TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI

KHOA CÔNG NGHỆ THÔNG TIN

---------o0o---------

A blue and yellow logo

Description automatically generated

BÁO CÁO BÀI TẬP LỚN

PROJECT 1

NHÓM 3

Đề tài: Xây dựng ứng dụng học

ngoại ngữ trên nền tảng di động

**Giảng viên hướng dẫn: Hoàng Văn Thông**

**Sinh viên thực hiện:**

|  |  |  |
| --- | --- | --- |
| STT | Họ và tên | Mã sinh viên |
| 1 | Vũ Bảo Lâm | 211241205 |
| 2 | Lê Hữu Chung | 211211256 |
| 3 | Trần Công Chiến | 211201878 |
| 4 | Đỗ Tuấn Khải | 211210438 |
| 5 | Nguyễn Xuân Bách | 211243054 |

**Lớp: Công nghệ thông tin 1**

**Khóa: 62**

**Hà Nội -2024**

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI

KHOA CÔNG NGHỆ THÔNG TIN

---------o0o---------

A blue and yellow logo

Description automatically generated

BÁO CÁO BÀI TẬP LỚN

PROJECT 1

NHÓM 3

Đề tài: Xây dựng ứng dụng học

ngoại ngữ trên nền tảng di động

Giảng viên hướng dẫn: Hoàng Văn Thông

Sinh viên thực hiện:

|  |  |  |
| --- | --- | --- |
| STT | Họ và tên | Mã sinh viên |
| 1 | Vũ Bảo Lâm | 211241205 |
| 2 | Lê Hữu Chung | 211211256 |
| 3 | Trần Công Chiến | 211201878 |
| 4 | Đỗ Tuấn Khải | 211210438 |
| 5 | Nguyễn Xuân Bách | 211243054 |

Lớp: Công nghệ thông tin 1

Khóa: 62

Hà Nội – 2024

LỜI NÓI ĐẦU

Trong bối cảnh toàn cầu hóa và sự phát triển vượt bậc của công nghệ thông tin, việc học ngoại ngữ, đặc biệt là tiếng Anh, đã trở thành một nhu cầu thiết yếu không chỉ trong môi trường học tập mà còn trong công việc và cuộc sống. Để đáp ứng nhu cầu này, các ứng dụng học tiếng Anh đã và đang đóng vai trò quan trọng trong việc hỗ trợ người học tiếp cận kiến thức một cách tiện lợi, hiệu quả và linh hoạt.

Ứng dụng **ENG2UTC** được xây dựng nhằm mang đến một công cụ học tiếng Anh toàn diện, kết hợp các yếu tố từ điển, học từ vựng, bài học ngữ pháp, và các bài kiểm tra được thiết kế theo chuẩn CEFR (Cambridge English Framework). Với mục tiêu giúp người học cải thiện trình độ tiếng Anh từ cơ bản đến nâng cao, ứng dụng này cung cấp một hệ sinh thái học tập thông minh, tương tác và thân thiện với người dùng.

Báo cáo này trình bày quá trình phát triển ứng dụng ENG2UTC, từ việc phân tích yêu cầu, thiết kế giao diện, đến triển khai chức năng và kiểm thử. Đồng thời, chúng em cũng nêu bật các thách thức gặp phải trong quá trình phát triển, các giải pháp đã thực hiện, và tiềm năng mở rộng ứng dụng trong tương lai.

Chúng em xin gửi lời cảm ơn đến giảng viên hướng dẫn, người đã hỗ trợ, và đội ngũ phát triển đã đóng góp ý tưởng cũng như công sức vào thành công của dự án. Hy vọng rằng ứng dụng ENG2UTC sẽ là một bước tiến nhỏ nhưng có ý nghĩa trong việc hỗ trợ người học tiếng Anh trên hành trình nâng cao kỹ năng ngôn ngữ.

Chúng em xin chân thành cảm ơn!

**Nhóm sinh viên thực hiện**

BẢNG PHÂN CÔNG CÔNG VIỆC

|  |  |  |
| --- | --- | --- |
| STT | Họ và Tên | Công việc |
| 1 | Trần Công Chiến | Xây dựng chức năng thống kê kết quả học tập, bài test  Xây dựng, thiết kế cơ sở dữ liệu  Xây dựng API |
| 2 | Lê Hữu Chung | Xây dựng chức năng bài tập từ vựng  Xây dựng chức năng làm bài kiểm tra  Làm Slide thuyết trình |
| 3 | Vũ Bảo Lâm | Thiết kế giao diện ứng dụng  Xây dựng chức năng đăng nhập, đăng ký  Xây dựng chức năng tra cứu từ điển  Xây dựng chức năng dịch văn bản  Xây dựng chức năng cài đặt Setting  Xây dựng chức năng đánh giá Rating |
| 4 | Đỗ Tuấn Khải | Xây dựng API  Xây dựng kiểm thử |
| 5 | Nguyễn Xuân Bách | Xây dựng chức năng học từ vựng  Hỗ trợ thiết kế và chỉnh sửa giao diện ứng dụng |

MỤC LỤC

[CHƯƠNG 1: SƠ LƯỢC VỀ ĐỀ TÀI 1](#_Toc184123294)

[1.1. Giới thiệu 1](#_Toc184123295)

[1.1.1. Lý do chọn đề tài 1](#_Toc184123296)

[1.2. Mục tiêu chọn đề tài 1](#_Toc184123297)

[1.3. Khảo sát đối tượng có nhu cầu học Tiếng Anh và giáo viên giảng dạy Tiếng Anh 2](#_Toc184123298)

[1.3.1. Đối với người học 2](#_Toc184123299)

[1.3.2. Đối với người giảng dạy Tiếng Anh 3](#_Toc184123300)

[1.4. Khảo sát các ứng dụng học tiếng anh 4](#_Toc184123301)

[1.5. Kết luận sau khảo sát 6](#_Toc184123302)

[CHƯƠNG 2: CÔNG NGHỆ SỬ DỤNG 7](#_Toc184123303)

[2.1. Hệ điều hành di động và nền tảng Android 7](#_Toc184123304)

[2.2. Ngôn ngữ lập trình Java và Python 8](#_Toc184123305)

[2.2.1. Công nghệ Java là gì? 8](#_Toc184123306)

[2.2.2. Lịch sử ra đời của Java 8](#_Toc184123307)

[2.2.3. Đặc điểm và các thành phần chính của Java 8](#_Toc184123308)

[2.2.4. Kiến thức cần thiết để làm việc với Java 9](#_Toc184123309)

[2.2.5. Ưu và nhược điểm của Java 10](#_Toc184123310)

[2.2.6. Python và lịch sử ra đời 10](#_Toc184123311)

[2.2.7. Đặc điểm và các thành phần chính của Python 10](#_Toc184123312)

[2.2.8. Kiến thức cần thiết để làm việc với Python 11](#_Toc184123313)

[2.3. Môi trường lập trình Android Studio 12](#_Toc184123314)

[2.3.1. Android Studio 12](#_Toc184123315)

[2.3.2. Kiến trúc của nền tảng Android Studio 12](#_Toc184123316)

[2.3.3. Bộ công cụ phát triển phần mềm Android Android SDK 14](#_Toc184123317)

[2.4. Firebase 14](#_Toc184123318)

[2.5. FastAPI 15](#_Toc184123319)

[CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG 17](#_Toc184123320)

[3.1. Mô tả tổng quan hệ thống 17](#_Toc184123321)

[3.1.1. Tổng quan hệ thống 17](#_Toc184123322)

[3.1.2. Các yêu cầu của hệ thống 17](#_Toc184123323)

[3.1.3. Các chức năng của hệ thống 18](#_Toc184123324)

[3.2. Phân tích các chức năng của hệ thống 19](#_Toc184123325)

[3.2.1. Sơ đồ phân rã chức năng 19](#_Toc184123326)

[3.2.2. Các biểu đồ use-case 20](#_Toc184123327)

[3.3. Phân tích thiết kế cơ sở dữ liệu 29](#_Toc184123328)

[3.3.1. Phân tích cơ sở dữ liệu 29](#_Toc184123329)

[3.3.2. Thiết kế cơ sở dữ liệu 30](#_Toc184123330)

[3.3.3. Sơ đồ quan hệ thực thể 37](#_Toc184123331)

[3.4. Danh sách API 38](#_Toc184123332)

[3.4.1. Danh sách API GET 39](#_Toc184123333)

[3.4.2. Danh sách API POST 40](#_Toc184123334)

[3.4.3. Danh sách API PUT 41](#_Toc184123335)

[CHƯƠNG 4: TRIỂN KHAI VÀ KẾT QUẢ 42](#_Toc184123336)

[4.1. Triển khai 42](#_Toc184123337)

[4.1.1. Công cụ phát triển 42](#_Toc184123338)

[4.1.2. Hướng dẫn cài đặt 42](#_Toc184123339)

[4.2. Giao diện ứng dụng 43](#_Toc184123340)

[CHƯƠNG 5: KIỂM THỬ 50](#_Toc184123341)

[5.1. Giới Thiệu Về Việc Kiểm Thử Trong Dự Án 50](#_Toc184123342)

[5.2. Trường hợp kiểm thử (Test Case) 50](#_Toc184123343)

[5.2.1. Chức năng đăng ký 50](#_Toc184123344)

[5.2.2. Chức năng đăng nhập 51](#_Toc184123345)

[5.2.3. Tra cứu từ điển, dịch từ vựng 51](#_Toc184123346)

[5.2.4. Học từ vựng 51](#_Toc184123347)

[5.2.5. Làm bài kiểm tra 52](#_Toc184123348)

[CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 54](#_Toc184123349)

[6.1. Kết luận 54](#_Toc184123350)

[6.2. Hướng phát triển 55](#_Toc184123351)

[TÀI LIỆU THAM KHẢO 56](#_Toc184123352)

MỤC LỤC HÌNH ẢNH

[CHƯƠNG 1: SƠ LƯỢC VỀ ĐỀ TÀI 1](#_Toc184122275)

[Hình 1‑1: Giao diện trang chủ ứng dụng Dunno Dict 2](#_Toc184122276)

[Hình 1‑2: Giao diện trang chủ ứng dụng TFlat 2](#_Toc184122277)

[Hình 1‑3: Giao diện dịch văn bản của ứng dụng Dunno Dict 2](#_Toc184122278)

[Hình 1‑4: Giao diện đề thi của Dunno Dict 2](#_Toc184122279)

[CHƯƠNG 2: CÔNG NGHỆ SỬ DỤNG 4](#_Toc184122280)

[Hình 2‑1: Hình icon của Android 4](#_Toc184122281)

[Hình 2‑2: Icon của môi trường Android Studio 9](#_Toc184122282)

[Hình 2‑3: Kiến trúc một ứng dụng Android 10](#_Toc184122283)

[Hình 2‑4: Hình ảnh Firebase 11](#_Toc184122284)

[Hình 2‑5: Hình ảnh FastAPI 12](#_Toc184122285)

[CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG 14](#_Toc184122286)

[Hình 3‑1: Biểu đồ use-case đăng kí 18](#_Toc184122287)

[Hình 3‑2: Sơ đồ luồng chức năng đăng kí 19](#_Toc184122288)

[Hình 3‑3: Biểu đồ use-case đăng nhập 20](#_Toc184122289)

[Hình 3‑4: Sơ đồ luồng hoạt động chức năng đăng nhập 21](#_Toc184122290)

[Hình 3‑5: Biểu đồ use-case dịch/tra cứu từ điển 21](#_Toc184122291)

[Hình 3‑6: Sơ đồ luồng hoạt động chức năng tra cứu từ điển 22](#_Toc184122292)

[Hình 3‑7: Biểu đồ use-case làm bài kiểm tra 23](#_Toc184122293)

[Hình 3‑8: Sơ đồ luồng hoạt động chức năng làm bài kiểm tra 24](#_Toc184122294)

[Hình 3-9: Biểu đồ use-case học từ vựng 25](#_Toc184122295)

[Hình 3-10: Sơ đồ luồng hoạt động chức năng học từ vựng 26](#_Toc184122296)

[Hình 3-11: Sơ đồ thực thể quan hệ 34](#_Toc184122297)

[Hình 3-12: Danh sách API 35](#_Toc184122298)

[CHƯƠNG 4: TRIỂN KHAI VÀ KẾT QUẢ 39](#_Toc184122299)

[Hình 4‑1: Trang Intro 40](#_Toc184122300)

[Hình 4‑2: Trang đăng kí 40](#_Toc184122301)

[Hình 4‑3: Trang đăng nhập 40](#_Toc184122302)

[Hình 4‑4: Trang chủ chính 40](#_Toc184122303)

[Hình 4‑5: Giao diện cài đặt 41](#_Toc184122304)

[Hình 4‑6: Đổi màu giao diện 41](#_Toc184122305)

[Hình 4‑7: Đổi ngôn ngữ hiển thị 41](#_Toc184122306)

[Hình 4‑8: Giao diện từ vựng 41](#_Toc184122307)

[Hình 4‑9: Giao diện từ vựng theo cấp độ 42](#_Toc184122308)

[Hình 4‑10: Giao diện bài học theo chủ đề 42](#_Toc184122309)

[Hình 4‑11: Giao diện danh sách từ vựng theo chủ đề 42](#_Toc184122310)

[Hình 4‑12: Giao diện học từ vựng 42](#_Toc184122311)

[Hình 4‑13: Sắp xếp đúng từ vựng 43](#_Toc184122312)

[Hình 4‑14: Sắp xếp sai từ vựng 43](#_Toc184122313)

[Hình 4‑15: Giao diện ngữ pháp 43](#_Toc184122314)

[Hình 4‑16: Giao diện danh sách các thì 43](#_Toc184122315)

[Hình 4‑17: Giao diện lý thuyết ngữ pháp 44](#_Toc184122316)

[Hình 4‑18: Giao diện dịch văn bản 44](#_Toc184122317)

[Hình 4‑19: Giao diện đề kiểm tra 44](#_Toc184122318)

[Hình 4‑20: Xem điểm các lần làm kiểm tra 44](#_Toc184122319)

[Hình 4‑21: Giao diện bài kiểm tra 45](#_Toc184122320)

[Hình 4‑22: Giao diện hiển thị số câu hỏi 45](#_Toc184122321)

[Hình 4‑23: Giao diện chấm điểm kiểm tra 45](#_Toc184122322)

[Hình 4‑24: Giao diện thống kê 45](#_Toc184122323)

[Hình 4‑25: Giao diện từ được tra cứu 46](#_Toc184122324)

[Hình 4‑26: Search Box tra cứu từ điển 46](#_Toc184122325)

[Hình 4‑27: Giao diện đánh giá ứng dựng 46](#_Toc184122326)

[CHƯƠNG 5: KIỂM THỬ 47](#_Toc184122327)

[Hình 5-1: Testcase Đăng ký 48](#_Toc184122328)

[Hình 5-2: Testcase Đăng nhập/Đăng xuất 48](#_Toc184122329)

[Hình 5-3: Testcase tra từ điển 48](#_Toc184122330)

[Hình 5-4: Testcase học từ vựng 49](#_Toc184122331)

[Hình 5-5: Testcase làm bài kiểm tra 49](#_Toc184122332)

[CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 51](#_Toc184122333)

# 

# SƠ LƯỢC VỀ ĐỀ TÀI

## Giới thiệu

Chương 1 này sẽ trình bày một cái nhìn tổng quan về các vấn đề liên quan đến đề tài nghiên cứu, đồng thời làm rõ lý do đã chọn đề tài cụ thể. Bên cạnh đó, đề ra mục tiêu và phạm vi nghiên cứu của dự án, mô tả phương pháp nghiên cứu mà đề tài sẽ sử dụng và xác định bố cục đề tài.

### Lý do chọn đề tài

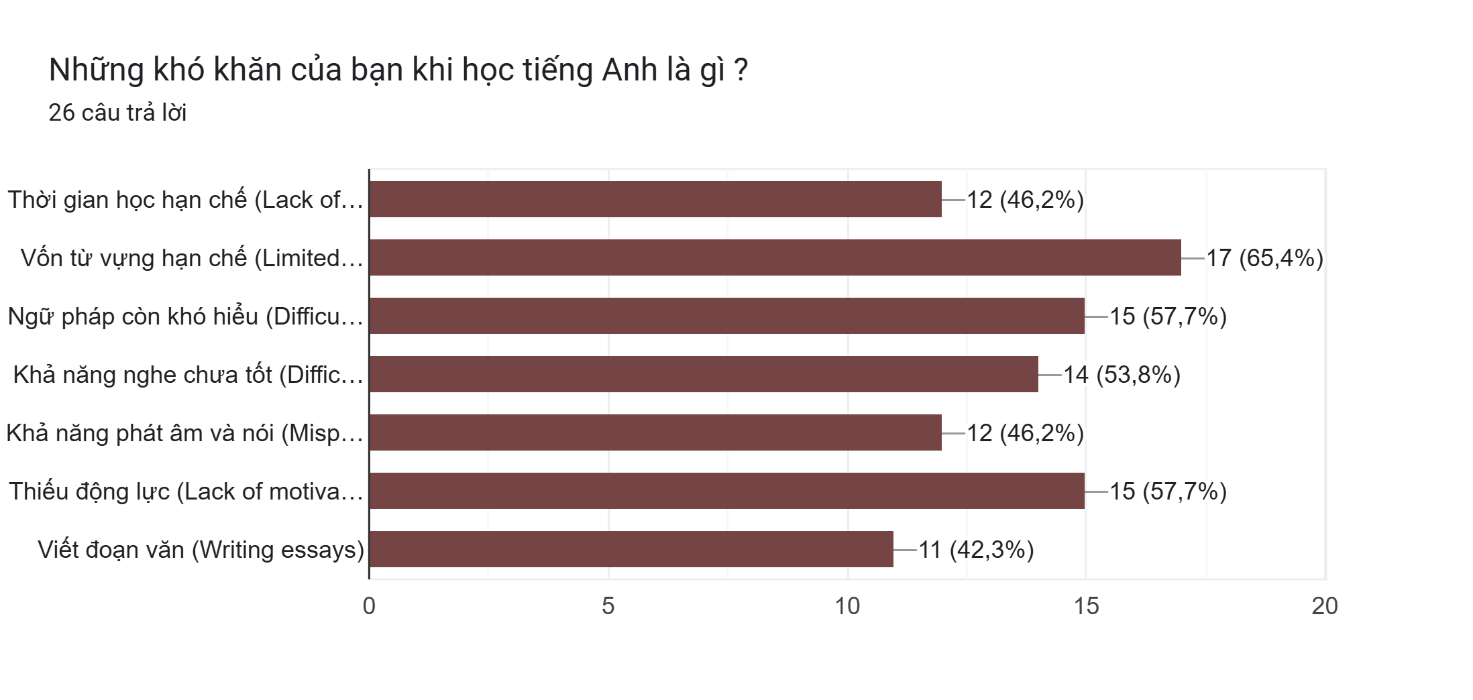
Tiếng Anh là một trong những ngôn ngữ thông dụng nhất trên thế giới, là chìa khóa để mở cửa thế giới. Có thể nói, trên rất nhiều lĩnh vực, tiếng Anh đã chinh phục tuyệt đối. Tầm quan trọng của tiếng Anh đang ngày càng được khẳng định. Ngày nay, tiếng Anh không còn chỉ dừng lại trong phạm vi một môn học mà dần trở thành công cụ giao tiếp của nhiều bạn trẻ. Hơn nữa, trong bối cảnh thời mở cửa như hiện nay, ngày càng có nhiều doanh nghiệp nước ngoài đầu tư sang thị trường Việt Nam, mang lại rất nhiều cơ hội việc làm cho các lao động trẻ. Việc học tiếng Anh rất cần thiết, tuy nhiên cần phải có phương pháp học phù hợp để mang lại hiệu quả tốt nhất, Thông thường, mọi người thường tìm các trung tâm để học. Việc này này sẽ tiêu tốn khá nhiều thời gian và tiền bạc, nhất là đối với những người ít có thời gian rãnh rỗi. Tuy nhiên, trong thời kỳ công nghệ phát triển, việc sở hữu một chiếc điện thoại thông minh là không ngoài khả năng của nhiều người.

## Mục tiêu chọn đề tài

* Mục tiêu chung: Xây dựng một ứng dụng học tiếng anh chất lượng, đáp ứng được nhu cầu của người dùng và có thể học được mọi lúc, mọi nơi.
* Mục tiêu cụ thể:
  + Xây dựng ứng dụng có giao diện thân thiện, dễ sử dụng.
  + Ứng dụng bao gồm các câu trắc nghiệm của nhiều cấp độ.
* Phạm vi nghiên cứu:
  + Ứng dụng sẽ được xây dựng và triển khai điện thoạt di động.
  + Tìm hiểu và đưa ra các ưu điểm và nhược điểm, đánh giá mô hình và nội dung tại các ứng dụng học tiếng anh hiện có.
  + Nghiên cứu lập trình mobile app bằng Android Studio, quản lý cơ sở dữ liệu Firebase.
* Đối tượng nghiên cứu:
  + Những người có nhu cầu học tiếng Anh như học sinh sinh viên
  + Giáo viên đang giảng dạy tiếng Anh

## Khảo sát đối tượng có nhu cầu học Tiếng Anh và giáo viên giảng dạy Tiếng Anh

### Đối với người học



Biểu đồ câu trả lời của biểu mẫu. Tên câu hỏi: Bạn mong muốn cải thiện kỹ năng nào nhất trong tiếng Anh ?
. Số lượng câu trả lời: 26 câu trả lời.

Biểu đồ câu trả lời của biểu mẫu. Tên câu hỏi: Bạn mong đợi điều gì từ một ứng dụng học tiếng Anh ?  
. Số lượng câu trả lời: 26 câu trả lời.

### Đối với người giảng dạy Tiếng Anh

Biểu đồ câu trả lời của biểu mẫu. Tên câu hỏi: Theo anh/chị, đâu là những yếu tố quan trọng nhất trong một bài học ngoại ngữ?
. Số lượng câu trả lời: 7 câu trả lời.

Biểu đồ câu trả lời của biểu mẫu. Tên câu hỏi: Nếu có, anh/chị sử dụng công cụ nào?  
. Số lượng câu trả lời: 7 câu trả lời.

Biểu đồ câu trả lời của biểu mẫu. Tên câu hỏi: Anh/chị thường sử dụng những loại bài kiểm tra nào 
. Số lượng câu trả lời: 7 câu trả lời.

## Khảo sát các ứng dụng học tiếng anh

- Khảo sát các ứng dụng học tiếng anh: DuoLingo, Cake, Tflat.

- Để có cái nhìn tổng quan về các ứng dụng học tiếng anh. Nội dung khảo sát bao gồm các khía cạnh sau:

✓ Giao diện thân thiện, dễ sử dụng: Giao diện ứng dung cần được thiết kế đơn giản, dễ nhìn, dễ thao tác để người dùng có thể học tiếng anh thoải mái.

✓ Chủ đề học phong phú: Ứng dụng sẽ sử dùng trắc nghiệm để người dùng làm và học, do đó có thể dễ dàng thêm các chủ đề khác nhau để người dùng có hứng thú. Các công nghệ được sử dụng vào ứng dụng học tiếng Anh.

|  |  |
| --- | --- |
| Hình 1‑2: Giao diện trang chủ ứng dụng Dunno Dict | Hình 1‑3: Giao diện trang chủ ứng dụng TFlat |
| Hình 1‑4: Giao diện dịch văn bản của ứng dụng Dunno Dict | Hình 1‑5: Giao diện đề thi của Dunno Dict |

## Kết luận sau khảo sát

Dựa trên kết quả khảo sát, nhóm chúng em đã thống nhất các chức năng quan trọng mà hệ thống sẽ triển khai:

* Chức năng đăng ký đăng nhập
* Chức năng học từ vựng và ngữ pháp
* Chức năng làm bài kiểm tra tích hợp các kĩ năng
* Chức năng tra cứu từ vựng
* Chức năng dịch văn bản song ngữ Anh-Việt và Việt-Anh
* Chức năng thống kê tiến trình người học

# CÔNG NGHỆ SỬ DỤNG

## Hệ điều hành di động và nền tảng Android

Thị trường điện thoại thông minh hiện nay đang chứng kiến sự cạnh tranh của hai nền tảng thiết bị lớn là ***Iphone*** với hệ điều hành ***iOS*** của hãng điện tử Mỹ Apple và điện thoại sử dụng hệ điều hành ***Android*** của Tập đoàn ***Google***, bên cạnh đó có rất nhiều những nền tảng khác như Window phone của Microsoft, và một số nền tảng cũ khác đã từng xuất hiện như Symbian của Nokia, BlackBerry OS của Research In Motion (RIM)…Iphone đã sớm tạo vị thế của mình trên thị trường nhờ những ưu điểm về thiết kế tinh tế, giao diện đơn giản mà bóng bảy, xây dựng được trải nghiệm người dùng tốt. Tuy nhiên, những chiếc điện thoại chạy hệ điều hành Android với sức mạnh cấu hình đa dạng, chức năng phong phú và giá cả hợp lý, phù hợp nhu cầu người sử dụng mới là những thiết bị có lượng người sử dụng phong phú, đa dạng nhất.

A green logo with black text

Description automatically generated

Hình 2‑1: Hình icon của Android

Hệ điều hành Android là một hệ điều hành di động phổ biến nhất trên thế giới, được phát triển bởi Google. Dưới đây là một số thông tin cơ bản về hệ điều hành Android:Lịch sử phát triển: Android được phát triển từ nền tảng Android, Inc., một công ty được thành lập vào năm 2003 và sau đó được Google mua lại vào năm 2005. Phiên bản đầu tiên của Android được phát hành vào năm 2008, và kể từ đó, hệ điều hành này đã trải qua nhiều phiên bản cập nhật, với tên mã dự án được đặt theo tên các loại kẹo, bánh ngọt, và các từ khác liên quan đến thế giới của kẹo. Mã nguồn mở: Android là một hệ điều hành mã nguồn mở, điều này có nghĩa là mã nguồn của nó có thể được sửa đổi và phát triển bởi cộng đồng lập trình viên trên toàn thế giới. Nền tảng mã nguồn mở này đã giúp thúc đẩy sự phát triển nhanh chóng và sáng tạo của các ứng dụng di động. Ngôn ngữ lập trình chính: Trong quá trình phát triển ứng dụng Android, Java đã được sử dụng làm ngôn ngữ lập trình chính. Tuy nhiên, từ năm 2017, Kotlin đã trở thành ngôn ngữ lập trình chính thức được Google khuyến khích sử dụng, và hiện nay, Kotlin đã trở thành lựa chọn phổ biến cho việc phát triển ứng dụng Android. Đa dạng thiết bị: Android là một hệ điều hành phổ biến trên nhiều loại thiết bị di động khác nhau, từ điện thoại thông minh và máy tính bảng đến đồng hồ thông minh, TV thông minh và thiết bị nhúng. Cửa hàng ứng dụng: Google Play Store là cửa hàng ứng dụng chính thức của Android, nơi người dùng có thể tải xuống và cài đặt các ứng dụng và trò chơi, cũng như nội dung số khác như sách, phim và âm nhạc. Bảo mật và cập nhật: Google cung cấp các bản cập nhật bảo mật định kỳ cho hệ điều hành Android để bảo vệ người dùng khỏi các mối đe dọa mạng và lỗ hổng bảo mật. Đối với các thiết bị của một số nhà sản xuất, cập nhật hệ điều hành Android có thể trễ do quá trình tùy chỉnh của họ.

## Ngôn ngữ lập trình Java và Python

### Công nghệ Java là gì?

Java là một ngôn ngữ lập trình và nền tảng (platform) được phát triển bởi Sun Microsystems (hiện nay thuộc Oracle) vào năm 1995. Java là một ngôn ngữ lập trình hướng đối tượng, đơn giản, mạnh mẽ và dễ học, được thiết kế với mục đích "viết một lần, chạy ở mọi nơi" (Write Once, Run Anywhere - WORA). Điều này có nghĩa là code Java có thể chạy trên bất kỳ nền tảng nào mà không cần phải biên dịch lại, nhờ môi trường thực thi Java Virtual Machine (JVM).

### Lịch sử ra đời của Java

Java được phát triển vào năm 1991 bởi James Gosling và nhóm kỹ sư tại Sun Microsystems. Ban đầu, Java có tên là Oak, nhưng sau đó được đổi thành Java, lấy cảm hứng từ hạt cà phê nổi tiếng tại đảo Java (Indonesia). Mục tiêu ban đầu của Java là dành cho các thiết bị điện tử nhỏ gọn, nhưng sau đó đã nhanh chóng phát triển và được sử dụng rộng rãi trên các máy tính và hệ thống lớn hơn.

Các mốc thời gian quan trọng:

* 1991: James Gosling bắt đầu dự án Java, ban đầu có tên là Oak.
* 1995: Java chính thức ra mắt với trình duyệt web HotJava có thể chạy applet (một dạng ứng dụng nhúng trong trình duyệt).
* 1996: JDK (Java Development Kit) 1.0 được phát hành, cung cấp các công cụ cần thiết để phát triển các ứng dụng Java.
* 2006: Java trở thành phần mềm mã nguồn mở khi Sun Microsystems phát hành Java dưới giấy phép GPL.
* 2010: Oracle mua lại Sun Microsystems, tiếp quản Java và phát triển tiếp các phiên bản sau này.

### Đặc điểm và các thành phần chính của Java

Đặc điểm nổi bật của Java:

* Đa nền tảng: Java chạy trên nhiều hệ điều hành khác nhau (Windows, Linux, macOS) nhờ JVM.
* Hướng đối tượng: Java hỗ trợ lập trình hướng đối tượng (Object-Oriented Programming - OOP), tập trung vào các đối tượng và thao tác với chúng.
* Bảo mật: Java cung cấp nhiều tính năng bảo mật như sandboxing, ký mã và xác thực để bảo vệ ứng dụng.
* Quản lý bộ nhớ tự động: Java có bộ thu gom rác (Garbage Collection) để quản lý bộ nhớ tự động, giúp giảm thiểu nguy cơ rò rỉ bộ nhớ.
* Đa luồng (Multithreading): Java hỗ trợ việc xử lý đa luồng, cho phép nhiều đoạn mã chạy song song, giúp tối ưu hiệu suất ứng dụng.
* Thư viện phong phú: Java cung cấp một hệ thống thư viện lớn (Java Standard Library) bao gồm các API đa dạng để hỗ trợ phát triển ứng dụng.

Các thành phần chính của Java:

* JVM (Java Virtual Machine): JVM là môi trường thực thi chương trình Java. JVM biên dịch bytecode (định dạng trung gian của mã Java) thành mã máy của nền tảng mà nó đang chạy.
* JDK (Java Development Kit): JDK cung cấp các công cụ và thư viện cần thiết để phát triển, biên dịch và gỡ lỗi các ứng dụng Java. JDK bao gồm JVM và các công cụ phát triển như javac (trình biên dịch), java (trình thực thi) và javadoc (trình tạo tài liệu).

JRE (Java Runtime Environment): JRE là môi trường chạy ứng dụng Java, bao gồm JVM và các thư viện cần thiết để thực thi mã Java.

### Kiến thức cần thiết để làm việc với Java

Kiến thức cơ bản:

* Ngôn ngữ lập trình hướng đối tượng: Hiểu rõ các khái niệm OOP như lớp, đối tượng, kế thừa, đa hình, đóng gói.
* Cấu trúc điều khiển: Thành thạo các cấu trúc điều kiện (if-else, switch), vòng lặp (for, while), và xử lý ngoại lệ (exception handling).
* Biến và kiểu dữ liệu: Hiểu các kiểu dữ liệu nguyên thủy và tham chiếu trong Java (int, float, double, String, ArrayList,...).
* Lập trình đa luồng: Nắm được cách tạo và quản lý luồng, đồng bộ hóa (synchronization), và cơ chế giao tiếp giữa các luồng (thread communication).
* Làm việc với bộ nhớ: Hiểu cơ chế quản lý bộ nhớ, phân phối và thu hồi bộ nhớ tự động qua garbage collection.

Kiến thức nâng cao:

* Java I/O: Làm việc với các tệp tin, đọc và ghi dữ liệu từ hệ thống file.
* Collections Framework: Hiểu các cấu trúc dữ liệu khác nhau trong Java như List, Set, Map, Queue, Stack và cách sử dụng chúng.
* JDBC (Java Database Connectivity): Làm việc với cơ sở dữ liệu, bao gồm kết nối, truy vấn, và quản lý dữ liệu.
* Java Concurrency: Làm việc với các tác vụ đồng thời và lập trình bất đồng bộ.
* Spring Framework: Sử dụng framework Spring để phát triển ứng dụng Java web, tích hợp với các hệ thống khác.

Hibernate: Làm việc với ORM (Object-Relational Mapping) để quản lý cơ sở dữ liệu dễ dàng hơn.

### Ưu và nhược điểm của Java

Ưu điểm:

* Độc lập nền tảng: Khả năng chạy trên nhiều hệ điều hành khác nhau thông qua JVM.
* Bảo mật cao: Java có các cơ chế bảo mật mạnh mẽ, đặc biệt khi chạy trong môi trường mạng.
* Cộng đồng lớn và thư viện phong phú: Java có một cộng đồng phát triển rất lớn và hệ sinh thái phong phú với nhiều thư viện và framework hỗ trợ.
* Hỗ trợ lập trình đa luồng: Giúp xây dựng các ứng dụng hiệu quả và có khả năng mở rộng cao.

Nhược điểm:

* Hiệu suất thấp hơn so với các ngôn ngữ biên dịch trực tiếp: Do Java chạy trên JVM nên hiệu suất thường thấp hơn so với các ngôn ngữ như C++.
* Sử dụng bộ nhớ nhiều: Quá trình quản lý bộ nhớ tự động đôi khi có thể gây ra việc sử dụng bộ nhớ không hiệu quả.
* Tính năng cấu hình phức tạp: Một số framework trong Java yêu cầu cấu hình phức tạp (ví dụ: Spring) khiến việc học và sử dụng có thể khó khăn.

### Python và lịch sử ra đời

Python là một ngôn ngữ lập trình thông dịch (interpreted), mạnh mẽ và linh hoạt, được thiết kế với triết lý dễ đọc và dễ sử dụng. Nó là một ngôn ngữ đa năng, phù hợp với nhiều mục đích từ phát triển web, phân tích dữ liệu, khoa học máy tính đến trí tuệ nhân tạo và học máy. Python nổi tiếng với cú pháp đơn giản, dễ học và khả năng mở rộng cao thông qua các thư viện và framework mạnh mẽ.

Python được tạo ra bởi Guido van Rossum vào cuối những năm 1980 và ra mắt lần đầu vào năm 1991. Guido phát triển Python nhằm thay thế cho ngôn ngữ ABC – một ngôn ngữ dễ học nhưng thiếu tính linh hoạt. Python được đặt tên theo chương trình hài "Monty Python's Flying Circus", với ý nghĩa thể hiện sự sáng tạo và tính giải trí.

### Đặc điểm và các thành phần chính của Python

Đặc điểm nổi bật của Python:

* Cú pháp dễ đọc và dễ học: Python có cú pháp gần gũi với ngôn ngữ tự nhiên, giúp lập trình viên dễ tiếp cận và học nhanh hơn so với các ngôn ngữ khác.
* Thông dịch: Python là ngôn ngữ thông dịch, có nghĩa là mã nguồn được thực thi từng dòng, giúp phát triển và kiểm tra nhanh chóng.
* Đa nền tảng: Python có thể chạy trên hầu hết các hệ điều hành phổ biến hiện nay như Windows, Linux, macOS.
* Hỗ trợ nhiều mô hình lập trình: Python hỗ trợ lập trình hướng đối tượng (OOP), lập trình hàm (functional programming), và lập trình mệnh lệnh (imperative programming).
* Thư viện phong phú: Python có một hệ sinh thái lớn với hàng ngàn thư viện và module hỗ trợ từ phát triển web (Django, Flask), đến khoa học dữ liệu (NumPy, Pandas), và trí tuệ nhân tạo (TensorFlow, PyTorch).

Các thành phần chính của Python:

* Interpreter (Trình thông dịch): Python chạy thông qua trình thông dịch, giúp bạn thực hiện và kiểm tra mã từng dòng một mà không cần biên dịch trước.
* Standard Library (Thư viện chuẩn): Python cung cấp một bộ thư viện chuẩn đa dạng hỗ trợ các tác vụ từ xử lý tệp tin, quản lý hệ thống, đến các thuật toán toán học phức tạp.
* Package Manager (Trình quản lý gói): pip là công cụ giúp cài đặt và quản lý các gói và thư viện bên ngoài một cách dễ dàng.

### Kiến thức cần thiết để làm việc với Python

Kiến thức cơ bản:

* Biến và kiểu dữ liệu: Hiểu các kiểu dữ liệu phổ biến trong Python như số nguyên (int), số thực (float), chuỗi (string), danh sách (list), và từ điển (dictionary).
* Cấu trúc điều khiển: Hiểu các lệnh điều kiện (if-else), vòng lặp (for, while), và cách xử lý ngoại lệ (exception handling) trong Python.
* Hàm và module: Sử dụng các hàm để tổ chức mã hiệu quả hơn, và biết cách tạo và sử dụng các module để tái sử dụng mã.
* Lập trình hướng đối tượng (OOP): Hiểu các khái niệm cơ bản về lớp, đối tượng, kế thừa và đa hình trong Python.

Kiến thức nâng cao:

* Xử lý dữ liệu lớn: Làm việc với các thư viện như NumPy, Pandas để xử lý dữ liệu khối lượng lớn và phức tạp.
* Lập trình mạng: Hiểu cách làm việc với các giao thức mạng, API và các công cụ như requests để giao tiếp với máy chủ web.
* Khoa học dữ liệu và học máy: Sử dụng các công cụ và thư viện mạnh mẽ như TensorFlow, Scikit-learn, PyTorch để phát triển các mô hình học máy và phân tích dữ liệu.
* Lập trình bất đồng bộ: Hiểu cách làm việc với các công cụ bất đồng bộ như asyncio để xây dựng các ứng dụng xử lý đồng thời hiệu quả.

## Môi trường lập trình Android Studio

### Android Studio



Hình 2‑2: Icon của môi trường Android Studio

Android studio: là môi trường phát triển tích hợp chính thức cho tất cả các ứng dụng Android, Android Studio dường như luôn đứng đầu danh sách các công cụ được các nhà lập trình Android ưa thích. Google đã tạo ra Android Studio vào năm 2013, nó đã thay thế công cụ phát triển Android Eclipse (ADT) làm IDE chính để phát triển ứng dụng Android gốc.Android Studio cung cấp các công cụ chỉnh sửa, gỡ lỗi và kiểm tra mã trong một giao diện kéo và thả dễ sử dụng. Android Studio được tải xuống miễn phí và được hỗ trợ không chỉ bởi Google mà còn bởi một cộng đồng lớn và tích cực tham gia của các nhà lập trình Android. Android Studio là một môi trường phát triển tích hợp (IDE) chính thức được phát triển bởi Google dành cho việc phát triển ứng dụng di động Android. Dựa trên IntelliJ IDEA của JetBrains, Android Studio cung cấp một loạt các công cụ và tính năng hữu ích giúp các nhà phát triển xây dựng, kiểm tra và triển khai ứng dụng Android một cách hiệu quả.

### Kiến trúc của nền tảng Android Studio

Các ứng dụng Android được viết bằng ngôn ngữ lập trình Java, và chúng chạy trong một máy ảo (Virtual Machine - VM). Tuy nhiên, máy ảo nào không phải là một máy ảo Java (Java Virtual Machine), mà là máy ảo Dalvik (Dalvik Virtual Machine), một công nghệ mã nguồn mở. Mỗi ứng dụng Android chạy trong một cá thể của máy ảo Dalvik, đến lượt mình, cá thể máy ảo này nằm trong một tiến trình do nhân Linux quản lý, như trình bày dưới đây.

A screenshot of a computer

Description automatically generated

Hình 2‑3: Kiến trúc một ứng dụng Android

Mỗi ứng dụng Android gồm có một hoặc nhiều thành phần sau đây:

* **Activity (Hoạt động):** Mỗi ứng dụng đều có một giao diện người dùng nhìn thấy được, và giao diện đó sẽ được hiển thị bởi các Activity. Một Activity tương ứng với một màn hình hiển thị đơn lẻ của ứng dụng (có thể liên tưởng tới một của sổ trong một ứng dụng trên nền Windows), mà thông qua đó ứng dụng và người dùng tương tác trực tiếp với nhau.
* **Services (Dịch vụ)**: Dịch vụ là các thành phần chạy dưới nền, thực thi các tác vụ cần thời gian dài. Một dịch vụ nên được sử dụng đối với bất kỳ ứng dụng nào cần tồn tại trong một thời gian dài, chẳng hạn như một trình giám sát mạng hoặc ứng dụng kiểm tra-cập nhật.
* **Content providers (Trình cung cấp nội dung)**: Công việc của một trình cung cấp nội dung là quản lý truy cập vào dữ liệu đang tồn tại, chẳng hạn như một cơ sở dữ liệu SQLite, tập tin trên thẻ nhớ, danh bạ của điện thoại… Thông qua trình cung cấp nội dung, ứng dụng có thể truy cập và chỉnh sửa dữ liệu (nếu được cập quyền).
* **Broadcast receivers (Trình thu sóng phát):** Một thành phần của ứng dụng Android có thể được khởi chạy để xử lý một phần tử dữ liệu hoặc đáp ứng một sự kiện, chẳng hạn như nhận một văn bản thông báo.

Một dự án Android có chứa một tập tin tên là AndroidManifest.xml. AndroidManifest.xml chứa các thông tin cấu hình cần thiết để cài đặt ứng dụng vào thiết bị. Nó gồm các tên lớp cần thiết và các kiểu sự kiện mà ứng dụng có thể xử lý, và các quyền hạn ứng dụng cần có để chạy. Mỗi ứng dụng có thể yêu cầu được cấp các quyền hạn riêng. Điều này giúp giảm bớt khả năng xảy ra là một ứng dụng xấu có thể truy cập đến những thành phần không được phép.

### Bộ công cụ phát triển phần mềm Android Android SDK

Cách dễ nhất để bắt đầu phát triển các ứng dụng Android là tải về Bộ công cụ phát triển phần mềm Android – Android SDK và Môi trường phát triển tích hợp Eclipse - Eclipse IDE. Việc phát triển Android có thể thực hiện trên hệ điều hành Windows, Mac OS hoặc Linux.

● Bộ công cụ Android SDK có đầy đủ các thành phần cần thiết cho việc phát triển một ứng dụng cho thiết bị chạy Android:

● Bộ thư viện lập trình của Android

● Trình quản lý máy ảo AVD, giúp tạo và quản lý các máy ảo Android (Emulator) phục vụ cho công việc testing và debugging.

● Bộ công cụ nhúng (plugin) cho Eclipse hỗ trợ việc lập trình như cửa sổ ghi log LogCat, trình duyệt file trên thiết bị File Explorer…

## Firebase



Hình 2‑4: Hình ảnh Firebase

Firebase là một trong những BaaS (Backend as a Service), tức là một dịch vụ cung cấp các giải pháp backend cho các ứng dụng web và di động. Bạn không cần phải tự thiết kế, triển khai và quản lý server hay cơ sở dữ liệu của mình, mà chỉ cần sử dụng các API và SDK do Firebase cung cấp để kết nối với các dịch vụ của nó. Bằng cách này, bạn có thể tiết kiệm thời gian, chi phí và công sức cho việc xây dựng backend. Firebase được ra đời vào năm 2011 bởi James Tamplin và Andrew Lee với tên gọi ban đầu là Evolve, một nền tảng cung cấp các API để tích hợp tính năng chat vào các trang web. Sau đó, họ nhận ra rằng nền tảng này được sử dụng để truyền dữ liệu ứng dụng chứ không chỉ là chat. Họ đã phát triển Evolve thành Firebase và công bố nó vào tháng 4 năm 2012. Đến tháng 10 năm 2014, Firebase đã được Google mua lại và trở thành một phần của Google Cloud Platform.

Hiện nay, Firebase có hơn 20 dịch vụ khác nhau để hỗ trợ các nhà phát triển ứng dụng web và di động. Các dịch vụ này có thể được chia thành ba nhóm chính: Build, Release & Monitor và Engage. Mỗi nhóm sản phẩm bao gồm nhiều công cụ và dịch vụ khác nhau để giải quyết các thách thức và nhu cầu phổ biến trong quá trình phát triển ứng dụng. Nhóm sản phẩm Build giúp bạn tăng tốc độ phát triển ứng dụng với cơ sở hạ tầng backend được quản lý hoàn toàn. Bạn không cần lo lắng về việc quản lý máy chủ hay cấu hình cơ sở dữ liệu. Firebase cung cấp cho bạn các giải pháp như: **Cloud Firestore:** Một cơ sở dữ liệu NoSQL đám mây có thể mở rộng để lưu trữ và truy vấn dữ liệu cho ứng dụng của bạn. **Authentication:** Một dịch vụ xác thực người dùng hỗ trợ nhiều phương thức đăng nhập khác nhau, như email, mật khẩu, số điện thoại  , Google, Facebook, Twitter... **Extensions:** Những gói mã nguồn mở được đóng gói sẵn để tự động hóa các tác vụ phát triển thông thường, như đồng bộ hóa dữ liệu giữa Cloud Firestore và BigQuery, tích hợp với Algolia để tìm kiếm, gửi email khi có sự kiện xảy ra, xử lý thanh toán với Stripe... **Crashlytics:** Một dịch vụ báo cáo sự cố giúp bạn xác định, ưu tiên và khắc phục các lỗi gây ra sự cố trong ứng dụng của bạn. **Performance Monitoring:** Một dịch vụ giám sát hiệu suất giúp bạn đo lường và cải thiện hiệu suất của ứng dụng như thời gian tải ứng dụng, thời gian phản hồi của mạng, tiêu thụ pin và nhiều hơn nữa. **Cloud Messaging:** Một dịch vụ cho phép bạn gửi tin nhắn và thông báo đẩy cho người dùng của bạn, như thông báo về các sự kiện, khuyến mãi, cập nhật tin tức, …

Cloud Firestore: là cơ sở dữ liệu phi quan hệ, là một loại cơ sở dữ liệu không tuân theo mô hình quan hệ của cơ sở dữ liệu truyền thống. Trong cơ sở dữ liệu quan hệ, dữ liệu được tổ chức thành các bảng có mối quan hệ với nhau thông qua các khóa ngoại. Tuy nhiên, trong cơ sở dữ liệu phi quan hệ, dữ liệu thường được tổ chức theo các cấu trúc linh hoạt hơn như tài liệu, cặp khóa-giá trị, cột hoặc đồ thị.

## FastAPI



Hình 2‑5: Hình ảnh FastAPI

FastAPI là framework hiện đại, hiệu suất cao để xây dựng web API với Python 3.7+ dựa trên các tiêu chuẩn của Python. Nó giúp các nhà phát triển code ít hơn, triển khai nhanh hơn và hỗ trợ docs API đầy đủ, cũng cải thiện tốc độ khi tích hợp API vì đã support docs API đầy đủ. Sẽ là một sự lựa chọn hoàn hảo cho một dự án cần triển khai trong thời gian ngắn nhưng vẫn đảm bảo đầy đủ các yếu tố đi kèm (người ta hay nói nhỏ nhưng có võ). **Các tính năng chính:**

* **Nhanh (Fast)**: Từ chính document của framework có thể tự tin so sánh tốc độ ngang bằng với NodeJs và Go.
* **Triển khai nhanh (Fast to code)**: Tăng tốc độ triển khai các tính năng từ 200% đến 300%.
* **Ít lỗi hơn (Fewer bugs)**: Giảm khoảng 40% lỗi do lập trình viên gây ra.
* **Trực quan và tiện dụng (Intuitive)**: Được nhiều trình chỉnh sửa hỗ trợ, có thể chạy trên nhiều nền tảng khác nhau. Tốn ít thời gian gỡ lỗi hơn (hưởng lợi từ Python là một ngôn ngữ thông dịch)
* **Dễ dàng tiếp cận (Easy)**: Thiết kế dễ sử dụng và hỏi tập nhanh thông qua docs.
* **Code ngắn gọn (Short)**: Code ngắn gọn, bao gồm nhiều tính năng đã được triển khai sẵn và giúp giảm thiểu bugs.
* **Mạnh mẽ (Robust)**: Code sẵn sàng cho môi trường production và API docs được sinh tự động.

# PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Mô tả tổng quan hệ thống

### Tổng quan hệ thống

Hệ thống ứng dụng học tiếng Anh có thể được thiết kế như một nền tảng trực tuyến, bao gồm các thành phần chính sau đây:

#### Kiến trúc tổng quan

* Lớp giao diện người dùng:
* Phụ trách hiển thị và tương tác với người dùng.
* Được xây dựng bằng Java để hỗ trợ nền tảng Android.
* Lớp logic xử lý nghiệp vụ:
* Xử lý dữ liệu, áp dụng logic nghiệp vụ và giao tiếp với cơ sở dữ liệu.
* Sử dụng công nghệ Python FastApi.
* Xử lý các tác vụ cập nhật, xử lý dữ liệu của hệ thống.
* Lớp cơ sở dữ liệu:
* Lưu trữ toàn bộ dữ liệu của hệ thống.
* Sử dụng cơ sở dữ liệu phi quan hệ Firebase để lưu trữ.

### Các yêu cầu của hệ thống

#### Yêu cầu chức năng

Bài học: Học vocab và grammar theo thang Cambridge A1-C2 ( Course: khóa học). Trong phần Course sẽ có những bài học (Lessons) và bài học phân theo chủ đề. Mỗi bài học sẽ bao gồm phần học lý thuyết và phần bài tập. Phần bài tập sẽ có dạng bài phù hợp cho các kỹ năng nghe nói đọc viết.

* Về phần bài học:
* Học từ vựng theo dạng flashcard
* Học từ vựng theo chủ đề: family, animals,...
* Học ngữ pháp thông qua lý thuyết theo thì (tenses), mệnh đề quan hệ (relative clauses), giới từ (prepositions)... và có các ví dụ cụ thể để người dùng thông hiểu.
* Vế phần bài tập ôn luyện:
* Nghe 1 từ tiếng anh được phát âm qua audio rồi chọn từ đúng trong 4 phương án
* Câu hỏi bài tập từ vựng bao gồm các dạng: sắp xếp ký tự thành từ có nghĩa, cho từ tiếng Việt hoặc tiếng Anh rồi chọn nghĩa của từ đó
* Câu hỏi bài tập ngữ pháp bao gồm các dạng: dạng multiple choice ( chọn phương án đúng), dạng điền đáp án đúng vào chỗ trống (fill in blank - chia động từ), hoàn thiện câu hoàn chỉnh bằng những từ cho trước (rewrite sentences), viết câu tường thuật ( reported speech), bài đọc hiểu văn bản và trả lời câu hỏi ( chọn đáp án đúng)
* Bài kiểm tra: Mỗi trình độ sẽ có những bài kiểm tra cuối khoá để kiểm tra kiến thức người học. Bài kiểm tra sẽ bao gồm kiểm tra về từ vựng và ngữ pháp. Dạng bài kiểm tra sẽ giống những dạng bài phần bài tập ôn luyện.

#### Yêu cầu phi chức năng

Đối với yêu cầu xử lý phi chức năng, cần đảm bảo các điều sau:

* Bảo mật:
* Hiệu năng:
* Đáp ứng nhu cầu người sử dụng:

Đối với giao diện người sử dụng (UI-UX):

* Thiết kế giao diện cần đẹp và thân thiện với người dùng, đảm bảo tương tác dễ dàng giữa hệ thống và người dùng.
* Các màn hình nhập và cập nhật dữ liệu cần đồng nhất về các thao tác trên bàn phím, màu sắc và font chữ.
* Các màn hình tra cứu và lọc dữ liệu cần được thống nhất về cách thức hiển thị.
* Các chức năng nên được sắp xếp một cách hợp lý và logic để người dùng có thể sử dụng dễ dàng.

### Các chức năng của hệ thống

|  |  |  |
| --- | --- | --- |
| Mục | Tên chức năng | Mô tả |
| 1 | Đăng kí tài khoản | Người dùng có thể thực hiện thao tác đăng ký để trở thành viên để có thể sử dụng các chức năng của hệ thống |
| 2 | Đăng nhập | Người dùng đăng nhập vào hệ thông, bắt buộc phải nhập user và password. Kiểm tra hợp lệ trùng với dữ liệu có sẵn thì hệ thống sẽ cho phép đăng nhập. |
| 3 | Tham gia khóa học | Cho phép người dùng có thể tham gia bắt đầu học khóa học |
| 4 | Học bài học | Cho phép người dùng học bài học ngữ pháp hoặc từ vựng |
| 5 | Làm đề thi trắc nghiệm | Người dùng có thể tham gia làm đề thi và xem kết quả sau khi làm bài |
| 6 | Dịch/tra cứu từ vựng | Cho phép người dùng tìm kiếm hay dịch từ vựng từ tiếng Anh qua tiếng Việt và ngược lại |
| 7 | Tạo danh sách từ vựng, thêm từ vựng vào danh sách | Cho phép người dùng có thể tạo ra các danh sách từ vựng, thêm từ vựng vào danh sách để học |
| 8 | Học từ vựng | Cho phép người dùng học từ vựng trong danh sách từ vựng bằng flashcard, có thể kiểm tra sau khi đã học xong danh sách |
| 9 | Cập nhật tài khoản | Người dùng có thể xem thông tin và thay đổi một số thông tin cá nhân |
| 10 | Xem phân tích người dùng | Người dùng có thể xem biểu đồ thống kề về từ vựng học và bài test đã làm |

## Phân tích các chức năng của hệ thống

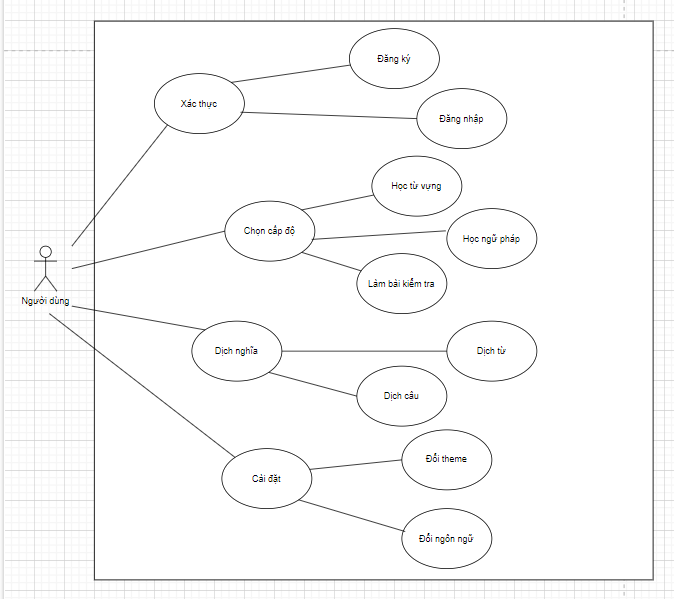
### Sơ đồ phân rã chức năng

**A diagram of a company

Description automatically generated**

### Các biểu đồ use-case

#### Biểu dồ tuần tự use-case tổng quát



#### Biểu đồ tuần tự use-case đăng kí

A diagram of a diagram

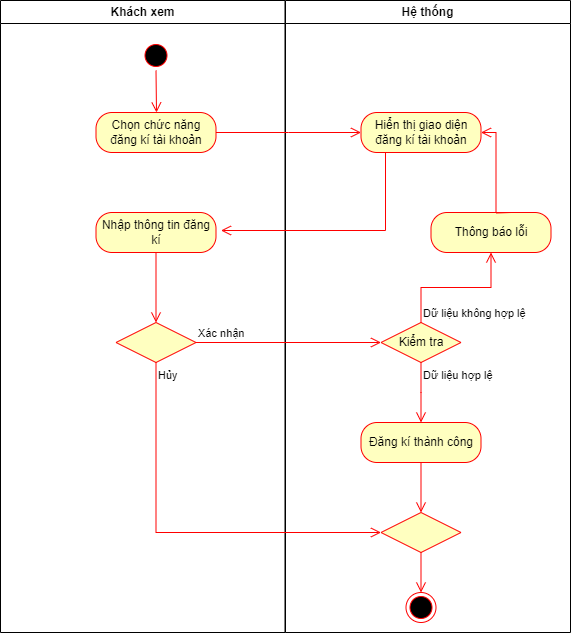
Description automatically generated

Hình 3‑1: Biểu đồ use-case đăng kí

* **Đặc tả use-case đăng kí**

|  |  |
| --- | --- |
| **Tác nhân** | Người dùng |
| **Mô tả** | Chức năng này cho phép người dùng đăng kí làm người dùng trong hệ thống |
| **Điều kiện trước** | Không có |
| **Dòng sự kiện chính** | 1. Người dùng chọn chức năng đăng ký  2. Hệ thống hiển thị form đăng ký.  3. Người dùng nhập thông tin cần thiết vào form.  4. Người dùng xác nhận gửi thông tin.  5. Hệ thống kiểm tra tính hợp lệ của thông tin.  6. Hệ thống xác nhận đăng ký thành công.  7. Người dùng được chuyển tới giao diện bên trong của hệ thống. |
| **Dòng sự kiện rẽ nhánh** | 5a. Nếu thông tin không hợp lệ:   * Hệ thống hiển thị thông báo lỗi và yêu cầu người dùng nhập lại thông tin. |
| **Kết quả** | Người dùng đăng ký thành công vào hệ thống và có thể thực hiện các chức năng của người dùng trong hệ thống. |

* **Sơ đồ luồng hoạt động chức năng đăng kí**



Hình 3‑2: Sơ đồ luồng chức năng đăng kí

#### Biểu đồ tuần tự use-case đăng nhập

A black screen with white text

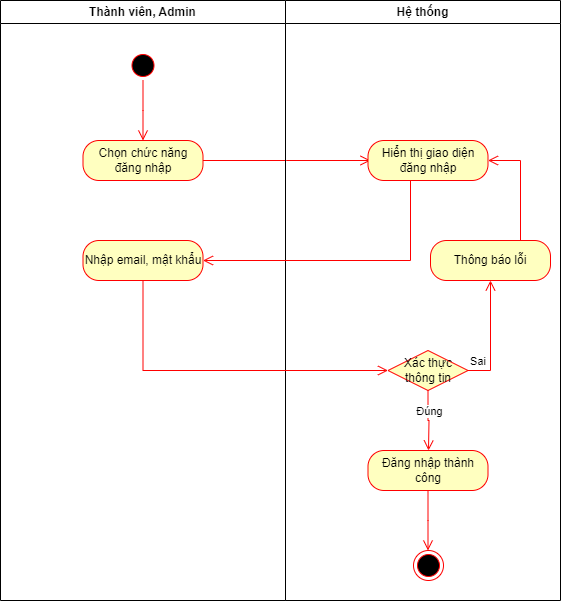
Description automatically generated

Hình 3‑3: Biểu đồ use-case đăng nhập

* **Đặc tả Use Case đăng nhập, đăng xuất**

|  |  |
| --- | --- |
| **Tác nhân** | Người dùng |
| **Mô tả** | Chức năng cho phép người dùng đăng nhập bằng email và mật khẩu qua Firebase Authentication. Sau khi xác thực thành công, hệ thống truy xuất thêm thông tin từ Firestore |
| **Điều kiện trước** | * Người dùng đã có tài khoản hợp lệ trong Firebase Authentication. * Firestore đã lưu thông tin bổ sung (document user có trường role, name, v.v.). * Thiết bị có kết nối internet. |
| **Dòng sự kiện chính** | 1. Người dùng mở ứng dụng và chọn nút “Đăng nhập”. 2. Hệ thống hiển thị form đăng nhập với các trường: email và mật khẩu. 3. Người dùng nhập thông tin đăng nhập và nhấn “Đăng nhập”. 4. Hệ thống kiểm tra hợp lệ dữ liệu đầu vào (client-side). 5. Hệ thống gửi thông tin đăng nhập đến **Firebase Authentication**. 6. Firebase xác thực tài khoản: nếu thành công, trả về UID. 7. Hệ thống sử dụng UID để truy xuất dữ liệu từ Firestore 8. Firestore trả về document chứa thông tin chi tiết của người dùng 9. Hệ thống kiểm tra vai trò và điều hướng người dùng đến màn hình tương ứng. |
| **Dòng sự kiện rẽ nhánh** | 1. Sai email hoặc mật khẩu: Firebase trả lỗi → hệ thống hiển thị thông báo: “Email hoặc mật khẩu không đúng.” 2. Tài khoản chưa tồn tại: Firebase trả lỗi → thông báo “Tài khoản không tồn tại.” 3. Kết nối mạng lỗi: Không thể gửi yêu cầu → hệ thống hiển thị thông báo lỗi kết nối. 4. Không tìm thấy user trong Firestore: Firebase trả UID nhưng không có document tương ứng → hiển thị lỗi nội bộ, có thể đăng xuất. 5. Vai trò không hợp lệ: Nếu role không xác định → thông báo “Không xác định quyền truy cập.” |
| **Kết quả** | Thành viên, Admin đăng nhập hoặc đăng xuất thành công. |

* **Sơ đồ luồng hoạt động chức năng đăng nhập**



Hình 3‑4: Sơ đồ luồng hoạt động chức năng đăng nhập

#### Biểu đồ tuần tự use-case dịch/tra cứu từ vựng

A diagram of a diagram

Description automatically generated

Hình 3‑5: Biểu đồ use-case dịch/tra cứu từ điển

* **Đặc tả Use Case tra cứu từ điển**

|  |  |
| --- | --- |
| **Tác nhân** | Người dùng |
| **Mô tả** | Chức năng này cho phép người dùng dịch/tra cứu từ vựng |
| **Điều kiện trước** | Không có |
| **Dòng sự kiện chính** | 1. Người dùng nhập từ vựng cần tra cứu vào ô nhập từ vựng  2. Hệ thống hiển thị từ vựng phù hợp |
| **Dòng sự kiện rẽ nhánh** | Không có |
| **Kết quả** | Người dùng có thể dịch/tra cứu từ vựng |

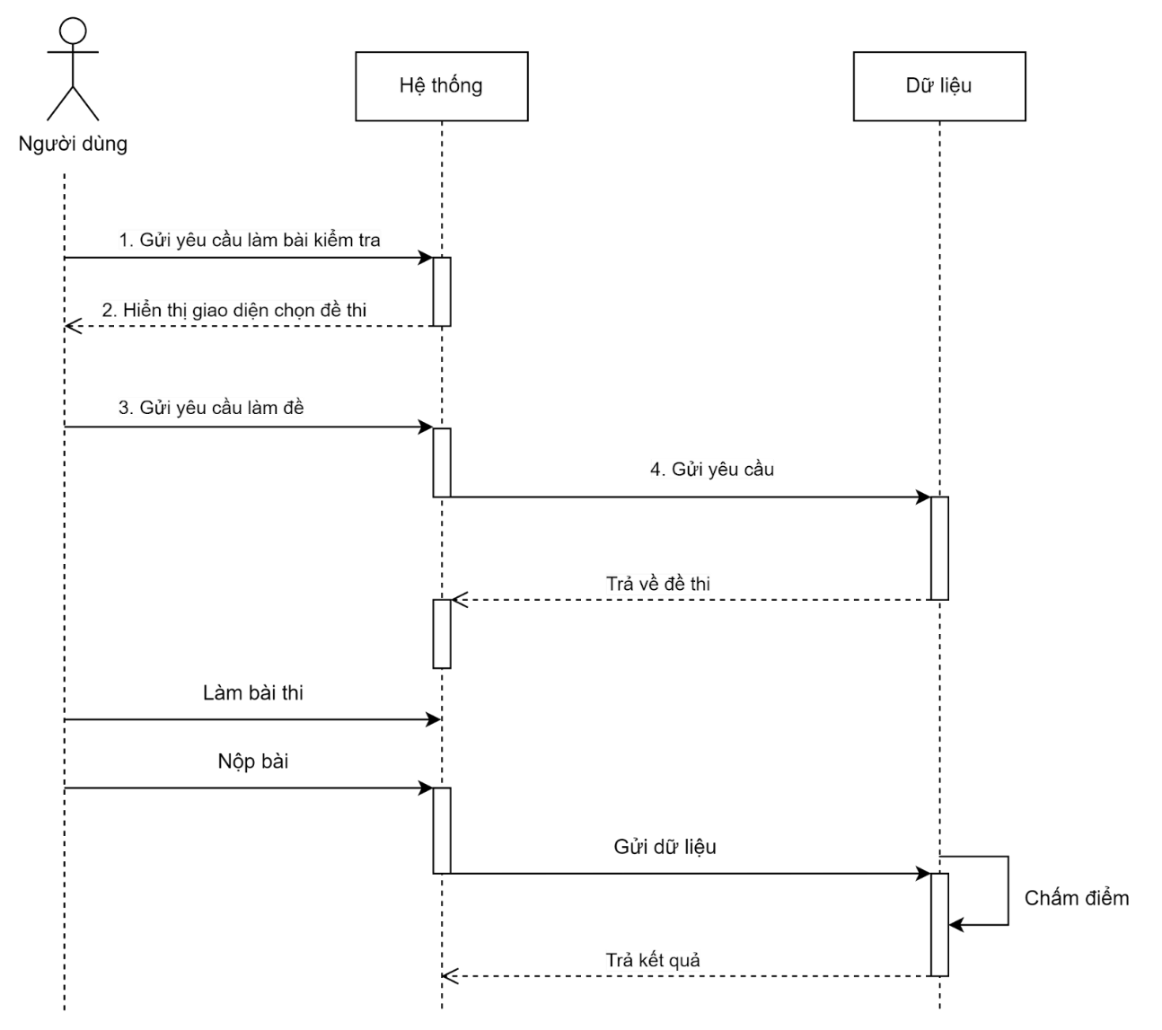
* **Sơ đồ luồng hoạt động chức năng tra cứu từ điển**

**A screenshot of a computer

Description automatically generated**

Hình 3‑6: Sơ đồ luồng hoạt động chức năng tra cứu từ điển

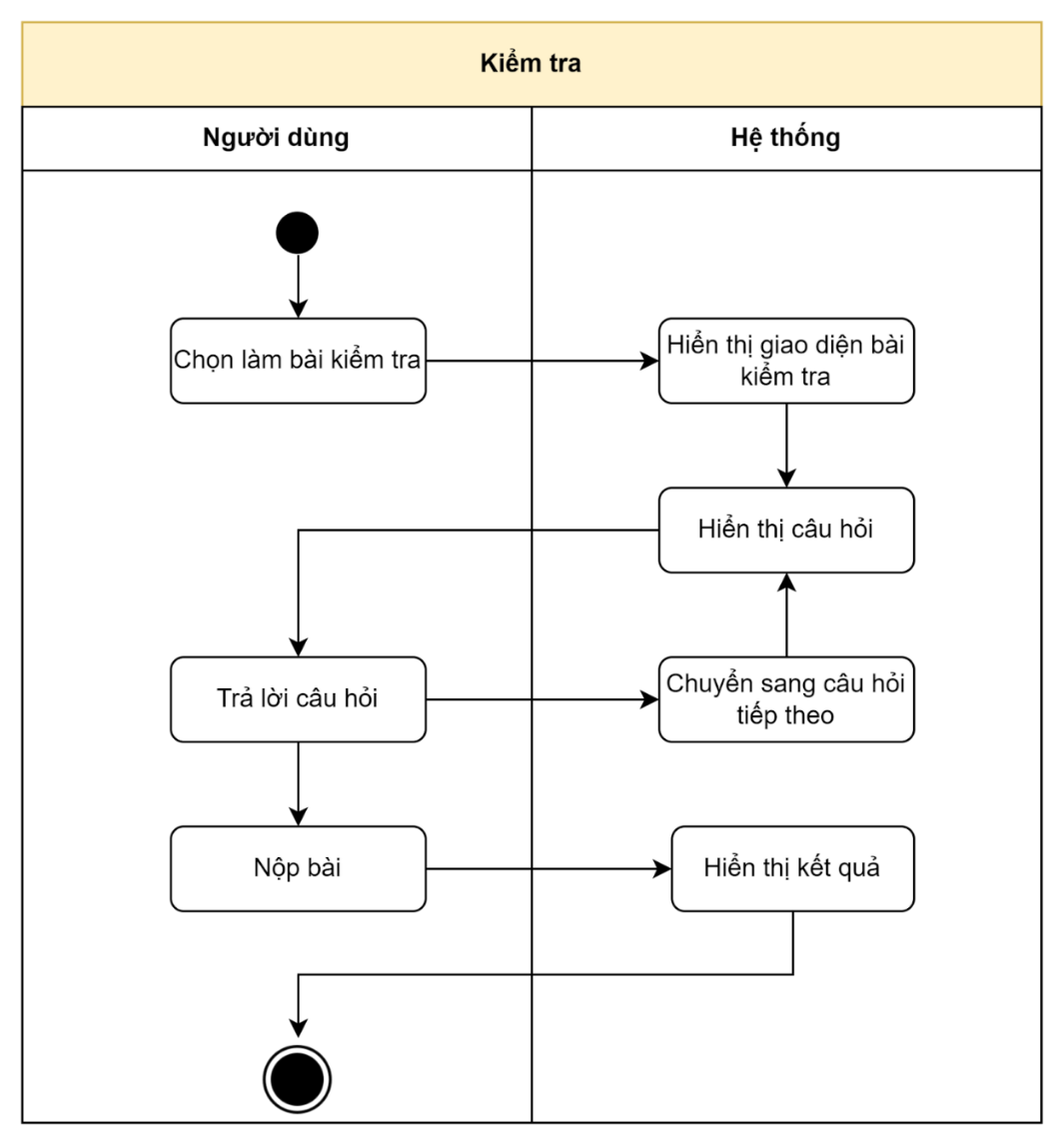
#### Biểu đồ tuần tự use-case làm bài kiểm tra



Hình 3‑7: Biểu đồ use-case làm bài kiểm tra

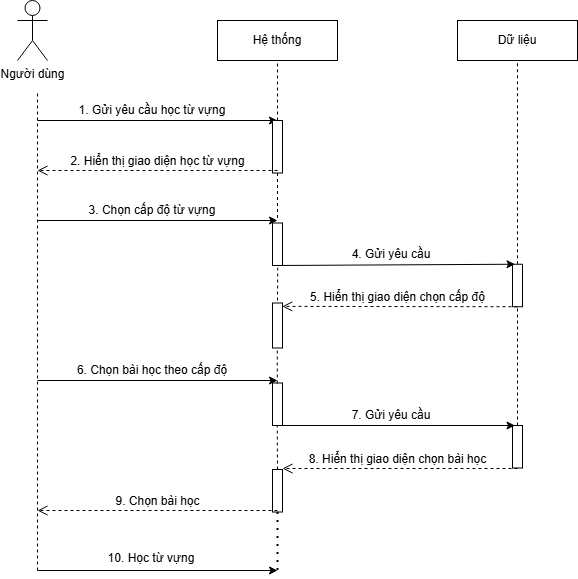
|  |  |
| --- | --- |
| **Tác nhân** | Người dùng |
| **Mô tả** | Chức năng cho phép người dùng xem và làm bài kiểm tra |
| **Điều kiện trước** | Phải có tài khoản |
| **Dòng sự kiện chính** | 1. Người dùng chọn biểu tượng Exam.  2. Hệ thống hiển thị màn hình chọn cấp độ.  3. Người dùng chọn cấp độ phù hợp.  4. Hệ thống đưa ra các đề phù hợp tương ứng có trong dữ liệu.  5. Nếu người dùng đã làm bài kiểm tra sẽ hiển thị điểm lần gần nhất làm, nếu chưa làm bài kiểm tra đó thì chỉ hiển thị làm bài.  6. Nếu người dùng nhấn vào xem lịch sử,hệ thống đưa ra các lần mà người dùng làm bài kiểm tra.  7. Trong giao diện xem lịch sử có thể xem lại bài mình đã làm.  8. Nếu bài kiểm tra chưa làm bao giờ khi ấn vào làm bài hệ thống sẽ hiện lên giao diện để người dùng làm bài.  9. Trong giao diện làm bài sẽ được chia làm các phần:   * Phần nghe: có một trình phát âm thanh * Phần đọc: có một đoạn văn ngắn * Phần chọn đáp án đúng: người dùng cần chọn đáp án đúng nhất * Phần viết lại câu: người dùng chọn các từ đã có để tạo thành câu hoàn chỉnh   10. Giao diện làm bài sẽ tải tất cả dữ liệu trước khi người dùng làm và có đồng hồ đếm ngược.  11. Nếu người hoàn thành bài và chọn submit hoặc đồng hồ đếm ngược kết thúc thì sẽ hiện lên một cửa sổ popup hiện kết quả của người dùng đồng thời đưa dữ liệu lên dữ liệu của hệ thống. |

**Sơ đồ luồng hoạt động chức năng làm bài kiểm tra:**



Hình 3‑8: Sơ đồ luồng hoạt động chức năng làm bài kiểm tra

#### Biểu đồ tuần tự use-case học từ vựng

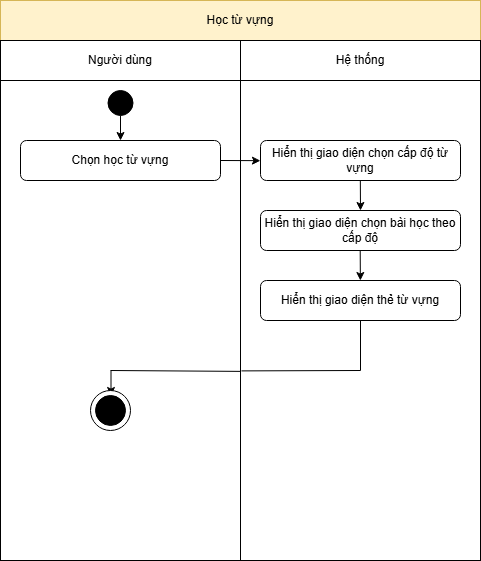


Hình 3-9: Biểu đồ use-case học từ vựng

* Đặc tả use-case học từ vựng

|  |  |
| --- | --- |
| **Tác nhân** | Người dùng |
| **Mô tả** | Chức năng cho phép người dùng học từ vựng thông qua các bài học theo cấp độ |
| **Điều kiện trước** | Phải có tài khoản để đăng nhập vào ứng dụng |
| **Dòng sự kiện chính** | 1. Người dùng chọn Button Vocabulary.  2. Hệ thống hiển thị màn hình chọn cấp độ từ vựng.  3. Người dùng chọn cấp độ phù hợp.  4. Hệ thống đưa ra các bài học từ vựng theo từng cấp độ.  5. Người dùng chọn bài học theo ý muốn.  6. Trong giao học từ vựng sẽ được chia làm các phần:   * Phần nghe: có một trình phát âm thanh * Phần đọc: có một đoạn văn bản hiển thị từ vựng, loại từ, phiên âm. |

* **Sơ đồ luồng hoạt động chức năng học từ vựng**



Hình 3-10: Sơ đồ luồng hoạt động chức năng học từ vựng

## Phân tích thiết kế cơ sở dữ liệu

### Phân tích cơ sở dữ liệu

Theo mô tả và đặc tả các yêu cầu của bài toán được nêu ở các mục trước đây, ta có thể xác định được các thực thể cần có trong một hệ thống học tiếng anh đó là:

* User: bao gồm các thông tin cơ bản của người dùng
* Lesson: bao gồm thông tin của các bài học trên hệ thống
* Test: bao gồm thông tin của các bài kiểm tra
* Question: bao gồm thông tin của các câu hỏi trong bài kiểm tra
* Answer: bao gồm thông tin của các câu trả lời của câu hỏi
* Vocabulary: bao gồm thông tin của các từ vựng
* Topic: bao gồm thông tin của chủ đề
* CERT Level: bao gồm thông tin của cập độ theo thang CEFT từ A1 – C2
* Part: bao gồm thông tin của phần trong bài kiểm tra

### Thiết kế cơ sở dữ liệu

Dựa vào lý thuyết về cơ sở dữ liệu quan hệ ta có danh sách các bảng sau:

#### Bảng USER

Chứa thông tin cơ bản của người dùng

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| USER\_ID | String | Primary key | Mã người dùng |
| USER\_NAME | String | NOT NULL | Tên tài khoản của người dùng |
| PHONE\_NUMBER | String | NOT NULL | Số điện thoại của người dùng |
| EMAIL | String | NOT NULL | Email của người dùng |
| PASSWORD | String | NOT NULL | Mật khẩu của người dùng |

#### Bảng CERT\_LEVEL

Chứa thông tin cấp độ theo thang CERT gồm 6 cấp độ (A1, A2, B1, B2, C1, C2)

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| CERT\_LEVEL\_ID | String | Primary key | Mã cấp độ của mỗi thang CERT |
| CERT\_LEVEL\_NAME | String | NOT NULL | Tên của cấp độ |
| DESCRIPTION | String | NOT NULL | Mô tả về cấp độ |
| IMAGE\_LINK | String | NOT NULL | Chứa link ảnh để hiện thị trong ứng dụng |

#### Bảng TOPIC

Chứa thông tin chủ đề

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| TOPIC\_ID | String | Primary key | Mã chủ đề |
| TOPIC\_NAME | String | NOT NULL | Tên của chủ đề |
| DESCRIPTION | String | NOT NULL | Mô tả về chủ đề |
| IMAGE\_LINK | String | NOT NULL | Chứa link ảnh để hiển thị trong ứng dụng |

#### Bảng LESSON

Chứa thông tin bài học

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| LESSON\_ID | String | Primary key | Mã bài học |
| CERT\_LEVEL\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng CERT\_LEVEL |
| CERT\_LEVEL\_NAME | String | NOT NULL | Tên của cập độ liên kết với bảng CERT\_LEVEL |
| TOPIC\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng TOPIC |
| TOPIC\_NAME | String | NOT NULL | Tên của chủ đề liên kết với bảng TOPIC |
| IMAGE\_LINK | String | NOT NULL | Chứa link ảnh để hiển thị trong ứng dụng |
| NAME\_OF\_LESSON | String | NOT NULL | Tên của bài học |

#### Bảng TEST

Chứa thông tin bài kiểm tra

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| TEST\_ID | String | Primary key | Mã bài kiểm tra |
| NAME\_OF\_TEST | String | NOT NULL | Tên của bài test |
| TEST\_TYPE\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng TEST\_TYPE |

#### Bảng QUESTION

Chứa thông tin của câu hỏi

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| QUESTION\_ID | String | Primary key | Mã câu hỏi |
| CONTENT | String | NOT NULL | Nội dung câu hỏi |
| QUESTION\_TYPE\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng QUESTION\_TYPE |
| TEST\_ID | String | NOT\_NULL | Khóa phụ liên kết đến bảng TEST |
| PART\_DETAIL\_ID | String | NOT\_NULL | Khóa phụ liên kết đến bảng PART\_DETAIL |
| ORDER | String | NOT\_NULL | Thứ tự câu hỏi hiển thị trong bài kiểm tra |
| CORRECT\_ANSWER\_ID | String | NULL | Khóa phụ liên kết tới bảng ANSWER mà nó là đáp án đúng của câu hỏi |
| IMAGE\_LINK | String | NULL | Chứa link ảnh để hiển thị trong ứng dụng |

#### Bảng ANSWER

Chứa thông tin của câu trả lời

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| ANSWER\_ID | String | Primary key | Mã của câu trả lời |
| CONTENT | String | NOT NULL | Nội dung câu trả lời |
| ORDER | String | NOT NULL | Thứ tự hiển thị trong câu hỏi |
| QUESTION\_ID | String | NOT NULL | Khóa phụ liên kết với bảng QUESTION |
| TEST\_ID | String | NOT NULL | Khóa phụ liên kết với bảng TEST |

#### Bảng VOCAB

Chứa thông tin của từ vựng

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| VOCAB\_ID | String | Primary key | Mã từ vựng |
| CERT\_LEVEL\_ID | String | NOT NULL | Khóa phụ liên kết tời bảng CERT\_LEVEL |
| CERT\_LEVEL\_NAME | String | NOT NULL | Tên của cập độ liên kết với bảng CERT\_LEVEL |
| TOPIC\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng TOPIC |
| TOPIC\_NAME | String | NOT NULL | Tên của chủ đề liên kết với bảng TOPIC |
| AUDIO\_LINK | String | NOT NULL | Chứa link audio để phát trong ứng dụng |
| LESSON\_ID | String | NULL | Khóa phụ liên kết tới bảng LESSON |
| MEANING | String | NOT NULL | Nghĩa tiếng Việt của từ vựng |
| PART\_OF\_SPEECH | String | NOT NULL | Chữa loại từ (danh từ, động từ…) |
| PRONUNCIATION | String | NOT NULL | Phát âm của từ |
| WORD | String | NOT NULL | Từ tiếng Anh |

#### Bảng PART\_DETAIL

Chứa thông tin của mỗi phần trong bài kiểm tra

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| PART\_DETAIL\_ID | String | Primary key | Mã của phần trong test |
| PART\_OF\_TEST\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng PART\_OF\_TEST |
| TEST\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng TEST |
| ORDER | String | NOT NULL | Thứ tự phần hiển thị trong bài kiểm tra |
| AUDIO\_LINK | String | NULL | Chứa link audio để phát trong ứng dụng |
| CONTENT | String | NULL | Chứa nội dụng của phần |

#### Bảng QUESTION\_TYPE

Chứa thông tin về loại câu hỏi (trắc nghiệm, tự luân…)

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| QUESTION\_TYPE\_ID | String | Primary key | Mã loại câu hỏi |
| NAME\_OF\_TYPE | String | NOT NULL | Tên của loại câu hỏi |

#### Bảng TEST\_TYPE

Chứa thông tin về loại bài kiểm tra dùng để phân loại bài kiểm tra

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| TEST\_TYPE\_ID | String | Primary key | Mã loại bài kiểm tra |
| NAME\_OF\_TEST\_TYPE | String | NOT NULL | Tên của loại bài kiểm tra |
| CERT\_LEVEL\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng CERT\_LEVEL |
| CERT\_LEVEL\_NAME | String | NOT NULL | Tên của cập độ liên kết với bảng CERT\_LEVEL |
| MAXIMUM\_SCORE | Double | NOT NULL | Điểm tối đa có thể đạt được |
| TOTAL\_DURATION | String | NOT NULL | Tổng thời gian làm bài |

#### Bảng PART\_OF\_TEST

Chứa thông tin của loại Part dùng đề phận loại các phần trong một loại bài kiểm tra

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| PART\_OF\_TEST\_ID | String | Primary key | Mã của loại phần trong loại bài kiểm tra |
| TEST\_TYPE\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng TEST\_TYPE |
| NAME\_OF\_PART | String | NOT NULL | Tên của phần |

#### Bảng USER\_TEST

Chứa thông tin làm bài kiểm tra của người dùng

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| USER\_TEST\_ID | String | Primary key | Mã của bảng |
| USER\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng USER |
| TEST\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng TEST |
| TEST\_DATE | String | NOT NULL | Ngày làm bài kiểm tra |
| SCORE | Double | NOT NULL | Điểm mà người dùng đạt được |
| DURATION | String | NOT NULL | Thời gian làm bài của người dùng |

#### Bảng USER\_ANSWER

Chứa thông tin câu trả lời của người dùng

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| USER\_ANSWER\_ID | String | Primary key | Mã của bảng |
| USER\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng USER |
| QUESTION\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng QUESTION mà người dùng làm |
| ANSWER\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng ANSWER mà người dùng trả lời |
| RAW\_USER\_ANSWER | String | NULL | Câu trả lời mà người dùng nhập |
| TEST\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng TEST mà người dùng làm |
| USER\_TEST\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng USER\_TEST |
| USER\_NOTE | String | NULL | Ghi chú của người dùng |

#### Bảng USER\_VOCAB

Chứa thông tin của người dùng đối với việc học mỗi từ vựng

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| USER\_VOCAB\_ID | String | Primary key | Mã của bảng |
| USER\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng USER |
| VOCAB\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng VOCAB |
| MEMORY\_LEVEL | Integer | NOT NULL | Cấp độ nhớ từ của người dùng (1 – 7) |
| USER\_NOTE | String | NULL | Ghi chú của người dùng |
| WORD | String | NOT NULL | Từ tiếng Anh |
| MEANING | String | NOT NULL | Nghĩa tiếng Việt của từ |

#### Bảng LESSON\_PROCESS

Chứa thông tin mỗi lần học của người dùng với mỗi một lesson

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| LESSON\_PROCESS\_ID | String | Primary key | Mã của bảng quá trình học |
| USER\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng USER |
| LESSON\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng LESSON |
| LEARN\_DATE | String | NOT NULL | Ngày mà người học |
| DURATION | String | NOT NULL | Thời gian hoàn thành mỗi lần học |
| STATUS | Integer | NOT NULL | Trạng thái đối với bài học với mỗi lần học (Đang học – 1, Hoàn thành – 2) |

#### Bảng DATE\_ATTENDING

Chứa thông tin về một ngày học của người dùng

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| DATE\_ATTENDING\_ID | String | Primary key | Mã của ngày học |
| ATTENDING\_DATE | String | NOT NULL | Ngày học |
| DAY\_DURATION | String | NOT NULL | Tổng thời gian học của người dùng trong một ngày |
| STREAK | String | NOT NULL | Số ngày học liên tiếp gần nhất |
| USER\_ID | String | NOT NULL | Khóa phụ liên kết đến bảng USER |

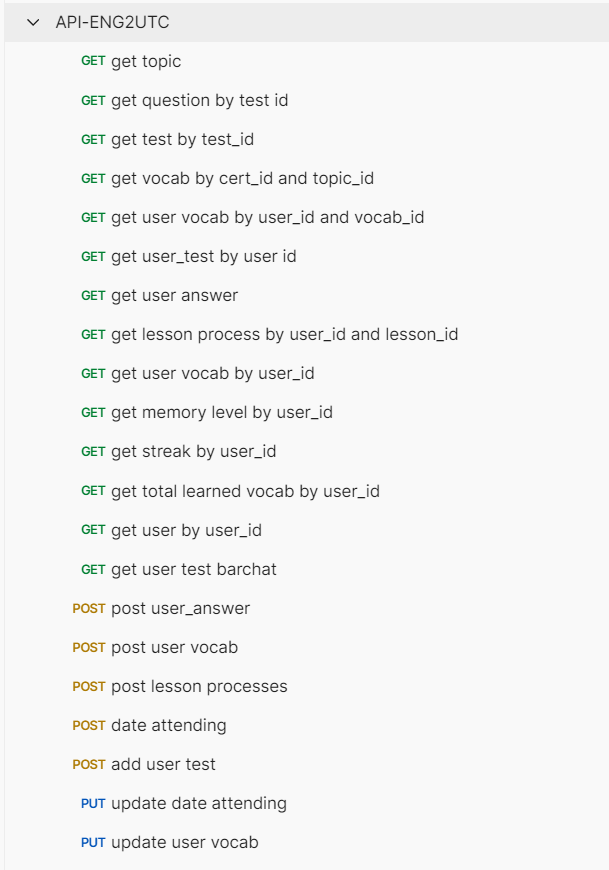
### Sơ đồ quan hệ thực thể

A diagram of a computer

Description automatically generated with medium confidence

Hình 3-11: Sơ đồ thực thể quan hệ

## Danh sách API



Hình 3-12: Danh sách API

### Danh sách API GET

* **/users/user**
  + **Tác dụng**: Lấy thông tin của người dùng dựa trên user\_id.
  + **Tham số**:
    - user\_id (tuỳ chọn): ID của người dùng.
* **/topics**
  + **Tác dụng**: Lấy danh sách tất cả các chủ đề.
* **/questions**
  + **Tác dụng**: Lấy danh sách câu hỏi dựa trên test\_id.
  + **Tham số**:
    - test\_id (tuỳ chọn): ID của bài kiểm tra.
* **/tests**
  + **Tác dụng**: Lấy thông tin bài kiểm tra dựa trên test\_id.
  + **Tham số**:
    - test\_id (tuỳ chọn): ID của bài kiểm tra.
* **/vocabularies/cert-and-topic**
  + **Tác dụng**: Lấy danh sách từ vựng dựa trên cert\_id và topic\_id.
  + **Tham số**:
    - cert\_id (tuỳ chọn): ID chứng chỉ.
    - topic\_id (tuỳ chọn): ID chủ đề.
* **/user\_vocabs/user-vocab**
  + **Tác dụng**: Lấy thông tin từ vựng của người dùng dựa trên user\_id và vocab\_id.
  + **Tham số**:
    - user\_id (tuỳ chọn): ID người dùng.
    - vocab\_id (tuỳ chọn): ID từ vựng.
* **/users/user\_tests/by\_user\_id**
  + **Tác dụng**: Lấy danh sách bài kiểm tra của người dùng dựa trên user\_id.
  + **Tham số**:
    - user\_id (tuỳ chọn): ID người dùng.
* **/users/user\_tests/user\_answers**
  + **Tác dụng**: Lấy danh sách câu trả lời của người dùng trong một bài kiểm tra dựa trên user\_test\_id và test\_id.
  + **Tham số**:
    - user\_test\_id (tuỳ chọn): ID bài kiểm tra của người dùng.
    - test\_id (tuỳ chọn): ID bài kiểm tra.
* **/lesson\_processes/lastest**
  + **Tác dụng**: Lấy quá trình học gần nhất của người dùng trong một bài học dựa trên user\_id và lesson\_id.
  + **Tham số**:
    - user\_id (tuỳ chọn): ID người dùng.
    - lesson\_id (tuỳ chọn): ID bài học.
* **/users/attendings/streak**
  + **Tác dụng**: Lấy chuỗi học tập liên tục của người dùng dựa trên user\_id và attending\_date.
  + **Tham số**:
    - user\_id (tuỳ chọn): ID người dùng.
    - attending\_date (tuỳ chọn): Ngày tham dự.
* **/users/user\_vocabs/word**
  + **Tác dụng**: Lấy danh sách từ vựng mà người dùng đã học dựa trên user\_id.
  + **Tham số**:
    - user\_id (tuỳ chọn): ID người dùng.
* **/users/user\_vocabs/memory\_level**
  + **Tác dụng**: Lấy thống kê số lượng từ vựng theo mức độ ghi nhớ của người dùng dựa trên user\_id.
  + **Tham số**:
    - user\_id (tuỳ chọn): ID người dùng.
* **/users/user\_vocabs/total\_words**
  + **Tác dụng**: Lấy tổng số từ vựng mà người dùng đã học dựa trên user\_id.
  + **Tham số**:
    - user\_id (tuỳ chọn): ID người dùng.
* **/users/tests/mark\_of\_test**
  + **Tác dụng**: Lấy điểm của người dùng sau mỗi bài test để thống kê
  + **Tham số**:
    - user\_id (tuỳ chọn): ID người dùng.

### Danh sách API POST

* **/users/user\_answers**
  + **Tác dụng**: Thêm câu trả lời của người dùng vào hệ thống.
  + **Dữ liệu yêu cầu**:
    - user\_id, question\_id, answer\_id, raw\_user\_answer, user\_note, test\_id, user\_test\_id.
* **/user\_vocabs**
  + **Tác dụng**: Thêm từ vựng vào danh sách từ vựng của người dùng.
  + **Dữ liệu yêu cầu**:
    - user\_id, vocab\_id, word, meaning, memory\_level, status, user\_note.
* **/lesson\_processes**
  + **Tác dụng**: Thêm tiến trình học của người dùng vào hệ thống.
  + **Dữ liệu yêu cầu**:
    - user\_id, lesson\_id, learn\_date, duration, status.
* **/date\_attendings**
  + **Tác dụng**: Thêm ngày tham gia học của người dùng.
  + **Dữ liệu yêu cầu**:
    - attending\_date, user\_id, day\_duration.
* **/users/user\_tests**
  + **Tác dụng**: Thêm thông tin bài kiểm tra của người dùng.
  + **Dữ liệu yêu cầu**:
    - user\_id, test\_id, test\_date, score, duration.

### Danh sách API PUT

* **/date\_attendings**
  + **Tác dụng**: Cập nhật thông tin ngày tham gia học của người dùng.
  + **Tham số**:
    - date\_attending\_id (tuỳ chọn): ID ngày tham gia học.
  + **Dữ liệu yêu cầu**:
    - attending\_date, user\_id, streak, day\_duration.
* **/user\_vocabs**
  + **Tác dụng**: Cập nhật thông tin từ vựng của người dùng.
  + **Tham số**:
    - user\_vocab\_id (tuỳ chọn): ID từ vựng của người dùng.
  + **Dữ liệu yêu cầu**:
    - memory\_level, word, meaning, status, user\_note.

# TRIỂN KHAI VÀ KẾT QUẢ

## Triển khai

### Công cụ phát triển

Android Studio

Pycharm

Firebase

### Hướng dẫn cài đặt

* Yêu cầu phần cứng:
  + CPU: 4 Core - 2 Ghz
  + Ram: 2 GB
* Yêu cầu phần mềm :
  + Android version : Android 7.0 Nougat trở lên.
* Hướng dẫn cài đặt : Sử dụng file apk được cung cấp để cài đặt lên thiết bị.

## Giao diện ứng dụng

|  |  |
| --- | --- |
| Hình 4‑1: Trang Intro | Hình 4‑2: Trang đăng kí |
| Hình 4‑3: Trang đăng nhập | Hình 4‑4: Trang chủ chính |
| Hình 4‑5: Giao diện cài đặt | Hình 4‑6: Đổi màu giao diện |
| Hình 4‑7: Đổi ngôn ngữ hiển thị | Hình 4‑8: Giao diện từ vựng |
| Hình 4‑9: Giao diện từ vựng theo cấp độ | Hình 4‑10: Giao diện bài học theo chủ đề |
| Hình 4‑11: Giao diện danh sách từ vựng theo chủ đề | Hình 4‑12: Giao diện học từ vựng |
| Hình 4‑13: Sắp xếp đúng từ vựng | Hình 4‑14: Sắp xếp sai từ vựng |
| Hình 4‑15: Giao diện ngữ pháp | Hình 4‑16: Giao diện danh sách các thì |
| Hình 4‑17: Giao diện lý thuyết ngữ pháp | Hình 4‑18: Giao diện dịch văn bản |
| Hình 4‑19: Giao diện đề kiểm tra | Hình 4‑20: Xem điểm các lần làm kiểm tra |
| Hình 4‑21: Giao diện bài kiểm tra | Hình 4‑22: Giao diện hiển thị số câu hỏi |
| Hình 4‑23: Giao diện chấm điểm kiểm tra | Hình 4‑24: Giao diện thống kê |
| Hình 4‑25: Giao diện từ được tra cứu | Hình 4‑26: Search Box tra cứu từ điển |
| Hình 4‑27: Giao diện đánh giá ứng dựng |  |

# KIỂM THỬ

## Giới Thiệu Về Việc Kiểm Thử Trong Dự Án

Trong dự án phát triển ứng dụng học từ vựng và ngữ pháp tiếng Anh này, việc kiểm thử đóng vai trò vô cùng quan trọng để đảm bảo chất lượng và hiệu quả của sản phẩm. Chúng tôi áp dụng phương pháp kiểm thử hộp đen (Black Box Testing) để kiểm tra các chức năng của ứng dụng mà không cần biết về cấu trúc bên trong hay mã nguồn.

**Mục tiêu của việc kiểm thử:**

* Đảm bảo tính chính xác của các chức năng: Kiểm tra các tính năng như đăng nhập, đăng ký, học từ vựng, học ngữ pháp và làm bài tập ôn luyện để đảm bảo chúng hoạt động đúng như mong đợi.
* Phát hiện và sửa lỗi: Xác định các lỗi tiềm ẩn trong ứng dụng và sửa chữa chúng trước khi sản phẩm được phát hành.
* Cải thiện trải nghiệm người dùng: Đảm bảo giao diện người dùng thân thiện, dễ sử dụng và các chức năng hoạt động mượt mà.

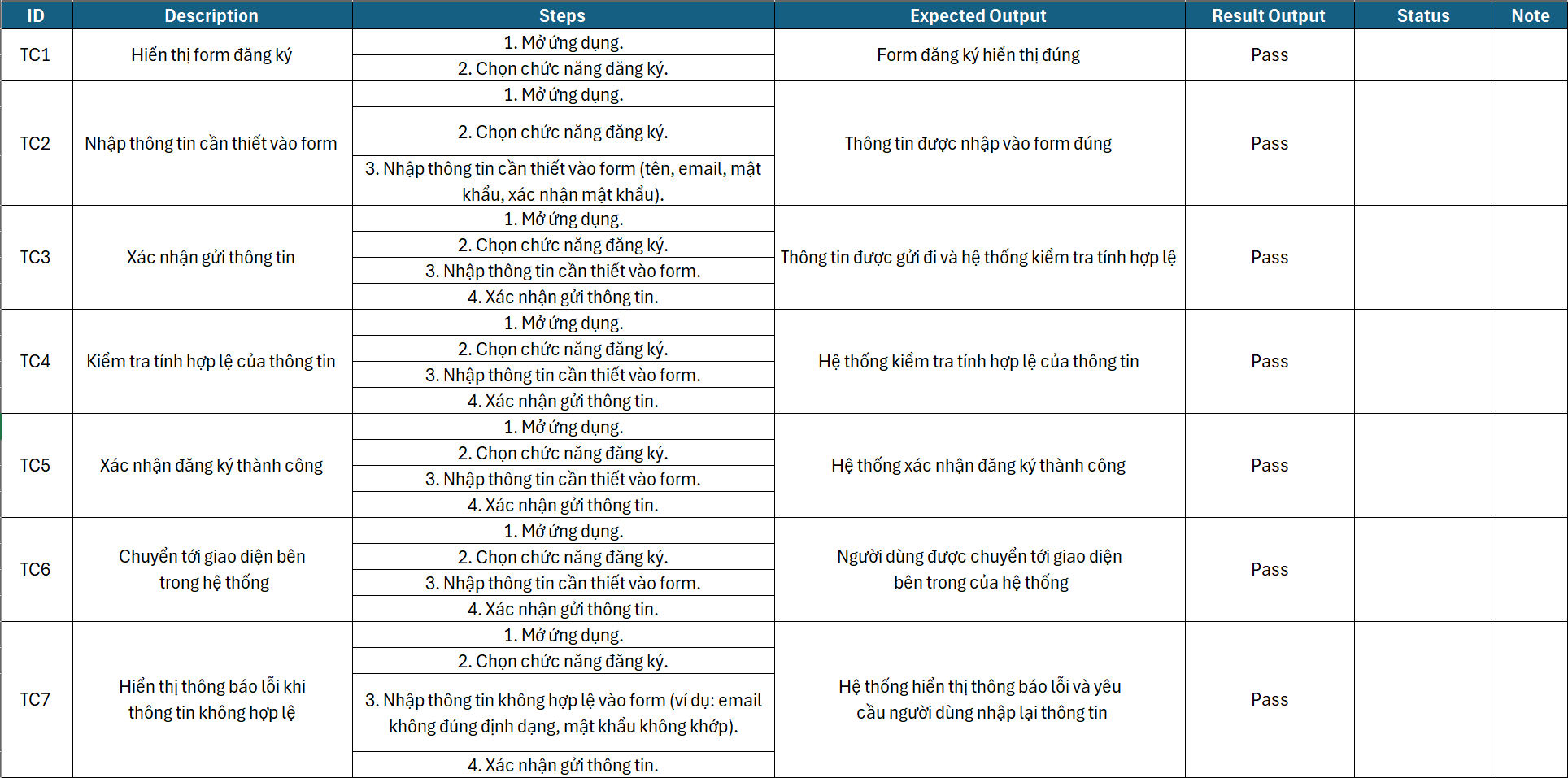
**Quy trình kiểm thử bao gồm:**

* Thiết kế testcase: Xây dựng các testcase chi tiết cho từng chức năng và giao diện của ứng dụng.
* Thực hiện kiểm thử: Thực hiện các testcase trên ứng dụng để kiểm tra tính chính xác và hiệu quả của các chức năng.
* Báo cáo kết quả: Ghi nhận kết quả kiểm thử, bao gồm các lỗi phát hiện được và đề xuất các giải pháp khắc phục.

Việc kiểm thử kỹ lưỡng giúp chúng tôi đảm bảo rằng ứng dụng không chỉ đáp ứng được các yêu cầu kỹ thuật mà còn mang lại trải nghiệm học tập tốt nhất cho người dùng.

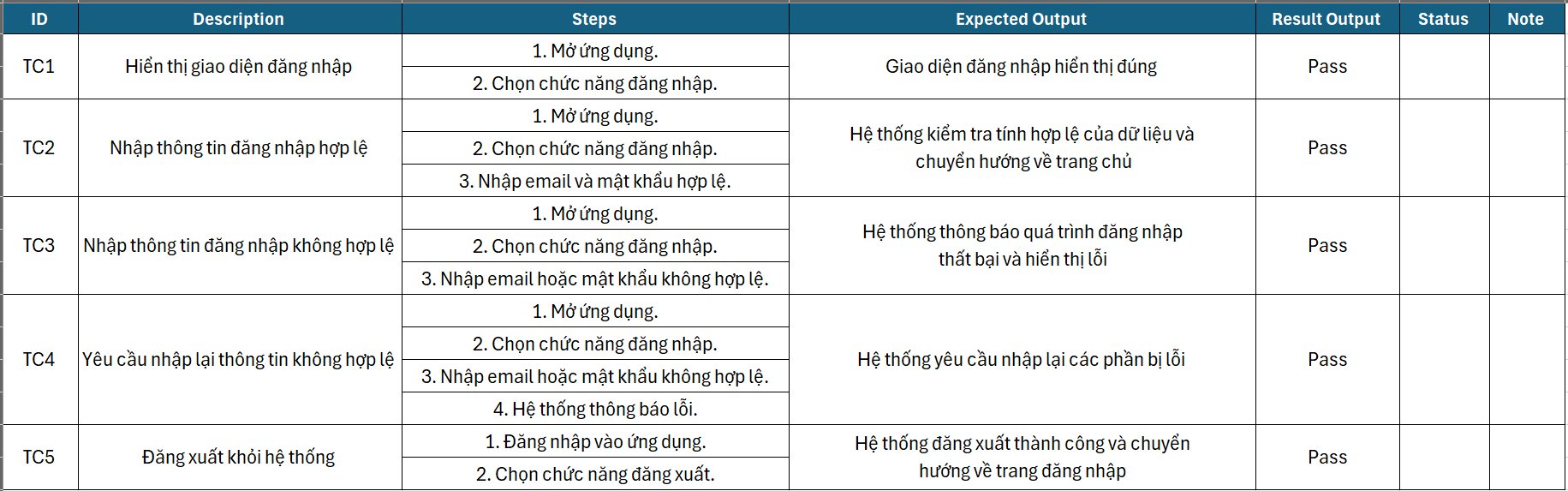
## Trường hợp kiểm thử (Test Case)

### Chức năng đăng ký



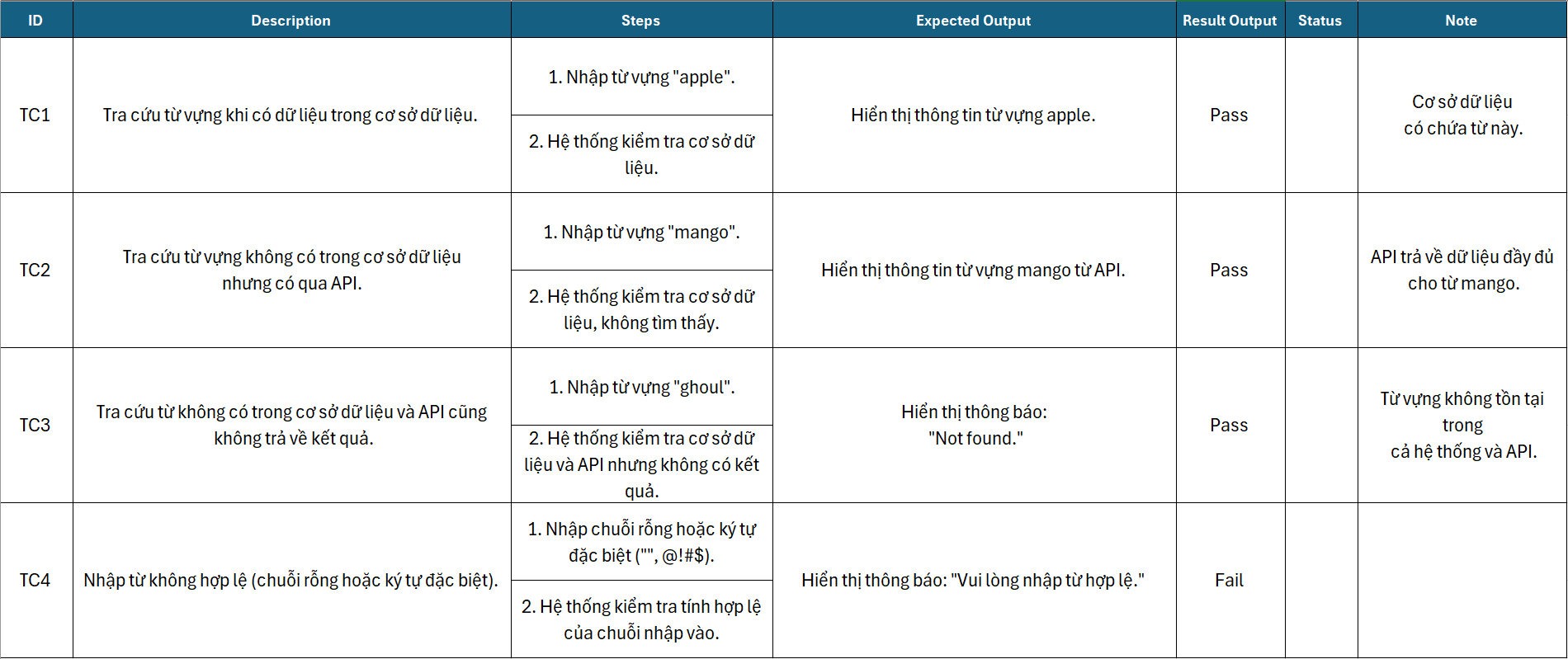
Hình 5-1: Testcase Đăng ký

### Chức năng đăng nhập



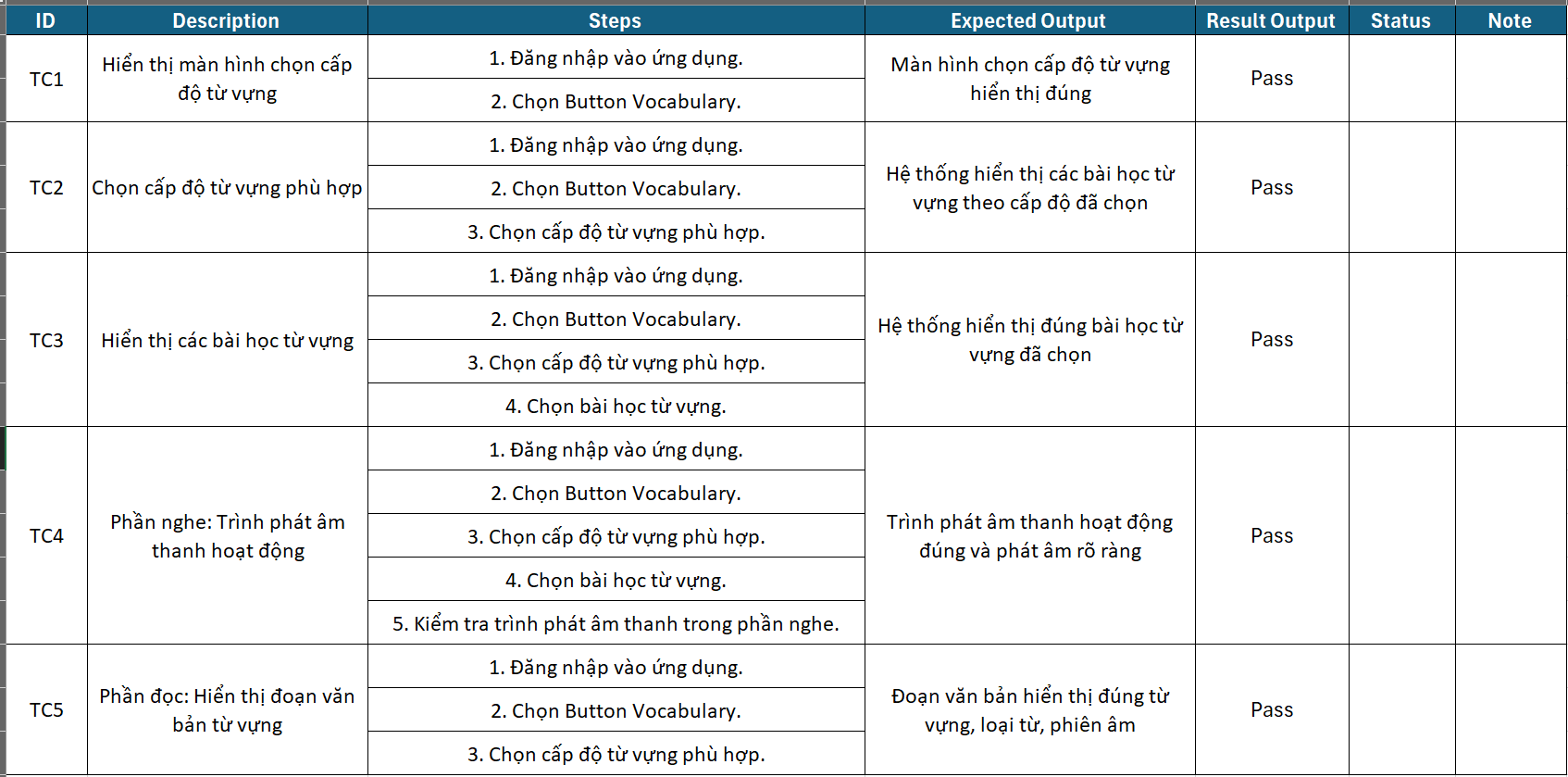
Hình 5-2: Testcase Đăng nhập/Đăng xuất

### Tra cứu từ điển, dịch từ vựng



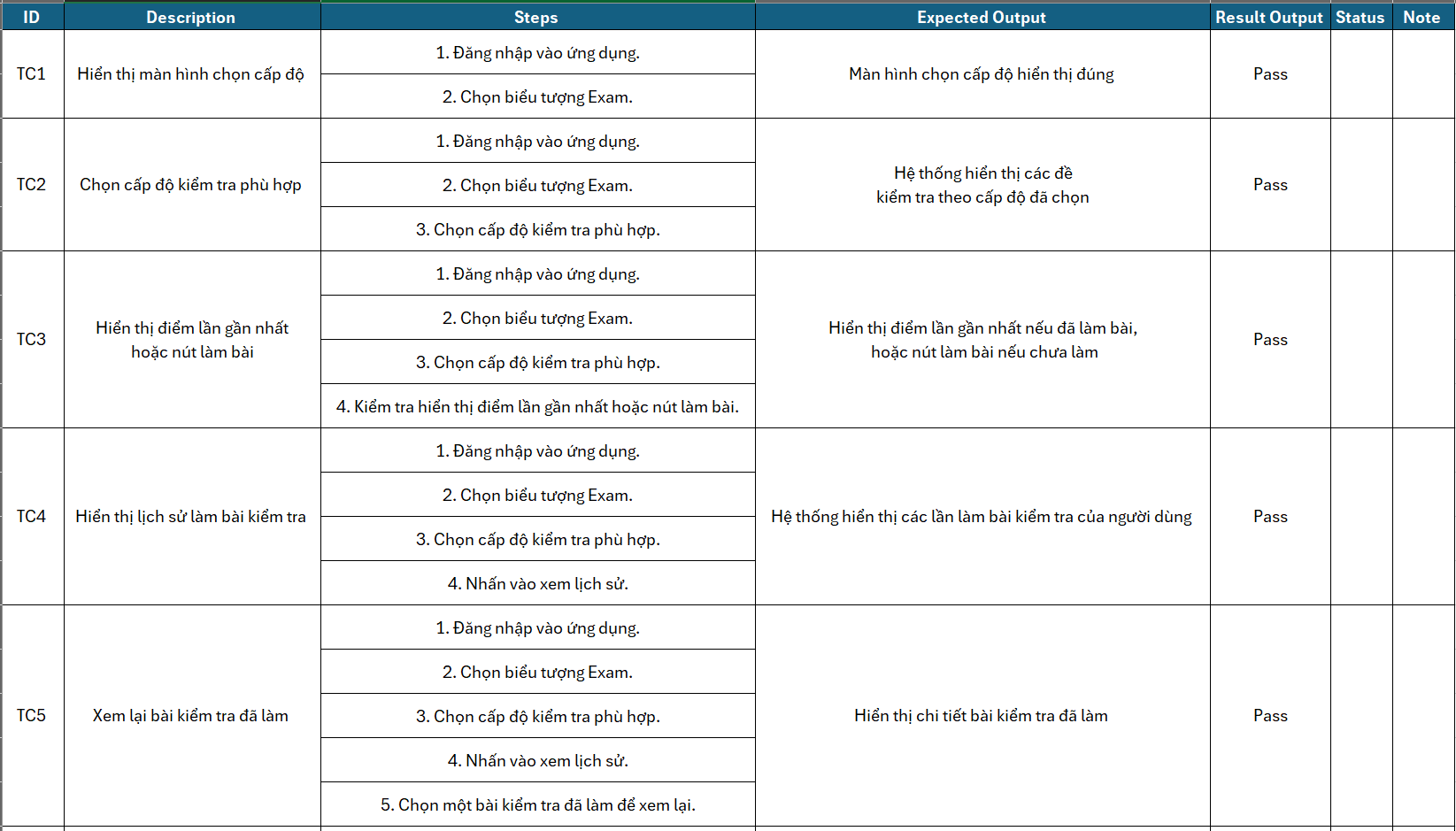
Hình 5-3: Testcase tra từ điển

### Học từ vựng

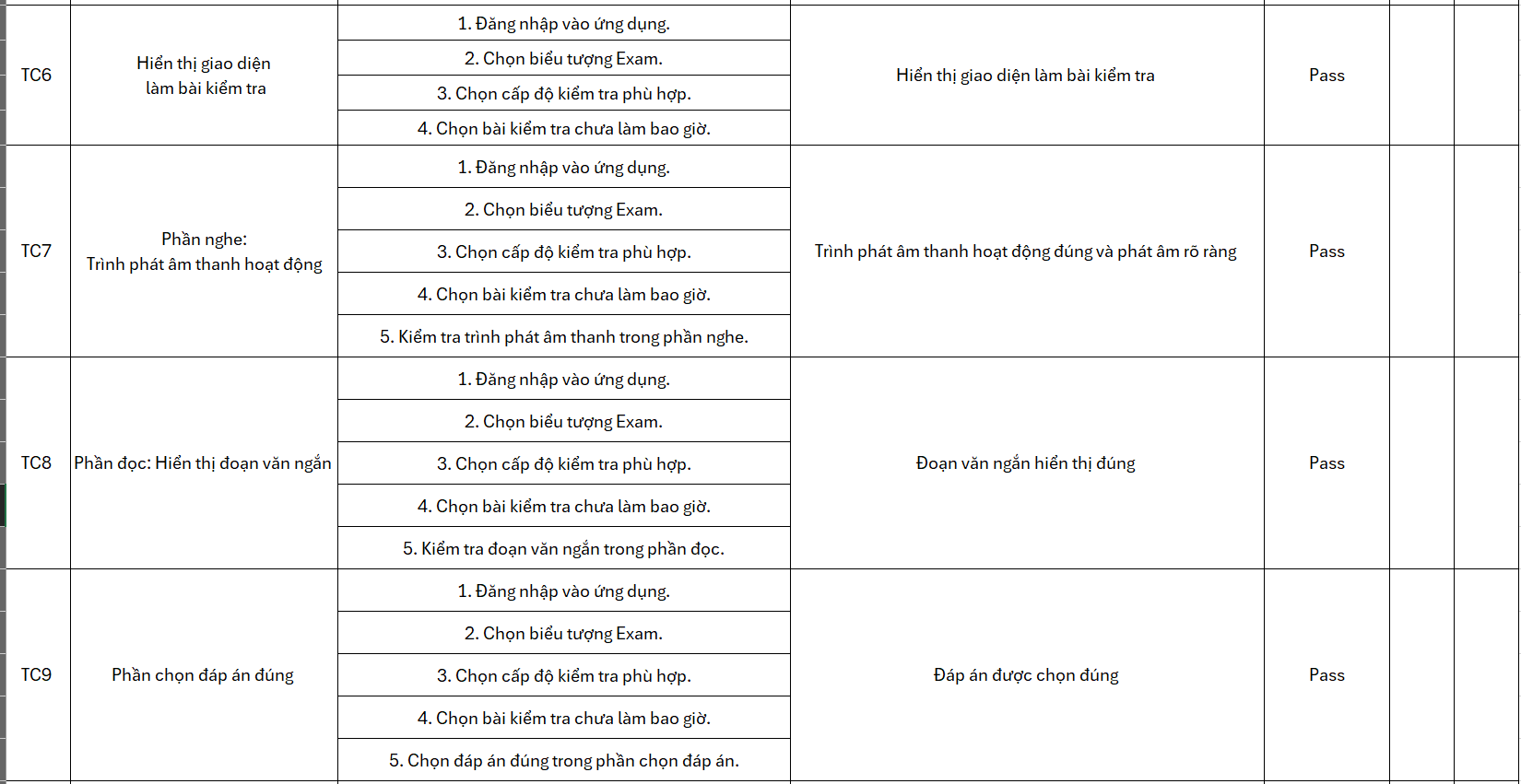


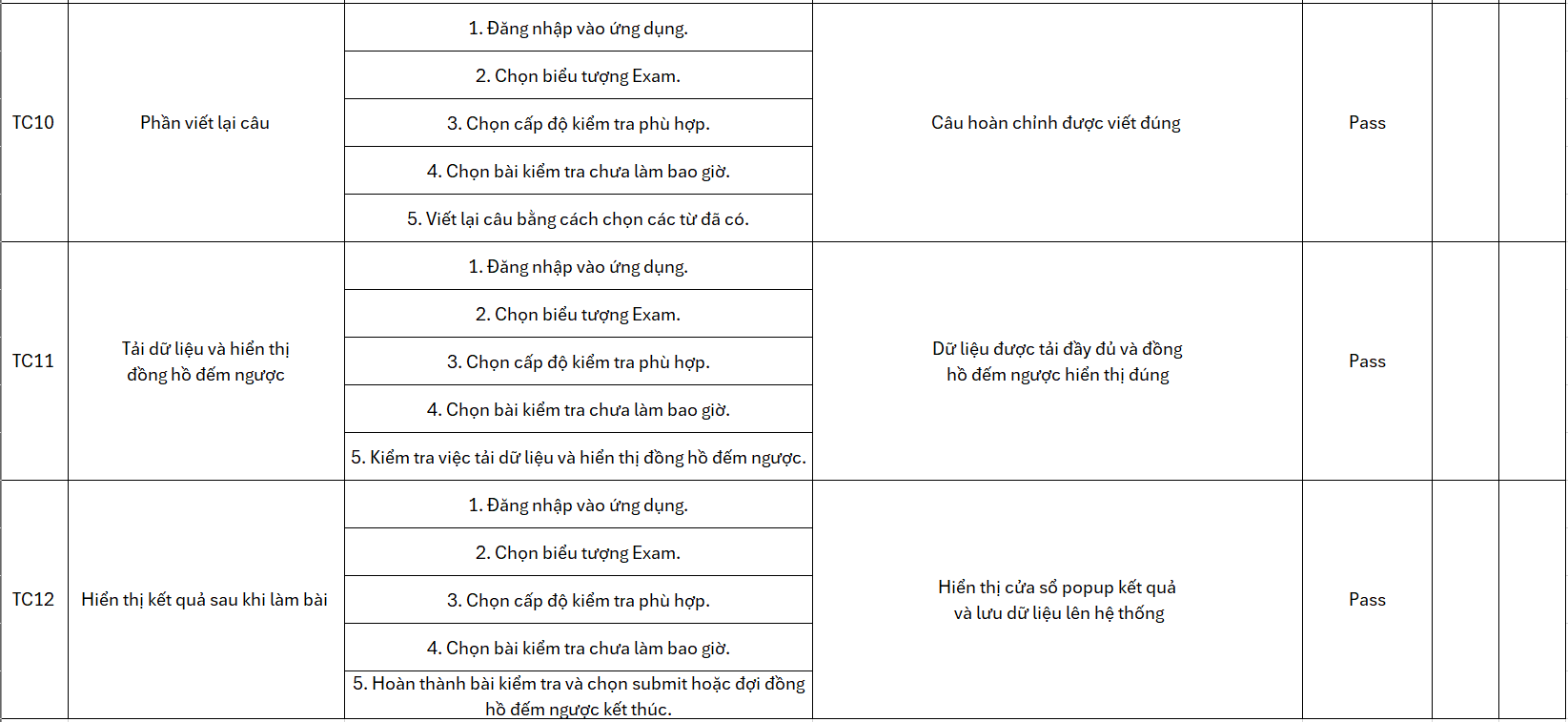
Hình 5-4: Testcase học từ vựng

### Làm bài kiểm tra



Hình 5-5: Testcase làm bài kiểm tra





# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

Với sự hướng dẫn nhiệt tình của các thầy/cô trong Khoa Công nghệ Thông tin, đặc biệt là sự chỉ bảo tận tình của thầy **Hoàng Văn Thông**, chúng em đã có cơ hội học hỏi, tích lũy nhiều kiến thức bổ ích và đạt được những kết quả quan trọng trong quá trình thực hiện đề tài “**Ứng dụng học tiếng Anh - Eng2UTC**”.

Thông qua dự án, chúng em đã:

* **Nắm vững quy trình xây dựng ứng dụng di động thực tế**: Từ bước khảo sát, phân tích, thiết kế cho đến việc triển khai và phát triển chức năng. Điều này giúp chúng em hiểu rõ hơn về toàn bộ chu trình phát triển phần mềm, từ ý tưởng ban đầu đến sản phẩm hoàn thiện.
* **Học cách phân tích và thiết kế hệ thống**: Thực hành vẽ và đặc tả các **Use Case**, xây dựng **biểu đồ hoạt động** (Activity Diagram), và **biểu đồ tuần tự** (Sequence Diagram) để mô phỏng quy trình hoạt động của ứng dụng.
* **Tích lũy thêm kiến thức về công nghệ và thư viện mới**: Sử dụng các công nghệ hiện đại như **Firebase** cho việc lưu trữ dữ liệu, xác thực người dùng và dịch thuật ngôn ngữ. Đồng thời, học cách tích hợp và xử lý các thư viện hỗ trợ cho ứng dụng di động, như **Firebase ML Kit** cho tính năng dịch thuật và **Android Jetpack** để tối ưu giao diện người dùng.
* **Phát triển kỹ năng gỡ lỗi và tối ưu chương trình**: Chúng em không chỉ học cách viết mã, mà còn biết cách kiểm tra, sửa lỗi và tối ưu hiệu suất ứng dụng để đảm bảo chất lượng sản phẩm cuối cùng.

**Những điểm chương trình làm được:**

* **Giao diện thân thiện:** Thiết kế đơn giản, dễ sử dụng, phù hợp với đa dạng đối tượng người học.
* **Tích hợp Firebase:** Lưu trữ dữ liệu người dùng an toàn, đáng tin cậy.
* **Dịch thuật ngôn ngữ:** Tính năng dịch sử dụng Firebase ML Kit hỗ trợ hai ngôn ngữ chính (Anh - Việt).
* **Bài học chuẩn hóa:** Nội dung ngữ pháp và từ vựng tuân thủ CEFR, giúp người học theo dõi tiến trình.
* **Kiểm tra đánh giá:** Bài test đa dạng với các kỹ năng nghe, đọc, viết giúp người học tự đánh giá.

**Những nhược điểm của hệ thống:**

* **Hệ thống:** Chưa xử lý được hết các exception của hệ thống.
* **Bảo mật:** Chưa có những phương pháp nâng cao bảo mật hệ thống.
* **Phạm vi ngôn ngữ hạn chế:** Hiện tại chỉ hỗ trợ tiếng Anh và tiếng Việt.
* **Hiệu suất Firebase ML:** Quá trình tải và sử dụng mô hình dịch thuật đôi khi bị chậm khi kết nối mạng yếu.
* **Chưa có cộng đồng học tập:** Thiếu các tính năng tương tác giữa người dùng như chia sẻ bài học hoặc giao lưu.
* **Thiếu gamification:** Chưa có hệ thống huy hiệu hoặc phần thưởng để tăng động lực học.

## Hướng phát triển

**Bổ sung công nghệ:**

* **AI và cá nhân hóa học tập:** Sử dụng AI để gợi ý bài học, bài tập phù hợp dựa trên kết quả và sở thích học tập của người dùng.
* **Speech Recognition nâng cao:** Tích hợp AI để phân tích và phản hồi về phát âm của người dùng.
* **Machine Learning:** Cải thiện các mô hình dịch thuật và đề xuất nội dung học phù hợp hơn.
* **Progressive Web App (PWA):** Phát triển phiên bản ứng dụng web để sử dụng đa nền tảng.
* **Cloud Computing:** Tăng khả năng lưu trữ và đồng bộ dữ liệu giữa các thiết bị.

**Tính năng mới:**

* Thêm lớp học trực tuyến (live classes) với giáo viên.
* Tích hợp chatbot AI hỗ trợ người học 24/7.
* Phát triển module luyện thi các chứng chỉ quốc tế như TOEFL, IELTS, TOEIC.

Mặc dù đã đạt được những kết quả khả quan, chúng em nhận thức rõ rằng ứng dụng vẫn còn những hạn chế cần khắc phục và hoàn thiện. Đây cũng chính là động lực để chúng em tiếp tục cải thiện ứng dụng, nghiên cứu thêm các công nghệ mới và mở rộng tính năng nhằm đáp ứng tốt hơn nhu cầu thực tế của người dùng.

Một lần nữa, chúng em xin chân thành cảm ơn sự hỗ trợ tận tình của các thầy/cô trong khoa, đặc biệt là thầy **Hoàng Văn Thông**, đã giúp chúng em hoàn thành đề tài này. **Eng2UTC** không chỉ là một sản phẩm học thuật mà còn là minh chứng cho sự tiến bộ trong quá trình học tập và nghiên cứu của chúng em.

TÀI LIỆU THAM KHẢO

**● Android Developers Website**

[*https://developer.android.com*](https://developer.android.com)

**● Firebase Website**

[*https://firebase.google.com*](https://firebase.google.com)

**● Mô hình MVVM (Model-View-ViewModel)**

[*https://learn.microsoft.com/en-us/dotnet/architecture/maui/mvvm*](https://learn.microsoft.com/en-us/dotnet/architecture/maui/mvvm)