Project Cloud Computing

In this project, students will deploy a **Todo List Application** in AWS. Students are free to choose technology or services they want to use to achieve this objective. This project aims to allow students to put their understanding of previous labs into further practice. Students will work in teams that have been assigned at the beginning of the course. Each team will work together to submit this project and present their project at the end of the course. Each team will spend 10-15 minutes to present their project with 5-10 minutes for questions.

Requirements:

- Deploy the dockerized existing Todo List Application https://github.com/dockersamples/todo-list-app
- Use Infrastructure as Code (IaC) to set up AWS cloud infrastructure for deployment.
- Store application data in a cloud database like **Amazon RDS.**
- Configure CI/CD Pipeline to automate the deployment of code changes.

Recommended Steps:

1. Push and commit code to Github repository. Make sure that you commit and push your code regularly during the course of your changes. Do not work locally and push code only once at the end.
2. Before deploying to EC2 or ECS, ensure that the app works locally. Run your application locally with docker as you have done in previous lab.
3. Set Up an EC2 Instance and install Docker on EC2.
4. Set Up database RDS. The to-do list application is using MYSQL docker container. In this project, you will need to make your own changes and deploy a database using AWS RDS.
5. Set Up CI/CD Pipeline with pipeline of your choice e.g. AWS Code Build, Code Pipeline, Travis CI or Github Action

Assessment Criteria:

- Is the Docker container built and deployed successfully on EC2?
- Does the EC2 instance have the proper configurations to run Docker (installation, permissions, etc.)?
- Does the application interact correctly with the database (RDS/DynamoDB)?

- Is the database connection secured and configured properly?
- Is the CI/CD pipeline fully automated and does it deploy the app reliably from source to production?
- Are build and deploy stages working as expected?
- Is the infrastructure deployment automated using IaC?
- Is the infrastructure modular and reusable?
- **Bonus points: Documentation, Scalability and Security**