CDS Technical Challenge - Devops

THE BERKELEY PROJECT

Citizen Disbursement System (CDS) Team

BACKGROUND

In 1972, the City of Carson was a growing city that houses close to millions of working class heroes. To support the growing city population, a bill was passed that:
‘each working class hero is to contribute a fraction of their yearly income towards city building’
This year, as part of the governor’s initiative to pony up surplus cash in the vault, each working class hero is gifted with taxation relief as recognition for their voluntary contribution to city building efforts.

The governor wishes to deploy this initiative as fast as possible. Also, he wishes to aim for the cloud - AWS, specifically.

To facilitate this, the governor has drafted out the Berkeley Project. This project aims to create a CI/CD pipeline to deploy the application to the cloud.

The governor wishes the following deliverables:

- A dockerize application
- A ci/cd pipeline solution
- A cloud infrastructure diagram

^: also known as employees

TASK

Deliverables within One Week.

Your objective is to provide the following deliverables within one week, utilizing specified technologies. Ensure meticulous tracking of your progress by committing frequently and maintaining a detailed log of encountered problems and their resolutions.

1. **[Application]**
   Create an application featuring a webpage displaying the message "This is the <x> visitor," where <x> is a counter fetched from Redis. The counter should increment with each visit to the index page. The application should be dockerize, and provide the link in github.

2. **[CI/CD Pipeline]**
   Devise a plan for the automated deployment of the above application to production using GitLab. Each stage should be documented in detail.

3. **[Cloud infrastructure]**
   Deploy an Amazon EKS cluster within the AWS ap-southeast-1 region using Infrastructure as Code (IaC) with Terraform. Do provide the architecture diagram as well.

Requirements

- Public Accessibility:
  - Ensure the application is publicly accessible via a designated public URL.
- Scalability:
  - Design the infrastructure for effective scaling to handle varying workloads.
- Version Control:
  - Upload your complete implementation, encompassing all infrastructure and resources, to a private Git repository.
- Terraform Integration:
  - Given our reliance on Terraform for deployment and maintenance, it is mandatory to present your solution using Terraform.
- Multiple environment:
  - Implementation should cater for different environments e.g. dev, uat, sit, prod

Ensure your AWS deployment aligns with best practices for security, reliability, and efficiency. Document your infrastructure design comprehensively, illustrating the cloud infrastructure layout to deploy the application seamlessly.

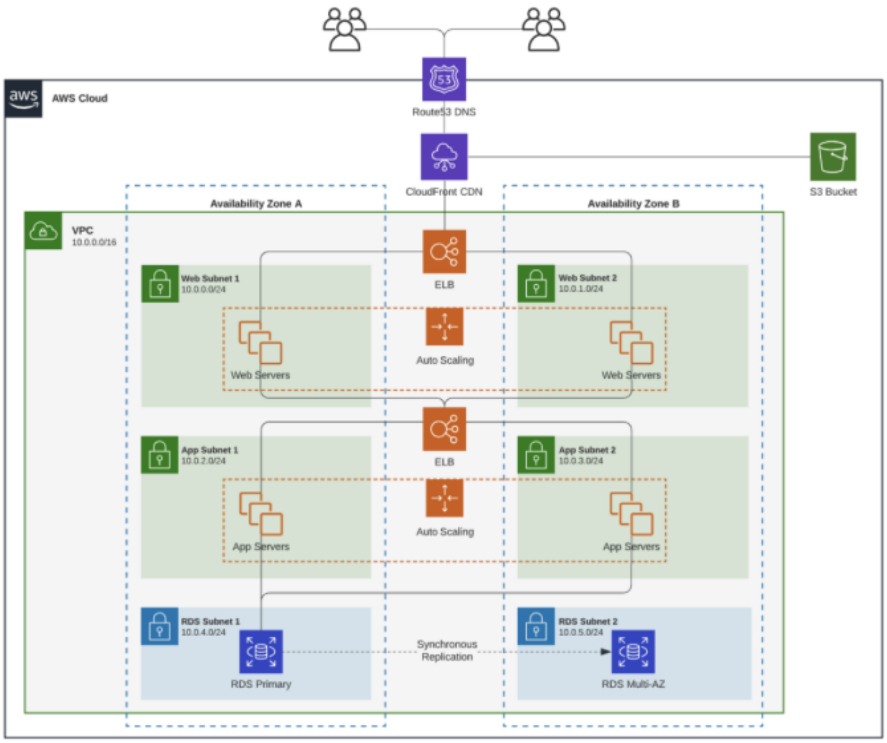
Sample deliverable format (This is just an example)

1. A link to the source code in github
   https://github.com/react-boilerplate/react-boilerplate

2. CI/CD Pipeline plan

Choice of ci/cd tools: Gitlab
Reason: "Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua”

3. Cloud infrastructure diagram



Looking forward to your solutions.
Good luck and enjoy!

Recommendation

- A few online tools that could help : https://cloudcraft.co/, https://www.draw.io/
- We appreciate creativity, curiosity, and thinking outside of the box. While deploying the resources feel free to add extra elements to your setup and use elegant approaches to deliver them, as long as you are able to demonstrate the added value of those and explain how they work.