**Pavan**

**Azure Devops Engineer**

**Mail id:**[**pavan.devops009@gmail.com**](mailto:pavan.devops009@gmail.com)

**Contact no: 972-525-9302**

**Professional Summary:**

* Cloud DevOps engineer with over 10+ years’ experience in **Cloud**(**Azure**, **AWS**, **GCP**), **DevOps**, **Configuration**

**management**, **Infrastructureautomation**, **ContinuousIntegration** and **Delivery(CI/CD),** can implement

effective strategies for N - Tier application development in both Cloud and On-premises environments. Experience

in dealing with Unix/Linux and Windows server administration.

* Expertise in Architecting and Implementing **AzureService** Offering, such as **Azurecloudservices**, Azure

**storage**, **IIS**, **AzureActiveDirectory** (**AD**), **AzureResourceManager** (**ARM**), **AzureStorage**, **Azure**, **Blob**

**Storage**, **AzureVMs**, **SQLDatabase**, **AzureFunctions**, Azure **ServiceFabric**, **AzureMonitor**, and **Azure**

**ServiceBus**.

* Hands on experience in Designing and configuring **AzureVirtualNetworks** (**VNets**), **subnets**, **Azurenetwork**

**settings**, **DHCPaddressblocks**, **DNSsettings**, **securitypolicies** and **routing**.

* Designed, deployed, maintained and implemented cloud solutions using Azure and underlying technologies using

**IAAS**&**PAAS** services as part of **SDLC**/**AgileMethods**.

* Experience in setting up alerts and deploying multiple dashboards for individual applications in **Azure** Kubernetes

(AKS) clusters using tools like Prometheus and **Grafana**.

* Expertise with Terraform templates for provisioning Infrastructure like **VirtualNetworks**, **LoadBalancers**,

**StorageAccounts**, **VirtualMachines**, **VirtualMachineScaleSets**, **AzureKubernetesCluster** (**A**KS), **Key**

Vaults and Log Analytics Workspace in **MicrosoftAzure** using **Terraform** modules.

* Experience in designing, deploying and maintaining various multitude applications utilizing AWS services stack

including **EC2**, **S3**, **EBS**, **EFS**, **ElasticBeanstalk**, **Route53**, **VPC**, **CloudFormation**, **ElasticCache**, **Cloud**

trail, SNS, **SQS** focusing on **HighAvailability**, **Fault** - **tolerance** and Auto **Scaling**.

* Built end to end **CI/CD** Pipelines in Jenkins to retrieve code, compile applications, perform tests and push build

artifacts to **Nexus**.

* Well versed with updating **Azure** Images in **AzureCompute** Galleries using Packer and update these Image

references for **VirtualMachines**, **VirtualMachineScaleSets** using **Terraform** across all environments.

* Experienced working with several **Docker** components such as **DockerEngine**, **DockerMachine**, Creating

**Docker** Images, **DockerCompose**, **DockerHub** and handling multiple images for middleware installations and

domain configuration.

* Extensively used **OpenShift** for container orchestration with **Kubernetes**, Container Storage, and automation, to

enhance container platform multi-tenancy.

* Expertise in Setting up **Kubernetes** (**k8s**) Clusters for running microservices and pushed microservices into

production with Kubernetes backed Infrastructure.

* Experience in Configuration management tools such as **Chef**, **Ansible**, and **Puppet**. Wrote **ChefRecipes** and

cookbooks in Ruby scripting.

* Experience in writing **Infrastructurea**s a **code** (**IaC**) in **Terraform**, **Azure resource management**, **AWS Cloud**

**formation**.

* Participated in creating, validating and reviewing solutions for migrating to **Azurecloud** environment also

conducting Proof of Concept for the latest **Azure** cloud-based service by using simplified automation using **JSON**

Templates and **PowerShell**.

* Proficient in installing, configuring and administering **JenkinsCI** tool on Linux machines and used Jenkins

Pipelines to drive all Microservices builds out to the **DockerRegistry** and then deployed to **Kubernetes**,Created

Pods and managed using **Kubernetes**.

* Experience in implementing **CI/CD** pipeline using **AzureDevOps**(**VSTS**,**TFS**) in both cloud and on-premises

with **GIT**, **MSBuild**, **Docker**, **Maven** along with **Jenkins** plugins.

* Extensively worked on **Bamboo** by installing, configuring and maintaining the purpose of **Continuous**

**Integration(CI)** and End-to-End automation for all build, deployments and implementing **CI/CD** pipelines using

Jenkins.

* **Good experience** in deploying and configuring **Elasticsearch**, **Logstash**, **Kibana** (**ELK**) and **AWSKinesis** for

log analytics and skilled in monitoring servers using **Nagios**, **Splunk**, AWS **CloudWatch**, **AzureMonitorELK**.

* Experienced in migrating on-premises **DevOpsplatform to Azure CI/CD processes by utilizing ARM**

**templates, Azure DevOps, App services** such as Repos, **TestPlans**, **Pipelines**, **WebApps**, and **Application**

**Insights**.

* Experienced with **AnsibleTower**, which provides an easy-to-use dashboard and role-based access control in

developing **Ansibleplaybooks** for managing the application/OS configuration files in **GitHub**, integrating with

**Jenkins**, and Verifying with **Jenkins** plugins, deploying the application in **Linux** environment.

* Skilled with **Python**, **Bash**/**Shell**, **PowerShell**, **Ruby**, **Perl**, **JSON**, **YAML**, **Groovy**. Developed Shell and **Python**

**Scripts** used to automate day to day administrative tasks and automation of the build and release process.

* Worked with **Helm** package manager in creating custom charts for various applications for implementing custom

**HelmCharts**.

* Good knowledge on system administration, configuration and customization of Atlassian tools like Bitbucket,

**JIRA** and **Confluence**.

* Well versed in configuring an automated tool **Puppet** that included installation and configuration of **Puppet** and

written manifests and modules for a puppet to do tasks like application deployment, configuration and create file

systems.

* Proficient with source controller tools like **GitHub** (**GIT**), **Subversion** (**SVN**) and software build tools - **Apache**

**Maven** and **Ant**.

* Experienced in database technology and warehousing such as **MySQL**, **PostgreSQL**, **Aurora**, and **Redshift**

including **NoSQL** databases such as **Apache**, **MongoDB**, **Cassandra** and **DynamoDB**.

**Technical Skills:**

* Azure, AWS, Open Stack.
* Docker, Kubernetes, OpenShift, Mesos.
* Jenkins, Hudson, Chef, Ansible, Puppet, Ant, Maven.
* GIT, CVS, SVN.
* JIRA, Confluence, Bit bucket, Bamboo.
* Nagios, Splunk, ELK, New Relic, Data Dog, Cloud watch, Dynatrace, Prometheus and Grafana.
* Java/J2EE, C, C++, SQL, JAVASCRIPT, WLST, Bash/Shell, Perl, Python, Ruby, power shell, Json, YAML,

Groovy & Go Lang.

* My SQL, MS Access, Oracle 9i/10g/11g, DB2, NoSQL (MongoDB, DynamoDB)
* WebLogic 9.x/10.x, WebSphere 6.x/7.x/8.x, JBOSS 4.1
* Apache HTTP, Nginx, Apache TOMCAT
* DNS, DHCP, SMTP, HTTP, SNMP
* Linux (Red Hat 4/5/6/7), UBUNTU, UNIX, WINDOWS.

**Professional Experience:**

**Client:Experian, CADuration: July 2021 – TillDate**

**Role: Azure Devops Engineer**

**Description:** The goal of this project was to work with the IT team to consolidate the company's DevOps tools, which had

previously been distributed across several service providers, into a single platform using Azure DevOps, Azure VNETs,

and Azure virtual machines. Terraform and Ansible were used to help with the migration process. As a consequence, a new

DevOps pipeline on the Azure platform was created.

**Responsibilities:**

* Configured **AzureVirtualNetworks**,**subnets**, **DHCPaddressblocks**, **Azurenetworksettings**, **DNSsettings**,

**securitypolicies** and **routing**. Also, deployed **AzureIaaSvirtualmachines** and **Cloudservices** (**PaaS** role

instances) into secure **VirtualNetworks** and **subnets**.

* Managed **AzureInfrastructure**, **AzureWebRoles**, **WorkerRoles**, **SQLAzure**, **AzureStorage**, **AzureAD**

**Licenses**, **Office365**, **VirtualMachineBackup** and Recover from a **RecoveryServicesVault** using **Azure**

**PowerShell** and Portal.

* Design and implement scenarios using **AzureQuickStartTemplates**, create **AzureManagedApps**, **Microsoft**

**AzureComputeinfrastructure** and **networking**, **AzureActiveDirectory**, **AzureDataStorage** and **AzureData**

**AccessStrategy**, **WebAppsapplications** that use mobile services and notifications to design advanced

applications.

* Created secure **VNets** and subnets using **AzureIaaSvirtualmachines** (**VMs**) and **AzureCloudservices** (**PaaS**

role instances); developed **VNets** and subscriptions to adhere to **AzureNetworkLimits**.

* Utilized **OpenShift** for container orchestration with **Kubernetes**, **ContainerStorage**, **automation**, to enhance

container platform multi-tenancy.

* Configured applications that run multi-container **Docker** applications by utilizing the **Docker** - **Compose** tool

which uses a file configured in **YAML** format.

* Enhanced **Kubernetes** to manage containerized applications using its **nodes**, **Config**-**Maps**, **selector**, **Services** and

deployed application containers as Pods.

* Creating **DockerImages**, **Compose**, **DockerHub** and handling multiple images for **middlewareinstallations** and

**domainconfiguration**.

* Continuous Integration using **Jenkins** and troubleshoot the deployment build issues using the triggered logs.
* Implemented **Docker -maven-plug-inMavenpom.xml files** to build **Dockerimages** for all microservices and

later used **Dockerfile** to build the **Dockerimages**.

* Implemented a **CI/CD** pipeline with **Docker**, **Jenkins** (**TFS**Plug-in installed), **TeamFoundationServer** (TFS),

**GitHub** and **Azure** Container Service, whenever a new **TFS/GitHub** branch gets started, **Jenkins**, our

**ContinuousIntegration** (**CI**) server, automatically attempts to build a new **Docker** container from it.

* Played a significant role in using **Terraform** along with **Jenkins** and Packer tools to create **custommachine**

**Images** in **AzureComputeGalleries** and used **Ansible** to install the software dependencies once the infrastructure

was provisioned.

* Assisted in setting up **AzureMonitoringtools** like **AzureLogAnalytics** and **AzureApplicationInsights** for

resource level logs of entire subscription and store the logs in the **EventHub**.

* Configured **VM's** availability sets using **AZURE** portal to provide resiliency for **IaaS** based solution and scale sets

using **AZUREResourceManager** to **managenetworktraffic**.

* Setting up end-to-end pipelines in **AzureDevOps** using **YAML** scripts and configuring service connections in

across multiple Projects in **AzureDevOps** Organization.

* Used **Ansible** to Setup/teardown of **ELK** stack (**Elasticsearch**, **Logstash**, **Kibana**) and troubleshoot the build

issues with **ELK** and work towards the solution.

* Creating resources in **MicrosoftAzure** such as **VNETs**, **VirtualMachines**, **ApplicationGateways**, **EventHubs**,

**StorageAccounts**, **AzureKubernetesCluster**, **KeyVaults**, and **PostgreSQL** across all environments using

reusable **Terraform** Modules.

* Configured **Bitbucket** for **ContinuousBuilds**, **Test** and **Deploy** applications to **AzureCloud** and created

**Bitbucket** repositories across multiple projects as well as generating PAT tokens to access them.

* Responsible for writing **AnsiblePlaybooks** to Patch and Reboot the **Ubuntu**, **Centos**, **RedHat** and **Windows**

**VirtualMachines** across all environments and written **AnsiblePlaybooks** to send the final report to the team

through **Email**.

* Deploying and configuring **Prometheus** to monitor **Kubernetes** nodes with **node-exporter,** monitor **Kubernetes**

**API** and resources with **Kube**-**state**-**metrics**.

* Resolved state lock issues in the production environment of **Terraform** state files in **Azure**, also being proficient

in importing **Terraform** resources into state files to resolve the issues.

* Built and managed a highly available monitoring infrastructure to monitor different application servers and its

components using **Nagios** with **Puppet** automation. Involved in editing the existing **ANT** files in case of errors or

changes in the project requirements.

* Provided POC's to provision the **AWSinfrastructure** using **Ansibleplaybooks** which includes checking the

health status, monitoring & gathering facts about the **EC2**-**instances**, **ELB**, **AutoScalinggroups**& other

resources.

* Designed, wrote and maintained systems in **Python** scripting for administering **GIT**, by using **Jenkins** as a full

cycle continuous delivery tool involving package creation, distribution, and deployment onto **Tomcatapplication**

servers via shell scripts embedded into **Jenkins** jobs.

* Utilized **JIRA** for creating bugs tickets, storyboarding, pulling reports from dashboard, creating and planning

sprints.

**Environment:**Azure Virtual Networks, subnets, DHCP address blocks, DNS, IaaS, PaaS, Azure Infrastructure, Azure

Web Roles, Worker Roles, SQL Azure, Azure Storage, Azure AD Licenses, Office365, Virtual Machine Backup, Azure

PowerShell, YAML, ELK, Ansible, VNETs, Virtual Machines, Application Gateways, Event Hubs, Storage Accounts,

Azure Kubernetes Cluster, Key Vaults, CI/CD,Bitbucket, Ansible, Prometheus, Kubernetes, Terraform, Nagios, Puppet,

AWS, EC2, ELB, ANT, Auto Scaling groups, Python, GIT, Tomcat application, Jenkins, JIRA.

**Client:Vanguard - Scottsdale, AZDuration: Nov 2019 - June 2021**

**Role: Azure Devops Engineer**

**Description:** By automating the deployment of IaaS and PaaS Azure resources using the Azure CLI in Power Shell,

resource group deployment time as compared to Azure RM templates was reduced from hours to minutes. App services,

SQL servers, virtual machines, Azure Docker container registries, service fabric mesh apps, and Cosmos DB accounts were

among the resources. To resolve Docker/Service Fabric Mesh compatibility concerns, I oversaw many build servers

running several OS systems.

**Responsibilities:**

* Managed **AzureInfrastructureAzureWebRoles**, **WorkerRoles**, **VMRole**, **AzureSQL**, **AzureStorage**,

**AzureADLicenses**, **VirtualMachineBackup** and **Recover** from a **RecoveryServicesVault** using **Azure**

**PowerShell** and **AzurePortal**.

* Enhanced **TerraformTemplates** to automate the **AzureIaaSvirtualmachines** using **Terraform** modules and

deployed virtual machine scale sets in production environment.

* Configured **AzureMulti**-**FactorAuthentication** (**MFA**) as a part of **AzureAD** Premium to securely authenticate

users and creating custom **Azuretemplates** for quick deployments and advanced **PowerShell** scripting. Deployed

**AzureSQLDB** with Sync to standby database in another region & Fail over configuration.

* Dealt with Windows **AZUREIaaS** - **VirtualNetworks**, **VirtualMachines**, **CloudServices**, **ResourceGroups**,

**ExpressRoute**, **TrafficManager**, **VPN**, **LoadBalancing**, **ApplicationGateways**, and **Auto**-**Scaling**.

* Responsible for configuring **AzureIdentityAccessManagement** (**IAM**) for improved login Authentication and

**createdGroups**, adding Users and Service Principals to those groups in **AzureActiveDirectory** (**AD**).

* Configured and maintained **AzureStorageFirewalls** and **VirtualNetworks** that use **VirtualNetworkService**

Endpoints to enable administrators to create network rules that allow traffic only from specific **VNets** and subnets

thereby creating a secure network boundary for their data.

* Enhanced **ARM** templates (**JSON**) to deploy update and delete multiple **Azureresources** and involved in

migrating on-premises resources to **Azure** with **AzureSiteRecovery** (**ASR**) and **AzureBackups**.

* Assisted in troubleshooting **VirtualMachines**, **VirtualMachineScaleSets** through Serial Console and

Connection Troubleshoot when there is a problem with **SSHlogging**.

* Configured **AzureDevOps** Pipeline to pull the latest **DockerImages** from **DockerHub**, tag the image with a

Build Number and push the image to **AzureContainerRegistry** (**ACR**) in **Azure**.

* Utilized **Terraform** modules and automated infrastructure management across multiple environments and created

Infrastructure in **MicrosoftAzure** using a reliable version of **Terraform**.

* Implemented **PowerShell** scripts to deploy new storage accounts across all the **Azure** Regions to reduce latency

and increase redundancy.

* Used **Ansible** for creating multiple playbooks for Resource creations and **SQLserver**, **clusterserver** and my **SQL**

installations.

* Created and configured **HTTP** Triggers in **AzureFunctions** with application insights for monitoring and

performing load testing on applications using the **VisualStudioTeamServices** (**VSTS**) also called as Azure

DevOps Services.

* Deployed windows **Kubernetes** (K8s) cluster with **AzureContainerService** (**ACS**), **AzureCLI** and Utilized

**Kubernetes** and **Docker** for the runtime environment of the **CI/CD** system to build, test and Octopus Deploy.

Used**OpenShift** for container orchestration with **Kubernetes** container storage.

* Designed various **Jenkins** jobs to continuously integrate the processes and executed **CI/CD** pipeline using

**Jenkins**,**AnsiblePlaybooks** and **AnsibleTower**.

* Performed configuration and management operations across multiple remote hosts using **Ansibleroles**, **Ansible**

**files**, **Ansiblehandlers**, **Ansibletasks**, **Ansibletemplates** and **Ansiblevariables** that were downloaded and

managed from **AnsibleGalaxy**.

* Responsible for creating **AzureResourceManager** (**ARM**) templates and **PowerShell** scripts to automate

deployment and resource provisioning across multiple **Azuresubscriptions**.

* Identifying and troubleshooting performance, connectivity and other issues for the applications that were hosted in

**MicrosoftAzure** platform.

* Setting up the **AzureKubernetesService** (**AKS**) to deploy spring applications, configure the **AzureContainer**

**Registry** (**ACR**) to store **DockerImages** and manage the **AzureKubernetescluster**.

* Proficient in using **GITWebhooks** and **PollSCM** with **Jenkins** to automate **Jenkins** tasks and responsible for

writing **Groovyscripts** for **Jenkins** Pipelines.

* 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.

* Performed **AzureKubernetes** (**AKS**) **Cluster** upgrades at Productivity Environment once the testing was

successfully completed at Lower Environments.

* Monitored system performance using **Nagios**, maintained **Nagios** servers and added new services & servers and

also built and managed a highly available monitoring infrastructure to monitor different application servers like

**JBoss**, **ApacheTomcat** and its components using **Nagios**.

**Environment:**Azure Infrastructure Azure Web Roles, Worker Roles, VM Role, Azure SQL, Azure Storage, Azure AD

Licenses, Virtual Machine Backup, Azure PowerShell, Azure Portal, Azure IaaS, Terraform, IAM, JSON, SQL server,

Ansible, HTTP, VSTS, ACS, Azure CLI, Kubernetes, CI/CD, OpenShift, Jenkins, ARM, PowerShell, AKS, ACR, Docker,

GIT, Groovy, Jira, Nagios.

**Client:Fidelityinvestments**, **Albany**, **NYDuration: April 2018 - Oct 2019**

**Role: Devops Engineer**

**Description:** Responsible for managing the Azure clouds resources and the Continuous Integration and Continuous

Deployment (CI/CD) pipeline for the rapid creation of strong applications. Being a part of the on boarding team, it is me

duty to carry out the DevOps transformation by collaborating with agile teams to move apps on the Azure platform.

**Responsibilities:**

* Provisioning of **IAAS**, **PAASvirtualmachines**, **webroles**, and **workerroles** on **MicrosoftAzureResource**

**Manager** was completed.

* Created **AZUREVMs**, **storageaccounts**, **VHDs**, **storagepools**, migrating on premise servers to **AZURE** and

creating availability sets in **AZURE**.

* Used **AzureKubernetesService**, **AzureTrafficManager**, and **AzureVNets** to complete the deployment of a

cloud-native application on **MicrosoftAzure**.

* Developed new **Terraform** plugins to provide new functionality and participated in the use of **Terraform** to move

traditional and monolithic systems to **Azure**.

* Migrating on premise to **WindowsAzure** using **AzureSiteRecovery** and **Azurebackups**.
* Creating and maintaining containerized micro services and configuring/maintaining private container registry on

**MicrosoftAzure** for **HostingImages** and using **WindowsActiveDirectory**.

* Implemented **CI/CD** pipeline for **AKS** using **Helmcharts** in **AzureDevOps**.
* Managed **Kubernetescluster** in **Azure** and created an **AKScluster** in the **Azureportal** with the **AzureCLI**, also

used template driven deployment options such as **ResourceManagerTemplates** and **Terraform**.

* Created custom sized **VPCs**, **Subnets**, and **NAT** using **AWS** Cloud Formation templates to ensure the successful

deployment of Web applications and Database templates.

* Built jobs in **Jenkins** to build **AWS** infrastructure from **GitHub** repositories containing **Terraformcode**, managed

builds in **Jenkins** and integrated tools from different **DevOps** stacks like **Jfrog**, **SonarQube**, and **chef**.

* Monitored **Helm** package releases, produced repeatable builds of the **Kubernetesapps**, and maintained the

**Kubernetesmanifestfiles**. On a node cluster using **Kubernetes**, scheduled, deployed, and managed container

replicas.

* Worked with **AWS** services such as **EC2**, **VPC**, **RDS**, **CloudWatch**, **CloudFront**, **Route53**, Configuring and

Networking of **VirtualPrivateCloud** (**VPC**).

* Creating **CI/CD** pipeline to enable end to end setup of build & deployment for applications and perform tests and

deploy code to multiple environments.

* Implemented an **ELK** cluster for enterprise logging and Maintained **Elasticsearch**, **Logstash**, and **Kibana**.
* Used **Terraform** to create and manage **infrastructure as code (IAC)** and ensure that the infrastructure is **reliable**,

**scalable**, and **secure**.

* Developing modules and **Puppet** manifests to automate the deployment process and integrating **Puppet** manifests

into **Jenkins** tasks for a **continuousdelivery(CD)** architecture.

* Worked with container-based deployments using **Docker**, Worked with **Dockerimages**, **DockerHub** and

**Docker**-**registries** and **Kubernetes**.

* Monitoring and deployment by using technologies like **CHEF**, **Puppet** and Responsible for designing and

deploying the best **SCM** processes and procedures with **GitHub** and **GIT**.

* Maintained **PowerShell** scripts for **Maven** builds that were connected to building new build configuration files and

making changes to existing ones.

* Developed **ChefCookbooks**, recipes, Roles, attributes in Ruby to maintain, update configurations and deploy.
* Worked on data management tools like **Splunk**, **ELK** and used **Splunk** to monitoring and metric collection for

applications in a cloud-based environment.

**Environment:**IAAS, PAAS virtual machines, web roles, AZURE VMs, storage accounts, VHDs, storage pools, Azure

Kubernetes Service, Azure Traffic Manager, Azure VNets, Terraform, CI/CD, Helm charts, Azure CLI, Jenkins, GitHub,

Jfrog, SonarQube, chef, ELK, IAC, Puppet, Docker, Puppet, GIT, SCM, PowerShell, Maven, Splunk.

**Client:Launchship Technology Solutions Pvt. Ltd.Duration: March 2014 - Dec 2016**

**Role:Build & Release engineer**

**Description:**Coordinating with developers to deliver software applications and understand learn latest build and

deployment tools as per the requirements and onboard current application on to new tools and train development teams

accordingly.

**Responsibilities:**

* Responsible for all software builds, including continuous **integrationbuilds**, **nightlybuilds**, and **releasebuilds**.
* Developed **Openstack** infrastructure with automation and **configurationmanagementtools** such as **Ansible**,

**Puppet**, or custom-built cloud hosted applications and used **Ansible** for continuous integrations.

* Created Jenkins job to build the artifacts using **maven**, deploy the **Terraform** templates to create the **stack**.
* Deployed **builtartifacts** to **WebSphere** application server using maven and **python** scripts.
* Developed build and deployment scripts using **ANT** and **MAVEN** as build tools in **Jenkins** to move from one

environment to other environments.

* Deployed **Puppet**, **PuppetDashboard** and **PuppetDB** for configuration management to existing infrastructure.
* Used **Jenkins** for the continuous integration process, automation and assisted in the code coverage and analysis by

generating reports.

* Implanted a **continuousDelivery** pipeline with **Docker**, **Jenkins** and **GitHub**. Whenever a new **GitHub** branch

gets started, **Jenkins** our continuous Integration server automatically attempts to build a new **Docker** container

from it.

* Utilized **Puppet** and **Puppetdashboard** for **ConfigurationManagement** of hosted Instances within **AWS**.
* Supervised and guided team to establish quality and CM standards as per **agile** methodology and implemented the

**agile** templates in **TFS**.

* Automating the regular tasks in Subversion and streamlining the process of access to **SVN**.Migrated Projects from

**SVN** to **GIT**.

* Setting up **ChefInfra**, **Bootstrappingnodes**, creating and uploading **ChefRecipes**, **ChefNode** convergence in

**ChefSCM**.

* Used **Hudson**/**Jenkins** to perform automated Builds and Deployments.
* Used **Perforce** as **VersionControlSystem** for maintaining the discipline required working with the codes on

different files.

* Administer and support **TeamFoundationServer**, Managed security and configuration for **TFS** projects. Worked

with **TFScodebranching**, **merging** and **versioning**.

* Specialized in automating tasks and process through scripts, implementing branching models, implementing

**Build**/**Releasesolutions**.

* Provided Build Scripts for Database Projects to **DeployDB** and populate Test data in **DB** for various

environments.

* Created **SubversionHooks** for **JIRA** integration and enforcing naming conventions. Installed and configured

**Subversion** repositories according to the custom needs of the development team and responsible for providing

administration support doing **codemigration**, **Harvestuser**/**applicationsupport**&**TFS**.

**Environment:**Open stack, Ansible, Puppet, maven, Terraform, WebSphere, ANT, Puppet DB, GitHub, agile, TFS, SVN,

GIT, Chef, Hudson, Jenkins, Perforce, TFS, Deploy DB, JIRA.

**Client:Fox Solutions Pvt.LtdDuration: July 2012 - Feb 2014**

**Role:Jr. Build& Release engineer**

**Description:**Deploying the website across many data centers and maintaining the code and employing shell scripting and

Perl to retrieve data and maintain the database.

**Responsibilities:**

* Designed and implemented **Puppet**-based configuration management system for all new **Linux** machines (physical

and virtual).

* Implemented maintained the branching and build/release strategies utilizing **Subversion**.
* Responsible to perform Build operations with **ANT** Scripts, while implementing the required modifications in the

**ANTScript** to perform build operations, in accordance with requirements.

* Used **Ant** scripts to perform the builds and wrote a few **buil. xmls** for application development teams.
* Implementing a **ContinuousDelivery** framework using **Jenkins**, **Chef**, **Maven** and **Nexus** in **Linux** environment.
* Performed all necessary day-to-day **Subversion** support for different projects.
* Set up **ContinuousIntegrations**& Manual Build Trigger for Build Definitions that does a Deployment into the

Servers.

* Created New Build Types in **TFS** and Assigned user security levels based on their designation in **TFS**.
* Created **continuousintegration** system using **SVN**, **ANT**, **Jenkins**, **Chef**, **Puppet** full automation
* Worked with **GITHUB** to store the code and integrated it to **AnsibleTower** to deploy the playbooks.
* Integrated **GIT** into **Jenkins** to automate the code check-out process.
* Responsible for design and maintenance of the **SubversionRepositories**, **views**, and the access control strategies.
* Created and updated **chefrecipes** and **cookbooks**, profiles and roles using **Ruby** and **JSON** scripting Migration of

all nodes from **Puppet** Configuration system to **Chef** System.

* Maintained source code repository in **subversion**, **GIT** and Automated deployment of builds to different

environments using **AnthillPro**.

* Deployed **Puppet**, **Chef** Dashboard for configuration management to existing infrastructure.
* Used **PowerShell** Scripting for automating process and collecting reports for daily tasks.
* Created **branches**, performed merges and labeled baselines in **ClearCase**.
* Used package repository with **Nexus**& Implemented **Maven** workflow to use public **JAR** files and packages to this

central repository.

* Wrote scripts in **Python**&**Shellscripting** to customize build, promotion and release process.

**Environment:**Puppet, Linux, Subversion, ANT, Jenkins, Chef, Maven, Nexus, Subversion, ANT, GITHUB, Ruby, JSON,

Clear Case, JAR, Python, Shell scripting.