



Šolski center Celje

Srednja šola za kemijo, elektrotehniko in računalništvo

Postavitev Raspberry Pi spletnega strežnika

Raziskovalna naloga

Avtorja: Tim Puhič, Luka Slapnik

Računalništvo

Mentor: Žiga Pušcl, univ. dipl. inž. el.

Mestna občina Celje, Mladi za Celje

Celje, april 2022

Zahvala

Zahvaljujeva se vsem, ki so nama pri izdelovanju naloge pomagali na tak ali drugačen način. V prvi vrsti gre zahvala najinemu mentorju, Žigi Pušelcu, ki naju je vseskozi nalogo usmerjal in nama svetoval, ko česa nisva vedela. Prav tako gre zahvala najinima družinama, ki sta nama omogočili raziskovanje samo. Hvala vsem, ki so naju med delom spodbujali in nama pomagali.

Povzetek

Ideja, da lahko celotno spletno stran gostujemo na napravi velikosti polovice pametnega telefona, je večini ljudem popolnoma tuja in mnogim celo neverjetna.

Ta koncept sva udeleževala in raziskala v raziskovalni nalogi. Ustvarila sva spletni strežnik z uporabo mikroračunalnika Raspberry Pi. Hkrati sva najprej teoretično, nato pa še z empiričnimi podatki, raziskala različne vrste uporab strežnikov ter našla najprimernejše rešitve za vse vrste uporabnikov. Želela sva ugotoviti, če je mogoče z uporabo Raspberry Pi-ja prihraniti z denarjem in energijo. Na njem sva preizkusila najbolj priljubljene sisteme spletnih strežnikov, spletnih aplikacij ter tudi sistemov za oddaljen dostop. Nalogo sva končala z izdelkom brezžičnega in prenosnega spletnega strežnika, ki ga je mogoče celo spraviti v žep.

Hipoteze

H1: Z Raspberry Pi-jem je mogoče uspešno gostovati spletno aplikacijo.

H2: Znanje, pridobljeno na računalniški srednji šoli, je zadostno za izdelavo lastnega spletnega strežnika

H3: Strežniki z uporabo Raspberry Pi-ja so zaradi energetske in finančne učinkovitosti primernejši za domače uporabnike.

Ključne besede

Raspberry Pi, spletni strežnik, mikroračunalnik, Raspberry Pi OS, strežniški sistem, Apache.

Abstract

There exists the idea that an entire webpage can be hosted on a device the size of half a smartphone. This is completely foreign to most, and many might not even believe it to be possible.

In this research paper we research and realise this concept, creating a web server and hosting a website using the Raspberry Pi microcomputer. We theoretically and empirically research the different types of web servers, finding the most suitable solutions for each type of end user. We wanted to find out if using a Raspberry Pi can save money and electricity, both short- and long-term. We tested the most popular web server systems, web applications, and remote access programs. The paper is concluded with the creation of a wireless and portable web server that can easily be put in one's pocket.

Hypotheses

H1: It is possible to host a webpage using the Raspberry Pi microcomputer.

H2: Knowledge from a Computer Science high school is enough for the creation of one's own web server.

H3: Raspberry Pi web servers are suitable for home users because of their energetic efficiency and inexpensiveness.

Keywords

Raspberry Pi, web server, microcomputer, Raspberry Pi OS, server system, Apache.

Kazalo vsebine

1	UVOD.....	1
	<i>Cilji raziskovalne naloge</i>	<i>1</i>
	<i>Metodologija</i>	<i>2</i>
2	PRIMERJAVA STREŽNIŠKIH SISTEMOV	3
3	RASPBERRY PI	4
3.1	<i>Operacijski sistem</i>	<i>4</i>
3.2	<i>Spletni strežnik</i>	<i>4</i>
4	NAMESTITEV SPLETNEGA STREŽNIKA.....	5
4.1	<i>Strojna oprema.....</i>	<i>5</i>
4.2	<i>Programska oprema</i>	<i>7</i>
	<i>Ustvaritev podatkovnih baz</i>	<i>11</i>
	<i>Programi za oddaljen dostop.....</i>	<i>15</i>
	<i>Vzpostavitev na javni domeni.....</i>	<i>15</i>
4.3	<i>Spletna stran.....</i>	<i>17</i>
4.4	<i>Odprtje vrat.....</i>	<i>22</i>
4.5	<i>Končni izgled sistema.....</i>	<i>23</i>
	<i>Brezžičen spletni strežnik</i>	<i>23</i>
5	ZAKLJUČEK.....	25
	<i>Analiza hipotez.....</i>	<i>25</i>
	<i>Viri in literatura.....</i>	<i>27</i>

Kazalo slik

Slika 1: Raspberry Pi Imager	6
Slika 2: Raspberry Pi OS	7
Slika 3: Apache privzeta stran	8
Slika 4: sudo apt-get.....	9
Slika 5: Delovanje spletne strani na lokalnem IP naslovu	9
Slika 6: Pridobitev IP naslova za dostop.....	10
Slika 7: Dostop do strani iz glavnega računalnika.....	10
Slika 8: Ustvaritev uporabniškega računa za podatkovno bazo.....	11
Slika 9: VIM	12
Slika 10: Ponovni zagon Apacheja	13
Slika 11: PHPMyAdmin začetna stran	13
Slika 12: Delovanje PHPMyAdmina	14
Slika 13: TeamViewer	15
Slika 14: Ngrok #1	16
Slika 15: Ngrok #2.....	16
Slika 16: Prenos WordPressa	17
Slika 17: Ekstrakcija datotek, dodeljvanje pravic uporabniku.....	18
Slika 18: Nastavitev vpisnih podatkov za podatkovno bazo	18
Slika 19: Ustvaritev podatkovne baze za WordPress	19
Slika 20: Ustvaritev uporabniškega računa za WordPress podatkovno bazo	19
Slika 21: Dodeljevanje pravic uporabniku WordPress podatkovne baze	20
Slika 22: Prvi pogled na uspešno namestitev WordPressa	20
Slika 23: Dokončno nameščanje WordPressa #1.....	21
Slika 24: Dokončno nameščanje WordPressa #2.....	21
Slika 25: Dokončno nameščanje WordPressa #3.....	21
Slika 26: Odprtje vrat.....	22
Slika 27: Dostopanje do strani preko vrat.....	22
Slika 28: Končni izgled sistema.....	23
Slika 29: Brezžičen spletni strežnik #1	24
Slika 30: Brezžičen spletni strežnik #2.....	24

1 UVOD

Preko vsesplošnega pomanjševanja tehnologije in razvoja mikroprocesorjev v današnji dobi smo prišli od računalnikov, velikih tri sobe, vse do celotnih računalniških sistemov, ki jih lahko spravimo v žep. Raspberry Pi je uresničitev ravno tega koncepta. Je točno to – zelo majhen računalnik z zmogljivostjo namiznega računalnika. Lahko ga uporabimo za vse, za kar uporabljamo namizne računalnike, njegova velikost pa omogoča še obilico drugih rab: danes je uporaba razširjena v vremenoslovju, za pametne hiše in tudi na splošno med računalniškimi navdušenci.

Za raziskovalno nalogo sva se odločila uporabiti ravno ta mikroračunalnik, in sicer je bil namen raziskati gostovanje spletnih aplikacij preko njega.

Cilji raziskovalne naloge

Osrednji cilj je ustvariti spletni strežnik z uporabo Raspberry Pi-ja, hkrati pa s pomočjo empirične raziskave prikazati energetske in finančne učinkovitosti sistema v primerjavi z alternativami. Raziskati namreč želimo, ali je smiselno namesto namenske strežniške strojne opreme oz. osebnega računalnika uporabljati Raspberry Pi kot domač ali pa celo profesionalen strežnik.

Natančnejše so cilji sledeči.

- Teoretični cilji:
 - predstaviti mikroračunalnik Raspberry Pi in njegove funkcije,
 - opisati je potrebno programsko opremo in proces namestitve le-te,
 - naštet različne vrste strežnikov, vrste uporab in skupine ljudi, ki jih uporabljajo,
 - raziskati, katera vrsta strežnika je teoretično najboljša za katero skupino ljudi.
- Praktični cilji:
 - ustvariti lasten spletni strežnik na osnovi Raspberry Pi-ja,
 - določiti, katera vrsta strežnika je praktično najboljša za katero skupino ljudi,
 - ugotoviti energetske učinkovitosti ter identificirati finančne breme mikroračunalnika,
 - praktično primerjati strežnik na mikroračunalniku z ostalimi vrstami,
 - z empiričnimi podatki odgovoriti na hipotezo.

Metodologija

Z nalogo sva začela s teoretičnim načrtovanjem, kaj bova delala in kako se bova tega lotila.

Najprej sva poiskala ustrezne spletne ter fizične vire, ki vsebujejo najino tematiko, na osnovi teh pa si postavila cilje, hipoteze ter načrt dela.

2 PRIMERJAVA STREŽNIŠKIH SISTEMOV

Poznamo tri glavne vrste strežniških sistemov. To so namenski strežnik, uporaba osebnega računalnika ter mikroračunalniški sistemi (v tej nalogi Raspberry Pi). Različne vrste sistemov so primerne za različne skupine uporabnikov, odvisno od njihovih potreb po procesni moči, energetski učinkovitosti ter finančni ugodnosti.

Namenski strežniki so specializirana strojna oprema, namenjena gostovanju aplikacij na omrežjih. Njihova specializiranost omogoča visoke hitrosti procesiranja in prenosa podatkov, a so razmeroma dragi za končne uporabnike, fizično veliki, porabljajo veliko elektrike, poleg tega pa so lahko zaradi zgradbe tudi glasni. Za osebo, ki bi želela gostovati preprostejše programe, predstavljajo prej omenjene lastnosti veliko oviro. V uporabi so predvsem v večjih podjetjih, ki potrebujejo veliko procesno moč.

Kot alternativo namenskih strežnikov lahko uporabimo za gostovanje tudi lasten osebni računalnik. Je ravno tako hiter kot uporabljen računalnik, v večini primerov zastoj in ni potrebno kupovati dodatne opreme. Problem uporabe osebnega računalnika je, da nam gostovanje trati procesno moč, ki jo potrebuje preostanek računalnika. Poleg tega je programska oprema osebnih računalnikov takšna, da ne moremo imeti nenehno vklopljenega sistema – to bi namreč prenasičilo napravo.

Zaradi teh razlogov je glavna rešitev, ki sva jo v nalogi raziskala, uporaba Raspberry Pi-ja. Ta je, vsaj na prvi pogled, relativno poceni, hitrost je sicer variabilna, a je mogoče sistem optimizirati za želeno delovanje, sploh pa je dokaj preprost za uporabo in ga je možno spremeniti v skoraj že namenski sistem.

3 RASPBERRY PI

Raspberry Pi je serija majhnih računalnikov, ki jih proizvaja Raspberry Pi Foundation v Združenem kraljestvu. Prvotno so bili zasnovani za učne namene, a se danes široko uporabljajo na številnih področjih, na primer v robotiki, vremenoslovju ter v pametnih domovih.

3.1 Operacijski sistem

Za potrebe naloge sva na sistem naložila operacijski sistem Raspberry Pi OS (včasih znan kot Raspbian). To je standarden OS za Raspberry Pi in je zanj tudi optimiziran, poleg tega pa je preprost za uporabo ter ima preprost uporabniški vmesnik. Ravno zaradi teh razlogov sva ga namestila na najin sistem.

Temelji na Linux operacijskem sistemu Debian. To pomeni, da je, za razliko od ostalih operacijskih sistemov npr. Windowsa, popolnoma odprtokoden in ima vsakdo dostop do njegove izvirne kode. Podpira veliko različnih jezikov, med drugimi tudi slovenščino. Uradno so ga leta 2013 izdali Raspberry Pi Foundation za potrebe njihove strojne opreme.

3.2 Spletni strežnik

Za gostovanje strani ter ostalih spletnih aplikacij sva potrebovala še spletni strežnik. Izbrala sva *Apache2*, saj nama je poznan že od prejšnjih projektov ter ni prezapleten za uporabo. Prav tako je to najbolj uporabljen HTTP strežnik na svetu, uporablja ga 48 % vseh spletnih strani.

Poleg tega sva za hranjenje podatkov in registracijo oz. prijavo na stran potrebovala podatkovno bazo. Za izdelavo le-te sva uporabila SQL ter za preglednost PHPMYAdmin. Torej namestiti sva morala SQL ter PHP ter še posebej PHPMYAdmin.

4 NAMESTITEV SPLETNEGA STREŽNIKA

Za izvedbo raziskovalne naloge sva uporabila Raspberry Pi 4 – najnovejšo različico z 8 GB pomnilnika. Potrebovala sva namreč zmogljiv sistem, ki bi bil zmožen z lahkoto gostiti spletne aplikacije.

Procesor, ki ga ima najina serija Raspberry Pi je štirijedrni, 64-bitni ter deluje na frekvenci 1,5 GHz, se imenuje *Broadcom BCM2711, Quad core Cortex-A72 64bit – SoC @ 1.5Ghz*.

Pomnilni prostor je v najini različici v velikosti 8 GB in deluje na delovni frekvenci 3200 MHz. Model se imenuje *LPDDR4-3200 SDRAM*. Obstajajo tudi druge različice z 2 GB in 4 GB pomnilne prostora. Kar se tiče povezav ima omrežno kartico, ki podpira standard IEEE 802.11ac na 2.4 GHz ter 5.0 GHz. Prav tako podpira tudi Bluetooth 5.0.

Raspberry Pi ima dva mirco HDMI vhoda, ki podpirata ločljivost do 3840x216060. Na voljo ima tudi AUX vhod za zvok. Za shranjevanje podatkov ima rezo za micro SD kartico. Na voljo so tudi štirje USB vhodi, med njimi sta dva USB 2.0, druga dva pa USB 3.0.

Napajanje Raspberry Pi se izvaja z minimalnim tokom 3 amperov preko 5-voltnega USB-C ali priključka ali GPIO.

4.1 Strojna oprema

Najprej sva morala izvesti inventuro stvari, ki jih že imava, in si zapisati še potrebno opremo. V škatli s samo napravo sva dobila ravno in izključno to – samo napravo in napravo kot samo. Ugotovila sva, da se ta polni preko USB-C priključka, za upravljanje s sistemom sva potrebovala še miško in tipkovnico z USB priključkom. Potrebno je tudi hranjenje podatkov, torej sva potrebovala pomnilni medij. Ker Raspberry Pi deluje z micro SD karticami, je bila najboljša rešitev uporaba micro SD kartice.

Za konec je pomemben še prikaz podatkov, da lahko opazujemo dogajanje. To je mogoče preko micro HDMI izhoda. K sreči sva vse potrebno, razen enega kabla, imela doma, torej za večino stvari ni bilo treba hoditi po trgovinah. Edina komponenta, ki je nisva imela, je bil kabel z micro HDMI priključkom na eni in navadnim HDMI priključkom na drugi strani, ki sva ga morala kupiti.

Po vsem tem sva končno lahko začela z glavnim postopkom. Micro SD kartico sva dala v osebni računalnik in nanjo s pomočjo programa *Raspberry Pi Imager* naložila izbran operacijski sistem – Raspberry Pi OS. O njem bolj podrobno govoriva v poglavju 3.1.



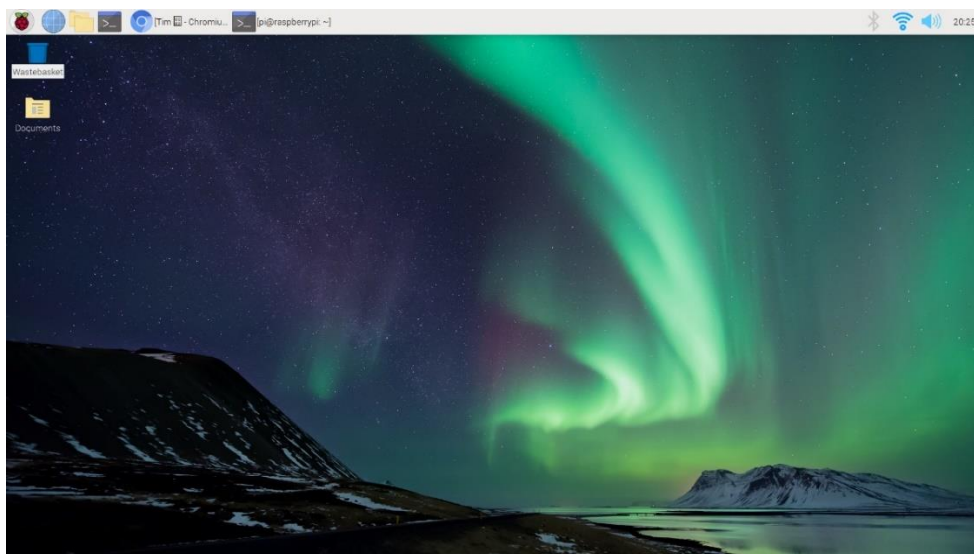
Slika 1: *Raspberry Pi Imager*

Proces je popolnoma avtomatiziran, morala sva le izbrati pravilni inštalacijski medij – micro SD kartico in pritisniti na gumb z napisom »ZAPIŠI«.

Po končani namestitvi operacijskega sistema sva kartico izstavila iz računalnika ter jo premestila v Raspberry Pi. Vanj sva priključila tudi potrebno periferijo in monitor, kar zaključuje strojni del namestitve. Sledilo je le še programsko delo.

4.2 Programska oprema

Po uspešnem zagonu naprave naju je pozdravil zaslon uspeha. Operacijski sistem je bil naložen in začelo resnično raziskovanje. Doslej je bil postopek lahek, toda sedaj je začelo postajati vse bolj zapleteno.



Slika 2: Raspberry Pi OS

Začela sva s tem, da sva skušala kar takoj namestiti opremo za spletni strežnik, a to ni bilo tako preprosto. Najprej sva šla skozi samodejni program dobrodošlice, ki nastavi jezik, tipkovnico, geslo in dostop do spleta. Za slednje sva se odločila uporabiti možnost WiFi-ja, tj. brezžične povezave. Lahko bi uporabila ožičeno povezavo, a je bilo ugodneje iti kar po brezžični poti.

Po vsem tem sva morala posodobiti programsko opremo, kar se je prav tako izvedlo samodejno (z uporabniško potrditvijo). Za vsak slučaj sva v terminalu še izvedla komandi `sudo apt-get update` in `sudo apt-get upgrade`, ki sta standardni Linux komandi za osvežitev programske opreme. Po ponovnem zagonu sva lahko pričela z delom.

Za gostovanje spletne strani sva izbrala Apache spletni strežnik. Apache igra ključno vlogo pri širjenju spleta. Predstavlja 44 % vseh spletnih strežnikov na svetu. Izbirala sva med Apache spletnim strežnikom in Nginx, ki je hitrejši pri določenih opravilih. Ampak sva se vseeno odločila za Apache, ker nama je bolj poznan že iz preteklih projektov.

Preden sva pričela z nameščanjem Apache serverja, sva morala zagotoviti, da je seznamov paketov posodobljen. To sva opravila z zagonom ukazov `sudo apt-get update` in `sudo apt-get upgrade`.

Nato sva nadaljevala z namestitvijo serverja. V konzolo sva vpisala ukaz za namestitev Apache spletnega strežnika, to je *sudo apt install apache2 -y*.

Ko se je namestitev uspešno zaključila, sva preverila delovanje strežnika. Zato sva morala najprej preveriti kakšen IP naslov ima najin Raspberry Pi in ga kasneje vnesla v spletni brskalnik. Ukaz za ugotovitev IP naslova je *hostname -I*.

Vnesla sva IP naslov v brskalnik in spletni strežnik je deloval. Prikazala se nama je privzeta stran za Apache, kot je razvidno na sliki spodaj.



 **Apache2 Debian Default Page**

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented** in `/usr/share/doc/apache2/README.Debian.gz`. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
|
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective `*-available/` counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2ensite`, `a2dissite`, and `a2enconf`, `a2disconf`. See their respective man pages for detailed information.
- The binary is called `apache2`. Due to the use of environment variables, in the default configuration, `apache2` needs to be started/stopped with `/etc/init.d/apache2` or `apache2ctl`. **Calling `/usr/bin/apache2` directly will not work** with the default configuration.

Document Roots

By default, Debian does not allow access through the web browser to *any* file apart of those located in `/var/www`, **public_html** directories (when enabled) and `/usr/share` (for web applications). If your site is using a web document root located elsewhere (such as in `/srv`) you may need to whitelist your document root directory in `/etc/apache2/apache2.conf`.

The default Debian document root is `/var/www/html`. You can make your own virtual hosts under `/var/www`. This is different to previous releases which provides better security out of the box.

Reporting Problems

Please use the `reportbug` tool to report bugs in the Apache2 package with Debian. However, check **existing bug reports** before reporting a new bug.

Please report bugs specific to modules (such as PHP and others) to respective packages, not to the web server itself.

Slika 3: Apache privzeta stran

```
pi@raspberrypi: ~
File Edit Tabs Help

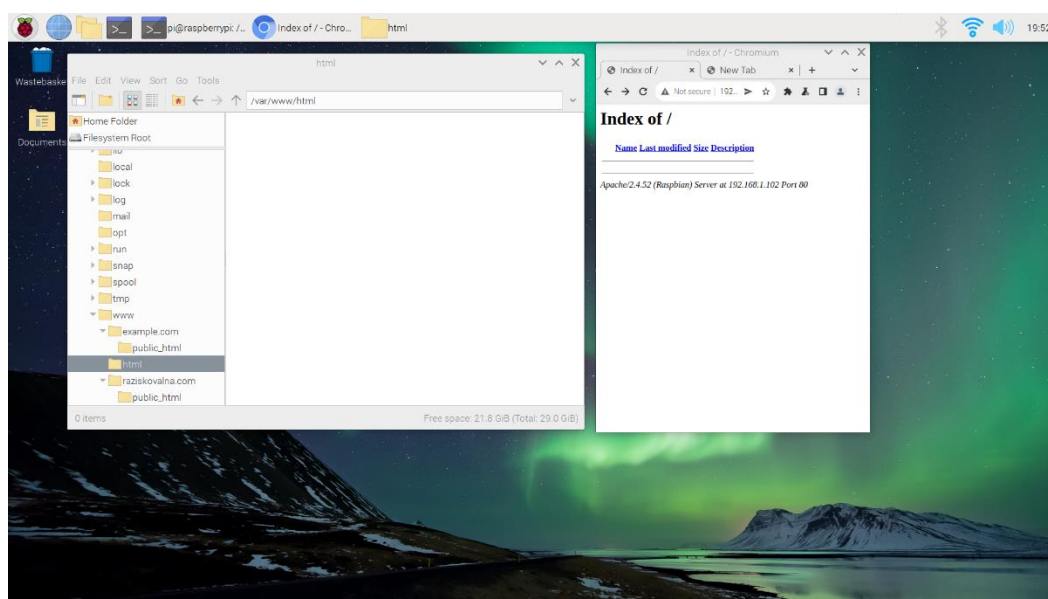
//rasbian.raspberrypi.org/raspbian bullseye InRelease' doesn't support architecture 'amd64'
N: Skipping acquire of configured file 'rpi/binary-amd64/Packages' as repository 'http://ras
pbian.raspberrypi.org/raspbian bullseye InRelease' doesn't support architecture 'amd64'
pi@raspberrypi:~$ sudo apt-get upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following package was automatically installed and is no longer required:
  libfuse2
Use 'sudo apt autoremove' to remove it.
The following packages will be upgraded:
  agnostics arandr chromium-browser chromium-browser-l10n chromium-codecs-ffmpeg-extra
  gldriver-test gstreamer1.0-plugins-good libapache2-mod-php7.4 libcamera-apps
  libcamera-tools libcamera0 libcryptsetup12 libexpat1 libexpat1-dev
  libjavascryptcoregtk-4.0-18 libmutter-7-0 libpixmap-1-0 libssl2-2 libssl2-modules-db
  libsmclient libssl1.1 libwbclient0 libwebkit2gtk-4.0-37 linux-libc-dev lxinput lxpanel
  lxpanel-data lxplug-bluetooth lxplug-updater mutter mutter-common openssl php7.4
  php7.4-bz2 php7.4-cli php7.4-common php7.4-curl php7.4-gd php7.4-json php7.4-mbstring
  php7.4-mysql php7.4-opcache php7.4-readline php7.4-xml php7.4-zip pi-greeter piclone
  raspberrypi-bootloader raspberrypi-kernel raspberrypi-sys-mods raspberrypi-ui-mods
  raspi-config rc-gui rp-bookshelf rp-prefapps rpi-eeprom samba-ls snapd teamviewer
  vcdbg
60 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 322 MB of archives.
After this operation, 16.1 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Slika 4: *sudo apt-get*

Na Linux sistemih so določene namestitve nadvse preproste, pri dotični namestitvi Apacheja je bilo treba le uporabiti komando *sudo apt install apache2 -y*. S tem so se potrebni programi namestili, po ponovnem zagonu pa sva lahko že skoraj preizkusila, če stran deluje.

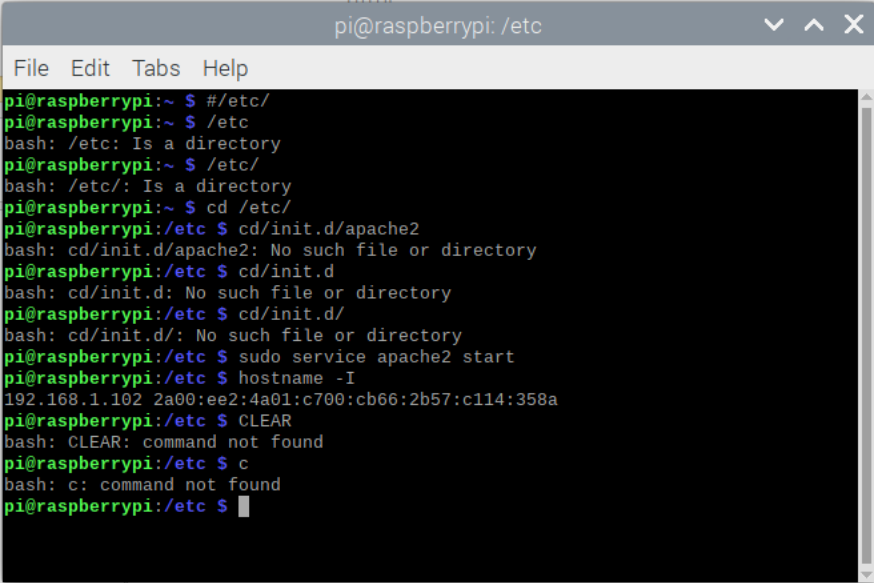
To še vendarle ni bilo vse, morala sva najprej ustvariti testno stran. Kot prvi test sva se odločila napisati v stran le »Zdravo, svet!« To sva ustvarila v datoteko »index.php« in jo prilepila v mapo */var/www/html*.

Po tem in ponovnem zagonu Apacheja je spletna stran delovala, kot bi morala.



Slika 5: Delovanje spletne strani na lokalnem IP naslovu

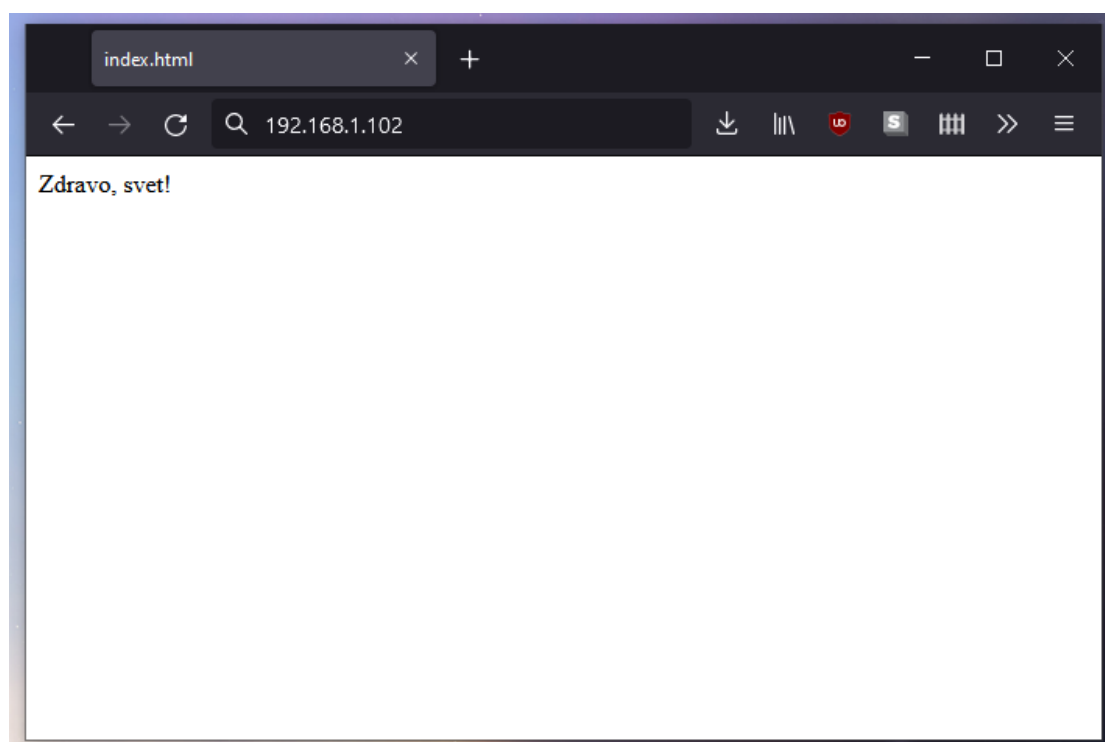
Sedaj sva se odločila, da morava urediti še dostop do strani preko drugih naprav. To še ni bilo mogoče z javno domeno, saj le-te še nisva imela ustvarjene. Do strani je bilo mogoče dostopati z javnim IP naslovom naprave, katerega sva dobila s preprosto komando *hostname -I*.



```
pi@raspberrypi: /etc
File Edit Tabs Help
pi@raspberrypi:~ $ #/etc/
pi@raspberrypi:~ $ /etc
bash: /etc: Is a directory
pi@raspberrypi:~ $ /etc/
bash: /etc/: Is a directory
pi@raspberrypi:~ $ cd /etc/
pi@raspberrypi:/etc $ cd/init.d/apache2
bash: cd/init.d/apache2: No such file or directory
pi@raspberrypi:/etc $ cd/init.d
bash: cd/init.d: No such file or directory
pi@raspberrypi:/etc $ cd/init.d/
bash: cd/init.d/: No such file or directory
pi@raspberrypi:/etc $ sudo service apache2 start
pi@raspberrypi:/etc $ hostname -I
192.168.1.102 2a00:ee2:4a01:c700:cb66:2b57:c114:358a
pi@raspberrypi:/etc $ CLEAR
bash: CLEAR: command not found
pi@raspberrypi:/etc $ c
bash: c: command not found
pi@raspberrypi:/etc $
```

Slika 6: Pridobitev IP naslova za dostop

Ta IP je v najinem primeru bil 192.168.1.102. Preizkusila sva dostopati do te domene preko glavnega računalnika, kar je delovalo in je vidno na sliki spodaj.



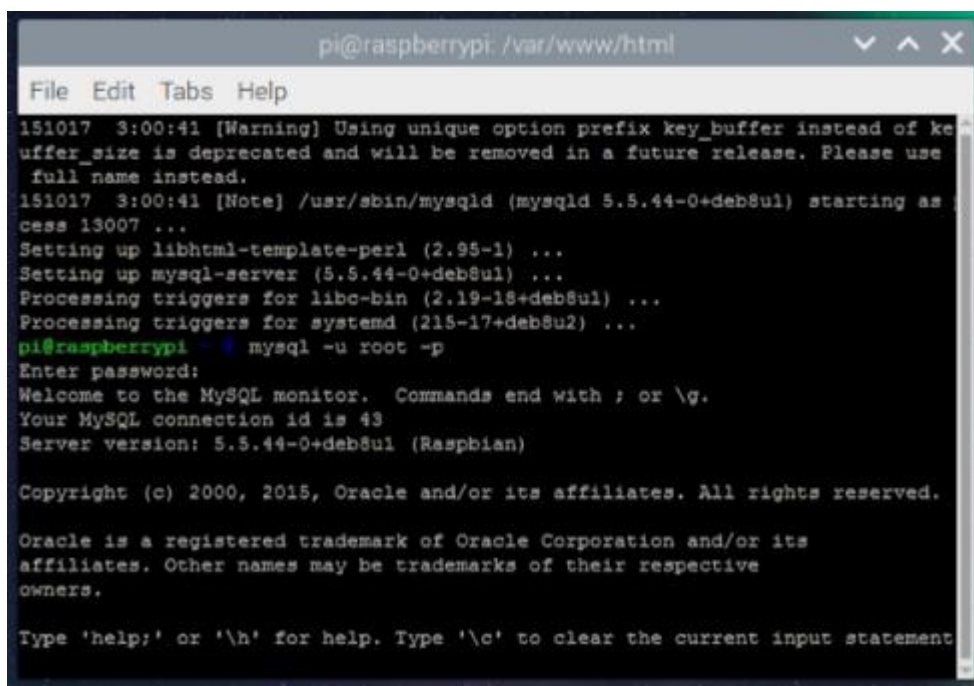
Slika 7: Dostop do strani iz glavnega računalnika

Ustvaritev podatkovnih baz

Za ustvaritev podatkovnih baz sva se odločila uporabiti PHPMyAdmin v programskem jeziku MySQL. To je orodje, ki omogoča lažje administriranje pri upravljanju podatkovne baze MySQL. Zaradi enostavnega upravljanja in preproste namestitve je priljubljena izbira med programerji in upravitelji podatkovnih baz. V kolikor nekdo želi, da bo PHPMyAdmin deloval, mora najprej imeti nameščen Apache spletni strežnik, kar sva midva že imela nastavljeno.

Orodje sva namestila z ukazom `sudo apt install phpmyadmin`. Ko se je le-ta uspešno namestil, sva ga še povezala z MySQL podatkovno bazo.

Da sva lahko prijavila vanj, sva morala ustvariti nov privzet uporabniški račun imenovan *root*, saj je po privzetih nastavitvah dostop onemogočen. To sva izvedla s komando `sudo mysql -u root -p`.



```
pi@raspberrypi: /var/www/html
File Edit Tabs Help
151017 3:00:41 [Warning] Using unique option prefix key_buffer instead of key
uffer_size is deprecated and will be removed in a future release. Please use
full name instead.
151017 3:00:41 [Note] /usr/sbin/mysqld (mysqld 5.5.44-0+deb8u1) starting as
cess 13007 ...
Setting up libhtml-template-perl (2.95-1) ...
Setting up mysql-server (5.5.44-0+deb8u1) ...
Processing triggers for libc-bin (2.19-18+deb8u1) ...
Processing triggers for systemd (215-17+deb8u2) ...
pi@raspberrypi ~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 43
Server version: 5.5.44-0+deb8u1 (Raspbian)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

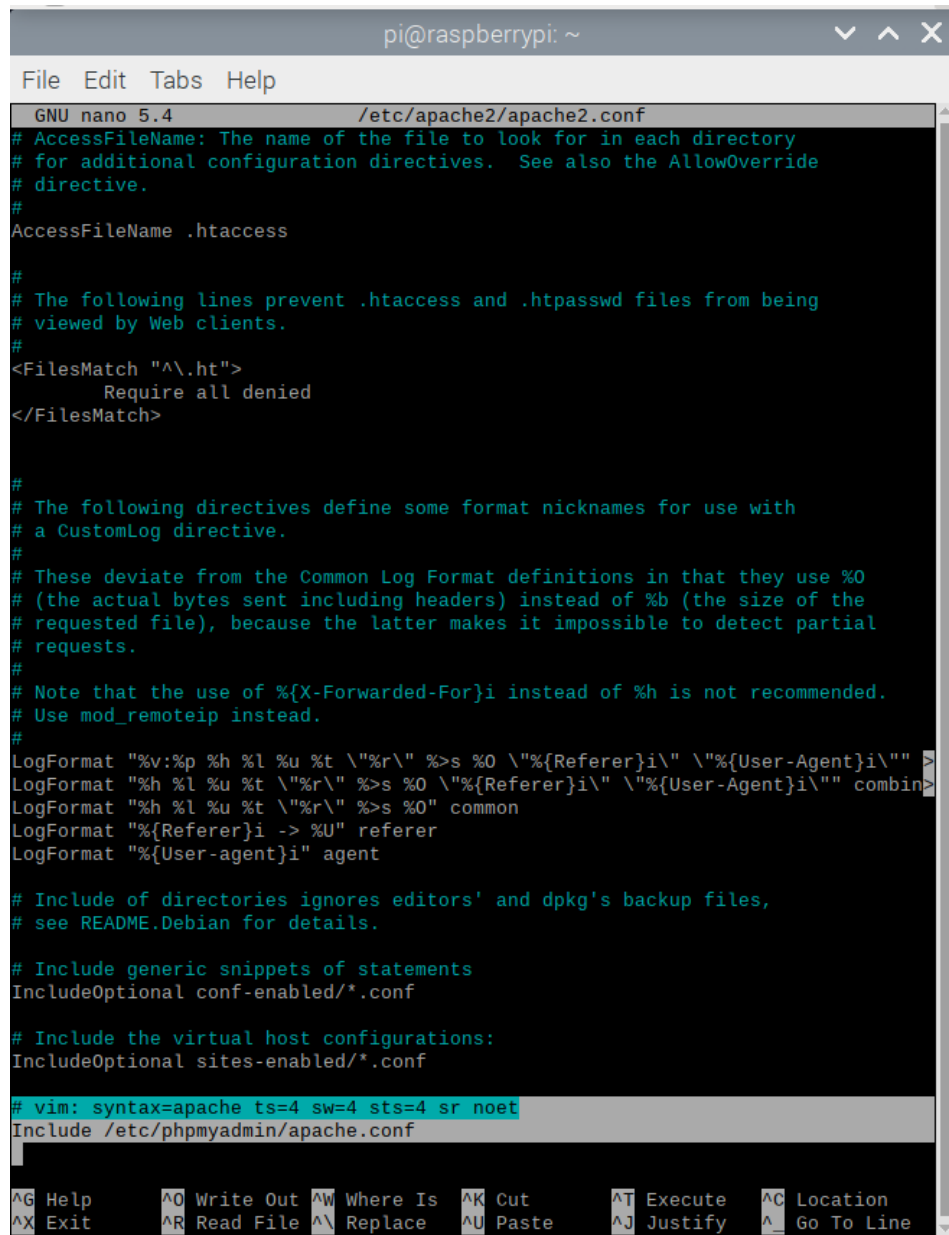
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement
```

Slika 8: Ustvaritev uporabniškega računa za podatkovno bazo

Tako sva prišla do MySQL vmesnika za ukazne pozive in z ukazom `GRANT ALL PRIVILEGES TO 'ime'@'localhost' IDENTIFIED BY 'geslo' WITH GRANT OPTION` ustvarila novega uporabnika za dostop.

Ker je to bilo vse, kar sva morala narediti v ukaznem pozivu za MySQL, sva s komando `quit` shranila in zapustila vmesnik.

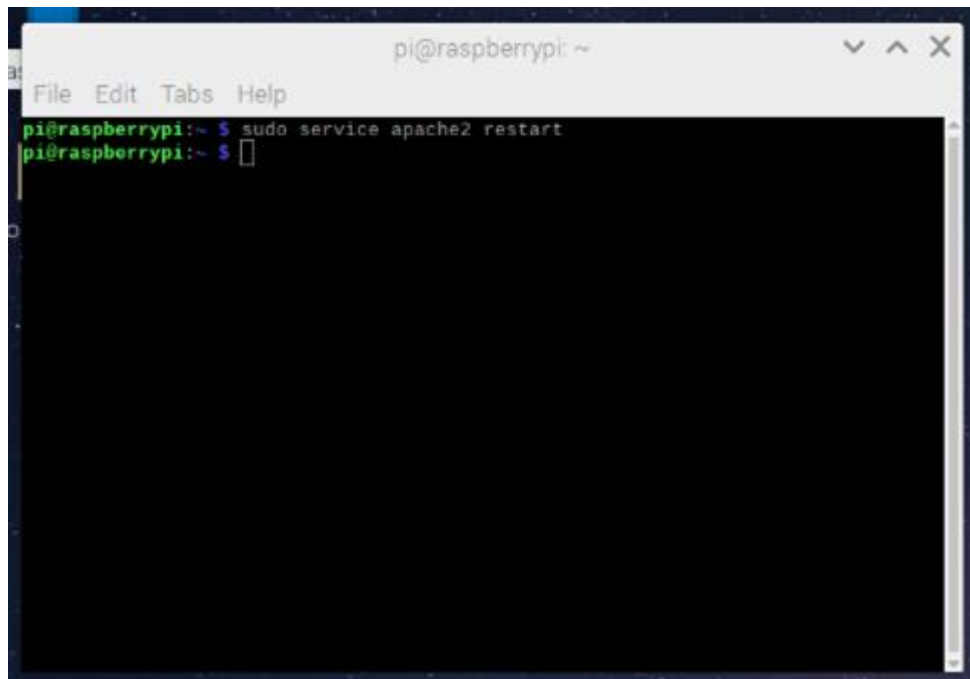
PHPMyAdmin je potrebno ročno prilagoditi, da lahko deluje s strežnikom Apache, kar sva tudi storila. Morala sva odpreti terminal za urejanje konfiguracije datoteke *apache2.conf*. Na njeno dno sva dodala še *Include /etc/phpmyadmin/apache.conf*. To je pomenilo, da je apache2, kar je novejša verzija starega apacheja, lahko normalno delovala na zasnovi starega in delujočega programa.



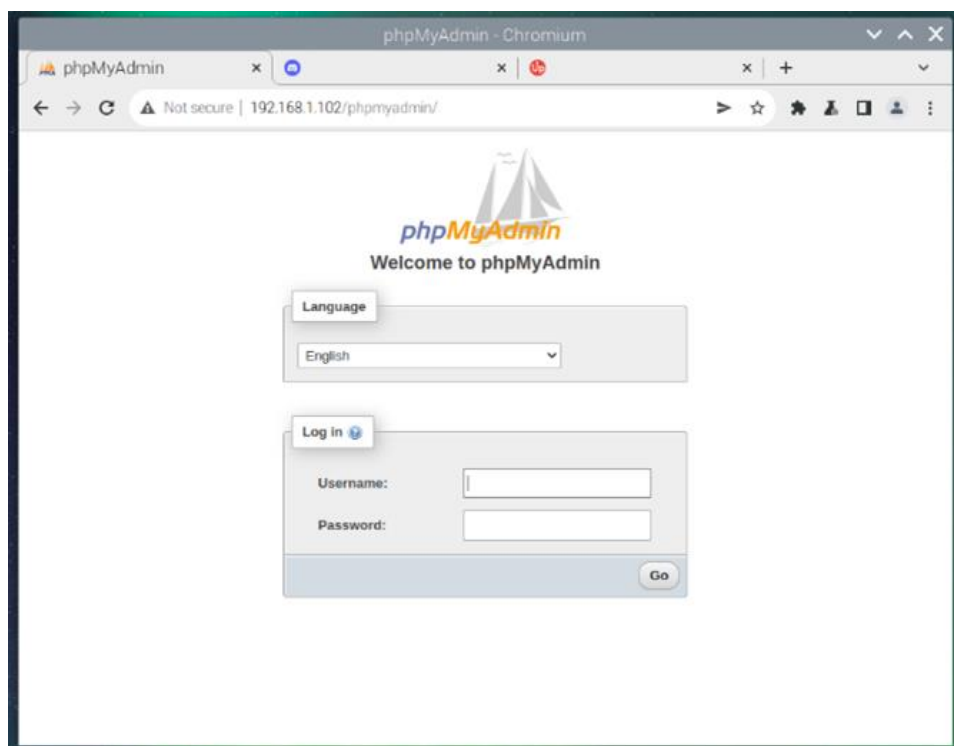
```
pi@raspberrypi: ~
File Edit Tabs Help
GNU nano 5.4 /etc/apache2/apache2.conf
# AccessFileName: The name of the file to look for in each directory
# for additional configuration directives.  See also the AllowOverride
# directive.
#
AccessFileName .htaccess
#
# The following lines prevent .htaccess and .htpasswd files from being
# viewed by Web clients.
#
<FilesMatch "\.ht">
    Require all denied
</FilesMatch>
#
# The following directives define some format nicknames for use with
# a CustomLog directive.
#
# These deviate from the Common Log Format definitions in that they use %O
# (the actual bytes sent including headers) instead of %b (the size of the
# requested file), because the latter makes it impossible to detect partial
# requests.
#
# Note that the use of %{X-Forwarded-For}i instead of %h is not recommended.
# Use mod_remoteip instead.
#
LogFormat "%v:%p %h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\""
LogFormat "%h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\" combin
LogFormat "%h %l %u %t \"%r\" %>s %O" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent
#
# Include of directories ignores editors' and dpkg's backup files,
# see README.Debian for details.
#
# Include generic snippets of statements
IncludeOptional conf-enabled/*.conf
#
# Include the virtual host configurations:
IncludeOptional sites-enabled/*.conf
# vim: syntax=apache ts=4 sw=4 ts=4 sr noet
Include /etc/phpmyadmin/apache.conf
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute  ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify  ^_ Go To Line
```

Slika 9: VIM

Ta sprememba dodaja PHPMyAdmin konfiguracijo in dovoljenje za prikazovanje preko Apache strežnika. Datoteko sva shranila in zaprla z hkratnim pritiskom tipk Ctrl ter X. Po vseh teh spremembah sva za pravilno delovanje morala ponovno zagnati Apache strežnik. To sva storila z ukazom *sudo service apache2 restart*.



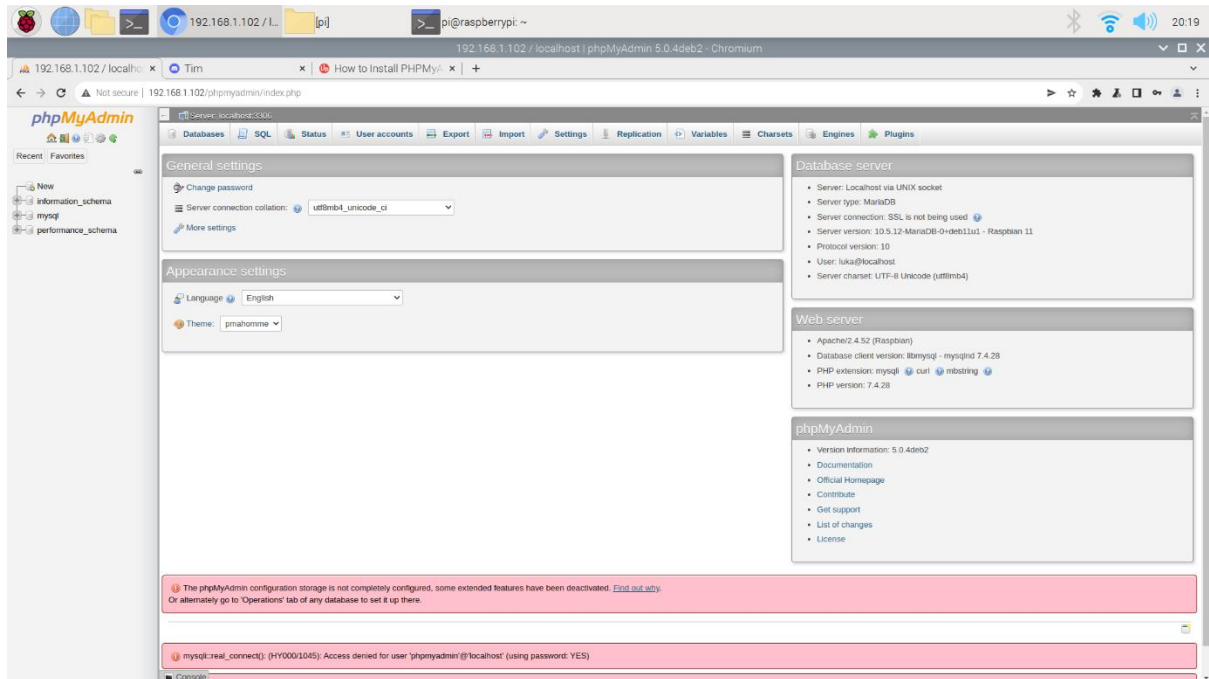
Slika 10: Ponovni zagon Apacheja



Slika 11: PHPMYAdmin začetna stran

Ko se je Apache strežnik ponovno zagnal, sva preizkusila delovanje PHPMYAdmina preko brskalnika. Kot povezavo do strani sva morala uporabiti IP naslov Raspberry PI-a. Da sva ga ponovno ugotovila sva izvedla ukaz v terminal.

S prej ustvarjenim uporabniškim računom sva se prijavila v orodje in kot pričakovano je delovalo.

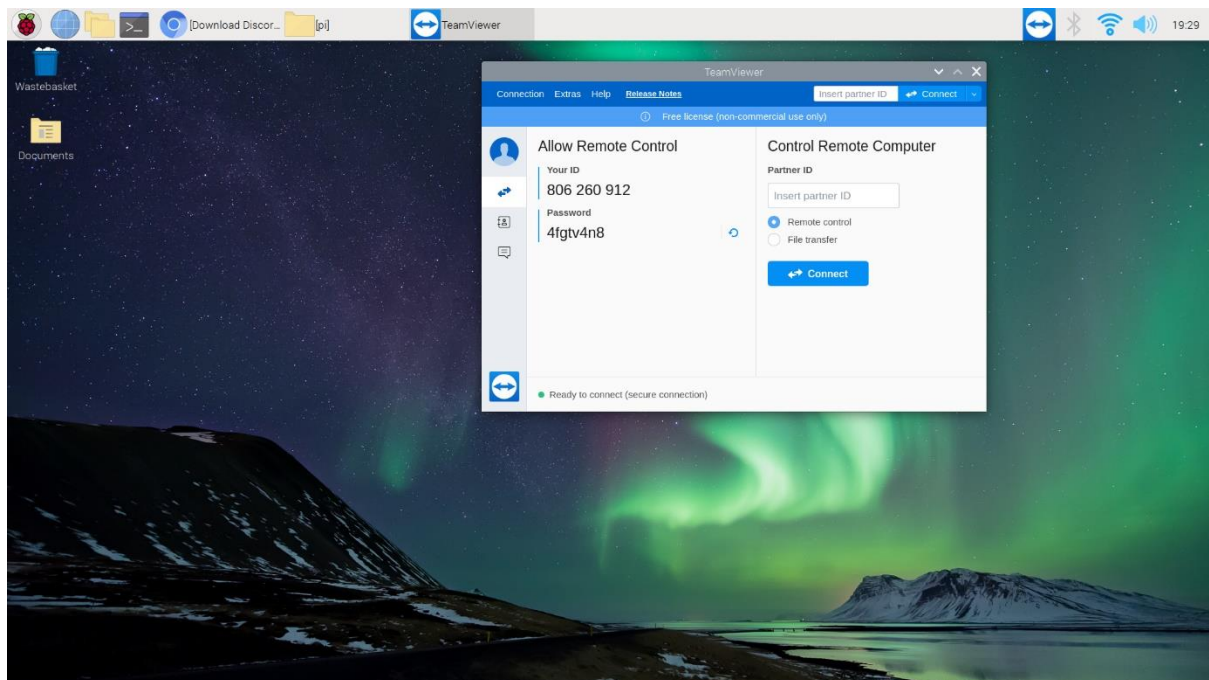


Slika 12: Delovanje PHPMYAdmina

Programi za oddaljen dostop

Nekje med ustvarjanjem sva se spomnila na obstoj programov oddaljenega dostopa. Pomislila sva, da bi bilo dobro imeti dostop do sistema tudi takrat, ko nisva fizično zraven naprave. Po dolgem in kompleksnem procesu ugotavljanja, kako se namešča programe na Linux, nama je uspelo nastaviti program TeamViewer.

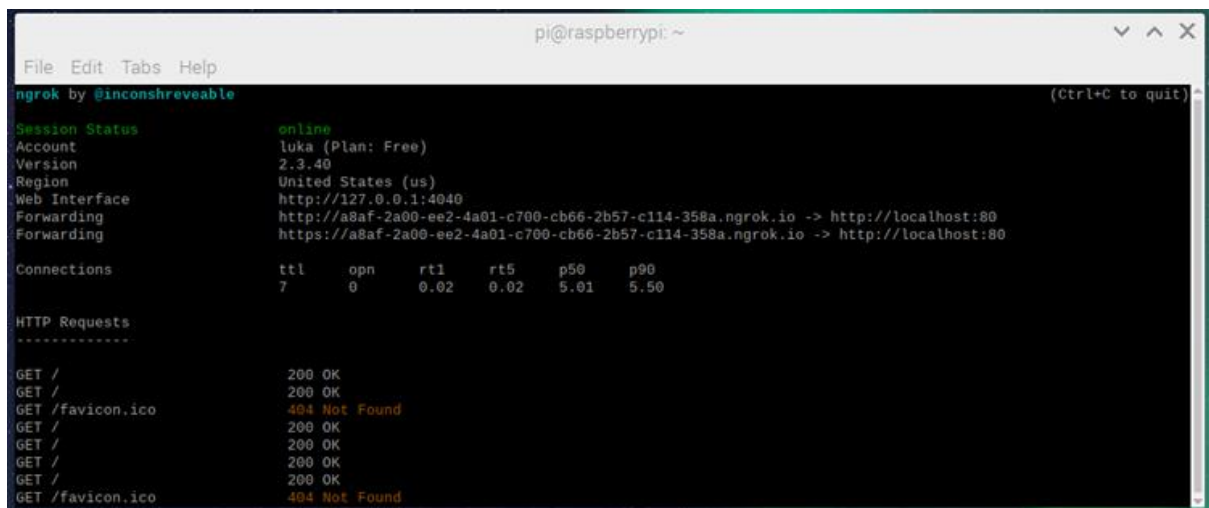
Žal sva kmalu zatem dojela, da je bil trud skoraj zaman, saj je vse skupaj delovalo zelo počasi, vse do točke, ko zaradi počasnosti sploh ni bilo smiselno programa uporabljati.



Slika 13: TeamViewer

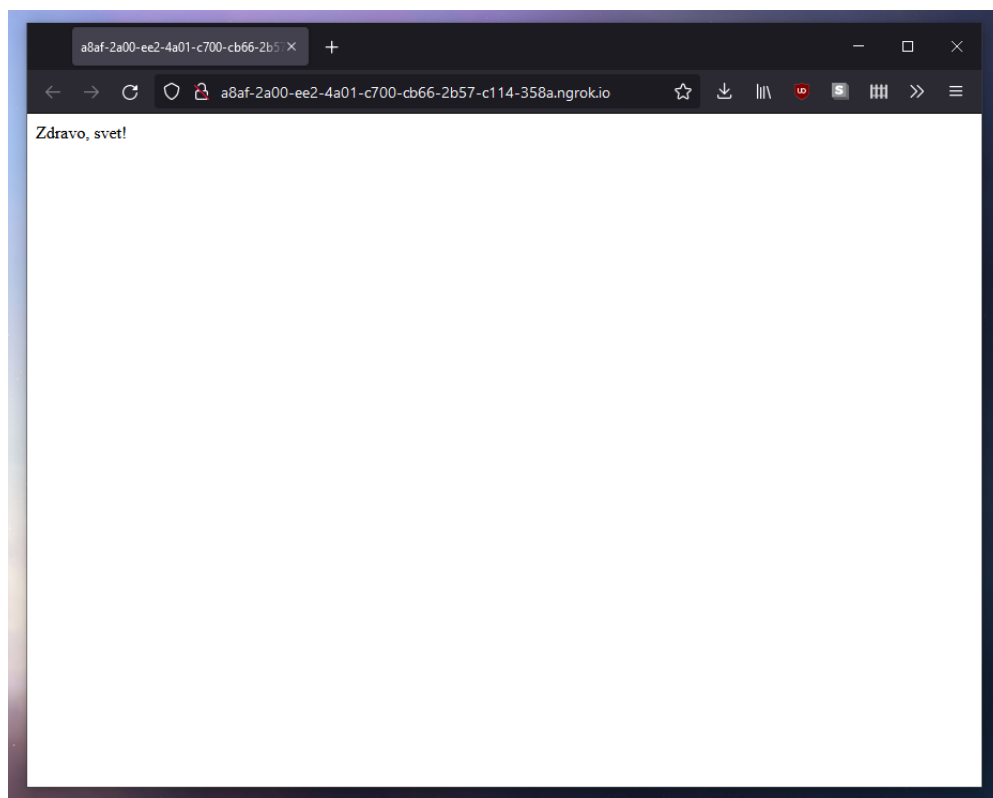
Vzpostavitev na javni domeni

Za izvedbo javne domene sva se odločila uporabiti ngrok. V svoji osnovi ngrok spremeni lokalno gostovanje (localhost) na vratih 80 preusmeri v zastonj javno domeno.



```
pi@raspberrypi: ~  
ngrok by @inconshreveable (Ctrl+C to quit)  
  
Session Status      online  
Account             luka (Plan: Free)  
Version             2.3.40  
Region              United States (us)  
Web Interface        http://127.0.0.1:4040  
Forwarding           http://a8af-2a00-ee2-4a01-c700-cb66-2b57-c114-358a.ngrok.io -> http://localhost:80  
Forwarding           https://a8af-2a00-ee2-4a01-c700-cb66-2b57-c114-358a.ngrok.io -> http://localhost:80  
  
Connections         ttl    opn    rt1    rt5    p50    p90  
                   7      0      0.02   0.02   5.01   5.50  
  
HTTP Requests  
-----  
GET /                200 OK  
GET /                200 OK  
GET /favicon.ico     404 Not Found  
GET /                200 OK  
GET /                200 OK  
GET /                200 OK  
GET /                200 OK  
GET /favicon.ico     404 Not Found
```

Slika 14: Ngrok #1



Slika 15: Ngrok #2

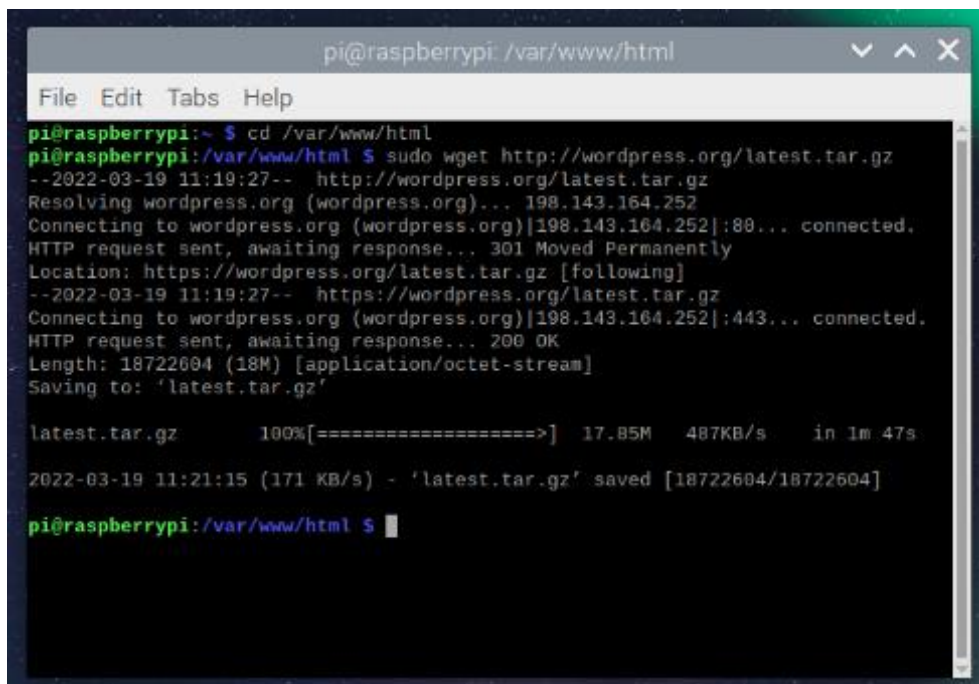
Spletna stran, zaenkrat še vedno preizkusna, je bila dostopna na javni domeni iz kjerkoli na svetu.

4.3 Spletna stran

Za potrebe spletne strani sva se odločila uporabiti WordPress, ki je zastoj in odprtokoden sistem za upravljanje z vsebino. Tako sva se odločila, saj spletna stran ni bistvo raziskovalne naloge in potrebujeva le nekaj preprostega z dobrim uporabniškim vmesnikom.

Ker WordPress deluje na osnovi Apache strežnika in MySQL podatkovne baze, sva morala najprej imeti to dvojico pravilno nastavljeno. (Ta postopek je opisan v delu 4.)

Začela sva s prenosom WordPress datotek v primerno datotečno mapo, in sicer v direktorij `/var/www/html`. To je mogoče v Linux terminalu opraviti z enovrstičnim ukazom.



```
pi@raspberrypi: /var/www/html
File Edit Tabs Help
pi@raspberrypi:~ $ cd /var/www/html
pi@raspberrypi:/var/www/html $ sudo wget http://wordpress.org/latest.tar.gz
--2022-03-19 11:19:27-- http://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)[198.143.164.252]:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://wordpress.org/latest.tar.gz [following]
--2022-03-19 11:19:27-- https://wordpress.org/latest.tar.gz
Connecting to wordpress.org (wordpress.org)[198.143.164.252]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 18722604 (18M) [application/octet-stream]
Saving to: 'latest.tar.gz'

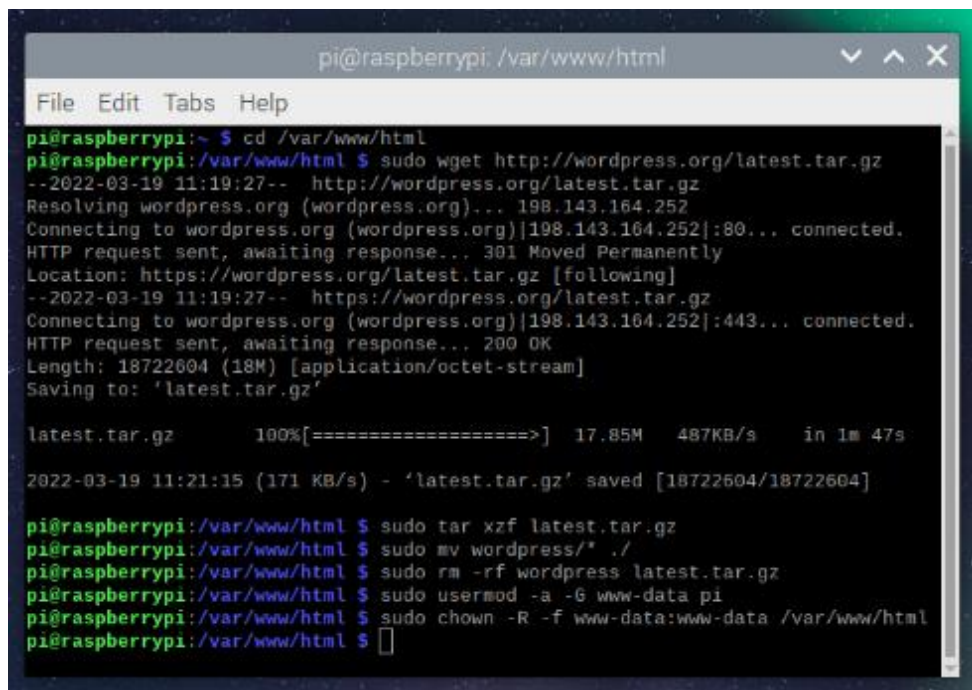
latest.tar.gz      100%[=====] 17.85M  487KB/s  in 1m 47s

2022-03-19 11:21:15 (171 KB/s) - 'latest.tar.gz' saved [18722604/18722604]

pi@raspberrypi:/var/www/html $
```

Slika 16: Prenos WordPressa

Te datoteke so v arhivirani (.zip) datoteki in so same po sebi neuporabne, torej sva jih morala ekstrahirati. Poleg tega sva morala dodeliti pravice uporabniku, da ne bi v nadaljnjem postopku prišlo do napake manjkajočih pravic. To je mogoče z ukazi, kar je razvidno v spodnji sliki.



```
pi@raspberrypi: /var/www/html
File Edit Tabs Help
pi@raspberrypi:~ $ cd /var/www/html
pi@raspberrypi:/var/www/html $ sudo wget http://wordpress.org/latest.tar.gz
--2022-03-19 11:19:27-- http://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)[198.143.164.252]:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://wordpress.org/latest.tar.gz [following]
--2022-03-19 11:19:27-- https://wordpress.org/latest.tar.gz
Connecting to wordpress.org (wordpress.org)[198.143.164.252]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 18722604 (18M) [application/octet-stream]
Saving to: 'latest.tar.gz'

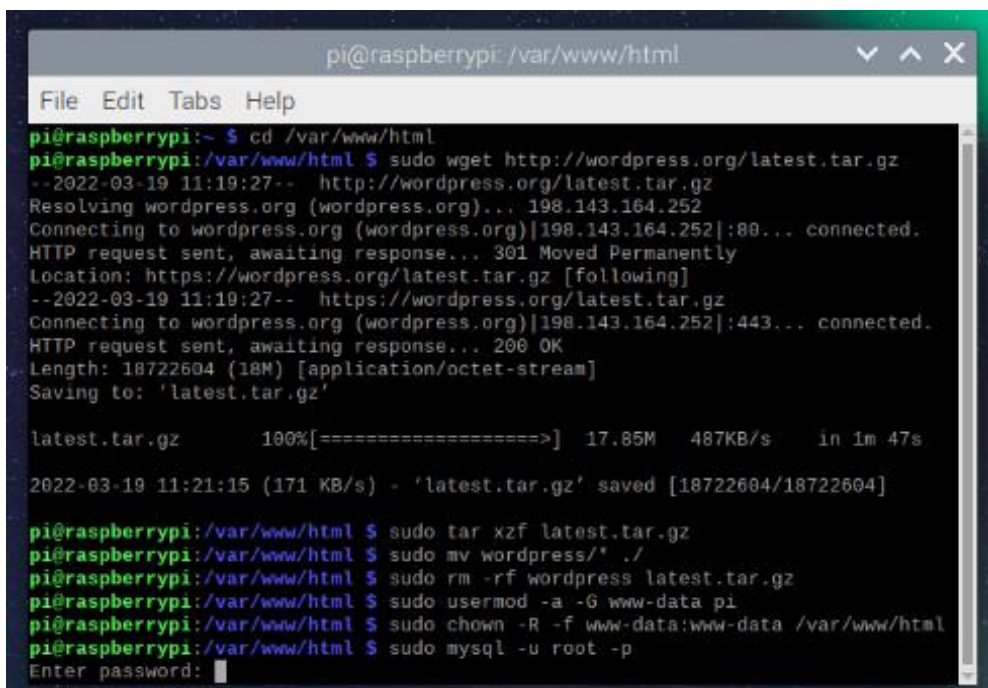
latest.tar.gz      100%[=====] 17.85M  487KB/s  in 1m 47s

2022-03-19 11:21:15 (171 KB/s) - 'latest.tar.gz' saved [18722604/18722604]

pi@raspberrypi:/var/www/html $ sudo tar xzf latest.tar.gz
pi@raspberrypi:/var/www/html $ sudo mv wordpress/* ./
pi@raspberrypi:/var/www/html $ sudo rm -rf wordpress latest.tar.gz
pi@raspberrypi:/var/www/html $ sudo usermod -a -G www-data pi
pi@raspberrypi:/var/www/html $ sudo chown -R -f www-data:www-data /var/www/html
pi@raspberrypi:/var/www/html $
```

Slika 17: Ekstrakcija datotek, dodeljvanje pravic uporabniku

Ko so bile vse potrebne datoteke na svojem mestu, sva morala še ustvariti podatkovno bazo. To se prav tako izvaja v terminalu z ukaznimi pozivi, in sicer s komando `mysql -u root -p`. Zatem je okno povprašalo po nastavitvi uporabniškega imena in gesla, kar sva tudi opravila.



```
pi@raspberrypi: /var/www/html
File Edit Tabs Help
pi@raspberrypi:~ $ cd /var/www/html
pi@raspberrypi:/var/www/html $ sudo wget http://wordpress.org/latest.tar.gz
--2022-03-19 11:19:27-- http://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)[198.143.164.252]:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://wordpress.org/latest.tar.gz [following]
--2022-03-19 11:19:27-- https://wordpress.org/latest.tar.gz
Connecting to wordpress.org (wordpress.org)[198.143.164.252]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 18722604 (18M) [application/octet-stream]
Saving to: 'latest.tar.gz'

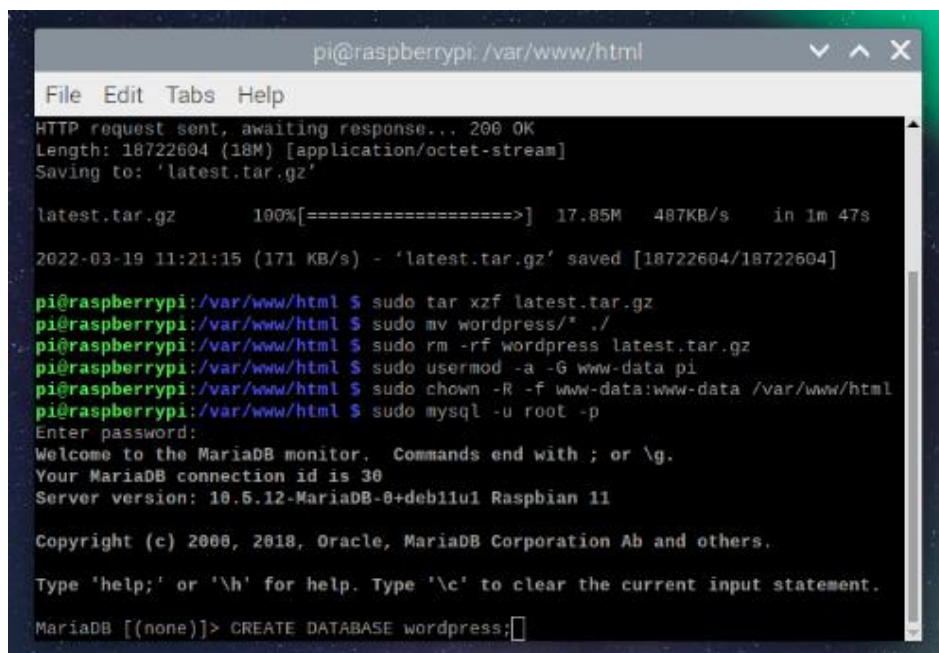
latest.tar.gz      100%[=====] 17.85M  487KB/s  in 1m 47s

2022-03-19 11:21:15 (171 KB/s) - 'latest.tar.gz' saved [18722604/18722604]

pi@raspberrypi:/var/www/html $ sudo tar xzf latest.tar.gz
pi@raspberrypi:/var/www/html $ sudo mv wordpress/* ./
pi@raspberrypi:/var/www/html $ sudo rm -rf wordpress latest.tar.gz
pi@raspberrypi:/var/www/html $ sudo usermod -a -G www-data pi
pi@raspberrypi:/var/www/html $ sudo chown -R -f www-data:www-data /var/www/html
pi@raspberrypi:/var/www/html $ sudo mysql -u root -p
Enter password: 
```

Slika 18: Nastavitev vpisnih podatkov za podatkovno bazo

Po uspešni nastavitvi vpisnih podatkov se je odprl program za dostop do podatkovnih baz. V njem sva takšno ustvaritev tudi izvedla s pozivom *CREATE DATABASE WordPress*.



```
pi@raspberrypi: /var/www/html
File Edit Tabs Help
HTTP request sent, awaiting response... 200 OK
Length: 18722604 (18M) [application/octet-stream]
Saving to: 'latest.tar.gz'

latest.tar.gz      100%[=====] 17.85M  487KB/s   in 1m 47s

2022-03-19 11:21:15 (171 KB/s) - 'latest.tar.gz' saved [18722604/18722604]

pi@raspberrypi:/var/www/html $ sudo tar xzf latest.tar.gz
pi@raspberrypi:/var/www/html $ sudo mv wordpress/* ./
pi@raspberrypi:/var/www/html $ sudo rm -rf wordpress latest.tar.gz
pi@raspberrypi:/var/www/html $ sudo usermod -a -G www-data pi
pi@raspberrypi:/var/www/html $ sudo chown -R -f www-data:www-data /var/www/html
pi@raspberrypi:/var/www/html $ sudo mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 30
Server version: 10.5.12-MariaDB-0+deb11u1 Raspbian 11

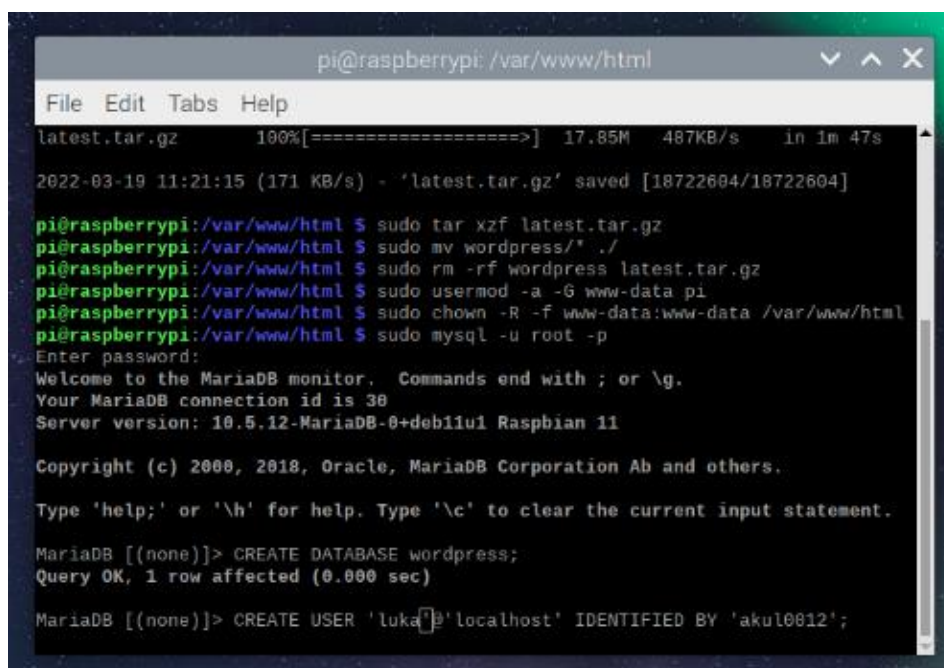
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE wordpress;
```

Slika 19: Ustvaritev podatkovne baze za WordPress

Z ustvarjeno podatkovno bazo sva morala ustvariti še uporabniški račun za dostop do nje. To sva izvedla z ukazom *CREATE USER 'ime'@'localhost' IDENTIFIED BY 'geslo';*.



```
pi@raspberrypi: /var/www/html
File Edit Tabs Help
latest.tar.gz      100%[=====] 17.85M  487KB/s   in 1m 47s

2022-03-19 11:21:15 (171 KB/s) - 'latest.tar.gz' saved [18722604/18722604]

pi@raspberrypi:/var/www/html $ sudo tar xzf latest.tar.gz
pi@raspberrypi:/var/www/html $ sudo mv wordpress/* ./
pi@raspberrypi:/var/www/html $ sudo rm -rf wordpress latest.tar.gz
pi@raspberrypi:/var/www/html $ sudo usermod -a -G www-data pi
pi@raspberrypi:/var/www/html $ sudo chown -R -f www-data:www-data /var/www/html
pi@raspberrypi:/var/www/html $ sudo mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 30
Server version: 10.5.12-MariaDB-0+deb11u1 Raspbian 11

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE wordpress;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> CREATE USER 'luka'@'localhost' IDENTIFIED BY 'akul0012';
```

Slika 20: Ustvaritev uporabniškega računa za WordPress podatkovno bazo

Novonastali uporabnik je bil brez kakršnih koli dovoljenj, torej sva morala to urediti. To je zopet bilo mogoče s preprosto komando `GRANT ALL ON wordpress.* TO 'ime'@'localhost';`

```
pi@raspberrypi: /var/www/html
```

File Edit Tabs Help

```
pi@raspberrypi:/var/www/html $ sudo usermod -a -G www-data pi
pi@raspberrypi:/var/www/html $ sudo chown -R -f www-data:www-data /var/www/html
pi@raspberrypi:/var/www/html $ sudo mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 30
Server version: 10.5.12-MariaDB-0+deb11u1 Raspbian 11

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE wordpress;
Query OK, 1 row affected (0.000 sec)

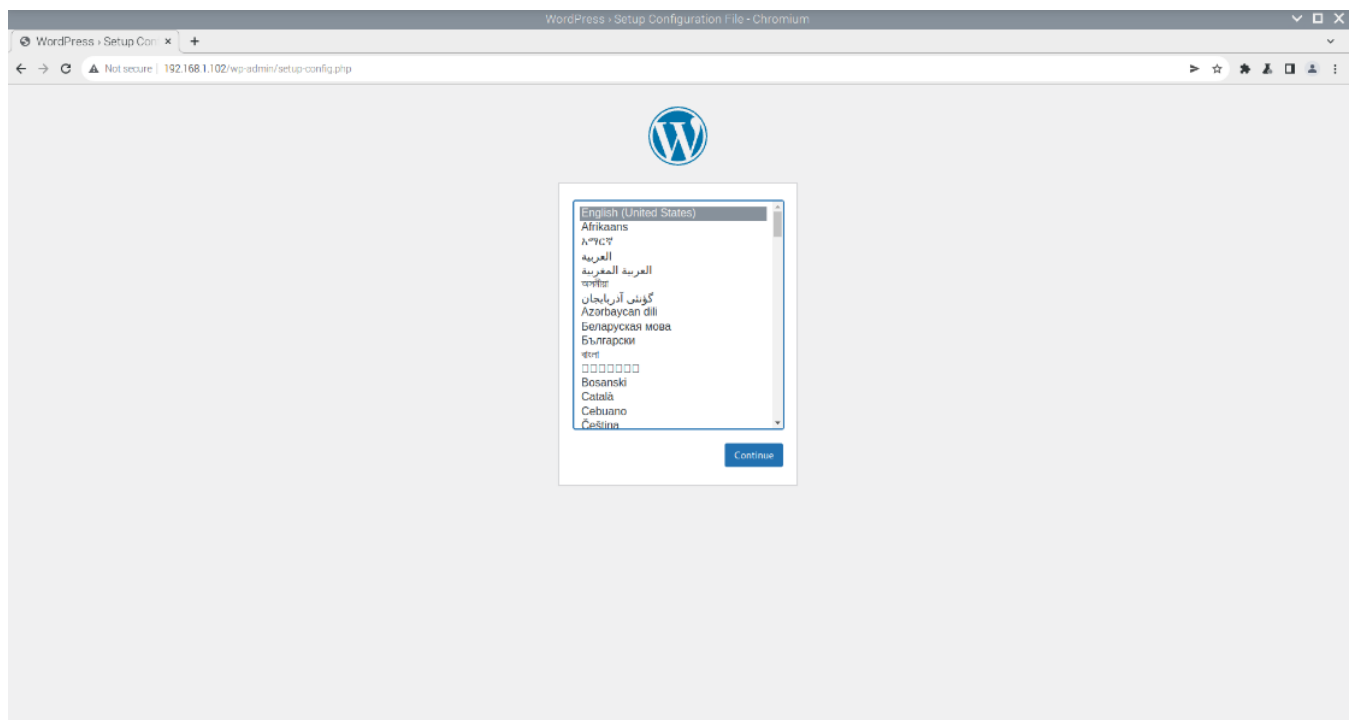
MariaDB [(none)]> CREATE USER 'luka'@'localhost' IDENTIFIED BY 'akul0012';
ERROR 1396 (HY000): Operation CREATE USER failed for 'luka'@'localhost'
MariaDB [(none)]> CREATE USER 'user'@'localhost' IDENTIFIED BY 'akul0012';
Query OK, 0 rows affected (0.022 sec)

MariaDB [(none)]> GRANT ALL ON wordpress.* TO 'user'@'localhost';
Query OK, 0 rows affected (0.043 sec)

MariaDB [(none)]>
```

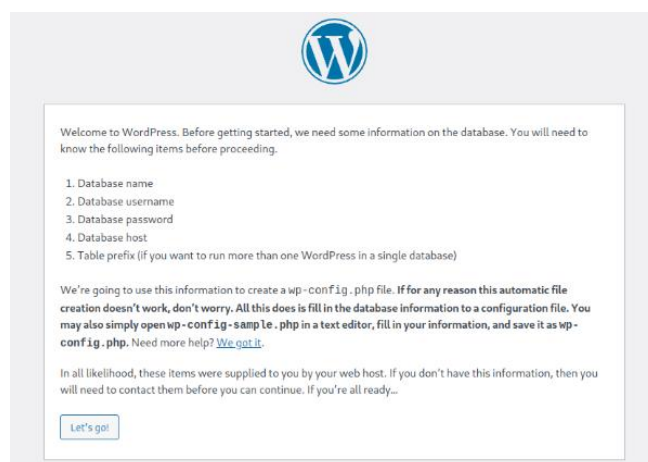
Slika 21: Dodeljevanje pravic uporabniku WordPress podatkovne baze

S tem postopkom končanim sva imela delujoč WordPress na spletni strani.



Slika 22: Prvi pogled na uspešno namestitev WordPressa

Za odprtjem spletne strani naju je sedaj pozdravil še čarovnik za končanje namestitve WordPressa. Hitro sva šla skozenj, vpisala vpisne podatke, ki sva jih pred tem nastavila v terminalu.



The screenshot shows the WordPress installation 'Welcome' screen. At the top is the WordPress logo. Below it, a text box contains the following information:

Welcome to WordPress. Before getting started, we need some information on the database. You will need to know the following items before proceeding.

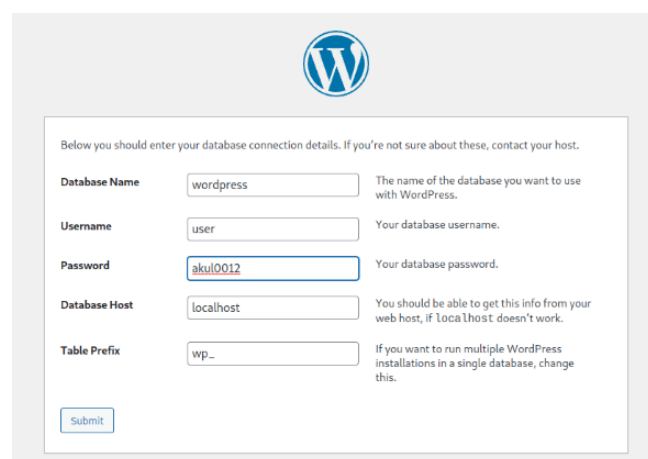
1. Database name
2. Database username
3. Database password
4. Database host
5. Table prefix (if you want to run more than one WordPress in a single database)

We're going to use this information to create a `wp-config.php` file. If for any reason this automatic file creation doesn't work, don't worry. All this does is fill in the database information to a configuration file. You may also simply open `wp-config-sample.php` in a text editor, fill in your information, and save it as `wp-config.php`. Need more help? [We got it.](#)

In all likelihood, these items were supplied to you by your web host. If you don't have this information, then you will need to contact them before you can continue. If you're all ready...

At the bottom is a button labeled 'Let's go!'.

Slika 23: Dokončno nameščanje WordPressa #1



The screenshot shows the WordPress installation 'Database Configuration' screen. At the top is the WordPress logo. Below it, a text box contains the following information:

Below you should enter your database connection details. If you're not sure about these, contact your host.

Database Name: The name of the database you want to use with WordPress.

Username: Your database username.

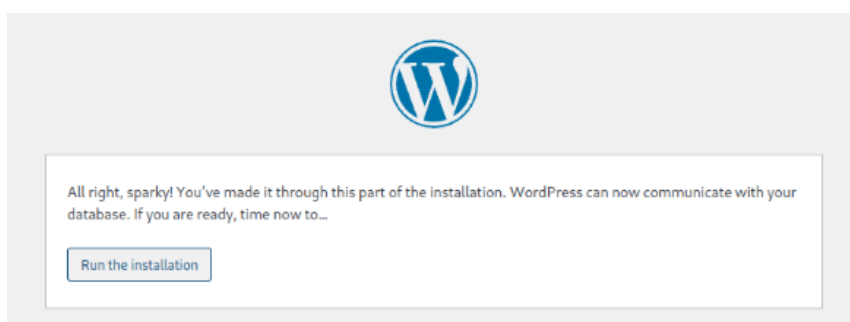
Password: Your database password.

Database Host: You should be able to get this info from your web host, if localhost doesn't work.

Table Prefix: If you want to run multiple WordPress installations in a single database, change this.

At the bottom is a button labeled 'Submit'.

Slika 24: Dokončno nameščanje WordPressa #2



The screenshot shows the WordPress installation 'Completion' screen. At the top is the WordPress logo. Below it, a text box contains the following information:

All right, sparky! You've made it through this part of the installation. WordPress can now communicate with your database. If you are ready, time now to...

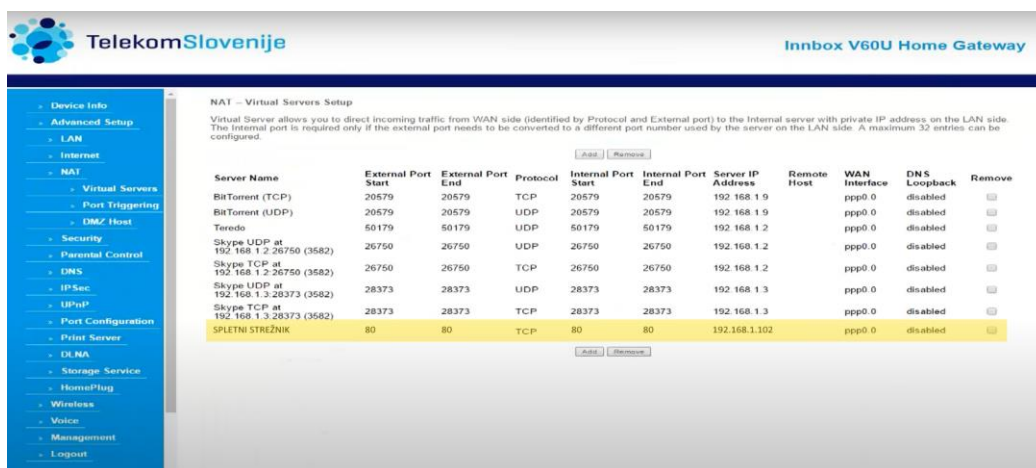
At the bottom is a button labeled 'Run the installation'.

Slika 25: Dokončno nameščanje WordPressa #3

4.4 Odprtje vrat

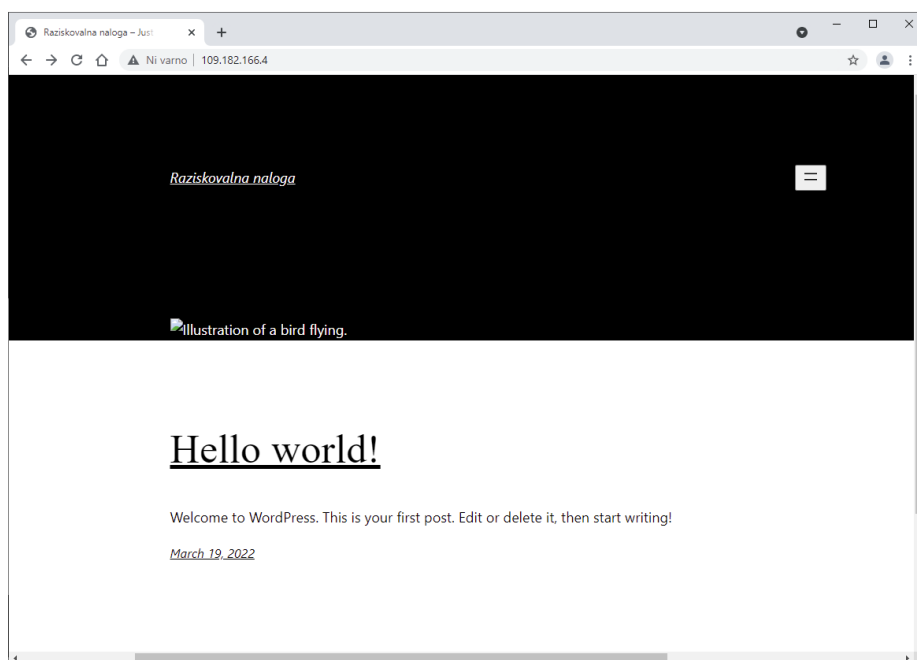
Ker sva želela imeti tudi neposreden dostop do naprave preko domačega usmerjevalnika, sva se odločila izvesti računalniški postopek znan kot odprtje/posredovanje vrat (angleško *port forwarding*). To omogoči, da lahko zunanji uporabnik direktno dostopa do strani z IP naslovom domačega usmerjevalnika.

Za odprtje vrat sva se povezala na domač ruter preko lokalnega naslova *192.168.0.1*. Naposled sva našla nastavitve za vzpostavitev virtualnih strežnikov in preko njih odprla vrata 80 (standardna vrata) preko protokola TCP.



Slika 26: Odprtje vrat

Stran je tako postala dostopna preko IP naslova *109.182.166.4*.



Slika 27: Dostopanje do strani preko vrat

4.5 Končni izgled sistema

Po vsem postopku končni sistem izgleda kot kaže slika 28.



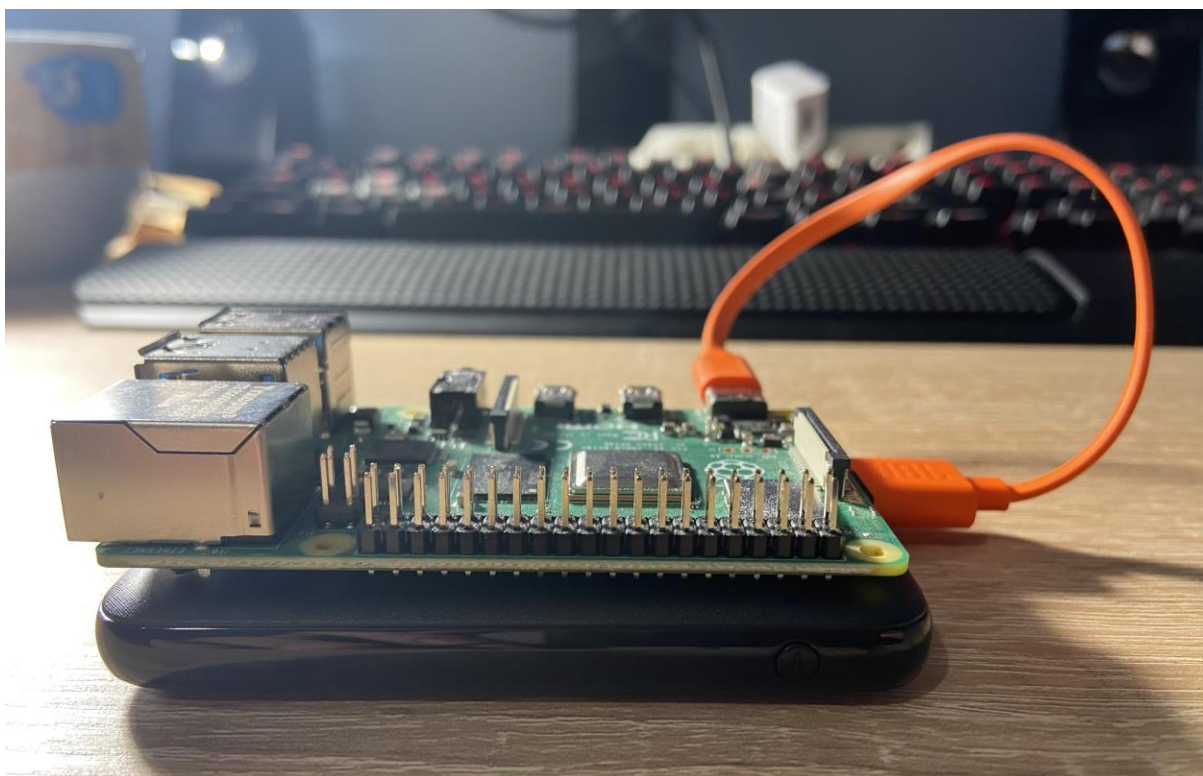
Slika 28: Končni izgled sistema

Za delovanje sicer ne potrebuje drugega kot napajanje preko kabla ter povezavo s spletom, razen v primeru, da želimo videti dogajanje preko uporabniškega vmesnika, a to v večini primerov ni potrebno, ko je enkrat strežnik že zagnan.

Brezžičen spletni strežnik

Med preizkušanjem novih načinov uporabe sistema sva med drugimi poskusila tudi napajanje preko prenosne baterije. Ugotovila sva, da lahko sistem deluje popolnoma brezžično, le potrebno je zagnati komando za zagon Apacheja in ostalih programov – ta niz ukazov je seveda mogoče avtomatizirati, da se izvede ob vsakem ponovnem zagonu.

To omogoča mnogo prednosti, saj lahko sistem fizično prosto prestavljamo po prostoru brez strahu pred nedelovanjem, na splošno pa to pomeni, da imamo zelo majhen in prenosen strežnik v velikosti računalniške miške, ki ga lahko brezskrbno spravimo tudi v žep.



Slika 29: Brezžičen spletni strežnik #1



Slika 30: Brezžičen spletni strežnik #2

5 ZAKLJUČEK

V nalogi sva ustvarila spletni strežnik, ki sva ga uporabila za gostovanje z Apache programsko opremo. Za daljinsko upravljanje sva preizkusila uporabo TeamViewerja, a le-ta zaradi nizkih hitrosti domače internetne povezave in nezmogljivosti naprave ni bil ustrezen. Ker vsak spletni strežnik potrebuje tudi podatkovno bazo, sva na svojega namestila orodje PHPMyAdmin, ki omogoča lažje ustvarjanje in urejanje baz ter podatkov v njih; to orodje ne bi delovalo brez MySQL-ja in PHP-ja, zato sva tudi to dvojce namestila.

Sprva sva za javno domeno uporabila orodje po imenu Ngrok, saj sva to videla kot najlažjo rešitev, nato pa sva odprla vrata domačega usmerjevalnika in uporabila to. Namestila sva WordPress, ki omogoča enostavno ustvarjanje in urejanje spletnih strani. Z njim sva ustvarila testno stran, ki je delovala kot preizkus, če vse deluje, kot bi moralo. Kot pričakovano je vse delovalo in nisva imela večjih težav v postopku.

Naučila sva se veliko koristnega in pridobila novo znanje, ki nama bo koristilo v nadaljnji izobrazbi ter tudi na splošno v življenju. Če bova kdaj potrebovala gostovanje spletne strani, bova imela že vnaprej ustvarjen sistem in bova tako prihranila s časom in denarjem.

Analiza hipotez

H1: Z Raspberry Pi-jem je mogoče uspešno gostovati spletno aplikacijo.

Hipoteza potrjena. Z Raspberry Pi-jem je res mogoče gostovati spletno aplikacijo, kar sva tudi dokazala in prikazala celoten postopek.

H2: Znanje, pridobljeno na računalniški srednji šoli, je zadostno za izdelavo lastnega spletnega strežnika

Hipoteza zavržena. Morala sva predelati mnogo dokumentacij za uporabljeno strojno in programsko opremo, na splošno pa v šoli nismo obdelovali potrebne snovi. (To pa ne pomeni, da bo tako na vsaki računalniški srednji šoli.)

H3: Strežniki z uporabo Raspberry Pi-ja so zaradi energetske in finančne učinkovitosti primernejši za domače uporabnike.

Hipoteza večinsko potrjena. Cenejši so kratkoročno in dolgoročno in primerni za veliko večino končnih uporabnikov, a moramo biti pozorni na specifikacije sistema. Za procesorsko zahtevnejše strežniške programe je Raspberry Pi manj primeren, je pa za dolgoročno osebno rabo manjših obsežnosti nedvomno najprimernejša ter energetske in finančno najučinkovitejša rešitev.

Viri in literatura

- Campbell, S. (20. Marec 2015). *Circuit Basics*. Pridobljeno 16. Januar 2022 iz 42 OF THE MOST USEFUL RASPBERRY PI COMMANDS:
<https://www.circuitbasics.com/useful-raspberry-pi-commands/>
- Cawley, C. (15. Februar 2021). *Make use of*. Pridobljeno 5. Marec 2022 iz How to Host a WordPress Site on Raspberry: <https://www.makeuseof.com/tag/host-wordpress-raspberry-pi/>
- Erdelji, M. (14. 6 2016). *Monitor*. Pridobljeno iz Uporabna malina:
<https://www.monitor.si/clanek/uporabnamalina/174787/>
- Hernandez, J. (8. Maj 2019). *Sumo logic* . Pridobljeno 29. Januar 2022 iz What is Apache? In-Depth Overview of Apache Web Server: <https://www.sumologic.com/blog/apache-web-server-introduction/>
- Iamabhijha. (6. December 2021). *GeeksforGeeks*. Pridobljeno 2. Marec 2022 iz How to Install PHP on Linux?: <https://www.geeksforgeeks.org/how-to-install-php-on-linux/>
- Kinsta . (28. Maj 2021). *Kinsta* . Pridobljeno 13. Februar 2022 iz What Is Apache Web Server?: <https://kinsta.com/knowledgebase/what-is-apache/>
- Lindevs. (5. Maj 2021). *Lindevs*. Pridobljeno 2. Marec 2022 iz Install phpMyAdmin on Raspberry Pi: <https://lindevs.com/install-phpmyadmin-on-raspberry-pi/>
- Linuxize. (9. Junij 2019). *Linuxize*. Pridobljeno 22. Februar 2022 iz How to Install Apache Web Server on Raspberry Pi: <https://linuxize.com/post/how-to-install-apache-on-raspberry-pi/>
- Loshin, P. (Februar 2022). *Teach target*. Pridobljeno 5. Marec 2022 iz Structured Query Language (SQL): <https://www.techtartget.com/searchdatamanagement/definition/SQL>
- Piltch, A. (2. Marec 2022). *Tom's hardware*. Pridobljeno 10. Marec 2022 iz Best microSD Cards for Raspberry Pi 2022: <https://www.tomshardware.com/best-picks/raspberry-pi-microsd-cards>
- Raspberry Pi Foundation*. (brez datuma). Pridobljeno iz Teach, Learn, and Make with Raspberry Pi: <https://www.raspberrypi.org/>

Raspberry Pi Foundation. (brez datuma). *Raspberry Pi*. Pridobljeno 22. Februar 2022 iz Raspberry Pi 4 Tech Specs: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/>

Raspberry Pi. (brez datuma). *Raspberry Pi*. Pridobljeno 25. Januar 2022 iz Raspberry Pi Documentation: <https://www.raspberrypi.com/documentation/computers/getting-started.html>

Shahid, H. (31. December 2021). *Pure VPN*. Pridobljeno 16. Marec 2022 iz How to Forward or Open Ports on Your Router?: <https://www.purevpn.com/blog/how-to-forward-ports-on-your-router/>

Snapcraft. (brez datuma). *Snapcraft*. Pridobljeno 20. Februar 2022 iz Install ngrok on Raspberry Pi: <https://snapcraft.io/install/ngrok/raspbian>

TeamViewer. (22. Januar 2022). *TeamViewer*. Pridobljeno iz Raspberry Pi Remote Desktop Access with TeamViewer: <https://www.teamviewer.com/en/solutions/remote-desktop/raspberry-pi/>

ThePi. (brez datuma). *ThePi*. Pridobljeno 19. Januar 2022 iz How to install Raspbian on the Raspberry Pi: <https://thepi.io/how-to-install-raspbian-on-the-raspberry-pi/>

IZJAVA*

Mentor ŽIGA PUŠELC v skladu z 20. členom Pravilnika o organizaciji mladinske raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje, zagotavljam, da je v raziskovalni nalogi z naslovom POSTAVITEV RASPBERRY P1 SPLETNEGA STREŽNIKA, katere avtorja sta TIM PUHAR IN LUKA SLAPNIK.

- besedilo v tiskani in elektronski obliki istovetno,
- pri raziskovanju uporabljeno gradivo navedeno v seznamu uporabljene literature,
- da je za objavo fotografij v nalogi pridobljeno avtorjevo dovoljenje in je hranjeno v šolskem arhivu,
- da sme Osrednja knjižnica Celje objaviti raziskovalno nalogo v polnem besedilu na knjižničnih portalih z navedbo, da je raziskovalna naloga nastala v okviru projekta Mladi za Celje,
- da je raziskovalno nalogo dovoljeno uporabiti za izobraževalne in raziskovalne namene s povzemanjem misli, idej, konceptov oziroma besedil iz naloge ob upoštevanju avtorstva in korektnem citiranju,
- da smo seznanjeni z razpisni pogoji projekta Mladi za Celje.

Celje, 12. APRIL 2022



Podpis mentorja

Puše

Podpis odgovorne osebe

Lazni

*

POJASNILO

V skladu z 20. členom Pravilnika raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje je potrebno podpisano izjavo mentorja (-ice) in odgovorne osebe šole vključiti v izvod za knjižnico, dovoljenje za objavo avtorja (-ice) fotografskega gradiva, katerega ni avtor (-ica) raziskovalne naloge, pa hrani šola v svojem arhivu.