DINESHKUMAR .G

Devops Engineer

+91 9052651446 gandikotadineshroyal6@gmail.com

PROFILE :

Detail-oriented and results-driven DevOps Engineer with 2 years and 9 months of experience in designing, automating, and optimizing mission-critical **deployments in microservices-based** insurance applications. Proven ability in containerization using **Docker, CI/CD integration using Jenkins,** and **Kubernetes orchestration**. Skilled at enhancing developer productivity, reducing deployment times, and maintaining consistent environments across dev, UAT, and production.

Core Competencies:

CI/CD Tools:	**Jenkins, GitHub Actions**
Containers & Orchestration:	**Docker,** Docker Compose, **Kubernetes (k8s), Helm Chart**
Cloud & Registries:	**AWS** (**ECR, EC2, S3, VPC, IAM, SNS, Auto Scaling**), **Azure (AKS, VMs, Storage Accounts, IAM)** **GCP (GKE, Compute Engine, Cloud Storage, IAM) | Docker Hub**
Monitoring & Logging:	**Prometheus, Grafana, ELK Stack ,Splunk**
Infrastructure as Code:	**Terraform,** Ansible
Scripting & Automation:	Bash, **Python**
Version Control:	**Git, GitHub**
Operating Systems:	**Linux (Ubuntu, CentOS)**
Build Tools:	**Maven,** Gradle
Scripting Languages:	YAML
Web Server:	**Apache Tomcat, Nginx**
Other Tools:	**SonarQube, Nexus,** JIRA, Confluence
Database:	**SQL Server**
Testing:	Manual testing
Front end :	**HTML, CSS, React.js**

Education:

R.G.M College of Engineering and Technology (B.Tech)

Certifications:

Oracle Cloud Infrastructure 2025 DevOps Professional (1Z0-1109-25)
Oracle Cloud Infrastructure 2025 Certified AI Foundations Associate (1Z0-1122-25)

Professional Experience:

DLV Software Pvt Ltd, Hyderabad

Project-1 Insurance Claims & Policy Microservices Platform
November 2022 – March 2025

- Architected and maintained **Docker-based containerization** for microservices like policy-service, claims-service, payment-service, and underwriting-service.
- Wrote and reviewed **Docker files**, ensuring security best practices, l**ightweight base images (Alpine)**, and **multi-stage builds** to optimize image size.
- Developed custom **Jenkins pipelines** using Jenkins files to automate the build, test, and deployment stages.
- Integrated **Docker image** builds into **CI/CD pipelines** and pushed to **Amazon ECR** for environment-wide use.
- Automate repetitive tasks and improve workflows using scripting languages such as **Bash,Python** or **Shell Scripting.**
- Used Docker Compose to spin up local dev environments for **multi-container** applications.
- Deployed **microservices on Kubernetes clusters** (staging & production) with Helm charts and managed scaling, health checks, and rollbacks.
- Manage and optimize cloud infrastructure on platforms like **AWS ,AZURE** or **GCP,** ensuring high availability,scalability, and cost efficiency.
- Implement effective monitoring and logging system to track the health of application,server, and infrastructure.Work with tools **Prometheus ,Grafana , ELK Stack.**
- Installed, configured, and maintained **Splunk Enterprise, Universal Forwarders**, and Heavy Forwarders across Linux and cloud environments (AWS/Azure)
- Automate infrastructure provisioning and management using **IaC tools (e.g..Terraform, CloudFormation)** to ensure scalable,repeatable, and secure infrastructure.
- Reduced deployment times from 30+ minutes to under 10 minutes, improving release velocity.
- Work closely with development ,QA,and operation teams to ensure fast feedback loops,system reliability, and optimal performance.

Project -2 E-Commerce

June 2022 to November 2022

- Desgin,implement, and maintain **continuous intergration and continuous deployment** (CI/CD) pipelines to ensure smooth and automated delivery of application across multiple environments.
- Write and update **test cases, test scripts**, and test data based on functional requirements and user stories.
- Collabration with cross-functional teams to identify requirements,desgin scalable solutions.
- Log, track, and prioritize bugs and issues using a defect tracking system such as JIRA.
- Configured Jenkins pipelines to **automatically trigger unit tests, integration tests**, and regression tests after each code commit.
- Configured Jenkins to **publish test reports and send Slack or email notifications** on test failures.
- Perform manual testing, including functional, regression, usability, smoke, and sanity testing
- Perform regression testing to ensure that fixes or changes have not introduced new issues into existing functionality.