L. I.

DevOps Engineer

4+ years of experience

DevOps Engineer with extensive technical experience in automating application deployments in cloud (AWS, GCP), containerizing applications, building CI/CD pipelines. Fast Learner, ambitious, self-motivated with ability to adapt to new environments quickly. Always eager to learn something new. Always looking for ways to bring something special to the team and company.

Technical capabilities

Cloud	AWS (2+ years), GCP (4 years)
IaaC	Terraform (4+ years), CloudFormation(2+ year)
Containerization	Docker(2+ year) Kubernetes (4+ years)
CI&CD	Jenkins, CircleCi, GitHub Actions, GitLab CI(2+year)
Monitoring	Prometheus&Grafana(4+years), Datadog(2+year), Cloud Stackdriver(2+ years), AWS CloudWatch (2+ years), Opsgenie, PagerDuty, Slack
Secret Management	Cloud SecretManager, AWS SecretManager (4 years)
Database	GCP: Cloud SQL(Postgres, MySQL, MsSQL), Spanner, Bigtable, Firestore. AWS: RDS(Postgres, MySQL, MSSQL)
Web Development	HTML, CSS, JS (1 year)
Scripting	Python, Bash (2+ years)
Operating Systems	Ubuntu, RHEL, CentOS 7/8, Amazon linux (4+ years)

Education

Kyrgyz National Technical University named by I.Razzako - Computer Science, Bachelor

Certifications

★ **Google Cloud Certified Professional Cloud Engineer**

Selected Projects

Monitoring the Couchbase DB with a Prometheus via the Couchbase Exporter

Role	DevOps Engineer
Technologies	Kubernetes, Helm, EKS, Docker, ECR, Secret Manager, GitLab CI
Description	Monitoring Couchbase DB with Prometheus via the Couchbase Exporter

Responsibilities	• Building a custom image and pushing it to ECR • Writing a custom Helm chart for a couchbase-exporter • Deploying it to the EKS cluster • Integrating it with Prometheus, Alertmanager and Grafana • Building a dashboard for a Couchbase DB • Creating alerts and configuring notification channels

Built an image with AWS Image Builder and established encryption with AWS KMS

Role	DevOps Engineer
Technologies	AWS Image Builder, AWS KMS, Terraform, CloudFormation StackSet, Bash scripting, IAM, CI/CD
Description	Build a new image and encrypt it with AWS KMS
Responsibilities	• Building a new image with Image Builder + Terraform • Replicating it across regions and accounts • Configuring CI/CD pipeline for Terraform code • Creating KMS key and spreading it across the accounts and regions with a CloudFormation StackeSet • Managing the key policy and rotation

Building cloud Infrastructure with IaC (Terraform, CloudFormation)

Role	DevOps Engineer
Technologies	Terraform, GCP resources, AWS services, CloudFormation
Description	Building Infrastructure in AWS (VPC, RDS, EKS, ECS, SNS, Lambda, CloudWatch etc) and GCP(GKE, VPC, Cloud SQL, NoSQL(Firestore), Cloud Functions, Cloud Scheduler, GCS, GCE, MIG, Load Balancer, IAM, Secret Manager, etc) using Terraform and CloudFormation. Writing documentation for Terraform modules.
Responsibilities	• Writing AWS&GCP networking modules including VPC, Private and Public Subnets, Route Tables, NAT Gateways, Bastion Host, Internet Gateway. • Writing a Terraform module for Cloud SQL and AWS RDS • Wrote a module for EKS, ECS, GKE, Firestore, App Engine, Cloud Functions, Cloud Scheduler, Cloud&AWS Secret Manager, AWS&GCP IAM, GCS, AWS S3, MongoDB Atlas etc. • Worked on 4 environments: dev, qa, stage, prod • Imported manually created infrastructure into terraform and modulerized all resources. • Writing a documentation(a READme file)for Terraform modules

Deploying Applications in the Kubernetes Cluster and Implementing Continuous Deployment with ArgoCD

Role	DevOps Engineer
Technologies	Kubernetes, Helm, GKE, EKS, Docker, GitOps, ECR, GCR, GAR, Secret Manager, IAM
Description	Deploying Applications in the Kubernetes Cluster(GCP GKE and AWS EKS) and Implementing Continuous Deployment with ArgoCD. Monitoring K8s Clusters and K8s objects with Prometheus and Grafana
Responsibilities	• Created ArgoCD applications and configured them to deploy services from Git repositories. • Deploying highly-available applications on a GKE cluster & AWS EKS cluster • Utilized Kubernetes features like rolling updates, automatic scaling, and self-healing to improve the reliability of the application. • Working with Services, Deployments, Statefulsets, RBAC, PV, PVCs, Pods, NS etc • Monitoring Kubernetes clusters with Prometheus&Grafana • Monitoring Kubernetes objects, Building • dashboards, creating alerts, integrating Alertmanager with slack

Creating Upstream and Downstream, meta pipelines

Role	DevOps Engineer

Technologies	Upstream and downstream Gitlab CI pipelines, Terraform, Jira, pipeline templates
Description	Creating different scenario pipelines: • creating templates for pipelines • upstream and downstream pipelines • meta-pipelines for prod • testing scenarios
Responsibilities	• Creating CitLab CI templates for Terraform: tflint, tfsec, init, plan, apply • Working with upstream and downstream pipelines • Working with meta-pipelines • Creating CM tickets

Automating testing scenarios with dsTest and Robot Framework

Role	DevOps Engineer
Technologies	DsTest, Robot Framework, Python, Terraform, AWS networking
Description	Automate dsTest scenarios with Robot Framework (custom library)
Responsibilities	• Performing resilience tests • Automating test scenarios with Robot Framework • Updating a custom Robot Framework Library

Implemented continuous database migration from AWS RDS to GCP SQL

Role	DevOps Engineer

Technologies	Cloud DMS, AWS RDS Postgres, AWS IAM, Cloud SQL Postgres, Terraform, Cloud Monitoring
Description	Writing a Terraform module that Implements continuous database migration from AWS RDS to Cloud SQL with Cloud DMS service. Monitoring Cloud SQL
Responsibilities	• Writing a terraform module for highly available Cloud SQL Postgres instance • Writing a terraform module for Cloud DMS • Establishing connection between AWS RDS Postgres instance and Cloud SQL Postgres instance • Establishing a continuous Database migration • Directing application endpoint from AWS RDS to Cloud SQL • Monitoring Cloud SQL: CLoud Monitoring(Stackdriver), creating dashboards and alerting rules, sending alerts to Opsgenie

Automating manual jobs with a bash script

Role	DevOps Engineer
Technologies	Bash, GCR(Google Cloud Registry), GAR(Google Artifact Registry), Circle CI, GKE ASM
Description	Automating the migration of images from GCR to GAR. Automating the deletion of 1+ year old images with bash. Installing Anthos Service Mesh on a GKE cluster with a bash script.
Responsibilities	• Automating the transfer of images from GCR to GAR. • Automating the filter and deletion of 1+ year old images from GCR with a bash script.

	• Writing a pipeline running the script once a week with a cron job • Automating the Installation of Anthos Service Mesh on a GKE cluster with Bash

Monitoring Infrastructure with different monitoring tools

Role	DevOps Engineer
Technologies	Prometheus&Grafana, Datadog, New Relic, Cloud Monitoring, AWS CloudWatch, Slack, Opsgenie, PagerDuty, Microsoft Teams
Description	Monitoring • Databases(Cloud SQL, AWS RDS) • Standalone servers(AWS EC2 instances, Compute Engine Instances) • Applications • Kubernetes clusters(AWS EKS, GKE) • Kubernetes objects
Responsibilities	• Installing Prometheus Stack on GKE • Setting up and configuring Prometheus metrics collection on Amazon EC2 instances (node exporter and service discovery) • Creating dashboards in grafana • Worked with Prometheus exporters • Creating alerting policies in Alert Manager and adding notification channels to different endpoints(email, Opsgenie, Slack etc) • Monitoring AWS Postgres with Prometheus Postgres Exporter • Creating alarm policies in Cloud Monitoring & AWS CloudWatch

Continuous Integration and Delivery (CI/CD) Pipeline

Role	DevOps Engineer
Technologies	Jenkins, GitLabCI, CircleCI, GitHub Actions
Description	Implementing CI&CD pipelines in Jenkins, CircleCI, GitLabCI, GitHub Actions
Responsibilities	• Designing Jenkins pipeline that builds, pushes Docker images to Dockerhub, and deploys a microservice application to GKE. • Implementing cronjobed pipelines • Set up Jenkins to automatically deploy the application to different environments, including development, staging, and production. • Configured Jenkins to integrate with source control management (SCM) tools such as Git to automatically build and test the application after each commit.