

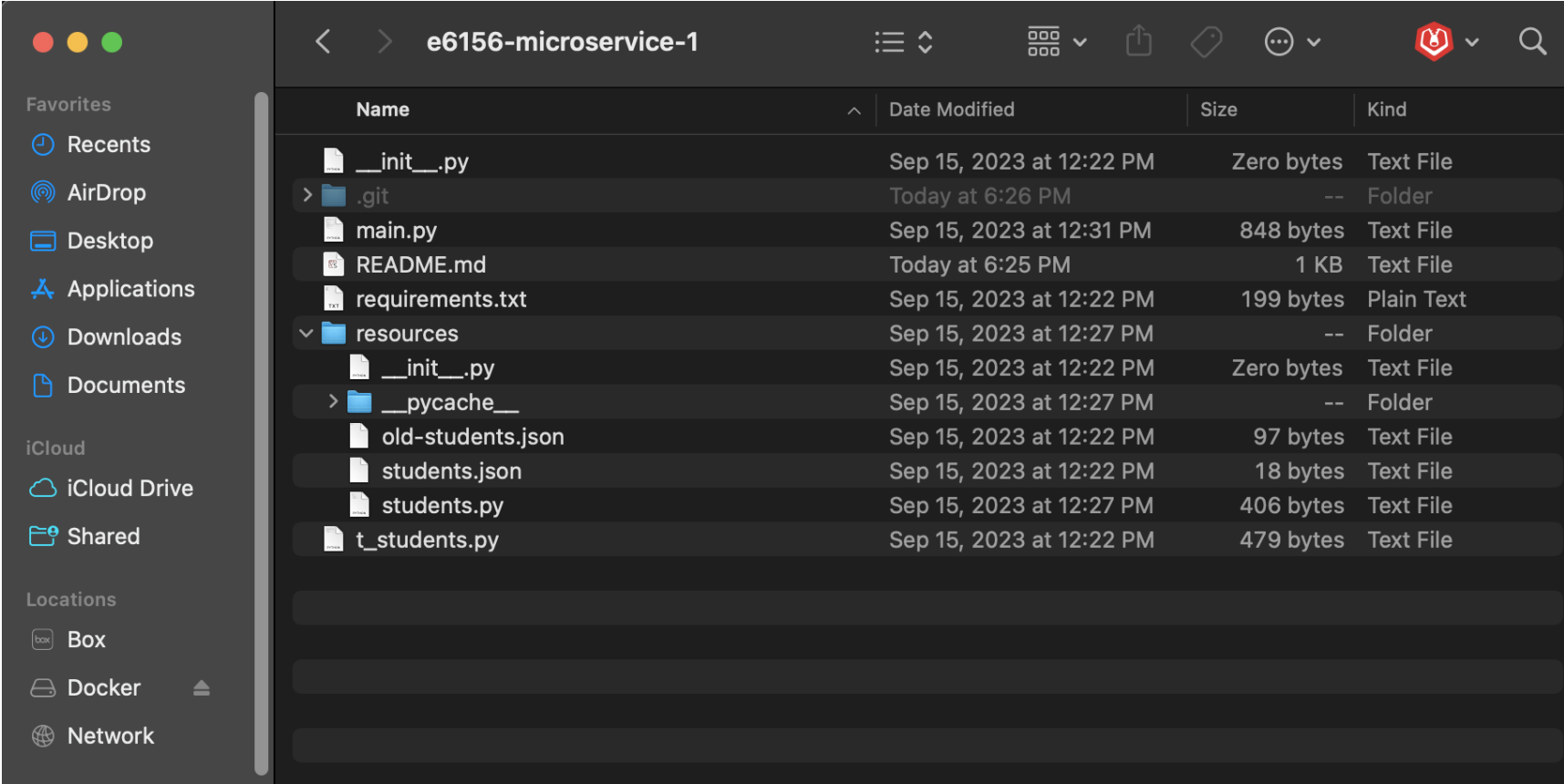
README.md

E6156 - Topics in SW Engineering: Cloud Computing

First Example Microservice

Author: Amrit Ramesh

Directory



File Explorer

The screenshot displays a code editor with a Python FastAPI application. The code defines a FastAPI app with two endpoints: a root endpoint returning "Hello World" and a hello endpoint returning a personalized message. A resource module is also defined with a get_students endpoint. The application is run using uvicorn.

```
1 from fastapi import FastAPI, Response
2
3 # I like to launch directly and not use the standard FastAPI startup
4 import uvicorn
5
6 from resources.students import StudentsResource
7
8 app = FastAPI()
9
10 students_resource = StudentsResource()
11
12 @app.get("/")
13 async def root():
14     return {"message": "Hello World"}
15
16
17 @app.get("/hello/{name}")
18 async def say_hello(name: str):
19     return {"message": f"Awesome cloud developer ar4532 says hello {name}"}
20
21
22 @app.get("/hello_text/{name}")
23 async def say_hello_text(name: str):
24     the_message = f"Awesome cloud developer ar4532 says Hello {name}"
25     rsp = Response(content=the_message, media_type="text/plain")
26     return rsp
27
28
29 @app.get("/students")
30 async def get_students():
31     result = students_resource.get_students()
32     return result
33
34
35 if __name__ == "__main__":
36     uvicorn.run(app, host="0.0.0.0", port=8012)
37
38
```

The file directory on the right shows the project structure:

Name	Date Modified
__init__.py	9/15/23 12:22 PM
main.py	9/15/23 12:31 PM
pictures	9/19/23 6:29 PM
README.md	9/19/23 6:30 PM
requirements.txt	9/15/23 12:22 PM
resources	9/15/23 12:27 PM
__init__.py	9/15/23 12:22 PM
__pycache__	9/15/23 12:27 PM
old-students.json	9/15/23 12:22 PM
students.json	9/15/23 12:22 PM
students.py	9/15/23 12:27 PM
t_students.py	9/15/23 12:22 PM

The console output shows the Python environment and IPython prompt:

```
Python 3.9.13 | packaged by conda-forge | (main, May 27 2022, 17:00:33)
Type "copyright", "credits" or "license" for more information.

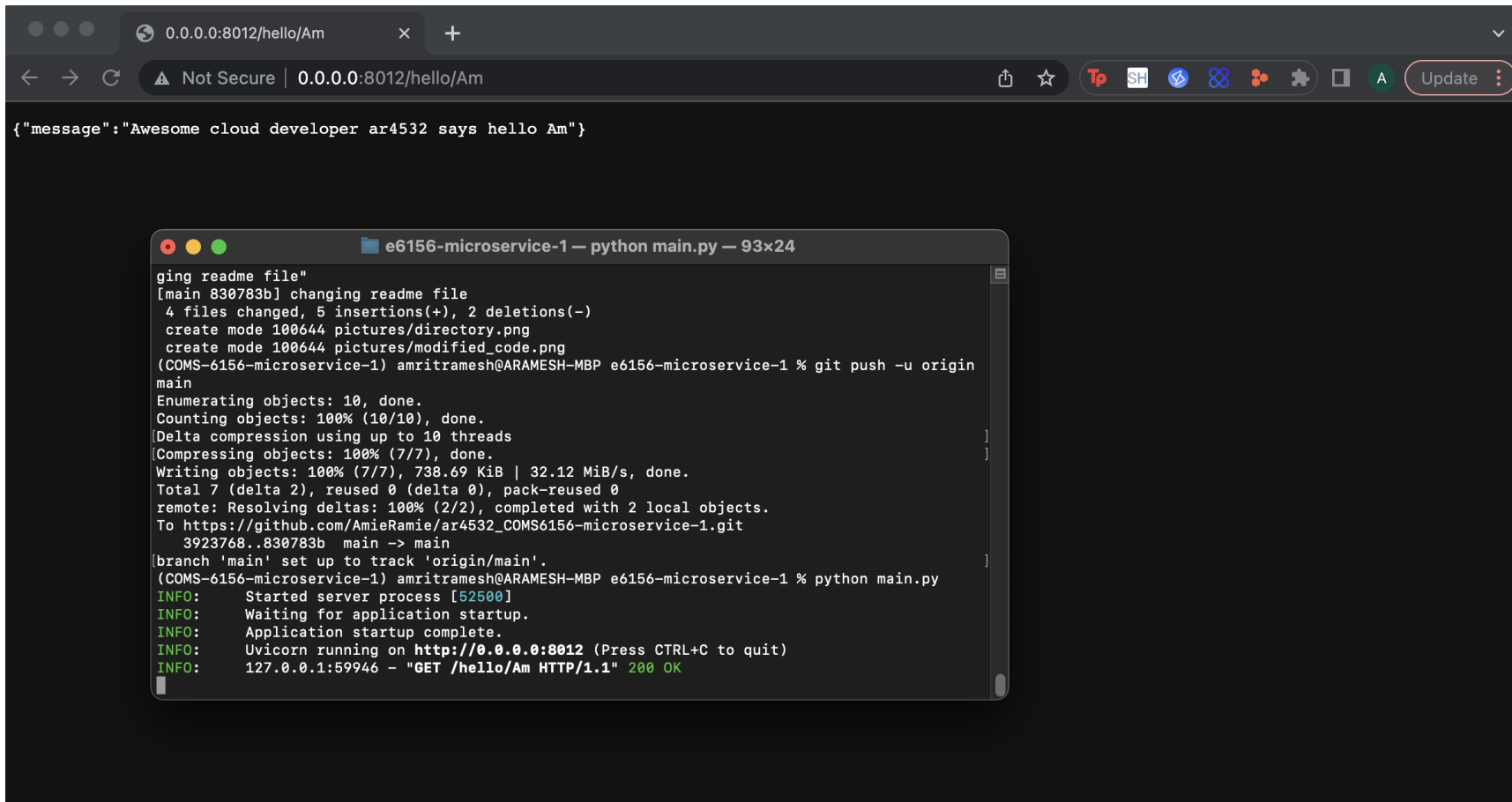
IPython 7.34.0 -- An enhanced Interactive Python.

In [1]:
```

Modified Code

Here is my file directory

Starter Project Executed



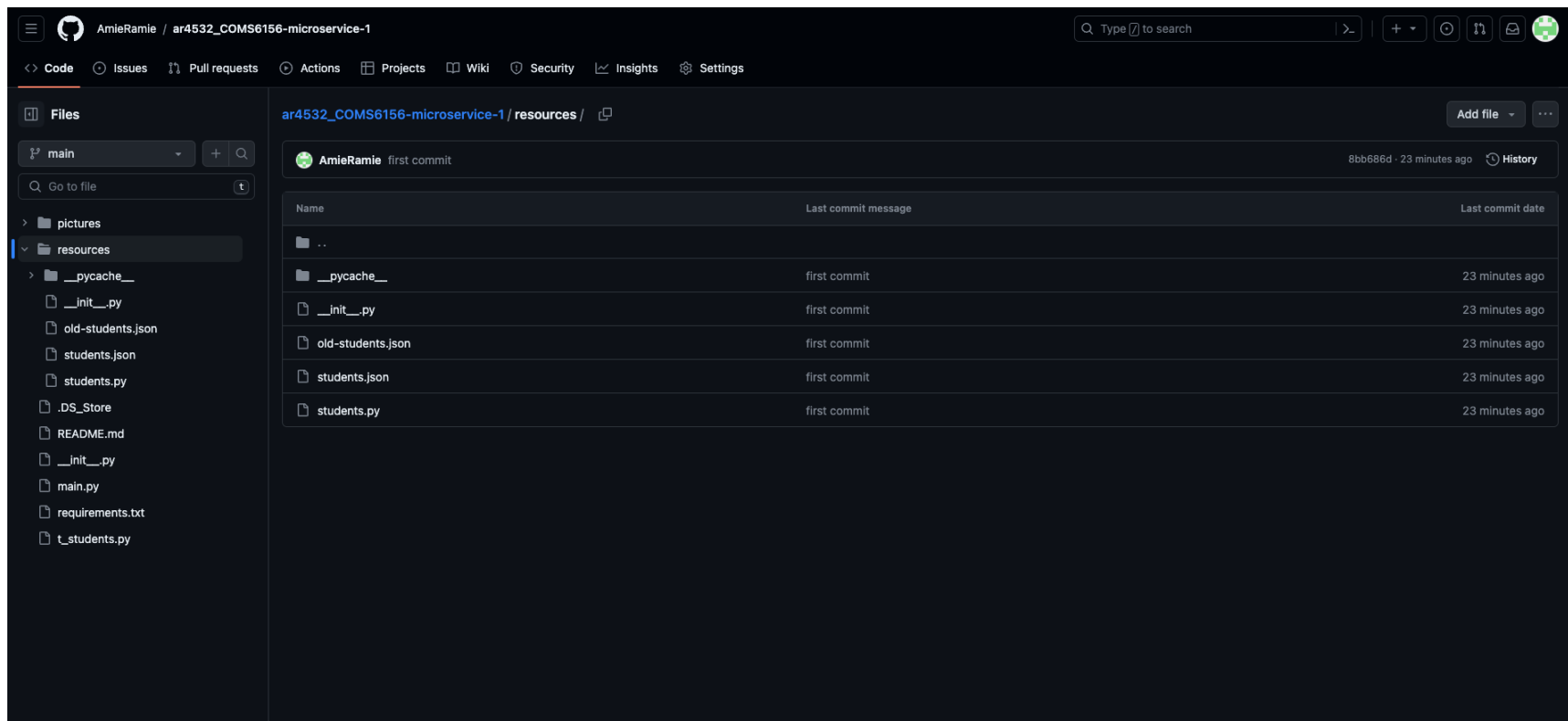
The screenshot shows a web browser window with the address bar displaying "0.0.0.0:8012/hello/Am". The browser's developer tools or console shows a JSON response: `{"message": "Awesome cloud developer ar4532 says hello Am"}`. In the foreground, a terminal window titled "e6156-microservice-1 — python main.py — 93x24" is open. The terminal output shows the following sequence of events:

```
ging readme file"
[main 830783b] changing readme file
4 files changed, 5 insertions(+), 2 deletions(-)
create mode 100644 pictures/directory.png
create mode 100644 pictures/modified_code.png
(COMS-6156-microservice-1) amritramesh@ARAMESH-MBP e6156-microservice-1 % git push -u origin main
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 10 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 738.69 KiB | 32.12 MiB/s, done.
Total 7 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/AmieRamie/ar4532_COMS6156-microservice-1.git
3923768..830783b main -> main
branch 'main' set up to track 'origin/main'.
(COMS-6156-microservice-1) amritramesh@ARAMESH-MBP e6156-microservice-1 % python main.py
INFO: Started server process [52500]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:8012 (Press CTRL+C to quit)
INFO: 127.0.0.1:59946 - "GET /hello/Am HTTP/1.1" 200 OK
```

Executing Script

Here you can see that we have started the flask server in main.py and sent a request to the /hello endpoint

Publish to personal GitHub repository



Personal Repo with Project

I cloned the starter repo Professor Ferguson made, deleted the .git file, created a new repo, and pushed all updates to that repo. This is that result

Deploying Project on AWS

EC2 > Instances > i-0530b81038b4b8180

Instance summary for i-0530b81038b4b8180 (COMS6156 Microservice 1) Info

Updated less than a minute ago

Refresh

Connect

Instance state ▼

Actions ▼

Instance ID

I-0530b81038b4b8180 (COMS6156 Microservice 1)

IPv6 address

—

Hostname type

IP name: ip-172-31-12-251.us-east-2.compute.internal

Answer private resource DNS name

IPv4 (A)

Auto-assigned IP address

3.145.101.164 [Public IP]

IAM Role

—

IMDSv2

Required

Public IPv4 address

3.145.101.164 [open address](#)

Instance state

Running

Private IP DNS name (IPv4 only)

ip-172-31-12-251.us-east-2.compute.internal

Instance type

t2.micro

VPC ID

vpc-05b3d5b9a57a30045

Subnet ID

subnet-0fcfd4e316b10480

Private IPv4 addresses

172.31.12.251

Public IPv4 DNS

ec2-3-145-101-164.us-east-2.compute.amazonaws.com [open address](#)

Elastic IP addresses

—

AWS Compute Optimizer finding

[Opt-in to AWS Compute Optimizer for recommendations.](#) [Learn more](#)

Auto Scaling Group name

—

Details | Security | Networking | Storage | Status checks | Monitoring | Tags

▼ Instance details Info

Platform

Amazon Linux (Inferred)

Platform details

Linux/UNIX

Stop protection

Disabled

Instance auto-recovery

Default

AMI Launch index

0

AMI ID

ami-00a9282ce3b5ddfb1

AMI name

al2023-ami-2023.1.20230906.1-kernel-6.1-x86_64

Launch time

Tue Sep 19 2023 18:22:18 GMT-0400 (Eastern Daylight Time) (about 1 hour)

Lifecycle

normal

Key pair assigned at launch

microservice-1

Monitoring

disabled

Termination protection

Disabled

AMI location

amazon/al2023-ami-2023.1.20230906.1-kernel-6.1-x86_64

Stop-hibernate behavior

Disabled

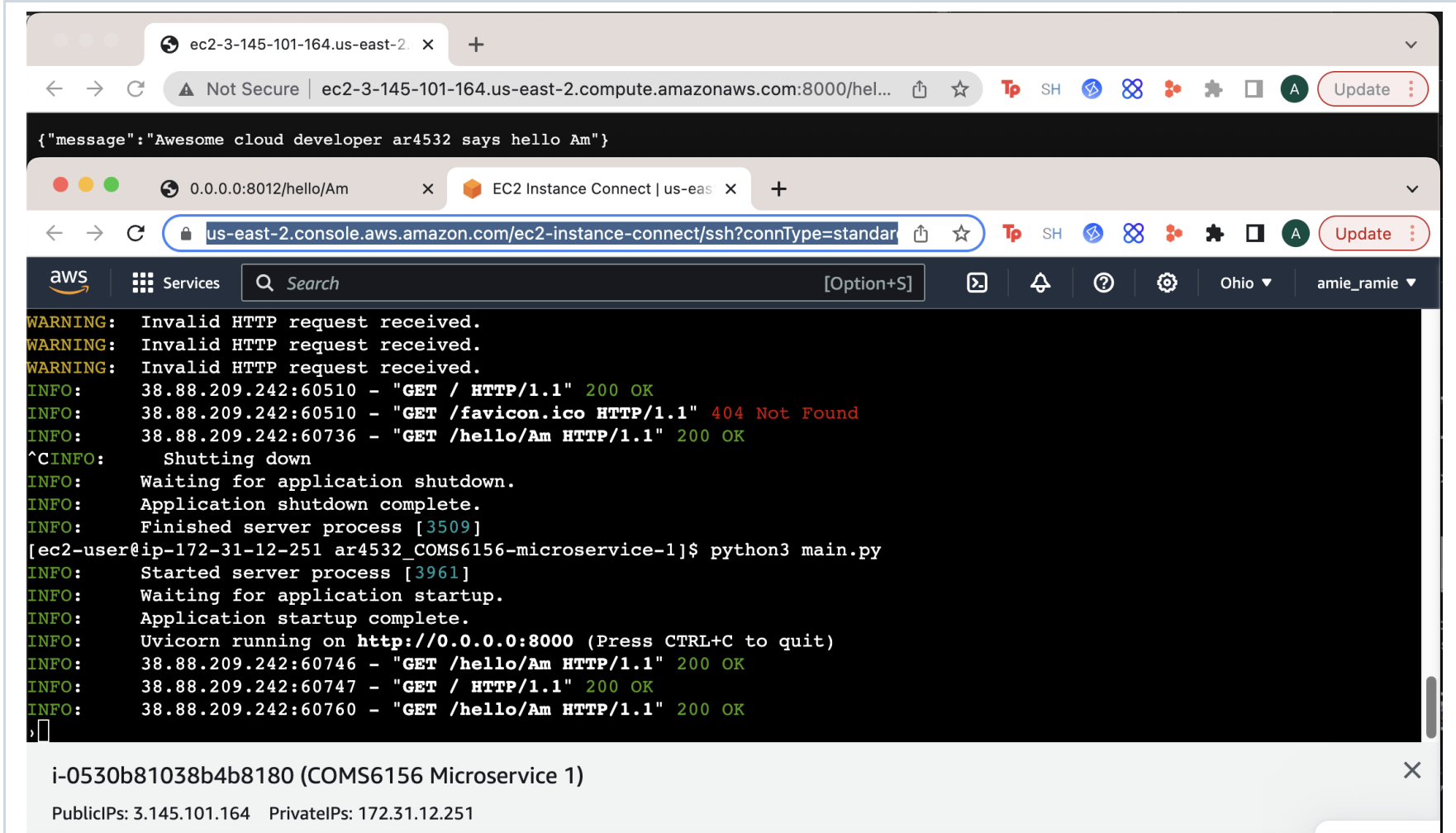
State transition reason

—

Information about EC2 VM

localhost:6419

6/7



Microservice Running on AWS

Due to some issues that I wasn't able to solve with FastAPI I could only access the public url AWS gave me using http vs https. It does not work with https.

Here is my repo: https://github.com/AmieRamie/ar4532_COMS6156-microservice-1/tree/main And though I may have stopped the instance by the time you start grading, here is the link to the app: <http://ec2-3-145-101-164.us-east-2.compute.amazonaws.com:8000/hello/Am>