

Week 3 Report – Continuous Deployment to Staging

Objectives

- Establish a **staging environment** for deployment.
- Automate deployment from GitHub to staging after successful builds.

Summary of Activities

1. Staging Platform Selection

- The team evaluated multiple platforms and selected **Render** for backend hosting and **Netlify** for frontend deployment.
- Decision made jointly by **Aine Levi**, **Ssentongo Henry**, **Tusiime Emmanuel**, and **Otim Ronald**.

2. Deployment Automation

- **Ssentongo and Aine** configured Render webhooks to auto-deploy Django backend on successful GitHub builds.
- **Otim Ronald** and **Tusiime** set up Netlify integration for React frontend to auto-deploy on `main` updates.
- Environment variables (API URLs, model paths) were securely added to staging settings.

3. Testing the Staging Environment

- The staging environment was tested for full request flow:
 - Image upload from frontend
 - API call to backend
 - YOLOv5 inference response displayed
- **Ssentongo Henry** verified functionality and documented staging results.

Deliverables

- Live staging deployment of frontend and backend
- Auto-deploy CI/CD pipeline configured
- Updated ClickUp staging-related tasks

Challenges & Future Plans

Django build time was long due to model loading; optimization was scheduled for Week 5. Next, the team will focus on production deployment and monitoring.