

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN TỐT NGHIỆP
NGÀNH: CÔNG NGHỆ THÔNG TIN
CHUYÊN NGÀNH: CÔNG NGHỆ THÔNG TIN –
NGÔN NGỮ NHẬT

ĐỀ TÀI
WEBSITE HỌC TIẾNG NHẬT E-LEARNING

Người hướng dẫn:	TS BÙI THỊ THANH THANH
Sinh viên thực hiện:	ĐOÀN THỊ PHƯỚC HUYỀN
Số thẻ sinh viên:	102210037
Lớp:	21TCLC_Nhat1

Đà Nẵng, 6/2025

**ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN**

ĐỒ ÁN TỐT NGHIỆP
NGÀNH: CÔNG NGHỆ THÔNG TIN
CHUYÊN NGÀNH: CÔNG NGHỆ THÔNG TIN –
NGÔN NGỮ NHẬT

ĐỀ TÀI
WEBSITE HỌC TIẾNG NHẬT E-LEARNING

Người hướng dẫn:	TS BÙI THỊ THANH THANH
Sinh viên thực hiện:	ĐOÀN THỊ PHƯỚC HUYỀN
Số thẻ sinh viên:	102210037
Lớp:	21TCLC_Nhat1

Đà Nẵng, 6/2025

BẢN TÓM TẮT BẢNG TIẾNG NHẬT

日本語学習 E ラーニングウェブサイト

Japanese Learning E-Learning Website

名前：DOAN THI PHUOC HUYEN (ドアン・ティ・フオック・フエン)

学生番号：102210037

指導教官：BUI THI THANH THANH

所属：ダナン大学・工科大学・情報学部

Abstract: This project introduces an E-learning website designed to support Japanese language learners with flexible and accessible online study. The system offers various learning resources such as video lessons, Kanji writing practice, and grammar exercises. Notably, it integrates an AI-powered Kanji lookup feature, allowing users to identify and learn Kanji characters efficiently. Built on a client-server architecture, the website ensures stability, security, and scalability. It also encourages interaction between learners and teachers through discussion forums, creating an open and effective learning environment.

1. はじめに

現在、ベトナムを中心に日本語学習の需要が高まっていますが、時間や場所の制約、教材へのアクセスの難しさが課題です。本研究では、インターネット環境があればどこからでも利用可能な日本語学習 E ラーニングサイトを開発し、学習者が柔軟に学べる環境を提供します。

本システムは学習者に応じた学習環境と教師との双方向コミュニケーション機能を備え、AI 技術を用いた漢字検索で効率的な語彙習得を支援します。

2. システム分析、設計

2.1. 機能分析

2.1.1. ユーザー

ユーザー認証：

- メールアドレスとパスワード、または Google アカウントでログイン
- メールアドレス、ユーザー名、パスワードでアカウント登録
- 登録後、メールでアカウントを有効化
- パスワードを忘れた場合：登録済みのアカウントに対して、パスワードリセット用のリンクをメール送信

コース管理：

- コース名、レベル（例：N5～N1）、カテゴリー（文法、語彙、模擬試験など）で検索
- カテゴリー別にコースを閲覧

- コースの詳細内容、講師情報、料金などを確認

- 支払いを行い、コースに登録して参加
- 登録済みのコース一覧を閲覧
- コースのレビューを閲覧し、参加後に評価を投稿

- 違反コンテンツを発見した場合に報告
- 各コースのディスカッションフォーラムに参加

学習および教材：

- 講師が作成したコース内のレッスン（動画、漢字学習、語彙フラッシュカード、練習問題など）にアクセス
- 学習進捗（完了したレッスン、学習時間、ステータスなど）を確認

学習支援ツール：

- 漢字語彙の検索
- 日本語文章の翻訳
- 日本語文法の誤りチェック

個人情報およびインタラクション：

- プロフィールの閲覧・編集
- 講師の情報および担当コースの閲覧
- 講師をフォローし、新しいコース開講の通知を受け取る

2.1.2. 教師

アカウント：

- 登録およびログイン
- パスワード変更
- 個人情報の閲覧・更新

コンテンツ管理：

- コースの作成、編集、削除

- レッソンの作成および管理（動画、漢字、フラッシュカード、練習問題を含む）

- 学習者からのコース参加リクエスト管理

統計：

- 参加学習者数の確認
- コース収益の追跡
- 担当コースの概要レポート閲覧

2.1.3. 管理者

認証：

- 管理システムへのログイン

システム管理：

- 教師リストの管理
- 学習者の管理
- コースカテゴリーの管理・分類
- 利用規約およびプラットフォームポリシーの管理

コンテンツ管理：

- システム内全コースの管理
- ユーザーからの違反報告の審査および対応

統計：

- ユーザー数、教師数、コース数の統計
- 時期別システム全体の収益統計

2.2. ユースケース図

2.2.1. 全体ユースケース図

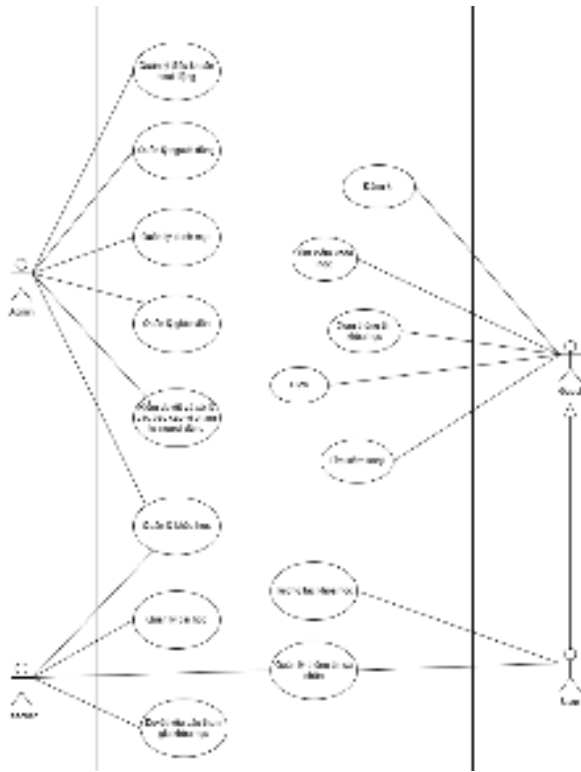


図1 全体ユースケース図

2.3. データベース図

2.3.1. ER図

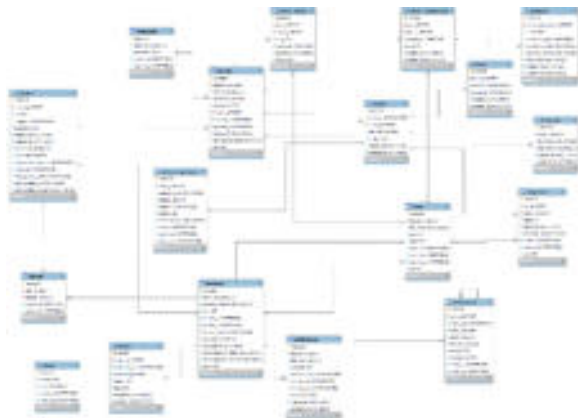


図2 ER図

2.3.2. リレーショナルモデル



図3 リレーショナルモデル

2.4. クラス図



図4 管理者ユースケース図

3. システムの実装

3.1. システム構成

3.1.1. 概要

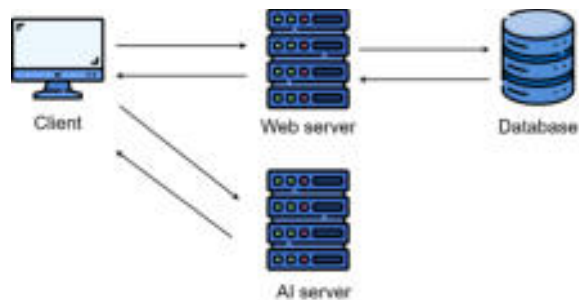


図5 概要

現在、本モデルは ETL Character Database の中でも、ETL-9G データセットを用いて学習されています。



図6 データセット

3.1.2. 技術とテクノロジー

a. バックエンド

- 使用言語：Ruby
- フレームワーク：Ruby on Rails
- データベース：MySQL

b. AI

- モデル：MobileNet
- 使用言語：Python
- フレームワーク：Flask

c. フロントエンド

- 使用技術：TSX、TailwindCSS、TypeScript、HTML
- フレームワーク：Next.js

d. デプロイ

- コンテナ技術：Docker
- 実行環境：Microsoft Azure 仮想マシン (Virtual Machine)

3.2. AI サーバの実装

3.2.1. データセット

3.2.2. データ前処理

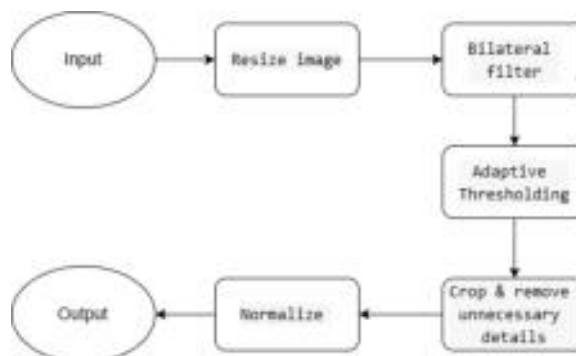


図7 データ前処理

3.2.3. トレーニング結果

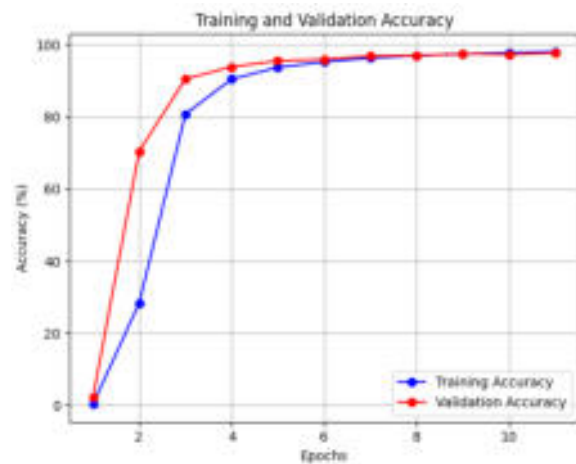


図8 トレーニング精度のグラフ

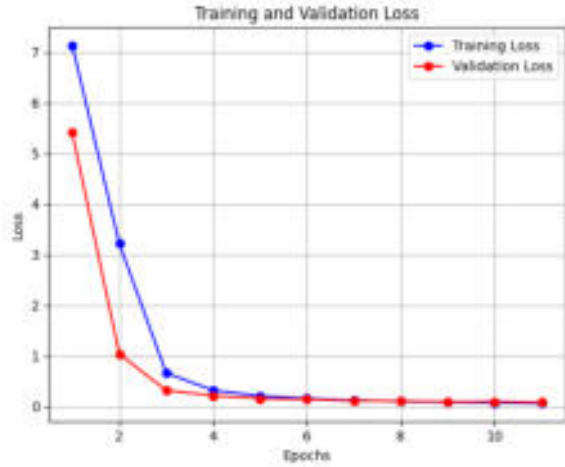


図9 トレーニング損失のグラフ

3.3. Web システムの実装

ログイン

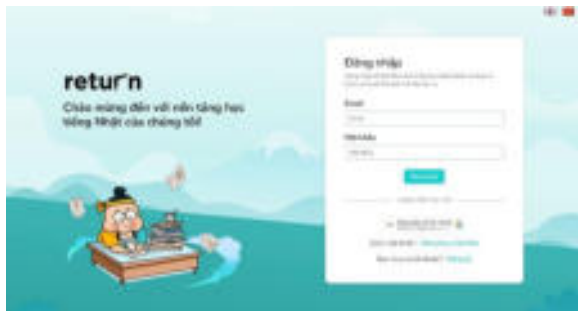


図10 ログイン

ホーム画面



図11 ホーム画面

漢字検索



図12 ログイン

マイコース画面



図13 マイコース画面

コースの詳細

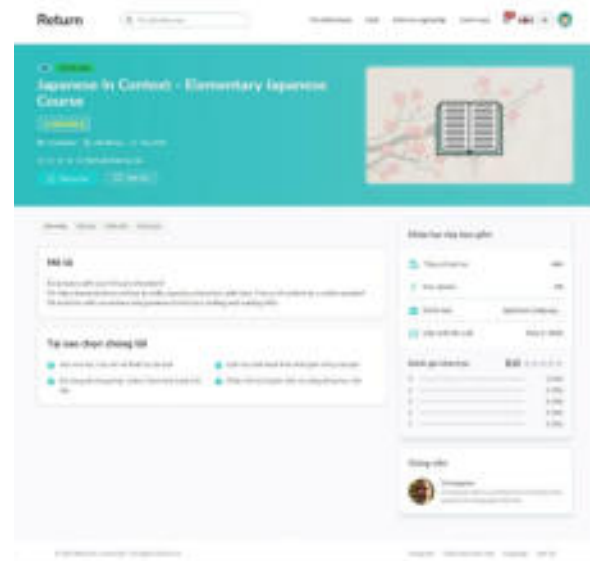


図14 コースの詳細

レッスン画面



図 15 レッスン画面

4. 結論と開発の方向性

4.1. 結論

本日本語学習システムは、動画講義、フラッシュカード、練習問題、手書き漢字の AI 検索機能など多様な機能を備え、初期目標を達成して成功裏に構築・展開されました。Docker、Flask、Rails、Next.js などの最新技術を用い、Azure の仮想環境で安定的かつ拡張可能に運用されています。語彙翻訳や文法チェック機能は外部 API を通じて統合され、ユーザー体験の向上に寄与しています。ユーザーインターフェースは役割別に使いやすく設計されており、効果的な管理・進捗追跡ツールも実装されています。システムは十分なテストを経ており、今後のさらなる発展に向けた堅実な基盤となっています。一方で、多様化する学習者のニーズに応えるための改善・拡張の余地

も多く、将来的な発展の可能性を秘めています。

4.2. 開発の方向性

- AI を活用し学習行動を分析して、個別に最適化された学習プランや提案を提供し、学習体験をパーソナライズ化。

- 教師やチャットボットとのチャット機能、学習者同士のフォーラム、オンライン双方向授業を強化し、インタラクションを向上。

- Azure Kubernetes Service (AKS) や Azure Cognitive Services などの高度な Azure クラウドサービスを活用し、スケーラビリティと AI 処理能力を強化。

- Web システムと連動した iOS/Android 向けモバイルアプリを開発し、いつでもどこでも学習可能に。

- データベース改善、Redis によるキャッシュ、AI パイプラインの最適化でシステムの応答速度を向上し、多数のユーザーでも快適な学習体験を実現。

- 現代的で使いやすく、一貫性のある UI デザインを導入し、多様なデバイスでの操作性とインタラクションを向

TÓM TẮT

Tên đề tài: Website học tiếng Nhật E-Learning

Sinh viên thực hiện: Đoàn Thị Phước Huyền

Số thẻ SV: 102210037

Lớp: 21TCLC_Nhat1

Trong bối cảnh hội nhập, nhu cầu học tiếng Nhật ngày càng tăng, nhưng việc tiếp cận các khóa học chất lượng, linh hoạt và phù hợp cá nhân vẫn còn nhiều hạn chế.

Trước thực trạng trên, việc xây dựng nền tảng học tiếng Nhật trực tuyến đầy đủ tính năng là giải pháp tối ưu. Đề tài “Website học tiếng Nhật E-learning” nhằm phát triển hệ thống học tập hiện đại, thân thiện và đa dạng tài nguyên. Hệ thống này cung cấp các dạng bài giảng phong phú như video bài học, bài học chữ Kanji, bài tập luyện tập, cùng với các công cụ hỗ trợ phát triển kỹ năng ngôn ngữ toàn diện.

Website được thiết kế theo mô hình client-server, áp dụng các công nghệ web tiên tiến nhằm đảm bảo tính ổn định, bảo mật thông tin và khả năng mở rộng linh hoạt trong tương lai. Người học có thể dễ dàng tìm kiếm, đăng ký và theo dõi tiến trình các khóa học phù hợp với trình độ cũng như mục tiêu cá nhân của mình. Bên cạnh đó, giáo viên được hỗ trợ các công cụ quản lý nội dung giảng dạy, cập nhật tài liệu và tương tác trực tiếp với học viên thông qua các diễn đàn thảo luận, tạo điều kiện thuận lợi cho việc trao đổi kiến thức và giải đáp thắc mắc kịp thời.

Ngoài ra, hệ thống còn tích hợp các công cụ hỗ trợ như luyện viết chữ Kanji, tra cứu Kanji và dịch thuật, giúp người học nâng cao kỹ năng một cách toàn diện và hiệu quả hơn. Sự kết hợp hài hòa giữa các yếu tố này tạo nên một môi trường học tập cộng tác, chủ động, phù hợp với nhu cầu đa dạng của người học trong bối cảnh chuyển đổi số đang diễn ra mạnh mẽ.

Việc xây dựng website không chỉ góp phần hiện đại hóa hình thức đào tạo tiếng Nhật mà còn thúc đẩy ứng dụng công nghệ thông tin trong lĩnh vực giáo dục. Qua đó, tạo điều kiện cho người học có thể tiếp cận kiến thức một cách linh hoạt, tiết kiệm chi phí và phù hợp với xu hướng chuyển đổi số đang phát triển mạnh mẽ hiện nay, đồng thời nâng cao chất lượng đào tạo và trải nghiệm học tập cho tất cả các đối tượng người dùng.

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên: Đoàn Thị Phước Huyền

Số thẻ sinh viên: 102210037

Lớp: 21TCLC_Nhat1

Khoa: Công nghệ thông tin

Ngành: CNTT Nhật

1. Tên đề tài đồ án:

Website học tiếng Nhật E-Learning

2. Đề tài thuộc diện: Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện

3. Các số liệu và dữ liệu ban đầu:

Không có

4. Nội dung các phần thuyết minh và tính toán:

- Mở đầu: Giới thiệu lý do chọn đề tài, mục tiêu, phạm vi và phương pháp thực hiện của đồ án.
- Chương 1: Cơ sở lý thuyết: Trình bày các kiến thức nền tảng và công nghệ liên quan phục vụ xây dựng hệ thống.
- Chương 2: Phân tích và thiết kế hệ thống: Phân tích yêu cầu và thiết kế kiến trúc tổng thể, cơ sở dữ liệu và các chức năng chính.
- Chương 3: Triển khai hệ thống và kết quả: Mô tả quá trình hiện thực hóa hệ thống từ thiết kế, lập trình đến kiểm thử và triển khai.
- Kết luận: Tổng kết kết quả đạt được, đánh giá ưu nhược điểm và đề xuất hướng phát triển trong tương lai.

5. Các bản vẽ, đồ thị (ghi rõ các loại và kích thước bản vẽ):

Không có

6. Họ tên người hướng dẫn: TS. Bùi Thị Thanh Thanh

7. Ngày giao nhiệm vụ đồ án:/...../202.....

8. Ngày hoàn thành đồ án:/...../202.....

Đà Nẵng, ngày tháng năm 202

Trưởng Bộ môn

Người hướng dẫn

LỜI NÓI ĐẦU

Trong thế kỷ 21 hiện đại, sự phát triển mạnh mẽ của công nghệ thông tin đã mở ra nhiều cơ hội đổi mới trong cách con người học tập và tương tác. Đặc biệt, trong bối cảnh toàn cầu hóa và nhu cầu nâng cao kỹ năng ngôn ngữ ngày càng tăng, các nền tảng học tập trực tuyến đã trở thành một giải pháp thiết yếu.

Dự án website học tiếng Nhật E-Learning của em được xây dựng với mục tiêu tạo ra một công cụ hỗ trợ học tập hiệu quả, thuận tiện và thông minh cho người học. Một trong những điểm nổi bật của hệ thống là việc tích hợp trí tuệ nhân tạo (AI) để triển khai tính năng tra cứu chữ viết tay Kanji (Hán tự), giúp người học nhanh chóng tra cứu ký tự thông qua thao tác viết tay – một thách thức phổ biến trong quá trình học tiếng Nhật.

Bên cạnh đó, hệ thống được thiết kế theo kiến trúc client-server linh hoạt, cho phép người dùng truy cập từ nhiều thiết bị khác nhau, đảm bảo khả năng học tập mọi lúc, mọi nơi. Điều này góp phần nâng cao tính tiện dụng và đáp ứng nhu cầu học tập ngày càng đa dạng trong thời đại số.

Em xin gửi lời cảm ơn chân thành đến Trường Đại học Bách Khoa – Đại học Đà Nẵng và quý thầy cô khoa Công nghệ Thông tin đã tạo điều kiện để em học tập và thực hiện đề án này. Đặc biệt, em xin bày tỏ lòng biết ơn sâu sắc đến cô Bùi Thị Thanh Thanh, người đã tận tình hướng dẫn, định hướng chuyên môn và luôn đặt ra những thử thách để em có cơ hội học hỏi và hoàn thiện dự án một cách tốt nhất.

Em hy vọng rằng dự án này sẽ đóng góp một phần nhỏ vào việc lan tỏa niềm yêu thích đối với ngôn ngữ và văn hóa Nhật Bản, đồng thời thể hiện tiềm năng ứng dụng công nghệ trong lĩnh vực giáo dục.

Em xin chân thành cảm ơn!

Sinh viên thực hiện

Đoàn Thị Phước Huyền

CAM ĐOAN

Em xin cam đoan

1. Nội dung trong đồ án này em thực hiện dưới sự hướng dẫn trực tiếp của TS. Bùi Thị Thanh Thanh. Các kết quả trong đồ án đều được thực hiện trong quá trình em đang học tập tại Khoa Công Nghệ Thông Tin, trường Đại học Bách Khoa Đà Nẵng.
2. Các tham khảo dùng trong đồ án đều được trích dẫn rõ ràng tên tác giả, tên công trình, thời gian, địa điểm công bố, được kèm với đường dẫn hợp lệ.
3. Nếu có những sao chép không hợp lệ, vi phạm quy chế đào tạo, em xin chịu hoàn toàn trách nhiệm.

Đà Nẵng, ngày.....tháng.....năm 2025

Sinh viên thực hiện

Đoàn Thị Phước Huyền

MỤC LỤC

TÓM TẮT	i
NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP	ii
LỜI NÓI ĐẦU	iii
CAM ĐOAN.....	iv
MỤC LỤC	v
DANH MỤC HÌNH ẢNH.....	ix
DANH MỤC BẢNG BIỂU.....	xii
DANH SÁCH CÁC KÍ HIỆU, CHỮ VIẾT TẮT.....	xiv
MỞ ĐẦU.....	1
1. Mục đích thực hiện.....	1
2. Mục tiêu đề tài.....	1
3. Phạm vi nghiên cứu.....	1
4. Đối tượng nghiên cứu.....	2
5. Phương pháp nghiên cứu.....	2
1. CHƯƠNG 1: CƠ SỞ LÝ THUYẾT	3
1.1. MobileNet	3
1.1.1. Tổng quan.....	3
1.1.2. Kiến trúc và đặc điểm nổi bật.....	3
1.1.3. MobileNetV2.....	6
1.2. Kiến trúc Client – Server	8
1.2.1. Khái niệm	8
1.2.2. Cấu trúc và hoạt động.....	8
1.2.3. Ưu điểm của mô hình Client – Server.....	9
1.2.4. Nhược điểm.....	9
1.2.5. Ứng dụng.....	9

1.3.	NextJS.....	10
1.4.	Cloudinary	11
1.5.	Tailwind CSS	12
1.6.	Ruby on Rails.....	13
1.7.	Flask.....	14
1.8.	MySQL	15
1.9.	Docker.....	17
1.10.	Azure	19
1.10.1.	Tổng quan.....	19
1.10.2.	Azure Virtual Machines (Azure VMs)	19
1.10.3.	Kết hợp Azure và Docker trong triển khai ứng dụng	20
1.10.4.	Lợi ích của việc kết hợp Azure và Docker	21
1.11.	Kiểm thử.....	22
1.11.1.	Rspec	22
1.11.2.	Selenium.....	23
1.12.	Kết luận chương 1	23
2.	CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG.....	24
2.1.	Yêu cầu chức năng.....	24
2.1.1.	Người dùng.....	24
2.1.2.	Giáo viên	24
2.1.3.	Quản trị viên	25
2.2.	Yêu cầu phi chức năng.....	25
2.2.1.	Hiệu năng	25
2.2.2.	Khả năng mở rộng	25
2.2.3.	Tính sẵn sàng và độ tin cậy	26
2.2.4.	Bảo mật.....	26
2.2.5.	Tính khả dụng trên nhiều nền tảng.....	26
2.2.6.	Dễ sử dụng và thân thiện với nhiều người dùng	26
2.2.7.	Khả năng bảo trì và phát triển	26

2.3.	Biểu đồ ca sử dụng.....	26
2.3.1.	Biểu đồ usecase tổng quan	26
2.3.2.	Biểu đồ usecase phân rã	28
2.4.	Lược đồ cơ sở dữ liệu.....	42
2.4.1.	Mô hình thực thể kết hợp ERD	42
2.4.2.	Mô hình quan hệ dữ liệu.....	43
2.4.3.	Phân tích bảng dữ liệu	44
2.5.	Biểu đồ lớp.....	53
2.6.	Sơ đồ hoạt động	53
2.7.	Sơ đồ tuần tự.....	59
2.8.	Kết luận chương 2.....	66
3.	CHƯƠNG 3: TRIỂN KHAI HỆ THỐNG VÀ KẾT QUẢ	68
3.1.	Cấu trúc hệ thống.....	68
3.1.1.	Tổng quan.....	68
3.1.2.	Luồng xử lý	68
3.1.3.	Kỹ thuật công nghệ.....	69
3.2.	Triển khai AI server	69
3.2.1.	Tập dữ liệu.....	69
3.2.2.	Tiền xử lý dữ liệu	71
3.2.3.	Quy trình huấn luyện.....	73
3.2.4.	Kết quả huấn luyện.....	80
3.2.5.	Triển khai mô hình bằng Flask.....	81
3.3.	Triển khai hệ thống lên môi trường thực tế	83
3.3.1.	Công nghệ triển khai sử dụng.....	83
3.3.2.	Các bước triển khai toàn hệ thống.....	83
3.4.	Kết quả.....	84
3.4.1.	Giao diện học viên.....	84
3.4.2.	Giao diện giáo viên.....	100
3.4.3.	Giao diện quản trị viên	108

3.5.	Kiểm thử hệ thống	114
3.5.1.	Phạm vi kiểm thử	114
3.5.2.	Unit test	115
3.5.3.	Automatic test.....	116
3.6.	Kết luận chương 3.....	120
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....		121
	Kết luận.....	121
	Hướng phát triển	122
TÀI LIỆU THAM KHẢO		123
PHỤ LỤC 1		125
PHỤ LỤC 2		126

DANH MỤC HÌNH ẢNH

Hình- 1-1: Mô tả Standard Convolutions	4
Hình- 1-2: Minh họa Depthwise Convolution.....	5
Hình- 1-3: Minh họa Pointwise Convolution	6
Hình- 1-4: Minh họa Inverted Residual Block.....	7
Hình- 1-5: Kết quả thực nghiệm hàm phi tuyến tính và tuyến tính tại layer input và output	8
Hình- 1-6: Minh họa kiến trúc Client-Server	10
Hình- 1-7: Minh họa SSR.....	10
Hình- 1-8: Luồng tải ảnh/ phương tiện lên Clouinary	12
Hình- 1-9: Flask.....	15
Hình- 1-10: Kiến trúc docker	18
Hình- 1-11: Kiến trúc Azure virtual machine.....	20
Hình- 1-12: Triển khai ứng dụng kết hợp Docker và Azure virtual machine.....	21
Hình- 2-1: Biểu đồ usecase tổng quan.....	27
Hình- 2-2: Biểu đồ usecase user	28
Hình- 2-3: Biểu đồ usecase teacher	29
Hình- 2-4: Biểu đồ usecase admin.....	30
Hình- 2-5: Mô hình thực thể kết hợp ERD.....	43
Hình- 2-6: Mô hình quan hệ dữ liệu	44
Hình- 2-7: Biểu đồ lớp.....	53
Hình- 2-8: Sơ đồ hoạt động đăng kí	54
Hình- 2-9: Sơ đồ hoạt động đăng nhập.....	55
Hình- 2-10: Sơ đồ hoạt động thanh toán khóa học.....	56
Hình- 2-11: Sơ đồ hoạt động cập nhật thông tin cá nhân	57
Hình- 2-12: Sơ đồ hoạt động học bài học.....	58
Hình- 2-13: Sơ đồ hoạt động tạo khóa học.....	59
Hình- 2-14: Sơ đồ tuần tự đăng kí và kích hoạt mail	60
Hình- 2-15: Sơ đồ tuần tự đăng nhập và xác thực người dùng	61
Hình- 2-16: Sơ đồ tuần tự thanh toán khóa học	63
Hình- 2-17: Sơ đồ tuần tự cập nhật thông tin cá nhân.....	63
Hình- 2-18: Sơ đồ tuần tự học bài học	65
Hình- 2-19: Sơ đồ tuần tự giáo viên tạo khóa học.....	65
Hình- 2-20: Sơ đồ tuần tự tạo bài học mới	66
Hình- 3-1: Tổng quan hệ thống	68

Hình- 3-2: Chữ viết được trích từ tập dataset ETL Character Database	70
Hình- 3-3: Minh họa hiệu quả bộ lọc Bilateral.....	72
Hình- 3-4: Sơ đồ khối các bước tiền xử lý dữ liệu	73
Hình- 3-5: Kết quả sau khi xử lý ảnh	73
Hình- 3-6: Sơ đồ khối minh họa Inverted Residual Block	76
Hình- 3-7: Sơ đồ khối kiến trúc MobileNetV2.....	78
Hình- 3-8: Đồ thị train accuracy.....	80
Hình- 3-9: Đồ thị train loss.....	81
Hình- 3-10: Giao diện trang chủ.....	85
Hình- 3-11: Giao diện tìm kiếm khóa học	86
Hình- 3-12: Giao diện dịch.....	87
Hình- 3-13: Giao diện kiểm tra ngữ pháp	87
Hình- 3-14: Giao diện tra cứu chữ Kanji.....	88
Hình- 3-15: Giao diện xem khóa học theo danh mục.....	88
Hình- 3-16: Giao diện đăng kí.....	89
Hình- 3-17: Giao diện đăng nhập	89
Hình- 3-18: Giao diện thông tin cá nhân	90
Hình- 3-19: Giao diện khóa học đã tham gia	91
Hình- 3-20: Giao diện chi tiết khóa học – Mô tả khóa học	92
Hình- 3-21: Giao diện chi tiết khóa học – Danh sách bài học.....	93
Hình- 3-22: Giao diện chi tiết khóa học – Đánh giá học viên.....	93
Hình- 3-23: Giao diện chi tiết khóa học – Diễn đàn	94
Hình- 3-24: Giao diện chi tiết bài học – Video bài giảng.....	95
Hình- 3-25: Giao diện chi tiết bài học – Flashcard	95
Hình- 3-26: Giao diện chi tiết bài học – Luyện viết Kanji.....	96
Hình- 3-27: Giao diện chi tiết bài học – Bài tập luyện tập.....	97
Hình- 3-28: Giao diện trang giới thiệu giáo viên	98
Hình- 3-29: Giao diện bảng điều khiển của người dùng – Tổng quan.....	99
Hình- 3-30: Giao diện bảng điều khiển của người dùng – Thống kê khóa học	99
Hình- 3-31: Giao diện bảng điều khiển của người dùng – Thống kê yêu cầu tham gia khóa học.....	100
Hình- 3-32: Giao diện bảng điều khiển của người dùng – Tiến độ học tập	100
Hình- 3-33: Giao diện đăng kí tài khoản giáo viên	101
Hình- 3-34: Giao diện danh sách khóa học	101
Hình- 3-35: Giao diện tạo khóa học mới.....	102
Hình- 3-36: Giao diện chỉnh sửa khóa học.....	102
Hình- 3-37: Giao diện danh sách yêu cầu tham gia khóa học.....	103

Hình- 3-38: Giao diện danh sách khóa học	103
Hình- 3-39: Giao diện tạo khóa học – Thông tin cơ bản	104
Hình- 3-40: Giao diện tạo khóa học – Danh sách Kanji	104
Hình- 3-41: Giao diện tạo khóa học – Danh sách flashcard.....	105
Hình- 3-42: Giao diện tạo khóa học – Bài tập.....	105
Hình- 3-43: Giao diện chỉnh sửa bài học.....	106
Hình- 3-44: Giao diện bảng điều khiển của giáo viên – Tổng quan	107
Hình- 3-45: Giao diện bảng điều khiển của giáo viên – Tài chính.....	107
Hình- 3-46: Giao diện bảng điều khiển của giáo viên – Yêu cầu đăng kí khóa học	108
Hình- 3-47: Giao diện bảng điều khiển của giáo viên – Khóa học	108
Hình- 3-48: Giao diện danh sách khóa học	109
Hình- 3-49: Giao diện danh sách giáo viên	110
Hình- 3-50: Giao diện danh sách người dùng	110
Hình- 3-51: Giao diện danh sách danh mục	111
Hình- 3-52: Giao diện danh sách điều khoản hợp đồng giáo viên	111
Hình- 3-53: Giao diện danh sách báo cáo	112
Hình- 3-54: Giao diện bảng điều khiển của quản trị viên - Tổng quan.....	112
Hình- 3-55: Giao diện bảng điều khiển của quản trị viên - Tài chính.....	113
Hình- 3-56: Giao diện bảng điều khiển của quản trị viên - Người dùng.....	113
Hình- 3-57: Giao diện bảng điều khiển của quản trị viên - Khóa học	114
Hình- 3-58: Kết quả unit test model	116
Hình- 3-59: Kết quả unit test controller	116
Hình- 3-60: Kết quả kiểm thử tự động trang tạo khóa học	117
Hình- 3-61: Kết quả kiểm thử tự động trang tạo bài học	118
Hình- 3-62: Kết quả kiểm thử tự động trang đăng kí	119
Hình- 3-63: Kết quả kiểm thử tự động trang đăng nhập	120

DANH MỤC BẢNG BIỂU

Bảng- 2-1: Đặc tả Use Case: Đăng ký tài khoản.....	31
Bảng- 2-2: Đặc tả Use Case: Đăng nhập.....	31
Bảng- 2-3: Đặc tả Use Case: Quên mật khẩu.....	32
Bảng- 2-4: Đặc tả Use Case: Tìm kiếm khóa học.....	33
Bảng- 2-5: Đặc tả Use Case: Xem danh sách khóa học theo danh mục.....	34
Bảng- 2-6: Đặc tả Use Case: Xem thông tin chi tiết khóa học.....	34
Bảng- 2-7: Đặc tả Use Case: Học bài học.....	35
Bảng- 2-8: Đặc tả Use Case: Quản lý thông tin cá nhân.....	36
Bảng- 2-9: Đặc tả Use Case: Dịch văn bản.....	37
Bảng- 2-10: Đặc tả Use Case: Tìm kiếm Kanji.....	37
Bảng- 2-11: Đặc tả Use Case: Tạo khóa học.....	38
Bảng- 2-12: Đặc tả Use Case: Sửa khóa học.....	39
Bảng- 2-13: Đặc tả Use Case: Xóa khóa học.....	40
Bảng- 2-14: Đặc tả Use Case: Duyệt yêu cầu tham gia khóa học.....	40
Bảng- 2-15: Đặc tả Use Case: Xem thống kê.....	41
Bảng- 2-16: Bảng accounts.....	44
Bảng- 2-17: Bảng categories.....	45
Bảng- 2-18: Bảng courses.....	45
Bảng- 2-19: Bảng lessons.....	46
Bảng- 2-20: Bảng follows.....	46
Bảng- 2-21: Bảng users.....	47
Bảng- 2-22: Bảng teachers.....	47
Bảng- 2-23: Bảng progresses.....	48
Bảng- 2-24: Bảng kanjis.....	48
Bảng- 2-25: Bảng flashcards.....	48
Bảng- 2-26: Bảng course_ratings.....	49
Bảng- 2-27: Bảng course_assignments.....	49
Bảng- 2-28: Bảng exercise_questions.....	50
Bảng- 2-29: forum_posts.....	50
Bảng- 2-30: Bảng Notifications.....	50
Bảng- 2-31: Bảng Payments.....	51
Bảng- 2-32: Bảng Reports.....	52
Bảng- 2-33: Bảng terms.....	52
Bảng- 3-1: Thông tin tệp dữ liệu trong ETL-9G.....	70

Bảng- 3-2: Bảng cấu hình các khối Inverted Residual.....	77
Bảng- 3-3: Kết quả đánh giá mô hình trên tập test.....	81

DANH SÁCH CÁC KÍ HIỆU, CHỮ VIẾT TẮT

Ký hiệu/ Chữ viết tắt	Viết tắt của	Diễn giải
E-learning	Electronic Learning	Học trực tuyến
AI	Artificial Intelligence	Trí tuệ nhân tạo
Kanji		Hán tự (Chữ Hán sử dụng trong tiếng Nhật)
Flashcard		Thẻ ghi nhớ
User		Người dùng
Teacher		Giáo viên
Admin		Quản trị viên
API	Application Programming Interface	Giao tiếp phần mềm
HTTP	HyperText Transfer Protocol	Giao thức web
JSON	JavaScript Object Notation	Dữ liệu JSON
UI	User Interface	Giao diện
URL	Uniform Resource Locator	Địa chỉ web
CNN	Convolutional Neural Network	Mạng nơ-ron tích chập
ReLU	Rectified Linear Unit	Hàm kích hoạt ReLU
SSR	Server-Side Rendering	Kết xuất phía máy chủ
SSG	Static Site Generation	Tạo trang tĩnh
ISR	Incremental Static Regeneration	Tái tạo tĩnh tăng dần
CDN	Content Delivery Network	Mạng phân phối nội dung
MVC	Model - View - Controller	Mô hình - Giao diện - Điều khiển

REST	Representational State Transfer	API kiểu REST
ORM	Object-Relational Mapping	Ánh xạ đối tượng-quan hệ
CSRF	Cross-Site Request Forgery	Giả mạo yêu cầu liên trang
XSS	Cross-Site Scripting	Tấn công kịch bản chéo trang
CORS	Cross-Origin Resource Sharing	Chia sẻ tài nguyên chéo miền
JWT	JSON Web Token	Mã web định dạng JSON
DBMS	Database Management Systems	Hệ quản trị cơ sở dữ liệu
RDBMS	Relational Database Management Systems	Hệ quản trị cơ sở dữ liệu quan hệ
SQL	Structured Query Language	Ngôn ngữ truy vấn có cấu trúc
CRUD	Create – Read – Update - Delete	Tạo – Đọc – Cập nhật – Xoá
SSL	Secure Sockets Layer	Lớp bảo mật socket
TLS	Transport Layer Security	Bảo mật tầng truyền tải
CI/CD	Continuous Integration/Continuous Deployment	Tích hợp liên tục / Triển khai liên tục
PaaS	Platform as a Service	Nền tảng dịch vụ
Azure VMs	Azure Virtual Machines	Máy ảo Azure
CPU	Central Processing Unit	Bộ vi xử lý
RAM	Random Access Memory	Bộ nhớ truy cập ngẫu nhiên
IP	Internet Protocol	Giao thức Internet
DNS	Domain Name System	Hệ thống tên miền
OAuth2	Open Authorization	Giao thức ủy quyền mở

MỞ ĐẦU

1. Mục đích thực hiện

Trong bối cảnh toàn cầu hoá và hội nhập sâu rộng, tiếng Nhật đang trở thành một trong những ngôn ngữ được ưa chuộng tại Việt Nam, mở ra nhiều cơ hội học tập, việc làm và giao lưu văn hoá. Tuy nhiên, người học thường gặp các trở ngại: thiếu môi trường luyện tập tương tác thực tế; khó tiếp cận nội dung học phong phú, phù hợp trình độ; và thiếu công cụ tự đánh giá tiến độ hiệu quả. Sự bùng nổ của công nghệ E-Learning cùng các mô hình học kết hợp (blended learning) mang lại giải pháp khả thi để khắc phục những hạn chế trên bằng cách cung cấp trải nghiệm học tập linh hoạt, cá nhân hoá.

Đề án “Website Học Tiếng Nhật E-Learning” được xây dựng nhằm đáp ứng nhu cầu đó. Hệ thống kết hợp Next.js ở tầng giao diện với Ruby on Rails ở tầng dịch vụ, cung cấp các chức năng đăng ký, đăng nhập an toàn; lộ trình học theo cấp độ JLPT; luyện viết và nhận diện Kanji bằng AI; kiểm tra ngữ pháp tự động; kho Kanji có ví dụ minh hoạ; cùng cơ chế theo dõi tiến độ cho từng người dùng. Nền tảng phục vụ ba nhóm đối tượng: người tự học, giáo viên và quản trị viên với kiến trúc phân vai rõ ràng, giúp quản lý nội dung cũng như tương tác dạy-học trở nên hiệu quả và dễ dàng.

2. Mục tiêu đề tài

Mục tiêu của đề tài là xây dựng một nền tảng học tiếng Nhật trực tuyến hiện đại, chuyên biệt, giúp người học vượt qua các khó khăn thường gặp trong quá trình học tập. Hệ thống sẽ cung cấp các công cụ hỗ trợ toàn diện, từ học lý thuyết đến thực hành, giúp người dùng nâng cao trình độ tiếng Nhật một cách hiệu quả, linh hoạt và tiết kiệm chi phí.

Cụ thể, đề tài hướng đến:

- Xây dựng một website E-learning chuyên biệt về tiếng Nhật với giao diện thân thiện, dễ sử dụng.
- Cung cấp khóa học tiếng Nhật chất lượng được phân chia theo các chủ đề và cấp độ rõ ràng.
- Tích hợp các công cụ hỗ trợ học tập như tra cứu từ vựng, nhận diện Kanji, luyện viết Kanji bằng AI.
- Tạo môi trường học tập linh hoạt giúp người dùng có thể học mọi lúc, mọi nơi, theo lộ trình cá nhân.

3. Phạm vi nghiên cứu

Phạm vi nghiên cứu sẽ tập trung vào cách thức phát triển và triển khai một dự án trang web học trực tuyến chuyên biệt dành cho tiếng Nhật tích hợp AI, với các ứng dụng cụ thể trong việc nhận diện kanji, luyện viết kanji cho người dùng trong môi trường thử nghiệm.

4. Đối tượng nghiên cứu

Đối tượng nghiên cứu bao gồm hành vi và nhu cầu học tập của người học, phương pháp giảng dạy của giáo viên trên các nền tảng E-Learning, cùng với các công nghệ được sử dụng trong phát triển website và ứng dụng trí tuệ nhân tạo.

5. Phương pháp nghiên cứu

Phương pháp nghiên cứu được áp dụng bao gồm:

- Xây dựng và triển khai sản phẩm
- Trải nghiệm và đánh giá hiệu suất của hệ thống

Trong đó, công nghệ để xây dựng và triển khai sản phẩm bao gồm:

- Giao diện sản phẩm (Frontend): NextJS, ReactJS, Tailwind CSS, Shadcn
- Máy chủ (Backend): Ruby on Rails, Flask
- Cơ sở dữ liệu: PostgreSQL
- Triển khai sản phẩm: Azure
- Quản lý mã nguồn: Git, Github, DockerHub
- Công cụ: Postman, Jira, Visual Studio Code, Docker

Cấu trúc của đề án

- Mở đầu
- Chương 1: Cơ sở lý thuyết
- Chương 2: Phân tích và thiết kế hệ thống
- Chương 3: Triển khai hệ thống và kết quả
- Kết luận

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

1.1. MobileNet

1.1.1. Tổng quan

MobileNet là một họ các mô hình mạng nơ-ron tích chập (CNN) được thiết kế nhằm tối ưu hóa hiệu suất và kích thước mô hình cho các thiết bị có tài nguyên tính toán hạn chế. Được phát triển bởi nhóm nghiên cứu của Google và lần đầu tiên được công bố vào năm 2017, MobileNet hướng đến việc xây dựng những mô hình nhẹ, nhanh nhưng vẫn duy trì được độ chính xác cao trong các bài toán thị giác máy tính như phân loại ảnh, phát hiện đối tượng và nhận diện khuôn mặt [1].

Không giống như các kiến trúc CNN truyền thống như VGGNet hay ResNet, vốn yêu cầu nhiều phép tính và bộ nhớ lớn, MobileNet ứng dụng một kỹ thuật đặc biệt gọi là depthwise separable convolution – tích chập tách biệt chiều sâu giúp giảm đáng kể số lượng tham số và phép toán trong mạng. Đây chính là yếu tố cốt lõi giúp MobileNet đạt được sự cân bằng giữa tốc độ và độ chính xác, khiến nó trở thành lựa chọn phổ biến trong các ứng dụng trí tuệ nhân tạo.

1.1.2. Kiến trúc và đặc điểm nổi bật

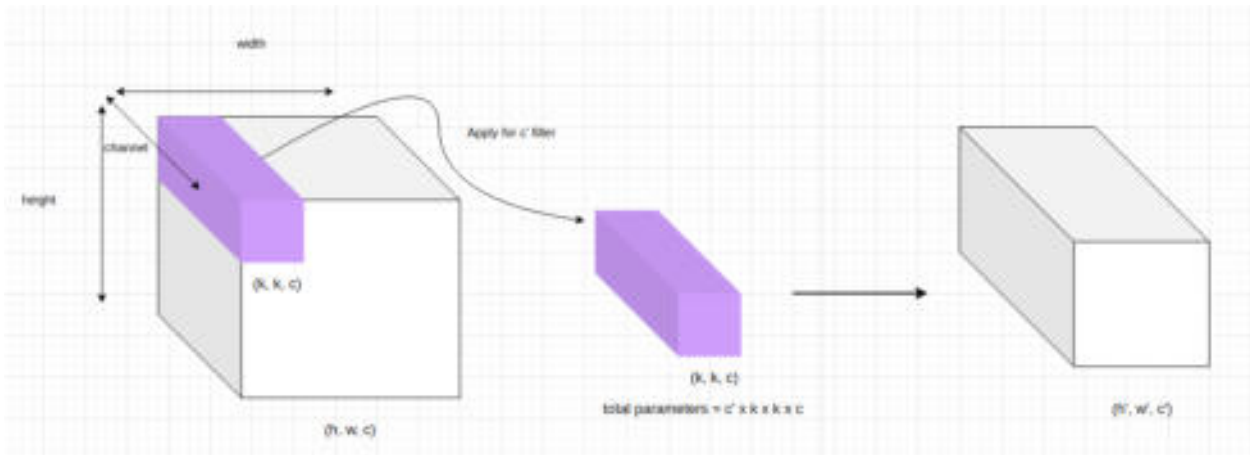
1.1.2.1. Tích chập 2 chiều thông thường (Standard 2D Convolution)

Tích chập 2 chiều thông thường (standard convolution) là kỹ thuật cơ bản và phổ biến nhất trong các mạng nơ-ron tích chập (CNN). Trong phương pháp này, mỗi bộ lọc (filter) thực hiện phép tích chập trên toàn bộ các kênh (channel) của đầu vào để tạo ra một kênh đầu ra (output channel).

Giả sử với một đầu vào kích thước $h \times w \times c$, tích chập thông thường sẽ cần $k \times k \times c$ tham số để thực hiện tích chập trên toàn bộ độ sâu của layers. Mỗi một bộ lọc sẽ tạo ra một ma trận output kích thước $h' \times w' \times 1$. Áp dụng c' bộ lọc khác nhau ta sẽ tạo ra đầu ra có kích thước $h' \times w' \times c'$ mà ma trận output khi áp dụng tích chập trên từng bộ lọc sẽ được concatenate theo chiều sâu. Khi đó số lượng tham số cần sử dụng cho một tích chập thông thường sẽ là: $c' \times k \times k \times c$.

Số lượng tham số sẽ không đáng kể nếu độ sâu của input channel c là nhỏ, thường là ở các layers ở vị trí đầu tiên. Tuy nhiên khi độ sâu tăng tiến dần về những layers cuối cùng của mạng CNN thì số lượng tham số của mô hình là sẽ một con số không hề nhỏ. Sự gia tăng tham số này tạo ra những mô hình cồng kềnh làm chiếm dụng bộ nhớ và ảnh hưởng tới tốc độ tính toán. AlexNet và VGGNet là những mô hình tiêu biểu sử dụng tích chập 2

chiều thông thường, dẫn đến số lượng tham số rất lớn. Tuy nhiên, độ chính xác của các mô hình này lại thấp hơn so với các mô hình thuộc họ MobileNet, vốn sử dụng ít tham số hơn nhiều lần. Sự khác biệt quan trọng nằm ở việc MobileNet lần đầu tiên áp dụng kiến trúc tích chập tách biệt theo chiều sâu (depthwise separable convolution).



Hình- 1-1: Mô tả Standard Convolutions

1.1.2.2. Tích chập tách biệt chiều sâu (Depthwise Separable Convolution).

Một trong những nguyên nhân chủ yếu làm gia tăng số lượng tham số trong mạng nơ-ron tích chập là do sự phụ thuộc vào độ sâu (số lượng kênh) của đầu vào. Để giải quyết vấn đề này, kiến trúc tích chập tách biệt chiều sâu (Depthwise Separable Convolution) được đề xuất như một cải tiến hiệu quả so với tích chập hai chiều thông thường. Phương pháp này tách quá trình tích chập thành hai bước độc lập: tích chập theo chiều sâu và tích chập điểm, qua đó giúp giảm đáng kể số lượng tham số và chi phí tính toán mà vẫn duy trì được khả năng biểu diễn của mô hình [2].

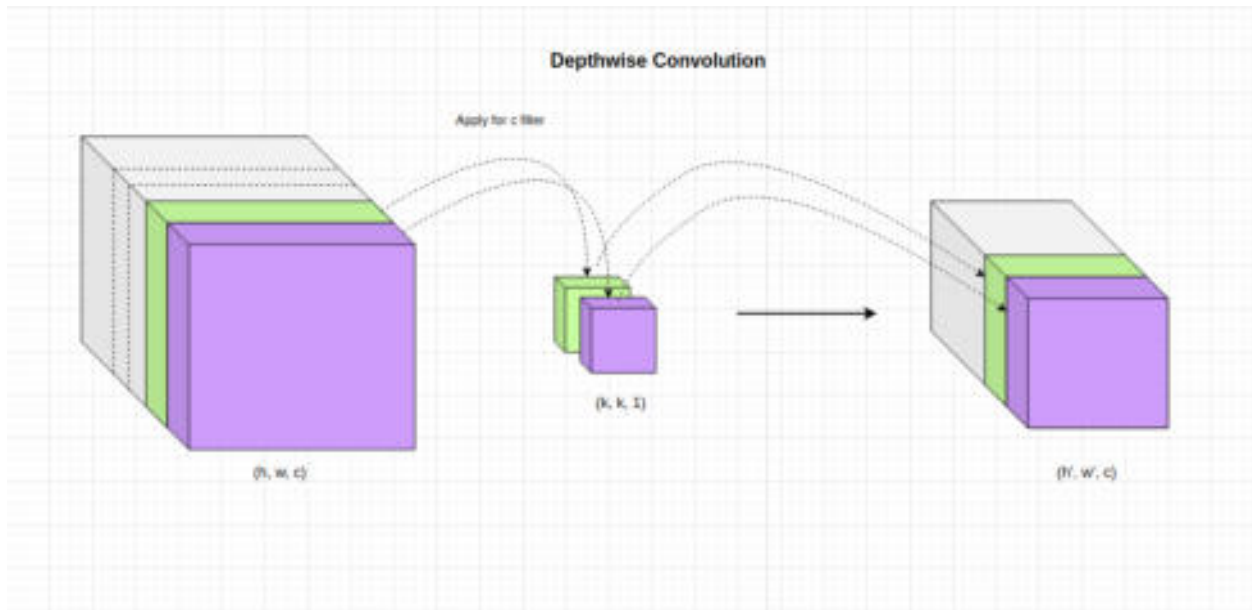
a. Tích chập theo chiều sâu (Depthwise Convolution)

Trong bước đầu tiên, mỗi kênh (channel) của tensor đầu vào ba chiều sẽ được xử lý riêng biệt bởi một bộ lọc riêng, thay vì sử dụng một tập bộ lọc áp dụng cho toàn bộ đầu vào như trong tích chập truyền thống. Cụ thể, với đầu vào có kích thước $h \times w \times c$, sẽ có c bộ lọc, mỗi bộ lọc có kích thước $k \times k$, tương ứng với từng channel.

Việc xử lý độc lập giữa các kênh mang lại ba lợi ích chính:

- Chuyên biệt hóa việc trích xuất đặc trưng: Mỗi channel có thể học các đặc trưng riêng biệt phù hợp với loại thông tin mà channel đó biểu diễn. Ví dụ, trong ảnh RGB, mỗi kênh màu sẽ được học thông qua một bộ lọc riêng, từ đó tăng cường khả năng nhận diện đặc trưng một cách chuyên sâu hơn.

- Giảm chi phí tính toán: Nếu tích chập truyền thống yêu cầu $k \times k \times c$ phép tính cho mỗi vị trí trên bản đồ đặc trưng đầu ra, thì tích chập theo chiều sâu chỉ cần $k \times k$ phép tính cho mỗi channel, từ đó giúp giảm chi phí tính toán xuống.
- Giảm số lượng tham số: Tổng số tham số trong tích chập theo chiều sâu là $k \times k \times c$, thấp hơn đáng kể so với tích chập truyền thống vốn cần $k \times k \times c \times c'$ tham số (với c' là số channel đầu ra).



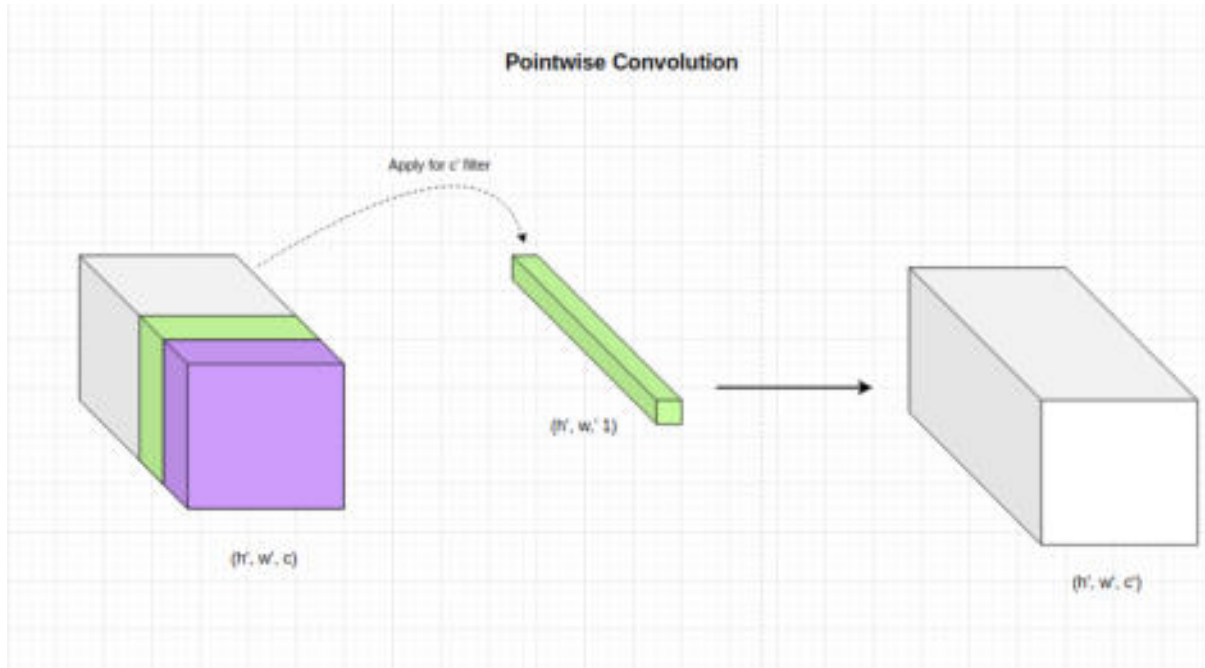
Hình- 1-2: Minh họa Depthwise Convolution

b. Tích chập điểm (Pointwise Convolution)

Bước thứ hai nhằm mục đích kết hợp thông tin giữa các channel và điều chỉnh số lượng channel đầu ra. Quá trình này sử dụng c' bộ lọc có kích thước 1×1 , mỗi bộ lọc hoạt động trên toàn bộ chiều sâu c của đầu vào.

Tích chập điểm giữ nguyên kích thước không gian của đầu vào (chiều rộng và chiều cao) trong khi thay đổi số lượng kênh từ c sang c' . Tổng số tham số cần thiết trong bước này là $c \times c'$.

Kết quả cuối cùng là một tensor đầu ra có kích thước $h' \times w' \times c'$, tương đương với kết quả từ một phép tích chập thông thường, nhưng đạt được với chi phí tính toán và số lượng tham số thấp hơn đáng kể [2].



Hình- 1-3: Minh họa Pointwise Convolution

1.1.3. MobileNetV2

1.1.3.1. Tổng quan

MobileNetV2 là một kiến trúc mạng nơ-ron tích chập nhẹ, được phát triển với mục tiêu tối ưu hóa hiệu suất cho các ứng dụng thị giác máy tính trên thiết bị di động và nhúng. Kế thừa từ MobileNetV1, phiên bản V2 mang đến nhiều cải tiến đáng kể về mặt hiệu quả tính toán cũng như độ chính xác. Hai đặc điểm nổi bật nhất của kiến trúc này là Inverted Residual Block và việc loại bỏ các hàm phi tuyến tại các vị trí đặc biệt trong kiến trúc mạng.

1.1.3.2. Inverted Residual Block

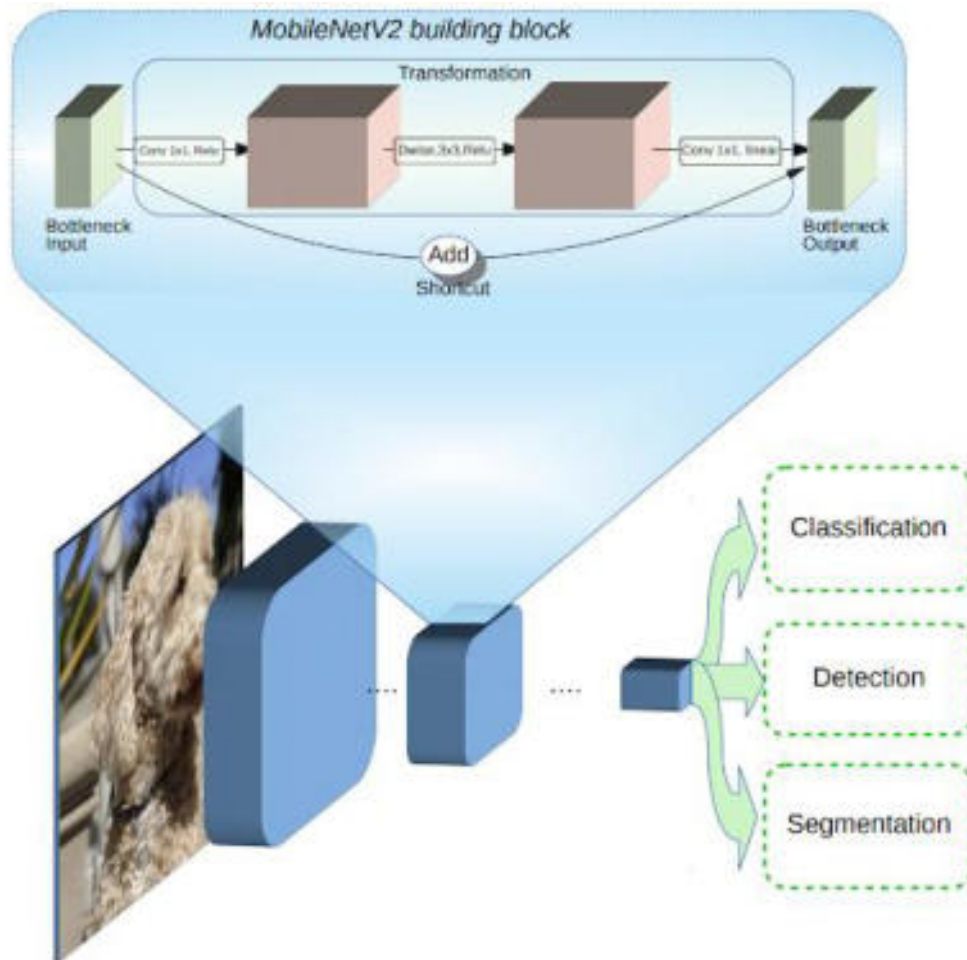
MobileNetV2 áp dụng cơ chế residual connection tương tự như ResNet, cho phép truyền trực tiếp thông tin từ đầu vào đến đầu ra của một khối tích chập, nhờ đó cải thiện việc lan truyền gradient và giảm thiểu hiện tượng mất mát thông tin trong quá trình huấn luyện.

Tuy nhiên, thay vì kết nối tắt giữa các tầng có số lượng kênh lớn như trong ResNet, MobileNetV2 sử dụng các bottleneck layers – các tầng có số lượng kênh thấp hơn ở đầu vào và đầu ra của mỗi khối. Kiến trúc này được gọi là Inverted Residual Block vì nó đảo ngược so với các kiến trúc residual truyền thống: thay vì giảm số lượng kênh ở giữa, MobileNetV2 mở rộng số kênh trong lớp trung gian (expansion layer), sau đó lại thu hẹp ở đầu ra.

Cấu trúc tổng quát của một inverted residual block bao gồm ba bước:

- Expansion layer (tăng số kênh): Sử dụng tích chập 1×1 để mở rộng chiều sâu (số kênh) của đầu vào.
- Depthwise convolution: Áp dụng tích chập tách biệt chiều sâu nhằm trích xuất đặc trưng mà không làm tăng số tham số quá nhiều.
- Projection layer (thu hẹp số kênh): Sử dụng tích chập 1×1 để giảm số kênh về như ban đầu, tạo điều kiện để thực hiện kết nối tắt (nếu có).

Kết nối tắt chỉ được áp dụng khi đầu vào và đầu ra của block có cùng kích thước. Nhờ thiết kế này, MobileNetV2 vừa giữ được tính phi tuyến của mạng, vừa giảm thiểu đáng kể chi phí tính toán [3].



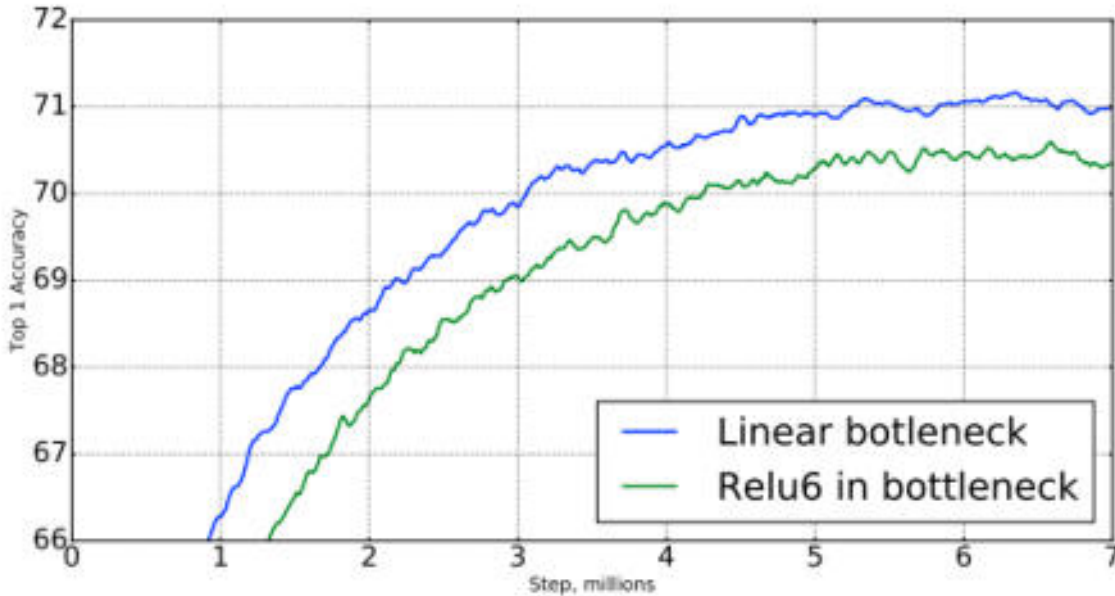
Hình- 1-4: Minh họa Inverted Residual Block

1.1.3.3. Loại bỏ non-linear

Một điểm đổi mới quan trọng khác trong MobileNetV2 là việc loại bỏ các hàm kích hoạt phi tuyến như ReLU tại lớp input và output của các inverted residual block. Việc áp

dùng các hàm phi tuyến ở những vị trí này có thể gây ra mất mát thông tin, đặc biệt là khi số chiều (số kênh) bị thu hẹp – hiện tượng này được gọi là "information bottleneck".

Thay vì sử dụng ReLU, các lớp đầu vào và đầu ra của mỗi block được xử lý bằng phép chiếu tuyến tính (linear projection), giúp bảo toàn tối đa thông tin khi truyền qua kết nối tắt. Kết quả thực nghiệm của tác giả (Sandler et al., 2018) cho thấy việc loại bỏ non-linearity tại các vị trí này giúp cải thiện độ chính xác mô hình đáng kể [4].



Hình- 1-5: Kết quả thực nghiệm hàm phi tuyến tính và tuyến tính tại layer input và output

1.2. Kiến trúc Client – Server

1.2.1. Khái niệm

Mô hình Client – Server là một kiến trúc phân tán trong đó các thiết bị hoặc chương trình được chia thành hai vai trò chính: Client và Server. Trong mô hình này, phía client gửi các yêu cầu đến phía server, còn server chịu trách nhiệm xử lý các yêu cầu đó và phản hồi kết quả cho client [5].

Mô hình này được sử dụng rộng rãi trong các hệ thống mạng máy tính, đặc biệt là trong các ứng dụng web, hệ thống cơ sở dữ liệu và dịch vụ trực tuyến.

1.2.2. Cấu trúc và hoạt động

a. Cấu trúc

- Client: Là bên khởi tạo giao tiếp, thường là giao diện người dùng hoặc ứng dụng frontend. Client gửi yêu cầu đến server thông qua các giao thức mạng (như HTTP/HTTPS)

và chờ phản hồi. Các client thường không chia sẻ tài nguyên giữa nhau và không giao tiếp trực tiếp.

- Server: Là bên nhận yêu cầu và xử lý chúng. Server có thể thực hiện các thao tác như truy xuất cơ sở dữ liệu, xử lý logic nghiệp vụ, xác thực người dùng, v.v. Sau khi xử lý xong, server gửi phản hồi trở lại client. Một server có thể phục vụ nhiều client cùng lúc.

b. Quy trình hoạt động

- Client gửi một request (yêu cầu) đến server.
- Server tiếp nhận và xử lý request.
- Server gửi response (phản hồi) về client.

1.2.3. Ưu điểm của mô hình Client – Server

- Tập trung hóa dữ liệu: Dữ liệu được lưu trữ và quản lý tập trung trên server, giúp dễ dàng sao lưu và bảo mật.
- Khả năng mở rộng tốt: Server có thể phục vụ nhiều client cùng lúc và có thể được nâng cấp phần cứng hoặc tối ưu hiệu suất khi cần.
- Quản lý dễ dàng: Do server quản lý logic và dữ liệu, việc cập nhật hệ thống hoặc triển khai các thay đổi được thực hiện tập trung.

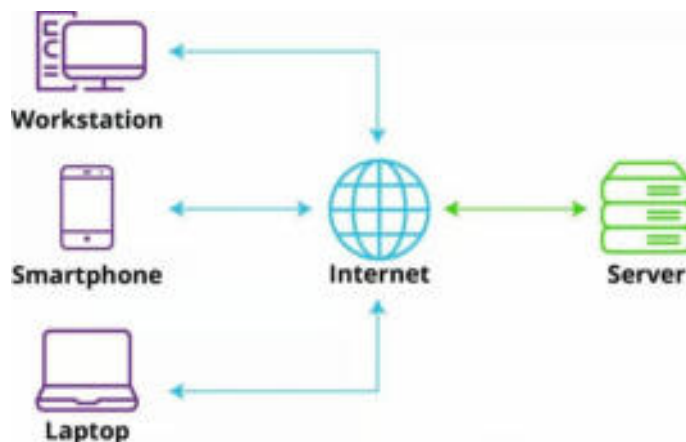
1.2.4. Nhược điểm

- Phụ thuộc vào server: Nếu server gặp sự cố, toàn bộ hệ thống có thể ngừng hoạt động.
- Chi phí đầu tư ban đầu: Cần đầu tư máy chủ có cấu hình đủ mạnh và hệ thống mạng ổn định.
- Tắc nghẽn tại server: Nếu có quá nhiều client truy cập cùng lúc mà server không đủ khả năng xử lý, có thể dẫn đến hiện tượng quá tải.

1.2.5. Ứng dụng

Mô hình client – server là nền tảng cho hầu hết các ứng dụng hiện đại như:

- Ứng dụng web (Frontend – Backend).
- Ứng dụng di động kết nối API.
- Hệ thống quản lý cơ sở dữ liệu.
- Dịch vụ cloud và hệ thống phân tán.



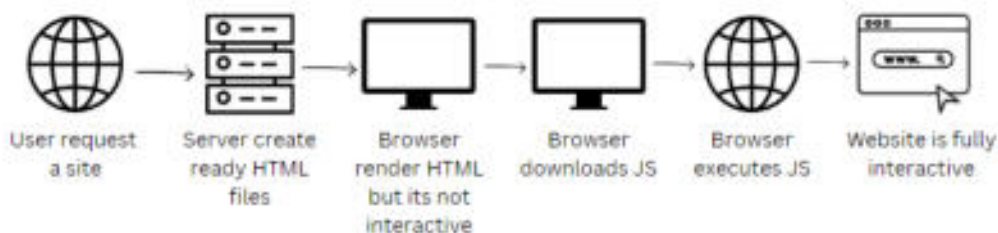
Hình- 1-6: Minh họa kiến trúc Client-Server

1.3. NextJS

Next.js là một framework front-end mạnh mẽ được phát triển dựa trên nền tảng React, nhằm hỗ trợ xây dựng các ứng dụng web hiện đại với hiệu suất cao và khả năng mở rộng linh hoạt. Được phát hành dưới dạng mã nguồn mở bởi công ty Vercel, Next.js tích hợp nhiều tính năng tiên tiến như render phía máy chủ SSR và tạo trang tĩnh SSG, giúp nâng cao tốc độ tải trang, cải thiện trải nghiệm người dùng và tối ưu cho các công cụ tìm kiếm [6].

Một trong những điểm nổi bật của Next.js là khả năng render phía máy chủ. Thay vì tải nội dung thông qua JavaScript ở phía trình duyệt như các ứng dụng React thông thường, SSR cho phép máy chủ thực thi mã JavaScript và trả về một trang HTML hoàn chỉnh ngay từ lần truy cập đầu tiên. Điều này giúp rút ngắn thời gian hiển thị nội dung cho người dùng và tăng cường khả năng lập chỉ mục của các công cụ tìm kiếm. SSR đặc biệt hiệu quả đối với các ứng dụng có nội dung động, cần phản hồi nhanh và tải dữ liệu từ server trong thời gian thực.

SSR (Server-side Rendering)



Hình- 1-7: Minh họa SSR

Bên cạnh đó, Next.js hỗ trợ tạo trang tĩnh thông qua cơ chế Static Site Generation. Với SSG, các trang HTML được tạo ra tại thời điểm biên dịch (build time), sau đó được phân phối như các tệp tĩnh thông thường. Cách tiếp cận này giúp giảm tải cho máy chủ và cải thiện hiệu năng đáng kể, phù hợp với các nội dung ít thay đổi như blog, trang giới thiệu hoặc tài liệu sản phẩm. Next.js phiên bản 14 còn tích hợp cơ chế ISR, cho phép tái tạo các trang tĩnh theo yêu cầu sau khi đã được triển khai, từ đó đảm bảo nội dung luôn cập nhật mà không cần build lại toàn bộ ứng dụng [7].

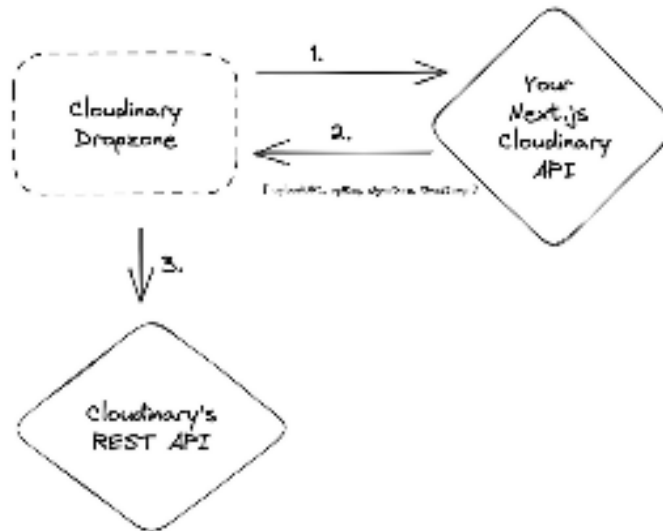
Ngoài khả năng render linh hoạt, Next.js còn tích hợp tính năng tự động phân tách mã (code splitting), giúp mỗi trang chỉ tải những đoạn mã cần thiết thay vì toàn bộ ứng dụng. Điều này góp phần tối ưu hiệu năng và giảm dung lượng dữ liệu tải về. Framework này cũng hỗ trợ định tuyến tự động dựa trên cấu trúc thư mục pages, giúp đơn giản hóa quá trình phát triển. Ngoài ra, việc tích hợp serverless functions và khả năng triển khai trực tiếp lên nền tảng Vercel giúp quá trình xây dựng, triển khai và mở rộng ứng dụng trở nên nhanh chóng và thuận tiện hơn [8].

Tóm lại, Next.js là một giải pháp toàn diện và linh hoạt cho các dự án phát triển ứng dụng web. Với sự kết hợp giữa SSR, SSG, ISR cùng khả năng tối ưu hiệu năng và SEO, Next.js đáp ứng tốt nhu cầu của các ứng dụng hiện đại, đặc biệt là các dự án cần tốc độ cao, dễ bảo trì và tối ưu cho người dùng cũng như công cụ tìm kiếm.

1.4. Cloudinary

Cloudinary là một nền tảng quản lý phương tiện truyền thông (media management platform) trên nền tảng đám mây, cung cấp giải pháp toàn diện cho việc lưu trữ, xử lý, tối ưu hóa và phân phối hình ảnh cũng như video trong các ứng dụng web và di động. Không giống như việc lưu trữ tệp truyền thống trên máy chủ riêng, Cloudinary cho phép các nhà phát triển quản lý nội dung đa phương tiện một cách hiệu quả thông qua hệ thống API mạnh mẽ và giao diện quản trị trực quan [9].

Một trong những điểm nổi bật của Cloudinary là khả năng xử lý hình ảnh và video động, bao gồm thay đổi kích thước, nén, chuyển định dạng, cắt xén, thêm watermark, hay áp dụng các hiệu ứng thị giác... mà không cần chỉnh sửa tệp gốc. Tất cả các thao tác này được thực hiện thông qua các tham số trong URL hoặc thông qua API, giúp giảm tải cho máy chủ backend và tăng tốc độ phản hồi cho người dùng.



Hình- 1-8: Luồng tải ảnh/ phương tiện lên Cloudinary

Bên cạnh đó, Cloudinary tích hợp hệ thống mạng phân phối nội dung (CDN), giúp tối ưu hóa hiệu suất tải hình ảnh và video theo vị trí địa lý người dùng, từ đó cải thiện trải nghiệm người dùng cuối và tiết kiệm băng thông. Cloudinary cũng hỗ trợ nhiều cơ chế bảo mật như kiểm soát truy cập, token xác thực và giới hạn sử dụng tài nguyên.

Với khả năng mở rộng linh hoạt và dễ tích hợp vào các hệ thống hiện đại như React, Next.js, Ruby on Rails, Django hay Node.js, Cloudinary đang được sử dụng rộng rãi trong nhiều lĩnh vực như thương mại điện tử, mạng xã hội, giáo dục và các nền tảng nội dung số. Trong bối cảnh nhu cầu xử lý và quản lý nội dung đa phương tiện ngày càng gia tăng, việc sử dụng các giải pháp như Cloudinary giúp tối ưu hóa quy trình phát triển, giảm chi phí vận hành và nâng cao hiệu quả của hệ thống [10].

1.5. Tailwind CSS

Tailwind CSS là một framework hiện đại theo phương pháp utility-first, hỗ trợ xây dựng giao diện web thông qua các lớp tiện ích nhỏ tương ứng với các thuộc tính CSS phổ biến như màu sắc, khoảng cách, căn lề, kích thước chữ, hiển thị linh hoạt, v.v. Thay vì sử dụng các thành phần giao diện cố định, Tailwind cho phép lập trình viên kết hợp linh hoạt các lớp để thiết kế giao diện trực tiếp trong mã HTML, từ đó tăng tốc độ phát triển, giảm phụ thuộc vào CSS tùy chỉnh và cải thiện khả năng bảo trì mã nguồn. Framework này hỗ trợ thiết kế theo hướng mobile-first và có khả năng tùy biến cao thông qua tệp cấu hình, cho phép mở rộng bảng màu, kích thước hoặc định nghĩa các lớp riêng theo yêu cầu dự án. Với khả năng tích hợp tốt vào các công cụ và framework hiện đại như React, Vue hoặc

Next.js, Tailwind ngày càng được ưa chuộng trong phát triển giao diện web hiện đại nhờ tính linh hoạt, hiệu quả và dễ mở rộng [11].

1.6. Ruby on Rails

Ruby on Rails, hay còn gọi là Rails, là một framework phát triển ứng dụng web mã nguồn mở được xây dựng trên ngôn ngữ lập trình Ruby. Ra đời vào năm 2004, Rails nhanh chóng trở thành một trong những công cụ phổ biến nhất trong lĩnh vực phát triển web nhờ triết lý "Convention over Configuration" (Ưu tiên quy ước hơn cấu hình) và "Don't Repeat Yourself" (Không lặp lại chính mình). Những triết lý này giúp đơn giản hóa quy trình phát triển phần mềm, tăng tốc độ xây dựng hệ thống và giảm thiểu lỗi do lặp lại logic trong mã nguồn [12].

Bên cạnh khả năng xây dựng các ứng dụng web truyền thống theo mô hình MVC, Rails còn hỗ trợ triển khai ở chế độ API-only, trong đó hệ thống chỉ tập trung vào xử lý logic phía máy chủ và cung cấp dữ liệu thông qua giao diện lập trình ứng dụng (API). Khi sử dụng tùy chọn `--api` trong quá trình khởi tạo, ứng dụng Rails sẽ tự động loại bỏ các thành phần liên quan đến giao diện như HTML, CSS, JavaScript và chỉ giữ lại các thành phần cần thiết để xây dựng API RESTful.

Trong mô hình phát triển hiện đại, việc tách biệt giữa frontend và backend ngày càng phổ biến. Rails API được sử dụng như một lớp backend chuyên xử lý dữ liệu và nghiệp vụ, trong khi phần giao diện được xây dựng bằng các framework hiện đại như Next.js, React, hoặc Vue.js. Rails đóng vai trò cung cấp dữ liệu dưới định dạng JSON thông qua các HTTP request, cho phép frontend có thể dễ dàng tương tác và hiển thị thông tin đến người dùng [13].

Một số ưu điểm nổi bật của Ruby on Rails khi triển khai dưới dạng API có thể kể đến như:

- Hiệu quả phát triển cao: Rails cung cấp hệ thống generator và scaffold mạnh mẽ, cho phép tạo nhanh các endpoint API với cấu trúc rõ ràng, tuân theo chuẩn REST.
- Quản lý dữ liệu hiệu quả: Với sự hỗ trợ của ActiveRecord – một ORM mạnh mẽ – việc thao tác cơ sở dữ liệu trở nên dễ dàng và linh hoạt hơn.
- Tổ chức mã nguồn khoa học: Rails giúp phân tách rõ ràng giữa các thành phần như controller, model, route..., từ đó nâng cao khả năng bảo trì và mở rộng.
- Bảo mật tích hợp sẵn: Rails cung cấp các cơ chế bảo vệ khỏi các lỗ hổng phổ biến như CSRF, XSS và SQL injection.

- Hệ sinh thái phong phú: Kho thư viện (gem) của Ruby rất đa dạng, hỗ trợ nhiều tính năng như xác thực người dùng, phân quyền, tải lên tệp, và nhiều công cụ hỗ trợ kiểm thử.
- Khả năng tích hợp cao: Rails API có thể dễ dàng kết nối với các frontend hiện đại thông qua các request HTTP chuẩn, đảm bảo khả năng tương tác mượt mà giữa client và server.

Tóm lại, Ruby on Rails là một lựa chọn phù hợp để xây dựng các hệ thống backend hiện đại. Với cú pháp rõ ràng, cấu trúc chặt chẽ và khả năng mở rộng cao, Rails API cho phép các nhóm phát triển phần mềm nhanh chóng xây dựng các dịch vụ web mạnh mẽ, dễ bảo trì và sẵn sàng tích hợp với nhiều nền tảng giao diện người dùng khác nhau.

1.7. Flask

Flask là một micro web framework mã nguồn mở được viết bằng ngôn ngữ lập trình Python. Được phát triển bởi Armin Ronacher và phát hành lần đầu tiên vào năm 2010, Flask được thiết kế theo triết lý "simple is better than complex", tức ưu tiên sự đơn giản, linh hoạt và dễ mở rộng. Mặc dù là một "micro framework", Flask vẫn đủ mạnh để xây dựng các ứng dụng web hoặc API phục vụ cho nhiều mục đích, từ các trang web nhỏ đến các hệ thống xử lý dữ liệu phức tạp và các dịch vụ trí tuệ nhân tạo [14].

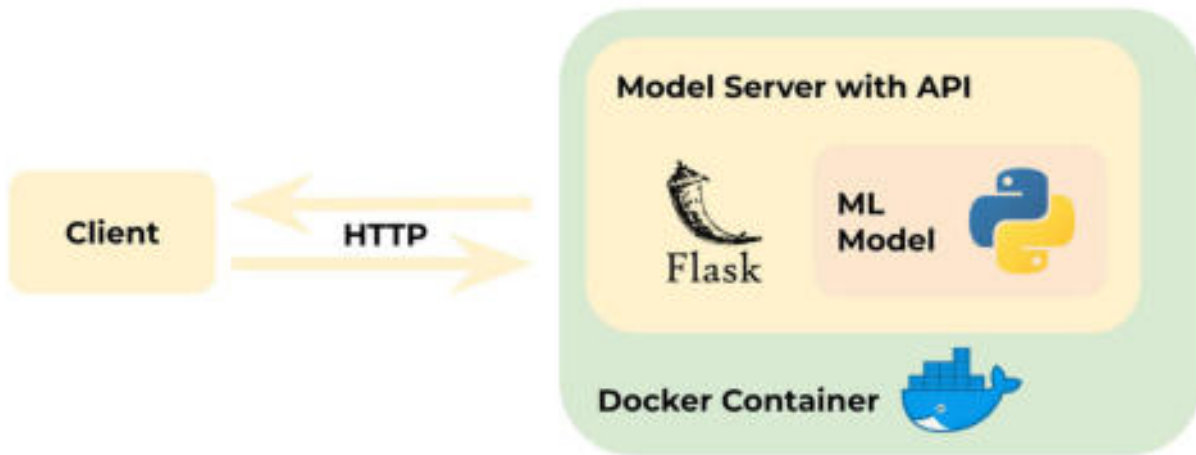
Khi xây dựng AI server, Flask thường được sử dụng như một giao diện trung gian (API layer) giữa người dùng và mô hình học máy. Các mô hình AI (như mô hình xử lý ngôn ngữ tự nhiên, nhận diện hình ảnh, phân loại, dự đoán...) sau khi được huấn luyện có thể được đóng gói và triển khai thông qua các endpoint Flask. Người dùng hoặc ứng dụng frontend chỉ cần gửi yêu cầu (HTTP request) kèm theo dữ liệu đầu vào (input), Flask sẽ tiếp nhận yêu cầu, xử lý bằng mô hình AI và trả về kết quả dưới dạng JSON.

Một số tính năng nổi bật của Flask

- Đơn giản, dễ học và dễ triển khai: Flask có cú pháp rõ ràng, gần gũi với Python thuần, rất phù hợp với sinh viên hoặc các nhóm nhỏ phát triển ứng dụng nhanh.
- Tự do lựa chọn cấu trúc và thư viện: Khác với các framework "nặng" như Django, Flask không áp đặt cấu trúc dự án cứng nhắc, cho phép lập trình viên tùy biến theo nhu cầu.
- Tích hợp dễ dàng với các thư viện AI/ML: Do Flask được viết bằng Python, việc tích hợp với các thư viện phổ biến như scikit-learn, TensorFlow, PyTorch, Transformers... trở nên đơn giản và hiệu quả.

- Xây dựng RESTful API nhanh chóng: Với Flask, có thể dễ dàng định nghĩa các route để nhận request và trả response, rất phù hợp cho các ứng dụng AI inference.
- Hệ sinh thái mở rộng: Có thể mở rộng chức năng thông qua nhiều extension như Flask-RESTful, Flask-CORS, Flask-JWT, v.v [15].

Việc sử dụng Flask giúp quá trình kết nối giữa hệ thống AI và người dùng trở nên dễ dàng và hiệu quả hơn. Ngoài ra, Flask cũng phù hợp để triển khai lên các nền tảng đám mây như Heroku, AWS, hoặc đơn giản là chạy nội bộ qua Docker trong các hệ thống microservice.



Hình- 1-9: Flask

1.8. MySQL

Trong lĩnh vực công nghệ thông tin, dữ liệu đóng vai trò then chốt trong việc vận hành và phát triển các hệ thống phần mềm, đặc biệt là trong bối cảnh chuyển đổi số hiện nay. Việc lưu trữ, tổ chức và truy xuất dữ liệu một cách hiệu quả không chỉ giúp đảm bảo tính chính xác và toàn vẹn của thông tin, mà còn góp phần nâng cao hiệu suất của toàn bộ hệ thống. Để thực hiện điều đó, các hệ quản trị cơ sở dữ liệu (DBMS) đã ra đời, trong đó nổi bật là các hệ quản trị cơ sở dữ liệu quan hệ (RDBMS). Một trong những hệ thống RDBMS phổ biến và được sử dụng rộng rãi nhất hiện nay là MySQL.

MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở, được phát triển lần đầu tiên vào năm 1995 bởi công ty MySQL AB (Thụy Điển), và hiện nay được quản lý bởi tập đoàn Oracle sau khi mua lại Sun Microsystems vào năm 2010. Với triết lý phát triển hướng đến sự ổn định, linh hoạt và dễ sử dụng, MySQL đã nhanh chóng trở thành một giải pháp phổ biến cho việc xây dựng các ứng dụng web, hệ thống quản lý thông tin doanh nghiệp, và các nền tảng xử lý dữ liệu lớn.

Về mặt kiến trúc, MySQL được xây dựng theo mô hình client-server, trong đó máy chủ cơ sở dữ liệu chịu trách nhiệm tiếp nhận và xử lý các truy vấn từ phía máy khách. Các truy vấn được viết bằng ngôn ngữ SQL – ngôn ngữ tiêu chuẩn quốc tế trong việc thao tác và quản lý cơ sở dữ liệu quan hệ. MySQL cho phép người dùng thực hiện đầy đủ các thao tác cơ bản như tạo, đọc, cập nhật và xóa dữ liệu (CRUD), đồng thời hỗ trợ xây dựng các bảng, ràng buộc và quan hệ giữa các thực thể trong hệ thống [16].

Một trong những điểm mạnh nổi bật của MySQL chính là hiệu năng xử lý cao, đặc biệt đối với các hệ thống yêu cầu tốc độ phản hồi nhanh và khối lượng dữ liệu lớn. Bên cạnh đó, MySQL hỗ trợ đa dạng các công cụ lưu trữ (storage engines) như InnoDB, MyISAM, MEMORY... Trong đó, InnoDB là công cụ mặc định với khả năng hỗ trợ giao dịch (transaction), khóa ngoại (foreign key), và đảm bảo tính toàn vẹn của dữ liệu. Khả năng xử lý đồng thời (concurrent access) và tối ưu hóa truy vấn (query optimization) cũng là những yếu tố giúp MySQL hoạt động hiệu quả ngay cả trong môi trường có nhiều người dùng truy cập đồng thời.

MySQL còn cung cấp một loạt các tính năng nâng cao phục vụ cho quản trị và vận hành hệ thống như: cơ chế sao lưu và phục hồi dữ liệu (backup and restore), đồng bộ hóa dữ liệu giữa các máy chủ (replication), phân vùng dữ liệu (partitioning), truy cập thông qua giao thức SSL/TLS nhằm đảm bảo bảo mật, và hệ thống phân quyền chi tiết cho từng người dùng hay nhóm người dùng. Ngoài ra, MySQL cũng hỗ trợ các tính năng lập chỉ mục (indexing), giúp tăng tốc độ truy vấn đáng kể, đặc biệt là với các bảng dữ liệu lớn.

Một yếu tố quan trọng giúp MySQL giữ vững vị trí hàng đầu trong cộng đồng phát triển phần mềm chính là tính tương thích cao với nhiều hệ điều hành và ngôn ngữ lập trình. Cụ thể, MySQL hoạt động ổn định trên các nền tảng như Windows, Linux, macOS và có thể tích hợp dễ dàng với các ngôn ngữ phổ biến như PHP, Python, Java, C# hay Ruby. MySQL cũng là lựa chọn mặc định hoặc được khuyến nghị trong nhiều framework phát triển ứng dụng như Laravel, Django hay Spring Boot, nhờ vào khả năng tích hợp thuận tiện và tài liệu hỗ trợ phong phú.

Không chỉ dừng lại ở góc độ kỹ thuật, MySQL còn được đánh giá cao về tính cộng đồng và hệ sinh thái hỗ trợ. Do là phần mềm mã nguồn mở, MySQL có một cộng đồng người dùng và nhà phát triển rộng lớn, góp phần liên tục cải tiến, cập nhật và hỗ trợ người dùng khi gặp khó khăn trong quá trình triển khai. Bên cạnh phiên bản miễn phí, MySQL cũng cung cấp các phiên bản thương mại với nhiều tính năng nâng cao, đáp ứng nhu cầu của các doanh nghiệp quy mô lớn.

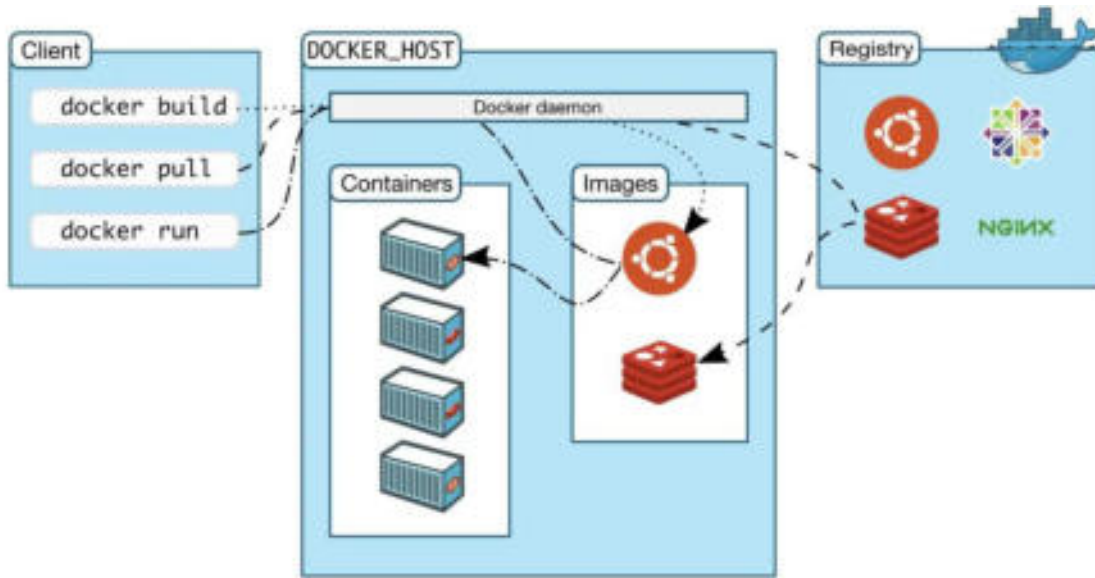
MySQL là một giải pháp quản trị cơ sở dữ liệu mạnh mẽ, linh hoạt và đáng tin cậy. Nhờ vào hiệu suất cao, khả năng mở rộng tốt, cùng với sự hỗ trợ từ cộng đồng phát triển rộng rãi, MySQL đã và đang giữ vai trò quan trọng trong việc xây dựng các hệ thống thông tin hiện đại. Việc lựa chọn và ứng dụng MySQL trong phát triển phần mềm là một quyết định hợp lý, đặc biệt đối với những hệ thống yêu cầu tính ổn định, khả năng xử lý dữ liệu nhanh chóng, và chi phí triển khai hợp lý.

1.9. Docker

Trong bối cảnh phát triển phần mềm hiện đại, đặc biệt là khi các ứng dụng ngày càng trở nên phức tạp và yêu cầu triển khai trên nhiều môi trường khác nhau, vấn đề về sự nhất quán giữa môi trường phát triển, kiểm thử và sản xuất trở thành một thách thức lớn đối với các nhóm kỹ sư phần mềm. Truyền thống trước đây thường dựa vào việc cấu hình hệ thống theo cách thủ công hoặc sử dụng máy ảo (virtual machines), tuy nhiên cách tiếp cận này thường tiêu tốn tài nguyên, khó quản lý và thiếu tính linh hoạt. Sự ra đời của Docker đã mang đến một bước tiến quan trọng trong lĩnh vực triển khai và phân phối phần mềm nhờ khả năng đóng gói ứng dụng cùng toàn bộ môi trường chạy vào các container nhẹ, linh hoạt và dễ dàng tái sử dụng.

Docker là một nền tảng mã nguồn mở, cung cấp khả năng tự động hóa việc triển khai ứng dụng trong các container. Khác với máy ảo truyền thống, nơi mỗi phiên bản cần một hệ điều hành riêng biệt, container chia sẻ cùng một kernel của hệ điều hành chủ (host OS) nhưng vẫn hoạt động như những môi trường độc lập. Điều này giúp giảm thiểu tối đa dung lượng tài nguyên sử dụng và tăng tốc độ khởi chạy ứng dụng.

Về mặt kiến trúc, Docker bao gồm ba thành phần chính: Docker Engine, Docker Images và Docker Containers. Docker Engine là công cụ lõi cung cấp môi trường chạy container và quản lý vòng đời của chúng. Docker Image là một bản mẫu chỉ đọc (read-only template) chứa tất cả những gì cần thiết để chạy một ứng dụng, bao gồm mã nguồn, thư viện, phụ thuộc và cấu hình hệ thống. Từ Image này, Docker có thể tạo ra một hoặc nhiều container – là các thực thể đang chạy của một image, hoạt động như một đơn vị ứng dụng độc lập.



Hình- 1-10: Kiến trúc docker

Việc tạo Docker Image thường được định nghĩa thông qua một tệp Dockerfile, trong đó mô tả từng bước cấu hình môi trường và cài đặt phần mềm cần thiết. Nhờ đó, toàn bộ quá trình triển khai trở nên minh bạch, có thể tái sử dụng, và dễ kiểm soát. Ngoài ra, Docker còn hỗ trợ Docker Compose, một công cụ cho phép định nghĩa và quản lý nhiều container cùng lúc thông qua một tệp YAML duy nhất. Điều này đặc biệt hữu ích trong các hệ thống vi dịch vụ (microservices), nơi mỗi chức năng của ứng dụng có thể được đóng gói và vận hành trong một container riêng biệt.

Một lợi ích lớn khác của Docker là khả năng di động hóa (portability), khi container được thiết kế để có thể chạy nhất quán trên bất kỳ hệ thống nào hỗ trợ Docker, từ máy tính cá nhân của lập trình viên cho đến máy chủ trên cloud như AWS, Google Cloud hay Azure. Nhờ đó, Docker giúp giảm thiểu rủi ro "chạy được ở máy tôi nhưng không chạy ở máy khác" – một vấn đề phổ biến trong phát triển phần mềm truyền thống.

Ngoài ra, Docker còn tích hợp chặt chẽ với các hệ thống quản lý container lớn hơn như Kubernetes, giúp mở rộng khả năng tự động hóa, cân bằng tải và phục hồi lỗi trong môi trường phân tán. Cộng đồng phát triển của Docker cũng rất năng động với hệ sinh thái phong phú như Docker Hub – nơi chia sẻ và lưu trữ các Docker Image công khai và riêng tư, hỗ trợ việc tái sử dụng và chuẩn hóa quy trình phát triển.

Tính đến hiện tại, Docker đã trở thành một công cụ không thể thiếu trong quy trình phát triển phần mềm hiện đại, đặc biệt là trong các mô hình phát triển như CI/CD Việc tích

hợp Docker vào pipeline phát triển giúp đẩy nhanh chu kỳ phát hành phần mềm, giảm thiểu lỗi do khác biệt môi trường và tăng cường khả năng kiểm thử tự động.

Tóm lại, Docker không chỉ là một công nghệ triển khai phần mềm, mà còn là một triết lý phát triển hướng đến tính đóng gói, nhất quán, linh hoạt và hiệu quả. Nhờ những lợi thế vượt trội về hiệu suất, khả năng mở rộng và tính di động, Docker đã và đang trở thành một phần không thể thiếu trong hạ tầng kỹ thuật của nhiều tổ chức, từ các startup nhỏ cho đến các tập đoàn công nghệ toàn cầu [17].

1.10. Azure

1.10.1. Tổng quan

Trong thời đại số hóa mạnh mẽ hiện nay, việc xây dựng và triển khai các hệ thống ứng dụng trên nền tảng điện toán đám mây (cloud computing) đã trở thành xu hướng tất yếu, giúp doanh nghiệp tiết kiệm chi phí hạ tầng, dễ dàng mở rộng quy mô và tăng cường khả năng quản trị hệ thống. Trong số các nhà cung cấp dịch vụ đám mây hàng đầu thế giới, Microsoft Azure nổi bật với hệ sinh thái đa dạng, độ tin cậy cao và khả năng tích hợp chặt chẽ với các công cụ phát triển phần mềm hiện đại.

Microsoft Azure là một nền tảng điện toán đám mây toàn diện được phát triển bởi tập đoàn Microsoft, cung cấp hơn 200 dịch vụ khác nhau, bao gồm hạ tầng như dịch vụ (IaaS), nền tảng như dịch vụ PaaS và phần mềm như dịch vụ SaaS.

1.10.2. Azure Virtual Machines (Azure VMs)

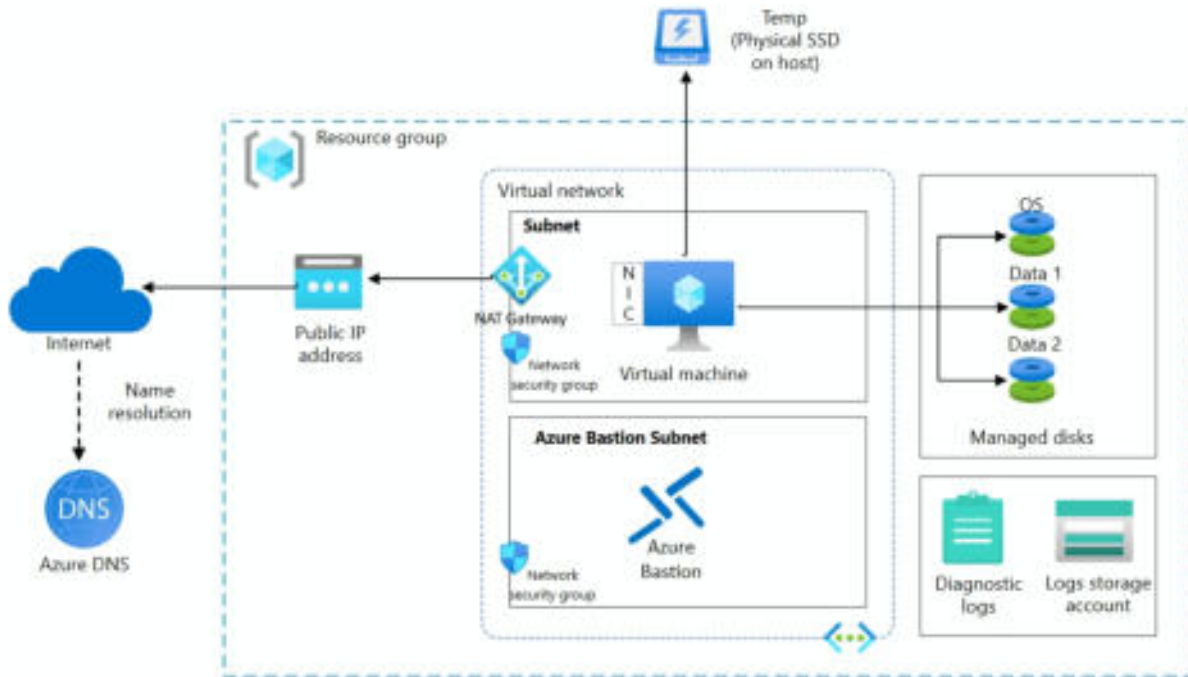
Azure Virtual Machines là một trong những dịch vụ IaaS quan trọng nhất của nền tảng Azure, cho phép người dùng tạo và quản lý các máy chủ ảo chạy trên hạ tầng của Microsoft. Dịch vụ này hỗ trợ nhiều hệ điều hành khác nhau như Windows Server, Ubuntu, Debian, CentOS... Người dùng có thể toàn quyền cấu hình máy ảo như lựa chọn vùng địa lý, kích thước máy (CPU, RAM), dung lượng lưu trữ, mạng, cũng như quyền truy cập hệ thống.

Máy ảo Azure được triển khai trên hạ tầng phân tán và dư thừa (redundant infrastructure), bảo đảm độ sẵn sàng và khả năng chịu lỗi cao. Ngoài ra, Azure cung cấp các công cụ quản trị linh hoạt như Azure Portal, Azure CLI và Azure Resource Manager giúp tự động hóa quá trình tạo, cấu hình và quản lý tài nguyên.

Đặc biệt, Azure Virtual Machines hỗ trợ cấp phát địa chỉ IP công cộng (public IP address) và tên miền DNS tùy chỉnh (DNS name) cho từng máy ảo, tạo điều kiện thuận lợi cho việc truy cập từ xa, triển khai ứng dụng trên Internet hoặc thiết lập các máy chủ dịch vụ. Người dùng có thể lựa chọn địa chỉ IP tĩnh hoặc động, tùy theo yêu cầu triển khai. Tên miền DNS có thể được gán thông qua dịch vụ Azure DNS, cho phép truy cập máy ảo thông

qua các tên miền có cấu trúc dễ nhớ. Điều này giúp đơn giản hóa quá trình kết nối, cấu hình máy chủ và triển khai ứng dụng trong các hệ thống phân tán.

Một điểm mạnh của Azure VM là khả năng tích hợp với các dịch vụ bảo mật, giám sát và sao lưu dữ liệu. Cụ thể, người dùng có thể sử dụng Azure Monitor để theo dõi hiệu suất hệ thống, Azure Security Center để phát hiện và xử lý rủi ro bảo mật, và Azure Backup để đảm bảo an toàn dữ liệu. Việc triển khai và mở rộng quy mô hệ thống cũng trở nên dễ dàng hơn nhờ tính năng autoscaling và load balancing được tích hợp sẵn [18].



Hình- 1-11: Kiến trúc Azure virtual machine

1.10.3. Kết hợp Azure và Docker trong triển khai ứng dụng

Docker và Azure là hai công nghệ bổ trợ lẫn nhau, khi Docker đảm nhận vai trò đóng gói và cô lập ứng dụng, còn Azure cung cấp môi trường hạ tầng linh hoạt để chạy và quản lý các container đó. Có nhiều cách để triển khai Docker trên nền tảng Azure, tùy thuộc vào nhu cầu và độ phức tạp của hệ thống.

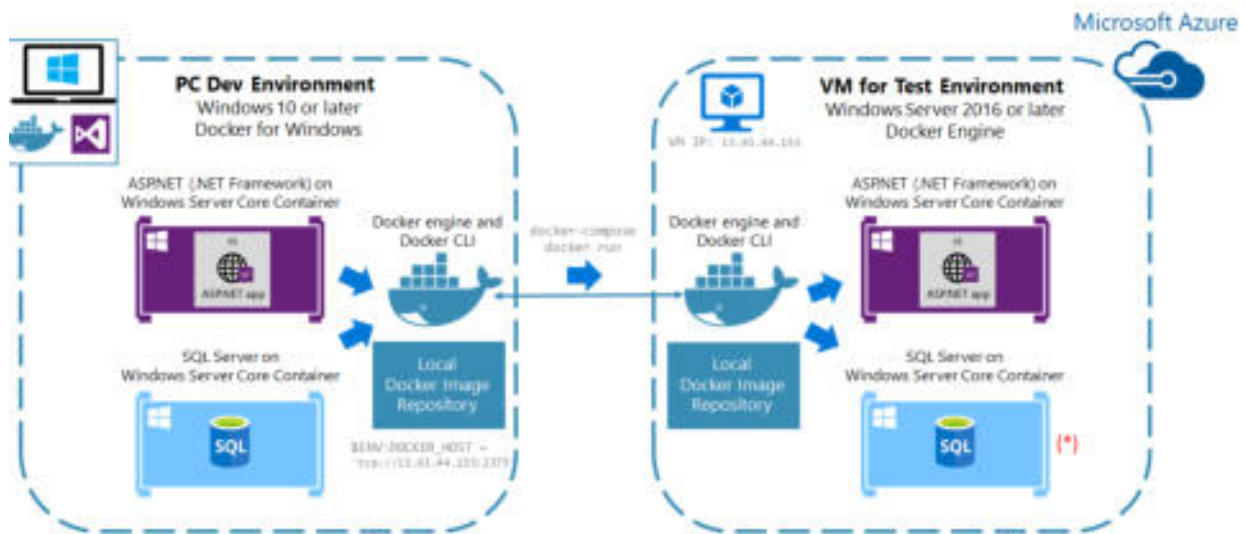
Triển khai Docker trên Azure Virtual Machines là phương pháp đơn giản và trực quan. Trong mô hình này, người dùng cài đặt Docker Engine trực tiếp lên máy ảo Azure, sau đó thực hiện các thao tác build image, khởi tạo container, hoặc chạy các dịch vụ như Docker Compose tương tự như trên môi trường local. Phương pháp này mang lại sự linh hoạt và

kiểm soát cao, phù hợp với các nhóm phát triển muốn tùy biến môi trường hoặc đang trong giai đoạn thử nghiệm, phát triển ứng dụng.

Đối với các hệ thống lớn hơn, Azure còn cung cấp Azure Container Instances (ACI) – một dịch vụ cho phép chạy container đơn lẻ mà không cần quản lý hạ tầng máy chủ, và Azure Kubernetes Service (AKS) – giải pháp mạnh mẽ để triển khai và điều phối các cụm container ở quy mô lớn, hỗ trợ tự động mở rộng, cân bằng tải và cập nhật liên tục (rolling update).

Đặc biệt, sự tích hợp chặt chẽ giữa Azure và Docker Hub hoặc Azure Container Registry (ACR) giúp tăng tính tự động hóa trong chuỗi triển khai ứng dụng. Các image Docker có thể được đẩy lên kho lưu trữ (registry) và được tự động kéo về để triển khai trên các máy ảo hoặc dịch vụ container của Azure.

Ngoài ra, Azure còn hỗ trợ triển khai ứng dụng container hóa thông qua Azure App Service for Containers, cho phép người dùng chạy ứng dụng web từ image Docker chỉ với vài bước cấu hình đơn giản, tiết kiệm thời gian và phù hợp với các hệ thống web quy mô vừa và nhỏ.



Hình- 1-12: Triển khai ứng dụng kết hợp Docker và Azure virtual machine

1.10.4. Lợi ích của việc kết hợp Azure và Docker

Sự kết hợp giữa Docker và Azure mang lại nhiều lợi ích nổi bật cho quá trình phát triển và triển khai phần mềm, bao gồm:

- Tính nhất quán: Docker đảm bảo môi trường chạy ứng dụng luôn đồng nhất giữa các giai đoạn phát triển, kiểm thử và triển khai thực tế.

- Tính mở rộng và sẵn sàng cao: Azure cung cấp hạ tầng mạnh mẽ và các dịch vụ hỗ trợ mở rộng tự động, giúp hệ thống dễ dàng đáp ứng nhu cầu tăng trưởng.
- Tự động hóa và DevOps: Việc tích hợp Docker vào các pipeline CI/CD trên Azure DevOps hoặc GitHub Actions cho phép tự động build, test và deploy ứng dụng hiệu quả.
- Bảo mật và giám sát toàn diện: Azure cung cấp các công cụ bảo mật và giám sát container, đảm bảo an toàn trong môi trường vận hành.

1.11. Kiểm thử

Kiểm thử phần mềm là một giai đoạn quan trọng trong quá trình phát triển phần mềm nhằm đảm bảo hệ thống hoạt động đúng theo yêu cầu, ổn định, và không có lỗi nghiêm trọng trước khi triển khai chính thức [19]. Việc kiểm thử giúp phát hiện sớm các lỗi, tăng tính tin cậy cho sản phẩm và giảm thiểu rủi ro trong vận hành. Trong các dự án hiện đại, kiểm thử thường được tích hợp vào quy trình phát triển liên tục (CI/CD) và được thực hiện tự động nhằm nâng cao hiệu quả [20].

Trong hệ thống này, hai công cụ kiểm thử chính được sử dụng là RSpec và Selenium.

1.11.1. RSpec

RSpec là một trong những thư viện kiểm thử phổ biến nhất dành cho ngôn ngữ lập trình Ruby. RSpec hỗ trợ các nhà phát triển xây dựng và thực hiện các kiểm thử tự động, đảm bảo rằng các thành phần của ứng dụng hoạt động chính xác theo yêu cầu.

Cấu trúc của RSpec được thiết kế dựa trên kỹ thuật Behavior-Driven Development (BDD), giúp nhà phát triển tập trung vào việc mô tả hành vi mong muốn của phần mềm trước khi viết mã.

Lợi ích của RSpec Unit Test

- Đảm bảo tính đúng đắn: Xác minh rằng từng đơn vị mã nguồn hoạt động chính xác.
- Hạn chế lỗi: Phát hiện lỗi trong giai đoạn phát triển, giảm thiểu rủi ro trong sản phẩm.
- Tăng cường khả năng bảo trì: Dễ dàng thay đổi hoặc mở rộng mã nguồn mà không lo ảnh hưởng đến các thành phần khác.
- Hỗ trợ tài liệu hóa: Mô tả rõ ràng các hành vi của phần mềm, giúp các nhà phát triển dễ dàng hiểu được yêu cầu.

Tóm lại, RSpec Unit Test là một công cụ mạnh mẽ giúp đảm bảo chất lượng của ứng dụng Ruby. Việc sử dụng RSpec không chỉ tăng hiệu quả phát triển mà còn giúp phát hiện và sửa lỗi sớm, tiết kiệm thời gian và chi phí trong quá trình phát triển phần mềm [21].

1.11.2. Selenium

SELENIUM(SE) là một phần mềm mã nguồn mở, được phát triển bởi Jason Huggins vào năm 2004 và được phát triển bởi Thoughtwork. Ông phát triển thư viện Javascript để tự động chạy các test trên nhiều trình duyệt, chính thư viện này đã trở thành Selenium Core tạo cơ sở cho Selenium IDE và Selenium Remote Control(RC). Selenium RC là một đột phá vì không sản phẩm nào cho phép bạn điều khiển trình duyệt với bất kì ngôn ngữ mình chọn [22].

Selenium gồm nhiều công cụ với cách tiếp cận khác nhau để hỗ trợ kiểm thử tự động. Bộ công cụ này rất linh hoạt cho phép lựa chọn các yếu tố giao diện hay so sánh kết quả thí nghiệm dự kiến với thực tiễn. Một trong những tính năng chính của Selenium là hỗ trợ kiểm thử trên nhiều trình duyệt. Selenium là một công cụ hỗ trợ kiểm tra tự động cho các ứng dụng chạy trên nền web. Selenium hỗ trợ kiểm tra hầu hết trên các trình duyệt phổ biến hiện nay như Firefox , Internet Explorer , Safari ,... cũng như các hệ điều hành chủ yếu như Windows , Linux , Mac ,... Selenium hỗ trợ một số lớn các ngôn ngữ lập trình như C# , Java , Perl , PHP , Python, Ruby,..

1.12. Kết luận chương 1

Tóm lại, chương cơ sở lý thuyết đã trình bày những khái niệm và công nghệ nền tảng được sử dụng trong quá trình phát triển hệ thống, bao gồm: MobileNet, kiến trúc client- server, NextJS, Cloudinary, TailwindCSS, Ruby on Rails, Flask, PostgreSQL, Rspec, Selenium. Những kiến thức này không chỉ giúp xây dựng nền tảng lý thuyết vững chắc mà còn định hướng cho việc phân tích và thiết kế hệ thống trong các chương sau.

CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

2.1. Yêu cầu chức năng

2.1.1. Người dùng

Xác thực người dùng:

- Đăng nhập bằng email và mật khẩu hoặc tài khoản google
- Đăng ký tài khoản
- Kích hoạt tài khoản qua email sau khi đăng ký
- Quên mật khẩu: Gửi link đặt lại mật khẩu qua email - chỉ áp dụng cho tài khoản đã kích hoạt

Quản lý khóa học:

- Tìm kiếm khóa học theo tên, lọc theo cấp độ (ví dụ: N5–N1), danh mục
- Xem khóa học theo từng danh mục
- Xem chi tiết nội dung khóa học, thông tin giảng viên, mức phí,...
- Thanh toán để đăng ký và tham gia học
- Xem danh sách khóa học đã đăng ký
- Xem đánh giá các khóa học và đánh giá sau khi đăng kí tham gia khóa học thành công
- Báo cáo khóa học khi phát hiện nội dung vi phạm
- Tham gia diễn đàn thảo luận của từng khóa học

Học tập và tài nguyên:

- Truy cập các bài học trong khóa học do giáo viên tạo, bao gồm: video, bài học Kanji, flashcard từ vựng, bài tập luyện tập
- Theo dõi tiến độ học tập (bài đã hoàn thành, thời gian học, trạng thái,...)

Công cụ hỗ trợ học tập:

- Tìm kiếm từ vựng Kanji
- Dịch đoạn văn tiếng Nhật
- Kiểm tra lỗi ngữ pháp tiếng Nhật

Thông tin cá nhân và tương tác:

- Xem và chỉnh sửa hồ sơ cá nhân
- Xem thông tin giảng viên và các khóa học mà giáo viên đó giảng dạy
- Theo dõi giáo viên để nhận thông báo khi họ mở khóa học mới

2.1.2. Giáo viên

Tài khoản:

- Đăng ký và đăng nhập
- Đổi mật khẩu
- Xem và cập nhật thông tin cá nhân

Quản lý nội dung:

- Tạo, chỉnh sửa, xóa khóa học
- Tạo và quản lý các bài học (bao gồm video, Kanji, flashcard, bài tập)
- Quản lý yêu cầu tham gia khóa học từ học viên

Thống kê:

- Xem thống kê số lượng học viên tham gia
- Theo dõi doanh thu khóa học
- Xem báo cáo tổng quan các khóa học đang giảng dạy

2.1.3. Quản trị viên

Xác thực:

- Đăng nhập vào hệ thống quản trị

Quản lý hệ thống:

- Quản lý danh sách giáo viên
- Quản lý học viên
- Quản lý và phân loại danh mục khóa học
- Quản lý điều khoản hoạt động và chính sách nền tảng

Quản lý nội dung:

- Quản lý toàn bộ các khóa học trên hệ thống
- Kiểm duyệt và xử lý các báo cáo vi phạm từ người dùng

Thống kê:

- Thống kê số lượng người dùng, giáo viên, khóa học
- Thống kê doanh thu toàn hệ thống theo thời gian

2.2. Yêu cầu phi chức năng

2.2.1. Hiệu năng

Hệ thống cần bảo đảm thời gian phản hồi nhanh, đặc biệt trong các thao tác tương tác trực tiếp với người học như làm bài kiểm tra, luyện viết Kanji, hoặc tra từ điển. Ngoài ra, hệ thống cần có khả năng xử lý đồng thời nhiều người dùng truy cập cùng lúc mà không gây ra tình trạng nghẽn mạng hoặc lỗi hệ thống.

2.2.2. Khả năng mở rộng

Hệ thống được thiết kế với kiến trúc có khả năng mở rộng linh hoạt, cho phép dễ dàng bổ sung tính năng mới, mở rộng cơ sở dữ liệu người dùng hoặc tích hợp thêm các dịch vụ trí tuệ nhân tạo như phân tích lỗi ngữ pháp, nhận diện chữ viết tay Kanji. Việc triển khai theo mô hình microservices và kết hợp Docker với nền tảng điện toán đám mây như Azure sẽ giúp hệ thống dễ dàng thích ứng khi quy mô người dùng tăng lên.

2.2.3. Tính sẵn sàng và độ tin cậy

Hệ thống cần đạt độ sẵn sàng cao, bảo đảm người học có thể truy cập mọi lúc, đặc biệt vào các khung giờ cao điểm.

2.2.4. Bảo mật

Do hệ thống có lưu trữ thông tin cá nhân người dùng (họ tên, email, mật khẩu) và dữ liệu học tập, việc đảm bảo an toàn thông tin là yêu cầu bắt buộc. Các kỹ thuật mã hóa như HTTPS, lưu mật khẩu dưới dạng băm (hash), cơ chế xác thực bảo mật (JWT, OAuth2) và phân quyền người dùng (admin, giáo viên, học viên) cần được triển khai. Ngoài ra, hệ thống phải có khả năng chống lại các cuộc tấn công phổ biến như SQL Injection, XSS hay CSRF.

2.2.5. Tính khả dụng trên nhiều nền tảng

Website cần hoạt động tốt trên nhiều thiết bị khác nhau, đồng thời tương thích với các trình duyệt phổ biến như Chrome, Firefox, Safari và Edge. Giao diện cần được thiết kế theo hướng responsive, đảm bảo hiển thị tốt trên nhiều kích thước màn hình.

2.2.6. Dễ sử dụng và thân thiện với nhiều người dùng

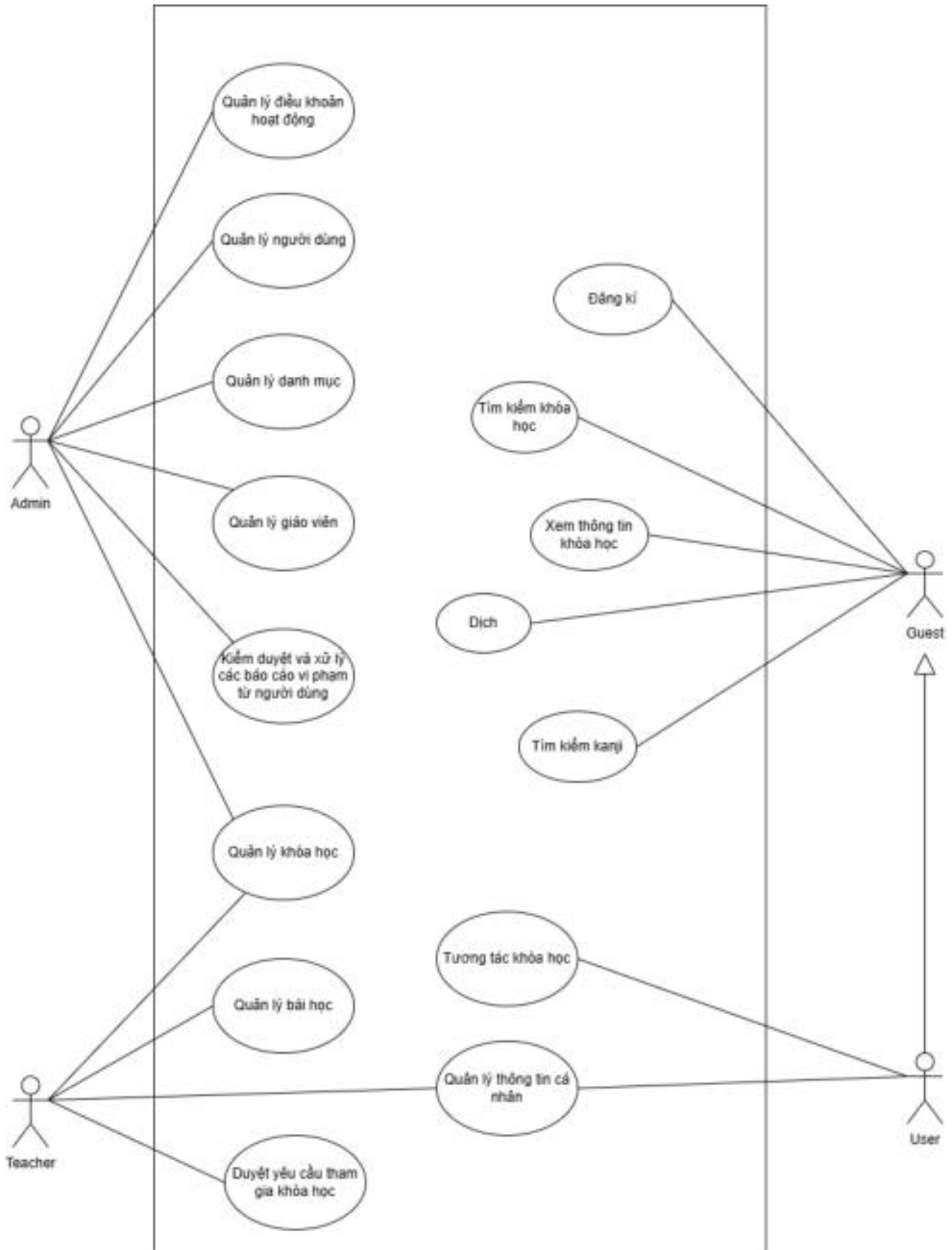
Hệ thống cần có giao diện trực quan, dễ sử dụng, phù hợp với đối tượng học sinh – sinh viên. Các chức năng như đăng ký tài khoản, đăng nhập, truy cập khóa học, làm bài kiểm tra, xem tiến độ học tập phải được bố trí rõ ràng, dễ thao tác. Ngoài ra, cần cung cấp tài liệu hướng dẫn hoặc phân trợ giúp để hỗ trợ người dùng mới làm quen.

2.2.7. Khả năng bảo trì và phát triển

Mã nguồn cần được tổ chức rõ ràng, dễ hiểu, áp dụng các nguyên tắc lập trình hướng đối tượng. Điều này giúp dễ dàng bảo trì, sửa lỗi và nâng cấp hệ thống trong tương lai. Việc sử dụng các công nghệ phổ biến và có cộng đồng hỗ trợ rộng rãi như Next.js, Ruby on Rails, MySQL cũng góp phần tăng khả năng duy trì lâu dài.

2.3. Biểu đồ ca sử dụng

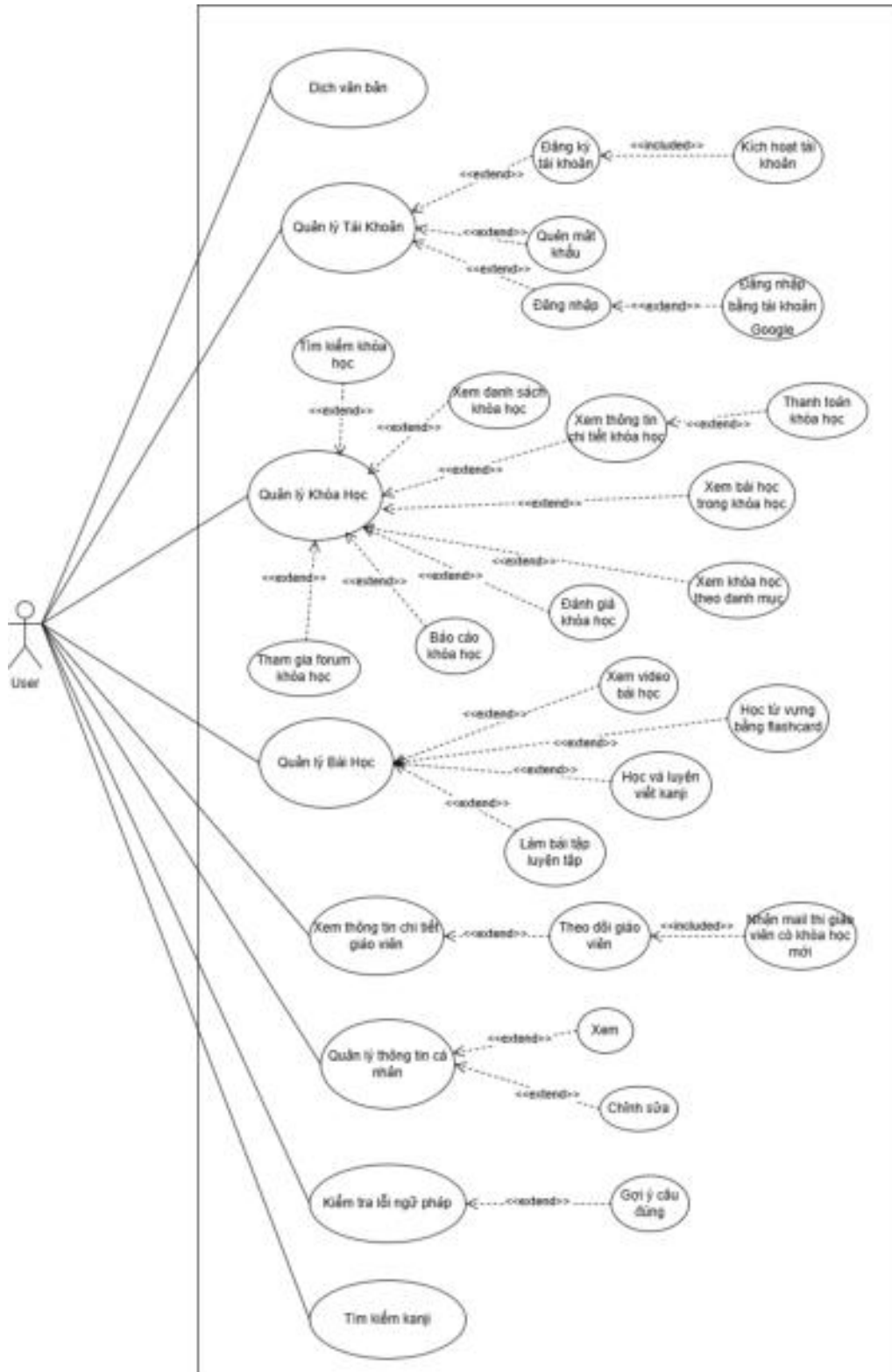
2.3.1. Biểu đồ usecase tổng quan



Hình- 2-1: Biểu đồ usecase tổng quan

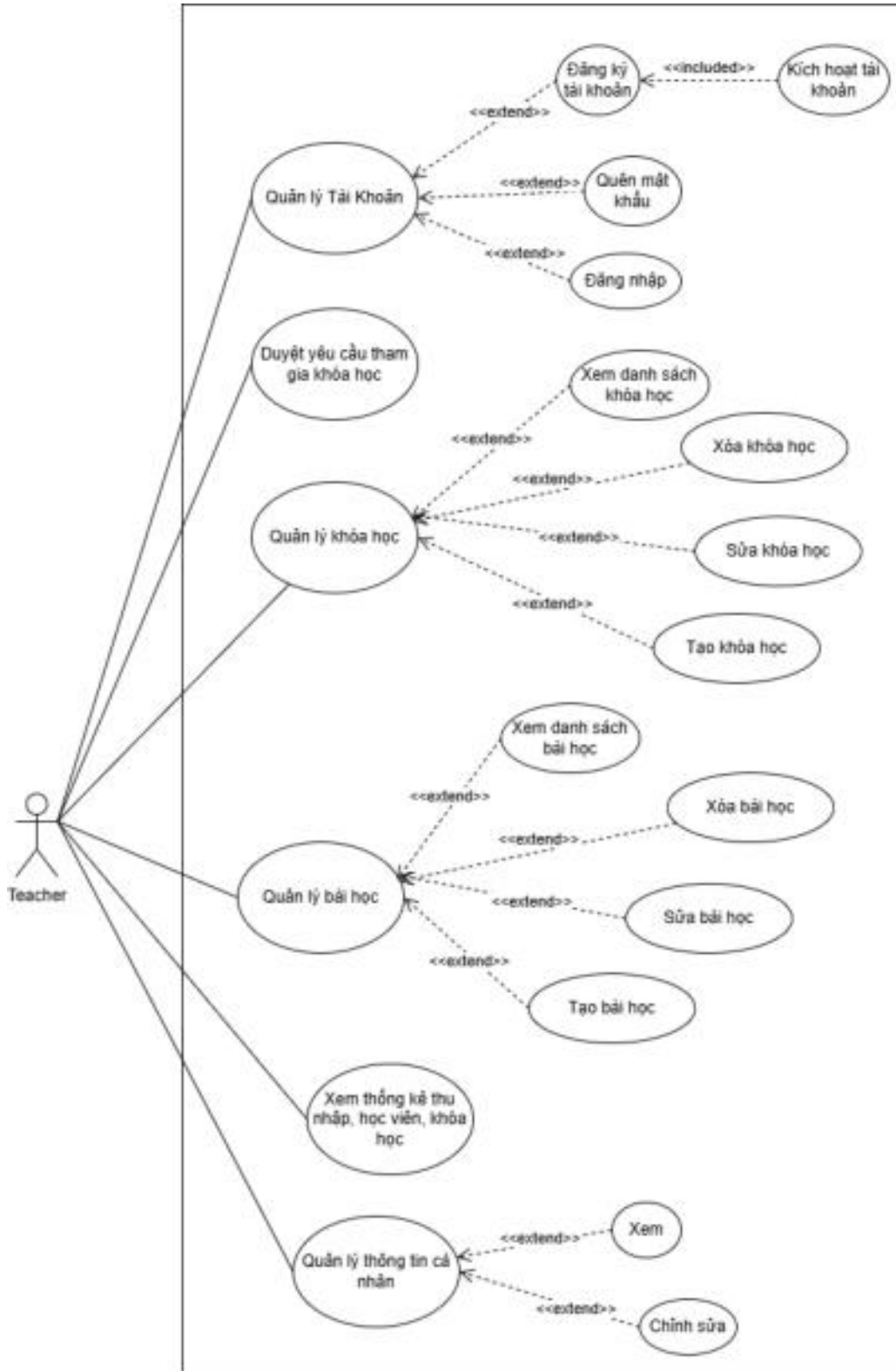
2.3.2. Biểu đồ usecase phân rã

2.3.2.1. Biểu đồ usecase user



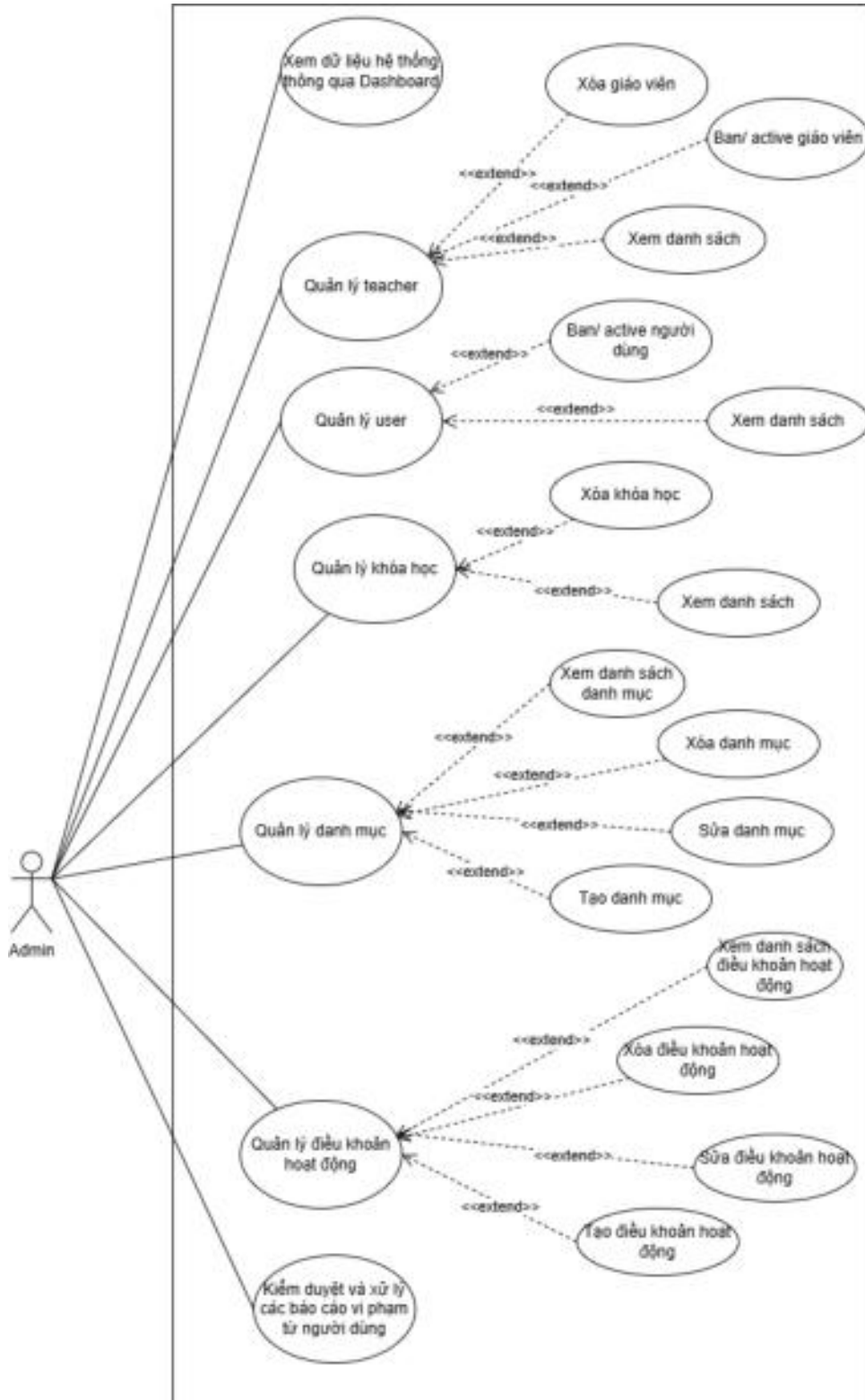
Hình- 2-2: Biểu đồ usecase user

2.3.2.2. Biểu đồ usecase teacher



Hình- 2-3: Biểu đồ usecase teacher

2.3.2.3. Biểu đồ usecase admin



Hình- 2-4: Biểu đồ usecase admin

2.3.3. Đặc tả biểu đồ usecase

Bảng- 2-1: Đặc tả Use Case: Đăng ký tài khoản

Mã ca sử dụng	UC-01
Tên ca sử dụng	Đăng ký tài khoản
Mô tả	Use case này cho phép người dùng mới tạo một tài khoản để sử dụng các tính năng của hệ thống.
Tác nhân	User (Người dùng chưa có tài khoản)
Điều kiện trước	Người dùng chưa có tài khoản trên hệ thống.
Điều kiện sau	- Tài khoản của người dùng được tạo và lưu trong hệ thống nhưng ở trạng thái chưa kích hoạt. - Hệ thống gửi email kích hoạt đến người dùng.
Các bước thực hiện	<ol style="list-style-type: none"> 1. Người dùng chọn chức năng "Đăng ký". 2. Hệ thống hiển thị form đăng ký, yêu cầu nhập. 3. Người dùng nhập đầy đủ thông tin và nhấn "Đăng ký". 4. Hệ thống kiểm tra tính hợp lệ của thông tin (email chưa tồn tại, mật khẩu đủ mạnh). 5. Hệ thống tạo tài khoản mới và lưu thông tin. 6. Hệ thống gửi email kèm liên kết kích hoạt đến địa chỉ email người dùng đã đăng ký. 7. Hệ thống thông báo đăng ký thành công và yêu cầu người dùng kiểm tra email để kích hoạt. (Bao gồm use case <i>Kích hoạt tài khoản</i>).

Bảng- 2-2: Đặc tả Use Case: Đăng nhập

Mã ca sử dụng	UC-02
Tên ca sử dụng	Đăng nhập

Mô tả	Use case này cho phép người dùng đã có tài khoản truy cập vào hệ thống.
Tác nhân	User
Điều kiện trước	Người dùng đã có một tài khoản đã được kích hoạt trên hệ thống.
Điều kiện sau	Người dùng đăng nhập thành công và được chuyển hướng đến trang chính của hệ thống.
Các bước thực hiện	<ol style="list-style-type: none"> 1. Người dùng chọn chức năng "Đăng nhập". 2. Hệ thống hiển thị form đăng nhập, yêu cầu nhập email và mật khẩu. 3. Người dùng nhập thông tin. 4. Hệ thống kiểm tra thông tin đăng nhập. 5. Nếu thông tin chính xác, hệ thống cho phép truy cập và tạo một phiên làm việc mới. <p>Luồng mở rộng:</p> <ol style="list-style-type: none"> 4a. Nếu người dùng chọn "Đăng nhập bằng tài khoản Google", hệ thống chuyển hướng sang trang xác thực của Google. Sau khi xác thực thành công, người dùng được đăng nhập vào hệ thống. 4b. Nếu thông tin không chính xác, hệ thống hiển thị thông báo lỗi.

Bảng- 2-3: Đặc tả Use Case: Quên mật khẩu

Mã ca sử dụng	UC-03
Tên ca sử dụng	Quên mật khẩu
Mô tả	Cho phép người dùng đã có tài khoản đặt lại mật khẩu của mình nếu họ quên.

Tác nhân	User
Điều kiện trước	Người dùng đã có tài khoản trên hệ thống.
Điều kiện sau	Người dùng nhận được email hướng dẫn để tạo mật khẩu mới.
Các bước thực hiện	<ol style="list-style-type: none"> 1. Tại trang đăng nhập, người dùng nhấn vào liên kết "Quên mật khẩu". 2. Hệ thống yêu cầu người dùng nhập địa chỉ email đã đăng ký. 3. Người dùng nhập email và nhấn "Gửi". 4. Hệ thống kiểm tra email có tồn tại trong cơ sở dữ liệu hay không. 5. Nếu có, hệ thống gửi một email chứa liên kết để đặt lại mật khẩu. 6. Hệ thống thông báo đã gửi email thành công.

Bảng- 2-4: Đặc tả Use Case: Tìm kiếm khóa học

Mã ca sử dụng	UC-04
Tên ca sử dụng	Tìm kiếm khóa học
Mô tả	Cho phép người dùng tìm kiếm các khóa học dựa trên từ khóa.
Tác nhân	User
Điều kiện trước	Không có.
Điều kiện sau	Hệ thống hiển thị danh sách các khóa học phù hợp với từ khóa tìm kiếm.
Các bước thực hiện	<ol style="list-style-type: none"> 1. Người dùng nhập từ khóa vào thanh tìm kiếm. 2. Người dùng nhấn nút tìm kiếm hoặc phím Enter. 3. Hệ thống truy vấn và trả về danh sách các khóa học có tên hoặc mô tả chứa từ khóa.

	4. Hệ thống hiển thị kết quả cho người dùng.
--	--

Bảng- 2-5: Đặc tả Use Case: Xem danh sách khóa học theo danh mục

Mã ca sử dụng	UC-05
Tên ca sử dụng	Xem danh sách khóa học theo từng danh mục
Mô tả	Cho phép người dùng xem tất cả các khóa học hiện có trên hệ thống theo từng danh mục.
Tác nhân	User
Điều kiện trước	Không có.
Điều kiện sau	Người dùng xem được danh sách các khóa học được trình bày một cách trực quan.
Các bước thực hiện	<ol style="list-style-type: none"> 1. Người dùng chọn vào danh mục mình muốn tìm kiếm. 2. Hệ thống hiển thị danh sách tất cả các khóa học. <p>Luồng mở rộng:</p> <ol style="list-style-type: none"> 2a. Người dùng có thể chọn một cấp độ cụ thể (ví dụ: N5, N4...), giá, ... 2b. Hệ thống lọc và chỉ hiển thị các khóa học đáp ứng điều kiện.

Bảng- 2-6: Đặc tả Use Case: Xem thông tin chi tiết khóa học

Mã ca sử dụng	UC-06
Tên ca sử dụng	Xem thông tin chi tiết khóa học
Mô tả	Cho phép người dùng xem thông tin chi tiết về một khóa học cụ thể.
Tác nhân	User

Điều kiện trước	Người dùng đang ở trang danh sách khóa học hoặc trang kết quả tìm kiếm.
Điều kiện sau	Người dùng xem được đầy đủ thông tin về khóa học để đưa ra quyết định đăng ký.
Các bước thực hiện	<p>1. Từ danh sách khóa học, người dùng nhấp vào một khóa học cụ thể.</p> <p>2. Hệ thống hiển thị trang chi tiết của khóa học bao gồm: tên, mô tả, mục tiêu, nội dung các bài học, thông tin giáo viên, học phí, và các đánh giá từ học viên khác.</p> <p>Luồng mở rộng:</p> <p>2a. Nếu người dùng chưa đăng nhập và muốn mua khóa học, hệ thống yêu cầu đăng nhập/đăng ký.</p> <p>2b. Người dùng có thể thực hiện thanh toán khóa học từ trang này.</p>

Bảng- 2-7: Đặc tả Use Case: Học bài học

Mã ca sử dụng	UC-07
Tên ca sử dụng	Học bài học
Mô tả	Cho phép người dùng đã đăng ký khóa học xem nội dung video của các bài học và học theo các phương pháp hỗ trợ như luyện viết kanji, flashcard, làm bài luyện tập.
Tác nhân	User
Điều kiện trước	<ul style="list-style-type: none"> - Người dùng đã đăng nhập. - Người dùng đã đăng ký (mua) khóa học tương ứng.
Điều kiện sau	Người dùng tiếp thu được kiến thức thông qua video.

Các bước thực hiện	<ol style="list-style-type: none"> 1. Người dùng truy cập vào khóa học đã đăng ký. 2. Người dùng chọn một bài học cụ thể. 3. Hệ thống hiển thị trình phát video của bài học đó. 4. Người dùng có thể thực hiện các thao tác: phát, tạm dừng, tua video. <p>Luồng mở rộng</p> <ol style="list-style-type: none"> 3a. Trong một bài học, người dùng chọn mục “flashcard” <ol style="list-style-type: none"> 1. Hệ thống hiển thị danh sách flashcard của bài học 2. Người dùng học từ vựng theo các thẻ ghi nhớ 3b. Trong một bài học, người dùng chọn mục “Luyện viết Kanji” <ol style="list-style-type: none"> 1. Hệ thống hiển thị danh sách kanji của bài học và ô luyện viết 2. Người dùng viết vào ô luyện viết, nhấn nút chấm điểm 3. Hệ thống hiển thị điểm 3c. Trong một bài học, người dùng chọn mục "Bài tập". <ol style="list-style-type: none"> 1. Hệ thống hiển thị danh sách các câu hỏi. 2. Người dùng trả lời các câu hỏi. 3. Sau khi hoàn tất, người dùng nhấn nút "Nộp bài". 4. Hệ thống chấm điểm, hiển thị kết quả và đáp án chi tiết.
--------------------	--

Bảng- 2-8: Đặc tả Use Case: Quản lý thông tin cá nhân

Mã ca sử dụng	UC-08
Tên ca sử dụng	Quản lý thông tin cá nhân
Mô tả	Cho phép người dùng xem và chỉnh sửa các thông tin cá nhân của mình.
Tác nhân	User
Điều kiện trước	Người dùng đã đăng nhập vào hệ thống.
Điều kiện sau	Thông tin cá nhân của người dùng được cập nhật trong hệ thống.

Các bước thực hiện	<ol style="list-style-type: none"> 1. Người dùng truy cập vào trang "Thông tin cá nhân". 2. Hệ thống hiển thị các thông tin hiện tại của người dùng (họ tên, email, ảnh đại diện...). (Use case <i>Xem</i>) 3. Người dùng thay đổi thông tin và nhấn "Lưu". 6. Hệ thống kiểm tra và cập nhật thông tin mới vào cơ sở dữ liệu. (Use case <i>Chỉnh sửa</i>)
--------------------	---

Bảng- 2-9: Đặc tả Use Case: Dịch văn bản

Mã ca sử dụng	UC-09
Tên ca sử dụng	Dịch văn bản
Mô tả	Cung cấp công cụ dịch một đoạn văn bản từ tiếng Nhật sang tiếng Việt và ngược lại.
Tác nhân	User
Điều kiện trước	Không có.
Điều kiện sau	Người dùng nhận được kết quả dịch của đoạn văn bản đã nhập.
Các bước thực hiện	<ol style="list-style-type: none"> 1. Người dùng truy cập vào trang "Dịch văn bản". 2. Người dùng nhập đoạn văn bản cần dịch vào ô nhập liệu. 3. Người dùng chọn ngôn ngữ nguồn và ngôn ngữ đích. 4. Người dùng nhấn nút "Dịch". 5. Hệ thống xử lý và hiển thị kết quả dịch.

Bảng- 2-10: Đặc tả Use Case: Tìm kiếm Kanji

Mã ca sử dụng	UC-10
Tên ca sử dụng	Tìm kiếm Kanji

Mô tả	Cho phép người dùng tra cứu thông tin chi tiết về một chữ Kanji.
Tác nhân	User
Điều kiện trước	Không có.
Điều kiện sau	Người dùng xem được thông tin chi tiết của chữ Kanji cần tra.
Các bước thực hiện	<ol style="list-style-type: none"> 1. Người dùng truy cập trang "Tìm kiếm Kanji". 2. Người dùng vẽ chữ kanji vào ô tìm kiếm. 3. Người dùng nhấn nút "Tìm". 4. Hệ thống hiển thị danh sách các chữ kanji phù hợp với chữ viết kanji người dùng đã nhập 5. Người dùng chọn chữ kanji muốn xem chi tiết 6. Hệ thống hiển thị thông tin chi tiết về chữ Kanji đó: âm On, âm Kun, nghĩa, số nét, bộ thủ, các từ vựng ví dụ.

Bảng- 2-11: Đặc tả Use Case: Tạo khóa học

Mã ca sử dụng	UC-11
Tên ca sử dụng	Tạo khóa học
Mô tả	Giáo viên tạo một khóa học mới trên hệ thống.
Tác nhân	Teacher (Giáo viên)
Điều kiện trước	- Giáo viên đã đăng nhập vào hệ thống với tài khoản của mình.
Điều kiện sau	<ul style="list-style-type: none"> - Một khóa học mới được tạo và lưu vào hệ thống, liên kết với tài khoản của giáo viên. - Khóa học mới được hiển thị trong danh sách khóa học của giáo viên.

Các bước thực hiện	<ol style="list-style-type: none"> 1. Giáo viên chọn chức năng "Quản lý khóa học" từ menu. 2. Giáo viên chọn "Tạo khóa học mới". 3. Hệ thống hiển thị form nhập thông tin khóa học. 4. Giáo viên điền đầy đủ thông tin và nhấn "Lưu". 5. Hệ thống kiểm tra tính hợp lệ của dữ liệu đầu vào. 6. Hệ thống lưu thông tin khóa học mới vào cơ sở dữ liệu, thông báo tạo khóa học thành công và chuyển hướng đến trang quản lý khóa học.
--------------------	---

Bảng- 2-12: Đặc tả Use Case: Sửa khóa học

Mã ca sử dụng	UC-12
Tên ca sử dụng	Sửa khóa học
Mô tả	Cho phép Giáo viên cập nhật lại thông tin của một khóa học đã tạo trước đó.
Tác nhân	Teacher (Giáo viên)
Điều kiện trước	<ul style="list-style-type: none"> - Giáo viên đã đăng nhập. - Khóa học cần sửa đã tồn tại và thuộc quyền quản lý của giáo viên đó.
Điều kiện sau	- Thông tin của khóa học được cập nhật thành công trong hệ thống.
Các bước thực hiện	<ol style="list-style-type: none"> 1. Giáo viên vào trang "Quản lý khóa học". 2. Hệ thống hiển thị danh sách các khóa học do giáo viên quản lý. 3. Giáo viên chọn khóa học muốn sửa và nhấn vào nút "Chỉnh sửa". 4. Hệ thống hiển thị form với các thông tin hiện tại của khóa học. 5. Giáo viên thay đổi thông tin cần thiết (tên, mô tả, giá tiền,...). 6. Giáo viên nhấn "Lưu thay đổi".

	7. Hệ thống xác nhận và cập nhật thông tin mới, thông báo cập nhật thành công.
--	--

Bảng- 2-13: Đặc tả Use Case: Xóa khóa học

Mã ca sử dụng	UC-13
Tên ca sử dụng	Xóa khóa học
Mô tả	Cho phép Giáo viên xóa một khóa học do mình quản lý khỏi hệ thống.
Tác nhân	Teacher (Giáo viên), Admin (Quản trị viên)
Điều kiện trước	- Giáo viên đã đăng nhập. - Khóa học cần xóa đã tồn tại và thuộc quyền quản lý của giáo viên đó.
Điều kiện sau	- Khóa học và tất cả các dữ liệu liên quan (bài học, yêu cầu tham gia...) bị xóa khỏi hệ thống.
Các bước thực hiện	<ol style="list-style-type: none"> 1. Giáo viên vào trang "Quản lý khóa học". 2. Từ danh sách khóa học, giáo viên chọn khóa học muốn xóa và nhấn nút "Xóa". 3. Hệ thống hiển thị thông báo yêu cầu xác nhận hành động xóa. 4. Giáo viên xác nhận. 5. Hệ thống xóa khóa học và các dữ liệu liên quan khỏi cơ sở dữ liệu, thông báo xóa thành công và cập nhật lại danh sách khóa học.

Bảng- 2-14: Đặc tả Use Case: Duyệt yêu cầu tham gia khóa học

Mã ca sử dụng	UC-14
Tên ca sử dụng	Duyệt yêu cầu tham gia khóa học

Mô tả	Cho phép Giáo viên xem xét và phê duyệt hoặc từ chối yêu cầu tham gia khóa học từ các học viên.
Tác nhân	Teacher (Giáo viên)
Điều kiện trước	- Giáo viên đã đăng nhập. - Có ít nhất một học viên gửi yêu cầu tham gia một khóa học của giáo viên.
Điều kiện sau	- Yêu cầu của học viên được cập nhật trạng thái (được duyệt hoặc bị từ chối). - Nếu được duyệt, học viên sẽ có quyền thanh toán khóa học.
Các bước thực hiện	<ol style="list-style-type: none"> 1. Giáo viên truy cập vào trang quản lý của một khóa học cụ thể. 2. Giáo viên chọn tab/mục "Yêu cầu tham gia". 3. Hệ thống hiển thị danh sách các học viên đang chờ duyệt. 4. Với mỗi yêu cầu, giáo viên có thể chọn "Đồng ý" hoặc "Từ chối". 5. Hệ thống cập nhật trạng thái yêu cầu, gửi thông báo (email và thông báo trên web) đến học viên về kết quả xét duyệt.

Bảng- 2-15: Đặc tả Use Case: Xem thống kê

Mã ca sử dụng	UC-18
Tên ca sử dụng	Xem thống kê thu nhập, học viên, khóa học
Mô tả	Cho phép Giáo viên xem các báo cáo, thống kê tổng quan về hoạt động giảng dạy của mình trên hệ thống. Cho phép Quản trị viên xem báo cáo, thống kê tổng quan về hệ thống
Tác nhân	Teacher (Giáo viên), Admin (Quản trị viên)
Điều kiện trước	- Giáo viên / Quản trị viên đã đăng nhập.

Điều kiện sau	- Giáo viên / Quản trị viên nắm được các số liệu về hiệu quả hoạt động của mình.
Các bước thực hiện	1. Người dùng (giáo viên / quản trị viên) chọn mục "Thống kê" hoặc "Báo cáo" từ bảng điều khiển. 2. Hệ thống hiển thị trang thống kê với các biểu đồ, số liệu.

2.4. Lược đồ cơ sở dữ liệu

2.4.1. Mô hình thực thể kết hợp ERD



Hình- 2-5: Mô hình thực thể kết hợp ERD

2.4.2. Mô hình quan hệ dữ liệu

Tên trường	Kiểu dữ liệu	Mô tả
id	Kiểu bigint	Mã tài khoản – tự sinh bởi cơ sở dữ liệu
email	Kiểu chuỗi	Email của tài khoản
password_digest	Kiểu chuỗi	Mật khẩu đã được hash
roles	Kiểu integer	Quyền hạn (mặc định: 0)
created_at	Kiểu timestamp	Ngày tạo tài khoản
updated_at	Kiểu timestamp	Ngày cập nhật cuối
activation_token	Kiểu chuỗi	Token kích hoạt
activated	Kiểu boolean	Tình trạng kích hoạt (mặc định: false)
activated_at	Kiểu timestamp	Ngày kích hoạt tài khoản
reset_password_token	Kiểu chuỗi	Chuỗi token xác thực đổi mật khẩu
reset_password_sent_at	Kiểu timestamp	Ngày gửi chuỗi token xác thực đổi mật khẩu
status	Kiểu enum	Tình trạng tài khoản

Bảng- 2-17: Bảng categories

Tên trường	Kiểu dữ liệu	Mô tả
id	Kiểu bigint	Mã danh mục – tự sinh bởi cơ sở dữ liệu
name	Kiểu chuỗi	Tên danh mục
description	Kiểu text	Mô tả danh mục
created_at	Kiểu timestamp	Ngày tạo danh mục
updated_at	Kiểu timestamp	Ngày cập nhật cuối danh mục

Bảng- 2-18: Bảng courses

Tên trường	Kiểu dữ liệu	Mô tả
------------	--------------	-------

id	Kiểu bigint	Mã khóa học – tự sinh bởi cơ sở dữ liệu
category_id	Kiểu bigint	Mã danh mục khóa học
title	Kiểu chuỗi	Tên khóa học
level	Kiểu chuỗi	Trình độ khóa học
description	Kiểu text	Mô tả khóa học
teacher_id	Kiểu bigint	Mã giáo viên
image_url	Kiểu chuỗi	Đường dẫn ảnh
price	Kiểu int	Giá khóa học
auto_accept_assign	Kiểu boolean	Khóa học có tự động chấp nhận yêu cầu tham gia hay không
created_at	Kiểu timestamp	Ngày tạo khóa học
updated_at	Kiểu timestamp	Ngày cập nhật cuối khóa học

Bảng- 2-19: Bảng lessons

Tên trường	Kiểu dữ liệu	Mô tả
id	Kiểu bigint	Mã bài học – tự sinh bởi cơ sở dữ liệu
course_id	Kiểu bigint	Mã khóa học
title	Kiểu chuỗi	Tên bài học
content	Kiểu text	Nội dung bài học
video_url	Kiểu chuỗi	Đường dẫn video bài học
created_at	Kiểu timestamp	Ngày tạo bài học
updated_at	Kiểu timestamp	Ngày cập nhật cuối bài học

Bảng- 2-20: Bảng follows

Tên trường	Kiểu dữ liệu	Mô tả
------------	--------------	-------

id	Kiểu bigint	Mã theo dõi – tự sinh bởi cơ sở dữ liệu
teacher_id	Kiểu bigint	Mã giáo viên được theo dõi
user_id	Kiểu bigint	Mã tài khoản của người theo dõi
created_at	Kiểu timestamp	Ngày bắt đầu theo dõi
updated_at	Kiểu timestamp	Ngày cập nhật cuối

Bảng- 2-21: Bảng users

Tên trường	Kiểu dữ liệu	Mô tả
id	Kiểu bigint	Mã người dùng – tự sinh bởi cơ sở dữ liệu
account_id	Kiểu bigint	Mã tài khoản
name	Kiểu chuỗi	Tên người dùng
avatar_url	Kiểu chuỗi	Đường dẫn ảnh đại diện
created_at	Kiểu timestamp	Ngày tạo
updated_at	Kiểu timestamp	Ngày cập nhật cuối

Bảng- 2-22: Bảng teachers

Tên trường	Kiểu dữ liệu	Mô tả
id	Kiểu bigint	Mã giáo viên – tự sinh bởi cơ sở dữ liệu
account_id	Kiểu bigint	Mã tài khoản
bio	Kiểu text	Tiểu sử giáo viên
image_url	Kiểu string	Ảnh avatar
experience	Kiểu text	Kinh nghiệm
job_title	Kiểu text	Vị trí giảng dạy
name	Kiểu string	Tên
bank_account_number	Kiểu string	Số tài khoản ngân hàng
bank_name	Kiểu string	Tên ngân hàng

bank_account_holder	Kiểu string	Tên tài khoản
created_at	Kiểu timestamp	Ngày tạo
updated_at	Kiểu timestamp	Ngày cập nhật cuối

Bảng- 2-23: Bảng progresses

Tên trường	Kiểu dữ liệu	Mô tả
id	Kiểu bigint	Mã tiến độ – tự sinh bởi cơ sở dữ liệu
user_id	Kiểu bigint	Mã người dùng
lesson_id	Kiểu bigint	Mã bài học
status	Kiểu enum	Trạng thái hoàn thành
score	Kiểu int	Điểm
started_at	Kiểu timestamp	Ngày bắt đầu
completed_at	Kiểu timestamp	Ngày hoàn thành
created_at	Kiểu timestamp	Ngày tạo
updated_at	Kiểu timestamp	Ngày cập nhật cuối

Bảng- 2-24: Bảng kanjis

Tên trường	Kiểu dữ liệu	Mô tả
id	Kiểu bigint	Mã kanji – tự sinh bởi cơ sở dữ liệu
character	Kiểu chuỗi	Ký tự Kanji
lesson_id	Kiểu bigint	Mã bài học
created_at	Kiểu timestamp	Ngày tạo
updated_at	Kiểu timestamp	Ngày cập nhật cuối

Bảng- 2-25: Bảng flashcards

Tên trường	Kiểu dữ liệu	Mô tả
------------	--------------	-------

id	Kiểu bigint	Mã flashcard – tự sinh bởi cơ sở dữ liệu
lesson_id	Kiểu bigint	Mã bài học
front_content	Kiểu chuỗi	Nội dung mặt trước
back_content	Kiểu chuỗi	Nội dung mặt sau
created_at	Kiểu timestamp	Ngày tạo
updated_at	Kiểu timestamp	Ngày cập nhật cuối
id	Kiểu chuỗi	Mã flashcard – tự sinh bởi cơ sở dữ liệu

Bảng- 2-26: Bảng course_ratings

Tên trường	Kiểu dữ liệu	Mô tả
id	Kiểu bigint	Mã đánh giá – tự sinh bởi cơ sở dữ liệu
course_id	Kiểu bigint	Mã khóa học
user_id	Kiểu bigint	Mã người dùng
rating	Kiểu integer	Điểm đánh giá (1-5)
review	Kiểu text	Bình luận
created_at	Kiểu timestamp	Ngày tạo
updated_at	Kiểu timestamp	Ngày cập nhật cuối

Bảng- 2-27: Bảng course_assignments

Tên trường	Kiểu dữ liệu	Mô tả
id	Kiểu bigint	Mã phân công – tự sinh bởi cơ sở dữ liệu
user_id	Kiểu bigint	Mã người dùng
course_id	Kiểu bigint	Mã khóa học
assigned_at	Kiểu timestamp	Ngày tạo đăng kí
status	Kiểu enum	Tình trạng yêu cầu đăng kí
created_at	Kiểu timestamp	Ngày tạo

updated_at	Kiểu timestamp	Ngày cập nhật cuối
------------	----------------	--------------------

Bảng- 2-28: Bảng exercise_questions

Tên trường	Kiểu dữ liệu	Mô tả
id	Kiểu bigint	Mã phân công – tự sinh bởi cơ sở dữ liệu
question_type	Kiểu chuỗi	Nội dung câu hỏi
question_text	Kiểu text	Mã khóa học
media_url	Kiểu text	Đường dẫn ảnh/ video/audio
options	Kiểu json	Câu trả lời
correct_answer	Kiểu string	Đáp án đúng
created_at	Kiểu timestamp	Ngày tạo
updated_at	Kiểu timestamp	Ngày cập nhật cuối

Bảng- 2-29: forum_posts

Tên trường	Kiểu dữ liệu	Mô tả
id	Kiểu bigint	Mã phân công – tự sinh bởi cơ sở dữ liệu
course_id	Kiểu bigint	Mã khóa học
author_type	Kiểu string	Kiểu tác giả
author_id	Kiểu bigint	Mã tác giả
parent_id	Kiểu bigint	Mã post cha
title	Kiểu string	Tiêu đề post
content	Kiểu text	Nội dung post
attachments	Kiểu json	Đính kèm post
created_at	Kiểu timestamp	Ngày tạo
updated_at	Kiểu timestamp	Ngày cập nhật cuối

Bảng- 2-30: Bảng Notifications

Tên trường	Kiểu dữ liệu	Mô tả
account_id	bigint	ID của tài khoản nhận thông báo (không được null)
title	string	Tiêu đề thông báo (không được null)
message	text	Nội dung chi tiết của thông báo (không được null)
notification_type	string	Loại thông báo (info, warning, error, v.v.), mặc định là "info"
reference_id	string	ID tham chiếu đến đối tượng liên quan (nếu có)
reference_type	string	Kiểu đối tượng được tham chiếu (ví dụ: "Course", "User")
read	boolean	Trạng thái đã đọc hay chưa, mặc định là false
created_at	datetime	Thời gian tạo thông báo
updated_at	datetime	Thời gian cập nhật cuối cùng

Bảng- 2-31: Bảng Payments

Tên trường	Kiểu dữ liệu	Mô tả
course_assignment_id	bigint	ID của bản ghi khóa học mà người dùng đã đăng ký (không được null)
amount	integer	Số tiền thanh toán (đơn vị: đồng, yên, etc.)
transaction_date	datetime	Thời gian giao dịch diễn ra
transaction_code	string	Mã giao dịch
payment_method	string	Phương thức thanh toán (ví dụ: "credit_card", "bank_transfer")

bank_code	string	Mã ngân hàng được sử dụng
created_at	datetime	Thời gian tạo bản ghi thanh toán
updated_at	datetime	Thời gian cập nhật cuối cùng

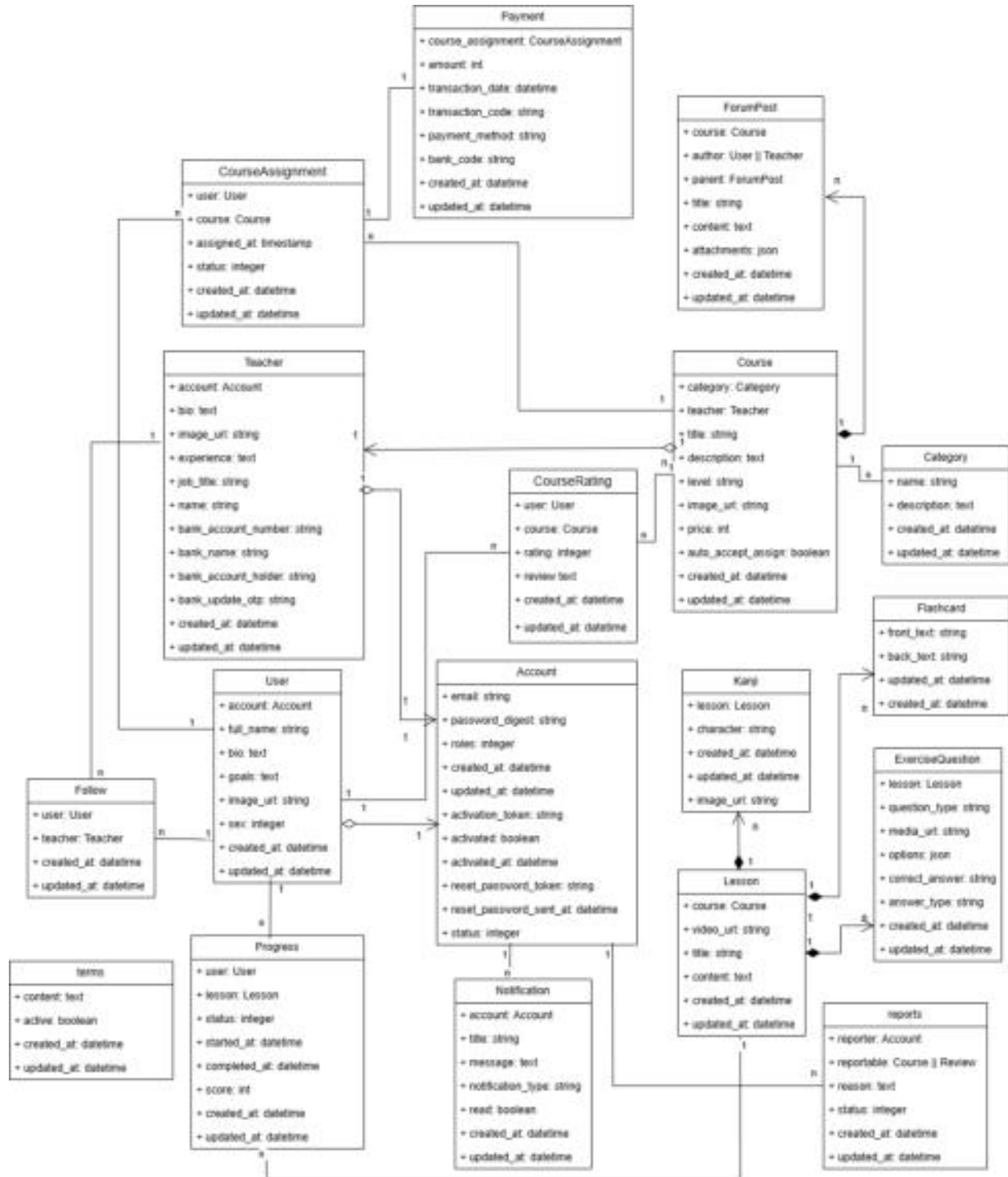
Bảng- 2-32: Bảng Reports

Tên trường	Kiểu dữ liệu	Mô tả
reporter_id	bigint	ID của người gửi báo cáo (không được null)
reportable_type	string	Kiểu đối tượng bị báo cáo (ví dụ: "User", "Comment") (không được null)
reportable_id	bigint	ID của đối tượng bị báo cáo (không được null)
reason	text	Lý do báo cáo (không được null)
status	integer	Trạng thái xử lý báo cáo (0: đang chờ, 1: đã xử lý...), mặc định là 0
created_at	datetime	Thời gian tạo báo cáo
updated_at	datetime	Thời gian cập nhật cuối cùng

Bảng- 2-33: Bảng terms

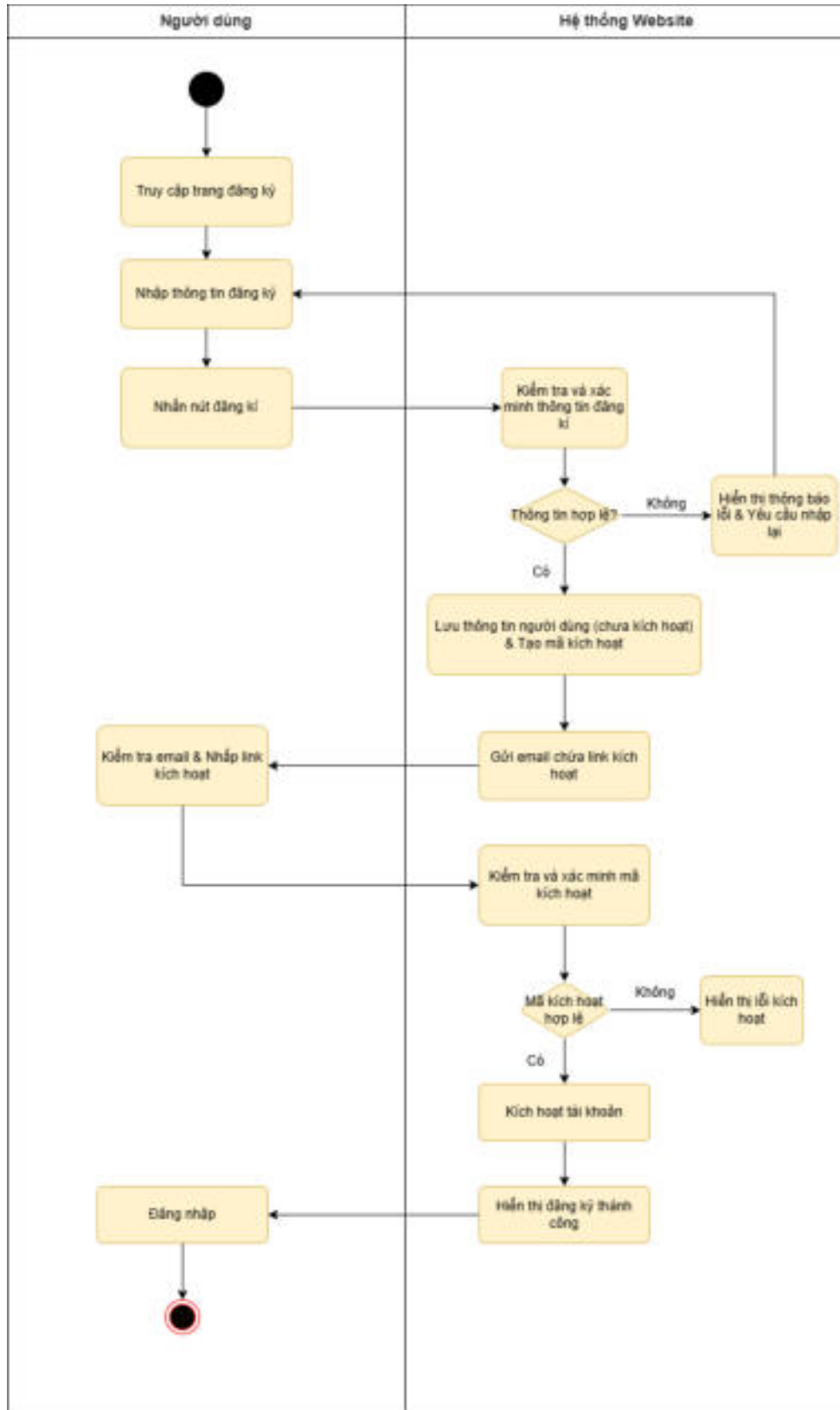
Tên trường	Kiểu dữ liệu	Mô tả
id	Kiểu bigint	Mã theo dõi – tự sinh bởi cơ sở dữ liệu
content	Kiểu text	Nội dung
active	Kiểu boolean	Trạng thái kích hoạt
created_at	Kiểu timestamp	Ngày bắt đầu theo dõi
updated_at	Kiểu timestamp	Ngày cập nhật cuối

2.5. Biểu đồ lớp

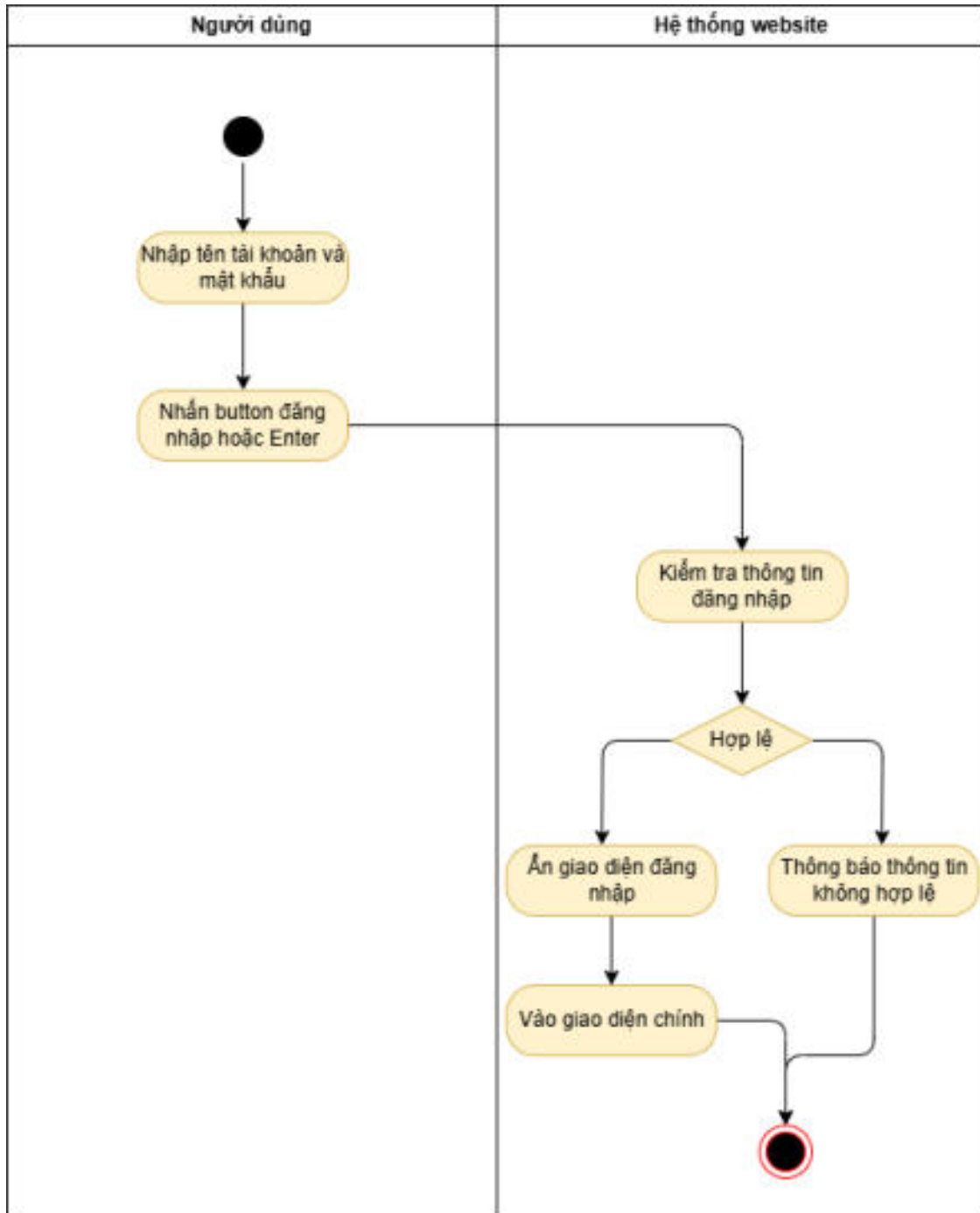


Hình- 2-7: Biểu đồ lớp

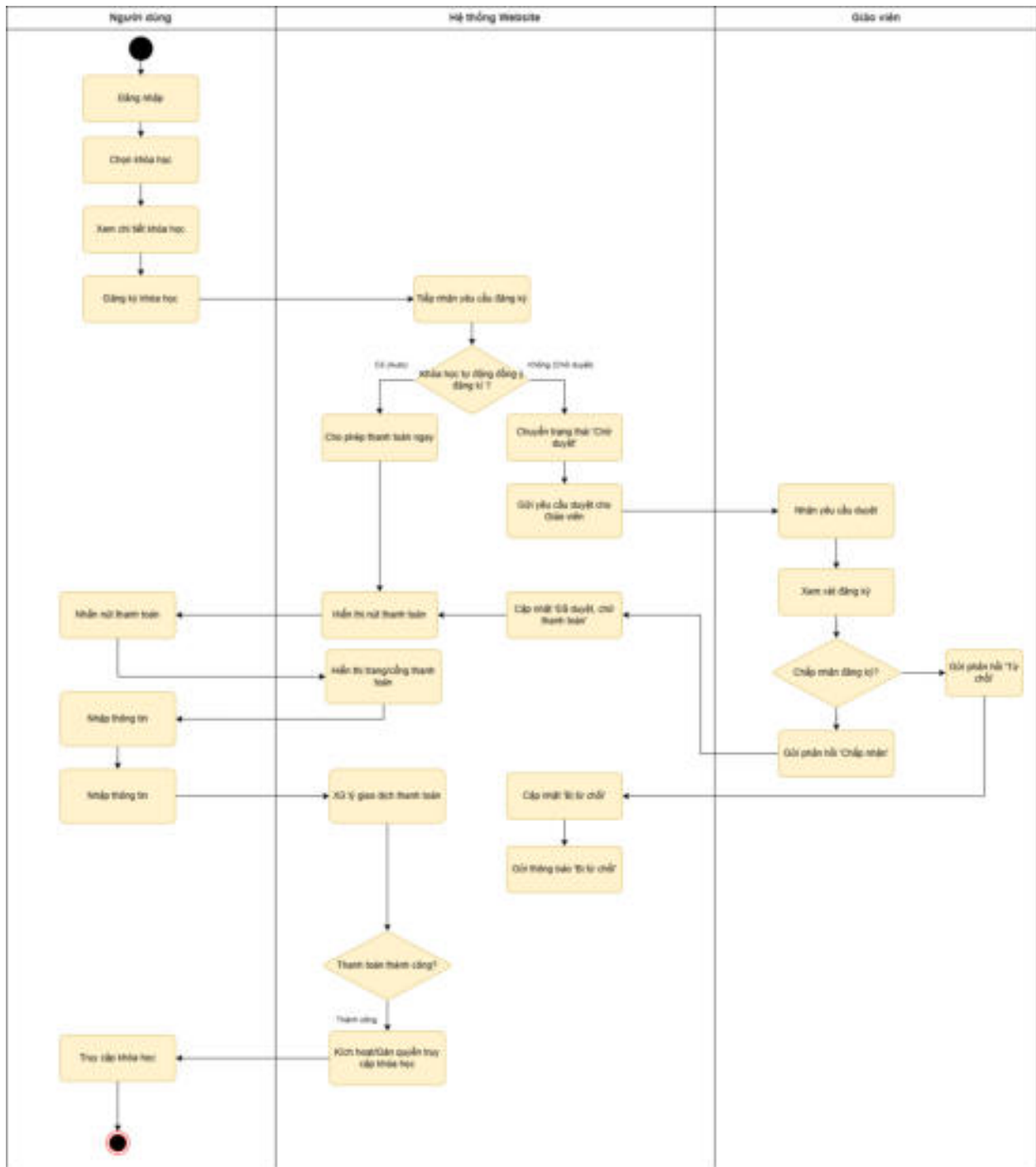
2.6. Sơ đồ hoạt động



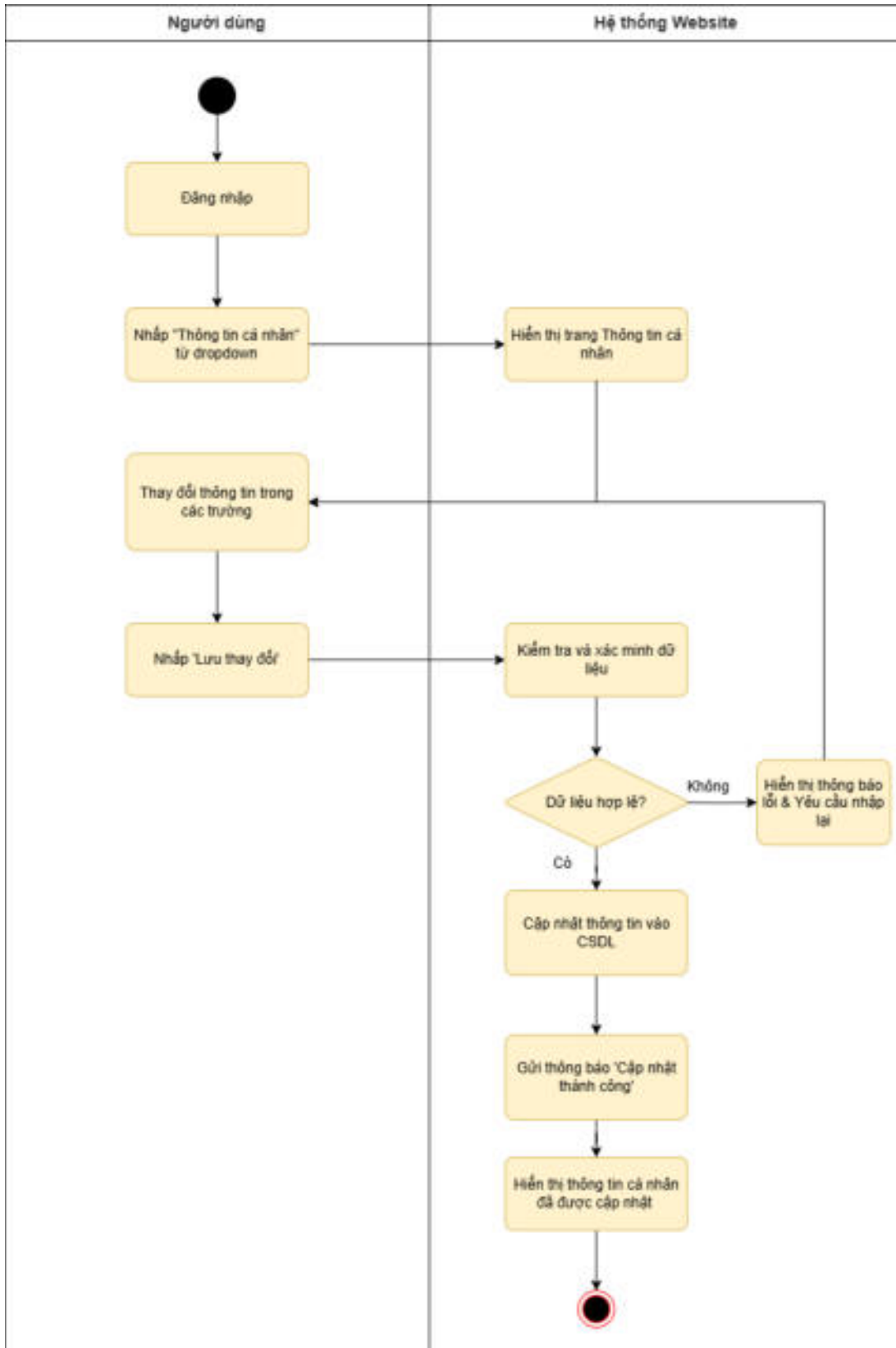
Hình- 2-8: Sơ đồ hoạt động đăng kí



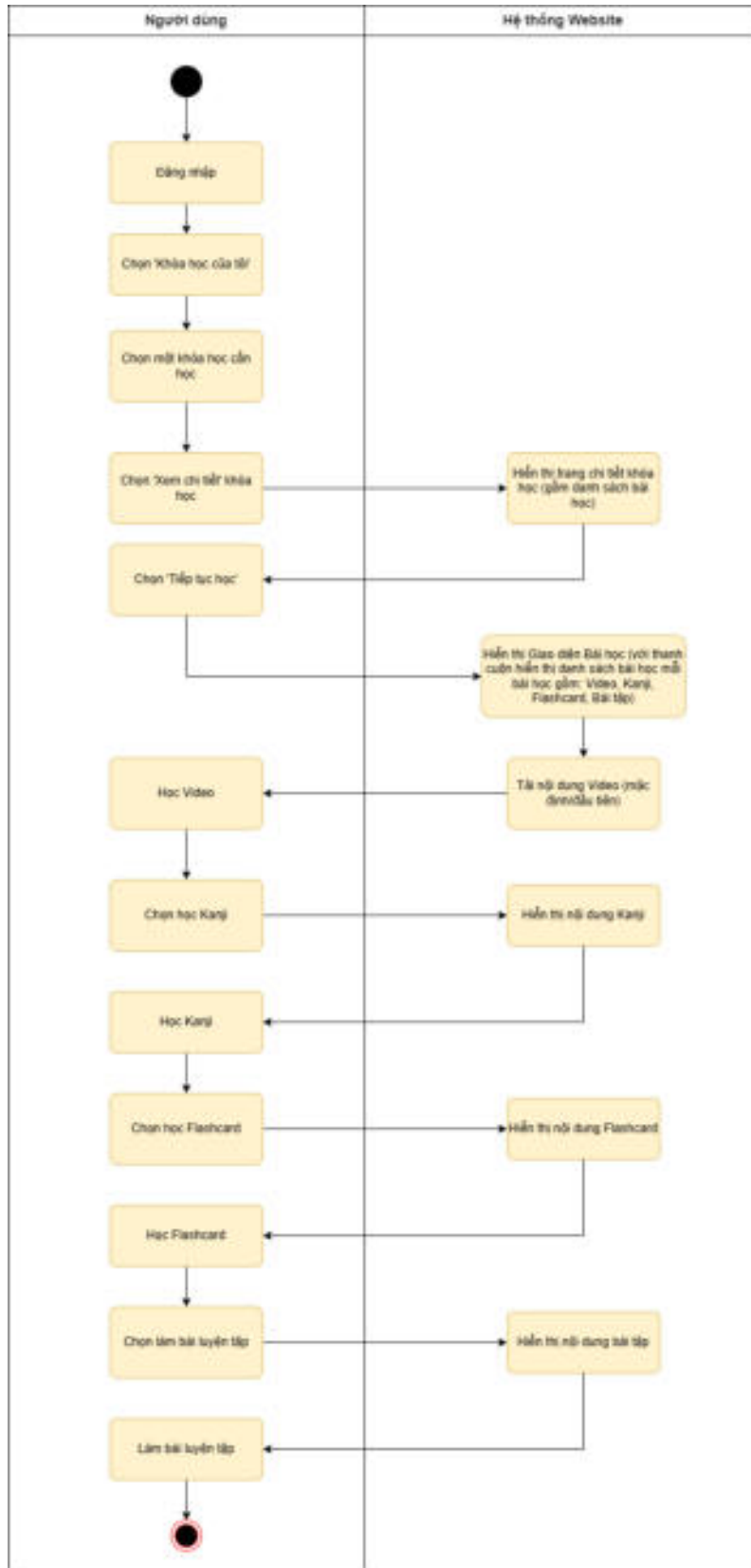
Hình- 2-9: Sơ đồ hoạt động đăng nhập



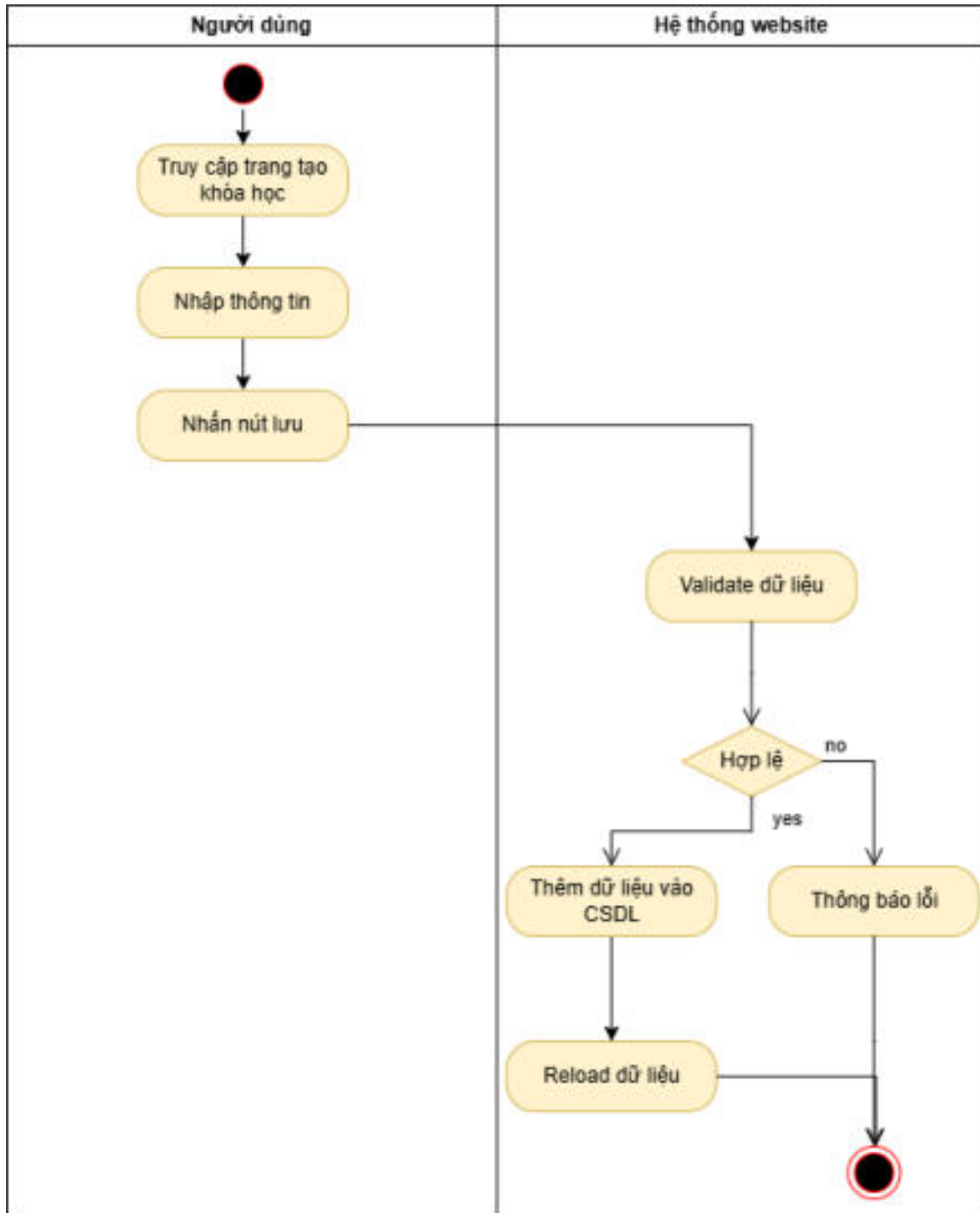
Hình- 2-10: Sơ đồ hoạt động thanh toán khóa học



Hình- 2-11: Sơ đồ hoạt động cập nhật thông tin cá nhân

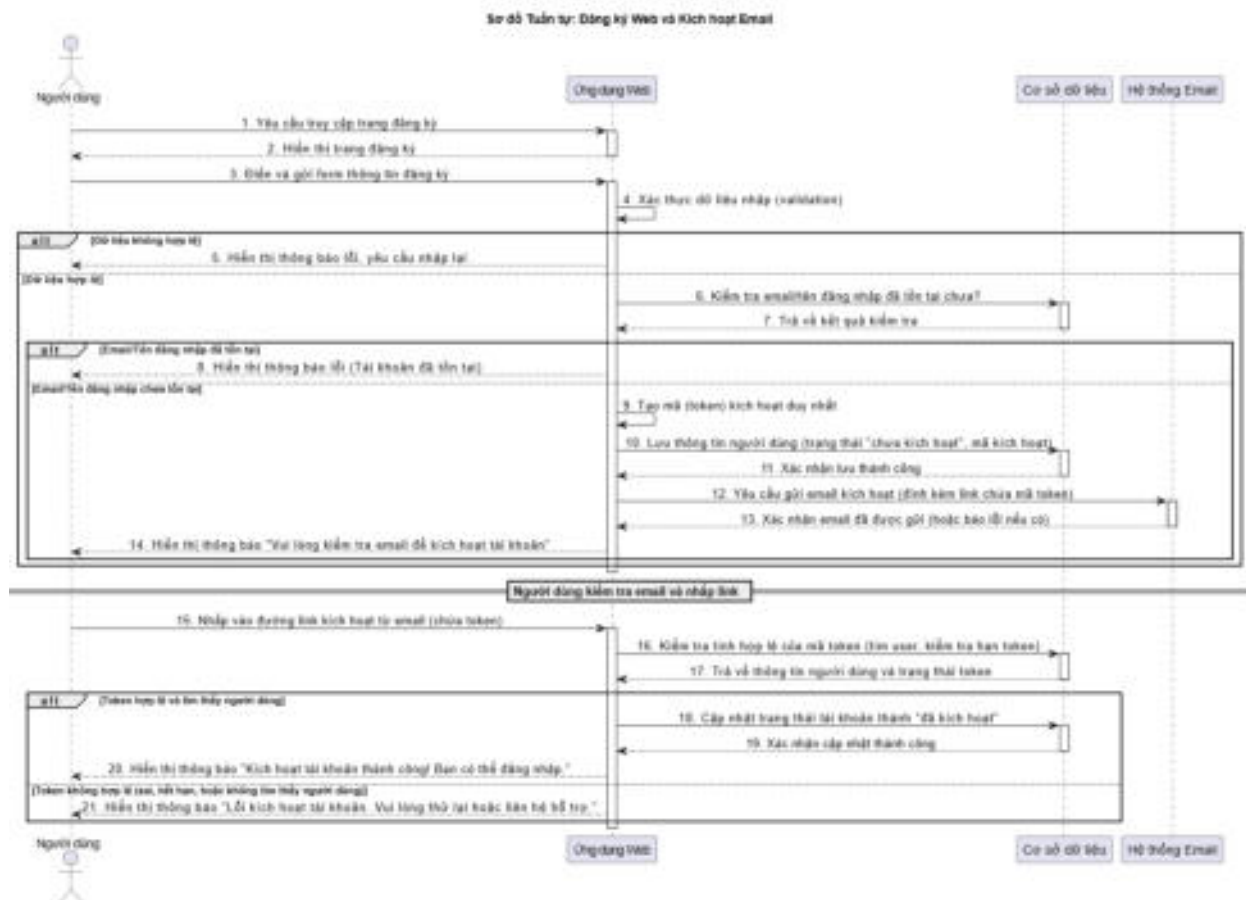


Hình- 2-12: Sơ đồ hoạt động học bài học

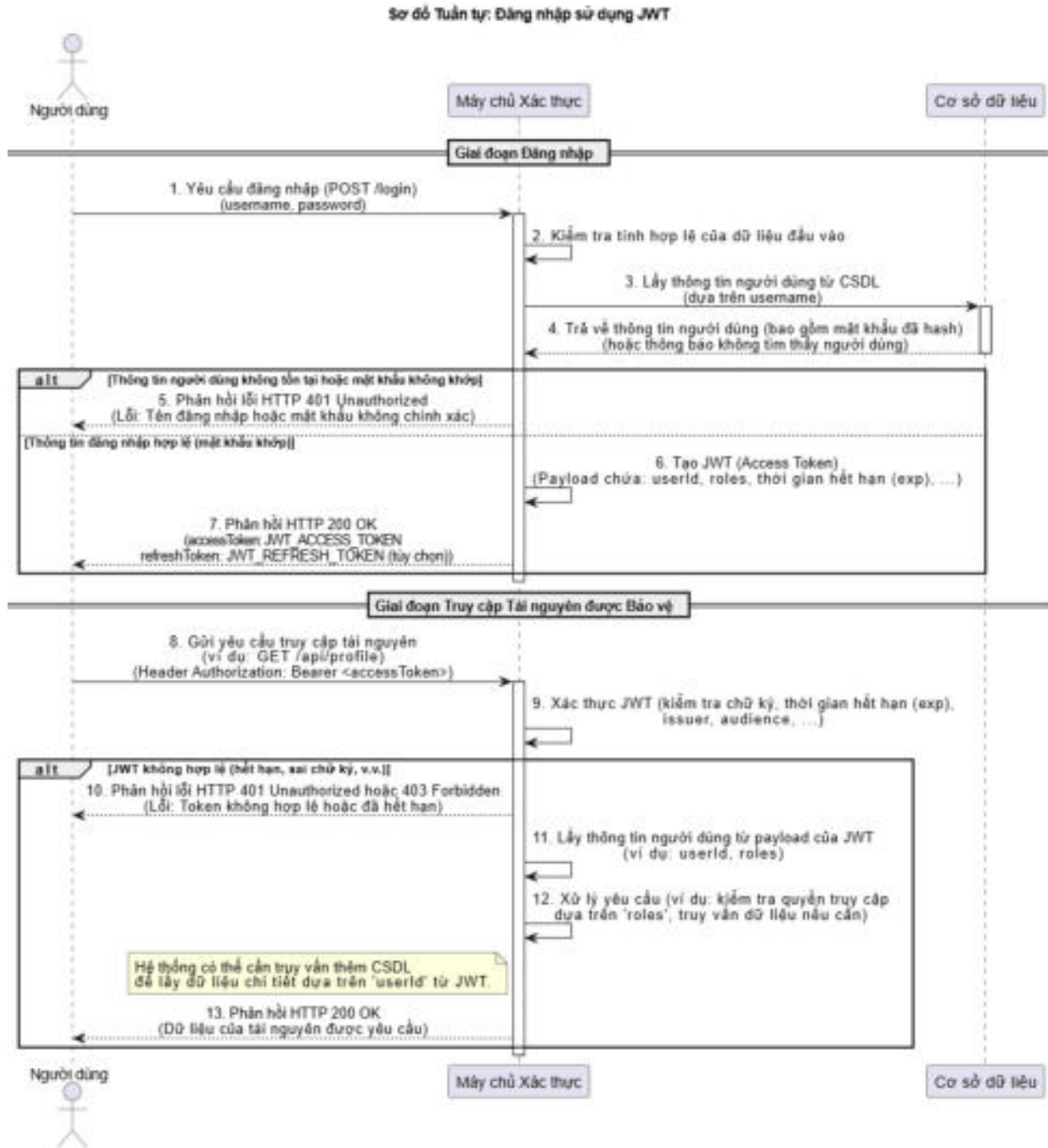


Hình- 2-13: Sơ đồ hoạt động tạo khóa học

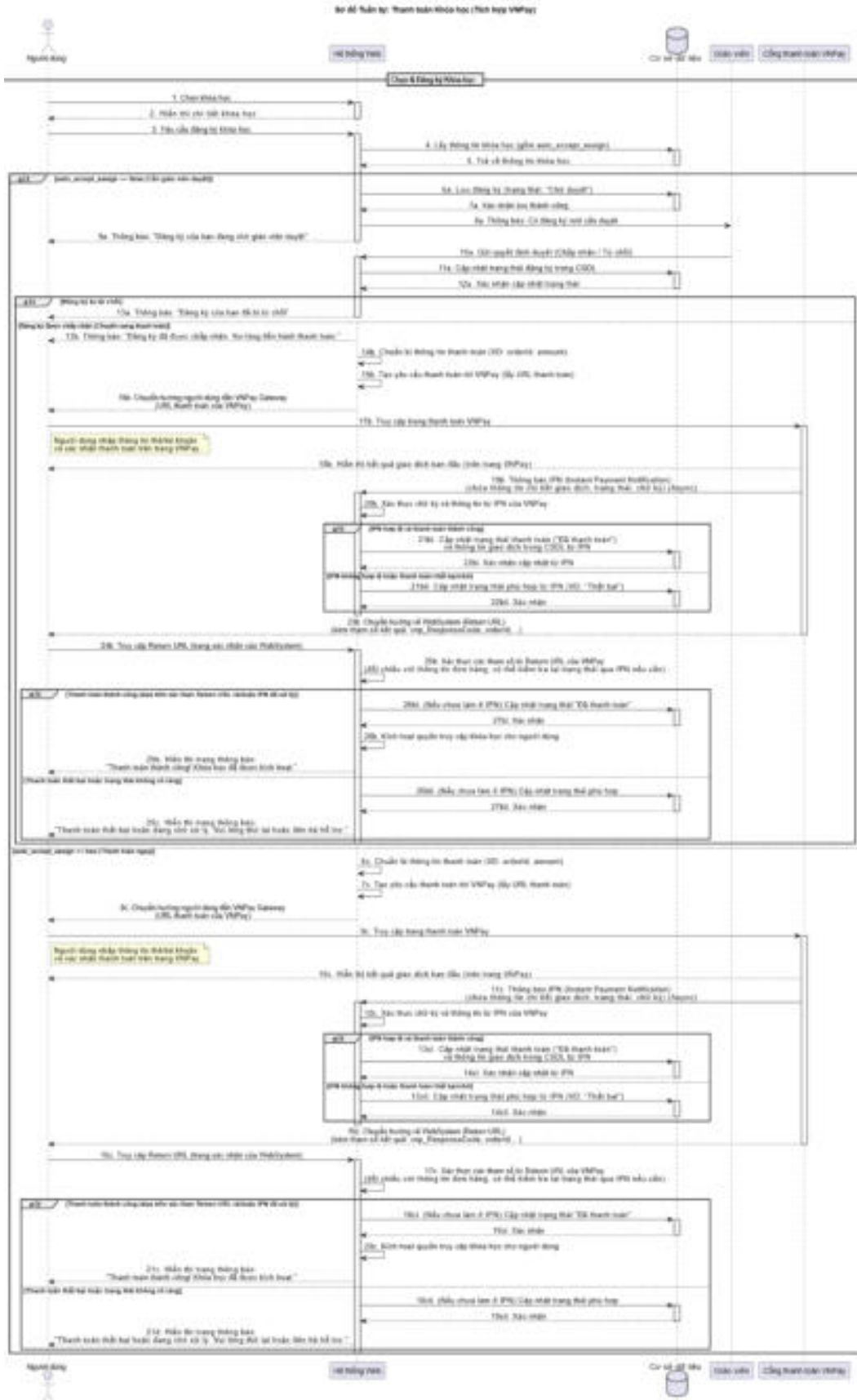
2.7. Sơ đồ tuần tự



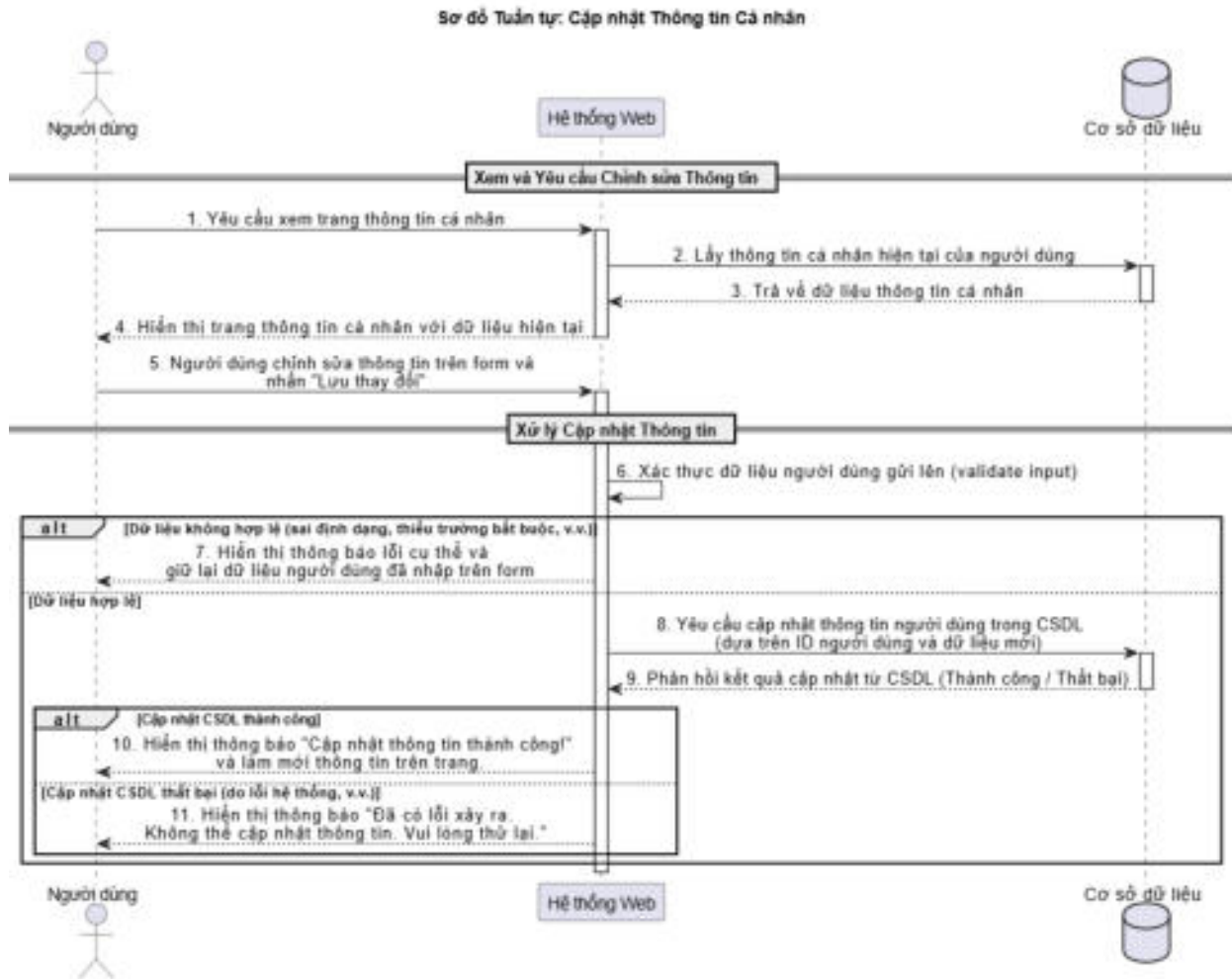
Hình- 2-14: Sơ đồ tuần tự đăng kí và kích hoạt mail



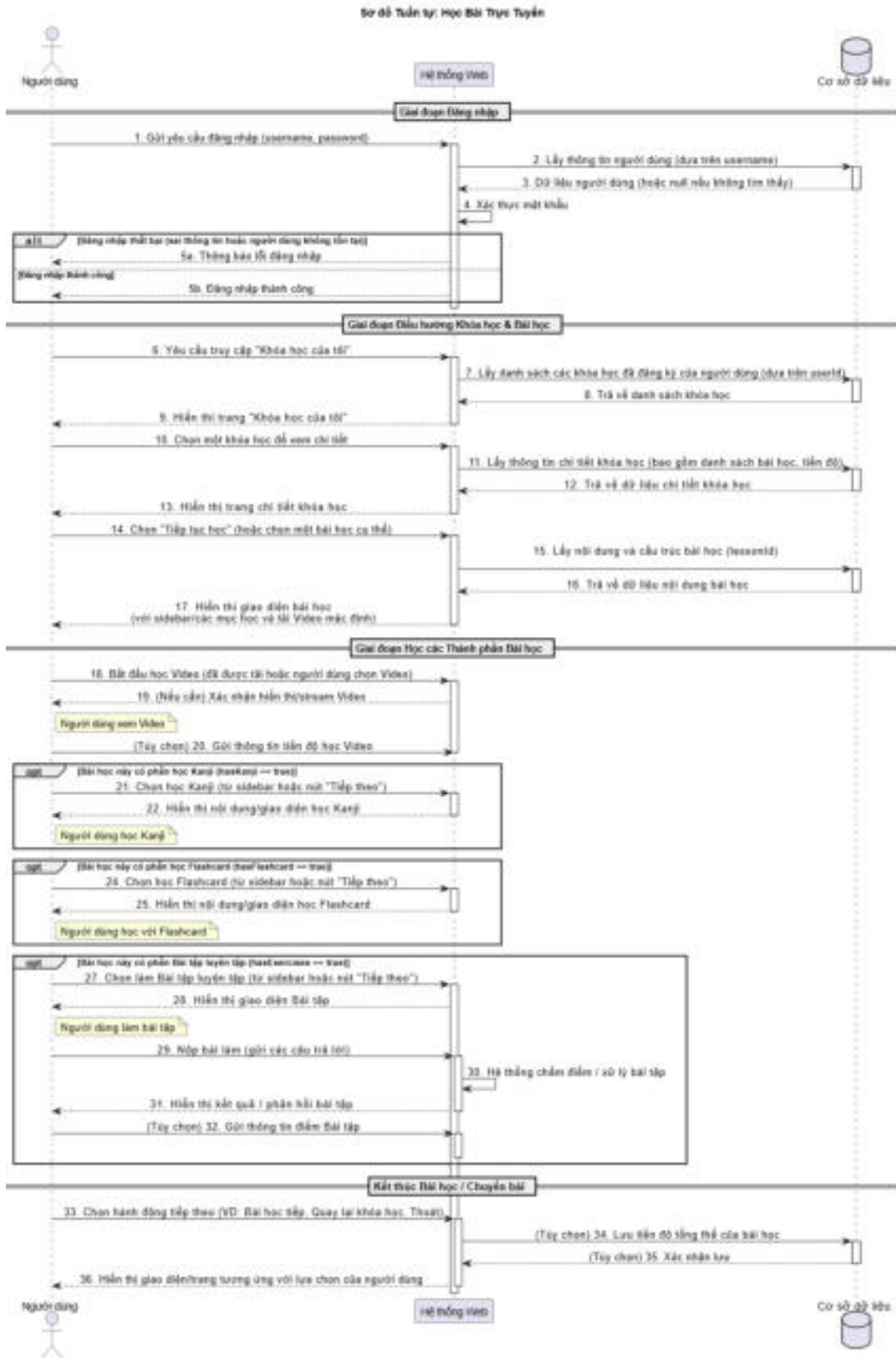
Hình- 2-15: Sơ đồ tuần tự đăng nhập và xác thực người dùng



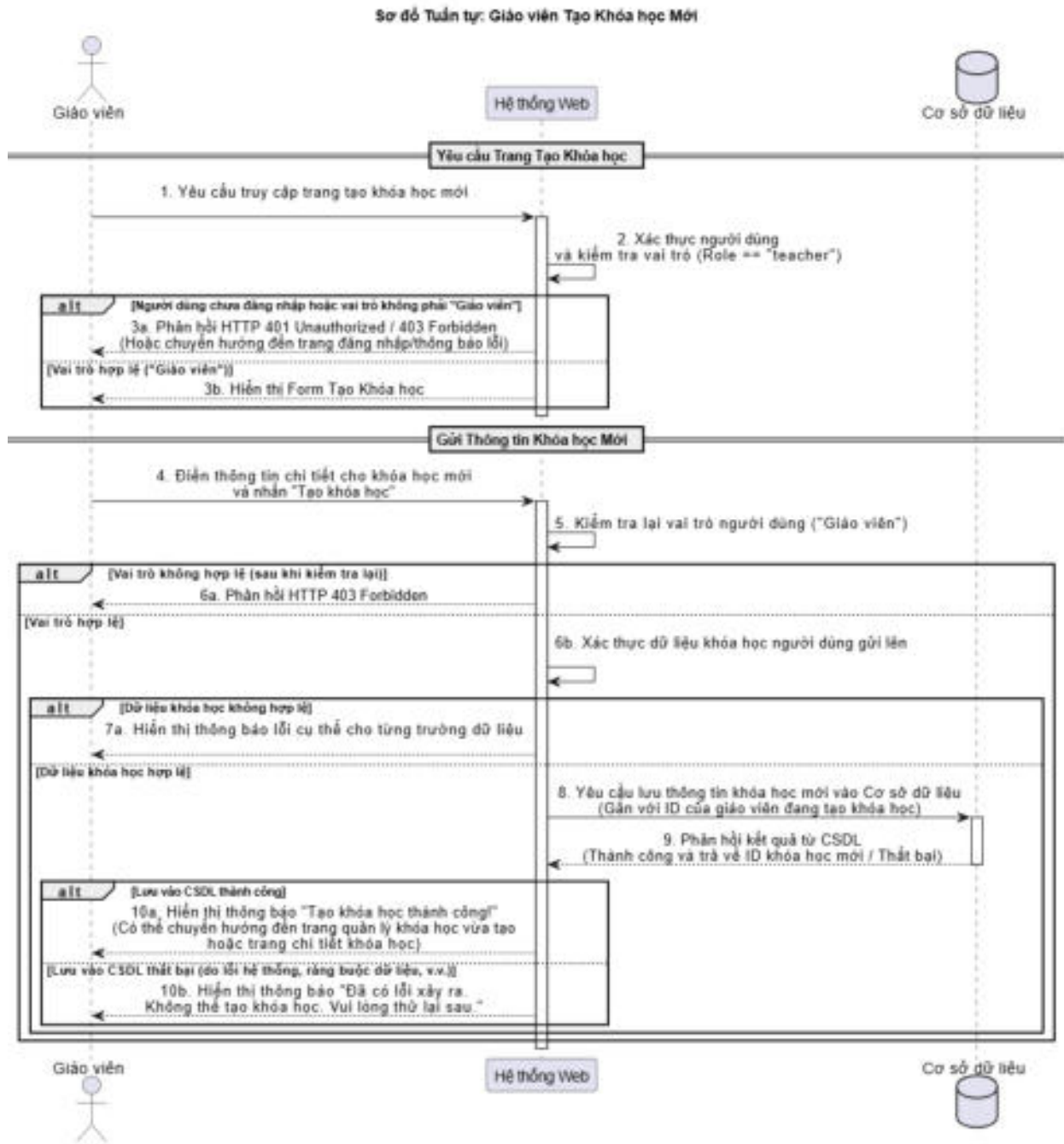
Hình- 2-16: Sơ đồ tuần tự thanh toán khóa học



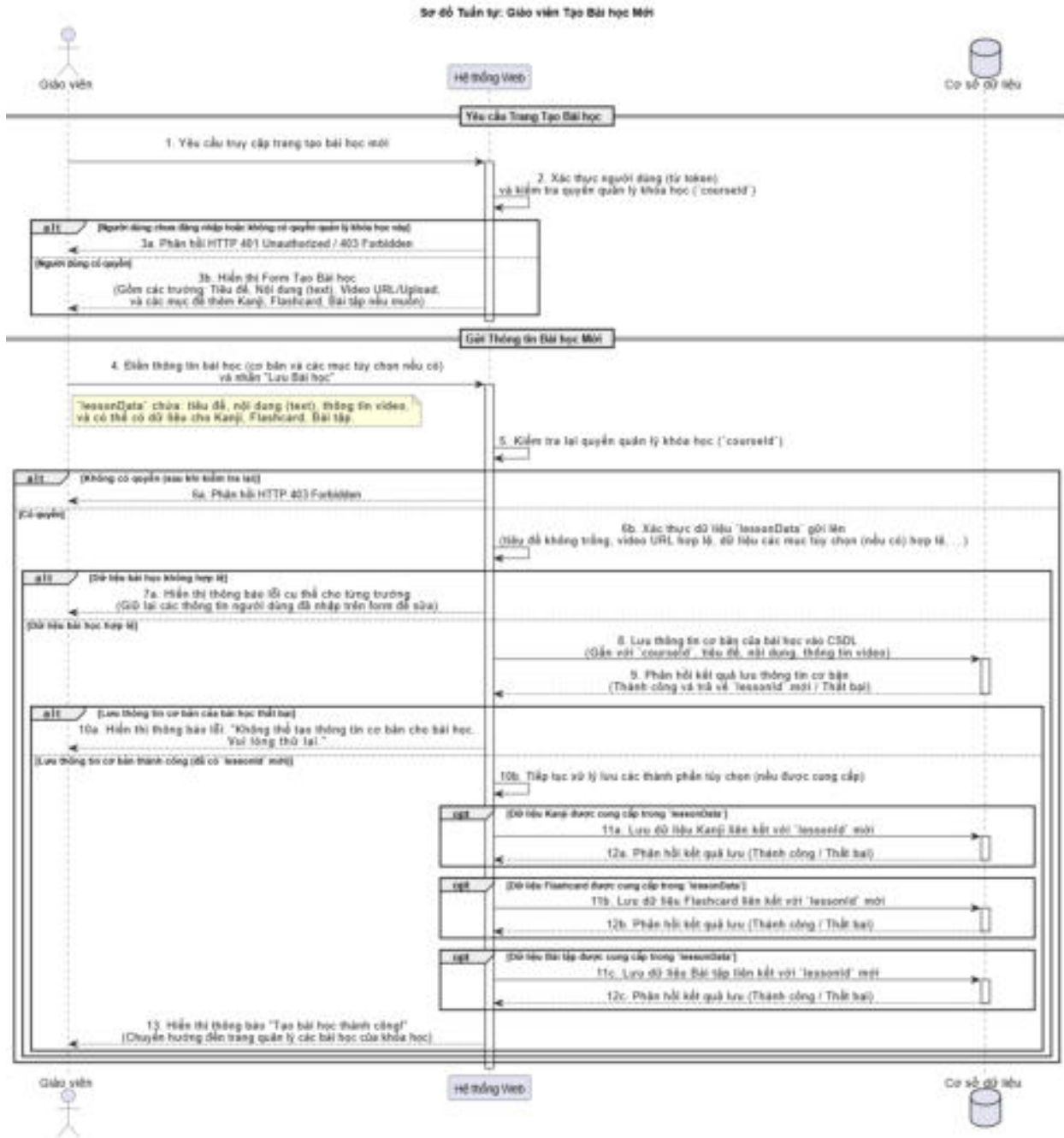
Hình- 2-17: Sơ đồ tuần tự cập nhật thông tin cá nhân



Hình- 2-18: Sơ đồ tuần tự học bài học



Hình- 2-19: Sơ đồ tuần tự giáo viên tạo khóa học



Hình- 2-20: Sơ đồ tuần tự tạo bài học mới

2.8. Kết luận chương 2

Chương 2 đã trình bày quá trình phân tích và thiết kế hệ thống một cách toàn diện, bao gồm việc xác định các yêu cầu chức năng và phi chức năng, từ đó làm cơ sở cho việc xây dựng hệ thống hiệu quả và phù hợp với nhu cầu thực tế.

Các sơ đồ Use Case đã được xây dựng nhằm mô tả rõ các chức năng chính và mối quan hệ giữa các tác nhân với hệ thống. Dựa trên phân tích chức năng, lược đồ cơ sở dữ liệu đã được thiết kế để đảm bảo khả năng lưu trữ, truy xuất và quản lý dữ liệu một cách hợp lý, hiệu quả.

Bên cạnh đó, sơ đồ hoạt động và sơ đồ tuần tự được sử dụng để thể hiện chi tiết luồng xử lý nghiệp vụ và sự tương tác giữa các thành phần trong hệ thống. Việc sử dụng các sơ đồ này giúp hình dung rõ ràng quy trình xử lý và hỗ trợ phát hiện sớm những điểm bất hợp lý trong thiết kế.

Việc hoàn thiện các bước phân tích và thiết kế ở chương này đóng vai trò quan trọng trong việc định hướng cho giai đoạn triển khai, đảm bảo hệ thống được phát triển một cách có cấu trúc, logic và đáp ứng tốt yêu cầu đặt ra.

CHƯƠNG 3: TRIỂN KHAI HỆ THỐNG VÀ KẾT QUẢ

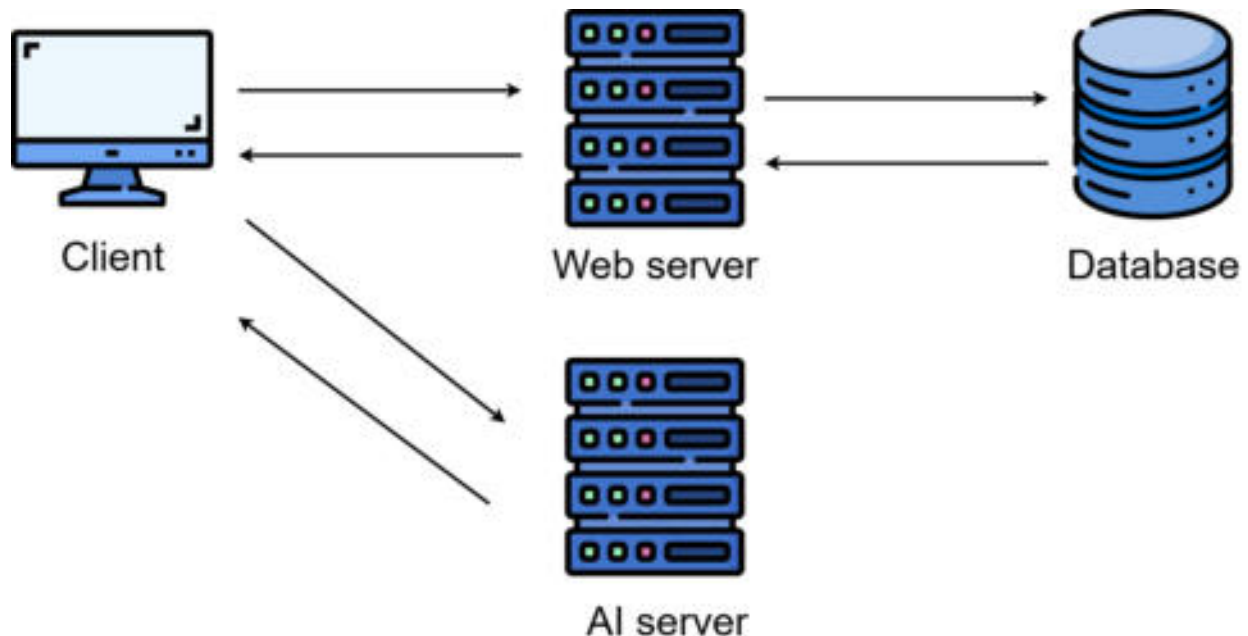
3.1. Cấu trúc hệ thống

3.1.1. Tổng quan

Để giải quyết bài toán phức tạp: Từ một hệ thống lớn phức tạp, sẽ đi chia hệ thống đó thành các hệ thống con nhỏ hơn, đơn giản hơn mỗi hệ thống đó sẽ đảm nhận một phần công việc khác nhau.

Kiến trúc hệ thống bao gồm bốn thành phần chính:

- Client: Đây là thành phần giao tiếp trực tiếp với người dùng. Người dùng sẽ gửi yêu cầu từ Client.
- Main Server: Đây là máy chủ chính của hệ thống, nhận yêu cầu từ Client, quản lý toàn bộ logic nghiệp vụ, xác thực người dùng, lưu trữ và xử lý dữ liệu học tập.
- AI Server: Đây là máy chủ AI, nơi các thuật toán AI chạy để xử lý các yêu cầu phức tạp từ Client.
- Database: Cơ sở dữ liệu là nơi lưu trữ toàn bộ thông tin của hệ thống, đóng vai trò trung tâm trong việc quản lý và truy xuất dữ liệu.



Hình- 3-1: Tổng quan hệ thống

3.1.2. Luồng xử lý

Luồng xử lý trong hệ thống diễn ra như sau:

- Client gửi yêu cầu đến Web Server để xử lý các tác vụ liên quan đến: dữ liệu người dùng, quản lý khóa học, bài tập, tiến độ học tập
- Nếu yêu cầu liên quan đến các tác vụ trí tuệ nhân tạo, chẳng hạn như tra cứu kanji:
 - Client gửi yêu cầu đến AI Server
 - AI Server nhận và xử lý yêu cầu từ Client
 - AI Server thực hiện các tính toán hoặc xử lý dữ liệu cần thiết
 - AI Server gửi kết quả trở lại Client.

3.1.3. Kỹ thuật công nghệ

a. AI

- Model : MobileNet
- Ngôn ngữ : Python
- Framework : Flask

b. Backend

- Ngôn ngữ : Ruby
- Framework : Ruby on Rails
- Database : MySql

c. Frontend

- Sử dụng : TSX, TailwindCSS, TypeScript, HTML
- Framework: NextJS

3.2. Triển khai AI server

3.2.1. Tập dữ liệu

Hiện tại, mô hình đang được huấn luyện trên bộ dữ liệu ETL Character Database, cụ thể là ETL-9G.

3.2.1.1. ETL Character Database

ETL Character Database là một tập hợp gồm khoảng 1,2 triệu hình ảnh các ký tự viết tay và in sẵn, bao gồm chữ số, ký hiệu, chữ cái Latin và ký tự tiếng Nhật. Dữ liệu này được phân chia thành 9 bộ riêng biệt từ ETL-1 đến ETL-9. Bộ dữ liệu được thu thập trong giai đoạn 1973–1984 bởi Phòng Thí nghiệm Điện kỹ thuật (nay thuộc Viện AIST) phối hợp với nhiều tổ chức nghiên cứu và các trường đại học tại Nhật Bản, nhằm phục vụ nghiên cứu về nhận dạng ký tự.

Cơ sở dữ liệu này được cung cấp miễn phí cho mục đích nghiên cứu, trước đây phân phối qua băng từ và CD-R, và từ tháng 4/2011 đã được cung cấp để tải trực tuyến. Từ tháng 1/2014, người dùng có thể truy cập và tải dữ liệu tại địa chỉ etlcdb.db.aist.go.jp.



Hình- 3-2: Chữ viết được trích từ tập dataset ETL Character Database

3.2.1.2. Bộ dữ liệu ETL-9G

a. Định dạng

- Mỗi bản ghi có độ dài cố định, không chứa chuỗi điều khiển.
- Dữ liệu được lưu ở dạng Big-endian
- 8 bit trên mỗi byte, 8199 byte trên mỗi bản ghi.

b. Định dạng tệp và mã mẫu

- Mỗi bộ dữ liệu gồm 3036 ký tự viết tay từ một người viết, tổng cộng có 12144 bản ghi (4 bộ ký tự × 3036 ký tự).
- Mỗi người viết đóng góp 20 tờ mẫu.

c. Tổ chức dữ liệu

Dữ liệu trong ETL-9 được chia thành các thư mục con, mỗi thư mục đại diện cho một ký tự Kanji. Mỗi thư mục chứa:

- 200 hình ảnh của ký tự đó.
- Một file .char.txt đi kèm chứa nhãn tương ứng.

Cách tổ chức này giúp việc gán nhãn, quản lý dữ liệu và chuẩn bị dữ liệu huấn luyện trở nên hiệu quả và dễ dàng hơn.

d. Thông tin tệp dữ liệu

Bảng- 3-1: Thông tin tệp dữ liệu trong ETL-9G

Tên tệp	Số bản ghi	Số loại ký tự	Số tập dữ liệu	Chỉ số tập dữ liệu	Số tờ
ETL9G_01	12144	3036	4	1-4	80

ETL9G_02	12144	3036	4	5-8	80
⋮	⋮	⋮	⋮	⋮	⋮
ETL9G_50	12144	3036	4	197-200	80

3.2.2. Tiền xử lý dữ liệu

Quá trình tiền xử lý hình ảnh được thực hiện nhằm chuẩn hóa và nâng cao chất lượng ảnh trước khi đưa vào huấn luyện mô hình nhận dạng ký tự. Dữ liệu sử dụng là từ tập ETL-9G thuộc ETL Character Database. Các bước tiền xử lý bao gồm:

3.2.2.1. Chuẩn Hóa Kích Thước Ảnh.

Ban đầu, ảnh được chuyển sang thang độ xám nhằm giảm độ phức tạp và tập trung vào phần hình dạng của ký tự. Sau đó, tất cả các ảnh được đưa về kích thước cố định 96x96 pixel để đảm bảo tính nhất quán trong toàn bộ tập dữ liệu, giúp mô hình dễ dàng tiếp nhận và xử lý.

3.2.2.2. Bộ Lọc Bilateral.

Bộ lọc Bilateral là một bộ lọc phi tuyến, được thiết kế để làm mịn và làm mờ ảnh trong khi vẫn giữ nguyên các đường biên rõ nét. Giá trị độ sáng của một điểm ảnh trung tâm (pixel được xét) trong một vùng cấu trúc được tính bằng cách áp dụng trung bình có trọng số của độ sáng của các pixel nằm trong vùng lân cận của pixel được xét. Trọng số phụ thuộc vào khoảng cách và sự khác biệt về độ sáng giữa pixel trung tâm và các pixel xung quanh. Quy trình lọc được biểu diễn như một phép tích chập của hàm đầu vào với hàm mặt nạ, được mô tả bằng công thức:

$$f(x_0) = \frac{1}{k} \sum_{i \in R} f(x_i) \cdot h_D(x_0 - x_i) \cdot h_I(x_0 - x_i)$$

Trong đó:

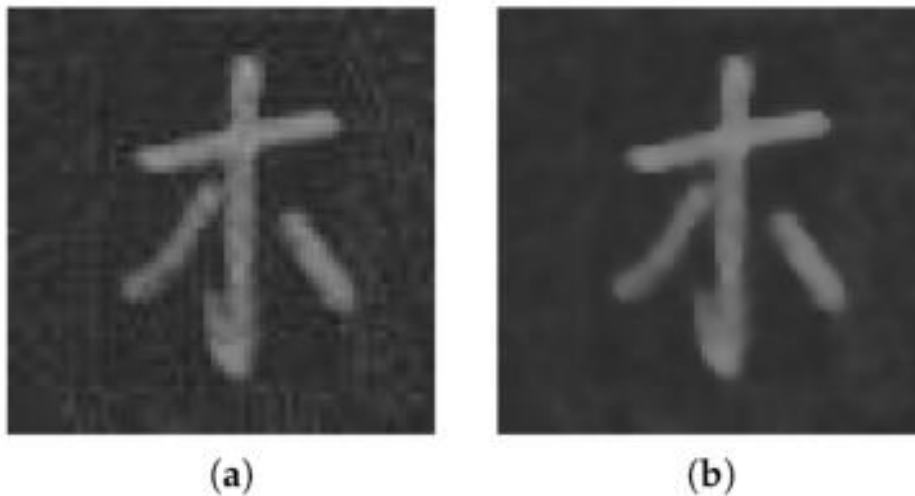
- $f(x_0)$: Giá trị tại điểm trung tâm sau khi lọc.
- k : Hệ số chuẩn hóa.
- R : Vùng lân cận của điểm x_0 .
- $H_D(x_0-x_i)$: Hàm khoảng cách.
- $h_I(x_0-x_i)$: Hàm độ sáng.

Hệ số chuẩn hóa k đảm bảo rằng tổng các trọng số được chuẩn hóa thành 1, được xác định như sau:

$$k = \sum_{i \in R} h_D(x_0 - x_i) \cdot h_I(x_0 - x_i)$$

- $h_D(x_0 - x_i)$: Hàm khoảng cách, đảm bảo các pixel gần điểm trung tâm được ưu tiên hơn trong quá trình lọc.
- $h_I(x_0 - x_i)$: Hàm cường độ, ưu tiên các pixel có độ sáng tương đồng với điểm trung tâm để bảo toàn các chi tiết và đường biên.

Hiệu quả của bộ lọc Bilateral



Hình- 3-3: Minh họa hiệu quả bộ lọc Bilateral

Hình trên minh họa hiệu quả của bộ lọc Bilateral, với ảnh gốc (a) và ảnh sau khi áp dụng bộ lọc Bilateral (b). Sau khi áp dụng, ảnh được làm mịn đáng kể, nhưng các đường biên vẫn được giữ nguyên rõ ràng. Điều này làm cho bộ lọc Bilateral trở thành một công cụ mạnh mẽ trong việc loại bỏ nhiễu mà không làm mất chi tiết quan trọng của ảnh.

3.2.2.3. Phân Ngưỡng Thích Nghi

Sau khi lọc, ảnh được chuyển thành nhị phân (đen trắng) sử dụng phương pháp Adaptive Gaussian Thresholding. Kỹ thuật này thiết lập ngưỡng dựa trên giá trị cục bộ của từng vùng ảnh, giúp làm nổi bật các ký tự trên nền, từ đó nâng cao độ tương phản và khả năng phân biệt ký tự.

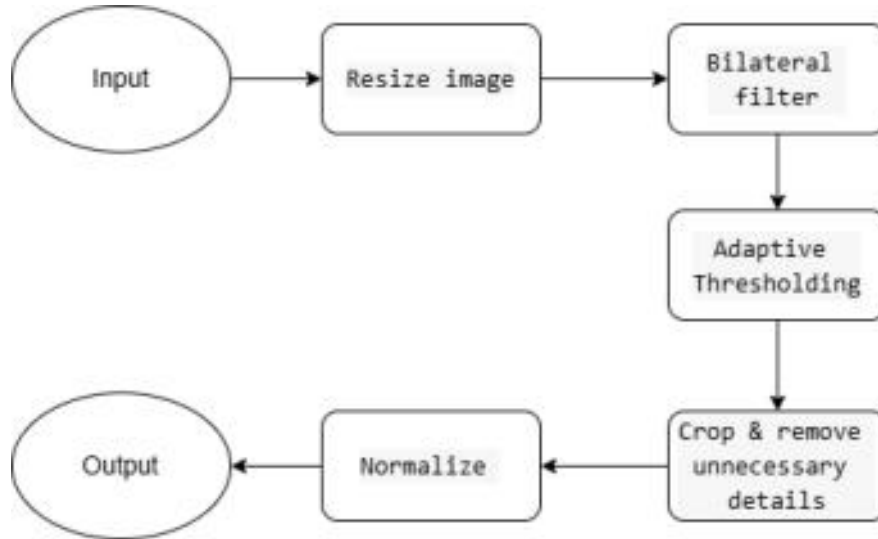
3.2.2.4. Cắt và Thêm Viền

Một phần nhỏ bên trái của ảnh (khoảng 10%) được cắt bỏ để loại bỏ các vùng nhiễu không cần thiết hoặc phần thừa trong ảnh. Sau đó, viền đen dày 10 pixel được thêm vào

xung quanh ảnh. Viền này giúp định hình rõ ràng vùng ký tự, cải thiện khả năng nhận diện của mô hình.

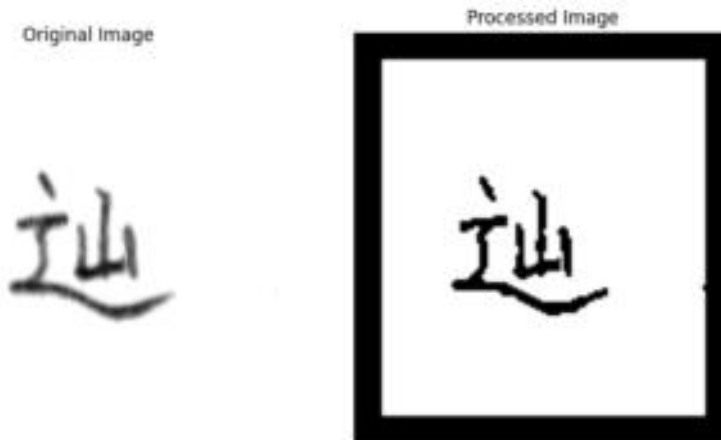
3.2.2.5. Chuẩn hóa dữ liệu ảnh

Cuối cùng, ảnh được chuẩn hóa bằng phương pháp min-max để đảm bảo giá trị pixel nằm trong khoảng từ 0 đến 255. Điều này giúp dữ liệu đầu vào nhất quán, ổn định và dễ dàng xử lý hơn trong quá trình huấn luyện.



Hình- 3-4: Sơ đồ khối các bước tiền xử lý dữ liệu

Kết quả sau khi xử lý ảnh



Hình- 3-5: Kết quả sau khi xử lý ảnh

3.2.3. Quy trình huấn luyện

3.2.3.1. Chia bộ dữ liệu

Trong quá trình huấn luyện mô hình học máy, việc chia dữ liệu thành các bộ con khác nhau là một bước quan trọng để đảm bảo tính hiệu quả và khả năng tổng quát của mô hình. Để tránh tình trạng overfitting và giúp mô hình có thể đánh giá được trên những dữ liệu chưa từng thấy trước đó, dữ liệu sẽ được phân chia thành ba phần: bộ huấn luyện (Training Set), bộ xác thực (Validation Set) và bộ kiểm tra (Test Set). Cách chia này giúp mô hình được huấn luyện tốt, đánh giá liên tục và cuối cùng kiểm tra hiệu suất trên dữ liệu hoàn toàn mới.

Tổng số dữ liệu: Bộ dữ liệu gốc chứa tổng cộng 607,200 hình ảnh. Đây là một tập dữ liệu lớn, giúp mô hình có đủ thông tin để học và khái quát hóa các đặc điểm của dữ liệu một cách chính xác.

Bộ huấn luyện (Training Set): Bộ huấn luyện chiếm 63% của tổng dữ liệu. Đây là bộ dữ liệu chính được sử dụng để huấn luyện mô hình. Trong quá trình huấn luyện, mô hình sẽ tiếp nhận các hình ảnh trong bộ huấn luyện để học cách phân loại các đối tượng, đồng thời cập nhật trọng số dựa trên các tín hiệu lỗi mà nó nhận được sau mỗi lần dự đoán. Việc sử dụng một bộ huấn luyện lớn giúp mô hình có thể học được các đặc trưng phức tạp và cải thiện khả năng tổng quát của mình.

Bộ xác thực (Validation Set): Bộ xác thực chiếm 25% của tổng dữ liệu. Bộ xác thực không được sử dụng trong quá trình huấn luyện trực tiếp mà chỉ được dùng để kiểm tra mô hình sau mỗi epoch. Việc đánh giá mô hình trên bộ xác thực giúp theo dõi được sự tiến triển của mô hình trong quá trình huấn luyện, từ đó điều chỉnh các tham số, tránh việc mô hình bị overfitting vào dữ liệu huấn luyện. Đồng thời, bộ xác thực còn giúp kiểm soát việc điều chỉnh quá mức các tham số trong quá trình học, làm cho mô hình không chỉ phù hợp với dữ liệu huấn luyện mà còn có thể hoạt động tốt với các dữ liệu chưa thấy.

Bộ kiểm tra (Test Set): Bộ kiểm tra chiếm 5% của tổng dữ liệu, tương đương với 30,360 hình ảnh. Đây là bộ dữ liệu cuối cùng dùng để kiểm tra hiệu suất của mô hình sau khi quá trình huấn luyện và tối ưu hóa đã hoàn thành. Bộ kiểm tra giúp đánh giá độ chính xác của mô hình trên dữ liệu chưa từng thấy trước đó và là thước đo khách quan nhất về khả năng tổng quát hóa của mô hình. Một mô hình có hiệu suất tốt trên bộ kiểm tra sẽ chứng tỏ rằng nó có thể áp dụng thành công vào các tình huống thực tế với dữ liệu mới.

Việc chia dữ liệu theo tỷ lệ này giúp đảm bảo rằng mô hình được huấn luyện một cách toàn diện và có khả năng phân loại chính xác, đồng thời giảm thiểu rủi ro của việc mô hình chỉ học thuộc các đặc điểm của dữ liệu huấn luyện mà không thể áp dụng vào các dữ liệu mới. Quá trình này không chỉ giúp cải thiện độ chính xác mà còn giúp mô hình trở nên ổn định và linh hoạt hơn khi triển khai thực tế.

3.2.3.2. Chuẩn bị mô hình

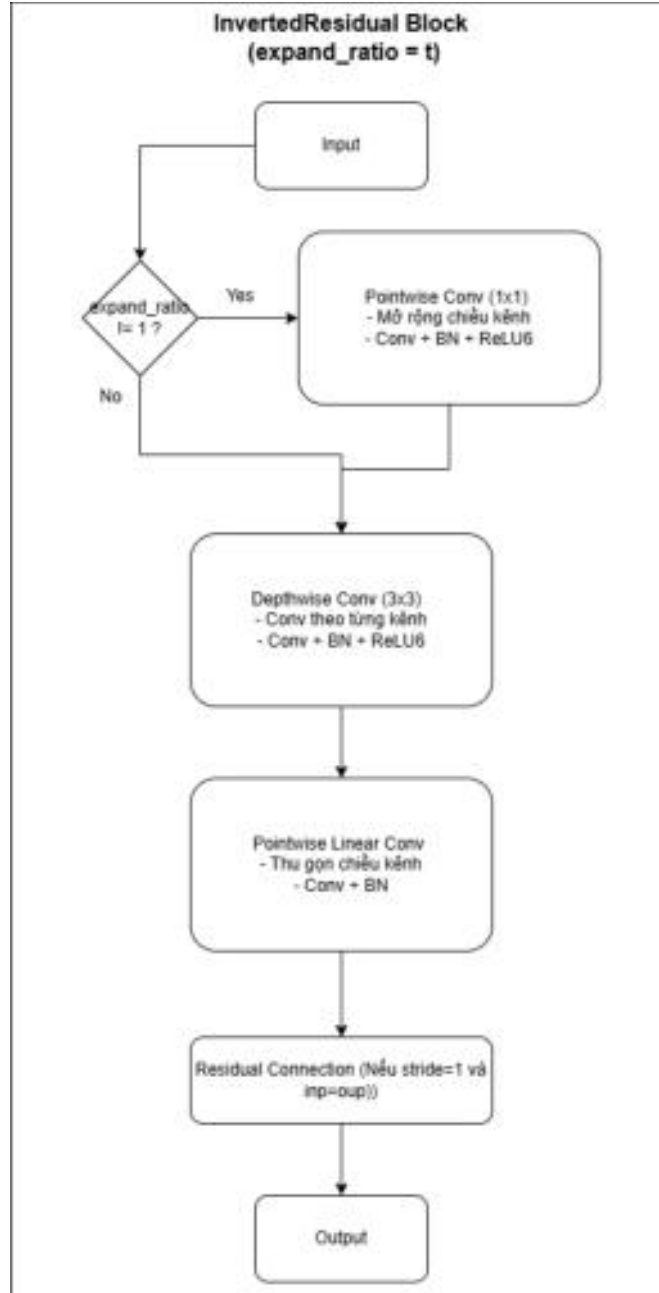
Mô hình được xây dựng dựa trên kiến trúc MobileNetV2, một mạng nơ-ron tích chập hiệu quả, thường được sử dụng trong các bài toán phân loại ảnh với yêu cầu cao về tốc độ và tài nguyên. Mạng đầu vào nhận ảnh đơn kênh và đầu ra là xác suất dự đoán trên 3036 lớp.

a. Khối Inverted Residual

Cấu trúc lõi của mô hình là khối Inverted Residual, bao gồm ba giai đoạn chính:

- Mở rộng kênh (Expansion): Nếu hệ số mở rộng (`expand_ratio`) lớn hơn 1, đầu vào được chuyển đổi thông qua lớp tích chập 1×1 để tăng số kênh, sau đó áp dụng chuẩn hóa batch và hàm kích hoạt ReLU6.
- Tích chập theo chiều sâu (Depthwise Convolution): Mỗi kênh được xử lý riêng biệt bằng lớp tích chập 3×3 với số nhóm bằng số kênh, duy trì tính toán nhẹ nhưng vẫn giữ được đặc trưng không gian.
- Thu hẹp kênh (Projection): Một lớp tích chập 1×1 khác dùng để đưa số kênh về đầu ra mong muốn, đi kèm chuẩn hóa batch. Không sử dụng hàm kích hoạt ở giai đoạn này để giữ tính tuyến tính (linear bottleneck).

Khi số kênh đầu vào và đầu ra giống nhau và `stride = 1`, khối này sử dụng kết nối tắt (residual connection) để tăng khả năng học đặc trưng sâu mà không làm mất thông tin ban đầu.



Hình- 3-6: Sơ đồ khối minh họa Inverted Residual Block

b. Kiến trúc tổng thể

Mô hình bao gồm ba phần chính:

- Lớp đầu vào: Áp dụng một lớp tích chập 3×3 với $\text{stride} = 2$ để giảm độ phân giải ảnh đầu vào và tăng số kênh lên $\text{initial_c} = 32 \times \text{width_mult}$, kèm theo chuẩn hóa batch và ReLU6.

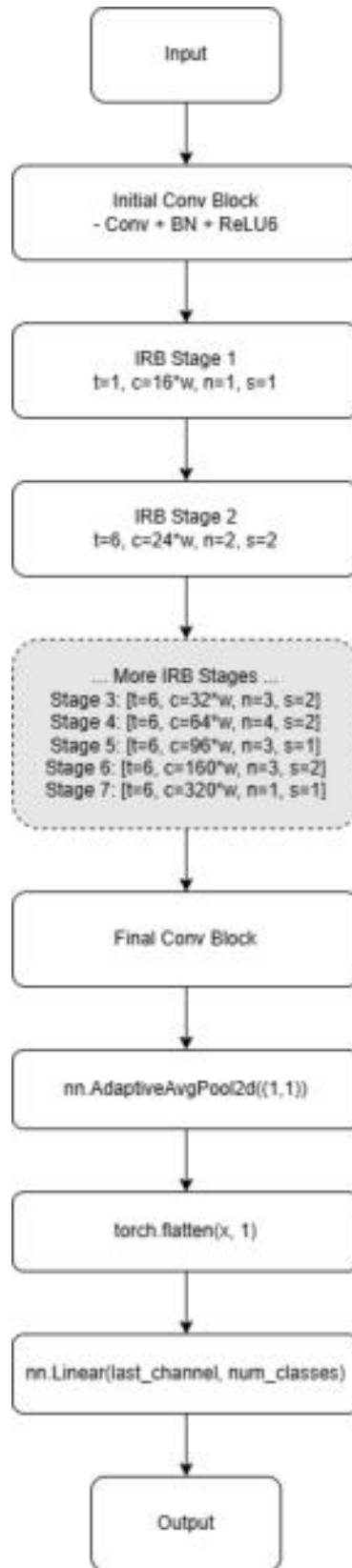
- Chuỗi khối Inverted Residual: Bao gồm nhiều nhóm khối được cấu hình theo bảng sau, trong đó t là hệ số mở rộng, c là số kênh đầu ra, n là số lần lặp lại khối, và s là stride của khối đầu tiên:

Bảng- 3-2: Bảng cấu hình các khối Inverted Residual

Hệ số mở rộng t	Số kênh đầu ra c	Số lần lặp n	Stride s
1	16	1	1
6	24	2	2
6	32	3	2
6	64	4	2
6	96	3	1
6	160	3	2
6	320	1	1

Mỗi khối tiếp theo được nối tiếp với khối trước đó, với số kênh được điều chỉnh bởi hệ số `width_mult`.

- Lớp kết thúc và phân loại: Sau các khối residual, áp dụng một lớp tích chập 1×1 để nâng số kênh lên `last_channel = 1280`, tiếp theo là chuẩn hóa và ReLU6. Sau đó, sử dụng pooling toàn cục (`AdaptiveAvgPool2d`) để đưa dữ liệu về kích thước cố định trước khi qua lớp phân loại (`Linear`) với đầu ra có số chiều bằng số lớp.



Hình- 3-7: Sơ đồ khối kiến trúc MobileNetV2

3.2.3.3. *Huấn luyện mô hình*

Mục tiêu: Mục tiêu chính trong phần này là huấn luyện một mô hình MobileNet có khả năng phân loại ảnh vào các lớp tương ứng với độ chính xác cao trên tập kiểm tra. Đồng thời, mô hình cần được tối ưu hóa để sử dụng tài nguyên phần cứng hiệu quả, bao gồm GPU, CPU và bộ nhớ RAM. Một yêu cầu quan trọng nữa là mô hình phải đạt được tốc độ suy luận nhanh, đủ khả năng triển khai trên các thiết bị thực tế như điện thoại di động hoặc các hệ thống nhúng.

Dữ liệu: Dữ liệu sử dụng trong quá trình huấn luyện là bộ dữ liệu ETL-9G, thuộc cơ sở dữ liệu ETL Character Database. Bộ dữ liệu này chứa các hình ảnh chữ viết tay từ các nguồn khác nhau, mỗi hình ảnh có một nhãn tương ứng với chữ cái hoặc số trong bộ ký tự.

Kích thước ảnh: Tất cả các hình ảnh trong bộ dữ liệu đều được resize về kích thước 96x96 pixel. Việc thay đổi kích thước giúp chuẩn hóa dữ liệu đầu vào và giảm tải cho hệ thống trong quá trình huấn luyện.

Số epoch: Quá trình huấn luyện được thực hiện qua 11 epoch. Mỗi epoch là một lần toàn bộ dữ liệu huấn luyện được đưa qua mô hình để cập nhật trọng số. Thời gian huấn luyện phụ thuộc vào kích thước của dữ liệu và số lượng epoch.

Batch size: Sử dụng batch size là 32, tức là mỗi lần cập nhật trọng số của mô hình, 32 mẫu dữ liệu sẽ được đưa vào mô hình và tính toán đồng thời. Batch size này giúp cân bằng giữa tốc độ huấn luyện và hiệu quả của việc học.

Framework: Mô hình được xây dựng và huấn luyện sử dụng framework PyTorch, một trong những thư viện phổ biến và mạnh mẽ trong lĩnh vực học sâu, hỗ trợ việc tối ưu hóa mô hình và triển khai hiệu quả.

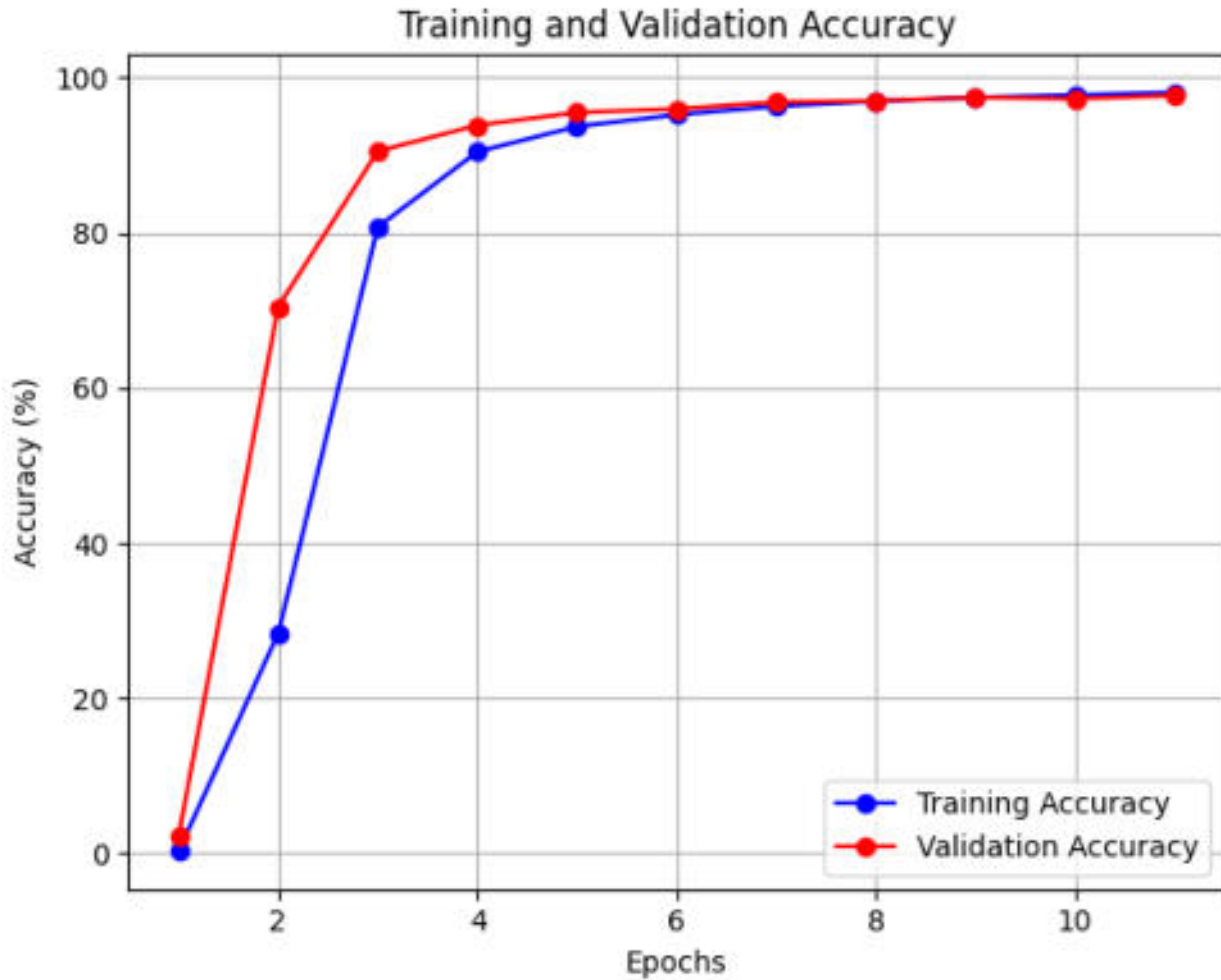
Môi trường huấn luyện: Quá trình huấn luyện được thực hiện trên môi trường với GPU NVIDIA T4, có sẵn trong môi trường Kaggle, giúp tăng tốc độ tính toán và huấn luyện mô hình. Hệ điều hành được sử dụng là Windows, đảm bảo sự tương thích với các công cụ phần mềm liên quan.

Vòng lặp Epoch: Trong mỗi epoch, mô hình sẽ tiếp nhận dữ liệu huấn luyện, thực hiện quá trình tính toán và cập nhật trọng số thông qua quá trình tối ưu hóa (optimizer). Quá trình này lặp lại cho đến khi mô hình đạt được sự cải thiện về độ chính xác.

Biểu đồ Loss và Accuracy: Trong suốt quá trình huấn luyện, hai chỉ số quan trọng là loss và accuracy sẽ được theo dõi liên tục. Loss đại diện cho độ sai lệch giữa dự đoán của mô hình và nhãn thực tế, trong khi accuracy cho biết tỷ lệ chính xác của mô hình. Việc theo dõi những chỉ số này giúp đảm bảo rằng mô hình đang học hiệu quả qua từng epoch.

Checkpoint: Mô hình sẽ lưu lại checkpoint sau mỗi epoch. Điều này giúp bảo vệ quá trình huấn luyện bằng cách ghi lại trạng thái của mô hình tại mỗi thời điểm, từ đó có thể phục hồi lại mô hình nếu gặp sự cố hoặc tiếp tục huấn luyện từ điểm đã lưu.

3.2.4. Kết quả huấn luyện



Hình- 3-8: Đồ thị train accuracy



Hình- 3-9: Đồ thị train loss

Bảng- 3-3: Kết quả đánh giá mô hình trên tập test

Chỉ số đánh giá	Giá trị
Tổng số mẫu (Total samples)	30,360
Số mẫu đúng (Correct)	27,695
Độ chính xác (Accuracy)	91.22%

3.2.5. Triển khai mô hình bằng Flask

Sau quá trình huấn luyện, mô hình được sử dụng để xây dựng một API phục vụ dự đoán. Flask được lựa chọn làm framework để triển khai máy chủ do tính đơn giản, nhẹ và

để tích hợp với model. Trong quá trình hỗ trợ người học luyện viết chữ Kanji, hệ thống triển khai hai chức năng cốt lõi giúp tăng cường trải nghiệm tương tác và cải thiện độ chính xác khi viết.

3.2.5.1. Dự đoán chữ Kanji từ ảnh viết tay

Thứ nhất, hệ thống cho phép người dùng gửi ảnh chứa chữ Kanji viết tay để thực hiện nhận dạng. Sau khi tiếp nhận dữ liệu hình ảnh, mô-đun AI sẽ tiến hành phân tích nội dung bằng cách trích xuất đặc trưng từ ảnh và so sánh với tập dữ liệu chữ mẫu đã được huấn luyện trước đó. Kết quả đầu ra là danh sách các ký tự Kanji được mô hình xác định là gần giống nhất với ảnh người dùng cung cấp, đi kèm với mức độ tin cậy của từng dự đoán. Điều này giúp người học không chỉ kiểm tra được mức độ chính xác của nét viết mà còn nhận diện được chữ viết tay của mình đang gần giống với ký tự nào, từ đó điều chỉnh để cải thiện.

3.2.5.2. Chấm điểm chữ viết tay

Thứ hai, nhằm hỗ trợ người học kiểm tra độ chính xác khi luyện viết một ký tự cụ thể, hệ thống cung cấp chức năng đánh giá mức độ khớp giữa chữ viết tay và ký tự Kanji mục tiêu do người dùng cung cấp. Cơ chế hoạt động của chức năng này không chỉ dừng lại ở việc xác định xem mô hình có đoán đúng hay không, mà còn tính đến xác suất dự đoán tương đối giữa các ký tự.

Sau khi nhận được ảnh viết tay và nhãn ký tự mục tiêu, hệ thống sẽ xử lý ảnh để đưa vào mô hình nhận dạng. Mô hình sau đó sinh ra một tập xác suất (probability distribution) ứng với tất cả các ký tự có trong tập dữ liệu huấn luyện. Dựa trên kết quả này, hệ thống xác định:

- Xác suất của nhãn đúng (p_{true}) – là xác suất mà mô hình gán cho ký tự mục tiêu mà người học khai báo.
- Xác suất của nhãn được dự đoán cao nhất (p_{pred}) – là xác suất tương ứng với ký tự mà mô hình cho là đúng nhất trong ảnh viết tay đã gửi.

Từ hai xác suất này, hệ thống tính điểm theo công thức:

$$score = \max \left(0.05, \sqrt{\frac{p_{true}}{p_{pred} + \epsilon}} \right)$$

Trong đó:

- p_{true} là xác suất của nhãn người học cung cấp.
- p_{pred} là xác suất của nhãn mà mô hình dự đoán là cao nhất.

- ϵ là một hằng số rất nhỏ (thường là 10^{-6}) để tránh chia cho 0.
- Hàm căn bậc hai được dùng để "làm mềm" tỉ lệ giữa hai xác suất, giúp điểm số không bị rơi xuống quá thấp khi có chênh lệch lớn.
- Hàm max đảm bảo rằng điểm số luôn lớn hơn hoặc bằng một ngưỡng tối thiểu (ở đây là 0.05), tránh trường hợp người học bị đánh giá quá thấp.

Trong trường hợp mô hình dự đoán chính xác ký tự mục tiêu, điểm số được gán mặc định ở mức cao (0.95), phản ánh độ khớp tối ưu.

Cách tiếp cận này không chỉ giúp đánh giá đúng chữ viết tay mà còn đưa ra phản hồi định lượng rõ ràng về mức độ chính xác, từ đó hỗ trợ người học tự điều chỉnh và cải thiện kỹ năng viết Kanji một cách hiệu quả hơn.

3.3. Triển khai hệ thống lên môi trường thực tế

3.3.1. Công nghệ triển khai sử dụng

Docker: Đây là công nghệ container hóa cho phép đóng gói toàn bộ ứng dụng và các thành phần phụ thuộc vào trong một đơn vị độc lập, gọi là container. Việc sử dụng Docker giúp đảm bảo rằng ứng dụng có thể chạy đồng nhất trên nhiều môi trường khác nhau, từ máy tính cá nhân cho đến máy chủ thực tế. Docker cũng giúp giảm thiểu lỗi cấu hình giữa các môi trường, hỗ trợ kiểm soát phiên bản môi trường một cách chặt chẽ. Ngoài ra, quá trình triển khai, mở rộng hay rollback hệ thống trở nên đơn giản hơn rất nhiều so với triển khai truyền thống.

Docker Compose: Là công cụ hỗ trợ định nghĩa và quản lý nhiều container Docker cùng một lúc thông qua một tệp cấu hình duy nhất (docker-compose.yml). Nhờ đó, các dịch vụ như AI server, web server, và client frontend có thể được khởi tạo và kết nối dễ dàng chỉ với một lệnh duy nhất. Điều này giúp tăng năng suất phát triển và giảm thiểu sai sót khi triển khai thủ công từng thành phần.

Azure Virtual Machines: Được sử dụng làm hạ tầng máy chủ chính để triển khai toàn bộ hệ thống. Azure VM cung cấp khả năng mở rộng linh hoạt, hiệu năng cao và hỗ trợ tích hợp với Docker. Ngoài ra, Azure cũng hỗ trợ cấp phát địa chỉ IP công khai (Public IP) và DNS, cho phép người dùng từ bên ngoài truy cập dễ dàng vào hệ thống sau khi triển khai.

3.3.2. Các bước triển khai toàn hệ thống

- Chuẩn bị mã nguồn và cấu hình Docker ở môi trường phát triển cục bộ. Mỗi thành phần như AI server, web server và client đều được đóng gói bằng Dockerfile. Tệp docker-compose.yml được dùng để định nghĩa các dịch vụ và thiết lập kết nối giữa chúng.

- Quản lý toàn bộ mã nguồn trên GitHub để thuận tiện cho việc đồng bộ và triển khai từ xa.
- Tạo máy ảo Ubuntu trên nền tảng Azure, thiết lập địa chỉ IP công khai và cấu hình DNS để đảm bảo khả năng truy cập hệ thống từ Internet.
- Kết nối đến máy ảo qua SSH để thực hiện cài đặt và triển khai.
- Cài đặt Docker và Docker Compose trên máy ảo để có thể build và quản lý các container.
- Clone mã nguồn từ GitHub về máy ảo bằng lệnh git clone.
- Tạo tệp .env và khai báo các biến môi trường như thông tin kết nối cơ sở dữ liệu, secret key, đường dẫn model AI, cổng truy cập, v.v. Tệp .env giúp tách biệt thông tin cấu hình ra khỏi mã nguồn, thuận tiện cho việc thay đổi và bảo mật.
- Khởi chạy toàn bộ hệ thống bằng lệnh docker compose up --build. Các dịch vụ sẽ được dựng và kết nối tự động trong môi trường Docker.
- Mở các cổng cần thiết để hệ thống có thể được truy cập từ trình duyệt hoặc thiết bị bên ngoài.

3.4. Kết quả

3.4.1. Giao diện học viên

a. Trang chủ

Trang chủ cung cấp cái nhìn tổng quan và lối vào nhanh đến các chức năng chính của hệ thống học tiếng Nhật.

Return

Search for courses

Search KANJI Translate Check grammar Category   Sign in

A new way to learn

未来に向けて、知識を創造する旅へ

A fresh learning platform designed to revolutionize your learning experience, deepen understanding, and enhance skills.

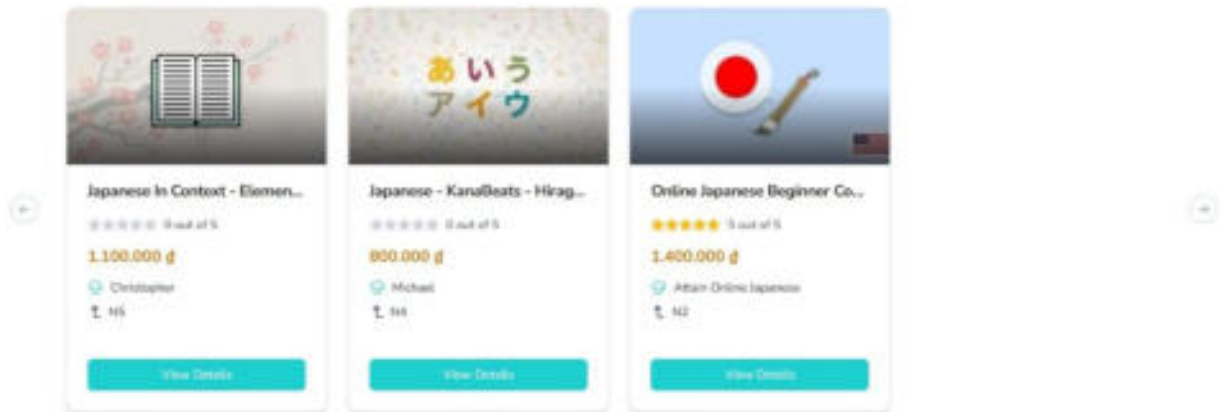
Get Started



Categories



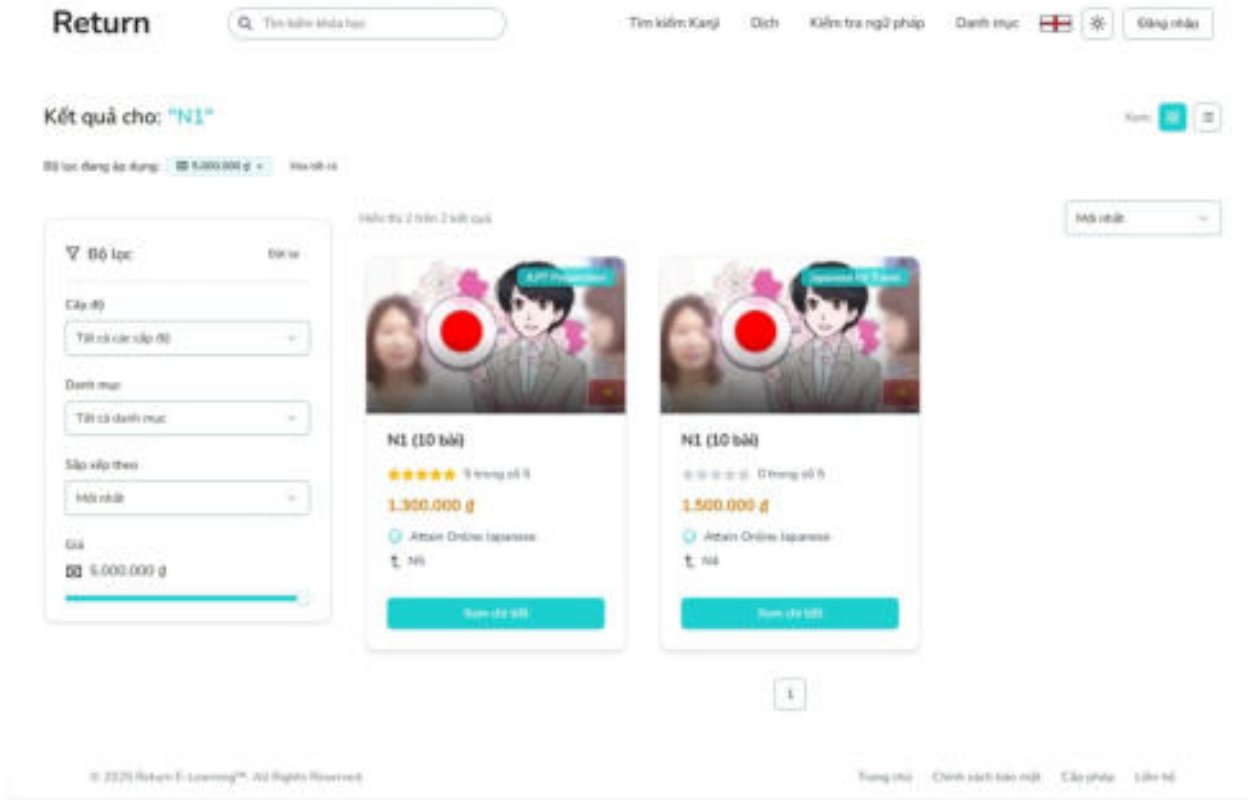
Japanese Language Basics



Hình- 3-10: Giao diện trang chủ

b. Tìm kiếm khóa học

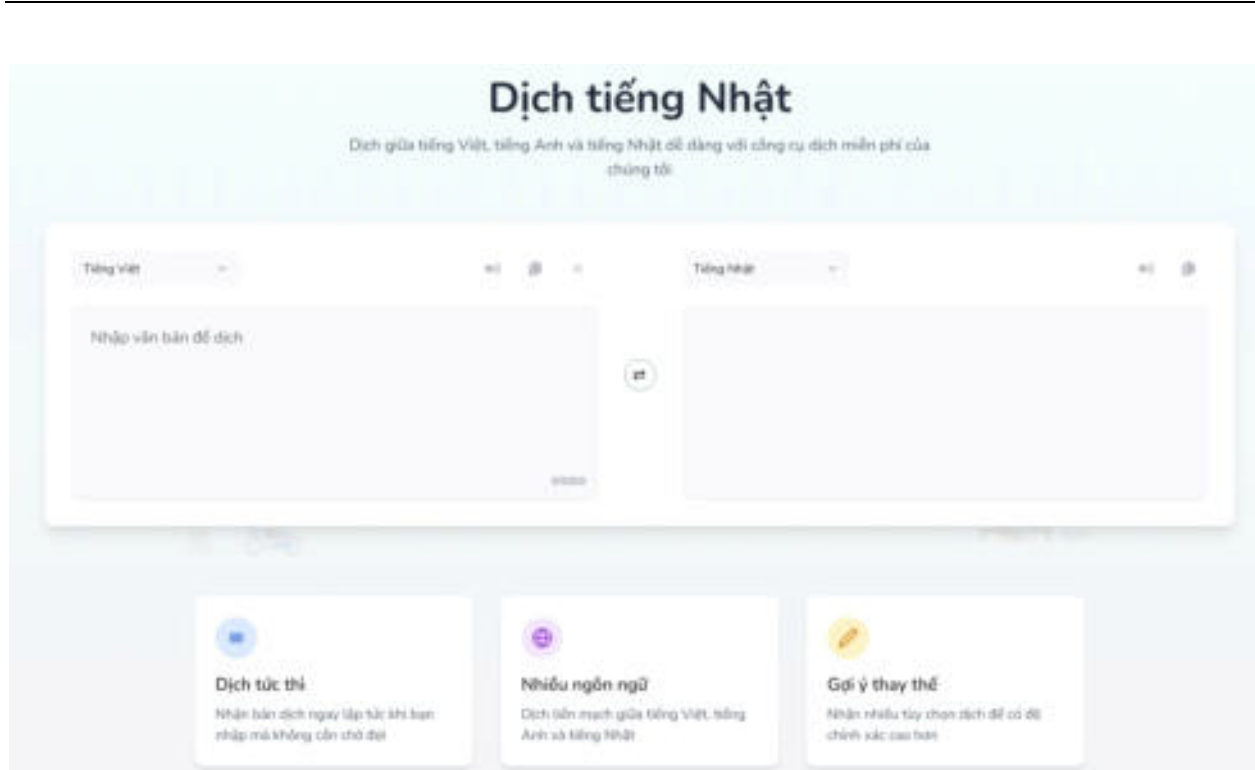
Chức năng tìm kiếm khóa học giúp người dùng nhanh chóng tìm được khóa học phù hợp theo từ khóa hoặc lọc theo danh mục, cấp độ mong muốn.



Hình- 3-11: Giao diện tìm kiếm khóa học

c. Dịch

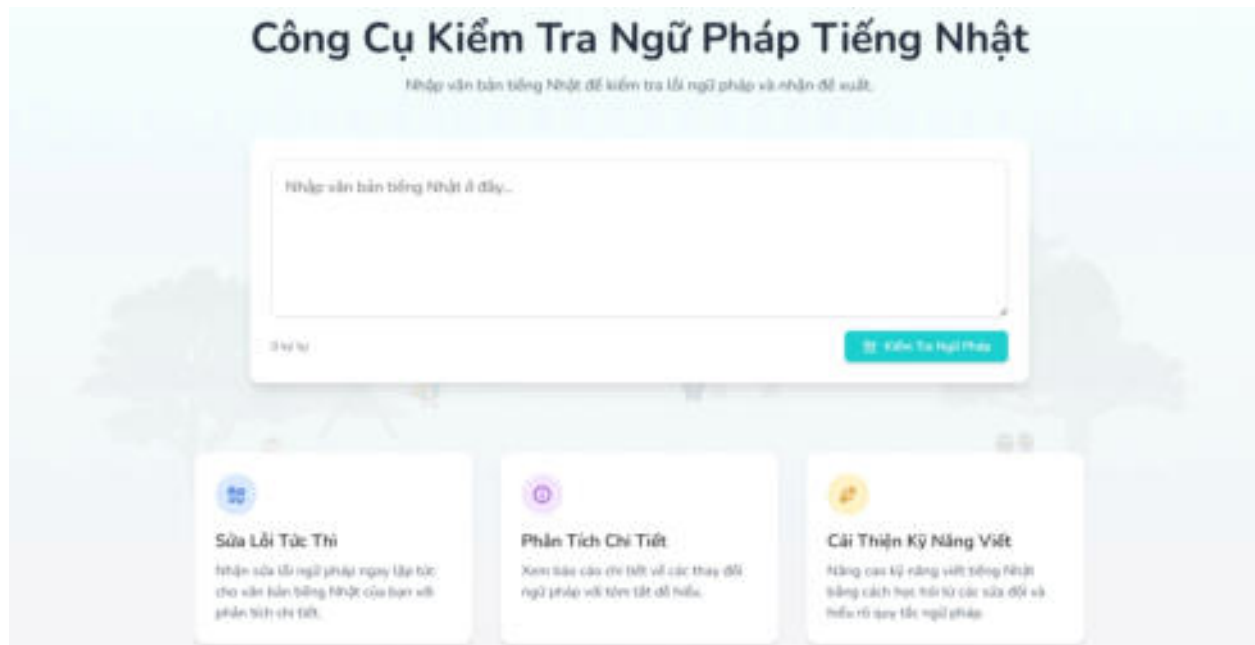
Chức năng dịch hỗ trợ người học tra nghĩa từ vựng tiếng Nhật một cách nhanh chóng và chính xác.



Hình- 3-12: Giao diện dịch

d. Kiểm Tra Ngữ Pháp Tiếng Nhật

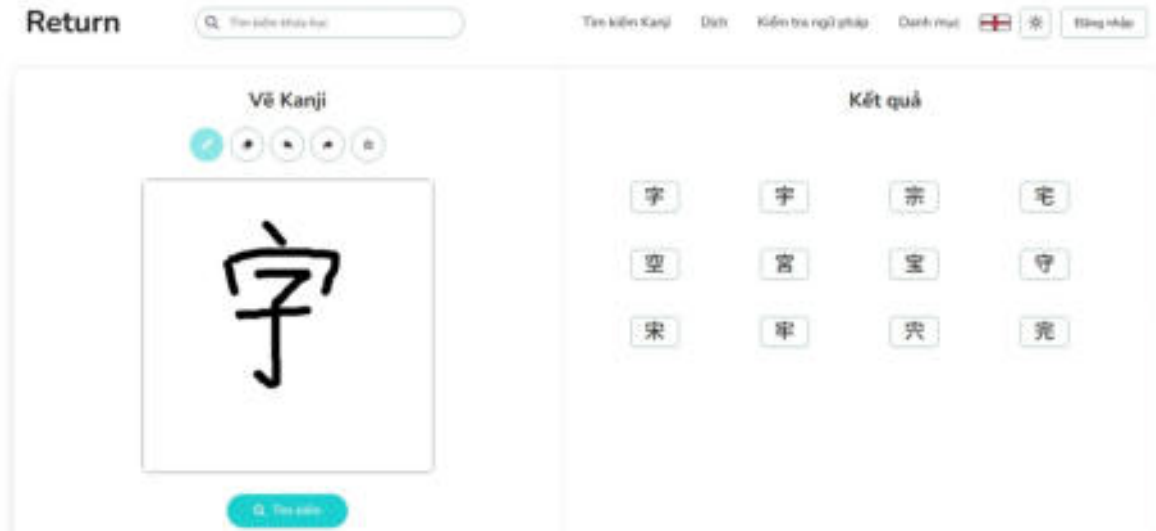
Chức năng Kiểm tra ngữ pháp tiếng Nhật giúp người học phát hiện và sửa lỗi trong câu tiếng Nhật một cách hiệu quả.



Hình- 3-13: Giao diện kiểm tra ngữ pháp

e. Tra cứu chữ Kanji

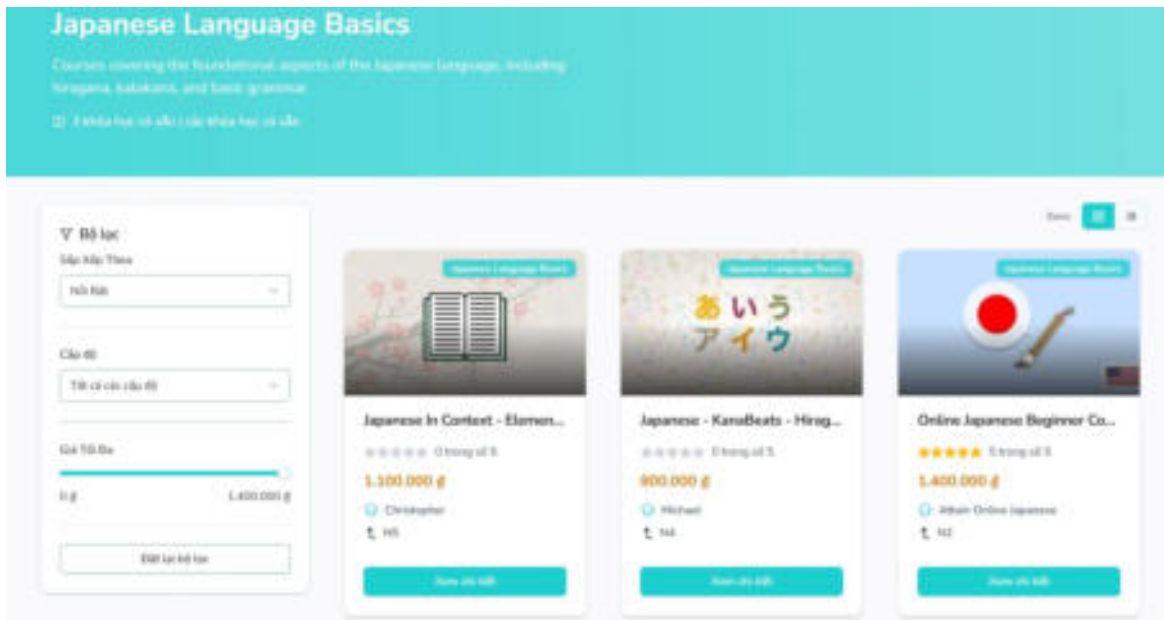
Chức năng Tra cứu chữ Kanji cho phép nhận diện và tìm kiếm chữ Kanji từ hình ảnh viết tay, hỗ trợ quá trình học viết và ghi nhớ Kanji.



Hình- 3-14: Giao diện tra cứu chữ Kanji

f. Xem khóa học theo danh mục

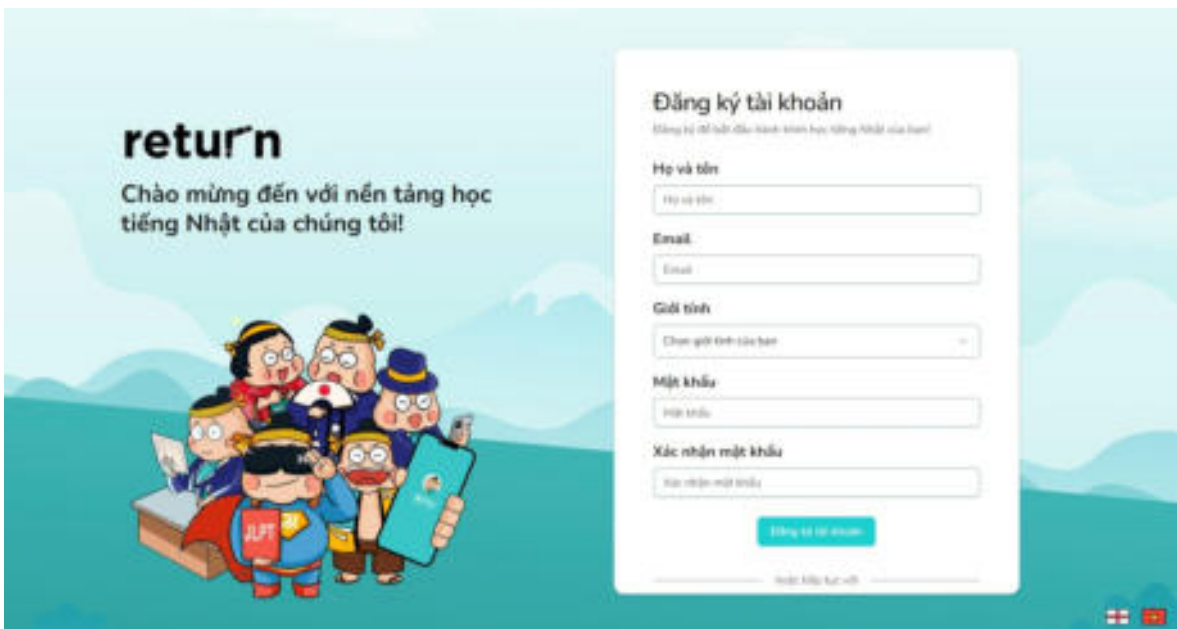
Chức năng xem khóa học theo danh mục cho phép người dùng dễ dàng lọc các khóa học theo cấp độ và mức giá phù hợp với nhu cầu học tập.



Hình- 3-15: Giao diện xem khóa học theo danh mục

g. Đăng kí

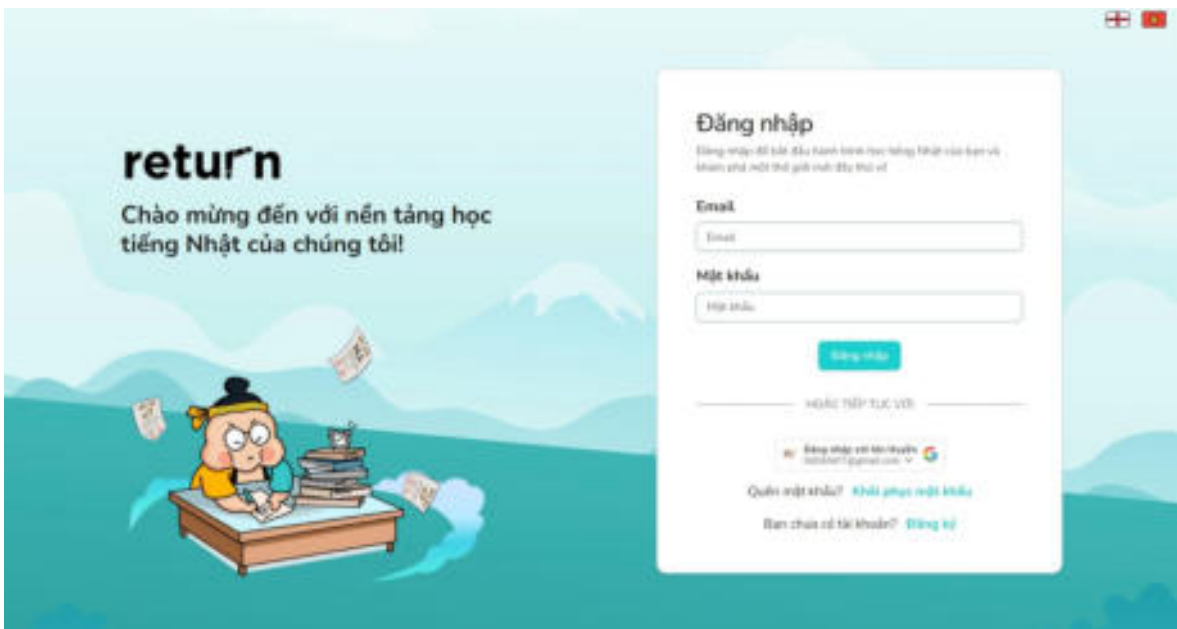
Cho phép người dùng tạo tài khoản mới để bắt đầu sử dụng hệ thống.



Hình- 3-16: Giao diện đăng kí

h. Đăng nhập

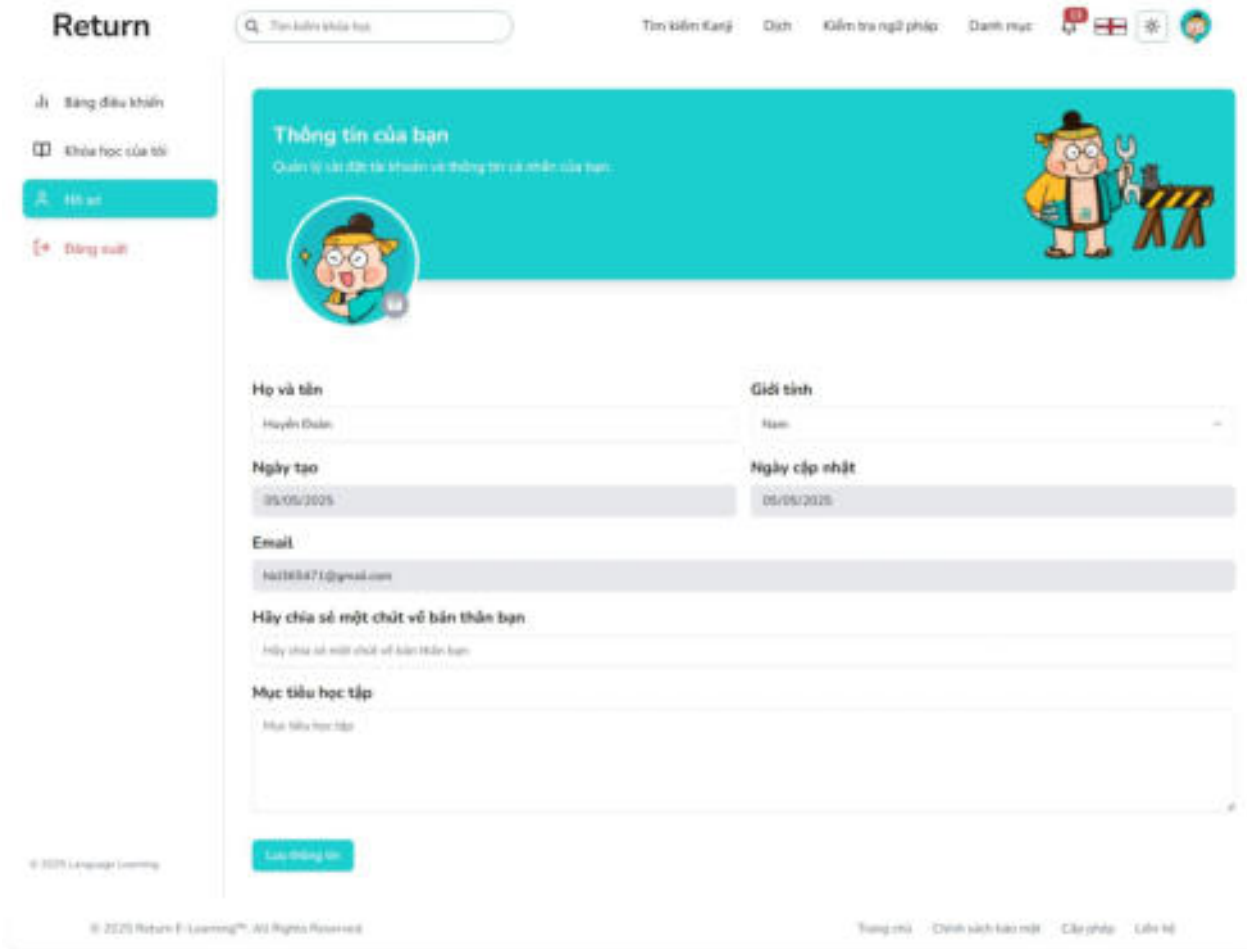
Hỗ trợ người dùng truy cập vào hệ thống bằng thông tin tài khoản đã đăng ký.



Hình- 3-17: Giao diện đăng nhập

i. Thông tin cá nhân

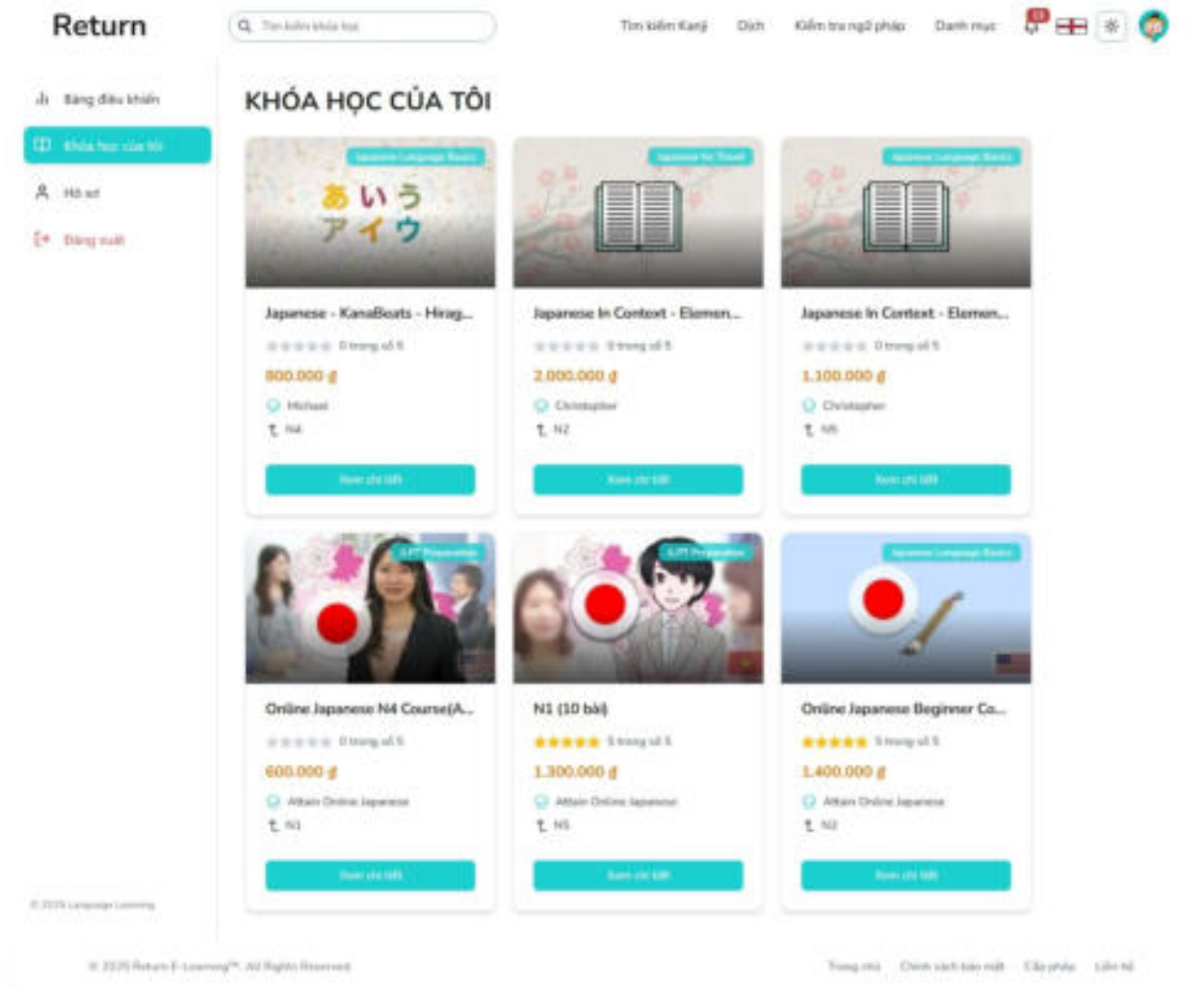
Chức năng thông tin cá nhân cho phép người dùng xem và chỉnh sửa các dữ liệu cá nhân như tên, email và ảnh đại diện.



Hình- 3-18: Giao diện thông tin cá nhân

j. Khóa học đã tham gia

Chức năng khóa học đã tham gia giúp người dùng theo dõi danh sách các khóa học mà mình đang học hoặc đã đăng ký trước đó.



Hình- 3-19: Giao diện khóa học đã tham gia

k. Chi tiết khóa học

Chức năng chi tiết khóa học cung cấp đầy đủ thông tin về nội dung, mục tiêu, cấp độ, đánh giá và giáo viên giảng dạy, giúp người học đưa ra lựa chọn phù hợp.

The screenshot displays the course page for 'Japanese In Context - Elementary Japanese Course' on the 'Return' website. The page features a teal header with the course title and a price tag of 1,100,000 VND. Below the header, there are navigation tabs for 'Giới thiệu', 'Bài học', 'Điểm đến', and 'Đánh giá'. The main content area is divided into three sections: 'Mô tả' (Description), 'Tại sao chọn chúng tôi' (Why choose us), and 'Khóa học này bao gồm' (What this course includes). The 'Mô tả' section lists 61 lectures, 42 video demonstrations, and 58 activities. The 'Tại sao chọn chúng tôi' section highlights four benefits: flexible learning, self-paced learning, diverse content, and support. The 'Khóa học này bao gồm' section provides statistics: 464 total lessons, NS difficulty level, Japanese Language category, and a last update on May 5, 2025. A 'Đánh giá khóa học' (Course Rating) section shows a 0.0 rating with a bar chart. The 'Giảng viên' (Instructor) section features a profile for Christopher, a professional translator and language instructor. The footer contains copyright information and navigation links.

Return [Tìm kiếm khóa học](#) [Dịch](#) [Kiểm tra ngữ pháp](#) [Danh mục](#)

Japanese In Context - Elementary Japanese Course

1,100,000 VND

Christopher 464 Bài học 5 Tháng 2025

Thành giờ khóa học này

[Tiếp tục học](#) [Báo cáo](#)

[Giới thiệu](#) [Bài học](#) [Điểm đến](#) [Đánh giá](#)

Mô tả

61 lectures with over 5 hours of content!
42 video demonstrations of how to write Japanese characters with over 2 hours of content by a native speaker!
58 activities with vocabulary and grammar to test your writing and reading skills!

Tại sao chọn chúng tôi

- Học mọi lúc, mọi nơi với thiết bị của bạn
- Lịch học linh hoạt theo thời gian rảnh của bạn
- Đa dạng phương pháp: video, flashcard, bài tập, bài thi
- Nhận hỗ trợ từ giáo viên và cộng đồng học viên

Khóa học này bao gồm

Tổng số bài học	464
Mức độ khó	NS
Danh mục	Japanese Language...
Cập nhật lần cuối	May 5, 2025

Đánh giá khóa học

0.0

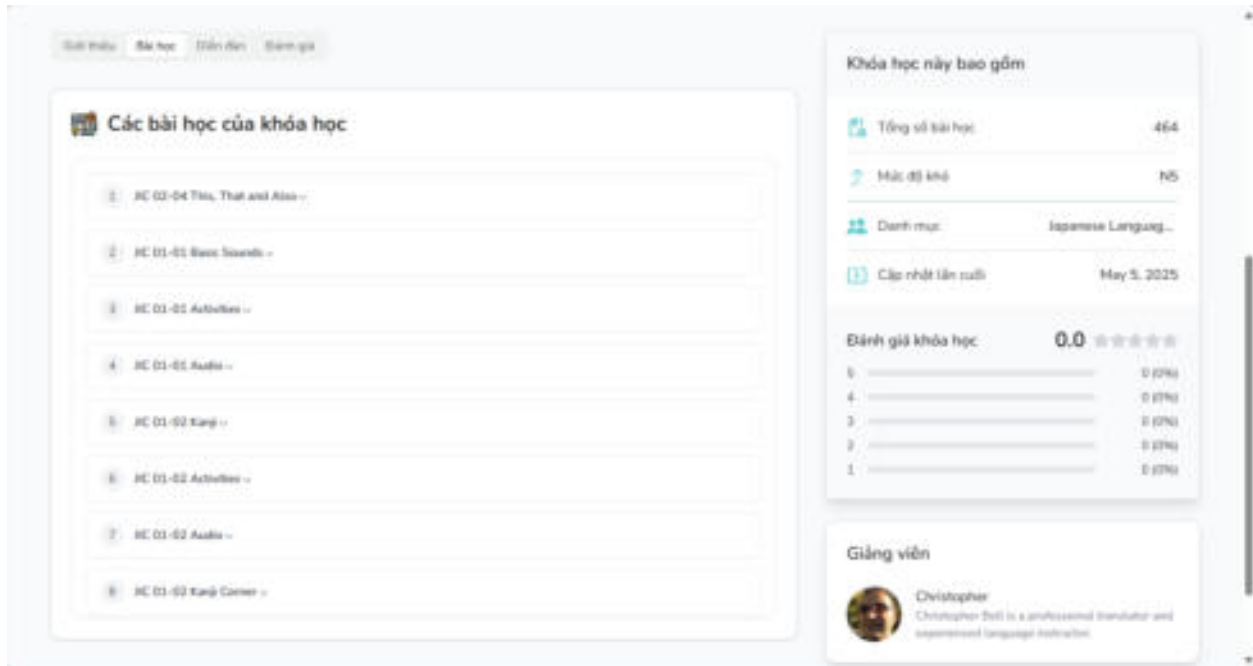
5	0 (0%)
4	0 (0%)
3	0 (0%)
2	0 (0%)
1	0 (0%)

Giảng viên

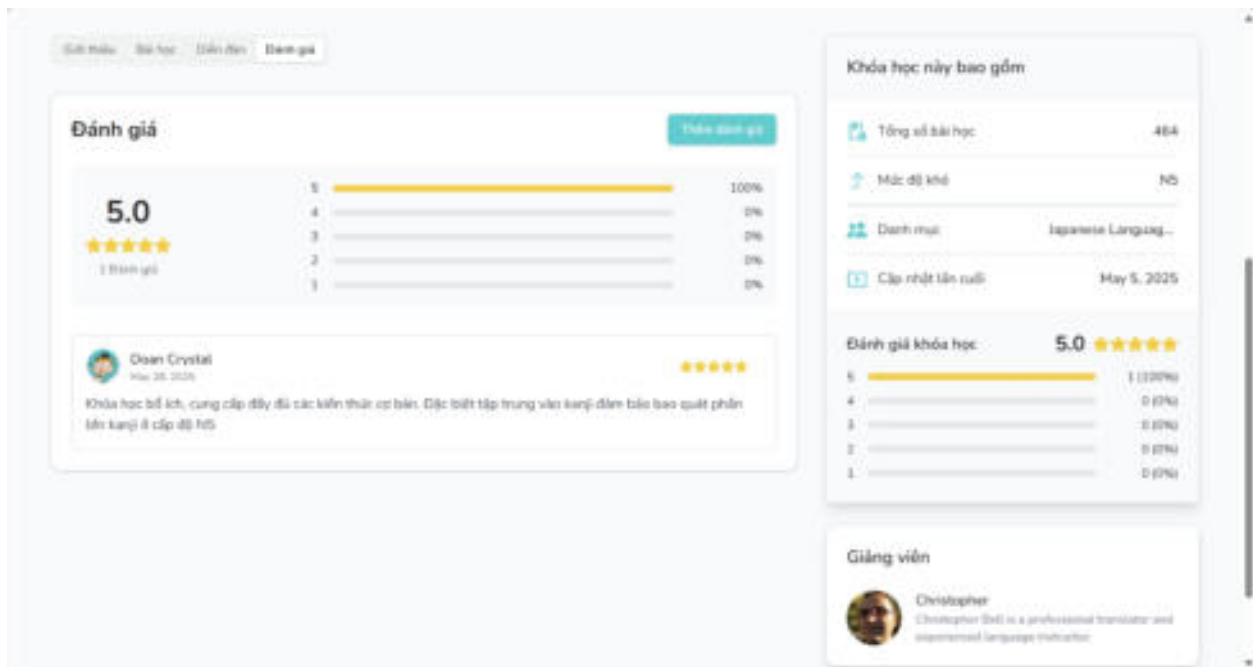
Christopher
Christopher (氏) is a professional translator and experienced language instructor.

© 2025 Return E-Learning™. All Rights Reserved. [Trang chủ](#) [Chính sách bảo mật](#) [Điều khoản](#) [Liên hệ](#)

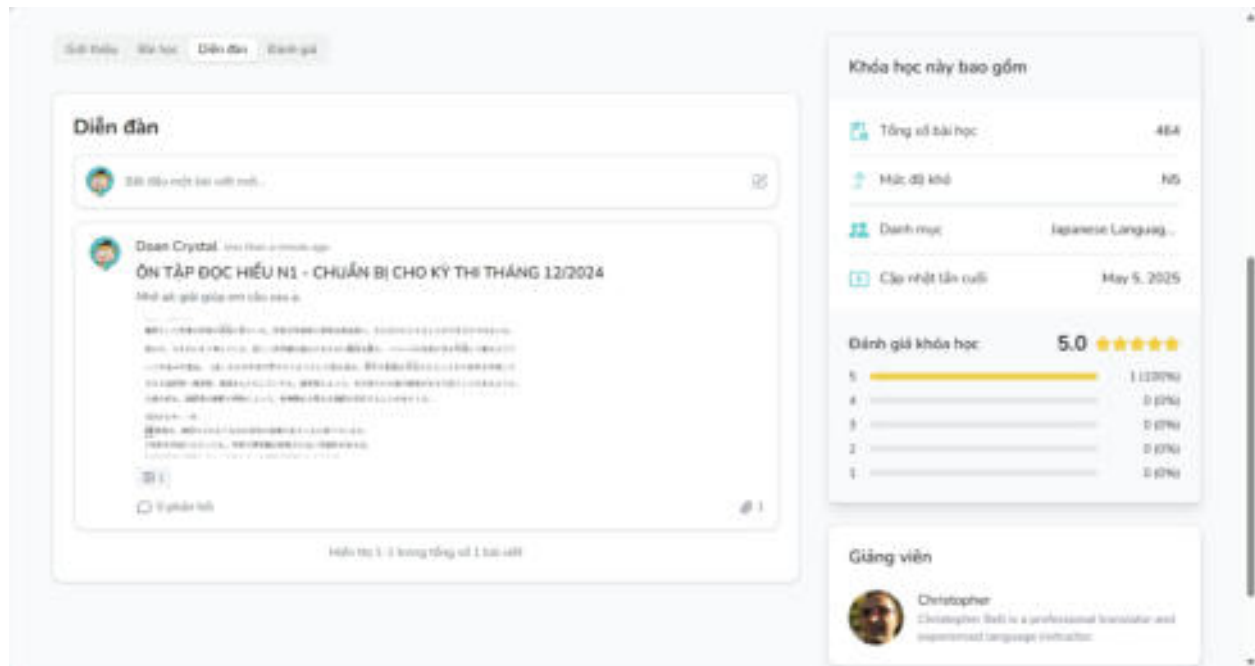
Hình- 3-20: Giao diện chi tiết khóa học – Mô tả khóa học



Hình- 3-21: Giao diện chi tiết khóa học – Danh sách bài học



Hình- 3-22: Giao diện chi tiết khóa học – Đánh giá học viên



Hình- 3-23: Giao diện chi tiết khóa học – Diễn đàn

I. Giao diện bài học

Chức năng chi tiết bài học cung cấp nội dung học tập toàn diện với video bài giảng, flashcard, luyện viết Kanji và các bài tập thực hành, giúp người học tiếp thu kiến thức hiệu quả và đa dạng.

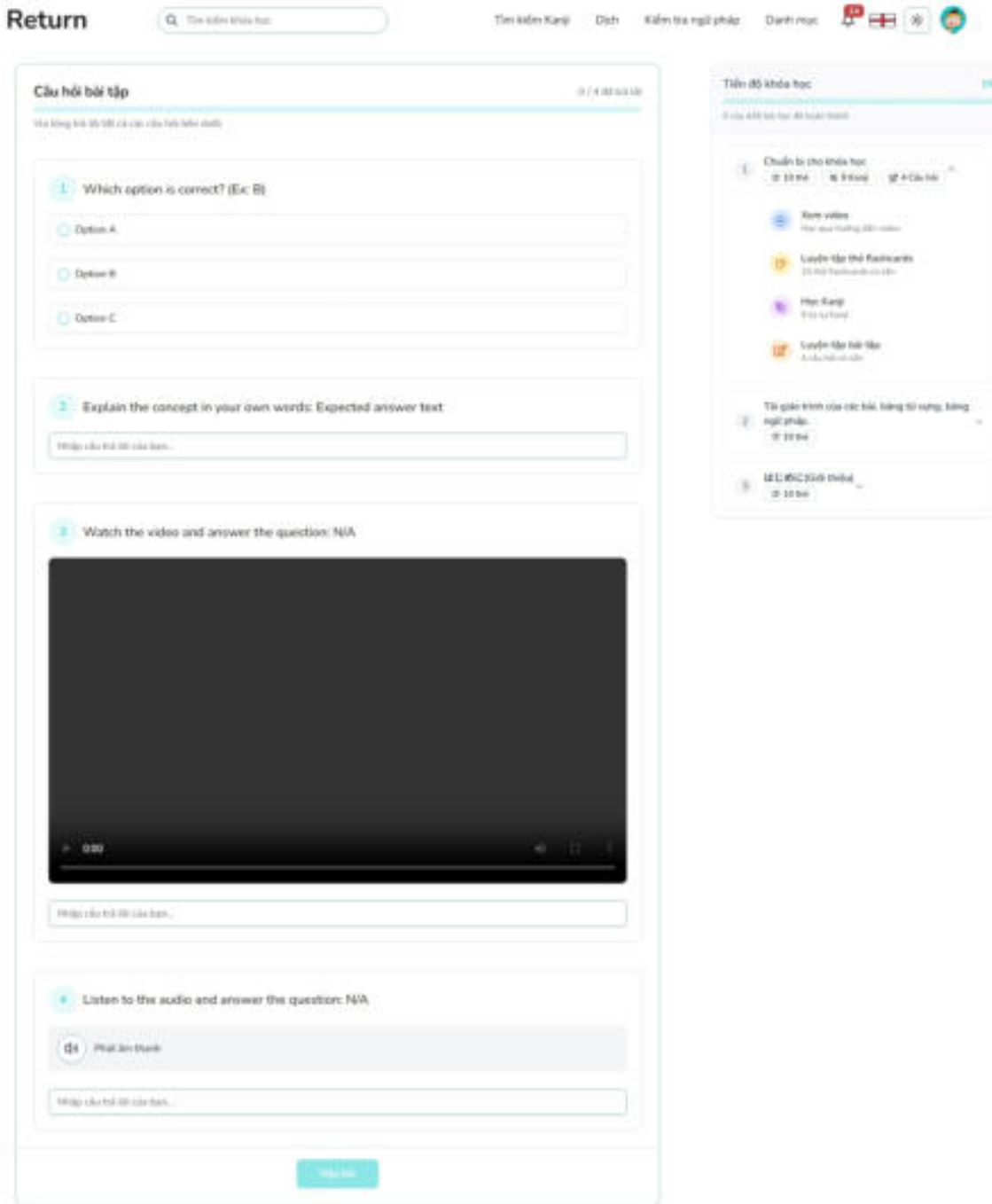
The screenshot displays the Return E-Learning website interface. At the top, there is a 'Return' logo, a search bar with the text 'Tìm kiếm khóa học', and navigation links for 'Tìm kiếm Kanji', 'Dịch', 'Kiểm tra ngữ pháp', and 'Danh mục'. There are also icons for a notification, a flag, a star, and a user profile.

The main content area is divided into several sections:

- Chọn Kanji:** A row of Kanji characters with '後' highlighted in a blue box. Other characters include 降, 横, 家, 族, 心, 配, 外, 国.
- 後 after, back:** The Kanji character '後' is shown with its meaning 'after, back' and 'Số nét: 3'. Below this are buttons for 'Thông tin' and 'Luyện tập'.
- Luyện tập:** A section for practicing writing the Kanji. It includes a 'Thứ tự nét' (Stroke order) guide showing the three strokes of '後' in order. Below this is a large box for writing practice with a faint '後' character and a set of control buttons: 'Bàn', 'Tùy', 'Hoàn tất', 'Làm lại', 'Hủy', and 'Khởi tạo mới'.
- Chuẩn bị cho khóa học:** A section for preparing for a course. It features the 'Attain Online Japanese' logo, '1 người theo dõi', and a 'Đăng ký' button. Below this is a grey box containing course details: 'Trình độ: Trung cấp, Khóa học: Chính ngữ văn, Đáp ứng tiêu chí: Trình độ: cấp 3, Thời gian: 40, Thành tích: 0/0', and 'Chỉ tiết bài học' with dates 'Ngày bắt đầu: 2025-05-05' and 'Ngày cập nhật: 2025-05-05'.

At the bottom of the page, there is a copyright notice: '© 2025 Return E-Learning™. All Rights Reserved.' and a footer with links for 'Trang chủ', 'Chính sách bảo mật', 'Câu hỏi', and 'Liên hệ'.

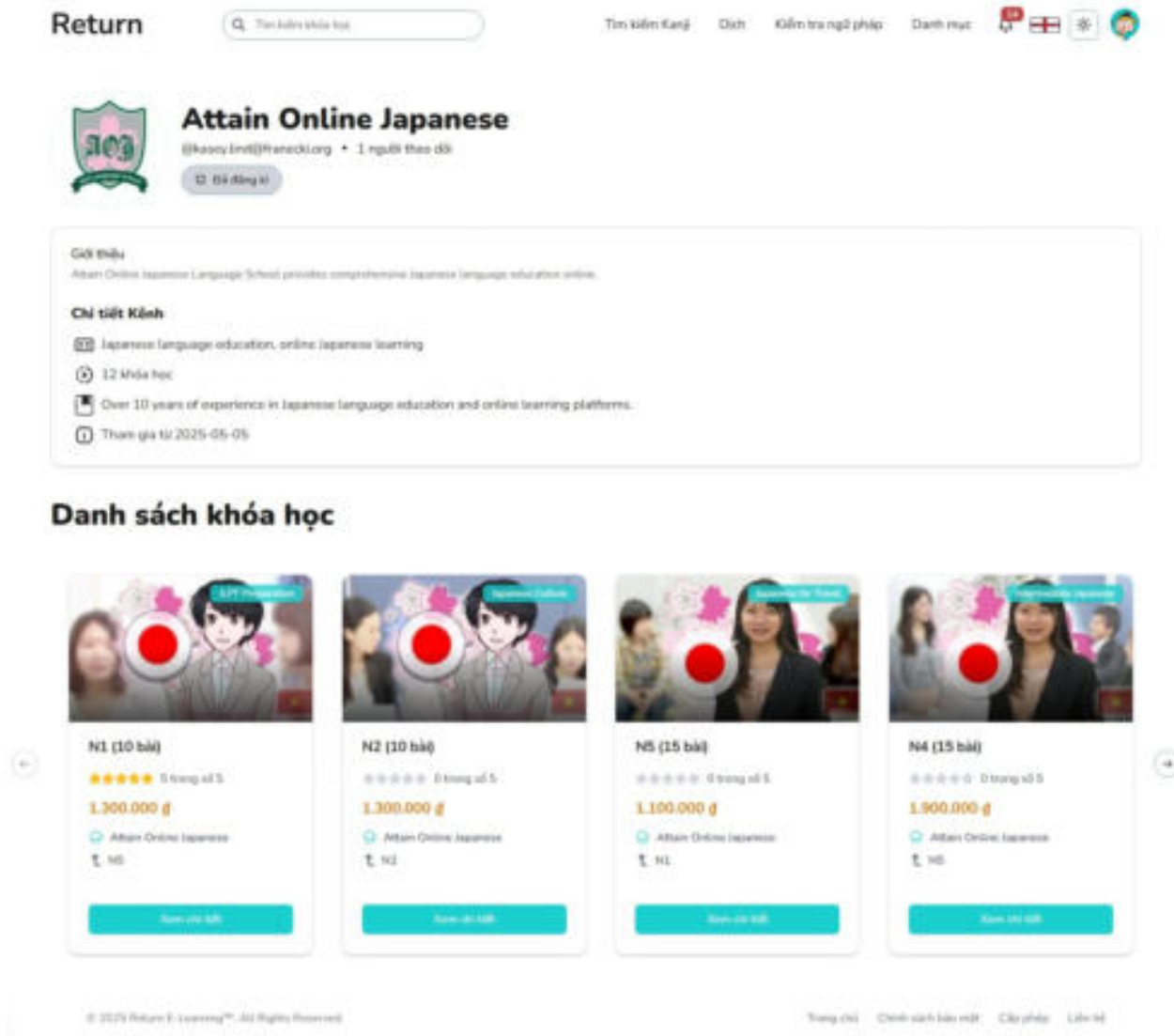
Hình- 3-26: Giao diện chi tiết bài học – Luyện viết Kanji



Hình- 3-27: Giao diện chi tiết bài học – Bài tập luyện tập

m. Giới thiệu giáo viên

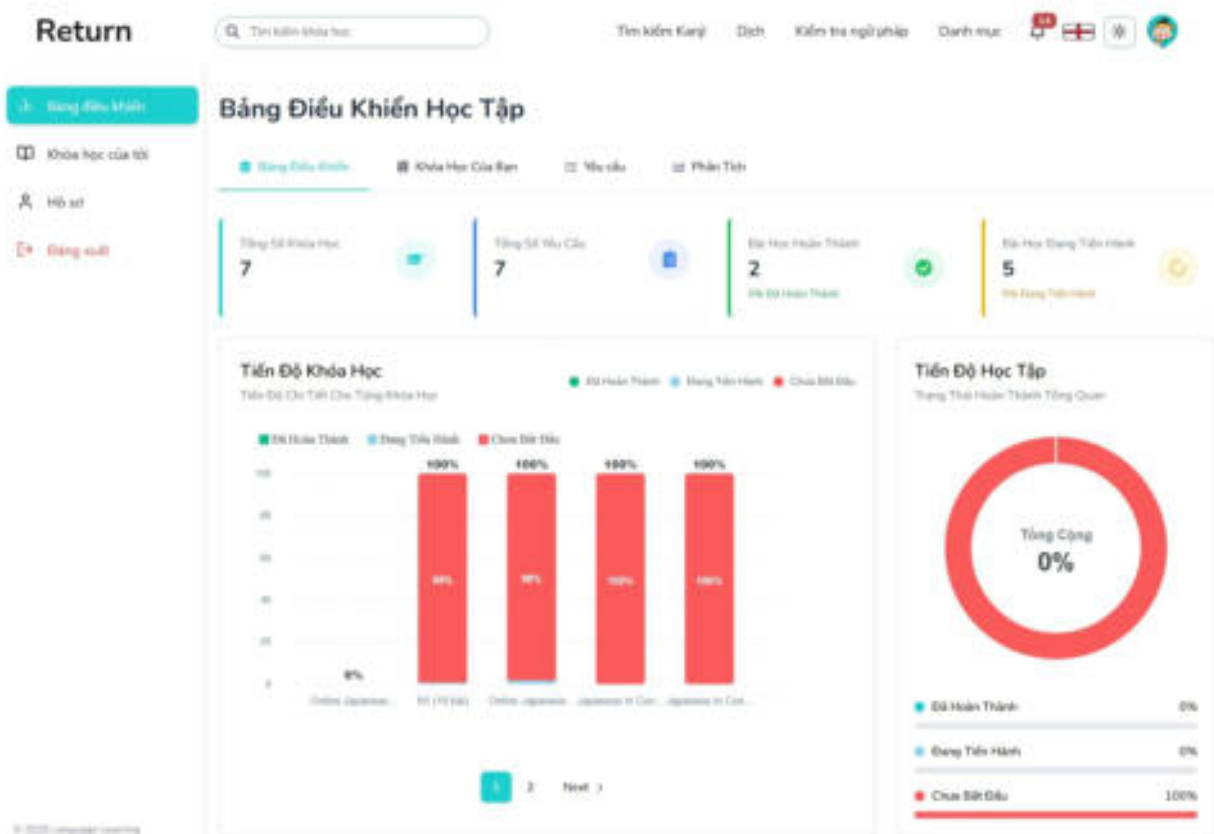
Trang giới thiệu giáo viên cung cấp thông tin về trình độ, kinh nghiệm và danh sách khóa học của từng giáo viên, giúp người học dễ dàng tìm hiểu về giáo viên và theo dõi giáo viên để nhận thông báo khi giáo viên tạo khóa học mới.



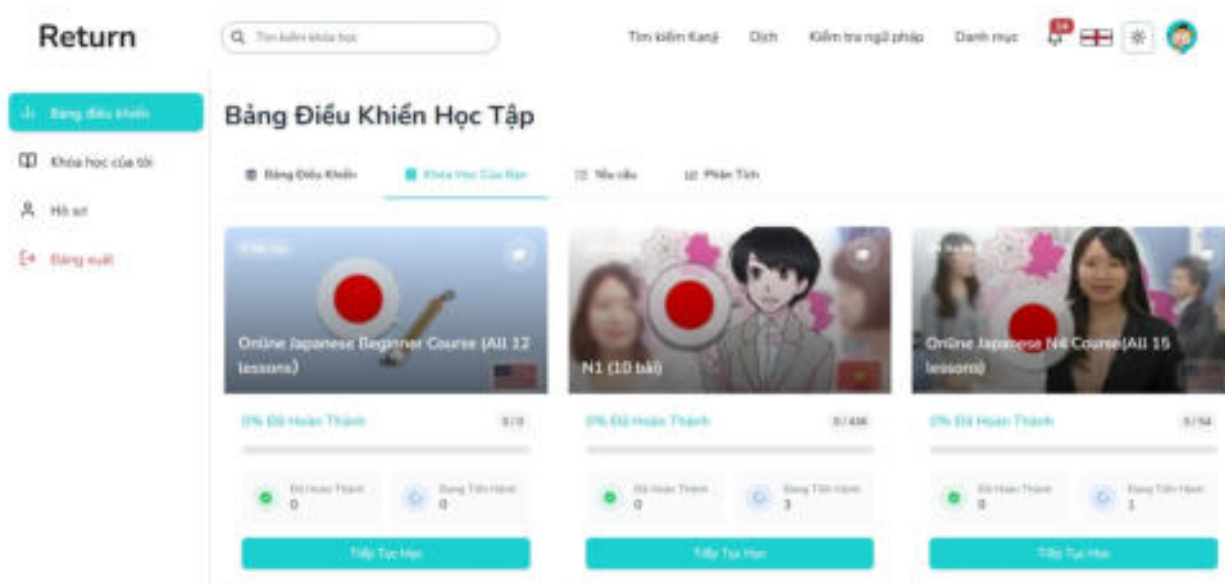
Hình- 3-28: Giao diện trang giới thiệu giáo viên

n. Bảng điều khiển

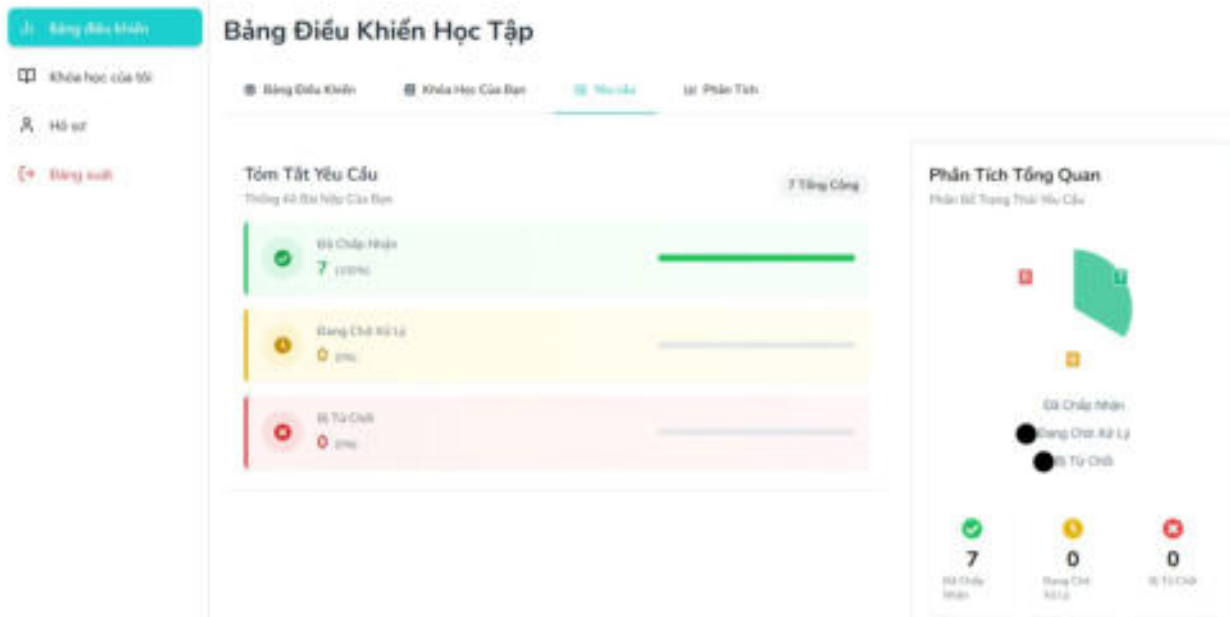
Bảng điều khiển người dùng hiển thị tổng quan quá trình học tập, các khóa học đã tham gia, tiến độ học và thông tin cá nhân, giúp người học quản lý việc học hiệu quả.



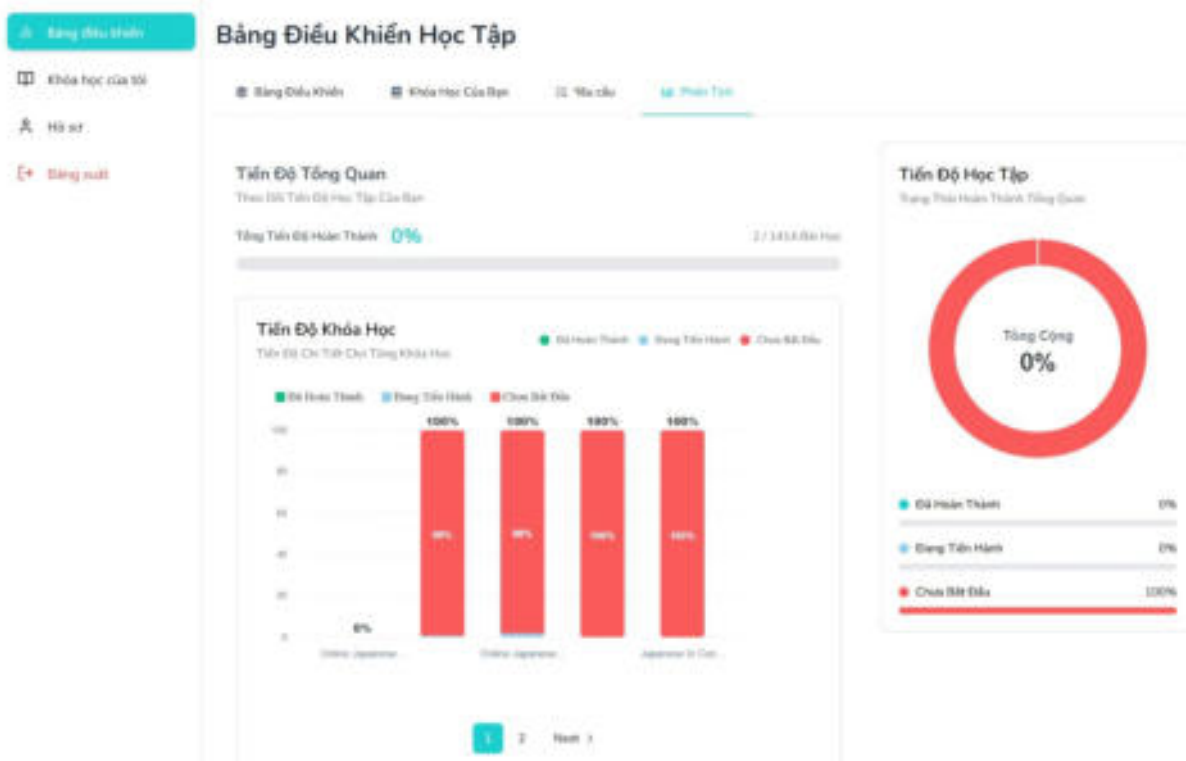
Hình- 3-29: Giao diện bảng điều khiển của người dùng – Tổng quan



Hình- 3-30: Giao diện bảng điều khiển của người dùng – Thống kê khóa học



Hình- 3-31: Giao diện bảng điều khiển của người dùng – Thống kê yêu cầu tham gia khóa học

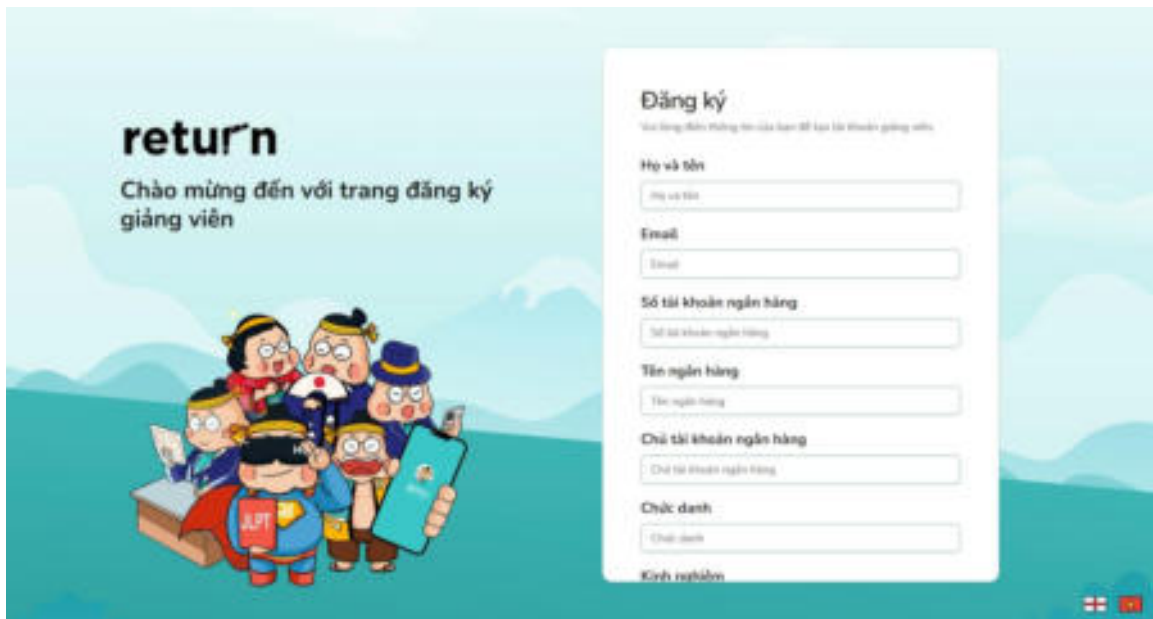


Hình- 3-32: Giao diện bảng điều khiển của người dùng – Tiến độ học tập

3.4.2. Giao diện giáo viên

a. Đăng kí

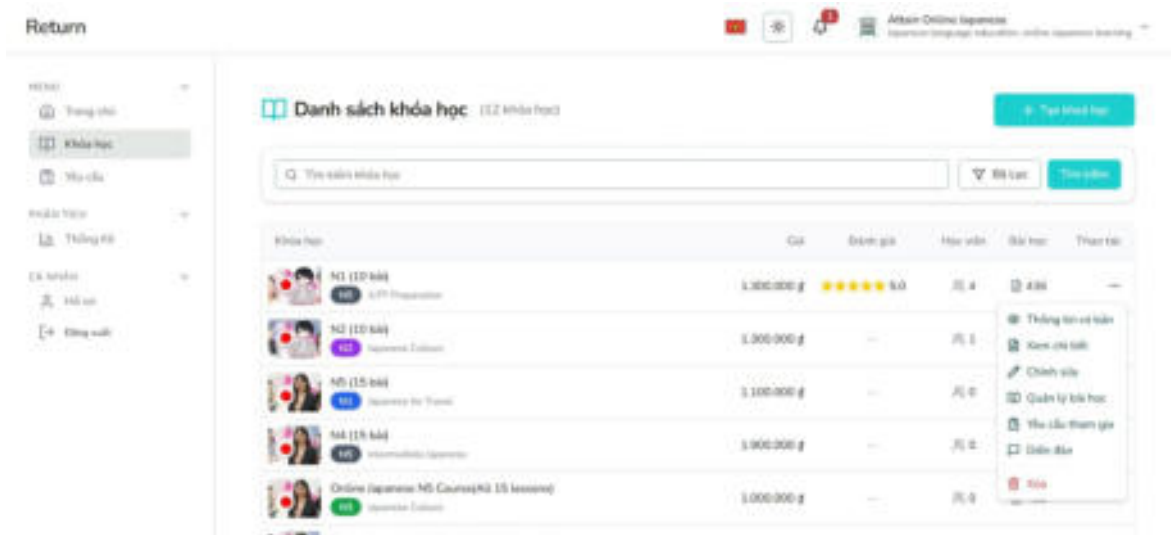
Công đăng ký dành cho những người muốn trở thành giáo viên, mở ra cơ hội chia sẻ kiến thức và đồng hành cùng học viên trên hành trình học tiếng Nhật.



Hình- 3-33: Giao diện đăng kí tài khoản giáo viên

b. Danh sách khóa học

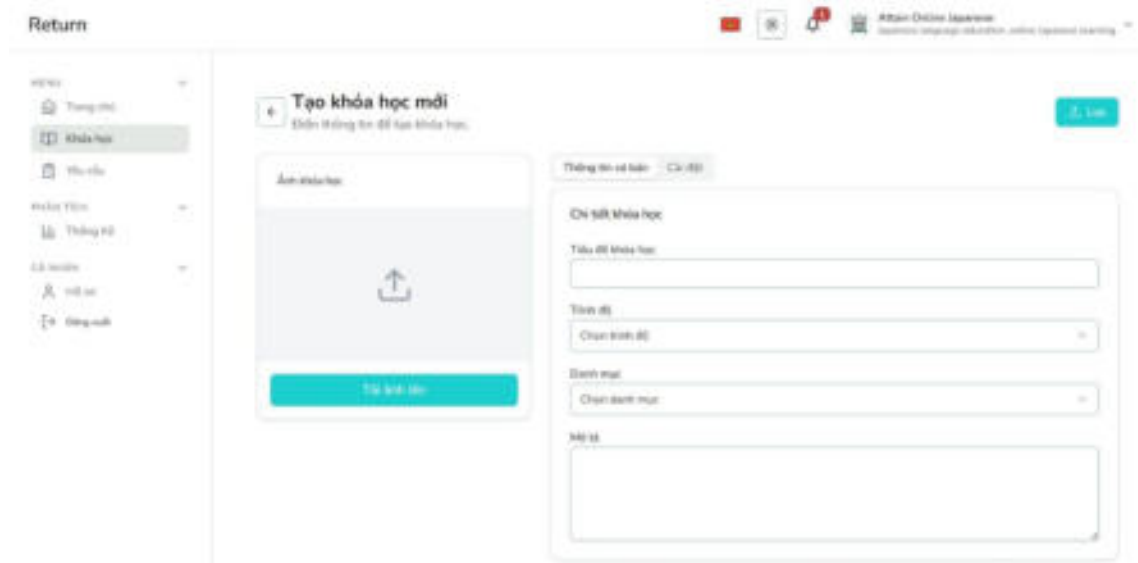
Nơi giáo viên theo dõi và quản lý toàn bộ các khóa học mình phụ trách, từ đó dễ dàng điều phối nội dung giảng dạy và tiến độ học tập của học viên.



Hình- 3-34: Giao diện danh sách khóa học

c. Tạo khóa học

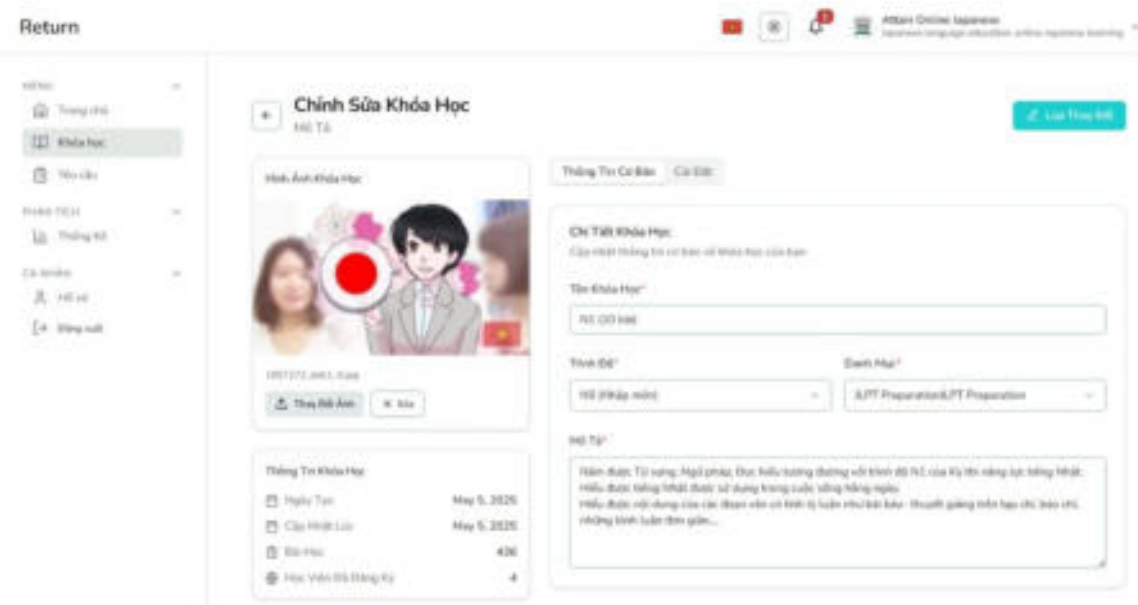
Cung cấp công cụ để giáo viên xây dựng khóa học mới một cách linh hoạt, đáp ứng nhu cầu học tập đa dạng của người học.



Hình- 3-35: Giao diện tạo khóa học mới

d. Chỉnh sửa khóa học

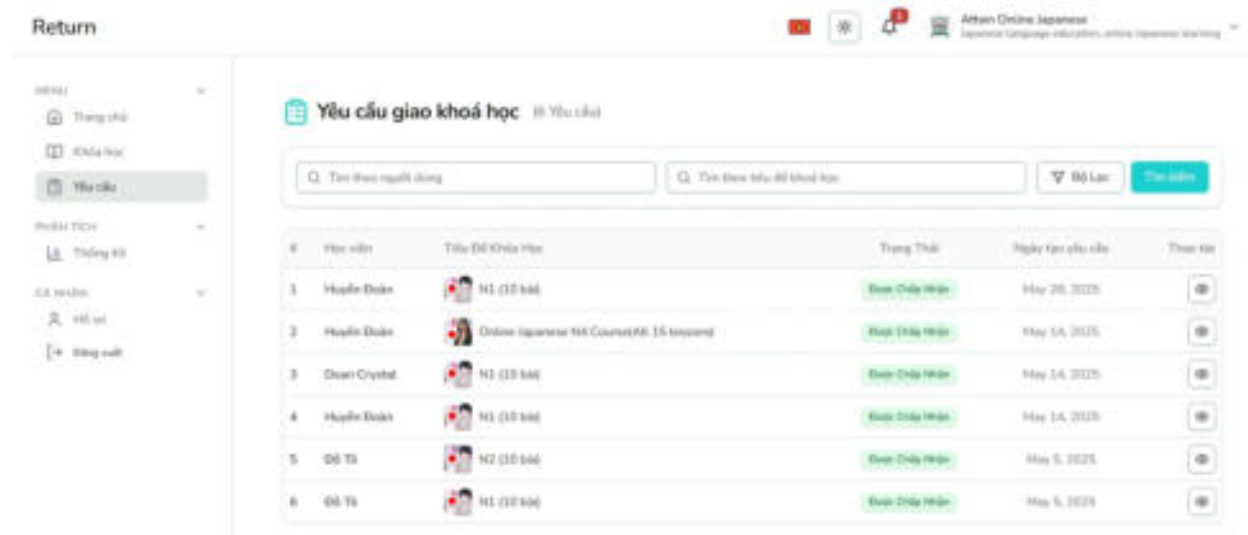
Cho phép điều chỉnh nội dung và thông tin khóa học đã tạo, nhằm đảm bảo chất lượng và tính cập nhật của chương trình giảng dạy.



Hình- 3-36: Giao diện chỉnh sửa khóa học

e. Danh sách yêu cầu tham gia khóa học

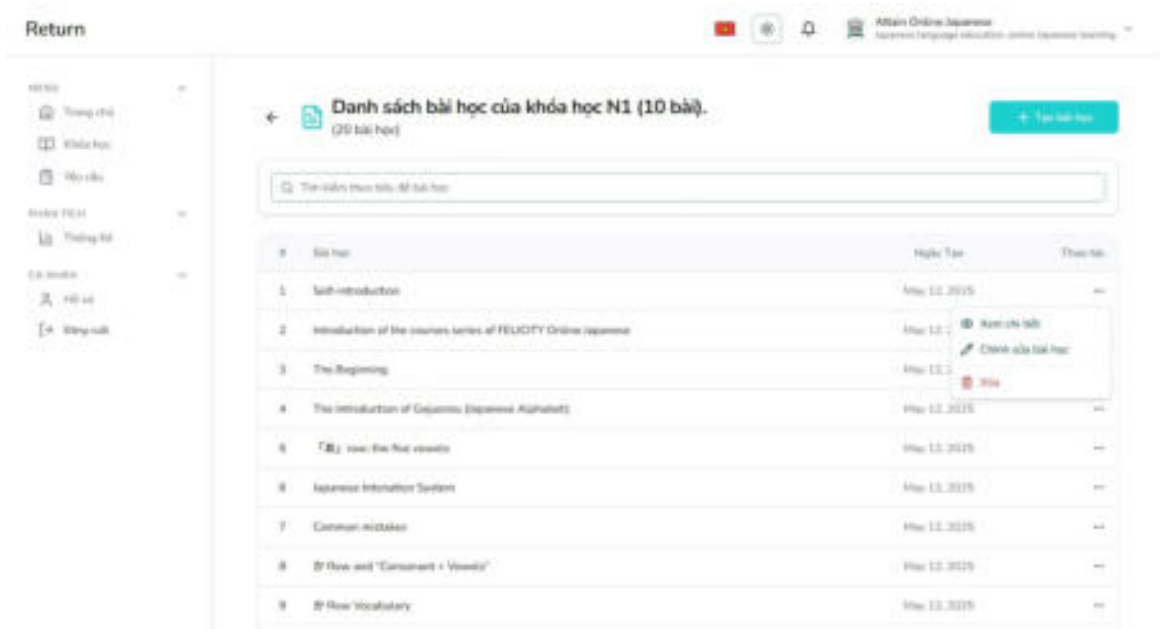
Trang tổng hợp các yêu cầu ghi danh từ học viên, hỗ trợ giáo viên dễ dàng xét duyệt và quản lý quá trình tham gia khóa học.



Hình- 3-37: Giao diện danh sách yêu cầu tham gia khóa học

f. Danh sách bài học

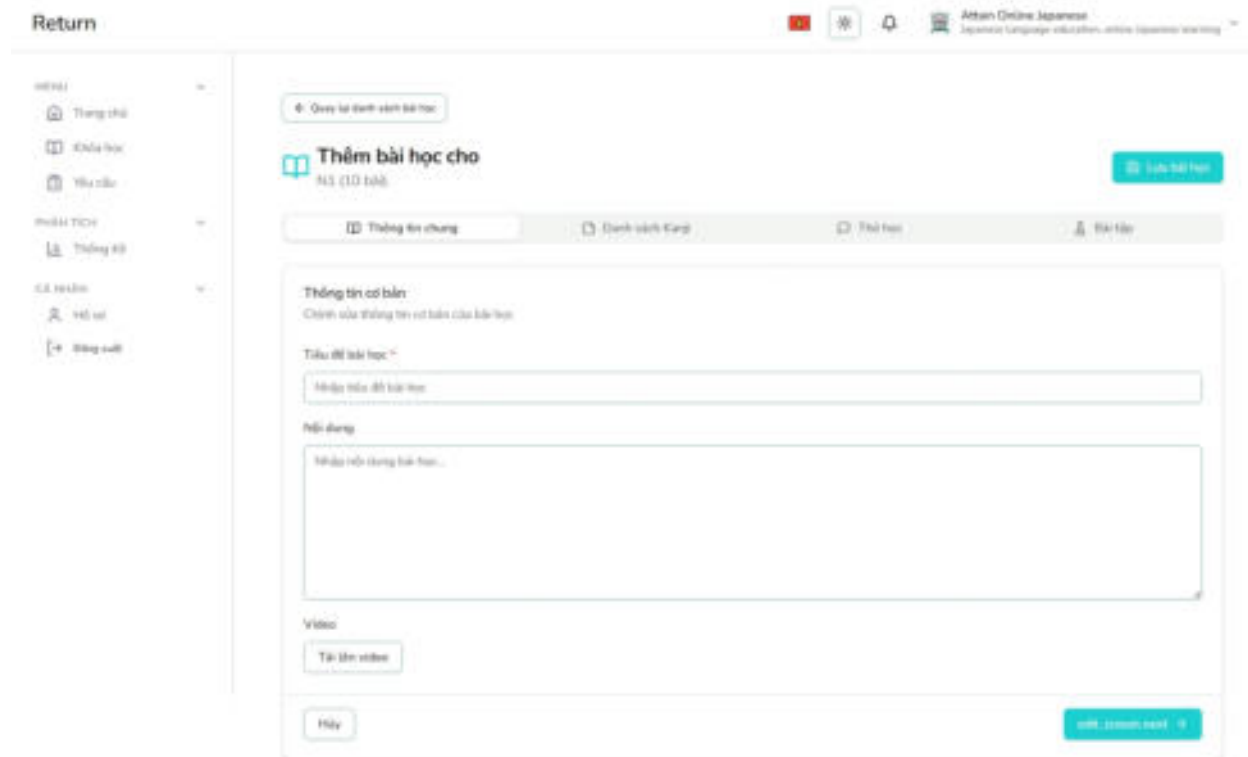
Tổng hợp toàn bộ các bài học thuộc khóa học của giáo viên, tạo điều kiện thuận lợi cho việc quản lý và chỉnh sửa nội dung giảng dạy.



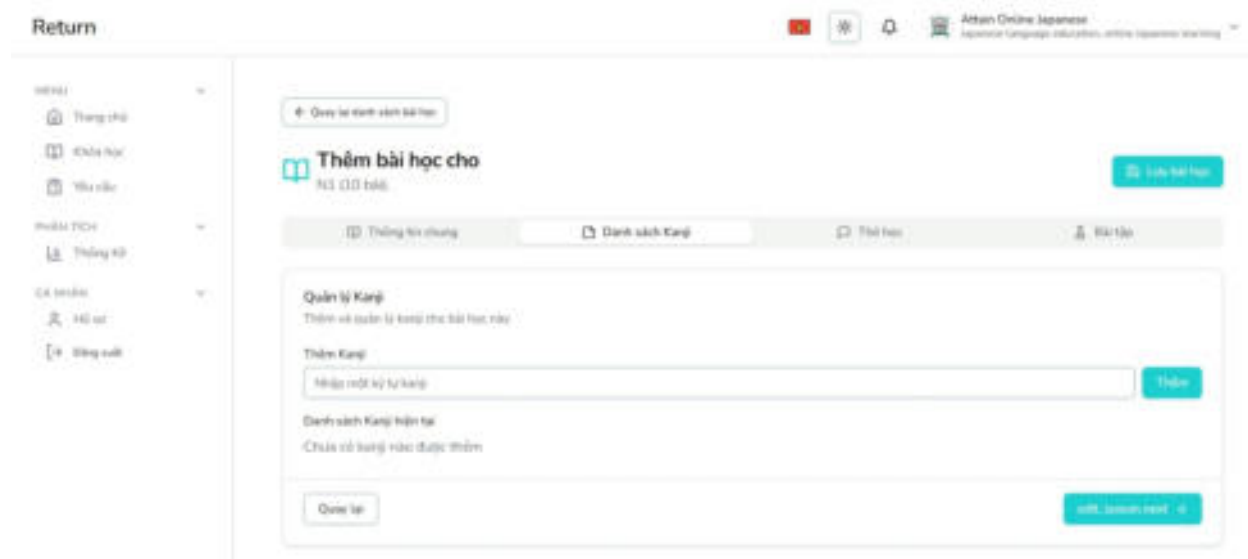
Hình- 3-38: Giao diện danh sách khóa học

g. Tạo bài học

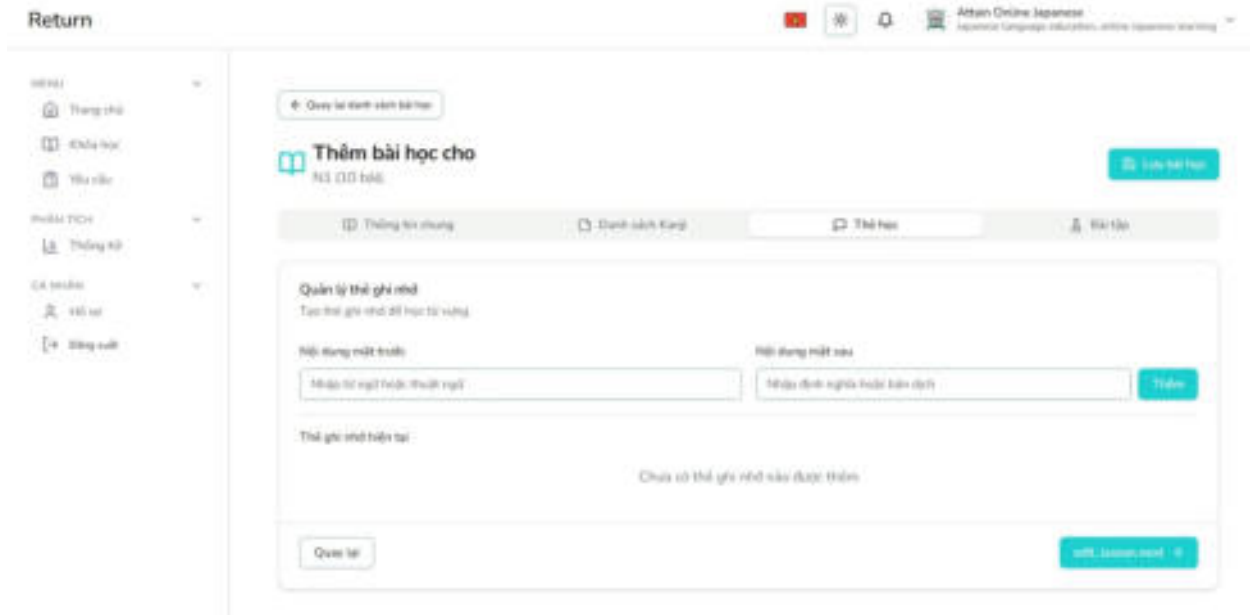
Giáo viên có thể tạo bài học mới với đầy đủ nội dung như video giảng dạy, flashcard, bài tập và phần luyện viết, giúp tăng tính tương tác và hiệu quả học tập.



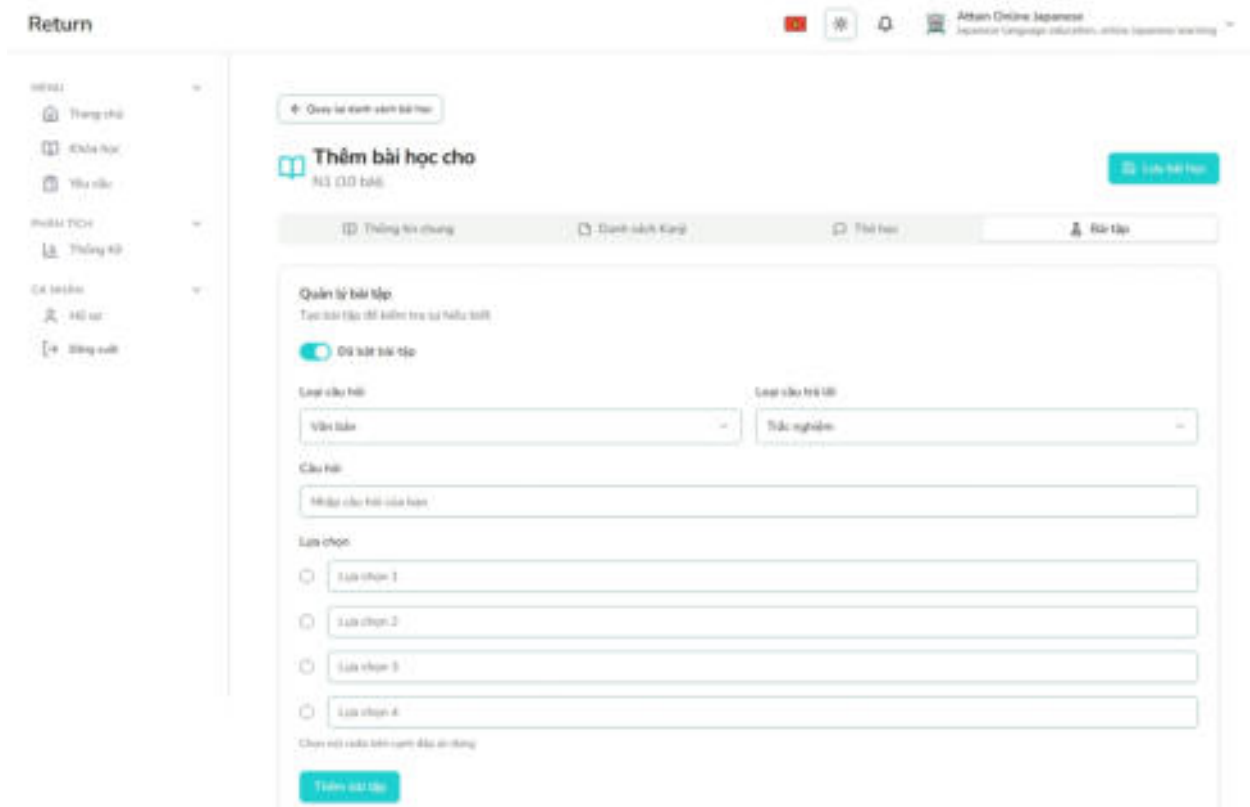
Hình- 3-39: Giao diện tạo khóa học – Thông tin cơ bản



Hình- 3-40: Giao diện tạo khóa học – Danh sách Kanji



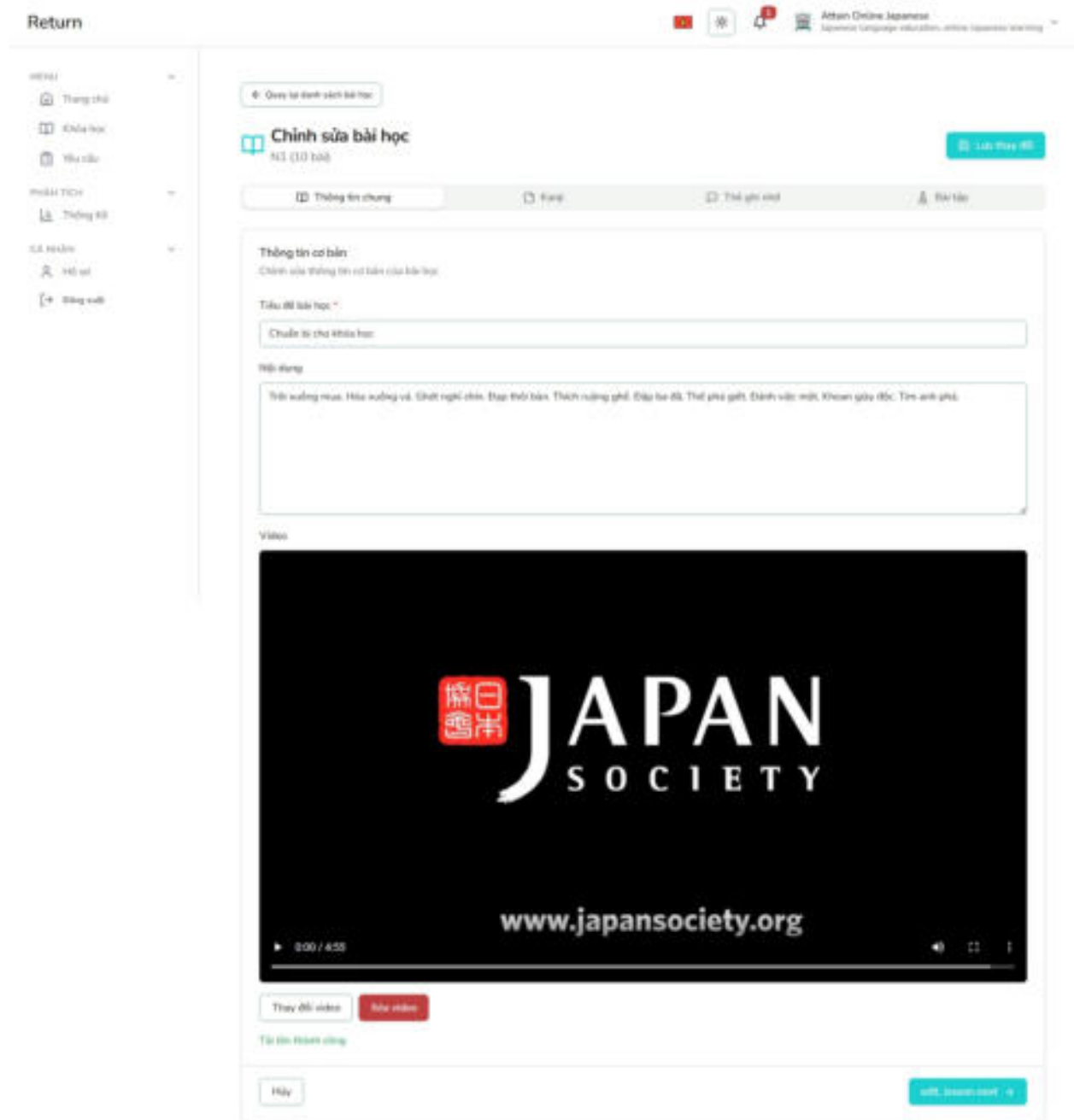
Hình- 3-41: Giao diện tạo khóa học – Danh sách flashcard



Hình- 3-42: Giao diện tạo khóa học – Bài tập

h. Chỉnh sửa bài học

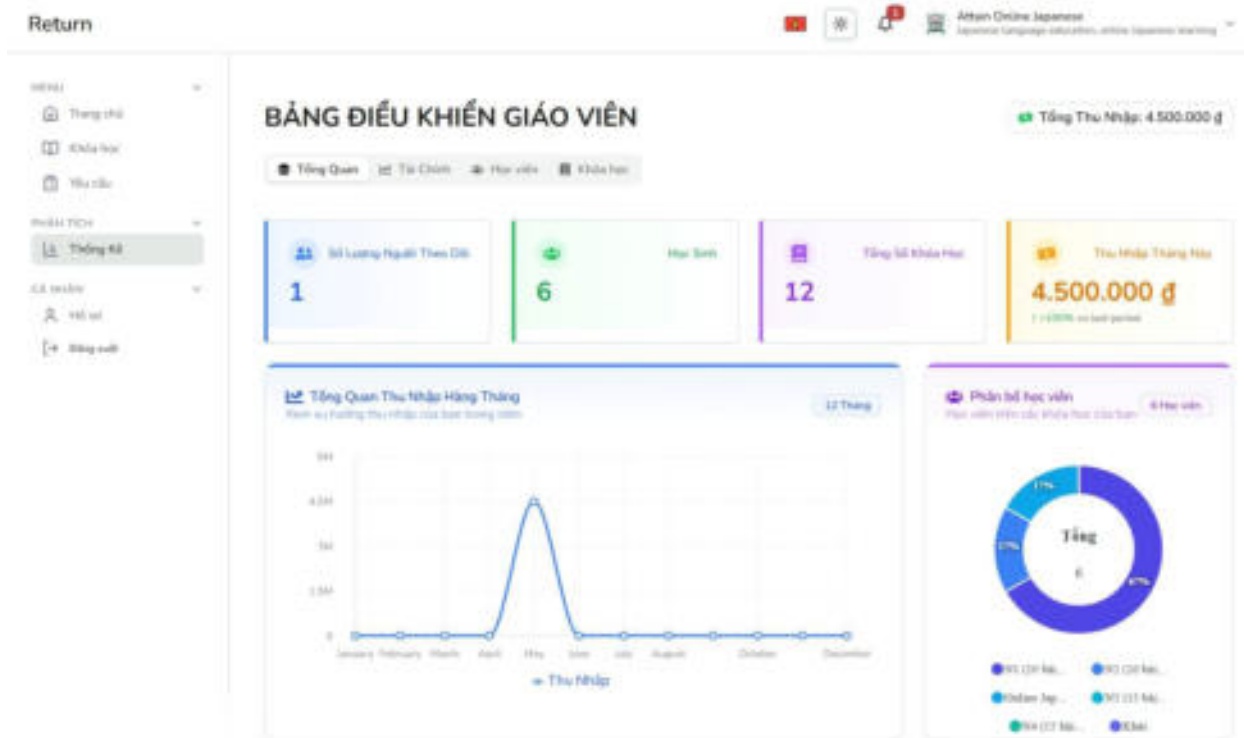
Công cụ cho phép giáo viên cập nhật và điều chỉnh nội dung bài học để cải thiện trải nghiệm học tập của người học theo từng thời điểm.



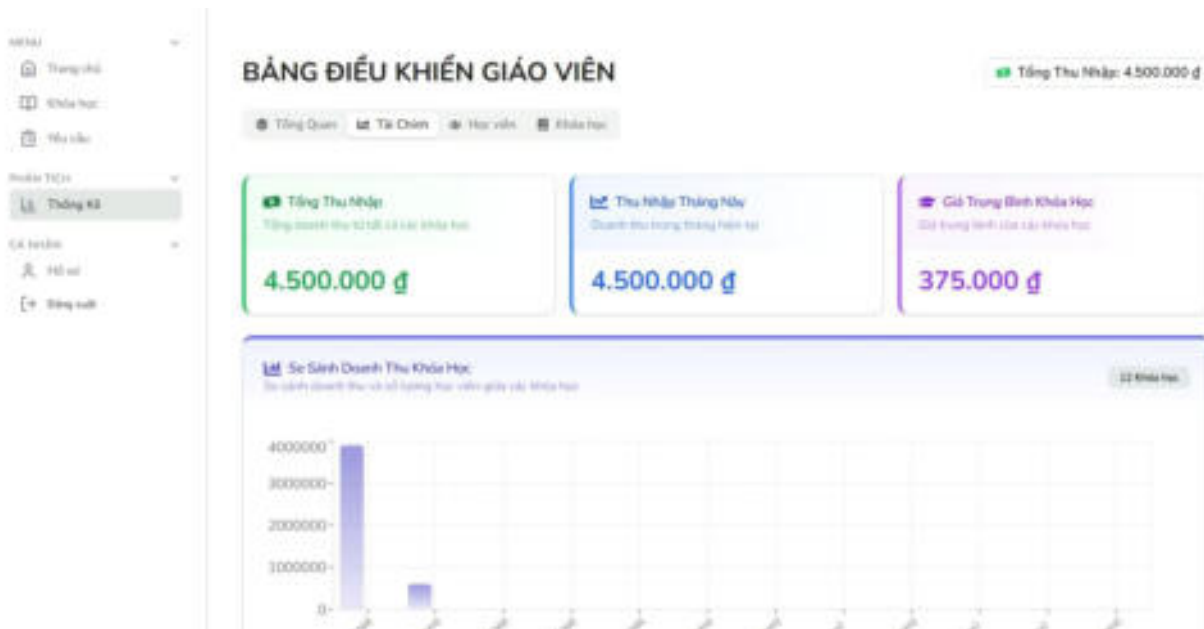
Hình- 3-43: Giao diện chỉnh sửa bài học

i. Thống kê

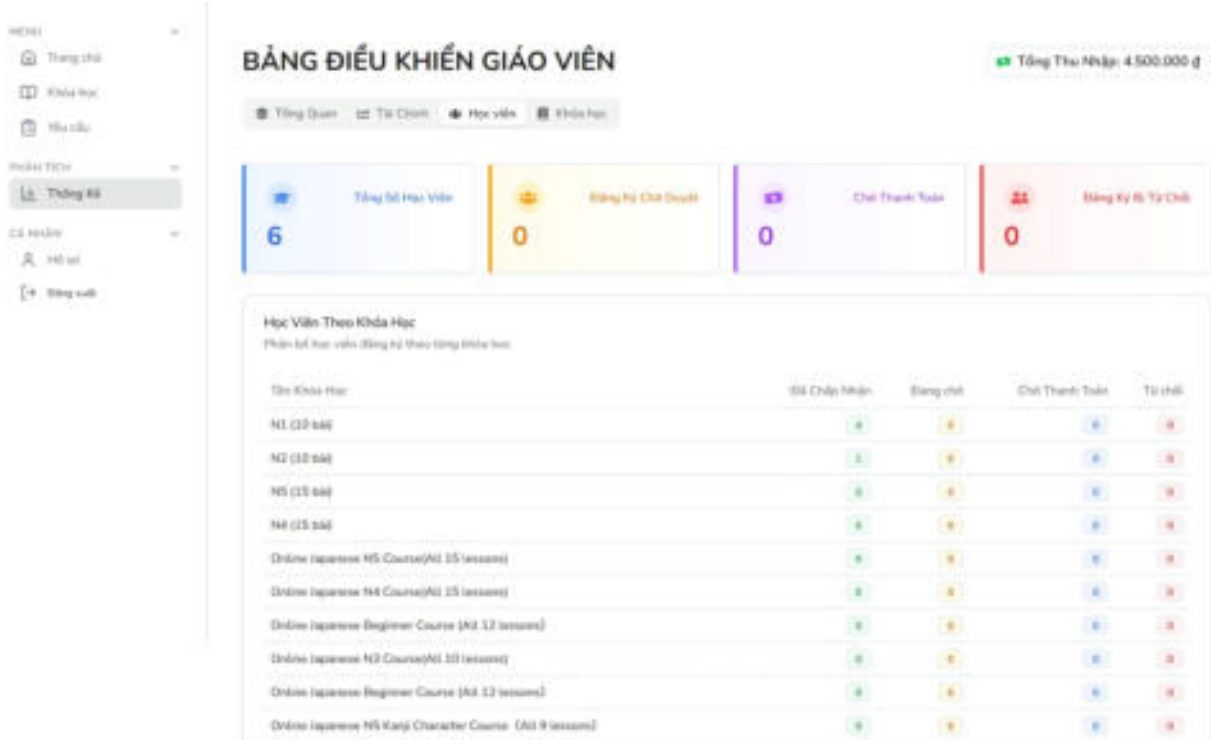
Trang thống kê cung cấp cái nhìn tổng quan về thu nhập, số lượng học viên tham gia, khóa học.



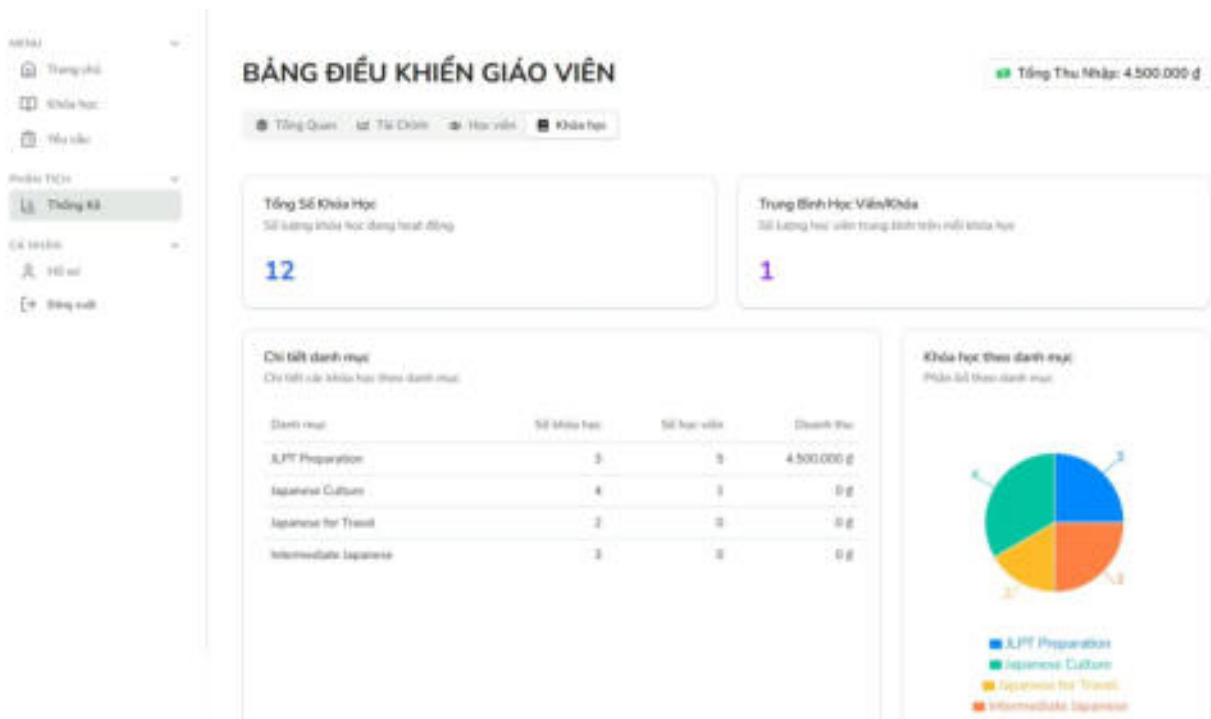
Hình- 3-44: Giao diện bảng điều khiển của giáo viên – Tổng quan



Hình- 3-45: Giao diện bảng điều khiển của giáo viên – Tài chính



Hình- 3-46: Giao diện bảng điều khiển của giáo viên – Yêu cầu đăng kí khóa học

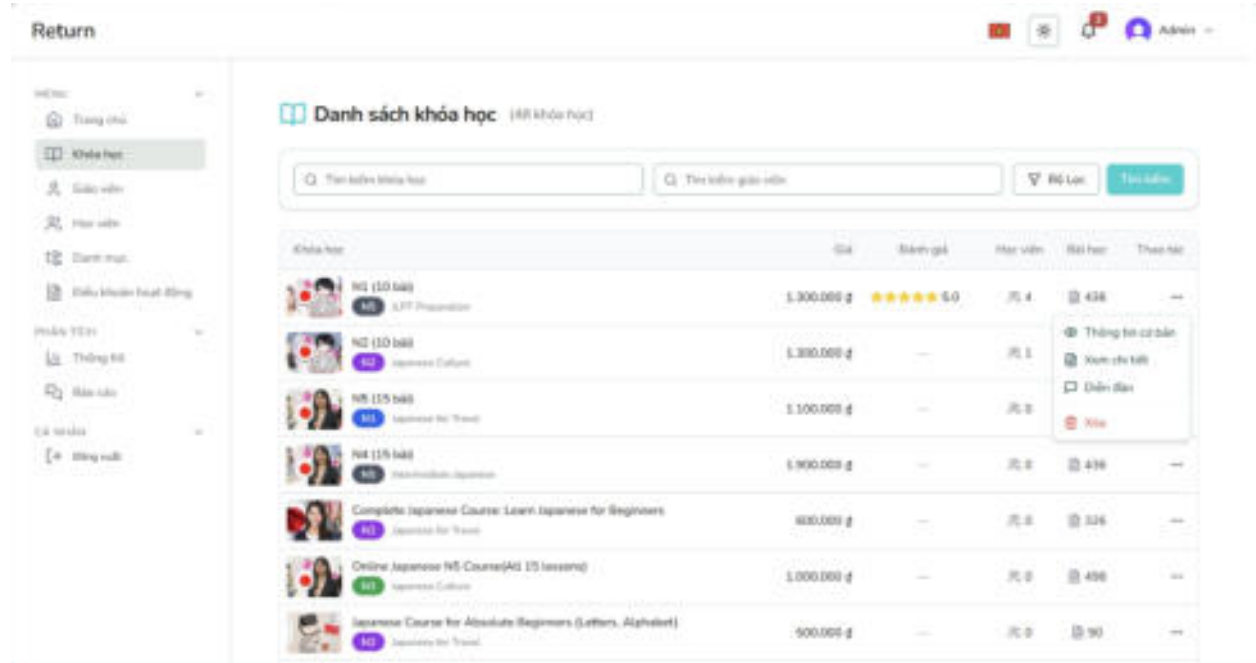


Hình- 3-47: Giao diện bảng điều khiển của giáo viên – Khóa học

3.4.3. Giao diện quản trị viên

a. Danh sách khóa học

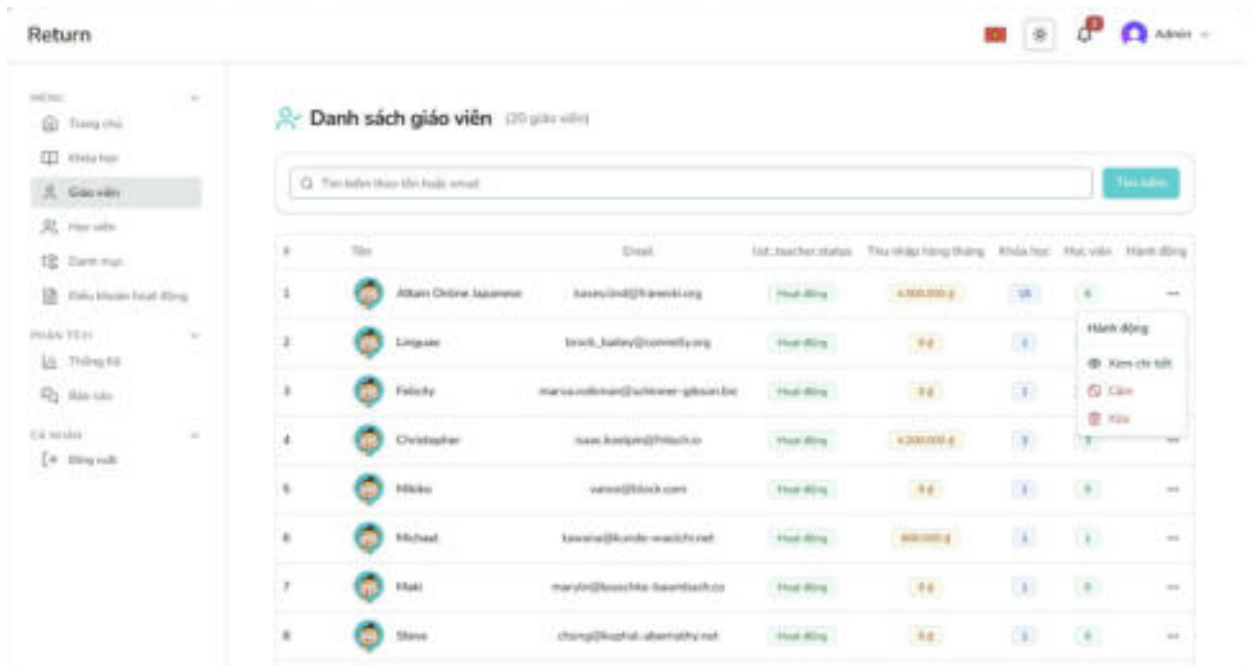
Quản trị viên có thể theo dõi và kiểm soát toàn bộ các khóa học trên hệ thống, đảm bảo nội dung giảng dạy phù hợp và chất lượng.



Hình- 3-48: Giao diện danh sách khóa học

b. Danh sách giáo viên

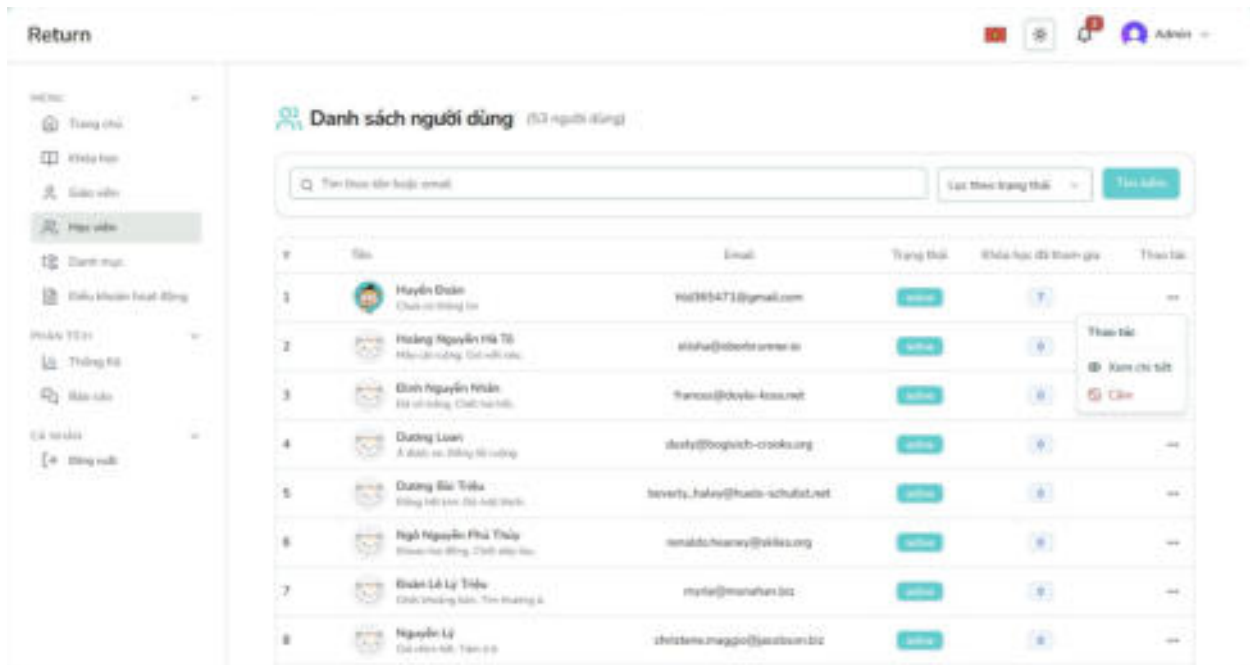
Giao diện hiển thị danh sách tất cả giáo viên, hỗ trợ việc quản lý hồ sơ và quyền truy cập của từng người.



Hình- 3-49: Giao diện danh sách giáo viên

c. Danh sách học viên

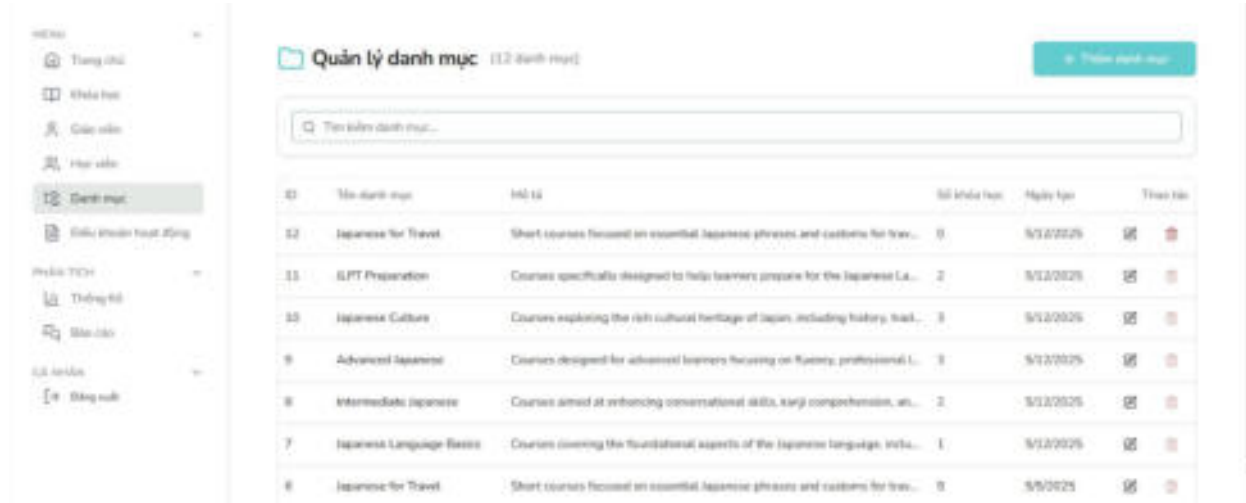
Cho phép quản trị viên giám sát thông tin của học viên nhằm đảm bảo tính minh bạch và hỗ trợ kịp thời.



Hình- 3-50: Giao diện danh sách người dùng

d. Danh sách danh mục

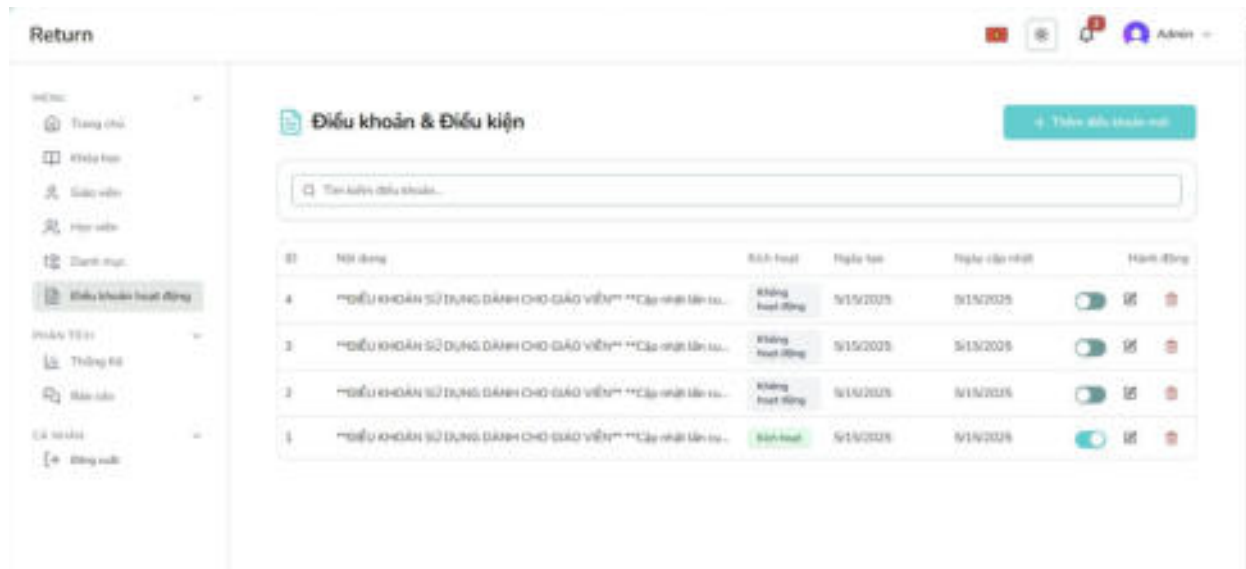
Trang quản lý các danh mục khóa học giúp hệ thống được tổ chức rõ ràng, hỗ trợ người dùng dễ dàng tìm kiếm và phân loại nội dung.



Hình- 3-51: Giao diện danh sách danh mục

e. Danh sách điều khoản hoạt động

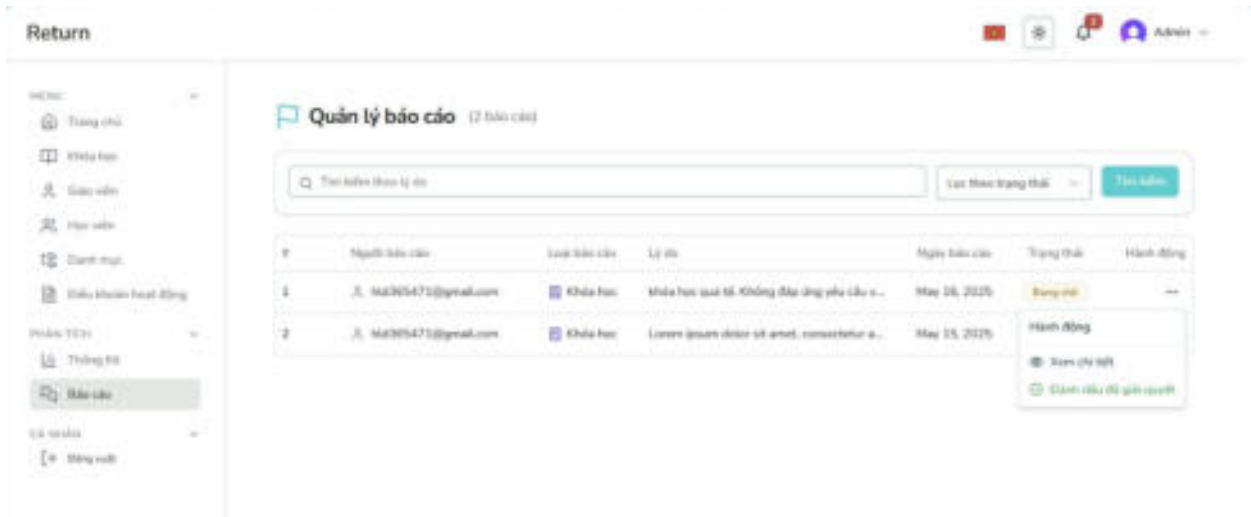
Nơi lưu trữ và chỉnh sửa các quy định, điều khoản sử dụng hệ thống nhằm đảm bảo tính pháp lý và chuẩn mực hoạt động.



Hình- 3-52: Giao diện danh sách điều khoản hợp đồng giáo viên

f. Danh sách báo cáo

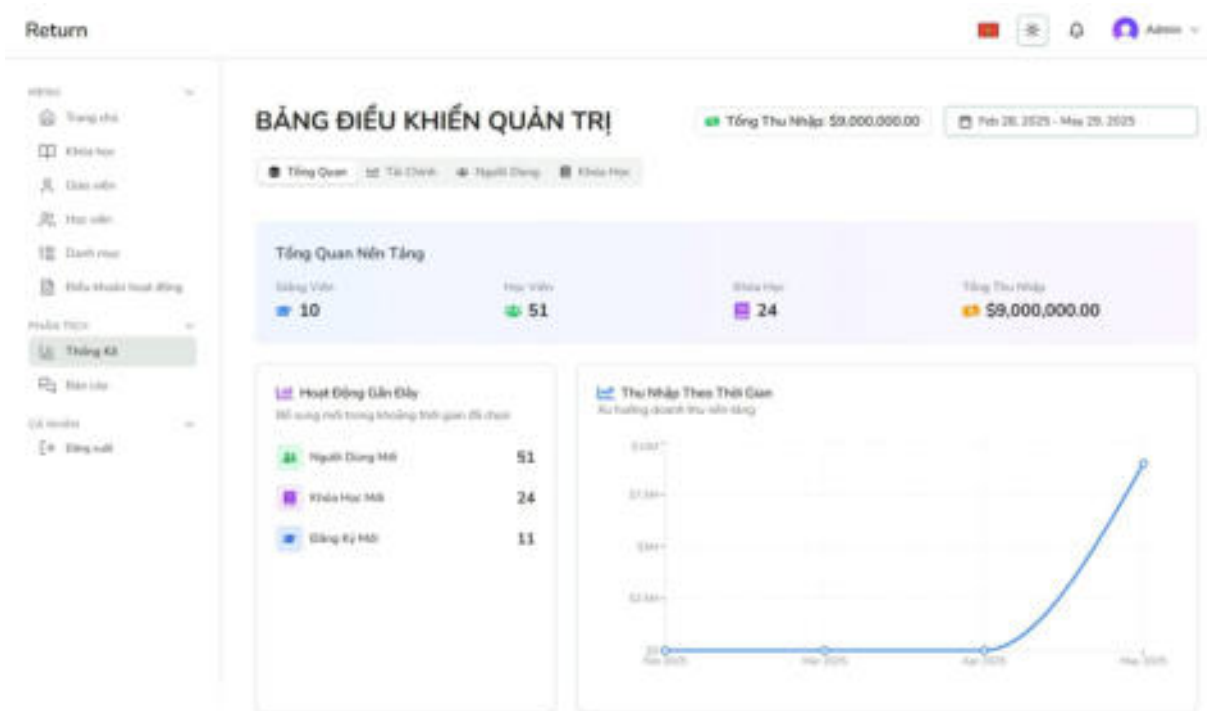
Tổng hợp các báo cáo từ người dùng về nội dung không phù hợp hoặc hành vi vi phạm, giúp quản trị viên xử lý và cải thiện môi trường học tập.



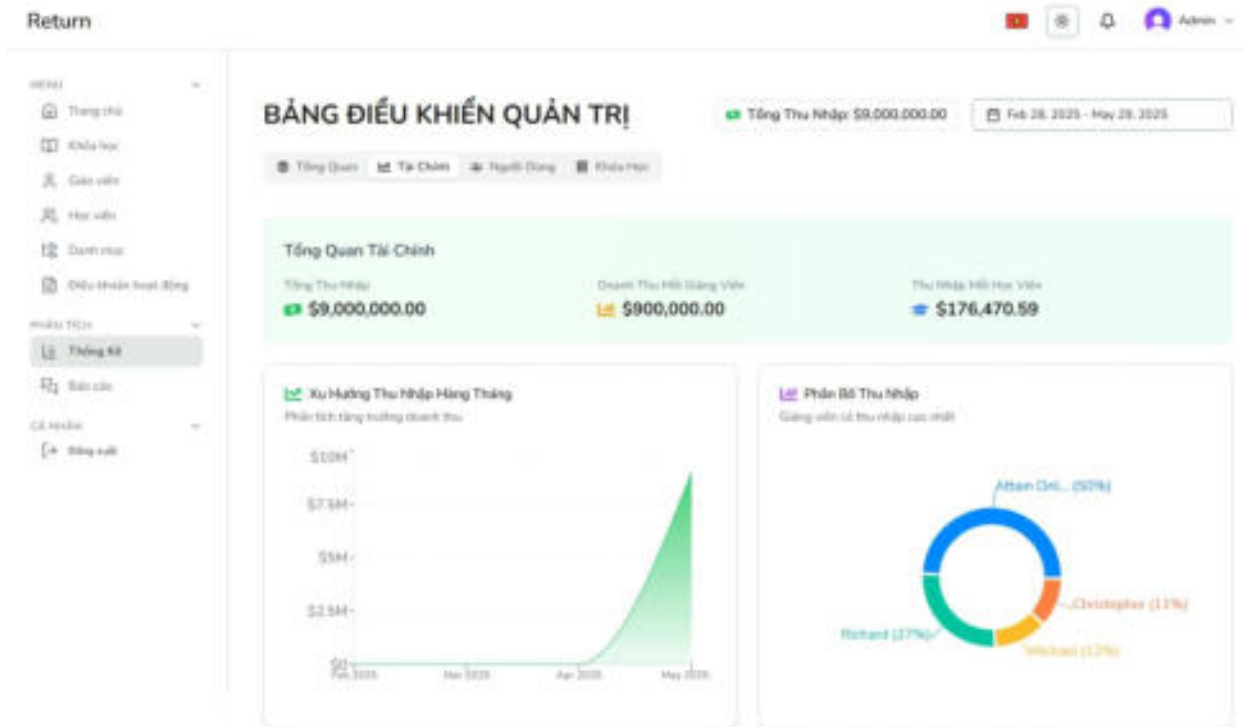
Hình- 3-53: Giao diện danh sách báo cáo

g. Thống kê

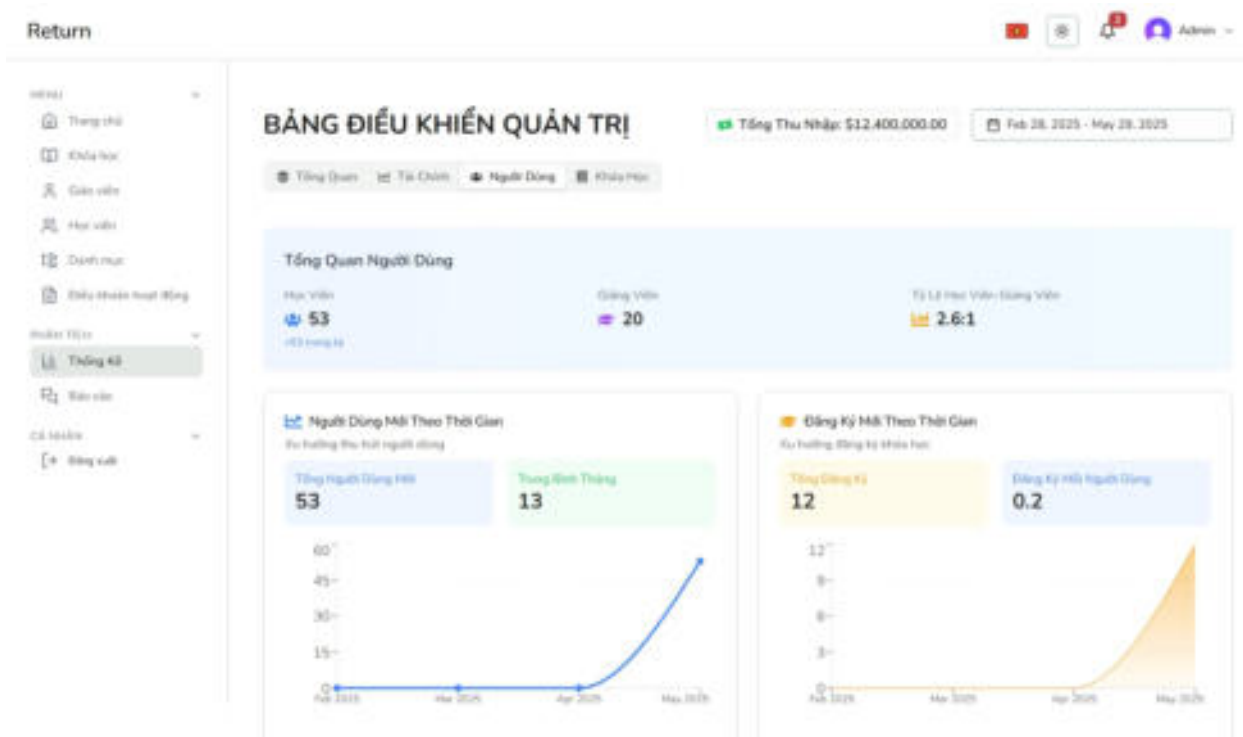
Cung cấp số liệu toàn diện về hoạt động hệ thống, bao gồm lượng người dùng, khóa học, tương tác... nhằm hỗ trợ việc ra quyết định và nâng cao hiệu quả quản lý.



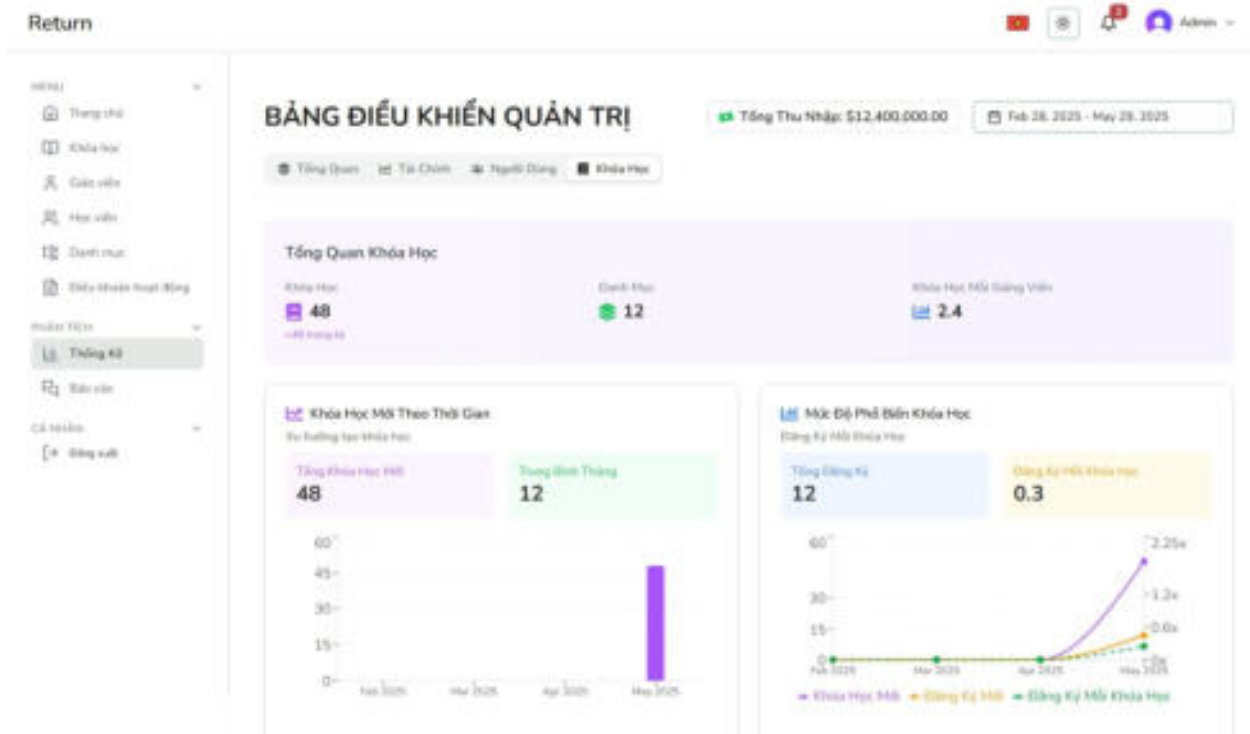
Hình- 3-54: Giao diện bảng điều khiển của quản trị viên - Tổng quan



Hình- 3-55: Giao diện bảng điều khiển của quản trị viên - Tài chính



Hình- 3-56: Giao diện bảng điều khiển của quản trị viên - Người dùng



Hình- 3-57: Giao diện bảng điều khiển của quản trị viên - Khóa học

3.5. Kiểm thử hệ thống

3.5.1. Phạm vi kiểm thử

Trong dự án này, kiểm thử được thực hiện dựa trên các thành phần quan trọng nhằm đảm bảo tính đúng đắn và hoạt động ổn định của ứng dụng. Phạm vi kiểm thử được xác định cụ thể như sau:

- Kiểm thử Model
 - Kiểm tra tính đúng đắn của các phương thức trong model.
 - Xác minh các ràng buộc (validations) để đảm bảo dữ liệu được lưu vào cơ sở dữ liệu luôn hợp lệ.
 - Kiểm tra mối quan hệ (associations) giữa các model như has_many, belongs_to, has_one.
 - Kiểm tra các scope được định nghĩa trong model.
- Kiểm thử Controller
 - Kiểm tra một số controller chính với các chức năng quan trọng của ứng dụng.
 - Đảm bảo các hành động (actions) như index, show, create, update, destroy hoạt động đúng.

- Xác minh dữ liệu trả về từ các controller là chính xác.
 - Kiểm tra luồng điều hướng (redirect) sau khi thực hiện các hành động.
 - Đảm bảo các hành động xử lý đúng các trường hợp dữ liệu không hợp lệ.
- Kiểm thử Form trong View
- Đảm bảo các form được hiển thị đúng trên giao diện, bao gồm các trường dữ liệu (input fields), nút bấm (buttons), và các thông báo lỗi (error messages).
 - Xác minh các trường hợp nhập dữ liệu không hợp lệ (invalid inputs) và hành vi khi gửi form.
 - Kiểm tra tính năng hiển thị thông báo lỗi (error messages) nếu có.
 - Đảm bảo các giá trị hợp lệ được gửi lên server đúng cách.

3.5.2. Unit test

3.5.2.1. Mục tiêu

Đảm bảo các thành phần riêng lẻ của hệ thống (model, controller, service...) hoạt động đúng như mong đợi.

3.5.2.2. Công cụ

- RSpec: Framework kiểm thử phổ biến trong cộng đồng Ruby, cho phép viết các test dễ đọc và dễ bảo trì.
- SimpleCov: Thư viện đo lường mức độ bao phủ kiểm thử (test coverage), giúp phát hiện các phần mã chưa được kiểm thử.

3.5.2.3. Kết quả kiểm thử

a. Model

Mô phỏng tương tác người dùng trên trình duyệt để kiểm thử các chức năng toàn diện và giao diện người dùng (end-to-end).

3.5.3.2. Công cụ

Selenium WebDriver: Giao tiếp với trình duyệt thật như Chrome để chạy test tự động.

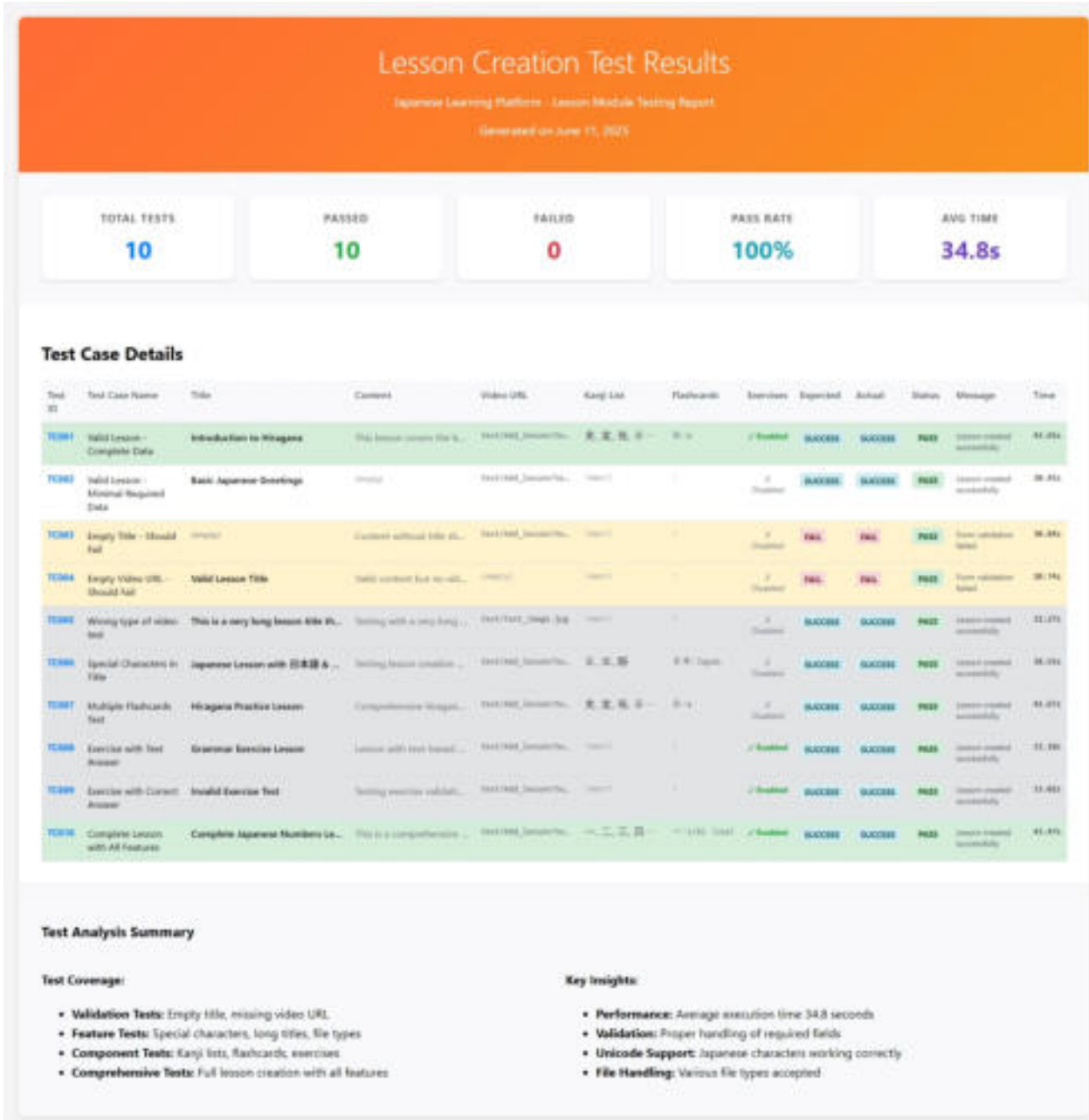
3.5.3.3. Kết quả kiểm thử

Các kịch bản kiểm thử tự động đã được thực hiện thành công trên trình duyệt thực tế. Các biểu mẫu quan trọng trong hệ thống như: tạo khóa học, tạo bài học, đăng nhập, đăng ký và tìm kiếm khóa học đều được kiểm thử và hoạt động ổn định. Các chức năng phản hồi đúng yêu cầu, giao diện người dùng hiển thị chính xác, không phát hiện lỗi trong quá trình tương tác.

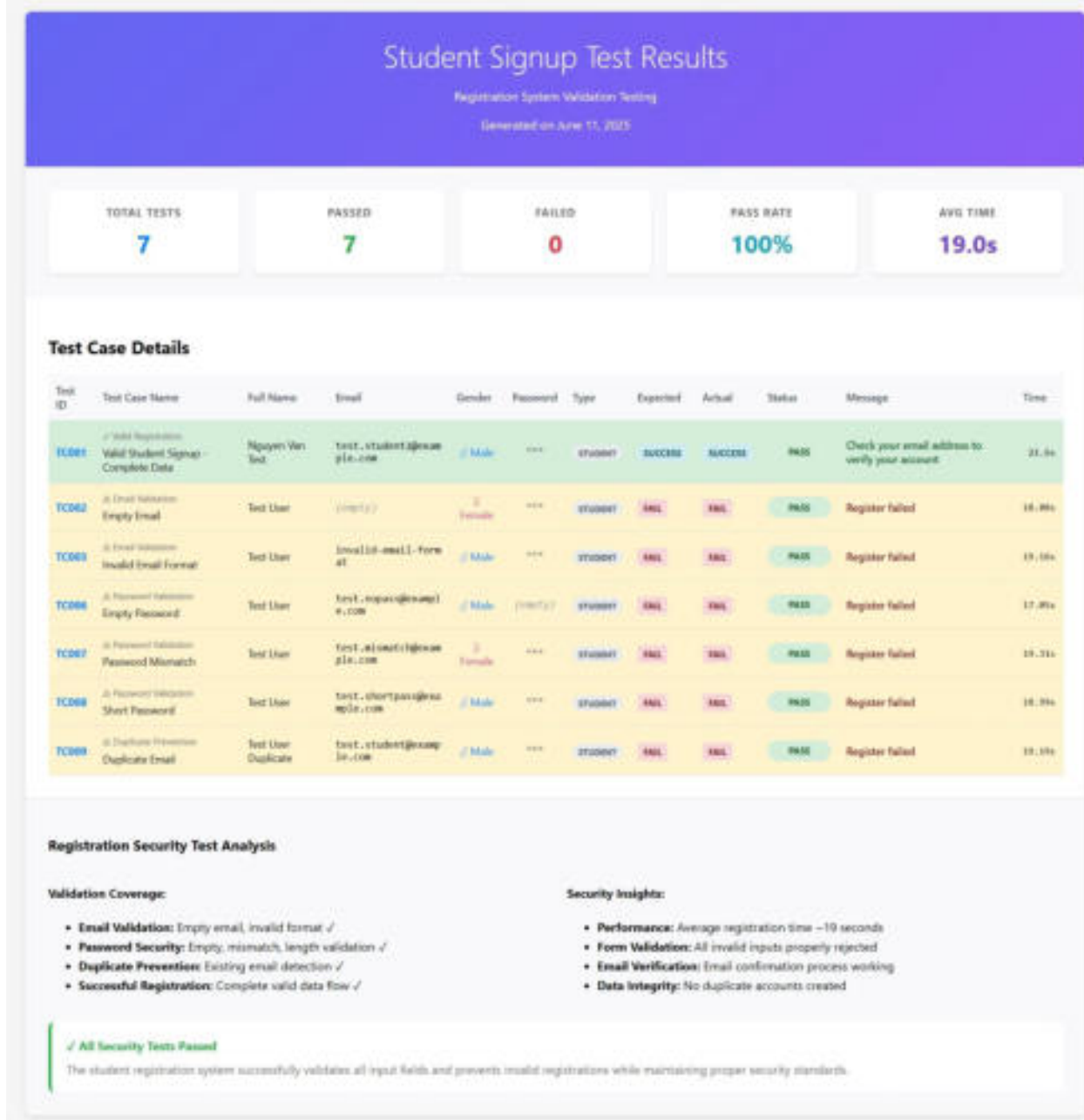
The screenshot displays a test results report titled "Course Creation Test Results" with a subtitle "Automated Testing Report - Generated on June 11, 2025". The summary section shows: TOTAL TESTS: 10, PASSED: 10, FAILED: 0, ERRORS: 0, and PASS RATE: 100.0%. Below this is a detailed table of test cases.

Test ID	Test Case Name	Title	Description	Level	Category	Price	Auto Accept	Expected	Actual	Status	Error Message	Time
TC001	Valid Course - N5 Level	Japanese For Absolute Beginner	A comprehensive course for b...	N5	Japanese Culture	1,000,000	✓	SUCCESS	SUCCESS	PASS	Course created successfully	45.82s
TC002	Valid Course - N3 Level	Intermediate Japanese Grammar	Master intermediate Japanese ...	N3	Japanese Culture	1,500,000	X	SUCCESS	SUCCESS	PASS	Course created successfully	46.91s
TC003	Free Course	Free Japanese Introduction	Free introductory course to Ja...	N5	Japanese Culture	0	✓	FAIL	FAIL	FAIL	Form validation failed - required fields missing	45.48s
TC004	Empty Title		Course with empty title should...	N3	Japanese Culture	1,000,000	✓	FAIL	FAIL	FAIL	Form validation failed - required fields missing	46.01s
TC005	Empty Description	Valid Title		N3	Japanese Culture	1,000,000	✓	FAIL	FAIL	FAIL	Form validation failed - required fields missing	44.73s
TC006	No Level Selected	Valid Title	Valid description for the course...		Japanese Culture	1,000,000	✓	FAIL	FAIL	FAIL	Form validation failed - required fields missing	42.97s
TC007	No Category Selected	Valid Title	Valid description for the course...	N3		1,000,000	✓	FAIL	FAIL	FAIL	Form validation failed - required fields missing	41.81s
TC008	Invalid Price - Negative	Valid Title	Valid description for the course...	N3	Japanese Culture	-100,000	✓	FAIL	FAIL	FAIL	Form validation failed - required fields missing	45.51s
TC009	Long Title Test	This is a very long course title that re...	Testing with a very long title L...	N2	Japanese Culture	2,000,000	X	SUCCESS	SUCCESS	PASS	Course created successfully	44.62s
TC010	Special Character Title	Japanese Course with 日本語 & Spec...	Testing course creation with L...	N1	Japanese Culture	1,000,000	✓	SUCCESS	SUCCESS	PASS	Course created successfully	45.78s

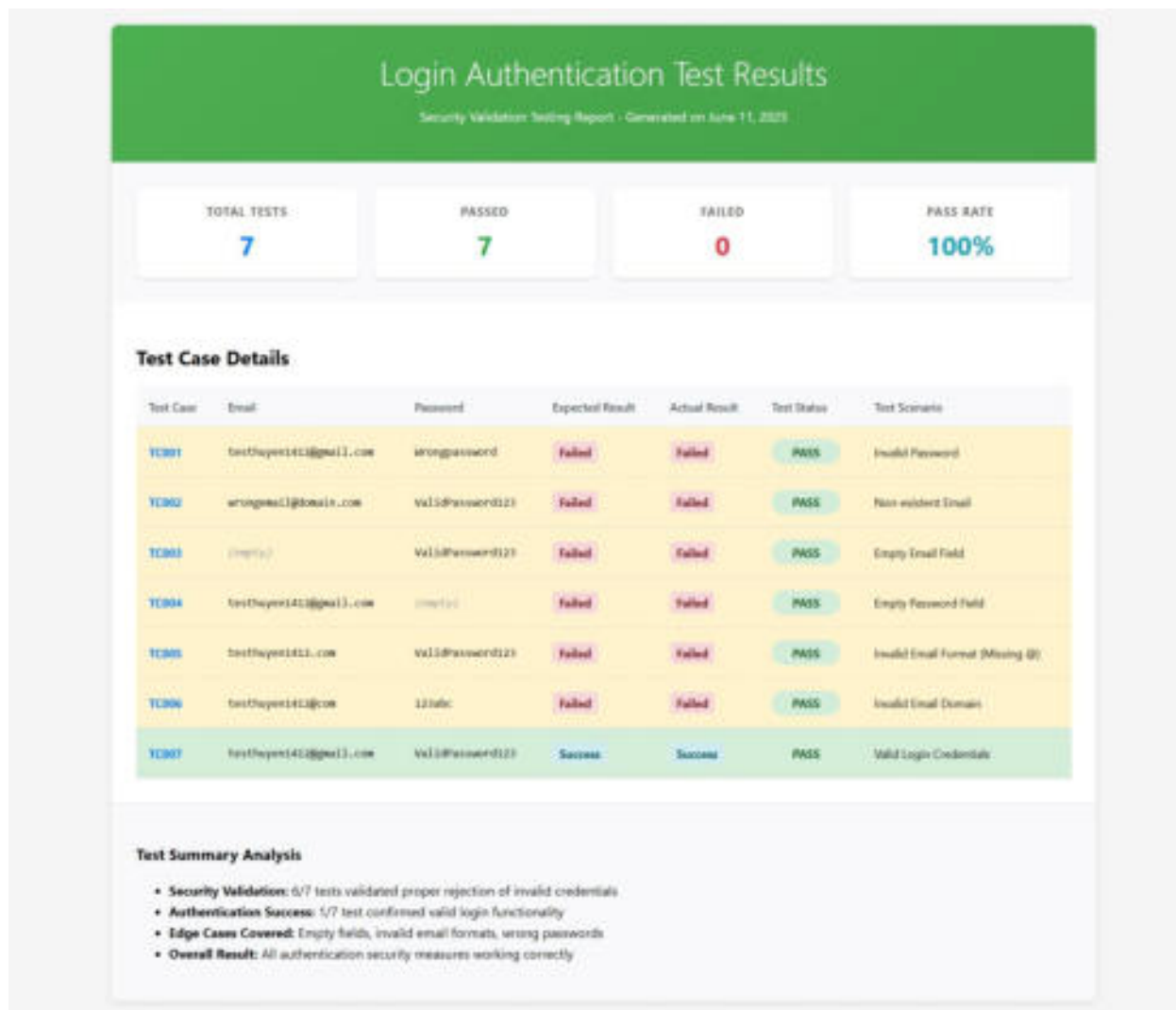
Hình- 3-60: Kết quả kiểm thử tự động trang tạo khóa học



Hình- 3-61: Kết quả kiểm thử tự động trang tạo bài học



Hình- 3-62: Kết quả kiểm thử tự động trang đăng kí



Hình- 3-63: Kết quả kiểm thử tự động trang đăng nhập

3.6. Kết luận chương 3

Chương 3 đã trình bày chi tiết quá trình triển khai hệ thống dựa trên thiết kế được trình bày ở chương 2, bao gồm cấu trúc tổng thể, lựa chọn công nghệ đến quá trình huấn luyện và triển khai mô hình AI. Bên cạnh đó, hệ thống web được phát triển với giao diện thân thiện, đáp ứng đầy đủ chức năng cho học viên, giáo viên và quản trị viên. Việc sử dụng Docker kết hợp với máy chủ ảo giúp đơn giản hóa quá trình triển khai, đảm bảo khả năng mở rộng và dễ dàng bảo trì hệ thống. Các bước kiểm thử, bao gồm kiểm thử đơn vị và kiểm thử tự động, đã xác nhận tính ổn định và độ tin cậy của hệ thống trước khi đưa vào sử dụng thực tế. Đây là nền tảng quan trọng để chuyển sang bước đánh giá hiệu quả và khả năng ứng dụng trong chương tiếp theo.

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Kết luận

Sau quá trình nghiên cứu và phát triển, hệ thống học tiếng Nhật đã được xây dựng và triển khai thành công, đáp ứng các mục tiêu ban đầu đề ra. Hệ thống không chỉ cung cấp nền tảng học tập đa dạng với video bài giảng, flashcard, bài luyện tập và nội dung theo cấp độ mà còn tích hợp các tính năng thông minh như nhận diện chữ Kanji viết tay, chấm điểm chữ viết tay Kanji. Những chức năng này góp phần hỗ trợ người học một cách hiệu quả, giúp cải thiện khả năng tự học và tăng cường tính tương tác trong quá trình học tập.

Về mặt kỹ thuật, dự án đã sử dụng các công nghệ hiện đại như Docker, Flask, Rails, Next.js cùng với các công cụ hỗ trợ kiểm thử tự động để đảm bảo chất lượng phần mềm. Việc áp dụng Docker trong quá trình triển khai đã giúp hệ thống dễ dàng đóng gói, triển khai đồng nhất trên nhiều môi trường và thuận tiện cho việc bảo trì, mở rộng trong tương lai. Hệ thống được triển khai trên nền tảng máy ảo Azure, mang lại môi trường chạy ổn định, khả năng mở rộng linh hoạt và hỗ trợ tốt cho việc vận hành hệ thống ở quy mô lớn.

Bên cạnh đó, các tính năng như dịch từ vựng và kiểm tra lỗi ngữ pháp tiếng Nhật đã được tích hợp thông qua các API bên thứ ba đáng tin cậy. Việc tích hợp này giúp mở rộng hệ thống, tăng cường chức năng, mang lại trải nghiệm học tập chất lượng và thuận tiện hơn cho người dùng.

Mô hình AI được xây dựng từ dữ liệu thực tế, qua các bước tiền xử lý, huấn luyện và đánh giá, cho thấy khả năng áp dụng trí tuệ nhân tạo vào bài toán giáo dục ngôn ngữ là hoàn toàn khả thi.

Các giao diện người dùng được thiết kế trực quan, phù hợp với từng vai trò trong hệ thống: học viên, giáo viên và quản trị viên. Các chức năng quản lý khóa học, tạo bài giảng, theo dõi tiến độ và thống kê hoạt động học tập được tích hợp đầy đủ, phục vụ hiệu quả cho cả việc dạy và học. Hệ thống cũng đã trải qua nhiều bước kiểm thử, từ kiểm thử đơn vị đến kiểm thử tự động, đảm bảo tính ổn định và độ tin cậy trước khi triển khai thực tế.

Tổng thể, sản phẩm đã đạt được sự hoàn thiện về mặt chức năng, kỹ thuật và tính ứng dụng, tạo nền tảng vững chắc cho việc phát triển thành một hệ thống học tiếng Nhật thông minh, linh hoạt và hiệu quả trong tương lai. Tuy nhiên, hệ thống vẫn còn nhiều tiềm năng

để cải tiến và mở rộng nhằm đáp ứng tốt hơn nhu cầu ngày càng đa dạng của người học, từ đó mở ra nhiều hướng phát triển trong tương lai.

Hướng phát triển

- Tích hợp cá nhân hóa trải nghiệm người học: Áp dụng trí tuệ nhân tạo để phân tích hành vi học tập, đưa ra gợi ý bài học, lộ trình học cá nhân hóa, cũng như thống kê và phản hồi phù hợp với từng người dùng.
- Tăng cường khả năng tương tác: Tích hợp thêm tính năng chat với giáo viên hoặc chatbot hỗ trợ học tập, diễn đàn trao đổi giữa người học, hoặc tổ chức các lớp học trực tuyến tương tác.
- Triển khai trên hạ tầng cloud mạnh hơn: Nâng cấp môi trường vận hành sang các dịch vụ Azure chuyên sâu như Azure Kubernetes Service (AKS), Azure Cognitive Services... để đảm bảo khả năng mở rộng, xử lý AI hiệu quả và dễ dàng tích hợp dịch vụ bên ngoài.
- Phát triển ứng dụng di động: Xây dựng phiên bản mobile (iOS/Android) đồng bộ với hệ thống web, giúp người dùng học tập tiện lợi hơn mọi lúc, mọi nơi.
- Tối ưu hiệu suất hệ thống: Tăng tốc độ phản hồi của giao diện và backend thông qua việc cải thiện cơ sở dữ liệu, cache hóa dữ liệu với Redis, và tối ưu pipeline AI, đảm bảo trải nghiệm học tập mượt mà ngay cả khi số lượng người dùng tăng cao.
- Tăng cường bảo mật: Áp dụng các biện pháp bảo mật nâng cao như xác thực hai yếu tố (2FA), kiểm soát truy cập theo vai trò (RBAC), mã hóa dữ liệu nhạy cảm và thường xuyên kiểm thử bảo mật để đảm bảo an toàn cho người dùng.
- Cập nhật giao diện người dùng (UI): Cải thiện thiết kế giao diện theo hướng hiện đại, thân thiện và nhất quán trên các thiết bị, giúp người học dễ sử dụng và tăng khả năng tương tác.

TÀI LIỆU THAM KHẢO

- [1] A. G. Howard and M. Zhu, “MobileNets: Open-Source Models for Efficient On-Device Vision,” Google Research Blog, June 14, 2017. [Online]. Available: <https://research.google/blog/mobilenets-open-source-models-for-efficient-on-device-vision/>. [Accessed: Jun. 3, 2025].
- [2] Pham D. Khanh, “Bài 48 – MobileNet model,” *Khoa học dữ liệu – Khanh's blog*, 19 Sep. 2020. [Online]. Available: <https://phamdinhhkhanh.github.io/2020/09/19/MobileNet.html>. [Accessed: Jun. 3, 2025].
- [3] M. Sandler and A. Howard, “MobileNetV2: The Next Generation of On-Device Computer Vision Networks,” *Google AI Blog*, Apr. 3, 2018. [Online]. Available: <https://ai.googleblog.com/2018/04/mobilenetv2-next-generation-of-on.html>
- [4] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” arXiv preprint arXiv:1801.04381, 2018. [Online]. Available: <https://arxiv.org/abs/1801.04381>. [Accessed: Jun. 3, 2025].
- [5] A. S. Tanenbaum and D. J. Wetherall, *Computer Networks*, 5th ed. Pearson, 2011.
- [6] Vercel, “Next.js Documentation,” *Next.js by Vercel*, 2024. [Online]. Available: <https://nextjs.org/docs>
- [7] Vercel, “Static Generation and Server-side Rendering,” *Next.js Documentation*, 2024. [Online]. Available: <https://nextjs.org/docs/pages/building-your-application/rendering/static-site-generation>
- [8] S. Devkar, “Next.js: The React Framework for Production,” *LinkedIn*, Jan. 8, 2025. [Online]. Available: <https://www.linkedin.com/pulse/nextjs-react-framework-production-suryakant-devkar-c3upf/>
- [9] Cloudinary, “Cloudinary Documentation,” *Cloudinary Docs*, 2024. [Online]. Available: <https://cloudinary.com/documentation>
- [10] Cloudinary, “Media Optimization for Web and Mobile,” *Cloudinary Website*, 2024. [Online]. Available: https://cloudinary.com/documentation/media_optimization

- [11] Tailwind Labs, “Tailwind CSS Documentation,” *Tailwind CSS*, 2024. [Online]. Available: <https://tailwindcss.com/docs>
- [12] D. Heinemeier Hansson, “Ruby on Rails Guides,” *Ruby on Rails Official Documentation*, 2024. [Online]. Available: <https://guides.rubyonrails.org/>
- [13] “Rails API mode,” *Ruby on Rails Guides*, 2024. [Online]. Available: https://guides.rubyonrails.org/api_app.html
- [14] A. Ronacher, “Flask Documentation,” *Pallets Projects*, 2024. [Online]. Available: <https://flask.palletsprojects.com/>
- [15] M. Grinberg, *Flask Web Development: Developing Web Applications with Python*, 2nd ed. O'Reilly Media, 2018.
- [16] Oracle Corporation, “MySQL :: MySQL Documentation,” *MySQL*, 2024. [Online]. Available: <https://dev.mysql.com/doc/>
- [17] Docker Inc., “What is Docker?,” *Docker Docs*, 2024. [Online]. Available: <https://docs.docker.com/get-started/overview/>
- [18] Microsoft, "Microsoft Azure – Cloud Computing Platform," *Microsoft Learn*, [Online]. Available: <https://learn.microsoft.com/en-us/azure/?product=popular>. [Accessed: Jun. 3, 2025].
- [19] G. Myers, *The Art of Software Testing*, 3rd ed. Hoboken, NJ, USA: Wiley, 2011.
- [20] S. Freeman and N. Pryce, *Growing Object-Oriented Software, Guided by Tests*, Addison-Wesley Professional, 2010.
- [21] "RSpec Documentation," *RSpec*, [Online]. Available: <https://rspec.info/documentation/>. [Accessed: Jun. 3, 2025].
- [22] "SeleniumHQ Browser Automation," *Selenium*, [Online]. Available: <https://www.selenium.dev/documentation/>. [Accessed: Jun. 3, 2025].

PHỤ LỤC 1

PHỤ LỤC 2