

**International School**

**Capstone Project 2**

**CMU-SE 451 – C2SE.12**

**Database Design**

**Version 1.1**

**Date: March 15th, 2021**

**Learn English Together**

**Submitted by**

**Ha, Le Thanh**

**Hieu, Le Xuan**

**My, Ngo Ngoc**

**Thong, Doan Trung**

**Approved by**

**MSc Huy, Truong Dinh**

**Proposal Review Panel Representative:**

Name Signature Date

**Capstone Project 2- Mentor:**

Name Signature Date

**PROJECT INFORMATION**

|  |  |  |  |
| --- | --- | --- | --- |
| **Project acronym** | LET | | |
| **Project Title** | Learn English Together | | |
| **Start Date** | 26 Feb 2021 | **End Date** | 08 Jun 2021 |
| **Lead Institution** | International School, Duy Tan University | | |
| **Project Mentor** | MSc Huy, Truong Dinh | | |
| **Scrum master / Project Leader & contact details** | Ha, Le Thanh  Email: lethanhhadtu@gmail.com  Tel: 0334002818 | | |
| **Partner Organization** | Duy Tan University | | |
| **Project Web URL** |  | | |
| **Team members** | Name | Email | Tel |
|  | Ha, Le Thanh | lethanhhadtu@gmail.com | 0334002818 |
|  | Hieu, Le Xuan | xuanhieu.le.1999@gmail.com | 0399706614 |
|  | My, Ngo Ngoc | ngongocmy851999@gmail.com | 0764497391 |
|  | Thong, Doan Trung | doanthong002@gmail.com | 0886428208 |

**DOCUMENT APPROVALS**

The following signatures are required for approval of this document.

|  |  |  |
| --- | --- | --- |
| Ha, Le Thanh  Student ID: 2321122516  *Scrum Master* | Signature | Date |
| Hieu, Le Xuan  Student ID: 2321124665  *Team Member* | Signature | Date |
| My, Ngo Ngoc  Student ID: 2321124970  *Team Member* | Signature | Date |
| Thong, Doan Trung  Student ID: 2321124144  *Team Member* | Signature | Date |

**REVISION HISTORY**

| **Version** | **Date** | **Comments** | **Author** | **Approval** |
| --- | --- | --- | --- | --- |
| 1.0 | March 15th, 2021 | Initial Release | Ngoc My Ngo |  |
| 1.1 | April 20th, 2021 | Update | Ngoc My Ngo |  |

**TABLE OF CONTENT**

[1. Introduction 5](#_Toc72791784)

[**1.1 Purpose 5**](#_Toc72791785)

[**1.2 Scope 5**](#_Toc72791786)

[**1.3 Introduction about MongoDB 5**](#_Toc72791787)

[2.Database Diagram 6](#_Toc72791788)

[**2.1 Table Overview 6**](#_Toc72791789)

[**2.2 Entity Relationship Diagram 7**](#_Toc72791790)

[**2.3 Table Relationship Diagram 8**](#_Toc72791791)

[3. Database Design for Sprint 9](#_Toc72791792)

[**3.1 Table Group 9**](#_Toc72791793)

[**3.2 Table User 9**](#_Toc72791794)

[**3.3 Table Frame 10**](#_Toc72791795)

[**3.4 Table PrivateMessage 11**](#_Toc72791796)

[**3.5 Table PublicMessage 11**](#_Toc72791797)

[**3.6 Table Events 11**](#_Toc72791798)

[**3.7 Table Course 12**](#_Toc72791799)

[**3.8 Table CourseVocabulary 12**](#_Toc72791800)

[**3.9 Table CourseOfUsser 13**](#_Toc72791801)

[**3.10 Table Rating 13**](#_Toc72791802)

[4.Hardware and software Requirements 14](#_Toc72791803)

1. 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.

1.1 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.

1.2 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 MongoDB

MongoDB is a NoSQL database which stores the data in form of key-value database and can be installed across different platforms like Windows, Linux etc. pairs. It is an **Open Source**, **Document Database** which provides high performance and scalability along with data modelling and data management of huge sets of data in an enterprise application.

MongoDB also provides the feature of Auto-Scaling. Since, MongoDB is a cross platform like Windows, Linux etc.

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 MongoDB Document. For example:**Student** object has attributes **name,** **rollno** and **subjects**, where subjects are a List.

Document for Student in MongoDB will be like:

{

name : "Stduytonight",

rollno : 1,

subjects : ["C Language", "C++", "Core Java"]

}

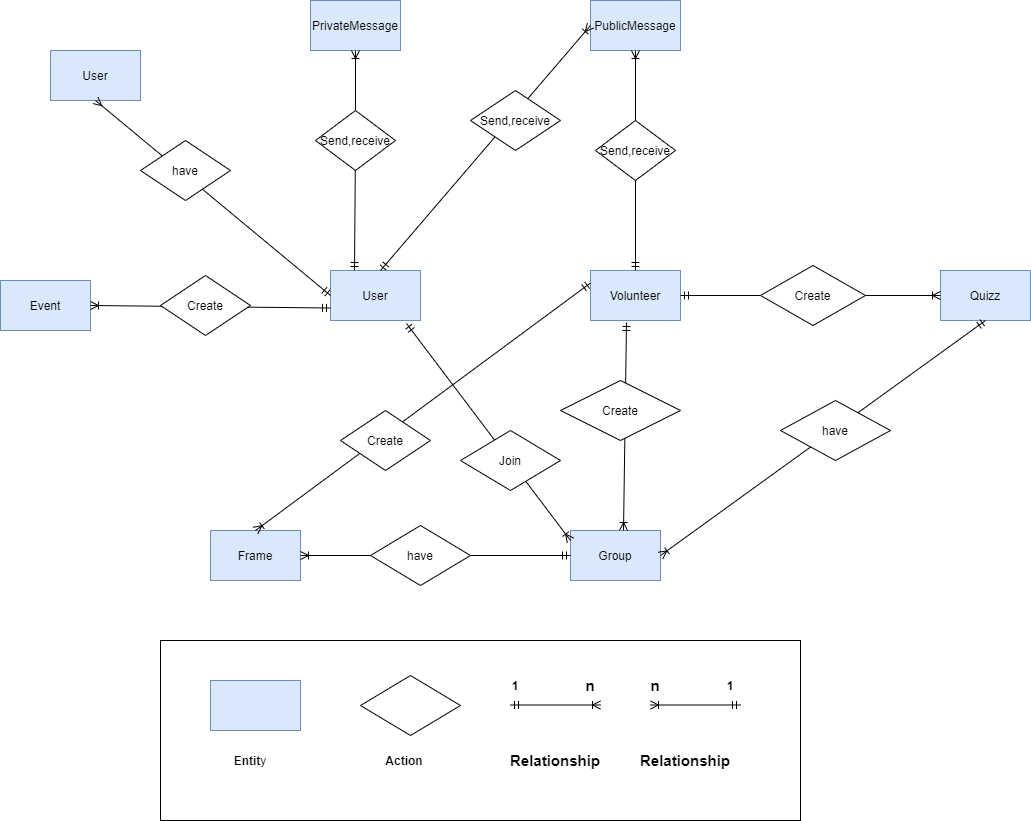
We can see, Documents are actually JSON representation of custom Objects. Also, excessive JOINS can be avoided by saving data in form of Arrays and Documents(Embedded) inside a Document.

2.Database Diagram

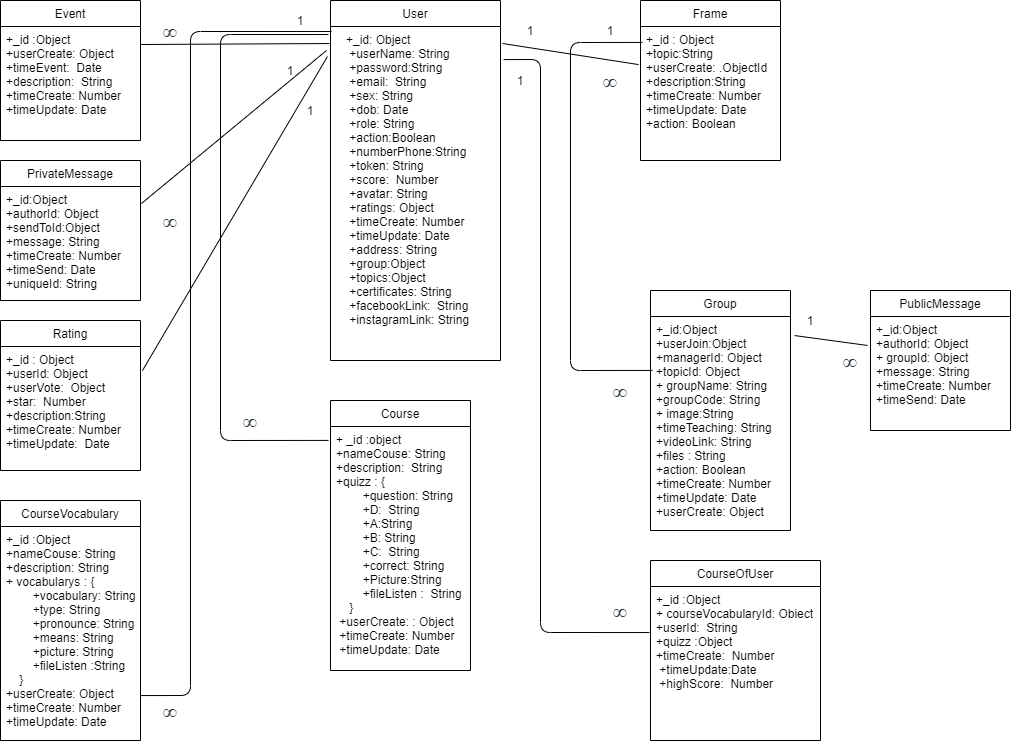
2.1 Table Overview

|  |  |
| --- | --- |
| **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. |

2.2 Entity Relationship Diagram

****

2.3 Table Relationship Diagram

****

3. Database Design for Sprint

3.1 Table Group

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Group | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | \_id | Object | Primary Key | the MongoDB driver automatically generates an ObjectId |
| 2 | userJoin | Object |  | Id user join group |
| 3 | managerId | Object |  | Id manager group |
| 4 | topicId | Object |  | Id topic group |
| 5 | groupName | String |  | Name of group |
| 6 | groupCode | String |  | Code group |
| 7 | image | String |  | Avatar of group |
| 8 | timeTeaching | date |  | Time teach |
| 9 | videoLink | String |  | Link join group when live stream |
| 10 | files | String |  | Files are uploaded to the group |
| 11 | action | Boolean |  | Action of group |
| 12 | timeCreate | Date |  | Time Create |
| 13 | timeUpdate | Date |  | Time Update |
| 14 | userCreate | Object |  | Id User Create Group |

3.2 Table User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | \_id | Object | Primary key | Id Mongodb Create |
| 2 | userName | String |  | Name of user. Display on website. |
| 3 | password | String |  | Password of user. password after saving to db will be encrypted |
| 4 | email | String |  | Connect with user and help user get their password back. Its value must be unique in the collection |
| 5 | sex | String |  | Sex of user. |
| 6 | dob | Date |  | Date of birth of user |
| 7 | role | String |  | user authorization. Three type: 1. Admin 2. Volunteer 3. Student |
| 8 | action | Boolean |  | Status of user |
| 9 | numberPhone | String |  | Phone of user |
| 10 | token | String |  | Token of user device |
| 11 | score | String |  | User's score when participating in learning on the web |
| 12 | avatar | String |  | Avatar of user. Save the path to cloud nary. Display on website. |
| 13 | ratings | Object |  | Rank of user |
| 14 | timeCreate | Date |  | User Time create |
| 15 | timeUpdate | Date |  | User Time Update |
| 16 | address | String |  | Address of user |
| 17 | group | Object |  | All group user actives |
| 18 | topics | String |  | Topics that users participate in learning |
| 19 | certificate | String |  | Certificate of user |
| 20 | facebookLink | String |  | The link to the user's personal Facebook |
| 21 | instagramLink | String |  | The link to the user's personal Instagram |

3.3 Table Frame

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Frame | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | \_id | Object | Primary key | Id Mongodb Create |
| 2 | topic | String |  | Name Topic |
| 3 | userCreate | Object |  | Id User Create |
| 4 | description | String |  | Content Topic |
| 5 | timeCreate | Number |  | Time Create |
| 6 | timeUpdate | Date |  | Time Update |
| 7 | action | Boolean |  | Status of Topic |

3.4 Table PrivateMessage

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PrivateMessage | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | \_id | Object | Primary key | Id Mongodb Create |
| 2 | authorId | Object |  | Id Sender |
| 3 | sendToId | Object |  | Id Receiver |
| 4 | message | String |  | Content Message |
| 5 | timeCreate | Number |  | Time Create |
| 6 | timeSend | Date |  | Time when message is sent |
| 7 | uniqueId | Number |  | Private Message ID |

3.5 Table PublicMessage

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PublicMessage | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | \_id | Object | Primary Key | the MongoDB driver automatically generates an ObjectId. It is ID of message. |
| 2 | authorId | Object |  | Id of Sender |
| 3 | groupID | Object |  | Id of the group that the user joins |
| 4 | message | String |  | Content of message |
| 5 | timeCreate | Number |  | Time when message is created |
| 6 | timeSend | Date |  | Time when message is sent |

3.6 Table Events

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Event | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | \_id | Object | Primary key | the MongoDB driver automatically generates an ObjectId. It is ID of event. |
| 2 | userCreate | Object |  | Event ID |
| 3 | timeEvent | Date |  | Event Time |
| 4 | description | String |  | Event Content |
| 5 | timeCreate | Nummber |  | Time when event is created |
| 6 | timeUpdate | Date |  | Time when message is updated |

3.7 Table Course

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Course | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | \_id | Object | Primary key | the MongoDB driver automatically generates an ObjectId. It is ID of course. Each course will create 1 course by id |
| 2 | nameCourse | String |  | Name of courrse |
| 3 | description | String |  | descriptions of the courses you take |
| 4 | quiz | String |  |  |
|  | question | String |  | questions of the test |
|  | D | String |  | multiple choice answer plan |
|  | A | String |  | multiple choice answer plan |
|  | B | String |  | multiple choice answer plan |
|  | C | String |  | multiple choice answer plan |
|  | correct | String |  | the correct answer to the question (A, B, C, D) |
|  | picture | String |  | simulation image of the question |
|  | fileListen | String |  | audio file of the question |
| 5 | userCreate | Object |  | Id of user create course |
| 6 | timecreate | Number |  | Time when user is created course |
| 7 | timeUpdate | Date |  | Time when user is updated course |

3.8 Table CourseVocabulary

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CourseVocabulary | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | \_id | Object | Primary key | the MongoDB driver automatically generates an ObjectId. It is ID of course. Each course will create 1 course by id |
| 2 | nameCourse | String |  | Name of course |
| 3 | description | String |  | descriptions of the courses you take |
| 4 | vocabularys | String |  |  |
|  | vocabulary | String |  | vocabulary |
|  | type | String |  | from that kind of vocabulary (v, n, adv, adj...) |
|  | pronounce | String |  | Write down the pronunciation of that word |
|  | means | String |  | meaning of the word |
|  | picture | String |  | simulation image of the vocabulary |
|  | fileListen | String |  | pronunciation of words |
| 5 | userCreate | Object |  | Id of user create course |
| 6 | timecreate | Number |  | Time when user is created course |
| 7 | timeUpdate | Date |  | Time when user is updated course |

3.9 Table CourseOfUsser

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CourseOfUsser | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | \_id | Object | Primary key | the MongoDB driver automatically generates an ObjectId. It is ID of course. |
| 2 | coursevocabularyId | Object |  | Id of course vocabulary |
| 3 | userId | Strin |  | Id of user join course |
| 4 | quizz | Object |  | Id of course vocabulary |
| 5 | timeCreate | Nummber |  | Time when user is created |
| 6 | timeUpdate | Date |  | Time when user is updated |
| 7 | highScore | Number |  | the user's score is achieved when learning vocabulary |

3.10 Table Rating

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rating | | | | |
| Id | Field | Type | Constrain | Description |
| 1 | \_id | Object | Primary key | the MongoDB driver automatically generates an ObjectId. It is ID of event. |
| 2 | userId | Object |  | id of user |
| 3 | userVote | Object |  | id of other people to vote for you |
| 4 | star | Number |  | star when you reach the highest rank in the leaderboard |
| 5 | description | String |  | Description of rating |
| 6 | timeCreate | Nummber |  | Time when event is created |
| 7 | timeUpdate | Date |  | Time when message is updated |

4.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** | MongoDB |
| **Software** | Reactjs, Nodejs, Polling, Elasticsearch, Redis, Websocket, Mail gun |
| **Hardware** | Computer |
| **Library** | Hook, Redux, React hook form, Bootstrap 4, Fontawesome, Material ui, Express.js. |