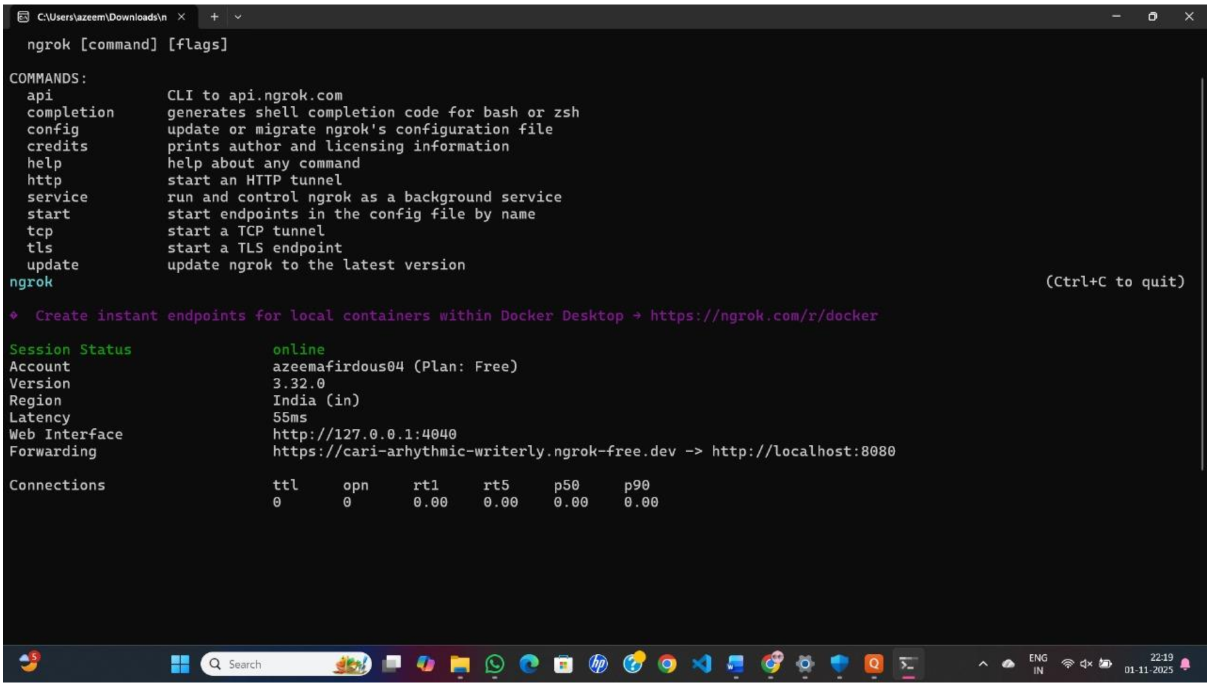
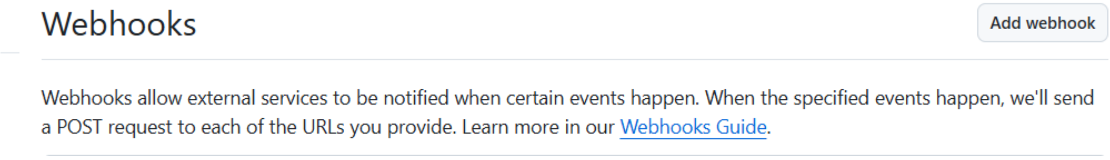
WEEK 11

NGROK:



```
  ngrok [command] [flags]

COMMANDS:
  api           CLI to api.ngrok.com
  completion    generates shell completion code for bash or zsh
  config        update or migrate ngrok's configuration file
  credits       prints author and licensing information
  help          help about any command
  http          start an HTTP tunnel
  service       run and control ngrok as a background service
  start         start endpoints in the config file by name
  tcp           start a TCP tunnel
  tls           start a TLS endpoint
  update        update ngrok to the latest version
ngrok                                                                                          (Ctrl+C to quit)

* Create instant endpoints for local containers within Docker Desktop → https://ngrok.com/r/docker

Session Status                online
Account                       azeemafirdous04 (Plan: Free)
Version                       3.32.0
Region                        India (in)
Latency                       55ms
Web Interface                 http://127.0.0.1:4040
Forwarding                    https://cari-arhythmic-writerly.ngrok-free.dev -> http://localhost:8080

Connections                   ttl     opn     rt1     rt5     p50     p90
                              0       0       0.00    0.00    0.00    0.00
```

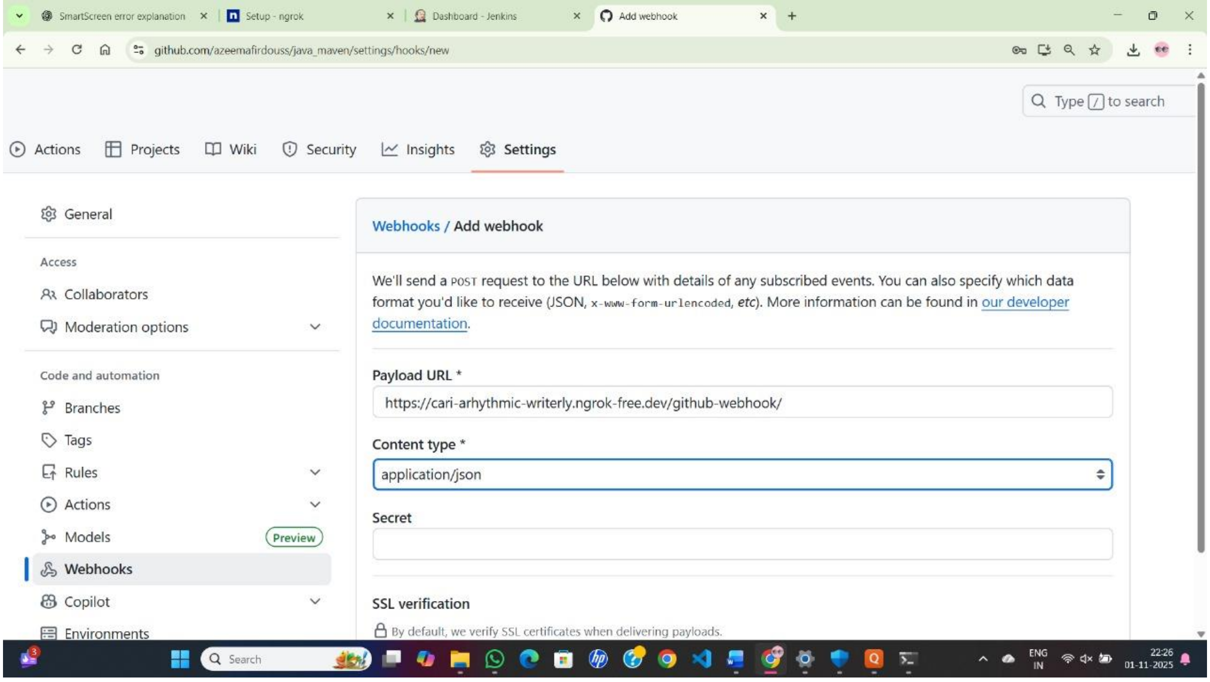
Setting Up Jenkins CI------using GitHub Webhook with Jenkins



Webhooks

Add webhook

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our Webhooks Guide.



Webhooks / Add webhook

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, *etc*). More information can be found in our developer documentation.

Payload URL *

https://cari-arhythmic-writerly.ngrok-free.dev/github-webhook/

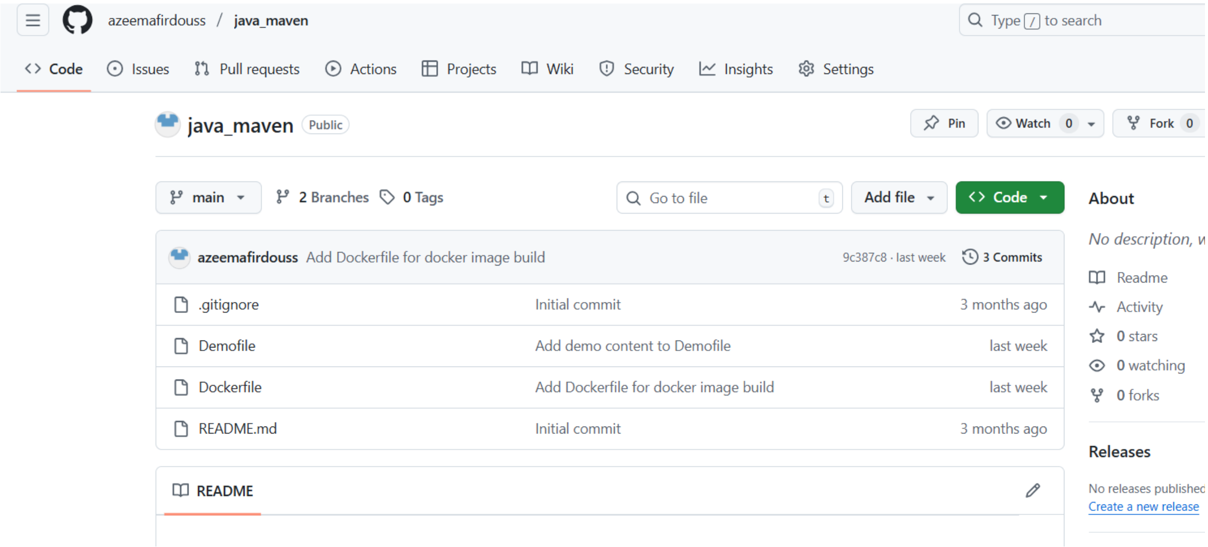
Content type *

application/json

Secret

SSL verification

By default, we verify SSL certificates when delivering payloads.

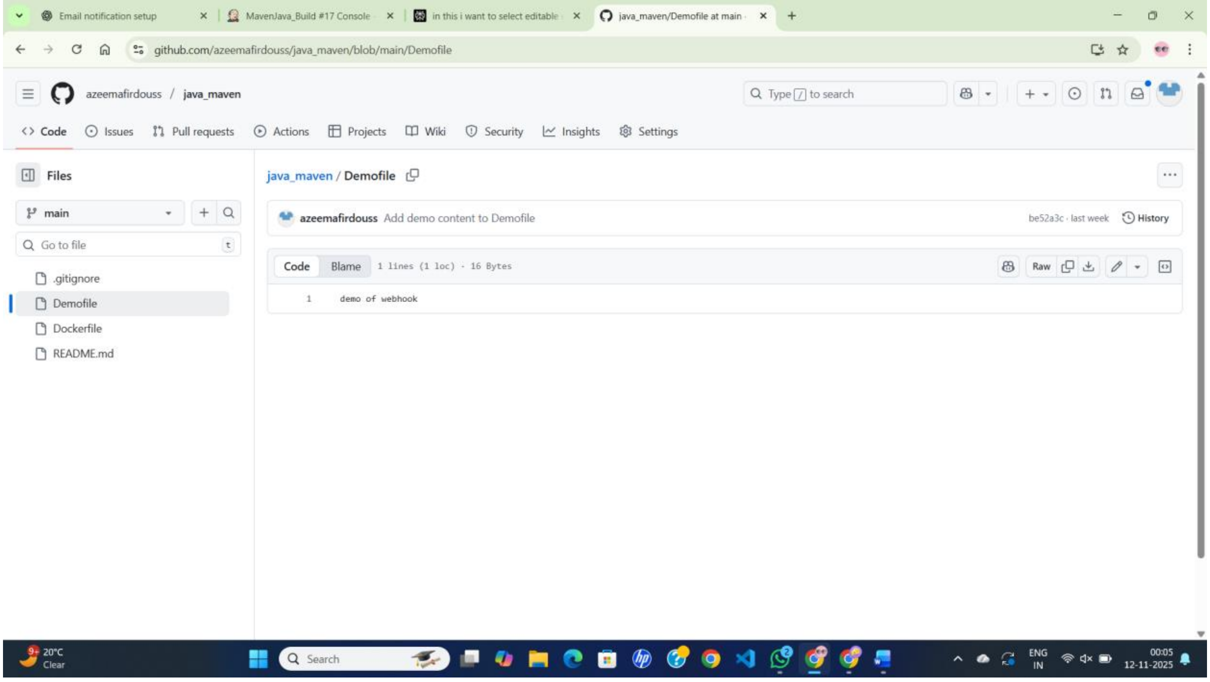


java_maven Public

azeemafirdouss Add Dockerfile for docker image build 9c387c8 · last week 3 Commits

.gitignore	Initial commit	3 months ago
Demofile	Add demo content to Demofile	last week
Dockerfile	Add Dockerfile for docker image build	last week
README.md	Initial commit	3 months ago

No description, w
0 stars
0 watching
0 forks
No releases publishec



github.com/azeemafirdouss/java_maven/blob/main/Demofile

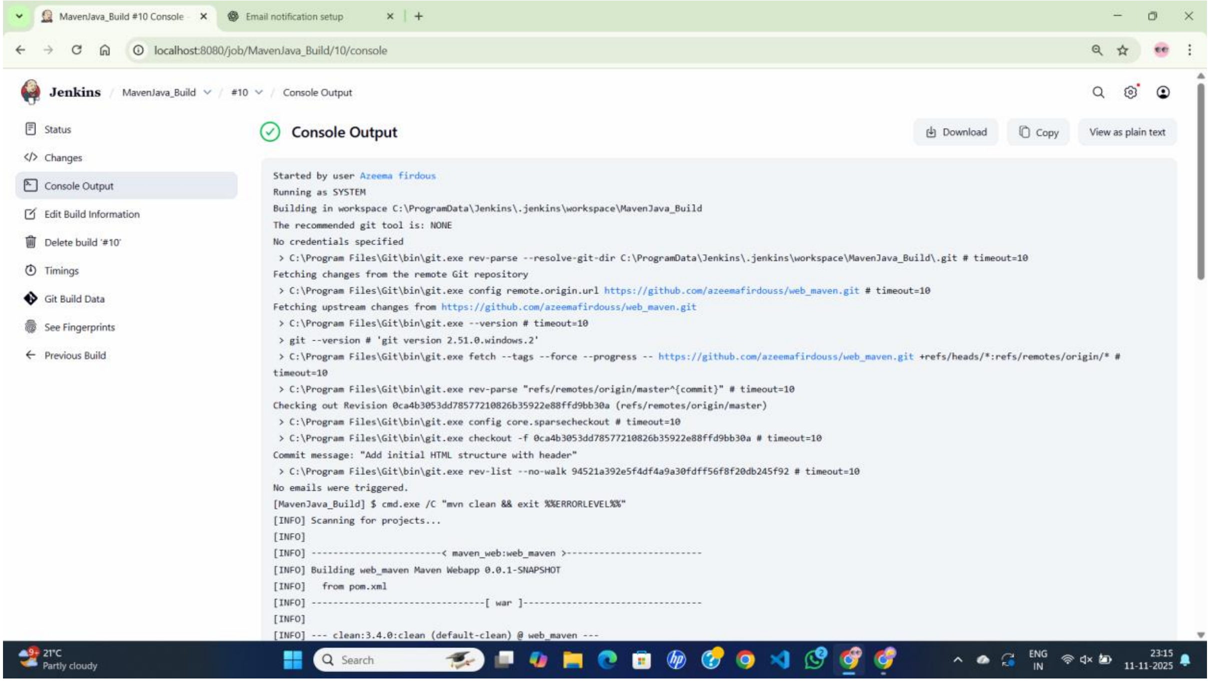
java_maven / Demofile

azeemafirdouss Add demo content to Demofile be52a3c · last week

1 lines (1 loc) · 16 Bytes

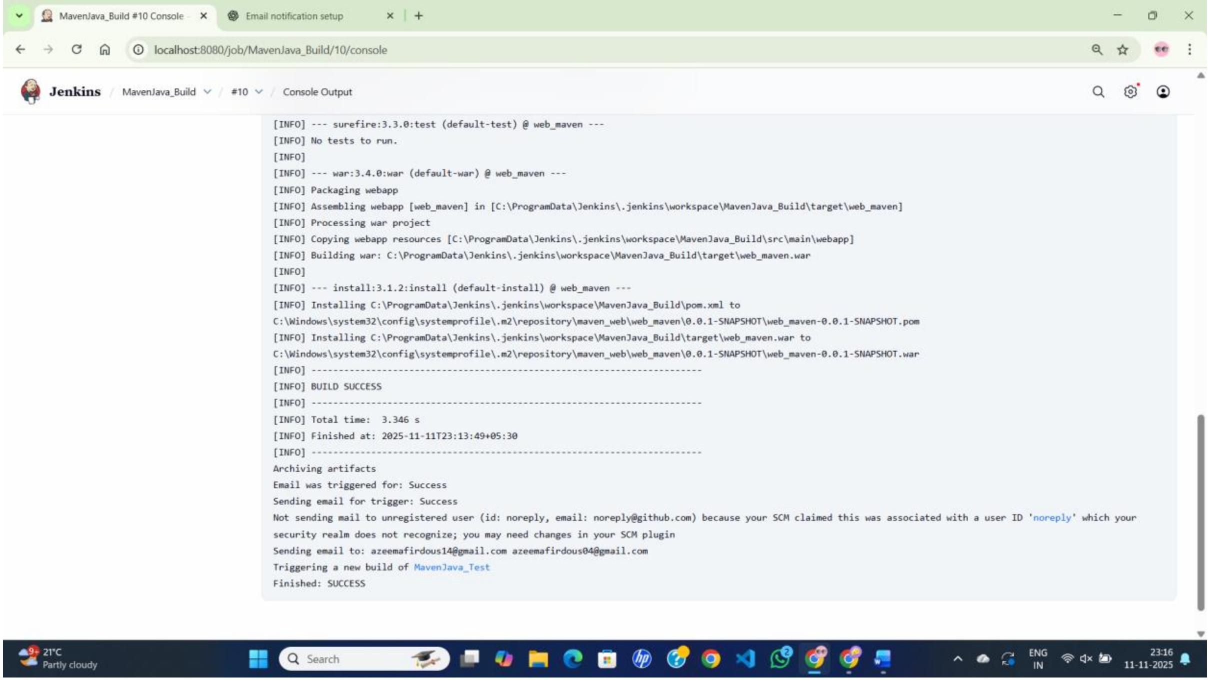
```
demo of webhook
```

EMAIL NOTIFICATION : BUILD SUCCESSFULLY



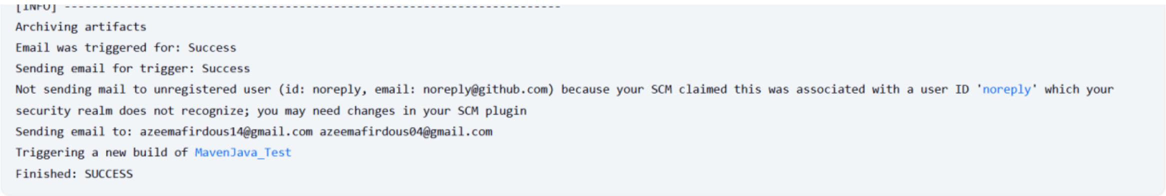
Console Output

```
Started by user Azeema firdous
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\MavenJava_Build
The recommended git tool is: NONE
No credentials specified
 > C:\Program Files\Git\bin\git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\.jenkins\workspace\MavenJava_Build\.git # timeout=10
Fetching changes from the remote Git repository
 > C:\Program Files\Git\bin\git.exe config remote.origin.url https://github.com/azeemafirdouss/web_maven.git # timeout=10
Fetching upstream changes from https://github.com/azeemafirdouss/web_maven.git
 > C:\Program Files\Git\bin\git.exe --version # timeout=10
 > git --version # 'git version 2.51.0.windows.2'
 > C:\Program Files\Git\bin\git.exe fetch --tags --force --progress -- https://github.com/azeemafirdouss/web_maven.git +refs/heads/*:refs/remotes/origin/* #
timeout=10
 > C:\Program Files\Git\bin\git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
Checking out Revision 0ca4b3053dd78577210826b35922e88ffd9bb30a (refs/remotes/origin/master)
 > C:\Program Files\Git\bin\git.exe config core.sparsecheckout # timeout=10
 > C:\Program Files\Git\bin\git.exe checkout -f 0ca4b3053dd78577210826b35922e88ffd9bb30a # timeout=10
Commit message: "Add initial HTML structure with header"
 > C:\Program Files\Git\bin\git.exe rev-list --no-walk 94521a392e5f4df4a9a30fdff56f8f20db245f92 # timeout=10
No emails were triggered.
[MavenJava_Build] $ cmd.exe /C "mvn clean && exit %%ERRORLEVEL%%"
[INFO] Scanning for projects...
[INFO]
[INFO] -----------------------< maven_web:web_maven >------------------------
[INFO] Building web_maven Maven Webapp 0.0.1-SNAPSHOT
[INFO]   from pom.xml
[INFO] --------------------------------[ war ]--------------------------------
[INFO]
[INFO] --- clean:3.4.0:clean (default-clean) @ web_maven ---
```

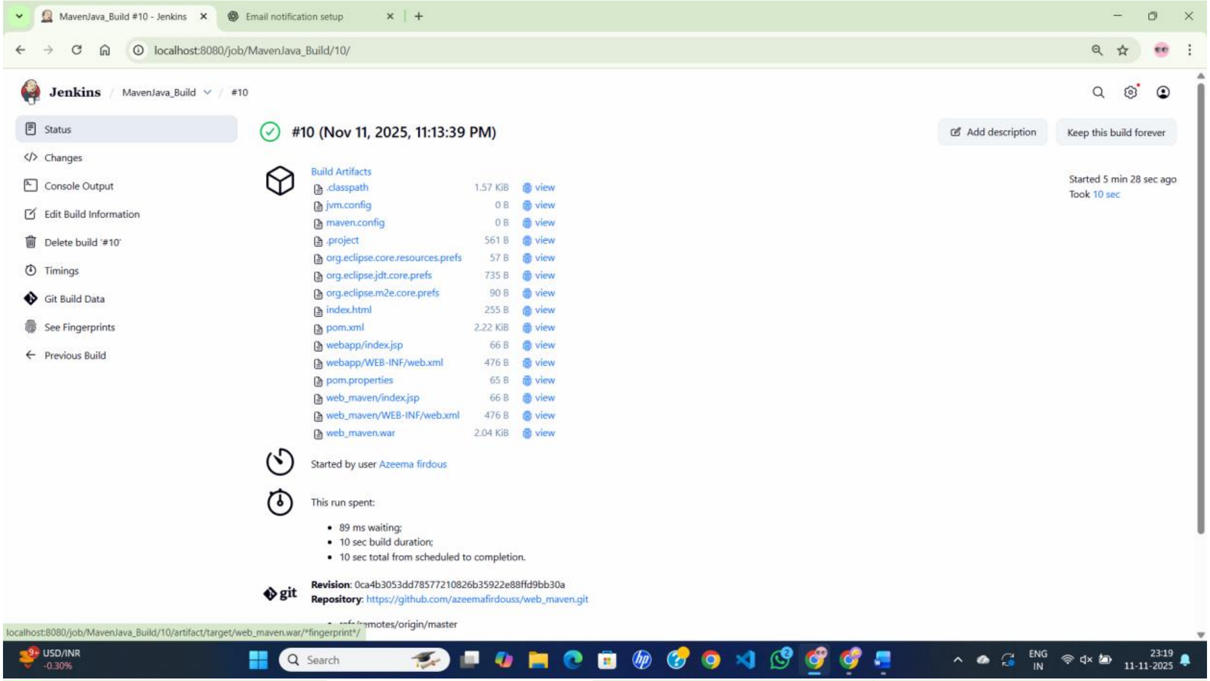


```
[INFO] --- surefire:3.3.0:test (default-test) @ web_maven ---
[INFO] No tests to run.
[INFO]
[INFO] --- war:3.4.0:war (default-war) @ web_maven ---
[INFO] Packaging webapp
[INFO] Assembling webapp [web_maven] in [C:\ProgramData\Jenkins\.jenkins\workspace\MavenJava_Build\target\web_maven]
[INFO] Processing war project
[INFO] Copying webapp resources [C:\ProgramData\Jenkins\.jenkins\workspace\MavenJava_Build\src\main\webapp]
[INFO] Building war: C:\ProgramData\Jenkins\.jenkins\workspace\MavenJava_Build\target\web_maven.war
[INFO]
[INFO] --- install:3.1.2:install (default-install) @ web_maven ---
[INFO] Installing C:\ProgramData\Jenkins\.jenkins\workspace\MavenJava_Build\pom.xml to
C:\Windows\system32\config\systemprofile\.m2\repository\maven_web\web_maven\0.0.1-SNAPSHOT\web_maven-0.0.1-SNAPSHOT.pom
[INFO] Installing C:\ProgramData\Jenkins\.jenkins\workspace\MavenJava_Build\target\web_maven.war to
C:\Windows\system32\config\systemprofile\.m2\repository\maven_web\web_maven\0.0.1-SNAPSHOT\web_maven-0.0.1-SNAPSHOT.war
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  3.346 s
[INFO] Finished at: 2025-11-11T23:13:49+05:30
[INFO] ------------------------------------------------------------------------
Archiving artifacts
Email was triggered for: Success
Sending email for trigger: Success
Not sending mail to unregistered user (id: noreply, email: noreply@github.com) because your SCM claimed this was associated with a user ID 'noreply' which your
security realm does not recognize; you may need changes in your SCM plugin
Sending email to: azeemafirdous14@gmail.com azeemafirdous04@gmail.com
Triggering a new build of MavenJava_Test
Finished: SUCCESS
```

Email notification success



```
[INFO] ------------------------------------------------------------------------
Archiving artifacts
Email was triggered for: Success
Sending email for trigger: Success
Not sending mail to unregistered user (id: noreply, email: noreply@github.com) because your SCM claimed this was associated with a user ID 'noreply' which your
security realm does not recognize; you may need changes in your SCM plugin
Sending email to: azeemafirdous14@gmail.com azeemafirdous04@gmail.com
Triggering a new build of MavenJava_Test
Finished: SUCCESS
```



Jenkins / MavenJava_Build / #10

- Status
- Changes
- Console Output
- Edit Build Information
- Delete build '#10'
- Timings
- Git Build Data
- See Fingerprints
- Previous Build

#10 (Nov 11, 2025, 11:13:39 PM)

Started 5 min 28 sec ago
Took 10 sec

Build Artifacts

.classpath	1.57 KiB	view
jvm.config	0 B	view
maven.config	0 B	view
.project	561 B	view
org.eclipse.core.resources.prefs	57 B	view
org.eclipse.jdt.core.prefs	735 B	view
org.eclipse.m2e.core.prefs	90 B	view
index.html	255 B	view
pom.xml	2.22 KiB	view
webapp/index.jsp	66 B	view
webapp/WEB-INF/web.xml	476 B	view
pom.properties	65 B	view
web_maven/index.jsp	66 B	view
web_maven/WEB-INF/web.xml	476 B	view
web_maven.war	2.04 KiB	view

Started by user Azeema firdous

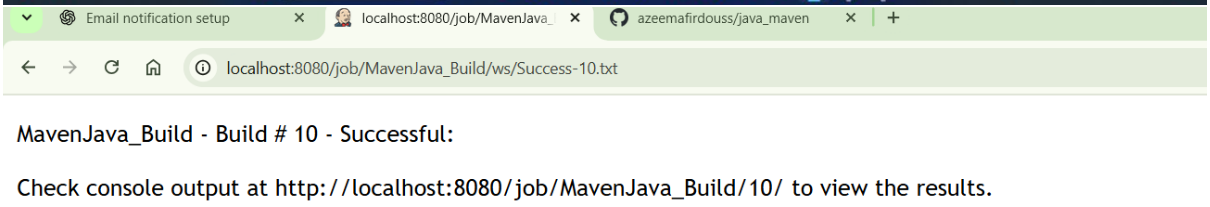
This run spent:

- 89 ms waiting;
- 10 sec build duration;
- 10 sec total from scheduled to completion.

Revision: 0ca4b3053dd78577210826b35922e88ffd9bb30a
Repository: https://github.com/azeemafirdouss/web_maven.git

- refs/remotes/origin/master

localhost:8080/job/MavenJava_Build/10/artifact/target/web_maven.war/*fingerprint*/

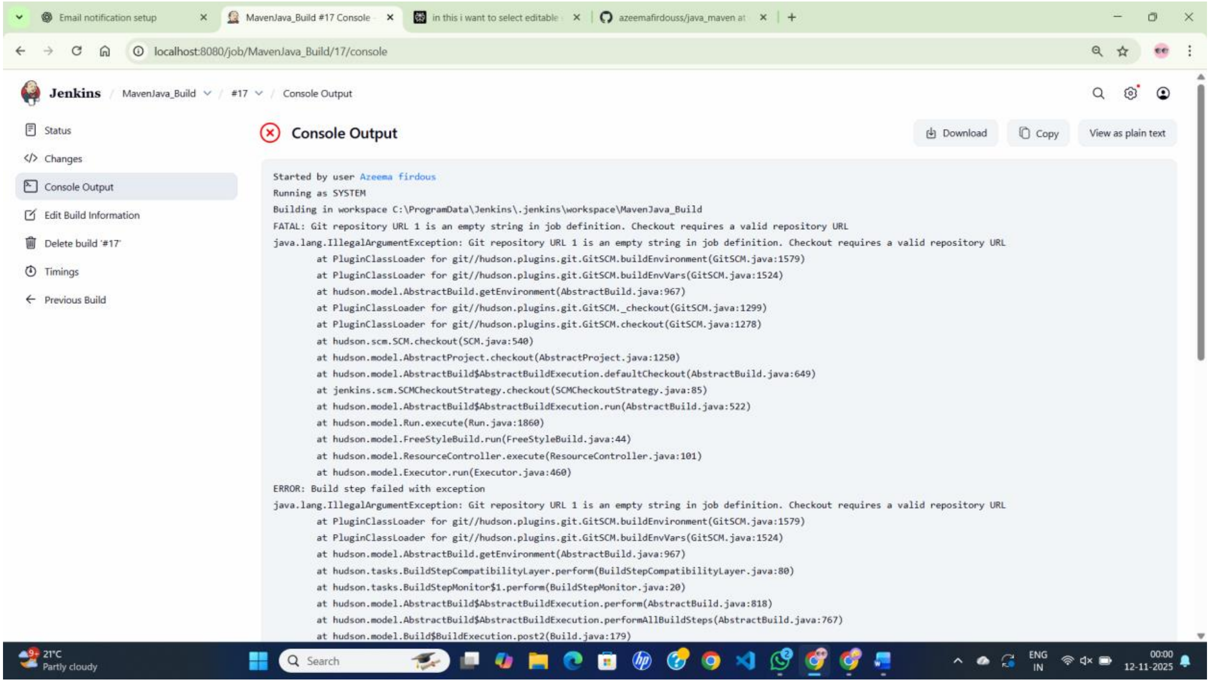


localhost:8080/job/MavenJava_Build/ws/Success-10.txt

MavenJava_Build - Build # 10 - Successful:

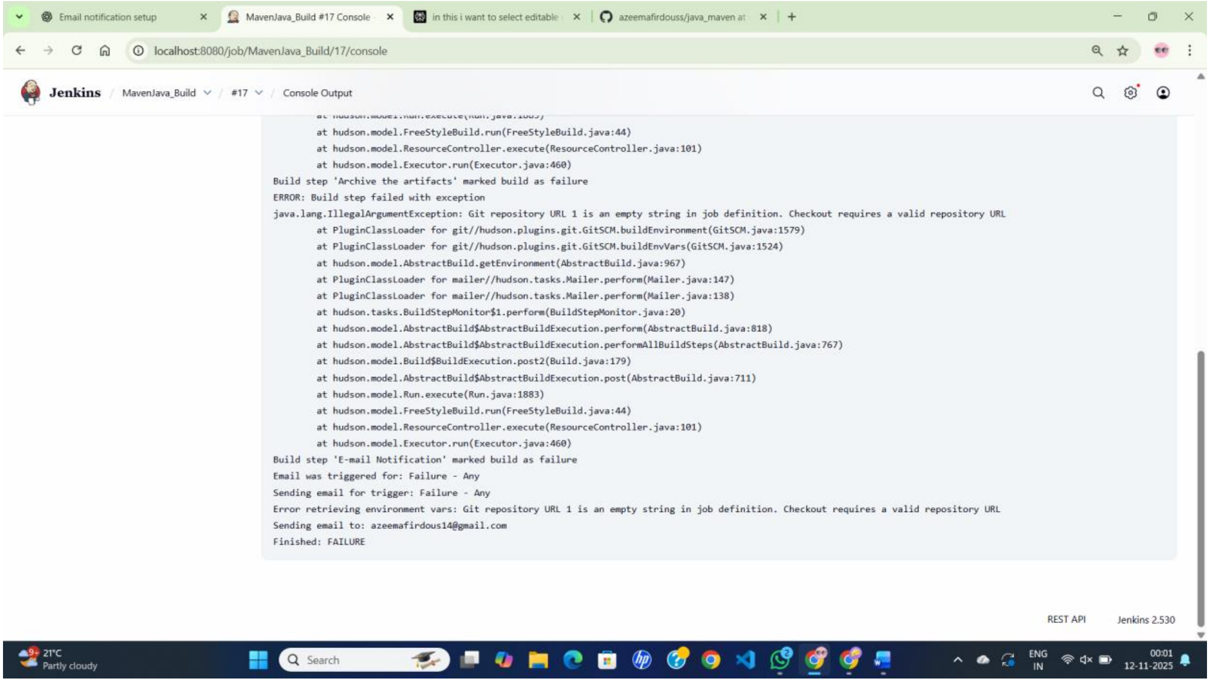
Check console output at http://localhost:8080/job/MavenJava_Build/10/ to view the results.

EMAIL NOTIFICATION BUILD FAILED



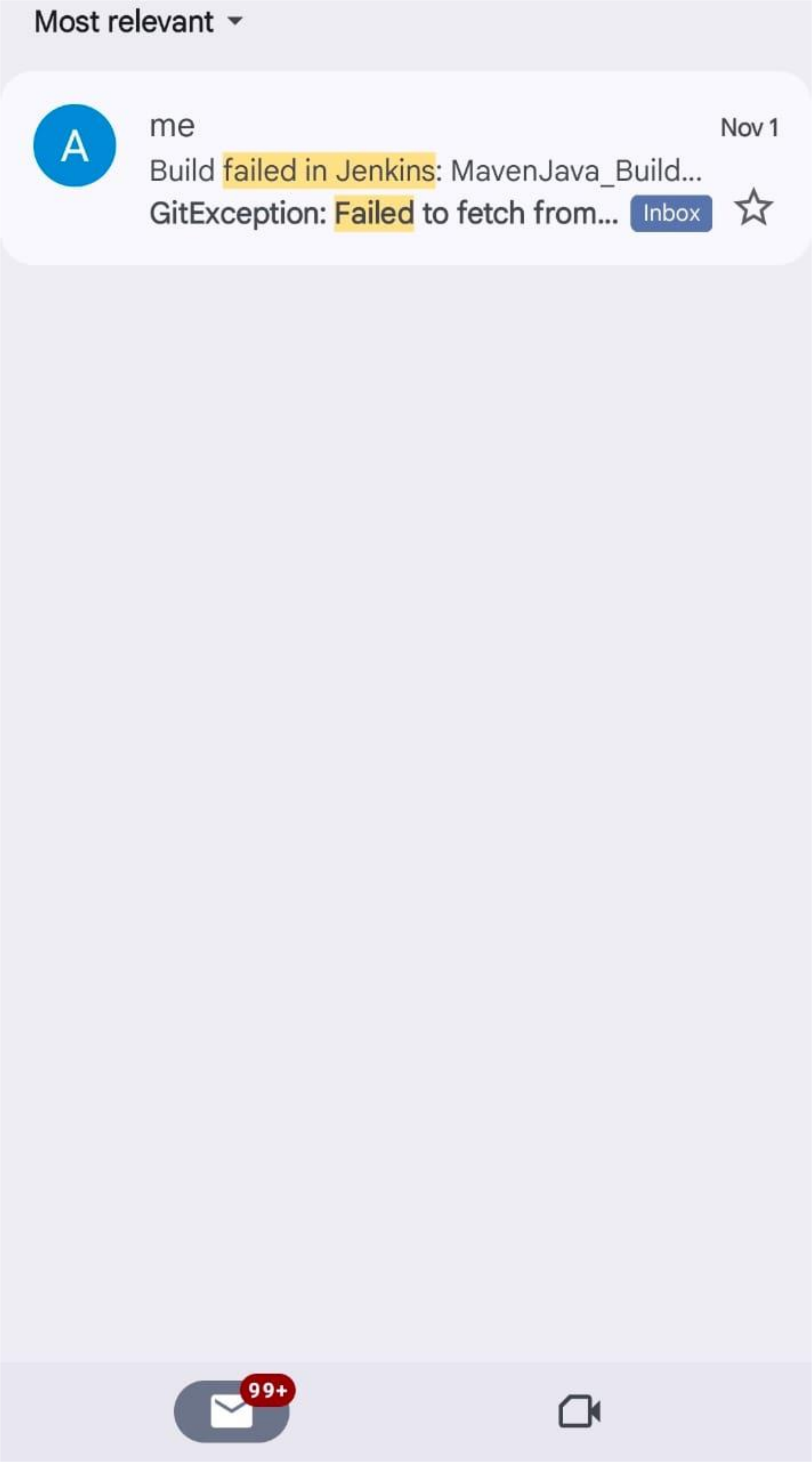
Console Output

```
Started by user Azeema firdous
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\MavenJava_Build
FATAL: Git repository URL 1 is an empty string in job definition. Checkout requires a valid repository URL
java.lang.IllegalArgumentException: Git repository URL 1 is an empty string in job definition. Checkout requires a valid repository URL
	at PluginClassLoader for git//hudson.plugins.git.GitSCM.buildEnvironment(GitSCM.java:1579)
	at PluginClassLoader for git//hudson.plugins.git.GitSCM.buildEnvVars(GitSCM.java:1524)
	at hudson.model.AbstractBuild.getEnvironment(AbstractBuild.java:967)
	at PluginClassLoader for git//hudson.plugins.git.GitSCM._checkout(GitSCM.java:1299)
	at PluginClassLoader for git//hudson.plugins.git.GitSCM.checkout(GitSCM.java:1278)
	at hudson.scm.SCM.checkout(SCM.java:540)
	at hudson.model.AbstractProject.checkout(AbstractProject.java:1250)
	at hudson.model.AbstractBuild$AbstractBuildExecution.defaultCheckout(AbstractBuild.java:649)
	at jenkins.scm.SCMCheckoutStrategy.checkout(SCMCheckoutStrategy.java:85)
	at hudson.model.AbstractBuild$AbstractBuildExecution.run(AbstractBuild.java:522)
	at hudson.model.Run.execute(Run.java:1860)
	at hudson.model.FreeStyleBuild.run(FreeStyleBuild.java:44)
	at hudson.model.ResourceController.execute(ResourceController.java:101)
	at hudson.model.Executor.run(Executor.java:460)
ERROR: Build step failed with exception
java.lang.IllegalArgumentException: Git repository URL 1 is an empty string in job definition. Checkout requires a valid repository URL
	at PluginClassLoader for git//hudson.plugins.git.GitSCM.buildEnvironment(GitSCM.java:1579)
	at PluginClassLoader for git//hudson.plugins.git.GitSCM.buildEnvVars(GitSCM.java:1524)
	at hudson.model.AbstractBuild.getEnvironment(AbstractBuild.java:967)
	at hudson.tasks.BuildStepCompatibilityLayer.perform(BuildStepCompatibilityLayer.java:80)
	at hudson.tasks.BuildStepMonitor$1.perform(BuildStepMonitor.java:20)
	at hudson.model.AbstractBuild$AbstractBuildExecution.perform(AbstractBuild.java:818)
	at hudson.model.AbstractBuild$AbstractBuildExecution.performAllBuildSteps(AbstractBuild.java:767)
	at hudson.model.Build$BuildExecution.post2(Build.java:179)
```


Email notification setup
MavenJava_Build #17 Console
in this i want to select editable
azeemafirdouss/java_maven at
localhost:8080/job/MavenJava_Build/17/console
Jenkins
MavenJava_Build
#17
Console Output
REST API
Jenkins 2.530
21°C
Partly cloudy
Search
ENG
IN
00:01
12-11-2025

```
	at hudson.model.FreeStyleBuild.run(FreeStyleBuild.java:44)
	at hudson.model.ResourceController.execute(ResourceController.java:101)
	at hudson.model.Executor.run(Executor.java:460)
Build step 'Archive the artifacts' marked build as failure
ERROR: Build step failed with exception
java.lang.IllegalArgumentException: Git repository URL 1 is an empty string in job definition. Checkout requires a valid repository URL
	at PluginClassLoader for git//hudson.plugins.git.GitSCM.buildEnvironment(GitSCM.java:1579)
	at PluginClassLoader for git//hudson.plugins.git.GitSCM.buildEnvVars(GitSCM.java:1524)
	at hudson.model.AbstractBuild.getEnvironment(AbstractBuild.java:967)
	at PluginClassLoader for mailer//hudson.tasks.Mailer.perform(Mailer.java:147)
	at PluginClassLoader for mailer//hudson.tasks.Mailer.perform(Mailer.java:138)
	at hudson.tasks.BuildStepMonitor$1.perform(BuildStepMonitor.java:20)
	at hudson.model.AbstractBuild$AbstractBuildExecution.perform(AbstractBuild.java:818)
	at hudson.model.AbstractBuild$AbstractBuildExecution.performAllBuildSteps(AbstractBuild.java:767)
	at hudson.model.Build$BuildExecution.post2(Build.java:179)
	at hudson.model.AbstractBuild$AbstractBuildExecution.post(AbstractBuild.java:711)
	at hudson.model.Run.execute(Run.java:1883)
	at hudson.model.FreeStyleBuild.run(FreeStyleBuild.java:44)
	at hudson.model.ResourceController.execute(ResourceController.java:101)
	at hudson.model.Executor.run(Executor.java:460)
Build step 'E-mail Notification' marked build as failure
Email was triggered for: Failure - Any
Sending email for trigger: Failure - Any
Error retrieving environment vars: Git repository URL 1 is an empty string in job definition. Checkout requires a valid repository URL
Sending email to: azeemafirdous14@gmail.com
Finished: FAILURE
```

```
Build step 'E-mail Notification' marked build as failure
Email was triggered for: Failure - Any
Sending email for trigger: Failure - Any
Error retrieving environment vars: Git repository URL 1 is an empty string in job definition. Checkout requires a valid repository URL
Sending email to: azeemafirdous14@gmail.com
Finished: FAILURE
```



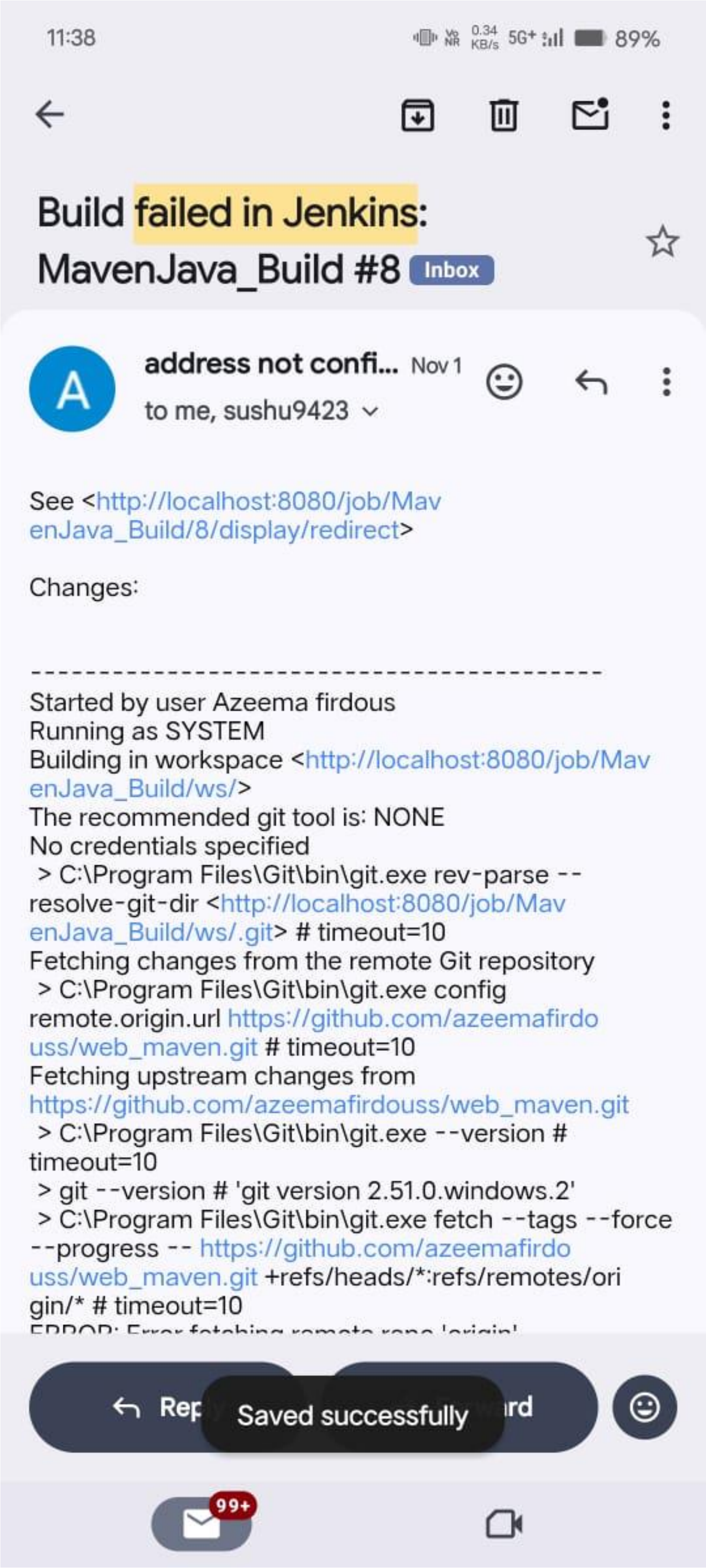
Most relevant

me

Nov 1

Build failed in Jenkins: MavenJava_Build...

GitException: Failed to fetch from...



Build failed in Jenkins: MavenJava_Build #8

address not confi... Nov 1

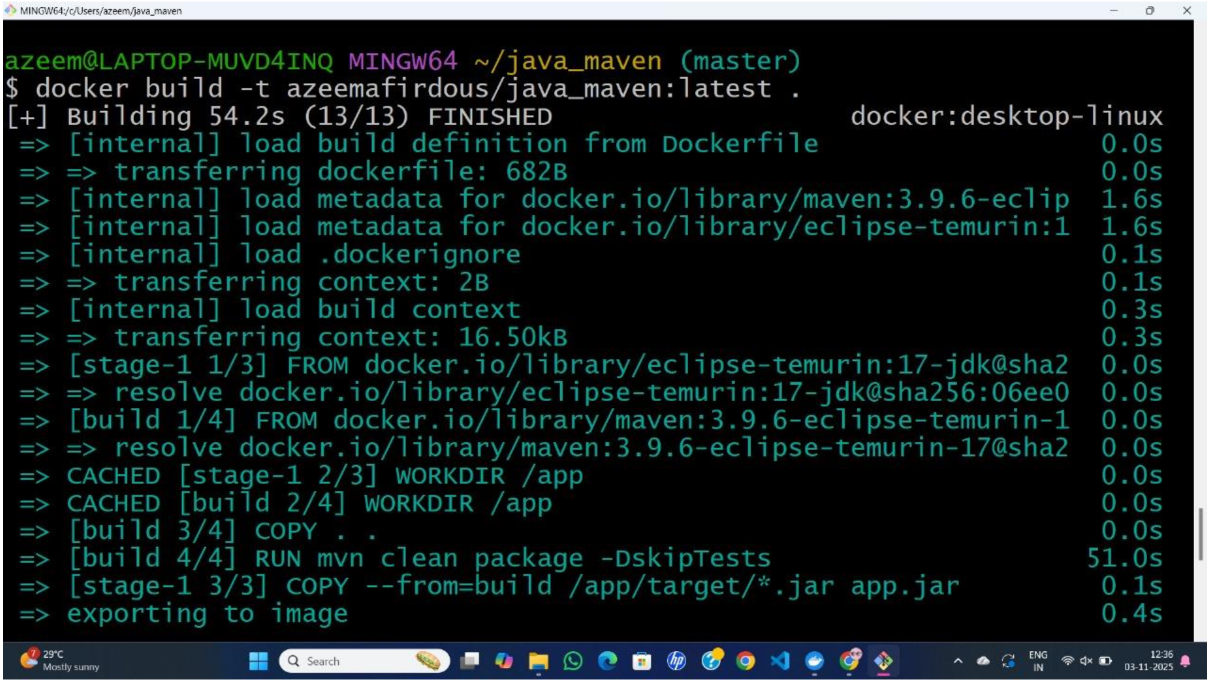
to me, sushu9423

See <http://localhost:8080/job/MavenJava_Build/8/display/redirect>

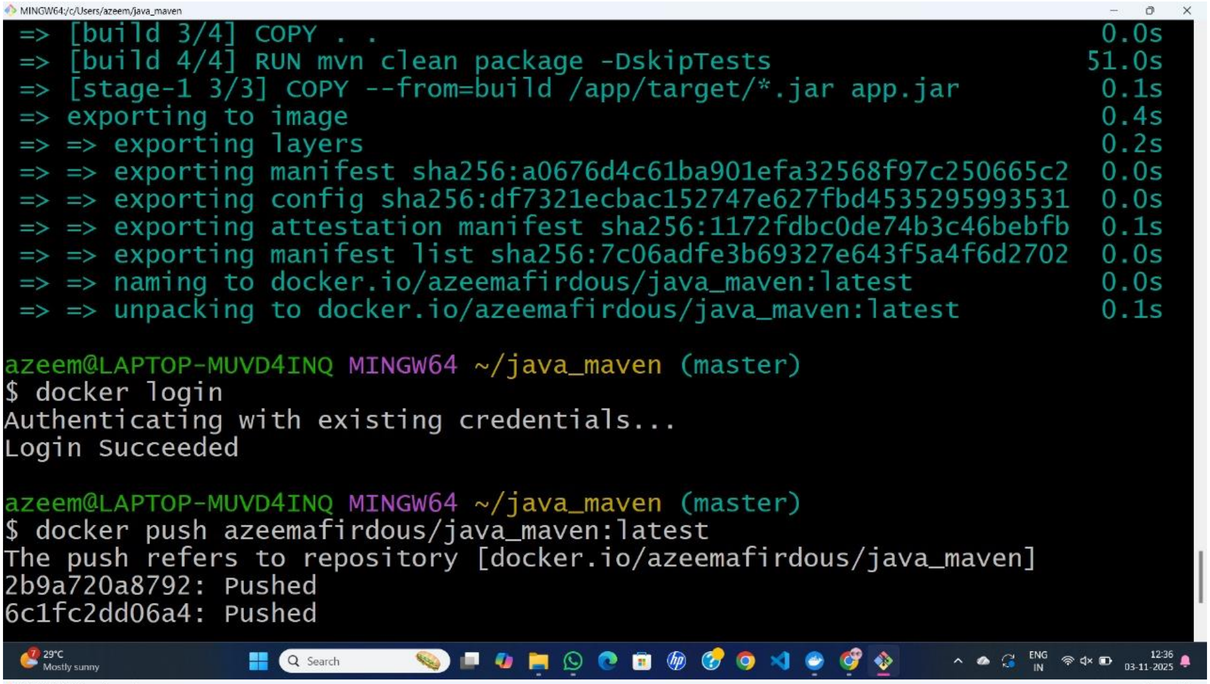
Changes:

```
------------------------------------------
Started by user Azeema firdous
Running as SYSTEM
Building in workspace <http://localhost:8080/job/MavenJava_Build/ws/>
The recommended git tool is: NONE
No credentials specified
 > C:\Program Files\Git\bin\git.exe rev-parse --resolve-git-dir <http://localhost:8080/job/MavenJava_Build/ws/.git> # timeout=10
Fetching changes from the remote Git repository
 > C:\Program Files\Git\bin\git.exe config remote.origin.url https://github.com/azeemafirdouss/web_maven.git # timeout=10
Fetching upstream changes from https://github.com/azeemafirdouss/web_maven.git
 > C:\Program Files\Git\bin\git.exe --version # timeout=10
 > git --version # 'git version 2.51.0.windows.2'
 > C:\Program Files\Git\bin\git.exe fetch --tags --force --progress -- https://github.com/azeemafirdouss/web_maven.git +refs/heads/*:refs/remotes/origin/* # timeout=10
ERROR: Error fetching remote repo 'origin'
```

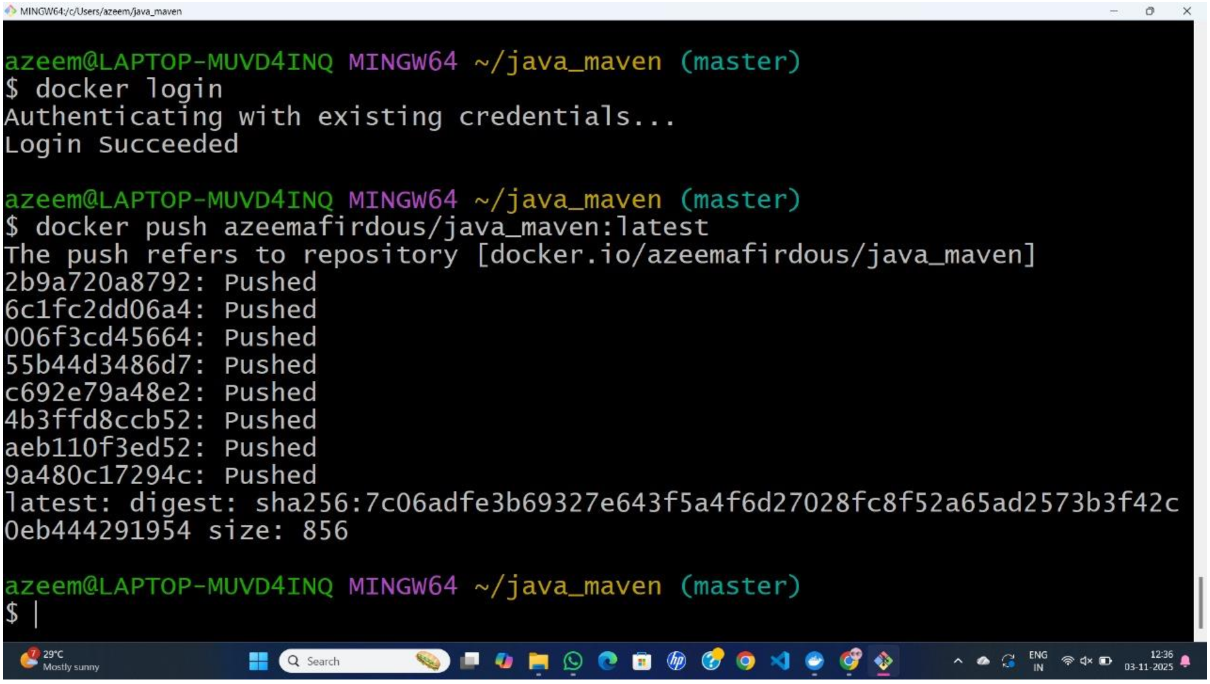
Docker hub



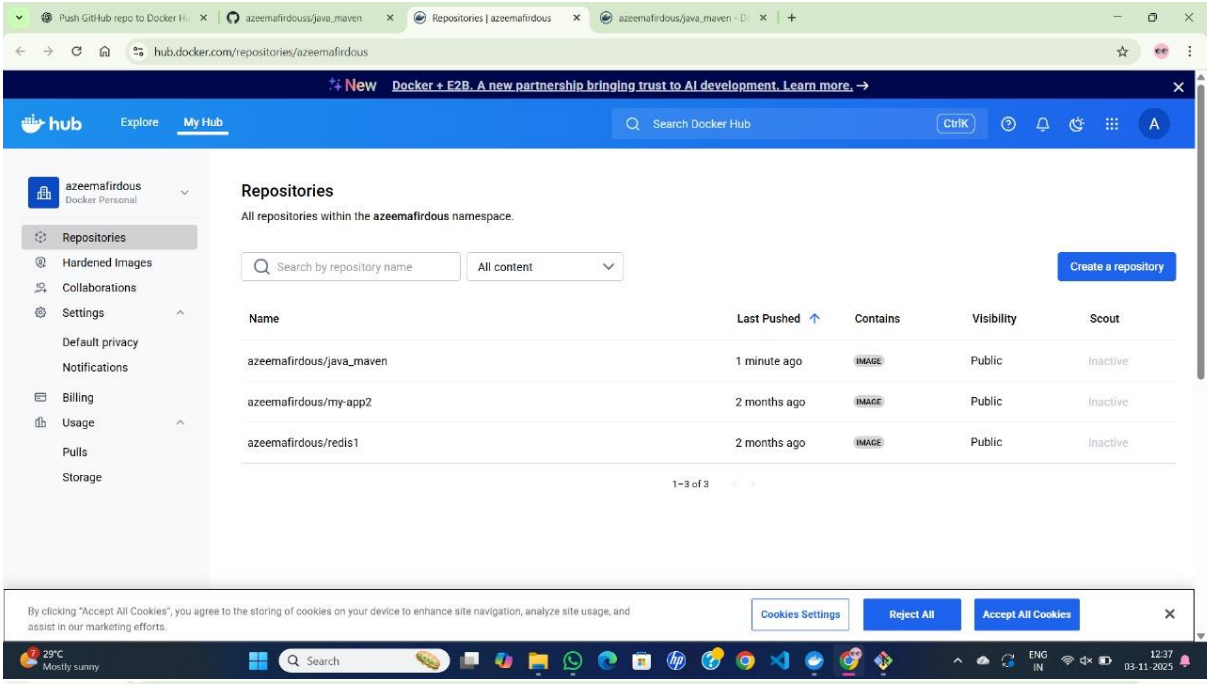
```
azeem@LAPTOP-MUVD4INQ MINGW64 ~/java_maven (master)
$ docker build -t azeemafirdous/java_maven:latest .
[+] Building 54.2s (13/13) FINISHED                    docker:desktop-linux
 => [internal] load build definition from Dockerfile                   0.0s
 => => transferring dockerfile: 682B                                   0.0s
 => [internal] load metadata for docker.io/library/maven:3.9.6-eclip   1.6s
 => [internal] load metadata for docker.io/library/eclipse-temurin:1   1.6s
 => [internal] load .dockerignore                                      0.1s
 => => transferring context: 2B                                        0.1s
 => [internal] load build context                                      0.3s
 => => transferring context: 16.50kB                                   0.3s
 => [stage-1 1/3] FROM docker.io/library/eclipse-temurin:17-jdk@sha2   0.0s
 => => resolve docker.io/library/eclipse-temurin:17-jdk@sha256:06ee0   0.0s
 => [build 1/4] FROM docker.io/library/maven:3.9.6-eclipse-temurin-1   0.0s
 => => resolve docker.io/library/maven:3.9.6-eclipse-temurin-17@sha2   0.0s
 => CACHED [stage-1 2/3] WORKDIR /app                                  0.0s
 => CACHED [build 2/4] WORKDIR /app                                    0.0s
 => [build 3/4] COPY . .                                               0.0s
 => [build 4/4] RUN mvn clean package -DskipTests                     51.0s
 => [stage-1 3/3] COPY --from=build /app/target/*.jar app.jar          0.1s
 => exporting to image                                                 0.4s
```



```
 => [build 3/4] COPY . .                                                0.0s
 => [build 4/4] RUN mvn clean package -DskipTests                      51.0s
 => [stage-1 3/3] COPY --from=build /app/target/*.jar app.jar           0.1s
 => exporting to image                                                  0.4s
 => => exporting layers                                                 0.2s
 => => exporting manifest sha256:a0676d4c61ba901efa32568f97c250665c2    0.0s
 => => exporting config sha256:df7321ecbac152747e627fbd4535295993531    0.0s
 => => exporting attestation manifest sha256:1172fdbc0de74b3c46bebfb    0.1s
 => => exporting manifest list sha256:7c06adfe3b69327e643f5a4f6d2702    0.0s
 => => naming to docker.io/azeemafirdous/java_maven:latest              0.0s
 => => unpacking to docker.io/azeemafirdous/java_maven:latest           0.1s

azeem@LAPTOP-MUVD4INQ MINGW64 ~/java_maven (master)
$ docker login
Authenticating with existing credentials...
Login Succeeded

azeem@LAPTOP-MUVD4INQ MINGW64 ~/java_maven (master)
$ docker push azeemafirdous/java_maven:latest
The push refers to repository [docker.io/azeemafirdous/java_maven]
2b9a720a8792: Pushed
6c1fc2dd06a4: Pushed
```



```

azeem@LAPTOP-MUVD4INQ MINGW64 ~/java_maven (master)
$ docker login
Authenticating with existing credentials...
Login Succeeded

azeem@LAPTOP-MUVD4INQ MINGW64 ~/java_maven (master)
$ docker push azeemafirdous/java_maven:latest
The push refers to repository [docker.io/azeemafirdous/java_maven]
2b9a720a8792: Pushed
6c1fc2dd06a4: Pushed
006f3cd45664: Pushed
55b44d3486d7: Pushed
c692e79a48e2: Pushed
4b3ffd8ccb52: Pushed
aeb110f3ed52: Pushed
9a480c17294c: Pushed
latest: digest: sha256:7c06adfe3b69327e643f5a4f6d27028fc8f52a65ad2573b3f42c
0eb444291954 size: 856

azeem@LAPTOP-MUVD4INQ MINGW64 ~/java_maven (master)
$ |
```

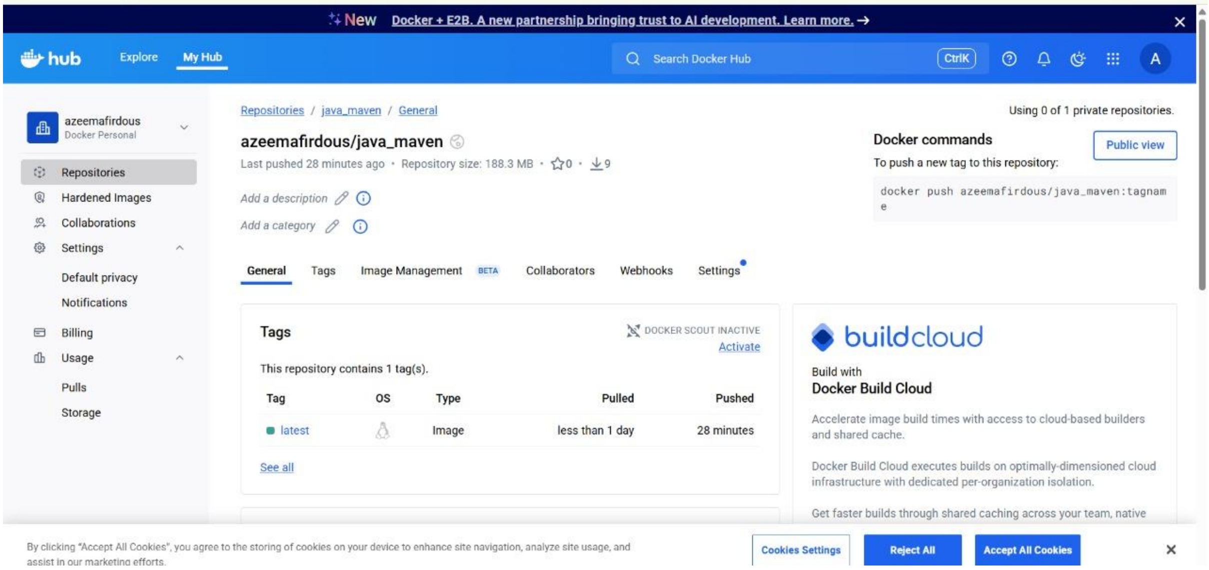


Repositories

All repositories within the **azeemafirdous** namespace.

Name	Last Pushed	Contains	Visibility	Scout
azeemafirdous/java_maven	1 minute ago	IMAGE	Public	Inactive
azeemafirdous/my-app2	2 months ago	IMAGE	Public	Inactive
azeemafirdous/redis1	2 months ago	IMAGE	Public	Inactive

By clicking "Accept All Cookies", you agree to the storing of cookies on your device to enhance site navigation, analyze site usage, and assist in our marketing efforts.



Using 0 of 1 private repositories.

azeemafirdous/java_maven

Last pushed 28 minutes ago · Repository size: 188.3 MB · ☆0 · ⤓9

Add a description

Add a category

Docker commands

To push a new tag to this repository:

```
docker push azeemafirdous/java_maven:tagnam
e
```

Tags

This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
latest		Image	less than 1 day	28 minutes

Build with
Docker Build Cloud

Accelerate image build times with access to cloud-based builders and shared cache.

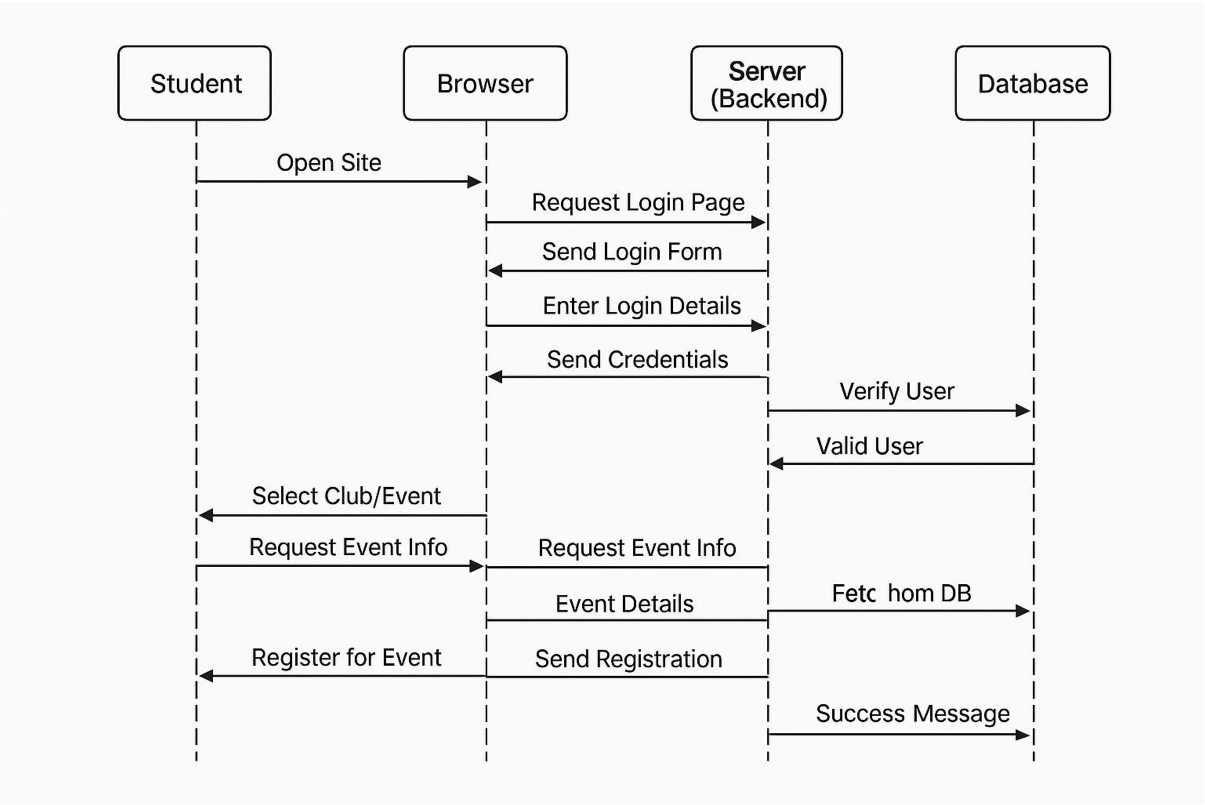
Docker Build Cloud executes builds on optimally-dimensioned cloud infrastructure with dedicated per-organization isolation.

Get faster builds through shared caching across your team, native

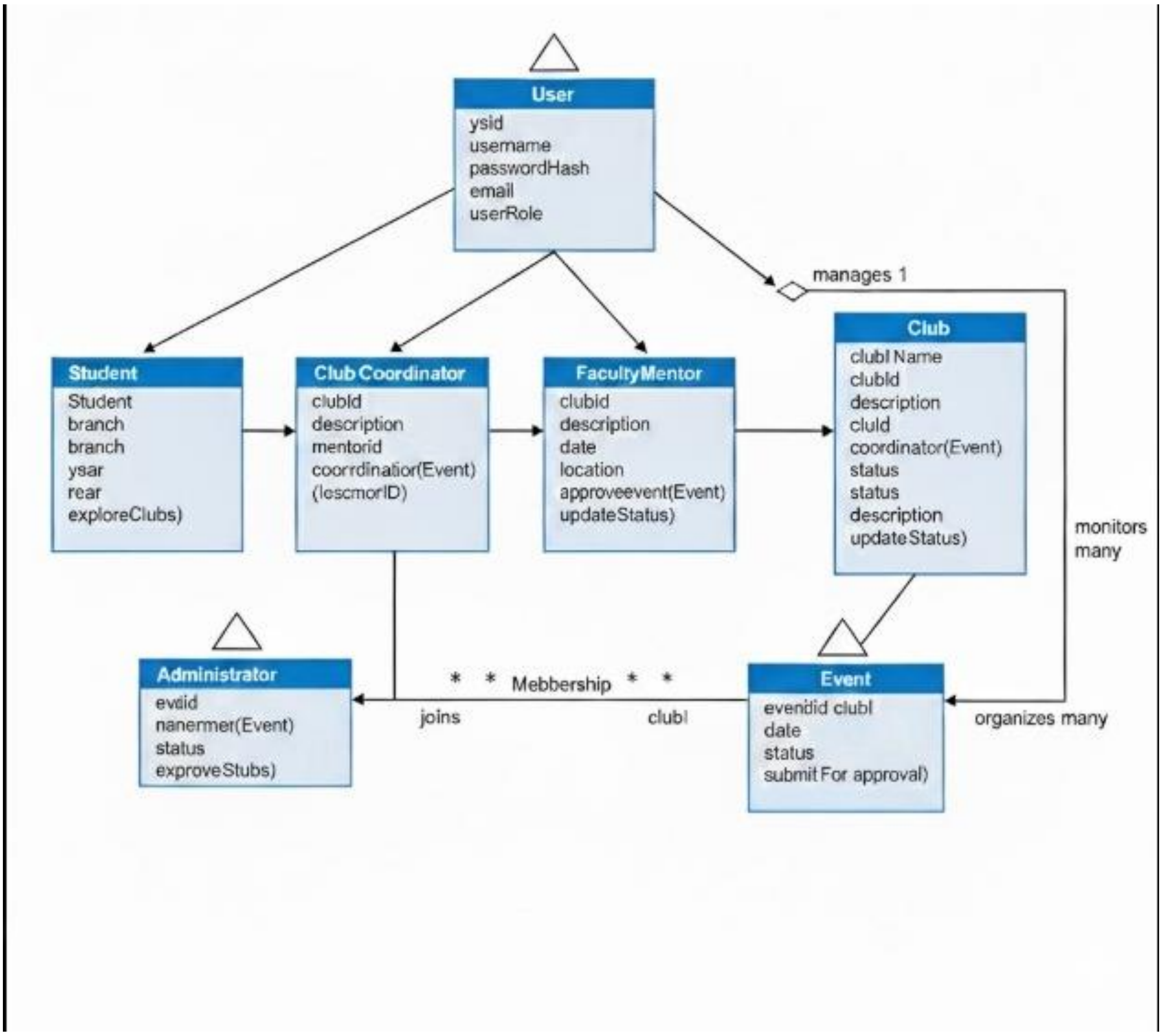
By clicking "Accept All Cookies", you agree to the storing of cookies on your device to enhance site navigation, analyze site usage, and assist in our marketing efforts.

UML DIAGRAMS

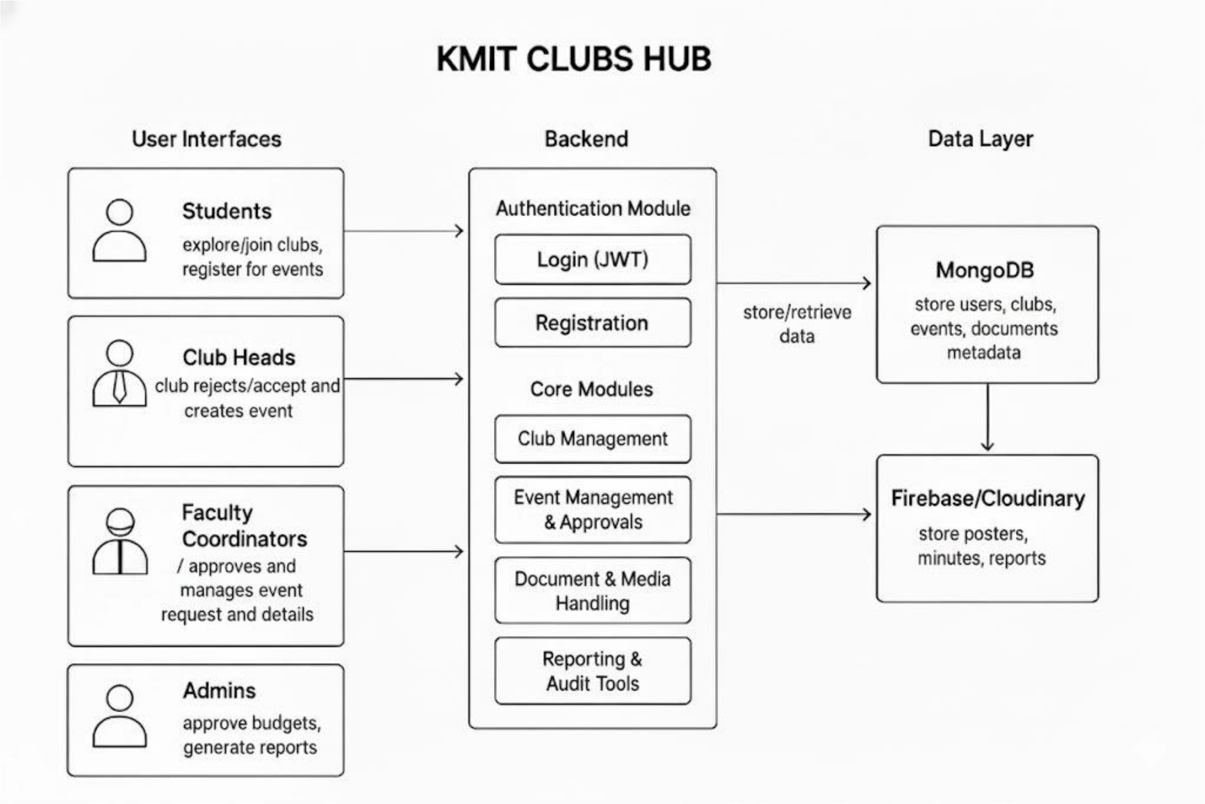
SEQUENCE DIAGRAM



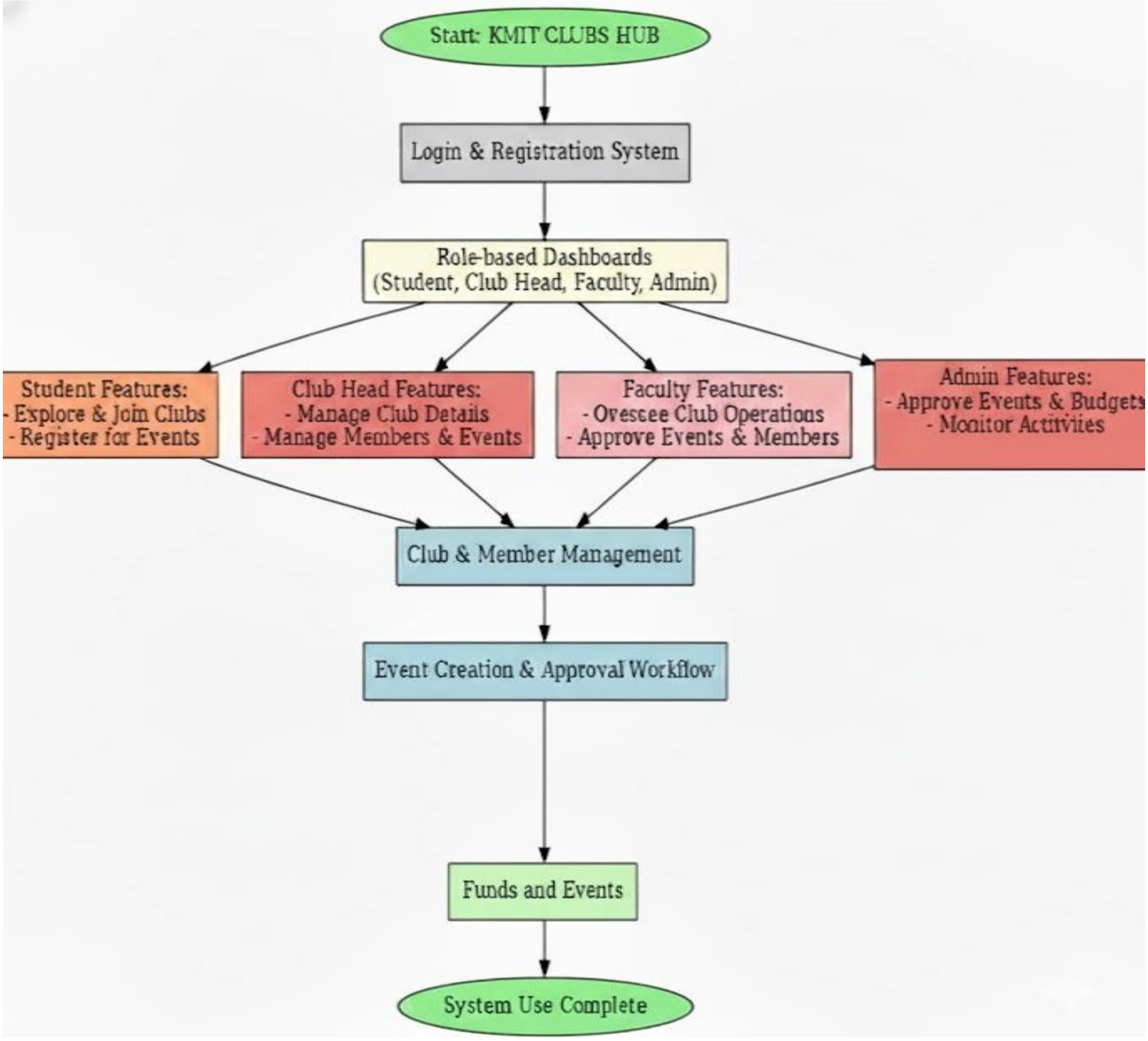
CLASS DAIGRAM


User
ysid
username
passwordHash
email
userRole
manages 1
Student
Student
branch
branch
ysar
rear
exploreClubs)
Club Coordinator
clubId
description
mentorid
coorrdinatior(Event)
(lescmorID)
FacultyMentor
clubid
description
date
location
approveevent(Event)
updateStatus)
Club
clubl Name
clubld
description
cluld
coordinator(Event)
status
status
description
update Status)
monitors
many
Administrator
evdid
nanermer(Event)
status
exproveStubs)
* * Mebbership * *
joins
clubl
Event
evendid clubl
date
status
submit For approval)
organizes many

COMPONENT DIAGRAM


KMIT CLUBS HUB
User Interfaces
Backend
Data Layer
Students
explore/join clubs, register for events
Club Heads
club rejects/accept and creates event
Faculty Coordinators
/ approves and manages event request and details
Admins
approve budgets, generate reports
Authentication Module
Login (JWT)
Registration
Core Modules
Club Management
Event Management & Approvals
Document & Media Handling
Reporting & Audit Tools
store/retrieve data
MongoDB
store users, clubs, events, documents metadata
Firebase/Cloudinary
store posters, minutes, reports

STATE CHART DIAGRAM


Start: KMIT CLUBS HUB
Login & Registration System
Role-based Dashboards (Student, Club Head, Faculty, Admin)
Student Features: - Explore & Join Clubs - Register for Events
Club Head Features: - Manage Club Details - Manage Members & Events
Faculty Features: - Oversee Club Operations - Approve Events & Members
Admin Features: - Approve Events & Budgets - Monitor Activities
Club & Member Management
Event Creation & Approval Workflow
Funds and Events
System Use Complete

SECNARIO BASED QUESTIONS

1. **What is Continuous Integration (CI)?**
   Continuous Integration (CI) is a software development practice where developers frequently merge their code changes into a shared repository, after which automated builds and tests run to detect errors early and ensure the codebase remains stable.
2. **What is Continuous Deployment or Continuous Delivery (CD)?**
   Continuous Deployment/Delivery (CD) is the process of automatically deploying every change that passes all tests to production (Continuous Deployment) or to a staging environment (Continuous Delivery), ensuring fast and reliable releases.
3. **What is the role of Jenkins in a CI/CD pipeline?**
   Jenkins automates the CI/CD process by integrating code building, testing, and deployment. It acts as an orchestrator that triggers builds, runs test scripts, and deploys applications to servers or cloud platforms automatically.

4. **What is a webhook in GitHub?**
   A webhook in GitHub is a mechanism that automatically sends real-time notifications (HTTP POST requests) to a specified URL whenever certain events occur in a repository, such as code push or pull requests.
5. **Why are webhooks used in Jenkins integration?**
   Webhooks are used in Jenkins integration to automatically trigger builds whenever code changes are pushed to GitHub, eliminating the need for manual build triggering or periodic polling.

6. What are the different types of build triggers available in Jenkins?
The common build triggers in Jenkins are:

- **Build periodically** (scheduled builds using cron syntax)
- **Poll SCM** (Jenkins checks the repository for changes)
- **GitHub webhook trigger** (automatic trigger via webhooks)
- **Manual build trigger** (clicked by a user)
- **Trigger builds remotely** (via URL or API)

7. What is the difference between polling and webhook triggers?

- **Polling**: Jenkins checks GitHub periodically to see if there are new commits — less efficient and adds delay.
- **Webhook**: GitHub instantly notifies Jenkins when a change happens — faster and more efficient.

8. What is ngrok and why is it used in Jenkins–GitHub integration?
Ngrok is a tool that creates a secure public URL (tunnel) for a local server. In Jenkins–GitHub integration, it is used to expose a locally running Jenkins server to the internet so GitHub can send webhook notifications.

9. How does ngrok help in setting up webhooks for Jenkins running on a local machine?
Ngrok provides a temporary public HTTPS URL that forwards requests to Jenkins running on localhost (e.g., http://localhost:8080). This allows GitHub webhooks to reach Jenkins for triggering builds even though Jenkins isn’t publicly hosted.

10. Why do we configure email notifications in Jenkins and how are they useful for monitoring build results?
Email notifications in Jenkins are configured to automatically inform developers or admins about build status (success, failure, unstable). This helps in quick monitoring, immediate issue detection, and faster debugging of failed builds.