Professional Computing - CITS3200 Team 30

**Mentor Meeting #2**horizontal line

**Date:** 09/09/22 **/ Time:** 2.00pm **/ Location: Online** (teams) **|** In Person - EZONE 2A

**Chair:** Edward

**Minutes:** Alexia

**Attendees:** Edward Giles, Alexia Fassetta, Nicodemus Ong, Judy Cooper, Brian Hislop, Jason Miller

**Apologies:** Zack Zou, Jordan Lee

## **Minutes Duration: 45 mins**

# The meeting was declared open at 2pm, quorum was present.

# The meeting was declared closed at 2.45pm, quorum was present.

## **Confirmation of Previous Minutes**

**Motion:** “To accept the 28/08 minutes as true and correct”

**Moved:** Alexia

**Moved Unanimously** / Moved by Majority

**AGENDA**

| **Topic:** | **Actioned**: |
| --- | --- |
| Meet Jaybro | All |
| AWS Key notes | All |
| AWS processing explained further | All |
| Edward’s Questions | Edward |

## **1.1 Meeting Jaybro**

Background

* Electrical Engineering
* IBM - Developer / Operations
* Navtovs - education facilities,working with AWS
  + Tried Azure

## **1.2 AWS Key Notes**

**Process:**

1. github
2. AWS Codebuild ( Web hook into git hub )
   1. Create Docker Images
   2. Checkin images into ECR( Elastic Code Registry)
3. AWS CodeDeploy to push your images to an ECS Instance

Recommend playing around with EC2 instances running ubuntu22.04lts.

Store your passwords into AWS Secrets. Allow the EC2 instances to access those secrets.

## **1.3 Processing Explained Further:**

Code repository inside AWS which will hook it into our Github repository

Create a EC2 instance → Boto3 library → pick a boot → operating system → install docker

Push the code to the EC2 Instance (Hooks into Github and AWS) → code build pipeline **(**Service in AWS / lynx container) → create the docker image after you push the code (will invoke handle and visually show where the code is)) → check into the ECR (elastic container registry) or ECS→ Deploy face which pushes into the EC2 instance.

Push the code into repository (github) pushes the trigger (web Hook) which lets code pipeline know there has been a change

Other terminology

* AWS Secrets → separate service → Generate tokens
* Fargate (operating system) - Coopernettys
* Docker to be passed as an environment variable → 12 factor.net (good knowledge)

**1.4 Edward’s Questions**

1. Azure Vs AWS in terms of setting up

* In Azure (Container Instance)- we would create a container repository to hold code from front end
* 2 containers - docker composed brings this up.

1. Considerations - cost

* AWS - Elastic container Services
* Microsoft - Azure Container Instance