Developers Tools

CodeCommit, CodeBuild, CodeDeploy & CodePipeline

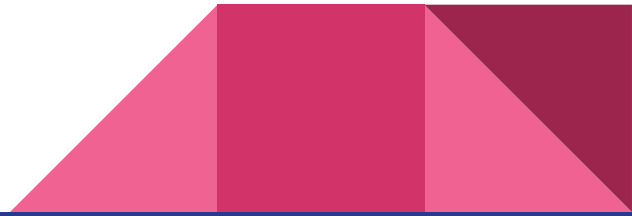
Building a Angular Web Application [Cycling Experience] and Hosting at AWS Cloud EC2

Presented by Wong Teck Choy, 31 August 2024

AWS Developer Tools

AWS Developer Tools is a set of services designed to enable developers and IT operations professionals to rapidly and safely deliver software.

This suite includes **CodeCommit**, **CodeBuild**, **CodeDeploy** and **CodePipeline**, which together facilitate continuous integration and continuous delivery (CI/CD) of applications.

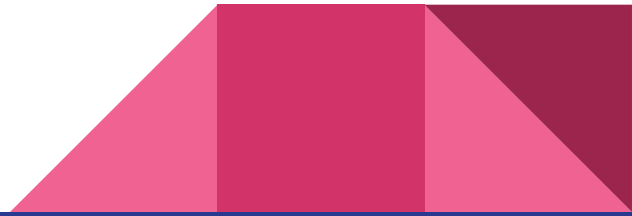


What does this AWS product do?

- **CodeCommit:** is a fully **managed source control service** that hosts Git repositories. It allows teams to store code securely and manage the lifecycle of their software development.
- **CodeBuild**: A fully **managed build service** that compiles source code, runs tests, and produces software packages ready for deployment. It scales continuously and processes multiple builds concurrently.
- **CodeDeploy**: A **deployment service** that automates the process of deploying applications to various compute services like Amazon EC2, AWS Lambda, and on-premises servers. It ensures deployments are safe and repeatable.
- **CodePipeline**: A **continuous integration and delivery service (CI/CD)** for fast and reliable application and infrastructure updates. CodePipeline **automates** the build, test, and deploy phases of your release process every time there is a code change.

Who will use this AWS Developer Tools

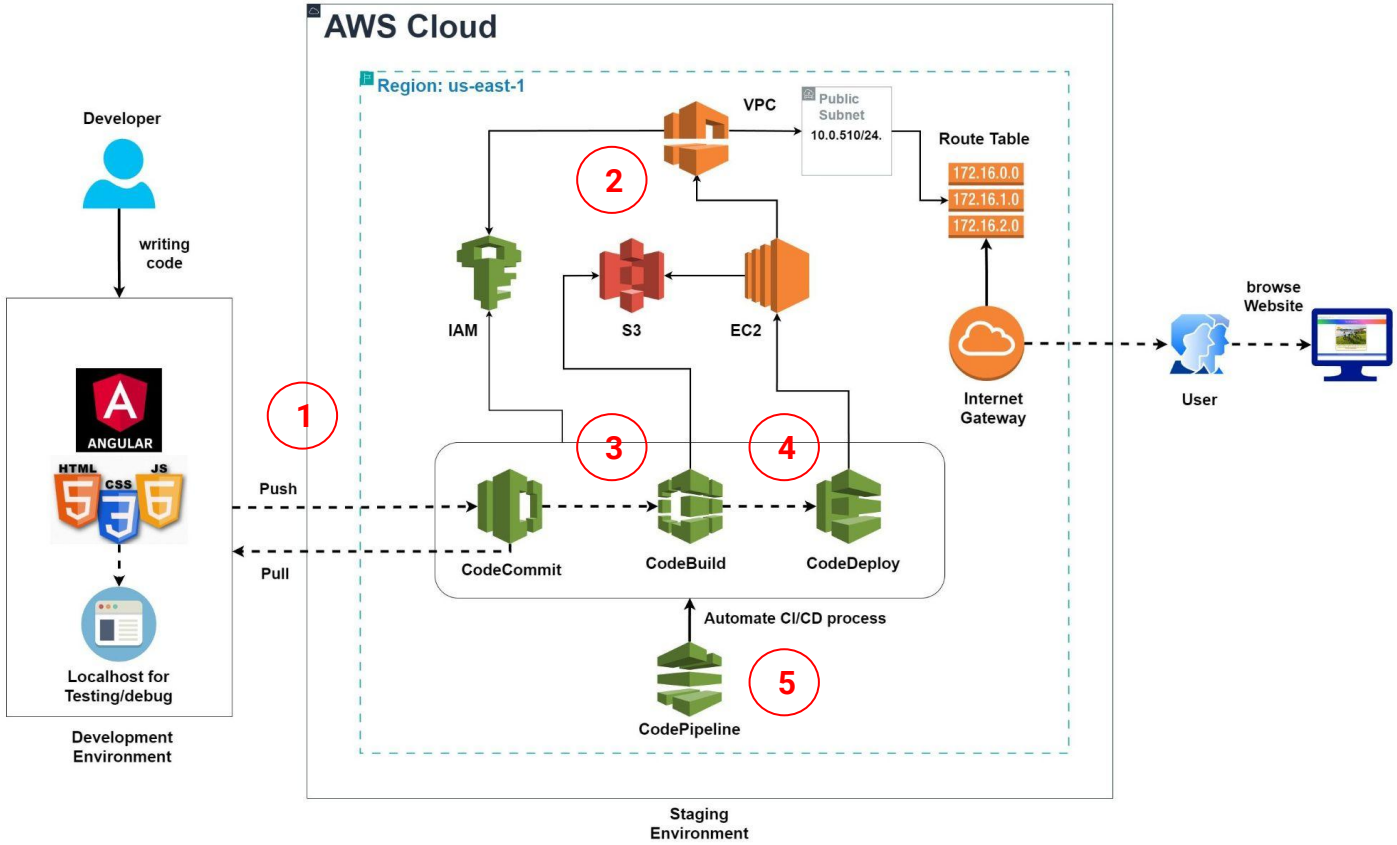
- **Software Developers** : Who need tools to build, test, and deploy code efficiently.
- **DevOps Engineers** : Who automate the software delivery process to improve deployment speed and reliability.
- **IT Operations Teams** : Who manage infrastructure and application updates.
- **QA Teams** : Who integrate testing into the CI/CD pipeline to ensure code quality



Agenda

1. Setup **CodeCommit** repository and upload the source code.
2. Setup **VPC**, **S3 bucket** and **EC2**
3. Setup **CodeBuild**
4. Setup **CodeDeploy**
5. Setup **CodePipeline**

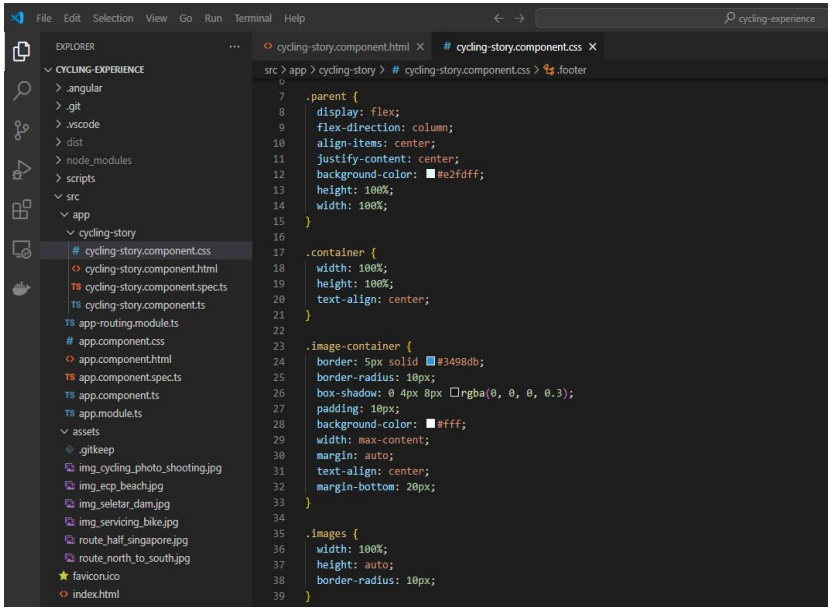
Architecture to Host Web Application in AWS Cloud

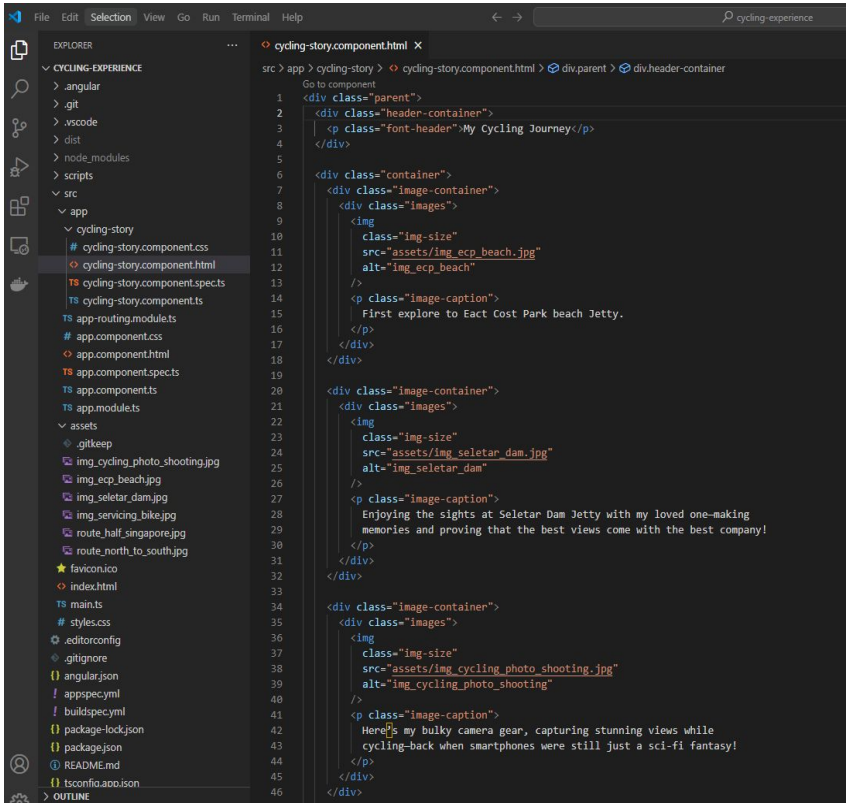


1. Setup CodeCommit repository and upload the source code

CodeCommit

- Creating **Angular web application** project using **Visual Studio Code** with **Angular Framework.**
- Writing code, build and testing at **local environment.**





CodeCommit

- Create “**buildspec.yml**” file.
- This is a configuration file used by **AWS CodeBuild** to define the **build process** for your project. It provides **instructions** on how to build, test, and package your code.

buildspec.yml

```yaml
version: 0.2
env:
  variables:
    APP_NAME: "cycling-experience"
phases:
  install:
    runtime-versions:
      nodejs: 20
    commands:
      - echo install process started
      - npm install && npm install -g @angular/cli
  build:
    commands:
      - echo build process started now
      - ng build
  post_build:
    commands:
      - echo build process finished, we should upload to S3 now
      - cp appspec.yml dist/$APP_NAME
      - cp -r scripts dist/$APP_NAME
      - ls -la dist/$APP_NAME
artifacts:
  files:
    - '**/*'
    - appspec.yml
    - scripts/**
  base-directory: 'dist/$APP_NAME'
```

CodeCommit

- Create “**appspec.yml” file** and **scripts**
- This file is crucial for **AWS CodeDeploy** to handle the deployment process on EC2 instance, such as **executing scripts** to install the Apache server and **managing services**.

scripts > $ install_dependencies.sh

```
1  #!/bin/bash
2  sudo yum install httpd -y
```

scripts > $ start_server.sh

```
1  #!/bin/bash
2  sudo service httpd start
```

scripts > $ stop_server.sh

```
1  #!/bin/bash
2  isExistApp = `pgrep httpd`
3  if [[ -n  $isExistApp ]]; then
4      sudo service httpd stop
5  fi
```

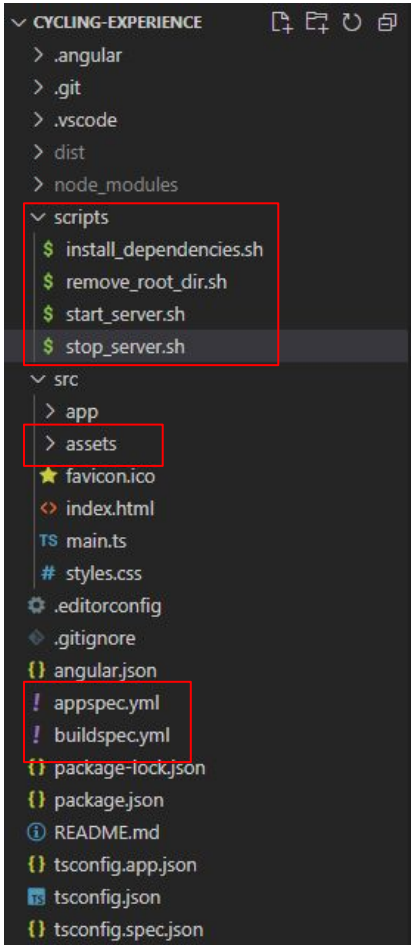
! appspec.yml

```
1   version: 0.0
2   os: linux
3   files:
4     - source: /
5       destination: /var/www/html
6   hooks:
7     BeforeInstall:
8       - location: install_dependencies.sh
9         timeout: 300
10        runas: root
11    ApplicationStart:
12      - location: start_server.sh
13        timeout: 300
14        runas: root
15    ApplicationStop:
16      - location: stop_server.sh
17        timeout: 300
18        runas: root
```

CodeCommit

Your project development **files and folder structure**.

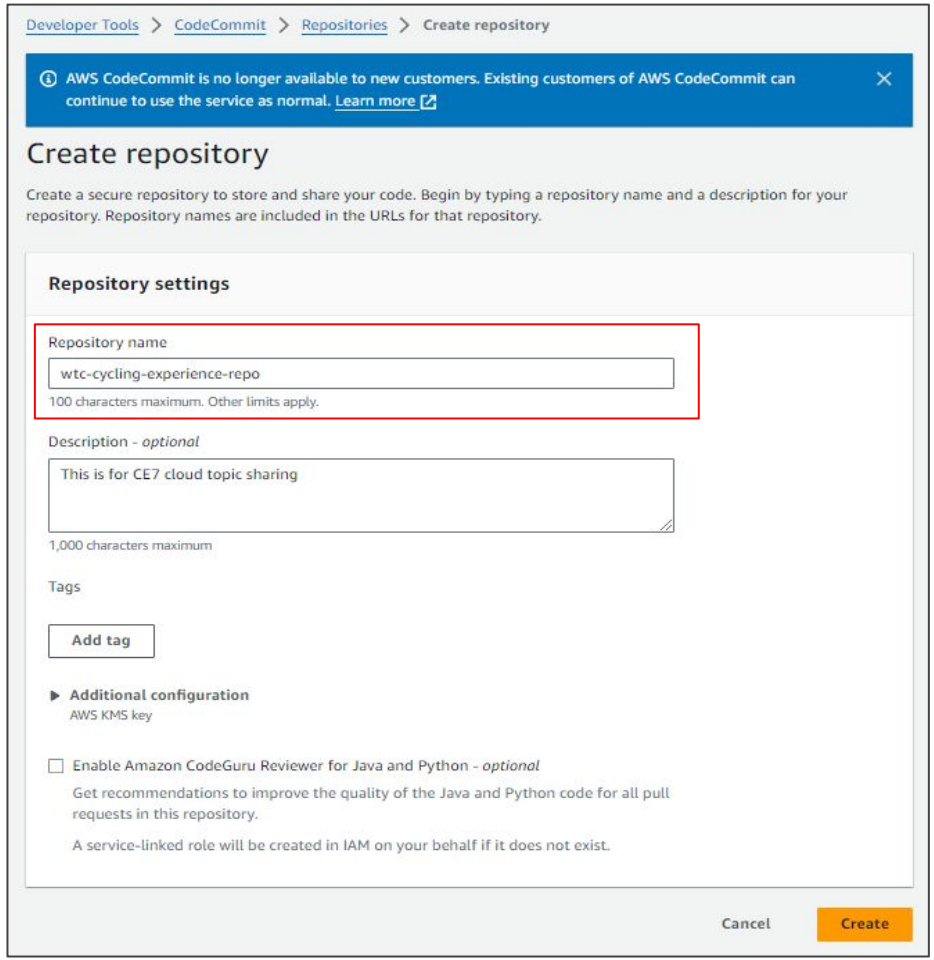
- Angular project source code and **app** folder
- All **yml** files
- **scripts** folder - all scripts file
- **assets** folder - all documents or image files



CodeCommit

Create new **CodeCommit repository**

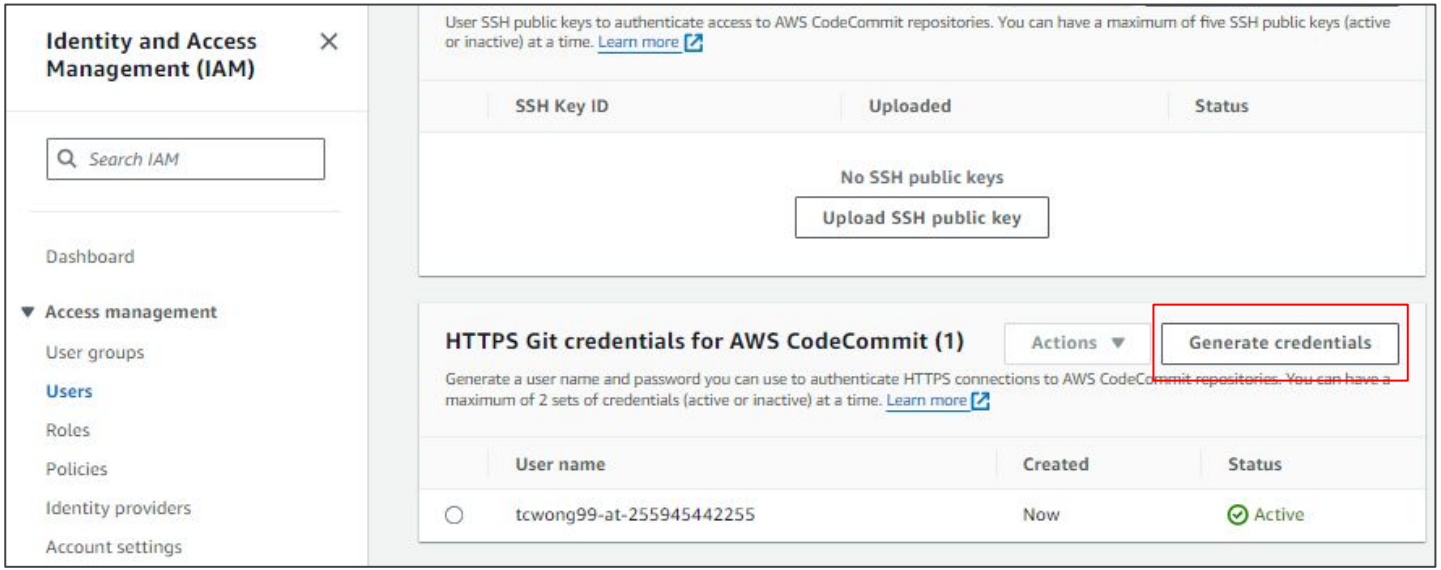
1. Repository name : **wtc-cycling-repo**
2. Enter the **description** and click “**Create**” button.



CodeCommit

Generate credentials for **HTTPS Git credentials** for AWS CodeCommit

1. Go to **IAM**->Users->HTTPS Git credentials for AWS CodeCommit->Click on “**Generate credential**” button.
2. Download the credentials **csv file** on your local computer, it contain your **username** and **password**.

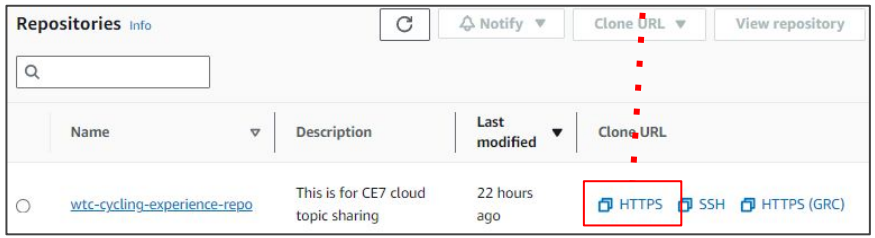


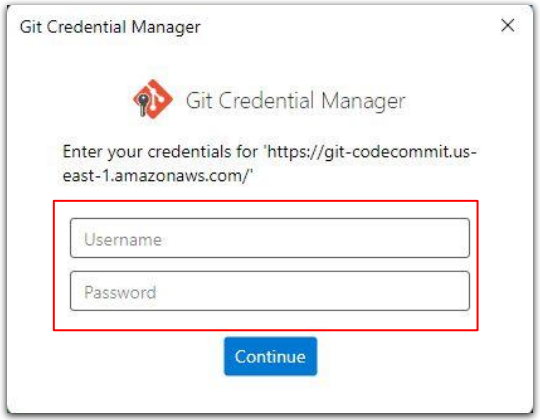
CodeCommit

Upload your code to CodeCommit

1. Please refer to the downloaded **csv file**. You may need to enter **username** and **password** if prompted.
2. Below git commands use to upload source code to AWS CodeCommit.

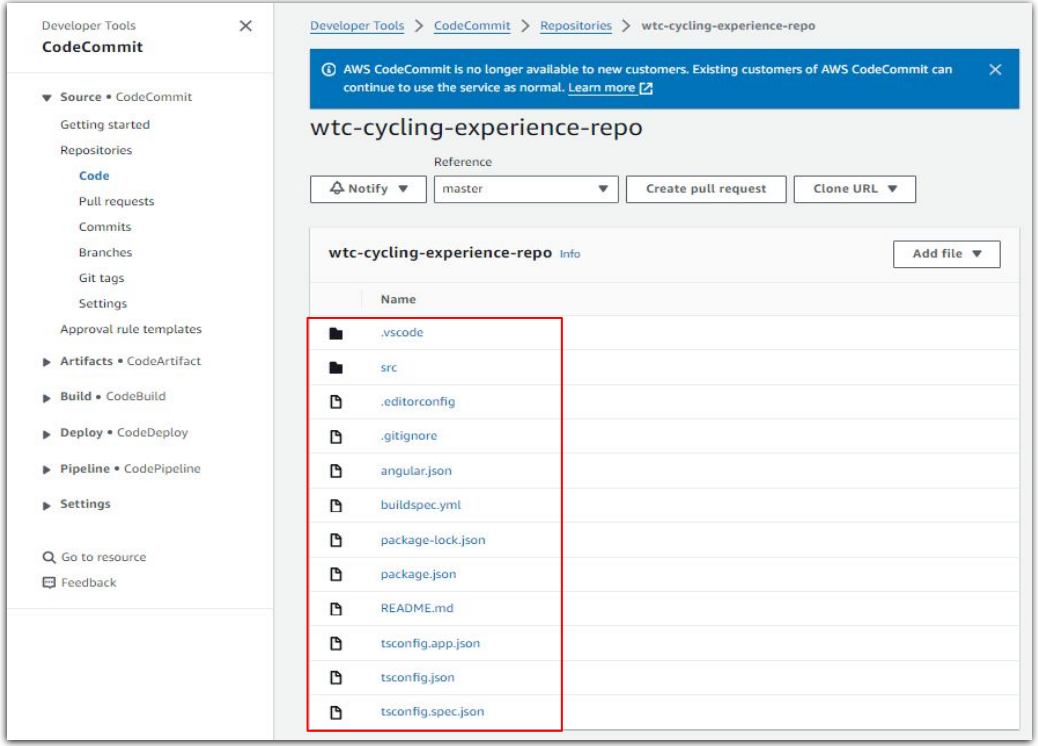
- git clone **<Your CodeCommit repository HTTPS url>** or
  git remote add origin **<Your CodeCommit repository HTTPS url>**
- git status
- git add .
- git commit -m “Initial project code”
- git push -u origin master
- git pull origin master





CodeCommit

- Once git push successfully execute, your source code will be available at AWS CodeCommit repository.
- To check source code go to CodeCommit -> Repository -> <Choose your repository>



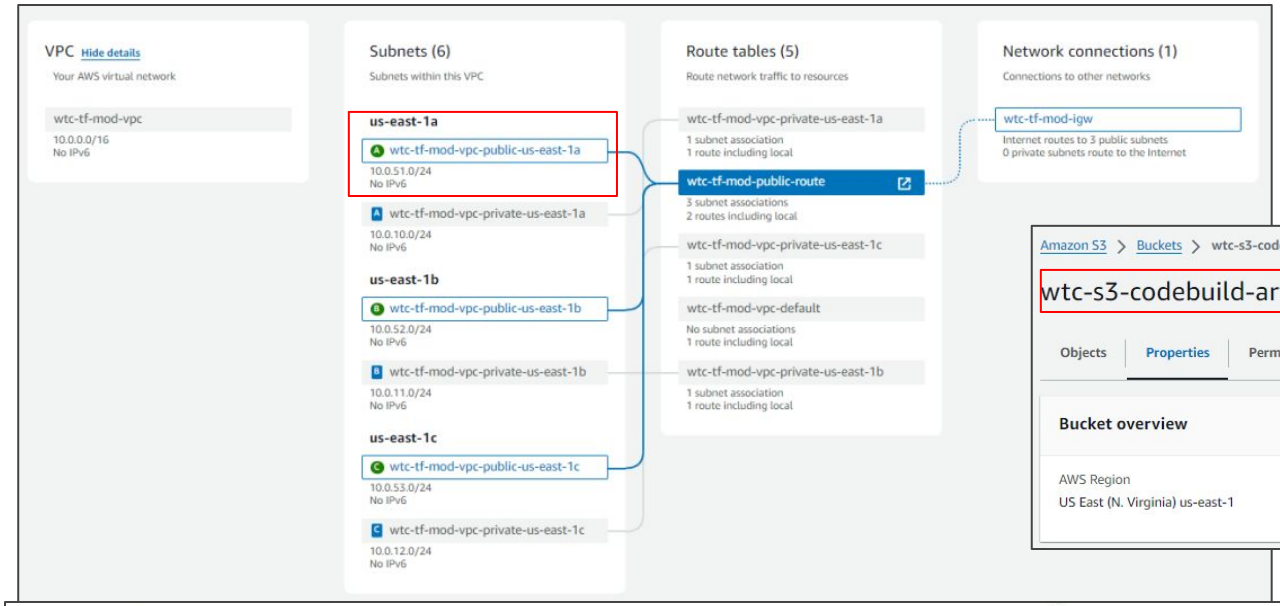
2. Setup VPC, S3 bucket and EC2

Setup VPC, S3 bucket and EC2

1. Setting up a **VPC** with at least 1 public subnet link to internet gateway.
2. Create a **S3 Bucket** use for CodeBuild.
3. Setting up a **EC2 using Amazon Linux** as OS.
    - Enter below command for EC2 **User data**

```
#!/bin/bash
sudo yum update -y
sudo yum install ruby -y
sudo yum install wget -y
cd /home/ec2-user
wget https://aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com/latest/install
sudo chmod +x ./install
sudo ./install auto
systemctl status codedeploy-agent
systemctl start codedeploy-agent
```

Setup VPC, S3 bucket and EC2



3. Setup CodeBuild

Demo in video

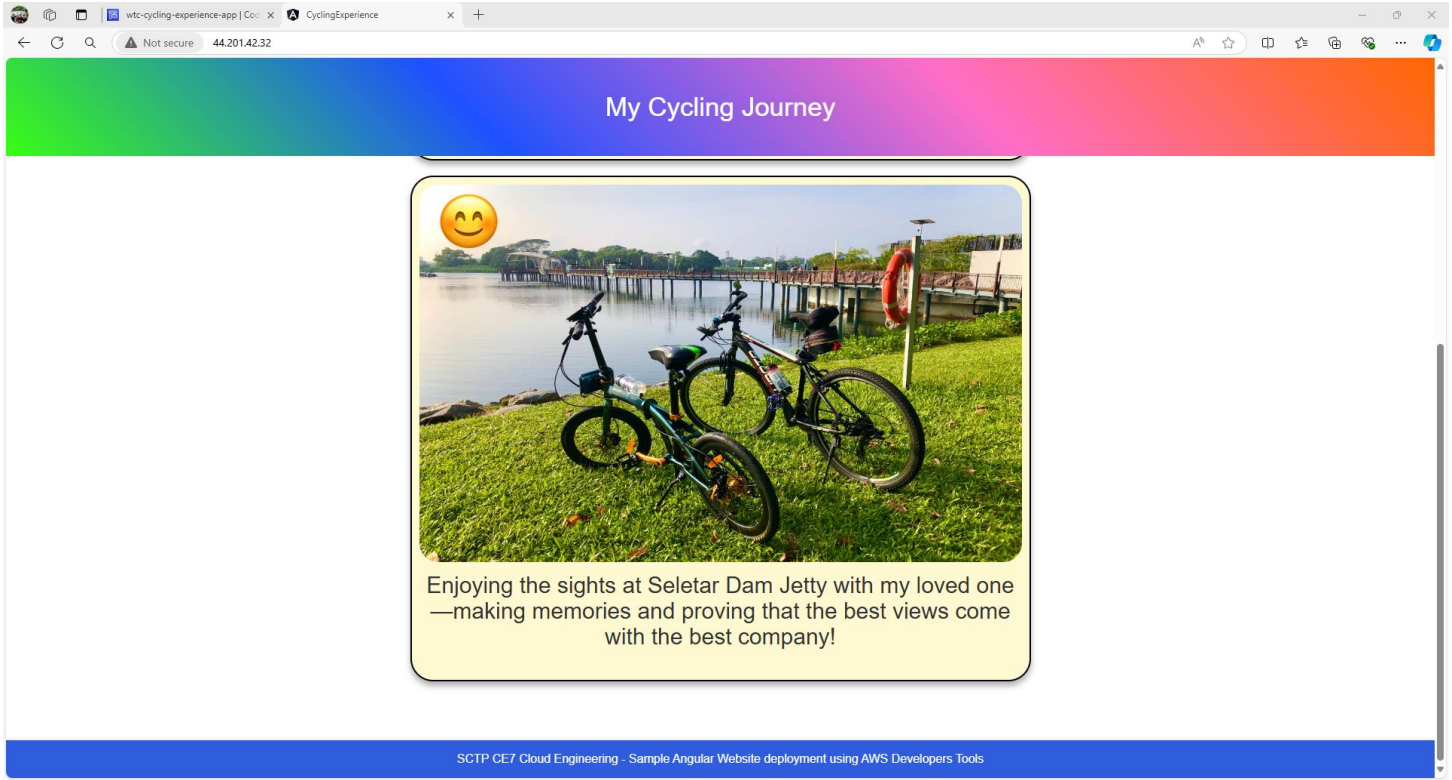
4. Setup CodeDeploy

Demo in video

5. Setup CodePipeline

Demo in video

Website Now Running on AWS Cloud



End of Topic Sharing

Thank you!