Project 3: Logical Database Design

Database Planning and Requirement Analysis
Event Management System: OccasionOrganizer

By

Group ID: 15

Miss Chommakorn	Sontesadisai	6488189
Miss Nattanicha	Sinsawet	6488190
Miss Ravikarn	Jarungjitvittawas	6488210

To
Asst. Prof. Dr. Charnyote Pluempitiwiriyawej

A Report Submitted in Partial Fulfillment of
the Requirements for

ITCS413 Database Design

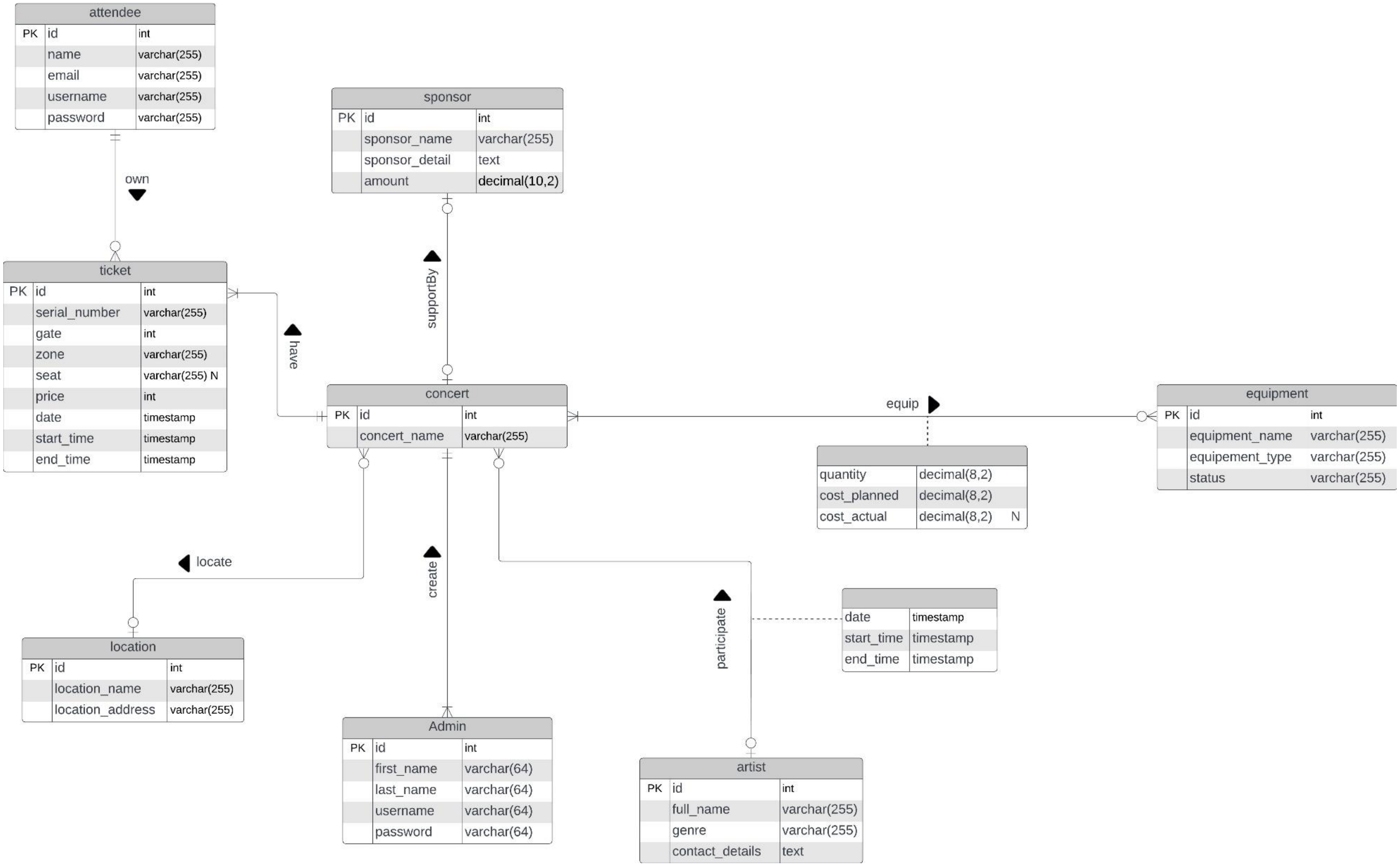
Faculty of Information and Communication Technology
Mahidol University

Last Updated: March 11, 2024

Table of Content

Project 3: Logical Database Design

ER diagram



attendee		
PK	id	int
	name	varchar(255)
	email	varchar(255)
	username	varchar(255)
	password	varchar(255)

sponsor		
PK	id	int
	sponsor_name	varchar(255)
	sponsor_detail	text
	amount	decimal(10,2)

ticket		
PK	id	int
	serial_number	varchar(255)
	gate	int
	zone	varchar(255)
	seat	varchar(255) N
	price	int
	date	timestamp
	start_time	timestamp
	end_time	timestamp

concert		
PK	id	int
	concert_name	varchar(255)

equipment		
PK	id	int
	equipment_name	varchar(255)
	equipement_type	varchar(255)
	status	varchar(255)

quantity	decimal(8,2)	
cost_planned	decimal(8,2)	
cost_actual	decimal(8,2)	N

location		
PK	id	int
	location_name	varchar(255)
	location_address	varchar(255)

Admin		
PK	id	int
	first_name	varchar(64)
	last_name	varchar(64)
	username	varchar(64)
	password	varchar(64)

date	timestamp
start_time	timestamp
end_time	timestamp

artist		
PK	id	int
	full_name	varchar(255)
	genre	varchar(255)
	contact_details	text

List of Changed ERD

1. Change the relationship between the attendee entity and the ticket entity from mandatory to optional. And add the relationship name named own.

2. Change the relationship between the concert entity and the ticket entity from one-to-one to one-to-many.

3. Change the relationship between the concert entity and the location entity. From mandatory to optional

4. Change the relationship between the sponsor and concert entity name from is_sponsor to supportBy.

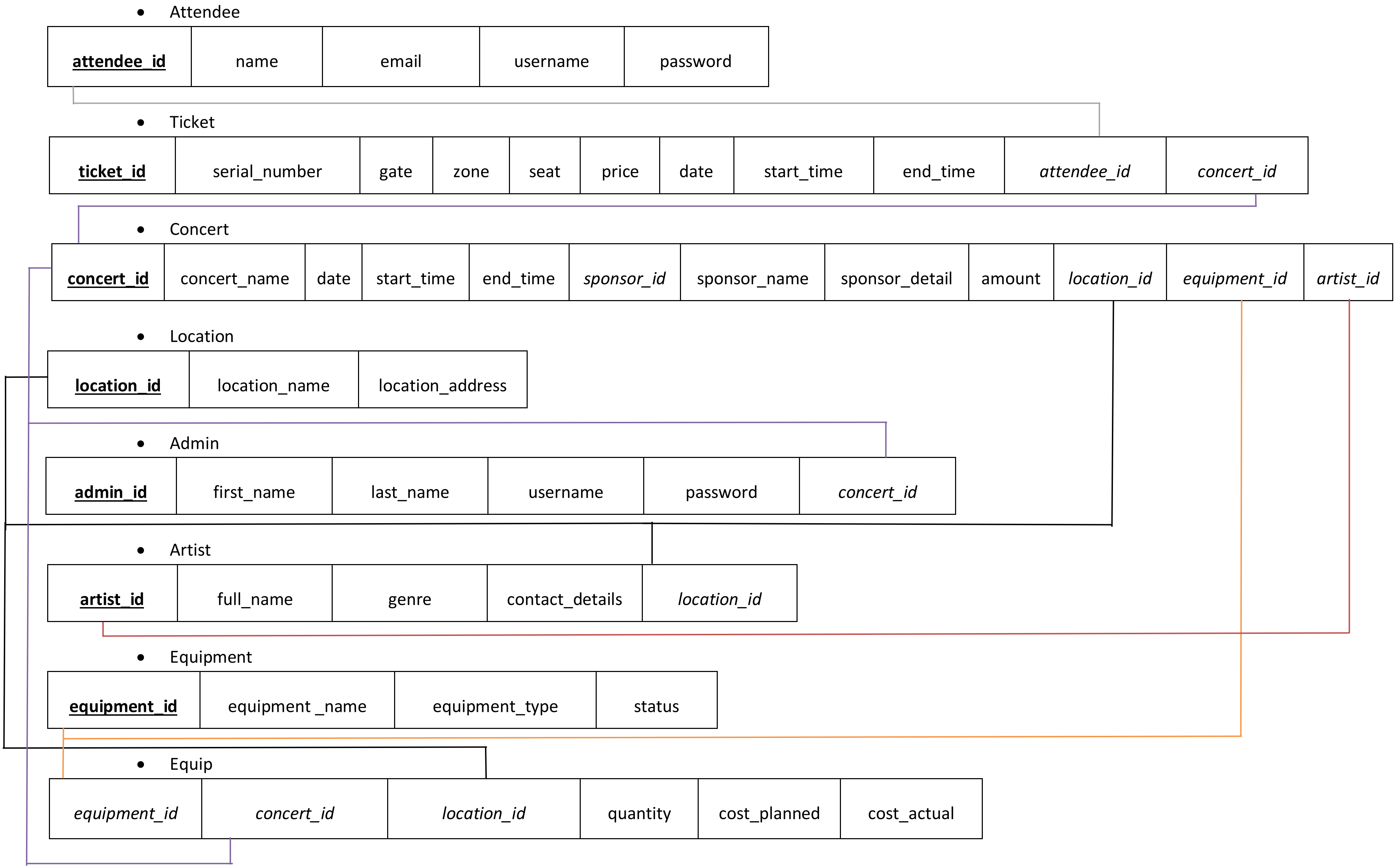
5. Change the relationship between the admin and the concert entity. Swap one-to-many to many-to-one

6. Change the relationship between the artist and the concert entity from mandatory to optional.

7. Add the mandatory to the concert entity and equipment entity.

Relation schemas and integrity constraints

- **Relation Schema**
  - **Attributes** – which are bold and underlined are the Primary Keys
  - *Attributes* – which are Italic are the Foreign Keys
  - ***Attributes*** – which are bold, italic and underlined are both Primary Keys and Foreign Keys



- Attendee

attendee_id	name	email	username	password

- Ticket

ticket_id	serial_number	gate	zone	seat	price	date	start_time	end_time	*attendee_id*	*concert_id*

- Concert

concert_id	concert_name	date	start_time	end_time	*sponsor_id*	sponsor_name	sponsor_detail	amount	*location_id*	*equipment_id*	*artist_id*

- Location

location_id	location_name	location_address

- Admin

admin_id	first_name	last_name	username	password	*concert_id*

- Artist

artist_id	full_name	genre	contact_details	*location_id*

- Equipment

equipment_id	equipment _name	equipment_type	status

- Equip

equipment_id	*concert_id*	*location_id*	quantity	cost_planned	cost_actual

- **Integrity constraints**

  Table: Attendee

  - Primary Key Constraint: attendee_id ensures that each attendee has a unique identifier.

  Table: Ticket

  - Primary Key Constraint: ticket_id uniquely identifies each ticket.
  - Foreign Key Constraint: attendee_id references the attendee_id in the Attendee table, ensuring tickets are tied to attendees.
  - Foreign Key Constraint: concert_id references the concert_id in the Concert table, connecting tickets to specific concerts.

  Table: Concert

  - Primary Key Constraint: concert_id serves as a unique identifier for each concert.
  - Foreign Key Constraint: sponsor_id references sponsor_id in the Sponsor table, associating sponsors with concerts.
  - Foreign Key Constraint: location_id references location_id in the Location table, tying concerts to specific venues.
  - Foreign Key Constraint: equipment_id references equipment_id in the Equipment table, linking concerts with the equipment used.
  - Foreign Key Constraint: artist_id references artist_id in the Artist table, connecting concerts to the performing artists.

  Table: Location

  - Primary Key Constraint: location_ID uniquely identifies each location.

  Table: Admin

  - Primary Key Constraint: admin_ID is the unique identifier for each admin.

- Foreign Key Constraint: concert_id references the concert_id in the Concert table, connecting admins to specific concerts.

Table: Artist

- Primary Key Constraint: artist_id is the unique identifier for each artist.
- Foreign Key Constraint: location_id references the location_id in the Location table, linking artists to specific locations.

Table: Equipment

- Primary Key Constraint: equipment_id uniquely identifies each piece of equipment.

Table: Equip

- Composite Primary Key Constraint: The combination of equipment_id and concert_id uniquely identifies each record in the Equip table.
- Foreign Key Constraint: equipment_id references equipment_id in the Equipment table, ensuring only valid equipment is allocated.
- Foreign Key Constraint: concert_id references concert_id in the Concert table, tying equipment to specific concerts.
- Foreign Key Constraint: location_id references location_id in the Location table, associating the equipment with a specific location.

- **Additional Considerations for Integrity Constraints:**

  - Not Null Constraint: It's essential to ensure that critical columns like name, email, username, and password in the Attendee table cannot be left empty. This guarantees that every attendee record is complete and usable.

  - Unique Constraint: This constraint is crucial for columns like email in the Attendee table to prevent duplicate entries. Every attendee should have a unique email address to ensure proper identification and communication.

  - Check Constraint: This constraint should be applied to columns that need to contain values within a certain range. For instance, the amount in the Sponsor table should always be a positive number to accurately reflect the financial contributions.

  - Data Type Constraint: This ensures that the data entered into each field matches the expected data type. For example, the email fields should only contain valid email addresses. This helps to maintain data accuracy and integrity.