

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Лабораторная работа №4

Знакомство с Docker

Группа: Р34082

Выполнила: Савельева Д.А.

Проверил:

к.т.н. преподаватель Белозубов А.В.

Санкт-Петербург

2024г.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
1 РАБОТА С DOCKER	4
ЗАКЛЮЧЕНИЕ	15
СПИСОК ИСТОЧНИКОВ.....	16

ВВЕДЕНИЕ

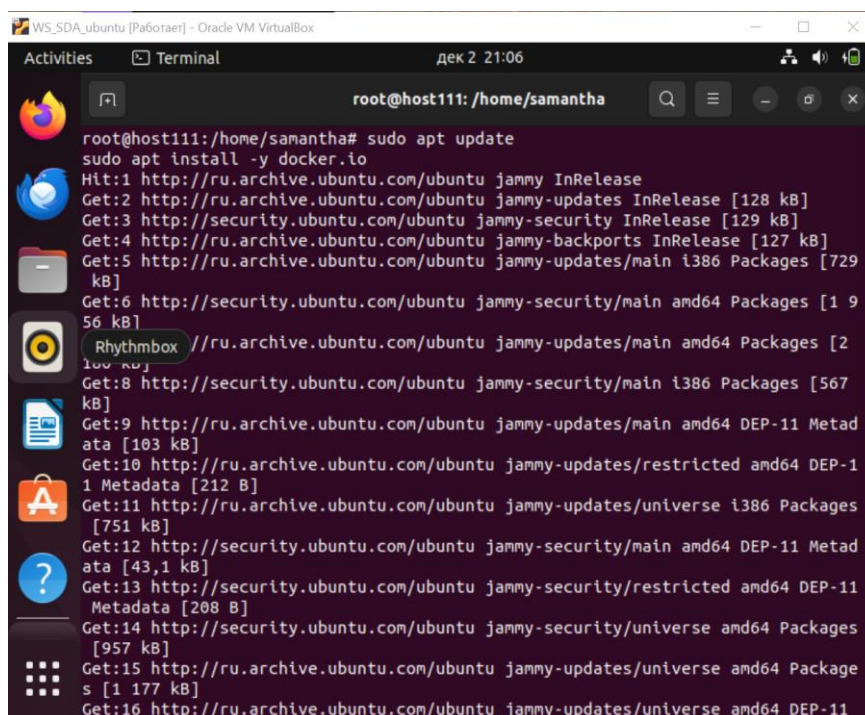
В данной лабораторной работе изучаются основы работы с контейнерами с использованием Docker — популярной платформы контейнеризации, которая обеспечивает удобство создания, развертывания и управления приложениями в изолированных окружениях.

Docker позволяет запускать приложения и их зависимости в одном окружении, что облегчает их переносимость и стабильность в различных средах. В рамках выполнения работы также задействуются инструменты, такие как MariaDB (реляционная база данных), Nginx (веб-сервер), NextCloud (система облачного хранения данных) и phpVirtualBox (веб-интерфейс для управления виртуальными машинами VirtualBox).

Цель лабораторной работы — изучение основ работы с Docker и инструментов контейнеризации, овладение навыками создания, запуска и управления контейнерами для развертывания многокомпонентных приложений.

1 РАБОТА С DOCKER

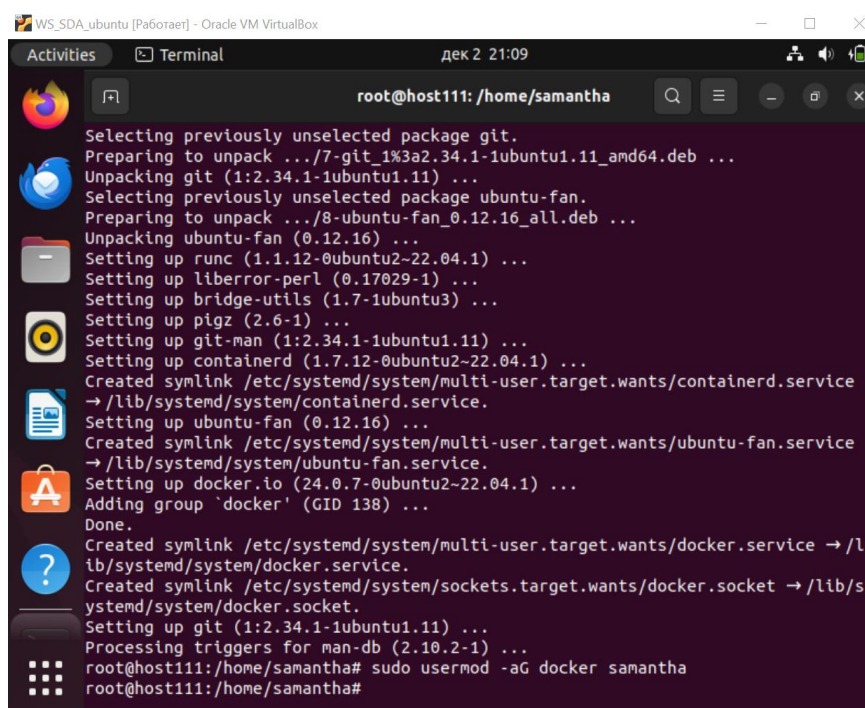
Для начала установим Docker и VirtualBox в WS_SDA_ubuntu. На рисунке 1 представлены команды для выполнения данных действий. На рисунке 2 представлено добавления пользователя в группу docker.



```
WS_SDA_ubuntu [Работаer] - Oracle VM VirtualBox
Activities Terminal дек 2 21:06
root@host111: /home/samantha

root@host111:/home/samantha# sudo apt update
sudo apt install -y docker.io
Hit:1 http://ru.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ru.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:4 http://ru.archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:5 http://ru.archive.ubuntu.com/ubuntu jammy-updates/main i386 Packages [729
kB]
Get:6 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1 9
56 kB]
Get:7 http://ru.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2
100 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/main i386 Packages [567
kB]
Get:9 http://ru.archive.ubuntu.com/ubuntu jammy-updates/main amd64 DEP-11 Metad
ata [103 kB]
Get:10 http://ru.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 DEP-1
1 Metadata [212 B]
Get:11 http://ru.archive.ubuntu.com/ubuntu jammy-updates/universe i386 Packages
[751 kB]
Get:12 http://security.ubuntu.com/ubuntu jammy-security/main amd64 DEP-11 Metad
ata [43,1 kB]
Get:13 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 DEP-11
Metadata [208 B]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages
[957 kB]
Get:15 http://ru.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Package
s [1 177 kB]
Get:16 http://ru.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 DEP-11
```

Рисунок 1 – Установка Docker

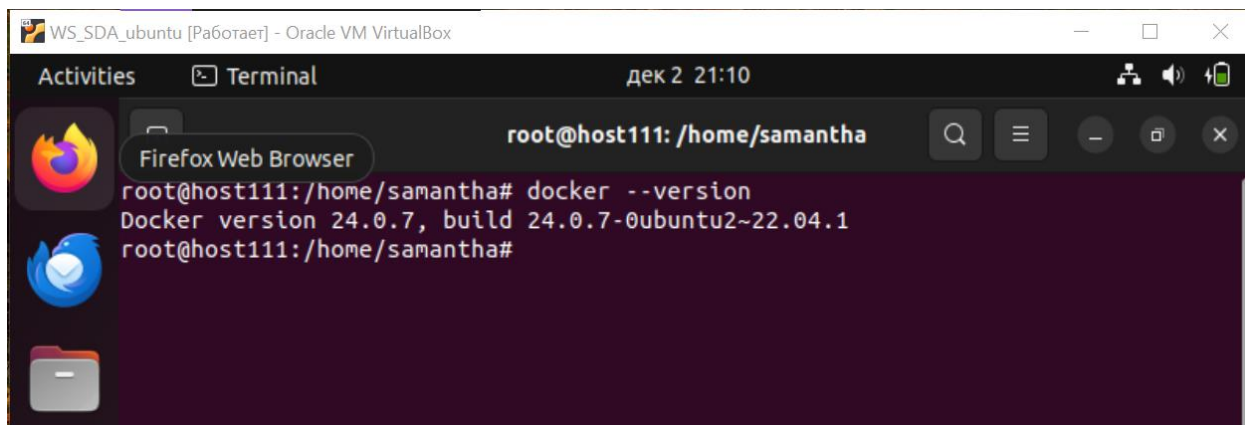


```
WS_SDA_ubuntu [Работаer] - Oracle VM VirtualBox
Activities Terminal дек 2 21:09
root@host111: /home/samantha

Selecting previously unselected package git.
Preparing to unpack .../7-git_1%3a2.34.1-1ubuntu1.11_amd64.deb ...
Unpacking git (1:2.34.1-1ubuntu1.11) ...
Selecting previously unselected package ubuntu-fan.
Preparing to unpack .../8-ubuntu-fan_0.12.16_all.deb ...
Unpacking ubuntu-fan (0.12.16) ...
Setting up runc (1.1.12-0ubuntu2-22.04.1) ...
Setting up liberror-perl (0.17029-1) ...
Setting up bridge-utils (1.7-1ubuntu3) ...
Setting up pigz (2.6-1) ...
Setting up git-man (1:2.34.1-1ubuntu1.11) ...
Setting up containerd (1.7.12-0ubuntu2-22.04.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service
→ /lib/systemd/system/containerd.service.
Setting up ubuntu-fan (0.12.16) ...
Created symlink /etc/systemd/system/multi-user.target.wants/ubuntu-fan.service
→ /lib/systemd/system/ubuntu-fan.service.
Setting up docker.io (24.0.7-0ubuntu2-22.04.1) ...
Adding group 'docker' (GID 138) ...
Done.
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /l
ib/systemd/system/docker.service.
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/s
ystemd/system/docker.socket.
Setting up git (1:2.34.1-1ubuntu1.11) ...
Processing triggers for man-db (2.10.2-1) ...
root@host111:/home/samantha# sudo usermod -aG docker samantha
root@host111:/home/samantha#
```

Рисунок 2 – Назначение группы docker пользователю Samantha

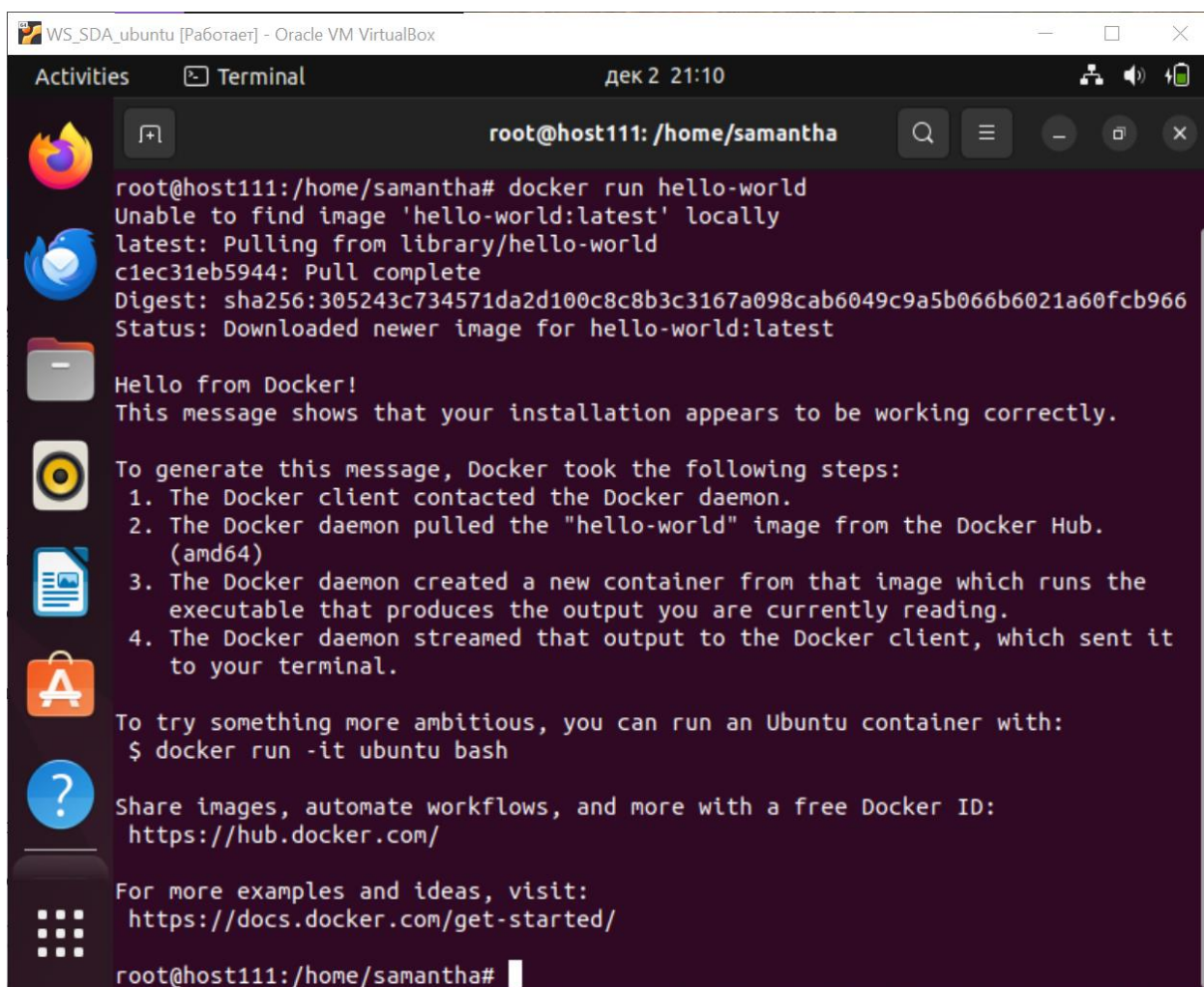
На рисунке 3 представлена проверка версии установленного на виртуальную машину Docker.



```
WS_SDA_ubuntu [Работает] - Oracle VM VirtualBox
Activities Terminal дек 2 21:10
root@host111: /home/samantha
Firefox Web Browser
root@host111:/home/samantha# docker --version
Docker version 24.0.7, build 24.0.7-0ubuntu2~22.04.1
root@host111:/home/samantha#
```

Рисунок 3 – Проверка версии Docker

На рисунке 4 представлен запуск контейнера hello-world.



```
WS_SDA_ubuntu [Работает] - Oracle VM VirtualBox
Activities Terminal дек 2 21:10
root@host111: /home/samantha
root@host111:/home/samantha# docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
c1ec31eb5944: Pull complete
Digest: sha256:305243c734571da2d100c8c8b3c3167a098cab6049c9a5b066b6021a60fcb966
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

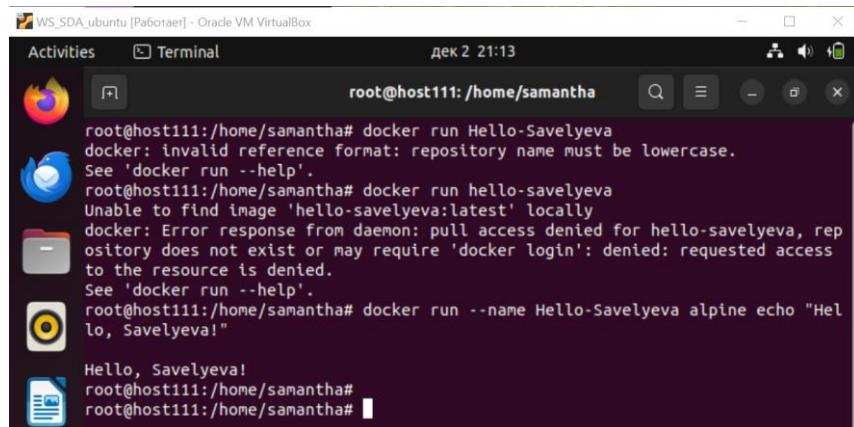
To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/
root@host111:/home/samantha#
```

Рисунок 4 – Запуск контейнера hello-world

На рисунке 5 представлены успешные и неудавшиеся попытки запуска контейнера Hello-Savelyeva. После неудачной попытки пытаемся запустить, указав новые параметры для docker run.

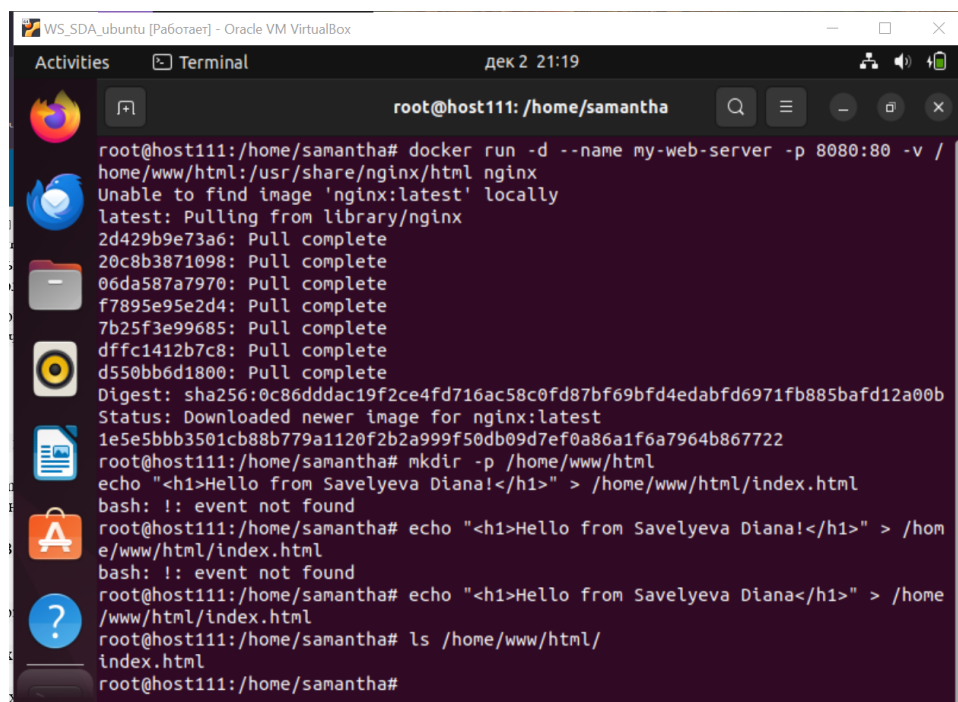


```
WS_SDA_ubuntu [Работает] - Oracle VM VirtualBox
Activities Terminal дек 2 21:13
root@host111: /home/samantha

root@host111:/home/samantha# docker run Hello-Savelyeva
docker: invalid reference format: repository name must be lowercase.
See 'docker run --help'.
root@host111:/home/samantha# docker run hello-savelyeva
Unable to find image 'hello-savelyeva:latest' locally
docker: Error response from daemon: pull access denied for hello-savelyeva, repository does not exist or may require 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.
root@host111:/home/samantha# docker run --name Hello-Savelyeva alpine echo "Hello, Savelyeva!"
Hello, Savelyeva!
root@host111:/home/samantha#
root@host111:/home/samantha#
```

Рисунок 5 – Запуск контейнера, указанного в задании

Далее осуществляем запуск контейнера с Nginx, указав в параметрах запуска порт 8080, установив рабочий каталог на хосте /home/www/html. На рисунке 6 представлена работа с Nginx контейнером. После запуска контейнера в директорию /home/www/html необходимо положить файл, который содержит страницу, по умолчанию содержащую личную информацию.



```
WS_SDA_ubuntu [Работает] - Oracle VM VirtualBox
Activities Terminal дек 2 21:19
root@host111: /home/samantha

root@host111:/home/samantha# docker run -d --name my-web-server -p 8080:80 -v /home/www/html:/usr/share/nginx/html nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
2d429b9e73a6: Pull complete
20c8b3871098: Pull complete
06da587a7970: Pull complete
f7895e95e2d4: Pull complete
7b25f3e99685: Pull complete
dffc1412b7c8: Pull complete
d550bb6d1800: Pull complete
Digest: sha256:0c86dddac19f2ce4fd716ac58c0fd87bf69bfd4edabfd6971fb885bafd12a00b
Status: Downloaded newer image for nginx:latest
1e5e5bbb3501cb88b779a1120f2b2a999f50db09d7ef0a86a1f6a7964b867722
root@host111:/home/samantha# mkdir -p /home/www/html
echo "<h1>Hello from Savelyeva Diana!</h1>" > /home/www/html/index.html
bash: !: event not found
root@host111:/home/samantha# echo "<h1>Hello from Savelyeva Diana!</h1>" > /home/www/html/index.html
bash: !: event not found
root@host111:/home/samantha# echo "<h1>Hello from Savelyeva Diana</h1>" > /home/www/html/index.html
root@host111:/home/samantha# ls /home/www/html/
index.html
root@host111:/home/samantha#
```

Рисунок 6 – Запуск контейнера Nginx

На рисунке 7 представлена созданная страница с личной информацией, запущенная на localhost с помощью Nginx.

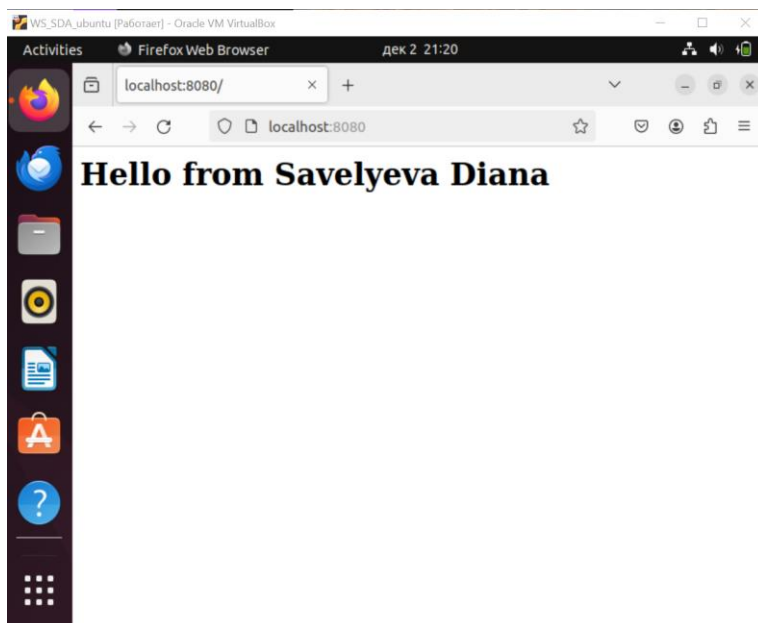


Рисунок 7 – Страница Nginx

Далее запустим контейнер MariaDB, указав рабочий каталог /home/DB, а также оставив порт по умолчанию 3306. Далее на рисунках 8 и 9 представлена работа с контейнером, создание БД Savelyeva.

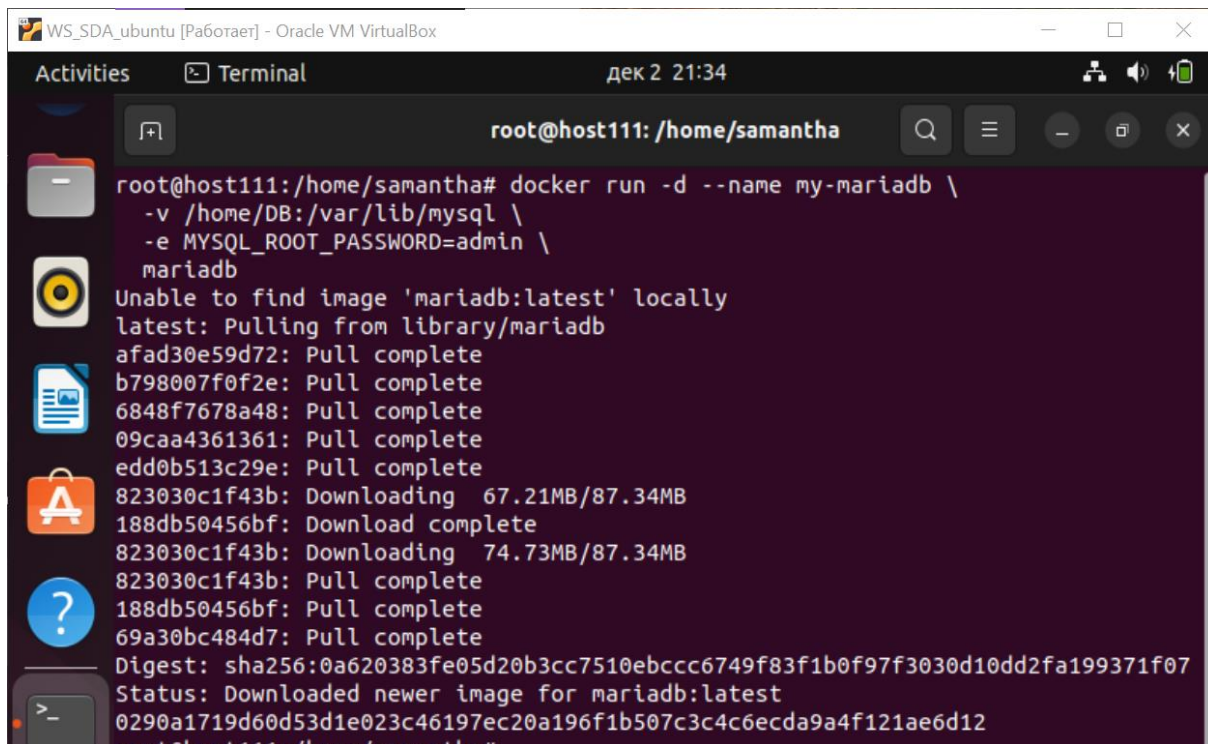
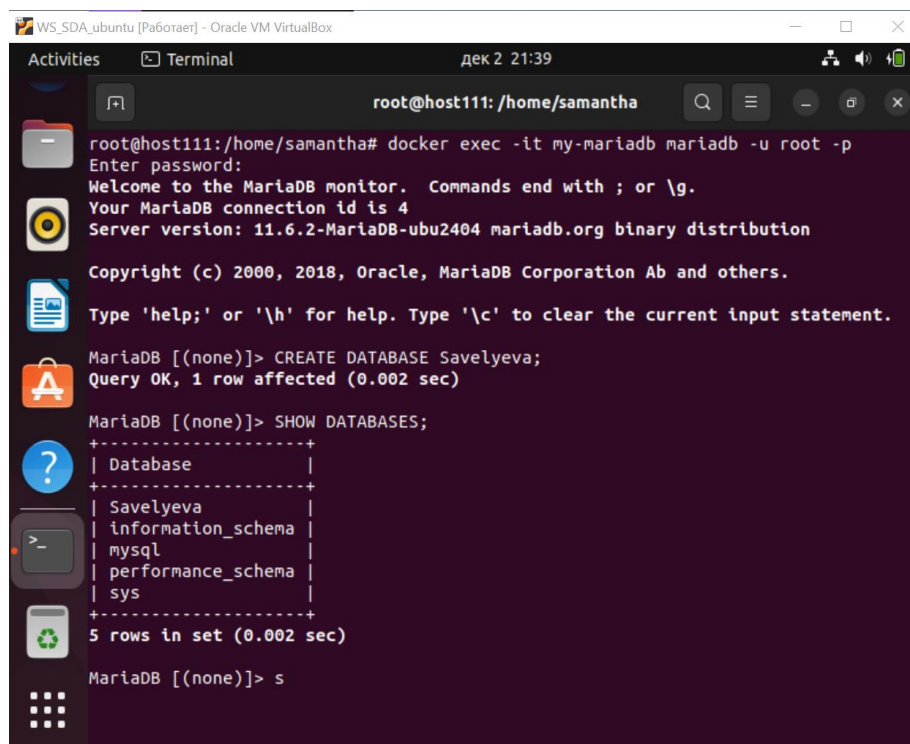


Рисунок 8 – Запуск контейнера MariaDB



```
WS_SDA_ubuntu [Работаer] - Oracle VM VirtualBox
Activities Terminal дек 2 21:39
root@host111: /home/samantha
root@host111:/home/samantha# docker exec -it my-mariadb mariadb -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 4
Server version: 11.6.2-MariaDB-ubu2404 mariadb.org binary distribution

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

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

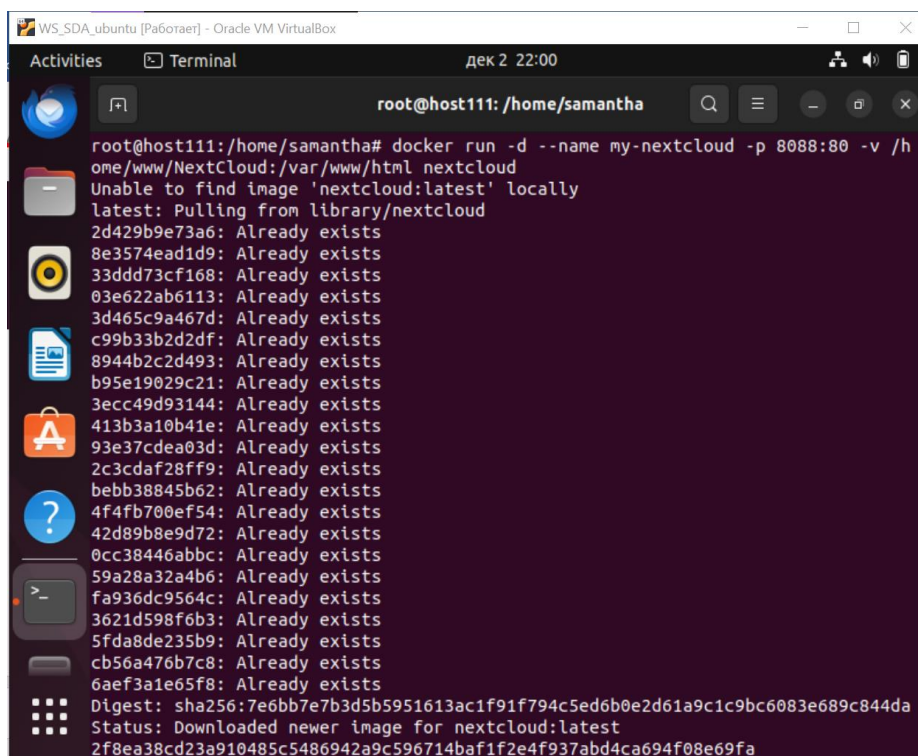
MariaDB [(none)]> CREATE DATABASE Savelyeva;
Query OK, 1 row affected (0.002 sec)

MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| Savelyeva |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.002 sec)

MariaDB [(none)]> s
```

Рисунок 9 – Создание базы данных в MariaDB

Далее запустим контейнер NextCloud, указав рабочий каталог /home/www/NextCloud, а также установив порт 8088. Далее на рисунках 10 и 11 представлена работа с контейнером, регистрация пользователей.



```
WS_SDA_ubuntu [Работаer] - Oracle VM VirtualBox
Activities Terminal дек 2 22:00
root@host111: /home/samantha
root@host111:/home/samantha# docker run -d --name my-nextcloud -p 8088:80 -v /home/www/NextCloud:/var/www/html nextcloud
Unable to find image 'nextcloud:latest' locally
latest: Pulling from library/nextcloud
2d429b9e73a6: Already exists
8e3574ead1d9: Already exists
33ddd73cf168: Already exists
03e622ab6113: Already exists
3d465c9a467d: Already exists
c99b33b2d2df: Already exists
8944b2c2d493: Already exists
b95e19029c21: Already exists
3ecc49d93144: Already exists
413b3a10b41e: Already exists
93e37cdea03d: Already exists
2c3cdaf28ff9: Already exists
bebb38845b62: Already exists
4f4fb700ef54: Already exists
42d89b8e9d72: Already exists
0cc38446abbc: Already exists
59a28a32a4b6: Already exists
fa936dc9564c: Already exists
3621d598f6b3: Already exists
5fda8de235b9: Already exists
cb56a476b7c8: Already exists
6aef3a1e65f8: Already exists
Digest: sha256:7e6bb7e7b3d5b5951613ac1f91f794c5ed6b0e2d61a9c1c9bc6083e689c844da
Status: Downloaded newer image for nextcloud:latest
2f8ea38cd23a910485c5486942a9c596714baf1f2e4f937abd4ca694f08e69fa
```

Рисунок 10 – Запуск контейнера NextCloud

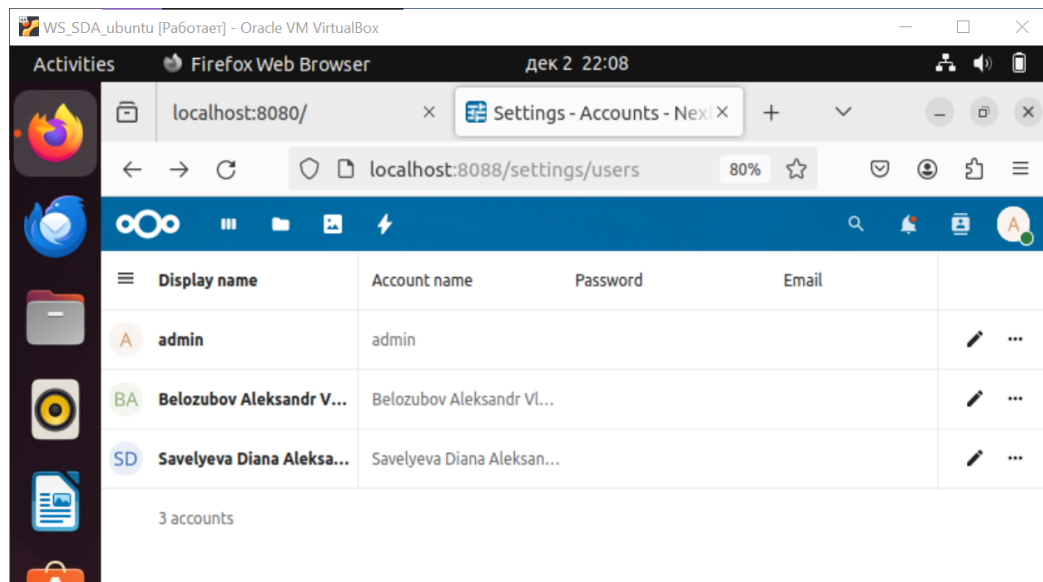


Рисунок 11 – Созданные пользователи Nextcloud

Далее устанавливаем и запускаем контейнер `phpvirtualbox`. На рисунках 12 и 13 представлена работа с контейнером, интерфейс `phpvirtualbox`.

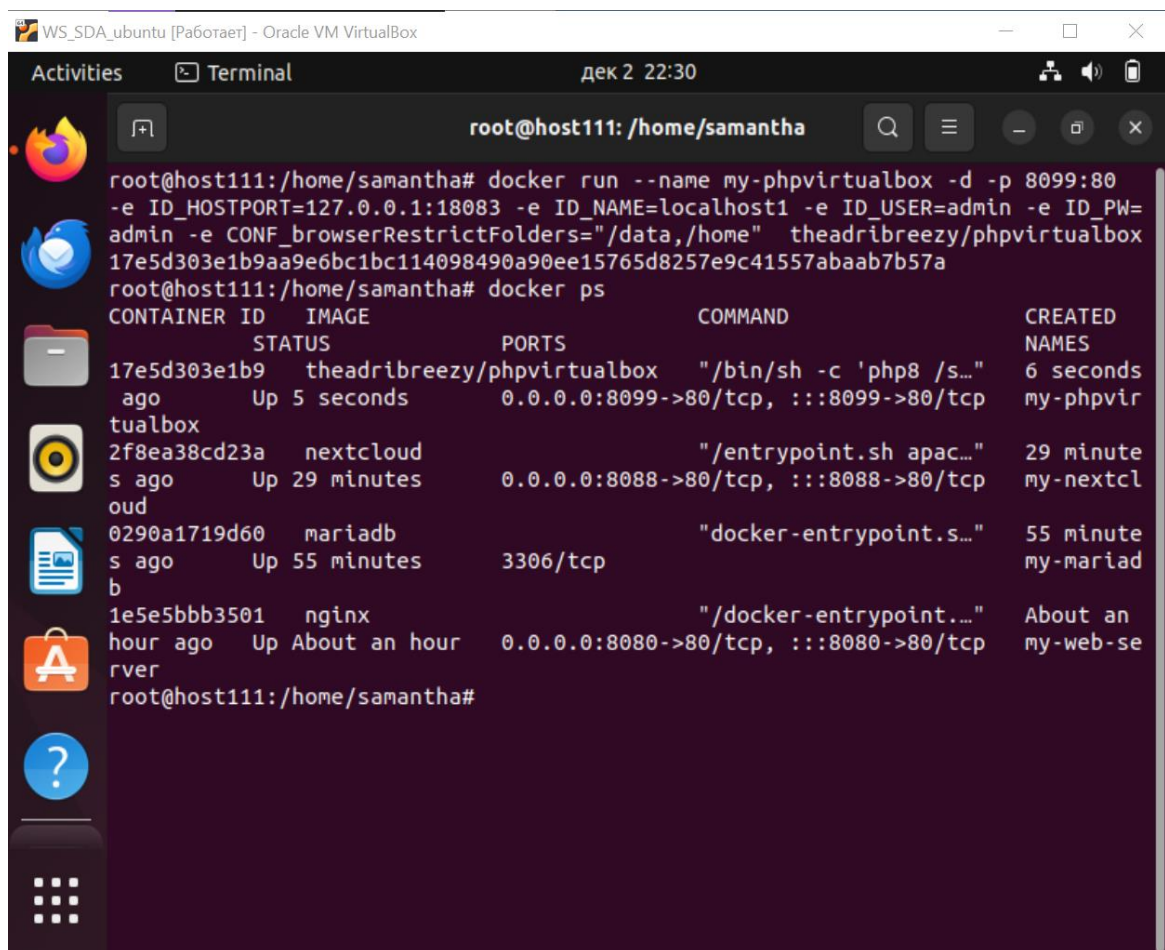


Рисунок 12 – Запуск контейнера `phpvirtualbox`

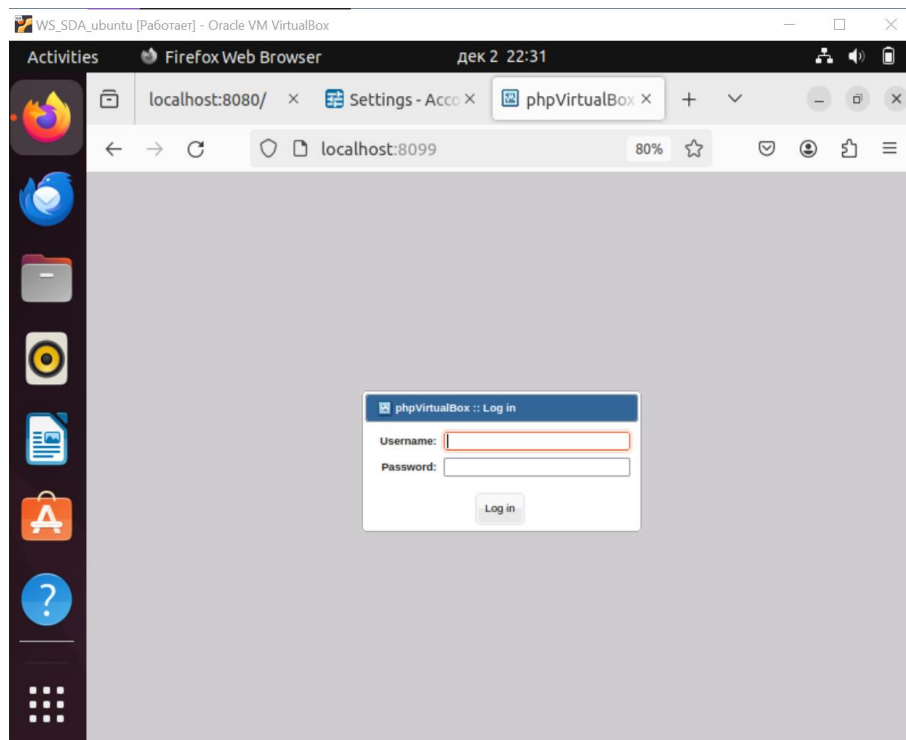


Рисунок 13 – Проверка работоспособности phpvirtualbox

Далее необходимо собрать проект Docker Compose, содержащий контейнеры NextCloud, Nginx и БД MariaDB. На рисунках 14-16 представлены этапы работы с проектом.

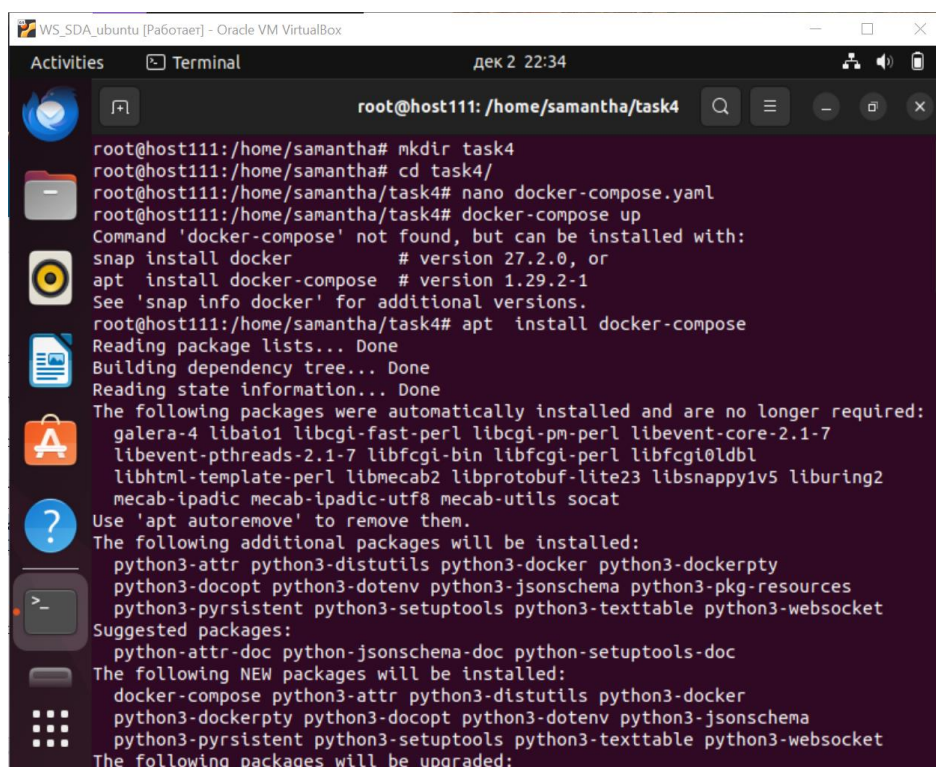
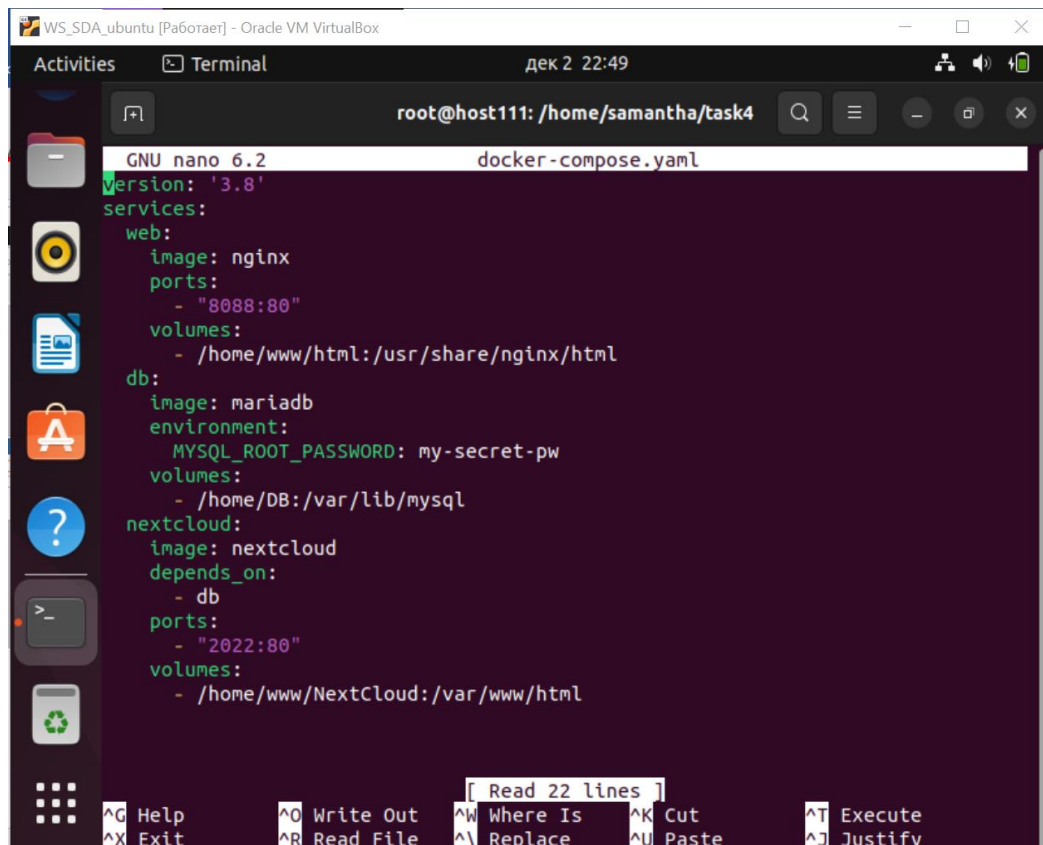
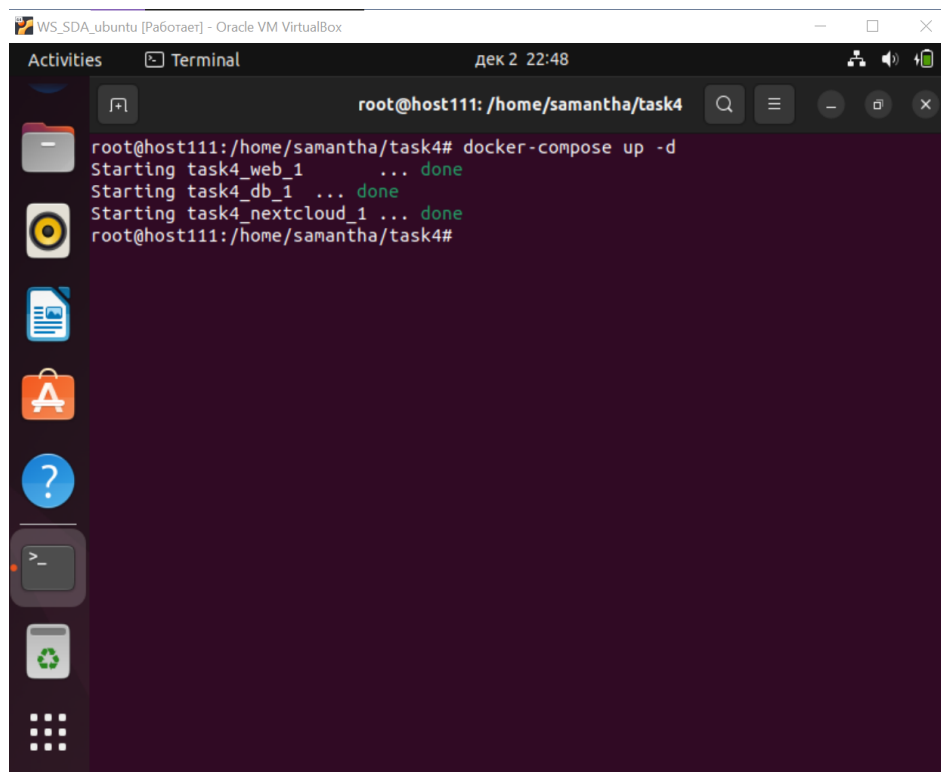


Рисунок 14 – Подготовка к развёртыванию compose-проекта



```
root@host111: /home/samantha/task4
GNU nano 6.2 docker-compose.yaml
version: '3.8'
services:
  web:
    image: nginx
    ports:
      - "8088:80"
    volumes:
      - /home/www/html:/usr/share/nginx/html
  db:
    image: mariadb
    environment:
      MYSQL_ROOT_PASSWORD: my-secret-pw
    volumes:
      - /home/DB:/var/lib/mysql
  nextcloud:
    image: nextcloud
    depends_on:
      - db
    ports:
      - "2022:80"
    volumes:
      - /home/www/NextCloud:/var/www/html
```

Рисунок 15 – Файл docker-compose.yaml



```
root@host111: /home/samantha/task4# docker-compose up -d
Starting task4_web_1 ... done
Starting task4_db_1 ... done
Starting task4_nextcloud_1 ... done
root@host111: /home/samantha/task4#
```

Рисунок 16 – Запуск контейнеров проекта

На рисунке 17 представлен результат запуска docker-compose проекта, а именно окно с NextCloud.

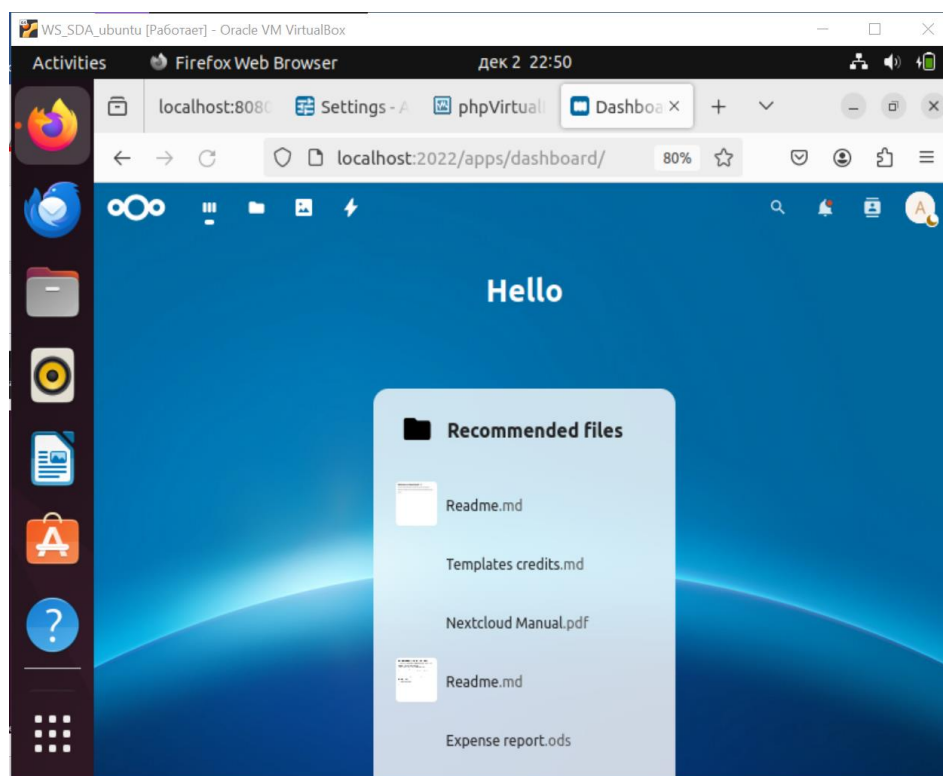


Рисунок 17 – Запущенный контейнер Nextcloud

На рисунке 18 представлена авторизация в DockerHub и начальный этап публикации проекта.

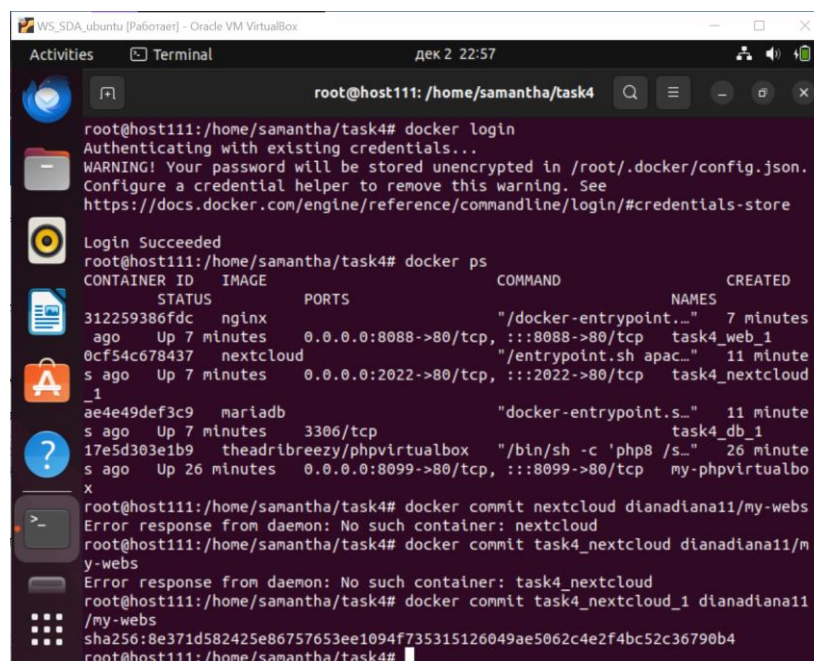
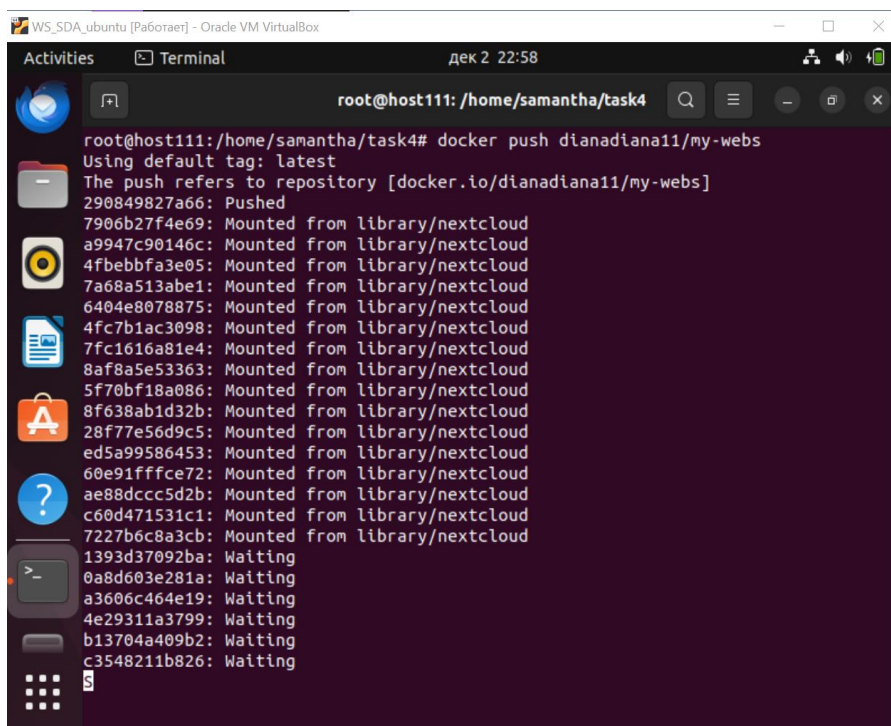


Рисунок 18 – Авторизация на DockerHub и публикация проекта

На рисунках 19 и 20 представлена публикация проекта на DockerHub.



```
WS_SDA_ubuntu [Работаer] - Oracle VM VirtualBox
Activities Terminal дек 2 22:58
root@host111: /home/samantha/task4
root@host111: /home/samantha/task4# docker push dianadiana11/my-webs
Using default tag: latest
The push refers to repository [docker.io/dianadiana11/my-webs]
290849827a66: Pushed
7906b27f4e69: Mounted from library/nextcloud
a9947c90146c: Mounted from library/nextcloud
4fbebffa3e05: Mounted from library/nextcloud
7a68a513abe1: Mounted from library/nextcloud
6404e8078875: Mounted from library/nextcloud
4fc7b1ac3098: Mounted from library/nextcloud
7fc1616a81e4: Mounted from library/nextcloud
8af8a5e53363: Mounted from library/nextcloud
5f70bf18a086: Mounted from library/nextcloud
8f638ab1d32b: Mounted from library/nextcloud
28f77e56d9c5: Mounted from library/nextcloud
ed5a99586453: Mounted from library/nextcloud
60e91ffffce72: Mounted from library/nextcloud
ae88dccc5d2b: Mounted from library/nextcloud
c60d471531c1: Mounted from library/nextcloud
7227b6c8a3cb: Mounted from library/nextcloud
1393d37092ba: Waiting
0a8d603e281a: Waiting
a3606c464e19: Waiting
4e29311a3799: Waiting
b13704a409b2: Waiting
c3548211b826: Waiting
```

Рисунок 19 – Публикация проекта на DockerHub

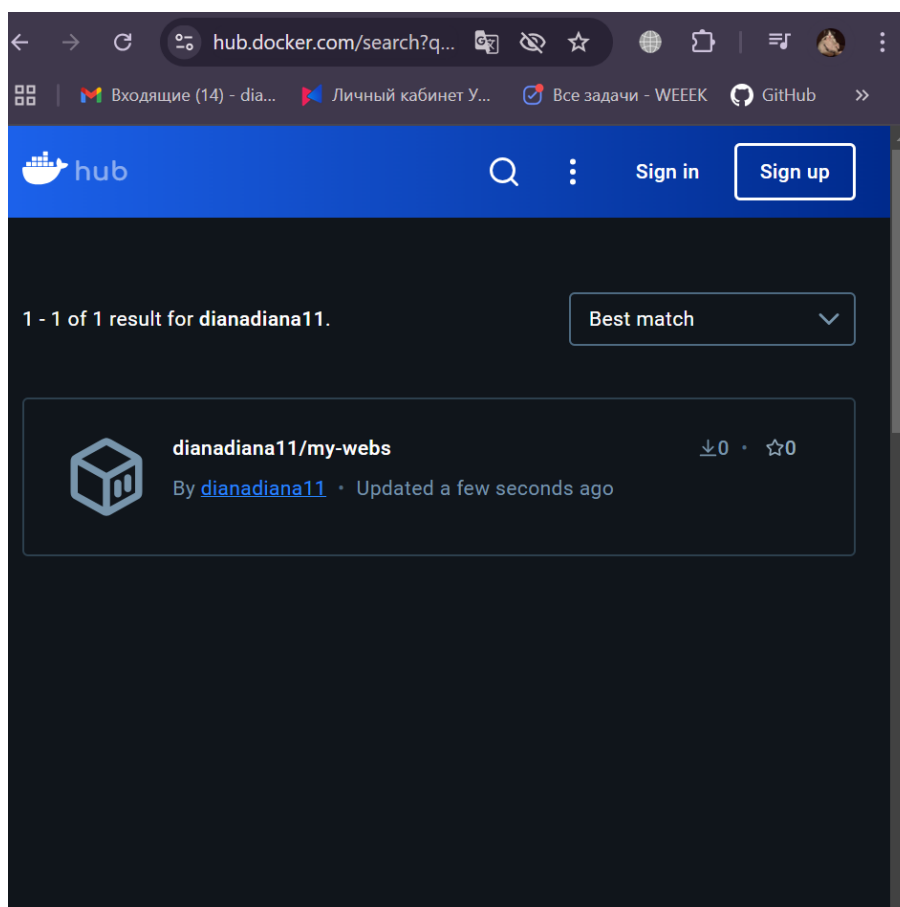
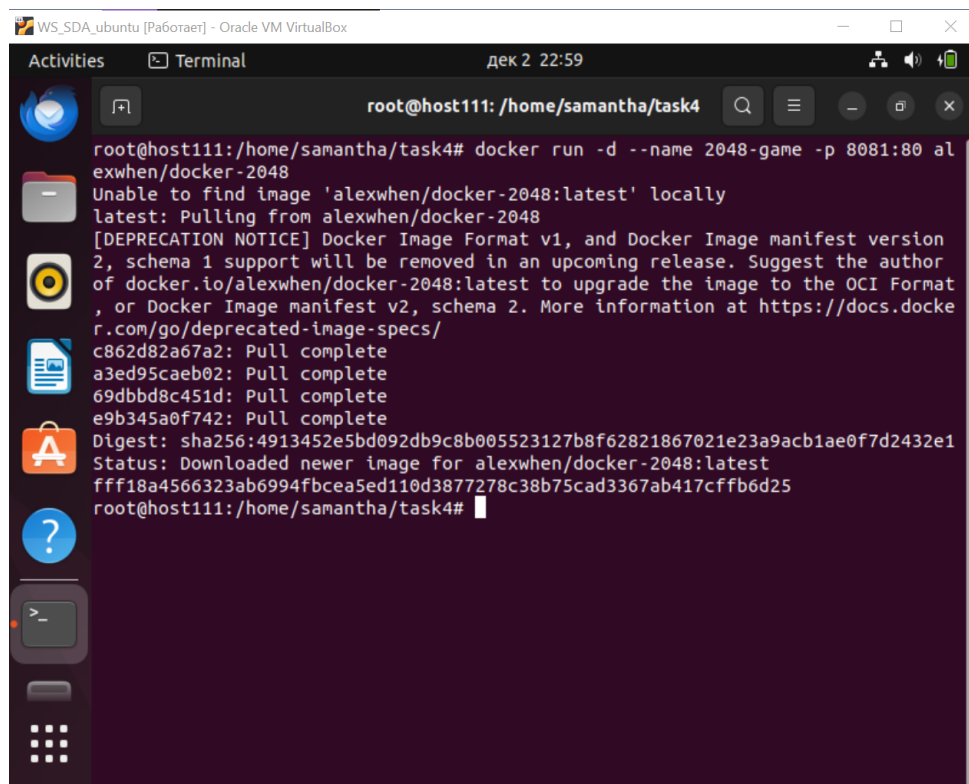


Рисунок 20 – Опубликованный проект на DockerHub

На рисунке 21 представлен запуск контейнера с игрой 2048. Для контейнера установлен порт 8081.



```
root@host111: /home/samantha/task4
root@host111:/home/samantha/task4# docker run -d --name 2048-game -p 8081:80 alexwhen/docker-2048
Unable to find image 'alexwhen/docker-2048:latest' locally
latest: Pulling from alexwhen/docker-2048
[DEPRECATION NOTICE] Docker Image Format v1, and Docker Image manifest version 2, schema 1 support will be removed in an upcoming release. Suggest the author of docker.io/alexwhen/docker-2048:latest to upgrade the image to the OCI Format, or Docker Image manifest v2, schema 2. More information at https://docs.docker.com/go/deprecated-image-specs/
c862d82a67a2: Pull complete
a3ed95cae02: Pull complete
69dbbd8c451d: Pull complete
e9b345a0f742: Pull complete
Digest: sha256:4913452e5bd092db9c8b005523127b8f62821867021e23a9acb1ae0f7d2432e1
Status: Downloaded newer image for alexwhen/docker-2048:latest
ffff18a4566323ab6994fbcea5ed110d3877278c38b75cad3367ab417cffb6d25
root@host111:/home/samantha/task4#
```

Рисунок 21 – Запуск контейнера с игрой 2048

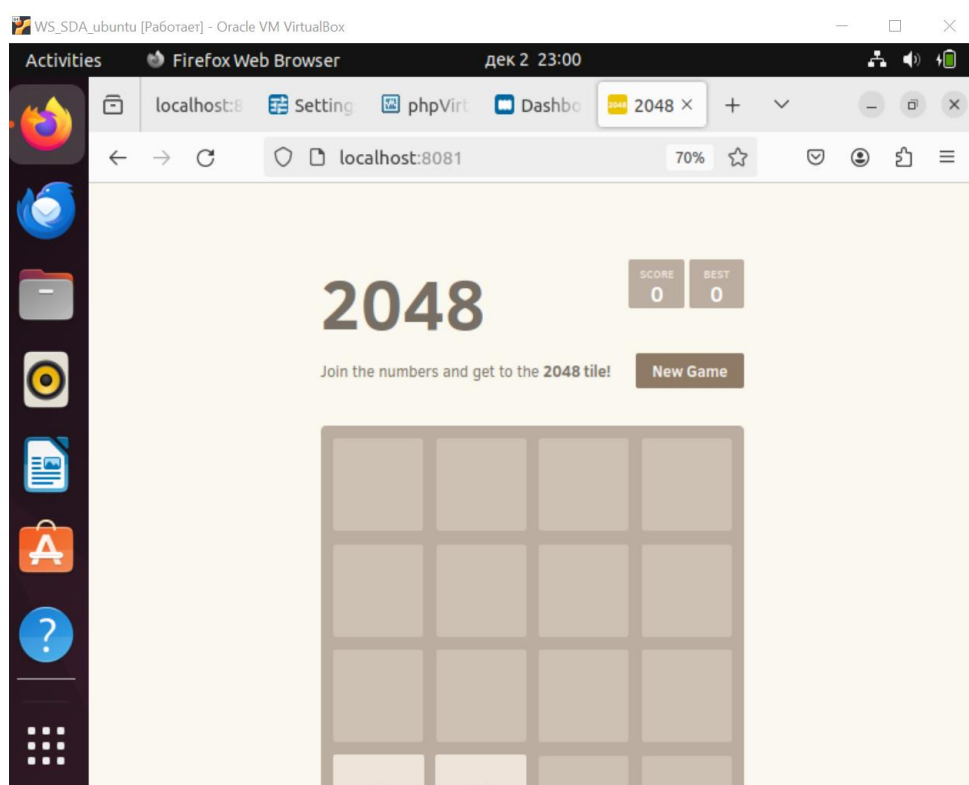


Рисунок 22 – Демонстрация работоспособности 2048

ЗАКЛЮЧЕНИЕ

В ходе выполнения лабораторной работы были изучены основные принципы работы с Docker, его архитектура и возможности для управления контейнерами. Практическая часть работы включала установку Docker и сопутствующих инструментов, запуск различных контейнеров с популярными приложениями (MariaDB, Nginx, NextCloud и phpVirtualBox), настройку взаимодействия между ними, а также автоматизацию запуска приложений с помощью Docker Compose.

Полученные знания позволили понять, как использовать Docker для развертывания изолированных и легко управляемых приложений. Кроме того, был продемонстрирован процесс создания и управления базой данных, работы с веб-приложением, а также размещение собственного проекта на платформе DockerHub для дальнейшего использования.

В результате выполнения данной работы были достигнуты следующие цели:

- Освоены основные команды и принципы работы с Docker.
- Разработан и протестирован проект, включающий несколько связанных приложений.
- Изучены подходы к автоматизации развёртывания и управлению инфраструктурой приложений.

СПИСОК ИСТОЧНИКОВ

1. <https://docs.docker.com/>
2. <https://hub.docker.com/>