Flower Power

Arhitekturni projekat

Verzija 1.0

Pregled izmena

| **Datum** | **Verzija** | **Opis** | **Autor** |
| --- | --- | --- | --- |
| 19.04.2023. | 1.0 | Inicijalna verzija | Kristina, Vladimir, Milena |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sadržaj

[**1. Cilj dokumenta 5**](#_heading=h.gjdgxs)

[**2. Opseg dokumenta 5**](#_heading=h.30j0zll)

[**3. Reference 5**](#_heading=h.1fob9te)

[**4. Predstavljanje arhitekture 5**](#_heading=h.2et92p0)

[**5. Ciljevi i ograničenja arhitekture 5**](#_heading=h.tyjcwt)

[**6. Pogled na slučajeve korišćenja 5**](#_heading=h.3dy6vkm)

[6.1 Dijagrami slučajeva korišćenja 6](#_heading=h.2bn6wsx)

[6.2 Kratak opis slučajeva korišćenja 9](#_heading=h.qsh70q)

[6.2.1 Komunikacija između cvećare i kupca 9](#_heading=h.3as4poj)

[6.2.2 Specijalna ponuda 9](#_heading=h.kajxo4yda880)

[6.2.3 Dodavanje u korpu 9](#_heading=h.1pxezwc)

[6.2.4 Potvrda kupovine 9](#_heading=h.49x2ik5)

[6.2.5 Tip plaćanja i završetak kupovine 9](#_heading=h.2p2csry)

[6.2.6 Plaćanje platnim karticama 9](#_heading=h.147n2zr)

[6.2.7 Registracija administratora 9](#_heading=h.3o7alnk)

[6.2.8 Registracija kupca 9](#_heading=h.23ckvvd)

[6.2.9 Registracija cvećare 9](#_heading=h.ihv636)

[6.2.10 Prijavljivanje cvećare 9](#_heading=h.32hioqz)

[6.2.11 Prijavljivanje kupca i administratora 9](#_heading=h.1hmsyys)

[6.2.12 Unos i ažuriranje podataka cvećare 9](#_heading=h.41mghml)

[6.2.13 Unos i ažuriranje podataka kupca 9](#_heading=h.2grqrue)

[6.2.14 Unos i ažuriranje podataka asortimana 9](#_heading=h.vx1227)

[6.2.15 Zahtev za validiranje 10](#_heading=h.3fwokq0)

[6.2.16 Validacija 10](#_heading=h.1v1yuxt)

[6.2.17 Žalba 10](#_heading=h.4f1mdlm)

[6.2.18 Revidiranje žalbe 10](#_heading=h.2u6wntf)

[6.2.19 Suspendovanje korisnika 10](#_heading=h.19c6y18)

[6.2.20 Brisanje korisnika 10](#_heading=h.3tbugp1)

[6.2.21 Komunikacija između cvećare i kupca 10](#_heading=h.yqul03orsw26)

[6.2.22 Ostavljanje i prikaz recenzija 10](#_heading=h.urx0zd6qu1ra)

[6.2.23 Prikaz cvećara 10](#_heading=h.lnflx9tzr1yw)

[6.2.24 Prikaz asortimana 10](#_heading=h.th05ssakpap6)

[6.2.25 Prikaz konkretnog asortimana 10](#_heading=h.vidpknajnl4)

[6.2.26 Filtriranje cvećare i proizvoda 10](#_heading=h.rf6kfcmp5b4f)

[**7. Pogled na logičku arhitekturu sistema 10**](#_heading=h.28h4qwu)

[7.1 Pregled arhitekture – organizacija paketa i podsistema u slojeve 11](#_heading=h.nmf14n)

[7.1.1 Korisnički interfejs 11](#_heading=h.37m2jsg)

[7.1.2 Aplikaciona logika 11](#_heading=h.1mrcu09)

[7.1.3 Pristup podacima 11](#_heading=h.46r0co2)

[7.1.4 ReactJS 11](#_heading=h.2lwamvv)

[7.1.5 ExpressJS 12](#_heading=h.111kx3o)

[7.1.6 MongoDB 12](#_heading=h.3l18frh)

[**8. Pogled na procese 12**](#_heading=h.206ipza)

[8.1 Procesi 12](#_heading=h.4k668n3)

[8.1.1 Web browser 12](#_heading=h.2zbgiuw)

[8.1.2 Web server 12](#_heading=h.1egqt2p)

[8.1.3 Node.js RESTful API 12](#_heading=h.3ygebqi)

[8.1.4 MongoDB 12](#_heading=h.2dlolyb)

[**9. Pogled na raspoređivanje sistema 13**](#_heading=h.sqyw64)

[9.1 Klijent 13](#_heading=h.3cqmetx)

[9.2 Web server 13](#_heading=h.1rvwp1q)

[9.3 DBMS server 13](#_heading=h.4bvk7pj)

[**10. Pogled na implementaciju sistema 13**](#_heading=h.2r0uhxc)

[10.1 Model domena 13](#_heading=h.1664s55)

[10.2 Šema baze podataka 14](#_heading=h.3q5sasy)

[10.3 Komponente sistema 15](#_heading=h.25b2l0r)

[10.3.1 Komponente korisničkog interfejsa 16](#_heading=h.kgcv8k)

[10.3.2 Komponente aplikacione logike 16](#_heading=h.34g0dwd)

[**11. Performanse 17**](#_heading=h.43ky6rz)

[**12. Kvalitet 17**](#_heading=h.2iq8gzs)

Arhitekturni projekat

# Cilj dokumenta

Cilj ovog dokumenta je detaljni opis arhitekture Flower Power shop-aa.

# Opseg dokumenta

Dokument se odnosi na Flower Power web shop koji će biti razvijen od strane PlantIT-a. Namena sistema je da omogući potencijalnim kupcima da na brz i efikasan način pronađu proizvod cvećare koji im je potreban, a sa druge strane, da cvećare na lak i brz način mogu da predstave svoje proizvode u personalizovanoj prodavnici na web shop-u.

# Reference

Spisak korišćene literature:

1. Flower Power - D01\_Predlog\_Projekta.docx, V1.0, 2023, PlantIT.
2. Flower Power – D02\_Vizija\_Sistema.docx, V1.0, 2023, PlantIT.
3. Flower Power – D03\_Plan\_Realizacije.docx, V1.0, 2023, PlantIT.
4. Flower Power – D03\_Raspored\_Aktivnosti.xml, V1.0, 2023, PlantIT.
5. Flower Power - D04\_Spec\_Zahteva.docx, V1.0, 2023, PlantIT

# 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. Flower Power web shop će biti implementiran kao Web aplikacija zasnovana na Express framework-u Node engine-a i MongoDB bazi podataka [4].
2. Klijentski deo Flower Power web shop će biti optimizovan za sledeće Web čitače: Google Chrome, Opera, Firefox (Mozilla) i Microsoft Edge. [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].

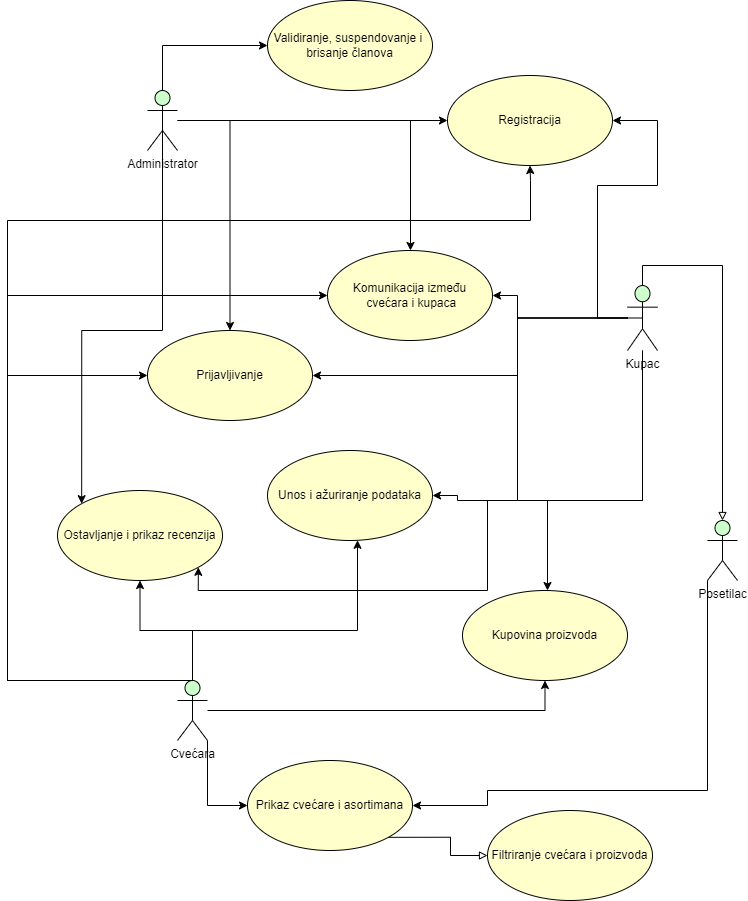
Slučajevi korišćenja PeNcIL portala su:

* Kupovina
  + *Komunikacija između cvećare i kupca*
  + *Specijalna ponuda*
  + *Dodavanje u korpu*
  + *Potvrda kupovine*
  + *Tip plaćanja i završetak kupovine*
  + *Plaćanje platnim karticama*
* Registracija
  + *Registracija administratora*
  + *Registracija kupca*
  + *Registracija cvećare*
* Prijavljivanje
  + *Prijavljivanje cvećare*
  + *Prijavljivanje kupca i administratora*
* Unos i ažuriranje podataka
  + *Unos i ažuriranje podataka cvećare*
  + *Unos i ažuriranje podataka kupca*
  + *Unos i ažuriranje podataka asortimana*
* Validiranje, suspendovanje i brisanje članova
  + *Zahtev za validiranje*
    - *Validacija*
  + *Žalba*
    - *Revidiranje žalbe*
    - *Suspendovanje korisnika*
    - *Brisanje korisnika*
* Prikaz cvećare i asortimana
  + *Prikaz cvećare*
  + *Prikaz asortimana*
    - *Prikaz konkretnog asortimana*
  + *Filtriranje cvećare i asortimana*

Ove slučajevi korišćenja mogu da iniciraju cvećare, kupci, posetioci i administratori.

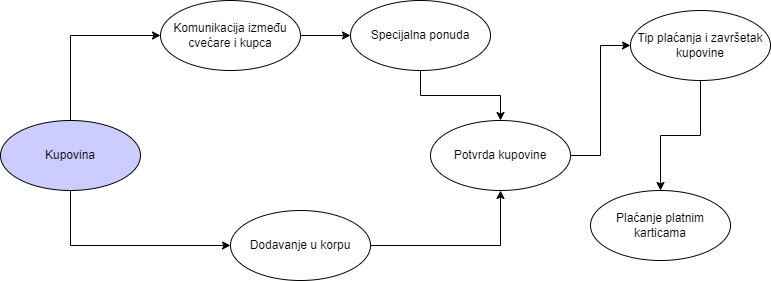
## Dijagrami slučajeva korišćenja

Osnovni UML dijagram koji prikazuje korisnike i slučajeve korišćenja Flower Power web shop-a prikazan je na sledećoj slici:

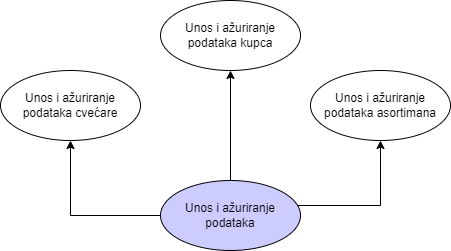


Slučajevi korišćenja *registracija, prijavljivanje, kupovina, validiranje, suspendovanje i brisanje članova, unos i ažuriranje podataka kao i prikaz cvećare i asortimana* obuhvataju složenije radnje koje se mogu razložiti dalje razložiti na pojedinačne slučajeve korišćenja.

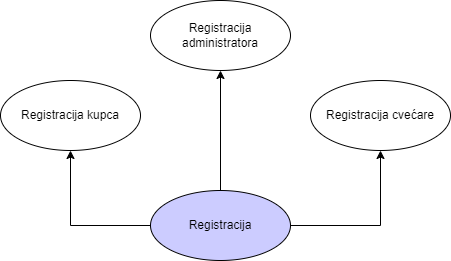
Detaljni UML dijagram za slučaj *kupovine* je prikazan na sledećoj slici:



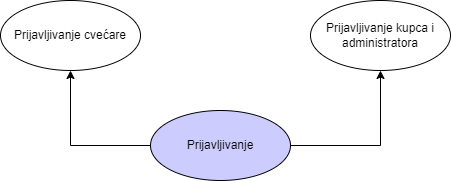
Detaljni UML dijagram za slučaj korišćenja *unos i ažuriranje podataka* je prikazan na sledećoj slici:



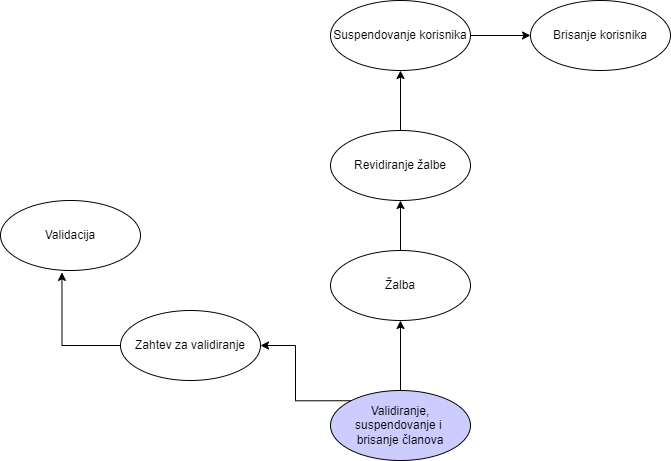
Detaljni UML dijagram za slučaj korišćenja *registracije* je prikazan na sledećoj slici:



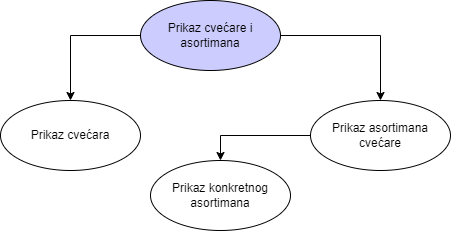
Detaljni UML dijagram za slučaj korišćenja *prijavljivanje* je prikazan na sledećoj slici:



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



Detaljni UML dijagram za slučaj korišćenja *prikaz cvećare i asortimana* je prikazan na sledećoj slici:



## Kratak opis slučajeva korišćenja

### Komunikacija između cvećare i kupca

Kratak opis: Kupac kontaktira cvećaru iz želje za posebnom ponudom, koju cvećara nema konkretno navedenu u shop-u.

Akteri koji iniciraju slučaj korišćenja: Kupac i Cvećara..

### Specijalna ponuda

Kratak opis: Cvećara šalje formalnu ponudu kupcu.

Akteri koji iniciraju slučaj korišćenja: Kupac i Cvećara.

### Dodavanje u korpu

Kratak opis: Kupac dodaje proizvode koje bi želeo da kupi u korpu i ažurira se stanje korpe.

Akteri koji iniciraju slučaj korišćenja: Kupac.

### Potvrda kupovine

Kratak opis: Kupac potvrđuje kupovinu.

Akteri koji iniciraju slučaj korišćenja: Kupac.

### Tip plaćanja i završetak kupovine

Kratak opis: Kupac bira način plaćanja izabranih proizvoda.

Akteri koji iniciraju slučaj korišćenja:Kupac i Cvećara.

### Plaćanje platnim karticama

Kratak opis: Kupac ima mogućnost plaćanja platnom karticom.

Akteri koji iniciraju slučaj korišćenja: Kupac..

### Registracija administratora

Kratak opis: Registrovanje administratora je moguće jedino od strane programera koji održava web shop.

Akteri koji iniciraju slučaj korišćenja: Administrator.

### Registracija kupca

Kratak opis: Popunjavanje forme za registraciju novog kupca.

Akteri koji iniciraju slučaj korišćenja: Kupac.

### Registracija cvećare

Kratak opis: Popunjavanje forme za registraciju nove cvećare i ažuriranje baze podataka.

Akteri koji iniciraju slučaj korišćenja: Cvećara.

### Prijavljivanje cvećare

Kratak opis: Prijavljivanje cvećare na web shop u cilju pristupa specifičnim funkcijama koje zahtevaju autorizaciju.

Akteri koji iniciraju slučaj korišćenja: Cvećara.

### Prijavljivanje kupca i administratora

Kratak opis: Prijavljivanje kupca ili administratora (korisnika) na web shop u cilju pristupa specifičnim funkcijama koje zahtevaju autorizaciju.

Akteri koji iniciraju slučaj korišćenja: Kupac, Administrator.

### Unos i ažuriranje podataka cvećare

Kratak opis: Ažuriranje podataka o sebi od strane prijavljene cvećare.

Akteri koji iniciraju slučaj korišćenja: Cvećara.

### Unos i ažuriranje podataka kupca

Kratak opis: Ažuriranje podataka o sebi od strane prijavljenog kupca.

Akteri koji iniciraju slučaj korišćenja: Kupac.

### Unos i ažuriranje podataka asortimana

Kratak opis: Ažuriranje podataka o asortimanu od strane prijavljene cvećare.

Akteri koji iniciraju slučaj korišćenja: Cvećara.

### Zahtev za validiranje

Kratak opis: Administrator dobija zahtev za validaciju profila cvećare.

Akteri koji iniciraju slučaj korišćenja: Cvećara, Administrator.

### Validacija

Kratak opis: Administrator validira profil cvećare.

Akteri koji iniciraju slučaj korišćenja: Cvećara, Administrator.

### Žalba

Kratak opis: Administrator je primio žalbu od cvećare ili kupca.

Akteri koji iniciraju slučaj korišćenja: Administrator, Cvećara i Kupac.

### Revidiranje žalbe

Kratak opis: Administrator revidira primljenu žalbu.

Akteri koji iniciraju slučaj korišćenja: Administrator.

### Suspendovanje korisnika

Kratak opis: Administrator suspenduje korisnika (cvećaru ili kupca).

Akteri koji iniciraju slučaj korišćenja: Administrator.

### Brisanje korisnika

Kratak opis: Korisniku koji je prethodno suspendovan se briše profil zbog nedoličnog ponašanja.

Akteri koji iniciraju slučaj korišćenja: Administrator, Cvećara i Kupac.

### Komunikacija između cvećare i kupca

Kratak opis: Uspostavljanje komunikacije između cvećare i kupca putem četa.

Akteri koji iniciraju slučaj korišćenja: Administrator, Cvećara i Kupac.

### Ostavljanje i prikaz recenzija

Kratak opis: Kupac i cvećara imaju mogućnost ostavljanja recenzija nakon ostvarene saradnje.

Akteri koji iniciraju slučaj korišćenja: Administrator, Cvećara i Kupac.

### Prikaz cvećara

Kratak opis: Prikazuje se stranica sa cvećarama i njihovim osnovnim podacima.

Akteri koji iniciraju slučaj korišćenja: Posetilac, Cvećara i Kupac.

### Prikaz asortimana

Kratak opis: Prikazuje se stranica sa asortimanima određene cvećare.

Akteri koji iniciraju slučaj korišćenja: Posetilac, Cvećara i Kupac.

### Prikaz konkretnog asortimana

Kratak opis: Prikazuje se stranica sa podacima o izabranom proizvodu.

Akteri koji iniciraju slučaj korišćenja: Posetilac, Cvećara i Kupac.

### Filtriranje cvećare i proizvoda

Kratak opis: Na stranici sa listom cvećara ili listom asortimana akter može filtrirati u odnosu na svoju potražnju.

Akteri koji iniciraju slučaj korišćenja: Posetilac i Kupac.

# 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.

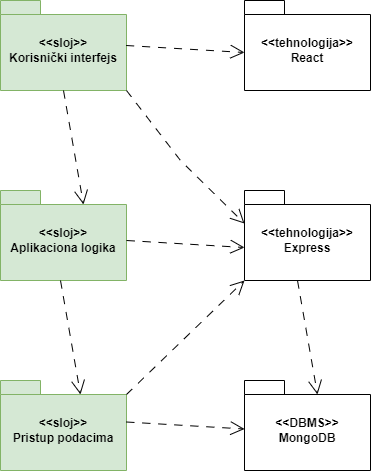
Logički pogled na Flower Power portal obuhvata 3 glavna paketa: Korisnički interfejs, Aplikaciona logika, Pristup podacima.

Paket *Korisnički interfejs* sadrži Web stranice, ReactJS komponente, Redux-Toolkit kao state menagment sistem i multimedijalni sadržaj koji realizuju grafički dizajn i forme preko kojih korisnici sistema komuniciraju sa sistemom (MUI 5).

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

Paket *Pristup podacima* sadrži ExpressJS skripte koje predstavljaju interfejs za pristup, dodavanje i ažuriranje podataka koji se čuvaju u bazi podataka (MongoDB).

## Pregled arhitekture – organizacija paketa i podsistema u slojeve



### Korisnički interfejs

sloj

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

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

### Aplikaciona logika

sloj

Sloj aplikacione logike je srednji sloj u troslojnoj arhitekturi Flower Power shop-a. Sadrži ExpressJS skripte koje realizuju funkcionalnost karakterističnu za domen primene shop-a i uspostavljaju vezu između korisničkog interfejsa i sloja za pristup podacima.

Ovaj sloj zavisi od sloja za pristup podacima i Node engine-a.

### Pristup podacima

sloj

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

Ovaj sloj ne zavisi od drugih slojeva, ali je zavisan od engine-a NodeJs i MongoDB baze podataka.

### ReactJS

tehnologija

Framework ReactJS definiše gradivne elemente stranica koje se prikazuju u Web čitaču i koje omogućavaju prikaz formatiranih informacija i realizaciju formi za unos i ažuriranje podataka kreirajući komponente koje se prevode u HTML kod.

### ExpressJS

tehnologija

Freamwork ExpressJS 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.

### MongoDB

DBMS

MongoDB predstavlja sistem za upravljanje bazama podataka koji će se koristiti za realizaciju Flower Power shop-a.

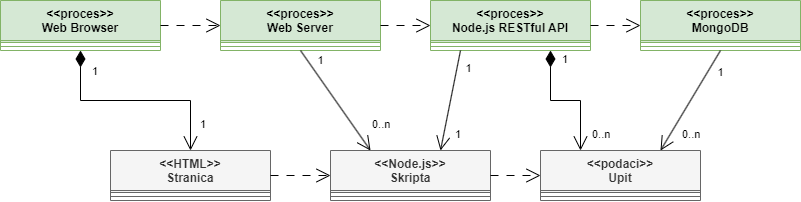
# Pogled na procese

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 Flower Power shop-a kao Web aplikacije.

## Procesi

Na sledećem UML dijagramu klasa prikazani su procesi koji učestvuju u izvršenju Flower Power shop-a. Dijagram je opšteg tipa i može se primeniti na bilo koju Web aplikaciju zasnovanu na Node engine-u i MongoDB bazi podataka.



### Web browser

Web pretraživač 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.

### Web server

Web server je proces koji izvršava funkcionalnost opsluživanja zahteva prispelih sa više Web čitača. Ukoliko je zahtevana stranica Express skript, Web server inicira izvršenje RESTful API procesa koji obrađuje odgovarajući skript i generiše sadržaj koji se vraća čitaču. Web server može paralelno da inicira veći broj RESTful API procesa.

### Node.js RESTful API

Node.js RESTful API proces obavlja posao obrade zadatog Express skripta i generiše odgovarajući tekstualni sadržaj koji Web server šalje Web pretraživaču. Za izvršenje Express skripta ovaj proces može da zahteva usluge MongoDB servera-a. Komunikacija između RESTful API procesa i MongoDB servera se obavlja preko prosleđivanja upita i vraćanja rezultata.

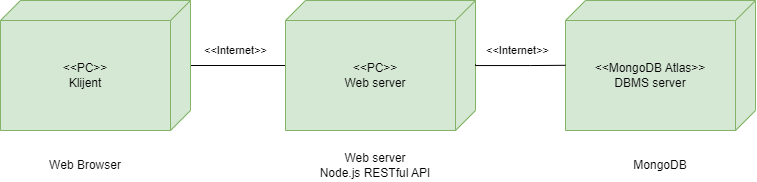
### MongoDB

MongoDB je proces koji izvršava funkcionalnost MongoDB 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 PeNcIL portala.



## Klijent

Pristup Flower Power shop-u se obavlja preko klijentskih računara na kojima se izvršava Web pretraživač. 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 Node.js RESTful API-a koji vrše obradu zadatih Express skripti. U najopštioj konfiguraciji DBMS se izvršava na posebnoj mašini koja je sa Web serverom povezana putem interneta.

## DBMS server

DBMS server je server računar dobijen od MongoDB Atlas provajdera na kome se izvršava MongoDB proces koji realizuje funkcionalnost sistema za upravljanje bazama podataka. Pristup bazi je ograničen konfiguracijom opcija putem MongoDB Atlas web aplikacije.

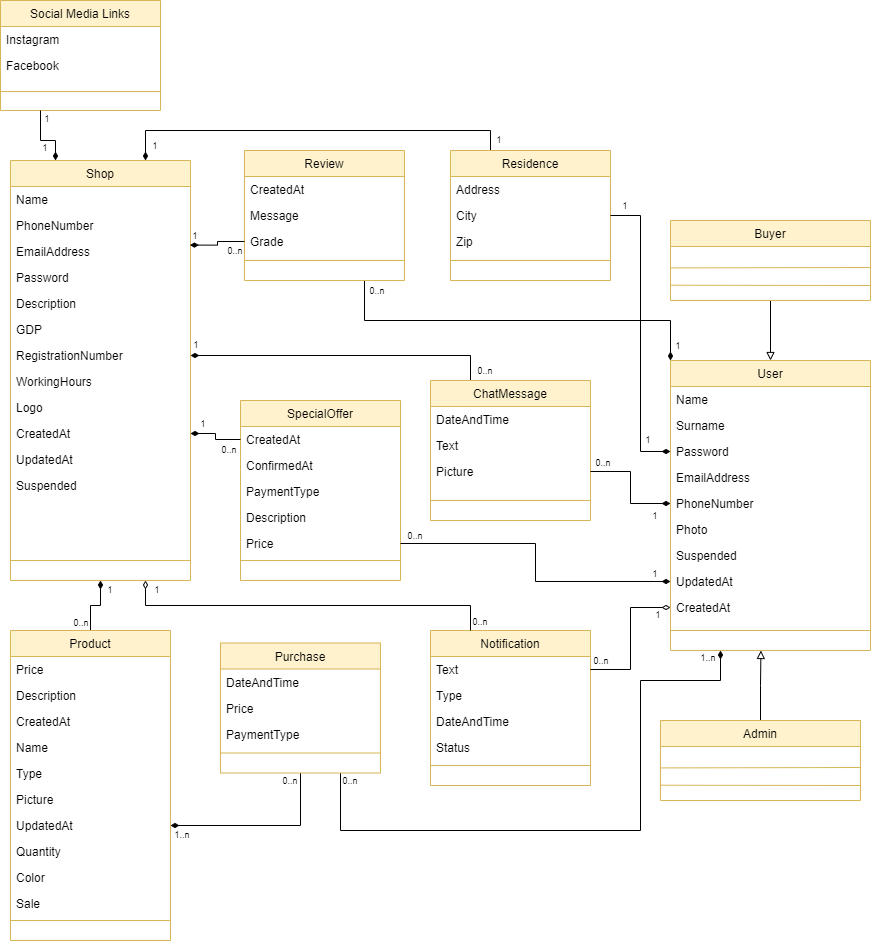
# Pogled na implementaciju sistema

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

## Model domena

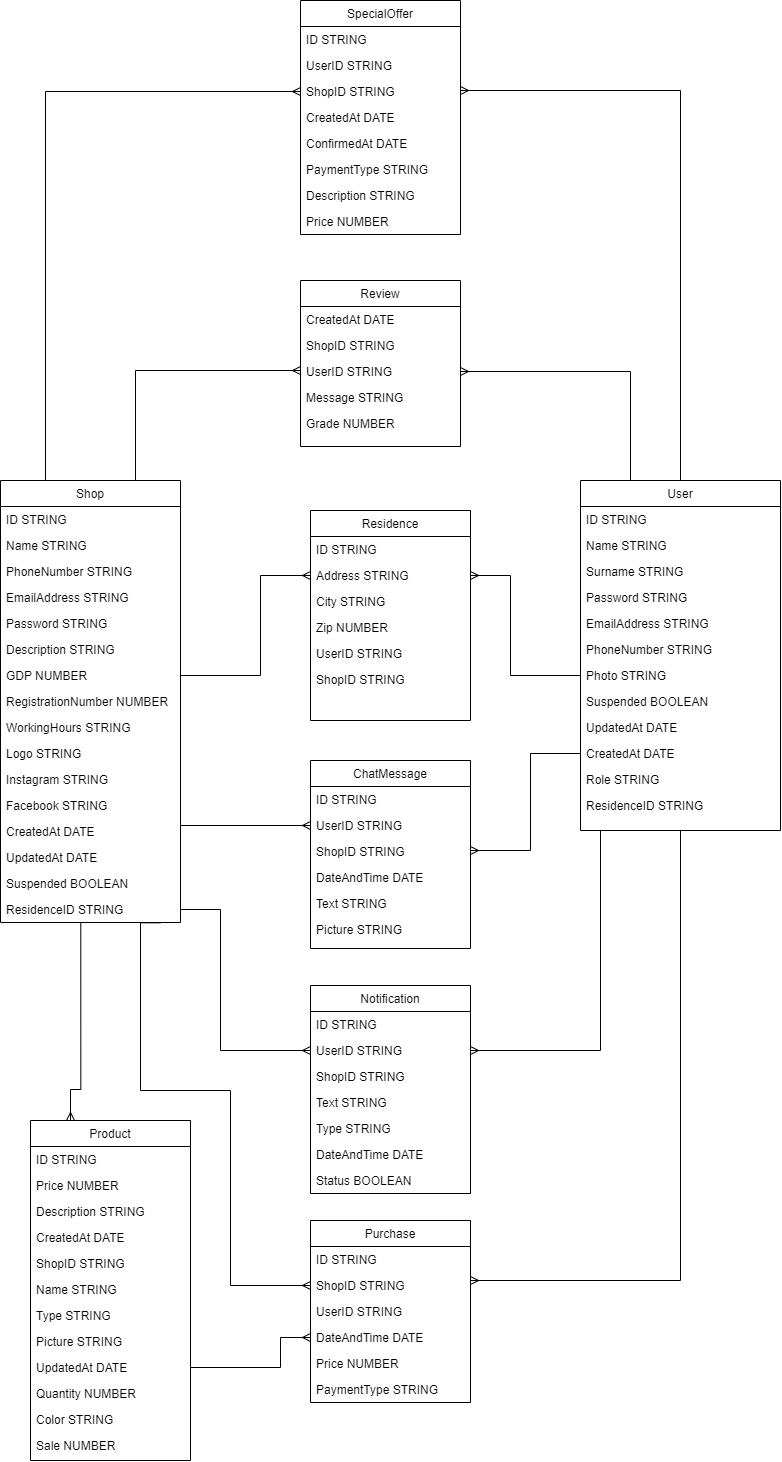
Model domena za koji se PeNcIL portal projektuje je ilustrovan UML dijagramom klasa. U njemu su prikazane domenske klase, neki od njihovih atributa, kao i veze koje se mogu identifikovati između njih.

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



## Šema baze podataka

Detaljna šema baze podataka je prikazana na sledećem dijagramu. Baza podataka i dijagram su kreirani korišćenjem *MongoDB* šema.



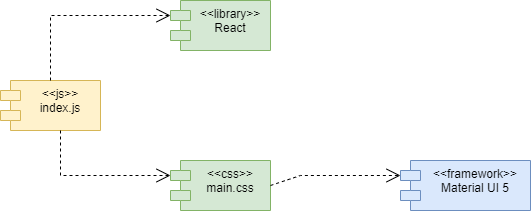
## Komponente sistema

*NAPOMENA: Flower Power projekat nije kompletno implementiran, tj. implementiran je na nivou arhitekturnog prototipa. Zbog ovoga sledeću sekciju trebate shvatiti kao ilustraciju kako bi dokument trebao da izgleda. Opis komponenti sistema za vaše projekte mora biti kompletan! Procena složenosti vašeg rešenja će zavisiti od opisa sistema koji ovde date.*

Komponente sistema Flower Power web shop-a su Node.js skripti čiji će pregled biti dat po arhitekturnim slojevima. Za ilustraciju će biti korišćeni UML dijagrami komponenti, ali i dijagrami klasa. U slučajevima gde je Node.js skript prikazan kao klasa atributi predstavljaju ulazne podatke koji se uzimaju iz GET ili POST dela HTTP poruke, dok metodi predstavljaju funkcije definisane u okviru skripta.

### Komponente korisničkog interfejsa

Dizajn korisničkog interfejsa je skup više različitih komponenti. Sastoji se od index.js skripte, React biblioteke i main.css:



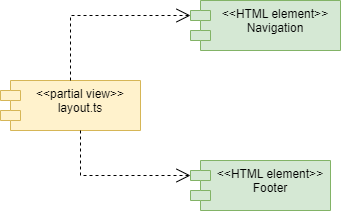
Komponenta **index.js** predstavlja centralnu komponentu korisničkog interfejsa i koristi se za dinamičko generisanje HTML stranica, čiji sadržaj zavisi od parametra koje zahteva korisnik.

Komponenta **main.css** predstavlja opis stilova za pojedine HTML elemente i primenjuje se nad elementima svih stranica.

Komponenta **React** predstavlja javascript biblioteku koja se koristi za implementaciju korisničkog interfejsa sistema.

Komponenta **Material UI 5** predstavlja CSS framework koji omogućava responzivan dizajn korisničkog interfejsa.

Za prikaz zajedničkih elemenata svih stranica koristi se mehanizam layout-a prikaza u okviru React router-a. Parcijalni prikaz se sastoji iz sledećih komponenti:

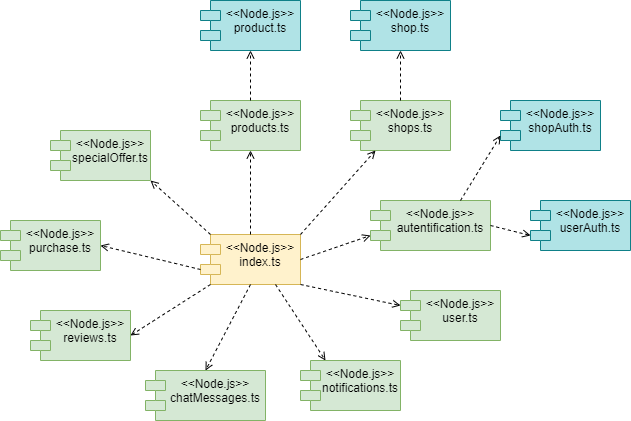


Komponenta **Navigation** predstavlja navigacioni meni sa vezama ka ostalim stranicama i logo-om sajta.

Komponenta **Footer** predstavlja podnožje sajta koje sadrži informacije o autoru, autorskom pravu, kontaktu i veze ka stranicama na društvenim mrežama nas developera.

### Komponente aplikacione logike

Komponente koje realizuju domen problema se uključuju isključivo preko **index.ts** 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:



... opis svih komponenti sa dijagrama sa detaljnim dijagramima po potrebi.

# 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 10000 simultanih pristupa korisnika portalu.
2. Vreme potrebno za pristupanje bazi podataka u cilju izvršenje nekog upita ne sme da bude veće od 3 sekundi.

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. Flower Power će biti dostupan 24 časa dnevno, 7 dana u nedelji. Vreme kada portal nije dostupan ne sme da pređe 10%.
2. Srednje vreme između dva sukcesivna otkaza ne sme da padne ispod 168 sati.