

**International School**

**Capstone Project 2**

## CMU-SE 447 JIS – FIMS

**Database Design**

**Version 1.2 Date: March 12th, 2024**

**FIMS Mobile Shop**

Submitted by

**Tan, Ngo Ngoc**

**Kien, Le Trung**

**Nam, Luu Hai**

**Van, Nguyen Nhu**

## Approved by

**MSc Huy, Truong Dinh**

## Proposal Review Panel Representative:

Name Signature Date

## CDIO - Mentor:

Name Signature Date

**PROJECT INFORMATION**

|  |  |  |  |
| --- | --- | --- | --- |
| **Project acronym** | LET | | |
| **Project Title** | Phone Store Manager | | |
| **Start Date** | 09 Jan 2024 | **End Date** | 12 Mar 2024 |
| **Lead Institution** | International School, Duy Tan University | | |
| **Project Mentor** | MSc Huy, Truong Dinh | | |
| **Scrum master / Project Leader & contact details** | Tan, Ngo Ngoc  Email:ngoctan4677@gmail.com  Tel: 0918809264 | | |
| **Partner Organization** | Duy Tan University | | |
| **Project Web URL** |  | | |
| **Team members** | Name | Email | Tel |
|  | Tan,Ngo Ngoc | [ngoctan4677@gmail.com](mailto:ngoctan4677@gmail.com) | 0918809264 |
|  | Kien, Le Trung | [letrungkien6@dtu.edu.vn](mailto:letrungkien6@dtu.edu.vn) | 0935632864 |
|  | Nam, Luu Hai | hainamluu01@gmail.com | 0972519223 |
|  | Van, Nguyen Nhu | [nguyennhuvan@dtu.edu.vn](mailto:nguyennhuvan@dtu.edu.vn) | 0345193154 |

# DOCUMENT APPROVALS

The following signatures are required for approval of this document.

|  |  |  |
| --- | --- | --- |
| Tan,Ngo Ngoc  Student ID: 27211237763  *Scrum Master* | Signature | Date |
| Kien, Le Trung  Student ID: 27211202741  *Team Member* | Signature | Date |
| Nam, Luu Hai  Student ID: 27211201241  *Team Member* | Signature | Date |
| Van,Nguyen Nhu  Student ID:27211200722  *Team Member* | Signature | Date |

**REVISION HISTORY**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Comments** | **Author** | **Approval** |
| 1.0 | Jan 20th, 2024 | Initial Release, Update Introduction, Database Diagram | Kien, Le Trung,  Van, Nguyen Nhu |  |
| 1.1 | Jan 22th, 2024 | Update Database Design for Sprint,  Hardware and Software Requirements | Team |  |
| 1.2 | Feb 27th, 2024 | Update Database | Kien, Tan |  |

# TABLE OF CONTENT

[1. Introduction 5](#_Toc156825889)

[1.1 Purpose 5](#_Toc156825890)

[1.2 Scope 5](#_Toc156825891)

[1.3 Introduction about MySQL: 5](#_Toc156825892)

[2. Database Diagram 6](#_Toc156825893)

[2.1 Table Overview 6](#_Toc156825894)

[2.2 Entity Relationship Diagram 8](#_Toc156825895)

[2.3 Table Relationship Diagram 9](#_Toc156825896)

[3. Database Design for Sprint 10](#_Toc156825897)

[3.1 Table Roles 10](#_Toc156825898)

[3.2 Table Account 10](#_Toc156825899)

[3.4 Table CustomerPaymentMethod 11](#_Toc156825900)

[3.5 Table Administrator 11](#_Toc156825901)

[3.6 Table News 12](#_Toc156825902)

[3.7 Table Promotion 12](#_Toc156825903)

[3.8 Table Category 13](#_Toc156825904)

[3.9 Table ModelCompany 13](#_Toc156825905)

[3.10 Table Product 13](#_Toc156825906)

[3.11 Table ProductImage 14](#_Toc156825907)

[3.12 Table CartItem 14](#_Toc156825908)

[3.13 Table Rating 14](#_Toc156825909)

[3.14 Table Invoice 15](#_Toc156825910)

[3.15 Table InvoiceDetail 15](#_Toc156825911)

[3.16 Table PaymentDetail 15](#_Toc156825912)

[4. Hardware and Software Requirements 17](#_Toc156825913)

# Introduction

The Database Design maps the logical data model to the target database management system with consideration to the system’s performance requirements. The Database Design converts logical or conceptual data constructs to physical data constructs (e.g., tables...) of the target Database Management System.

## Purpose

The purpose of the Database Design is to ensure that every database transaction meets or exceeds its performance requirements. This document takes into account data and transaction volume to produce a schema and environment that will meet necessary performance.

## Scope

The Database Design Document has the following objectives:

* + - To describe the design of a database, that is, a collection of related data stored in one or more computerized files that can be accessed by users or developers via a DBMS.
    - To serve as a basis for implementing the database and related software units. It provides the acquirer visibility into the design and provides information necessary for software development.

## 1.3 Introduction about MySQL:

MySQL is a popular open-source relational database management system (RDBMS) that is widely used for building and managing databases. Developed by MySQL AB, now owned by Oracle Corporation, MySQL is known for its reliability, performance, and ease of use. It is an integral part of the LAMP (Linux, Apache, MySQL, PHP/Python/Perl) stack, which is a widely adopted software stack for web development.

Key features of MySQL include:

1. Relational Database Management System (RDBMS): MySQL follows the relational model, organizing data into tables with rows and columns. This structure allows for efficient and organized storage, retrieval, and manipulation of data.

2. Open Source: MySQL is an open-source software, which means that its source code is freely available and can be modified and distributed by users. This has contributed to its widespread adoption and community support.

3. Cross-Platform Compatibility: MySQL is compatible with various operating systems, including Linux, Windows, and macOS. This flexibility makes it suitable for a wide range of applications.

4. Scalability: MySQL can handle large amounts of data and is scalable to meet the requirements of growing applications. It supports features such as replication and clustering to distribute and manage data across multiple servers.

5. Performance: MySQL is designed for high performance, providing quick and efficient data storage and retrieval. Its indexing mechanisms and optimization features contribute to fast query execution.

6. Security: MySQL offers robust security features, including access controls, authentication, and encryption, to protect data and ensure only authorized users have access to the database.

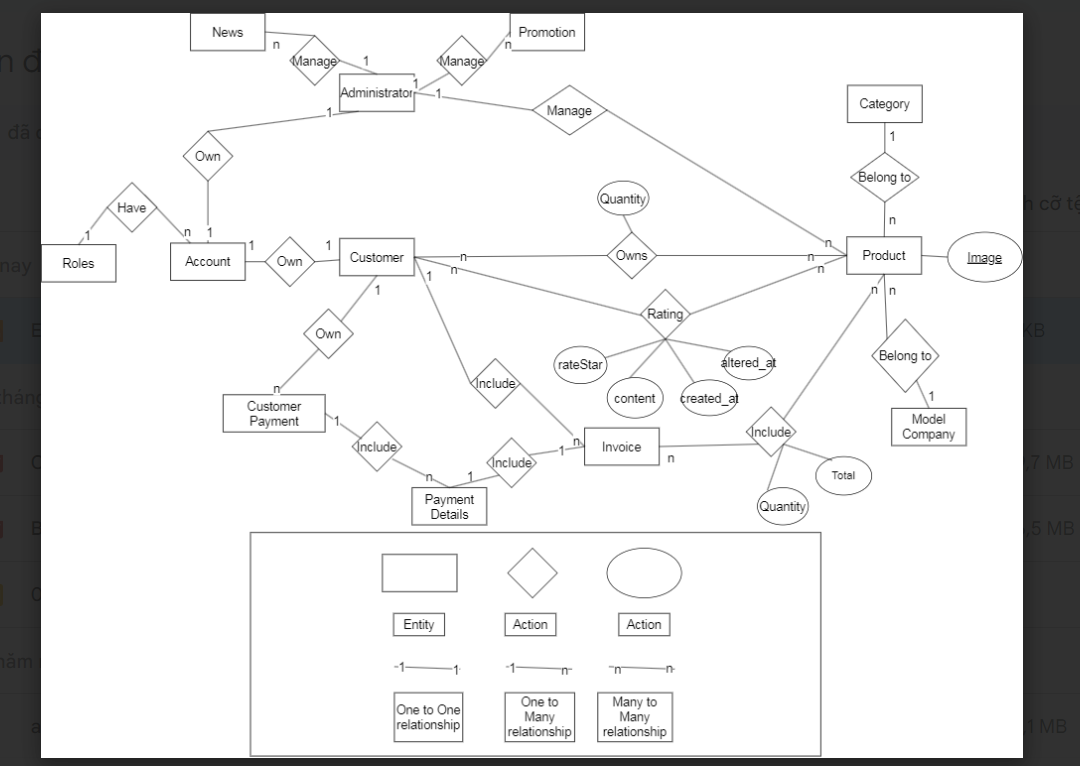
MySQL is widely used in web development, content management systems, e-commerce platforms, and various other applications where structured data storage and retrieval are essential. Its combination of performance, reliability, and open-source nature has contributed to its popularity in the software development community.

# Database Diagram

## Table Overview

|  |  |
| --- | --- |
| **Table Name** | **Short Description** |
| Roles | This table contains information about site roles |
| Account | This table store account information of all users (Including Customer and Admin) |
| Administrator | This table store admin information, and also uses queries for admin-related information. |
| News | This table stores a variety of news related to website events and announcements, such as discounts or promotional codes. |
| Promotion | This table stores the website's discount codes if a discount event takes place. |
| Customer | This table stores user information, and also uses queries for user-related information. |
| CustomerPaymentMethod | This table contains payment methods and their information details. |
| Category | This table contains the phone type category (Mobile, Tablet) |
| ModelCompany | This table contains information about phone brands (Samsung, LG, Sony, Bphone...) |
| Product | This table contains detailed information of the product (Name, Screen, Camera, RAM, ROM, ...), inventory quantity, and its unit price. |
| ProductImage | This table contains images related to a specific product model. |
| Rating | This table contains detailed review information about a specific phone model. |
| CartItem | This table lets customers select, store, and manage items before buying them |
| Invoice | This table allows customers to view invoice information after placing an order including total amount and discount rate. |
| InvoiceDetail | This table contains information about the items purchased, and their quantities |
| PaymentDetail | This table allows viewing the total payment amount, payment method and payment time. |

## Entity Relationship Diagram



* 1. **Table Relationship Diagram**

A screenshot of a computer

Description automatically generated

# Database Design for Sprint

* 1. **Table Roles**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Roles** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID roles, primary key, auto-incrementing value, starting from 1 |
| 2 | Name | Nvarchar(255) | Unique, Not Null | Name of role (Admin, Customer, ...) |

## Table Account

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Account** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID account, primary key, auto-incrementing value, starting from 1 |
| 2 | Username | Nvarchar(50) | Unique, Not Null | Account name (Unique, cannot be left blank) |
| 3 | Password | Nvarchar(255) | Not Null | Account password (Cannot be left blank) |
| 4 | Avatar | Image |  | Representative photo of the person using that account |
| 5 | Role | Int | Foreign key to Roles(ID) | Role ID of the person using this account |
| 6 | Created\_at | Datetime |  | Time to create information. |
| 7 | Altered\_at | Datetime |  | The last time since the information was edited |

* 1. **Table Customer**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Customer** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID customer, primary key, auto-incrementing value, starting from 1 |
| 2 | Name | Nvarchar(255) | Not null | Name of the customer |
| 3 | Sex | Bit |  | Sex of the customer |
| 4 | Birthdate | Date |  | Customer's date of birth |
| 5 | Address | Nvarchar(255) |  | Customer's address (Used as shipping address) |
| 6 | Email | Nvarchar(255) |  | Customer email |
| 7 | Telephone | Nvarchar(255) |  | Customer's phone number |
| 8 | Account\_ID | Int | Unique,  Not null,  Foreign key to Account(ID) | ID account that owned by the customer |
| 9 | Created\_at | Datetime |  | Time to create information. |
| 10 | Altered\_at | Datetime |  | The last time since the information was edited |

## Table PaymentMethod

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PaymentMethod** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID customer payment method, primary key, auto-incrementing value, starting from 1 |
| 2 | Name | Nvarchar(255) | Not null | Name of payment method (Paypal, Debit Card, ...) |
| 3 | Provider | Nvarchar(255) | Not null | Name of service provider (Visa, MasterClass, ...) |
| 4 | Expiration\_Date | Date | Not null | Expiration date of debit card |
| 5 | Card\_Number | Nvarchar(20) | Not null | Credit Card Number |
| 6 | Security\_Code | Int | Not null | Card security code |
| 7 | Bank\_Account\_Name | Nvarchar(255) | Not null | Bank account name (account number). |
| 8 | Bank\_Name | Nvarchar(255) | Not null | Bank name used for transactions |
| 9 | Customer\_ID | Nvarchar(255) | Not null, Foreign key to Customer(ID) | ID of the customer using the payment method |

* 1. **Table Administrator**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Administrator** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID administrator, primary key, auto-incrementing value, starting from 1 |
| 2 | Name | Nvarchar(255) | Not null | Name of the administrator |
| 3 | Sex | Bit |  | Sex of the administrator |
| 4 | Birthdate | Date |  | Administrator’s date of birth |
| 5 | Email | Nvarchar(255) |  | Administrator email |
| 6 | Telephone | Nvarchar(255) |  | Administrator 's phone number |
| 7 | Account\_ID | Int | Unique,  Not null,  Foreign key to Account(ID) | ID account that owned by the administrator |
| 8 | Created\_at | Datetime |  | Time to create information. |
| 9 | Altered\_at | Datetime |  | The last time since the information was edited |

## Table News

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **News** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID news, primary key, auto-incrementing value, starting from 1 |
| 2 | Title | Ntext | Not null | The title of the news |
| 3 | Content | Ntext | Not null | Content of news |
| 4 | Tittle\_Photo | Image |  | Opening photo of the news |
| 5 | Admin\_ID | Int | Not null,  Foreign key  Administrator(ID) | ID of the administrator who created the news |
| 6 | Created\_at | Datetime |  | Time to create information. |
| 7 | Altered\_at | Datetime |  | The last time since the information was edited |

* 1. **Table Promotion**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Promotion** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID promotion, primary key, auto-incrementing value, starting from 1 |
| 2 | Code | Nvarchar(255) | Unique, Not null | Discount code, unique |
| 3 | Discount\_Percentage | Float | Discount\_Percentage ≥ 0 and Discount\_Percentage ≤ 100 | Discount percentage of that promotional code |
| 4 | Is\_One\_Time | Bit | Not null | Mark whether this is a one-time discount code or not |
| 5 | End\_at | Date | Not null,  End\_at ≥ GetDate() | The time when the discount code expires |
| 6 | Admin\_ID | Int | Not null, Foreign key Administrator(ID) | Admin ID that generated discount code |
| 7 | Created\_at | Datetime |  | Time to create information. |
| 8 | Altered\_at | Datetime |  | The last time since the information was edited |

## Table Category

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Category** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID category, primary key, auto-incrementing value, starting from 1 |
| 2 | Name | Nvarchar(255) | Unique, Not null | Category name (Mobile, Tablet, ...) |

* 1. **Table ModelCompany**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ModelCompany** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID model company, primary key, auto-incrementing value, starting from 1 |
| 2 | Name | Nvarchar(255) | Unique, Not null | Product brand name (Samsung, LG, Sony, Panasonic) |

## Table Product

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Product** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID product, primary key, auto-incrementing value, starting from 1 |
| 2 | Name | Nvarchar(255) | Unique, Not null | Product name |
| 3 | Screen | Nvarchar(255) |  | Screen type and screen size |
| 4 | Operating\_System | Nvarchar(255) |  | Operating system version |
| 5 | Color | Nvarchar(255) |  | Device color |
| 6 | Rear\_Camera | Int |  | Rear Camera Resolution |
| 7 | Front\_Camera | Int |  | Front Camera Resolution |
| 8 | Chip | Nvarchar(255) |  | Type of CPU used |
| 9 | Ram | Int |  | RAM capacity of the product |
| 10 | Rom | Int |  | ROM capacity of the product |
| 11 | Sim | Nvarchar(255) |  | Type of SIM used and number of SIMs |
| 12 | Connect | Nvarchar(255) |  | Connection type used |
| 13 | Battery | Int |  | Battery capacity of the product |
| 14 | Describe | Ntext |  | Description of product details |
| 15 | Quantity | Int |  | Number of product in stock |
| 16 | Cost | Int |  | Price of product |
| 17 | Category\_ID | Int | Not null, Foreign key: Category(ID) | Product category ID, indicating what type the product belongs to |
| 18 | Company\_ID | Int | Not null, Foreign key: Company(ID) | Firm ID, indicating which company the product belongs to |
| 19 | Admin\_ID | Int | Not null, Foreign key: Administrator(ID) | ID of the management administrator |
| 20 | Created\_at | Datetime |  | Time to create information. |
| 21 | Altered\_at | Datetime |  | The last time since the information was edited |

## Table ProductImage

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ProductImage** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID image of product, primary key, auto-incrementing value, starting from 1 |
| 2 | Image | Image |  | Photo of the product |
| 3 | Product\_ID | Int | Not null, Foreign key: Product(ID) | ID of the product that owns the photo |

## Table CartItem

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CartItem** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID item in cart, primary key, auto-incrementing value, starting from 1 |
| 2 | Customer\_ID | Int | Not null,  Foreign key: Customer(ID) | ID of the customer |
| 3 | Product\_ID | Int | Not null,  Foreign key:  Product(ID) | ID of the product that the customer saved |
| 4 | Quantity | Int |  | Number of product in the shopping cart |

## Table Rating

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Rating** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID rating, primary key, auto-incrementing value, starting from 1 |
| 2 | Customer\_ID | Int | Not null,  Foreign key: Customer(ID) | ID of the customer |
| 3 | Product\_ID | Int | Not null,  Foreign key: Customer(ID) | ID of the product the user rated |
| 4 | RateStar | Int | Not null | Number of stars rated by users |
| 5 | Content | Ntext |  | Detailed review content |
| 6 | Created\_at | Datetime |  | Time to create information. |
| 7 | Altered\_at | Datetime |  | The last time since the information was edited |

## Table Invoice

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Invoice** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID invoice, primary key, auto-incrementing value, starting from 1 |
| 2 | Total | Int |  | Total amount to be paid |
| 3 | Discount\_Percentage | Float |  | Percentage discount |
| 4 | Customer\_ID | Int | Not null,  Foreign key: Customer(ID) | ID of the customer who owns this invoice |
| 5 | Created\_at | Datetime |  | Time to create information. |
| 6 | Altered\_at | Datetime |  | The last time since the information was edited |

## Table InvoiceDetail

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **InvoiceDetail** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID mobile, primary key, auto-incrementing value, starting from 1 |
| 2 | Invoice\_ID | Int | Not null,  Foreign key: Invoice(ID) | ID of the invoice |
| 3 | Product\_ID | Int | Not null,  Foreign key: Product(ID) | ID of the product that was purchased |
| 4 | Quantity | Int |  | Quantity of the product purchased |
| 5 | Total | Int |  | Cost of each product |

## Table PaymentDetail

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PaymentDetail** | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | ID | Int | Primary key, Increment | ID mobile, primary key, auto-incrementing value, starting from 1 |
| 2 | Amount | Int |  | Total amount paid |
| 3 | Time\_Payment | Datetime |  | Transaction time |
| 4 | Invoice\_ID | Int | Not null, Foreign key: Invoice(ID) | ID of invoice |
| 5 | Customer\_Payment\_ID | Int | Not null,  Foreign key:  CustomerPaymentMethod(ID) | ID of the payment method used by the customer |

1. **Hardware and software Requirements**

This section provides an overview of hardware and software requirements. Below are descriptions of the technological components of the Easy English Website:

|  |  |
| --- | --- |
| **Attributes of Easy English WEBSITE** | |
| **Attributes** | **Descriptions** |
| **Database** | MySQL |
| **Software** | HTML, CSS, JavaScript, Java Servelet |
| **Hardware** | Computer |
| **Library** | Java Database Connectively (JDBC) |