Izrada blagajne za Raspbery PI2 platformu

Barić, Matej

Undergraduate thesis / Završni rad

2016

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:814094

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-20**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
ELEKTROTEHNIČKI FAKULTET

Stručni studij

IZRADA BLAGAJNE ZA RASPBERRY PI 2 PLATFORMU

Završni rad

Matej Barić

Osijek, 2016.

Sveučilište Josipa Jurja Strossmayera u Osijeku

ETFOS ELEKTROTEHNIČKI FAKULTET OSIJEK

Obrazac Z1S: Obrazac za imenovanje Povjerenstva za obranu završnog rada na stručnom studiju

Osijek,2016.

Odboru za završne i diplomske ispite

Imenovanje Povjerenstva za obranu završnog rada na stručnom studiju

Ime i prezime studenta:	Matej Barić
Studij, smjer:	Stručni studij Informatike
Mat. br. studenta, godina upisa:	AI4295, 2013./2014.
Mentor:	Doc. dr. sc. Krešimir Nenadić
Sumentor:	
Predsjednik Povjerenstva:	
Član Povjerenstva:	
Naslov završnog rada:	Izrada blagajne za Raspberry Pi 2 platformu
Primarna znanstvena grana rada:	**Informatika**
Sekundarna znanstvena grana (ili polje) rada:	
Zadatak završnog rada	Kratko objasniti mogućnosti Raspberry Pi2 platforme kao i njene specifikacije. Postaviti web poslužitelj na platformu i objasniti način kako se to radi. Dizajnirati bazu podataka koja će se koristiti za blagajnu. Izraditi web aplikaciju s korisničkim sučeljem i funkcionalnostima za pristup bazi podataka. Opisati mogućnosti izrađenog rješenja te testirati rad.
Prijedlog ocjene pismenog dijela ispita (završnog rada):	
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: Postignuti rezultati u odnosu na složenost zadatka: Jasnoća pismenog izražavanja: Razina samostalnosti:

Potpis sumentora:	Potpis mentora:

Dostaviti:

1. Studentska služba

U Osijeku, 2016. godine

Potpis predsjednika Odbora:

ETFOS
ELEKTROTEHNIČKI FAKULTET OSIJEK
Sveučilište Josipa Jurja Strossmayera u Osijeku

IZJAVA O ORIGINALNOSTI RADA

Osijek, 2016

Ime i prezime studenta:	Matej Barić
Studij :	Stručni studij Informatike
Mat. br. studenta, godina upisa:	AI4295, 2013./2014.

Ovom izjavom izjavljujem da je rad pod nazivom**: Izrada blagajne za Raspberry Pi 2 platformu**

izrađen pod vodstvom mentora

Doc. dr. sc. Krešimir Nenadić

i sumentora

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

SADRŽAJ

1. UVOD ... **1**

2. RASPBERRY PI 2 I NJEGOVO POSTAVLJANJE ... **2**

2.1. RASPBIAN ... 3
2.2. NGINX WEB POSLUŽITELJ ... 3
2.3. PHP (HYPERTEXT PREPROCESSOR) ... 3
2.4. MYSQL ... 4
2.5. PHPMYADMIN ... 4
2.6. INSTALACIJA RASPBIAN OPERACIJSKOG SUSTAVA ... 5
2.7. POSTAVLJANJE RASPBIAN OPERACIJSKOG SUSTAVA ... 6
2.8. INSTALACIJA PROGRAMSKE PODRŠKE ZA DALJNJI RAD ... 10
2.8.1. Konfiguracija NGINX web poslužitelja ... *16*
2.8.2. Konfiguracija phpMyAdmina ... *18*

3. STRUKTURA I PROGRAMSKI KOD BLAGAJNE ... **19**

3.1.1. HTML ... *19*
3.1.2. CSS ... *19*
3.1.3. PHP ... *19*
3.1.4. SQL ... *19*
3.2. STRUKTURA DATOTEKA BLAGAJNE ... 20
3.3. PROGRAMSKI KOD BLAGAJNE ... 21
3.3.1. otvorivezu.php ... *21*
3.3.2. zaglavlje.php ... *22*
3.3.3. podnozje.php ... *22*
3.3.4. provjera.php ... *23*
3.3.5. login.php ... *23*
3.3.6. index.php ... *25*
3.3.7. blagajna.php ... *25*
3.3.8. proizvodi.php ... *30*
3.3.9. ***sviproizvodi.php*** ... *31*
3.3.10. stanje.php ... *32*
3.3.11. dodajproizvod.php ... *33*
3.3.12. logout.php ... *33*
3.3.13. izbrisi.php ... *34*
3.3.14. uredi.php ... *34*

4. IZGLED I FUNKCIONALNOST BLAGAJNE ... **36**

4.1. IZGLED BAZE PODATAKA ... 39

5. ZAKLJUČAK ... **41**

LITERATURA ... **42**

SAŽETAK ... **43**

ABSTRACT ... **44**

ŽIVOTOPIS ... **45**

1. UVOD

Zadatak ovog završnog rada je izrada blagajne za Raspberry Pi 2 platformu. Opis zadatka je: Kratko objasniti mogućnosti Raspberry Pi 2 platforme kao i njene specifikacije. Postaviti web poslužitelj na platformu i objasniti način kako se to radi. Dizajnirati bazu podataka koja će se koristiti za blagajnu. Izraditi web aplikaciju s korisničkim sučeljem i funkcionalnostima za pristup bazi podataka. Opisati mogućnosti izrađenog rješenja te testirati rad.

Blagajna će biti izrađena uz pomoć jezika koji spadaju u skupinu web tehnologija. Jezici koji se koriste su HTML, CSS, PHP i SQL. Najvećim dijelom bit će napisana u PHP-u. Uz PHP koristit će se HTML i CSS kako bi se napravio dizajn i dobilo željeno oblikovanje. SQL će se koristiti za komunikaciju s poslužiteljem, dohvaćanje, odnosno zapisivanje podataka u bazu podataka. Raspberry Pi 2 platforma izabrana je zbog njene jednostavnosti, dostupnosti te izvedbe same platforme. Blagajna je napravljena tako da bude što jednostavnija za korištenje krajnjem korisniku, pouzdana i uvijek dostupna. Velika prednost blagajne na Raspberry Pi 2 platformi je što će se uređaj koristiti i kao poslužitelj i kao klijent. Korisnik će biti u mogućnosti pristupiti blagajni sa samog Raspberry Pi 2 uređaja.

Projektni zadatak:

Kratko objasniti mogućnosti Raspberry Pi 2 platforme kao i njene specifikacije. Postaviti web poslužitelj na platformu i objasniti način kako se to radi. Dizajnirati bazu podataka koja će se koristiti za blagajnu. Izraditi web aplikaciju s korisničkim sučeljem i funkcionalnostima za pristup bazi podataka. Opisati mogućnosti izrađenog rješenja te testirati rad.

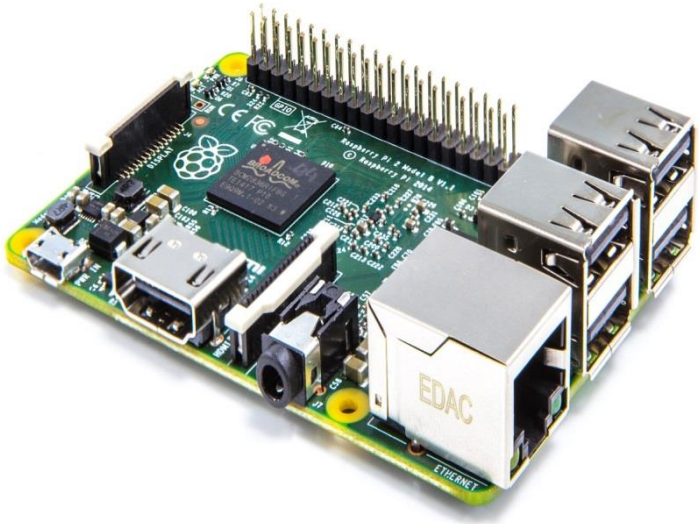
2. RASPBERRY PI 2 I NJEGOVO POSTAVLJANJE

Raspberry Pi je računalo veličine kreditne kartice koje se uključuje u TV ili monitor te koristi tipkovnicu i miš. Može činiti sve što i jedno stolno računalo, od pretraživanja Interneta, procesiranja visoko razlučivog video sadržaja, uređivanja teksta do pokretanja igrica. Štoviše, postoji i mogućnost povezivanja s ostalim uređajima. Raspberry Pi 2 Model B je računalo druge generacije Raspberry Pia. Specifikacije Raspberry Pi 2 Model B računala su:

- 900 MHz četverojezgreni ARM Cortex-A7 procesor,
- 1 GB RAM,
- 4 USB izlaza,
- HDMI izlaz,
- Ethernet izlaz,
- Postolje za MicroSD karticu,
- VideoCore IV 3D grafička jezgra.

Druga generacija uspješno pokreće Android, Linux, a za razliku od prijašnjih, pokreće i Windows 10 operacijski sustav. Za potrebe završnog rada koristit će *Raspbian* distribuciju Linux operacijskog sustava.

U sljedećim potpoglavljima ukratko će se objasniti *Raspbian* operacijski sustav te programska podrška koja će se koristiti u radu.



Sl. 2.1. *Raspberry Pi 2 Model B*

2.1. Raspbian

Raspbian je temeljen na *Debian* distribuciji Linux operacijskog sustava. Ciljano je dizajniran za sklopovlje Raspberry Pia te se sastoji od osnovnih programa i alata koji su potrebni za rad. Napravila ga je mala grupa entuzijasta, a najzaslužniji su Mike Thompson i Peter Green koji su prvu inačicu izdali u lipnju 2012. godine. *Raspbian* je još uvijek u aktivnom razvoju kako bi se poboljšala stabilnost i performanse samog sustava.

2.2. NGINX web poslužitelj

NGINX je besplatan; HTTP poslužitelj otvorenog koda, visokih performansi, kao i IMAP/POP3 poslužitelj. Razvoj je počeo Igor Syseov 2002. godine, a osnovao ga je Syseov u lipnju 2012. godine kako bi izdao proizvod i pružio programsku podršku za isti. Moguće ga je pokrenuti na Linux, Unix, BSD, OS X, Solaris i Windows operacijskim sustavima. Poznat je po stabilnosti, visokim performansama, jednostavnoj konfiguraciji te niskoj potrošnji resursa. Za razliku od ostalih poslužitelja, ne oslanja se na procesorske niti kako bi obradio zahtjeve. Koristi unaprijed određene količine memorije. Kao poslužitelj koriste ga mnoge poznate stranice kao što su Netflix, CloudFlare, WordPress, GitHub te mnoge druge.

2.3. PHP (Hypertext Preprocessor)

PHP je široko korišten skriptni programski jezik namijenjen programiranju web stranica. Distribuiran je kao skriptni programski jezik otvorenog koda. Ističe se širokom podrškom raznih baza podataka i Internet protokola te mogućnosti komponiranja u HTML. Prvu inačicu PHP-a razvio je Rasmus Lerdorf 1995. godine za brojanje posjeta na svojoj privatnoj stranici. Moguće ga je koristiti s gotovo svim web poslužiteljima i operacijskim sustavima. PHP je danas jedan od najzastupljenijih skriptnih programskih jezika za programiranje web aplikacija.

2.4. MySQL

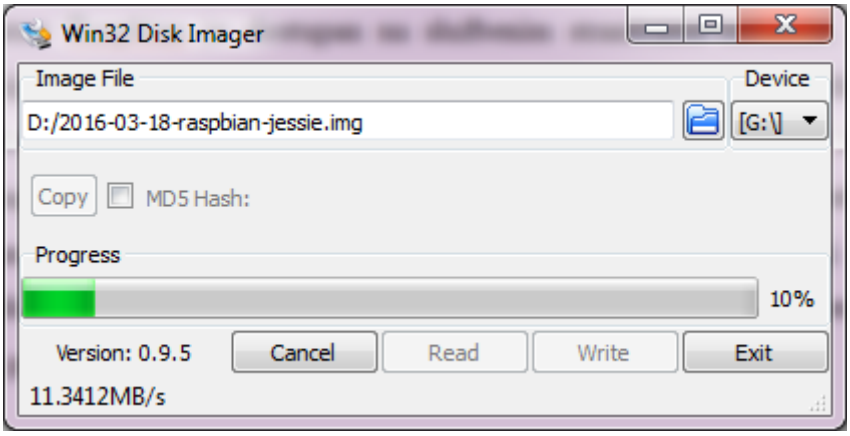
MySQL je besplatan; poslužitelj otvorenog koda za upravljanje bazom podataka. Napisan je u C i C++ programskom jeziku te ga je moguće koristiti na više operacijskih sustava, a često dolazi i kao sastavni dio nekih Linux distribucija. Osnovali su ga David Axmark, Allan Larsson i Michael Widenius. Prva inačica se pojavila u svibnju 1995. godine. *MySQL* baze su relacijskog tipa koji se pokazao kao najbolji način skladištenja i pretraživanja velikog broja podataka. Popularan je izbor za baze podataka koje koriste web aplikacije.

2.5. phpMyAdmin

phpMyAdmin je besplatna aplikacija napisana u PHP-u, namijenjena administriranju MySQL-a putem weba. Nudi široki niz mogućnosti koje se mogu napraviti putem korisničkog sučelja, ali isto tako nudi mogućnost direktnog prosljeđivanja SQL naredbi. Tobias Ratschiller je 1998. godine počeo raditi na web aplikaciji za SQL pisanoj u PHP-u. Zbog nedostatka vremena napustio je projekt. Grupa programera 2001. godine je preuzela projekt i nastavila s razvojem. Danas je *phpMyAdmin* jedna od najpopularnijih web aplikacija za administraciju *MySQL* poslužitelja.

2.6. Instalacija Raspbian operacijskog sustava

Kako bi se uopće započelo s radom na Raspberry Piu, prvo je potrebno instalirati operacijski sustav. Operacijski sustav *Raspbian* je dostupan na službenim stranicama računala. Nakon što je preuzet operacijski sustav, potrebno ga je prenijeti na MicroSD karticu veličine minimalno 4 GB. Program koji se koristi kako bi se prenio operacijski sustav na MicroSD karticu naziva se *Win32DiskImager*. Nakon pokretanja *Win32DiskImagera*, odabire se datoteka operacijskog sustava koja je preuzeta ranije te slovo kartice koje je postojeći operacijski sustav dodijelio istoj. Nakon odabira, pritiskom na prozorčić „*Write*“ počinje proces instalacije operacijskog sustava kako se vidi na priloženoj slici (Sl. 2.2.).

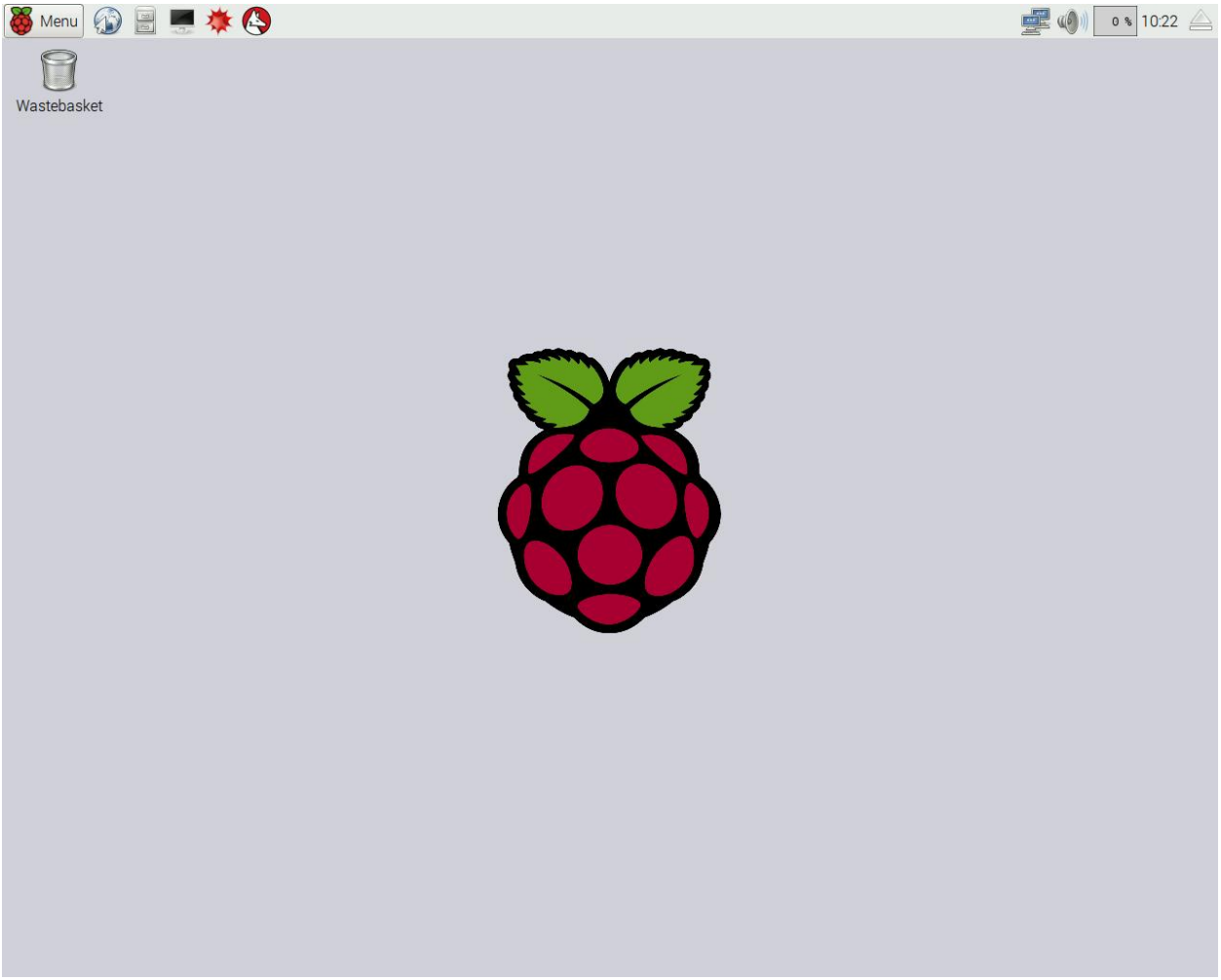


Sl. 2.2. *Proces instalacije Raspbian operacijskog sustava*

Nakon završetka instalacije preostaje još samo umetnuti MicroSD karticu u postolje Raspberry Pia.

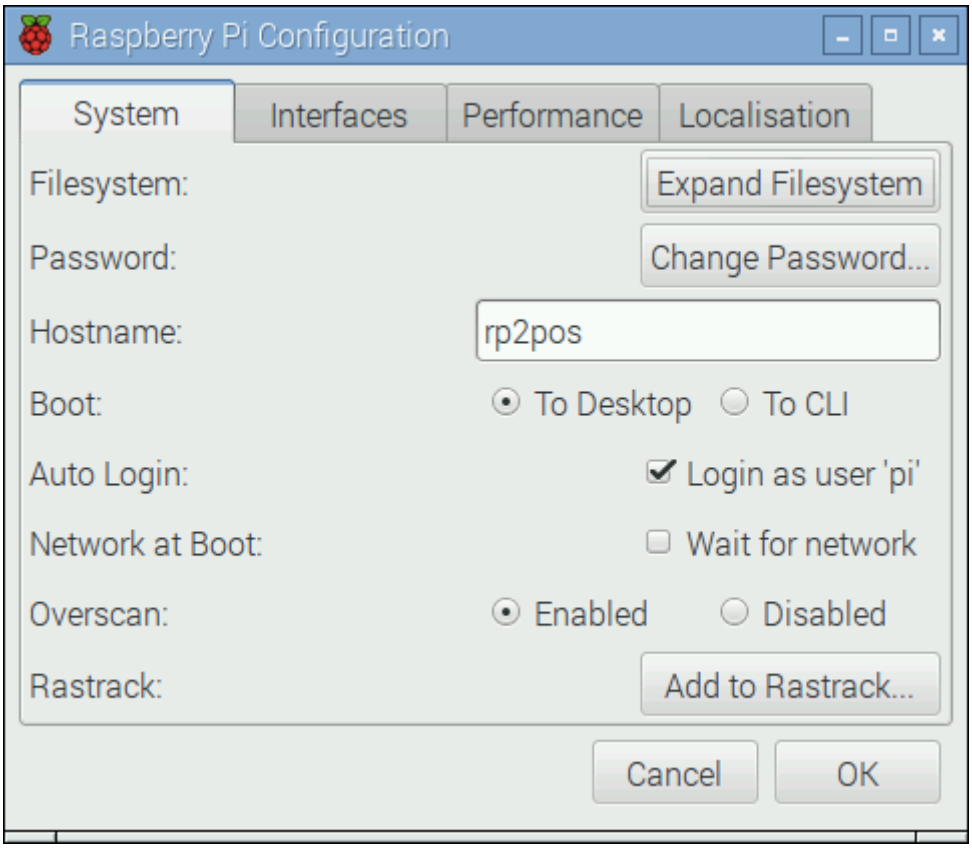
2.7. Postavljanje Raspbian operacijskog sustava

Nakon što je MicroSD kartica umetnuta u Raspberry Pi te uključen u izvor napajanja, pojavio se početni zaslon, odnosno radna površina.



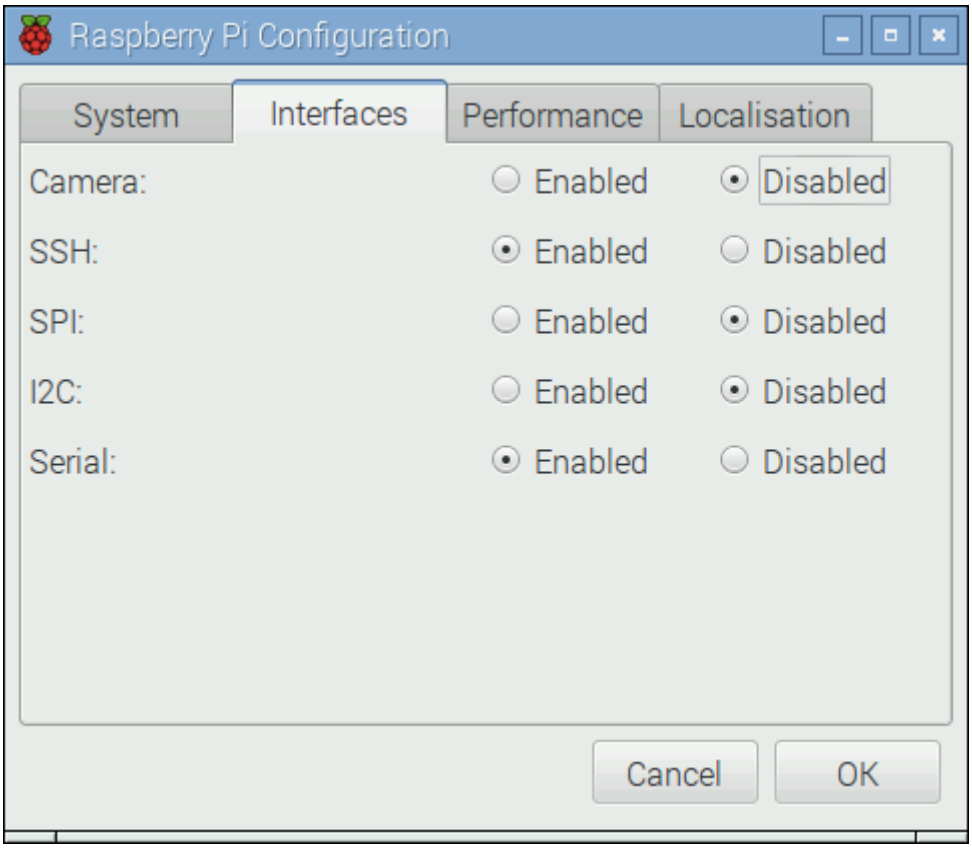
Sl. 2.3. *Radna površina Raspbian operacijskog sustava*

Kako bi se promijenile postavke, kako operacijskog sustava tako i samog uređaja, koristi se već ugrađeni program *Raspberry Pi Configuration*. Kako bi sustav normalno funkcionirao, potrebno je proširiti MicroSD karticu na njezinu maksimalnu veličinu. To će se učiniti klikom na prozorčić „*Expand Filesystem*“ koje je vidljivo na slici (Sl. 2.4). Mora se promijeniti zaporka jer korisnik „pi“ ima neograničena administratorska prava te promijeniti naziv računala kako bi se lakše našao na mreži. Kao naziv računala koristi se „rp2pos“.



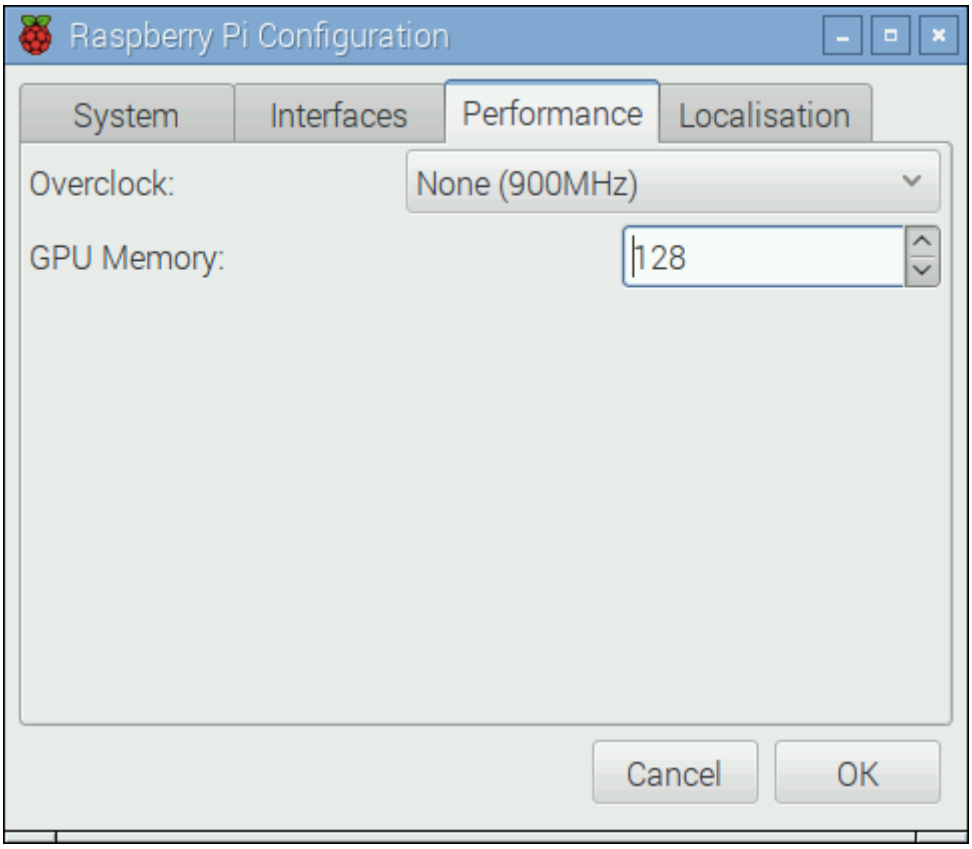
Sl. 2.4. *Raspberry Pi Configuration sučelje (System kartica)*

Kartica „*Interfaces*“ odnosi se na sučelja koja su se u mogućnosti uključiti, odnosno isključiti. Najvažnija stavka na ovoj kartici je SSH (engl. *Secure Shell*). SSH je mrežni protokol koji korisnicima omogućuje uspostavu sigurnog komunikacijskog kanala između računala. SSH je najvažnija stavka koja omogućuje udaljeni pristup računalu.



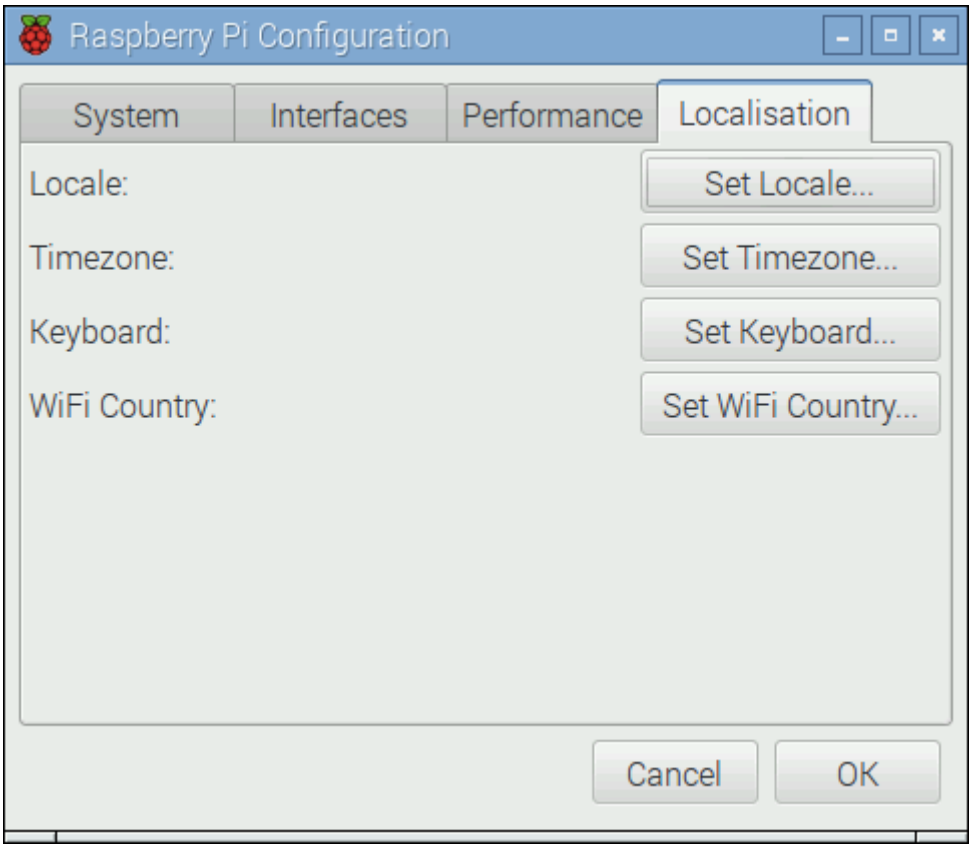
Sl. 2.5. *Raspberry Pi Configuration sučelje (Interfaces kartica)*

Kartica „*Preformance*“ odnosi se na performanse samog uređaja. Ovdje će se povećati memorija grafičkog čipa s 64 MB na 128 MB zbog boljih performansi Internet preglednika i prikaza sadržaja.



Sl. 2.6. *Raspberry Pi Configuration sučelje (Preformance kartica)*

Na kraju ostaje još jedna kartica pod nazivom „*Localisation*“. Ovdje se podešavaju parametri kao što su jezik, vremenska zona, raspored tipkovnice te WiFi regija.



Sl. 2.7. *Raspberry Pi Configuration sučelje (Localisation kartica)*

Nakon spremanja postavki potrebno je ponovno pokrenuti računalo.

2.8. Instalacija programske podrške za daljnji rad

Kao što je spomenuto u uvodu, da bi blagajna funkcionirala, potrebna je određena programska podrška. Kako bi se uopće pokrenula blagajna, moraju se instalirati određeni programi. Za web poslužitelj bit će korišten *NGINX* web poslužitelj koji je dostupan za Linux platforme, PHP module kako bi web poslužitelj pročitao PHP, *MySQL* poslužitelj i *MySQL* klijent za bazu podataka te *phpMyAdmin* kao sučelje za upravljanje bazom podataka. Sada će biti objašnjena instalacija svakog pojedinog programa za Raspberry Pi 2 platformu, a u daljnjim potpoglavljima ukratko će biti objašnjen svaki od programa i njegova svrha.

Kako bih se uopće počelo s instalacijom programa, prvo se moraju dohvatiti najnovije liste i ažuriranja za operacijski sustav. To će se učiniti pomoću sljedećih naredbi koje se upisuju u Terminal:

```
sudo apt-get update && sudo apt-get upgrade
```

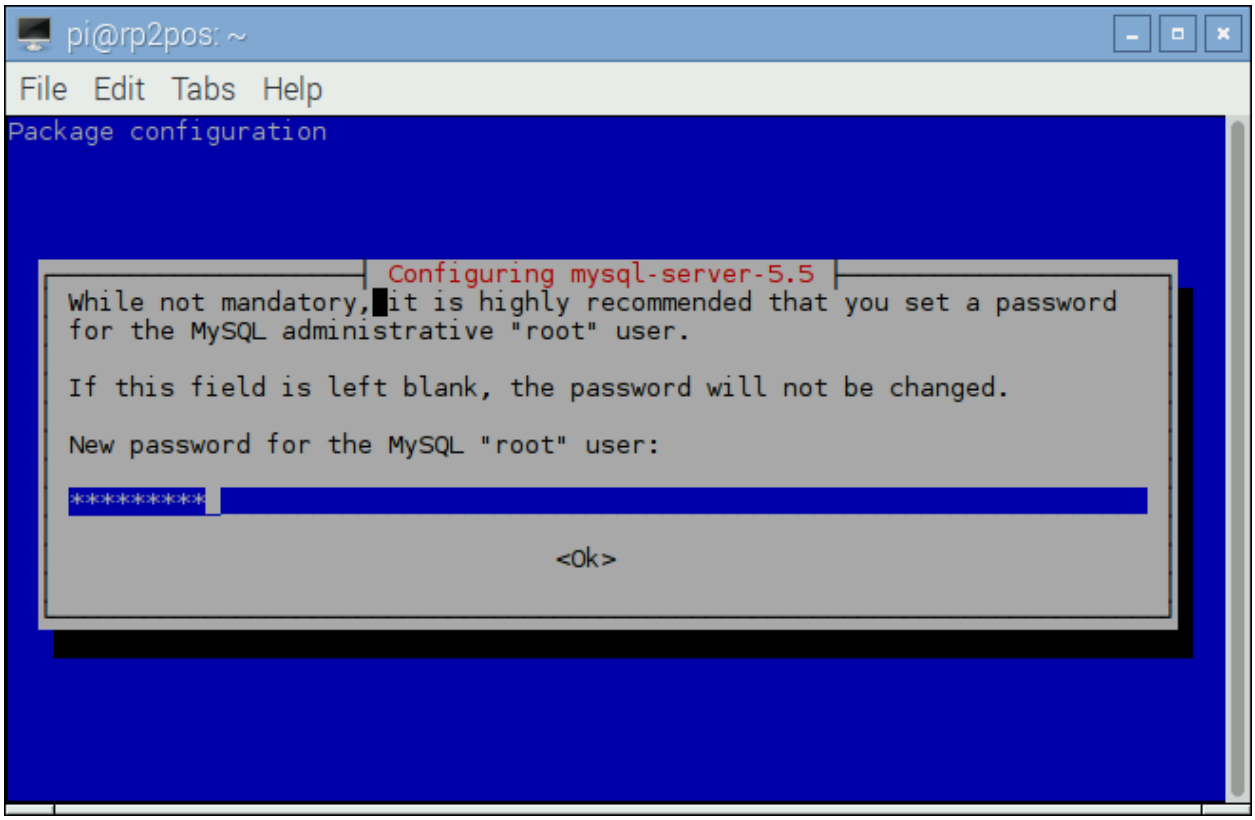
Nakon što je operacijski sustav ažuriran i dohvaćene nove liste programa, kreće se s instalacijom programa potrebnih kako bi blagajna funkcionirala. Kako bi se instalirao *NGINX* web poslužitelj i PHP moduli u *Terminal*, upisuju se sljedeće naredbe:

```
sudo apt-get install nginx php5 php5-fpm php5-mysql
```

Nakon instalacije *NGINX* web servera i PHP modula, dolazi na red *MySQL* poslužitelj i *MySQL* klijent za bazu podataka koja se koristi za blagajnu. Naredba koju se upisuje u *Terminal* je:

```
sudo apt-get install mysql-server mysql-client
```

Tijekom instalacije MySQL poslužitelja potrebno je unijeti tražene podatke. Prvi korak je unijeti novu lozinku za administratorskog korisnika „*root*“ te na sljedećem koraku ponovno unijeti istu lozinku iz sigurnosnih razloga.

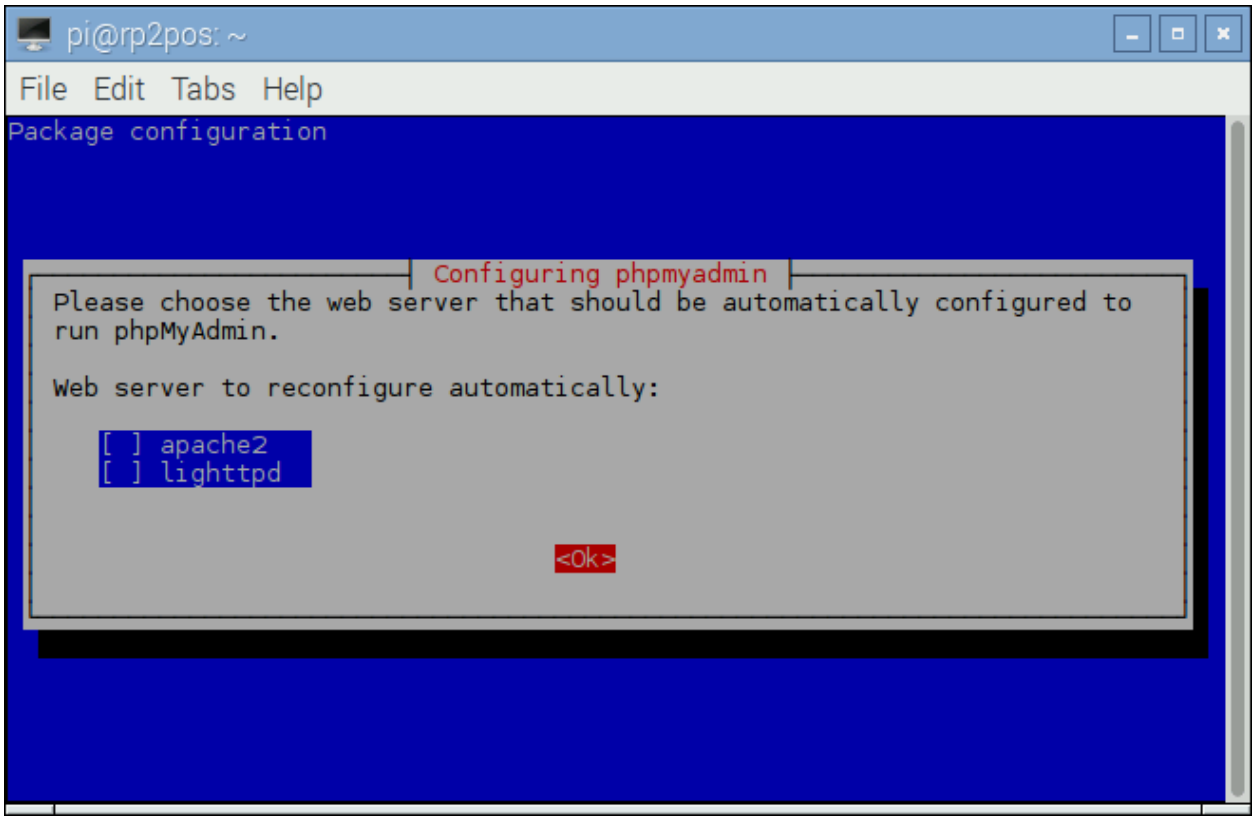


Sl. 2.8. *Zaporka administrativnog korisnika „root“ za MySQL poslužitelj*

Nakon što je proces instalacije *MySQL* poslužitelja završio, preostalo je još instalirati *phpMyAdmin*, a njega se instalira unošenjem naredbe:

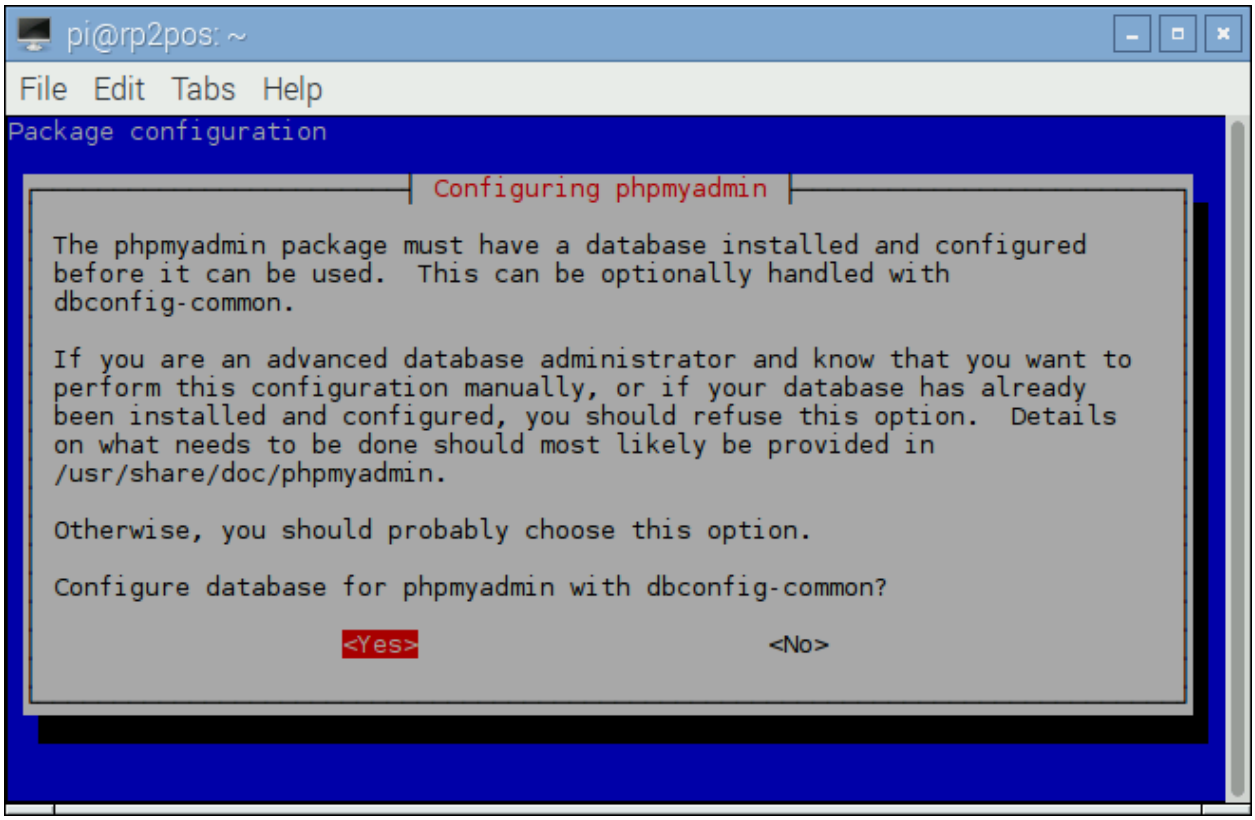
```
sudo apt-get install phpmyadmin
```

Instalacija *phpMyAdmina* s *NGINX* web poslužiteljem je nešto kompliciranija nego s ostalim web poslužiteljima. Prvi korak pri instalaciji je izbjegavanje automatske konfiguracije *phpMyAdmina* s poslužiteljem. Ovaj korak se preskače tako da se ukloni oznaka za preporučene poslužitelje i potvrdi oznakom „*Ok*“ kako je vidljivo na slici (Sl.2.9.).



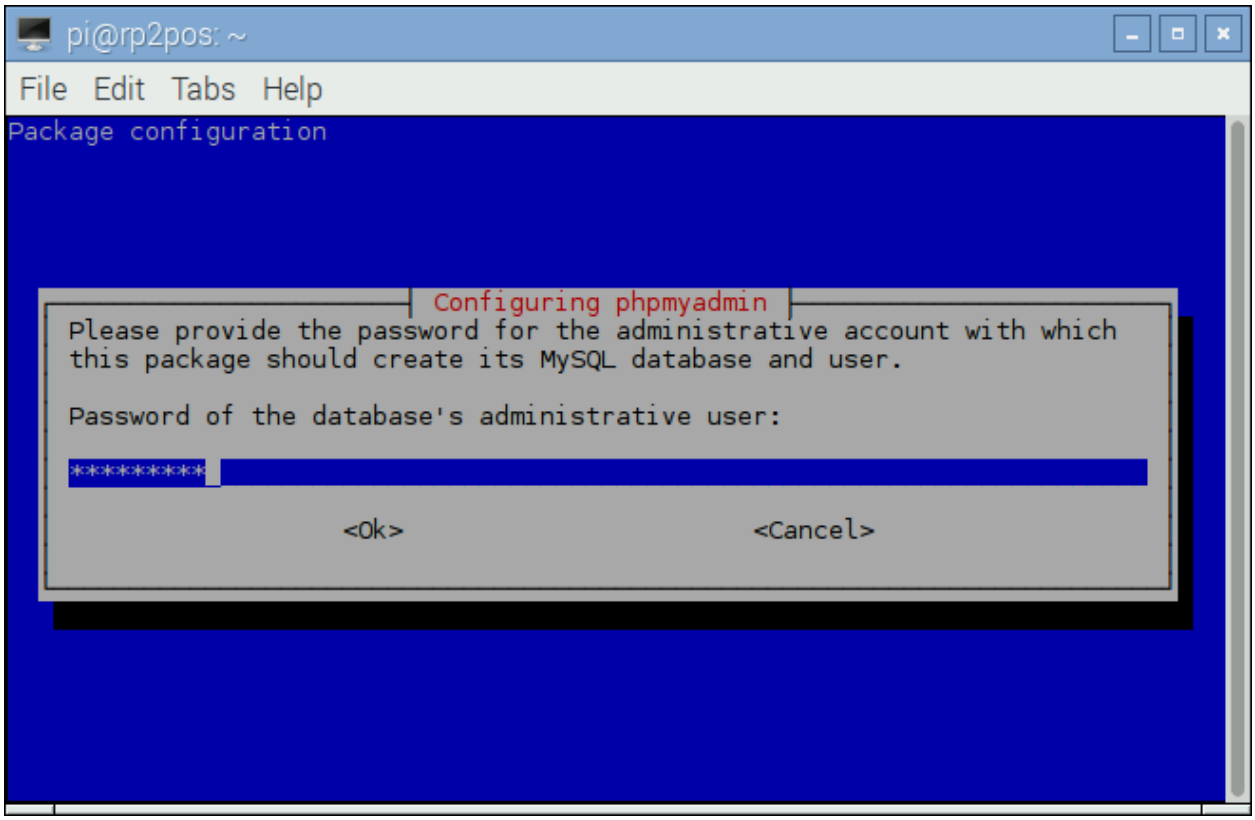
Sl. 2.9. *Preskakanje prvog koraka pri instalaciji phpMyAdmina*

Drugi korak je konfiguracija baze za *phpMyAdmin* sučelje. Na pitanje, želi li se konfigurirati baza za *phpMyAdmin* s *dbconfig-config*, odgovara se pritiskom oznake „*Yes*“ kao što je vidljivo na slici (Sl.2.10.).



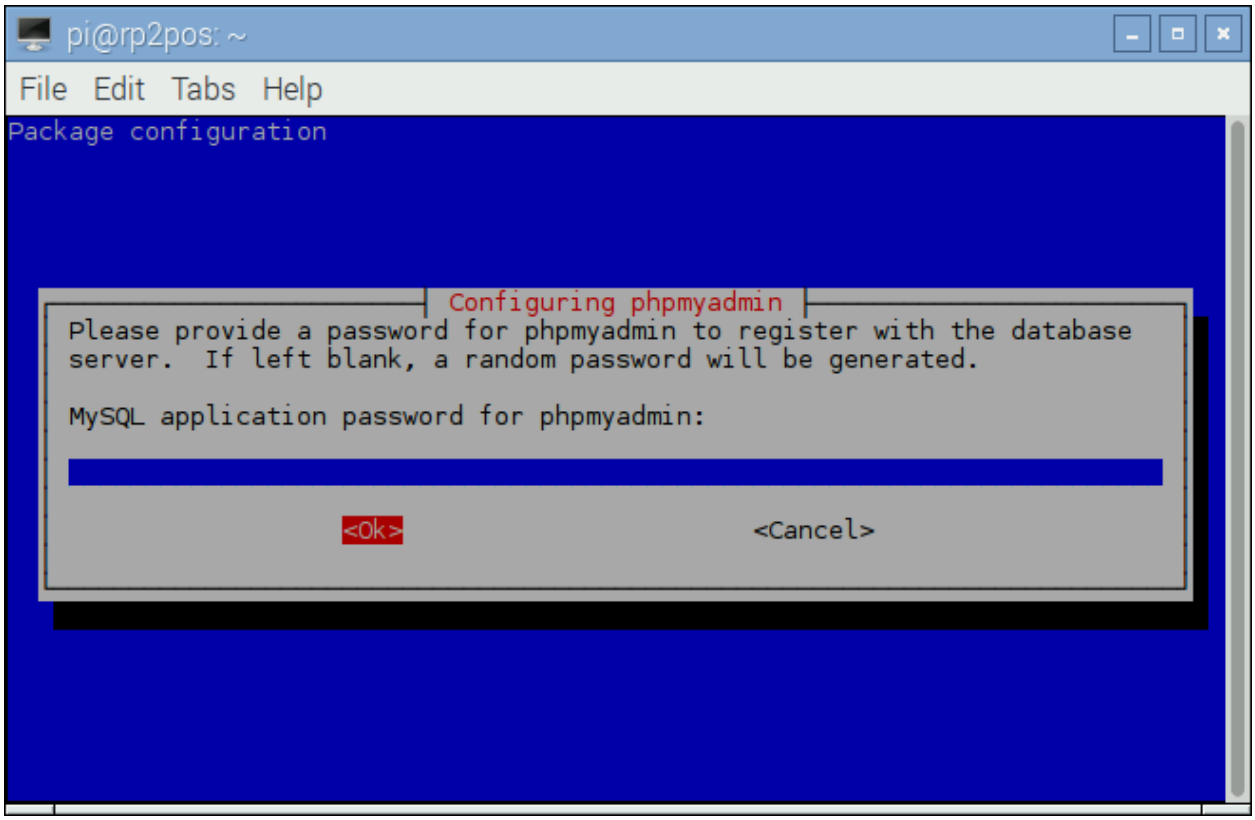
Sl. 2.10. *Konfiguracija baze podataka za phpMyAdmin*

Treći korak instalacije je unos administrativne zaporke korisnika „*root*“ kako bi čarobnjak za instalaciju napravio novu bazu podataka te upisao podatke za *phpMyAdmin* sučelje.



Sl. 2.11. *Unos administrativne zaporke korisnika „root“*

Nakon unosa administrativne zaporke korisnika „*root*“, čarobnjak pita koju lozinka se želi postaviti za bazu *phpMyAdmina*. Preporuča se da se na ovom koraku polje za zaporku ostavi prazno jer će se onda nasumično generirati lozinka. Stoga se na ovom koraku samo odabire oznaka „*Ok*“.



Sl. 2.12. *Unos zaporke za phpMyAdmin bazu podataka*

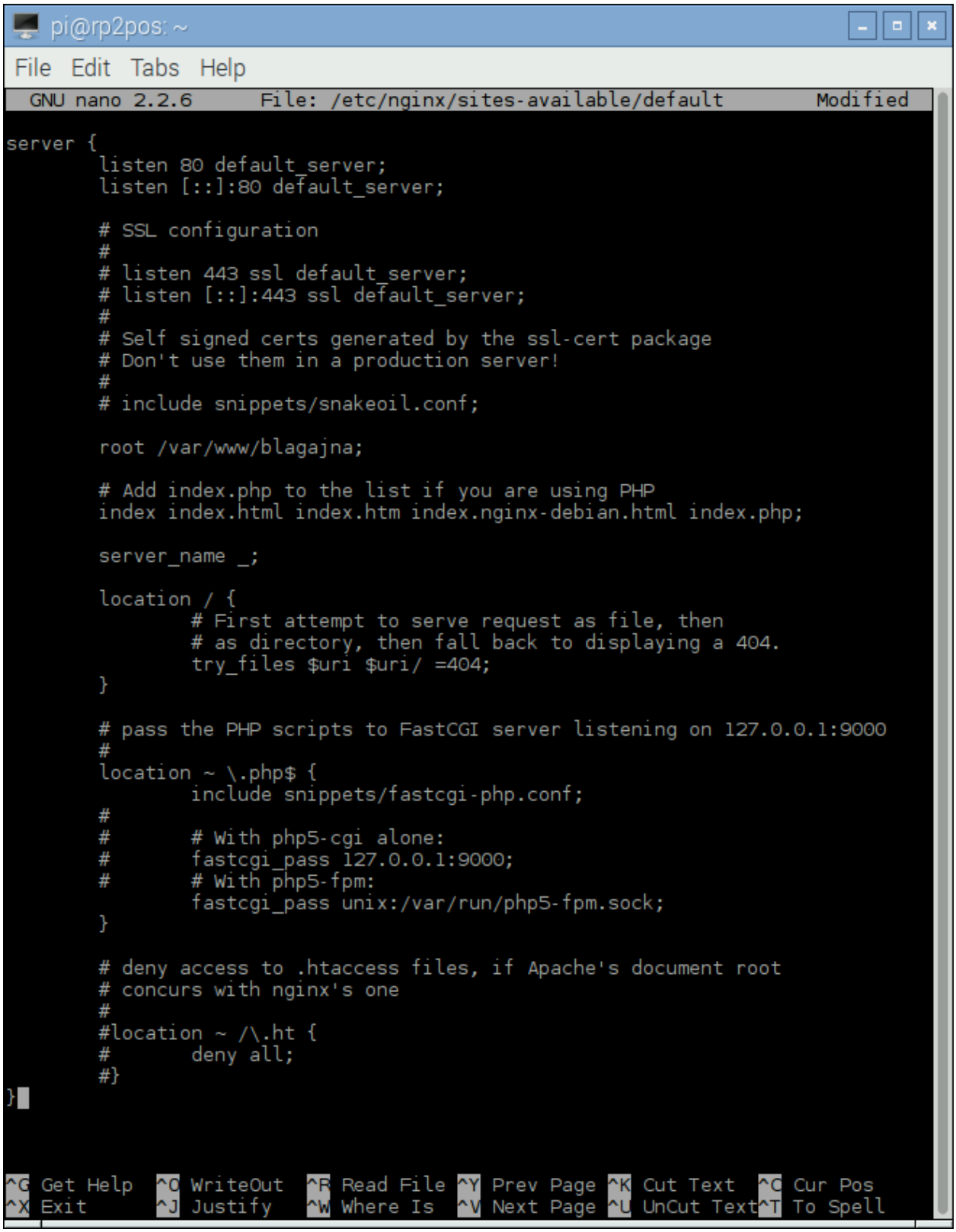
Ovo je bio zadnji korak instalacije *phpMyAdmina*.

2.8.1. Konfiguracija NGINX web poslužitelja

Kako bi web poslužitelj bio funkcionalan i koristio PHP, mora ga se konfigurirati. Za konfiguraciju je potrebna aplikacija za uređivanje teksta. Aplikacija koja se koristi zove se *nano*. Pomoću naredbe *nano* otvara se konfiguracijska datoteka web poslužitelja.

```
sudo nano /etc/nginx/sites-available/default
```

Kada se otvorila konfiguracijska datoteka, mijenja se parametar *root* koji govori gdje je glavna mapa koju će web poslužitelj gledati, dodaje se *index.php* kako bi web poslužitelj prepoznao ukoliko postoji početna stranica te se uključuje PHP kako bi radio sa web poslužiteljem. Konfiguracijsku datoteku možete vidjeti na slici (Sl. 2.13.).



Sl. 2.13. *Izmijenjena konfiguracijska datoteka web poslužitelja za potrebe blagajne*

Još je potrebno kreirati mapu koja je ranije postavljena u konfiguracijskoj datoteci te ponovno pokrenuti poslužitelj. To će se učiniti pomoću naredbi:

```
sudo mkdir /var/www/blagajna

sudo service nginx restart
```

2.8.2. Konfiguracija phpMyAdmina

Zbog ranije spomenutog problema, *phpMyAdmin* nije vidljiv *NGINX* web poslužitelju te je zbog toga potrebno napraviti prečac pomoću kojega će *phpMyAdmin* biti vidljiv poslužitelju. To će se učiniti sljedećom naredbom:

```
sudo ln –s /usr/share/phpmyadmin /var/www/blagajna/
```

3. STRUKTURA I PROGRAMSKI KOD BLAGAJNE

U ovome poglavlju bit će opisana struktura i programski kod blagajne. Prvo je potrebno spomenuti korištene jezike pri izradi blagajne, a zatim strukturu datoteka i same datoteke blagajne. Jezici koji su korišteni za izradu blagajne su HTML, CSS i PHP. Svaki programski jezik biti će objašnjen u par rečenica.

3.1. Jezici korišteni za izradu blagajne

3.1.1. HTML

HTML je kratica za *HyperText Markup Language* što znači prezentacijski jezik za izradu web stranica. Opće je prihvaćen i popularan, a zbog svoje jednostavnosti i vrlo raširen. Tim Berner-Lee je HTML po prvi puta spomenuo krajem 1991. godine, a prva inačica HTML jezika objavljena je 1993. godine. Danas se koristi HTML5 kao novija inačica jezika te donosi brojne mogućnosti.

3.1.2. CSS

CSS je kratica za *Cascading Style Sheets*, radi se o takozvanom stilskom jeziku za opis HTML jezika. Razvojem HTML se uvidjela potreba za stilskim jezikom koji će HTML osloboditi oblikovanja sadržaja. CSS-om se uređuje izgled stranice. CSS je razvio Hakon Wium Lee koji je radio na CERN-u zajedno s Timom Bernerom u listopadu 1993. godine.

3.1.3. PHP

PHP je objašnjen u poglavlju 2.3. kao poslužitelj i kao programski skriptni jezik.

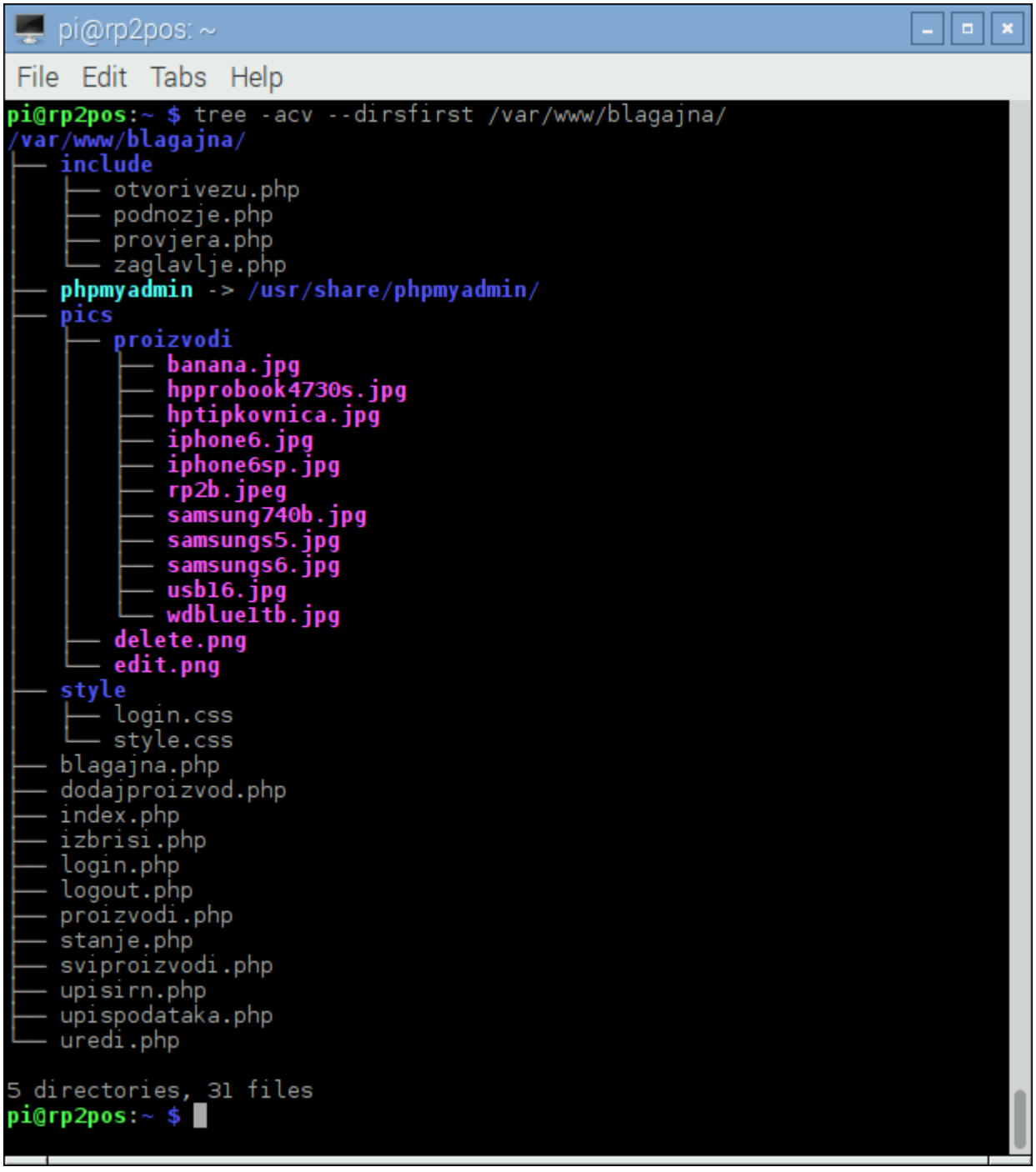
3.1.4. SQL

SQL (engl. *Structured Query* Language) je najpopularniji računalni jezik za traženje, izradu, brisanje i ažuriranje podataka iz relacijskih baza podataka. Nastao je 1970. godine i opće je prihvaćen.

3.2. Struktura datoteka blagajne

Blagajna se sastoji od 31 datoteke i 5 mapa. Na slici (Sl. 3.1.) je prikazana struktura datoteka i mapa. Struktura je prikazana sljedećom naredbom:

```
tree –acv –dirsfirst /var/www/blagajna/
```



Sl. 3.1. *Struktura datoteka i mapa blagajne*

3.3. Programski kod blagajne

U ovome potpoglavlju bit će objašnjen najvažniji dio iz svake pojedine datoteke. Za potrebe prikaza bit će korišten besplatni program *Notepad++*. Objašnjenja kreću od mape *include* i njezinih datoteka.

3.3.1. otvorivezu.php

Najvažniji dio koda u ovoj datoteci je funkcija *otvoriVezu()*. Funkcija služi kako bi se uspostavila komunikacija s bazom podataka. Varijabla *$veza* poprima vrijednosti naredbe *mysql_connect* te njezine parametre kao što su adresa poslužitelja, korisničko ime i lozinka za spajanje s bazom podataka. Naredbom *mysql_select_db* se odabire baza podataka, a njezini parametri su baza podataka te poslužitelj.

```
function otvoriVezu()
{
	$veza = mysql_connect("localhost", "matej", "qv5utUKcGfaMVCeD");
	if ($veza)
	{
		if(mysql_select_db(pos, $veza))
		{
			return $veza;
		}
		else
		{
			exit("Baza ne postoji!");
		}
	}
	else
	{
		exit("Nemoguće spajanje na MySQL bazu!");
	}
}
```

Odsječak programskog koda 3.1. – *Funkcija za uspostavljanje komunikacije s bazom podataka*

3.3.2. zaglavlje.php

U datoteci se nalazi zaglavlje koje je pozvano na svakoj stranici blagajne. Zaglavlje HTMLa sadrži podatke o imenu stranice, skupu znakova koji su korišteni na stranici te poziva vanjsku datoteku CSS-a u kojoj je oblikovanje stranice. U tijelu se nalazi kod koji ispisuje korisničko ime korisnika koji je trenutno prijavljen u sustav, a ukoliko je odjavljen, izbacuje poruku za ponovnu prijavu.

```
<!DOCTYPE html>
<html>
<head>
	<title>POS blagajna</title>
	<meta charset="utf-8">
	<link rel="stylesheet" type="text/css" href="style/style.css">
</head>
<body>
	<?php
		if(isset($_SESSION["username"]))
		{
			echo "<h4 class='korisnik'>Korisnik: " .
$_SESSION["username"] . " <a href='logout.php'>Odjava</a></h4><br>";
		}
		else
		{
			echo 'Molimo Vas prijavite se: <a href="login.php">Prijavi
se</a>';
		}
		 ?>
```

Odsječak programskog koda 3.2. – *Zaglavlje svakog dokumenta blagajne*

3.3.3. podnozje.php

U ovoj datoteci predefinirano je podnožje svake stranice te su zatvoreni *body* i *html* tagovi. Podnožje se poziva u svakoj stranici blagajne.

```
<footer>Copyright: mbaric @ ETFOS 2016</footer>
	</body>
</html>
```

Odsječak programskog koda 3.3. – *Podnožje svakog dokumenta blagajne*

3.3.4. provjera.php

U ovoj datoteci se nalazi dio koda koji započinje sjednicu i, ukoliko korisnik nije prijavljen, vraća ga na stranicu za prijavu.

```
<?php
	session_start();
	if(!isset($_SESSION["username"]))
	{
		header("Location: login.php");
	}
?>
```

Odsječak programskog koda 3.4. – *Pokretanje sjednice za pojedinog korisnika*

3.3.5. login.php

Prvi red koda poziva datoteku *otvorivezu.php* zbog poziva funkcije *otvoriVezu()*. Kod provjerava jesu li podaci koji su upisani u bazu podataka jednaki unesenim podacima u obrazac te, ukoliko se podaci podudaraju, korisnik je preusmjeren na glavni izbornik blagajne. Drugi dio koda je napisan u HTML-u i predstavlja obrazac u kojemu se od korisnika traži da unese svoje podatke za prijavu. Unos korisničkog imena predstavlja običan tekst dok kod unosa zaporke, zaporka nije vidljiva.

```php
<?php
	include("include/otvorivezu.php");
	session_start();
	$poruka = "";
	if (isset($_POST["username"]) && isset($_POST["password"])){
	  $korisnickoime = addSlashes($_POST["username"]);
	  $lozinka = addSlashes($_POST["password"]);
	  $veza = otvoriVezu();
	  $rezultat = mysql_query("SELECT * FROM korisnici WHERE username =
'$korisnickoime' AND password = '$lozinka'");
		if(mysql_num_rows($rezultat) == 1){
			$korisnik = mysql_fetch_array($rezultat);
			$_SESSION["username"] = $korisnickoime;
			$_SESSION["userid"] = $korisnik["id"];
			header("Location: index.php");
		}
		else{
			$poruka = "Neuspješna prijava";
		}
mysql_free_result($rezultat);
mysql_close($veza);
}
 ?>
<!DOCTYPE html>
<html>
<head>
	<meta charset="utf-8">
	<title>Prijava POS</title>
	<link rel="stylesheet" type="text/css" href="style/login.css">
</head>
<body>
	<div class="login-page">
		<div class="form">
			<form class="login-form" method="POST" action="login.php">
				<input type="text" name="username"
placeholder="Korisničko ime"></input>
				<input type="password" name="password"
placeholder="Lozinka"></input>
				<button>Prijava</button>
			</form>
		</div>
	</div>
</body>
</html>
```

Odsječak programskog koda 3.5. – *Prijava korisnika u sustav*

3.3.6. index.php

Na samom početku datoteke nalazi se kod u kojemu se pozivaju dvije datoteke iz mape *include*. Taj dio koda objašnjen je u ranijim potpoglavljima. Dalje se nalazi naslov kojemu je poravnanje centrirano, glavni izbornik blagajne te se na kraju poziva podnožje.

```
<?php
include("include/provjera.php");
include("include/zaglavlje.php");
 ?>
<h1 align="center">Navigacija</h1>
<ul class="nav">
      <li class="item"><a href="blagajna.php">POS Blagajna</a></li>
      <li class="item"><a href="proizvodi.php">Proizvodi</a></li>
      <li class="item"><a href="logout.php">Odjava</a></li>
</ul>
<?php
require_once("include/podnozje.php");
 ?>
```

Odsječak programskog koda 3.6. – *Navigacija glavnog izbornika*

3.3.7. blagajna.php

Na početku datoteke pozivaju se datoteke koje su ranije objašnjene u ovome radu. Sljedeći dio koda prikazuje *switch* petlju koja se sastoji od slučajeva *dodaj, izbriši i isprazni*. Slučaj *dodaj* dodaje proizvod na račun po barkodu te odabranoj količini proizvoda koji se dodaju na račun. Slučaj *izbriši* briše proizvod s računa također po njegovom barkodu. Slučaj *isprazni* odmah uklanja sve stavke s računa.

```php
<?php
include("include/provjera.php");
include("include/zaglavlje.php");
require_once("include/otvorivezu.php");
$db_kontrola = new KontrolaBaze();
if(!empty($_GET["action"])) {
switch($_GET["action"]) {
	case "dodaj":
		if(!empty($_POST["kolicina"])) {
			$proizpokodu = $db_kontrola->runQuery("SELECT * FROM
proizvodi WHERE code='" . $_GET["code"] . "'");
			$poljeProizvodi =
array($proizpokodu[0]["code"]=>array('imeproizvoda'=>$proizpokodu[0]["imeproi
zvoda"], 'code'=>$proizpokodu[0]["code"], 'kolicina'=>$_POST["kolicina"],
'cijena'=>$proizpokodu[0]["cijena"]));

			if(!empty($_SESSION["rn_proiz"])) {

	if(in_array($proizpokodu[0]["code"],$_SESSION["rn_proiz"])) {
				foreach($_SESSION["rn_proiz"] as $k => $v) {
					if($proizpokodu[0]["code"] == $k)

	$_SESSION["rn_proiz"][$k]["kolicina"] = $_POST["kolicina"];
				}
			} else {
				$_SESSION["rn_proiz"] =
array_merge($_SESSION["rn_proiz"],$poljeProizvodi);
			}
		} else {
			$_SESSION["rn_proiz"] = $poljeProizvodi;
		}
	}
	break;
	case "izbrisi":
		if(!empty($_SESSION["rn_proiz"])) {
			foreach($_SESSION["rn_proiz"] as $k => $v) {
				if($_GET["code"] == $k)
					unset($_SESSION["rn_proiz"][$k]);

				if(empty($_SESSION["rn_proiz"]))
					unset($_SESSION["rn_proiz"]);
			}
		}
	break;
	case "isprazni":
		unset($_SESSION["rn_proiz"]);
	break;
}
}
?>
```

Odsječak programskog koda 3.7. – *Zaglavlje i switch petlja za korištenje funkcija blagajne*

Ovaj dio koda prikazuje tablicu koja je zamišljena kao dodavanje stavki na račun. Tablica se sastoji od stupaca *ime, oznaka, količina, cijena i izbriši*. U stupac *ime* upisuju se imena proizvoda koji su dodani na račun, stupac *oznaka* se odnosi na barkod ili unikatnu oznaku proizvoda, *količina* se odnosi na broj istog proizvoda koji je dodan na račun. Stupac *cijena* se odnosi na cijenu proizvoda koji je dodan na račun te stupac *izbriši* nudi mogućnost uklanjanja pojedine stavke s računa. *Foreach* petlja upisuje podatke o dodanom proizvodu u tablicu spomenutu u prijašnjem tekstu. Varijabla *$item_total* množi cijenu proizvoda s količinom proizvoda te zbraja sve proizvode koji su dodani na račun. Varijabla *$item_total_pdv* množi kupan iznos s 1.25 kako bi se dobila stopa PDV-a koji u Republici Hrvatskoj iznosi 25%. Daljnji kod prikazuje još jednu *switch* petlju sa slučajem ispiši gdje se prilikom ispisa računa podaci o računu upisuju u bazu podataka. Na samom kraju dodana su tri retka u tablicu gdje su prikazani podaci o ukupnoj cijeni, cijeni s PDV-om te prozorčić za ispis računa.

```
<table cellpadding="10" cellspacing="1">
<tbody>
<tr>
<th><strong>Ime</strong></th>
<th><strong>Oznaka</strong></th>
<th><strong>Količina</strong></th>
<th><strong>Cijena</strong></th>
<th><strong>Izbrši</strong></th>
</tr>
<?php
    foreach ($_SESSION["rn_proiz"] as $item){
            ?>
                        <tr>
                        <td><strong><?php echo $item["imeproizvoda"];
?></strong></td>
                        <td><?php echo $item["code"]; ?></td>
                        <td><?php echo $item["kolicina"]; ?></td>
                        <td align=right><?php echo $item["cijena"]. " kn";
?></td>
                        <td><a href="blagajna.php?action=izbrisi&code=<?php
echo $item["code"]; ?>" class="btnIzbrisi">Izbriši proizvod</a></td>
                        </tr>
                        <?php
        $item_total += ($item["cijena"]*$item["kolicina"]);
        $item_total_pdv = ($item_total*1.25);
        $zaokruzi_2dec = number_format((double)$item_total_pdv, 2, '.', '');
        $racun[]= $item;
        }
            ?>
            <?php // print_r($racun);?>
            <?php
            if(!empty($_GET["action"])) {
            switch($_GET["action"]) {
            case "ispisi":
            $veza = otvoriVezu();
            if(is_array($racun)) {

            $sql = "INSERT INTO prodano (imeproiz, cijena, barkod, kol,
blagajnik, ukupno) values ";
      $valuesArr = array();
      foreach($racun as $row){
        $ime_proiz = $row['imeproizvoda'];
        $cijena = mysql_real_escape_string( $row['cijena'] );
        $kod = mysql_real_escape_string( $row['code'] );
        $kolic = mysql_real_escape_string($row['kolicina']);
        $blag = mysql_real_escape_string($_SESSION['username']);
        $ukup = $zaokruzi_2dec;
        $valuesArr[] = "('$ime_proiz', '$cijena', '$kod', '$kolic', '$blag',
'$ukup')";
    }
    $sql .= implode(',', $valuesArr);
    mysql_query($sql) or exit(mysql_error());

}
    $poruka = "Uspjesno ste izdali racun!";
      break;
}
}
             ?>

<tr>
```

```
<td colspan="5" align=right><strong>Ukupno:</strong> <?php echo $item_total. "
kn"; ?></td>
</tr>
<tr>
<td colspan="5" align=right><strong>Ukupno s PDV:</strong> <?php echo
$zaokruzi_2dec. " kn"; ?></td>
</tr>
<tr>
<td colspan="5" align=right><a id="btnIsprazni"
href="blagajna.php?action=ispisi">Ispiši račun</a><?php echo $poruka; ?></td>
</tr>
</tbody>
</table>
  <?php
}
?>
</div>
```

Odsječak programskog koda 3.8. – *Dodavanje stavki na račun, logika za računanje PDV a i upis prodanih proizvoda u bazu podataka*

Zadnji dio koda u ovoj datoteci prikazuje listu proizvoda koji su ranije dodani u bazu podataka. Proizvodi se ispisuju tako da im je prikazano ime proizvoda, barkod ili oznaka, slika te cijena proizvoda. Na kraju se nalazi prozorčić pomoću kojega se proizvod dodaje na račun.

```
<div id="proizvodi-lista">
      <div class="txt-naslov">Proizvodi u bazi</div>
      <?php
      $proizvodi_ispis = $db_kontrola->runQuery("SELECT * FROM proizvodi ORDER
BY id ASC");
      if (!empty($proizvodi_ispis)) {
            foreach($proizvodi_ispis as $key=>$value){
      ?>
            <div class="proiz-item">
                  <form method="post"
action="blagajna.php?action=dodaj&code=<?php echo
$proizvodi_ispis[$key]["code"]; ?>">
                  <div class="proiz-slika"><img src="<?php echo
$proizvodi_ispis[$key]["slika"]; ?>"></div>
                  <div><strong><?php echo
$proizvodi_ispis[$key]["imeproizvoda"]; ?></strong></div>
                  <div class="proiz-cijena"><?php echo
$proizvodi_ispis[$key]["cijena"]. " kn"; ?></div>
                  <div><input type="text" name="kolicina" value="1" size="2"
/><input type="submit" value="Dodaj na račun" class="btnDodaj" /></div>
                  </form>
            </div>
      <?php
                  }
      }
      ?>
</div>
```

Odsječak programskog koda 3.9. – *Dohvaćanje i prikaz proizvoda iz baze podataka*

3.3.8. proizvodi.php

U ovoj datoteci napravljen je izbornik za proizvode pomoću neodređene liste.

```
<?php
include("include/provjera.php");
include("include/zaglavlje.php");
?>
<h1 align="center">Navigacija</h1>
<ul class="nav">
      <li class="item"><a href="sviproizvodi.php">Svi proizvodi</a></li>
      <li class="item"><a href="stanje.php">Stanje proizvoda</a></li>
      <li class="item"><a href="dodajproizvod.php">Dodaj proizvod</a></li>
      <li class="item"><a href="index.php">Natrag</a></li>
</ul>
<?php require_once("include/podnozje.php"); ?>
```

Odsječak programskog koda 3.10. – *Izbornik napisan pomoću neodređene liste*

3.3.9. **sviproizvodi.php**

Prvi dio koda poziva funkciju *otvoriVezu()* iz datoteke otvorivezu.php kako bi se dohvatio popis svih proizvoda iz baze podataka. Svi podaci su upisani u tablicu koja sadrži podatke o proizvodu kao što su id, ime proizvoda, proizvođač, masa, količina, dobavljač, barkod te cijena. Tablica sadrži i jedan dodatni stupac u kojem se nalazi dio u kojemu je moguće izbrisati proizvod iz baze ili ga urediti. Na kraju je potrebno prekinuti komunikaciju s MySQL poslužiteljem.

```
<h1 align="center">Proizvodi</h1>
        <hr>
        <?php
        $veza = otvoriVezu();
        $rezultat = mysql_query("SELECT * FROM proizvodi");
        echo "<table class='tablica' border='1'>
                        <tr>
                        <th>ID</th>
                        <th>Ime Proizvoda</th>
                        <th>Proizvođač</th>
                        <th>Masa</th>
                        <th>Količina</th>
                        <th>Dobavljač</th>
                        <th>Barkod</th>
                        <th>Cijena</th>
                        <th>Admin</th>
                </tr>";
                while($redak = mysql_fetch_array($rezultat)){
                        echo "<tr>";
                                echo "<td>" . $redak['id'] . "</td>";
                                echo "<td>" . $redak['imeproizvoda'] .
"</td>";
                                echo "<td>" . $redak['proizvodac'] . "</td>";
                                echo "<td>" . $redak['masa'] . "</td>";
                                echo "<td>" . $redak['kolicina'] . "</td>";
                                echo "<td>" . $redak['dobavljac'] . "</td>";
                                echo "<td>" . $redak['code'] . "</td>";
                                echo "<td>" . $redak['cijena'] . "</td>";
                                echo "<td><a href=\"izbrisi.php?id="
.$redak['id']. "\"><img alt='Izbrisi' src='pics/delete.png' width='18'
height='18'></a><a href=\"uredi.php?id=" .$redak['id']. "\"><img alt='Uredi'
src='pics/edit.png' width='18' height='18'></a></td>";
                        echo "</tr>";
                }
        echo "</table>";
        mysql_close($veza);
            echo "</table>";
        mysql_close($veza);
        ?>
```

Odsječak programskog koda 3.11. – *Dohvaćanje podataka iz baze i ispis istih u tablicu*

3.3.10. stanje.php

Kod ove datoteke je vrlo sličan kodu datoteke *sviproizvodi.php* osim što se u tablicu ispisuju podaci o stanju nabavljenih proizvoda.

```
<h1> align="center">Proizvodi</h1>
        <hr>
        <?php
        $veza = otvoriVezu();
        $rezultat = mysql_query("SELECT proizvodi.imeproizvoda,
proizvodi.code, proizvodi.kolicina, prodano.kol, prodano.barkod,
(proizvodi.kolicina - prodano.kol) AS razlika FROM proizvodi JOIN prodano ON
proizvodi.code=prodano.barkod GROUP by code");
        echo "<table class='tablica' border='1'>
                        <tr>
                        <th>ID</th>
                        <th>Ime Proizvoda</th>
                        <th>Barkod</th>
                        <th>Nabavljena količina</th>
                        <th>Stvarno stanje</th>
                        <th>Admin</th>
                </tr>";
                while($redak = mysql_fetch_array($rezultat)){
                        echo "<tr>";
                                echo "<td>" . $redak['id'] . "</td>";
                                echo "<td>" . $redak['imeproizvoda'] .
"</td>";
                                echo "<td>" . $redak['code'] . "</td>";
                                echo "<td>" . $redak['kolicina'] . "</td>";
                                echo "<td>" . $redak['razlika'] . "</td>";
                                echo "<td><a href=\"izbrisi.php?id="
.$redak['id']. "\"><img alt='Izbrisi' src='pics/delete.png' width='18'
height='18'></a><a href=\"uredi.php?id=" .$redak['id']. "\"><img alt='Uredi'
src='pics/edit.png' width='18' height='18'></a></td>";
                        echo "</tr>";
                }
        echo "</table>";
        mysql_close($veza);
            echo "</table>";
        mysql_close($veza);
        ?>
```

Odsječak programskog koda 3.12. – *Dohvaćanje podataka iz baze i ispis u tablicu stanja*

3.3.11. dodajproizvod.php

Kod prikazuje obrazac za dodavanje novog proizvoda u bazu podataka koji je napravljen u obliku neodređene liste. Klikom na prozorčić „Upiši podatke“ poziva se datoteka *upispodataka.php* te se podaci upisuju u bazu podataka.

```
<h1 align="center">Unos proizvoda</h1>
                <hr>
                <form id="unosforma" method="post" action="upispodataka.php">
                        <ul>
              <li class="label">Ime proizvoda: </li>
              <li class="unos"> <input type="text" name="imeproizvoda"/></li>
              <li class="label">Proizvođač: </li>
              <li class="unos"> <input type="text" name="proizvodac"/></li>
              <li class="label">Masa: </li>
              <li class="unos"> <input type="text" name="masa"/></li>
              <li class="label">Količina: </li>
              <li class="unos"> <input type="text" name="kolicina"/></li>
              <li class="label">Dobavljac: </li>
              <li class="unos"> <input type="text" name="dobavljac"/></li>
              <li class="label">Barkod: </li>
              <li class="unos"> <input type="text" name="code"/></li>
              <li class="label">Cijena: </li>
              <li class="unos"> <input type="text" name="cijena"/></li>
                        </ul>
                        <input id="gumb" type="submit" value="Upiši podatke">
                        <hr>
                </form>
```

Odsječak programskog koda 3.13. – *Obrazac za dodavanje novoga proizvoda*

3.3.12. logout.php

Prikazan je dio koda kojim se uništava postojeća sjednica te se korisnik preusmjerava na stranicu prijave.

```
<?php
      session_start();
      session_destroy();
      header("Location: login.php");
 ?>
```

Odsječak programskog koda 3.14. – *Odjava iz sustava*

3.3.13. izbrisi.php

Varijabla *$id* poprima vrijednost rednog broja unesenog proizvoda te se po tome broju izvršava brisanje proizvoda iz baze podataka.

```php
<?php
include("include/provjera.php");
include("include/otvorivezu.php");
$id = $_GET['id'];
$veza = otvoriVezu();
$rezultat = mysql_query("DELETE FROM proizvodi WHERE id='".$id."'");
header("Location: sviproizvodi.php");
?>
```

Odsječak programskog koda 3.15. – *Brisanje proizvoda iz baze podataka pomoću identifikacijske oznake*

3.3.14. uredi.php

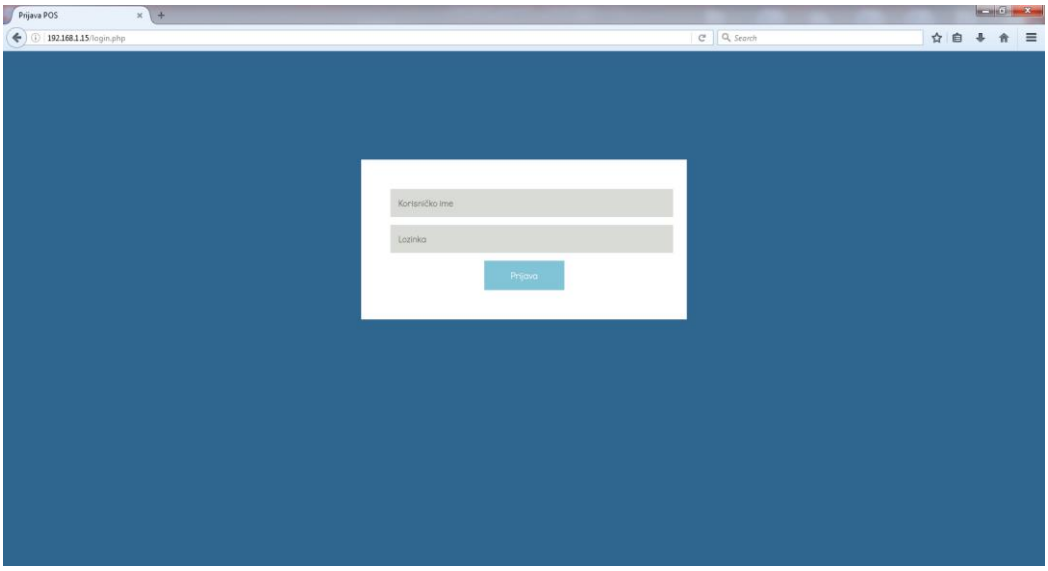
Uređivanje se, kao i brisanje, provodi putem rednog broja proizvoda. Kod ove datoteke sličan je kodu datoteke *dodajproizvod.php.* U obrazac su ispisani svi podaci o proizvodu koji se uređuje te nakon promjene nekoga od podataka klikom na prozorčić „Uredi podatke“, šalje se SQL naredba koja će ažurirati podatke u bazi podatka.

```php
<?php
include("include/provjera.php");
include("include/otvorivezu.php");
include("include/zaglavlje.php");
$id = $_GET['id'];
$veza = otvoriVezu();
$rezultat = mysql_query("SELECT * FROM proizvodi WHERE id='$id'");
$redak = mysql_fetch_array($rezultat);
// Logika
if (isset($_POST['uredi'])){
$sql = "UPDATE proizvodi SET imeproizvoda='$_POST[imeproizvoda]',
proizvodac='$_POST[proizvodac]', masa='$_POST[masa]',
kolicina='$_POST[kolicina]', dobavljac='$_POST[dobavljac]',
code='$_POST[code]', cijena='$_POST[cijena]' WHERE id='$_POST[userid]'";
if (!mysql_query($sql, $veza)){
        die('Greska: ' . mysql_error());
}
else{
        header("Location: sviproizvodi.php");
}
}
mysql_close($veza);
?>
<h1 align="center">Uredi proizvod</h1>
                <hr>
                <form id="editforma" method="post" action="uredi.php">
                        <ul>
                                <li class="label">ID: </li>
                                <li class="unos"> <input type="text"
name="userid" value="<?php echo $id; ?>" readonly /></li>
                                <li class="label">Ime proizvoda: </li>
                                <li class="unos"> <input type="text"
name="imeproizvoda" value="<?php echo $redak['imeproizvoda'] ?>" /></li>
                                <li class="label">Proizvođač: </li>
                                <li class="unos"> <input type="text"
name="proizvodac" value="<?php echo $redak['proizvodac'] ?>" /></li>
                                <li class="label">Masa: </li>
                                <li class="unos"> <input type="text"
name="masa" value="<?php echo $redak['masa'] ?>" /></li>
                                <li class="label">Količina: </li>
                                <li class="unos"> <input type="text"
name="kolicina" value="<?php echo $redak['kolicina'] ?>" /></li>
                                <li class="label">Dobavljač: </li>
                                <li class="unos"> <input type="text"
name="dobavljac" value="<?php echo $redak['dobavljac'] ?>" /></li>
                                <li class="label">Barkod: </li>
                                <li class="unos"> <input type="text"
name="code" value="<?php echo $redak['code'] ?>" /></li>
                                <li class="label">Cijena: </li>
                                <li class="unos"> <input type="text"
name="cijena" value="<?php echo $redak['cijena'] ?>" /></li>
                        </ul>
                        <input id="gumb" name="uredi" type="submit"
value="Uredi podatke">
                        <hr>
                </form>
<?php echo '<a class="naz_nav" href="sviproizvodi.php">Natrag</a>'; ?>
<?php require_once("include/podnozje.php"); ?>
```

Odsječak programskog koda 3.16. – *Obrazac za promjenu podataka i logika za ažuriranje baze*

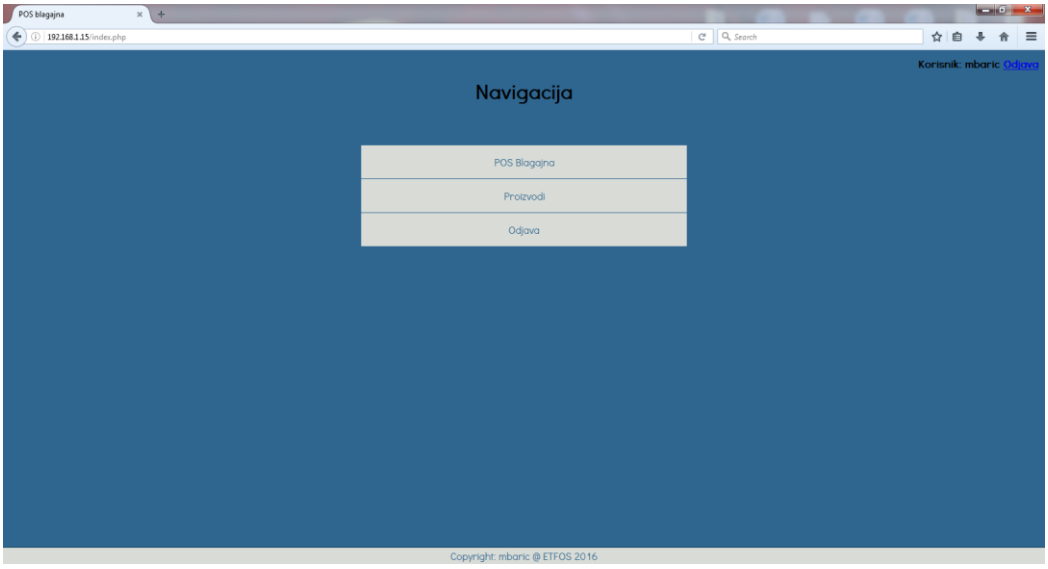
4. IZGLED I FUNKCIONALNOST BLAGAJNE

Prilikom izrade blagajne za Raspberry Pi 2 platformu pozornost nije bila samo na funkcionalnosti blagajne, već i na njezinom izgledu i oblikovanju. Na sljedećim slikama se može vidjeti rezultat rada.



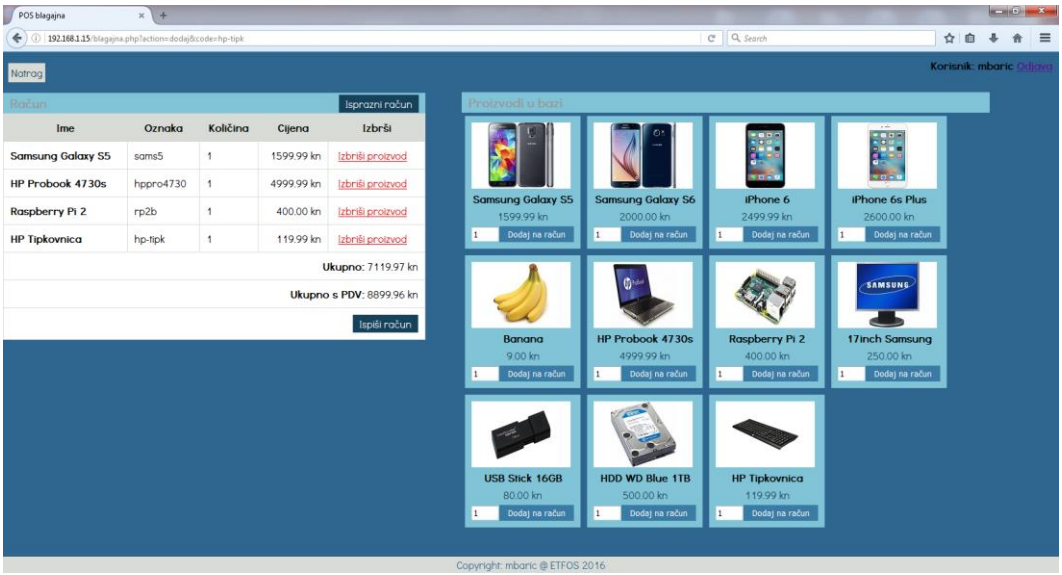
Sl. 4.1. *Prijava u sustav blagajne*

Kako bi korisnik mogao pristupiti blagajni, mora imati odgovarajuće korisničko ime i zaporku. Prilikom prijave korisnik svoje korisničko ime upisuje u polje *korisničko ime* te svoju zaporku u polje *lozinka*. Nakon što su sva polja popunjena, prijava se izvršava pritiskom na prozorčić „Prijava“.



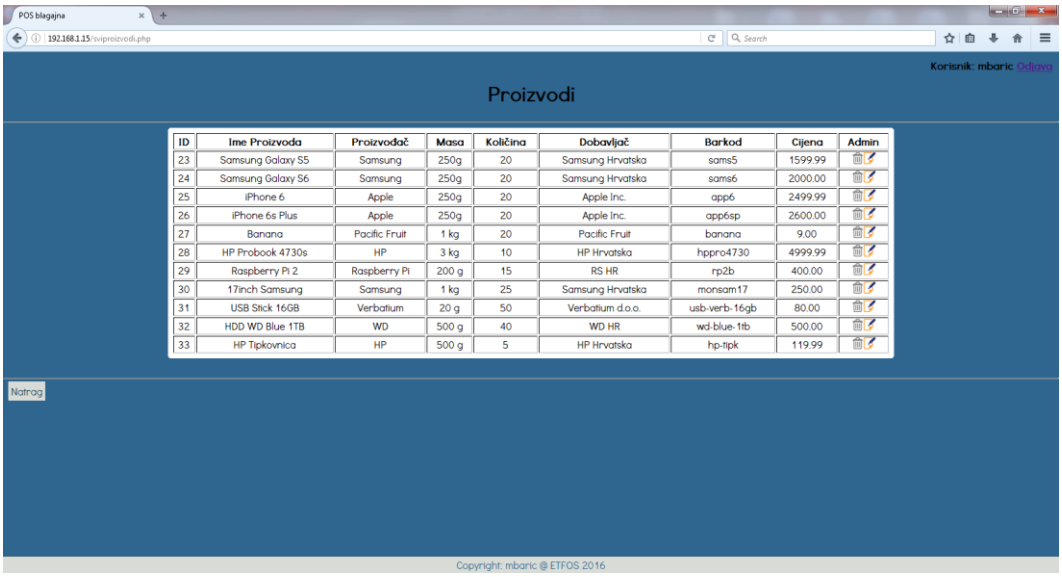
Sl. 4.2. *Početna stranica i glavni izbornik blagajne*

Ako je korisnik uspješno prijavljen, nalazi se na glavnom izborniku blagajne gdje se nalaze tri mogućnosti: pristup blagajni, pristup proizvodima te odjava iz aplikacije.



Sl. 4.3. *Stranica blagajne s proizvodima ispisanima iz baze, te dodanim proizvodima na račun*

Slika (Sl. 4.3.) prikazuje blagajnu. S desne strane mogu se vidjeti proizvodi koji su upisani u bazu podataka, njihove slike, cijena. Korisnik može odabrati količinu proizvoda koje želi dodati na račun, a zatim klikom na prozorčić „Dodaj na račun“ proizvod se dodaje u tablicu. Tablica s lijeve strane slike (Sl. 4.3.) predstavlja proizvode koji su dodani na račun. U tablici su prikazani ime proizvoda, oznaka ili barkod, količina dodanog proizvoda i cijena proizvoda. Nakon što su dodani željeni proizvodi, skripta izračunava ukupan iznos računa te dodaje PDV koji u Hrvatskoj iznosi 25%. U tablici se nalaze i dva prozorčića; prvi prozorčić, „Isprazni račun“, briše sve stavke računa, a prozorčić „Ispiši račun“ ispisuje račun te sprema podatke u bazu podataka.



Sl. 4.4. *Ispis svih proizvoda i podataka o istima iz baze podataka*

Slika (Sl. 4.4.) predstavlja stranicu na kojoj su ispisani svi proizvodi i podaci koji se nalaze u bazi podataka. U stupcu *admin* nalaze se dva prozorčića „Izbriši“ i „Uredi“ koja omogućuju korisniku da obriše proizvode iz baze podataka ili urede postojeće podatke o proizvodu.

Sl. 4.5. *Unos novih proizvoda u bazu podataka*

Slika (Sl.4.5.) prikazuje obrazac za unos novih proizvoda u bazu podataka. Kako bi se proizvod upisao u bazu, potrebno je unijeti sljedeće podatke: ime proizvoda, proizvođač, masa, količina, dobavljač, barkod, cijena. Nakon što su popunjena sva polja obrasca, klikom na prozorčić „Upiši podatke“ podaci se upisuju u bazu podataka.

4.1. Izgled baze podataka

Baza korištena za potrebe aplikacije blagajne naziva se „pos“, a sastoji se od tri tablice: korisnici, proizvodi i prodano.

Tablica *korisnici* koristi se za unos korisnika koji će koristiti blagajnu. Sastoji se od 5 atributa. Atributi tablice *korisnici* su: id, username, password, ime, prezime. Atribut *id* je primarni ključ ove tablice i drugačiji je za svakog korisnika. Struktura tablice *korisnici* može se vidjeti u tablici (Tab. 4.1.).

Tab. 4.1. *Prikaz strukture tablice korisnici u bazi podataka „pos“.*

Ime	Tip	Dodatno	Opis
id	INT	Primarni ključ	Id korisnika
username	VARCHAR(50)	-	Korisničko ime
password	VARCHAR(50)	-	Zaporka
ime	VARCHAR(50)	-	Ime korisnika
prezime	VARCHAR(50)	-	Prezime korisnika

Tablica *proizvodi* sadrži sve informacije o proizvodima koji su upisani od strane korisnika aplikacije blagajne. Sadrži sljedeće atribute: id, imeproizvoda, proizvodac, masa, kolicina, dobavljac, cijena, code i slika. Primarni ključ tablice je *id* koji mora biti različit za svaki proizvod. Postoje i dva unikatna ključa, a to su code i slika. Svaki proizvod mora imati drugačiju oznaku, odnosno barkod i na tome principu radi skripta za dodavanje, odnosno brisanje proizvoda sa računa. Struktura tablice može se vidjeti na slici (Tab. 4.2.).

Tab. 4.2. *Struktura tablice proizvodi u bazi podataka „pos“.*

Ime	Tip	Dodatno	Opis
Id	INT	Primarni ključ	Id proizvoda
imeproizvoda	VARCHAR(50)	-	Ime proizvoda
proizvodac	VARCHAR(50)	-	Proizvođač
masa	VARCHAR(10)	-	Masa proizvoda
kolicina	INT	-	Količina proizvoda
dobavljac	VARCHAR(50)	-	Dobavljač
cijena	DOUBLE(10,2)	-	Cijena proizvoda
code	VARCHAR(255)	Unikatni ključ	Barkod ili oznaka
slika	TEXT	Unikatni ključ	Slika proizvoda

Zadnja tablica u bazi podataka „pos“ *prodano* u koju se upisuju podaci o proizvodima nakon što je za njih izdan račun. Bilježe se osnovni podaci kao što su vrijeme izdavanja računa, broj, računa, ukupan iznos i drugo. Atributi tablice *prodano* su: rnid, datum, blagajnik, imeproiz, kol, cijena, barkod i ukupno. Struktura tablice *prodano* prikazana je tablicom (Tab. 4.3.).

Tab. 4.3. *Struktura tablice prodano u bazi podataka „pos“.*

Ime	Tip	Dodatno	Opis
Rnid	INT	Primarni ključ	Id računa
Datum	TIMESTAMP	Trenutni datum i vrijeme	Datum i vrijeme izdavanja rn.
Blagajnik	VARCHAR(255)	-	Blagajnik koji je izdao račun
Imeproiz	VARCHAR(50)	-	Ime proizvoda
Kol	INT	-	Prodana količina
Cijena	DOUBLE(10,2)	-	Cijena proizvoda
Barkod	VARCHAR(255)	Unikatni ključ	Barkod ili oznaka
Ukupno	DOUBLE(10,2)	-	Ukupan iznos računa

5. ZAKLJUČAK

Raspberry Pi 2 je jednopločno računalo veličine kreditne kartice. Dovoljno je snažno da može činiti sve što i jedno stolno računalo. Time se ta platforma postavlja kao idealno rješenje za blagajnu. Nečujno pri radu, dovoljno malo da ne smeta radnicima u obavljanju poslova i dovoljno snažno da pokreće blagajnu i ostale programe koje radnici koriste. Platformu nije potrebno nikada gasiti jer se ne grije, a potrošnja resursa je minimalna što znači da je sve uvijek dostupno. Velika prednost poslodavca je što se može u svakom trenutku udaljeno spojiti na računalo i provjeriti stanja proizvoda i blagajne te preuzeti dokumente koje su mu potrebni. Isto tako, ni osoba koja održava sustav ne mora biti fizički prisutna nego može udaljeno riješiti eventualni problem.

Blagajna je napisana u jezicima koje spadaju u grupu web tehnologija iz razloga što imaju velike prednosti kao što su udaljeni pristup blagajni, brzina rada blagajne, dizajn te sama jednostavnost korištenja. Jezike HTML i CSS je vrlo lako naučiti i zapamtiti što je za programera velika prednost, a nude puno mogućnosti, od prikaza običnog teksta do prikaza video zapisa. PHP je nešto složeniji skriptni programski jezik koji nudi pregršt mogućnosti, a u ovome projektu je korišten za spajanje s bazom podataka, ispisivanje podataka, upisivanje podataka u bazu, operacije računanja te ostale mogućnosti koje jezici HTML i CSS nisu u mogućnosti podržati.

Spajanjem Raspberry Pi 2 platforme i jezika web tehnologija dobivena je cjelina koja je pogodna za korištenje ovakvoga i sličnih projekata.

LITERATURA

https://www.raspberrypi.org/products/raspberry-pi-2-model-b/ - Raspberry Pi 2 Model B – 8.6.2016.

https://www.raspbian.org/ - Raspbian – 8.6.2016.

https://en.wikipedia.org/wiki/Raspbian - Raspbian – 9.6.2016.

https://hr.wikipedia.org/wiki/Secure_Shell - SSH - 9.6.2016.

https://www.nginx.com/resources/wiki/# - NGINX – 9.6.2016.

https://en.wikipedia.org/wiki/Nginx - NGINX - 9.6.2016.

http://php.net/manual/en/introduction.php - PHP - 9.6.2016.

https://hr.wikipedia.org/wiki/PHP - PHP - 9.6.2016.

https://hr.wikipedia.org/wiki/MySQL - MySQL - 9.6.2016.

https://en.wikipedia.org/wiki/MySQL - MySQL - 9.6.2016.

https://www.phpmyadmin.net/ - phpMyAdmin – 9.6.2016.

https://hr.wikipedia.org/wiki/HTML- HTML - 22.6.2016.

https://hr.wikipedia.org/wiki/CSS - CSS - 22.6.2016.

https://en.wikipedia.org/wiki/Cascading_Style_Sheets#History – CSS – 22.6.2016.

SAŽETAK

Raspberry Pi je računalo veličine kreditne kartice koje se uključuje u TV ili monitor te koristi tipkovnicu i miš. Može činiti sve što i jedno stolno računalo, od pretraživanja Interneta, procesiranja visoko razlučivog video sadržaja, uređivanja teksta do pokretanja igrica. Štoviše, postoji i mogućnost povezivanja s ostalim uređajima. Raspberry Pi 2 Model B je računalo druge generacije Raspberry Pia. Druga generacija uspješno pokreće Android, Linux, a za razliku od prijašnjih, pokreće i Windows 10 operacijski sustav. Za potrebe završnog rada koristiti će se operacijski sustav Raspbian. Raspbian je temeljen na Debian distribuciji Linux operacijskog sustava. Ciljano je dizajniran za sklopovlje Raspberry Pia te se sastoji od osnovnih programa i alata koji su potrebni za rad. Programi koji će biti korišteni kako bi blagajna funkcionirala su: NGIN web poslužitelj, PHP moduli, MySQL poslužitelj i phpMyAdmin. Nakon instalacije svih programa potrebno je konfigurirati NGINX web poslužitelj tako da nadgleda mapu u kojoj se nalaze datoteke blagajne. Isto tako je potrebno i konfigurirati phpMyAdmin kako bi bio vidljiv web poslužitelju.

Pri izradi blagajne korišteni su HTML, CSS, PHP i SQL. HTML je kratica za HyperText Markup Language što znači prezentacijski jezik za izradu web stranica. Opće je prihvaćen i popularan, a zbog svoje jednostavnosti i vrlo raširen. Razvojem HTML se uvidjela potreba za stilskim jezikom koji će HTML osloboditi oblikovanja sadržaja . CSS-om se uređuje izgled stranice. PHP je široko korišten skriptni programski jezik namijenjen programiranju web stranica. Distribuiran je kao skriptni programski jezik otvorenog koda. Ističe se širokom podrškom raznih baza podataka i Internet protokola te mogućnosti komponiranja u HTML. SQL je najpopularniji računalni jezik za traženje, izradu, brisanje i ažuriranje podataka iz relacijskih baza podataka.

Ključne riječi: Raspberry Pi, Raspbian, PHP, HTML, CSS, SQL, NGINX, MySQL, phpMyAdmin, blagajna

ABSTRACT

Development of cash register for Raspberry Pi 2 platform

Raspberry Pi is credit-card sized computer that plugs into TV or computer monitor and uses mouse and keyboard. It's capable of doing everything you would expect a desktop computer to do, from Internet browsing, playing high-definition video, editing text to playing games. What's more it can be connected to other devices. Raspberry Pi 2 Model B is second generation of Raspberry Pi devices. Second generation is capable to run Android, Linux but unlike previous generations it can run Windows 10 operating system. For purposes of this project we will use Raspbian operating system. Raspbian is based on Debian distribution of Linux operating system. It is optimized for Raspberry Pie's hardware and it has basic programs and utilities that are required for normal work. Programs that will be used for this project are: NGINX web server, MySQL server and phpMyAdmin. After installation of all programs it is necessary to configure NGINX web server to watch directory where application files are. It is also necessary to configure phpMyadmin so it can be watched by web server.

In preparation of application are used HTML, CSS, PHP and SQL. HTML stands for HyperText Markup Language which means the presentation language for web development. It is widely accepted and popular and because of its simplicity is its highly prevalent. With development of HTML there was need for stylistic language that would relieve HTML design. CSS is governing the page layout. PHP is widely used scripting programming language used for web development. It is distributed as open-source scripting programming language. It is emphasized with wide support for various databases and Internet protocols and it has possibility to be used with HTML. SQL is the most popular special-purpose programming language for searching, creating, deleting and updating data from relation databases.

Keywords: Raspberry Pi, Raspbian, PHP, HTML, CSS, SQL, NGINX, MySQL, phpMyAdmin, cash desk

ŽIVOTOPIS

Matej Barić rođen je 7. lipnja 1994. godine u Vinkovcima. Odrastao je i živi u Ivankovu u peteročlanoj obitelji s roditeljima i dva mlađa brata. Školovanje je započeo u osnovnoj školi „August Cesarec“ u Ivankovu. Srednjoškolsko obrazovanje nastavio je u Ekonomskoj i trgovačkoj školi Ivana Domca u Vinkovcima gdje je 2013. godine maturirao i stekao zvanje ekonomiste. Nakon završenog srednjoškolskog obrazovanja nastavio je daljnje školovanje na Elektrotehničkom fakultetu u Osijeku na stručnom studiju Elektrotehnike, smjer Informatika. Tijekom studija obavljao je stručnu praksu u HŽ Infrastrukturi na najmodernijem sustavu elektronske regulacije, kontrole i upravljanja prometom vlakova u Republici Hrvatskoj na relaciji Vinkovci – Tovarnik.