

RESPONSI PRATIKUM TEKNOLOGI CLOUD



Disusun oleh :

Nama : Indah Suryaningsih

NIM : 225410062

Kelas : IF-2

**PROGRAM STUDI INFORMATIKA
PROGRAM SARJANA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS TEKNOLOGI DIGITAL INDONESIA
YOGYAKARTA
2023**

RESPONSI

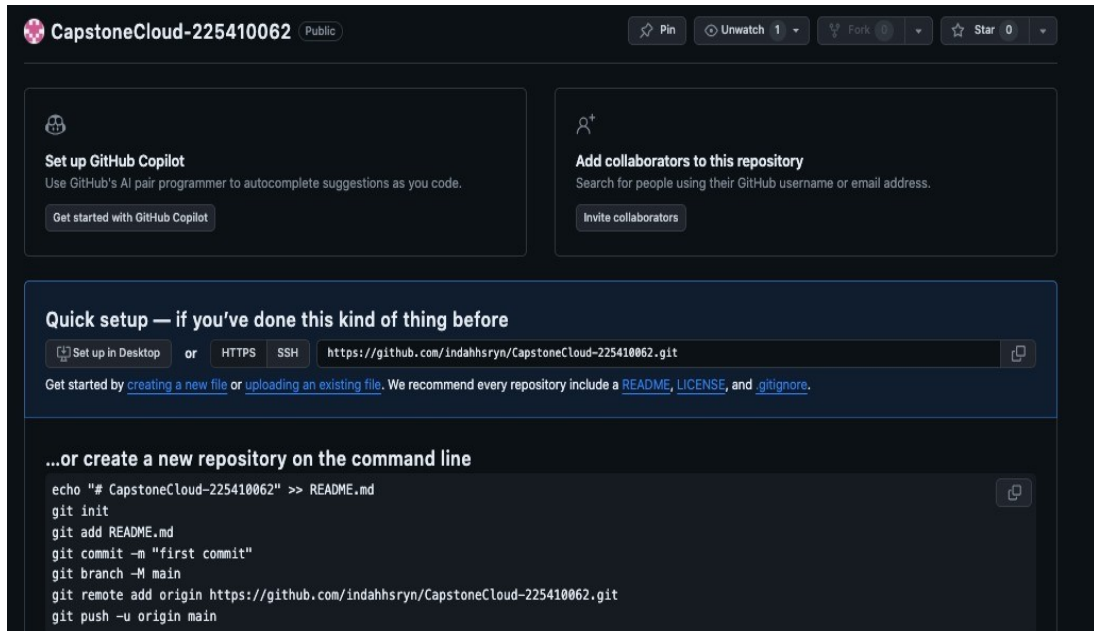
PRAKTIKUM TEKNOLOGI CLOUD

A. TUJUAN

-Mahasiswa dapat membuat proyek capstone

B. PEMBAHASAN

- Membuat Repository Baru di Github:



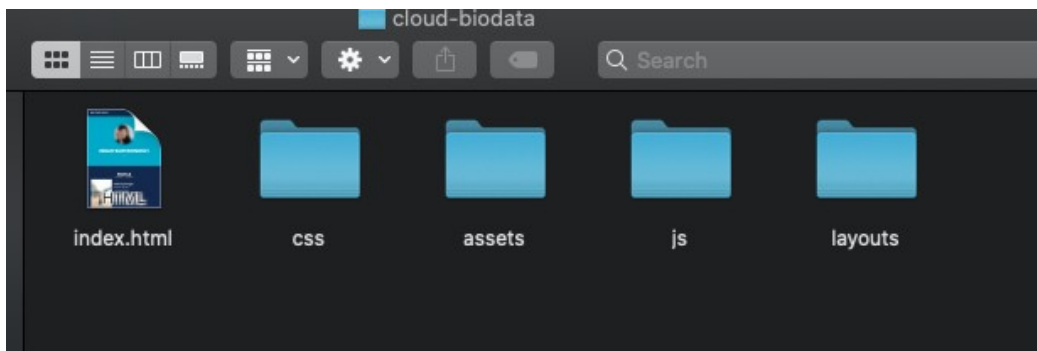
Pembahasan:

Langkah 1 saya membuat repository di github dulu untuk menampung halaman website statis yang berisi informasi , foto , nama , nim.

Link repository github saya bisa diakses di :

<https://github.com/indahhsryn/CapstoneCloud-225410062.git>

- Upload File dari Komputer ke github:



Pembahasan:

halaman profile saya gunakan dengan html, css, dan juga javascript(js) dan assets untuk menampung foto saya.

- Kode Upload ke Github

```
Last login: Sun Dec 24 20:59:54 on ttys000
[macbookpro@192 cloud-biodata % git remote -v
[macbookpro@192 cloud-biodata % git init
Reinitialized existing Git repository in /Applications/XAMPP/xamppfiles/
[macbookpro@192 cloud-biodata % git remote add origin https://github.com/
[macbookpro@192 cloud-biodata % git add .
[macbookpro@192 cloud-biodata % git status
[On branch master
[
No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   assets/aku.jpeg
        new file:   assets/akuprofile-1.png
        new file:   css/custom.css
        new file:   css/styles.css
        new file:   index.html
        new file:   js/scripts.js

[macbookpro@192 cloud-biodata % git commit -m "RESPONSI PRAKLOUD"
[master (root-commit) ae813af] RESPONSI PRAKLOUD
[Committer: MacBook Pro <macbookpro@192.168.100.50>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

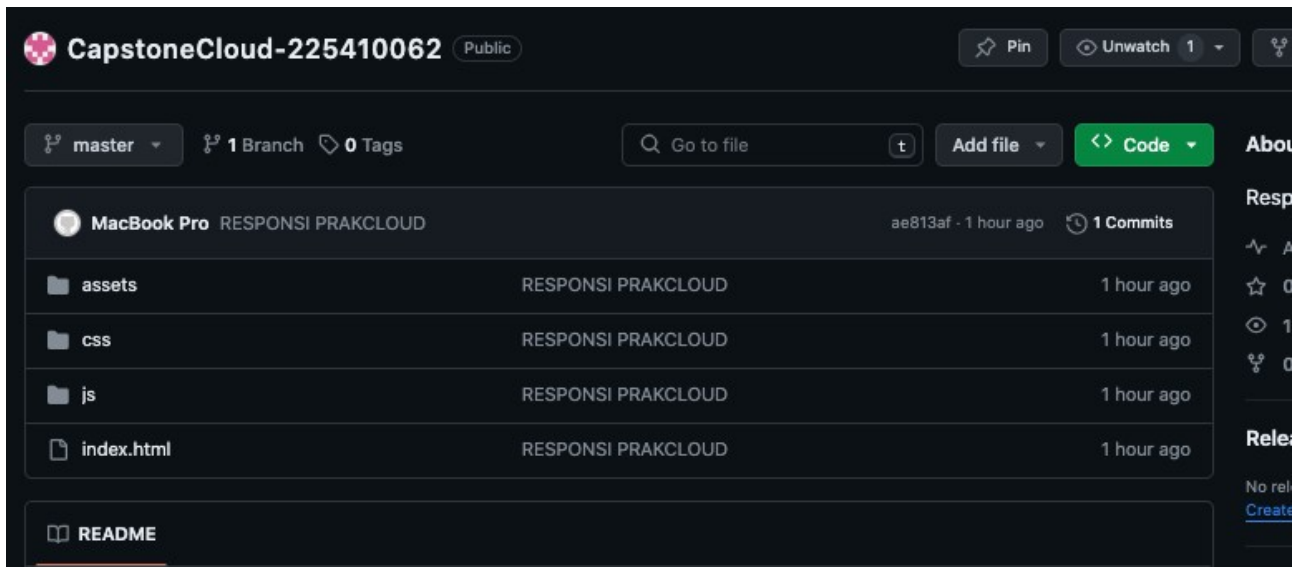
6 files changed, 11795 insertions(+)
create mode 100644 assets/aku.jpeg
create mode 100644 assets/akuprofile-1.png
create mode 100755 css/custom.css
create mode 100755 css/styles.css
create mode 100644 index.html
create mode 100755 js/scripts.js
[macbookpro@192 cloud-biodata % git push origin master
Username for 'https://github.com': indahhsryn
[Password for 'https://indahhsryn@github.com':
remote: Support for password authentication was removed on August 13, 20
remote: Please see https://docs.github.com/en/get-started/getting-starte
f authentication.
fatal: Authentication failed for 'https://github.com/indahhsryn/Capstone
[macbookpro@192 cloud-biodata % git push origin master
Username for 'https://github.com': indahhsryn
Password for 'https://indahhsryn@github.com':
Enumerating objects: 11, done.
[Counting objects: 100% (11/11), done.
[Delta compression using up to 4 threads
[Compressing objects: 100% (10/10), done.
[Writing objects: 100% (11/11), 785.79 KiB | 10.17 MiB/s, done]
```

Pembahasan:

setelah kode dibuat didalam komputer , maka di upload ke halaman github dengan kode diatas , mulai dari dengan git init , git add origin (link github), git add . (untuk menambahkan file) , git status (untuk cek status

file apakah sudah di upload atau belumnya) , git commit -m (untuk menandai nama file di gitnya, git push origin master untuk menambahkan ke dalam github, tunggu proses sampai selesai

- Hasil upload file ke github:



Pembahasan :

Setelah berhasil mengunggah file-file dari komputer ke repositori GitHub, langkah berikutnya adalah memastikan bahwa proyek saya dapat dengan mudah diakses dan dikelola oleh tim .

Github saya bisa diakses di:

<https://github.com/indahhsryn/CapstoneCloud-225410062.git>

- Membuat Repository dockerhub

A screenshot of the 'Create repository' form on Docker Hub. The 'Namespace' dropdown is set to 'indahhh235'. The 'Repository Name' field contains 'capstonecloud-225410062'.

Pembahasan:

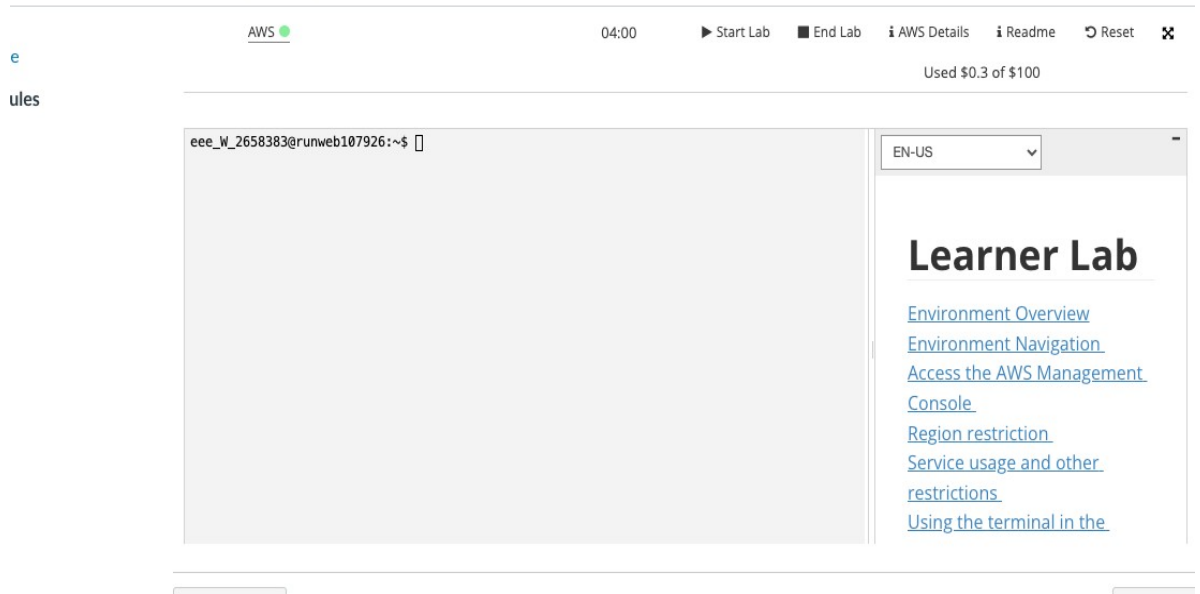
Setelah berhasil membuat repository GitHub, langkah selanjutnya adalah menciptakan repository Docker Hub dengan nama "capstonecloud-225410062" sesuai dengan ketentuan pada instruksi Docker Hub. Docker Hub memungkinkan Anda menyimpan, mengelola, dan mendistribusikan image Docker. Dengan membuat repository di Docker Hub, Anda dapat dengan mudah berbagi image Docker dengan komunitas atau menggunakan image tersebut pada berbagai platform.

link repository docker hub bisa diakses di:

<https://hub.docker.com/r/indahhh235/capstonecloud-225410062>

- Masuk AWS dan Start Lab

ALLv1-39... > Modules > AWS Acad...
> Launch AWS Academy Learner Lab



Pembahasan :

Setelah Repository Githu dan dockerhub selesai dibuat , langkah selanjutnya adalah dengan masuk ke AWS untuk Dockerfile untuk membangun Docker image yang berisi situs web statis.

Langkah langkah untuk masuk ke AWS adalah login dengan email dan password , lalu pilih lauch aws→ec2→connect→start lab seperti gambar diatas

-- Launch dan Instance

Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags [Info](#)

Name

CapstoneCloud-225410062


[Add additional tags](#)

Recents

Quick Start

Amazon
Linux


macOS


Ubuntu


Windows


Red Hat


SI
>



[Browse more AMIs](#)

Including AMIs from
AWS, Marketplace and
the Community

Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type

Free tier eligible ▼

ami-0c7217cdde317cfec (64-bit (x86)) / ami-05d47d29a4c2d19e1 (64-bit (Arm))

Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2023-12-07

Architecture

64-bit (x86) ▼

AMI ID

ami-0c7217cdde317cfec

Verified provider

Subnet [Info](#)

No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)

Enable

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☒ Create security group

☐ Select existing security group

We'll create a new security group called 'launch-wizard-2' with the following rules:

☒ Allow SSH traffic from

Helps you connect to your instance


Anywhere
0.0.0.0/0

☒ Allow HTTPS traffic from the internet

To set up an endpoint, for example when creating a web server

☒ Allow HTTP traffic from the internet

To set up an endpoint, for example when creating a web server

 Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

✕

▼ **Configure storage** [Info](#)

[Advanced](#)

1x GiB Root volume (Not encrypted)

Pembahasan :

Pada langkah ini, beri nama proyek AWS Anda sebagai "CapstoneCloud-225410062". Ketika memilih opsi *quick start*, pilih sistem operasi Ubuntu.

Selanjutnya, pastikan untuk memeriksa opsi *subnet* agar sesuai dengan konfigurasi seperti yang ditunjukkan dalam gambar yang telah diberikan sebelumnya. Proses ini memastikan bahwa proyek Anda dapat berjalan dengan benar di lingkungan AWS dan menggunakan konfigurasi jaringan yang sesuai dengan kebutuhan proyek.

-Clone Souce di github

```
ubuntu@ip-172-31-49-122:~$ git clone https://github.com/indahhsryn/CapstoneCloud-225410062.git
Cloning into 'CapstoneCloud-225410062'...
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 11 (delta 0), reused 11 (delta 0), pack-reused 0
Receiving objects: 100% (11/11), 785.79 KiB | 28.06 MiB/s, done.
ubuntu@ip-172-31-49-122:~$
```

Pembahasan:

Langkah selanjutnya adalah dengan meng clone source code yang ada di github ke AWS , menggunakan kode "git clone <https://github.com/indahhsryn/CapstoneCloud-225410062.git>" Perintah di atas akan menyalin seluruh kode sumber proyek "CapstoneCloud-225410062" dari repositori GitHub ke instance AWS saya .

Setelah proses pengklonan selesai, Anda akan memiliki salinan lokal dari proyek di dalam instance AWS. Langkah selanjutnya adalah mengeksekusi atau mengonfigurasi proyek sesuai dengan persyaratan dan dependensi yang diperlukan. .

- masuk directory dan edit file

```
ubuntu@ip-172-31-49-122:~/CapstoneCloud-225410062$ pico docker file
ubuntu@ip-172-31-49-122:~/CapstoneCloud-225410062$
```

Pembahasan :

Masuk ke Direktori Proyek: Gunakan perintah `cd` untuk masuk ke direktori proyek. proyek disalin ke direktori "CapstoneCloud-225410062", perintahnya akan seperti berikut:

`cd CapstoneCloud-225410062`

Edit File Dockerfile dengan Pico: Gunakan perintah `pico` untuk mengedit file Dockerfile. , perintahnya adalah: `pico Dockerfile`

Simpan dan Keluar dari Editor: Setelah selesai mengedit, simpan perubahan dengan menekan tombol `Ctrl + X`, lalu tekan `Enter`. Kemudian, keluar dari editor dengan menekan tombol `Y`

Dengan langkah-langkah di atas, saya dapat masuk ke direktori proyek di AWS, mengedit file Dockerfile menggunakan Pico

- Edit File

```
GNU nano 6.2
FROM httpd:latest

COPY . /usr/local/apache2/htdocs/

EXPOSE 80

CMD ["httpd-foreground"]

^G Help      ^C Write Out  ^W Where Is
^X Exit      ^R Read File  ^\ Replace
```

Pembahasan :

Edit File Dockerfile dengan Pico: Gunakan perintah `pico` untuk mengedit file Dockerfile. , perintahnya adalah: `pico Dockerfile` , dan isi tulisan seperti diatas .

Simpan dan Keluar dari Editor: Setelah selesai mengedit, simpan perubahan dengan menekan tombol `Ctrl + X`, lalu tekan `Enter`. Kemudian, keluar dari editor dengan menekan tombol `Y`

- build ke docker hub

```
ubuntu@ip-172-31-49-122:~/CapstoneCloud-225410062$ sudo docker build -t indahhh235/capstonecloud-225410062 .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
             Install the buildx component to build images with BuildKit:
             https://docs.docker.com/go/buildx/

Sending build context to Docker daemon  1.879MB
Step 1/4 : FROM httpd:latest
--> 6fd77d7e5eb7
Step 2/4 : COPY . /usr/local/apache2/htdocs/
--> Using cache
--> f79dell186a30
Step 3/4 : EXPOSE 80
--> Using cache
--> 32037d199a24
Step 4/4 : CMD ["httpd-foreground"]
--> Using cache
--> c355c4e5b482
Successfully built c355c4e5b482
Successfully tagged indahhh235/capstonecloud-225410062:latest
ubuntu@ip-172-31-49-122:~/CapstoneCloud-225410062$
```

Pembahasan:

Docker akan melakukan proses build untuk image dengan tag `indahhh235/capstonecloud-225410062` dari konteks saat ini, yang diasumsikan berada dalam direktori proyek Dockerfile (dilambangkan dengan titik di akhir perintah). Hasil build seperti gambar diatas

- Login ke docker hub:

```
ubuntu@ip-172-31-49-122:~/CapstoneCloud-225410062$ sudo docker login -u indahhh235
Password:
WARNING! Your password will be stored unencrypted in /root/.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-172-31-49-122:~/CapstoneCloud-225410062$
```

Pembahasan:

Perintah ini meminta Anda untuk memasukkan nama pengguna Docker Hub (<username>), dan setelah itu, sistem akan meminta kata sandi untuk akun Docker Hub yang bersangkutan. Setelah berhasil login, saya dapat melakukan berbagai operasi, seperti membangun dan mengunggah image Docker ke Docker Hub menggunakan akun yang telah diotorisasi.

- push ke directory docker hub

```
ubuntu@ip-172-31-49-122:~/CapstoneCloud-225410062$ sudo docker push indahhh235/capstonecloud-225410062
Using default tag: latest
The push refers to repository [docker.io/indahhh235/capstonecloud-225410062]
2b41fa2c820: Pushed
9cd4026118f7: Mounted from library/httpd
9c10599774e2: Mounted from library/httpd
87939ee964f4: Mounted from library/httpd
5f70bf18a086: Mounted from library/httpd
1b65777f1238: Mounted from library/httpd
7292cf786aa8: Mounted from library/httpd
latest: digest: sha256:06a54bca033c62b3ced2da6d1bda86f4a4d5c0f4c41812d1ca8794df9d075bae size: 1783
ubuntu@ip-172-31-49-122:~/CapstoneCloud-225410062$
```

Pembahasan:

`sudo docker push indahhh235/capstonecloud-225410062`, adalah langkah selanjutnya setelah membangun image Docker. Perintah tersebut mengunggah image yang sudah dibuat ke Docker Hub di bawah repository dengan nama "indahhh235/capstonecloud-225410062".

Setelah login berhasil, saya dapat menjalankan perintah `sudo docker push indahhh235/capstonecloud-225410062` untuk mengunggah image ke repository Docker Hub saya. Ini akan membuat image Anda dapat diakses oleh orang lain dan digunakan di berbagai lingkungan Docker.

- running web yang sudah di push

```
ubuntu@ip-172-31-49-122:~/CapstoneCloud-225410062$ sudo docker run -d -p 80:80 indahhh235/capstonecloud-225410062
576552cbde95c35b01554fabe7b97f7a4cfc1fd41ed26b6666400ae0636b9c8
ubuntu@ip-172-31-49-122:~/CapstoneCloud-225410062$
```

- Link website

i-0d4aa3c5acc7ed3eb (CapstoneCloud-225410062)

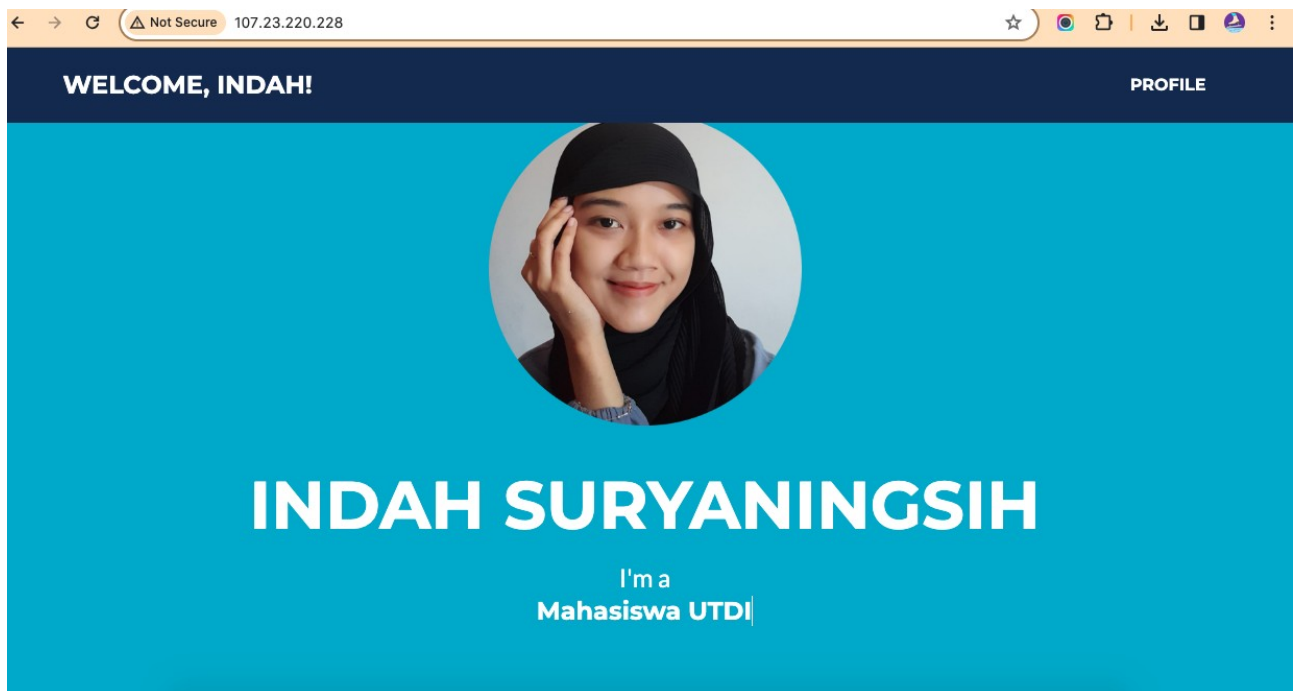
PublicIPs: 107.23.220.228 PrivateIPs: 172.31.49.122

Pembahasan:

Perintah `sudo docker run -d -p 80:80 indahhh235/capstonecloud-225410062` digunakan untuk menjalankan sebuah container Docker berdasarkan image yang sudah di-push sebelumnya ke repository Docker Hub dengan nama "indahhh235/capstonecloud-225410062".

Setelah menjalankan perintah ini, container Docker akan berjalan, dan aplikasi web yang terkandung di dalamnya dapat diakses melalui browser pada alamat 107.23.220.228

- Hasil



WELCOME, INDAH!

PROFILE

PROFILE



Indah Suryaningsih

Small Action , Big Impact

Nama: Indah Suryaningsih

NIM: 225410062

✉ indahsuryaningsih538@gmail.com

Pembahasan:

Hasil web diatas adalah dari link 107.23.220.228 , yang merupakan source code yang diambil dari github dan ditampilkan lewat docker hub , melaui langkah langkah diatas

C. KESIMPULAN

Dengan menggunakan Dockerfile di atas, saya dapat membangun image Docker yang berisi situs web statis menggunakan server web Nginx. Pada dasarnya, Dockerfile ini mendefinisikan langkah-langkah untuk membangun image, termasuk menggunakan image dasar, menyalin file konfigurasi dan situs web statis, mengekspos port 80, dan menentukan perintah yang akan dijalankan saat container berjalan. Dengan cara ini, saya dapat dengan mudah membuat image yang dapat digunakan untuk menyajikan situs web statis menggunakan Docker.

D. LAMPIRAN

—

