Univerzitet Singidunum

Centar Novi Sad

Diplomski rad

**VIŠESLOJNA WEB APLIKACIJA ZA ON-LINE PRODAJU RAČUNARA**

Mentor: prof. Đorđe Obradović

Student: Jan Vranka

Smer: Informatika i računarstvo

Broj indeksa: 2015270067

Novi Sad, mart 2020.

1. **SADRŽAJ**
2. Uvod
3. Elektronska trgovina i razvoj
   1. Online prodaja
   2. Modeli elektronske trgovine
      1. B2b
      2. B2c
      3. C2c
      4. Trgovina unutar kompanije
      5. E obrazovanje
      6. Mobilna trgovina
4. Tehnologije
   1. Angularjs
   2. Expressjs
   3. MongoDB
   4. NODEjs
5. Arhitekture
   1. 4.1. MVC Model
   2. 4.2. Klijent-server arhitektura
6. UML dijagrami
7. Web aplikacija
8. Zaključak
9. Literatura
10. **UVOD**

U 21. veku, svedoci smo rapidnog razvoja tehnologije zavidnog nivoa, bilo iz aspekta njenog kvaliteta illi brzine nastanka. Posmatrajući napredak razvoja samih fizičkih, opipljivih, tehnoloških komponenata (eng. *Hardware*) i softvera (eng. *Software*) koji čini nezaobilazni deo svake od njih, dolazimo do evidentnog zaključka, svet današnjice se zasniva na istim, bilo da je posredi lična ili poslovna upotreba, tehnologija je inkorporirana u svakodnevnicu svakog pojedinca.

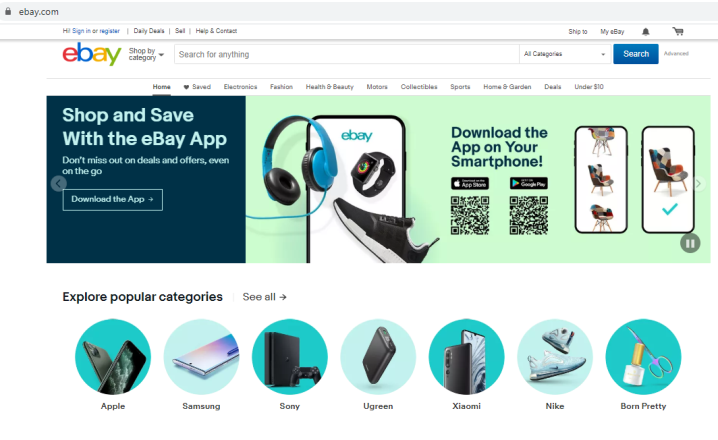
Sveukupni gore napomenuti razvoj praćen je razvojem, globalizacijom i popularizacijom Interneta i svih usluga koje on nudi u okviru tzv. „Ere Interneta“. Umrežavanjem preduzeća i celokupne javne administracije, omogućeno je internet poslovanje, koje ruši sve do sada postavljene granice, da li u vidu geografskih, valutnih ili jezičkih barijera. Omogućena je brza komunikacija, prenos velike količine podataka, skladištenje i pristup istim u bilo kom datom momentu.

Veliki deo internet poslovanja se sastoji u internet trgovini koja podrazumeva procese prodaje, kupovine, transfera proizvoda ili informacija. Online prodaja je široko zastupljena u celom svetu, trend koji uzima maha, pa se beleži i eksponencijalni rast istog u Srbiji. Pogodnosti koje internet kupovina nosi sa sobom i čini je privlačnom širokim masa, jeste mogućnost kupovine od kuće, bez odlaska do svake pojedinačne prodavnice, zatim nemerljiva količina izbora, dostupnost proizvoda izvan zemlje življenja, kao i ušteda vremena koje nam je svima prekopotrebno. Sama on-line prodaja omogućava korisniku da putem interneta odabere i poruči željeni proizvod koji mu nakon toga biva dostavljen na kućnu adresu.

Inicijalni problem kod internet kupovine je predstavljala nepouzdanost kupovine „na neviđeno“, nepoverenje, strah od različitih malverzacija, zatim različitih vrsta krađe i generalni gubitak novca koji nikom nije u interesu. Danas, kako je ovaj trend uzeo svog maha, celokupni proces je napredavao sa brojnim oblicima zaštite korisnika usluge i različitih tipova osiguranja kod pokušaja bilo kog oblika malverzacija, koji doprinose daljem razvoju, široj upotrebi ovog načina poslovanja i vraćanja poverenja u ovaj jedinstveni sistem. Kao jedan od vidova zaštite, prilikom kupovine sa brojnih sajtova (eng. *Site*) potrebno je uneti lične podatke i upoznati se sa uslovima i pravilima korišćenja sajta, posledično u slučaju bilo kakve neispravnosti istih pojedinac odgovara za svoje postupke.

U osnovi rada se nalazi kreirana Web aplikacija prezentacionog tipa, odnosno nije pravljena namenski za upotrebu neke kompanije. Ideja za rad se javila tokom slušanja predmeta „Interakcija čovek – računar“ sa željom budućeg poslovanja u ovom domenu. Web prodavnica za prodaju računara ilustrativna i trenutno ne podržava sve opcije on-line kupovine sa mogućnošću daljeg nadograđivanja i usavršavanja iste.

1. **ELEKTRONSKA TRGOVINA I RAZVOJ**
   1. **Online prodaja**

Pojam internet trgovine i internet poslovanja se prvi put spominje polovinom 80-ih godina 20. veka. Prva internet prodaja je bio CD Engleskog umetnika Sting-a, pod nazivom „*Ten Summoner’s Tales*“, od strane grupe studenata Američkog *Swarthmore* *College*-a. Zatim, 1996. godine, dolazi do nastanka prvog *website*-a za online prodaju. Tvorac ovog *site*-a je Pjer Omidar pod nazivom „*Auction Web*“. Kao prva stavka za prodaju je postavljen Pjerov lični pokvareni laser, koji je tada dostigao cenu od 14.83$. Ovo je bio prvi *site* za online prodaju koji je zaživeo i postavio temelje današnje trgovine putem interneta. Kasnije, *site* je preimenovan u „Echo Bay“ ili „eBay“ koji njegovom tvorcu doneo milione za ovo revolucionarno otkriće. Vremenom, dolazi do usavršavanja ovog načina trgovine kako bi u 21. veku on zauzeo vodeću poziciju [1].

Slika 1. eBay 1999. godine Slika 2. eBay 2020. godine

Početak online trgovine u Srbiji se vezuje za 29. jun 1993. godine i osnivanje Jugoslovenske asocijacije za elektronsku razmenu podataka pod nazivom „*Yugoslav* *Association for Electronic Data Interchange- YUEDI*“. Cilj je bio popularizacija primene EDI (*Electronic Data Interchange*) sistema u tadašnjoj Jugoslaviji. Prva pojava *Internet* *provider*-a se dešava 1995. godine [2].

Početak elektronske trgovine u Srbiji se karakterisao samo odabirom i naručivanjem proizvoda preko interneta, koji bi zatim bio poslat na adresu kupca, a novčana transakcija se obavljala pouzećem prilikom dostave. Prvi elektronski virtuelni trgovinski centar se se otvara u julu 1998. godine, nastao u saradnji Eunet-a i firme YUGate. Ovim potezom dolazi do otvaranja većeg broja online prodavnica. Broj prodavnica i njihova funkcionalnost nastavlja da raste sa pojavom elektronskih kartica čime je bilo omogućeno online plaćanje. Prvi i jedini *Internet Payment Provider* E-Bank.co.yu je osnovan 1999. godine i namenjen je procesiranju platnih kartica. Ovo predstavlja drugu fazu u razvoju online trgovine u Srbiji.

Dalji razvoj online trgovine u Srbiji tj. treća faza razvoja počinje 2002. godine i smatra se da traje do danas. Broj *site*-ova i online prodavnica svakodnevno raste, zajedno sa njihovom popularnošću. Danas svi veliki brendovi imaju svoju online prodavnicu, dok za njima ne zaostaju ni manje trgovačke kompanije.

Prilikom svake kupovine, kupac prolazi kroz jasno definisane korake, odabir proizvoda, načina plaćanja, načina isporuke i sigurnost tokom celog postupka. Određene prodavnice omogućavaju i zamenu proizvoda uz troškove transporta koje snosi kupac. Na kraju same porudžbine, pre njenog slanja u prodavnicu na obradu postoji opcija provere svih unetih podataka iznova i finalna završna potrvda iste. Kod većine prodavnica na svakom koraku postoji pitanje „Da li ste sigurni?“ i opcija za povratak na prethodnu stranu ukoliko je došlo do neke greške prilikom unosa informacija i njihove ispravke [3].

* 1. **Modeli elektronske trgovine**

Postoji više modela tj. tipova elektronske trgovine koji su danas u upotrebi. Najveći prihod dolazi od B2B modela, čak 85% od ukupne elektronske trgovine, dok preostalih 15% dolazi od preostalih modela. Osnovni modeli kroz koje se posmatra elektronska trgovina su:

* + 1. **B2B – biznis prema biznisu**

Ovaj model elektronske trgovine (eng. ***Business – to – Business*** ) se zasniva na prodaji proizvoda i usluga između kompanija i korporacija. Predstavlja automatizovanu razmenu informacija u okviru jedne kompanije ili između različitih kompanija. Ovaj tip trgovine obuhvata širok spektar primene motornih vozila, hrane, kućnih potrepština, kompjuterske elektronike i dalje. U okviru B2B postoje tri osnovna modela – model prodajne strane, model kupovne strane i razmena. Kao najvažniji oblik elektronske trgovine obuhvata većinu elektronskog trgovanja i odgovoran je za 85% ukupnih prihoda elektronske trgovine na Internetu.

Skupi i glomazni EDI zatvoreni sistemi koji su godinama korišćeni od strane velikih kompanija su zamenjene ovim modelom elektronske kupovine i postaju otvoreni. Svojom pojavom na tržištu doprinosi smanjenju troškova, povećanoj prodaji, eliminaciji grešaka i efikasnijoj komunikaciji.

* + 1. **B2C – biznis prema kupcu**

Model elektronske trgovine B2C (eng. ***Business – to – Customer***) uzima prvo mesto u popularnosti online trgovine usmerene ka individualnom potrošaču. Poznat je i pod nazivom elektronska maloprodaja. Po definiciji predstavlja oblik trgovine preko interneta između kompanije i ličnih potreba kupca. Obuhvata dva osnovna načina za obavljanje trgovine – direktna prodaja kupcu i prodaja preko posrednika.

Internet potrošači se odnose prema Web-u kao prema novom tržišnom prostoru. Razvijaju odnos prema određenim site-ovima, kojima se uvek prvo okreću. Mogućnost provere cene i proizvoda na drugim sajtovima ostaje uvek otvorena čime je ceo proces kupovine mnogo olakšan i pristupačniji. Osnovne prednosti ovog načina trgovine jesu:

* Porudžbine stižu za 24h
* Proširuje se tržište
* Kupci se lakše opredele za kupovinu
* Niže cene prezentovanja proizvoda i usluga putem online kataloga [4]
  + 1. **C2C – kupac prema kupcu**

Model **C2C** (eng. ***Consumer – to – Consumer***) predstavlja novi oblik slobodne trgovine među potrošačima. Potrošači direktno trguju sa drugim potrošačima. Kompanije koje podržavaju ovaj način prodaje nalaze svoju naplatu u nekim vidovima klasičnog oblika malog procenta transakcije, članarina ili drugi vid poslovne kombinacije. Prednosti ovog modela trgovine se zasnivaju na sve širem potencijalnom tržištu, eleminisanju posrednika, ažuriranju sadržaja i dostupnost sa bilo kog mesta u bilo koje vreme. Dok negativne strane se ogledaju u nedostaku kontrole kvaliteta i garancije izvršenog plaćanja.

* + 1. **Trgovina unutar kompanije**

Elektronska trgovina unutar kompanije ili ***Intrabusiness*** (eng.). podrazumeva sve aktivnosti razmene dobara, servisa i informacija unutar jedne kompanije. Postoje dve forme ove trgovine:

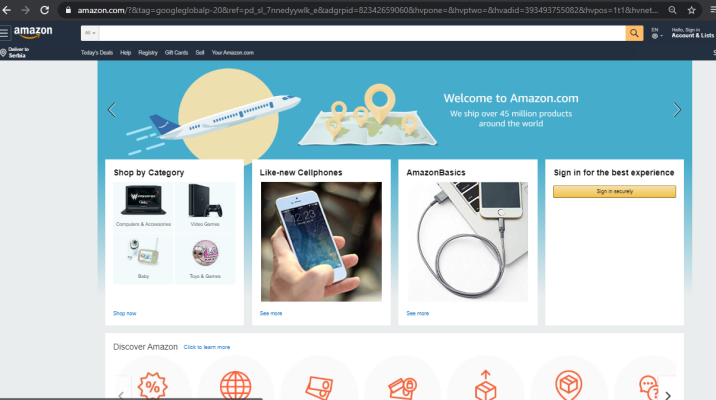
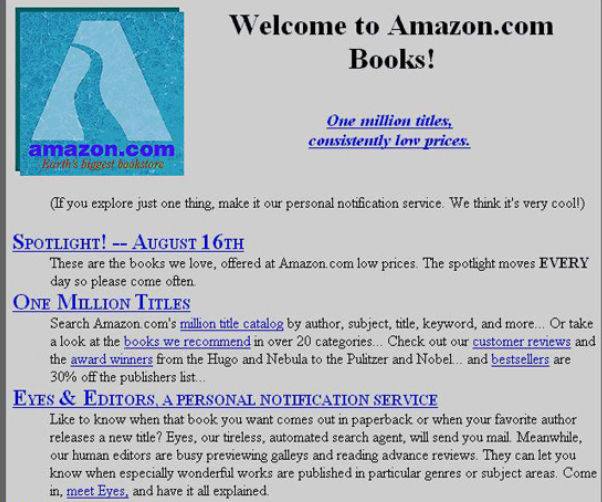
* Trgovina između i unutar organizacionih jedinica
* Trgovina između zaposlenih u organizaciji [5]
  + 1. **E – obrazovanje**

Savremeni način učenja putem informacionih i telekomunikacionih tehnologija spada u domen E-obrazovanja. Pod pojam E-obrazovanja spada i domen tehnologije naprednog učenja (eng. ***Advanced Learning Technology ALT***). Ovaj oblik obrazovanja se koristi kao dopuna klasičnog obrazovanja, ne njegova zamena. Prednosti ovog vida obrazovanja se ogledaju u mogućnosti učenja tokom celog života i van školskih ustanova, dok svakako i izostanak vremenskih odrednica tokom dana igra veliku ulogu. Ovaj vid obrazovanja se nalazi na svojoj lestvici rasta i tek će se zabeležiti njegov napredak u narednom periodu.

* + 1. **Mobilna trgovina**

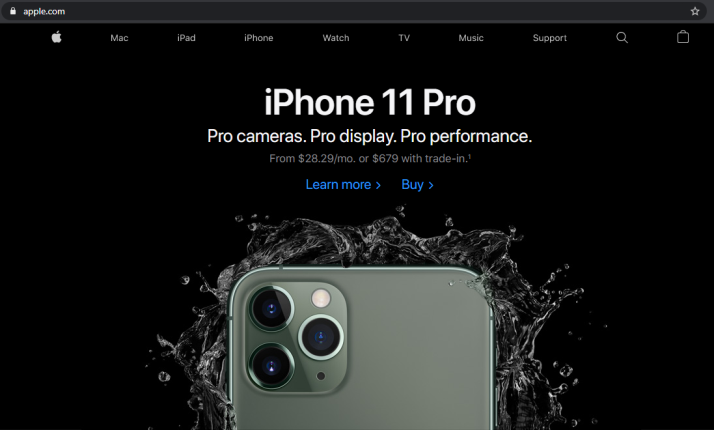
Mobilna trgovina (eng. ***m - commerce***) predstavlja model online trgovine putem bežičnih uređaja. Ovaj vid trgovine se zasniva na upotrebi mobilnih aplikacija za trgovinu, razmenu dobara, bankovnih transakcija i pristup plaćenim sadržajima i informacijama. Zastupljenost mobilnih uređaja i njihova današnja ekspanzija kako u upotrebi tako i u maksimumu razvojne tehnologije na kojoj počivaju doprinosi masovnosti ovog vida prodaje kao i popularnosti iste [6].

* 1. **Prikaz Webshop-ova**

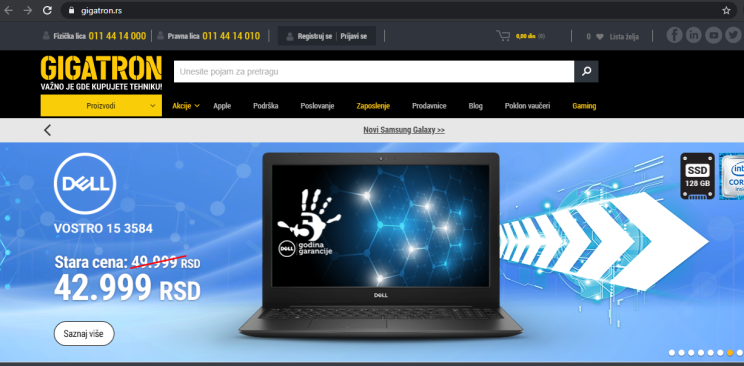


Slika 3. Amazon 1999. godine [7] Slika 4. Amazon 2020. godine [8]



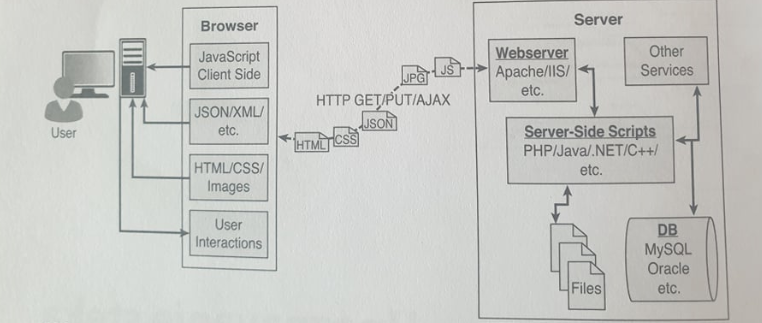


Slika 5. Apple 1997. godine [7] Slika 6. Apple 2020. godine [9]

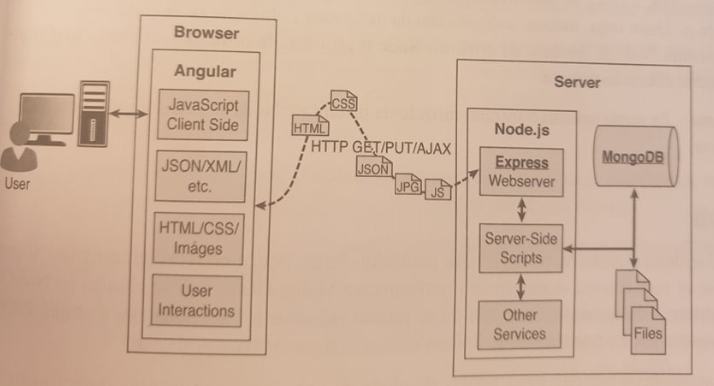
Usled kasnije pojave i razvitka Webshop-ova tj. prodavnica na srpskom tržištu, njihov današnji izgled se ne razlikuje u velikom od prethodnih godina.

Slika 7. Gigatron 2020. Godine [10]

1. **TEHNOLOGIJE**

U ovom poglavlju ćemo se osvrnuti na osnovne komponente website-ova. Nezavisno od razlika u izgledu i ponašanju website-ova, većina njih se sastoji od sledećih osnovnih komponenti, to su korisnik, pregledač, veb server i pozadinski serveri koje se nalaze u nekom obliku.

Slika 8. Dijagram osnovnih komponenti website-ova

U izradi ove Web aplikacije korišćene su komponente steka od Node.js do Angulara Node.js koji omogućavaju osnovnu platformu za razvoj. Komponente podrazumevaju Node.js, MongoDB, Express i Angular.js. Sve ove komponente se zasnivaju na jeziku JavaScript, što olakšava implementaciju i ponovo korišćenje koda na nivoima steka za razvoj veba. U daljem tekstu opisaćemo svaku od ovih tehnologija i njihovu primenu u osnovni model website-a/aplikacije.

Slika 9. Dijagram prikaza uklapanja Node.js, Express, MongoDB i Angular.js

* 1. **Node.js**

Tvorac Node.js je Ryan Dahl 2009. godine. Node.js predstavlja razvojno okruženje zasnovno na „Google-ovoj“ V18 JavaScript mašini kao serversko okruženje koje odgovara klijentskom okruženju u pregledaču. Programerima je omogućeno lakše premoštavanje jaza između klijenta i servera. Karakteriše se odličnim ekosistemom i kontinuirano se objavljuju nove ekstenzije. Jednostavno se instalira, konfiguriše i primenjuje. Nakon instalacije nije potrebna dodatna konfiguracija da biste započeli kreiranje Node.js aplikacija. Node.js se piše u JavaScriptu nakon čega biva kompajliran u mašinski kod ove mašine, koji se potom izvršava.

Popularizacija Node.js se brzo događa kao i njegovo favorizovanje od strane mnogih kompanija, pre svega zbog lakšeg održavanja i bržeg razvoja. Kompanije koje koriste tehnologiju Node.js:

* Yahoo!
* LinkedIn
* eBay
* New York Times
* Dow Jones
* Microsoft

Node.js se najčešće upotrebljava kao veb server. Može se koristiti i na različitim drugim veb serverima, poput:

* API-ji veb servisa, poput REST
* Igre u realnom vremenu za više igrača
* Pozadinski veb servisi (međudomenski i serviski zahtevi)
* Web aplikacije
* Komunikacija između više klijenata (IM)

U okviru Node.js mogu biti napisani mnogi pozadinski servisi, serverski skriptovi i sve podržane funkcije Web aplikacija. Prednost zasnivanja Node.js na JavaScript-u je mogućnost preuzimanja funkcija iz klijentskog skripta i njihovog smeštanja u serverski skript ili mogućnost funckionisanja veb servera direktno u okviru platforme Node.js kao Node.js modula.

Prednosti Node.js:

* **JavaScript** od jednog do drugog kraja – problem smeštanja logistike u klijentske skriptove, pri čemu dolazi do stvaranja glomaznog i nezgrapnog klijenta koji izvršava previše zadataka, ili smeštanja logistike u serverske skriptove kada dolazi do usporavanja Web aplikacija je rešen zahvaljujući okruženju Node.js koji omogućava preuzimanje JavaScript na klijentu i njegovo prilagođavanje serveru i obrnuto.
* Skalabilnost – jedinstvena logika za upravljanje veb zahtevima se ogleda u sposobnosti Node.js da obrađuje zahteve na istoj niti koristeći osnovni model događaja. Ova vrsta skalabilnosti, tj. **skalabilnost vođena događajima** omogućava skaliranje Node.js na način koji nije moguć na tradicionalnim veb serverima.
* **Proširivost** – postojanje aktivne zajednice Node.js programera kontinuirano omogućava i nudi nove module za proširenje funkcionalnosti njegovog okruženja.
* **Vreme** – za nekoliko minuta je moguće instalirati Node.js i dobiti radni veb server.
  1. **MongoDB**

MongoDB predstavlja skalabilnu NoSQL bazu podataka zasnovanu na modelu dokumenta u kome se objekti skladište kao zasebni dokumenti unutar kolekcije. Naziv potiče od reči „humongous“ (ogromno) čime se objašnjava skalabilnost i performanse MongoDb-a i njegovo zasnivanje na NoSQL modelu skladištenja dokumenata. Podaci se skladište u bazi podataka u obliku JSON objekata za razliku od tradicionalnih kolona i redova relacione baze podataka.

Koncept NoSQL (Not Only SQL) podrazumeva skladištenje i korišćenje podataka bez ograničenja tradicionalnih SQL modela relacionih baza podataka. NoSQL omogućava programerima implementaciju modela na način koji se više uklapa u protok podataka potreban za njihove sisteme.

Korišćenjem MongoDB-a omogućeno je pozadinsko skladište za veb sajtove sa gustim saobraćajem, kod kojih postoji potreba skladištenja podataka tipa korisničkih komentara, blogova, informacija o korisničkom nalogu i artikala na kupovnoj kartici. Node.js podržava različite programe za upravljanje za pristup DB drajverima. Pokazatelji dobrog uklapanja MongoDB-a u Node.js stek:

* Orijentacija dokumenata – MongoDB je orijentisan ka dokumentu i podaci se čuvaju u bazama podataka u formatu sličnom koji se koristi na serverski i klijentskim skriptovima.
* Visoke performanse – MongoDB predstavlja dostupnu bazu podataka sa najvišim performansama.
* Visoka dostupnost – olakšava održavanje skalabilnosti uz zadržavanje visokih performansi.
* Visoka skalabilnost – olakšano horizontalno sklairanje deljenjem podataka na više servera.
* Bez SQL injektovanja – MongoDB baza podataka nije podložna SQL injektovanju (nije ugrožena bezbednost baze podataka).

Prvi korak je instaliranje MongoDB servera. Nakon ovoga se pristupa pokretanju programa baze podataka. Nakon instalacije, konfiguracije i pokretanja moguće mu je pristupiti pomoću MangoDB komandnog okruženja. Zatim je omogućeno dalje upravljanje bazama podataka i nalozima korisnika.

* 1. **Express**

Modul Express se ponaša kao veb server u steku do Node.js do Angular.js. Funkcioniše u okruženju Node.js čime se olakšava konfiguracija, implementacija i kontrola. Sam modul proširuje Node.js obezbeđivanjem važnih komponenata za upravljanje veb zahtevima. Express proširuje funkciju modula http kako bi se lakše upravljalo rutama servera, odgovorima, „kolačićima“ i statusima HTTP zahteva.

Ovaj modul obezbeđuje podešavanje URL-ova (odredišnih ruta) sa kojima se povezuju korisnici. Takođe, omogućava i odlične funkcije prilikom korišćenja HTTP zahteva uključujući i „kolačiće“.

Funkcije Expressa:

* Upravljanje rutiranjem – lako definisanje rute koja se direktno povezuje sa funkcijama Node.js skripta na serveru.
* Obrada grešaka – u sebi ima ugrađenu obradu greške koja se javlja kada dokument nije pronađen.
* Jednostavna integracija – lako se implementira na u pozadini postojećeg reverznog proksi sistema, poput Nginx ili Varnish-a.
* Kolačići – omogućava jednostavno upravljanje njima.
* Omogućava upravljanje sesijama i keširanjem.
  1. **Angular.js**

Angular je razvijen od strane kompanije „Google“ i predstavlja JavaScript okruženje na strani klijenta. Sve funkcije potrebne za upravljanje korisničkim unosom u pregledaču, manipulisanje podacima na strani klijenta i za kontrolisanje prikaza elemenata u pregledaču su obezbeđene Angularom. Angular se takođe piše pomoću JavaScripta i olakšava implementaciju aplikacija koristeći MVC/MVVM radni okvir.

Angular obezbeđuje odličan radni okvir koji ubrzava i olakšava kreiranje klijentskih JavaScript aplikacija. Za rad u Angularu potrebno je:

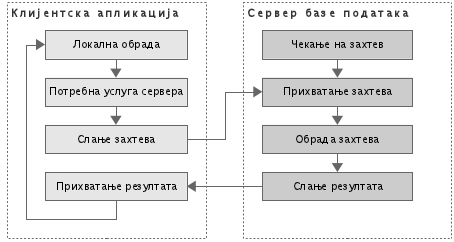
* Angular biblioteka za kreiranje Angular aplikacija
* Veb server za opsluživanje datoteka u pregledaču
* Transpajler za konvertovanje TypeScript koda u JavaScript
* „Čuvar“ koji obaveštava transpajler kada dođe do promene datoteke
* Uređivač teksta u kome ćete pisati kod

Prednosti Angulara:

* Jednostavno povezivanje podataka sa HTML elementima preko mehanizma oblasti važenja.
* Angular arhitektura dozovljava lako proširenje skoro svakog aspekta jezika.
* U Angularu ste prinuđeni da pišete čist logički kod.
* Olakšano pisanje koda za ponovnu upotrebu.
* Podrška od strane kompanije „Google“ koja mnogo ulaže u Angular projekat.
* Kompatabilnost – zasnovan i usko povezan sa JavaScriptom omogućava ponovno korišćenje postojećih delova koda unutar strukture radnog okvira Angulara [11].

1. **ARHITEKTURE**
   1. **Klijent-Server Arhitektura**

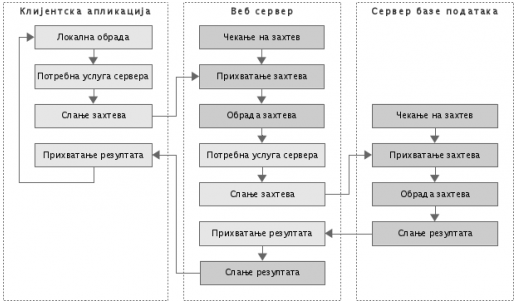
Klijent-server arhitektura predstavlja jedan od danas najčešće korišćenih pristupa kod distribuirane obrade podataka. Koreni ove arhitekture nalaze se kod mejnfrejm računara i na njih priključenih terminala. Sličnost sa ovom arhitekturom jeste postojanje jednog člana sposobnog za izvršavanje zadataka koji su van mogućnosti ostalih članova mreže. Postoje, međutim, bitne razlike između ove dve arhitekture. Kod mejn-frejm arhitekture terminali nemaju nikakvu mogućnost obrade podataka, dok kod klijent-server arhitekture klijenti od servera dobijaju podatke koje, zatim, koriste u lokalnom procesu obrade. Zatim, mejn-frejm računari predstavljaju autonomne članove mreže koji za proces obrade podataka koriste lokalne resurse. Nasuprot tome, server se u vidu klijenta može obratiti drugim serverima u mreži za određeni resurs ili distribuiranu obradu.



Slika 10. Dvoslojna kijent-server arhitektura

Osnovna klijent-server arhitektura podrazumeva postojanje dva sloja - sloja klijenta i sloja servera. Kod dvoslojne klijent-server arhitekture softversko rešenje se razlaže na jednu isključivo klijentsku, i jednu isključivo serversku komponentu. Primer osnovne, dvosjlojne klijent-server arhitekture jeste mreža u kojoj korisnici putem aplikacija na svojim računarima pristupaju centralnoj bazi podataka.

U praksi su česte i klijent-server arhitekture sa tri sloja jer se njima lakše rešavaju pitanja zaštite podataka i kontrole pristupa. Na primer, Veb aplikacije najčešće koriste troslojnu klijent-server arhitekturu na takav način da osnovni klijentski deo čini Veb brauzer, srednji sloj čini aplikativni Veb server, a server baze podataka čini treći, serverski sloj. U takvoj organizaciji srednji sloj je realizovan kao serverska komponenta, gledano sa aspekta Veb brauzera, odnosno kao klijentska komponenta, gledano sa aspekta servera baze podataka.

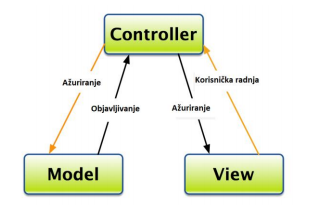


Slika 11. Troslojna klijent-server arhitektura

Na primeru troslojne arhitekture jasno se može steći uvid da se pojmovi „klijent“ i „server“ ne odnose na hardverske karakteristike računara, već na tipove procesa, ili čak njihove procedure, koji se na tim računarima izvršavaju. Ukoliko neki proces, ili jedan njegov deo, zahteva uslugu od drugog procesa (bilo na lokalnom ili na udaljenom računaru) njegovo ponašanje spada u klijentski deo klijent-server arhitekture. I obrnuto, ukoliko neki proces čeka na zahteve drugih procesa, njegovo ponašanje spada u serverski deo klijent-server arhitekture. Opšte pravilo za određivanje uloge kod klijent-server arhitekture definiše da je klijentska strana ona koja od druge strane zahteva uslugu, a da je serverska strana ona koja čeka na zahtev od druge strane i isporučuje odgovor na njega.

* 1. **MVC Arhitektura**

MVC arhitektura (engl. Model-View-Controller) je softverski šablon (engl. software pattern) koji razdvaja aplikaciju na tri međusobno povezana dela. Klasičan MVC pristup smešta svu logiku na web server, dok su tehnologije (Ajax, AngularJS, EmberJS, JavaScriptMVC, Backbone) omogućile da se MVC komponente delimično izvršavaju i na strani klijenta. Primenom MVC-a se pruža mogućnosti prikaza modela na različite načine, a olakšani su procesi dodavanja novih ili izmene postojećih izveštaja, izmene interakcije sa korisnikom kao i izmene poslovne logike. Model – Obrađuje podatke dobijene putem Controller-a. View – Prikaz podataka, shodno promenama u modelu. Controller – Tumači zahtev korisnika i prosleđuje ga na Model ili View komponente

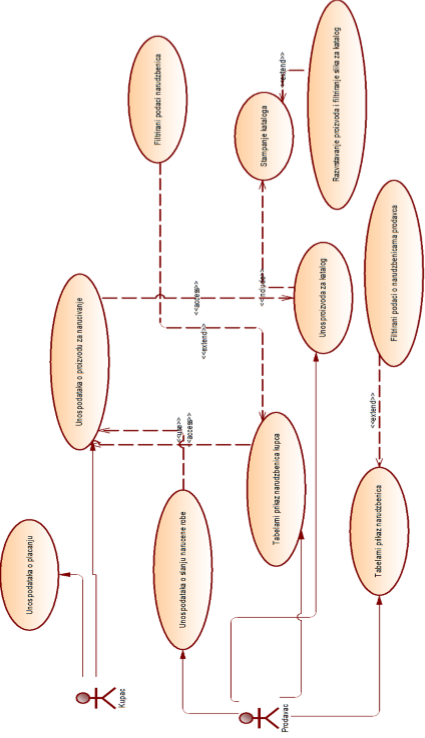


Slika 11. Prikaz rada MVC modela

Ukratko, MVC arhitektura deli aplikaciju u tri osnovne komponente Model, Pogled (View) i Kontroler (Controller). Takva podjela omogućuje razdvajanje programiranja same obrade podataka od programiranja sučelja i zaprimanja zahteva korisnika, čime se omogućava razvoj aplikacije u nekom od objektnih programskih jezika, a u isto vreme i kontrola nad HTML-om, CSS-om i JavaScript-om s pomoću kojih se izgrađuju Pogled i Kontroler. Ova napredna tehnologija omogućuje razvoj od jednostavnih do vrlo složenih aplikacija, budući da se model potpuno odvaja od sučelja i kontrolera, koji se mogu razvijati zasebno, a zatim spojiti u jednu aplikaciju. Da se podsjetimo, MVC aplikacija komunicira s korisnikom putem Pogleda (eng. View), gde korisnik unosom zahtjeva šalje naredbe Kontroleru (eng. Controller), dok Kontroler od Modela traži da obavi radnju i vrati mu rezultat. Rezultat radnje zatim Kontroler šalje u Pogled gde ga korisnik može videti. Neke od karakteristika (ujedno i prednosti) MVC aplikacija su: Olakšano upravljanje razvojem složenih aplikacija zbog mogućnosti posebnog razvoja svake od komponenti aplikacije, čime se olakšava organizacija posla razvoja aplikacije, a takođe i nadogradnja novim verzijama (ponekad je češće potrebno unaprijediti Poglede i Kontrolere, a Model ostaje isti), Olakšano testiranje aplikacije (tzv. Test-Driven Development - TDD), jer je moguće odvojeno testirati pojedine komponente aplikacije i nije uvijek za testiranje potrebno koristiti web server (neke komponente se mogu testirati lokalno), Olakšano procesiranje zahteva koje šalje korisnik, jer se za tu svrhu koristi Front Controller koji kroz jedan kontroler šalje sve zahtjeve korisnika. Kada je dobro koristiti MVC arhitekturu? Kada aplikaciju razvija veći tim developer-a i kada dizajneri trebaju veći stepen kontrole nad ponašanjem aplikacije. Nasuprot tome, ako se ne želi koristiti MVC, web aplikacije se mogu razviti i kao Web form aplikacije. One su pogodne za manje složene aplikacije koje razvija manji tim developer-a.

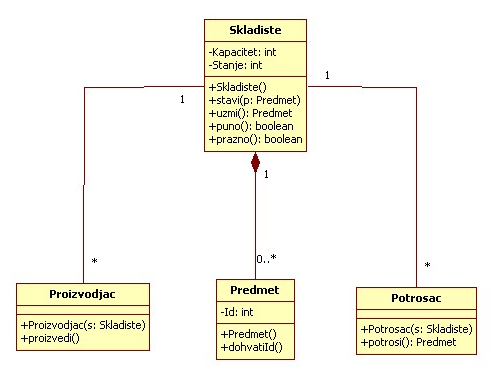
1. **UML DIAGRAMI**

Use Case Diagram služi za prikazivanje informacija o tome kako neki poslovni sistem funkcioniše trenutno. Sadrži alate koji omogućavaju modelovanje funkcionisanja poslovnih procesa.



Slika 12. UML use case diagram

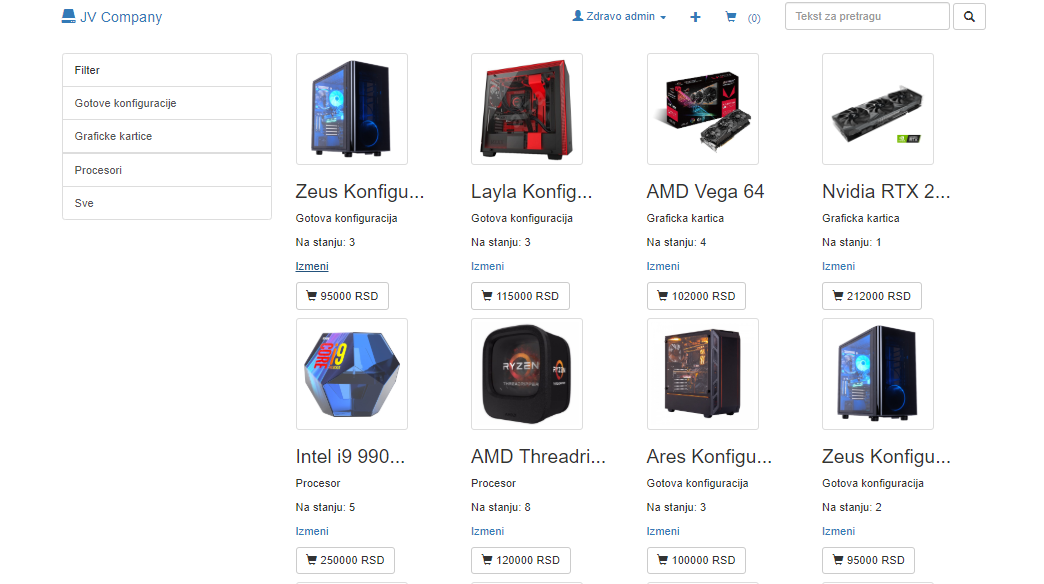
Dijagram klasa daje grafički prikaz veza između tipova objekata, vrste veza kao i spisak svojstava (atributa) i operacija (metoda-funkcija) prikazanih tipova.



Slika 13. UML Klasni diagram

1. **KORISNIČKO UPUSTVO**

U ovom delu je dat primer web aplikacije i demonstrativnog je tipa i takođe nije u potpunosti funkcionalna.Primer je urađen da bi se prikazalo stečeno znanje. Aplikacija nije u potupunosti završena i ima prostora za usavršavanje.

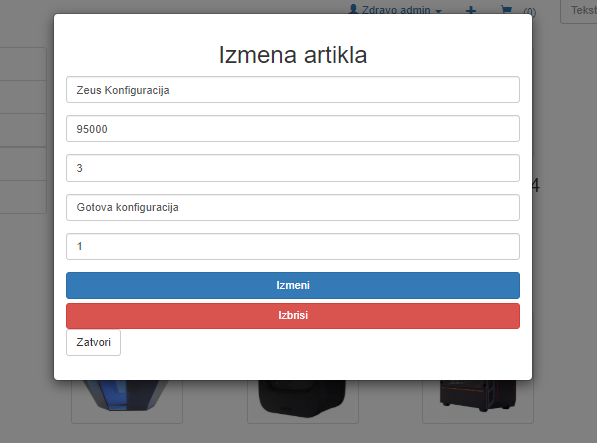


Slika 14. Prikaz web aplikacije

U ovom konkretnom primeru je napravljen web shop za prodaju računara. Da bi ste mogli koristi morate biti prijavljeni.

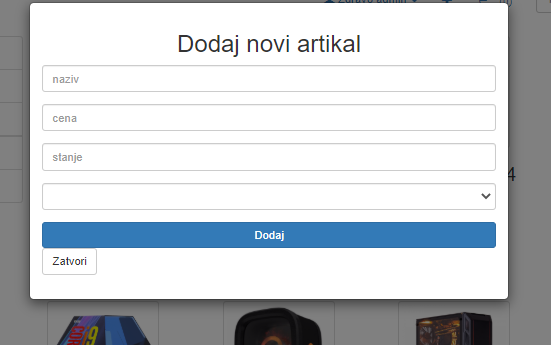
Login forma ima dva statusa korisnika, a to su status Administrator i status Korisnik (kupac). U zavisnosti od toga da li se prijavite kao administrator ili kao korisnik, biće omogućen prikaz odgovarajućeg (različitog) sadržaja. Da bi ste bili u mogćnosti da se prijavite morate se prvo registrovati.

Kada ste prijavljeni kao administrator u mogućnosti ste da menjate proizvod da ga dodajete ili brišete.

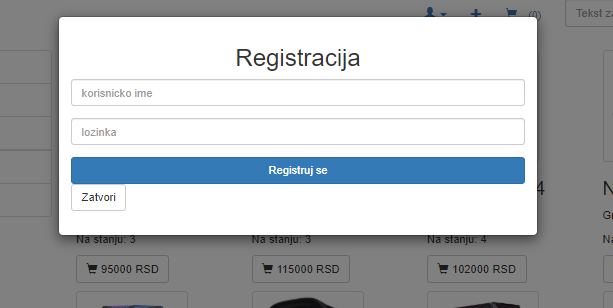


Slika 15. Prikaz dela sa izmenom artikla za administratora

Administrator može menjati svaku od kategorija proizvoda i tako izmeniti opis, cenu, stanje i količinu a može ga i izbrisati u slučaju da je potrebno. U delu za dodavanje artikla treba da unese parametre i doda novi artikal.

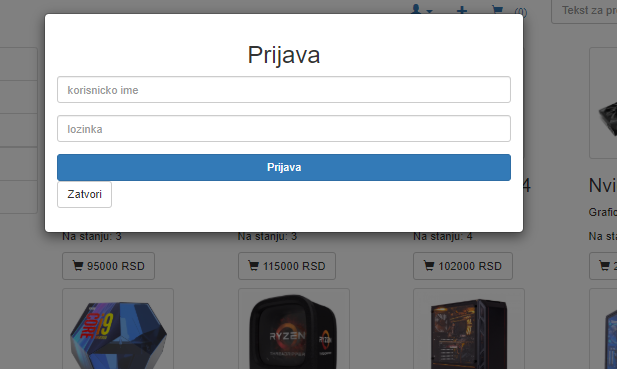


Slika 16. Prikaz dela za dodavanje novog artikla



Slika 17. Izgled forme za registraciju

Kada se registrujete kao novi korisnik, prijavite se i možete započeti kupovinu.



Slika 18. Izgled forme za prijavljivanje

Sam izgled web prodavnice je napravljen da bude lak za korišćenje, sa leve strane se nalazi opcija sa proizvodima gde vi birate shodno onome šta vama treba.



Slika 19. Filter proizvoda

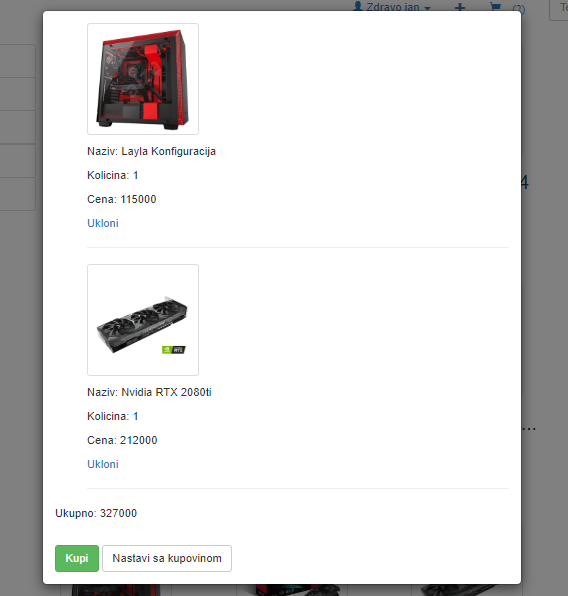
Ukoliko tražite nešto određeno a ne možete da pronađete u ovom meniju napravljen je i deo sa petragom gde treba samo da uneste naziv artikla koji tražite.



Slika 20. Pretraga

Kada ste odlučili šta želite i kliknuli da želite to da kupite sledi korpa. Sama korpa vam dozvoljava da brišete iz nje ako ste promenili mišljenje i ilustrativno vam pokazuje sta je u njoj i koja je cena.

Kada ste odlučili šta želite i kliknuli da želite to da kupite sledi korpa. Sama korpa vam dozvoljava da brišete iz nje ako ste promenili mišljenje i ilustrativno vam pokazuje sta je u njoj i koja je cena.



Slika 21. Korpa

Kada kliknete da dugme “Kupi” i završite kupovinu prošli ste celi ciklus kupovine od registracije, prijavljivanja, pretrage i kupovine.

1. **PRIMERI KODA**

**7.1 Korpa**

U ovome delu koda cemo objasniti kako funkcionise pretraga.kada korisnik zeli da nadje zeljeni produkt, klikom na sekciju pretraga (eng. search) posle unosa naziva produkta

rezulat pretrage ce biti specifican trazeni produkt bez potrebe za trazenjem po kategorijama.

**$scope.searchText = "";**

**$scope.search = function () {**

**$scope.filteredProducts = [];**

**if ($scope.searchText.length == 0) {**

**$scope.filteredProducts = $scope.products;**

**} else {**

**for (var item of $scope.products) {**

**if (item.naziv.toLowerCase().match($scope.searchText.toLowerCase())) {**

**$scope.filteredProducts.push(item);**

**}**

**}**

**}**

**};**

**7.2 Registracija i prijava**

U ovome delu koda cemo objasniti kako funkcionise registracija i prijava.Da bi ste mogli kupovati na web prodavnici morate prvo da se registrujete.Sama registracija se sastoji od

kreiranja unikatnog korisnickog imena i lozinke.Da bi kasnije to korisnicko ime i lozinku koristili pri prijavljivanju na sam sajt i uspesno zavrsili kupovinu.

**$scope.login = function (login) {**

**for (user of $scope.users) {**

**if (login.user == user.name && login.password == user.password) {**

**$scope.loggedIn = true;**

**$scope.user = user;**

**$scope.closeModal();**

**login.user = "";**

**login.password = "";**

**login.errorMessage = "";**

**return;**

**}**

**}**

**login.errorMessage = "Niste dobro uneli podatke.";**

**};**

**$scope.register = function (register) {**

**var user = { "name": register.user, "password": register.password };**

**$scope.users.push(user);**

**//upis za bazu**

**register.errorMessage = "Uspesno ste se registrovali sa korisnickim imenom " + register.user + " i lozinkom " + register.password;**

**}**

**7.3 Korpa**

u ovom delu koda cemo objasniti kako funkcionise korpa.Kada se odlucite za odredjeni produkt on ce se prikazati u vasoj korpi.Gde finalizujete vasu kupovinu i proveravate da

li vam odgovara ono sto se hteli da kupite

**$scope.addToCart = function (item) {**

**if ($scope.loggedIn == false)**

**return $scope.openInfoModal("Greska: Morate biti prijavljeni da bi ste kupovali.");**

**if (item.stanje < 1)**

**return $scope.openInfoModal("Nema toliko na stanju.");**

**var cartItem = angular.copy(item);**

**for (var i of $scope.cart) {**

**if (i.naziv == item.naziv) {**

**if (i.stanje < item.stanje) {**

**i.stanje++;**

**$scope.cartPrice += i.cena;**

**return;**

**} else {**

**return $scope.openInfoModal("Nema toliko na stanju.");**

**}**

**}**

**}**

**cartItem.stanje = 1;**

**$scope.cartPrice += cartItem.cena;**

**$scope.cart.push(cartItem);**

**};**

1. **ZAKLJUČAK**

U radu je prikazan proces razvoja Web aplikacije za online prodaju računara. Obuhvaćen je razvoj aplikacije od same specifikacije zahteva i analize semantike rada, pa sve do kreiranja softvera. Posebno je dat naglasak na elementima višeslojne arhitekture softvera.

Za izradu web aplikacije korišćenje su sledeće tehnologije:

* AngularJS
* MongoDB
* ExpressJS
* NodeJS

Softver je realizovan kao prototip, sa mnogim mogućnostima unapređenja.

Kao sto su naprimer:

* Token autentifikacija
* Preporucivanje proizvoda na osnovu popularnosti
* Pracenje proizvoda od momenta isporucivanja do momenta dostave
* Ocenjivanje proizvoda
* Filtriranje spama po web prodavnici
* Customer support

**Literatura**

1. <https://www.ebayinc.com/company/our-history/>
2. Končar J., „Elektronska trgovina“, Ekonomski fakultet, Subotica, 2003.
3. Božidar R., Marijana Despotović-Zrakić, Zorica B., „Elektronsko poslovanje“, Fakultet organizacionih nauka, Beograd, 2015.
4. Stankić R., „Poslovna informatika“, Ekonomski fakultet, Beograd, 2001.
5. Milosavljević, M. Mišković, V., „Elektronska trgovina“, Univerzitet Singidunum, Beograd, 2011.
6. Uroš T., „Elektronsko poslovanje“, Beogradska poslovna škola, Beograd 2006.
7. <http://home.bt.com/tech-gadgets/internet/websites/the-internet-time-machine-how-your-favourite-websites-used-to-look-11363950974005?fbclid=IwAR1J84fKHaQyzcCUSKys_2DDMV1EJO13aPmFm06o-RlLexAEreg13imR86g>
8. <https://www.amazon.com/ref=nav_logo>
9. <https://www.apple.com/>
10. <https://www.gigatron.rs/>
11. Brad Dayley, Brendon Dayley, Caleb Dayley, „Node.js, MongoDB i Angular – integrisane alatke zatazvoj veb strana“, drugo izdanje, Kompjuter biblioteka, Beograd 2018.