

Spletna trgovina

Poklicna matura

Mentor: Andrej Zadnik, prof. Avtor: Luka Rosić, R4.A

Ljubljana, april 2017

**Povzetek**

V tej nalogi Vam bom predstavil svoj izdelek za poklicno maturo z naslovom spletna trgovina. Spletna trgovina je namenjena prodaji moških oblačil. Šli bomo čez vsa področja izdelave spletne trgovine. Govoril bomo od kje ideja za prav ta naslov, kako sem začel, na kakšne težave sem naletel med delom, na katerih področjih je bilo moje znanje premalo. Opisal bom najtežje dele izdelave spletne trgovine, in kako sem rešil te težave. Izkazalo se je, da sem veliko naredil brez kakšnih večjih problemov, vendar so se vseeno našli deli izdelave, za katere sem rabil malo več pomisliti in se dobro vprašati kako to narediti.

Ključne besede: spletna trgovina, SQL, PHP, xampp, HTML, podatkovne baze, spletna aplikacija

**Abstract**

In this project I am going to talk about my matura project spletna trgovina. I will go through all of the parts of building an online store. I will talk about where we got the idea from, how did we begin, what problems did I encounter, on which parts was my lack of knowledge in this field too big. We will describe the hardest parts of the project and how I got over these problems. It turned out that for the most part of the project there weren't any big problems, but there were still certain bits of the project which had to be well examined before doing anything.

Key words: web store, SQL, PHP, xampp, HTML, databases, web a

**Kazalo vsebine**

[1. Uvod 4](#_Toc479878610)

[2. Metodologija 5](#_Toc479878611)

[2.1 Oprema potrebna za izdelavo 5](#_Toc479878614)

[2.2 Postopek dela 6](#_Toc479878615)

[3. Navodila za uporabnike 15](#_Toc479878616)

[4. Zaključek 18](#_Toc479878617)

[5. Viri 19](#_Toc479878618)

**Kazalo slik**

[Slika 1: Primer spletne trgovine (ebay.com) 4](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877019)

[Slika 2: Deli xampp-a 5](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877020)

[Slika 3: Klicanje funkcije za novega uporabnika 7](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877021)

[Slika 4: Iskanje pravega uporabnika iz tabele login 7](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877022)

[Slika 5: Shranjevanje v sejo 8](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877023)

[Slika 6: Meni 8](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877024)

[Slika 7: Meni ko ga zožamo dovolj 8](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877025)

[Slika 8: Vsa koda v odjavi.php 9](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877026)

[Slika 9: Pogled na meni s strani admina 9](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877027)

[Slika 10: Koda za prevrjanje admina 10](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877028)

[Slika 11: Poizvedba za vstavljane v tabelo izdelek 10](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877029)

[Slika 12: Štetje izdelkov 10](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877030)

[Slika 13: Pridobivanje vseh izdelkov 10](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877031)

[Slika 14: Izpis enega izdelka 11](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877032)

[Slika 15: Pidobivanje košarice ter vseh izdelkov iz tabele 11](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877033)

[Slika 16: Izpis izdelkov, ki jih je stranka izbrala 12](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877034)

[Slika 17: Poizvedba za id stranke 12](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877035)

[Slika 18: Dodajanje nakupa 12](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877036)

[Slika 19: Zadnji koraki zapisa nakupa 13](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877037)

[Slika 20: Zapis v postavko 13](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877038)

[Slika 21: Poizvedba za zadnji nakup 13](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877039)

[Slika 22: Pred in po dodanem izdelku 14](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877040)

[Slika 23: Koda za košarico v meniju 14](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877041)

[Slika 24: Stavek za preprečitev dostopa brez prijave 14](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877042)

[Slika 25: Funkcija za spremembo podatkov 15](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877043)

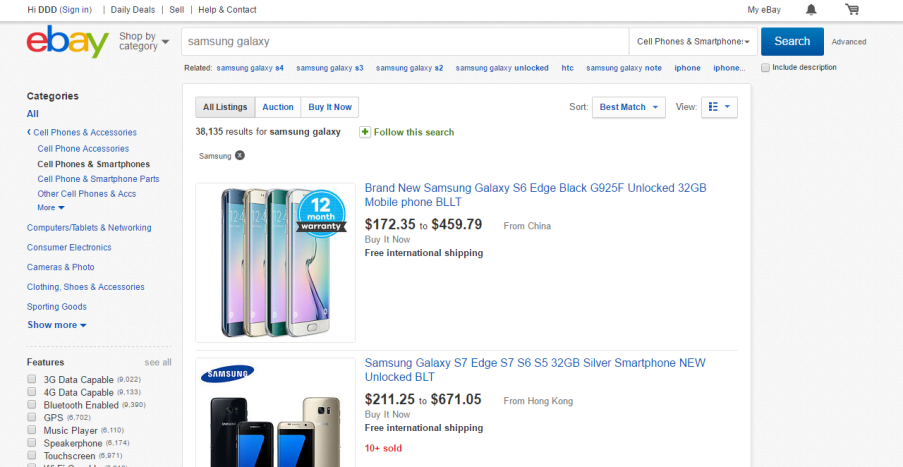
[Slika 26: Prijavno okno 15](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877044)

[Slika 27: Okvir izdelka 16](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877045)

[Slika 28: Sprememba ikone po dodajanju 16](file:///C:\xampp\htdocs\trgovina\Spletna%20trgovina.docx#_Toc479877046)

# Uvod

Najbolj znane spletne trgovine danes kot so ebay in amazon so nastale že leta 1993 in 1994. Takrat izbira ni bila velika, zato so si jih ljudje tudi dobro zapominili. Danes so že povsem vsakdanje in veliko ljudi se je že srečalo z njimi. Sestavljene so iz nekih izdelkov ali storitev, ki nam jih človek za to trgovino ponuja. So zelo priročne za ljudi, ki nimajo volje se voziti do trgovine ali pa so preprosto preazaposleni. So zelo lahke za uporabo, če so pravilno narejene. Spletne trgovine nam oljašajo, ko želimo kupiti nek izdelek in se lahko hitro pozanimamo, kakšna je cena pri drugih ponudnikih brez, da bi se kamor koli premaknili. Vse kar potrebujemo za nakup je dostop do interneta, kar danes ni nek velik problem. Slaba stran spletnih trgovin je, da sami nemoremo zagotoviti varnosti svojih podatkov. Če imamo novo trgovino, je dosti pomembno, da imamo tudi spletno trgovino, ker se pri večini rast zaradi tega hitro poveča. Za svojo maturitetno nalogo sem se tako odločil, da je to tema, ki jo bom izbral. Glavni razlog za tem je, da sem se malo s to temo že spoznal med lanskim praktičnem usposabljanju za delo. Takrat sem se ukvarjal s spletnimi trgovinami, vendar le na podatkovnem področju. Odločil sem se, da bom ta projekt dokončal s spletno stranjo, ki se bo povezala z mojo bazo. Namen moje maturitene naloge je predstaviti zgradbo moje spletne trgovine, reševanje problemov, pokazati težje dele izdelave, itd. Izdelki, ki bi jih moja spletna trgovina ponujala so moška oblačila.



Slika : Primer spletne trgovine (ebay.com)

# Metodologija



## Oprema potrebna za izdelavo

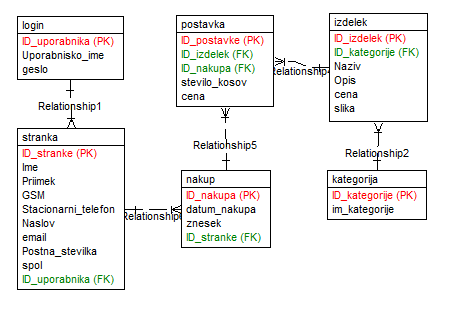
Za izdelavo svoje spletne trgovine nisem potreboval veliko elemntov. Potreboval sem orodje Case studio. To je orodje, s katerim narišemo svoje entitetno relacijske modele. ER modeli so zelo pomembni pri projektih kot je moj, saj nam skozi celotno izdelavo pomaga, da si lažje vizualno predstavljamo kaj s čim povezati. Program je tudi zelo uporaben po izdelavi ER modela, saj lahko z njem generiramo SQL kodo, s katero se nato ustvari dejansko bazo v programu naše izbire. Jaz sem za ta del uporabil zelo močno orodje z imenom xampp. To orodje je sestavljeno iz treh glavnih delov: Apache strežnik, MariaDB, PHP tolmač. Vsi trije deli so bili potrebni za moj projekt. Apache je trenutno najbolj popularen HTTP strežnik, ki prikazuje tvoje datoteke na izbranem brskalniku. MariaDB je samo novejša verzija MySql-a. MySql je sistem za upravljanje s podatkovnimi bazami, ki uporablja jezik SQL. SQL ali struktuirani povpraševalni jezik za delo s podatkovnimi bazami (angl. Structured Query Language) je najbolj razširjen in standardiziran jezik za delo z bazami. Z njim lahko beremo podatke iz baze (SELECT), spreminjamo podatke (UPDATE), brišemo podatke iz baze (DELETE), vnašamo podatke (INSERT) in še mnogo stvari, ki se ne tiče podatkov. Na koncu imamo še PHP, ki je prav tako zelo razširjen programski jezik, ki se uporablja za razvoj dinamičnih spletnih strani. Je zelo podoben jeziku C++, ki smo ga vzemali v šoli. PHP se izvaja na našem spletnem strežniku. Kot vhodni paramater vzema našo izvorno kodo, vrne nam pa spletno stran, ki smo jo napisali. Izbral sem najbolj razširjena orodja, saj to pomeni, da si lahko veliko pomagamo z internetom, če se kaj kje ustavi. Za projekt je bilo potrebno vsaj neko osnovno znanje s programiranjem v SQL, PHP, HTML. Xampp vsebuje zelo doboro orodje phpmyadmin. Z njem imamo vpogled na vse baze, ki se nahajajo na našem strežniku in omogoča zelo preprosto spreminjanje tabel, če ti SQL ni pri srcu si lahko s tem orodjem zelo pomagamo.



Slika : Deli xampp-a

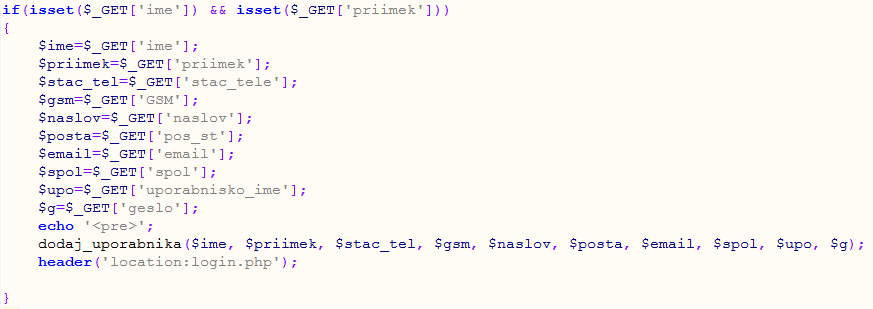
## Postopek dela

Začel sem izdelavo svoje spletne strani najprej z vidika podatkovnega modela. Z orodjem case studio sem narisal ER model baze, ki sem jo potreboval.



Moja baza je sestavljena iz šestih tabel kot je razvidno na sliki (login, postavka, stranka, nakup, izdelek in kategorija). Za vsako od tabel sem vnesel vse potrebne atribute in njihovo tip zapisa (besedilo, število, ...). Bilo je potrebno dobro razmisliti vnaprej kaj vse bom potreboval, da nisem slučajno kaj pozabil. Ko sem se odločil, da je to moj končni izgled baze, sem generiral .sql datoteko, ki je vsebovala kodo za to bazo. To .sql datoteko sem nato uvozil v phpmyadmin in naredil svojo bazo z imenom spletnatrgovinav2. Struktura modela se je tam končala in na vrsti je bila izdelava strani ter povezava strani z bazo.

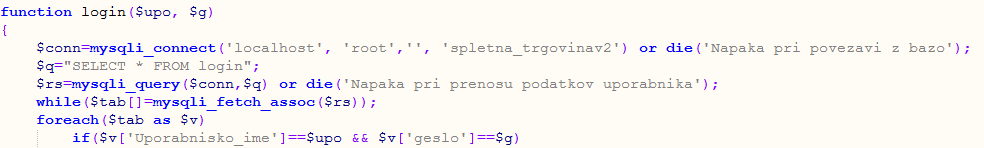
S strani programiranja sem se odločil, da bom začel s kreiranjem uporabnika. Forma je vsebovala vse atribute, ki smo jih potrebovali za stranko: ime, priimek, stacionarni telefon, gsm telefon, naslov, poštno številko, email, spol, novo uporabniško ime in geslo. Prva posebost te forme je bila, da za spol lahko izbiramo le med m in ž. Druga posebnost je bila, da je geslo, ko si ga vnašal bilo skrito in si lahko videl samo dolžino geslo s pikami. Forma se sklicovala nazaj nase in vsakič preverila če se nahaja v tabeli $\_GET nova spremenljivka $\_GET['ime'] in $\_GET['priimek']. Ko sta se te dve spremenljivki znašli v tabeli $\_GET, se iz nje shranijo v lokalne spremenljivke z enakim imenom. Ko se je vse sharnilo, se pokliče iz datoteke funkcije.php funkcijo nov\_uporabnik(). Kot vhodne parametre dobi vse lokalne spremenljivke v določenem vrstnem redu. Funkcija za identifikator stranke ne potrebuje paramatera, ker se povečuje sam (tipa AUTO\_INCREMENT). Najprej se v funkciji izvrši SQL stavek, ki vse te podatke shrani v tabelp stranka, za tem pa še podatka uporabniško ime in geslo v tabelo login. Za tem še pošlje id\_login v tabelo stranka. To je potrebno, da vemo od katere stranke je katero uporabniško ime in geslo.



Slika : Klicanje funkcije za novega uporabnika

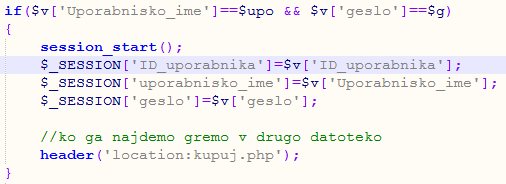
Zadnjo vrstico header('location:login.php') sem dodal, ko sem naredil datoetko login.php. Torej, ko se je naredil uporabnik, te je stran sama poslala na prijavo, kjer si se lahko vpisal s svojim novim uporabniškim imenom in geslom.

Naslednja zadeva, ki je bila narejena je bila stran za prijavo. Prav tako vsebuje formo. Sestavljena je iz štirih elementov: vnosno polje za uporabniškega imena in geslo, gumb s katerim se prijavimo ter povezava: ''Ustvari nov račun''. Povezava te pošlje na prejšno stran, kjer smo ustavrjali uporabnike. Prijavimo se lahko le z gesli in uporabniškimi imeni, ki se nahajajo v tabeli login. Če vneseš napačno geslo ali uporabniško ime, ti stran to tudi izpiše. Pri tej formi sem uporabil metodo post, ker se mi je zdela bolj primerna, saj nepokaže kaj smo vnesli kasneje v URL-ju. Funkcija, ki se kliče ko pritisnemo gumb prijava, najprej najde vse zapise, ki so v tabeli login in jih preverja s tistim kar smo vnesli.



Slika : Iskanje pravega uporabnika iz tabele login

Ko najde pravilnega, začne sejo in v sejo, v tabelo $\_SESSION shrani tri spremenljivke: idenitifikator uporabnika, uporabniško ime in geslo. Za tem nas pošlje na našo prvo stran kupuj.php.

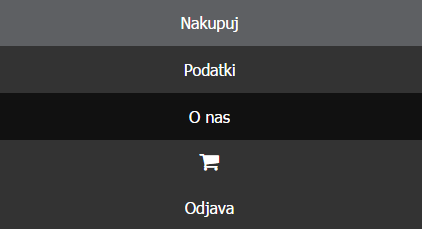


Slika : Shranjevanje v sejo

Za naslednji korak sem se odločil, da bom naredil svoj meni in vsaj približno obliko strani. Odločil sem se za temnejšo temo. Sivo ozadje med tem, ko je meni črn z belimi napisi. Meni je sestavljen iz petih elemntov: kupuj, podatki, o nas, košarica in odjava. Ko ekran dovolj zmanjšamo se oblika menija spremeni.

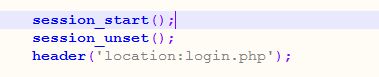


Slika : Meni



Slika : Meni ko ga zožamo dovolj

Najprej sem se odločil, da bom šel na najlažji del prvo, to je odjava. Vse kar je bilo potrebno narediti v odjavi je začeti sejo in zbrisati vse kar je v njej in poslati na prijavno stran.



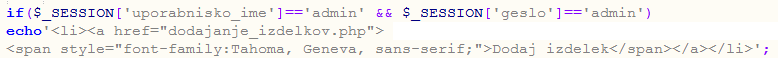
Slika : Vsa koda v odjavi.php

Ker sem se odločil, da bom šel od lažjega dela proti težjemu sem na drugo mesto postavil izdelavo strani o nas. Ta stran nima nekega posebnega pomena. Zdelo se mi je, da stran mora biti tam, saj jo ima vsaka spletna trgovina. Ta stran je sestavljena le iz HTML kode in ni nič posebnega kar se tiče programiranja.

Naslednji korak je bil vnos izdelkov. To sem naredil, da bi lahko napolnil svojo prvo stran. Ponovno sem izdelal vnosno formo, ki vsebuje toliko vnosnih polj kot jih rabimo za vsak izdelek: naziv, opis, cena, pot slike in kategorijo. Kategorije sem lastno ročno vnesel v bazo in naj se ne bi spreminjale. Zato sem lahko naredil statičen izbirni meni za kategorijo pri formi. Pri sliki sem se odločil, da bom raje sharnjeval v bazo pot slike in ne sliko celo. Razlog za tem je, da sem naletel na neke probleme pri kasnejšem prikazovanju slike in sem se odločil za alternativo. Stran za dodajanje izdelka lahko uporablja samo admin. Admina sem naredil tako, da sem naredil novega uporabnika z imenom in geslom admin. Na vsaki strani potem pogleda prvo, če je uporabniško ime in geslo, ki je shranjeno v seji enako adminu. Če to drži, dobi meni še eno stran z imenom dodaj izdelek.



Slika : Pogled na meni s strani admina



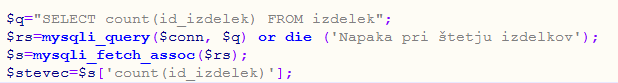
Slika : Koda za prevrjanje admina

Ko pritisnemo na gumb za dodajanje izdelka se pokliče funkcija. V funkciji se najprej preveri, če se kategorija, ki smo jo izbrali ujema z eno od kategorij iz tabele kategorija. Če se ne ujema, nas vrže ven. Če pa drži, se pa izvede preprosta poizvedba za vstavljanje v tabelo izdelek s stavkom INSERT. Podatke za vnos dobi iz tabele $\_GET.

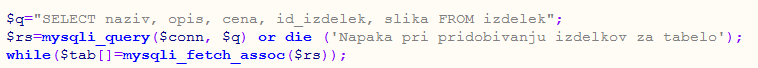


Slika : Poizvedba za vstavljane v tabelo izdelek

Ko sem zapolnil tabelo izdelek z dovolj izdelki, sem šel na izdelavo moje prve strani kupuj. Stran vsebuje vse izdelke in vsak je v svojem okvirju oz. div-u. Div je sestavljen iz imena na vrhu, pod tem se nahaja slika izdelka in pod tem opis ter cena. V desnem spodnjem kotu imamo povezavo ''V košarico'', ki nam izdelek, da v košarico. Funkcija za izpis teh izdelkov se začne s tem, da dobimo vse izdelke iz tabele. Ko sem klical funkcijo myslqi\_fetch\_assoc(), sem imel problem. Dobil sem tabelo vseh izdelkov, vendar mi je na koncu dodalo še en prazen element, ki je delal težave mojemu izpisu. Izpisal je še dodaten prazen div. Ta problem sem rešil tako, da sem preštel koliko izdelkov je v tabeli in bral podatke dokler se števec ni ujemal s število izdelkov v tabeli. Kasneje sem ta problem reševal s preprostim stavkom unsest($tab[count($tab)-1], vendar se takrat še nisem spoznal s tem in problem rešil na svoj način.

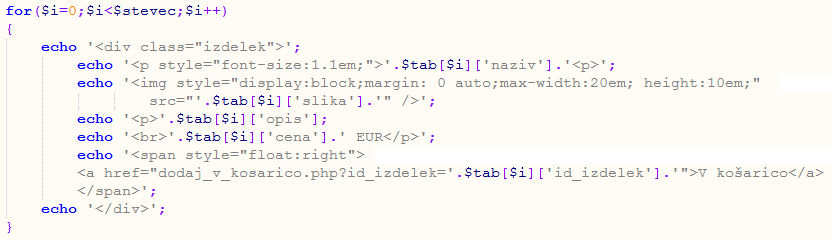


Slika : Štetje izdelkov



Slika : Pridobivanje vseh izdelkov

Nato sem prišel do še enega problema. Nisem vedel kako narediti, da ko kliknem v košarico, da vem kateri izdelek sem vbistvu zares izbral. Ta problem sem rešil tako, da sem v samo povezavo shranil identifikator tega izdelka. Ko pritisneš na povezavo, se prek URL-ja pošlje identifikator do ciljne strani, ki je bila v mojem primeru dodaj\_v\_košarico.php.



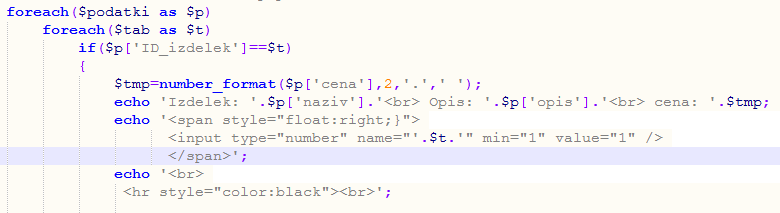
Slika : Izpis enega izdelka

V datoteki dodaj\_v\_košarico.php sem nato naredil novo spremenljivko oz. tabelo z imenom tabela\_kosarice, ki sem jo shranil v sejo. V seji sem jo potreboval, ker nisem smel zgubiti, kaj sem dal v košarico že prej. Edini cilj te datoteke je, da shrani v sejo, vse kar smo mi izbrali in nas nato vrne na prvo stran, da lahko nadaljujemo z našim kupovanjem.

Sedaj smo imeli v seji shranjeno katere izdelke je uporabnik izbral. V košarici je bilo potrebno te podatke prikazati. Če je košarica prazna nam je to tudi izpiše. Ker imamo v seji samo tabelo ključev, sem v funkciji za izpis košarice moral najprej pogledati kateri identifikator pripada kateremu izdelku. S poizvedbo sem dobil vse zapise v izdelku in jih primejral z mojo tabelo košarice. Naredil sem formo, ki se sklicuje na isto stran na kateri je. Ko sem našel, da se identifikatorja ujemata, sem v formi izpisal ime tega izdelka, opis ter ceno. Zraven sem še dodal, da stranka izbira število kosov. Da sem vedel za naprej kateremu izdelku pripada katero število kosov, sem vnosnim poljem pod atributom name oz. ime, shranil notri identifikator izdelka.

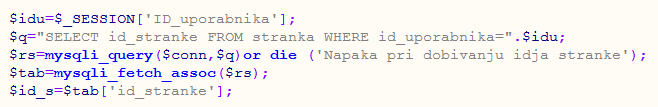


Slika : Pidobivanje košarice ter vseh izdelkov iz tabele



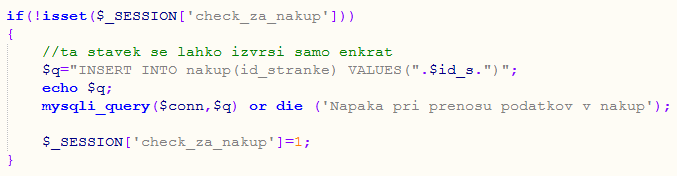
Slika : Izpis izdelkov, ki jih je stranka izbrala

Ko se stranka pritisne gumb kupi, se v $\_GET zapiše nova spremenljivka $\_GET['kupljeno']. V tej datoteki nato pogledam, če ta spremenljivka obsataja. Če obstaja pokličem funkcijo, ki shrani nov nakup v tabelo nakupi. Za vhodni parameter vzame število kosov in identifikator izdelkov, ki jih dobi iz tabele $\_GET. Ker imamo v nakupu stolpec id\_stranke, sem moral pridobiti ta identifikaor od nekod. To sem rešil s preprosto poizvedbo. Ker imam v seji shranjen identifikator uporabnika, sem lahko preko tega prišel do identifikatorja naše stranke.



Slika : Poizvedba za id stranke

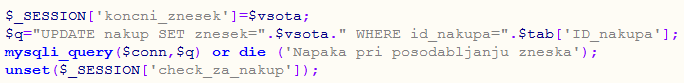
Ker imamo v tabeli nakup stolpec končni znesek, je bilo v to tabelo prvo vneseti samo identifikator stranke, šele kasneje posodobiti končni znesek. Končni znesek lahko dobimo šele, ko so vse postavke zapisane, vendar postavk nemoremo zapisati, če še nimamo nakupa. Nakup se je lahko zapisal samo enkrat, ker sem kupljene izdelke vnašal v funkcijo preko zanke, sem bil primoran dodati spremenljivko z imenom 'check\_za\_nakup' in jo postaviti na 1, ko se je nakup zgodil. Tako vsakič, ko je šel nov krog zanke se ni zapisal nov nakup ampak samo nova postavka.



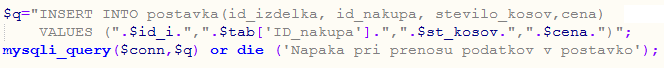
Slika : Dodajanje nakupa

Ko se zapiše nakup, sem moral nekako vedeti, kateri nakup iz tabele sedaj vzeti za vse postavke, ki so še na vrsti. Uporabil sem stolpec, ki vsebuje datum nakupa iz tabele nakup. Pogledal sem vse nakupe in jih razvrstil padajoče po času in prvi, ki je bil na seznamu je bil zadnji nakup oz. nakup, ki ga potrebujem za naprej. Iz tega nakupa sem nato vzel identifikator in ga dal v poizvedbo za postavko.

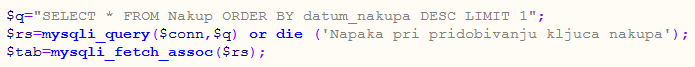
Tukaj se zanka konča in zapisane imamo vse postavke, ki so pod istim nakupom. Bilo je le še potrebno dodati končno ceno v nakup. To je nova funkcija v kateri prvo pogledamo ponovno kateri je zadnji nakup. Vzamemo identifikator tega nakupa in gremo čez vse postavke in najdemo tiste, ki imajo za identifikator nakupa enak identifikatorju, ki smo ga pridobili. Nato pogledamo ceno vsake postavke in te cene seštejemo. Skupni znesek sem shranil v sejo, ker sem vedel, da ga bom potrebaval za naprej. Nato še posodobimo vrednost stolpca koncni\_znesek za zadnji nakup in zbrišemo spremenljivko $\_SESSION['check\_za\_nakup'], saj sedaj lahko pride do novih nakupov.



Slika : Zadnji koraki zapisa nakupa



Slika : Zapis v postavko



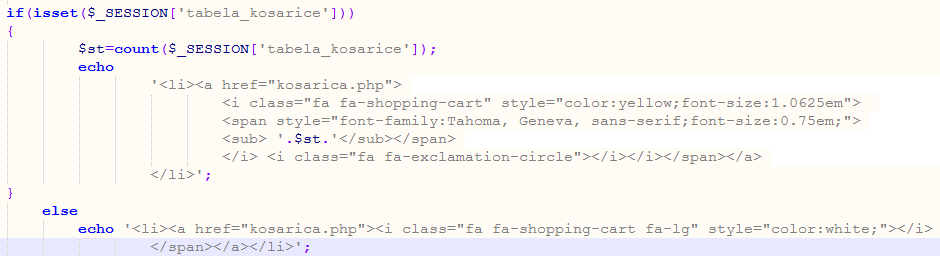
Slika : Poizvedba za zadnji nakup

Ko imamo dokončan zapis za nakup in postavke nas pošlje na stran placilo.php. Na tej strani simuliramo dejansko plačilo. Prikaže del naših podatkov, nato lahko vpišemo naslov prejemanja ter številko trrja. Te podatki se ne zapišejo nikamor. Prikaže nam tudi končni znesek, ki ga pripeljemo iz seje. Ko pritisnemo na gumb nam stran javi, da je bil nakup uspešen. Spodaj imamo nato povezavo, ki nas vrne na prvo stran.

Nato sem se odločil, da za košarico dodam nekaj, da se bo opazilo, da je v njej več elementov. Odločil sem se, da bom to spremenil kar v meniju. Ko je košarica prazna je znak za košarico bel, ko pa dodaš nek nov izdelek se pa spremeni barva košarice, doda se število izdelkov ter doda klicaj kot neka opomba.



Slika : Pred in po dodanem izdelku



Slika : Koda za košarico v meniju

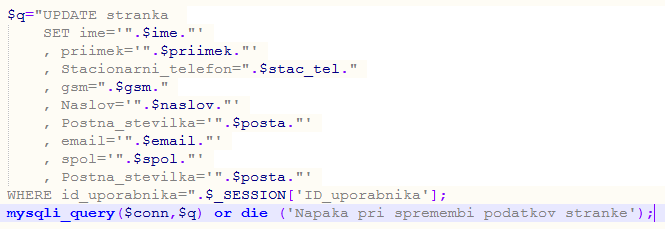
Kasneje sem dodal še stavek, da nemore kdo vstopiti na katero koli stran brez, da bi bil prijavljen. Ta stavek sem dodal na vsaki strani.



Slika : Stavek za preprečitev dostopa brez prijave

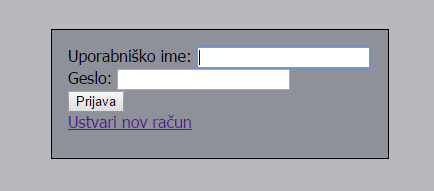
Zadnja stran, ki sem jo pustil za na konec je stran podatki.php. Sestavljena je iz dveh delov. Prvi del nam izpiše vse naše podatke drugi del pa predstavlja zgodovino našega nakupovanja. Za zgodovino nakupa potrebujemo prvo identifikator stranke, ki stran trenutno uporablja. To izvemo ponovno s pomočjo identifikaroja uporabnika.Naslednji korak je, da najdemo vse nakupe in postavke, ki jih je ta stranka izvedla. Za tem potrebujemo še tabelo vseh izdelkov, da bomo lahko izpisali ime izdelkov, saj imamo v tabeli postavka shranjene le identifikatorje izdelkov. Ko smo zbrali vse te podatke, gremo čez vsak nakup in za vsak napis naredi nov okvir oz. div. Ta vsebuje identifikator nakupa, datum nakupa, imena izdelkov, ki smo jih kupili, število kosov, ki smo jih kupili ter, cena izdelka ter na koncu se končni znesek računa. Če nismo opravili še nobenega nakupa, nam to tudi izpiše. Zraven strankinih podatkov je povezava, ki te pošlje, na stran na kateri lahko spremeniš svoje podatke. Stran je skoraj enaka strani za novega uporabnika. Razlika je v klicu funkcije in tukaj ni vnosnega polja za geslo in uporabniško ime, tega se ne da spremeniti.

# Navodila za uporabnike



Slika : Funkcija za spremembo podatkov

Kar se tiče uporabe moje spletne trgovine, sem se potrudil stran narediti čim bolj preprosto. Stran je narejena, da v njo brez prijave nemoremo vstopiti. Ko odpremo stran, se nam torej najprej pojavi prijavna forma.



Slika : Prijavno okno

Če še nimamo svojega računa, naredimo to s spodnjo povezavo. Povezava nas odpelje na stran, ki vsebuje formo. V formo vnesemo vse potrebne podatke in pritsnemo ustvari račun. Ko to naredimo, nas stran vrne nazaj na prijavno formo, v katero vpišemo uporabniško ime in geslo, ki smo ga pravkar ustvarili. Če smo se poskušali vpisati z napačnimi podatki, nam to stran sporoči. Če je uporabniško ime in geslo pravilno, nas preusemeri na prvo stran. Prva stran predstavlja v meniju zavihek kupuj. Na tej strani najdemo vse izdelke, ki so nam na voljo. Vsak izdelek je prikazan v svojem okviru. Vsak okvir vsebuje ime izdelka, pod tem slika izdelka ter pod tem opis in cena izdelka.



Slika : Okvir izdelka

V spodnjem desnem kotu najdemo povezavo 'V košarico'. S tem, ko pritisnemo na to povezavo, nam ta izdelek shrani v košarico. Da se je izdelek shranil, vidimo to tudi iz spremembe izgleda menija.



Slika : Sprememba ikone po dodajanju

Ko smo se odločili, da smo konec z nakupom, pritisnemo v meniju na ikono s košarico. Ta nas preusmeri na novo stran. Na strani so vidni izdelki, ki smo jih izbrali, dodano imajo še, da izberemo število kosov, ki jih želimo. Ko smo se odločili, pritisnemo spodaj gumb pojdi na plačilo. To nas preusmeri na novo stran. Na tej strani dokončamo plačilo. Vnesmo naslov prejemanja ter številko TRR računa. Nato pritisnemo gumb potrdi plačilo. Stran nam s pojavnim oknom sporoči, da je bil nakup uspešen. Nato pritisnemo spodnjo povezavo in vrnemo se na prvo stran.

Če pritisnemo na zavihek o nas, ti izpiše podatke te spletne trgovine. Napisano je nekaj par besed o podjetju, ter kontakti podjetja.

V zavihku podatki najdemo na začetku vse naše podatke, ki smo jih vnesli, ko smo se naredili svoj račun. V spodnjem desnem gumbu imamo povezavo, ki nas preusmeri na stran za spreminjanje podatkov. Če smo se pri kakšnem zapisu zmotili, ali pa je prišlo do sprememb, se to na tej povezavi ureja. Ko tam uredimo vse podatke na novo, pritisnemo gumb posodobi račun. Na zavihku podatki se nahaja še zgodovina vseh nakupov, ki smo jih opravili. Nakupi so razdeljeni vsak v svojem okvirju in vsebujejo številko računa, datum nakupa, vse izdelke z njihovo kolčino ter ceno in na koncu končni znesek.

Desno zgoraj v meniju imamo še gumb za odjavo. Ko pritisnemo na ta gumb, nas izpiše iz sistema in vrne na prijavno stran.

# Zaključek

S svojim izdelkom sem zelo izboljšal svoje znanje v vseh treh programskih jezikih, ki sem jih uporabljal. Spoznal sem, da izdelava spletne trgovine ni tako lahka. Veliko kode je potrebno napisati ter zelo dobro v naprej razmišljati. Dobro je treba razumeti povezave med vsemi tabelami. Med samo izdelavo strani sem le dvakrat prišel do večjega zapleta. Prvi zaplet se je zgodil, ko sem dodajal uporabnika in me je zanimalo katere uporabnike že imamo v tabeli, da se uporabniška imena ne morejo podvojiti. Ko sem prebral vse stranke, je oblika v kateri mi je shranilo bilo zelo neorganizirana. Vrednosti, ki so se shranile so bile pravilne, a so vse dobile zelo veliko število za indentifiktor, ki ni bilo povezanom z ničemer. Ta problem mi je vzel zelo veliko časa, a sem nato celo funkcijo napisal od začetka in vse je bilo dobro. Drugi problem se je pojavil pri spreminjanju izgleda ikone košarice. Ikona košarice se ni hotela spreminjati, ko je bilo v košarici nekaj izdelkov. Problem je bil sestavljen iz več manjših napak, ki sem jih kasneje odpravil. Stran bi se dalo še na veliko načinov izboljšati. Lahko bi popravil izgled strani, saj smisla za oblikovanje nimam kaj preveč. Bilo bi pametno dodati, da admin lahko spreminja vrednosti izdelkov. Lahko bi še omejil kaj vse uporabnik lahko vnese v vnosna polja. Tega sem se lotil, a sem zaradi neznane napake javascripta to pustil in se posvetil drugim stvarem. Atributa kategorija nisem posebi uporabljal. Lahko bi se ga dalo implementirati na prvi strani. Imeli bi nek dodaten stranski izbirni meni, ki bi vseboval vse kategorije. Tam bi lahko odkljukali izdelke katerih kategorij hočemo imeti. V tem meniju bi lako tudi določili ceno, od kod do kod lahko sega cena izdelkov. Mislim, da sem svoj izdelek zelo dobro naredil in bi ga želel tudi dokončati. Problem je, da se kakšnih idej žal prepozno spomniš.

# Viri

* <https://sl.wikipedia.org/wiki/Spletni_stre%C5%BEnik_Apache> (11.4.2017)
* <https://en.wikipedia.org/wiki/XAMPP> (11.4.2017)
* <https://sl.wikipedia.org/wiki/MySQL> (11.4.2017)
* <https://sl.wikipedia.org/wiki/SQL> (11.4.2017)
* <https://sl.wikipedia.org/wiki/PHP> (11.4.2017)

# Izjava o avtorstvu

Izjavljam, da je strokovno poročilo Spletna trgovina v celoti moje avtorsko delo, ki sem ga izdelal/-a samostojno s pomočjo navedene literature in pod vodstvom mentorja/-ice.

Ljubljana, 10.4.2017 Luka Rosić