Technical Test for DevOps Engineer Role (1-2 Days)

This technical test is designed to assess your skills in various DevOps areas relevant to our role at Magicfeedback. It can be completed within 1-2 days, depending on your experience level.

Scenario: We are building a customer feedback application with a Node.js or Python API running on Google Cloud Run. This test will focus on automating the deployment and management of this application.

Part 1: Infrastructure Automation with Terraform (60 minutes)

- **Task:** Using Terraform configuration files, define the infrastructure for the application, **primarily deploying the application to Google Cloud Run**.
  - A VPC network with a subnet.
  - A firewall rule to allow HTTP traffic.
  - Configuration for deploying the chosen backend API (Node.js or Python) to Google Cloud Run.
- **Bonus:** Demonstrate your knowledge by including a Dockerfile for containerizing the application specifically for deployment on Cloud Run.

Part 2: CI/CD Pipeline with GitLab CI/CD (60 minutes)

- **Task:** Define a `.gitlab-ci.yml` file that includes the following stages:
  - **Build:** Installs dependencies and builds the chosen backend API (Node.js or Python).
  - **Test:** Runs unit tests for the application. (You can provide a sample test script)
  - **Deploy:** Uses Terraform to deploy the infrastructure and application to a designated GCP environment (staging or production).

Part 3: Collaboration and Continuous Improvement (30 minutes)

- **Scenario:** The application deployment was successful! However, we'd like your input on how to improve the overall process and what next steps you would recommend.
- **Task:**
  - Identify areas for improvement in terms of scalability, security, or efficiency within the Cloud Run environment.
  - Propose specific next steps you would take to enhance the DevOps practices for this application, considering Cloud Run's functionalities for scaling.
    - How would you approach vertical scaling of the application running on Cloud Run?
    - How would you design a horizontal scaling strategy for the application on Cloud Run to handle increased traffic?

Additional Information:

- The API should have an endpoint (URL) that accepts POST requests to collect customer feedback data. The request body should be a JSON object containing relevant information like customer ID, feedback message, and rating.
- The delivery part of the application involves integrating the API with our frontend application. This is outside the scope of this test, but consider mentioning how your solution would facilitate this integration.

Evaluation Criteria:

- Functionality and completeness of the Terraform configuration.
- Correctness and efficiency of the CI/CD pipeline using GitLab CI/CD.
- Problem-solving approach and identification of improvement opportunities.
- Communication and explanation of proposed solutions and next steps.
- Understanding of the API endpoint definition and delivery considerations.
- Code quality and adherence to best practices. (For Terraform & GitLab CI/CD configuration files)

Additional Notes:

- You can use any additional resources you find helpful for completing the tasks (e.g., Terraform documentation, GitLab CI/CD documentation).
- This test is designed to assess your practical skills. We encourage you to ask clarifying questions during the test.
- We may adjust the difficulty of the scenarios based on your experience level.

Bonus Points:

- Demonstrate your experience with containerization technologies (Docker, Kubernetes) by incorporating them into your solutions (e.g., building a Dockerfile for Cloud Run deployment).