****

**CEF 440: INTERNET AND MOBILE PROGRAMMING**

**GROUP MEMBERS:**

|  |  |
| --- | --- |
| **NAME** | **MATRICULE NUMBER** |
| FORMASIT CHIJOH FOKUNAG | FE20A041 |
| GAMUAH PADJINOU RYANE JOY | FE20A044 |
| MOSONGO ADINA SAKWE NANGERI | FE20A066 |
| NDALEGH NOELA LUM MBAH | FE20A070 |
| RACHEAL NKONGHO TAFONGOH | FE20A100 |

**PROJECT TITLE: DESIGN AND IMPLEMENTATION OF A MARKET MANAGEMENT SYSTEM**

**TASK 6: DATABASE IMPLEMENTATION AND DESIGN**

**COURSE INSTRUCTOR**

DR. NKEMENI VALERY

June, 2 2023.

Table of Contents

[INTRODUCTION 3](#_Toc136622427)

[DATABASE DESIGN 3](#_Toc136622428)

[CLASS DIAGRAM 4](#_Toc136622429)

[DATABASE SCHEMA 4](#_Toc136622430)

[DESCRIPTION 4](#_Toc136622431)

[ER-DIAGRAM 5](#_Toc136622432)

[OVERVIEW OF THE APPLICATION 5](#_Toc136622433)

[TYPES OF DATA AND THEIR RELATIONSHIP 6](#_Toc136622434)

[TYPES OF DATA 6](#_Toc136622435)

[THEIR RELATIONSHIP 7](#_Toc136622436)

[DATABASE IMPLEMENTATION. 7](#_Toc136622437)

# INTRODUCTION

A database is a structured set of data. This data is stored for various reasons such as performing queries to obtain information about the business, analyzing data to understand current trends and predict future trends, which will help the organization in decision making.

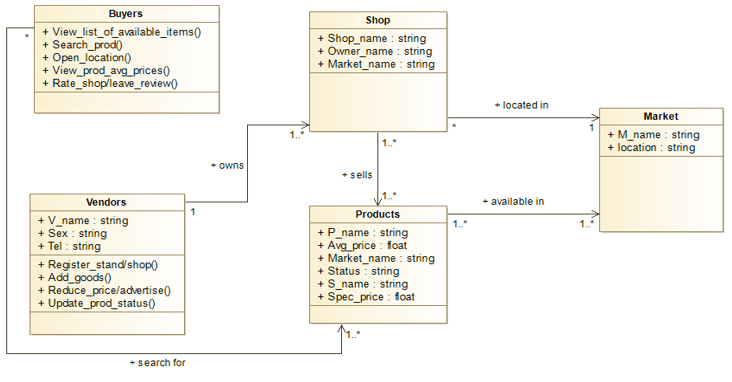
In this regard, it is of prime importance for a well-structured database to be established for our market management system.

# DATABASE DESIGN

In order to design the database, the data to be stored must be identified.

* First, the class diagram is drawn to view the required data, its attributes, and the relationship between the different sets of data to be stored.
* Next the entity relationship diagram (ER-DIAGRAM) is drawn from the class diagram, to simplify the view of the database.
* From the ER-DIAGRAM, the relational schema is gotten, which shows the identifiers of each table in the database, and how data from one table is reused in another table.

## CLASS DIAGRAM



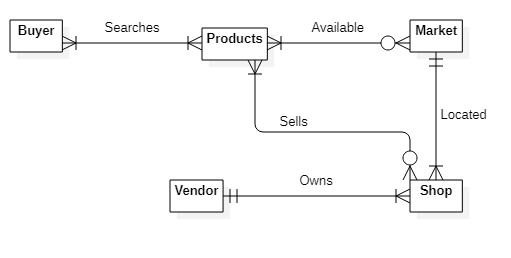
The class diagram shows the different entities in the application, and the inter-relationship between them.

## DATABASE SCHEMA

### DESCRIPTION

* A buyer searches for one or more products, and a product can be requested by one or more buyers.
* A product can be available in zero or more markets, and a market has at least one product.
* A market has one or more shops, but a shop can be found in one and only one market.
* A vendor may own one or more shops, and a shop is owned by only one vendor.
* A product can be found in zero or more shops, and a shop can sell one or more products.

### ER-DIAGRAM



# OVERVIEW OF THE APPLICATION

The flow of activities is as described below:

* The vendor registers to the application with his credentials, and the required information about the goods he sells, the location of his shop, and the market in which he sells.
* The system displays the products to the buyers, is able to sort the products in terms of category, and give an average price of a specific product to the customer for him to be able to prepare well enough for purchase.
* The buyers are not required to login to the application; their sessions are simply recorded temporarily. They are able to search for products, view products, rate shops and contact the vendor for a product they may be interested in.

From the description, it is clear that the buyer’s information is not stored in the database, and will be taken note of when building the whole application.

## TYPES OF DATA AND THEIR RELATIONSHIP

There are six tables in our database based on the requirements as earlier mentioned. The tables and the data stored are listed below:

### TYPES OF DATA

1. VENDORS TABLE

Here, we keep track of the vendor’s name, gender, password and contact.

1. PRODUCTS TABLE

The product name and average price are stored in this table.

1. MARKET TABLE

The market name and its location are recorded in this table.

1. SHOP TABLE

The shop name, vendor name, and market name are stored in this table. The vendor name and market name are referenced from the VENDORS table and MARKET TABLE.

1. SHOP PRODUCTS TABLE

The data stored here include the shop name, product name, status of the product (available, limited or not available). The shop name is referenced from the shop table, and the product name is referenced from the product table.

1. MARKET PRODUCTS TABLE

It stores the market name, and the product name. They are referenced from the market and product tables respectively.

### THEIR RELATIONSHIP

On registering, the vendor enters his personal information such as his name and contact, the products he sells, his shop and its location in the market which he is required to specify.

From the different vendor registrations, there is a list of products, markets, shops, and the products sold in the shops and markets.

# DATABASE IMPLEMENTATION.

The database is implemented in MYSQL using XAMPP. This is because MYSQL is a relational database, which will help to reduce duplicate entries in the database by linking up the tables, as data in one table may be needed in another table for further processing.

The database is implemented by executing the following command:

DROP DATABASE IF EXISTS sellam;

CREATE DATABASE sellam;

USE sellam;

The database tables and their various columns are created by executing the commands below:

CREATE TABLE vendor (

vname varchar(100) PRIMARY KEY,

sex varchar(6) NOT NULL DEFAULT 'Others',

tel varchar(14) NOT NULL,

passwd varchar(8) NOT NULL

);

CREATE TABLE product (

pname varchar(100) PRIMARY KEY,

avg\_price float NOT NULL

);

CREATE TABLE market (

mname varchar(20) PRIMARY KEY,

mlocation varchar(100) NOT NULL

);

CREATE TABLE shop (

shopname varchar(100) PRIMARY KEY,

vname varchar(100) NOT NULL,

mname varchar(100) NOT NULL,

FOREIGN KEY (vname) REFERENCES vendor(vname) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (mname) REFERENCES market(mname) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE shop\_products (

shopname varchar(100) NOT NULL,

pname varchar(100) NOT NULL,

stat varchar(10) NOT NULL,

spec\_price float NOT NULL,

disc\_price float NULL,

PRIMARY KEY (shopname, pname),

FOREIGN KEY (shopname) REFERENCES shop(shopname) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (pname) REFERENCES product(pname) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE market\_products (

mname varchar(100) NOT NULL,

pname varchar(100) NOT NULL,

PRIMARY KEY (mname, pname),

FOREIGN KEY (mname) REFERENCES market(mname) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (pname) REFERENCES product(pname) ON DELETE CASCADE ON UPDATE CASCADE

);