

JESSICA CLAIRE

100 Montgomery St. 10th Floor • (555) 432-1000 • resumesample@example.com

Professional Summary

An ardent professional with an experience of 4 years in IT industry comprising DevOps Engineering, System Administration, software development, Build/Release Management, Automation, Deployment, Cloud computing Platforms like AWS, Microsoft Azure and GCP and Containerization in the complete Software Development Life Cycle (SDLC), waterfall, Agile environment and server-side deployment in the application, middleware layers.

Skills

Cloud Services: AWS - EKS, ECS, EC2, S3, ELB, Redshift, EMR, RDS, SQS, SNS, Cloud Watch, Cloud Trail, ELK, Elasticsearch, Azure, Azure DNS, GCP

Operating Systems: Linux (RHEL5/6/7, CentOS), Ubuntu, Debian, Windows.

Container/Orchestration Tools: AWS ECS, EKS, Docker, Kubernetes, Docker Swarm

Automation / Build: Jenkins, Bamboo, Ansible, Chef, TeamCity, Maven, Ant, Gradle, Nexus, Artifactory, Terraform

Version Control Tools: GitLab/ GitHub, Bit Bucket, Subversion, TFS, ClearCase

APP/ Web Servers: Tomcat, Web sphere, Web Logic, JBoss, Nginx, Oracle Tuxedo

Monitoring Tools: Splunk, Nagios, Cloud Watch, Kibana

Scripting Languages: PowerShell, Bash, Perl, Ruby, Groovy and Python

Databases: MySQL, PostgreSQL, MongoDB, Cassandra, SQL Server, Oracle DB2

Networking: DNS, DHCP, TCP/IP, SMTP, LDAP

Ticketing Tools: Jira, Rally, IBM Clear Quest, Fisheye

VMware: vSphere, VMware ESX 5.5 and 6.0, Virtual center server 2.5 and 4.0, VMware

Work History

DevOps Engineer, 07/2020 to Current

Paypal – Michigan, ND

- Implemented highly available and fault-tolerant applications and utilizing orchestration technologies such as Kubernetes and Apache Mesos working on Amazon Web Services (AWS).
- Developed Python scripts to automate AWS services which includes ELB, Cloud Front distribution, EC2, Route53, Auto scaling, ECS, Security groups, Aurora and S3 bucket.

- Set-up databases in AWS using RDS, storage using S3 bucket and configuring instance backups to S3 bucket.

- Prototype CI/CD system with GitLab on AWS utilizing Kubernetes and Docker for runtime environment for CI/CD systems to build and test and deploy.

- Worked on Setting up AWS Firewall rules to allow or deny traffic to and from VM's instances based on specified configuration and used AWS cloud CDN (content delivery network) to deliver content from AWS cache locations drastically improving user experience and latency.

- Automated various infrastructure activities like Continuous Deployment, Application Server setup using Ansible playbooks and has Integrated Ansible with Jenkins.

- Refined automation components with scripting and configuration management (Ansible).

- Wrote Ansible Playbooks with Python SSH as Wrapper to Manage Configurations of AWS Nodes and Test Playbooks on AWS instances using Python.

- Run Ansible Scripts to provision Dev servers.

- Worked with installation, configuration and administration of Amazon web services cloud windows Server.

- Used AWS Kubernetes Service to deploy managed Kubernetes cluster in AWS and created EKS cluster in AWS, with AWS CLI, also used template driven deployment options such as Resource Manager templates and Terraform.

- Managed Kubernetes using Helm charts, Created reproducible builds of Kubernetes applications, managed Kubernetes manifest files and Managed releases of Helm packages.

- Created Terraform templates to create custom sized VPC, NAT subnets, for deployment of Web applications and databases.

- Installed and configured RabbitMQ and HashiCorp Vault as docker container.

- Maintained build related scripts developed in shell for Maven builds.

- Created and modified build configuration files including POM.xml.

- Used MAVEN as build tool on java projects for development of build artifacts on source code.

- Installed and configured Nagios as monitoring tool for disk usage, CPU utilization, list of open files etc.

- And sending alerts to teams based on same.

- Working with Linux/UNIX environments and bash & Shell Scripting for creating projects in Jenkins and Jfrog artifactory as part of release process.

- Environment: Amazon Web Services (AWS), Helm, ELK, Terraform, Ansible, Kubernetes, Docker, Hashicorp Vault, RabbitMQ, Jenkins, Groovy, Git, Prometheus, Grafana, Unix/ Linux environment, Maven, JFrog Artifactory, Shell scripting.

DevOps Engineer, 05/2019 to 06/2020

Paypal – Ontario, CA

- Designed, planned and implementation for existing On-Prem applications to Azure Cloud (ARM), Configured and deployed Azure Automation Scripts utilizing Azure stack (Compute, Web and Mobile, Blobs, Resource Groups, Azure Data Lake, HDInsight Clusters, Azure Data Factory, Azure SQL, Cloud Services and ARM), Services and Utilities focusing on Automation.

- Deployed and configured Azure Web App and Hosting Plan for ASP.NET web application with SQL database.

- This also included configuring web app and database backups, setting up DNS records, purchasing and configuring SSL certifications etc.

- Azure AD - Implemented Azure AD using AD connect, configured SSO and multifactor Authentication.

- Also configured SSO from Window 10 based computer which is joined to Azure AD.

- Implemented and managed AD synchronization.

- Migrated Microservice Based applications from Virtual Machines to Docker containers and managed clustered containers using Kubernetes.

- Managed Kubernetes using Helm charts, Created reproducible builds of Kubernetes applications, managed Kubernetes manifest files and managed releases of Helm packages.

- Worked with Chef to manage Web Applications, Configure Files, Database, Users, Mount Points, packages and responsible for creating/managing Chef Cookbooks.

- Configured, maintained and fixed application builds, automated build process and designed applications installers to fit in automated builds environment.

- Implemented CI/CD pipeline using Jenkins and extensively worked for end-to-end automation for all build and deployments.

- Implemented Blue/Green pattern deployment methodologies and roll backs with zero downtime to deploy process in Tomcat and AKS API server and worker node upgrade.

- Installed and configured Splunk as monitoring tool for disk usage, CPU utilization, list of open files etc.

- And sending alerts to teams based on same.

- Reporting windows event logs from remote machines to Splunk using Splunk universal forwarder.

- Also ran query from Splunk UI to access data from Microsoft DB using Splunk DB Connect.

- Deployed and configured Git repositories with branching, forks, tagging, and notifications.

- Environment: Azure, Java/J2EE, JavaScript, JFrog, chef, Azure Devops, Kubernetes, Helm, Docker, Splunk, Golang, GIT, Unix/ Linux environment.

Education

Master's Degree

University of Central Missouri - Lee's Summit, MO

Bachelor's degree

Andhra University - India