**Project Report**

**CricketApp**

Submitted by

NIJIL P.T

KAVITHA M

SOORAJ T.S

**CHAPTER ONE**

1.1. Introduction

Cricket is one of international sport and it is widely popular in India. There are number of fans for these kind of sports, and want to know live updates of the matches. Currently there are number of apps and websites for live match scores for individual sports. So the need of such application which can show the match updates of sports is very high. Here we are introducing such a kind of web application.

1.2. Objective

In this project we are creating a Web application which is showing the current and old cricket matches played around the world. Also the details about each match including score and other statistics has to be displayed. Also providing then option for finding current cricket matches, bookmark favourite matches and recommend most liked/favourite matches to user.

1.2.1 Specific Objectives

The specific objectives of the project are:

* Developing user friendly application where user can view current or old cricket matches
* Enabling calendaron the home page as a quick view to the user
* Match statistics is displayed for a selected cricket match
* User can register/login to the application, find, add interested matches to favourite list and view recommended matches.
* Authorized User can add a match to favourite list and should be able to view favourite matches.
* Implementing a recommendation engine

1.3 Significance of the project

Some of the significances of the project are:

* Enables the users to get a fast access on cricket related updates.
* Reminding users about upcoming cricket schedule by providing calendar on home page.
* Recommendation system that is able to store all the unique favourite matches from all the users and maintain counter for number of users added a particular match into favourite list.

**CHAPTER TWO**

Description of existing System

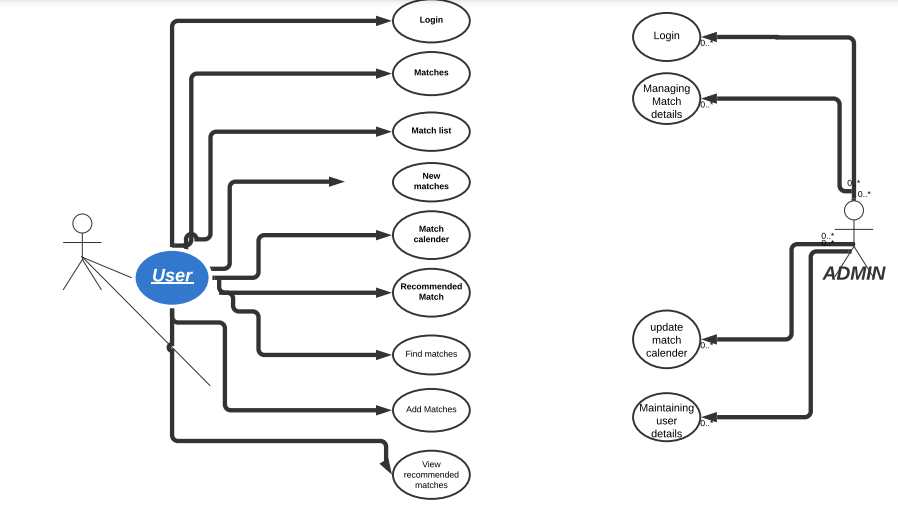
Cricket is one of the most famous games in the world. People are very fond of this game. In the past we use radio and television for getting the game information but now all these things are one click away. Nowadays technology is developed and introducing many website, application etc. One of such application isCricbuzz, it is one of the most popular mobile apps for cricket news and scores in India.Here we are introducing such a system which provides the entire details of a match. This is about showing current and old cricket matches played around the world. It displays details about each match including score and other statistics. The user can view current or old cricket matches. It will also provide schedule of upcoming fixtures on the homepage as a calendar as a quick view to the user.Match statistics can be displayed for a selected cricket match. It provides security by providing private credential for each and every user to log in. Users are allowed to find and add their favorite matches to favorite list and they are also provided with recommendation according to their priorities. This system help to easily access the cricket matches. It’s not a complex one anyone can log in easily and access data about matches. It’s a user friendly and this have a massive impact in our nation due to the cricket history we have.

**CHAPTER THREE**

Object-oriented analysis and design

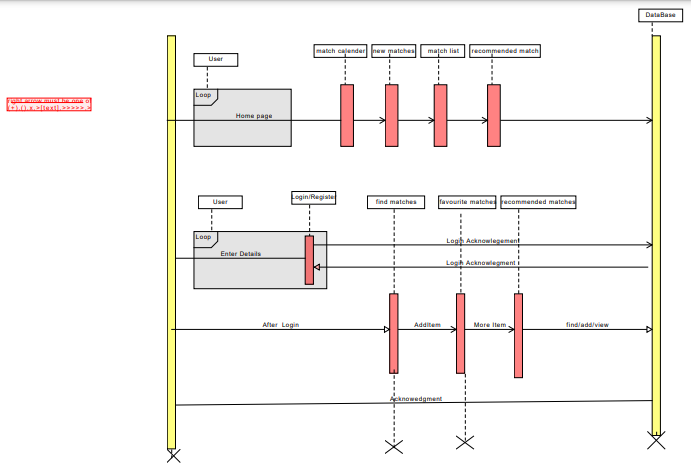
3.1. Use Case diagram

A UML use case diagram is the primary form of system/software requirements for a new software program underdeveloped. Use cases specify the expected behaviour (what), and not the exact method of making it happen (how). Use cases once specified can be denoted both textual and visual representation. A key concept of use case model is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behaviour in the user's terms by specifying all externally visible system behaviour.



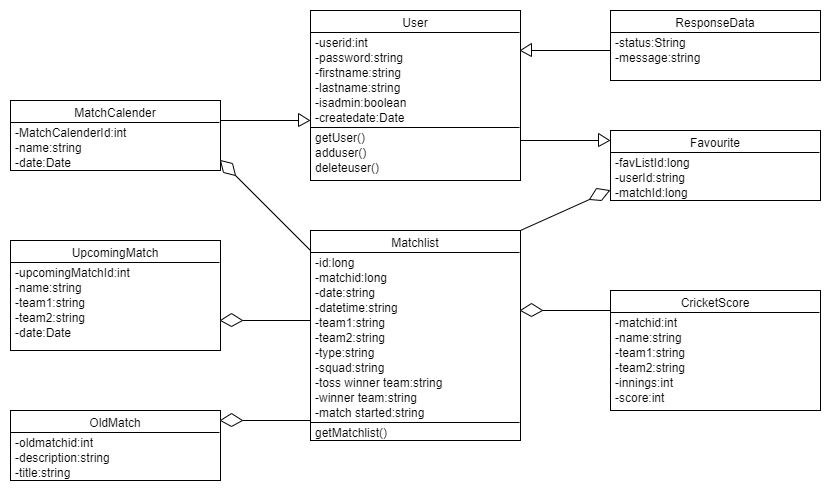
3.2. Sequence Diagram

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration .Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.



3.3. Class Diagram

A class diagram in the Unified Modeling Language is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes,operations, and the relationships among objects.



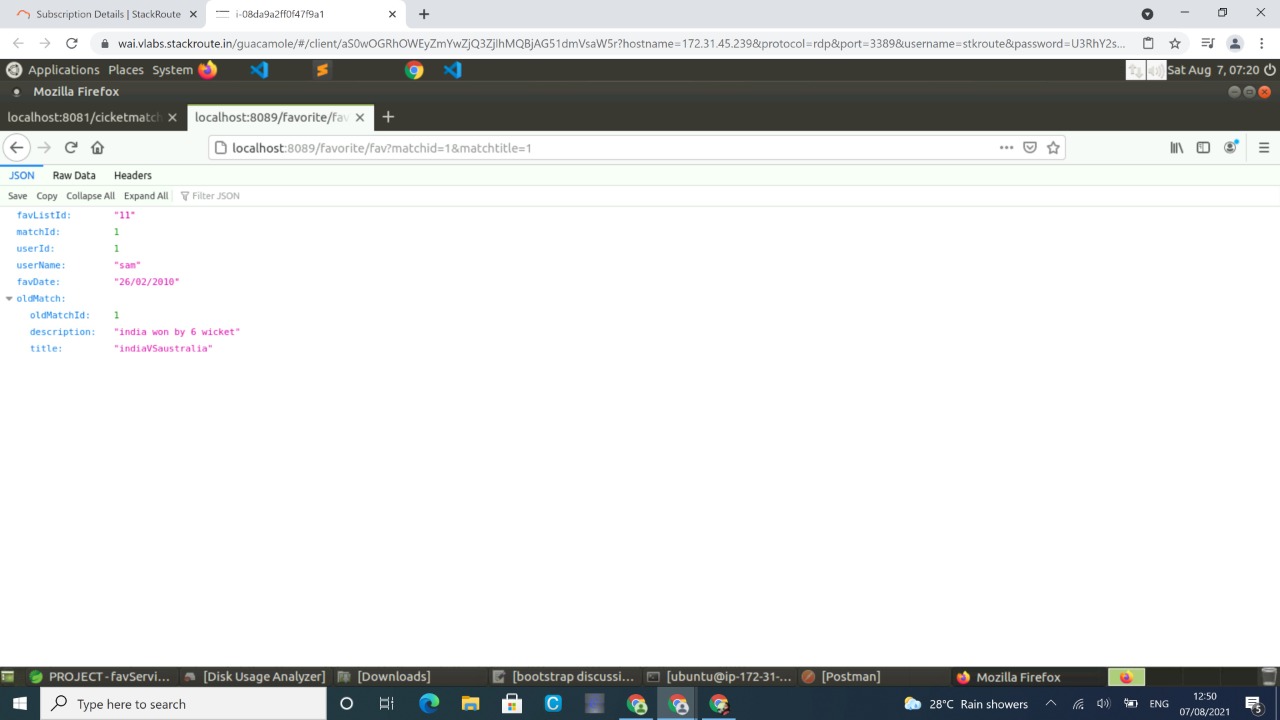
**CHAPTER FOUR**

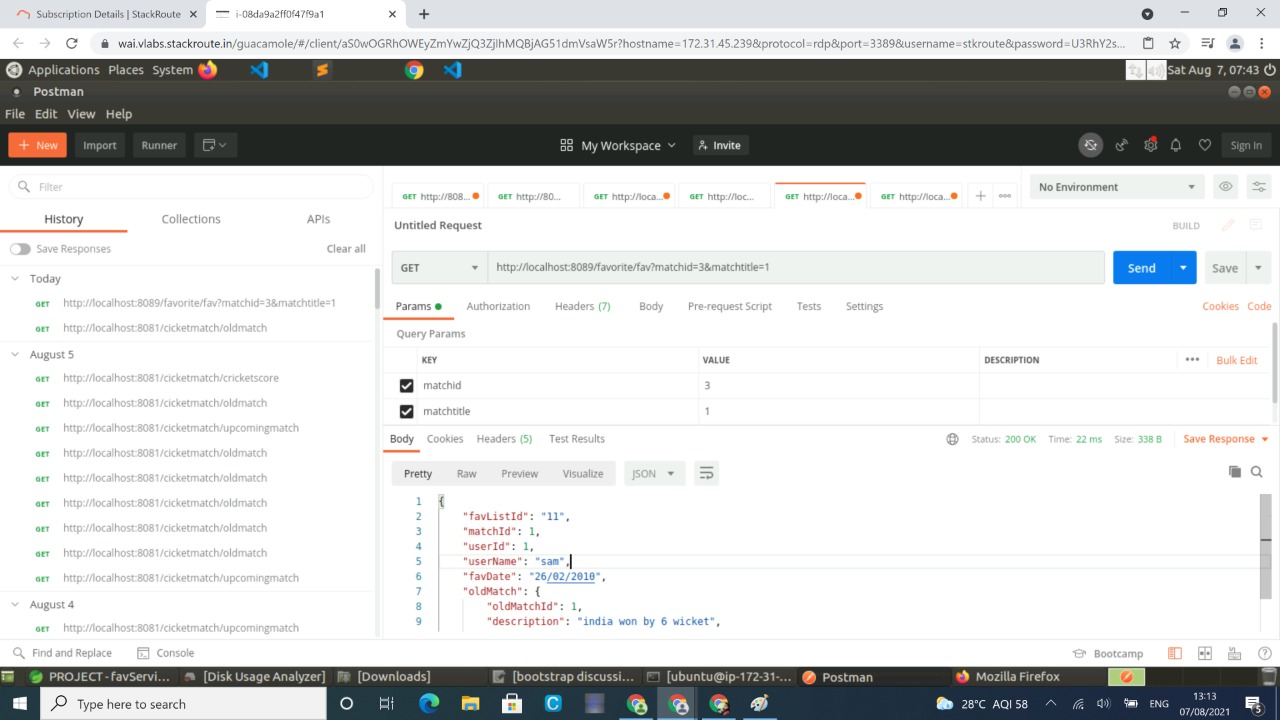
Technologies used in Project

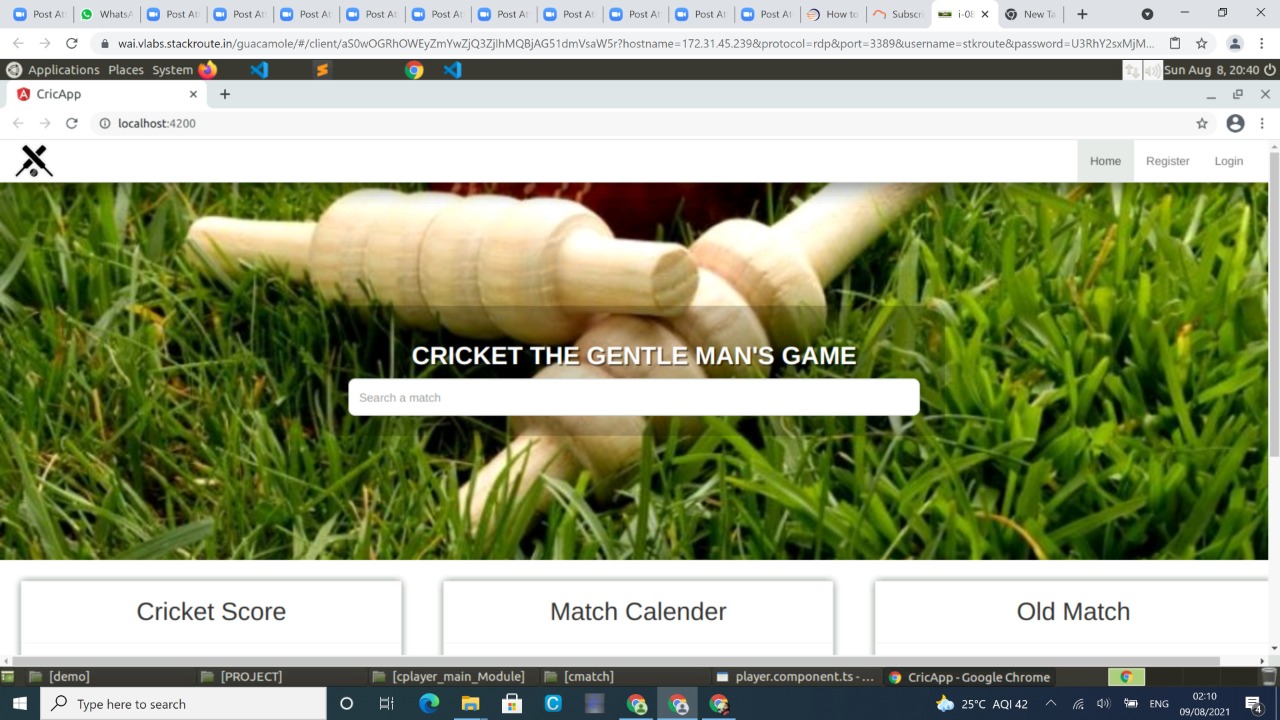
* VCS GITLAB
* Framework: Spring Boot
* Frond end: Swagger/Postman
* Data store:MongoDB/MySQL
* Testing:Spring Boot App
* IDE:STS

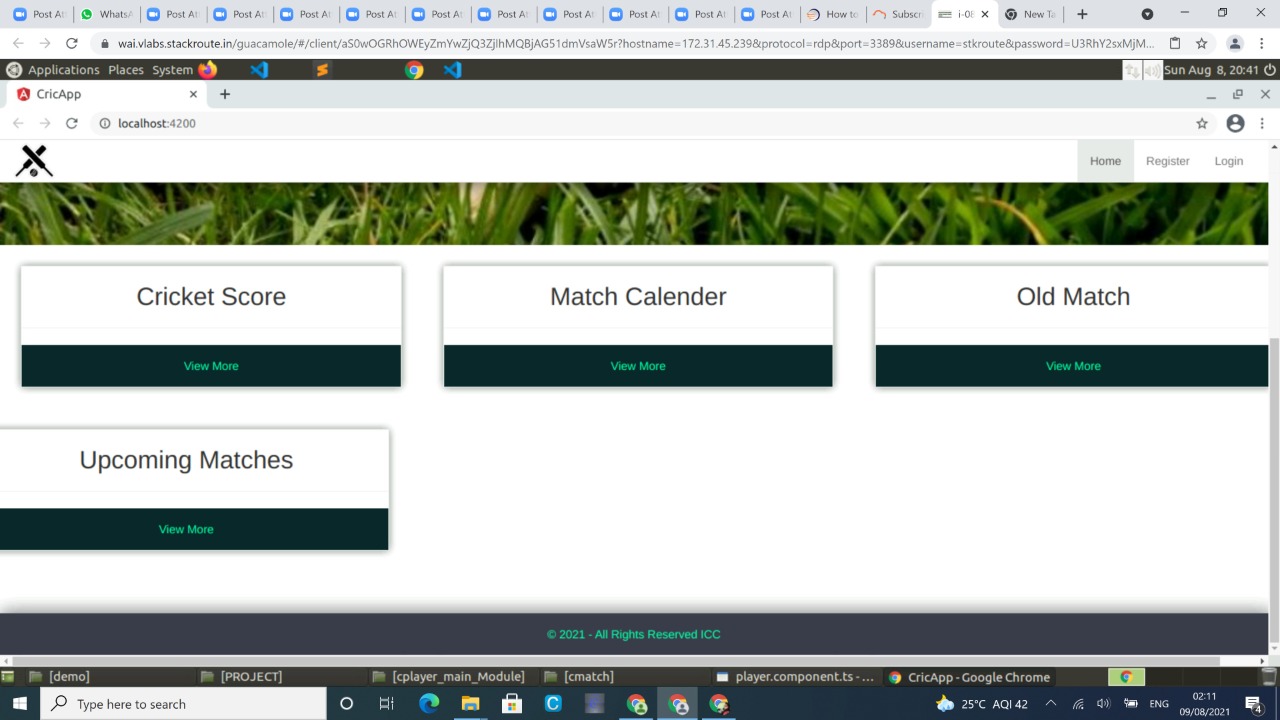
**CHAPTER FIVE**

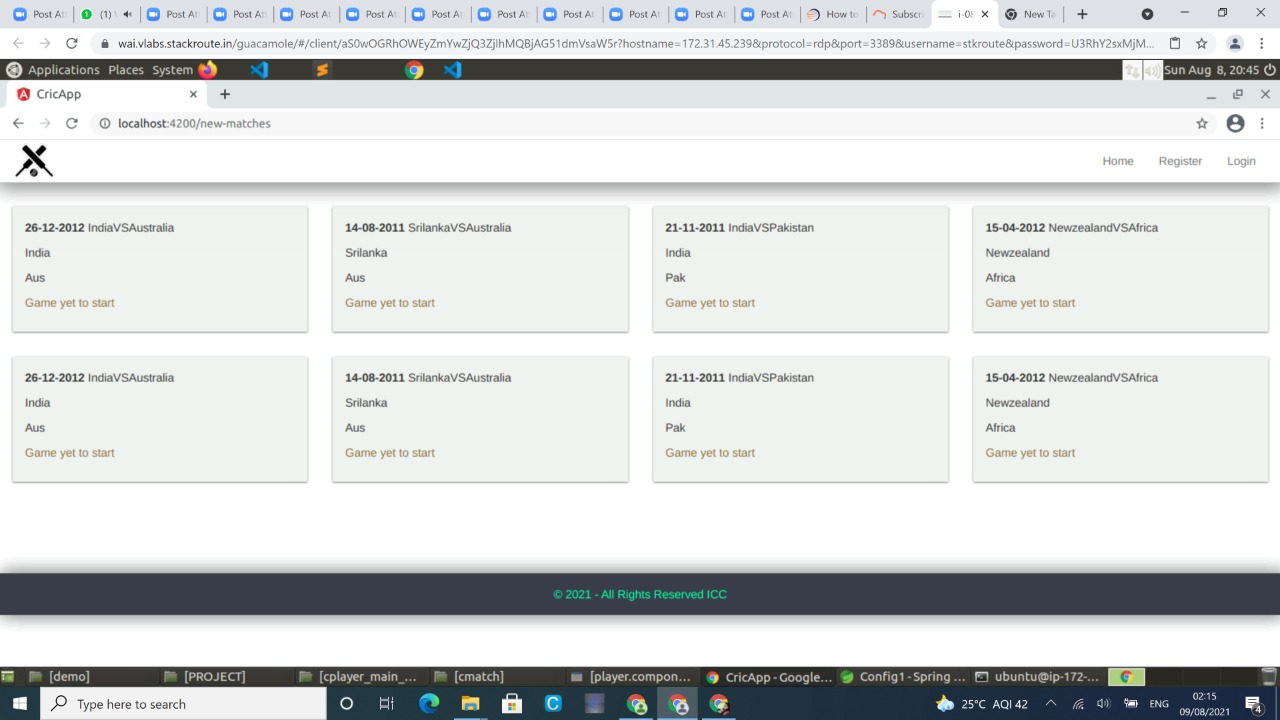
Working Screenshots

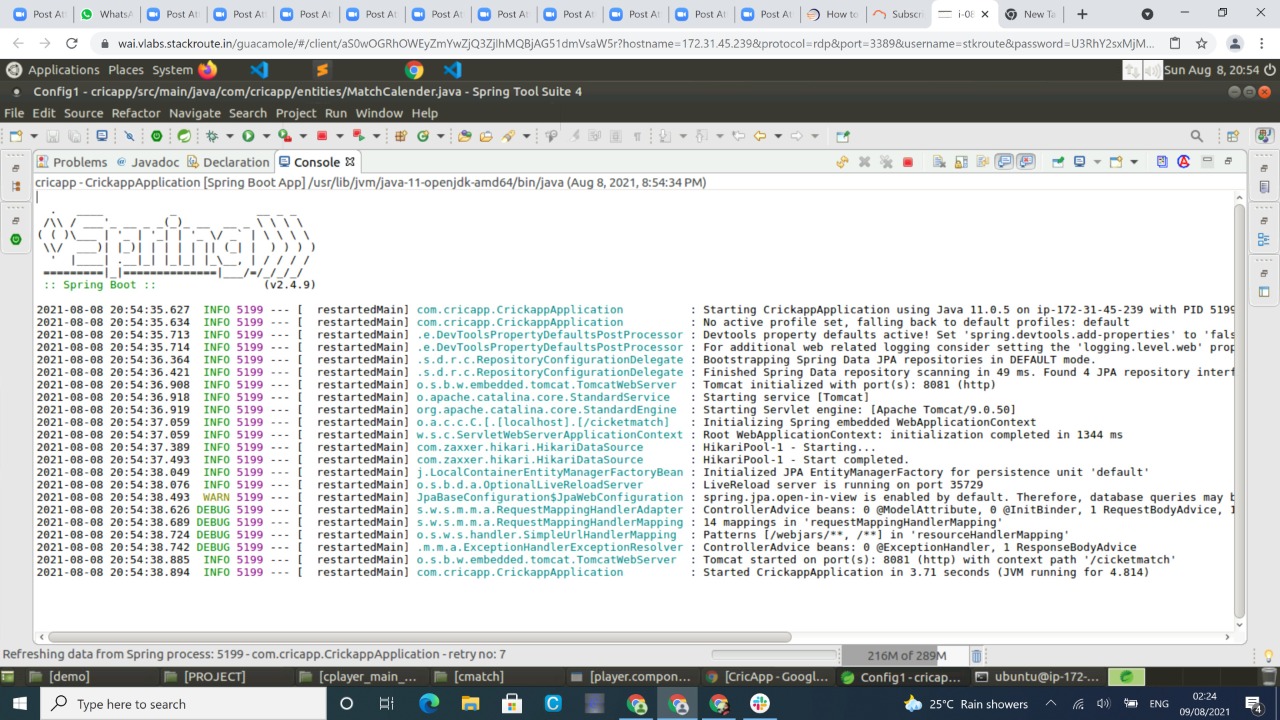


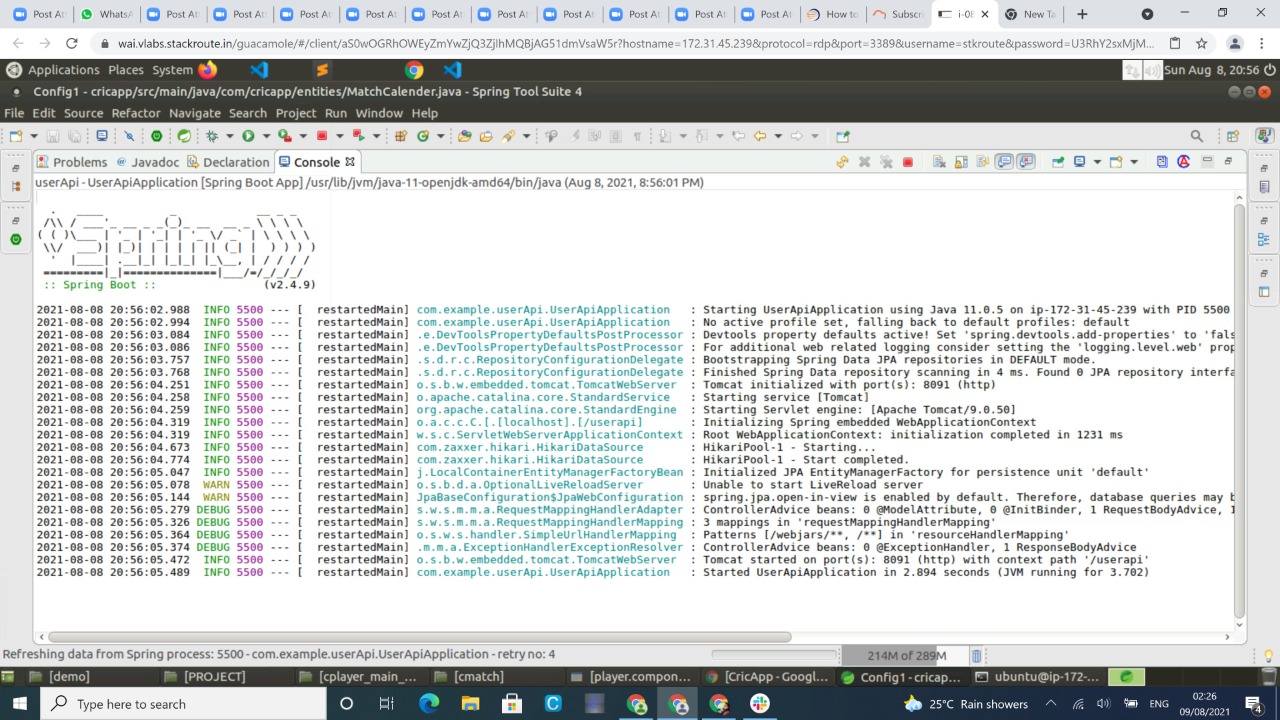


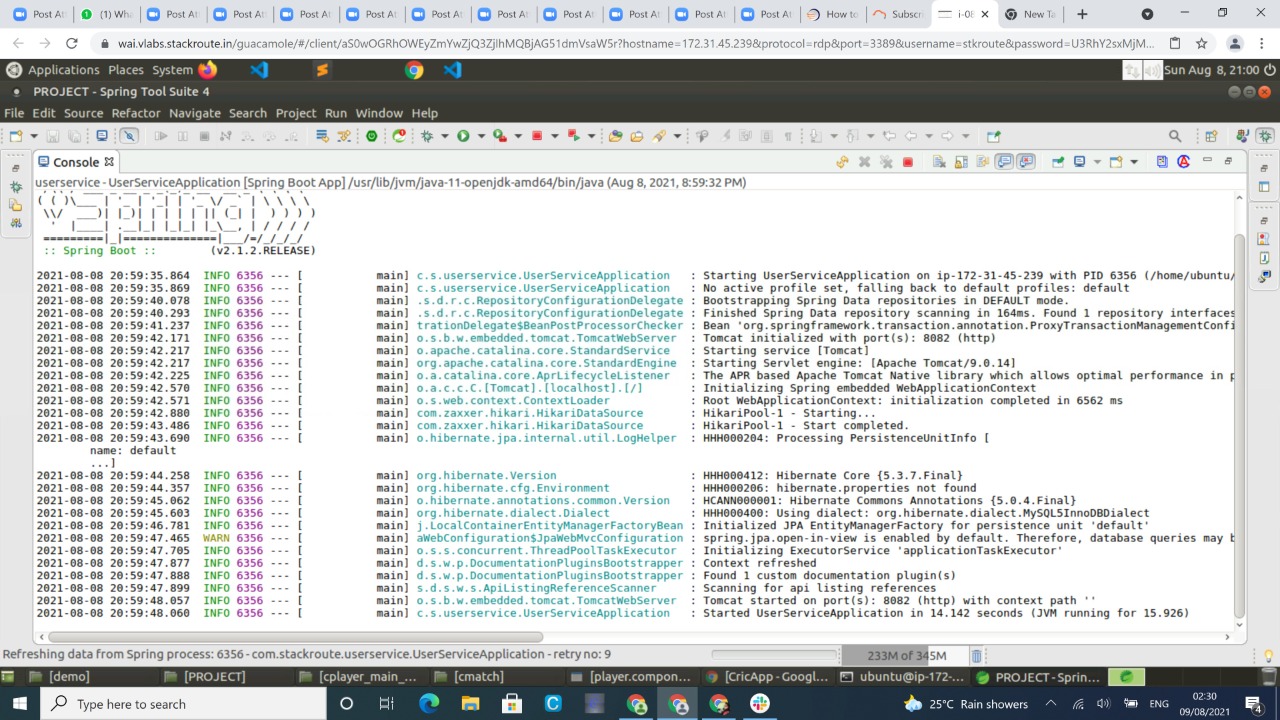
****

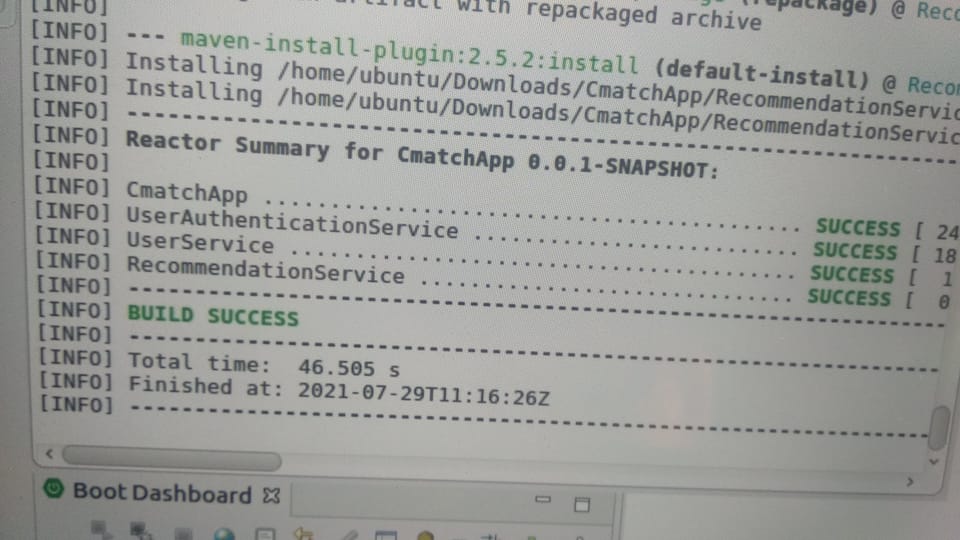
****

****

****

****

****

****

**CHAPTER SIX**

Scope and Limitation

5.1. Scope

In this system we are not able to obtain entire details of a cricket tournament .That means we can’t see the statistics of the entire tournament. Instead of that,system shows every match as individual matches and only gives that match statistics. So in future it will be more effective if we do something to include a way to get the entire statistics of a tournament. And also it will be better if we can add the point tables of tournaments.

5.2 Limitation

* The user didn’t get the entire statistics for a tournament.
* It can’t provide point tables as well as standings of each team in a tournament.
* The API using is an external API so if there isany error in their server it will affect our system too.

**CHAPTER SEVEN**

**Reference**

* [**https://www.javainuse.com/fullstack/ecommerce**](https://www.javainuse.com/fullstack/ecommerce)
* [**https://www.javatpoint.com/spring-boot-tutorial**](https://www.javatpoint.com/spring-boot-tutorial)
* [**https://www.w3schools.com/bootstrap/**](https://www.w3schools.com/bootstrap/)
* [**https://angular.io/guide/styleguide**](https://angular.io/guide/styleguide)
* [**https://www.edureka.co/blog/microservices-tutorial-with-example**](https://www.edureka.co/blog/microservices-tutorial-with-example)