

# DevOps ext program

## Winter 2020-2021

**Creating basic DevOps chain connecting  
local machine, GitHub repository,  
Jenkins and GCP instance.**

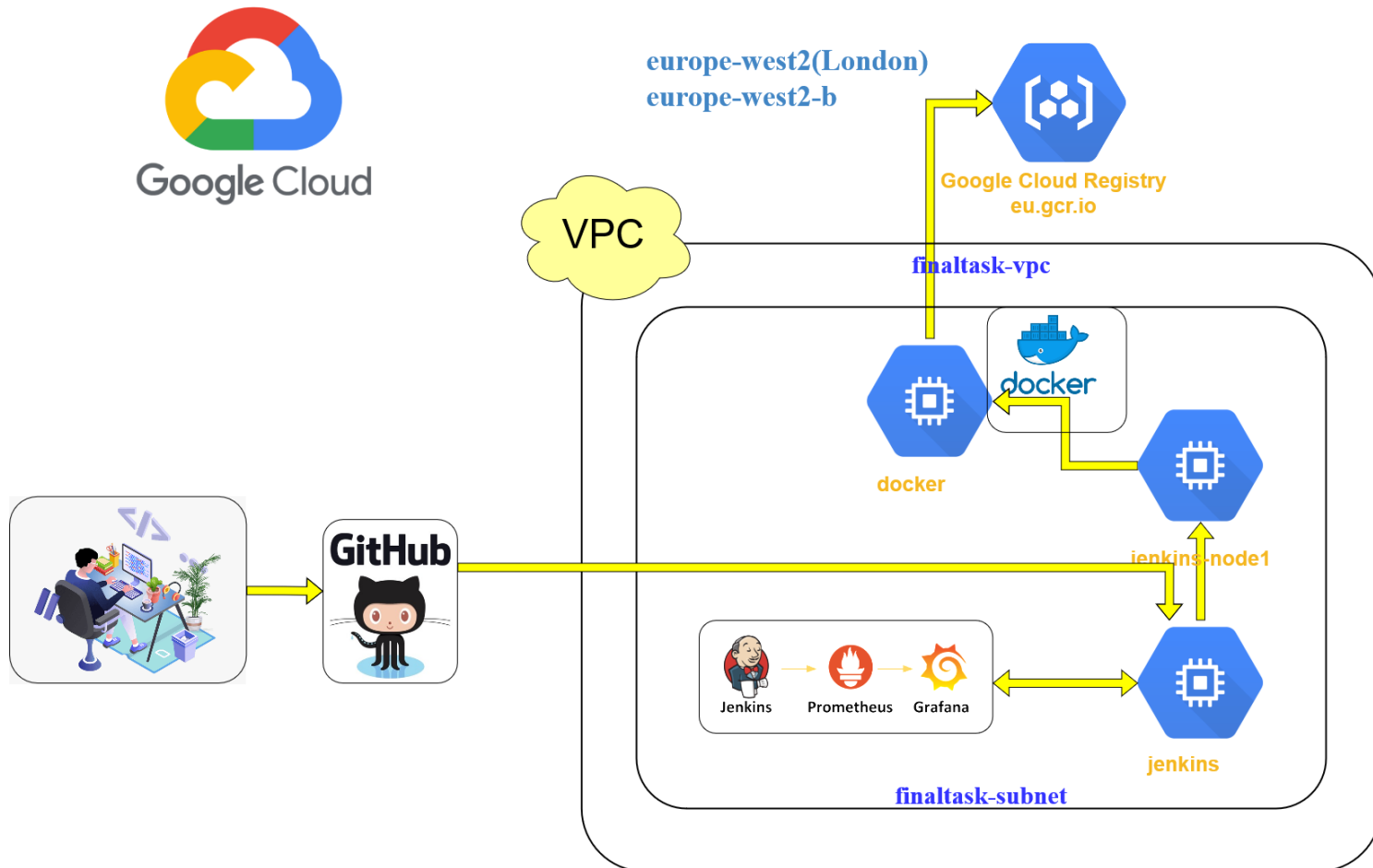
**Andrii Kozak**



## Whoami

- I was born and grew up in the pretty town in the Ukraine. Time stopped in this town.
- I graduated from Kiev University with a degree in applied mathematics.
- The topic of my thesis: solving one dynamic problem by the finite elements method.
- After university I worked in Ivano-Frankivsk at a factory in the ACS department. I was a Software Engineer on PDP-11 computers. I think, PDP-11 computers were the best computers of all time.
- I programmed in various programming languages while was studying in the University and was working at the ACS.
- I worked for the Elcom Company.
- After Elcom, I had worked for the ITC LLC as a lead manager for almost twenty years.
- And finally I worked for the firm Kopiyochka in the IT department.
- Now I'm looking for a job that I like.
- Before I went on the Epam course, I had studied on the SoftServe IT Academy course: Flexible methodologies of developing automated configuration management.
- I am a quick learner and master new technologies.
- I like read books. My favorite writer is Sir Terence David John Pratchett.
- My favorite television series is Lexx.
- My favorite number is 1729, the third Carmichael number.
- I like steamships and hydrofoils.

# Final task diagram



# Task descriptions

- In this project we created automated process for fast deployment site to server after changes.




## **In project we used:**

- Google Cloud Platform;
- GitHub is a provider of Internet hosting for software development and version control using Git;
- Jenkins is a free and open source automation server;
- Terraform is an open-source infrastructure as code software tool created by HashiCorp;
- Bash scripts;
- Docker is a set of platform as a service (PaaS) products that use OS-level virtualization to deliver software in packages called containers.
- docker-compose;
- Django is a Python-based free and open-source web framework that follows the model-template-views (MTV) architectural pattern;
- Prometheus is a free software application used for event monitoring and alerting;
- Grafana is a multi-platform open source analytics and interactive visualization web application;
- GCR.IO - Container Registry is a single place for team to manage Docker images.

# Terraform



- main.tf – create VPC, Subnetwork, VPC firewall configuration.
- docker.tf – create docker instance ( docker.sh – install docker, docker-compose).
- jenkins.tf – create jenkins instance ( jenkins.sh –install Java, jenkins, docker, docker-compose).
- Jenkins-node1.tf – create jenkins-node1 instance ( jenkins-node1.sh – install Java).
- All instances have static IP.

Filter VM instances							Columns
<input type="checkbox"/> Name ^	Zone	Recommendation	In use by	Internal IP	External IP	Connect	
<input type="checkbox"/>  docker	europe-west2-b			10.10.0.4 (nic0)	35.230.158.242 ↗	SSH ▾	⋮
<input type="checkbox"/>  jenkins	europe-west2-b			10.10.0.2 (nic0)	35.242.173.66	SSH ▾	⋮
<input type="checkbox"/>  <u>jenkins-node1</u>	europe-west2-b			10.10.0.3 (nic0)	34.105.139.64	SSH ▾	⋮

# GitHub

# GitHub



- We create the following repositories:

KozakAG / myProject

<> Code ⓘ Issues 🔗 Pull requests ⚙ Actions 📁 Projects 📖 Wiki 🛡 Security 📊 Insights ⚙ Settings

main 1 branch 0 tags

Go to file

Add file

Code



KozakAG modified .pptx

a59709e 2 days ago 10 commits

infrastructure	modified .pptx	2 days ago
logo	delete directory	4 days ago
terraform/final_task/gcp-vpc-terraform	delete directory	4 days ago
Andrii Kozak Final Task.pptx	modified .pptx	2 days ago
CV_DevOps(2)_Lv_Andrii_Kozak.pdf	add file CV & project doc	2 days ago
README.md	first commit	9 days ago

KozakAG / mywebsite

<> Code ⓘ Issues 🔗 Pull requests ⚙ Actions 📁 Projects 📖 Wiki 🛡 Security 📊 Insights ⚙ Settings

main 1 branch 0 tags

Go to file

Add file

Code



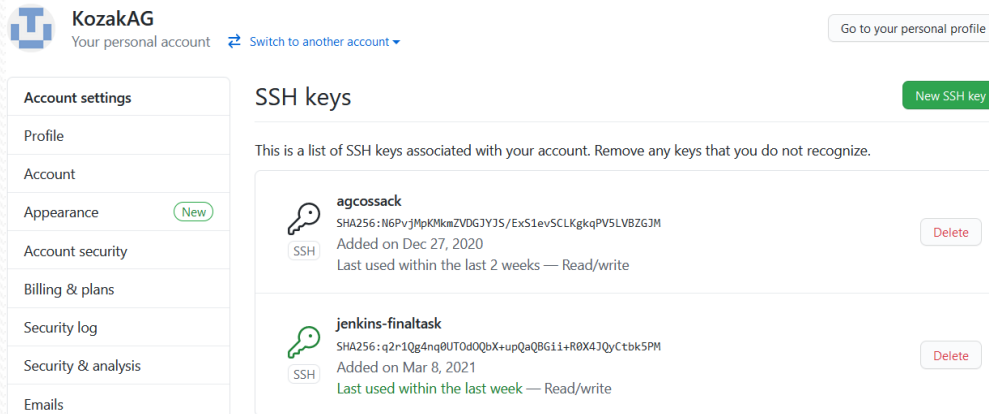
KozakAG test3

30c69fe 4 days ago 10 commits

mywebsite	test3	4 days ago
Dockerfile	rename directory	6 days ago
README.md	first commit	6 days ago
docker-compose.yml	rename directory	6 days ago
requirements.txt	rename directory	6 days ago

# GitHub

We add the ssh keys to the repository:

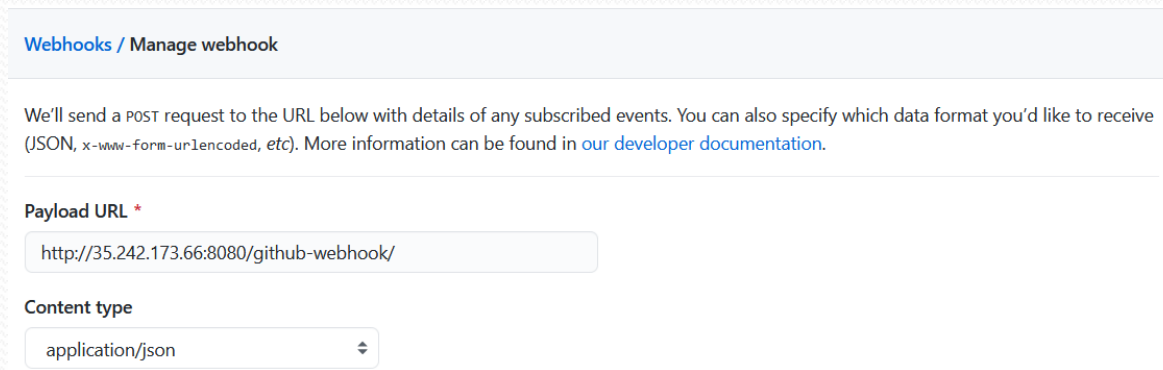


The screenshot shows the GitHub account settings page for user KozakAG. The left sidebar contains a menu with options: Account settings, Profile, Account, Appearance (marked as 'New'), Account security, Billing & plans, Security log, Security & analysis, and Emails. The main content area is titled 'SSH keys' and includes a 'New SSH key' button. Below the title, a message states: 'This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.' There are two SSH keys listed:

Key Name	SHA256 Fingerprint	Added On	Last Used	Permissions	Action
agccossack	SHA256:N6PvjMpl0HkmZVDGJY3S/ExS1evSCLKqkqPV5LV8ZGJM	Added on Dec 27, 2020	Last used within the last 2 weeks	Read/write	Delete
jenkins-finaltask	SHA256:q2r1Qg4nq0UT0d0Q0bX+upQaQBG1i+R8X41QyCtbk5PM	Added on Mar 8, 2021	Last used within the last week	Read/write	Delete



We add webhook to the repository



The screenshot shows the GitHub Webhooks configuration page. The title is 'Webhooks / Manage webhook'. The main text reads: 'We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in our [developer documentation](#).' Below this, there are two input fields:

**Payload URL \***

**Content type**

# Jenkins



## Configure node1 (slave)

Dashboard > Nodes > 34.105.139.64

Back to List

Status

Delete Agent

Configure

Build History

Load Statistics

Script Console

Log

System Information

Disconnect

**Build Executor Status** ^

Idle

Idle

Idle

Name	<input type="text" value="34.105.139.64"/>	
Description	<input type="text"/>	
# of executors	<input type="text" value="4"/>	
Remote root directory	<input type="text" value="/home/agcossack"/>	
Labels	<input type="text" value="34.105.139.64"/>	
Usage	<input type="text" value="Use this node as much as possible"/>	
Launch method	<input type="text" value="Launch agents via SSH"/>	
Host	<input type="text" value="34.105.139.64"/>	
Credentials	<input type="text" value="agcossack"/>	
Host Key Verification Strategy	<input type="text" value="Manually trusted key Verification Strategy"/>	

Availability

Require manual verification

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	34.105.139.64	Linux (amd64)	In sync	17.23 GB	0 B	17.23 GB	37ms
	master	Linux (amd64)	In sync	16.36 GB	0 B	16.36 GB	0ms
Data obtained		41 min	41 min	41 min	41 min	41 min	41 min



# Jenkins



Configure our job: finaltask.

### Add Credentials

Domain: Global credentials (unrestricted) ▼

Kind: SSH Username with private key ▼

Scope: Global (Jenkins, nodes, items, all child items, etc) ▼

ID: github-ssh-key

Description: github-ssh-key

Username: KozakAG

Private Key: ☒ Enter directly

Key: -----BEGIN RSA PRIVATE KEY--  
MIIEpAIBAAKCAQEAvSA41w7zbIng  
o3fM3/ZYRp04V+KhvXtBs36msu1N

Passphrase:

General Source Code Management **Build Triggers** Build Environment Build

- ☐ Trigger builds remotely (e.g., from scripts)
- ☐ Build after other projects are built
- ☐ Build periodically
- ☒ GitHub hook trigger for GITScm polling
- ☐ Poll SCM

# Jenkins



## Configure our job: finaltask.

☒ Restrict where this project can be run

Label Expression

Label **34.105.139.64** matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.

☒ Send files or execute commands over SSH before the build starts

SSH Publishers

SSH Server

Name

Advanced...

Transfers

Transfer Set

Source files

Remove prefix

Remote directory

Exec command

All of the transfer fields (except for Exec timeout) support substitution of [Jenkins environment variables](#)

```
agcossack@docker:/var/www$ cat script.sh
cd /var/www/mywebsite/ && docker-compose stop && cd ..
cd /var/www/ && sudo rm -r mywebsite/ && git clone https://github.com/KozakAG/mywebsite
cd /var/www/mywebsite/ && docker-compose up -d
agcossack@docker:/var/www$
```

# Prometheus



Prometheus

**Prometheus** Alerts Graph Status ▾ Help

☐ Enable query history

Expression (press Shift+Enter for newlines)

Execute - insert metric at cursor - ▾

Graph Console

- 1h +

◀ Until ▶

Res. (s)

☐ stacked

Expression (press Shift+Enter for newlines)

Execute - insert metric at cursor - ▾

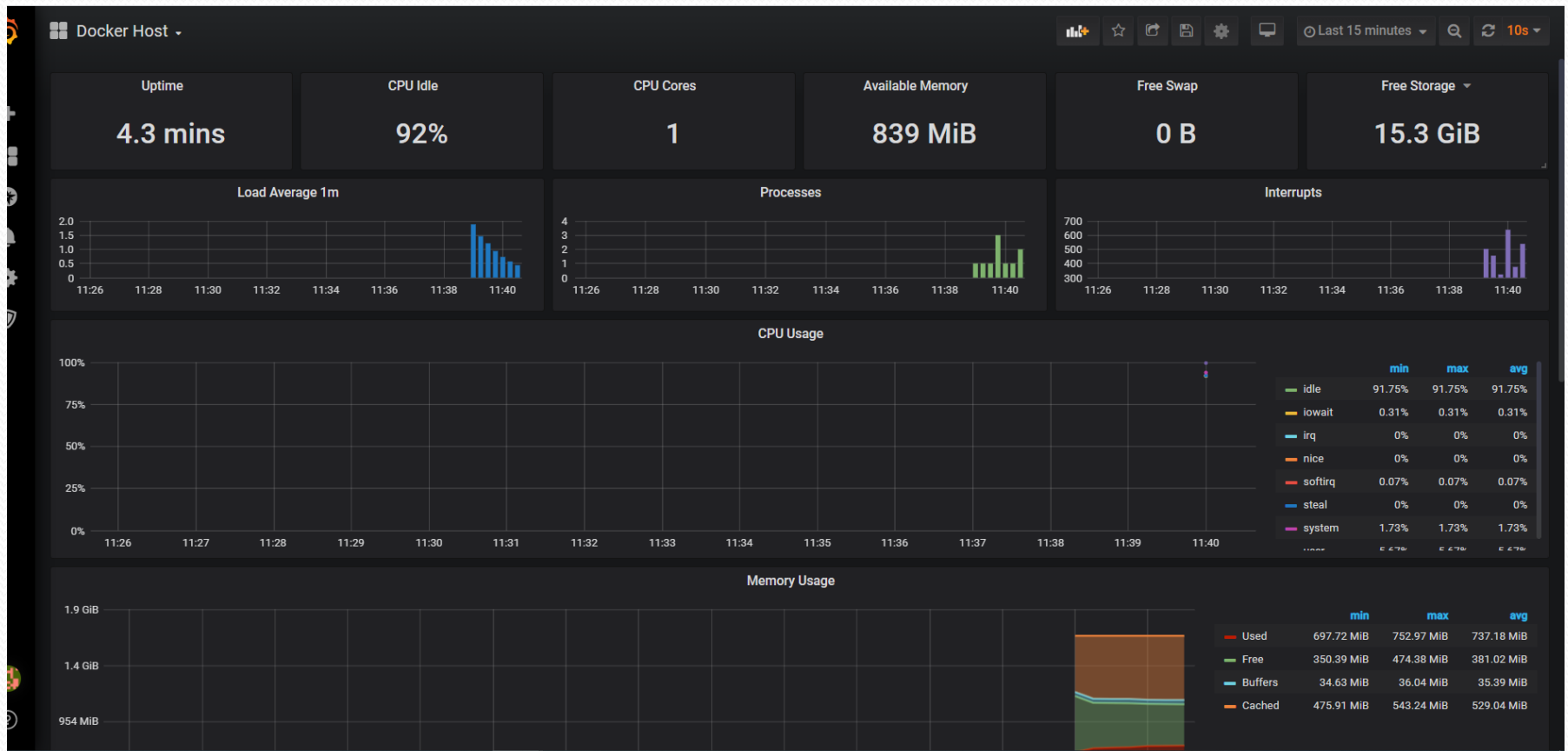
Graph Console

◀ Moment ▶

Element	Value
no data	

Add Graph

# Grafana





Prometheus



Grafana

# Prometheus + Grafana install

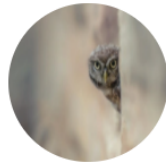
- We will use Einsteinish/Docker-Compose-Prometheus-and-Grafana, which includes Prometheus, Grafana, cAdvisor, NodeExporter and AlertManager.
- Clone the repository and run the installation:

```
git clone https://github.com/Einsteinish/Docker-Compose-Prometheus-and-Grafana.git
cd Docker-Compose-Prometheus-and-Grafana
docker-compose up -d
```

# Django



```
agcossack@docker:/var/www/mywebsite$ tree
.
├── Dockerfile
├── README.md
├── docker-compose.yml
├── mywebsite
│   ├── base
│   │   ├── pycache
│   │   │   ├── __init__.cpython-37.pyc
│   │   │   ├── admin.cpython-37.pyc
│   │   │   ├── admin.cpython-39.pyc
│   │   │   ├── models.cpython-37.pyc
│   │   │   ├── models.cpython-39.pyc
│   │   │   ├── urls.cpython-37.pyc
│   │   │   ├── urls.cpython-39.pyc
│   │   │   ├── views.cpython-37.pyc
│   │   │   └── views.cpython-39.pyc
│   │   ├── admin.py
│   │   ├── apps.py
│   │   ├── migrations
│   │   │   ├── pycache
│   │   │   │   ├── __init__.cpython-37.pyc
│   │   │   └── __init__.cpython-39.pyc
│   │   ├── models.py
│   │   ├── templates
│   │   │   ├── base
│   │   │   │   └── home.html
│   │   ├── tests.py
│   │   ├── urls.py
│   │   └── views.py
│   ├── db.sqlite3
│   ├── manage.py
│   └── mywebsite
│       ├── pycache
│       │   ├── __init__.cpython-37.pyc
│       │   ├── settings.cpython-37.pyc
│       │   ├── settings.cpython-39.pyc
│       │   ├── urls.cpython-37.pyc
│       │   ├── urls.cpython-39.pyc
│       │   ├── wsgi.cpython-37.pyc
│       │   └── wsgi.cpython-39.pyc
│       ├── settings.py
│       ├── urls.py
│       └── wsgi.py
└── staticfiles
    ├── css
    │   └── main.css
    └── images
        ├── andriikozak.jpg
        ├── andriikozak1.jpg
        └── facebook.png
```



**Andrii Kozak**

## Contact information:

Ukraine, Ivano-Frankivsk  
Ready to relocate: Lviv  
**Mobile:** +38 (096) 732-79-94



**Email:** agcossack@gmail.com

**Static website:** <http://andriikozak.website.s3-website-eu-west-1.amazonaws.com>  
**GITHUB:** <https://github.com/KozakAG>  
**GITHUB:** [projecthttps://github.com/KozakAG/KDevOps\\_online\\_Lviv\\_2020Q42021Q1](https://github.com/KozakAG/KDevOps_online_Lviv_2020Q42021Q1)

## My Skills

Warranty service of Dell equipment, repair of laptops and computers, installation and recovery of software, installation and maintenance of Eset software, hardware setup

**Cloud platforms:** GCP, AWS

**Operating systems:** Linux (Ubuntu, Centos, Debian), Windows (7/8/10)

**Virtualization/Containerization:** Docker, VirtualBox

**Deployment and configuration:** Terraform, Vagrant, Ansible

**CI/CD tools:** Git, Jenkins

**Applications/WebServers:** Nginx, Apache

**Databases:** MySQL

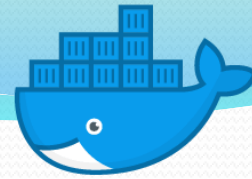
## Experience / IT related educational practice

01.2018 – 11.2018 – NewVitaGroup – IT Department Specialist

11.1998 – 01.2018 – LLC "ITC" – Leading Manager

12.1993 – 11.1998 – "Elcom" – Leading Specialist of Computer Department

08.1985 – 12.1993 – factory "Positron" – ACS Department – Software Engineer



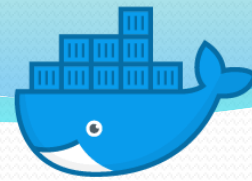
docker

# Docker

- Add the following content to the Dockerfile.

```
docker-compose.yml  Dockerfile X  requirements.txt
C: > MyFolder > work > EPAM > Homework > myweb > Dockerfile > FROM
1  FROM python:3
2
3  ENV PYTHONUNBUFFERED 1
4
5
6  WORKDIR /mywebsite
7
8
9  COPY requirements.txt /mywebsite/
10
11 RUN pip install -r requirements.txt
12 COPY . /mywebsite/
13
```

This Dockerfile starts with a Python 3 parent image. The parent image is modified by adding a new code directory.



# Docker

# docker

- The parent image is further modified by installing the Python requirements defined in the requirements.txt file.

```
docker-compose.yml  Dockerfile  requirements.txt X
C: > MyFolder > work > EPAM > Homework > myweb > requirements.txt
1  Django>=3.0,<4.0
2  psycopg2-binary>=2.8
3
```

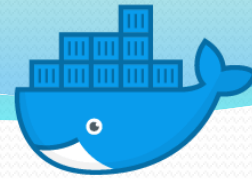
```
docker-compose.yml X  Dockerfile  requirements.txt
C: > MyFolder > work > EPAM > Homework > myweb > docker-compose.yml > version
1  version: '3'
2
3  services:
4    mywebsite:
5      build:
6        context: .
7      ports:
8        - "80:8000"
9      volumes:
10       - ./mywebsite:/mywebsite
11      restart: unless-stopped
12      command: >
13        sh -c "python manage.py runserver 0.0.0.0:8000"
14
```

The docker-compose.yml file describes the services that make our app. In this example those service is a web server.

The compose file also describes which Docker image this service use, which volume need to be mounted inside the container.

Finally, the docker-compose.yml file describes which ports this service expose.





# Docker

# docker

We create the Django project by running the docker-compose run command as follows.

```
agcossack@docker: /var/www - Google Chrome
ssh.cloud.google.com/projects/final-task-306508/zones/europe-west2-b/instances/docker?useAdminProxy=true&authuser=0&hl=en_U...
agcossack@docker:/var/www$ docker-compose run mywebsite sh -c "django-admin.py startproject mywebsite ."
Creating network "www_default" with the default driver
Building mywebsite
Step 1/6 : FROM python:3
--> 254d4a8a8f31
Step 2/6 : ENV PYTHONUNBUFFERED 1
--> Using cache
--> dd07dea27fcb
Step 3/6 : WORKDIR /mywebsite
--> Using cache
--> fe217e92ca6d
Step 4/6 : COPY requirements.txt /mywebsite/
--> Using cache
--> 8afe671bab94
Step 5/6 : RUN pip install -r requirements.txt
--> Using cache
--> 7a0afa343799
Step 6/6 : COPY . /mywebsite/
--> Using cache
--> e49dc1ff50df
Successfully built e49dc1ff50df
Successfully tagged www_mywebsite:latest
WARNING: Image for service mywebsite was built because it did not already exist. To rebuild this image you must use
'docker-compose build' or 'docker-compose up --build'.
Creating www_mywebsite_run ... done
agcossack@docker:/var/www$
```



# Docker

docker

Change the ownership of the new files.

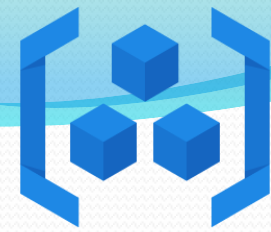
```
sudo chown -R $USER:$USER .
```

Run the docker-compose up command from the top level directory for your project.

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

```
agcossack@docker: /var/www - Google Chrome
ssh.cloud.google.com/projects/final-task-306508/zones/europe-west2-b/instances/docker?useAdminProxy=true&authuser=0&hl=en_U...
agcossack@docker:/var/www$ docker-compose up -d
Starting www_mywebsite_1 ... done
agcossack@docker:/var/www$
```

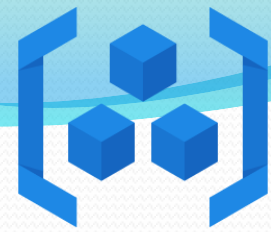
# GCR.IO



Container Registry is a single place for team to manage Docker images, perform vulnerability analysis, and decide who can access what with access control. Existing CI/CD integrations let you set up fully automated Docker pipelines to get fast feedback.

```
agcossack@docker:/var/www/mywebsite$ docker images
REPOSITORY          TAG          IMAGE ID       CREATED        SIZE
www_mywebsite       latest       e49dc1ff50df   6 days ago    937MB
mywebsite_mywebsite latest       77280caba9cc   6 days ago    937MB
hello-world         latest       d1165f221234   9 days ago    13.3kB
python              3           254d4a8a8f31   2 weeks ago    885MB
gcr.io/final-task-306508/quickstart-image tag1         bc5c421ecd6c   3 years ago    9.86MB
gcr.io/google-samples/hello-app 1.0         bc5c421ecd6c   3 years ago    9.86MB
agcossack@docker:/var/www/mywebsite$ docker tag www_mywebsite eu.gcr.io/final-task-306508/mywebsite
agcossack@docker:/var/www/mywebsite$ docker images
REPOSITORY          TAG          IMAGE ID       CREATED        SIZE
www_mywebsite       latest       e49dc1ff50df   6 days ago    937MB
eu.gcr.io/final-task-306508/mywebsite latest       e49dc1ff50df   6 days ago    937MB
mywebsite_mywebsite latest       77280caba9cc   6 days ago    937MB
hello-world         latest       d1165f221234   9 days ago    13.3kB
python              3           254d4a8a8f31   2 weeks ago    885MB
gcr.io/google-samples/hello-app 1.0         bc5c421ecd6c   3 years ago    9.86MB
gcr.io/final-task-306508/quickstart-image tag1         bc5c421ecd6c   3 years ago    9.86MB
agcossack@docker:/var/www/mywebsite$ docker push eu.gcr.io/final-task-306508/mywebsite
Using default tag: latest
The push refers to repository [eu.gcr.io/final-task-306508/mywebsite]
84b8937f2a37: Pushed
72bfc2dabe93: Pushed
2c2261dcaa51: Pushed
6e8f1275bd9b: Pushed
7d999a918ae9: Layer already exists
5b164865b353: Layer already exists
302cf02dcc7c: Layer already exists
e3d73f29c674: Layer already exists
10bf86ff9f6a: Layer already exists
da654bc8bc80: Layer already exists
4ef81dc52d99: Layer already exists
909e93c71745: Layer already exists
7f03bfe4d6dc: Layer already exists
latest: digest: sha256:d259e9d7d2be7248c7874bc2b6e1561dcd41cc7db6c911e9928062a59046c383 size: 3050
```

# GCR.IO



☰

Google Cloud Platform

finaltask ▼

🔍 Search products and resources

▼

Container Registry

Repositories

REFRESH

finaltask

Filter

All hostnames ▼ ⓘ

Name ^	Hostname	Visibility ⓘ
mywebsite	eu.gcr.io	Public



# Thanks for attention!

Student: Andrii Kozak

Email: [agcossack@gmail.com](mailto:agcossack@gmail.com)