# **Capstone Project – Medicare App**

**Project Objective & Background:** My Medicare Application is an frontend online movie ticket booking web application with a rich and user-friendly interface developed using Angular framework and also dynamic and responsive web application for medical products online.

**Name: - Priyanshi Sharma**

**Github : -** [**https://github.com/7Priyanshi/Capstone\_Project\_FSD**](https://github.com/7Priyanshi/Capstone_Project_FSD)

**Core Concepts used for Project:**

1. Java - 1.8

2. Maven - 3.x.x

3. Spring Boot - 2.2.1.RELEASE

4. Spring Security

5. JWT (Json Web Token package)

6. Spring Data JPA

7. MySQL

8. H2-Database

9. PostgreSQL

10. Lombok

11. Git and GitHub

12. Agile Scrum Methodology

13. Docker

14. Jenkins

15. Angular - 11

16. Angular Material

17. Bootstrap 5.1

18. HTML

19. CSS

20. JAVASCRIPT

**Sprint Planning**

The Implementation is done in Eight sprints which are mentioned below:

**Sprint 1:**  
Goal: Set up the project, define the architecture, and implement the basic functionality of the web application.

* Set up the development environment and create the project structure.
* Define the architecture of the web application, including the database schema, APIs, and user interface components.
* Implement the login and registration functionality.
* Implement the CRUD operations for the medicine categories.

**Sprint 2:**  
  
Goal: Implement the main features of the web application, such as searching and ordering medicines, and ensure that the application is responsive.

* Implement the search functionality for medicines, including filtering and sorting options.
* Implement the shopping cart and checkout functionality.
* Implement the user profile page and settings.
* Ensure that the web application is responsive, i.e., it looks good and functions well on different devices and screen sizes.

**Sprint 3:**  
Goal: Improve the user experience of the web application by adding features such as reviews, ratings, and recommendations.

* Implement the review and rating functionality for medicines.
* Implement the recommendation system for medicines based on user preferences and purchase history.
* Implement the notification system for important events, such as order confirmation and shipment.
* Improve the user interface and user experience based on feedback from users and stakeholders.

**Sprint 4:**  
  
Goal: Testing, Deployment, and CI/CD.

* Implement automation testing to test the application thoroughly.
* Set up a CI/CD pipeline using Jenkins to automate the build and deployment process.
* Deploy the application on an AWS EC2 instance.
* Ensure the application is secure by implementing necessary security features like HTTPS.

**Screenshot**

**Graphical user interface, text, application

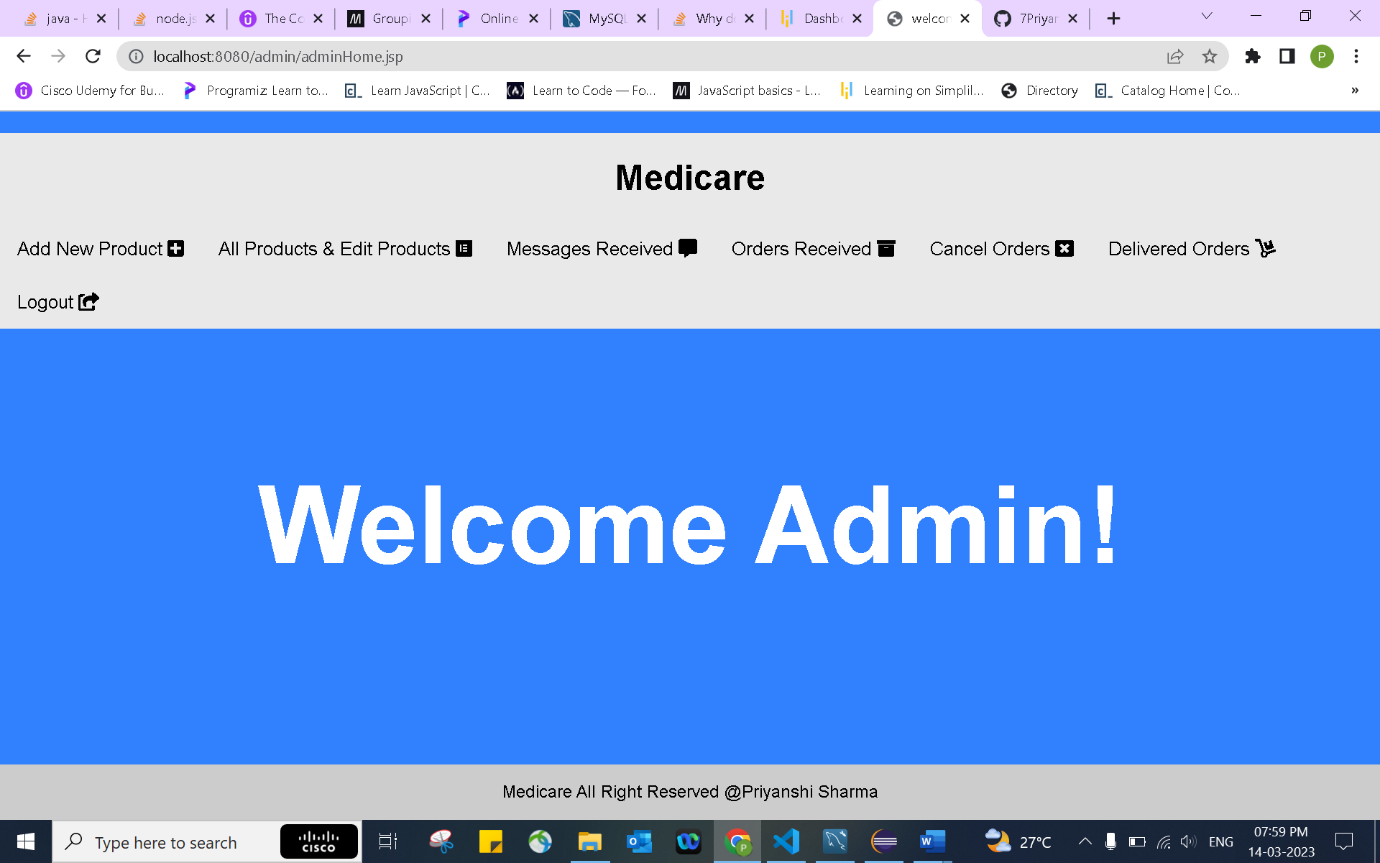
Description automatically generated**

**Graphical user interface, application

Description automatically generated**

**Graphical user interface, application, Teams

Description automatically generated**

****

**Graphical user interface

Description automatically generated**

**Graphical user interface, text, application, Word, email

Description automatically generated**

**Graphical user interface, text, application, email

Description automatically generated**

**Graphical user interface, text, application, Word

Description automatically generated**

**Text

Description automatically generated**

**Text

Description automatically generated**

**Graphical user interface, text, application, email

Description automatically generated**

**Graphical user interface, application

Description automatically generated**

**Graphical user interface, application

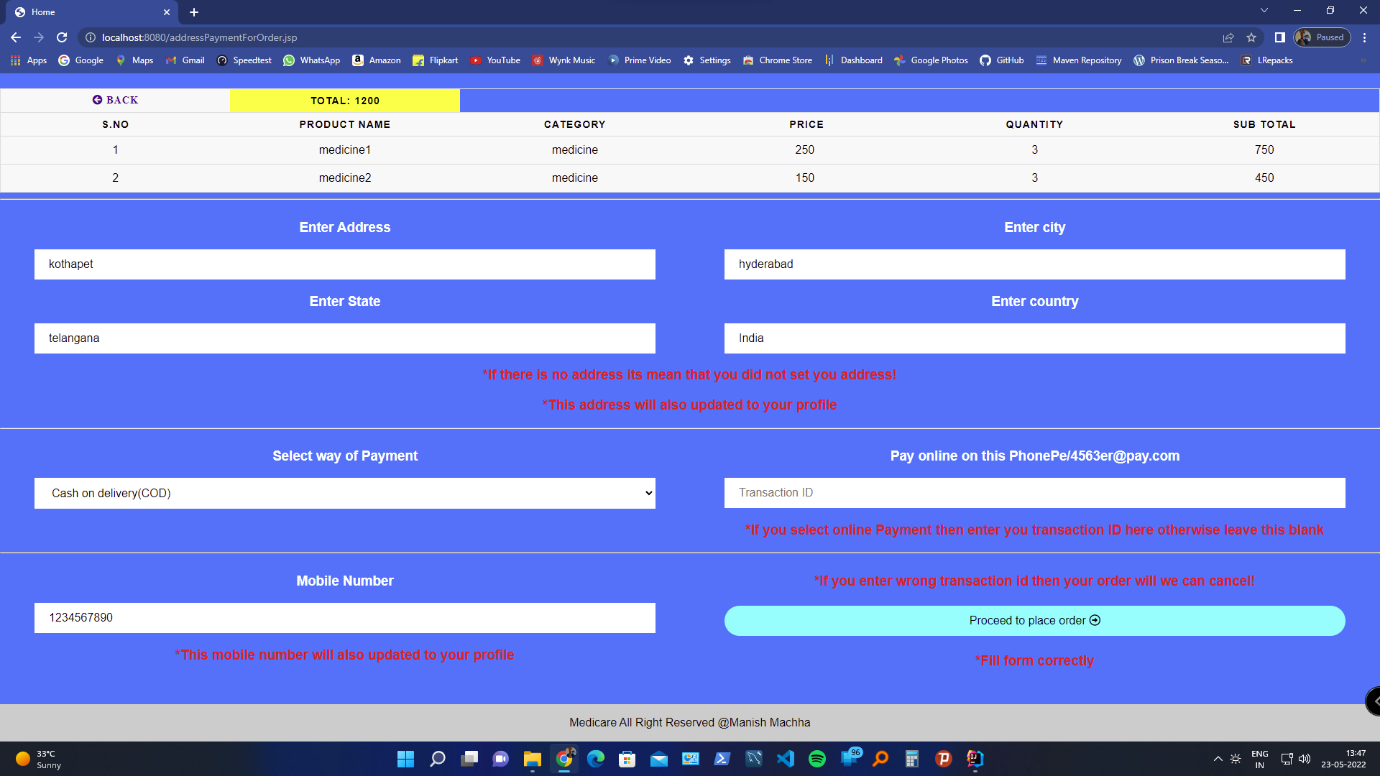
Description automatically generated**

**Graphical user interface, text, application, email

Description automatically generated**

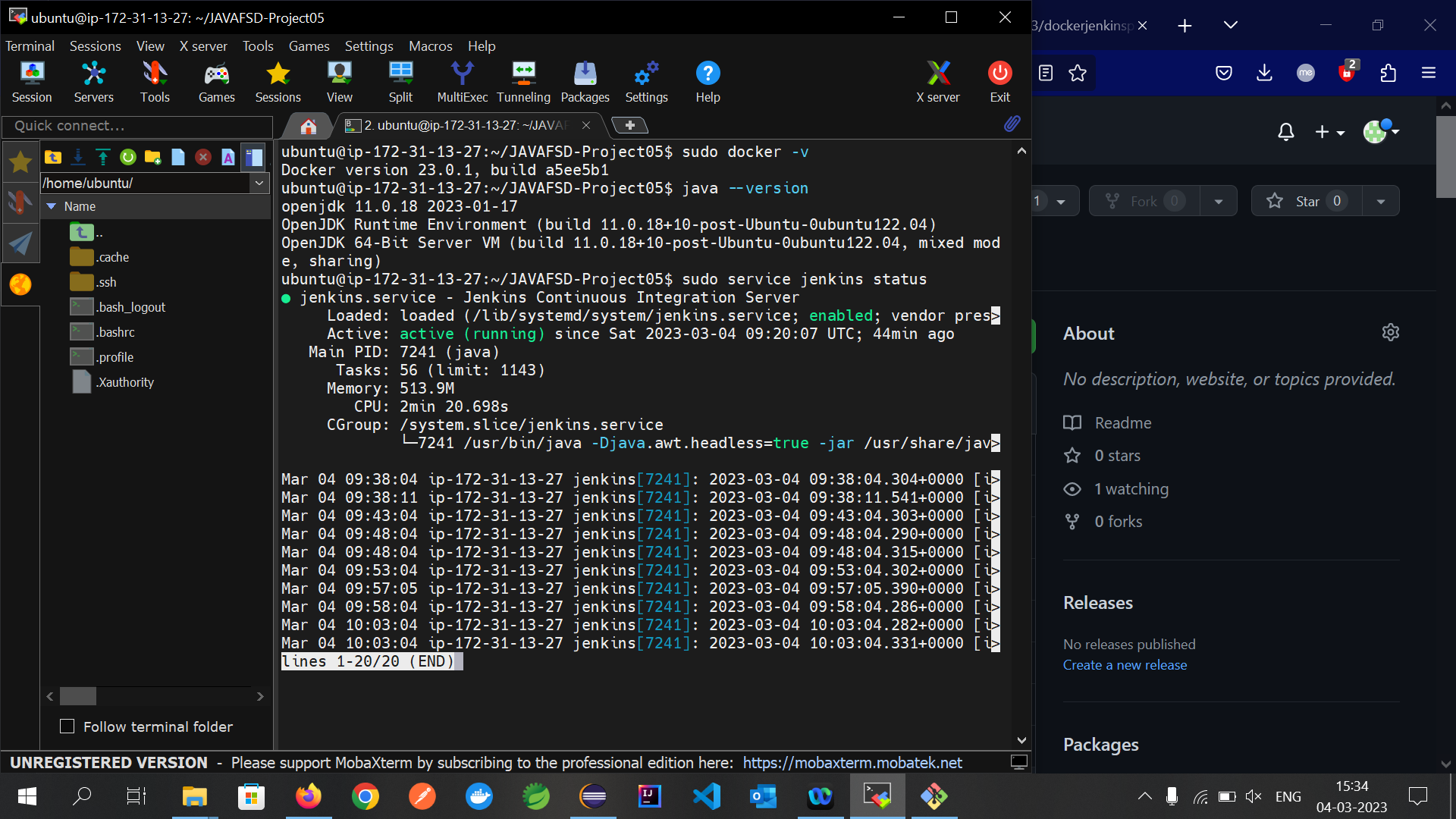
**Graphical user interface, text, application, email

Description automatically generated**

****

**Graphical user interface, text

Description automatically generated**

****

**Graphical user interface, text, application

Description automatically generated**