apptrainers

Translating ideas to reality through innovative education

CAPSTONE PROJECT (FOODTEK PROJECT)

GROUP (16)

Requirements Analysis and Entity Relationship Design for Delivery Management system

Web (.Net) group

Names:

Reem Abd-Alhadi

Gofran Al-Kesbah

5/4/2025

FoodTek: is a comprehensive food (delivery management system) tailored for a single restaurant. The goal of the system is to streamline and manage the entire process of food ordering, delivery, and backend operations, covering Customers, Drivers, and Admins.

This version of the project focuses on the Database Design & Requirements Analysis Phase.

Project timeline and effort:

The initial phase of the project took (**one full week**) of focused teamwork and planning, divided as follows:

Days 1-4: Analysis & Planning

During the first four days of the project, we focused heavily on understanding the system and translating business needs into a solid database structure

What we Did:

1- Studied and dissected the Business Requirements Document (BRD) carefully to extract all the business logic and system expectations.
2- Conducted 2 in-depth team meetings to break down the problem and plan the system.

First meeting Focus:

1- Identifying all entities in the system
2- Listing down every attribute for each entity
3- Mapping out the relationships between entities (one-to-many, many-to-many, etc.)
4- Discussing normalization rules to avoid data redundancy (generalization and speciation)

Second meeting focus:

In the second meeting, we focused on finalizing everything before starting the coding phase. Here's what we did:

1- We finalized the tables and made sure all the entities, attributes, and relationships were clear and correct.
2- We agreed on naming conventions for tables and columns to keep everything organized and consistent.
3- We planned how we would write the SQL code — starting with creating the tables, then inserting sample data, and finally writing stored procedures.
4- We divided the work among team members so everyone knew what to do during the coding phase.

This meeting helped us make sure everything was ready and clear before we moved on to building the actual database.

Entity and Attribute:

Shared entity for each table

(Created By, Updated By, Is Active, Update Date, Creation Date)

1- Super admin
   - (id, username, password, email)

2- Admin
   - (id, first name, last name, email, user name, phone number, password, profile image, join date)

3- Role
   - (id, Role Name in English, Role Name in Arabic)

4- Employee
   - (id, first name, last name, email, user name, phone number, password, profile image, join date)

5- Client

- (id, Image, First Name, Last Name, Email, Phone, Join Date, Status)

6- Order

- (Id, Total Price, Rate, Status, Actions)

7- Issue

- (Id, Title, Email, Status)

8- Suggest

- (id, Title, Email, Status)

9- Deliveries

- (Id, Delivery Statues, Pick Up Location, Drop Location)

10-Category

- (id, Name English, Name Arabic, Image)

10- Discount & offers

- (id, Title EN, Title AR, Description EN, Description AR, Start Date, End Date, Limit Amount, Code, image, Discount Percentage)

11- Item

- (id, Name EN, Name AR, Image, Description AR, Description EN, Price)

12- Item Option

- (id, Name AR, Name EN, Is Required)

13- Notifications

- (Id, Title, Notification Type, Is Read)

14- Driver

- (id, first name, last name, email, user name, phone number, password, profile image, join date)

15- Payment

- (Id, Amount, Payment Type, Payment Statues)

16- Reviews

- (Id, Rating, Comment)

17- Order history

- (id, total price, rate, status)

Generalization and Specialization:

1-we merge

Super admin, Admin, client, Driver, Employee in one table (users)

Relationship:

1-(Role, User) → one to many

2- (User (Admin), category)→ one to many

3- (user(admin), Discount & Offers)→ one to many

4- (user(Admin), item)→ one to many

5- (user, Item Option) → one to many

6- (user, Notifications) → one to many

7- (user , order) → one to many

8- (user (client), category) → many to many

9- (user (client), item) → many to many

10- (user ,Issue) → one to many

11- (user ,suggest) → one to many

12- (user ,Deliveries) → one to many

13-(user (client), discount & offers) → many to many

14- (user ,Reviews)→ one to many

15- (user ,Payment) → one to many

16- (user ,order history) → one to many

17- (Order, item) → many to many

18- (Order, Reviews) → One to One

19- (Order, Payment) → one to one

20- (Order, order history) → one to one

21- (Order, discount & offers) → one to many

22- (Item, item option) → one to many

23- (Item, discount & offers) → one to many

24- (Item, review) → one to many

25- (Category, item) → one to many

26- (Deliveries, Order) → One to One

27- (Offer, notification) → one to one

28- (Notification, issues) → one to many

29- (Category, discount & offers) → many to many

Setups & Installation Instructions:

This project focuses on the database layer of the FoodTek system. Below are the tools, environments, and steps we used to build and test the system locally.

Tools and Technologies used:

Purpose	**Tool/ Technology**
Main database engine	SQL Server 2019
Writing and testing SQL queries	SQL Server Management Studio (SSMS)
Development environment	Microsoft Windows
Writing BRD and documentation	Word

Future Plane:

1- Front-End Development: Develop user interfaces using ASP.NET MVC for Admin panel and Flutter for mobile applications.
2- API Development: Build APIs to connect the front-end with the database (CRUD operations, user authentication, and payment integration).

3- Database Optimization: Optimize database performance with indexing and caching techniques.
4- User Authentication & Security: Implement OAuth 2.0 for secure user login and data protection.
5- Mobile Application: Develop a mobile app using Flutter for customers and drivers.
6- Testing & QA: Conduct unit tests for SQL queries and end-to-end testing for the entire system.
7- Reporting & Analytics: Add features for generating reports on orders, deliveries, and ratings.
8- Deployment: Deploy the system to a live server for real-time use.
9- Enhancing User Experience: Add notifications and real-time tracking for orders.