



COMSATS University Islamabad

Attock Campus

Program: BS

Fall 2024

TISE

Assignment 3

November 12, 2024

Total Marks: (30)
Time Allowed: (150 minutes)

Assignment No. 3

In this assignment we'll learn how to build a pipeline that automatically checks your website project whenever you update it on GitHub. This will help us catch any issues early on.

What you'll need:

- A GitHub account (free)
- Jenkins installed on your computer (Already installed during last class)

Part 1: Create a Simple Website Project on GitHub

1. **Make a new project on GitHub** called "jenkins-website-ci".
2. **Create these files in your project:**
 - **index.html:** This is your main website page code.
 - **styles.css:** This file styles your website (colors, fonts, etc.).
 - **hello.sh:** This is a simple script that pretends to build your website (we'll add more later). We'll make it run a command later.

Here's what the files should look like (copy and paste if needed):

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">

  <title>My Website</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <h1>Welcome to My Website!</h1>

  <p>This is a simple website project for Jenkins CI/CD
demonstration.</p>
</body>
</html>
```

```

CSS
body {
    font-family: Arial, sans-serif;
    background-color: #f0f0f0;
    color: #333;
}

h1 {
    color: #007bff;
}

Bash
#!/bin/bash
echo "Building Website Project..."
echo "Website build completed successfully!"

```

3. **Make the "hello.sh" script executable:** Open a terminal and navigate to your project folder. Then type:

```

Bash
chmod +x hello.sh

```

4. **Commit and push these files to your GitHub repository.**
5. **Generate a Personal Access Token (PAT) on GitHub:** This lets Jenkins connect to your account. Go to your GitHub settings > Developer settings > Personal access tokens. Click "Generate new token" and give it repo and admin:repo_hook permissions.

Part 2: Connect GitHub to Jenkins

1. **Tell Jenkins about your GitHub PAT:** Go to Jenkins > Manage Jenkins > Manage Credentials. Add a new credential of type "Secret Text" and give it a name like "GitHub PAT". Paste your PAT from step 5 part 1 here.
2. **Set up a Webhook on GitHub:** Go to your GitHub repository settings > Webhooks. Click "Add webhook" and enter these details:
 - **Payload URL:** <http://localhost:8080/github-webhook/> (This is Jenkins' address)
 - **Content type:** application/json
 - **Events:** Select "Push event" only.
 - Click "Add Webhook".

Part 3: Create a CI Pipeline in Jenkins

1. **Open Jenkins on your computer:** <http://localhost:8080> (replace the port if it is not 8080)
2. **Click "New Item" and choose "Pipeline".** Name it "Website-Build-Pipeline".
3. **Under "Build Triggers", check "GitHub hook trigger for GITScm polling".**
4. **Configure the pipeline script:**
 - Select "Pipeline script from SCM" and choose "Git" as the SCM.
 - Enter your GitHub repository URL (<https://github.com/your-username/jenkins-website-ci.git>).
 - Choose your GitHub credentials (the PAT you added earlier).

5. **Create a Jenkinsfile in your GitHub repository:** This file tells Jenkins what to do when it builds your project. Create a new file called "Jenkinsfile" in your project and paste this code:

Groovy

```
pipeline {
    agent any
    stages {
        stage('Clone Repository') {
            steps {
                git url: 'https://github.com/your-username/jenkins-website-
ci.git'
            }
        }
        stage('Build Website') {
            steps {
                sh './hello.sh'
            }
        }
        stage('HTML Validation') {
            steps {
                echo 'Running HTML Validation...'
                sh 'tidy -q -e index.html || echo "HTML issues detected!"'
            }
        }
    }
}
```

Grading Criteria:

- Correct setup of GitHub repository and Jenkins configuration (06 Marks)
- Successful configuration of the Webhook for automatic triggering (06 Marks)
- Implementation of a Jenkins pipeline with at least two build stages (09 Marks)
- Successful triggering and execution of the pipeline on GitHub commits (06 Marks)
- Submission completeness and analysis (03 Marks)