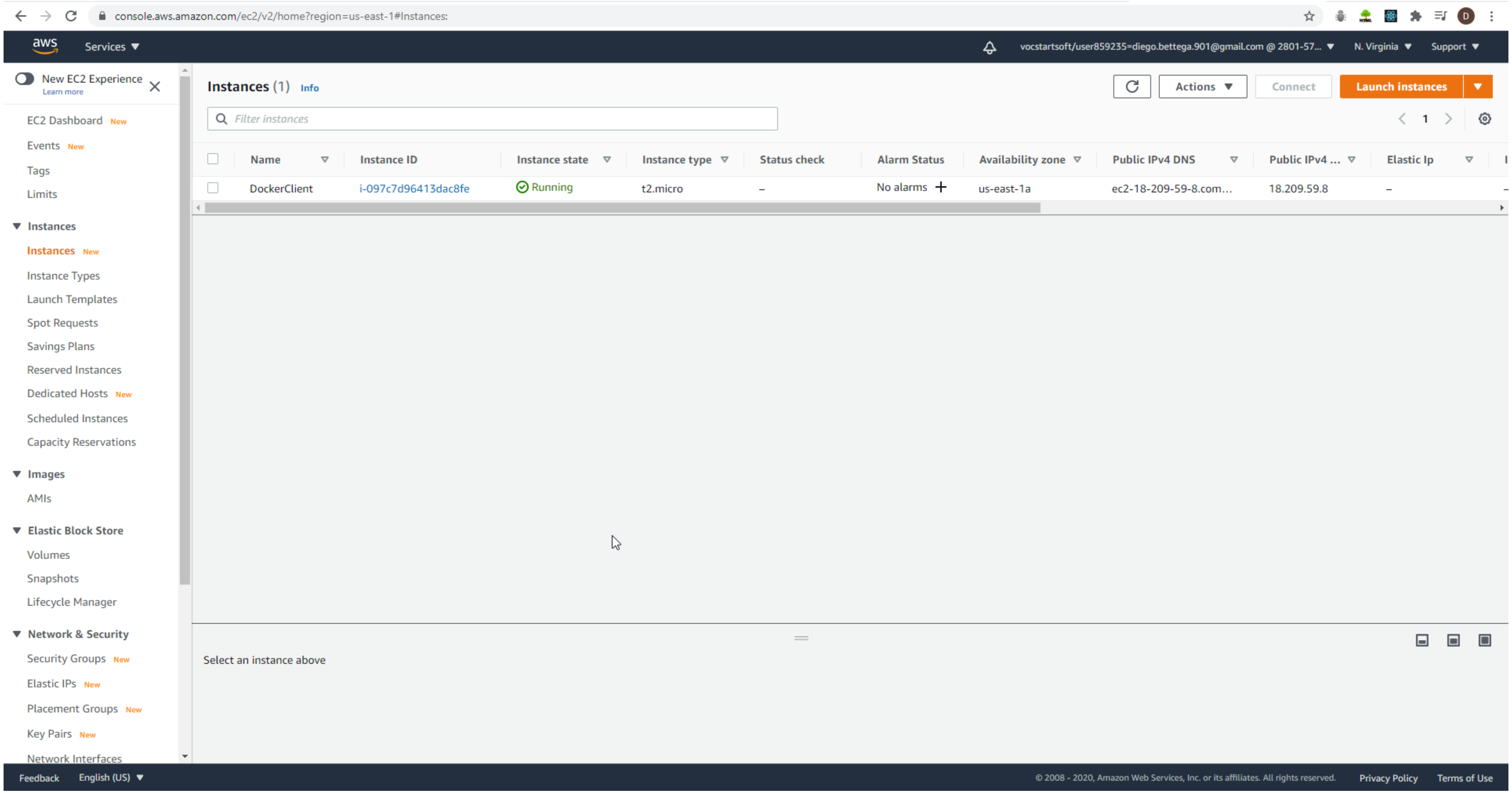
Create the EC2 instance

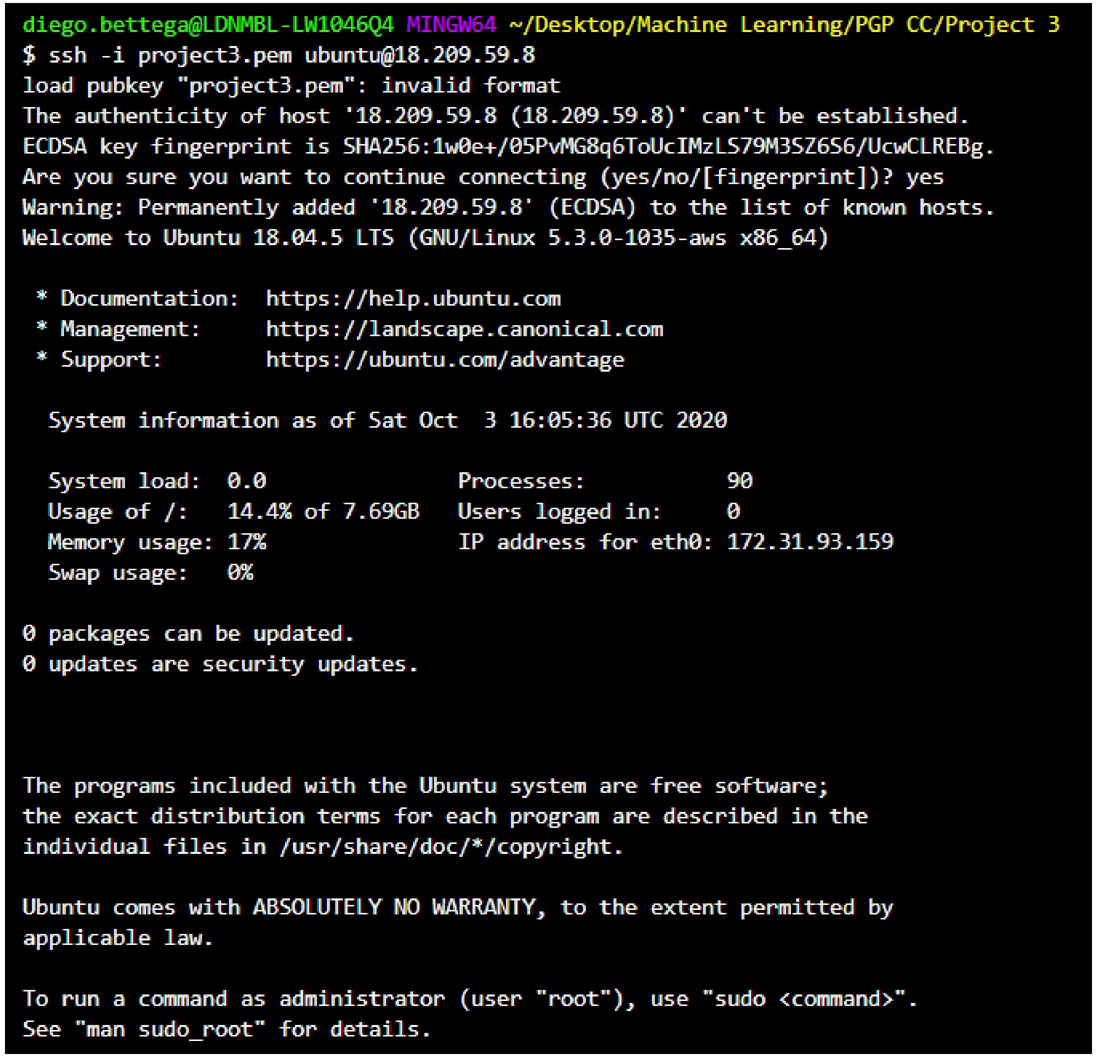


Instances (1) Info

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm Status	Availability zone	Public IPv4 DNS	Public IPv4 ...	Elastic Ip
□	DockerClient	i-097c7d96413dac8fe	✓ Running	t2.micro	–	No alarms +	us-east-1a	ec2-18-209-59-8.com...	18.209.59.8	–

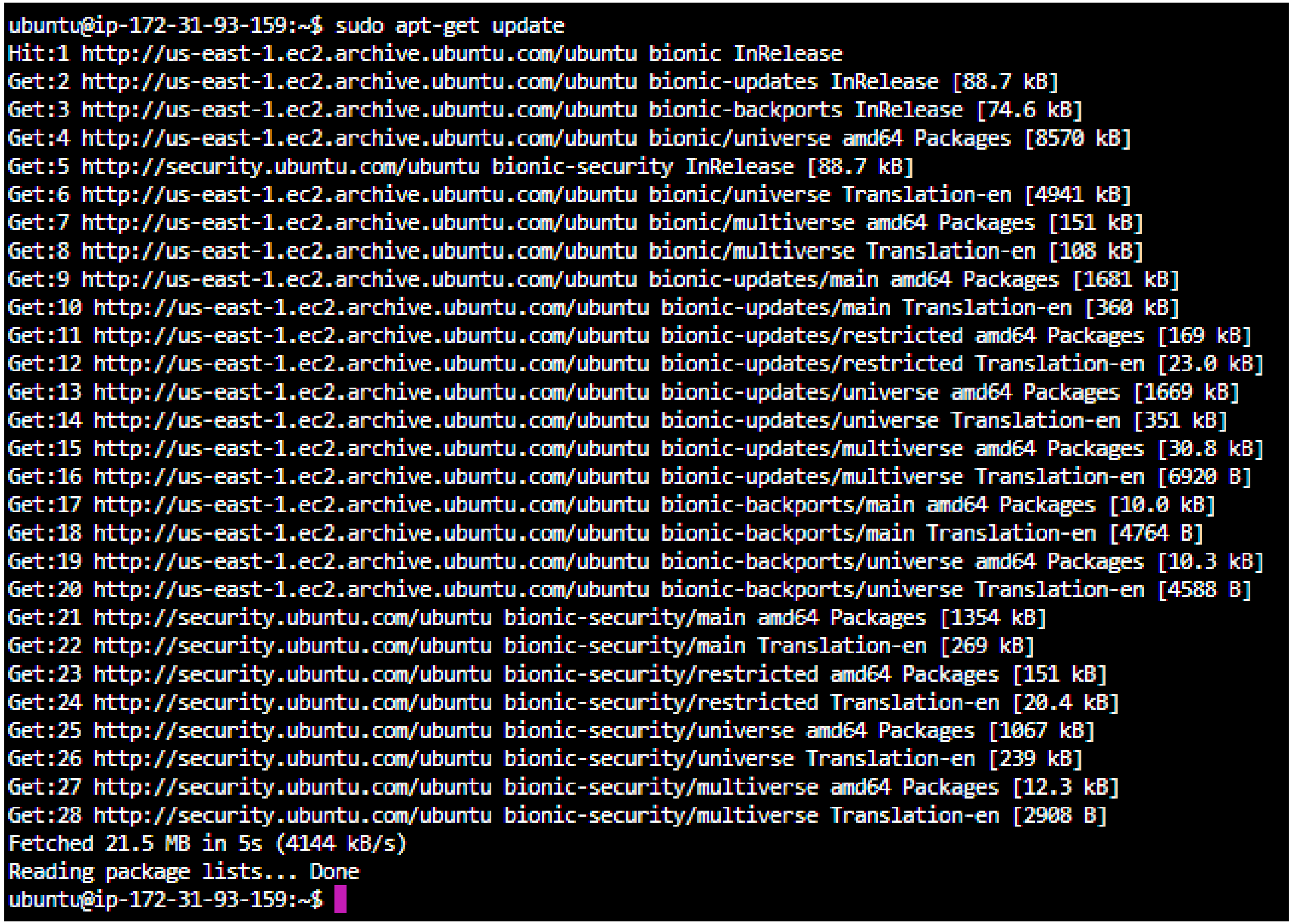
Select an instance above

SSH in EC2 instance



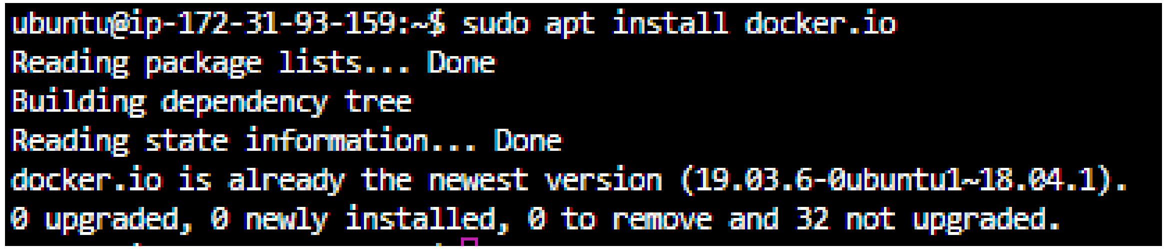
```
diego.bettega@LDNMBL-LW1046Q4 MINGW64 ~/Desktop/Machine Learning/PGP CC/Project 3
$ ssh -i project3.pem ubuntu@18.209.59.8
load pubkey "project3.pem": invalid format
The authenticity of host '18.209.59.8 (18.209.59.8)' can't be established.
ECDSA key fingerprint is SHA256:1w0e+/05PvMG8q6ToUcIMzLS79M3SZ6S6/UcwCLREBg.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '18.209.59.8' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.3.0-1035-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sat Oct  3 16:05:36 UTC 2020

  System load:  0.0               Processes:           90
  Usage of /:   14.4% of 7.69GB   Users logged in:     0
  Memory usage: 17%               IP address for eth0: 172.31.93.159
  Swap usage:   0%

0 packages can be updated.
0 updates are security updates.



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```

Update the Ubuntu Packages



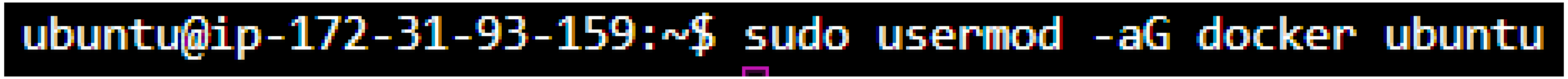
```
ubuntu@ip-172-31-93-159:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [8570 kB]
Get:5 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/universe Translation-en [4941 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/multiverse amd64 Packages [151 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic/multiverse Translation-en [108 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [1681 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/main Translation-en [360 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/restricted amd64 Packages [169 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/restricted Translation-en [23.0 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [1669 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/universe Translation-en [351 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/multiverse amd64 Packages [30.8 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/multiverse Translation-en [6920 B]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-backports/main amd64 Packages [10.0 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-backports/main Translation-en [4764 B]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-backports/universe amd64 Packages [10.3 kB]
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-backports/universe Translation-en [4588 B]
Get:21 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [1354 kB]
Get:22 http://security.ubuntu.com/ubuntu bionic-security/main Translation-en [269 kB]
Get:23 http://security.ubuntu.com/ubuntu bionic-security/restricted amd64 Packages [151 kB]
Get:24 http://security.ubuntu.com/ubuntu bionic-security/restricted Translation-en [20.4 kB]
Get:25 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 Packages [1067 kB]
Get:26 http://security.ubuntu.com/ubuntu bionic-security/universe Translation-en [239 kB]
Get:27 http://security.ubuntu.com/ubuntu bionic-security/multiverse amd64 Packages [12.3 kB]
Get:28 http://security.ubuntu.com/ubuntu bionic-security/multiverse Translation-en [2908 B]
Fetched 21.5 MB in 5s (4144 kB/s)
Reading package lists... Done
ubuntu@ip-172-31-93-159:~$
```

Install Docker Client



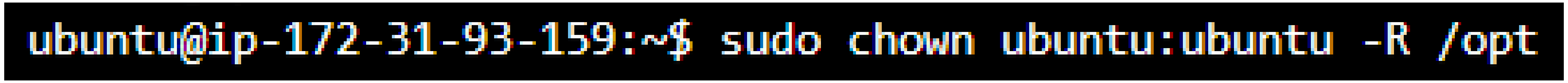
```
ubuntu@ip-172-31-93-159:~$ sudo apt install docker.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
docker.io is already the newest version (19.03.6-0ubuntu1~18.04.1).
0 upgraded, 0 newly installed, 0 to remove and 32 not upgraded.
```

Add Docker to the Ubuntu group

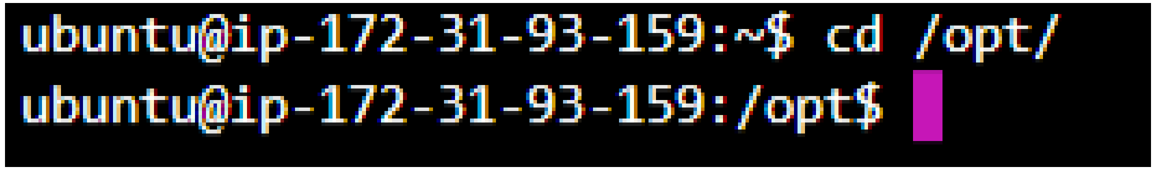


```
ubuntu@ip-172-31-93-159:~$ sudo usermod -aG docker ubuntu
```

Create and cd in the opt folder

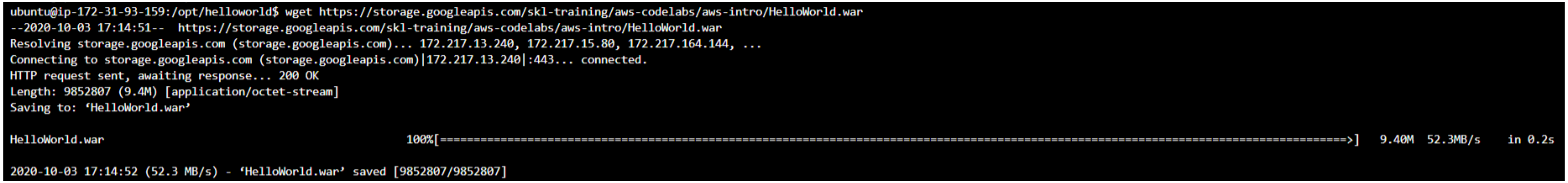


```
ubuntu@ip-172-31-93-159:~$ sudo chown ubuntu:ubuntu -R /opt
```



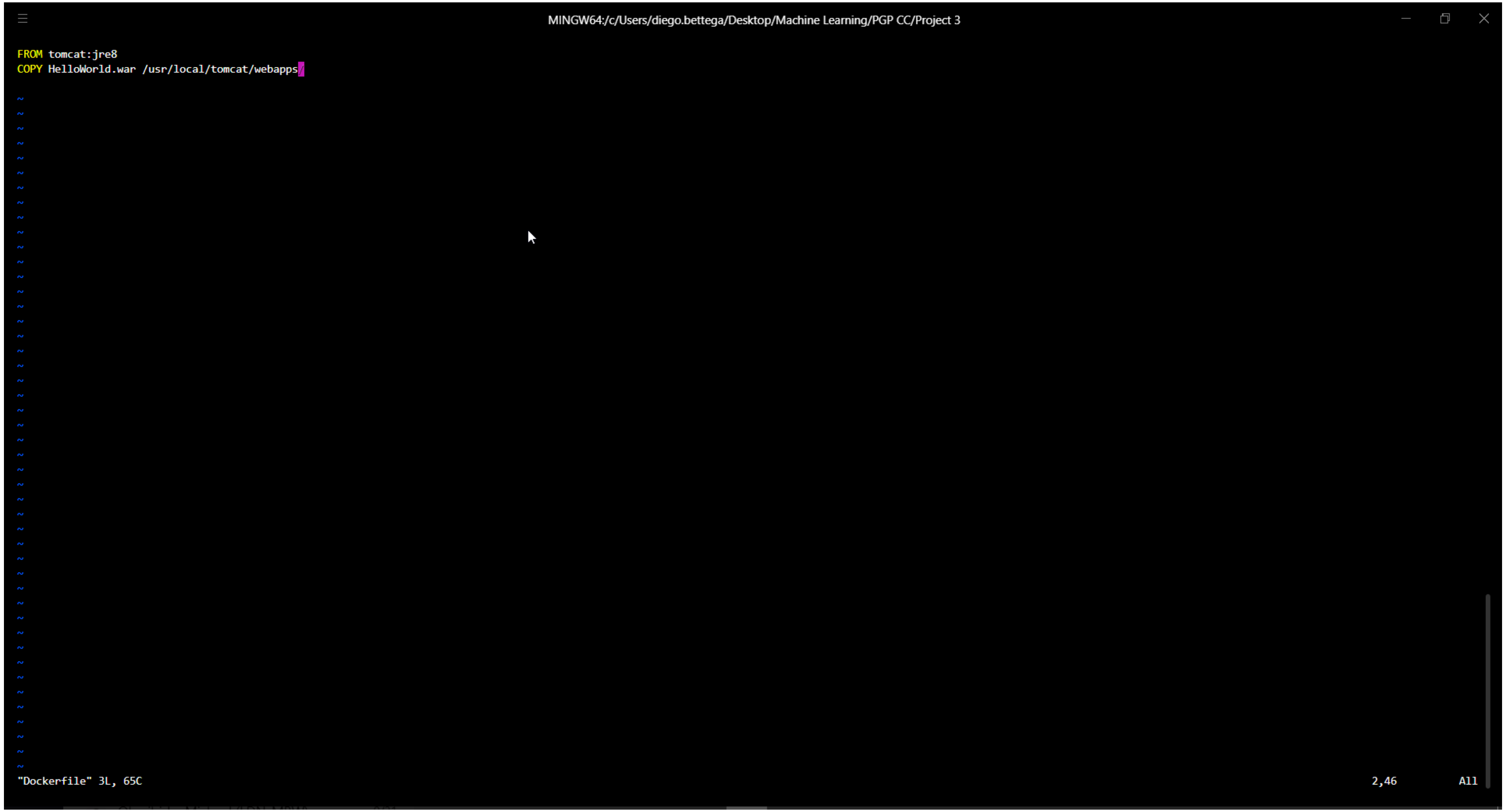
```
ubuntu@ip-172-31-93-159:~$ cd /opt/
ubuntu@ip-172-31-93-159:/opt$
```

Download the sample web app



```
ubuntu@ip-172-31-93-159:/opt/helloworld$ wget https://storage.googleapis.com/skl-training/aws-codelabs/aws-intro/HelloWorld.war
--2020-10-03 17:14:51--  https://storage.googleapis.com/skl-training/aws-codelabs/aws-intro/HelloWorld.war
Resolving storage.googleapis.com (storage.googleapis.com)... 172.217.13.240, 172.217.15.80, 172.217.164.144, ...
Connecting to storage.googleapis.com (storage.googleapis.com)|172.217.13.240|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 9852807 (9.4M) [application/octet-stream]
Saving to: ‘HelloWorld.war’

HelloWorld.war                                             100%[======================================================================================================================================>]   9.40M  52.3MB/s    in 0.2s

2020-10-03 17:14:52 (52.3 MB/s) - ‘HelloWorld.war’ saved [9852807/9852807]
```

Create the Docker file

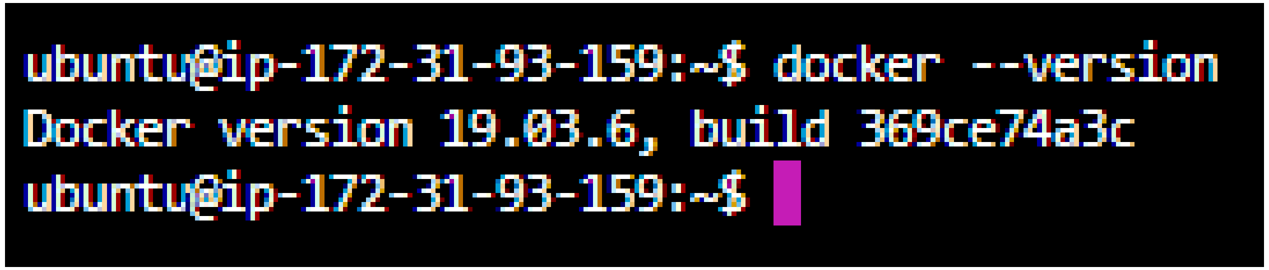


MINGW64:/c/Users/diego.bettega/Desktop/Machine Learning/PGP CC/Project 3

```
FROM tomcat:jre8
COPY HelloWorld.war /usr/local/tomcat/webapps/
```

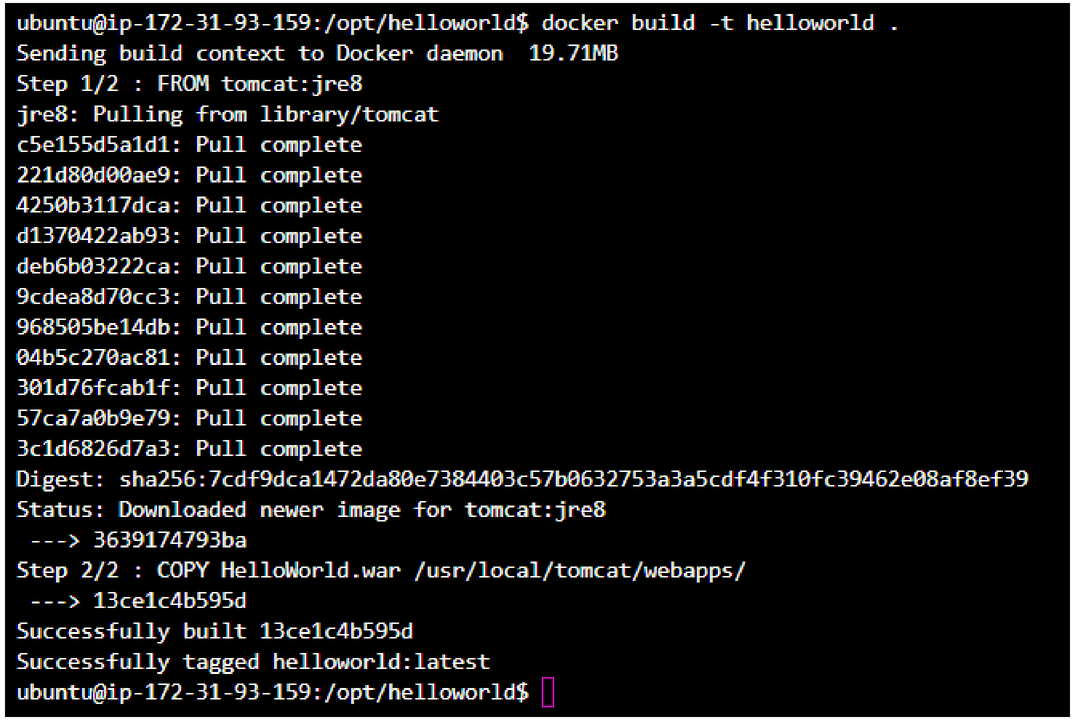
```
"Dockerfile" 3L, 65C                                                    2,46          All
```

Verify the Docker is properly installed



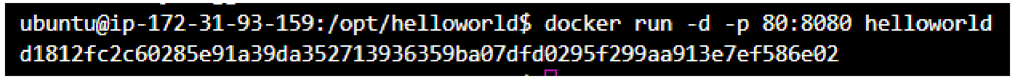
```
ubuntu@ip-172-31-93-159:~$ docker --version
Docker version 19.03.6, build 369ce74a3c
ubuntu@ip-172-31-93-159:~$
```

Run the build command that bring in the required image to build the custom application



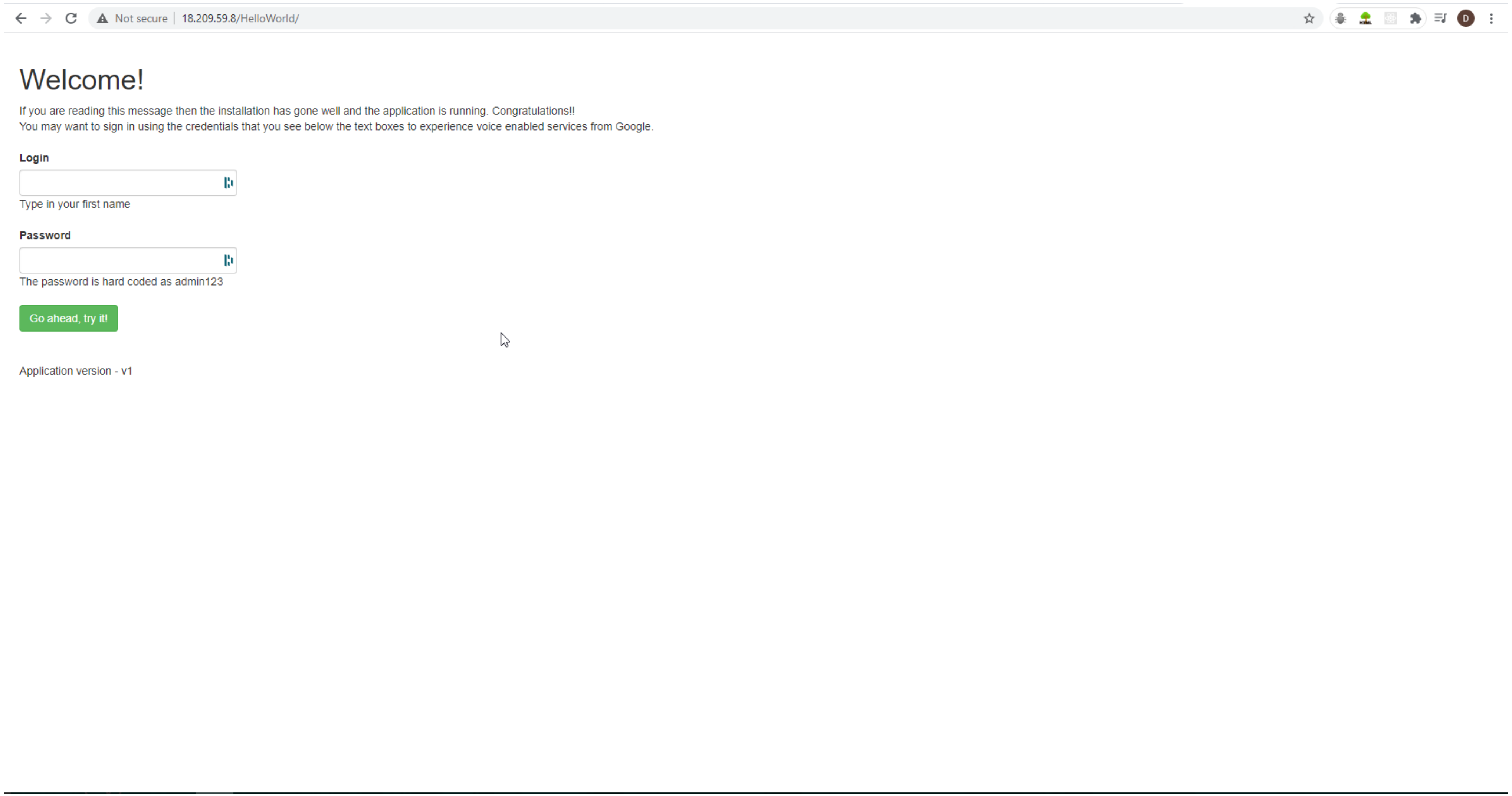
```
ubuntu@ip-172-31-93-159:/opt/helloworld$ docker build -t helloworld .
Sending build context to Docker daemon  19.71MB
Step 1/2 : FROM tomcat:jre8
jre8: Pulling from library/tomcat
c5e155d5a1d1: Pull complete
221d80d00ae9: Pull complete
4250b3117dca: Pull complete
d1370422ab93: Pull complete
deb6b03222ca: Pull complete
9cdea8d70cc3: Pull complete
968505be14db: Pull complete
04b5c270ac81: Pull complete
301d76fcab1f: Pull complete
57ca7a0b9e79: Pull complete
3c1d6826d7a3: Pull complete
Digest: sha256:7cdf9dca1472da80e7384403c57b0632753a3a5cdf4f310fc39462e08af8ef39
Status: Downloaded newer image for tomcat:jre8
 ---> 3639174793ba
Step 2/2 : COPY HelloWorld.war /usr/local/tomcat/webapps/
 ---> 13ce1c4b595d
Successfully built 13ce1c4b595d
Successfully tagged helloworld:latest
ubuntu@ip-172-31-93-159:/opt/helloworld$
```

Verify the image is accessible from local box



```
ubuntu@ip-172-31-93-159:/opt/helloworld$ docker run -d -p 80:8080 helloworld
d1812fc2c60285e91a39da352713936359ba07dfd0295f299aa913e7ef586e02
```

Access the application by using the public IP address



Welcome!

If you are reading this message then the installation has gone well and the application is running. Congratulations!!
You may want to sign in using the credentials that you see below the text boxes to experience voice enabled services from Google.

Login

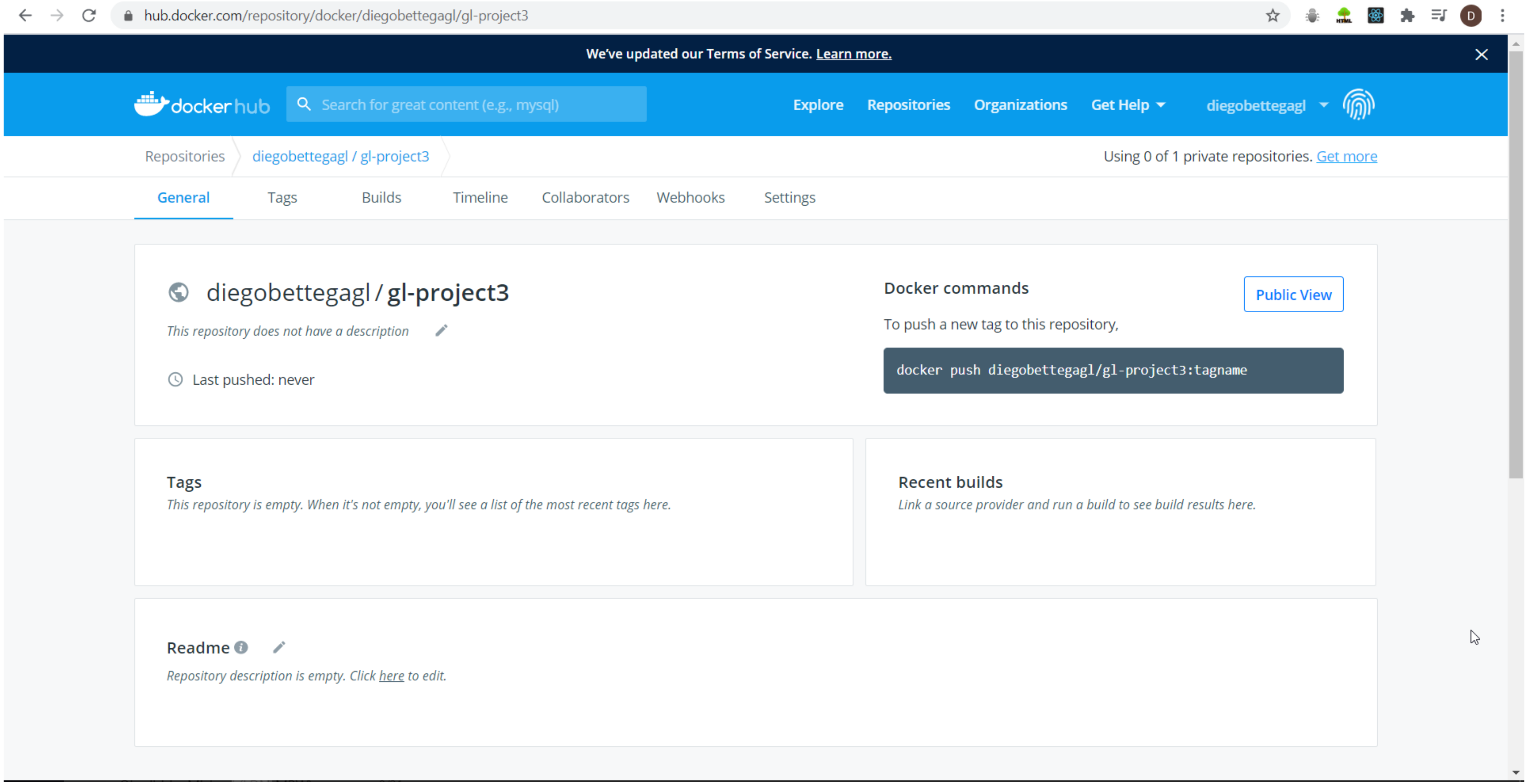
Type in your first name

Password

The password is hard coded as admin123

Application version - v1

Create a public repository on Docker Hub



hub.docker.com/repository/docker/diegobettegagl/gl-project3

We've updated our Terms of Service. Learn more.

Using 0 of 1 private repositories. Get more

diegobettegagl / gl-project3

This repository does not have a description

Last pushed: never

Docker commands

To push a new tag to this repository,

```
docker push diegobettegagl/gl-project3:tagname
```

Tags

This repository is empty. When it's not empty, you'll see a list of the most recent tags here.

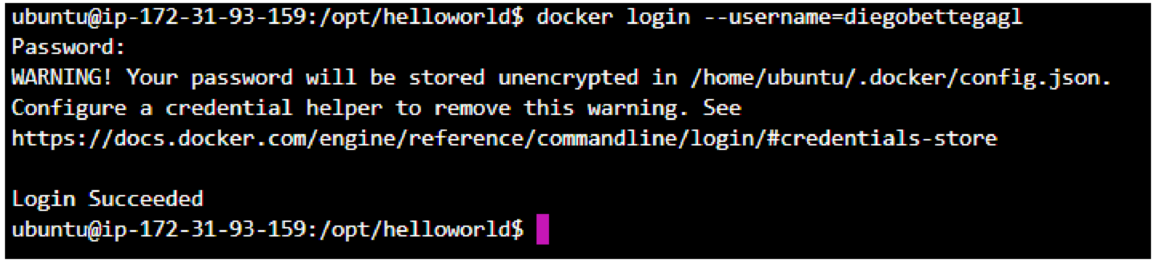
Recent builds

Link a source provider and run a build to see build results here.

Readme

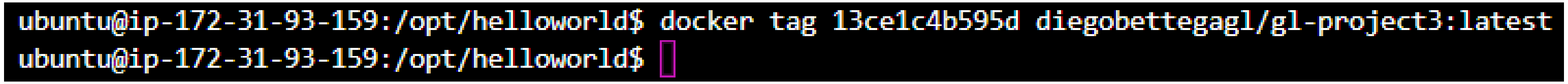
Repository description is empty. Click here to edit.

Login to the Docker Hub



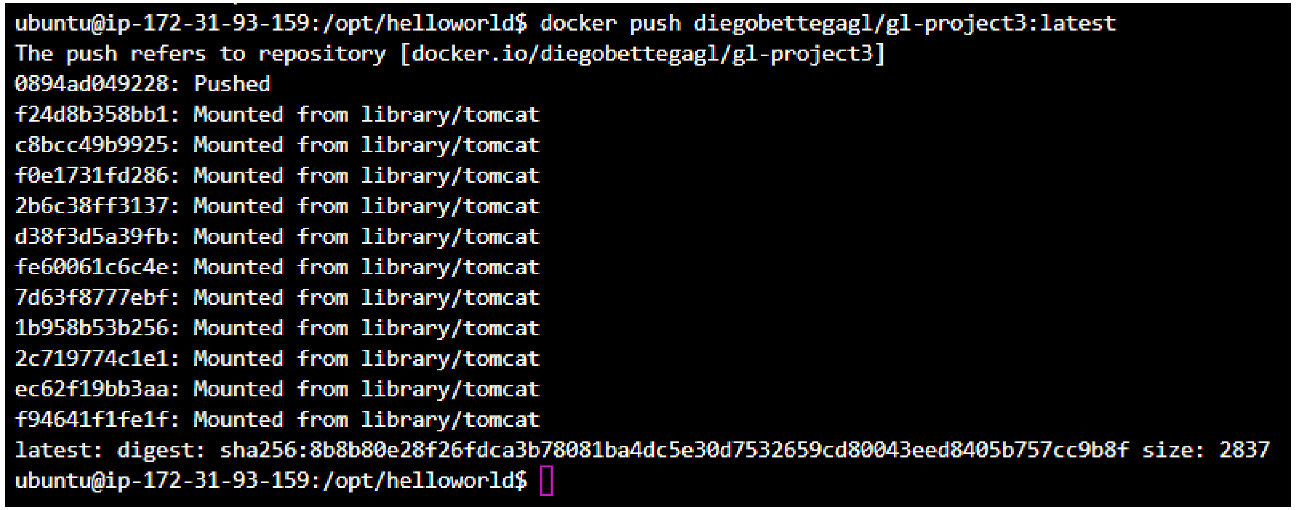
```
ubuntu@ip-172-31-93-159:/opt/helloworld$ docker login --username=diegobettegagl
Password:
WARNING! Your password will be stored unencrypted in /home/ubuntu/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
ubuntu@ip-172-31-93-159:/opt/helloworld$
```

Create the tag with the correct image ID



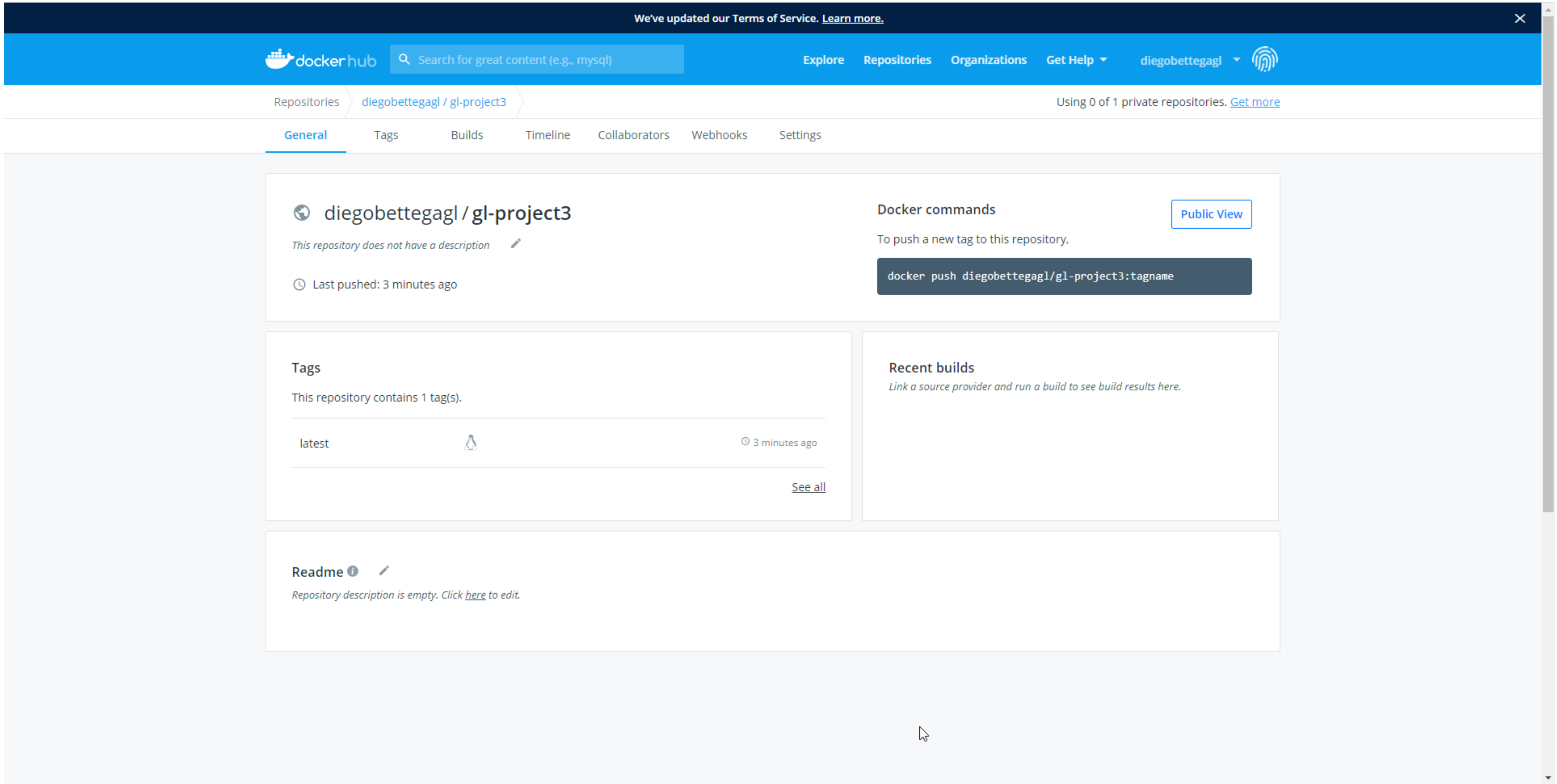
```
ubuntu@ip-172-31-93-159:/opt/helloworld$ docker tag 13ce1c4b595d diegobettegagl/gl-project3:latest
ubuntu@ip-172-31-93-159:/opt/helloworld$ 
```

Push the local image on to the Docker



```
ubuntu@ip-172-31-93-159:/opt/helloworld$ docker push diegobettegagl/gl-project3:latest
The push refers to repository [docker.io/diegobettegagl/gl-project3]
0894ad049228: Pushed
f24d8b358bb1: Mounted from library/tomcat
c8bcc49b9925: Mounted from library/tomcat
f0e1731fd286: Mounted from library/tomcat
2b6c38ff3137: Mounted from library/tomcat
d38f3d5a39fb: Mounted from library/tomcat
fe60061c6c4e: Mounted from library/tomcat
7d63f8777ebf: Mounted from library/tomcat
1b958b53b256: Mounted from library/tomcat
2c719774c1e1: Mounted from library/tomcat
ec62f19bb3aa: Mounted from library/tomcat
f94641f1fe1f: Mounted from library/tomcat
latest: digest: sha256:8b8b80e28f26fdca3b78081ba4dc5e30d7532659cd80043eed8405b757cc9b8f size: 2837
ubuntu@ip-172-31-93-159:/opt/helloworld$
```

Check if the Docker Image has been pushed to the Docker Hub



We've updated our Terms of Service. Learn more.

Using 0 of 1 private repositories. Get more

diegobettegagl / gl-project3

This repository does not have a description

Last pushed: 3 minutes ago

Docker commands

To push a new tag to this repository,

```
docker push diegobettegagl/gl-project3:tagname
```

Tags

This repository contains 1 tag(s).

latest 3 minutes ago

See all

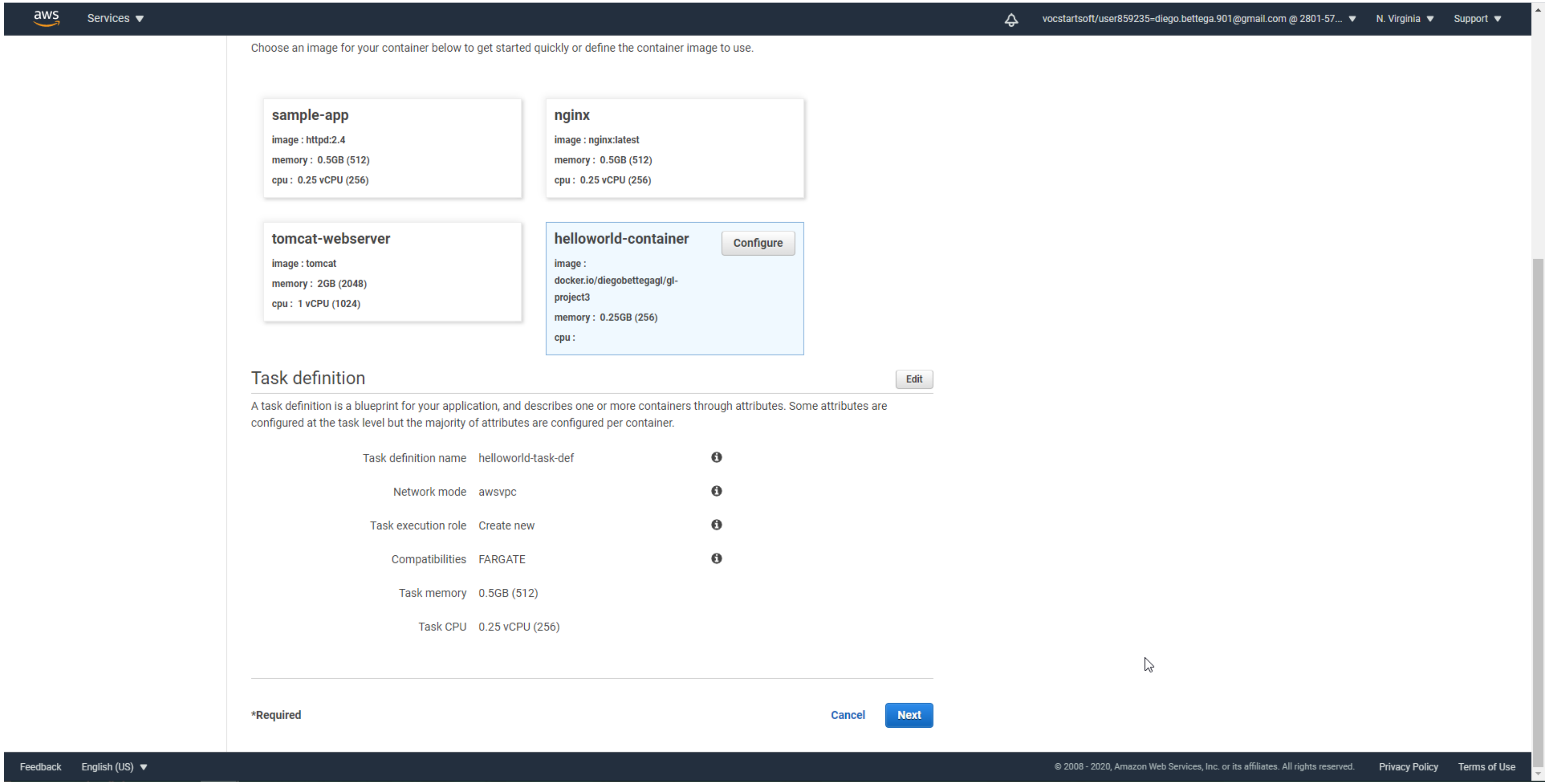
Recent builds

Link a source provider and run a build to see build results here.

Readme

Repository description is empty. Click here to edit.

Configure ECS


aws
Services ▼
vocstartsoft/user859235=diego.bettega.901@gmail.com @ 2801-57... ▼
N. Virginia ▼
Support ▼

Choose an image for your container below to get started quickly or define the container image to use.

sample-app
image : httpd:2.4
memory : 0.5GB (512)
cpu : 0.25 vCPU (256)

nginx
image : nginx:latest
memory : 0.5GB (512)
cpu : 0.25 vCPU (256)

tomcat-webserver
image : tomcat
memory : 2GB (2048)
cpu : 1 vCPU (1024)

helloworld-container
image : docker.io/diegobettegagl/gl-project3
memory : 0.25GB (256)
cpu :

Task definition

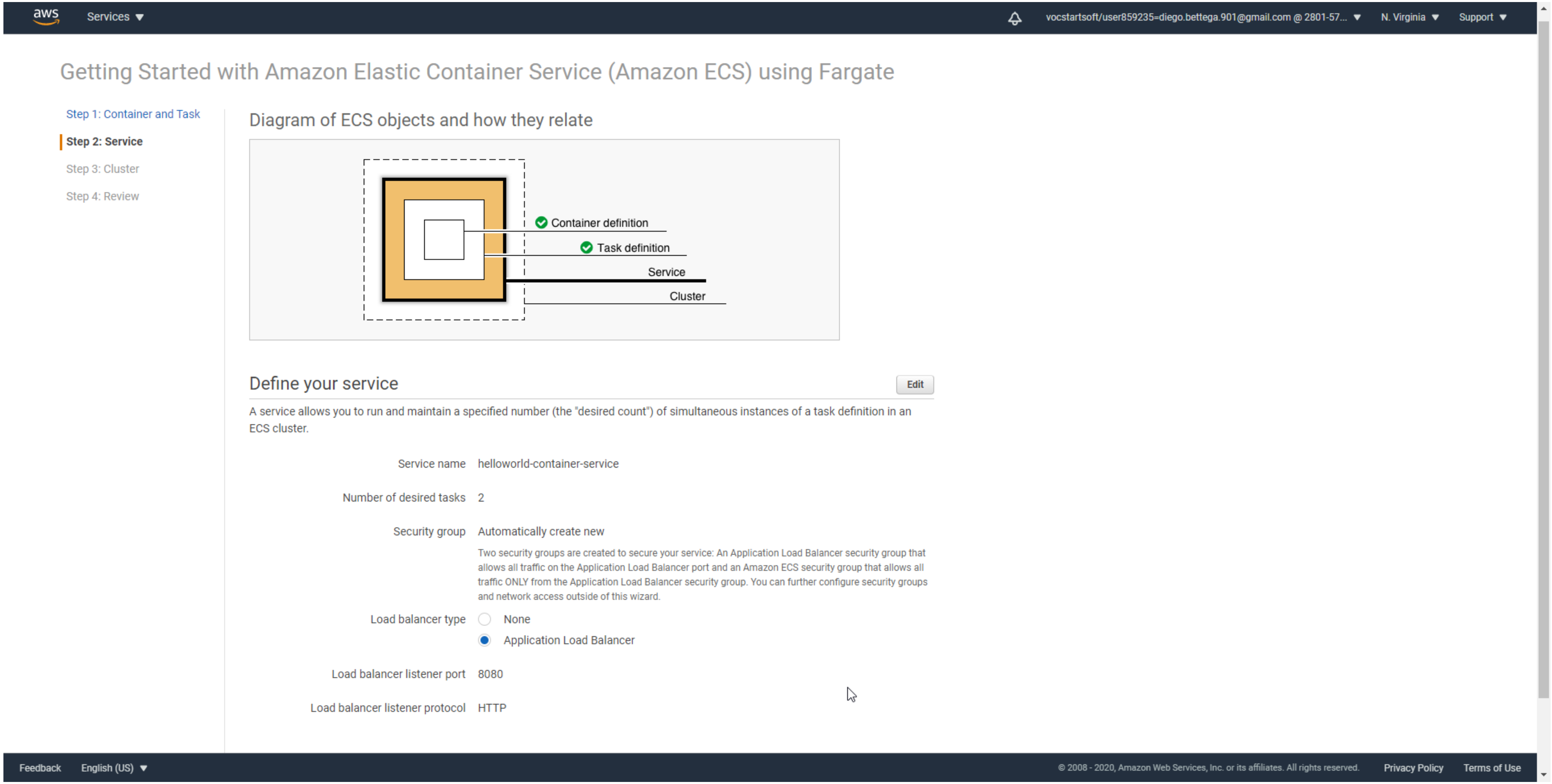
A task definition is a blueprint for your application, and describes one or more containers through attributes. Some attributes are configured at the task level but the majority of attributes are configured per container.

Task definition name	helloworld-task-def
Network mode	awsvpc
Task execution role	Create new
Compatibilities	FARGATE
Task memory	0.5GB (512)
Task CPU	0.25 vCPU (256)

*Required

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Configure ECS



aws
Services ▼
vocstartsoft/user859235=diego.bettega.901@gmail.com @ 2801-57... ▼
N. Virginia ▼
Support ▼

Getting Started with Amazon Elastic Container Service (Amazon ECS) using Fargate

Diagram of ECS objects and how they relate

Define your service

A service allows you to run and maintain a specified number (the "desired count") of simultaneous instances of a task definition in an ECS cluster.

Service name helloworld-container-service

Number of desired tasks 2

Security group Automatically create new

Two security groups are created to secure your service: An Application Load Balancer security group that allows all traffic on the Application Load Balancer port and an Amazon ECS security group that allows all traffic ONLY from the Application Load Balancer security group. You can further configure security groups and network access outside of this wizard.

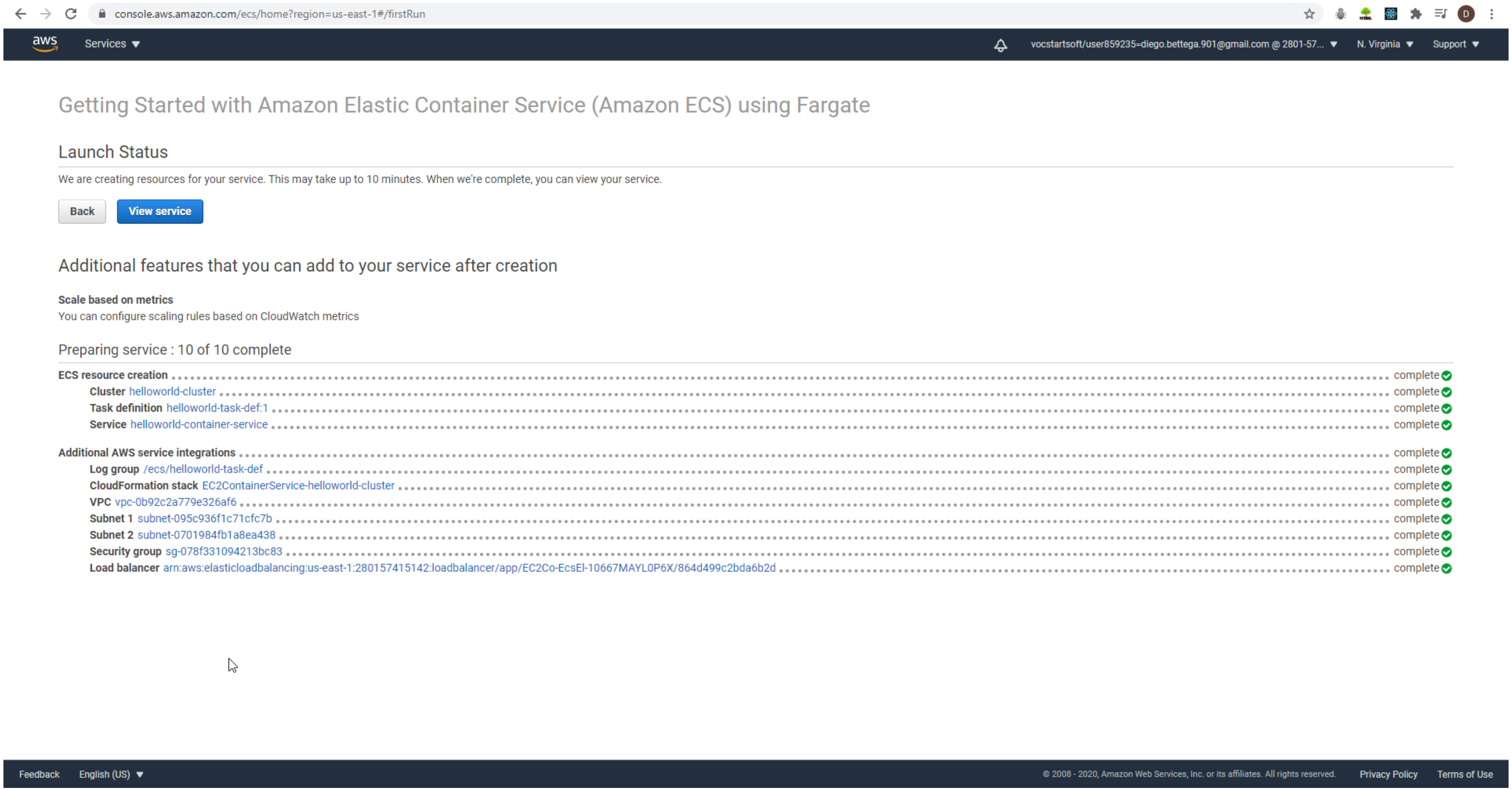
Load balancer type ○ None ● Application Load Balancer

Load balancer listener port 8080

Load balancer listener protocol HTTP

Feedback
English (US) ▼
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Check if the 10 steps are completed


console.aws.amazon.com/ecs/home?region=us-east-1#/firstRun
aws
Services ▼
vocstartsoft/user859235=diego.bettega.901@gmail.com @ 2801-57... ▼
N. Virginia ▼
Support ▼
Back
View service
Feedback
English (US) ▼
Privacy Policy
Terms of Use

Getting Started with Amazon Elastic Container Service (Amazon ECS) using Fargate

Launch Status

We are creating resources for your service. This may take up to 10 minutes. When we're complete, you can view your service.

Additional features that you can add to your service after creation

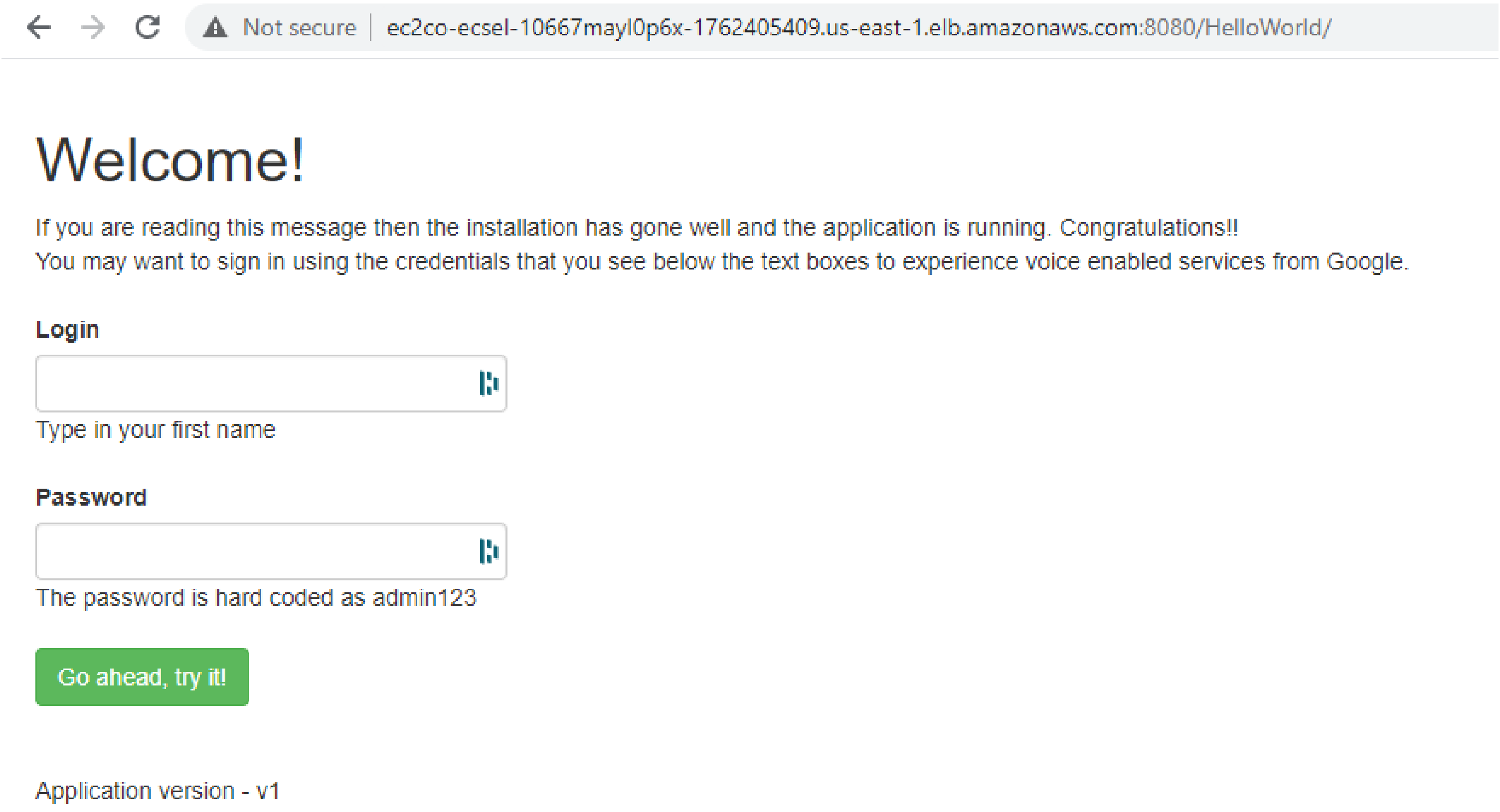
Scale based on metrics
You can configure scaling rules based on CloudWatch metrics

Preparing service : 10 of 10 complete

```
ECS resource creation ........................................ complete ✓
    Cluster helloworld-cluster ............................... complete ✓
    Task definition helloworld-task-def:1 .................... complete ✓
    Service helloworld-container-service ..................... complete ✓

Additional AWS service integrations .......................... complete ✓
    Log group /ecs/helloworld-task-def ....................... complete ✓
    CloudFormation stack EC2ContainerService-helloworld-cluster complete ✓
    VPC vpc-0b92c2a779e326af6 ................................ complete ✓
    Subnet 1 subnet-095c936f1c71cfc7b ........................ complete ✓
    Subnet 2 subnet-0701984fb1a8ea438 ........................ complete ✓
    Security group sg-078f331094213bc83 ...................... complete ✓
    Load balancer arn:aws:elasticloadbalancing:us-east-1:280157415142:loadbalancer/app/EC2Co-EcsEl-10667MAYL0P6X/864d499c2bda6b2d .... complete ✓
```

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Access the deployed application from the LB



Not secure
ec2co-ecsel-10667mayl0p6x-1762405409.us-east-1.elb.amazonaws.com:8080/HelloWorld/

Welcome!

If you are reading this message then the installation has gone well and the application is running. Congratulations!!
You may want to sign in using the credentials that you see below the text boxes to experience voice enabled services from Google.

Login

Type in your first name

Password

The password is hard coded as admin123

Go ahead, try it!

Application version - v1