



## Web Development Project Requirements

### Egyptian Premier League Match Reservation System

#### Objectives:

1. ILO1: Practice fundamental web design theories.
2. ILO2: Develop a multi-user and a multi-authority system.
3. ILO3: Practice front-end and back-end web development.

#### Introduction:

In this project, you are a web designer and you will create a web application requested by a client. The client specifies the required business model and the main features. You will be free to design the website hierarchy and the suitable user interface.

**An online automated ticket reservation system for football matches in the Egyptian Premier League** is requested by the client. The system allows the EFA (Egyptian Federation Association) management to create, manage, and maintain current and future matches. The system should allow the users to reserve specific seats for the matches they want to attend.

You are required to **design** and **develop** an automated system for the reservation process. The system is used by the EFA management to keep track of matches schedules, number of attendees, match officials and the registered users.

#### Business Model:

The EFA Management provided you with the following description of each user:

<b>Site Administrator</b>	The <i>Site Administrator</i> is the IT engineer who manages the users' creation and website authorities.
<b>EFA Management</b>	The <i>EFA (Egyptian Federation Association) managers</i> are responsible for managing, creating, and modifying the details of matches. The details of each match include information about the home team, away team, match venue and start time.
<b>Customers (Fans)</b>	A <i>customer</i> is the user that has registered their personal data as a spectator or a fan with the purpose of attending league matches. They can reserve any number of tickets for any match as long as these matches are not clashing.
<b>Guest</b>	They are <i>unregistered users</i> or users who have not logged in yet. They can log in, register (sign up), and/or view the current match details.

Each match should have the following **details**:

1. Home Team. (One of possible 18 teams)
2. Away Team. (One of possible 18 teams, should not be the same as the home team).
3. Match Venue (One of the stadiums approved by the EFA managers)
4. Date & Time.
5. Main Referee.
6. Two Linesmen.

Each user should have the following **personal data**:

1. Username (**Must be Unique**)
2. Password
3. First Name
4. Last Name
5. Birth Date
6. Gender
7. City
8. Address (Optional and could be skipped by the user).
9. Email Address.
10. Role: Manager/ Fan

### **Functionality Requirements:**

The main features required are:

#### **Site Administrator:**

Feature	Description
F1: Approve new users as an authority.	New Users should sign-up & the user account is given an authority when the administrator approves their authority.
F2: Remove an existing user.	The administrator can remove an existing account.

#### **EFA Managers:**

Feature	Description
F3: Create a new match event	The EFA managers can create a new match event and add all its details.
F4: Edit the details of an existing match.	The EFA managers can change/edit the details of a certain match.
F5: Add a new stadium.	The EFA managers can add a new stadium and define its shape and number of seats as shown below.
F6: View match details	The EFA managers can view matches details.
F7: View vacant/reserved seats for each match.	The EFA managers can view the overall seat status for each event (vacant/reserved).

**Customers (Fans):**

Feature	Description
F8: Edit their data.	The customer can edit their personal data (except for the username and email address).
F9: View matches details	The customer can view all matches details as well as the vacant seats for each match.
F10: Reserve vacant seat(s) in future matches.	The customer can select vacant seat/s only. The customer is asked to enter a credit card number & its pin number. Then the reservation is confirmed and a reservation ticket number (unique) is generated.
F11: Cancel a reservation	The customer can cancel a reserved ticket only 3 days before the start of the event. The seat/s in the reservation should be vacant again.

**Guests:**

Feature	Description
F12: Register a new account.	The guest can register a new account (whether as a fan or as a manager).
F13: Sign in as an existing account	The guest can log in as an existing account.
F14: View matches details (same as F9)	The guest can view matches details (whether signed in or not).

**Additional required specifications for any user type:****1. Stadium shape:**

Since the matches are played behind closed doors. The fans will only be seated in the VIP lounge. The lounge shape is rectangular, where the length and width is defined by the number of rows in the VIP area and the number of seats per row respectively.

**2. Reserving a specific seat graphically:**

The whole VIP area seats should be visible to the user and the vacant/reserved seats should be clearly distinguishable.

**3. Reservation Conflicts:**

A specific seat is reserved to a single user.

For example, if two users try to reserve a specific seat at the same time, the system should reserve the seat to only one of them.

**4. Automatic seats status update:**

The seats status should be updated automatically when the user is in the reservation view. i.e. if another client reserved a seat, it should be displayed to the user who is currently opening the reservation view.

**The main web pages that must exist:**

1. Website Home Page (The main page for the website)
2. Login Page (A page where the user should enter their username & password)

**You are free to use the following frameworks for front-end and back-end respectively:**

**Front End:** AngularJS, React, or any other framework.

**Back End:** Laravel, Node.js, or any other framework.

**For CSS and styling:** Bootstrap, or any other framework.

**Teams**

Each team consists of **3 or 4** members.

**Final Delivery & Deadline:**

The deadline will be **on Week 12**

You should submit all the files you used in a neat and an organized hierarchy.

**- Note: Plagiarized projects from the internet will be graded zero without tolerance.**