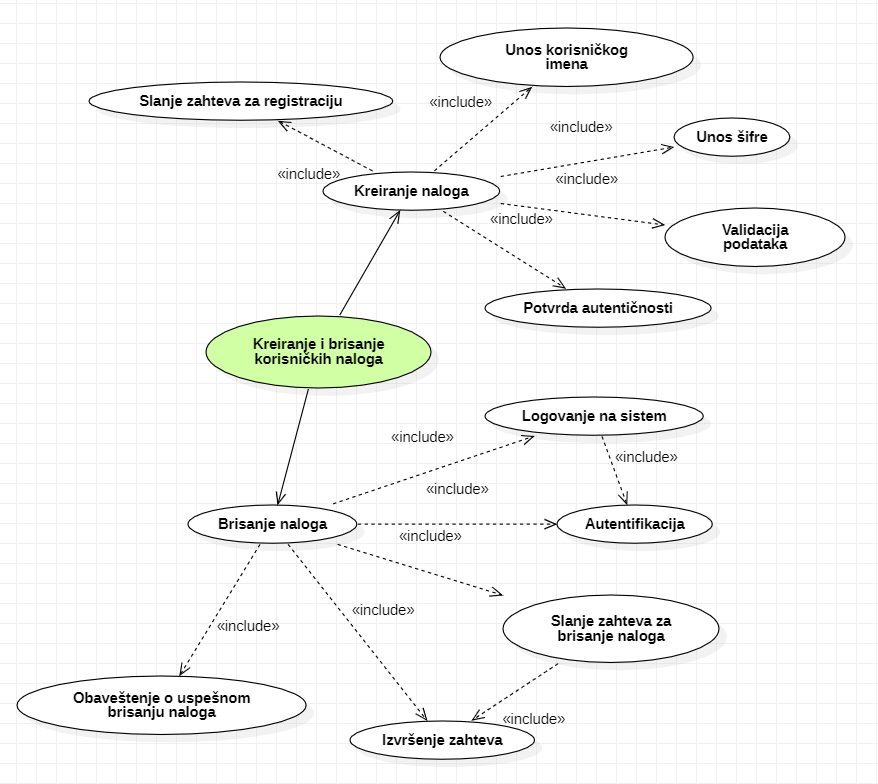
Detaljni UML dijagram za slučaj korišćenja *uvid u prijavljene takmičare* je prikazan na sledećoj slici:



Detaljni UML dijagram za slučaj korišćenja *pregled informacija o SSC-u* je prikazan na sledećoj slici:



Detaljni UML dijagram za slučaj korišćenja *kreiranje i brosanje korisničkih naloga* je prikazan na sledećoj slici:



## Kratak opis slučajeva korišćenja

### Pregled informacija o SSC-u

Kratak opis:Prikaz stranica sa informacijama Studentskog Sportskog Centra.

Akteri koji iniciraju slučaj korišćenja:Anoniman korisnik, Student, Predstavnik parlamenta za sport, Sportski trener, Sportski Savez Srbije, Zaposleni, Administrator.

### Ažuriranje informacija o SSC-u

Kratak opis: Ažuriranje informacija o Studentskom Sportskom Centru.

Akteri koji iniciraju slučaj korišćenja: Sportski Savez Srbije, Administrator.

### Pregled informacija o sportovima

Kratak opis: Prikaz stranice sa informacijama o sportovima koji pripadaju Studentskom Sportskom Centru.

Akteri koji iniciraju slučaj korišćenja: Anonimni korisnik, Student, Predstavnik parlamenta ya sport, Sportski trener, Sportski Savez, Administrator.

### Pregled spiska takmičenja

Kratak opis: Prikaz stranica sa spiskom takmičenja koji se realizuju u okviru Studentskog Sportskog Centra.

Akteri koji iniciraju slučaj korišćenja: Anoniman korisnik, Student, Predstavnik parlamenta za sport, Sportski trener, Sportski Savez Srbije, Zaposleni, Administrator.

### Pregled spiska studenata

Kratak opis: Prikaz stranice aplikacije sa spiskom studenata Studentskog Sportskog Centra.

Akteri koji iniciraju slučaj korišćenja: Student, Predstavnik parlamenta za sport, Sportski Savez Srbije, Zaposleni, Administrator.

### Pregled podataka o određenom studentu

Kratak opis: Prikaz stranice sa informacijama o određenom studentu.

Akteri koji iniciraju slučaj korišćenja: Student, Predstavnik parlamenta za sport, Sportski trener, Sportski Savez Srbije, Zaposleni, Administrator.

### Pregled spiska sportskih trenera

Kratak opis: Prikaz stranice aplikacije sa spiskom sportskih trenera Studentskog Sportskog Centra.

Akteri koji iniciraju slučaj korišćenja: Student, Predstavnik parlamenta za sport, Sportski Savez Srbije, Zaposleni, Administrator.

### Pregled podataka o određenom sportskom treneru

Kratak opis: Prikaz stranice sa informacijama o određenom sportskom treneru.

Akteri koji iniciraju slučaj korišćenja: Student, Predstavnik parlamenta za sport, Sportski trener, Sportski Savez Srbije, Zaposleni, Administrator.

### Dodavanje novog termina treninga

Kratak opis: Dodavanje novog termina treninga od strane sportskog trenera.

Akteri koji iniciraju slučaj korišćenja: Sportski trener, Administrator.

### Dodavanje novog takmičenja

Kratak opis: Dodavanje novog takmičenja od strane Sportskog Saveza Srbije koji je organizator svih održanih takmičenja u Studentskom Sportskom Centru.

Akteri koji iniciraju slučaj korišćenja: Sportski Savez, Administrator.

### Ažuriranje podataka o studentu

Kratak opis: Ažuriranje podataka o sebi od strane prijavljenog studenta koji pripada Studentskom Sportskom Centru.

Akteri koji iniciraju slučaj korišćenja: Student, Predstavnik parlamenta za sport, Administrator.

### Ažuriranje podataka o sportskom treneru

Kratak opis: Ažuriranje podataka o sebi od strane prijavljenog sportskog trenera koji pripada Studentskom Sportskom Centru.

Akteri koji iniciraju slučaj korišćenja: Sportski trener, Administrator.

### Ažuriranje podataka o takmičenju

Kratak opis: Izmena podataka o takmičenju od strane Sportskog Saveza Srbije.

Akteri koji iniciraju slučaj korišćenja: Sportski Savez Srbije, Administrator.

### Ažuriranje termina treninga

Kratak opis: Izmena podataka o terminu treninga od strane sportskog trenera.

Akteri koji iniciraju slučaj korišćenja: Sportski trener, Administrator.

### Promena sporta

Kratak opis: Student vrše promenu sporta.

Akteri koji iniciraju slučaj korišćenja: Student, Administrator.

### Prijavljivanje

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

Akteri koji iniciraju slučaj korišćenja: Student, Predstavnik parlamenta za sport, Sportski trener, Sportski Savez, Administrator.

### Kreiranje korisničkog naloga

Kratak opis: Kreiranje korisničkog naloga za novog člana SSC-a.

Akteri koji iniciraju slučaj korišćenja: Sportski Savez, Administrator.

### Brisanje korisničkog naloga

Kratak opis: Brisanje korisničkog naloga i podataka za postojećeg člana SSC-a.

Akteri koji iniciraju slučaj korišćenja: Sportski Savez, Administrator.

### Evidencija o plaćenim mesečnim članarinama

Kratak opis: Vodi se evidencija o plaćenim mesečnim članarinama.

Akteri koji iniciraju slučaj korišćenja: Zaposleni, Administrator.

### Obaveštenja o kašnjenju

Kratak opis: Obaveštenje o neplaćenim članarinama.

Akteri koji iniciraju slučaj korišćenja: Zaposleni, Administrator.

### Zakazivanje termina treninga

Kratak opis: Student zakazuje termin treninga.

Akteri koji iniciraju slučaj korišćenja: Student, Administrator.

### Otkazivanje termina treninga

Kratak opis: Student otkazuje termin treninga.

Akteri koji iniciraju slučaj korišćenja: Student, Administrator.

### Pregled informacija o takmičenjima

Kratak opis: Prikaz informacija o takmičenjima na koja su se prijavili takmičari, odnosno studenti.

Akteri koji iniciraju slučaj korišćenja: Anoniman korisnik, Student, Predstavnik parlamenta za sport, Sportski trener, Sportski Savez Srbije, Administrator.

### Fakultet na kome se takmiče studenti

Kratak opis: Prikaz informacija o fakultetu na kom se takmiče prijavljeni takmičari.

Akteri koji iniciraju slučaj korišćenja: Sportski Savez Srbije, Administrator.

### Ukupan broj takmičara

Kratak opis: Prikaz ukupnog broja takmičara na takmičenju.

Akteri koji iniciraju slučaj korišćenja: Anoniman korisnik, Student, Predstavnik parlamenta za sport, Sportski trener, Sportski Savez Srbije, Administrator.

### Prijavljivanje na takmičenja

Kratak opis: Student se prijavljuje za učešće na željeno takmičenje.

Akteri koji iniciraju slučaj korišćenja: Student, Administrator.

### Pregled informacija o timovima

Kratak opis: Prikaz stranice sa informacijama o timovima po fakultetima.

Akteri koji iniciraju slučaj korišćenja: Anonimni korisnik, Student, Predstavnik parlamenta za sport, Trener, Sportski Savez, Administrator.

### Formiranje tima

Kratak opis: Trener formira tim.

Akteri koji iniciraju slučaj korišćenja: Trener, Administrator.

# 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 ASSC aplikaciju obuhvata 3 glavna paketa: Korisnički interfejs, Aplikaciona logika, Pristup podacima.

Paket *Korisnički interfejs* sadrži Single Page React Aplikaciju 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 .NET klase zadužene za realizaciju funkcionalnosti specifičnih za domen sistema koji se razvija.

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

## Pregled arhitekture – organizacija paketa i podsistema u slojeve

Diagram

Description automatically generated

### Korisnički interfejs

sloj

Ovaj sloj realizuje korisnički interfejs aplikacije. On sadrži Single Page React Aplikaciju, koja komunicira sa serverskom stranom koja je bazirana na .NET klasama.

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

### Aplikaciona logika

sloj

Sloj aplikacione logike je srednji sloj u troslojnoj arhitekturi ASSC aplikacije. Sadrži .NET klase koje realizuju funkcionalnost karakterističnu za domen primene platforme i uspostavljaju vezu između korisničkog interfejsa i sloja za pristup podacima.

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

### Pristup podacima

sloj

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

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

### React

tehnologija

React tehnologija omogućava korišćenje gotovih komponenti koje se prikazuju u Web čitaču i služe za prikaz formatiranih informacija i realizaciju formi za unos i ažuriranje podataka.

### .NET

tehnologija

Tehnologija .NET obezbeđuje mehanizam za pisanje i izvršavanje serverske strane aplikacije.

### SQL Server

DBMS

SQL Server predstavlja sistem za upravljanje bazama podataka koji će se koristiti za realizaciju ASSC aplikacije.

# 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.

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

## Procesi

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

Diagram

Description automatically generated

### Web čitač

Web čitač je proces koji izvršava funkcionalnost aplikacije za prikaz React SPA dobijene od strane Web servera. Web čitač u jednom trenutku prikazuje React SPA.

Web čitač zavisi od Web servera koji generiše i vraća odgovarajuću React SPA na zahtev.

### Web server

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

### .NET

.NET proces obavlja posao obrade zadatog .NET kontrolerima i generiše odgovaraće odgovore na klijentske zahteve pristigle od strane React SPA. .NET proces može da zahteva usluge SQL servera. Komunikacija između .NET procesa i SQL servera se obavlja preko prosleđivanja upita i vraćanja rezultata.

### SQL 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.

# 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 ASSC aplikacije.

Diagram

Description automatically generated

## Klijent

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

## Web server

Računar na kome se izvršava Web server opslužuje više klijenata koji pristupaju preko Interneta. Pored osnovnog procesa koji realizuje funkcionalnost Web servera, na ovom računaru mogu da se izvršavaju i procesi .NET koji izvršavaju funkcionalnosti kontrolera. U najopštijoj konfiguraciji DBMS se izvršava na posebnoj mašini koja je sa Web serverom u lokalnoj mreži (LAN).

## 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).

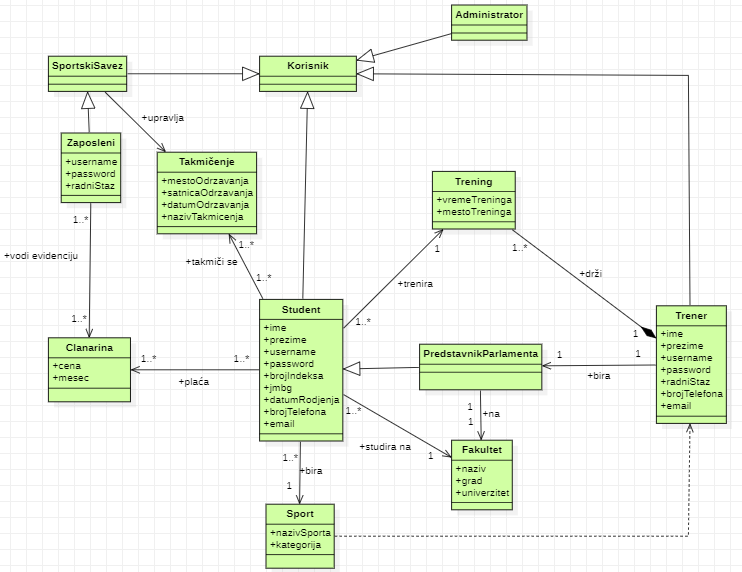
# Pogled na implementaciju sistema

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

## Model domena

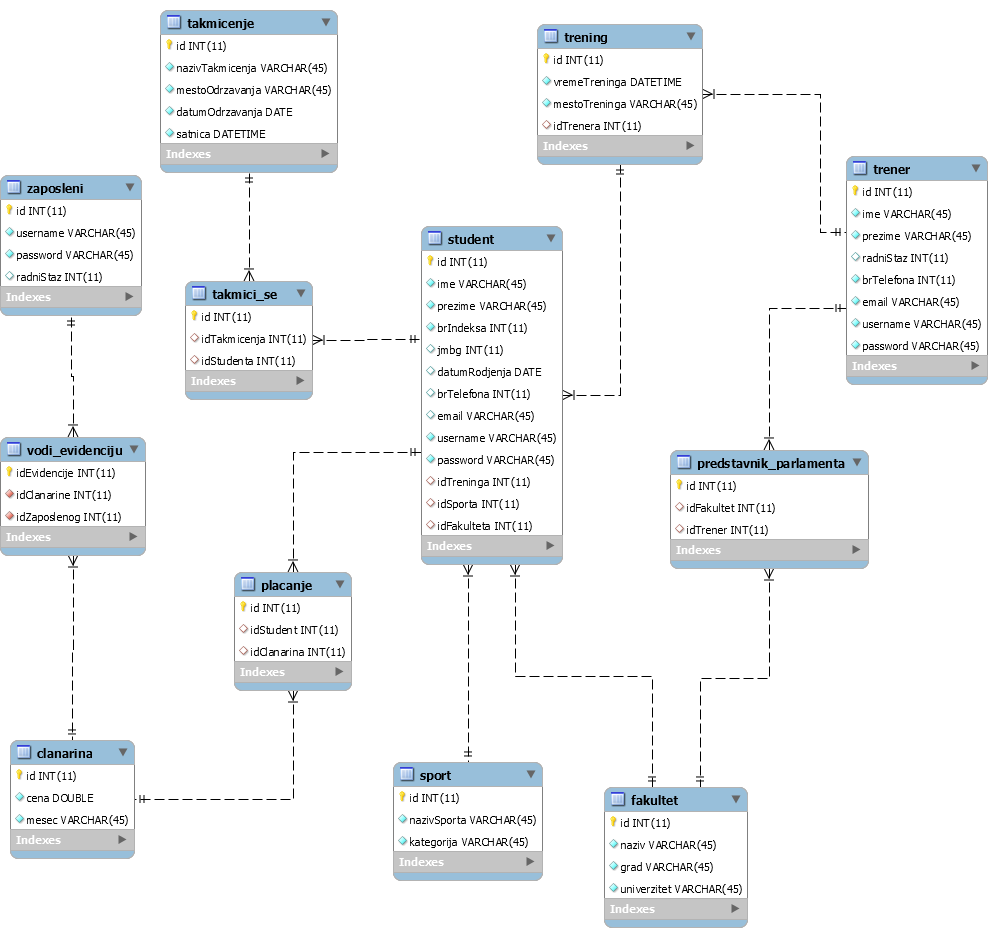
Model domena za koji se ASSC aplikacija projektuje je ilustrovan UML dijagramom klasa. U njemu su prikazane domenske klase, neki od njihovih atributa, kao i veze koje se mogu identifikovati između njih.

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



## Šema baze podataka

Detaljna šema baze podataka je prikazana na sledećem dijagramu. Baza podataka i migracija na SQL Server su obavljeni korišćenjem *EntityFramework* ORM alata, dok je dijagram generisan pomoću *MySQL Workbench*-a.

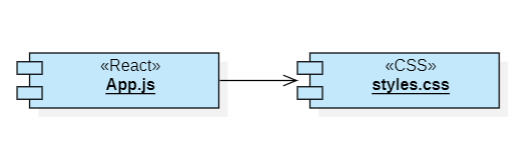


## Komponente sistema

Komponente aplikacije ASSC su React komponente na klijentskoj strani, a API kontroleri na serverskoj strani čiji će pregled biti dat po arhitekturnim slojevima. Za ilustraciju će biti korišćeni UML dijagrami komponenti, ali i dijagrami klasa.

### Komponente korisničkog interfejsa

Dizajn korisničkog interfejsa je obuhvaćen dvema komponentama:

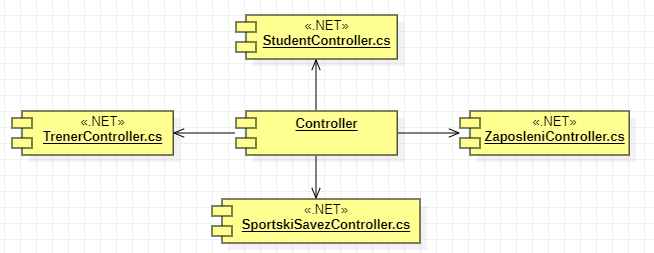


Komponenta **App.js** implementira stranicu aplikacije čiji sadržaj može da varira od parametra koji joj se proslede u zahtevu.

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

### Komponente aplikacione logike na serverskoj strani

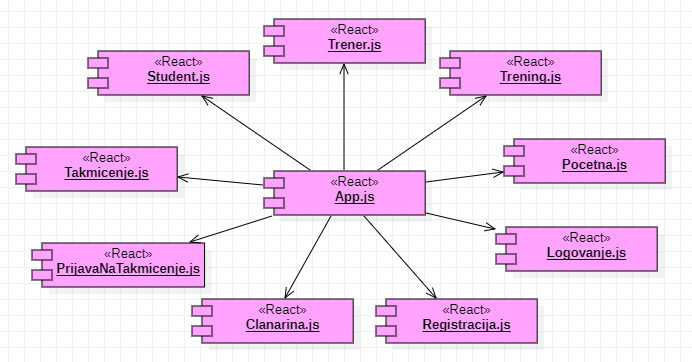
Komponente na serverskoj strani su API kontroleri. *Controller* sistem u okviru .NET odlučuje o tome koji će kontroler biti pozvan na osnovu zahteva klijenta. Sva logika aplikacije je smeštena u okviru kontrolera – pristup bazi podataka i slanje odgovora klijentu na osnovu obrade zahteva.



Svaka komponenta je zadužena za obradu određenih zahteva, a na osnovu imena može se videti grupa zahteva koje komponenta obrađuje.

### Komponente aplikacione logike na klijentskoj strani

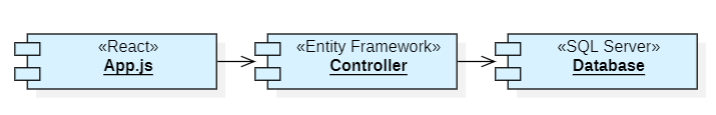
### Komponente koje realizuju domen problema se uključuju isključivo preko **App.js** komponente korisničkog interfejsa. Na taj način zadržavaju sva podešavanja stila definisana u ovom skriptu. Na sledećem dijagramu su prikazane komponente ovog sloja i njihove međusobne zavisnosti:



### Komponente za pristup podacima

Pristup bazi podataka se vrši pomoću *EntityFramework*-a. Upiti se ne izvršavaju direktno, već korišćenjem ORM-a.

Na sledećem UML dijagramu klasa pobrojane su funkcije za pristup podacima iz baze:



# 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 5000 simultanih pristupa korisnika portalu.
2. Vreme potrebno za pristupanje bazi podataka u cilju izvršenje nekog upita ne sme da bude veće od 3.4 sekunde.

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

# Kvalitet

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

1. ASSC aplikacija će biti dostupna 24 časa dnevno, 7 dana u nedelji. Vreme kada aplikacija nije dostupna ne sme da pređe 5%.
2. Srednje vreme između dva sukcesivna otkaza ne sme da padne ispod 60 sati.