

UBND TỈNH NINH BÌNH
TRƯỜNG ĐẠI HỌC HOA LƯ

**BÁO CÁO KẾT QUẢ THỰC HIỆN
NHIỆM VỤ KHOA HỌC VÀ CÔNG NGHỆ CẤP CƠ SỞ**

**XÂY DỰNG PHẦN MỀM ỨNG DỤNG BỎ PHIẾU
TRÊN THIẾT BỊ DI ĐỘNG**

**Chủ nhiệm nhiệm vụ: ThS. ĐẶNG THỊ THU HÀ
Đơn vị: KHOA NGOẠI NGỮ - CÔNG NGHỆ THÔNG TIN**

NINH BÌNH, 2024

UBND TỈNH NINH BÌNH
TRƯỜNG ĐẠI HỌC HOA LƯ

**BÁO CÁO KẾT QUẢ THỰC HIỆN
NHIỆM VỤ KHOA HỌC VÀ CÔNG NGHỆ CẤP CƠ SỞ**

**XÂY DỰNG PHẦN MỀM ỨNG DỤNG BỎ PHIẾU
TRÊN THIẾT BỊ DI ĐỘNG**

Chủ nhiệm nhiệm vụ: Ths. Đặng Thị Thu Hà

Đơn vị: Khoa Ngoại ngữ - Công nghệ thông tin

Thành viên: Ths. Đào Sỹ Nhiên

Thành viên: Ths. Nguyễn Thị Thu Hà

Đơn vị: Khoa Ngoại ngữ - Công nghệ thông tin

Thành viên: Ths. Bùi Thị Tuyết

Đơn vị: Phòng Đào tạo - Quản lý khoa học

Thành viên: Ths. Phạm Xuân Nguyên

Đơn vị: Trung tâm Ngoại ngữ - Tin học

Xác nhận của Chủ tịch HĐ nghiệm thu

Chủ nhiệm đề tài

NINH BÌNH, 2024

MỤC LỤC

MỞ ĐẦU	1
1. Tính cấp thiết của đề tài	1
2. Tổng quan tình hình nghiên cứu	1
3. Mục tiêu.....	3
4. Đối tượng và phạm vi nghiên cứu của đề tài	3
4.1. Đối tượng nghiên cứu.....	3
4.2. Phạm vi nghiên cứu.....	4
5. Cách tiếp cận và phương pháp nghiên cứu của đề tài.....	4
5.1. Cách tiếp cận	4
5.2. Phương pháp nghiên cứu.....	4
NỘI DUNG NGHIÊN CỨU	5
CHƯƠNG 1: CƠ SỞ LÝ LUẬN CỦA ĐỀ TÀI	5
1.1. Thực trạng công tác bỏ phiếu tại Trường Đại học Hoa Lư.....	5
1.2. Phân tích, đánh giá về phần mềm ứng dụng (App) để tiến hành bỏ phiếu trên các thiết bị di động	6
1.3. Hướng xây dựng phần mềm ứng dụng (App) để bỏ phiếu trên thiết bị di động tại Trường Đại học Hoa Lư.....	8
1.3.1. Chức năng của app bỏ phiếu trên thiết bị di động (Mobile)	8
1.3.2. Phạm vi.....	8
1.3.3. Phân quyền	8
CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ PHẦN MỀM.....	9
2.1. Phân tích hệ thống phần mềm	9
2.1.1. Xác định yêu cầu	9
2.1.2. Thiết kế kiến trúc hệ thống.....	9
2.1.3. Thiết kế giao diện người dùng (UI/UX)	9
2.1.4. Đảm bảo bảo mật.....	10
2.1.5. Kiểm thử hệ thống.....	10
2.1.6. Triển khai và bảo trì	10
2.2. Thiết kế hệ thống theo hướng chức năng.....	10

2.2.1. Phân tích chức năng của hệ thống.....	10
2.2.2. Sơ đồ chức năng	11
2.2.3. Mô tả chi tiết các chức năng	14
2.3. Thiết kế Cơ sở dữ liệu	20
2.3.1. Các Bảng (table) của hệ thống	21
2.3.2. Mô tả mối quan hệ giữa các Bảng.....	22
2.3.3. Mô hình ERD (Entity-Relationship Diagram).....	24
2.4. Các công cụ xây dựng hệ thống phần mềm	25
2.4.1. Hệ quản trị cơ sở dữ liệu MySQL.....	25
2.4.2. Ngôn ngữ lập trình API (Application Programming Interface) -Laravel .	26
2.4.3. Ngôn ngữ lập trình Flutter.....	27
CHƯƠNG 3. XÂY DỰNG PHẦN MỀM	29
3.1. Xây dựng hệ thống Cơ sở dữ liệu	29
3.1.1. Tạo các bảng cơ sở dữ liệu trên hệ quản trị cơ sở dữ liệu MySQL	29
3.1.2. Mô hình ERD (Entity-Relationship Diagram) với bảng dữ liệu ở trên và mối quan hệ giữa chúng	34
3.2. Xây dựng các ứng dụng bằng Laravel	34
3.3. Xây dựng giao diện người dùng bằng Flutter	40
3.4. Cài đặt thử nghiệm	47
3.4.1. Mục tiêu thử nghiệm	47
3.4.2. Quy trình cài đặt thử nghiệm	47
3.4.3. Giao diện phần mềm	49
3.4.4. Đánh giá kết quả.....	53
KẾT LUẬN VÀ KIẾN NGHỊ	54
TÀI LIỆU THAM KHẢO	56

DANH MỤC HÌNH ẢNH

Hình 2.1. Sơ đồ chức năng của hệ thống	11
Hình 2.2. Biểu đồ Use Case của hệ thống.....	13
Hình 3. 1. Mô hình cơ sở dữ liệu “voting_hluv” dưới dạng ERD	34
Hình 3. 2. Form đăng nhập hệ thống.....	49
Hình 3. 3. Bảng điều khiển.....	50
Hình 3. 4. Quản lý danh sách cuộc bỏ phiếu.....	50
Hình 3. 5. Quản lý danh sách ứng cử viên	50
Hình 3. 6. Tạo cuộc bầu cử mới	51
Hình 3. 7. Xem thông tin ứng cử viên.....	51
Hình 3. 8. Xóa thông tin ứng cử viên.....	51
Hình 3. 9. Quản lý ứng cử viên và cuộc bỏ phiếu tương ứng	52
Hình 3. 10. Thống kê kết quả cuộc bỏ phiếu	52
Hình 3. 11. Đăng nhập hệ thống	52
Hình 3. 12. Giao diện thể hiện Danh sách các cuộc bầu cử.....	53
Hình 3. 13. Giao diện thể hiện về các ứng cử viên	53

THÔNG TIN KẾT QUẢ NGHIÊN CỨU

Ngày nay, các ứng dụng di động đã nổi lên như một thành phần thiết yếu trong nỗ lực chiến lược của bất kỳ tổ chức nào nhằm tiếp cận và tương tác với đối tượng mục tiêu. Với điện thoại thông minh và máy tính bảng đã trở thành phương tiện thống trị để liên lạc, giải trí và năng suất, việc phát triển các ứng dụng dành cho thiết bị di động phục vụ cho các nhu cầu và sở thích khác nhau là một nỗ lực đầy thách thức và bổ ích. Thiết kế Mobile App giúp tối ưu hóa quy trình vận hành và quản lý, nâng cao hiệu suất công việc, từ đó tăng trải nghiệm người dùng là cần thiết.

Ứng dụng (App) bỏ phiếu trên Mobile mang lại nhiều tiện lợi, dễ dàng cho người sử dụng, khắc phục khó khăn về địa lý, tiết kiệm thời gian, chi phí. Hơn nữa nó còn giúp các nhà quản lý có thể thu thập dữ liệu ý kiến người dùng một cách nhanh chóng, hiệu quả chính xác, dễ dàng xem xét, theo dõi và phân tích kết quả.

Trường Đại học Hoa Lư đang tổ chức đào tạo ngành Công nghệ thông tin, việc nghiên cứu, áp dụng các bài toán thực tế và ứng dụng các ngôn ngữ mới để xây dựng phần mềm ứng dụng chạy trên các nền tảng di động là rất cần thiết, giúp ích rất nhiều trong việc nâng cao trình độ chuyên môn, đồng thời nâng cao chất lượng đào tạo của Nhà trường.

Kết quả nghiên cứu của đề tài đạt được đó là:

- Giúp giảng viên công nghệ thông tin cập nhật, rèn luyện kỹ năng với các ngôn ngữ lập trình trên thiết bị di động, gồm: framework tạo ứng dụng web Laravel; ngôn ngữ lập trình Draft, Flutter.

- Thiết kế, xây dựng phần mềm ứng dụng chạy trên thiết bị di động.

Những đóng góp về giáo dục, đào tạo và tính dụng của đề tài, cụ thể như sau:

- Sản phẩm của đề tài gắn liền với hoạt động đào tạo ngành công nghệ thông tin, giúp ích việc nâng cao công tác giảng dạy học phần chuyên ngành.

- Kết quả của đề tài là tài phần mềm ứng dụng chạy trên thiết bị di động có tính ứng dụng, không những áp dụng vào việc giảng dạy và nâng cao khả năng chuyên môn của giảng viên mà còn tạo ra hướng phát triển chuyên môn đối với bộ môn công nghệ thông tin.

MỞ ĐẦU

1. Tính cấp thiết của đề tài

Trên thế giới, chuyển đổi số bắt đầu được nhắc đến nhiều vào khoảng năm 2015, phổ biến từ năm 2017. Ở Việt Nam, chuyển đổi số bắt đầu được nhắc đến nhiều vào khoảng năm 2018. [1] Thủ tướng Chính phủ phê duyệt Chương trình Chuyển đổi số quốc gia vào ngày 03/6/2020. Chuyển đổi số là bước phát triển tiếp theo của tin học hóa, có được nhờ vào sự tiến bộ vượt bậc của những công nghệ mới mang tính đột phá, nhất là công nghệ số. Chuyển đổi số là quá trình thay đổi tổng thể và toàn diện của cá nhân, tổ chức về cách sống, cách làm việc và phương thức sản xuất dựa trên các công nghệ số.

Chuyển đổi số trong giáo dục và đào tạo [1] bao gồm chuyển đổi số trong hoạt động dạy - học và chuyển đổi số trong hoạt động quản trị - quản lý các nhà trường. Ngành giáo dục và đào tạo có nhiều thuận lợi nhưng thách thức cũng rất lớn đó là ý chí, nhận thức, văn hóa, hạ tầng số, nhân lực, cơ chế chính sách, số hóa dữ liệu, quy trình, dịch vụ.

Ngày nay, các ứng dụng di động đã nổi lên như một thành phần thiết yếu trong nỗ lực chiến lược của bất kỳ tổ chức nào nhằm tiếp cận và tương tác với đối tượng mục tiêu. Với điện thoại thông minh và máy tính bảng đã trở thành phương tiện thống trị để liên lạc, giải trí và năng suất, việc phát triển các ứng dụng dành cho thiết bị di động phục vụ cho các nhu cầu và sở thích khác nhau là một nỗ lực đầy thách thức và bổ ích. Thiết kế Mobile App giúp tối ưu hóa quy trình vận hành và quản lý, nâng cao hiệu suất công việc, từ đó tăng trải nghiệm người dùng là cần thiết.

App bỏ phiếu trên Mobile mang lại nhiều tiện lợi, dễ dàng cho người sử dụng, khắc phục khó khăn về địa lý, tiết kiệm thời gian, chi phí. Hơn nữa nó còn giúp các nhà quản lý có thể thu thập dữ liệu ý kiến người dùng một cách nhanh chóng, hiệu quả chính xác, dễ dàng xem xét, theo dõi và phân tích kết quả.

Trước những nhu cầu của thực tế và để nắm bắt xu hướng của xã hội, trong khi Trường Đại học Hoa Lư đang tổ chức đào tạo ngành Công nghệ thông tin, việc nghiên cứu, áp dụng các bài toán thực tế và ứng dụng các ngôn ngữ mới để xây dựng phần mềm ứng dụng chạy trên các nền tảng di động là rất cần thiết, giúp ích rất nhiều trong việc nâng cao trình độ chuyên môn, đồng thời nâng cao chất lượng đào tạo của Nhà trường.

2. Tổng quan tình hình nghiên cứu

Ngày nay xu hướng công nghệ thế giới nói chung và Việt nam nói riêng đang dần chuyển sang phát triển Mobile App nhiều hơn. Trong vài năm tới đây

thì hầu như các ngành công nghiệp cũng sẽ đưa mobile app trở thành chiến lược chủ đạo để tạo bước tiến mới trong kinh doanh thời kỳ chuyển đổi số.

Mobile App (ứng dụng di động) là phần mềm được thiết kế để chạy trên các thiết bị di động như điện thoại thông minh và máy tính bảng. Những ứng dụng này có thể cung cấp nhiều dịch vụ và giải pháp đa dạng, từ giải trí, mua sắm trực tuyến, dịch vụ ngân hàng, đến giáo dục và y tế. Việc phát triển Mobile App mang đến cơ hội tiếp cận người dùng một cách tiện lợi và đơn giản.

Bỏ phiếu điện tử là việc thực hiện quyền biểu quyết theo hình thức điện tử thông qua hệ thống kỹ thuật số thay vì trên giấy. Bỏ phiếu điện tử đang được sử dụng rộng rãi thay vì bỏ phiếu truyền thống như trước đây bởi chúng mang lại nhiều lợi ích. Ứng dụng bỏ phiếu trên điện thoại di động là các ứng dụng được phát triển để giúp người dùng tham gia vào quá trình bầu cử hoặc tham gia vào các cuộc khảo sát, thăm dò ý kiến, hoặc bất kỳ quyết định nào khác yêu cầu thu thập ý kiến của cộng đồng hoặc đám đông.

Tại Việt Nam bỏ phiếu điện tử mới chỉ dừng ở mục đích bầu chọn, bình chọn (bầu chọn Vịnh Hạ Long là di sản Thiên nhiên thế giới, bình bài hát hay trên sóng truyền hình, bình chọn hoa hậu, ca sỹ..) song chưa thể triển khai vào các cuộc bầu cử quan trọng do còn nhiều hạn chế. Bỏ phiếu kín là phương pháp bỏ phiếu mà lựa chọn của người bỏ phiếu trong cuộc biểu quyết sẽ được giữ bí mật, đồng thời người bỏ phiếu sẽ không viết tên mình lên phiếu bầu. Đây là hình thức bỏ phiếu đang được nhiều cơ quan đoàn thể và doanh nghiệp áp dụng vì sẽ giúp ngăn ngừa những tác động bên ngoài gây ảnh hưởng đến người bỏ phiếu.

Công đoàn EVNHCMC đã phối hợp với Công ty Công nghệ thông tin EVNHCMC và Tập đoàn Bưu chính - Viễn thông Việt Nam (VNPT) xây dựng phần mềm tổ chức đại hội công đoàn. Phần mềm được áp dụng tại tất cả đại hội công đoàn cơ sở trực thuộc Công đoàn EVNHCMC cũng như tại Đại hội đại biểu Công đoàn EVNHCMC. Theo đó, các hoạt động trong đại hội như điểm danh, kiểm tra số lượng đại biểu tham dự, hướng dẫn sơ đồ chỗ ngồi, kiểm tra số phiếu bầu cử đều được các đại biểu thực hiện qua App Công đoàn EVNHCMC. Việc “số hóa” công tác tổ chức đại hội công đoàn là yêu cầu cần thiết và phù hợp với xu thế phát triển chung, góp phần tích cực, hiệu quả vào việc thực hiện chủ trương chuyển đổi số, phát triển kinh tế số và xã hội số của Chính phủ.

Trên thế giới một số nước như Mỹ, Anh, Thụy Sĩ đã triển khai bỏ phiếu trên mobile dưới dạng thăm dò dư luận, hay khảo sát người dân về sở thích hoặc thăm dò ý kiến của họ trước cuộc tuyển cử.

Hàn Quốc đã triển khai ứng dụng di động cho bỏ phiếu trực tuyến trong bầu cử quốc hội và các cuộc trưng cầu dân ý khác. Ứng dụng bỏ phiếu trên điện thoại mVoting được coi là một trong những dịch vụ của chính phủ điện tử đại diện cho

chính quyền Seoul. Ứng dụng này đã sáu lần liên tiếp đứng thứ nhất trong bảng đánh giá chính phủ điện tử của các thành phố lớn trên thế giới, mVoting là sự kết hợp kết hợp di động và bỏ phiếu. Tại Hàn Quốc, đã có hơn 37 triệu điện thoại thông minh đang được sử dụng ở Hàn Quốc trong giai đoạn 2015 - 2017, ứng dụng điện thoại thông minh này được phát triển để thu thập ý kiến của công dân một cách nhanh chóng bằng cách sử dụng bỏ phiếu trực tuyến theo thời gian thực. Công dân có thể mở và tham gia trực tiếp không chỉ vào các phiếu bình chọn chính sách mà còn trong các phiếu bình chọn gắn với tất cả các loại tình hình thực tế. Bỏ phiếu có mục tiêu có thể tiến hành cho các nhóm riêng hoặc cho công dân trong một khoảng cách xác định. Với các ưu việt như vậy, ứng dụng mVoting đã thu hút tất cả mọi người dân cùng tham gia vào xây dựng chính sách phát triển chung của xã hội, chính phủ. Người dân có thể tham gia (vào) các vấn đề chính sách mà có thể gắn với chính cuộc sống của mình, chẳng hạn như các tuyến xe buýt, hạn chế lái xe khi có nhiều ô nhiễm không khí hoặc chỉ định các khu vực cấm hút thuốc trong công viên. Đó thực sự là một cách để công dân bày tỏ ý kiến của họ về các dịch vụ mà họ tiếp xúc nhiều nhất hoặc ảnh hưởng nhiều nhất đến họ và gia đình của họ. Các phiếu bầu trên mVoting không được bao gồm trong các cuộc bầu cử "thực". Ứng dụng này chủ yếu dành cho người dân để đưa ra phản hồi cho chính quyền thành phố về tất cả các vấn đề, dù là nhỏ nhất hay nghiêm trọng.

3. Mục tiêu

- Giúp giảng viên công nghệ thông tin cập nhật, nâng cao kỹ năng sử dụng các ngôn ngữ lập trình trên thiết bị di động, gồm: framework tạo ứng dụng web Laravel; ngôn ngữ lập trình Draft, Flutter.

- Phân tích, thiết kế và xây dựng phần mềm ứng dụng để bỏ phiếu trên các thiết bị di động hướng tới ứng dụng cho việc bỏ phiếu tại Trường Đại học Hoa Lư.

4. Đối tượng và phạm vi nghiên cứu của đề tài

4.1. Đối tượng nghiên cứu

- Nghiên cứu về quy trình bỏ phiếu tại Trường Đại học Hoa Lư.

- Nghiên cứu các công cụ để xây dựng phần mềm gồm:

+ Framework xây dựng giao diện người dùng trên nhiều nền tảng

+ Ngôn ngữ lập trình cho ứng dụng Mobile

+ Bộ công cụ lập trình Visual Studio

+ Hệ quản trị cơ sở dữ liệu

+ Framework xây dựng ứng dụng web

+ Nghiên cứu xây dựng các chức năng của phần mềm.

4.2. Phạm vi nghiên cứu

- Phần mềm hỗ trợ hoạt động bỏ phiếu tại trường Đại học Hoa Lu.
- Framework xây dựng giao diện người dùng trên nhiều nền tảng: Flutter
- Ngôn ngữ lập trình cho ứng dụng mobile: Draft
- Hệ quản trị cơ sở dữ liệu: My SQL
- Framework xây dựng ứng dụng web: Laravel 10
- Quy trình xây dựng App bỏ phiếu trên Mobile

5. Cách tiếp cận và phương pháp nghiên cứu của đề tài

5.1. Cách tiếp cận

Tiếp cận từ lý thuyết → ứng dụng vào thực tiễn → giải pháp công nghệ

- Nghiên cứu về quy trình bỏ phiếu, các loại báo cáo kết quả bỏ phiếu tại Trường Đại học Hoa Lu.

- Nghiên cứu các app trên điện thoại có chức năng bỏ phiếu tương tự.
- Nghiên cứu các công cụ để xây dựng phần mềm.
- Xây dựng thử nghiệm phần mềm, chạy thử và đánh giá, chỉnh sửa.

5.2. Phương pháp nghiên cứu

- Phương pháp nghiên cứu lý thuyết: nghiên cứu về quy trình bỏ phiếu, các mẫu biểu báo cáo bỏ phiếu tại Trường Đại học Hoa Lu; các App bỏ phiếu tương tự trên mobile và các công cụ để thực hiện.

- Phương pháp thực nghiệm: xây dựng phần mềm chạy và chỉnh sửa.

NỘI DUNG NGHIÊN CỨU

CHƯƠNG 1: CƠ SỞ LÝ LUẬN CỦA ĐỀ TÀI

1.1. Thực trạng công tác bỏ phiếu tại Trường Đại học Hoa Lư

Bỏ phiếu kín là phương pháp bỏ phiếu trong đó lựa chọn của người bỏ phiếu trong một cuộc bầu cử hay trưng cầu dân ý được giữ bí mật và người bỏ phiếu không viết tên của mình lên trên phiếu. Việc bỏ phiếu kín giúp ngăn ngừa các tác động bên ngoài ảnh hưởng đến người bỏ phiếu.

Bỏ phiếu kín là phương pháp bỏ phiếu phù hợp với nhiều hệ thống bỏ phiếu khác nhau. Hình thức đơn giản nhất là dùng các mảnh giấy trắng và người bỏ phiếu chỉ viết lên đó lựa chọn của mình.

Người bỏ phiếu sẽ không cho ai biết sự lựa chọn của mình và bỏ các lá phiếu đó vào trong một hòm kín, hòm này sau đó sẽ được mở ra để kiểm đếm.

Hiện nay, một trong những hình thức bỏ phiếu kín thường gặp nhất là in sẵn những lá phiếu trên đó có tên của các ứng cử viên. Các điểm bỏ phiếu sẽ cung cấp những lá phiếu này cho người bỏ phiếu để họ bí mật lựa chọn.

Công tác bỏ phiếu tại Trường Đại học Hoa Lư thường được sử dụng trong Đại hội Đảng của Đảng ủy Trường Đại học Hoa Lư, Đại hội các Chi bộ, Hội nghị của Hội đồng trường, Đại hội Công đoàn Trường, hội nghị bình bầu các danh hiệu thi đua... cách thức tiến hành thường được thực hiện thành 02 cấp, đó là: cấp đơn vị và cấp trường; hình thức bỏ phiếu: Đồng ý/Không đồng ý và Chọn X ứng cử viên trong Y ứng cử viên ($X \leq Y$).

Dưới đây là một mô tả chi tiết về từng bước trong công tác bỏ phiếu tại Trường Đại học Hoa Lư:

Chuẩn bị:

- Phạm vi tại đơn vị đó là: các cuộc bỏ phiếu được thực hiện trong các hội nghị tại các đơn vị (phòng, khoa, trung tâm, ban).

- Phạm vi cấp trường đó là: các cuộc bỏ phiếu được thực hiện trong các Đại hội, hội nghị do Trường Đại học Hoa Lư tổ chức.

- Xác định danh sách người được bầu: căn cứ các qui định, tiêu chí về công tác Đảng, công tác của Hội đồng trường, công tác về Công đoàn trường và công tác về thi đua khen thưởng. Các đơn vị, Phòng ban chịu trách nhiệm tham mưu cho Nhà trường căn cứ theo qui định lập danh sách những người đủ điều kiện được đưa vào phiếu bầu.

- Hình thức bỏ phiếu: Đồng ý/Không đồng ý; chọn X ứng cử viên trong Y ứng cử viên ($X \leq Y$), trường hợp ($X=Y$) là bỏ phiếu không có số dư.

Thực hiện bỏ phiếu:

- Trường hợp hình thức bỏ phiếu Đồng ý/Không đồng ý thì trên phiếu thể hiện có các ô tương ứng để lựa chọn đồng ý hoặc không đồng ý và người bỏ phiếu tích vào ô tương ứng để thể hiện ý kiến bầu;

- Trường hợp chọn X ứng cử viên trong Y ứng cử viên thì trên phiếu chỉ thể hiện danh sách người được bầu và người bầu thực hiện việc lựa chọn bằng hình thức gạch ngang trường hợp mình không bỏ (không lựa chọn) và không gạch ngang trường hợp mình bỏ (lựa chọn).

Kiểm phiếu:

- Mở hòm phiếu: Sau khi kết thúc thời gian bỏ phiếu, hòm phiếu được mở dưới sự giám sát của các bên liên quan.

- Đếm phiếu: Các lá phiếu được đếm và kiểm tra để đảm bảo không có gian lận hoặc sai sót.

- Ghi nhận kết quả: Kết quả kiểm phiếu được ghi lại và công bố.

Công bố kết quả:

- Thông báo kết quả bầu cử: Kết quả bỏ phiếu được công bố theo qui định hoặc được thông tin chính thức.

- Giải quyết khiếu nại: Nếu có bất kỳ khiếu nại nào về quá trình bỏ phiếu, sẽ được xem xét và giải quyết theo quy định.

Lưu trữ:

- Lưu trữ hồ sơ: Tất cả các hồ sơ liên quan đến cuộc bỏ phiếu, bao gồm phiếu bầu và biên bản kiểm phiếu, được lưu trữ theo quy định.

1.2. Phân tích, đánh giá về phần mềm ứng dụng (App) để tiến hành bỏ phiếu trên các thiết bị di động

Việc sử dụng các phần mềm ứng dụng (App) để tiến hành bỏ phiếu trên các thiết bị di động hiện nay là một xu hướng đang phát triển, mang lại nhiều lợi ích nhưng cũng đối mặt với không ít thách thức. Dưới đây là một phân tích chi tiết về việc sử dụng các App để bỏ phiếu:

Lợi ích của việc sử dụng App để bỏ phiếu:

- Tiện lợi:

+ Tiết kiệm thời gian: Giảm thiểu thời gian chờ đợi tại các điểm bỏ phiếu truyền thống.

+ Dễ dàng tiếp cận: Cử tri có thể bỏ phiếu từ bất kỳ đâu, chỉ cần có kết nối internet. Điều này đặc biệt hữu ích cho những người sống xa địa điểm bỏ phiếu hoặc có lịch trình bận rộn.

- Tăng cường sự tham gia:

+ Trong thời đại công nghệ thông tin ngày nay thì việc ứng dụng sản phẩm công nghệ vốn đã quen thuộc, nên việc sử dụng App để bỏ phiếu rất thuận tiện.

+ Giảm rào cản cho người khuyết tật: Giúp người khuyết tật tiếp cận và tham gia bỏ phiếu dễ dàng hơn.

- Hiệu quả kinh tế:

Giảm chi phí: Giảm chi phí in ấn, vận chuyển phiếu bầu và thiết lập các điểm bỏ phiếu truyền thống.

- Cải thiện quá trình kiểm phiếu:

Nhanh chóng và chính xác: Quá trình kiểm phiếu được tự động hóa, giảm thiểu sai sót do con người.

Thách thức và rủi ro

- An ninh:

+ Rủi ro bảo mật: Các ứng dụng bỏ phiếu cần phải đối mặt với các nguy cơ về tấn công mạng, gian lận và bảo vệ thông tin cá nhân.

+ Đảm bảo tính toàn vẹn: Cần có các biện pháp để đảm bảo rằng phiếu bầu không bị thay đổi hoặc giả mạo.

- Bảo mật và quyền riêng tư:

+ Tính bí mật: Đảm bảo rằng không ai có thể biết được lựa chọn của người bỏ phiếu.

+ Quản lý dữ liệu cá nhân: Bảo vệ thông tin cá nhân của người bỏ phiếu khỏi việc bị lạm dụng hoặc rò rỉ.

- Khả năng tiếp cận và công bằng:

+ Chênh lệch số hóa: Không phải ai cũng có truy cập internet hoặc thiết bị di động, đặc biệt là người già hoặc người sống ở vùng sâu vùng xa.

+ Hỗ trợ kỹ thuật: Cử tri cần có sự hỗ trợ kỹ thuật nếu gặp khó khăn trong việc sử dụng App.

- Tính hợp pháp và chấp nhận:

Hợp pháp hóa: Cần có khung pháp lý rõ ràng để công nhận và điều chỉnh việc bỏ phiếu qua App.

Niềm tin công chúng: Cử tri và các bên liên quan cần tin tưởng vào tính minh bạch và công bằng của hệ thống bỏ phiếu qua App.

Kết luận

Việc sử dụng App để tiến hành bỏ phiếu mang lại nhiều lợi ích tiềm năng, từ sự tiện lợi đến hiệu quả kinh tế. Tuy nhiên, để triển khai thành công, cần phải giải quyết một loạt các thách thức liên quan đến an ninh, bảo mật, khả năng tiếp cận, và tính hợp pháp. Sự cân nhắc kỹ lưỡng và các biện pháp bảo vệ thích hợp là cần thiết để đảm bảo rằng hệ thống bỏ phiếu qua App hoạt động một cách công bằng, minh bạch và đáng tin cậy.

1.3. Hướng xây dựng phần mềm ứng dụng (App) để bỏ phiếu trên thiết bị di động tại Trường Đại học Hoa Lư

1.3.1. Chức năng của app bỏ phiếu trên thiết bị di động (Mobile)

- Thực hiện các cuộc bỏ phiếu trong các hội nghị, báo cáo, tổng kết... Có 2 hình thức bỏ phiếu: Đồng ý/Không đồng ý và Chọn X ứng cử viên trong Y ứng cử viên ($X \leq Y$).

- Thư ký có thể thực hiện tạo các cuộc bỏ phiếu và nhập dữ liệu danh sách ứng cử viên, kết xuất biên bản kết quả bỏ phiếu (% số phiếu được chọn trên tổng số phiếu) theo đúng mẫu quy định.

- App có thể lưu trữ kết quả lâu dài, khi cần có thể kết xuất dữ liệu

1.3.2. Phạm vi

- Phạm vi tại đơn vị: các cuộc bỏ phiếu được thực hiện trong các hội nghị tại các đơn vị (phòng, khoa, trung tâm, ban).

- Phạm vi cấp trường: các cuộc bỏ phiếu được thực hiện trong các Đại hội, hội nghị do Trường Đại học Hoa Lư tổ chức.

1.3.3. Phân quyền

- Lãnh đạo trường có quyền xem được kết quả tất cả các cuộc bỏ phiếu

- Các cuộc bỏ phiếu được thực hiện tại **các đơn vị (Phòng, Khoa, Trung tâm, Ban)**, thư ký của đơn vị đó chỉ xem và kết xuất biên bản được kết quả bỏ phiếu của đơn vị mình, không xem được kết quả của đơn vị khác.

- Các cuộc bỏ phiếu được thực hiện **toàn trường**, thư ký của cuộc bỏ phiếu đó có thể xem và kết xuất biên bản được kết quả bỏ phiếu.

CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ PHẦN MỀM

2.1. Phân tích hệ thống phần mềm

Phân tích hệ thống cho một phần mềm bỏ phiếu trên ứng dụng (App) bao gồm nhiều bước để đảm bảo hệ thống hoạt động hiệu quả và đáp ứng được nhu cầu của người dùng [8]. Dưới đây là một chi tiết về cách phân tích một hệ thống bỏ phiếu trên ứng dụng:

2.1.1. Xác định yêu cầu

- Yêu cầu chức năng:

- + Đăng ký và đăng nhập người dùng.
- + Tạo và quản lý các cuộc bỏ phiếu.
- + Cho phép người dùng bỏ phiếu một cách bảo mật.
- + Hiện thị kết quả bỏ phiếu theo thời gian thực.
- + Quản lý danh sách ứng cử viên và cử tri.

- Yêu cầu phi chức năng:

- + Bảo mật dữ liệu.
- + Khả năng mở rộng để hỗ trợ số lượng lớn người dùng.
- + Giao diện người dùng thân thiện và dễ sử dụng.
- + Khả năng tích hợp với các hệ thống khác (ví dụ: hệ thống email hoặc SMS để xác thực người dùng).

2.1.2. Thiết kế kiến trúc hệ thống

- Kiến trúc Client-Server:

- + Client: Ứng dụng di động hoặc web mà người dùng sử dụng để tương tác với hệ thống.
- + Server: Xử lý logic kinh doanh, quản lý dữ liệu và xác thực người dùng.

- Cơ sở dữ liệu:

- + Lưu trữ thông tin người dùng, cuộc bỏ phiếu, phiếu bầu và kết quả.
- + Sử dụng các hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) hoặc NoSQL tùy thuộc vào yêu cầu hệ thống.

2.1.3. Thiết kế giao diện người dùng (UI/UX)

- Tạo wireframes và mockups để xác định luồng người dùng.
- Đảm bảo giao diện dễ sử dụng với các hướng dẫn rõ ràng.

2.1.4. Đảm bảo bảo mật

- Sử dụng mã hóa dữ liệu để bảo vệ thông tin người dùng và phiếu bầu.
- Xác thực và phân quyền người dùng để đảm bảo chỉ những người có quyền mới có thể truy cập thông tin nhạy cảm.
- Thực hiện các biện pháp chống lại tấn công như SQL injection, XSS, CSRF...

2.1.5. Kiểm thử hệ thống

Kiểm thử chức năng: Đảm bảo tất cả các chức năng hoạt động đúng như mong đợi.

Kiểm thử phi chức năng: Đánh giá hiệu suất, khả năng mở rộng, và bảo mật của hệ thống.

Kiểm thử người dùng: Thu thập phản hồi từ người dùng thử nghiệm để cải thiện giao diện và trải nghiệm người dùng.

2.1.6. Triển khai và bảo trì

Đưa hệ thống vào hoạt động trên môi trường thực tế.

Cung cấp tài liệu hướng dẫn và hỗ trợ người dùng.

Theo dõi và sửa lỗi, cập nhật hệ thống định kỳ để đảm bảo tính ổn định và an toàn.

2.2. Thiết kế hệ thống theo hướng chức năng

2.2.1. Phân tích chức năng của hệ thống

*** Đăng ký và Đăng nhập:**

- Cho phép người dùng mới tạo tài khoản với thông tin cá nhân như tên, email, số điện thoại.

- Cho phép người dùng đăng nhập bằng thông tin xác thực (tên người dùng và mật khẩu).

- Tích hợp các phương thức xác thực bổ sung như OTP qua email hoặc SMS.

*** Quản lý tài khoản người dùng:**

- Cho phép người dùng cập nhật thông tin cá nhân.

- Cung cấp chức năng quên mật khẩu và đặt lại mật khẩu.

*** Tạo và Quản lý cuộc bỏ phiếu:**

- Cho phép quản trị viên tạo các cuộc bỏ phiếu mới với các chi tiết như tiêu đề, mô tả, thời gian bắt đầu và kết thúc.

- Quản lý danh sách ứng cử viên trong mỗi cuộc bỏ phiếu.
- Cho phép sửa đổi và hủy bỏ các cuộc bỏ phiếu nếu cần thiết.

*** Tham gia bỏ phiếu:**

- Cung cấp giao diện để người dùng xem danh sách các cuộc bỏ phiếu có sẵn.
- Cho phép người dùng tham gia bỏ phiếu với quyền bảo mật.
- Kiểm tra và ngăn chặn việc bỏ phiếu trùng lặp.

*** Xem Kết quả bỏ phiếu:**

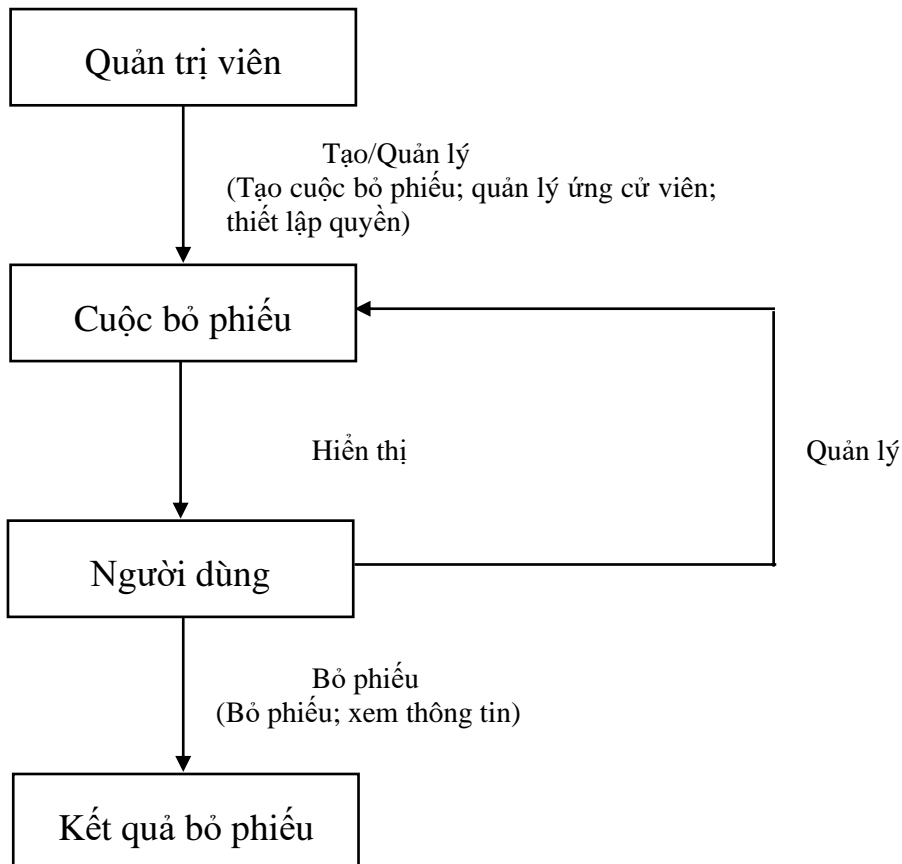
- Hiển thị kết quả bỏ phiếu theo thời gian thực với biểu đồ và số liệu thống kê.
- Cung cấp tính năng lọc và sắp xếp kết quả dựa trên các tiêu chí khác nhau.

*** Thông báo và nhắc nhở:**

- Gửi thông báo đẩy cho người dùng về các cuộc bỏ phiếu sắp tới và kết quả mới nhất.
- Cung cấp nhắc nhở cho người dùng chưa tham gia bỏ phiếu gần đến hạn.

2.2.2. Sơ đồ chức năng

Lưu đồ:



Hình 2.1. Sơ đồ chức năng của hệ thống

Biểu đồ Use Case (Use Case Diagram) là một công cụ hữu ích để mô tả các chức năng chính của một hệ thống và mối quan hệ giữa các tác nhân (actors) và các trường hợp sử dụng (use cases). Dưới đây là một mô tả về biểu đồ Use Case cho phần mềm bỏ phiếu trên App.

*** Các tác nhân (Actors):**

- Người dùng (Voter): Người tham gia bỏ phiếu.
- Quản trị viên (Admin): Người quản lý hệ thống, thiết lập cuộc bỏ phiếu và theo dõi kết quả.
- Hệ thống xác thực (Authentication System): Hệ thống chịu trách nhiệm xác thực đăng nhập và bảo mật.

*** Các Use Case chính:**

- **Người dùng (Voter):** Người tham gia bỏ phiếu.
 - + Đăng ký tài khoản: Người dùng có thể đăng ký tài khoản mới bằng cách nhập thông tin cần thiết.
 - + Đăng nhập: Người dùng đăng nhập vào hệ thống để truy cập các chức năng bỏ phiếu.
 - + Xem danh sách cuộc bỏ phiếu: Người dùng có thể xem danh sách các cuộc bỏ phiếu hiện có.
 - + Tham gia bỏ phiếu: Người dùng có thể chọn một cuộc bỏ phiếu và thực hiện bỏ phiếu.
 - + Xem kết quả bỏ phiếu: Người dùng có thể xem kết quả của cuộc bỏ phiếu (nếu được phép).
 - + Quản lý tài khoản: Người dùng có thể cập nhật thông tin cá nhân và thay đổi mật khẩu.
- **Quản trị viên (Admin):** Người quản lý hệ thống, thiết lập cuộc bỏ phiếu và theo dõi kết quả.
 - + Quản lý người dùng: Quản trị viên có thể thêm, xóa, hoặc chỉnh sửa thông tin người dùng.
 - + Tạo cuộc bỏ phiếu: Quản trị viên có thể tạo mới một cuộc bỏ phiếu, bao gồm đặt tiêu đề, mô tả và các lựa chọn cho người dùng.
 - + Quản lý cuộc bỏ phiếu: Quản trị viên có thể chỉnh sửa hoặc xóa các cuộc bỏ phiếu hiện có. Xem và xuất kết quả: Quản trị viên có thể xem và xuất kết quả bỏ phiếu ra các định dạng khác nhau.

*** Mỗi quan hệ giữa các Use Case và Actors:**

- Người dùng (Voter): liên kết với các Use Case: Đăng ký tài khoản, Đăng nhập, Xem danh sách cuộc bỏ phiếu, Tham gia bỏ phiếu, Xem kết quả bỏ phiếu, Quản lý tài khoản.

- Quản trị viên (Admin): liên kết với các Use Case: Quản lý người dùng, Tạo cuộc bỏ phiếu, Quản lý cuộc bỏ phiếu, Xem và xuất kết quả.

- Hệ thống xác thực (Authentication System): liên kết với các Use Case: Đăng ký tài khoản, Đăng nhập.

*** Mô tả về Biểu đồ Use Case:**

- Actors: Người dùng (Voter); Quản trị viên (Admin); Hệ thống xác thực (Authentication System).

- Use Cases: Đăng ký tài khoản; Đăng nhập; Xem danh sách cuộc bỏ phiếu; Tham gia bỏ phiếu; Xem kết quả bỏ phiếu; Quản lý tài khoản; Quản lý người dùng; Tạo cuộc bỏ phiếu; Quản lý cuộc bỏ phiếu; Xem và xuất kết quả.

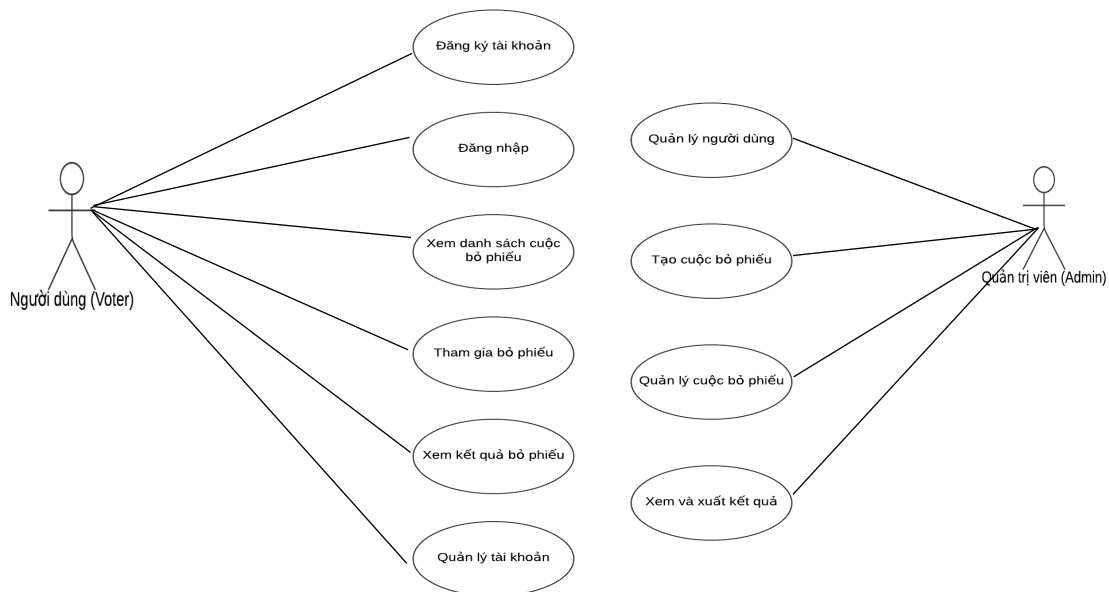
- Môi quan hệ:

+ Người dùng ↔ Đăng ký tài khoản, Đăng nhập, Xem danh sách cuộc bỏ phiếu, Tham gia bỏ phiếu, Xem kết quả bỏ phiếu, Quản lý tài khoản

+ Quản trị viên ↔ Quản lý người dùng, Tạo cuộc bỏ phiếu, Quản lý cuộc bỏ phiếu, Xem và xuất kết quả

+ Hệ thống xác thực ↔ Đăng ký tài khoản, Đăng nhập

*** Biểu đồ:**



Hình 2.2. Biểu đồ Use Case của hệ thống

2.2.3. Mô tả chi tiết các chức năng

*** Chức năng Đăng ký (Sign Up):**

Mục đích: Người dùng mới tạo tài khoản để sử dụng phần mềm bỏ phiếu.

- Nhập thông tin cá nhân:

+ Họ và tên

+ Email/ Số điện thoại

+ Mật khẩu

+ Xác nhận mật khẩu

- Xác thực thông tin: Xác thực Email/ Số điện thoại (bằng cách gửi mã xác thực OTP hoặc liên kết xác nhận)

- Tạo tài khoản:

+ Lưu trữ thông tin người dùng vào cơ sở dữ liệu

+ Tạo hồ sơ người dùng (Profile)

- Hoàn tất đăng ký: người dùng được chuyển hướng đến màn hình đăng nhập hoặc trực tiếp đăng nhập vào ứng dụng sau khi đăng ký thành công.

*** Chức năng Đăng nhập (Login):**

Mục đích: Người dùng đăng nhập vào tài khoản đã tạo để sử dụng các chức năng của phần mềm bỏ phiếu.

- Nhập thông tin đăng nhập:

+ Email/ Số điện thoại

+ Mật khẩu

+ Xác thực thông tin đăng nhập:

+ Kiểm tra thông tin người dùng trong cơ sở dữ liệu

+ Xác nhận mật khẩu đúng

- Đăng nhập thành công:

+ Chuyển hướng người dùng đến trang chính của ứng dụng (Dashboard)

+ Quên mật khẩu (nếu có): tùy chọn để người dùng có thể yêu cầu khôi phục mật khẩu bằng cách gửi mã OTP hoặc liên kết đến email/ số điện thoại.

*** Chức năng xem danh sách cuộc bỏ phiếu:**

- Bắt đầu (Start): Đại diện cho việc người dùng mở ứng dụng và chọn chức năng "Xem danh sách cuộc bỏ phiếu".

- Kiểm tra trạng thái đăng nhập:

(Decision) Đã đăng nhập? Có (Yes): Chuyển đến bước tiếp theo; Không (No): Yêu cầu đăng nhập (Return to login screen).

- Truy vấn danh sách cuộc bỏ phiếu: (Process) Truy vấn dữ liệu từ cơ sở dữ liệu.

- Hiển thị danh sách cuộc bỏ phiếu: (Display) Hiển thị danh sách các cuộc bỏ phiếu với các tùy chọn lọc và sắp xếp.

- Tương tác với từng cuộc bỏ phiếu: (Decision) Người dùng nhấp vào một cuộc bỏ phiếu?

+ Có (Yes): Hiển thị chi tiết cuộc bỏ phiếu.

+ Không (No): Chờ tương tác tiếp theo hoặc thoát.

- Kết thúc (End): Người dùng có thể quay lại giao diện chính hoặc thoát khỏi phần xem danh sách.

*** Chức năng tham gia bỏ phiếu:**

- Bắt đầu (Start): Đại diện cho việc người dùng chọn một cuộc bỏ phiếu để tham gia.

- Kiểm tra trạng thái đăng nhập: (Decision) Đã đăng nhập?

+ Có (Yes): Chuyển đến bước tiếp theo.

+ Không (No): Yêu cầu đăng nhập.

- Kiểm tra tính hợp lệ của cuộc bỏ phiếu: (Decision) Cuộc bỏ phiếu đang diễn ra?

+ Có (Yes): Chuyển đến bước tiếp theo.

+ Không (No): Hiển thị thông báo và dừng quy trình.

- Kiểm tra trạng thái bỏ phiếu của người dùng: (Decision) Người dùng đã bỏ phiếu chưa?

+ Chưa (No): Chuyển đến bước tiếp theo.

+ Rồi (Yes): Hiển thị thông báo đã bỏ phiếu.

- Hiển thị giao diện bỏ phiếu: (Process) Hiển thị các tùy chọn bỏ phiếu.

- Xác nhận lựa chọn: (Decision) Người dùng xác nhận lựa chọn?

+ Có (Yes): Chuyển đến bước tiếp theo.

- + Không (No): Quay lại giao diện bỏ phiếu.
- Lưu kết quả bỏ phiếu: (Process) Lưu kết quả bỏ phiếu vào cơ sở dữ liệu.
- Hiện thị thông báo hoàn tất: (Display) Hiện thị thông báo bỏ phiếu thành công.
- Kết thúc (End): Người dùng có thể quay lại giao diện chính hoặc thoát ứng dụng.

*** Chức năng xem kết quả bỏ phiếu:**

- Bắt đầu (Start): Người dùng chọn một cuộc bỏ phiếu từ danh sách để xem kết quả.
- Kiểm tra trạng thái đăng nhập: (Decision) Đã đăng nhập?
- + Có (Yes): Chuyển đến bước tiếp theo.
- + Không (No): Yêu cầu đăng nhập.
- Kiểm tra tính hợp lệ của cuộc bỏ phiếu: (Decision) Cuộc bỏ phiếu đã kết thúc?
- + Có (Yes): Chuyển đến bước tiếp theo.
- + Không (No): Hiện thị thông báo và dừng quy trình.
- Kiểm tra quyền truy cập kết quả: (Decision) Người dùng có quyền xem kết quả?
- + Có (Yes): Chuyển đến bước tiếp theo.
- + Không (No): Hiện thị thông báo và dừng quy trình.
- Truy vấn và hiện thị kết quả: (Process) Truy vấn kết quả từ cơ sở dữ liệu. (Display) Hiện thị kết quả (số lượng phiếu, tỷ lệ phần trăm, biểu đồ).
- Tùy chọn tải xuống hoặc chia sẻ kết quả: (Optional) Người dùng có thể tải xuống hoặc chia sẻ kết quả.
- Kết thúc (End): Người dùng có thể quay lại giao diện chính hoặc thoát ứng dụng.

*** Chức năng quản lý tài khoản:**

- Bắt đầu (Start): Người dùng chọn mục "Quản lý tài khoản".
- Kiểm tra trạng thái đăng nhập: (Decision) Đã đăng nhập?
- + Có (Yes): Chuyển đến bước tiếp theo.

- + Không (No): Yêu cầu đăng nhập.
- Hiển thị giao diện quản lý tài khoản: (Display) Hiển thị các tùy chọn: Cập nhật thông tin cá nhân, Thay đổi mật khẩu, Xóa tài khoản.
- Cập nhật thông tin cá nhân (Nếu chọn):
 - + (Process) Nhập thông tin mới.
 - + (Optional) Xác thực bằng mật khẩu hoặc OTP.
 - + (Process) Lưu thông tin mới vào cơ sở dữ liệu.
 - + (Display) Thông báo cập nhật thành công.
- Thay đổi mật khẩu (Nếu chọn):
 - + (Process) Nhập mật khẩu hiện tại và mật khẩu mới.
 - + (Optional) Yêu cầu OTP để xác nhận.
 - + (Process) Cập nhật mật khẩu mới.
 - + (Display) Thông báo thay đổi mật khẩu thành công.
- Xóa tài khoản (Nếu chọn):
 - + (Process) Yêu cầu xác nhận bằng mật khẩu hoặc OTP.
 - + (Process) Xóa tài khoản khỏi cơ sở dữ liệu.
 - + (Display) Thông báo tài khoản đã được xóa.
- Kết thúc (End): Người dùng quay lại màn hình chính hoặc thoát ứng dụng.

*** Chức năng quản lý người dùng:**

Cung cấp cho quản trị viên hệ thống khả năng quản lý người dùng của ứng dụng bỏ phiếu, bao gồm việc tạo, chỉnh sửa, xóa và quản lý quyền truy cập của người dùng.

- Bắt đầu (Start): Quản trị viên chọn mục "Quản lý người dùng".
- Xác thực quyền quản trị: (Decision) Người dùng có quyền quản trị không?
 - + Có (Yes): Chuyển đến bước tiếp theo.
 - + Không (No): Hiển thị thông báo lỗi và dừng quy trình.
- Hiển thị danh sách người dùng: (Process) Truy vấn và hiển thị danh sách người dùng.

- Tùy chọn quản lý người dùng: (Decision) Chọn một trong các hành động sau: Tạo, Chỉnh sửa, Xóa, Thay đổi quyền.

- Tạo người dùng mới (Nếu chọn):

+ (Process) Nhập thông tin người dùng mới và lưu vào cơ sở dữ liệu.

+ (Display) Thông báo tạo người dùng thành công.

- Chỉnh sửa thông tin người dùng (Nếu chọn):

+ (Process) Cập nhật thông tin người dùng và lưu vào cơ sở dữ liệu.

+ (Display) Thông báo cập nhật thành công.

- Xóa người dùng (Nếu chọn):

+ (Process) Xác nhận xóa và thực hiện thao tác xóa trong cơ sở dữ liệu.

+ (Display) Thông báo xóa thành công.

- Thay đổi quyền của người dùng (Nếu chọn):

+ (Process) Cập nhật quyền truy cập của người dùng và lưu vào cơ sở dữ liệu.

+ (Display) Thông báo cập nhật thành công.

- Kết thúc (End): Quản trị viên quay lại giao diện chính hoặc thoát ứng dụng.

*** Chức năng tạo cuộc bỏ phiếu:**

Cho phép người dùng (thường là quản trị viên hoặc người có quyền) tạo một cuộc bỏ phiếu mới trên ứng dụng, bao gồm việc nhập thông tin chi tiết, thiết lập thời gian và lựa chọn tùy chọn bỏ phiếu.

- Bắt đầu (Start): Người dùng chọn mục "Tạo cuộc bỏ phiếu".

- Xác thực quyền tạo cuộc bỏ phiếu: (Decision) Người dùng có quyền tạo cuộc bỏ phiếu không?

+ Có (Yes): Chuyển đến bước tiếp theo.

+ Không (No): Hiện thị thông báo lỗi và dừng quy trình.

- Nhập thông tin cuộc bỏ phiếu: (Process) Nhập tiêu đề, mô tả, ngày bắt đầu/kết thúc, các tùy chọn bỏ phiếu, và các cài đặt khác.

- Cấu hình cài đặt nâng cao (Nếu có): (Process) Thiết lập chế độ ẩn danh, giới hạn số lần bỏ phiếu, cài đặt quyền truy cập.

- Xác nhận và lưu cuộc bỏ phiếu: (Decision) Thông tin đã hợp lệ chưa?
- + Có (Yes): Lưu vào cơ sở dữ liệu và hiển thị thông báo thành công.
- + Không (No): Hiển thị thông báo lỗi yêu cầu sửa đổi.
- Quản lý cuộc bỏ phiếu (Nếu cần): (Optional) Chỉnh sửa cuộc bỏ phiếu, Xóa cuộc bỏ phiếu trước khi nó bắt đầu.
- Kết thúc (End): Người dùng có thể quay lại giao diện chính hoặc tiếp tục quản lý các cuộc bỏ phiếu khác.

Chức năng quản lý cuộc bỏ phiếu:

Cho phép quản trị viên hoặc người có quyền quản lý các cuộc bỏ phiếu trên ứng dụng, bao gồm việc chỉnh sửa, xóa, theo dõi và kết thúc các cuộc bỏ phiếu.

- Bắt đầu (Start): Quản trị viên chọn mục "Quản lý cuộc bỏ phiếu".
- Xác thực quyền quản trị: (Decision) Người dùng có quyền quản trị không?
- + Có (Yes): Chuyển đến bước tiếp theo.
- + Không (No): Hiển thị thông báo lỗi và dừng quy trình.
- Hiển thị danh sách cuộc bỏ phiếu: (Process) Truy vấn và hiển thị danh sách cuộc bỏ phiếu.
- Tùy chọn quản lý cuộc bỏ phiếu:
Decision) Chọn một trong các hành động sau: Chỉnh sửa, Xóa, Theo dõi tiến trình, Kết thúc cuộc bỏ phiếu.

- Chỉnh sửa thông tin cuộc bỏ phiếu (Nếu chọn):
- + (Process) Chỉnh sửa thông tin và lưu vào cơ sở dữ liệu.
- + (Display) Thông báo cập nhật thành công.
- Xóa cuộc bỏ phiếu (Nếu chọn):
- + (Process) Xác nhận xóa và thực hiện thao tác xóa trong cơ sở dữ liệu.
- + (Display) Thông báo xóa thành công.
- Theo dõi tiến trình cuộc bỏ phiếu (Nếu chọn): (Process) Hiển thị số lượng người tham gia, tỷ lệ lựa chọn, và số phiếu đã bỏ.
- Kết thúc cuộc bỏ phiếu thủ công (Nếu chọn):

- + (Process) Xác nhận kết thúc và hệ thống thực hiện kết thúc cuộc bỏ phiếu.
- + (Display) Thông báo kết thúc thành công.
- Kết thúc (End): Quản trị viên quay lại giao diện chính hoặc tiếp tục quản lý các cuộc bỏ phiếu khác.

*** Chức năng xem và xuất kết quả**

Cho phép người dùng xem kết quả của cuộc bỏ phiếu và xuất kết quả này ra định dạng file (như PDF, CSV) để lưu trữ hoặc chia sẻ.

- Bắt đầu (Start): Người dùng chọn mục "Xem kết quả" từ giao diện chính.
- Chọn cuộc bỏ phiếu cần xem kết quả:
 - + (Process) Hiển thị danh sách các cuộc bỏ phiếu.
 - + (Decision) Người dùng chọn cuộc bỏ phiếu cần xem kết quả.
 - Kiểm tra quyền truy cập kết quả: (Decision) Người dùng có quyền xem kết quả không?
 - + Có (Yes): Chuyển đến bước tiếp theo.
 - + Không (No): Hiển thị thông báo lỗi và dừng quy trình.
 - Hiển thị kết quả cuộc bỏ phiếu: (Process) Truy vấn và hiển thị kết quả cuộc bỏ phiếu.
 - Chọn tùy chọn xuất kết quả: (Decision) Người dùng chọn định dạng file để xuất kết quả (PDF, CSV, Excel, v.v.).
 - Xác nhận và thực hiện xuất kết quả:
 - + (Process) Hệ thống yêu cầu xác nhận, tạo file và cho phép tải xuống.
 - + (Display) Thông báo xuất kết quả thành công.
 - Kết thúc (End): Người dùng quay lại giao diện chính hoặc tiếp tục xem và xuất kết quả của các cuộc bỏ phiếu khác.

2.3. Thiết kế Cơ sở dữ liệu

Thiết kế cơ sở dữ liệu (Database Design) của phần mềm bỏ phiếu chạy trên thiết bị di động (App) là một phần quan trọng để đảm bảo rằng hệ thống có thể quản lý dữ liệu hiệu quả và hỗ trợ các chức năng như đăng ký, bỏ phiếu, quản lý

cuộc bỏ phiếu và xem kết quả. Dưới đây là một mô hình cơ sở dữ liệu đề xuất cho phần mềm bỏ phiếu này:

2.3.1. Các Bảng (table) của hệ thống

a) Bảng `Users` (Người dùng)

Chứa thông tin về tất cả người dùng trong hệ thống, bao gồm quản trị viên và người tham gia bình thường.

- **user_id** (PK): Khóa chính, định danh duy nhất cho mỗi người dùng.
- **username**: Tên đăng nhập của người dùng.
- **password_hash**: Mật khẩu của người dùng được mã hóa.
- **fullname_user**: Họ và tên người dùng
- **email**: Email của người dùng.
- **role**: Vai trò của người dùng (admin, user).
- **created_at**: Thời gian tạo tài khoản.
- **updated_at**: Thời gian cập nhật lần cuối.

b) Bảng `Polls` (Cuộc bỏ phiếu)

Chứa thông tin về các cuộc bỏ phiếu mà người dùng có thể tham gia.

- **poll_id** (PK): Khóa chính, định danh duy nhất cho mỗi cuộc bỏ phiếu.
- **title**: Tiêu đề của cuộc bỏ phiếu.
- **description**: Mô tả chi tiết về cuộc bỏ phiếu.
- **user_id** (FK): Khóa ngoại tham chiếu đến **user_id** trong bảng **Users** (người tạo).
- **start_date**: Ngày bắt đầu của cuộc bỏ phiếu.
- **end_date**: Ngày kết thúc của cuộc bỏ phiếu.
- **status**: Trạng thái của cuộc bỏ phiếu (chưa bắt đầu, đang diễn ra, đã kết thúc).
- **poll_type**: Loại cuộc bỏ phiếu (yes_no hoặc choose_x_from_y).
- **max_selections**: Số lượng lựa chọn tối đa (chỉ áp dụng cho cuộc bỏ phiếu choose_x_from_y).
- **is_public**: Quản lý các cuộc bỏ phiếu công khai và riêng tư.
- **visibility**: Quy định ai có thể xem cuộc bỏ phiếu (ví dụ: public, private, nhóm người dùng cụ thể).
- **created_at**: Thời gian tạo cuộc bỏ phiếu.
- **updated_at**: Thời gian cập nhật lần cuối.

c) Bảng `Candidate` (Ứng cử viên)

- **candidate_id** (PK): Khóa chính, định danh duy nhất cho mỗi kết quả.
- **fullname_user**: Họ và tên ứng viên.
- **birthday**: Ngày sinh ứng viên.
- **picture**: Lưu trữ đường dẫn URL của ảnh ứng viên
- **description**: Mô tả về ứng viên

- work_unit: đơn vị công tác.
- updated_at: Thời gian cập nhật lần cuối.

d) Bảng `Votes` (Bỏ phiếu)

Chứa thông tin về các phiếu bầu của người dùng.

- **vote_id** (PK): Khóa chính, định danh duy nhất cho mỗi phiếu bầu.
- **user_id** (FK): Khóa ngoại tham chiếu đến `user_id` trong bảng `Users`.
- **option_id** (FK): Khóa ngoại tham chiếu đến `option_id` trong bảng `Options`.
- **voted_at**: Thời gian người dùng bỏ phiếu.
- status: int

e) Bảng `polls_candidate`

- poll_candidate_id (PK): Khóa chính, định danh duy nhất cho mỗi kết quả.
- candidate_id (FK): int
- poll_id (FK): int
- **created_at**: Thời gian tạo lựa chọn.
- **updated_at**: Thời gian cập nhật lần cuối.

f) Bảng `Options` (Các lựa chọn)

Chứa thông tin về các lựa chọn trong mỗi cuộc bỏ phiếu.

- **option_id** (PK): Khóa chính, định danh duy nhất cho mỗi lựa chọn.
- **option_description**: varchar (mô tả về vsự lựa chọn)
- **option_decision**: int
- **poll_candidate_id** (FK): int
- **created_at**: Thời gian tạo lựa chọn.
- **updated_at**: Thời gian cập nhật lần cuối.

g) Bảng `User_Permissions` (Quyền của người dùng)

Chứa thông tin về các quyền của từng người dùng (dành cho hệ thống có phân quyền phức tạp hơn).

- **user_id** (FK): Khóa ngoại tham chiếu đến `user_id` trong bảng `Users`.
- **permission**: Tên quyền (ví dụ: tạo cuộc bỏ phiếu, xem kết quả).
- **granted_by**: Theo dõi người đã cấp quyền cho người dùng.
- **created_at**: Thời gian cấp quyền.
- **updated_at**: Thời gian cập nhật lần cuối.

2.3.2. Mô tả mối quan hệ giữa các Bảng

Dưới đây là mô tả chi tiết về quan hệ giữa các bảng trong cơ sở dữ liệu voting_hluv:

a) Bảng Users:

Users là bảng lưu thông tin tất cả người dùng trong hệ thống, bao gồm cả quản trị viên và người dùng thông thường. Bảng này kết nối với một số bảng khác thông qua khóa ngoại user_id.

b) Bảng Polls:

Bảng Polls lưu thông tin các cuộc bỏ phiếu.

Quan hệ: Có một-nhiều với Users thông qua user_id, nghĩa là mỗi cuộc bỏ phiếu do một người dùng (admin hoặc user) tạo ra, nhưng một người dùng có thể tạo ra nhiều cuộc bỏ phiếu.

c) Bảng Candidate:

Bảng Candidate chứa thông tin về các ứng viên trong cuộc bỏ phiếu.

Quan hệ: Nhiều-nhiều với Polls thông qua bảng trung gian polls_candidate.

d) Bảng Votes:

Bảng Votes lưu thông tin về từng phiếu bầu mà người dùng thực hiện trong các cuộc bỏ phiếu.

Quan hệ:

- Một-nhiều với Users thông qua user_id, nghĩa là mỗi người dùng có thể bỏ nhiều phiếu.

- Một-nhiều với Options thông qua option_id, cho biết mỗi phiếu sẽ có một lựa chọn từ bảng Options.

e) Bảng Polls_candidate:

Đây là bảng trung gian, thể hiện quan hệ Nhiều-nhiều giữa bảng Polls và Candidate. Một cuộc bỏ phiếu có thể có nhiều ứng cử viên và một ứng cử viên có thể tham gia nhiều cuộc bỏ phiếu khác nhau.

Quan hệ:

- Một-nhiều với Polls thông qua poll_id.

- Một-nhiều với Candidate thông qua candidate_id.

f) Bảng Options:

Bảng Options lưu các lựa chọn mà người dùng có thể chọn trong mỗi cuộc bỏ phiếu.

Quan hệ:

- Một-nhiều với polls_candidate qua poll_candidate_id, tức là mỗi poll_candidate_id có thể có nhiều lựa chọn khác nhau trong cuộc bỏ phiếu.

- Một-nhiều với Votes qua option_id, cho phép lưu lại các phiếu bầu của người dùng dựa trên các lựa chọn này.

g) Bảng User_Permissions:

Bảng User_Permissions lưu các quyền của người dùng, cho phép phân quyền nâng cao trong hệ thống.

Quan hệ: Một-nhiều với Users qua user_id, mỗi người dùng có thể được cấp nhiều quyền khác nhau.

2.3.3. Mô hình ERD (Entity-Relationship Diagram)

Mô hình CSDL dưới dạng mô hình ERD với các bảng và mối quan hệ như sau:

a) Quan hệ giữa các bảng:

- Users - Polls:

Quan hệ 1-n: Một người dùng (Users) có thể tạo nhiều cuộc bỏ phiếu (Polls). Khóa ngoại user_id trong bảng Polls tham chiếu đến user_id trong bảng Users.

- Users - Votes:

Quan hệ 1-n: Một người dùng có thể bỏ nhiều phiếu (Votes). Khóa ngoại user_id trong bảng Votes tham chiếu đến user_id trong bảng Users.

- Users - User_Permissions:

Quan hệ 1-n: Một người dùng có thể được gán nhiều quyền (User_Permissions). Khóa ngoại user_id trong bảng User_Permissions tham chiếu đến user_id trong bảng Users.

- Polls - polls_candidate:

Quan hệ 1-n: Một cuộc bỏ phiếu có thể có nhiều ứng viên thông qua bảng trung gian polls_candidate. Khóa ngoại poll_id trong bảng polls_candidate tham chiếu đến poll_id trong bảng Polls.

- Candidate - polls_candidate:

Quan hệ 1-n: Một ứng viên có thể tham gia nhiều cuộc bỏ phiếu thông qua bảng polls_candidate. Khóa ngoại candidate_id trong bảng polls_candidate tham chiếu đến candidate_id trong bảng Candidate.

- Polls_candidate - Options:

Quan hệ 1-n: Mỗi kết hợp giữa cuộc bỏ phiếu và ứng viên trong bảng polls_candidate có thể có nhiều lựa chọn (Options) tương ứng. Khóa ngoại poll_candidate_id trong bảng Options tham chiếu đến poll_candidate_id trong bảng polls_candidate.

- Options - Votes:

Quan hệ 1-n: Mỗi lựa chọn trong một cuộc bỏ phiếu có thể nhận nhiều phiếu (Votes). Khóa ngoại option_id trong bảng Votes tham chiếu đến option_id trong bảng Options.

b) Mô hình ERD có thể được giản lược như sau:

Users --- (1-n) --- Polls

Users --- (1-n) --- Votes

Users --- (1-n) --- User_Permissions

Polls --- (1-n) --- Polls_candidate --- (n-1) --- Candidate

polls_candidate --- (1-n) --- Options

Options --- (1-n) --- Votes

2.4. Các công cụ xây dựng hệ thống phần mềm

2.4.1. Hệ quản trị cơ sở dữ liệu MySQL [7],[10]

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS - Relational Database Management System) mã nguồn mở phổ biến nhất hiện nay. Được phát triển bởi công ty MySQL AB vào năm 1995, MySQL đã trở thành một trong những công cụ cơ sở dữ liệu được sử dụng rộng rãi trong các ứng dụng web, nhờ tính hiệu quả, dễ sử dụng, và hiệu suất cao.

Các đặc điểm chính của MySQL:

- Mã nguồn mở: MySQL là phần mềm mã nguồn mở, cho phép người dùng tự do sử dụng, chỉnh sửa và phân phối lại dưới các giấy phép GPL (GNU General Public License).

- Hiệu suất cao: MySQL được tối ưu hóa để xử lý các tác vụ đọc và ghi với tốc độ cao, phù hợp với các ứng dụng yêu cầu truy vấn dữ liệu lớn và cần tính toán phức tạp.

- Tính mở rộng: MySQL hỗ trợ các bảng dữ liệu lớn, có thể chứa hàng triệu bản ghi và cung cấp các công cụ mạnh mẽ để quản lý và tối ưu hóa cơ sở dữ liệu.

- Hỗ trợ nhiều nền tảng: MySQL có thể chạy trên nhiều hệ điều hành khác nhau, bao gồm Windows, Linux, macOS và Unix. Điều này làm cho MySQL trở thành một lựa chọn linh hoạt cho các nhà phát triển.

- Ngôn ngữ truy vấn: MySQL sử dụng SQL (Structured Query Language) làm ngôn ngữ truy vấn chính, một chuẩn công nghiệp phổ biến trong quản lý cơ sở dữ liệu.

- Bảo mật cao: MySQL cung cấp nhiều lớp bảo mật, bao gồm xác thực người dùng, quyền hạn truy cập chi tiết, và mã hóa dữ liệu, giúp bảo vệ dữ liệu trước các mối đe dọa an ninh.

- Hỗ trợ cho các tính năng nâng cao: MySQL cung cấp các tính năng như lưu trữ giao dịch (transactional storage), chỉ mục toàn văn (full-text indexing) và replication, hỗ trợ khả năng dự phòng và mở rộng.

- Ứng dụng của MySQL: MySQL thường được sử dụng trong các ứng dụng web như trang web thương mại điện tử, hệ thống quản lý nội dung (CMS) như WordPress, Joomla, Drupal, và các nền tảng quản lý cơ sở dữ liệu cho các dịch vụ lớn như Facebook, Twitter, và YouTube.

2.4.2. Ngôn ngữ lập trình API (Application Programming Interface) - Laravel [5]

API (Application Programming Interface) chính là phương thức trung gian kết nối các ứng dụng và thư viện khác nhau. Nó cung cấp khả năng truy xuất đến một tập các hàm hay dùng, từ đó có thể trao đổi dữ liệu giữa các ứng dụng.

Laravel là một framework PHP mã nguồn mở, được thiết kế để phát triển các ứng dụng web theo kiến trúc MVC (Model-View-Controller). Laravel nổi tiếng với cú pháp đẹp, dễ đọc và cung cấp nhiều tính năng mạnh mẽ giúp tăng năng suất cho các nhà phát triển.

Sau đây là một số đặc điểm nổi bật của Laravel:

- Kiến trúc MVC: Model-View-Controller (MVC): Laravel tuân theo mô hình MVC, giúp tách biệt logic ứng dụng (Model) khỏi giao diện người dùng (View) và luồng điều khiển (Controller). Điều này làm cho mã nguồn dễ dàng bảo trì và mở rộng.

- Routing: Laravel cung cấp hệ thống định tuyến mạnh mẽ, cho phép định nghĩa các route cho ứng dụng một cách dễ dàng và linh hoạt.

- Middleware: Laravel hỗ trợ middleware, cho phép xử lý các yêu cầu HTTP trước khi chúng đến controller, giúp quản lý các tác vụ như xác thực, logging và kiểm soát truy cập.

- Blade Template Engine: Laravel đi kèm với Blade, một công cụ template mạnh mẽ và linh hoạt, cho phép tái sử dụng mã HTML và tích hợp logic PHP một cách dễ dàng.

- Eloquent ORM: Hệ thống ORM của Laravel, Eloquent, cung cấp cách tiếp cận đơn giản và mạnh mẽ để tương tác với cơ sở dữ liệu. Nó cho phép sử dụng

các class PHP để thao tác với các bảng trong cơ sở dữ liệu, hỗ trợ các mối quan hệ phức tạp giữa các bảng một cách dễ dàng.

- Artisan CLI: Laravel cung cấp một công cụ dòng lệnh tên là Artisan, cho phép thực hiện nhiều tác vụ phát triển như tạo code boilerplate, di chuyển cơ sở dữ liệu và chạy các script.

- Tích hợp hệ thống kiểm thử: Laravel tích hợp các công cụ kiểm thử giúp dễ dàng viết và chạy các bài kiểm tra đơn vị (unit tests) và kiểm thử tính năng (feature tests) để đảm bảo ứng dụng hoạt động chính xác.

- Queueing: Laravel hỗ trợ hệ thống queue, giúp quản lý các tác vụ nền (background tasks) một cách hiệu quả.

- Events và Broadcasting: Laravel có cơ chế xử lý sự kiện và broadcasting, cho phép xây dựng các ứng dụng thời gian thực một cách dễ dàng.

- API: Laravel cung cấp các công cụ và thư viện hỗ trợ việc xây dựng API RESTful một cách nhanh chóng và dễ dàng.

- Laravel có một cộng đồng lớn và hệ sinh thái phong phú với nhiều gói mở rộng, giúp giảm bớt công sức phát triển và tăng tính năng của ứng dụng.

2.4.3. Ngôn ngữ lập trình Flutter [4]

Flutter không phải là một ngôn ngữ lập trình mà là một framework phát triển ứng dụng mã nguồn mở được phát triển bởi Google. Flutter cho phép xây dựng các ứng dụng di động, web và desktop từ một mã nguồn duy nhất.

Những điểm nổi bật của Flutter:

- Dart: Flutter sử dụng ngôn ngữ lập trình Dart, cũng được phát triển bởi Google. Dart là một ngôn ngữ hướng đối tượng, với cú pháp tương tự như JavaScript và Java. Nó được thiết kế để giúp phát triển ứng dụng dễ dàng và hiệu quả hơn.

- Giao diện người dùng tùy chỉnh:

- + Widget-based Architecture: Flutter sử dụng hệ thống widget để xây dựng giao diện người dùng. Mọi thứ trong Flutter đều là widget, từ các thành phần giao diện cơ bản như nút bấm và văn bản đến các thành phần phức tạp hơn như danh sách và lưới.

- + Hot Reload: Flutter hỗ trợ tính năng hot reload, cho phép xem ngay lập tức các thay đổi trong mã nguồn mà không cần phải khởi động lại ứng dụng, giúp tăng tốc quá trình phát triển.

- Hiệu suất cao:

Native Compilation: Flutter biên dịch mã nguồn Dart thành mã máy gốc, cho phép ứng dụng chạy trực tiếp trên các nền tảng mà không cần qua các lớp chuyển đổi. Điều này giúp cải thiện hiệu suất và tốc độ của ứng dụng.

- Phong cách thiết kế:

Material Design và Cupertino: Flutter hỗ trợ hai phong cách thiết kế chính: Material Design (dành cho các ứng dụng Android) và Cupertino (dành cho các ứng dụng iOS). Điều này cho phép tạo ra các ứng dụng có giao diện đồng nhất với các hệ điều hành mà chúng chạy trên đó.

- Kết nối với các API và dịch vụ:

Plugin và Packages: Flutter có một hệ sinh thái phong phú với nhiều plugin và packages, cho phép dễ dàng tích hợp với các API và dịch vụ bên ngoài như cơ sở dữ liệu, dịch vụ web, và các dịch vụ đám mây.

Hỗ trợ đa nền tảng:

Mobile, Web và Desktop: Với Flutter có thể phát triển ứng dụng cho cả nền tảng di động (iOS và Android), web, và desktop (Windows, macOS, và Linux) từ cùng một cơ sở mã nguồn.

- Cộng đồng và tài liệu: Flutter có một cộng đồng người dùng và nhà phát triển lớn và năng động, cùng với tài liệu phong phú và nhiều tài nguyên học tập, giúp dễ dàng tìm kiếm sự hỗ trợ và học hỏi.

CHƯƠNG 3. XÂY DỰNG PHẦN MỀM

3.1. Xây dựng hệ thống Cơ sở dữ liệu

3.1.1. Tạo các bảng cơ sở dữ liệu trên hệ quản trị cơ sở dữ liệu MySQL [7], [9],[10]

Tên cơ sở dữ liệu của hệ thống: voting_hluv

a) Tạo các Bảng cơ sở dữ liệu:

(1) Bảng `Users` (Người dùng), tạo bằng câu lệnh như sau:

```
CREATE TABLE Users (  
    user_id INT AUTO_INCREMENT PRIMARY KEY,  
    username VARCHAR(50) NOT NULL UNIQUE,  
    password_hash VARCHAR(255) NOT NULL,  
    fullname_user VARCHAR(100),  
    email VARCHAR(100) UNIQUE,  
    role ENUM('admin', 'user') NOT NULL,  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
ON UPDATE CURRENT_TIMESTAMP );
```

(2) Bảng `User_Permissions` (Quyền của người dùng)

```
CREATE TABLE User_Permissions (  
    user_id INT,  
    permission VARCHAR(100),  
    granted_by INT,  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
ON UPDATE CURRENT_TIMESTAMP,  
    PRIMARY KEY (user_id, permission),  
    FOREIGN KEY (user_id) REFERENCES Users(user_id) ON  
DELETE CASCADE,  
    FOREIGN KEY (granted_by) REFERENCES Users(user_id) ON  
DELETE SET NULL );
```

(3) Bảng `Polls` (Cuộc bỏ phiếu)

```
CREATE TABLE Polls (  
    poll_id INT AUTO_INCREMENT PRIMARY KEY,  
    title VARCHAR(255) NOT NULL,  
    description TEXT,  
    user_id INT,  
    start_date DATE,  
    end_date DATE,  
    status ENUM('chưa bắt đầu', 'đang diễn ra', 'đã kết thúc') NOT  
NULL,  
    poll_type ENUM('yes_no', 'choose_x_from_y') NOT NULL,  
    max_selections INT DEFAULT 1,  
    is_public BOOLEAN DEFAULT 1,  
    visibility ENUM('public', 'private', 'specific_group') DEFAULT  
'public',  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
ON UPDATE CURRENT_TIMESTAMP,  
    FOREIGN KEY (user_id) REFERENCES Users(user_id) ON  
DELETE CASCADE );
```

(4) Bảng `Candidate` (Ứng cử viên)

```
CREATE TABLE Candidate (  
    candidate_id INT AUTO_INCREMENT PRIMARY KEY,  
    fullname_user VARCHAR(100) NOT NULL,  
    birthday DATE,  
    picture VARCHAR(255),  
    description TEXT,  
    work_unit VARCHAR(100),  
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
ON UPDATE CURRENT_TIMESTAMP) ;
```

(5) Bảng 'Polls_candidate'

```
CREATE TABLE polls_candidate (  
    poll_candidate_id INT AUTO_INCREMENT PRIMARY KEY,  
    candidate_id INT,  
    poll_id INT,  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
ON UPDATE CURRENT_TIMESTAMP,  
    FOREIGN KEY (candidate_id) REFERENCES  
Candidate(candidate_id) ON DELETE CASCADE,  
    FOREIGN KEY (poll_id) REFERENCES Polls(poll_id) ON  
DELETE CASCADE);
```

(6) Bảng 'Options' (Các lựa chọn)

```
CREATE TABLE Options (  
    option_id INT AUTO_INCREMENT PRIMARY KEY,  
    option_description VARCHAR(255),  
    option_decision INT,  
    poll_candidate_id INT,  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
ON UPDATE CURRENT_TIMESTAMP,  
    FOREIGN KEY (poll_candidate_id) REFERENCES  
polls_candidate(poll_candidate_id) ON DELETE CASCADE);
```

(7) Bảng 'Votes' (Bỏ phiếu)

```
CREATE TABLE Votes (  
    vote_id INT AUTO_INCREMENT PRIMARY KEY,  
    user_id INT,  
    option_id INT,  
    voted_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    status INT DEFAULT 1,
```

```
FOREIGN KEY (user_id) REFERENCES Users(user_id) ON
DELETE CASCADE,
```

```
FOREIGN KEY (option_id) REFERENCES Options(option_id) ON
DELETE CASCADE);
```

b) Tạo các trigger để tự động hóa công việc

(1) Cập nhật số lượng phiếu bầu cho từng ứng cử viên

```
BEGIN
  UPDATE poll_candidates
  SET vote_count = vote_count + 1
  WHERE id = NEW.id;
END
```

(2) Cập nhật bảng option sau khi có thông tin được insert trong công poll_candidate

```
BEGIN
  INSERT INTO options (option_id, option_description,
option_decision, poll_candidate_id, created_at, updated_at)
  VALUES (
    NEW.id,
    CONCAT('Lựa chọn cho cuộc bầu cử ', NEW.poll_id, ' và ứng cử
viên ', NEW.candidate_id),
    '1',
    NEW.id,
    CURRENT_TIMESTAMP(),
    CURRENT_TIMESTAMP()
  );
END
```

(3). Kiểm tra trạng thái cuộc bầu cử trước khi bỏ phiếu

```
BEGIN
  DECLARE poll_status VARCHAR(255);

  -- Assuming the Votes table has a column named poll_id
  SELECT status INTO poll_status
  FROM Polls
  WHERE id = NEW.id;
  IF poll_status != 'đang diễn ra' THEN
    SIGNAL SQLSTATE '45000'
    SET MESSAGE_TEXT = 'Cuộc bỏ phiếu này đã kết thúc hoặc chưa
bắt đầu.';
  END IF;
END
```

c). Tạo các bảng ảo để hiển thị dữ liệu

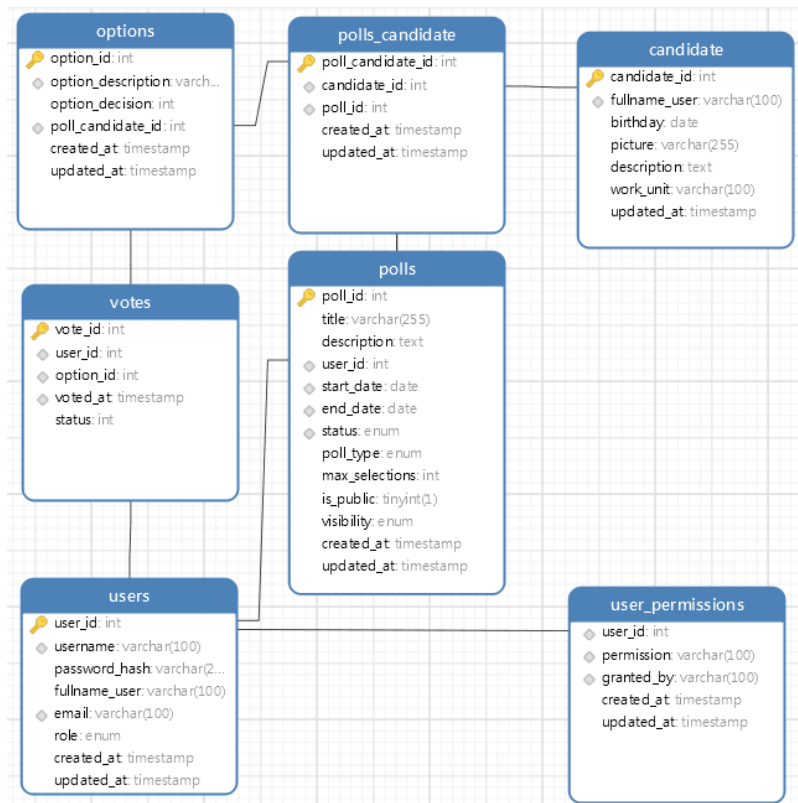
(1) View hiển thị thông tin về một candidate

```
BEGIN
SELECT
  c.id,
  c.fullname_user,
  c.birthday,
  c.picture,
  c.description,
  c.work_unit,
  p.title AS poll_title,
  p.start_date AS poll_start_date,
  p.end_date AS poll_end_date,
  COUNT(v.id) AS vote_count
FROM
  candidates c
  LEFT JOIN poll_candidates pc ON c.id = pc.candidate_id
  LEFT JOIN polls p ON pc.poll_id = p.id
  LEFT JOIN options o ON pc.id = o.poll_candidate_id
  LEFT JOIN votes v ON o.option_id = v.option_id
WHERE
  pc.poll_id = poll_id
GROUP BY
  c.id, p.id;
END
```

(2) Bảng hiển thị kết quả

```
select `p`.`id` AS `poll_id`, `p`.`title` AS `poll_title`,`c`.`id` AS
`candidate_id`,`c`.`fullname_user` AS `candidate_name`, count(`v`.`id`)
AS `vote_count`,`p`.`start_date` AS `poll_start_date`,`p`.`end_date` AS
`poll_end_date`
from (((`votedb`.`polls` `p` join
`votedb`.`poll_candidates` `pc` on(`p`.`id` = `pc`.`poll_id`)) join
`votedb`.`candidates` `c` on(`pc`.`candidate_id` = `c`.`id`)) left join
`votedb`.`options` `o` on(`pc`.`id` = `o`.`poll_candidate_id`)) left join
`votedb`.`votes` `v` on(`o`.`option_id` = `v`.`option_id`)) group by
`p`.`id`,`p`.`title`,`c`.`id`,`c`.`fullname_user`,`p`.`start_date`,`p`.`end_da
te` order by `p`.`id`,count(`v`.`id`) desc
```

3.1.2. Mô hình ERD (Entity-Relationship Diagram) với bảng dữ liệu ở trên và mối quan hệ giữa chúng [6]



Hình 3. 1. Mô hình cơ sở dữ liệu “voting_hluv” dưới dạng ERD

3.2. Xây dựng các ứng dụng bằng Laravel [5]

Bước 1: Thiết lập môi trường Laravel gồm:

(1) Cài đặt Laravel (composer create-project laravel/laravel vote_app)

(2) Cấu hình cơ sở dữ liệu qua lệnh như sau:

```
php artisan make:model User -m
```

```
php artisan make:model Poll -m
```

```
php artisan make:model Candidate -m
```

```
php artisan make:model Vote -m
```

```
php artisan make:model PollCandidate -m
```

```
php artisan make:model Option -m
```

```
php artisan make:model UserPermission -m;
```

(3) Tạo các controller cho API backend:

```
php artisan make:controller UserController
```

```
php artisan make:controller PollController
```



```
php artisan make:controller VoteController
php artisan make:controller OptionController
php artisan make:controller AuthController
php artisan make:controller CandidateController
php artisan make:controller ElectionResultsController
php artisan make:controller HomeController
php artisan make:controller PollCandidateController
php artisan make:controller VoterController
```

Bước 2: Xác định các route API trong file routes/web.php gồm:

(1) Định nghĩa route cho UserController:

```
use App\Http\Controllers\UserController;
Route::prefix('users')->group(function () {
    Route::get('/', [UserController::class, 'index']); // Lấy danh sách tất
    cả người dùng
    Route::post('/', [UserController::class, 'store']); // Tạo người dùng mới
    Route::get('/{id}', [UserController::class, 'show']); // Xem chi tiết
    một người dùng
    Route::put('/{id}', [UserController::class, 'update']); // Cập nhật
    thông tin người dùng
    Route::delete('/{id}', [UserController::class, 'destroy']); // Xóa người dùng
});
```

(2) Định nghĩa route cho PollController:

```
use App\Http\Controllers\PollController;
Route::prefix('polls')->group(function () {
    Route::get('/', [PollController::class, 'index']); // Lấy danh sách các
    cuộc bỏ phiếu
    Route::post('/', [PollController::class, 'store']); // Tạo cuộc bỏ phiếu mới
    Route::get('/{id}', [PollController::class, 'show']); // Xem chi tiết
    cuộc bỏ phiếu
    Route::put('/{id}', [PollController::class, 'update']); // Cập nhật cuộc
    bỏ phiếu
```

```
Route::delete('/{id}', [PollController::class, 'destroy']); // Xóa cuộc  
bỏ phiếu
```

```
});
```

(3) Định nghĩa route cho VoteController:

```
use App\Http\Controllers\VoteController;
```

```
Route::prefix('votes')->group(function () {
```

```
Route::get('/', [VoteController::class, 'index']); // Lấy danh sách các  
phiếu bầu
```

```
Route::post('/', [VoteController::class, 'store']); // Tạo phiếu bầu mới
```

```
Route::get('/{id}', [VoteController::class, 'show']); // Xem chi tiết phiếu bầu
```

```
});
```

(4) Định nghĩa route cho OptionController:

```
use App\Http\Controllers\OptionController;
```

```
Route::prefix('options')->group(function () {
```

```
Route::get('/', [OptionController::class, 'index']); // Lấy danh sách  
các lựa chọn
```

```
Route::post('/', [OptionController::class, 'store']); // Tạo lựa chọn mới
```

```
Route::get('/{id}', [OptionController::class, 'show']); // Xem chi tiết  
lựa chọn
```

```
Route::put('/{id}', [OptionController::class, 'update']); // Cập nhật  
lựa chọn
```

```
Route::delete('/{id}', [OptionController::class, 'destroy']); // Xóa lựa chọn  
});
```

(5) Định nghĩa router cho PollCandidateController:

```
Route::resource('pollcandidate', 'PollCandidateController');
```

(6) Định nghĩa router cho ElectionResultsController:

```
Route::get('election-results', [ElectionResultsController::class,  
'index']->name('election_results.index'));
```

(7) Định nghĩa route cho file routes/api.php:

```
Route::post('register', [AuthController::class, 'register']);
```

```
Route::post('login', [AuthController::class, 'login']);
```

```
Route::post('logout', [AuthController::class, 'logout'])->  
middleware('auth:sanctum');
```

```
Route::get('/polls', [PollController::class, 'index_api']);
```

```
Route::post('/polls/{poll}/vote', [PollController::class, 'vote']);
```

Bước 3: Xây dựng API trong Laravel với các phương thức CRUD (Create, Read, Update, Delete) cho từng controller và xử lý các chức năng như đăng ký, đăng nhập, quản lý cuộc bỏ phiếu, ứng viên, và phân quyền người dùng. Thực hiện như sau:

(1) Xử lý Đăng ký, Đăng nhập, Phân quyền qua UserController

Tạo các phương thức trong UserController:

register(): Đăng ký tài khoản mới.

login(): Đăng nhập và tạo token xác thực.

show(): Lấy thông tin người dùng.

update(): Cập nhật thông tin người dùng.

delete(): Xóa tài khoản.

assignRole(): Phân quyền người dùng (admin, user).

*** code cho UserController:**

```
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use App\Models\User;
use Illuminate\Support\Facades\Hash;
use Illuminate\Support\Facades\Auth;
class UserController extends Controller
{
    public function register(Request $request)
    {
        $validatedData = $request->validate([
            'username' => 'required|unique:users',
            'password' => 'required|min:6',
            'email' => 'required|email|unique:users',
            'fullname_user' => 'required'
        ]);
        $user = User::create([
            'username' => $validatedData['username'],
            'password_hash' => Hash::make($validatedData['password']),
            'fullname_user' => $validatedData['fullname_user'],
            'email' => $validatedData['email'],
            'role' => 'user'
        ]);
        return response()->json(['message' => 'User registered successfully'], 201);
    }
    public function login(Request $request)
```

```

{
    $credentials = $request->only('username', 'password');
    if (Auth::attempt($credentials)) {
        $user = Auth::user();
        $token = $user->createToken('authToken')->plainTextToken;
        return response()->json(['token' => $token], 200);
    }
    return response()->json(['message' => 'Invalid credentials'], 401);
}
// Các phương thức khác (show, update, delete, assignRole) sẽ được
// định nghĩa tương tự
}

```

(2) Tạo, Cập nhật, Xóa, Lấy Thông tin Cuộc bỏ phiếu qua PollController

Thêm các phương thức như index(), store(), show(), update(), destroy() cho PollController để quản lý các cuộc bỏ phiếu.

*** code cho PollController:**

```

namespace App\Http\Controllers;
use Illuminate\Http\Request;
use App\Models\Candidate;
class CandidateController extends Controller
{
    public function store(Request $request)
    {
        $validatedData = $request->validate([
            'fullname_user' => 'required',
            'birthday' => 'required|date',
            'description' => 'nullable',
            'work_unit' => 'nullable'
        ]);
        $candidate = Candidate::create($validatedData);
        return response()->json($candidate, 201);
    }
    // Các phương thức index(), show(), update(), destroy() sẽ được định
    // nghĩa tương tự.
}

```

(3) Thêm, Sửa, Xóa Ứng viên qua CandidateController

Thêm các phương thức index(), store(), show(), update(), destroy() cho CandidateController.

*** code cho CandidateController**

```
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use App\Models\Candidate;
class CandidateController extends Controller
{
    public function store(Request $request)
    {
        $validatedData = $request->validate([
            'fullname_user' => 'required',
            'birthday' => 'required|date',
            'description' => 'nullable',
            'work_unit' => 'nullable'
        ]);
        $candidate = Candidate::create($validatedData);
        return response()->json($candidate, 201);
    }
    // Các phương thức index(), show(), update(), destroy() sẽ được định nghĩa tương tự
}
```

(4) Thêm, Sửa, Xóa Lựa chọn ứng viên qua OptionController

Thêm các phương thức index(), store(), show(), update(), destroy() cho OptionController.

*** code cho OptionController.**

```
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use App\Models\Option;
class OptionController extends Controller
{
    public function store(Request $request)
    {
        $validatedData = $request->validate([
            'option_description' => 'required',
            'option_decision' => 'required|integer',
            'poll_candidate_id' => 'required|exists:polls_candidate,poll_candidate_id'
        ]);
        $option = Option::create($validatedData);
        return response()->json($option, 201);
    }
}
```

```
// Các phương thức index(), show(), update(), destroy() sẽ được định nghĩa tương tự
}
```

(5) Xử lý quyền người dùng qua UserPermissionsController

Thêm các phương thức index(), store(), show(), update(), destroy() cho UserPermissionsController để quản lý quyền người dùng.

** code cho UserPermissionsController.*

```
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use App\Models\UserPermission;
class UserPermissionsController extends Controller
{
    public function store(Request $request)
    {
        $validatedData = $request->validate([
            'user_id' => 'required|exists:users,user_id',
            'permission' => 'required|string',
            'granted_by' => 'required|exists:users,user_id'
        ]);
        $userPermission = UserPermission::create($validatedData);
        return response()->json($userPermission, 201);
    }
    // Các phương thức khác (index(), show(), update(), destroy()) sẽ được định nghĩa tương tự
}
```

3.3. Xây dựng giao diện người dùng bằng Flutter [4]

Bước 1: Thiết lập môi trường:

(1) Cài đặt Flutter (flutter create vote_app)

Trong thư mục dự án, cấu trúc dự án sẽ bao gồm lib/ nơi có các thành phần chính.

(2) Thiết lập các gói cần thiết

Flutter có một số gói phổ biến để làm việc với HTTP và quản lý trạng thái:

http: Gói này dùng để gửi yêu cầu đến API Laravel.

provider hoặc bloc: Dùng để quản lý trạng thái ứng dụng.

shared_preferences: Lưu trữ thông tin như token đăng nhập của người dùng.

(3) Tạo cấu trúc thư mục

Trong thư mục lib/, tạo các thư mục con để quản lý mã nguồn:

models/: Chứa các lớp mô hình (User, Poll, Candidate, Vote, Option).

services/: Chứa các lớp để gọi API (UserService, PollService).

screens/: Chứa các màn hình UI chính như đăng nhập, trang chủ, quản lý bỏ phiếu.

providers/: Quản lý trạng thái của ứng dụng.

Bước 2: Thiết lập API Calls trong Flutter

Tạo một tệp tin trong thư mục services/ như auth_service.dart:

```
import 'dart:convert';
import 'package:http/http.dart' as http;
class AuthService {
  final String apiUrl = 'http://192.168.10.228000/api';
  Future<bool> login(String username, String password) async {
    final response = await http.post(
      Uri.parse('$apiUrl/login'),
      body: {'username': username, 'password': password},
    );
    if (response.statusCode == 200) {
      // Lưu token vào SharedPreferences hoặc xử lý đăng nhập thành công
      return true;
    } else {
      // Xử lý lỗi
      return false;
    }
  }
}
```

Bước 3: Xây dựng các màn hình UI cơ bản

LoginScreen: Tạo màn hình đăng nhập, sử dụng AuthService để gửi yêu cầu đăng nhập.

```
@override
Widget build(BuildContext context) {
  return Scaffold(
    body: Stack(
      children: [
        // Background image
        Container(
          decoration: BoxDecoration(
            image: DecorationImage(
              image: AssetImage('assets/logo.png'), // Your main logo image
              fit: BoxFit.cover,
            ),
          ),
      ],
    ),
  ),
);
```

```

// Login form overlay
Center(
  child: Padding(
    padding: const EdgeInsets.all(20.0),
    child: Card(
      elevation: 8,
      shape: RoundedRectangleBorder(
        borderRadius: BorderRadius.circular(15),
      ),
      child: Padding(
        padding: const EdgeInsets.all(20.0),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.min,
          children: [
            // Title
            Text(
              'Hệ thống bỏ phiếu Online',
              style: TextStyle(
                fontSize: 24,
                fontWeight: FontWeight.bold,
                color: Colors.blueAccent,
              ),
            ),
            SizedBox(height: 10),
            // Subtitle
            Text(
              'Trường Đại học Hoa Lu',
              style: TextStyle(
                fontSize: 18,
                color: Colors.blueGrey,
              ),
            ),
            SizedBox(height: 20),
            // University logo
            Image.asset(
              'assets/logo_truong.png', // Your university logo
              height: 100, // Adjust height as needed
            ),
            SizedBox(height: 20),
            // Password input
            TextField(
              controller: _passwordController,
              obscureText: true,

```



```

        decoration: InputDecoration(
          labelText: 'Password',
          border: OutlineInputBorder(
            borderRadius: BorderRadius.circular(10),
          ),
          prefixIcon: Icon(Icons.lock),
        ),
      ),
      SizedBox(height: 20),
      // Login button
      ElevatedButton(
        onPressed: _login,
        style: ElevatedButton.styleFrom(
          padding: EdgeInsets.symmetric(
            horizontal: 30, vertical: 15),
          shape: RoundedRectangleBorder(
            borderRadius: BorderRadius.circular(10),
          ),
        ),
        child: Text('Login', style: TextStyle(fontSize: 18)),
      ),
    ],
  ),
),
),
),
),
],
),
);
}

```

PollListScreen: Màn hình hiển thị các cuộc bỏ phiếu. Trong file này có hai phương thức để kết nối và lấy cơ sở dữ liệu từ hệ thống:

```

Future<void> fetchPolls() async {
  print('Fetching polls...');
  final response =
    await
http.get(Uri.parse('http://192.168.10.22:8000/api/polls'));

  if (response.statusCode == 200) {
    print('Response body: ${response.body}'); // Log the response
body
    setState() {
      polls = jsonDecode(response.body);

```

```

        print('Polls: $polls'); // Log the polls list
    });
} else {
    print(
        'Failed to load polls: ${response.statusCode}'); // Log the error status
    throw Exception('Failed to load polls');
}
}
Future<List<dynamic>> fetchCandidates(int pollId) async {
    final response = await http.get(
        Uri.parse('http://192.168.10.22:8000/api/polls/$pollId/candidates'));

    if (response.statusCode == 200) {
        return json.decode(response.body);
    } else {
        throw Exception('Failed to load candidates');
    }
}
}

```

PollItem: Màn hình chi tiết cuộc bỏ phiếu cho phép người dùng tham gia bỏ phiếu.

```

Widget build(BuildContext context) {
    return GestureDetector(
        onTap: () => onTap(),
        child: Container(
            padding: EdgeInsets.all(16),
            margin: EdgeInsets.symmetric(vertical: 8, horizontal: 16),
            decoration: BoxDecoration(
                color: Colors.white,
                borderRadius: BorderRadius.circular(8),
                boxShadow: [
                    BoxShadow(color: Colors.grey, blurRadius: 4, offset: Offset(0, 2)),
                ],
            ),
        child: Row(
            children: [
                Icon(Icons.ballot, color: Colors.blue),
                SizedBox(width: 10),
                Expanded(child: Text(title, style: TextStyle(fontSize: 18))),
            ],
        ),
    );
}

```

CandidatesScreen: Màn hình danh sách các ứng cử viên trong mỗi cuộc bỏ phiếu cho phép người dùng lựa chọn.

```
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Container(
        alignment: Alignment.center, // Center the title
        child: Text(
          pollName, // Poll name
          style: TextStyle(
            fontSize: 20, // Adjust font size as needed
            fontWeight: FontWeight.bold, // Optional: make it bold
          ),
        ),
      ),
    centerTitle: true, // Center the title in the AppBar
  ),
  body: ListView.builder(
    itemCount: candidates.length,
    itemBuilder: (context, index) {
      return CandidateItem(
        id: candidates[index]['id'],
        name: candidates[index]['fullname_user'] ?? 'No Name
Available',
        description: candidates[index]['description'] ?? 'No
Description',
        imageUrl: candidates[index]['imageUrl'] ?? "",
        backgroundColor: index.isEven
          ? Colors.white
          : Colors.grey[200]!, // Optional image URL
      );
    },
  ),
);
}
```

CandidateItem. Màn hình hiển thị thông tin chi tiết của từng thí sinh để người dùng có thể chọn lựa.

```
@override
Widget build(BuildContext context) {
  return Card(
    color: backgroundColor,
    margin: const EdgeInsets.symmetric(vertical: 8, horizontal: 16),
    elevation: 4,
```

```

child: ListTile(
  leading: imageUrl.isNotEmpty
    ? CircleAvatar(
      backgroundImage: NetworkImage(imageUrl),
    )
    : CircleAvatar(child: Icon(Icons.person)),
  title: Text(
    name,
    style: TextStyle(fontWeight: FontWeight.bold, fontSize: 18),
  ),
  subtitle: Text(
    description,
    style: TextStyle(color: Colors.grey[600]),
  ),
  trailing: Icon(Icons.arrow_forward),
  onTap: () async {
    try {
      await vote(1, id as int); // Replace 1 with the actual user ID
      Navigator.pop(context);
      ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(content: Text('Vote recorded successfully')),
      );
    } catch (e) {
      Navigator.pop(context); // This will return to the previous screen
      ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(content: Text('Failed to record vote')),
      );
    }
  },
),
);
}

```

Bước 4: Quản lý trạng thái

Sử dụng provider để quản lý trạng thái đăng nhập của người dùng và cập nhật UI khi có thay đổi từ API.

```

import 'package:flutter/material.dart';
class AuthProvider with ChangeNotifier {
  bool _isLoggedIn = false;
  bool get isLoggedIn => _isLoggedIn;
  void login() {
    _isLoggedIn = true;
    notifyListeners();
  }
}

```

```
    }  
    void logout() {  
        _isLoggedIn = false;  
        notifyListeners();  
    }  
}
```

Bước 5: Kết nối với API

Sử dụng AuthProvider và AuthService để điều khiển quá trình đăng nhập, đăng xuất và các chức năng khác trong ứng dụng.

Bước 6: Chạy ứng dụng và kiểm tra

Chạy flutter run để kiểm tra ứng dụng trên thiết bị mô phỏng hoặc thiết bị thật.

3.4. Cài đặt thử nghiệm

3.4.1. Mục tiêu thử nghiệm

Mục tiêu của phần thử nghiệm là đảm bảo tất cả các tính năng của hệ thống bỏ phiếu hoạt động chính xác theo yêu cầu đã định. Các thành phần cần thử nghiệm gồm:

Tính năng đăng ký, đăng nhập, phân quyền người dùng.

Quản lý cuộc bỏ phiếu (tạo, cập nhật, xóa cuộc bỏ phiếu).

Xem và quản lý ứng viên.

Tham gia bỏ phiếu.

Quản lý quyền người dùng và bảo mật.

3.4.2. Quy trình cài đặt thử nghiệm

Bước 1: Chuẩn bị môi trường thử nghiệm

Thiết lập Backend:

Chạy hệ thống API Laravel trên localhost hoặc một máy chủ thử nghiệm.

Kiểm tra kết nối với MySQL và đảm bảo dữ liệu mẫu được thiết lập.

Thiết lập Frontend:

Dùng lệnh flutter run để chạy ứng dụng Flutter trên thiết bị mô phỏng hoặc thiết bị thật.

Kiểm tra các gói như http, provider và shared_preferences đã được cài đặt đầy đủ trong pubspec.yaml.

Cấu hình dữ liệu thử nghiệm:

Tạo tài khoản mẫu với các vai trò khác nhau (admin, user).

Tạo các cuộc bỏ phiếu mẫu với các trạng thái khác nhau (đang diễn ra, đã kết thúc).

Tạo các ứng viên mẫu và các lựa chọn để kiểm tra tính năng chọn lựa.

Bước 2: Thực hiện kiểm thử các chức năng chính

Kiểm thử đăng nhập và đăng ký:

Kiểm tra đăng ký với các đầu vào hợp lệ và không hợp lệ.

Đăng nhập bằng tài khoản admin và user, xác nhận mỗi tài khoản có quyền tương ứng.

Kiểm thử chức năng quản lý cuộc bỏ phiếu:

Kiểm tra chức năng tạo cuộc bỏ phiếu (chọn các tùy chọn, thêm ứng viên, chọn trạng thái).

Kiểm tra cập nhật và xóa cuộc bỏ phiếu với các trạng thái khác nhau.

Xác minh xem các cuộc bỏ phiếu công khai và riêng tư hiển thị đúng theo quyền người dùng.

Kiểm thử ứng viên và lựa chọn:

Kiểm tra thêm, sửa, và xóa ứng viên trong cuộc bỏ phiếu.

Đảm bảo người dùng có thể chọn đúng số lượng lựa chọn tối đa cho phép trong cuộc bỏ phiếu.

Kiểm thử tính năng bỏ phiếu:

Thực hiện bỏ phiếu và kiểm tra cập nhật trạng thái.

Kiểm tra để đảm bảo người dùng không thể bỏ phiếu hai lần cho cùng một cuộc bỏ phiếu.

Kiểm thử phân quyền người dùng:

Đảm bảo người dùng chỉ có quyền truy cập tính năng được phân công (ví dụ: chỉ admin mới có thể tạo và quản lý cuộc bỏ phiếu).

Kiểm tra xử lý khi người dùng không có quyền truy cập vào một tính năng cụ thể.

Bước 3: Kiểm thử giao diện người dùng (UI)

Kiểm tra bố cục và vị trí các thành phần UI trên các màn hình chính (đăng nhập, trang chủ, cuộc bỏ phiếu).

Đảm bảo tất cả các nút và liên kết hoạt động chính xác.

Xác minh rằng các thông báo lỗi hiển thị đúng khi người dùng nhập dữ liệu không hợp lệ.

Bước 4: Kiểm thử bảo mật

Đảm bảo rằng token người dùng được lưu trữ an toàn và được xác thực đúng khi gửi yêu cầu API.

Thử nghiệm xem người dùng có thể truy cập tính năng vượt quyền hay không.

Kiểm tra đăng xuất và xác minh rằng sau khi đăng xuất, người dùng không thể tiếp tục truy cập vào các trang bảo mật.

Bước 5: Ghi nhận và báo cáo kết quả kiểm thử

Ghi lại tất cả các lỗi hoặc vấn đề phát hiện trong quá trình thử nghiệm.

Đối với mỗi lỗi, ghi chú rõ: mô tả lỗi, bước tái hiện lỗi, kết quả mong đợi, và kết quả thực tế.

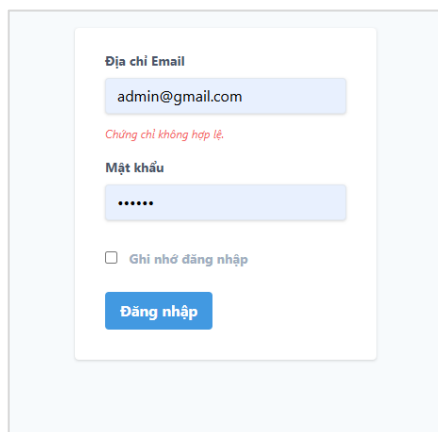
Sau khi khắc phục các lỗi, thử nghiệm lại các tính năng để xác nhận rằng lỗi đã được sửa.

Bước 6. Sau khi chạy thử trên máy localhost, chương trình được cài đặt chạy trên mạng LAN.

3.4.3 Giao diện phần mềm

a) Chức năng dành cho người quản trị

* Đăng nhập hệ thống

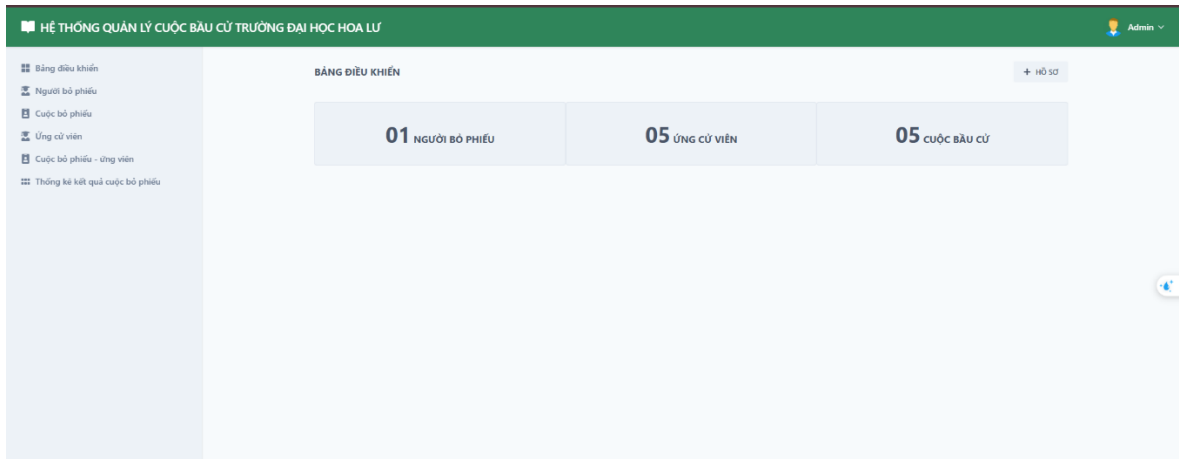


The image shows a login form with the following elements:

- Label: "Địa chỉ Email"
- Input field: "admin@gmail.com"
- Error message: "Chữ không hợp lệ." (in red)
- Label: "Mật khẩu"
- Input field: "....."
- Checkbox: "Ghi nhớ đăng nhập"
- Button: "Đăng nhập"

Hình 3. 2. Form đăng nhập hệ thống

* Quản lý thông tin hệ thống



Hình 3. 3. Bảng điều khiển

* Quản lý thông tin cuộc bầu cử (thêm, sửa, xóa, hiển thị danh sách cuộc bầu cử)

DANH SÁCH CUỘC BẦU CỬ

TÊN	MÔ TẢ	TRẠNG THÁI	NGƯỜI TẠO	NGÀY BẮT ĐẦU	NGÀY KẾT THÚC	SỐ LƯỢNG BẦU TỐI ĐA	CHẾ ĐỘ	HÀNH ĐỘNG
Đại hội chi bộ	Đại hội chi bộ khoa Ngoại ngữ và Công nghệ thông tin nhiệm kỳ 2023 – 2027	chưa bắt đầu	Trưởng khoa CNTT	2024-09-25	2024-09-25	4	group	✎ ✕ 🔍
Đại hội công đoàn bộ phận	Đại hội tổ công đoàn khoa Ngoại ngữ và Công nghệ thông tin nhiệm kỳ 2024 – 2026	đang diễn ra	Trưởng khoa CNTT	2024-09-26	2024-09-26	4	private	✎ ✕ 🔍
Hội nghị tổng kết năm học	Hội nghị kết kết năm học 2024 - 2025 của khoa Ngoại ngữ và Công nghệ thông tin	đã kết thúc	Trưởng khoa CNTT	2024-09-27	2024-09-27	4	group	✎ ✕ 🔍

Hình 3. 4. Quản lý danh sách cuộc bỏ phiếu

* Quản lý thông tin ứng cử viên

DANH SÁCH ỨNG CỬ VIÊN

TÊN	NGÀY SINH	MÔ TẢ	ĐƠN VỊ CÔNG TÁC	HÀNH ĐỘNG
Đào Sỹ Nhiên	09-08-1979	Bi thư chi Bộ - Trưởng khoa	Khoa Ngoại ngữ - CNTT	✎ ✕ 🔍
Nguyễn Thị Hồng Tuyền	09-11-1978	Chi ủy viên - Phó trưởng khoa	Khoa Ngoại ngữ - CNTT	✎ ✕ 🔍
Đông Thị Thu	22-04-1979	Phó trưởng khoa	Khoa Ngoại ngữ - CNTT	✎ ✕ 🔍
Nguyễn Thị Hoàng Huế	27-09-1982	Chi ủy viên - Giảng viên	Khoa Ngoại ngữ - CNTT	✎ ✕ 🔍
Phạm Thị Thanh	21-11-2024	Phó Giám đốc Trung Tâm Ngoại ngữ, Tin học	Trung Tâm Ngoại ngữ Tin học	✎ ✕ 🔍

Hình 3. 5. Quản lý danh sách ứng cử viên

* Tạo cuộc bầu cử mới

The screenshot shows a web form for creating a new poll. The form includes the following fields and options:

- Tên cuộc bầu cử: [Text input field]
- Mô tả: [Text input field]
- Người tạo: --Chọn người-- [Dropdown menu]
- Ngày khởi đầu: [Text input field]
- Ngày kết thúc: [Text input field]
- Trạng thái: chưa bắt đầu đang diễn ra đã kết thúc
- Loại bầu cử: yes_no choose_x_from_y
- Số lượng bầu tối đa: [Text input field]
- Chế độ hiển thị: public private group
- Button: Create poll

Hình 3. 6. Tạo cuộc bầu cử mới

* Tạo thông tin ứng cử viên (thêm mới, sửa và xóa ứng cử viên)

The screenshot shows the 'THÔNG TIN CHI TIẾT ỨNG CỬ VIÊN' form. It includes the following fields and a photo:

- Tên ứng cử viên: Đào Sỹ Nhiên
- Ngày sinh: 1979-08-04
- Ảnh: [Portrait photo of a man]
- Mã số: B Khoa (1) Bộ - Trường Khoa
- Đơn vị ứng cử: Khoa Ngoại ngữ - CNTT
- Chức danh bầu: [Text input field]
- Button: Đăng tin

Hình 3. 7. Xem thông tin ứng cử viên

The screenshot shows a table of candidates with a red dialog box overlaid on top. The dialog box asks: "Are you sure you want to delete candidate?" with a "DELETE" button. A red arrow points to the delete icon in the 'HÀNH ĐỘNG' column of the first row.

TÊN	ĐƠN VỊ CÔNG TÁC	HÀNH ĐỘNG
Đào Sỹ Nhiên	Khoa Ngoại ngữ - CNTT	[Edit] [Delete] [Add]
Nguyễn Thị Hồng Tuyền	Khoa Ngoại ngữ - CNTT	[Edit] [Delete] [Add]
Đông Thị Thu	Khoa Ngoại ngữ - CNTT	[Edit] [Delete] [Add]
Nguyễn Thị Hoàng Huệ	Khoa Ngoại ngữ - CNTT	[Edit] [Delete] [Add]
Phạm Thị Thanh	Trung Tâm Ngoại ngữ Tin học	[Edit] [Delete] [Add]

Hình 3. 8. Xóa thông tin ứng cử viên

* Hiện thị thông tin ứng cử viên trong mỗi cuộc bỏ phiếu

DANH SÁCH ỨNG CỬ VIÊN + THÊM CUỘC BỎ PHIẾU - ỨNG VIÊN

CUỘC BỎ PHIẾU	ỨNG VIÊN	LẦN CHỈNH SỬA	HÀNH ĐỘNG
Đại hội chi bộ	Đào Sỹ Nhiên	2024-09-20 10:47:30	↻ ✎ 🗑️
Đại hội chi bộ	Nguyễn Thị Hồng Tuyền	2024-09-20 10:47:30	↻ ✎ 🗑️
Đại hội chi bộ	Đồng Thị Thu	2024-09-20 10:47:30	↻ ✎ 🗑️
Đại hội chi bộ	Nguyễn Thị Hoàng Huế	2024-09-20 10:47:30	↻ ✎ 🗑️
Đại hội chi bộ	Phạm Thị Thanh	2024-11-08 08:43:38	↻ ✎ 🗑️
Đại hội công đoàn bộ phận	Đào Sỹ Nhiên	2024-09-20 10:47:30	↻ ✎ 🗑️
Đại hội công đoàn bộ phận	Nguyễn Thị Hồng Tuyền	2024-09-20 10:47:30	↻ ✎ 🗑️
Đại hội công đoàn bộ phận	Đồng Thị Thu	2024-09-20 10:47:30	↻ ✎ 🗑️
Đại hội công đoàn bộ phận	Nguyễn Thị Hoàng Huế	2024-09-20 10:47:30	↻ ✎ 🗑️
Hội nghị tổng kết năm học	Đào Sỹ Nhiên	2024-09-20 10:47:30	↻ ✎ 🗑️

< 1 2 >

Hình 3. 9. Quản lý ứng cử viên và cuộc bỏ phiếu tương ứng

* Kết quả các cuộc bầu cử

Chọn cuộc thăm dò: Đại hội chi bộ [Xem kết quả](#)

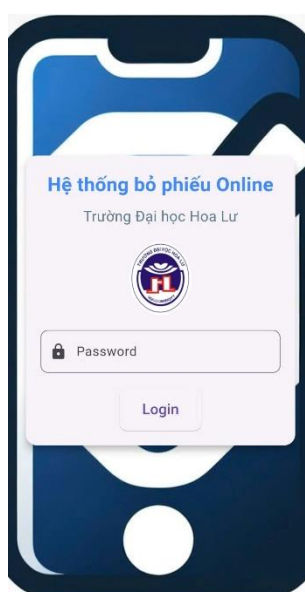
Kết quả bầu cử

Tên cuộc thăm dò	Tên ứng cử viên	Số phiếu bầu	Ngày bắt đầu	Ngày kết thúc
Đại hội chi bộ	Nguyễn Thị Hồng Tuyền	7	25-09-2024	25-09-2024
Đại hội chi bộ	Đồng Thị Thu	6	25-09-2024	25-09-2024
Đại hội chi bộ	Nguyễn Thị Hoàng Huế	3	25-09-2024	25-09-2024
Đại hội chi bộ	Đào Sỹ Nhiên	2	25-09-2024	25-09-2024
Đại hội chi bộ	Phạm Thị Thanh	0	25-09-2024	25-09-2024

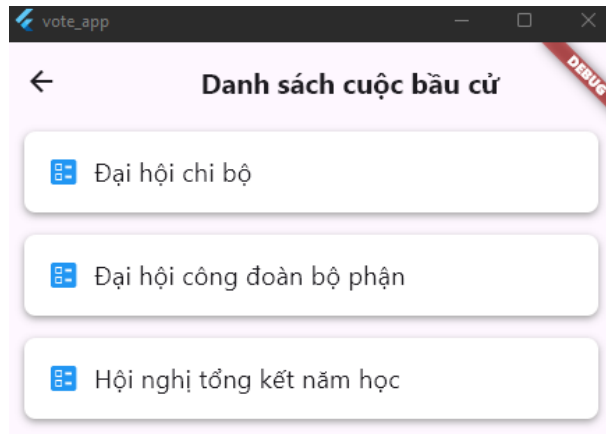
Hình 3. 10. Thống kê kết quả cuộc bỏ phiếu

b) Chức năng của người bỏ phiếu

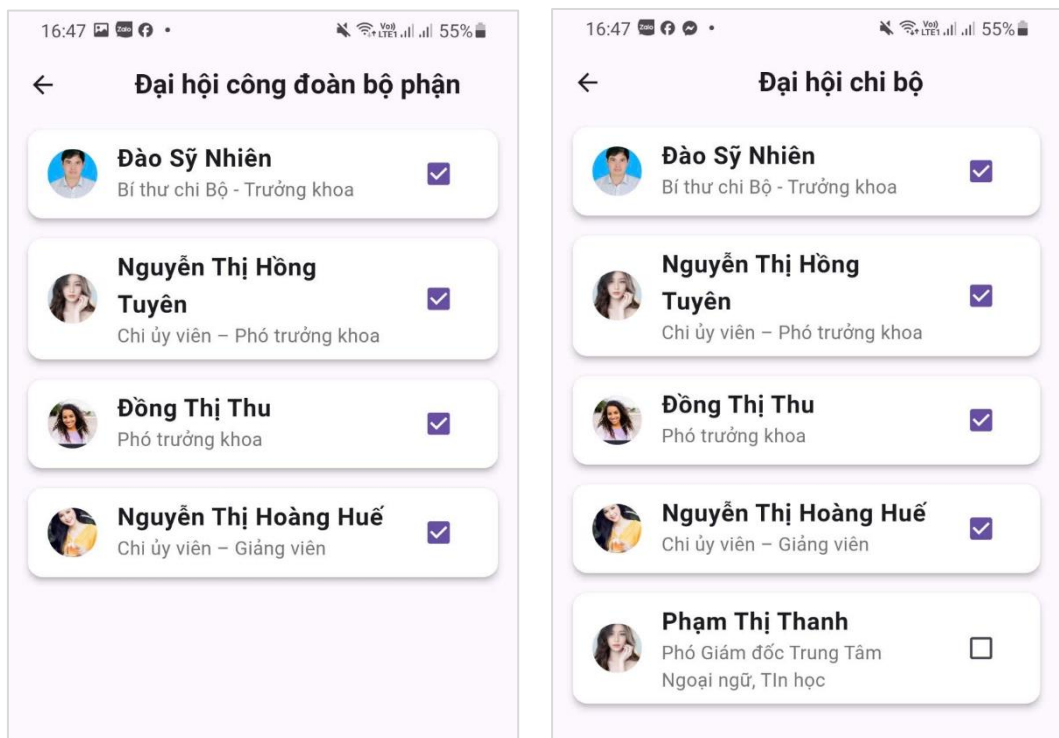
* Đăng nhập hệ thống



Hình 3. 11. Đăng nhập hệ thống



Hình 3. 12. Giao diện thể hiện Danh sách các cuộc bầu cử



Hình 3. 13. Giao diện thể hiện về các ứng cử viên

3.4.4. Đánh giá kết quả

Bước cài đặt thử nghiệm phần mềm bỏ phiếu trên thiết bị di động ta nhận kết quả là hệ thống hoạt động cơ bản đảm bảo yêu cầu:

- Thư ký có thể tạo, nhập danh sách ứng cử viên, quản lý các cuộc bỏ phiếu.
- Cử tri tham gia bỏ phiếu cài đặt ứng dụng, chọn cuộc bỏ phiếu và tích chọn cho những người được tín nhiệm.

Tuy nhiên còn có những chức năng chưa được hoàn thiện như quản lý tài khoản, phân quyền và mức độ bảo mật chỉ là cơ bản.

KẾT LUẬN VÀ KIẾN NGHỊ

KẾT LUẬN

Những kết quả hệ thống bỏ phiếu trực tuyến đã thực hiện được:

Qua quá trình phát triển và thử nghiệm, hệ thống bỏ phiếu trực tuyến đã đạt được các yêu cầu cơ bản về tính năng, bảo mật ban đầu và trải nghiệm người dùng. Cụ thể:

- Với việc hoàn thiện backend trên Laravel, hệ thống cho phép người quản trị tạo, quản lý các cuộc bỏ phiếu, kết quả bỏ phiếu hiển thị ngay tức thì, công khai minh bạch

- Frontend trên Flutter: hệ thống cho phép người dùng tham gia bỏ phiếu nhanh chóng, dễ dàng bằng cách đăng nhập vào hệ thống, chọn cuộc bỏ phiếu và tích chọn cho các ứng cử viên được tín nhiệm.

Kết quả thử nghiệm cho thấy hệ thống vận hành ổn định, đáp ứng các yêu cầu về hiệu suất. Đây là nền tảng tốt để phát triển thêm các tính năng mở rộng trong tương lai.

Những tồn tại mà hệ thống chưa thực hiện được:

- Hệ thống mới chỉ cài đặt chạy trên hệ điều hành Android, hệ điều hành iOS chưa tạo được file cài đặt (nhóm đề tài không có máy tính Macbook để xuất bản).

- Đề tài mới chỉ thực hiện trên 01 phạm vi đó là các cuộc bỏ phiếu diễn ra ở cấp trường, chưa phân định các quyền với mức độ khác nhau để thực hiện các cuộc bỏ phiếu diễn ra ở cấp cơ sở (Khoa, phòng, ban, Trung tâm).

- Bất kỳ người sử dụng nào cài đặt phần mềm trên điện thoại đều có thể tiến hành bỏ phiếu (có thể bỏ phiếu được nhiều lần liên tục), hệ thống chưa có tính năng quản lý tài khoản, xác thực 2 lớp.

KIẾN NGHỊ

Để hệ thống bỏ phiếu trực tuyến đạt hiệu quả cao hơn và phục vụ đưa vào sử dụng, nhóm đề tài khuyến nghị một số cải tiến như sau:

- Hoàn thiện chức năng để mỗi cử tri có thể chỉ có thể bầu chọn duy nhất một lần.

- Hoàn thiện các chức năng quản lý tài khoản, phân quyền, tăng cường bảo mật thông qua xác thực hai lớp (2FA), tối ưu hóa hiệu suất với caching và tối ưu cơ sở dữ liệu, phát triển thêm các tính năng thống kê, thông báo và nâng cao giao diện UI/UX để cải thiện trải nghiệm người dùng.

- Hoàn thiện ứng dụng trên hệ điều hành iOS: Để có thể tạo file cài đặt IPA cho điện thoại cài hệ điều hành iOS cần:

- Xcode (là một môi trường phát triển tích hợp (IDE) do Apple phát triển, được thiết kế dành riêng cho việc xây dựng ứng dụng trên các hệ điều hành của Apple như iOS, macOS, watchOS và tvOS).

- Máy tính Macbook

- Yêu cầu server chứa phần quản trị dữ liệu (backend):

- Máy chủ web: Apache hoặc Nginx

- Cơ sở dữ liệu: MySQL hoặc MariaDB

- Công cụ khác: Trình quản lý phụ thuộc cho PHP (Composer), cần thiết để cài đặt các gói và thư viện của Laravel.

TÀI LIỆU THAM KHẢO

- [1] Bộ Thông tin và Truyền thông (2021). *Cẩm nang chuyển đổi số*, Thông tin và Truyền thông.
- [2] Dusan Petkovic (2020). *Microsoft SQL Server 2019 A Beginner's Guide Edition 7*, McGraw-Hill Education eBooks.
- [3] Iggy Fernandez (2008). *Beginning Oracle Database 11g Administration*, Berkeley, CA
- [4] Lê Hồng Duy (2024). *Lập trình Flutter toàn tập*. <https://laptrinh-online.vn/course/lap-trinh-flutter-co-ban>.
- [5] Matt Stauffer (2019). *Laravel a Framework for Building Modern PHP Apps*, O'Reilly.
- [6] Navicat (2024). *Cách thức sử dụng navicat-premium để xây dựng cơ sở dữ liệu trong MySQL*, Navicat Premium | Connects to multiple databases on a single GUI
- [7] Freetuts (2024). *Tài liệu MySQL nâng cao* (<https://freetuts.net/hoc-mysql/mysql-nang-cao>).
- [8] Phạm Văn Cường (2020). *Giáo trình Phân tích thiết kế hệ thống thông tin*, Học viện Công nghệ Bưu chính Viễn thông.
- [9] Sourcee (2024). *Cách áp dụng XAMPP để thực hiện tạo lập cơ sở dữ liệu với công cụ phpMyAdmin* (<https://sourceforge.net/projects/xampp/>).
- [10] Ray Harris - Joel Murach (2019). *Lập Trình Cơ Bản PHP và MySQL*, Bách khoa Hà Nội.
- [11] Regina Obe và Leo Hsu (2012). *PostgreSQL: Up and Running*, O'Reilly.