### **✅ What is Poll SCM in Jenkins?**

**Poll SCM** is a Jenkins feature that checks a source control repository (like Git) **at regular intervals** to see if **any new changes (commits)** have been made. If changes are found, the Jenkins job is triggered.

| **Feature** | **Description** |
| --- | --- |
| **Full Name** | Poll Source Code Management (SCM) |
| **Trigger Type** | **Conditional** — Only triggers if there are new changes in the repo |
| **Tool Used** | Cron syntax (e.g., H/5 \* \* \* \*) |
| **Common Use** | Automating builds based on Git commits without constant manual triggering |

### **⚙️ How It Works:**

1. Jenkins runs a **Git status check** at the interval you set.
2. If **new commits are detected**, the job is triggered.
3. If **no changes**, Jenkins **skips the build**.

### **✍️ Your Notes (Explained):**

From your handwritten notes:

#### **- Poll SCM:**

"By given time, Jenkins job will trigger **with condition**."

Correct: **Unlike Build Periodically**, Poll SCM **only triggers** when a change is detected.

#### **- Condition:**

Before triggering the job, Jenkins **verifies Git status**.

* **If commit found** → Jenkins triggers the build.
* **If no commit** → Jenkins does nothing.

#### **- Example:**

"At 45th minute, my Jenkins job tries to trigger but no commit"  
 "I commit at 46th min — Jenkins needs to wait till 47th min"

Correct: Jenkins **polls at fixed times**. If you commit between two polls, Jenkins **waits until next poll**.

### **🧠 Scenario:**

| **Time** | **Action** | **Result** |
| --- | --- | --- |
| 12:00 | Jenkins polls, **no changes** | Build not triggered |
| 12:01 | Developer pushes commit | Nothing happens immediately |
| 12:02 | Jenkins polls again, sees commit | Build triggered |

### **✅ Solution You Mentioned: GitHub Hooks**

Correct: Instead of Poll SCM, use **Webhooks** for instant response.

| **Trigger Type** | **Description** |
| --- | --- |
| **Poll SCM** | Jenkins checks periodically |
| **GitHub Webhook** | GitHub tells Jenkins instantly |

🛠️ Poll SCM Syntax Example:

H/5 \* \* \* \*

Runs every 5 minutes and checks for new commits.

🔄 Comparison

| **Feature** | **Poll SCM** | **GitHub Webhook** |
| --- | --- | --- |
| Trigger | Based on schedule | Triggered by GitHub event (push/PR) |
| Response time | May be delayed until next check | Instant |
| Server load | Higher (polling frequently) | Low |
| Internet access | Jenkins pulls | GitHub pushes to Jenkins (must be reachable) |

### **✅ When to Use Poll SCM?**

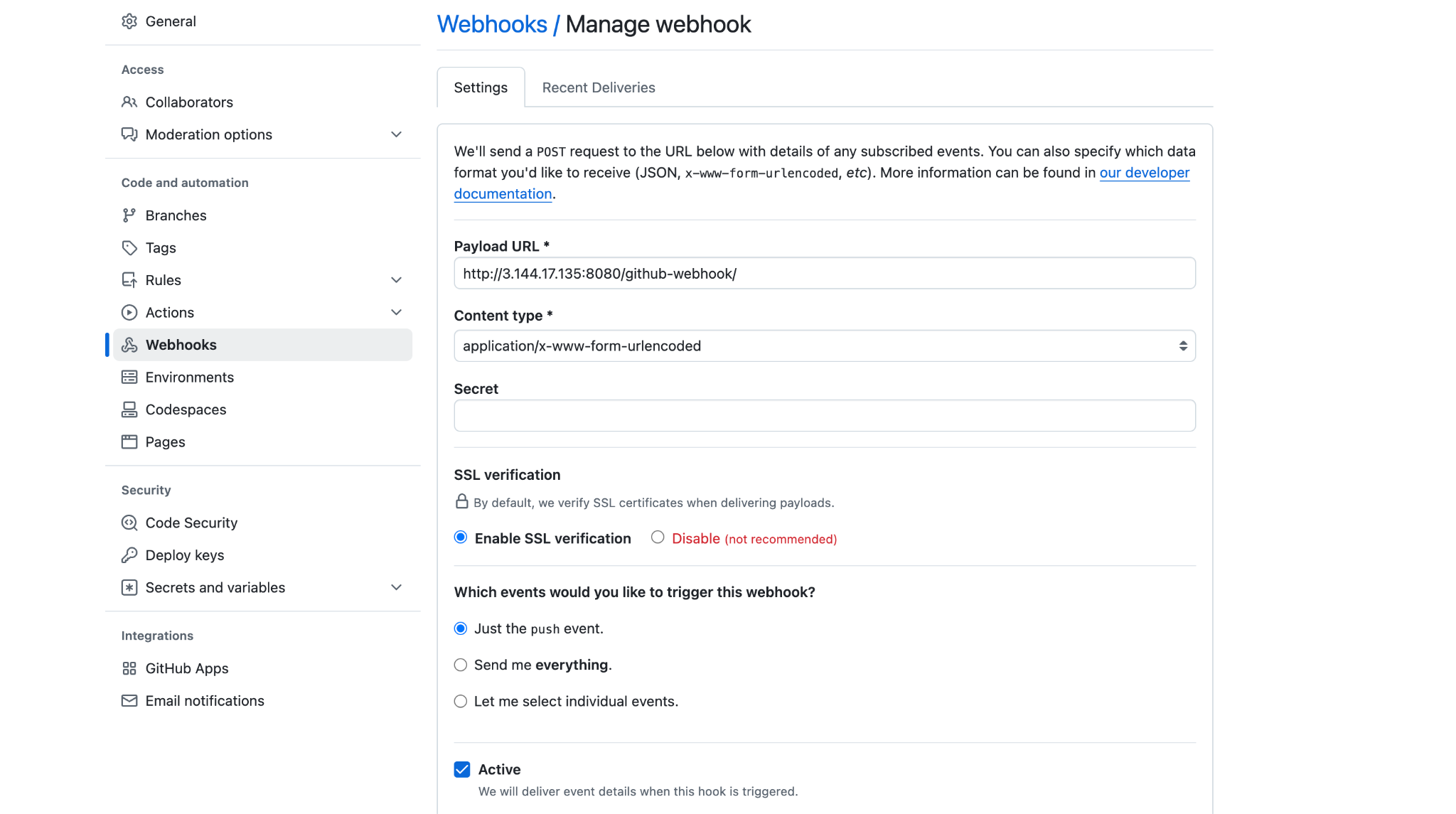
* Jenkins **cannot receive incoming webhooks** (firewall, internal server).
* You need **scheduled checks** instead of instant build.
* You prefer **controlled load** instead of real-time triggers.

## **✅ Step-by-Step Jenkins + GitHub + Poll SCM Integration**

### **🔧 Step 1: Create GitHub Webhook**

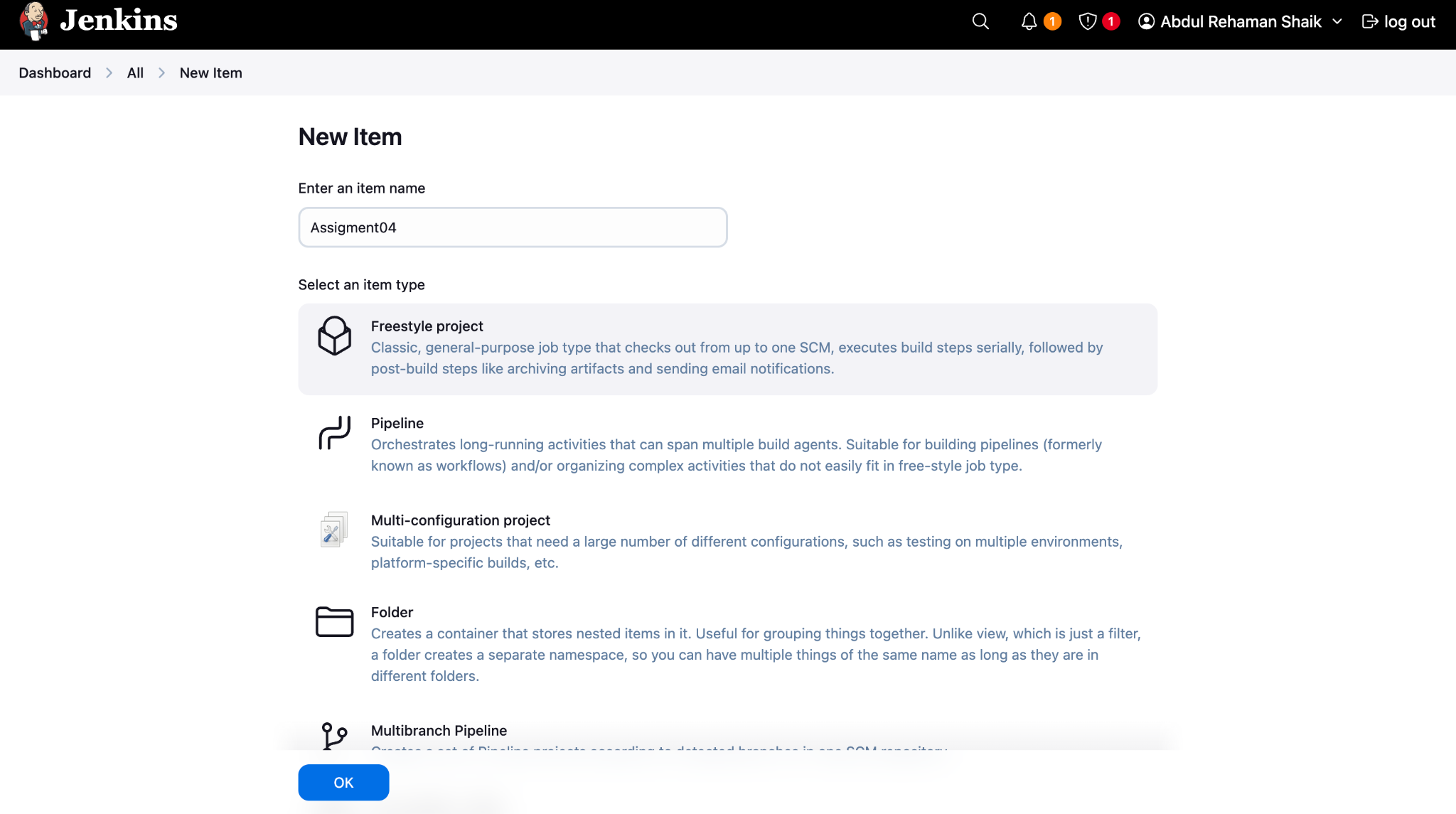
* **Navigate to**: GitHub repo → Settings → Webhooks
* **Payload URL**: http://<your-public-ip>:8080/github-webhook/
* **Content type**: application/x-www-form-urlencoded
* **Events**: Choose Just the push event
* **Active**: Yes ✅

🔥 This is useful if you later switch to using GitHub webhook instead of polling. For now, you're using Poll SCM, so webhook is optional.



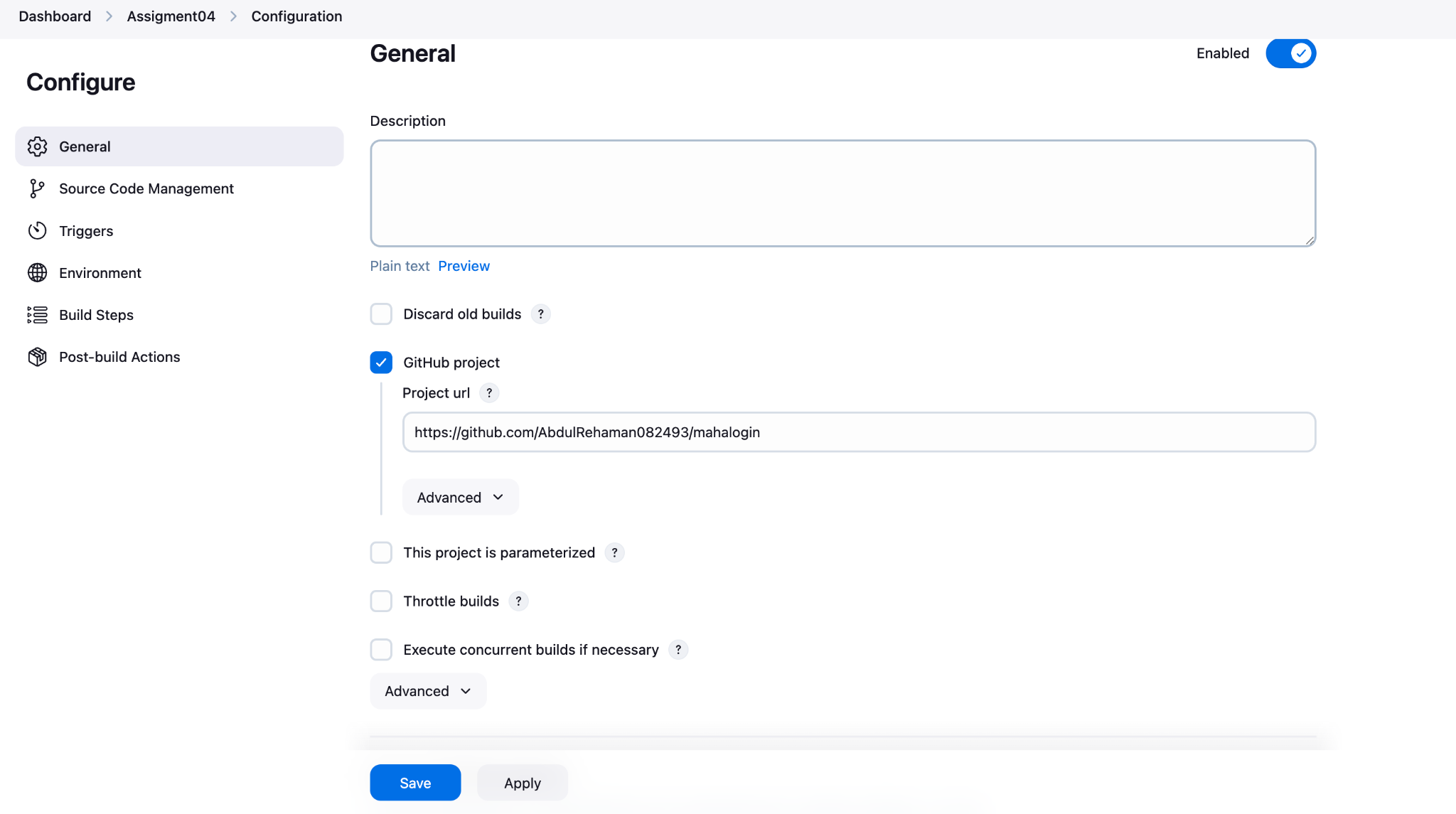
### **🏗️ Step 2: Create Jenkins Job**

* **New Item** → Name: Assignment04
* Choose: **Freestyle Project**



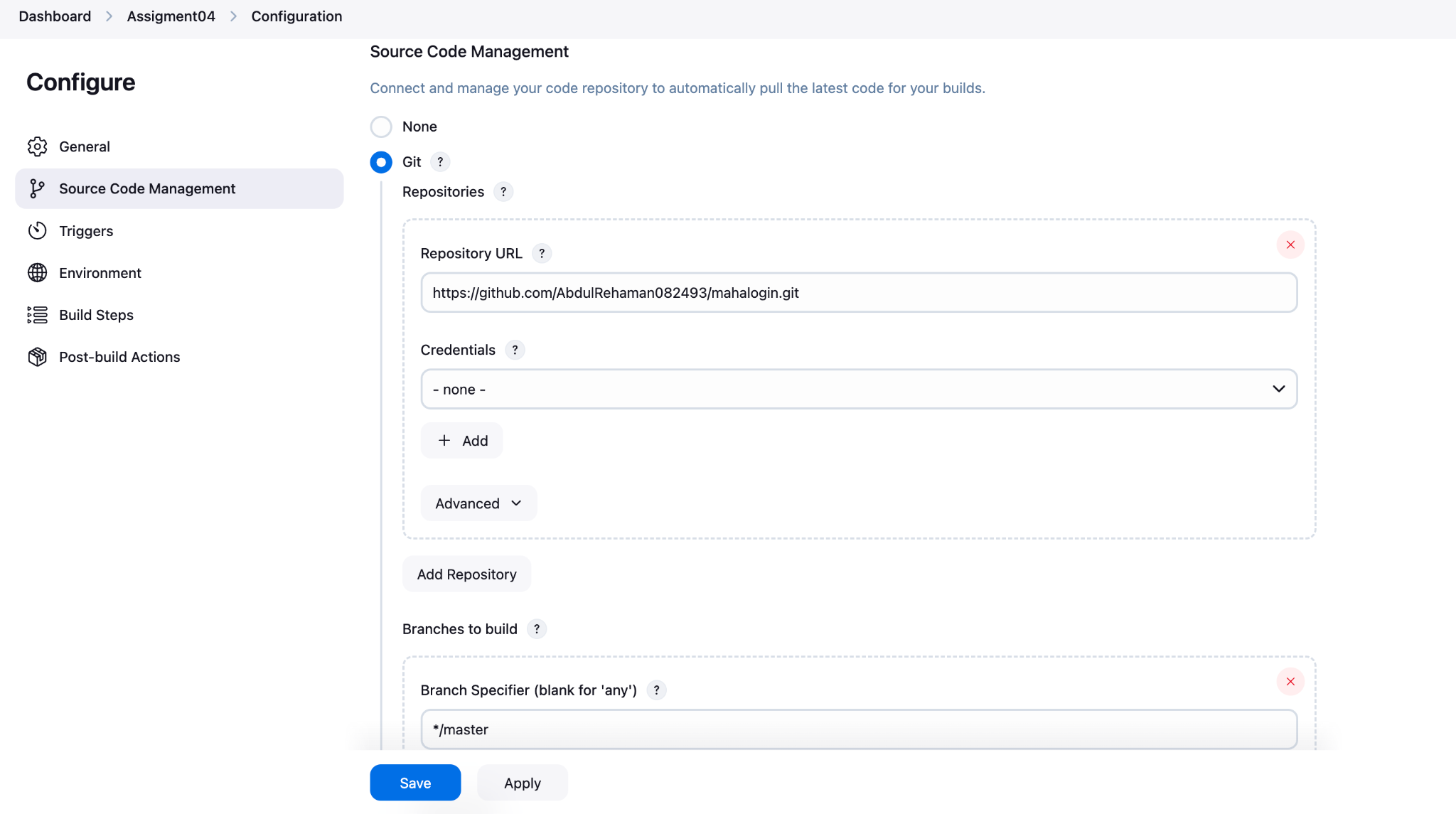
### **⚙️ Step 3: General Configuration**

* ✅ Check **GitHub project**
* Project URL: https://github.com/AbdulRehaman082493/mahalogin



### **🔄 Step 4: Source Code Management (Git)**

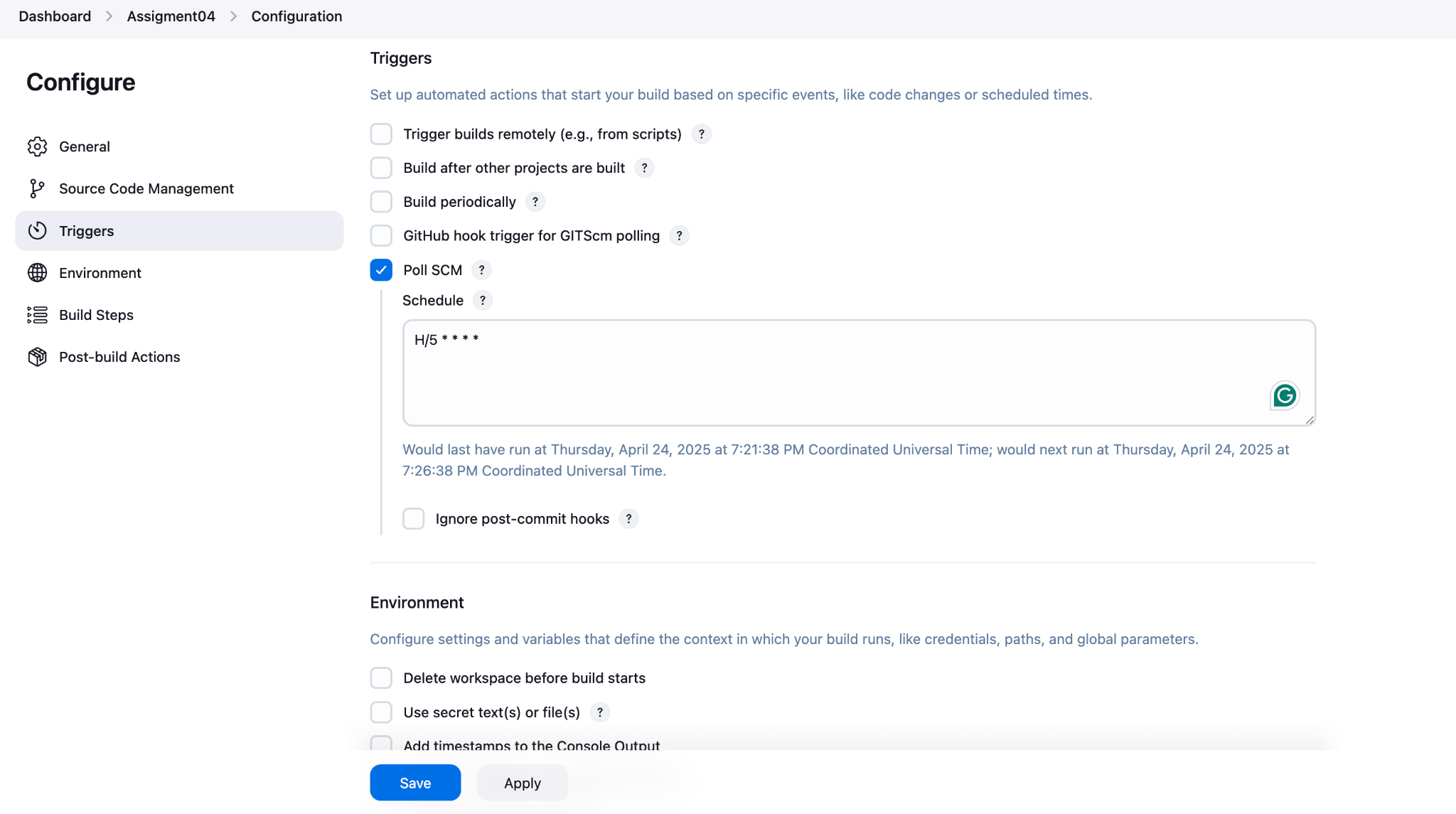
* Repository URL: https://github.com/AbdulRehaman082493/mahalogin.git
* Branches to build: \*/master
* ❗ Credentials: Currently set to - none -  
  + If your repo is **private**, this will cause issues. You’ll need to **add GitHub credentials** under "Add" → Jenkins credentials.



### **⏰ Step 5: Configure Triggers**

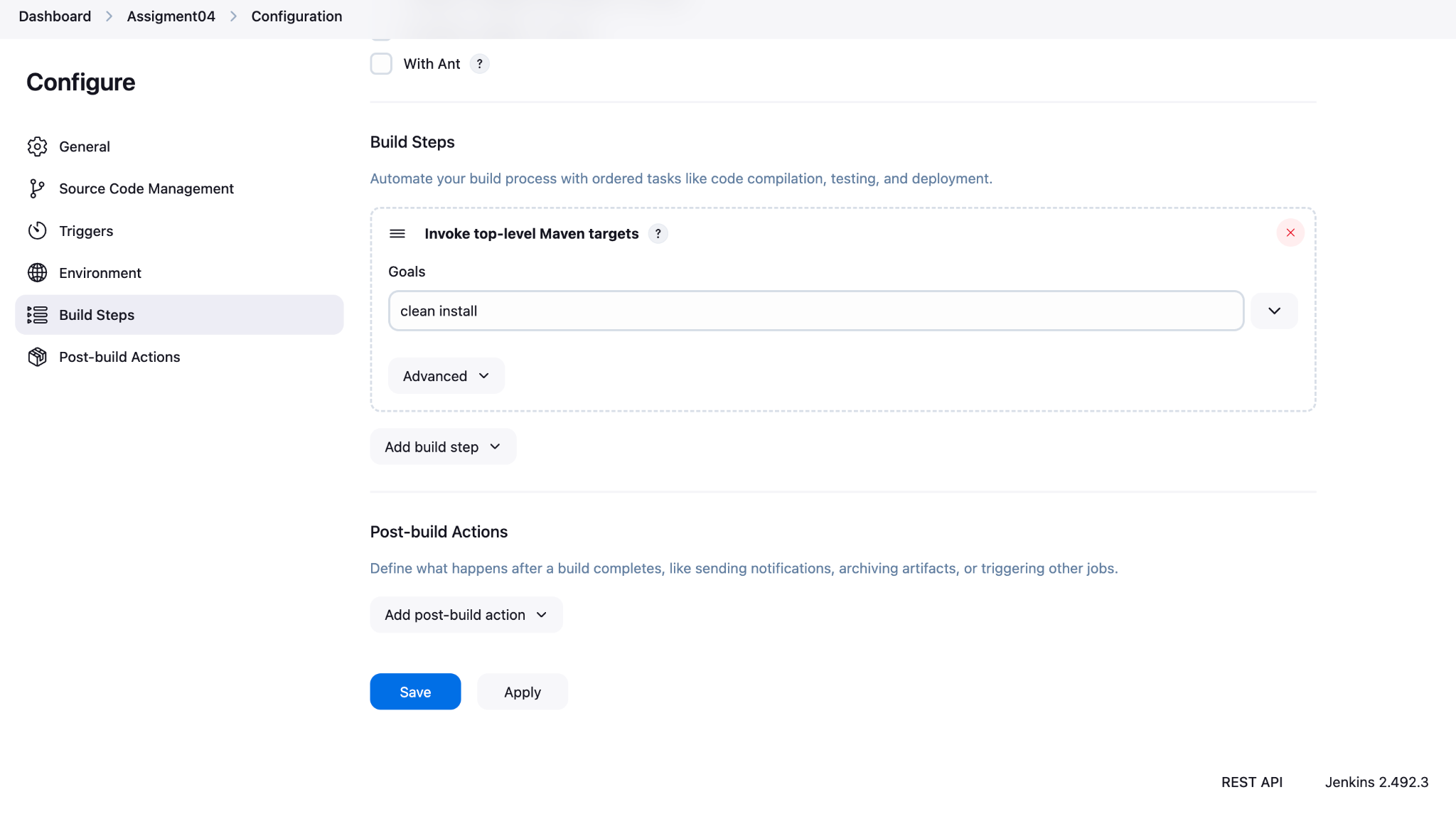
* ✅ Enable **Poll SCM**
* Schedule: H/5 \* \* \* \*  
  + Jenkins will **poll every 5 minutes** for Git changes.
  + **If it finds new commits → it triggers the build**
  + **If not → build is skipped**

🧠 This is **not instant** like a webhook, but useful when Jenkins is behind a firewall or webhook is not reliable.



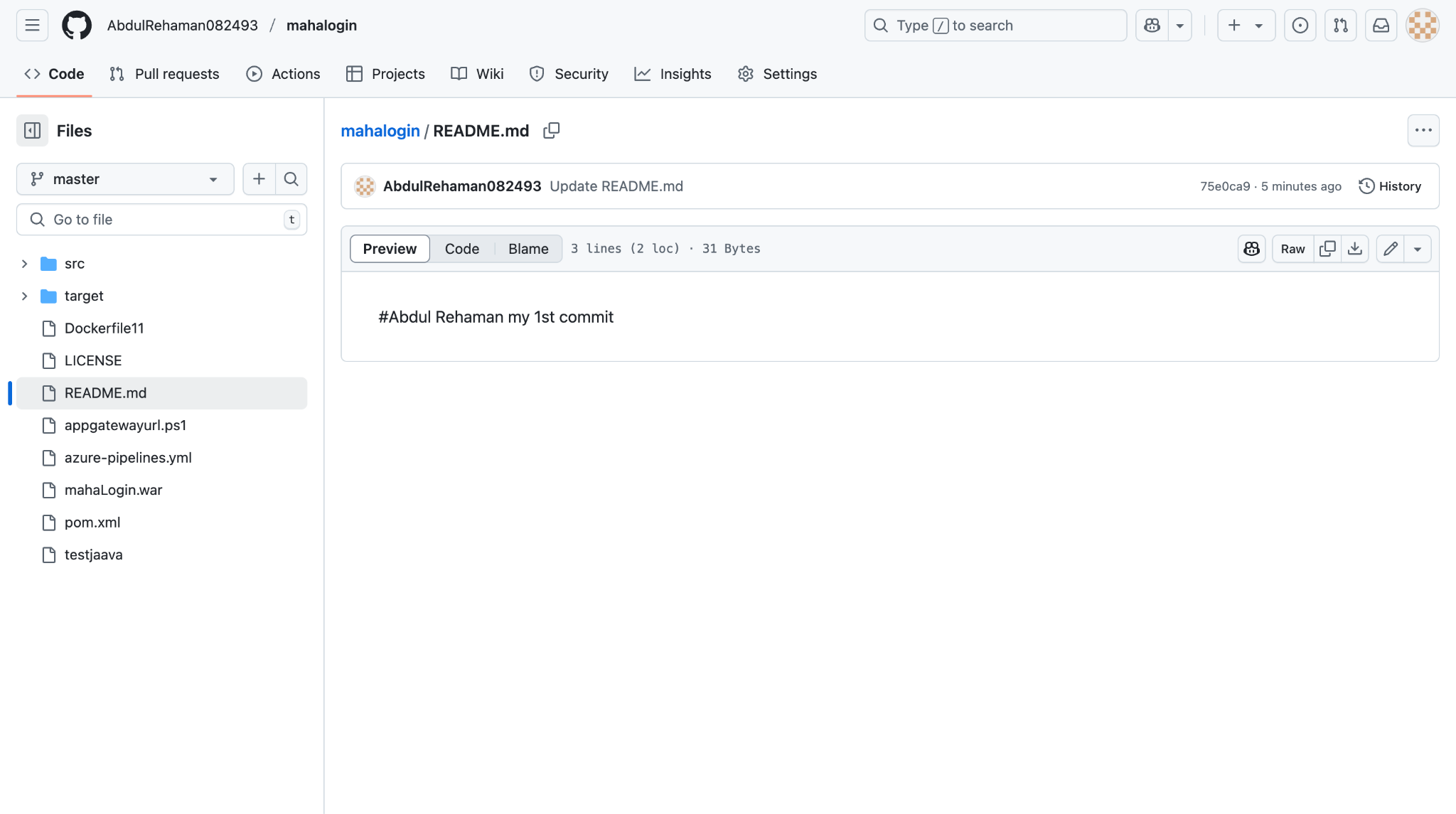
### **🔨 Step 6: Build Step - Maven**

* Goal: clean install
* Jenkins runs mvn clean install to compile, test, and build the .war file from your code.



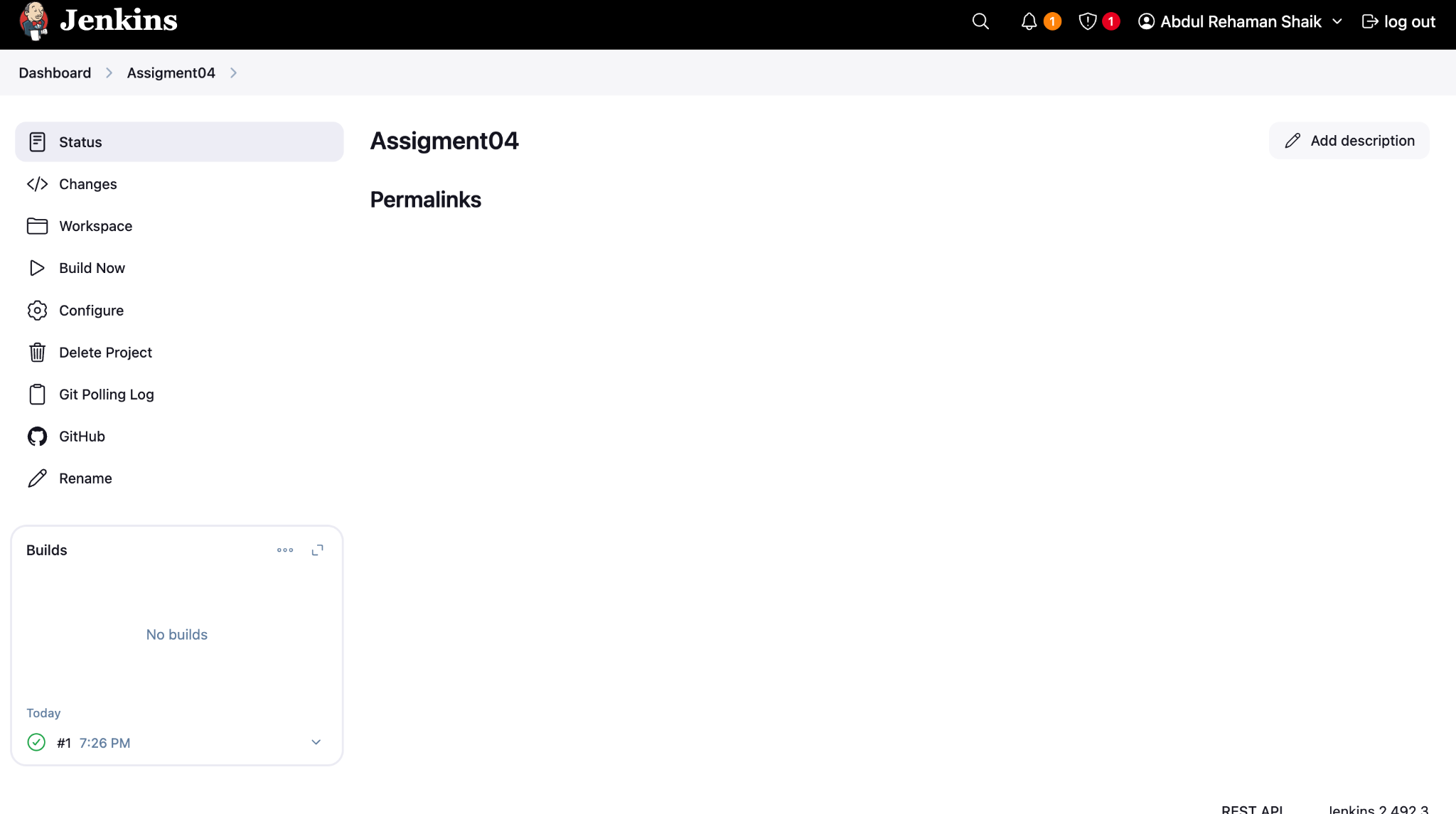
### **📦 Step 7: GitHub Commit Made**

* You pushed this message: Abdul Rehaman my 1st commit
* Jenkins polls GitHub  
  + If commit is new → ✅ build triggers
  + If no change → ❌ no action



### **🧪 Step 8: Check Jenkins Status**

* Go to Dashboard → Assignment04



* ✅ First build triggered at 7:26 PM (as per screenshot)

🔁 Poll SCM vs Webhook (Quick Recap)

| **Feature** | **Poll SCM** | **GitHub Webhook** |
| --- | --- | --- |
| Trigger | Schedule-based (cron) | Event-based (push, PR) |
| Speed | May delay until next poll | Instant trigger |
| Jenkins Server | Can be private/internal | Must be public (receives HTTP) |
| Server Load | Slightly higher (polling) | Lightweight |

### **✅ Tips to Improve Setup**

1. **Switch to webhook** for faster builds if your Jenkins server is public.
2. **Add credentials** in Jenkins Git config if GitHub repo is private.
3. Use **post-build actions** (e.g., archive .war, deploy, email notification).