

**International School**

**CDIO Project**

**CMU-CS 447 LIS**

**Database Design**

**Version 1.1 Date: March 6th, 2024**

**ECHO EPIC BOOKSTORE**

**Submitted by**

**Tran Phuc Nhan**

**Huynh Anh Tuan**

**Tran Thanh Vu**

**Truong Thi Mai Thi**

**Approved by**

**MSc Huy, Truong Dinh**

**Proposal Review Panel Representative:**

Name Signature Date

## Capstone Project 2- Mentor:

Name Signature Date

**PROJECT INFORMATION**

|  |  |  |  |
| --- | --- | --- | --- |
| **Project acronym** | EEBOOK | | |
| **Project Title** | Echo Epic Bookstore | | |
| **Start Date** | 10 Jan 2024 | **End Date** | 06 Mar 2024 |
| **Lead Institution** | International School, Duy Tan University | | |
| **Project Mentor** | MSc Huy, Truong Dinh | | |
| **Scrum master / Project Leader & contact details** | Nhan, Tran Phuc  Email: [phucnhant@gmail.com](mailto:lethanhhadtu@gmail.com) Tel: 0985261764 | | |
| **Partner Organization** | Duy Tan University | | |
| **Project Web URL** |  | | |
| **Team members** | Name | Email | Tel |
|  | Nhan, Tran Phuc | [phucnhant@gmail.com](mailto:phucnhant@gmail.com) | 0985261764 |
|  | Tuan, Huynh Anh | huynhanhtuan9@dtu.edu.vn | 0369705323 |
|  | Vu, Tran Thanh | vutv579@gmail.com | 0702374029 |
|  | Thi, Truong Thi Mai | maithivip611@gmail.com | 0901527574 |

# DOCUMENT APPROVALS

The following signatures are required for approval of this document.

|  |  |  |
| --- | --- | --- |
| Nhan, Tran Phuc  Student ID: 27211242187  *Scrum Master* | Signature | Date |
| Tuan, Huynh Anh  Student ID: 27211200911  *Team Member* | Signature | Date |
| Vu, Tran Thanh  Student ID: 27211241797  *Team Member* | Signature | Date |
| Thi, Truong Thi Mai  Student ID:27201202169  *Team Member* | Signature | Date |

**REVISION HISTORY**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Comments** | **Author** | **Approval** |
| **1.0** | Nhan, Tran Phuc | 20-Feb-2024 | Create Project plan document | **1.0** |
| **1.1** | Nhan, Tran Phuc | 06-Mar-2024 | Fix Project plan document | **1.1** |

# TABLE OF CONTENT

1. [Introduction 5](#_bookmark0)
   1. [Purpose 5](#_bookmark1)
   2. [Scope 5](#_bookmark2)
   3. [Introduction about MongoDB 5](#_bookmark3)
2. [Database Diagram 6](#_bookmark4)
   1. [Table Overview 6](#_bookmark5)
   2. [Entity Relationship Diagram 7](#_bookmark6)
   3. [Table Relationship Diagram 8](#_bookmark7)
3. [Database Design for Sprint 9](#_bookmark8)
   1. [Table Group 9](#_bookmark9)
   2. [Table User 9](#_bookmark10)
   3. [Table Frame 10](#_bookmark11)
   4. [Table PrivateMessage 11](#_bookmark12)
   5. [Table PublicMessage 11](#_bookmark13)
   6. [Table Events 11](#_bookmark14)
   7. [Table Course 12](#_bookmark15)
   8. [Table CourseVocabulary 12](#_bookmark16)
   9. [Table CourseOfUsser 13](#_bookmark17)
   10. [Table Rating 13](#_bookmark18)
4. [Hardware and software Requirements 14](#_bookmark19)

## 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.

## Introduction about MySQL

MySQL is a popular open-source relational database management system (RDBMS) known for its flexibility and performance. Unlike MongoDB, which is a NoSQL database, MySQL organizes data into structured tables with rows and columns, utilizing the SQL (Structured Query Language) for querying and manipulating data.

MySQL can be deployed across various platforms such as Windows, Linux, macOS, and more, making it highly versatile and accessible. It offers features like user management, permissions, transactions, and powerful data management tools.

A Document is nothing but a data structure with name-value pairs like in JSON. It is very easy to map any custom Object of any programming language with a MySQL Document. For example: **Category** object has attributes **name, enable**, where subjects are a List.

Document for Student in MySQL will be like:

{

name : "Sách khoa học ", enable : 1,

}

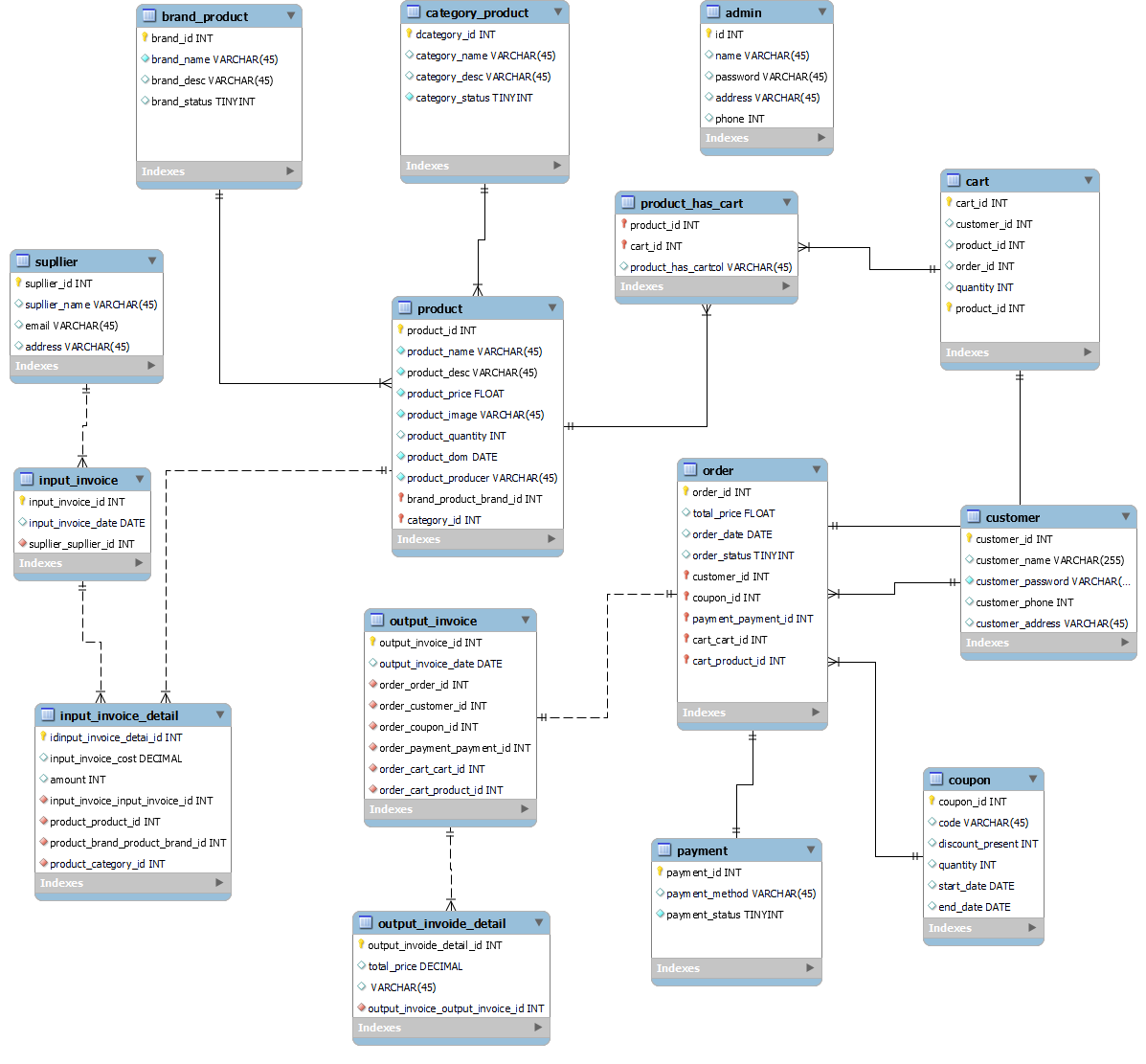
We can see, Documents are actually JSON representation of custom Objects. Also, excessive JOINS can be avoided by saving data in form of Arrays and Documents(Embedded) inside a Document.

## Database Diagram

## Table Overview

|  |  |
| --- | --- |
| **Table Name** | **Short Description** |
| Group | This table shows group information including all related fields. |
| User | This table shows member information including all related fields. |
| Frame | This table shows topic of a group. |
| PrivateMessage | This table shows messages between members. |
| PublicMessage | This table shows messages between User in the group. |
| Quizz | This table shows quiz of the group. |
| Event | This table shows all the events for the website. |

## Entity Relationship Diagram

****

## Database Design for Sprint

## Table Blog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Blog | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | Id | Int | Primary key | Blog id |
| 2 | Content | String |  | Content of blog |
| 3 | Create\_at | Date |  | Date of create blog |
| 4 | Description | String |  | Description blog |
| 5 | Title | String |  | Sex of user. |
| 6 | Image\_id | Int | Foreign key | Image id |
| 7 | User\_id | Int | Foreign key | User id |

## Table Blog\_tag

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Frame | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | blog\_id | Int | Foreign key | Blog id |
| 2 | Tag\_id | Int | Primary key | Tag id |

## Table Category

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Category | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | Id | Int | Primary key | Category id |
| 2 | Enable | Boolean |  | Enable category |
| 3 | Name | String |  | Name of category |

## Table Image

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Image | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | Id | Int | Primary Key | Image id |
| 2 | Data | Object |  | Image |
| 3 | Size | Long |  | Size of image |
| 4 | Type | String |  | Type of image |
| 5 | Uploaded\_by | String |  | Name of person who upload image |

## Table Oders

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Oders | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | Id | Int | Primary key | Oders id |
| 2 | Address | String |  | Address |
| 3 | Country | String |  | Country |
| 4 | Email | String |  | Email |
| 5 | Phone | String |  | Phone |
| 6 | Total\_price | Float |  | Price |
| 7 | User\_id | Int | Foreign key | User id |
| 8 | City | String |  | City |
| 9 | District | String |  | District |
| 10 | Name | String |  | Name |
| 11 | Phone\_number | String |  | Phone\_Number |
| 12 | Ward | String |  | Ward |
| 13 | oder\_info | String |  | Oder\_info |
| 14 | First\_name | String |  | First name |
| 15 | Last\_name | String |  | Last name |
| 16 | Note | String |  | Note |
| 17 | post\_code | String |  | Post code |
| 18 | State | String |  | State |
| 19 | Town | String |  | Town |

## Table Product

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Product | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | \_id | Int | Primary key | Product id |
| 2 | Description | String |  | Description of product |
| 3 | Name | String |  | Name of product |
| 4 | Price | Float |  | Price of product |
| 5 | Quantity | Int |  | Quantity of product |
| 6 | Category\_id | Int | Foreign key | Id category |
| 7 | Discount | String |  | Discount |
| 8 | Flash\_sale | String |  | Flash sale |

## Table User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | \_id | Int | Primary key | User id |
| 2 | Address | String |  | Address |
| 3 | Country | String |  | Country |
| 4 | Email | String |  | Email |
| 5 | Enable | Boolean |  | Enable |
| 6 | First\_name | String |  | First name |
| 7 | Last\_name | String |  | Last name |
| 8 | Password | String |  | Password |
| 9 | Phone | String |  | Phone |
| 10 | Username | String |  | Username |

## Table User role

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User role | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | role\_id | Int | Primary key | Role id |
| 2 | User\_id | Int | Foreign key | User id |

## 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** | Angular, Java |
| **Hardware** | Computer |

## 