

1. Project Idea

The project aims to design a comprehensive database for a mobile phone store. The goal is to digitally and efficiently manage all store operations, including:

- Managing data related to customers, suppliers, and employees
- Tracking mobile phone types, prices, and inventory
- Recording sales transactions and generating invoices
- Tracking invoice details and linking them to specific phones and customers
- Organizing various mobile categories

Features:

- Minimizing data redundancy
- Facilitating easy querying and analysis
- Providing the ability to integrate later with a web application or management system

2. Project Data

Entity Name	Attribute Name	Data Type
Customer	Customer_ID	INT (Primary Key)
	CustomerName	VARCHAR(100)
	Customer_Email	VARCHAR(100)
	Mobile_Brand	VARCHAR(50)
Employee	Employee_ID	INT (Primary Key)
	EmployeeName	VARCHAR(100)
	Employee_Email	VARCHAR(100)
	Salary	DECIMAL(10,2)
Mobile	Mobile_ID	INT (Primary Key)
	Brand	VARCHAR(50)
	Model_Name	VARCHAR(100)
	Price	DECIMAL(10,2)
	AvailableQuantity	INT
Sale	Sale_ID	INT (Primary Key)
	Sale_Date	DATE
	Customer_ID	INT (Foreign Key)
	Employee_ID	INT (Foreign Key)

	Mobile_ID	INT (Foreign Key)
	Price	DECIMAL(10,2)
Payment	Payment_ID	INT (Primary Key)
	Sale_ID	INT (Foreign Key)
	PaymentMethod	VARCHAR(50)
	Amount_Paid	DECIMAL(10,2)
	Payment_Date	DATE
Supplier	Supplier_ID	INT (Primary Key)
	SupplierName	VARCHAR(100)
	Supplier_Email	VARCHAR(100)
Purchase	Purchase_ID	INT (Primary Key)
	Purchase_Date	DATE
	Supplier_ID	INT (Foreign Key)
Purchase_Mobile	Purchase_ID	INT (Foreign Key)
	Mobile_ID	INT (Foreign Key)
	Quantity	INT
	Cost_PerUnit	DECIMAL(10,2)
	(Primary Key)	(Purchase_ID, Mobile_ID)

3. ERD

The ERD diagram represents a comprehensive system for managing mobile phone sales and purchases efficiently

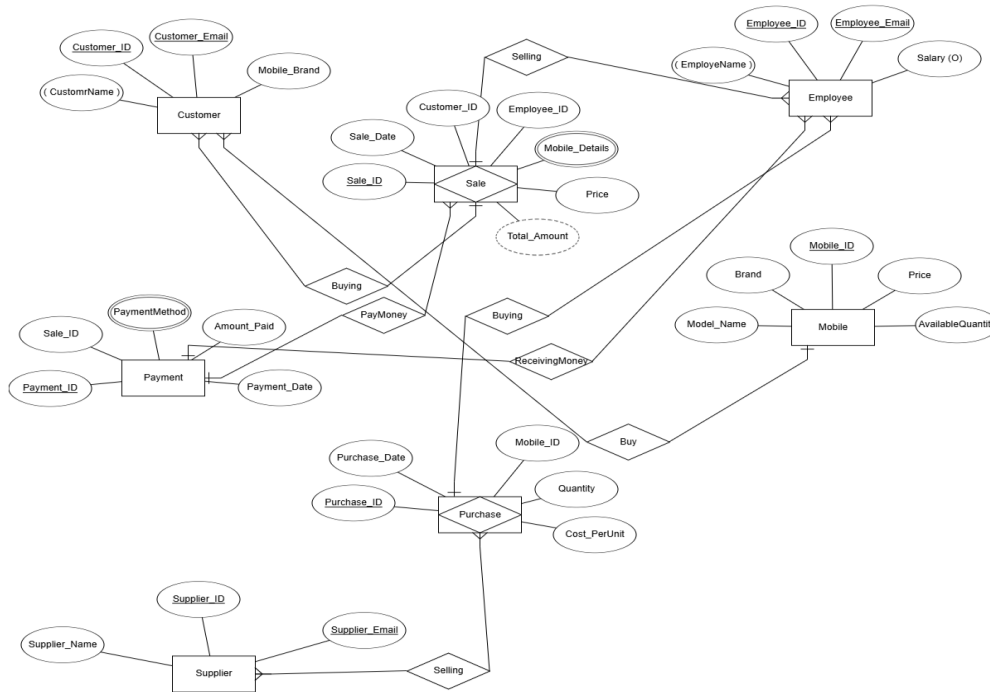
It connects customers, employees, suppliers, and mobiles through entities like Sales and Purchases

The Payment entity records details such as payment method, amount, and payment date

The diagram tracks information like price, available quantity, and total amounts for each transaction

The system is designed to facilitate tracking of commercial operations and effective inventory management

ERD Drawing :



4. Mapping

The Mapping diagram represents a database for managing mobile phone sales and purchases systematically

It includes main tables such as Customer, Employee, Supplier, and Mobile

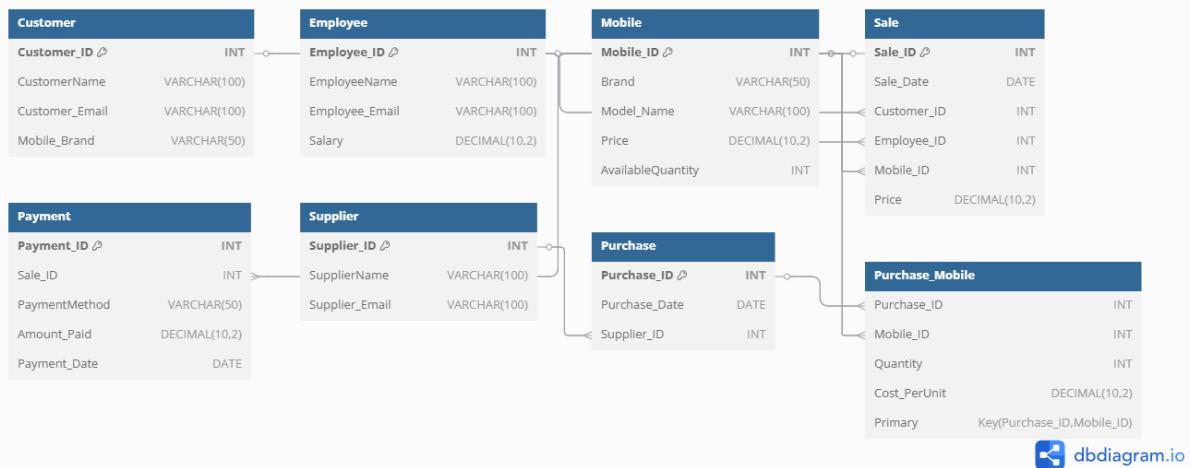
The Sale table links the customer, employee, and mobile, containing data like sale date and price

The Purchase table connects the supplier and mobile, recording purchase date, quantity, and cost per unit

The Payment table, linked to sales, includes payment method, amount, and payment date

The system efficiently manages inventory and transactions with precise tracking of all commercial operations

Mapping Drawing :



5. Normalization

Normalization techniques were applied to ensure the database structure is efficient, avoids redundancy, and maintains data integrity. The design follows the standard three normal forms:

First Normal Form (1NF):

- Each column contains atomic (indivisible) values
- No repeating groups are allowed

Second Normal Form (2NF):

- All non-key attributes are fully dependent on the primary key
- Tables are structured to isolate related data and avoid partial dependencies

Third Normal Form (3NF):

- No transitive (indirect) dependencies exist
- Related data is stored separately and referenced using keys only

Example:

Tables such as **Mobile** and **Supplier** are separated properly to avoid data duplication and ensure each piece of data depends only on the primary key of its table

This structured normalization reduces redundancy and improves maintainability and scalability of the database

6. SQL

An SQL file has been created containing a database with organized tables and customer data to build a database suitable for the shop and the project and it how we organized the file :

- **Customer:** To store customer details like name, email, and preferred mobile type.
- **Employee:** To record employee information such as names and salaries.
- **Mobile:** To list mobile details like brand, model, price, and available quantity.
- **Sale:** To track sales with data like sale date, customer, and employee.
- **Payment:** To log payment details such as method, amount, and date.
- **Supplier:** To add supplier information like names and emails.
- **Purchase:** To record purchase transactions with purchase dates.
- **Purchase_Mobile:** To link purchases with mobiles, noting quantity and cost per unit.

Then, The engaging initial data was added to each table, bringing the database to life and making it ready for action! To ensure seamless connections between tables, I used foreign keys to create strong and organized relationships, such as linking sales to customers and mobiles, and purchases to suppliers. The result? A comprehensive system that efficiently manages sales, purchases, and inventory

7. Queries

We used queries and wrote some codes in MySQL to test the data is it correct or has some problems and the result was :

- 1- We use function `"SELECT * FROM Customer;"` to show all data in Customer table
- 2- We use function `"SELECT * FROM Customer WHERE CustomerName LIKE '%Ali%';"` to search for string data where we search for customer who's name is Ali
- 3- We use function `"SELECT AVG(Salary) AS AverageSalary FROM Employee;"` to show the average of the salary of employees
- 4- We use function `"SELECT * FROM Mobile ORDER BY Price ASC;"` to show all phones arranged from cheapest to most expensive
- 5- We use function `"SELECT Brand, SUM(AvailableQuantity) AS TotalStock FROM Mobile GROUP BY Brand;"` to show all available quantity for each mobile brand
- 6- We use function `"SELECT C.CustomerName, M.Model_Name, E.EmployeeName, S.Sale_Date, S.Price FROM Sale S JOIN Customer C ON S.Customer_ID = C.Customer_ID JOIN Mobile M ON S.Mobile_ID = M.Mobile_ID JOIN Employee E ON`

`S.Employee_ID = E.Employee_ID ORDER BY S.Sale_Date DESC;` to show all sales transactions with details of the mobile, client, employee, and date of sale sorted in descending order by date