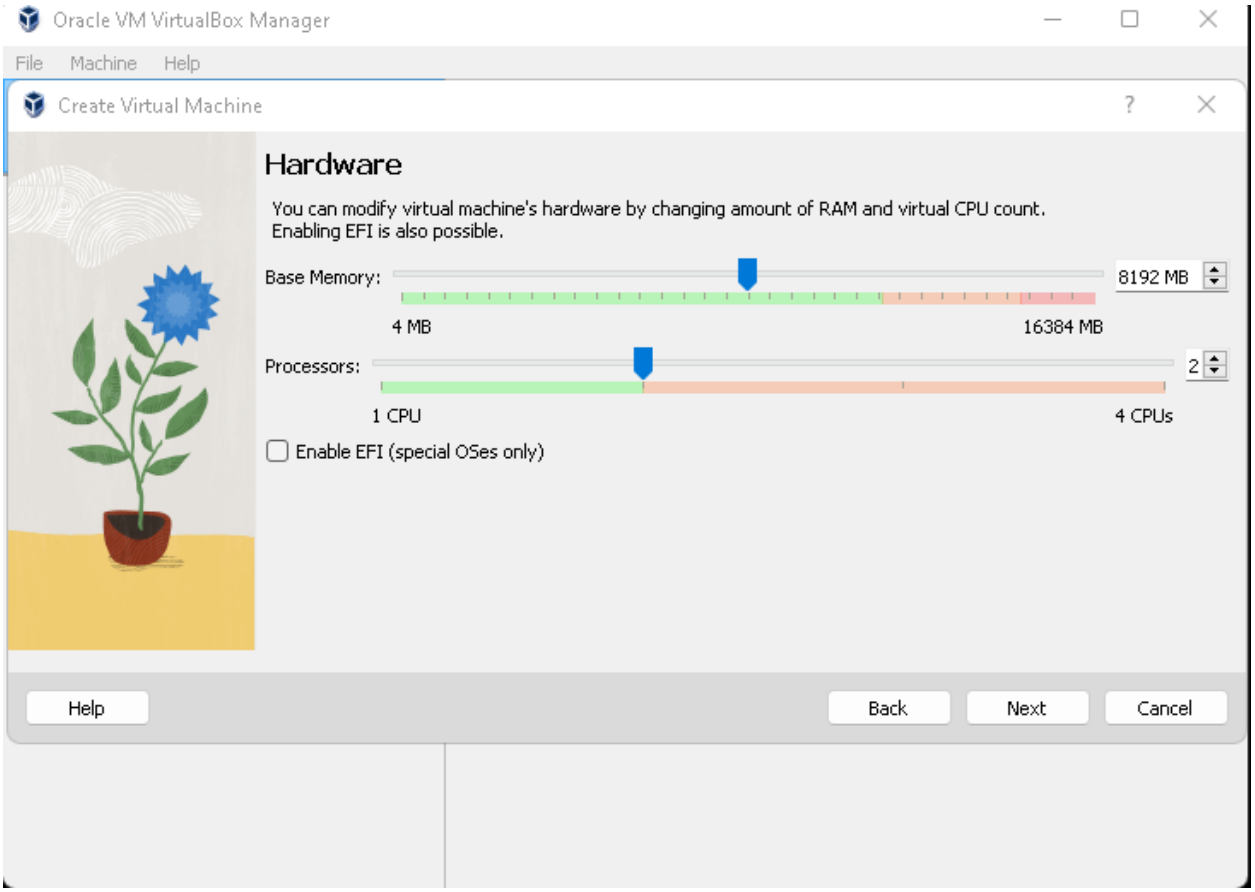
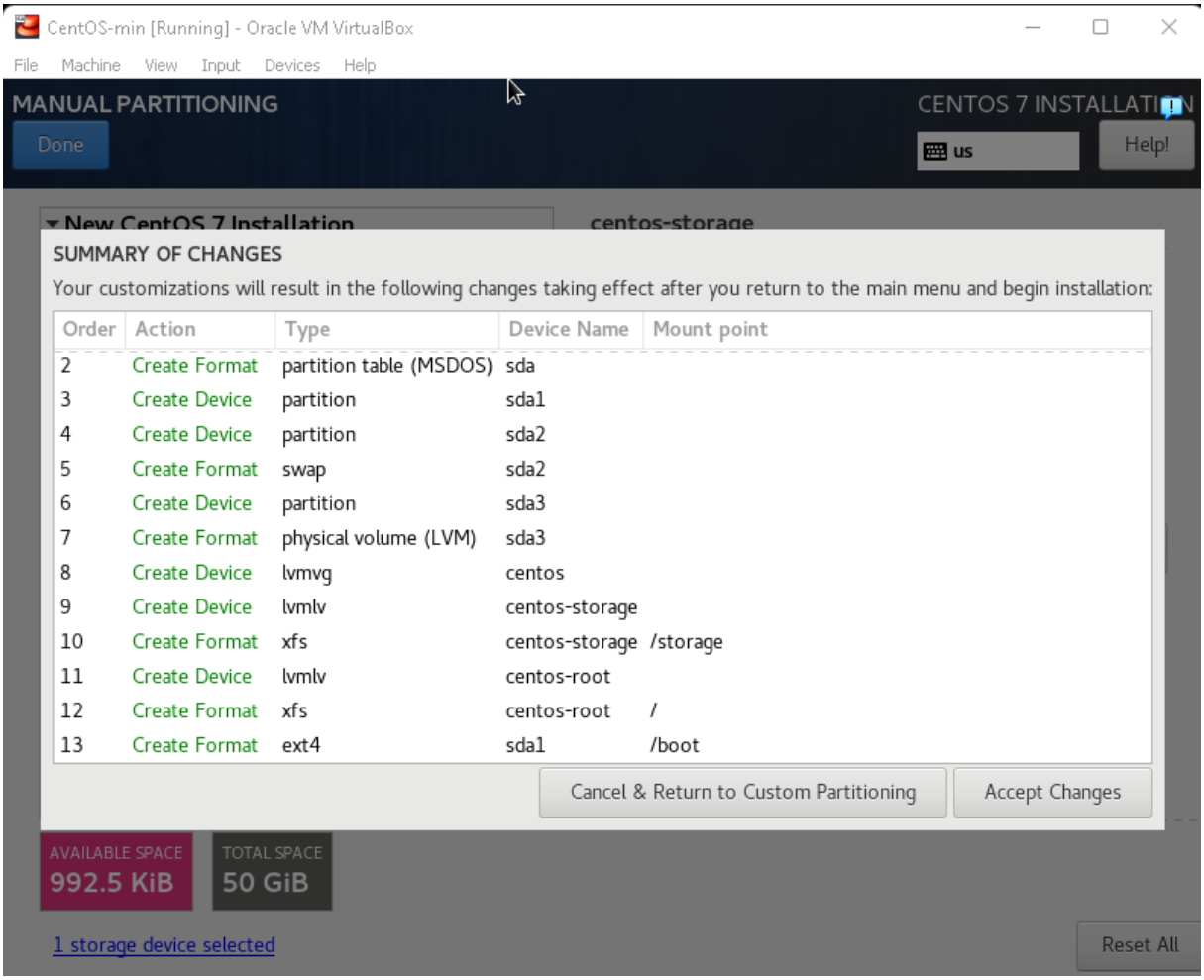
1. Kreiranje virtuelne mašine

Odabir RAM memorije i CPU

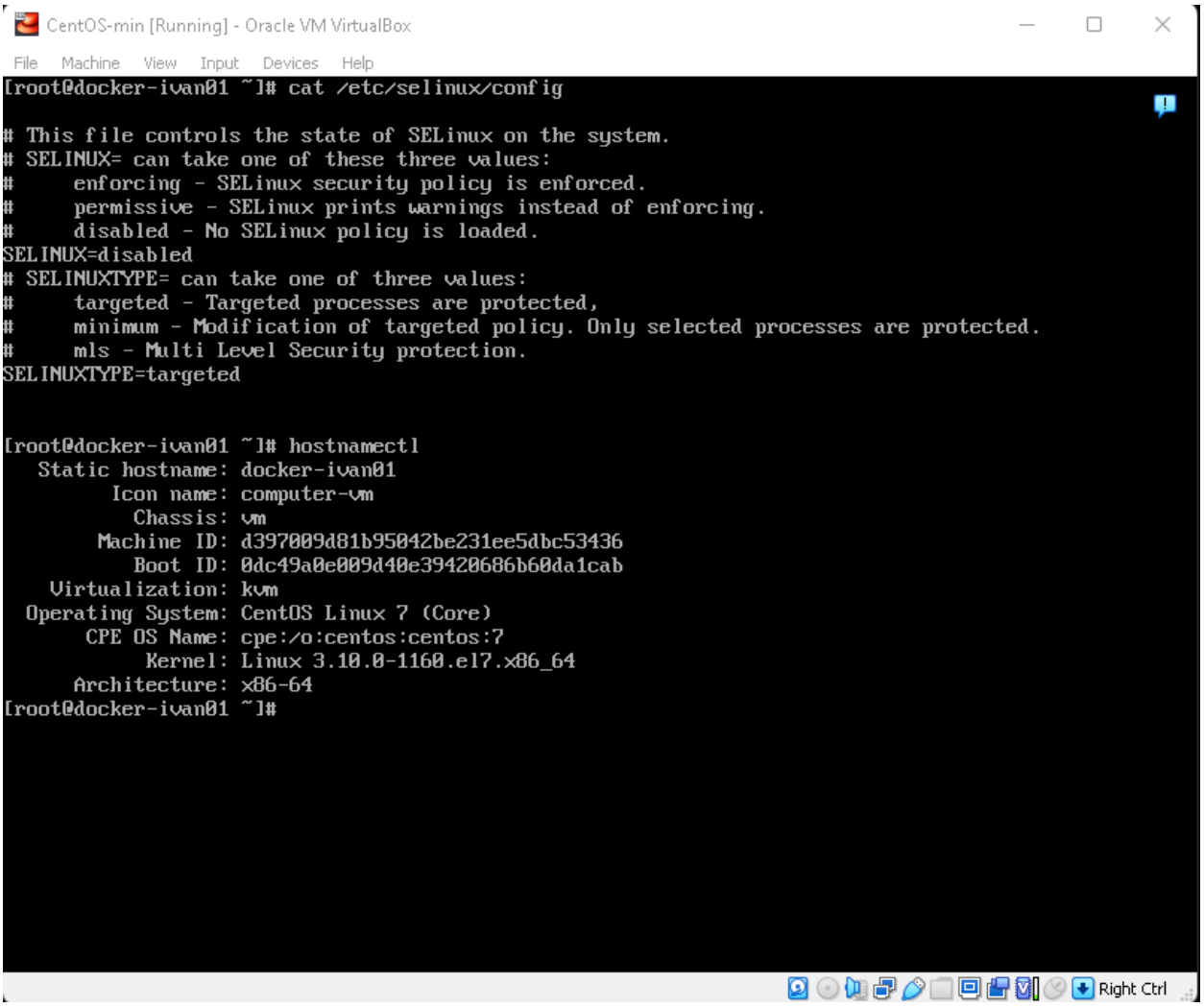


Kreiranje particija na slici ispod



1.1. Podesavanja Linuxa i instalacija docker-a

Disable selinux i prikaz hostname-a



Prikaz konfiguracije mreze (DHCP)

Nakon toga koriscena je docker dokumentacija za instalaciju Docker-a uputstvo se moze naci na sledecem linku

https://docs.docker.com/engine/install/centos/

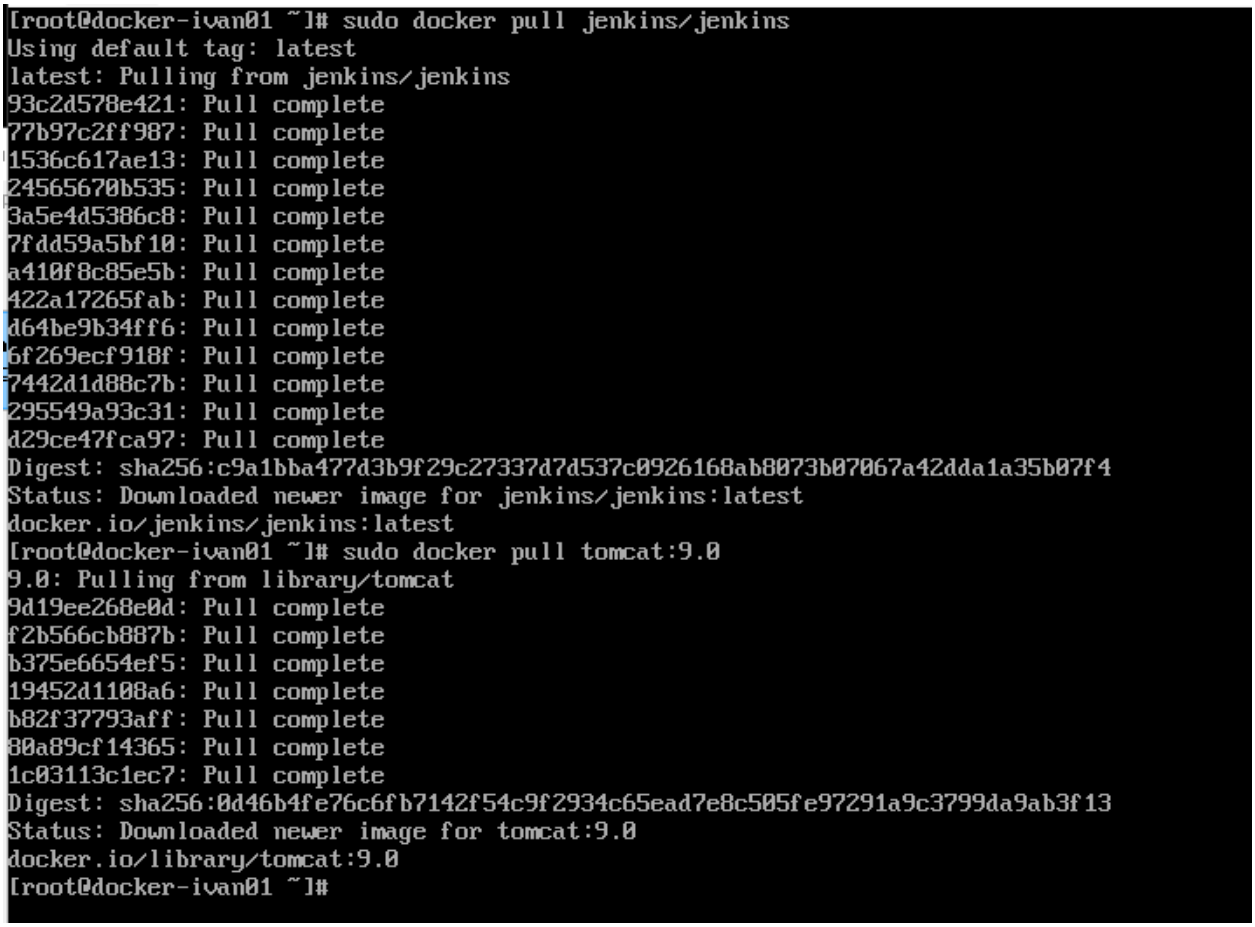
2. Pullovanje Docker image-a

Za pull-ovanje docker image koristiti sledeće komande:

```
docker pull jenkins/jenkins
```

```
docker pull tomcat:9.0
```

Prikazano na slici ispod



2.2. Pokretanje Jenkins kontejnera

Koristimo komandu kako bi pokrenuli kontejner da bi instalirali jenkins i podesili korisnika username: **admin**, passowrd: **Admin321!**

1. Porecemo kontejner kako bi kopirali fajlove za volume:

`sudo docker run --name jenkins --rm -d -p 8080:8080 jenkins/jenkins:latest`

2. Zatim koristimo sledecu komandu da kopiramo `jenkins_home` direktorijum u volume koji se nalazi na hostu

`sudo docker cp jenkins:/var/jenkins_home /sotrage/jenkins-home`

3. Menjamo permisije za ostale korisnike za direktorijume i foldere, pomocu komande date ispd koji se nalaze unutra jenkins volume-a kako bi konetejner mogao da ih koristi

`sudo chmod -R o+rwx /storage/jenkins-home/`

4. Stopiramo kontejner pomocu komande `sudo docker stop jenkins`

2.3. Porkretanje tomcat kontejnera

1. Porecemo kontejner kako bi kopirali fajlove za volume:

`sudo docker run --name tomcat --rm -d -p 8888:8080 tomcat:9.0`

2. Koristimo komandu za kopiranje tomcat direktorijuma iz kontejnera na hostu

`sudo docker cp tomcat:/usr/local/tomcat/ /storage/tomcat-home`

3. Kopiramo manager aplikaciju u webapps kako bi bila dostupna

```
[ivan@docker-ivan01 tomcat-home]$ ls
bin          CONTRIBUTING.md  logs           README.md      temp          work
BUILDING.txt lib              native-jni-lib RELEASE-NOTES  webapps
conf         LICENSE          NOTICE         RUNNING.txt    webapps.dist
[ivan@docker-ivan01 tomcat-home]$ sudo cp -r webapps.dist/. webapps/
[ivan@docker-ivan01 tomcat-home]$ ls webapps
docs  examples  host-manager  manager  ROOT
[ivan@docker-ivan01 tomcat-home]$ _
```

4. Menjamo tomcat konfiguraciju odnosno dodajemo korisnika izmenom fajla `/storage/tomcat-home/conf/tomcat-users.xml`

```
  <role rolename="manager-script"/>
  <role rolename="manager-gui"/>
  <user username="tom_deploy" password="12345678" roles="manager-gui,manager-script"/>
```

5. Menjamo `context.xml` fajl kako bi omogucili pristup tomcat manager aplikaciji.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  Licensed to the Apache Software Foundation (ASF) under one or more
  contributor license agreements.  See the NOTICE file distributed with
  this work for additional information regarding copyright ownership.
  The ASF licenses this file to You under the Apache License, Version 2.0
  (the "License"); you may not use this file except in compliance with
  the License.  You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
-->
<Context antiResourceLocking="false" privileged="true" >
  <CookieProcessor className="org.apache.tomcat.util.http.Rfc6265CookieProcessor"
                   sameSiteCookies="strict" />
<!--  <Valve className="org.apache.catalina.valves.RemoteAddrValve"
         allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" /> -->
  <Manager sessionAttributeValueClassNameFilter="java\.lang\.(?:Boolean|Integer|Long|Number|String)|
org\.apache\.catalina\.filters\.CsrfPreventionFilter\$LruCache(?:\$1)?|java\.util\.(?:Linked)?HashMa
p"/>
</Context>
```

6. Stopiramo kontejner pomocu komande `sudo docker stop tomcat`

3. Docker compse fajl

Da bi pokrenuli file potrebno je da se pozicioniramo u direktorijum u kome se nalazi fajl u ovom slucaju `/home/ivan/docker/docker-compose.yaml`

Zatim pokrenemo `sudo docker compose up -d` komandu.

```yaml
version: "3"

services:
  jenkins:
    image: jenkins/jenkins:latest
    container_name: jenkins
    restart: always
    ports:
      - 8080:8080
    volumes:
      - /sotrage/jenkins-home:/var/jenkins_home

  tomcat:
    image: tomcat:9.0
    container_name: tomcat
    restart: always
    ports:
      - 8888:8080
    volumes:
```

```
    - /storage/tomcat-home:/usr/local/tomcat

mysql:
  image: mysql
  container_name: mysql
  restart: always
  environment:
    MYSQL_ROOT_PASSWORD: ivan
    MYSQL_DATABASE: gogs
    MYSQL_USER: ivan
    MYSQL_PASSWORD: ivan
  volumes:
    - /storage/mysql-data:/var/lib/mysql

gogs:
  image: gogs
  container_name: gogs
  restart: always
  depends_on:
    - mysql
  ports:
    - 10080:3000
  volumes:
    - /storage/gogs-git-data:/data

nagios:
  image: manios/nagios:latest
  container_name: nagios
  restart: always
  ports:
    - 0.0.0.0:8090:80
  volumes:
    - /storage/nagios-data:/opt/nagios/etc/
```

3.1. Kopiranje podataka iz MySQL baze

1. Kreiramo direktorijum `scripts` u home direktorijumu korisnika (/home/ivan) pomocu sledece komande mkdir scripts

2. U direktorijumu kreiramo fajl `mysql-backup.sh` kao na slici ispod

```
#!/bin/bash

# Dest where to save backup files
dest="/storage/mysql-data-backup"

# Directory for backup
backup_directory="/storage/mysql-data"

# Current date
current_date=$(date +%Y-%m-%d)

# Archive file name
archive="mysql-data-$current_date.tar.gz"

sudo tar -zcvf $dest/$archive $backup_directory

echo "Finished"
```

3. Promenimo privilegije pomocu komande `sudo chmod ugo+x mysql-backup.sh`

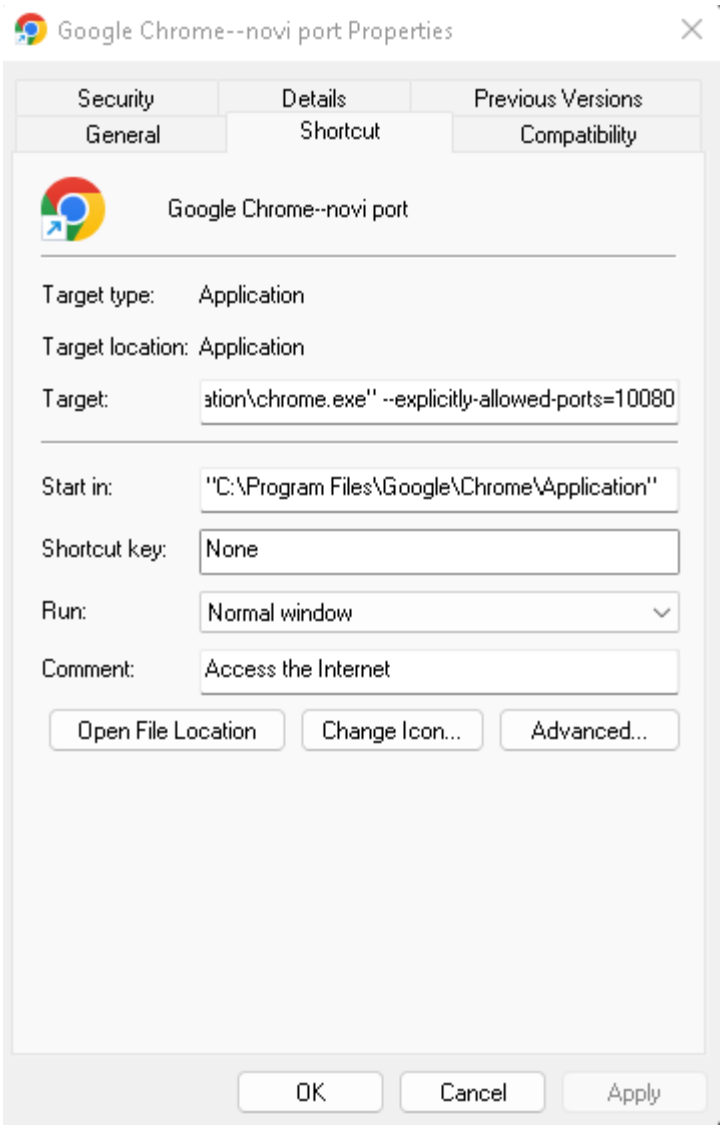
4. Otvorimo crontab pomocu sledece komande `sudo crontab -e` i dodamo sledecu liniju za izvrsavanje skripte svakog dana u 8 PM

```
# Every day at 8 PM
0 20 * * * sudo bash /home/ivan/scripts/mysql-backup.sh
```

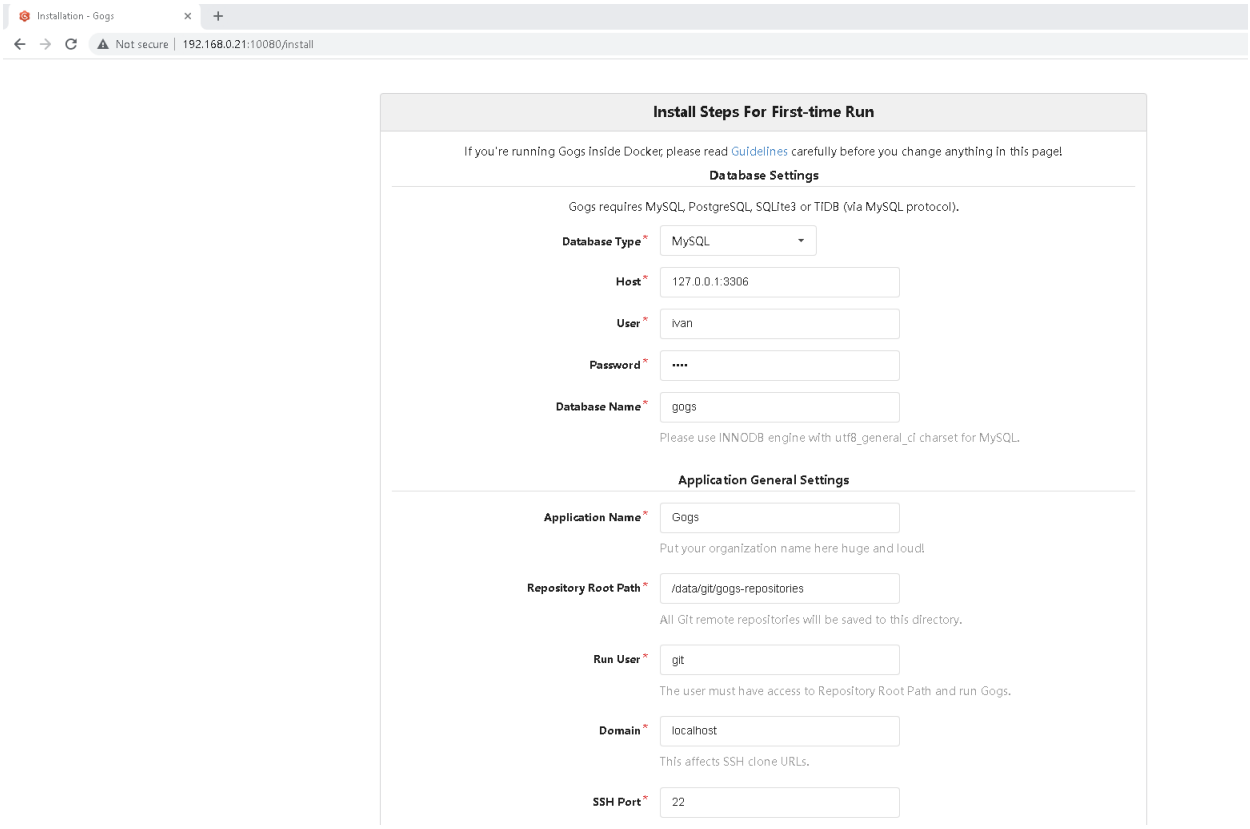
4. Instalacija gogs i Jenkins servera

4.1. Instalacija gogs servera i komitovanje projekta na server

Da bi pristupili gogs serveru preko borwsera potrebno je da eksplicitno dozvolimo port 10080 u browseru jer je blokiran to se moze uraditi na sledeci nacin dodati u target *--explicitly-alowed-ports=10080*



Nakon toga mozemo pokrenuti instalaciju gogs-a u borwseru



*Kod HOST-a je upisan mysql (naziv kontejnera MySQL baze) i pre toga je pomocu komande iz komandne linije `sudo docker exec -it mysql bash` dodeljena privilegija korisniku ivan da moze da koristi bazu namenjenu gogs-u. Nakon instalacije ulogujemo se pomocu korisnickog imena i lozinke koju smo naveli i mozemo napraviti repozitorijum

```
bash-4.4# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.33 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

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.

mysql> grant all privileges on `gogs`.* to `ivan`;
Query OK, 0 rows affected (0.02 sec)
```

Nakon toga kireramo/kloniramo repozitorijum koji je napravljen na gogs-u

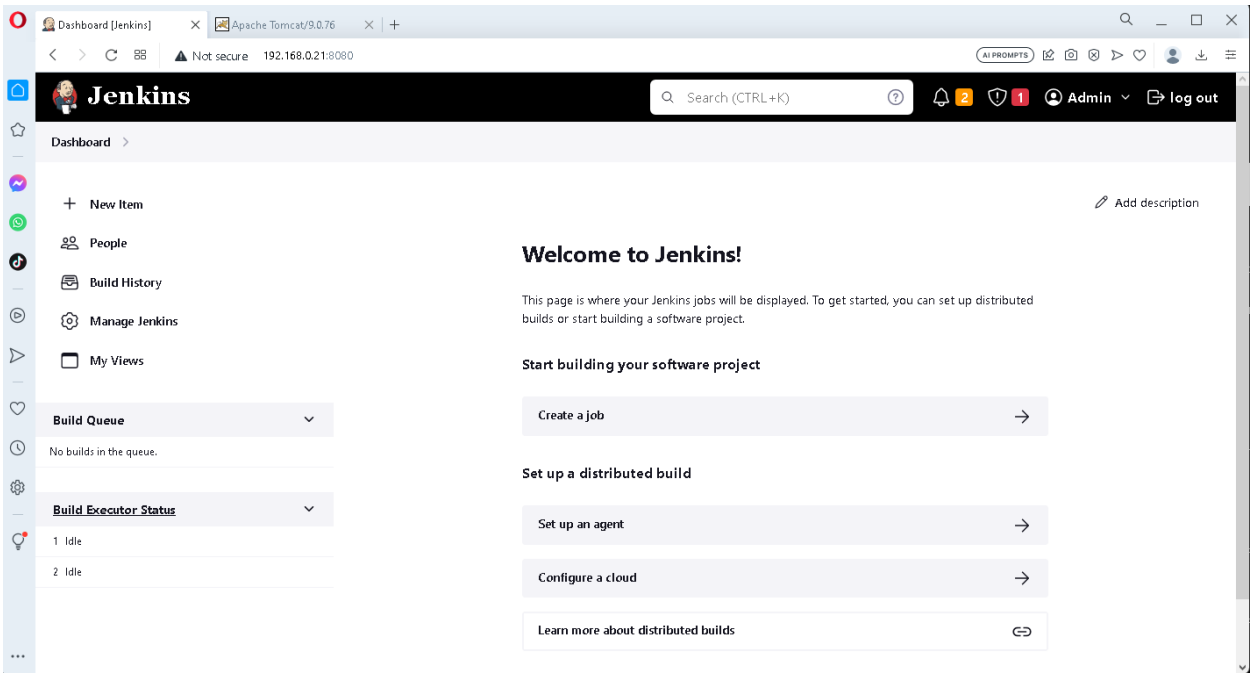
```
MyComp@DESKTOP-INJ5BML MINGW64 ~/Desktop/ivan-git
$ git clone http://192.168.0.21:10080/ivan/hello-world-war.git
Cloning into 'hello-world-war'...
warning: You appear to have cloned an empty repository.
```

Skinut projekat sa https://github.com/efsavage/hello-world-war komitujemo na gogs server

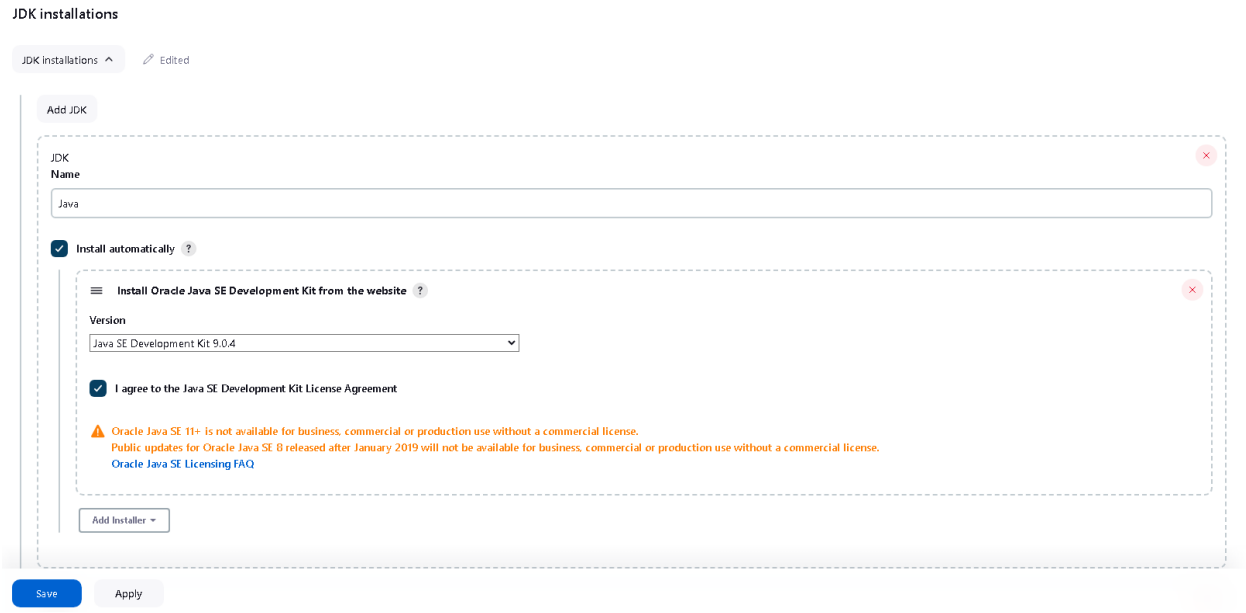
```
MyComp@DESKTOP-INJ5BML MINGW64 ~/Desktop/ivan-git/hello-world-war (master)
$ git push -u origin master
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 2 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (13/13), 2.99 KiB | 1021.00 KiB/s, done.
Total 13 (delta 0), reused 0 (delta 0), pack-reused 0
To http://192.168.0.21:10080/ivan/hello-world-war.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

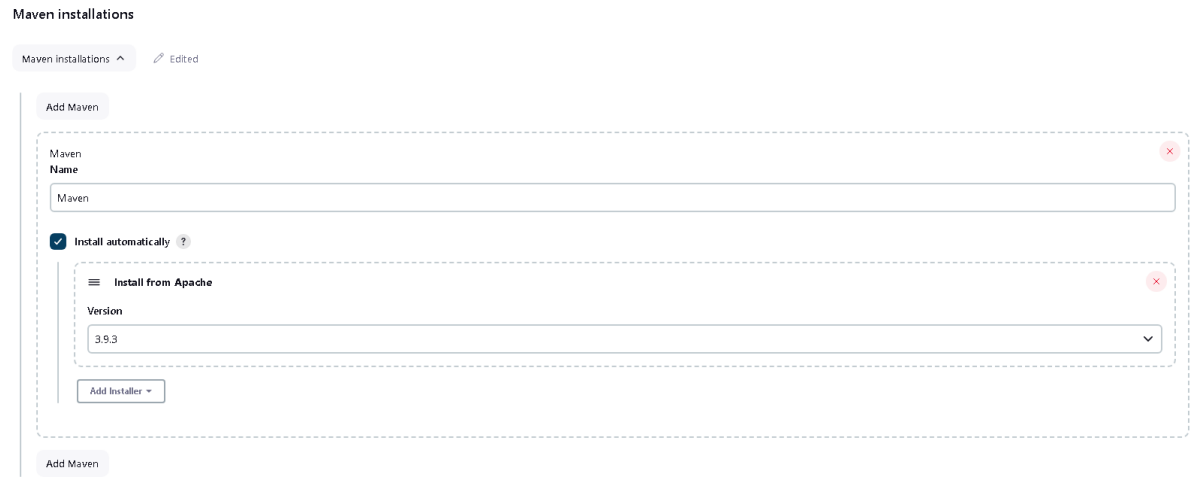
4.2. Instalacija Jenkins servera i podesavanje CI/CD-a

Kada pokrenemo jenkins kontejner u borseru upisemo ip adresu i port na kome se nalazi i mozemo pokrenuti instalaciju jenkinsa, koriscena je default instalacija.

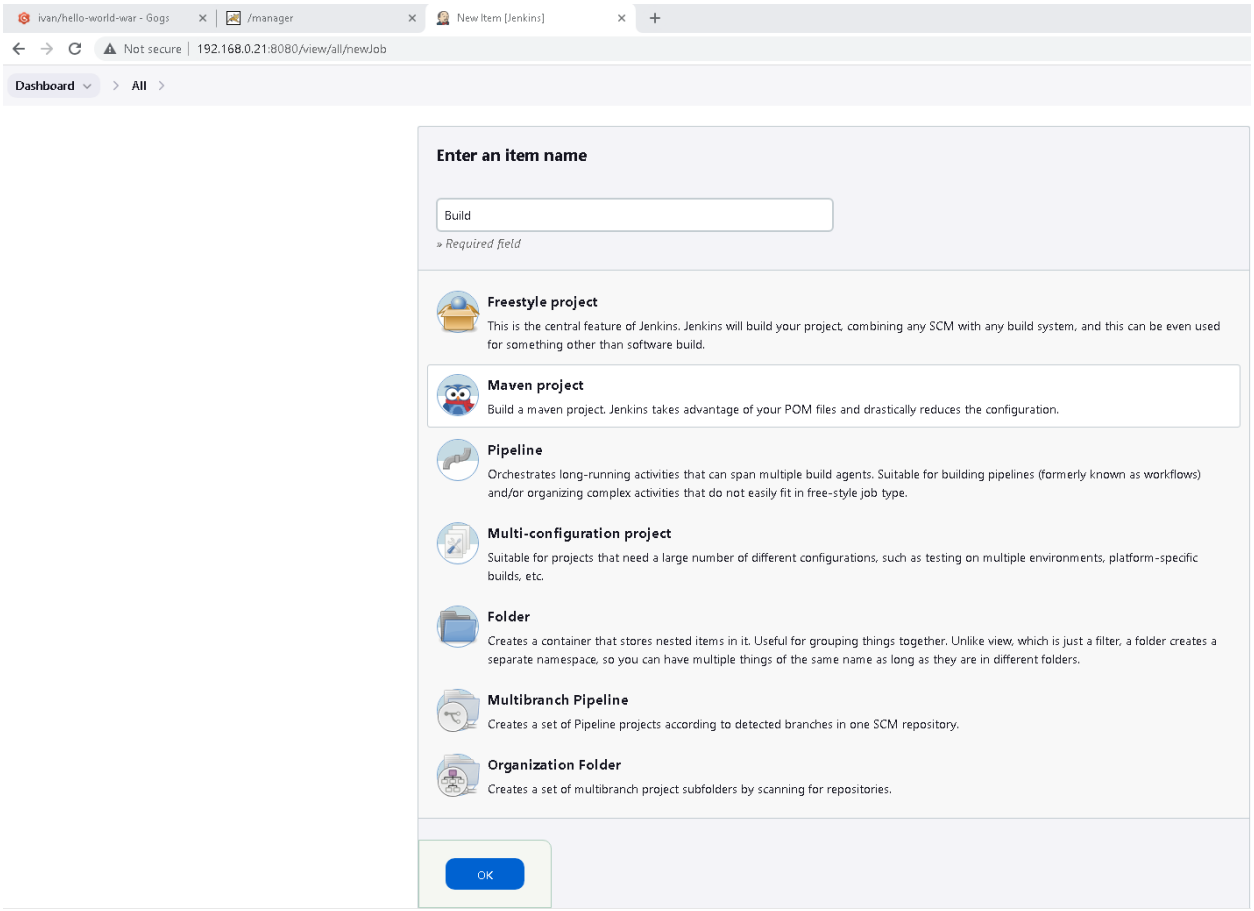


Zatim u opciji **Manage Jenkins>Tools** dodajemo instalaciju za JDK i Maven



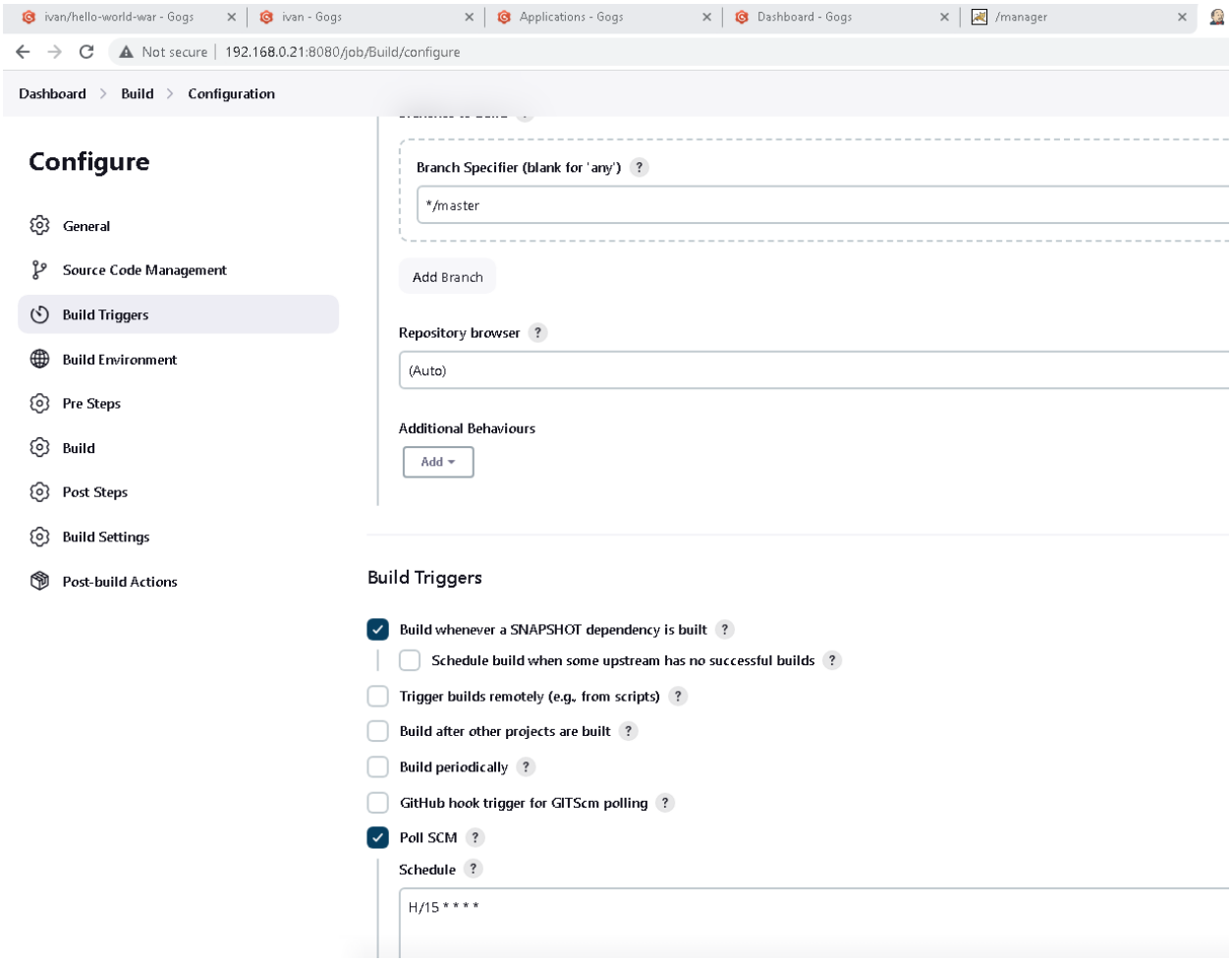


Nakon toga kreiramo CI CD pipeline klikom na **New Item**



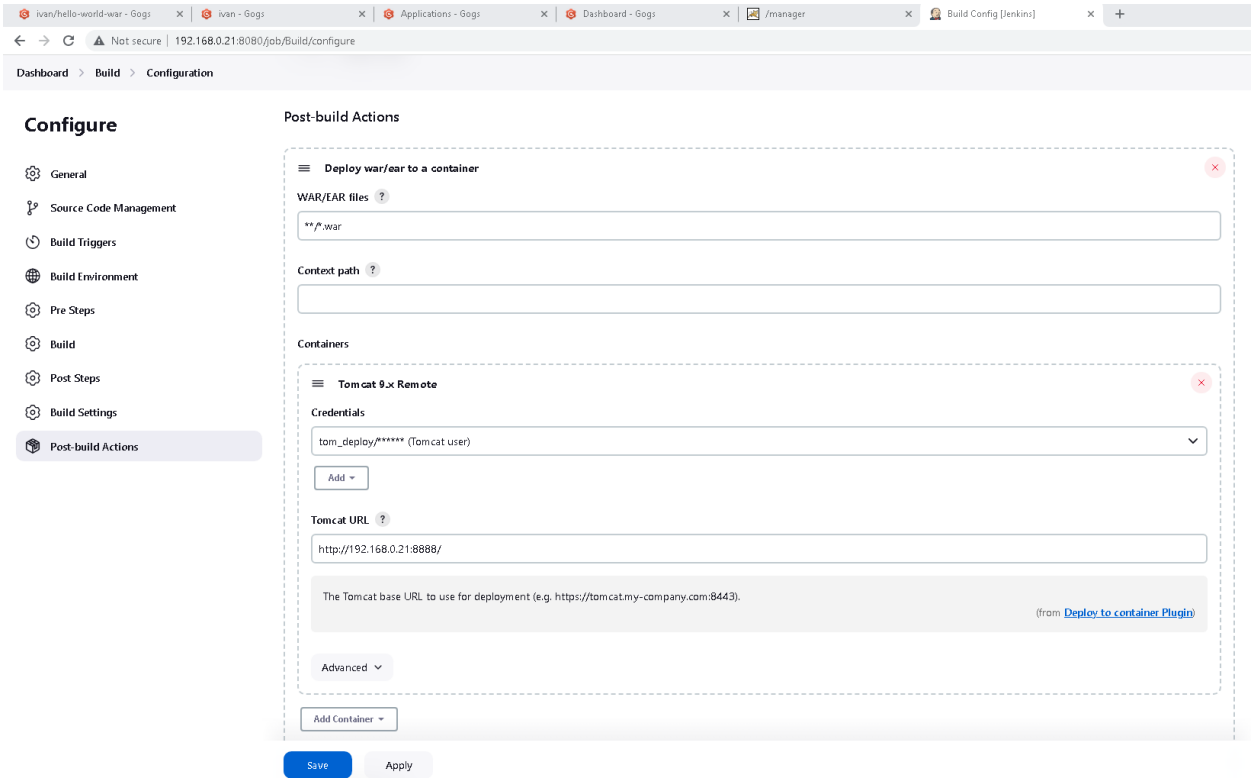
Biramo **Maven project**

Zatim **Source Code manager > Git** i unosimo repo URL kao i kredencijale za korisnika



Nakon toga biramo koju granu prati jenkins, i podesavamo SCM pool na svakih 15 minuta da proverava da li se desila promena u **master** grani.

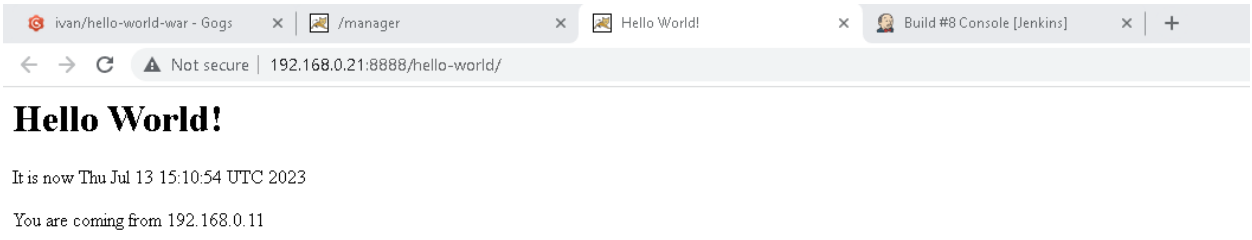
Takodje u **Post build Actions** biramo **Deply war to a container** (prethodno instaliran plugin). Gde unosimo URL odnono ip adresu tomcat servera/containera kao i kredencijale tom_deploy user i password i sacuvamo konfiguraciju.



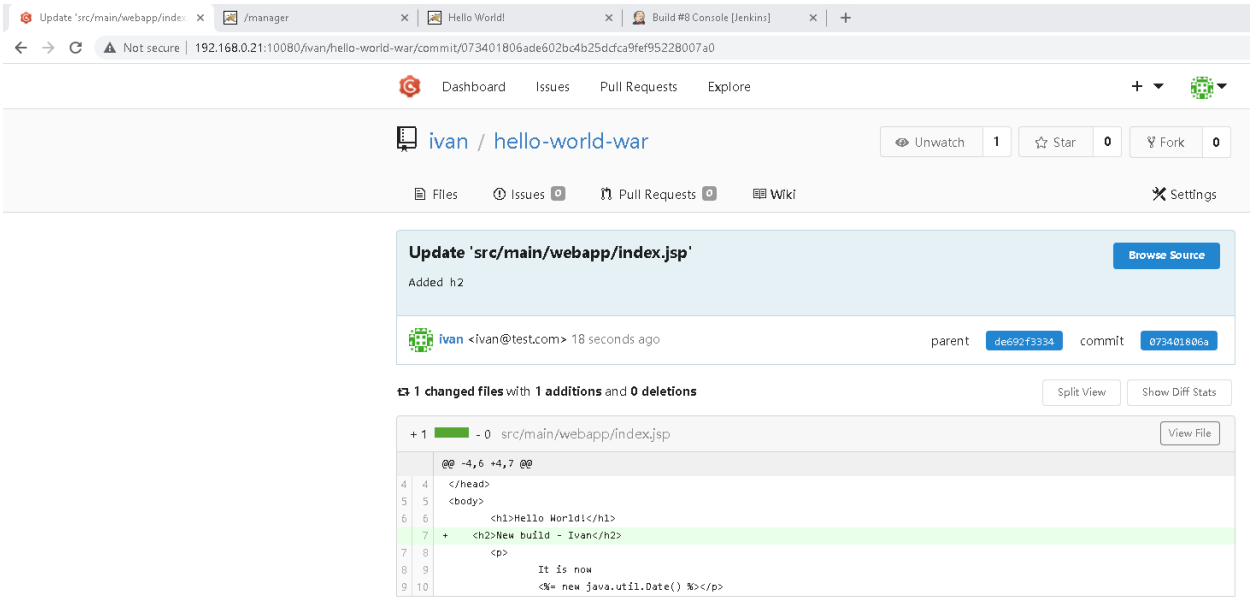
Nakon toga pocece build iz repozitorijuma i deploy na tomcat container

```
[INFO] Packaging webapp
[INFO] Assembling webapp [hello-world-war] in [/var/jenkins_home/workspace/Build/target/hello-world-war-1.0.0]
[INFO] Processing war project
[INFO] Copying webapp resources [/var/jenkins_home/workspace/Build/src/main/webapp]
[INFO] Webapp assembled in [55 msecs]
[INFO] Building war: /var/jenkins_home/workspace/Build/target/hello-world-war-1.0.0.war
[INFO] WEB-INF/web.xml already added, skipping
[INFO]
[INFO] --- install:3.1.1:install (default-install) @ hello-world-war ---
[INFO] Installing /var/jenkins_home/workspace/Build/pom.xml to /var/jenkins_home/.m2/repository/com/efsavage/hello-world-war/1.0.0/hello-world-war-1.0.0.pom
[INFO] Installing /var/jenkins_home/workspace/Build/target/hello-world-war-1.0.0.war to /var/jenkins_home/.m2/repository/com/efsavage/hello-world-war/1.0.0/hello-world-war-1.0.0.war
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  16.425 s
[INFO] Finished at: 2023-07-13T15:09:48Z
[INFO] ------------------------------------------------------------------------
Waiting for Jenkins to finish collecting data
[JENKINS] Archiving /var/jenkins_home/workspace/Build/pom.xml to com.efsavage/hello-world-war/1.0.0/hello-world-war-1.0.0.pom
[JENKINS] Archiving /var/jenkins_home/workspace/Build/target/hello-world-war-1.0.0.war to com.efsavage/hello-world-war/1.0.0/hello-world-war-1.0.0.war
channel stopped
[DeployPublisher][INFO] Attempting to deploy 2 war file(s)
[DeployPublisher][INFO] Deploying /var/jenkins_home/workspace/Build/dist/hello-world.war to container Tomcat 9.x Remote with context null
  [/var/jenkins_home/workspace/Build/dist/hello-world.war] is not deployed. Doing a fresh deployment.
  Deploying [/var/jenkins_home/workspace/Build/dist/hello-world.war]
[DeployPublisher][INFO] Deploying /var/jenkins_home/workspace/Build/target/hello-world-war-1.0.0.war to container Tomcat 9.x Remote with context null
  [/var/jenkins_home/workspace/Build/target/hello-world-war-1.0.0.war] is not deployed. Doing a fresh deployment.
  Deploying [/var/jenkins_home/workspace/Build/target/hello-world-war-1.0.0.war]
Finished: SUCCESS
```

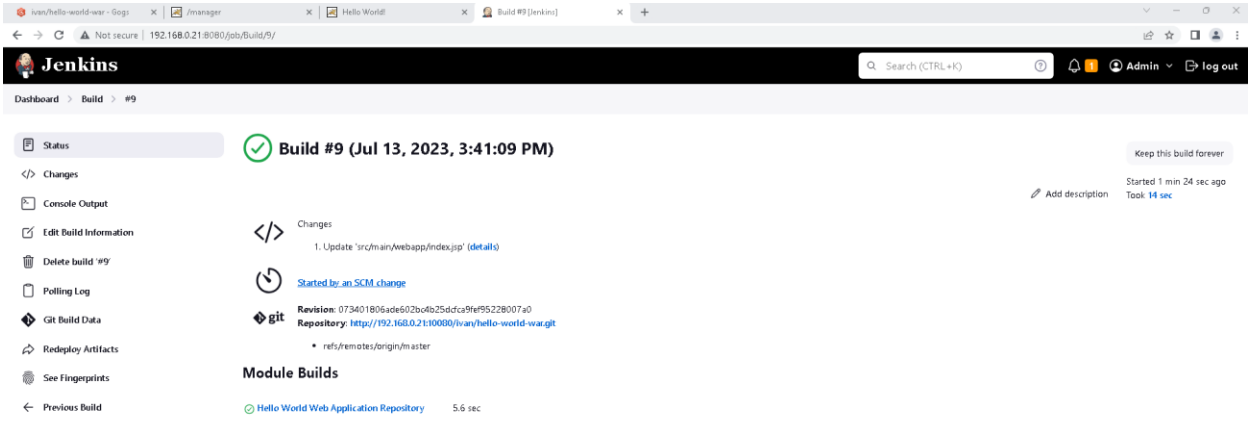
Buildovana aplikacija i ispod slika deploy na tomcat odnosno aplikacija koja je pokrenuta



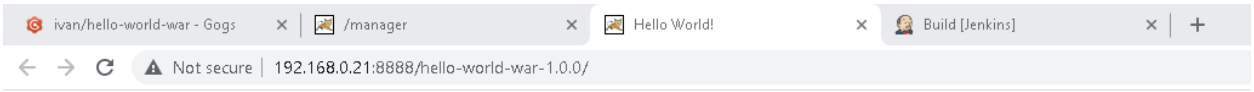
Sada mozemo da komitujemo neku promenu u repozitorijum



Npr. dodati neki novi element u HTML-u kao na slici i sacekom jenskins da pokrene build posle nekoliko minuta.



Posle toga mozemo otvoriti i novu web aplikaciju sa izmenom



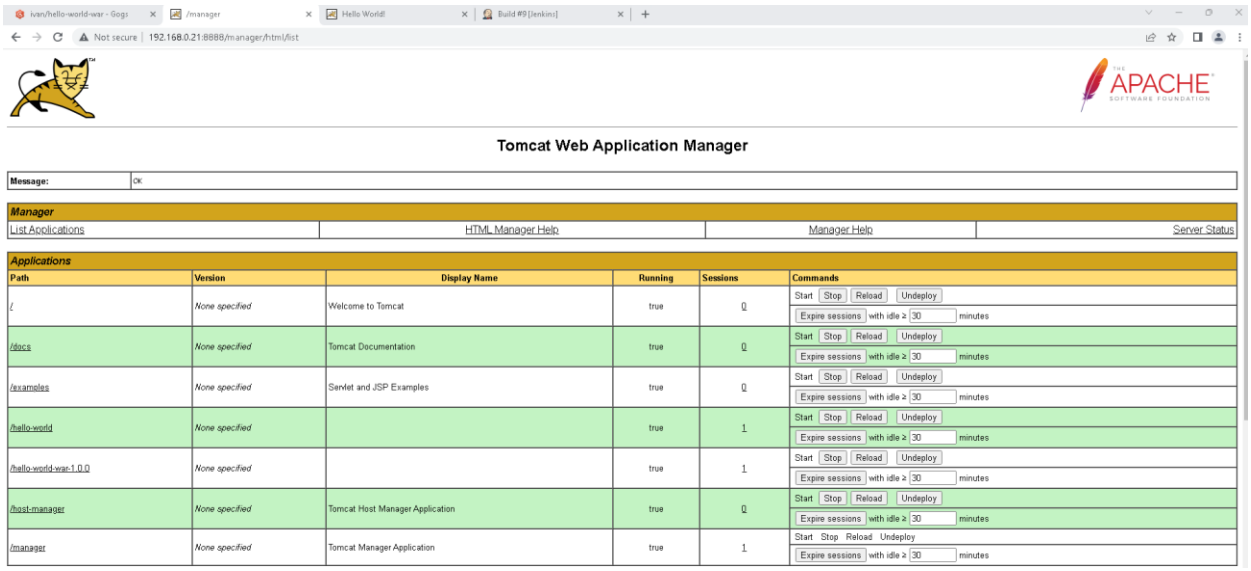
Hello World!

New build - Ivan

It is now Thu Jul 13 15:41:50 UTC 2023

You are coming from 192.168.0.11

Koju takodje mozemo videti i u tomcat manageru



5. Podesavanje Nagios servera

1. Za pokretanje Nagios servera koristimo docker kontejner manios/nagios, pullujemo kontejner pomocu sledece komande

```
docker pull manios/nagios:latest
```

2. Koristimo komandu da kopiramo konfiguracione fajlove iz kotnejnera na hostu.

```
sudo docker cp nagios:/opt/nagios/etc /storage/nagios-data
```

3. Dodamo fajl sa sadrzajem kao na slici ispod, na hostu gde planiramo da napravimo volume za nagios kontejner u ovom slucaju `/storage/nagios-data/objects/docker.cfg` u kome se nalaze podaci o hostovima koje je potrebno pratiti odnosno kontejnerima.

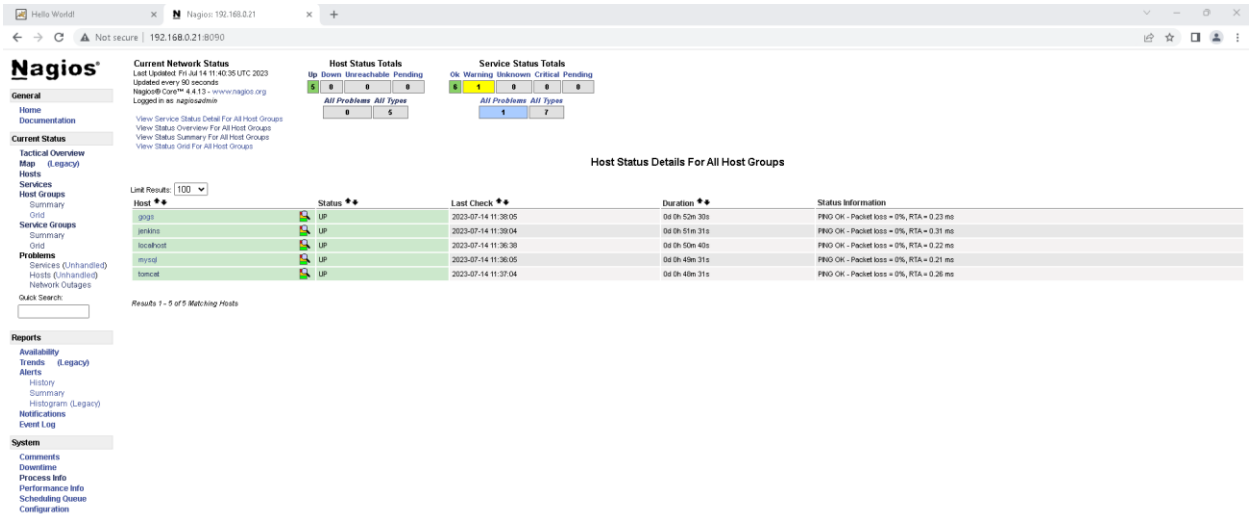
```
define host {
    use                 linux-server
    host_name           jenkins
    alias               JENKINS-SERVER
    address             jenkins
}

define host {
    use                 linux-server
    host_name           tomcat
    alias               TOMCAT-SERVER
    address             tomcat
}
define host {
    use                 linux-server
    host_name           gogs
    alias               GOGS-SERVER
    address             gogs
}
define host {
    use                 linux-server
    host_name           mysql
    alias               MYSQL-SERVER
    address             mysql
}
```

4. Dodamo putanju do prethodno kreiranog docker.cfg fajla u fajlu koji se nalazi u `/storage/nagios-data/nagios.cfg` kao na slici ispod

```
# Definitions for monitoring the local (Linux) host
cfg_file=/opt/nagios/etc/objects/localhost.cfg
cfg_file=/opt/nagios/etc/objects/docker.cfg
```

5. Pokrecemo kontejnere pomocu docker compose fajla.

6. Nakon toga mozemo da otvorimo nagios u borwseru na adresi 192.168.0.21:8090 kao na slici ispod gde mozemo da vidimo statuse hostova odnosno kontejnera.


Hello World!
Nagios: 192.168.0.21
Not secure | 192.168.0.21:8090
Nagios
General
Home
Documentation
Current Status
Tactical Overview
Map (Legacy)
Hosts
Services
Host Groups
Summary
Grid
Service Groups
Summary
Grid
Problems
Services (Unhandled)
Hosts (Unhandled)
Network Outages
Quick Search:
Reports
Availability
Trends (Legacy)
Alerts
History
Summary
Histogram (Legacy)
Notifications
Event Log
System
Comments
Downtime
Process Info
Performance Info
Scheduling Queue
Configuration
Current Network Status
Last Updated: Fri Jul 14 11:40:35 UTC 2023
Updated every 90 seconds
Nagios® Core™ 4.4.13 - www.nagios.org
Logged in as nagiosadmin
View Service Status Detail For All Host Groups
View Status Overview For All Host Groups
View Status Summary For All Host Groups
View Status Grid For All Host Groups
Host Status Totals
Up Down Unreachable Pending
5 0 0 0
All Problems All Types
0 5
Service Status Totals
Ok Warning Unknown Critical Pending
5 1 0 0 0
All Problems All Types
1 7
Host Status Details For All Host Groups
Limit Results: 100
Host
Status
Last Check
Duration
Status Information
gogs UP 2023-07-14 11:38:05 0d 0h 52m 30s PING OK - Packet loss = 0%, RTA = 0.23 ms
jenkins UP 2023-07-14 11:39:04 0d 0h 51m 31s PING OK - Packet loss = 0%, RTA = 0.31 ms
localhost UP 2023-07-14 11:36:38 0d 0h 50m 40s PING OK - Packet loss = 0%, RTA = 0.22 ms
mysql UP 2023-07-14 11:36:05 0d 0h 49m 31s PING OK - Packet loss = 0%, RTA = 0.21 ms
tomcat UP 2023-07-14 11:37:04 0d 0h 48m 31s PING OK - Packet loss = 0%, RTA = 0.26 ms
Results 1 - 5 of 5 Matching Hosts