**MAKERERE UNIVERSITY**

COLLEGE OF COMPUTING AND INFORMATION SCIENCES

DEPARTMENT OF NETWORKS

BACHELOR OF SCIENCE IN SOFTWARE ENGINEERING (YEAR 2)

RECESS TERM 2 (BSE 2301)

SDS FOR:

ANALYSIS ON FIFA 18 DATASET

**PROJECT MEMBERS (GROUP 10)**

|  |  |  |  |
| --- | --- | --- | --- |
| **NAME** | **REG NO** | **STUDENT NO** | **SIGNATURE** |
| SUNDAY DEOGRATIAS | 16/U/11792/PS | 216002567 |  |
| MUWONGE EMMANUEL | 16/U/7842/PS | 216003360 |  |
| WAMBOLI JOSEPHAT | 16/U/12290/EVE | 216004076 |  |
| AJAMBO CATHERINE | 16/U/19172/PS | 216021557 |  |

**PROJECT LEADER**

MUWONGE EMMANUEL

**SUPERVISOR:** MBABAZI ISAAC

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

SOFTWARE ENGINEERING RECESS PROJECT BSE 2301

6TH JULY, 2018

Table of Contents

[**1. Introduction** 4](#_Toc518397672)

[**1.1. Document Outline** 4](#_Toc518397673)

[**1.2. Document Description** 4](#_Toc518397674)

[**1.2.1. Introduction** 4](#_Toc518397675)

[**1.2.2. System Overview** 5](#_Toc518397676)

[**2. Design Considerations** 6](#_Toc518397677)

[**2.1. Assumptions and Dependencies** 6](#_Toc518397678)

[**2.2.General Constraints** 7](#_Toc518397679)

[**2.3. Goals and Guidelines** 7](#_Toc518397680)

[**2.4. Development Methods** 7](#_Toc518397681)

[**3. Architectural Strategies** 7](#_Toc518397682)

[**Architectural Style** 7](#_Toc518397683)

[**Web Features** 8](#_Toc518397684)

[**Security** 8](#_Toc518397685)

[**Data Storage** 8](#_Toc518397686)

[**Future Enhancements** 8](#_Toc518397687)

[**4. System Architecture** 8](#_Toc518397688)

[**4.1 Subsystem Architecture** 9](#_Toc518397689)

[**5. Policies and Tactics** 9](#_Toc518397690)

[**6. Detailed System Design** 9](#_Toc518397691)

[**6.1. Dataset component.** 9](#_Toc518397692)

[**6.1.1. Definition** 9](#_Toc518397693)

[**6.1.2. Responsibilities** 9](#_Toc518397694)

[**6.1.3. Constraints** 10](#_Toc518397695)

[**6.1.4. Composition** 10](#_Toc518397696)

[**6.1.5. Uses/Interactions** 10](#_Toc518397697)

[**6.1.6. Resources** 10](#_Toc518397698)

[**6.1.7. Processing** 10](#_Toc518397699)

[**6.2. Analysis component** 10](#_Toc518397700)

[**6.2.1. Definition** 10](#_Toc518397701)

[**6.2.2. Responsibilities** 10](#_Toc518397702)

[**6.2.3. Constraints** 10](#_Toc518397703)

[**6.2.4. Composition** 10](#_Toc518397704)

[**6.2.5. Uses/Interactions** 10](#_Toc518397705)

[**6.2.6. Resources** 11](#_Toc518397706)

[**6.2.7. Processing** 11](#_Toc518397707)

[**6.2.8. Interfaces/Exports** 11](#_Toc518397708)

[**6.3. Analysis Display component** 11](#_Toc518397709)

[**6.3.1. Definition** 11](#_Toc518397710)

[**6.3.2. Responsibilities** 11](#_Toc518397711)

[**6.3.3. Constraints** 11](#_Toc518397712)

[**6.3.4. Composition** 11](#_Toc518397713)

[**6.3.5. Uses/ Interactions** 11](#_Toc518397714)

[**6.3.6. Resources** 11](#_Toc518397715)

[**6.3.7. Processing** 12](#_Toc518397716)

[**6.3.8. Interface** 12](#_Toc518397717)

[7. References 16](#_Toc518397718)

# 

# **1. Introduction**

## **1.1. Document Outline**

The Software Design Specification is divided into 10 sections with various subsections. The sections of the Software Design Specification are:

* Introduction
* System Overview
* Design Considerations
  + Assumptions and Dependencies
  + General Constraints
  + Goals and Guidelines
  + Development Methods
* Architectural Strategies
  + Architectural style
  + Web features
  + Security
  + Data storage
  + Future Enhancement
* System Architecture
* Policies and Tactics
* Detailed System Design
* Glossary
* Bibliography

## **1.2. Document Description**

Here is the description of the contents (by section and subsection) of the software design specifications:

### **1.2.1. Introduction**

The Software Design Specification is a document to provide documentation which will be used to aid in software development by providing the details of how the system/software should be built. Within the Software Design Specification are narrative and graphical documentation of the software design for the project including use case models, sequence diagrams, collaboration models, object behavior models, and other supporting requirement information [1].

#### **Purpose**

This document's purpose is to provide a high-level design framework around which to build our project FIFA 18 datasets analysis system [2]. This document will define the design of the FIFA 18 analysis system. It contains specific information about the expected input, output, classes, and functions. It also provides a list of requirements against which to test the final project and determine whether we were able to successfully implement the system according to design.

#### **Scope**

This Design Specification is to be used by Software Engineering and Software Quality Engineering as a definition of the design to be used to implement the FIFA 18 analysis system [3].

It provides the architecture and design of Release 1.0 of the FIFA 18 analysis system. It will show how the design will accomplish the functional and non-functional requirements detailed in the FIFA 18 analysis Software Requirements Specification (SRS) document.

#### **Intended Audience**

This document is written on a technical level to address the technical department of the users that will continue using the system after development [4]. In contrast to the Software Requirements Specification (that is written for the client and user), most of this Software Design Specification is written for knowledgeable software professionals and designers. Thus the Client will not be within the intended audience for this document.

#### **Definitions, Acronyms, and Abbreviations**

GUI - Graphical User Interface

SOA- System Oriented Architecture

PC - Personal Computer (desktop or laptop)

PDF– Portable Document Format [5].

### **1.2.2. System Overview**

**Product Perspective**

This product analyses, visualizes and predicts certain features in the FIFA 18 datasets [6]. It has some regression models that helps in determining some values in the FIFA 18 datasets.

#### **Design Method**

The design of this product utilizes an object-oriented approach.

#### **User Interfaces**

The user of this analysis product will be interfacing with the system to help put in some inputs when required. The product allows the user to get familiar with the software with the least computer knowledge.

#### **Hardware Interfaces**

This analysis software can run on most computer with compatible graphics card which is required due to need of graphical visualization of the analysis.

#### **Software Interfaces**

This system will execute on all operating systems platform running R and R Studio IDE.

#### **Memory Constraints**

This program takes up about 7 kb of memory. The output results are modest in size and take up about 7 kb.

#### **Operations**

The user will be required to enter the parameters/values for the analysis run from the user interface.

#### **Site Adaptation Requirements**

This software is intended to execute on any operating system platform with no modifications needed to support different sites.

#### **User Characteristics**

The general characteristics of the intended users are computer literates and people with some knowledge about football and preferably the EA FIFA games.

# **2. Design Considerations**

This section describes many of the issues which need to be addressed or resolved before attempting to devise a complete design solution.

## **2.1. Assumptions and Dependencies**

* This analysis system is only based on only FIFA 18 datasets i.e. for only a single year with no previous data.
* As an analysis based application, the analysis system assumes that the client and server machines running the software program will have R version 3.5.0 or later and R studio version 1.1.453 or later.
* The FIFA 18 analysis system will have GUI for users to perform their proposed analysis.

## **2.2.General Constraints**

This application can only run on a system that supports R and R studio IDE.

Requires data to run the system via web over http to transfer responses between the program and the external system.

Needs at least 2GB RAM computer.

## **2.3. Goals and Guidelines**

Other than network failures, no internal software failures are accepted.

Support a SOA using industry accepted standards and specification.

Where possible use known design patterns and employ software best practices.

Emphasis shall be placed on Usability as the User Interface will be used by users without much training.

The design must reflect the quality of Modifiability as the customer must be able to adapt it for various uses by the final end users.

The system must be fully functional, tested and deployable within the scheduled time frame.

The system must be able to be modified by the user to display the target analysis in various reports for various purposes.

## **2.4. Development Methods**

The incremental software development process will be used for the FIFA 18 analysis system architecture development. An incremental software development process is necessary to incrementally produce a functionality by the deadline time hence the FIFA 18 analysis system will follow this paradigm.

# **3. Architectural Strategies**

Given that FIFA 18 analysis application shall integrate information from the different columns of FIFA 18 datasets and perform systematic correlation of present time information. The FIFA 18 analysis system architecture respects the following architectural principles and technologies.

## **Architectural Style**

The architectural style for FIFA 18 analysis system is SOA. A SOA is an architecture principle that is based on the key concept of services. A service, in its simplest form, consists of an interface and an implementation. SOA defines software applications in terms of discrete services, which are implemented using service components that can be used to perform analysis activities for a given analysis process. SOA is not tied to a specific technology and can be implemented using a wide range of interoperability standards. The FIFA 18 analysis project will use the web services to implement this architecture.

## **Web Features**

Shiny [7]: Shiny is an R package that makes it easy to build interactive web apps straight from R. Shiny combines the computational power of R with the interactivity of the modern web.

## **Security**

The software requirement as written for this release will require login for the analysis and logout later after analysis is conducted.

## **Data Storage**

The software requirement for the Analysis do not warrant the use of a database for data storage management.

## **Future Enhancements**

The Architecture is designed to accommodate enhancements for future development. The enhancements are documented to ensure the ideas were not lost and could be considered for future implementation efforts.

# **4. System Architecture**

The system consists of three components; The Dataset, Analysis component and Display. The Analysis component depends on the Dataset component and the Display component depends on the Analysis component.

Display

Data imported

Analysis displayed

Fig 4.1. System architecture showing how components interact.

Analysis

Dataset

This above architecture encapsulates functionalities and allows easy system debugging.

## **4.1 Subsystem Architecture**

Dataset component: This is where the FIFA 18 dataset and other newly created datasets are stored. It takes up storage space since physical data is stored here.

Analysis component: Here the datasets from the above components are imported for analysis. The analysis is conducted by using the R Studio, the powerful and popular data analysis software. The analysis results are sent to the following components.

Display component: This component displays analysis results. The analysis results are in form of statistical figures and visual charts. An interactive end user interface is provided to aid users navigate through the analysis results.

# **5. Policies and Tactics**

1. End users can only interact with an imported copy of the dataset but not the original one. This is to avoid alteration of data which can interfere with the normal working of the system.
2. The user interface is interactive to provide a user friendly and detailed method of interaction with the system.
3. Each type of analysis is coded as a different module. This is to maximize cohesion and minimize coupling which comes with too much dependencies. This aids faster requirements traceability.
4. Use of R Studio to perform analysis since it’s easy to integrate with other systems.

# **6. Detailed System Design**

## **6.1. Dataset component.**

### **6.1.1. Definition**

This component contains the dataset(s) to be analyzed. For this system the dataset is the FIFA 18 data and other created datasets during development.

### **6.1.2. Responsibilities**

Store the FIFA 18 dataset and other dataset that may be created during the course of development. It acts as a feeder to the Analysis component.

### **6.1.3. Constraints**

* Requires physical storage space.
* End users can’t interact with this component. This is to avoid them altering the content of the dataset

### **6.1.4. Composition**

It contains FIFA 18 dataset and other datasets created by developers.

### **6.1.5. Uses/Interactions**

It interacts with the Analysis component. The latter imports the dataset from here.

### **6.1.6. Resources**

It uses the physical storage (The hard disk).

### **6.1.7. Processing**

The datasets are imported from it using scripts written in the analysis component.

## **6.2. Analysis component**

### **6.2.1. Definition**

This component conducts analysis on the dataset.

### **6.2.2. Responsibilities**

* Import dataset from where they are stored.
* Create new datasets from existing ones.
* Conduct analysis on imported datasets.

### **6.2.3. Constraints**

* Only accessible to developers.
* Requires availability of reasonable space on Main Memory.

### **6.2.4. Composition**

It has the R Studio and related libraries for data analysis.

### **6.2.5. Uses/Interactions**

It imports data from the Dataset component and displays analysis results in the Analysis component. It is also used by developers when working on the system.

### **6.2.6. Resources**

* Hard disk storage
* RAM
* Processor

### **6.2.7. Processing**

Codes (R Scripts) written in R studio are executed to perform required analyses.

### **6.2.8. Interfaces/Exports**

The main interfaces here are the script window and the console found in R Studio. The exported item from this component is Analysis results to the Display component.

## **6.3. Analysis Display component**

### **6.3.1. Definition**

A component that displays detailed results from analysis conducted.

### **6.3.2. Responsibilities**

* Display analysis results.
* Provide interface for end user interaction with the system.

### **6.3.3. Constraints**

* Requires a browser
* Changes made here shouldn’t affect the system.

### **6.3.4. Composition**

Contains exported graphs and statistical figures. It can also consist of a browser.

### **6.3.5. Uses/ Interactions**

This component interacts with the Analysis component by getting analysis results from the latter.

### **6.3.6. Resources**

* Hard disk
* RAM
* Processor

### **6.3.7. Processing**

Here analysis results are graphically displayed on the monitor for the users to view. Stimuli generated by user clicks are transferred to the Analysis component for response generation.

### **6.3.8. Interface**

A graphical user interface with clickable buttons and menus are provided to facilitate user friendliness and interactions with the end users.

Fig 6:1 showing log in page.

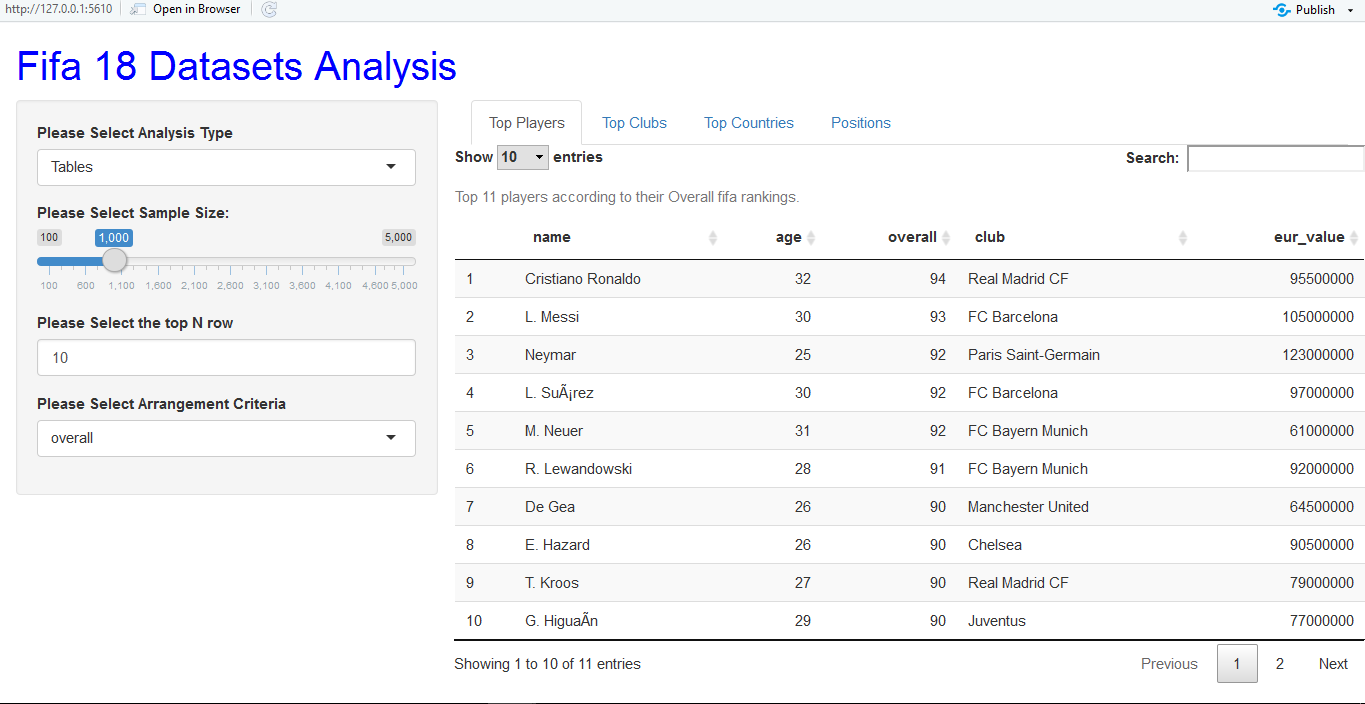
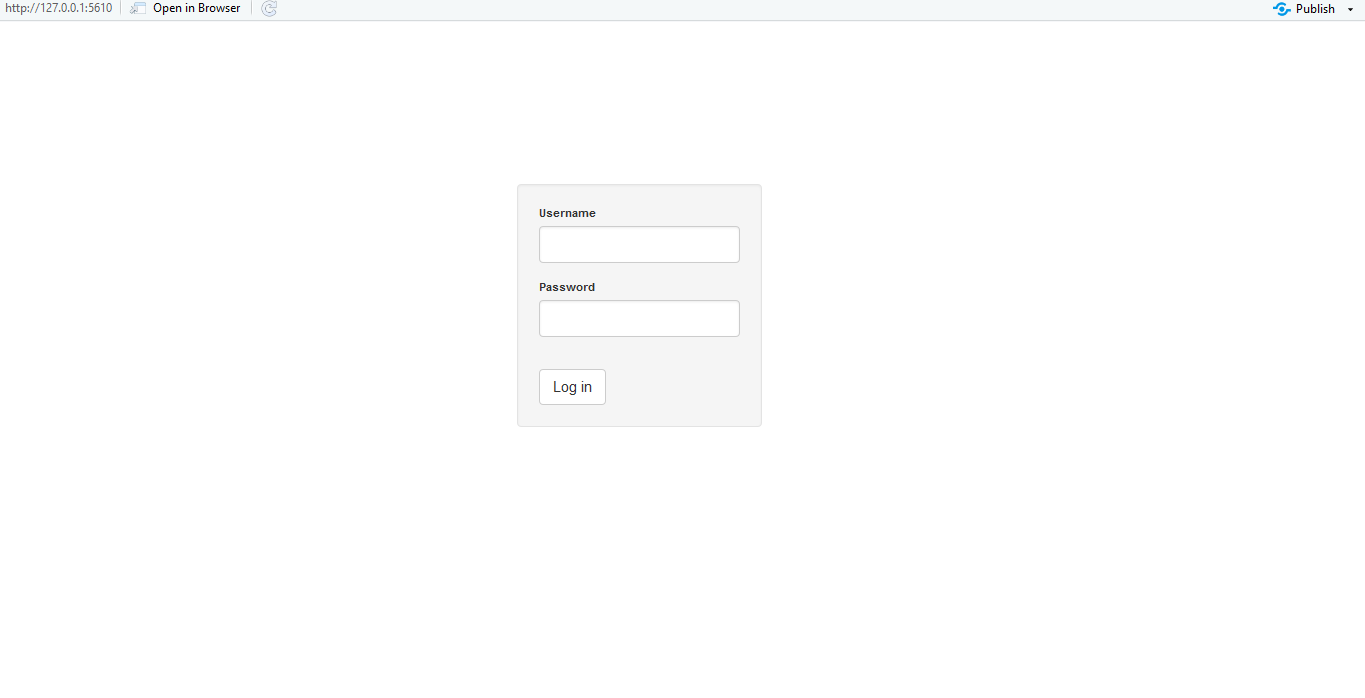


Fig 6:2 showing the Home Page which loads with the top players Table.

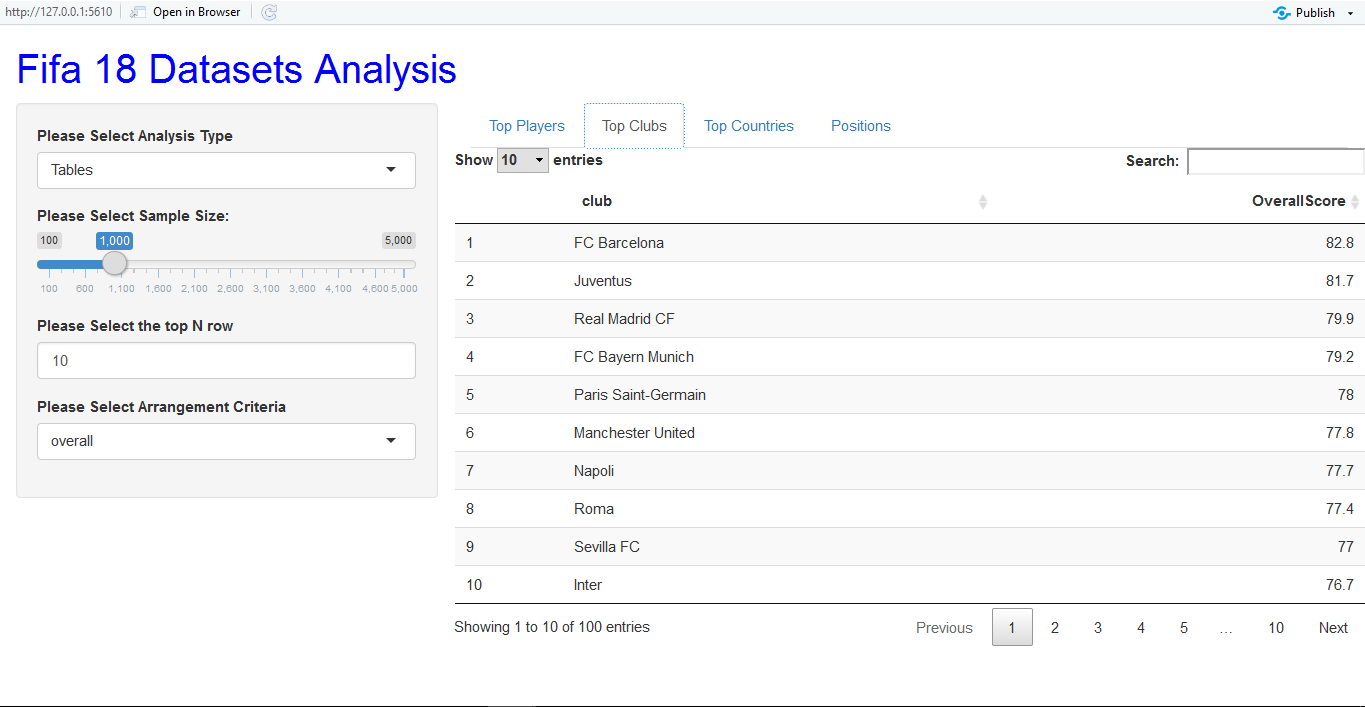


Fig 6:3 showing a table for Top Clubs based on mean overall score.

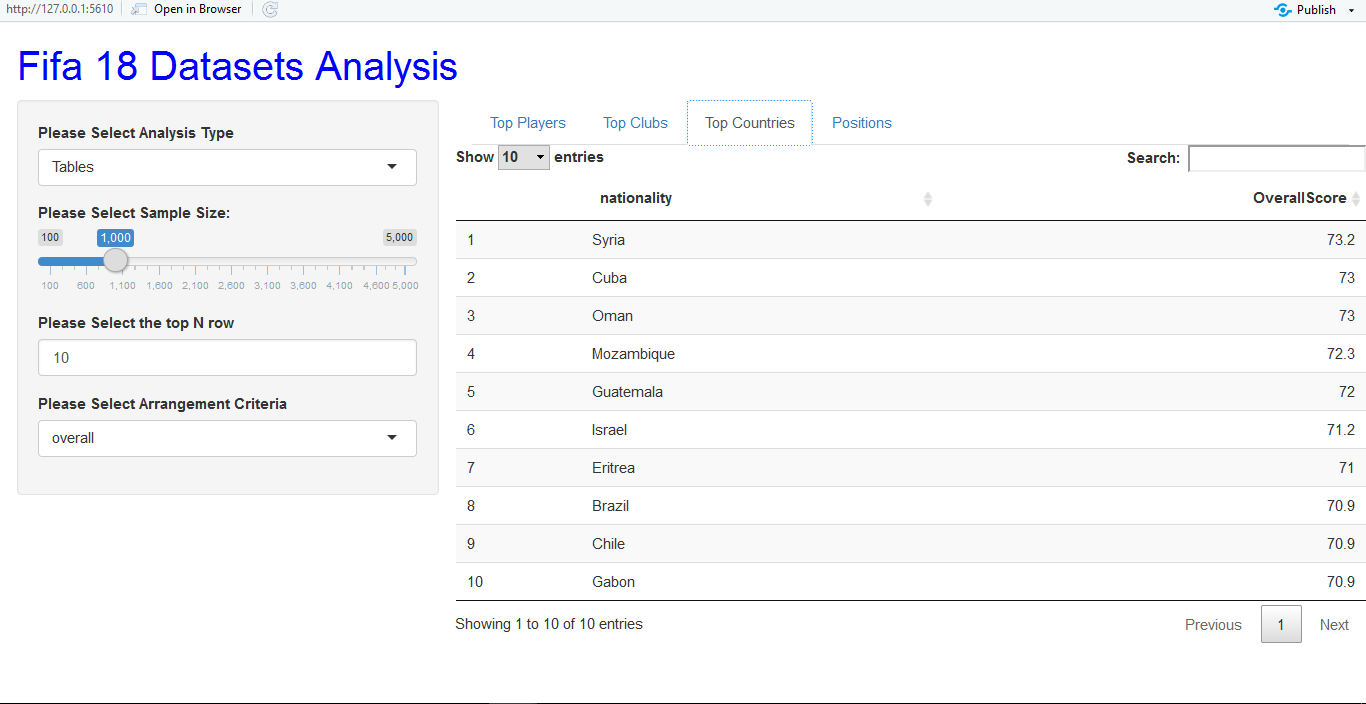


Fig 6:4 showing Top countries based on average overall score.

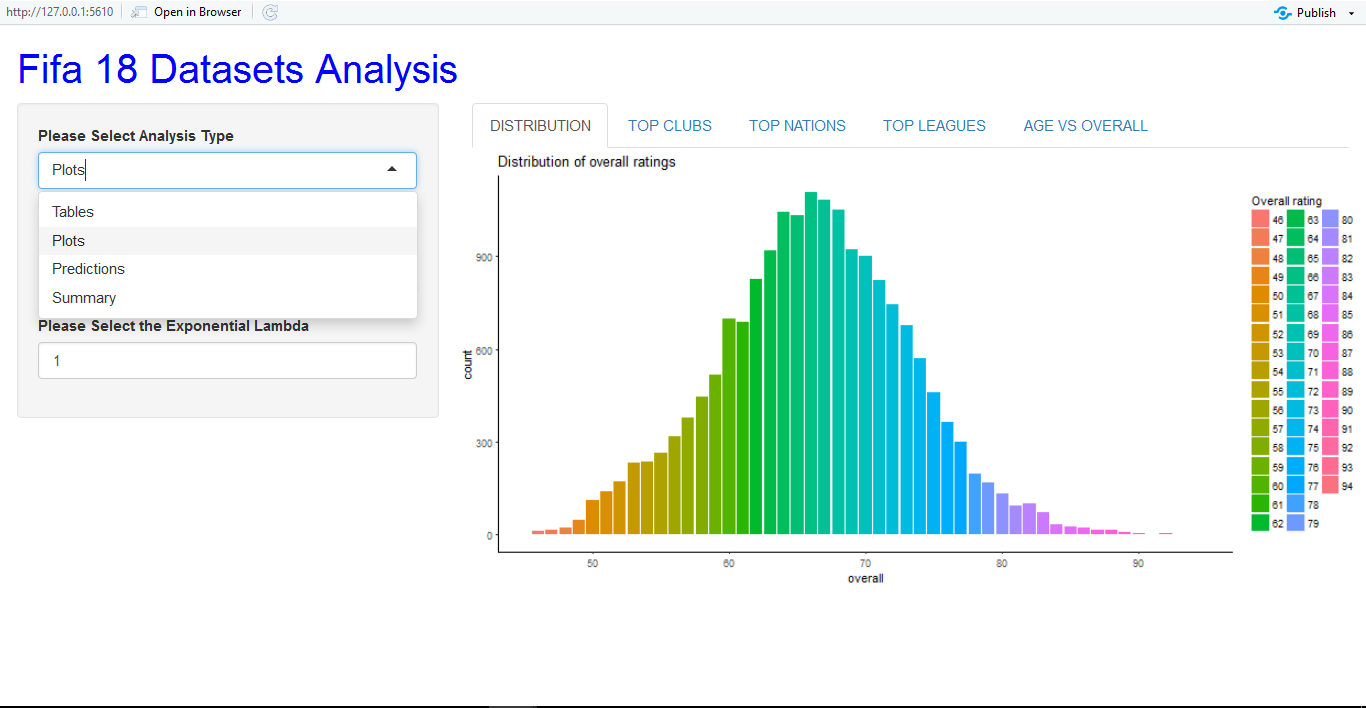


Fig 6:5 showing plot of overall score distribution

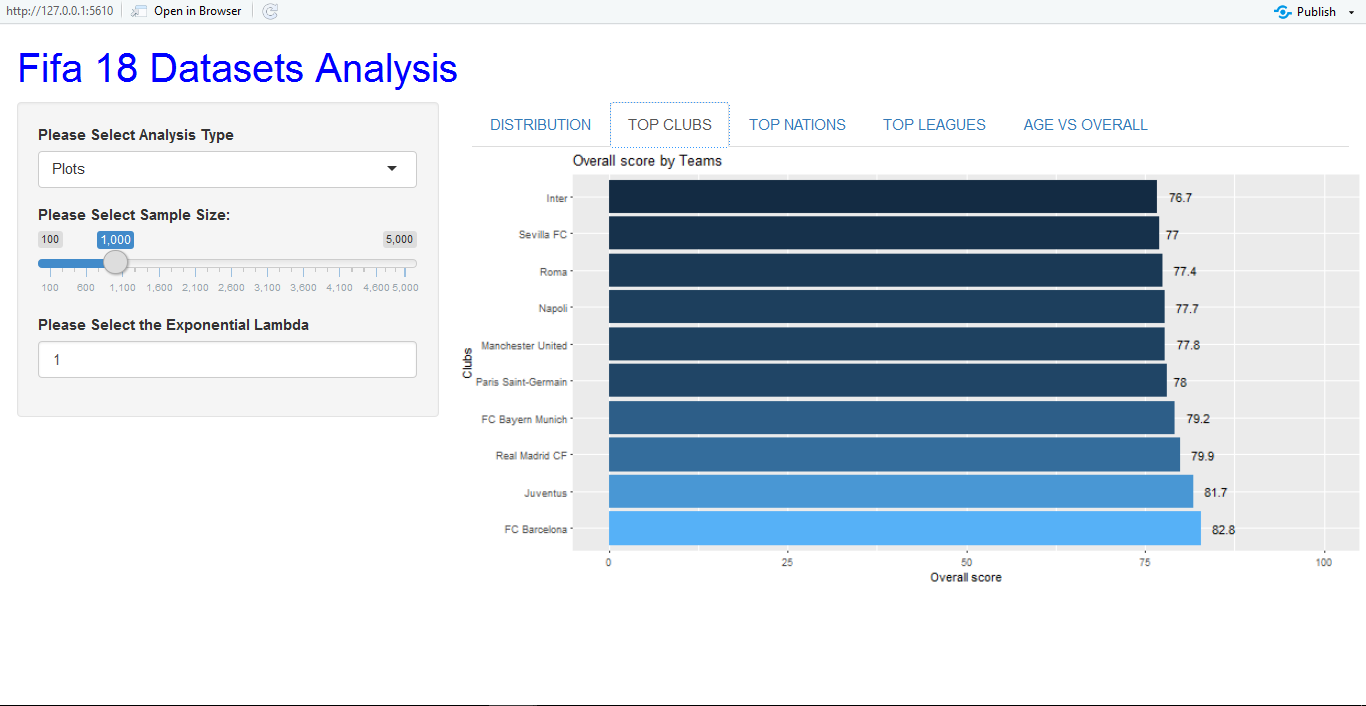


Fig 6:6 showing a plot of top clubs based on overall score.

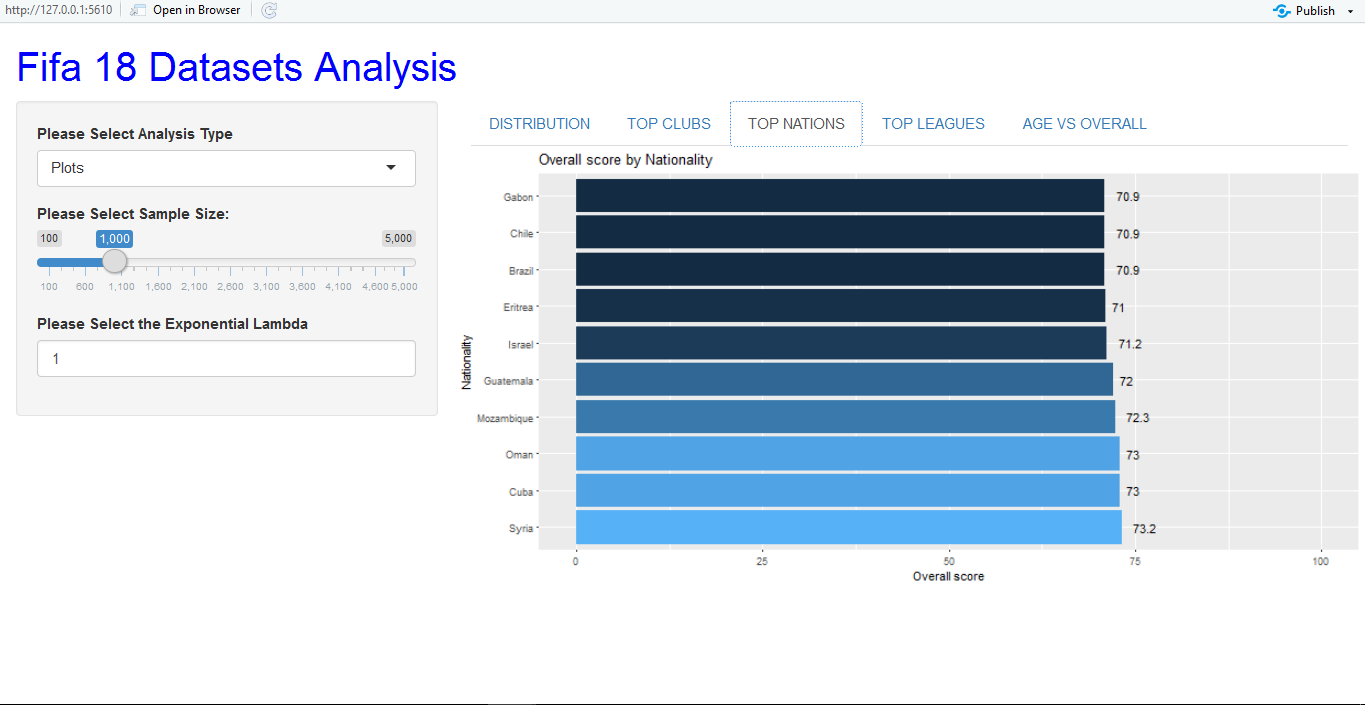


Fig 6:7 showing a plot of top nations based on mean overall score.

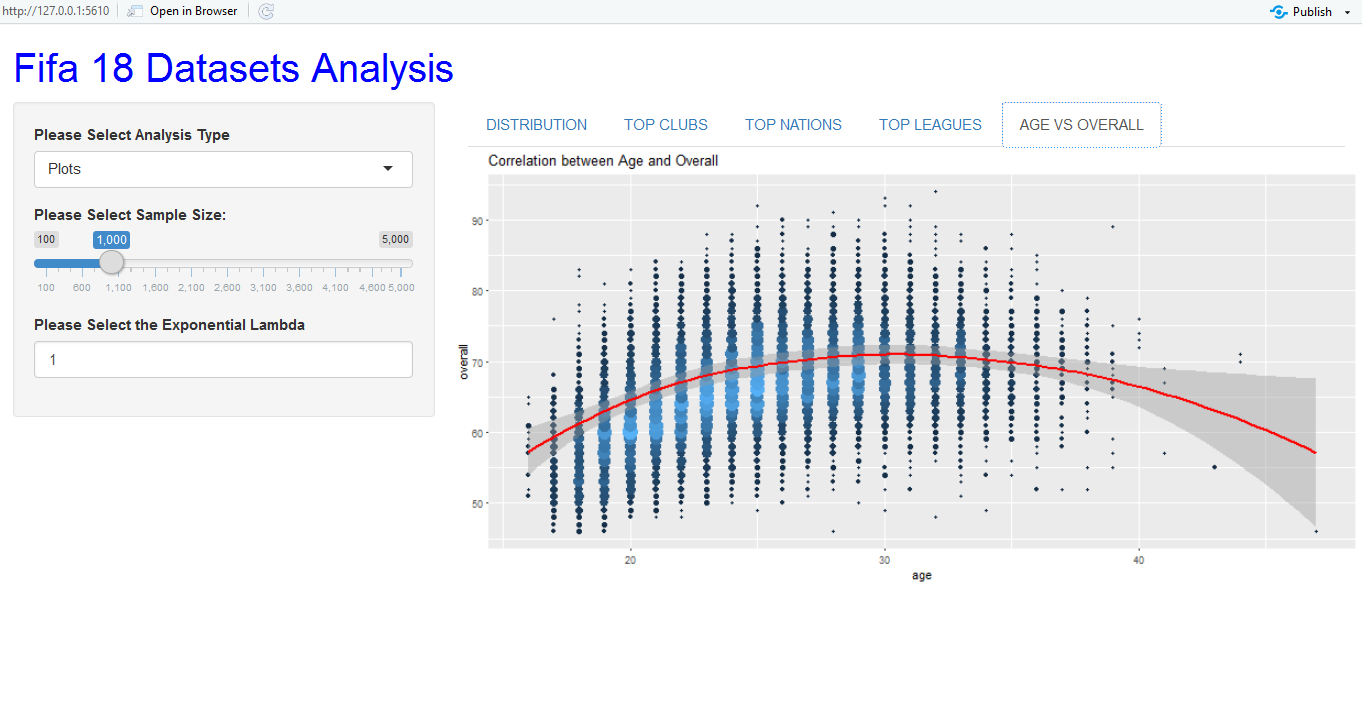


Fig 6:8 showing a plot of age distribution vs Overall score.

# 7. References

1. <https://www.robotics.ee.uwa.edu.au/courses/design/examples/example_design.pdf>

2. <https://www2.rivier.edu/faculty/vriabov/CS552_SW_Design_Specification_Example.pdf>

3.[**https://www.cs.utah.edu/~jamesj/ayb2005/docs/SDS\_v2.htm**](https://www.cs.utah.edu/~jamesj/ayb2005/docs/SDS_v2.htm)

4.<http://portal.unimap.edu.my/portal/page/portal30/Lecturer%20Notes/KEJURUTERAAN_KOMPUTER/Semester%202%20Sidang%20Akademik%2020112012/EKT420%20Software%20Engineering/Example%20of%20Software%20Design%20Document(SDD)>

5. <http://en.wikipedia.org/wiki/Portable_Document_Format>

6. <https://www.kaggle.com/thec03u5/fifa-18-demo-player-dataset>

7.<https://www.datacamp.com/courses/building-web-applications-in-r-with-shiny?utm_source=adwords_ppc&utm_campaignid=1455363063&utm_adgroupid=65083631908&utm_device=c&utm_keyword=&utm_matchtype=b&utm_network=g&utm_adpostion=1t1&utm_creative=278443377107&utm_targetid=aud-299261629574:dsa-464026017093&utm_loc_interest_ms=&utm_loc_physical_ms=1012834&gclid=EAIaIQobChMI7Nro2taC3AIVGZ7VCh1h7AoHEAAYASAAEgKEH_D_BwE>