### TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN KHOA CÔNG NGHỆ THÔNG TIN

**Hệ thống quản lý, chăm sóc học viên cho trung tâm đào tạo**

## ĐỒ ÁN TỐT NGHIỆP CỬ NHÂN CNTT

**Giáo viên hướng dẫn**

**Thầy : Huỳnh Nam**

**Nhóm sinh viên thực hiện**

**Nguyễn Đại Tài - 1512470**

**Vũ Anh Tài – 1512475**

**Nguyễn Tấn Hữu Tâm – 1512483**

**Phạm Lê Thiện Tâm – 1512486**

**Trần Nhật Tâm – 1512488**

### TP. HCM,2019

# LỜI CẢM ƠN

Trước tiên, chúng em xin gửi lời cảm ơn sâu sắc đến thầy Huỳnh Nam, người đã nhiệt tình, tận tâm giúp đỡ, hướng dẫn chúng em theo cách tốt nhất có thể để hoàn thành tốt đề tài luận văn này.

Chúng em xin chân thành cảm ơn các quý Thầy / Cô trong Khoa đã tận tình giảng dạy, trang bị cho chúng em những kiến thức quý báu trong những năm học vừa qua, và hơn hết, Thầy / Cô đã tạo cơ hội và điều kiện tốt nhất để chúng em có thể thực hiện luận văn tốt nghiệp này. Đó là hành trang giúp chúng em vững bước trên con đường sự nghiệp cũng như đường đời phía trước.

Trong suốt quá trình học tập trên giảng đường Đại học, chúng em không thể tránh khỏi những thiếu sót, trở ngại. Tuy nhiên, chúng em rất biết ơn vì đã có gia đình, người thân, bạn bè, những người đã luôn ủng hộ, giúp đỡ và động viên chúng em vượt qua những khó khăn ấy.

Thời gian thực hiện luận văn đầy thử thách nhưng đem lại cho chúng em nhiều kinh nghiệm đáng quý. Chúng em đã dành hết mọi tâm huyết, nỗ lực nhưng do thời gian, khả năng có hạn nên chắc chắn không tránh khỏi thiếu sót và hạn chế. Chúng em kính mong nhận được sự cảm thông, tận tình chỉ bảo của quý Thầy Cô và các bạn giúp chúng em hoàn thiện hơn.

Một lần nữa, chúng em xin trân trọng cảm ơn và gửi lời chúc sức khoẻ đến quý Thầy Cô!

TPHCM, ngày 05 tháng 04 năm 2019

# MỤC LỤC

[LỜI CẢM ƠN](#_bookmark0)

[MỤC LỤC](#_bookmark1)

[DANH MỤC CÁC HÌNH](#_bookmark2)

[DANH MỤC CÁC BẢNG](#_bookmark3)

[TÓM TẮT ĐỒ ÁN](#_bookmark4)

[Chương 1 Giới thiệu](#_bookmark5)

* 1. [Giới thiệu tổng quan và nhu cầu hiện tại](#_bookmark6)
  2. [Mục tiêu của đề tài](#_bookmark7)

[Chương 2 Khảo sát hiện trạng](#_bookmark8)

* 1. [Khảo sát các trang web, hệ thống trung tâm đào tạo](#_bookmark9)
  2. [Đề xuất chức năng hệ thống](#_bookmark39)

2.2.1. [Những chức năng cần xây dựng](#_bookmark40)

2.2.2. [Những vấn đề cần khắc phục](#_bookmark41)

* 1. [Kết luận](#_bookmark42)

[Chương 3 Các vấn đề liên quan](#_bookmark43)

* 1. [Giới thiệu về môi trường Web](#_bookmark44)

[Website là gì ?](#_bookmark45)

[Mô hình Client-Server](#_bookmark46)

* 1. [Giới thiệu HTML5 và CSS3](#_bookmark48)

[HTML / HTML5 là gì ?](#_bookmark49)

[CSS/CSS3 là gì ?](#_bookmark50)

* 1. [Tìm hiểu NodeJS](#_bookmark51)

[NodeJS là gì ?](#_bookmark52)

[Lý do chọn NodeJS để xây dựng hệ thống](#_bookmark53)

[Các tính năng của NodeJS](#_bookmark54)

[Mô hình hoạt động của Node.JS](#_bookmark55)

[Ví dụ cách sử dụng NodeJS](#_bookmark57)

* 1. [Giới thiệu Framework Express](#_bookmark58)

[Giới thiệu sơ lược](#_bookmark59)

[Các tính năng chính của Express](#_bookmark60)

* 1. [Tìm hiểu ReactJS](#_bookmark61)

[Giới thiệu sơ lược](#_bookmark62)

[Lý do chọn ReactJS để xây dựng hệ thống](#_bookmark63)

[Các ưu điểm và nhược điểm của ReactJS](#_bookmark64)

[Ví dụ cách sử dụng ReactJS](#_bookmark65)

* 1. [Tìm hiểu MongoDB](#_bookmark66)

[Giới thiệu sơ lược](#_bookmark67)

[Lý do sử dụng hệ cơ sở dữ liệu MongoDB](#_bookmark68)

[Các đặc điểm của MongoDB](#_bookmark69)

[Ví dụ sử dụng MongoDB trong NodeJS](#_bookmark70)

* 1. [Giới thiệu Socket.IO](#_bookmark71)

[Giới thiệu sơ lược](#_bookmark72)

[Ví dụ sử dụng Socket.IO trong NodeJS](#_bookmark73)

* 1. [Quy trình đề xuất xây dựng hệ thống](#_bookmark74)

[Xây dựng hệ thống giảng dạy](#_bookmark77)

[Dữ liệu cho hệ thống giảng dạy](#_bookmark78)

* 1. [Kết luận](#_bookmark79)

[Chương 4 Giới thiệu Trung tâm đào tạo 37](#_bookmark80)

* 1. [Giới thiệu 37](#_bookmark81)

[Use-case 37](#_bookmark82)

[Các giai đoạn chính của Trung tâm đào tạo 40](#_bookmark85)

* 1. [Cấu trúc resource của ứng dụng 42](#_bookmark87)
  2. [Giao diện của Trung tâm đào tạo 43](#_bookmark89)

[Giao diện trang đăng nhập 43](#_bookmark90)

[Giao diện trang đăng ký 44](#_bookmark92)

* 1. [Kết luận 59](#_bookmark125)

[Chương 5 Phân tích hệ thống 60](#_bookmark126)

* 1. [Giới thiệu 60](#_bookmark127)
  2. [Mô tả cơ sở dữ liệu 61](#_bookmark128)

[Mô hình cơ sở dữ liệu 61](#_bookmark129)

[Mô tả chi tiết 61](#_bookmark131)

* 1. [Phân tích những vấn đề gặp phải, phương pháp và hướng giải quyết 69](#_bookmark147)

[Giáo viên upload sách mới 69](#_bookmark148)

[Số hóa sách giáo khoa 69](#_bookmark149)

[Xây dựng mô hình bản đồ tương tác 70](#_bookmark150)

[Vấn đề xử lý dữ liệu theo thời gian thực (real-time) 70](#_bookmark151)

* 1. [Kết luận 70](#_bookmark152)

[Chương 6 Kết luận và hướng phát triển 71](#_bookmark153)

* 1. [Kết quả đạt được 71](#_bookmark154)

[Về lý thuyết 71](#_bookmark155)

[Về ứng dụng 72](#_bookmark156)

[Kiểm thử ứng dụng 72](#_bookmark157)

[So sánh kết quả đạt được với các trang web đã khảo sát 73](#_bookmark162)

* 1. [Hướng phát triển của đề tài 74](#_bookmark164)

[DANH MỤC TÀI LIỆU THAM KHẢO 75](#_bookmark165)

[PHỤ LỤC 77](#_bookmark166)

# DANH MỤC CÁC HÌNH

[Hình 2-1 Dashboard 4](#_bookmark11)

# DANH MỤC CÁC BẢNG

[Bảng 2-1 Bảng tóm tắt các trang](#_bookmark22)

[Bảng 4-1 Bảng danh sách use-cases](#_bookmark84)

[Bảng 5-1 Bảng USER](#_bookmark132)

[Bảng 5-2 Bảng ATTENDANCE](#_bookmark133)

[Bảng 5-3 Bảng COURSE](#_bookmark134)

[Bảng 5-4 Bảng EXERCISE](#_bookmark135)

[Bảng 5-5 Bảng QUIZ](#_bookmark136)

[Bảng 5-6 Bảng listQuiz](#_bookmark136)

[Bảng 5-7 Bảng SubQuiz](#_bookmark136)

[Bảng 5-8 Bảng SubExercise](#_bookmark136)

[Bảng 5-9 Bảng studentQuiz](#_bookmark136)

[Bảng 5-10 Bảng studentExercise](#_bookmark136)

# TÓM TẮT ĐỒ ÁN

Ngày nay sự phát triển công nghệ nhanh chóng của các thiết bị cá nhân như điện thoại, máy tính bảng, … khiến cho cả thế giới thay đổi cách sống, cách sinh hoạt và các thói quen giải trí. Cách học cũng ảnh hưởng để bắt kịp với nhịp sống hiện tại. Sự đột phá của trung tâm online so với trung tâm truyền thống đã tạo nên một cuộc cách mạng trong những năm gần đây.

Nắm được nhu cầu đó nhiều trung tâm chạy đua phát triển các ứng dụng hỗ trợ sinh viên , nhiều nhà phát triển ứng dụng không ngừng phát triển những ứng dụng hỗ trợ sinh viên tương tác tuyệt vời hơn để đáp ứng nhu cầu ngày càng tăng của người dùng.

Đề tài luận văn này, chúng em muốn tạo ra sự khác biệt về việc giáo viên học sinh tương tác. Người dùng – đặc biệt là các em học sinh và giáo viên, những người hằng ngày phải phụ thuộc vào những phần mềm cứng nhắc, nay có thể tự điều chỉnh việc dạy học cho riêng mình, từ đó có thể nâng cao khả năng tự tìm hiểu, tự tìm tòi.

### Chương 1

**Giới thiệu**

## Giới thiệu tổng quan và nhu cầu hiện tại

Với sự phát triển của công nghệ hiện nay, chiếc máy tính bảng hay một chiếc điện thoại thông minh dần dần được nhiều người sử dụng, do nó đáp ứng được các nhu cầu của người dùng: đọc sách, lướt web, tìm kiếm thông tin, học tập hay giải trí,... Hiểu được mức độ ảnh hưởng của công nghệ đối với xã hội nói chung và dạy học nói riêng . Chính vì vậy nên phần mềm ra đời để giúp ích cho giáo viên và học sinh.

Hệ thống giáo dục trên thế giới đã phát triển, những bài học tương tác dần đã thay thế những cuốn sách giáo khoa truyền thống. Ở Việt Nam, việc dạy học vẫn đang còn phụ thuộc những bài học trong sách giáo khoa - dần dần đã trở nên rất cổ điển, và có thể là khô khan đối với học sinh. Dù vài năm qua, bộ Giáo dục Đào tạo đã có những thay đổi trong việc dạy học (mở các lớp đào tạo từ xa, các trung tâm hướng nghiệp...) nhưng vẫn gặp không ít khó khăn về tài nguyên và cơ sở vật chất. Điều này càng khẳng định, việc ứng dụng công nghệ vào giảng dạy và học tập là điều vô cùng cần thiết. Đó cũng là mục tiêu phát triển của đề tài này.

## Mục tiêu của đề tài

Thông qua việc nghiên cứu, khảo sát, nhóm đã đặt ra một số mục tiêu cần đạt được khi thực hiện đề tài của đồ án này:

* + - Khảo sát các trung tâm dạy học tìm ra thế mạnh, hạn chế của nó nhằm lập kế hoạch xây dựng hệ thống hoàn thiện hơn.
    - Khảo sát những vấn đề của học sinh trong việc học online và tự học ở nhà
    - Tìm hiểu công nghệ, các kỹ thuật, thư viện, framework hỗ trợ nhằm xây dựng, tối ưu hoá các tính năng, giúp cho ứng dụng có tính sinh động, lôi cuốn hơn và có tính giáo dục cao.
    - Khắc phục những hạn chế của các ứng dụng và hệ thống chăm sóc ,quản lý học viên cho trung tâm đào tạo càng nhiều càng tốt.
    - Xây dựng chức năng đánh giá học tập, lưu điểm, kết quả học tập của học sinh thông qua các bài tập, bài thi trực tuyến.

Nội dung khóa luận bao gồm 6 chương:

**Chương 1**: Giới thiệu khái quát về thực trạng và nhu cầu xây dựng đề tài.

**Chương 2**: Khảo sát những vấn đề của học sinh trong việc học online và tự học

**Chương 3**: Các vấn đề liên quan, môi trường phát triển, các framework cũng như công nghệ được sự dụng trong luận văn.

**Chương 4**: Giới thiệu hệ thống , trình bày sơ lược use-case và giao diện các màn hình trong hệ thống.

**Chương 5**: Phân tích hệ thống, trình bày về cách tổ chức và cấu trúc dữ liệu hệ thống.

**Chương 6**: Kết luận những gì đã đạt được trong quá trình làm đề tài và hướng phát triển của đề tài trong tương lai.

### Chương 2

**Khảo sát hiện trạng**

## Khảo sát các trang web trung tâm dạy học

## Khảo sát

## Đề xuất chức năng hệ thống

#### 2.3.1 Những chức năng cần xây dựng

* + - Liên kết các trang mạng xã hội (Google, Facebook,…) để người dùng dễ dàng theo dõi các bài học / tình hình một cách nhanh chóng.
    - Thiết kế các bài học bằng video.
    - Hệ thống bài học bằng sơ đồ tư duy giúp học sinh nắm bắt kiến thức tốt hơn.
    - Tạo kênh liên kết cho học sinh và phụ huynh, hỗ trợ chia sẻ các cách học hay và hiệu quả cho học sinh.
    - Kết hợp các trò chơi, câu đố… để học sinh vừa củng cố bài học, vừa thư giãn nhẹ nhàng.
    - Phân chia các môn học theo chủ đề, lớp.
    - Xây dựng một sơ đồ lịch sử trực quan, liền mạch qua các thời đại.

#### 2.3.2 Những vấn đề cần khắc phục

* + - Hỗ trợ giảng viên trong việc cập nhật nội dung bài học thường xuyên.
    - Bổ sung nguồn tài liệu phong phú, đa dạng (thêm hiệu ứng hình ảnh, video trong chương trình học).
    - Hỗ trợ chỉnh sửa sách theo ý người dùng.
    - Đưa hệ thống đến các em học sinh miễn phí.
    - Nâng cao hiệu ứng sinh động.

## Kết luận

### 

### Chương 3

**Các vấn đề liên quan**

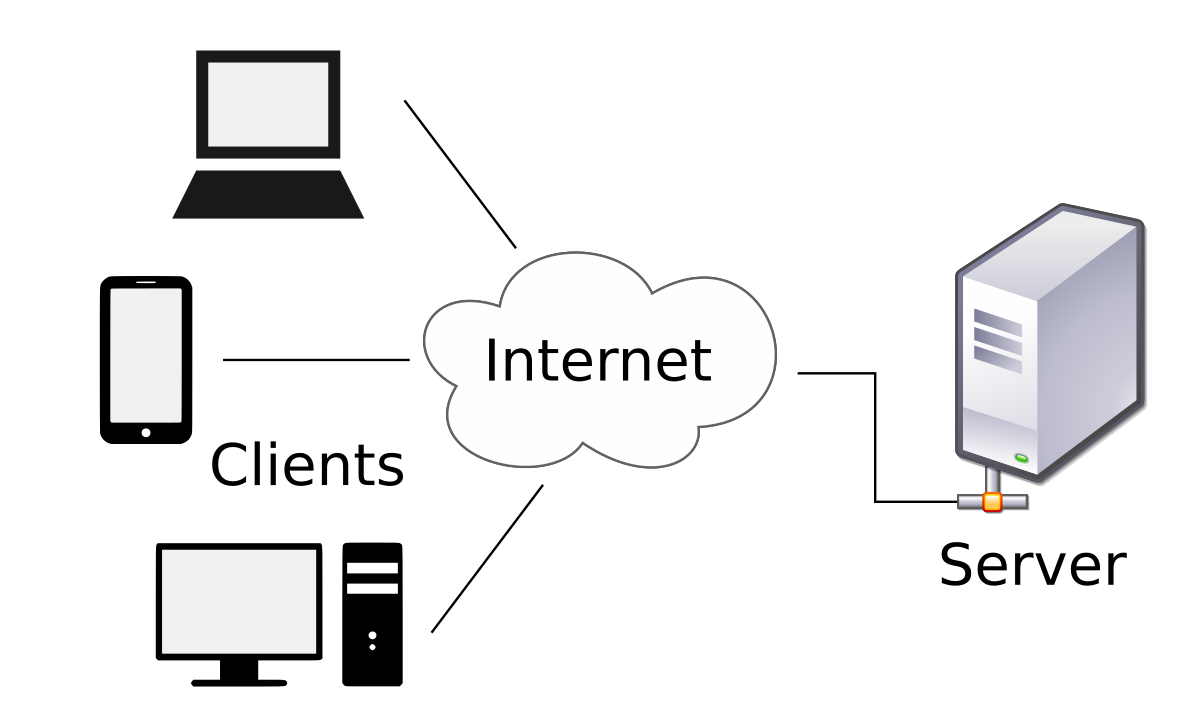
* Nội dung chương này trình bày kiến thức tổng quan về môi trường phát triển web, tìm hiểu kỹ thuật (HTML5, CSS3, Javascript, jQuery), cách sử dụng ReactJS và Framework NodeJS, Socket.io, từ đó đề xuất hướng phát triển hệ thống.*

## Giới thiệu về môi trường Web

#### Website là gì ?

Website còn gọi là trang web, trang mạng, là một tập hợp bao gồm văn bản, hình ảnh, video, flash,… thường nằm trong một tên miền. Trang web sẽ được lưu trữ trên server và người dùng có thể truy cập thông qua Internet.

#### Mô hình Client-Server

Sơ đồ bên dưới thể hiện cách thức hoạt động của mô hình Client-Server (Hình 3-1) [11]

**Hình 3-1 Sơ đồ hoạt động của mô hình Client-Server**

Người dùng sẽ sử dụng thiết bị của mình như máy tính, laptop, điện thoại,.. có kết nối đến internet, sau đó thiết bị sẽ gửi các yêu cầu tới Server. Tại đây Server sẽ xử lý tất cả các yêu cầu đó và trả lại kết quả về thiết bị của người dùng. Cuối cùng thiết bị sẽ hiển thị thông tin ra màn hình.

## Giới thiệu HTML5 và CSS3

#### HTML / HTML5 là gì ?

* **HTML** là chữ viết tắt của **H**yper**T**ext **M**arkup **L**anguage, có nghĩa là ngôn ngữ đánh dấu siêu văn bản. Nó dùng để định dạng bố cục, các thuộc tính liên quan đến cách hiển thị của một đoạn text và được hiển thị trên một chương trình đặc biệt gọi là *browser* (trình duyệt web). Hiện nay có khá nhiều browser như Firefox, Chrome, Cốc Cốc, ... Tất cả browser đều có điểm chung là giúp người dùng thao tác với website và nó đều có khả năng biên dịch những đoạn mã HTML, CSS và JavaScript.
* **HTML5** là phiên bản thứ 5 của ngôn ngữ HTML và hiện tại vẫn đang được phát triển bởi *World Wide Web Consortium* và *WHATWG*. Mục tiêu cốt lõi khi thiết kế ngôn ngữ là cải thiện khả năng hỗ trợ đa phương tiện mới nhất trong khi vẫn giữ được việc con người và các thiết bị, các chương trình máy tính như trình duyệt web, trình đọc màn hình, v.v.. có thể đọc, hiểu, hay xử lý một cách dễ dàng. HTML5 vẫn sẽ giữ lại những đặc điểm cơ bản của HTML4 và bổ sung thêm các đặc tả nổi trội của XHTML, DOM, đặc biệt là JavaScript.

#### CSS/CSS3 là gì ?

* **CSS** là chữ viết tắt của **C**ascading **S**tyle **S**heets, là một ngôn ngữ được sử dụng để tìm và định dạng lại các phần tử được tạo ra bởi các ngôn ngữ đánh dấu (ví dụ như HTML). Phương thức hoạt động của CSS là nó sẽ tìm dựa vào các vùng chọn, vùng chọn có thể là tên một thẻ HTML, tên một ID, class hay nhiều kiểu khác. Sau đó là nó sẽ áp dụng các thuộc tính cần thay đổi lên vùng chọn đó.
* **CSS3** là một chuẩn mới nhất của CSS. Nó được chia ra làm nhiều module và mỗi module sẽ có những nhiệm vụ khác nhau, khi kết hợp chúng với nhau sẽ tạo nên những hiệu ứng mới.

## Tìm hiểu NodeJS

#### NodeJS là gì ?

* + - NodeJS là một nền tảng Server-side được xây dựng dựa trên Javascript Engine (V8 Engine). Node.js được phát triển bởi Ryan Dahl năm 2009. [12]
    - Node.js là một nền tảng dựa vào Chrome Javascript runtime để xây dựng các ứng dụng nhanh, có độ lớn. Node.js sử dụng các phần phát sinh các sự kiện (event-driven), mô hình non-blocking I/O để tạo ra các ứng dụng nhẹ và hiệu quả cho các ứng dụng về dữ liệu thời gian thực chạy trên các thiết bị phân tán.
    - Node.js là một mã nguồn mở, đa nền tảng cho phát triển các ứng dụng phía Server và các ứng dụng liên quan đến mạng. Ứng dụng Node.js được viết bằng Javascript và có thể chạy trong môi trường Node.js trên hệ điều hành Window, Linux...
    - Node.js cũng cung cấp cho chúng ta các module Javascript đa dạng, có thể đơn giản hóa sự phát triển của các ứng dụng web sử dụng Node.js với các phần mở rộng.

#### Lý do chọn NodeJS để xây dựng hệ thống

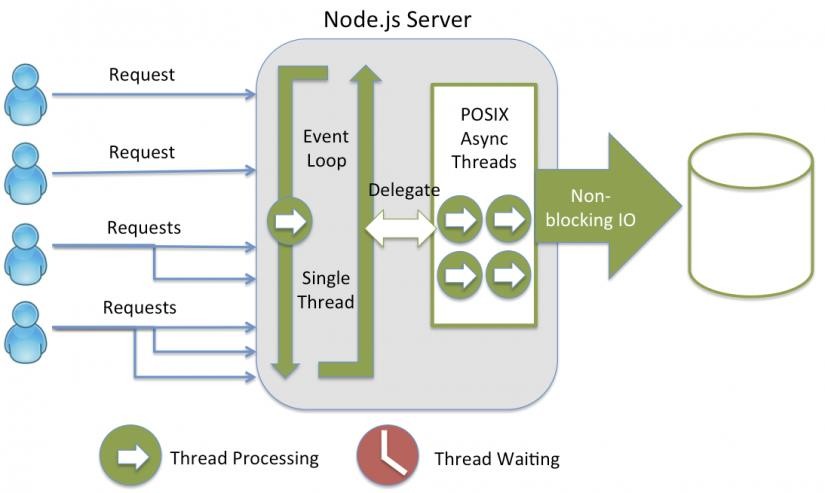
* + - Với tốc độ xử lý nhanh, NodeJS có thể xử lý rất nhiều yêu cầu của người dùng.
    - Khả năng mở rộng ứng dụng cao
    - Cộng đồng sử dụng NodeJS rất lớn

#### Các tính năng của NodeJS

* + - **Bất đồng bộ và hướng sự kiện**: Tất cả các thư viện API của Node đều bất đồng bộ, theo cơ chế non-blocking. Cơ chế hướng sự kiện sẽ giúp server xử lý kết quả trả về của API.
    - **Chạy rất nhanh**: NodeJS được xây dựng dựa vào nền tảng V8 Javascript Engine nên việc thực thi chương trình rất nhanh.
    - **Đơn luồng nhưng khả năng mở rộng cao**: Node.js sử dụng một mô hình luồng duy nhất với sự kiện lặp. Cơ chế tổ chức sự kiện giúp các máy chủ dễ đáp ứng một cách không ngăn chặn và làm cho máy chủ có khả năng mở rộng như trái ngược với các máy chủ truyền thống được tạo để hạn chế để xử lý yêu cầu. Node.js sử dụng một chương trình đơn luồng và các chương trình tương tự có thể cung cấp dịch vụ cho một số lượng lớn hơn nhiều so với yêu cầu máy chủ truyền thống như Apache HTTP Server.
    - **Không đệm**: NodeJS không đệm bất kì một dữ liệu nào và các ứng dụng này chủ yếu là đầu ra dữ liệu.

#### Mô hình hoạt động của Node.JS

NodeJS hoạt động dựa theo mô hình mô tả bên dưới (Hình 3-2) [13]



**Hình 3-2 Sơ đồ mô hình hoạt động của NodeJS**

Server Node.js sẽ luôn lắng nghe các yêu cầu từ phía Client, sau đó các yêu cầu sẽ được Node.js xử lý theo cơ chế bất đồng bộ để có thể trả về kết quả nhanh nhất cho Client.

#### Ví dụ cách sử dụng NodeJS

// Khai báo các module sẽ sử dụng cho NodeJS và tạo một server với giao thức http.

var express = require("express"); var app = express();

var server = require('http').Server(app);

// Khai báo server sẽ lắng nghe trên port 3000. server.listen(process.env.PORT || 3000, () => console.log('Server started on port 3000'))

//Tạo các phương thức ‘post’ hoặc ‘get’ ở phía server, server sẽ lắng nghe các request và trả về response cho phía client. app.get('/path, function (req, res) {

//do something;

});

app.post('/path, function (req, res) {

//do something;

});

## Giới thiệu Framework Express

#### Giới thiệu sơ lược

Express là một framework nhỏ và tiện ích để xây dựng các ứng dụng web, cung cấp một lượng lớn tính năng mạnh mẽ để phát triển các ứng dụng web và mobile. Nó rất dễ dàng để phát triển các ứng dụng nhanh dựa trên Node.js cho các ứng dụng Web. [14]

Dưới đây là các tính năng cơ bản của Express framework.

#### Các tính năng chính của Express

* Express hỗ trợ chúng ta phát triển ứng dụng theo mô hình MVC.
* Cho phép thiết lập các lớp trung gian để trả về các HTTP request.
* Định nghĩa bằng routing có thể được sử dụng với các hành động khác nhau dựa trên phương thức HTTP và URL.
* Cho phép trả về các trang HTML dựa vào các tham số truyền vào đến template.

## Tìm hiểu ReactJS

#### Giới thiệu sơ lược

ReactJS là thư viện Javascript dùng để xây dựng giao diện người dùng, được phát triển bởi Facebook và ra mắt vào năm 2013 để phục vụ các dự án hướng tới mô hình Single Page apps. ReactJS được các website phổ biến trên thế giới sử dụng ngày càng nhiều. Bên cạnh Facebook và instagram còn có các đơn vị khác đang sử dụng ReactJS cho dự án của mình như BBC, PayPal, Yahoo,…

#### Lý do chọn ReactJS để xây dựng hệ thống

* Cấu trúc đơn giản, tốc độ xử lý nhanh
* Cung cấp khả năng tạo một ứng dụng single-page
* Cộng đồng sử dụng lớn

#### Các ưu điểm và nhược điểm của ReactJS

 Ưu điểm

* **Tốc độ nhanh**: ReactJS tạo ra một DOM ảo và các components được lưu trong đó, khi có bất kỳ sự thay đổi nào ReactJS sẽ tính toán và chỉ cập nhật những components có liên quan, chính điều này làm cho tốc độ của ReactJS cực nhanh.
* **JSX (Javascript mở rộng):** ReactJS giúp việc viết các đoạn code JS dễ dàng hơn: Nó dùng cú pháp đặc biệt là JSX cho phép ta trộn giữa code HTML và Javascript. Ta có thể thêm vào các đoạn HTML vào trong hàm *render()* mà không cần phải nối chuỗi. Nó sẽ chuyển đổi các đoạn HTML thành các hàm khởi tạo đối tượng HTML bằng bộ biến đổi JSX.
* **Có nhiều công cụ phát triển**: ReactJS cho phép bạn tích hợp các công cụ phát triển ngay tại trình duyệt Chorme. Nó giúp bạn debug code dễ dàng hơn. Sau khi bạn cài đặt ứng dụng này, bạn sẽ có cái nhìn trực tiếp vào virtual DOM như thể bạn đang xem cây DOM thông thường.
* **Render ngay tại tầng server**: ReactJS có thể vừa render ở ngoài trình duyệt sử dụng DOM và cũng có thể render bằng các chuỗi HTML mà server trả về, giúp tối ưu hoá tốt hơn cho việc tối ưu công cụ tìm kiếm.

 Nhược điểm

* **ReactJS chỉ phục vụ ở tầng View**: ReactJS chỉ là View Library, nó không phải là một MVC framework như những framework khác. Đây chỉ là thư viện của Facebook giúp render ra phần view. Vì thế React sẽ không có phần Model và Controller, mà phải kết hợp với các thư viện khác. React cũng sẽ không có 2-way binding hay là Ajax.
* **ReactJS khá nặng:** Nếu so với các framework khác thì ReactJS khá nặng. Ví dụ như ReactJS có kích thước tương đương với AngularJS, tuy nhiên AngularJS là một framework hoàn chỉnh, trong khi Reactjs chỉ phục vụ ở tầng View.
* **Khó tiếp cận với người mới**: Với những đặc tính như vậy thì ReactJS khá khó cho những người mới học lập trình web.

#### Ví dụ cách sử dụng ReactJS

// Thêm vào reactjs cho phía client. import React from "react";

// Khai báo một component với các phương thức của ReactJS và hiển thị vào vị trí “id” trong html.

class NameComponent extends React.Component { constructor(props) {

// khai báo các biến state cho component

}

render(){

// trả về html

}

componentDidMount(){

// thực hiện các tác vụ sau khi render

}

componentWillMount(){

// thực hiện các tác vụ trước khi render

}

componentWillUnmount(){

// thực hiện các tác vụ khi tách component

}

componentWillRecieveProps(){

// thực hiện liên tục khi props thay đổi

}

shouldComponentUpdate(nextProps, nextState){

// thực hiện khi state hoặc props thay đổi, và trả về true/false

}

ComponentWillUpdate(nextProps, nextState){

// thực hiện khi hàm shouldComponentUpdate trả về true

}

componentDidUpdate(prevProps, prevState){

// thực hiện khi component được render lại

}

} render(<NameComponent/>,window.document.getElementById("id"));

## Tìm hiểu MongoDB

#### Giới thiệu sơ lược

#### Lý do sử dụng hệ cơ sở dữ liệu MongoDB

#### Các đặc điểm của MôngDB

#### Ví dụ sử dụng MôngDB trong NodeJS

## Giới thiệu Socket.IO

#### Giới thiệu sơ lược

Là một module của NodeJS được xây dựng nhằm mục đích tạo ra real-time NodeJS application. Socket.io cung cấp cho lập trình viên các đặc trưng như event, room và tự động phục hồi lại kết nối.

Khi client muốn kết nối tới Socket.io server, nó sẽ gửi cho server một “handshake HTTP request”. Server sẽ phân tích request đó với những thông tin cần thiết trong suốt quá trình kết nối. Nó sẽ tìm cấu hình của middleware đã được đăng ký với server và thực thi chúng trước khi đưa ra sự kiện kết nối. Khi kết nối thành công thì connection event listener được thực thi, tạo ra một instance mới của socket có thể coi như định danh của client mà mỗi một client kết nối tới sẽ có một định danh. [16]

#### Ví dụ sử dụng Socket.IO trong NodeJS

// Khai báo module của Socket.IO sử dụng trong NodeJS. var io = require('socket.io')(server);

// Khởi tạo connection của Socket.IO với các hàm thực thi. io.sockets.on('connection', function(socket){

socket.on('[tên của socket]', function(data){

// nhận dữ liệu từ biến data và thực thi các tác vụ

});

socket.emit('[tên của socket]', data) // gửi dữ liệu từ biến data

đến duy nhất client đã gọi.

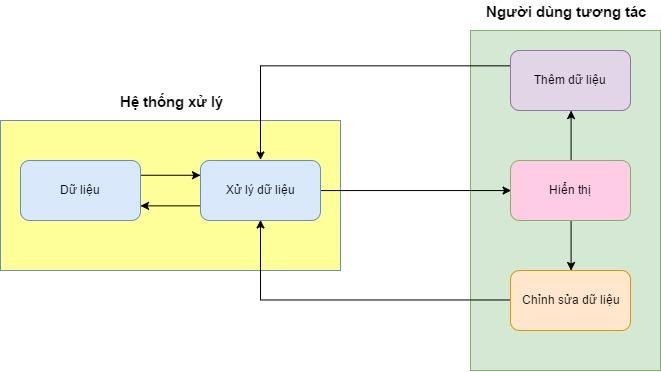
io.sockets.emit('[tên của socket]', data) // gửi dữ liệu từ biến

data đến tất cả các client đang lắng nghe. socket.broadcast.emit('[tên của socket]', data) // gửi dữ liệu từ

biến data đến tất cả các client đang lắng nghe trừ client đã gọi.

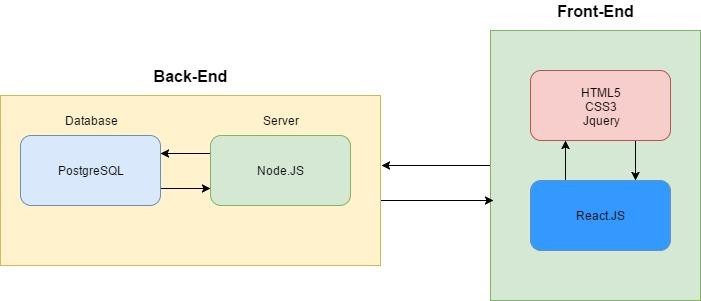
});

## Quy trình đề xuất xây dựng hệ thống

Dựa trên việc khảo sát những phần mềm có liên quan, môi trường xây dựng, các framework, nhóm đề xuất một quy trình hệ thông như sau:

**Hình 3-3 Mô hình quy trình hệ thống**

Hình 3-3 thể hiện mô hình quy trình của hệ thống. Người dùng có thể tương tác hai chiều trên hệ thống.



**Hình 3-4 Mô hình công nghệ xây dựng hệ thống**

Hình 3-4 thể hiện mô hình các công nghệ sẽ sử dụng để xây dựng nên hệ thống. Phần Front-end sẽ sử dụng ReactJS bên cạnh HTML5/CSS3 và Jquery để có thể tạo ra một hệ thống Single-page Application với mục đích chính là nâng cao trải nghiệm người dùng tốt hơn, hiệu quả hơn, thoải mái hơn. Phần Back-end sẽ sử dụng server NodeJS để có thể phản hồi nhanh, cùng với dữ liệu được lưu trữ bằng PostgreSQL.

#### Xây dựng hệ thống giảng dạy

**Vấn đề đặt ra**

Nhằm đáp ứng nhu cầu học tập của học sinh, hệ thống sẽ cung cấp các chức năng giảng dạy với nhiều hình thức khác nhau như sách giao khoa, video, sơ đồ tư duy (mindmap). Tuy nhiên theo khảo sát của chúng em điều đó vẫn chưa giải quyết được hết các khó khăn mà học sinh phải gặp trong quá trình học, và tiếp nhận các bài học.

**Giải quyết vấn đề**

Để giải quyết vấn đề trên ngoài việc chỉ cho học sinh tiếp nhận bài học một cách thụ động thì chúng em sẽ xây dựng các thành một diễn đàn, nơi mà những học sinh có thể tương tác trực tiếp với nhau để cùng giúp đỡ nhau học tập. Ngoài ra học sinh cũng có thể đánh giá các bài học của giáo viên để giúp nâng cao chất lượng của

bài học. Hệ thống cũng sẽ ghi nhận lại kết quả học tập của học sinh để học sinh có thể tự giám sát quá trình học của mình.

#### Dữ liệu cho hệ thống giảng dạy

**Vấn đề đặt ra**

Để tối ưu chất lượng giảng dạy nhầm đem đến kết quả tốt nhất cho người học thì hệ thống của chúng em cần có một cơ sở dữ liệu về bài học, ngân hàng bài tập rất lớn và độ chính xác cao, đồng thời cũng phải có khả năng điều chỉnh linh động khi có sự thay đổi.

**Giải quyết vấn đề**

Để giải quyết vấn đề trên thì hệ thống của chúng em sẽ có thêm một phân hệ người dùng là *‘Giáo viên’*. Những người này sẽ có nhiệm vụ biên soạn các tài liệu về bài học, bài tập, giải đáp các thắc mắc của học sinh và bổ sung trực tiếp vào hệ thống.

## Kết luận

Như vậy sau khi thực hiện các bước khảo sát người dùng, khảo sát các ứng dụng giảng dạy phổ biến hiện nay, chúng em đã tìm hiểu và nghiên cứu để lựa chọn các công nghệ thích hợp để xây dựng hệ thống, đồng thời đưa ra được hướng giải quyết cho các vấn đề mắc phải. Từ những thông tin có được đó nhóm quyết định đưa ra một mô hình để xây dựng nên một hệ thống giảng dạy đảm bảo được tính hệ thống, có tính liên kết và khả năng tuỳ biến cao.

### 

### Chương 4

**Giới thiệu hệ thống LMS**

## Giới thiệu

#### Use-case

**Mô hình use-case**



**Giám đốc trung tâm - ADMIN**

## C:\Users\Admin\Downloads\Copy of Untitled Diagram-Giáo vụ.jpg

**Giáo vụ**

## C:\Users\Admin\Downloads\Copy of Untitled Diagram-Cố vấn đào tạo (1).jpg

**Cố vấn đào tạo**

## C:\Users\Admin\Downloads\Copy of Untitled Diagram-Giáo viên.jpg

**Giáo viên**

## C:\Users\Admin\Downloads\Copy of Untitled Diagram-Học viên.jpg

**Học viên**

**4.1.1.2.Đạc tả use case:**

**1.Danh sách các use case:**

|  |  |  |
| --- | --- | --- |
| **STT** | **Usecase** | **Ý nghĩa** |
| 1 | Đăng ký tài khoản | Cho phép người dùng đăng ký tài khoản trên website hệ thống. |
| 2 | Thêm thông tin bài học | Người quản lý có thể thêm các bài học còn thiếu trong môn học. |
| 3 | Thêm câu hỏi trắc nghiệm | Người quản lý có thể thêm một bài kiểm tra trắc nghiệm, trước khi tạo bài tập cho buổi học. |
| 4 | Thêm khóa học | Người quản lý khóa học, tạo cột điểm, Tự tạo bài học cho từng buổi học, và các thông tin cơ bản. |
| 5 | Quản lý thông tin cá nhân | Người dùng có thể quản lý thông tin cá nhân, mật khẩu,hình ảnh. |
| 6 | Quản lý thời khóa biểu | Cho phép Chỉnh sửa, xem thời khóa biểu của khóa học, bài học |
| 7 | Tạo tài khoản, phân quyền | Tạo tài khoản cho nhân viên, và phân quyền sử dụng |
| 8 | Quản lý bài tập | Tạo bài tập cho học viên làm tại lớp, hoặc tại nhà |
| 9 | Điểm danh | Điểm danh học viên tham gia buổi học |
| 10 | Quản lý học viên | Cho xem điểm, bài học, số các buổi vắng |
| 11 | Chấm điểm | Chấm điểm các bài tập làm bằng giấy, file pdf, word, document. |
| 12 | Quản lý cột điểm | Cho phép chấm điểm, sửa điếm, xem điểm của từng học viên |
| 13 | Làm trắc nghiệm | Cho học sinh làm bài trắc nghiệm, không hủy được bài kiểm tra |
| 14 | Làm bài tập thông qua nộp file | Học viên làm bài và nộp bài tập khi làm bài xong, có thể hủy bài kiểm tra |
| 15 | Ghi danh khóa học | Cho học viên ghi danh khóa học muốn tham gia |
| 16 | Quản lý thông tin cá nhân học viên | Xem điểm, lịch học thời khóa biểu, số buổi vắng, buổi học bù. |
| 17 | Bình luận trong bài tập | Sinh viên trao đổi bên dưới bài tập |

**2.Đặc tả:**

* 1. **Usecase đăng ký tài khoản**

|  |  |
| --- | --- |
| **Usecase ID** | 1 |
| **Tên** | Đăng ký. |
| **Tóm tắt** | Use-case này cho phép người dùng đăng ký tài khoản trên website hệ thống. |
| **Tác nhân** | Học viên. |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống. |
| **Kết quả** | Đăng ký thành công. |
| **Kịch bản chính** | - Usecase này bắt đầu khi người dùng truy cập vào website hệ thống và chọn đăng ký mới.  - Hiển thị form đăng ký bao gồm có các ô tên đăng nhập, họ tên đầy đủ, ngày sinh, email, trường, lớp, mật khẩu, xác nhận mật khẩu, và nút đăng ký tài khoản hệ thống.  - Sau khi người dùng thực hiện các thao tác cần thiết khi đăng ký thì hệ thống web-server sẽ kiểm tra thông tin đăng ký tài khoản.  - Hệ thống tự động đăng nhập tài khoản vừa đăng ký và chuyển sang trang chủ website hệ thống, bắt đầu đăng nhập. |
| **Kịch bản phụ** | Đăng ký không thành công:  Lý do:  + Tên đăng nhập chứa các ký tự đặc biệt, không hợp lệ.  + Mật khẩu và xác nhận mật khẩu không trùng khớp.  + Email không đúng định dạng.  + Tên đăng nhập đã tồn tại. |
| **Ràng buộc phi chức năng** | Không có. |

Bảng 2. : Usecase đăng ký tài khoản

* 1. **Usecase thêm thông tin bài học**

|  |  |
| --- | --- |
| **Usecase ID** | 2 |
| **Tên** | Thêm thông tin bài học |
| **Tóm tắt** | Use-case này cho phép người quản lý thêm bài học. |
| **Tác nhân** | Admin, giám đốc, người cố vấn. |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống, và có tài khoản được phân quyền. |
| **Kết quả** | Tạo bài học thành công. |
| **Kịch bản chính** | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Hiển thị danh sách các bài học và nut thêm bài học.  - Nhấn vào nút thêm bài học.  - Hiện thị mẫu thêm bài học.  - Thêm thông tin bài học theo mẫu.  - Nhấn vào nút thêm bài học.  - Bài học được hiển thị vào danh sách bài học. |
| **Kịch bản phụ** | Thêm không thành công.  Lý do: bài học đã tồn tại trước đó. |
| **Ràng buộc phi chức năng** | Không có. |

Bảng 2. : Usecase them thông tin bài học

* 1. **Usecase thêm câu hỏi trắc nghiệm**

|  |  |
| --- | --- |
| **Usecase ID** | 3 |
| **Tên** | Thêm câu hỏi trắc nghiệm. |
| **Tóm tắt** | Use-case này cho phép người quản lý thêm danh sách câu hỏi trắc nghiệm. |
| **Tác nhân** | Admin, giám đốc, người cố vấn, giáo viên. |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống, và có tài khoản được phân quyền. |
| **Kết quả** | Tạo trắc nghiệm thành công. |
| **Kịch bản chính** | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Nhấn vào nút thêm câu hỏi trắc nghiệm bên thanh sidebar.  - Hiện thị mẫu thêm câu hỏi trắc nghiệm.  - Thêm thông tin bài học theo mẫu.  - Nhấn vào nút thêm câu hỏi trắc nghiệm.  - Câu hỏi trắc nghiệm được hiển thị vào danh sách trắc nghiệm.  Import file csv:  - Nhập câu hỏi theo template có sẵn.  - Chọn file CSV đã tạo.  - Nhấn nút Lưu câu hỏi. |
| **Kịch bản phụ** | Thêm không thành công.  Lý do: List câu hỏi trắc nghiệm đã tồn tại trước đó. |
| **Ràng buộc phi chức năng** | Không có. |

Bảng 2. : Usecase thêm câu hỏi trắc nghiệm

* 1. **Usecase quản lý thông tin cá nhân**

|  |  |
| --- | --- |
| **Usecase ID** | 4 |
| **Tên** | Quản lý thông tin cá nhân. |
| **Tóm tắt** | Cho phép xem, sửa thông tin cá nhân. |
| **Tác nhân** | Mợi phân hệ. |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống, và có tài khoản được phân quyền. |
| **Kết quả** | Thay đổi thông tin cá nhân thành công. |
| **Kịch bản chính** | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Hiển thị một menu lựa chọn các thao tác trong đó có thao tác vào trang thông tin cá nhân ở góc phải trên màn hình giao diện website.  - Hiển thị menu và chọn “Thay đổi thông tin”.  - Người dùng có thể xem và chỉnh sửa thông tin cá nhân, thay đổi mật khẩu tài khoản đăng nhập vào website hệ thống. |
| **Kịch bản phụ** | Không có. |
| **Ràng buộc phi chức năng** | Không có. |

Bảng 2. : Usecase quản lý thông tin cá nhân

* 1. **Usecase thêm khóa học**

|  |  |
| --- | --- |
| **Usecase ID** | 5 |
| **Tên** | Thêm Khóa học. |
| **Tóm tắt** | Use-case này cho phép người quản lý khóa học, tạo cột điểm, Tự tạo bài học cho từng buổi học, và các thông tin cơ bản. |
| **Tác nhân** | Admin, giám đốc, người cố vấn. |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống, và có tài khoản được phân quyền. |
| **Kết quả** | Tạo Khóa học thành công. |
| **Kịch bản chính** | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Thanh sidebar left chọn khóa học -> chọn thêm khóa học.  - Hiện thị mẫu thêm khóa học.  - Thêm thông tin khóa học theo mẫu.  - Nhấn vào nút thêm khóa học.  - Khóa học được hiển thị vào danh sách khóa học. |
| **Kịch bản phụ** | Thêm không thành công.  Lý do: Khóa học đã tồn tại trước đó. |
| **Ràng buộc phi chức năng** | Không có. |

Bảng 2. : Usecase thêm khóa học

* 1. **Usecase quản lý thời khóa biểu**

|  |  |  |
| --- | --- | --- |
| **Usecase ID** |  | 6 |
| **Tên** |  | Quản lý thời khóa biểu. |
| **Tóm tắt** |  | Cho phép Chỉnh sửa, xem thời khóa biểu của khóa học, bài học. |
| **Tác nhân** |  | Admin, giám đốc, người cố vấn, giáo viên. |
| **Điều kiện tiên quyết** |  | Người dùng phải truy cập vào website hệ thống, và có tài khoản được phân quyền. |
| **Kết quả** |  | Thay đổi thời khóa biểu thành công. |
| **Kịch bản chính** |  | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Thanh sidebar left chọn “Thời khóa biểu” -> chọn “Chỉnh sửa thời khóa biểu”.  - Chọn ngày thay đổi.  - chọn “lưu thời khóa biểu”.  - Hiển thị lại thời khóa biểu khi được chỉnh sửa. |
| **Kịch bản phụ** |  | Lưu thất bại:  Ngày của buổi học 1 > buổi học 2. |
| **Ràng buộc phi chức năng** |  | Không có |

Bảng 2. : Usecase quản lý thời khóa biểu

* 1. **Usecase tạo tài khoản và phân quyền**

|  |  |
| --- | --- |
| **Usecase ID** | 7 |
| **Tên** | Tạo tài khoản, phân quyền |
| **Tóm tắt** | Tạo tài khoản cho nhân viên, và phân quyền sử dụng |
| **Tác nhân** | Admin, giám đốc, người cố vấn. |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống, và có tài khoản được phân quyền. |
| **Kết quả** | Tạo tài khoản thành công. |
| **Kịch bản chính** | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Thanh sidebar left chọn “Tạo tài khoản”.  - Hiện thị mẫu thêm tài khoản  - Thêm thông tin tài khoản theo mẫu.  - Chọn “Tạo tài khoản”. |
| **Kịch bản phụ** | Lưu thất bại:  Tài khoản đã tồn tại |
| **Ràng buộc phi chức năng** | Không có |

Bảng 2. : Usecase Tạo tài khoản và phân quyền

* 1. **Usecase tạo bài tập thông qua file cho buổi học**

|  |  |
| --- | --- |
| **Usecase ID** | 8 |
| **Tên** | Tạo bài tập qua file cho buổi học. |
| **Tóm tắt** | Tạo bài tập qua file, cho từng buổi của khóa học. |
| **Tác nhân** | Admin, giám đốc, người cố vấn, giáo viên. |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống, và có tài khoản được phân quyền. |
| **Kết quả** | Tạo bài tập thành công. |
| **Kịch bản chính** | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Thanh sidebar left chọn “Khóa học” -> chọn “Danh sách khóa học” -> chọn bài học muốn thêm bài tập -> Thêm bài tập.  - Hiện thị mẫu thêm tài khoản.  - Thêm thông tin tài khoản theo mẫu.  - Chọn “Tạo bài tập”.  - Hiển thị danh sách các bài tập. |
| **Kịch bản phụ** | Không có. |
| **Ràng buộc phi chức năng** | Không có. |

Bảng 2. : Usecase tạo bài tập thông qua file

* 1. **Usecase tạo bài tập trắc nghiệm cho buổi học**

|  |  |
| --- | --- |
| **Usecase ID** | 9 |
| **Tên** | Tạo bài tập Trắc nghiệm cho buổi học. |
| **Tóm tắt** | Tạo bài tập trắc nghiệm cho từng buổi học của khóa học. |
| **Tác nhân** | Admin, giám đốc, người cố vấn, giáo viên. |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống, và có tài khoản được phân quyền. |
| **Kết quả** | Tạo bài tập thành công. |
| **Kịch bản chính** | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Thanh sidebar left chọn “Khóa học” -> chọn “Danh sách khóa học” -> chọn bài học muốn thêm bài tập trắc nghiệm -> Thêm bài tập trắc nghiệm  - Hiển thị danh sách các câu trắc nghiệm.  - Chọn câu trắc nghiệm muốn thêm.  - Chọn “Tạo bài tập”.  - Hiển thị danh sách các bài trắc nghiệm. |
| **Kịch bản phụ** | Không có. |
| **Ràng buộc phi chức năng** | Không có. |

Bảng 2. : Usecase Tạo bài tập trắc nghiệm

* 1. **Usecase điểm danh học viên**

|  |  |
| --- | --- |
| **Usecase ID** | 10 |
| **Tên** | Điểm danh. |
| **Tóm tắt** | Điểm danh học viên trong từng buổi học. |
| **Tác nhân** | Admin, giám đốc, người cố vấn, giáo viên. |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống, và có tài khoản được phân quyền. |
| **Kết quả** | Điểm danh thành công. |
| **Kịch bản chính** | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Thanh sidebar left chọn “Điểm danh”  - Chọn danh sách khóa học bên góc phải.  - Hiển thị danh sách các buổi học -> chọn buổi học điểm danh.  - Tik vô các button ở cột “điểm danh” để diểm danh.  - Chọn “Chỉnh sửa điểm danh”.  - Hiển thị danh sách các học viên với tính trạng điểm danh. |
| **Kịch bản phụ** | Không có. |
| **Ràng buộc phi chức năng** | Không có. |

Bảng 2. : Usecase điểm danh

* 1. **Usecase chấm điểm**

|  |  |
| --- | --- |
| **Usecase ID** | 11 |
| **Tên** | Chấm điểm |
| **Tóm tắt** | Usecase này cho phép người được phân quyền chấm điểm, sửa điểm từng học viên |
| **Tác nhân** | Admin, giám đốc, người cố vấn, giáo viên. |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống, và có tài khoản được phân quyền. |
| **Kết quả** | Chấm điểm thành công cho bài kiểm tra. |
| **Kịch bản chính** | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Thanh sidebar left chọn “Khóa học của tôi”.  - Chọn khóa học muốn xem thông tin -> chọn buổi học cần chấm điểm -> chọn bài tập chấm điểm.  - Chấm điểm cho từng học viện.  - Chọn “Thay đổi”  - Cập nhật điểm. |
| **Kịch bản phụ** | Không có. |
| **Ràng buộc phi chức năng** | Không có. |

Bảng 2. : Usecase Chấm điểm

* 1. **Usecase quản lý điểm**

|  |  |
| --- | --- |
| **Usecase ID** | 12 |
| **Tên** | Quản lý điểm. |
| **Tóm tắt** | Quản lý cột điểm của từng khóa học, chọn bài tập đã chọn trong từng buổi học vào từng cột điểm. |
| **Tác nhân** | Admin, giám đốc, người cố vấn, giáo viên. |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống, và có tài khoản được phân quyền. |
| **Kết quả** | Cập nhật cột điểm cho khóa học thành công. |
| **Kịch bản chính** | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Thanh sidebar left chọn “Quản lý cột điểm”.  - Chọn khóa học muốn cập nhật cột điểm -> chọn cột điểm -> chọn bài tập hoặc câu hỏi trắc nghiệm để chọn lấy điểm.  - Cập tên bài tập. |
| **Kịch bản phụ** | Không có. |
| **Ràng buộc phi chức năng** | Không có. |

Bảng 2. : Usecase quản lý điểm

* 1. **Usecase làm bài tập trắc nghiệm**

|  |  |
| --- | --- |
| **Usecase ID** | 13 |
| **Tên** | Làm bài tập trắc nghiệm |
| **Tóm tắt** | Học viên làm bài tập trắc nghiệm, thông qua website hệ thống. |
| **Tác nhân** | Học viên |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống, và có tài khoản. |
| **Kết quả** | Làm bài và cập nhật điểm vô hệ thống thành công |
| **Kịch bản chính** | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Thanh sidebar left chọn “Khóa học của tôi”.  - Chọn khóa học tham gia -> Chọn buổi học cần làm bài tập -> Chọn bài tập trắc nghiệm chưa làm -> Bắt đầu làm bài.  Hiển thị điểm số và quay lại màn hình bài học. |
| **Kịch bản phụ** | Bài học trắc nghiệm đã làm:  Hiển thị điểm bài tập đã làm. |
| **Ràng buộc phi chức năng** | Không có. |

Bảng 2. : Usecase làm bài tập trắc nghiệm

* 1. **Usecase làm bài tập nộp file**

|  |  |
| --- | --- |
| **Usecase ID** | 14 |
| **Tên** | Làm bài tập qua nộp file |
| **Tóm tắt** | Học viên làm bài tập và nộp file đính kèm, thông qua website hệ thống. |
| **Tác nhân** | Học viên |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống, và có tài khoản. |
| **Kết quả** | Nộp bài thành công |
| **Kịch bản chính** | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Thanh sidebar left chọn “Khóa học của tôi”.  - Chọn khóa học tham gia -> Chọn buổi học cần làm bài tập.  - Chọn file bài tập cần nộp.  - Chọn “Nộp”.  Hiển thị nộp bài thành công. |
| **Kịch bản phụ** | Bài tập đã có bài nộp:  - Chọn hủy file.  - Chọn lại file mới cần nộp.  - Chọn “Nộp”.  - Hiển thị nộp bài thành công. |
| **Ràng buộc phi chức năng** | Không có. |

Bảng 2. : Usecase làm bài tập nộp file

* 1. **Usecase ghi danh**

|  |  |
| --- | --- |
| **Usecase ID** | 15 |
| **Tên** | Ghi danh |
| **Tóm tắt** | Học viên muốn tham gia vào khóa học |
| **Tác nhân** | Học viên |
| **Điều kiện tiên quyết** | Người dùng phải truy cập vào website hệ thống, và có tài khoản. |
| **Kết quả** | Nộp bài thành công |
| **Kịch bản chính** | - Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Thanh sidebar left chọn “Xem khóa học hiện có”.  - Chọn “Ghi danh”.  - Cập nhật trên hệ thống.  - Hiển thị ghi danh. |
| **Kịch bản phụ** | Hủy ghi danh:  Use-case này bắt đầu khi người dùng truy cập vào website hệ thống và đã đăng nhập thành công với tài khoản tác nhân.  - Thanh sidebar left chọn “Xem khóa học hiện có”.  - Chọn “Hủy ghi danh” (admin chưa phê duyệt).  - Cập nhật trên hệ thống.  - Hiển thị hủy thành công. |
| **Ràng buộc phi chức năng** | Không có. |

Bảng 2. : Usecase ghi danh

## 4.2.Cấu trúc resource của ứng dụng:

## C:\Users\Admin\Downloads\Untitled Diagram (1).jpg

**Hình ảnh cấu trúc resource của ứng dụng**

## 4.3.Giao diện của ứng dụng

## A.Phiên bản Web:

## 1.Giao diện trang đăng nhập:

## 

Hình ảnh giao diện trang đăng nhập

## 2.Giao diện trang chủ:

## 

Hình ảnh màn hình trang chủ

## 3.Giao diện trang Danh sách bài học:

## 

Hình ảnh giao diện trang Danh sách bài học

## 4.Giao diện trang Thêm khoá học:

## 

Hình ảnh giao diện trang thêm khoá học - 1

## 

Hình ảnh giao diện trang thêm khoá học - 2

## 

Hình ảnh giao diện trang thêm khoá học - 3

### 

## 5.Giao diện trang Khoá học của tôi:

## 

Hình ảnh giao diện trang Khoá học của tôi

## 6.Giao diện trang Nội dung khoá học:

### 

Hình ảnh giao diện trang Nội dung khoá học

## 7.Giao diện trang Nội dung khoá học chi tiết:

## 

Hình ảnh giao diện trang Nội dung khoá học chi tiết - 1

### 

Hình ảnh giao diện trang Nội dung khoá học chi tiết - 2

## 8.Giao diện trang Danh sách các khoá học (Admin):

### 

Hình ảnh giao diện trang Danh sách các khoá học (Admin)

## 9.Giao diện trang Quản lý khoá học(Admin,Giáo vụ):

Hình ảnh giao diện trang Quản lý khoá học

## 10.Giao diện trang Chỉnh sửa khoá học(Admin,Giáo vụ): chỉnh sửa các mục tương tự như thêm khoá học

Hình ảnh giao diện trang chỉnh sửa khoá học

### 11.Giao diện trang Điểm danh học viên(Admin,Giáo vụ,Giáo viên):

Hình ảnh giao diện trang điểm danh học viên-1

### 

Hình ảnh giao diện trang Điểm danh học viên – 2 (Điểm danh cho từng ngày)

### 12.Giao diện trang Xem lịch sử điểm danh:

### 

Hình ảnh giao diện trang Xem lịch sử điểm danh

### 

Hình ảnh giao diện trang Xem lịch sử điểm danh (cho từng buổi học)

### 13.Giao diện trang Sửa thời khoá biểu(Admin,Giáo vụ,Giáo viên):

### 

Hình ảnh giao diện trang Sửa thời khoá biểu - 1

### 

Hình ảnh giao diện trang Sửa thời khoá biểu - 2

### Chương 5

**Phân tích hệ thống**

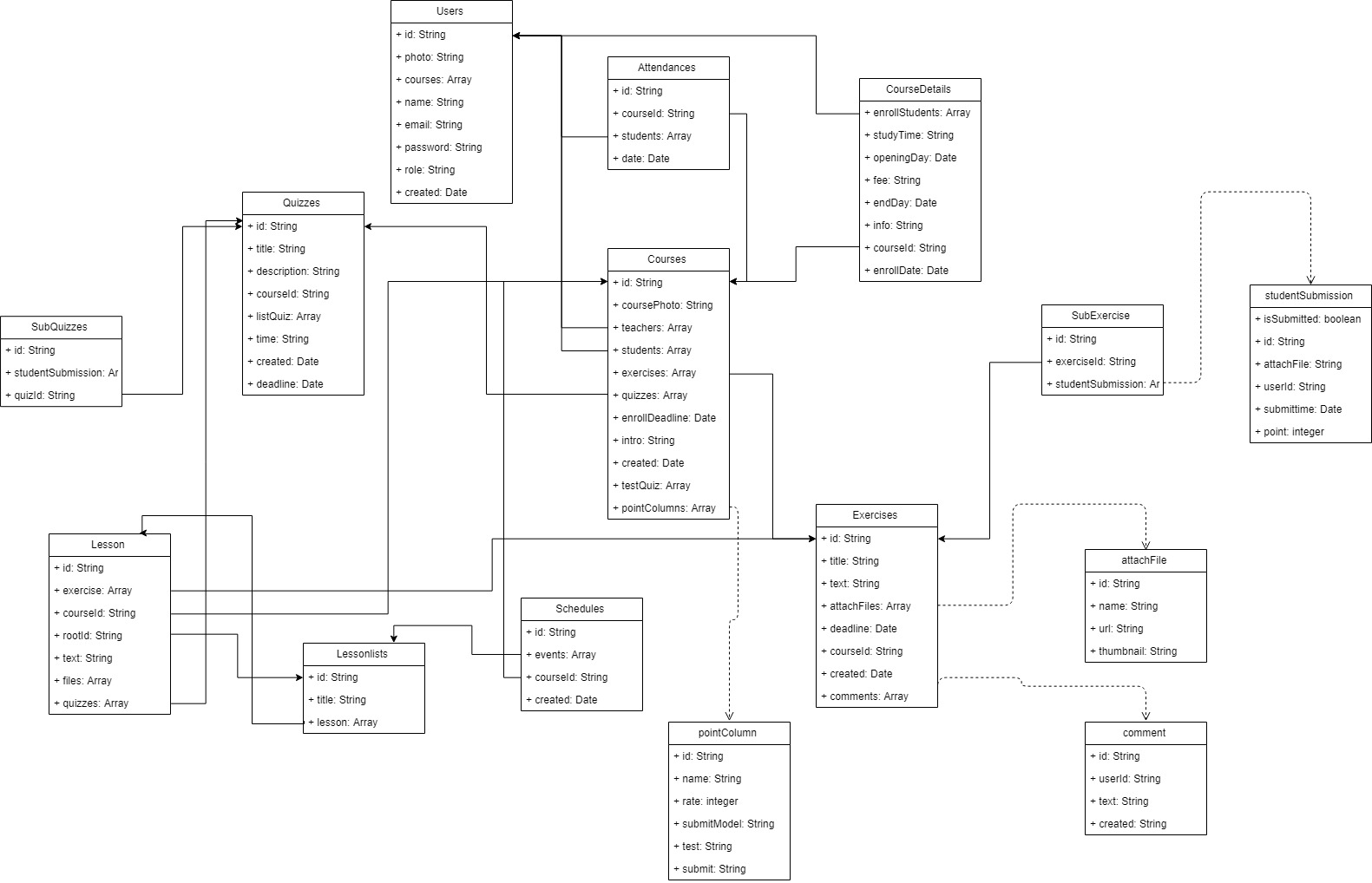
## Giới thiệu

Bất cứ hệ thống nào đang hoạt động hiện nay đều có cơ sở dữ liệu. Tổ chức, thiết kế và tổ chức dữ liệu sao cho phù hợp với mục đích phát triển của hệ thống chính là vấn đề cần được giải quyết trước khi thiết kế hệ thống.

Đối với đề tài này, chúng em sử dụng cấu trúc MongoDB để thiết kế cơ sở dữ liệu (như đã trình bày ở phần [3.6](#_bookmark66)).

## Mô tả cơ sở dữ liệu

#### Mô hình cơ sở dữ liệu

******

**Hình 5-1 Mô hình cơ sở dữ liệu**

Hình 5-1 thể hiện sơ đồ liên kết giữa các bảng trong cơ sở dữ liệu.

#### Mô tả chi tiết

* **User**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolID | String | Mã số User |
| Photo | String | Đường dẫn tới hình ảnh đại diện của user |
| Courses | Array | Danh sách các khóa học đã tham gia |
| Name | String | Tên user |
| Email | String | Email user |
| Password | String | Password đăng nhập |
| Role | String | Vai trò của user để phân quyền |
| Created | Date | Ngày tạo user |

* **Attendances**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolID | String | Mã số attendance |
| courseId | String | Khóa ngoại với course |
| student | Array | Danh sách học sinh |
| Date | Date | Ngày ghi danh |

* **CourseDetails**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| enrollStudents | Array | Danh sách các học sinh ghi danh |
| studyTime | String | thời gian học |
| openingDay | Date | Ngày khai giảng |
| fee | String | Học phí |
| endDay | Date | Ngày tổng kết |
| info | String | Thông tin |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolcourseId | String | Mã course |
| enrollDate | Date | Ngày mở ghi danh |

* **Courses**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolId | String | Mã số course |
| coursePhoto | String | đường dẫn tới ảnh đại diện khóa học |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolteacher | Array | Danh sách teacher (khóa ngoại) |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolstudent | Array | Danh sách student (khóa ngoại) |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolexercise | Array | Danh sách exercise (khóa ngoại) |
| enrollDeadline | Date | ngày mở ghi danh |
| intro | String | Thông tin khóa học |
| created | Date | Ngày tạo |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symboltestQuiz | Array | Danh sách quiz (khóa ngoại) |
| pointcolumns | Array | danh sách cột điểm |

* **Exercise**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolID | String | Mã số Exercise |
| Title | String | Tiêu đề bài tập |
| text | String | Mô tả bài tập |
| attachFile | Array | Danh sách file đính kèm |
| deadline | Date | Ngày nộp |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolcourseId | String | Danh sách course (khóa ngoại) |
| createdDate | Date | ngày tạo |
| comments | Array | danh sách các bình luận |

* **SubExercise**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolId | String | Mã số SubExercise |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolexerciseId | String | Mã số Exercise (khóa ngoại) |
| studentExercise | Array | danh sách bài tập |
| studentSubmission | Array | danh sách bài nộp của học sinh |

* **Quizzes**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolID | String | Mã số Quizzes |
| title | String | Tiêu đề của quiz |
| description | String | Mô tả quiz |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolcourseId | String | Mã số course (khóa ngoại) |
| listQuiz | Array | Danh sách quiz |
| time | String | thời gian làm quiz |
| created | Date | Ngày tạo quiz |
| deadline | Date | thời hạn nộp |

* **SubQuizzes**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolID | String | Mã số SubQuizzes |
| studentSubmission | Array | Danh sách bài nộp của học sinh |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolQuizId | String | Mã số Quizzes (Khóa ngoại) |

* **Schedule**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolID | String | Mã số Schedule |
| events | Array | danh sách các bài học diễn ra |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolcourseId | String | Mã số course (khóa ngoại) |
| created | Date | Ngày tạo schedule |

* **LessonList**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolId | String | Mã số LessonList |
| title | String | Tiêu đề bài học |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbollesson | Array | Danh sách các bài học (khóa ngoại) |

* **Lesson**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolID | String | Mã số Lesson |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolexercise | Array | Danh sách exercise (khóa ngoại) |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolcourseId | String | Danh sách course (khóa ngoại) |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolrootId | String | Danh sách lessonLists (khóa ngoại) |
| text | String | Mô tả Lesson |
| files | Array | Danh sách các file bài học |
| Káº¿t quáº£ hÃ¬nh áº£nh cho key symbolquizzes | Array | Danh sách quiz (khóa ngoại) |

## Kết luận

Trong chương này, chúng em đã giới thiệu về cấu trúc của cơ sở dữ liệu hệ thống, mô tả cơ sở dữ liệu, cách thức xây dựng, cách thức hoạt động. Với những vấn đề và khó khăn gặp phải, nhóm đã đề ra những giải pháp khắc phục nhằm giúp cho hệ thống chạy ổn định hơn, tốt hơn và nhanh hơn. Bên cạnh đó, kiến trúc được xây dựng một cách linh hoạt, mang tính mở rộng và phát triển cao.

### Chương 6

**Kết luận và hướng phát triển**

## Kết quả đạt được

Thông qua quá trình nghiên cứu, tìm hiểu và khảo sát các kỹ thuật, vận dụng vào luận văn, nhóm đã xây dựng được một hệ thống giúp học sinh thuận tiện trong quá trình dạy và học, cũng như giúp giảng viên biên soạn giáo trình dễ dàng, hiệu quả hơn.

Sau quá trình này kết quả chúng em đạt được như sau:

#### Về lý thuyết

* Rèn luyện kỹ năng lập trình CSS3, HTML5, Javascript và NodeJS, cũng như áp dụng những framework, công nghệ có sẵn để xây dựng hệ thống (reactJS, socket.io).
* Cải thiện kỹ năng làm việc nhóm, phương pháp phân chia công việc và tổng hợp kết quả hiệu quả hơn.
* Nắm rõ thêm về kiến trúc của Client-Server và kiến trúc web, cách quản lý và xây dựng ứng dụng.
* Dù không thể áp dụng hết được tất cả các công nghệ, nhưng từ quá trình khảo sát và nghiên cứu chúng em đã hiểu thêm về ưu và khuyết điểm để đưa ra quyết định thích hợp nhằm giải quyết được vấn đề đặt ra.
* Học hỏi được các kỹ năng xử lý vấn đề khi hệ thống bị lỗi.
* Rèn luyện kỹ năng khảo sát những phần mềm có liên quan, phân tích vấn đề và đưa ra hướng giải quyết.

#### Về ứng dụng

* Xây dựng được một website cho phép người dùng miễn phí.
* Trang web chính thức của nhóm :
* Hỗ trợ người dùng tương tác với nhau thông qua diễn đàn thảo luận, có hệ thống làm bài thi, kiểm tra định kỳ và đánh giá kết quả học tập.

#### Kiểm thử ứng dụng

**Cấu hình phần cứng máy**

|  |  |  |  |
| --- | --- | --- | --- |
| CPU | RAM | HDD | Architecture |
| Intel Core i7, 2.5 GHz | 8 GB | 1 TB | 64 bit |

**Bảng 6-1 Bảng mô tả cấu hình phần cứng máy test**

**Cấu hình phần mềm**

|  |  |  |
| --- | --- | --- |
| Tên phần mềm | Phiên bản | Loại |
| Microsoft Windows 10 Education | 10 | Hệ điều hành |
| Mozilla Firefox | 49.0.2 | Trình duyệt web |
| Google Chrome | 60.0.3112.90 | Trình duyệt web |
| Microsoft Edge | 40.15063.0.0 | Trình duyệt web |

**Bảng 6-2 Bảng mô tả cấu hình phần mềm test**

**Kiểm thử khả năng tương thích (Compatibility testing)**

* + Kịch bản kiểm thử : Tất cả các trang của ứng dụng.
  + Kết quả kiểm thử :

|  |  |  |
| --- | --- | --- |
| Trình duyệt web | Kết quả mong đợi | Kết quả thực tế |
| Mozilla Firefox | Tất cả các trang hoạt động được. | Hoạt động được. |
| Google Chrome | Tất cả các trang hoạt động được. | Hoạt động được. |
| Microsoft Edge | Tất cả các trang hoạt động được. | Hoạt động được. |

**Bảng 6-3 Bảng kết quả kiểm thử tương thích**

**Load testing**

* + Kịch bản kiểm thử : Tất cả các trang của ứng dụng.
  + Kết quả kiểm thử :

|  |  |  |
| --- | --- | --- |
| Số người dùng | Kết quả mong đợi | Kết quả thực tế |
| 10 | 100% đăng nhập thành công. | 100% đăng nhập thành công. |
| 1000 | 100% đăng nhập thành công. | 100% đăng nhập thành công. |
| 10000 | 100% đăng nhập thành công. | 72% đăng nhập thành công. |

**Bảng 6-4 Bảng kết quả load testing**

#### So sánh kết quả đạt được với các trang web đã khảo sát

## Hướng phát triển của đề tài

Do sự phát triển không ngừng của các ứng dụng, cùng với nhu cầu ngày một tăng của người dùng, nhóm dự định sẽ cập nhật thêm những tính năng mới trong tương lai:

* + - Thực hiện phát triển hệ thống trên nhiều nền tảng khác nhau: Android, iOS, BlackBerry, …
    - Hỗ trợ thêm cho người sử dụng nhiều loại hiệu ứng hơn để đáp ứng nhu cầu sáng tạo cần thiết (hiệu ứng bôi đen, highlight và lưu sách theo từng cá nhân)
    - Tạo trang để giáo viên upload tài liệu mới (phần tóm tắt kiến thức).
    - Hỗ trợ phần kiểm tra tự luận cho hệ thống.
    - Hỗ trợ xây dựng mô hình tương tác 3DDANH MỤC TÀI LIỆU THAM

# TÀI LIỆU THAM KHẢO

# PHỤ LỤC

## Hướng dẫn thao tác sử dụng phần mềm LMS