DevOps Intern Assessment R1

Important Note
At the end of this page, you will find an **"Acknowledge Assessment"** section. Please make sure to submit your acknowledgment after you have finished reading through all the assessment details. This confirms that you have reviewed the requirements and are ready to begin the assessment task.

Introduction

The assessment task is to deploy Directus (a headless CMS) on a VM hosted in AWS/DigitalOcean using Docker and create a GitLab CI/CD pipeline to validate the deployment.
What is Directus?
What is GitLab (it's not GitHub!)?

Assessment Details

Overview

1. Use AWS/DigitalOcean or any cloud provider of your choice (you can choose free tier)
2. Deploy Directus site using Docker in a Virtual Machine
3. Make Directus site accessible via HTTP publicly
4. Create a GitLab CI/CD pipeline to curl the Directus site public homepage
5. Document your task implementation

Required Task Deliverables

#	Deliverable	Description
1	**Working Directus Site**	Accessible via http://Your-VM-Server-IP
2	**GitLab Repository**	Contains all your code and config files
3	**GitLab CI/CD Pipeline**	Automated pipeline that tests the deployment
4	Implementation doc	Google Doc explaining your implementation process

Assessment Steps (6)

Your task is to deploy Directus (a headless CMS) on a VM in AWS/DigitalOcean or any other cloud provider of your choice using Docker, and set up a CI/CD pipeline to validate the deployment.

Step 1: Create Free Account on AWS or DigitalOcean

Feel free to use any cloud service provider of your choice.
Please do not terminate the instance, until you receive a confirmation on your application review from our team.

Steps to create Free Account in AWS

Steps to create Free Account in DigitalOcean

Step 1 Complete!
By the end of this step you have successfully completed creating a free account on AWS/DigitalOcean.
Next, you are ready to launch a new VM to install Directus.

Step 2: Install and Deploy Directus

Help and References

Directus Docs Self Hosting | Directus Docs
- choose the installation based on your choice of cloud provider

Deploy Directus on a Virtual Machine server using docker-compose with a PostgreSQL database and an S3 bucket for storage.
Follow the steps in the

Self Hosting | Directus Docs
and complete all installation steps.

Tasks to complete in this step:

1. Install Docker and Docker Compose on the virtual machine
2. Create project structure with proper directories
3. Configure environment variables securely
4. Create docker-compose.yml for Directus and PostgreSQL
5. Launch docker containers and verify deployment
6. Make sure that the Directus site is publicly accessible by forwarding the Directus server port publicly on http://Your-VM-Server-IP VM's public IP address.

• You may have to modify the VM's firewall on AWS/DigitalOcean to expose the port and access the Directus site publicly.

7. After accessing the Directus site at http://Your-VM-Server-IP in your browser, verify login by accessing the Admin panel with the credentials you set in the Docker environment variables.

• ADMIN_EMAIL and ADMIN_PASSWORD are the Directus Admin login credentials. Their values are set in the Docker environment configuration (Dockerfile or docker-compose.yml).

Step 2 Complete!

By the end of this step you have successfully deployed Directus using Docker on your VM and verified it's publicly accessible in your browser.
Next, you'll create a GitLab account to set up Git version control and CI/CD.

Step 3: Create GitLab Account

1. Visit GitLab Signup page directly or go to gitlab.com and click "Sign Up".
2. Enter your first name, last name, a unique username (check availability), email address, and a strong password (at least 8 characters). Complete any CAPTCHA, then click "Register" or "Continue".
3. Check your email for a verification code or link from GitLab, enter it on the site, and click "Verify email address".
4. Answer optional questions about your role or skip them. Optionally, personalize your profile (add bio, photo) via the avatar menu > Edit Profile. Your dashboard appears, ready for creating projects.

Step 3 Complete!

By the end of this step you have successfully created your GitLab account and are ready to create a public project to host your code and pipeline.

Step 4: Create a Public GitLab Project

1. On the top bar, click Menu → Projects → Create new project (or the ”New project” button on your dashboard).
2. Choose ”Create blank project”.
3. In Project name, type the name you want (for example: directus-project).
4. Confirm the Project slug (URL path), it auto-fills from the project name and usually doesn’t need changes.
5. In the "Visibility Level" section on the same form, select Public.
6. Once the project is created and opened, look near the project name, you should see an indicator showing Public.
7. 
8.
9. Once the project is created, you need to add the necessary configuration files to the GitLab project root directory.

- **docker-compose.yml** - Copy the Docker Compose configuration file you used for your Directus deployment on the VM. This file defines your Directus and PostgreSQL services and will be used by the validate_config job in the pipeline.
- **.gitlab-ci.yml** - Create an empty file for adding the CI/CD pipeline configuration that defines all the stages and jobs that you’ll add in Step 5. This is the main file that GitLab reads to execute your pipeline automatically.
- These files are essential for the pipeline to function correctly.

Step 4 Complete!

By the end of this step you have successfully created a public GitLab project and added the necessary files to the project root directory.
Next, you’ll set up your GitLab CI/CD pipeline to automate deployment validation.

Step 5: Setting Up CI/CD Pipeline

Goal of the CI/CD pipeline

In this CI/CD pipeline, we verify that Directus site created in

DevOps Intern Assessment R1 - Step 2: Install and Deploy Directus
is publicly accessible by using the curl command to download the Directus homepage HTML. This simple HTTP request confirms that:

- The Directus server is running and responding to requests

- The deployment is accessible from outside the VM (publicly available)
- The web service is serving content correctly

By successfully downloading the homepage HTML file, we validate that the entire deployment stack — from Docker containers to network configuration — is working as expected and can be reached by external users.

CI/CD Pipeline Requirements

Create a CI/CD configuration file named .gitlab-ci.yml, following the syntax of the GitLab CI YAML files:

CI/CD YAML syntax reference | GitLab Docs

.

You'll have access to GitLab Shared Runners, which are provided by GitLab to execute your CI/CD pipelines. The Shared Runners automatically pick up your pipeline jobs and run them in isolated environments, so you don't need to set up or maintain your own runners.

Pipeline Stages Overview:

Stage Name	Stage requirements	**Jobs**
Validate	Check configuration files	validate_config
Test	Run health checks on deployment	health_checkcapture_homepage
Report	Generate deployment report	generate_report

The CI/CD pipeline consists of three stages that run sequentially:

- **Validate Stage:** Checks the syntax and validity of the docker-compose.yml file using the docker compose config command to validate configuration syntax.
- **Test Stage:** Runs curl health checks on the deployed Directus application to verify it's accessible and functioning correctly, and captures the homepage HTML
- **Report Stage:** Generates a comprehensive deployment summary report documenting the results of the validation and testing stages

Pipeline Jobs Details:

Review the job requirements below carefully and implement them accordingly.

Job Name	Stage	Job requirements	Artifact
validate_config	validate	Validate docker-compose.yml syntax	NA
health_check	test	Test if Directus responds via HTTP	NA
capture_homepage	test	Download and save homepage HTML as an artifact	directus_homepage.html
generate_report	report	Create deployment summary	deployment_report.md

Each stage contains specific jobs with defined requirements:

- **validate_config** (Validate stage): Validates the docker-compose.yml file syntax and produces a configuration report as an artifact
- **health_check** (Test stage): Tests HTTP connectivity to verify the Directus site responds correctly and saves health check results as an artifact
- **capture_homepage** (Test stage): Downloads the Directus homepage HTML using curl and saves it as directus_homepage.html artifact
- **generate_report** (Report stage): Creates a deployment summary in Markdown format and saves it as deployment_report.md artifact

Help and References

1. GitLab Runners:
2. 
3. GitLab Runner | GitLab Docs
4. cURL command to download a site as HTML file:
5. 
6. Nono Martínez Alonso Download a Website with cURL Following Redirects
7. Bash CI/CD Pipeline Template:
8. 
9. GitLab lib/gitlab/ci/templates/Bash.gitlab-ci.yml · master · GitLab.org / GitLab · GitLab

10. Setting up Pipeline Artifacts in GitLab:
11. 
12. Job artifacts | GitLab Docs

GitLab CI/CD FAQs
How to use GitLab CI/CD?
What is the official GitLab YAML file syntax reference for creating .gitlab-ci.yml file?
How to validate the .gitlab-ci.yml file syntax?
Tutorial for setting up a CI/CD pipeline using GitLab CI/CD?
What to do if you encounter any issues while working on the Assessment?

Step 5 Complete!
By the end of this step you have successfully configured a GitLab CI/CD pipeline with validation, testing, and reporting stages.
Now it's time to document your entire implementation process.

Step 6: Create Implementation Doc
Create a Google Doc explaining all the steps you took to complete the assessment.
Include the following details:
1. Steps taken to install Directus in a VM and make it publicly accessible
2. GitLab pipeline setup process and implementation
3. Screenshots of deployed Directus site homepage, and successful GitLab pipeline run.

Additionally, you can include information about different challenges you encountered while working on the assessment and how you overcame them.

Make Your Google Doc Public
Please ensure your implementation doc is accessible to us by making it public:
Open your Google Doc → Click the **Share** button in the top-right corner → Under "General access," click **Change** → Select **Anyone with the link** → Set the permission to **Viewer** → Click **Done**

Step 6 Complete!

By the end of this step you have successfully documented your entire implementation process.
You're now ready to submit your assessment for evaluation!

Common Issues

If you encounter problems, refer to these resources:

Deployment issues

CI/CD issues

Research strategy recommendations

Assessment Submission

Assessment Submission Checklist

1. Ensure your Directus site is available and running at http://Your-VM-Server-IP.
2. Can log in to the Directus site with Directus Admin credentials.
3. The GitLab CI/CD pipeline is configured and it should run successfully by executing the curl command to download the HTML file, and saving it as an artifact.
4. Share the public URL of your GitLab Project, so we can access it.

How to submit the Assessment for evaluation?

The first email we sent you for the Assessment details includes a link to the Assessment Form.

Open the Assessment Form link, fill in all the required details accurately, and submit it for evaluation.

Asking Queries?

If you have any questions during the Assessment, please reply to the Assessment email and provide a detailed description of your query. We will make every effort to address them promptly.

When asking questions, please include:

- What you're trying to accomplish

- What you've already tried (at least 3 attempts)
- Exact error messages
- Relevant log excerpts
- Your environment details (Cloud platform, Ubuntu version, Docker version, etc.)

Good Luck!
- **Start early** - don't wait until the deadline
- **Document everything** - your implementation process matters
- **Have fun** - this is your chance to learn something new!

We're excited to see what you build!

Acknowledge Assessment

Please acknowledge that you have read and understood the assessment details and are beginning the assessment.