PROJECT REPORT

Luis Martins

Database Design Report: Lawnmower Sales Office

1. Introduction:

This database design project aims to create a comprehensive system for a lawnmower sales office, defining entities that are part of the business's functioning. The resulting database will not only define internal processes but also improve customer experience through proper information management.

2. Task One - Database Design:

2.1 - Clearly explain why each entity needs to be included. For example if you have a staff entity explain how this benefits the overall dataset.

The identification of entities for the lawnmower sales office database started with the following classes:

- Stock
- Staff
- Customers
- Orders
- Deliveries

This task was not done all in one go, it was an ongoing, iterative process. New entities, such as Suppliers, Payment, and Services, were added as the need arose, as I progressively worked on different tasks that the project consist of.

2.2 - Identify the relevant entities of the dataset with their respective attributes, entity types (strong or weak) and primary keys.

1- Requirement Analysis

The initial phase focused on understanding the project specifications and requirements. Entities like Customers, Orders, Deliveries, and Stock were identified, as being part of the spec.

2- Identification of Entities

Following the requirement analysis, other entities such as Address, Product, Role and Supplier were identified as needed.

3- Normalization

The normalization process was thought before-hand, and applied to eliminate data redundancy and dependency issues. This involved breaking down larger tables into smaller, well-organized tables, ensuring each table had a specific purpose and followed normalization principles. This was when entities like OrderItem and Status were identified.

4- Attribute Definition

For each identified entity, attributes were defined to store information. Attributes included primary keys, such as CustomerID or ProductID, and other attributes like FirstName, LastName, Model, and Price.

5- Entity Type Determination

Entities were classified as strong or weak based on their independence and relationship with other entities. Strong entities, like Customer or Product, had a standalone existence, while weak entities, like OrderItem or Stock, depended on others.

6- Primary Key Assignment

Primary keys were assigned to each entity to ensure a unique identifier for database records. This involved usually in a simple Integer auto-incremental field called after the entity, plus “ID”, e.g ProductID.

7- Validation and Refinement

The entire model structure was reviewed. Any discrepancies or redundancies identified during normalization were addressed, and the model was improved and polished as needed.

2.3 Identify the relationships, cardinalities and constraints with supporting business rules and assumptions.

1- Relationship Identification

Relationships between entities were identified by examining how they interacted within the system. For example, the relationship between Staff and Store was established as any Store needs employees to run (Staff).

2- Cardinality Determination

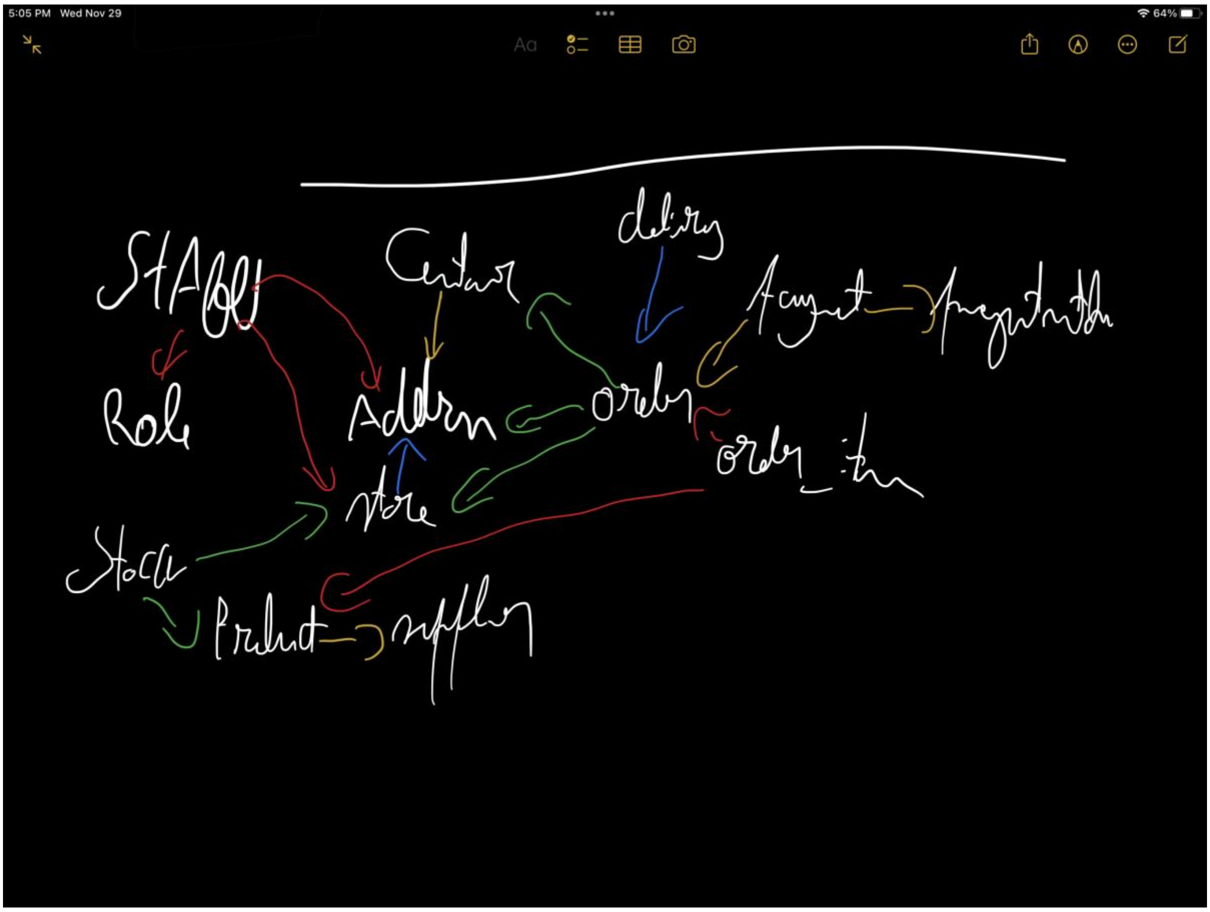
The cardinality of each relationship was determined by determining how the number of instances in one entity related to the number of instances in another. The cardinality of Order and Customer was identified as Many-to-One, since one customer can place many orders.

3- Constraints Formulation

Constraints were formulated to define the rules governing the relationships. These constraints ensured data integrity and adherence to business rules. For example, in the relationship between Product and Status, the constraint specified that each product could have only one status.

4- Business Rules and Assumptions

The final step involved defining business rules and making assumptions based on the identified relationships and constraints. These rules and assumptions clarified the expected behaviour of the system. For instance, in the relationship between Store and Address, the business rule stated that each store had a unique address.


5:05 PM Wed Nov 29
64%
Aa
Role
Address
order
order_item
store
Stock
Product

2.4 Draw an ER diagram for the system depicting the entities, relationships, cardinalities, participations using your preferred ERD notation.

1- Initialisation:

To draw the ER diagram, firstly I went back to previous power-point presentations presented on previous classes, to understand how I should design the diagram.

2- Tooling and Set-up:

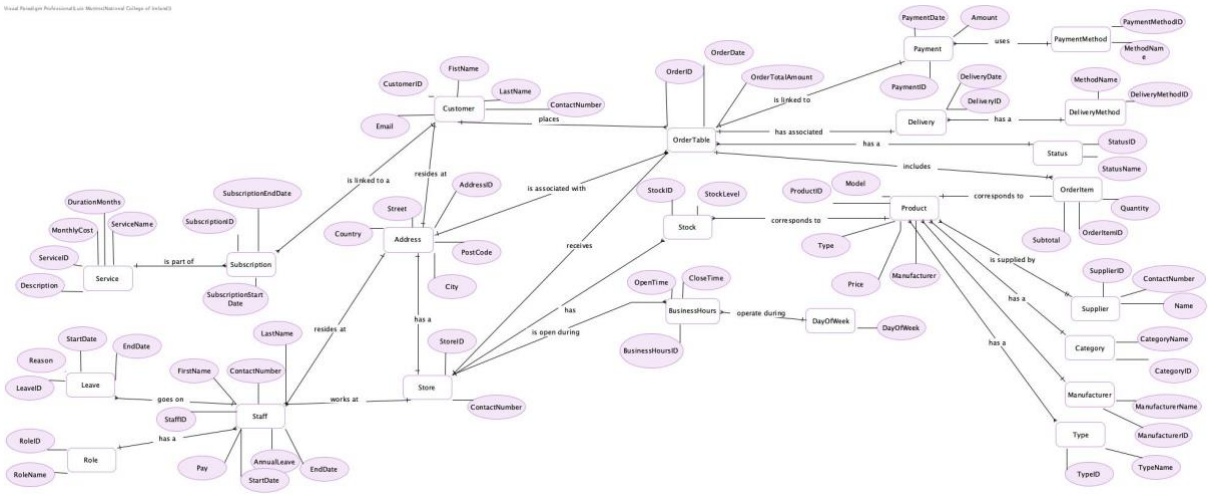
The second step involved launching Visual Paradigm and selecting the ERD diagram type to begin the process.
Placed the main entities identified during the database design phase onto the canvas, including Stock, Staff, Customers, Orders, Deliveries, Suppliers, Payment, Services, Address, Product, Role, OrderItem, and Status.

3- Relationship Establishment:

Utilizing Crow's Feet notation to establish relationships between entities, I visually indicated the cardinality of the relationships.
Connected entities based on the relationships identified in the previous tasks, using lines with appropriate crow feet symbols to represent one-to-one, one-to-many, and many-to-many relationships.

4- Refinement and Validation:

Regularly reviewed the evolving ERD to validate its alignment with the database design from previous tasks, and project requirements.



2.4 Convert the entities in the conceptual design into logical dataset(relational model). Use a logical layout of entities which is easily understood by any reader. Ensure that tables are in the correct format to allow for the best possible representation of data.

The logical dataset follows the design of the ERD done for task 5. Each table represents a distinct entity, with columns reflecting the attributes and relationships established during the design phase. The conversion process maintains the clarity and comprehensibility of the initial model, ensuring that any reader can easily understand the logical organization of the lawnmower sales office database.