**Jenkins Pipeline with Manual Approval for Production Deployment**

Yes, **it is absolutely possible** in Jenkins to design a pipeline that:

1. **Builds and deploys to staging**
2. **Waits for manual approval**
3. **Deploys to production only after approval**

This is commonly implemented using a **Jenkins Declarative Pipeline** with the input step.

**Example Jenkinsfile with Manual Approval Gate**

pipeline {

agent any

stages {

stage('Build') {

steps {

echo 'Building the app...'

// Add build commands here

}

}

stage('Deploy to Staging') {

steps {

echo 'Deploying to staging...'

// Deploy to staging server

}

}

stage('Approval') {

steps {

script {

input message: 'Approve to deploy to Production?', ok: 'Deploy'

}

}

}

stage('Deploy to Production') {

steps {

echo 'Deploying to production...'

// Deploy to production server

}

}

}

}

**What You Need**

* A working **Jenkinsfile** (as shown above) in your source repo
* Jenkins with the **Declarative Pipeline plugin** (usually pre-installed)
* Proper **user permissions** to approve deployments in the UI
* If using a **Scripted Pipeline**, you can still use input, but it requires more handling

To limit who can approve, use the submitter parameter:

input message: 'Deploy to Production?', ok: 'Proceed', submitter: 'admin,devops-team'

**(OPTIONAL for testing) Tips for Safety**

* Use the **Role-Based Authorization Strategy** plugin to control access
* Alert your team for approvals:
  + Send notifications via **Slack**, **email**, etc.
  + Get approvals via **Jenkins Blue Ocean** or similar dashboards
* Ensure only **team leads** or **authorized personnel** can approve production deployments