

Pregled i usporedba suvremenih web tehnologija

Udovičić, Lea

Undergraduate thesis / Završni rad

2021

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka / Sveučilište u Rijeci**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:195:300432>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-07-12**



Repository / Repozitorij:

[Repository of the University of Rijeka, Faculty of Informatics and Digital Technologies - INFORI Repository](#)



Sveučilište u Rijeci – Odjel za informatiku

Preddiplomski jednopredmetni studij informatike

Lea Udovičić

Pregled i usporedba suvremenih web tehnologija

Završni rad

Mentor: doc. dr. sc. Lucia Načinović Prskalo

Rijeka, rujan 2021.

Rijeka, 22.4.2021.

Zadatak za završni rad

Pristupnik: Lea Udovičić

Naziv završnog rada: Pregled i usporedba suvremenih web tehnologija

Naziv završnog rada na eng. jeziku: An overview and comparison of contemporary web technologies

Sadržaj zadatka: Zadaci završnog rada su dati pregled suvremenih web tehnologija, osvrnuti se na njihovu povijest i razvoj. U radu će se također opisati osnovne karakteristike, dati primjeri te usporediti različiti tipovi web tehnologija poput web preglednika, programskih jezika i biblioteka za web, razvojnih okruženja te baza podataka koje se koriste u razvoju web aplikacija.

Mentor

Doc. dr. sc. Lucia Načinović Prskalo

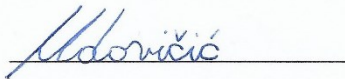


Voditelj za završne radove

doc. dr. sc. Miran Pobar



Zadatak preuzet: 22.4.2021.



(potpis pristupnika)

SADRŽAJ

Sažetak	4
1 Uvod	5
2 World Wide Web	6
2.1 Web stranica.....	6
2.2 Web aplikacija.....	7
2.3 Arhitektura weba	7
3 Tipovi web tehnologija	7
3.1 Web preglednici	8
3.1.1 Google Chrome.....	8
3.1.2 Safari	9
3.1.3 Microsoft Edge.....	11
3.1.4 Mozilla Firefox	12
3.2 Frontend i backend web tehnologije	13
3.3 HTML i CSS	15
3.4 Programski jezici za web.....	17
3.4.1 JavaScript	17
3.4.2 Python.....	18
3.4.3 Java.....	18
3.4.4 PHP	19
3.5 Biblioteke za web	20
3.5.1 Bootstrap.....	20
3.5.2 JQuery	20
3.6 Razvojna okruženja	21
3.6.1 React.....	21
3.6.2 Vue.....	22
3.6.3 Angular.....	23
3.6.4 Express	24
3.6.5 Django	25
3.6.6 Laravel.....	26
3.7 Baze podataka	27

3.7.1	MySQL.....	28
3.7.2	PostgreSQL.....	28
3.7.3	MongoDB.....	29
3.7.4	Firebase	29
4	Usporedba i diskusija	30
4.1	Usporedba programskih jezika za backend	30
4.2	Usporedba razvojnih okruženja za frontend.....	32
4.3	Usporedba razvojnih okruženja za backend	35
4.4	Usporedba relacijskih baza podataka	38
4.5	Usporedba nerelacijskih baza podataka	41
5	Zaključak	44
	Literatura	46
	Popis slika.....	53

SAŽETAK

Tema završnog rada je pregled i usporedba suvremenih web tehnologija. Web tehnologije su različiti alati i tehnike koje se koriste u procesu komunikacije između različitih vrsta uređaja putem interneta. Nakon kratkog uvoda o World Wide Webu, rad je fokusiran na različite tipove web tehnologija koje se koriste za razvoj weba odnosno web aplikacija i njihovu usporedbu. Pošto nam oni omogućuju pregled svih sadržaja na WWW-u, web preglednici su osnovna web tehnologija koju opisujemo i kratko uspoređujemo. Web aplikacije sastoje se od dva glavna dijela: frontend i backend, odnosno prednjeg i stražnjeg dijela. Stoga se web tehnologije za razvoj weba na isti način dijele i u ovom završnom radu. Nakon definiranja, objašnjene su funkcionalnosti i specifikacije pojedinih web tehnologija kao što su: programski jezici, razvojna okruženja, biblioteke i sustavi za upravljanje bazama podataka. Spomenute web tehnologije uspoređuju se unutar istih tipova, putem tri kriterija koji su: jednostavnost/složenost korištenja, popularnost i prema tipu aplikacije koja se izrađuje. Cilj ovog završnog rada je upoznati se sa suvremenim web tehnologijama, vidjeti njihove prednosti i mane te usporedbom njihovih karakteristika, prikazati pogodnost pojedinih tehnologija za različite zadatke koje je potrebno riješiti.

Ključne riječi: World Wide Web, Web preglednik, Frontend, Backend, Programski jezici, Biblioteke za web, Web razvojna okruženja, Web aplikacija

1 UVOD

Računala ne komuniciraju međusobno na način na koji to rade ljudi. Umjesto toga, računala zahtijevaju kodove ili upute. Ti binarni kodovi i naredbe omogućuju računalima da obrađuju potrebne informacije. Svake sekunde obrađuju se milijarde nula i jedinica kako bismo mi kao korisnici interneta dobili tražene informacije ili usluge.

Web tehnologije su različiti alati i tehnike koje se koriste u procesu komunikacije između različitih vrsta uređaja putem interneta. U posljednjih nekoliko desetljeća web tehnologija je doživjela dramatičan prijelaz, od statičnih web stranica do mogućnosti obavljanja vrlo specifičnih i kompleksnih zadataka na mreži [1].

Pojam web tehnologija možemo podijeliti na dva dijela „web“ i „tehnologija“. Web se u ovom slučaju odnosi na World Wide Web, poznatiji kao WWW [1]. On je baziran na više drugačijih tehnologija kao: web preglednici, HTML jezik (engl. *Hypertext Markup Language*) i HTTP protokol (engl. *Hypertext Transfer Protocol*). Dok se tehnologije odnose na sam razvoj, izradu i održavanje web stranica i aplikacija. Uključuje aspekte poput web dizajna, web objavljivanja, web programiranja i upravljanja bazom podataka. Pojednostavljeno rečeno, odnosi se na stvaranje aplikacije koja radi preko interneta [2].

U ovom završnom radu prvo ćemo pregledati povijest i razvoj World Wide Weba i njegovih komponenta. Zatim ćemo detaljnije opisati tipove web tehnologija. Započet ćemo s web preglednicima, te ćemo onda definirati i objasniti razliku između frontend i backend web tehnologija. Za oboje, frontend i backend, pregledat ćemo jezike, biblioteke i razvojna okruženja koja se koriste za razvoj weba. Nadalje, raspodijelit ćemo baze podataka na relacijske i nerelacijske te detaljnije opisati sustave za upravljanje tim bazama podataka. I za kraj ćemo po kategorijama usporediti i diskutirati o navedenim web tehnologijama.

2 WORLD WIDE WEB

World Wide Web (WWW, W3 ili samo Web) je skup tehnologija koje dopuštaju da informacije na internetu budu povezane u dokumentima korištenjem linkova ili veza. Jezik koji se koristi za pisanje tih dokumenata je HTML (engl. *Hypertext Markup Language*), kojeg ćemo detaljnije opisati kasnije [3].

Često se miješa s internetom, ali koncept Weba nastao je kad je Internet već bio uspostavljen, te je WWW usluga ili aplikacija Interneta. Predstavio ga je **Sir Tim Berners-Lee** 1989. godine dok je radio u CERN-u (Europska organizacija za nuklearna istraživanja) u Ženevi. Problem s kojim se je susreo je bio prijenos ideja i rezultata pretraživanja kroz tu organizaciju uz uvjet da svi zadrže svoj operacijski sustav, ali da su i dalje u mogućnosti jednostavno dijeliti informacije. Njegovo rješenje navedenog problema je suradnja između hiperteksta i Interneta. Koncept hiperteksta omogućuje korisnicima da čitaju i istražuju na nelinearni način, jer hipertekst sadrži linkove na drugi tekst, sliku, video ili audio. Ali prateći linkove odnosno veze, korisnik nije prisiljen pratiti nikakav redoslijed, već korisnik sam određuje slijed praćenja informacija tokom čitanja [3].

Prva verzija Weba bila je o izgradnji „mreže čvorova“ za pohranu „hiper-tekstualne stranice“ koje pregledavaju „preglednici“ na mreži. HTTP protokol i HTML su bili primarni mehanizmi za funkcioniranje weba. HTTP (engl. *Hypertext Transfer Protocol*) je protokol aplikacijskog sloja koji služi za izmjenjivanje podataka između web preglednika i servera. Odnosno, HTTP je jednostavan način na koji jedno računalo (klijent) traži od drugog sustava (poslužitelja) web stranice. HTML (engl. *Hypertext Markup Language*) je jezik kojim su te stranice napisane tako da ih bilo koje računalo može razumjeti i ispravno prikazati uz pomoć web preglednika. Od važnijih pojmova, tu je i URL (engl. *Uniform Resource Locator*) što je „adresa“ web stranice i pruža jedinstvenu shemu označavanja za adresiranje resursa putem mreže [3].

2.1 WEB STRANICA

Web stranica je dokument koji se prikazuje u web pregledniku. Mogu biti statičke i dinamičke. Statičke web stranice su konstante odnosno nepromjenjive. Sadrže isti unaprijed izgrađeni sadržaj pri svakom učitavanju stranice.

Dinamičke web stranice se mijenjaju. Sadrže programski kod na strani poslužitelja što omogućuje generiranje jedinstvenog koda svaki put kada se stranica učita. Na primjer može prikazati trenutno vrijeme i datum ili može ispisati jedinstveni odgovor na temelju web obrasca koji je korisnik ispunio. Puno dinamičkih web stranica koristi kod na poslužiteljskoj strani za dohvat podatka iz baze podataka. Zahtjev dinamičke web stranice sastoji se od dvije razine sustava odgovora koji uključuju web poslužitelj i poslužitelj aplikacije. Primarna zadaća web poslužitelja je da prikaže sadržaj web stranice, dok je poslužitelj aplikacije zadužen za logiku odnosno za interakciju između korisnika i prikazanog sadržaja [3].

2.2 WEB APLIKACIJA

Web stranice su prvenstveno informativnog karaktera, dok web aplikacije dopuštaju korisniku izvršavanje radnji. Web aplikacije su klijent-poslužitelj softverske aplikacije gdje se klijent (ili korisničko sučelje) pokreće u web pregledniku. One zapravo obuhvaćaju sve aplikacije koje komuniciraju s korisnikom putem HTTP-a. To uključuje i jednostavne aplikacije poput online kalkulatora ili kalendara, kao i one kompleksnije aplikacije [3].

2.3 ARHITEKTURA WEBA

Web je u osnovi **korisničko-poslužiteljski** (klijent-server) sustav. Web preglednici imaju ulogu klijenta, podnoseći zahtjeve na web poslužitelje. Web poslužitelji odgovaraju na zahtjeve s traženim podacima. Web aplikacije su obično implementirane s dvoslojnom, troslojnom ili višeslojnom (n-slojnom) arhitekturom. Svaki sloj je platforma (klijent ili poslužitelj) s jedinstvenom odgovornošću [3].

U dvoslojnoj arhitekturi, prvi sloj je klijentska platforma koja sadrži web preglednik, a drugi sloj je poslužiteljska platforma koja sadrži sve softver komponente poslužitelja. Ova arhitektura je jeftina, ali ima ograničenu skalabilnost. Pogodna je za 10-100 korisnika [3].

U troslojnoj arhitekturi, treći sloj preuzima funkciju poslužitelja. Drugi sloj obično preuzima upravljanje podacima. Ova arhitektura pruža poboljšane performanse i skalabilnost. Troslojna arhitektura može se proširiti na n-arhitekturu, koja sadrži nekoliko posebnih aplikacijskih poslužitelja [3].

3 TIPOVI WEB TEHNOLOGIJA

Već smo rekli da su web tehnologije različiti alati i tehnike koje se koriste u procesu komunikacije između različitih vrsta uređaja putem interneta. U ovom ćemo poglavlju detaljnije opisati i razraditi različite tipove najpopularnijih web tehnologija. Krenut ćemo s web preglednicima koji nam omogućuju sam pregled sadržaja. Zatim ćemo se najviše fokusirati na sam razvoj weba.

Web razvoj (engl. *web development*) se donosi na izgradnju, izradu i održavanje same web stranice ili aplikacije. Uključuje web dizajn, programiranje i upravljanje bazom podataka. Tipovi web tehnologija koje se koriste za sam razvoj weba koje ćemo opisati u nastavku jesu [2]:

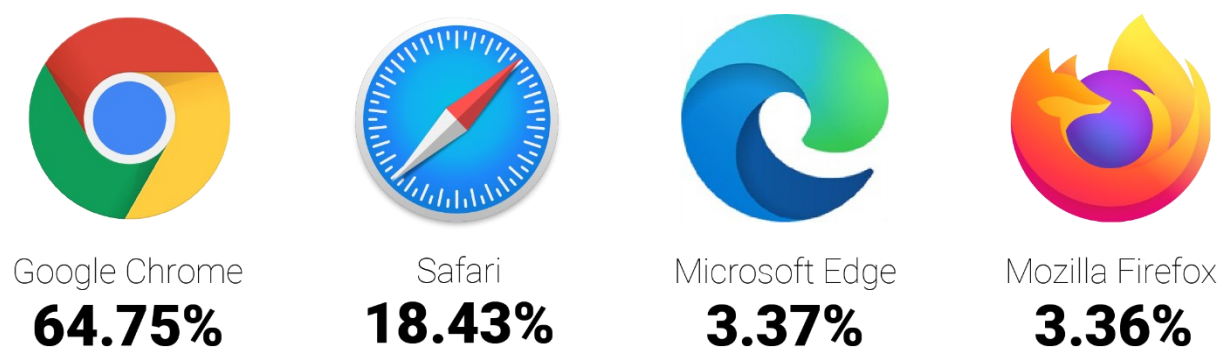
- HTML i CSS

- Programski jezici i biblioteke za web
- Razvojna okruženja
- Sustavi za upravljanje bazom podataka

3.1 WEB PREGLEDNICI

Web preglednik omogućuje korisnicima pregled svih sadržaja koji su dio World Wide Weba. Kada korisnik zatraži web stranicu, web preglednik preuzima potreban sadržaj s web poslužitelja te prikazuje web stranicu u formatu koji korisnik vidi i razumije [4].

Na slici 3.1 je prikazan postotak koji predstavlja udio korištenja web preglednika, odnosno postotak posjetitelja web stranica koji koriste određeni web preglednik. Podaci su iz svibnja 2021. godine i preuzeti su s web stranice Wikipedia [5]. Možemo vidjeti da **Google Chrome** ima visokih 65.75% te je daleko najpopularniji i najviše korišten web preglednik na tržištu. Nakon njega dolazi web preglednik **Safari** s 18.43%, **Microsoft Edge** s 3.37% te **Mozilla Firefox** s 3.36%. Kako su ovo četiri najpopularnija web preglednika među korisnicima, detaljnije ćemo ih pregledati u nastavku.



Slika 3.1 Udio korištenja web preglednika izražen u postocima (Izvor: autorski rad)

3.1.1 Google Chrome

Google Chrome je trenutno najomiljeniji web preglednik među korisnicima. Razvila ga je američka tvrtka Google, a prva stabilna verzija objavljena je 11. prosinca 2008. za Microsoft Windows operacijski sustav te je nešto kasnije prenesen na Linux, MacOS, iOS i Android [6].

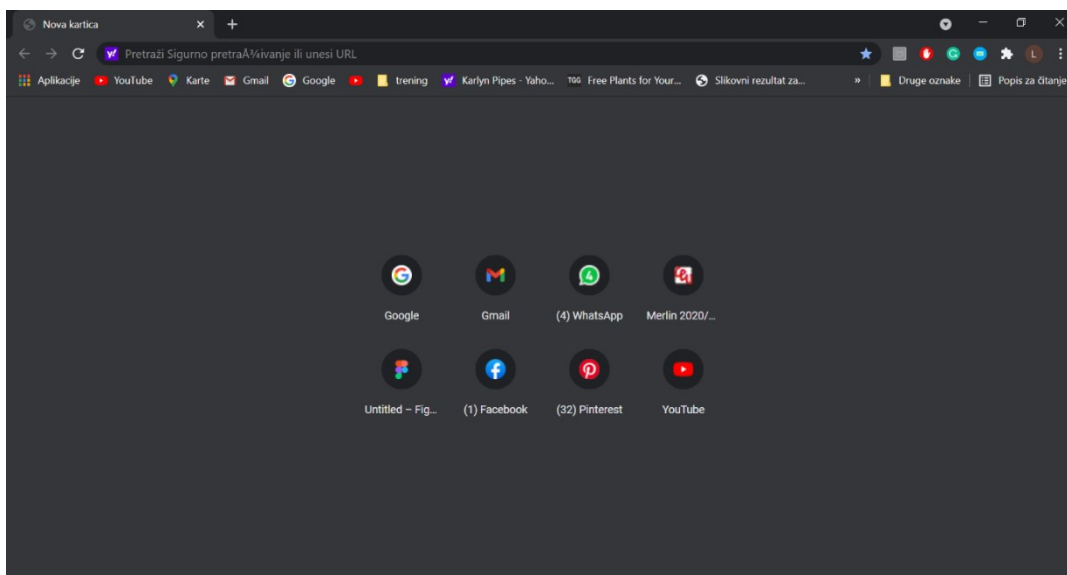
Atraktivnog je izgleda i brzih performansi. Najbolje ocjenjeni preglednik na HTML5Test web stranici. Chrome koristi više RAM-a od ostalih Windows preglednika, ali djelomice je to zbog unaprijed učitano sadržaja što znatno ubrzava rad samog preglednika. Chrome se ne može pohvaliti jedinstvenim značajkama kao što su: ugrađeni VPN, način čitanja, gumb za dijeljenje, alat za snimanje zaslona i ormarić za kripto-valute. No s druge strane ima odličnu biblioteku

dodataka, podršku i sinkronizaciju, značajke automatskog popunjavanja i alate za web developere [7].

Konstanta sigurnosna ažuriranja koja se izvršavaju svakih 6-8 tjedana čine ga jednim od najsigurnijih web preglednika koji otklanja moguće prijetnje jako brzo. Korisnicima daje jasno upozorenje kada koriste nesigurnu HTTP vezu, a JavaScript se može u potpunosti blokirati [8].

Iako je Chrome na vrhu liste kada se radi o sigurnosti korisnika, privatnost korisnika nije njegova jača strana. Za razliku od Safarija, Microsoft Edgea i Firefoxa, Chrome sve prikupljene podatke kao što su povijest pregledavanja, podatke o upotrebi i lokacije povezuje direktno s uređajima i pojedincima odnosno korisnicima [9].

Na slici 3.2 je prikazana početna stranica korisničkog sučelja Google Chromea. Vidimo da imamo prikazane najčešće posjećene stranice u sredini sučelja, a na alatnoj traci imamo označene web aplikacije koje možemo sami definirati, ili Chrome to odradi umjesto nas. U gornjem desnom kutu su nam prikazani dodatni alati odnosno proširenja koje smo instalirali u web preglednik. Chrome ima veliku biblioteku dodatnih alata za instalaciju. Ovdje na primjer možemo vidjeti proširenje aplikacije *Grammarly* koja ispravlja greške u tekstu napisanom na engleskom jeziku i možemo vidjeti *AD blocker* koji sprječava otvaranje skočnih prozora. Google Chrome ima najveću biblioteku proširenja u usporedbi s ostalim web preglednicima. Ispod dodatnih alata možemo vidjeti popis za čitanje, gdje možemo spremiti web stranice te ih čitati kasnije, kad god to poželimo. To je jedna od značajki koje drugi web preglednici nemaju.



Slika 3.2 Početna stranica korisničkog sučelja Google Chrome web preglednika (Izvor: autorski rad)

3.1.2 Safari

Safari je web preglednik kojeg je razvila američka tvrtka Apple. Prva verzija objavljena je 2003. i dostupan je samo na operacijskim sustavima MacOS i iOS [10].

Safari je poznat po svom jednostavnom i minimalističkom sučelju, nema opcija prilagođavanja izgleda npr. mijenjanja boje, teme, fonta itd., kao što je slučaj kod ostalih web preglednika. Brzo učitava stranice, lagan je i jednostavan za korištenje.

Sprječava učitavanje sumnjivih web stranica i upozorava korisnika na potencijalnu opasnost. Pokretanjem web stranica u sandboxu, također sprječava utjecaj zlonamjernog koda s jedne stranice na cijeli preglednik ili pristup podacima korisnika. Svojom nedavno objavljenom *Intelligent Tracking Prevention (ITP)* značajkom unaprijedio je korisnikovu privatnost time što je otežao web stranicama odnosno oglašivačima praćenje korisnika. Nova značajka također je otežala web stranicama trećih strana da ostavljaju podatke u predmemoriji korisnika te tako korisnik ostaje anoniman na mreži [8].

Za razliku od ostalih navedenih web preglednika, Safari jedini nije niti djelomično „open-source“ koda. Na slici 3.3 vidimo početnu stranicu korisničkog sučelja web preglednika Safari. Od svih web preglednika, Safari ima najmanje mogućnosti personalizacije pogotovo kad je u pitanju izgled. Drugi web preglednici imaju puno izbora teme, različitih boja, dizajna, slika i još puno toga. U Safariju imamo svjetlu ili tamnu temu, koja se ne može postaviti posebno za web preglednik, nego ona ovisi o tome je li na vašem cijelom operacijskom sustavu odnosno uređaju postavljena svjetla ili tamna tema. Također, ima najmanju biblioteku proširenja u usporedbi s drugim web preglednicima. Početna stranica ne razlikuje se puno od ostalih - vidimo alatnu traku za pretraživanje koja je dosta minimalističkog izgleda, na nju se mogu dodati označena web mjesta. Imamo prikazane favorite, odnosno najposjećenije stranice i izvješće privatnosti.

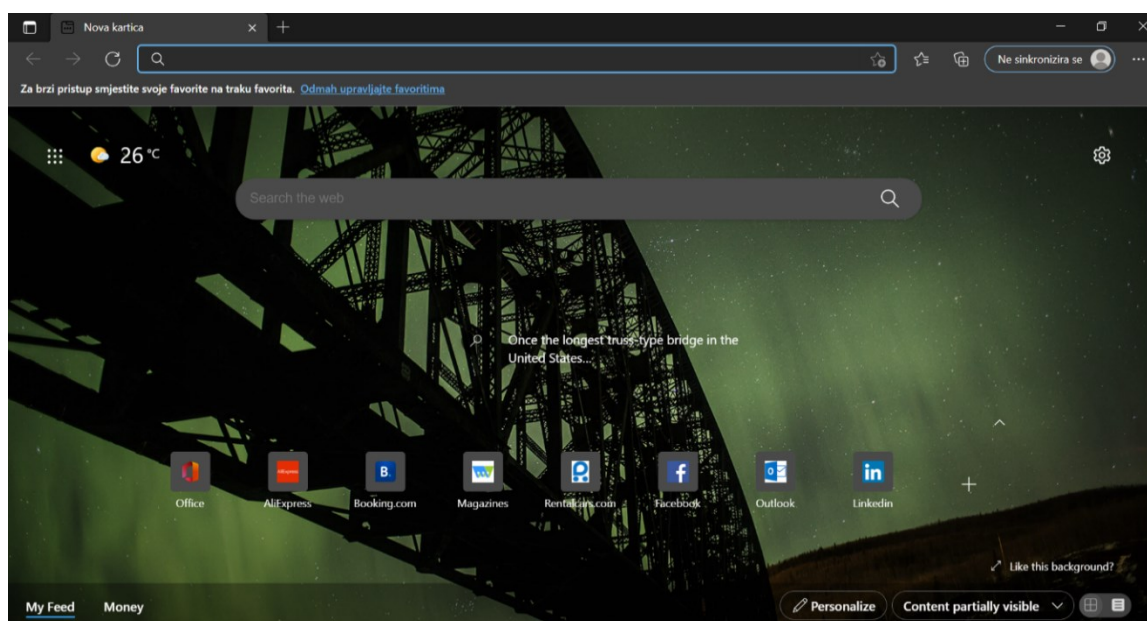


Slika 3.3 Početna stranica korisničkog sučelja preglednika Safari [78]

3.1.3 Microsoft Edge

Microsoft Edge je web preglednik kojeg je razvila tvrtka Microsoft, prva verzija objavljena je 29. srpnja 2015. kao zamjena za Internet Explorer. Zadani je preglednik operacijskog sustava Windows, a dostupan je i na MacOS, Android i iOS operacijskim sustavima [11]. Vodeći je preglednik u performansama, kao i štedljivoj upotrebi memorije i diska. Nudi tri postavke razine privatnosti: osnovnu, uravnoteženu i strogu. Sam preglednik ažurira svoj softver jednom tjedno, većinom poboljšavajući sigurnost. Ono što ga odvaja od konkurencije je: podrška za Dolby Digital, objectRTC i API za snimanje zaslona [7].

Na slici 3.4 vidimo početnu stranicu korisničkog sučelja web preglednika Microsoft Edgea. Vrlo je sličan ostalima po izgledu - imamo traku za pretraživanje i listu najposjećenijih web stranica. Također možemo dodati favorite u traku za favorite. Isto tako ima velike mogućnosti personalizacije dizajna, boja i tema. Kako pozadinska slika nije postavljena od strane mene nego od strane samog preglednika, vidimo da nas u donjem desnom kutu pita svida li nam se odabrana pozadinska slika kako bi ju mogao prilagoditi nama. Ono što njegovu početnu stranicu razlikuje od početnih stranica ostalih preglednika, je što u gornjem lijevom kutu imamo izbornik svih Microsoft Office alata, pošto je Edge zadani preglednik operacijskog sustava Windows. Pored toga imamo ikonu koja nam prikazuje vrijeme temeljno na našoj trenutnoj lokaciji ili zadanoj lokaciji. Na samom dnu imamo dio „My Feed“, koji nam, kada odemo na njega, prikazuje vijesti, koje također imaju velike mogućnosti personalizacije i prati podatke korisnika te prikazuje vijesti prilagođene korisniku i njegovim preferencijama. Ima i „Money“ sekciju gdje se prikazuje tržište dionica, a koja po mom mišljenju ta sekcija nema dovoljno dobre postavke personalizacije. Ali ova dva dodatka odnosno sekcije drugi preglednici nemaju. Također, sve je to povezano s operacijskim sustavom Windows - iste vijesti, personalizacija i prilagođen sadržaj prikazuju se i na sekciji „My Feed“ na samom operacijskom sustavu.



Slika 3.4 Početna stranica korisničkog sučelja web preglednika Microsoft Edge (Izvor: autorski rad)

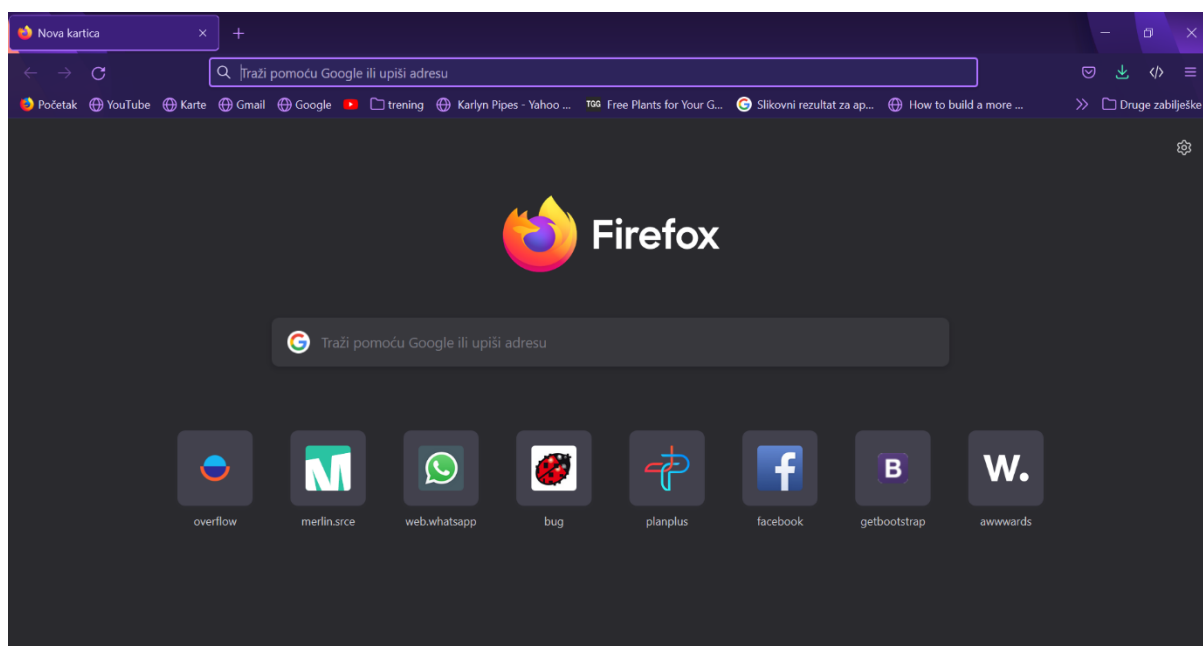
3.1.4 Mozilla Firefox

Mozilla Firefox je jedini u cijelosti „open source“ web preglednik, te je također jedini preglednik razvijen od strane neprofitabilne organizacije. Prva verzija objavljena je 2002. godine. Dostupan je na Windows, MacOS, Linux i Android operacijskim sustavima [12].

Prvi preglednik koji je omogućio promjenu i prilagodbu teme preglednika, te najbrže podržava nove HTML5 i CSS mogućnosti. Nudi novu uslugu upravljanja lozinkama koja generira složene lozinke, sinkronizira ih među uređajima i sve ih zaštiti s jakom glavnom lozinkom [7]. Ne ažurira se često kao Google Chrome ili Edge, ali s obzirom da je Mozilla neprofitabilna organizacija impresivno je vidjeti kako njeni programerski volonteri u nekoliko tjedana obnove Firefox najnovijim sigurnosnim značajkama [9].

Firefox nudi brojne sigurnosne značajke: zaštita od krađe identiteta i zlonamjernog softvera, blokiranje web stranice koje su prijavljene za napad ili krivotvorine, upozorenje korisnika kada web mjesto pokušava instalirati dodatke, zabrana platformama poput Facebooka praćenje korisnikove aktivnosti izvan same platforme. Od svih navedenih preglednika Firefox ima najveću razinu privatnosti za korisnika [8].

Na slici 3.5 vidimo početnu stranicu korisničkog sučelja web preglednika Mozilla Firefox. Opet, vrlo je slična kao i kod ostalih. Imamo alatnu traku, gdje imamo označene web stranice kao favorite. U gornjem desnom kutu su nam prikazana instalirana proširenja. Iako Mozilla ima bogatu biblioteku proširenja, ovdje imamo samo RESTED dodatak. Na sredini ekrana imamo traku za pretraživanje i kao i kod ostalih, popis najposjećenijih stranica. Kao i kod Chromea i Edgea, ima velike mogućnosti personalizacije dizajna, tema, boje, favorita, dodataka i svega ostalog.



Slika 3.5 Početna stranica korisničkog sučelja web preglednika Mozilla Firefox (Izvor: autorski rad)

3.2 FRONTEND I BACKEND WEB TEHNOLOGIJE

U sljedećim poglavljima, raspravljat ćemo o web tehnologijama i alatima koji se koriste za razvoj weba. No prije toga moramo jasno definirati razliku između dva različita dijela web stranice ili aplikacije: frontend i backend, jer se i web tehnologije razlikuju s obzirom na dio web aplikacije koji se razvija. Tako da ćemo raspodijeliti web tehnologije na one koje se koriste za razvoj frontenda i one koje se koriste za razvoj backenda te ih po tome i usporediti.

Frontend je prednji dio web aplikacije i odnosi se na sve one dijelove web stranice koje korisnik može vidjeti na svom ekranu i s kojima vrši interakciju. Često se još naziva i klijentski dio aplikacije [2]. Zadaća programera koji razvijaju frontend odnosno prednji dio aplikacije je napisati kod vizualnih elemenata. Usredotočeni su na ono šta korisnik vidi kada posjeti web lokaciju, a pritom je bitno da je web mjesto jednostavno za interakciju, a da pritom radi glatko i ima dobre performanse. Navest ćemo web tehnologije koje se koriste za razvoj prednjeg dijela web aplikacija [13].

Prva web tehnologija su programski jezici. Kod razvoja prednjeg dijela weba najčešće se koriste tri jezika: HTML, CSS i programski jezik JavaScript (slika 3.6). Detaljnije ćemo ih obraditi u nastavku.



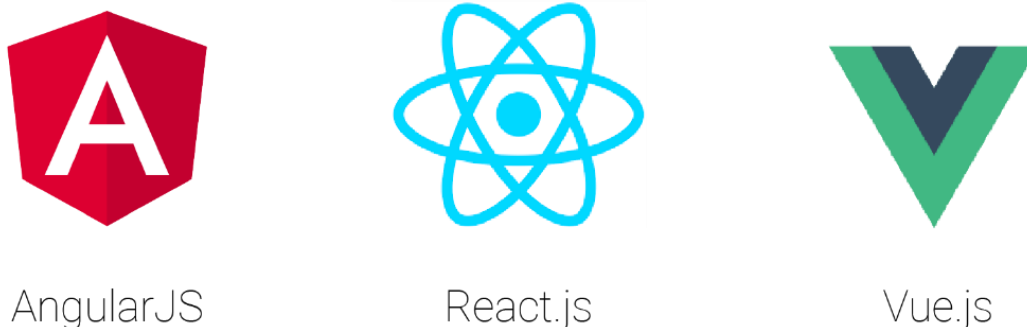
Slika 3.6 Jezici za frontend (Izvor: autorski rad)

Nadalje, imamo biblioteke. Dvije najpopularnije trenutno za razvoj prednjeg dijela weba su: Bootstrap i JQuery (slika 7).



Slika 3.7 Biblioteke za frontend (Izvor: autorski rad)

Te zadnje imamo razvojna okruženja. Spomenuti ćemo tri razvojna okruženja za razvoj prednjeg dijela weba koja su trenutno najpopularnije među korisnicima i najviše se koriste - svi su za programski jezik JavaScript. To su Angular, React.js i Vue.js (slika 3.8).



Slika 3.8 Razvojna okruženja za frontend (Izvor: autorski rad)

Backend je stražnji dio web aplikacije koja se sastoji od poslužitelja koji pruža podatke na zahtjev, aplikacije i baze podataka u kojoj se nalaze i organiziraju svi potrebni podaci i informacije [14]. U backendu se nalaze funkcije, metode i elementi za manipuliranje podacima. To je dio aplikacije koju korisnik ne vidi i s kojom ne vrši direktnu interakciju [3]. Iako korisnik ne vrši direktnu interakciju sa stražnjim dijelom web aplikacije, indirektno vrši interakciju kroz elemente u prednjem dijelu aplikacije. Često se još naziva i poslužiteljski dio aplikacije. Aktivnosti, kao pisanje API-ja (skup određenih pravila i specifikacija koje programeri slijede tako da se mogu služiti uslugama ili resursima operacijskog sustava ili nekog drugog složenog programa [15]), stvaranja biblioteka i rad s komponentama sustava bez korisničkog sučelja, također su dio backend razvoja [14]. Kao i za frontend, navest ćemo web tehnologije koje se koriste za razvoj stražnjeg dijela weba.

Prvo imamo programske jezike. U ovom radu ćemo detaljnije opisati i usporediti tri trenutno najpopularnija jezika za razvoj backenda, a to su: PHP, Python i Java (slika 3.9). Uz njih se još koriste i Ruby, C#, C++ itd., ali znatno rjeđe.



Slika 3.9 Programski jezici za backend (Izvor: autorski rad)

Nakon programskih jezika, imamo razvojna okruženja. U ovom radu ćemo detaljnije opisati i usporediti tri najpopularnija razvojna okruženja za razvoj stražnjeg dijela weba. To su: Express.js, Django i Laravel (slika 3.10). Koriste se još i: Flask, Ruby on Rails, ASP.net i drugi.



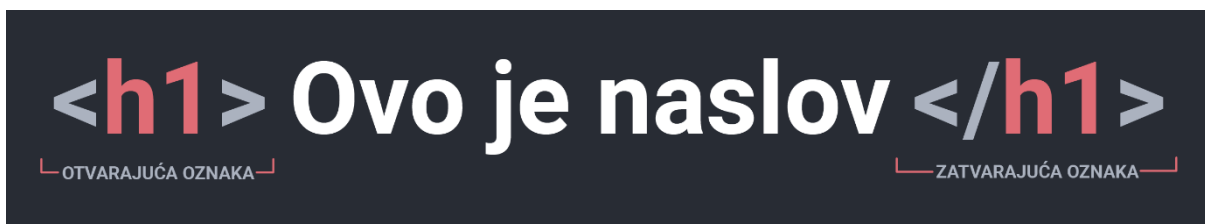
Slika 3.10 Razvojna okruženja za backend (Izvor: autorski rad)

3.3 HTML i CSS

HTML i CSS dvije su osnovne tehnologije za razvoj weba. Pojednostavljeno rečeno, HTML pruža strukturu stranice, dok CSS pruža izgled stranice. Uz grafiku i skriptiranje, HTML i CSS su neizostavni u izradi web stranica i aplikacija.

HTML (engl. *HyperText Markup Language*) je prezentacijski jezik koji definira značenje i strukturu web aplikacije, a to radi pomoću HTML elemenata koji su smješteni unutar izlomljenih zagrada [16]. Elementi se najčešće sastoje od dvije oznake (engl. tags): otvarajuće i zatvarajuće oznake. Završna oznaka ima dodatnu kosu crticu. Svaki HTML element daje web pregledniku informaciju o tome šta se nalazi između oznaka. HTML ima nekoliko verzija, trenutna je verzija 5 odnosno HTML5 [17].

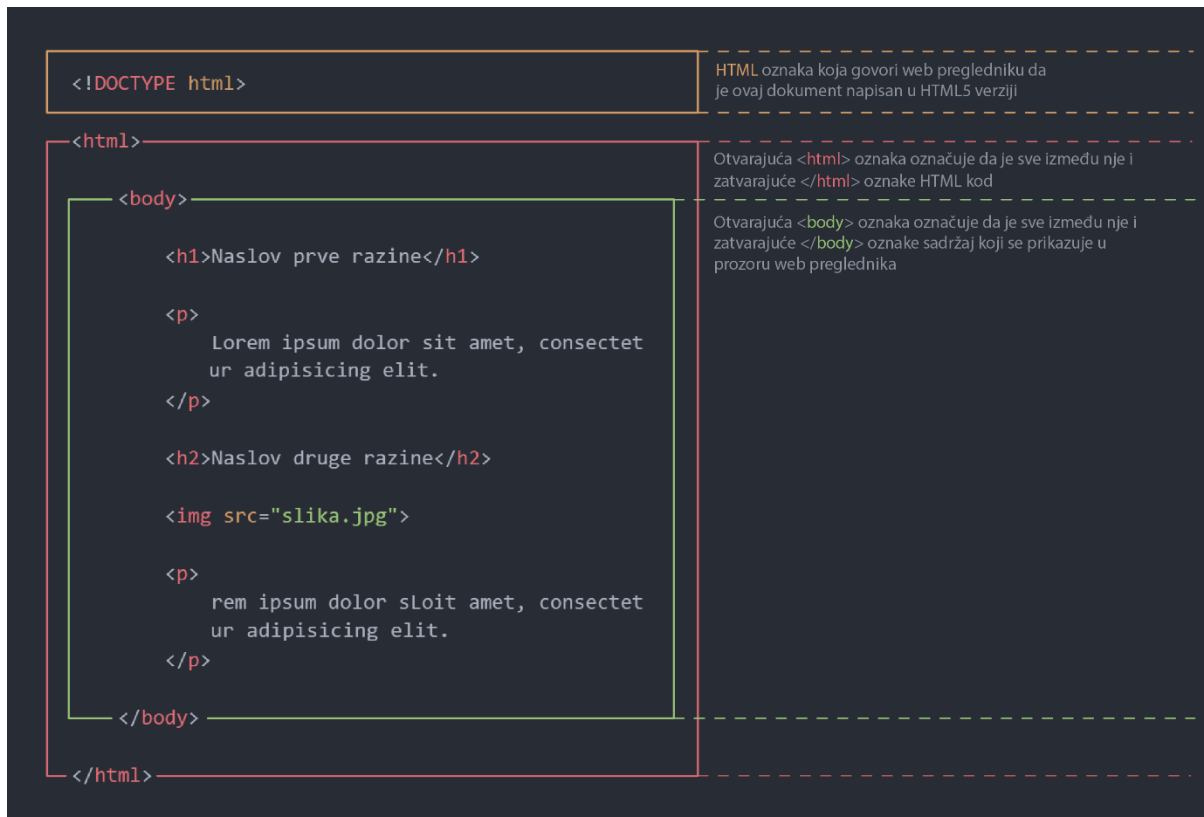
Na slici 3.11 vidimo primjer HTML elementa naslova prve razine. Označene su nam otvarajuće i zatvarajuće oznake `<h1>` `</h1>` koje daju web pregledniku informaciju da je tekst između njih naslov prve razine. Web preglednik će dati neki osnovni dizajn tom naslovu, koji se mijenja pomoću CSS-a.



Slika 3.11 Primjer HTML elementa - Naslov prve razine (Izvor: autorski rad)

Na slici 3.12 možemo vidjeti primjer jednog HTML dokumenta. Prvo imamo izraz `<!DOCTYPE html>` - to je HTML oznaka koja govori pregledniku u kojoj verziji HTML-a je dokument napisan, u ovom slučaju to je verzija 5, ujedno i zadnja verzija. Nakon toga vidimo da imamo oznake `<html>` `</html>` koju moramo napisati kako bi web preglednik znao da je sve između tih oznaka HTML kod. Također imamo `<body>` `</body>` oznake koje su tu da bi web preglednik znao da je sve između njih sadržaj koji treba prikazati. Unutar tih oznaka

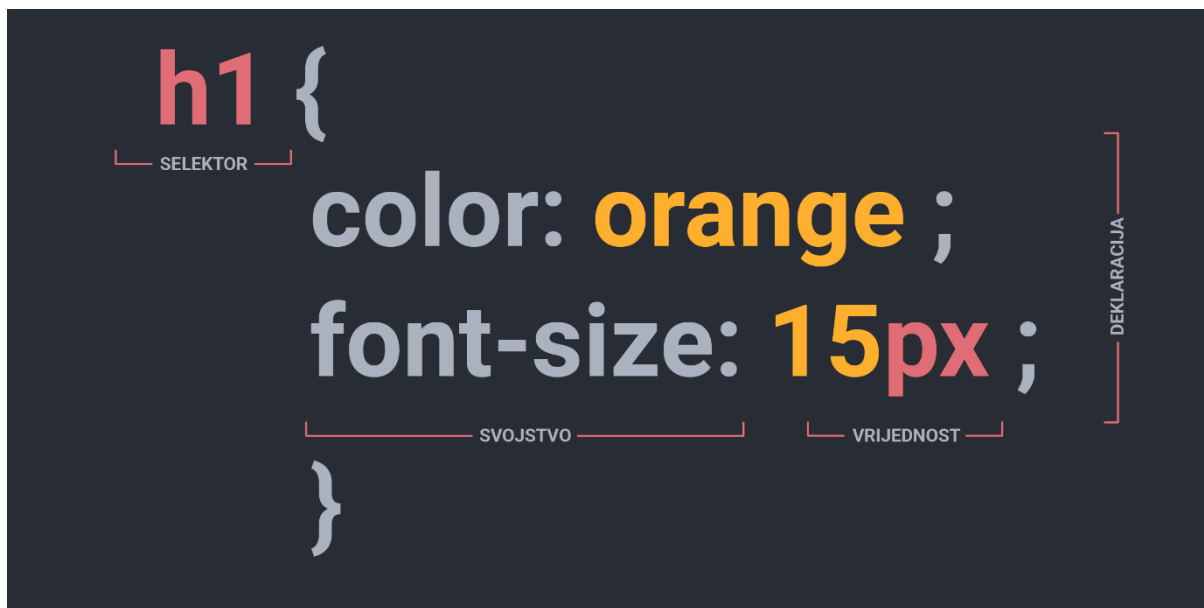
imamo naslov prve razine, dva paragrafa označena oznakama `<p>` `</p>`, te imamo `` oznaku za označavanje fotografija te možemo primijetiti da ona nema zatvarajuću oznaku.



Slika 3.12 Primjer HTML dokumenta (Izvor: autorski rad)

CSS (engl. *Cascading Style Sheets*) je stilski jezik koji omogućuje pisanje pravila koja specificiraju izgled elementa web aplikacije odnosno HTML elementa [18]. HTML-om, primjerice, definiramo da je neki tekst naslov, a CSS-om definiramo da će on biti npr.: crvene boje, pisan „Times New Roman“ fontom i veličine 25px. CSS pravilo se sastoji od dva dijela: selektora i deklaracije koja je definirana kao svojstvo i vrijednost tog svojstva unutar vitičastih zagrada. Selektor označava element na koji se pravilo unutar zagrada aplicira, a deklaracija definira pravilo kako selektirani element treba izgledati [17].

Na slici 3.13 možemo vidjeti primjer CSS pravila. Selektor nam označava da mijenjamo sve naslove prve razine u HTML dokumentu. A u deklaraciji imamo dva pravila. Prvo pravilo mijenja boju naslova u narančastu. U ovom pravilu „color“ je svojstvo kojeg mijenjamo, a „orange“ je vrijednost tog svojstva. Drugo pravilo mijenja veličinu svih naslova prve razine na 15 piksela. U ovom pravilu svojstvo je „font-size“, a vrijednost tog svojstva je „15px“.



Slika 3.13 Primjer CSS pravila (Izvor: autorski rad)

3.4 PROGRAMSKI JEZICI ZA WEB

Programski jezici predstavljaju način komunikacije s računalima te su zato najosnovnija komponenta izrade web aplikacija. Postoji mnogo različitih programskih jezika za web, svaki različite sintakse, programskog stila i složenosti. Svake godine popis programskih jezika za web sve je veći, time je neizbježno da jezici zastare i zamjene ih popularniji.

Što se tiče programskih jezika za razvoj prednje strane weba (frontend), nemamo šta drugo spomenuti osim JavaScripta u kojem je napisano 97,6% svih web stranica [19]. Za razvoj stražnjeg dijela weba (backend) je situacija drugačija i ima puno više opcija.

3.4.1 JavaScript

JavaScript je skriptni programski jezik koji je omogućio interaktivnost web stranica. Trenutno je jedan od najpopularnijih i vodećih programskih jezika u svijetu tehnologije. Jedan od razloga za to je što se može koristiti i na prednjem dijelu (frontend) weba i na stražnjem dijelu (backend) odnosno i na klijentskoj i na poslužiteljskoj strani [20].

Interpreterski je jezik, što znači da se ne mora kompajlirati [20]. Kako web preglednici imaju ugrađeni JavaScript engine, programski kod se može izvršavati i interpretirati direktno u pregledniku ili kroz Node.js koji se interpretira na poslužiteljskoj strani što nam omogućuje pisanje JavaScripta direktno unutar HTML dokumenta između `<script>` `</script>` HTML oznake [19]. To ga čini jako kompatibilnim s drugim jezicima. Bilo da se radi o PHP-u, Python-u, Javi ili Ruby-u, korisnik će imati uvijek isto iskustvo bez obzira koji programski jezik se koristio na poslužiteljskoj strani [21].

Snaga JavaScripta nije samo to što se može pokrenuti na pregledniku i poslužitelju koristeći Node.js, već i u brojnim bibliotekama i razvojnim okruženjima koje postoje za razvoj web aplikacija. Na primjer, može se koristiti React.js ili Angular za korisničko sučelje, Node.js za backend i React Native za izradu mobilnih aplikacija [22].

JavaScript ima asinkrone karakteristike, što znači da može komunicirati s poslužiteljem u pozadini, bez ometanja korisničke interakcije koja se odvija. Omogućuje osvježavanje web aplikacije bez potrebe za ponovnim učitavanjem, automatsko dovršavanje, animacije elemenata web aplikacije, reprodukcija audio i video zapisa, validaciju obrazaca i još mnogo toga. Također je vrlo prikladna za web zbog uske integracije s DOM-om (engl. *Document Object Model*) [21]. DOM je prikaz podataka objekta koji obuhvaćaju strukturu dokumenata na webu [23].

Jedna od značajnijih značajki JavaScripta je što uključuje programiranje osnovano na događajima, ima ugrađene događaje kao što su „onClick“ i „onHover“ koji promatraju interakciju korisnika s web aplikacijom prije izvršavanja određenog segmenta koda. Na primjer kada pritisnemo gumb koji pokreće noćni način rada, to je događaj koji pokreće dio JavaScript programskog koda koji mijenja CSS cijele aplikacije iz svijetle u tamnu temu [24].

Možemo sa sigurnošću reći da je u ovom trenutku teško nadmašiti ovaj jezik i njegovu uporabu i da će se JavaScript još dugo koristiti za razvoj web aplikacija.

3.4.2 Python

Python je interpreterski objektno-orijentirani programski jezik koji se prvi puta pojavio 1991. godine. Iako postoji već godinama, nedavno je stekao jako veliku popularnost zbog iznimno jednostavne i lako čitljive sintakse. Rješenje problema koje bi zahtijevalo oko 10 redaka koda napisanih u bilo kojem drugom programskom jeziku, u Pythonu bi se lako mogao napisati u 2 do 3 retka [1].

Trenutno je drugi po popularnosti programskih jezika za razvoj web aplikacija. Ima jednostavne skripte i s velikom lakoćom se povezuje s bazom podataka [25]. Ima velik broj razvojnih okvira, biblioteka i alata koji ubrzavaju i uvelike pomažu u stvaranju web aplikacija kao na primjer Django, Flask i Pyramid [20].

Python iznimno olakšava prikazivanje podataka u formatu koji se može lako razumjeti pomoću različitih grafova i grafikona koji su veoma učinkoviti za razumijevanje. Postoji puno Python biblioteka u razvoju web aplikacija koje se koriste za vizualizaciju [26]. Iako je jako jednostavan, iznimno je moćan jezik i omogućuje izvedbu jako složenih zadataka u backendu, uključujući AI (engl. *Artificial intelligence*) i *Data Science* [27].

3.4.3 Java

Java je objektno-orijentirani programski jezik koji se koristi za programiranje na poslužiteljskoj strani. Jedan od najpopularnijih jezika na svijetu općenito, a pogotovo za razvoj web aplikacija.

Mora se kompajlirati, što ju čini bržom od npr. PHP-a i Pythona koji su interpreteri. Doduše njena primarna prednost je što radi na „jednom napišeš, svugdje radi“ (engl. *write once, works everywhere*) principu, odnosno kada se Java jednom kompajlira može se izvršiti na bilo kojoj platformi koja podržava Javu bez potrebe za ponovnim kompajliranjem. Jednom kompajlirani kod vrti se na Java virtualnoj mašini (engl. *Java Virtual Machine*) te joj to daje neovisnost o platformi na kojoj se razvija. Podržava *multithreading* koji omogućuje istodobno izvršavanje dvije ili više radnje istovremeno za maksimalnu iskoristivost CPU-a. Memorija je podijeljena na dva dijela što pomaže zadržati podatke i lako ih vratiti [20]. Postoji mnogo Java platformi za razvoj weba. Ovo su neke od najistaknutijih tehnologija za razvoj Java web aplikacija:

- **Servlet API** - povećava mogućnosti poslužitelja koji se koriste za *hosting* web aplikacije. Web aplikacije razvijene putem Servleta slijede zahtjev-odgovor (engl. *request-response*) model [28].
- **JavaServer Pages (JSP)** - Omogućava umetanje Java koda u HTML pomoću svojih JSP oznaka. JSP stranica sadrži statičke podatke koji se mogu zapisati u HTML-u, WML-u, XML-u itd. Također sadrži elemente tehnologije JSP-a koji upravljaju dinamičkim kretanjem web sadržaja [28].
- **Java Database Connectivity (JDBC) API** – sadrži metode i upite za pristup bazi podataka. Klijenti mogu ažurirati sve podatke u bazi putem web aplikacija koje sadrže JDBC upravljačke programe [28].
- **Java Persistence API (JPA)** – objektno-relacijsko mapiranje za povezivanje objektno-orijentiranog modela s bazom podataka. Pomaže u pohrani ili preuzimanju velike količine podataka u/iz baze podataka [28].
- **JavaServer Faces Tehnology** – razvojno okruženje korisničkog sučelja. Nudi jednostavan model za dizajniranje komponenti u različitim jezicima [28].

Bez konkurencije kad su u pitanju stabilnost i povjerenje, Java nastavlja zauzimati mjesto među vodećim jezicima za razvoj web aplikacija.

3.4.4 PHP

PHP je skriptni objektno-orijentirani programski jezik koji se koristi za programiranje na strani poslužitelja. Njegova namjena je prvenstveno bila za razvoj web stranica i aplikacija. Također je i interpreterski jezik, odnosno ne kompajlira se. Može se izravno pisati unutar HTML koda što mu povećava praktičnost [20].

Objavljen je 1995. godine i koristi ga veliki broj korisnika te iz tog razloga, iako trenutno nije jako popularan, 79,1% svih web stranica koristi upravo PHP za programiranje na strani poslužitelja. Među njima su Facebook, Twitter, Wikipedia i WordPress [29].

Ima veliki skup ugrađenih alata i dodatnih modula, stoga je njegova upotreba velika za razvoj web aplikacija. Jednostavan je, dinamičan i lako se povezuje s bazom podataka [30]. Postoji veliki broj dostupnih biblioteka i razvojnih okruženja koji još više proširuju njegove mogućnosti. Najpopularniji primjeri su: Laravel, Symfony, Phalcon i drugi [29].

Iako je PHP moćan jezik podržan od strane puno korisnika, postoje jednostavniji programski jezici za razvoj web aplikacija. Iz tog razloga programeri koji tek počinju učiti češće radije odaberu Python kao svoj alat.

3.5 BIBLIOTEKE ZA WEB

Biblioteke su grupe unaprijed napisanih isječaka programskog koda koji omogućuju veliku funkcionalnost, a da ne morate sve pisati sami. Brinu se za to da je kod učinkovit i da radi dobro na svim web preglednicima i uređajima. Opisat ćemo dvije najpopularnije [31].

3.5.1 Bootstrap

Bootstrap je biblioteka za razvoj prednjeg dijela (frontend) web aplikacija koja omogućava brži i lakši razvoj. Sadrži predloške dizajna odnosno gotove klase i komponente temeljene na HTML-u i CSS-u za različita responzivna korisnička sučelja, kao i dodatke za JavaScript [32]. Bootstrap klase možemo podijeliti u 3 glavne grupe [33]:

- Klase za stvaranje i manipuliranje mrežom (engl. *grid*) - adaptivan raspored stranice
- Klase za oblikovanje sadržaja - tekst, slike, tablice i slično
- Klase za servis - za rješavanje pomoćnih problema kao što su poravnanje, kontrola prikaza, dodavanje obruba itd.

Uz klase, također sadrži gotove komponentne kao gumbe, breadcrumb, navigacijske izbornike, padajuće izbornike, skočne prozore, itd. Jednostavan i lagan je za učenje kao i korištenje. Ubrzava i pojednostavljuje razvoj prednjeg dijela web aplikacije svojim velikim brojem gotovih predložaka. Mana Bootstrap-a je što su sve web aplikacije razvijene njime međusobno vrlo slične: isti gumbi, navigacija, struktura, itd. Svaka nova web aplikacija previše je slična već mnogim stvorenim, što otežava stvaranje jedinstvenih i drugačijih web aplikacija [33].

3.5.2 JQuery

JQuery je JavaScript biblioteka koja je brza, bogata značajkama i lagana. Glavna svrha joj je znatno olakšati korištenje JavaScripta kod razvoja web aplikacija [34]. Neke od njegovih značajka jesu [35]:

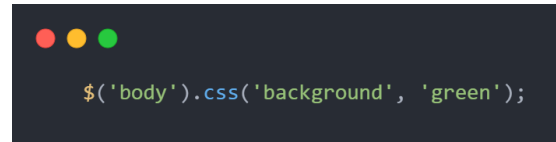
- HTML/DOM manipulacija
- manipulacija CSS-om

- metode HTML događaja
- efekti i animacije
- AJAX



```
function changeColor(color) {
    document.body.style.background = color;
}
Onload = changeColor('green');
```

Slika 3.14 Primjer JavaScript koda (Izvor: autorski rad)



```
$('#body').css('background', 'green');
```

Slika 3.15 Primjer JQuery koda (Izvor: autorski rad)

Puno je jednostavniji od JavaScripta i tim lakši za razumijevanje. Zahtjeva znatno manje napisanog koda od JavaScripta za isti zadatak. Kao što možemo vidjeti na primjerima (slika 3.14 i slika 3.15) JavaScript kod puno je duži i kompliciraniji od jQuery koda. Oba primjera mijenjaju boju pozadine u zeleno, JavaScript to radi u 3 linije koda, jQuery u jednoj. Iz čega možemo zaključiti da jQuery minimizira kod i da je njegova sintaksa jednostavnija za uporabu. To ga automatski čini bržim, efikasnijim i puno boljim za SEO (engl. *search engine optimization*) [34].

3.6 RAZVOJNA OKRUŽENJA

Razvojna okruženja (engl. *framework*) mogu se definirati kao kostur koda odnosno kao grubi nacrt web aplikacije. Razvojno okruženje nam daje gotove bazične funkcije, ali kako bi naš program znao kako obavljati te funkcije ipak mu moramo dati detaljnije upute [1].

Razvojna okruženja postala su jako bitan dio u razvoju web aplikacija. Standardi web aplikacija svakim danom sve više rastu, a time raste i složenost razvoja takvih aplikacija. Razvojna okruženja su tu da olakšaju taj posao, odnosno da olakšaju rad s programskim jezicima i učine proces razvoja i održavanja web aplikacija bržim, efikasnijim i lakšim.

U moru opcija teško je odabrati koji programski jezik i koje razvojno okruženje će najbolje zadovoljiti zahtjeve web aplikacije. U nastavku ćemo opisati neke od trenutno najpopularniji razvojnih okruženja za oboje: prednji i stražnji dio razvoja web aplikacija.

3.6.1 React

React je jedna od najpopularnijih razvojnih okruženja programskog jezika JavaScript za razvoj prednjeg dijela web aplikacija. Razvijena je od strane Facebooka. Radi na način da se kod razvoja korisničkog sučelja, korisničko sučelje dijeli na izolirane komponente. Na taj način svaka komponenta ima svoja svojstva i funkciju, što uvelike olakšava ispravljanje greška i modifikacije u kodu [36].

Prednosti:

- **Virtualni DOM** – Poboljšava iskustvo korisnika kao i rad programera. Pomaže u ažuriranju promjena korisnika bez smetnji drugih dijelova upravo zbog izoliranih komponenti. Odnosno ažurira samo one stavke u stvarnom DOM-u koje su promijenjene, umjesto da ažurira sve komponente [37].
- **Izolirane komponente** – Uvelike štedi vrijeme svojim odvojenim komponentama jer ih programeri mogu ponovno koristiti kad god im zatrebaju što proces razvoja web aplikacija čini znatno bržim [37].
- **Jednosmjerni tok podataka** – Rad sa svakom komponentom zahtjeva protok podataka u jednom smjeru. To rezultira lakšim pronalaskom problema koji se javlja u aplikacija i otklanjanjem istog, što čini kod iznimno stabilnim [37].

Mane:

- **Manjak dokumentacije** – Zbog brzog razvoja i velike popularnosti u kratkom vremenskom periodu ima puno ažuriranja i inovacija, a nisu sve opisane u odgovarajućoj dokumentaciji, što otežava pronalazak svih potrebnih detalja i informacija [37].

3.6.2 Vue

Vue je razvojno okruženje programskog jezika JavaScript koje je prvobitno zamišljeno za izradu web aplikacija s jednom stranicom i korisničkog sučelja. No, s vremenom Vue.js se značajno proširio. Pristupačne značajke, funkcionalnost i jednostavno učenje napravilo ga je jednim od omiljenih alata za razvoj prednjeg dijela web aplikacija [38]. Vrlo brzo su ga usvojile velike tvrtke kao što su: Adobe, Behance, Alibaba, Gitlab i Xiaomi [39].

Baziran je na dvosmjernom reaktivnom vezanju podataka i ne zahtjeva nikakve dodatne biblioteke. Fleksibilan je i može se jednostavno integrirati s različitim bibliotekama i primijeniti za veće projekte.

Prednosti:

- **Jasnoća i jednostavnost** – Jednostavan je za naučiti i koristiti, te time privlači početnike one koji tek započinju svoju karijeru u razvoju web aplikacija [37].
- **Detaljna i opsežna dokumentacija** – Redovno ažurirana i dobro napisana dokumentacija čine korištenje Vuea puno lakšim. Vrlo je dobro strukturirana i pokriva sve teme, od instalacije do složenijih stvari poput skaliranja aplikacije [37].
- **Jednostavna integracija i ponovna upotreba koda** – Zbog svoje logičke strukture, programeri mogu komponente koda ponovno koristiti kao predloške za slične elemente sustava, čak i u drugim projektima [39].
- **Proširenja za razvojni alat preglednika** – Posjeduje puno vrlo korisnih proširenja za preglednike što olakšava razvoj web aplikacija [37].
- **Virtualni DOM** – Jednako kao React, Vue koristi virtualni DOM što pruža bolje korisničko sučelje bez potrebe za ažuriranjem pravog DOM-a [39].

- **Dvosmjerno vezanje podataka** – Dvosmjerno vezivanje podataka Vue je naslijedio od Angulara. Dvosmjerno povezivanje podataka veza je između ažuriranja podataka modela i prikaza odnosno korisničkog sučelja. Omogućava lakše ažuriranje komponenti i praćenje podataka o ažuriranjima [39].

Mane:

- **Pretjerana fleksibilnost** – S jedne strane, fleksibilnost koda pruža velike mogućnosti programerima. No s druge strane, fleksibilnost može dovesti do više nepravilnosti i pogrešaka u kodu jer se nekoliko različitih programskih pristupa može primijeniti istovremeno unutar istog tima odnosno unutar iste web aplikacije [39].
- **Nedostatak skalabilnosti** – U usporedbi s drugim popularnim okruženjima kao što su React i Angular još uvijek je mlad. Ima malenu zajednicu i relativno malu skupinu programera i stoga nije skalabilan i prikladan za velike projekte. Da bi se mogao koristiti u velikim projektima, treba imati stabilnu i snažnu podršku tako da se problemi mogu brzo riješiti [39].
- **Dvosmjerno vezanje podataka** – Iako smo spomenuli dvosmjerno vezanje podataka kao prednost, ono može predstavljati i problem. Dvosmjerno povezivanje ponovno renderira samo one dijelove koda koji su aktivirani i često griješi prilikom čitanja podataka [39].

3.6.3 Angular

Angular je još jedno razvojno okruženje za programski jezik JavaScript razvijen za prednji dio web aplikacija od strane Googlea. Doista je razvojno okruženje, a ne samo veliki paket biblioteka. Programeri se doista mogu usredotočiti na rješavanje svojih zadataka, a ne na traženje raznih biblioteka za rješavanje problema. To je definitivno ono što ga čini jednim od najboljih okruženja za stvaranje korisničkog sučelja. Kreiran je u TypeScriptu, što pridodaje njegovim prednostima kao što su funkcije sa strelicama i sintaksu klasa [37].

Prednosti:

- **Komponente** – Omogućuje stvaranje korisničkog sučelja putem komponenti, odnosno te komponente mogu se ponovo koristiti u aplikaciji. Što ubrzava razvoj i održavanje [37].
- **TypeScript** – Već smo spomenuli da je kreiran u TypeScriptu, znači da se kompajlira u JavaScript. To olakšava proces programiranja što znači da je kod kompajliran prije nego web preglednik uopće učita web aplikaciju [37].
- **Dvosmjerno povezivanje podataka** – Pomaže korisnicima u razmjeni podataka iz komponente u pregled i obrnuto. To smanjuje vrijeme potrebno za razvoj aplikacije jer nije potrebno pisati dodatan kod koji će sinkronizirati komponentu i pregled [40].
- **Rad u velikim timovima** – Stvoren je za rad u velikim timovima. Koristi module tako da svaki dio tima može raditi na svom dijelu koda bez straha da će to narušiti i slomiti tuđi kod, što osigurava kvalitetu [40].

- **Zajednica** – Ima iznimno snažnu i veliku zajednicu programera koja je zaslužna za puno materijala, rasprava i alata koji daju rješenje za gotovo svaki problem [40].

Mane:

- **Težina učenja** – Iako je Angular odličan, puno je teži za naučiti u usporedbi s Vueom ili Reactom. To ga čini nedostupnijim za početnike zbog svoje kompleksnosti [41].
- **Ograničene mogućnosti SEO-a** – Veliki nedostatak kod korištenja Angulara su ograničene SEO mogućnosti i loša pristupačnost za alate pretraživanja, što je danas jako bitno u svijetu Internet oglašavanja [41].

3.6.4 Express

Express je razvojno okruženje za Node.js razvijen za programiranje stražnjeg dijela web aplikacija. Zahvaljujući ogromnoj popularnosti Node.js-a, Express je brzo postao jedan od najboljih razvojnih okruženja za razvoj modernih web aplikacija. Kompatibilan je s drugim okruženjima kao što su Kraken, Sails i Loopback. Minimalan, brz i neograničen. Pruža osnovne funkcionalnosti razvojnog okruženja bez zanemarivanja značajki Node-a, a pri tome poboljšava performanse asinkronog Node.js-a. Prilično je fleksibilan i podržava sve aplikacije kao i REST API-je [42].

Prednosti:

- **Brzi razvoj aplikacija** - Express omogućuje da se programski jezik JavaScript koristi i u prednjem i u stražnjem dijelu web aplikacije. To znači da jedan programer može lakše i brže razvijati obje strane web aplikacije [43].
- **Upravljanje I/O zahtjevima** – Zbog podržavanja I/Q zahtjeva izvrstan je za aplikacije koje obrađuju puno zahtjeva i obavijesti korisnika [44].
- **Jednostavna integracija usluga trećih strana** – Ima razne pakete za rješavanje različitih problema. Minimalistički je i nema strogo utvrđena pravila o tome kako rješavati određene zadatke i koje komponente odabrati. Programer sam odlučuje kako strukturirati aplikaciju jer ne postoji jedan pravi način za to, što može ujedno biti i mana u nekom velikom timu programera. Može se koristiti bilo koji posrednički softver koji odgovara rješenju problema [43].
- **Jednostavan za naučiti** – Većina programera kaže da je jednostavan za učenje i da je jako dobro dokumentiran što uvelike olakšava proces [43].

Mane:

- **Nedostatak standardizacije** - Iako smo to naveli i kao prednost, jer to i je, ova karakteristika može biti i mana. Nema određenu posebnu strukturu i to može biti problem kod razvoja web aplikacija u velikim timovima gdje dolazi do problema kada svaki programer na različiti način razvija funkcionalnosti aplikacija pa su moguće greške u kodu. Zato je jako bitno da se timovi drže iste strukture za cijeli projekt [45].

- **Povratni pozivi** – Povratni pozivi su problem o kojem se najviše raspravlja kod Expressa jer programeri ne razumiju komplicirani stil pisanja [45].

3.6.5 Django

Django je razvojno okruženje za Python programski jezik i služi za razvoj stražnjeg dijela web aplikacija. Daje mogućnost razvoja složenih i skalabilnih web aplikacija. Koristi *Model View Template* (MVT) arhitekturu koja se temelji na skupu najboljih praksi za organizaciju programskog koda [46].

Prednosti:

- **„Baterije uključene“** – Django se predstavlja kao razvojno okruženje „s uključenim baterijama“. U prenesenom značenju to se odnosi na to da dolazi s puno funkcionalnosti koje su spremne za upotrebu, koje možete, ali i ne morate koristiti. Ovisno o potrebama web aplikacije. Umjesto da morate pisati vlastiti kod, trebate samo uključiti pakete koje želite koristiti. Neke od funkcionalnosti jesu [46]:
 - Podrška na više stranica i jezika,
 - AJAX podrška,
 - Besplatni API,
 - Usmjeravanje URL-a,
 - HTTP biblioteke i biblioteke predložaka.
- **Python** – Budući da koristi programski jezik Python, koji je jedan od najjednostavnijih i najpopularnijih programskih jezika danas, preuzima i brojne njegove prednosti [46].
- **Zajednica** – Velika zajednica uvijek pridonosi velikoj pomoći i jednostavnosti pronalaženja rješenja problemu, kao i detaljnoj dokumentaciji [46].
- **Skalabilan** – Većina programera kada razmišlja o odabiru razvojnog okruženja razmišljaju o budućnosti. Zato je skalabilnost prilično bitna, a Django to omogućuje. Odnosno prikladan je i za male kao i za velike projekte odnosno web aplikacije [46].
- **Sučelje administratora** – Stvoriti administratorsko sučelje pomoću Djanga je uistinu jednostavno. Potpuno opremljeno administratorsko sučelje sa samo nekoliko reda koda [46].

Mane:

- **Kompatibilnost unazad** – S vremenom postaje sve veći i teži, što dovodi do problema s brzinom. Unatoč tome Django izričito više brine o produktivnosti razvoja i kompatibilnosti unazad nego o brzini [46].
- **Obraduje jedan po jedan zahtjev** - Ne može se koristiti za razvoj web aplikacija u stvarnom vremenu jer je temeljen na WSGI-u koji je sinkronizirani protokol. WSGI poslužitelj može istovremeno obraditi samo jedan zahtjev [46].

- **Regularni izrazi za navođenje URL-a** – Možete stvoriti jednostavne i lako čitljive URL-ove. Međutim, koristi regularne izraze za određivanje URL-a što čini kod većim i stvara komplekse sintakse, a to je pogotovo nepogodno za početnike [46].
- **Monolitna** – Gura vas u zadane okvire. Postoji jedan popularan i preporučan način rada i treba ga se pridržavati, što može dati osjećaj monolitnosti. Puno je interesantnije samostalno odabrati strukturu i dizajn elemenata [46].

3.6.6 Laravel

Laravel je najpoznatije razvojno okruženje programskog jezika PHP koja slijedi *Model View Controller* ili MVC model. Posjeduje pristojnu količinu paketa koji bi mogli proširiti njegovu efikasnost. Postoji web stranica Laracasts s priručnicima i videozapisima o PHP-u, Laravelu i o svim tehnologijama u Laravel ekosustavu koja je od velike pomoći, pogotovo za početnike. Što se performansa tiče, ne može se usporediti s Django ili Expressom, što je nedostatak, pogotovo za velike projekte [47].

Prednosti:

- **Najnovije značajke PHP-a** – Velika prednost Laravela pred drugim razvojnim okruženjima programskog jezika PHP je ta što koristi sve najnovije značajke tog jezika kao što su: sučelja, anonimne funkcije, kraća sintaksna polja, itd [47].
- **Dokumentacija** – Sve verzije Laravela imaju odgovarajuću detaljno napisanu dokumentaciju u kojoj se mogu pronaći dobra objašnjenja klasa, stil koda i metoda. Kao što smo već spomenuli, dobro napisana dokumentacija jako je bitna programerima koji koriste razvojna okruženja u svom projektu jer im uvelike olakšava pronalazak rješenja problema [47].
- **Integracija s e-mail uslugama** – Vrlo je bitno da web aplikacije mogu slati korisnicima nove informacije, ponude i sl. Laravel je opremljen API-jem koji nudi upravljačke programe: SMTP, Mailgun, SparkPost, MAndrill, PHP-ovu funkciju „e-mail-a“, Amazon SES koji pomažu u slanju e-pošte putem lokalnih usluga ili u oblaku [48].
- **Priručna memorija** – Pruža podršku za priručnu memoriju putem ugrađenih upravljačkih programa, osiguravajući da su objekti pohranjeni u unutarnjem sustavu datoteka. To donosi prednosti kao što su: brzina izvršavanja zadataka, napredno upravljanje memorijom i poboljšane performanse [47].
- **MVC arhitektura** – MVC arhitektura je odgovorna za podjelu između logike i sintakse. To programerima smanjuje vrijeme koje je potrebno za dodavanje novih značajki i ispravljanje grešaka. Lako je promijeniti korisničko sučelje bez da se ometa osnovne funkcije [48].
- **Usmjeravanje URL-a** – Pojednostavljeno je usmjeravanje URL-a. Moguće je stvoriti veze do imenovanih ruta što smanjuje rizik od preusmjeravanja praznog URL-a [48].

Mane:

- **Nema ugrađenu funkciju podrške** – Laravel je dosta mali okvir, za razliku od Djanga ili Ruby on Railsa te ima ograničeno ugrađenu podršku [47].
- **Problematične nadogradnje** – Dosta često se javlja problem u nadogradnjama verzija Laravela [47].
- **Spori razvoj aplikacija**– u usporedbi s drugim razvojnim okruženjima, razvoj web aplikacije u Laravelu je znatno sporiji [47].
- **Zajednica** – Kao dosta novo razvojno okruženje, nema veliku zajednicu. Također iako se može činiti jednostavnim na prvu, to baš i nije. Iz tog razloga ima malo visokokvalificiranih stručnjaka za Laravel što otežava rad s njim na velikim projektima, pogotovo u usporedbi s drugim većim i starijim razvojnim okruženjima [48].

3.7 BAZE PODATAKA

Sve podatke koji se razmjenjuju na webu potrebno je negdje spremiti. U tu svrhu neizbježno je da web aplikacije imaju svoje baze podataka. U nastavku ćemo detaljnije opisati neke najpopularnije baze podataka, koje se koriste u svrhu razvoja web aplikacija, podijeljene u dvije kategorije: relacijske i nerelacijske baze podataka.

Relacijska baza podataka je skup informacija pohranjenih u tablicama koje su međusobno povezane vezama odnosno relacijama. Relacijske baze podataka komuniciraju pomoću **SQL** (engl. *Structured Query Language*) jezika. Jako su moćne i sigurne, ali mogu biti malo ograničene jer zahtijevaju određenu strukturiranu shemu podataka. Detaljnije ćemo pregledati i usporediti dva najpopularnija sustava za upravljanje relacijskim bazama podataka: MySQL i PostgreSQL (slika 3.16). A još neki koji se koriste jesu: Microsoft SQL Server, Oracle, MariaDB, itd. [49].



MySQL



PostgreSQL

Slika 3.16 Sustavi za upravljanje relacijskim bazama podataka (Izvor: autorski rad)

Nerelacijske baze podataka (često zvane **NoSQL** baze podataka) razlikuju se od tradicionalnih relacijskih baza podataka po tome što ne pohranjuju podatke u obliku tablica, odnosno ne postoji određena shema, te podaci nisu nužno strukturirani. Radi toga dolazi do puno različitih vrsta nerelacijskih baza podataka poput dokumentnog modela, ključ-vrijednost modela, graf modela, stupčanog modela, itd. Dokumentni model, na primjer, može biti vrlo detaljan, a sadržavati niz

različitih vrsta informacija u različitim oblicima. Sposobnost uspoređivanja i organiziranja različitih vrsta podataka čini nerelacijske baze podataka mnogo fleksibilnijima od relacijskih.

Pojavljaju se iz potrebe za organizacijom velikih količina podataka koji su raznoliki. Često su brže od relacijskih jer upit ne mora prolaziti kroz nekoliko tablica da bi dao odgovor. To ih čini idealnim za pohranu podataka koji se često mijenjaju ili za aplikacije koje obrađuju velik broj podataka različite vrste. Detaljnije ćemo pregledati i usporediti dva najpopularnija sustava za upravljanje nerelacijskim bazama podataka, a to su: MongoDB i Firebase (slika 3.17). A drugi koji se još koriste su: CouchDB, OrientDB, Cassandra, itd. [50].



Slika 3.17 Sustavi za upravljanje ne relacijskim bazama podataka (Izvor: autorski rad)

3.7.1 MySQL

MySQL je jedan od najpopularnijih sustava za upravljanje bazama podataka. Razvijen je od strane Oraclea. Temelji se na sustavu klijent/poslužitelj koji se sastoji od višenitnog SQL servera koji podržava nekoliko različitih klijentskih programa i biblioteka, administrativne alate i širok raspon API-ja. Napisan je C i C++ programskim jezikom [51].

Brz je, skalabilan i jednostavan za korištenje u usporedbi s ostalim sustavima za upravljanje bazama podataka kao što su Microsoft SQL Server i Oracle. Obično se koristi zajedno s PHP skriptama za stvaranje dinamičkih poslovnih aplikacija na strani poslužitelja. Ima jako puno funkcionalnosti a neke od njih su [51]:

- Mogućnost oporavka,
- Podržava programske jezike poput Perla, Pythona i PHP-a,
- Omogućuje veličinu baze podataka s približno 50 milijuna redaka,
- Podržava intuitivno grafičke sučelje,
- Pruža sigurnost, fleksibilnost i pouzdanost,
- Podržava replikaciju i transakcije.

3.7.2 PostgreSQL

PostgreSQL je objektno-relacijski sustav baze podataka koji uključuje neke ključne značajke poput pouzdanosti, robusnosti i značajki i performansi. Koristi i proširuje SQL jezik u

kombinaciji s mnogim značajkama koje sigurno pohranjuju i skaliraju najkompliciranija radna opterećenja podataka. Održava integritet podataka i skalabilna je. Može komunicirati s više poslužitelja. Neke od najbitnijih funkcionalnosti PostgreSQL sustava su [51]:

- Velika funkcionalnost,
- Stvaranje sustava otpornog na greške,
- Pruža zaštitu podataka,
- Podržava JSON oblik spremanja podataka,
- Dobra sigurnost i mehanizam oporavka,
- Prilagođene metode upita i tipova podataka

3.7.3 MongoDB

MongoDB je NoSQL sustav za upravljanje bazama podataka koji za pohranu i obradu različitih vrsta podataka koristi dokumentni model nerelacijskih baza podataka. Odnosno baza podataka sadrži različite kolekcije slično kao što MySQL baza podataka sadrži tablice. Unutar kolekcija nalaze se dokumenti koji sadrže podatke koje želimo spremiti. Jedna kolekcija može sadržavati više dokumenata te kako nerelacijske baze podataka nemaju određenu krutu shemu, nije nužno da su podaci u dokumentima slični jedan drugome. Dokumenti pohranjuju i dohvaćaju podatke kao ključ-vrijednost (engl. *key-value*) parove [52].

Dokumenti su pohranjeni u JSON ili XML formatu i dozvoljava se pohranjivanje ugniježđenih podataka. To omogućuje stvaranje kompleksnih veza između podataka koji se mogu spremiti u isti dokument što olakšava rad sa podacima i dohvaćanje istih te čini cijelu bazu podataka puno efikasnijom od relacijske baze podataka. Neke od funkcionalnosti MongoDB sustava su [51]:

- Nepostojanje sheme,
- Pruža veliku fleksibilnost podataka spremljenim u dokumentima,
- Sadrži heterogene podatke,
- Pruža visoke performanse, dostupnost i skalabilnost.

3.7.4 Firebase

Firebase je platforma za razvoj mobilnih i web aplikacija koja pruža mnoštvo alata i usluga koje pomažu u razvoju aplikacija. Jedna od tih usluga je Firebase Realtime baza podataka. Riječ je o nerelacijskoj bazi podataka koja se nalazi u oblaku i omogućuje pohranu i sinkronizaciju podataka u stvarnom vremenu. Baza podataka u stvarnom vremenu je jedan veliki JSON objekt kojim programeri mogu upravljati u stvarnom vremenu. Sinkronizacija u stvarnom vremenu svakom korisniku omogućuje pristup podacima s bilo kojeg uređaja, te olakšava međusobnu suradnju korisnika .

Dokumentna je, odnosno podaci se isto kao i kod MongoDB sustava pohranjuju u dokumentima u ključ-vrijednost formatu, a ti se dokumenti pohranjuju u raznim kolekcijama.

Jedna kolekcija može imati više dokumenta, kao i jedan dokument više podataka, te dokumenti ne moraju sadržavati slične podatke zbog nepostojanja krute sheme [53].

4 USPOREDBA I DISKUSIJA

U poglavlju koje slijedi, međusobno ćemo usporediti web tehnologije za razvoj weba koje smo prethodno naveli i opisali neke njihove glavne značajke. Kod frontenda imamo samo jedan programski jezik (JavaScript), te ga nemamo s čime usporediti. Usporedit ćemo backend programske jezike: PHP, Java i Python. Zatim ćemo usporediti zasebno razvojna okruženja koja se koriste za razvoj stražnjeg dijela web aplikacije (backend): Express, Django i Laravel, a zasebno razvojna okruženja koja se koriste za razvoj prednjeg dijela web aplikacije (frontend): Angular, React i Vue. I na kraju, usporedit ćemo sustave za upravljanje relacijskim bazama podataka: MySQL i PostgreSQL te zasebno sustave za upravljanje nerelacijskim bazama podataka: MongoDB i Firebase.

4.1 USPOREDBA PROGRAMSKIH JEZIKA ZA BACKEND

1. Usporedba prema jednostavnosti/složenosti korištenja

Python i PHP su dinamički jezici, dok je Java statički jezik. Kada se programira u Javi, moraju se jasno definirati tipovi varijabla jer prije izvršavanja programa kompajler provjerava točnost tipova varijabli. Dok kod dinamičkih jezika kao što su PHP i Python definiranje tipova varijabli u kodu nije potrebno - interpreter će ih sam prepoznati i provjeriti njegovu točnost za vrijeme izvršavanja programa. To znači da PHP i Python zahtijevaju manje pisanja koda od Jave te to povećava njihovu jednostavnost.

U Javi i PHP-u koriste se vitičaste zagrade kojima se označava klasa, funkcija, petlja, itd. Python se po tome razlikuje jer ne koristi zagrade za označavanje početka i kraja, nego to radi uvlačenjem koda. Nekima je jednostavnije čitanje koda preko vitičastih zagrada, nekima je jednostavnije čitanje Pythonove sintakse [54] [55].

```
1 print("Hello World")
```

Slika 4.2 Primjer programskog koda u jeziku Python
(Izvor: autorski rad)

```
1 <?php echo "Hello, World";
```

Slika 4.1 Primjer programskog koda u jeziku PHP
(Izvor: autorski rad)

```
1 class HelloWorldApp {
2     public static void main(String[] args) {
3         System.out.println("Hello World!");
4     }
5 }
```

Slika 4.3 Primjer programskog koda u jeziku Java (Izvor: autorski rad)

Na slikama 4.1, 4.2 i 4.3 je prikazan programski kod koji ispisuje „*Hello, World*“ na sva tri programska jezika [56], dok na slikama 4.5, 4.6 i 4.7 možemo vidjeti primjer programskog koda za petlju koja ispisuje brojeve unazad [57]. Iz programskog koda prikazanog na slikama možemo zaključiti da Python ima daleko najjednostavniju sintaksu i zahtjeva najmanje napisanog programskog koda. Ne zahtjeva nikakve vitičaste zagrade, niti znak točka-zarez kao ni definiranje tipova varijabli. To ga čini vrlo efikasnim i najjednostavnijim programskim jezikom za razvoj stražnje strane web aplikacija. PHP je negdje između po kompleksnosti. Zahtjeva da se ispred svake varijable piše prefiks \$, ima vitičaste zagrade i sama sintaksa je sličnija Javi nego Pythonu. Dok je Java vidno najzahtjevnija i kompleksnija od ostalih. Zahtjeva najviše napisanog koda, što usporava razvoj same web aplikacije. Možda je za neke preglednija i detaljnija, ali je najsloženija za korištenje u usporedbi s Pythonom i PHP-om.

```
1 for k in range(10,0,-1):
2     print k
```

Slika 4.4 Primjer programskog koda u jeziku Python (Izvor: autorski rad)

```
1 <?php
2 for ($k=10; $k>=1; $k--) {
3     print ("$k\n");
4 }
5 ?>
```

Slika 4.5 Primjer programskog koda u jeziku PHP (Izvor: autorski rad)

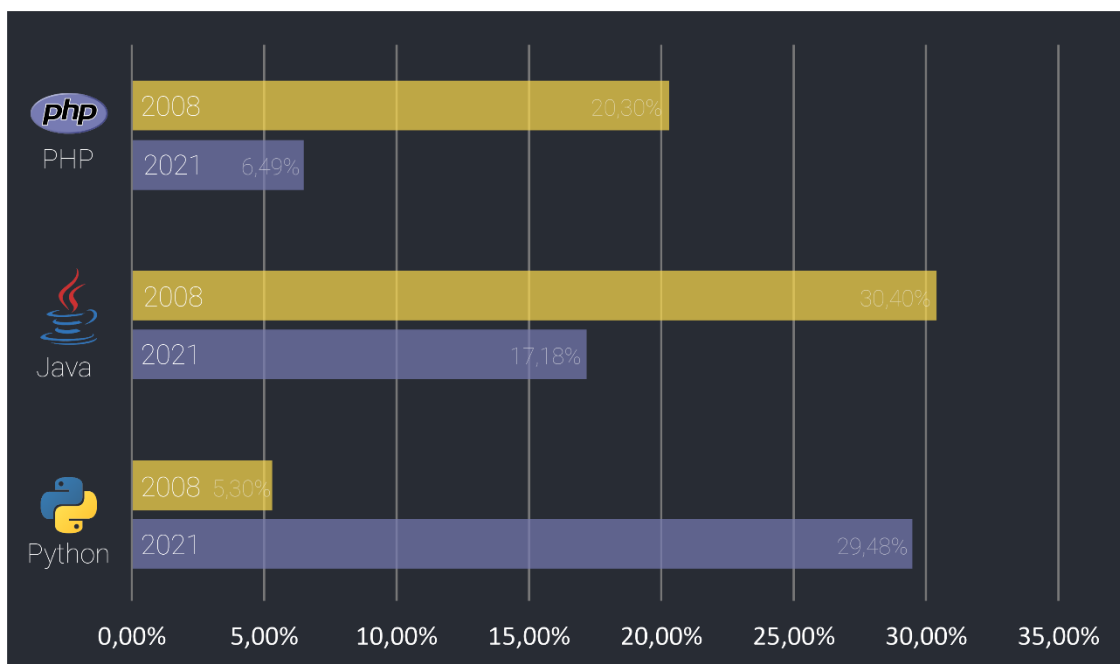
```
1 public class revl {
2     public static void main(String [] args) {
3         int k;
4         for (k=10; k>=1; k--) {
5             System.out.println(k);
6         }
7     }
```

Slika 4.6 Primjer programskog koda u jeziku Java (Izvor: autorski rad)

2. Usporedba prema popularnosti

Što se popularnosti tiče, prema PYPL (engl. *Popularity of Programming Language*) indeksu [58] Python je trenutno najpopularniji programski jezik u cijelom svijetu. PYPL indeks se određuje prema Google pretraživanju pojedinog jezika. Na drugom mjestu nalazi se Java. U svijetu razvoja web aplikacija Java i Python dugo dijele prvo mjesto, ali Python sve više dobiva na popularnosti upravo zbog svoje jednostavnosti. Lakoća korištenja i učenja ga čini vrlo privlačnim za početnike, te sve više novih programera odabiru Python kao svoj alat za razvoj web aplikacija. Iako je PHP najkorišteniji jezik na webu, sve je manje popularan među programerima. 2008. godine PHP-ov postotak prelazio je 20,30%, dok 2021. godine iznosi 6,49%. Dok je, na primjer, Java 2008. godine imala postotak veći od 30%, a sada je na 17,18%, odnosno Java je ipak održala svoju popularnost tokom godina puno više od PHP-a. Pythonov

postotak u 2021. godini iznosi 29,48%, dok je u 2008. iznosio 5,30%. Grafički prikaz usporedbe postotka iz 2021. i 2008. godine možemo vidjeti na slici 4.7.



Slika 4.7 Grafički prikaz PYPL indexa za 2021. i 2008. godinu (Izvor: autorski rad)

3. Usporedba prema podršci za biblioteke

Python i Java značajno nadjačavaju PHP u ovoj kategoriji. Python i Java imaju iznimno dobro razvijene biblioteke za gotovo sve vrste aplikacija. PHP dosta zaostaje u aspektu biblioteka.

Mnoge organizacije razvijaju web aplikacije koje podržavaju strojno učenje, a Python nudi izvrsne biblioteke za to poput TensorFlow, Keras, Theano, Scikit Learn itd. Ove biblioteke su brze, jednostavne za korištenje i sjajno se integriraju s razvojnim okruženjima za web. Zbog postojanja raznovrsnih biblioteka, razvoj web aplikacija pomoću Pythona jednostavniji je u usporedbi s PHP-om. Java također ima velik broj biblioteka koje znatno olakšavaju razvoj web aplikacija, te je jako teško odrediti jesu li one bolje i raznovrsnije od Pythonovih biblioteka, ali sa sigurnošću možemo reći da su puno moćnije od PHP-ovih [54] [55].

4.2 USPOREDBA RAZVOJNIH OKRUŽENJA ZA FRONTEND

1. Usporedba prema jednostavnosti/složenosti korištenja

Angular je dosta zahtjevan za savladati i zahtjeva povezivanje koncepta poput TypeScript-a i MVC-a, te zna imati dosta kompleksnu sintaksu. Ima najsloženiju strukturu projekta, a budući da je potpuno razvijeno frontend okruženje, oslanja se na više koncepta [59] [60].

React je najjednostavniji i najbrži za postavljanje, morate uključiti samo jednu biblioteku i već možete krenuti pisati svoju React aplikaciju sa samo par linija koda. Sintaksa mu nije pretjerano

komplicirana i ima vrlo dobru dokumentaciju u kojoj je sve detaljno objašnjeno što olakšava razvoj web aplikacije. Angular i Vue se pišu s HTML-om, dok se u Reactu mora pisati JSX, što je sintaksa koja omogućuje suživot između HTML-a i JavaScripta. JSX sintaksa čini React malo složenijim za korištenje [59] [60].

Vue je malo kompleksniji za postaviti od Reacta. Pruža veću prilagodljivost, pa ga je lakše naučiti nego Angular i React. Cijela ideja Vue razvojnog okruženja je da uzme najbolje od oba svijeta, odnosno najbolje funkcionalnosti od Angulara i Reacta i spoji ih u jedno. To znači da ima puno istih funkcionalnosti, kao, primjerice, upotreba komponenta, te je zapravo prijelaz s njih na Vue prilično lagan. Iako je Vue jako prilagodljiv i jednostavan za korištenje, to može vrlo lako rezultirati lošim kodom, težim ispravljanjem grešaka i kompliciranim testiranjem [59] [60].

Kao zaključak možemo reći da je Angular najsloženiji za korištenje, dok je Vue najjednostavniji. Ali treba biti oprezan sa jednostavnošću Vuea jer može rezultirati loše napisanim kodom i znatno otežati održavanje same web aplikacije [59] [60].

2.Usporedba prema popularnosti

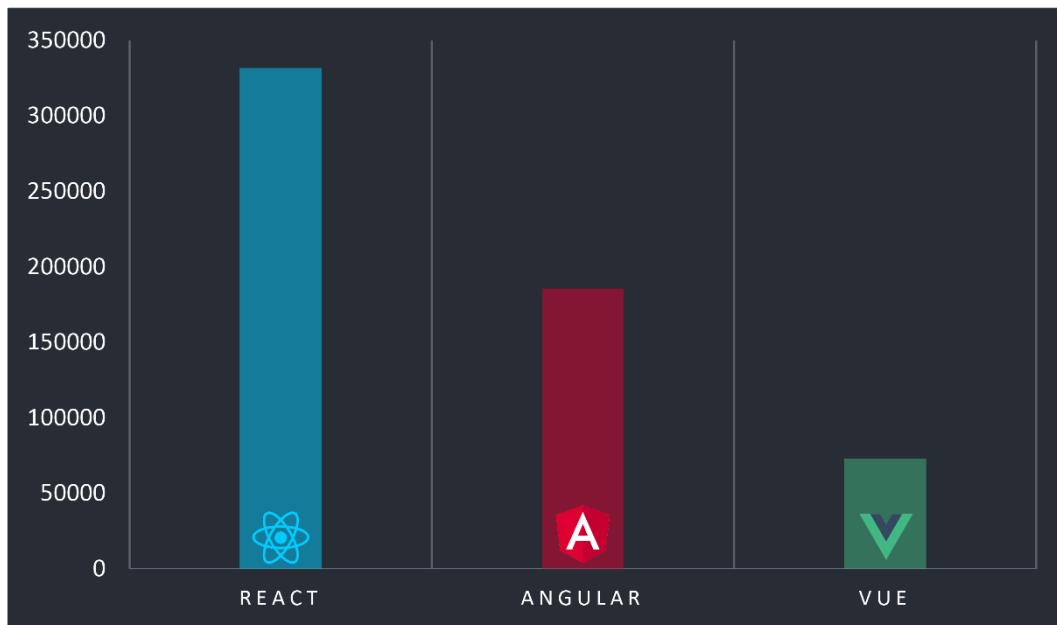
Popularnost razvojnih okruženja za frontend ćemo prvo usporediti prema statistici njihovih repozitorijima na GitHubu [61] [62] [63].

	 Angular	 Vue	 React
 Watch	3.2k	6.2k	6.7k
 Star	76.3k	188k	174k
 Fork	20k	30.1k	35.1k

Slika 4.8 Prikaz Statistike frontend razvojnih okruženja sa stranica GitHub (Izvor: autorski rad)

Kao što možemo vidjeti na slici 4.8, Vue i React imaju jako velik broj gledatelja, zvjezdica i forkova. Angular prema ovim brojkama malo kaska za njima. Vue je najviše ljudi označilo sa zvjezdicom na GitHubu što pokazuje veliku popularnost Vuea među korisnicima i njegovu trenutno vrijednost i zainteresiranost programera za ovo razvojno okruženje. React mu je također vrlo blizu. To je stanje popularnosti na stranicama GitHuba. Drugi izvor za usporedbu popularnosti biti će stranica LinkedIn [64], odnosno trenutni broj oglasa za posao koji traže programere sa znanjem nekog od ovih razvojnih okruženja

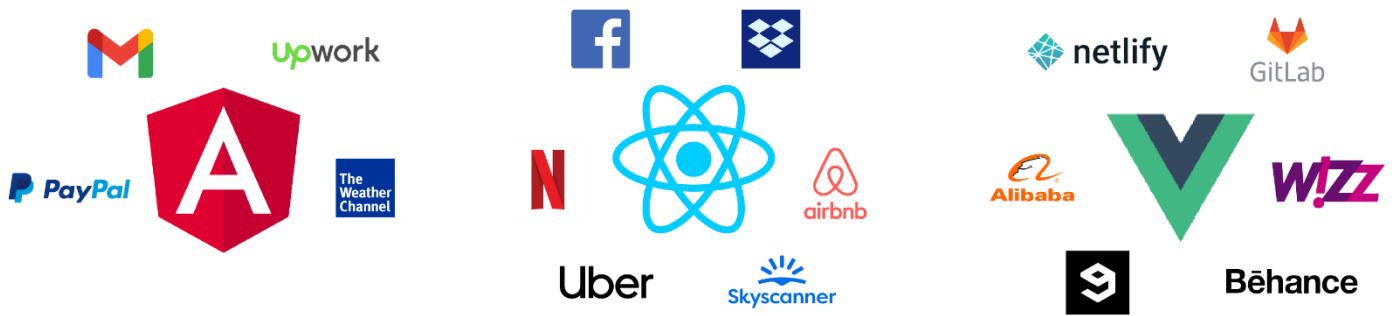
Kao što vidimo na slici 4.9 ovdje je potpuno drugačija situacija. Kompanije daleko najviše traže React developere, a najmanje traženo razvojno okruženje je Vue. Tako da, iako je Vue možda trenutno najpopularniji među korisnicima, React je najpopularniji na tržištu. I iako Angular nije možda toliko popularan trenutno među korisnicima, definitivno je traženiji na tržištu od Vuea.



Slika 4.9 Grafički prikaz statistike pretraživanja frontend razvojnih okruženja na stranici LinkedIn (Izvor: autorski rad)

3.Usporedba prema tipu aplikacije koji se izrađuje

Angular se koristi kod velikih i složenih web aplikacija. Iako se s njim može razvijati neka manja i jednostavnija aplikacija, zna biti dosta „težak“ za takve projekte i tu se bolje uklapaju Vue ili React. Idealan je za web aplikacije s dinamičkim sadržajem, gdje se sadržaj dinamički mijenja ovisno o ponašanju korisnika i njegovim preferencijama. Najbolja je opcija za pouzdanu skalabilnost zbog svojeg modularnog sastava razvoja. React je također dobra opcija za skalabilne projekte zbog svoje metode zasnovane na komponentama. Što se tiče skalabilnosti, Vue malo zaostaje zbog svoje sintakse zasnovane na predlošcima. Vue se koristi za manje i jednostavnije aplikacije kojima su potrebne visoke performanse i brz razvoj. React se koristi za aplikacije s puno komponenti s različitim i često promjenjivim stanjima kao aktivne/neaktivne navigacijske stavke, dinamički ulazi, korisnička prijava i dopuštenja itd. [59] [60].



Slika 4.10 Prikaz aplikacija razvijenih s pojedinim frontend razvojnim okruženjem (Izvor: autorski rad)

Na slici 4.10 su prikazane neke najpoznatije web aplikacije razvijene s pojedinim razvojnim okruženjem. Angularom su razvijene poznate web aplikacije kao PayPal, Gmail, Upwork i Weather.com. Kod Reacta imamo naravno Facebook, koji koristi u nekim dijelovima malo i Vue-a, Dropbox, Netflix, Airbnb, Uber i Skyscanner. Vue-om su razvijeni GitLab, Wizz air, Behance, 9gag, Alibaba i Netlify.

Možemo zaključiti da su sve ove web aplikacije raznovrsne i da sva tri razvojna okruženja imaju svoje prednosti i nedostatke. S razlogom su najpopularniji, jer su najbolji na tržištu, te će svako odabrati ono razvojno okruženje koje najbolje ispunjava njegove individualne zahtjeve.

4.3 USPOREDBA RAZVOJNIH OKRUŽENJA ZA BACKEND

1. Usporedba prema jednostavnosti/složenosti korištenja

Express, Django i Laravel prilično su jednostavni, elegantni i imaju lako razumljivu sintaksu. Svo troje uključuju ogroman broj izuzetnih alata za razvoj web aplikacija i brojna rješenja koja se mogu stvoriti sa samo nekoliko linija koda. Jednostavnost korištenja ovisi o programeru i njegovim preferencijama, te ponajviše o preferencijama programskog jezika kojim su napisani. Iako je Django razvojno okruženje za programski jezik Python, za koji smo rekli da je najjednostavniji za korištenje i ima najjednostavniju sintaksu. Ono što Django čini složenijim i kompliciranijim za korištenje je usmjeravanje. Django za usmjeravanje koristi regularne izraze, čije korištenje može biti složeno.







Također, iako Express ne zahtjeva specifičnu arhitekturu, uobičajeno je da se koristi MVC (engl. *Model View Controller*) arhitektura. Istu tu arhitekturu koristi i Laravel razvojni okvir. Dok Django prati MVT (engl. *Model View Template*) arhitekturu, koja je mala varijacija široko rasprostranjene MVC arhitekture. Značajna razlika je što datoteka predloška koja sadrži HTML i Django predloške obavlja posao kontrolora olakšavajući stvaranje dinamičkih web aplikacija. Slijedom toga, postaje lagano upravljati scenarijima u kojima bi svaki korisnik trebao imati prilagođeni sadržaj, kao na primjer na Instagramu [65].

Unatoč činjenici da Laravel pruža veliki broj alata i biblioteka, velik broj tih alata i biblioteka neće se koristiti u projektima. Ti projekti su konfigurirani i spremni za uporabu što je jako pozitivno jer ne moramo pronalaziti i instalirati pakete za nekakve opće potrebe, i ubrzava razvoj

same web aplikacije i pojednostavljuje korištenje. Loša strana je što dobivamo jako težak projekt, sa puno paketa od kojih nam neki vjerojatno nikada neće trebati. Dok s druge strane kod Expressa i Djanga, ukoliko nam treba neki paket, možemo ga jednostavno instalirati. To je pozitivno jer nemamo puno teških paketa u projektu koje nećemo koristiti, ali i mana jer ipak treba odvojiti vrijeme za pronalazak, instalaciju i konfiguraciju pojedinih paketa i biblioteka što zna usporiti i otežati sam razvoj web aplikacije [66].

2.Usporedba prema popularnosti

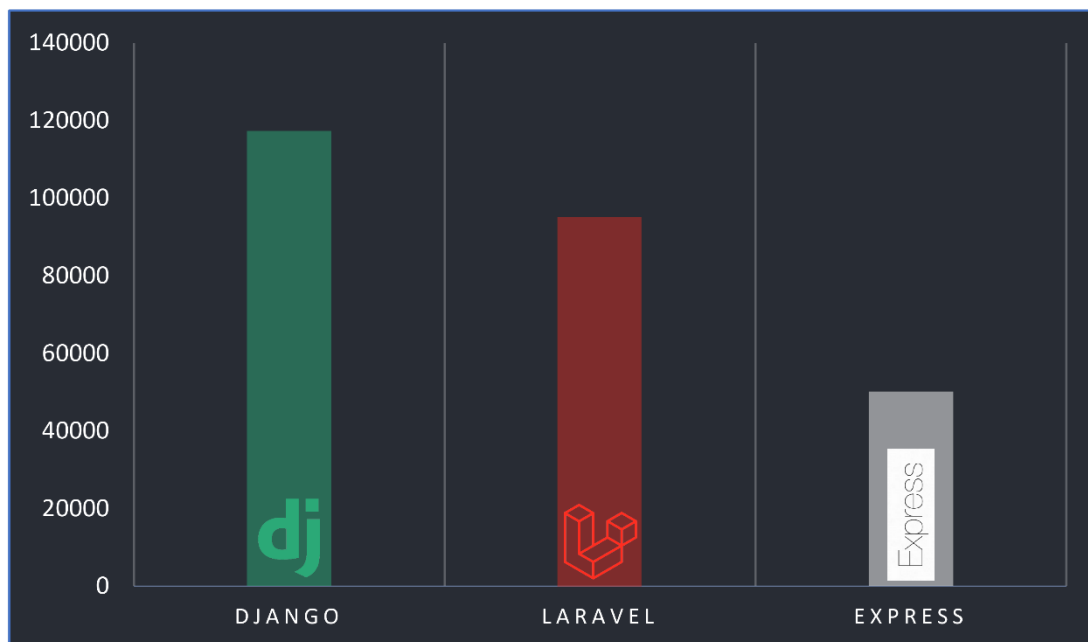
Kao i kod razvojnih okruženja prvo ćemo usporediti popularnost na temelju njihovih repozitorija na GitHubu [67] [68] [69].

	 Express	 Laravel	 Django
 Watch	1.8k	4.7k	2.3k
 Star	54.4k	66.5k	59.6k
 Fork	9.2k	21.5k	25.4k

Slika 4.11 Prikaz statistike backend razvojnih okruženja sa stranica GitHub (Izvor: autorski rad)

Na slici 4.11 možemo vidjeti da im je popularnost među korisnicima, odnosno na GitHub stranicama poprilično jednaka. Laravel je malo ispred ostale dvojice, nakon njega dolazi Django i Express je zadnji. Možemo također zaključiti da su njihove brojke znatno manje od razvojnih okruženja za razvoj prednjeg dijela web aplikacija, te da nisu trenutno toliko popularni među korisnicima.

Dalje ćemo ih usporediti prema broju oglasa za posao na LinkedIn stranicama [64].



Slika 4.12 Grafički prikaz statistike pretraživanja backend razvojnih okruženja na stranici LinkedIn (Izvor: autorski rad)

Prema slici 4.12 možemo vidjeti da su Django i Laravel dosta blizu jedan drugome, dok Express kaska za njima. Django je prema broju oglasa koji traže programere u ovim razvojnim okruženjima pobjednik, odnosno najtraženiji je trenutno na tržištu. Express je najmanje tražen u usporedbi sa Django i Laravelom. Isto tako možemo primijetiti da ni u ovom slučaju brojke nisu onoliko visoke kao kod razvojnih okruženja za razvoj prednjeg dijela web aplikacija.

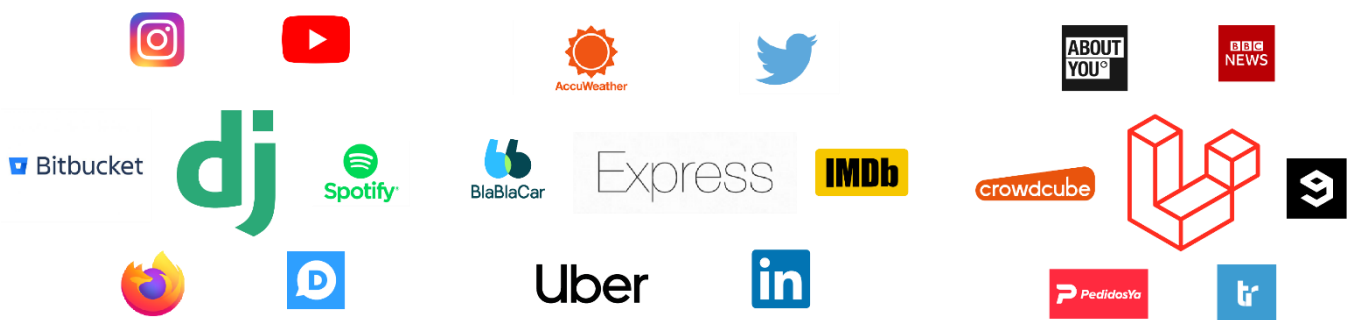
3.Usporedba prema tipu aplikacije koji se izrađuje

Laravel je dobar sa korištenje kod aplikacija temeljenih na SaaS-u ili aplikacija za e-učenje. Također je dobar za web aplikacije s neakvim nagradama i značajkama prepoznavanja ili neakvih web aplikacija koje samostalno prate performanse web stranica. Puno je prikladniji i lakši za razvijanje manjih aplikacija od npr. Djanga, koji zauzima puno *bandwidth-a* i vrijeme obrade poslužitelja. S druge strane, to ga čini jako skalabilnim za razliku od Laravela [70].

Već smo spomenuli da je Django jako pogodan za društvene mreže zbog svoje arhitekture zbog koje lakše implementira prilagođeni sadržaj korisnicima i dijeljenje bilo kakvog medijskog sadržaja. Također, Django je jako prikladan za izradu web aplikacija koje zahtijevaju veliku sigurnost koja štiti transakcije i osjetljive podatke kao na primjer Internet trgovina, jer prema zadanim postavkama skriva izvorni kod i jako dobro reagira na ranjivost u kodu. Kod Internet trgovine ili nekog drugog tipa web aplikacija koja ima velik broj rezervacija, još jedna prednost Djanga je izrada CRMA-a (kombinacija strategije i tehnologije za izgradnju i poboljšanje poslovnih odnosa s kupcima [71]). Iako je sjajan temelj za izgradnju velikih projekata, njegova teška i već spomenuta monolitna struktura može biti smetnja programerima za izradu prilagodljive i brže web aplikacije. Dok je Expressova nestrukturirana arhitektura odlična za razvoj prilagođenih web aplikacija.

Express je prikladan za web aplikacije kod kojih korisnici dijele i izvoz slika kao na primjer Unsplash. Također, web aplikacije poput GoDaddy za registriranje domena i ponude web hostinga ili kao Coursera web aplikacija koja nudi online tečajevi. Express je kao i Django dobar kod web aplikacija kojima treba praćenja posjetitelja i vizualizacije njihovih interakcija u stvarnom vremenu [66] [70].

Na slici 4.13 vidimo neke od najpopularnijih aplikacija čiji je stražnji dio web aplikacije razvijen određenim razvojnim okruženjem. Kod Djanga su to aplikacije Instagram, YouTube, web preglednik Mozilla Firefox, itd. Express razvojnim okruženjem su razvijene web aplikacije: Twitter, BlaBlaCar, Uber, LinkedIn, IMDb, itd. Kod Laravel razvojnog okruženja imamo BBC News, About You online trgovina, 9gag, itd.



Slika 4.13 Prikaz aplikacija razvijenih pojedinim backend razvojnim okruženjem (Izvor: autorski rad)

4.4 USPOREDBA RELACIJSKIH BAZA PODATAKA

1. Usporedba prema jednostavnosti/složenosti korištenja

PostgreSQL je baza podataka jednostavna za korištenje sa svojim nizom značajki. Instalacija je jednostavna. Isto tako je i s MySQL bazom podataka. Poznata je po jednostavnosti korištenja. Instalacija i postavljanje MySQL okruženja je jednostavna u svim operacijskim sustavima. Što se tiče jednostavnosti sintakse, prilično su slične. Jedina mala mana MySQLa je da ne podržava sve SQL sintakse, kao PostgreSQL. Što može kod složenijih baza podataka rezultirati složenijom SQL sintaksom. S druge strane, PostgreSQL je potpuno SQL kompatibilna baza podataka i podržava sve standardne značajke, što ga kod složenijih baza podataka čini jednostavnijim za korištenje.

Oboje, PostgreSQL i MySQL, ne samo da podržavaju tradicionalne tipove podataka kao numeričke, nizove, datum itd., već podržavaju i nestrukturirane vrste podataka kao JSON, XML i HSTORE. Ono što PostgreSQL razlikuje još od MySQL je podrška za širi raspon vrsta podataka kao što su nizovi, mreže i geometrijske tipove podataka što također olakšava rad s kompleksnim podacima u stvarnom vremenu [72].

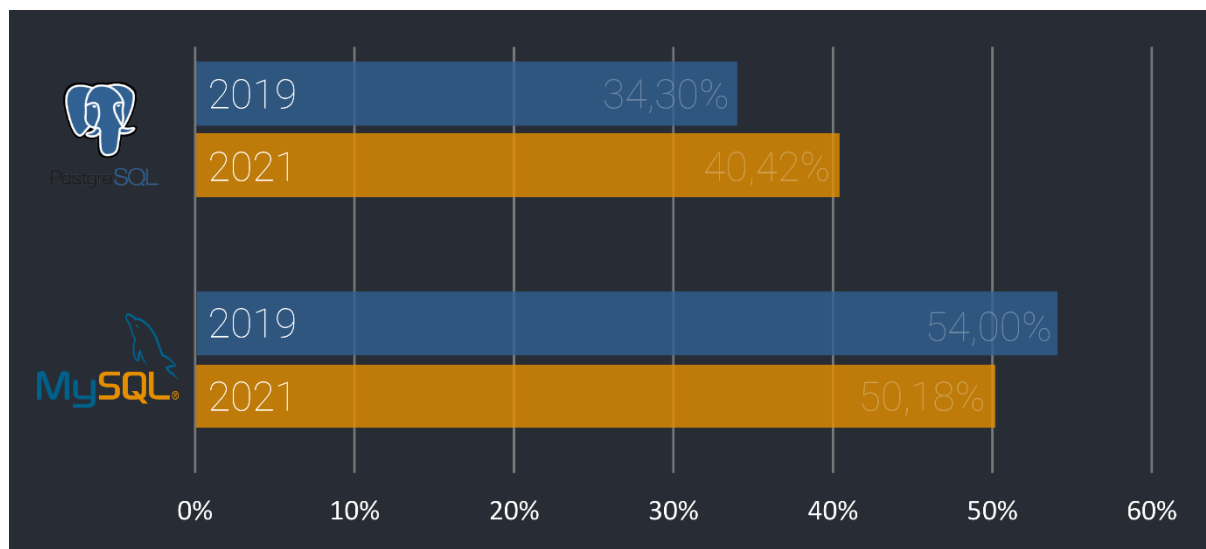
Obje baze imaju mogućnost replikacije i grupiranja te mogu osigurati vodoravnu distribuciju podataka. Međutim, MySQL podržava mehanizam replikacije primarne replike koji osigurava

da se sve promjene podataka repliciraju iz primarne baze podataka u repliku putem SQL-a. Zato replikacija može biti samo asinkrona, što može biti nepovoljno po pitanju skalabilnosti. Dobra strana je što te replike nisu samo za čitanje, te ako aplikacija ne uspije u replikama kada se primarna baza podataka sruši, replike mogu biti čitane i pisane kako bi osigurale besprijekoran rad aplikacija, što može samu aplikaciju učiniti sporom. Dok je PostgreSQL replikacija vrlo popularna i pouzdana, za razliku od MySQL-a temelji se na WAL datotekama što ju čini ne samo bržom nego i lakšom za upravljanje. Razlikuje se još od MySQL replike po tome što osim primarne replike, podržava i kaskadnu replikaciju, koja može biti sinkrona ili asinkrona [72] [73].

Možemo zaključiti da je za kompleksne baze podataka, PostgreSQL jednostavnija opcija zbog raznih funkcionalnosti koje olakšavaju upravljanje složenijim bazama podataka. Dok je MySQL također jednostavna i intuitivna za korištenje, ali može dovesti do složenog SQLa kod kompleksnijih sustava.

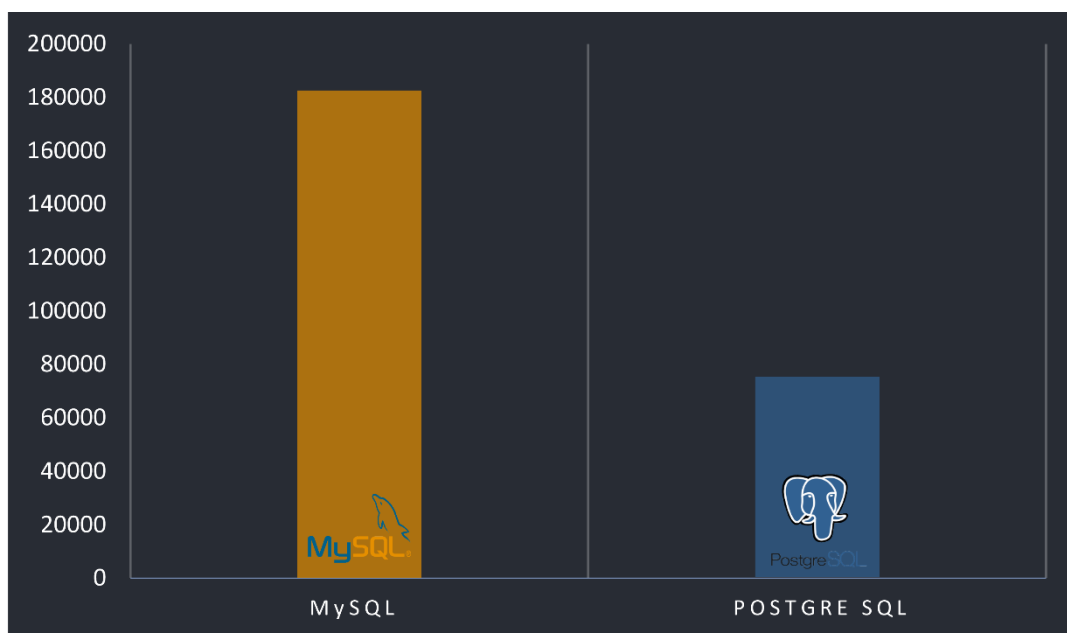
2.Usporedba prema popularnosti

Prema Stack Overflow anketi iz 2021. godine [74], kao što možemo vidjeti na slici 4.14, od 73 317 ispitanika njih 50.18% je u zadnjih godinu dana radilo s MySQL bazom podataka, a njih 40.42% je radilo sa PostgreSQL bazom podataka u zadnjih godinu dana. Ako pogledamo rezultate iste te ankete samo iz 2019. godine [75], od 75 023 ispitanika njih 54% je u zadnjih godinu dana radilo s MySQL bazom podataka, a njih 34.3% je radilo sa PostgreSQL bazom podataka. Iako je MySQL i dalje najpopularniji sustav za upravljanje bazama podataka, PostgreSQL svake godine postaje sve više popularniji i približava se MySQLu.



Slika 4.14 Grafički prikaz popularnosti relacijskih baza podataka (Izvor: autorski rad)

Ako usporedimo njihovu popularnost na LinkedIn stranicama [64] (slika 4.15), također možemo zaključiti da je MySQL trenutno traženiji i popularniji od PostgreSQL sustava za upravljanje bazama podataka na tržištu.

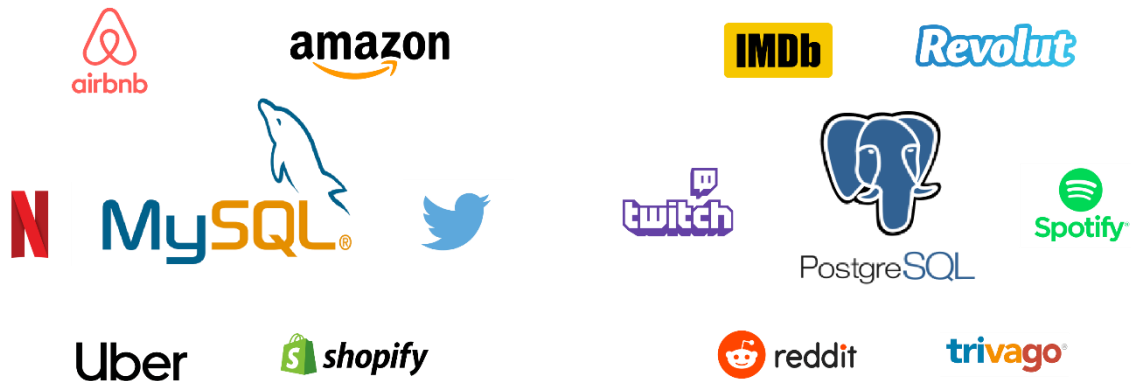


Slika 4.15 Grafički prikaz statistike pretraživanja baza podataka na stranici LinkedIn (Izvor: autorski rad)

3.Usporedba prema tipu aplikacije koji se izrađuje

PostgreSQL je obično najbolji izbor kada web aplikacija zahtjeva proširivu bazu podataka, koja je puna značajki i sadrži složene operacije čitanja i pisanja na velikim skupovima podataka. Odnosno ako se zahtjevi web aplikacije više vrte oko složenih postupaka, integracije, kompleksnog dizajna i integriteta podataka, a ne oko velike brzine i jednostavnosti postavljanja. S druge strane, MySQL je bolja za web aplikacije u oblaku koje imaju manje skupove podataka i brze upite koji se samo čitaju. Također, idealan je odabir ako je potrebna visoka sigurnost baze podataka za web aplikaciju, ali nema složenih zadataka unutar same baze [72] [73].

Na slici 4.16 možemo vidjeti koje popularne web aplikacije koriste MySQL i PostgreSQL sustav za upravljanje bazama podataka. Vidimo da MySQL koriste web aplikacije Airbnb, Amazon, Twitter, Netflix, Uber, itd. Dok PostgreSQL koriste web aplikacije Trivago, Spotify, Twitch, itd.



Slika 4.16 Prikaz aplikacija koje koriste pojedinu relacijsku bazu podataka (Izvor: autorski rad)

4.5 USPOREDBA NERELACIJSKIH BAZA PODATAKA

1. Usporedba prema jednostavnosti/složenosti korištenja

MongoDB i Firebase pohranjuju podatke u JSON obliku koji su pohranjeni u dokumentima, a organizirani po kolekcijama. MongoDB zahtjeva više stručnosti u programiranju i idealan je za velike aplikacije koje nemaju vremensko ograničenje razvoja i dobro podnosi veliki promet.

Firebase je puno lakše za postaviti i krenuti raditi s njim, što ga čini dosta pogodnim za početnike, ali ne podnosi dobro veliki promet što može otežati korištenje. Budući da Firebase podržava Google, tvrtkama je lakše promovirati svoj proizvod. Štoviše, alat za indeksiranje kod Firebasea poboljšava ljestvicu pretraživanja aplikacije na Googleu. Dok kod MongoDBa indeksiranje i pretraživanje nije tako jednostavno i dobro implementirano kao kod Firebasea [76].

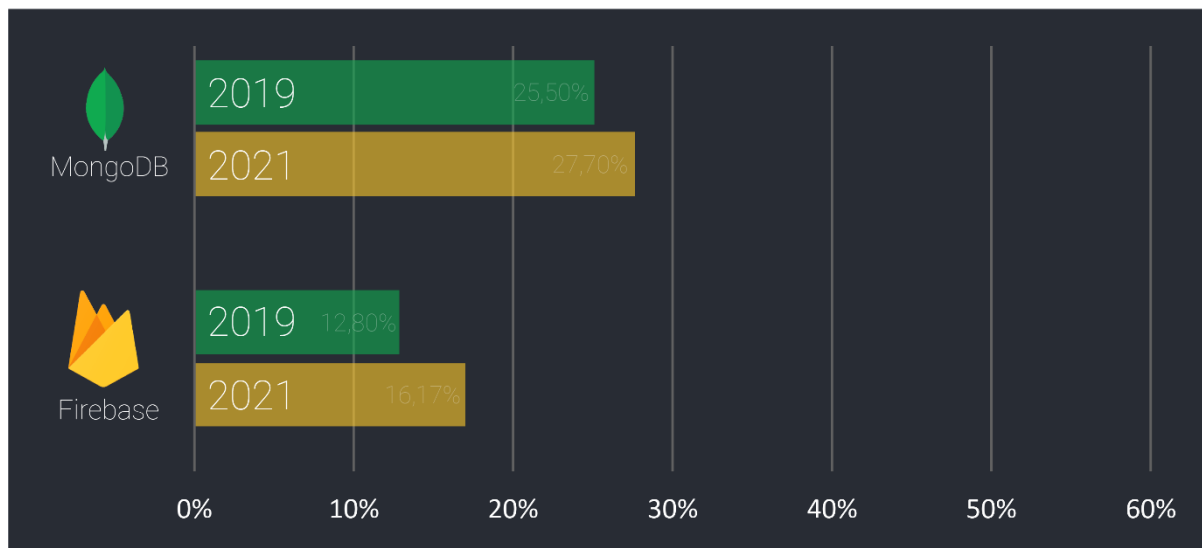
Firebase omogućuje pohranjivanje podataka u oblak, te se svi podaci sinkroniziraju na svim uređajima u stvarnom vremenu, što može olakšati samo korištenje. MongoDB ima višu provjeru autentičnosti i ne može mu se pristupiti na bilo kojem uređaju. To MongoDB čini sigurnijim i stabilnijim, ali isto tako čini Firebase jednostavnijim i pristupačnijim za korištenje [77].

2. Usporedba prema popularnosti

Popularnost ćemo opet prvo usporediti prema anketi objavljenoj na stranicama Stack Overflow gdje je 2021. godine [74] 73 317 ispitanika upitano s kojim sustavom za upravljanje bazama podataka su radili u prethodnoj godini. MongoDB ima postotak od 27.7%, dok Firebase ima postotak od 16.17%.

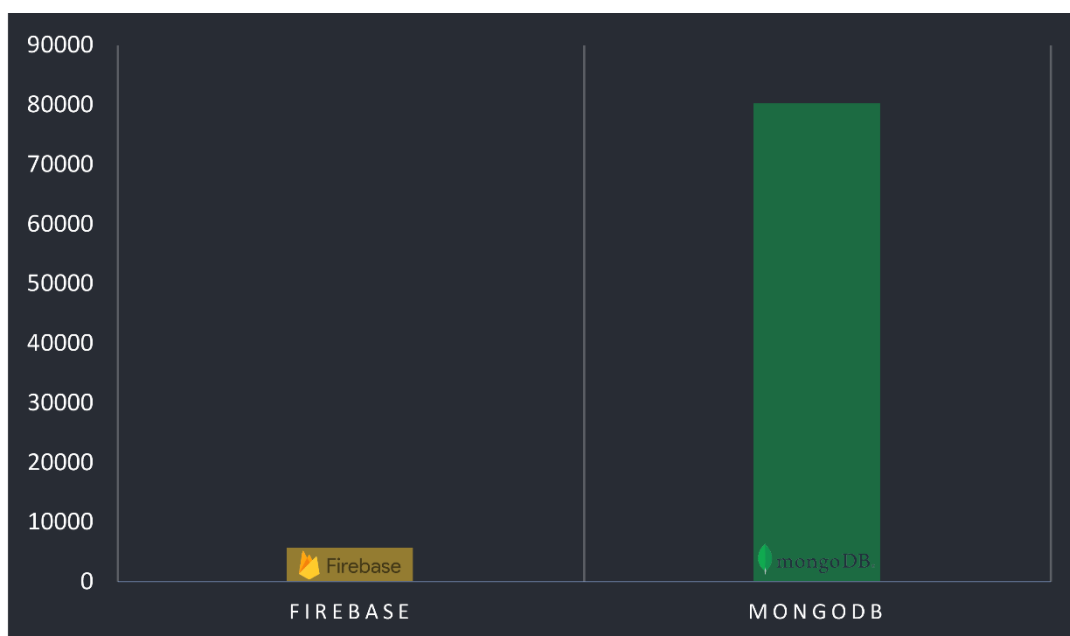
U istoj toj anketi, ali iz 2019. godine [75], situacija nije znatno drugačija - MongoDB ima 25.5%, dok Firebase ima 12.8%. Možemo zaključiti da je MongoDB ipak popularniji i više korišten od Firebasea. Grafički prikaz spomenutih postotaka možemo vidjeti na slici 4.17.

Isto tako možemo zaključiti da sustavi za upravljanje relacijskim bazama podataka imaju veće postotke i da su više korišteni od sustava za upravljanje nerelacijskim bazama podataka. MongoDB je sve popularniji i u anketi iz 2021. godine je četvrti na listi. Ispred njega su: MySQL, PostgreSQL i SQLite.



Slika 4.17 Grafički prikaz ankete s StackOwerflow stranica za nerelacijske baze podataka (Izvor: autorski rad)

Sljedeći kriterij usporedbe nam je broj oglasa za posao na web stranicama LinkedIn [64]. Kao što možemo vidjeti na slici 4.18, MongoDB je daleko traženiji i popularniji na tržištu od Firebasea.



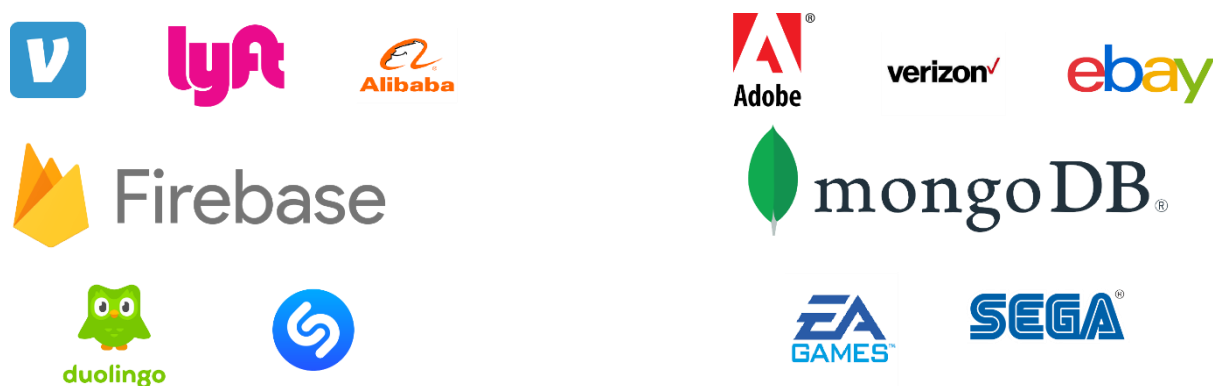
Slika 4.18 Grafički prikaz statistike pretraživanja baza podataka na stranici LinkedIn (Izvor: autorski rad)

3.Usporedba prema tipu aplikacija koja se izrađuje

Kada uspoređujemo ova dva sustava, oni imaju puno sličnosti i puno različitosti, te se biraju ovisno o aplikaciji koju je potrebno razviti.

Firebase dolazi s dodatnim funkcionalnostima kao, primjerice, sinkronizacije u stvarnom vremenu, pohrane podataka i oglašavanje. Pogodniji je za male aplikacije s manjim prometom, koje zahtijevaju brz razvoj i ne tako visoku sigurnost i performanse. To su aplikacije poput društvenih mreža, aplikacije za razmjenu poruka, aplikacije za igre itd. [76]. Dok je MongoDB puno moćniji i pogodniji za velike aplikacije s velikom količinom prometa koje su visokoučinkovite te zahtijevaju visoku razinu sigurnosti. Također je bolji kod skalabilnosti aplikacija i predmemoriranja. Vrlo je pogodan za analizu podataka pogotovo kod složenih upita, kao i za održavanje podataka temeljenih na korisnikovoj lokaciji [77].

Na slici 4.19 možemo vidjeti neke od najpopularnijih web aplikacija koje koriste ove sustave za upravljanje bazama podataka. Vidimo da Firebase sustav koriste web aplikacije: Duolingo, Alibaba, Lyft, itd., dok MongoDB sustav za upravljanje svojom bazom podataka koriste: Adobe, Ebay, EA games, itd.



Slika 4.19 Prikaz aplikacija koje koriste pojedinu nerelacijsku bazu podataka (Izvor: autorski rad)

5 ZAKLJUČAK

Predmet ovog rada je pregled i usporedba suvremenih web tehnologija. Nakon usporedbe web tehnologija po pojedinim tipovima prema tri kriterija: jednostavnost/složenost korištenja, popularnost i prema tipu aplikacije koja se izrađuje možemo zaključiti sljedeće:

- 1) U web tehnologijama za frontend dio - HTML za strukturu, CSS za izgled i JavaScript za interaktivnost su najčešće osnova za svaku web stranicu odnosno aplikaciju. Nemamo ih s čime usporediti jer će gotovo svaka web aplikacija imati u bilo kakvom obliku ta tri jezika. Iako TypeScript kao programski jezik polako dobiva na popularnosti kao programski jezik za frontend razvoj, on nije zamjena za JavaScript te nije pogodan za sve vrste projekata. Tako da JavaScript ostaje najomiljeniji programski jezik za razvoj prednjeg dijela web aplikacija i možemo pretpostaviti da će nastaviti dominirati tržištem. Kod razvojnih okruženja za frontend razvoj tri koja smo spomenuli i koja se jako ističu od ostalih su: Angular, React i Vue. React možemo izdvojiti kao trenutno najpopularniji na tržištu, kao što smo mogli vidjeti iz broja oglasa na stranici LinkedIn, ali po svojim funkcionalnostima ne razlikuje se toliko od Angulara i oba razvojna okruženja su vrlo snažna i nude veliki broj funkcionalnosti. Vue malo kaska za preostalom dvojicom, ali njegova jednostavnost ga čini popularnim među programerima. Sva tri razvojna okruženja imaju svoje prednosti i mane, ali s razlogom su tako popularna te s obzirom na zahtjeve projekta i osobne preferencije programera, jedno će biti prikladnije od ostalih.
- 2) U web tehnologijama za backend, programskih jezika koji s koriste ima znatno više nego što je to slučaj kod frontenda. Spomenuli smo PHP, Javu i Python kao tri trenutno najznačajnija jezika na tržištu. PHP je jedini razvijen specifično za web, te je vrlo velik broj web stranica i aplikacija razvijen njime. Međutim, tokom godina Python mu je preuzeo popularnost ne samo radi svojih funkcionalnosti nego i zbog jednostavnosti sintakse i korištenja što kao posljedicu ima da većina novih programera odabire Python kao jezik za razvoj stražnjeg dijela web aplikacija. Java je, kao što smo vidjeli, bila popularna i zadržala je tu svoju popularnost znatno bolje nego PHP. Kompleksnija je, ali to opravdava svojim brojnim prednostima. Kao zaključak možemo reći da svi ovi programski jezici imaju svoje prednosti i mane, te da sve ovisi o preferencijama. Razvojna okruženja za backend su prema popularnosti dosta slična. Što se funkcionalnosti tiče, Laravel je pogodniji za male aplikacije, dok je za velike i kompleksnije aplikacije pogodniji Django. Express je najprikladniji kod aplikacija kojima treba praćenja posjetitelja i vizualizacije njihovih interakcija u stvarnom vremenu. Svako razvojno okruženje je pogodnije za različite zadatke i zahtjeve same web aplikacije.
- 3) Kao što je već rečeno, imamo dvije osnovne vrste baza podataka: relacijske i nerelacijske. Iako nerelacijske baze podataka postaju sve popularnije, relacijske baze podataka se još uvijek više koriste. Kod relacijskih baza podataka MySQL je najpopularniji na tržištu, ali PostgreSQL mu je svake godine sve bliži. MySQL je

pogodniji za jednostavnije baze podataka, dok je za složenije baze ipak jednostavnije koristiti PostgreSQL. Kod sustava za nerelacijske baze podataka, MongoDB se dosta ističe u odnosu na Firebase kad su u pitanju funkcionalnosti i popularnosti Iako je MongoDB kompleksniji i ima veći broj funkcionalnosti, Firebase je također dosta zanimljiv izbor, pogotovo kada je u pitanju Internet oglašavanje i SEO optimizacija.

- 4) Sve web tehnologije koje smo naveli imaju mnoštvo funkcionalnosti, prednosti i mana. Odabir jedne od njih potpuno ovisi o zadatku odnosno problemu koji je potrebno riješiti. Svaka od njih je bolje rješenje za različite probleme.

World Wide Web se razvija zapanjujućom brzinom i u budućnosti se očekuje sve veći broj korisnika, te uređaja i resursa koji su dostupni na mreži. Stoga i web tehnologije rastu iznimnom brzinom. WWW je započeo kao Web 1.0 sa statičkim web stranicama napisanim samo običnim HTML-om. Zatim je prešao na 2.0 s pojavom JavaScripta i svih drugih tehnologija koje su omogućile korisnicima interakciju s web sadržajem kao i izradu vlastitog. Uskoro web ide korak dalje i ulazi u svoju treću fazu odnosno Web 3.0, koje će biti daleko više decentraliziran nego što je to slučaj danas. Očekuje se da će moderne tehnologije poput AI-a (engl. *Artificial Intelligence*) i IoT-a (engl. *Internet of Things*) uskoro dominirati webom [1].

LITERATURA

- [1] S. Reeves, »Web Technologies: A Journey From HTML To Web 3.0,« Good Core, 12 12 2019. [Mrežno]. Available: <https://www.goodcore.co.uk/blog/web-technologies/>. [Pokušaj pristupa Rujan 2021].
- [2] »Web Technology,« Geeks For Geeks, 13 9 2021. [Mrežno]. Available: <https://www.geeksforgeeks.org/web-technology/>. [Pokušaj pristupa rujan 2021].
- [3] D. A. Kumar, Web Technology theory and practice, Taylor & Francis Group, LLC, 2019.
- [4] Wikipedia, »Web browser,« 16 8 2021. [Mrežno]. Available: https://en.wikipedia.org/wiki/Web_browser. [Pokušaj pristupa Rujan 2021].
- [5] Wikipedia, »Usage share of web browsers,« 2 9 2021. [Mrežno]. Available: https://en.wikipedia.org/wiki/Usage_share_of_web_browsers. [Pokušaj pristupa Rujan 2021].
- [6] Wikipedia, »Google Chrome,« 29 4 2021. [Mrežno]. Available: https://hr.wikipedia.org/wiki/Google_Chrome. [Pokušaj pristupa Rujan 2021].
- [7] M. Muchmore, »Chrome, Edge, Firefox, Opera or Safari, which browser is best?,« PCmag, 20 5 2021. [Mrežno]. Available: <https://www.pcmag.com/picks/chrome-edge-firefox-opera-or-safari-which-browser-is-best>. [Pokušaj pristupa Rujan 20215].
- [8] Jamie, »Ranked: best (and worst) web browser for privacy in 2021,« ExpressVPN, Kolovoz 2021. [Mrežno]. Available: <https://www.expressvpn.com/blog/best-browsers-for-privacy/>. [Pokušaj pristupa Rujan 2021].
- [9] C. Marshall, »The best browser,« Tech radar, 7 9 2021. [Mrežno]. Available: <https://www.techradar.com/best/browser>. [Pokušaj pristupa Rujan 2021].
- [10] Wikipedia, »Safari (software),« 13 9 2021. [Mrežno]. Available: [https://en.wikipedia.org/wiki/Safari_\(software\)](https://en.wikipedia.org/wiki/Safari_(software)). [Pokušaj pristupa Rujan 2021].
- [11] »Microsoft Edge,« Wikipedia, 11 8 2021. [Mrežno]. Available: https://hr.wikipedia.org/wiki/Microsoft_Edge. [Pokušaj pristupa Rujan 2021].
- [12] Wikipedia, »Firefox,« 23 9 2020. [Mrežno]. Available: <https://bs.wikipedia.org/wiki/Firefox>. [Pokušaj pristupa Rujan 2021].

- [13] C. Castiglione, »Frontend vs backend developer,« One month, 27 6 2019. [Mrežno]. Available: <https://learn.onemonth.com/frontend-vs-backend-developers/>. [Pokušaj pristupa Rujan 2021].
- [14] »Frontend vs Backend,« Geeks for geeks, 8 31 2021. [Mrežno]. Available: <https://www.geeksforgeeks.org/frontend-vs-backend/>. [Pokušaj pristupa Rujan 2021].
- [15] Wikipedia, »API,« 3 9 2020. [Mrežno]. Available: <https://hr.wikipedia.org/wiki/API>. [Pokušaj pristupa Rujan 2021].
- [16] »HTML: HyperText Markup Language,« MDN Web Docs, 6 8 2021. [Mrežno]. Available: <https://developer.mozilla.org/en-US/docs/Web/HTML>. [Pokušaj pristupa Rujan 2021].
- [17] J. Duckett, HTML & CSS design and build websites, John Wiley & Sons, Inc., 2011.
- [18] »CSS: Cascading Style Sheets,« MDN Web Docs, 2 9 2021. [Mrežno]. Available: <https://developer.mozilla.org/en-US/docs/Web/CSS>. [Pokušaj pristupa Rujan 2021].
- [19] »JavaScript,« Wikipedia, 22 9 2021. [Mrežno]. Available: <https://en.wikipedia.org/wiki/JavaScript>. [Pokušaj pristupa Rujan 2021].
- [20] »Top 7 Programming Languages for Backend Web Development,« Geeks for Geeks, 27 4 2021. [Mrežno]. Available: <https://www.geeksforgeeks.org/top-7-programming-languages-for-backend-web-development/>. [Pokušaj pristupa Rujan 2021].
- [21] M. Dionne, »Reasons Why JavaScript is Omnipresent in Modern Development,« Snipcart, 11 4 2019. [Mrežno]. Available: <https://snipcart.com/blog/why-javascript-benefits>. [Pokušaj pristupa Rujan 2021].
- [22] »Top 5 Programming languages for Web development in 2021,« Medium, 13 1 2021. [Mrežno]. Available: <https://medium.com/javarevisited/top-5-programming-languages-for-web-development-in-2021-f6fd4f564eb6>. [Pokušaj pristupa Rujan 2021].
- [23] »Introduction to the DOM,« MDN Web Docs, 2 9 2021. [Mrežno]. Available: https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model/Introduction.
- [24] S. Khan, »What makes JavaScript so popular,« General Assembly, 15 3 2021. [Mrežno]. Available: <https://generalassemb.ly/blog/what-makes-javascript-so-popular/>. [Pokušaj pristupa Rujan 2021].

- [25] H. Shnaider, »Top 7 Languages for Web App Development,« Forty Seven, 30 6 2020. [Mrežno]. Available: <https://fortyseven47.com/news/top-7-languages-for-web-app-development/>. [Pokušaj pristupa Rujan 2021].
- [26] J. Korsun, »The 16 Most Important Pros and Cons of using Python for Web Development,« DjangoStars, 7 6 2021. [Mrežno]. Available: <https://djangostars.com/blog/python-web-development/>. [Pokušaj pristupa Rujan 2021].
- [27] T. Madeira, »Why use Python for Web Development?,« Imaginary Cloud, 24 12 2020. [Mrežno]. Available: <https://www.imaginarycloud.com/blog/why-use-python-for-web-development/>. [Pokušaj pristupa Rujan 2021].
- [28] R. Vats, »Top 5 Java Web Application Technologies You Should Master in 2021,« UpGrad Blog, 2 1 2021. [Mrežno]. Available: <https://www.upgrad.com/blog/top-java-web-application-technologies/>. [Pokušaj pristupa Rujan 2021].
- [29] A. Roznovsky, » Why Use PHP? Main Advantages and Disadvantages,« Light IT, 21 2 2021. [Mrežno]. Available: <https://light-it.net/blog/why-use-php-main-advantages-and-disadvantages/>. [Pokušaj pristupa Rujan 2021].
- [30] Y. Illiv, » The Best Web Application Development Languages in 2021,« Inverta, 10 35 2021. [Mrežno]. Available: <https://invertasoft.com/blog/the-best-web-application-development-languages-in-2021>. [Pokušaj pristupa Rujan- 2021].
- [31] C. Flynn, »14 Technologies Every Web Developer Should Be Able to Explain,« Differential, 27 1 2015. [Mrežno]. Available: <https://www.differential.com/posts/14-technologies-every-web-developer-should-be-able-to-explain/>. [Pokušaj pristupa Rujan 2021].
- [32] »Differences between Bootstrap and JQuery UI,« Geeks for Geeks, 7 2 2019. [Mrežno]. Available: <https://www.geeksforgeeks.org/differences-between-bootstrap-and-jquery-ui/>. [Pokušaj pristupa Rujan 2021].
- [33] R. Berezhnoi, »What is Bootstrap?,« F5 Studio, 18 1 2019. [Mrežno]. Available: <https://f5-studio.com/articles/what-is-bootstrap-and-how-to-use-it-in-web-development/>. [Pokušaj pristupa Rujan 2021].
- [34] »Why we use jQuery in our web application ?,« Geeks for Geeks, 12 8 2021 . [Mrežno]. Available: <https://www.geeksforgeeks.org/why-we-use-jquery-in-our-web-application/?ref=rp>. [Pokušaj pristupa Rujan 2021].
- [35] »jQuery Introduction,« W3 Schools, [Mrežno]. Available: https://www.w3schools.com/jquery/jquery_intro.asp. [Pokušaj pristupa Rujan 2021].

- [36] Wikipedia, »React (JavaScript library),« 10 9 2021. [Mrežno]. Available: [https://en.wikipedia.org/wiki/React_\(JavaScript_library\)](https://en.wikipedia.org/wiki/React_(JavaScript_library)). [Pokušaj pristupa Rujan 2021].
- [37] »Top Front-End Frameworks in 2021,« Existek, 14 1 2020. [Mrežno]. Available: <https://existek.com/blog/top-front-end-frameworks-2021/>. [Pokušaj pristupa Rujan 2021].
- [38] »Pros and Cons of Vue.js Framework Programming,« DDI Development , Prosinac 2020. [Mrežno]. Available: <https://ddi-dev.com/blog/programming/the-good-and-the-bad-of-vue-js-framework-programming/>. [Pokušaj pristupa Rujan 2021].
- [39] »The Good and the Bad of Vue.js Framework Programming,« Altexsoft, 11 9 2019. [Mrežno]. Available: <https://www.altexsoft.com/blog/engineering/pros-and-cons-of-vue-js/>. [Pokušaj pristupa Rujan 2021].
- [40] »The Good and the Bad of Angular Development,« Altexsoft, 25 3 2020. [Mrežno]. Available: <https://www.altexsoft.com/blog/engineering/the-good-and-the-bad-of-angular-development/>. [Pokušaj pristupa Rujan 2021].
- [41] » Advantages and Disadvantages of Angular,« Knowledge Hut, 15 3 2021. [Mrežno]. Available: <https://www.knowledgehut.com/blog/web-development/advantages-and-disadvantages-of-angular>. [Pokušaj pristupa Rujan 2021].
- [42] »Express.js,« Devopedia, 1 3 2019. [Mrežno]. Available: <https://devopedia.org/express-js>. [Pokušaj pristupa Rujan 2021].
- [43] V. Sagar, »Demystifying ExpressJS pros and cons,« Pattem, 20 8 2020. [Mrežno]. Available: <https://pattemdigital.com/insight/advantages-of-express-js/>. [Pokušaj pristupa Rujan 2021].
- [44] H. Dhaduk, »Node.js vs. Express: Determining the Best Backend Technology,« Simform, 12 4 2021. [Mrežno]. Available: <https://www.simform.com/blog/node-js-vs-express/#topic-2>. [Pokušaj pristupa Rujan 2021].
- [45] H. Dhaduk, »Node.js vs. Express: Determining the Best Backend Technology,« Simform, 12 4 2021. [Mrežno]. Available: <https://www.simform.com/blog/node-js-vs-express/#topic-3>. [Pokušaj pristupa Rujan 2021].
- [46] Y. Nader, »What is Django? Advantages and Disadvantages,« Hacrk.io, 1 11 2020. [Mrežno]. Available: <https://hackr.io/blog/what-is-django-advantages-and-disadvantages-of-using-django>. [Pokušaj pristupa Rujan 2021].

- [47] S. Jaiswal, »What is Laravel? Advantages & Disadvantages,« Courseya, 17 11 2019. [Mrežno]. Available: <https://www.courseya.com/blog/what-is-laravel-features-and-advantages/>. [Pokušaj pristupa Rujan 2021].
- [48] »Advantages and disadvantages of Laravel Framework for web Development,« DDI Development , Srpanj 2020. [Mrežno]. Available: <https://ddi-dev.com/blog/programming/pros-and-cons-of-laravel-framework-for-web-app-development/>. [Pokušaj pristupa Rujan 2021].
- [49] »Relational Databases,« IBM Cloud Education, 6 8 2019. [Mrežno]. Available: <https://www.ibm.com/cloud/learn/relational-databases>. [Pokušaj pristupa Rujan 2021].
- [50] D. Pawlan, »Relational vs. Non-Relational Database: Pros & Cons,« [Mrežno]. Available: <https://aloo.co/blog/relational-vs-non-relational-database-pros-cons>. [Pokušaj pristupa Rujan 2021].
- [51] A. Saini, »Some Popular Database for Web Development,« Geeky Humans, 12 11 2020. [Mrežno]. Available: <https://geekyhumans.com/database-for-web-development/>. [Pokušaj pristupa Rujan 2021].
- [52] »What is MongoDB – Working and Features,« Geeks for Geeks, 6 6 2021. [Mrežno]. Available: <https://www.geeksforgeeks.org/what-is-mongodb-working-and-features/>. [Pokušaj pristupa Rujan 2021].
- [53] »Introduction to Firebase,« Hackernoon, 27 12 2017. [Mrežno]. Available: <https://hackernoon.com/introduction-to-firebase-218a23186cd7>. [Pokušaj pristupa Rujan 2021].
- [54] Y. Nadar, »Python vs Java in 2021,« Hackr.io, 21 8 2021. [Mrežno]. Available: <https://hackr.io/blog/python-vs-java>. [Pokušaj pristupa Rujan 2021].
- [55] A. Goel, »Python vs PHP in 2021,« Hackr.io, 27 8 2021. [Mrežno]. Available: <https://hackr.io/blog/python-vs-php>. [Pokušaj pristupa Rujan 2021].
- [56] » Say 'Hello World' in 28 Different Programming Languages,« Excel With Business, [Mrežno]. Available: <https://excelwithbusiness.com/blog/say-hello-world-in-28-different-programming-languages/>. [Pokušaj pristupa Rujan 2021].
- [57] »The same very simple program in many different programming languages,« Wellho, [Mrežno]. Available: http://www.wellho.net/mouth/2700_The-same-very-simple-program-in-many-different-programming-languages.html. [Pokušaj pristupa Rujan 2021].
- [58] »PYPL PopularitY of Programming Language,« 2021. [Mrežno]. Available: <https://pypl.github.io/PYPL.html>. [Pokušaj pristupa Rujan 2021].

- [59] R. Kuts, »React vs Angular vs Vue.js — What Is the Best Choice in 2021?,« Tech Magic, 24 1 2021. [Mrežno]. Available: <https://www.techmagic.co/blog/reactjs-vs-angular-vs-vuejs-what-to-choose-in-2020/>. [Pokušaj pristupa Rujan 2021].
- [60] S. Daityari, »Angular vs React vs Vue: Which Framework to Choose in 2021,« Code in WP, 15 3 2021. [Mrežno]. Available: <https://www.codeinwp.com/blog/angular-vs-vue-vs-react/>. [Pokušaj pristupa Rujan 2021].
- [61] »Angular,« GitHub, [Mrežno]. Available: <https://github.com/angular/angular>. [Pokušaj pristupa Rujan 2021].
- [62] »Vuejs,« GitHub, [Mrežno]. Available: <https://github.com/vuejs/vue>. [Pokušaj pristupa Rujan 2021].
- [63] »React,« GitHub, [Mrežno]. Available: <https://github.com/facebook/react>. [Pokušaj pristupa Rujan 2021].
- [64] »Jobs Search,« LinkedIn, [Mrežno]. Available: <https://www.linkedin.com/jobs/search?position=1&pageNum=0>. [Pokušaj pristupa Rujan 2021].
- [65] Y. Nader, »Django Vs Laravel: Which framework to choose?,« Hackr.io, 1 11 2020. [Mrežno]. Available: <https://hackr.io/blog/django-vs-laravel>. [Pokušaj pristupa Rujan 2021].
- [66] S. Asadi, »Express.js vs Java Spring vs PHP Laravel,« Medium, 19 1 2019. [Mrežno]. Available: <https://medium.com/@lvlr.xaus/express-js-vs-java-spring-vs-php-laravel-fe74a68828b3>. [Pokušaj pristupa Rujan 2021].
- [67] »expressjs,« GitHub, [Mrežno]. Available: <https://github.com/expressjs/express>. [Pokušaj pristupa Rujan 2021].
- [68] »Laravel,« GitHub, [Mrežno]. Available: <https://github.com/laravel/laravel>. [Pokušaj pristupa Rujan 2021].
- [69] »Django,« GitHub, [Mrežno]. Available: <https://github.com/django/django>. [Pokušaj pristupa Rujan 2021].
- [70] H. Dhaduk, »Django vs Laravel: The Most Analytical Breakdown To Determine The Better Framework,« Simform, 12 2 2021. [Mrežno]. Available: <https://www.simform.com/blog/django-vs-laravel/>. [Pokušaj pristupa Rujan 2021].
- [71] »Što je CRM?,« Vtiger, [Mrežno]. Available: <https://www.vtiger.com/hr/what-is-crm/>. [Pokušaj pristupa Rujan 2021].
- [72] »PostgreSQL vs. MySQL: A 360-degree Comparison [Syntax, Performance, Scalability and Features],« EDB Blog, 11 12 2019. [Mrežno]. Available:

<https://www.enterprisedb.com/blog/postgresql-vs-mysql-360-degree-comparison-syntax-performance-scalability-and-features>. [Pokušaj pristupa Rujan 2021].

- [73] B. Rogojan, »PostgreSQL Vs MySQL: Different Databases For Different Use Cases,« Panoply Blog, 15 1 2021. [Mrežno]. Available: <https://blog.panoply.io/postgresql-vs.-mysql>. [Pokušaj pristupa Rujan 2021].
- [74] »2021 Developer Survey,« Stack Overflow, Svibanj 2021. [Mrežno]. Available: <https://insights.stackoverflow.com/survey/2021#most-popular-technologies-database>. [Pokušaj pristupa Rujan 2021].
- [75] »Developer Survey 2019,« Stack Owerflow, Svibanj 2019. [Mrežno]. Available: <https://insights.stackoverflow.com/survey/2019>. [Pokušaj pristupa Rujan 2021].
- [76] R. Sharma, »Firebase Vs MongoDB: Difference Between Firebase & MongoDB,« UpGrad Blog, 15 7 2020. [Mrežno]. Available: <https://www.upgrad.com/blog/firebase-vs-mongodb/>. [Pokušaj pristupa Rujan 2021].
- [77] K. Shah, »Firebase Vs MongoDB: Battle of the Best Databases For 2021,« Third Rock Techkno, 20 8 2021. [Mrežno]. Available: <https://www.thirdrocktechkno.com/blog/firebase-vs-mongodb-battle-of-the-best-databases-for-2021/>. [Pokušaj pristupa Rujan 2021].
- [78] F. Esposito, »9To5Mac,« 2020 9 2021. [Mrežno]. Available: <https://9to5mac.com/2020/09/16/safari-14-is-now-available-as-a-standalone-update-for-macos-catalina-and-mojave-users/>. [Pokušaj pristupa Rujan 2021].

POPIS SLIKA

Slika 3.1 Udio korištenja web preglednika izražen u postocima (Izvor: autorski rad)	8
Slika 3.2 Početna stranica korisničkog sučelja Google Chrome web preglednika (Izvor: autorski rad) .	9
Slika 3.3 Početna stranica korisničkog sučelja preglednika Safari [78]	10
Slika 3.4 Početna stranica korisničkog sučelja web preglednika Microsoft Edge (Izvor: autorski rad)	11
Slika 3.5 Početna stranica korisničkog sučelja web preglednika Mozilla Firefox (Izvor: autorski rad)	12
Slika 3.6 Jezici za frontend (Izvor: autorski rad)	13
Slika 3.7 Biblioteke za frontend (Izvor: autorski rad)	13
Slika 3.8 Razvojna okruženja za frontend (Izvor: autorski rad).....	14
Slika 3.9 Programski jezici za backend (Izvor: autorski rad)	14
Slika 3.10 Razvojna okruženja za backend (Izvor: autorski rad)	15
Slika 3.11 Primjer HTML elementa - Naslov prve razine (Izvor: autorski rad)	15
Slika 3.12 Primjer HTML dokumenta (Izvor: autorski rad)	16
Slika 3.13 Primjer CSS pravila (Izvor: autorski rad)	17
Slika 3.14 Primjer JavaScript koda (Izvor: autorski rad)	21
Slika 3.15 Primjer JQuery koda (Izvor: autorski rad)	21
Slika 3.16 Sustavi za upravljanje relacijskim bazama podataka (Izvor: autorski rad)	27
Slika 3.17 Sustavi za upravljanje ne relacijskim bazama podataka (Izvor: autorski rad).....	28
Slika 4.2 Primjer programskog koda u jeziku PHP (Izvor: autorski rad)	30
Slika 4.1 Primjer programskog koda u jeziku Python (Izvor: autorski rad).....	30
Slika 4.3 Primjer programskog koda u jeziku Java (Izvor: autorski rad)	30
Slika 4.5 Primjer programskog koda u jeziku Python (Izvor: autorski rad).....	31
Slika 4.4 Primjer programskog koda u jeziku PHP (Izvor: autorski rad)	31
Slika 4.6 Primjer programskog koda u jeziku Java (Izvor: autorski rad)	31
Slika 4.7 Grafički prikaz PYPL indexa za 2021. i 2008. godinu (Izvor: autorski rad)	32
Slika 4.8 Prikaz Statistike frontend razvojnih okruženja sa stranica GitHub (Izvor: autorski rad)	33
Slika 4.9 Grafički prikaz statistike pretraživanja frontend razvojnih okruženja na stranici LinkedIn (Izvor: autorski rad)	34
Slika 4.10 Prikaz aplikacija razvijenih s pojedinim frontend razvojnim okruženjem (Izvor: autorski rad)	35
Slika 4.11 Prikaz statistike backend razvojnih okruženja sa stranica GitHub (Izvor: autorski rad)	36
Slika 4.12 Grafički prikaz statistike pretraživanja backend razvojnih okruženja na stranici LinkedIn (Izvor: autorski rad)	37
Slika 4.13 Prikaz aplikacija razvijenih pojedinim backend razvojnim okruženjem (Izvor: autorski rad)	38
Slika 4.14 Grafički prikaz popularnosti relacijskih baza podataka (Izvor: autorski rad)	39
Slika 4.15 Grafički prikaz statistike pretraživanja baza podataka na stranici LinkedIn (Izvor: autorski rad)	40
Slika 4.16 Prikaz aplikacija koje koriste pojedinu relacijsku bazu podataka (Izvor: autorski rad)	41
Slika 4.17 Grafički prikaz ankete s StackOwerflow stranica za nerelacijske baze podataka(Izvor: autorski rad).....	42
Slika 4.18 Grafički prikaz statistike pretraživanja baza podataka na stranici LinkedIn (Izvor: autorski rad)	42

Slika 4.19 Prikaz aplikacija koje koriste pojedinu nerelacijsku bazu podataka (Izvor: autorski rad) ... 43