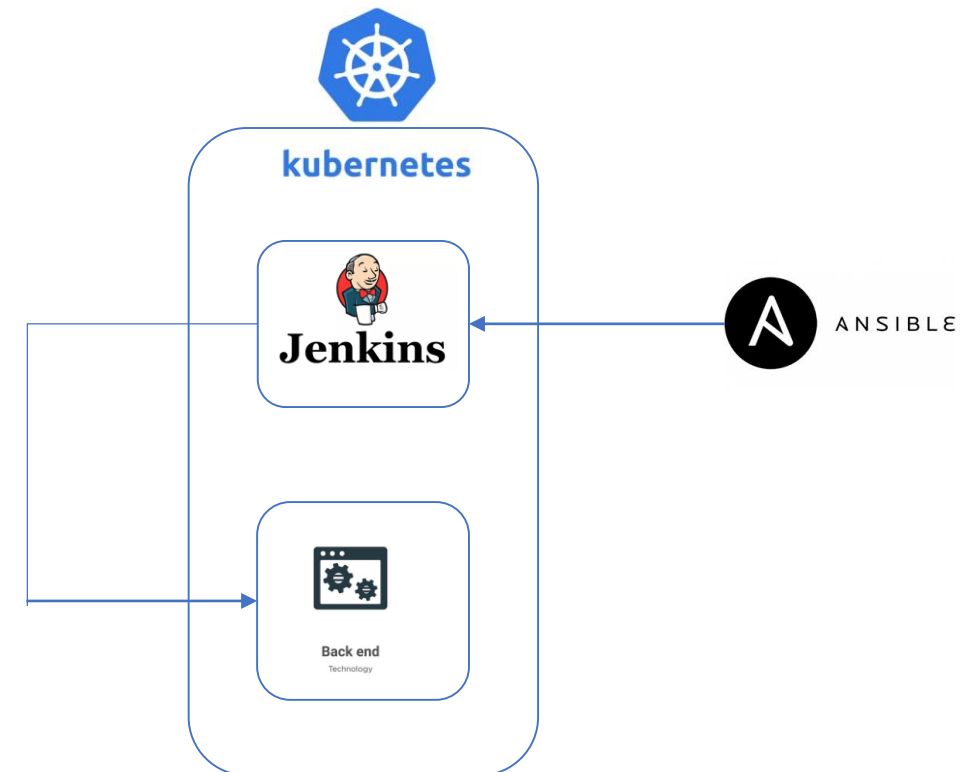# ITI Graduation Project

Important:

- Create a publicly accessible repository (GitHub preferred)

- Commit often (make your progress visible).

- It is OK to copy and paste snippets as long as you reference them.

- Just deliver the best you can within the constraints.
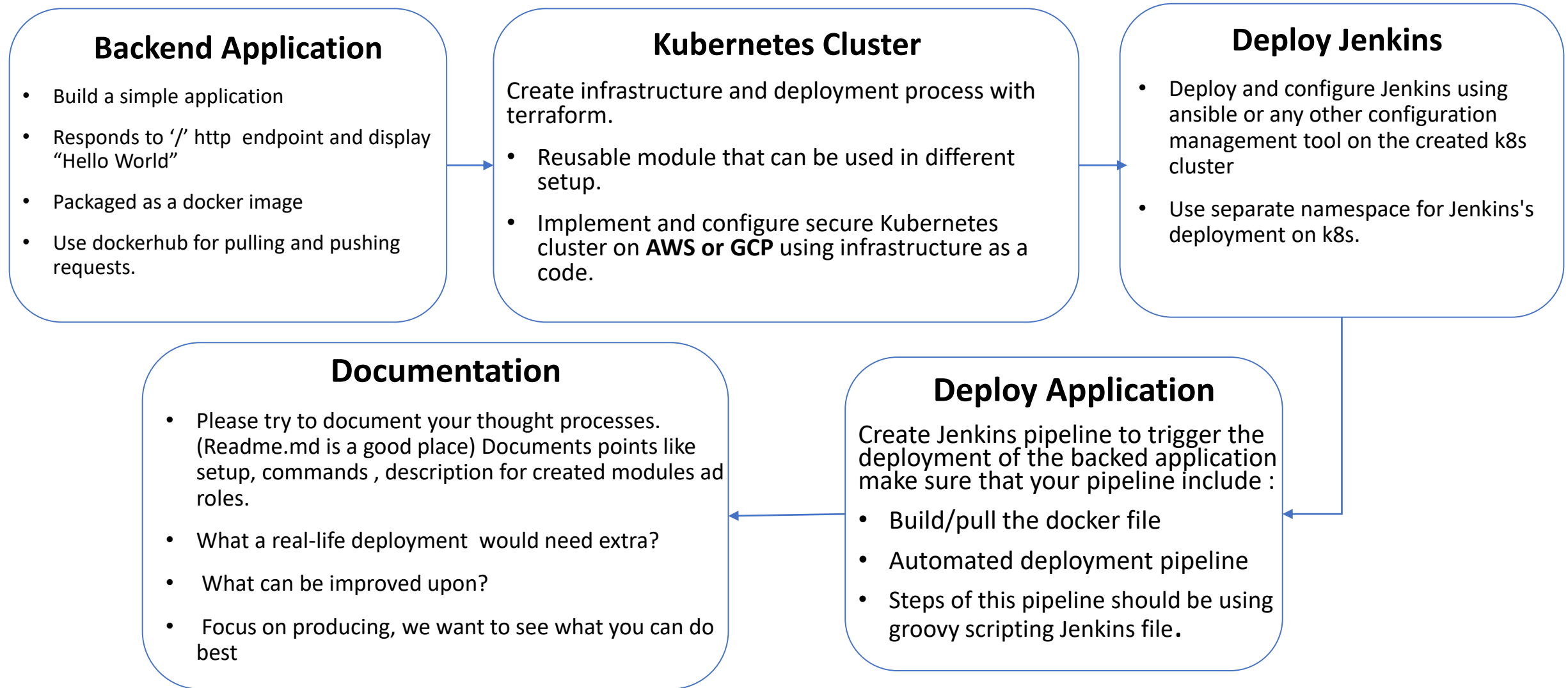
The Project : Backend application Deployment

Deploy backend application on kubernetes cluster using CI/CD jenkins pipeline using the following steps and high-level diagram :

1. Implement secure kubernetes cluster
2. Deploy and configure jenkins on kubernetes using Ansinle
3. Deploy backend application on kuberetes using jenkins piepline

Use the next slide for more details and requirments

The Project : Backend application Deployment

Backend Application

- Build a simple application

- Responds to '/' http endpoint and display "Hello World"

- Packaged as a docker image

- Use dockerhub for pulling and pushing requests.

Kubernetes Cluster

Create infrastructure and deployment process with terraform.

- Reusable module that can be used in different setup.

- Implement and configure secure Kubernetes cluster on **AWS or GCP** using infrastructure as a code.

Deploy Jenkins

- Deploy and configure Jenkins using ansible or any other configuration management tool on the created k8s cluster

- Use separate namespace for Jenkins's deployment on k8s.

Documentation

- Please try to document your thought processes. (Readme.md is a good place) Documents points like setup, commands , description for created modules ad roles.

- What a real-life deployment would need extra?

- What can be improved upon?

- Focus on producing, we want to see what you can do best

Deploy Application

Create Jenkins pipeline to trigger the deployment of the backed application make sure that your pipeline include :

- Build/pull the docker file

- Automated deployment pipeline

- Steps of this pipeline should be using groovy scripting Jenkins file.

Last Notices:

- Focus on producing, we want to see what you can do best
- Feel free to add more features if you wish.

Good Luck !