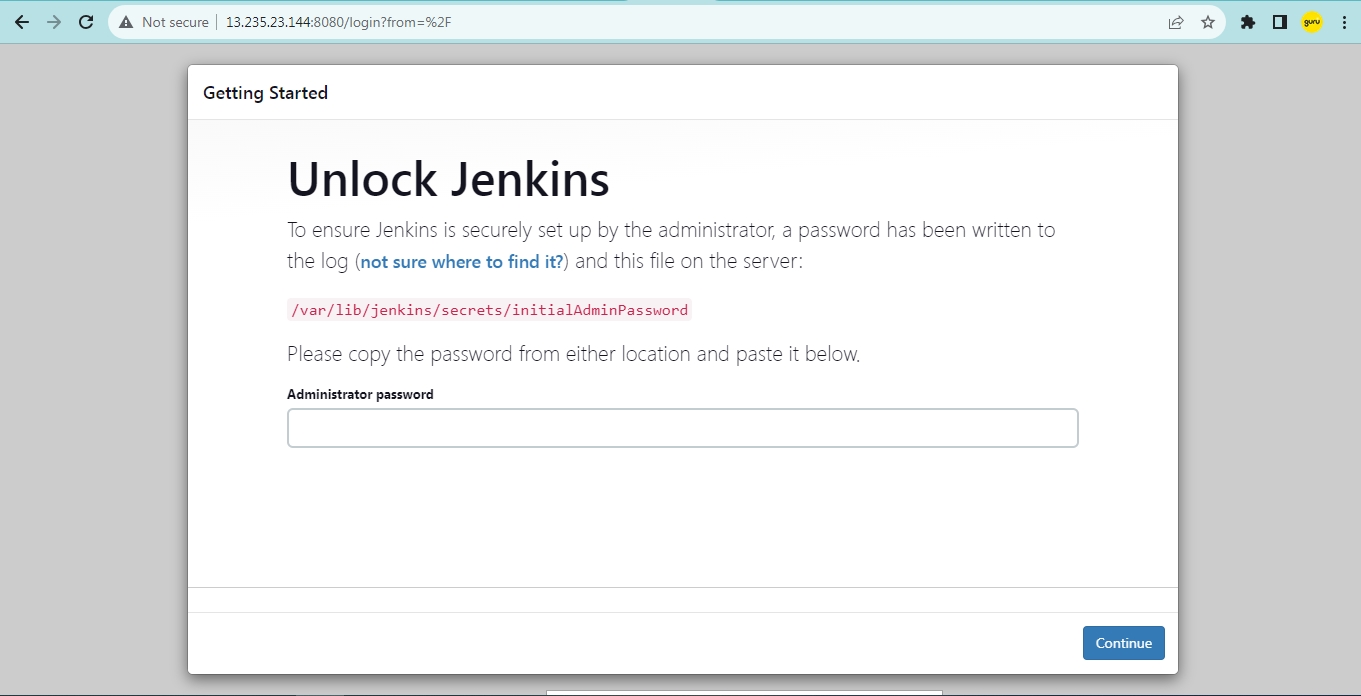
**Weekly task**

Setting up Jenkins for Continuous Integration/Continuous Deployment (CI/CD) involves several steps. Here, We shall go through the process of configuring Jenkins for the described tasks.

Jenkins Installation:

Install Jenkins on your server or local machine. You can download it from Jenkins official website. Install the necessary plugins for Git integration, pipeline support, and any plugins required for your specific application or environment. Ensure your sample application is stored in a version control system. We'll assume Git in this example.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

**Task 1: Automated Testing in CI**

Configure Jenkins Pipeline:

Open Jenkins in your web browser and log in.

Create a new item (Freestyle project or Pipeline) and configure it:

Connect it to your version control system (e.g., Git).

Add build steps to compile and execute unit tests.

**Task 2: Scheduled Builds**

Configure Jenkins Job for Scheduled Builds:

Edit the existing job or create a new one.

In the Build Triggers section, select "Build periodically."

Set the schedule (e.g., 0 0 \* \* \* for daily builds at midnight).

Save the configuration.

**Task 3: Rollback Mechanism in CD**

Configure Jenkins Pipeline for CD:

Extend your existing Jenkinsfile or create a new one for CD.

Implement deployment steps using tools like Docker, Kubernetes, or your preferred deployment method.

Introduce a health check mechanism:

After deployment, add a step to verify the health of the application.

Implement rollback steps:

If health check fails, trigger a rollback process.

Use Jenkins' "Copy Artifact" plugin to retrieve the last stable version.

Save and execute the pipeline manually.

A screenshot of a computer

Description automatically generated