

Movie Management System

Table of Content

1.	Company Introduction	1
1.1.	Introduction to the Dhungana Cinemas	1
1.2.	Current Business Activities and Operations	2
1.3.	Business Rules	3
1.4.	Creation of Entity and Attribute	4
1.5.	ER Diagram	5
1.6.	Use Case diagram	6
2.	Database Design	7
2.1.	Assumption	7
2.2.	Normalization	7
2.3.	UNF (Un-Normalized Form)	7
2.4.	1NF	8
2.5.	2NF	9
2.6.	3NF	10
2.7.	ER Diagram	11
3.	Data Dictionary	12
4.	Implementation	19
5.	Putting values/data to the Tables	23
6.	Ten Queries of our choice	30
6.1.	Queries to drop table:	37
7.	Critical Evaluation	38
8.	Future Enhancement	39

9. Summary	40
References	41

Table of Figures

Figure 1: ER Diagram.....	5
Figure 2: Use Case Diagram.....	6
Figure 3: ER diagram with relationship.....	11
Figure 4: Creating the database; thedhunganacinas.....	19
Figure 5: Creating table movie_show in the database.....	20
Figure 6: Creating the table membership_card in the database.....	20
Figure 7: Creating the table seats in the database	21
Figure 8: Creating table employee in the database.....	21
Figure 9: Creating table others in database	22
Figure 10: Creating table Customer in database.....	22
Figure 11: Creating table Booking_tickets in the database	23
Figure 12: Putting the values in table “customer”	24
Figure 13: Putting the data in table movie_show.....	25
Figure 14: Putting the data in table others.....	26
Figure 15: Putting the value/data in table employee	27
Figure 16: Putting the data in the table booking_ticket.....	28
Figure 17: Inserting data in the table seat	29
Figure 18: Data in table membership_card	30
Figure 19: Viewing the data from the table movie_shows	31
Figure 20: To find the third-highest salary from the employee table.....	32
Figure 21: Calculating Even record from table	33
Figure 22: Calculating Odd records from table	33
Figure 23: To get the most costly movie ticket	34
Figure 24: To get the cheapest movie ticket.....	34
Figure 25: length of the text in the movie_name	35

Figure 26: seat number and seat row column together	35
Figure 27: employees name in upper case	36
Figure 28: seat row in ascending order	36
Figure 29: movies name whose payment is done by esewa or khalti.....	37
Figure 30: Union all the hall number from table seats and movie_shows table	37

1. Company Introduction

1.1. Introduction to the Dhungana Cinemas

The Dhungana cinemas is the largest Multiplex of Nepal which was established on 2003 Jan 20. The Dhungana cinemas was established in the motive of quality cinema halls which provides the best services to its customers. The Dhungana cinemas is known throughout the country because of its 80+ cinema halls in the country with 77 districts with optimum customer satisfaction. The head office of the Dhungana cinemas is at Kathmandu Metropolitan City-01.

The cinema halls in Nepal are not much modernized as much as the Dhungana cinemas does. The Dhungana Cinemas was established around 19 years before as the technology was not much developed, coming through the long journey the technology has evolved so much and to run with the technology the dhungana cinemas is always successful. As we have mentioned, the dhungana cinemas is known for the optimum customer satisfaction, the extra offers announced by the company on the regular interval of time and the always running offers are the plus point of the company. The vision and motive of the company is to make the customer happy and give the best shrives. We, the dhungana cinemas focus on even if the movie is worst, the services of the cinemas halls makes the viewers/customer the best services.

1.2. Current Business Activities and Operations

- a. The cinema halls provide the best comfortable seats for all as it may vary from the pricing of the ticket and the special reservations if done.
- b. Free 150ml soft drinks on every Wednesday to our regular customer i.e. card holder/subscription. Same offer may be made on other days according to the company's decision.
- c. No ticket price to the customer below age of 0-5, as no price ticket if they don't occupy the seat.
- d. Regular offers on festivals and special discounts as per the company's decision.
- e. The timing of the cinema hall is from morning 6 AM to mid-night 12 AM. This is only for the cinemas halls in the major cities of Nepal. Other cinema halls have the timing of morning 7 AM to 8 PM.
- f. The dhungana cinemas host every type of movie as per the directive of Film Development board, Nepal.
- g. Pre-booking is available and special reservations can also be done.
- h. Pricing of tickets may differ according to the services and seats.
- i. Free Wi-Fi services are available in all the movie theaters of The Dhungana Cinemas.

1.3. Business Rules

- a. Customer information must be kept, contact number is must.
- b. The employee of the company must follow the company's dress code.
- c. The employee must talk politely and obey the customers and help them in anyway.
- d. The viewers/customers are not allowed to take the food inside the hall. If any medical verification is needed.
- e. Recording the cinema with the time interval of more than 3 minutes, if found the customer/viewer has to look after the consequence as per the company's rule.
- f. Mobile phones must be silenced.
- g. Legal action will be taken if any harm is caused to the cinema hall by the customers/viewers.
- h. Company will not tolerate any kind of false information to grab the offers by the customer.
- i. Every customer are equal for us.
- j. Customers with the gold card (Subscription) of the company and shareholders of the company will be given extra benefits time and accordingly.
- k. Customers with card bellow gold will only get the discounts on the purchase of ticket.
- l. Card holder customer will get the points according to the purchase of ticket as they can level up their positions in card.
- m. Loss of the customer's belonging, the company will not be responsible.
- n. The system should be for the employee and customer/user/viewers.

1.4. Creation of Entity and Attribute

- Entity

An entity is a real-world object that represents data in the Relational Database Management System (RDBMS).

- Attribute

Attributes are properties that describe an entity. (Pediaa, 2017)

Entity	Attributes
1. Movie Shows	Movie_name(PK), Genre, Hall number, Timing, Movie Type
2. Customer	Customer ID(PK), Customer name, contact number, sex, Email address
3. Employee	Employee id(PK), employee name, salary, address, contact number
4. Booking Ticket	Ticket number(PK), movie name, Timing, Venue, payment type, pricing
5. Card Details	Card number(PK), Card Holder's Name, Type of card, contact number, Email address
6. Seats	Seat_number (PK), seat row, hall number
7. others	Provision_id(PK), Special provision(PK), Reservation, emergency, media partner

1.5. ER Diagram

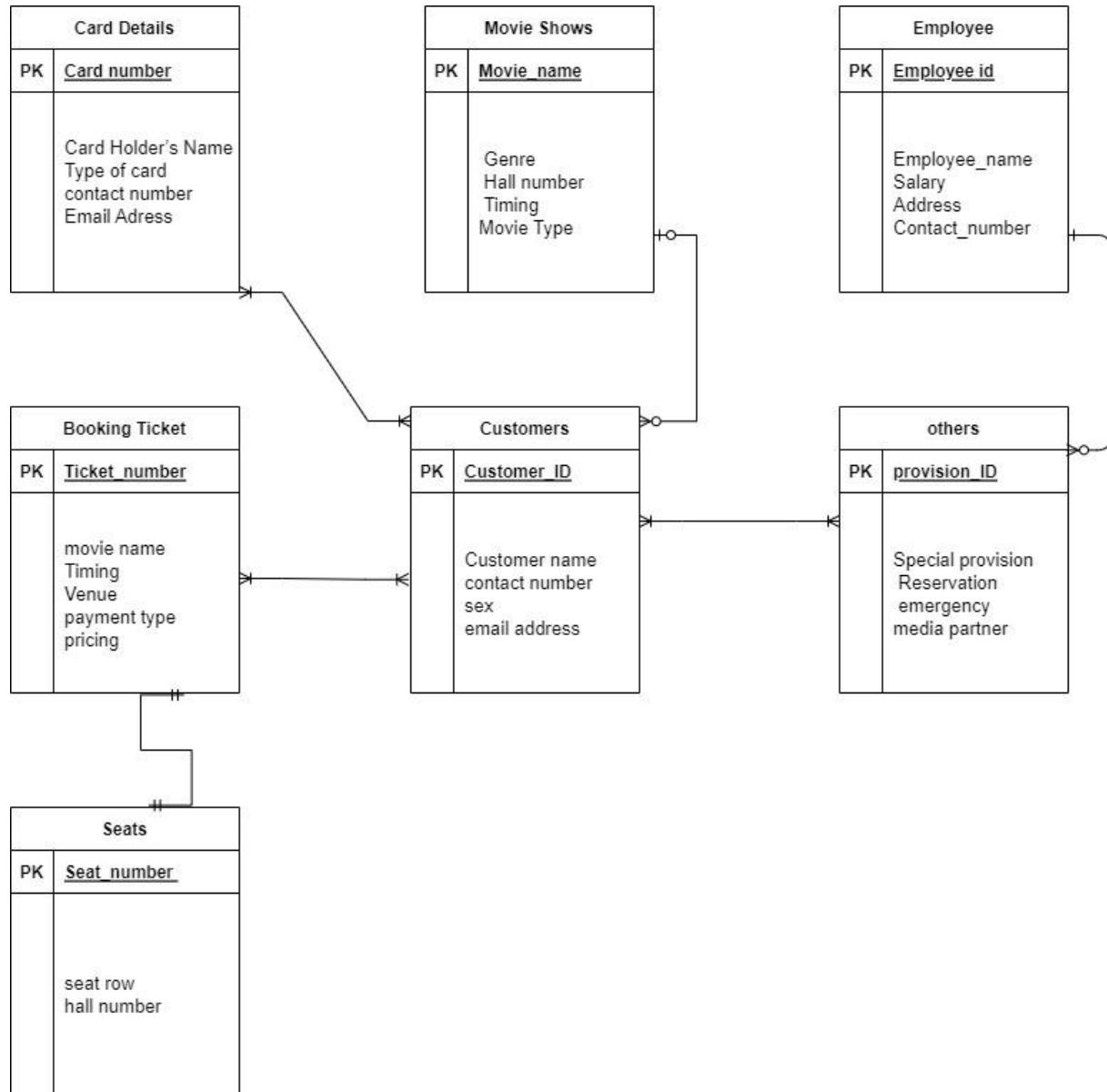


Figure 1: ER Diagram

1.6. Use Case diagram

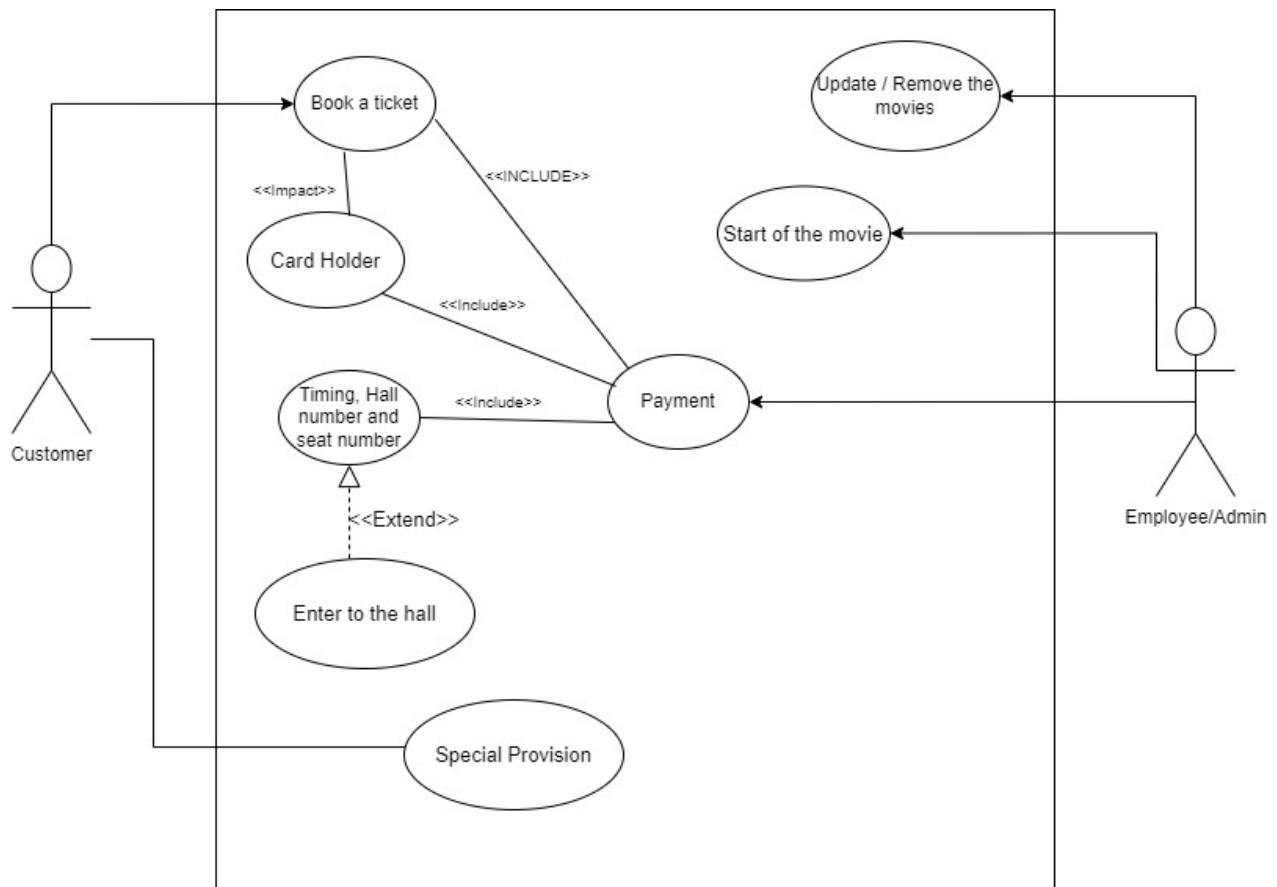


Figure 2: Use Case Diagram

2. Database Design

2.1. Assumption

The assumption are done to justify the ER diagram, they are:

- Every movie has its own value i.e. pricing, hall number, venue etc.
- A person can be both customer and employee.
- The entity other is also made to make the customer easy whether they want to contact us, want the special reservation and any other help will come under the entity “other”.
- Card is issued by the dhungana cinemas and valid only in the cinema halls, card is used for discounts and other facilities.
- Customers with only one ticket can only use one seat, one and only one.

2.2. Normalization

Normalization is a database design technique that reduces data redundancy and eliminates undesirable characteristics like Insertion, Update and Deletion Anomalies. Normalization rules divides larger tables into smaller tables and links them using relationships. (Peterson, 2022).

2.3. UNF (Un-Normalized Form)

Movie_show(Movie_name(PK), genre, hall_number, timing, movie_type)
Employee(Employee_ID(PK), Employee_name, Salary, Address,
Contact_number)
Customers(Customer_ID(PK), Customer_name, contact_number, sex,
email_address)
Seats(seat_number(PK), seat_row, hall_number)
Others(provision_ID(PK), special_provision, emergency, media_partner)

Card_details(Crad_number(PK), Card_holders_name, Typeofcard,
Contact_number, Email_Address)
Booking_Ticket(Ticket_number(PK), movie_name, timing, venue,
payment_type, pricing)

2.4. 1NF

Basic Rules for first Normal Form:

- a. The data field should be atomic.
- b. It eliminates duplicate rows and columns from the same table.
- c. It minimizes the data redundancy in the database table.
(pudasaini, et al., n.d.)

Movie_show(Movie_name(PK), genre, hall_number, timing, movie_type)
Employee(Employee_ID(PK), Employee_name, Salary, Address, Contact_number)
Customers(Customer_ID(PK), Customer_name, contact_number, sex, email_address)
Seats(seat_number(PK), seat_row, hall_number)
Others(provision_ID(PK), special_provision, emergency, media_partner)
Card_details(Crad_number(PK), Card_holders_name, Typeofcard, Contact_number,
Email_Address)
Booking_Ticket(Ticket_number(PK), movie_name, timing, venue, payment_type, pricing)

2.5. 2NF

Basic Rules for second Normal form:

- a. It should be in the first normal form.
- b. It identifies data dependencies.
- c. Non key attribute are functionally depends on key attributer.

(pudasaini, et al., n.d.)

Movie_show(Movie_name(PK), genre, hall_number, timing, movie_type)

Employee(Employee_ID(PK), Employee_name, Salary, Address, Contact_number)

Customers(Customer_ID(PK), Customer_name, contact_number, sex, email_address)

Seats(seat_number(PK), seat_row, hall_number)

Others(provision_ID(PK), special_provision, emergency, media_partner)

Card_details(Crad_number(PK), Card_holders_name, Typeofcard, Contact_number, Email_Address)

Booking_Ticket(Ticket_number(PK), movie_name, timing, venue, payment_type, pricing)

2.6. 3NF

Basic Rules for Third Normal form:

- a. It should be the second normal form.
- b. It removes transitive dependencies in a table.

Movie_show(Movie_name(PK), genre, hall_number, timing, movie_type)

Employee(Employee_ID(PK), Employee_name, Salary, Address, Contact_number)

Customers(Customer_ID(PK), Customer_name, contact_number, sex, email_address)

Seats(seat_number(PK), seat_row, hall_number)

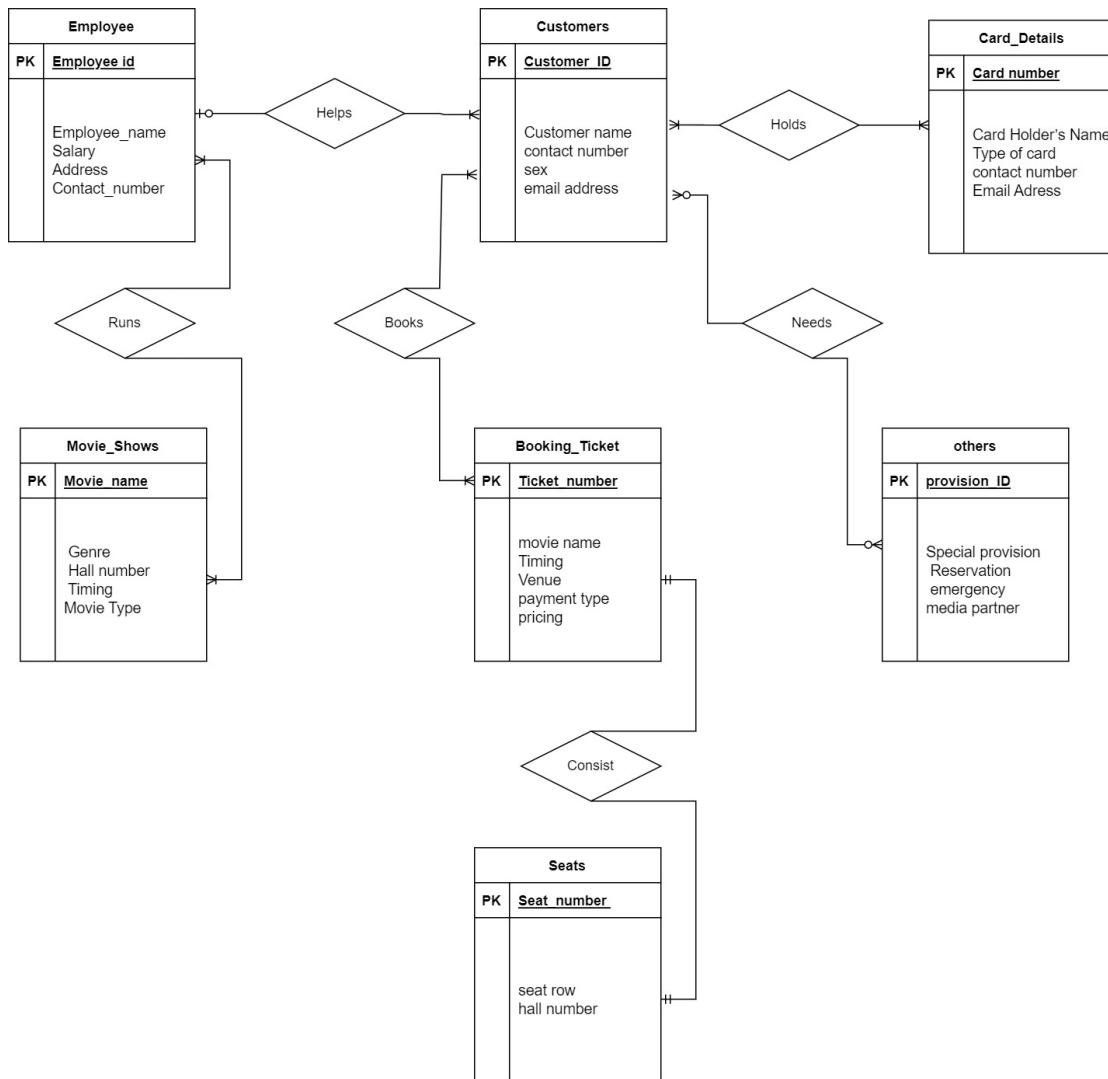
Others(provision_ID(PK), special_provision, emergency, media_partner)

Card_details(Crad_number(PK), Card_holders_name, Typeofcard, Contact_number, Email_Address)

Booking_Ticket(Ticket_number(PK), movie_name, timing, venue, payment_type, pricing)

2.7. ER Diagram

ER model stands for an Entity-Relationship Diagram. It is a high-level data model. This model is used to define the data elements and relationship for a specified system. It develops a conceptual design for the database. It also develops a very simple and easy to design view of data. (javatpoint, 2021)



ER Diagram By: Anirudha Dhungana(2226677)

Figure 3: ER diagram with relationship

3. Data Dictionary

The data dictionary is very important as it contains information such as what is in the database, who is allowed to access it, where is the database physically stored etc. The users of the database normally don't interact with the data dictionary, it is only handled by the database administrators.

The data dictionary in general contains information about the following:

- Names of all the database tables and their schemas.
- Details about all the tables in the database, such as their owners, their security constraints, when they were created etc.
- Physical information about the tables such as where they are stored and how.
- Table constraints such as primary key attributes, foreign key information etc.
- Information about the database views that are visible. (tutorialspoint, 2022)

Movie Shows

Attributes	Data type	Field Length	Constraint	Description	Example
Movie Name	VARCHAR	20	PK	The movie name cannot be change and all the data can be fetch using this PK.	DDLJ
Genre	VARCHAR	20	Not Null	Movie based on which style.	Family Drama
Hall Number	INT	10	FK	The particular hall no. in which the movie will be forecasted.	5
Timing	FLOAT	20	Not Null	Timing of movie	10 AM
Movie Type	VARCHAR	20	Not Null	Which type of movie is this	Bollywood

Seats

Attributes	Data type	Field Length	Constraint	Description	Example
Seat Number	int	10	PK	Seat number determines the seat that is occupied by which customer/viewers.	10
Seat row	char	10	Not Null	It determines the row.	X
Hall Number	int	10	FK	The particular hall no. in which the movie will be forecasted.	8

Membership Card/Card Details

Attributes	Data type	Field Length	Constraint	Description	Example
Card_number	int	20	PK	It is the registered number to gain the offers and other facilities in the cinema hall.	02147566
Card Holder's Name	char	30	Not Null	Full name of card holder.	Ribesh Nepal
Type of card	VARCHAR	30	Not Null	It helps to specify in which level the card is.	Platinum Card
Email Address	VARCHAR	50	Not Null	It must contain .com	Rbs.n@gmail.com

Employee

Attributes	Data type	Field Length	Constraint	Description	Example
Employee_ID	VARCHAR	20	PK	It helps to find out the details of employee.	22475998
Employee name	Char	30	Not Null	Name cannot be null	Shira Khatiwada
Salary	Float	20	Not Null	Salary can be in decimal value.	30000
Address	VARCHAR	50	Not Null	It helps to keep the records of employee.	ithari
Contact Number	int	15	FK	In case of emergency company might need to contact to its employee.	9852144735

Customer

Attributes	Data type	Field Length	Constraint	Description	Example
Customer ID	VARCHAR	20	PK	The unique customer ID helps to keep the records in it.	4526
Customer name	char	20	Not Null	Name cannot be null.	Manish Adhikari
sex	VARCHAR	10	Not Null	It helps to keep the record, which group of people come mostly to watch the movies and which.	Male
Email Address	VARCHAR	50	Not Null	The ticket related details can be mailed to the customers and let them about the offers.	Adhikari.m@gmail.com

Booking Tickets

Attributes	Data type	Field Length	Constraint	Description	Example
Ticket Number	int	20	PK	Ticket number helps to trace the customer in the case of emergency.	225479
Movie Name	VARCHAR	20	FK	It helps to specify the hall number later.	KGF
Venue	VARCHAR	20	Not Null	The dhungana Cinemas has 80+ branches, venue makes easier in which state/city's ticket is purchased.	Lalitpur
Payment Type	VARCHAR	20	Not Null	It specifies which method of payment the ticket is purchased.	Ime pay
Price	int	150	Not Null	Which type of movie is this	500

Others

Attributes	Data type	Field Length	Constraint	Description	Example
Provision_ID	VARCHAR	100	PK	Any provision or special kind of help needed it comes under it.	2254789
Reservation	VARCHAR	100	Not Null	Any special function or any other arrangements.	Conference
Emergency	int	20	Not Null	This helps to contact us in emergency.	012456879
Media Partner	VARCHAR	100	Not Null	Our media partner must be featured and be benefited.	Kantipur TV

4. Implementation

- Creating the Database: The Dhungana Cinemas

Query: CREATE DATABASE thedhunganacinemas;

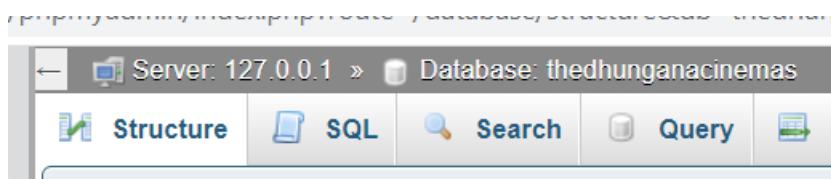


Figure 4: Creating the database; thedhunganacinemas

- Creating the tables in the database

- a. Creating table movie_shows in the database

Query: CREATE TABLE Movie_Shows(
 Movie_name VARCHAR(20) Not NULL,
 GENRE VARCHAR(20),
 HALL_NUMBER INT,
 TIMING VARCHAR,
 MOVIE_TYPE VARCHAR(20)
);

The screenshot shows the MySQL Workbench interface. A green status bar at the top indicates: "MySQL returned an empty result set (i.e. zero rows). (Query took 0.0007 sec)". Below this, a query window displays the SQL command: "SELECT * FROM `movie_shows`". Underneath the query window is a toolbar with options: Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]. Below the toolbar is a row of table headers: Movie_name, GENRE, HALL_NUMBER, TIMING, MOVIE_TYPE. At the bottom of the interface is a button labeled "Query results operations".

Figure 5: Creating table movie_show in the database

b. Creating the table Membership_card in the database

Query: CREATE TABLE Membership_card(
Card_number int(20),
Card_holders_name VARCHAR(30),
Typeofcard VARCHAR(30),
Email_Address VARCHAR(50)
);

The screenshot shows the MySQL Workbench interface. A green status bar at the top indicates: "MySQL returned an empty result set (i.e. zero rows). (Query took 0.0005 sec)". Below this, a query window displays the SQL command: "SELECT * FROM `membership_card`". Underneath the query window is a toolbar with options: Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]. Below the toolbar is a row of table headers: Card_number, Card_holders_name, Typeofcard, Email_Address. At the bottom of the interface is a button labeled "Query results operations".

Figure 6: Creating the table membership_card in the database

c. Creating the table seats in the database

Query: CREATE TABLE Seats(
seat_number int(10),

```
        Seat_row char(10),  
        Hall_number int (10)  
    );
```

The screenshot shows a MySQL database interface. At the top, there are several buttons: 'Profiling' (disabled), 'Edit inline', 'Edit', and 'Explain SQL'. Below these buttons, the table structure for 'seats' is displayed. The columns are listed as 'seat_number', 'Seat_row', and 'Hall_number'. The 'Seat_row' column is highlighted with a blue background.

seat_number	Seat_row	Hall_number
-------------	----------	-------------

Figure 7: Creating the table seats in the database

d. Creating table employee in the database

```
Query: CREATE TABLE employee(  
        Employee_ID int(20),  
        Employee_name VARCHAR(30),  
        Salary float(20),  
        Address VARCHAR(50),  
        Contact_number int(15)  
    );
```

The screenshot shows a MySQL database interface. At the top, there are several buttons: 'Profiling' (disabled), 'Edit inline', 'Edit', 'Explain SQL', 'Create PHP code', and 'Refresh'. Below these buttons, the table structure for 'employee' is displayed. The columns are listed as 'Employee_ID', 'Employee_name', 'Salary', 'Address', and 'Contact_number'. The 'Employee_ID' column is highlighted with a blue background.

Employee_ID	Employee_name	Salary	Address	Contact_number
-------------	---------------	--------	---------	----------------

Figure 8: Creating table employee in the database

e. Creating table others in database

```
Query: CREATE TABLE others(  
        provision_ID VARCHAR(100),  
        Reservation VARCHAR(100),  
        Emmergency int(20),
```

```
        Address VARCHAR(50),  
        Media_partner Varchar(15)  
    );
```

The screenshot shows the MySQL Workbench interface. At the top, there is a toolbar with various buttons: Profiling, Edit inline, Edit, Explain SQL, Create PHP code, and Refresh. Below the toolbar, a tab bar is visible with tabs labeled provision_ID, Reservation, Emmergency, Address, and Media_partner. A large text area below the tabs contains the SQL code for creating the 'others' table. At the bottom of the interface, there is a button labeled Query results operations.

Figure 9: Creating table others in database

f. Creating table Customer in database

```
Query: CREATE TABLE customer(  
        Customer_ID int(20) NOT NULL,  
        Customer_name VARCHAR(30),  
        Sex VARCHAR(10),  
        Address VARCHAR(50),  
    );
```

The screenshot shows the MySQL Workbench interface. At the top, there is a toolbar with various buttons. Below the toolbar, a tab bar is visible with tabs labeled Customer_ID, Customer_name, Sex, Address, and Email_Address. A large text area below the tabs contains the SQL code for creating the 'Customer' table. At the bottom of the interface, there is a button labeled Query results operations.

Figure 10: Creating table Customer in database

g. Creating table Booking_tickets in the database

```
Query: CREATE TABLE Booking_tickets(  
        Ticket_Number int(20) NOT NULL,  
        Movie_name VARCHAR(20),  
        Venue VARCHAR(20),  
        Payment_type VARCHAR(20),  
        Price int(150)  
    );
```

The screenshot shows a MySQL Workbench interface. At the top, a green message bar says "MySQL returned an empty result set (i.e. zero rows). (Query took 0.000 sec)". Below it, a SQL query window contains the command: "SELECT * FROM `booking_tickets`". To the right of the query window are several buttons: "Profiling", "Edit inline", "Edit", "Explain SQL", and "Create PHP code". Below the query window is a table header row with columns: "Ticket_Number", "Movie_name", "Venue", "Payment_type", and "Price". Underneath the table header is a section titled "Query results operations".

Figure 11: Creating table Booking_tickets in the database

5. Putting values/data to the Tables

- Putting the data in table “customer”

Query:

```
INSERT INTO Customer (Customer_ID, Customer_Name, Sex, Address)VALUES  
('284', 'Ribesh Nepal', 'Male', 'Kapan'),  
('546', 'Bishal Timalsina', 'Male', 'Kapan'),  
('756', 'Manish Adhikari', 'Male', 'Bhaktapur'),  
('210', 'Arush Khatiwada', 'Male', 'ithari'),  
('987', 'Arushi Khatiwada', 'Female', 'inaruwa'),  
('236', 'Gita Dhungana', 'female', 'kathmandu'),  
('889', 'Saru Humagain', 'female', 'germany');
```

Customer_ID	Customer_name	Sex	Address
284	Ribesh Nepal	Male	Kapan
546	Bishal Timalsina	Male	Kapan
756	Manish Adhikari	Male	Bhaktapur
210	Arush Khatiwada	Male	ithari
987	Arushi Khatiwada	Female	inaruwa
236	Gita Dhungana	female	kathmandu
889	Saru Humagain	female	germany

Figure 12: Putting the values in table “customer”

b. Putting the data in table movie_show

Query:

```
INSERT INTO movie_shows (Movie_name, Genre, Hall_Number, Timing,
Movie_type)VALUES
('KGF', 'Action', '4', '10 AM','Tollywood'),
('DDLJ', 'Romantic', '7', '8 AM','Bollywood'),
('Pushpa: The rise', 'Action', '4', '2 PM','Tollywood'),
('Xaka Panja', 'Drama', '1', '11 AM','Kollywood'),
('RRR', 'Action', '6', '10 PM','Tollywood'),
('Tadap', 'Action-Romantic', '2', '12 PM','Bollywood'),
('Kabhi Khusi Kabi Gam', 'Family drama', '7', '11 PM','Bollywood');
```

Movie_name	GENRE	HALL_NUMBER	TIMING	MOVIE_TYPE
KGF	Action	4	10 AM	Tollywood
DDLJ	Romantic	7	8 AM	Bollywood
Pushpa: The rise	Action	4	2 PM	Tollywood
Xaka Panja	Drama	1	11 AM	Kollywood
RRR	Action	6	10 PM	Tollywood
Tadap	Action-Romantic	2	12 PM	Bollywood
Kabhi Khusi Kabi Gam	Family drama	7	11 PM	Bollywood

Figure 13: Putting the data in table movie_show

c. Putting the data in table others

Query:

```
INSERT INTO others (provision_ID, Reservation, Emmergency, Address,
Media_partner)VALUES
('555', 'Sick', '14424890', 'ithari','Himalayan TV'),
('951', 'Engagement', '14457190', 'Bhaktapur','RONB'),
('5145', 'Medical Illness', '2147483647', 'Mustang','Trending Nepal'),
('489', 'Physically ill', '2147483647', 'Simara','ABC TV'),
('4526', 'Movie Promotion', '2147483647', 'Kathmandu','Secret_life2306'),
('5710', 'Movie Promotion', '2147483647', 'Dhangadi','Kantipur TV'),
('0297', 'Conference', '17532149', 'Kavre','Nice Television HD');
```

provision_ID	Reservation	Emmergency	Address	Media_partner
555	Sick	14424890	ithari	Himalayan TV
951	Engagement	14457190	Bhaktapur	RONB
5145	Medical Illness	2147483647	Mustang	Trending Nepal
489	Physically ill	2147483647	Simara	ABC TV
4526	Movie Promotion	2147483647	Kathmandu	Secret_life2306
5710	Movie Promotion	2147483647	Dhangadi	Kantipur TV
0297	Conference	17532149	Kavre	Nice Television

Figure 14: Putting the data in table others

d. Putting the value/data in table employee

Query:

```
INSERT INTO employee (Employee_ID, Employee_name, salary, Address,
Contact_number)VALUES
('0904', 'Shira Khatiwada', '30000', 'ithari', '2147483647'),
('0606', 'Kritika Pokhrel', '25000', 'Gaushala', '2147483647'),
('0547', 'Manila Adhikari', '20000', 'Bhaktapur', '2147483647'),
('4819', 'Rahul Shah', '25000', 'Biratnagar', '2147483647'),
('2647', 'Ujuwal Bhatrai', '20000', 'Budhanilkantha', '2147483647'),
('1006', 'Aarush Dhungana', '50000', 'Kathmandu', '2147483647'),
('0610', 'Aarushi Dhungana', '60000', 'Lalitpur', '2147483647');
```

Employee_ID	Employee_name	Salary	Address	Contact_number
904	Shira Khatiwada	30000	ithari	2147483647
606	Kritika Pokhrel	25000	Gaushala	2147483647
547	Manila Adhikari	20000	Bhaktapur	2147483647
4819	Rahul Shah	25000	Biratnagar	2147483647
2647	Ujuwal Bhattai	20000	Budhanilkantha	2147483647
1006	Aarush Dhungana	50000	Kathmandu	2147483647
610	Aarushi Dhungana	60000	Lalitpur	2147483647

Figure 15: Putting the value/data in table employee

e. Putting the data in the table booking_ticket

Query:

```
INSERT INTO booking_tickets (Ticket_Number, Movie_name, venue,
Payment_type, price)VALUES
('0904', 'KGF', 'ithari', 'Fonepay QR', '500'),
('0606', 'DDLJ', 'Gaushala', 'Esewa', '475'),
('0547', 'Pushpa: The rise', 'Bhaktapur', 'khalti', '300'),
('4819', 'Kabhi Khusi Kabi Gam', 'Biratnagar', 'ime pay', '350'),
('2647', 'Xaka Panja', 'Budhanilkantha', 'Namaste pay', '350'),
('1006', 'RRR', 'Kathmandu', 'Connectips', '500'),
('0214', 'Tadap', 'Lalitpur', 'pravupay', '450');
```

Ticket_Number	Movie_name	Venue	Payment_type	Price
904	KGF	ithari	Fonepay QR	500
606	DDLJ	Gaushala	Esewa	475
547	Pushpa: The rise	Bhaktapur	khalti	300
4819	Kabhi Khusi Kabi Gam	Biratnagar	ime pay	350
2647	Xaka Panja	Budhanilkantha	Namaste pay	350
1006	RRR	Kathmandu	Connectips	500
214	Tadap	Lalitpur	pravupay	450

Figure 16: Putting the data in the table booking_ticket

f. Inserting data in the table seat

Query:

```
INSERT INTO seats (seat_number, seat_row, Hall_number)VALUES
('0904', 'A', '5'),
('1056', 'X', '5'),
('879', 'D', '5'),
('441', 'Q', '5'),
('126', 'Z', '5'),
('0246', 'M', '5'),
('0477', 'L', '5');
```

seat_number	Seat_row	Hall_number
904	A	5
1056	X	5
879	D	5
441	Q	5
126	Z	5
246	M	5
477	L	5

Figure 17: Inserting data in the table seat

g. Inserting the data in table membership_card

Query:

```
INSERT INTO membership_card (Card_number, Card_holders_name,
Typeofcard,Email_Address)VALUES
('5554568', 'Ghanshyam Niraula', 'Basic Card', 'g.shyam@gmail.com'),
('5483214', 'Dasrath Chaulagain', 'Gold Card', 'd.ch@gmail.com'),
('9980135', 'Ram Timalsina', 'Basic Card', 'ram.sina1@gmail.com'),
('2579614', 'Jiwan Nepali', 'Silver Card', 'nepali11@gmail.com'),
('0322475', 'Pratap Singh', 'Platinum Crad', 'pratap.singh69@gmail.com'),
```

('0214766', 'Alex Modi', 'Silver Card', 'modi.al98@gmail.com'),
 ('0027449', 'Gaynendra Shah', 'BGold Card', 'hhyz.098@gmail.com');

Card_number	Card_holders_name	Typeofcard	Email_Address
5554568	Ghanshyam Niraula	Basic Card	g.shyam@gmail.com
5483214	Dasrath Chaulagain	Gold Card	d.ch@gmail.com
9980135	Ram Timalsina	Basic Card	ram.sina1@gmail.com
2579614	Jiwan Nepali	Silver Card	nepali11@gmail.com
322475	Pratap Singh	Platinum Crad	pratap.singh69@gmail.com
214766	Alex Modi	Silver Card	modi.al98@gmail.com
27449	Gaynendra Shah	BGold Card	hhyz.098@gmail.com

Figure 18: Data in table membership_card

6. Ten Queries of our choice

- a. View all the records of the table

Syntax: Select *from movie_shows

Showing rows 0 - 6 (7 total, Query took 0.0004 seconds.)

```
SELECT *FROM movie_shows;
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table

[Extra options](#)

Movie_name	GENRE	HALL_NUMBER	TIMING	MOVIE_TYPE
KGF	Action	4	10 AM	Tollywood
DDLJ	Romantic	7	8 AM	Bollywood
Pushpa: The rise	Action	4	2 PM	Tollywood
Xaka Panja	Drama	1	11 AM	Kollywood
RRR	Action	6	10 PM	Tollywood
Tadap	Action-Romantic	2	12 PM	Bollywood
Kabhi Khusi Kabi Gam	Family drama	7	11 PM	Bollywood

Figure 19: Viewing the data from the table movie_shows

- b. To find the third-highest salary from the employee table

Query:

```
SELECT Salary  
FROM employee
```

Employee_ID	Employee_name	Salary	Address	Contact_number
904	Shira Khatiwada	30000	ithari	2147483647
606	Kritika Pokhrel	25000	Gaushala	2147483647
547	Manila Adhikari	20000	Bhaktapur	2147483647
4819	Rahul Shah	25000	Biratnagar	2147483647
2647	Ujuwal Bhattarai	20000	Budhanilkantha	2147483647
1006	Aarush Dhungana	50000	Kathmandu	2147483647
610	Aarushi Dhungana	60000	Lalitpur	2147483647

ORDER BY Salary DESC

LIMIT 2, 1;

The screenshot shows a MySQL query results interface. At the top, there is a code editor containing the SQL query:

```
SELECT Salary FROM employee ORDER BY Salary DESC LIMIT 2,1;
```

Below the code editor is a toolbar with the following buttons: Profiling, Edit inline, Edit, Explain SQL, Create PHP code, and Refresh. There is also a checkbox labeled "Extra options".

The main area displays the results of the query. A table header "Salary" is shown in a grey box. Below it, the value "30000" is displayed in a white box.

Figure 20: To find the third-highest salary from the employee table

- c. To calculate even and odd records from the table

For Even, Query: `SELECT *FROM booking_tickets
WHERE MOD(Ticket_Number,2)=0;`

Showing rows 0 - 3 (4 total, Query took 0.0008 seconds.)

```
SELECT *FROM booking_tickets WHERE MOD(Ticket_Number,2)=0;
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table

Extra options

Ticket_Number	Movie_name	Venue	Payment_type	Price
904	KGF	ithari	Fonepay QR	500
606	DDLJ	Gaushala	Esewa	475
1006	RRR	Kathmandu	Connectips	500
214	Tadap	Lalitpur	pravupay	450

Figure 21: Calculating Even record from table

For Odd, Query:

```
SELECT *FROM booking_tickets
WHERE MOD(Ticket_Number,2)=1;
```

Showing rows 0 - 2 (3 total, Query took 0.0008 seconds.)

```
SELECT *FROM booking_tickets WHERE MOD(Ticket_Number,2)=1;
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table

Extra options

Ticket_Number	Movie_name	Venue	Payment_type	Price
547	Pushpa: The rise	Bhaktapur	khalti	300
4819	Kabhi Khusi Kabi Gam	Biratnagar	ime pay	350
2647	Xaka Panja	Budhanilkantha	Namaste pay	350

Figure 22: Calculating Odd records from table

- d. To get the costly and cheapest movie tickets among all the price

For costly, Query:

```
SELECT *from booking_tickets WHERE price=(SELECT max(price) FROM booking_tickets);
```

Ticket_Number	Movie_name	Venue	Payment_type	Price
904	KGF	ithari	Fonepay QR	500
1006	RRR	Kathmandu	Connectips	500

Figure 23: To get the most costly movie ticket

For Cheapest, Query:

```
SELECT *from booking_tickets WHERE price=(SELECT MIN(price) FROM booking_tickets);
```

Ticket_Number	Movie_name	Venue	Payment_type	Price
547	Pushpa: The rise	Bhaktapur	khalti	300

Figure 24: To get the cheapest movie ticket

- e. To get the length of the text in the “movie_name” column from movie_show table.

Query: `SELECT LENGTH(Movie_name) FROM movie_shows;`

LENGTH(Movie_name)
3
4
16
10
3
5
20

Figure 25: length of the text in the movie_name

- f. Get the seat number and seat row column together

Query: `SELECT CONCAT(seat_number,",Seat_row) FROM seats;`

CONCAT(seat_number,",Seat_row)
904A
1056X
879D
441Q
126Z
246M
477L

Figure 26: seat number and seat row column together

- g. Get all the employees name in upper case.

Query: Select upper (Employee_name) From employee;

upper (Employee_name)
SHIRA KHATIWADA
KRITIKA POKHREL
MANILA ADHIKARI
RAHUL SHAH
UJJUWAL BHATTRAI
AARUSH DHUNGANA
AARUSHI DHUNGANA

Figure 27: employees name in upper case

- h. Get the seat row in ascending order

Query: SELECT * FROM seats ORDER BY Seat_row ASC;

seat_number	Seat_row	Hall_number
904	A	5
879	D	5
477	L	5
246	M	5
441	Q	5
1056	X	5
126	Z	5

Figure 28: seat row in ascending order

- i. Get movies name whose payment is done by esewa or khalti.

Query: SELECT *FROM booking_tickets WHERE Payment_type IN ("esewa","khalti");

Ticket_Number	Movie_name	Venue	Payment_type	Price
606	DDLJ	Gaushala	Esewa	475
547	Pushpa: The rise	Bhaktapur	khalti	300

Figure 29: movies name whose payment is done by esewa or khalti.

- j. Union all the hall number from table seats and movie_shows table

Query: SELECT Hall_number FROM seats

UNION ALL

SELECT Hall_number FROM movie_shows

ORDER BY Hall_number;

Hall_number
1
2
4
4
5
5
5
5
5
5
6
7
7

Figure 30: Union all the hall number from table seats and movie_shows table

6.1. Queries to drop table:

Query:

1. Drop table employee;
2. Drop table movie_shows;
3. Drop table Booking_tickets;
4. Drop table others;
5. Drop table customers;
6. Drop table seats;
7. Drop table membership_card;

7. Critical Evaluation

Doing the fear evaluation, the current system as it is in its developing phase the current scenario of the system is very good and it works completely. Inserting the details, manipulating the details, removing the details and other kind of operations can be done with a simple query.

The system allows the purchase of the ticket by both customers and by the help of employees as well. It allows a different way of payment. It allows you to reserve the seats for your use, you don't need to visit the hall/the company's office. Timing of the movie, hall number, seat number, discounts, offers are clearly mentioned which is very essential in the movie management system as due to lack of proper guidance/ proper information a huge crowd had occurred. Due to the payment there would be a clear balance sheet of the company.

The employee doesn't have any much role as the role of customer and employee looks quite similar in this system. More access must be given to the employee.

8. Future Enhancement

The Dhungana Cinemas has seen a lot of changes from its established time to this time. The system will get better and perform faster in the upcoming days. The database system/ the movie management system is successful to store the data, compare the data and perform the various tasks successfully.

As the dhungana cinemas is the top multiplex/Cinema Halls of Nepal, due to the help of this movie management system it will further help to grow up the business and give the customer a better service. The services can be added to link the card with the system by the user themselves. Film Industry is a never ending industry all around the globe any incident will be held the movies will be made based on them as the cinema halls are never going to be shut down, so is the dhungana cinemas. Enhancement on the system of movie management will help the dhungana cinemas to be at the top of Mt.Everest.

9. Summary

This project “Movie Management System” as it seems to be a simple topic in the beginning but only the bearer knows the pain, time, effort we have to give this to make this successful. Here, we have seven different tables with attributes and data as well. More the table the information can be given and stored more.

This project helps us to get the deep knowledge about the database and queries, how it runs and what operations to be operated in which condition. Making the ER diagram was the toughest task, to show the relationship to manage and handle data is not an easy job. Here, in the query every syntax must be read carefully one single mistake makes a huge mistake.

However it was not an easy task for us but learning new things and experiencing new ideas is what we need to learn each and every day. Overall, this project is very useful and helped my knowledge to get stronger in the database.

References

Buddha Publicarions PVT LTD, n.d. 2017 ed. s.l.:Buddha Publicarions PVT LTD.

javatpoint, 2021. [Online]

Available at: <https://www.javatpoint.com/dbms-er-model-concept>

Pediaa, 2017. *Difference Between Entity and Attribute*. [Online]

Available at: <https://pediaa.com/what-is-the-difference-between-entity-and-attribute/>
[Accessed 08 08 2022].

Peterson, R., 2022. *guru99*. [Online]

Available at: <https://www.guru99.com/database-normalization.html>

tutorialspoint, 2022. *data dictionary*. [Online]

Available at: <https://www.tutorialspoint.com/What-is-Data-Dictionary>