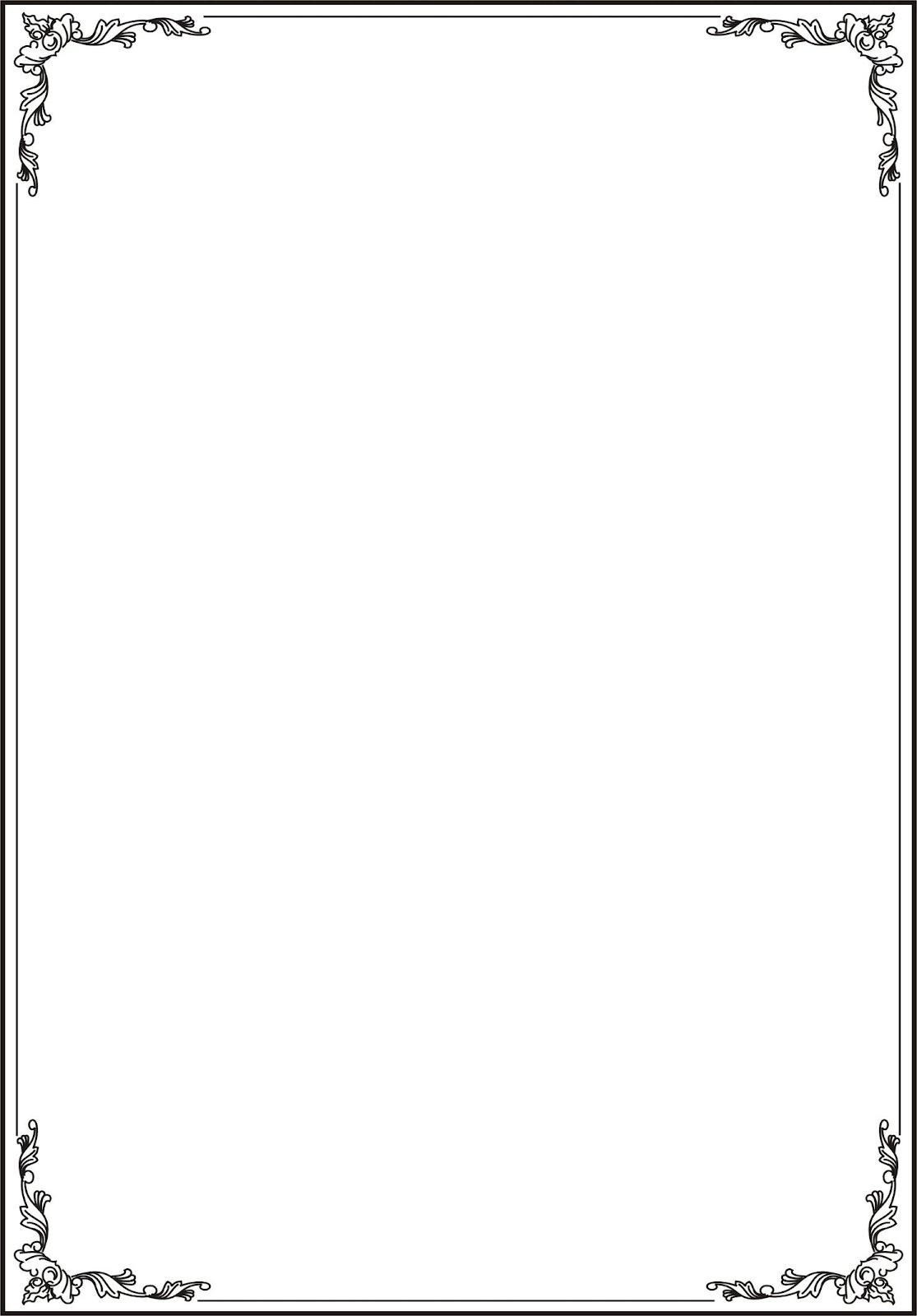
# ĐẠI HỌC QUỐC GIA HÀ NỘI TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN



# KHOA TOÁN - CƠ - TIN HỌC

## ---oooOooo---

# BÁO CÁO THỰC TẬP

# MÔN PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG THÔNG TIN

**Đề tài:**

# MANUAL TESTING VÀ HỆ THỐNG AUTHEN & AUTHOR

**Đơn vị thực tập : Công ty TNHH LG CNS Việt Nam**

**Giảng viên : TS. Nguyễn Thị Bích Thủy**

**TS. Vũ Tiến Dũng Người hướng dẫn : Nguyễn Anh Thương**

**Sinh viên thực tập : Nguyễn Mạnh Trí - 20001984**

Hà Nội - 2024

## Lời cảm ơn

Hai tháng thực tập ngắn ngủi là cơ hội cho em tổng hợp và hệ thống hóa lại những kiến thức đã học, đồng thời kết hợp với thực tế để nâng cao kiến thức chuyên môn. Tuy chỉ có hai tháng thực tập, nhưng qua quá trình thực tập, em đã được mở rộng tầm nhìn và tiếp thu rất nhiều kiến thức thực tế. Từ đó em nhận thấy, việc cọ xát thực tế là vô cùng quan trọng – nó giúp sinh viên xây dựng nền tảng lý thuyết được học ở trường vững chắc hơn. Trong quá trình thực tập, từ chỗ còn bỡ ngỡ cho đến thiếu kinh nghiệm, em đã gặp phải rất nhiều khó khăn nhưng với sự giúp đỡ của các anh chị trong Công ty đã giúp em có được những kinh nghiệm quý báu để hoàn thành tốt kì thực tập này cũng như viết lên bài báo cáo cuối kỳ. Em xin chân thành cảm ơn.

Lời cảm ơn đầu tiên em xin gửi đến ban lãnh đạo cùng các phòng ban, anh chị trong Công ty LG CNS Việt Nam– đơn vị đã tiếp nhận và nhiệt tình tạo mọi điều kiện thuận lợi cho em tiếp cận thực tế sản xuất và nắm bắt quy trình công nghệ. Em cũng xin gửi lời cảm ơn chân thành đến cô Nguyễn Thị Bích Thuỷ và thầy Vũ Tiến Dũng đã tạo cơ hội cho em được đi thực tập trong kỳ hè này, và đặc biệt đã hỗ trợ cho em trong quá trình phỏng vấn thành công để được thực tập công ty LG CNS Việt Nam để em được học hỏi thêm từ các anh chị trong công ty.

Vì thời gian và kiến thức còn hạn hẹp nên bài báo cáo không thể tránh khỏi những thiếu sót, rất mong sự góp ý của Công ty, thầy cô và các bạn, để em rút kinh nghiệm và hoàn thành tốt hơn. Em xin chân thành cảm ơn!

# MỤC LỤC

[Lời cảm ơn 2](#_TOC_250005)

[MỤC LỤC 3](#_TOC_250004)

1. [Giới thiệu về công ty thực tập 5](#_TOC_250003)
   1. Giới thiệu tổng quan 5
   2. Vị trí thực tập 5
2. Visual Studio Code (VSC) 5
3. Hệ quản trị cơ sở dữ liệu MySQL 6
   * Ưu điểm 6
   * Nhược điểm 6
4. Chat GPT 6
5. Github Copilot 7
6. Maildev 7
   1. Đề tài thực tập 7

3.1Giới thiệu về đề tài: 7

* + - Mô Hình MVC 7
    - Java 9
    - Spring Boot 9
    - JS 13

3.2Front - end 14

4.Lịch làm việc……………………………………………………………14

1. [Nội dung thực tập 14](#_TOC_250002)
   1. Manual tester 14
      1. Quy trình tìm hiểu user story 14
      2. Quy trình viết test case 14
      3. Quy trình log bug và evidence 14
      4. Quy trình quản lý tiến độ fix bug…………………………………….
   2. Hệ thống xác thực và phân quyền người dùng sử dụng OAuth 2.0 15
   3. Mục đích 15
      * Chức năng đăng ký 15
      * Chức năng đăng nhập 15
      * Cập nhật Proﬁle 15
      * Chức năng quên mật khẩu 16
      * Xem các topic và các từ vựng thuộc mỗi topic 16
      * Làm bài thi 16
   4. .Flow diagram 17
      * Chức năng tạo bài thi(câu hỏi) 17
      * Chức năng quản lý người dùng 17
      * Chức năng quản lý topic 17
      * Chức năng quản lý câu hỏi 17
      * Chức năng quản lý từ 17

2.3 Exception

* 1. Thiết kế cơ sở dữ liệu 18

1. [Kết quả 20](#_TOC_250001)
2. Test cases 20
   1. quản lý topic 20
   2. Quản lý word 21
   3. Quản lý câu hỏi 22
   4. Quản lý người dùng, chỉnh sửa thông tin người dùng 23
3. OAuth 2.0 login server 24
   1. Giao diện đăng ký 24
   2. Giao diện đăng nhập 25
   3. Giao diện chính 25
   4. Quên mật khẩu 26
   5. Cập nhật Proﬁle 27
   6. Learning(demo) 28
   7. Exam 30
4. [Tài liệu tham khảo 33](#_TOC_250000)

## Giới thiệu chung.

* 1. Đề tài.

Trang web học từ vựng tiếng Anh ra đời với mục tiêu chắt lọc và tối ưu hóa quá trình học từ vựng, là một phần không thể thiếu trong việc nâng cao khả năng giao tiếp và hiểu biết tiếng Anh. Việc xây dựng trang web này bắt nguồn từ nhận thức sâu sắc về vai trò quan trọng của từ vựng trong việc tiếp cận và vượt qua rào cản ngôn ngữ. Với mục đích giúp người học tiếp cận ngôn ngữ một cách hiệu quả hơn, trang web tập trung vào việc cung cấp danh sách từ vựng đa dạng, phong phú theo các chủ đề khác nhau, từ những từ cơ bản cho đến những từ vựng nâng cao. Sự tiện lợi của trang web học từ vựng tiếng Anh thể hiện qua khả năng truy cập linh hoạt từ mọi nơi và thiết bị. Không còn cản trở thời gian và địa điểm, người học có thể dành thời gian học tập linh hoạt theo sở thích và lịch trình riêng của mình. Hơn nữa, tính tương tác của trang web cũng là một yếu tố quan trọng, cho phép người học kiểm tra kiến thức thông qua các bài kiểm tra, câu hỏi trắc nghiệm hoặc bài tập thú vị. Bằng cách liên tục cập nhật nội dung, theo dõi xu hướng ngôn ngữ và tích hợp các công nghệ học tập tiên tiến, trang web học từ vựng tiếng Anh hứa hẹn mang đến môi trường học tập thú vị, hiệu quả và phù hợp với mọi trình độ ngôn ngữ. Từ việc xây dựng nền tảng vững chắc cho người mới bắt đầu, đến việc mở rộng cơ hội mở rộng kiến thức cho những người đam mê tiếng Anh, trang web này đóng góp quan trọng vào việc làm cho quá trình học từ vựng trở nên thú vị và tiến bộ hơn bao giờ hết.

* 1. Môi trường và công cụ xử lý bài toán.

Công cụ chính được sử dụng là Visual Studio Code và MySQL

1. Visual Studio Code (VSC).

Visual Studio Code (VSC) có khả năng hỗ trợ gần như toàn bộ ngôn ngữ lập trình phổ biến hiện nay như: Python, JavaScript, HTML, CSS,...và còn nhiều hơn thế thông qua các Extension, với sự hỗ trợ đông đảo- Visual Studio Code (VSC) có hệ thống phím tắt sinh động, trực quan giúp thao tác trở nên nhanh chóng hơn. Ngoài ra các tiện ích trên VSC sẽ tự động nhắc mã, phát hiện đoạn mã không đầy đủ hoặc tự động bổ xung cú pháp khi còn thiếu sẽ giúp chúng ta dễ dàng trong việc thực hiện code hơn.

1. Hệ quản trị cơ sở dữ liệu MySQL.

* Ưu điểm:
  + Miễn phí và mã nguồn mở: MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở, có nghĩa là bạn có thể sử dụng nó mà không cần phải trả bất kỳ khoản phí nào. Điều này rất hữu ích đối với các dự án có nguồn tài nguyên hạn chế.
  + Phù hợp với dự án nhỏ và trung bình: MySQL là lựa chọn phổ biến cho các dự án cỡ nhỏ và trung bình với tải lưu lượng dữ liệu không quá lớn. Nó dễ dàng triển khai và quản lý.
  + Tốc độ xử lý nhanh: MySQL cung cấp hiệu suất tốt với khả năng xử lý truy vấn nhanh chóng, đặc biệt là trong các tình huống tải nhẹ hoặc trung bình.
  + Hỗ trợ cộng đồng lớn: Có một cộng đồng rộng lớn và nhiều tài liệu hỗ trợ cho MySQL, điều này có nghĩa bạn có thể tìm kiếm và tìm thấy giải pháp cho nhiều vấn đề khác nhau.
* Nhược điểm:
  + Khả năng mở rộng hạn chế: MySQL có thể gặp khó khăn trong việc xử lý các ứng dụng lớn hoặc có tải lưu lượng dữ liệu cao. Khả năng mở rộng của nó có thể bị giới hạn so với một số hệ quản trị cơ sở dữ liệu khác.
  + Hạn chế về tích hợp: Trong một số trường hợp, việc tích hợp MySQL với các công nghệ mới và phức tạp có thể khó khăn hơn do thiếu một số tính năng hiện đại.
  + Quản lý phức tạp: Trong khi MySQL dễ dàng triển khai và quản lý đối với các dự án nhỏ, nhưng khi dự án phức tạp hơn, việc quản lý cơ sở dữ liệu có thể trở nên phức tạp và đòi hỏi kiến thức sâu rộng về cơ sở dữ liệu.
  + Hiệu năng yếu trong một số tình huống: Trong khi MySQL có thể xử lý tốt trong tải nhẹ và trung bình, nó có thể gặp khó khăn trong môi trường có tải lưu lượng dữ liệu cao hoặc yêu cầu xử lý phức tạp.

1. *Chat GPT.*
   * Hỏi đáp các thắc mắc và hỗ trợ trong việc sửa lỗi viết code.
2. Github Copilot.
   * Hỗ trợ viết code trong quá trình thực hiện dự án.
3. Maildev.
   * MailDev là một công cụ phát triển và kiểm tra email trong môi trường phát triển và kiểm thử ứng dụng. Nó giúp các nhà phát triển kiểm tra việc gửi và nhận email trong các ứng dụng mà họ đang phát triển mà không cần phải thực sự sử dụng hệ thống email thực sự. Điều này giúp họ tránh việc gửi email thực sự đến địa chỉ thư điện tử thật và giữ quá trình phát triển và kiểm thử tách biệt.
   * MailDev cung cấp một máy chủ SMTP (Simple Mail Transfer Protocol) cục bộ để chấp nhận email được gửi từ ứng dụng. Nó cũng cung cấp một giao diện người dùng web cho phép người dùng xem các email được gửi đến và nhận từ ứng dụng của họ. Người dùng có thể xem nội dung email, tiêu đề, tên người gửi và các thông tin khác liên quan.
   1. Công nghệ sử dụng:
   2. Back - end.
      * Mô Hình MVC

Mô hình MVC (viết tắt của Model-View-Controller) là một mẫu kiến trúc phần mềm để tạo lập giao diện người dùng trên máy tính. MVC chia

một ứng dụng thành ba phần tương tác được với nhau để tách biệt giữa cách thức mà thông tin xử lý nội hàm và phần thông tin được trình bày, tiếp nhận từ phía người dùng.

Khi sử dụng đúng cách, MVC giúp cho người phát triển phần mềm cô lập các nguyên tắc nghiệp vụ và giao diện người dùng một cách rõ ràng hơn. Phần mềm phát triển theo MVC tạo nhiều thuận lợi cho việc bảo trì vì các nguyên tắc nghiệp vụ và giao diện ít liên quan đến nhau.

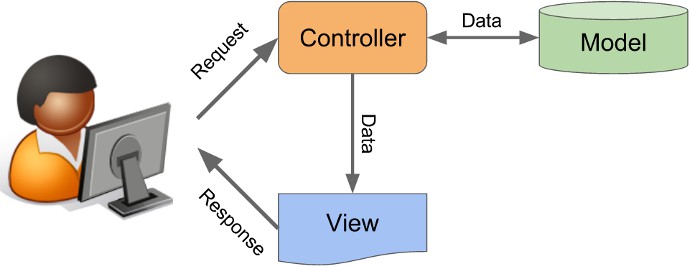
Trong MVC, gồm 3 thành phần chính:

+ Model: chứa cấu trúc dữ liệu có chức năng lưu trữ toàn bộ các thông tin dữ liệu của một ứng dụng. Trong mô hình MVC thì Model đóng vai trò kết nối cho 2 thành phần View và Controller.

+ View: Là giao diện người dùng, chứa các thành phần tương tác với người dùng như menu, button, image, text, và là nơi nhận

dữ liệu từ controller để hiển thị cho người dùng.

+ Controller: là thành phần trung gian giữa Model và View, đảm nhận vai cho tiếp nhận yêu cầu từ người dùng thông qua Model để lấy dữ liệu sau đó thông qua View hiển thị cho người dùng.



Ưu, nhược điểm của mô hình MVC:

- Ưu điểm:

+ Tách biệt logic: Mô hình MVC giúp tách biệt rõ ràng giữa logic ứng dụng, giao diện người dùng và dữ liệu. Điều này làm cho mã nguồn dễ dàng quản lý, bảo trì và mở rộng hơn.

+ Tái sử dụng: Tách biệt giữa Model, View và Controller cho phép tái sử dụng thành phần một cách hiệu quả. Ví dụ, có thể thay đổi giao diện người dùng mà không cần thay đổi logic xử lý dữ liệu.

+ Phân chia công việc: Mô hình MVC giúp phân chia công việc giữa các thành viên trong nhóm phát triển. Những người làm việc trên giao diện người dùng có thể tập trung vào việc thiết kế và hiển thị dữ liệu một cách tốt nhất, trong khi những người làm việc trên logic ứng dụng có thể tập trung vào xử lý dữ liệu.

+ Dễ dàng kiểm thử: Với việc tách biệt logic, bạn có thể dễ dàng kiểm thử từng phần riêng biệt của ứng dụng. Điều này giúp tăng tính ổn định và chất lượng của mã nguồn.

- Nhược điểm:

+ Phức tạp: Mô hình MVC có thể tạo ra sự phức tạp trong việc phát triển ứng dụng nhỏ hoặc đơn giản. Đôi khi, sự tách biệt quá mức có thể làm cho việc theo dõi và hiểu mã nguồn trở nên khó khăn hơn.

+ Khả năng lỗi: Với nhiều lớp và tầng tương tác, khả năng xảy ra lỗi có thể tăng lên. Nếu không được quản lý cẩn thận, điều này có thể dẫn đến việc khó xác định nguồn gốc của lỗi.

+ Quản lý tập trung: Trong mô hình MVC, Controller có thể trở nên quá phức tạp và quản lý nhiều logic xử lý khác nhau. Điều này có thể gây khó khăn trong việc bảo trì và mở rộng.

+ Hiệu suất: Mô hình MVC có thể gây ra một số tốn thời gian vì việc chuyển đổi giữa các tầng và tầng tương tác. Điều này có thể ảnh hưởng đến hiệu suất của ứng dụng trong những trường hợp đặc biệt.

Tóm lại, mô hình MVC có nhiều ưu điểm khi giúp tách biệt logic và giúp quản lý dự án dễ dàng hơn. Tuy nhiên, nó cũng có nhược điểm khi có thể gây ra sự phức tạp và khả năng xảy ra lỗi, đặc biệt trong các dự án nhỏ.

* Java.
  + là một ngôn ngữ lập trình phổ biến với khả năng chạy trên nhiều nền tảng khác nhau.
  + Java hỗ trợ lập trình hướng đối tượng, bảo mật cao, và có cộng đồng lớn. Nó được sử dụng rộng rãi trong phát triển ứng dụng máy tính, di động, web và nhiều lĩnh vực công nghệ khác.
* Spring Boot.
  + Spring framework

+ Spring Ecosystem: Spring Web Framework là một phần của hệ sinh thái Spring, bao gồm rất nhiều dự án và công nghệ liên quan như Spring Security, Spring Data, Spring Boot, Spring Cloud, và nhiều dự án khác.

+ Dependency Injection Spring Web Framework sử dụng phương pháp tiêm zạ để quản lý và kết nối các thành phần của ứng dụng. Điều này giúp tạo ra các ứng dụng có mã nguồn sạch sẽ, dễ bảo trì và dễ kiểm thử.

+ Đa dạng công nghệ view(template engine) Spring hỗ trợ nhiều công nghệ view như JSP, Thymeleaf, FreeMarker, và Velocity.

+ RESTful Web Services Spring có hỗ trợ mạnh mẽ cho việc xây dựng và tiêu thụ RESTful Web Services thông qua các module như Spring Web MVC và Spring Webflux.

* Spring Boot là một dự án của Spring Framework giúp đơn giản hóa việc phát triển ứng dụng Java bằng cách cung cấp cấu hình tự động, triển khai dễ dàng và các thư viện tiện ích sẵn có.
* Đi theo mô hình MVC, cung cấp giải pháp fullstack cho phát triển

ứng dụng web. Spring Boot hoàn chỉnh, gồm các thành phần sau:

+ Controller: Lớp này đảm nhiệm điều phối các yêu cầu từ người dùng hoặc giao diện người dùng đến các thành phần xử lý logic tương ứng. Nhiệm vụ chính của controller là nhận dữ liệu đầu vào, gọi các phương thức từ lớp service để thực hiện các xử lý logic cần thiết và sau đó chuẩn bị dữ liệu để trả về cho View.

+ Service: chứa các code tính toán, xử lý. Khi Controller yêu cầu, thì Service tương ứng sẽ tiếp nhận và cho ra dữ liệu trả cho Controller (trả về Model). Controller sẽ gửi về View.

+ Repository: Lớp repository cung cấp một giao diện để tương tác với cơ sở dữ liệu hoặc nguồn dữ liệu khác. Nhiệm vụ của repository là ẩn đi chi tiết kỹ thuật của cách dữ liệu được lưu trữ và truy xuất, cho phép service tương tác với dữ liệu mà không cần quan tâm đến cách thức thực hiện.

+ Model chỉ đơn giản là các đối tượng được Service tính toán xong trả về cho Controller. View thì có 2 loại, một là dạng truyền thống là trả về 1 cục HTML có data rồi. Lúc này Controller sẽ chuyển dữ liệu vào View và return về (Spring MVC có JSP hoặc template engine như Thymeleaf làm điều đó). View dạng 2 là dạng View tách riêng (không thuộc về project Spring boot). Thường có trong các hệ thống dùng API. View sẽ được viết riêng bằng React, Angular,... Controller sẽ đưa dữ liệu Model thông qua API cho View, và cũng nhận lại các yêu cầu qua API luôn.

* Ưu điểm của cấu trúc "controller - service - repository":

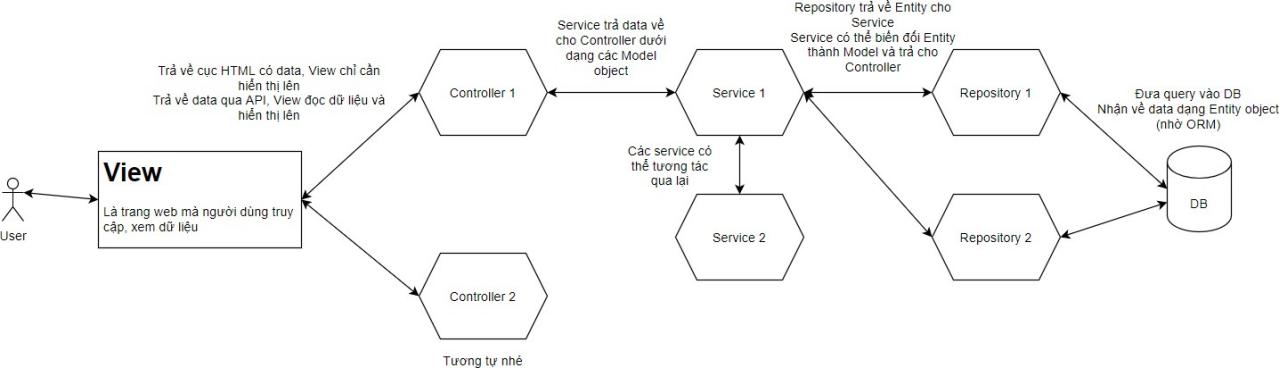
+ Tính rõ ràng và dễ bảo trì: Phân tách các phần khác nhau của

ứng dụng giúp dễ dàng theo dõi và bảo trì mã nguồn.

+ Tái sử dụng và mở rộng: Cấu trúc này thúc đẩy việc tái sử dụng code và cho phép mở rộng ứng dụng một cách dễ dàng, do các phần riêng biệt dễ dàng thay thế và nâng cấp.

+ Kiểm thử dễ dàng: Việc có các thành phần riêng biệt giúp thực hiện kiểm thử đơn vị và kiểm thử tích hợp dễ dàng hơn.

- Sơ đồ luồng đi trong spring boot.



+ Đầu tiên, user sẽ vào View để xem, tương tác

+ Khi user bắt đầu load dữ liệu (ví dụ click nút Reload), thì 1 request từ View gửi cho Controller (kiểu như "ê, cho tao cái danh sách user với")

+ Controller nhận được yêu cầu, bắt đầu đi hỏi ông Service (trong code là gọi method của Service)

+ Service nhận được yêu cầu từ Controller, đối với các code đơn giản có thể tính toán và trả về luôn. Nhưng các thao tác cần đụng tới database thì Service phải gọi Repository để lấy dữ liệu trong DB

+ Repository nhận được yêu cầu từ Service, sẽ thao tác với DB. Data lấy ra trong DB được hệ thống ORM (như JPA hoặc Hibernate) mapping thành các object (trong Java). Các object này gọi là Entity.

Và bây giờ sẽ là đi ngược lại trả về cho user:

+ Service nhận các Entity được Repository trả về, biến đổi nó.

Biến đổi ở đây là có thể thực hiện tính toán, thêm bớt các field,... và cuối cùng biến Entity thành Model. Model sẽ được trả lại cho Controller.

+ Controller nhận được Model, nó sẽ return cho View. Có 2 cách, một là dùng template engine pass dữ liệu Model vào trang HTML, rồi trả về cục HTML (đã có data) cho client. Cách 2 là gửi qua API, View tự parse ra và hiển thị tương ứng (hiển thị thế nào tùy View).

+ Khi View hiển thị xong, user sẽ thấy danh sách user hiện lên trang web.

- Một số annotations

+ @RestController: Được sử dụng để đánh dấu một lớp là một Controller và các phương thức trong lớp này trả về dữ liệu dưới dạng JSON.

+ @RequestMapping: Xác định các yêu cầu HTTP mà một phương thức trong Controller có khả năng xử lý.

+ @GetMapping, @PostMapping, @PutMapping, @DeleteMapping: Các Annotation này đánh dấu các phương thức để xử lý các yêu cầu GET, POST, PUT và DELETE tương ứng.

+ @Autowired: Sử dụng để chú thích các trường, phương thức setter hoặc constructor để Spring tự động tiêm (inject) các phụ thuộc.

+ @Service, @Repository: Các Annotation này đánh dấu các lớp để Spring Boot hiểu rằng đó là các thành phần có thể quản lý và tự động tạo (autowire).

+ @PathVariable: Sử dụng để trích xuất giá trị từ URL của yêu cầu HTTP và sử dụng trong các phương thức Controller.

+ @RequestParam: Sử dụng để trích xuất các tham số từ querystring của URL trong yêu cầu HTTP.

+ @ResponseBody: Được sử dụng để chỉ định rằng phản hồi của phương thức sẽ được trả về trực tiếp mà không cần render qua một view.

+ @ManyToMany: thể hiện mối quan hệ nhiều - nhiều giữa hai thực thể.

+ @ManyToOne: thể hiện mối quan hệ nhiều - một giữa hai thực thể.

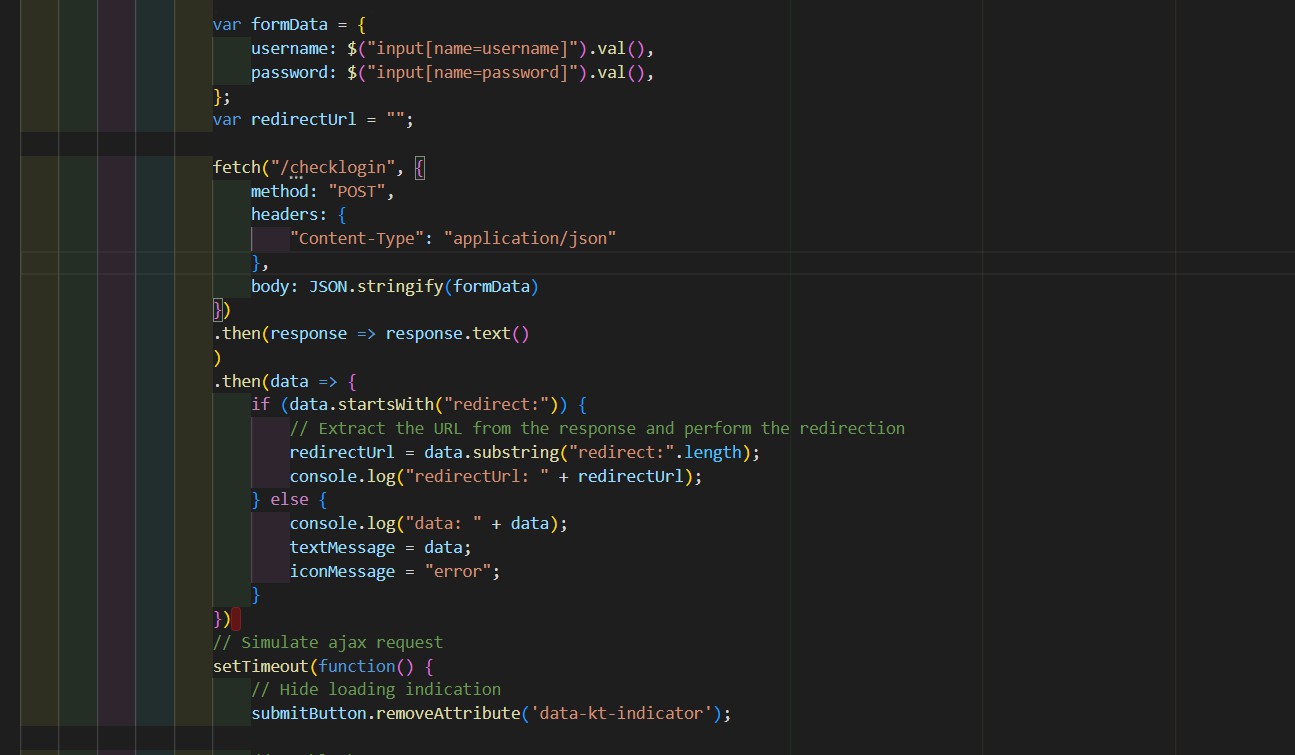
+ @OneToMany: thể hiện mối quan hệ một - nhiều giữa hai thực thể.

+ @JoinColumn: dùng để xác định cột khóa ngoại được sử dụng để thiết lập quan hệ giữa hai thực thể trong cơ sở dữ liệu. Bên sở hữu khóa ngoại là bên nhiều trong mối quan hệ.

* JS

JavaScript là một ngôn ngữ lập trình phía client được sử dụng để tạo tính năng động, tương tác trên trang web. Nó xử lý sự kiện, thay đổi nội dung và làm việc với DOM để thay đổi trang mà không cần tải lại. JavaScript cũng hỗ trợ lập trình bất đồng bộ, giao tiếp với máy chủ qua AJAX và có nhiều thư viện/framework hỗ trợ.

Ví dụ:



Dùng lệnh fetch để tạo post request đến “/checklogin” với requestBody là để xử lý chức năng đăng nhập trong controller, nhận là response là url sau khi check, login thành công.

* 1. Front - end.
* HTML.
* CSS.
* BOOTSTRAP.
* JQUERY.

## Công việc triển khai.

* 1. Xác định đối tượng.
     1. *Người sử dụng (User).*

User là đối tượng truy cập website để xem thông tin các bài học về từ vựng, bài thi và làm bài thi, các chức năng chủ yếu:

* + - * Đăng ký tài khoản.
      * Đăng nhập, đăng xuất, chỉnh sửa profile (bao gồm email, tên hiển thị, password).
      * Quên mật khẩu.
      * Xem danh sách các bài thi, các mục phân loại.
      * Hoc và làm bài thi.
    1. *Quản trị viên(Admin).*

Admin là đối tượng truy cập website để tạo từ vựng liên quan tường topic và quản lý các bài thi:

* + - * Quản lý người dùng: xem danh sách người dùng; thêm người dùng; chỉnh sửa profile của người dùng.
      * Tạo, xem, sửa, xóa (CRUD) các danh mục
      * CRUD các bài thi (bao gồm tạo từ vựng theo topic, câu hỏi, các lựa chọn đúng/sai, thời gian làm bài thi).
    1. Word.

Word là đối tượng quan trọng trong web học từ vừng. Word bao gồm các loại danh từ, động từ, tính từ, trạng từ,... Mỗi từ thường đi kèm với nghĩa hoặc nhiều nghĩa khác nhau. Trong việc học từ vựng, việc hiểu rõ ý nghĩa của từ là quan trọng. Word là nền tảng chính trong việc phân loại topic giúp người dùng dễ dàng tổ chức và tìm kiếm từ vựng. Ngoài ra, word còn là tiền để tạo nên các bài tập và kiểm tra liên quan đến từ vựng đã học.

* 1. Thiết kế chức năng cho đối tượng sử dụng.
  2. Chức năng chung.
     + Chức năng đăng ký.

Cho phép người dùng đăng kí để vào học và làm bài thi.

* + - * Trên trang chủ, khi người dùng ấn vào Register thì vào màn hình đăng ký, người dùng có thể đăng ký tài khoản bằng cách: Tạo tài khoản bằng email: các trường cần nhập username, email, phone, address, password, password confirmation (password bắt buộc trên 8 ký tự).
      * Sau khi nhập hết các trường tài khoản sẽ được đăng ký thông qua:

+ *registrationService.register(newUser)*; // đăng ký user mới vào database

+ register() sẽ gửi email kèm token khi click vào trước khi token hết hạn sẽ tạo get request để confirm token bằng hàm *emailSender.send(user.getEmail(),buildEmail(user.getFull*

*name(), link));*

+ Sau khi xác nhận tài khoản sẽ đăng ký thành công vào chuyển đến màn hình đăng nhập.

* + - Chức năng đăng nhập.

Cho phép người dùng đã có tài khoản đăng nhập vào hệ thống.

* Trên trang chủ, khi người chơi ấn vào Login thì hiện ra màn hình đăng nhập người dùng có thể đăng nhập bằng cách nhập username, password.
* hệ thống check username có tồn tại không rồi sau đó check password nếu đúng sẽ đăng nhập thành công: *checkLogin()*
  + - Cập nhật Profile.

Cho phép dùng sửa thông tin cá nhân của mình, khi ấn vào Edit Account thì màn hình sẽ hiển thị bao gồm các trường sau:

+ Email

+ Password, password confirmation, current password

+ Username

+ Avatar

* *showView()* : show profile user
* *viewProﬁle()*: để vào trang cập nhật profile.
* *updateProﬁle()*: lưu lại thông tin khi cập nhật.
* *updateEmail()*: cập nhật email.
* *updatePassword()*: cập nhật mật khẩu.
  + Chức năng quên mật khẩu.

Cho phép người dùng thay đổi mật khẩu và xác nhận bằng email đã đăng ký tài khoản. Khi nhập email đã dùng trong đăng ký tài khoản, hệ thống sẽ gửi mã OTP về email để xác nhận lấy lại mật khẩu. Nhập đúng mã OTP, người dùng có thể đặt lại mật khẩu mới cho mình.

* + - Người dùng muốn thay đổi tài khoản sau khi nhập email dùng để đăng ký. Hệ thống sẽ kiểm tra xem có tồn tại không sau đó sẽ gửi mail kèm mã OTP đến email người dùng: *sendOTP()*
    - Sau khi nhập OTP hệ thống kiểm tra trùng mã OTP gửi về mail không rồi cho phép người dùng thay đổi mật khẩu: *verifyOTP()*
    - Update password gọi user từ email, set password mới, save user:

*updatePassword()*

* + Xem các topic và các từ vựng thuộc mỗi topic

Hiển thị các thông tin danh sách topic của trang web, sau khi bấm vào mỗi topic sẽ hiện lên tất cả từ vựng thuộc topic đó hiển thị thông tin chi tiết của word (Name, Mean, Attributes và Example).

* + Làm bài thi.

Khi người dùng bấm vào bài thi thì sẽ hiển thị thông tin: tiêu đề, số câu hỏi, thời gian làm bài

Khi người dùng bấm vào làm bài thi, các thông tin hiển thị:

+ từng câu hỏi kèm theo các lựa chọn về đáp án và kèm theo câu hỏi về độ chắc chắn khi lựa chọn. Nếu người dùng chọn đáp án đúng nhưng không chắc chắn, chắc chắn đúng nhưng câu trả lời sai thì hệ thống sẽ đề xuất học lại từ vựng thuộc topic nào đó.

+ Đồng hồ đếm ngược thời gian làm bài.

+ Khi làm xong bài, người thi ấn nút nộp bài(Submit) hoặc khi hết thời gian, bài thi sẽ tự động được nộp.

* 1. *. Chức năng cho quản trị viên*
     + Chức năng tạo bài thi(câu hỏi).
       - Tạo bài thi:

+ Tiêu đề bài thi.

+ Thời gian làm bài (tính theo phút).

+ Từng câu hỏi trong đề thi, bao gồm câu hỏi, số điểm của mỗi câu hỏi, mỗi câu hỏi bao gồm các lựa chọn.

+ Chọn đáp án sai/đúng cho mỗi câu hỏi.

* + - Chức năng quản lý người dùng.

Admin có xem được tất cả các user sử dụng trang web bao gồm thông tin chi tiết của user khi mà đăng ký. Admin cũng có thể thêm, xóa người dùng.

* + - Chức năng quản lý topic.

Admin có thể xem được danh sách các topic hiện có trên phần mềm bao gồm thông tin chi tiết từng topic có thể thêm sửa và xóa topic ngoài ra có thể thêm cả từ mới vào topic bao gồm từ vựng, cách phát âm, loại từ, nghĩa tiếng việt, luyện nghe, danh sách bài tập kèm theo.

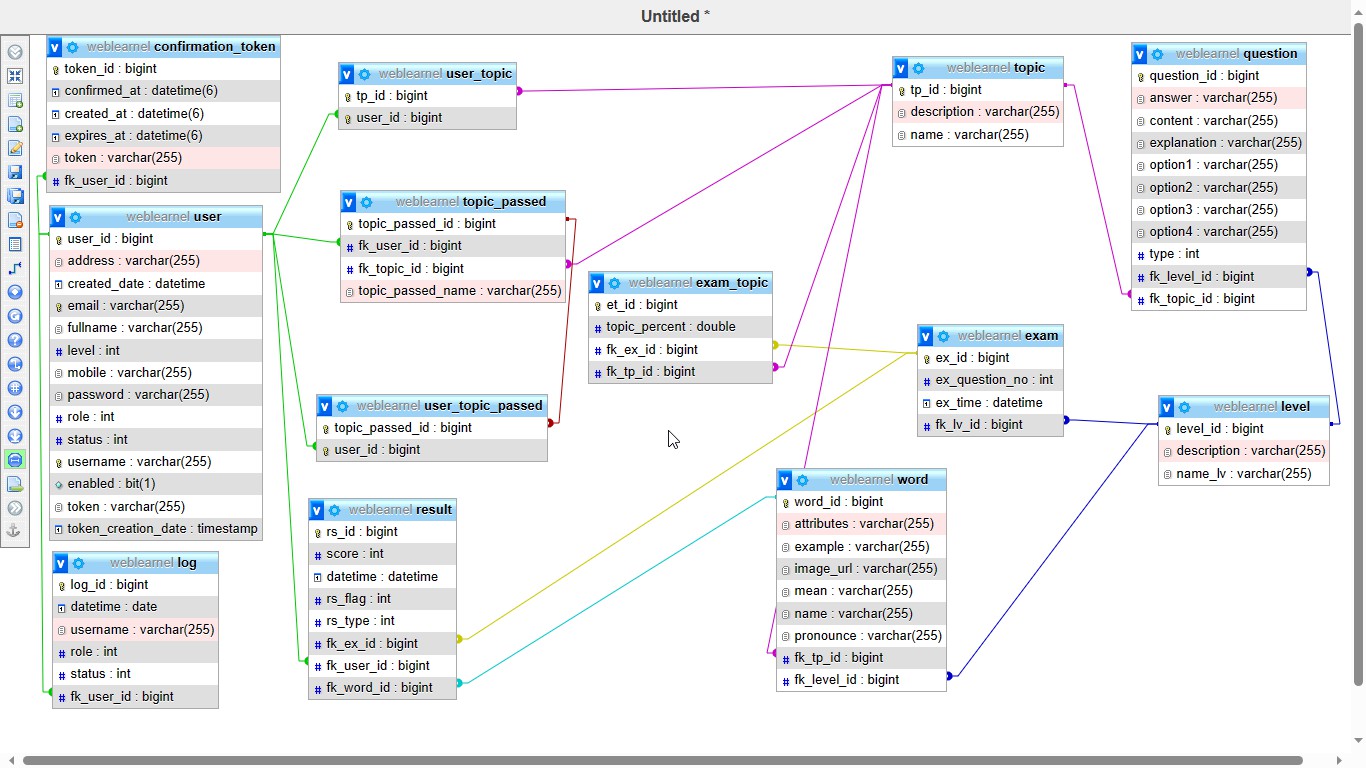
* + - Chức năng quản lý câu hỏi.

Admin có thể xem toàn bộ các câu hỏi và có thể thêm sửa xóa câu hỏi.

* + - Chức năng quản lý từ.

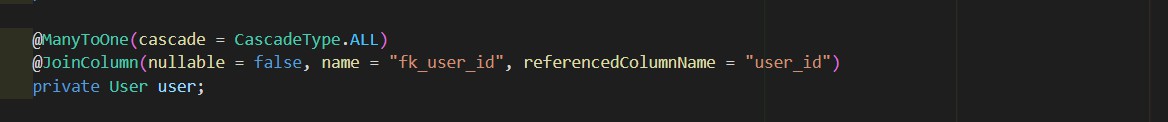
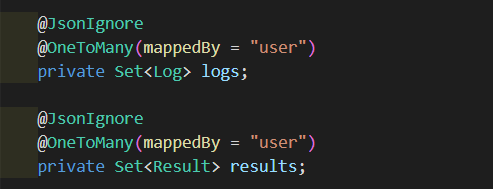
Admin có thể xem toàn bộ các từ và có thể thêm sửa xóa.

* 1. Thiết kế cơ sở dữ liệu.

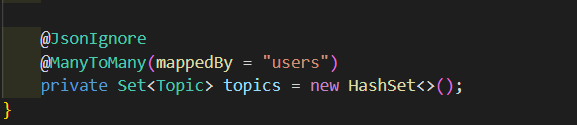


Mô tả quan hệ cơ sở dữ liệu:

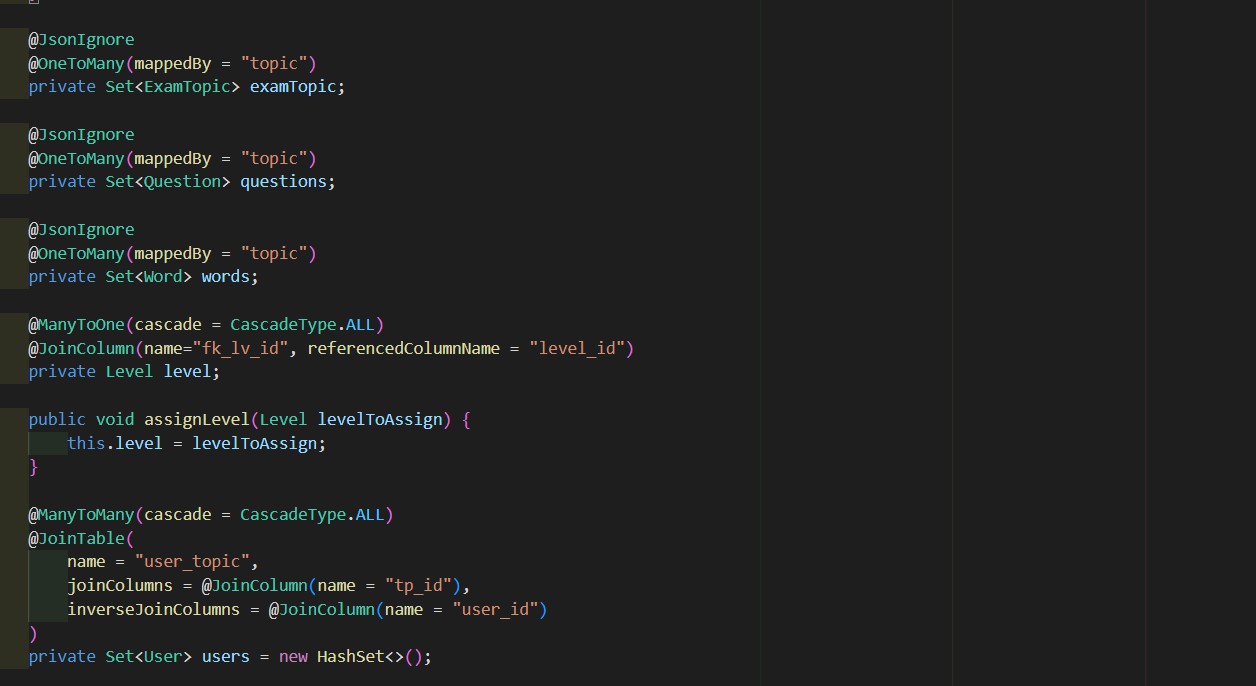
* Một user có nhiều result, log(phân quyền), confirmation\_token.



* Nhiều user có nhiều topic: user\_topic, topic\_passed, user\_topic\_passed.



* Một topic có nhiều exam\_topic, word, question.



* Một exam có nhiều exam\_topic, result.



* Nhiều question, exam, word thuộc cùng một level.

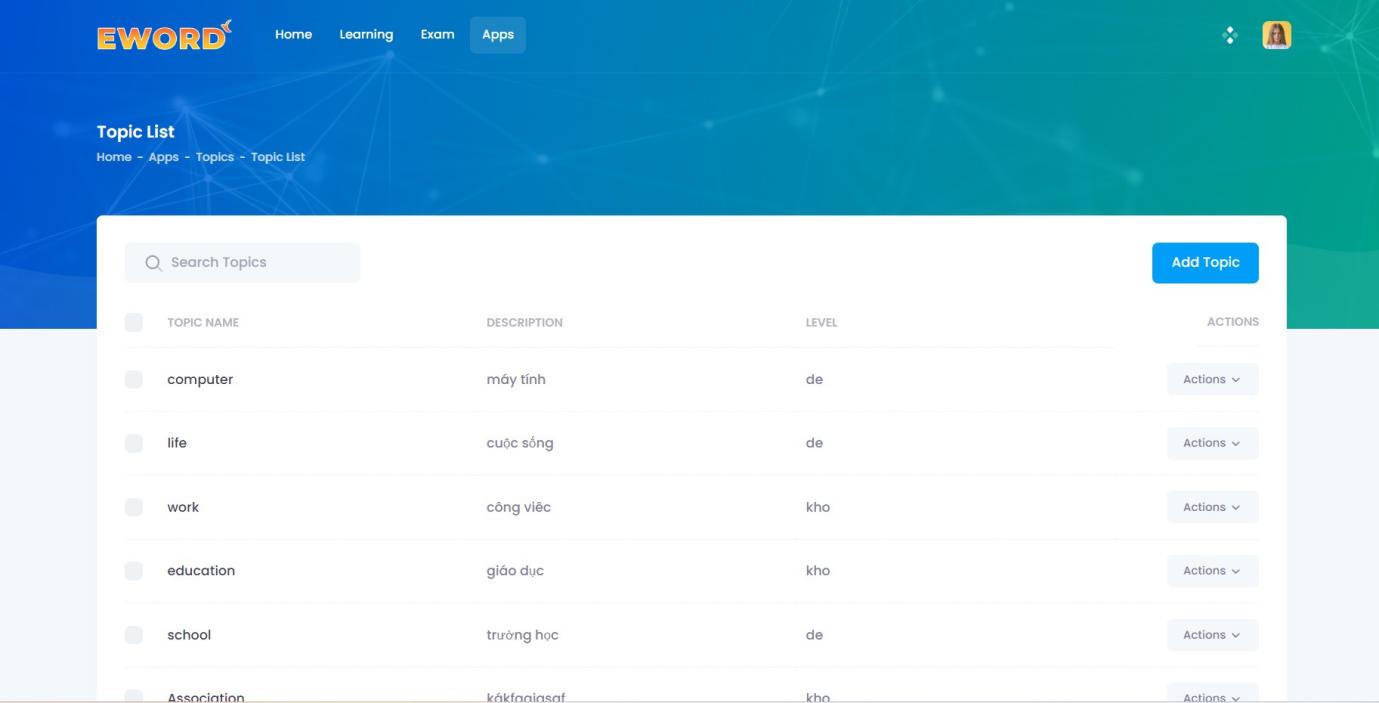


## Kết quả

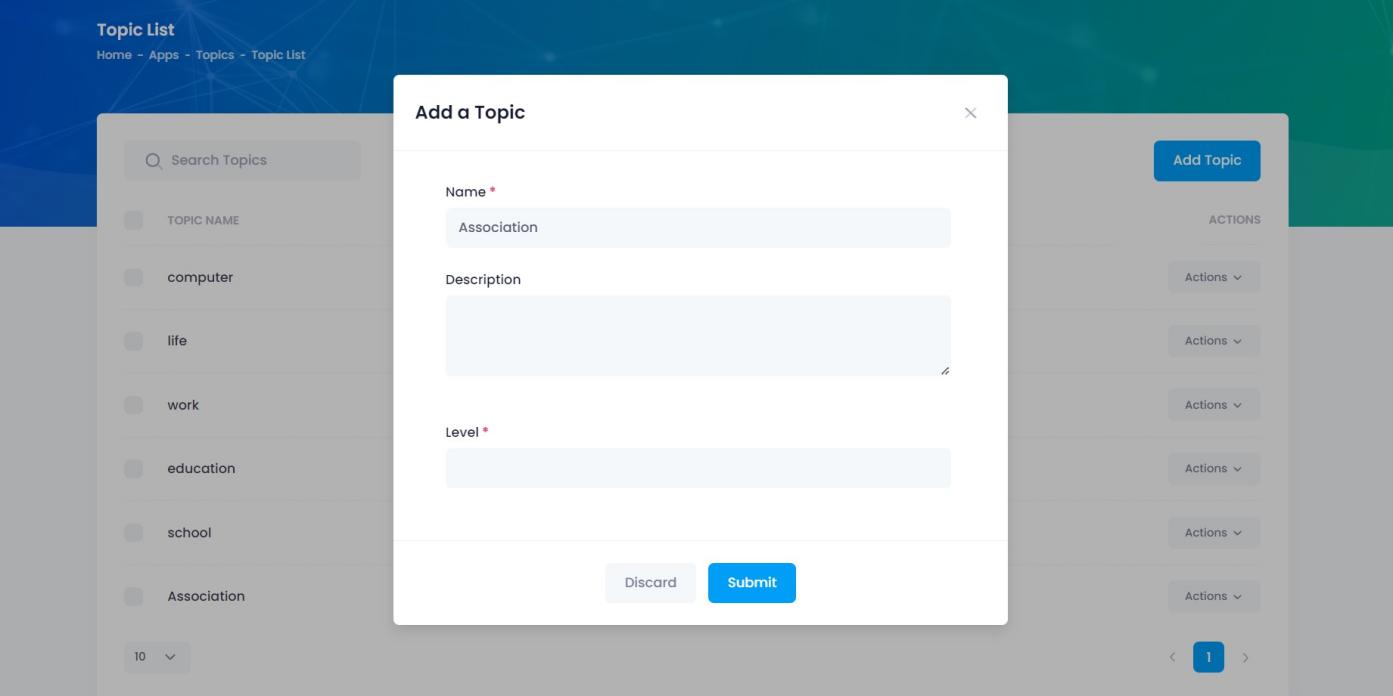
1. Giao diện Admin.

Giao diện sau khi đăng nhập quyền Admin.

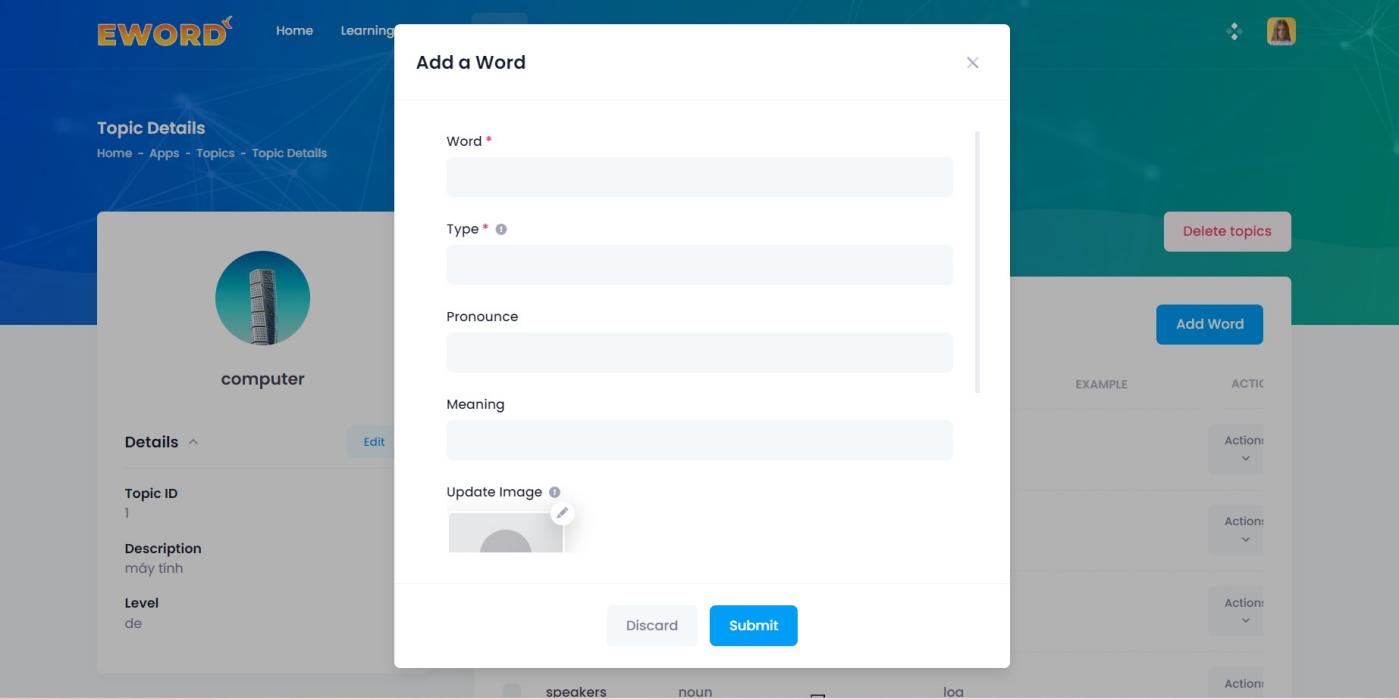
* 1. quản lý topic.



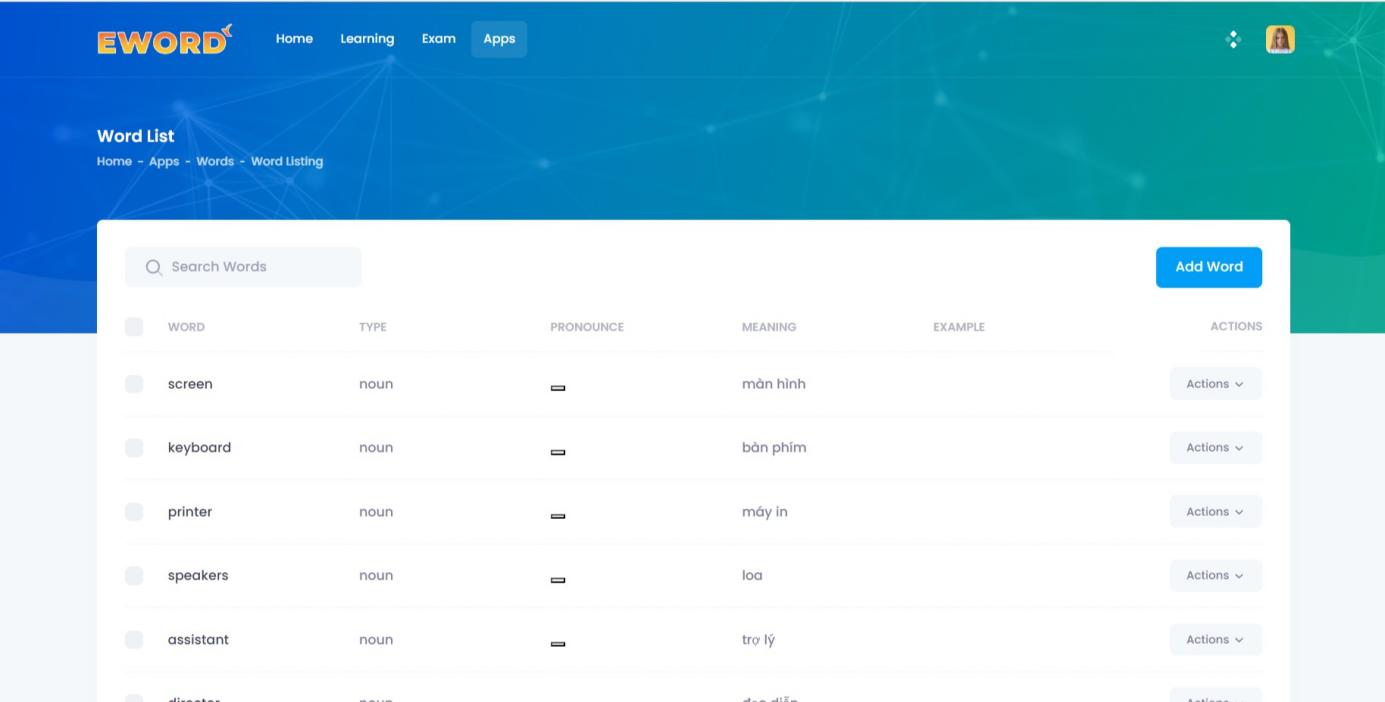
* + - thêm topic:



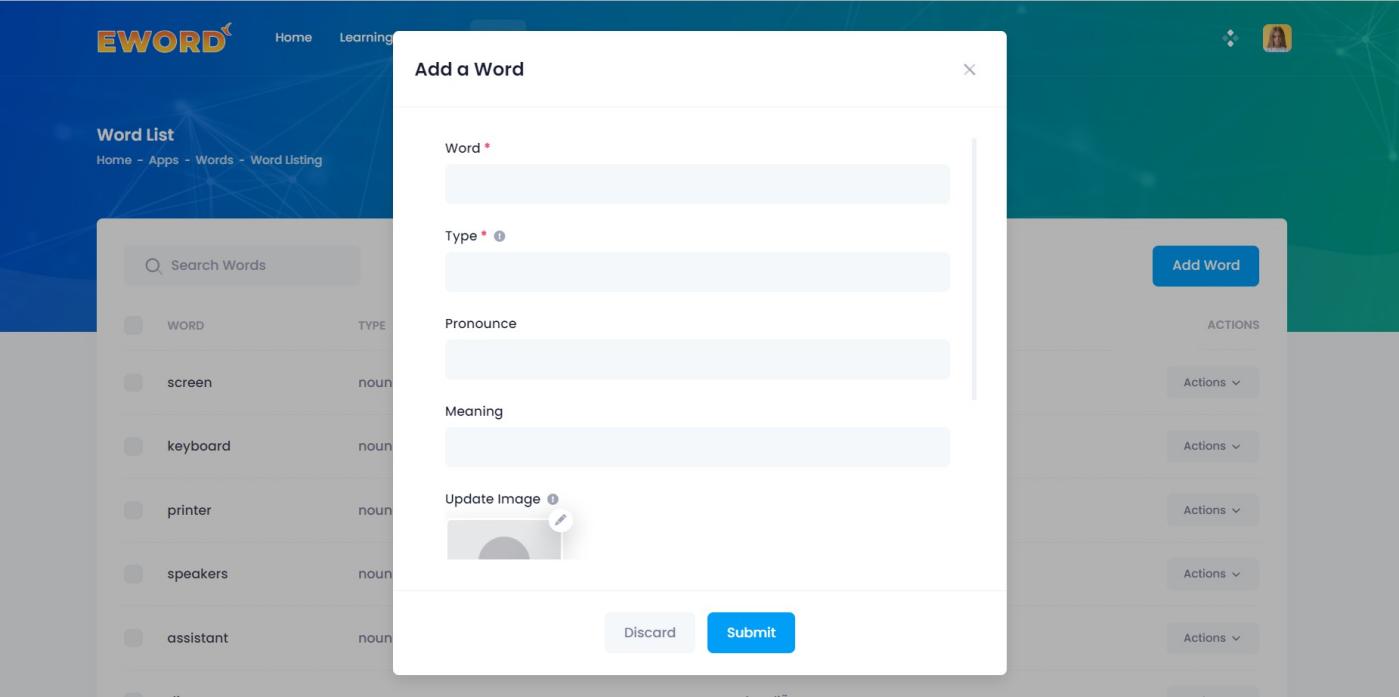
* + - thêm từ trong topic



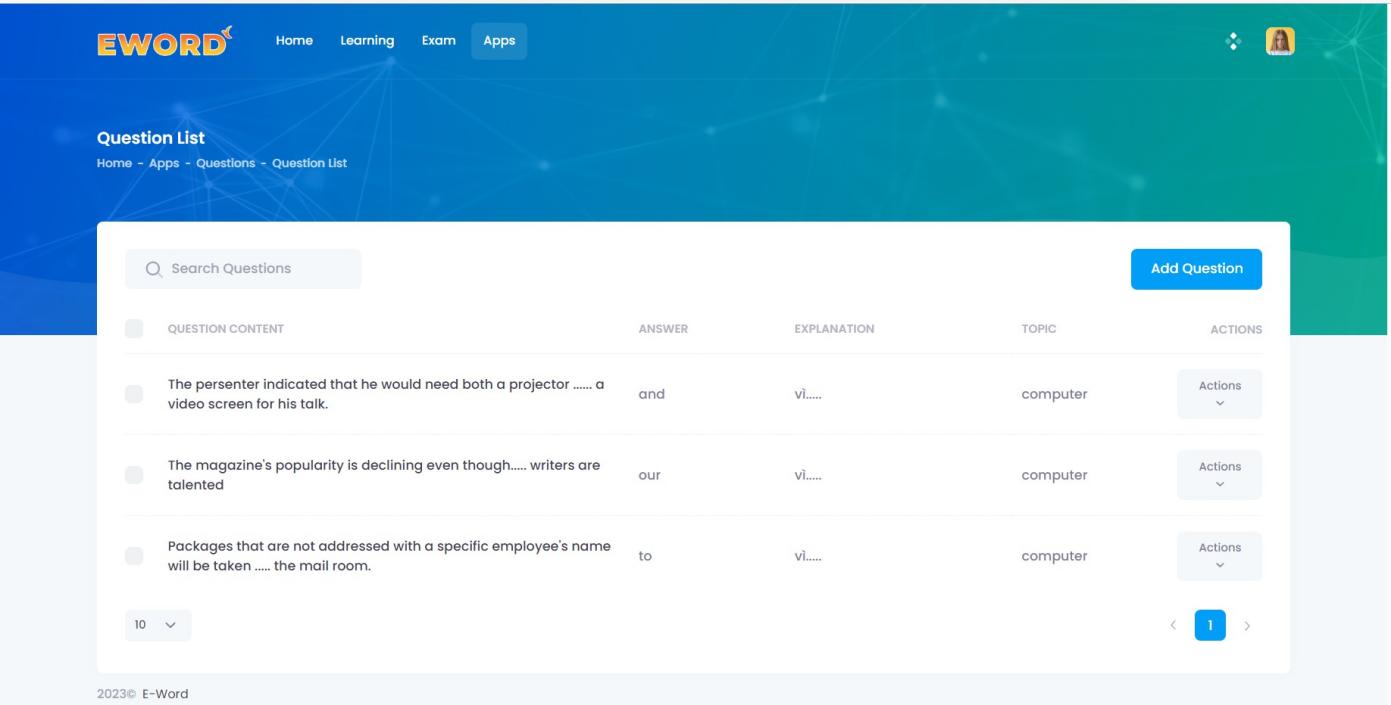
* 1. Quản lý word.
     + danh sách word.



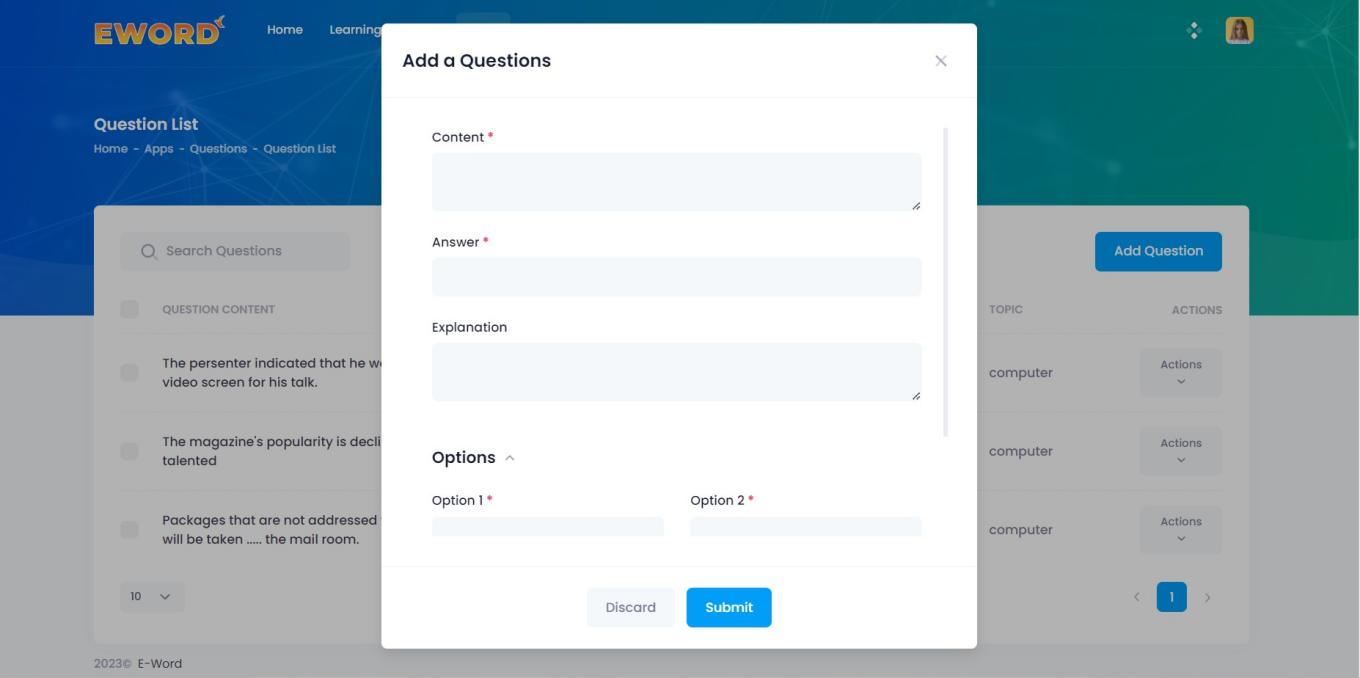
* + - thêm word.



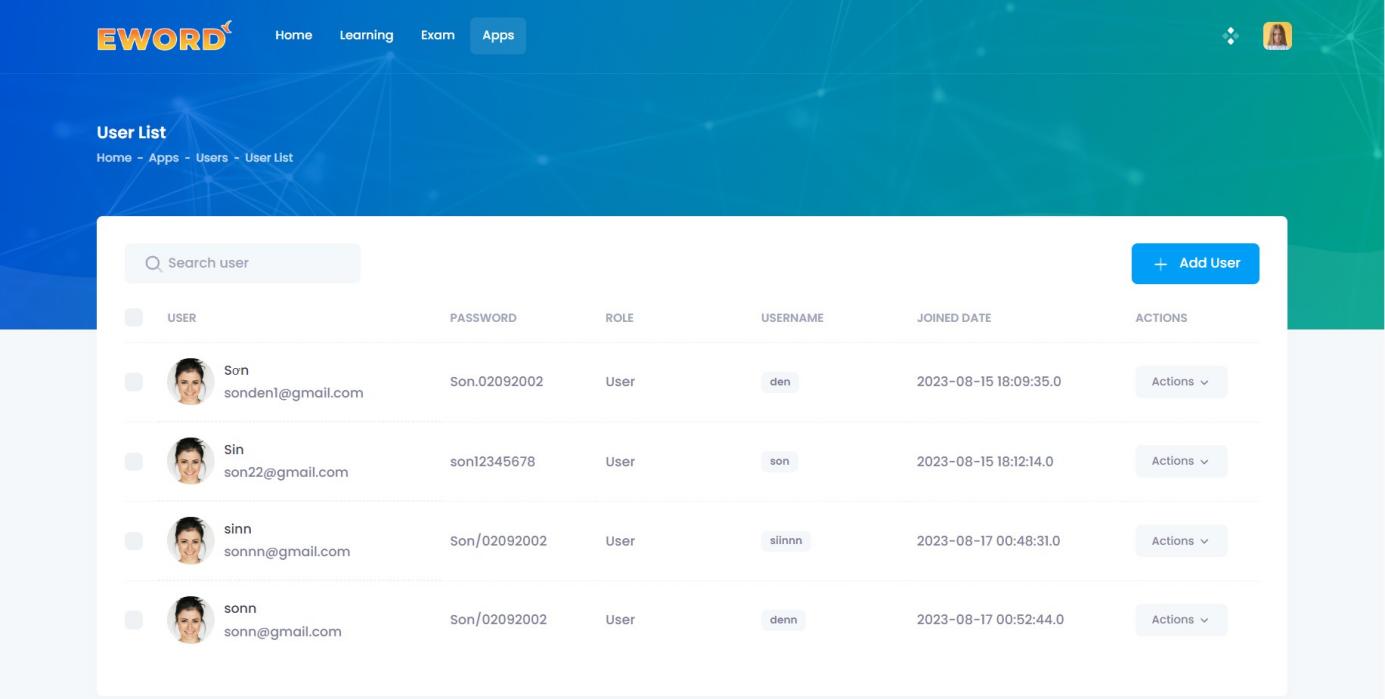
* 1. Quản lý câu hỏi.
     + Danh sách câu hỏi.



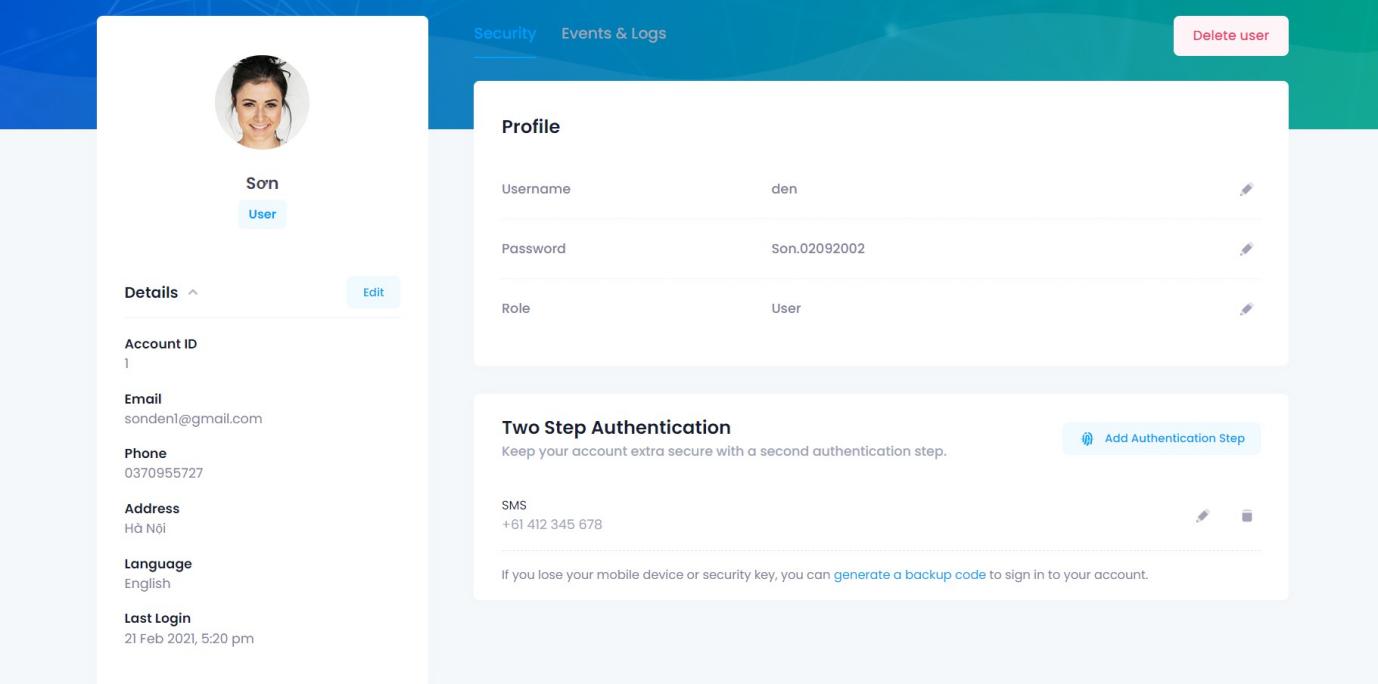
* + - thêm câu hỏi.



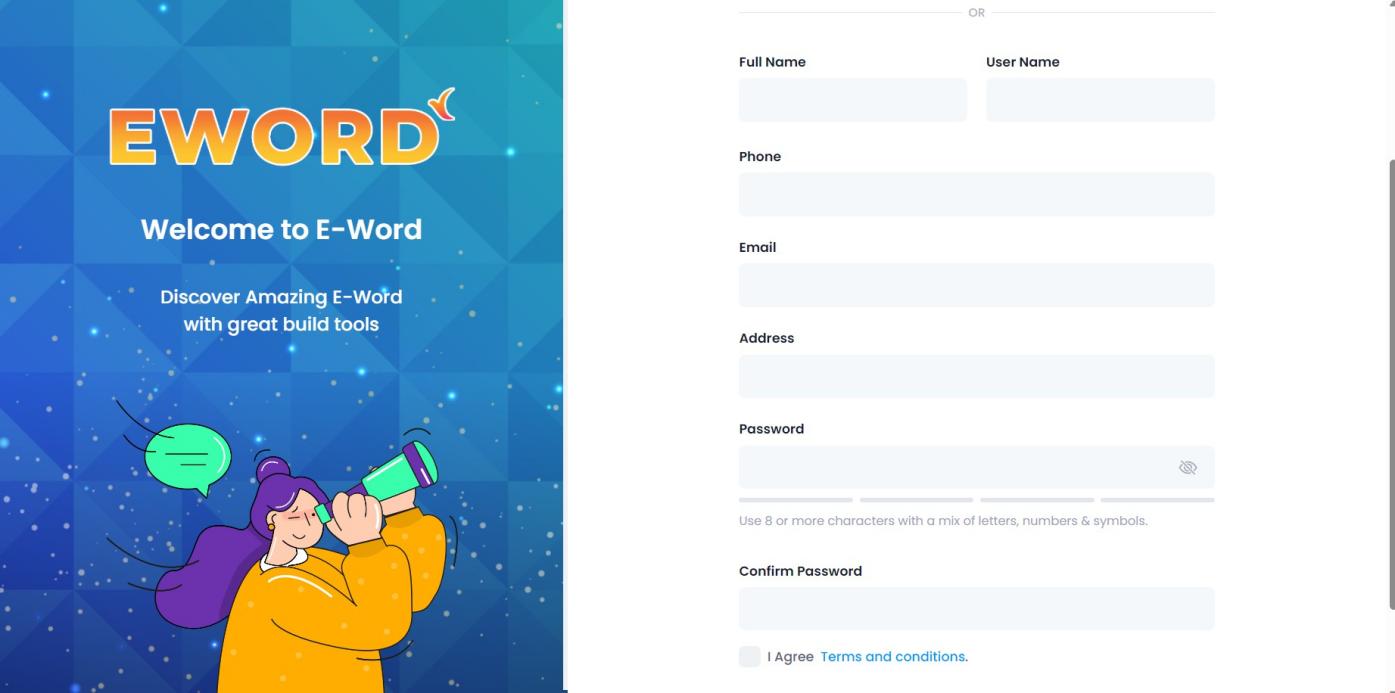
* 1. Quản lý người dùng, chỉnh sửa thông tin người dùng.
     + Xem danh sách người dùng.



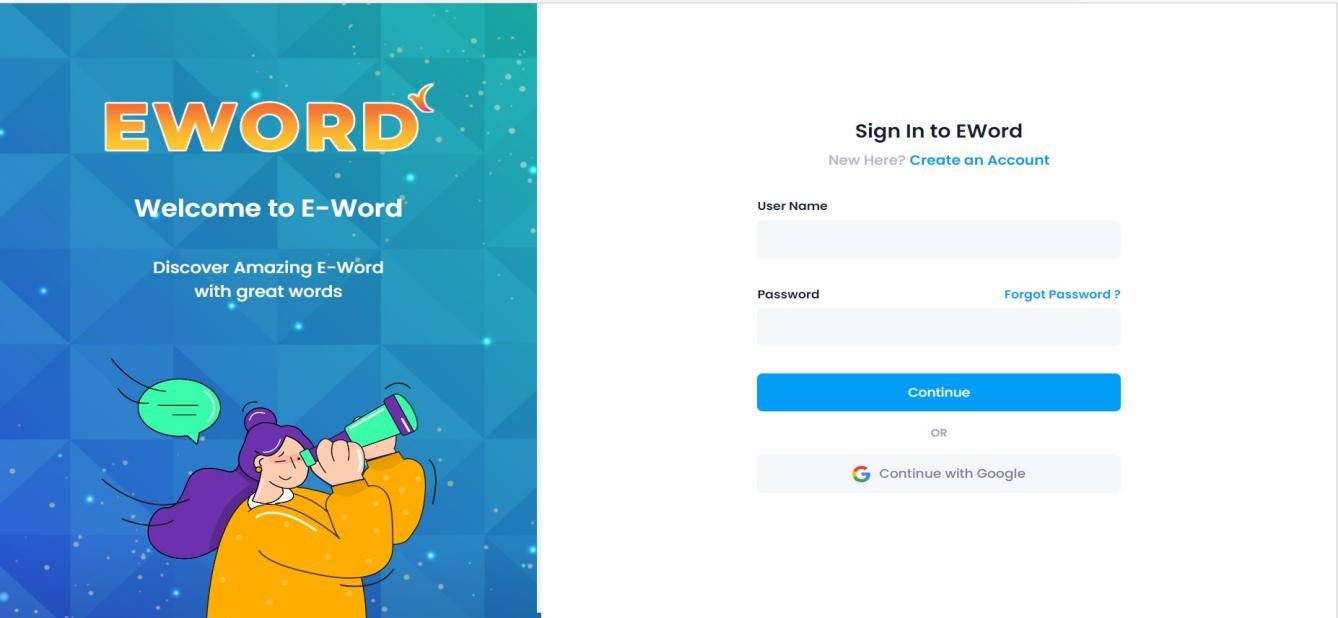
* + - Xem thông tin từng user.



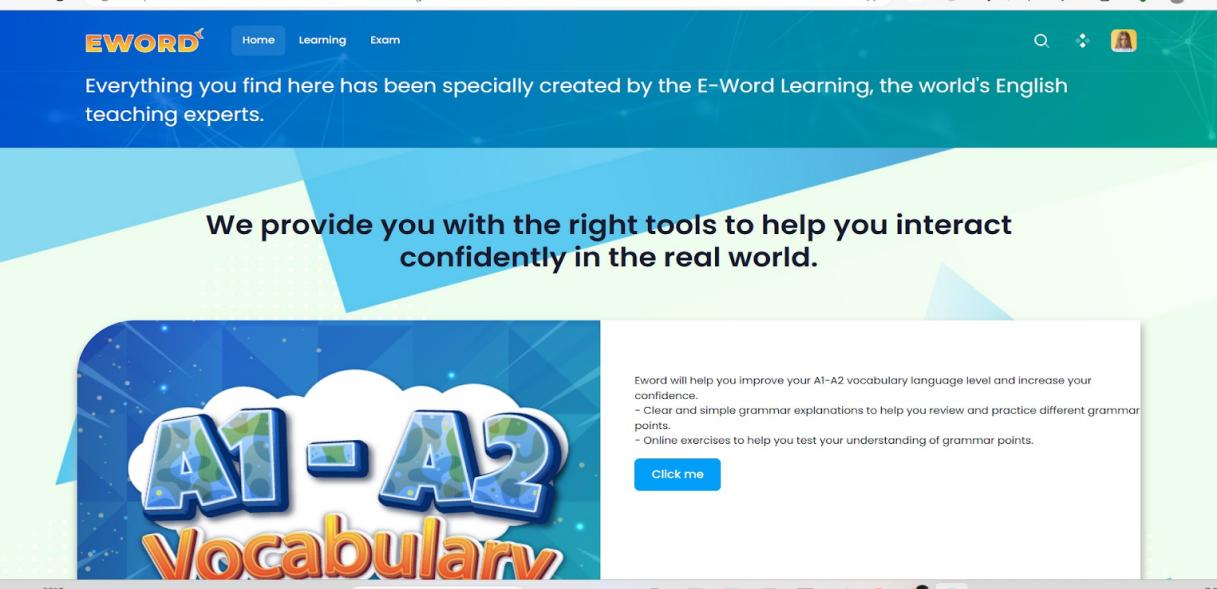
1. Giao diện User.
   1. Giao diện đăng ký.



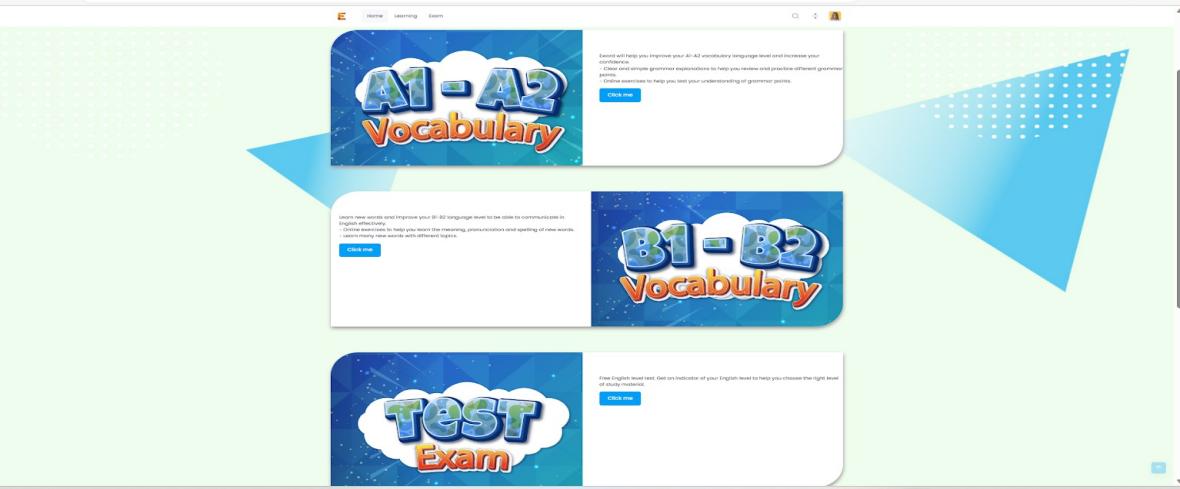
* 1. Giao diện đăng nhập.



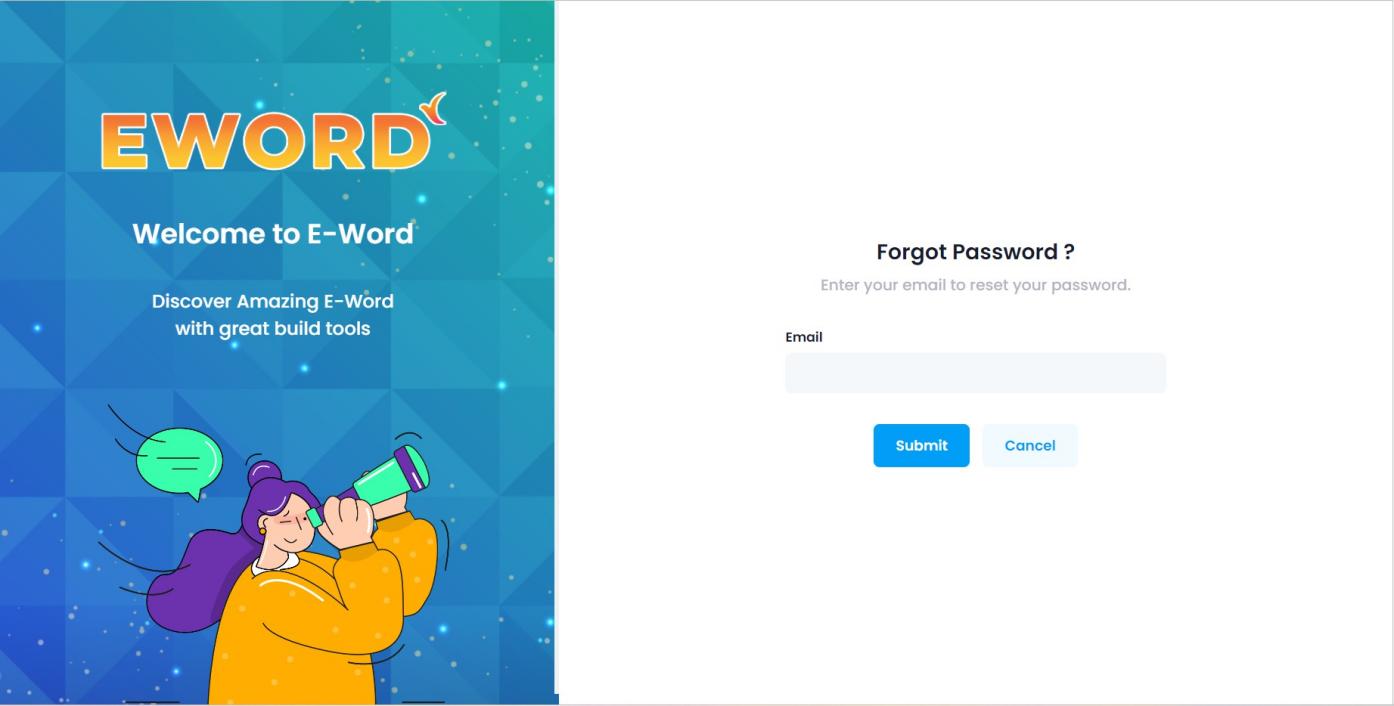
* 1. Giao diện chính.



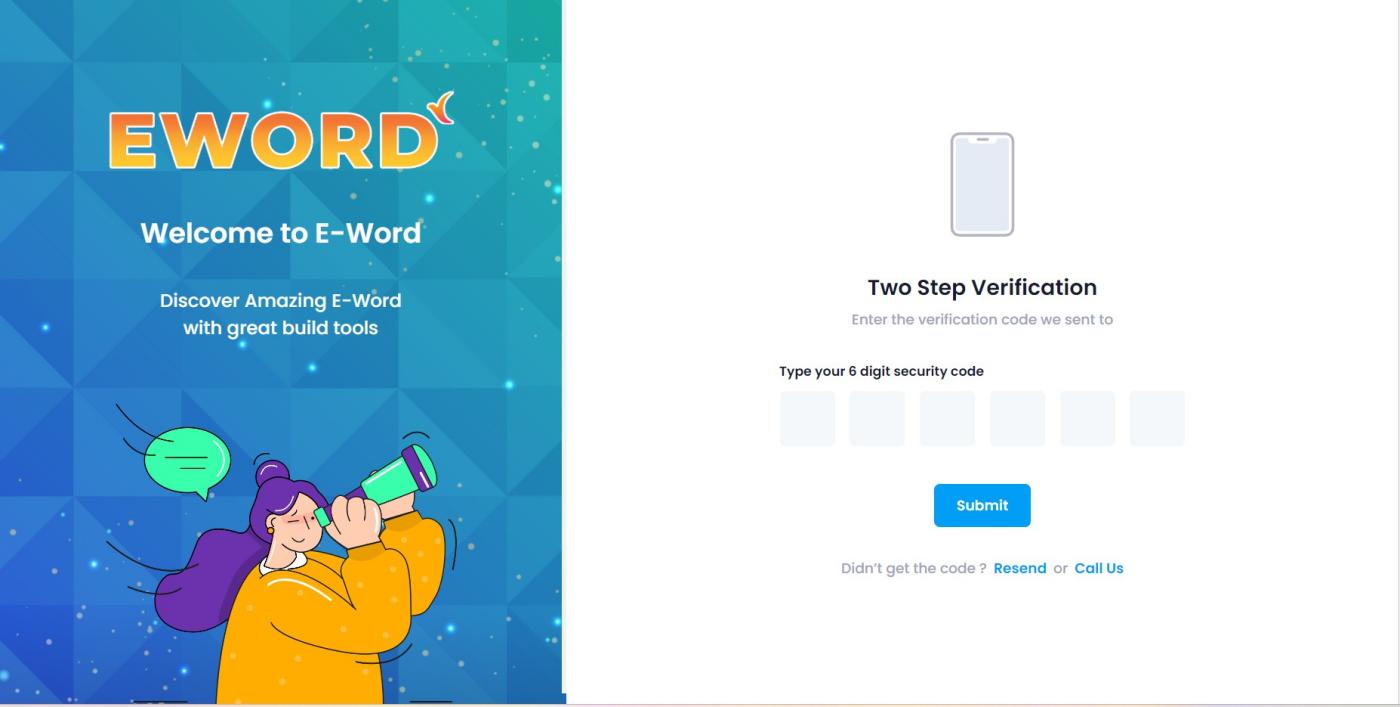
* + - Tại trang chủ, thiết lập truy cập nhanh đến Learning A1-A2, Learning B1-B2 và Test Exam.



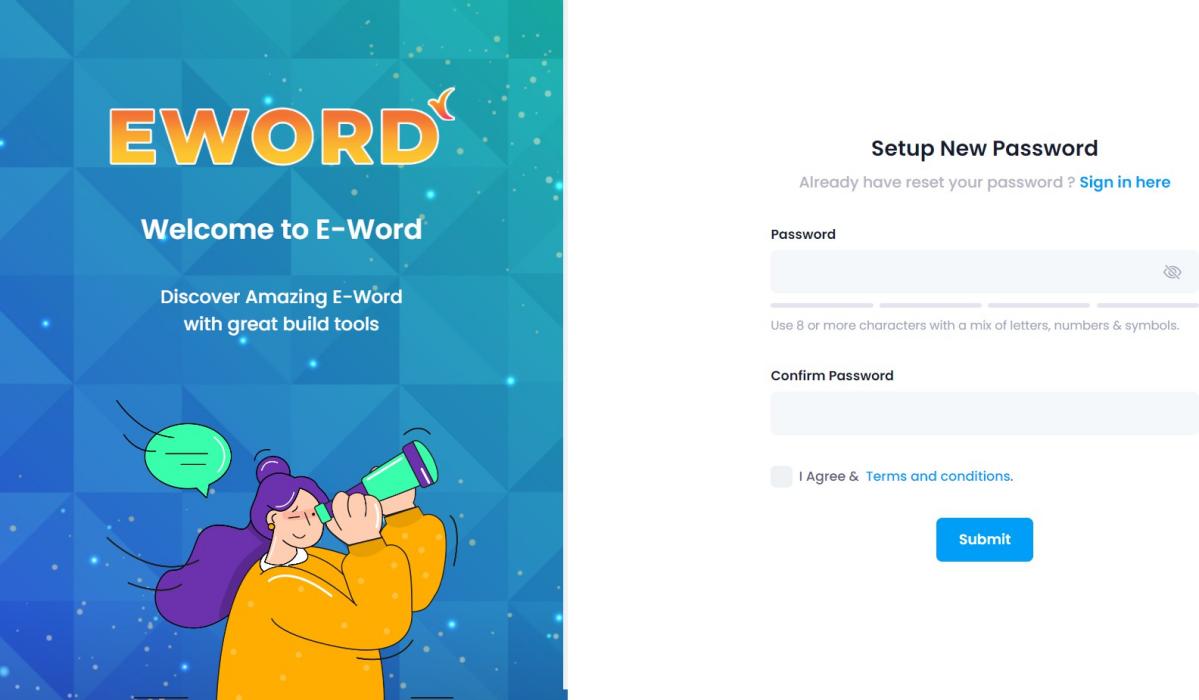
* 1. Quên mật khẩu.
     + Nhập email dùng khi đăng ký để nhận mã OTP.



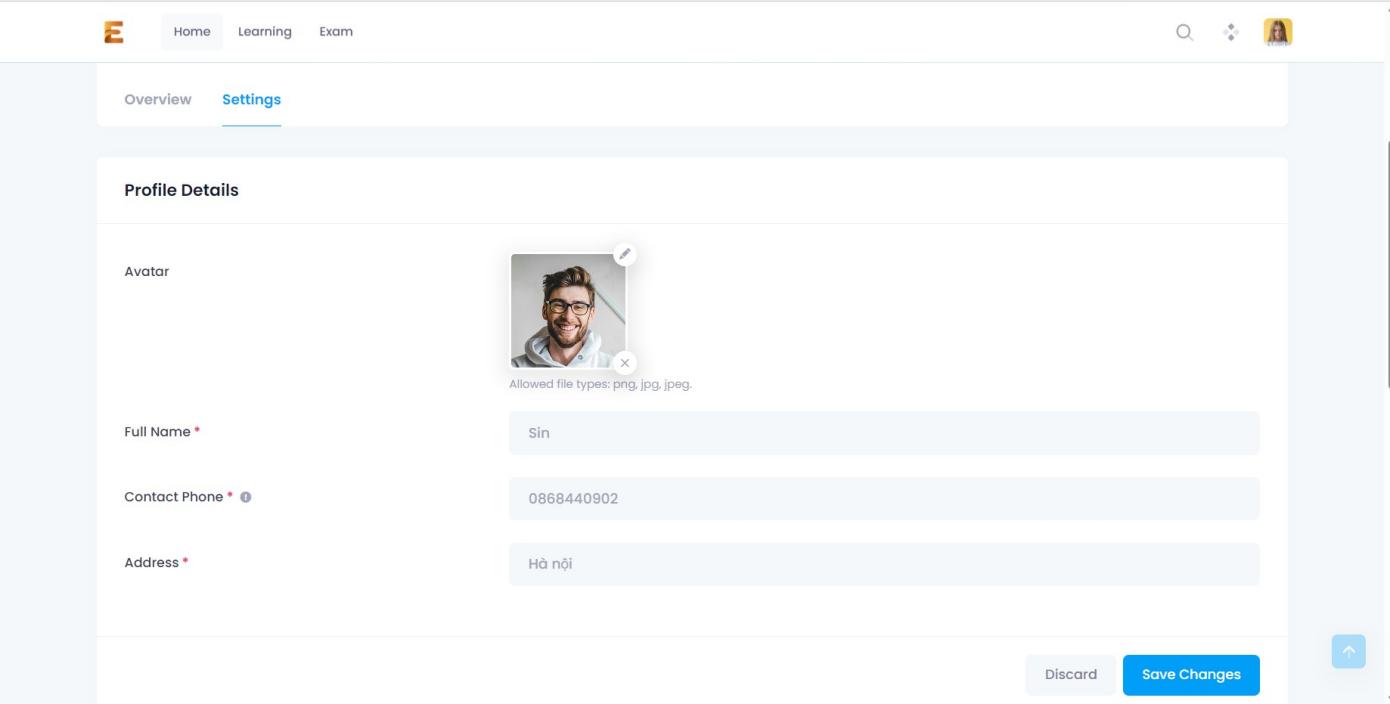
* + - Nhập mã OTP được gửi đến email để xác nhận thay đổi mật khẩu.



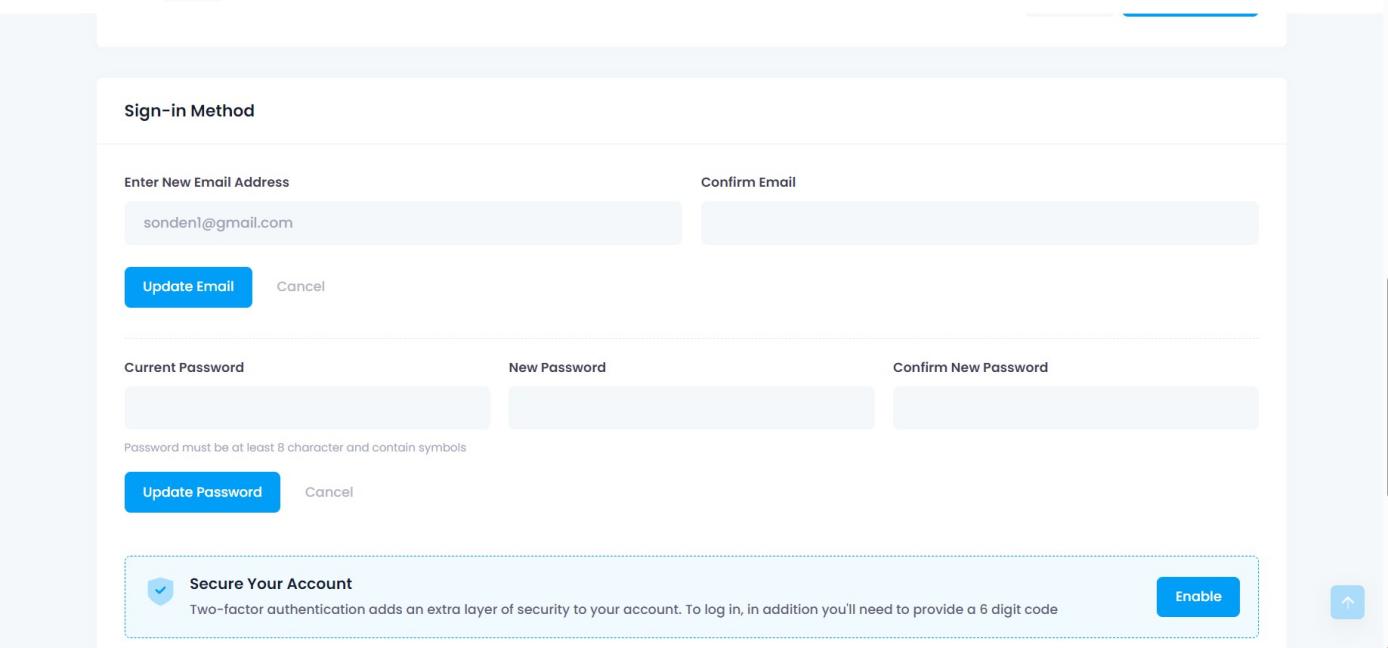
* + - Đặt mật khẩu mới và xác nhận mật khẩu.



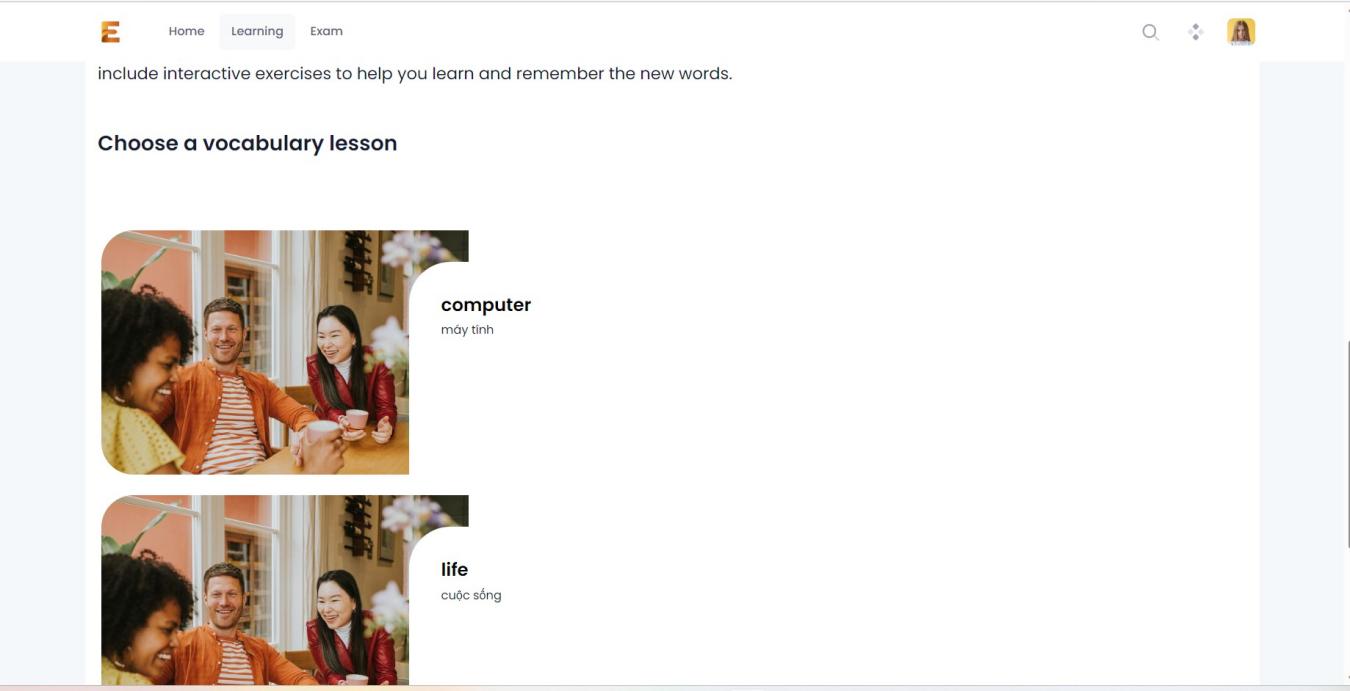
* 1. Cập nhật Profile.
     + Cập nhật thông tin.

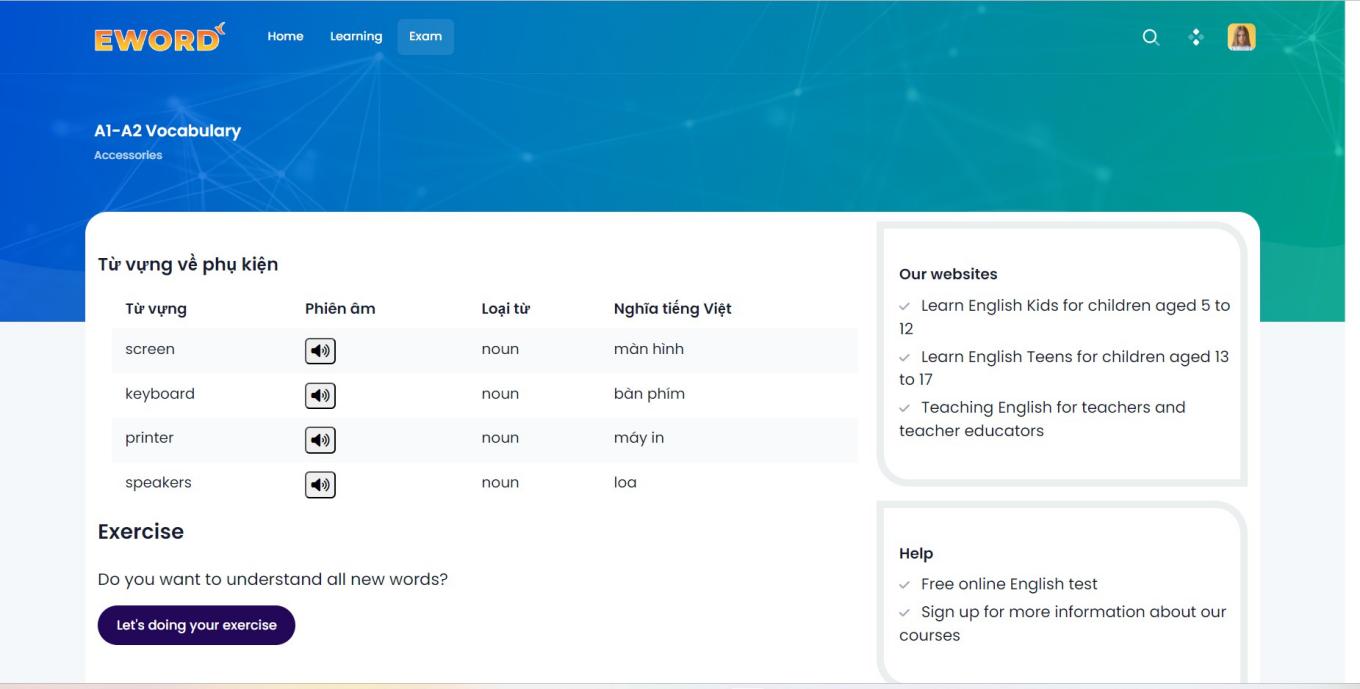


* + - cập nhật email và password.

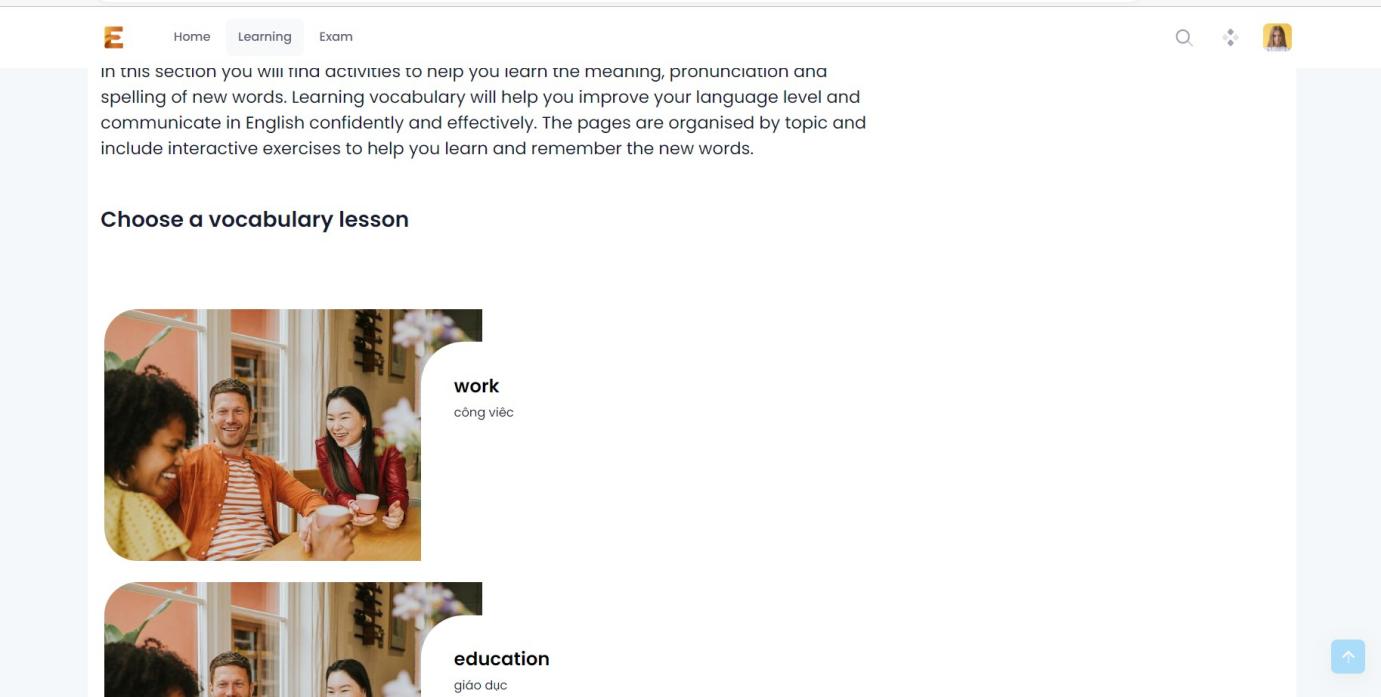


* 1. Learning(demo)
     + A1-A2 vocabulary





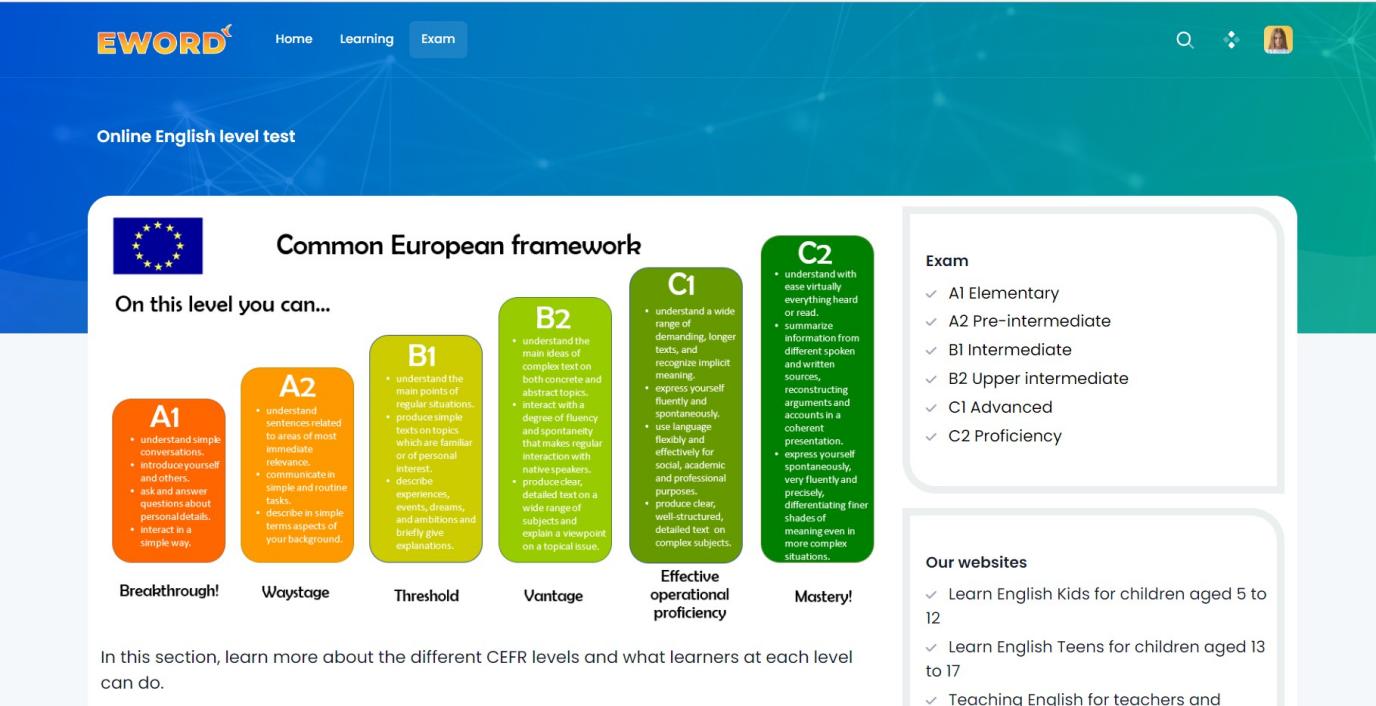
* + - B1-B2 vocabulary



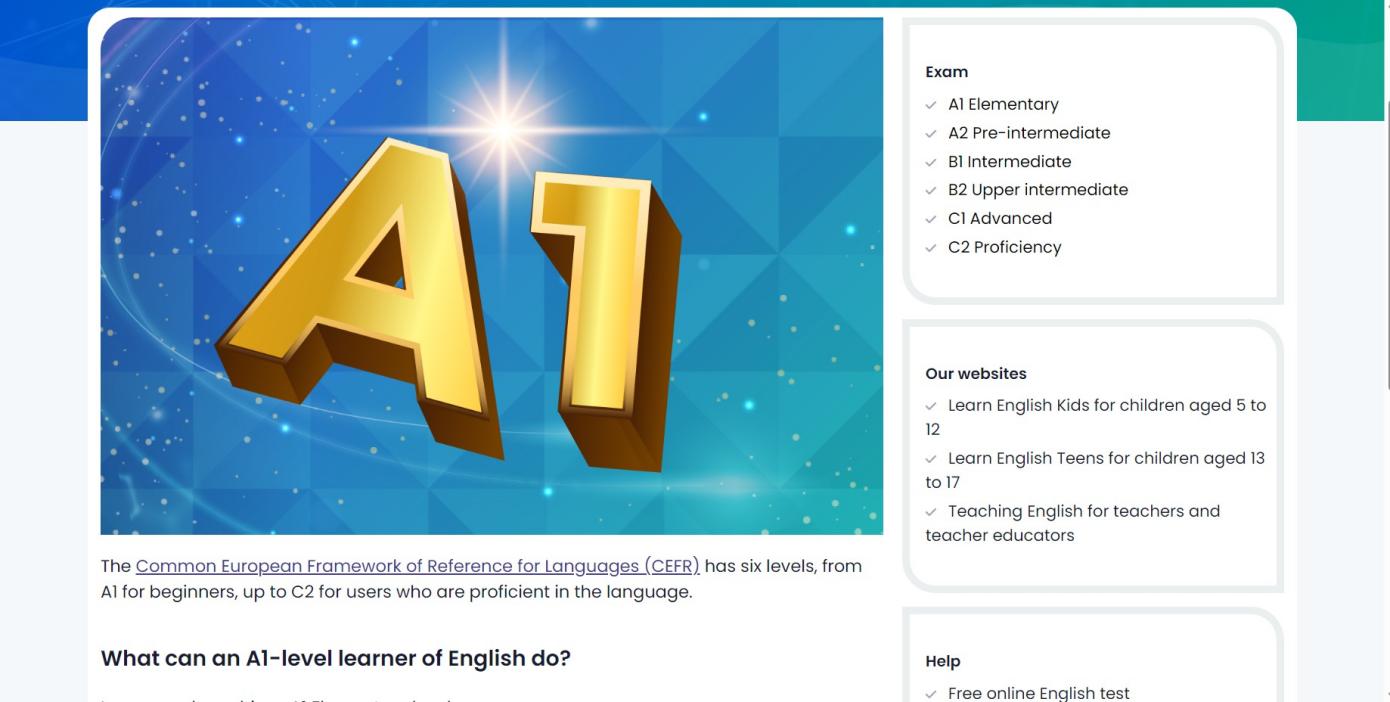
* 1. Exam.
     + Exam Overview: Giới thiệu tổng quan về CEFR levels và các level chính của nó, bao gồm: A1, A2, B1, B2, C1, C2. Đối với từng level,

người dùng sẽ biết được mình cần những kiến thức gì để đạt được level đó.

< trang overview >

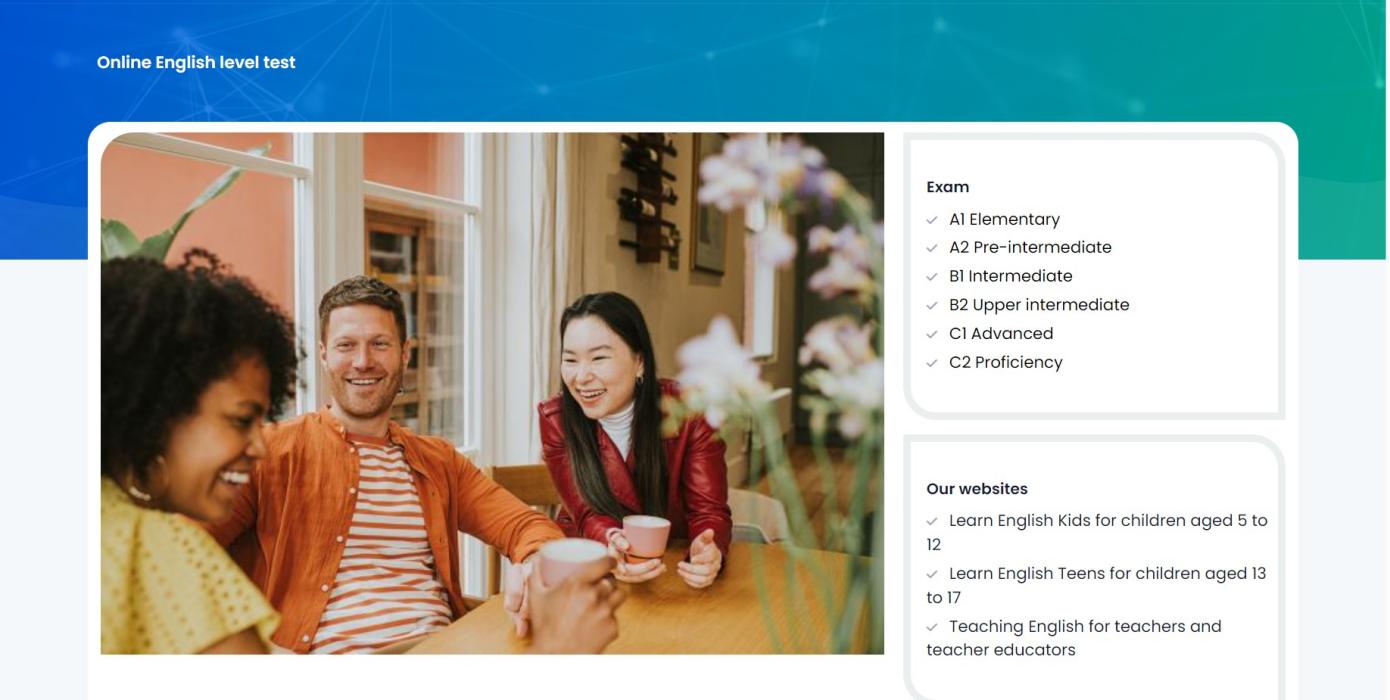


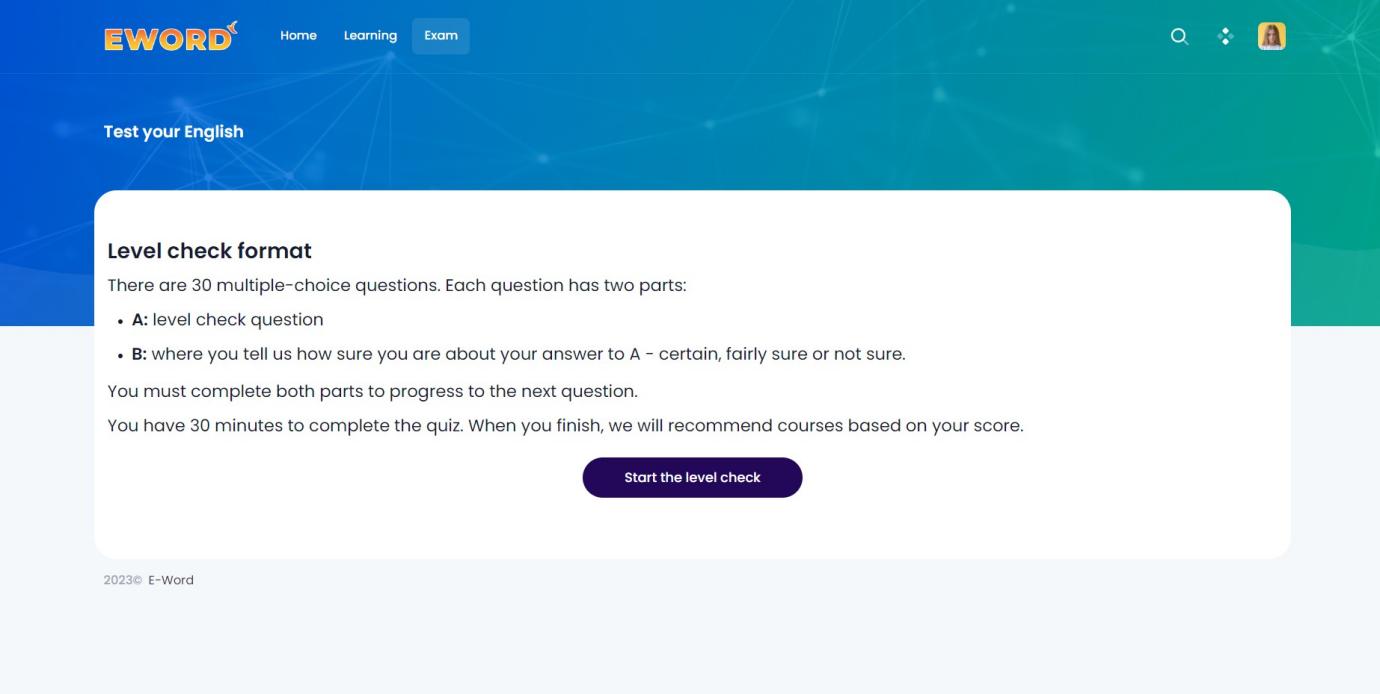
<trang A1 hoặc A2/B1/B2/C1/C1>

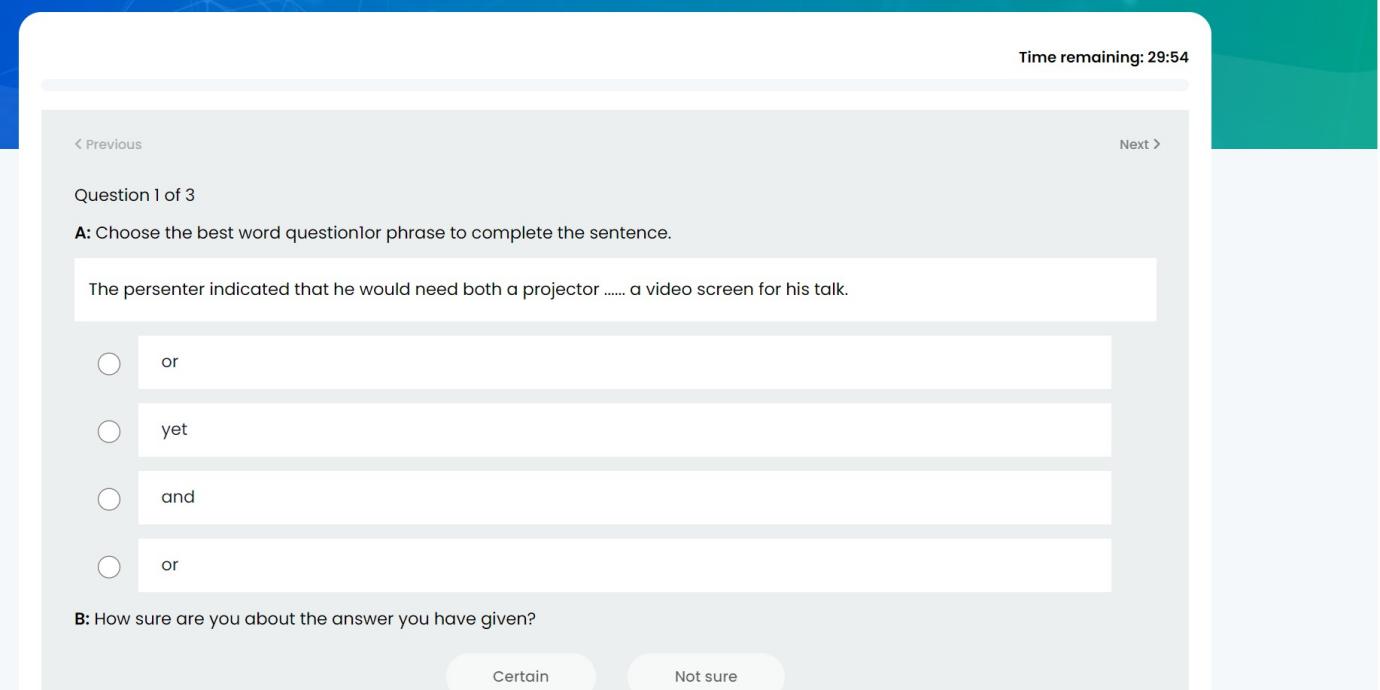


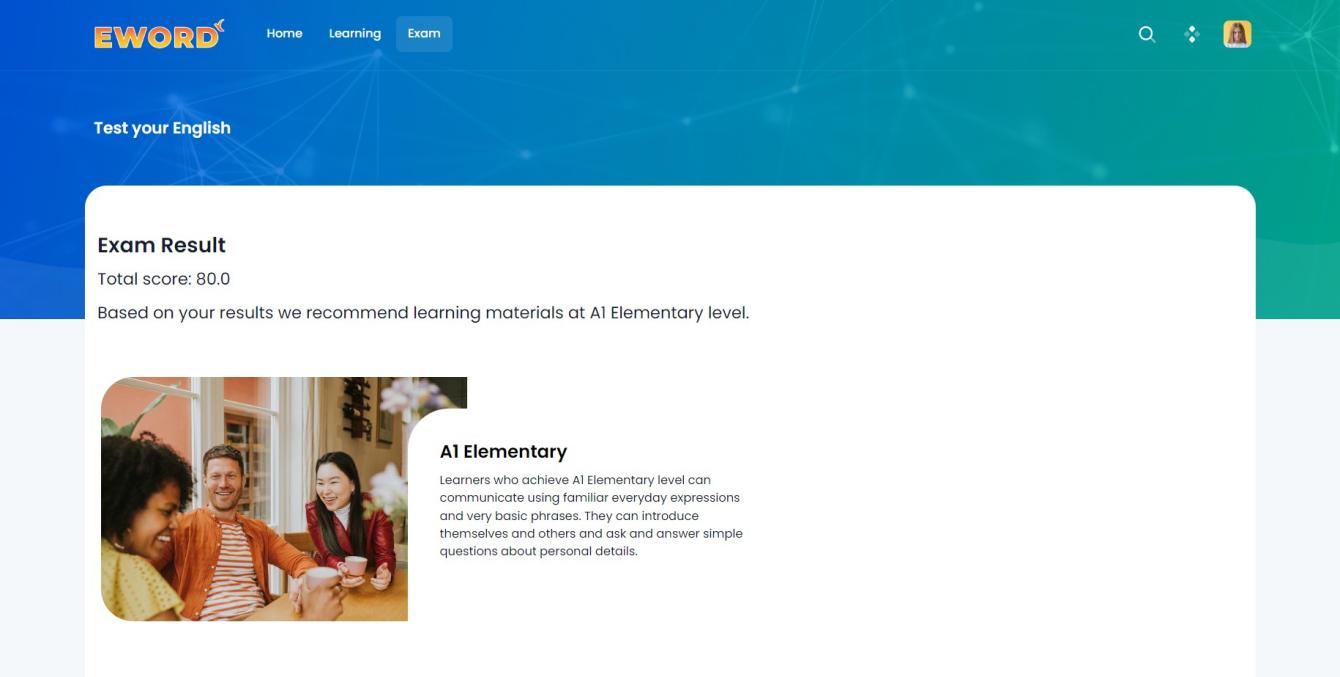
* Online English level test: Bài kiểm tra có thời gian là 30 phút, bao gồm

30 câu hỏi. Mỗi câu hỏi sẽ có hai phần là A - câu hỏi kiểm tra trình độ và B - mức độ chắc chắn đối với câu trả lời đó. Mỗi khi hoàn thành một câu hỏi, thanh tiến độ phía trên sẽ được cập nhật. Thời gian làm bài cũng sẽ được cập nhật liên tục ở phía trên cùng bên phải của bài thi. Sau khi hết thời gian làm bài, hệ thống sẽ tự động submit hoặc khi người dùng đã trả lời hết các câu hỏi hệ thống sẽ đưa ra thông báo submit cho người dùng.



-

* demo bài test.
* hiển thị kết quả.



## Tài liệu tham khảo.

* [https://www.youtube.com/@amigoscode](https://www.youtube.com/%40amigoscode)
* [http://bealdung.com/?fbclid=IwAR1XUhrvu6C\_HDkyzr0CtI6ejlw4nB](http://bealdung.com/?fbclid=IwAR1XUhrvu6C_HDkyzr0CtI6ejlw4nBuLfTMaxbkvfJjrwCmJ2GPxI7EhbbY) [uLfTMaxbkvfJjrwCmJ2GPxI7EhbbY](http://bealdung.com/?fbclid=IwAR1XUhrvu6C_HDkyzr0CtI6ejlw4nBuLfTMaxbkvfJjrwCmJ2GPxI7EhbbY)
* https://viblo.asia/p/luong-di-trong-spring-boot-ORNZqdELK0n