

Kavitha P

Email: kavithapcareer@gmail.com

Mobile: (940) 277-6248

LinkedIn: <https://www.linkedin.com/in/kavithaprp/>

Website: <https://kp.devopscommit.com/>



PROFESSIONAL SUMMARY:

- Over 7 Years of experience in IT industry with imperative concentration on **Cloud Computing (AWS, Azure, Google cloud), Build & Release, DevOps, Linux and Network Administration.**
- Designed, configured and managed public/private cloud infrastructures utilizing **Amazon Web Services (AWS)** like **EC2, Elastic Load-balancers, Elastic Container Service (Docker Containers), S3, Elastic Beanstalk, Cloud Front, Elastic File System, RDS, DynamoDB, DMS, VPC, Direct Connect, Route53, Cloud Watch, Cloud Trail, Cloud Formation, IAM, EMR and Elastic Search.**
- Expertise in working on Multiple **AWS instances**, set the **security groups, Elastic Load Balancer and AMIs, Auto scaling** to design cost effective, fault tolerant and highly available systems.
- Accomplished in creating configured **AWS IAM** and Security Group in Public and Private Subnets in **VPC** and Created **AWS Route53** to route traffic between different regions.
- Experience in performance tuning and query optimization in **AWS Redshift.**
- Worked on setting up the life cycle policies to back the data from **AWS S3** to **AWS Glacier**, Worked with various **AWS, EC2 and S3 CLI** tools.
- Build infrastructure as a code using tools like **Terraform** and **Cloud Formation** Templates.
- Experience on using **AWS Lambda**, running scripts/code snippets in response to events occurring in **AWS Cloud** and maintaining it.
- Strong Experience on Administration and Troubleshooting of **Azure IAAS Components (VM, Storage, VNET, OMS, NSG, Site to Site VPN, RBAC, Load Balancers, Availability Sets).**
- Proficient in Azure Development, worked on **Azure web application, App services, Azure storage, Azure SQL Database, Virtual Machines, Fabric controller, Azure AD, Azure search, and Notification hub.**
- Expertise in Microsoft **Azure Cloud Services (PaaS & IaaS), Application Insights, Document DB, Internet of Things (IoT), Azure Monitoring, Key Vault, Visual Studio Online (VSO) and SQL Azure.**
- Sound knowledge in **migrating** on premise to **Windows Azure** using **Azure Site Recovery** and **Azure backups.**
- Good understanding with other technologies like **Google Cloud Platform** and **Open stack.**
- Planned, developed & maintained Infrastructure as code using **CI/CD** deployments using Terraform.
- Expertise in implementing DevOps culture through **CI/CD** tools like **Repos, Jenkins, Code Deploy, Code Pipeline, Maven**, and configuration management tools like **Chef, Puppet & Ansible.** Had set up staging and production environments for **CI** workflow.
- Hands on experience on automating various infrastructure activities like **Continuous Deployment, Application Server setup, Stack monitoring** using **Ansible playbooks** and on **Integrating Ansible** with **Run deck** and **Jenkins.**
- Implemented and rolled out proof of concepts for logging management (**ELK, Logentries, Sumologic**), writing **Chef** code to deploy these solutions, mocking up dashboards and prototyping example queries of high-value searches.
- Automated deployment activities in middleware environment which includes portal applications running on **Java/AWS** using **Puppet** master and Agent nodes and Integrated **Puppet** with **Apache** and **Passenger.**
- Handled **Kubernetes** to deploy scale, load balance, scale and manage **docker** containers with multiple name spaced versions.
- Exposure to **Mesos** and **Marathon** environment for application deployments and **Docker Containers.** Installing of **Docker** using **Docker toolbox**, created **Docker images** using a **Docker file.** Also, worked on **Docker container snapshots**, removing **images**, and managing **Docker volumes.**
- Experienced with deployment automation of all the **microservices** to pull image from the private **Docker Registry** and deploy to **Docker swarm** cluster using **Ansible.**
- Expertise in setting up (**CI/CD**) Continuous integration and Deployment Pipeline for **J2EE, NodeJS** and **Python** applications as well as **Docker Containers** using tools such as **Jenkins, Anthill pro, Team city, Bamboo** and **AWS Code Pipeline.**

- Extensive experience in building artefacts for **J2EE** applications using **ANT, Maven, XCODE, and AWS Code Build**.
- Experienced in monitoring servers using **Nagios, Data dog, Cloud watch, Splunk**.
- Implemented **Kafka** security features using **SSL** without **Kerberos**, further with more grain-fines security I used to set up **Kerberos** to have users and groups this will enable more advanced security features.
- Worked on analysis and visualization of data using **ELK (Elasticsearch, Logstash and Kibana)** and **MySQL databases** in a **Linux (CentOS)** environment.
- Used **JIRA** as a Change Management/ Work Management/ SCRUM Agile Tool. Deployment and management of many servers via script and **Chef**, utilizing cloud providers as a direct **Jira**.
- Experience in maintaining **Atlassian products** like **JIRA, Confluence, Fisheye, Bamboo, Crowd** etc.
- Expertise in Querying **RDBMS** such as **Oracle, MySQL, and SQL Server** by using **PL/SQL** for **data integrity** and **proficient in multiple databases** like **MongoDB, Cassandra, MySQL, PostgreSQL, ORACLE, DynamoDB, nosql**.
- Expertise in **Python, Perl, shell scripting (ksh, bash), Groovy, Ruby, Yaml** to automate administration tasks.
- Experience on Deployed the Software builds to **QA, UAT and Production servers**.
- Experience in System Administration on various platforms which includes **Red hat Linux, Solaris, SUSE Linux, CestOS, Windows Environment, VMware ESX/ESXI and VMware vSphere**.
- Experts in creating and managing user accounts, security, rights, disk space and process monitoring in **Solaris and Red Hat Linux**.
- Experience on Web servers such as **Tomcat 3.x, 4.x, 5.x,6.x, and 7.x & Apache Server 2.x** and **integrating** them with **WebLogic 10.x servers**.
- Working knowledge on configuring access lists. **Troubleshooting DNS/DHCP** issues within the **LAN** network.
- Experienced in supporting **24x7** production computing environments, on-call and weekend support.
- Have the responsibility as a **UAT** support for the project until the project goes into the production.
- Experience on Deployed the Software builds to **QA, UAT and Production servers**.
- Organized and facilitated project planning, daily stand-up **meetings, reviews, retrospectives, sprint, release planning, demos, and other Scrum-related meetings**.

PROFESSIONAL EXPERIENCE

Markel Corporation - Remote

Dec 2021 - till date

Role: Sr. AWS DevOps Engineer

Responsibilities:

- Developed and implemented **Software Release Management strategies** for various applications according to the **agile** process.
- Created and maintained a mission critical, no **downtime** central service infrastructure.
- Provision resources to build highly available and secure environment using **AWS** services like **VPC, EC2, EBS, EFS, S3, ELB, Auto-Scaling, RDS**, etc.
- Developed **API** for using **AWS Lambda** to manage the servers and run the code in **AWS**.
- Coordinating with implementation team, to build and engineer services for **Linux** and **Windows OS** on **AWS, GCP** and **Azure** platforms. Provisioned Instances, Storages & monitoring Services and **CI/CD pipeline** through **Jenkins**.
- Worked with **AZURE** infrastructure management (**AZURE web roles, worker roles, SQL AZURE, AZURE storage, AZURE AD licenses**). Virtual machine backup and recover from a recover services vault using azure **PowerShell** and portal.
- Designed **AWS Cloud Formation templates** to create custom sized **VPC, subnets, NAT** to ensure successful deployment of **Web applications** and **database** templates.
- Worked on **AWS Lambda** to run the code in response to events, such as changes to data in an **Amazon S3 bucket, Amazon DynamoDB** table, **HTTP** requests using **AWS API Gateway** and invoked the code using **API** calls made using **AWS SDKs**.
- Maintained **DNS** records using **Route53** and Used **AWS Route53** to manage **DNS zones** and give public **DNS** names to **elastic load balancers IP's**.
- Provisioned the highly available **EC2 Instances** using **Terraform** and wrote new plugins to support new functionality in **Terraform**.

- Worked in an **IAAS** environment called **Terraform**, to manage application infrastructures such as storage and networking.
- Extensively involved in **infrastructure as code**, execution plans, resource graph and change automation using **Terraform**. Managed **AWS** infrastructure as code using **Terraform**.
- Expertise in writing new plug-ins to support new functionality in **Terraform**.
- Developed custom **Kubernetes** Operators using **Kubebuilder** to automate deployment, scaling, and management of containerized applications, improving operational efficiency by 30%.
- Designed and maintained multiple **Helm charts** for packaging applications into reusable Kubernetes components, resulting in a 50% reduction in deployment times and simplified version control across environments.
- Engineered robust backend services and microservices in **Golang**, optimizing application performance and scalability in high-load, distributed systems.
- Led a team of DevOps engineers in adopting new technologies like **Kubebuilder** and **Helm**, fostering a culture of innovation and continuous improvement.
- Experience with **Docker** and **Kubernetes** on multiple cloud providers, from helping developers build and **containerize** their application (**CI/CD**) to deploying either on public or private cloud.
- Experience in **Virtualization** technologies like **VMWare**, **Vagrant** and worked with **containerizing** applications like **Docker**, **Kubernetes**.
- Developed container support for **cloud environments** and **Docker**. Deployed the applications in **Docker containers** and managed the **containers** in enterprise level.
- Leverage **Docker containers** and **Docker** consoles for managing the application lifecycle and worked on setting up the automated build on **Docker HUB** and deployed CoreOS **Kubernetes Clusters** to manage **Docker containers** with light weight **Docker Images** as base files.
- **Jenkins** is built on **Docker container** and the **master** controller **Kubernetes** controls **Pods**.
- Worked on deployment automation of all the **Microservices** to pull image from the private **Docker registry** and deploy to **Docker swarm cluster** using Ansible.
- Involved in all projects that move to **production** and work closely with the **Data Centre, Development, Quality Assurance** and Management teams to ensure cross communication and confirmed approval of all **production** changes.
- Wrote **Ansible Playbooks** with **Python SSH** as the Wrapper to Manage Configurations of **AWS** nodes and Tested Playbooks on **AWS instances** using **Python**.
- Wrote **Ansible playbooks** from scratch in **YAML**. Installing, setting up & Troubleshooting **Ansible**, created and automated platform environment setup.
- Worked on **Ansible** to manage Web Applications, Config Files, Data Base, Commands, Users Mount Points, and Packages. **Ansible** to assist in building automation policies.
- Create **Ansible playbooks** to deploy **Elasticsearch LogStash Kibana** stack to monitor search queries traffic/latency.
- Querying and reporting from **Elastic Search/Kibana, Splunk** and **MySQL**.
- Created and automated the **Jenkins pipeline** using pipeline **groovy** script for the applications.
- Extensively worked on **Jenkins CI/CD pipeline** jobs for end-to-end automation to **build, test** and **deliver artifacts** and **troubleshoot** the build issue during the **Jenkins** build process.
- Created supported documentation for users to have their **SSH** key into **GitHub** and into **Jenkins** vault.
- Build scripts using **ANT** and **MAVEN** build tools in **Jenkins, Sonar** to move from one environment to another environment.
- Technical leader to align processes and tools, such as branching, source control structure, dependency management, Linux\Windows hybrid build infrastructure, code review, and check-in policies, that are developed and instrumented by teams across projects globally.
- Configuring, maintaining network and troubleshooting such as **LDAP, DNS, NIS, NFS, Web, Mail, FTP** and **SNMP**.
- Source code admin – Manage and configure **GitHub**, resolve issue regarding source code management, manage branching and merging, code freeze process.
- Implemented deployment strategies **Blue-Green** Deployment and **Rolling** Deployment

- Migrated the production **MySQL** schema to the new **AWS RDS Postgres** instance.
- Proficiency in multiple databases like **MongoDB, Cassandra, MySQL, ORACLE** and **MS SQL** Server.
- Wrote **Python, Shell** and **Perl** scripts for managing day-to-day transactions & for automation of routine tasks.
- Installed/Configured and Managed **JFrog Artifactory** Manager.
- Successfully deployed the application on the **android marketplace** and supported the app for different releases of **android OS** and also configured **CI/CD** for **PlayStore** deployments.
- Perform Deployment of Release to various **DEV, QA, UAT & PROD** environments.
- Involved with the build team in troubleshooting and fixing day-to-day problems of the applications in **production** on **24/7** schedule.

Environment: AWS, AZURE, GCP Terraform, Docker, Kubernetes, Chef, Ansible, Nagios, Splunk, SonarQube, Oracle, MSSQL, MySQL, AWS EC2, IAM, S3, VPC, Cloud Watch, ELB, SNS, GIT, GITHUB, Maven, Jenkins, Python, Perl, Shell, Jboss, Tomcat.

Change Healthcare, Nashville-TN

Jan 2020 – Nov 2021

Role: Sr. AWS DevOps Engineer

Responsibilities:

- Communication with team members for both **Ansible Core** and **Ansible Tower** teams to clarify requirements and overcome obstacles.
- Maintained high availability clustered and standalone server environments and refined automation components with scripting and **configuration management (Ansible)**
- Designed, Installed and Implemented **Ansible** configuration management system.
- Used **Ansible** to manage Web applications, Environments configuration Files, Users, Mount points and Package
- Migrating present Linux environment to AWS by creating and executing a migration plan, deployed EC2 instances in VPC, configured security groups & NACL's, attached profiles and roles using AWS Cloud Formation templates and **Ansible** modules.
- Configured JMS servers using Configuration Management Process using Ansible.
- Implemented **CI/CD** pipeline using Jenkins, **Ansible Playbooks** and **Ansible Tower**.
- Worked on containerizing **OpenStack** services in Docker using ansible.
- Managed **Ansible** Playbooks with **Ansible modules**, implemented CD automation using **Ansible**, managing existing servers and automation of build/configuration of new servers.
- Worked on running **Jenkins** in **Docker** containers with EC2 slaves in AWS Cloud configuration with **Mesos** and **Kubernetes**
- Spinning up **Docker** containers for installing application services.
- Implemented multiple **CI/CD** pipelines as part of DevOps role for on-premises and cloud-based software using **Jenkins, Chef and AWS/Docker**.
- Developed procedures to unify, streamline and automate application development and deployment procedures with **CLI** and **Linux** container technology using Docker.
- Worked with Kubernetes to manage **Docker** containers
- Setup NFS servers as OpenShift storage backend and configured various storage claims for developers
- Support and enhance SDLC cycle by using Docker containers delivered with O to **OpenShift Container Platform**
- Our Continuous Integration stack consists of **Git, Jenkins CI-CDX**, the **Docker** Trusted Registry, and **OpenShift Enterprise**
- Created playbooks for **OpenStack** deployments and bug fixes with Ansible.
- Created branches and managed the source code for various applications in **GIT**.
- Configured **GIT** with **Jenkins** and schedule jobs using **POLL SCM** option.
- Responsible for installing Jenkins master and slave nodes and configuring Jenkins builds for continuous integration and delivery.
- Automate and maintain Jenkins tasks using **groovy scripting**.

- Using **Groovy, Java, Jenkins, python** and expect utility designed and implemented one click deployment to development, integration and QA environments.
- Used Code Quality Analysis techniques to performed integration– Check style, Find bugs with CI tools.
- Built scripts using ANT and MAVEN build tools in Jenkins to move from one environment to other environments.
- Created and setup automated nightly build environment for java projects using **Gradle**.
- Collaborated with development support teams to setup a continuous delivery environment with the use of **Docker**.
- Designed, Installed and Implemented **Ansible** configuration management system.
- Used **Ansible** to manage Web applications, Environments configuration Files, Users, Mount points and Package
- Migrating present Linux environment to AWS by creating and executing a migration plan, deployed EC2 instances in VPC, configured security groups & NACL's, attached profiles and roles using AWS Cloud Formation templates and **Ansible** modules.
- Implemented **CI/CD** pipeline using Jenkins, **Ansible Playbooks** and **Ansible Tower**.
- classes for the Application logic and utilities.
- Automated webserver content deployments **via shell scripts**.
- Used **cookbooks, recipes, roles, data bags, run list** to deploy web applications, applications servers and vendor applications.
- Used Python Scripting for uploading a file to s3 and deploying them and creating GO serverless applications and deploying AWS Lambda.
- Used Load ash library to make java scripting modular and productive.
- Worked on running **Jenkins** in **Docker** containers with EC2 slaves in AWS Cloud configuration with **Mesos** and **Kubernetes**
- Spinning up **Docker** containers for installing application services.
- Implemented a Continuous Delivery pipeline with **GitHub, Jenkins, Docker** and **AWS AMI's**.
- Converted our Production and staging environment from a handful AMI's to a single bare metal host running **Docker**.
- Used infrastructure management tool to keep track of all releases and request of the developers.

Environment: AWS, AZURE, GCP Terraform, Docker, Kubernetes, Chef, Ansible, Nagios, Splunk, SonarQube, Oracle, MSSQL, MySQL, AWS EC2, IAM, S3, VPC, Cloud Watch, ELB, SNS, GIT, GITHUB, Maven, Jenkins, Python, Perl, Shell, Jboss, Tomcat.

ValueLabs, Hyderabad-TS

Nov 2018 - Dec 2019

Role: AWS DevOps Engineer

Responsibilities:

- Created highly available and scalable infrastructure in **AWS** cloud by using various **AWS** services like **EC2, S3, ELB, VPC, Route53, IAM**.
- Designed roles and groups for users and resources using **AWS Identity Access Management (IAM)** for accessing AWS resources.
- Performed **AWS S3** buckets creation, policies, **IAM** role based polices and customizing the JSON template.
- Managed storage in AWS using **Elastic Black Storage, S3, Glacier**. Created Volumes and configured **Snapshots**.
- Used **Amazon Glacier** for archiving data, **Amazon RDS** to manage, create snapshots, and automate backup of database.
- Created **load balancers(ELB)** and used **Route53** with failover and latency options for high availability and fault tolerance.
- Scaling web applications and services developed with **Java, .Net, PHP, Node.js, Ruby, Python** on servers like **Apache, Nginx, IIS** using **AWS Elastic Bean Stalk**.
- Partner with product management and product development to drive adoption and massively **scale IoT** capabilities throughout the enterprise.

- Help develop an overall **IoT** and Mobile **ecosystem engagement strategy**.
- Responsible for the acquisition and management of **IoT services** at the launch of new solutions.
- Worked on **Terraform** for managing the infrastructure through the terminal sessions and executing the scripts and creating alarms and notifications for EC2 instances using Cloud Watch.
- Involve in maintaining the user accounts (**IAM**), policies, **RDS, DynamoDB, SQS and SNS** services in AWS cloud.
- Created Python scripts to totally automate AWS services, which includes **web servers, ELB, Cloud Front distribution, EC2 and Security groups, S3** bucket and application configuration, this script creates stacks, single servers, or joins web servers to stacks.
- Integrated **Docker** container orchestration framework using **Kubernetes** by creating pods, config Maps, deployments.
- Implemented new **Docker** container creation process for each **GitHub** branch. Container gets started using **Jenkins** as Continuous Integration server.
- Managed **Kubernetes** charts using **Helm**. Created reproducible builds of the Kubernetes applications, managed Kubernetes manifest files and managed releases of Helm packages.
- Used **Docker** to virtualize deployment containers and push the code to EC2 cloud using PCF. Built additional **Docker Slave** nodes for **Jenkins** using custom built **Docker images** and instances.
- Worked on **Docker** container snapshots, attaching to a running container, removing images, managing directory structures, and managing containers in **AWS ECS**.
- Deployed and hosted applications on **AWS** using **Ops Works**, automated single click deployment onto AWS using **Ansible**. Developed automation scripts in **YAML** using **Ansible** to deploy and manage Java applications across Linux servers.
- Worked on **Vagrant** for configure lightweight, reproducible, and portable development environments
- Designed the project workflows/pipelines using **Jenkins** for Continuous Integration and deployment into different **Web/Application Servers**.
- Upgrading, administering, **plug-in management**, and user management and job creation in **Jenkins**.
- Used **Maven** dependency management system to deploy snapshot and release artifacts to **Nexus**, share artifacts across projects.
- Experienced in static code analysis tool **SonarQube** right from setup to configuration in **Jenkins** and analyzing the different programming languages like Java, C# and JavaScript.
- Used **Python, Perl, Shell** scripts in automation and integration.
- Managed code by maintaining code repository in **GIT (Bit Bucket)**; improve practices of branching and code merge to custom needs of development team.
- Analyze and resolve compilation and deployment errors related to **code development, branching, merging**, and building of source code.
- Monitoring and application analysis using **Nagios, Splunk, Prometheus**.
- Used the **JIRA** for bug tracking, creating the dashboard for issues.
- Facilitated **Scrum** ceremonies like **Sprint** planning, retrospectives, Daily stand-ups, etc.

Environment: AWS, Terraform, Docker, Kubernetes, Chef, Ansible, Nagios, Splunk, SonarQube, Oracle, MSSQL, MySQL, AWS EC2, IAM, S3, VPC, Cloud Watch, ELB, SNS, GIT, GITHUB, Maven, Jenkins, Python, Perl, Shell, Jboss, Tomcat.

Posidex Technologies, Hyderabad, TS.

April 2018 – Oct 2018

Role: Release/Build Engineer

Responsibilities:

- Monitored software, hardware and/or middleware updates and utilizing technologies like **Jenkins/Hudson, Ant, MS Build**, and **Subversion**.
- Enabled Continuous Delivery through Deployment into several environments of Test, QA, Stress and Production using **Jenkins**. Implementing a Continuous Delivery framework using **Jenkins, Maven** in **Linux** environment.
- Set up **puppet** master, client and wrote scripts to deploy applications on Dev, QA, production environment.

- Created **Puppet** manifests, classes and modules to automate system operations. Developed the scripts to push patches, files and maintain configuration drift through the **Puppet** Tool.
- Deployed Puppet, Puppet Dashboard, and Puppet DB for configuration management to existing infrastructure.
- Involved in installing **Puppet** client on Red hat & Solaris servers for Automation purpose.
- Involved in Installing **Jenkins** on a **Linux** machine and created a master and slave configuration to implement multiple parallel builds through a build farm.
- Implemented new project builds using **Jenkins** and **Maven** as build framework tools, inspected builds in staging environment before rolling out to production environment.
- Defined dependencies and plugins in **Maven** pom.xml for various activities and integrated **Maven** with **GIT** to manage and deploy project related tags.
- Implemented **GIT** Lab for version control of **Puppet** modules and process documentation.
- Manage Code Repository by, maintaining code repository in **GIT**, improve practices of branching and code merge to custom needs of development team.
- Initiated responsibility for administering the **SVN** and **GIT** servers which included install, upgrade, backup, adding users, creating repository/branches, performance tuning, troubleshooting issues and maintenance.
- Implemented **Shell** and **Perl** scripts for release and build automation. Manipulated and automated scripts to suit requirement.

Environment: Puppet, Jenkins, Hudson, ANT, Maven, Git, SVN, Jira, Bash, Shell, Perl, Python, Ruby, Tomcat, WebLogic, Auto scaling, WebSphere, Nagios, RHEL.

Cheqs InfoSoft Solutions, Hyderabad, TS

May 2017 – March 2018

Role: Release/Build Engineer

Responsibilities:

- Installed, configured and administered **Jenkins** CI tool on **Linux** machines.
- Configuration Management (**SCM**) tools like **Subversion (SVN)**, **GIT**. Setup **Jenkins** Build server and maintenance.
- Experience in User Management and Plug-in Management for **Jenkins** to automate the daily process.
- Worked in creating WebSphere Application Server Clustered Environments and handling Load Balancing for QA, UAT and Production.
- Automated **Linux** production server setup using **Puppet** scripts. Designed and implemented **Puppet**-based configuration management system for all new Linux machines (physical and virtual).
- Managed the **Maven** Repository using **Nexus** tool to automate the build process and used the same to share the snapshots and releases of internal projects and **SVN** repositories for branching, merging, and tagging.
- Created **Maven** shell scripts for build and release activities in QA, Staging, and Production environments.
- Integrate **GIT** into **Jenkins** to automate the code check-out process. Performed all necessary day-to-day **Subversion/GIT** support for different projects.
- Responsible for design and maintenance of the Stash **GIT** Repositories, views, and the access control strategies. Written **Perl script** to monitor and notify team about the Rsync updates.
- Developed and maintained **UNIX/Perl/ANT** scripts for **Java/J2EE** build and release tasks.
- Implemented Release schedules, communicated the Release status, created Roll Out Plans, tracked the Project Milestones, prepared the reports and chaired the Release calls and worked for a **Jira** Application.
- Deployed Enterprise Application Archive (EAR), Web Application Archive (WAR), Staging and Production environments. Modified **ANT** scripts to build the JAR's, Class files, WAR files and EAR files from VOB's.
- Strong ability to troubleshoot any issues generated while building, deploying and in production support. Good Interpersonal Skills, team-working attitude, takes initiatives and very proactive in solving problems and providing best solutions. Good knowledge on ITIL Process and have worked on coordinating releases across the projects.

Environment: Linux (RHEL, CentOS), Puppet, Jenkins, Hudson, ANT, Maven, Git, SVN, Puppet, Jira, Bash, Shell, Perl, Python, Tomcat, WebLogic, WebSphere, JBOSS, Nagios, RHEL.