

КИИИ Лаб 4

Class Assignment / Homework

- Part 1: Setup Jenkins on your machine
 - Use the `Dockerfile` and the `docker-compose.yml` files;
 - Start the orchestration, unlock Jenkins, select plugins to install, create your admin profile, create an ngrok temporary public URL, setup your secret credentials (GitHub and Docker Hub);
-

PART 1

We create Dockerfile which creates our images when run

```
FROM jenkins/jenkins:lts

USER root

# Install necessary tools and dependencies
RUN apt-get update && \
    apt-get install -y sudo && \
    apt-get clean

# Add a user to the 'sudo' group and allow passwordless sudo
RUN usermod -aG sudo jenkins && \
    echo "jenkins ALL=(ALL) NOPASSWD:ALL" >> /etc/sudoers

USER jenkins
```

We also create a `docker-compose.yml` which defines our containers

```

1  |version: '3'
2
3  ∨ services:
4  ∨   jenkins:
5  ∨     build:
6  ∨       context: .
7  ∨       dockerfile: Dockerfile
8  ∨       container_name: jenkins
9  ∨     ports:
10 ∨       - "8080:8080"
11 ∨     volumes:
12 ∨       - jenkins_home:/var/jenkins_home
13 ∨     environment:
14 ∨       - TZ=UTC
15 ∨     networks:
16 ∨       - jenkins_network
17
18 ∨ networks:
19 ∨   jenkins_network:
20
21 ∨ volumes:
22 ∨   jenkins_home:

```

We run command
Docker-compose up -d

```

=> => extracting sha256:83597e279c9f1542d69f412029b3582c72c298b34928fa27bd8f3ba92e2a8fc5 0.1s
=> => extracting sha256:6c39e3e60898a5bef94bdecca14f7e1e2035e437cb977b973a87abf480a2cc20 0.6s
=> => extracting sha256:29150353c9ba70a3419db154fe2443157c81c3f7b75f5aa51b4cc63c3c64ba15 0.0s
=> => extracting sha256:33fe415a61395401e8e4ce998364bf3c01d2e8bb5831883d65050ed64d37c868 0.0s
=> => extracting sha256:c5d8e31128fd0589f1fd85329c5ad959fb5e162d8aa2f1bf9946906e49a69414 0.0s
=> [jenkins 2/3] RUN apt-get update && apt-get install -y sudo && apt-get clean 6.6s
=> [jenkins 3/3] RUN usermod -s /bin/bash jenkins && echo "jenkins ALL=(ALL) NOPASSWD:ALL" >> /etc/sudoers 0.4s
=> [jenkins] exporting to image 0.2s
=> => exporting layers 0.2s
=> => writing image sha256:24b17d4229198479f1b94846e5948d7c407fcfb24b25f023a8b71ac0afa866e5 0.0s
=> => naming to docker.io/library/jenkinscid-jenkins 0.0s
[+] Running 1/3
 - Network jenkinscid-jenkins_network Created 0.7s
 - Volume "jenkinscid-jenkins_home" Created 0.6s
  ✓ Container jenkins Started 0.6s

```

PS C:\Users\berat\OneDrive\Documents\Github\FINKI\CICD - Continuous Integration & Delivery\Playground\JenkinsCICD>

Unlock Jenkins:

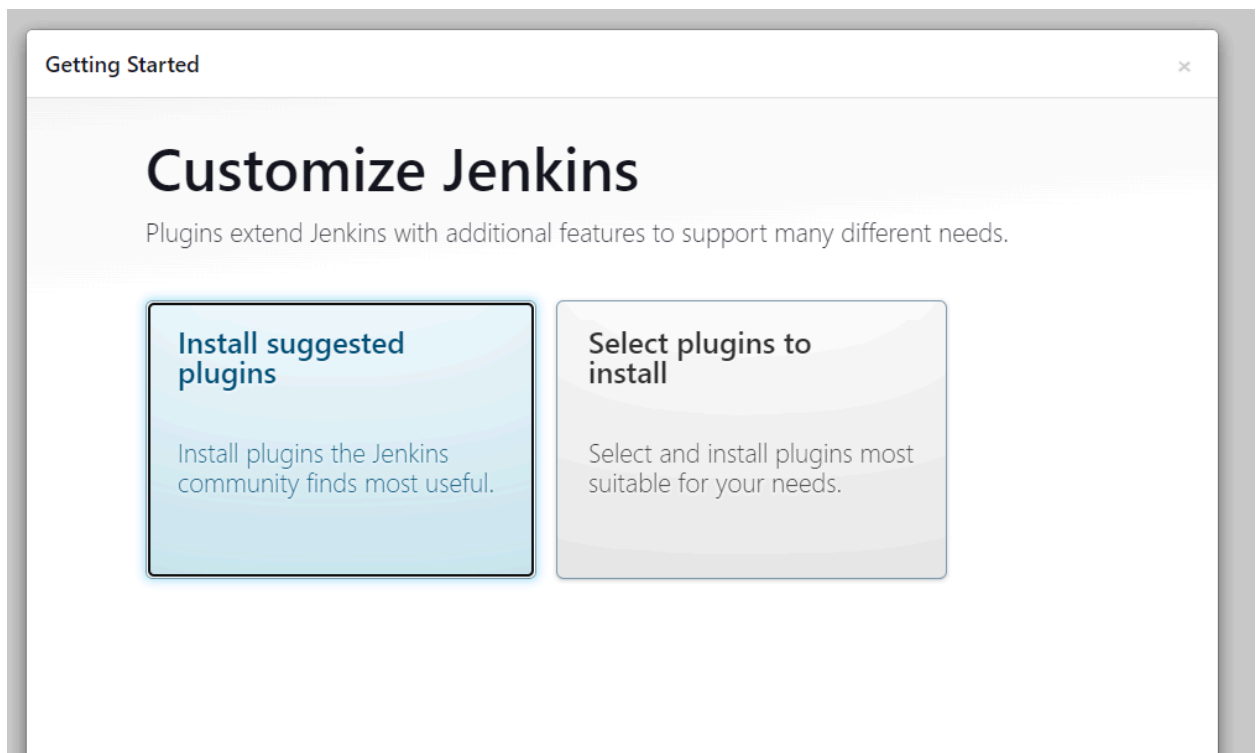
After the containers are up and running, we access Jenkins on browser by visiting `http://localhost:8080`.

We need admin password, we run this command

```
docker exec -it jenkins cat /var/jenkins_home/secrets/initialAdminPassword
```

We get this

```
a15060a1b55d4cf19f5821ac618b7ee0
```



Getting Started

Getting Started

<input checked="" type="checkbox"/> Folders	<input checked="" type="checkbox"/> OWASP Markup Formatter	<input type="checkbox"/> Build Timeout	<input type="checkbox"/> Credentials Binding	** Icons API Folders OWASP Markup Formatter
<input type="checkbox"/> Timestamper	<input type="checkbox"/> Workspace Cleanup	<input type="checkbox"/> Ant	<input type="checkbox"/> Gradle	
<input type="checkbox"/> Pipeline	<input type="checkbox"/> GitHub Branch Source	<input type="checkbox"/> Pipeline: GitHub Groovy Libraries	<input type="checkbox"/> Pipeline: Stage View	
<input type="checkbox"/> Git	<input type="checkbox"/> SSH Build Agents	<input type="checkbox"/> Matrix Authorization Strategy	<input type="checkbox"/> PAM Authentication	
<input type="checkbox"/> LDAP	<input type="checkbox"/> Email Extension	<input type="checkbox"/> Mailer	<input type="checkbox"/> Dark Theme	



Jenkins

Search (CTRL+K)



Berat Ahmetaj

log out

Dashboard

+ New Item

People

Build History

Manage Jenkins

My Views

Build Queue



No builds in the queue.

Build Executor Status



1 Idle

2 Idle

Add description

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job



Set up a distributed build

Set up an agent



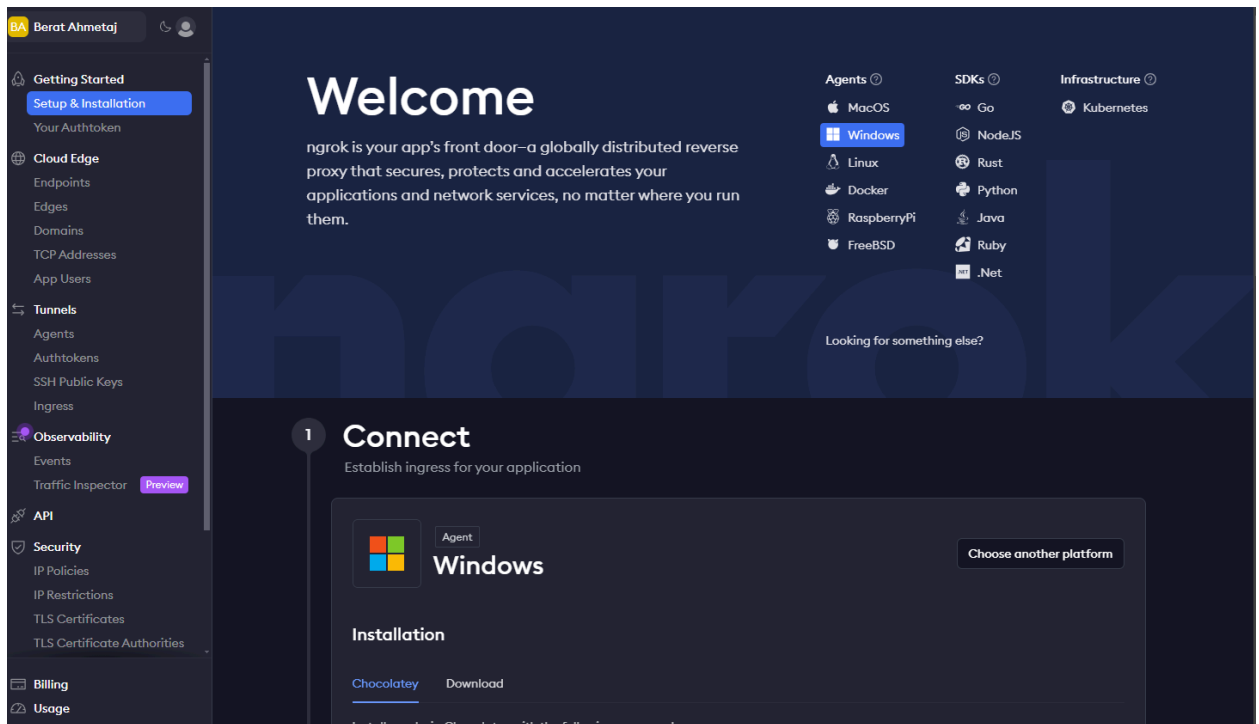
Configure a cloud



Learn more about distributed builds



Install Ngrok (Multiplatform Tunneling Online)



We Get Authkey and install Ngrok through docker

```
docker run -it -e NGROK_AUTHTOKEN=<token> ngrok/ngrok http 80
```

We Install

```
ngrok

Try our new Traffic Inspector Dev Preview: https://ngrok.com/r/ti

Session Status      online
Account             Berat Ahmetaj (Plan: Free)
Update              update available (version 3.8.0, Ctrl-U to update)
Version              3.8.0
Region              Europe (eu)
Latency              50ms
Web Interface        http://0.0.0.0:4040
Forwarding           https://b8db-92-53-30-216.ngrok-free.app -> http://localhost:80

Connections          ttl      opn      rt1      rt5      p50      p90
                     0        0        0.00     0.00     0.00     0.00
```

We Use The Given Tunneling Link to insert it into our Jenkins

Jenkins Location

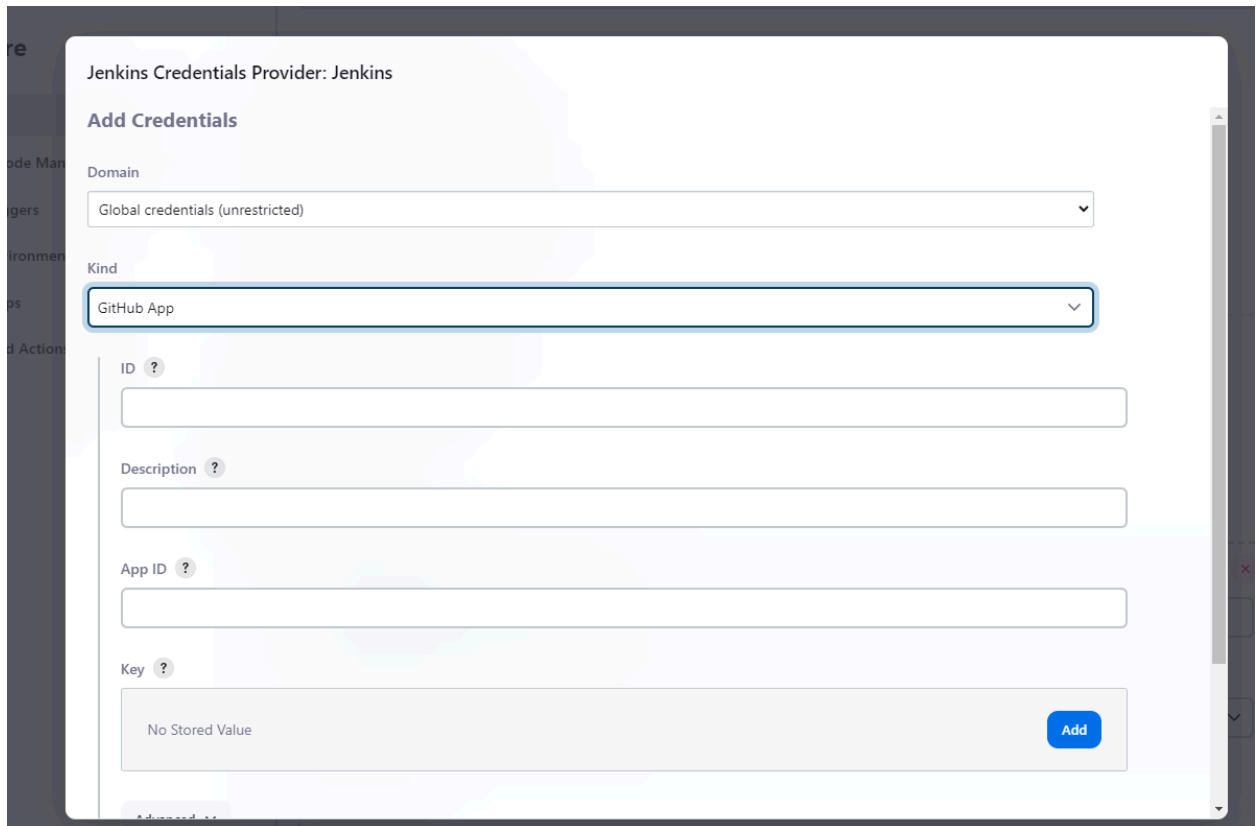
Jenkins URL ?

`https://b8db-92-53-30-216.ngrok-free.app/`

- **Part 2: Create a Freestyle project**
 - Connect it to a simple GitHub repository from your account;
 - Add a Build Step with a simple `ls -l` command;
 - Set up an automatic build, on each minute;

PART 2

We now create Freestyle Project connect it with github



The screenshot shows the 'Add Credentials' dialog in Jenkins. The 'Domain' dropdown is set to 'Global credentials (unrestricted)'. The 'Kind' dropdown is set to 'GitHub App'. Below these, there are input fields for 'ID', 'Description', and 'App ID', all of which are currently empty. At the bottom, there is a 'Key' field with a placeholder text 'No Stored Value' and a blue 'Add' button.

Jenkins Credentials Provider: Jenkins

Add Credentials

Domain
Global credentials (unrestricted)

Kind
GitHub App

ID ?

Description ?

App ID ?

Key ?
No Stored Value Add

Give it secret parameters (which arent secret)


FreestyleTest

Is - I README GITHUB


 Edit description

Disable Project

Permalinks

 #1 (28 Mar 2024, 23:53:24)

Keep this build forever

 Add description

Started 11 sec ago
Took **3.9 sec**



No changes.



Started by user [Berat Ahmetaj](#)



Revision: e90a7f9c06d028b34f2c2d295284be5d514b994b

Repository: <https://github.com/BeratAhmetaj/BeratAhmetaj>




- refs/remotes/origin/master

✓ Console Output

```
Started by user Berat Ahmetaj
Running as SYSTEM
Building in workspace /var/jenkins_home/workspace/FreestyleTest
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/BeratAhmetaj/BeratAhmetaj
> git init /var/jenkins_home/workspace/FreestyleTest # timeout=10
Fetching upstream changes from https://github.com/BeratAhmetaj/BeratAhmetaj
> git --version # timeout=10
> git --version # 'git version 2.39.2'
> git fetch --tags --force --progress -- https://github.com/BeratAhmetaj/BeratAhmetaj +refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/BeratAhmetaj/BeratAhmetaj # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision e90a7f9c06d028b34f2c2d295284be5d514b994b (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f e90a7f9c06d028b34f2c2d295284be5d514b994b # timeout=10
Commit message: "Update README.md"
First time build. Skipping changelog.
[FreestyleTest] $ /bin/sh -xe /tmp/jenkins8559775056839798688.sh
+ ls -l
total 4
-rw-r--r-- 1 jenkins jenkins 558 Mar 20 23:53 README.md
Finished: SUCCESS
```

Permalinks

- [Last build \(#1\), 7 min 41 sec ago](#)
- [Last stable build \(#1\), 7 min 41 sec ago](#)
- [Last successful build \(#1\), 7 min 41 sec ago](#)
- [Last completed build \(#1\), 7 min 41 sec ago](#)

<input type="checkbox"/>	Name	Image	Status
<input type="checkbox"/>	 admiring_kowalevski b0b34f9b9420 	ngrok/ngrok	Running
<input type="checkbox"/>	>  jenkinscid		Running (

<input type="checkbox"/>	Name	Tag	Status	Created	Size	Actions
<input type="checkbox"/>	jenkinscid-jenkins 24b17d422919	latest	In use	21 minutes ag	489.01 MB	▶ ⋮
<input type="checkbox"/>	ngrok/ngrok 9242eef19955	latest	In use	N/A	139.44 MB	▶ ⋮

<input type="checkbox"/>	ID	Name	Builder	Status	Duration	Created	Author	
<input type="checkbox"/>	KGFTQ6	JenkinsCICD	desktop-linux	✓ Completed	48.3s	47 minutes ago	N/A	🗑
<input type="checkbox"/>	NIR7XX	composeapp	desktop-linux	✓ Completed	1m 22s	13 days ago	N/A	🗑

- Part 3: Create a Pipeline in Blue Ocean
 - Create a new GitHub repository with the supplied **Dockerfile** and **Jenkinsfile** files;
 - Modify the **Jenkinsfile** to match your repo and credentials;
 - Create a new Pipeline with Blue Ocean and connect it to the GitHub repo;
 - Create a Webhook in the GitHub repository;
 - Make changes in the GitHub repo, and see the automatic Pipeline run happening in Jenkins;
 - Create a new 'dev' branch in the repo. Make a change in it. Create a pull request from 'dev' to 'main'. Accept the PR and merge the changes. Watch what happens in Jenkins.



PART 3

Kind

Username with password

Scope ?


Global (Jenkins, nodes, items, all child items, etc)


Jenkins				
Pipelines		Administration	Logout	
Pipelines <input type="text" value="search pipelines..."/> <input type="button" value="New Pipeline"/>				
NAME	HEALTH	BRANCHES	PR	
KIII-Exercise-01		-	-	


Pipeline in Blue Ocean


✓


Where do you store your code?

 Bitbucket Cloud

 Bitbucket Server

 GitHub

 GitHub Enterprise

 Git

○

Connect to GitHub

Jenkins needs an access token to authorize itself with GitHub.
[Create an access token here.](#)

○

Complete

We connect Github

✓

Connect to GitHub

Jenkins needs an access token to authorize itself with GitHub.
[Create an access token here.](#)

STATUS	RUN	COMMIT	BRANCH	MESSAGE	DURATION	COMPLETED	
✓	1	51736a1	dev	Branch indexing	44s	a few seconds a	↺
✓	1	be9c75d	main	Branch indexing	44s	a few seconds a	↺

Configuration

- General
- Branch Sources
- Build Configuration
- Scan Repository Triggers
- Orphaned Item Strategy
- Appearance
- Health metrics
- Properties

Add property

Add source

Build Configuration

Mode

by Jenkinsfile

Script Path ?

Jenkinsfile

Scan Repository Triggers

☐ Periodically if not otherwise run ?

Orphaned Item Strategy

GITHUB REPO

16 lines (16 sloc) | 469 Bytes

Raw

Blame

✎

⌵

📄

🗑

```

1  node {
2      def app
3      stage('Clone repository') {
4          checkout scm
5      }
6      stage('Build image') {
7          app = docker.build("mjovanovik/kiii-jenkins")
8      }
9      stage('Push image') {
10         docker.withRegistry('https://registry.hub.docker.com', 'dockerhub') {
11             app.push("${env.BRANCH_NAME}-${env.BUILD_NUMBER}")
12             app.push("${env.BRANCH_NAME}-latest")
13             // signal the orchestrator that there is a new version
14         }
15     }

```

✓

KILL-Exercise-02 < 2

Pipeline

Changes

Tests

Artifacts

↺

✎

⚙️

🔗

Logout

✕

Branch: main

🕒 34s

Changes by noreply

Commit: f05d2c5

🕒 a minute ago

Push event to branch main



Push image - 19s



✓	> Checks if running on a Unix-like node	<1s
✓	> docker tag "\$JD_ID" "\$JD_TAGGED_IMAGE_NAME" — Shell Script	<1s
✓	> Checks if running on a Unix-like node	<1s
✓	> docker push "\$JD_TAGGED_IMAGE_NAME" — Shell Script	10s
✓	> Checks if running on a Unix-like node	<1s
✓	> docker tag "\$JD_ID" "\$JD_TAGGED_IMAGE_NAME" — Shell Script	<1s
✓	> Checks if running on a Unix-like node	<1s
✓	> docker push "\$JD_TAGGED_IMAGE_NAME" — Shell Script	4s

```

3 556c7114696f: Preparing
4 a1bd4a5c5a79: Preparing
5 597a12cbab02: Preparing
6 8820623d95b7: Preparing
7 338a545766ba: Preparing
8 e65242c66bbe: Preparing
9 3af14c9a24c9: Preparing
10 3af14c9a24c9: Waiting
11 e65242c66bbe: Waiting
12 556c7114696f: Layer already exists
13 a1bd4a5c5a79: Layer already exists
14 338a545766ba: Layer already exists
15 8820623d95b7: Layer already exists
16 597a12cbab02: Layer already exists
17 3af14c9a24c9: Layer already exists
18 e65242c66bbe: Layer already exists
19 main-latest: digest: sha256:917c1115ba445697525ab70fa782a399b43c6b2b5a2e81e051b0b536c909c6ab
size: 1777
  
```