MINI PROJECT

(2020-21)

Building and Deployment of Web application

UNPLUG THE PLAYERS

PROJECT REPORT

Institute of Engineering & Technology

Submitted by: -

Sarvesh Kumar Sharma (181500625)

Satyam Kumar Jha (181500627)

Ashutosh Tripathi (181500152)

Jeevesh Gangwar (181500298)

Sachi Tripathi (181500598)

Supervised By: -

Mr. Vaibhav Diwan

Technical Trainer

Department of Computer Engineering & Applications

GLA University

Mathura- 281406,

INDIA

Department of computer Engineering and Applications

GLA University, Mathura

17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,

DECLARATION

I/we hereby declare that the work which is being presented in the B.Tech. Project **"UNPLUG THE PLAYERS"**, in partial fulfilment of the requirements for the award of the ***Bachelor of Technology*** in Computer Science and Engineering and submitted to the Department of Computer Engineering and Applications of GLA University, Mathura, is an authentic record of my/our own work carried under the supervision of **Mr. Vaibhav Diwan.**

The contents of this project report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree.

Sign ________________________
Name : Sarvesh Kumar Sharma
University Roll No.: 181500625

Sign ________________________
Name: Sachi Tripathi
University Roll No.: 181500598

Sign ________________________
Name: Satyam Kumar Jha
University Roll No.: 181500627

Sign ________________________
Name: Jeevesh Gangwar
University Roll No.: 181500298

Sign ________________________
Name: Ashutosh Tripathi
University Roll No.: 181500152

Date:

Department of computer Engineering and Applications

GLA University, Mathura

17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,

CERTIFICATE

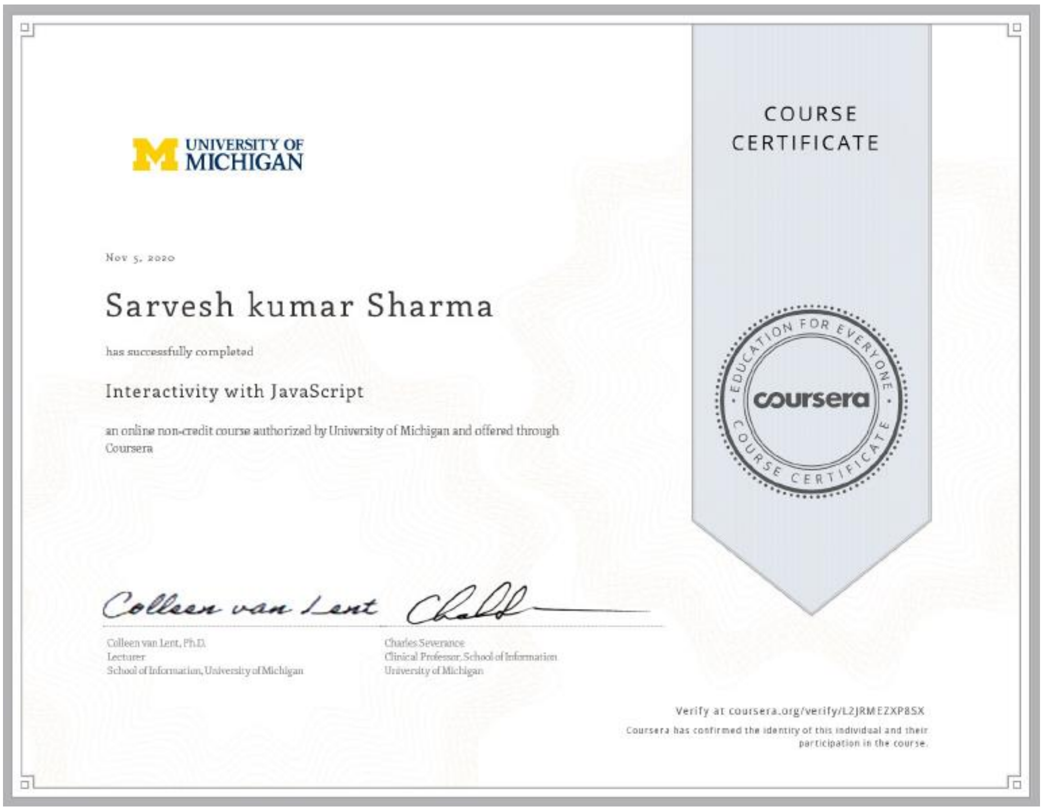
This is to certify that the project entitled “**Unplug the Players**”, carried out in Mini Project – I Lab, is a bonafide work by Sarvesh Kumar Sharma (181500625), Satyam Kumar Jha (181500627), Sachi Tripathi (181500598), Jeevesh Gangwar (181500298), and Ashutosh Tripathi (181500152) and is submitted in partial fulfillment of the requirements for the award of the degree Bachelor of Technology (Computer Science & Engineering).

Signature of Supervisor:

Name of Supervisor: Mr. Vaibhav Diwan

Date:

TRAINING CERTIFICATES



UNIVERSITY OF MICHIGAN

COURSE CERTIFICATE

Nov 5, 2020

Sarvesh kumar Sharma

has successfully completed

Interactivity with JavaScript

an online non-credit course authorized by University of Michigan and offered through Coursera

EDUCATION FOR EVERYONE · coursera · COURSE CERTIFICATE

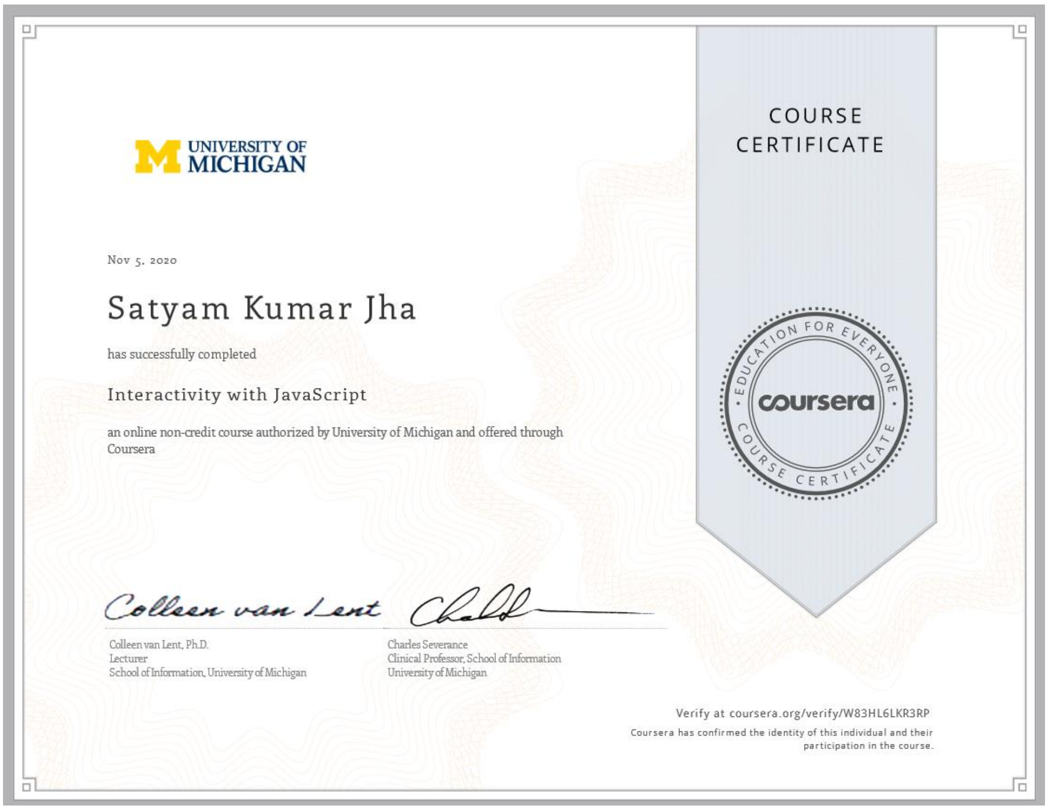
Colleen van Lent

Colleen van Lent, Ph.D.
Lecturer
School of Information, University of Michigan

Charles Severance
Clinical Professor, School of Information
University of Michigan

Verify at coursera.org/verify/L2JRMEZXP8SX

Coursera has confirmed the identity of this individual and their participation in the course.



UNIVERSITY OF MICHIGAN

COURSE CERTIFICATE

Nov 5, 2020

Satyam Kumar Jha

has successfully completed

Interactivity with JavaScript

an online non-credit course authorized by University of Michigan and offered through Coursera

EDUCATION FOR EVERYONE · coursera · COURSE CERTIFICATE

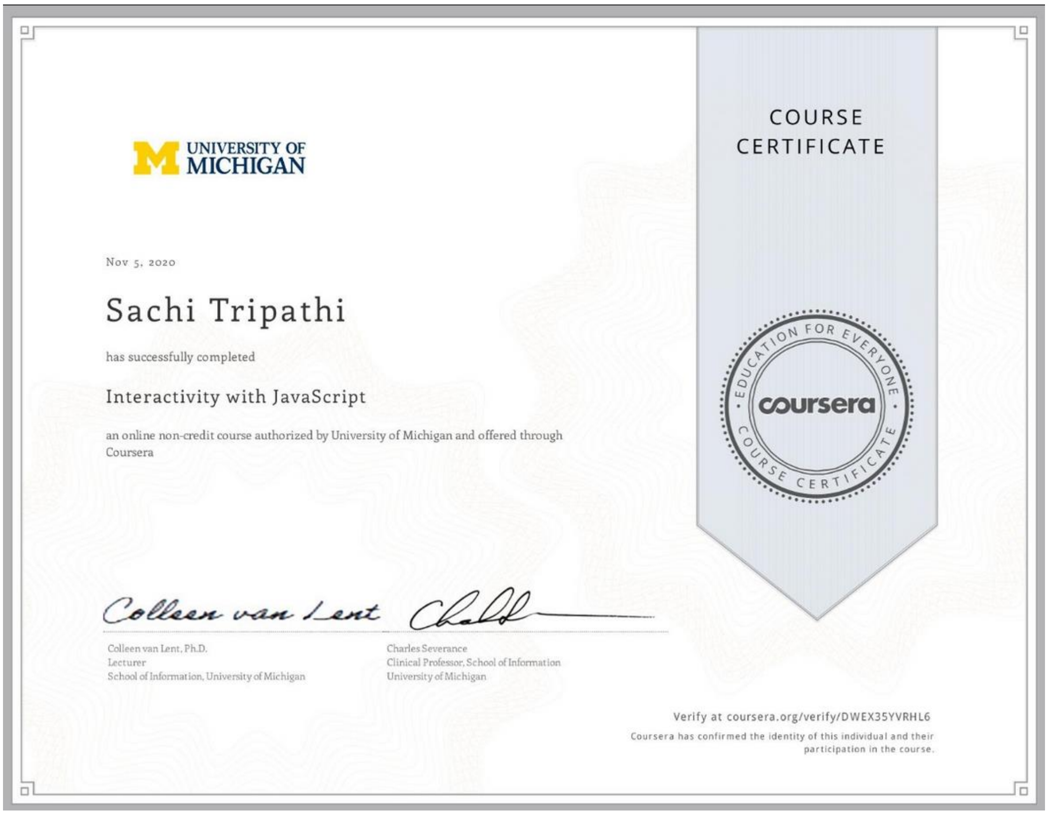
Colleen van Lent

Colleen van Lent, Ph.D.
Lecturer
School of Information, University of Michigan

Charles Severance
Clinical Professor, School of Information
University of Michigan

Verify at coursera.org/verify/W83HL6LKR3RP

Coursera has confirmed the identity of this individual and their participation in the course.



UNIVERSITY OF MICHIGAN

COURSE CERTIFICATE

Nov 5, 2020

Sachi Tripathi

has successfully completed

Interactivity with JavaScript

an online non-credit course authorized by University of Michigan and offered through Coursera

EDUCATION FOR EVERYONE · coursera · COURSE CERTIFICATE

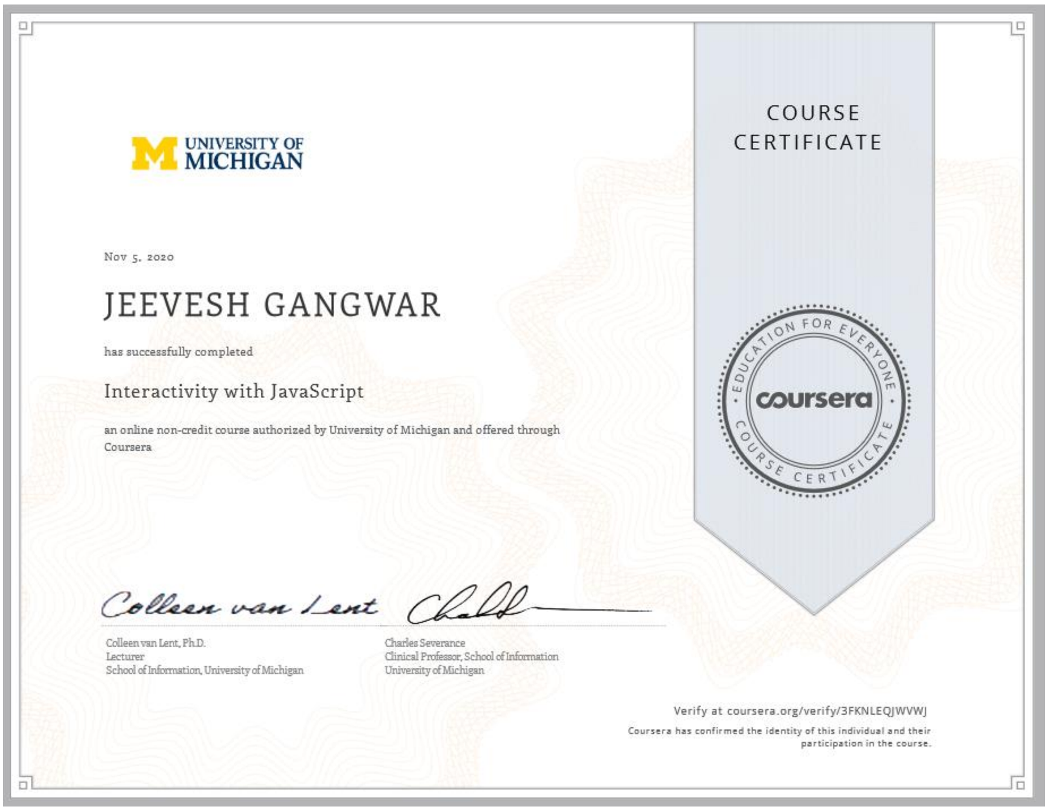
Colleen van Lent

Colleen van Lent, Ph.D.
Lecturer
School of Information, University of Michigan

Charles Severance
Clinical Professor, School of Information
University of Michigan

Verify at coursera.org/verify/DWEX35YVRHL6

Coursera has confirmed the identity of this individual and their participation in the course.



UNIVERSITY OF MICHIGAN

COURSE CERTIFICATE

Nov 5, 2020

JEEVESH GANGWAR

has successfully completed

Interactivity with JavaScript

an online non-credit course authorized by University of Michigan and offered through Coursera

EDUCATION FOR EVERYONE · coursera · COURSE CERTIFICATE

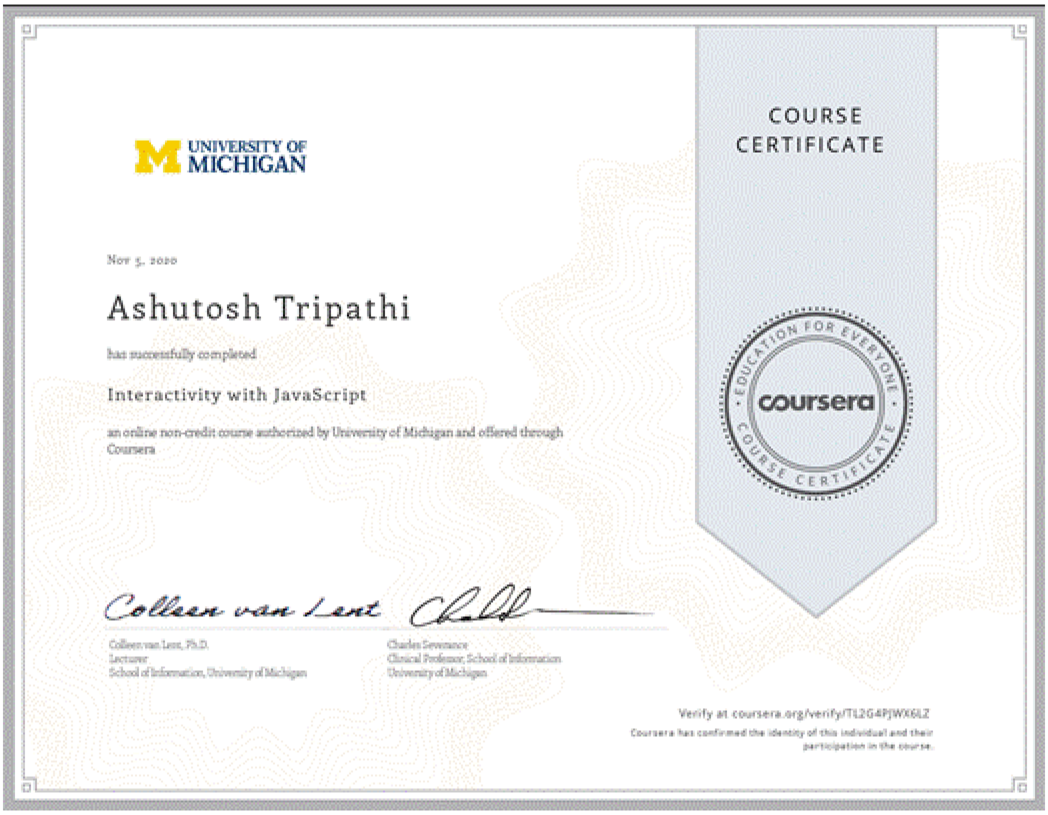
Colleen van Lent

Colleen van Lent, Ph.D.
Lecturer
School of Information, University of Michigan

Charles Severance
Clinical Professor, School of Information
University of Michigan

Verify at coursera.org/verify/3FKNLEQJWVWJ

Coursera has confirmed the identity of this individual and their participation in the course.



UNIVERSITY OF MICHIGAN

COURSE CERTIFICATE

Nov 5, 2020

Ashutosh Tripathi

has successfully completed

Interactivity with JavaScript

an online non-credit course authorized by University of Michigan and offered through Coursera

EDUCATION FOR EVERYONE
coursera
COURSE CERTIFICATE

Colleen van Lent

Colleen van Lent, Ph.D.
Lecturer
School of Information, University of Michigan

Charles Severance
Clinical Professor, School of Information
University of Michigan

Verify at coursera.org/verify/TL2G4PJWX6LZ

Coursera has confirmed the identity of this individual and their participation in the course.

ACKNOWLEDGEMENT

First and foremost, praises and thanks to the God, the Almighty, for His showers of blessings throughout our mini project to complete the project successfully.

I/we would like to express our deep and sincere gratitude to our college faculties for giving us this opportunity to do a mini project. I/We am extremely grateful to my mentor, Mr. Vaibhav Diwan, for his invaluable guidance throughout this mini project. His dynamism, vision, sincerity and motivation have deeply inspired us. He has guided us so well. It was a great privilege and honor to work and study under his guidance. I/we are extremely grateful for what he has offered us. I would also like to thank him for his empathy. I/we am extremely thankful to our friends and family for their acceptance and patience during this mini project.

I/We are extremely grateful to our parents for their love, prayers, caring and sacrifices for educating and preparing us for my future.

Name : Sarvesh Kumar Sharma
University Roll No.: 181500625

Name: Satyam Kumar Jha
University Roll No.: 181500627

Name: Ashutosh Tripathi
University Roll No.: 181500152

Name: Sachi Tripathi
University Roll No.: 181500598

Name: Jeevesh Gangwar
University Roll No.: 181500298

ABSTRACT

In this we will build a simple web application which will be based on QUIZ GAME related to the field of sport, 'Football'. We will then deploy the that web application using Flask with python.

This work deals with the development of web-based application game namely: Unplug the Players. This application is developed for knowledge purpose, allowing the users to play a quiz and enhance their knowledge in the field of sports particularly Football. This application is based on giving the scores to the player who have attempted to play.

For the development of this web application, data is collected from different sources about the football players from all over the world. This application will make use of JavaScript, Flask, CSS, Python.

A photograph will be displayed on the screen along with some information about the player and the user have to identify the player based on that information. After each try the photograph displayed will be changed and a new photograph will be displayed on the screen. The player will be given several attempts and after every chance some message will be displayed according to the players performance and based on that scores will be awarded.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1. INTRODUCTION

This project is designed to build a web-based application for the entertainment and knowledge of the people, where someone can play a quiz and be entertained and as well as can enhance his knowledge in that field. A simple enjoyable game to test anyone’s knowledge in the field of sports and particularly in one sport as it will be a little problematic if every sport will be included in a single game.

This mainly focuses on the sport Football. Football is a family of team sports that involves, to varying degrees, kicking a ball to score a goal. Unqualified, the word football normally means the form of football that is the most popular where the word is used. Sports commonly called football include association football (known as soccer in some countries); gridiron football (specifically American football or Canadian football); Australian rules football; rugby football (either rugby union or rugby league); and Gaelic football. These various forms of football share to varying extent common origins and are known as football codes.

There are a number of references to traditional, ancient, or prehistoric ball games played in many different parts of the world. Contemporary codes of football can be traced back to the codification of these games at English public schools during the 19th century.

In 1888, The Football League was founded in England, becoming the first of many professional football competitions.

1.2. OBJECTIVE

The main objective of the project is to create a web app game which allows to explore our knowledge in football. It will be like quiz-based game where game player will be awarded different points on his correct answer at different stage. The pre-objective is to gather the complete data and pre-process the data on which our web app will run.

1.3. MOTIVATION

"Knowing Is Not Enough; We Must Apply. Wishing Is Not Enough; We Must Do."

The motivation behind the building of this web application is that as we all know today there are many people who are going away from the sports as they are busy in their virtual world of Smartphone. So, making them aware about Sports in their way we came up with this idea of making them aware by using their Smartphone. As they don't have time for the real world so we will give them the awareness in their smartphone's virtual world without making them do any other efforts just by playing this Web game.

1.4. FEATURES

a) The game designed is interactive and user friendly.

b) Simple and easy to play

c) Easy to modify

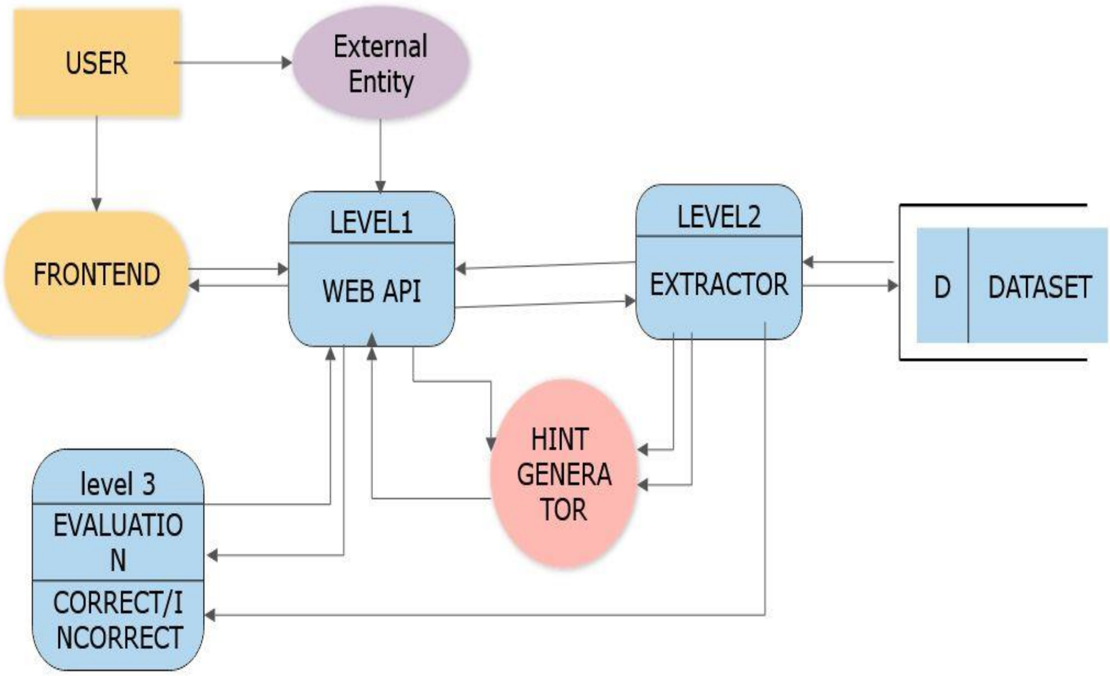


Figure 1: General working

1.5. TECHONOLOGIES USED

1.5.1. Machine Learning

Machine Learning is a discipline for artificial intelligence for building computer programs that automatically improve through experience and make predictions. The term Machine Learning was coined by Arthur Samuel in 1959, an American pioneer in the field of computer gaming and artificial intelligence and stated that "it gives computers the ability to learn without being explicitly programmed".

And in 1997, Tom Mitchell gave a "well-posed" mathematical and relational definition that "A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.

Machine learning is a set of algorithms that perform a certain task with the input data and also improve their performance. These algorithms match the input to the output, thereby resulting in the prediction of patterns. The more data is fed to the algorithms, the more accurate the predictions are.

Machine learning implementations are classified into three major categories, depending on the nature of the learning "signal" or "response" available to a learning system which are as follows: -

Supervised learning: When an algorithm learns from example data and associated target responses that can consist of numeric values or string labels, such as classes or tags, in order to later predict the correct response when posed with new examples comes under the category of Supervised learning. This approach is indeed similar to human learning under the supervision of a teacher. The teacher provides good examples for the student to memorize, and the student then derives general rules from these specific examples.

Unsupervised learning: When an algorithm learns from plain examples without any associated response, leaving to the algorithm to determine the data patterns on its own. This type of algorithm tends to restructure the data into something else, such as new features that may represent a class or a new series of un-correlated values. They are quite useful in providing humans with insights into the meaning of data and new useful inputs to supervised machine learning algorithms.

Reinforcement learning: When you present the algorithm with examples that lack labels, as in unsupervised learning. However, you can accompany an example with positive or negative feedback according to the solution the algorithm proposes comes under the category of Reinforcement learning, which is connected to applications for which the algorithm must make decisions (so the product is prescriptive, not just descriptive, as in unsupervised learning), and the decisions bear consequences. In the human world, it is just like learning by trial and error.

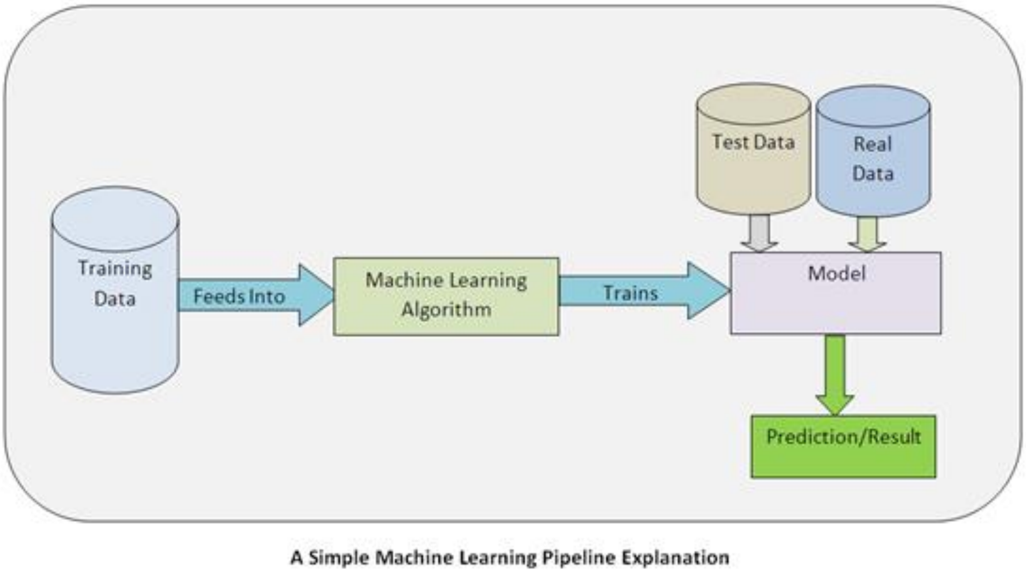

Training Data
Feeds Into
Machine Learning Algorithm
Trains
Test Data
Real Data
Model
Prediction/Result
A Simple Machine Learning Pipeline Explanation

Figure 2:ML Model Working

1.5.2. Flask

Flask is a web application framework written in Python. It is developed by Armin Ronacher, who leads an international group of Python enthusiasts named Pocco. Flask is based on the Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.

What is Web Framework?

Web Application Framework or simply Web Framework represents a collection of libraries and modules that enables a web application developer to write applications without having to bother about low-level details such as protocols, thread management etc.

WSGI

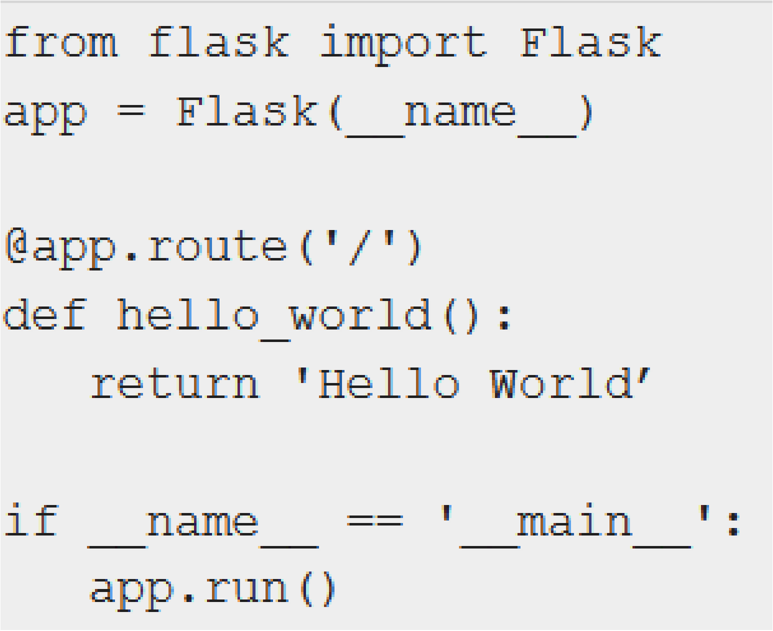
Web Server Gateway Interface (WSGI) has been adopted as a standard for Python web application development. WSGI is a specification for a universal interface between the web server and the web applications.

Werkzeug

It is a WSGI toolkit, which implements requests, response objects, and other utility functions. This enables building a web framework on top of it. The Flask framework uses Werkzeug as one of its bases.

Jinja2

Jinja2 is a popular templating engine for Python. A web templating system combines a template with a certain data source to render dynamic web pages. Flask is often referred to as a micro framework. It aims to keep the core of an application simple yet extensible. Flask does not have built-in abstraction layer for database handling, nor does it have form a validation support. Instead, Flask supports the extensions to add such functionality to the application.



```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_world():
   return 'Hello World’

if __name__ == '__main__':
   app.run()
```

Figure 3:Flask Basic Skelton

Importing flask module in the project is mandatory. An object of Flask class is our WSGI application. Flask constructor takes the name of current module (__name__) as argument. The route() function of the Flask class is a decorator, which tells the application which URL should call the associated function.

```
app.route(rule, options)
```

The rule parameter represents URL binding with the function. The options are a list of parameters to be forwarded to the underlying Rule object. In the above example, '/' URL is bound with hello_world() function. Hence, when the home page of web server is opened in browser, the output of this function will be rendered. Finally, the run() method of Flask class runs the application on the local development server.

```
app.run(host, port, debug, options)
```

Sr.No.	Parameters & Description
1	**Host** Hostname to listen on. Defaults to 127.0.0.1 (localhost). Set to '0.0.0.0' to have server available externally
2	**port** Defaults to 5000
3	**debug** Defaults to false. If set to true, provides a debug information
4	**options** To be forwarded to underlying Werkzeug server.

Table 1: Flask Parameter Descriptions

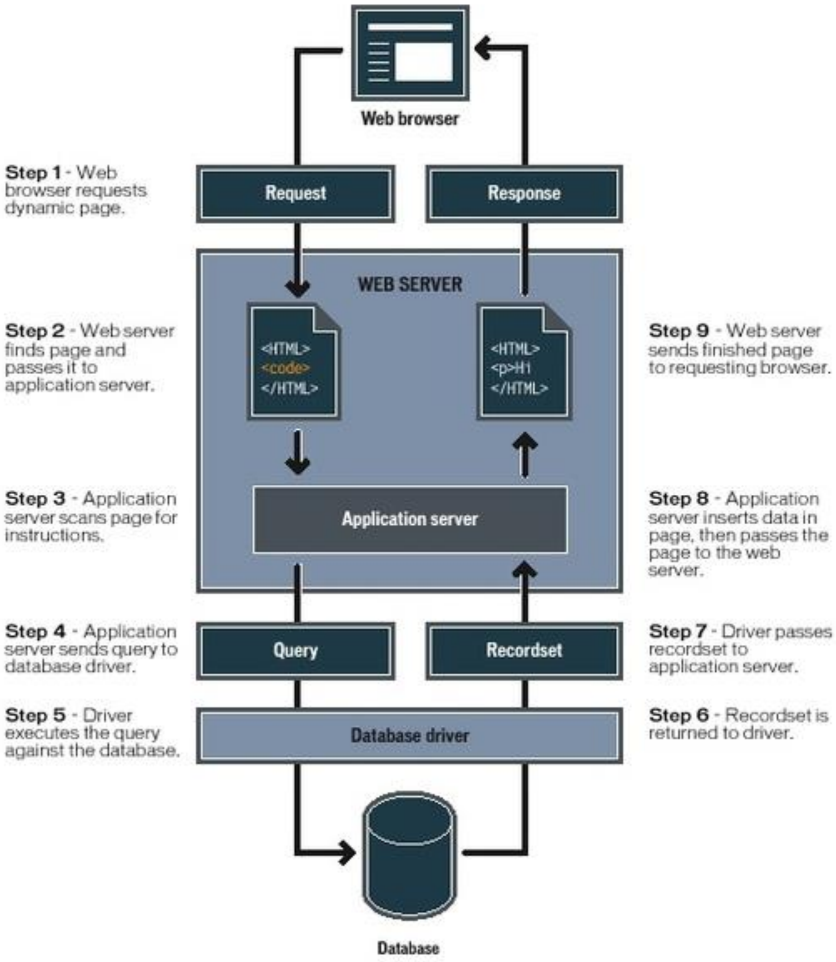

Web browser
Step 1 - Web browser requests dynamic page.
Request
Response
WEB SERVER
Step 2 - Web server finds page and passes it to application server.
<HTML>
<code>
</HTML>
<HTML>
<p>Hi
</HTML>
Step 9 - Web server sends finished page to requesting browser.
Step 3 - Application server scans page for instructions.
Application server
Step 8 - Application server inserts data in page, then passes the page to the web server.
Step 4 - Application server sends query to database driver.
Query
Recordset
Step 7 - Driver passes recordset to application server.
Step 5 - Driver executes the query against the database.
Database driver
Step 6 - Recordset is returned to driver.
Database

Figure 4: Web App Working

1.5.3. SciPy

SciPy is a free and open-source Python library used for scientific computing and technical computing. SciPy contains modules for optimization, linear algebra, integration, interpolation, special functions, FFT, signal and image processing, ODE solvers and other tasks common in science and engineering.

In this project, we used SciPy spatial Distance . Distance matrix computation from a collection of raw observation vectors stored in a rectangular array.

Here are used functions.

pdist(X[, metric])	Pairwise distances between observations in n-dimensional space.
cdist(XA, XB[, metric])	Compute distance between each pair of the two collections of inputs.
squareform(X[, force, checks])	Convert a vector-form distance vector to a square-form distance matrix, and vice-versa.
directed_hausdorff(u, v[, seed])	Compute the directed Hausdorff distance between two N-D arrays.

Table 2:SciPy Functions1

Predicates for checking the validity of distance matrices, both condensed and redundant. Also contained in this module are functions for computing the number of observations in a distance matrix.

is_valid_dm(D[, tol, throw, name, warning])	Return True if input array is a valid distance matrix.
is_valid_y(y[, warning, throw, name])	Return True if the input array is a valid condensed distance matrix.
num_obs_dm(d)	Return the number of original observations that correspond to a square, redundant distance matrix.
num_obs_y(Y)	Return the number of original observations that correspond to a condensed distance matrix.

Table 3: SciPy Validation

Distance functions between two numeric vectors `u` and `v`. Computing distances over a large collection of vectors is inefficient for these functions. Use pdist for this purpose.

1.5.4. Front-end Design: HTML, CSS, Bootstrap

Hypertext Markup Language(HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets(CSS) and JavaScript, it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages.

HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. Cascading Style Sheets(CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML.

CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Bootstrap is a free and open-source front-end library for designing websites and web applications. It contains HTML and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. Unlike many web frameworks, it concerns itself with front-end development only.

1.5.5. Client-side validation: JavaScript, jQuery

JavaScript often abbreviated as JS, is ahigh-level, interpreted programming language. It is a language which is also characterized as dynamic, weakly typed, prototype-based and multi-paradigm. Alongside HTML and CSS, JavaScript is one of the three core technologies of the World Wide Web. JavaScript enables interactive web pages and thus is an essential part of web applications. The vast majority of websites use it, and all major web browsers have a dedicated JavaScript engine to execute it.

jQuery is a cross-platform JavaScript library designed to simplify the client-side scripting of HTML. It is free, open-source software using the permissive MIT License. Web analysis indicates that it is the most widely deployed JavaScript library by a large margin.

1.5.6. Python:

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL).

Python's large standard library, commonly cited as one of its greatest strengths, provides tools suited to many tasks. For Internet-facing applications, many standard formats and protocols such as MIME and HTTP are supported.

There are many third-party python libraries used in this application for performing various tasks, the list is as follows:

- Flask
- Pandas
- SciPy Etc.

1.6. SCOPE

The project that has been created is a fully automated version of kind of a quiz game. The aim of this project is to provide people of all ages with an interactive quiz environment where they can improve their sport’s knowledge particularly of football by advancing through the game. This web app can be a booster in youth by including more features like general knowledge, field description, type of events, teams and many more and finally giving a complete virtual feel of football to the player which allow game player to play games in different domains and other sports.

2. SOFTWARE REQUIREMENT ANALYSIS

2.1. INTRODUCTION

The aim of this part is to gather and analyze and give an in-depth insight of the complete **UNPLUG THE PLAYERS PROJECT** by defining the problem statement in detail. Nevertheless, it also concentrates on the capabilities required by stakeholders and their needs while defining high-level product features. The detailed requirements of the UNPLUG THE PLAYERS PROJECT are provided in this document.

2.1.1. Purpose

The purpose of the document is to collect and analyze all assorted ideas that have come up to define the system, its requirements with respect to consumers. Also, we shall predict and sort out how we hope this product will be used in order to gain a better understanding of the project, outline concepts that may be developed later, and document ideas that are being considered, but may be discarded as the product develops.

In short, the purpose of this report document is to provide a detailed overview of our software product, its parameters and goals. This document describes the project's target audience and its user interface, hardware and software requirements. It defines how our client, team and audience see the product and its functionality. Nonetheless, it helps any designer and developer to assist in software delivery lifecycle (SDLC) processes.

2.1.2. Document Convention:

In this text, it will use font small 2 and overstriking for primary title, font small 3 for secondary title and font 4 for the content. And it will use the italic when mentions the name of the application UNPLUG THE PLAYERS.

2.1.3. Intended Audience:

This SRS about UNPLUG THE PLYERS is for developers, mentors, users and testers. The article mainly introduces the overall description, external interface requirements, system features and other non-functional requirements. I suppose mentor to read the whole article carefully and user pay attention to overall description especially. Users and testers read the system features carefully.

2.1.4. Definitions, Acronyms, and Abbreviations.

Configuration	It means a product which is available / Selected from a catalogue can be customized.
FAQ	Frequently Asked Questions
CRM	Customer Relationship Management
RAID 5	Redundant Array of Inexpensive Disk/Drives

2.2. PERSPECTIVE:

A web application preferably using the Flask framework (http://flask.pocoo.org/).

2.3. PRODUCT FUNCTIONS:

Help the users to use this application to grab the information of football players, and play an interactive quiz game.

The web application has a simple interface with a single button [START THE GAME] and User will click on that to play the game. Accepting the command, the software will redirect to a different page and will return a table of different football players with some common features between them and provide a hint button to take hint from the system in order to identify the correct player.

Depending upon the number of hints user takes in order to correctly identify the player, software will return some points to user. If the user fails to identify the correct player or select the wrong player, software will return zero points to the user and will restart the game instantly.

- RESPONSE TIME: < 1 Second.

2.4. USER CLASSES AND CHARACTERISCTICS:

Our application mainly those who love playing games. Mostly they are School and College going students, but it doesn’t rule out some other people want to use this application to get knowledge about football players.

The web application users interact with this application through a web browser. Other web applications are those applications like bots, third party application, can also use this application through it web Api. All requests should meet the specifications of application Api.

2.5. OPERATING ENIRNOMENT:

Our software is a multi-functional software system based on the windows platform. It is compatible and can run on 64 - bit laptop or ordinary desktop.

2.6. DESIGN CONSTRAINS:

Our application must accept the command and then start the game. We must consider about the arrangement and beautification of the interface; Prioritization of processing operations and it deepens the difficulty of coding and testing. This application needs users to enroll in the game and then choosing the players with similar characteristics. Every time, it will be different players will be chosen. So, the application must analyses a variety of players from dataset. It may raise the error rate.

2.7. ASSUMPTIONS AND DEPENDENCIES:

The people who manage the application should know about the knowledge of the football and football players in order to update the dataset at regular basis.

UNPLUG THE PLAYERS is a web app having *FLASK* app for backend interactions or third-party application interactions. It also uses *python* packages and *JavaScript* for frontend interactions.

List- Python, Flask, SciPy, Pandas, JavaScript, HTML, and CSS.

2.8. FUNCTIONAL REQUIREMENT:

2.8.1. User Interfaces:

UI-1: This will show a start button which will redirect to game page. It will also show the rules and regulations for the game the codes to be taken care during game play, along with it will show the user to get in contact to the developers for any query regarding game.

UI-2: This will show the game page. It will contain 5 different player pictures without their names and will show a hint button to the user. With each click user can get a hint regarding that player. The hint will continue to show till user takes 5 hints in a row.

UI-3: This will show the output of the result if user has selected the correct player or not and corresponding points he earned for that round.

2.9. PERORMANCE REQUIREMENTS:

Since the system use Flask based server client architecture, and use the large source information from distant server, it fetches data through internet. Internet bandwidth is major performance parameter. As System uses GitHub repository as source of massive data, multiple request and response is needed to handle in QuickTime for instant result back to user. Large system queries are handled by the fast operating systems for both the mobile and the web-based process. Worker are used to provide faster HTTP Request handling.

2.9.1. Hardware Requirements:

The requests of the hardware for the web application are as followed:

- 64 bits laptop or desktop.
- TCP protocol.

2.9.2. Software Requirements:

To access a web portal of this application, its only need a PC/Laptop/Mobile with an integrated and updated web browser.

Desktop browser: Safari, Chrome, Firefox, Opera, IE9+.

Mobile browsers: Android, Chrome Mobile, iOS Safari.

On the server side: a PC/Web Server which meets these specifications:

- Window Operating System
- At least 2 GB RAM and 150 GB Free Space
- Redis Server Installed
- Python Compiler Installed

2.10. COMMUNICATION PROTOCOLS AND INTERFACES:

A. TCP protocol.

B. Secure transmission and encryption techniques.

C. File transfer rate.

3. SOFTWARE DESIGN

Function Oriented Design for procedural approach and different diagram to show the designing of the application.

3.1. DATAFLOW DIAGRAM

3.1.1. DFD Level 0

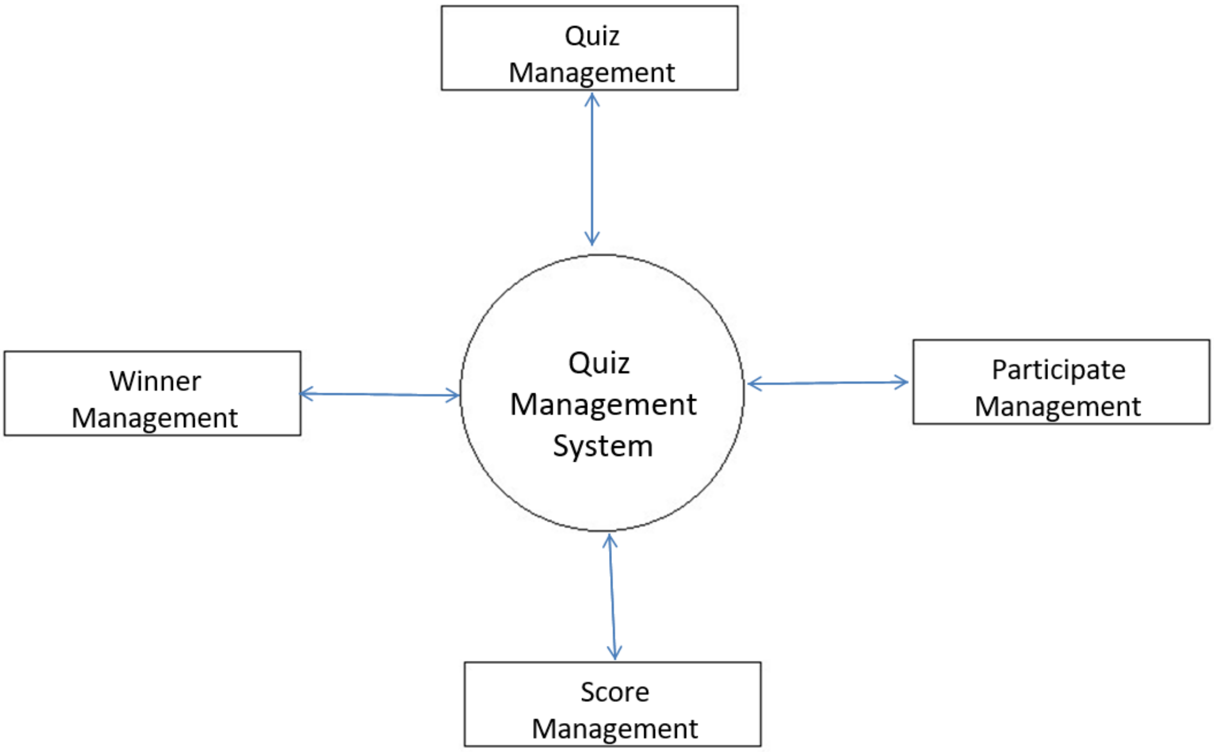


Figure 5:DFD Level 0

3.1.2. DFD Level 1

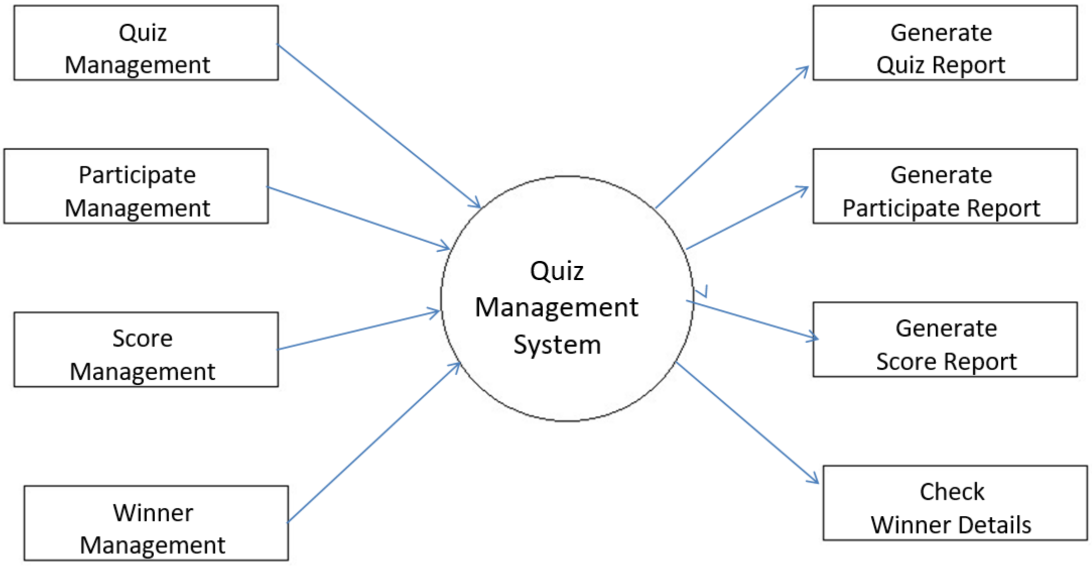


Figure 6:DFD Level 1

3.1.3. DFD Level 2

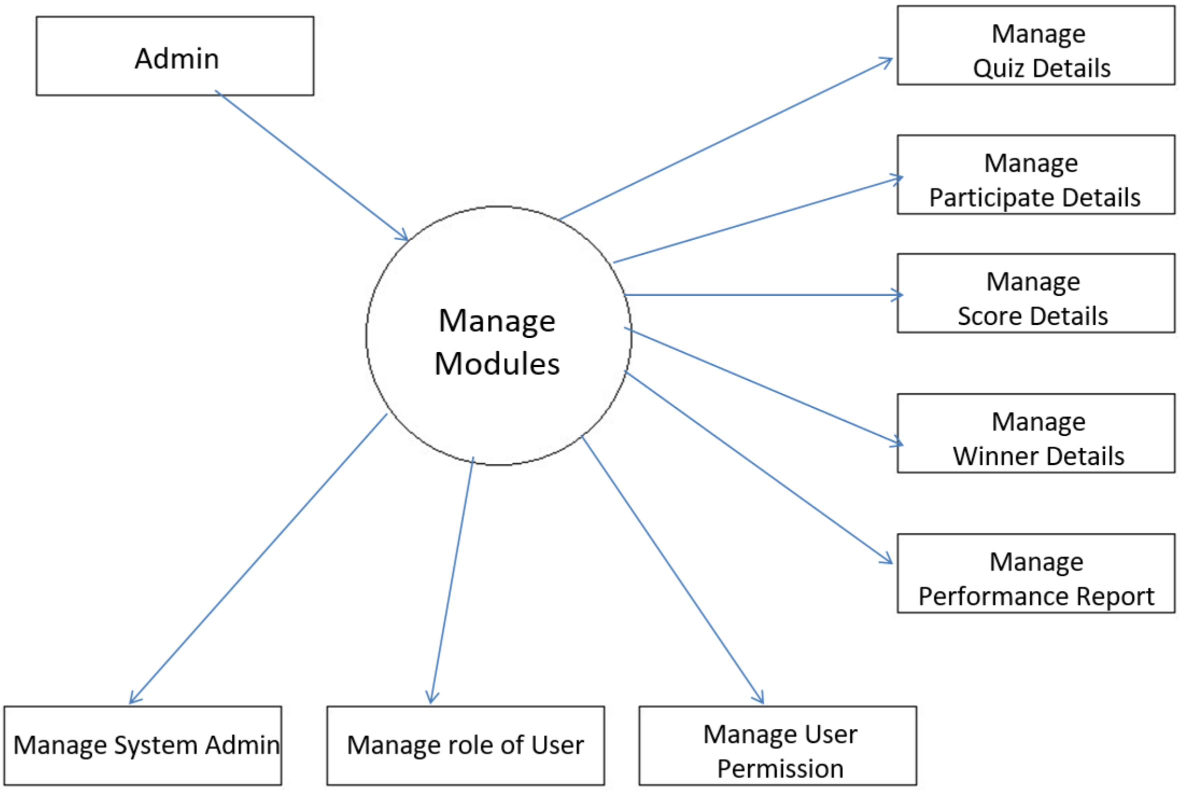


Figure 7: DFD Level 2

3.2. UML DIAGRAMS

3.2.1. Sequence Diagram

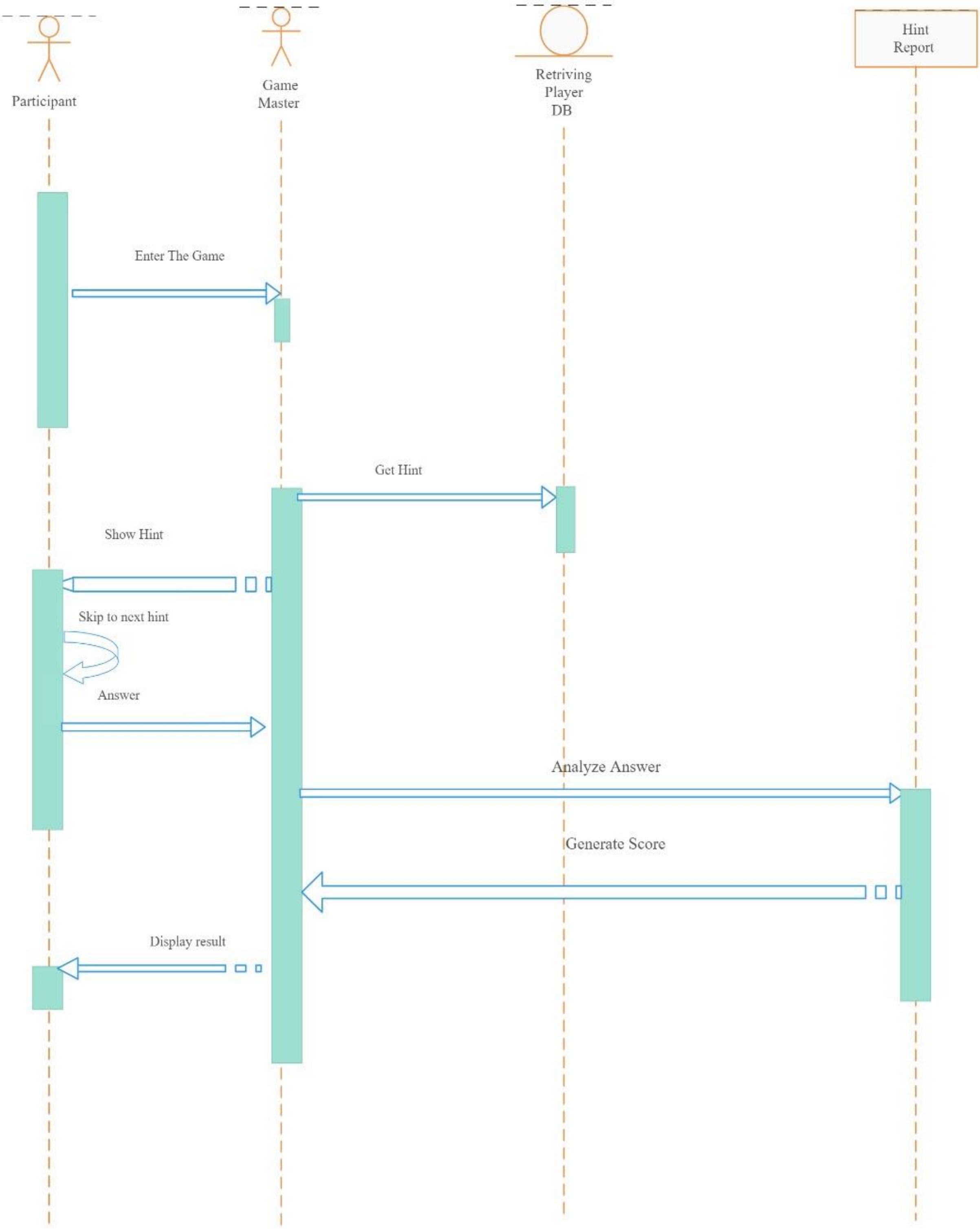


Figure 8:Sequence Diagram

3.2.2. Class Diagram

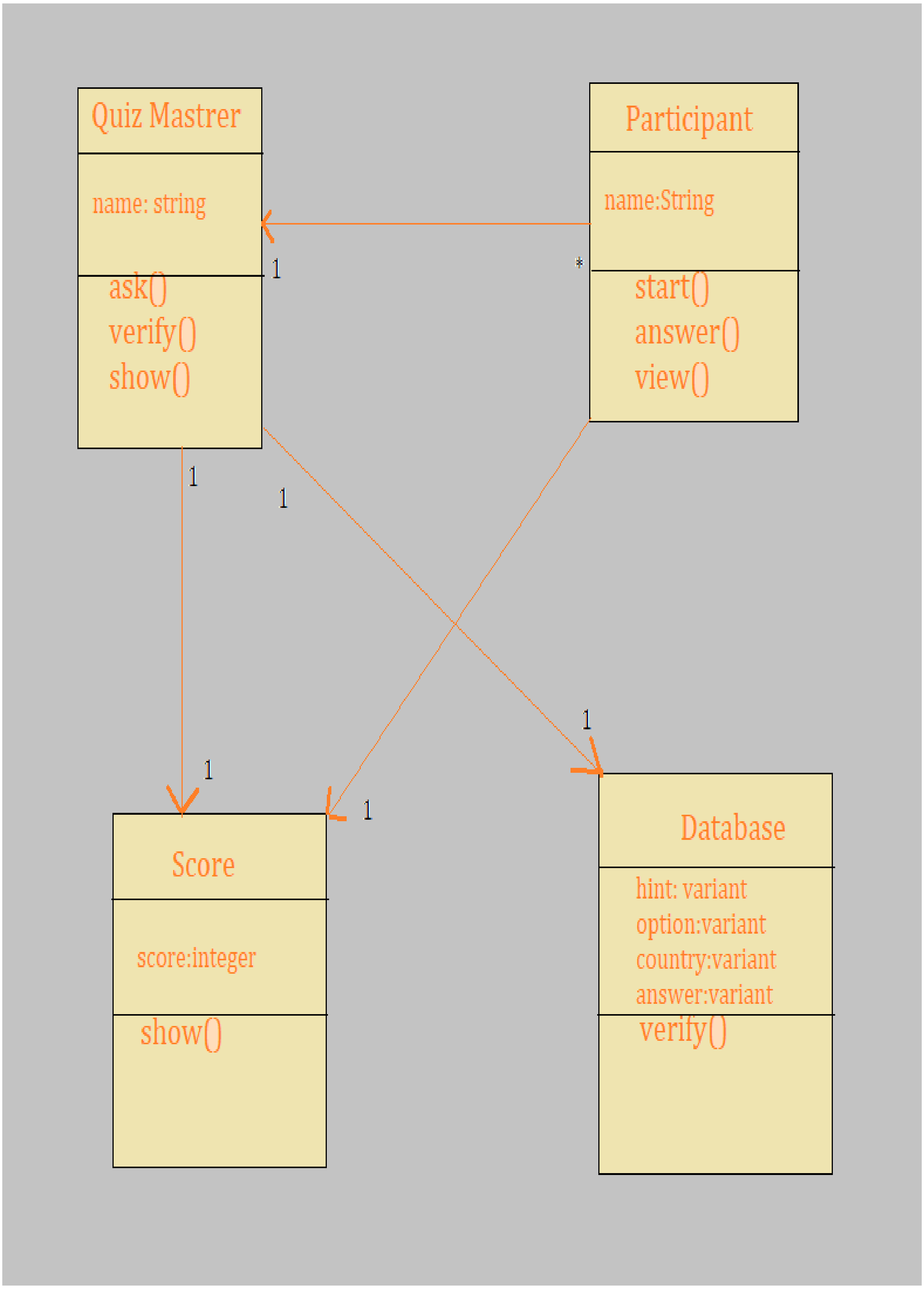


Figure 9:Class Diagram

3.2.3. UseCase Diagram

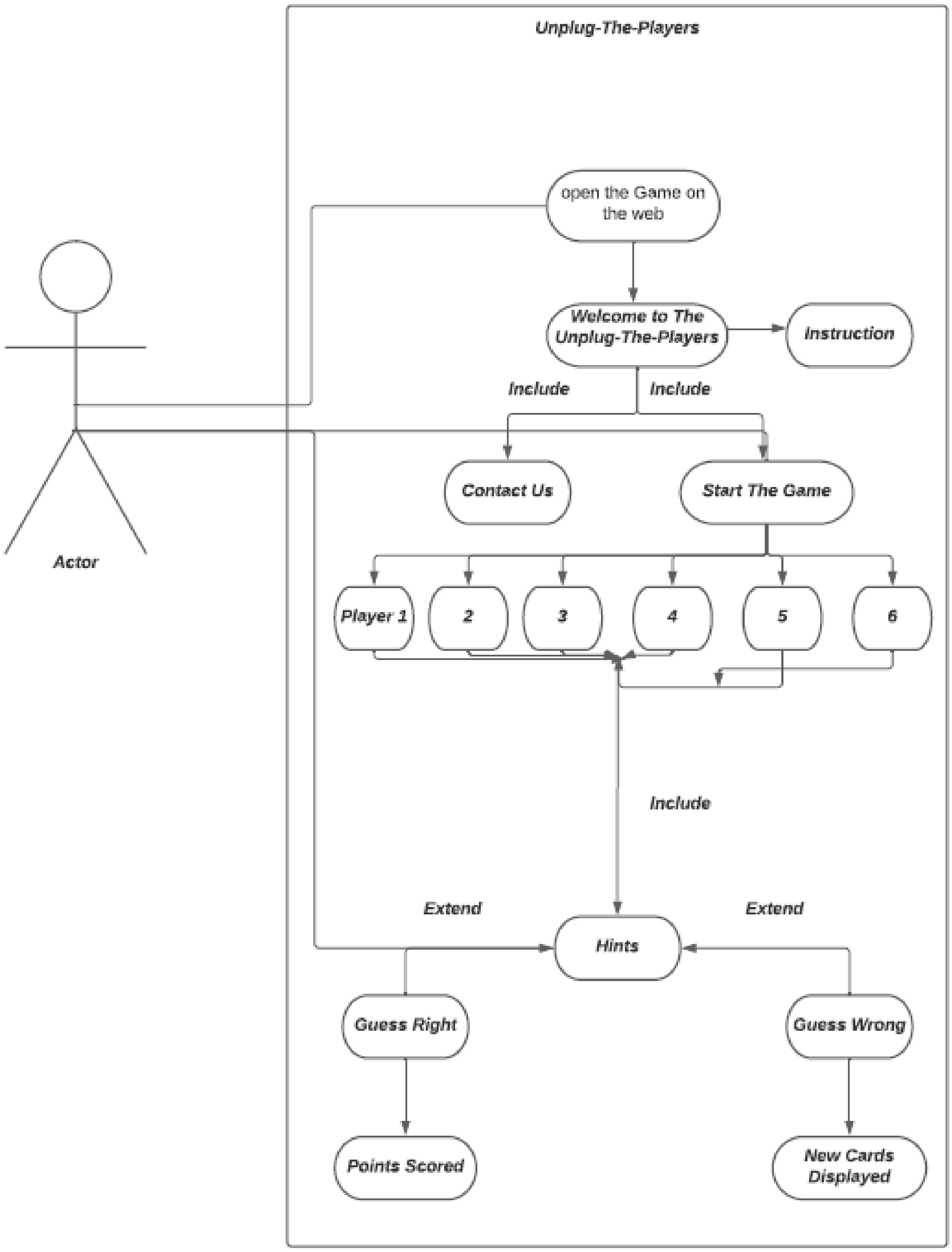


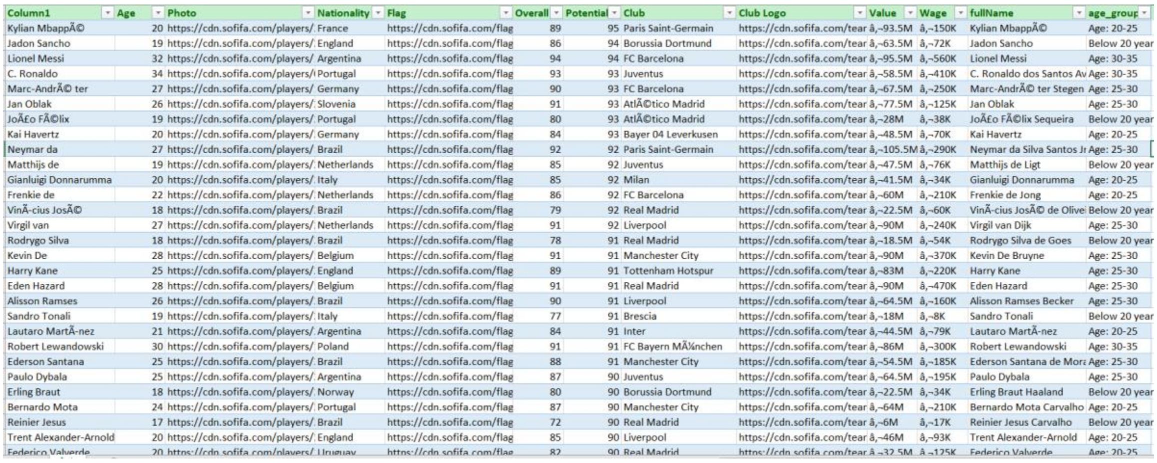
Figure 10:UseCase Diagram

3.3. DATABASE DESIGN

3.3.1. Dataset Description

There is no actual database for the project UNPLUG THE PLAYER but a dataset in form of a .csv file. Our dataset basically contains 13 attributes or features and 500 instances or examples. These features include Name ,Age, Photo, Nationality, Flag, Overall Potential, Club, Club Logo, Value, Wage, full Name and age_ group of different players who played FIFA 2020. Basically, our we extract the similar data from this data set using spatial distance measurement and will choose randomly six among them.

Our dataset looks like this.



Column1	Age	Photo	Nationality	Flag	Overall	Potential	Club	Club Logo	Value	Wage	fullName	age_group
Kylian MbappÃ©	20	https://cdn.sofifa.com/players/	France	https://cdn.sofifa.com/flag	89	95	Paris Saint-Germain	https://cdn.sofifa.com/tear	â‚¬93.5M	â‚¬150K	Kylian MbappÃ©	Age: 20-25
Jadon Sancho	19	https://cdn.sofifa.com/players/	England	https://cdn.sofifa.com/flag	86	94	Borussia Dortmund	https://cdn.sofifa.com/tear	â‚¬63.5M	â‚¬72K	Jadon Sancho	Below 20 year
Lionel Messi	32	https://cdn.sofifa.com/players/	Argentina	https://cdn.sofifa.com/flag	94	94	FC Barcelona	https://cdn.sofifa.com/tear	â‚¬95.5M	â‚¬560K	Lionel Messi	Age: 30-35
C. Ronaldo	34	https://cdn.sofifa.com/players/	Portugal	https://cdn.sofifa.com/flag	93	93	Juventus	https://cdn.sofifa.com/tear	â‚¬58.5M	â‚¬410K	C. Ronaldo dos Santos Av	Age: 30-35
Marc-AndrÃ© ter	27	https://cdn.sofifa.com/players/	Germany	https://cdn.sofifa.com/flag	90	93	FC Barcelona	https://cdn.sofifa.com/tear	â‚¬67.5M	â‚¬250K	Marc-AndrÃ© ter Stegen	Age: 25-30
Jan Oblak	26	https://cdn.sofifa.com/players/	Slovenia	https://cdn.sofifa.com/flag	91	93	AtlÃ©tico Madrid	https://cdn.sofifa.com/tear	â‚¬77.5M	â‚¬125K	Jan Oblak	Age: 25-30
JoÃ£o FÃ©lix	19	https://cdn.sofifa.com/players/	Portugal	https://cdn.sofifa.com/flag	80	93	AtlÃ©tico Madrid	https://cdn.sofifa.com/tear	â‚¬28M	â‚¬38K	JoÃ£o FÃ©lix Sequeira	Below 20 year
Kai Havertz	20	https://cdn.sofifa.com/players/	Germany	https://cdn.sofifa.com/flag	84	93	Bayer 04 Leverkusen	https://cdn.sofifa.com/tear	â‚¬48.5M	â‚¬70K	Kai Havertz	Age: 20-25
Neymar da	27	https://cdn.sofifa.com/players/	Brazil	https://cdn.sofifa.com/flag	92	92	Paris Saint-Germain	https://cdn.sofifa.com/tear	â‚¬105.5M	â‚¬290K	Neymar da Silva Santos Jr	Age: 25-30
Matthijs de	19	https://cdn.sofifa.com/players/	Netherlands	https://cdn.sofifa.com/flag	85	92	Juventus	https://cdn.sofifa.com/tear	â‚¬47.5M	â‚¬76K	Matthijs de Ligt	Below 20 year
Gianluigi Donnarumma	20	https://cdn.sofifa.com/players/	Italy	https://cdn.sofifa.com/flag	85	92	Milan	https://cdn.sofifa.com/tear	â‚¬41.5M	â‚¬34K	Gianluigi Donnarumma	Age: 20-25
Frenkie de	22	https://cdn.sofifa.com/players/	Netherlands	https://cdn.sofifa.com/flag	86	92	FC Barcelona	https://cdn.sofifa.com/tear	â‚¬60M	â‚¬210K	Frenkie de Jong	Age: 20-25
VinÃ-cius JosÃ©	18	https://cdn.sofifa.com/players/	Brazil	https://cdn.sofifa.com/flag	79	92	Real Madrid	https://cdn.sofifa.com/tear	â‚¬22.5M	â‚¬60K	VinÃ-cius JosÃ© de Olive	Below 20 year
Virgil van	27	https://cdn.sofifa.com/players/	Netherlands	https://cdn.sofifa.com/flag	91	92	Liverpool	https://cdn.sofifa.com/tear	â‚¬90M	â‚¬240K	Virgil van Dijk	Age: 25-30
Rodrygo Silva	18	https://cdn.sofifa.com/players/	Brazil	https://cdn.sofifa.com/flag	78	91	Real Madrid	https://cdn.sofifa.com/tear	â‚¬18.5M	â‚¬54K	Rodrygo Silva de Goes	Below 20 year
Kevin De	28	https://cdn.sofifa.com/players/	Belgium	https://cdn.sofifa.com/flag	91	91	Manchester City	https://cdn.sofifa.com/tear	â‚¬90M	â‚¬370K	Kevin De Bruyne	Age: 25-30
Harry Kane	25	https://cdn.sofifa.com/players/	England	https://cdn.sofifa.com/flag	89	91	Tottenham Hotspur	https://cdn.sofifa.com/tear	â‚¬83M	â‚¬220K	Harry Kane	Age: 25-30
Eden Hazard	28	https://cdn.sofifa.com/players/	Belgium	https://cdn.sofifa.com/flag	91	91	Real Madrid	https://cdn.sofifa.com/tear	â‚¬90M	â‚¬470K	Eden Hazard	Age: 25-30
Alisson Ramses	26	https://cdn.sofifa.com/players/	Brazil	https://cdn.sofifa.com/flag	90	91	Liverpool	https://cdn.sofifa.com/tear	â‚¬64.5M	â‚¬160K	Alisson Ramses Becker	Age: 25-30
Sandro Tonali	19	https://cdn.sofifa.com/players/	Italy	https://cdn.sofifa.com/flag	77	91	Brescia	https://cdn.sofifa.com/tear	â‚¬18M	â‚¬8K	Sandro Tonali	Below 20 year
Lautaro MartÃ-nez	21	https://cdn.sofifa.com/players/	Argentina	https://cdn.sofifa.com/flag	84	91	Inter	https://cdn.sofifa.com/tear	â‚¬44.5M	â‚¬79K	Lautaro MartÃ-nez	Age: 20-25
Robert Lewandowski	30	https://cdn.sofifa.com/players/	Poland	https://cdn.sofifa.com/flag	91	91	FC Bayern MÃ¼nchen	https://cdn.sofifa.com/tear	â‚¬86M	â‚¬300K	Robert Lewandowski	Age: 30-35
Ederson Santana	25	https://cdn.sofifa.com/players/	Brazil	https://cdn.sofifa.com/flag	88	91	Manchester City	https://cdn.sofifa.com/tear	â‚¬54.5M	â‚¬185K	Ederson Santana de Mora	Age: 25-30
Paulo Dybala	25	https://cdn.sofifa.com/players/	Argentina	https://cdn.sofifa.com/flag	87	90	Juventus	https://cdn.sofifa.com/tear	â‚¬64.5M	â‚¬195K	Paulo Dybala	Age: 25-30
Erling Braut	18	https://cdn.sofifa.com/players/	Norway	https://cdn.sofifa.com/flag	80	90	Borussia Dortmund	https://cdn.sofifa.com/tear	â‚¬22.5M	â‚¬34K	Erling Braut Haaland	Below 20 year
Bernardo Mota	24	https://cdn.sofifa.com/players/	Portugal	https://cdn.sofifa.com/flag	87	90	Manchester City	https://cdn.sofifa.com/tear	â‚¬64M	â‚¬210K	Bernardo Mota Carvalho	Age: 20-25
Reinier Jesus	17	https://cdn.sofifa.com/players/	Brazil	https://cdn.sofifa.com/flag	72	90	Real Madrid	https://cdn.sofifa.com/tear	â‚¬6M	â‚¬17K	Reinier Jesus Carvalho	Below 20 year
Trent Alexander-Arnold	20	https://cdn.sofifa.com/players/	England	https://cdn.sofifa.com/flag	85	90	Liverpool	https://cdn.sofifa.com/tear	â‚¬46M	â‚¬93K	Trent Alexander-Arnold	Age: 20-25
Federico Valverde	20	https://cdn.sofifa.com/players/	Uruguay	https://cdn.sofifa.com/flag	82	90	Real Madrid	https://cdn.sofifa.com/tear	â‚¬32.5M	â‚¬125K	Federico Valverde	Age: 20-25

Figure 11: Dataset

The Attributes has the following property: -

Feature	Data Type	Description
Name	String	Contain the names of the players.
Age	Int	Contain the age of each player.
Photo	String	It contains the link corresponding to the players' image.

Nationality	String	Contains players Nationality.
Flag	String	Contain link of players' Country image.
Overall	Int	This feature tells the overall score of the player.
Potential	Int	It tells the potential of the player.
Club	String	It contains the club name of the players.
Club Logo	String	It contains the link to the logo of players' club.
Value	String	Signify the total value of the player.
Wages	String	Tells the salary of the player.
Full Name	String	Contains the full name of the player.
Age Group	String	Tells about the age group of players.

Table 4:Dataset Description.

Apart from this. A complete directory for the image of the players, their country flag and the club logo stored in jpg format named according to players' name.

3.3.2. Players' Images

This Directory contain the image of the players in .jpg format. This Directory contains 1000 images of 1000 players. These are the players who played FIFA 2020 or in the selected team. Each image is in ration 1:1 and has dimension of 60*60 px with same background. Image contain the front face of the player in portrait form. Each image has named according to the players' full name. so that It can be directory called dynamically to the webpage.

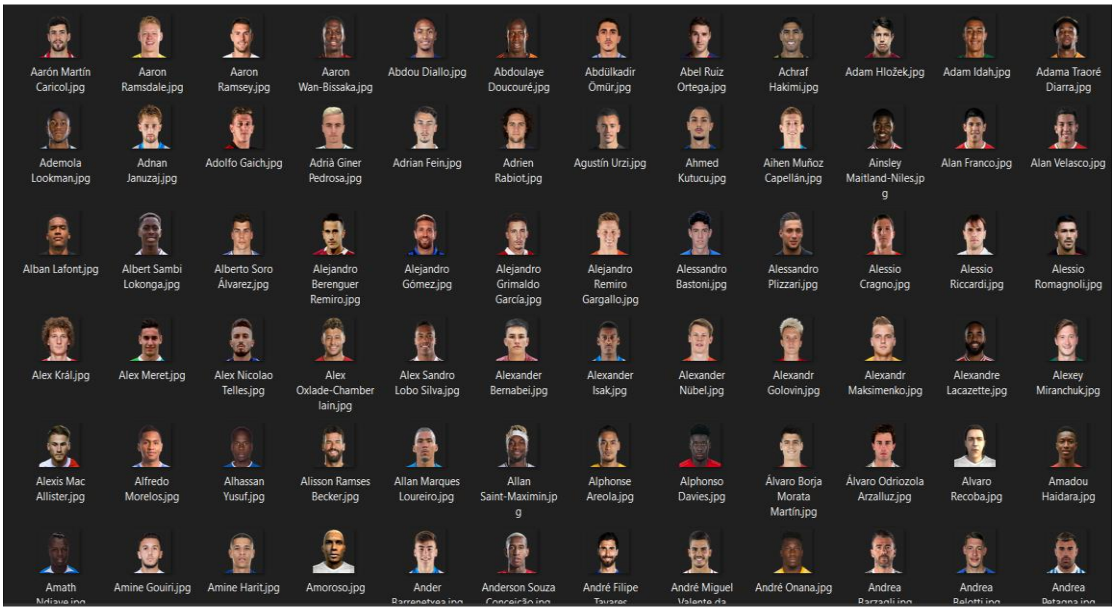


Figure 12: Players' image Directory

3.3.3. Club images

This Directory contain the clubs' image of the players in .jpg format. This Directory contains 1000 images of 1000 players. Each image is in ratio 1:1 and has dimension of 30*30 px with same background. Image contain the Club Logo of the player. Each image has named according to the players' name.

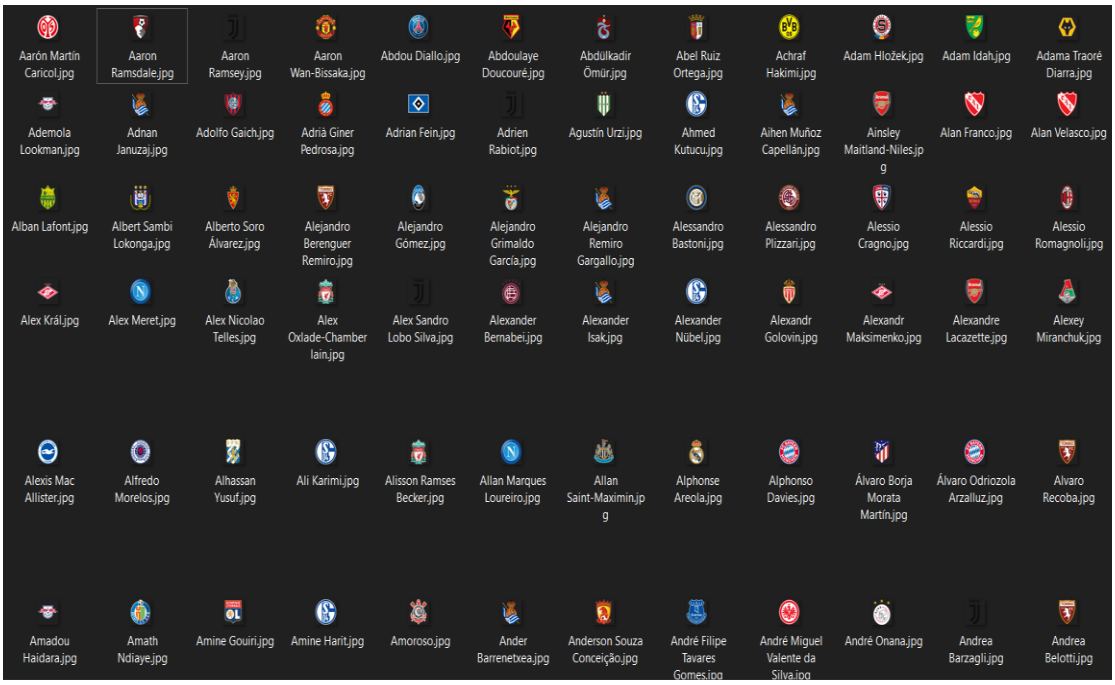


Figure 13: Club Logo Directory

3.3.4. Country Flag

This Directory contain the National flag of the players in .jpg format. This Directory contains 1000 Flag images of 1000 players. Each image is in ratio 4:3 and has dimension of 32*24 px with same background. Image contain the Flag of the player. Each image has named according to the players' name.

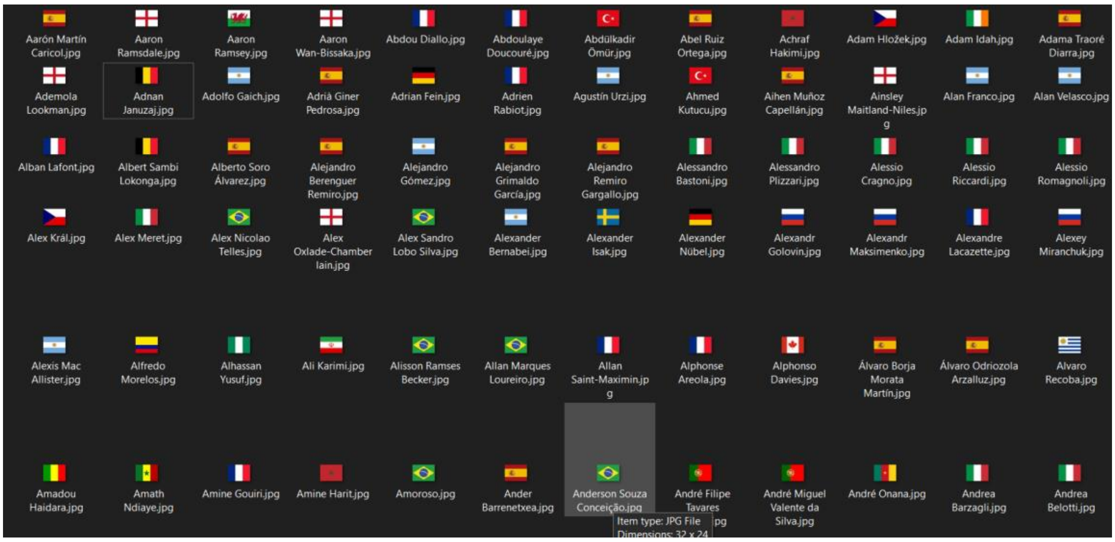


Figure 14:National Flag Directory

3.3.5. Entity Relationship Diagram

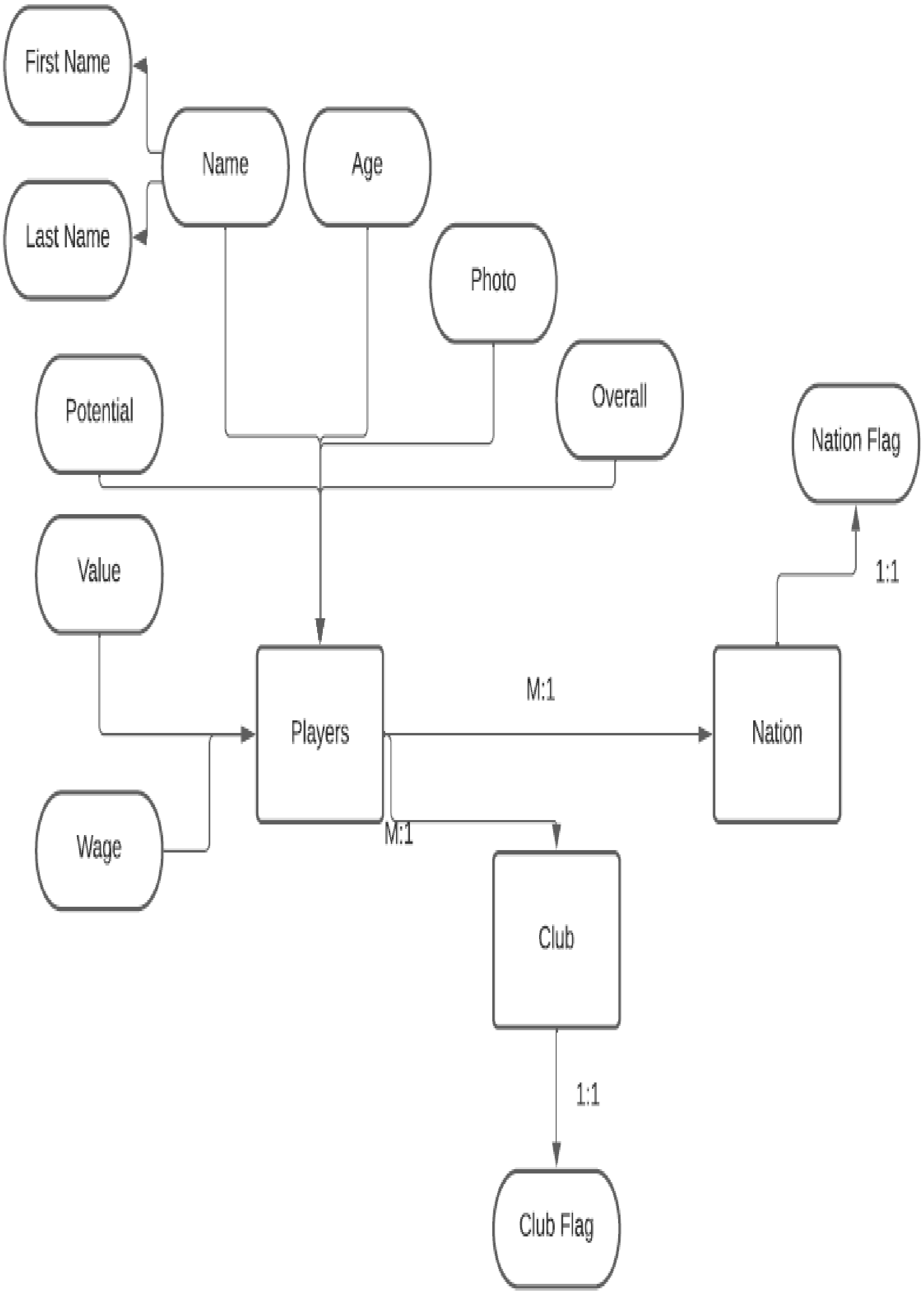


Figure 15:ER Diagram

4. TESTING

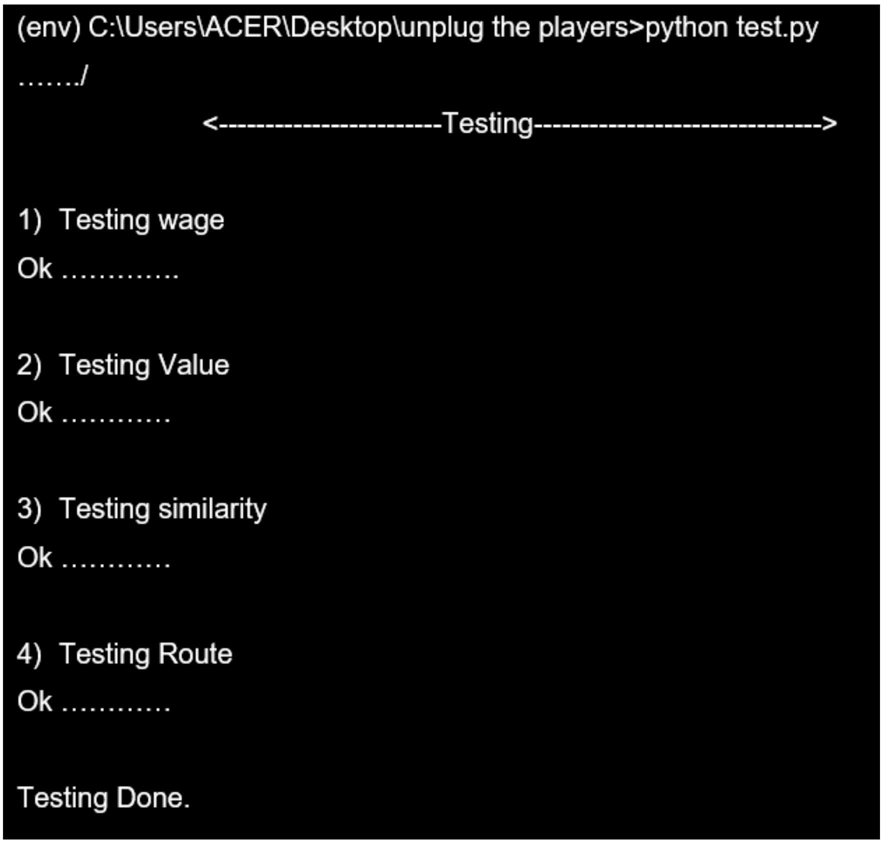
4.1. TEST PLAN

UNIT Testing: - Unit testing is a software development process in which the smallest testable parts of an application, called units, are individually and independently scrutinized for proper operation.

Unit testing is often automated but it can also be done manually. A unit test is an automated piece of code that invokes a unit of work in the system and then checks a single assumption about the behavior of that unit of work.

In this application, a manually written unit test script method is used for testing function those perform unit amount of work and provides functionality.

4.2. TEST REPORT



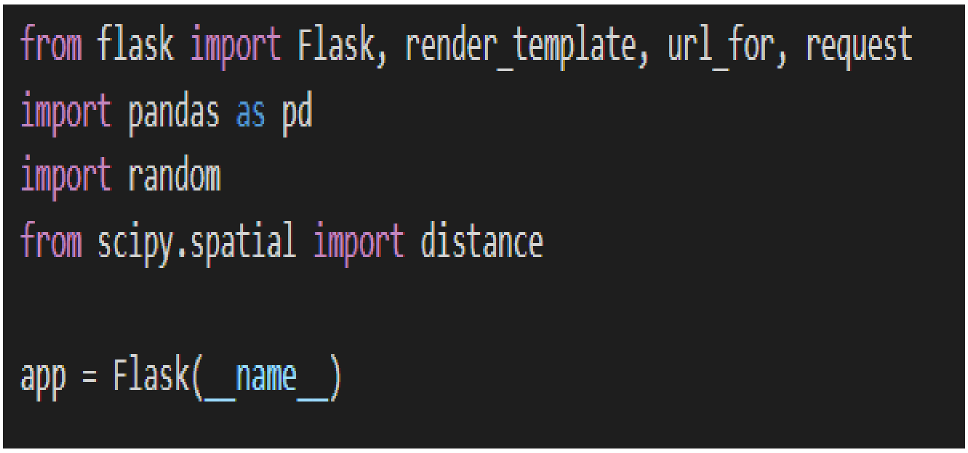
```
(env) C:\Users\ACER\Desktop\unplug the players>python test.py
……./
                    <-----------------------Testing----------------------------->

1)  Testing wage
Ok ………….

2)  Testing Value
Ok …………

3)  Testing similarity
Ok …………

4)  Testing Route
Ok …………

Testing Done.
```

Figure 16: Testing Report

5. IMPLEMENTATION AND USER INTERFACES

5.1. FLASK APP (App.py)

This will contain the base of our application. In this, we have created an app object, which is an instance of the Flask object. It'll act as the central configuration object for the entire application. It's used to set up pieces of the application required for extended functionality, e.g., a database connection and help with authentication. It will also set up the routes that will become the application's points of interaction. To explain what this means, let's look at the code it corresponds to.



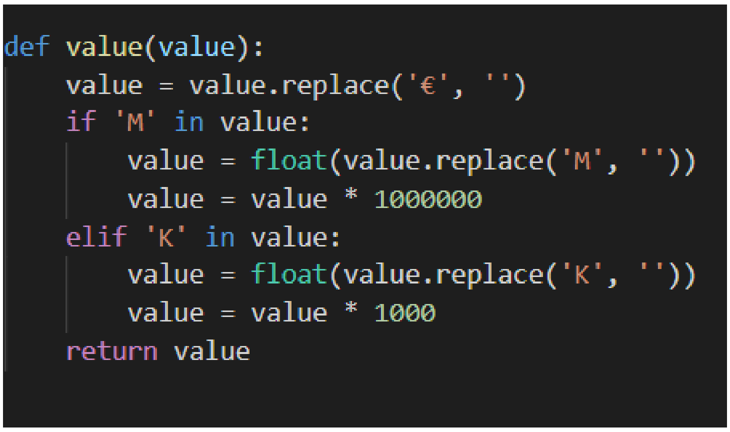
```
from flask import Flask, render_template, url_for, request
import pandas as pd
import random
from scipy.spatial import distance

app = Flask(__name__)
```

Figure 17:Flask Object

This is the most basic initialization of Flask application. app is an instance of Flask, taking in the __name__ of the script file. This lets Python know how to import from files relative to this one. The app. route decorator decorates the first view function; it can specify one of the routes used to access the application.

5.1.1. <u>Functions in Flask app</u>

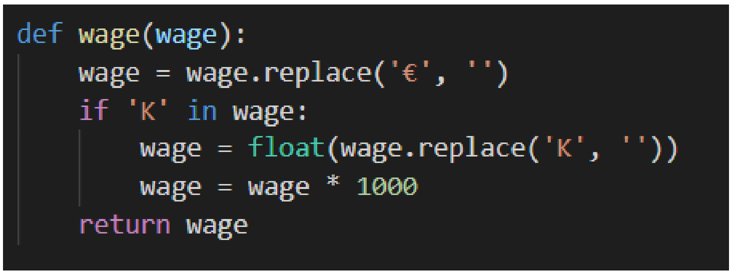
VALUE: - This function will be used to clean the value of the players. In our dataset, value of the players are available in the form of String to represent the value in pound and million. So, Our Function Value will accept parameter ‘List of values of the players’ of the player and will return a list of integers corresponding to the amount of each player. This is how the function looks like.



```python
def value(value):
    value = value.replace('€', '')
    if 'M' in value:
        value = float(value.replace('M', ''))
        value = value * 1000000
    elif 'K' in value:
        value = float(value.replace('K', ''))
        value = value * 1000
    return value
```

Figure 18:Value Function

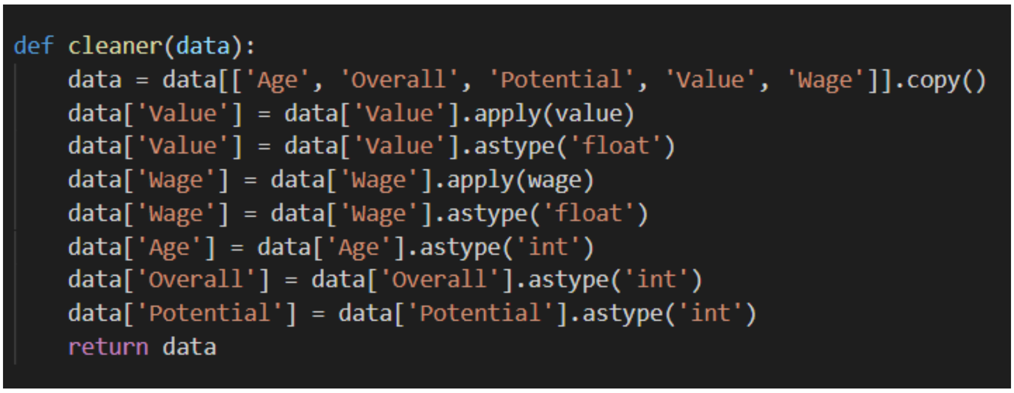
Wage: - This function will be used to clean the wage of the players by removing unwanted characters from the players wages and will convert them into string format. This Function will Accept a list of strings and will return a list of integers. Here the function actually looks like.



```python
def wage(wage):
    wage = wage.replace('€', '')
    if 'K' in wage:
        wage = float(wage.replace('K', ''))
        wage = wage * 1000
    return wage
```

Figure 19:Wage Function

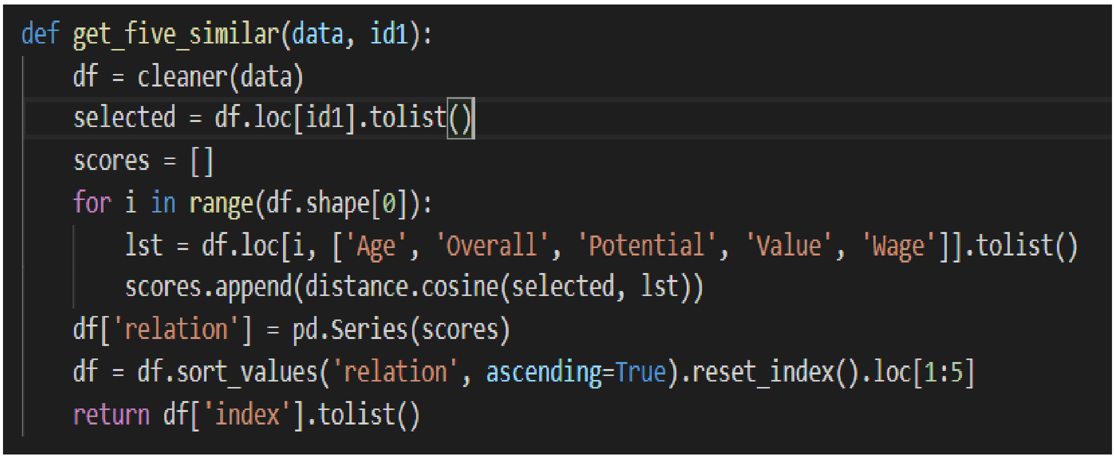
Cleaner: - This Function will be used to clean the data coming from the dataset using the above-mentioned functions and will return a clean data. This function will be the base for getting every attribute in their proper way. This is the code will look like.



```python
def cleaner(data):
    data = data[['Age', 'Overall', 'Potential', 'Value', 'Wage']].copy()
    data['Value'] = data['Value'].apply(value)
    data['Value'] = data['Value'].astype('float')
    data['Wage'] = data['Wage'].apply(wage)
    data['Wage'] = data['Wage'].astype('float')
    data['Age'] = data['Age'].astype('int')
    data['Overall'] = data['Overall'].astype('int')
    data['Potential'] = data['Potential'].astype('int')
    return data
```

Figure 20: Cleaner Function

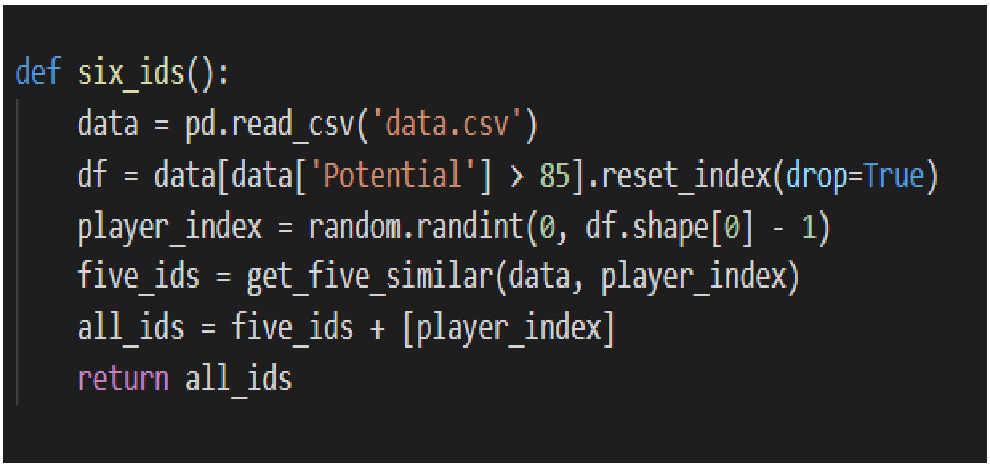
get_five_similar: - This Function will pick five similar players from the dataset as per their id's. To find the similarity, we have measured Spatial Distance and used cosine similarity between different attributes of the players. And will be selecting the 5 players based on the closeness in their property. To find the spatial distance between players property we have imported a predefined library from SciPy. From SciPy, we have used distance function available in spatial module. Cosine similarity measures the similarity between two vectors of an inner product space. It is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in roughly the same direction. Here is how function look like.



```python
def get_five_similar(data, id1):
    df = cleaner(data)
    selected = df.loc[id1].tolist()
    scores = []
    for i in range(df.shape[0]):
        lst = df.loc[i, ['Age', 'Overall', 'Potential', 'Value', 'Wage']].tolist()
        scores.append(distance.cosine(selected, lst))
    df['relation'] = pd.Series(scores)
    df = df.sort_values('relation', ascending=True).reset_index().loc[1:5]
    return df['index'].tolist()
```

Figure 21:get Five similar Function

Get_Id: - This Function will pick the ids from the dataset available in form of csv file. And based on potential of the player, it will randomly pick some ids. It will accept the whole data file as a parameter to work on. This is how the function looks like.



```python
def six_ids():
    data = pd.read_csv('data.csv')
    df = data[data['Potential'] > 85].reset_index(drop=True)
    player_index = random.randint(0, df.shape[0] - 1)
    five_ids = get_five_similar(data, player_index)
    all_ids = five_ids + [player_index]
    return all_ids
```

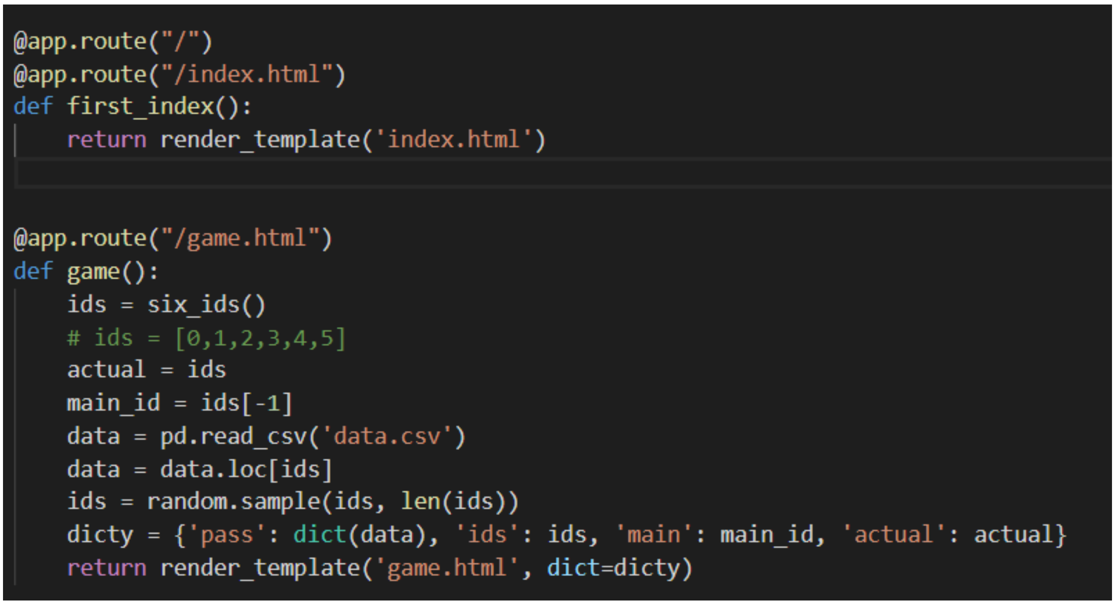
Figure 22: Getid Function

5.1.2. App Route and Rendering

To render the application in form of a web page, flask rendering templates have been import from flask library. Using that user will be able to render to home page as well as game page of the app. This will be easy for user easily navigate between pages.

To give the application a basic route of working, app.route will make it possible to find the way for application to follow. Each route will render to a web page where user can interact with.

This is how the code looks like.



```
@app.route("/")
@app.route("/index.html")
def first_index():
    return render_template('index.html')


@app.route("/game.html")
def game():
    ids = six_ids()
    # ids = [0,1,2,3,4,5]
    actual = ids
    main_id = ids[-1]
    data = pd.read_csv('data.csv')
    data = data.loc[ids]
    ids = random.sample(ids, len(ids))
    dicty = {'pass': dict(data), 'ids': ids, 'main': main_id, 'actual': actual}
    return render_template('game.html', dict=dicty)
```

Figure 23: App route and rendering

5.2. HOME PAGE

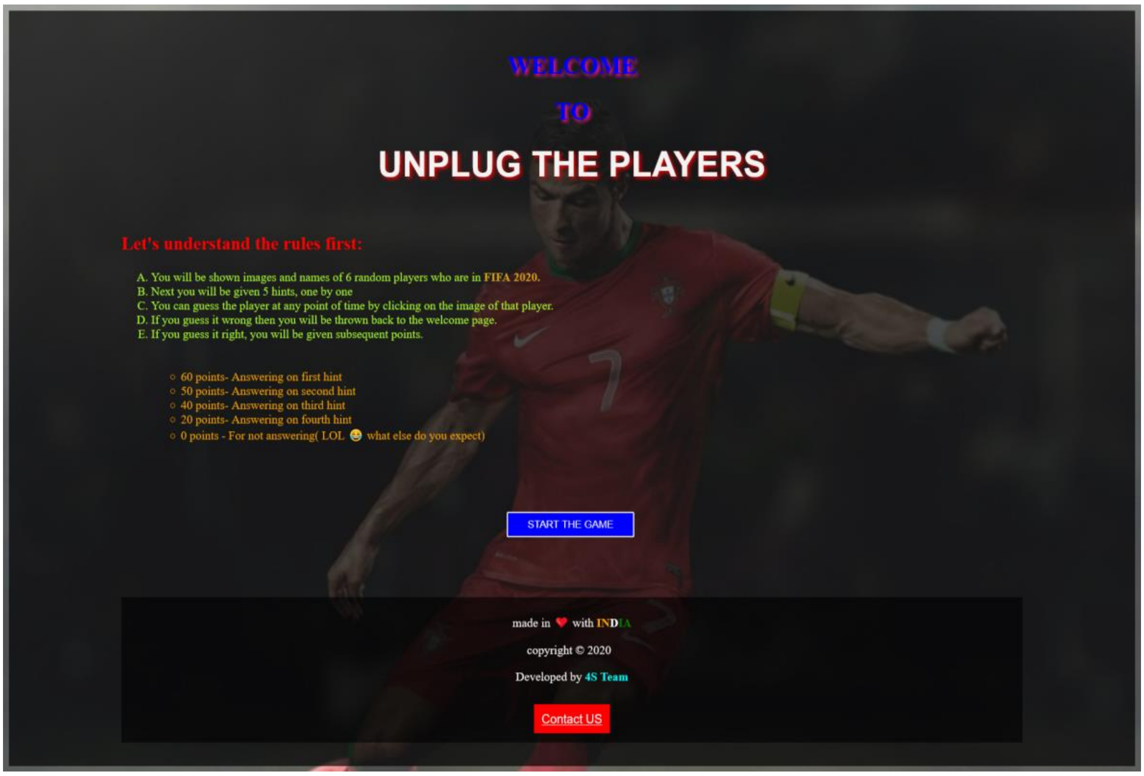
This is the first page which will be visible to the user. This page was designed to provide information to the users about the rules and regulation of the web page. Home page has a name of the game with bouncing effect to make the web page attractive and stylish so that user may found it attractive.

Number of css properties like animation, bouncing effect, hovering effect, media queries and JavaScript function like typing effect are used backend to make it more user friendly. This page will inform user about all the rules enlisted and every details user should know. Like what is the game all about. How they will be given points and what are the different points allotted after each hint.

The home page is having a start button to make the user enter to the game page where the actual game will start from.

Along with all these details, this page will inform user about the developers and for any bug/ suggestion how they can contact us in the lower of the web page.

This is how the home page should appear to the user.



WELCOME

TO

UNPLUG THE PLAYERS

Let's understand the rules first:

A. You will be shown images and names of 6 random players who are in FIFA 2020.
B. Next you will be given 5 hints, one by one
C. You can guess the player at any point of time by clicking on the image of that player.
D. If you guess it wrong then you will be thrown back to the welcome page.
E. If you guess it right, you will be given subsequent points.

- 60 points- Answering on first hint
- 50 points- Answering on second hint
- 40 points- Answering on third hint
- 20 points- Answering on fourth hint
- 0 points - For not answering(LOL 😂 what else do you expect)

made in ❤ with INDIA

copyright © 2020

Developed by 4S Team

Figure 24: Home page

5.3. GAME PAGE

This page will be the user interface for playing the game. This page will display pictures of six players along with their names using the python function in backend. All the players will have similarity between them.

Game page will also show the user start taking the hint to proceed and based on the hint's user will select a player. Depending on the correct/ incorrect guess, they will be given points as per the number of hint and rules given on the home page of the game.

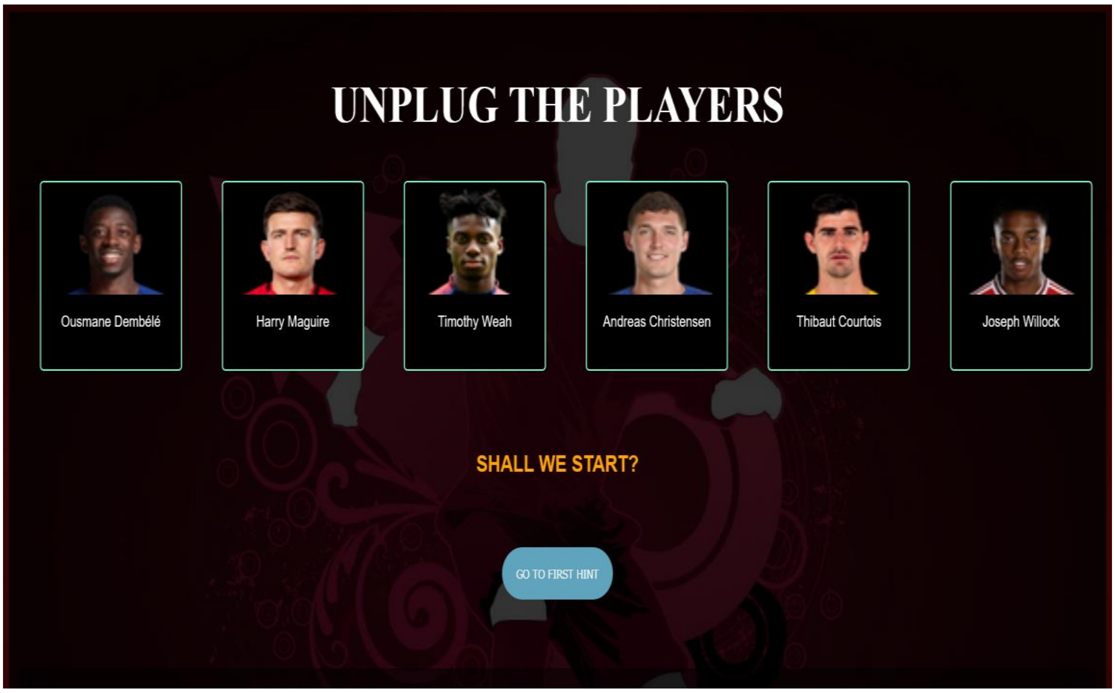
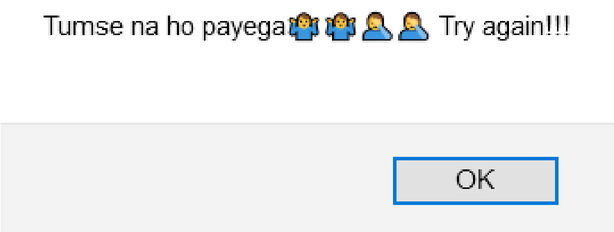


Figure 25: Game page

Based on the correct/ incorrect selection, user will be shown an alert box where result and points obtained by the user will be shown. With each time this page will refresh itself and will show different images to the user based on similarity.

This page will keep on refreshing until user want to close the application. This page is highly dynamic and number of css properties, JavaScript function as well as python function is working backend to give user a better experience than some other quiz-based games.



Tumse na ho payega🤷🤷🤦🤦 Try again!!!

Figure 26: Result box

5.3.1. Implementation of Game page.

As the game is loaded, the user can see pictures along with the names of six players respectively and also a button to start displaying the respective hints one by one. There are total five hints for the user and for each correct hint some point is rewarded to the user as mentioned in index page.

We have used six variables for hints, six for buttons including the one to start showing the hints and six for every player displayed.

Variables hint0, hint1, hint2, hint3, hint4, hint5 holds a reference to the object with the id zero hint, first hint, second hint, third hint as so on respectively.

Similarly, btn0, btn1, btn2, btn3, btn4, and btn5 holds the reference to the buttons having id next_hint0, next_hint1 and so on. Similar thing is done with cards having the pictures and names of the players whose reference is stored in variables card1, card2, card3, card4, and card5 and card6.

when the first button having value "GO TO FIRST HINT" and id as 'next_hint0'(whose reference is stored in btn0) is clicked (btn0.addEventListener('click',function())) , then display of the block having id as 'zero_hint'(reference stored in hint0) is changed to 'none' (hint0.style.display= 'none';) i.e. now it is not displayed on the screen and that block is replaced by block having id 'first_hint' as its display property is changed to 'block' (hint1.style.display= 'block';).

Similarly, in case when next button is clicked, the block having id 'second_hint' is displayed and the block which was already there gets replaced.

This process ends when user reaches the last hint, an alert box is displayed and the same page is reloaded (location.reload()).

If the user guesses the player in the first step, it means the condition shown below is true

hint1.style. display == 'block'

an alert box is displayed congratulating the player and showing the points he had earned.

Similarly, for a correct guess at any step similar condition is there in the else if part of code for example if user rightly guesses the player in fourth step then the condition below:

hint4.style. display=='block'

is true and the respective alert box is displayed followed by reloading the game page for a fresh game:

alert('Took some time.But well done. Points Earned: 20');

When the player and user guesses is the wrong one, In that case user would be clicking on any one of the rest five cards and an event for the corresponding card is triggered.

An alert box is displayed having information that the user has made a wrong guess and he should try again followed by reloading the game page.

For example, suppose the user clicked on player 2 and suppose its card no is card2 and he is not the right player according to the hints so the function with card 2 will be called and will make the alert for the user.

When there is no hint on web page.

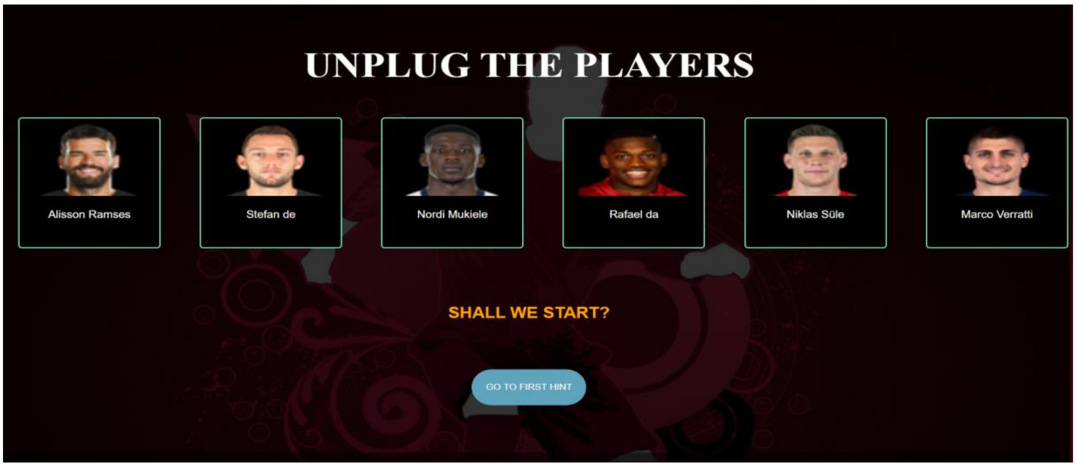
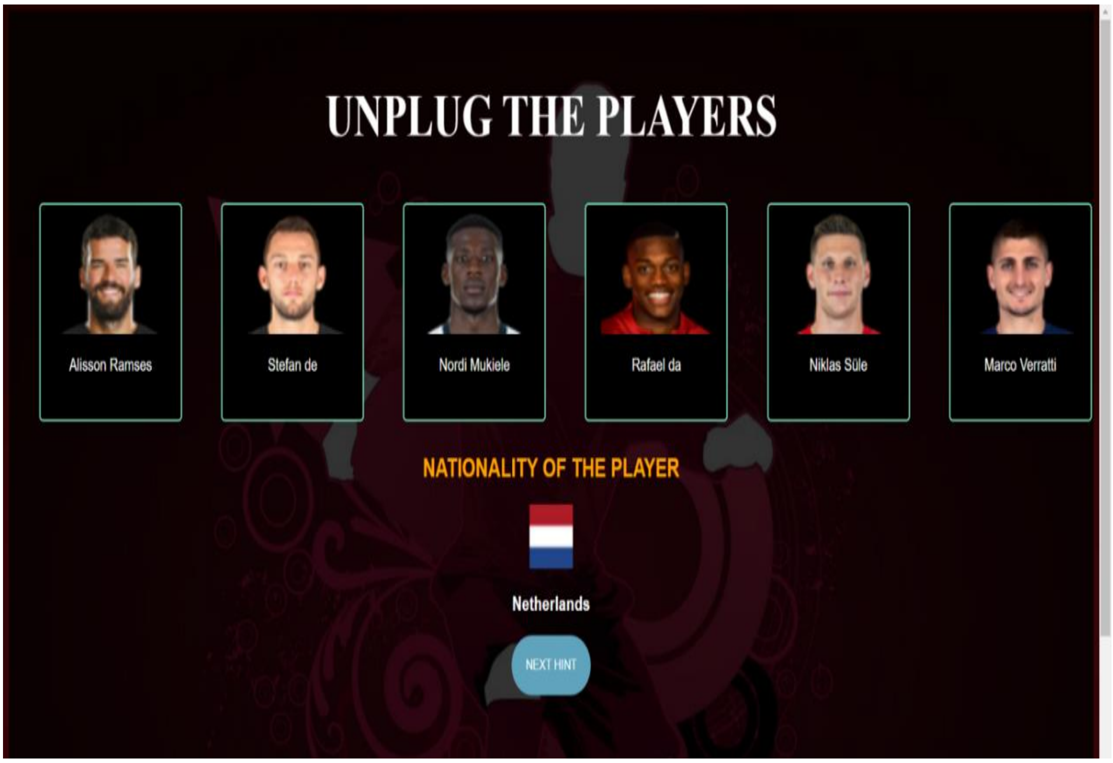


Figure 27:Game page without hint

When ‘GO TO FIRST HINT’ button is pressed.



UNPLUG THE PLAYERS

Alisson Ramses

Stefan de

Nordi Mukiele

Rafael da

Niklas Süle

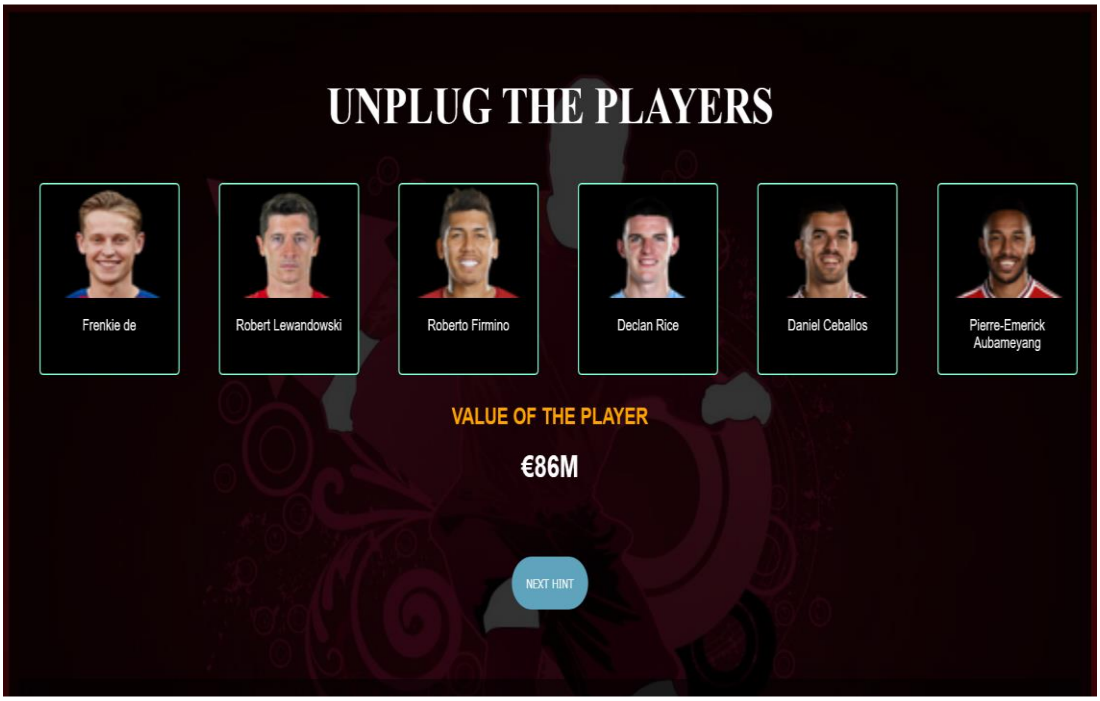
Marco Verratti

NATIONALITY OF THE PLAYER

Netherlands

Figure 28:Game page 1st hint

Second hint of the game.



UNPLUG THE PLAYERS

Frenkie de

Robert Lewandowski

Roberto Firmino

Declan Rice

Daniel Ceballos

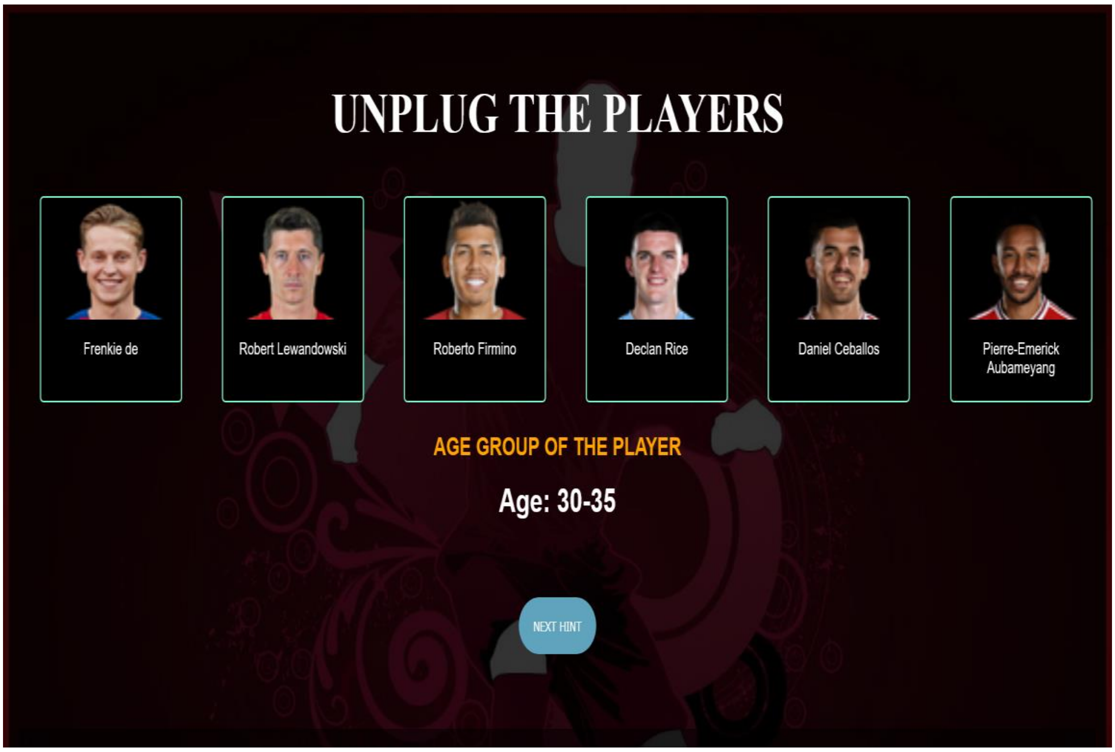
Pierre-Emerick Aubameyang

VALUE OF THE PLAYER

€86M

Figure 29:Game page 2nd hint

Game Page with third hint.



UNPLUG THE PLAYERS

Frenkie de

Robert Lewandowski

Roberto Firmino

Declan Rice

Daniel Ceballos

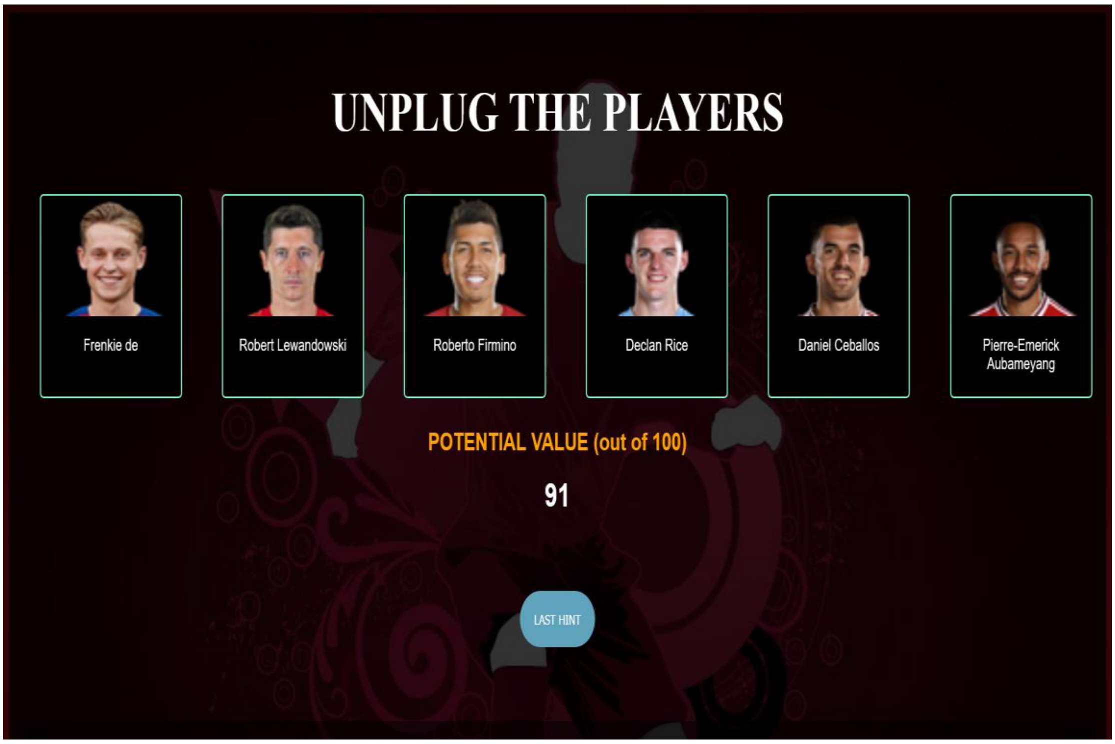
Pierre-Emerick Aubameyang

AGE GROUP OF THE PLAYER

Age: 30-35

Figure 30:Game page 3rd hint

Game page with 4th hint.



UNPLUG THE PLAYERS

Frenkie de

Robert Lewandowski

Roberto Firmino

Declan Rice

Daniel Ceballos

Pierre-Emerick Aubameyang

POTENTIAL VALUE (out of 100)

91

Figure 31:Game page 4th hint

Game page with 5th hint.

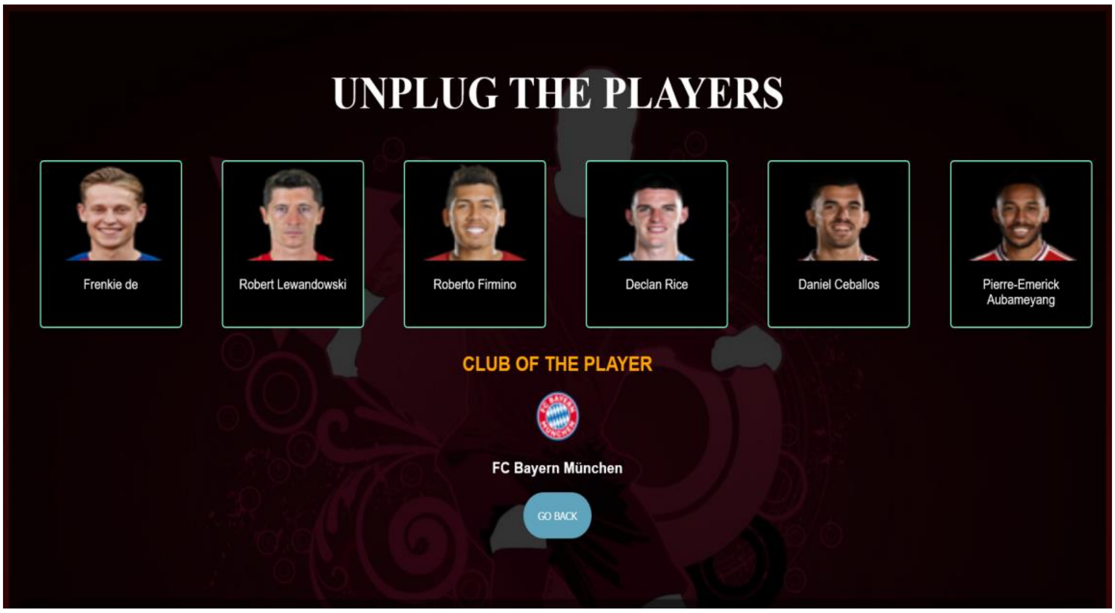
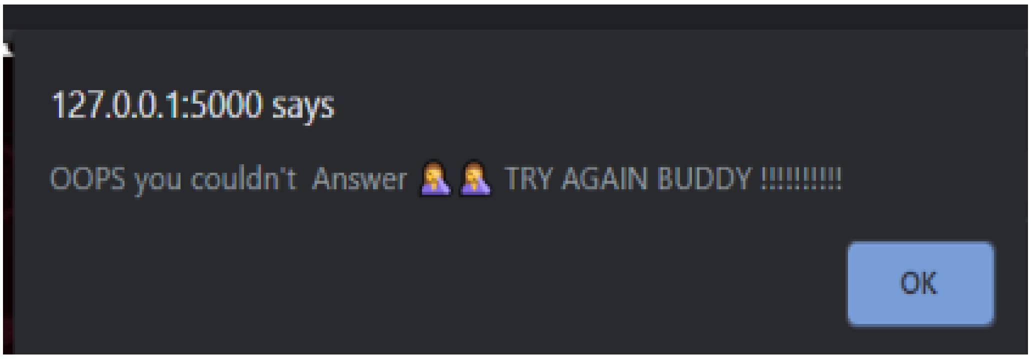


Figure 32:Game page 5th hint

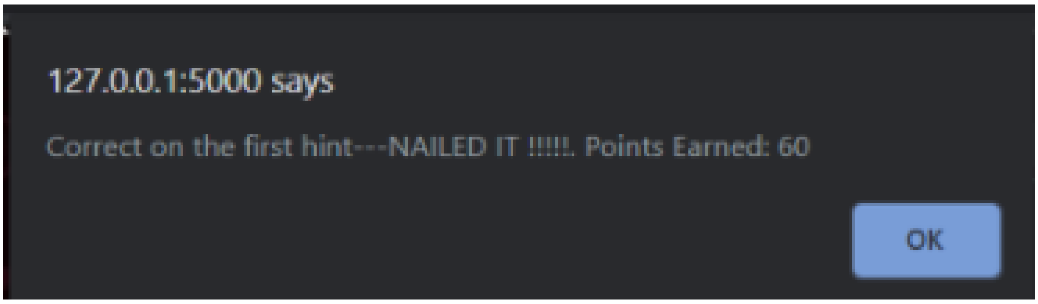
When all the hints are already exhausted :


127.0.0.1:5000 says

OOPS you couldn't Answer 🤦‍♀️🤦‍♀️ TRY AGAIN BUDDY !!!!!!!!!

Figure 33:Exhausted hint box

When user guess correctly After first hint


127.0.0.1:5000 says

Correct on the first hint---NAILED IT !!!!!. Points Earned: 60

Figure 34:Correct on first hint box

5.4. DEPLOYMENT

The flask app is deployed on Heroku. Heroku is a cloud platform as a service (PaaS) supporting several programming languages. One of the first cloud platforms, Heroku has been in development since June 2007, when it supported only the Ruby programming language, but now supports Java, Node.js, Scala, Clojure, Python, PHP, and Go. For this reason, Heroku is said to be a polyglot platform as it has features for a developer to build, run and scale applications in a similar manner across most languages.

To deploy this flask app on Heroku. We had created a python virtual environment in the application directory and installed all the dependencies in that environment. We need to add the requirement.txt file to the root directory by command

pip freeze > requirements.txt

In order for us to successfully deploy any application to Heroku, we must add a *Procfile* to that application. Before we can add a Procfile, we need to first install a web server called Gunicorn. Run the following command within the application folder.

pip install gunicorn

Update the requirements file by running

pip freeze > requirements.txt

Create a new file with Procfile as the name and do not add any extension. Add this line below

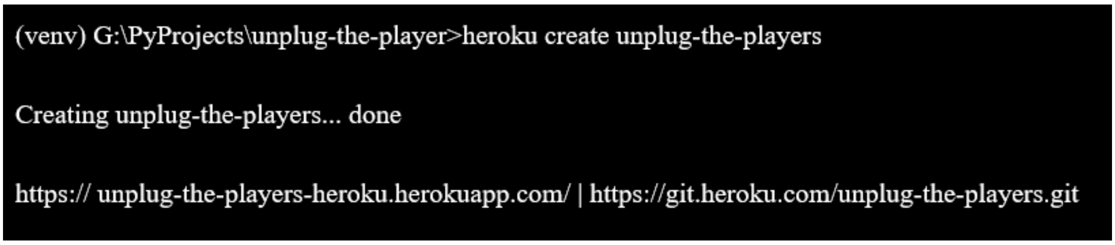
web: gunicorn app:app

web is used by Heroku to start a web server for the application. The *app:app* specifies the module and application name. In our application we have the app module and our flask application is also called app.

Now we will create a Heroku app using command.

heroku create unplug-the-players

unplug-the-players is the name of the application, this has to be unique across Heroku.



```
(venv) G:\PyProjects\unplug-the-player>heroku create unplug-the-players

Creating unplug-the-players... done

https:// unplug-the-players-heroku.herokuapp.com/ | https://git.heroku.com/unplug-the-players.git
```

Figure 35: Heroku app creation

The https://git.heroku.com/unplug-the-players.git is the Heroku git remote repository where our application lives on Heroku. We now have to push our application to the master branch of the above git URL using

git push Heroku master

5.4.1. How to Run the app on local:

1. Open the Terminal.
2. Clone the repository by entering

   $ git clone https://github.com/shsarv/UNPLUG-THE_PLAYER.git.

3. Ensure that Python3 and pip are installed on the system.
4. change the directory to repository name using

   $ cd [Repository name].

5. Create a python virtual environment by executing the following command:

   virtualenv mygame.

6. Activate the mygame virtual environment by executing the following command:

   source env/bin/activate.

7. Enter the cloned repository directory and execute

```
pip install -r requirements.txt.
```

8. Now, execute the following command: flask run and it will point to the localhost server with the port 5000.
9. Enter the IP Address: http://localhost:5000 on a web browser and use the application.

6. CONCLUSION AND FUTURE WORK

6.1. CONCLUSION

The main purpose of our project is to develop an application that offers new aspects of learning and improving knowledge in educational area. Most of the available apps are entertainment-based, which mostly do not contribute to the academic enhancement of the students. The theme of our quiz is to provide user to practice for understanding games and information, so in this app we focus sport. This quiz is useful for enhancing knowledge for sports.

This quiz includes three 5 functions which are collectively named as hints that help users to answer questions correctly. These functions can be used only once by a user. When user tap/click on "Hint", it will show user the information regarding the correct player. It goes on the dept of 5, then user can easily select an option from the six. We have learned a lot about Flask API and SciPy library in python and also about user's behavior. We have found that the data collection process is hard and time-consuming, but it can be managed by a team work. We hope that other developers will take advantage from our experience/from our development.

6.2. FUTURE WORK

We are planning to keep managing the project and improving it based on user feedback. Here is our to do list for future

- We will add some more categories in our app.
- We'll try to make it more user friendly than it is now.
- We'll try to improve its quality.
- We'll work on another feature in our app to add a module namely adding other sport games quiz which is helpful for everyone.

7. REFERENCES

1. https://www.tutorialspoint.com/flask/index.htm
2. https://docs.scipy.org/doc/scipy/reference/spatial.distance.html
3. https://flask.palletsprojects.com/en/1.1.x/deploying/
4. https://www.freecodecamp.org/news/how-to-build-a-web-application-using-flask-and-deploy-it-to-the-cloud-3551c985e492/
5. https://pandas.pydata.org/
6. https://www.w3schools.com/js/default.asp
7. https://www.coursera.org/learn/javascript
8. https://www.coursera.org/projects/python-flask
9. https://www.coursera.org/specializations/web-design

APPENDIX

Source Code –

- Unplug the Players app:
  https://github.com/shsarv/UNPLUG-THE-PLAYER

- Deployment Link:
  https://unplug-the-players.herokuapp.com/