|  |  |
| --- | --- |
|  |  |
| CHU VĂN QUÝ | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
|  |
| ĐA, KLTN ĐẠI HỌC  NGÀNH CÔNG NGHỆ THÔNG TIN |
|  |
| **XÂY DỰNG ỨNG DỤNG ĐẶT LỊCH KHÁM ONLINE CHO THÚ CƯNG** |
|  |
|  |
| **CBHD : ThS. Đoàn Văn Trung** |
| NGÀNH CÔNG NGHỆ THÔNG TIN | **Sinh viên : Chu Văn Quý** |
| **Mã số sinh viên : 2021606690** |
|  |
|  |
|  |
| Hà Nội – Năm 2025 |

|  |
| --- |
| **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
|  |
| ĐA, KLTN ĐẠI HỌC  NGÀNH CÔNG NGHỆ THÔNG TIN |
|  |
| |  | | --- | |  | | **XÂY DỰNG ỨNG DỤNG ĐẶT LỊCH KHÁM ONLINE CHO THÚ CƯNG** | |
|  |
|  |
| **CBHD : ThS. Đoàn Văn Trung** |
| **Sinh viên : Chu Văn Quý** |
| **Mã số sinh viên : 2021606690** |
|  |
|  |
| Hà Nội – Năm 2025 |

# LỜI CẢM ƠN

Trước tiên với tình cảm sâu sắc và chân thành nhất, cho phép chúng em được bày tỏ lòng biết ơn đến tất cả thầy cô và nhà trường đã tạo điều kiện hỗ trợ, giúp đỡ chúng em trong suốt quá trình học tập và nghiên cứu đề tài này. Trong suốt thời gian từ khi bắt đầu học tập tại trường đến nay, chúng em đã nhận được rất nhiều sự quan tâm, giúp đỡ của quý Thầy Cô và bạn bè.

Với lòng biết ơn sâu sắc nhất, chúng em xin gửi đến quý Thầy Cô ở khoa Công Nghệ Thông Tin đã truyền đạt vốn kiến thức quý báu cho chúng em trong suốt thời gian học tập tại trường. Nhờ có những lời hướng dẫn, dạy bảo của các thầy cô nên đề tài nghiên cứu của chúng em mới có thể hoàn thiện tốt đẹp.

Một lần nữa, chúng em xin chân thành cảm ơn thầy **Đoàn Văn Trung**– người đã trực tiếp giúp đỡ, quan tâm, hướng dẫn chúng em hoàn thành tốt bài báo cáo này trong thời gian qua.

Bài báo cáo thực hiện trong khoảng thời gian không nhiều bước đầu đi vào thực tế của chúng em còn hạn chế và còn nhiều bỡ ngỡ nên không tránh khỏi những thiếu sót. Chúng em rất mong nhận được những ý kiến đóng góp quý báu của quý Thầy Cô để kiến thức của chúng em trong lĩnh vực này được hoàn thiện hơn đồng thời có điều kiện bổ sung, nâng cao kiến thức của mình.

Chúng em xin chân thành cảm ơn!

Sinh viên thực hiện

Chu Văn Quý

# LỜI MỞ ĐẦU

Sự phát triển của công nghệ thông tin đã thay đổi cách chúng ta quản lý và điều hành nhiều khía cạnh trong cuộc sống, bao gồm cả lĩnh vực chăm sóc sức khỏe cho thú cưng. Ngày nay, việc đặt lịch khám cho thú cưng đã trở nên dễ dàng và tiện lợi hơn bao giờ hết, nhờ vào sự phát triển của các ứng dụng đặt lịch trực tuyến.

Trước đây, khi chưa có ứng dụng này, việc đặt lịch khám cho thú cưng thường phải qua điện thoại hoặc đến trực tiếp tại phòng khám, điều này thường gây ra sự bất tiện cho người nuôi thú cưng. Tuy nhiên, với sự xuất hiện của ứng dụng đặt lịch khám cho thú cưng, người nuôi thú cưng có thể dễ dàng chọn lịch hẹn theo thời gian phù hợp với họ, thậm chí có thể nhận được thông báo nhắc nhở trước ngày hẹn.

Ứng dụng này không chỉ mang lại sự tiện lợi cho người nuôi thú cưng mà còn giúp các phòng khám thú cưng quản lý lịch hẹn của họ một cách hiệu quả hơn. Họ có thể theo dõi số lượng lịch hẹn, xác định thời gian sẵn sàng phục vụ và quản lý thông tin về các bệnh nhân thú cưng của họ một cách dễ dàng.

Ứng dụng đặt lịch khám cho thú cưng không chỉ giúp tiết kiệm thời gian mà còn cung cấp cho người nuôi thú cưng và các phòng khám thú cưng sự thuận tiện và hiệu quả. Nó là một bước tiến quan trọng trong việc cung cấp dịch vụ chăm sóc sức khỏe tốt hơn cho các thành viên bốn chân của gia đình của chúng ta.

Mục Đích Nghiên Cứu:

Nghiên cứu này nhằm nghiên cứu và phát triển một ứng dụng đặt lịch khám thú cưng hiệu quả và tiện lợi, với mục tiêu cung cấp giải pháp tối ưu cho việc quản lý và sắp xếp cuộc hẹn khám cho thú cưng. Mục đích chính của nghiên cứu này là:

Tối ưu hóa Quản Lý Lịch: Nghiên cứu nhằm xây dựng một hệ thống quản lý lịch mạnh mẽ và thông minh, giúp bác sĩ thú y, nhân viên và chủ thú cưng có thể dễ dàng xem, quản lý và cập nhật lịch khám.

Tăng Cường Tiện Ích cho Khách Hàng: Tạo ra một giao diện dễ sử dụng cho người dùng cuối, cho phép họ dễ dàng đặt lịch khám thú cưng, chọn bác sĩ thú y, và nhận thông tin chi tiết về cuộc hẹn, cũng như nhắc nhở và thông báo quan trọng liên quan đến sức khỏe của thú cưng.

Nâng Cao Hiệu Suất Công Việc cho Bác Sĩ Thú Y: Ứng dụng cần hỗ trợ bác sĩ thú y trong việc quản lý lịch làm việc của họ, cung cấp thông tin chi tiết về các ca khám, lịch trình công việc và dữ liệu sức khỏe của các bệnh nhân thú cưng.

Tối ưu hóa Chăm Sóc Sức Khỏe Thú Cưng: Đảm bảo rằng thông tin về sức khỏe của thú cưng được lưu trữ một cách an toàn, dễ dàng truy cập và cập nhật. Ứng dụng có thể cung cấp lịch sử khám bệnh, thông tin về tiêm chủng và lời khuyên về chăm sóc thú cưng.

Tích hợp Công Nghệ: Sử dụng công nghệ hiện đại như trí tuệ nhân tạo (AI), Internet of Things (IoT), và dữ liệu lớn (Big Data) để cải thiện khả năng dự đoán, phân tích dữ liệu, và cung cấp dịch vụ cá nhân hóa.

Nâng Cao Hiệu Quả Kinh Doanh: Ứng dụng đặt lịch còn có thể giúp tăng doanh thu và lợi nhuận của các phòng khám thú y thông qua quảng cáo dịch vụ, thu thập đánh giá từ khách hàng, và tối ưu hóa quy trình làm việc.

Đảm Bảo An Toàn Dữ Liệu: Đặt mục tiêu đảm bảo an toàn và bảo mật dữ liệu cá nhân của khách hàng và thông tin về thú cưng, tuân thủ tất cả các quy định về quyền riêng tư.

Phương pháp nghiên cứu cho dự án ứng dụng đặt lịch khám thú cưng có thể bao gồm các bước và phương pháp sau đây:

1. Nghiên cứu Thư viện:

Tìm hiểu về các ứng dụng và dịch vụ tương tự đã có trên thị trường để hiểu về các tính năng và quy trình hoạt động hiện tại.

Tìm hiểu về các công nghệ mới nhất và tiêu chuẩn trong ngành thú y và phát triển ứng dụng di động.

2. Xác định Yêu cầu và Chức năng:

Tổ chức cuộc họp với các bác sĩ thú y và chủ thú cưng để xác định các yêu cầu cụ thể của họ đối với ứng dụng.

Xác định các tính năng quan trọng như đặt lịch, quản lý thông tin sức khỏe thú cưng, thông báo, tích hợp thanh toán, và giao diện người dùng.

3. Thiết kế Giao diện Người Dùng (UI) và Trải nghiệm Người Dùng (UX):

Phát triển bản thiết kế giao diện người dùng dựa trên yêu cầu và phản hồi từ người dùng mục tiêu.

Sử dụng các phương pháp thử nghiệm người dùng để cải thiện trải nghiệm người dùng và tối ưu hóa giao diện.

4. Phát triển Ứng dụng:

Sử dụng mô hình phát triển phần mềm như Agile hoặc Scrum để phát triển ứng dụng theo từng giai đoạn.

Sử dụng công nghệ phù hợp để xây dựng ứng dụng trên nền tảng di động và web.

5. Kiểm thử và Chấp nhận:

Tiến hành kiểm thử tự động và thủ công để đảm bảo tính ổn định và tính năng của ứng dụng.

Đưa ứng dụng vào giai đoạn chấp nhận, cho phép người dùng cuối và các chuyên gia thú y thử nghiệm và đánh giá.

6. Tích hợp Công nghệ:

Tích hợp các công nghệ như trí tuệ nhân tạo (AI) để dự đoán lịch hẹn thú cưng hoặc cung cấp gợi ý chăm sóc.

Kết nối với hệ thống thông tin y tế thú y để cập nhật thông tin về sức khỏe thú cưng.

7. Triển khai và Theo dõi:

Triển khai ứng dụng trên nền tảng di động và web, và cung cấp hỗ trợ kỹ thuật cho người dùng.

Theo dõi hiệu suất ứng dụng, thu thập dữ liệu về việc sử dụng và phản hồi từ người dùng để thực hiện các cải tiến và sửa lỗi.

8. Đánh giá Kết quả:

Đánh giá mức độ thỏa mãn của người dùng và bác sĩ thú y với ứng dụng.

Xem xét các chỉ số hiệu suất như tỷ lệ đặt lịch thành công, sự tương tác của người dùng, và tăng trưởng người dùng.

9. Cải Tiến và Phát Triển Tiếp:

Dựa trên phản hồi và dữ liệu thu thập, thực hiện các cải tiến và phát triển tiếp ứng dụng để đáp ứng nhu cầu ngày càng biến đổi của thị trường.

10. Bảo Mật và Quản lý Dữ liệu:

Đảm bảo bảo mật thông tin cá nhân và dữ liệu về sức khỏe thú cưng thông qua mã hóa và quản lý quyền truy cập. Sinh viên thực hiện

# CHƯƠNG 1: TỔNG QUAN

## 1.1 Tổng quan về đề tài được nghiên cứu

### Khảo sát thực tế

Trong thời đại mà công nghệ ngày càng phát triển, việc cải thiện và tối ưu hóa các phần của ngành chăm sóc sức khỏe cho thú cưng trở nên quan trọng hơn bao giờ hết. Hiện nay, việc đặt lịch khám cho thú cưng đã trở nên quan trọng và tiện lợi hơn bao giờ hết, nhất là đối với những người nuôi thú cưng trong gia đình.

Tuy nhiên, vẫn còn nhiều thách thức trong việc quản lý lịch hẹn và dịch vụ chăm sóc thú cưng một cách hiệu quả. Chưa có nhiều ứng dụng di động dành riêng cho việc đặt lịch khám và quản lý sức khỏe của thú cưng, dẫn đến tình trạng phải gọi điện thoại hoặc đến trực tiếp tại phòng khám, điều này thường gây ra sự bất tiện cho người nuôi thú cưng.

Chúng tôi đã nhận thức được những khó khăn này và đã phát triển một ứng dụng đặt lịch khám cho thú cưng dành riêng cho các "bố mẹ" thú cưng. Ứng dụng của chúng tôi giúp bạn dễ dàng chọn lịch hẹn theo thời gian phù hợp với bạn và thú cưng, và bạn có thể nhận được thông báo nhắc nhở trước ngày hẹn.

Ngoài ra, ứng dụng này còn giúp các phòng khám thú cưng quản lý lịch hẹn của họ một cách hiệu quả hơn, cung cấp thông tin về sức khỏe của thú cưng và giúp bạn theo dõi tiến trình chăm sóc sức khỏe của thú cưng một cách dễ dàng.

Ứng dụng đặt lịch khám cho thú cưng của chúng tôi không chỉ giúp tiết kiệm thời gian mà còn mang lại sự thuận tiện và hiệu quả trong việc quản lý sức khỏe cho các thành viên bốn chân của gia đình

#### Ưu điểm

Không cần đầu tư nhiều kinh phí vào các thiết bị thông minh hoặc các phần mềm cho việc quản lý thông tin.

#### Nhược điểm

Khi lưu trữ thông tin về thú cưng, lịch hẹn, dịch vụ chăm sóc, và các thông tin liên quan, chúng ta thường phải dựa vào phương thức truyền thống như sổ sách và giấy tờ, đòi hỏi sự tốn kém về thời gian và công sức, và thường cần nhiều nhân viên để quản lý được việc này một cách hiệu quả.

Khi cần truy xuất thông tin về thú cưng, lịch hẹn, hoặc lịch sử sức khỏe, việc tìm kiếm và tra cứu trên giấy tờ trở nên khó khăn, đặc biệt là khi có một lượng lớn thông tin. Điều này dẫn đến tình trạng mất nhiều thời gian và công sức trong việc quản lý và truy cập thông tin quan trọng.

## 1.2 Nhiệm vụ đồ án

Từ các vấn đề trên, chúng ta cần phải xây dựng một phần mềm, một hệ thống mới có yêu cầu kỹ thuật, quản lý chuyên nghiệp hơn, giải quyết được các khuyết điểm của hệ thống quản lý cũ.

### Đối tượng và phạm vi của ứng dụng

#### Đối tượng

Hiện nay, việc chăm sóc và quản lý cho thú cưng đã trở thành một phần quan trọng trong cuộc sống hàng ngày của mọi người. Chúng ta biết rằng việc đảm bảo sức khỏe và hạnh phúc cho các thành viên lông lốc của gia đình không phải lúc nào cũng dễ dàng.

Trong thời đại số hóa ngày nay, việc quản lý thông tin và lịch hẹn cho thú cưng cũng đòi hỏi sự tiện lợi và hiệu quả. Chúng tôi đã tạo ra một ứng dụng đặt lịch khám cho thú cưng mà chủ nuôi thú cưng có thể dễ dàng sử dụng. Ứng dụng của chúng tôi không chỉ giúp bạn quản lý lịch hẹn cho thú cưng một cách thuận tiện, mà còn mang đến nhiều tính năng hữu ích khác để bạn có thể chăm sóc thú cưng của mình một cách tốt nhất.

#### Mục Tiêu

Đồ án Android này đã được xây dựng dành riêng cho những người yêu thú cưng. Với đam mê và tình yêu đối với thú cưng của mình, việc quản lý thông tin và lịch hẹn cho thú cưng là một phần quan trọng của cuộc sống hàng ngày.

#### Lợi ích mang lại

* Tiết kiệm nguồn nhân lực, thời gian, tiền bạc, và công sức trong quản lý.
* Cung cấp thống kê nhanh chóng, giúp người quản lý dễ dàng nắm bắt thông tin, tính toán, và xác định chiến lược phát triển cho các chi nhánh, cơ sở hoặc địa điểm kinh doanh.
* Tạo sự tiện lợi cho người sử dụng, tăng hiệu suất làm việc.
* Tự động hóa các tác vụ quản lý, tạo sự chuyên nghiệp trong việc điều hành kinh doanh của quán ăn.

#### Các bước thực hiện đồ án

* Nghiên cứu, khảo sát thực tế về tình hình và cơ cấu hoạt động của các cơ sở kinh doanh hiện nay.
* Lập ra các kế hoạch cụ thể để phát triển hệ thống.
* Phân tích yêu cầu của phần mềm
* Thiết kế CSDL
* Thiết kế giao diện
* Hiện thực phần mềm bằng Android Studio
* Kiểm thử

# **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

## Các khái niệm và cơ chế hoạt động

### Công nghệ sử dụng

#### Hệ điều hành Android

Android Studio là IDE (môi trường phát triển tích hợp) chính thức để phát triển ứng dụng Android và nó dựa trên phần mềm  Intellij IDEA của JetBrain.

Vì những đặc điểm đặc trưng về cấu tạo của Android Studio nên trong quá trình soạn thảo mã và công cụ phát triển mạnh mẽ của IntelliJ thì Android Studio sẽ cung cấp những tính năng đã được nâng cao để làm tăng hiệu suất của bạn trong việc xây dựng ứng dụng Android.

Android Studio cung cấp nhiều tính năng tuyệt vời giúp năng cao hiệu suất khi xây dựng ứng dụng Android, chẳng hạn như:

* Một môi trường hỗn hợp, nơi một người có thể phát triển cho tất cả các thiết bị Android.
* Áp dụng thay đổi để đẩy code và thay đổi tài nguyên cho ứng dụng đang chạy mà không cần khởi động lại.
* Trình giả lập nhanh và nhiều tính năng.
* Tích hợp mẫu code và Github để hỗ trợ bạn phát triển các tính năng ứng dụng phổ biến và nhập code mẫu.
* Các công cụ và framework thử nghiệm mở rộng.
* Hỗ trợ C++ và NDK.
* Hỗ trợ tích hợp cho Google Cloud Platform, giúp dễ dàng tích hợp Google Cloud Messaging và App Engine…
* Cung cấp các công cụ GUI giúp đơn giản hóa các phần ít thú vị hơn trong quá trình phát triển ứng dụng.
* Dễ dàng tích hợp với cơ sở dữ liệu thời gian thực *firebase*.
* **Ưu điểm của Android Studio:**
* Tích hợp nhiều công cụ mạnh mẽ hỗ trợ nhà phát triển bao gồm: trình biên dịch, trình gỡ lỗi, quản lý phiên bản, xây dựng tự động. Điều này giúp tăng năng suất và cải thiện chất lượng của mã nguồn.
* Cộng đồng hỗ trợ: Android Studio có một cộng đồng lớn và nhiều nguồn tài nguyên trực tuyến, bao gồm tài liệu, diễn đàn và mã nguồn mở.
* Dễ sử dụng: Android Studio có một giao diện người dùng thân thiện và trực quan, giúp người dùng dễ dàng thao tác và điều hướng trong quá trình phát triển.
* **Nhược điểm của Android Studio:**
* Yêu cầu cấu hình máy tính tương đối cao và tốn tài nguyên hệ thống: Là công cụ hỗ trợ tích hợp tất cả nên dữ liệu phải phát triển tối ưu nhất. Lượng lớn dữ liệu chiếm nhiều không gian bộ nhớ máy tính của bạn.
* Có thể kiểm tra được cash hoạt động của app thông qua giả lập của Android studio. Điều này làm đơ máy, lag, nóng máy tính và gây tiêu tốn nhiều pin.
* Khó sử dụng với người mới bắt đầu : Android studio là công cụ có nhiều tính năng và công cụ phức tạp đòi hỏi người dùng phải có thêm kiến thức chuyên sâu về lập trình android
* Lỗi phát sinh thường xuyên : do tính năng phức tạp của android studio nên nhiều lỗi phát sinh thường xuyên nhất là khi cập nhật phiên bản mới
* Không hỗ trợ tốt cho các dự án đa nền tảng

#### Android Studio IDE

**Android Studio** là IDE chính thức được sử dụng trong phát triển ứng dụng Android dựa trên IntelliJ IDEA.

Chức năng chính của Android Studio là cung cấp các giao diện giúp người dùng có thể tạo các ứng dụng và xử lý các công cụ file phức tạp sau hậu trường. Ngôn ngữ lập trình được sử dụng trong Android Studio là Java và nó sẽ được cài đặt sẵn trên thiết bị.

Khi sử dụng Android Studio thì bạn chỉ cần viết, chỉnh sửa và lưu trữ chúng trên các dự án của mình và các file nằm trong dự án đó. Đồng thời, Android Studio còn cung cấp quyền truy cập vào Android SDK.

Ngoài ra, bạn có thể xem Android Studio là đuôi cho code Java cho phép nó chạy trơn tru trên các thiết bị Android rồi tận dụng được lợi thế của phần cứng gốc. Bạn chỉ cần sử dụng ngôn ngữ lập trình Java để có thể viết chương trình, khi đó Android SDK sẽ có nhiệm vụ kết nối các phần này lại với nhau.

#### Ngôn ngữ lập trình Java

**Java** được biết đến là ngôn ngữ lập trình bậc cao, hướng đối tượng và giúp bảo mật mạnh mẽ, và còn được định nghĩa là một Platform. Java được phát triển bởi Sun Microsystems, do James Gosling khởi xướng và ra mắt năm 1995. Java hoạt động trên rất nhiều nền tảng như Windows, Mac và các phiên bản khác nhau của UNIX..

Tương tự C++, hướng đối tượng hoàn toàn. Trong quá trình tạo ra một ngôn ngữ mới phục vụ cho mục đích chạy được trên nhiều nền tảng, các kỹ sư của Sun MicroSystem muốn tạo ra một ngôn ngữ dễ học và quen thuộc với đa số người lập trình. Vì vậy họ đã sử dụng lại các cú pháp của C và C++.

Tuy nhiên, trong Java thao tác với con trỏ bị lược bỏ nhằm đảo bảo tính an toàn và dễ sử dụng hơn. Các thao tác overload, goto hay các cấu trúc như struct và union cũng được loại bỏ khỏi Java.

Tính đến năm 2019 Java là một trong những ngôn ngữ lập trình phổ biến nhất được sử dụng theo GitHub, đặc biệt cho các ứng dụng web máy khách-máy chủ, với 9 triệu nhà phát triển đã được báo cáo.

#### SQLite

SQLite là hệ quả trị cơ sở dữ liệu (DBMS) quan hệ tương tự như MySQL... Đặc điểm nổi bật của SQLite so với các DBMS khác là gọn, nhẹ, đơn giản, đặt biệt không cần mô hình server-client, không cần cài đặt, cấu hình hay khởi động nên không có khái niệm user, password hay quyền hạn trong SQLite Database. Dữ liệu cũng được lưu ở một file duy nhất.

SQLite thường không được sử dụng với các hệ thống lớn nhưng với những hệ thống ở quy mô vừa và nhỏ thì SQLite không thua các DBMS khác về chức năng hay tốc độ. Vì không cần cài đặt hay cấu hình nên SQLite được sử dụng nhiều trong việc phát triển, thử nghiệm… vì tránh được những rắc rối trong quá trình cài đặt.

**Tính năng của SQLite:**

* Giao dịch trong SQLite tuân thủ theo nguyên tắc (ACID) ngay cả sau hi hệ thống treo và mất điện.
* Không cấu hình
* Không cần thiết lập hoặc quản trị
* SQLite hỗ trợ với đầy đủ tính năng với các khả năng nâng cao như các chỉ mục 1 phần, các chỉ mục về các biểu thức, JSON và các biểu thức bảng chung.
* Một sở dữ liệu hoàn chỉnh được lưu trữ trong một tệp đa nền tảng duy nhất. Phù hợp với sử dụng dưới dạng định dạng tệp ứng dụng
* Hỗ trợ các cơ sở dữ liệu có kích thước terabyte và các chuỗi có kích thước gigabyte.
* API
* Đơn giản dễ sử dụng
* Nhanh: Trong một số trường hợp, SQLite nhanh hơn hệ thống tệp tin trực tiếp I/O.
* Được viết bằng ANSI-C.
* Bindings cho hàng chục ngôn ngữ khác có sẵn 1 cách riêng biệt.
* Mã nguồn đầy, nguồn mở đủ có thể kiểm tra nhánh 100%.
* Nền tảng đa nền tảng
* SQLite là có sẵn trên Android, \*BSD, iOS, Linux, Mac, Solaris, Windows,.. Dễ dàng dịch chuyển sang các hệ thống khác.

**Ứng dụng chủ yếu của SQLite:**

* Cơ sở dữ liệu cho Internet Of Things.
* SQLite là lựa chọn phổ biến cho các công cụ cơ sở dữ liệu trong điện thoại di động, PDA, máy nghe nhạc mp3, hộp set-top, và các tiện ích điện tử khác.
* Định dạng tệp ứng dụng.
* Thay vì sử dụng fopen() để viết XML, JSON, CSV hoặc một số định dạng động quyền vào các tệp đĩa được ứng dụng của bạn sử dụng, hãy sử dụng SQLite.
* Cơ sở dữ liệu cho web.
* Bởi vì SQLite không yêu cầu cấu hình và lưu trữ thông tin trong các tệp đĩa thông thường nên SQLite là lựa chọn phổ biến làm cơ sở dữ liệu để quay lại các trang web vừa và nhỏ.
* Stand-in cho một doanh nghiệp RDBMS:SQLite được sử dụng như một RDBMS doanh nghiệp cho mục đích trình diễn hoặc để thử nghiệm vì SQLite nhanh và không yêu cầu thiết lập.Các thư viện và công cụ hỗ trợ

#### MPAndroidChart

**MPAndroidChart** là một thư viện biểu đồ mạnh mẽ và dễ sử dụng cho Android. Nó support việc vẽ biểu đồ, chia tỷ lệ và xử lí animations trong Chart.Luôn dùng trong việc thống kê biểu đồ chia cột doanh thu và tổng số lượng,..

com.github.PhilJay:MPAndroidChart:v3.0.2

<https://github.com/PhilJay/MPAndroidChart>

#### CircleImageView

**CircleImageView** là một thự viện tạo thành một hình ảnh ở khuôn hình tròn nhằm để tạo hình thiết kế cho đẹp mắt hơn.

de.hdodenhof :circleimageview:3.1.0

<https://github.com/hdodenhof/CircleImageView>

## Xác định yêu cầu

### Yêu cầu hệ thống

Ứng dụng dễ dàng đặt lịch hẹn cho thú cưng, theo dõi lịch sử sức khỏe, và quản lý thông tin quan trọng về thú cưng một cách tiện lợi, cung cấp các vai trò khác nhau cho người dùng, bao gồm Quản lý (Admin)

Với vai trò Admin, bạn có quyền toàn quyền trong việc quản lý các chức năng của ứng dụng. Bạn có thể thêm và xóa tài khoản của các Nhân viên và xem thống kê chi tiết về lịch khám và sức khỏe của thú cưng và khách hàng sử dụng

### Yêu cầu phi chức năng

- Điện thoại sử dụng hệ điều hành Android phiên bản 4.0 trở lên

- Có kết nối Internet

- Quá trình đăng nhập diễn ra nhanh chóng

- Database có khả năng lưu trữ thông tin của 10000 người dùng.

- Tài khoản được đăng ký đúng chính xác nhất

- Giao diện đẹp mắt

# CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Bảng mô tả đối tượng – Usecase

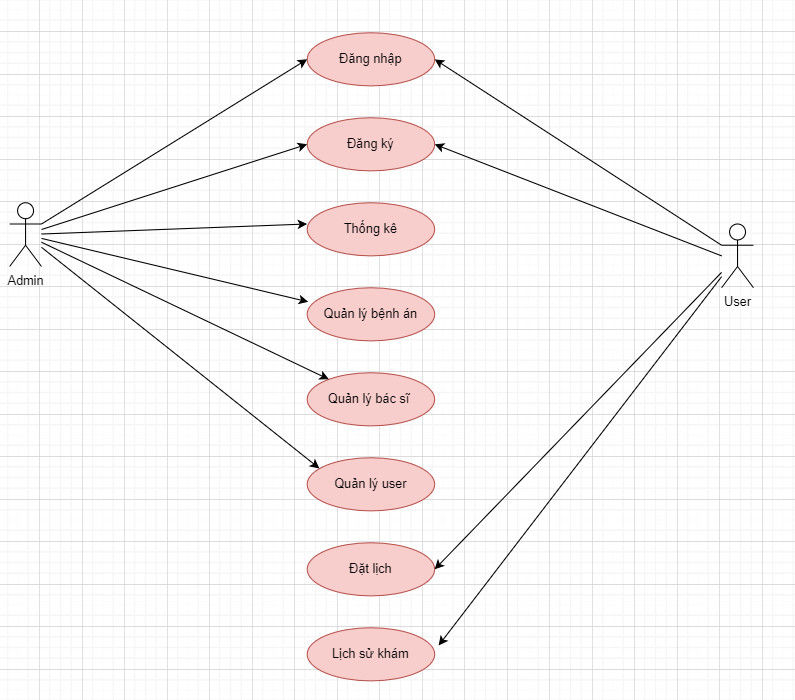
### Mô tả Actor

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên Actor** | **Mô tả** |
| 1 | Admin | Là quản trị viên của hệ thống, có mọi quyền và thực hiện mọi chức năng |
| 2 | Khách hàng | Là khách hàng sử dụng |

### Mô tả Usecase

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên Usecase** | **Mô tả** |
| 1 | Quản lý thú cưng | Khách hàng có thể thêm thú cưng của mình vào hồ sơ |
| 2 | Đặt lịch khám | Khách hàng đặt lịch khám bệnh cho thú cưng của mình |
| 3 | Lịch sử khám | Khách hàng cõ thể theo dõI tình trạng và lịch sử khám cho thú cưng |
| 4 | Quản lý thống kê | Actor admin là người có quyền vận hành quản lý thống kê bằng cách xem danh sách hóa đơn và tổng doanh thu |
| 5 | Quản lý bệnh án thú cưng | Actor admin quản lý trạng tháI và tạo bệnh án cho thú cưng của khách hàng |

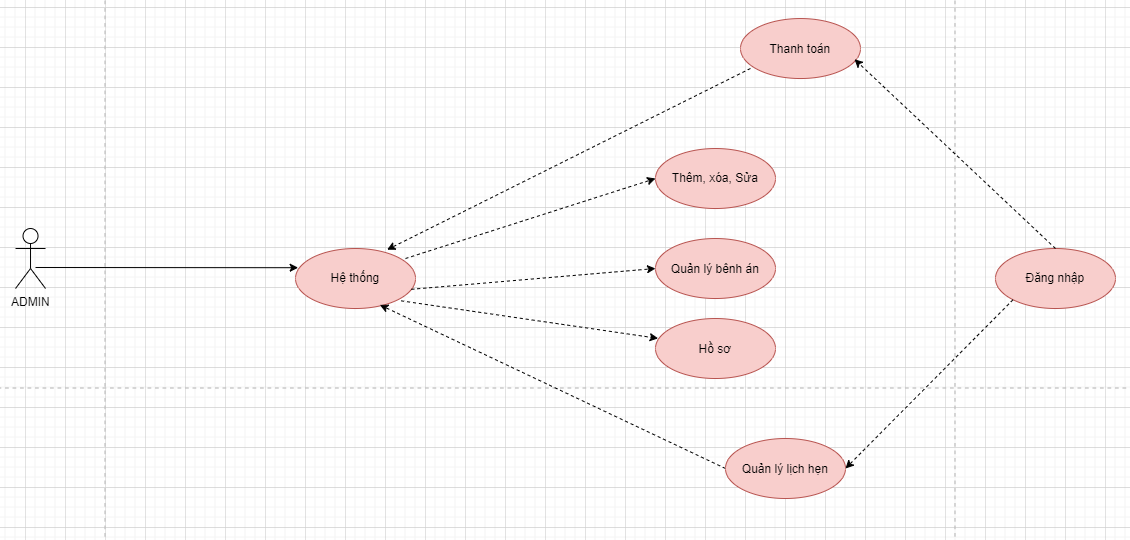
## Sơ đồ Usecase



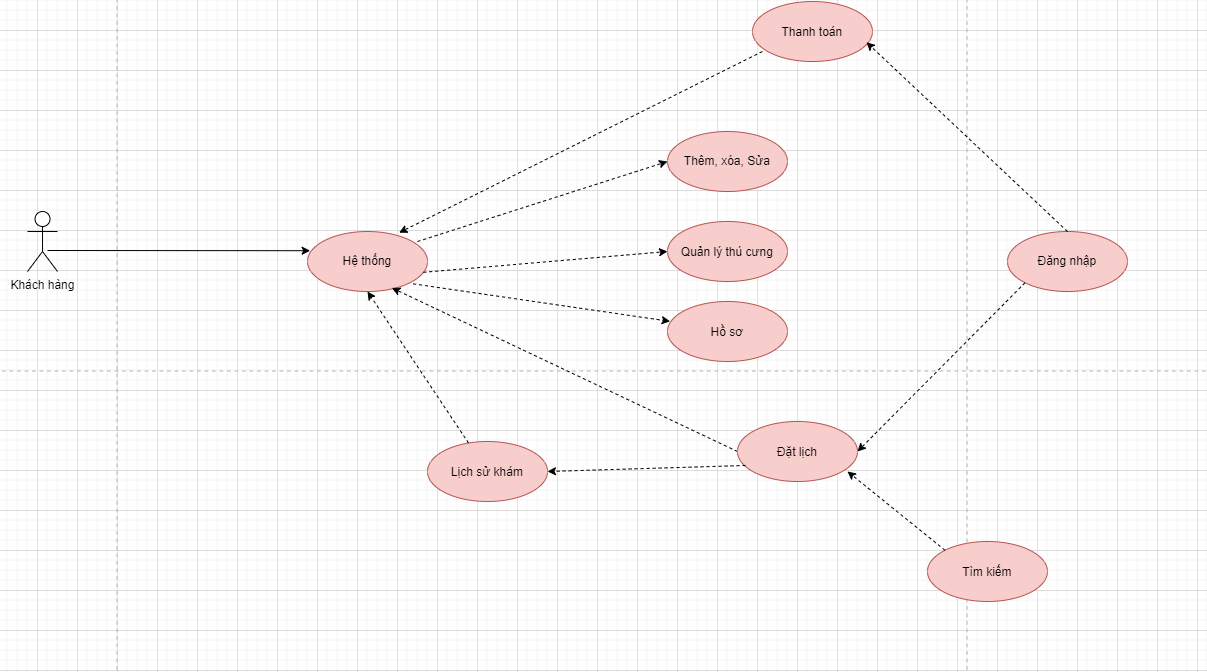
Sơ đồ tổng quan về các chức năng của các Actor:

* Đối với Admin thì Actor này có thể quản lý được tất cả các chức năng như: Thống kê doanh thu, Lịch đang chờ gần nhất, Quản lý bác sĩ, Quản lý trạng thái đặt lịch, Tạo bệnh án, tạo hóa đơn, Quản lý người dùng
* Đối với khách hàng: Đặt lịch khám online (điền phiếu khám, chọn bác sĩ, chọn dịch vụ), Thông báo đặt thành công, Gọi cấp cứu, Nhắn tin tư vấn, Lịch sử khám, Xem thông tin bác sĩ, Tìm kiếm, Trạng thái lịch hẹn: lịch chờ, xác nhận, hoàn thành, hủy, Hóa đơn, Thông tin cá nhân, Thêm, sửa, xóa thú cưng, Đổi mật khẩu

## Sơ đồ phân rã chức năng

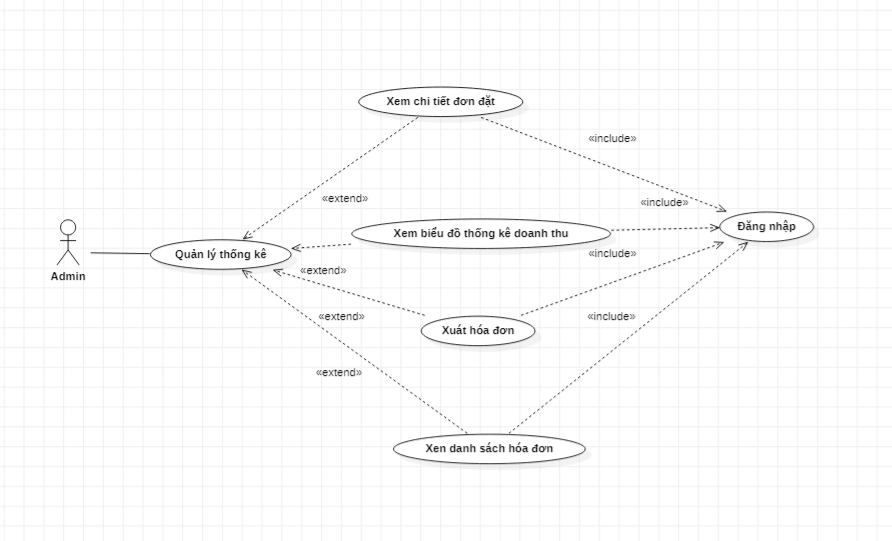


Về chức năng quản lý bàn hệ thống sẽ phân quyền cho khách hàng



Với quản lý đặt lịch khám cho thú cưng, cả 2 Actor này đều có thể thực hiện các chức năng như: tạo, sửa đổi, xóa các lịch hẹn đặt khám cho thú cưng. Hệ thống cũng cho phép quản lý các thông tin liên quan đến việc đặt lịch khám, bao gồm thời gian, ngày, và các dịch vụ liên quan.

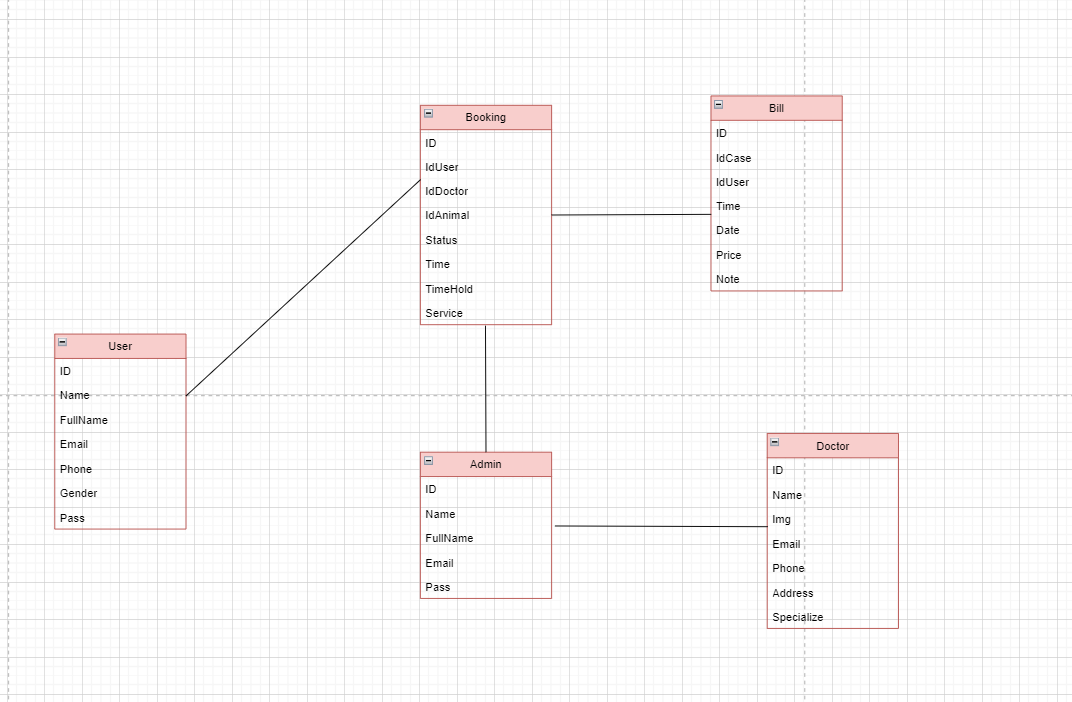
Điều này giúp cung cấp tính linh hoạt cho tất cả các vai trò để quản lý và theo dõi lịch hẹn khám cho thú cưng một cách thuận tiện và hiệu quả."



Ở chức năng quản lý thông kê này thì chỉ có Actor Admin mới được thao tác ở trên nó. Để Admin có thể quản lý thống kê như:

* Xem chi tiết đơn hàng
* Xem biểu đồ thống kê doanh thu
* Xuất hóa đơn
* Xem danh sách hóa đơn

## Sơ đồ Class Diagram



## Sơ đồ Thực thể

# CHƯƠNG 4: XÂY DỰNG HỆ THỐNG VÀ CÁC CHỨC NĂNG CHO CHƯƠNG TRÌNH

## Xây dựng hệ thống

### Môi trường phát triển

- Chương trình được phát triển trên môi trường Android Studio IDE.

- Ngôn ngữ Java.

- SQLite

### Công nghệ sử dụng

**SQLite:** một cơ sở dữ liệu SQL mã nguồn mở, nó lưu trữ dữ liệu vào một tập tin văn bản trên một thiết bị

## Cấu trúc chương trình

### DTO (Data Transfer Object)

Chứa các thực thể của dữ liệu phạm vi của thuộc tính Object, kèm theo các phương Get/Set.

Việt hoá lại các usecase, sao cho đồng bộ, viết các usecase chính, có thể chụp các màn hình.

|  |  |  |
| --- | --- | --- |
| **Use case: UC\_01\_Đăng ký** | | |
| Mục đích: | User đăng ký tài khoản và đăng nhập vào hệ thống. | |
| Mô tả: | Use case này cho phép người dùng đăng đăng ký tài khoản. | |
| Tác nhân: | Khách hàng | |
| Điều kiện đặc biệt: | Không có. | |
| Tiền điều kiện: | Không có. | |
| Luồng cơ bản (Basic flows): | **Tác nhân** | **Hệ thống** |
| 1. . Use case bắt đầu khi người dùng click vào nút “Đăng ký” trên màn hình | 2. Hệ thống sẽ chuyển sang màn hình “Đăng ký” và Form đăng ký. |
| 3. Người dùng nhập Họ tên, Email, Số điện thoại, Giới tính, Mật khẩu, Xác nhận mật khẩu sau đó click vào nút “Đăng ký”. | 4. Hệ thống kiểm tra thông tin rồi lưu thông tin tài khoản vào bảng Users trong cơ sở dữ liệu sau đó hiển thị giao diện trang chủ.  Use case kết thúc. |
| Luồng rẽ nhánh (Alternative flows): | **Tác nhân** | **Hệ thống** |
| 1. Tại bước 2 trong luồng cơ bản nếu người dùng không nhập Họ tên, Email, Số điện thoại, Giới tính, Mật khẩu, Xác nhận mật khẩu, kiểu dữ liệu không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi. Người dùng cần nhập đúng thông tin để tiếp tục. | 2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc. |
| Hậu điều kiện: | Không có. | |
| Điểm mở rộng: | Không có. | |



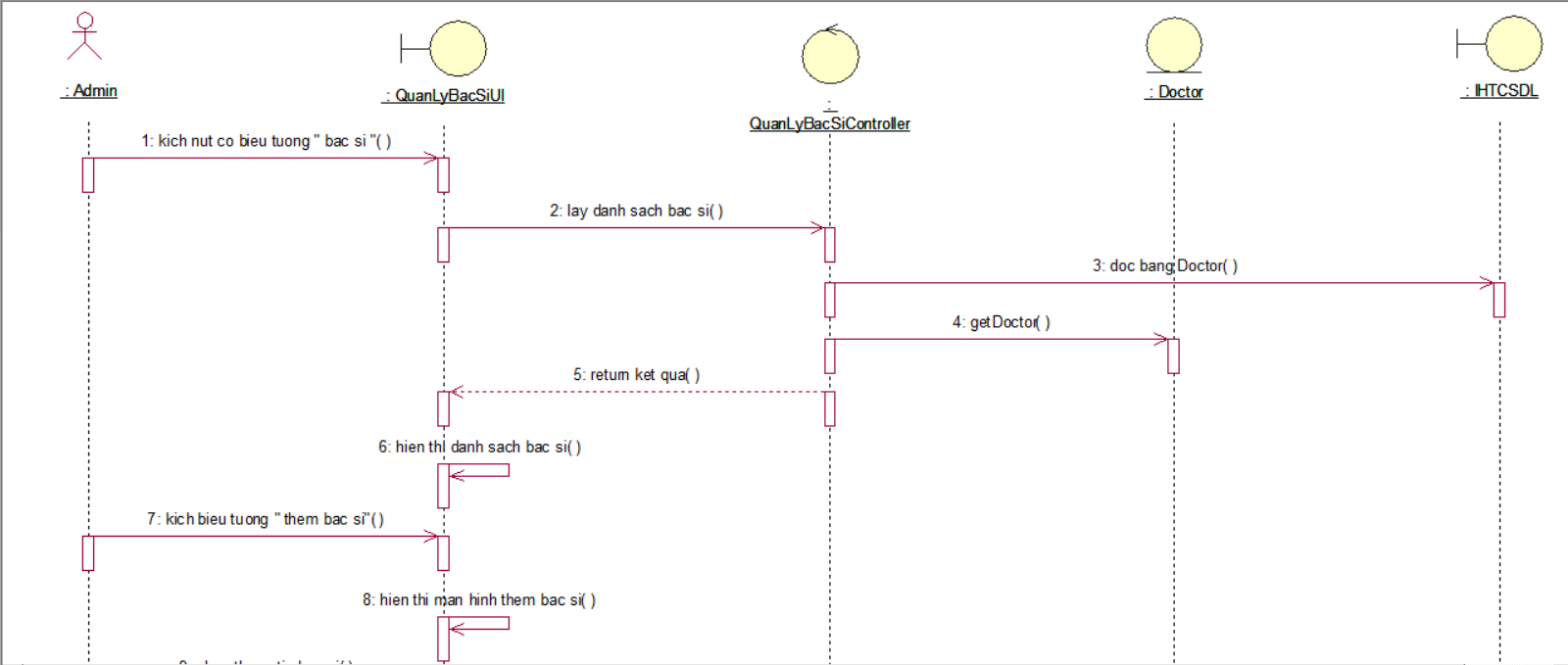


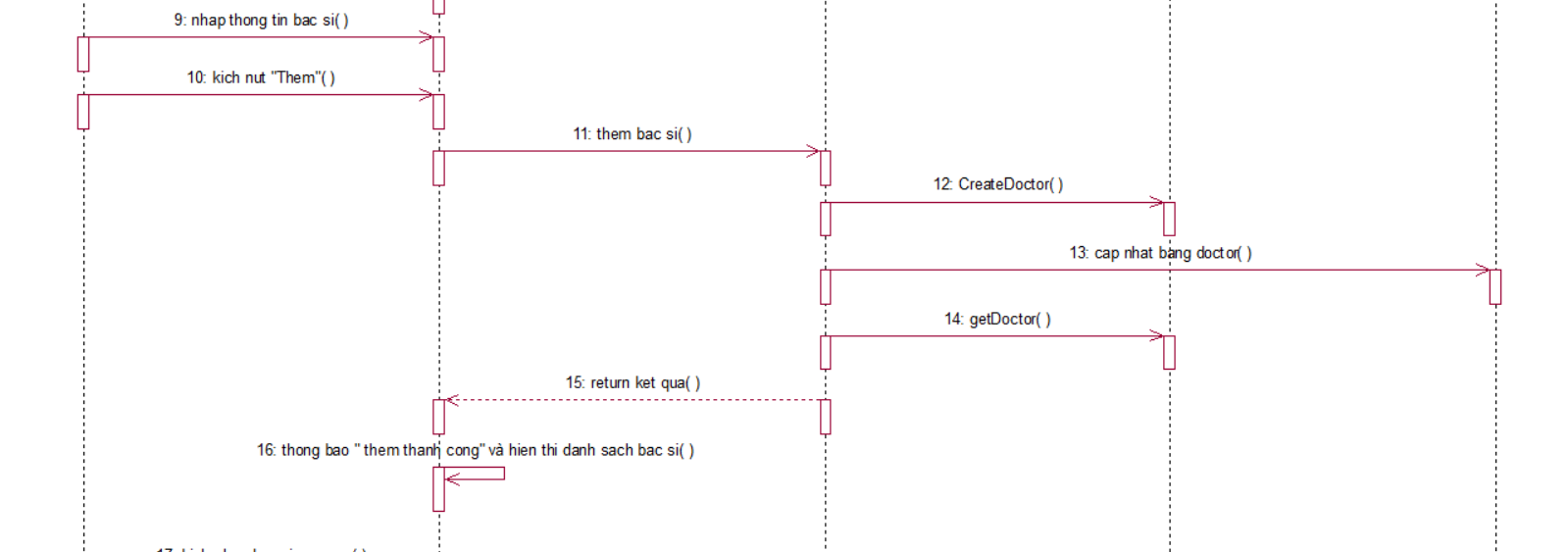
|  |  |  |
| --- | --- | --- |
| **Use case: UC\_02\_Đăng Nhập** | | |
| Mục đích: | User đăng nhập vào hệ thống | |
| Mô tả: | Use case này cho phép người dùng đăng nhập vào hệ thống | |
| Tác nhân: | Khách hàng, Admin, Bác sĩ | |
| Điều kiện đặc biệt: | Không có. | |
| Tiền điều kiện: | Không có. | |
| Luồng cơ bản (Basic flows): | **Tác nhân** | **Hệ thống** |
| 1. . Use case bắt đầu khi người dùng click vào nút “Đăng nhập” trên màn hình | 2. Hệ thống sẽ chuyển sang màn hình “Đăng nhập” và Form đăng nhập. |
| 3. Người dùng nhập tên tài khoản và mật khẩu sau đó click vào nút Đăng nhập. Hệ thống xác nhận tài khoản và mật khẩu có hợp lệ không, nếu hợp lệ thì đăng nhập thành công hệ thống chuyển màn hình đến trang chủ. Use case kết thúc. |
| Luồng rẽ nhánh (Alternative flows): | **Tác nhân** | **Hệ thống** |
| 1. Tại bước 2 trong luồng cơ bản nếu người dùng không nhập tên hoặc mật khẩu, kiểu dữ liệu không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi. Người dùng cần nhập đúng thông tin để tiếp tục. | 2. Tại bước 2 trong luồng cơ bản nếu kiểm tra tài khoản không có trong bảng Users thì hệ thống sẽ hiển thị thông báo lỗi. |
| 3. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc. |
| Hậu điều kiện: | Không có. | |
| Điểm mở rộng: | Không có. | |

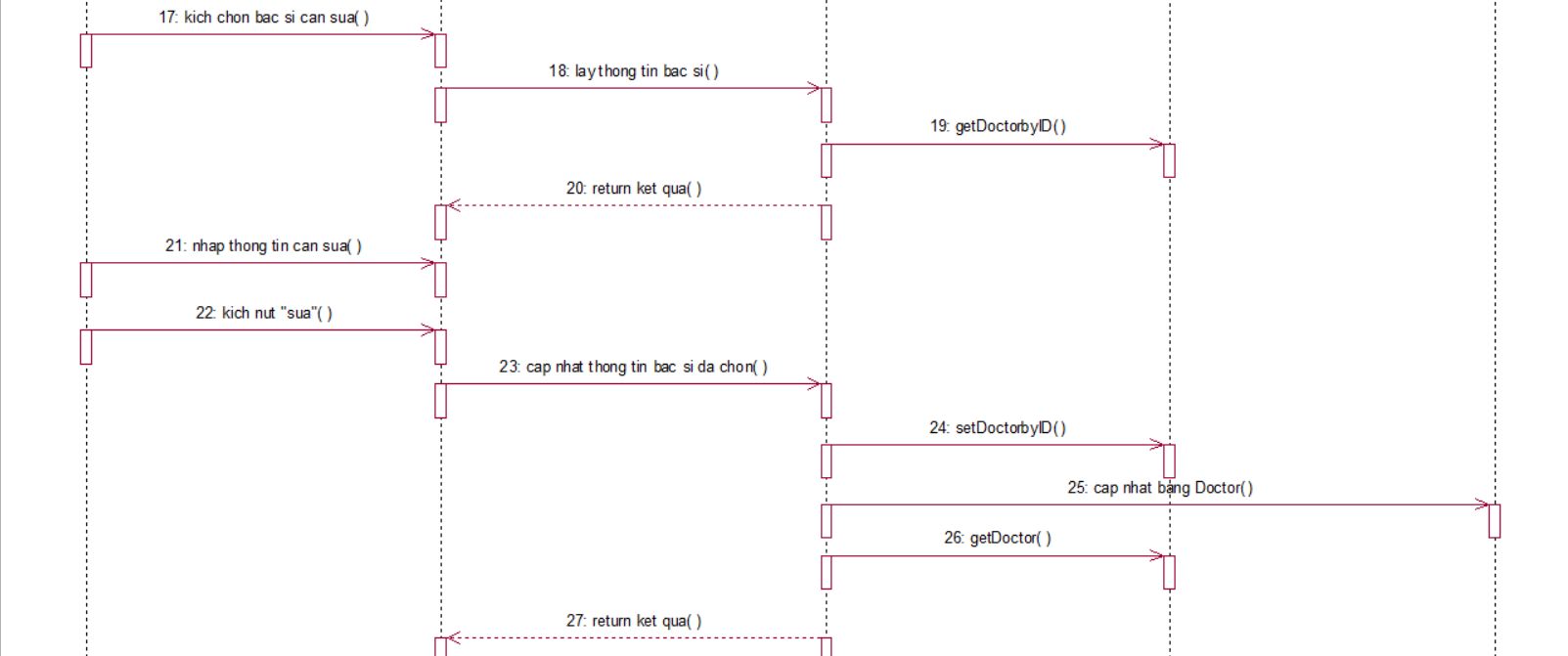


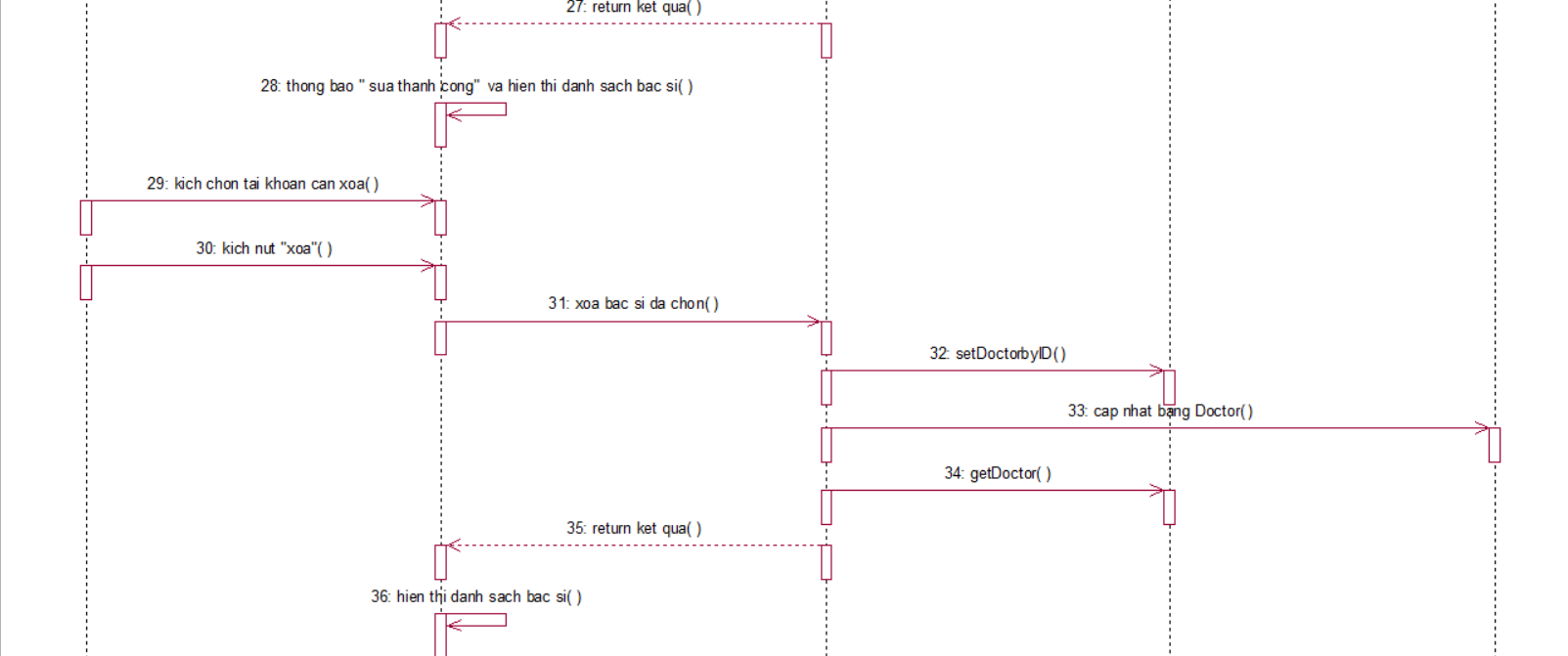


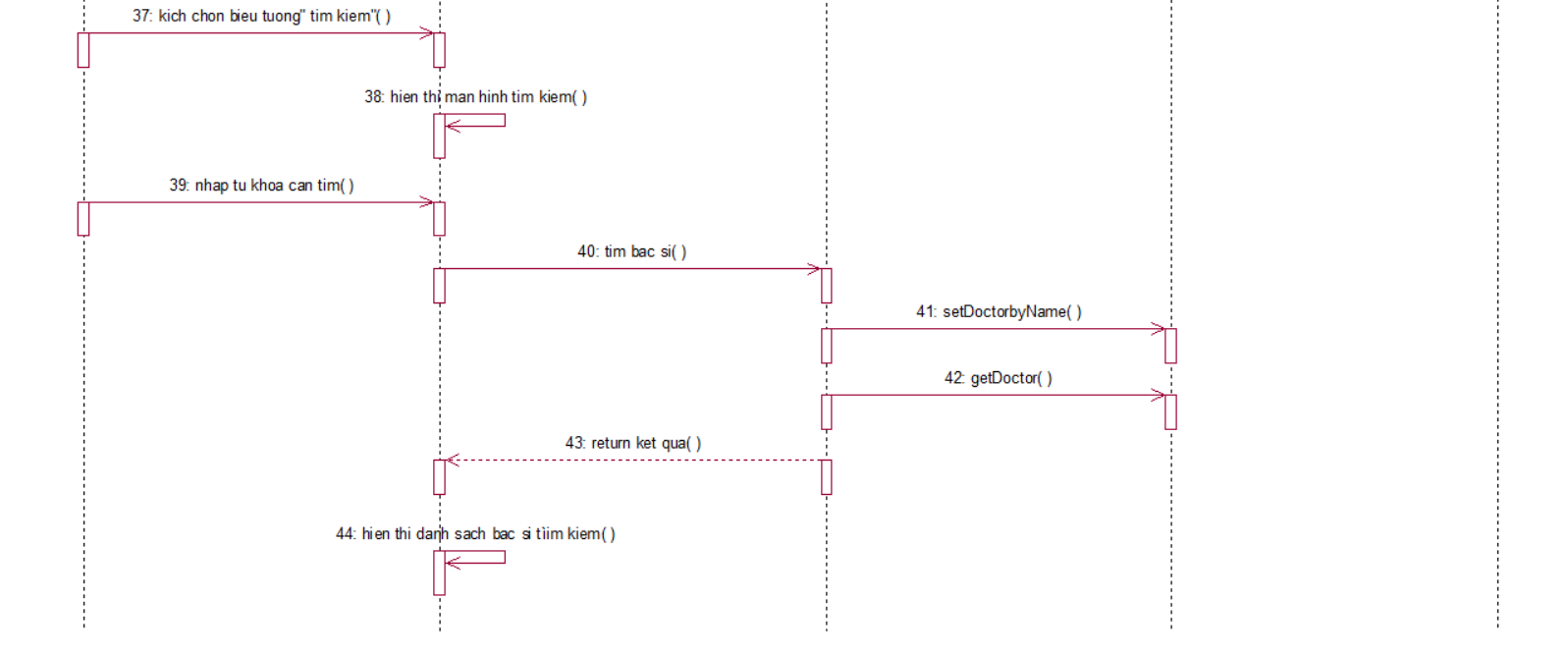
|  |  |  |
| --- | --- | --- |
| **Use case: UC\_03\_Quản lý bác sĩ** | | |
| Mục đích: | Quản lý các bác sĩ trong phòng khám thú y. | |
| Mô tả: | Use case này cho phép người quản trị thực hiện xem, thêm, sửa thông tin, tìm các bác sĩ trong phòng khám thú y. | |
| Tác nhân: | Quản lý | |
| Điều kiện đặc biệt: | Use case này chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện. | |
| Tiền điều kiện: | Người quản trị cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case. | |
| Luồng cơ bản (Basic flows): | **Tác nhân** | **Hệ thống** |
| 1. Use case này bắt đầu khi người quản trị kích vào nút có biểu tượng “bác sĩ” trên menu quản trị. | 2. . Hệ thống sẽ lấy thông tin chi tiết về danh sách các bác sĩ bao gồm thông tin về ảnh đại diện, tên bác sĩ, email, giới tính từ bảng Doctor và hiển thị giao diện quản lý bác sĩ trong phòng khám thú y |
| 3.1. Người quản trị kích nút “Thêm bác sĩ”. | 3.2. Hệ thống sẽ hiển thị một phiếu thêm thông tin cho bác sĩ. |
|  | 3.3. Người quản trị nhập thông tin cho bác sĩ bao gồm tên, ảnh đại diện, số điện thoại, giới tính, email, địa chỉ, chuyên ngành và kích nút “Thêm”. | 3.4. Hệ thống sẽ lấy thông tin và lưu vào cơ sở dữ liệu. |
|  | 4.1. Người quản trị chọn nguyên liệu cần sửa và kích sửa. | 4.2. Hệ thống sẽ lấy thông tin bao gồm ảnh đại diện, tên bác sĩ, số điện thoại, giới tính, email, địa chỉ, chuyên ngành từ bảng DOCTOR về bác sĩ được chọn rồi hiển thị thông tin về bác sĩ lên màn hình. |
|  | 4.3. Người quản trị nhập thông tin cần sửa rồi bấm “Sửa”. | 4.4. Hệ thống sẽ lấy các thông tin từ màn hình hiển thị rồi lưu vào trong cơ sở dữ liệu. |
|  | 5.1. Người quản trị kích vào nút “tìm kiếm” ở màn hình quản lý bác sĩ. | 5.2. Hệ thống sẽ hiển thị thông tin bác sĩ tìm được. |
|  | 5.3. Người quản trị kích nút “Yes”. | 5.4. Hệ thống sẽ xóa nguyên liệu mà người quản trị đã chọn và cập nhật lại danh sách bác sĩ lên màn hình quản lý. |
| Luồng rẽ nhánh (Alternative flows): | **Tác nhân** | **Hệ thống** |
| 1. Tại bước 3.3 hoặc 4.3 hoặc 5.3 trong luồng cơ bản nếu người quản trị kích “Hủy”. | 2. Hệ thống sẽ bỏ qua thao tác và quay trở lại màn hình hiển thị danh sách các nguyên liệu. |
| 3. Tại bất cứ thời điểm nào trong quá trình thực hiện Use Case không kết nối được với cơ sở dữ liệu. | 4. Hệ thống sẽ hiển thị một thông báo lỗi và Use Case kết thúc. |
| Hậu điều kiện: | Nếu use case kết thúc thành công thì thông tin về bác sĩ được cập nhật trong cơ sở dữ liệu . | |
| Điểm mở rộng: | Không có. | |







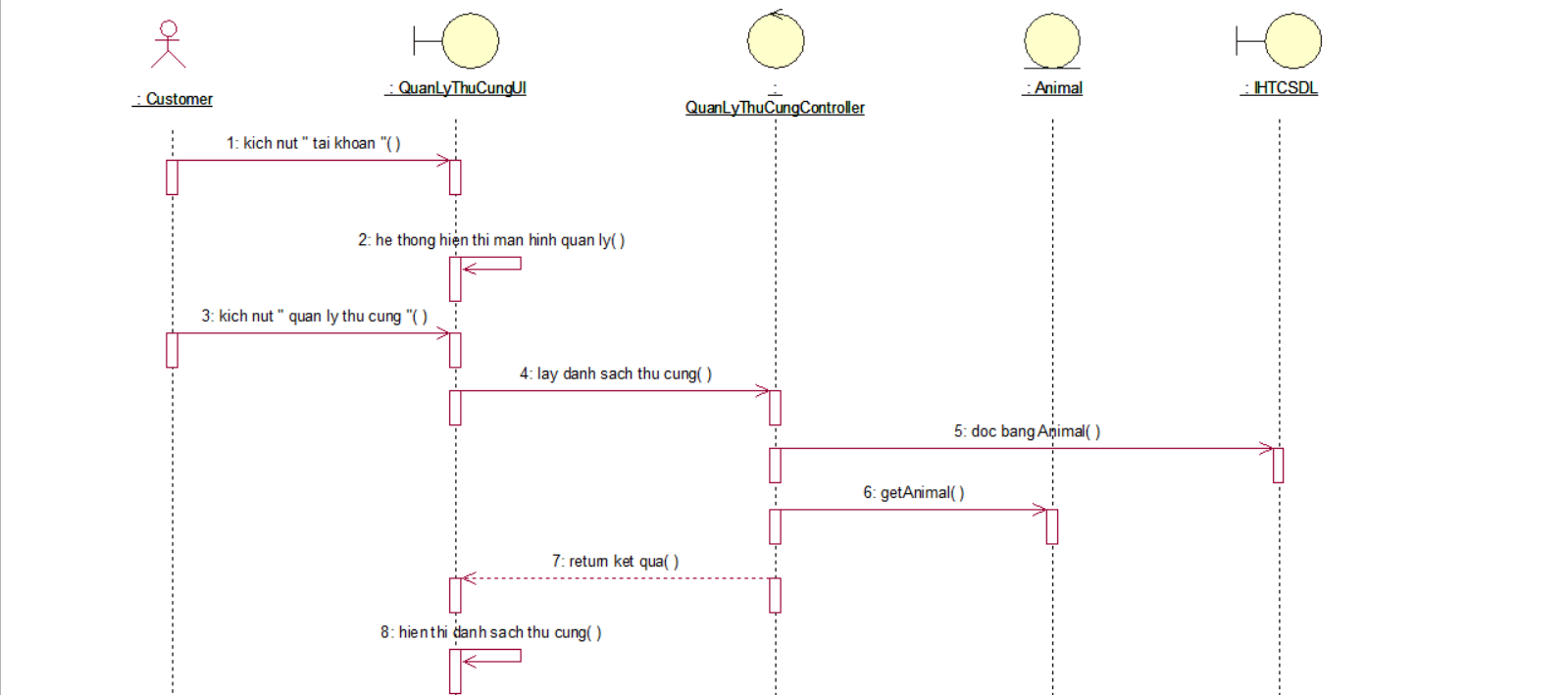


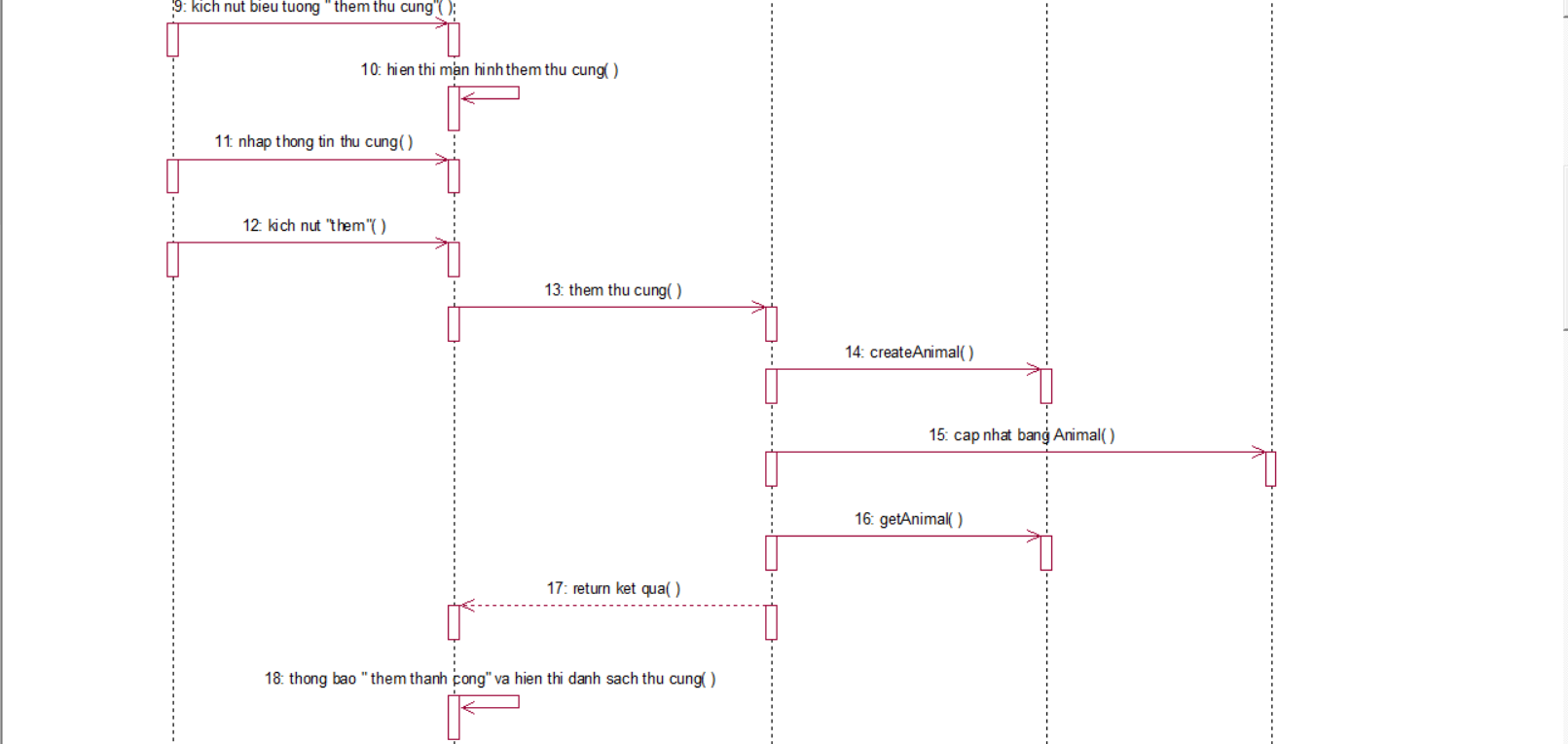


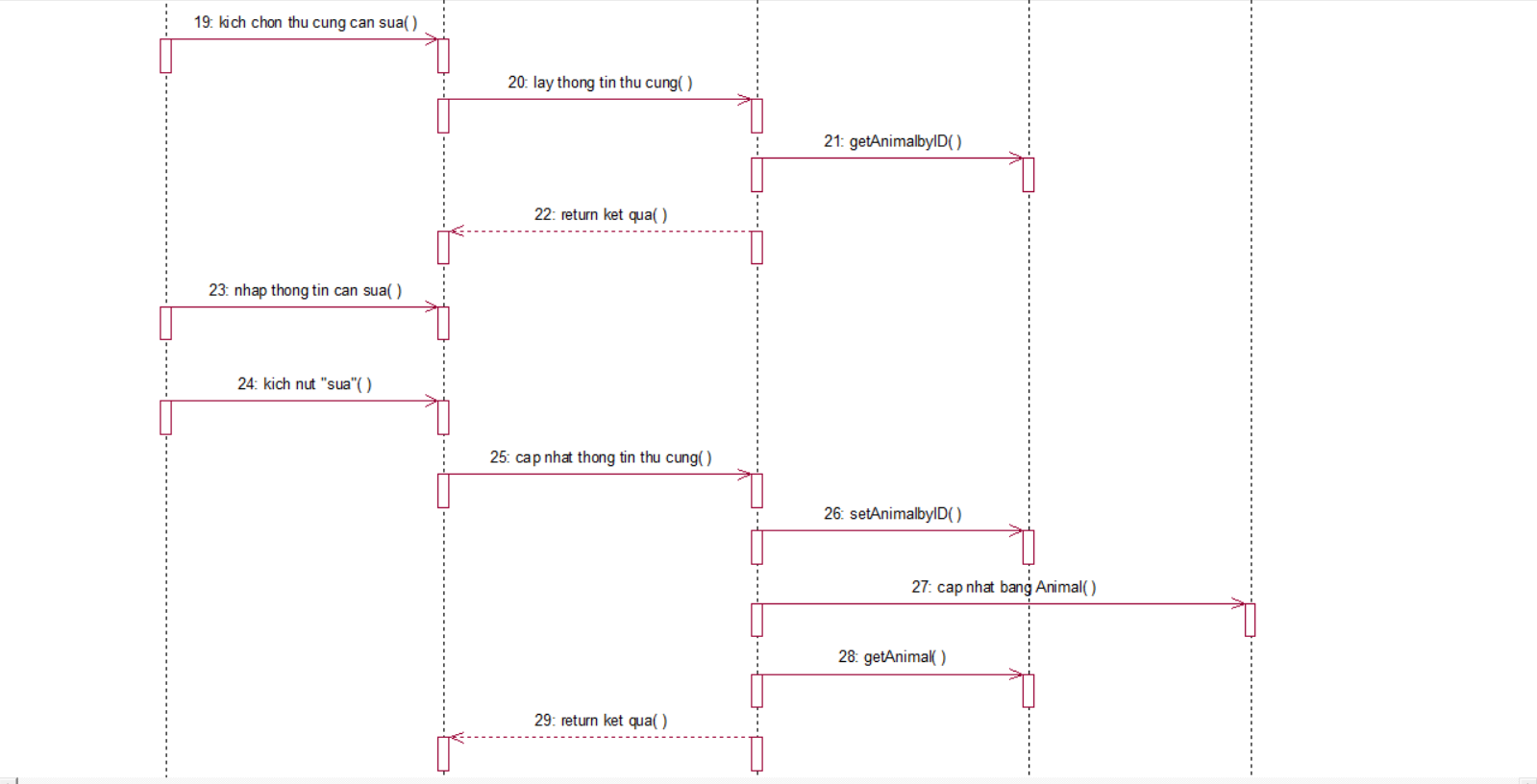


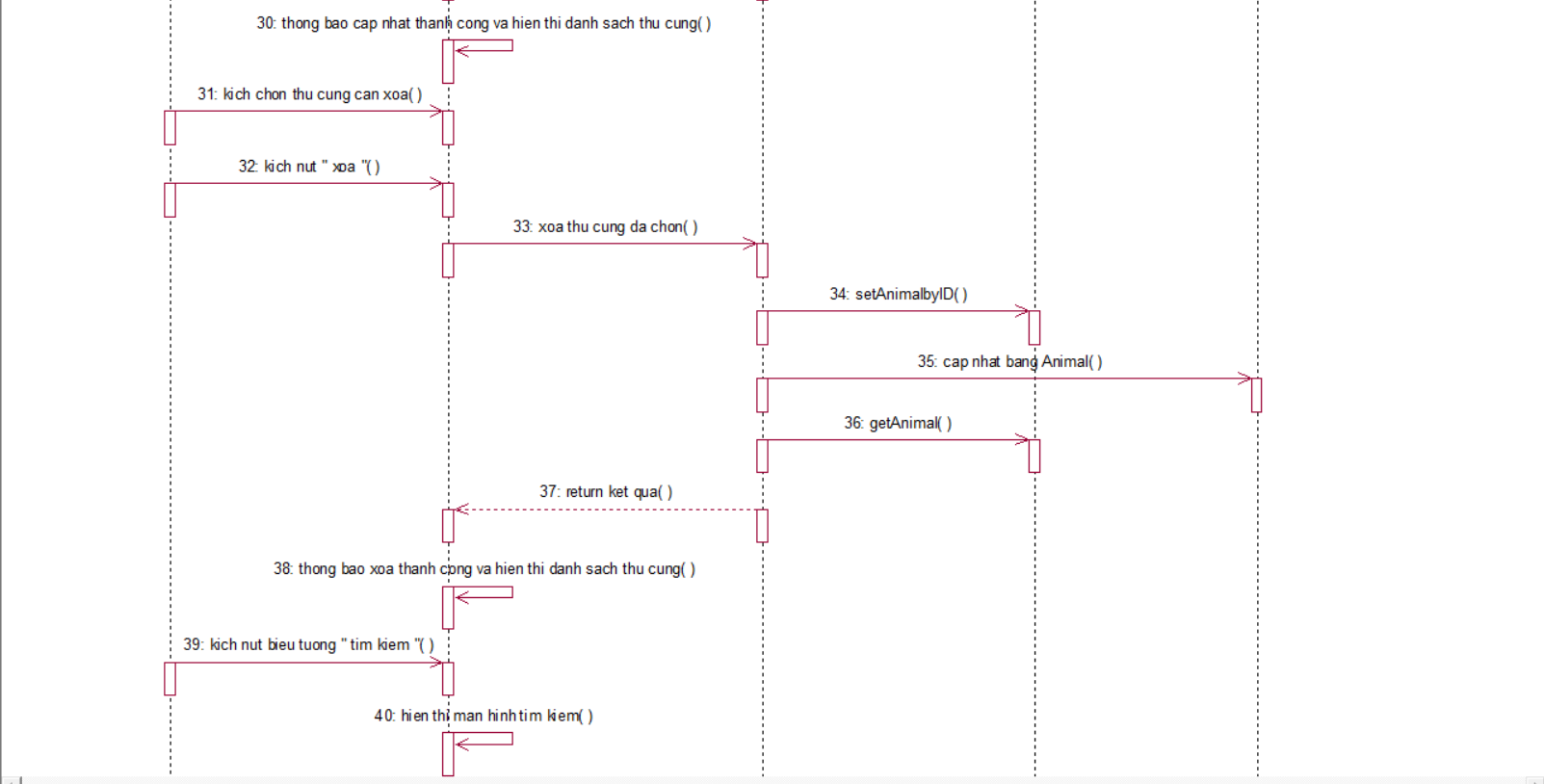
|  |  |  |
| --- | --- | --- |
| **Use case: UC\_04\_Quản lý thú cưng** | | |
| Mục đích: | Quản lý thú cưng của khách hàng | |
| Mô tả: | Use case này cho phép khách hàng thực hiện xem, thêm, sửa, xóa thú cưng của khách hàng | |
| Tác nhân: | Quản lý | |
| Điều kiện đặc biệt: | Use case này chỉ cho phép người dùng là khách hàng đăng nhập tài khoản thực hiện. | |
| Tiền điều kiện: | Khách hàng cần đăng nhập vào hệ thống trước khi có thể thực hiện use case. | |
| Luồng cơ bản (Basic flows): | **Tác nhân** | **Hệ thống** |
| 1. Use case này bắt đầu khi khách hàng kích vào nút “Tài khoản” trên menu quản trị, sau đó kích vào nút “Quản lý thú cưng” hiện thị trên màn hình | 2.Hệ thống sẽ lấy thông tin chi tiết về danh sách thú cưng bao gồm thông tin về ảnh đại diện, tên, tuổi đời, đặc điểm từ bảng Animal và hiển thị lên giao diện quản lý thú cưng |
| 3.1. khách hàng kích nút “Thêm thú cưng”. | 3.2. Hệ thống sẽ hiển thị một phiếu thêm thông tin cho thú cưng |
|  | 3.3. khách hàng nhập thông tin cho thú cưng bao gồm ảnh đại diện, tên, tuổi, đặc điểm và kích nút “Thêm”. | 3.4. Hệ thống sẽ lấy thông tin và lưu vào cơ sở dữ liệu. |
|  | 4.1. khách hàng chọn thú cưng cần sửa và kích sửa. | 4.2. Hệ thống sẽ lấy thông tin bao gồm ảnh đại diện, tên, tuổi, đặc điểm từ bảng Animal về thú cưng được chọn rồi hiển thị thông tin về thú cưng lên màn hình. |
|  | 4.3. khách hàng nhập thông tin cần sửa rồi bấm “Update”. | 4.4. Hệ thống sẽ lấy các thông tin từ màn hình hiển thị rồi lưu vào trong cơ sở dữ liệu. |
|  | 5.1. khách hàng kích vào nút “Xóa” ở màn hình quản lý thú cưng | 5.2. Hệ thống sẽ hiển thị thông báo xác nhận xóa thú cưng |
|  | 5.3. khách hàng kích nút “Yes”. | 5.4. Hệ thống sẽ xóa nguyên liệu mà khách hàng đã chọn và cập nhật lại danh sách thú cưng lên màn hình quản lý. |
| Luồng rẽ nhánh (Alternative flows): | **Tác nhân** | **Hệ thống** |
| 1. Tại bước 3.3 hoặc 4.3 hoặc 5.3 trong luồng cơ bản nếu khách hàng trị kích “Hủy”. | 2. Hệ thống sẽ bỏ qua thao tác và quay trở lại màn hình hiển thị danh sách các thú cưng. |
| 3. Tại bất cứ thời điểm nào trong quá trình thực hiện Use Case không kết nối được với cơ sở dữ liệu. | 4. Hệ thống sẽ hiển thị một thông báo lỗi và Use Case kết thúc. |
| Hậu điều kiện: | Nếu use case kết thúc thành công thì thông tin về thú cưng được cập nhật trong cơ sở dữ liệu . | |
| Điểm mở rộng: | Không có. | |

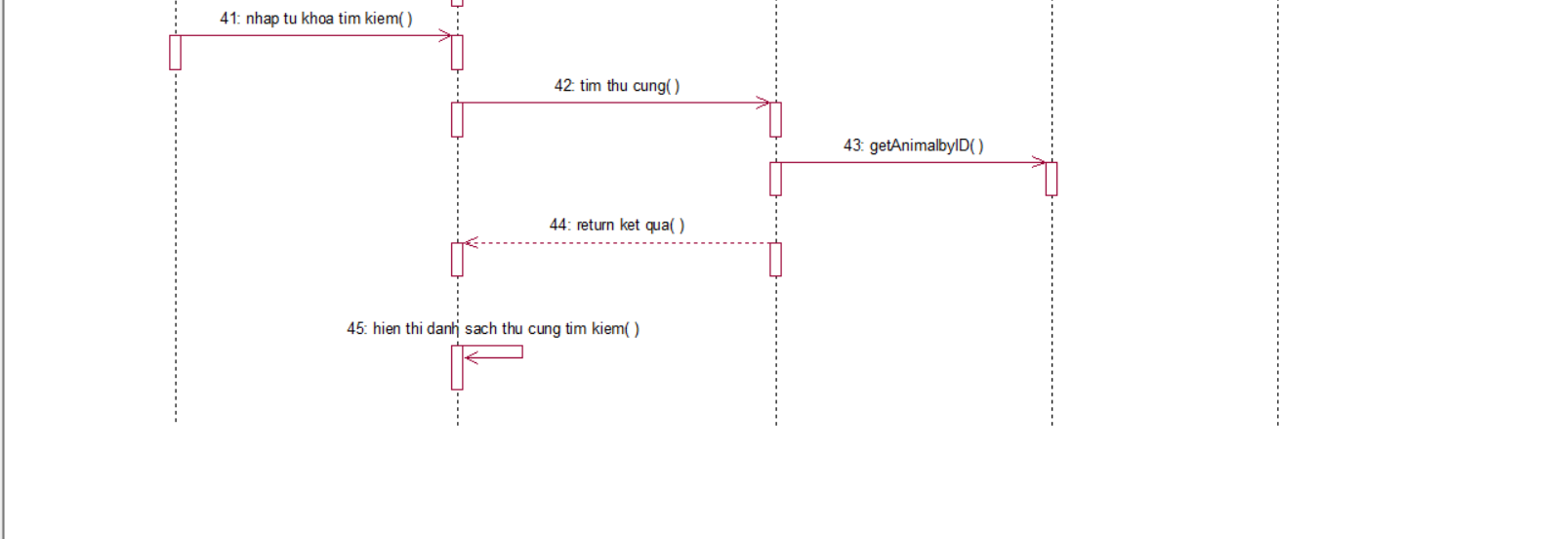












|  |  |  |
| --- | --- | --- |
| **Use case: UC\_05\_Đặt lịch khám** | | |
| Mục đích: | Đặt lịch khám cho thú cưng | |
| Mô tả: | Use case này cho phép khách hàng đặt lịch khám cho thú cưng | |
| Tác nhân: | Người dùng | |
| Điều kiện đặc biệt: | Không có. | |
| Tiền điều kiện: | Khách hàng phải đăng nhập vào hệ thống. | |
| Luồng cơ bản (Basic flows): | **Tác nhân** | **Hệ thống** |
| 1. Use case bắt đầu khi người dùng click vào nút “Đặt lịch” hiển thị trên màn hình trang chủ. | 2. Hệ thống sẽ hiện thị form đặt lịch cho khách hàng. |
| 3. Người dùng kích vào tên 1 bác sĩ | 4. Hệ thống sẽ hiển thị danh sách các các bác sĩ để khách hàng lựa chọn |
|  | 4. Người dùng kích vào tên 1 thú cưng  6.Hệ thống sẽ ghi nhận dữ liệu đặt lịch của khách hàng và thông báo đặt lịch thành công.  Usecase kết thúc. | 5. Hệ thống sẽ hiển thị danh sách các các bác sĩ để khách hàng lựa chọn. khách hàng tiếp tụ nhập ảnh tình trạng của thú cưng, nơi khám, ngày hẹn, giờ hẹn, địa chỉ và dịch vụ. Sau đó kích vào nút “đặt lịch”. |
| Luồng rẽ nhánh (Alternative flows): | **Tác nhân** | **Hệ thống** |
| 1. Tại bước 3 và 5 trong luồng cơ bản, nếu khách hàng chọn cùng 1 bác sĩ và cùng ngày hẹn và giờ hẹn đã có lịch | 2. hệ thống hiển thị “trùng lịch hẹn, mời bạn chọn ngày khác”. |
| 3. Tại bất cứ thời điểm nào trong quá trình thực hiện Use Case không kết nối được với cơ sở dữ liệu. | 4. Hệ thống sẽ hiển thị một thông báo lỗi và Use Case kết thúc. |
| Hậu điều kiện: | Không có | |
| Điểm mở rộng: | Không có. | |

|  |  |  |
| --- | --- | --- |
| **Use case: UC\_06\_Quản lý người dùng** | | |
| Mục đích: | Quản lý các tài khoản người dùng. | |
| Mô tả: | Use case này cho phép người quản trị thực hiện xem, sửa, tìm kiếm tài khoản người dùng | |
| Tác nhân: | Quản lý | |
| Điều kiện đặc biệt: | Use case này chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện. | |
| Tiền điều kiện: | Người quản trị cần đăng nhập với vai trò quản trị hệ thống trước khi có thể thực hiện use case. | |
| Luồng cơ bản (Basic flows): | **Tác nhân** | **Hệ thống** |
| 1. Use case này bắt đầu khi người quản trị kích vào nút “tài khoản” trên menu quản trị, sau đó kích vào nút “ quản lý người dùng” | 2. Hệ thống sẽ hiển thị giao diện quản lý người dùng và danh sách các tài khoản người dùng gồm có họ tên và số điện thoại từ bảng Users trong cơ sở dữ liệu và hiển thị lên màn hình thông tin của các tài khoản. |
| 3.1. Người quản trị kích vào tài khoản 1 người dùng | 3.2. Hệ thống sẽ hiển thị thông tin chi tiết của người dùng bao gồm tên tài khoản, mật khẩu, họ tên, email, số điện thoại, giới tính lấy từ bảng Users rồi hiển thị lên màn hình . |
|  | 3.3. Người quản trị nhập thông tin cần sửa rồi bấm “Cập nhật”. | 3.4. Hệ thống cập nhật thông tin tương ứng của tài khoản được chọn vào bảng Users và cập nhật lại danh sách các tài khoản lên màn hình quản lý tài khoản. |
|  | 4.1. Người quản trị kích vào biểu tượng “tìm kiếm” | 4.2. Hệ thống sẽ hiển thị danh sách các tài khoản người dùng gồm có họ tên và số điện thoại từ bảng Users trong cơ sở dữ liệu tương ứng với tên tìm kiếm của người quản trị  Use case kết thúc. |
|  |
|  |
|  |
| Luồng rẽ nhánh (Alternative flows): | **Tác nhân** | **Hệ thống** |
| 1. Tại bất cứ thời điểm nào trong quá trình thực hiện Use Case không kết nối được với cơ sở dữ liệu. | 2. Hệ thống sẽ hiển thị một thông báo lỗi và Use Case kết thúc. |
| Hậu điều kiện: | Nếu use case kết thúc thành công thì thông tin về tài khoản sẽ được cập nhật trong cơ sở dữ liệu. | |
| Điểm mở rộng: | Không có. | |