PRACTICAL NO. 3

GitLab Essentials

LOB2 To obtain knowledge of Version Control Systems to effectively track changes with Git, GitHub and understand their best practices in team environments.

LO2 Demonstrate the use of Git and GitHub to manage version control in projects and compare different workflows.

- **Introduction to GitLab**

GitLab is a web-based platform that helps software teams manage projects, collaborate, and build software. It is built around Git, a distributed version control system.

GitLab helps teams reduce product lifecycles and increase productivity, which in turn creates value for customers.

The application doesn't require users to manage authorizations for each tool. If permissions are set once, then everyone in the organization has access to every component.

GitLab was originally fully free and open-source software distributed under the MIT License. It was split into two distinct versions in July 2013 –

- **GitLab CE (Community Edition) and**
- **GitLab EE (Enterprise Edition)**.

- What does GitLab do?
  - Project management: Plan projects, track issues, and manage tasks.
  - Code review: Review code and collaboration on issues and projects.
  - Continuous integration and deployment (CI/CD): Automate repetitive tasks like building and testing new changes.
  - Security: Incorporate security measures from the start and throughout the development process.
  - Version control: Manage source code, database structure, libraries, and more
- How does GitLab help?
  - Collaboration: Helps distributed teams work together more cohesively and productively.
  - Efficiency: Reduces product lifecycles and increases productivity.
  - Cost savings: Helps businesses save money on licensing costs
- GitLab features
  - Free and open-source Community Edition (CE).
  - Enterprise Edition (EE) with a restricted license.
  - AI-powered workflows to support teams throughout the software development lifecycle.

- **Creating a GitLab Repository**
  - Sign in to GitLab at [GitLab](https://gitlab.com/)
  - Click on New Project → Create Blank Project
  - Enter Project Name and select Visibility (Private/Public)
  - Click Create Project
  - One can use GitLab Web IDE that allows you to edit, commit, and push changes directly from the browser.

- **Git Commands to Interact with GitHub**
  - Although GitLab has its own repository system, developers often work with GitHub as well. Here are the essential Git commands:

Command	Description
`git init`	Initialize a new Git repository
`git clone <repo_url>`	Clone a repository
`git status`	Show modified files
`git add <file>`	Add files to staging area
`git commit -m "message"`	Commit changes
`git push origin main`	Push commits to GitLab/GitHub
`git pull origin main`	Pull the latest changes

- **Overview of CI/CD Workflow**
  - GitLab CI/CD Workflow
    Developer pushes code to GitLab
    GitLab CI/CD pipeline runs the `.gitlab-ci.yml` script
    The pipeline executes build, test, and deploy stages
    Deployment is triggered if all tests pass
  - CI/CD Capabilities of GitLab
    GitLab CI/CD enables automation of:
    - Build processes (e.g., compiling Java, Node.js apps)
    - Testing (e.g., unit tests, integration tests)
    - Deployment (e.g., deploying to a web server, cloud, Kubernetes)
    - Monitoring (e.g., logging, performance analysis)

- **About `.gitlab-ci.yml` File**
  - The `.gitlab-ci.yml` file defines CI/CD pipelines in GitLab.
  - Structure of `.gitlab-ci.yml`

```yaml
image: ubuntu:latest  # Base Docker image

stages:
  - build
  - test
  - deploy

build:
  stage: build
  script:
    - echo "Building the application"

unit-test:
  stage: test
  script:
    - echo "Running unit tests"

deploy:
  stage: deploy
  script:
    - echo "Deploying application"
  only:
    - main
```

  - Explanation of Key Sections
    - `image:` Specifies the Docker image for the pipeline.

- `stages:` Defines pipeline stages (e.g., build, test, deploy).
- `script:` Contains commands to execute in each stage.
- `only:` Runs the job only on a specific branch (e.g., `main`).

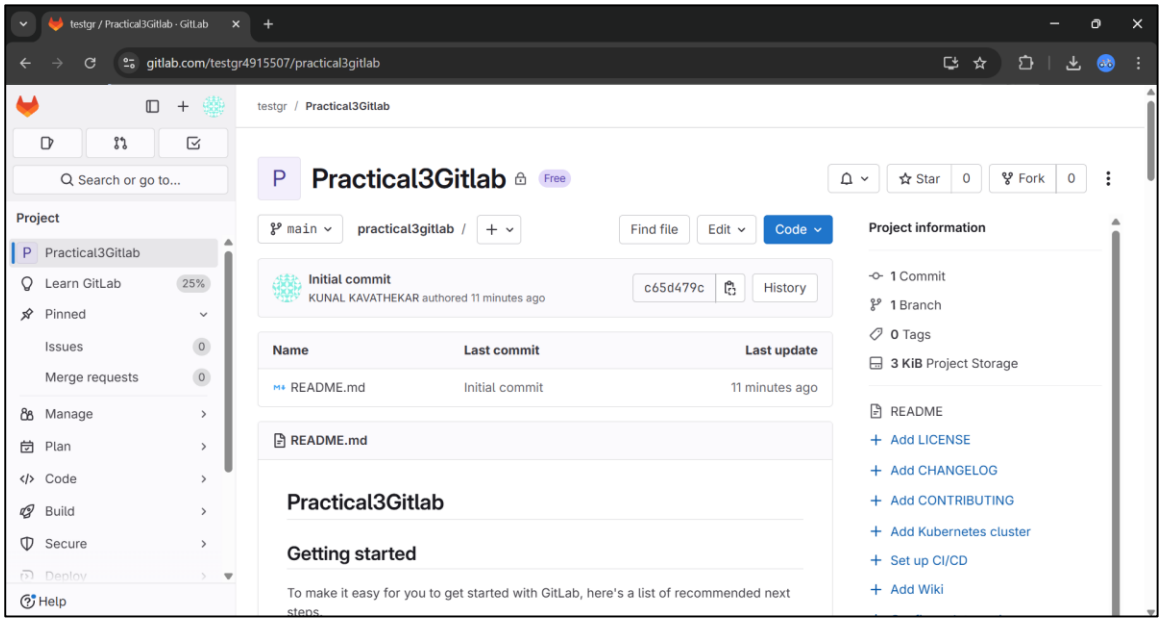
- **Running and Monitoring the Pipeline**
  - Triggering the Pipeline –
    - Commit and push the `.gitlab-ci.yml` file:

      ```
      git add .gitlab-ci.yml
      git commit -m "Added GitLab CI/CD pipeline"
      git push origin main
      ```

    - Go to GitLab → CI/CD → Pipelines
    - Click Run Pipeline (if not triggered automatically)
  - Viewing Pipeline Status –
    - Go to GitLab → CI/CD → Pipelines
    - Click on a pipeline to see logs and results of each stage.

Exercise:

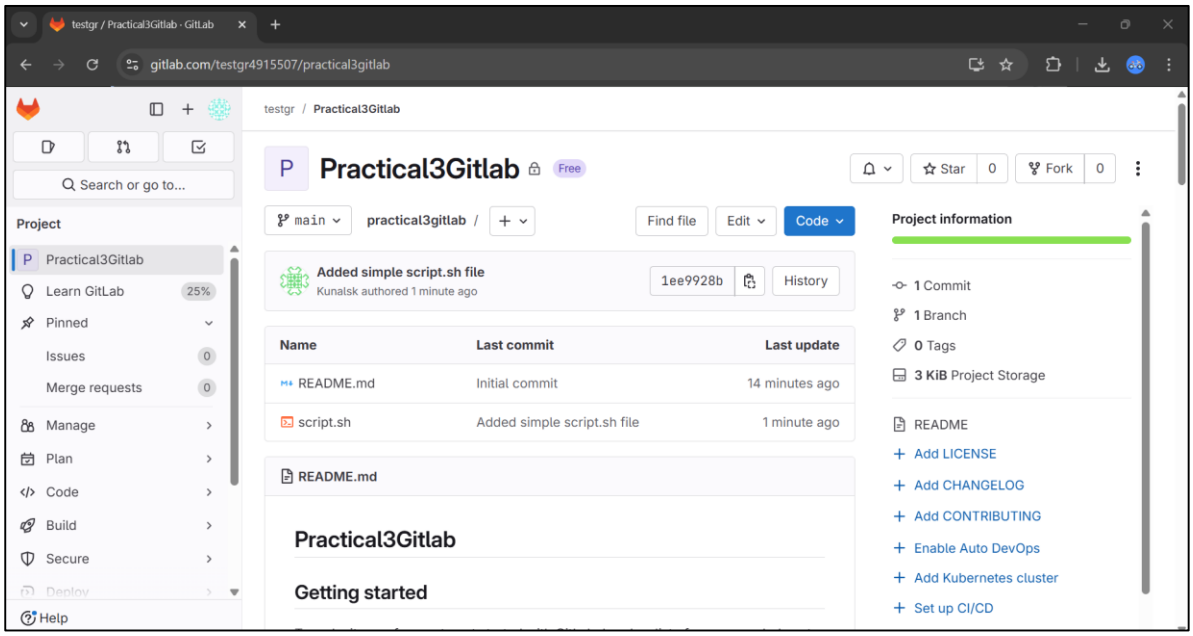
- Basic GitLab Repository Management – A team is starting a new project and needs to set up a GitLab repository by performing following tasks –
    a. Create a new repository in GitLab.



    b. Clone the repository locally.

```
Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps
$ mkdir Practical3

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps
$ cd Practical3

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3
$ git clone https://gitlab.com/testgr4915507/practical3gitlab.git
Cloning into 'practical3gitlab'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3
$ cd practical3gitlab

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3/practical3gitlab (m
ain)
$ echo "Hello I am kunal" > script.sh

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3/practical3gitlab (m
ain)
$ git add script.sh

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3/practical3gitlab (m
ain)
$ git commit -m "Added simple script.sh file"
[main 1ee9928] Added simple script.sh file
 1 file changed, 1 insertion(+)
 create mode 100644 script.sh

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3/practical3gitlab (m
ain)
$ git push origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
```

c. Add simple script file, commit changes, and push them to GitLab.

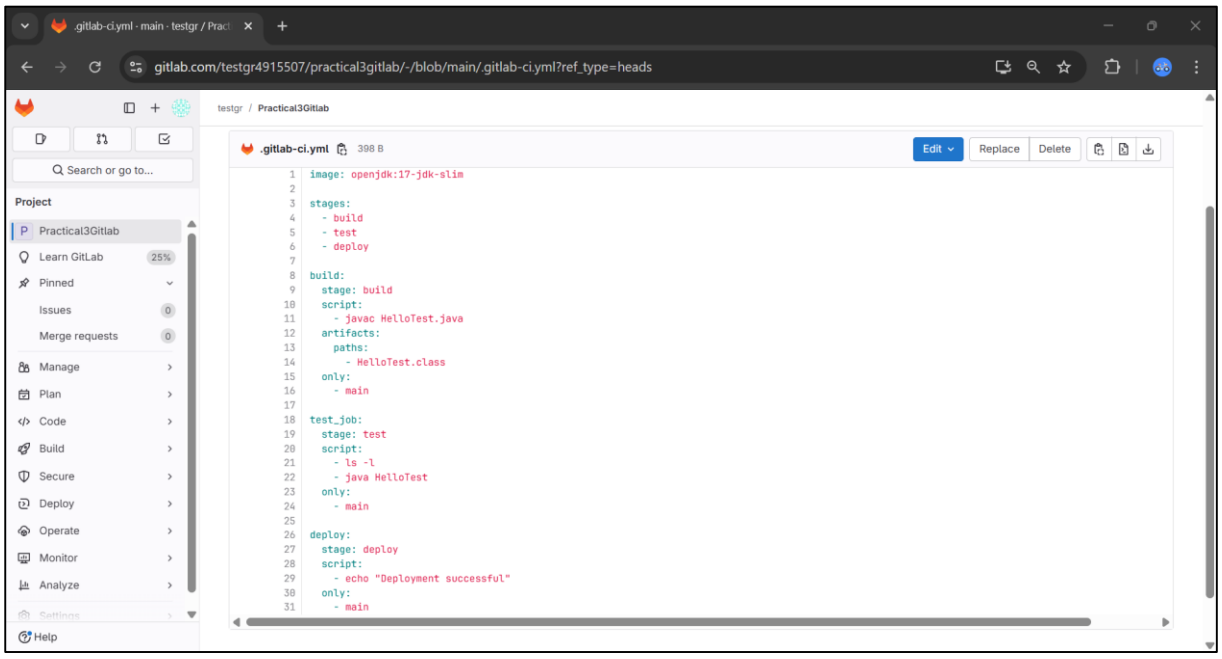


d. Create and switch branches for different features.

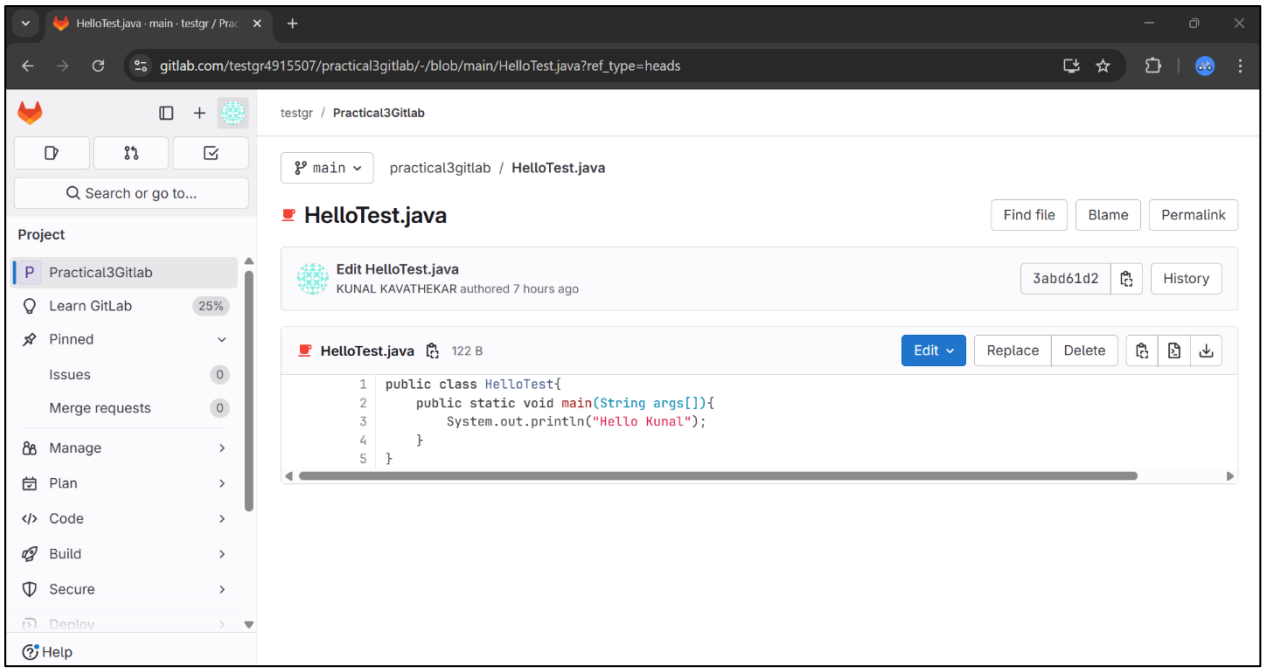
```
Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3/practical3gitlab (main)
$ git checkout -b feature1
Switched to a new branch 'feature1'

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3/practical3gitlab (feature1)
$
```

- Setting Up GitLab CI/CD Pipeline – Automate build and testing for a Java project.

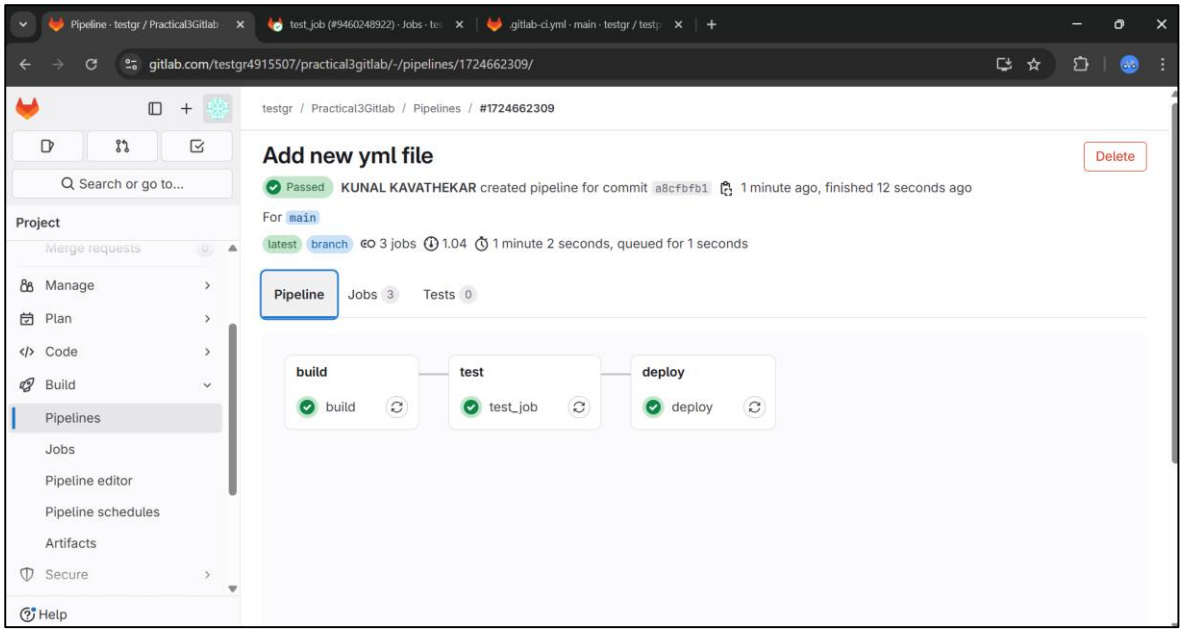
a. Write a `.gitlab-ci.yml` file to define pipeline stages.

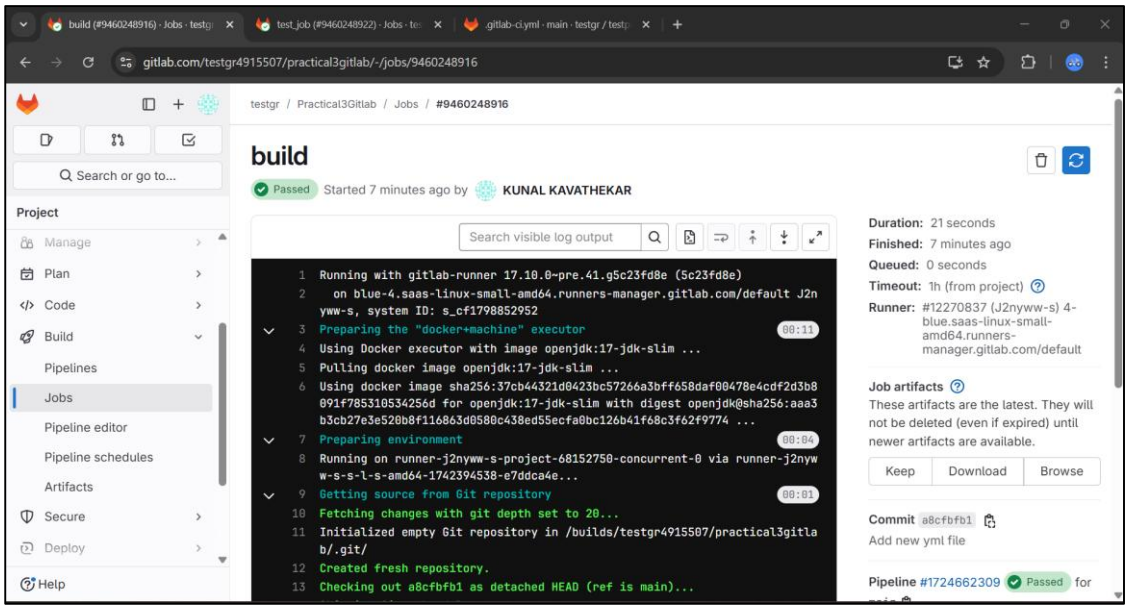


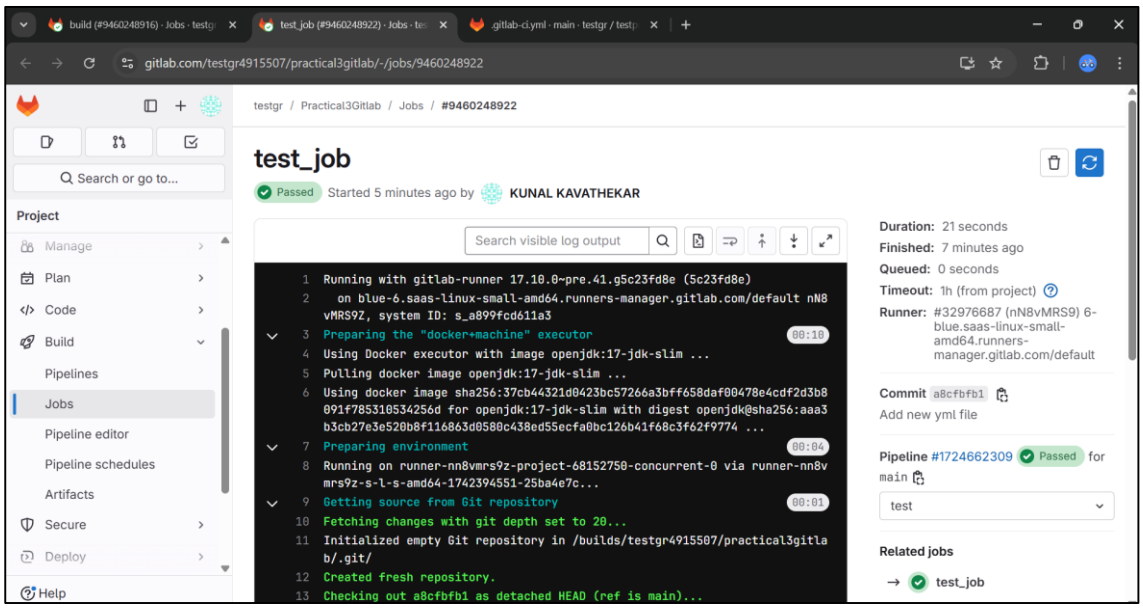
b. Use an appropriate Docker image for Java compilation.

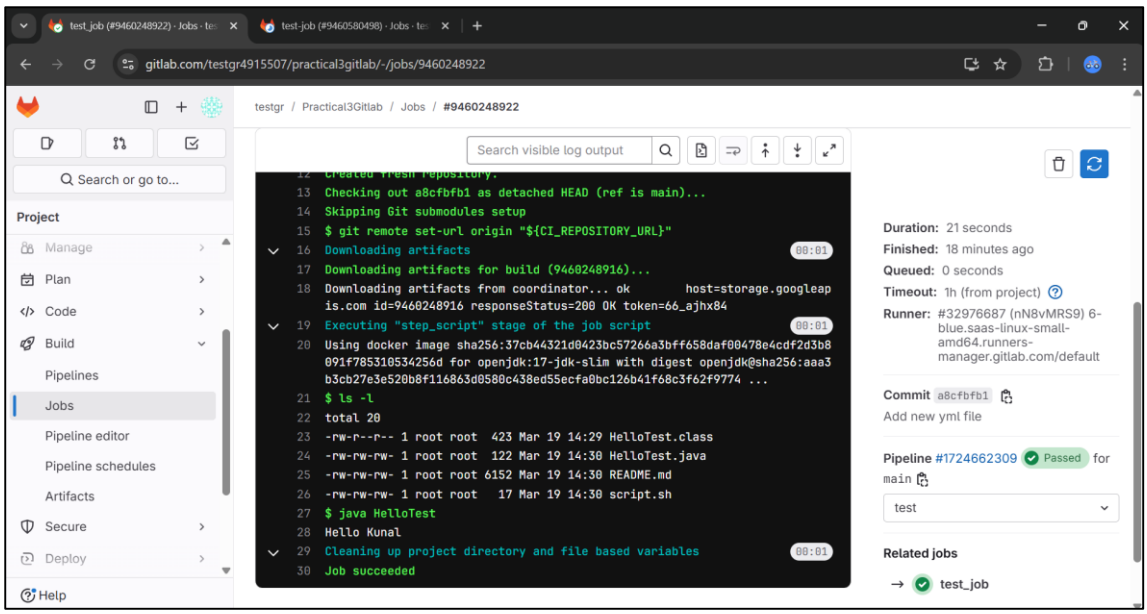


c. Configure stages for Build, Test, and Deploy.

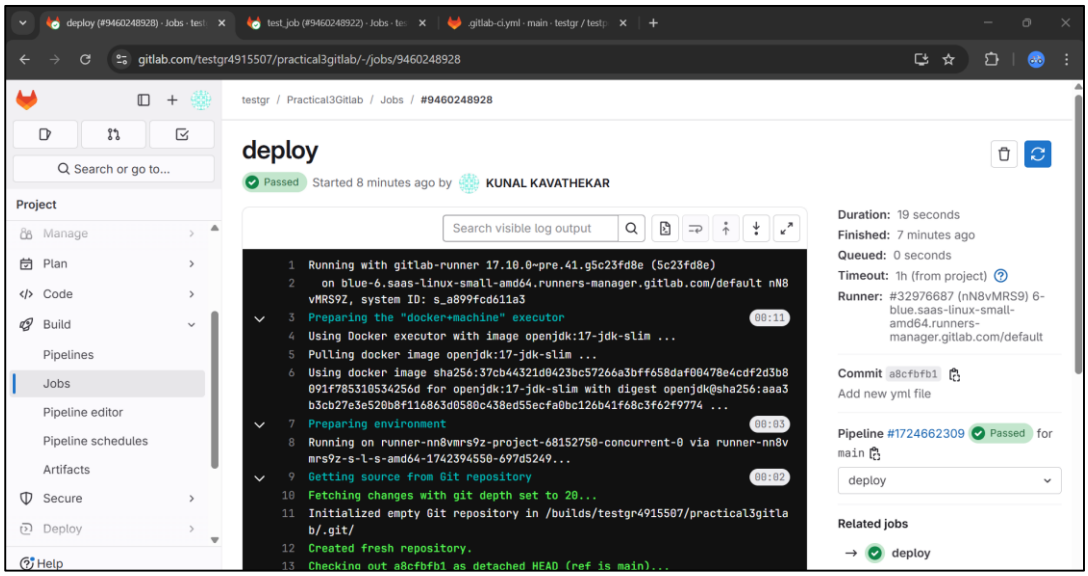



gitlab.com/testgr4915507/practical3gitlab/-/jobs/9460248916
testgr / Practical3Gitlab / Jobs / #9460248916
build
Passed Started 7 minutes ago by KUNAL KAVATHEKAR
Search visible log output
Running with gitlab-runner 17.10.0~pre.41.g5c23fd8e (5c23fd8e)
on blue-4.saas-linux-small-amd64.runners-manager.gitlab.com/default J2nyww-s, system ID: s_cf1798852952
Preparing the "docker+machine" executor 00:11
Using Docker executor with image openjdk:17-jdk-slim ...
Pulling docker image openjdk:17-jdk-slim ...
Using docker image sha256:37cb44321d0423bc57266a3bff658daf00478e4cdf2d3b8091f78531053425 6d for openjdk:17-jdk-slim with digest openjdk@sha256:aaa3b3cb27e3e520b8f116863d0580c438ed55ecfa0bc126b41f68c3f62f9774 ...
Preparing environment 00:04
Running on runner-j2nyww-s-project-68152750-concurrent-0 via runner-j2nyww-s-s-l-s-amd64-1742394538-e7ddca4e...
Getting source from Git repository 00:01
Fetching changes with git depth set to 20...
Initialized empty Git repository in /builds/testgr4915507/practical3gitlab/.git/
Created fresh repository.
Checking out a8cfbfb1 as detached HEAD (ref is main)...
Duration: 21 seconds
Finished: 7 minutes ago
Queued: 0 seconds
Timeout: 1h (from project)
Runner: #12270837 (J2nyww-s) 4-blue.saas-linux-small-amd64.runners-manager.gitlab.com/default
Job artifacts
These artifacts are the latest. They will not be deleted (even if expired) until newer artifacts are available.
Keep Download Browse
Commit a8cfbfb1
Add new yml file
Pipeline #1724662309 Passed for


gitlab.com/testgr4915507/practical3gitlab/-/jobs/9460248922
testgr / Practical3Gitlab / Jobs / #9460248922
test_job
Passed Started 5 minutes ago by KUNAL KAVATHEKAR
Search visible log output
Running with gitlab-runner 17.10.0~pre.41.g5c23fd8e (5c23fd8e)
on blue-6.saas-linux-small-amd64.runners-manager.gitlab.com/default nN8vMRS9Z, system ID: s_a899fcd611a3
Preparing the "docker+machine" executor 00:10
Using Docker executor with image openjdk:17-jdk-slim ...
Pulling docker image openjdk:17-jdk-slim ...
Using docker image sha256:37cb44321d0423bc57266a3bff658daf00478e4cdf2d3b8091f78531053425 6d for openjdk:17-jdk-slim with digest openjdk@sha256:aaa3b3cb27e3e520b8f116863d0580c438ed55ecfa0bc126b41f68c3f62f9774 ...
Preparing environment 00:04
Running on runner-nn8vmrs9z-project-68152750-concurrent-0 via runner-nn8vmrs9z-s-l-s-amd64-1742394551-25ba4e7c...
Getting source from Git repository 00:01
Fetching changes with git depth set to 20...
Initialized empty Git repository in /builds/testgr4915507/practical3gitlab/.git/
Created fresh repository.
Checking out a8cfbfb1 as detached HEAD (ref is main)...
Duration: 21 seconds
Finished: 7 minutes ago
Queued: 0 seconds
Timeout: 1h (from project)
Runner: #32976687 (nN8vMRS9) 6-blue.saas-linux-small-amd64.runners-manager.gitlab.com/default
Commit a8cfbfb1
Add new yml file
Pipeline #1724662309 Passed for main
test
Related jobs
test_job


gitlab.com/testgr4915507/practical3gitlab/-/jobs/9460248922
testgr / Practical3Gitlab / Jobs / #9460248922
Search visible log output
Checking out a8cfbfb1 as detached HEAD (ref is main)...
Skipping Git submodules setup
$ git remote set-url origin "${CI_REPOSITORY_URL}"
Downloading artifacts 00:01
Downloading artifacts for build (9460248916)...
Downloading artifacts from coordinator... ok host=storage.googleapis.com id=9460248916 responseStatus=200 OK token=66_ajhx84
Executing "step_script" stage of the job script 00:01
Using docker image sha256:37cb44321d0423bc57266a3bff658daf00478e4cdf2d3b8091f78531053425 6d for openjdk:17-jdk-slim with digest openjdk@sha256:aaa3b3cb27e3e520b8f116863d0580c438ed55ecfa0bc126b41f68c3f62f9774 ...
$ ls -l
total 20
-rw-r--r-- 1 root root 423 Mar 19 14:29 HelloTest.class
-rw-rw-rw- 1 root root 122 Mar 19 14:30 HelloTest.java
-rw-rw-rw- 1 root root 6152 Mar 19 14:30 README.md
-rw-rw-rw- 1 root root 17 Mar 19 14:30 script.sh
$ java HelloTest
Hello Kunal
Cleaning up project directory and file based variables 00:01
Job succeeded
Duration: 21 seconds
Finished: 18 minutes ago
Queued: 0 seconds
Timeout: 1h (from project)
Runner: #32976687 (nN8vMRS9) 6-blue.saas-linux-small-amd64.runners-manager.gitlab.com/default
Commit a8cfbfb1
Add new yml file
Pipeline #1724662309 Passed for main
test
Related jobs
test_job

d. Commit and push changes to trigger the pipeline.

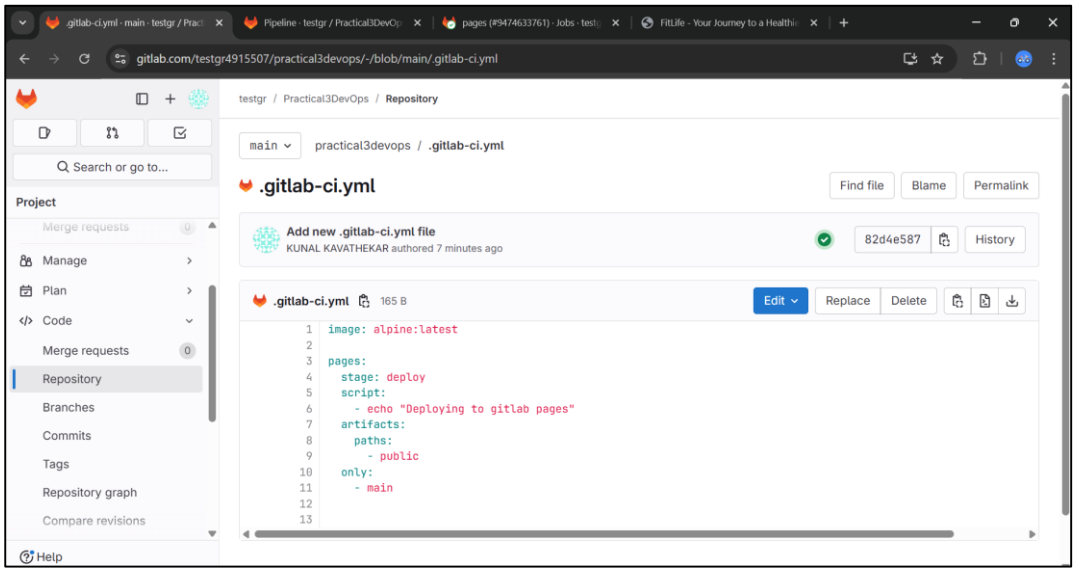


- Deploying a Static Website Using GitLab Pages – A company wants to host a simple HTML website on GitLab Pages.

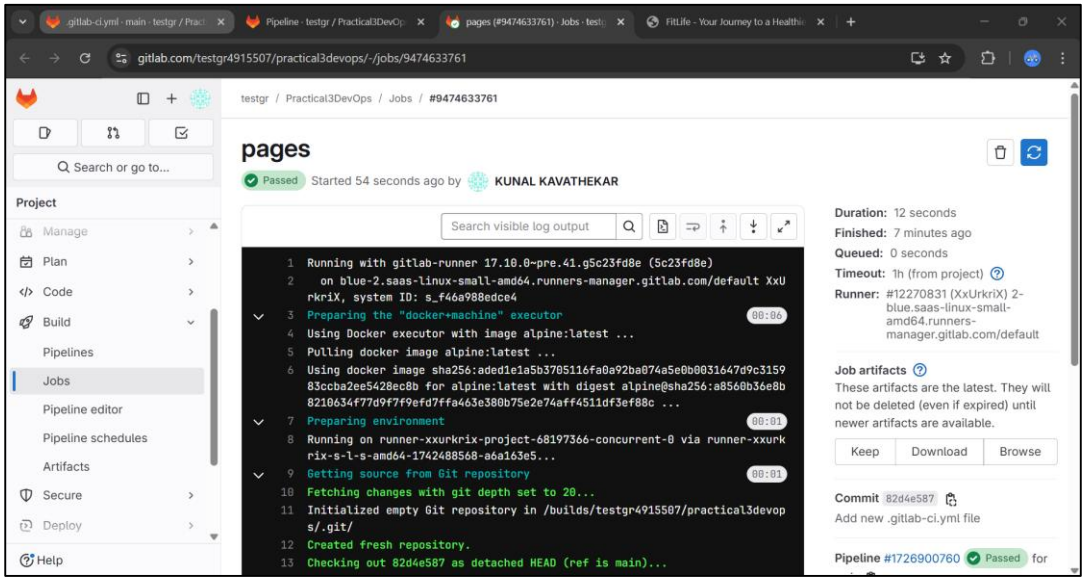
a. Create an HTML/CSS website and push it to GitLab.

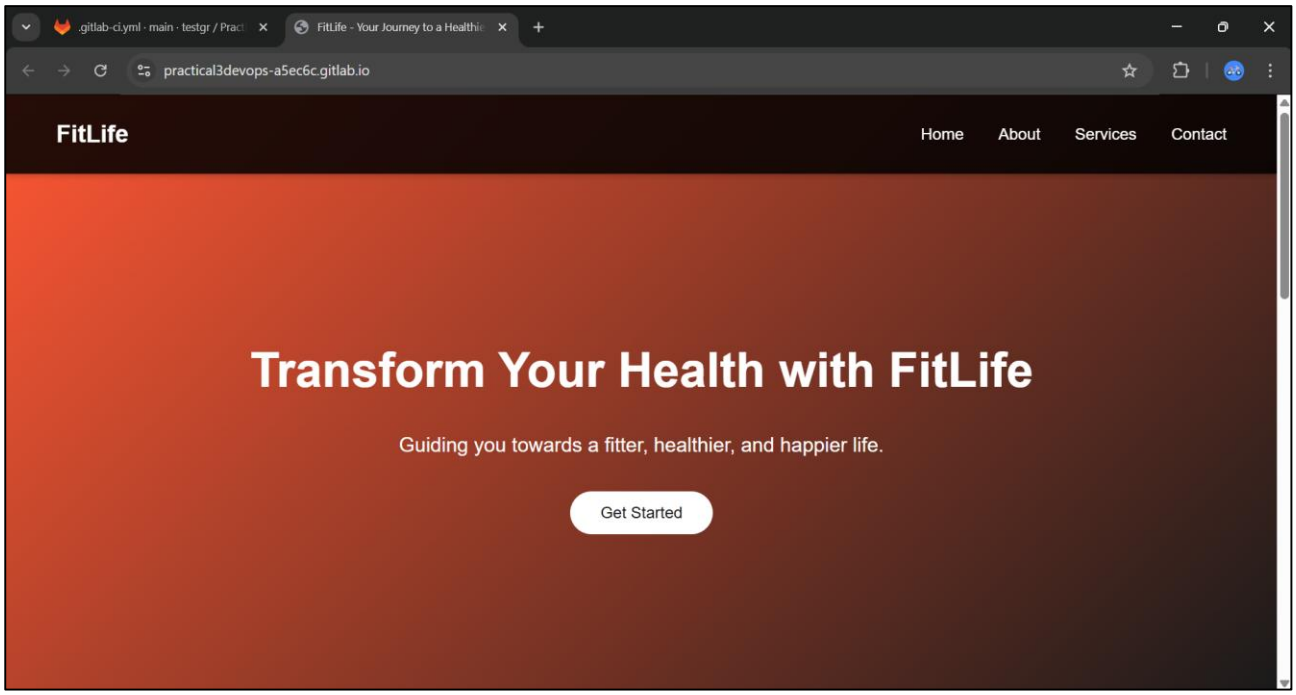
```
Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3Q3/practical3devops (main)
$ mkdir public

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3Q3/practical3devops (main)
$ cd public

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3Q3/practical3devops/public (main)
$ touch index.html

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3Q3/practical3devops/public (main)
$ vi index.html

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3Q3/practical3devops/public (main)
$ cd ..

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3Q3/practical3devops (main)
$ git add public/

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3Q3/practical3devops (main)
$ git commit -m "Added new public folder"
[main (root-commit) 2d4a4a1] Added new public folder
 1 file changed, 198 insertions(+)
 create mode 100644 public/index.html

Admin@Kunalsk36-Laptop MINGW64 /d/MCAsemII/DevOps/Practical3Q3/practical3devops (main)
$ git push origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 16 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 1.88 KiB | 1.88 MiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://gitlab.com/testgr4915507/practical3devops.git
 * [new branch]      main -> main
```

b. Define `.gitlab-ci.yml` to generate and deploy the site using GitLab Pages.



c. Test the deployed website from the provided GitLab Pages URL.





d. Configure a scheduled pipeline for this web site.

e.