**Deploying Open5g applications on Kubernetes with continuous integration**

**(Akhil Reddy Nalla, 811172556,** [***analla3@kent.edu***](mailto:analla3@kent.edu)**)**

**(Divya Sri Karingula, 811172271 -** [**dkaringu@kent.edu**](dkaringu@kent.edu%20) **)**

**Introduction:**

* This Open5g can be used to configure your own NR/LTE network. If gNB/eNB and USIM are available, you can build a private network using Open5GS.
* Open5GS implemented 5GC and EPC using C-language and WebUI is provided for testing purposes and is implemented in Node.JS and React.

**Project Statement & Solution:**

**Problem**: Open5g Application developers can repeatedly build, test, and deploy newer versions of their applications, no matter where they are running and without any focus on Provisioning, Deploying Updating, Upgrading, availability & scalability

**Solution**: The main goal of this project is to create a platform to maintain Continuous integration, Continuous deployment, and continuous testing for an Open5g system over cloud

The main aim of this project is to provide Open5g Application Deployment pipeline end to end automation using GitHub-Jenkins-Docker-Kubernetes

GitHub - Source Code Management Tool

Where developers push their code into repository by creating a branch.

JENKINS - for continuous integration and continuous delivery where we fetch the source code from pipeline and build a customized image using DOCKER and push the image to docker Registry

Docker - Containerization Tool, to build customized images & push the customized images to Docker Hub Registry

KUBERENTES - for Orchestration where we deploy the manifest files using JENKINS pipeline

**Tools**

* Git
* GitHub Code Repo
* AWS Management Console
* AWS Service like Ec2, EKS
* Docker Engine
* Docker Hub Registry
* Kubernetes
* Jenkins

**System Architecture:**

Graphical user interface, application

Description automatically generated

**Fig**: Project workflow

Here, an Open5g developer has developed code and pushed it to the Open5g GitHub repository (https://github.com/open5gs/open5gs). From there, we will clone the code and write Docker files, build Docker images, push those images to the Docker Hub Registry, and then deploy those images across multiple containers using Kubernetes, all while utilizing the AWS cloud environment for maintaining Ec2 instances (for maintaining Linux-based operating system machines) and AWS Elastic Kubernetes Service (for maintaining for too many containers / cluster) and entire flow will continuously integrate with help of Jenkins.

**Reference**

1. <https://github.com/open5gs/open5gs>
2. Nikolaos Apostolakis, Marco Gramaglia, and Pablo Serrano - Design and Validation of an Open-Source Cloud Native Mobile Network
3. <https://open5gs.org/open5gs>
4. <https://medium.com/rahasak/5g-core-network-setup-with-open5gs-and-ueransim-cd0e77025fd7>
5. Docker Hub Account: <https://hub.docker.com/u/akhilreddy62>
6. GitHub Repo link: <https://github.com/AkhilReddy62/open5g>