**Cricket Analyzing Scoreboard**

**Team Software Development**

**Cardiff Metropolitan University**

**ICBT**

**CIS-7015**

**Project report compiled by**

|  |  |  |  |
| --- | --- | --- | --- |
| Author | Role | Date | Description |
| Himarsha Ekanayake | Developer/QA | February 2018 | control and reporting |
| Amila Uduwerella | Project Manager/System admin | February 2018 | control and reporting |
| Dilshan Perera | Business Analyst | February 2018 | control and reporting |

**1) Contributions to the project**

**Himarsha Ekanayake : 20126802**

Development of the system using PHP/MYSQL, Lead QA for the project.

**Amila Uduwerella : 20128479**

Project Manager for the Cricket Analyzing Scoreboard project. System administrator in charge of deployment and maintenance of services. Support QA for the project.

**Dilshan Perera : 20126805**

Business analyst in charge of bringing the requirements to the team. Support QA for the project to test and verify that the project delivers the right product.

**2) About the client**

ThePapare.com which is owned by Dialog Axiata PLC, the number 1 telco in Sri Lanka is a self appointed watchdog of Sri Lankan sports. Almost all local sporting events be it school, club, national or international are covered live, reported through their website and on social media or TV channel to all the sports enthusiasts in and around the country.

**3) Software process used for the project** - Waterfall methodology

**4) Why use waterfall?**

Waterfall software methodology is used for projects with requirements well known. It will be quite clear in section 6) below that the requirements are well known for the project Cricket Analyzing Scoreboard hence why we have decided to go with the said methodology.

**5) Existing system**

Main objective of the system was to provide live updates of school and club matches played. But the system was only capable of providing very limited data

of a cricket match which is being played at a time. For example, current score of the team, number of balls bowled, individual score of both players who are

playing now. These figures were manually entered to a file by a user and then it need to be saved in order to those data to be available to system.

Unfortunately these data were never being collected or analized.

There will be a file which has to be opened using a text editor (ex: microsoft notepad), File is created according to a pre-defined format.

So there is a location which we need to update above mentioned values; such as score, marks etc. then once the file is saved, data from

the text file will be fed to website, so that visitors to website can see over the internet.

**6) Requirements gathering**

Sources used for requirements gathering :

* Technical manager
* Operations manager
* Technical assistants

Techniques used to gather requirements

* Working in the business environment
* Casual brainstorming session with the Technical Manager, Operations Manager and Technical Assistants
* Prototyping of the system to the users

**7) Proposed system and features**

Our project team has compiled the below SRS which clearly indicates the requirements gathered during the requirements analysis stage, how they will be tackled with the system with it’s nifty features during the design and development phase of the project.

Some of the key features are listed below followed by the SRS presented to the client.

* *Score of playing match*
* *scores of both players who are batting at the moment*
* *can view graphs of both teams performance*
* *save match summary in to a database.*
* *by selecting a role model player, we can see who are the new players out there who has similar playing style etc.*

**SRS**

**Cricket Analyzing Scoreboard**

1.0 Introduction

1.1 Purpose

The purpose of this document is to provide a written explanation of the Cricket Analysing Scoreboard. An explanation of the systems purpose, features, functionalities, constraints under which it will operate and how it will react to external stimuli is what will be included in the system. Document is intended for all stakeholders of the system and the project to be approved by the client.

1.2 Scope of project

The project aims at implementing a platform to keep track of the score, analyze and produce meaningful stats on cricket matches which would make it easy to help identify potential players and report to viewers.

1.3 Glossary

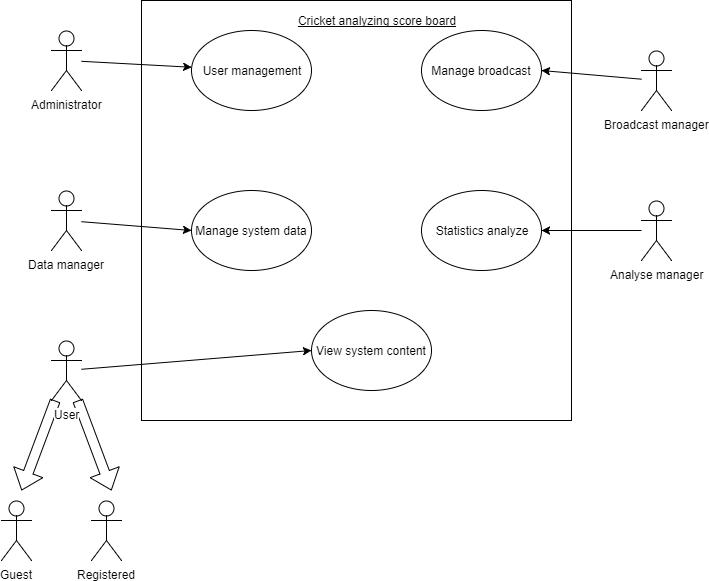
|  |  |
| --- | --- |
| Term | Definition |
| Database | Collection of all the data gathered by the system |
| Software Requirements Specification | A document which describes all functions of a proposed system and constraints |
| Stakeholder | A person who has an interest in the system |
| User | A person who operates the system |
|  |  |

1.4 Overview of document

The next chapter which is the Overall Description section provides an overview of the functionality of the system. It provides an explanation of the informal requirements of the system and is used to establish a context for the technical requirements specification in the following chapter.

2.0 Overall description

2.1 System environment



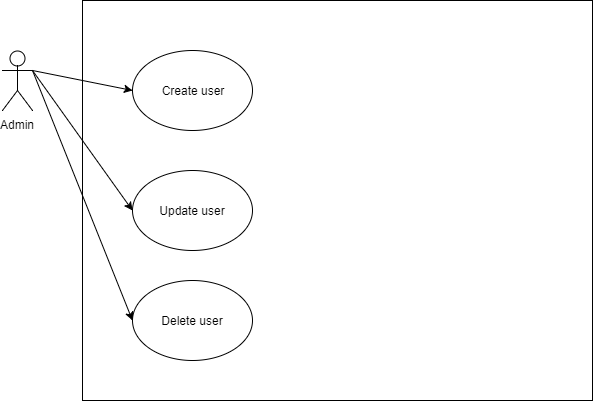
**Figure 1 :System environment**

Cricket analyzing scoreboard consists of 6 actors and one cooperating system. The web based system allows users, analyse manager, broadcast manager, data manager and administrator to interact with the system online.

2.2 Functional requirement specification

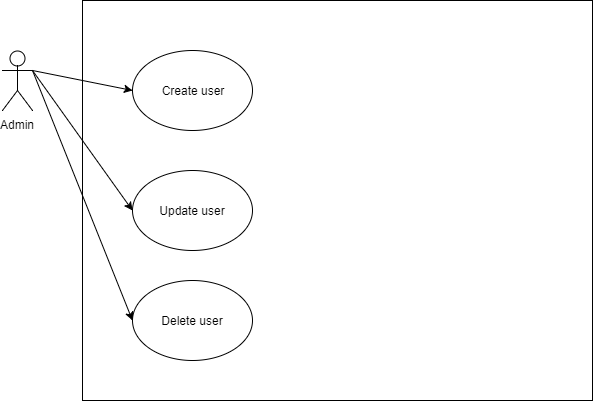
Use cases of the system are graphically represented and briefly described in this segment. Broadcast manager is the main actor in the system.

2.2.1 Administrator use case



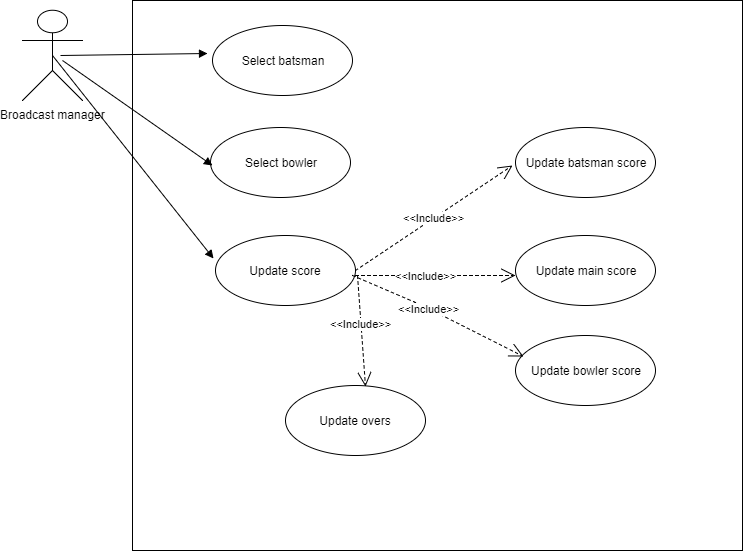
**Figure 2 : Admin use case**

2.2.2 Data manager use case



**Figure 3 : Data manager use case**

2.2.3 Broadcast manager use case



**Figure 4 : Broadcast manager use case**

2.3 User characteristics

Apart from the administrator whose task is to manage access of users to the system, everyone else interacting with the system is expected to have a fair knowledge in the game of cricket.

2.4 Non-functional requirements

System will be hosted in a AWS hosted cloud based VPC. This would contain the server which manages the web requests, load balancers, application servers and databases.

It is expected that there will be a high load on the system provided due to the fact that lot of users would be using the system to get constant updates on matches.

**8) Design**

Once we finalized the idea and list down all the requirements which includes both existing features and proposed features, We sketch User Interfaces. Then it helped to start developments. Once the development started we managed to complete the User interfaces.

User Interfaces

Login Screen



Depending on the user’s role type, options available to the user after loggin in to the system may differ.

Mainly there are five user types.

1. Broadcast Manager
2. Administrator
3. Data Manager
4. Analyse manager
5. User
   1. Registered
   2. Guest

**Broadcast Manager**

Mainly broadcast managers responsibility is to provide live updates of matches to the system. There are main three responsible updates required to be done by Broadcast Manager.

* Update Batsmen (Select batsman)
* Update bowler (Select bowler)
* Update Score
  + Update Individual batsman’s score
  + Update Individual Bowler’s score
  + Update main score of both teams
  + Update overs





**Administrator**

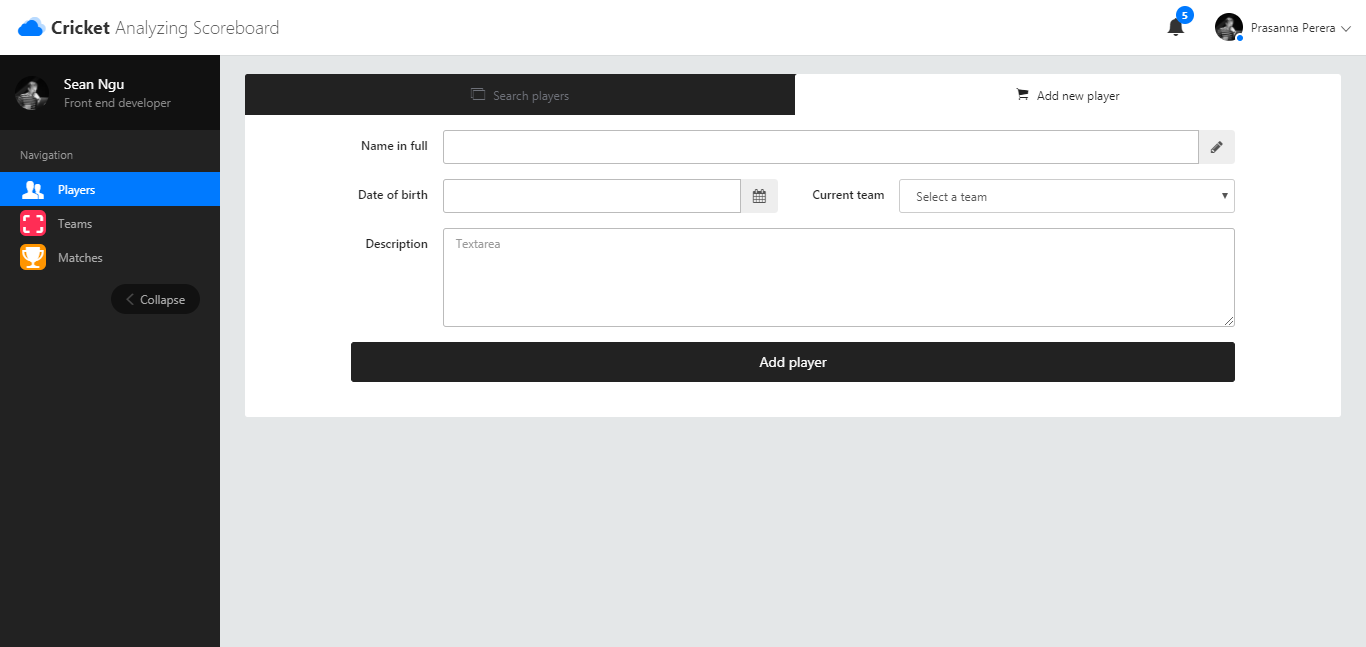
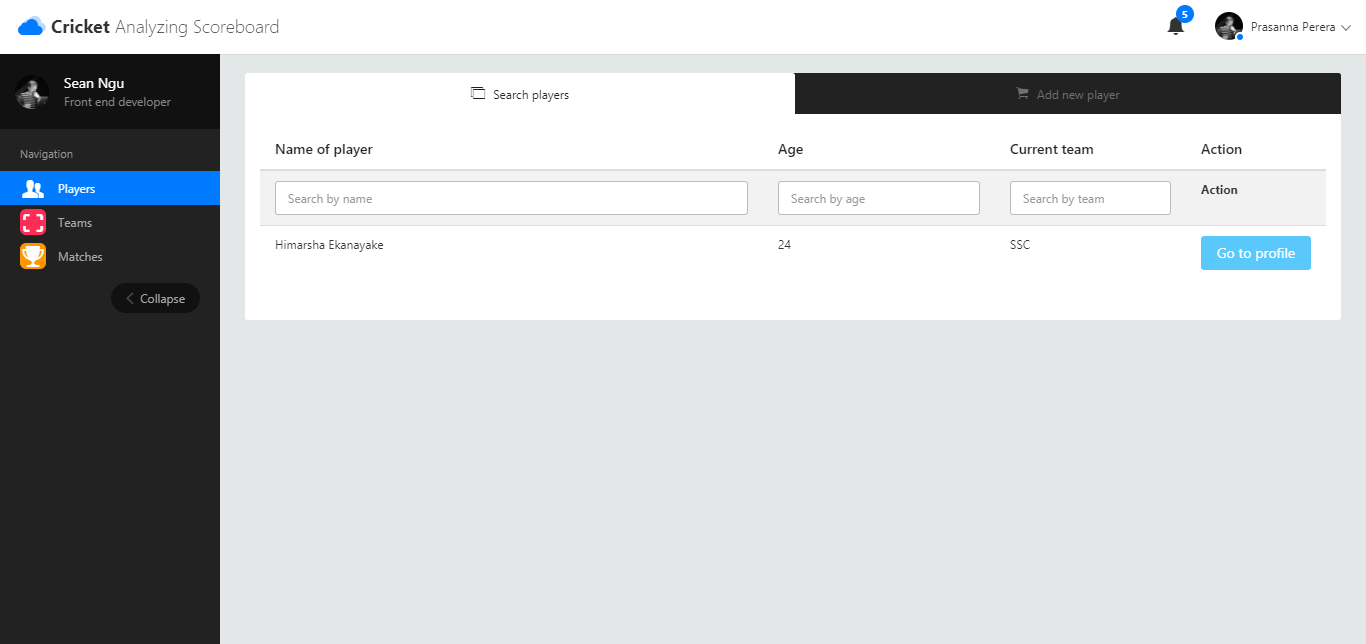
Administrator is the person who manages users of the system. Administrator has three responsibilities.

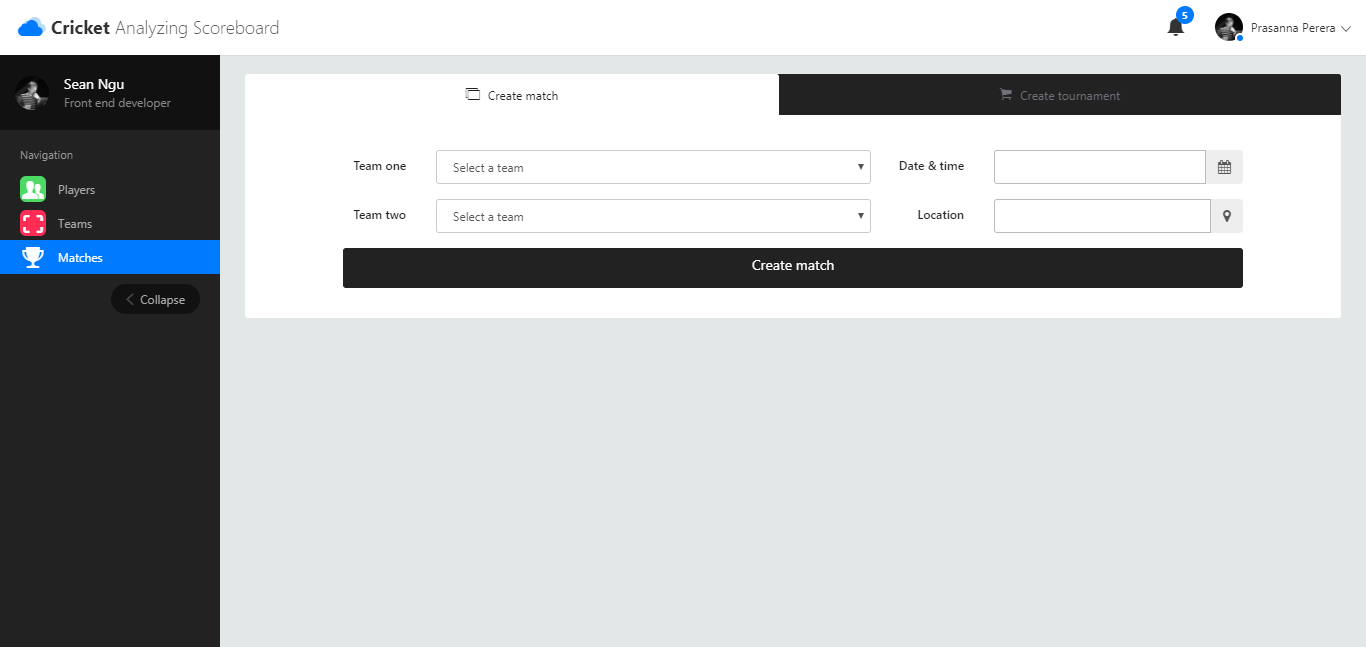
* Create Users
* Update Users
* Update Users

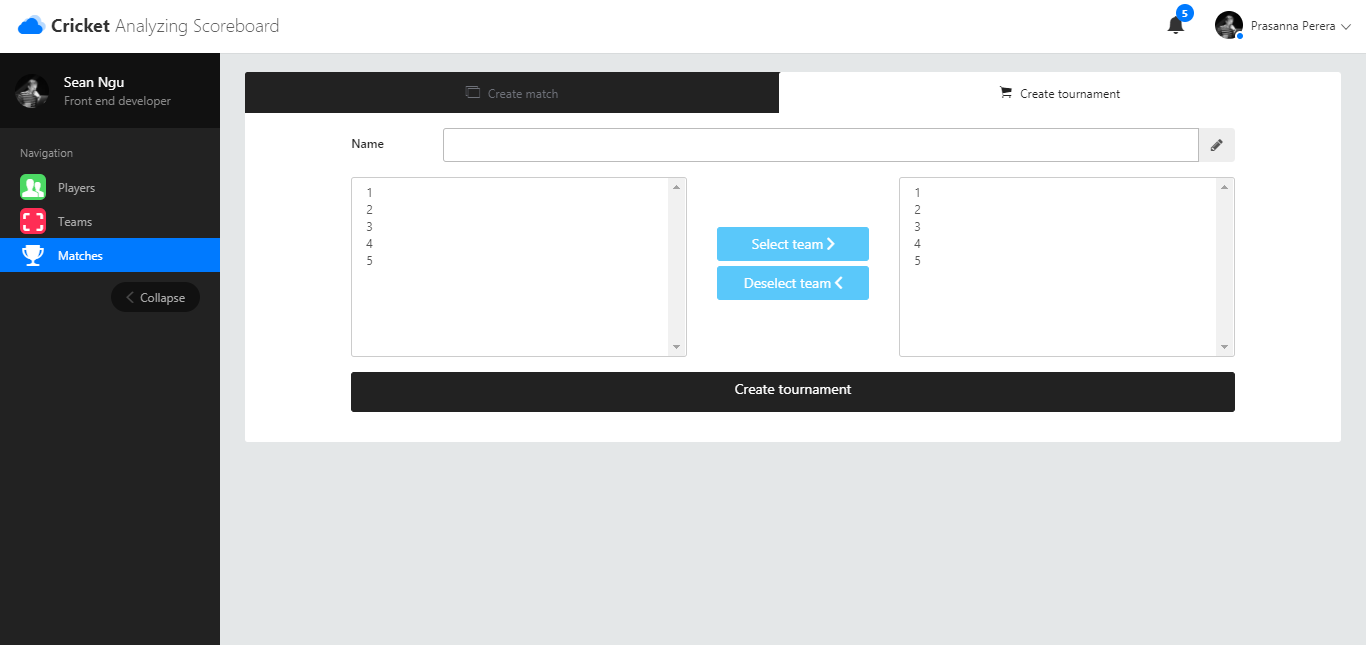
**Data manager**

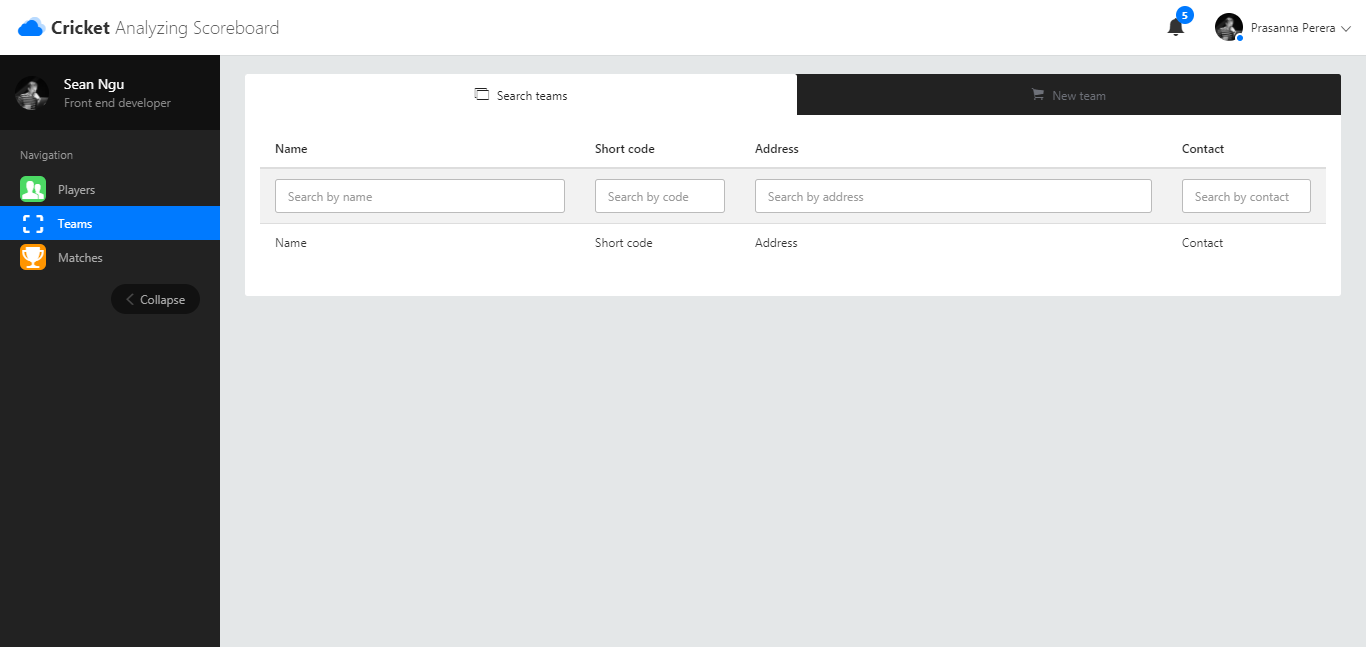
Person who create matches, select teams, team members, which includes managing run time data of the system are the main responsibilities of the datamanager.

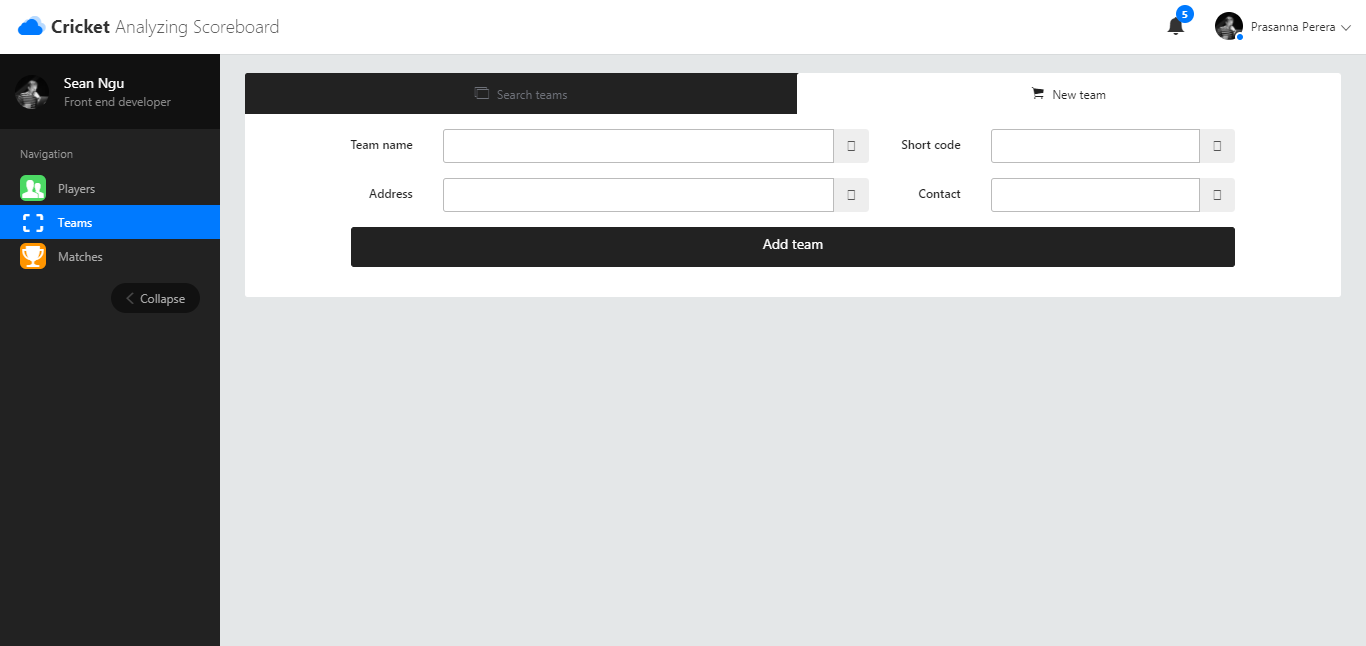
* Create teams
* Add players to teams
* Create matches/tournaments
* Select teams for matches/tournaments





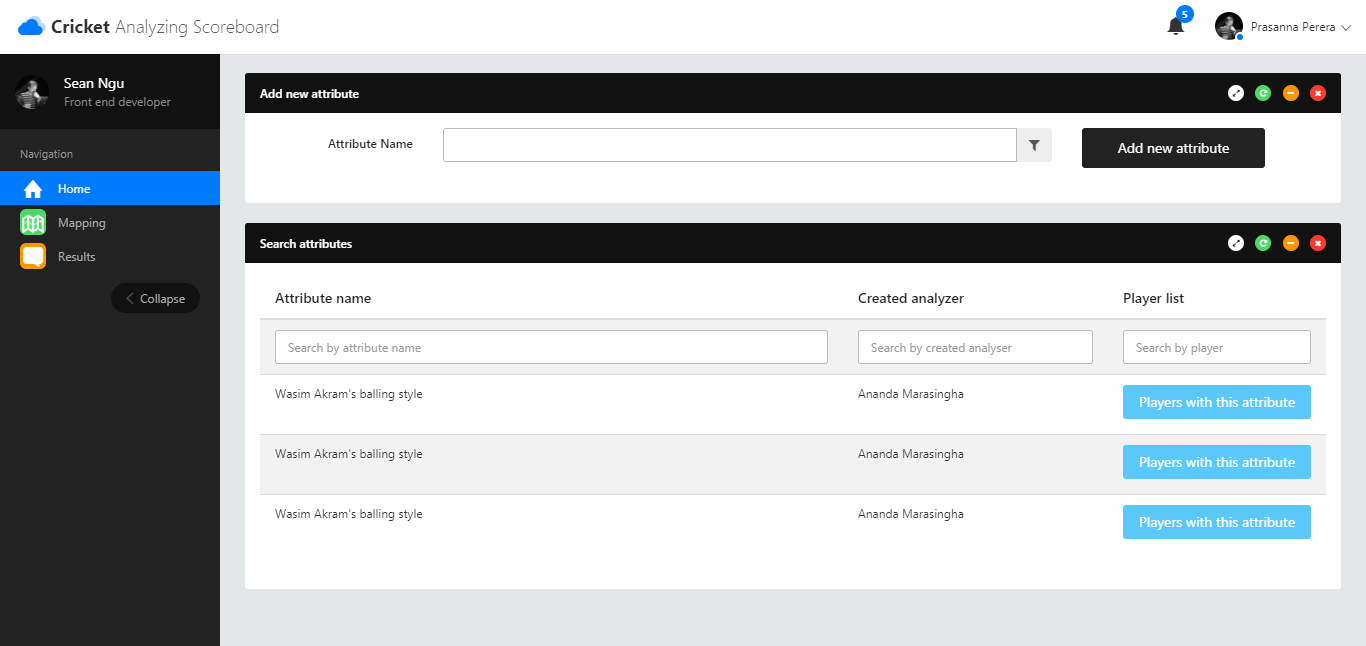


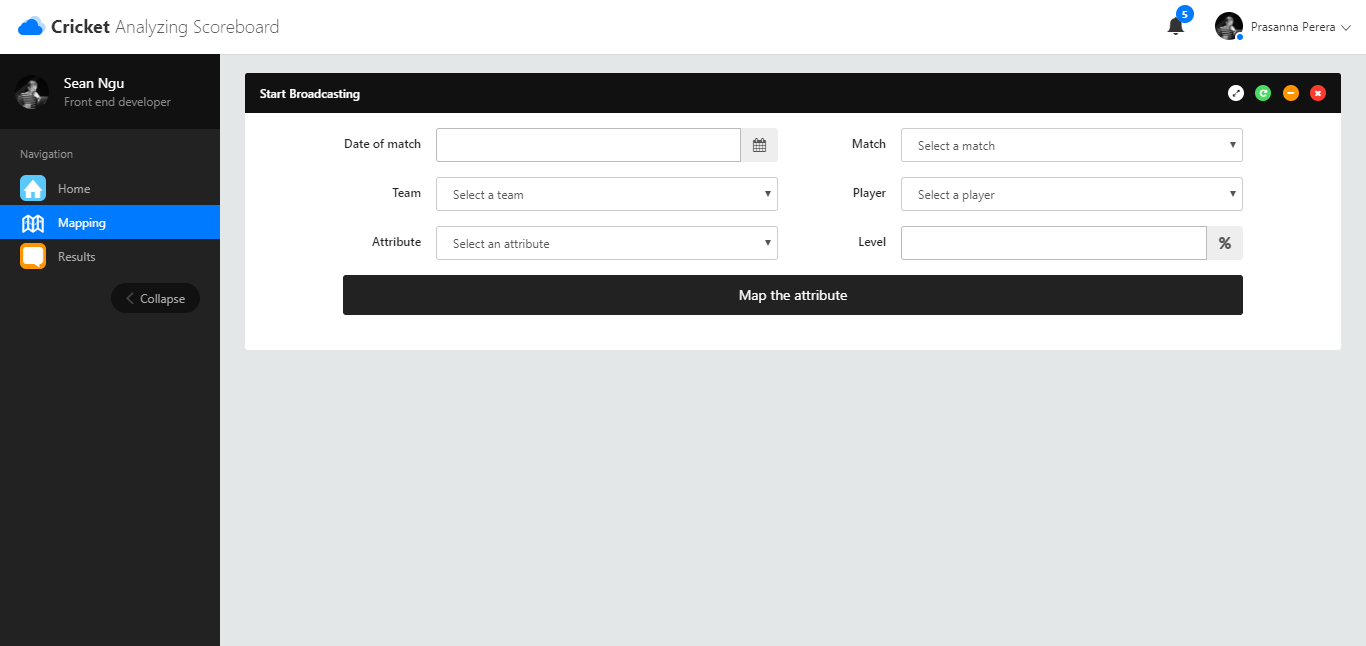


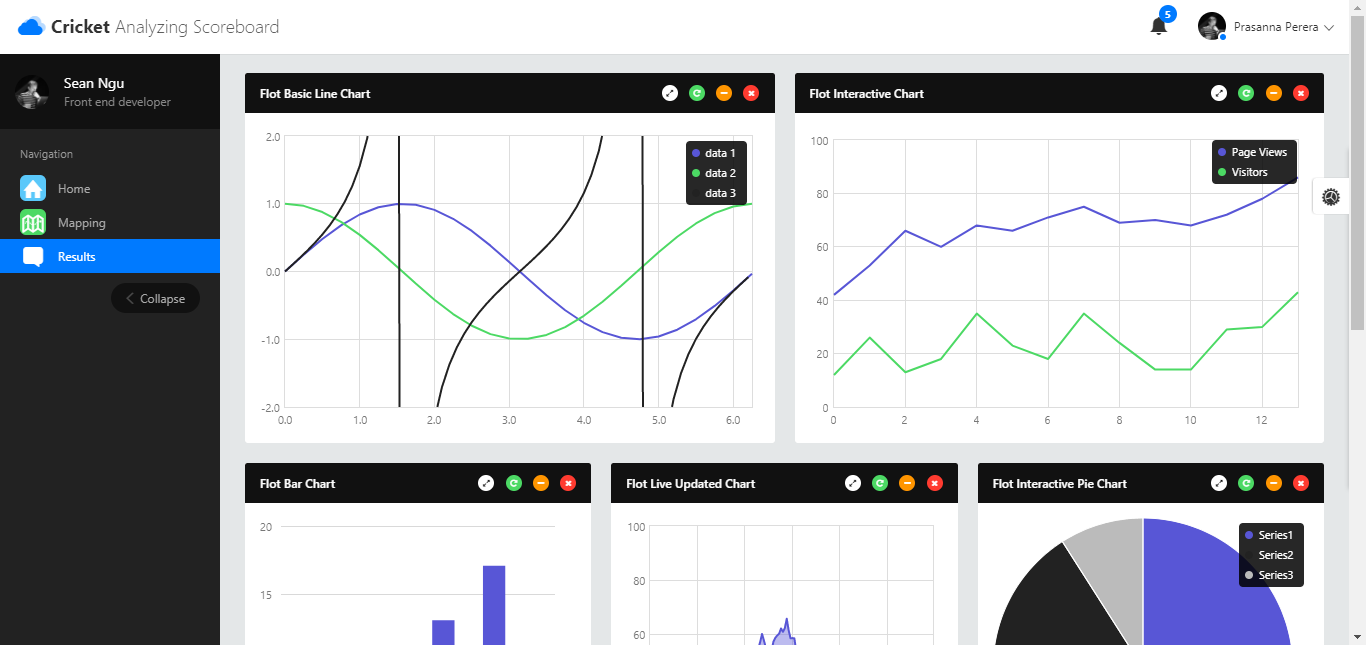


**Analyse manager**

Analyse manager is the person who makes comments about players, create role model players. Produce reports.







**Users**

Users are the people uses this system directly or indirectly. People who watch and get updates about matches through our system are considered as users here. Mainly there are two types of users.

* Registered users
* Guest users

Registered users has read only access to all data of the system, while guest users who are not registered will not have access to all data.

**9) Development**

Development was carried on phpstroam. And we used github as the code repository. We mainly have three branches in our repository;

1. Inprogress
2. UAT
3. Mater

All changes made to source code is on inprogress branch, code which is supposed to deployed to UAT environment is merged to UAT branch and then finally the Master branch contain code which is ready to go live, or already in live.

No development work is done on any branch except inprogress, if there is any development task need to be done, it first need to be done on inprogress, then do developer testing and then UAT branch and finally to master branch.

**10) Testing**

*Testing process started just after the requirement gathering was finished.*

*We had to verify and test whether the requirements that we captured from existing*

*system are correct and also the requirements and features of the proposed system*

*are accurate. So that we are confident about the product we are building.*

*- during the requirement gathering stage we analyzed requirements and verify each of them.*

*- review design and database table relationships*

*- testing each function while developments are still in progress*

|  |  |  |
| --- | --- | --- |
|  | Verification | Validation |
| 1 | Are we building it right? | Are we building the right thing? |
| 2 | Ensure we met all proposed functionalities | Ensure the build functionalities are the expectation |
| 3 | While the development process goes on we continue with the verification - Mainly by Developer | Once a predefined set of tasks are completed, we start the validation - mainly by others |
| 4 | We used the whitebox testing | We used blackbox testing |
| 5 | It has static activities, as it includes collecting reviews, walkthroughs, and inspections to verify a software. | It has dynamic activities, as it includes executing the software against the requirements. |
| 6 | It is an objective process and no subjective decision should be needed to verify a software. | It is a subjective process and involves subjective decisions on how well a software works. |

**11) Deployment**

In a world where even users are hesitant to install much applications at least on their mobile devices, the latest trend for companies or rather businesses is to produce web applications which can be accessed from any location at users convenience.

Hence we have decided to develop our product based on web. Due to the popular demand of Amazon Web Services which we call as AWS we have chosen as the host for our development since it offers many solutions ranging from computing facilities to storage etc.

Infrastructure utilized for the project :

* Linux VPC Hosted in AWS
* PHP 5.5.22
* MySQL Server 5.6.32

When it comes to developments, we kept our source codes on a github account. Developer was very responsible about the changes and he always kept the bit bucket account updated. In github we kept three branches of the repository.

1. Inprogress
2. UAT
3. Master

Inprogress

This is the branch that developer keeps updating his development code. Inprogress branch contains code with are not tested. Usually we do not deploy code in inprogress to any environment except the development.

UAT

Once the developer has tested his code on development environment and he is confident about it, he merge the inprogress branch to UAT branch. Then we have automated the deployment process in UAT environment, where every hour there will be a release don to UAT environment from the UAT branch of the github. But that happens only of the code on UAT branch has changed. Then Project Manager start testing on UAT environment, when required Both Dev and BA helps the PM. If there is any changes required to UAT, developer will continue with the change and merge the dev in to UAT branch again.

Master

Master branch is where the ultimate ready to go code is located. When the UAT environment is OK , developer merge it with master. Similar to UAT environment we have automated the production deployments as well. Once the change to master branch appears, it will be deployed to p Production ( pre-production) environment automatically. Then Business Analyst start testing the pre-production environment.

Once the system goes live, we stop the automated deployments from the system, rather than deploying code from master branch, system will required to provide a tag created on the master branch, then there will be a person who manages these deployments will execute the deployment, where the script will grab code from the release tag created and pushed to the production server. Incase we required a roll back, we just need to do another deployment using the release tag of the last deployment.

**12) References**

[**https://en.wikipedia.org/wiki/Systems\_design**](https://en.wikipedia.org/wiki/Systems_design)

[**https://en.wikipedia.org/wiki/View\_model**](https://en.wikipedia.org/wiki/View_model)