CSE302: Database Systems (Section No.7) [FALL 25]

Project Report

https://github.com/MahAbubALamBadHon/ewu_event_management/upload/main
https://github.com/kamrulhasan-arch/Ewu-event-management
https://github.com/sanzidasultanarupai/EWU-event-management

EWU Event Management WebApp

Submitted by (Group8):

Student ID	Student Name	Contribution Percentage
2023-3-60-130	**Mahabub Alam Badhon**	40
2023-3-60-148	Kamrul Hasan	30
2023-2-50-007	Sanzida Sultana Rupai	30

1. Project Description

The objective of this project is to design and implement a **web-based event management system** for East West University (EWU), enabling:

- Efficient management of events, venues, meals, bookings, and finances.
- Role-based access for **Administrators** and **Customers**.
- Real-time validation of bookings, payments, and guest limits.
- Automated financial recording and reporting.

---

2. Key Features

Administrative Functions:

- Add, edit, and delete events
- Manage venues and meal packages
- Monitor bookings and payments
- View reports (event summary, booking details)

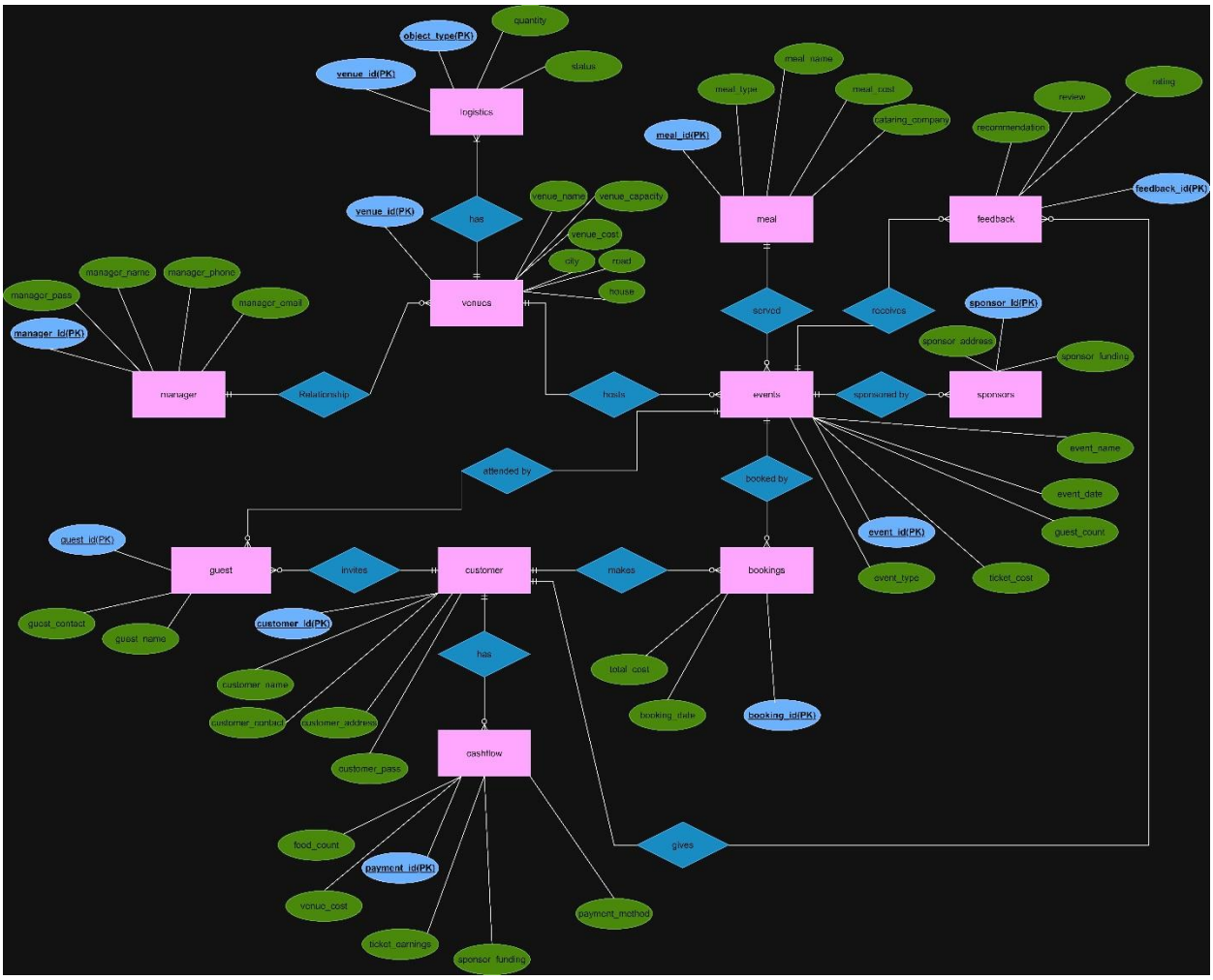
Customer Functions:

- Register and login securely
- View events and book tickets
- Receive booking confirmation and payment details
- View personal booking history

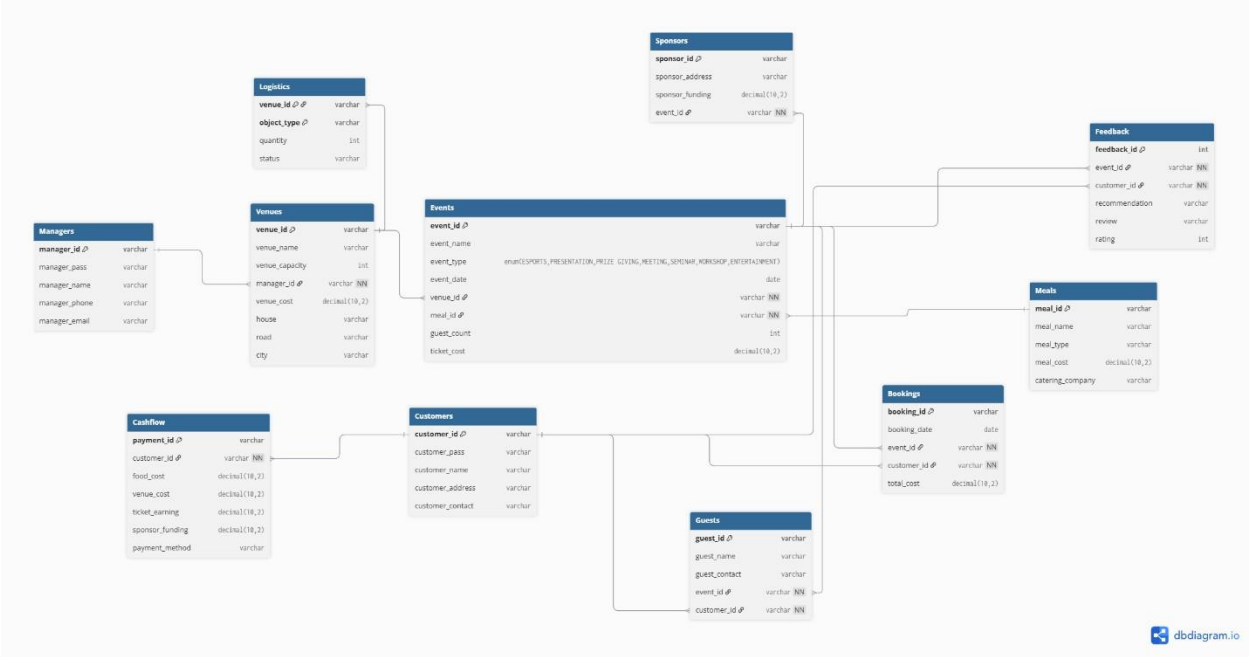
---

3. Database Design

3.1 E-R Model


object_type(PK)
quantity
venue_id(PK)
status
logistics
has
venue_id(PK)
venue_name
venue_capacity
venue_cost
city
road
house
venues
manager_name
manager_phone
manager_pass
manager_email
manager_id(PK)
manager
Relationship
hosts
meal_name
meal_type
meal_cost
catering_company
meal_id(PK)
meal
served
receives
feedback
review
rating
recommendation
feedback_id(PK)
sponsor_id(PK)
sponsor_address
sponsor_funding
sponsors
sponsored by
events
event_name
event_date
guest_count
event_id(PK)
event_type
ticket_cost
attended by
booked by
guest_id(PK)
guest
invites
customer
makes
bookings
guest_contact
guest_name
customer_id(PK)
customer_name
customer_contact
customer_address
customer_pass
has
cashflow
total_cost
booking_date
booking_id(PK)
food_count
payment_id(PK)
venue_cost
ticket_earnings
sponsor_funding
payment_method
gives

3.2 Relational Data Model



MANAGER
├── manager_id VARCHAR(5) [PRIMARY KEY]
├── manager_pass VARCHAR(60) [NOT NULL]
├── manager_name VARCHAR(100)
├── manager_phone VARCHAR(20)
└── manager_email VARCHAR(100)

VENUES
├── venue_id VARCHAR(5) [PRIMARY KEY]
├── venue_name VARCHAR(100)
├── venue_capacity INT
├── manager_id VARCHAR(5) [FOREIGN KEY → Managers.manager_id]
├── venue_cost DECIMAL(12,2)
├── house VARCHAR(100) ├──
road VARCHAR(100)
└── city VARCHAR(100)

MEALS
├── meal_id VARCHAR(5) [PRIMARY KEY]
├── meal_name VARCHAR(100)
├── meal_type VARCHAR(50)
├── meal_cost DECIMAL(12,2)
└── catering_company VARCHAR(100)

EVENTS
├── event_id VARCHAR(5) [PRIMARY KEY]
├── event_name VARCHAR(100)
├── event_type ENUM('ESPORTS','PRESENTATION','PRIZE GIVING','MEETING','SEMINAR','WORKSHOP','ENTERTAINMENT')
├── event_date DATE
├── venue_id VARCHAR(5) [FOREIGN KEY → Venues.venue_id]
├── meal_id VARCHAR(5) [FOREIGN KEY → Meals.meal_id]
├── guest_count INT
├── ticket_cost DECIMAL(12,2)
└── UNIQUE CONSTRAINT (venue_id, event_date, event_type)

CUSTOMERS
├── customer_id VARCHAR(5) [PRIMARY KEY]
├── customer_pass VARCHAR(60) [NOT NULL]
├── customer_name VARCHAR(100)
├── customer_address VARCHAR(200)
└── customer_contact VARCHAR(20)

BOOKINGS
├── booking_id VARCHAR(5) [PRIMARY KEY]
├── booking_date DATE
├── event_id VARCHAR(5) [FOREIGN KEY → Events.event_id CASCADE DELETE] ├── customer_id VARCHAR(5) [FOREIGN KEY → Customers.customer_id CASCADE DELETE]
└── total_cost DECIMAL(12,2)

GUESTS
├── guest_id VARCHAR(5) [PRIMARY KEY]
├── guest_name VARCHAR(100)
├── guest_contact VARCHAR(20)

├── event_id VARCHAR(5) [FOREIGN KEY → Events.event_id CASCADE DELETE]
└── customer_id VARCHAR(5) [FOREIGN KEY → Customers.customer_id CASCADE DELETE]

SPONSORS
├── sponsor_id VARCHAR(5) [PRIMARY KEY]
├── sponsor_address VARCHAR(200) ├── sponsor_funding DECIMAL(12,2)
└── event_id VARCHAR(5) [FOREIGN KEY → Events.event_id CASCADE DELETE]

CASHFLOW
├── payment_id VARCHAR(5) [PRIMARY KEY]
├── customer_id VARCHAR(5) [FOREIGN KEY → Customers.customer_id CASCADE DELETE]
├── food_cost DECIMAL(12,2)
├── venue_cost DECIMAL(12,2)
├── ticket_earning DECIMAL(12,2)
├── sponsor_funding DECIMAL(12,2)
└── payment_method VARCHAR(50)

FEEDBACK
├── feedback_id INT [PRIMARY KEY AUTO_INCREMENT]
├── event_id VARCHAR(5) [FOREIGN KEY → Events.event_id CASCADE DELETE] ├── customer_id VARCHAR(5) [FOREIGN KEY → Customers.customer_id CASCADE DELETE]
├── recommendation VARCHAR(100)
├── review VARCHAR(500)
└── rating INT

LOGISTICS
├── venue_id VARCHAR(5) [FOREIGN KEY → Venues.venue_id CASCADE DELETE]
├── object_type VARCHAR(50)
├── quantity INT
├── status VARCHAR(50)
└── PRIMARY KEY (venue_id, object_type)

---

4. Tools & Technologies Used

1. HTML,CSS(Front end)
2. PHP (Back end)
3. MySql(Database)
4. XAMPP (for local hosting)

5. Role Assignment

Administrator (Manager / Admin)
Login: Manager ID and password (manager table)
Access: Full system control
Responsibilities:

- Manage events (events), venues (venues), and meal packages (meals)
- Oversee bookings (bookings) and guest lists (guests)
- Track finances (cashflow) and sponsors (sponsors)
- Manage logistics (logistics) and equipment
- Generate reports and analytics (event_summary, booking_details)

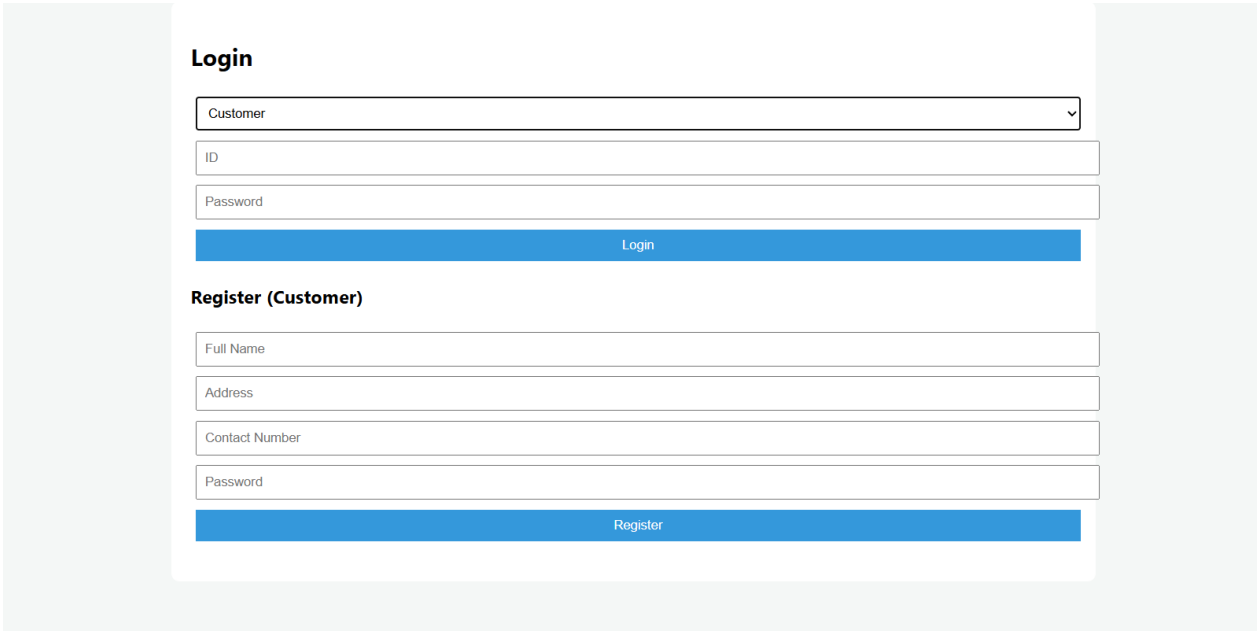
Customer (Students)
Login: Customer ID and password (customers table)
Access: Personal account
Responsibilities:

- Manage personal profile (customers)
- Browse and book events (events), manage guests (guests)
- Make payments (cashflow)
- View booking history (bookings) and event dashboard
- Submit feedback and ratings (feedback)

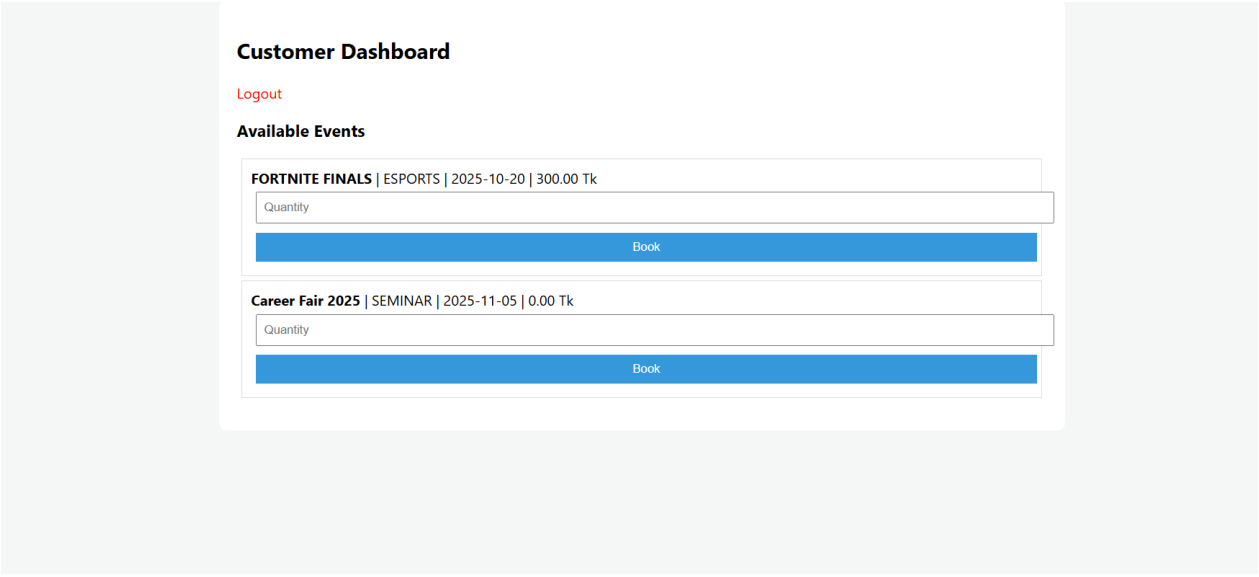
---

6. GUI Screenshot Home

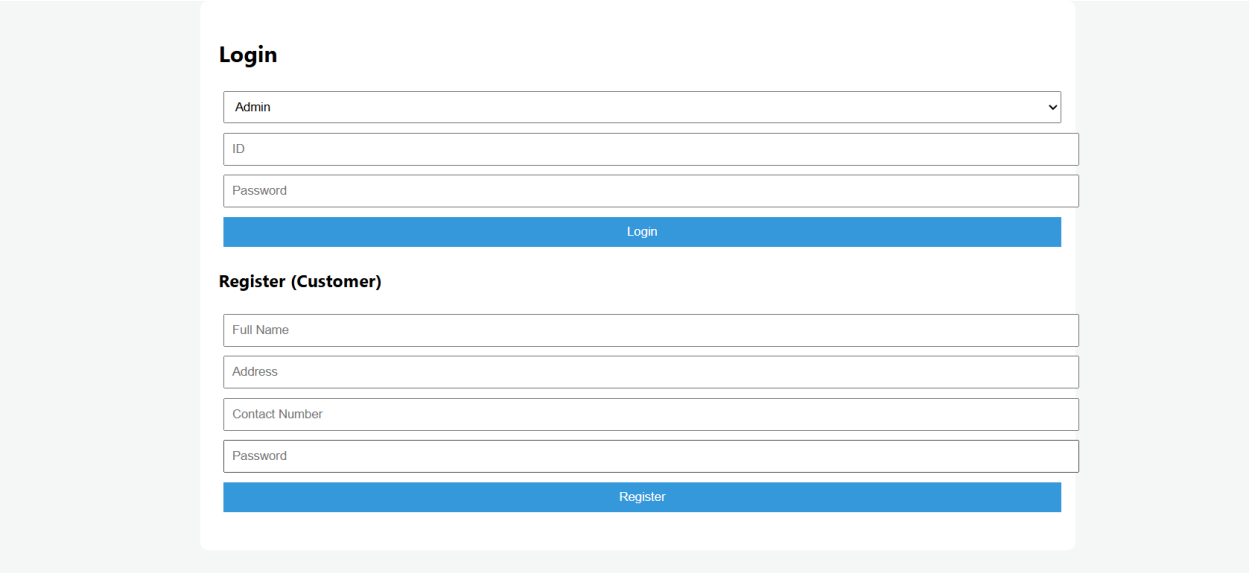
Student/Customer Login
Page:



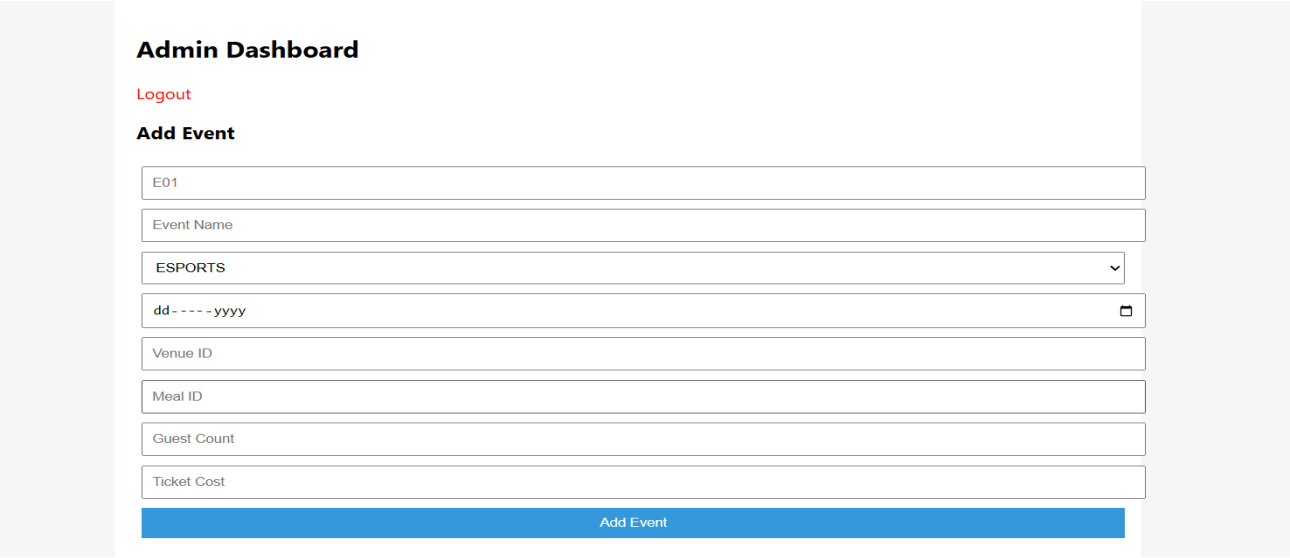
STUDENT/CUSTOMER DASHBORD:



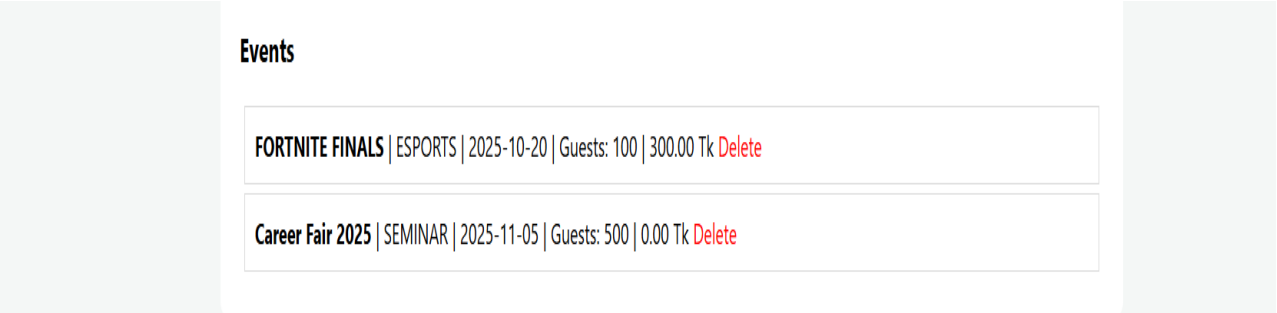
Admin Side:



Event Creation management page:



Events and Details:



7. Conclusion

This project successfully implemented a comprehensive event management system that demonstrated practical application of database design principles, role-based authentication, and complex business logic while providing valuable experience in web application development and real-world problem solving.
