Phase - 5 Practice Project

3. Deploy Application on Cloud.

• Algorithm

1. Prepare Your Spring Boot Application:

- Ensure your Spring Boot application is ready for deployment. It should be a standalone, executable JAR file that can run independently.

2. Build Your Spring Boot Application:

- Use your preferred build tool (e.g., Maven or Gradle) to build the executable JAR of your Spring Boot application.

3. Create an AWS Account:

- If you don't have an AWS account, sign up for one at https://aws.amazon.com/ and set up your account credentials.

4. Install and Configure AWS CLI:

- Install the AWS Command Line Interface (CLI) on your local machine and configure it with your AWS credentials using the aws configure command.

5. Set up an AWS EC2 instance:

- Log in to the AWS Management Console.
- Navigate to the EC2 Dashboard.
- Launch an EC2 instance, selecting an appropriate Amazon Machine Image (AMI) based on your needs (Amazon Linux, Ubuntu, etc.).
- Configure the instance with appropriate security groups, key pairs, and instance type.
- Connect to the EC2 instance

6. Use WinSCP to transfer your Spring Boot JAR file to the EC2 instance. Ensure that you have the necessary permissions and credentials to access the instance via SSH.

7. Connect to the EC2 instance via SSH:

- Use a terminal or SSH client to connect to the EC2 instance.

- Navigate to the directory where you placed the Spring Boot JAR file.
- Give the necessary permissions to the jar file.

8. Install Java on the EC2 instance:

- Check if Java is already installed: java -version
- If not installed, install Java (OpenJDK or Oracle JDK) on the EC2 instance.

9. Run the Spring Boot application:

- Start the Spring Boot application using the following command: java -jar AmazonEC2App-0.0.1-SNAPSHOT.jar

10. Verify that the application is running correctly. Using the Public IPv4 Address with port 8080.