**Project Name: CI/CD Deployment for Springboot Application**

**Developed by Palak Chaurasia**

**1.CI/CD Deployment for Springboot Application**

1.1 Project Description

In this project a spring boot application is hosted on an Amazon Web Service EC2 Ubuntu instance with continuous integration and continuous deployment using tools such as Jenkins and Docker.

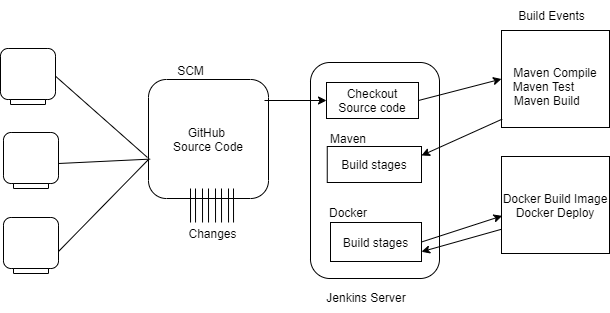
1.2 Technology Used

* Agile and Scrum
* Maven
* Postman
* JSON
* SQL
* Docker
* Jenkins
* GitHub
* Amazon Web Services
* Mobaxterm
* Eclipse
* Java
* Data Structure
* Spring Boot
* HTML

**2.Work Flow**

2.1 Flowchart-

FLOW OF DEPLYOMENT



2.2 Project User Stories

1. As a user I want continuous integration of springboot application.
2. As a user I want continuous deployment of springboot application.
3. As a developer I want to automate the integration of springboot application.
4. As a developer I want to automate the development of springboot application.

2.3 Sprint Planning

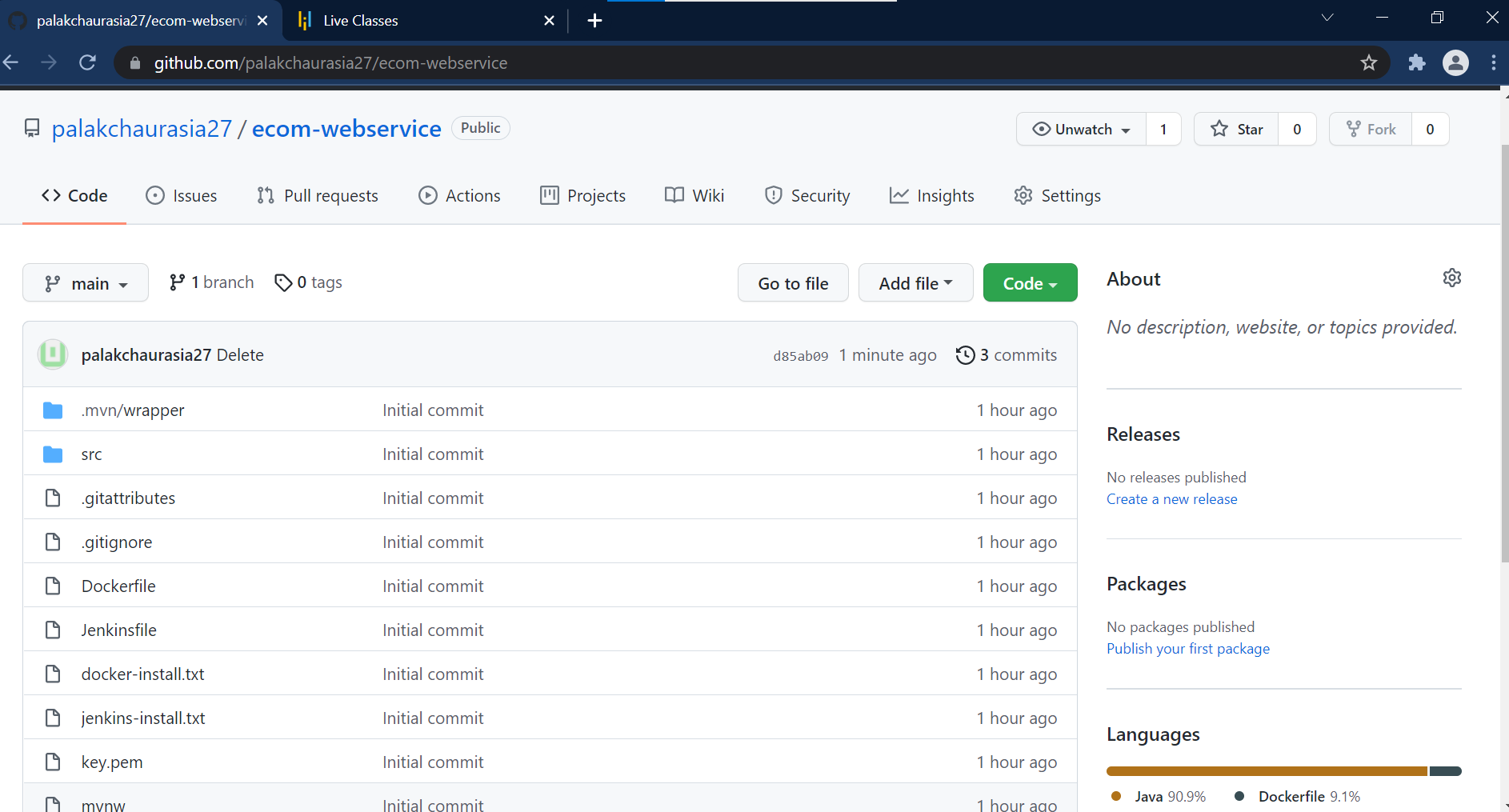
One sprint was required- 7 Days

Sprint 1- Duration:1 week (7 days)

1. As a user I want continuous integration of springboot application.
2. As a user I want continuous deployment of springboot application.
3. As a developer I want to automate the integration of springboot application.
4. As a developer I want to automate the development of springboot application.

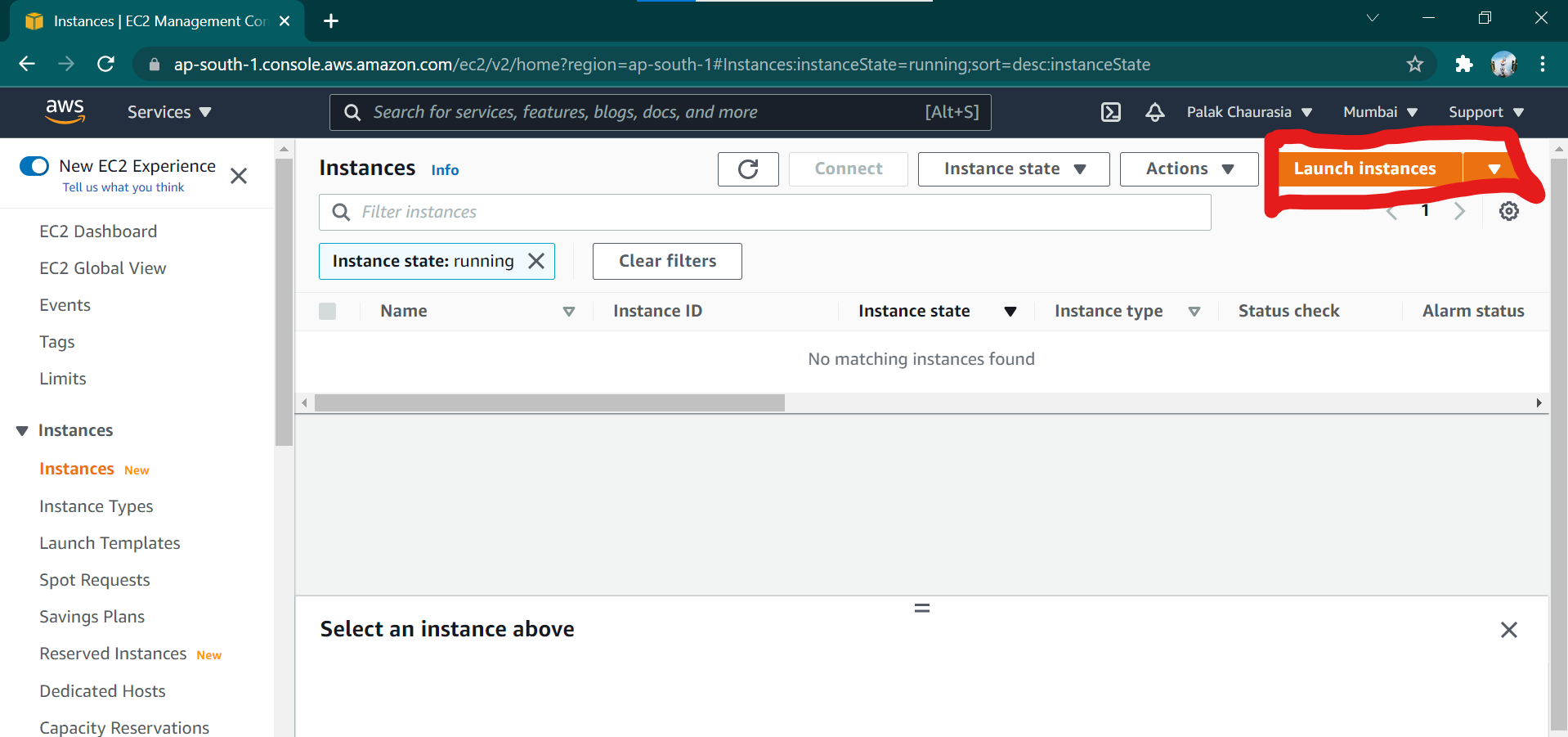
**3. Project Git Repository**

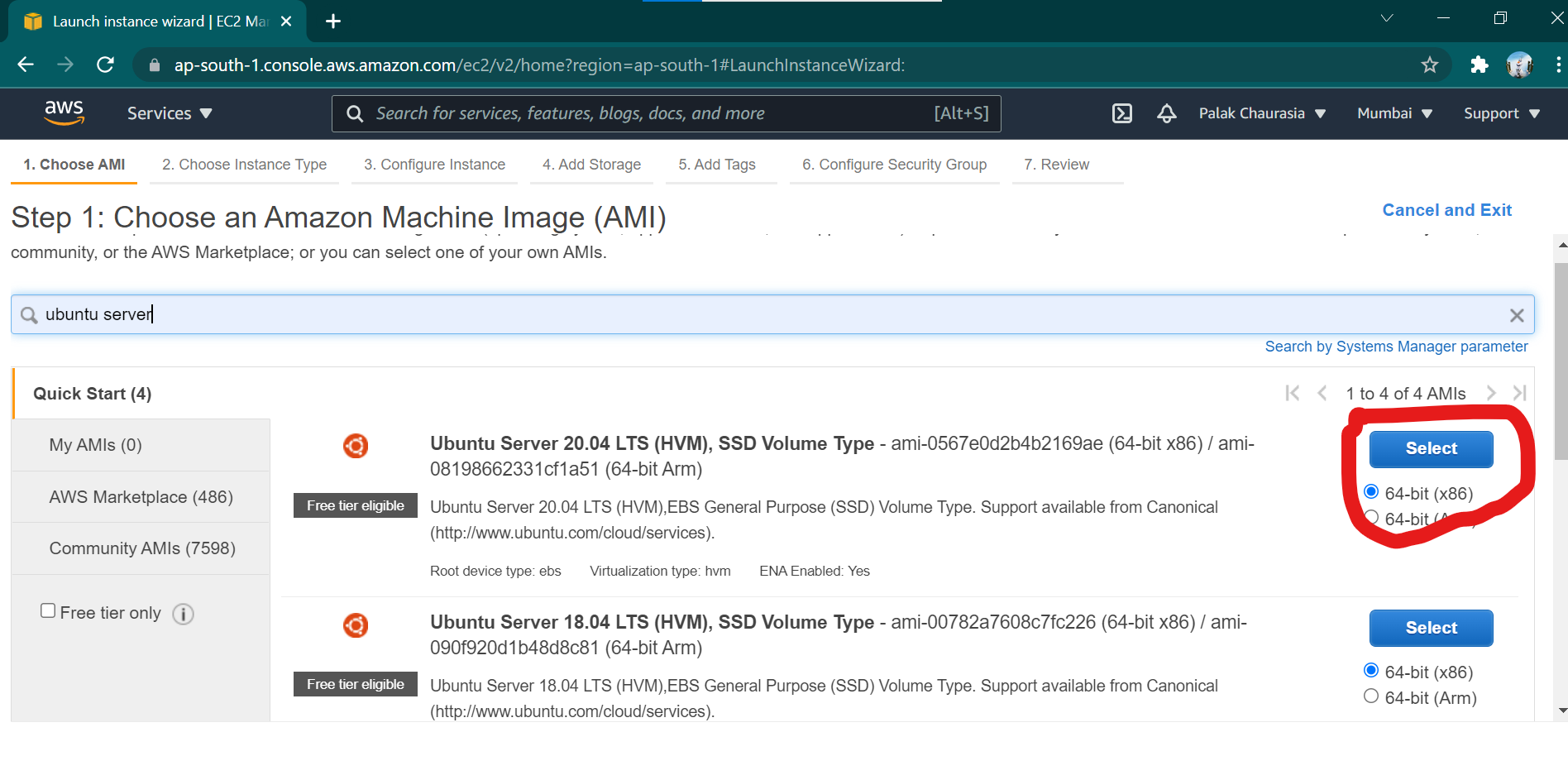
1. GitHub CLI: gh repo clone palakchaurasia27/ecom-webservice
2. GitHub Repository Link: https://github.com/palakchaurasia27/ecom-webservice.git
3. clone git : git clone https://github.com/palakchaurasia27/ecom-webservice.git
4. Screenshot:

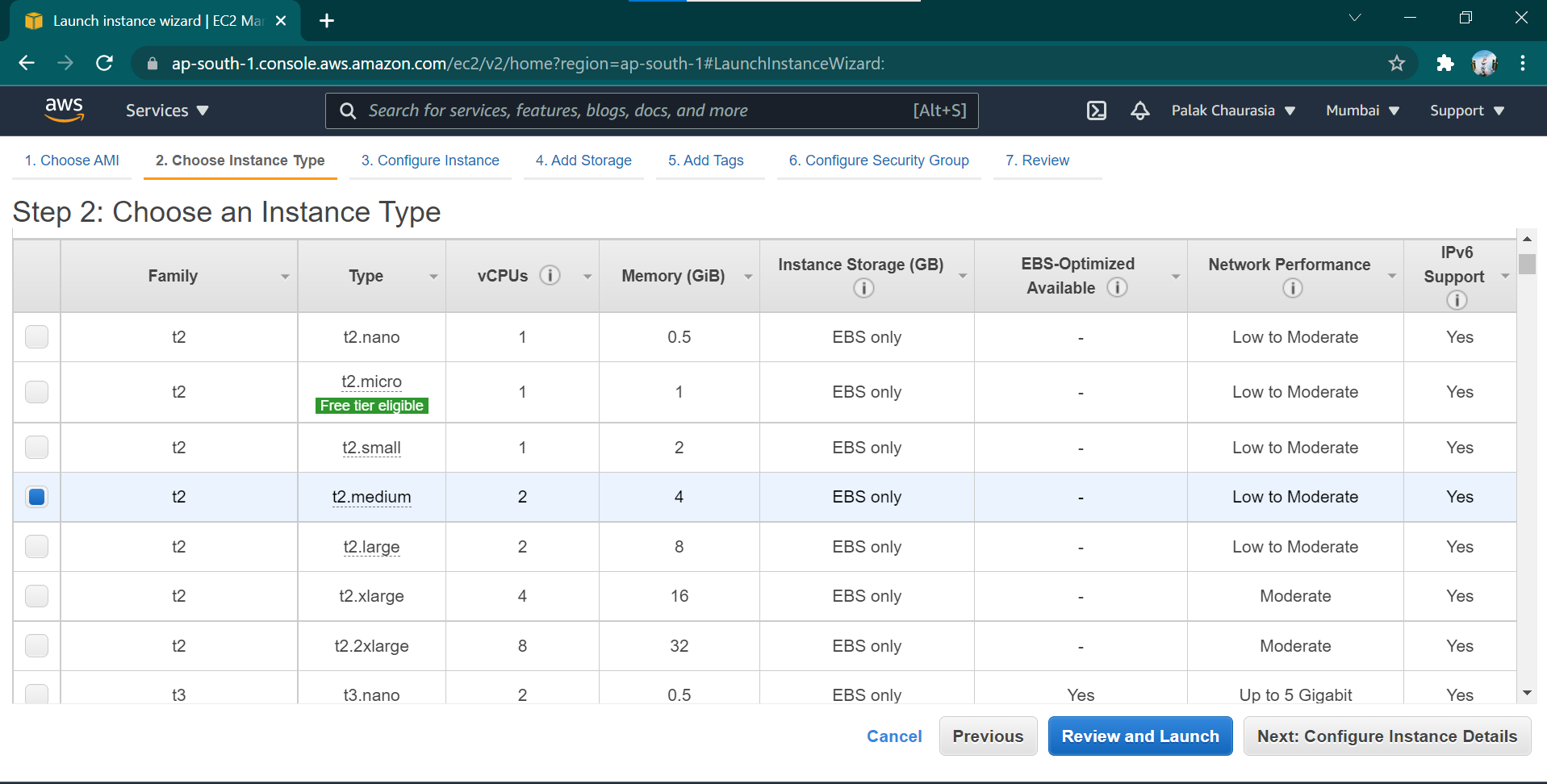


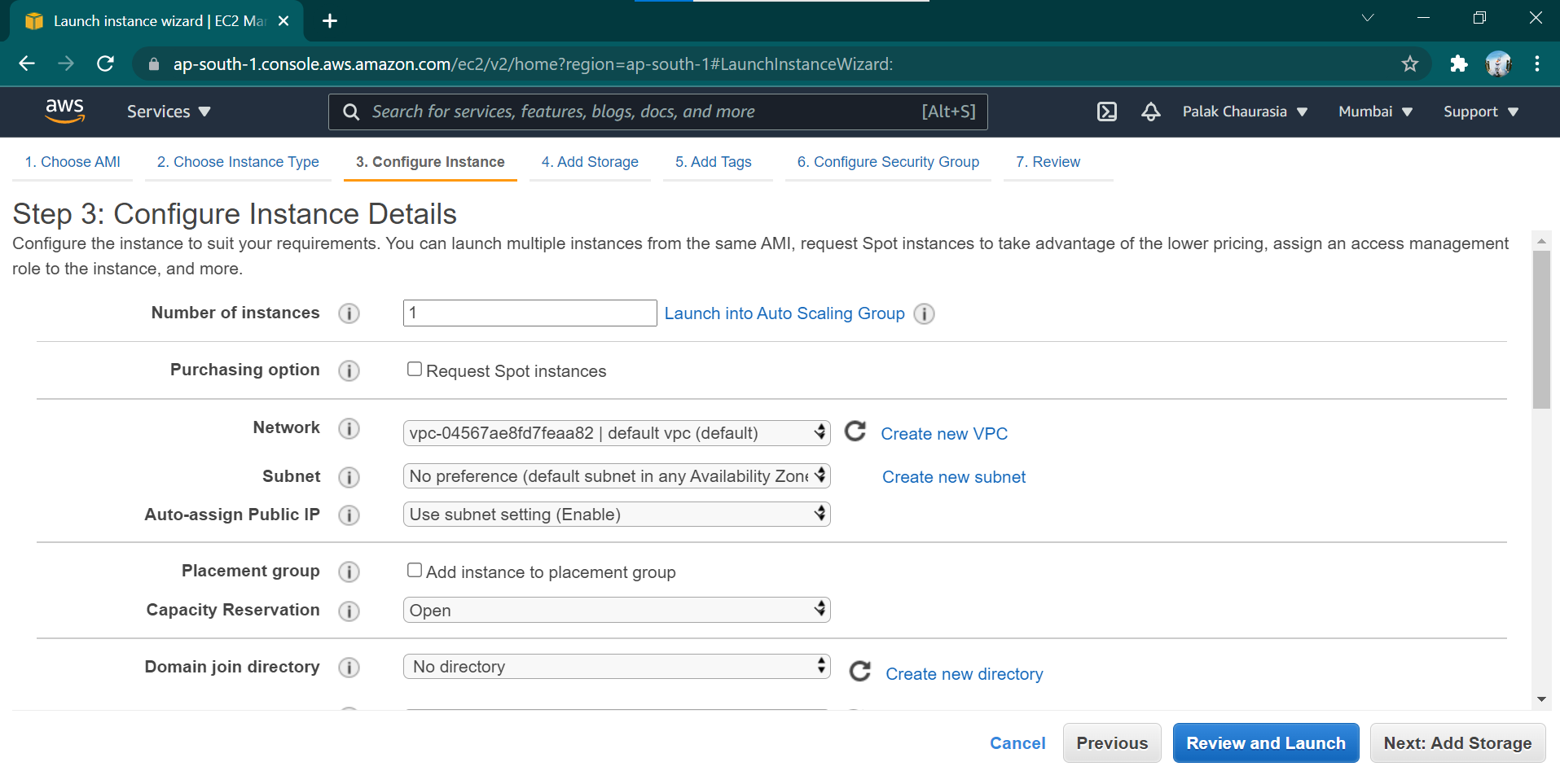
**4. Steps By Step Process**

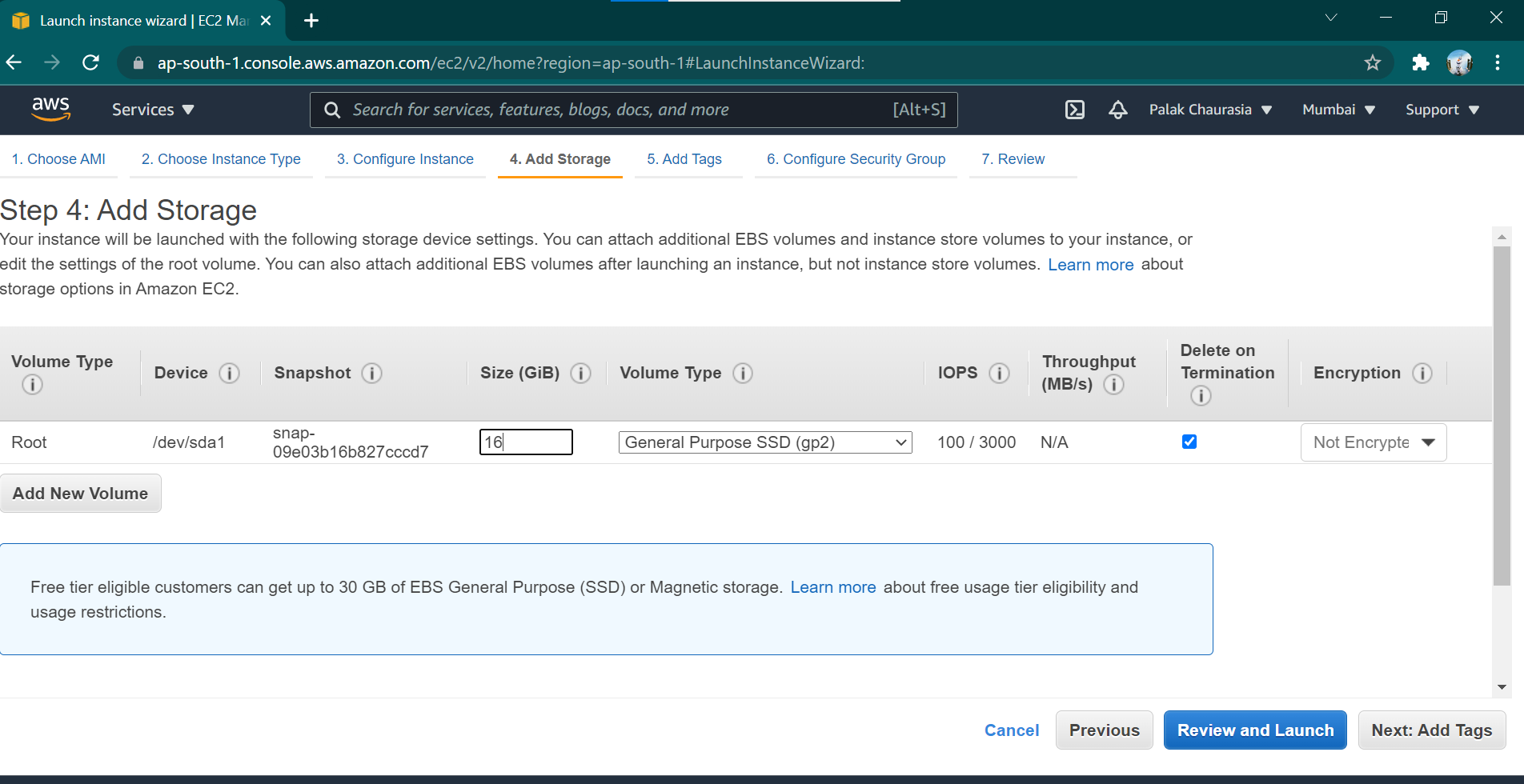
1. **Create an EC2 instance**

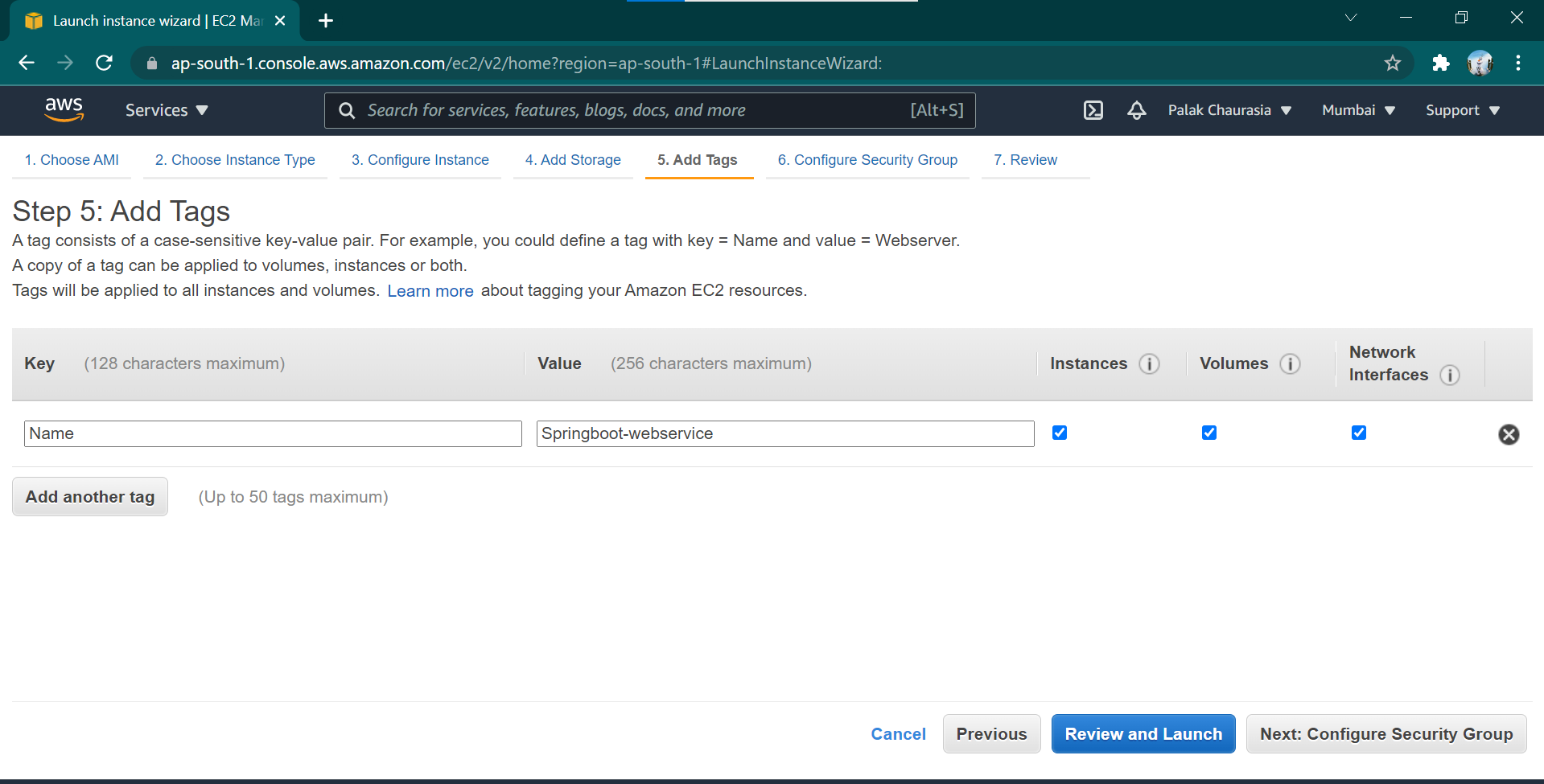
****

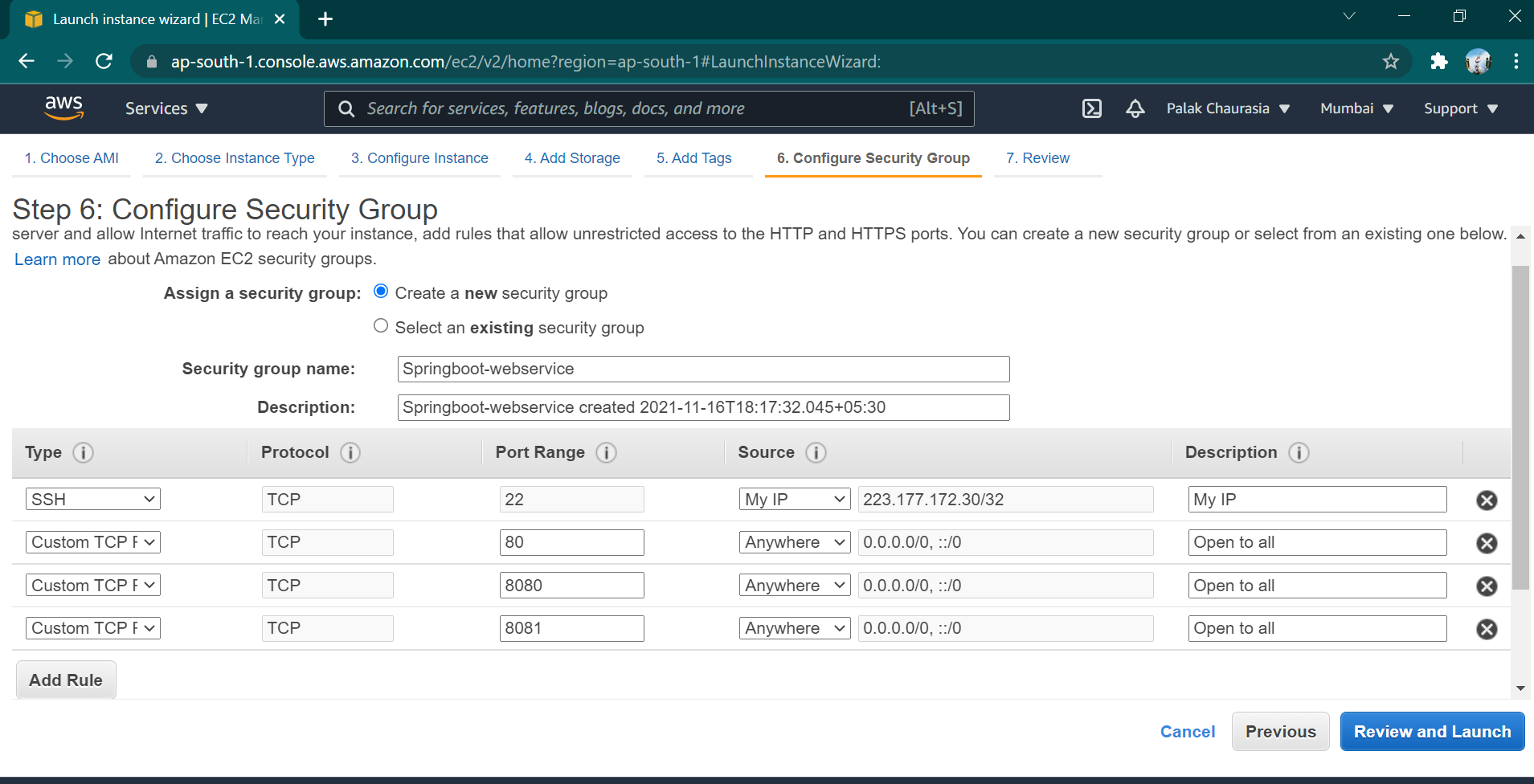
****

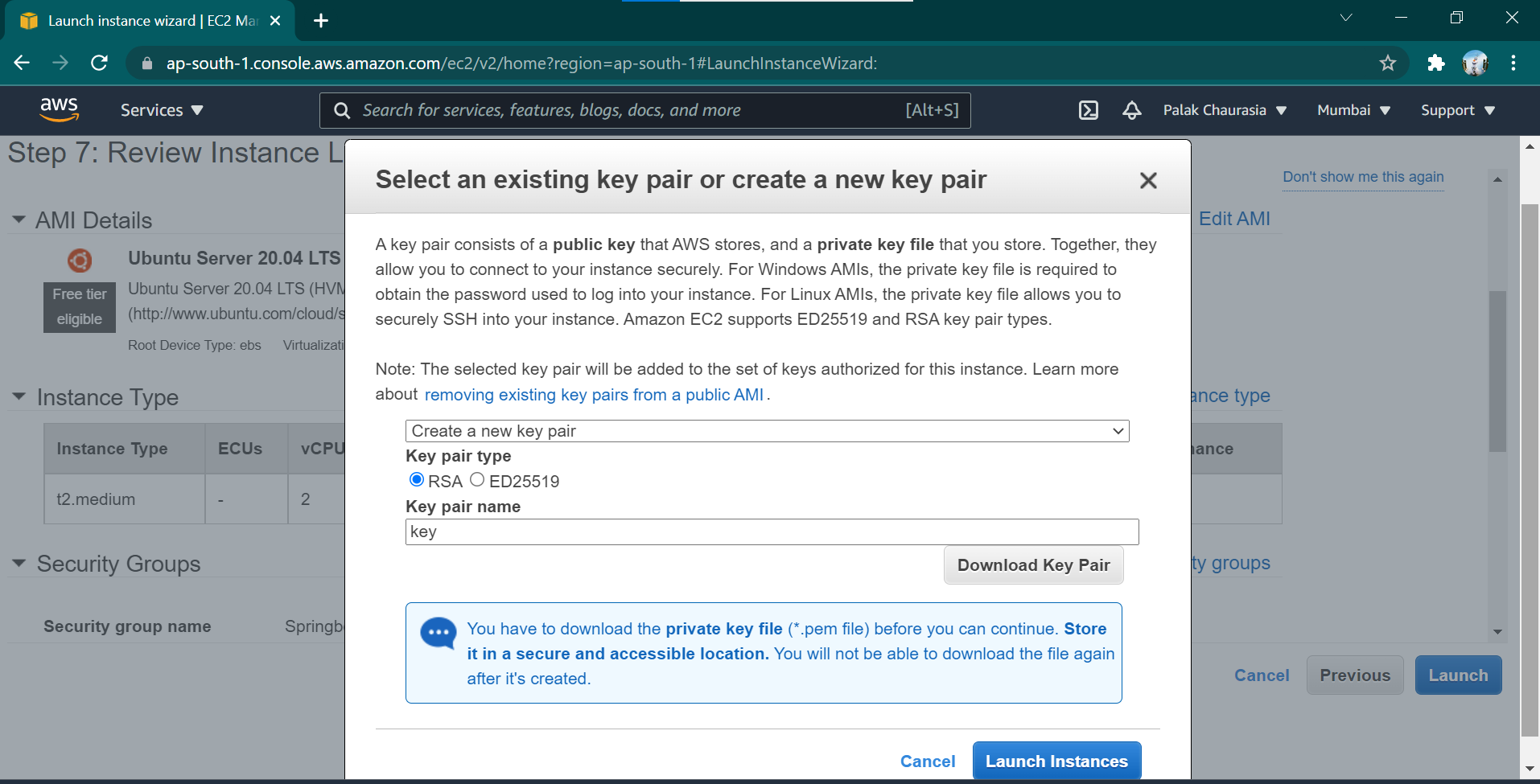
****

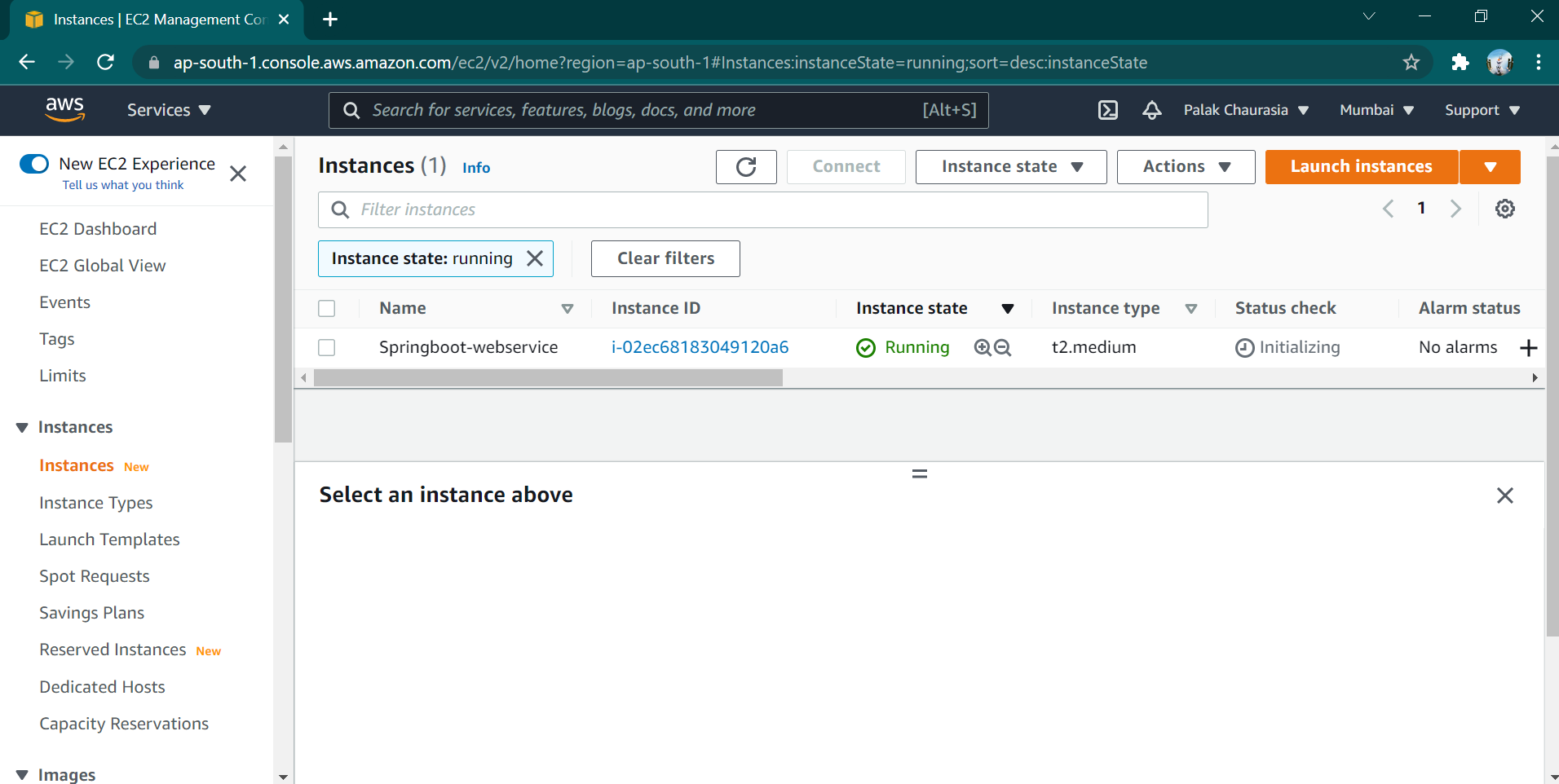
****

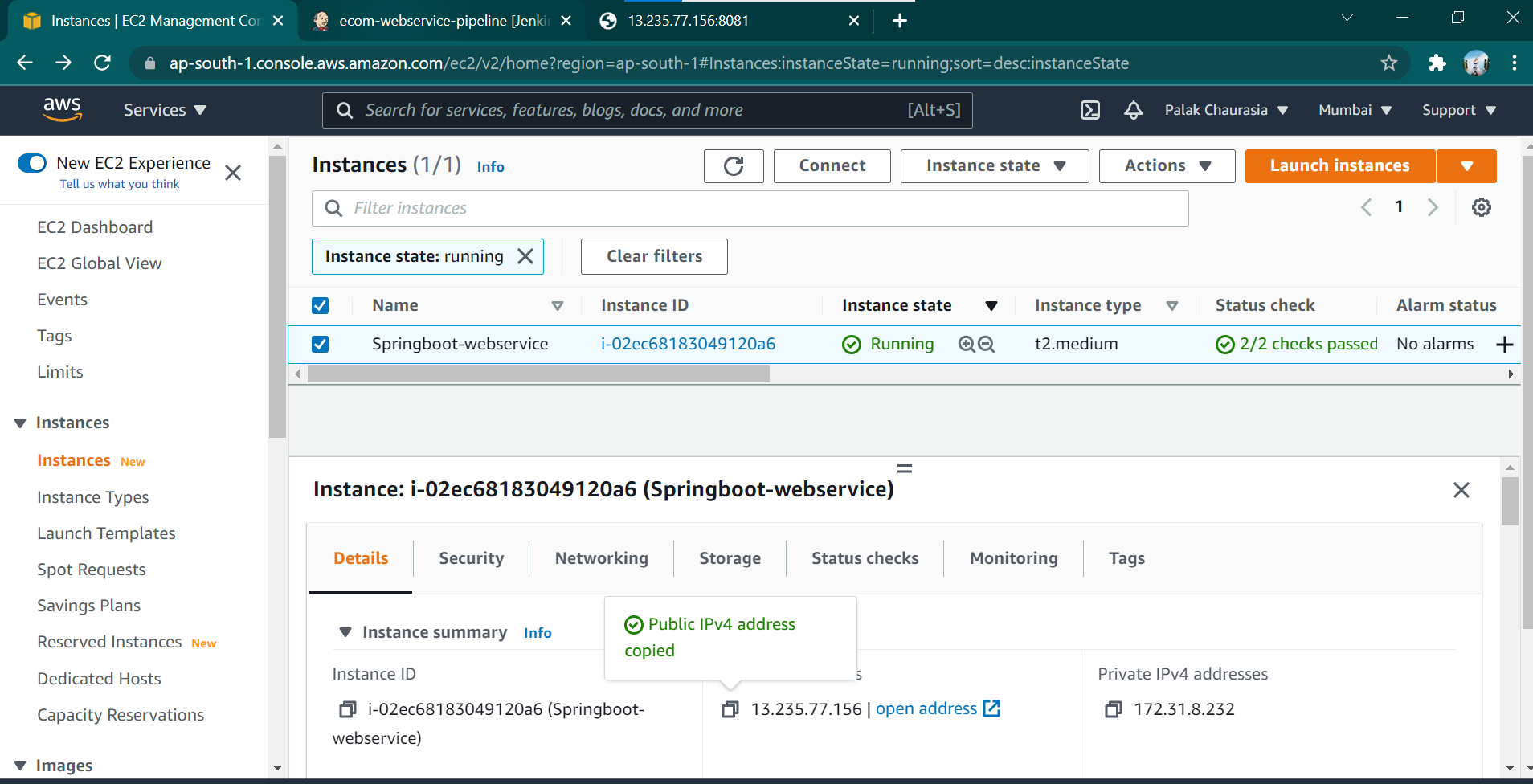
****

****

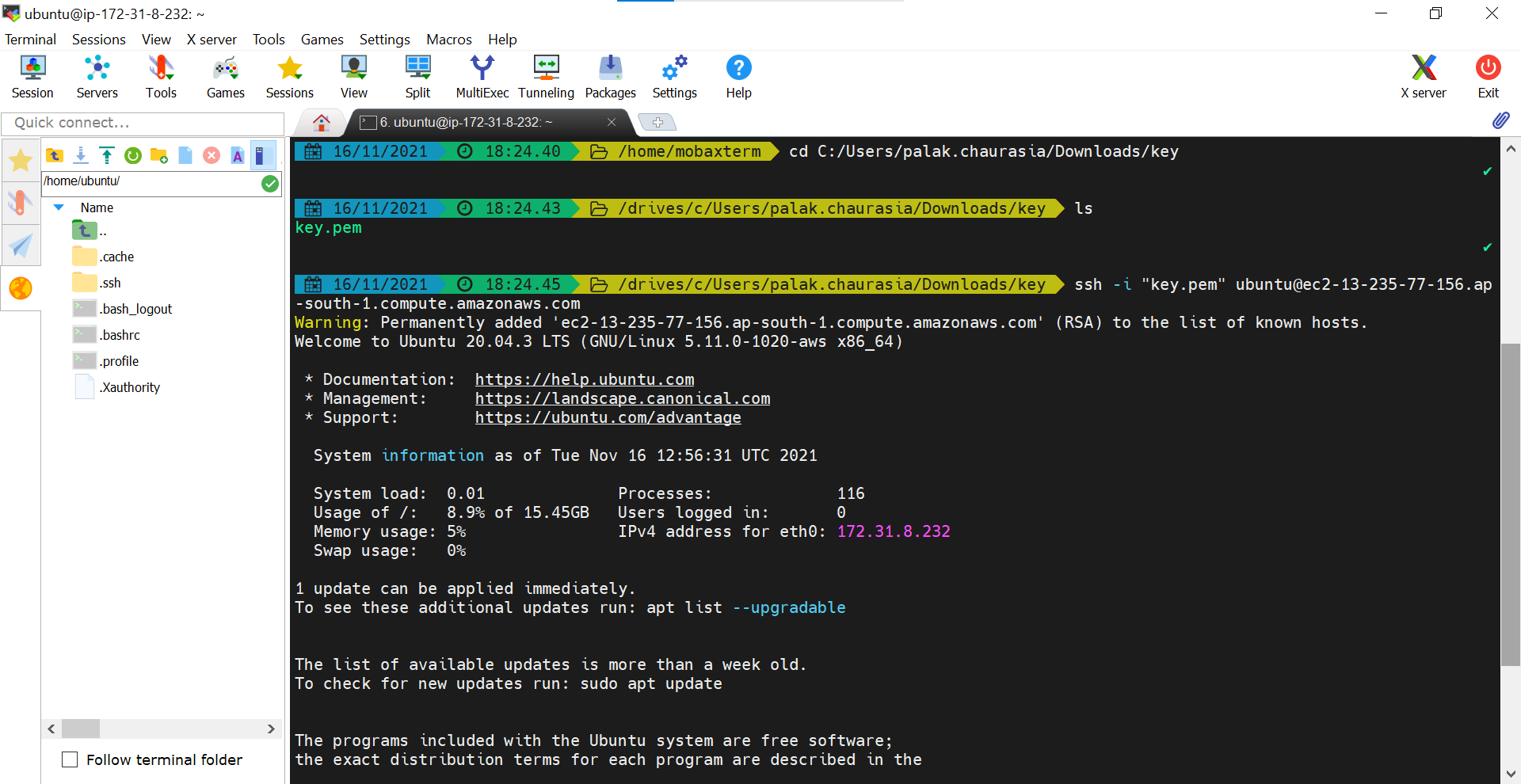
****

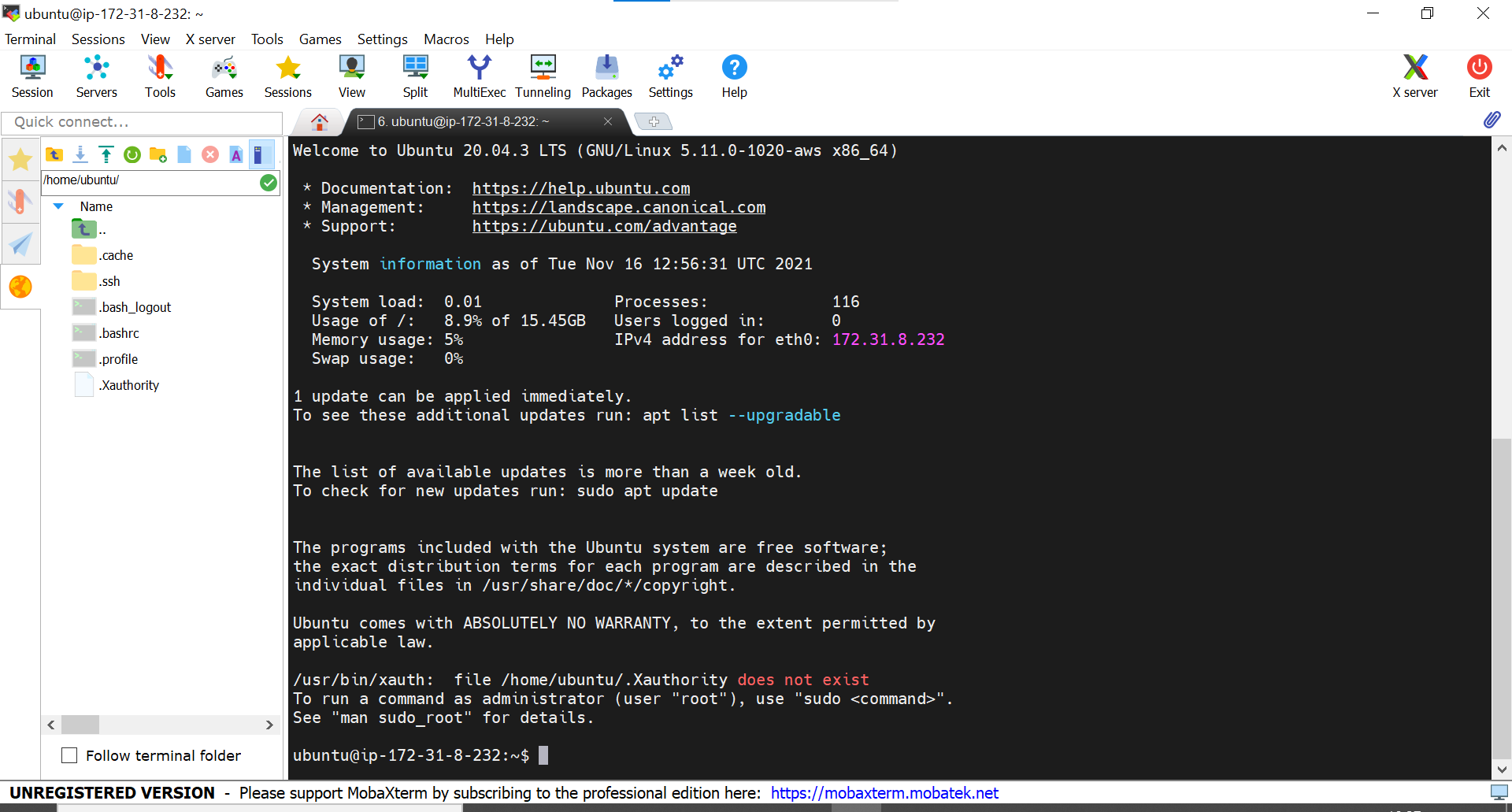
****

****

****

1. **Connect to EC2 Instance**

****

****

1. **Install Docker on Ubuntu Instance**

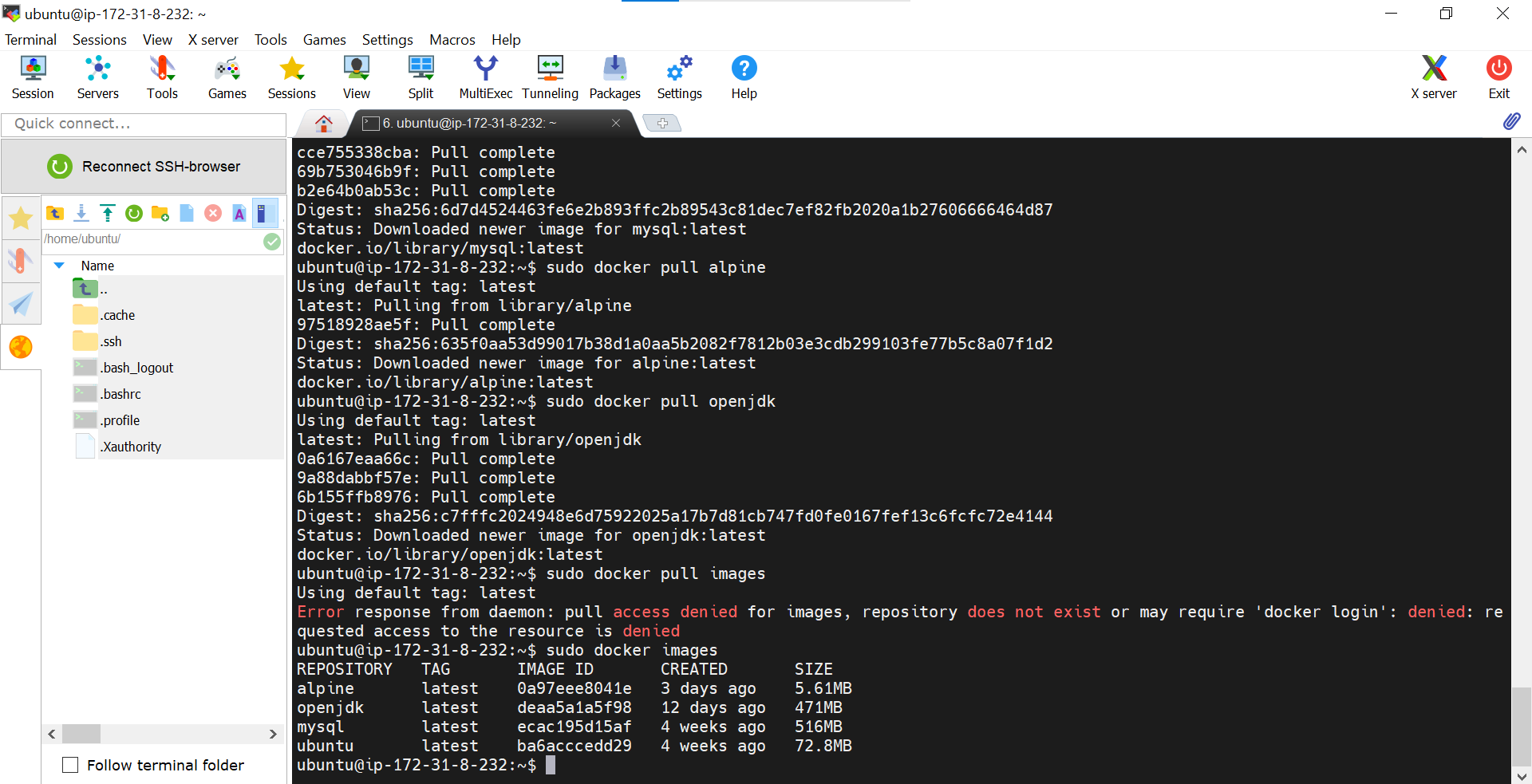
Using these commands

* sudo apt –get update
* sudo apt-get install ca-certificates curl gnupg lsb-release -y
* curl-fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
* echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu $(lsb\_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
* sudo apt-get update
* sudo apt-get install docker-ce docker-ce-cli containerd.io –y
* sudo docker –v

1. **Docker pull images**

Using these commands

* sudo docker pull Ubuntu
* sudo docker pull mysql
* sudo docker pull alpine
* sudo docker pull openjdk
* sudo docker images



1. **Install JDK, Maven, Sql on Instance**

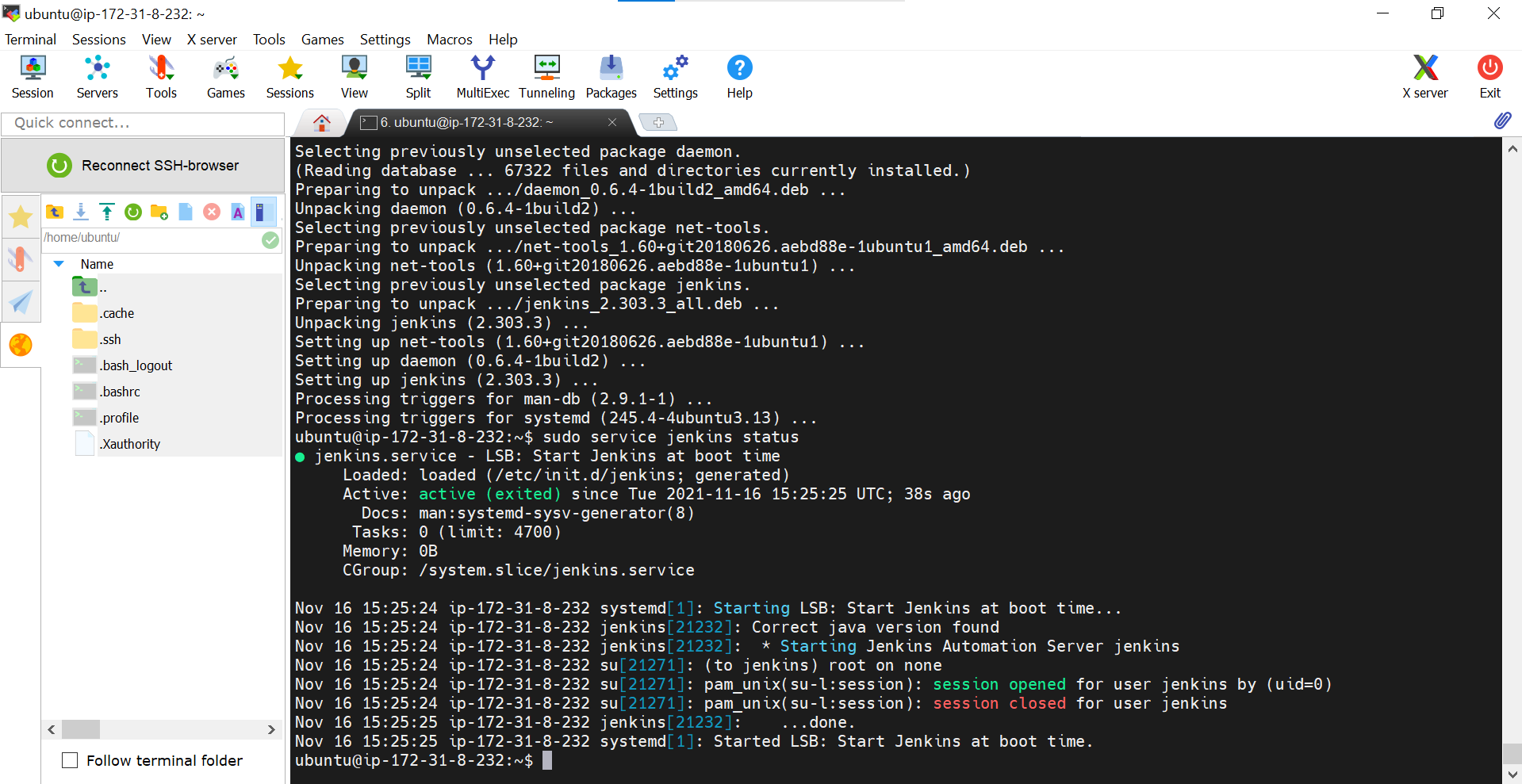
Using these commands

* sudo apt install default-jdk –y
* sudo apt install maven –y
* sudo apt install mysql-server -y

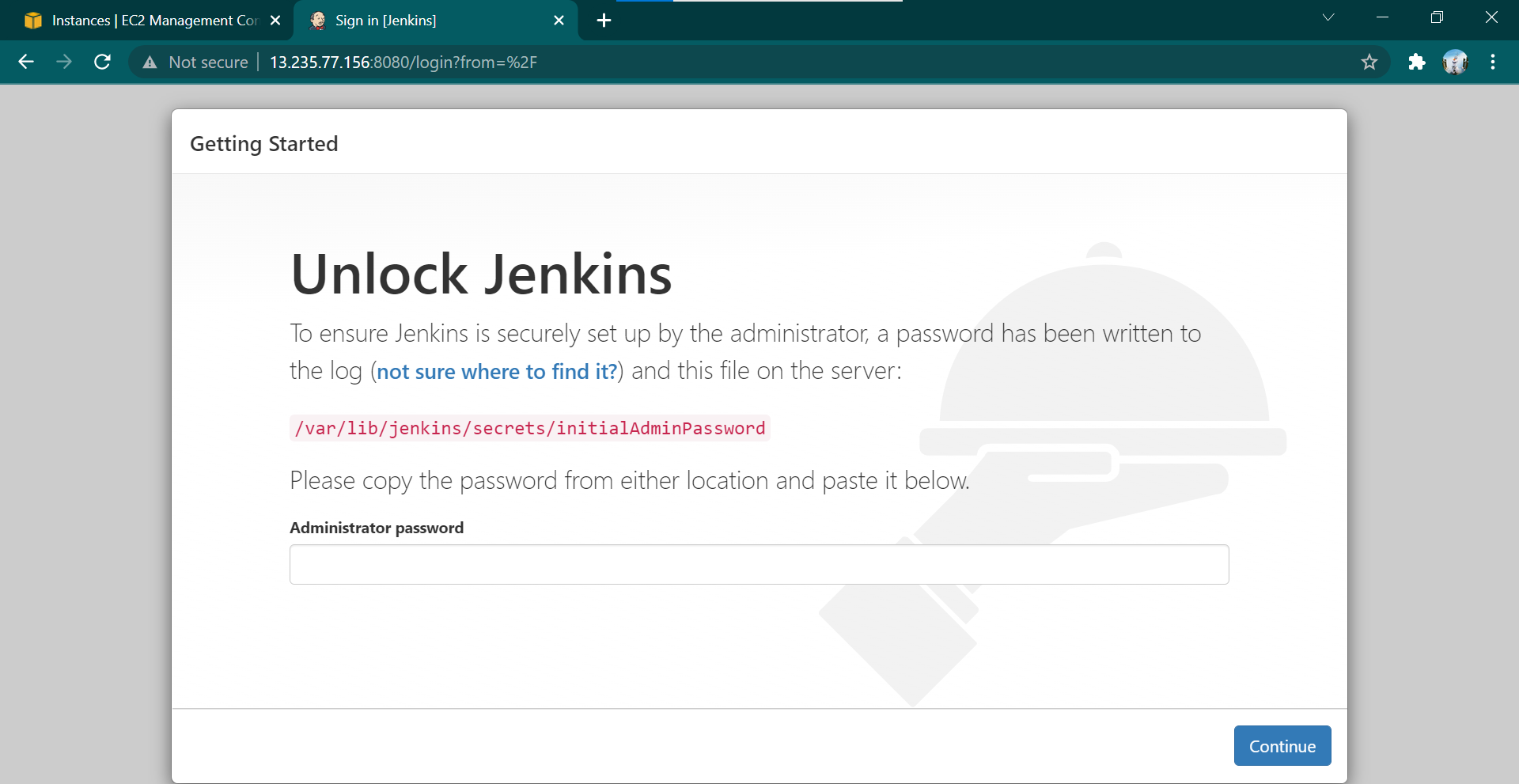
1. **Install Jenkins**

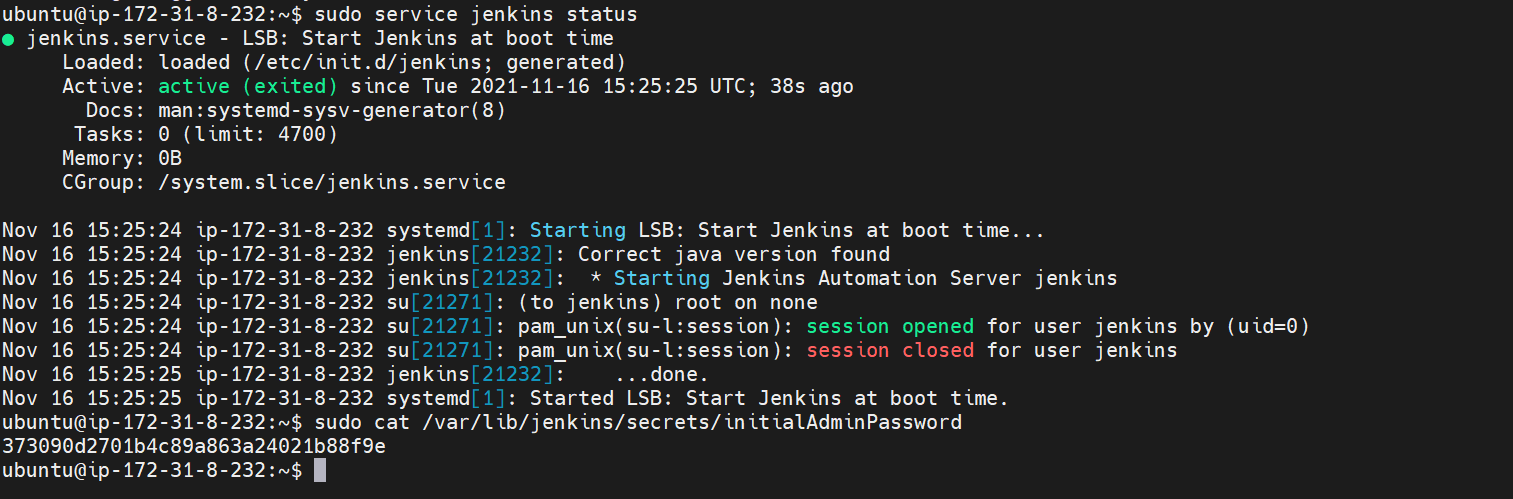
Using these commands

* wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add –
* sudo sh -c 'echo deb https://pkg.jenkins.io/debian-stable binary/ > /etc/apt/sources.list.d/jenkins.list'
* sudo apt-get update
* sudo apt-get install Jenkins
* sudo service jenkins status

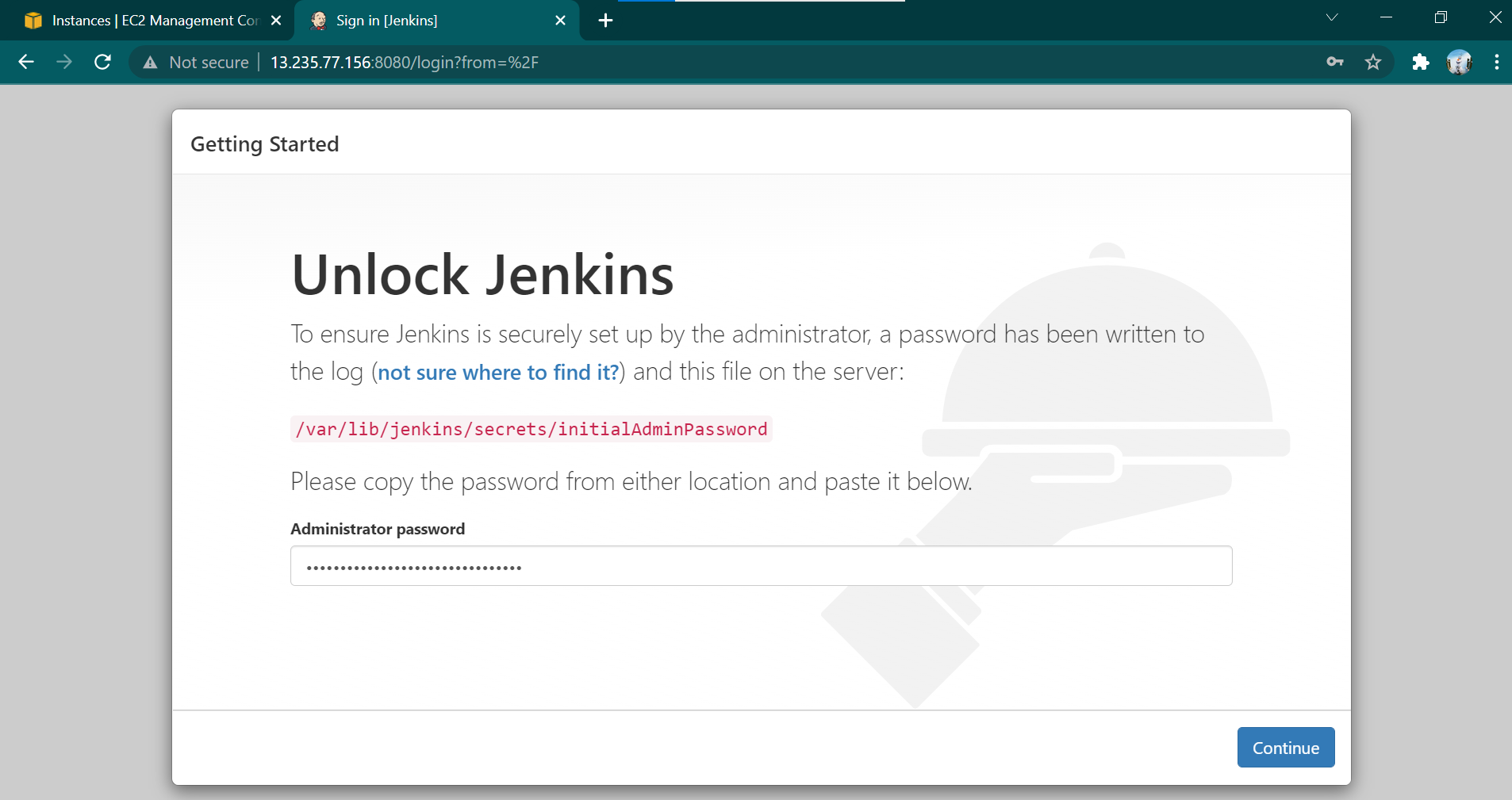


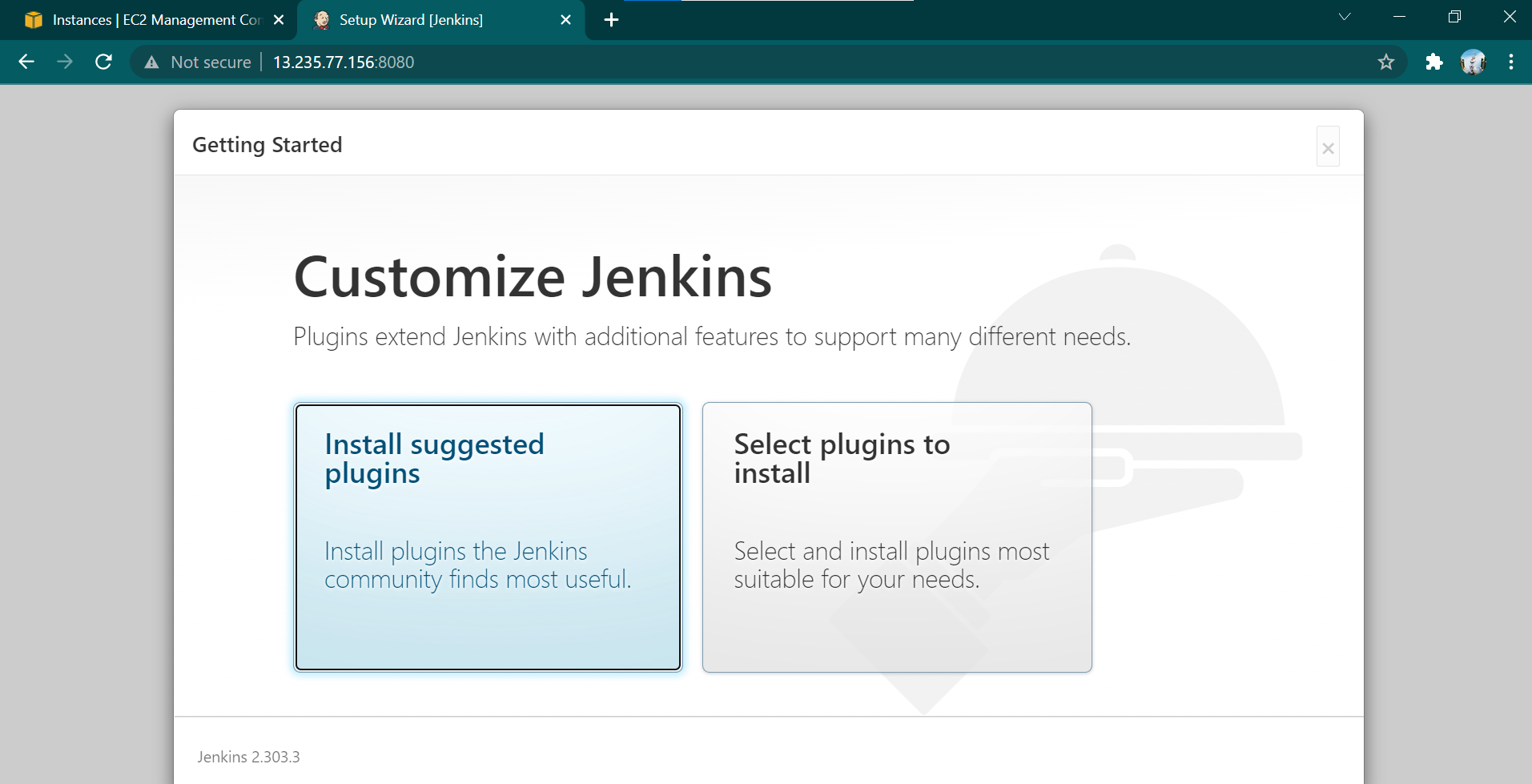
1. **Configure Jenkins**

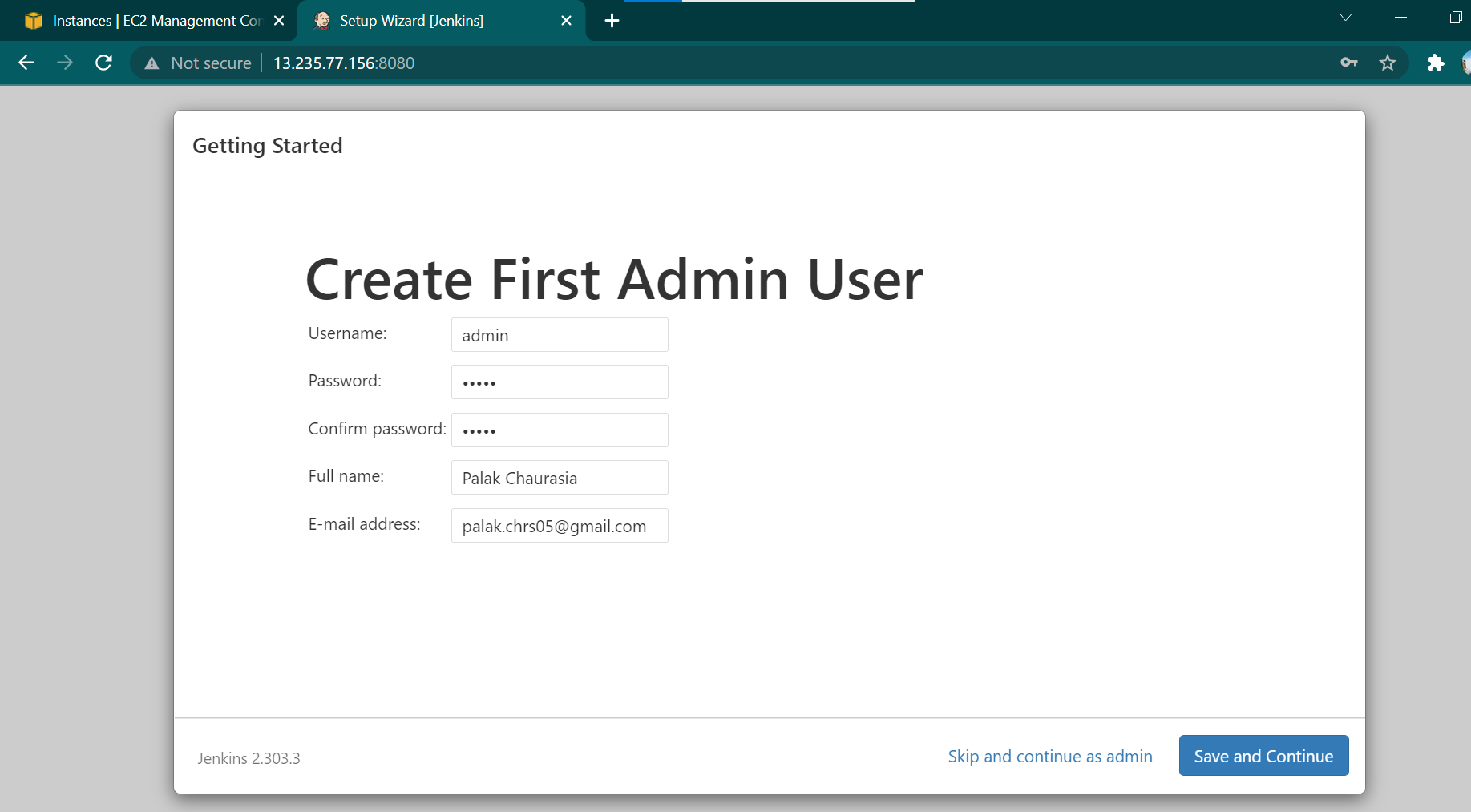
****

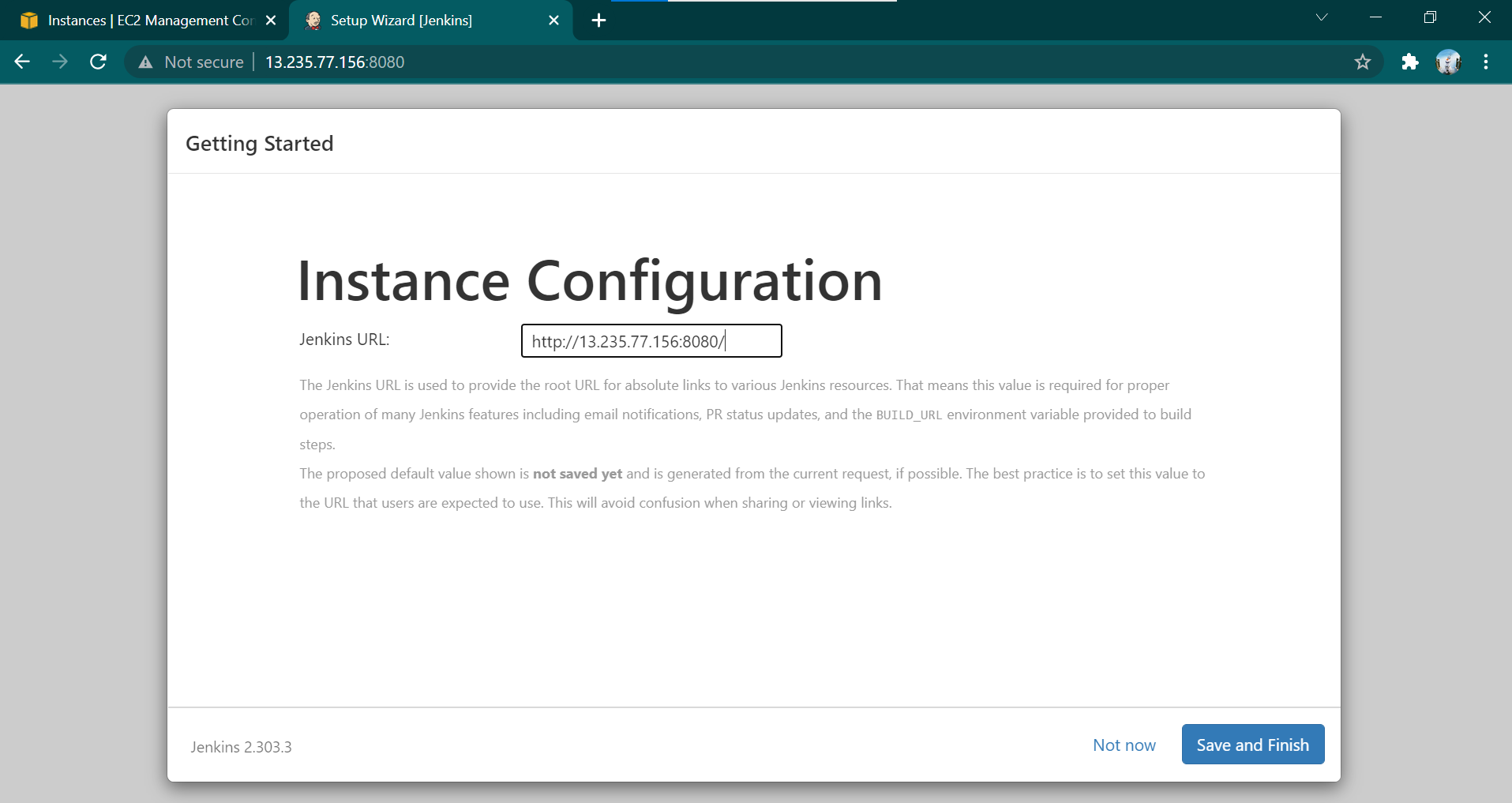
****

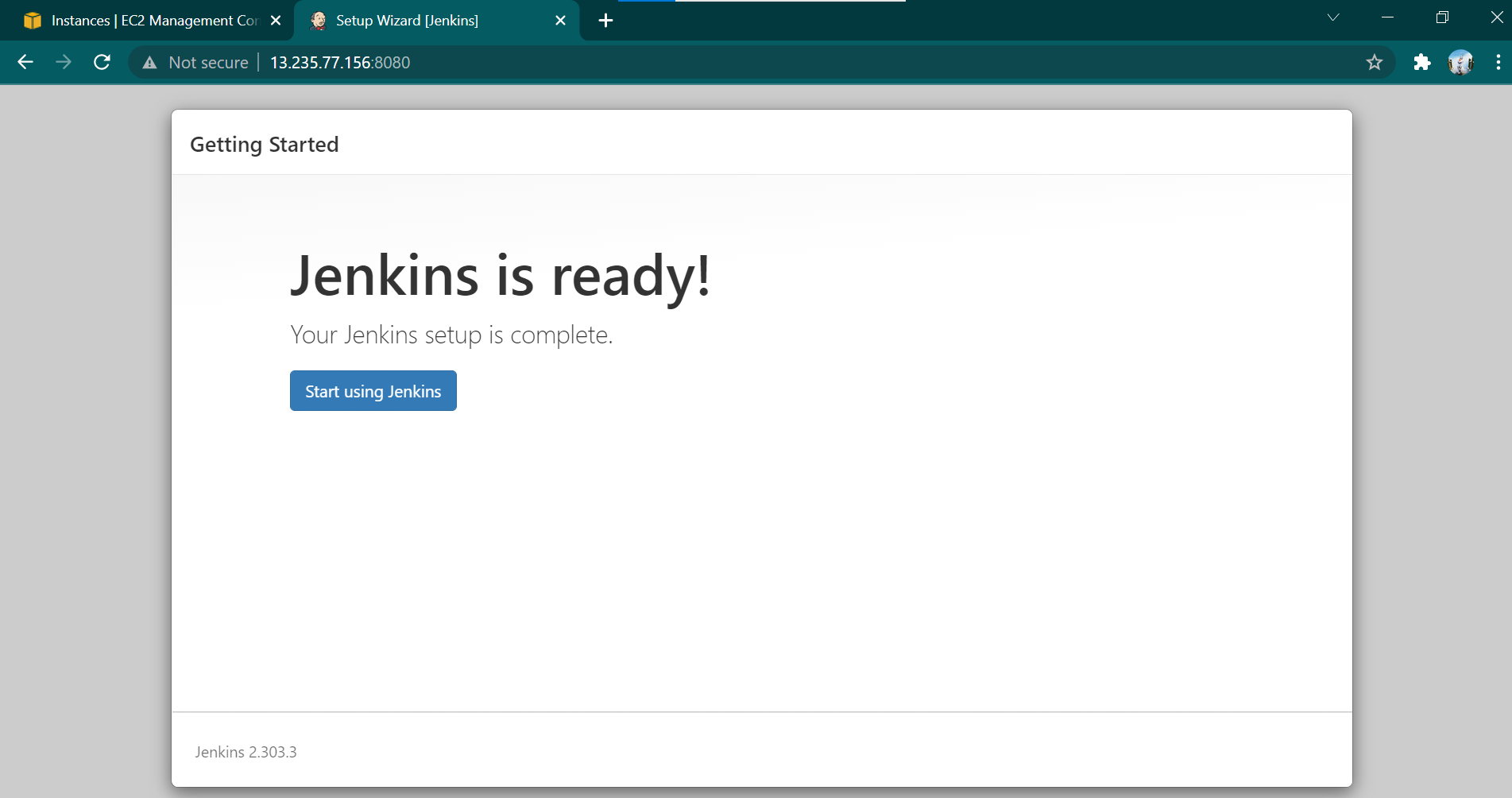
* Run http://13.235.77.156:8081/

****

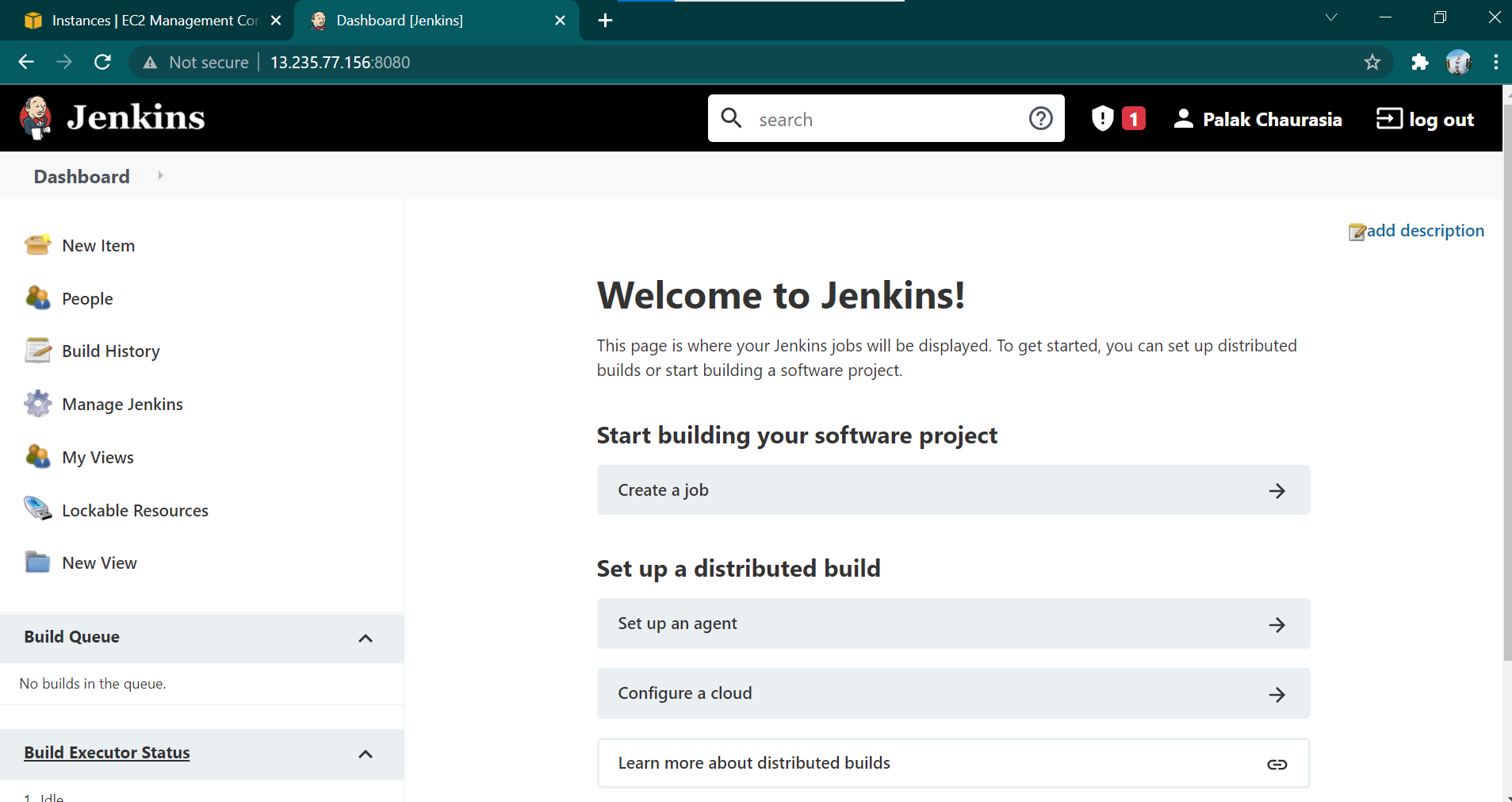


Password:admin****

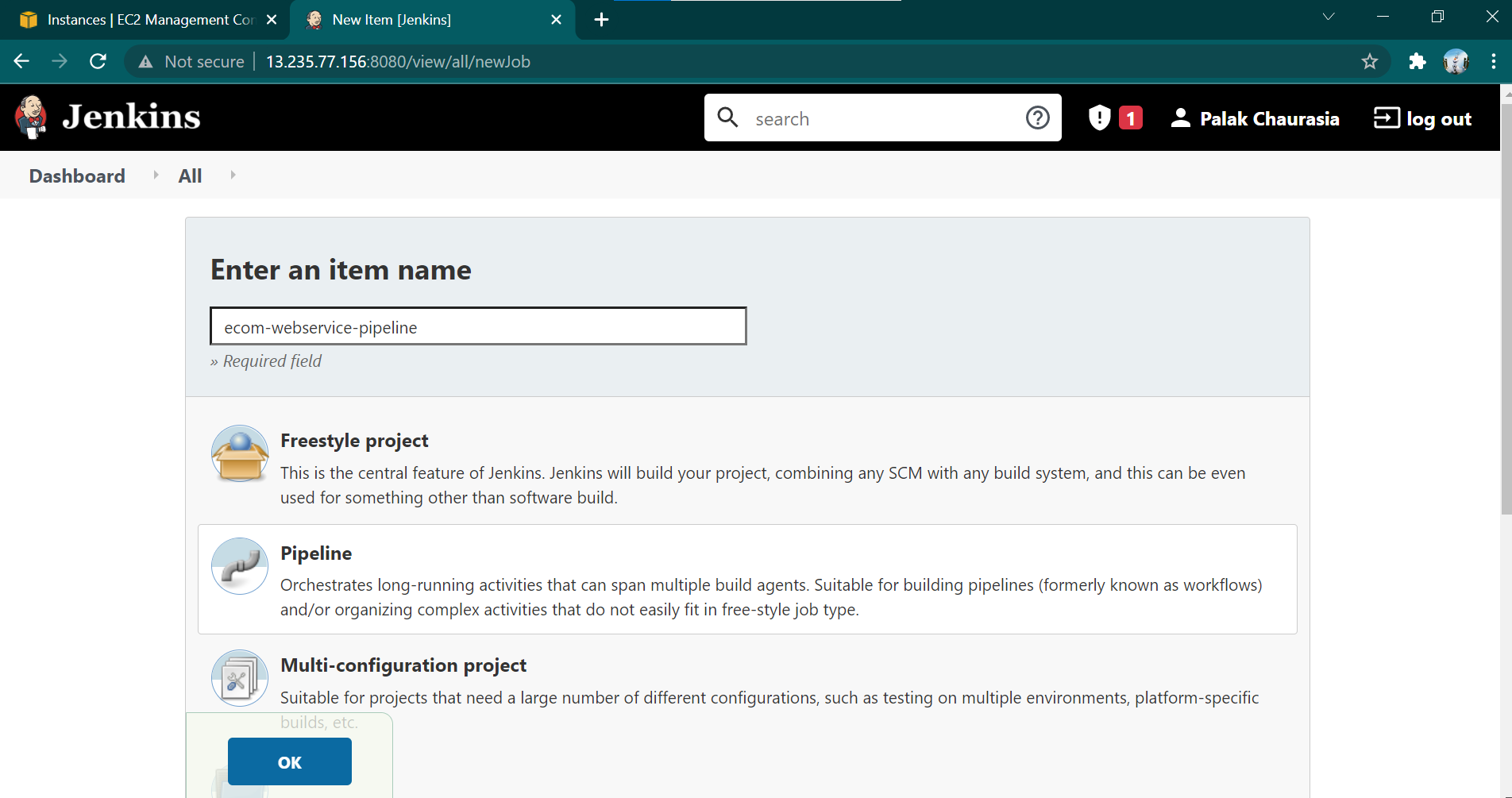


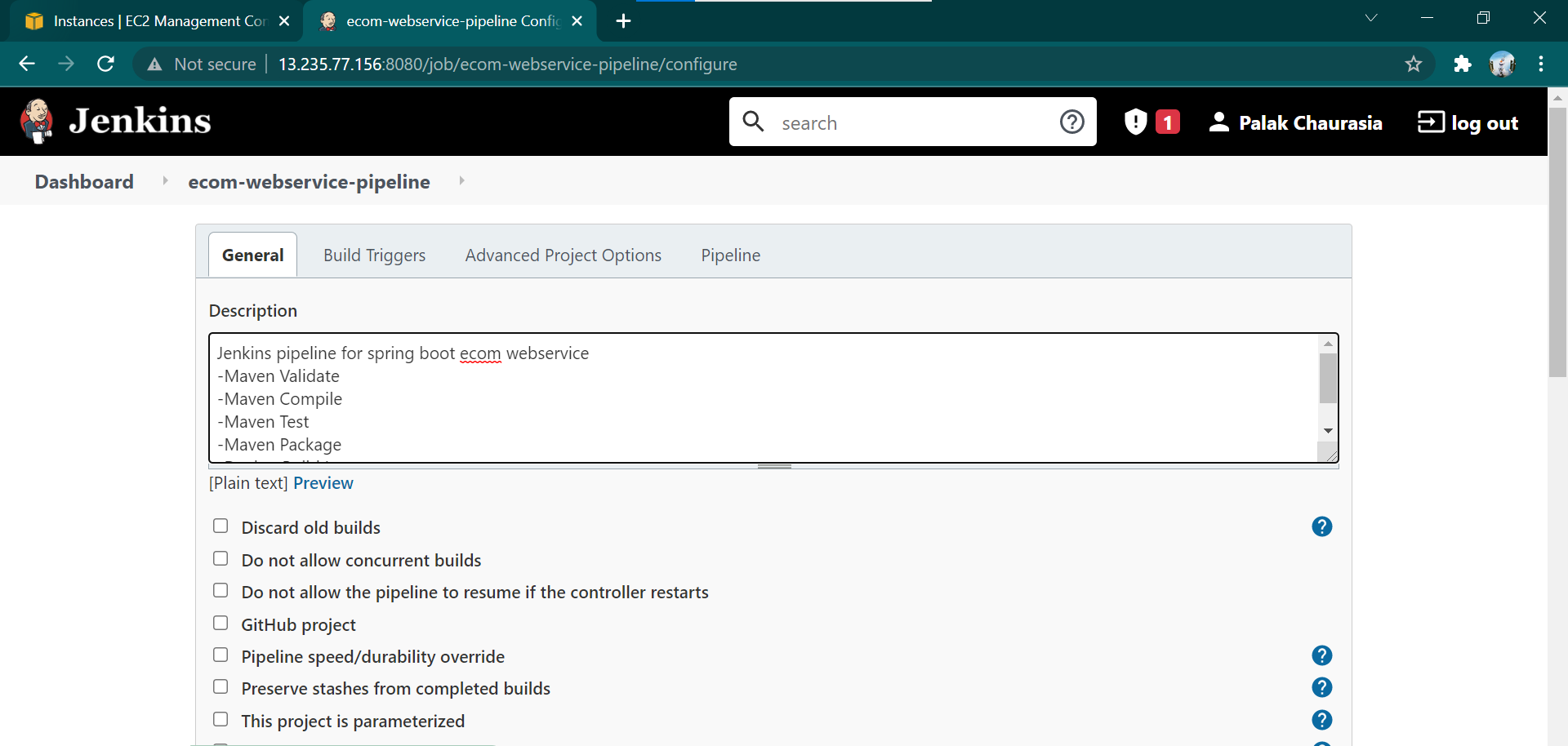
****

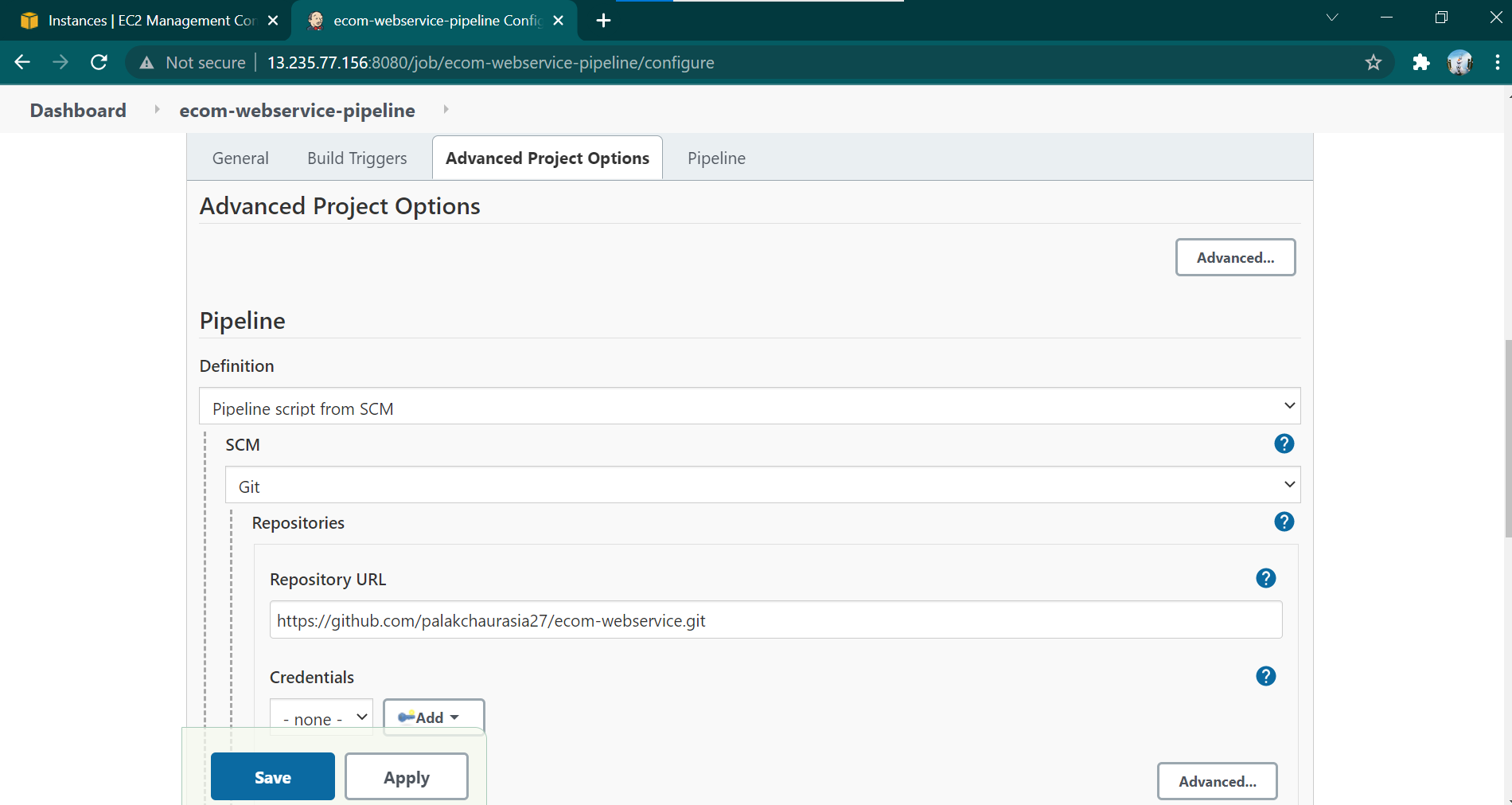
1. **Configure CI Pipeline**

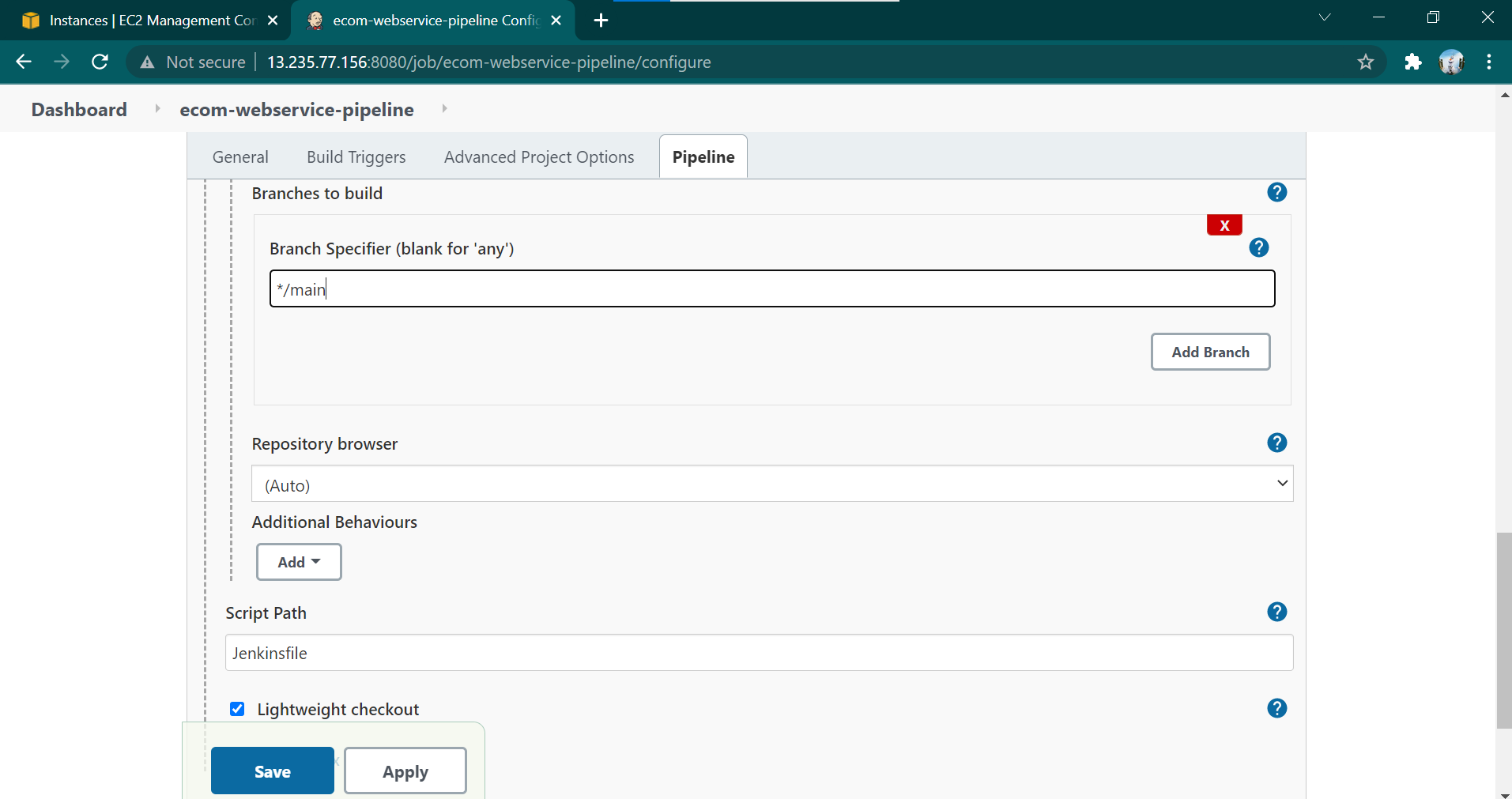


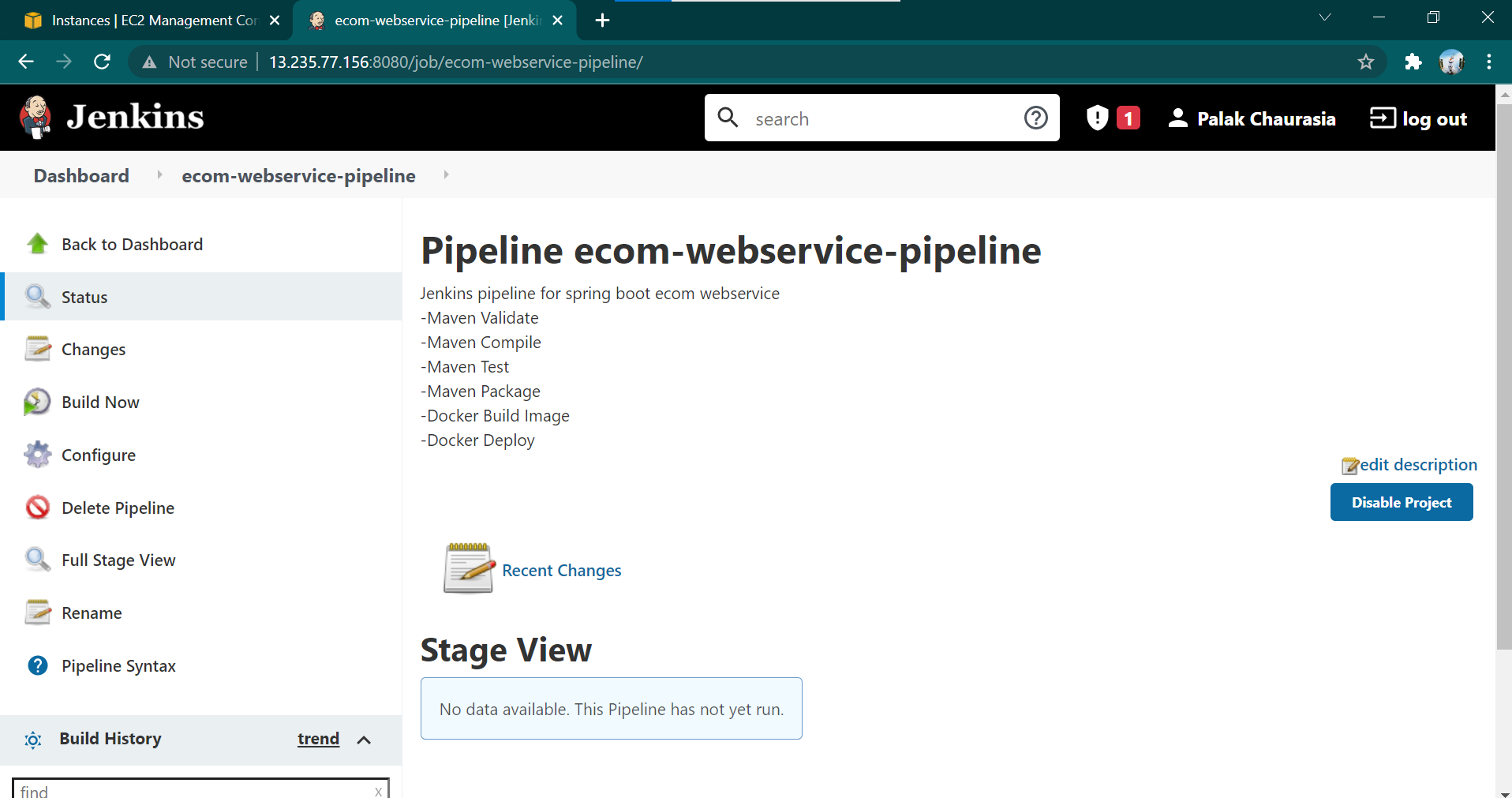
* Add new item: ecom-webservice pipeline

****

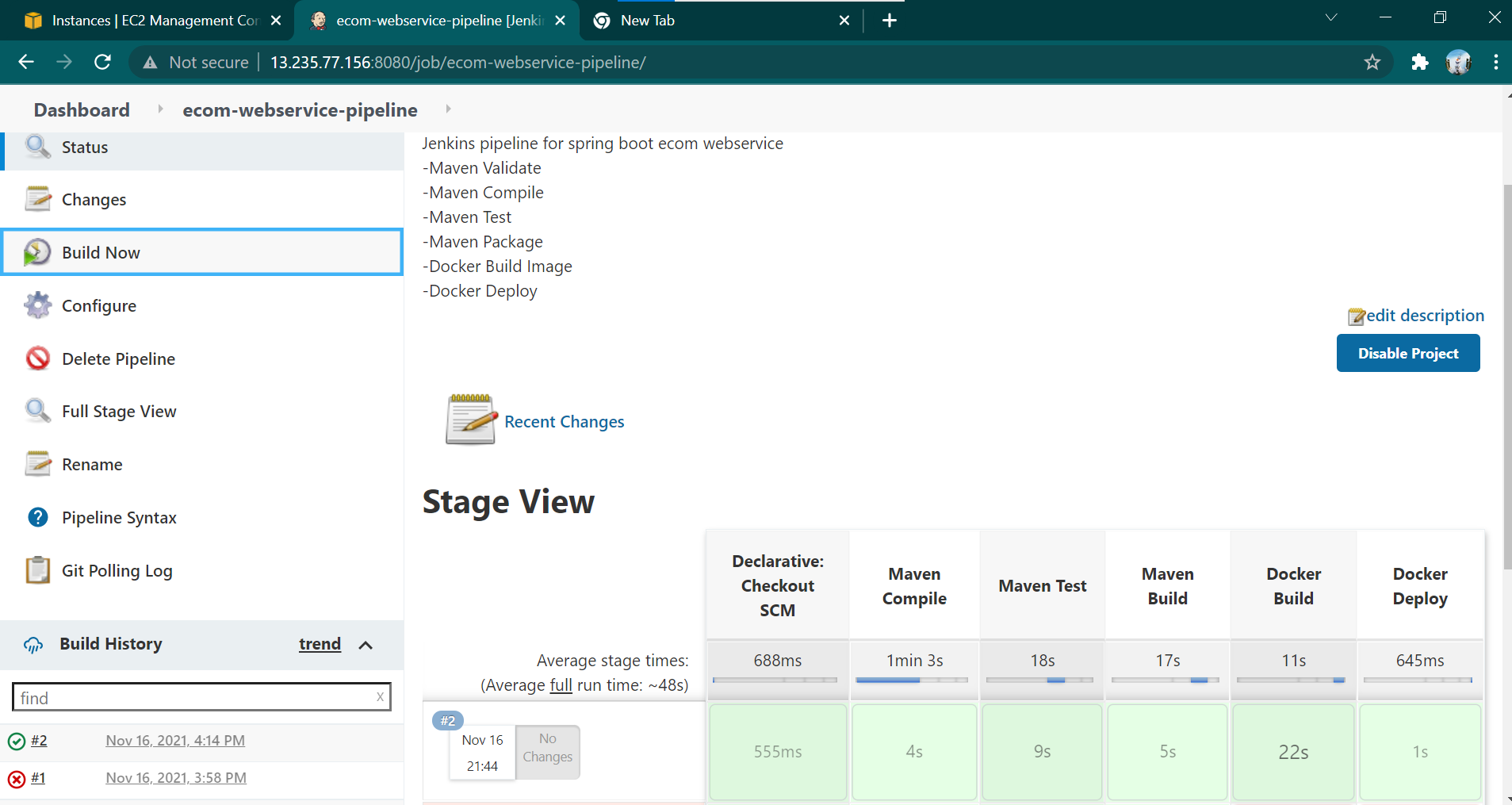
****

****

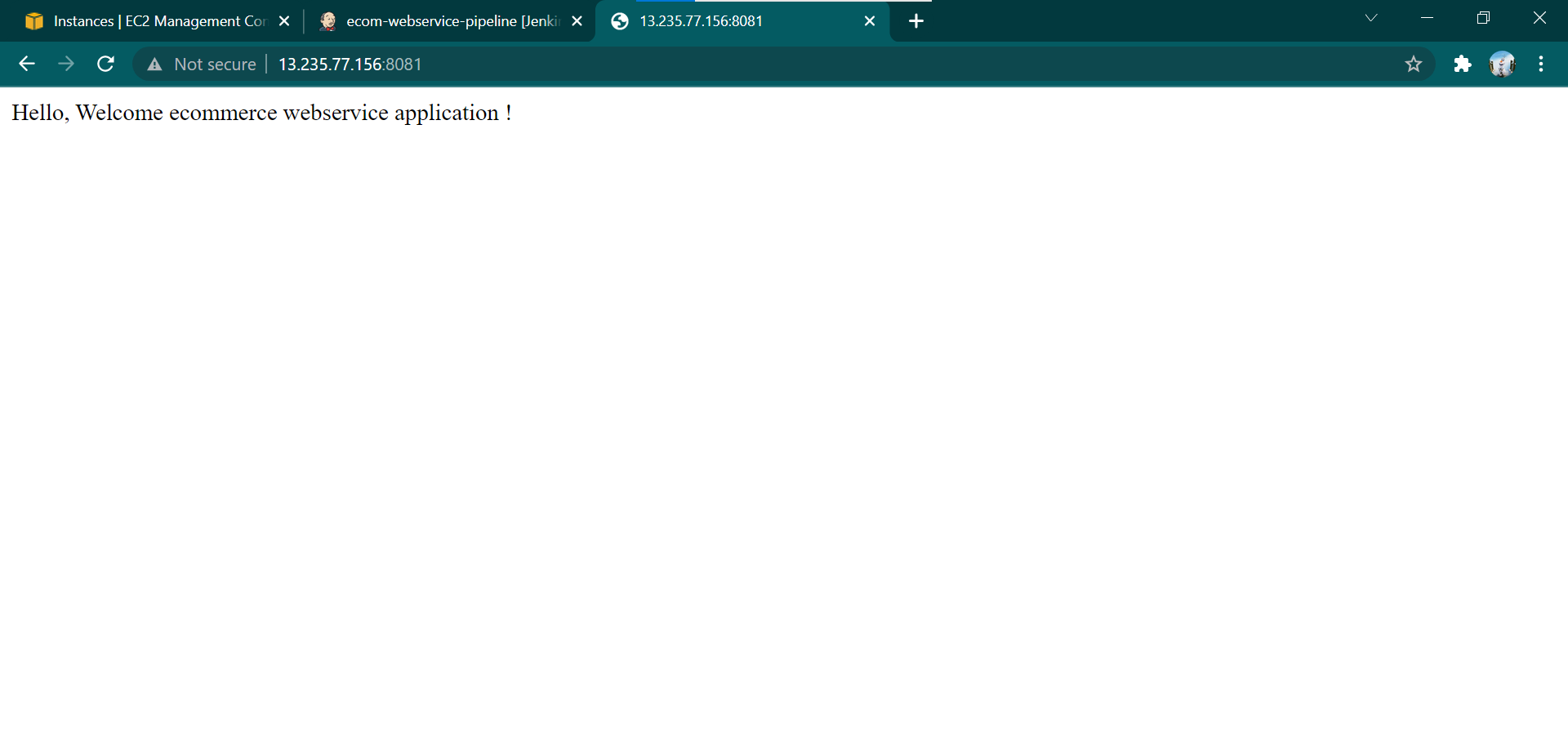
****

****

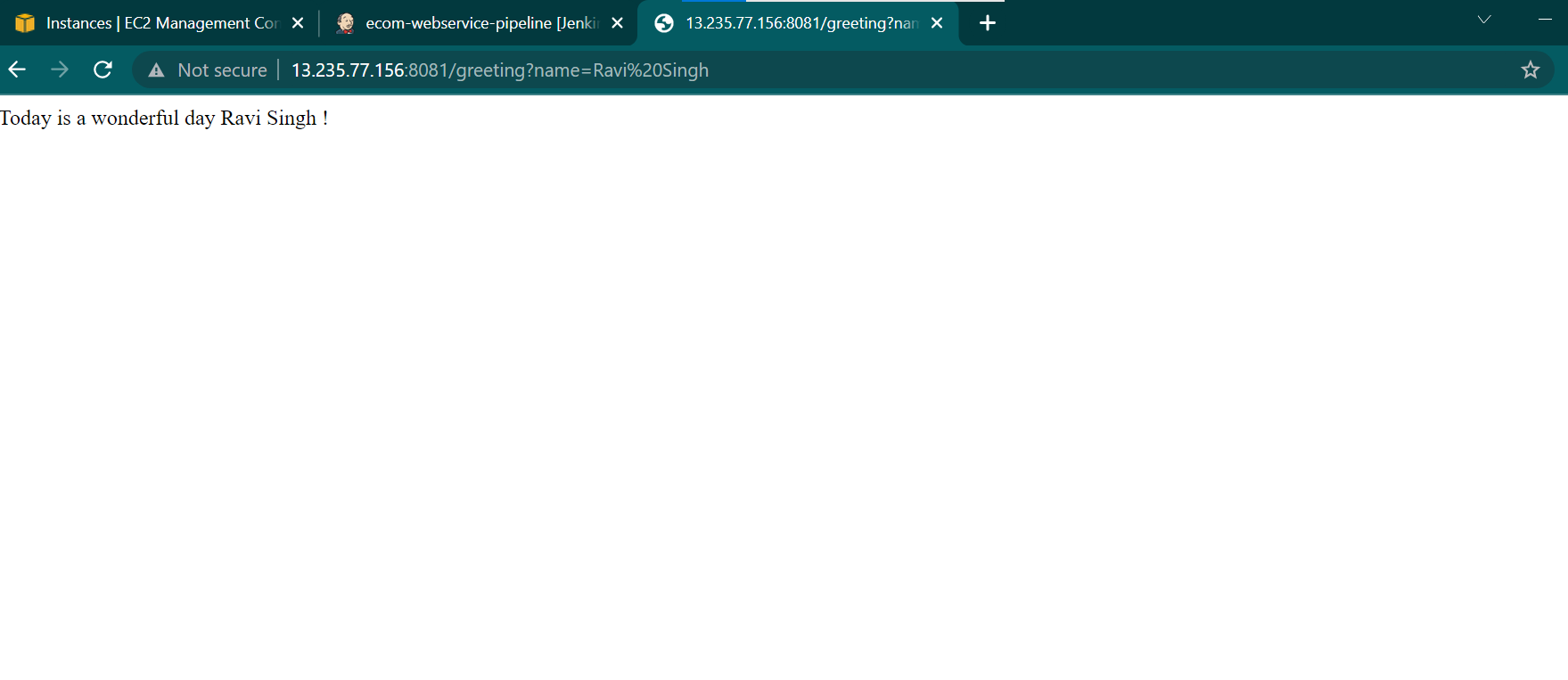
* Build Now

****

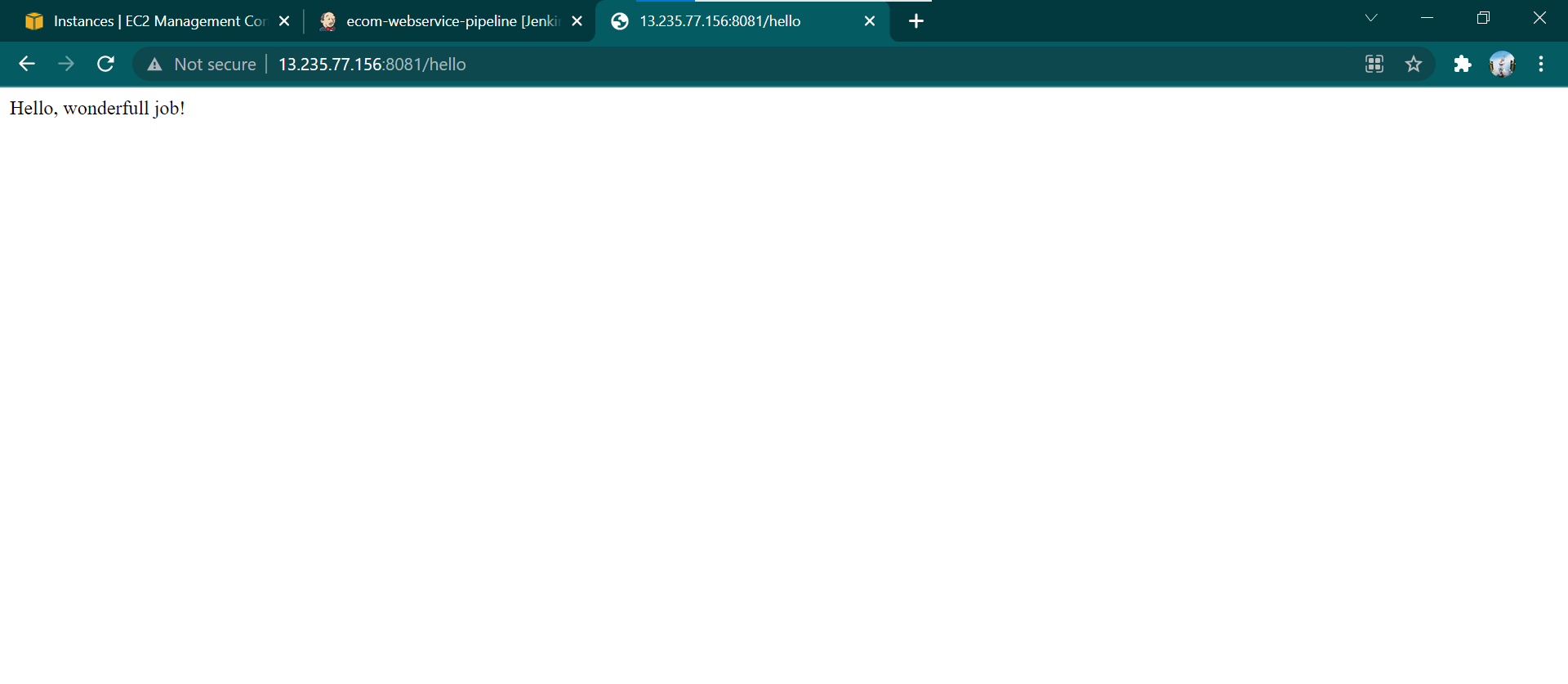
1. **Steps to Run**
2. Run <http://13.235.77.156:8081/>
3. Run <http://13.235.77.156:8081/greeting?name=Ravi%20Singh>
4. Run <http://13.235.77.156:8081/hello>
5. Run http://13.235.77.156:8081/employees/
6. https://github.com/palakchaurasia27/ecom-webservice.git
7. **Steps To Test**
   1. Run IP <http://13.235.77.156:8081/>



* 1. Run <http://13.235.77.156:8081/greeting?name=Ravi%20Singh>



* 1. Run <http://13.235.77.156:8081/hello>



* 1. Run <http://13.235.77.156:8081/employees/>

