

# BỘ CÔNG THƯƠNG

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**------------------------**

****

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC NGÀNH CÔNG NGHỆ THÔNG TIN

XÂY DỰNG HỆ THỐNG HỖ TRỢ TỰ HỌC VÀ ĐÁNH GIÁ TRỰC TUYẾN

GVHD : TS. Đặng Trọng Hợp

Sinh viên : Đinh Văn Duy Mã số sinh viên : 2021604611

Hà Nội - Năm 2025

i

# MỤC LỤC

DANH MỤC TỪ VIẾT TẮT.................................................................................

v

DANH MỤC HÌNH ẢNH......................................................................................

vi

DANH MỤC BẢNG BIỂU....................................................................................

viii

v

**DANH MỤC TỪ VIẾT TẮT**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Từ viết tắt** | **Tiếng Anh** | **Nghĩa tiếng việt** |
| 1 | API |  |  |

**DANH MỤC HÌNH ẢNH**

vi

Hình 1.1 :API

**DANH MỤC BẢNG BIỂU**

Bảng 1.1: 10

# MỞ ĐẦU

### Lý do chọn đề tài

Hiện tại là thời kỳ chuyển đổi số rất mạnh mẽ, giáo dục trực tuyến là một phần không thể thiếu trong việc nâng cao chất lượng đào tạo, chất lượng học tập dành cho các sinh viên. Các trường đại học cần triển khai mạnh mẽ các hệ thống hỗ trợ học trực tuyến cho sinh viên. Tuy nhiên hầu hết các hệ thống chỉ hỗ trợ các bài tập dạng trắc nghiệm hoặc tự luận, chưa có tính năng thực hành lập trình trực tuyến trên nền tảng. Điều này gây khó khăn cho các sinh viên trong việc học tập các môn có liên quan tới lập trình, do phải sử dụng phần mềm (IDE) bên ngoài để viết mã nguồn.

Nhắm vào những yếu tố đó, em muốn phát triển mạnh mẽ một tính năng có thể trực tiếp lập trình trên chính trang website của trường đại học Công Nghiệp Hà Nội của mình. Chính vì vậy đề tài “Hệ thống hỗ trợ tự học và đánh giá trực tuyến” được thực hiện nhằm xây dựng một nền tảng tự học trực tuyến có tích hợp tính năng lập trình trực tuyến trên website.

### Mục tiêu nghiên cứu

* + Xây dựng nền tảng học tập trực tuyến, cho phép sinh viên truy cập các học phần, bài học lý thuyết và thực hành lập trình.
  + Tích hợp trình soạn thảo code trực tuyến, hỗ trợ nhiều ngôn ngữ lập trình.
  + Cho phép sinh viên viết, chạy thử và nộp bài tập ngay trên nền tảng.
  + Tự động chấm điểm bài tập lập trình.
  + Hỗ trợ giảng viên trong quản lý và đánh giá sinh viên.
  + Ứng dụng công nghệ hiện đại để phát triển hệ thống: Docker, ASP.NET Core Web API, ReactJs,..

### Đối tượng nghiên cứu

* + Website học kết hợp của trường đại học Công Nghiệp Hà Nội.
  + Sinh viên, giảng viên.
  + Mô hình học tập kết hợp.

### Phạm vi nghiên cứu

* + Sinh viên: Sử dụng hệ thống để học lý thuyết, thực hành lập trình trực tuyến và nộp bài tập.
  + Giảng viên: Quản lý bài học, tạo bài tập lập trình, theo dõi và đánh giá kết quả của sinh viên.
  + Hệ thống chấm điểm tự động: Xử lý bài làm của sinh viên, kiểm tra tính đúng đắn của mã nguồn dựa trên test case, thời gian xử lý,..
  + Lập trình trực tuyến: Hỗ trợ sinh viên viết code, chạy thử và nộp bài ngay trên trình duyệt mà không cần cài đặt phần mềm bên ngoài.
  + Học trực tuyến: Sinh viên có thể truy cập học phần, bài giảng lý thuyết và tài liệu tham khảo.

## Nội dung nghiên cứu

* + Nghiên cứu mô hình học kết hợp của trường Đại học Công Nghiệp Hà Nội.
  + Phân tích và thiết kế xây dựng theo các học phần.
  + Xác định các yêu cầu chức năng và phi chức năng của website.
  + Đề xuất mô hình hệ thống phù hợp với nhu cầu của sinh viên và giảng viên.
  + Thiết kế cơ sở dữ liệu để lưu trữ thông tin học phần, bài giảng, bài tập và kết quả nộp bài.
  + Xây dựng API trên nền tảng ASP.NET Core Web API.
  + Phát triển Frontend bằng ReactJS, tối ưu hiệu suất.
  + Tích hợp trình soạn thảo code online hỗ trợ nhiều ngôn ngữ lập trình để sinh viên có thể chọn ngôn ngữ tùy thích.

### Bố cục bản báo cáo

Báo cáo gồm các phần sau đây:

# 

# CHƯƠNG 1. TỔNG QUAN HỆ THỐNG HỖ TRỢ TỰ HỌC VÀ ĐÁNH GIÁ TRỰC TUYẾN

## Giới thiệu về hệ thống hỗ trợ tự học và đánh giá trực tuyến.

Trong thời đại công nghệ số phát triển mạnh mẽ, việc ứng dụng công nghệ thông tin vào giáo dục ngày càng trở nên phổ biến và cần thiết. Đặc biệt là trong bối cảnh dịch bệnh Covid-19 vừa qua, hình thức học tập trực tuyến đã dần thay thế cho phương thức học truyền thống tại nhiều cơ sở giáo dục. Tuy nhiên, các hệ thống học trực tuyến hiện nay đa số chỉ hỗ trợ trình bày nội dung bài học mà thiếu đi sự tương tác thực hành, đặc biệt là đối với các môn học lập trình, đòi hỏi người học phải thường xuyên luyện tập và nhận được phản hồi tức thì.

Trước nhu cầu đó, đề tài ***"Xây dựng hệ thống hỗ trợ tự học và đánh giá trực tuyến"*** ra đời với mục tiêu xây dựng một nền tảng học trực tuyến tích hợp chức năng lập trình online và hệ thống chấm điểm tự động. Hệ thống không chỉ giúp sinh viên có thể tiếp cận nội dung bài học ở bất kỳ đâu, mà còn cho phép họ thực hành lập trình trực tiếp trên trình duyệt, nộp bài và nhận kết quả đánh giá ngay lập tức.

Với mô hình tổ chức theo từng học phần, mỗi học phần gồm nhiều bài học lý thuyết đi kèm bài tập thực hành, hệ thống này hỗ trợ giảng viên dễ dàng tạo nội dung giảng dạy và bài kiểm tra, đồng thời theo dõi tiến độ học tập của sinh viên. Qua đó, hệ thống góp phần nâng cao hiệu quả tự học và tạo môi trường học tập chủ động, linh hoạt cho người học.

## Mô tả bài toán

Bài toán đặt ra là xây dựng một hệ thống học tập và đánh giá trực tuyến, đặc biệt dành cho các môn học lập trình, cho phép người học không chỉ tiếp cận nội dung bài giảng mà còn có thể thực hành và được chấm điểm một cách tự động. Cụ thể, hệ thống cần đáp ứng các yêu cầu sau:

Tổ chức học phần và bài học: Các môn học (học phần) được chia thành các bài học, mỗi bài học bao gồm phần lý thuyết (tài liệu, video, văn bản) và bài tập lập trình đi kèm.

Soạn thảo và thực hành lập trình trực tuyến: Hệ thống tích hợp trình soạn thảo code (code editor) cho phép sinh viên viết, chạy thử và nộp bài trực tiếp trên nền tảng, không cần cài đặt phần mềm bên ngoài.

Chấm điểm tự động: Bài làm của sinh viên được kiểm tra bằng hệ thống test case, trả về kết quả đúng/sai, điểm số, thời gian thực thi và số lượng test case vượt qua.

**Quản lý người dùng:**

Sinh viên: Có thể đăng ký, đăng nhập, tham gia học phần, làm bài tập và theo dõi kết quả học tập.

Giảng viên: Có thể tạo học phần, bài học, bài tập, theo dõi tiến độ sinh viên và xem thống kê kết quả học tập.

Hệ thống quản trị: Quản trị viên có quyền quản lý học phần, người dùng, bài học, bài tập, kết quả và các báo cáo thống kê hệ thống.

**Yêu cầu về kỹ thuật:**

Backend sử dụng ASP.NET Core Web API để cung cấp dữ liệu và xử lý nghiệp vụ.

Frontend sử dụng ReactJS để xây dựng giao diện tương tác hiện đại, dễ sử dụng.

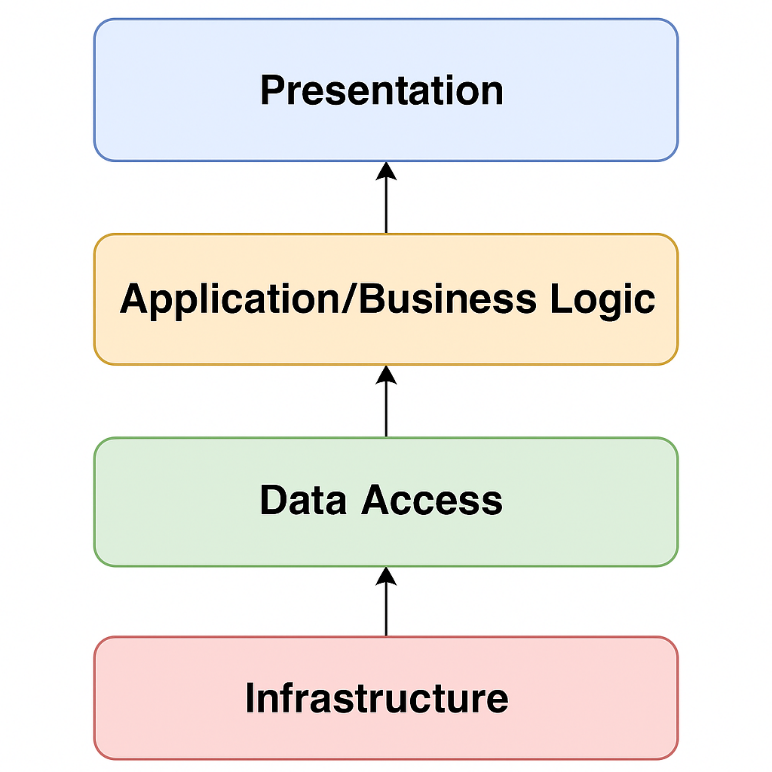
Cơ sở dữ liệu SQL Server để lưu trữ toàn bộ thông tin người dùng, học phần, bài tập, kết quả.

Tích hợp Docker + Judge để xử lý và đánh giá bài nộp một cách an toàn, cách ly môi trường thực thi.

Mục tiêu của hệ thống là xây dựng một nền tảng học tập toàn diện, trực quan, dễ sử dụng, đồng thời nâng cao tính chủ động của sinh viên trong việc học và thực hành lập trình, tạo ra môi trường học tập thực tế và hiệu quả hơn trong giáo dục đại học.

## Mô hình và công nghệ Web

## Kiến trúc phân lớp (Layered Architecture)

Kiến trúc phân lớp là một mô hình tổ chức phần mềm phổ biến, trong đó hệ thống được chia thành nhiều lớp riêng biệt, mỗi lớp đảm nhiệm một vai trò cụ thể trong quy trình xử lý dữ liệu và nghiệp vụ. Việc chia nhỏ hệ thống thành các lớp giúp tăng tính mô-đun, dễ bảo trì, mở rộng và kiểm thử phần mềm.

Trong đề tài này, hệ thống được xây dựng theo kiến trúc phân lớp gồm các lớp chính sau:

**Lớp Presentation (Giao diện người dùng)**

Là phần giao tiếp giữa người dùng và hệ thống.

Trong đề tài, lớp này được phát triển bằng ReactJS, hiển thị các giao diện học tập, bài học, bài tập và cho phép sinh viên tương tác như gửi code, xem kết quả, theo dõi tiến độ.

**Lớp Business Logic (Xử lý nghiệp vụ)**

Chịu trách nhiệm xử lý các logic nghiệp vụ như tính điểm, kiểm tra kết quả bài làm, xử lý các yêu cầu từ frontend.

Lớp này được tổ chức thành các service trong ASP.NET Core, đảm bảo rằng mọi thao tác từ giao diện đều đi qua lớp xử lý nghiệp vụ.

**Lớp Data Access (Truy cập dữ liệu)**

Chịu trách nhiệm giao tiếp với cơ sở dữ liệu, thực hiện các thao tác thêm, sửa, xóa, truy vấn dữ liệu.

Lớp này sử dụng mô hình Repository và UnitOfWork để tách biệt logic truy cập dữ liệu khỏi logic xử lý nghiệp vụ, đồng thời tăng khả năng tái sử dụng và kiểm thử.

**Lớp Infrastructure (Hạ tầng)**

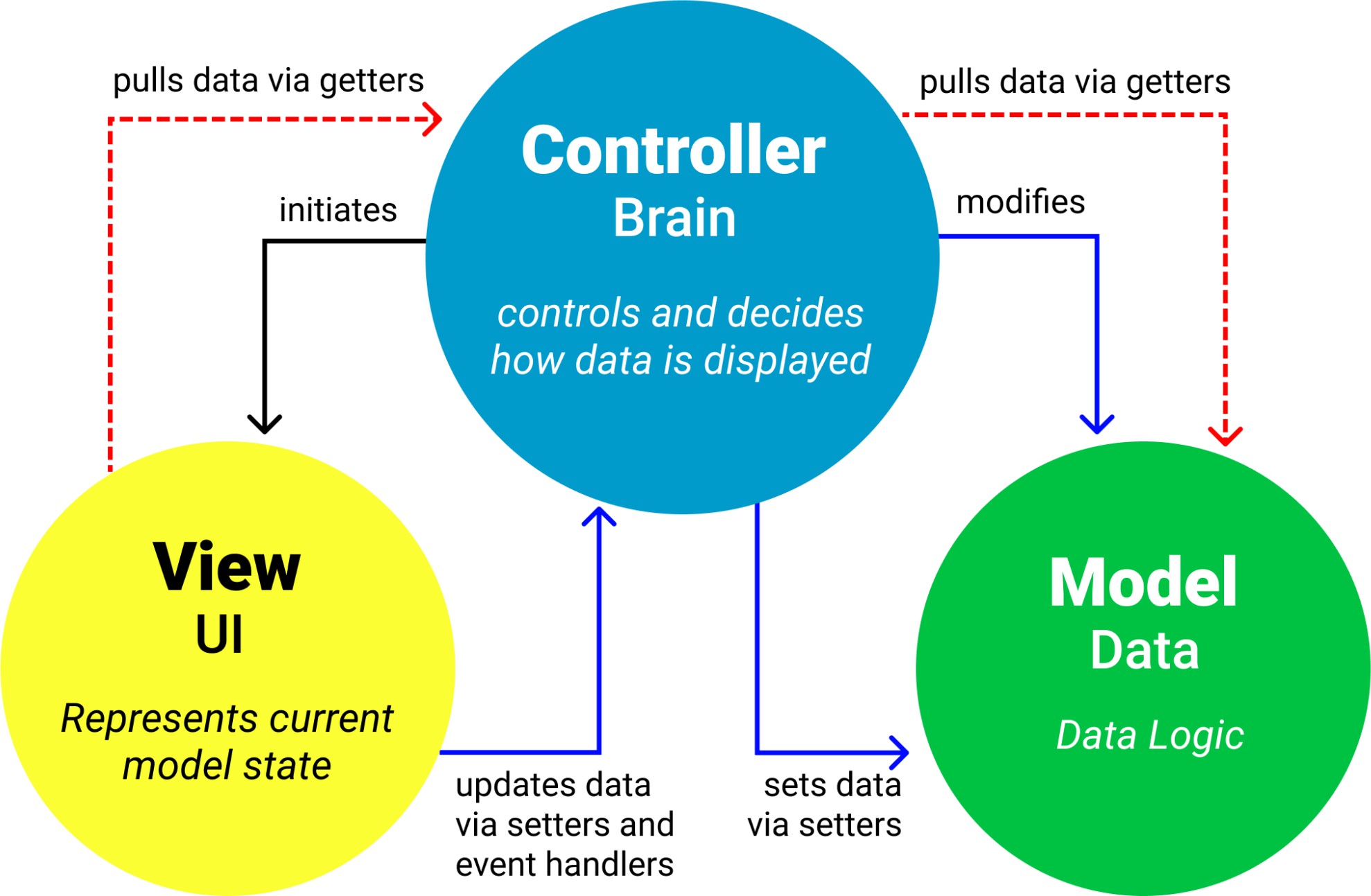
Bao gồm các thành phần hỗ trợ tích hợp với các dịch vụ bên ngoài như hệ thống chấm bài tự động (Judge API).

Giúp hệ thống mở rộng linh hoạt và dễ dàng tích hợp với các công nghệ khác trong tương lai.

Việc áp dụng kiến trúc phân lớp giúp hệ thống đảm bảo được tính rõ ràng về cấu trúc, dễ dàng bảo trì, mở rộng và tái sử dụng mã nguồn. Đồng thời giúp nhóm phát triển phân chia công việc hiệu quả hơn giữa các thành viên frontend và backend.

## Mô hình kiến trúc MVC

MVC [2] (Model-View-Controller) là một mô hình kiến trúc phần mềm được sử dụng để tổ chức và quản lý mã nguồn trong quá trình phát triển ứng dụng. Kiến trúc này tách biệt ứng dụng thành ba thành phần chính có thể tương tác được với nhau để tách biệt giữa cách thức mà thông tin được xử lý nội hàm và phần thông tin được trình bày và tiếp nhận từ phía người dùng.



*Hình 1.1 : Kiến trúc phần mềm Model-View-Controller*

* Model: là nơi lưu trữ thông tin, thực hiện các thao tác cập nhật và truy vấn dữ liệu mà không quan tâm đến cách dữ liệu được hiển thị hoặc tương tác với người dùng.
* View: là thành phần hiển thị giao diện người dùng. View chịu trách nhiệm hiển thị dữ liệu một cách đẹp mắt và dễ thao tác đối với người dùng.
* Controller: là thành phần tiếp nhận và xử lý sự kiện từ người dùng. Controller nhận yêu cầu từ người dùng thông qua View, sau đó thao tác với Model và điều hướng hiển thị trở lại View.

## Ngôn ngữ lập trình C# và ASP.NET

C# là một ngôn ngữ lập trình đa năng, được phát triển bởi Microsoft, thuộc dòng ngôn ngữ lập trình .NET. C# được sử dụng phổ biến trong phát triển phần mềm, đặc biệt là trong việc xây dựng các ứng dụng web với ASP.NET. ASP.NET là một framework phát triển ứng dụng web mạnh mẽ, được Microsoft phát triển, cho phép tạo ra các ứng dụng web, API, và các dịch vụ web hiệu quả.

Một số ưu điểm khiến ngôn ngữ lập trình C# và ASP.NET trở nên phổ biến:

* Hiệu suất cao và bảo mật tốt: C# và ASP.NET cung cấp hiệu suất vượt trội nhờ vào việc biên dịch mã nguồn sang mã máy (C#) và tối ưu hóa quá trình xử lý dữ liệu trong ASP.NET. Ngoài ra, ASP.NET còn tích hợp các tính năng bảo mật mạnh mẽ giúp bảo vệ các ứng dụng web khỏi các cuộc tấn công như SQL Injection, XSS, CSRF, v.v.
* Mã nguồn mở và miễn phí: C# và ASP.NET Core là mã nguồn mở, miễn phí và có thể sử dụng trên nhiều nền tảng như Windows, Linux và macOS. Điều này giúp tiết kiệm chi phí phát triển và triển khai ứng dụng.
* Tính linh hoạt: C# có thể được sử dụng để phát triển ứng dụng trên nhiều nền tảng, từ các ứng dụng máy tính để bàn, di động đến các ứng dụng web phức tạp. ASP.NET hỗ trợ phát triển ứng dụng web đa dạng từ các trang web động đến các API RESTful, WebSocket, và hơn thế nữa.
* Cộng đồng và thư viện hỗ trợ: C# và ASP.NET có một cộng đồng phát triển mạnh mẽ và số lượng thư viện hỗ trợ phong phú, giúp các nhà phát triển dễ dàng giải quyết các bài toán phức tạp và tối ưu hóa quá trình phát triển ứng dụng.
* Tính tương thích với các hệ quản trị cơ sở dữ liệu: ASP.NET có thể kết nối với nhiều hệ quản trị cơ sở dữ liệu như SQL Server, MySQL, PostgreSQL, SQLite và Oracle, giúp xây dựng các ứng dụng với cơ sở dữ liệu linh hoạt và mạnh mẽ.

Mặc dù có nhiều ưu điểm, C# và ASP.NET cũng có một số hạn chế, đặc biệt là về khối lượng tài nguyên hệ thống khi sử dụng cho các ứng dụng quy mô lớn. Tuy nhiên, những vấn đề này có thể được giải quyết nhờ vào các công cụ tối ưu hóa và cấu hình đúng đắn trong quá trình phát triển.

## Cơ sở dữ liệu MS SQL

MS SQL (Microsoft SQL Server) là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) được phát triển bởi Microsoft, sử dụng ngôn ngữ truy vấn có cấu trúc (SQL). MS SQL quản lý dữ liệu hiệu quả nhờ việc tổ chức dữ liệu dưới dạng các bảng và mối quan hệ giữa các bảng, giúp giảm thiểu sự lặp lại dữ liệu, tăng tính rõ ràng và hiệu quả của dữ liệu.

**Ưu điểm của MS SQL:**

Tốc độ cao, ổn định, dễ sử dụng: MS SQL là cơ sở dữ liệu tốc độ cao, ổn định, dễ sử dụng và hoạt động trên các hệ điều hành Windows, cung cấp nhiều tính năng tiện ích mạnh mẽ.  
Độ bảo mật cao: MS SQL rất thích hợp cho các ứng dụng có truy cập CSDL trên Internet, với các tính năng bảo mật mạnh mẽ như mã hóa, kiểm tra quyền truy cập và chính sách bảo mật nâng cao.  
Khả năng mở rộng: MS SQL hỗ trợ quy mô dữ liệu lớn và có khả năng mở rộng linh hoạt, cho phép xử lý các yêu cầu phức tạp và dữ liệu lớn trong môi trường doanh nghiệp.

**Nhược điểm của MS SQL:**

Chi phí cao: MS SQL yêu cầu giấy phép sử dụng, điều này có thể gây tốn kém cho các tổ chức nhỏ hoặc các dự án cần tiết kiệm chi phí.  
Yêu cầu tài nguyên hệ thống lớn: MS SQL yêu cầu phần cứng và tài nguyên hệ thống mạnh mẽ, đặc biệt là khi xử lý các cơ sở dữ liệu lớn hoặc các tác vụ phức tạp.

# Ngôn ngữ lập trình ReactJs

ReactJS là một thư viện JavaScript được phát triển bởi Facebook, chuyên dùng để xây dựng giao diện người dùng (UI) cho các ứng dụng web. React cho phép phát triển ứng dụng đơn trang (SPA) với khả năng tái sử dụng các thành phần giao diện, tăng hiệu quả phát triển và dễ bảo trì. ReactJS nổi bật với tính năng Virtual DOM, giúp tối ưu hóa hiệu suất của ứng dụng bằng cách giảm thiểu số lần cập nhật giao diện thực sự trên trình duyệt.

**Ưu điểm của ReactJS:**

Dễ sử dụng và học hỏi: ReactJS có cú pháp đơn giản, dễ hiểu và rất dễ tiếp cận đối với lập trình viên JavaScript.  
Tái sử dụng component: ReactJS cho phép tái sử dụng các component giao diện, giúp giảm thiểu sự trùng lặp mã nguồn và tăng tính mô-đun cho ứng dụng.  
Tốc độ và hiệu suất cao: Nhờ vào Virtual DOM, ReactJS có khả năng cập nhật giao diện nhanh chóng và tối ưu hóa hiệu suất của ứng dụng web.  
Cộng đồng mạnh mẽ: ReactJS có một cộng đồng phát triển lớn mạnh, với rất nhiều tài liệu, plugin và thư viện hỗ trợ.

**Nhược điểm của ReactJS:**

SEO khó khăn: Mặc dù có thể cải thiện SEO với các kỹ thuật như server-side rendering (SSR), nhưng ReactJS không tối ưu cho việc tối ưu hóa công cụ tìm kiếm (SEO) mặc định.  
Tài liệu chưa đầy đủ: Một số phần của ReactJS có tài liệu chưa đầy đủ hoặc khó hiểu, đòi hỏi người học phải tự nghiên cứu hoặc tìm kiếm từ cộng đồng.  
Cập nhật nhanh chóng: Với sự phát triển nhanh chóng của ReactJS, việc cập nhật và làm quen với các thay đổi mới trong thư viện có thể gây khó khăn cho lập trình viên.

## Kết luận chương 1

Chương 1 đã trình bày tổng quan về hệ thống hỗ trợ tự học và đánh giá trực tuyến, nhu cầu xây dựng hệ thống thi lập trình trực tuyến cho các trường đại học. Từ đó, đề tài nghiên cứu được xác định với mục tiêu phát triển một hệ thống cho phép sinh viên lập trình trực tiếp trên web, gửi bài và chấm điểm tự động bằng test case. Để đáp ứng các yêu cầu này, các công nghệ phù hợp được lựa chọn là ngôn ngữ lập trình ASP.NET Core, ReactJS, hệ quản trị CSDL MS SQL, và sử dụng Docker để triển khai hệ thống.