

## Harjoitustyö 2

### Palvelun kontitus

Timo Lehosvuo, TTV18S1

Tatu Alatalo, TTV19SMM

Tuukka Bordi, TTV18S1

Harjoitustyö

Palveluiden automatisointi, Juho Pekki

Maaliskuu 2021

Tekniikan ja liikenteen ala

## Sisältö

<b>1</b>	<b>Johdanto .....</b>	<b>3</b>
<b>2</b>	<b>Tehtävänannot .....</b>	<b>3</b>
2.1	Vaihe 1 .....	3
2.1.1	WordPress kontti .....	3
2.1.2	MySQL kontti .....	6
2.1.3	Konttien toiminnan testaus .....	8
2.1.4	Kommentit Vaiheeseen 1 liittyen .....	11
2.2	Vaihe 2 .....	11
2.2.1	Kommentit Vaiheeseen 2 liittyen .....	15
2.3	Vaihe 3 .....	16
2.4	Vaihe 4 .....	20
<b>3</b>	<b>Lähteet .....</b>	<b>24</b>

## Kuviot

Kuvio 1.	WordPressin <i>Dockerfile</i> .....	3
Kuvio 2.	WordPressin volume .....	4
Kuvio 3.	Populoitu WordPressin volume .....	4
Kuvio 4.	Imagen buildaaminen wptesti -tagilla .....	5
Kuvio 5.	<i>wp-config.php</i> .....	5
Kuvio 6.	<i>wp-config.php</i> suolaus .....	6
Kuvio 7.	MySQL-kontin käynnistäminen .....	6
Kuvio 8.	MariaDB-kontin <i>Dockerfile</i> .....	7
Kuvio 9.	MariaDB-kontin määrittystiedosto .....	8
Kuvio 10.	MariaDB:n volumet .....	8
Kuvio 11.	WordPressin setup-sivun läpikäynti .....	9
Kuvio 12.	Valmis WordPress setup .....	9
Kuvio 13.	WordPress admin-näkymä .....	10
Kuvio 14.	WordPress tietokantataulut .....	10

Kuvio 15. Käynnissä olevat kontit .....	11
Kuvio 16. Openrekisteri <i>/etc/hosts</i> tiedostossa .....	11
Kuvio 17. Sertifikaatin kopiointi virtuaalikoneeseen .....	12
Kuvio 18. Sertifikaatti oikeassa sijainnissa .....	12
Kuvio 19. Docker WordPress image push toimii .....	12
Kuvio 20. Docker MySQL image push toimii .....	13
Kuvio 21. Docker WordPress image pull lataa tiedostoja .....	13
Kuvio 22. Docker MySQL image pull lataa tiedostoja .....	14
Kuvio 23. Imaget ladattu .....	14
Kuvio 24. Imaget ladattu ja laitetty pyörimään .....	15
Kuvio 25. Palvelu toiminnassa .....	15
Kuvio 26. WordPressin <i>Dockerfilen</i> muutokset.....	16
Kuvio 27. WordPressin ja WP-CLI:n automaattinen asennus .....	17
Kuvio 28. <i>wp-config.php</i> -tiedoston automaattinen täydennys WP-CLI työkalulla käyttäen Dockerfilen <i>env variableja</i> .....	17
Kuvio 29. Taulujen automaattinen luonti tietokantaan WP-CLI työkalun avulla käyttäen Dockerfilen <i>env variableja</i> .....	18
Kuvio 30. Muutoskenaariona WordPressin teeman vaihtaminen WP-CLI työkalulla	18
Kuvio 31. MySQL-kontin volumet.....	19
Kuvio 32. WordPressin volumet .....	19
Kuvio 33. Kontit pyörivät virheettä .....	19
Kuvio 34. Uuden koneen <i>/etc/hosts</i> tiedosto .....	20
Kuvio 35. crt-tiedosto lisätty uuteen koneeseen .....	20
Kuvio 36. <i>docker-compose</i> asennettu onnistuneesti .....	21
Kuvio 37. Docker-compose tiedosto .....	22
Kuvio 38. Kontit käynnissä uudella koneella .....	23
Kuvio 39. Palvelut käynnissä uudella koneella.....	23

# 1 Johdanto

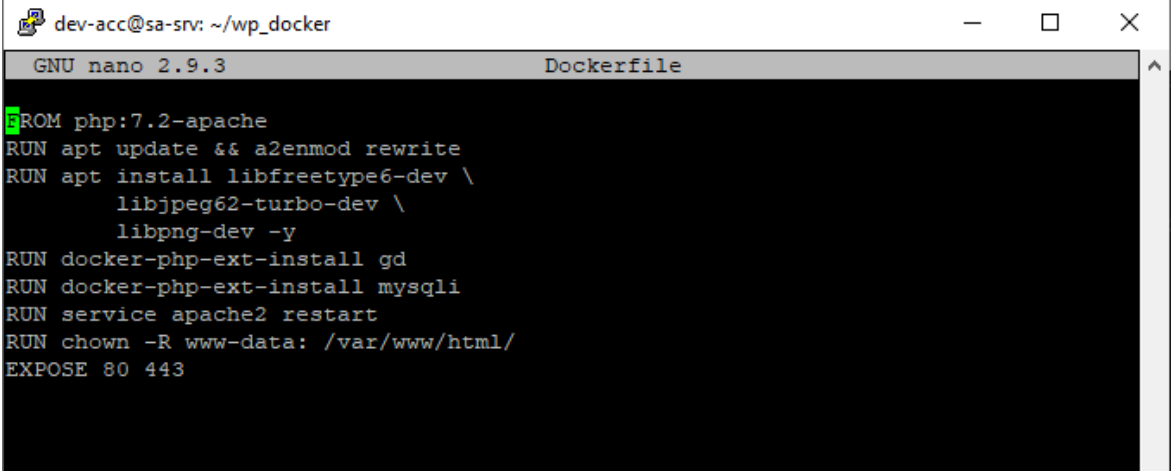
Tämä harjoitustyö on osa Jyväskylän Ammattikorkeakoulun Palveluiden Automatisointi-kurssia. Harjoituksen tarkoituksena on kontittaa palvelu, eli rakentaa oma *Dockerfile*. Harjoitustyössä luodaan myös *docker-compose.yml*-tiedosto palvelusta, jossa kaikki kontit pyörivät samalla koneella.

## 2 Tehtävänannot

### 2.1 Vaihe 1

#### 2.1.1 WordPress kontti

Katsoimme DockerHubista sopivan PHP-kontin ja päädyimme valitsemaan version 7.2 joka sisältää valmiiksi asennettuna Apachen. Otimme saman version kuin mitä käytimme edellisessä harjoitustyössä. Dockerfile muodostui seuraavanlaiseksi:



```
dev-acc@sa-srv: ~/wp_docker
GNU nano 2.9.3 Dockerfile
FROM php:7.2-apache
RUN apt update && a2enmod rewrite
RUN apt install libfreetype6-dev \
    libjpeg62-turbo-dev \
    libpng-dev -y
RUN docker-php-ext-install gd
RUN docker-php-ext-install mysqli
RUN service apache2 restart
RUN chown -R www-data: /var/www/html/
EXPOSE 80 443
```

Kuvio 1. WordPressin *Dockerfile*

Tärkeää *Dockerfile*:ssä on se, että siinä asennetaan muutama kirjasto, joita WordPress tarvitsee (Florian 2016; PHP 2021). Teimme WordPressille myös volumen:

```

dev-acc@sa-srv: ~/wp_docker
dev-acc@sa-srv:~/wp_docker$ sudo docker volume create wp
[sudo] password for dev-acc:
wp
dev-acc@sa-srv:~/wp_docker$ sudo docker volume inspect wp
[
  {
    "CreatedAt": "2021-02-19T10:56:10Z",
    "Driver": "local",
    "Labels": {},
    "Mountpoint": "/var/snap/docker/common/var-lib-docker/volumes/wp/_data",
    "Name": "wp",
    "Options": {},
    "Scope": "local"
  }
]
dev-acc@sa-srv:~/wp_docker$

```

Kuvio 2. WordPressin volume

Seuraavaksi lataimme WordPressin tarvitsemat tiedostot ja siirsimme ne luomaamme volumeen.

```
$ cd /tmp && wget https://wordpress.org/latest.tar.gz
```

```
$ tar zxvf latest.tar.gz
```

```
$ sudo mv wordpress/* /var/snap/docker/common/var-lib-docker/volumes/wp/_data
```

```
$ rm -rf latest.tar.gz && rm -rf wordpress
```

Volume saatiin populoitua:

```

dev-acc@sa-srv: /tmp
dev-acc@sa-srv:/tmp$ ls -ls /var/snap/docker/common/var-lib-docker/volumes/wp/_data
ls: cannot access '/var/snap/docker/common/var-lib-docker/volumes/wp/_data': Permission denied
dev-acc@sa-srv:/tmp$ sudo ls -ls /var/snap/docker/common/var-lib-docker/volumes/wp/_data
total 216
 4 -rw-r--r--  1 dev-acc dev-acc  405 Feb  6  2020 index.php
20 -rw-r--r--  1 dev-acc dev-acc 19915 Feb 12  2020 license.txt
 8 -rw-r--r--  1 dev-acc dev-acc  7278 Jun 26  2020 readme.html
 8 -rw-r--r--  1 dev-acc dev-acc  7101 Jul 28  2020 wp-activate.php
 4 drwxr-xr-x  9 dev-acc dev-acc  4096 Feb  3 21:11 wp-admin
 4 -rw-r--r--  1 dev-acc dev-acc   351 Feb  6  2020 wp-blog-header.php
 4 -rw-r--r--  1 dev-acc dev-acc  2328 Oct  8 21:15 wp-comments-post.php
 4 -rw-r--r--  1 dev-acc dev-acc  2913 Feb  6  2020 wp-config-sample.php
 4 drwxr-xr-x  4 dev-acc dev-acc  4096 Feb  3 21:11 wp-content
 4 -rw-r--r--  1 dev-acc dev-acc  3939 Jul 30  2020 wp-cron.php
12 drwxr-xr-x 25 dev-acc dev-acc 12288 Feb  3 21:11 wp-includes
 4 -rw-r--r--  1 dev-acc dev-acc  2496 Feb  6  2020 wp-links-opml.php
 4 -rw-r--r--  1 dev-acc dev-acc  3300 Feb  6  2020 wp-load.php
52 -rw-r--r--  1 dev-acc dev-acc 49831 Nov  9 10:53 wp-login.php
12 -rw-r--r--  1 dev-acc dev-acc  8509 Apr 14  2020 wp-mail.php
24 -rw-r--r--  1 dev-acc dev-acc 20975 Nov 12 14:43 wp-settings.php
32 -rw-r--r--  1 dev-acc dev-acc 31337 Sep 30 21:54 wp-signup.php
 8 -rw-r--r--  1 dev-acc dev-acc  4747 Oct  8 21:15 wp-trackback.php
 4 -rw-r--r--  1 dev-acc dev-acc  3236 Jun  8  2020 xmlrpc.php
dev-acc@sa-srv:/tmp$

```

Kuvio 3. Populoitu WordPressin volume

Buildaamme imagen *Dockerfile*:stä ja annamme sille tagin *'wptest'*:

```

dev-acc@sa-srv: ~/wp_docker
dev-acc@sa-srv:~/wp_docker$ sudo docker build -t wptesti .
Sending build context to Docker daemon 2.048kB
Step 1/6 : FROM php:7.2-apache
--> c61d277263e1
Step 2/6 : RUN apt update && a2enmod rewrite
--> Using cache
--> 3ee7019495c0
Step 3/6 : RUN apt install libfreetype6-dev libjpeg62-turbo-dev libpng-dev -y
--> Using cache
--> e2d711cd5795
Step 4/6 : RUN docker-php-ext-install gd
--> Using cache
--> 8420bbaefc9e
Step 5/6 : RUN chown -R www-data: /var/www/html/
--> Running in 9f2607bbc538
Removing intermediate container 9f2607bbc538
--> d6cf4343d120
Step 6/6 : EXPOSE 80 443
--> Running in 1dca20ca772a
Removing intermediate container 1dca20ca772a
--> 9d1c690779fa
Successfully built 9d1c690779fa
Successfully tagged wptesti:latest
dev-acc@sa-srv:~/wp_docker$ sudo docker images
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
wptesti              latest          9d1c690779fa    26 seconds ago 435MB
mysqlatesti          latest          a1243cb91bf1    16 minutes ago 352MB
mariadb              10.1.48         3ce131c51a9f    4 weeks ago    352MB
mariadb              latest          ade39f0469a3    4 weeks ago    407MB
php                  7.2-apache      c61d277263e1    2 months ago   410MB
dev-acc@sa-srv:~/wp_docker$

```

Kuvio 4. Imagen buildaaminen wptesti -tagilla

Loimme kopion *wp-config-sample.php*:sta, nimesimme se nimellä *wp-config.php* ja syötimme tiedot manuaalisesti:

```

dev-acc@sa-srv: ~/wp_docker
GNU nano 2.9.3 /var/snap/docker/common/var-lib-docker/volumes/wp/_data/wp-config.php

*
* @link https://wordpress.org/support/article/editing-wp-config-php/
*
* @package WordPress
*/

// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'wordpress' );

/** MySQL database username */
define( 'DB_USER', 'wordpressuser' );

/** MySQL database password */
define( 'DB_PASSWORD', 'qwerty123' );

/** MySQL hostname */
define( 'DB_HOST', '192.168.1.10' );

/** Database Charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8mb4' );

/** The Database Collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', 'utf8mb4_unicode_ci' );

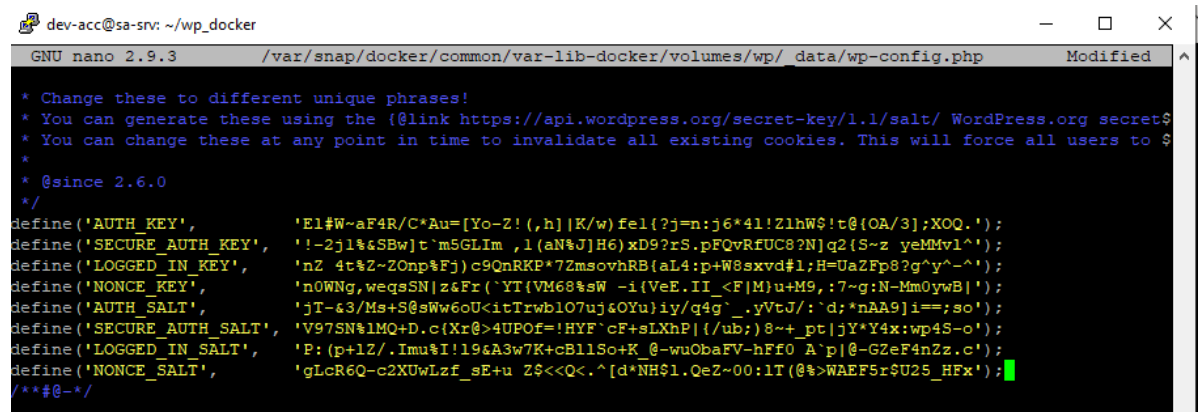
/**#@+
 * Authentication Unique Keys and Salts.
 */

```

Kuvio 5. *wp-config.php*

Seuraavaksi kopioimme samaiseen config-tiedostoon turva-avaimet ynnä muut turvallisuusominaisuudet. Tämä tehtiin seuraavalla komennolla:

```
$ curl -s https://api.wordpress.org/secret-key/1.1/salt/
```



```
dev-acc@sa-srv: ~/wp_docker
GNU nano 2.9.3 /var/snap/docker/common/var-lib-docker/volumes/wp_data/wp-config.php Modified
* Change these to different unique phrases!
* You can generate these using the (link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org secret$
* You can change these at any point in time to invalidate all existing cookies. This will force all users to $
*
* @since 2.6.0
*/
define('AUTH_KEY', 'El#W~aF4R/C*Au=[Yo~Z!(,h]|K/w)fel{?j=n:j6*4l!ZlhW$!t@{OA/3];XOQ.>');
define('SECURE_AUTH_KEY', '!-2jl%&SBw]t`m5GLIm ,l(aN%J]H6)xD9?rS.pFQvRfUC8?N]q2{S~z yeMMv1^');
define('LOGGED_IN_KEY', 'nZ 4t%Z~ZOnp%Fj)c9QnRKP*7ZmsovhRB{aL4:p+W8sxxvd#l;H=UaZfP8?g^y^~^');
define('NONCE_KEY', 'n0WNg,weqsN|z&Fr(`YT{VM68%$W -i{VeE.II_<F|M}u+M9,:7~g:N-Mm0ywB|');
define('AUTH_SALT', 'jT-43/Ms+S@sWw6oU<itTrwb1O7uj&OYu}iy/q4g_.yVtJ/:`d;*nAA9]i==;so');
define('SECURE_AUTH_SALT', 'V97SN%lMQ+D.c(Xr@>4UPOf=!HYF`cF+sLXhP|(/ub;)8~+_pt|jY*Y4x:wp4S-o');
define('LOGGED_IN_SALT', 'P:(p+lZ/.Imu%I!l9&A3w7K+cBllSo+K @-wuObaFV-hFf0 A`p|@-GZeF4nZz.c');
define('NONCE_SALT', 'gLcR6Q-c2XUwLzf_sE+u Z$<<Q<.^[d*NH$1.QeZ~00:1T(@%>WAEF5r$U25_HFx');
/**#@-*/
```

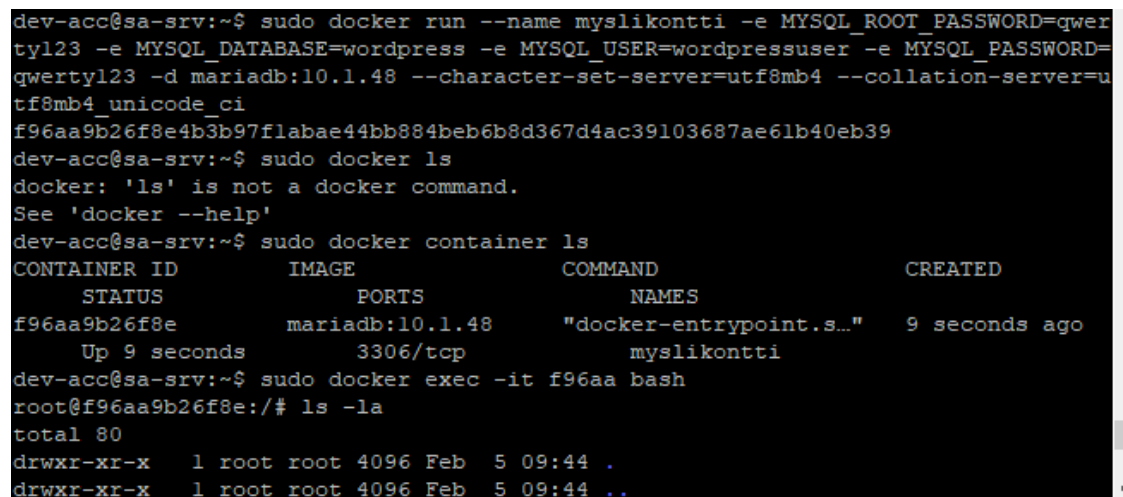
Kuvio 6. *wp-config.php* suolaus

Laitoimme WordPressin pyörimään seuraavalla komennolla:

```
$ sudo docker run -dp 80:80 -v wp:/var/www/html --name wpkontti wptesti
```

## 2.1.2 MySQL kontti

MySQL-kontin käynnistimme seuraavalla komennolla:



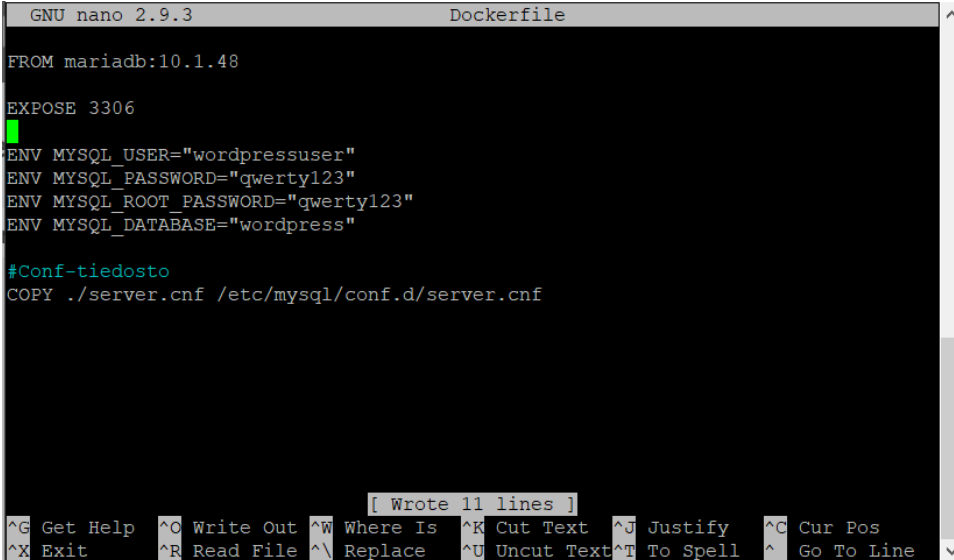
```
dev-acc@sa-srv:~$ sudo docker run --name myslikontti -e MYSQL_ROOT_PASSWORD=qwer
tyl23 -e MYSQL_DATABASE=wordpress -e MYSQL_USER=wordpressuser -e MYSQL_PASSWORD=
qwertyl23 -d mariadb:10.1.48 --character-set-server=utf8mb4 --collation-server=u
tf8mb4_unicode_ci
f96aa9b26f8e4b3b97flabae44bb884beb6b8d367d4ac39103687ae61b40eb39
dev-acc@sa-srv:~$ sudo docker ls
docker: 'ls' is not a docker command.
See 'docker --help'
dev-acc@sa-srv:~$ sudo docker container ls
CONTAINER ID        IMAGE               COMMAND             CREATED
STATUS             PORTS              NAMES
f96aa9b26f8e        mariadb:10.1.48    "docker-entrypoint.s..." 9 seconds ago
Up 9 seconds       3306/tcp           myslikontti
dev-acc@sa-srv:~$ sudo docker exec -it f96aa bash
root@f96aa9b26f8e:/# ls -la
total 80
drwxr-xr-x  1 root root 4096 Feb  5 09:44 .
drwxr-xr-x  1 root root 4096 Feb  5 09:44 ..
```

Kuvio 7. MySQL-kontin käynnistäminen

Käyttämämme komento myös alla:

```
$ sudo docker run \
  --name myslikontti \
  -e MYSQL_ROOT_PASSWORD=qwerty123 \
  -e MYSQL_DATABASE=wordpress \
  -e MYSQL_USER=wordpressuser \
  -e MYSQL_PASSWORD=qwerty123 \
  -d mariadb:10.1.48 \
  --character-set-server=utf8mb4 \
  --collation-server=utf8mb4_unicode_ci
```

Kontin käynnistäminen tällä tavalla toimii, mutta komento on aika pitkä. Tätä paikkaamaan asetimme Environment Variablet *Dockerfile*ssä ja loimme *server.cnf* -tiedoston volumen sisälle, joka määrittelee esimerkiksi MariaDB:n character setit ym. Tärkeät asetukset (MariaDB 2021).



```
GNU nano 2.9.3 Dockerfile
FROM mariadb:10.1.48
EXPOSE 3306
ENV MYSQL_USER="wordpressuser"
ENV MYSQL_PASSWORD="qwerty123"
ENV MYSQL_ROOT_PASSWORD="qwerty123"
ENV MYSQL_DATABASE="wordpress"
#Conf-tiedosto
COPY ./server.cnf /etc/mysql/conf.d/server.cnf
[ Wrote 11 lines ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line
```

Kuvio 8. MariaDB-kontin *Dockerfile*



```

dev-acc@sa-srv: ~/mysql_docker
GNU nano 2.9.3 server.cnf

[server]
collation_server=utf8mb4_unicode_ci
bind_address=0.0.0.0
character_set_server=utf8mb4

[ File 'server.cnf' is unwritable ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line

```

Kuvio 9. MariaDB-kontin määrittystiedosto

Lisäksi teimme toisen volumen MariaDB:n datalle, jotta tietokanta ei häviä uudelleenkäynnistyksen yhteydessä.

```

dev-acc@sa-srv:~/mysql_docker$ sudo docker inspect mysql_conf mysql_data
[
  {
    "CreatedAt": "2021-02-26T09:28:08Z",
    "Driver": "local",
    "Labels": {},
    "Mountpoint": "/var/lib/docker/volumes/mysql_conf/_data",
    "Name": "mysql_conf",
    "Options": {},
    "Scope": "local"
  },
  {
    "CreatedAt": "2021-03-16T11:27:48Z",
    "Driver": "local",
    "Labels": {},
    "Mountpoint": "/var/lib/docker/volumes/mysql_data/_data",
    "Name": "mysql_data",
    "Options": {},
    "Scope": "local"
  }
]
dev-acc@sa-srv:~/mysql_docker$


```

Kuvio 10. MariaDB:n volumet

Mounttasimme mysql\_data volumen kansioon `/var/lib/mysql/` ja mysql\_conf volumen kansioon `/etc/mysql/conf.d/`.

### 2.1.3 Konttien toiminnan testaus

Setup-sivu aukenee, joka tarkoittaa sitä, että tietokantayhteys toimii:



## Welcome

Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

### Information needed

Please provide the following information. Don't worry, you can always change these settings later.

**Site Title**

**Username**   
 Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.

**Password**  [Hide](#)  
 Very weak  
**Important:** You will need this password to log in. Please store it in a secure location.

**Confirm Password** ☒ Confirm use of weak password


**Your Email**   
 Double-check your email address before continuing.

**Search engine visibility** ☐ Discourage search engines from indexing this site  
 It is up to search engines to honor this request.

[Install WordPress](#)

Kuvio 11. WordPressin setup-sivun läpikäynti

Asetuksien asetus onnistui:



## Success!

WordPress has been installed. Thank you, and enjoy!

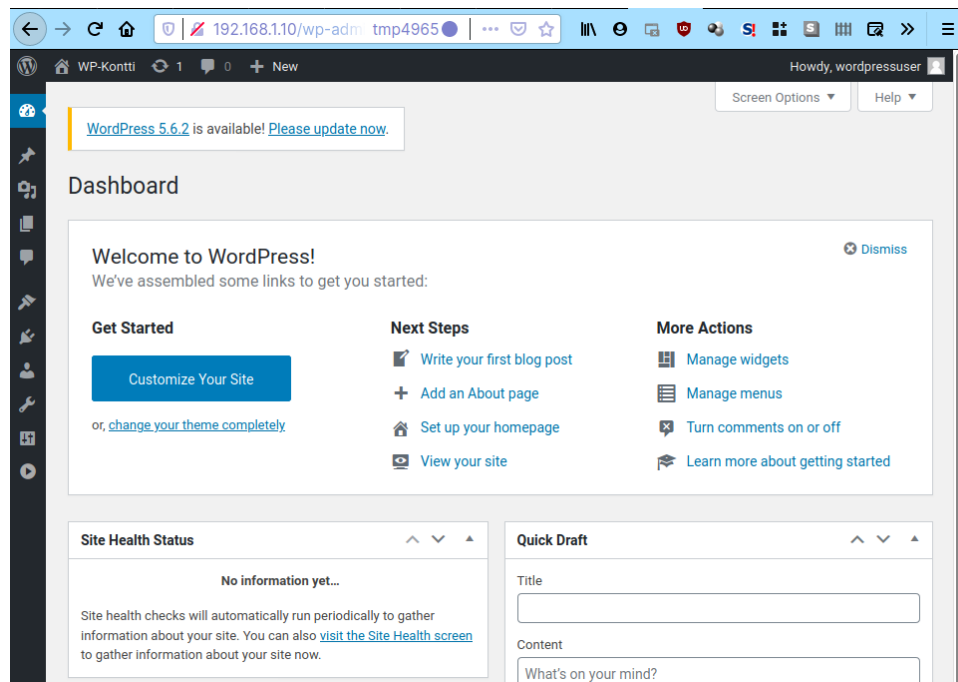
**Username** wordpressuser

**Password** Your chosen password.

[Log In](#)

Kuvio 12. Valmis WordPress setup

Admin-näkymä aukeni setupin suorittamisen jälkeen:



Kuvio 13. WordPress admin-näkymä

Tarkastimme vielä tietokannasta sen, olivatko taulut luotu:

```
Database changed
MariaDB [wordpress]> show tables;
+-----+
| Tables_in_wordpress |
+-----+
| wp_commentmeta      |
| wp_comments         |
| wp_links            |
| wp_options          |
| wp_postmeta         |
| wp_posts            |
| wp_term_relationships |
| wp_term_taxonomy    |
| wp_termmeta         |
| wp_terms            |
| wp_usermeta         |
| wp_users            |
+-----+
12 rows in set (0.00 sec)

MariaDB [wordpress]>
```

Kuvio 14. WordPress tietokantataulut

Vielä kuva pyörivistä konteista:

```
dev-acc@sa-srv:~$ sudo docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED
STATUS        PORTS     NAMES
81709f000155   wptesti    "docker-php-entrypoi..." 5 days ago
Up 9 minutes   0.0.0.0:80->80/tcp, 443/tcp  wpkontti
fbcaa4352244   mysqltesti "docker-entrypoint.s..." 5 days ago
Up 9 minutes   0.0.0.0:3306->3306/tcp      mysqlkontti
dev-acc@sa-srv:~$
```

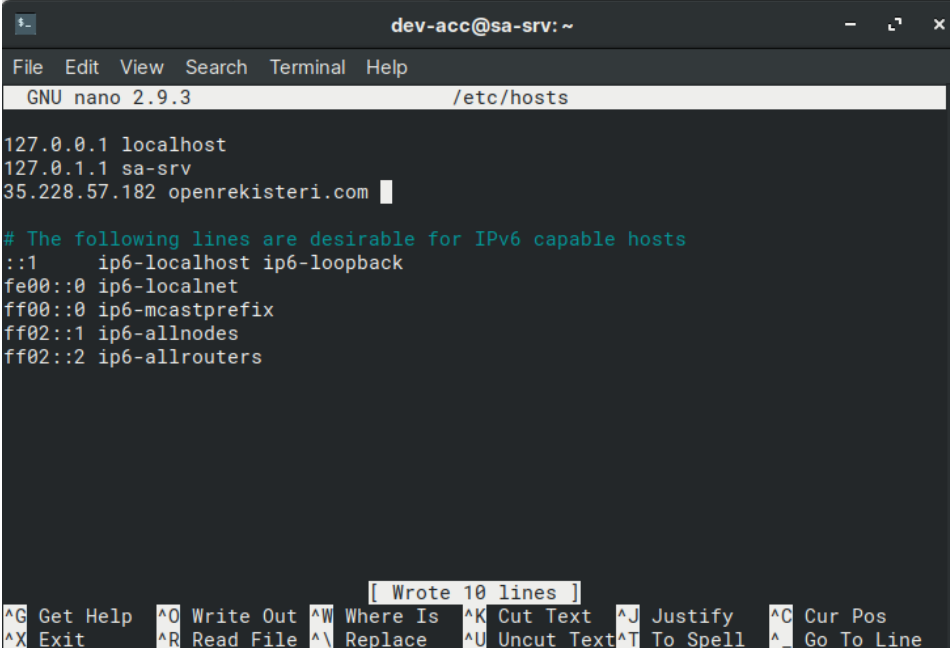
Kuvio 15. Käynnissä olevat kontit

#### 2.1.4 Kommentit Vaiheeseen 1 liittyen

- WordPressin voluumin voisi populoida automaattisesti Dockerfileä käyttämällä (esim. COPY komento).
- WordPressin konfiguraation voisi myös niin ikään automatisoida Dockerfilellä ja skripteillä.
- Tulemmekin käyttämään automatisointia enemmän hyödyksi Vaiheessa 3. Emme alkaneet suorittamaan tätä vaihetta uudestaan vaikka löysimmekin paljon fiksumpia tapoja tehdä asioita myöhemmissä vaiheissa.

## 2.2 Vaihe 2

Lisäsimme `/etc/hosts` tiedostoon rivin `"35.228.57.182 openrekisteri.com"`:



```
dev-acc@sa-srv: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 /etc/hosts

127.0.0.1 localhost
127.0.1.1 sa-srv
35.228.57.182 openrekisteri.com

# The following lines are desirable for IPv6 capable hosts
::1      ip6-localhost ip6-loopback
fe00::0  ip6-localnet
ff00::0  ip6-mcastprefix
ff02::1  ip6-allnodes
ff02::2  ip6-allrouters

[ Wrote 10 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line
```

Kuvio 16. Openrekisteri `/etc/hosts` tiedostossa

Tämän jälkeen lataimme tarvittavan sertifiikaatin palvelimelle:

```
tuukka@tuukka-N501VW:~/Downloads$ scp -P 49999 ca.crt dev-acc@m2296.firewall-gateway.com:~
Warning: the ECDSA host key for '[m2296.firewall-gateway.com]:49999' differs from the key for the IP address '[80.186.203.162]:49999'
Offending key for IP in /home/tuukka/.ssh/known_hosts:38
Matching host key in /home/tuukka/.ssh/known_hosts:50
Are you sure you want to continue connecting (yes/no)? yes
ca.crt
100% 1984 512.9KB/s 00:00
tuukka@tuukka-N501VW:~/Downloads$
```

Kuvio 17. Sertifiikaatin kopiointi virtuaalikoneeseen

Siirsimme sertifiikaatin kansioon `/etc/docker/certs.d/openrekisteri.com:443/ca.crt`:

```
dev-acc@sa-srv: ~
dev-acc@sa-srv:~$ sudo ls -ls /etc/docker/certs.d/openrekisteri.com\:443/
total 4
4 -rwxr-xr-x 1 root root 1984 Feb 26 10:29 ca.crt
dev-acc@sa-srv:~$
```

Kuvio 18. Sertifiikaatti oikeassa sijainnissa

Uudelleenkäynnistimme Docker-palvelu komennolla:

`$ sudo systemctl restart docker`

Kun palvelu oli käynnistynyt uudelleen, testasimme toimiko “push” komento:

```
dev-acc@sa-srv: ~
/ryhma8_mysql
dev-acc@sa-srv:~$ sudo docker image push openrekisteri.com:443/ryhma8_wp
The push refers to repository [openrekisteri.com:443/ryhma8_wp]
0d91683d08a8: Pushed
489e16aee4a5: Pushed
aff2b2b6eb12: Pushed
6b09b3cff8fa: Pushed
75f996fd5e3c: Pushed
0a5ad425add4: Pushed
5dc980197467: Pushed
cc45506c4447: Pushed
6ec4d4ce53cc: Pushed
9a60d912a14f: Pushed
ce60a0c97d4a: Pushing 47.7MB/54.11MB
e1cd0107ea85: Pushed
914aleddd57a: Pushing 13.62MB
0ff9183bd099: Pushed
d666585087a1: Pushed
bc0429138e0d: Pushing 7.55MB/45.89MB
02eef72b445f: Pushing 5.12kB
e45a78df7536: Waiting
ddcd8d2fcf7e: Waiting
87c8a1d8f54f: Waiting
```

Kuvio 19. Docker WordPress image push toimii

```
dev-acc@sa-srv:~$ sudo docker image push openrekisteri.com:443/ryhma8_mysql
The push refers to repository [openrekisteri.com:443/ryhma8_mysql]
becfc32d49e0: Pushed
8d810cbc5fb2: Pushed
c7bb90f98231: Pushing 136.4MB/270.2MB
eef6d575cbca: Pushed
7f87780aaf84: Pushed
09d986b8286b: Pushing 548.9kB/3.473MB
268cd8475e2c: Pushing 1.536kB
0aa571e40e9f: Pushing 1.164MB/3.857MB
b5fa2c166202: Pushing 393.2kB/10.84MB
b8fa98b1d1bf: Waiting
9f10818f1f96: Waiting
27502392e386: Waiting
c95d2191d777: Waiting
```

Kuvio 20. Docker MySQL image push toimii

Poistimme vanhat imaget/kontit ja latasimme ne uudestaan opettajan rekisteristä:

```
dev-acc@sa-srv: ~
dev-acc@sa-srv:~$ sudo docker pull openrekisteri.com:443/ryhma8_wp
Using default tag: latest
latest: Pulling from ryhma8_wp
6ec7b7d162b2: Downloading 2.507MB/27.1MB
db606474d60c: Download complete
afb30f0cd8e0: Downloading 5.407MB/76.65MB
3bb2e8051594: Download complete
4c761b44e2cc: Downloading 1.573MB/18.68MB
c2199db96575: Waiting
1b9a9381eea8: Waiting
fd07bbc59d34: Waiting
72b73ab27698: Waiting
983308f4f0d6: Waiting
6c13f026e6da: Waiting
e5e6cd163689: Waiting
5c5516e56582: Waiting
154729f6ba86: Waiting
31213de2d322: Waiting
aef10437549e: Waiting
```

Kuvio 21. Docker WordPress image pull lataa tiedostoja

```
dev-acc@sa-srv:~$ sudo docker pull openrekisteri.com:443/ryhma8_mysql
Using default tag: latest
latest: Pulling from ryhma8_mysql
d519e2592276: Downloading 2.228MB/26.71MB
d22d2dfcfa9c: Download complete
b3afe92c540b: Download complete
b66e1d55d718: Download complete
68bbd8eb6c49: Downloading 4.538MB/4.811MB
ba484d25fdf8: Download complete
f3721839f3a0: Download complete
0624c9c3aad2: Waiting
075626307450: Waiting
69098fc75611: Waiting
ab7243881889: Waiting
8634d847d477: Waiting
3d201307e5c6: Waiting
9454dfd86ead: Waiting
```

Kuvio 22. Docker MySQL image pull lataa tiedostoja

```
dev-acc@sa-srv: ~
dev-acc@sa-srv:~$ sudo docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
openrekisteri.com:443/ryhma8_wp	latest	6d3b624d29d8	2 hours ago	490MB
openrekisteri.com:443/ryhma8_mysql	latest	220b02364a2f	3 hours ago	352MB

```
dev-acc@sa-srv:~$
```

Kuvio 23. Imaget ladattu

Käynnistimme imaget komennoilla:

```
$ sudo docker run \
  -dp 3306:3306 \
  -v mysql_conf:/etc/mysql/conf.d/ \
  -v mysql_data:/var/lib/mysql:rw \
  --name myslikontti \
  openrekisteri.com:443/ryhma8_mysql
```

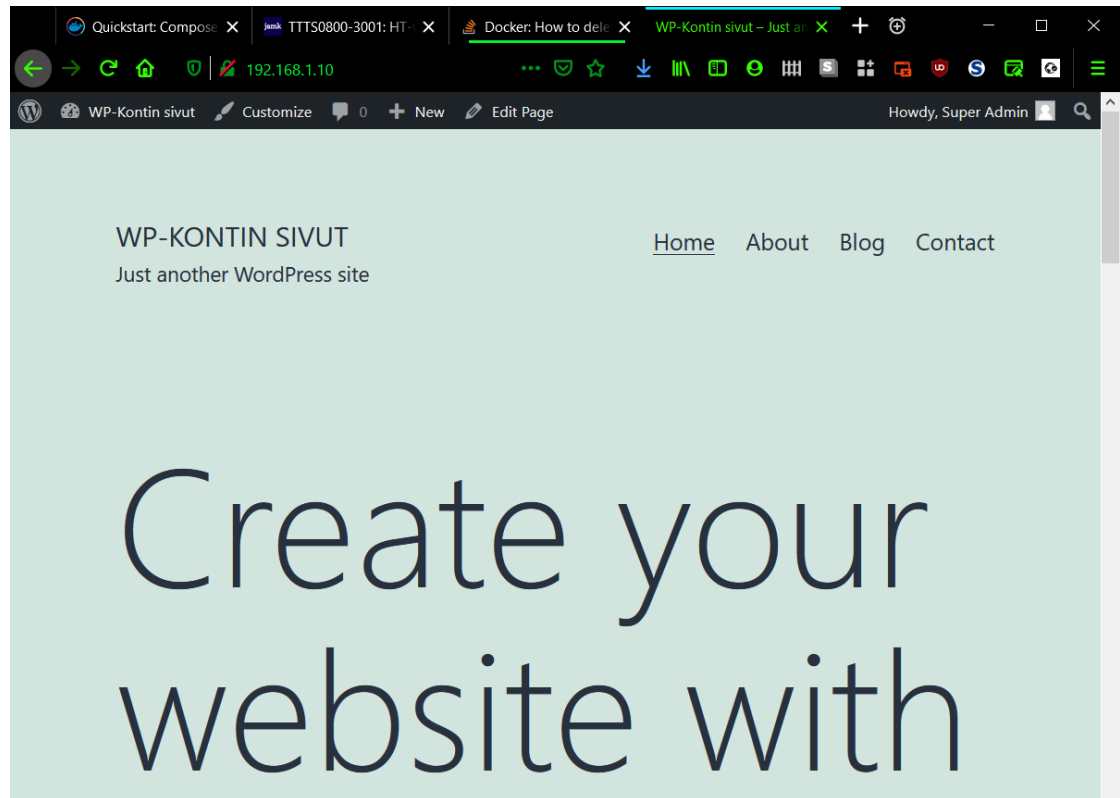
ja

```
$ sudo docker run \
  -dp 80:80 \
  -v wp:/var/www/html \
  --name wpkontti \
  openrekisteri.com:443/ryhma8_wp
```

```
dev-acc@sa-srv:~$ sudo docker ps -a
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS                               NAMES
0fff61584ec29   openrekisteri.com:443/ryhma8_wp    "/bin/sh -c /wp.sh"    2 minutes ago   Up 2 minutes   0.0.0.0:80->80/tcp, 443/tcp        wpkontti
64dd332e495     openrekisteri.com:443/ryhma8_mysql  "docker-entrypoint.s..." 2 minutes ago   Up 2 minutes   0.0.0.0:3306->3306/tcp            mysqlkontti
```

Kuvio 24. Imaget ladattu ja laitetty pyörimään

Tarkistimme vielä, että WordPress on käynnissä.



Kuvio 25. Palvelu toiminnassa

### 2.2.1 Kommentit Vaiheeseen 2 liittyen

Meillä oli ongelmia saada Docker toimimaan vaaditulla sertifikaatilla. Ongelmaksi paljastui loppupelissä se, että Docker oli asennettu *snap* -paketinhallintatyökalulla. Emme saaneet sertifikaatteja toimimaan, vaikka toimimme sertifikaatin asennuksessa *snap*in ohjeiden mukaisesti. Lopulta poistimme Dockerin *snap*-asennuksen ja latsimme Dockerin uudelleen *apt*-työkalulla. Sen jälkeen kaikki toimi ongelmitta.



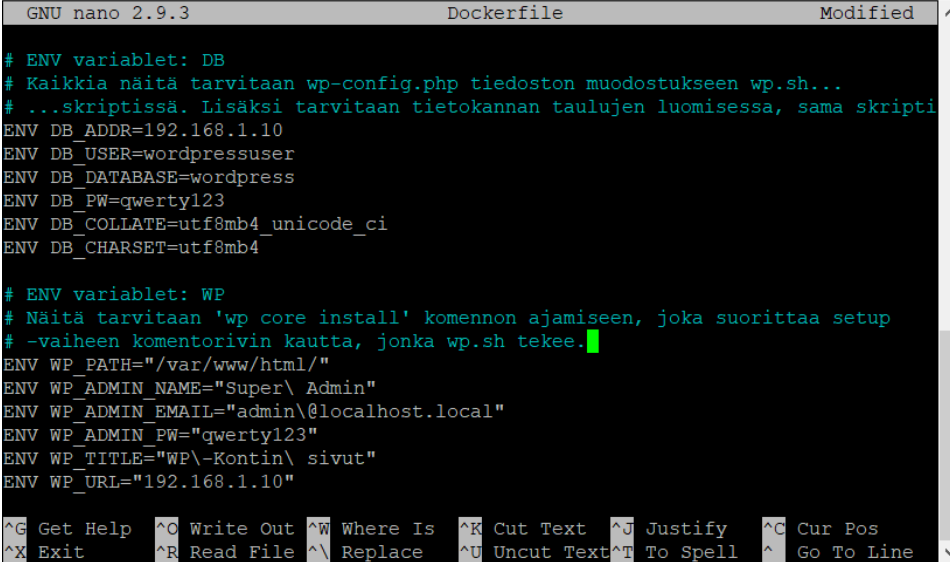
## 2.3 Vaihe 3

Teimme tätä vaihetta varten erillisen *wp.sh* -skriptin, joka automatisoi seuraavat asiat:

- Volumen alustus.
- *wp-cli* -työkalun asennus.
- *wp-config.php* -tiedoston luonti ja oikeiden arvojen asetus *wp-cli* -työkalulla.
- Taulujen luonti tietokantaan.
- WordPressin setupin automaattinen läpikäyminen (admin-tilin luominen jne.)
- Muustosskenaariona WordPressin teeman vaihtaminen.

Eli Bash-skripti hoitaa muutosskenaarion toteutuksen meidän työssämme.

Bash-skriptiä varten piti myös muuttaa WordPressin Dockerfilen sisältöä:



```
GNU nano 2.9.3 Dockerfile Modified
# ENV variabellet: DB
# Kaikkia näitä tarvitaan wp-config.php tiedoston muodostukseen wp.sh...
# ...skriptissä. Lisäksi tarvitaan tietokannan taulujen luomisessa, sama skripti
ENV DB_ADDR=192.168.1.10
ENV DB_USER=wordpressuser
ENV DB_DATABASE=wordpress
ENV DB_PW=qwerty123
ENV DB_COLLATE=utf8mb4_unicode_ci
ENV DB_CHARSET=utf8mb4

# ENV variabellet: WP
# Näitä tarvitaan 'wp core install' komennon ajamiseen, joka suorittaa setup
# -vaiheen komentorivin kautta, jonka wp.sh tekee.
ENV WP_PATH="/var/www/html/"
ENV WP_ADMIN_NAME="Super\ Admin"
ENV WP_ADMIN_EMAIL="admin@localhost.local"
ENV WP_ADMIN_PW="qwerty123"
ENV WP_TITLE="WP\ -Kontin\ sivut"
ENV WP_URL="192.168.1.10"

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line
```

Kuvio 26. WordPressin *Dockerfilen* muutokset

Bash-skripti on kuvattu seuraavissa kuvissa:

```

GNU nano 2.9.3 wp.sh

#!/bin/bash

# Installing WordPress if it does not exist.
wp_files_n=$(ls $WP_PATH | wc -l)

if [ "$wp_files_n" = "0" ]; then
    echo -e "WordPress installation directory is empty. Downloading files..."
    cd ~ && \
    curl -O https://wordpress.org/latest.tar.gz && \
    tar zxvf latest.tar.gz && \
    mv wordpress/* /var/www/html && \
    chown -R www-data: /var/www/html/ && \
    rm -rf latest.tar.gz && rm -rf wordpress
    # Pohdintaa: Wordpress chmod asetukset? Jäävät nyt määrittämättä.
    # Esim. setgid, että uudet tiedostot päivittyvät automaattisesti...
else
    echo -e "Found Wordpress installation on $WP_PATH. Skipping installation..."
fi

# Installing wp-cli if it does not exist
WPCLI=/usr/local/bin/wp
if test -f "$WPCLI"; then
    echo -e "$WPCLI exists. Skipping configuration..."
else
    echo -e "$WPCLI does not exist. Installing..."
    cd ~ && \
    curl -O https://raw.githubusercontent.com/wp-cli/builds/gh-pages/phar/wp-cli.phar && \
    chmod ug+x wp-cli.phar && \
    mv wp-cli.phar /usr/local/bin/wp && \
    chown www-data:www-data /usr/local/bin/wp
    # debug-komento:
    # wp --info
fi

# Sleep so mysql could initialize to avoid race condition
sleep 5

```

Kuvio 27. WordPressin ja WP-CLI:n automaattinen asennus

```

# Creating wp-config.php if it does not exist
WPCONF=/var/www/html/wp-config.php
if test -f "$WPCONF"; then
    echo -e "$WPCONF exists. Skipping configuration..."
else
    # This MUST be run as root for some reason. Else it fails.
    echo -e "$WPCONF does not exist. Creating a configuration file..."
    cd $WP_PATH && \
    wp config create \
    --dbname=$DB_DATABASE \
    --dbuser=$DB_USER \
    --dbpass=$DB_PW \
    --dbhost=$DB_ADDR \
    --dbcharset=$DB_CHARSET \
    --dbcollate=$DB_COLLATE \
    --allow-root
fi

```

Kuvio 28. *wp-config.php* -tiedoston automaattinen täydennys WP-CLI työkalulla käyttäen Dockerfilen *env variableja*

```

# Checking if database is populated. If not, let's initiate tables.
# We will not check if database exists, because it gets created by our MySQL Docker...
db_table_n=$(mysql \
  -h $DB_ADDR \
  -u $DB_USER \
  -s --skip-column-names \
  -e "SELECT COUNT(DISTINCT table_name) \
      FROM information_schema.columns \
      WHERE table_schema = '$DB_DATABASE'" \
  -p$DB_PW)
if [ "$db_table_n" = "0" ]; then
  echo "Database is empty. Creating tables..." ; \
  su wordpressuser -c \
  "cd $WP_PATH && \
  wp core install \
  --url=$WP_URL \
  --title=$WP_TITLE \
  --admin_user=$WP_ADMIN_NAME \
  --admin_password=$WP_ADMIN_PW \
  --admin_email=$WP_ADMIN_EMAIL"
else
  echo -e "Database tables exist ($db_table_n tables). Skipping table creation..."
fi

```

Kuvio 29. Taulujen automaattinen luonti tietokantaan WP-CLI työkalun avulla käyttäen Dockerfilen *env variableja*

```

# muutosskenaario
echo -e "Changing WordPress theme..."
cd $WP_PATH && \
wp theme install graphene \
--activate \
--allow-root

# set apache as main process to prevent exit
echo -e "Starting main process..."
/usr/sbin/apache2ctl -D FOREGROUND;

```

^G Get Help    ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify    ^C Cur Pos  
 ^X Exit        ^R Read File    ^\ Replace    ^U Uncut Text    ^T To Linter    ^\_ Go To Line

Kuvio 30. Muutosskenaariona WordPressin teeman vaihtaminen WP-CLI työkalulla

```
dev-acc@sa-srv: ~
},
"Mounts": [
  {
    "Type": "volume",
    "Name": "mysql_conf",
    "Source": "/var/lib/docker/volumes/mysql_conf/_data",
    "Destination": "/etc/mysql/conf.d",
    "Driver": "local",
    "Mode": "z",
    "RW": true,
    "Propagation": ""
  },
  {
    "Type": "volume",
    "Name": "mysql_data",
    "Source": "/var/lib/docker/volumes/mysql_data/_data",
    "Destination": "/var/lib/mysql",
    "Driver": "local",
    "Mode": "rw",
    "RW": true,
    "Propagation": ""
  }
],
"Config": {
```

Kuvio 31. MySQL-kontin volumet

```
"Mounts": [
  {
    "Type": "volume",
    "Name": "wp",
    "Source": "/var/lib/docker/volumes/wp/_data",
    "Destination": "/var/www/html",
    "Driver": "local",
    "Mode": "z",
    "RW": true,
    "Propagation": ""
  }
],
```


Kuvio 32. WordPressin volumet

```
dev-acc@sa-srv:~$ sudo docker ps -a
CONTAINER ID        IMAGE               PORTS              COMMAND                NAMES
CREATED            STATUS              5 days ago        Exited (0) 5 days ago  wpkonttidemo
96826a8b2d4b
3aa5d7cd756f       openrekisteri.com:443/ryhma8_wp  "docker-php-entrypoi..."  wpkontti
11 days ago        Exited (128) 6 days ago
bb1f938961e6       openrekisteri.com:443/ryhma8_mysql  "docker-entrypoint.s..."  myslikontti
11 days ago        Exited (0) 6 minutes ago
```

Kuvio 33. Kontit pyörivät virheettä

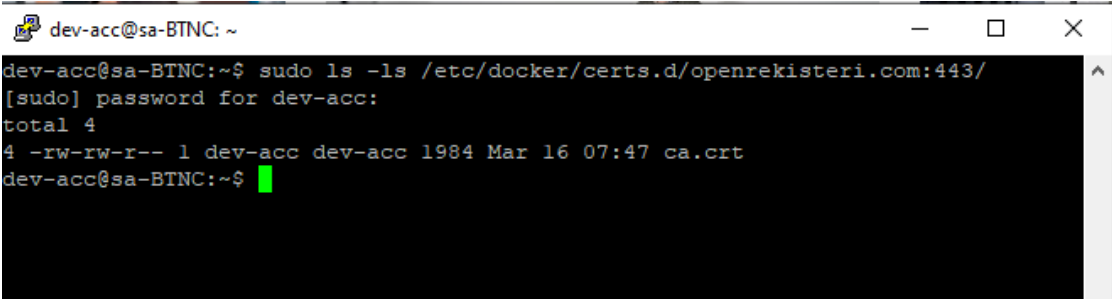
## 2.4 Vaihe 4

Luotiin uusi virtuaalikone, johon lisättiin opettajan cert-tiedosto, jotta voimme ladata edellisissä vaiheissa luodut imaget opettajan repositoriosta.

A terminal window titled 'dev-acc@sa-BTNC: ~' showing the /etc/hosts file being edited with GNU nano 2.9.3. The file contains three entries: 27.0.0.1 localhost, 127.0.1.1 sa-BTNC, and 35.228.57.182 openrekisteri.com. Below these, there is a comment about IPv6 and several lines for IPv6 addresses and their corresponding hostnames.

```
dev-acc@sa-BTNC: ~  
GNU nano 2.9.3 /etc/hosts  
27.0.0.1 localhost  
127.0.1.1 sa-BTNC  
35.228.57.182 openrekisteri.com  
  
# The following lines are desirable for IPv6 capable hosts  
::1 ip6-localhost ip6-loopback  
fe00::0 ip6-localnet  
ff00::0 ip6-mcastprefix  
ff02::1 ip6-allnodes  
ff02::2 ip6-allrouters
```

Kuvio 34. Uuden koneen */etc/hosts* tiedosto

A terminal window titled 'dev-acc@sa-BTNC: ~' showing the output of the command 'sudo ls -ls /etc/docker/certs.d/openrekisteri.com:443/'. It shows a directory listing for 'ca.crt' with permissions -rw-rw-r-- and a size of 1984 bytes.

```
dev-acc@sa-BTNC:~$ sudo ls -ls /etc/docker/certs.d/openrekisteri.com:443/  
[sudo] password for dev-acc:  
total 4  
4 -rw-rw-r-- 1 dev-acc dev-acc 1984 Mar 16 07:47 ca.crt  
dev-acc@sa-BTNC:~$
```

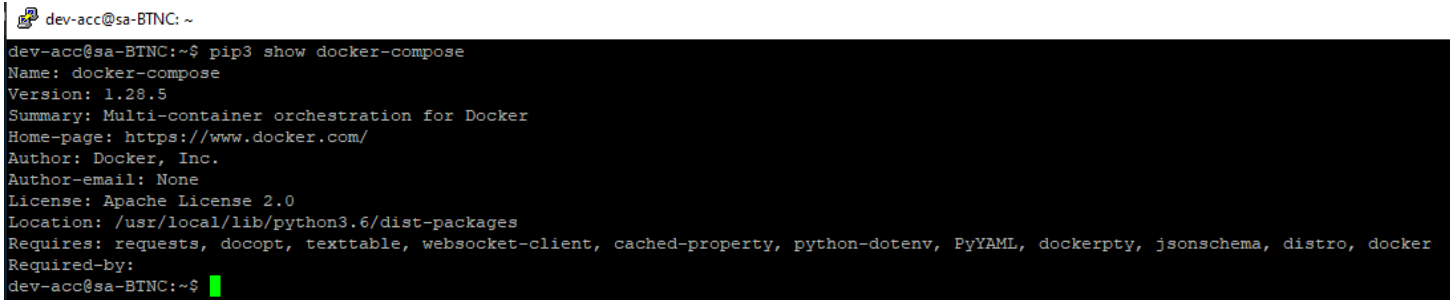
Kuvio 35. crt-tiedosto lisätty uuteen koneeseen

Tämän jälkeen asennettiin pip3 komennolla:

```
$ sudo apt install python3-pip
```

jonka jälkeen saatiin asennettua *docker-compose* komennolla:

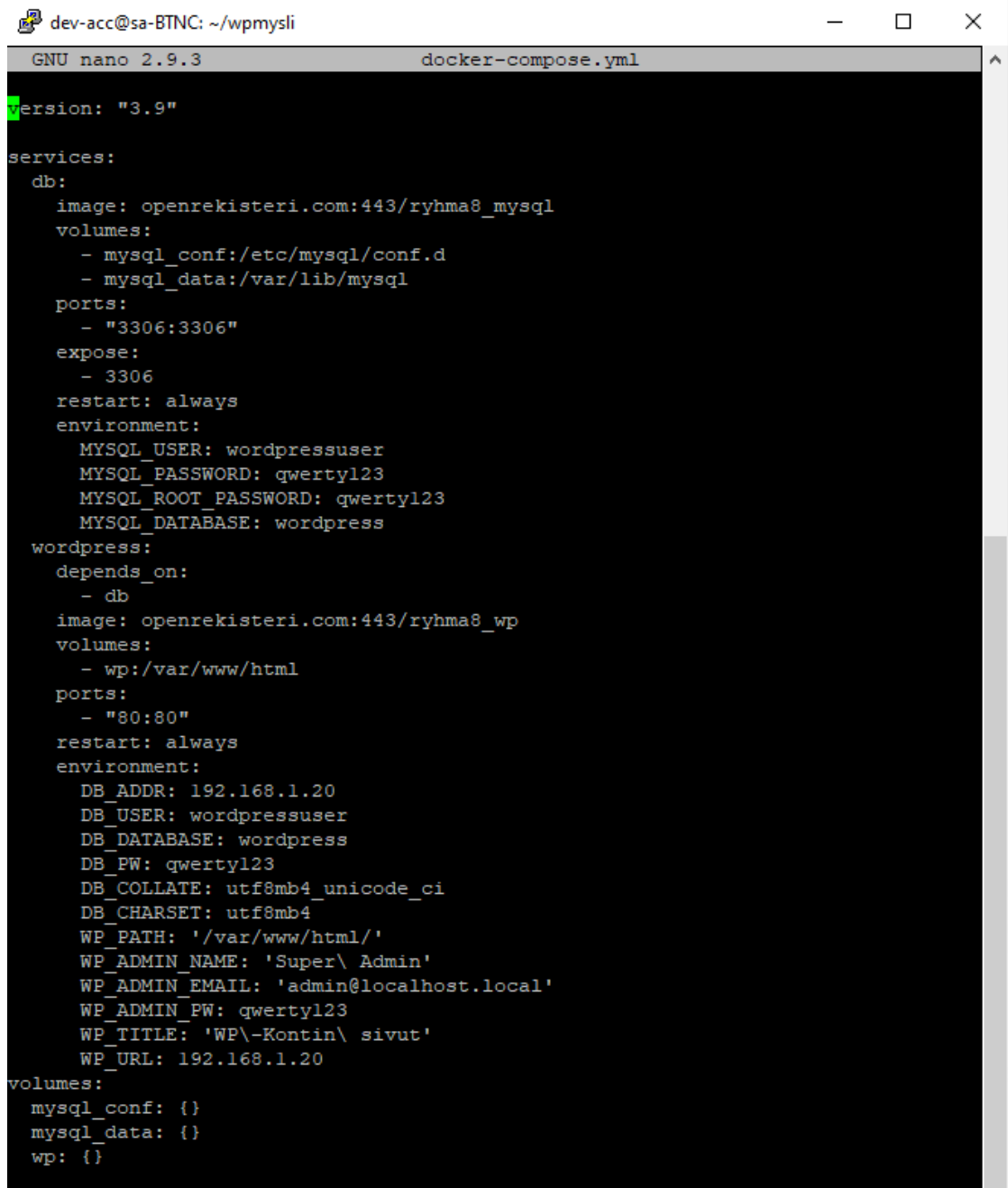
```
$ sudo pip3 install docker-compose (Install Docker Compose n.d).
```



```
dev-acc@sa-BTNC: ~  
dev-acc@sa-BTNC:~$ pip3 show docker-compose  
Name: docker-compose  
Version: 1.28.5  
Summary: Multi-container orchestration for Docker  
Home-page: https://www.docker.com/  
Author: Docker, Inc.  
Author-email: None  
License: Apache License 2.0  
Location: /usr/local/lib/python3.6/dist-packages  
Requires: requests, docopt, texttable, websocket-client, cached-property, python-dotenv, PyYAML, dockerpty, jsonschema, distro, docker  
Required-by:  
dev-acc@sa-BTNC:~$
```

Kuvio 36. *docker-compose* asennettu onnistuneesti

Tämän jälkeen loimme uuden kansion '*wpmysli*' johon teimme *docker-compose.yml* tiedoston. Tähän otettiin mallia dockerin omista dokumentaatiosta (Quickstart: Compose and WordPress n.d.). Tiedostossa määritellään kaksi palvelua db ja wordpress, sekä luodaan tarvittavat volumet. Palveluiden sisällä hyödynnettiin *env\_variable*ja jotka luotiin aiemmissa vaiheissa:



```

GNU nano 2.9.3          docker-compose.yml

version: "3.9"

services:
  db:
    image: openrekisteri.com:443/ryhma8_mysql
    volumes:
      - mysql_conf:/etc/mysql/conf.d
      - mysql_data:/var/lib/mysql
    ports:
      - "3306:3306"
    expose:
      - 3306
    restart: always
    environment:
      MYSQL_USER: wordpressuser
      MYSQL_PASSWORD: qwerty123
      MYSQL_ROOT_PASSWORD: qwerty123
      MYSQL_DATABASE: wordpress
  wordpress:
    depends_on:
      - db
    image: openrekisteri.com:443/ryhma8_wp
    volumes:
      - wp:/var/www/html
    ports:
      - "80:80"
    restart: always
    environment:
      DB_ADDR: 192.168.1.20
      DB_USER: wordpressuser
      DB_DATABASE: wordpress
      DB_PW: qwerty123
      DB_COLLATE: utf8mb4_unicode_ci
      DB_CHARSET: utf8mb4
      WP_PATH: '/var/www/html/'
      WP_ADMIN_NAME: 'Super\ Admin'
      WP_ADMIN_EMAIL: 'admin@localhost.local'
      WP_ADMIN_PW: qwerty123
      WP_TITLE: 'WP\~Kontin\ sivut'
      WP_URL: 192.168.1.20
volumes:
  mysql_conf: {}
  mysql_data: {}
  wp: {}

```

Kuvio 37. Docker-compose tiedosto

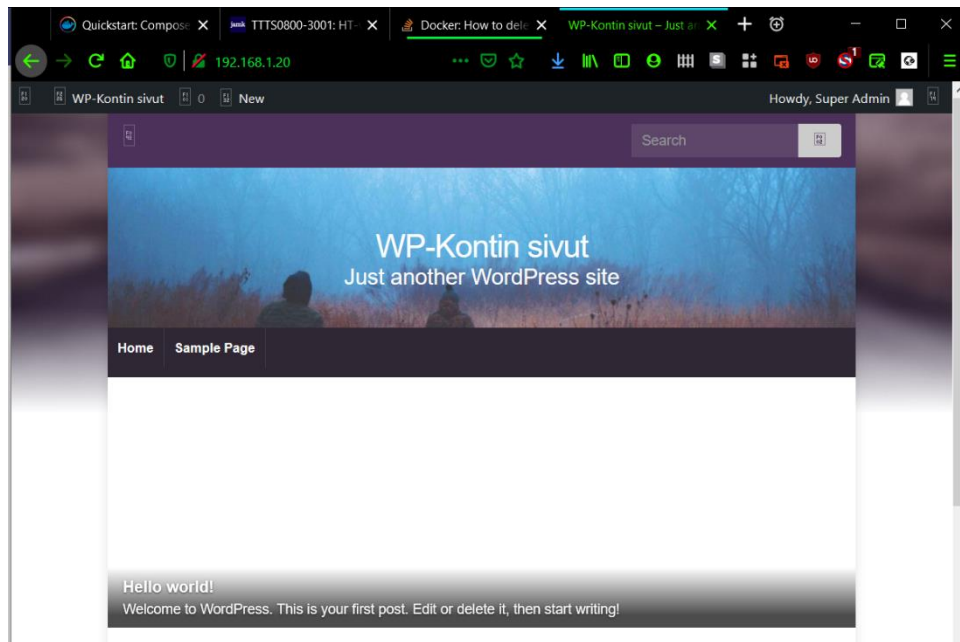
Lopulta kun compose-tiedosto oli valmis, projekti voitiin buildata komennolla:

```
$ docker-compose up -d
```

Tämän jälkeen tarkistettiin, että kontit ovat käynnissä ja onko palvelu käynnissä osoitteessa 192.168.1.20.

```
dev-acc@sa-BTNC: ~  
dev-acc@sa-BTNC:~$ sudo docker ps -a  
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS                               NAMES  
c4e15b018b2f   openrekisteri.com:443/ryhma8_wp    "/bin/sh -c /wp.sh"     3 hours ago   Up 3 hours   0.0.0.0:80->80/tcp, 443/tcp        wpmysql_wordpress_1  
cf7ee3193fc8   openrekisteri.com:443/ryhma8_mysql "docker-entrypoint.s..." 3 hours ago   Up 3 hours   0.0.0.0:3306->3306/tcp            wpmysql_db_1  
164b422afe23   hello-world                          "/hello"                 6 days ago    Exited (0) 6 days ago                                nifty_maxwell  
dev-acc@sa-BTNC:~$
```

Kuvio 38. Kontit käynnissä uudella koneella



Kuvio 39. Palvelut käynnissä uudella koneella



### 3 Lähteet

Florian (nimimerkki). 2016. Vastaus "How to install extension for php via docker-php-ext-install?" keskusteluketjun avaukseen Stack Overflow sivustolla. Julkaistu 30.5.2016. Viitattu 16.3.2021. <https://stackoverflow.com/a/37527960>.

Install Docker Compose. N.d. Dockerin ohjeet, kuinka Docker Compose asennetaan. Viitattu 16.3.2021. <https://docs.docker.com/compose/install/>.

MariaDB. 2021. Viitattu 16.3.2021. [https://hub.docker.com/\\_/mariadb](https://hub.docker.com/_/mariadb).

PHP. 2021. Viitattu 16.3.2021. [https://hub.docker.com/\\_/php](https://hub.docker.com/_/php).

Quickstart: Compose and WordPress. N.d. Dockerin omat pikaohjeet, kuinka composerilla voi ottaa WordPressin käyttöön. Viitattu 16.3.2021. <https://docs.docker.com/compose/wordpress/>.