DevOps Coding Challenge

Steps and Requirements:

1. **Cloud Provider Setup:**
    - Create a test account on one of the three major cloud providers (AWS, Azure, GCP).
2. **Kubernetes Cluster:**
    - Set up a Kubernetes cluster on the chosen cloud provider. This can be done using managed Kubernetes services like EKS (AWS), AKS (Azure), or GKE (GCP).
    - Ensure the cluster is configured for high availability and can scale based on load.
3. **Infrastructure as Code (optional):**
    - Use an Infrastructure as Code (IaC) tool like Terraform, Pulumi, or CloudFormation (for AWS) to automate the deployment of the infrastructure.
    - Provide the IaC scripts along with documentation on how to use them.
4. **Application:**
    - Deploy a simple CORS proxy. You can use any existing solutions available.
    - The service should handle HTTP requests, add the appropriate CORS headers, and forward the requests to the target server.
5. **Scalability:**
    - Configure the Kubernetes cluster to automatically scale the CORS proxy service based on CPU/memory usage or request count.
    - Ensure that the service can handle up to 1000 requests per second.
6. **Load Testing (optional):**
    - Optionally perform load testing on the service to verify its scalability.
    - Use tools like Apache JMeter, Locust, or k6 to simulate the load.
    - Provide the load testing scripts and results.
7. **Documentation and Deliverables:**
    - Detailed documentation on the entire process, including:
        - Instructions for deploying the Kubernetes cluster using the IaC tool.
        - Steps to deploy and scale the service on Kubernetes.
        - Load testing methodology and results (if performed).
        - What are the limitations of the resulting setup?
        - What steps would you have to take so it can scale even further? E.g. 10k, 100k req/s.
    - All source code and configuration files used in the challenge.

Submit the result in the most suitable way in your opinion.

As a benchmark this should take you 3-6h.