SYLLABUS

DEVOPS INTRO

- WHAT IS SDLC
- WATERFALL MODEL
- ADVANTAGES & DISADVANTAGES
- AGILE METHODLOGY
- ADVANTAGES & DISADVANTAGES
- SCRUM
- DEVOPS HISTORY
- DEVOPS LIFECYCLE
- NEED OF DEVOPS
- ROLES & RESPONSIBILITIES
- BEFORE AND AFTER DEVOPS
- DEVOPS GOALS
- DEVOPS TOOLS
- ADVANTAGES
- ROADMAP

LINUX

- LINUX INTRODUCTION
- FLAVORS ON LINUX
- LINUX HISTORY
- LINUX ADVANTAGES
- SYSTEM INFORMATION
- HARDWARE INFORMATION
- FILE COMMANDS
- VIM EDITOR
- USERS & GROUPS
- FILE PERMISSIONS
- NETWORKING
- COMPRESSION/ARCHIVE
- DISK USAGE
- PACKAGE INSTALLATION
- FHS/DIRECTORY SYSTEM
- SED FILTERS
- GREP FILTERS
- SOME ADDITIONAL COMMANDS
- ASSIGNMENT

GIT

- WHAT IS VCS
- VCS HISTORY
- SCCM
- REVISION CONTROL SYSTEM
- SUBVERSION
- CONCURRENT VERSIONS SYSTEM
- WHY GIT
- GIT STAGES
- WORKING DIRECTORY
- STAGING AREA
- REPOSITORY (LOCAL, CENTRAL, REMOTE)
- GIT INSTALLATION
- GIT LIFECYCLE
- GIT LOGS
- GIT PUSH
- GIT PULL
- GIT CLONING
- GIT BRANCH
- GIT MERGE
- GIT STASH
- GIT CHERRY PICK
- GIT REVERT
- GIT MERGE
- MERGE CONFLICTS
- COMMIT A FILE USING GIT
- CONFIGURATION OF USER
- IGNORING CONTENT
- BRANCHING STRATEGIES
- GIT FLOW
- GIT-HUB FLOW
- GIT-LAB FLOW
- TRUNK BASED DEVELOPMENT
- GIT BRANCH (CREATE, DELETE, RENAME, SWITCH)

GITHUB:

- GIT REPOS (PRIVATE & PUBLIC)
- INTEGRATING REPOS
- FORKING

- GITHUB WIKIS
- LINKING PROJECTS
- GITHUB FILEADD
- TOKENS
- COMPARE & PULL REQUEST
- RENAMING REPOS
- DANGER ZONE OPTIONS
- MAKING PUBLIC REPO AS PRIVATE
- OWNERSHIP TRANSFER
- ARCHIVING REPOS
- DELETING REPOS
- ACCESSING THE PRIVATE REPOS
- ADVANTAGES & DISADVANTAGES
- DIFFERENCE BETWEEN GIT AND OTHER TOOLS
- INTERVIEW QUESTIONS

GIT ASSIGNMENT: IN THIS ASSIGNMNET USER-1 IS GOING TO CREATE BRANCHES AND COMMITCODE LATER PUSH TO GITHUB USER-2 IS GOING TO FORK THE REPO AND ADD SOME MORE CODE ON IT THEN LATER THE BOTH OCDES ARE GOING TO MERGE TO THE MASTER BRANCH AND FINALLY WE WILL GET THE CODE TO REVIEW.

MAVEN

- INTRODUCTION
- MAVEN OBJECTIVES
- PROJECT INFORMATION
- FEATURES
- DOWNLOAD REQUIRMENTS
- PLUGINS
- MAVEN GOALS
- WHAT MAVEN DOES
- WHAT IS BUILD TOOL?
- POM.XML
- REQUIREMENTS FOR BUILD
- MAVEN ARCHITECTURE
- MAVEN BUILD LIFE CYCLE
- MAVEN DIRECTORY STRUCTURE
- GENERATING JAR FILE
- JAVA PROJECT STRUCTURE
- PROBLEMS WITHOUT MAVEN
- MAVEN VS ANT

- INTERVIEW QUESTIONS

ASSIGNMENT: IN THIS ASSIGNMENT WE ARE GOING TO GENERATE ARTIFACTS FORM THE BASE CODE BY USING THE MAVEN LIFECYCLE AND UPLOAD TO SRTIFACT STORAGE.

JENKINS

- INTRODUCTION
- WORKFLOW
- CONTINUOUS INTEGRATION
- CONTINUOUS DELIVERY
- CONTINUOUS DEPLOYMENT
- JENKINS SETUP
- JENKINS SETUP REQUIRMENTS
- JAVA INSTALLATION
- GIT INTEGRATION
- MAVEN INTEGRATION
- JOBS TYPES IN JENKINS
- FREE STYLE JOB
- PIPELINE
- MULTIBRANCH
- MAVEN JOB, TASK
- PARAMETER BUILDING
- CHOICE PARAMETER
- FILE PARAMETER
- STRING PARAMETER
- MULTI-STRING PARAMETER
- CREDENTIALS PARAMETER
- BRANCH BUILDING
- SCHEDULE PROJECT
- CRON JOBS
- POLL SCM
- WEBHOOKS
- LINKED PROJECTS
- UP STREAM
- DOWN STREAM
- BUILD PIPELINE VIEW
- MASTER – SLAVE ARCHITECTURE
- MASTER – SLAVE SETUP
- ENVIRONMENT VARIABLES
- LOCAL VARIABLE

- GLOBAL VARIABLE
- TIMEOUT
- TIME STAMPS
- CUSTOM WORKSPACE
- WORKSPACE ACCESS
- DISABLING JOBS
- RENAMING JOBS
- BUILD PIPELINE VIEW
- SHELL EXECUTE
- BUILD TRIGGERS
- AUTHENTICATION FOR BUILD TRIGGER
- BUILD THROUGH TERMINAL
- THROTTLE BUILD
- PIPELINES
- SCRIPTED PIPELINES
- DECLARATIVE PIPELINES
- PIPELINE STAGES
- PIPELINE SYNTAX
- HELLO WORLD PIPELINE
- STAGED PIPELINE
- MULTI-STAGED PIPELINE
- PIPELINE AS A CODE
- MULTIPLE COMMANDS ON PIPELINE
- PAAC VARIABLES
- PAAC PARAMETERS
- PAAC INPUT PARAMETER
- PAAC POSTBUILD
- PAAC INPUT
- PIPELINE LOGS
- PIPELINE THROUGH BUILD PIPELINE
- PIPELIEN POST-BUILD
- PLUGIN MANAGEMENT
- CONSOLE OUTPUT
- PERMALINKS
- LIST VIEW
- CUSTOM VIEW
- BUILD HISTORY
- CRON SYNTAX GENERATOR
- EDITING BUILD INFO
- AGENT SETUP

- USER MANAGEMENT
- ADDING USERS
- ASSIGNING ROLES
- MANAGING ROLES
- BLUE OCEAN INTRODUCTION
- ADVANTAGES
- JENKINS ALTERNATIVES
- JENKINS METRICS & TRENDS
- JENKINS VS BAMBOO
- INTERVIEW QUESTION

ASSIGNMENT: WE ARE GOING TO SETUP MULTIPLE SLAVES AND AUTOMATE THE TASKS BY USING PIPELINES & JOBS WHICH INCLUDE VARIABLES, SCHEDULING AND OTHER TOPICS.

ANSIBLE

- ANSIBLE INTRODUCTION
- ANSIBLE USECASES
- ANSIBLE HISTORY
- ANSIBLE ARCHITECTURE
- ANSIBLE INVENTORY HOST PATTERN
- ANSIBLE EXECUTION TYPES
- AD-HOC COMMANDS
- WORKING WITH ADHOC
- ANSIBLE MODULES INTRODUCTION
- WORKING WITH ANSIBLE MODULES
- ANSIBLE PLAYBOOKS
- YAML INTRODUCTION
- YAML INDENTATION & SEPERATION
- COMMENTING SECTIONS
- ANSIBLE VARIABLES
- ANSIBLE HANDLERS
- ANSIBLE LOOPS
- ANSIBLE CONDITIONS
- ANSIBLE VAULT
- ANSIBLE ROLES
- ANSIBLE TAGS
- ANSIBLE DRY RUN
- ANSIBLE PIP
- ANSIBLE GALAXY

- ANSIBLE USERINFO
- ANSIBLE SETUP MODULES
- ANSIBLE DEBUG MODULE
- ANSIBLE RAW MODULE
- ANSIBLE COMMAND MODULE
- ANSIBLE COPY MODULE
- ANSIBLE SHELL
- ANSIBLE STRATEGIES
- ANSIBLE JINJA2
- ANSIBLE LOOKUP
- ERROR HANDLING
- CREATING LOCAL RESOURCES
- ANSIBLE GIT INTEGRATION
- DEPLOYING EC2 INSTANCE
- DOCKER WITH ANSIBLE
- JENKINS WITH ANSIBLE
- STATIC WEBSITE USING PLAYBOOK
- ADVANTAGES & DISADVANTAGES
- CREATING RESOURCES USING ANSIBLE
- ASYNCHRONOUS ACTIONS & POLLING
- ANSIBLE VS CHEF VS PUPPET VS SALT STACK
- INTERVIEW QUESTIONS

ASSIGNMENT: WE ARE HOSTING THE WEBSITE ON DIFFERENT HOSTS USING THE ANSIBLE PLAYBOOKS AND OTHER ROLES USING LOOPS, HANDLERS AND VAULT FOR DIFFERENT CONFIGURATIONS FOR DIFFERENT SERVERS.

DOCKER

- MICROSERVICES INTRODUCTION
- DIFFERENCE BETWEEN MONLITHIC & MICROSERVICES
- MICRO SERVICES ADVANTAGES
- MICRO SERVICES LIMTATIONS
- VIRTUALIZATION INTRO
- DOCKER VS VMS VS SERVERS
- BEFORE DOCKER VS AFTER DOCKER
- DOCKER INTRODUCTION
- UNDERSTANDING DOCKER
- DOCKER ARCHITECTURE
- DOCKER CLIENT
- DOCKER SERVER
- DOCKER DAEMON

- DOCKER OBJECTS
- DOCKER BASIC COMMANDS
- STOP VS KILL
- RUN VS CMD VS ENTRYPOINT
- ADD VS COPY
- RUN VS PULL
- DOCKER IMAGES
- DOCKER CONTAINERS
- DOCKER VOLUMES
- WORKING WITH CONTAINERS
- BUILD IMAGE FROM CONTAINER
- CREATING CUSTOM IMAGES
- WAYS OF CONTAINER CREATION
- DOCKER FILE
- DOCKER FILE COMPONENTS
- DOCKER FILE CREATION
- DOCKER VOLUMES
- USES OF VOLUMES
- CREATING VOLUMES
- CREATING VOLUMES BY USING COMMANDS
- VOLUMES (CONTAINER - CONTAINER)
- VOLUMES (HOST - CONTAINER)
- CREATING VOLUME FROM FILE
- CRREATING VOLUME FORM COMMANDS
- VOLUME MOUNTING
- VOLUME MODIFIYING
- DOCKER PORT MAPPING
- DOCKER MULTISTAGE BUILD
- CREATING JENKINS BY DOCKER
- DOCKER REGISTRY TYPES
- CLOUD BASED REGISRTY VS LOCAL REGISTY
- DOCKER LOGIN
- DOCKER HUB ACCOUNT CREATION
- IMAGE PUSHING TO REGISTRY
- DOCKER NETWORKING
- DOCKER SWARM
- DOCKER SWARM ARCHITECTURE
- DOCKER SWARM COMPONENTS
- DOCKER SWARM MANAGER
- DOCKER SWARM WORKER

- DOCKER SWARM SERVICE
- DOCKER PORTAINER
- DOCKER COMPOSE
- DOCKER COMPOSE ADVANTAGES
- DOCKER COMPOSE INSTALLATION
- DOCKER COMPOSE FILE
- DOCKER COMPOSE FILE IN JSON
- DOCKER COMPOSE FILE IN YAML
- DOCKER COMPOSE SINGLE SERVICE
- DOCKER COMPOSE MULTI SERVICES
- DOCKER COMPOSE FILE COMMANDS
- DOCKER COMPOSE SCALE
- DOCKER STACK
- PORTAINER INTRODUCTION
- PORTAINER USE CASES
- PORTAINER SETUP
- WORKING WITH PORTAINER
- MEMORY MANAGERMNET
- MEMORY SHARING
- ALLOCATIONG CPU
- LIMITING THE HARDWARE BLW CONTAINERS
- DOCKER NETWORKING
- BRIDGE NETWORK
- SOME ADVANCE COMMANDS
- INTERVIEW QUESTIONS

ASSIGNMENT: WE ARE DOCKERIZING A BASIC APPLICATION USING THE DOCKER FILE IN THE SWARM MODE AND PUSHING IT TO THE DOCKER REGISTRY.

KUBERNETES

- CONTAINER ORCHESTRAION
- KUBERNETES INTODUCTION
- ARCHITECTURE
- MASTER COMPONENTS
- NODE COMPONENTS
- WORKING WITH K8s
- ROLE OF MASTER
- COMPONENTS OF CONTROL PLANE
- KUBE-API SERVER
- ETDC

- FEATURES
- KUBE-SCHEDULER
- CONTROL MANAGER
- NODE COMPONENTS
- KUBELET
- CONTAINER ENGINE
- KUBE-PROXY
- POD
- POD LIMITATIONS
- POD MULTI CONTAINER POD
- KOPS
- KOPS ADATNAGES
- KOPS INSTALLATION
- KUBECTL INSTALLATION
- CLUSTER CREATION USING KOPS
- DEPLOYING A POD
- MANIFEST FILES
- NAME SPACES
- POD DEPLIYMENT BY MANIFEST
- LIMITATIONS
- SERVICE IN KUBERNETS
- TYPES OF SERVICES
- NODEPORT VS CLUSTERIP VS LB
- HIGHER LEVEL K8s OBJECTS
- IMPORTANT NOTATIONS
- SCALEUP PROBLEMS
- FEATURES
- DOCKER SWARM Vs K8s
- KUBEADM INTRODUCTION
- KUBESPRAY INTRODUCTION
- KOPS VS KUBEADM VS KUBESPRAY
- INTERVIEW QUESTIONS

NAGIOS

- HISTORY
- WHY NAGIOS
- FEATURES
- PHASES OF MONITORING
- NAGIOS ARCHITECTURE
- HOW IT WORKS

- PREREQUISITES
- DASHBOARD OVERVIEW
- INSTALLATION OF NAGIOS
- CLIENT SERVER SETUP
- INTERVIEW QUESTIONS
- CLIENT SERVER SERTUP
- MONITOR CLIENT SERVER

ASSIGNMENT: HERE WE ARE GOING TO MONITOR THE DIFFERENT SERVICES ON NAGIOS CLIENT SERVER FOR DIFFERENT ENVIRONMENTS.

TERRAFORM

- INTRODUCTION
- HISTORY
- USES
- TERRAFORM INSTALLATION
- TERRAFROM INIT
- TERRAFORM PLAN
- TERRAFORM APPLY
- TERRAFORM DESTROY
- TERRAFROM VALIDATE
- TERRAFROM UPDATE
- CREATING A MAIN.TF
- ROLE BASED AUTHENTICATION
- S3 BACKEND SETUP
- TERRAFORM VARIABLES
- STRING
- NUMBER
- BOOLEAN
- LIST
- TERRAFORM LOOPS
- TERRAFORM WORKSPACES
- TERRAFORM LOCALS
- TERRAFORM OUTPUTS
- CREATING VPC
- CREATING EC2
- CREATING S3 BUCKET
- CREATING SECURITY GROUPS
- CREATING SUBNETS
- CREATING EBS

- CREATING EFS
- CREATING RDS
- CREATING LOCAL RESOURCES
- TERRAFORM MULTIPLE TFVAR FILES
- TERRAFORM CLI
- VERSION CONSTRAINTS
- LOCAL VERSION CHANGING
- AWS VERSION UPDATING
- DYNAMIC BLOCK
- TERRAFORM STATE FILES
- TERRAFORM IMPORT
- ALIAS & PROVIDERS
- TERRAFORM TAINT
- TERRAFORM IMPORT
- TERRAFORM REFRESH
- TERRAFORM LIFECYCLE
- TERRAFORM REPLACE
- TERRAFORM TARGET
- COMMENTS
- CREATE BEFORE DESTROY
- PREVENT DESTROY
- IGNORE CHANGES
- TERRAFORM UPDATE
- TROUBLESHOOTING
- TERRAFORM MODULES
- ADVANTAGES & DIS ADVANTAGES
- ALTERNATIVES

ASSIGNMENT: HERE WE ARE GOING TO CREATE MULTIPLE RESOURCES BY USING THE WORKSPACE AND MODULES FOR REAL TIME USEAGE.

CLOUD INTRO

- WHAT IS CLOUD COMPUTING
- WITH AND WITHOUT CLOUD
- TYPES OF CLOUD COMPUTING
- SERVICE MODELS
- INFRASTRUCTURE AS A SERVICE
- PLATFORM AS A SERVICE
- SOFTWARE AS A SERVICE
- DEPLOYMENT MODEL

- PUBLIC CLOUD
- PRIVATE CLOUD
- COMMUNITY CLOUD
- HYBRID CLOUD
- WHAT IS AMAZON WEB SERVICES
- ADVANTAGES
- TOP 10 CLOUD PROVIDERS
- MAIN ROLES
- CERTIFICATIONS

AWS SERVICES

ELASTIC COMPUTE CLOUD (EC2)

- INTRO
- SERVER VS VM
- TYPES OF SERVERS
- CREATING EC2
- CONFIGURE INSTANCE
- PROVIDING SECURITY FOR EC2
- LIMITATIONS

SIMPLE STORAGE SERVICE (S3)

- INTRO TO STORAGE
- BUCKETS
- OBJECTS
- OBJECT LIMTATIONS
- OBJECT UPLOADING TYPES
- RULES TO CREATE BUCKET
- VERSIONING
- CROSS REGION REPLICATION (CRR)
- USE CASES OF CRR
- CREATING RULES FOR REPLICATION

VIRTUAL PRIVATE CLOUD (VPC)

- INTRO TO VPC
- WHY WE NEED VPC
- FEATURES

- TYPES OF VPC
- ARCHITECTURE
- VPC COMPONENTS
- PUBLIC VS PRIVATE SUBNETS
- ROUTING THE TRAFFIC
- CONNECTING THE RESOURCES
- CREATING VPC

ELASTIC BLOCK STORAGE (EBS)

- INTRO TO EBS
- TYPES OF BLOCK STORAGES
- VOLUME TYPES
- SSD VS HDD
- SNAPSHOTS
- INCREMENTAL VOLUMES
- EBS ENCRYPTION
- ENCRYPTION WAYS
- CONVERTING VOLUMES TO SNAPSHOTS
- MIGRATING TO REGIONS
- CREATING IMAGES
- ATTACHING TO MULTIPLE INSTANCES

IDENTITY AND ACCESS MANAGEMENT (IAM)

- IAM INTRO
- LIMITATIONS
- FEATURES
- IAM TERMS
- PRINCIPAL & REQUESTS
- AUTHENTICATION VS AUTHORIZATION
- IAM IDENTITIES
- USERS
- GROUPS
- ROLES
- POLICIES

COMMAND LINE INTERFACE (CLI)

- INTRO
- USE CASES

- ADVANTAGES
- INSTANCES
- BUCKETS
- IAM

RELATIONAL DATABASE SERVICE (RDS)

- INTRO
- ADVANTAGES
- USER SETTINGS
- DB ENGINE TYPES
- LICENSING OPTIONS
- RDS STORAGE
- TEMPLATES

ELASTIC LOAD BALANCER (ELB)

- INTRO
- USE CASES
- TYPE OF LOAD BALANCERS
- OSI MODEL
- ARCHITECTURE
- LISTNER GROUPS
- TARGET GROUPS
- CREATING & WORKING

OTHER SERVICES

- SIMPLE NOTIFICATION SERVICE (SNS)
- SIMPLE QUEUE SERVICE (SQS)
- CODE BUILD
- CODE COMMIT
- CODE DEPLOY
- AWS CODE PIPELINE
- AMAZON MACHINE IMAGE (AMI)
- ELASTIC BEANSTALK (EBS)

BASH SCRIPTING

- WHAT IS BASH
- BASH SCRIPT
- USER INPUT

- VARIABLES
- DATE FORMATTING
- IF STATEMENT
- ELSE IF STATEMENT
- CASE
- FOR LOOP
- WHILE LOOP
- UNTIL
- BASH STRING
- SRING LENGTH
- CONCATENATION STRING
- READ FILE
- WRITE FILE
- BASH COMMENTS
- BASH QUOTES

CODE & ARTIFACT STORAGE

JFROG

- INTRO
- USE CASES
- SETUP
- CREATING REPO
- CREATING USERS

SONARQUBE

- INTRO
- ADVATNAGES
- SETUP
- CONNECTING

OTHER TOPICS

LAMP SETUP

- WHAT IS LAMP
- WHY LAMP
- LAMP COMPONENTS
- SETUP

VIRTUALIZATION

- WHAT IS VIRTUALIZATION
- TYPES
- SETUP
- ADVANTAGES & DISADVANTAGES

SOFTWARE ARCHITECTURES

- TYPES OF ARCHITECTURES
- CLIENT LAYER
- APPLICATION LAYER
- DATABASE LAYER
- ONE TIER ARCHITECURE
- TWO TIER ARCHITECURE
- THREE TIER ARCHITECURE
- N-TEIR ARCHITECURE

MONOLITHIC TYPE

- WHAT IS MONOLITHIC
- MONOLITHIC ARCHITECTURE
- ADVATNAGES
- DISADVANTAGES

MICROSERVICES

- WHAT IS MICROSERVICES
- MICROSERVICES ARCHITECTURE
- ADVATNAGES
- DISADVANTAGES
- MONOLITHIC VS MICROSERVICES

WEB & APP SERVERS

- WHAT IS A WEBSERVER
- WHAT IS WEBAPP SERVER
- WEBSERVER VS WEBAPP SERVERS
- APACHE HTTPD
- APACHE TOMCAT
- NGINX
- IIS

DEPLOYMENT TYPES

- WHAT IS DEPLOYMENT
- TYPES OF DEPLOYMENTS
- NEED FOR DEPLOYMENTS
- PRE-CONSIDERATIONS
- ROLLING STRATEGIES
- CANARY DEPLOYMENT
- RECREATE STATEGY
- CUSTOM STATEGY
- BLUE-GREEN DEPLOYMENT
- A/B DEPLOYMENT

REAL TIME PROJECTS

AUTOMATING INFRA THROUGH JENKINS:
TOOLS: GIT, GITHUB, JENKINS, TERRAFORM

DESCRIPTION: THIS IS THE FIRST STAGE FOR ANY PROJECT ON REAL TIME, IN THIS PROJECT WE ARE GOING TO CREATE INFRASTRUCTURE FOR OUR PROJECT BY AUTOMATION USING THE JENKINS DECLARATIVE PIPELINE.

SOFTWARE INSTALLATIONS:

DESCRIPTION: AFTER CREATING INFRA WE ARE GOIN TO INSTALL SOFTWARES ON 3 TEIR ARCHITECTURE WE ARE INSTALLING WEBSERVERS, WEBAPP TOMCAT AND PROGRAMMING SOFTWRAES LIKE JAVA, PTHON NODE AND PHP AND FINALLY DATABASES LIKE ARANGO AND MYSQL.

AUTOMATION DEPLOYMENT:
TOOLS: GIT, GITHUB, JENKINS, MAVEN, S3 STORAGE & TOMCAT.

DESCRIPTION: ONCE INFRA AND SETUP IS DONE, WE ARE GOING TO DEPLOY THE WEBAPPLICATION ON A APPLICATION SERVER LIKE TOMCAT AND STORING THE ARTIFACTS ON AWS S3 AND WILL BE ABLE TO ROLL BACK ONCE IT FAILED.

DOCKERIZING APPLICATION:
TOOLS: GIT, GITHUB, JENKINS & DOCKER

DESCRIPTION: WRITE THE DOCKER FILE TO DEPLOY STATIC WEBSITE IN APACHE, PUSH IT INTO GITHUB AND BUILD THAT DOCKER FILE IN JENKINS BY INTEGRATED GIT WITH JENKINS.

CI/CD USING ANSIBLE:

TOOLS: GIT, GITHUB, ANSIBLE, TOMCAT.

DESCRIPTION: CREATE AWS CODE PIPELINE TO IMPLEMENT CONTINOIUS INTEGRATION AND CONTINIOUS DEPLOYMENT. WHENEVER DEVELOPER PUSH THE CODE INTO CENTRAL REPO, AUTOMATICALLY IT PIPELINE GETS RELEASED AND DEPLOYED.

CI/CD USING AWS CODE PIPELINE:

SERVICES: GIT, S3, IAM, AWS CODECOMMIT, CODE BUILD, CODE DEPLOY, CODE PIPELINE.

DESCRIPTION: CREATE AWS CODE PIPELINE TO IMPLEMENT CONTINOIUS INTEGRATION AND CONTINIOUS DEPLOYMENT. WHENEVER DEVELOPER PUSH THE CODE INTO CENTRAL REPO, AUTOMATICALLY IT PIPELINE GETS RELEASED AND DEPLOYED.