



Symbiosis Institute of Technology, Pune  
Faculty of Engineering  
CSE- Academic Year 2025-26  
DevOps Lab -4 Batch 2022-26

Name of Student	Praneet Mahendrakar
PRN No.	22070122155
Batch	2022-26
Class	DevOpsL-4
Academic Year & Semester	2025-26 Sem-7
Date of Performance	08/08/25

## DevOps Lab Assessment Questions

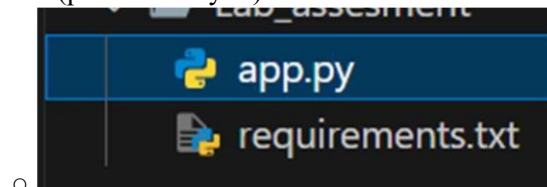
### 1. TW1 - Continuous Evaluation: Foundational DevOps Skills (10 Marks)

Instructions: Complete the following tasks in your lab environment. Document your commands and outputs as evidence.

Scenario: You are a developer working on a new feature for a web application.

#### 1. Git Workflow & Collaboration (4 Marks)

- Task 1.1: Initialize a new Git repository for a simple "Hello World" Python Flask application (provided to you). Add the initial code and commit it to the main branch. (1 Mark)



```

Mode                               LastWriteTime         Length Name
----                               -
-a----          12-08-2025      10:49           238 app.py
-a----          12-08-2025      10:55           16 requirements.txt
On branch main
Your branch is up to date with 'origin/main'.

* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://192.168.29.112:5000
Press CTRL+C to quit
127.0.0.1 - - [12/Aug/2025 11:00:11] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [12/Aug/2025 11:00:12] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [12/Aug/2025 11:00:50] "GET /favicon.ico HTTP/1.1" 404 -

```

- Task 1.2: Create a new branch named feature/user-auth. Make a small modification (e.g., add a new print statement) in this branch, commit the changes, and push the branch to a remote repository (e.g., GitHub/GitLab). (1.5 Marks)

```

PS C:\Users\kharw\OneDrive\Desktop\DevOps_CA\Lab_assesment> # Create and switch to the new branch
>> git checkout -b feature/user-auth
>>
Switched to a new branch 'feature/user-auth'
PS C:\Users\kharw\OneDrive\Desktop\DevOps_CA\Lab_assesment> # Stage the change
>> git add app.py
>>
>> # Commit
>> git commit -m "Added placeholder for user authentication"
>>
[feature/user-auth 1c79b9a] Added placeholder for user authentication
git push -u origin feature/user-auth(+)
>> C:\Users\kharw\OneDrive\Desktop\DevOps_CA\Lab_assesment>
branch 'feature/user-auth' set up to track 'origin/feature/user-auth'.
Everything up-to-date

```

○

- Task 1.3: Simulate a conflict: On the main branch, modify the same line you changed in feature/user-auth. Commit this change. Now, attempt to merge feature/user-auth into main. Resolve the merge conflict manually and commit the resolution. Push the updated main branch. (1.5 Marks)

## App.py

```

lab_assesment > app.py > ...
1  from flask import Flask
2  app = Flask(__name__)
3
4  @app.route('/')
5  def hello_world():
6      # This is the line we will modify later
7      <<<<<<< HEAD (Current Change)
8      ~~~~~~ return 'Hello, World from MAIN branch edit!'
9
10 =====
11 ~~~~~~ print("User authentication module placeholder")
12
13         return 'Hello, World from main!'
14 >>>>>> feature/user-auth (Incoming Change)
15
16 if __name__ == '__main__':
17     app.run(host='0.0.0.0', port=5000)

```

Resolve in Merge Editor

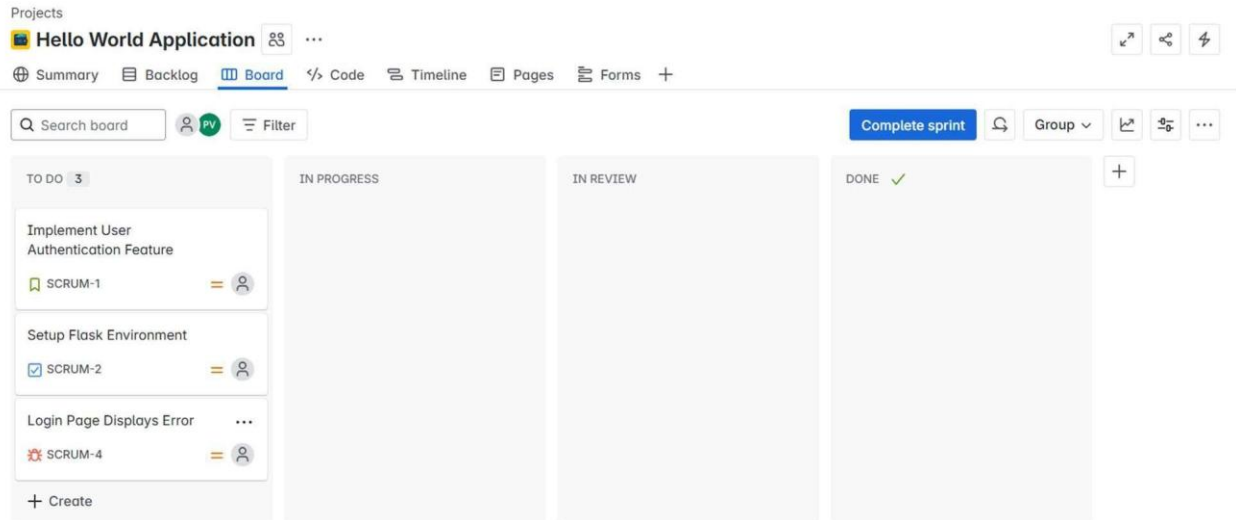
```

PS C:\Users\kharw\OneDrive\Desktop\DevOps_CA\Lab_assesment> git add app.py
>> git commit -m "Edited return line in main branch to create merge conflict"
>>
[main b48f6b3] Edited return line in main branch to create merge conflict
1 file changed, 2 insertions(+), 1 deletion(-)
PS C:\Users\kharw\OneDrive\Desktop\DevOps_CA\Lab_assesment> git merge feature/user-auth
>>
Auto-merging app.py
CONFLICT (content): Merge conflict in app.py
PS C:\Users\kharw\OneDrive\Desktop\DevOps_CA\Lab_assesment> git add app.py
>> git commit -m "Resolved merge conflict between main and feature/user-auth"
>> C:\Users\kharw\OneDrive\Desktop\DevOps_CA\Lab_assesment>
[main 461176e] Resolved merge conflict between main and feature/user-auth
PS C:\Users\kharw\OneDrive\Desktop\DevOps_CA\Lab_assesment> git push origin main
>>
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 583 bytes | 583.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/harshkharwar1204/Lab_assesment.git
   bad3cf6..461176e  main -> main
PS C:\Users\kharw\OneDrive\Desktop\DevOps_CA\Lab_assesment>

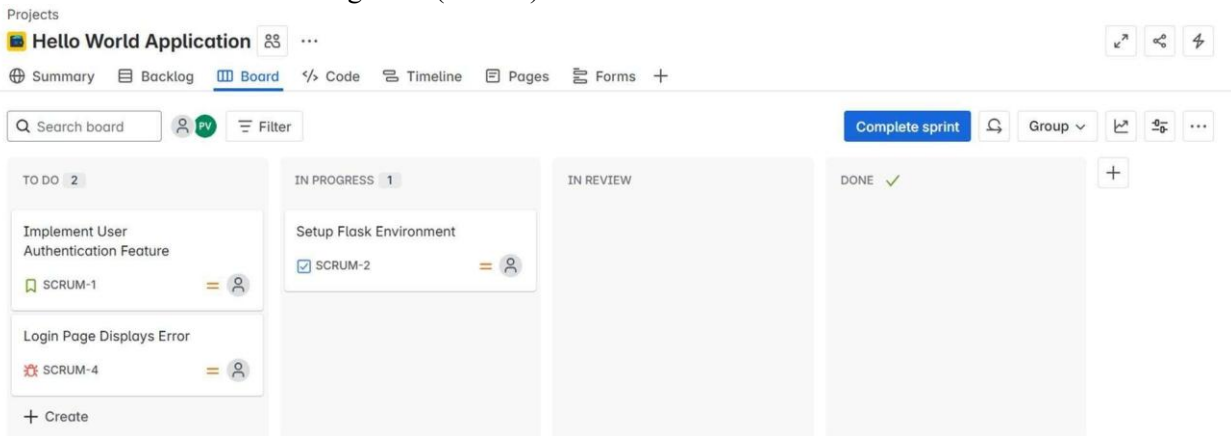
```

## 2. Jira Project & Issue Tracking (3 Marks)

- Task 2.1: Log in to your free Jira Cloud instance. Create a new "Scrum" project for the "Hello World" application. (1 Mark)
- Task 2.2: Create the following issues within your new project:
  - One "Story" titled "Implement User Authentication Feature".
  - One "Task" titled "Setup Flask Environment".
  - One "Bug" titled "Login Page Displays Error". (1 Mark)

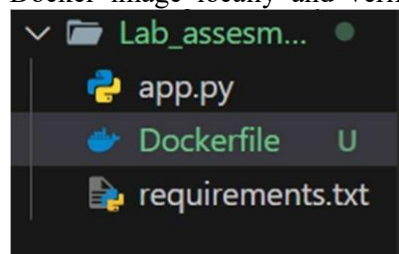


- Task 2.3: Navigate to the project's board. Move the "Setup Flask Environment" task from "To Do" to "In Progress". (1 Mark)



### 3. Basic Containerization (Docker) & Jenkins Freestyle Project (3 Marks)

- Task 3.1: Create a Dockerfile for your Python Flask "Hello World" application. The Dockerfile should build a minimal image that runs the application on port 5000. Build the Docker image locally and verify it runs correctly. (1.5 Marks)



Dockerfile



```
Lab_assesment > Dockerfile > ...
1  # Use minimal Python base image
2  FROM python:3.9-slim
3
4  # Set working directory
5  WORKDIR /app
6
7  # Copy requirements.txt and install dependencies
8  COPY requirements.txt requirements.txt
9  RUN pip install --no-cache-dir -r requirements.txt
10
11 # Copy application code
12 COPY . .
13
14 # Expose Flask default port
15 EXPOSE 5000
16
17 # Run the application
18 CMD ["python", "app.py"]
19
>> C:\Users\kharw\OneDrive\Desktop\DevOps_CA\Lab_assesment>
[+] Building 14.8s (11/11) FINISHED          docker:desktop-linux
=> [internal] load build definition from Dockerfile      0.1s
=> => transferring dockerfile: 413B                      0.0s
=> [internal] load metadata for docker.io/library/python:3 3.4s
=> [auth] library/python:pull token for registry-1.docker. 0.0s
=> [internal] load .dockerignore                        0.0s
>>
* Serving Flask app 'app'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment
production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://172.17.0.2:5000
Press CTRL+C to quit
172.17.0.1 - - [12/Aug/2025 05:58:49] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Aug/2025 05:58:50] "GET /favicon.ico HTTP/1.1" 404 -
172.17.0.1 - - [12/Aug/2025 05:59:15] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [12/Aug/2025 05:59:16] "GET /favicon.ico HTTP/1.1" 404 -
```

Task 3.2: Set up a Jenkins Freestyle project. Configure it to pull your Git repository (from Task 1.1) and execute a build step that lists the contents of the workspace. Trigger a build and provide a screenshot of the successful build output. (1.5 Marks)

<input type="checkbox"/>	Name	Container ID	Image	Port(s)	CPU (%)	Last started	Actions
<input type="checkbox"/>	jenkins	1f544d34803b	<a href="#">jenkins/jenkins:lts</a>	<a href="#">55000:55000</a> <a href="#">8080:8080</a> <a href="#">Show less</a>	0.42%	2 minutes ago	   

Dashboard > Devops-Lab-assesment > #1 > Console Output

```

Started by user harshkharwar1204
Running as SYSTEM
Building in workspace /var/jenkins_home/workspace/Devops-Lab-assesment
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/harshkharwar1204/Lab_assesment
  > git init /var/jenkins_home/workspace/Devops-Lab-assesment # timeout=10
Fetching upstream changes from https://github.com/harshkharwar1204/Lab_assesment
  > git --version # timeout=10
  > git --version # 'git version 2.39.5'
  > git fetch --tags --force --progress --
https://github.com/harshkharwar1204/Lab_assesment
+refs/heads/*:refs/remotes/origin/* # timeout=10
  > git config remote.origin.url https://github.com/harshkharwar1204/Lab_assesment #
timeout=10
  > git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* #
timeout=10
Avoid second fetch
  > git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 461176e3b2d214a1032a0ec6a5270fbb2514b170
(refs/remotes/origin/main)
  > git config core.sparsecheckout # timeout=10
  > git checkout -f 461176e3b2d214a1032a0ec6a5270fbb2514b170 # timeout=10
Commit message: "Resolved merge conflict between main and feature/user-auth"
First time build. Skipping changelog.
[Devops-Lab-assesment] $ /bin/sh -xe /tmp/jenkins11332081280038327871.sh
+ echo Listing workspace contents:
Listing workspace contents:
+ dir
app.py  requirements.txt
Finished: SUCCESS

```

