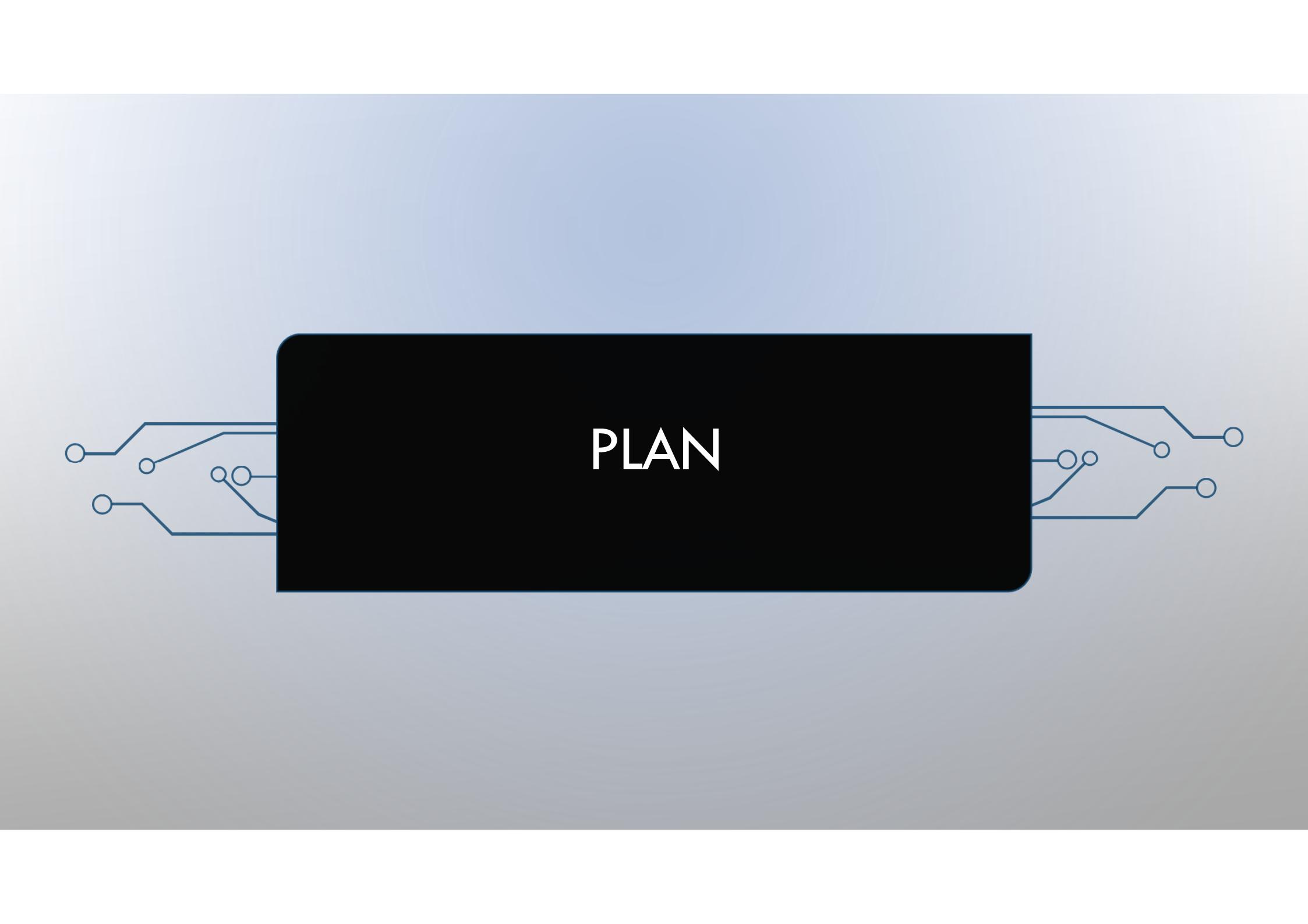




# B8IT122 CLOUD INFRASTRUCTURE & VIRTUALISATION: SERVER VIRTUALISATION

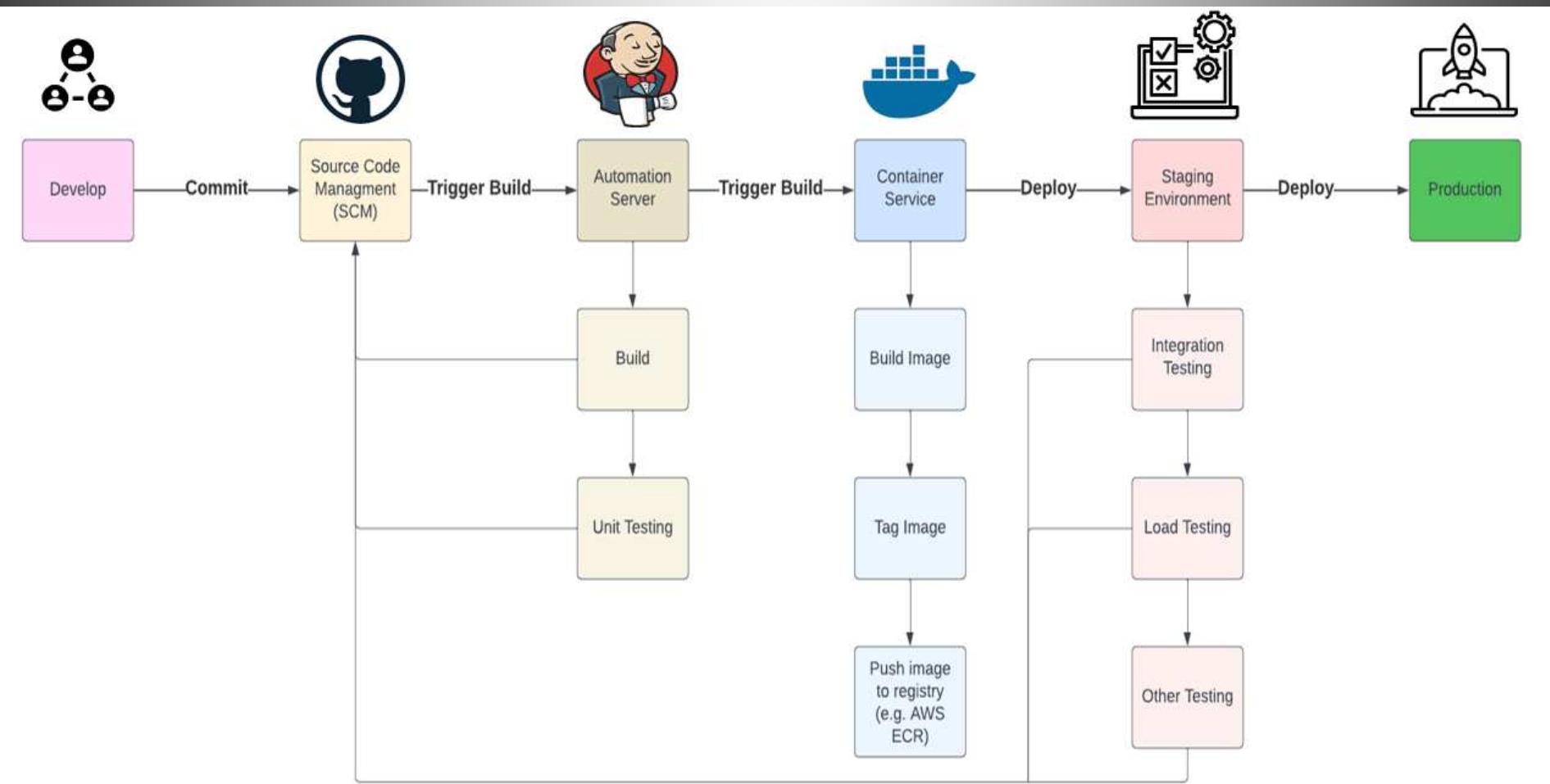
AOIFE O'CONNOR - 10609093

FIONA RYAN - 10616413



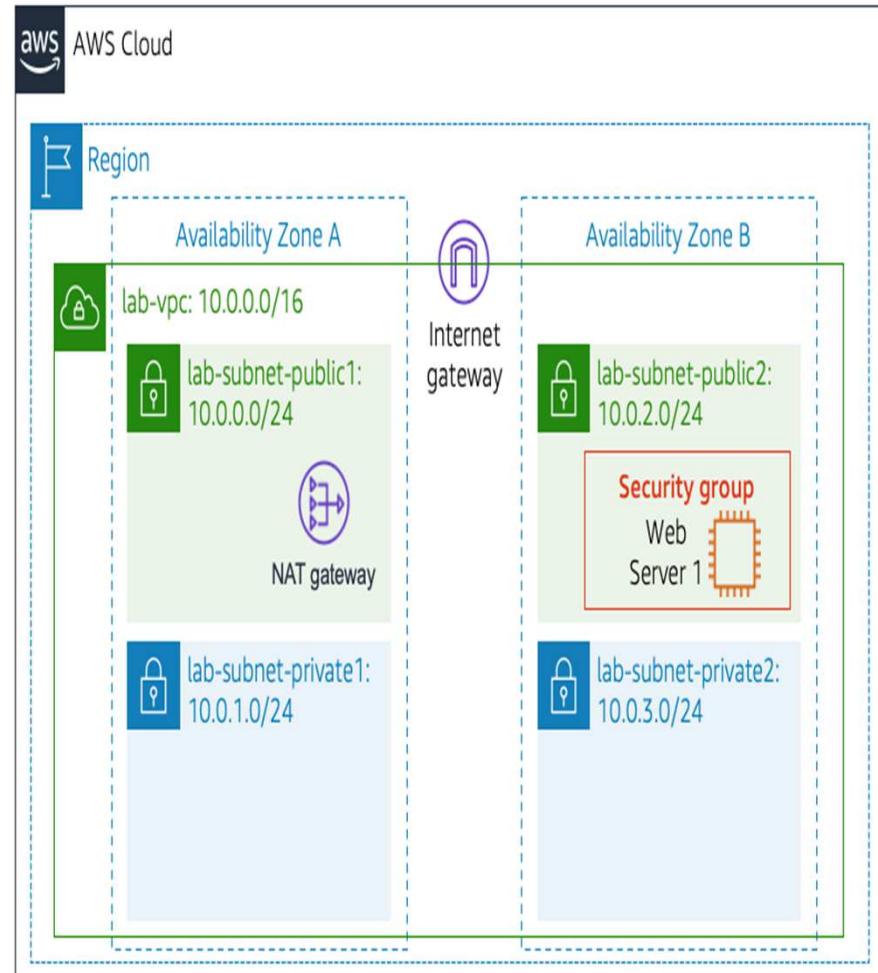
PLAN

## Continuous Integration and Continuous Delivery (CI/CD Pipeline)



# VPC INFRASTRUCTURE

- EC2 instance
- Security Groups
- Availability zones
- Subnets and routing tables
- Amazon Linux 2023 AMI



Public Route Table

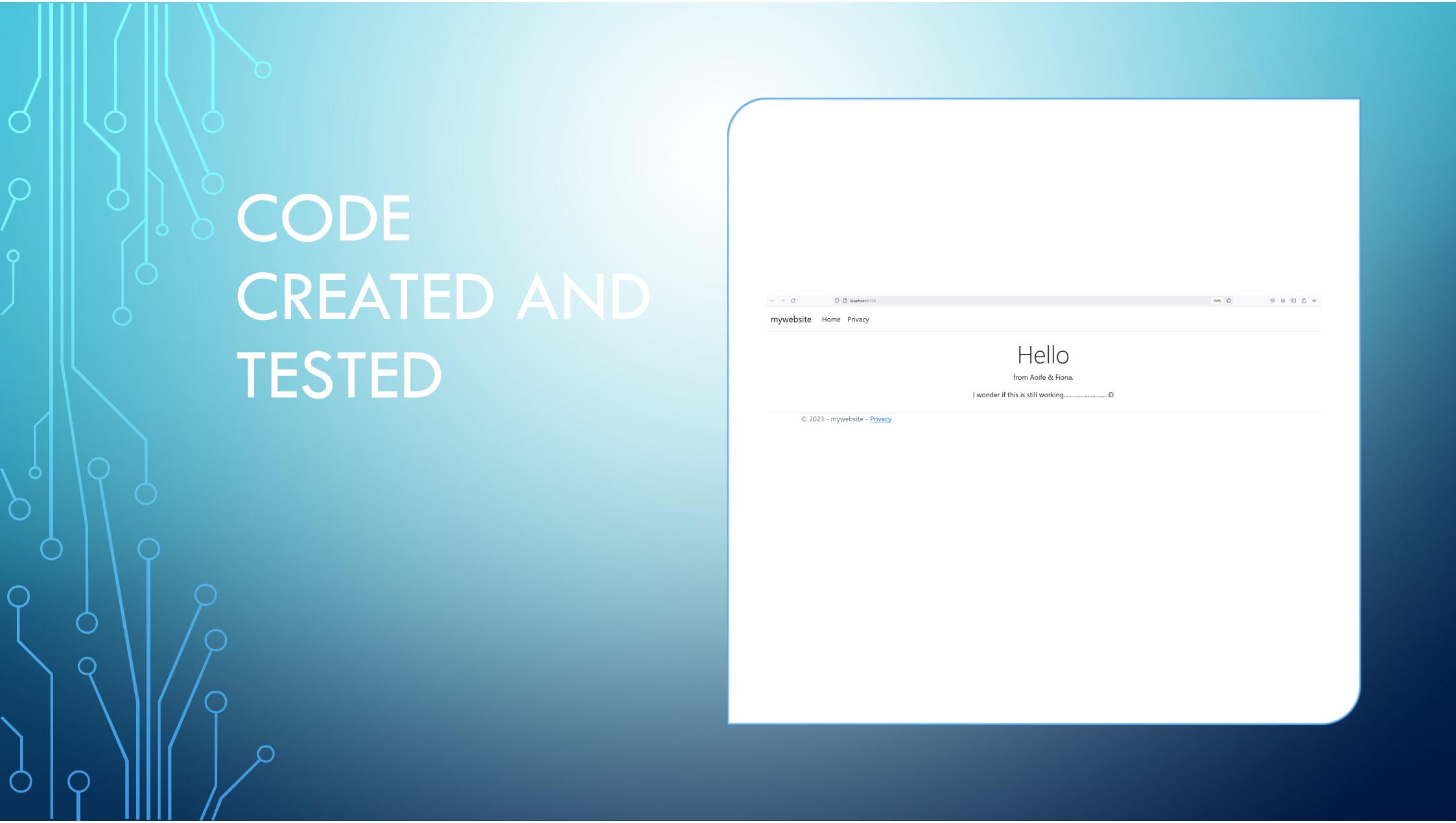
Destination	Target
10.0.0.0/16	local
0.0.0.0/0	Internet gateway

Private Route Tables

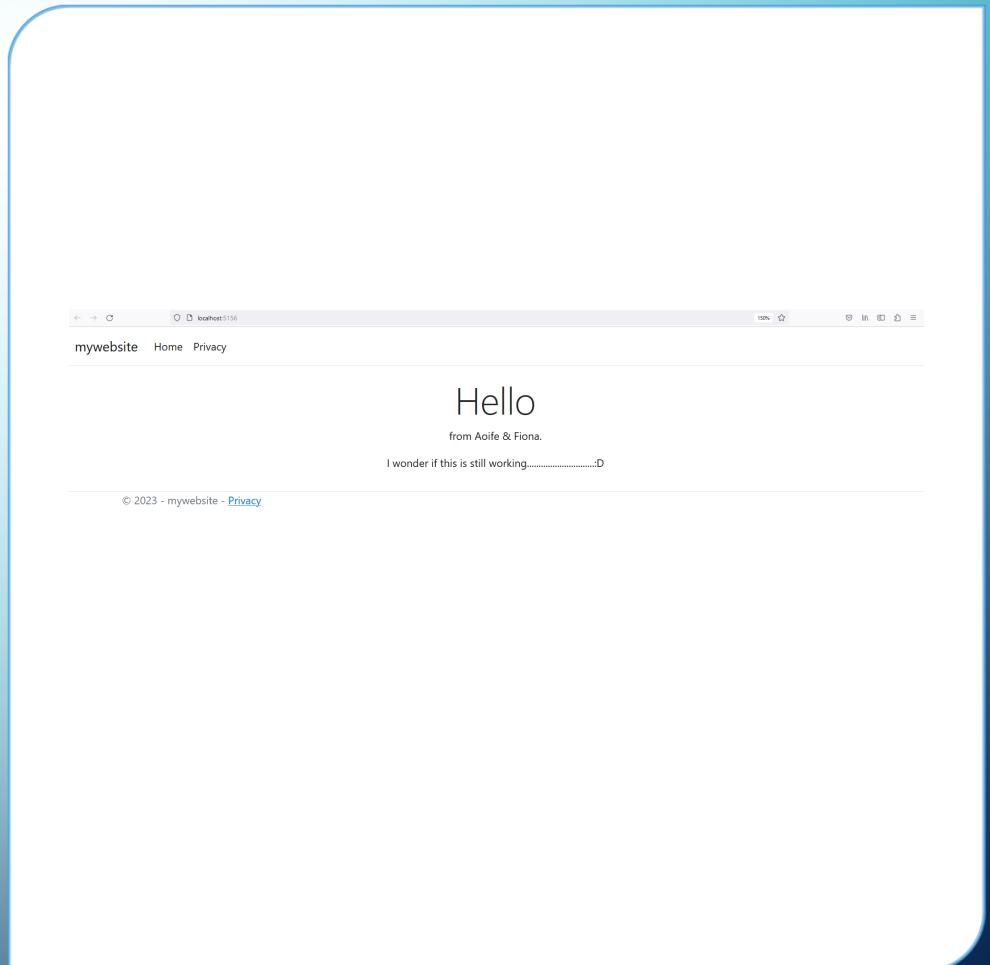
Destination	Target
10.0.0.0/16	local
0.0.0.0/0	NAT gateway



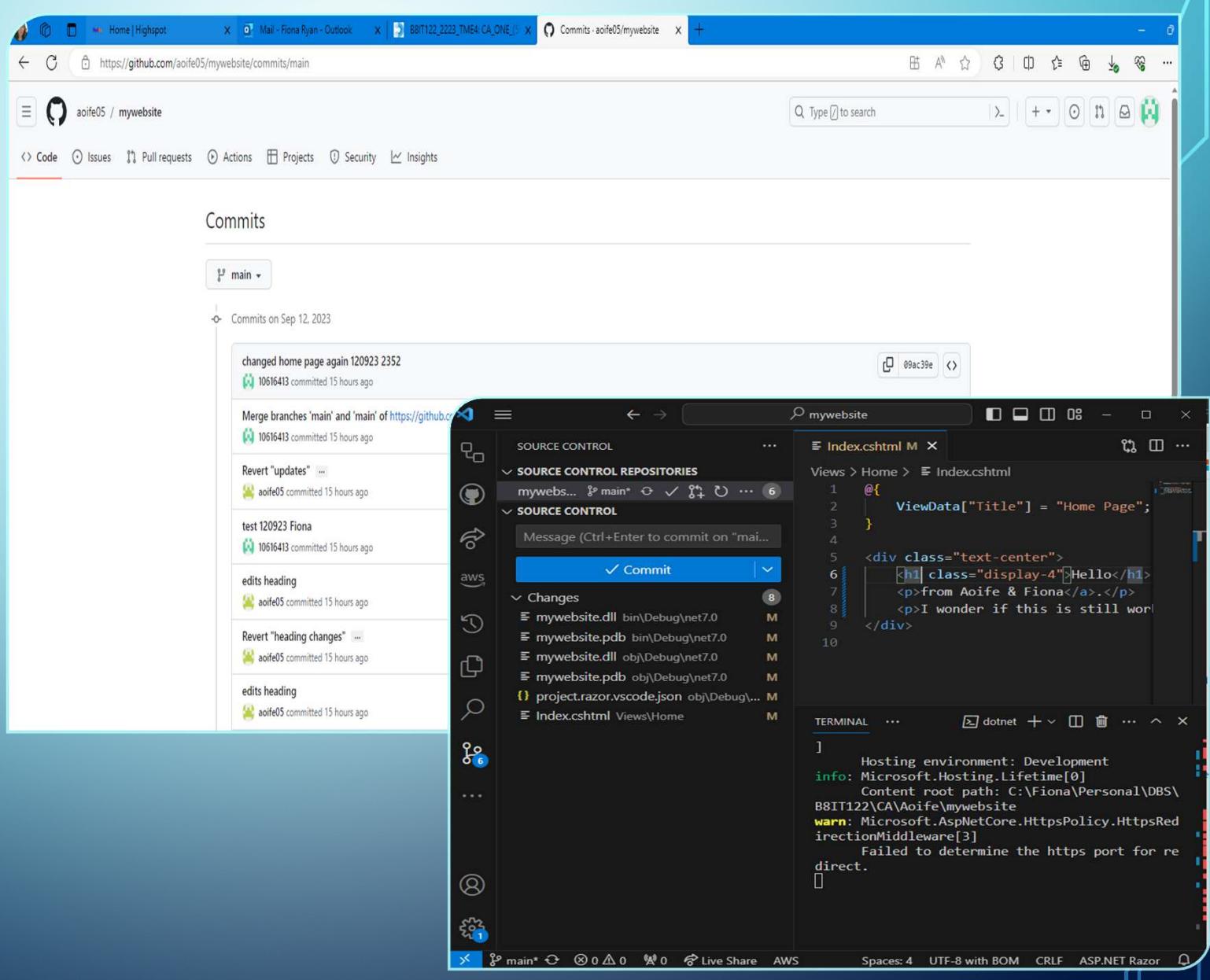
WHAT'S DONE

A decorative background graphic featuring a dense network of blue and white lines forming a circuit board pattern, with various nodes represented by small circles.

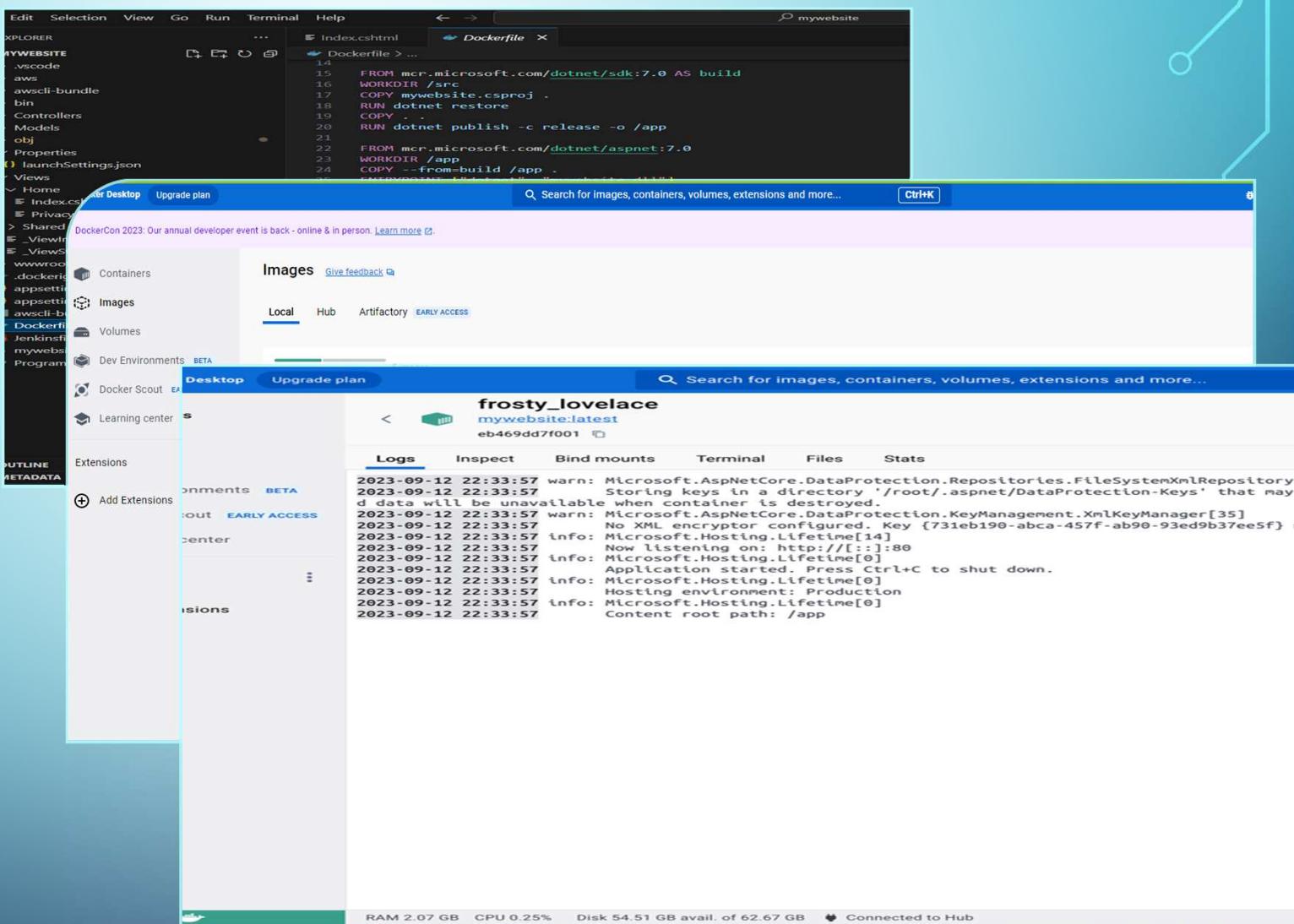
# CODE CREATED AND TESTED



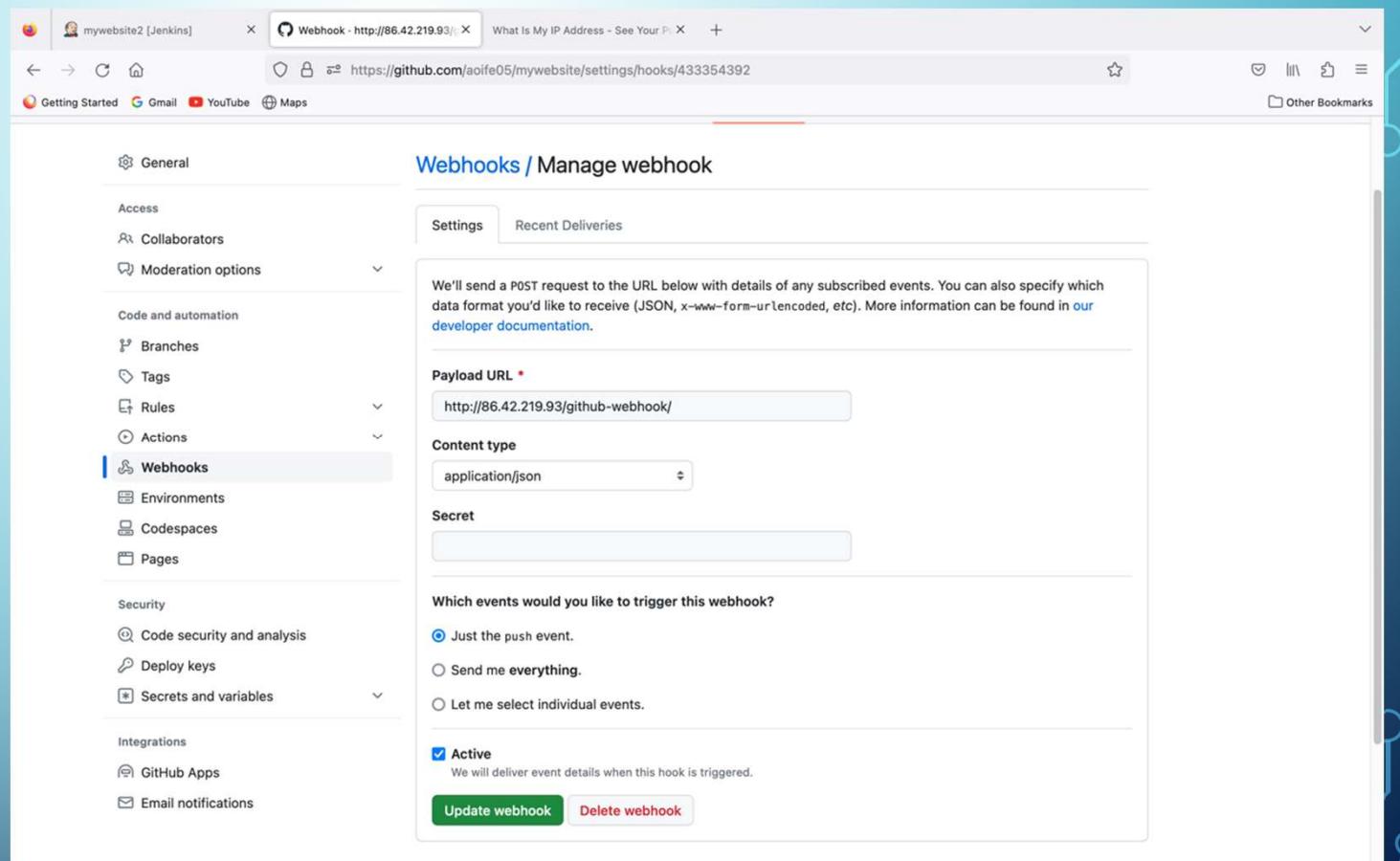
# GITHUB SHARED & CLONED & TALKING TO VS CODE



# DOCKER DESKTOP RUNNING & TALKING TO VISUAL STUDIO CODE



# JENKINS: TALKING TO GITHUB VIA WEBHOOKS



# JENKINS: DEPLOYMENT CONFIGURED

The screenshot shows the Jenkins Pipeline mywebsite2 Stage View. On the left, there's a sidebar with various options like Status, Changes, Build Now, Configure, Delete Pipeline, Full Stage View, Rename, Pipeline Syntax, GitHub Hook Log, Build History (selected), and Filter builds... Below the sidebar, a build history section shows a single build (#1) from Sep 13, 2023, at 1:52 PM. At the bottom of this section are links for Atom feed for all and Atom feed for failures.

The main area is titled "Pipeline mywebsite2" and "Stage View". It displays four stages: Declarative: Checkout SCM, Build, Dockerize, and Deploy to AWS Elastic Beanstalk. The "Declarative: Checkout SCM" stage took 2min 0s. The "Build" stage took 2s. The "Dockerize" stage took 279ms. The "Deploy to AWS Elastic Beanstalk" stage took 224ms. All stages are marked as failed. A summary table shows these times:

Declarative: Checkout SCM	Build	Dockerize	Deploy to AWS Elastic Beanstalk
2min 0s	2s	279ms	224ms
2min 0s	2s	279ms	224ms

Average stage times: 2min 0s, 2s, 279ms, 224ms.

Below the stage view, there's a "Permalinks" section with a link to the last build.

At the bottom right, there are links for REST API and Jenkins 2.414.1.

# EC2 CONFIGURED & UP AND RUNNING

The screenshot displays three separate tabs from the AWS Management Console:

- EC2 > Instances > i-0df46ffdb42c735d0 (LiveE2Pets)**: Shows the instance summary for the specified instance. Key details include:
  - Instance ID: i-0df46ffdb42c735d0 (LiveE2Pets)
  - Public IPv4 address: 3.89.136.144
  - Instance state: Running
  - Private IP DNS name (IPv4 only): ip-172-31-40-81.ec2.internal
  - Instance type: t2.micro
- AmazonSSMRoleForInstancesQuickSetup**: Shows the security group configuration for the instance. It lists the security group (sg-0f7ca8ed7ae5cc8aa) and its inbound rules. The rules allow traffic on ports 443, 22, 80, and 8080 from 0.0.0.0/0 to specific security group IDs.

Name	Security group rule ID	Port range	Protocol	Source	Security groups	Description
-	sgr-0698121a79e20a385	443	TCP	0.0.0.0/0	LiveSecurityGroupPets	-
-	sgr-021f9564bc757e3c	22	TCP	78.18.92.189/32	LiveSecurityGroupPets	-
-	sgr-03ee1fb09b81e4ad	80	TCP	0.0.0.0/0	LiveSecurityGroupPets	-
-	sgr-0b564f3b3a3c452t20	8080	TCP	0.0.0.0/0	LiveSecurityGroupPets	-
- Subnets (6)**: Shows the list of subnets in the VPC. There are six subnets listed, all in an available state across different VPCs and with various IPv4 CIDR ranges.

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR	Available IPv4 addresses
-	subnet-0b942844eaefcf562	Available	vpc-0dc12fee9f78142e1	172.31.32.0/20	-	4089
-	subnet-0b9bd01f2d716d317	Available	vpc-0dc12fee9f78142e1	172.31.64.0/20	-	4091
-	subnet-0e15863b0edc33168	Available	vpc-0dc12fee9f78142e1	172.31.0.0/20	-	4091
-	subnet-0ae8e2b7bfcaf44ce	Available	vpc-0dc12fee9f78142e1	172.31.48.0/20	-	4091
-	subnet-0830d094bf99477e	Available	vpc-0dc12fee9f78142e1	172.31.16.0/20	-	4091
-	subnet-09a7afb632bc920a9	Available	vpc-0dc12fee9f78142e1	172.31.80.0/20	-	4091

# AMAZON LINUX 2023 SERVER ACCESSIBLE & JENKINS IS RUNNING

```
[ec2-user@ip-172-31-40-81:~]
[ec2-user@ip-172-31-40-81:~]$ login as: ec2-user
[ec2-user@ip-172-31-40-81:~]$ Authenticating with public key "Live_KeyPair_Pets"

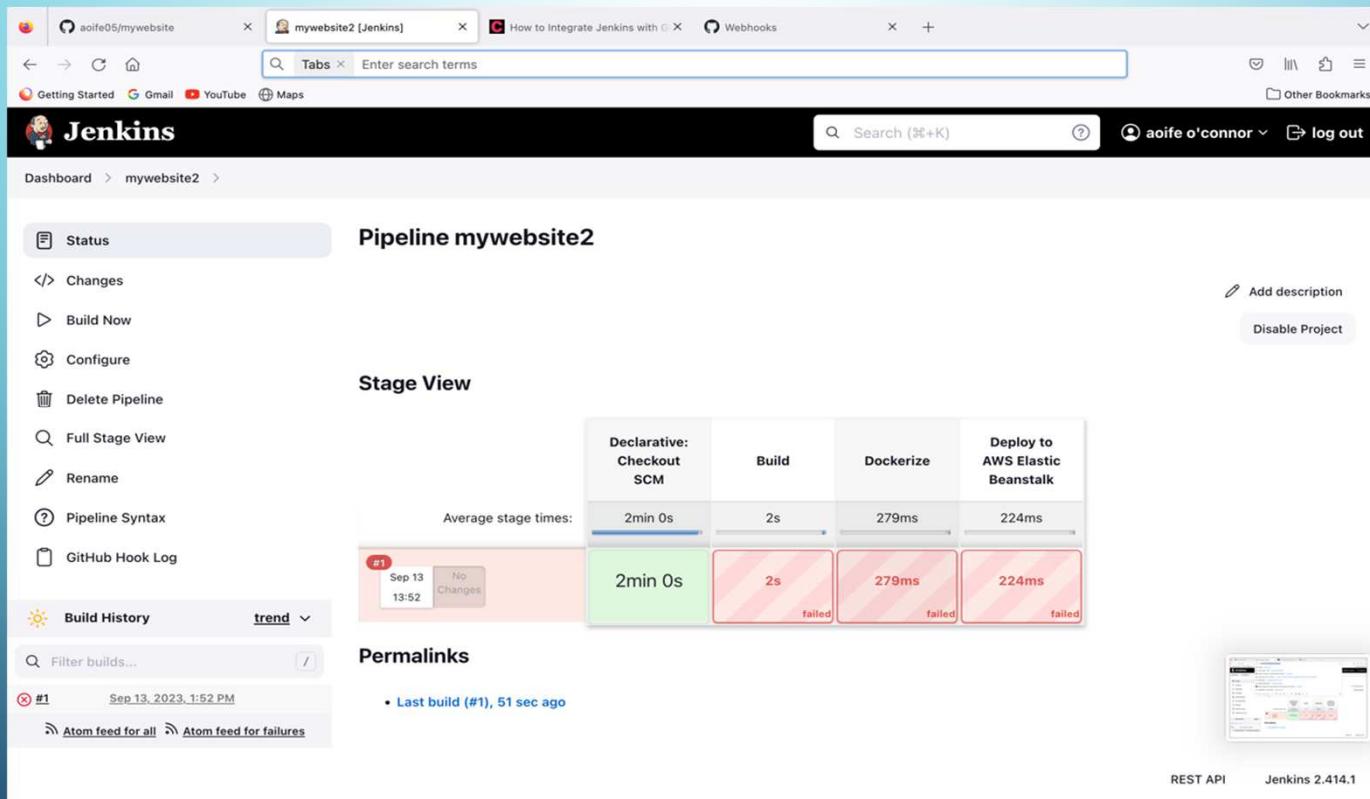
A newer release of "Amazon Linux" is available.
Version 2023.1.20230912:
Run "/usr/bin/dnf check-release-update" for full release and version update info
      _#_
     ~\ _###_          Amazon Linux 2023
    ~~ \####\
    ~~ \###|
    ~~  \|/ https://aws.amazon.com/linux/amazon-linux-2023
    ~~   V~'-'>
    ~~   /
    ~~ .-' /'
    /m'`'

Last login: Tue Sep 12 21:55:00 2023 from 78.18.92.189
[ec2-user@ip-172-31-40-81:~]$ jenkins --version
2.414.1
[ec2-user@ip-172-31-40-81:~]$ jenkins service start
Running from: /usr/share/java/jenkins.war
webroot: /home/ec2-user/.jenkins/war
Exception in thread "main" java.lang.IllegalArgumentException: Multiple command
line argument specified: start
        at winstone.CmdLineParser.parse(CmdLineParser.java:68)
        at winstone.Launcher.getArgsFromCommandLine(Launcher.java:506)
        at winstone.Launcher.main(Launcher.java:468)
        at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Nativ
e Method)
        at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(Native
MethodAccessorImpl.java:62)
        at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(De
legatingMethodAccessorImpl.java:43)
        at java.base/java.lang.reflect.Method.invoke(Method.java:566)
        at executable.Main.main(Main.java:347)
[ec2-user@ip-172-31-40-81:~]$
```



WHAT'S LEFT TO DO

# KNIT & TEST



# CHALLENGES



TIME

- Don't know if we have the right plan of action
- Don't know what we don't know!
- So many applications to understand!
- Will costs be incurred due to lack of knowledge/in-experience on AWS?

TIME