**Name: MOUNICA M**

**DEVOPS ENGINEER**

**Email: reddy.mounica8951@gmail.com Contact no: 858-255-4496**

**LinkedIn: www.linkedin.com/in/m-reddy-professional**

# **Summary**:

Around 6 Years of specialized IT experience in DevOps/Cloud, Linux administration, Expertise in Architecting and implementing Azure Service Offering, such as Azure cloud services, Azure storage, IIS, Azure Active Directory (AD), Azure Resource Manager (ARM), Azure Storage, Azure Blob Storage, Azure VMs, SQL Database, Azure Functions, Azure Service Fabric, Azure Monitor, and Azure Service Bus, Implemented AWS solutions using EC2, Route53, S3, Cloud Front, RDS, IAM, EBS, Elastic Beanstalk and Elastic Load Balancer, Auto scaling groups, Optimized volumes and EC2 instances, Cloud Watch, and Cloud formation, Hands-on experience with DevOps tools like Chef, Puppet, Ansible, Docker, Jenkins and AppDynamics.

# **PROFESSIONAL SUMMARY**

* Good understanding of **Azure Cloud**products like Virtual Machines, Storage Accounts, Application Insights, Automation Accounts, Databases, Networking and IaaS/PaaS resources.
* Expertise in building **CI/CD** on AWS environment using **AWS Code Commit**, **Code Build**, **Code Deploy** and **Code Pipeline** and experience in using **AWS CloudFormation, API Gateway,** and **AWS Lambda** in automation and securing the infrastructure on **AWS**.
* Hands on experience in AWS provisioning and good knowledge of **AWS services** like **EC2, S3, Glacier, ELB, RDS, Redshift, IAM, Route 53, VPC, Auto scaling, Cloud Front, Cloud Watch, Cloud Trail, Cloud Formation, Security Groups**.
* Experience in working on all cloud Platforms such **as Google Cloud Platform (GCP), Amazon Web Services (AWS), Azure, Pivotal Cloud Foundry (PCF)**and Implemented a Continuous Delivery framework using Jenkins, Ansible, GIT, SVN, Docker, Kubernetes, Maven and Nexus in Linux environments along with the installation, configuration, and maintenance of **Pivotal Cloud Foundry (PCF) on AWS**
* Deployed and optimized two tier **Java, Python**web applications to **Azure DevOps CI/CD** to focus on development by using services such as **Repos** to commit codes, **Test Plans** to unit test, deploy **App Service**, **Azure Application Insight** collects health performance and usage data of the process, stored artifacts in blob storages.
* Written runbooks in PowerShell and deployed using **CICD Az DevOps**to **Automation Accounts**. Release pipelines use **Azure AD Application Registration service principal**in the Azure DevOps **service connections**for authentication to Azure. Runbook uses **connections**for Authentication.
* Implemented **Git**for **Branching**, **Merging**, **Tagging** and maintaining the version across the environments on **Linux** platforms and, designed and implemented **Git Metadata** including elements, labels, attributes, **triggers**, hyperlinks.
* Proficiency in using **Docker Hub, Docker Engine, Docker images, Docker Weave, Docker Compose, Docker Swarm**, and **Docker Registry**and used **containerization**to make applications platform when moved into different environments.
* Expertise with **Docker** images using a **Docker** file, worked on container snapshots, removing Images and managing **Docker** volumes. Orchestration of **Docker** images and Containers using **Kubernetes** by creating master and node.
* Experience in building and deploying the application code using **Kubernetes CLI** called **kubectl, kubelet, kubeadm, kubespray** and schedule the jobs using **Kube scheduler.** Managed **K8s** charts using **Helm** and created reproducible builds of the **Kubernetes** applications.
* Extensive experience in installing, configuring and administering **Jenkins CI**tool on Linux machines and used **Jenkins Pipelines**to drive all Microservices builds out to the **Docker Registry**and then deployed to **Kubernetes**, Created Pods and managed using Kubernetes.
* Developed **Ansible** playbook in a variety of areas including: Docker base deployment, Docker Swarm configuration, oracle deployment, linux system provisioning, **Jenkins**management (deploy seed job from ansible), Vsphere (**VMware**guest) management, module development.
* Experience in writing **Infrastructure as a code**(IaC) in **Terraform, Azure resource management, AWS Cloud formation.**Created reusable **Terraform modules**in both Azure and AWS cloud environments.
* Skilled with **Python, Bash/Shell, PowerShell, Ruby**, **Perl**, **PowerShell**, **JSON**, **YAML**, **Groovy**. Developed **Shell** and **Python** Scripts used to automate day to day administrative tasks and automation of the **build** and **release process**.
* Expertise **in JIRA**for issue tracking and project management. Experienced with Installing and Configuring the **NEXUS**Repository manager for sharing the artifacts within the company. Also, Supported and developed tools for integration, automated testing and release management.
* Developed Terraform scripts to automate AWS services which include Web servers, ELB, Cloud Front distribution, Database, EC2, database security groups and S3 bucket.
* Extensively worked on Jenkins for Continuous Integration and strong experience with integration of **Ant** and **Maven** Build as build tools for the building of deployable **artifacts (jar, war & ear)** from source code.
* Hands-On experience in the functional usage and deployment of application in **WebLogic**, **WebSphere** and **Apache Tomcat Server**
* Extensive experience in configuration, deployment automation with various App Servers like Oracle **WebLogic**, Web Sphere, Webservers like Apache, Tomcat and more modern **Web Containers like Nginx** etc.
* Expertise in implementing and configuring **Nagios** for continuous monitoring of applications in the production environment and enabled notifications via emails and text messages.

# **Technical Skills**

|  |  |
| --- | --- |
| **PROGRAMMING LANGUAGES & SCRIPTING LANGUAGES** | C, SQL, Java, PYTHON, SHELL, YAML, PERL, SHELL, BASH, |
| **OPERATING SYSTEMS** | WINDOWS, MAC OS, LINUX, UNIX, RHEL, CENT OS, SOLARIS, CITRIX XENSERVER. |
| **FRAMEWORKS** | DJANGO, FLASK |
| **DATABASES** | MS ACCESS, ORACLE, MYSQL, SQL SERVER, MONGODB, CASSANDRA, ORM LIBRARIES, |
| **VERSIONING TOOLS** | GIT, SUBVERSION (SVN), CLEAR CASE. |
| **BUILD TOOLS** | MAVEN, PYBUILDER |
| **CONTAINERIZATION TOOLS** | DOCKER, KUBERNETES |
| **MONITORING TOOLS:** | SPLUNK, NAGIOS, KAFKA. |
| **CI/CD TOOLS** | BAMBOO, GITLAB AND JENKINS. |
| **TICKETING TOOL** | JIRA, BITBUCKET, SERVICENOW |
| **TESTING TOOLS** | SELENIUM, UNIT TESTING, INTEGRATION TESTING |
| **AUTOMATION TOOLS:** | ANSIBLE, CHEF, TERRAFORM. |
| **WEB SERVERS** | WEB LOGIC, WEB SPHERE, APACHE TOMCAT, JBOSS. |
| **WEB SERVICES** | REST API’S, SOAP API’S, RESTFUL SERVICES |
| **FRONT-END TECHNOLOGIES** | HTML, CSS, JAVA SCRIPT, ANGULAR, REACT |
| **CLOUD TECHNOLOGIES** | AWS, AZURE, GCP |
| **DATA SCIENCE CONCEPTS** | STATISTICAL ANALYSIS, DATA PREPROCESSING, FEATURES ENGINEERING, FEATURE SELECTION, MACHINE LEARNING ALGORITHMS, NLP (NATURAL LANGUAGE PROCESSING), SENTIMENTAL ANALYSIS |

# Work Experience

**CLIENT: PENNY MAC LOAN SERVICES| CA OCTOBER 2019 – PRESENT**

**SENIOR DEVOPS ENGINEER/ CLOUD**

**RESPONSIBILITIES**

* Used **Jenkins** as Code Deploy plugin to deploy to **Azure** and extensively worked on other tools like Bamboo, Hudson for continuous integration and for all end to end automation functions with all build and deployments.
* Implemented **Jenkins** pipelines into Azure pipelines to drive all micro services builds out to the **Docker** registry and then deployed to **Kubernetes**, Created Pods and managed using AKS**.**
* Involved in managing **Azure Container Registry** to store private **Docker images**, which are deployed and **Azure pipelines** to build, test, and deployments. **Azure Monitor** to collect metrics and logs. Configured **Monitor** to track performance and maintain security, and used **Splunk** tool to collect metrics, queries.
* Involved in using **Terraform** and **Ansible**, migrate legacy and monolithic systems to **Azure** and managing Ubuntu and RHEL virtual servers on **Azure** by creating **Ansible** Nodes.
* Used Azure Kubernetes Service to deploy a managed Kubernetes cluster in Azure and created an AKS cluster in the Azure portal, with the Azure CLI, also used template driven deployment options such as Resource Manager templates and Terraform .
* Written Automation scripts in PowerShell, which make API calls to Azure DevOps and find users who have not accessed Azure DevOps for more than 90 days (Cost Optimization Project).
* Deploying a Linux Kubernetes Clusters with ACS from the Azure CLI.
* Implemented a CI/CD pipeline using **Azure DevOps(VSTS,TFS)** in both cloud and on-premises with **GIT, MS Build, Docker, Maven**along with **Jenkins**plugins.
* Used **Azure Kubernetes**service to deploy a managed Kubernetes cluster in **Azure**and created an AKS cluster in the Azure portal, with the Azure CLI, also used template driven deployment options such as Resource Manager templates and Terraform.
* Used Kubernetes to manage containerized applications using its nodes, Config-Maps, selector, Services and deployed application containers as Pods.
* Worked on **Kubernetes** for scheduling, deploying, and managing pods, replication controllers, deployments, labels, health checks and ingress by writing **YAML files** onto a node cluster using **Helm Charts.** Created reproducible builds of the **Kubernetes** applications, managed Kubernetes manifest files and managed releases of **Helm** packages
* Deploying windows **Kubernetes (K8s)** cluster with **Azure Container Service (ACS)** from **Azure CLI** and Utilized **Kubernetes** and **Docker** for the runtime environment of the **CI/CD** system to **build**, **test** and **Octopus Deploy**.
* Integrated **Docke**r container-based test infrastructure to **Jenkins CI** test flow and set up build environment **integrating** with **Git** and **Jira** to trigger builds using **Webhooks** and **Slave Machines**.
* Used to write**yaml** files to define multiple services of**docker** applications and used **docker compose** to launch services in different Environments and group the **docker containers** into clusters based on their services and mange those clusters with the **swarm Manger**.
* Created Ansible cloud modules for interacting with Azure services which provides the tools to easily create and orchestrate infrastructure on Azure and automated cloud-native applications in Azure using Azure microservices such as Azure functions and Kubernetes on Azure .
* Used **Ansible** and **Ansible Tower** as Configuration management tool, to automate repetitive tasks, quickly deploys critical applications, and proactively manages change.
* Used **Ansible Tower,** which provides an easy-to- use dashboard and role-based access control, so that it is easier to allow individual teams access to use **Ansible** for their deployments.
* Working with **Ansible tower**to manage Web Applications, Config Files, Data Base, Commands, User Mount Points, Packages and for running **playbooks**stream in real-time and amazed to see the status of every running job without any further reloads.
* Written **Ansible handlers** with multiple tasks to trigger multiple handlers and to decouple handlers from their names, making it easier to share handlers among Playbooks and Roles.
* Used **Nagios**as a monitoring tool to identify and resolve infrastructure problems before they affect critical processes and worked on Nagios Event handlers in case of automatic restart of failed applications and services.
* Using **GIT** to manage source code, Created Branches, Labels and performed Merges in Stash, analyzing and resolve conflicts related to **merging** of source code for **GIT.**
* Used the version control system **GIT** to access the repositories and used in coordinating with CI tools.

**Environment:** Azure DevOps, Kubernetes, Docker ACS & AKS, Splunk, Terraform, Ansible, Jenkins, Git, Azure Boards, Maven, SonarQube, Python, PowerShell, YAML, Visual Studio Code, Tomcat, Nginx, Linux, Windows servers 2012/2016

**CLIENT: GILEAD SCIENCES| CA JULY 2018 – SEPTEMBER 2019**

**DEVOPS ENGINEER | AWS**

**RESPONSIBILITIES**

* Designed and converted existing **AWS** infrastructure to 'Server-less' architecture deployed via **Terraform** which includes **AWS** **Elastic Beanstalk** for app deployments and worked on **AWS** **Lambda** with **Amazon Kinesis** and integrated **Dynamo DB** using **Lambda** for value storage and for streams backup.
* 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.**
* Worked on creation of **Docker containers**, **Docker files** and **Docker consoles** for managing the application life cycle, configured deployment automation of all the microservices to pull image from the private docker registry and deploy to **docker swarm** **cluster**
* Involved in providing the highly available EC2 Instances using Terraform and cloud formation and wrote new plugins to support new functionality in Terraform and Configured Elastic Load Balancers with EC2 Auto scaling groups.
* Involved in creating alarms in Cloud watch service for monitoring the servers' performance, CPU Utilization, disk usage etc.
* Designed AWS CloudFormation templates to create custom sized VPC, subnets, NAT to ensure successful deployment of Web applications and database templates.
* Created load balancers (ELB) and used Route53 with failover and latency options for high availability and fault tolerance.
* Creating Lambda function to automate snapshot back up on AWS and set up the scheduled backup.
* Configured AWS Identity and Access Management (IAM) users and groups for improved login authentication.
* Setting up databases in AWS using **RDS**, storage using **S3 buckets** and configuring **instance** backups to **S3 bucket**.
* Used Bash and Python included Boto3 to supplement automation provided by Ansible and Terraform for tasks such as encrypting EBS volumes backing AMIs and scheduling Lambda functions for routine AWS tasks.
* Created a Python Script to stop all the instances with a specific tag in AWS Instance using Lambda functions and made it into Cloud Watch Scheduler to schedule it every night.
* Utilize AWS CLI to automate backups of ephemeral data-stores to S3 buckets, EBS and create nightly AMIs for mission critical production servers as backups.
* Wrote Ansible playbooks to launch AWS instances and used Ansible to manage web applications, configuration files, used mount points, packages and Worked on playbooks for Ansible in YAML.
* Building/Maintaining Docker container clusters managed by Kubernetes, Linux, Bash, GIT, Docker, on AWS and utilized Kubernetes and Docker for the runtime environment of the CI/CD system to build, test deploy.
* Used Kubernetes for creating new Projects, Services for load balancing and adding them to Routes by accessing from outside, created Pods through new application and controlling, scaling and troubleshooting pods through SSH.
* Used **Docker** in build automation pipeline and production deployment & implemented the setup for Master- slave architecture to improve the Performance of Jenkins. Used **Jenkins** for Continuous Integration and deployment into **TomCat**Application Server.
* Created the **Kubernetes Clusters** under Pod networks to group the **Docker containers**. Used **Kubernetes** and **Docker** to maintain the runtime environment for the **CI/CD system** to build, test and deploy the applications across different environments.
* Integrated **Jenkins** with **Docker** container using Cloud bees **Docker** pipeline plugin and provisioned the EC2 instance using Amazon EC2 plugin.
* Integrated **Docker** container-based test infrastructure to Jenkins **CI/CD** test flow and executed build environment by integrating with Git and Jira to trigger builds using WebHooks and Slave Machines.
* Worked on **AWS costs**by writing the **Ansible Playbooks**for auto **Start/Stop**of AWS resources at the time of the day by triggering it from **Jenkins**and knowledge on **RHEL**on **Ansible Playbooks**, **Modules**and **Roles**.
* Automated the tasks using Ansible playbooks, Shell scripting and Python and migrated servers with the required configurations changes and testing and deploying the machines using Ansible commands
* Used **Ansible**to Setup/teardown of **ELK**stack ( **Elasticsearch, Log stash, Kibana**) and troubleshoot the build issues with **ELK**and work towards the solution.
* Designed various **Jenkins**jobs to continuously integrate the processes and executed **CI/CD**pipeline using **Jenkins**, **Ansible Playbooks**and **Ansible Tower.**
* Used **Git** version control to manage the source code and integrating **Git** with **Jenkins** to support build **automation** and integrated with **Jira** to **monitor** the commits.
* Involved in setting up **CICD** pipeline integrating various tools with Jenkins to build and run **Terraform** jobs to create infrastructure in **AWS**

**Environment:** AWS (IAM, EC2, S3, EBS, Glacier, ELB, CloudFormation, CloudWatch, CloudTrail, SNS, SQS, Route53, RDS), OpenStack, Kafka, Git, Ansible, Terraform, SDN, Splunk, Bash, Shell, Dynamo DB, RHEL 4/5/6, CentOS, Apache Tomcat.

**CLIENT: SYNDICATE BANK, HYDERABAD, INDIA MARCH 2016 – JANUARY 2018**

**DEVOPS ENGINEER**

**RESPONSIBILITIES**

* Created functions and assigned roles in AWS Lambda to run python scripts and wrote Lambda functions in Python for AWS Lambda and invoked Python scripts for data transformations and analytics on large data sets in EMR clusters and AWS Kinesis data streams
* Created Chef Cookbooks to provision Apache Web servers, Tomcat servers, Nginx, Apache Spark and used GZIP to forward compressed files to destination node/instances.
* Worked on configuring EC2 instances by writing Terraform templates, Chef cookbooks, recipes and pushed them on to Chef server.
* Writing shell scripts to handle server processing Migrating applications from .Net to Python Monitoring applications/servers performance with Splunk, Jenkins, etc.
* Implemented a production ready, load balanced, fault tolerant, auto scaling **Kubernetes AWS**infrastructure and **microservice**container orchestration.
* Automated the cloud deployment using **Chef, Python** and **AWS Cloud Formation** Templates. Used **Chef** for unattended **bootstrapping** in **AWS**.
* Involved in continuous improvement to agile software development teams by working with **Jenkins**under the **CI/CD**pipeline. Integrated **Ant, Nexus, Jenkins, Confluence**and **JIRA**and Used **Git** with **Jenkins** to integrate to automate the code checkout process.
* Installed and Configured the Nexus repository manager for sharing the artifacts within the company.
* Managed **Nexus** to save the results of the **ANT** projects in **jar format** which is deployed by **Jenkins**. And also used **Nexus** to share the snapshots and releases of internal projects.
* Worked on Designing and Setting-Up the Splunk Architecture in the organization and creating Applications on Splunk to analyze the Big Data.
* Used both Splunk for monitoring and analyzing log information. Worked on in Troubleshooting Splunk search, quotas, monitor Inputs, WMI Issues, Splunk crash logs and Alert scripts.
* Implemented **Docker** - **maven**-plugin in **Maven pom.xml** files to build **Docker images** for all **microservices** and later used **Docker File** to build the **Docker** images from the **Java jar files** also Created **Docker** images using a **Docker File**, worked on **Docker container** snapshots, **removing** images and managing **Docker volumes**
* Worked with **Docker swarm** and compose in building distributed Docker containers and able to script Docker steps that can be used for automated deployment for developers in Local, staging and production after a continuous integration build.
* Worked on Docker container snapshots, attaching to a running container, removing images, managing Directory structures, and managing containers. Built Docker files to create containers using Docker engine and Docker Machine environments, to deploy the microservices oriented environments for scalable applications.
* Prepared Continuous Integration (CI) and Continuous Delivery (CD) process implementation using Jenkins along with Python and Shell scripts to automate routine jobs.
* Used **Git**for source code version control and integrated with **Jenkins for CI/CDpipeline**, code quality tracking and user management with build tools **Maven**.
* Creating Docker images from Centos Linux as per requirement of the product/service and storing those Docker images to the ECR or Nexus Repository.
* Written **Chef Cookbooks**for various DB configurations to modularize and optimize end product configuration, converting production support scripts to **Chef Recipes**and **AWS**server provisioning using Chef Recipes.
* Worked on installation & configuration of Jenkins to support various Java builds and Jenkins plugins to automate continuous builds and publishing **DOCKER**images to the Nexus repository.
* Created/configured many jobs in Jenkins to help CI/CD pipelines using many plugins like **DSLplugin**, parameterized **Trigger plugin**, pipeline view plugin, Extended email plugin etc
* Worked on using Collections in **Python** for **manipulating** and **looping** through objects, Created and maintained the **Python** deployment scripts for **Tomcat**web application servers
* Worked on **Database** Administration activities such as installation, database creation, sizing, backup and recovery, performing turning and capacity planning

**Environment:** Linux, Windows, Unix shell scripting, PHP, Python, Oracle, Progress, MySQL, Jenkins v 1.532/1.651/2.8 , Maven, ANT, JIRA, AWS, Docker, Jenkins, GIT, Splunk, Linux (centos), Python, Chef,

**CLIENT: HINDUSTAN UNILEVER, MUMBAI, INDIA MAY 2015 – FEBRUARY 2016**

**SYSTEM ANALYST**

**RESPONSIBILITIES**

* Designed, wrote and maintained systems in **Python scripting** for administering by using **Jenkins** as a full cycle **continuous delivery** tool involving package creation, distribution, and deployment onto **Tomcat application** servers via shell scripts embedded into **Jenkins** jobs.
* Integrating Defect tracking tool JIRA with Confluence and Jenkins CI server for identifying, logging and, tracking and documenting defects/issues in real-time.
* User account management and administration Troubleshooting User’s login and home directory related issues reset password and unlock user accounts in **Linux**Systems.
* Developed automation scripting in **Python**(core) using **Puppet**to deploy and manage **Java**applications across **Linux**servers.
* Developed automated scripts using **Python, Shell Scripting** for data collection and transfer to databases.
* Automated installing the Linux packages and administering Linux service using **Chef** and automating product installation configuration.
* Worked on configuring Puppet master servers and installing Puppet client software on Linux servers and deployed Puppet, Puppet Dashboard, and Puppet DB for configuration management to existing infrastructure
* Maintained build related scripts developed in **ANT, Python and shell.**Modified build configuration files including **Ant’s build.xml**.
* Written Chef Cookbooks and recipes in Ruby to Provision several pre-prod environments consisting of Cassandra DB installations, WebLogic domain creations and several proprietary middleware installations.
* Used ANT as a build tools on java projects for the development of build artifacts on the source code and Worked with development team to migrate **Ant scripts to Maven.**

**Environment:** Ubuntu, Linux, Jenkins, Mesos, Chef, Puppet, ANT, GIT, AWS, Docker, Bash, Shell, Tomcat and Apache, Maven

**CLIENT: RADIANT INFO SYSTEMS, HYDERABAD SEPTEMBER 2014 – APRIL 2015**

**SYSTEM ANALYST**

**RESPONSIBILITIES**

* Installation and administration of Linux systems, upgrading and configuring RedHat Linux 5.x, 6.x,7.x using Gold Disk Image.
* Management and Implementation of Linux (Centos, RedHat) and Windows (windows Server 2008/2012 R2), Middleware (Apache, PHP, Tomcat, JBoss and Fuse), and java applications
* Build out server automation with Continuous Integration - Continuous Deployment tools like Jenkins/Maven for deployment and build management system.
* Creating users, groups and setting file permissions to different groups and users and created & migrated SSH keys for application ID to enable sftp/scp capability.
* Creating a change requests, work orders and problem tickets using BMC Remedy tool and getting approvals from process owners.
* Handled the scheduling tasks (cron jobs and task scheduler) for the scripts for various purposes.
* Configured kickstart server and updating/applying patches to the servers using Red Hat Satellite server.
* Install host firewall IP Tables for RHEL and Centos.
* Managing Automation toolset such as Puppet by writing Manifests and modules.
* Bash Shell scripting for various automation of system administrative tasks.
* Delivery, Release planning, procedures, scripting automation.
* Responsible for design and maintenance of the Subversion (SVN) Repositories and the access control strategies.

**Environment**: Linux (Red Hat Enterprise, Cent OS), Windows 2000/NT, Subversion, Python, shell scripting, Jenkins, BMC Remedy.

**EDUCATION**

**MASTER OF SCIENCE: DATA SCIENCE**