UE21CS351A: Database Management System

MINI PROJECT USER REQUIREMENT SPECIFICATION
STADIUM MANAGEMENT SYSTEM

ARVIN NOOLI   **ASHLIN FURTADO**
PES1UG21CS112   **PES1UG21CS119**

Introduction:

1.1 Purpose of the project:

The Stadium Management System will serve as a comprehensive solution for efficiently managing various aspects of a stadium or sports facility. Its primary purpose is to enhance the overall ticketing experience and streamline the various processes associated with managing a stadium.

By integrating all the functionalities that we plan on putting up, our Stadium Management System aims to create a more efficient, secure, and enjoyable experience for spectators, organisers, and Event managers. It's a vital tool for modern sports and entertainment venues looking to optimise their operations and provide a seamless experience for all stakeholders.

1.2 Scope of the project:

The scope of a Stadium Management System project encompasses the comprehensive optimization of sports and entertainment venue operations. This multifaceted system seamlessly integrates ticketing, event scheduling, and parking management.By efficiently managing these aspects, the project aims to enhance overall stadium functionality, ensuring seamless event planning and an enriched fan experience. The system's scope extends to sponsorship and marketing coordination, enabling revenue generation. With a focus on technology-driven solutions, this project addresses the diverse needs of modern stadiums, promoting operational efficiency and elevating the overall quality of events and services provided.

2.Project Description
Project Overview:

The Stadium Management System is a comprehensive software solution designed to streamline and enhance the operations of a stadium or sports venue. It offers a range of functionalities to efficiently manage ticket sales, event scheduling, parking, and vendor and staff management. The system is intended to improve the overall experience of both event organizers and attendees.

Key Features:

1.Ticket Sales Management:
- Ticket booking and purchase.

- Seat selection and reservation.

- Support for different ticket types (e.g., VIP, standard).

2.Event Scheduling:
- Calendar-based event scheduling and management.
- Easy event creation, modification, and cancellation.
- Integration with the ticketing system to automatically update event availability.

3.Parking Management:
- Automated parking space allocation.
- Real-time parking availability updates on the stadium's application.
- Integration with ticketing system for reserved parking spots.
- Payment processing for parking fees.

4.Vendor Management:
- Vendor registration and onboarding process.
- Vendor performance tracking and ratings.

5.Staff Management:
- Employee scheduling and shift management.

6.Security and Access Control:
Role-based access control for administrators, vendors, and staff.

3.Function Requirements

1.HTML:HTML is used to structure and format web pages. It provides the foundation for displaying event information, ticket sales, and other content on the system's user interface, ensuring cross-platform compatibility and accessibility for stadium attendees and staff.

2.CSS: To control the visual presentation and layout of web pages. CSS ensures that the system's user interface, including fonts, colours, spacing, and responsiveness, is visually appealing and consistent across various devices and screen sizes, enhancing the overall user experience.

3.FLASK: We'll use Flask for backend to run MySQL and add functionalities .Flask is characterised by its simplicity, which means it offers only what's essential for building web applications, making it approachable and easy to learn. Its minimalist design keeps the core framework lightweight, allowing developers to add only the components they need, reducing unnecessary complexity.

4.MYSQL: For the database we'll use MySQl ,because it is easy to use ,secure,has high efficiency and it's open source.

4.ENTITY RELATIONSHIP DIAGRAM

Entity Relationship Diagram - Stadium Management System
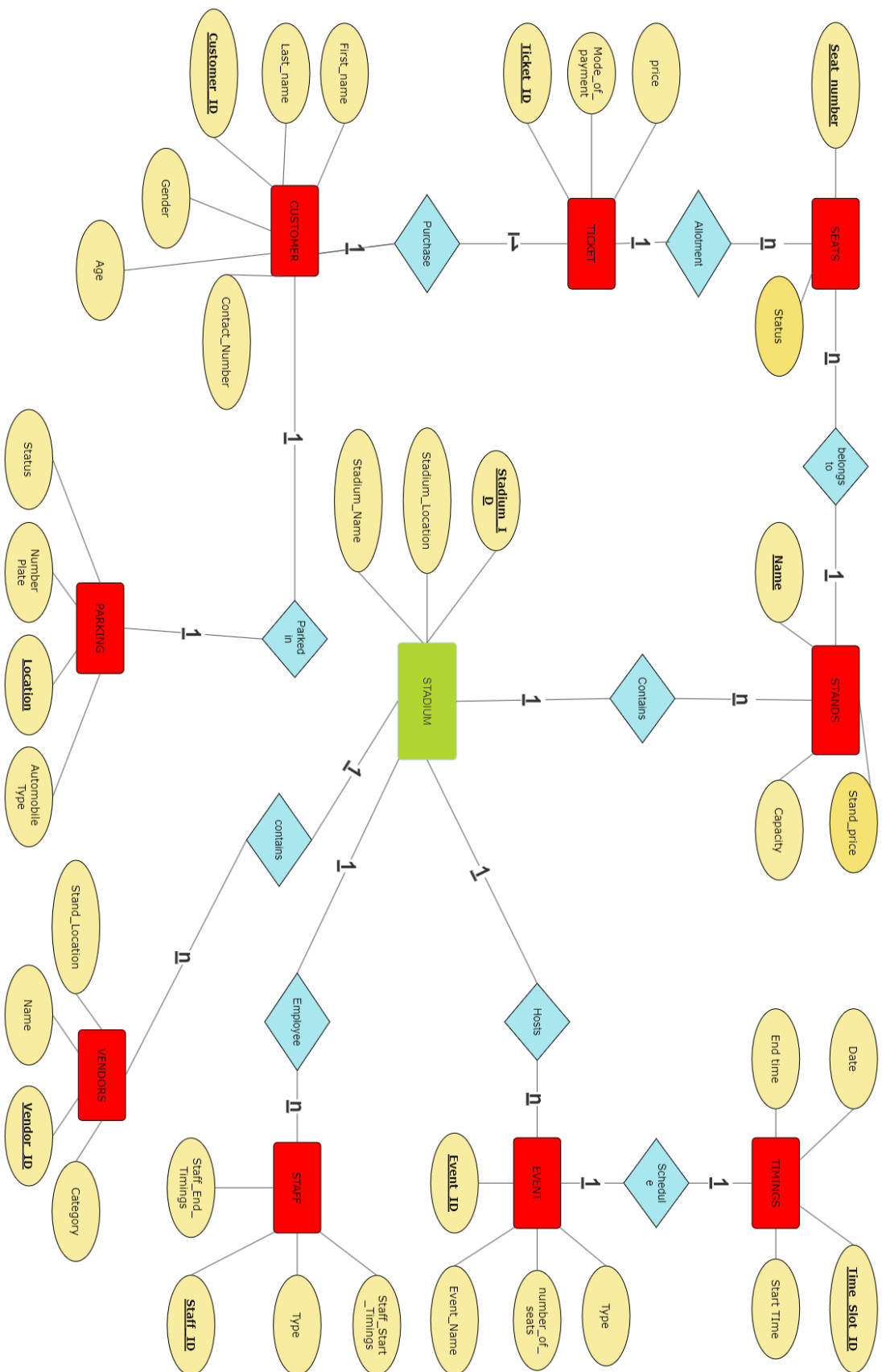
5.Relational schema:

Customer:

Customer _Id	First_name	Last_name	Gender	Age	Contact info	Ticket_id	Location

Parking:

Location	Number_plate	Status	automobile_type

Timing:

time_slot_ID	Date	Start_time	End_time

Ticket:

Ticket_ID	mode_of_payment	price

Seats:

Seat_ID	Status	Ticket_ID	Stand_name

Stadium

Stadium_ID	Stadium_name	Stadium_location

Event:

Event_ID	Event_name	no._of _seats	type	stadium_ID	Time_slot_ID

Staff:

Staff_ID	Staff_start_time	Staff_end_time	type	Stadium_ID

Stands:

Stand_Name	Stand_price	Stadium_ID

Vendors:

Vendor_ID	Stand_location	Name	Category	Stadium_ID