**Final Project-1**

**Application Deployment**

**GitHub Repo URL**: [**https://github.com/Nandoo-03/Final-Project**](https://github.com/Nandoo-03/Final-Project)

Deployed site URL: http://<EC2-Public-IP>:80/  
Port number: 80

**DockerHub Repo URL**: <https://hub.docker.com/repository/docker/nandoo03/prod/general>

A screenshot of a computer

AI-generated content may be incorrect.

**Note: Created 2 repos for dev and prod to push images**

* **Prod Repo (PRIVATE):** **nandoo03/prod**

A screenshot of a computer

AI-generated content may be incorrect.

* **Dev Repo (PUBLIC):** **nandoo03/dev**

A screenshot of a computer

AI-generated content may be incorrect.

**Version Control:**

**Note:** All Git operations, including pushing code to the dev branch and merging, were performed using the CLI. Merging was carried out both through the CLI and via direct pull requests on GitHub.

**Webhook:**  
**Note:** Configured Jenkins to automatically trigger builds from GitHub on both dev and master branches.

A screenshot of a computer

AI-generated content may be incorrect.

**Jenkins:**

**Note:** The project has been implemented using a Jenkins multi-branch pipeline to handle both dev and master branch workflows. And enabled necessary plugins for Jenkins (Docker, Git, AWS EC2, Pipeline stage view and multi branch pipeline

**A screenshot of a computer

AI-generated content may be incorrect.**

A screenshot of a computer

AI-generated content may be incorrect.

**Configuration:**

A white rectangular object with blue lines

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A white rectangular object with blue lines

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Dev branch stage view:**

**Note:** The deployment uses the **production image** built from the **master** branch. Therefore, changes pushed to the dev branch will not trigger a deployment. **Deployment occurs only after dev is merged into master.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Master branch stage view:**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Console Output:**

The console output is captured from the master branch, as deployments are triggered only when changes are merged into master.

A screenshot of a computer program

AI-generated content may be incorrect.

**AWS EC2 instance:**

Note: Although the project requirements specified **t2.micro**, it was unavailable. Therefore, a **t3.small** instance was provisioned for deployment.

**A screenshot of a computer

AI-generated content may be incorrect.**

**Security group and inbound rules:**

**Note:** SSH login to the server is restricted to my IP address only, as per the security configuration. Application access on port 80 is open for all users.

**A screenshot of a computer

AI-generated content may be incorrect.**

**Prometheus and Grafana:**

**Note:** I configured Prometheus and Grafana on AWS EC2 via a setup script (available in the GitHub repo). Application health monitoring was implemented using Blackbox Exporter, with its script also uploaded to the repository.  
 **Prometheus:**

A screenshot of a computer

AI-generated content may be incorrect.

**Grafana:**

**Node Exporter for system status:**

A screenshot of a computer

AI-generated content may be incorrect.

**Blackbox Exporter for Application status:**

A screenshot of a computer

AI-generated content may be incorrect.

**Deployed site:  
A screenshot of a cell phone

AI-generated content may be incorrect.**