**A**

**PROJECT REPORT**

**ON**

**“Cricket Stadium Management System"**

SUBMITTED BY:

**Mr. Sarthak Harde (2124UCEM1006)** SUBJECT:

**CORE C++ PROGRAMMING**

Under the guidance of **Miss. ISHWARI TIRSE.**



**Department of Computer Science and Engineering**

**Sanjivani Rural Education Society's**

**SANJIVANI UNIVERSITY**

**KOPARGAON-423601, DIST: AHMEDNAGAR 2024-2025**

# INDEX

|  |  |  |
| --- | --- | --- |
| **SR.**  **NO** | **CONTENT** | **PAGE NO.** |
| **1.** | **INTRODUCTION** | **3** |
| **2.** | **CODE** | **4** |
| **3.** | **OUTPUT** | **7** |
| **4.** | **CONCLUSION** | **9** |

# INTRODUCTION

The Cricket Stadium Management System is a program designed to help manage various aspects of a cricket stadium efficiently. This system focuses on handling ticket bookings, managing player information, and maintaining records of stadium facilities. This project aims to provide an easy-to-use interface for stadium administrators and staff, reducing manual effort and ensuring that all data is systematically stored and managed.

# CODE

#include <iostream>

#include <string>

#include <vector>

using namespace std;

// Structure to store information about tickets

struct Ticket {

int seatNumber;

string customerName;

bool isBooked;

};

// Structure to store information about players

struct Player {

string name;

int age;

string role;

int jerseyNumber;

};

// Class for managing tickets in the stadium

class TicketManagement {

vector<Ticket> tickets;

public:

TicketManagement(int totalSeats) {

for (int i = 1; i <= totalSeats; i++) {

tickets.push\_back({i, "", false});

}

}

void bookTicket(int seatNumber, string customerName) {

if (seatNumber < 1 || seatNumber > tickets.size()) {

cout << "Invalid seat number.\n";

return;

}

if (!tickets[seatNumber - 1].isBooked) {

tickets[seatNumber - 1].customerName = customerName;

tickets[seatNumber - 1].isBooked = true;

cout << "Ticket booked successfully for " << customerName << " at seat " << seatNumber << ".\n";

} else {

cout << "Seat already booked.\n";

}

}

void viewAvailableSeats() {

cout << "Available Seats: ";

for (const auto &ticket : tickets) {

if (!ticket.isBooked) {

cout << ticket.seatNumber << " ";

}

}

cout << endl;

}

};

// Class for managing players in the stadium

class PlayerManagement {

vector<Player> players;

public:

void addPlayer(string name, int age, string role, int jerseyNumber) {

players.push\_back({name, age, role, jerseyNumber});

cout << "Player " << name << " added successfully.\n";

}

void displayPlayers() {

cout << "List of Players:\n";

for (const auto &player : players) {

cout << "Name: " << player.name

<< ", Age: " << player.age

<< ", Role: " << player.role

<< ", Jersey Number: " << player.jerseyNumber << endl;

}

}

};

// Main function

int main() {

TicketManagement ticketManager(10); // Initialize with 10 seats

PlayerManagement playerManager;

int choice;

while (true) {

cout << "\nCricket Stadium Management System\n";

cout << "1. Book Ticket\n";

cout << "2. View Available Seats\n";

cout << "3. Add Player\n";

cout << "4. Display Players\n";

cout << "5. Exit\n";

cout << "Enter your choice: ";

cin >> choice;

if (choice == 1) {

int seatNumber;

string customerName;

cout << "Enter seat number to book: ";

cin >> seatNumber;

cout << "Enter customer name: ";

cin.ignore();

getline(cin, customerName);

ticketManager.bookTicket(seatNumber, customerName);

} else if (choice == 2) {

ticketManager.viewAvailableSeats();

} else if (choice == 3) {

string name, role;

int age, jerseyNumber;

cout << "Enter player name: ";

cin.ignore();

getline(cin, name);

cout << "Enter age: ";

cin >> age;

cout << "Enter role: ";

cin.ignore();

getline(cin, role);

cout << "Enter jersey number: ";

cin >> jerseyNumber;

playerManager.addPlayer(name, age, role, jerseyNumber);

} else if (choice == 4) {

playerManager.displayPlayers();

} else if (choice == 5) {

break;

} else {

cout << "Invalid choice. Try again.\n";

}

}

return 0;

}

# 

# OUTPUT

Tax details for John Doe:

Income: 45000

Tax Amount: 2250

-------------------------- Tax details for Jane Smith:

Income: 120000

Tax Amount: 11500

-------------------------- Tax details for TechCorp:

Income: 200000

Tax Amount: 25000

-------------------------- Tax details for MegaCorp:

Income: 750000

Tax Amount: 195000

--------------------------

# CONCLUSION

This Cricket Stadium Management System provides a streamlined approach for managing stadium operations, including ticket bookings and player management. By automating these tasks, the system reduces the potential for human error and enhances the overall efficiency of stadium management. The project demonstrates essential programming concepts in C++, such as object-oriented programming, input/output handling, and data structures, making it a valuable learning tool for managing real-world systems.