

### NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

This image shows a full page of white paper with horizontal dotted lines, typical of primary school writing paper. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Trà Vinh, ngày ..... tháng ..... năm .....

**Giáo viên hướng dẫn**  
(Ký tên và ghi rõ họ tên)

Phan Thị Phương Nam

## NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG

[illegible]

Trà Vinh, ngày ..... tháng ..... năm .....

**Thành viên hội đồng**  
(Ký tên và ghi rõ họ tên)

## LỜI CẢM ƠN

Trước hết, em xin gửi lời cảm ơn chân thành và sâu sắc đến cô Phan Thị Phương Nam, giảng viên hướng dẫn, người đã luôn tận tình chỉ bảo, hướng dẫn và hỗ trợ em trong suốt quá trình thực hiện đồ án chuyên ngành này. Sự tận tâm, kiên nhẫn và những kiến thức chuyên môn quý báu của cô đã giúp em vượt qua những khó khăn, thách thức trong quá trình nghiên cứu và thực hiện đồ án. Em vô cùng biết ơn vì sự hỗ trợ và chỉ dẫn đầy nhiệt huyết của cô.

Bên cạnh đó, em cũng muốn gửi lời cảm ơn sâu sắc đến các bạn bè đã luôn đồng hành cùng em trong suốt quá trình học tập và nghiên cứu. Những buổi thảo luận, chia sẻ và hỗ trợ của các bạn đã giúp em rất nhiều trong việc hoàn thiện đồ án này. Sự gần gũi và tình bạn quý báu của các bạn đã làm cho em có động lực hoàn thành đồ án một cách thật tốt và hoàn chỉnh nhất có thể.

Cuối cùng, em xin gửi lời cảm ơn đến gia đình và người thân đã luôn bên cạnh, động viên và khích lệ em trong suốt quá trình học tập. Sự yêu thương, chăm sóc và sự hy sinh của gia đình đã là nguồn động lực lớn lao giúp em vượt qua mọi khó khăn, hoàn thành tốt công việc học tập và nghiên cứu. Em biết ơn gia đình vì đã luôn là chỗ dựa vững chắc, tiếp thêm sức mạnh cho em trên con đường học vấn.

Một lần nữa, em xin chân thành cảm ơn tất cả những ai đã giúp đỡ, hỗ trợ và đồng hành cùng em trong suốt quá trình thực hiện đồ án chuyên ngành này.

---

## MỤC LỤC

DANH MỤC BẢNG BIỂU.....	6
TÓM TẮT ĐỒ ÁN CHUYÊN NGÀNH.....	7
MỞ ĐẦU .....	8
CHƯƠNG 1: TỔNG QUAN .....	11
1.1. Giới thiệu về vấn đề nghiên cứu:.....	11
1.2. Mục tiêu và ý nghĩa của nghiên cứu:.....	11
1.3. Tình hình nghiên cứu hiện nay .....	13
1.4. Các vấn đề cần giải quyết.....	14
1.5. Phạm vi nghiên cứu .....	15
CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT .....	16
2.1. Cơ sở lý thuyết.....	16
2.1.1 Kỹ năng đọc hiểu trong học tiếng Anh.....	16
2.1.2 Dạng bài True/False/Not Given trong thi cử:.....	16
2.1.3 Cơ sở dữ liệu NoSQL .....	17
2.1.4 Công nghệ MERN (MongoDB, Express.js, React.js, Node.js) .....	23
2.1.5 Hệ quản trị cơ sở dữ liệu MongoDB .....	23
2.2. Phương pháp nghiên cứu .....	28
2.3. Tổng kết.....	29
CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU .....	30
3.1. Các bước tiến hành: .....	30
3.2. Phân tích hệ thống .....	31
3.3. Kiến trúc hệ thống .....	31
CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU .....	35
4.1. Kết quả Đạt được.....	35
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	42
5.1. Kết luận: .....	42
5.2. Hướng phát triển.....	42
DANH MỤC TÀI LIỆU THAM KHẢO .....	44
PHỤ LỤC .....	45

## **DANH MỤC HÌNH ẢNH**

HÌNH 1: CÁC LOẠI CƠ SỞ DỮ LIỆU NOSQL .....	18
HÌNH 2: LOẠI CƠ SỞ DỮ LIỆU DẠNG DOCUMENT – BASED .....	18
HÌNH 3: VÍ DỤ CƠ SỞ DỮ LIỆU DẠNG GRAPH-BASED.....	19
HÌNH 4: CƠ SỞ DỮ LIỆU DẠNG KEYS - VALUES .....	20
HÌNH 5: CÁCH MONGODB HOẠT ĐỘNG .....	28
HÌNH 6: BIỂU DIỄN CÁC COLLECTIONS .....	32
HÌNH 7: GIAO DIỆN TRANG CHỦ .....	35
HÌNH 8: GIAO DIỆN TRANG ĐĂNG KÝ .....	36
HÌNH 9: GIAO DIỆN TRANG ĐĂNG NHẬP .....	36
HÌNH 10: GIAO DIỆN TRANG DANH SÁCH BÀI THI.....	37
HÌNH 11: GIAO DIỆN TRANG LÀM BÀI THI .....	38
HÌNH 12: GIAO DIỆN CHẤM ĐIỂM .....	40
HÌNH 13: GIAO TRANG TRANG ADMIN.....	41

## **DANH MỤC BẢNG BIỂU**

BẢNG 1: SO SÁNH SQL SERVER VÀ MONGODB .....	22
BẢNG 2: CÁC KIỂU DỮ LIỆU TRONG MONGODB .....	25
BẢNG 3: CÁC CÂU LỆNH TRONG MONGODB .....	26

## TÓM TẮT ĐỒ ÁN CHUYÊN NGÀNH

### Tóm tắt vấn đề cần nghiên cứu:

Đề tài nghiên cứu của em tập trung vào việc phát triển một trang web thi trắc nghiệm dạng đọc hiểu (reading True/False/Not Given) dành cho người học tiếng Anh. Với nhu cầu ngày càng cao trong việc cải thiện kỹ năng đọc hiểu cho người học tiếng Anh, em quyết định xây dựng một ứng dụng trực tuyến giúp người dùng có thể luyện tập và kiểm tra khả năng đọc hiểu thông qua các bài thi trắc nghiệm.

### Các hướng tiếp cận:

Thiết kế và phát triển sản phẩm: Xây dựng giao diện trang web dễ sử dụng và thân thiện với người dùng. Sử dụng công nghệ web hiện đại như MERN(MongoDB, Express, React.js, Node.js) để phát triển giao diện và xây dựng hệ thống quản lý bài thi, sử dụng API để lấy câu hỏi được lưu trữ từ MongoDB, lưu câu hỏi và chấm điểm.

Cách giải quyết vấn đề:

- Phân tích yêu cầu: Xác định các tính năng cần thiết cho trang web, bao gồm quản lý tài khoản người dùng, tạo bài thi, lưu trữ câu hỏi, kết quả bài thi và dashboard admin.
- Thiết kế giao diện: Thiết kế giao diện trang web đơn giản, dễ sử dụng, hỗ trợ người học dễ dàng tương tác và hoàn thành bài thi.
- Phát triển và thử nghiệm: Triển khai các tính năng cơ bản và kiểm tra với người dùng thực tế để cải thiện tính năng và giao diện.

### Kết quả đạt được:

Sau quá trình phát triển, em đã hoàn thành được một trang web thi trắc nghiệm với các tính năng cơ bản như: người dùng có thể đăng ký, đăng nhập, chọn bài thi, làm bài thi trắc nghiệm, nhận kết quả ngay lập tức và xem lại các câu mình làm nếu sai sẽ có đáp án chính xác và giải thích đáp án đúng. Hệ thống cũng hỗ trợ lưu trữ các bài thi dựa trên các câu hỏi trắc nghiệm có mức độ khó khác nhau.

## MỞ ĐẦU

### Lý do chọn đề tài:

Trong bối cảnh hội nhập quốc tế, kỹ năng đọc hiểu tiếng Anh ngày càng trở nên quan trọng, đặc biệt đối với các kỳ thi chuẩn quốc tế như IELTS, TOEFL hoặc các kỳ thi đánh giá năng lực. Một trong những dạng câu hỏi thường gặp của kỹ năng Reading là True/False/Not Given, đòi hỏi thí sinh không chỉ hiểu nội dung mà còn phân tích, đánh giá thông tin một cách chính xác. Tuy nhiên, nhiều người dùng gặp khó khăn trong việc làm quen và rèn luyện dạng bài này do thiếu tài liệu và công cụ luyện tập phù hợp.

Do đó, em quyết định chọn đề tài "Xây dựng hệ thống thi trắc nghiệm trực tuyến kỹ năng Reading dạng True/False/Not Given" nhằm:

Tạo môi trường học tập linh hoạt: Ứng dụng công nghệ để cung cấp một nền tảng trực tuyến, giúp người dùng có thể luyện tập mọi lúc, mọi nơi.

Phân tích kết quả: Sử dụng cơ sở dữ liệu NoSQL để lưu trữ và xử lý thông tin một cách nhanh chóng và hiệu quả, đáp ứng nhu cầu của hệ thống thời gian thực.

Đáp ứng nhu cầu đa dạng của người dùng: Xây dựng hệ thống thông minh, cá nhân hóa các bài kiểm tra dựa trên trình độ và nhu cầu của từng người dùng.

Việc thực hiện đề tài không chỉ giúp em áp dụng kiến thức đã học vào thực tế mà còn góp phần xây dựng một công cụ hữu ích, hỗ trợ người dùng nâng cao kỹ năng đọc hiểu tiếng Anh một cách hiệu quả.

### Mục đích nghiên cứu:

Đề tài "Xây dựng hệ thống thi trắc nghiệm trực tuyến kỹ năng Reading dạng True/False/Not Given với cơ sở dữ liệu NoSQL" hướng đến việc tạo ra một nền tảng trực tuyến hỗ trợ người dùng luyện tập kỹ năng đọc hiểu tiếng Anh một cách hiệu quả, đặc biệt với dạng bài True/False/Not Given – một dạng bài thường gây khó khăn trong các kỳ thi tiếng Anh.

Hệ thống được xây dựng nhằm đáp ứng nhu cầu tự học linh hoạt, cho phép người dùng truy cập và luyện tập mọi lúc, mọi nơi. Việc ứng dụng cơ sở dữ liệu NoSQL giúp tối ưu hóa khả năng lưu trữ và xử lý thông tin, đảm bảo hệ thống hoạt động nhanh chóng, chính xác, và dễ dàng mở rộng trong tương lai.



Hệ thống cung cấp các bài kiểm tra chất lượng với nhiều chủ đề và độ khó khác nhau giúp người học nhận biết được điểm mạnh, điểm yếu, từ đó điều chỉnh cách học phù hợp. Mục tiêu cuối cùng là giúp người dùng cải thiện kỹ năng đọc hiểu và đạt kết quả tốt hơn trong các kỳ thi, đồng thời góp phần vào xu hướng số hóa giáo dục hiện đại.

### **Đối tượng nghiên cứu:**

Đối tượng nghiên cứu của đề tài bao gồm hệ thống thi trắc nghiệm trực tuyến và người học tiếng Anh. Trọng tâm của nghiên cứu là việc thiết kế và xây dựng một nền tảng hỗ trợ rèn luyện kỹ năng đọc hiểu, tập trung vào dạng bài True/False/Not Given – một dạng bài phổ biến nhưng đòi hỏi khả năng phân tích và đánh giá thông tin chính xác.

Hệ thống trực tuyến này không chỉ đáp ứng nhu cầu luyện tập của người dùng mà còn tích hợp công nghệ cơ sở dữ liệu NoSQL để tối ưu hóa việc quản lý dữ liệu. NoSQL được lựa chọn do khả năng lưu trữ linh hoạt, hiệu suất truy xuất cao, và khả năng mở rộng phù hợp với các hệ thống giáo dục trực tuyến. Nghiên cứu xem xét cách ứng dụng NoSQL vào việc tổ chức và xử lý dữ liệu bài thi, kết quả người dùng, cũng như tạo ra trải nghiệm học tập cá nhân hóa.

Việc kết hợp nhu cầu thực tiễn của người học với các công nghệ web hiện đại như MERN (MongoDB, Express.js, React.js, Node.js) giúp đảm bảo hệ thống vận hành hiệu quả, thân thiện với người dùng, và hỗ trợ tối đa quá trình học tập. Mục tiêu cuối cùng là xây dựng một công cụ hữu ích, tạo điều kiện thuận lợi để người dùng cải thiện kỹ năng đọc hiểu và nâng cao thành tích trong các kỳ thi tiếng Anh.

### **Phạm vi nghiên cứu:**

Phạm vi nghiên cứu của đề tài tập trung vào việc thiết kế và phát triển một hệ thống thi trắc nghiệm trực tuyến hỗ trợ luyện tập kỹ năng đọc hiểu tiếng Anh, đặc biệt là dạng bài True/False/Not Given. Hệ thống được xây dựng trên nền tảng công nghệ MERN (MongoDB, Express.js, React.js, Node.js), đảm bảo tính linh hoạt, hiệu suất cao trong việc lưu trữ, xử lý dữ liệu và cung cấp trải nghiệm người dùng mượt mà.

Về công nghệ, nghiên cứu tập trung vào việc sử dụng hệ quản trị cơ sở dữ liệu MongoDB để xây dựng cơ sở dữ liệu NoSQL, hỗ trợ lưu trữ linh hoạt thông tin bài kiểm tra, kết quả người dùng và các dữ liệu liên quan. Phần backend sử dụng

Express.js và Node.js để xây dựng các API mạnh mẽ, đảm bảo việc giao tiếp giữa client và server diễn ra hiệu quả. Phần frontend được phát triển bằng React.js, cung cấp giao diện thân thiện, dễ sử dụng và có khả năng đáp ứng nhanh chóng với thao tác của người dùng.

Về nội dung, phạm vi nghiên cứu giới hạn ở dạng bài True/False/Not Given, với mục tiêu tạo ra một hệ thống chuyên sâu, hiệu quả trong việc hỗ trợ người học xử lý dạng bài này. Các dạng bài khác hoặc kỹ năng tiếng Anh khác (nghe, nói, viết) không nằm trong phạm vi của đề tài.

Đối tượng sử dụng chính là các người dùng đang luyện thi các kỳ thi tiếng Anh quốc tế như IELTS hoặc TOEFL, với nhu cầu cải thiện kỹ năng đọc hiểu. Hệ thống trực tuyến này không giới hạn về phạm vi địa lý, nhưng ngôn ngữ hiển thị sẽ ưu tiên tiếng Anh và tiếng Việt để phù hợp với người dùng mục tiêu.

Mục tiêu của nghiên cứu là không chỉ phát triển một công cụ học tập hiệu quả mà còn xây dựng một hệ thống có khả năng mở rộng, phù hợp với sự phát triển của công nghệ MERN (MongoDB, Express.js, React.js, Node.js) và nhu cầu người học trong tương lai.

## CHƯƠNG 1: TỔNG QUAN

### 1.1. Giới thiệu về vấn đề nghiên cứu:

Trong bối cảnh toàn cầu hóa và sự phát triển mạnh mẽ của công nghệ thông tin, việc học tiếng Anh, đặc biệt là kỹ năng đọc hiểu, trở thành một yếu tố quan trọng trong việc nâng cao năng lực cá nhân và đáp ứng yêu cầu của các kỳ thi chuẩn Quốc tế như IELTS, TOEFL. Một trong những dạng bài thường gặp trong các kỳ thi này là dạng True/False/Not Given, yêu cầu thí sinh không chỉ hiểu nội dung bài đọc mà còn phải phân tích, đánh giá tính chính xác của các thông tin được cung cấp.

Tuy nhiên, việc luyện tập và rèn luyện kỹ năng này gặp khó khăn do thiếu các công cụ học tập hiệu quả, dễ sử dụng và phù hợp với nhu cầu của người học. Do đó, việc xây dựng một hệ thống thi trắc nghiệm trực tuyến hỗ trợ người học luyện tập kỹ năng đọc hiểu, đặc biệt với dạng bài True/False/Not Given, là cần thiết và cấp bách.

Đề tài "Xây dựng hệ thống thi trắc nghiệm trực tuyến kỹ năng Reading dạng True/False/Not Given với cơ sở dữ liệu NoSQL" được thực hiện với mục tiêu cung cấp một nền tảng trực tuyến giúp người dùng rèn luyện kỹ năng đọc hiểu tiếng Anh một cách hiệu quả. Hệ thống này sử dụng công nghệ MERN (MongoDB, Express.js, React.js, Node.js) để xây dựng nền tảng vững chắc, tối ưu hóa việc lưu trữ, truy xuất dữ liệu, và mang lại trải nghiệm người dùng mượt mà, thân thiện.

Hệ thống không chỉ cung cấp các bài kiểm tra theo dạng True/False/Not Given mà còn hỗ trợ phân tích kết quả người dùng, giúp họ nhận diện điểm mạnh, điểm yếu và cải thiện chiến lược học tập của mình. Đề tài này không chỉ giải quyết nhu cầu học tiếng Anh của người dùng mà còn mang đến một công cụ hữu ích, phù hợp với xu hướng số hóa trong giáo dục, giúp người dùng luyện tập mọi lúc, mọi nơi, và đạt được kết quả cao hơn trong các kỳ thi.

Thông qua việc ứng dụng các công nghệ web hiện đại và giải pháp giáo dục sáng tạo, đề tài mong muốn đóng góp vào việc cải thiện kỹ năng đọc hiểu tiếng Anh cho người dùng, đồng thời cung cấp một nền tảng học tập linh hoạt và hiệu quả.

### 1.2. Mục tiêu và ý nghĩa của nghiên cứu:

Mục tiêu chính của đề tài "Xây dựng hệ thống thi trắc nghiệm trực tuyến kỹ năng Reading dạng True/False/Not Given với cơ sở dữ liệu NoSQL" là phát triển một nền tảng trực tuyến giúp người dùng luyện tập và cải thiện kỹ năng đọc hiểu tiếng

Anh, đặc biệt là trong các bài thi có dạng câu hỏi True/False/Not Given. Cụ thể, mục tiêu của đề tài bao gồm:

- Xây dựng hệ thống trực tuyến: Phát triển một hệ thống thi trắc nghiệm trực tuyến với các bài kiểm tra dạng True/False/Not Given, giúp người học có thể luyện tập và tự đánh giá khả năng đọc hiểu mọi lúc, mọi nơi.
- Ứng dụng công nghệ MERN: Sử dụng công nghệ MERN (MongoDB, Express.js, React.js, Node.js) để xây dựng nền tảng vững chắc, tối ưu hóa hiệu suất lưu trữ, xử lý và truy xuất dữ liệu, tạo ra một hệ thống nhanh chóng, ổn định và dễ sử dụng.
- Tạo ra môi trường học tập linh hoạt: Cung cấp một công cụ học tập trực tuyến, giúp người học có thể tiếp cận tài liệu và luyện tập mọi lúc, mọi nơi, từ đó nâng cao hiệu quả học tập và chuẩn bị tốt hơn cho các kỳ thi tiếng Anh.

#### **Ý nghĩa nghiên cứu:**

Đề tài có ý nghĩa quan trọng cả về mặt lý thuyết và thực tiễn. Cụ thể:

- Giải quyết nhu cầu thực tiễn: Đề tài giúp đáp ứng nhu cầu học tiếng Anh của người dùng, đặc biệt trong việc luyện tập kỹ năng đọc hiểu, một trong những kỹ năng quan trọng trong các kỳ thi tiếng Anh quốc tế như IELTS, TOEFL. Việc sử dụng dạng bài True/False/Not Given sẽ giúp người dùng rèn luyện khả năng phân tích, đánh giá thông tin chính xác từ văn bản.
- Cải thiện kỹ năng đọc hiểu: Hệ thống này sẽ giúp người dùng nâng cao khả năng đọc hiểu, một kỹ năng cần thiết không chỉ trong các kỳ thi mà còn trong công việc và học tập sau này. Việc cải thiện kỹ năng đọc hiểu sẽ giúp người dùng tự tin hơn khi làm bài kiểm tra và đạt điểm cao hơn trong các kỳ thi tiếng Anh.
- Ứng dụng công nghệ vào giáo dục: Đề tài ứng dụng các công nghệ web hiện đại và cơ sở dữ liệu NoSQL, góp phần thúc đẩy sự phát triển của các nền tảng học tập trực tuyến. Hệ thống có thể được mở rộng và cải tiến trong tương lai, tạo ra một công cụ học tập linh hoạt, hiệu quả và dễ tiếp cận cho mọi đối tượng người học.
- Tạo cơ hội học tập cho mọi người: Việc xây dựng hệ thống trực tuyến giúp người dùng có thể luyện tập và cải thiện kỹ năng mọi lúc, mọi nơi mà không bị giới hạn về thời gian hay không gian. Điều này đặc biệt quan trọng trong bối cảnh giáo dục ngày càng chuyển hướng sang học trực tuyến và tự học.

### 1.3. Tình hình nghiên cứu hiện nay

Trong những năm gần đây, nghiên cứu và phát triển các hệ thống học trực tuyến đã trở thành xu hướng quan trọng trong giáo dục, đặc biệt trong bối cảnh sự phát triển mạnh mẽ của công nghệ thông tin và yêu cầu ngày càng cao về các kỹ năng ngoại ngữ, đặc biệt là tiếng Anh. Các nền tảng học trực tuyến hỗ trợ luyện tập kỹ năng đọc hiểu tiếng Anh, đặc biệt là các dạng bài trắc nghiệm như True/False/Not Given, đang ngày càng nhận được sự quan tâm từ các nhà nghiên cứu và phát triển phần mềm giáo dục.

Hệ thống học trực tuyến và thi trắc nghiệm: Các nghiên cứu gần đây đã chỉ ra rằng việc sử dụng các nền tảng trực tuyến để rèn luyện kỹ năng đọc hiểu là một giải pháp hiệu quả, đặc biệt khi kết hợp với các bài kiểm tra dạng trắc nghiệm. Những nghiên cứu này tập trung vào việc phát triển các công cụ học tập linh hoạt, dễ tiếp cận và hỗ trợ người học tự đánh giá kết quả của mình. Tuy nhiên, phần lớn các hệ thống hiện nay chưa có sự cá nhân hóa cao và chưa khai thác triệt để khả năng phân tích kết quả để đưa ra các gợi ý học tập phù hợp.

Ứng dụng công nghệ NoSQL trong giáo dục: Các nghiên cứu về việc ứng dụng cơ sở dữ liệu NoSQL trong các hệ thống học trực tuyến đang ngày càng phát triển. NoSQL, với ưu điểm lưu trữ linh hoạt và khả năng mở rộng, đã được sử dụng trong nhiều ứng dụng giáo dục, đặc biệt là trong việc quản lý dữ liệu người dùng, kết quả bài kiểm tra và tài liệu học tập. Tuy nhiên, việc kết hợp NoSQL với các công nghệ như MERN trong phát triển hệ thống thi trắc nghiệm trực tuyến còn hạn chế, đặc biệt trong bối cảnh ứng dụng trong giáo dục đọc hiểu tiếng Anh.

Ứng dụng công nghệ web hiện đại (MERN): Các nghiên cứu gần đây cũng tập trung vào việc áp dụng công nghệ web hiện đại, đặc biệt là các framework như React.js và Node.js, vào việc xây dựng các nền tảng học trực tuyến. React.js được sử dụng để xây dựng giao diện người dùng dễ sử dụng, đáp ứng nhanh chóng các thao tác của người học, trong khi Node.js và Express.js cung cấp các công cụ mạnh mẽ để phát triển backend, tạo ra một môi trường học tập ổn định và hiệu quả. Tuy nhiên, việc ứng dụng công nghệ MERN vào xây dựng các hệ thống thi trắc nghiệm cho các kỹ năng cụ thể như Reading vẫn còn là một lĩnh vực mới và ít được nghiên cứu.

Cải thiện kỹ năng đọc hiểu tiếng Anh: Các nghiên cứu về cải thiện kỹ năng đọc hiểu tiếng Anh thường tập trung vào việc sử dụng các bài tập và phương pháp luyện

tập truyền thống. Tuy nhiên, với sự phát triển của công nghệ, việc áp dụng các phương pháp học trực tuyến và công nghệ phân tích dữ liệu đã trở thành một xu hướng mới. Các nghiên cứu cho thấy rằng việc kết hợp bài kiểm tra tự động với phân tích kết quả có thể giúp người dùng nhận diện được điểm yếu của mình và cải thiện kỹ năng đọc hiểu một cách hiệu quả hơn.

Tuy đã có những tiến bộ đáng kể trong việc phát triển các công cụ học tập trực tuyến và các phương pháp rèn luyện kỹ năng đọc hiểu tiếng Anh, nhưng vẫn còn nhiều thách thức, đặc biệt trong việc xây dựng hệ thống học tập có tính cá nhân hóa cao, dễ sử dụng và đáp ứng nhu cầu đa dạng của người học. Đề tài của bạn về xây dựng hệ thống thi trắc nghiệm trực tuyến kỹ năng Reading dạng True/False/Not Given với cơ sở dữ liệu NoSQL sẽ đóng góp vào việc giải quyết những hạn chế này, mang lại một giải pháp học tập hiệu quả và hiện đại.

#### **1.4. Các vấn đề cần giải quyết**

Trong quá trình thực hiện đề tài "Xây dựng hệ thống thi trắc nghiệm trực tuyến kỹ năng Reading dạng True/False/Not Given với cơ sở dữ liệu NoSQL", có một số vấn đề quan trọng cần được giải quyết để đảm bảo hệ thống hoạt động hiệu quả và đáp ứng nhu cầu học tập của người dùng. Các vấn đề này bao gồm:

- Thiết kế và phát triển hệ thống thi trắc nghiệm trực tuyến:
  - Tạo ra bài kiểm tra chất lượng: Đảm bảo các bài kiểm tra được xây dựng đúng chuẩn, phù hợp với mục tiêu luyện tập kỹ năng đọc hiểu, đặc biệt là dạng bài True/False/Not Given. Các bài kiểm tra phải có sự phân bố hợp lý về độ khó và nội dung.
  - Giao diện người dùng thân thiện: Phát triển một giao diện người dùng dễ sử dụng, trực quan và thân thiện để người học có thể dễ dàng truy cập và tham gia bài kiểm tra mà không gặp phải khó khăn về mặt kỹ thuật.
- Ứng dụng cơ sở dữ liệu NoSQL hiệu quả:
  - Lưu trữ và truy xuất dữ liệu nhanh chóng: Cần đảm bảo hệ thống có khả năng lưu trữ, quản lý và truy xuất dữ liệu nhanh chóng, đặc biệt là dữ liệu về kết quả người dùng, bài kiểm tra và tiến độ học tập. Cơ sở dữ liệu NoSQL như MongoDB sẽ cần được tối ưu hóa để xử lý lượng dữ liệu lớn, đồng thời cung cấp các tính năng mở rộng trong tương lai.

- Quản lý dữ liệu người dùng: Hệ thống cần phải quản lý dữ liệu người dùng hiệu quả, từ thông tin cá nhân đến các kết quả bài kiểm tra, điểm số, giúp cá nhân hóa trải nghiệm học tập cho từng người.
- Khả năng mở rộng hệ thống:
  - Khả năng mở rộng: Hệ thống cần được thiết kế sao cho dễ dàng mở rộng trong tương lai, khi số lượng người dùng và bài kiểm tra ngày càng tăng. Cần tính đến khả năng bổ sung thêm tính năng mới hoặc thay đổi cấu trúc mà không làm gián đoạn hoạt động của hệ thống.

### 1.5. Phạm vi nghiên cứu

Đề tài nghiên cứu này tập trung vào việc xây dựng một hệ thống thi trắc nghiệm trực tuyến để rèn luyện kỹ năng đọc hiểu tiếng Anh, đặc biệt với dạng bài True/False/Not Given, nhằm hỗ trợ người học chuẩn bị cho các kỳ thi Quốc tế như IELTS và TOEFL. Hệ thống sẽ được phát triển sử dụng công nghệ MERN (MongoDB, Express.js, React.js, Node.js) để xây dựng nền tảng vững chắc và dễ mở rộng. Phạm vi nghiên cứu bao gồm thiết kế hệ thống thi trắc nghiệm với nhiều chủ đề và mức độ khó khác nhau, tự động chấm điểm và cung cấp kết quả cho người dùng, kèm theo đáp án và giải thích chi tiết. Mục tiêu là tạo ra một trải nghiệm học tập cá nhân hóa, đồng thời cung cấp một môi trường học trực tuyến linh hoạt, dễ dàng truy cập từ nhiều thiết bị.

## CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

### 2.1. Cơ sở lý thuyết

Trong phần này, em sẽ trình bày các lý thuyết, khái niệm và cơ sở khoa học liên quan đến đề tài nghiên cứu của mình. Đây là phần nền tảng lý thuyết giúp giải thích và làm rõ các vấn đề nghiên cứu, đồng thời cung cấp cơ sở để xây dựng và phát triển hệ thống trong đồ án.

#### 2.1.1 Kỹ năng đọc hiểu trong học tiếng Anh

Kỹ năng đọc hiểu là khả năng tiếp nhận và xử lý thông tin từ văn bản, từ đó trả lời các câu hỏi, hiểu được thông điệp mà văn bản truyền tải. Đối với học sinh, việc luyện tập kỹ năng này không chỉ giúp nâng cao khả năng hiểu ngôn ngữ mà còn cải thiện khả năng phân tích và tư duy phản biện.

Đặc điểm của kỹ năng đọc hiểu: Là một kỹ năng quan trọng trong quá trình học ngoại ngữ, kỹ năng đọc hiểu đòi hỏi người học phải có khả năng nhận diện từ vựng, cấu trúc câu và ngữ cảnh để hiểu đúng nghĩa của văn bản.

Các yếu tố ảnh hưởng đến kỹ năng đọc hiểu: Bao gồm từ vựng, cấu trúc ngữ pháp, kỹ năng phân tích văn bản, và chiến lược đọc hiểu.

#### 2.1.2 Dạng bài True/False/Not Given trong thi cử:

Dạng bài True/False/Not Given là một phần quan trọng trong các bài kiểm tra kỹ năng đọc hiểu trong các kỳ thi tiếng Anh Quốc tế. Mục đích của dạng câu hỏi này là kiểm tra khả năng nhận diện thông tin chính xác trong một bài đọc.

- True (Đúng): Câu trả lời đúng khi thông tin trong câu hỏi khớp chính xác với nội dung bài đọc.
  - False (Sai): Câu trả lời sai khi thông tin trong câu hỏi mâu thuẫn hoặc không phù hợp với nội dung bài đọc.
  - Not Given (Không cho biết): Câu trả lời là "Not Given" khi thông tin trong câu hỏi không có trong bài đọc, nghĩa là không đủ dữ liệu để xác định đúng hay sai.
- Dạng bài này không chỉ giúp đánh giá khả năng đọc hiểu mà còn kiểm tra khả năng phân tích thông tin và xử lý các tình huống khi không có đủ thông tin. Việc phát triển các bài kiểm tra dạng này trong hệ thống trực tuyến giúp người dùng luyện tập và cải thiện kỹ năng đọc hiểu một cách hiệu quả.



### 2.1.3 Cơ sở dữ liệu NoSQL

Cơ sở dữ liệu NoSQL (hay NoSQL databases) là hệ thống quản lý dữ liệu phi quan hệ được xây dựng dành riêng cho mô hình dữ liệu và có schema (lược đồ) rất linh hoạt.

Việc xây dựng cơ sở dữ liệu phi quan hệ có mục đích dành cho các kho dữ liệu phân tán nhằm đáp ứng nhu cầu lưu trữ dữ liệu lớn.

Cơ sở dữ liệu NoSQL được ứng dụng vào các web có thời gian giống với thời gian thực và có nguồn dữ liệu lớn thu thập hàng ngày như Google hay Facebook.

Loại cơ sở dữ liệu này là thuật ngữ được viết tắt từ cụm “Not Only SQL”. Cụ pháp SQL được sử dụng với cơ sở dữ liệu quan hệ RDBMS nhằm lưu trữ và truy xuất dữ liệu khi cần thiết.

Bên cạnh đó, hệ thống cơ sở dữ liệu NoSQL có thể lưu dữ liệu có cấu trúc, bán cấu trúc, phi cấu trúc hay dữ liệu đa hình. Cơ sở dữ liệu NoSQL được ứng dụng rộng rãi nhờ có thể thực hiện ở quy mô lớn, dễ phát triển cũng như nhiều chức năng.

#### **Một số lý do cụ thể hơn mà các công ty nên sử dụng NoSQL là:**

NoSQL có tính linh hoạt cao: Cơ sở dữ liệu này sẽ cung cấp các sơ đồ linh hoạt giúp phát triển các công đoạn nhanh hơn cùng khả năng lặp lại. Mô hình dữ liệu có thể chuyển từ cơ sở dữ liệu phi quan hệ thành ý tưởng cho các loại dữ liệu không cấu trúc hoặc cấu trúc chưa hoàn chỉnh.

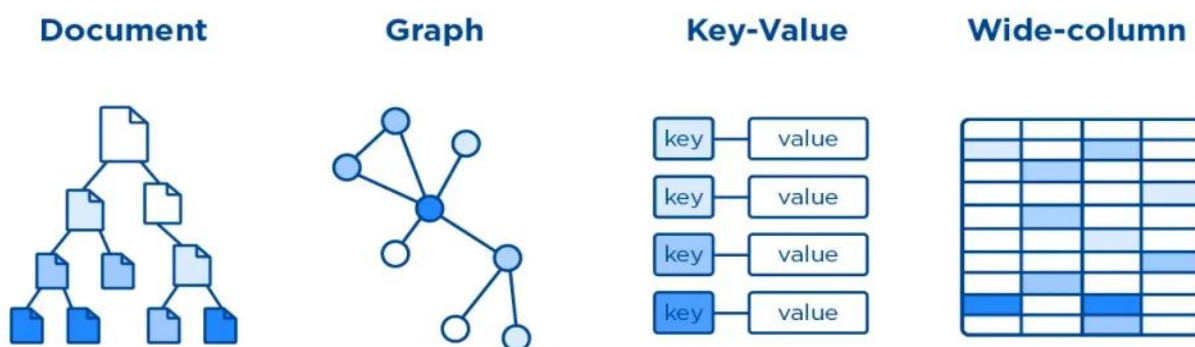
NoSQL có khả năng thay đổi quy mô: NoSQL được thiết kế nhằm tăng quy mô bằng các cụm phần cứng được phân phối thay vì bổ sung máy chủ mạnh và tốn kém.

Hiệu năng của NoSQL cao: Cơ sở dữ liệu phi quan hệ sẽ được tối ưu hoá từng mô hình cụ thể và mẫu truy cập. Điều này sẽ giúp tăng hiệu năng so với việc đạt chức năng tương tự bằng cơ sở dữ liệu quan hệ.

Tính thiết thực cao: Cơ sở dữ liệu phi quan hệ có cung cấp các kiểu dữ liệu thiết thực như API được xây dựng cho từng mô hình dữ liệu riêng.

#### **Các loại cơ sở dữ liệu NoSQL:**

Cơ sở dữ liệu phi quan hệ có thể được chia ra làm 4 loại chính bao gồm: Document, Graph, Key-Value và Column.

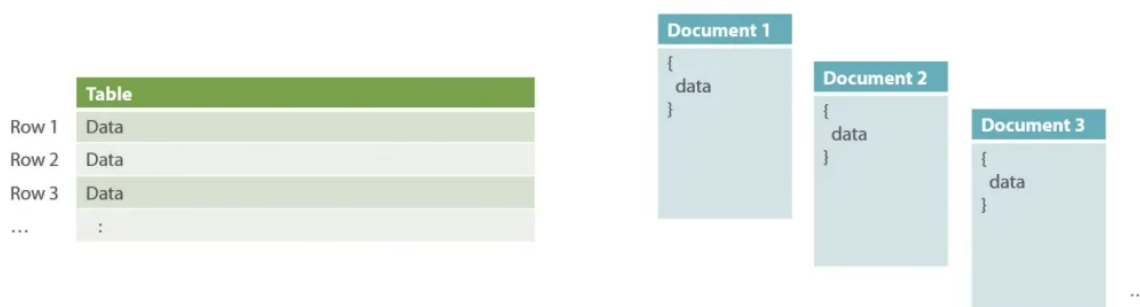


Hình 1: Các loại cơ sở dữ liệu NoSQL

## 1. Document-based

Loại cơ sở dữ liệu dạng document – based (hay còn gọi là cơ sở dữ liệu dạng tài liệu) có vai trò lưu trữ, truy xuất dữ liệu dưới dạng một cặp khoá giá trị (key value) nhưng phần giá trị sẽ được lưu trữ ở dạng tài liệu. Phần tài liệu ở đây sẽ có định dạng XML hoặc JSON.

Cơ sở dữ liệu dạng tài liệu sẽ giúp nhà phát triển lưu trữ dễ dàng cũng như truy vấn dữ liệu hiệu quả hơn bằng cách sử dụng cùng một định dạng mô hình văn bản họ sử dụng trong mã ứng dụng.



Hình 2: Loại cơ sở dữ liệu dạng document – based

Với sự linh hoạt, bán cấu trúc và phân cấp của các văn bản và cơ sở dữ liệu tài liệu cho phép chúng có thể phát triển phù hợp với nhiều ứng dụng.

Mô hình dữ liệu kiểu tài liệu sẽ phát huy hiệu quả với danh mục, hồ sơ người dùng cũng như trong hệ thống quản lý nội dung.

Hiện nay, Amazon DocumentDB và MongoDB là những cơ sở dữ liệu kiểu tài liệu được sử dụng phổ biến và có cung cấp API mạnh mẽ, trực quan để phát triển tính lặp lại và sự linh hoạt.

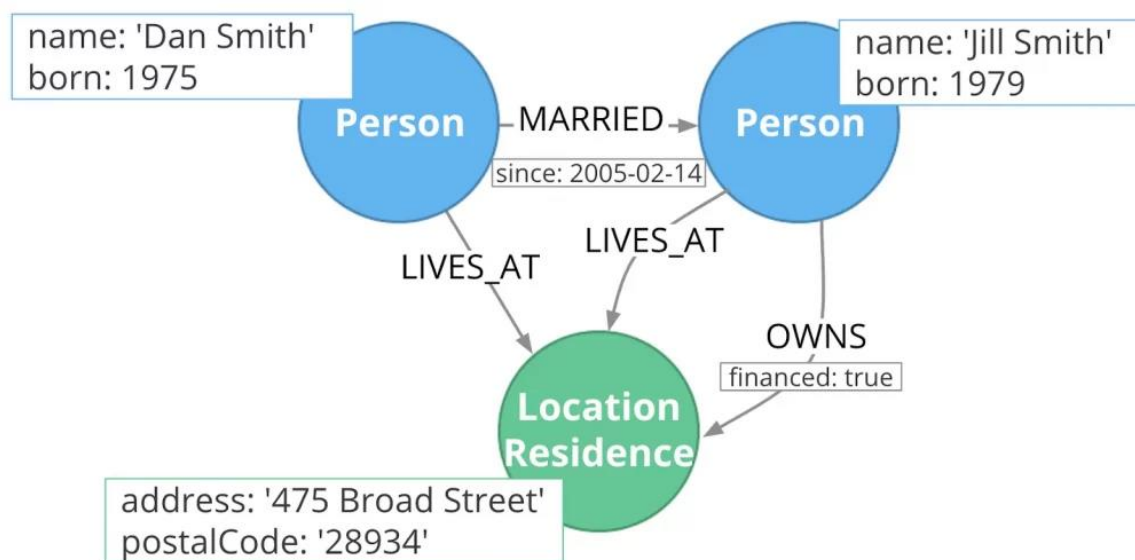
Ngoài ra, một số hệ thống DBMS document-based NoSQL tiêu biểu khác có thể kể đến là Riak, Lotus Notes hay CouchDB.

Giới hạn của dữ liệu document-based là thông tin cơ sở bị trùng lặp nhiều trên tài liệu và có thiết kế phức tạp dẫn đến sự không nhất quán.

## 2. Graph-based

Graph-based hay cơ sở dữ liệu kiểu đồ thị có thể lưu trữ các thực thể và các mối quan hệ giữa các thực thể đó.

Những thực thể này được lưu dưới dạng một node với mỗi quan hệ là các cạnh. Mỗi một cạnh sẽ cho biết một mối quan hệ giữa các node. Mỗi node và cạnh đều chỉ có một mã định danh.



Hình 3: Ví dụ cơ sở dữ liệu dạng Graph-based

Mục đích của cơ sở dữ liệu đồ thị là dựng và chạy ứng dụng hoạt động cùng với các bộ dữ liệu có khả năng kết nối cao trở nên dễ dàng hơn.

Cơ sở dữ liệu dạng graph-based được sử dụng cho các mạng xã hội, đồ thị tri thức, công cụ đề xuất và phát hiện ra lừa đảo. Lý do bởi mối quan hệ giữa cơ sở dữ liệu

liệu dạng graph-based có bản chất là đa quan hệ và có thể truyền tải nhanh chóng bởi nó đã được ghi lại vào DB.

**Một số giới hạn của cơ sở dữ liệu kiểu graph-based vẫn còn tồn tại là:**

Dữ liệu ở dạng graph-based thiếu tính đồng thời hiệu suất cao. Ở nhiều trường hợp, graph-based chỉ cung cấp kiểu đọc, kiểu ghi đơn dẫn đến làm giảm hiệu suất và ngăn cản sự đồng thời, từ đó hạn chế tính song song phân luồng (hay threaded parallelism).

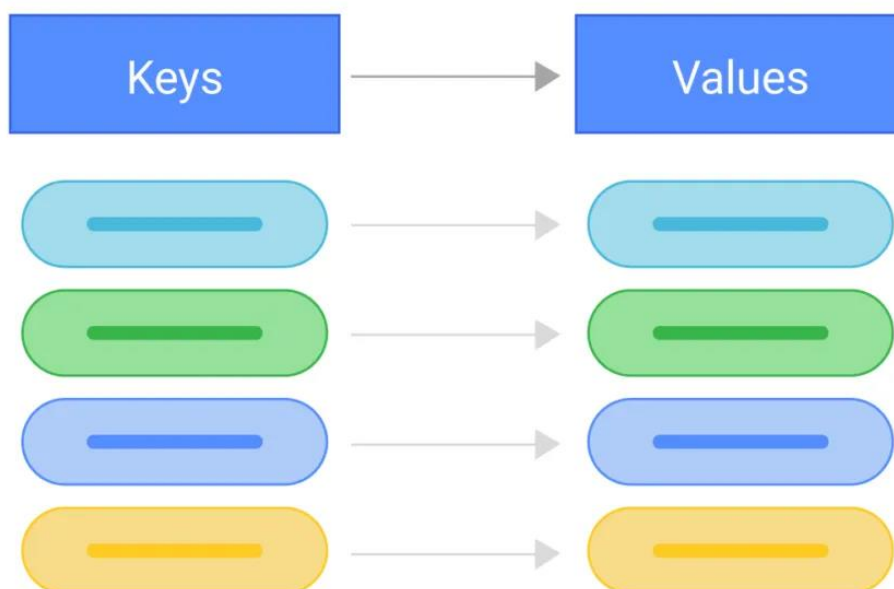
Thứ hai, dữ liệu dạng đồ thị thiếu ngôn ngữ chuẩn trong thiết lập và khai báo.

Dữ liệu graph-based thiếu tính song song (parallelism). Dữ liệu dạng này sẽ không cung cấp các truy vấn song song trên các biểu đồ lớn dẫn đến việc phân vùng biểu đồ khó khăn hơn.

### 3. Key value

Với cơ sở dữ liệu kiểu khoá – giá trị, dữ liệu sẽ được lưu trữ trong các cặp khoá – giá trị (key – value pairs). Kiểu cơ sở dữ liệu này được thiết kế để xử lý nhiều dữ liệu và tải nặng.

Cơ sở dữ liệu khoá – giá trị sẽ lưu trữ dữ liệu dưới dạng bảng trong đó mỗi khoá sẽ là duy nhất và giá trị đó có thể là BLOB hay JSON.



Hình 4: Cơ sở dữ liệu dạng Keys - Values

Cơ sở dữ liệu kiểu key-value có khả năng phân mảnh cao, cho phép thay đổi quy mô theo chiều ngang mà các loại hình cơ sở dữ liệu khác không thể.

Loại cơ sở dữ liệu này được ứng dụng cho công nghệ quảng cáo, các trò chơi và rất thích hợp với IoT.

Nó được dùng như từ điển, bộ sưu tập hay mảng kết hợp. Cơ sở dữ liệu key-value cũng cho phép nhà phát triển lưu trữ dữ liệu mà không có schema.

**Cơ sở dữ liệu kiểu key-value có một số hạn chế như:**

- Không có mối quan hệ với Multiple data.
- Nếu bạn đang lưu trữ số lượng lớn khóa và không thể lưu một trong các khóa, bạn không thể quay về các phần còn lại của thao tác thực hiện (multi operation transactions).
- Kết quả tìm kiếm “khóa” dựa vào một số thông tin được tìm thấy ở phần “giá trị” của các cặp key-value (query data by “value”).
- Do các hoạt động bị giới hạn trong một khóa tại một thời điểm nhất định nên không có cách để chạy nhiều khóa trong cùng một khoảng thời gian (operation by groups).

Một số cơ sở dữ liệu kiểu key-value được sử dụng phổ biến hiện nay là DynamoDB, Berkeley DB, Redis hay Riak.

#### **4. Column – Based**

Với kiểu cơ sở dữ liệu column-based hay kiểu cơ sở dữ liệu dạng cột thì dữ liệu sẽ được lưu trữ trong database dưới dạng các cột. Các cột này sẽ được xử lý tách biệt và giá trị của cơ sở dữ liệu column-based sẽ được lưu liên kế nhau.

Cơ sở dữ liệu dạng cột có thể mang lại hiệu suất cao cho các truy vấn tổng hợp như AVG, SUM, MIN hay COUNT,... vì đã có sẵn dữ liệu trong một cột. Cơ sở dữ liệu NoSQL cũng dựa trên cột để quản lý CRM, warehouse hay library card catalogs,...

**Hạn chế của cơ sở dữ liệu NoSQL:**

Cơ sở dữ liệu phi quan hệ được tạo ra nhằm mục đích giải quyết các hạn chế của cơ sở dữ liệu SQL. Mặc dù có rất nhiều ưu thế nhưng cơ sở dữ liệu NoSQL vẫn còn có những hạn chế nhất định.

### **Sự nhất quán:**

Cơ sở dữ liệu NoSQL không có tính nhất quán mạnh mẽ hoặc tức thì để có hiệu suất và tính khả dụng tốt hơn.

### **Cơ sở dữ liệu thông thường sẽ đảm bảo được các hoạt động:**

Có tính nguyên tử (atomicity): các thành phần giao dịch thành công hoặc không có giao dịch.

Có tính biệt lập (isolation): giao dịch không có cạnh tranh.

Có tính bền vững (durability): có khả năng phục hồi tình trạng hư hỏng về trạng thái đã biết cuối cùng.

Có tính nhất quán (consistency): tất cả người sử dụng xem cùng một chế độ dữ liệu.

Những thuộc tính trên được viết tắt thành ACID và chúng được xử lý khác nhau trên các hệ thống NoSQL.

Cơ sở dữ liệu NoSQL sẽ phải nói lỏng một số thuộc tính của ACID để có được mô hình dữ liệu linh hoạt và có khả năng thay đổi quy mô theo chiều ngang.

Bên cạnh đó vẫn có một số cơ sở dữ liệu phi quan hệ có cơ chế giải để giải quyết vấn đề trên. MongoDB là một trong số đó. Nó đảm bảo sự nhất quán cho các hoạt động riêng lẻ nhưng không thể đảm bảo cho tất cả cơ sở dữ liệu.

### **Thuật ngữ SQL so với NoSQL:**

Với bảng dưới đây, bạn sẽ so sánh được thuật ngữ sử dụng trong cơ sở dữ liệu phi quan hệ với thuật ngữ sử dụng trong cơ sở dữ liệu quan hệ.

Bảng 1: So sánh SQL Server và MongoDB

<b>SQL Server</b>	<b>MongoDB</b>
Bảng	Bộ sưu tập
Hàng	Tài liệu
Cột	Trường
Khoá chính	ID đối tượng
Chỉ mục	Chỉ mục
Chế độ xem	Chế độ xem
Bảng hoặc đối tượng lồng ghép nhau	Văn bản nhúng
Mảng	Mảng

#### 2.1.4 Công nghệ MERN (MongoDB, Express.js, React.js, Node.js)

MERN là bộ công nghệ phát triển ứng dụng web phổ biến hiện nay, bao gồm bốn công nghệ chính: MongoDB, Express.js, React.js và Node.js. Các công nghệ này được kết hợp để tạo ra các ứng dụng web mạnh mẽ, hiệu quả và dễ dàng mở rộng.

- **MongoDB:** Là hệ quản trị cơ sở dữ liệu NoSQL là nơi lưu trữ dữ liệu chính trong hệ thống, MongoDB hỗ trợ lưu trữ và truy xuất dữ liệu phi cấu trúc, giúp hệ thống dễ dàng mở rộng và quản lý dữ liệu người dùng, kết quả thi và bài kiểm tra.
- **Express.js:** Đây là một framework cho Node.js, giúp phát triển phần backend của ứng dụng. Express.js hỗ trợ xử lý các yêu cầu HTTP, quản lý các API và kết nối với hệ quản trị cơ sở dữ liệu MongoDB. Nó đóng vai trò quan trọng trong việc xử lý và trả về các dữ liệu của hệ thống.
- **React.js:** Là một thư viện JavaScript phát triển giao diện người dùng (UI). React.js cho phép xây dựng các thành phần UI động, giúp tạo ra một giao diện người dùng mượt mà và dễ sử dụng. Nó giúp người học tương tác với hệ thống thi trắc nghiệm trực tuyến một cách dễ dàng và hiệu quả.
- **Node.js:** Là môi trường runtime cho JavaScript trên server-side. Node.js cho phép xử lý các yêu cầu đồng thời và đảm bảo tốc độ truy cập nhanh chóng cho người dùng. Node.js phù hợp với các ứng dụng cần xử lý nhiều yêu cầu đồng thời như các hệ thống học trực tuyến.

#### 2.1.5 Hệ quản trị cơ sở dữ liệu MongoDB

##### 1. Định nghĩa về MongoDB

MongoDB là một cơ sở dữ liệu mã nguồn mở và là cơ sở dữ liệu NoSQL hàng đầu, được hàng triệu người sử dụng. MongoDB được viết bằng C++.

Ngoài ra, MongoDB là một cơ sở dữ liệu đa nền tảng, hoạt động trên các khái niệm Collection và Document, nó cung cấp hiệu suất cao, tính khả dụng cao và khả năng mở rộng dễ dàng.

## 2. Các thuật ngữ hay sử dụng trong MongoDB

**\_id:** Là trường bắt buộc có trong mỗi document. Trường \_id đại diện cho một giá trị duy nhất trong document MongoDB. Trường \_id cũng có thể được hiểu là khóa chính trong document. Nếu thêm mới một document thì MongoDB sẽ tự động sinh ra một \_id đại diện cho document đó và là duy nhất trong cơ sở dữ liệu MongoDB.

**Collection :** Là nhóm của nhiều document trong MongoDB. Collection có thể được hiểu là một bảng tương ứng trong cơ sở dữ liệu RDBMS (Relational Database Management System). Collection nằm trong một cơ sở dữ liệu duy nhất. Các collection không phải định nghĩa các cột, các hàng hay kiểu dữ liệu trước.

**Cursor :** Đây là một con trỏ đến tập kết quả của một truy vấn. Máy khách có thể lặp qua một con trỏ để lấy kết quả.

**Database:** Nơi chứa các Collection, giống với cơ sở dữ liệu RDMS chúng chứa các bảng. Mỗi Database có một tập tin riêng lưu trữ trên bộ nhớ vật lý. Một máy chủ MongoDB có thể chứa nhiều Database.

**Document:** Một bản ghi thuộc một Collection thì được gọi là một Document. Các Document lần lượt bao gồm các trường tên và giá trị.

**Field:** Là một cặp name – value trong một document. Một document có thể có không hoặc nhiều trường. Các trường giống các cột ở cơ sở dữ liệu quan hệ.

**JSON:** Viết tắt của JavaScript Object Notation. Con người có thể đọc được ở định dạng văn bản đơn giản thể hiện cho các dữ liệu có cấu trúc. Hiện tại JSON đang hỗ trợ rất nhiều ngôn ngữ lập trình.

**Index:** Là những cấu trúc dữ liệu đặc biệt, dùng để chứa một phần nhỏ của các tập dữ liệu một cách dễ dàng để quét. Chỉ số lưu trữ giá trị của một fields cụ thể hoặc thiết lập các fields, sắp xếp theo giá trị của các fields này. Index hỗ trợ độ phân tích một cách hiệu quả các truy vấn. Nếu không có chỉ mục, MongoDB sẽ phải quét tất cả các documents của collection để chọn ra những document phù hợp với câu truy vấn. Quá trình quét này là không hiệu quả và yêu cầu MongoDB để xử lý một khối lượng lớn dữ liệu.

## 3. Các kiểu dữ liệu trong MongoDB

MongoDB hỗ trợ các kiểu dữ liệu sau:



Bảng 2: Các kiểu dữ liệu trong MongoDB

Type	Number	Alias
Double	1	“double”
String	2	“string”
Object	3	“object”
Array	4	“array”
Binary data	5	“binData”
Undefined	6	“undefined”
ObjectId	7	“objectId”
Boolean	8	“bool”
Date	9	“date”
Null	10	“null”
Regular Expression	11	“regex”
DBPointer	12	“dbPointer”
JavaScript	13	“javascript”
Symbol	14	“symbol”
JavaScript (with scope)	15	“javascriptWithScope”
32-bit integer	16	“int”
Timestamp	17	“timestamp”
64-bit integer	18	“long”
Decimal128	19	“decimal”
Min key	-1	“minKey”
Max key	127	“maxKey”

**String:** Đây là kiểu dữ liệu được sử dụng phổ biến nhất để lưu giữ dữ liệu. Chuỗi trong MongoDB phải là UTF-8 hợp lệ.

**Int:** Kiểu dữ liệu này được sử dụng để lưu một giá trị số. Số nguyên có thể là 32 bit hoặc 64 bit phụ thuộc vào Server của bạn.

**Boolean:** Kiểu dữ liệu này được sử dụng để lưu giữ một giá trị Boolean (true/false).

**Double:** Kiểu dữ liệu này được sử dụng để lưu các giá trị số thực dấu chấm động.

**Min/ Max keys:** Kiểu dữ liệu này được sử dụng để so sánh một giá trị với các phần tử BSON thấp nhất và cao nhất.

**Array:** Kiểu dữ liệu này được sử dụng để lưu giữ các mảng hoặc danh sách hoặc nhiều giá trị vào trong một key.

**Timestamp:** Đánh dấu thời điểm một Document được sửa đổi hoặc được thêm vào.

**Object:** Kiểu dữ liệu này được sử dụng cho các Document được nhúng vào.

**Null:** Kiểu dữ liệu này được sử dụng để lưu một giá trị Null.

**Symbol:** Kiểu dữ liệu này được sử dụng giống như một chuỗi

**Date :** Kiểu dữ liệu này được sử dụng để lưu giữ date và time hiện tại trong định dạng UNIX time.

**Object ID:** Kiểu dữ liệu này được sử dụng để lưu giữ ID của Document.

**Binary data:** Kiểu dữ liệu này được sử dụng để lưu giữ dữ liệu nhị phân.

**Code:** Kiểu dữ liệu này được sử dụng để lưu giữ JavaScript code vào trong Document.

**Regular expression:** Kiểu dữ liệu này được sử dụng để lưu giữ Regular Expresion.

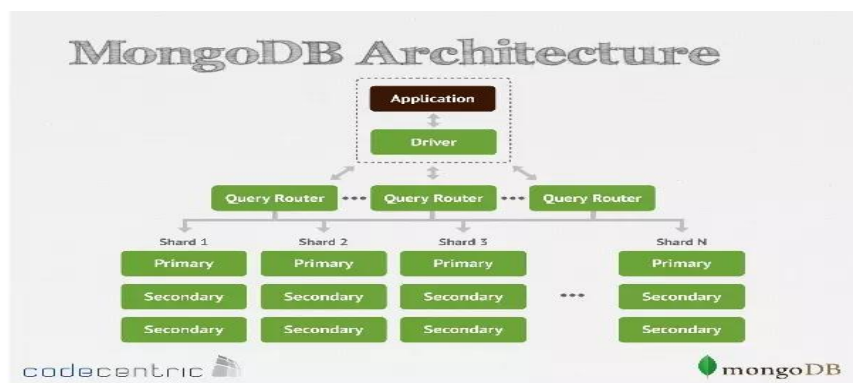
#### 4. So sánh một số câu lệnh dùng trong SQL và MongoDB

Bảng 3: Các câu lệnh trong SQL và MongoDB

Câu lệnh	SQL	MongoDB
Create table	CREATE TABLE people (id MEDIUMINT NOT NULL AUTO_INCREMENT, user_id Varchar(30), age Number, status char(1), PRIMARY KEY (id))	db.people.insertOne({User_id: "abc123", Age: 55, Status: "A"})
Drop table	DROP TABLE people	db.people.drop()
Insert records into tables	INSERT INTO people(user_id, age, status) VALUES ("bcd001", 45, "A")	db.people.insertOne( { user_id: "bcd001", age: 45, status: "A" })

Câu lệnh	SQL	MongoDB
Select	SELECT *FROM people	db.people.find()
	SELECT id,user_id, status FROM people	db.people.find( { }, { user_id: 1, status: 1 } )
	SELECT * FROM people WHERE status = "A"	db.people.find( { status: "A" } )
	SELECT * FROM people WHERE status = "A" AND age = 50	db.people.find( { status: "A", age: 50 } )
	SELECT * FROM people WHERE status = "A" OR age = 50	db.people.find( { \$or: [ { status: "A" } , { age: 50 } ] } )
	SELECT * FROM people WHERE user_id like "%bc%"	db.people.find( { user_id: /bc/ } ) db.people.find( { user_id: { \$regex: /bc/ } } )
	SELECT COUNT(user_id) FROM people	db.people.count( { user_id: { \$exists: true } } ) db.people.find( { user_id: { \$exists: true } } ).count()
Update records	UPDATE people SET status = "C" WHERE age > 25	db.people.updateMany( { age: { \$gt: 25 } }, { \$set: { status: "C" } } )
	UPDATE people SET age = age + 3 WHERE status = "A"	db.people.updateMany( { status: "A" } , { \$inc: { age: 3 } } )
Delete Records	DELETE FROM people WHERE status = "D"	db.people.deleteMany( { status: "D" } )
	DELETE FROM people	db.people.deleteMany({})

## 5. Cách MongoDB hoạt động



Hình 5: Cách MongoDB hoạt động

MongoDB hoạt động dưới một tiến trình ngầm service, luôn mở một cổng (Cổng mặc định là 27017) để lắng nghe các yêu cầu truy vấn, thao tác từ các ứng dụng gửi vào sau đó mới tiến hành xử lý.

Mỗi một bản ghi của MongoDB được tự động gắn thêm một field có tên “\_id” thuộc kiểu dữ liệu ObjectId mà nó quy định để xác định được tính duy nhất của bản ghi này so với bản ghi khác, cũng như phục vụ các thao tác tìm kiếm và truy vấn thông tin về sau. Trường dữ liệu “\_id” luôn được tự động đánh index (chỉ mục) để tốc độ truy vấn thông tin đạt hiệu suất cao nhất.

Mỗi khi có một truy vấn dữ liệu, bản ghi được cache (ghi đệm) lên bộ nhớ Ram, để phục vụ lượt truy vấn sau diễn ra nhanh hơn mà không cần phải đọc từ ổ cứng.

Khi có yêu cầu thêm/sửa/xóa bản ghi, để đảm bảo hiệu suất của ứng dụng mặc định MongoDB sẽ chưa cập nhật xuống ổ cứng ngay, mà sau 60 giây MongoDB mới thực hiện ghi toàn bộ dữ liệu thay đổi từ RAM xuống ổ cứng.

### 2.2. Phương pháp nghiên cứu

#### 1. Tổng quan về phương pháp nghiên cứu lý thuyết:

Phương pháp nghiên cứu lý thuyết là việc thu thập, phân tích, và tổng hợp các tài liệu, công trình nghiên cứu có sẵn nhằm xây dựng cơ sở lý luận cho đề tài. Phương pháp này giúp hệ thống hóa các kiến thức liên quan, làm rõ các khái niệm và nguyên lý cơ bản, từ đó đưa ra các giả thuyết khoa học và định hướng cho nghiên cứu thực nghiệm.

Trong bối cảnh đề tài "Xây dựng hệ thống thi trắc nghiệm trực tuyến kỹ năng Reading dạng True/False/Not Given với cơ sở dữ liệu NoSQL", phương pháp nghiên

cứu lý thuyết đóng vai trò quan trọng trong việc hiểu rõ các vấn đề về giáo dục trực tuyến, dạng bài thi True/False/Not Given, cơ sở dữ liệu NoSQL, và công nghệ MERN.

## **2. Nội dung nghiên cứu lý thuyết**

### **Nghiên cứu về giáo dục trực tuyến**

Tìm hiểu về các đặc điểm, lợi ích và hạn chế của giáo dục trực tuyến, đặc biệt là trong việc phát triển kỹ năng đọc hiểu. Tham khảo các tài liệu về các hệ thống học trực tuyến hiện tại, từ đó phân tích hiệu quả của các hệ thống này trong việc hỗ trợ người dùng.

### **Dạng bài True/False/Not Given trong kiểm tra kỹ năng đọc hiểu**

Phân tích cấu trúc, yêu cầu và vai trò của dạng bài True/False/Not Given trong các kỳ thi tiếng Anh quốc tế như IELTS, TOEFL. Tìm hiểu các chiến lược làm bài và cách đánh giá kết quả để áp dụng vào hệ thống trực tuyến.

### **Ứng dụng cơ sở dữ liệu NoSQL:**

Nghiên cứu về khái niệm, cấu trúc, ưu điểm và hạn chế của cơ sở dữ liệu NoSQL. Tập trung vào MongoDB, một hệ thống NoSQL phổ biến, để hiểu rõ cách quản lý dữ liệu phi cấu trúc và hỗ trợ các ứng dụng web quy mô lớn.

### **Công nghệ MERN trong phát triển hệ thống web:**

Tìm hiểu về các thành phần của MERN (MongoDB, Express.js, React.js, Node.js) và cách chúng phối hợp với nhau để xây dựng các ứng dụng web hiện đại. Phân tích khả năng áp dụng MERN (MongoDB, Express.js, React.js, Node.js) vào việc phát triển hệ thống thi trắc nghiệm trực tuyến.

## **2.3. Tổng kết**

Phân cơ sở lý thuyết, lý luận và phương pháp nghiên cứu đã cung cấp một nền tảng khoa học vững chắc cho việc xây dựng hệ thống thi trắc nghiệm trực tuyến kỹ năng Reading dạng True/False/Not Given. Các lý thuyết về giáo dục trực tuyến, dạng bài kiểm tra True/False/Not Given, cơ sở dữ liệu NoSQL và công nghệ MERN đã được phân tích và làm rõ. Những lý thuyết này hỗ trợ việc thiết kế và triển khai hệ thống, đảm bảo tính hiệu quả và tính thực tiễn.

## CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

### 3.1. Các bước tiến hành:

#### Đối tượng sử dụng:

Thí sinh (người dùng chính):

- Muốn có một nền tảng dễ sử dụng, đăng ký nhanh chóng.
- Yêu cầu giao diện thân thiện để làm bài thi, xem lịch sử và kết quả bài thi.
- Cần có chức năng làm bài thi với thời gian giới hạn.

Quản trị viên (admin):

- Muốn theo dõi lịch sử làm bài của thí sinh để quản lý dễ dàng.
- Yêu cầu khả năng phân quyền để giới hạn các chức năng.
- Thống kê các quả và người dùng.

#### Yêu cầu chức năng (Functional Requirements):

Đăng ký/Đăng nhập:

- Người dùng có thể đăng ký bằng email và mật khẩu.
- Xác thực quyền truy cập bằng JWT để bảo mật.

Quản lý bài thi:

- Quản trị viên có thể tạo bài thi mới, thêm câu hỏi, và đặt thời gian giới hạn.
- Quản trị viên có thể sửa hoặc xóa bài thi.
- Thí sinh chỉ có quyền truy cập bài thi được cấp.

Làm bài thi:

- Hiện thị câu hỏi dạng True/False/Not Given với giới hạn thời gian.
- Tự động lưu kết quả làm bài và tính điểm.
- Hiện thị thời gian còn lại trong khi làm bài.

Lịch sử làm bài:

Thí sinh có thể xem danh sách các bài thi đã làm, bao gồm:

- Kết quả của bài thi mình vừa làm.
- Điểm số và thời gian hoàn thành.
- Đáp án đúng, sai nếu sai có giải thích bên dưới.

Quản lý người dùng:

- Quản trị viên có thể xem danh sách và kết quả tất cả thí sinh.
- Phân quyền giữa thí sinh và quản trị viên.

### **Yêu cầu phi chức năng (Non-functional Requirements):**

Hiệu năng:

Hệ thống có thể hỗ trợ đồng thời nhiều thí sinh làm bài thi cùng lúc.

Khả năng mở rộng:

Dễ dàng thêm tính năng mới (ví dụ: thêm loại câu hỏi mới hoặc dạng bài thi mới).

Tính bảo mật:

- Mật khẩu phải được mã hóa trước khi lưu trong cơ sở dữ liệu (dùng bcrypt).
- Dữ liệu bài thi phải được bảo vệ để tránh gian lận.

Khả năng sử dụng (Usability):

Giao diện trực quan, dễ sử dụng cho mọi đối tượng, kể cả những người không quen công nghệ.

Tính ổn định:

Đảm bảo hệ thống không bị treo khi có nhiều người truy cập.

### **3.2. Phân tích hệ thống**

Dữ liệu hệ thống được tổ chức và lưu trữ trên nền tảng MongoDB, với cấu trúc linh hoạt để quản lý thông tin người dùng, bài thi, và kết quả. Chẳng hạn, Collection Users lưu trữ thông tin cá nhân và phân loại người dùng thành thí sinh hoặc quản trị viên. Collection ReadingTests chứa nội dung bài thi, bao gồm danh sách câu hỏi và thời gian làm bài. Các lần làm bài được ghi nhận trong Collection Attempts, còn kết quả từng bài thi được lưu trong Collection TestResults.

Việc thiết kế này giúp đảm bảo dữ liệu được lưu trữ hợp lý, hỗ trợ tốt các yêu cầu truy vấn từ hệ thống, chẳng hạn như hiển thị danh sách bài thi, lưu kết quả hoặc theo dõi tiến độ người dùng.

### **3.3. Kiến trúc hệ thống**

Hệ thống được thiết kế theo kiến trúc Client-Server để đảm bảo sự tách biệt giữa giao diện người dùng (front-end) và xử lý logic (back-end).

#### **Front-end:**

Sử dụng React.js để xây dựng giao diện người dùng.

Các component chính bao gồm:

- Giao diện đăng nhập, đăng ký.
- Giao diện danh sách bài thi.

- Trang làm bài thi với hiển thị câu hỏi và thời gian còn lại.
- Trang hiển thị kết quả bài thi và lịch sử làm bài.

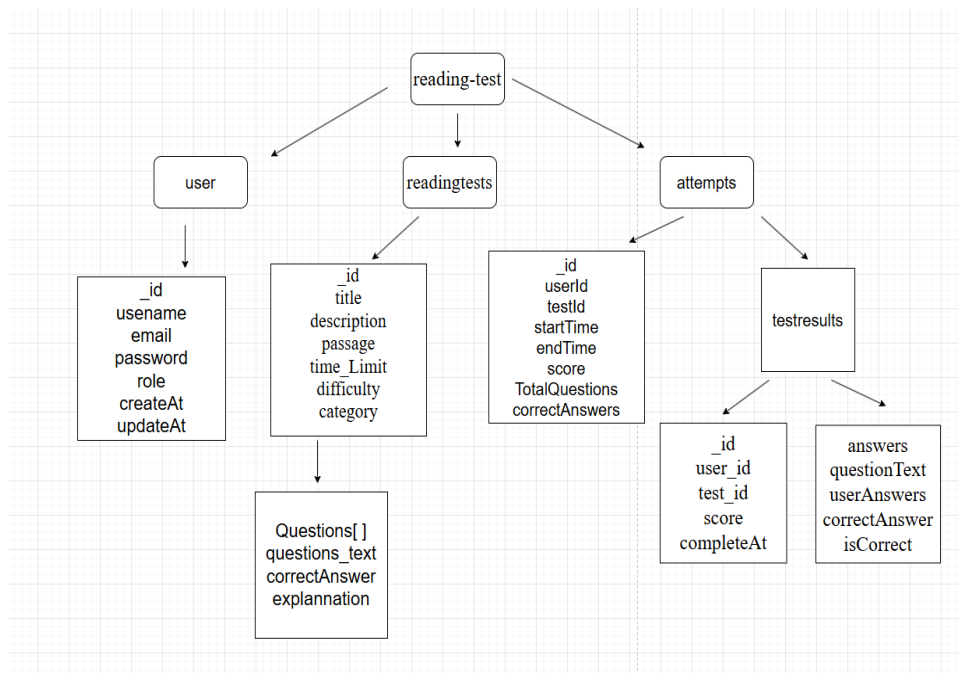
### Back-end:

Sử dụng Node.js và Express.js để xây dựng API RESTful, cung cấp các chức năng:

- Xác thực người dùng với JWT (JSON Web Token).
- Quản lý bài thi, câu hỏi, và kết quả.
- Truy xuất dữ liệu từ MongoDB và trả về client.

### Cơ sở dữ liệu:

MongoDB được sử dụng để lưu trữ dữ liệu linh hoạt. Các collections được thiết kế để hỗ trợ các truy vấn nhanh và hiệu quả.



Hình 6: Biểu diễn các collections

#### 1. Collection: users

Lưu trữ thông tin người dùng tham gia vào hệ thống, bao gồm cả người dùng thường và quản trị viên.

- Các trường chính:
  - `_id` (ObjectId): Mã định danh duy nhất của mỗi người dùng.
  - `username` (String): Tên người dùng.
  - `email` (String): Email dùng để đăng nhập.
  - `password` (String): Mật khẩu được mã hóa.



- role (String): Vai trò của người dùng (admin hoặc user).
- createdAt (Date): Ngày tài khoản được tạo.
- updatedAt (Date): Ngày thông tin tài khoản được cập nhật.

Kết nối:

Liên kết với attempts thông qua userId.

## 2. Collection: readingtests

Lưu trữ thông tin về các bài kiểm tra kỹ năng đọc hiểu, bao gồm tiêu đề, nội dung bài đọc, câu hỏi và đáp án đúng.

- Các trường chính:
  - \_id (ObjectId): Mã định danh duy nhất của mỗi bài kiểm tra.
  - title (String): Tiêu đề của bài kiểm tra.
  - description (String): Mô tả ngắn gọn về bài kiểm tra.
  - passage (String): Nội dung bài đọc.
  - timeLimit (Number): Thời gian làm bài (tính bằng phút).
  - difficulty (String): Mức độ khó (vd: easy, medium, hard).
  - category (String): Danh mục bài kiểm tra (vd: technology, environment).
  - questions (Array): Danh sách câu hỏi của bài kiểm tra.
    - question\_text (String): Nội dung câu hỏi.
    - correctAnswer (String): Đáp án đúng.
    - explanation (String): Giải thích chi tiết cho đáp án.

Kết nối:

Liên kết với attempts thông qua testId.

## 3. Collection: attempts

Ghi nhận từng lần làm bài của người dùng, bao gồm kết quả và thông tin liên quan.

Các trường chính:

- \_id (ObjectId): Mã định danh duy nhất của mỗi lần làm bài.
- userId (ObjectId): Liên kết đến người dùng trong collection users.
- testId (ObjectId): Liên kết đến bài kiểm tra trong collection readingtests.
- startTime (Date): Thời gian bắt đầu làm bài.
- endTime (Date): Thời gian kết thúc làm bài.

- score (Number): Điểm số đạt được.
- totalQuestions (Number): Tổng số câu hỏi trong bài kiểm tra.
- correctAnswers (Number): Số câu trả lời đúng.
- Kết nối:
  - Liên kết với users và readingtests.
  - Liên kết với testresults qua \_id (được lưu trong attempt\_id).

#### 4. Collection: testresults

Lưu trữ kết quả chi tiết từng câu trả lời của người dùng cho một lần làm bài cụ thể.

Các trường chính:

- \_id (ObjectId): Mã định danh duy nhất của kết quả bài kiểm tra.
- attempt\_id (ObjectId): Liên kết đến lần làm bài trong collection attempts.
- answers (Array): Danh sách các câu trả lời của người dùng.
  - questionText (String): Nội dung câu hỏi.
  - userAnswer (String): Câu trả lời của người dùng.
  - correctAnswer (String): Đáp án đúng.
  - isCorrect (Boolean): Xác định câu trả lời đúng hay sai.
- Kết nối:
  - Liên kết với attempts qua attempt\_id.

#### Luồng dữ liệu trong hệ thống:

##### 1. Người dùng (User):

Đăng ký tài khoản và đăng nhập để làm bài kiểm tra.

##### 2. Bài kiểm tra (Reading Test):

Người dùng chọn bài kiểm tra từ danh sách và bắt đầu làm bài.

##### 3. Lần làm bài (Attempt):

Khi người dùng bắt đầu làm bài, một bản ghi mới được tạo trong attempts để theo dõi tiến trình.

##### 4. Kết quả bài kiểm tra (Test Results):

Sau khi hoàn thành bài kiểm tra, câu trả lời của người dùng được lưu vào testresults, bao gồm so sánh với đáp án đúng.

##### 5. Thống kê:

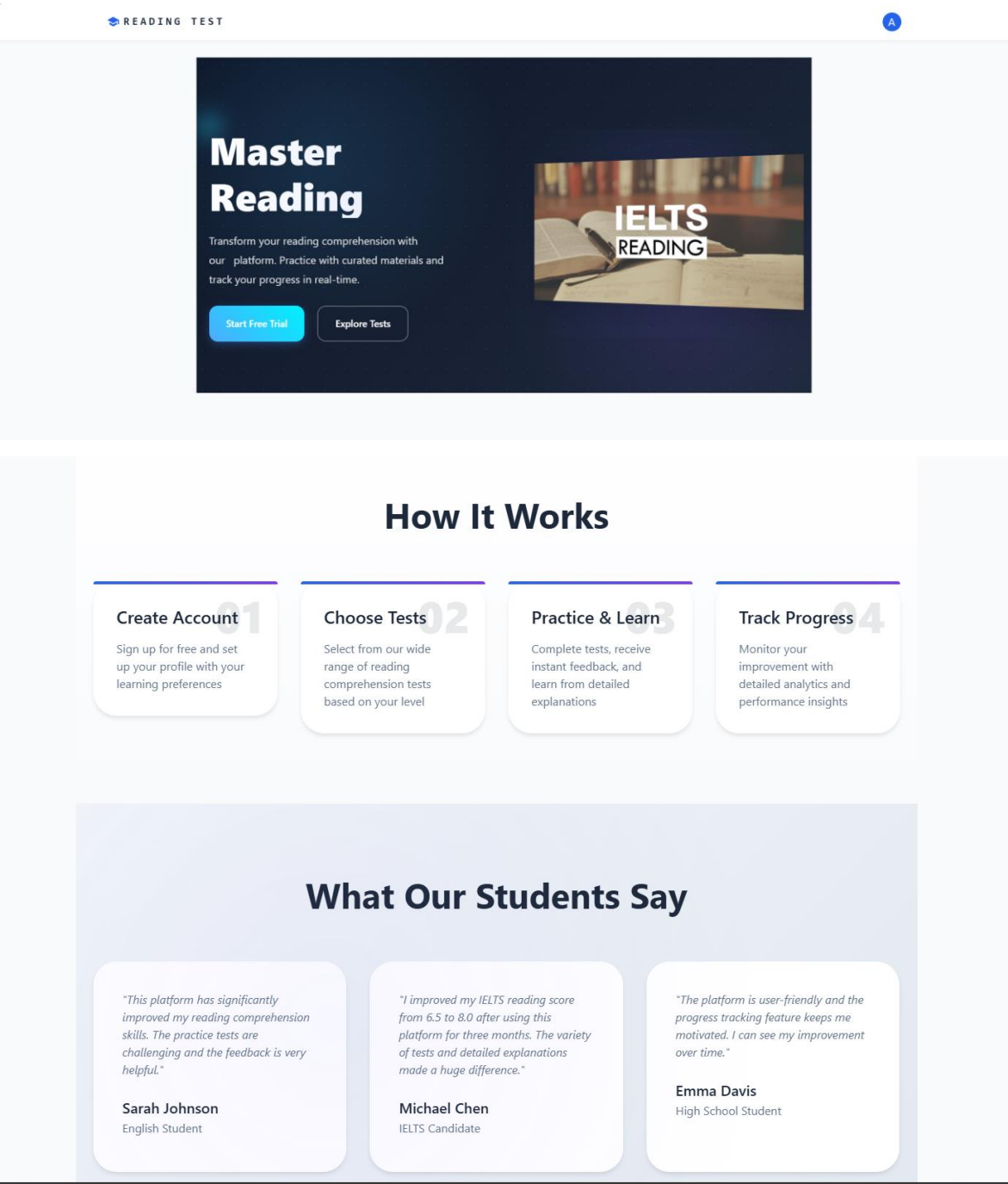
Quản trị viên có thể xem báo cáo tổng quan và chi tiết kết quả qua các collection.

## CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

### 4.1. Kết quả Đạt được

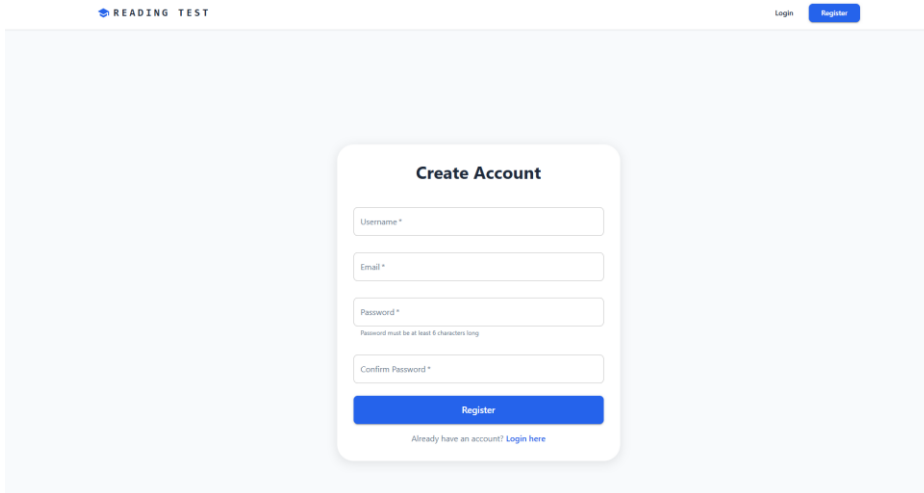
Kết quả Đạt được Phần Giao diện (Front-end):

- **Trang chủ:**



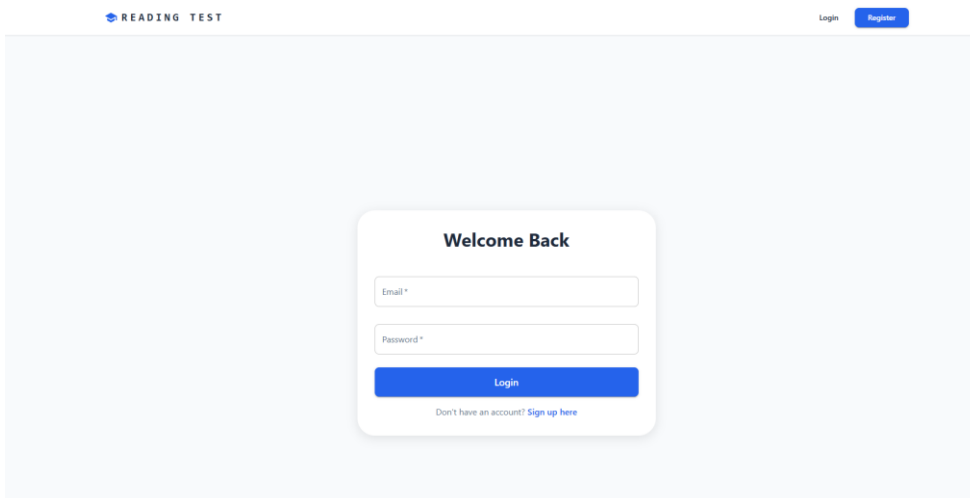
Hình 7: Giao diện trang chủ

- Giao diện trang chủ của hệ thống được thiết kế đơn giản, trực quan, đáp ứng đầy đủ các yêu cầu:
  - Chào đón người dùng: Hiển thị thông tin giới thiệu ngắn gọn về hệ thống, cùng với các hướng dẫn sử dụng cơ bản.
  - Điều hướng chính:
  - Nút truy cập nhanh đến các tính năng chính như Đăng nhập, Đăng ký, và Làm bài thi.
- **Trang đăng ký/đăng nhập:**

The image shows a web browser window with the title 'READING TEST'. In the top right corner, there are links for 'Login' and 'Register'. The main content area features a 'Create Account' form. The form has four input fields: 'Username \*', 'Email \*', 'Password \*', and 'Confirm Password \*'. Below the password field, there is a small text note: 'Password must be at least 8 characters long'. At the bottom of the form is a blue 'Register' button. Below the button, there is a link that says 'Already have an account? Login here'.

Hình 8: Giao diện trang đăng ký

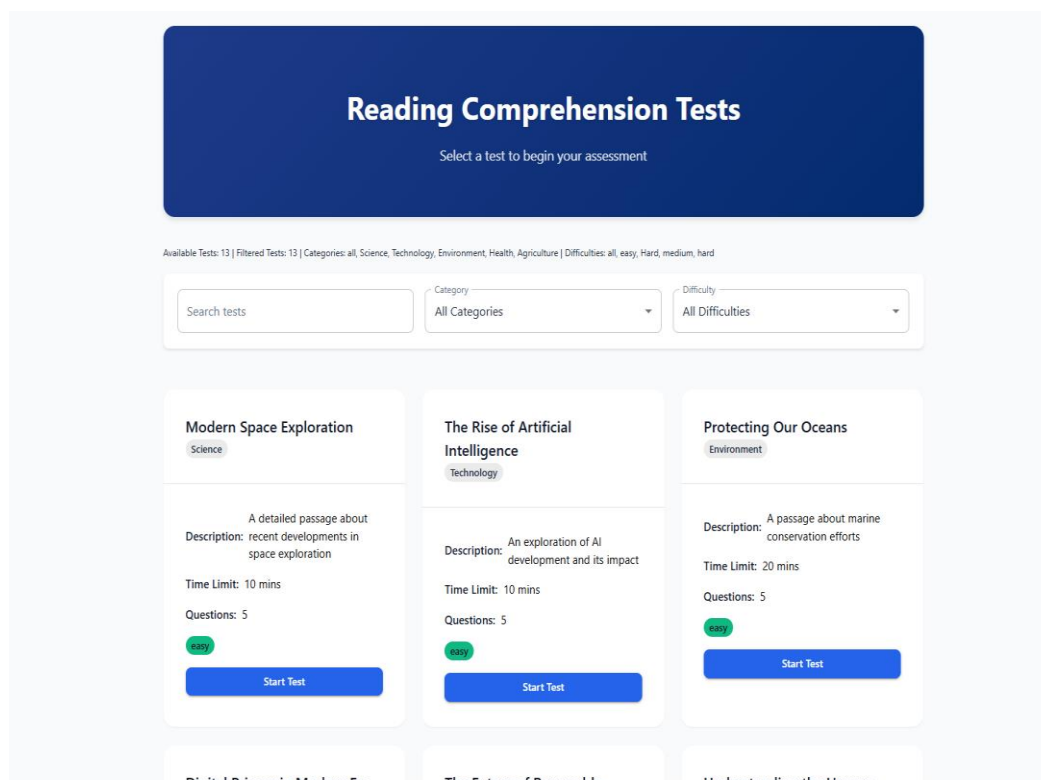
Trước khi vào thi người dùng phải đăng ký tài khoản cá nhân bằng cách điền vào các ô trong form: Username, Email, Password và Confirm Password.

The image shows a web browser window with the title 'READING TEST'. In the top right corner, there are links for 'Login' and 'Register'. The main content area features a 'Welcome Back' login form. The form has two input fields: 'Email \*' and 'Password \*'. At the bottom of the form is a blue 'Login' button. Below the button, there is a link that says 'Don't have an account? Sign up here'.

Hình 9: Giao diện trang đăng nhập

Sau khi đã đăng ký tài khoản trong trang đăng ký người dùng nhập tài khoản vừa tạo và trang đăng nhập.

- **Trang danh sách bài thi:**



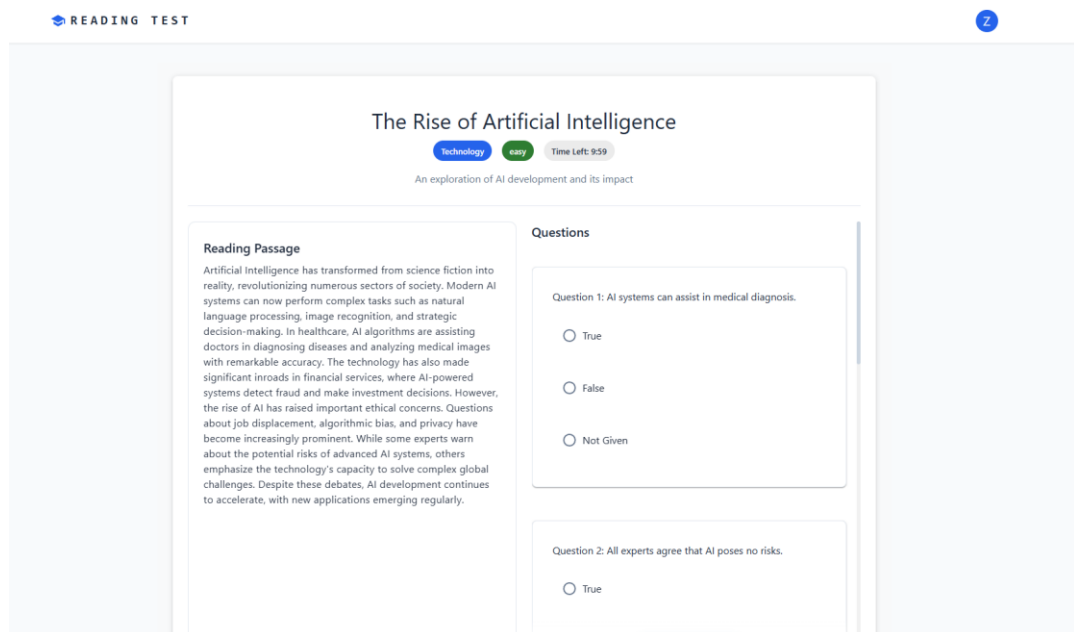
Hình 10: Giao diện trang danh sách bài thi

Trang danh sách bài thi hiển thị toàn bộ các bài kiểm tra hiện có trên hệ thống, được tổ chức theo cách dễ tìm kiếm và truy cập.

Các Thành Phần Chính:

- Tiêu đề và mô tả:
- Phần tiêu đề lớn với dòng chữ "Reading Comprehension Tests" để thông báo chức năng của trang.
- Phần mô tả ngắn gọn phía dưới để hướng dẫn thí sinh chọn bài thi phù hợp.
- Bộ lọc và tìm kiếm:
- Thanh tìm kiếm: Cho phép nhập từ khóa để tìm nhanh bài thi theo tên.
- Bộ lọc theo danh mục: Hiển thị danh sách các danh mục bài thi như Khoa học (Science), Công nghệ (Technology), Môi trường (Environment), v.v.
- Bộ lọc theo độ khó: Bao gồm các tùy chọn như dễ (easy), trung bình (medium), và khó (hard).
- Danh sách bài thi:
- Mỗi bài thi được hiển thị dưới dạng thẻ riêng biệt với các thông tin quan trọng:

- Tên bài thi: Ví dụ, "Modern Space Exploration."
- Danh mục: Được đánh dấu rõ ràng (Science, Technology, Environment).
- Mô tả: Tóm tắt nội dung bài thi.
- Thời gian làm bài: Ví dụ, "10 phút."
- Số câu hỏi: Số lượng câu hỏi trong bài thi.
- Mức độ khó: Được hiển thị bằng nhãn màu sắc (dễ, trung bình, khó).
- Nút "Start Test" (Bắt đầu làm bài): Cho phép thí sinh bắt đầu bài thi ngay lập tức.
- **Giao diện trang làm bài thi:**



Hình 11: Giao diện trang làm bài thi

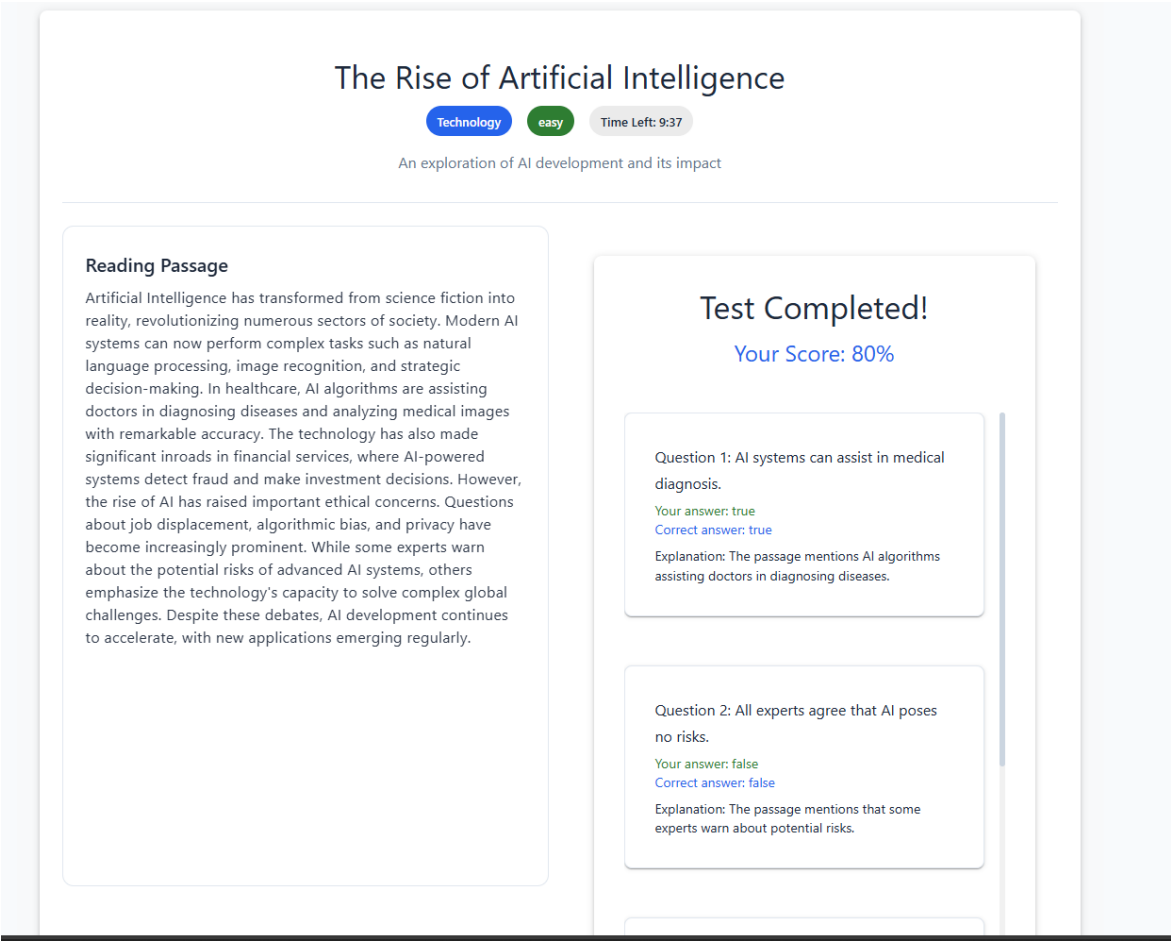
Trang làm bài thi được thiết kế để cung cấp trải nghiệm thân thiện và tập trung cho người dùng trong quá trình thực hiện bài kiểm tra đọc hiểu.

Các Thành Phần Chính:

- Tiêu đề Bài Thi:
- Hiển thị tên bài kiểm tra, như "The Rise of Artificial Intelligence."
- Kèm theo các thông tin:
- Thể loại: Ví dụ: "Technology."
- Độ khó: Ví dụ: "Easy."

- Thời gian còn lại: Được hiển thị dưới dạng đếm ngược, giúp thí sinh quản lý thời gian.
- Đoạn Văn Đọc Hiểu (Reading Passage):
- Hiển thị nội dung bài đọc để người dùng tham khảo.
- Kích thước hộp văn bản tối ưu, có thể cuộn để đọc hết nội dung.
- Danh Sách Câu Hỏi:
- Mỗi câu hỏi bao gồm:
- Câu hỏi: Ví dụ: "Question 1: AI systems can assist in medical diagnosis."
- Tùy chọn đáp án: Các tùy chọn "True," "False," và "Not Given" với radio button để chọn.
- Giao diện câu hỏi rõ ràng, dễ thao tác.
- Nút Gửi Bài (Submit Test):
- Nút màu xanh với nhãn "Submit Test," nổi bật và rõ ràng.
- Khi nhấp, bài làm sẽ được gửi để chấm điểm hoặc lưu trữ.
- Đặc Điểm Thiết Kế
- Tính trực quan: Tách biệt rõ ràng giữa đoạn văn đọc hiểu và danh sách câu hỏi.
- Tương tác dễ dàng: Người dùng có thể chọn đáp án nhanh chóng mà không cần di chuyển nhiều.
- Thân thiện với người dùng: Phần thời gian đếm ngược giúp thí sinh tập trung và tránh bị quên thời gian.
- Trải Nghiệm Người Dùng
- Dễ sử dụng: Mọi thông tin cần thiết đều được hiển thị trong một trang, không gây rối mắt.
- Hỗ trợ tốt: Đoạn văn đọc hiểu dễ dàng truy cập và các câu hỏi được sắp xếp hợp lý.
- Đáp ứng nhu cầu: Người dùng có thể hoàn thành bài kiểm tra trong thời gian quy định mà không gặp khó khăn.

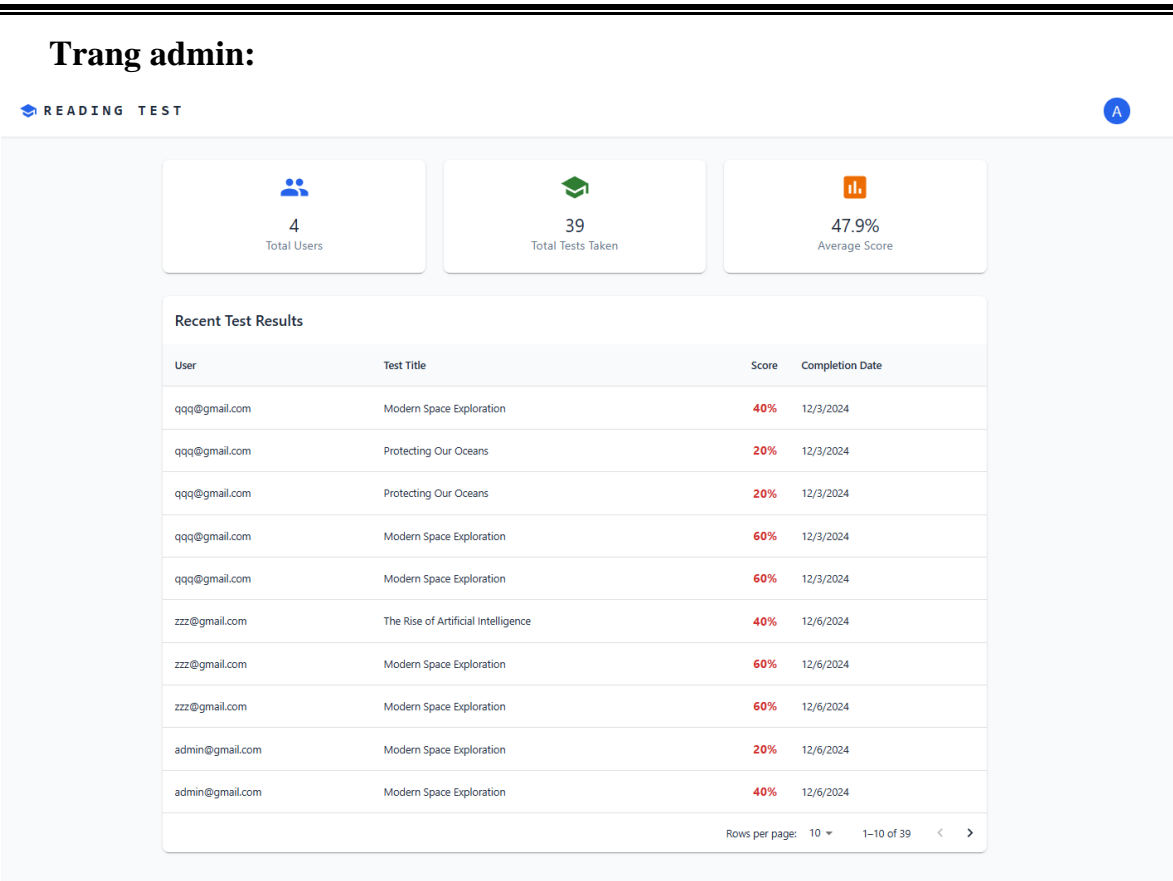
Trang chấm điểm:



Hình 12: Giao diện chấm điểm

- Hiện thị tên bài thi, điểm số đạt được và thời gian hoàn thành.





Hình 13: Giao trang trang admin

Hiện thị các chỉ số quan trọng:

- Total Users: Tổng số người dùng (4).
- Total Tests Taken: Tổng số bài kiểm tra đã được thực hiện (39).
- Average Score: Điểm trung bình (%), ví dụ: 47.9%.

Recent Test Results (Bảng kết quả gần đây):

- Cột thông tin:
- User: Email của người dùng (ví dụ: qqq@gmail.com).
- Test Title: Tên bài kiểm tra (Modern Space Exploration, Protecting Our Oceans...).
- Score: Điểm đạt được (phần trăm).
- Completion Date: Ngày hoàn thành bài kiểm tra.
- Tính năng phân trang:
- Rows per page: Cho phép chọn số hàng hiển thị (ví dụ: 10, 20, 50).
- Hiện thị trạng thái phân trang (ví dụ: 1–10 of 39).

## CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 5.1. Kết luận:

Đồ án đã hoàn thành với các kết quả quan trọng, đáp ứng mục tiêu đặt ra. Hệ thống web app thi trắc nghiệm reading true/false/not given đã được xây dựng thành công, cung cấp nền tảng học tập trực tuyến nhằm hỗ trợ người dùng nâng cao kỹ năng đọc hiểu tiếng Anh. Hệ thống đã hoàn thiện các chức năng chính, bao gồm quản lý người dùng, tổ chức thi trắc nghiệm, phản hồi kết quả và theo dõi tiến độ học tập.

Về mặt giao diện, hệ thống được thiết kế đơn giản, thân thiện với người dùng và tối ưu hóa hiệu năng, mang lại trải nghiệm sử dụng mượt mà. Với việc sử dụng MongoDB làm công cụ lưu trữ, hệ thống đảm bảo tốc độ xử lý nhanh, khả năng mở rộng linh hoạt và lưu trữ dữ liệu hiệu quả.

Đồ án đã đóng góp mới trong việc ứng dụng công nghệ web để xây dựng môi trường học tập trực tuyến, hỗ trợ người học tự đánh giá năng lực và cải thiện kết quả học tập. Đồng thời, các tính năng như phản hồi tức thời và công cụ theo dõi tiến độ đã giúp người học nhận biết điểm mạnh, điểm yếu của bản thân để có kế hoạch học tập hiệu quả hơn.

Những đề xuất mới cho hệ thống bao gồm việc bổ sung chức năng gợi ý bài thi phù hợp với trình độ người dùng, tích hợp chế độ luyện tập theo chủ đề cụ thể và cải tiến tính năng thống kê để cá nhân hóa trải nghiệm học tập.

### 5.2. Hướng phát triển

Dựa trên kết quả đạt được, hệ thống có thể được phát triển theo nhiều hướng để tăng cường hiệu quả và mở rộng phạm vi ứng dụng.

Thứ nhất, mở rộng ngân hàng câu hỏi bằng cách bổ sung nhiều chủ đề và cấp độ khó khác nhau. Việc cập nhật thường xuyên các câu hỏi mới sẽ giúp người học có thêm nội dung đa dạng để rèn luyện.

Thứ hai, phát triển các tính năng cá nhân hóa trải nghiệm học tập. Điều này bao gồm việc tích hợp thuật toán gợi ý bài thi dựa trên lịch sử làm bài và năng lực của người học, giúp họ lựa chọn bài thi phù hợp với mục tiêu học tập.

Thứ ba, tích hợp tính năng xã hội hóa vào hệ thống. Người dùng có thể chia sẻ kết quả và so sánh tiến độ học tập với bạn bè, hoặc tham gia vào các nhóm học tập để nâng cao sự tương tác và động lực học tập.

Thứ tư, ứng dụng công nghệ AI vào hệ thống. AI có thể được sử dụng để phân tích dữ liệu, đánh giá chi tiết năng lực của người học và đưa ra lộ trình học tập tối ưu. Điều này sẽ nâng cao hiệu quả học tập và tăng cường giá trị của hệ thống.

Thứ năm, phát triển ứng dụng di động là một hướng đi quan trọng, cho phép người dùng truy cập và luyện tập mọi lúc, mọi nơi. Điều này sẽ đáp ứng nhu cầu linh hoạt của người học hiện đại.

Những hướng phát triển này sẽ đảm bảo hệ thống tiếp tục đáp ứng hiệu quả nhu cầu học tập hiện tại, đồng thời mở rộng khả năng phục vụ cho các nhu cầu đa dạng và phức tạp hơn trong tương lai.

## DANH MỤC TÀI LIỆU THAM KHẢO

- [1]. Cao Lê Viết Tiến. (n.d.). *NoSQL là gì? Cơ sở dữ liệu NoSQL là gì?*. Vietnix. Truy cập từ <https://vietnix.vn/nosql-la-gi/#co-so-du-lieu-nosql-la-gi>
- [2]. Meier, A., & Kaufmann, M. (2019). *SQL & NoSQL Databases*. Springer Fachmedien Wiesbaden: Springer Vieweg.
- [3]. Celko, J. (2014). *Joe Celko's Complete Guide to NoSQL*. Amsterdam: Elsevier/Morgan Kaufmann.
- [4]. Sarrion, E. (2022). *Web Development with JavaScript: Building Web Applications*. Birmingham, UK: Packt Publishing Ltd.
- [5]. Bradshaw, S. (2020). *Database Management with Open Source Software: Object-Oriented Databases*. Beijing: O'Reilly Media, Inc.
- [6]. Phú, L.T. (2021). *MERN Stack là gì? Mức độ phổ biến như thế nào?*. Truy cập ngày 17 tháng 12 năm 2024, từ <https://phu.lt/mern-stack-la-gi-muc-djo-pho-bien-nhu-the-nao/>
- [7]. Voca.vn. (n.d.). *Phương pháp đọc hiểu tiếng Anh hiệu quả*. Truy cập ngày 17 tháng 12 năm 2024, từ <https://www.voca.vn/blog/phuong-phap-doc-hieu-tieng-anh-hieu-qua-1738>
- [8]. Twinkl.com. (n.d.). *Cách luyện kỹ năng đọc hiểu tiếng Anh hiệu quả với 9 bước*. Truy cập ngày 18 tháng 12 năm 2024, từ <https://www.twinkl.com/blog/cach-luyen-ky-nang-doc-hieu-tieng-anh-hieu-qua-voi-9-buoc>
- [9]. IELTS IDP. (n.d.). *A closer look at True/False/Not Given*. Truy cập ngày 19 tháng 12 năm 2024, từ <https://ielts.idp.com/vietnam/prepare/article-a-closer-look-at-true-false-not-given>
- [10]. British Council Việt Nam. (n.d.). *Chín lý do tại sao bạn nên học tiếng Anh trực tuyến*. Truy cập ngày 25 tháng 12 năm 2024, từ <https://www.britishcouncil.vn/hoc-tieng-anh/tieng-anh-nguoi-lon/kinh-nghiem/chin-ly-do-ban-nen-hoc-tieng-anh-truc-tuyen>

## **PHỤ LỤC**