

DevOps Intern Assessment: Directus Deployment & CI/CD Pipeline

Name: Darshan Patgar

Gmail: darshanpatgar230@gmail.com

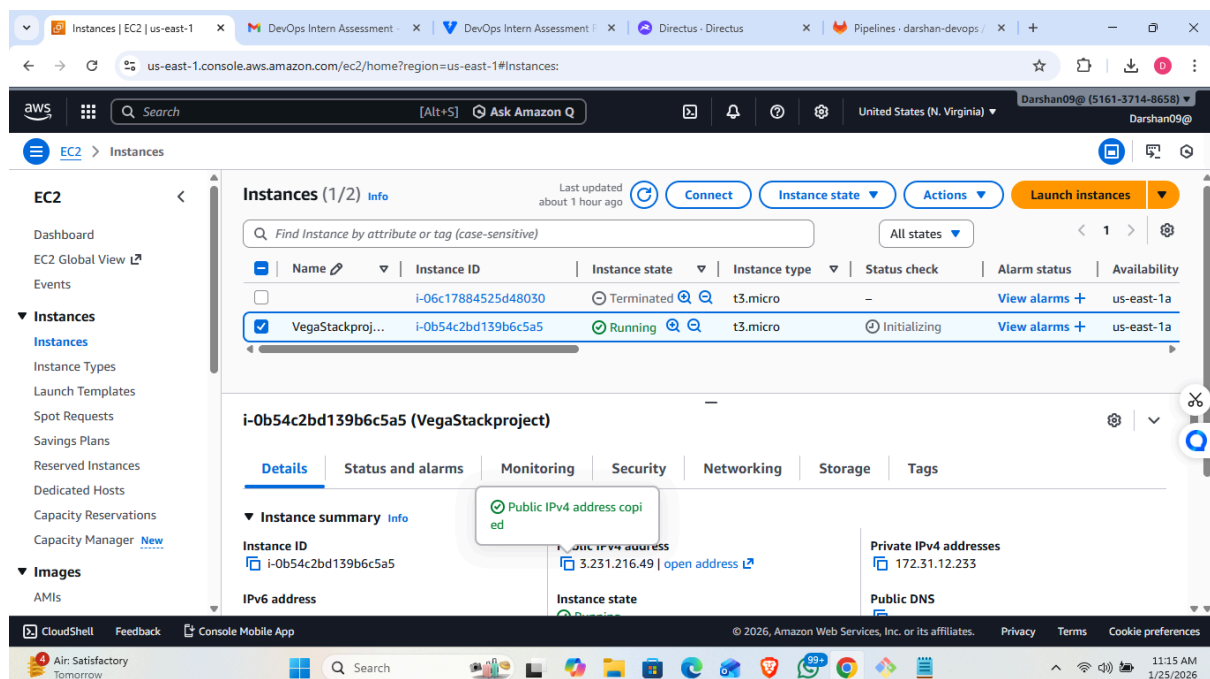
Target URL: <http://3.231.216.49>

GitLab Repository: <https://gitlab.com/darshan-devops/directus-devops-assessment>

1. Infrastructure Setup

For this assessment, I provisioned a Virtual Machine on **AWS EC2** using the **t3.micro** instance type.

- **Operating System:** Ubuntu 22.04 LTS.
- **Security Configuration:** I configured the AWS Security Group to allow inbound traffic on **Port 22** (SSH for management) and **Port 80** (HTTP for public access to the Directus site).



2. Directus Deployment using Docker

I utilized **Docker** and **Docker Compose** to ensure a portable and reliable deployment.

- **Services:** The deployment consists of a **directus** service and a **postgres** database service.

- **Networking:** I mapped the internal Directus port 8055 to the host's port 80 to make the application accessible at the root IP address.
- **Persistence:** Docker volumes were implemented to ensure data remains persistent across container restarts.

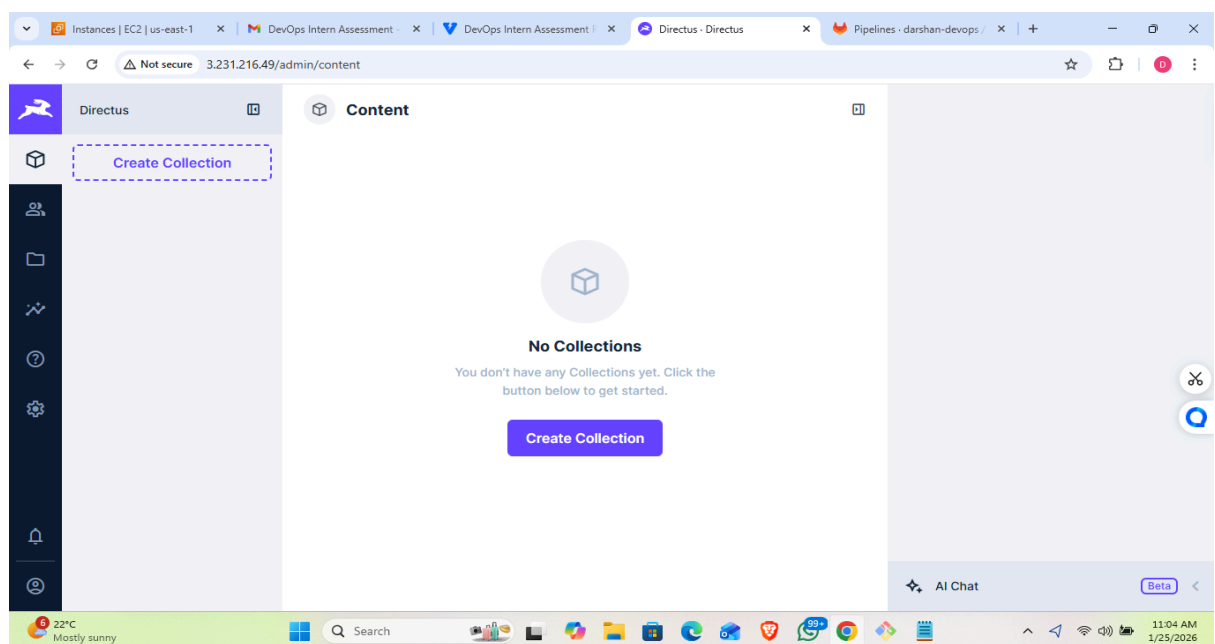
```

ubuntu@ip-172-31-12-233: ~/directus-project
ubuntu@ip-172-31-12-233:~$ docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED    STATUS    PORTS    NAMES
ubuntu@ip-172-31-12-233:~$ mkdir directus-project && cd directus-project
ubuntu@ip-172-31-12-233:~/directus-project$ ls
ubuntu@ip-172-31-12-233:~/directus-project$ sudo -i
root@ip-172-31-12-233:~# ls
snap
root@ip-172-31-12-233:~# exit
logout
ubuntu@ip-172-31-12-233:~/directus-project$ ls
ubuntu@ip-172-31-12-233:~/directus-project$ vim docker-compose.yml
ubuntu@ip-172-31-12-233:~/directus-project$ sudo docker compose up -d
WARN[0000] /home/ubuntu/directus-project/docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] up 30/30
✔ Image directus/directus:latest      Pulled          25.6s
✔ Image postgres:15                  Pulled          14.3s
✔ Network directus-project_default    Created         0.1s
✔ Container directus-project-database-1 Created         0.2s
✔ Container directus-project-directus-1 Created         0.1s
ubuntu@ip-172-31-12-233:~/directus-project$ docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED    STATUS    PORTS    NAMES
d916ee3bdd31   directus/directus:latest "docker-entrypoint.s..." 8 seconds ago Up 7 seconds 0.0.0.0:80->8055/tcp, [::]:80->8055/tcp directus-project-directus-1
fc96af367759   postgres:15 "docker-entrypoint.s..." 8 seconds ago Up 8 seconds 5432/tcp   directus-project-database-1
ubuntu@ip-172-31-12-233:~/directus-project$

```

3. Web Verification

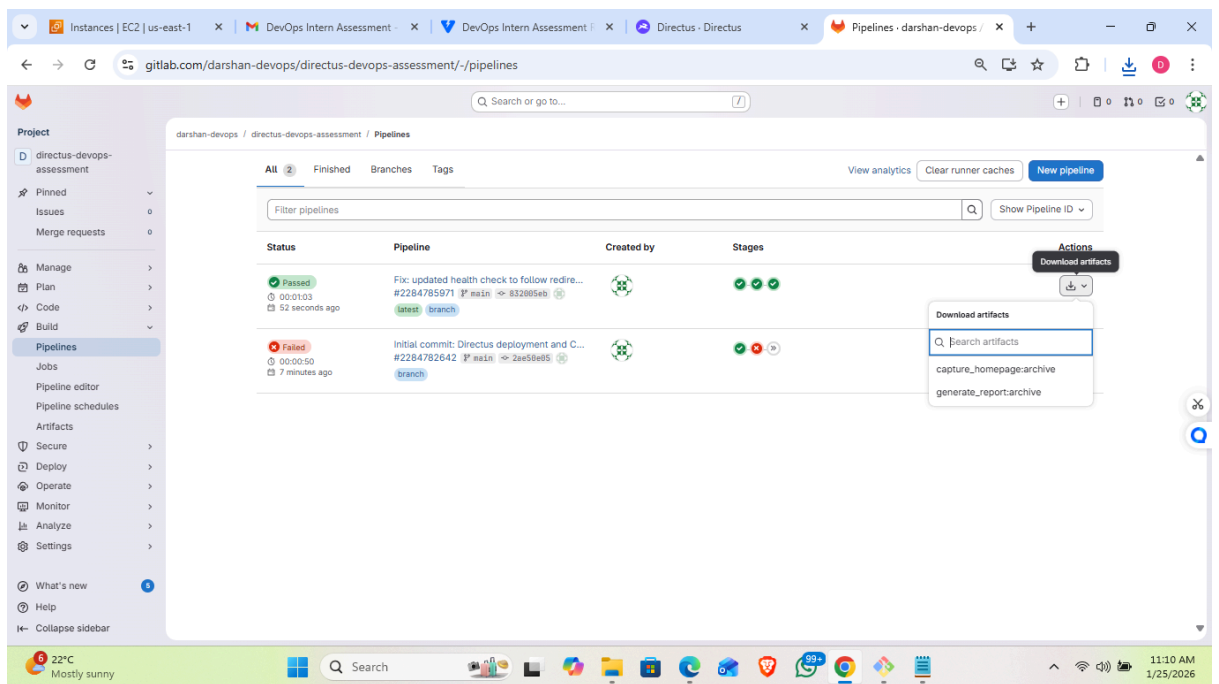
The Directus site was successfully deployed and is publicly accessible. I verified the administrative login and accessed the CMS dashboard.



4. GitLab CI/CD Pipeline Implementation

I created a robust CI/CD pipeline in GitLab to automate the validation of the deployment.

- **Validate Stage:** Uses `docker compose config` to check the syntax of the configuration files.
- **Test Stage:** * `health_check`: Uses `curl` to verify the site is online and responding correctly (handling 200 and 302 status codes).
 - `capture_homepage`: Automatically downloads the homepage HTML.
- **Report Stage:** Generates a `deployment_report.md` as a final deliverable.



5. Challenges & Solutions

During the pipeline setup, the initial `health_check` failed because Directus was issuing a 302 Redirect to the login page.

- **Solution:** I utilized the `vim` editor on the EC2 instance to modify the `.gitlab-ci.yml` file. I updated the health check to follow redirects (`-L`) and validated that the response code was either 200 or 302, ensuring the pipeline correctly identifies a healthy, reachable application.
-