

Foodlawn Database

Group 8

Project Members:

Harish Annavajjala
Aditya Rajkarne
Supreeth Shivanand

Guided by:

Elli Bourlai

CONTENTS

- **Project Description**
- **List of Project Members and a brief description of their expected roles**
- **Client Profile**
- **User Profile**
- **Database Objectives**
- **Data Model**
- **Data Profile**
- **User-centered Design Decisions**
- **Queries**
- **Database Assessment**

Project Description:

Foodlawn is a premier food court in Indiana University, which caters to thousands of students every day. Additionally it has around a hundred employees. The food court currently doesn't have a proper database and all its information is stored in a crude manner like in excel files. The food court needs to keep track of all the employees, their work hours, pay (including perks like employee meals). Additionally, they need to maintain a record of food orders, items, billing and as well as the customers. A database that has tables for the functions mentioned above can be used as a solution. We are aiming to create a database so as to manage all the information that is related to this food court. The users would be the manager and employees working in the food court.

Group members:

Harish Annavajjala

Aditya Rajkarne

Supreeth Shivanand

All the team members equally participated in all the tasks like deciding the project, gathering the user requirements, designing the database, creating the database, running various queries on it etc.

Database Environment:

A. Client Profile:

Foodlawn is a university based food court, run by 150 employees, catering to around 2,500 students on campus. Along with full timers, the organization consists of 100 part-time employees. The food court has concept stalls (ex:salad) which sell a particular set of food items. The client needs a centralized data system to store all its employee, employee shifts, customer and sales data. The system of maintaining paper records is not only vulnerable to data loss but also susceptible to human entry errors. Adding to that it is very inefficient to retrieve data and can be a time consuming process.

A detailed diagram of the organization's information flow/business process before the database implementation is shown in Figure 1.

The food court would benefit from the database in several ways:

(1) It will provide a unified, centralized system for entering and storing employee info.

- (2) It will reduce the risk of errors they were susceptible to while using a paper-based system.
- (3) It will reduce the amount of time necessary to record related information, increasing the efficiency.
- (4) It will allow them to automate many of their business processes as maintaining salary record and employee meals allocated.

Data Flow Diagrams:

Below diagrams give a glimpse of how information flows through the system. The below diagram shows how information flows before implementation our database.

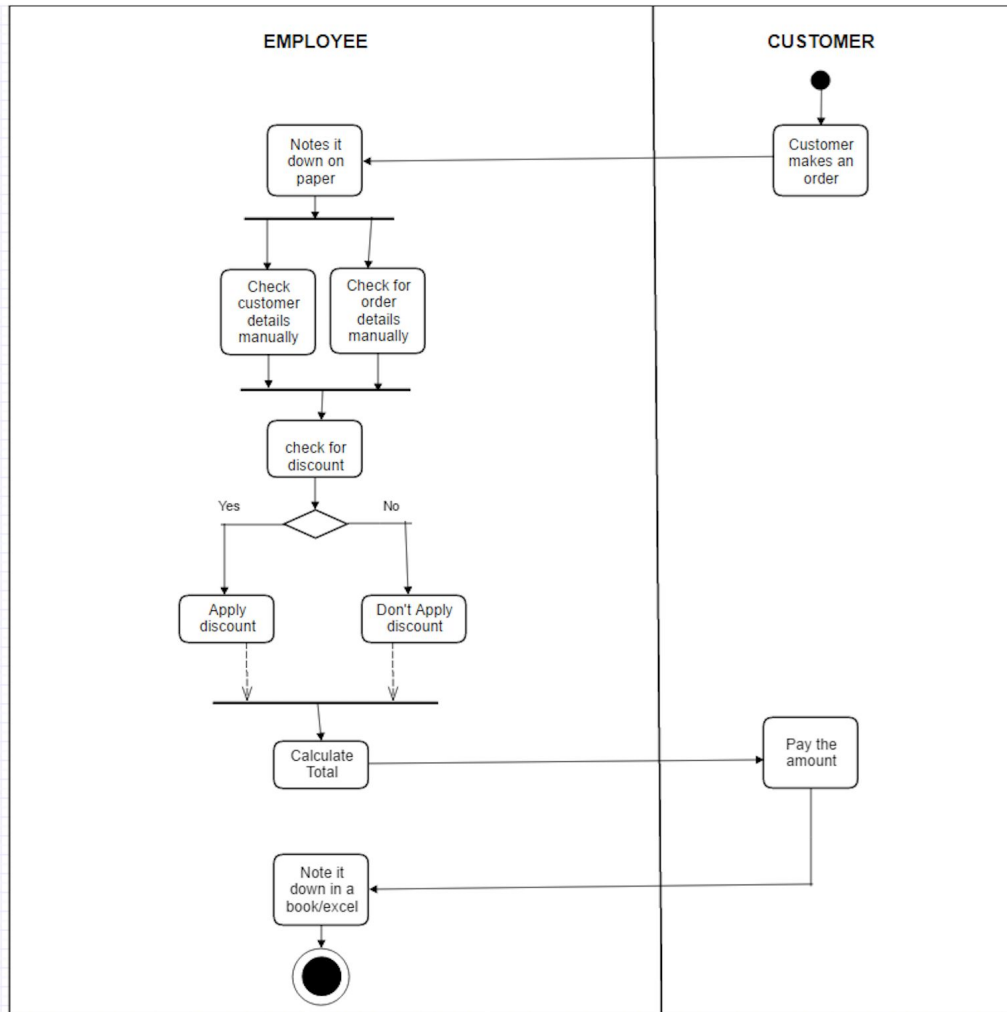


Fig. The diagram shows how information flows before implementing our database.

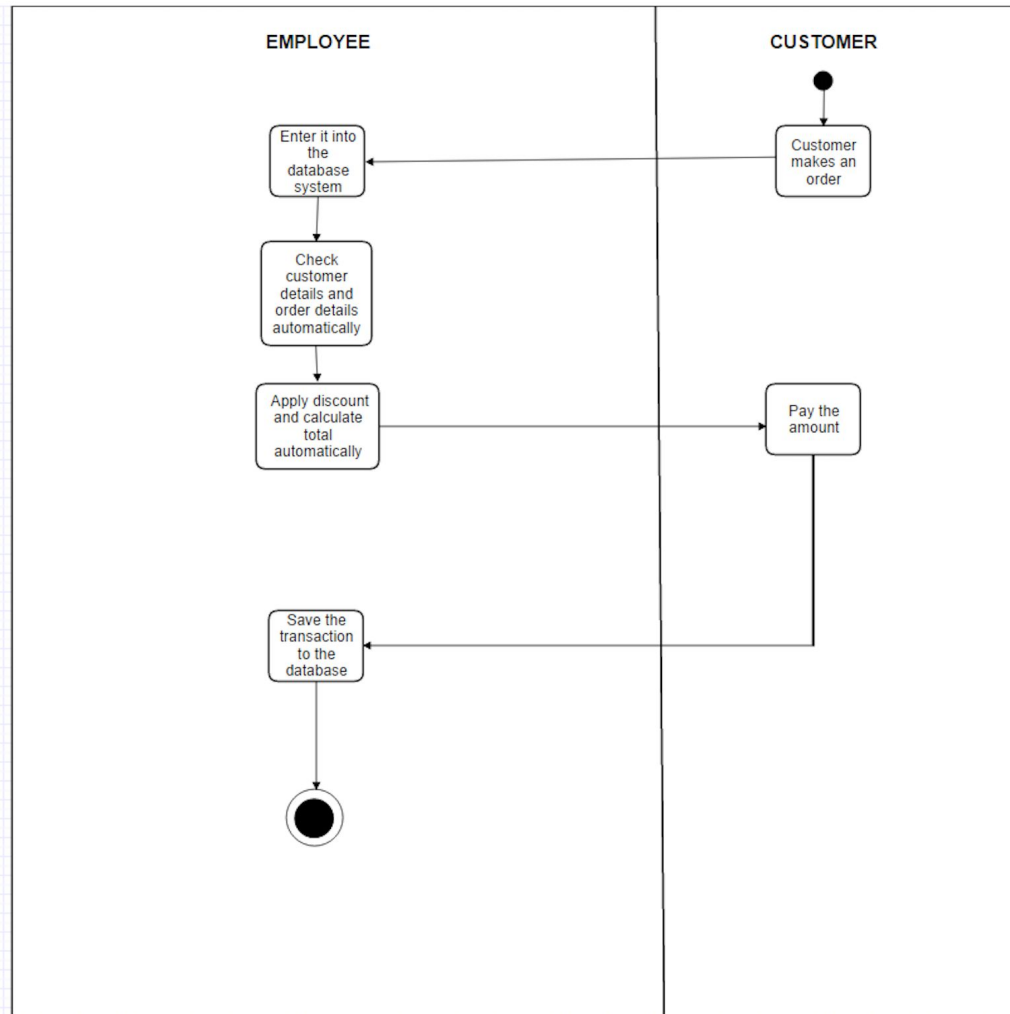


Fig. The diagram shows how information flows after implementing our database.

B. User Profile:

The users of the database are the employees of the food court who receive and cash out orders.

They will enter the items belonging to their respective concept stalls, add all such items and apply discounts based on the customer profile. The database will also keep track of their personal and contact details as well as their shift assignments.

The managers are responsible for recording employee information (e.g. name, address, mailing address, etc.) and maintain/update shift details. The food court also gives employee meals to employees working on a shift.

Employee and shift information is currently recorded by hand, creating many opportunities for error on the part of the managers recording this information. The manager then enters this information into an Excel Spreadsheet.

Goals and Objectives:

Client goals & user requirements for the database :

Streamlining the data entry process.

Providing a unified, centralized system for entering and storing orders along with the customers who got these orders.

Keeping track of employee records and their assignments.

Allowing the food court employees to quickly and easily generate bills based on not just the orders but also the kind of customers who gave these orders.

Database objectives :

Eliminating the need for a paper based system to store and access information about the billing, employees, etc .

Creating a more efficient and uniform method of storing information.

Providing the ability to access all information in one place.

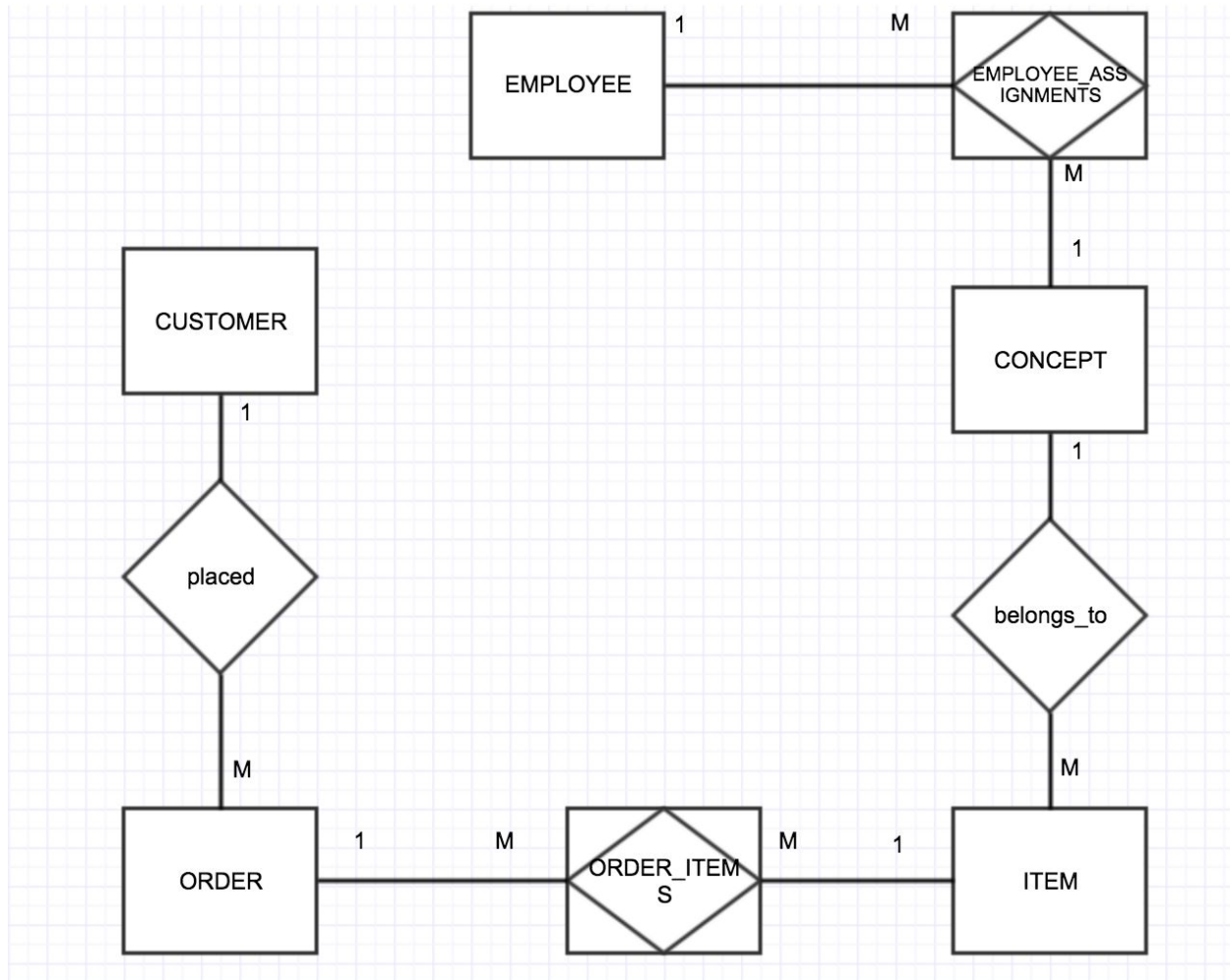
Create a database that is easy/intuitive for users with various levels of computers skills, since data will be entered by various employees of the food court.

Database Model:

Business Rules:

1. Each Customer can place many orders. Each order can be placed by only one Customer.
2. Each Order can have many Order Items. Each Order Item belongs to one order along with a quantity.
3. Each Order Item has an Item. An Item can be in many order items.
4. Each Item belongs to a concept. Each concept can have many items.
5. Each Employee can be assigned to one Concept at any time, but can have many different assignments at different times.
6. Each Concept can have many Employees.

E-R Diagram:



List of Tables:-

EMPLOYEE (employee_id, employee_name, employee_type, address, contact_number , email_id, payrate)

CONCEPT (concept_id, concept_name, description , from , to)

ITEM (item_id, item_name, concept_id, price, description)

EMPLOYEE_ASSIGNMENT (serial_num , employee_id, date, from, to, concept_id)

ORDER(order_id , customer_id , time , total)

ORDER_ITEMS: (order_id , item_id , quantity)

CUSTOMER: (customer_id , customer_name, contact, email_id, discount)

Entities and Relationships:

EMPLOYEE:

Primary Key: employee_id (Employee ID)

This table contains information about all the employees in the food court along with their name, contact number, email ID, Employee Type(Full-Time or Part-Time), Address, Payrate(in USD per hour)

CONCEPT:

Primary Key: concept_id (Concept ID)

This table contains information about each concept in the food court. Concept is a kiosk in the food courts which serves a particular set of cuisines of food. This table contains name and description of the concept and from_time to to_time when the concept is open each day.

ITEM:

Primary Key: item_id (Item ID)

Foreign Key: concept_id (CONCEPT(concept_id))

This table contains information about the food items served in the food court along with its associated concept, price and description.

EMPLOYEE_ASSIGNMENT:

Primary Key: serial_num (Serial Number)

Foreign Key: employee_id (EMPLOYEE(employee_id))

Foreign Key: concept_id (CONCEPT(concept_id))

This is a bridge table connecting Employee table and Concept table breaking Many-to-many relationship between them. This table contains information about all the Employee shifts assigned to a concept along with the time information.

ORDER:

Primary Key: order_id (Order Id)

Foreign Key: customer_id (CUSTOMER(customer_id))

This table contains all the order details like customer id of the customer who placed the order and the time when the order was placed along with the total amount of the order.

ORDER_ITEMS:

Composite Primary Key: order_id and item_id (Order Id and Item Id)

Foreign Key: order_id (ORDER(order_id))

Foreign Key: item_id (ITEM(item_id))

This is a bridge table between Order table and Item table to break the many-to-many relationship. This table has an entry for each item in an order along with its quantity.

CUSTOMER:

Primary Key: customer_id (Customer Id)

This table contains all information about the customers of the food court like their name, contact number, email id and the discount percentage that they are eligible for.

Table Dictionary:

Table	Attribute	Key	Ref. Table of FK
Employee	empID	PK	
	empType		
	empFirst_Name		
	empLast_Name		
	contact		
	address		
	email		
	payrate		
Emp_Assign	sno	PK	
	empID	FK	Employee
	date		
	from		
	to		
Concept	conceptId	FK	Concept
	conceptId	PK	
	conceptName		
	description		
	from		
Item	to		
	itemID	PK	
	itemName		
	conceptId	FK	Concept
	price		
Order_Items	description		
	orderID	PK,FK	Order
	itemID	PK,FK	Item
Order	quantity		
	orderID	PK	
	custID	FK	Customer
	dateTime		
Customer	total		
	custID	PK	
	cusFirst_Name		
	cusLast_Name		
	contact		
	email		
	discount		

DB Profile:

- Provide an ease of access to all the information.
- Ability to perform insert , update and delete data to/from the database.
- Obviates unnecessary paper based storing and retrieval system.
- The users can interact with this system either directly or through an interface.
- The user interface would be very interactive, intuitive and designed in such a way that even anyone with basic computer skill can use it comfortably. It would have all the necessary GUI components and backend layer to handle all the crud operations.

User-centered Design Decisions :

The database has been designed after taking all the user requirements into consideration. Some of the decisions:

1. We kept the database simple and intuitive to understand as users might not be tech savvy.
2. All the data corresponding to one thing is placed in one table so that it would be easy for the user to look/debug into the database.

Database Assessment:

- ❖ For the client, database has the ability to store, access, update and delete information.
- ❖ It has completely obviated the need for maintaining a paper based information tracking system.
- ❖ Our database not only meets all the user requirements but also improves data integrity and consistency.
- ❖ Limitations and Future Scope:
 1. Maintaining a log of all data modifications.
 2. Adding constraints for data entry operations.

Queries:

1. Average Pay Rate of FullTime employees:

```
select avg(payrate) from "EMPLOYEE" group by employee_type HAVING  
employee_type='FullTime';
```

2. Print the total money an employee has earned from the shifts along with his/her details :

```
select e.employee_id, e.employee_name, e.employee_type, employee_hours.hours,  
employee_hours.hours*e.payrate as total_earned from  
(select employee_id, sum("to"-"from") as time_worked, EXTRACT(HOURS FROM  
sum("to"-"from")) as hours from "EMPLOYEE_ASSIGNMENT" group by employee_id)
```

```
employee_hours, "EMPLOYEE" e where e.employee_id = employee_hours.employee_id;
```

3. Print the final total cost after applying discount for each order:

```
select order_cost.order_id, order_cost.total_cost, order_discount.discount,  
(order_cost.total_cost - order_cost.total_cost*order_discount.discount/100) as final_cost from
```

(select order_id, sum(cost) as total_cost from

(select oi.item_id, oi.order_id as order_id, i.price, oi.quantity, oi.quantity*i.price as cost
from "ITEM" i, "ORDER_ITEMS" oi where oi.item_id = i.item_id) as cost_table group by
order_id) as order_cost,

(select o.order_id as order_id, o.customer_id, c.discount as discount from "ORDER" o,
"CUSTOMER" c where o.customer_id = c.customer_id) as order_discount

where order_cost.order_id = order_discount.order_id;