

**ĐẠI HỌC ĐÀ NẴNG  
TRƯỜNG ĐẠI HỌC BÁCH KHOA  
KHOA CÔNG NGHỆ THÔNG TIN**



**PBL2: DỰ ÁN CƠ SỞ LẬP TRÌNH**

**ĐỀ TÀI 22:**

**XÂY DỰNG CHƯƠNG TRÌNH  
QUẢN LÝ GIẢI VÔ ĐỊCH VLEAGUE**

**SINH VIÊN THỰC HIỆN : Lê Nguyễn Thành Thật    LỚP: 22T\_DT5**

**Phạm Quốc Đức                      LỚP: 22T\_DT1**

**GIẢNG VIÊN HƯỚNG DẪN: Th.S Nguyễn Văn Nguyên**

**Đà Nẵng, 01/2024**

## LỜI MỞ ĐẦU

Thể thao đóng một vai trò quan trọng trong đời sống của chúng ta. Thể thao giúp chúng ta có sức khỏe, giảm bệnh tật và cũng là một biện pháp thư giãn hiệu quả. Trong các môn thể thao, bóng đá được mệnh danh là môn thể thao vua. Và vị trí đó vẫn không thể thay đổi cho tới thời điểm hiện tại.

Đối với Việt Nam, một quốc gia hơn 100 triệu dân thì môn bóng đá là một món ăn tinh thần không thể thiếu. Và một phần của món ăn đó là Giải bóng đá chuyên nghiệp vô địch quốc gia VLeague. Giải đấu lớn nhất cho cấp độ câu lạc bộ diễn ra hàng năm ở nước ta.

Với lòng yêu thích thể thao, nhóm chúng em đã chọn đề tài quản lý giải bóng đá vô địch quốc gia VLeague cho đồ án cuối môn học. Vì thời gian thực hiện hạn chế và phạm vi của đề tài khá rộng nên nội dung của đồ án chưa thật đầy đủ và chính xác với thực tế. Hi vọng khi điều kiện cho phép, nội dung đồ án có thể được mở rộng hơn đầy đủ hơn và có thể ứng dụng vào thực tế. Chúng em xin chân thành cảm ơn Thầy ThS. Nguyễn Văn Nguyên đã truyền đạt cho chúng em những kiến thức cần thiết cho môn học phục vụ cho đề tài này. Chúng em cũng xin cảm ơn vì sự hướng dẫn tận tình của Thầy trong việc tư vấn cho đồ án của chúng em.

## MỤC LỤC

LỜI MỞ ĐẦU .....	2
MỤC LỤC.....	3
GIỚI THIỆU ĐỀ TÀI .....	4
CHƯƠNG 1 PHÂN TÍCH CHỨC NĂNG HỆ THỐNG.....	5
1.1. Quản lý thông tin cầu thủ.....	5
1.2. Quản lý thông tin đội bóng .....	5
1.3. Quản lý thông tin trận đấu .....	5
1.4. Quản lý dữ liệu .....	6
CHƯƠNG 2 THIẾT KẾ CẤU TRÚC DỮ LIỆU .....	7
2.1. Phát biểu bài toán.....	7
2.2. Phân tích và ứng dụng cấu trúc dữ liệu trong hệ thống .....	8
CHƯƠNG 3 ÁP DỤNG HƯỚNG ĐỐI TƯỢNG TRONG CHƯƠNG TRÌNH VÀ TRIỂN KHAI HỆ THỐNG.....	10
3.1. Cấu trúc hệ thống hướng đối tượng .....	10
3.2. Kết quả .....	13
3.2.1. Giao diện chính của chương trình .....	13
3.2.2. Kết quả thực thi của chương trình .....	15
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....	19
Kết luận .....	19
Hướng phát triển.....	19
TÀI LIỆU THAM KHẢO.....	20

## GIỚI THIỆU ĐỀ TÀI

Trong thời đại số hóa và sự phát triển của công nghệ thông tin, việc quản lý thông tin trong lĩnh vực thể thao đã trở nên quan trọng hơn bao giờ hết. Trong bối cảnh này, bóng đá - môn thể thao vua đang nắm giữ một vị trí đặc biệt và đòi hỏi sự quản lý thông tin chặt chẽ để đảm bảo hoạt động của các đội bóng diễn ra một cách hiệu quả và chuyên nghiệp.

Giải bóng đá VLeague tại Việt Nam là một ví dụ rõ ràng về sự phát triển của bóng đá chuyên nghiệp tại đất nước hình chữ S. Đối với các chuyên gia bóng đá, huấn luyện viên, người hâm mộ và những người quản lý giải, việc quản lý thông tin về các đội bóng, cầu thủ và trận đấu trong V-League trở thành một nhiệm vụ không thể thiếu để đảm bảo tính chính xác và hiệu quả của mọi quyết định và hoạt động.

Đề tài này tập trung vào việc xây dựng một hệ thống quản lý thông tin giải bóng đá V-League sử dụng ngôn ngữ lập trình C++ và áp dụng hướng đối tượng. Hướng đối tượng giúp tổ chức mã nguồn một cách cấu trúc và dễ bảo trì, tạo điều kiện thuận lợi cho việc phát triển và mở rộng dự án trong tương lai.

Mục tiêu chính của đề tài là xây dựng một ứng dụng có khả năng quản lý thông tin về cầu thủ, đội bóng và trận đấu trong giải VLeague. Ứng dụng này cung cấp khả năng thêm mới thông tin, hiển thị, sửa đổi, xóa, tìm kiếm và thực hiện các hoạt động quản lý một cách hiệu quả.

## CHƯƠNG 1 PHÂN TÍCH CHỨC NĂNG HỆ THỐNG

### 1.1. Quản lý thông tin cầu thủ

**Thêm Cầu Thủ:** Hệ thống cho phép người dùng thêm thông tin về một cầu thủ mới bao gồm số CCCD (hoặc số hộ chiếu), họ tên, quốc tịch, ngày sinh, chiều cao, cân nặng, vị trí thi đấu, v.v. **Sửa Thông Tin Cầu Thủ:** Người dùng có khả năng chỉnh sửa thông tin của một cầu thủ hiện có, bao gồm việc cập nhật thông tin như số CCCD, vị trí thi đấu, và nhiều trường thông tin khác.

**Xóa Cầu Thủ:** Hệ thống cho phép người dùng xóa thông tin của một cầu thủ, xóa cầu thủ ra khỏi danh sách hoặc đánh dấu cầu thủ đã nghỉ hưu.

**Hiển Thị Thông Tin Cầu Thủ:** Hệ thống cung cấp khả năng hiển thị thông tin chi tiết về một cầu thủ, bao gồm số CCCD, họ tên, quốc tịch, ngày sinh, chiều cao, cân nặng, vị trí thi đấu, v.v.

**Liệt Kê Danh Sách Cầu Thủ Theo Tiêu Chí:** Người dùng có thể liệt kê danh sách các cầu thủ dựa trên các tiêu chí như năm sinh, vị trí thi đấu, quốc tịch, v.v.

**Tìm Cầu Thủ Cao Nhất, Thấp Nhất, Trẻ Nhất Giải:** Hệ thống cung cấp tính năng tìm kiếm và hiển thị thông tin về cầu thủ cao nhất, thấp nhất và trẻ nhất giải V-League.

### 1.2. Quản lý thông tin đội bóng

**Thêm Đội Bóng:** Hệ thống cho phép người dùng thêm thông tin về một đội bóng mới bao gồm tên đội bóng, địa phương, huấn luyện viên và danh sách các cầu thủ thuộc đội bóng.

**Sửa Thông Tin Đội Bóng:** Người dùng có khả năng chỉnh sửa thông tin của một đội bóng hiện có, bao gồm việc cập nhật thông tin như tên đội bóng, địa phương, và danh sách cầu thủ.

**Xóa Đội Bóng:** Hệ thống cho phép người dùng xóa thông tin của một đội bóng, xóa đội bóng ra khỏi danh sách hoặc đánh dấu đội bóng đã giải thể.

**Hiển Thị Thông Tin Đội Bóng:** Hệ thống cung cấp khả năng hiển thị thông tin chi tiết về một đội bóng, bao gồm tên đội bóng, địa phương, huấn luyện viên, và danh sách các cầu thủ thuộc đội bóng đó.

**Hiển Thị Danh Sách Cầu Thủ Trong Đội Bóng:** Hệ thống cho phép người dùng xem danh sách các cầu thủ thuộc đội bóng và thông tin chi tiết về họ.

### 1.3. Quản lý thông tin trận đấu

**Thêm Trận Đấu:** Hệ thống cho phép người dùng thêm thông tin về một trận đấu mới, bao gồm ngày thi đấu, sân thi đấu, tên hai đội bóng và tỷ số trận đấu.

**Sửa Thông Tin Trận Đấu:** Người dùng có khả năng chỉnh sửa thông tin của một trận đấu hiện có, bao gồm việc cập nhật thông tin như ngày thi đấu, sân thi đấu và tỷ số trận đấu.

**Xóa Trận Đấu:** Hệ thống cho phép người dùng xóa thông tin của một trận đấu ra khỏi danh sách hoặc đánh dấu trận đấu đã hoàn thành.

**Hiển Thị Thông Tin Trận Đấu:** Hệ thống cung cấp khả năng hiển thị thông tin chi tiết về một trận đấu, bao gồm ngày thi đấu, sân thi đấu, tên hai đội bóng và tỷ số trận đấu.

**Liệt Kê Kết Quả Trận Đấu Theo Ngày, Tháng Hoặc Toàn Giải:** Người dùng có thể liệt kê kết quả của các trận đấu trong một ngày cụ thể, trong một tháng hoặc toàn bộ giải.

**Tính Điểm Cho Đội Bóng Tại Một Thời Điểm:** Hệ thống tính điểm cho các đội bóng dựa trên kết quả của các trận đấu và hiển thị bảng xếp hạng.

### **1.4. Quản lý dữ liệu**

**Lưu trữ dữ liệu:** Hệ thống hỗ trợ lưu trữ thông tin trong file định dạng text.

**Xuất dữ liệu:** Người dùng có thể xuất dữ liệu ra các file để lưu trữ và chia sẻ.

## CHƯƠNG 2 THIẾT KẾ CẤU TRÚC DỮ LIỆU

### 2.1. Phát biểu bài toán

- **Đầu vào :**

1. Thêm Cầu thủ:

Yêu cầu nhập thông tin về cầu thủ bao gồm số CCCD, họ tên, quốc tịch, ngày sinh, chiều cao, cân nặng, vị trí thi đấu.

2. Sửa thông tin cầu thủ:

Yêu cầu nhập số CCCD hoặc số hộ chiếu của cầu thủ cần sửa.

Cho phép người dùng cập nhật thông tin của cầu thủ.

3. Xóa cầu thủ:

Yêu cầu nhập số CCCD hoặc số hộ chiếu của cầu thủ cần xóa.

4. Hiển thị thông tin cầu thủ:

Yêu cầu nhập số CCCD hoặc số hộ chiếu của cầu thủ để hiển thị thông tin chi tiết của cầu thủ đó.

5. Liệt kê danh sách cầu thủ theo tiêu chí: Cho phép người dùng chọn tiêu chí (ví dụ: năm sinh, vị trí thi đấu) và hiển thị danh sách cầu thủ theo tiêu chí đó.

6. Tìm cầu thủ cao nhất, thấp nhất, trẻ nhất giải:

Yêu cầu hiển thị thông tin về cầu thủ cao nhất, thấp nhất và trẻ nhất trong giải.

- **Đầu Ra :**

1. Kết quả thực hiện thao tác thêm, sửa, xóa cầu thủ:

Thông báo xác nhận thao tác đã được thực hiện thành công hoặc thông báo lỗi nếu có.

2. Thông tin chi tiết cầu thủ:

Hiển thị thông tin chi tiết về cầu thủ bao gồm số CMND, họ tên, quốc tịch, ngày sinh, chiều cao, cân nặng, vị trí thi đấu

3. Danh sách cầu thủ theo tiêu chí:

Hiển thị danh sách các cầu thủ theo tiêu chí đã chọn, bao gồm thông tin cơ bản của họ.

4. Thông tin cầu thủ cao nhất, thấp nhất, trẻ nhất giải:

Hiển thị thông tin chi tiết về cầu thủ cao nhất, thấp nhất và trẻ nhất trong giải.

5. Kết quả thực hiện thao tác thêm, sửa, xóa đội bóng:

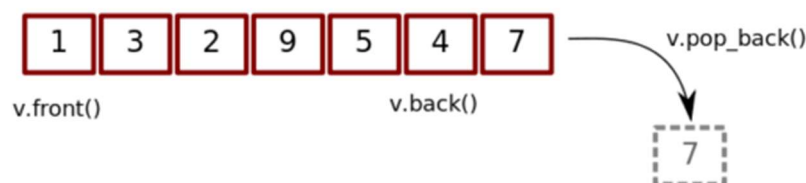
Tương tự như thao tác quản lý cầu thủ, cho phép thực hiện các thao tác quản lý đội bóng.

6. Kết quả thực hiện thao tác thêm, sửa, xóa trận đấu:

Tương tự như thao tác quản lý cầu thủ, cho phép thực hiện các thao tác quản lý trận đấu.

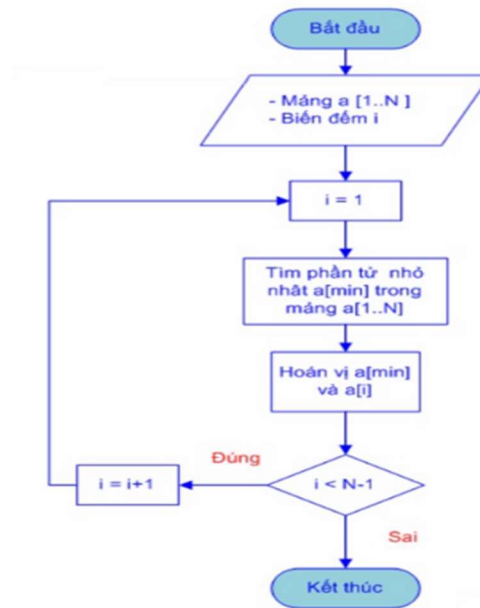
## 2.2. Phân tích và ứng dụng cấu trúc dữ liệu trong hệ thống

- Ứng dụng cấu trúc dữ liệu vector (mảng cấp phát động) để lưu trữ dữ liệu.
  - + Vector trong C++ là một mảng động dùng để lưu trữ dữ liệu. Khác với array – dùng để lưu trữ dữ liệu tuần tự các vector có độ linh hoạt cao hơn nhiều. Vector có thể tự động resize khi có phần tử được chèn hay xóa khỏi array, trong khi đó mảng thông thường có kích thước cố định.



- Sử dụng các thuật toán sắp xếp và tìm kiếm để tìm cầu thủ lớn tuổi nhất, nhỏ tuổi nhất, cao nhất, thấp nhất.
  - + Lưu đồ thuật toán lính canh :





+ Thuật toán tìm kiếm tuyến tính :

1. Gán  $i = 0$ .

2. So sánh giá trị của  $A[i]$  và  $x$

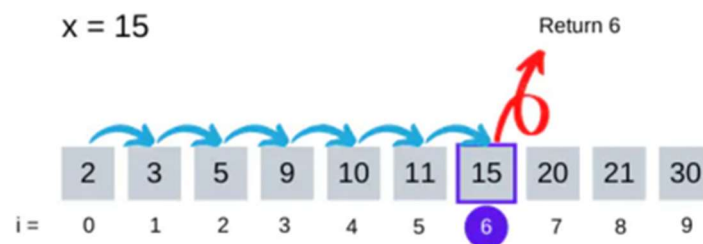
Nếu  $A[i] == x$  thì dừng và trả về giá trị  $i$  là vị trí của  $x$  trong mảng  $A$ .

Nếu  $A[i] != x$  thì sang bước 3.

3. Gán  $i = i + 1$ .

Nếu  $i == n$  (hết mảng) thì dừng lại và trả kết quả không tìm thấy.

Nếu  $i < n$  thì quay lại bước 2.



- Với độ phức tạp thuật toán là :  $O(n)$ .

## CHƯƠNG 3 ÁP DỤNG HƯỚNG ĐỐI TƯỢNG TRONG CHƯƠNG TRÌNH VÀ TRIỂN KHAI HỆ THỐNG

### 3.1. Cấu trúc hệ thống hướng đối tượng

#### 1. Class CauThu

##### Các thuộc tính:

- + Căn cước công dân.
- + Họ và tên.
- + Quốc tịch.
- + Ngày tháng năm sinh.
- + Chiều cao.
- + Cân nặng.
- + Vị trí thi đấu.
- + Thuộc đội bóng.

##### Các phương thức:

- + Các constructor và destructor.
- + string getID( ) : lấy căn cước công dân của cầu thủ.
- + string getFullname( ) : lấy tên của cầu thủ.
- + string getInternational( ) : lấy quốc tịch của cầu thủ.
- + string getBirth( ) : lấy ngày sinh của cầu thủ.
- + string getHeight( ) : lấy chiều cao của cầu thủ.
- + string getWeight( ) : lấy cân nặng của cầu thủ.
- + string getPosition( ) : lấy vị trí thi đấu của cầu thủ.
- + string getTendoibong( ) : lấy tên đội bóng của cầu thủ.
- + int AddPlayer( ) : thêm cầu thủ vào danh sách.
- + string modifyPlayer( ) : sửa cầu thủ trong danh sách.
- + void listingPlayer( ) : liệt kê danh sách các cầu thủ theo tùy chọn.

- + void getYoungestPlayer( ) : tìm cầu thủ trẻ tuổi nhất.
- + void getTallestPlayer( ) : tìm cầu thủ cao nhất.
- + void getShortestPlayer( ) : tìm cầu thủ thấp nhất.
- + int handleModifyPlayer( ) : xử lý sửa cầu thủ.
- + void getPlayer( ) : xuất thông tin của cầu thủ.

## 2. Class doibong

### Các thuộc tính:

- + Tên đội bóng.
- + Địa phương.
- + Huấn luyện viên.
- + Danh sách các cầu thủ.
- + Điểm số.

### Các phương thức:

- + Các constructor và destructor.
- + void suadoibong( ) : sửa đội bóng trong danh sách.
- + void themcauthuvaodanh sach( ) : thêm cầu thủ vào danh sách.
- + void xoacauthukhoids( ) : xóa cầu thủ khỏi danh sách.
- + void setDiem( ) : tính điểm cho các đội bóng vào cùng thời điểm.
- + string getTenDoiBong( ) : lấy tên đội bóng.
- + string getDiaPhuong( ) : lấy địa phương đội bóng.
- + string getHuanLuyenVien( ) : lấy huấn luyện viên của đội bóng.
- + vector<CauThu> getdscauthu( ) : lấy danh sách cầu thủ.
- + int getdodaidscauthu( ) : lấy số lượng cầu thủ trong danh sách.
- + void suadanh sachcauthu( ) : sửa danh sách cầu thủ.
- + void inThongTin( ) : xuất thông tin của đội bóng.
- + void inDiem( ) : xuất điểm của đội bóng.

## 3. Class TranDau

### Các thuộc tính :

- + Sân thi đấu.

- + Ngày thi đấu.
- + Team 1.
- + Team 2.
- + Tỷ số.

**Các phương thức:**

- + Các constructor và destructor.
- + int getNgaythidau( ) : lấy ngày thi đấu.
- + int getThangthidau( ) : lấy tháng thi đấu.
- + int getNamthidau( ) : lấy năm thi đấu.
- + string getTyso( ) : lấy tỷ số trận đấu.
- + string getTeam1( ) : lấy tên đội 1.
- + string getTeam2( ) : lấy tên đội 2.
- + void inThongtin( ) : xuất thông tin trận đấu.

### 3.2. Kết quả

#### 3.2.1. Giao diện chính của chương trình

Quan ly cau thu
Quan ly doi bong
Quan ly tran dau
Tinh diem cho cac doi bong
Thoat

Xem thong tin cau thu
Them moi cau thu
Sua cau thu
Xoa cau thu
Hien thi cau thu tre tuoi nhat
Hien thi cau thu cao nhat
Hien thi cau thu thap nhat
Quay lai

Thêm đôi bong

Sửa đôi bong

Xóa đôi bong

Hiện thi thông tin đôi bong

Quay lại

Thêm tran dau

Xem tran dau trong ngay

Xem tran dau trong thang

Xem tat ca tran dau

Quay lại

3.2.2. Kết quả thực thi của chương trình

+ Kết quả thực thi của chương trình là các bảng dữ liệu được in ra màn hình.

NHAP CCCD CAN TIM

So CCCD

987654321

So CCCD	Ten	Ngay Sinh	Quoc Tich	Chieu cao	Can nang	Vi tri thi dau	Thuoc doi bong
987654321	Lionel Messi	24/06/1987	Argentina	170.0	72.0	Tien dao	Manchester United

Quay lai

---

Ten doi bong	Diem
Manchester United	3
Liverpool FC	6
Real Madrid	3

Quay lai

Them thanh cong!!

So CCCD	<input type="text" value="049204009171"/>	
Ten	<input type="text" value="pham quoc duc"/>	
Ngay Sinh	<input type="text" value="10/04/2004"/>	
Quoc Tich	<input type="text" value="vietnam"/>	
Chieu cao	<input type="text" value="300.0"/>	
Can nang	<input type="text" value="150.0"/>	
Vi tri thi dau	<input type="text" value="hau ve"/>	
Thuoc doi bong	<input type="text" value="Liverpool FC"/>	<input type="button" value="Them"/>

NHAP CCCD CAN XOA

So CCCD	<input type="text" value="10085508"/>
---------	---------------------------------------

XOA THANH CONG !!!!



PBL2: DỰ ÁN CƠ SỞ SỔ LẬP TRÌNH

So CCCD	Ten	Ngày Sinh	Quốc Tịch	Chiều cao	Cân nặng	Vị trí thi đấu	Thuộc đội bóng
111122223	phạm quốc đức	05/02/2004	Portugal	187.0	83.0	hỏi cup	Liverpool FC
049204009171	phạm quốc đức	10/04/2004	vietnam	300.0	150.0	hau ve	Liverpool FC

Quay lại

So CCCD	Ten	Ngày Sinh	Quốc Tịch	Chiều cao	Cân nặng	Vị trí thi đấu	Thuộc đội bóng
049204009171	phạm quốc đức	10/04/2004	vietnam	300.0	150.0	hau ve	Liverpool FC

Quay lại

So CCCD	Ten	Ngày Sinh	Quốc Tịch	Chiều cao	Cân nặng	Vị trí thi đấu	Thuộc đội bóng
987654321	Lionel Messi	24/06/1987	Argentina	169.0	72.0	Tiền đạo	Manchester United

Quay lại

Ten doi bong

Ten dia phuong

Huan luyen vien

So CCCD

Ten

Ngay Sinh

Quoc Tich

Chieu cao

Can nang

Vi tri thi dau

Thuoc doi bong

Them

Quay lai

NHAP TEN DOI BONG CAN TIM

Ten doi bong

Liverpool FC

Ten doi bong	Ten dia phuong	Huan luyen vien
Liverpool FC	liverpool	ABCDEFGH

So CCCD	Ten	Ngay Sinh	Quoc Tich	Chieu cao	Can nang	Vi tri thi dau	Thuoc doi bong
111122223	pham quoc duc	05/02/2004	Portugal	187.0	83.0	hoi cup	Liverpool FC
111222333	frieren	24/06/1000	Argentina	187.0	72.0	ap	Liverpool FC
111111111	ABCDEFGHTIJK	24/06/1000	Argentina	187.0	72.0	ap	Liverpool FC
049204009171	pham quoc duc	10/04/2004	vietnam	300.0	150.0	hau ve	Liverpool FC

Quay lai

## KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### Kết luận

Dự án quản lý thông tin giải bóng đá VLeague đã định ra một cơ sở mạnh mẽ cho việc tổ chức và quản lý dữ liệu liên quan đến giải bóng đá một cách hiệu quả. Bằng cách sử dụng hướng đối tượng và cấu trúc dữ liệu, chúng ta đã xây dựng một hệ thống linh hoạt có khả năng thực hiện nhiều chức năng quản lý, từ quản lý cầu thủ và đội bóng đến việc theo dõi trận đấu và tính điểm.

Hệ thống này cung cấp cho người quản lý giải bóng đá một công cụ mạnh mẽ để theo dõi và cập nhật thông tin về cầu thủ, đội bóng và trận đấu. Nó cũng giúp tạo ra bảng xếp hạng và theo dõi kết quả thi đấu một cách dễ dàng.

### Hướng phát triển

Dự án có thể phát triển và mở rộng với các tính năng và cải tiến sau:

Giao diện người dùng (UI): Phát triển một giao diện người dùng đồ họa (GUI) để làm cho chương trình dễ sử dụng hơn và cung cấp trải nghiệm người dùng tốt hơn.

Bảng xếp hạng chi tiết: Thêm thông tin chi tiết về bảng xếp hạng như số trận đấu đã đá, số bàn thắng ghi được, số bàn thua, v.v.

Thống kê thêm: Cung cấp thống kê và biểu đồ về kết quả thi đấu, sự phát triển của cầu thủ, và đội bóng trong suốt thời gian diễn ra giải.

Bảo mật dữ liệu: Cải tiến bảo mật để đảm bảo dữ liệu của các đội bóng và cầu thủ được bảo vệ.

Xử lý sự kiện thời gian thực: Thêm tính năng xử lý thông tin trận đấu và kết quả thời gian thực, cập nhật bảng xếp hạng ngay sau khi kết thúc một trận đấu.

Ghi nhật ký (Logging): Thêm hệ thống ghi nhật ký để theo dõi và ghi lại tất cả các hoạt động quản lý và thay đổi dữ liệu.

Tích hợp cơ sở dữ liệu: Sử dụng hệ quản lý cơ sở dữ liệu (ví dụ: MySQL, PostgreSQL) để lưu trữ và truy xuất dữ liệu một cách hiệu quả.

Phát triển ứng dụng di động: Tạo phiên bản ứng dụng di động cho người dùng truy cập thông tin giải bóng đá trên các thiết bị di động.

Hệ thống thông báo: Cung cấp thông báo tự động về kết quả trận đấu và thông tin cập nhật cho người hâm mộ.

Liên kết dữ liệu ngoại vi: Kết hợp với các nguồn dữ liệu ngoại vi như trang web giải bóng đá chính thức hoặc các nguồn thống kê để cập nhật thông tin một cách tự động.

## TÀI LIỆU THAM KHẢO

- [1] Lê Thị Mỹ Hạnh, *Lập trình hướng đối tượng C++*
- [2] Trần Văn Lăng, *Lập trình hướng đối tượng C++*
- [3] Scott Robert Ladd, *C++ Kỹ thuật và Ứng dụng*
- [4] Nguyễn Thanh Thủy, *Bài tập lập trình hướng đối tượng*
- [5] Peter Müller, *Introduction to Object-Oriented Programming Using C++*
- [6] Gs. Phạm Văn Ất, *Giáo trình C++ và lập trình hướng đối tượng*
- [7] Grady Booch, *Object-Oriented Analysis and Design with Applications*, Addison-Wesley.
- [8] Alan Kay, *The Early History of Smalltalk*.
- [9] Bertrand Meyer, *Object-Oriented Software Construction*, Prentice Hall.
- [10] James Rumbaugh, Michael Blaha, William Premerlani, Frederick Eddy, William Lorensen, *Object-Oriented Modeling and Design*, Prentice Hall.

## PHỤ LỤC

File DOIBONG.CPP

```
#include<iostream>
#include"DOIBONG.H"
#include"CAUTHU.H"
#include<string>
using namespace std;

doibong::doibong(string tenDoiBong, string diaPhuong, string
huanLuyenVien, vector<CauThu>& danhsachcauthu)
: tenDoiBong(tenDoiBong), diaPhuong(diaPhuong),
huanLuyenVien(huanLuyenVien) {
    for (int i = 0; i < danhsachcauthu.size(); i++) {
        this->danhsachcauthu.push_back(danhsachcauthu[i]);
    }
}

void doibong::suadoibong(string ten, string diaphuong, string
huanluyenvien) {
    this->tenDoiBong = ten;
    this->diaPhuong = diaphuong;
    this->huanLuyenVien = huanluyenvien;
}

void doibong::themcauthuvaodanhsach(CauThu& t) {
    this->danhsachcauthu.push_back(t);
}

void doibong::xoacauthukhoids(string id) {
    for (auto i = this->danhsachcauthu.begin(); i != this-
>danhsachcauthu.end(); i++) {
        if (i->getID() == id) {
            this->danhsachcauthu.erase(i);
            break;
        }
        if (i == this->danhsachcauthu.end() - 1) {
```

```
cout << "Khong tim thay!" << endl;
    }
}
}

void doibong::getdanhsachcauthu() {
    for (auto& i : this->danhsachcauthu) {
        i.getPlayer();
        cout << "-----" << endl;
    }
}

int doibong::getdodaidscauthu() const {
    return danhsachcauthu.size();
}

vector<CauThu> doibong::getdscauthu() const {
    return danhsachcauthu;
}

string doibong::getTenDoiBong() const { return tenDoiBong; }
string doibong::getDiaPhuong() const { return diaPhuong; }
string doibong::getHuanLuyenVien() const { return huanLuyenVien; }

void doibong::inThongTin() {
    cout << "Ten doi bong: " << this->tenDoiBong << endl;
    cout << "Dia phuong: " << this->diaPhuong << endl;
    cout << "Huan luyen vien: " << this->huanLuyenVien << endl;
    cout << "Danh sach cau thu : " << endl;
    for (int i = 0; i < this->danhsachcauthu.size(); i++) {
        this->danhsachcauthu[i].getPlayer();
        cout << "-----" << endl;
    }

}
```

**File DOIBONG.H**

```
#pragma once
#pragma once
#ifndef DOIBONG_H
#define DOIBONG_H
#include<iostream>
#include"CAUTHU.H"
#include<string>
#include<vector>
using namespace std;

class doibong {
public:
    doibong(string, string, string, vector<CauThu>& danhsachcauthu);
    void suadoibong(string, string, string);
    void themcauthuvaodanhsach(CauThu& t);
    void xoacauthukhoids(string);

    string getTenDoiBong() const;
    string getDiaPhuong() const;
    string getHuanLuyenVien() const;
    vector<CauThu> getdscauthu() const;
    int getdodaidscauthu() const;
    void getdanhsachcauthu();
    void inThongTin();

private:
    string tenDoiBong;
    string diaPhuong;
    string huanLuyenVien;
    vector<CauThu> danhsachcauthu;
};

#endif
```

**File CAUTHU.CPP**

```
#include <iostream>
```

```
#include <string>
```

```
#include <vector>
```

```
#include "CAUTHU.H"
```

```
using namespace std;
```

```
CauThu::CauThu(string id, string fullName, string intenational, string  
birth, string height, string weight, string position)  
    : id(id), fullName(fullName), intenational(intenational), birth(birth),  
height(height), weight(weight), position(position) {}
```

```
CauThu::CauThu(string id, string fullName, string intenational, string  
birth, string height, string weight, string position, string doibong)  
    : id(id), fullName(fullName), intenational(intenational), birth(birth),  
height(height), weight(weight), position(position), doibong(doibong) {}
```

```
CauThu::~CauThu() {
```

```
}
```

```
string CauThu::getID() const {
```

```
    return this->id;
```

```
}
```

```
int CauThu::AddPlayer(vector<CauThu> listPlayerCT) {
```

```
    string cccd;
```

```
    cout << "nhap CCCD : "; getline(cin, this->id);
```

```
    for (int i = 0; i < listPlayerCT.size(); i++) {
```

```
        if (listPlayerCT[i].getID() == this->id) {
```

```
            cout << "so can cuoc cong dan da ton tai!!" << endl;
```

```
            return 0;
```

```
        }
```

```
        if (i == listPlayerCT.size() - 1) {
```

```
            cout << "nhap ho va ten : "; getline(cin, this->fullName);
```

```
            cout << "nhap quoc tich : "; getline(cin, this-
```

```
>intenational);
```

```
            cout << "nhap ngay thang nam sinh : "; getline(cin, this-
```

```
>birth);
```



```
cout << "nhap chieu cao : "; getline(cin, this->height);
        cout << "nhap can nang : "; getline(cin, this->weight);
        cout << "nhap vi tri thi dau : "; getline(cin, this-
>position);
        cout << "Thuoc doi bong : "; getline(cin, this->doibong);
        return 2;
    }
}
}
```

```
void CauThu::deletePlayer(vector<CauThu>& listPlayerCT) {

    string CMND;
    cout << "vui long nhap CMND cau thu can xoa : ";
    cin >> CMND;
    for (auto i = listPlayerCT.begin(); i != listPlayerCT.end(); i++) {
        if (CMND == i->getID()) {
            listPlayerCT.erase(i);
            cout << "Xoa thanh cong !" << endl;
            break;
        }
        if (i == listPlayerCT.end() - 1) {
            cout << "khong tim thay!!!!!!!!!" << endl;
        }
    }
}
```

```
void CauThu::modifyPlayer(vector<CauThu>& listPlayerCT) {
    string CMND, select;
    cout << "vui long nhap CMND cau thu can sua : ";
    cin >> CMND;
    for (auto i = listPlayerCT.begin(); i != listPlayerCT.end(); i++) {
        if (CMND == i->getID()) {
            cout << "Sua toan bo thong tin cua cau thu (yes/no) : ";
            cin >> select;
            cin.ignore();
            if (select == "yes") {
                int selection;
```

```
do {
    selection = i-
>handleModifyPlayer(listPlayerCT, i->getID());
    if (selection == 2) {
        cout << "Sua doi thanh cong!!!" <<
endl;
    }
} while (selection != 2);

}
if (select == "no") {
    int choose;
    i->getPlayer();
    cout << "8. Thuoc doi bong: " << i-
>getTendoibong() << endl;
    cout << "Chon noi dung can sua : ";
    cin >> choose;
    cin.ignore();
    switch (choose) {
    case 1:
    {
        string _id;
        cout << i->id << " --> ";
        cin >> _id;
        for (auto j = listPlayerCT.begin(); j !=
listPlayerCT.end(); j++) {
            if (j->id == i->id) continue;
            if (_id == j->id) {
                cout << "Sua doi that bai! So
CCCD nay da ton tai! " << endl;
                break;
            }
            if (j == listPlayerCT.end() - 1) {
                i->id = _id;
                cout << "Sua doi thanh cong !"
<< endl;
            }
        }
    }
}
```

**}****break;****case 2:****cout << i->fullName << " --> ";****getline(cin, i->fullName);****cout << "Sua doi thanh cong !" << endl;****break;****case 3:****cout << i->intentional << " --> ";****getline(cin, i->intentional);****cout << "Sua doi thanh cong !" << endl;****break;****case 4:****cout << i->birth << " --> ";****cin >> i->birth;****cout << "Sua doi thanh cong !" << endl;****break;****case 5:****cout << i->height << " --> ";****cin >> i->height;****cout << "Sua doi thanh cong !" << endl;****break;****case 6:****cout << i->weight << " --> ";****cin >> i->weight;****cout << "Sua doi thanh cong !" << endl;****break;****case 7:****cout << i->position << " --> ";****getline(cin, i->position);****cout << "Sua doi thanh cong !" << endl;****break;****case 8:****cout << i->getTendoibong() << "-->";****getline(cin, i->doibong);****cout << "Sua doi thanh cong !" << endl;****break;****default:**

```
cout << "Khong hop le !!" << endl;
                                break;
                                }
                                }
                                break;
                                }
                                if (i == listPlayerCT.end() - 1) {
                                    cout << "khong tim thay!!!!!!!!!" << endl;
                                }
                            }
                        }

void CauThu::listingPlayer(vector<CauThu> listPlayerCT) {
    cout << "Liet ke danh sach cau thu theo nam sinh chon 1. " << endl;
    cout << "Liet ke danh sach cau thu theo vi tri thi dau chon 2. " <<
endl;
    int select;
    cout << "vui long chon : ";
    cin >> select;
    cin.ignore();
    switch (select) {
    case 1:
    {
        string yearSelect;
        int count = 0;
        cout << "Nhap nam sinh : "; cin >> yearSelect;
        for (auto i = listPlayerCT.begin(); i != listPlayerCT.end(); i++) {
            if (i->birth.find(yearSelect) != -1) {
                i->getPlayer();
                cout << "-----" << endl;
                count++;
            }
            if (count == 0 && i == listPlayerCT.end() - 1) {
                cout << "Khong tim thay!!" << endl;
            }
        }
    }
    break;
}
```

**case 2:**

```
{
    string positionSelect;
    cout << "Nhap vi tri thi dau : ";
    getline(cin, positionSelect);
    for (auto i = listPlayerCT.begin(); i != listPlayerCT.end(); i++) {
        if (i->position == positionSelect) {
            i->getPlayer();
            cout << "-----" << endl;
        }
        if (i->position != positionSelect && i ==
listPlayerCT.end() - 1) {
            cout << "Khong tim thay!!" << endl;
        }
    }
}
break;
}
```

**void CauThu::getYoungestPlayer(vector<CauThu>& listPlayerCT) {**

```
    CauThu min = listPlayerCT[0];
    for (int i = 1; i < listPlayerCT.size(); i++) {
        int count1 = 0;
        int count2 = 0;
        for (int j = 0; j < listPlayerCT[i].birth.length(); j++) {
            if (min.birth[j] == '/') count2++;
            if (count2 == 2) {
                count2 = j + 1;
            }
            if (listPlayerCT[i].birth[j] == '/') count1++;
            if (count1 == 2) {
                count1 = j + 1;
            }
        }
        if (stoi(min.birth.substr(count1)) <
stoi(listPlayerCT[i].birth.substr(count2))) {
```

```
min = listPlayerCT[i];
    }
}
min.getPlayer();
}
```

```
void CauThu::getTallestPlayer(vector<CauThu>& listPlayerCT) {
    CauThu max = listPlayerCT[0];
    for (int i = 1; i < listPlayerCT.size(); i++) {

        if (stoi(max.height) < stoi(listPlayerCT[i].height)) {
            max = listPlayerCT[i];
        }
    }
    max.getPlayer();
}
```

```
void CauThu::getShortestPlayer(vector<CauThu>& listPlayerCT) {
    CauThu min = listPlayerCT[0];
    for (int i = 1; i < listPlayerCT.size(); i++) {

        if (stoi(min.height) > stoi(listPlayerCT[i].height)) {
            min = listPlayerCT[i];
        }
    }
    min.getPlayer();
}
```

```
int CauThu::handleModifyPlayer(vector<CauThu>& listPlayerCT, string
ID){
    string cccd;
    cout << "nhap CCCD : "; getline(cin, cccd);
    for (int i = 0; i < listPlayerCT.size(); i++) {
        if (listPlayerCT[i].getID() == ID) {
            continue;
        }
        if (listPlayerCT[i].getID() == cccd) {
            cout << "so can cuoc cong dan da ton tai!!" << endl;
        }
    }
}
```

```
        return 0;
    }
    if (i == listPlayerCT.size() - 1) {
        this->id = cccd;
        cout << "nhap ho va ten : "; getline(cin, this->fullName);
        cout << "nhap quoc tich : "; getline(cin, this-
>intentional);
        cout << "nhap ngay thang nam sinh : "; getline(cin, this-
>birth);
        cout << "nhap chieu cao : "; getline(cin, this->height);
        cout << "nhap can nang : "; getline(cin, this->weight);
        cout << "nhap vi tri thi dau : "; getline(cin, this-
>position);
        cout << "Thuoc doi bong : "; getline(cin, this->doibong);
        return 2;
    }
}
```

```
void CauThu::getPlayer() const{
    cout << "1. " << "So CCCD : " << this->id << endl;
    cout << "2. " << "Ho va ten : " << this->fullName << endl;
    cout << "3. " << "Quoc tich : " << this->intentional << endl;
    cout << "4. " << "Ngay/Thang/Nam sinh : " << this->birth << endl;
    cout << "5. " << "Chieu cao : " << this->height << endl;
    cout << "6. " << "Can nang : " << this->weight << endl;
    cout << "7. " << "Vi ti thi dau : " << this->position << endl;
}
```

**File CAUTHU.H**

**#pragma once**

**#ifndef CAUTHU\_H**

**#define CAUTHU\_H**

**#include<iostream>**

**#include<vector>**

**#include<string>**

**using namespace std;**

**class CauThu {**

**private:**

**string id;**

**string fullName;**

**string intenational;**

**string birth;**

**string height;**

**string weight;**

**string position;**

**string doibong;**

**public:**

**CauThu(string, string, string, string, string, string, string);**

**CauThu(string, string, string, string, string, string, string, string);**

**CauThu() {};**

**~CauThu();**

**string getID() const;**

**string getFullname() const { return this->fullName; }**

**string getInternational() const { return this->intenational; }**

**string getBirth() const { return this->birth; }**

**string getHeight() const { return this->height; }**

**string getWeight() const { return this->weight; }**

**string getPosition() const { return this->position; }**



```
string getTendoibong() const { return this->doibong; }

    int AddPlayer(vector<CauThu> listPlayerCT);

    void deletePlayer(vector<CauThu>& listPlayerCT);

    void modifyPlayer(vector<CauThu>& listPlayerCT);

    void listingPlayer(vector<CauThu> listPlayerCT);

    void getYoungestPlayer(vector<CauThu>& listPlayerCT);

    void getTallestPlayer(vector<CauThu>& listPlayerCT);

    void getShortestPlayer(vector<CauThu>& listPlayerCT);

    int handleModifyPlayer(vector<CauThu>& listPlayerCT, string ID);

    void getPlayer() const;

};

#endif
```

**File MAIN.CPP**

**#include <iostream>**

**#include<iomanip>**

**#include <vector>**

**#include <string>**

**#include<fstream>**

**#include <sstream>**

**#include"CAUTHU.H"**

**#include"TRANDAU.H"**

**#include"DOIBONG.H"**

**using namespace std;**

**void setDataCTFromdataDb(vector<CauThu>& listPlayerCT, const  
vector<doibong>& danh sach doibong) {**

**for (const auto& doibong : danh sach doibong) {**

**for (const auto& cauthu : doibong.getdscauthu()) {**

**listPlayerCT.push\_back(cauthu);**

**}**

**}**

**}**

**void setCTvaoDB(const CauThu &t, vector<CauThu>  
&danh sach cauthu) {**

**danh sach cauthu.push\_back(t);**

**}**

**void setDatDBFromFile(vector<doibong>& danh Sach Doi Bong) {**

**ifstream file("E:\\dataD\\doibong.txt");**

**if (file.is\_open()) {**

**int sl = 0;**

**string line;**

```
string tenDoiBong, diaPhuong, huanLuyenVien;
string id, fn, it, bi, he, we, po;
while (getline(file, line)) {
    if (line.find("Ten doi bong: ") != string::npos) {
        tenDoiBong = line.substr(14);
    }
    else if (line.find("Dia phuong: ") != string::npos) {
        diaPhuong = line.substr(12);
    }
    else if (line.find("Huan luyen vien: ") != string::npos) {
        huanLuyenVien = line.substr(17);
    }
    vector<CauThu> danhSachcauthu;
    if (line.find("Danh sach cau thu: ") != string::npos) {
        sl = stoi(line.substr(19));
        getline(file, line);
        for (int i = 0; i < sl; i++) {
            for (int j = 0; j < 7; j++) {
                if (line.find("So CMND: ") != string::npos) {
                    id = line.substr(9);
                    getline(file, line);
                }
                else if (line.find("Ho ten: ") != string::npos) {
                    fn = line.substr(8);
                    getline(file, line);
                }
                else if (line.find("Quoc tich: ") != string::npos) {
                    it = line.substr(11);
                }
            }
        }
    }
}
```

```
        getline(file, line);
    }
    else if (line.find("Ngày sinh: ") != string::npos) {
        bi = line.substr(11);
        getline(file, line);
    }
    else if (line.find("Chiều cao: ") != string::npos) {
        he = line.substr(11);
        getline(file, line);
    }
    else if (line.find("Cân nặng: ") != string::npos) {
        we = line.substr(10);
        getline(file, line);
    }
    else if (line.find("Vị trí thi đấu: ") != string::npos) {
        po = line.substr(16);
        getline(file, line);
        getline(file, line);
        CauThu T(id, fn, it, bi, he, we, po, tenDoiBong);
        danhSachcauthu.push_back(T);
    }
}

doibong doibong1(tenDoiBong, diaPhuong, huanLuyenVien,
danhSachcauthu);

danhSachDoiBong.push_back(doibong1);
}
}

file.close();
```

```
    }  
    else {  
        cout << "Khong the mo tep tin doibong.txt." << endl;  
    }  
}
```

```
void setDataTrandauFromFile(vector<TranDau> &danhsachtrandau)  
{  
    ifstream file("E:\\dataD\\trandau.txt");  
    if (file.is_open()) {  
        int d, m, y;  
        string sandau, t1, t2, tyso, line;  
        while (getline(file, line)) {  
            if (line.find("ngay : ") != string::npos) {  
                d = stoi(line.substr(7));  
            }  
            else if (line.find("thang : ") != string::npos) {  
                m = stoi(line.substr(8));  
            }  
            else if (line.find("nam : ") != string::npos) {  
                y = stoi(line.substr(6));  
            }  
            else if (line.find("san dau : ") != string::npos) {  
                sandau = line.substr(10);  
            }  
            else if (line.find("team 1 : ") != string::npos) {  
                t1 = line.substr(9);  
            }  
            else if (line.find("team 2 : ") != string::npos) {
```

```
t2 = line.substr(9);
    }
    else if (line.find("ty so : ") != string::npos) {
        tyso = line.substr(8);
        danh sach tran dau.push_back(TranDau(d, m, y, sandau, t1, t2,
tyso));
    }
}
file.close();
}
else {
    cout << "Khong the mo tep tin doibong.txt." << endl;
}
}
```

```
void menu() {
    cout << "*****" <<
endl;
    cout << "1. Hien thi thong tin 1 cau thu " << endl;
    cout << "2. Them moi cau thu " << endl;
    cout << "3. Xoa thong tin cau thu" << endl;
    cout << "4. Sua thong tin cau thu" << endl;
    cout << "5. Hien thi danh sach cau thu" << endl;
    cout << "6. Hien thi cau thu tre tuoi nhat" << endl;
    cout << "7. Hien thi cau thu cao nhat" << endl;
    cout << "8. Hien thi cau thu thap nhat" << endl;
    cout << "0. Quay lai" << endl;
    cout << "*****" <<
endl;
```

```
    }

    void luuThongTinDoiBong(const vector<doibong>&
danhSachDoiBong) {
        ofstream file("E:\\dataD\\doibong.txt");
        if (file.is_open()) {
            danhSachDoiBong.begin()->getdscauthu();
            for (const auto& doiBong : danhSachDoiBong) {
                file << "Ten doi bong: " << doiBong.getTenDoiBong() << endl;
                file << "Dia phuong: " << doiBong.getDiaPhuong() << endl;
                file << "Huan luyen vien: " << doiBong.getHuanLuyenVien()
<< endl;
                file << "Danh sach cau thu: " << doiBong.getdodaidscauthu()
<< endl;
                for (const auto& cauThu : doiBong.getdscauthu()) {
                    file << "So CMND: " << cauThu.getID() << endl;
                    file << "Ho ten: " << cauThu.getFullname() << endl;
                    file << "Quoc tich: " << cauThu.getInternational() << endl;
                    file << "Ngay sinh: " << cauThu.getBirth() << endl;
                    file << "Chieu cao: " << cauThu.getHeight() << endl;
                    file << "Can nang: " << cauThu.getWeight() << endl;
                    file << "Vi tri thi dau: " << cauThu.getPosition() << endl;
                    file << "-----" << endl;
                }
                file << "-----" << endl;
            }
            file.close();
        }
        else {
```

```
    cout << "Khong the mo tep tin doibong.txt." << endl;
}
}

void luuThongTinCauThu(const vector<CauThu>& listPlayerCT) {
    ofstream file("E:\\dataD\\dsplayer.txt");
    if (file.is_open()) {
        for (const auto& cauThu : listPlayerCT) {
            file << "So CMND: " << cauThu.getID() << endl;
            file << "Ho ten: " << cauThu.getFullname() << endl;
            file << "Quoc tich: " << cauThu.getInternational() << endl;
            file << "Ngay sinh: " << cauThu.getBirth() << endl;
            file << "Chieu cao: " << cauThu.getHeight() << endl;
            file << "Can nang: " << cauThu.getWeight() << endl;
            file << "Vi tri thi dau: " << cauThu.getPosition() << endl;
            file << "Thuoc doi bong: " << cauThu.getTendoibong() << endl;
            file << "-----" << endl;
        }
        file.close();
    }
    else {
        cout << "Khong the mo tep tin doibong.txt." << endl;
    }
}

int main() {
    vector<TranDau> danhSachtrandau;
    vector<doibong> danhSachDoiBong;
```



```
vector<CauThu> listPlayerCT;
setDatDBFromFile(danhSachDoiBong);
setDataCTFromdataDb(listPlayerCT, danhSachDoiBong);
setDataTrandauFromFile(danhSachtrandau);
luuThongTinCauThu(listPlayerCT);
int luaChon;

do {
    cout << "===== Menu =====" << endl;
    cout << "1. Quan ly Doi bong" << endl;
    cout << "2. Quan ly Cau thu" << endl;
    cout << "3. Quan ly Tran dau" << endl;
    cout << "4. Tinh diem cho cac doi bong" << endl;
    cout << "5. Thoat" << endl;
    cout << "===== " << endl;
    cout << "Nhap lua chon cua ban (1-5): ";
    cin >> luaChon;

    switch (luaChon) {
    case 1:
        //{
        cout << "1. Them Doi bong" << endl;
        cout << "2. Sua Doi bong" << endl;
        cout << "3. Xoa Doi bong" << endl;
        cout << "4. Hien thi thong tin 1 doi bong" << endl;
        int luaChonDoiBong;
        cout << "Nhap lua chon cua ban (1-4): ";
        cin >> luaChonDoiBong;
```

```
        if (luaChonDoiBong == 1) {
            string tenDoiBong, diaPhuong, huanLuyenVien;
            cout << "Nhap ten doi bong: ";
            cin.ignore();
            getline(cin, tenDoiBong);
            cout << "Nhap dia phuong: ";
            getline(cin, diaPhuong);
            cout << "Nhap ten huan luyen vien: ";
            getline(cin, huanLuyenVien);
            int dem;
            vector<CauThu> danhSachcauthu;
            cout << "Nhap danh sach cau thu: " << endl;
            cout << "So luong cau thu can them: ";
            cin >> dem;
            cin.ignore();
            for (int i = 0; i < dem; i++) {
                string id, ten, quoctich, sinhnhat, chieucao, cannang,
vitritd;

                cout << "Nhap cau thu thu: " << i + 1 << endl;
                cout << "nhap CCCD : "; getline(cin, id);
                for (int j = 0; j < listPlayerCT.size(); j++) {

                    if (listPlayerCT[j].getID() == id) {
                        cout << "so can cuoc cong dan da ton tai!!" << endl <<
"Vui long nhap lai." << endl;
                        i--;

                        break;
                    }
                }
            }
        }
    }
}
```

```
    }

    if (j == listPlayerCT.size() - 1) {
        cout << "nhap ho va ten : "; getline(cin, ten);
        cout << "nhap quoc tich : "; getline(cin, quoctich);
        cout << "nhap ngay thang nam sinh : "; getline(cin,
sinhnhat);

        cout << "nhap chieu cao : "; getline(cin, chieucao);
        cout << "nhap can nang : "; getline(cin, cannang);
        cout << "nhap vi tri thi dau : "; getline(cin, vitritd);
        danhsachcauthu.push_back(CauThu(id, ten, quoctich,
sinhnhat, chieucao, cannang, vitritd, tenDoiBong));
        listPlayerCT.push_back(CauThu(id, ten, quoctich,
sinhnhat, chieucao, cannang, vitritd, tenDoiBong));
        break;
    }
}

    danhSachDoiBong.push_back(doibong(tenDoiBong,
diaPhuong, huanLuyenVien, danhsachcauthu));
    cout << "Them doi bong thanh cong." << endl;

    auto i = danhSachDoiBong.end() - 1;
    cout << i->getTenDoiBong();

    // LẤY THÔNG TIN VÀO TẬP TIN
    lưuThôngTinDoiBong(danhSachDoiBong);
    lưuThôngTinCauThu(listPlayerCT);
}
```

```
else if (luaChonDoiBong == 2) {
    string tenDoiBong;
    cout << "Nhap ten doi bong can sua: ";
    cin.ignore();
    getline(cin, tenDoiBong);

    // Tìm kiếm tên bóng cần sửa
    int index = -1;
    for (int i = 0; i < danhSachDoiBong.size(); ++i) {
        if (danhSachDoiBong[i].getTenDoiBong() == tenDoiBong) {
            index = i;
            break;
        }
    }

    if (index != -1) {
        string diaPhuong, huanLuyenVien;
        int luaChon;
        cout << "Nhap dia phuong moi: ";
        getline(cin, diaPhuong);
        cout << "Nhap ten huan luyen vien moi: ";
        getline(cin, huanLuyenVien);
        cout << "1. Them cau thu vao danh sach." << endl;
        cout << "2. Xoa cau thu khoi danh sach." << endl;
        cout << "0. Khong thay doi danh sach cau thu." << endl;
        cin >> luaChon;
        switch (luaChon) {
            case 1:
```

```
        {  
            string id, ten, quoctich, sinhnhat, chieucao, cannang,  
vitritd;  
  
            cout << "nhap CCCD : ";  
            cin.ignore();  
            getline(cin, id);  
            for (int i = 0; i <  
danhSachDoiBong[index].getdscauthu().size(); i++) {  
                if (danhSachDoiBong[index].getdscauthu()[i].getID()  
== id) {  
                    cout << "So CCCD da ton tai!" << endl;  
                    break;  
                }  
                if (i == danhSachDoiBong[index].getdscauthu().size() -  
1) {  
                    cout << "nhap ho va ten : "; getline(cin, ten);  
                    cout << "nhap quoc tich : "; getline(cin, quoctich);  
                    cout << "nhap ngay thang nam sinh : "; getline(cin,  
sinhnhat);  
  
                    cout << "nhap chieu cao : "; getline(cin, chieucao);  
                    cout << "nhap can nang : "; getline(cin, cannang);  
                    cout << "nhap vi tri thi dau : "; getline(cin, vitritd);  
                    CauThu cauthu(id, ten, quoctich, sinhnhat,  
chieucao, cannang, vitritd, tenDoiBong);  
                    listPlayerCT.push_back(cauthu);  
  
                    danhSachDoiBong[index].themcauthuvaodanhhsach(cauthu);  
                }  
            }  
            break;
```

```
    }

    case 2:
        string id;
        cin.ignore();
        cout << "nhap CCCD cau thu can xoa: "; getline(cin,
id);

        danhSachDoiBong[index].xoacauthukhoids(id);
        break;
    }

    //Cá°-p nhá°-t thă°ng tin Ä‘á»™i bă°ng
    danhSachDoiBong[index].suadoibong(tenDoiBong,
diaPhuong, huanLuyenVien);

    // Lă°u thă°ng tin vă° o tá»p tin
    luuThongTinDoiBong(danhSachDoiBong);
    luuThongTinCauThu(listPlayerCT);

    cout << "Sua thong tin doi bong thanh cong." << endl;
}
else {
    cout << "Khong tim thay doi bong co ten " << tenDoiBong
<< "." << endl;
}
}

// Xá»- lă°½ xă°a Ä‘ á»™i bă°ng
else if (luaChonDoiBong == 3) {
    string tenDoiBong;
```

```
    cout << "Nhap ten doi bong can xoa: ";
    cin.ignore();
    getline(cin, tenDoiBong);

    int index = -1;
    for (int i = 0; i < danhSachDoiBong.size(); ++i) {
        if (danhSachDoiBong[i].getTenDoiBong() == tenDoiBong) {
            index = i;
            break;
        }
    }

    if (index != -1) {
        // Xóa phần tử khỏi danh sách
        danhSachDoiBong.erase(danhSachDoiBong.begin() +
index);

        // Lưu thông tin vào tệp tin
        lưuThôngTinDoiBong(danhSachDoiBong);

        cout << "Xóa đôi bóng thành công." << endl;
    }
    else {
        cout << "Không tìm thấy đôi bóng có tên " << tenDoiBong
<< "." << endl;
    }
}

else if (luaChonDoiBong == 4) {
    string tenDB;
```

```
    cout << "Nhap ten doi bong can tim :";
    cin.ignore();
    getline(cin, tenDB);
    for (auto i = danhSachDoiBong.begin(); i !=
danhSachDoiBong.end(); i++) {
        if (i->getTenDoiBong() == tenDB) {
            i->inThongTin();
            break;
        }
        if (i == danhSachDoiBong.end() - 1) {
            cout << "Khong tim thay!" << endl;
        }
    }
}
else {
    cout << "Lua chon khong hop le." << endl;
}
break;
//}

case 2:
    // xá»- lă½ quá°£n lă½ cá°§u thá»§
    {
        int select;
        menu();
        cout << "-----";
        cout << "\nchon so de thuc hien : ";
        cin >> select;
        cin.ignore();
        switch (select) {
```



**case 1:**

```
{  
    string cmnd;  
    cout << "vui long nhap CCCD can tim : ";  
    cin >> cmnd;  
    for (int i = 0; i < listPlayerCT.size(); i++) {  
        if (cmnd == listPlayerCT[i].getID()) {  
            listPlayerCT[i].getPlayer();  
            break;  
        }  
        if (i == listPlayerCT.size() - 1) {  
            cout << "khong tim thay!!!!!!!!!!" << endl;  
        }  
    }  
}  
break;
```

**case 2:**

```
{  
    CauThu t;  
    int check;  
    do {  
        check = t.AddPlayer(listPlayerCT);  
        if (check == 2) {  
            listPlayerCT.push_back(t);  
            for (auto &db : danhSachDoiBong) {  
                if (t.getTendoibong() == db.getTenDoiBong()) {  
                    db.themcauthuvaodanh sach(t);  
                    luuThongTinDoiBong(danhSachDoiBong);  
                }  
            }  
        }  
    } while (check != 0);  
}
```

```
        }
        }
        cout << "nhap du lieu thanh cong !" << endl;
        break;
    }
} while (check);
luuThongTinCauThu(listPlayerCT);
}
break;
case 3:
{
    string cccd;
    cout << "vui long nhap CCCD cau thu can xoa : ";
    cin >> cccd;
    for (auto i = listPlayerCT.begin(); i != listPlayerCT.end();
i++) {
        if (cccd == i->getID()) {
            for (auto& db : danhSachDoiBong) {
                if (db.getTenDoiBong() == i->getTendoibong()) {
                    db.xoacauthukhoids(cccd);
                }
            }
            listPlayerCT.erase(i);
            cout << "Xoa thanh cong !" << endl;
            break;
        }
        if (i == listPlayerCT.end() - 1) {
            cout << "khong tim thay!!!!!!!!!!" << endl;
        }
    }
}
```

```
    }  
    luuThongTinDoiBong(danhSachDoiBong);  
    luuThongTinCauThu(listPlayerCT);  
}  
break;  
case 4:  
{  
    string id;  
    CauThu t;  
    id = t.modifyPlayer(listPlayerCT);  
    for (auto& ct : listPlayerCT) {  
        if (ct.getID() == id) {  
            for (auto& db : danhSachDoiBong) {  
                if (db.getTenDoiBong() == ct.getTendoibong()) {  
                    db.suadanh sachcauthu(ct);  
                }  
            }  
        }  
    }  
    luuThongTinDoiBong(danhSachDoiBong);  
    luuThongTinCauThu(listPlayerCT);  
}  
break;  
case 5:  
{  
    CauThu t;  
    t.listingPlayer(listPlayerCT);  
}
```

```
break;
    case 6:
    {
        CauThu t;
        cout << "cau thu tre nhat la : " << endl;
        t.getYoungestPlayer(listPlayerCT);
    }
    break;
    case 7:
    {
        CauThu t;
        cout << "cau thu cao nhat la : " << endl;
        t.getTallestPlayer(listPlayerCT);
    }
    break;
    case 8:
    {
        CauThu t;
        cout << "cau thu thap nhat la : " << endl;
        t.
            getShortestPlayer(listPlayerCT);
    }
    break;
    }
break;
case 3:
    // xá»- lă½ quă°£n lă½ trá°-n Ä‘á°Yu
```

```
{  
    cout << "-----" << endl;  
    cout << "1. Xem cac tran dau trong 1 ngay" << endl;  
    cout << "2 .Xem cac tran dau trong 1 thang" << endl;  
    cout << "3. Xem tat ca tran dau" << endl;  
    cout << "4. Them tran dau" << endl;  
    cout << "0. Quay lai." << endl;  
  
    int select;  
    cout << "Moi chon : ";  
    cin >> select;  
    switch (select) {  
    case 1 :  
    {  
        int day;  
        cout << "Nhap ngay can xem : ";  
        cin >> day;  
        for (auto& td : danh sach tran dau) {  
            if (td.getNgaythidau() == day) {  
                td.inThongtin();  
            }  
        }  
    }  
    break;  
    case 2:  
    {  
        int thang;  
        cout << "Thang can xem : ";
```

```
cin >> thang;
    for (auto& td : danhsachtrandau) {
        if (td.getThangthidau() == thang) {
            td.inThongtin();
        }
    }
}
break;
case 3:
{
    for (auto& td : danhsachtrandau) {
        td.inThongtin();
    }
}
break;
case 4:
{
    int ngay, thang, nam;
    string sandau, t1, t2, tyso;
    cout << "Nhap ngay thi dau : "; cin >> ngay;
    cout << "Nhap thang thi dau : "; cin >> thang;
    cout << "Nhap nam thi dau : "; cin >> nam;
    cin.ignore();
    cout << "Nhap san thi dau : "; getline(cin, sandau);
    cout << "Nhap team 1 : "; getline(cin, t1);
    cout << "Nhap team 2 : "; getline(cin, t2);
    cout << "Nhap ty so : "; getline(cin, tyso);
    danhsachtrandau.push_back(TranDau(ngay, thang, nam,
sandau, t1, t2, tyso));
```

```
    }  
        break;  
    case 0:  
        break;  
    default:  
        cout << "Lua chon khong hop le!!!" << endl;  
    }  
}  
    break;  
case 4:  
{  
    struct dsdb {  
        string ten;  
        int diem;  
    };  
    vector<dsdb> _danhsachdoibong;  
    for (auto& db : danhSachDoiBong) {  
        dsdb t;  
        t.ten = db.getTenDoiBong();  
        t.diem = 0;  
        _danhsachdoibong.push_back(t);  
    }  
  
    for (auto& td : danhsachtrandau) {  
        int t1, t2;  
        t1 = stoi(td.getTyso().substr(0));  
        t2 = stoi(td.getTyso().substr(2));  
        if (t1 > t2) {
```

```
        for (auto& db : _danhsachdoibong) {
            if (td.getTeam1() == db.ten) {
                db.diem += 3;
            }
        }
    }
    else if (t1 < t2) {
        for (auto& db : _danhsachdoibong) {
            if (td.getTeam2() == db.ten) {
                db.diem += 3;
            }
        }
    }
    else {
        for (auto& db : _danhsachdoibong) {
            if (td.getTeam1() == db.ten) {
                db.diem += 1;
            }
        }
        for (auto& db : _danhsachdoibong) {
            if (td.getTeam2() == db.ten) {
                db.diem += 1;
            }
        }
    }
    cout << setw(30) << left << "Ten doi bong " << "Diem" <<
endl;

    for (auto& db : _danhsachdoibong) {
```



```
        cout << setw(30) << left << db.ten << db.diem << endl;
    }
}
break;
case 5:
    cout << "thoat khoi chuong trinh." << endl;
    break;
default:
    cout << "lua chon khong hop le. vui long chon lai." << endl;

}
} while (luaChon != 5);

return 0;
}
```