

# PLAYER MANAGEMENT SYSTEM

## Project Documentation

---

### COVER PAGE

**Student Name:** Irabaruta Willy NORBERT

**Registration Number:** 27463

**Course:** Web Design

**Project Title:** Player Management System

**Academic Year:** 2024/2025

**Date:** August 2025

**Institution:** AUCA

**Department:** Software Engineering

---

## TABLE OF CONTENTS

1. <u>Project Introduction</u> .....	2
2. <u>Problem Statement</u> .....	2
3. <u>System Requirements</u> .....	3
4. <u>System Design</u> .....	3
5. <u>Implementation</u> .....	4
6. <u>Database Design</u> .....	4
7. <u>Testing</u> .....	5
8. <u>Challenges Faced</u> .....	5
9. <u>Conclusion</u> .....	6

---

## 1. PROJECT INTRODUCTION

### Title

**Player Management System** - A web-based application for managing football players.

### Case Study

This system is built for **football club managers** who need to manage their player rosters and **general public** who want to view player information.

## Purpose

The system solves key problems in sports management:

- Eliminates scattered player records
- Provides public access to player information
- Offers secure management for authorized users
- Ensures data consistency and professional presentation

## Technologies Used

- **HTMLS:** Structure and semantic markup
  - **CSS3:** Styling and responsive design
  - **JavaScript:** Client-side validation and interactions
  - **PHP:** Server-side logic and form processing
  - **MySQL:** Database management and storage
- 

## 2. PROBLEM STATEMENT

### The Problem

Traditional player management relies on paper-based records that are easily lost, disconnected systems with inconsistent data, and limited accessibility requiring physical presence to view information.

### Target Users

- **Team Managers:** Need to add, edit, and remove players from their teams
- **General Public:** Want easy access to current player information without registration
- **Club Administrators:** Require oversight of player data

### Key Features

#### For Managers:

- Secure login with session management
- Complete CRUD operations (Create, Read, Update, Delete)
- Photo management for players
- Personal dashboard showing managed players

#### For Public Users:

- Clean homepage displaying all players
  - No registration required for viewing
  - Mobile-responsive design
- 

### 3. SYSTEM REQUIREMENTS

#### Software Requirements

- **Code Editor:** Visual Studio Code
- **Local Server:** XAMPP (Apache, MySQL, PHP)
- **Web Browser:** Chrome, Firefox, Safari, or Edge
- **Database:** MySQL 5.7+

#### Hardware Requirements (Optional)

- **Processor:** Intel i3 or equivalent
  - **RAM:** 4GB minimum
  - **Storage:** 2GB free space
- 

### 4. SYSTEM DESIGN

#### User Flow

```
[Homepage] → [View Players (Public)]
    ↓
[Manager Login] → [Dashboard] → [Add/Edit/Delete Players]
    ↓           ↓
[Register]     [Logout] → [Homepage]
```

#### Page Structure

##### Public Pages:

- `index.php` - Homepage displaying all players
- `login.php` - Manager login
- `register.php` - Manager registration

##### Protected Pages:

- `dashboard.php` - Manager control panel

- `add-player.php` - Add new player
- `edit-player.php` - Edit existing player
- `delete-player.php` - Remove player

### Supporting Files:

- `config.php` - Database configuration
- `style.css` - Complete styling
- `scriptjs` - Frontend validation

### Navigation

- **Public:** Home -. Login -. Register
  - **Manager:** Dashboard -. Add Player-. Edit Player-. Logout
  - **Security:** Unauthorized access redirects to login
- 



## 5. IMPLEMENTATION

### HTML/CSS/JavaScript

#### HTMLS Features:

- Semantic markup with proper elements
- Form elements with appropriate input types
- Responsive meta tags for mobile

#### CSS3 Implementation:

- Color scheme: Orange ( `#FFA500`) primary, Black ( `#000000`) secondary
- CSS Grid for responsive player cards
- Flexbox for navigation and forms
- Media queries for mobile-first design

#### JavaScript Functionality:

- Client-side form validation (email format, required fields)
- Real-time feedback for user input
- Image fallback handling
- Auto-hide alerts for better UX

## PHP Backend

### Key Features:

- Session management for authentication
- PDO database connection with error handling
- Prepared statements preventing SQL injection
- Password hashing for security
- Input sanitization using `htmlspecialchars0`

### CRUD Operations:

```
// Create Player
$stmt = $pdo->prepare("INSERT INTO players (name, position, team, photo, manager_id) VALUES (?, ?, ?, ?, ?)");

// Read Players
$stmt = $pdo->query("SELECT * FROM players ORDER BY created_at DESC");

// Update Player
$stmt = $pdo->prepare("UPDATE players SET name = ?, position = ? WHERE id = ? AND manager_id = ?");

// Delete Player
$stmt = $pdo->prepare("DELETE FROM players WHERE id = ? AND manager_id = ?");
```

---

## 6. DATABASE DESIGN

### ER Diagram

MANAGERS (1) ----< (N) PLAYERS

### Database Tables

#### managers table:

Column	Type	Constraints
id	INT	PRIMARY KEY, AUTO_INCREMENT
name	VARCHAR(100)	NOT NULL
email	VARCHAR(100)	NOT NULL, UNIQUE
password	VARCHAR(255)	NOT NULL
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

players table:

Column	Type	Constraints
id	INT	PRIMARY KEY, AUTO_INCREMENT
name	VARCHAR(100)	NOT NULL
position	VARCHAR(S0)	NOT NULL
team	VARCHAR(100)	NOT NULL
photo	VARCHAR(255)	NULL
manager_id	INT	FOREIGN KEY references managers(id)
created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP

### Relationships


- **One-to-Many:** One Manager can manage multiple Players
- **Foreign Key:** (players.manager\_id) references (managers.id)
- **Security:** Managers can only modify their own players

## 7. TESTING

### Testing Approach

Manual testing was conducted systematically across all features:

#### Authentication Testing:

- Valid registration with unique email
- Login with correct credentials
- Session management across pages
- Logout functionality
-  Protected page access control

### **Player Management Testing:**

- Add new players with all fields
- Edit existing player information
- Delete players with confirmation
- View players on public homepage
- Photo URL validation and fallback

### **Validation Testing:**

- Client-side: Email format, required fields
- Server-side: Input sanitization, SQL injection prevention
- Form validation on both frontend and backend

### **Responsive Design Testing:**

- Desktop browsers (Chrome, Firefox, Safari)
- Mobile devices (375px width)
- Tablet sizes (768px width)

### **Bugs Fixed**

1. **Image Loading:** Added fallback for broken image URLs
  2. **Form Data Loss:** Preserved form data on validation errors
  3. **Mobile Navigation:** Fixed alignment issues on small screens
- 

## **8. CHALLENGES FACED**

### **Technical Challenges**

#### **Database Connection Issues**

- **Problem:** Difficulty establishing reliable PDO connection
- **Solution:** Implemented comprehensive error handling and connection validation

#### **Session Security**

- **Problem:** Ensuring secure session handling across multiple pages
- **Solution:** Proper session regeneration and timeout handling

#### **Responsive Design**



- **Problem:** Consistent experience across different device sizes
- **Solution:** Mobile-first approach with CSS Grid and Flexbox

## Security Challenges

### SQL Injection Prevention

- **Problem:** Protecting against database attacks
- **Solution:** Used prepared statements with parameter binding

### Input Sanitization

- **Problem:** Preventing XSS attacks while preserving valid data
- **Solution:** Context-appropriate sanitization using `htmlspecialchars()`

## Problem-Solving Strategies

- Research official documentation and security guidelines
  - Iterative development with testing at each step
  - Code organization with consistent naming conventions
- 

## 9. CONCLUSION

### Learning Outcomes

This project successfully demonstrates full-stack web development skills integrating frontend and backend technologies. Key skills acquired include:

#### Technical Skills:

- Frontend development with HTML5, CSS3, and JavaScript
- Backend PHP programming with secure coding practices
- MySQL database design and management
- User authentication and session management
- Responsive web design principles

#### Security Implementation:

- Password hashing and verification
- SQL injection prevention
- XSS protection through proper sanitization

- Secure session management

## Project Achievements

- Complete authentication system
- Full CRUD operations for player management
- Responsive design across all devices
- Secure database operations
- Professional user interface

## Future Improvements

### Short-term:

- File upload system for player photos
- Advanced search and filtering capabilities
- Email notifications for user actions

### Long-term:

- Team management expansion
- Player statistics tracking
- Mobile app integration via API
- Advanced analytics dashboard

## Personal Development

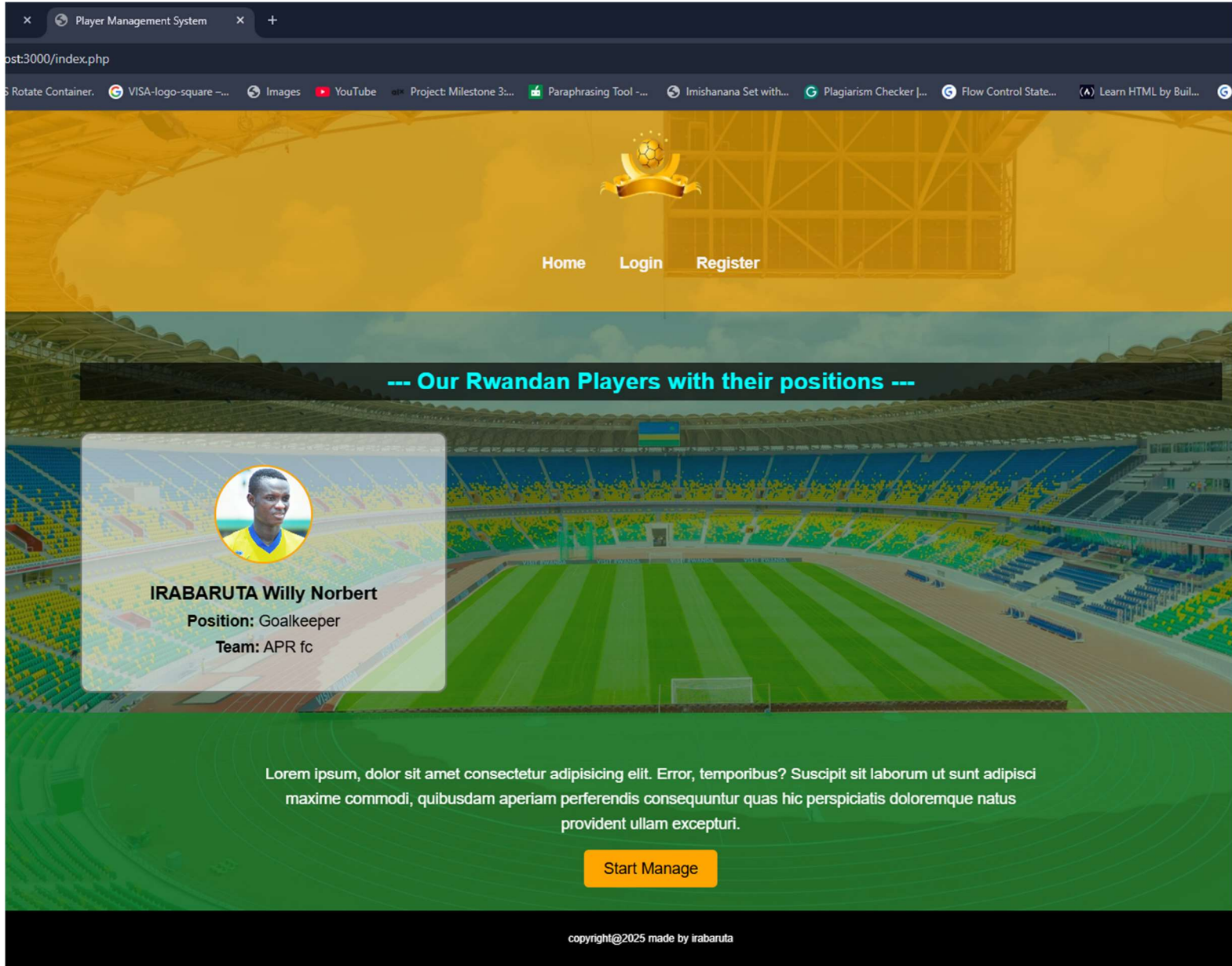
This project enhanced problem-solving skills, taught project management principles, and provided hands-on experience with industry-standard web development practices. The comprehensive nature of the project mirrors real-world development scenarios and has prepared me for professional software development challenges.

The experience emphasized the importance of security-first thinking, user-centered design, and systematic development approaches that will guide future technology projects.

---

## SCREENSHOTS SECTION

1. **Homepage** - Public view showing all players



## 2. Login Page - Manager authentication form



← Back to Home

## Login to Dashboard

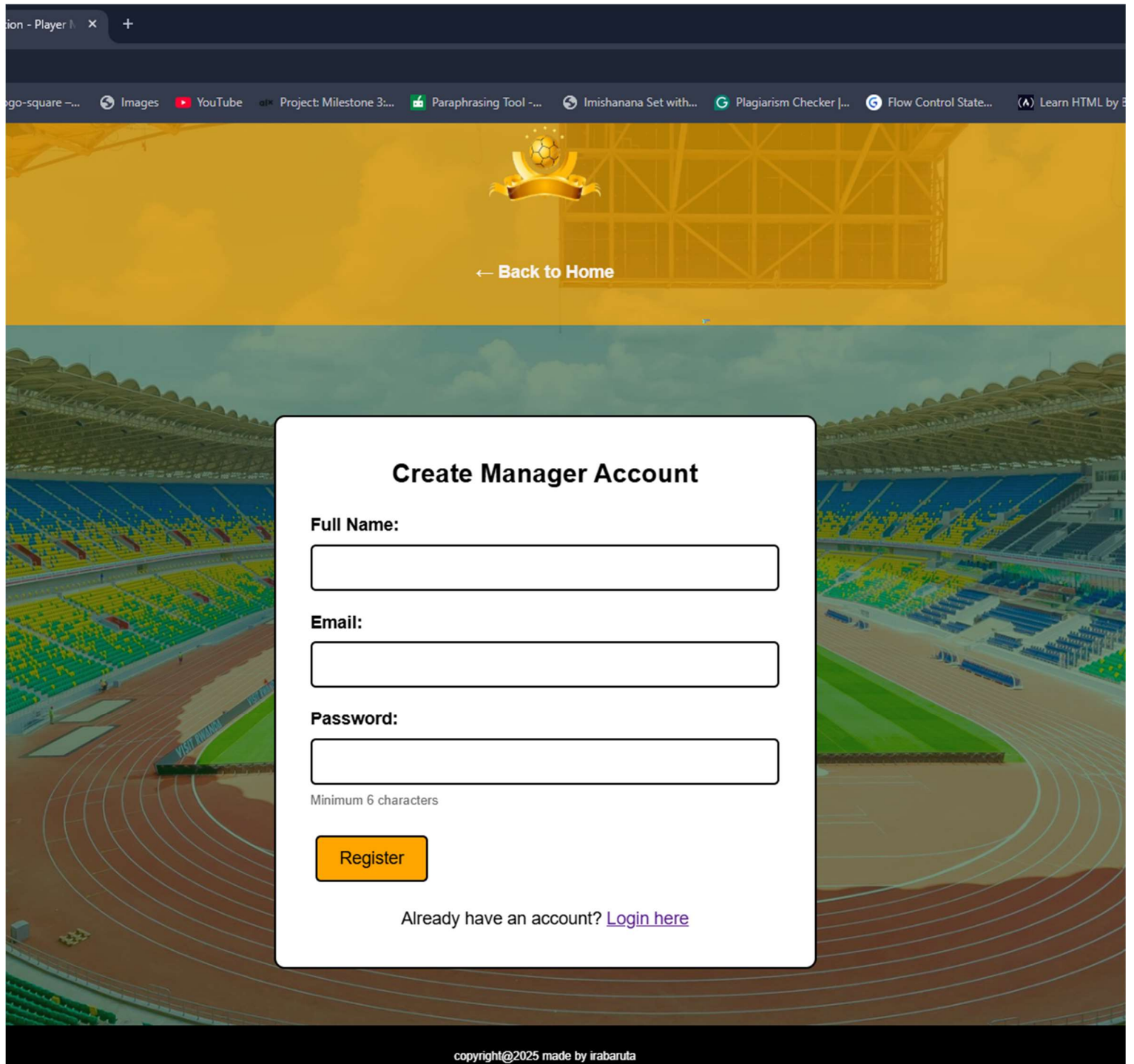
Email:

Password:

Login

Don't have an account? [Register here](#)

### 3. Registration Page - New manager signup



The image shows a web browser window with a registration form for a manager account. The browser's address bar shows a URL starting with 'ion - Player'. The page has a yellow header with a soccer ball icon and a 'Back to Home' link. The main content area features a large, stylized image of a stadium with a running track. Overlaid on this is a white registration form titled 'Create Manager Account'. The form contains three input fields for 'Full Name:', 'Email:', and 'Password:'. Below the password field is a note 'Minimum 6 characters'. A yellow 'Register' button is positioned below the form. At the bottom of the form, there is a link 'Already have an account? Login here'. The footer of the page, visible below the form, reads 'copyright@2025 made by irabaruta'.

ion - Player × +

go-square ... Images YouTube Project: Milestone 3... Paraphrasing Tool ... Imishanana Set with... Plagiarism Checker ... Flow Control State... Learn HTML by B

← Back to Home

### Create Manager Account

**Full Name:**

**Email:**

**Password:**

Minimum 6 characters

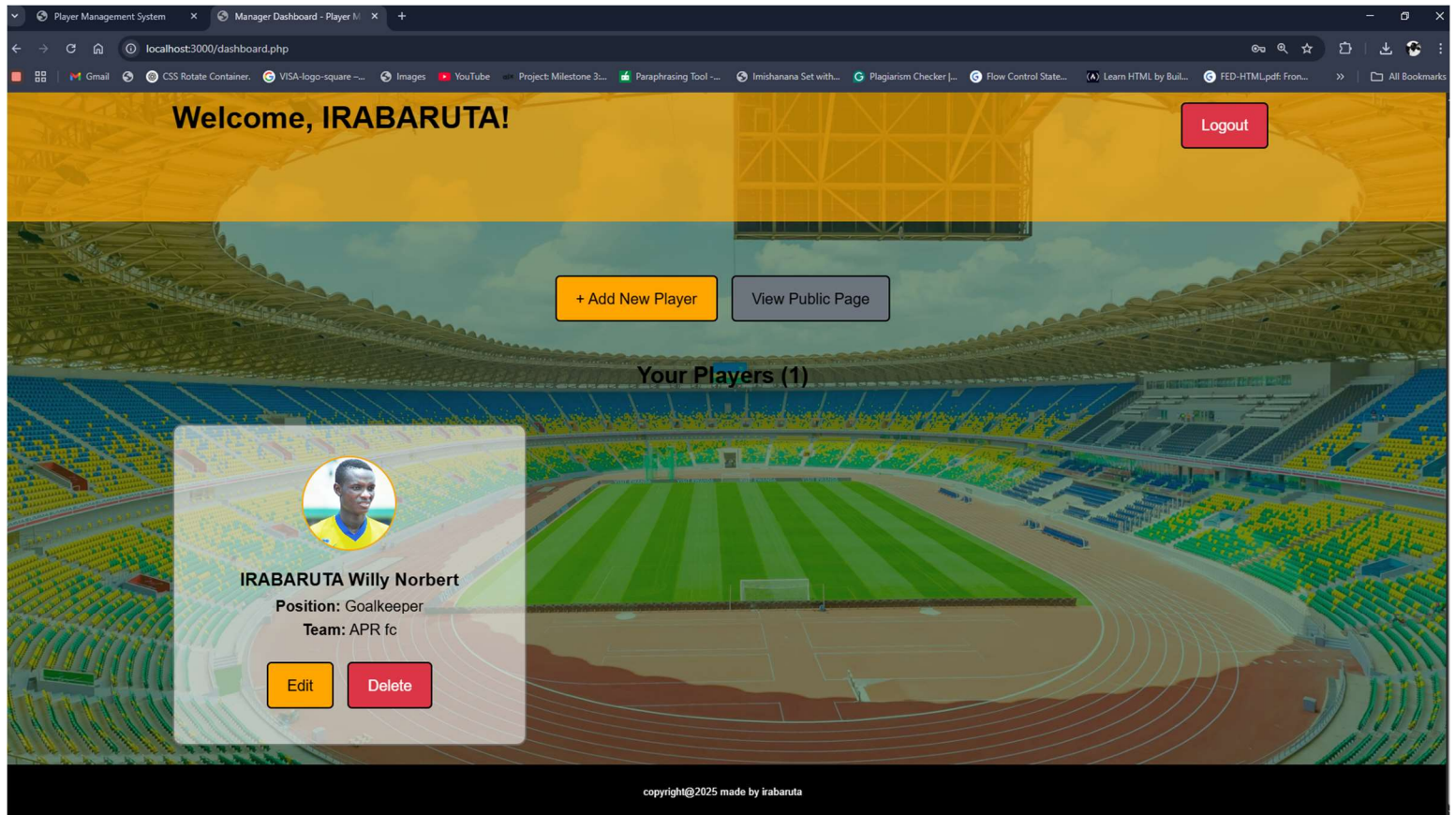
Register

Already have an account? [Login here](#)

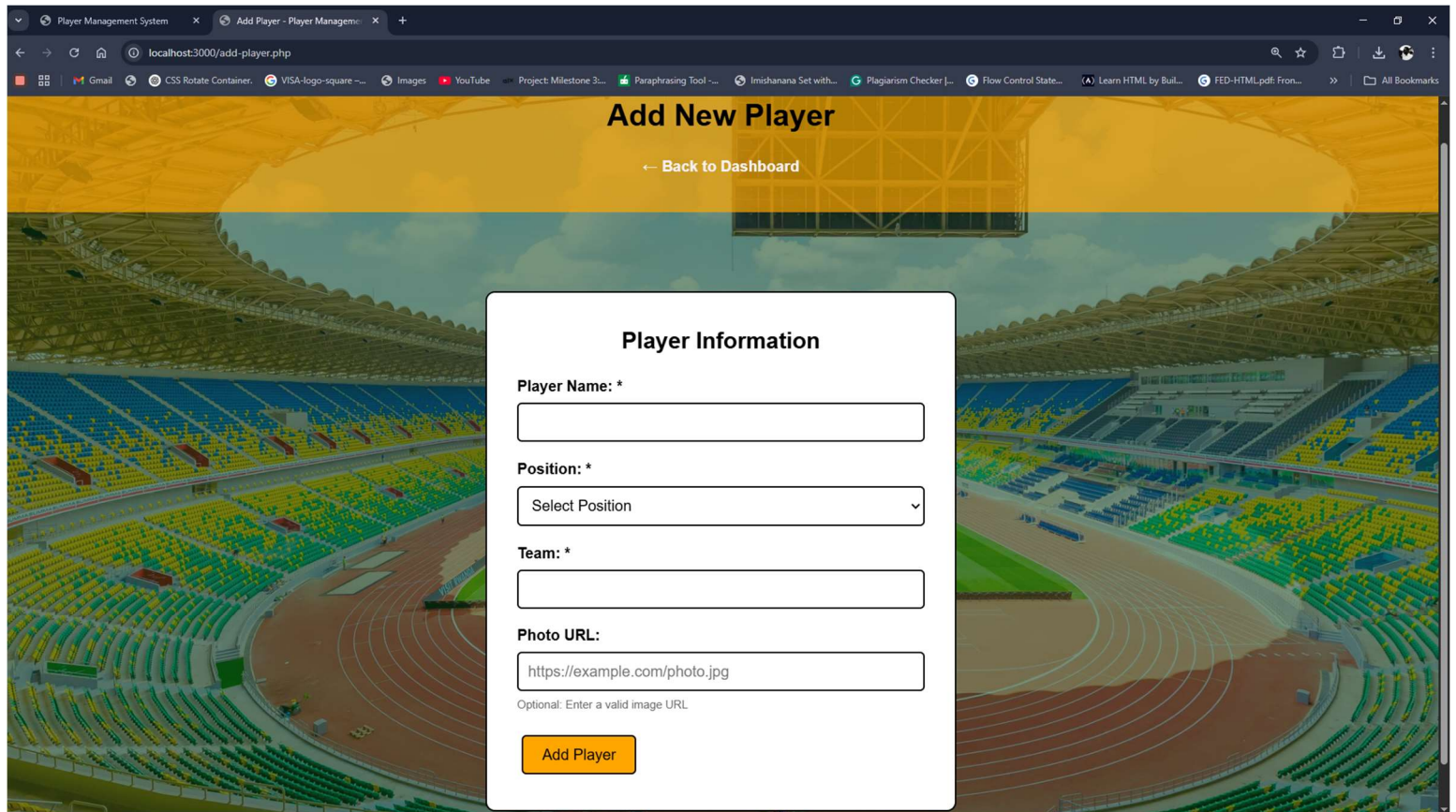
copyright@2025 made by irabaruta

### 4. Dashboard - Manager control panel with player list





## 5. Add Player Form - New player creation interface



## 6. Edit Player Form - Player information modification

## 7. Database Structure - phpMyAdmin showing tables

## 8. Mobile View - Responsive design on phone screen

---

## **End of Documentation**

*Total Pages: 6*

*Prepared by: /rabaruta Willy NORBERT*

*Registration: 27463*

*Date: August 2025*