

BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC PHENIKAA



## **BÀI TẬP LỚN**

### **HỌC PHẦN: CÔNG NGHỆ .NET**

**Đề tài: Xây dựng phần mềm hỗ trợ quá trình xử phạt vi phạm  
hành chính lĩnh vực lâm nghiệp trên nền tảng mobile**

Giảng viên hướng dẫn : Phạm Văn Hà  
Khoa chuyên môn : Công nghệ thông tin  
Nhóm sinh viên thực hiện : 03

Họ và tên	Mã sinh viên
1. Dương Thị Thu An	20010835
2. Nguyễn Đại Phát	21010625
3. Lương Tú Đạt	21011592
4. Nguyễn Anh Tuấn	21010626
5. Trịnh Huy Hoàng	21010622
6. Nguyễn Trọng Huynh	21012501

HÀ NỘI, 6/2023

# MỤC LỤC

MỤC LỤC .....	2
MỤC LỤC HÌNH ẢNH .....	3
MỤC LỤC BẢNG .....	4
LỜI MỞ ĐẦU .....	7
CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN .....	8
<b>1. Đặt vấn đề .....</b>	<b>8</b>
<b>2. Tổng quan về Xamarin.....</b>	<b>8</b>
2.1. Xamarin là gì? .....	8
2.2. Một số ứng dụng.....	9
CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG .....	11
<b>1. Phân tích yêu cầu của khách hàng.....</b>	<b>11</b>
<b>2. Thiết kế chức năng hệ thống .....</b>	<b>11</b>
<b>3. Thiết kế cơ sở dữ liệu .....</b>	<b>12</b>
<b>4. Thiết kế giao diện .....</b>	<b>15</b>
CHƯƠNG 3: PHÂN TÍCH CHƯƠNG TRÌNH.....	20
<b>1. Công nghệ và kỹ thuật áp dụng .....</b>	<b>20</b>
<b>2. Thiết kế chương trình hoạt động .....</b>	<b>20</b>
2.1. Sơ đồ hoạt động .....	20
2.2. Sơ đồ Use case.....	21
2.3. Đặc tả Use case.....	21
CHƯƠNG 4: KẾT LUẬN.....	24
<b>1. Đánh giá .....</b>	<b>24</b>
<b>2. Kết luận.....</b>	<b>24</b>

## MỤC LỤC HÌNH ẢNH

Hình 1: Giao diện khách khi chưa đăng nhập .....	15
Hình 2: Thông tin chi tiết, tìm kiếm và trang flyout.....	16
Hình 3: Giao diện giới thiệu ứng dụng, đăng nhập và đăng ký .....	17
Hình 4: Giao diện người dùng và admin.....	18
Hình 5: Các chức năng chính của admin .....	19
Hình 6: Sơ đồ hoạt động.....	20
Hình 7: Sơ đồ Use case .....	21

## MỤC LỤC BẢNG

Bảng 1: Bảng phân chia công việc.....	5
Bảng 2: Use case đăng nhập .....	21
Bảng 3: Use case thêm chương, điều, khoản.....	22
Bảng 4: Use case sửa chương, điều, khoản.....	22
Bảng 5: Use case xóa chương, điều, khoản.....	23
Bảng 6: Use case xem chi tiết chương, điều, khoản.....	23

## Phân chia công việc

*Bảng 1: Bảng phân chia công việc*

STT	Thông tin sinh viên	Công việc
1	Nguyễn Đại Phát – 21010625 – K15CNTT2 ( Nhóm trưởng )	(Backend) Kết nối database, hiện dữ liệu ra màn hình, xử lý đăng kí đăng nhập, đăng xuất, xét quyền admin, tìm kiếm, xem chi tiết các chapter, article và section Duyệt và sửa báo cáo lần 2
2	Dương Thị Thu An – 20010835 - K14CNTT3	(Frontend) Màn hình chính cho các chapter, article, section, chỉnh sửa các giao diện frontend còn lại. Thêm và sửa báo cáo lần 1
3	Lương Tú Đạt – 21011592- K15CNTT2	(Frontend) Giao diện detail cho chapter, article, section. Làm báo cáo chương 1
4	Trịnh Huy Hoàng – 21010622- K15CNTT2	(Frontend) Giao diện add, edit cho chapter, article, section. Làm báo cáo chương 2
5	Nguyễn Anh Tuấn – 21010626 – K15CNTT2	(Frontend) Giao diện đăng nhập, đăng kí, flyout, trang thông tin. Làm báo cáo chương 3
6	Nguyễn Trọng Huỳnh – 21012501 – K15CNTT5	(Backend) Thêm, sửa, xóa các chapter, article và section. Làm báo cáo chương 4

## LỜI CẢM ƠN

Chúng em xin gửi lời cảm ơn chân thành đến thầy *Phạm Văn Hà* – giảng viên *khoa Công nghệ thông tin* đã trang bị cho chúng em những kiến thức, kỹ năng cơ bản cần có để hoàn thành đề tài nghiên cứu này.

Tuy nhiên trong quá trình nghiên cứu đề tài, do kiến thức chuyên ngành còn hạn chế nên chúng em vẫn còn nhiều thiếu sót khi tìm hiểu, đánh giá và trình bày về đề tài. Rất mong nhận được sự quan tâm, góp ý của thầy/cô giảng viên bộ môn để đề tài của chúng em được đầy đủ và hoàn chỉnh.

*Xin chân thành cảm ơn!*

## **LỜI MỞ ĐẦU**

Trong thời đại công nghệ hiện đại, việc áp dụng phần mềm để tối ưu hóa các quy trình hành chính đã trở thành xu hướng không thể phủ nhận. Trong lĩnh vực lâm nghiệp, việc xử phạt vi phạm hành chính là một quá trình quan trọng nhằm bảo vệ tài nguyên và đảm bảo sự tuân thủ các quy định pháp luật. Tuy nhiên, việc thực hiện quy trình xử phạt vi phạm này vẫn gặp phải nhiều khó khăn và hạn chế trong việc thu thập thông tin, xử lý dữ liệu và tương tác với người dùng.

Để giải quyết những vấn đề này, đề tài này tập trung vào xây dựng một phần mềm hỗ trợ quá trình xử phạt vi phạm hành chính lĩnh vực lâm nghiệp trên nền tảng mobile sử dụng công nghệ Xamarin Forms và ngôn ngữ lập trình C#. Đây là một nền tảng mạnh mẽ và linh hoạt, cung cấp các công cụ và khung phát triển hiệu quả cho việc xây dựng ứng dụng di động.

Phần mềm được thiết kế nhằm tối ưu hóa quy trình xử phạt vi phạm, giúp người sử dụng thu thập thông tin nhanh chóng và chính xác, xử lý dữ liệu một cách hiệu quả và tương tác trực tiếp với các bên liên quan. Đồng thời, giao diện người dùng thân thiện và dễ sử dụng trên các thiết bị di động sẽ giúp nâng cao hiệu quả và tốc độ trong quá trình xử phạt vi phạm hành chính.

Phạm vi của nghiên cứu sẽ tập trung vào xây dựng hệ thống tra cứu pháp luật trên mobile, bao gồm khả năng tìm kiếm, tra cứu thông tin chi tiết, đồng bộ hóa dữ liệu và thông báo cập nhật. Nghiên cứu sẽ tập trung vào việc sử dụng công nghệ Xamarin Forms và ngôn ngữ lập trình C# để xây dựng giao diện người dùng và xử lý logic của hệ thống.

Về mặt khoa học, đề tài đóng góp vào lĩnh vực công nghệ phần mềm và phát triển ứng dụng di động. Việc sử dụng công nghệ Xamarin Forms và ngôn ngữ C# trong xây dựng hệ thống tra cứu pháp luật trên mobile mang lại những giải pháp kỹ thuật mới, khả năng tái sử dụng mã nguồn và tăng tính tương thích đa nền tảng. Điều này có ý nghĩa trong việc phát triển các ứng dụng di động khác trong tương lai.

## CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN

### 1. Đặt vấn đề

Trong lĩnh vực lâm nghiệp, việc duy trì sự tuân thủ các quy định và quy trình pháp luật là rất quan trọng để bảo vệ tài nguyên và đảm bảo sự phát triển bền vững. Quá trình xử phạt vi phạm hành chính đóng vai trò then chốt trong việc thực hiện các biện pháp kiểm soát và giám sát hợp lý.

Tuy nhiên, quá trình này vẫn đối mặt với nhiều thách thức. Hiện nay, hầu hết các quy trình xử phạt vi phạm trong lĩnh vực lâm nghiệp vẫn được thực hiện thủ công, sử dụng các hồ sơ giấy và quy trình rườm rà. Điều này gây ra sự chậm trễ, phức tạp và dễ xảy ra sai sót trong việc thu thập thông tin, xử lý dữ liệu và tương tác với người dùng.

Để giải quyết những hạn chế này, đề tài này tập trung vào xây dựng một phần mềm hỗ trợ quá trình xử phạt vi phạm hành chính lĩnh vực lâm nghiệp trên nền tảng mobile. Bằng cách áp dụng công nghệ di động và phát triển trên nền tảng dotnet, phần mềm này sẽ đem lại nhiều lợi ích đáng kể.

Với phần mềm hỗ trợ, quy trình xử phạt vi phạm hành chính sẽ được tự động hóa và tối ưu hóa, giúp giảm thiểu thời gian và công sức trong việc thu thập và xử lý thông tin. Các hồ sơ vi phạm, biên bản xử phạt, và các tài liệu liên quan sẽ được quản lý một cách hiệu quả và an toàn, tránh mất mát hoặc sai sót.

Giao diện người dùng thân thiện và tương tác trực tiếp trên các thiết bị di động giúp nâng cao sự tiện lợi và linh hoạt. Nhân viên có thể nhập liệu, xử lý và truy cập dữ liệu mọi lúc, mọi nơi thông qua các thiết bị di động, giúp tăng cường khả năng làm việc và đảm bảo tính chính xác của quy trình xử phạt.

### 2. Tổng quan về Xamarin

#### 2.1. Xamarin là gì?

Xamarin là một nền tảng phát triển ứng dụng di động được ra mắt vào năm 2011, được mua lại bởi Microsoft vào năm 2016. Nền tảng này cho phép các nhà phát triển xây dựng ứng dụng di động đa nền tảng sử dụng ngôn ngữ lập trình C# và .NET.

Truyền thống, việc phát triển ứng dụng di động đa nền tảng đòi hỏi việc xây dựng và duy trì các mã nguồn riêng biệt cho từng nền tảng, chẳng hạn như Android, iOS và Windows. Tuy nhiên, Xamarin giải quyết vấn đề này bằng cách cho phép nhà phát triển chia sẻ mã nguồn chung cho tất cả các nền tảng, từ đó giảm thiểu công sức và thời gian phát triển.



Với Xamarin, người phát triển có thể viết mã nguồn một lần duy nhất bằng C# và sử dụng các thư viện và API của .NET để xây dựng ứng dụng di động cho Android, iOS và Windows. Xamarin sử dụng một công cụ gọi là Xamarin.Forms để tạo giao diện người dùng đa nền tảng, cho phép bạn thiết kế một lần và chạy trên nhiều nền tảng khác nhau.

Xamarin cung cấp môi trường phát triển tích hợp (IDE) là Visual Studio, cho phép các nhà phát triển sử dụng các tính năng mạnh mẽ và công cụ hỗ trợ để phát triển ứng dụng di động. Ngoài ra, Xamarin cũng cung cấp khả năng truy cập đầy đủ vào API và tính năng của mỗi nền tảng, cho phép bạn tận dụng sức mạnh của từng hệ điều hành di động.

Tóm lại, Xamarin là một nền tảng phát triển ứng dụng di động đa nền tảng mạnh mẽ, cho phép nhà phát triển xây dựng ứng dụng di động chất lượng cao sử dụng C# và .NET và chia sẻ mã nguồn giữa các nền tảng khác nhau.

## **2.2. Một số ứng dụng**

Xamarin đã được sử dụng rộng rãi trong nhiều lĩnh vực khác nhau để phát triển ứng dụng di động đa nền tảng. Dưới đây là một số ví dụ về ứng dụng Xamarin trong các lĩnh vực khác nhau:

1. Lĩnh vực công nghệ y tế: Các ứng dụng di động được xây dựng bằng Xamarin có thể được sử dụng trong y tế để theo dõi sức khỏe cá nhân, đặt lịch hẹn khám bệnh, gửi thông báo nhắc nhở đến bệnh nhân, cung cấp tư vấn sức khỏe, và quản lý dữ liệu y tế.
2. Lĩnh vực giáo dục: Xamarin có thể được sử dụng để phát triển các ứng dụng giáo dục đa nền tảng như ứng dụng học trực tuyến, ứng dụng từ điển, ứng dụng học ngôn ngữ, và ứng dụng quản lý học tập.
3. Lĩnh vực du lịch và khách sạn: Xamarin cho phép phát triển các ứng dụng đa nền tảng cho việc đặt phòng khách sạn, tìm kiếm thông tin du lịch, đưa ra đề xuất điểm tham quan, và cung cấp dịch vụ hướng dẫn du lịch.
4. Lĩnh vực thương mại điện tử: Xamarin có thể được sử dụng để xây dựng các ứng dụng mua sắm trực tuyến, ứng dụng thanh toán di động, và ứng dụng quản lý đơn hàng trong lĩnh vực thương mại điện tử.
5. Lĩnh vực giải trí và truyền thông: Các ứng dụng di động cho phát trực tiếp video, xem phim, nghe nhạc, đọc tin tức, và tương tác với nội dung truyền thông đã được phát triển bằng Xamarin.

6. Lĩnh vực logistics và vận chuyển: Xamarin có thể được sử dụng để xây dựng các ứng dụng quản lý định vị, theo dõi hàng hóa, quản lý lịch trình vận chuyển, và cung cấp thông tin vận chuyển cho người dùng.

## CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### 1. Phân tích yêu cầu của khách hàng

Dựa trên yêu cầu của khách hàng, hệ thống sẽ được thiết kế và giải quyết theo hướng sử dụng công nghệ Xamarin Forms và ngôn ngữ lập trình C#.

- Tìm kiếm nhanh chóng: Để đáp ứng yêu cầu này, chúng ta sẽ sử dụng công nghệ Xamarin Forms để xây dựng giao diện người dùng đồng nhất trên cả hai nền tảng di động iOS và Android. C# sẽ được sử dụng để viết mã logic và xử lý tìm kiếm, sử dụng các thuật toán tìm kiếm hiệu quả để đảm bảo tìm kiếm nhanh chóng và chính xác.
- Tra cứu thông tin chi tiết: Xamarin Forms cung cấp khả năng xây dựng giao diện đa nền tảng, cho phép chúng ta hiển thị thông tin chi tiết của các văn bản pháp luật một cách trực quan trên cả iOS và Android. Bằng cách sử dụng C# và các thư viện hỗ trợ, chúng ta có thể truy cập và hiển thị nội dung chi tiết của các văn bản pháp luật, bao gồm cả các điều khoản, mục lục và thông tin liên quan.
- Đồng bộ hóa dữ liệu: Công nghệ Xamarin Forms cho phép chúng ta kết nối và giao tiếp với các nguồn dữ liệu pháp luật từ các nguồn khác nhau. Sử dụng C# và các thư viện hỗ trợ, chúng ta có thể thực hiện việc đồng bộ hóa dữ liệu để đảm bảo rằng thông tin pháp luật trong hệ thống luôn được cập nhật và đầy đủ.
- Giao diện thân thiện với người dùng: Sử dụng Xamarin Forms, chúng ta có thể xây dựng giao diện người dùng đồng nhất và tương thích trên cả hai nền tảng di động iOS và Android. Giao diện sẽ được thiết kế để đơn giản, trực quan và dễ sử dụng, đảm bảo người dùng có trải nghiệm tốt khi sử dụng hệ thống tra cứu pháp luật trên mobile.
- Quyền truy cập và bảo mật: Sử dụng C# và các thư viện bảo mật, chúng ta có thể xây dựng các biện pháp bảo mật để đảm bảo quyền truy cập và bảo mật thông tin pháp luật. Các biện pháp như xác thực người dùng, mã hóa dữ liệu và các quy tắc phân quyền sẽ được áp dụng để đảm bảo an toàn thông tin.

### 2. Thiết kế chức năng hệ thống

Hệ thống gồm những chức năng sau :

- Tìm kiếm thông tin: Khi người dùng gõ vào ô tìm kiếm từ khóa sẽ hiện lên các điều khoản tương ứng với từ khóa đó. Sau khi gõ vào ô tìm kiếm chữ vận chuyển lâm sản hệ thống sẽ liệt kê các hành vi về vận chuyển lâm sản – khung xử phạt – mức xử phạt – thẩm quyền xử phạt.
- Hiển thị thông tin: Khi người dùng bấm vào nút hiển thị thông tin sẽ hiển thị ra các trường thông tin như content, create time, decree,...

- Cập nhật thông tin: Hệ thống sẽ cho phép nhóm quản trị (Admin) quyền có thể cập nhật thông tin của các điều khoản.

+ Khi chọn 1 Chương/Mục/Điều/Khoản, hệ thống hiển thị nút Sửa hoặc Xóa.

+ Người dùng chọn “Sửa”

+ Hệ thống sẽ hiển thị các Textbox cho phép sửa các thông tin liên quan.

+ Người dùng Sửa các thông tin liên quan.

+ Người dùng chọn “Lưu” để lưu thông tin hoặc “Hủy” để hủy bỏ quá trình sửa.

- Xóa thông tin.

+ Khi chọn 1 Chương/Mục/Điều/Khoản, hệ thống hiển thị nút Sửa hoặc Xóa

+ Người dùng chọn “Xóa”

+ Hệ thống hiển thị hộp thoại “Có chắc chắn muốn xóa hay không”

+ Người dùng chọn “Có” để xóa thông tin hoặc “Hủy” để hủy bỏ quá trình xóa.

- Thêm thông tin: Hệ thống sẽ cho phép nhóm quản trị (Admin) quyền có thể thêm thông tin của các điều khoản

+ Khi bấm vào nút “Thêm” hệ thống sẽ hiển thị Textbox cho phép nhập các thông tin liên quan đến các Chương/Mục/Điều/Khoản mới.

+ Người dùng thêm các thông tin liên quan.

+ Người dùng chọn “Lưu” để lưu thông tin hoặc “Hủy” để hủy bỏ quá trình sửa.

### **3. Thiết kế cơ sở dữ liệu**

Cơ sở dữ liệu của ứng dụng tìm kiếm văn bản pháp luật là nơi lưu trữ thông tin về các văn bản luật.

Dưới đây là cấu trúc cơ sở dữ liệu cho hệ thống:

## - User

Tên	Kiểu dữ liệu	Mô tả
ID	INTEGER	Mã duy nhất định danh cho người dùng.
User Name	TEXT	Tên đăng nhập của người dùng.
Password	TEXT	Mật khẩu được mã hóa của người dùng.
Email	TEXT	Địa chỉ email của người dùng.
Mobile	TEXT	Số điện thoại người dùng.
Role	TEXT	Quyền truy cập người dùng
Ràng buộc: ID là trường khóa chính (PRIMARY KEY) và không được trùng lặp. User Name là trường duy nhất (UNIQUE) để đảm bảo không có tên đăng nhập trùng nhau.		

## - Chapters

Tên	Kiểu dữ liệu	Mô tả
ID	INTEGER	Mã duy nhất định danh cho chương.
Title	TEXT	Tên chương.
Content	TEXT	Nội dung chương.
Create Time	TEXT	Thời gian tạo chương.
Update Time	TEXT	Thời gian sửa chương.
Decree	TEXT	Số nghị định.
Ràng buộc: ID là trường khóa chính (PRIMARY KEY) và không được trùng lặp.		

## - Articles

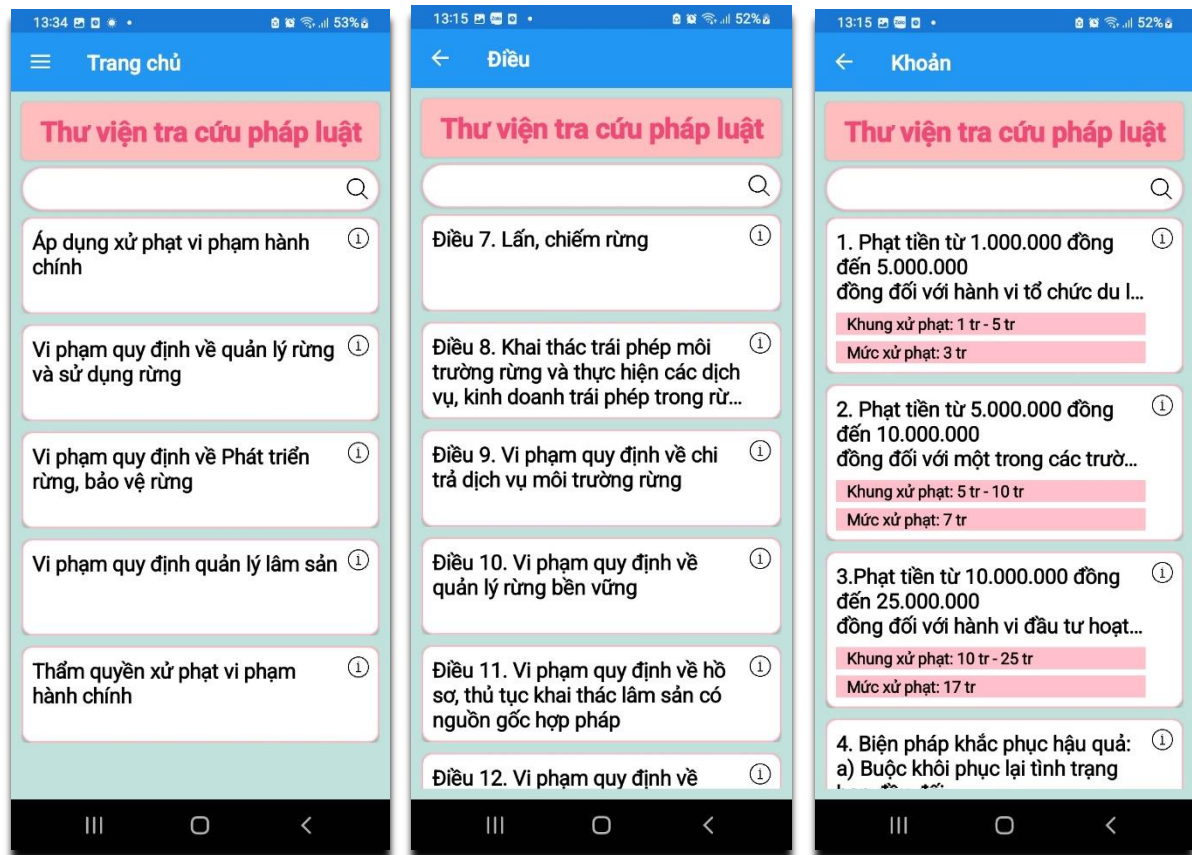
Tên	Kiểu dữ liệu	Mô tả
ID	INTEGER	Mã duy nhất định danh cho điều.
Title	TEXT	Tên điều.
Create Time	TEXT	Thời gian tạo điều.
Update Time	TEXT	Thời gian sửa điều.
Chapter ID	TEXT	Mã duy nhất định danh cho chương.
Ràng buộc: ID là trường khóa chính (PRIMARY KEY) và không được trùng lặp. Chapter ID là trường khóa ngoại (FOREIGN KEY) liên kết với trường ID trong bảng “Chapter”.		

## - Sections

Tên	Kiểu dữ liệu	Mô tả
ID	INTEGER	Mã duy nhất định danh cho khoản.
Title	TEXT	Tên khoản.
Content	TEXT	Nội dung khoản.
Min	TEXT	Mức phạt thấp nhất cho khoản.
Max	TEXT	Mức phạt cao nhất cho khoản.
Avg	TEXT	Mức phạt trung bình cho khoản.
Create Time	TEXT	Thời gian tạo khoản.
Update Time	TEXT	Thời gian sửa khoản.
Article ID	TEXT	Mã duy nhất định danh cho điều.
Decree ID	TEXT	Mã duy nhất định danh cho nghị định.
Ràng buộc: ID là trường khóa chính (PRIMARY KEY) và không được trùng lặp. Article ID là trường khóa ngoại (FOREIGN KEY) liên kết với trường ID trong bảng “ Articles”		

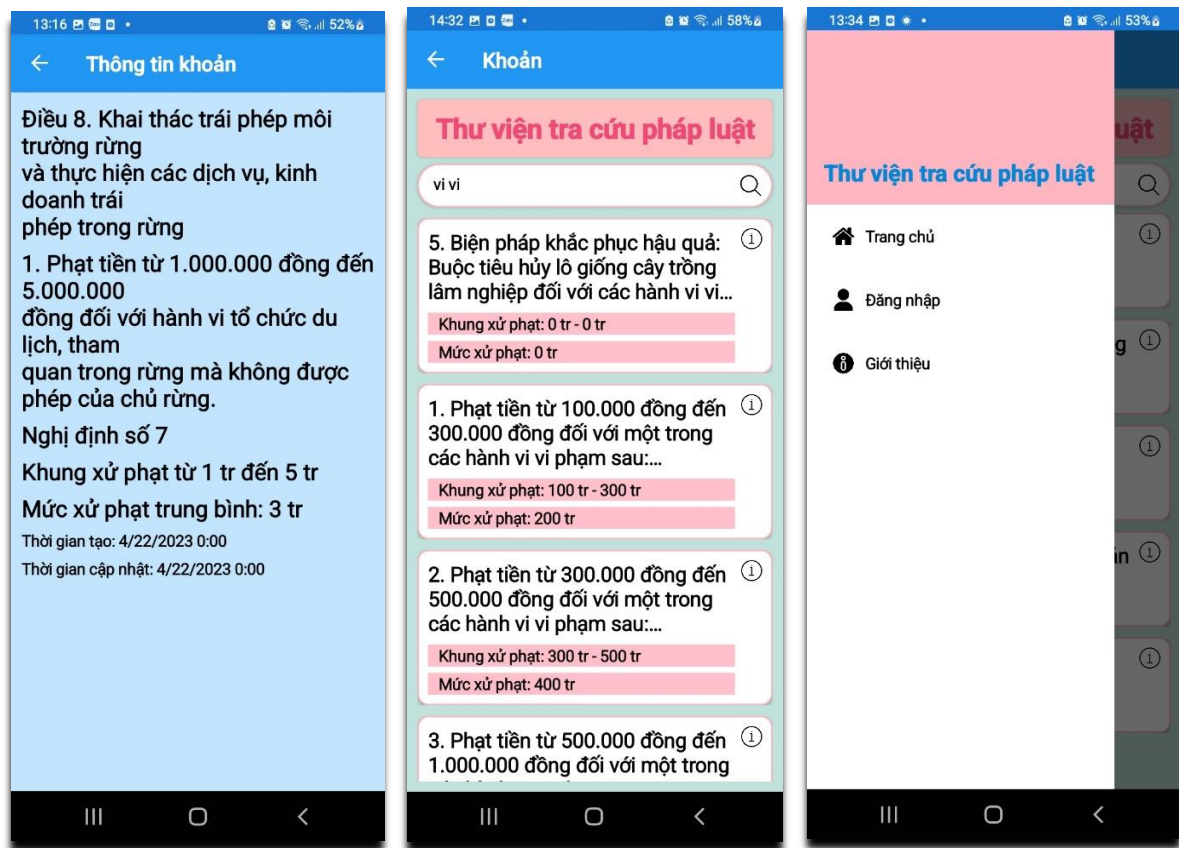
## 4. Thiết kế giao diện

Sau đây là các giao diện của hệ thống



Hình 1: Giao diện khách khi chưa đăng nhập

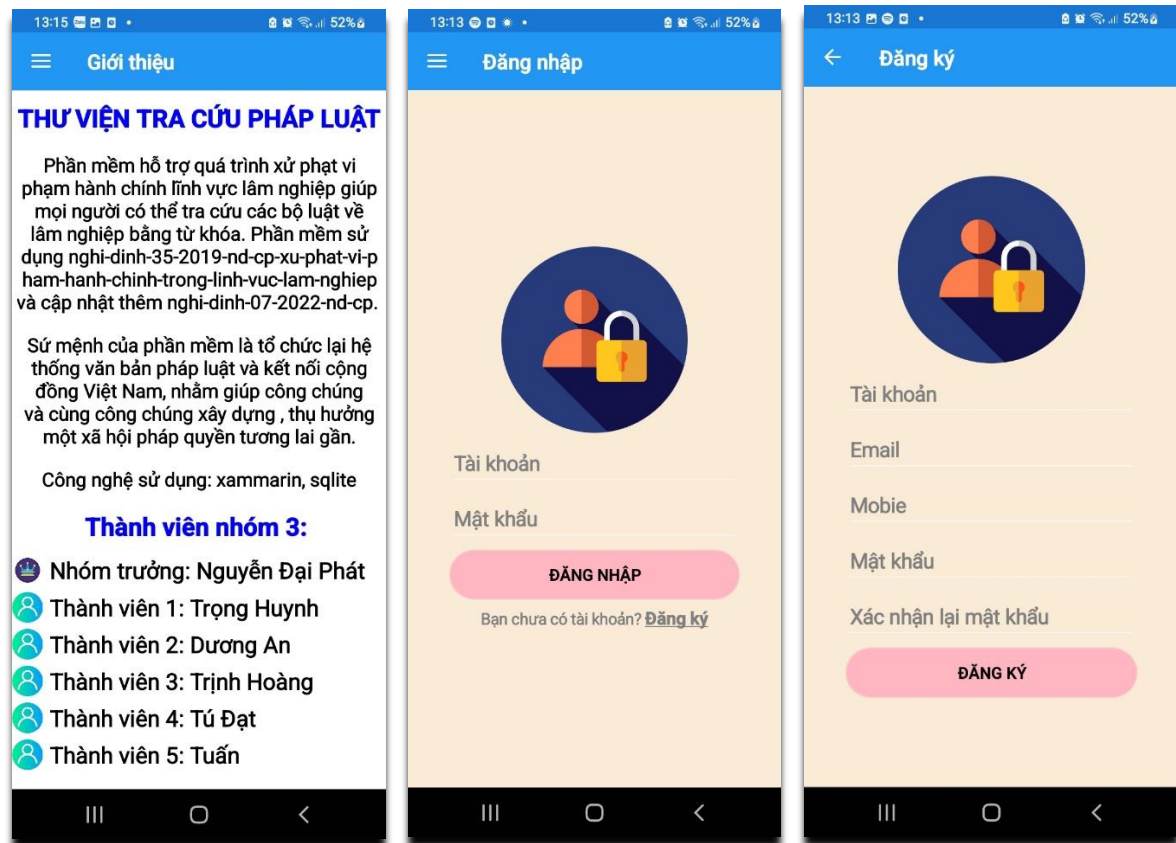
Hình 1 là giao diện khi người dùng mở ứng dụng chưa đăng nhập (khách). Giao diện trang chủ sẽ được hiển thị mặc định bao gồm tiêu đề, thanh ngang tùy chọn, thanh tìm kiếm và danh sách các Chương. Danh sách các chương, điều, khoản đều được cố định hiển thị nhiều nhất là 3 dòng và một nút xem chi tiết. Khi người dùng chọn một Chương bất kì, hệ thống sẽ chuyển hướng sang trang Điều với các điều tương ứng của chương. Tương tự, khi chọn một Điều, hệ thống chuyển sang trang Khoản với các Khoản tương ứng.



Hình 2: Thông tin chi tiết, tìm kiếm và trang flyout

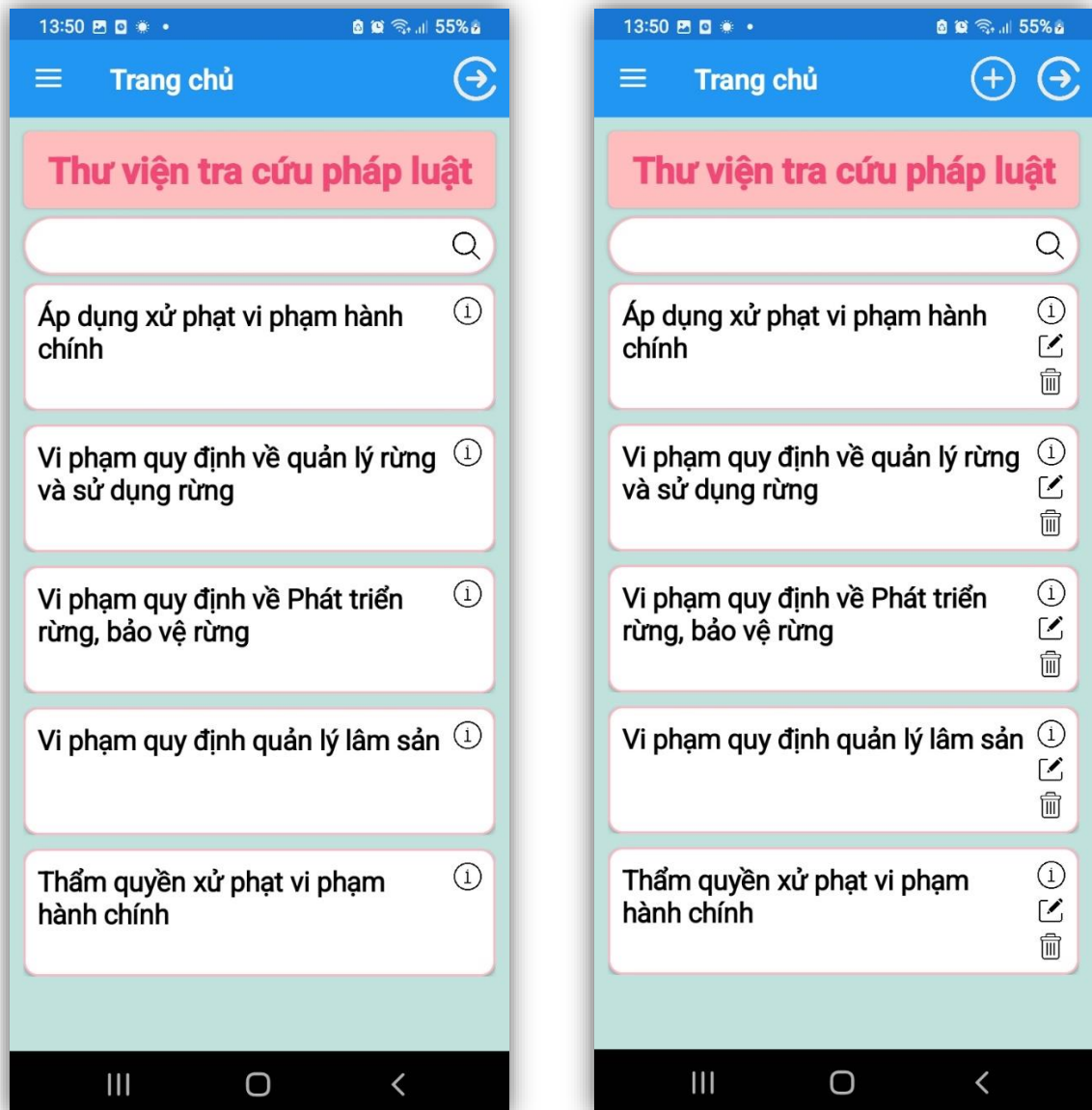
Khi người dùng (khách) bấm xem chi tiết, hệ thống chuyển hướng sang trang thông tin chi tiết Chương, Khoản hoặc Điều. Khi tìm kiếm, khách nhập văn bản bất kỳ và bấm nút, hệ thống sẽ hiện ra danh sách các Khoản với khung xử phạt và mức xử phạt trung bình tương ứng. Ngoài ra khách có thể bấm thanh ngang tùy chọn để hiển thị một trang flyout dùng để xem thông tin giới thiệu, đăng nhập hay đơn giản là trở về trang chủ.





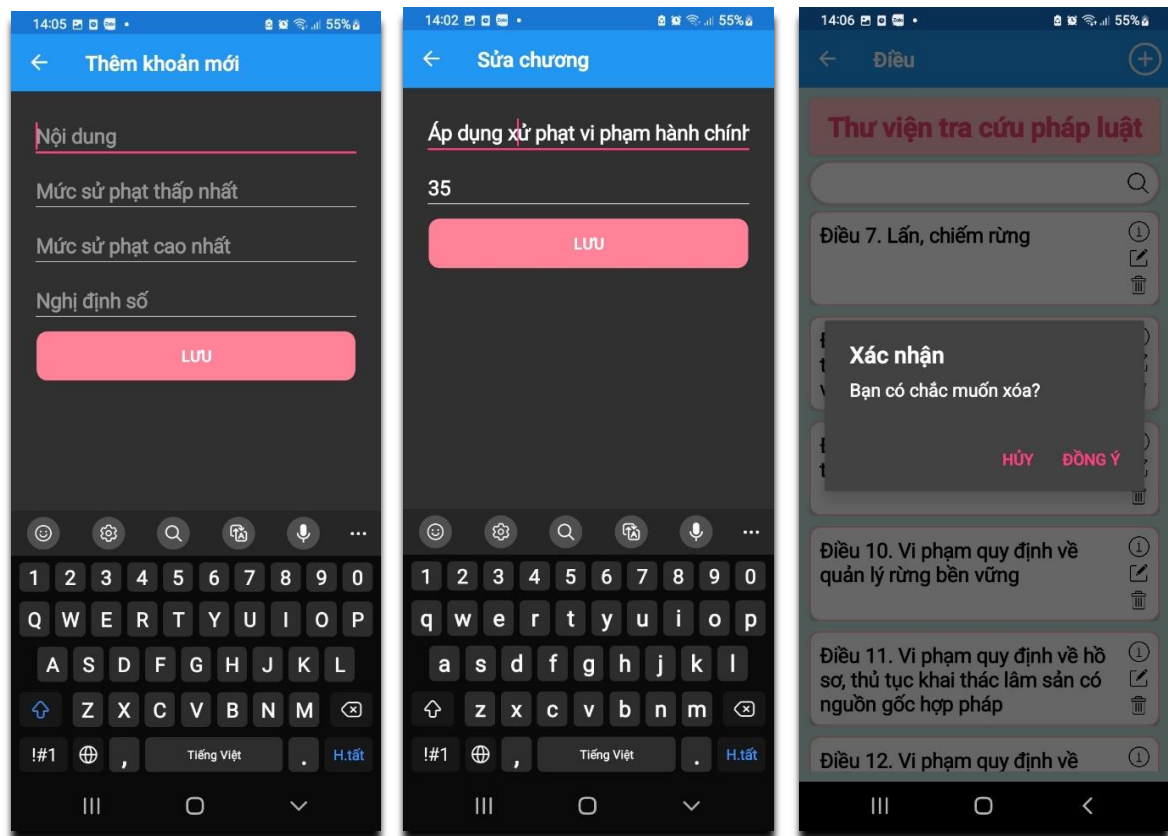
Hình 3: Giao diện giới thiệu ứng dụng, đăng nhập và đăng ký

Hình 3 bao gồm giao diện thông tin ứng dụng, đăng nhập và đăng ký. Giao diện thông tin ứng dụng gồm có tiêu đề, lợi ích, công cụ, công nghệ sử dụng và danh sách thành viên nhóm 3. Giao diện đăng nhập có 2 thanh văn bản cho người dùng (khách) nhập. Sau khi nhập xong, người dùng (khách) nhấn vào nút đăng nhập để chuyển đến giao diện trang chủ của hệ thống, nếu nhập sai thông tin tài khoản, mật khẩu hoặc không nhập thì hệ thống thông báo lỗi tương ứng. Nếu người dùng chưa có tài khoản thì nhấn vào chữ đăng ký bên dưới nút đăng nhập, hệ thống sẽ chuyển sang giao diện đăng kí để người dùng tạo tài khoản cho mình. Giao diện đăng kí tương tự như đăng nhập và có 5 thanh nhập tài khoản, email, mobie, mật khẩu và xác nhận mật khẩu.



Hình 4: Giao diện người dùng và admin

Sau khi đăng nhập, tùy vào vai trò của tài khoản, hệ thống sẽ dẫn người dùng (khách) tới giao diện người dùng hoặc giao diện admin. Giao diện người dùng có hết các chức năng của khách, ngoài ra có thêm chức năng đăng xuất. Khi nhấn nút đăng xuất, người dùng sẽ trở về trạng thái khách. Đối với giao diện admin thì sẽ có thêm 3 chức năng đó là thêm, sửa, xóa các chương, điều và khoản.



Hình 5: Các chức năng chính của admin

Hình 5 là một số ví dụ về chức năng chính của admin bao gồm thêm, sửa, xóa các chương, điều, khoản.

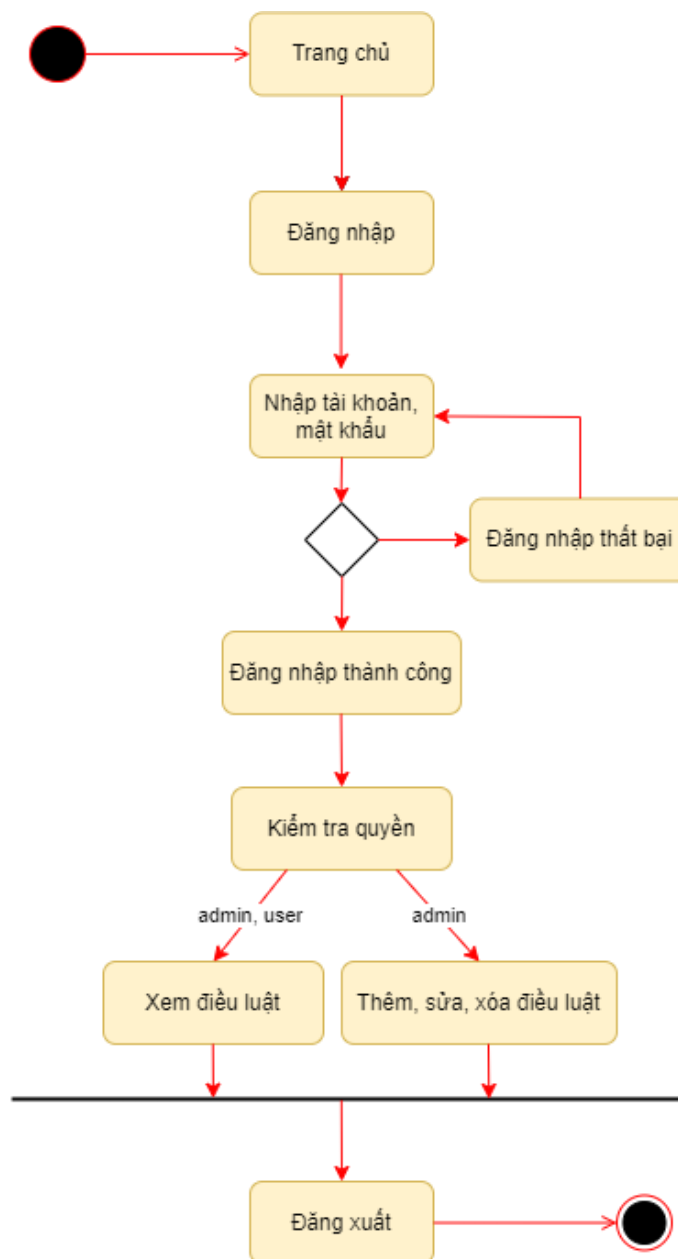
## CHƯƠNG 3: PHÂN TÍCH CHƯƠNG TRÌNH

### 1. Công nghệ và kỹ thuật áp dụng

- Ngôn ngữ lập trình: C#
- Môi trường: Xamarin Form
- Cơ sở dữ liệu: Sqlite

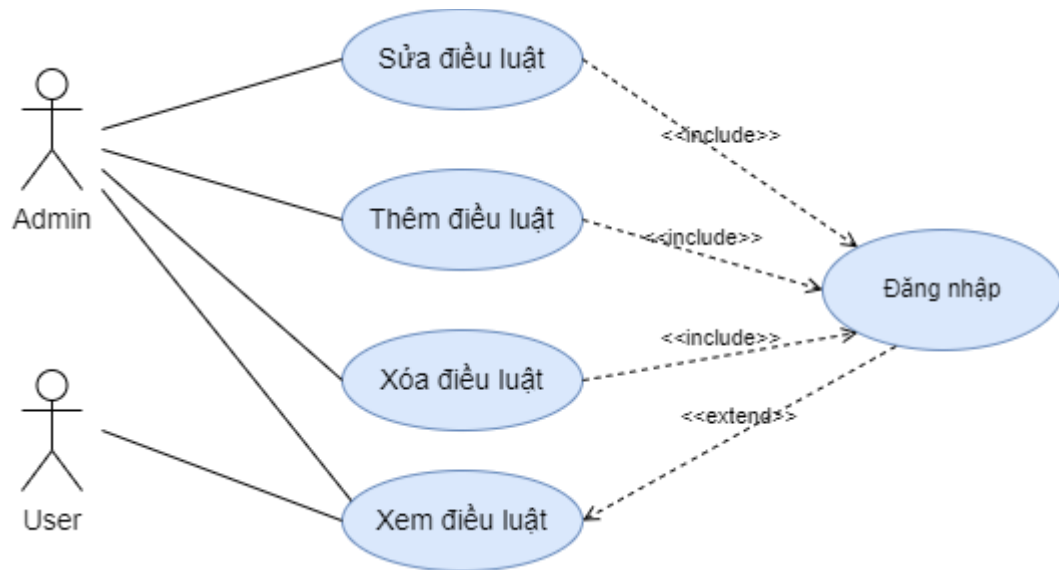
### 2. Thiết kế chương trình hoạt động

#### 2.1. Sơ đồ hoạt động



Hình 6: Sơ đồ hoạt động

## 2.2. Sơ đồ Use case



Hình 7: Sơ đồ Use case

## 2.3. Đặc tả Use case

Bảng 2: Use case đăng nhập

Use Case ID	UC1
Tên Use Case	Đăng nhập
Mô tả	Cho phép người sử dụng đăng nhập vào hệ thống thư viện tra cứu điều luật, tùy theo đối tượng người dùng mà hệ thống sẽ cho phép người dùng thực hiện các hoạt động khác nhau
Người sử dụng	Admin, User
Hoạt động	<p>Usecase này bắt đầu khi người dùng chọn đăng nhập vào hệ thống</p> <ol style="list-style-type: none"> <li>Hệ thống yêu cầu người dùng nhập tên đăng nhập và mật khẩu</li> <li>Hệ thống kiểm tra thông tin người dùng và xác nhận hợp lệ</li> <li>Tùy theo đối tượng người dùng, hệ thống sẽ đến các form với chức năng phù hợp</li> <li>Hệ thống thông báo đăng nhập thành công</li> <li>Người dùng có thể chọn đăng xuất bằng cách chọn đăng xuất trên màn hình</li> </ol>
	<p>Sai thông tin đăng nhập</p> <p>Nếu, trong luồng cơ sở, người dùng đăng nhập với những</p>

	thông tin không có trong cơ sở dữ liệu người dùng, thì hệ thống sẽ thông báo lỗi và yêu cầu đăng nhập lại. Ngoài ra, người dùng có thể hủy bỏ đăng nhập, kết thúc use case.
Yêu cầu phi chức năng	Mật khẩu của người dùng phải được mã hóa

Bảng 3: Use case thêm chương, điều, khoản

<b>Use Case ID</b>	<b>UC2</b>
Tên Use Case	Thêm chương, điều, khoản
Mô tả	Cho phép admin thêm chương, điều, khoản mới vào hệ thống thư viện tra cứu điều luật, tùy theo cấp độ điều luật admin thêm mà hệ thống sẽ hiển thị form thêm điều luật khác nhau
Người sử dụng	Admin
Hoạt động	<p>Use case này bắt đầu khi admin chọn biểu tượng dấu (+) trên màn hình</p> <ol style="list-style-type: none"> <li>Hệ thống yêu cầu admin nhập thông tin về điều luật theo form mẫu có sẵn và bấm “LƯU”</li> <li>Hệ thống kiểm tra thông tin và xác nhận hợp lệ</li> <li>Hệ thống thông báo thêm thành công, điều luật mới sẽ được hiển thị ra màn hình</li> <li>Ngoài ra admin có thể bấm biểu tượng (←) trên màn hình để kết thúc use case</li> </ol>

Bảng 4: Use case sửa chương, điều, khoản

<b>Use Case ID</b>	<b>UC3</b>
Tên Use Case	Sửa chương, điều, khoản
Mô tả	Cho phép admin sửa chương điều khoản có sẵn trong hệ thống thư viện tra cứu điều luật
Người sử dụng	Admin
Hoạt động	<p>Use case này bắt đầu khi admin chọn biểu tượng sửa trên màn hình</p> <ol style="list-style-type: none"> <li>Hệ thống hiển thị ra các thông tin có sẵn của điều luật</li> <li>Admin thực hiện sửa các thông tin mình muốn, sau đó nhấn “LƯU”</li> <li>Hệ thống thông báo sửa thành công, điều luật đã sửa sẽ được cập nhật hiện ra màn hình</li> </ol>

	4. Ngoài ra admin có thể bấm biểu tượng (←) trên màn hình để kết thúc use case
--	--

Bảng 5: Use case xóa chương, điều, khoản

Use Case ID	UC4
Tên Use Case	Xóa chương, điều, khoản
Mô tả	Cho phép admin xóa điều luật có sẵn trong hệ thống thư viện tra cứu điều luật
Người sử dụng	Admin
Hoạt động	<p>Use case này bắt đầu khi admin chọn biểu tượng xóa trên màn hình</p> <ol style="list-style-type: none"> <li>1. Hệ thống hiển thị yêu cầu xác nhận xóa</li> <li>2. Admin nhấn “ĐỒNG Ý”</li> <li>3. Hệ thống thông báo xóa thành công, điều luật được xóa khỏi màn hình</li> <li>4. Ngoài ra admin có thể bấm “HỦY” trên màn hình để kết thúc use case</li> </ol>

Bảng 6: Use case xem chi tiết chương, điều, khoản

Use Case ID	UC5
Tên Use Case	Xem chi tiết chương, điều, khoản
Mô tả	Cho phép người dùng xem chi tiết các thông tin của điều luật có sẵn trong hệ thống
Người sử dụng	Admin. User
Hoạt động	<p>Use case này bắt đầu khi người dùng chọn biểu tượng (i) trên màn hình</p> <ol style="list-style-type: none"> <li>1. Hệ thống hiển thị toàn bộ các thông tin của điều luật</li> <li>2. Ngoài ra admin có thể bấm biểu tượng (←) trên màn hình để kết thúc use case</li> </ol>

## CHƯƠNG 4: KẾT LUẬN

### 1. Đánh giá

#### ❖ Giao diện người dùng

Giao diện người dùng trong ứng dụng tra cứu luật được thiết kế một cách hấp dẫn và trực quan. Sử dụng Xamarin Forms, nhóm đã tận dụng các phần tử điều khiển có sẵn và khả năng tùy chỉnh để tạo ra một giao diện thân thiện với người dùng.

Người dùng có thể dễ dàng tương tác với ứng dụng và tìm kiếm thông tin pháp lý. Giao diện đa nền tảng của Xamarin Forms đã giúp nhóm đảm bảo nhất quán và trải nghiệm tốt trên các thiết bị di động khác nhau.

#### ❖ Chức năng

Ứng dụng đáp ứng được yêu cầu đầy đủ và chính xác về thông tin pháp lý. Các chức năng cần thiết dành cho admin và người dùng đã được phân nhánh và triển khai một cách hiệu quả trên các nền tảng di động.

#### ❖ Khả năng mở rộng

Ứng dụng có khả năng mở rộng và cập nhật trong tương lai. Tái sử dụng mã nguồn và khả năng phát triển đa nền tảng của Xamarin Forms chính là cơ sở để mở rộng ứng dụng một cách dễ dàng và bảo trì nhanh chóng. Điều này sẽ giúp nhóm cập nhật và mở rộng ứng dụng tra cứu luật theo yêu cầu mới và nhu cầu thay đổi.

#### ❖ Khả năng tương thích

Với Xamarin Forms, ứng dụng cho phép tương thích trên nhiều nền tảng di động, bao gồm cả iOS và Android. Giao diện và các chức năng cũng đã được nhóm tối ưu và thử trên nhiều loại thiết bị với kích thước, phiên bản hệ điều hành khác nhau. Điều này đảm bảo rằng ứng dụng có thể truy cập và sử dụng trên các thiết bị di động phổ biến hiện nay đồng thời tăng tính nhất quán trong trải nghiệm người dùng trên mọi nền tảng.

### 2. Kết luận

#### ❖ Kết quả đạt được của đồ án

Biết thêm về những công nghệ Xamarin, SQLite, áp dụng vào dự án làm những phần xây dựng ứng dụng di động đa nền tảng 1 cách hiệu quả. Công nghệ giúp lưu trữ dữ liệu, tương tác với hệ thống và phân cứng, và xây dựng giao diện người dùng đồng nhất trên nhiều nền tảng khác nhau. Điều này giúp bạn tiết kiệm thời gian và công sức trong quá trình phát triển và đảm bảo ứng dụng của bạn hoạt động một cách nhất quán trên các thiết bị di động khác nhau, người dùng đa nền tảng, dùng đa nền tảng.



Phần mềm đã đáp ứng các phần mục được đảm nhận. Có 1 số thiếu sót ở phần người dùng thông thường và người chưa đăng nhập không có sự khác nhau nên quyền lợi của người dùng đã đăng ký để sử dụng hệ thống sẽ không có. Trong tương lai chúng tôi sẽ phát triển và có thêm các chức năng cho phần người dùng đã đăng ký để sử dụng ứng dụng.

Kết luận, việc xây dựng hệ thống tra cứu pháp luật trên mobile mang lại nhiều lợi ích và giá trị cho người dùng trong việc truy cập và tra cứu thông tin pháp luật. Việc sử dụng công nghệ Xamarin Forms và ngôn ngữ lập trình C# giúp tạo ra một ứng dụng đa nền tảng, linh hoạt và tiện lợi.

Ứng dụng này cung cấp khả năng tra cứu thông tin chi tiết, đồng bộ hóa dữ liệu và thông báo cập nhật về pháp luật. Người dùng có thể dễ dàng tìm kiếm và truy cập thông tin pháp luật theo nhu cầu của mình từ mọi lúc, mọi nơi thông qua thiết bị di động.

Đối với người dùng như luật sư, nhân viên văn phòng, sinh viên và công chúng, ứng dụng này giúp tăng cường sự hiểu biết và chuyên nghiệp trong lĩnh vực pháp luật. Ngoài ra, việc cung cấp thông tin mới nhất về sửa đổi và thay đổi trong pháp luật giúp người dùng cập nhật thông tin và tuân thủ quy định hiện hành.

Về mặt khoa học, việc sử dụng công nghệ Xamarin Forms và C# trong xây dựng hệ thống tra cứu pháp luật trên mobile đóng góp vào lĩnh vực phát triển ứng dụng di động và mang lại các giải pháp kỹ thuật mới.

Tổng quát, việc xây dựng hệ thống tra cứu pháp luật trên mobile là một bước tiến quan trọng trong việc cung cấp thông tin pháp luật dễ dàng, tiện lợi và nhanh chóng cho người dùng, đồng thời nâng cao hiệu suất và chất lượng công việc trong lĩnh vực pháp luật.

#### ❖ Những hạn chế

Mặc dù việc xây dựng phần mềm hỗ trợ quá trình xử phạt vi phạm hành chính trong lĩnh vực lâm nghiệp trên nền tảng mobile mang lại nhiều lợi ích, nhưng cũng có một số hạn chế cần được lưu ý:

1. Sự phụ thuộc vào công nghệ di động: Việc phát triển phần mềm trên nền tảng di động yêu cầu kiến thức về công nghệ di động, đặc biệt là khi sử dụng Xamarin. Nhóm phát triển cần có kiến thức và kỹ năng về lập trình di động và nền tảng Xamarin để hiệu quả trong việc xây dựng và duy trì ứng dụng.
2. Đa dạng thiết bị và hệ điều hành: Trên nền tảng di động, có nhiều thiết bị và hệ điều hành khác nhau như Android, iOS và Windows. Việc đảm bảo tính tương thích và hiệu năng của ứng dụng trên tất cả các thiết bị và hệ điều hành có thể là một thách thức.

Cần kiểm thử kỹ lưỡng và đảm bảo rằng ứng dụng hoạt động một cách ổn định trên các nền tảng này.

3. Vấn đề bảo mật: Xử phạt vi phạm hành chính liên quan đến thông tin nhạy cảm và quan trọng. Đảm bảo an toàn và bảo mật dữ liệu trong quá trình thu thập, xử lý và lưu trữ là một yếu tố quan trọng. Cần có các biện pháp bảo mật chặt chẽ để ngăn chặn truy cập trái phép và lạm dụng thông tin.

4. Tương tác với hệ thống sẵn có: Trong một số trường hợp, phần mềm cần tương tác với hệ thống sẵn có như hệ thống quản lý lâm nghiệp hoặc hệ thống thông tin quản lý. Việc tích hợp và tương tác với các hệ thống này có thể gặp khó khăn và đòi hỏi kiến thức về quy trình và giao tiếp với hệ thống hiện có.

5. Sự chấp nhận từ người dùng: Để thành công, ứng dụng cần được người dùng chấp nhận và sử dụng. Việc thay đổi phương thức làm việc và sử dụng công nghệ mới có thể gặp sự khó khăn trong việc thay đổi tư duy và thói quen của người dùng hiện tại. Điều này đòi hỏi quá trình đào tạo và hỗ trợ tận tâm để đảm bảo sự chấp nhận và sử dụng hiệu quả của ứng dụng.

#### ❖ Hướng phát triển tiếp theo

1. Tích hợp tính năng định vị GPS: Để giúp quản lý việc xử phạt vi phạm hiệu quả, bạn có thể xem xét tích hợp tính năng định vị GPS vào ứng dụng. Điều này cho phép nhân viên hoặc cán bộ kiểm tra vi phạm xác định chính xác vị trí của vi phạm và thu thập thông tin liên quan một cách chính xác.

2. Cải thiện khả năng xử lý dữ liệu: Lĩnh vực lâm nghiệp có thể liên quan đến việc xử lý một lượng lớn dữ liệu, chẳng hạn như thông tin về vi phạm, thông tin cây trồng, vùng đất, và các hồ sơ liên quan khác. Bạn có thể tăng cường khả năng xử lý dữ liệu của ứng dụng để giúp quản lý và tìm kiếm thông tin một cách hiệu quả.

3. Mở rộng chức năng hỗ trợ quản lý: Ngoài việc hỗ trợ quá trình xử phạt, bạn có thể xem xét mở rộng chức năng của ứng dụng để hỗ trợ quản lý chung trong lĩnh vực lâm nghiệp. Ví dụ, bạn có thể bổ sung tính năng quản lý cây trồng, theo dõi sự phát triển của cây, quản lý tài nguyên rừng, và hỗ trợ quy trình giấy phép lâm nghiệp.

4. Tích hợp công nghệ AI và IoT: Công nghệ trí tuệ nhân tạo (AI) và Internet of Things (IoT) có thể cung cấp những tiện ích và giải pháp thông minh trong lĩnh vực lâm nghiệp. Bạn có thể xem xét tích hợp công nghệ AI để phát hiện tự động vi phạm và gợi ý các biện pháp xử lý. Ngoài ra, việc kết nối với các thiết bị IoT, chẳng hạn như cảm biến đo mức nước, cũng có thể cung cấp thông tin quan trọng để hỗ trợ quyết định trong quá trình xử phạt.

5. Tích hợp công nghệ Blockchain: Công nghệ Blockchain có thể cung cấp tính toàn vẹn và minh bạch cho dữ liệu trong quá trình xử phạt. Bạn có thể xem xét tích hợp

công nghệ Blockchain để ghi lại thông tin vi phạm, quy trình xử phạt, và các thay đổi liên quan khác để đảm bảo tính chính xác và không thể sửa đổi của dữ liệu.

6. Nâng cao giao diện người dùng và trải nghiệm người dùng: Tạo ra một giao diện người dùng trực quan và dễ sử dụng là rất quan trọng để đạt được sự chấp nhận và sự hài lòng từ người dùng. Bạn có thể xem xét cải thiện giao diện người dùng, tối ưu hóa trải nghiệm người dùng và tăng cường khả năng tương tác trong ứng dụng của mình.