Database Systems Project: MouKhane Electronics Shop

*Project Submission Report*

Mohammad Al Moukhallalati, Khaled Al Madani, Jane Daou

**Abstract**—This report illustrates the design and implementation of a new Electronics shop MouKhane. This website contains a database of items and services items can buy or acquire, in addition to extra features like registering for courses and competitions for customers who become members. The functionalities provided by the website will allow to access, process, change, and delete information found in the database.

**Index Terms**—Customer, Item, Member, Staff

# Contents

[contents 1](#_Toc122297611)

[1 Introduction 1](#_Toc122297612)

[2 Background 1](#_Toc122297613)

[3 Proposals 2](#_Toc122297614)

[4 Experimental Evaluation 3](#_Toc122297615)

[5 Conclusion 5](#_Toc122297616)

[**References** 5](#_Toc122297617)

# 1 Introduction

T

HIS document is the project report of the electronics shop MouKhane project, including explanations regarding the website development, database definition and manipulation, and the connection between both. This project consists in developing a DB and related software applications to handle the information needs of the shop. Electronics shop usually has a DB containing details regarding the items, services, customers, staff, as well as information about customers who are members, and the extra features these members get. These features include registering for courses, which professional teachers give them, and participating in competitions where the winner will get a reward. Customers can buy items or ask for services like soldering and help in projects, in which staff members will help them.

# 2 Background

## 2.1 Context and Problem Analysis

The analysis of the objective in hand was done by first deciding the functionalities with the DB system be able to provide to different types of users (including admins and customers).

The admin is responsible for the allocating rewards to the winners of competitions and assigning the teachers of the courses.

The customer can:

1. Buy items from the purchase page.
2. Acquire services like soldering and guidance in projects. Staff members will help them in these services.
3. Become members.

The members can:

1. Register for courses which are taught by professional teachers.
2. Participate in competitions where the winner of each competition will get a reward.

The analysis and implementation of this project is based upon the above-mentioned functionalities, which will be exploited and explained further in the following sections of the report.

## 2.2 Information Needs

The information needs include information regarding items, related to mainly the item type, price, and quantities found in the storage. Also, information related to the customers are needed, including their names, addresses, emails, and phone numbers.

# 3 Proposals

## 3.1 Conceptual Data Model (CDM)

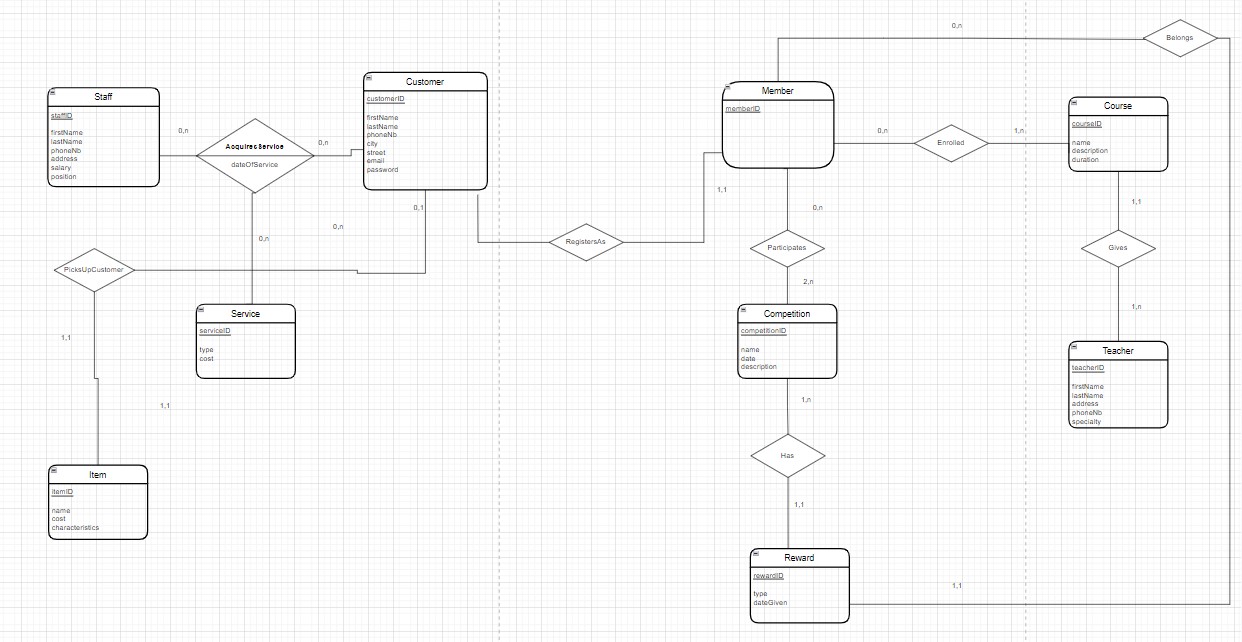
The conceptual data model (fig. 1) shows the existence of 9 entity types: Staff, Item, Customer, Item, Member, Competition, Course, Teacher, and Reward.

Figure 1: Conceptual Data Model.

Notes regarding the conceptual data model:

1. Staff entity type: A staff member is identified by a staffID (Primary Key). Attributes firstName, lastName, phoneNb, address, salary, and position are present to keep track of the staff members’ personal information.
2. Service entitity type: A service is identified by a serviced (Primary Key), and each service has a type and cost.
3. Item entity type: An item is identified by an itemID (Primary Key), and each item has attributes name, cost, and characteristics.
4. Customer entity type: A customer is identified by a customerID (Primary Key). Attributes firstName, lastName, phoneNb, city, street, email, and password are present to keep track of the customers’ personal information.
5. Member entity type: A member is identified by a memberID (Primary Key).
6. Competition entity type: A competition is identified by a competitionID (Primary Key), and each competition has a name, date, and description.
7. Reward entity type: A reward is identified by a rewardID (Primary Key), and each reward has attributes type and dateGiven.
8. Course entity type: A course is identified by a courseID (Primary Key), and each course has name, description, and duration.
9. Teacher entity type: A teacher is identified by a teacherID (Primary Key). Attributes firstName, lastName, address, phoneNb, and specialty are present to keep track of the teachers’ personal information.
10. 8 relationship types: AcquiresService, PicksUpCustomer, RegistersAs, Participates, Has, Gives, Enrolled, Belongs.

## Logical Data Model (LDM)

The Logical data model was obtained by simply converting the conceptual data model to a logical data model that was acquired during the class lecture.

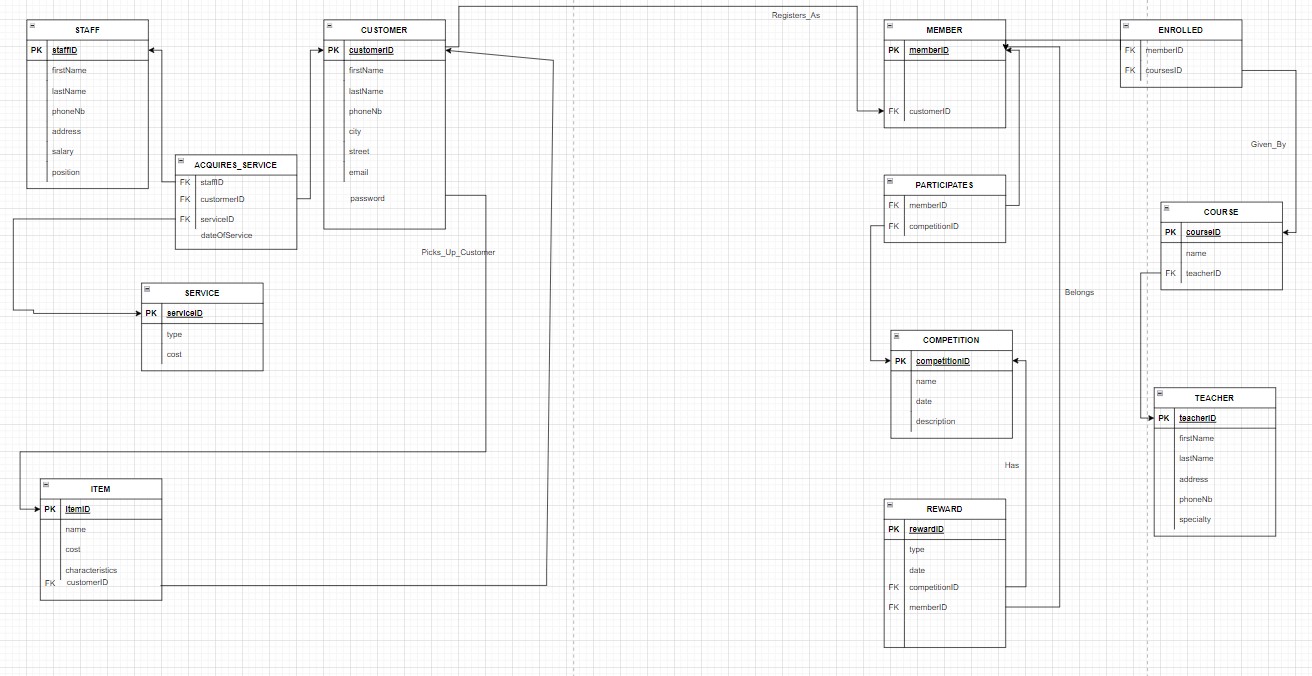


Figure 2: Logical Data Model.

## Software Application and Database Design and Implementation

The tools that were used in this project are the following:

1. WAMP Server v 2.5
2. Apache Server v 2.4.6
3. PHPMyAdmin v 4.1.14
4. MySQL Community Server v 5.6.17
5. PHP v 5.5.12
6. HTML/CSS/Javascript

WampServer is a Windows web development environment [1], which allows creating web applications with Apache2, PHP, and a MySQL database. Alongside, PHPMyAdmin helps in managing databases easily since it allows users to access and manipulate the databases created and import the database sql script to connect the website to the database. This is the backend part description of the project.

The design of the website was done using HTML and CSS, in addition to one page (purchase page) done using HTML, CSS, and Javascript. Using these helped in giving the website a better-looking design. Some of the nice design features in the website are the purchase page which contains a cart to buy items, and the links to Twitter, Facebook, and Instagram by just pressing their respective icons in the Contact section of the homepage.

# 4 Experimental Evaluation

This section will provide query tests as both expert and non-expert users. Non-expert users are bounded to the software interface, providing for example a bunch of keywords, through textboxes. Expert users can write the SQL statements without the need of a software or a graphical interface. This section will show the non-expert and expert uses of all the functionalities of the website.

1. Login page:

The customer cannot access the website unless he/she have a customer account. If the customer does not have an account, he/she can register as a customer.

**Non-expert:** The non-expert user who wants to register an account will be provided with textboxes where he/she can specify their names, address, phone number email, and password. The email and password will be used in the future if he/she wants to sign into their account. If the non-expert user has an account, all he/she has to do is to insert his/her email and password correctly in order to login to the website.

**Expert:** The input of the registeration of an account will be translated into the SQL statement:

INSERT INTO CUSTOMER VALUES (‘John’,’Doe’, ‘70926606’, ‘Beirut’, ‘Sabra’, ‘john@email.com’, ‘haha’);

The input of the logging into the customer’s account of an account will be translated into the SQL statement:

SELECT \* FROM CUSTOMER WHERE email=’$email’ AND password=’$pass’;

1. Homepage:

In the homepage, there are buttons which take the customer to a new page where he/she acquire services and buy items. Members can also access the “Become a Member!” page where he/she can register for courses and competitions.

1. Acquire a service:

**Non-expert:** The non-expert user will be provided with textboxes where he/she can put the service he/she wants to acquire, and the time he/she will come to the shop. The non-expert user will also be asked to put his/her email to keep record of the data into the database. The “Submit your service registeration” button allows the user to confirm the service registeration.

**Expert:** The functionalities above can be translated into the following SQL statements:

The following is to choose a staff member that is available:

SELECT \* FROM staff s WHERE s.staffID <= ALL(SELECT s.staffId FROM staff s WHERE s.staffID NOT IN (SELECT s.staffID from staff s,acquires\_service a,service se WHERE se.type = 'Soldering' AND s.staffID = a.staffID AND se.serviceID = a.serviceID AND s.position = 'Helper' AND a.dateOfService = '$date'))

The following is to insert data into the acquires\_service table in the database:

INSERT INTO acquires\_service VALUES('staffID','$customerID','serviceID', '$date')

1. Purchase items:

**Non-expert:** The non-expert user will be provided with a list of items he/she can put in a cart, in which he/she can select multiple items with multiple quantities to buy. The purchase occurs when the user clicks on the “BUY NOW!” button.

**Expert:** The functionalities above can be translated into the following SQL statements:

INSERT INTO purchase VALUES ('itemID','customerID','$price')

1. Becoming a member:

**Non-expert:** The non-expert user can become a member, if he/she are not registered as a member before, by putting his/her email and password in the textboxes. If he/she are already registered as a member, he/she can just press the “Click here” link and this will take him/her to the “Become a Member!” page.

**Experts:** The functionalities above can be translated into the following SQL statements:

INSERT INTO member VALUES ('memberID','$email', '$pass' ,'$customerID')

1. Registeration for courses as a member:

**Non-expert:** The non-expert user will be provided with a set of textboxes where he/she will type which course he/she register and his/her username to keep record of the data in the database. The “Register for course” button allows the user to confirm the course registeration.

**Expert:** The functionalities above can be translated into the following SQL statements:

The following is to select memberID:

SELECT memberID FROM member WHERE email = '$email'

The following is to insert the data in the table of the database:

INSERT INTO emrolled VALUES('memberID','courseID')

1. Registeration for competitions as a member:

**Non-expert:** The non-expert user will be provided with a set of textboxes where he/she will type which competition he/she register and his/her username to keep record of the data in the database. The “Register for competition” button allows the user to confirm the competition registeration.

**Expert:** The functionalities above can be translated into the following SQL statements:

The following is to select memberID:

SELECT memberID FROM member WHERE email = '$email'

The following is to insert the data in the table of the database:

INSERT INTO participates VALUES('memberID','competitionID')

1. Admin page:

The admin page can be accessed by the admin inserting his/her email and password. There will be a certain email and password which will take the admin to the admin page. In this admin page, the admin can allocate the reward of each competition to the winner and can add teachers.

Query for admin to login:

SELECT \* FROM staff WHERE firstName = '$email' AND password = '$pass' AND position = 'Admin'

1. Allocating reward to winner:

**Non-expert:** The admin will be provided with a set of textboxes where the admin will type in the memberID, rewardID, reward type, and the date the reward is given. The “Allocate Reward” button allows the admin to successfully allocate the reward to the member.

**Expert:** The functionalities above can be translated into the following SQL statements:

INSERT INTO reward VALUES ('$rewardID','rewardType','competitionID','memberID')

1. Register teacher:

**Non-expert:** The admin will be provided with a set of textboxes where the admin will type in the name, address, phone number, and specialty of the teacher. The “Register teacher” button allows the admin to successfully add the teacher to the Teacher table in the database.

**Expert:** The functionalities above can be translated into the following SQL statements:

INSERT INTO teacher VALUES ('teacherID','fname','lname','address','phomeNumber','speciality')

Please note that the above SQL statements and functionalities are limited and minimized due to limited space. Please refer to the PHP files and website for more details.

# 5 Conclusion

This was the report of the Database System Course final project having the title “MouKhane Electronics Shop”. The different functionalities were described in the report. This project was not difficult; however, it required a lot of work and research which actually led to the acquiring of new programming languages such as PHP, CSS, HTML, Javascript.

**References**

[1] “WampServer, la plate-forme de développement Web sous Windows ..” https://www.wampserver.com/en/ (accessed: Dec. 18, 2022)