We have one application on Jenkins
One repository on Angular
One Repository for JAVA app
One repository for admin app
Angular App is a code
Java app has its own repository here
There are 2 JAVA app one is admin module based on JAVA
There is a REST API Application Service
There is a GIT hub Repository
There are 3 micro services within a big application

I use various DevOps tools such as GIT, Jenkins, Docker, Ansible and AWS

on AWS: I am working on AWS cloud EC2 instances, Mostly working on compute services,

I am creating EC2 instances

I am having Java applications

I am making .war file

It is a Maven Build

We are maintaining the source code into the GIT hub

And the moment the developer triggers the code in the master branch or push the code into the Master branch through the poll scm it triggers the job in Jenkins and the maven build happens which creates .war file

After that the code quality checks happen through the sonar Qube (plugin) and if there is 70 percent code quality is okay, then it triggers the next job where we write/define Docker file we define .war file in Docker file and we create Docker image then it goes to next step where this Docker image goes to JFrog Artifactory.

We monitor Ec2 instances by using cloud Watch

In Jenkins we set up user id and password so that the whole team can access

We Setup access for different servers

For database third party application is used

Feature Branch for CI CD pipeline
Integration Test, UAT Test
CI CD - New Feature developed and pushed
Unit test cases are executed and passed,

When they are passed they go to
System Testing or Functional Test Cases
Stages 2 gets executed
If stage 1 passes
Test Cases be executed
Maven Parameters
Jenkins we will schedule jobs

UI Test Automation:
Back End Test automation: Rest Assured API
Web Services UI: GIT Hub where tests are executed
Daily Execution: Jenkins
Failure Notifications can be configure in Jenkins

Spring Boot Application
Back End: API Services
3 Micro Applications

Between these three there is flight information are connected through API
Backend APIS are automated and integrated
Data load and data extractions all data is serviced from APIS
Admin Panel: Master Control: Data Routing
Feeds are updated
Flight data is uploaded by Admin
In order to run an application load balancing
Pre - QA Phase: While deployments load balancing concepts: Docker && Kubernetes
When particular application is deployed
Availability
2 servers: Main Servers, Backup Server
Load Balancing Kubernetes
Master Slave concept

In order to design Infrastructure architecture - AWS
There are 3 applications
Primary Instance
1000 requests per minute are saved
1 lacs request is there how you serve requests

Dev / QA different stages: integrated
Jenkins particular Jobs: Test Cases, Unit Test cases: Every 12 hours every 24 hours

We need to build an executable file out of this GIT that is called CI. The famous tool we use is Jenkins. We use Jenkins with Maven to build the code. We take the raw code from the developer we check whether the code is correct or not or is getting compiled or not, whether unit test cases are running or not. All these things we check we integrate the code. Once the code is ready we need to host it we need to deploy the code we need to setup an infrastructure. Take a machine

install Java Install Tomcat and then Dependencies copy the code and then deploy the code. We have CM tools to make changes to our infrastructure that is called CM. Once the environment is setup we use containers to host the applications like .war file or tomcat or application we use containers. We are packaging together and running it as a container; we are using Docker and container to manage the application

From the configuration management side I am using Ansible,

From Ansible I installing the packages on the target system copy the files running the shell command