# 3. Koriscene tehnologije

Informacionu tehnologiju, Americka asocijacija za informacione tehnologije definise kao “izucavanje, projektovanje, razvoj, implementacija i podrska ili upravljanje racunarskim informacionim sistemima, softverskim aplikacijama i hardverom”.

Za uspesan i neometan razvoj aplikativnih resenja neophodna je prethona analiza i odabir odredjenih tehnologija koje se najbolje uklapaju u zahteve istog resenja. Bez raznovrsnih modernih tehnologija ne bi bilo moguce izraditi moderna resenja. Iz tog razloga, na tehnologije se obraca posebna paznja i konstantno se radi na njihovom usavrsavanju.

U okrviru ovog poglavlja opisane su sve tehnologije koje su koriscene prilikom razvoja softverskog resenja i istaknut njivoh znacaj.

## 3.1 Prednja strana

Prednja strana (eng. *Front End*) je vidljivi deo veb aplikacije, odnosno deo koji se prikazuje u internet pretrazivacu preko kojeg korisnik komunicira sa servisima zadnje strane. Svako moderno veb resenje se sastoji od tehnologija kao sto su *HTML* (*Hypertext Markup Language*), *CSS* i *JavaScript*. U slucaju da prednja strana ne sadrzi *JavaScript* onda se takva prednja strana naziva statickom.

Prilikom izrade prednje strana treba se teziti prilagodljivom dijaznu (eng. *Responsive design*) koji omogucava optimizovano prikazivanje, jednostavno koriscenje, citanje i kretanje kroz veb sajt koji ce biti pregledan u razlicitim tipovima uredjaja.

### 3.1.1 Angular

Angular je platforma kao i okvir (eng. Framework) koji se upotrebljava za jednostavniju i skalabilniju izradu SPA (eng. Single Page Aplication) aplikacije.

Arhitektura Angular aplikacije se oslanja na odredjene temeljne koncepte. Osnovni gradivni elementi Angular okvira su Angular komponente koje su organizovane u NgModule (eng. *NgModules*). NgModuli prikupljaju srodni kod u funkcionalne skupove, sto implicira da je aplikacija Angular definisana skupom NgModula. Oslanja se na sledece tehnologije:

1. *HTML* (eng. *Hypertext Markup Language*) je opisni jezik koji sluzi za opisivanje izgleda prednje strane, odnosno html je kostur prednje strane.
2. *CSS* (eng. *Cascading Style Sheets*) je jezik za formatiranje pomocu kog se definise izgled elemenata vec stranice.
3. *TypeScript* je jezik koji opisuje ponasanje prednje strane kao i za uspostavljanje komunikacije sa zadnjom stranom.

## 3.2 Zadnja strana

Zadnja strana (eng. *Back End*) je nevidljivi deo koji sadrzi svu funkcionalnost poslovne logike veb aplikacije koja se izvrsava na serveru, prima zahteve od prednje strane i direktno komunicira sa bazom podataka dajuci u isto vreme klijentskoj strain interfejs za pristup tim podacima.

Zadnja strana se uobicajeno realizuje kao RESTful API (eng. Representational State Transfer) koji predstavlja aplikativni interfejs (API) koji je u skladu sa ogranicenjima *REST* arhitektonskog stila i omogucava interakciju sa RESTful veb servisima.

API (eng. Application Programming Interface) je skup definicija i protokola za izgradnju i integraciju aplikativnog softvera.

### 3.2.1 .NET Core

.NET Core je najnovija platforma za opstu namenu koju odrzava Majkrosoft (eng. Microsoft). Radi na razlicitim platformama i redizajniran je na nacin koji cini .NET brzim, fleksibilnim i modernim. Podrzava C#, F# i Visual Basic programske jezike.

### 3.2.2 MVC obrazac

MVC (eng. Model View Controller) je obrazac dizajna koji se koristi za razdvajanje korisnickog interfejsa, modela i aplikacione logike. Obrazac pomaze u postizanju razdvajanja zabrinutosti (eng. Separation of concerns). Koristeci MVC obrazac za veb aplikacije, zahtevi se usmeravaju do kontrolera koji je odgovoran za rad sa modelom za izvrsavanje radnji i/ili preuzimanje podataka. Kontrolor bira pogled za prikaz i daje mu model. Pogled prikazuje konacnu stranicu na osnovu podataka u modelu. Ceo proces je prikazan na slici 3.1.

Graphical user interface, application, timeline

Description automatically generated  
Slika 3.1 MVC obrazac

### 3.2.3 Docker

Doker (eng. Docker) je otvorena platforma za razvoj, isporuku i pokretanje aplikacija, slika 3.2.

Diagram

Description automatically generated  
Slika 3.2 Prezentacija dokera

Doker vam omogucava da odvojite aplikacije od infrastructure, tako da mozete brzo isporuciti softver. Doker omogucava pakovanje aplikacija u standardizovane jedinice za razvoj softvera (eng. Docker Container).

## 3.3 Baza podataka

Baza podataka (eng. Database) je kolekcija podataka organizovanih za brzo pretrazivanje i pristup, koja zajedno sa sistemom za administraciju, organizovanje i memorisanje tih podataka, cini sistem baze podataka.

### 3.3.1 Entity Framework Core i Code First pristup

*Entity Framework (EF) Core* je lagana, prosiriva verzija otvorenog koda koja je podrzana za razlicite platforme za pristup podacima iz baze podataka. Moze sluziti kao objektno-relacioni mapper koji:

* Omogucava .NET programerima rad sa bazom podataka pomocu .NET objekata,
* Eliminise potrebu za vecinom koda za pristup podacima koji obicno treba da se napise

Pristup prvo kod (eng. Core First) predstavlja nacin modelovanja relacionih tabela, realizuje se tako sto se prvo u kodu kreira model, slika 3.3, pa se zatim pomocu naredbe *Add-Migration “naziv-migracije”* kreira nova tabela u kodu koja se nalazi u migraciji. Naredbom *Update-Database* se primenjuje migracija na bazom podataka i dobija se relaciona tabela za te modele podataka, slika 3.4.

(SLIKA)  
Slika 3.3 Primer modela u kodu

(SLIKA)  
Slika 3.4 Primer relacione tabele

### 3.3.2 MSSQL

MSSQL (eng. Microsoft SQL Server) je razvijen relacioni sistem za upravljanje bazama podataka. Relacioni model je predstavljan u tabelernom obliku, slika 3.4, i mogu se podaci pratiti uz pomoc Majkrosoftovog sql menadzmet studia(eng. Microsoft SQL Management Studio).