On-campus Event Management System
Database Design Document

1. **Introduction:**

   The following document explains the database system design of the on-campus event management system, which has functionalities like creating events, managing budgets, booking venues, managing inventory items, managing organizations, analyzing participation information, and marketing the events.

2. **Business and Problems Addressed:**

   The database is to solve and boost the following issues/challenges:

   - **Event creation, management, and scheduling:** One place to manage everything. It makes things easy for both the host as well as the management/admin
   - **Participant enrolment and follow-up:** Nudge the participants who have attended the previous events to enroll in new events
   - **Finance management, budgeting, and ticketing:** Plan and organize budgets and tickets for each event privately
   - **Marketing:** Notifying and reminding participants
   - **Data analysis:** Analysis of the participants' data for each event along with feedback for the organizers to track the performance of their events

3. **Entities Description:**

3.1 User

- **Attributes**: UserID (Primary Key), Password, FirstName, LastName, DOB, Gender, Email
- **Relationships:** Has User Type, SuperType of Admin, SuperType of Participant, SuperType of Host

3.2 User Type

- **Attributes**: UserID (Foreign Key -> User(UserID)), UserType
- **Relationships**: Owned by User

3.3 Admin

- **Attributes**: StartDate, Responsibility, Inherits all the attributes from User

- **Relationships:** Fulfills Requests, Manage Tickets, Manage Venues, Manage Items, Create Organization, Create Host, Subtype of User

3.4 Participant

- **Attributes**: MajorName, StartYear, EduType, T-Shirt Size, Inherits all the attributes from User
- **Relationships:** Gives Feedback, Registration to Event Seats, Subtype of User

3.5 Host

- **Attributes**: OrgID (Foreign Key -> Organization(OrgID)), HostStatus, Inherits all the attributes from User
- **Relationships:** Belongs to Organization, Raises Requests, Post Events, Is created by Admin, Subtype of User

3.6 Requests

- **Attributes**: RequestID (Primary Key), Description, Status, HostID(Foreign Key -> Host(HostID))
- **Relationships:** Raised by Host, Fulfilled by Admin

3.7 Organization

- **Attributes**: OrgID (Primary Key), OrgName, FoundingYear, Founder
- **Relationships:** Has Host, Created by Admin

3.8 Feedback

- **Attributes**: EventID (Primary Key, Foreign Key -> Event(EventID)), UserID (Primary Key, Foreign Key -> User(UserID)), Message, Rating
- **Relationships:** Given by Participant, Rated to Event

3.9 Budget

- **Attributes**: BudgetID(Primary Key).
- **Relationships:** Allocated to Event, Budget Sheet ot Items

3.10 Event

- **Attributes**: EventID (Primary Key), EventName, Description, Status, OrgID (Foreign Key -> Host(OrgID)), VenueID (Foreign Key -> Venue(VenueID)), ReservationDate, BookingDateTime, CancelDateTime, CheckinDateTime
- **Relationships:** Posted by Host, Receive Rating, Book Venue, Registration by Participants for Seats, Get Budget Allocated, Have Seats

3.11 Registration (Associative Entity)

- **Attributes**: TicketQuantity, TransactionTime, Remarks
- **Relationships:** Registered by Participant for an Event for Seats

3.12 Budget Sheet (Associative Entity)

- **Attributes**: Quantity
- **Relationships:** Budget has Budget Sheet of an Item

3.13 Venue

- **Attributes**: VenueID (Primary Key), VenueName, Location, Capacity
- **Relationships:** Booked by Event, Managed by Admin

3.14 Seats

- **Attributes**: EventID(Primary Key, Foreign Key -> Event(EventID)), TicketID(Primary Key, Foreign Key -> Tickets(TicketID)), Description, MaxQuantity, AutoApprove
- **Relationships:** Assigned by Event, Assigned by Registration, Assigned by Tickets

3.15 Item

- **Attributes**: ItemID (Primary Key), ItemName, ItemCost
- **Relationships:** Has Budget Sheet for Budget, Managed by Admin

3.16 Tickets

- **Attributes**: TicketID (Primary Key), Cost
- **Relationships:** Categorize Seats, Managed by Admin

4. Entity Relationships:

4.1 User:

- **User Type:** One-to-Many relationship (each user can have one or more user types)
- Is a supertype of **Admin, Participant, and Host** (a user can be an admin, participant, or host) with an **overlap** (Example: User can be a host and participant simultaneously)

4.2 User Type:

- **User:** Many-to-one relationship(Multiple user types can be linked to same User)

4.3 Admin:

- Is a subtype of **User**
- **Tickets:** One-to-many relationship (One admin can manage multiple tickets)
- **Item:** One-to-many relationship (One admin can manage multiple items)
- **Venue:** One-to-many relationship (One admin can manage multiple venues)
- **Organization:** One-to-many relationship (One admin can create multiple organizations)
- **Host:** One-to-many relationship (One admin can create multiple Hosts)
- **Requests:** One-to-many relationship (One admin can fulfill multiple requests)

4.4 Participant:

- Is a subtype of **User**
- **Event:** Many-to-many via Registration Associative relationship entity (One participant can register for multiple events for different times and tickets)
- **Feedback:** One-to-one relationship (A user can give single feedback associated to an event)

4.5 Host:

- Is a subtype of **User**
- **Organization:** Many-to-one relationship (Multiple hosts can belong to the same organization)
- **Requests:** One-to-many relationship (One host can raise multiple requests)
- **Admin:** Many-to-one relationship (Many hosts are created by a single admin)
- **Events:** One-to-many relationship (One host can post multiple events)

4.6 Requests:

- **Host:** Many-to-one relationship (Many requests are created by a single host)
- **Admin:** Many-to-one relationship (Many requests are fulfilled by a single admin)

4.7 Organization:

- **Host:** One-to-many relationship (One organization can have multiple hosts)
- **Admin:** Many-to-one relationship (Many organizations are created by a single admin)

4.8 Feedback:

- **Event:** Many-to-one relationship (Many feedbacks can be rated to the same event)
- **Participant:** One-to-one relationship (One feedback is given by only one participant)

4.9 Budget:

- **Event:** One-to-one relationship (One budget is associated to only one event)
- **Item:** Many-to-many via Budget Sheet Associative relationship entity (One budget has budget sheets for multiple items with different quantity)

4.10 Event:

- **Host:** Many-to-one relationship (Many events are posted by a single host)
- **Venue:** One-to-one relationship (One event at a particular time can be booked for a particular venue)
- **Seats:** One-to-many relationship (One event can have multiple seats)
- **Participant:** Many-to-many via Registration Associative relationship entity (One even is registered by multiple participants with at different time and quantity)
- **Feedback:** One-to-many relationship (One event can be rated with multiple feedbacks)
- **Budget:** One-to-one relationship (One event can have only one budget)

4.11 Venue:

- **Event:** One-to-one relationship (One venue can be booked by only one event for a particular BookingDateTime)
- **Admin:** Many-to-one relationship (Many venues are managed by a single admin)

4.12 Seats:

- **Events:** Many-to-one relationship (Many seats have to belong a single event)
- **Tickets:** Many-to-one relationship (Many seats can be categorized to one ticket)

4.13 Item:

- **Budget:** Many-to-many via Budget Sheet Associative relationship entity (One item is having a budget sheet to multiple budgets with different quantity)
- **Admin:** Many-to-one relationship (Many items are managed by a single admin)

4.14 Tickets:

- **Seats:** One-to-many relationship (One ticket can be categorized into multiple seats)
- **Admin:** Many-to-one relationship (Many tickets are managed by a single admin)

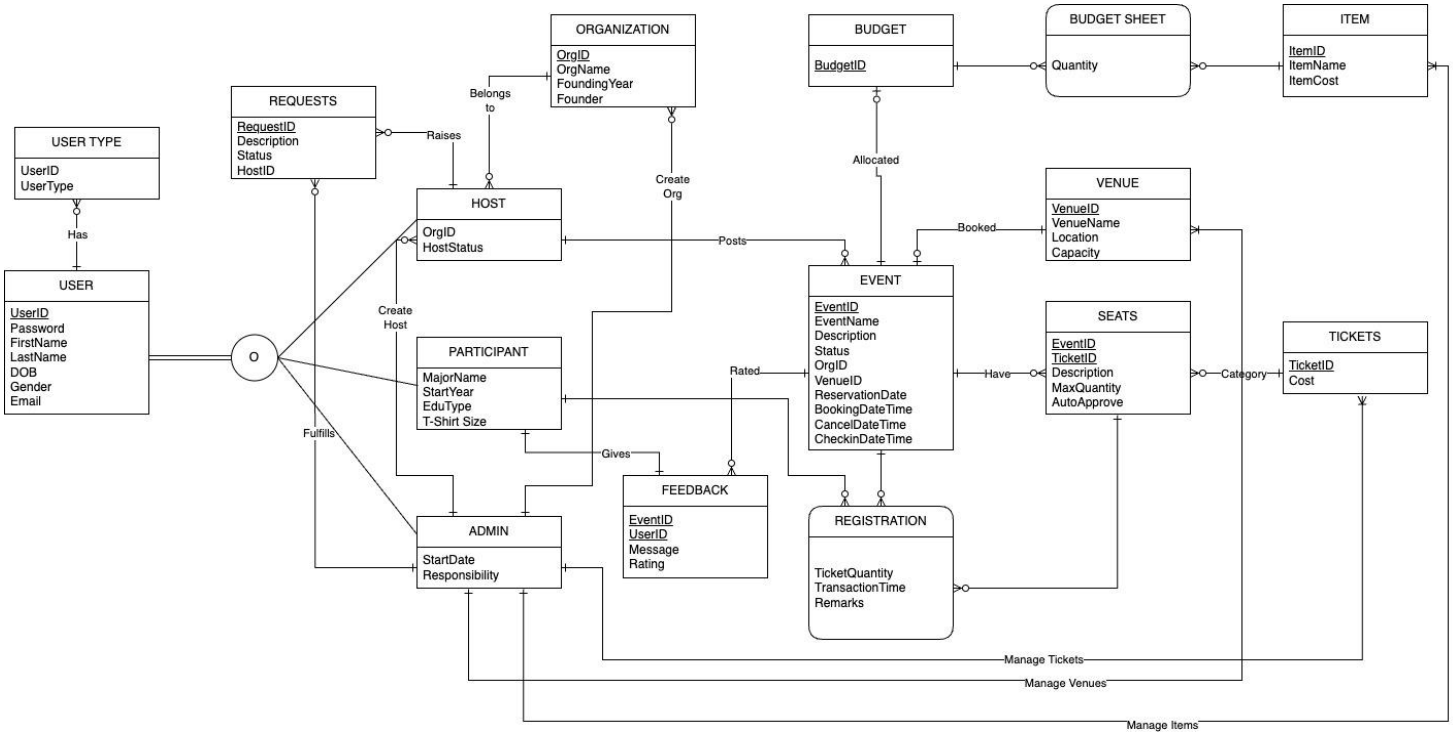
4.15 Registration:

- It is an Associative Entity between Participant, Event, and Seats

4.16 Budget Sheet:

- It is an Associative Entity between Budget and Items

5. ER Diagram:


ORGANIZATION
OrgID
OrgName
FoundingYear
Founder
BUDGET
BudgetID
BUDGET SHEET
Quantity
ITEM
ItemID
ItemName
ItemCost
REQUESTS
RequestID
Description
Status
HostID
Belongs to
Raises
USER TYPE
UserID
UserType
Has
HOST
OrgID
HostStatus
Create Org
Allocated
VENUE
VenueID
VenueName
Location
Capacity
Booked
Posts
USER
UserID
Password
FirstName
LastName
DOB
Gender
Email
O
Create Host
EVENT
EventID
EventName
Description
Status
OrgID
VenueID
ReservationDate
BookingDateTime
CancelDateTime
CheckinDateTime
SEATS
EventID
TicketID
Description
MaxQuantity
AutoApprove
Have
Category
TICKETS
TicketID
Cost
PARTICIPANT
MajorName
StartYear
EduType
T-Shirt Size
Rated
Fulfills
Gives
FEEDBACK
EventID
UserID
Message
Rating
ADMIN
StartDate
Responsibility
REGISTRATION
TicketQuantity
TransactionTime
Remarks
Manage Tickets
Manage Venues
Manage Items