TRƯỜNG ĐẠI HỌC PHENIKAA

KHOA CÔNG NGHỆ THÔNG TIN

⸎⸎⸎⸎⸎

A logo of a university

Description automatically generated

BÀI TẬP LỚN KẾT THÚC HỌC PHẦN

“LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG”

Đề tài: “Phần mềm quản lý chuyến bay”

Nhóm: 14

Sinh viên : Lê Trung Hiếu – 22010482 (Nhóm trưởng)

Phạm Việt Hoàng – 22010503

Nguyễn Đình Đức Trung -22010448

Lớp : Lập trình hướng đối tượng-1-2-23(N04)

Giảng viên : Phạm Đức Bắc

*HÀ NỘI, THÁNG 3 NĂM 2024*

PHÂN CHIA CÔNG VIỆC

|  |  |  |
| --- | --- | --- |
| Thành viên | % Đóng góp | Nhiệm vụ |
| Lê Trung Hiếu (Trưởng nhóm) | 40% | * Quản lý và điều hành dự án, phân công công việc, giám sát tiến độ, đảm bảo chất lượng sản phẩm. * Hoàn thiện phần Controller và Model của dự án: Xử lý sự kiện đầu vào, kết nối cơ sở dữ liệu… |
| Nguyễn Đình Đức Trung | 30% | * Hoàn thiện phần Database, thiết kế cơ sở dữ liệu |
| Phạm Việt Hoàng | 30% | * Thiết kế giao diện người dùng |

*(Đây là đóng góp tính tới thời điểm nộp báo cáo, có thể sẽ thay đổi trong quá trình hoàn thiện bài)*

MỤC LỤC

[LỜI MỞ ĐẦU 3](#_Toc162008505)

[CHƯƠNG 1: TỔNG QUAN VỀ HOẠT ĐỘNG ĐẶT VÉ MÁY BAY 4](#_Toc162008506)

[1. Hoạt động đặt vé máy bay 4](#_Toc162008507)

[2. Những khó khăn khi tái hiện lại quy trình đặt vé của Vietnam Airline và các sửa đổi 4](#_Toc162008508)

[CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ BÀI TOÁN 5](#_Toc162008509)

[1. Mô hình Model – View – Controller (MVC) 5](#_Toc162008510)

[2. Vì sao cần sử dụng MVC 5](#_Toc162008511)

[3. Ứng dụng mô hình MVC vào quản lý chuyến bay 6](#_Toc162008512)

[*a.* *Đăng ký – Dành cho người chưa có tài khoản* 6](#_Toc162008513)

[*b.* *Đăng nhập – Dành cho người đã có tài khoản* 6](#_Toc162008514)

[CHƯƠNG 3: GIAO DIỆN VÀ CÁCH VẬN HÀNH 8](#_Toc162008515)

[1. Màn hình đăng nhập 8](#_Toc162008516)

[2. Màn hình đăng ký 9](#_Toc162008517)

[3. Màn hình giao diện chính 11](#_Toc162008518)

[a. Giao diện sử dụng của tài khoản “Người dùng” 11](#_Toc162008519)

[b. Giao diện sử dụng của tài khoản “Người quản lý” 18](#_Toc162008520)

[CHƯƠNG 4: CÁC TÍNH NĂNG CÓ THỂ SẼ ĐƯỢC BỔ SUNG THÊM 21](#_Toc162008521)

[1. Tính năng đổi mật khẩu 21](#_Toc162008522)

[2. Tính năng thông báo 21](#_Toc162008523)

[DANH MỤC TÀI LIỆU THAM KHẢO 21](#_Toc162008524)

# LỜI MỞ ĐẦU

Trước sự phát triển không ngừng của ngành hàng không, việc quản lý chuyến bay trở thành một yếu tố quan trọng đối với các hãng hàng không và các tổ chức liên quan. Để đáp ứng nhu cầu này, các hệ thống phần mềm quản lý chuyến bay đã trở thành một phần không thể thiếu, hỗ trợ quản lý mọi khía cạnh của quá trình vận hành hàng không.

Trong thế giới kỹ thuật phần mềm, lập trình hướng đối tượng (OOP) đã trở thành một phương pháp tiêu biểu cho việc phát triển phần mềm. Với việc tập trung vào các đối tượng và mối quan hệ giữa chúng, OOP giúp tạo ra các ứng dụng linh hoạt, dễ bảo trì và mở rộng. Trong bối cảnh này, việc tự tay xây dựng một ứng dụng quản lý chuyến bay không chỉ là một bước đi đúng đắn mà còn là cơ hội để hiểu sâu hơn về lập trình hướng đối tượng và áp dụng nó vào thực tiễn.

Bằng cách tham gia vào quá trình xây dựng một ứng dụng quản lý chuyến bay, chúng ta sẽ có cơ hội tiếp cận và thực hành các nguyên lý OOP như kế thừa, đa hình và đóng gói. Điều này giúp chúng ta nắm vững cách tổ chức và quản lý mã nguồn một cách hiệu quả hơn, từ đó tăng tính linh hoạt và tái sử dụng của mã nguồn. Hơn nữa, việc tự tay xây dựng một ứng dụng sẽ là cơ hội để áp dụng các kiến thức lập trình vào một dự án thực tế, từ đó củng cố và mở rộng kỹ năng lập trình của chúng ta.

Trong báo cáo này, chúng ta sẽ đi sâu vào việc phân tích, thiết kế và triển khai một phần mềm quản lý chuyến bay, nhấn mạnh vào việc áp dụng lập trình hướng đối tượng trong quá trình này và những lợi ích mà việc tự tay xây dựng một ứng dụng mang lại. Chúng ta cũng sẽ xem xét các thách thức có thể phát sinh và cách vượt qua chúng để đạt được một hệ thống phần mềm chất lượng và linh hoạt.

# CHƯƠNG 1: TỔNG QUAN VỀ HOẠT ĐỘNG ĐẶT VÉ MÁY BAY

1. **Hoạt động đặt vé máy bay**

* Để có thể tiến hành đặt vé máy bay, như chúng tôi có tham khảo trên trang Web chính thức của Vietnam Airline, người dùng sẽ được yêu cầu cung cấp địa chỉ đi và địa chỉ đến cũng như ngày bay và ngày trở về để hệ thống có thể truy vấn và đưa ra được những chuyến bay có sẵn đáp ứng đầy đủ các yêu cầu của khách hàng.
* Khi khách hàng lựa chọn được chuyến bay phù hợp với lộ trình và ngày giờ cất hạ cánh của mình, họ sẽ tiếp tục được lưa chọn hai yếu tố khác là hạng ghế và loại vé máy bay. Với các tuỳ chọn như ghế thương gia, ghế thường hay vé loại một chiều hay hai chiều.
* Sau khi đã chọn hoàn thành việc chọn ghế ngồi, phần mềm sẽ dựa trên các thông số người dùng đã chọn để tính toán số tiền mà người dùng phải chi trả. Xuất hoá đơn và yêu cầu người dùng thanh toàn.
* Ở bước thanh toán này, do đây là hoạt động vận chuyển người liên tỉnh, liên quốc gia nên sẽ yêu cầu người đặt vé phải cung cấp đầy đủ các thông tin như: Họ, tên, ngày tháng năm sinh, số điện thoại, email để tiện liên lạc và gửi xác nhận đặt vé sau đó.
* Đây là quy trình tốt nhất mà chúng tôi hướng đến khi ứng dụng lập trình hướng đối tượng vào giải quyết bài toán quản lý chuyến bay.

1. **Những khó khăn khi tái hiện lại quy trình đặt vé của Vietnam Airline và các sửa đổi**

* Vấn đề lớn nhất khi thực hiện lại quy trình trên đó là dữ liệu, Vietnam Airline có một dữ liệu khổng lồ về danh sách các chuyến bay đến và đi cả trong và ngoài nước do đó, có thể dễ dàng thực hiện các truy vấn và đưa ra danh sách các chuyến bay thích hợp cho người dùng lựa chọn.
* Tuy nhiên với project nhóm nhỏ như thế này, chúng ta sẽ rất khó để tạo ra một lượng dữ liệu lớn như vậy để cho người dùng tiến hành thao tác chọn. Do đó, nhóm đã quyết định sẽ đổi phương hướng và coi như luôn luôn có một chuyến bay có thể đáp ứng được yêu cầu của khách hàng để tránh việc phải tự tạo ra một lượng lớn dữ liệu.
* Khó khăn thứ 2 là chúng ta không hề biết giá vé được tính như thế nào, thế nên là chúng tôi sẽ tự đưa ra công thức tính giá vé của riêng mình. Chúng tôi sẽ ghi lại khoảng cách giữa các thành phố với nhau và đưa ra giá cố định cho mỗi kilomet là 5 nghìn đồng và sẽ tăng thêm 15% nếu lựa chọn hạng thương gia.

# CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ BÀI TOÁN

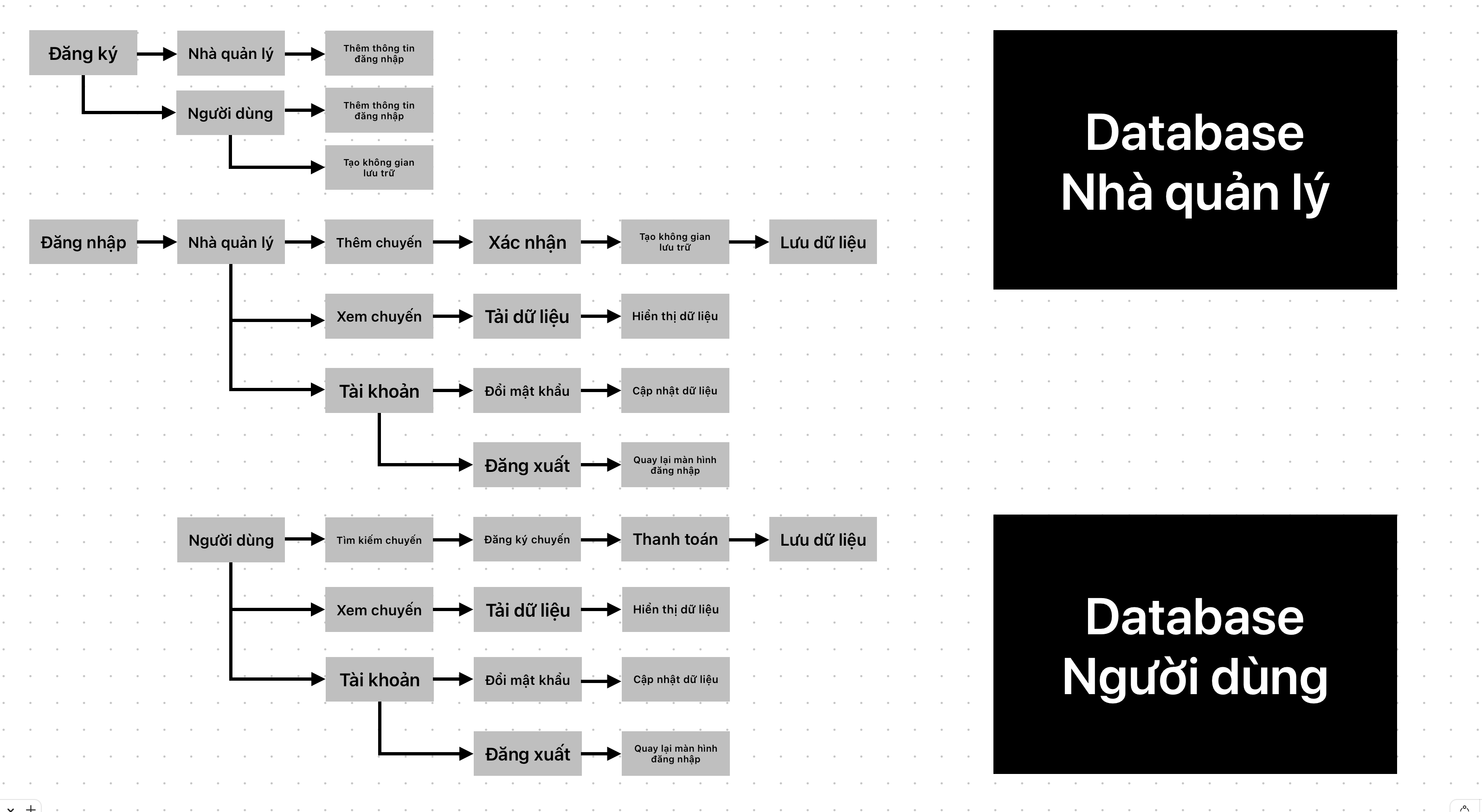
1. **Mô hình Model – View – Controller (MVC)**

* Model – View – Controller (MVC) là một mẫu thiết kế nhằm mục tiêu chia tách từng phân của giao diện và code để dễ quản lý, phát triển và bảo trì, MVC chia ứng dụng phần mềm ra thành 3 phần có tương tác với nhau là Model (Dữ liệu), View(Giao diện), Controller(Code điều khiển các chức năng).
* Model: Bộ phân có chức năng lưu trữ dữ liệu của ứng dụng và là cầu nối giữa 2 thành phần bên trên là View và Controller. Ở trong bài toàn lần này, chúng tôi đã sử dụng JDBC - Java Database Connectivity (JDBC). Đây là một API được thiết kế dành cho ngôn ngữ lập trình Java hỗ trợ Java trong việc truy cập Cơ Sở Dữ Liệu (CSDL). Nó gồm có những phương thức thực hiện truy vấn và cập nhật CSDL gián tiếp qua Java. Cụ thể, trong ứng dụng của chúng tôi, phần mềm sẽ được kết nối trực tiếp đến MySQL để tiện lợi trong quá trình truy xuất dữ liệu.
* View: Đây là phần giao diện dành cho người sử dụng. View là phương tiện hiển thị các đối tượng của trong một ứng dụng. Chẳng hạn như hiển thị các cửa sổ, nút, văn bản,…Nó bao gồm tất cả những thứ người dùng có thể nhìn thấy được.
* Controller: Là bộ phận có nhiệm vụ xử lý các yêu cầu người dùng đưa đến thông qua View. Một Controller bao gồm cả Model lẫn View. Nó nhận input và thực hiện các update tương ứng

1. **Vì sao cần sử dụng MVC**

* Tính tách biệt Logic:
* Mô hình MVC giúp chia nhỏ ứng dụng thành các thành phần riêng biệt, mỗi thành phần chịu trách nhiệm về một phần cụ thể của ứng dụng.
* Model chịu trách nhiệm xử lý dữ liệu và logic liên quan.
* View hiển thị dữ liệu cho người dùng một cách thích hợp.
* Controller điều khiển luồng dữ liệu và tương tác giữa Model và View.
* Dễ bảo trì và mở rộng:
* Sự tách biệt rõ ràng giữa các thành phần trong mô hình MVC làm cho việc bảo trì và mở rộng trở nên dễ dàng hơn.
* Mỗi thành phần có thể được phát triển và kiểm thử một cách độc lập, giúp giảm thiểu rủi ro gây ra mỗi khi thay đổi mã nguồn.
* Tính tái sử dụng cao:
* Do các thành phần của MVC được tách biệt một cách rõ ràng, chúng có thể được tái sử dụng trong nhiều phần của ứng dụng hoặc trong các ứng dụng khác.
* Model có thể tái sử dụng logic xử lý dữ liệu cho các giao diện người dùng khác nhau.
* View có thể tái sử dụng các thành phần giao diện người dùng cho các mô hình dữ liệu khác nhau.
* Phân chia công việc
* Mô hình MVC phân chia công việc giữa các nhóm phát triển một cách hiệu quả, giúp tăng khả năng quản lý và phát tiển ứng dụng.
* Nhóm phát triển giao diện người dùng có thể tập trung và phát triển giao diện mà không cần quan tâm về Logic dữ liệu.
* Nhóm phát triển logic dữ liệu có thể tập trung vào xử lý dữ liệu mà không cần quan tâm đến các hiển thị.

1. **Ứng dụng mô hình MVC vào quản lý chuyến bay**



Sơ đồ MVC của bài toán quản lý chuyến bay

Bài toàn được chia thành các luồng xử lý khác nhau phù hợp cho các tình huống cụ thể: người dùng chưa có tài khoản, người dùng đã có tài khoản, người dùng là nhà quản lý và người dùng là khách hàng đang muốn đặt chuyến.

1. ***Đăng ký – Dành cho người chưa có tài khoản***

* Với người dùng chưa có tài khoản, họ sẽ sử dụng chức năng đăng ký để tạo tài khoản mới. Người dùng sẽ được đăng lý dưới hai dạng là khách hàng và nhà quản lý. Với cả hai loại tài khoản người dùng đều được yêu cầu nhập hai trường “Username” và “Password” nhưng với từng loại tài khoản sẽ được lưu trữ trong database khác nhau.

1. ***Đăng nhập – Dành cho người đã có tài khoản***

* Khi tiến hành đăng nhập người dùng cũng sẽ được yêu cầu lựa chọn một trong hai loại tài khoản “Nhà quản lý” và “Người dùng”.
* Giao diện của hai loại tài khoản này sẽ có nhiều sự khác biệt để phù hợp với tình năng của từng loại tài khoản, cụ thể:
* *Với giao diện của người dùng, khách hàng sẽ có tính năng đặt chuyến bay mới, xem danh sách các chuyến bay đã đặt, huỷ chuyến, thiết lập lại tài khoản..*
* *Với giao diện của nhà quản lý, chúng ta sẽ có thể thêm các chuyến bay mới vào trong hệ thống, thêm sửa xoá các chuyến bay, xem danh sách các chuyến bay đã thêm theo ngày, thiết lập lại tài khoản…*
* *Về giao diện và cơ chế hoạt động của từng chức năng, chúng tôi sẽ giới thiệu chi tiết ở phần sau.*

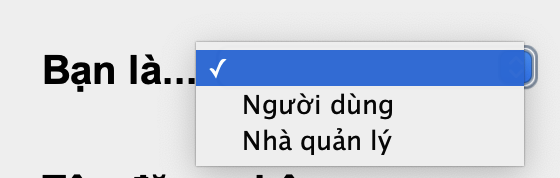
# CHƯƠNG 3: GIAO DIỆN VÀ CÁCH VẬN HÀNH

1. **Màn hình đăng nhập**



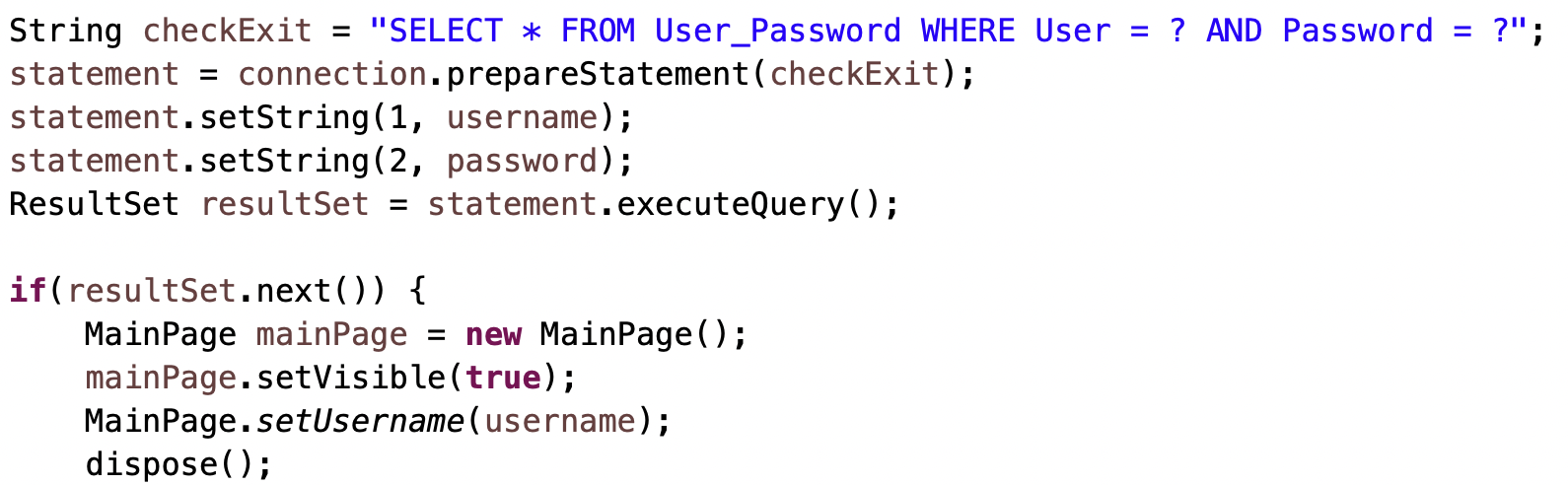
Hình 1. Giao diện màn hình đăng nhập

* Với giao diện màn hình đăng nhập, trước hết người dùng cần chọn loại tài khoản mình đang sở hữu:



Hai loại tài khoản khác nhau sẽ sử dụng hai database riêng biệt để lữu trữ nên nếu không chọn đúng loại tài khoản, người dùng sẽ không thể đăng nhập.

* Tiếp đó là người dùng sẽ phải điền đầy đủ hai trường dữ liệu “Username” và “Password”, hệ thống sẽ sử dụng dữ liệu từ hai trường này, truy xuất vào cơ sở dữ liệu đúng theo loại tài khoản người dùng đã chọn và kiểm tra xem người dùng có tồn tại hay không, và mật khẩu đi kèm với tài khoản này có chính xác hay không:



Nếu mật khẩu và tài khoản chính xác, giao diện sử dụng sẽ hiện ra còn nếu mật khẩu hoặc tài khoản sai thì sẽ hiện lên bảng thông báo:

A screenshot of a phone

Description automatically generated

1. **Màn hình đăng ký**

* Nếu người dùng chưa có tài khoản, có thể sử dụng nút “Đăng ký” ở góc phải phần đăng nhập để tạo tài khoản mới, hệ thống sẽ cung cấp một giao diện mới để người dùng điền các thông tin cần thiết:

A screenshot of a computer

Description automatically generated

* Trước hết người dùng sẽ phải chọn loại tài khoản mình muốn đăng ký.
* Sau đó hoàn thiện các TextField bao gồm: Tên đăng nhập, mật khẩu, nhập lại mật khẩu. Trường “Nhập lại mật khẩu” có chức năng đảm bảo người dùng sẽ không vô tình gõ nhầm mật khẩu dẫn tới việc không thể đăng nhập sau khi đã đăng ký tài khoản.
* Cơ chế làm việc của màn hình đăng ký như sau:
* Khi người dùng nhập hoàn thiện đầy đủ cả 3 TextField, hệ thống sẽ kiểm tra xem các thông tin người dùng nhập đã đúng với yêu cầu chưa: Kiểm tra xem “Mật khẩu” và “Nhập lại mật khẩu” đã giống nhau chưa, kiểm tra xem “Mật khẩu” đã đủ dài hay chưa.
* Sau khi việc kiểm tra hoàn tất sẽ tạo một truy vấn đến Table “User\_Password” và lưu trữ “Tên đăng nhập” và “Mật khẩu” của người dùng vào bảng này.
* Đồng thời tiếp tục tạo thêm một truy vấn nữa đến CSDL yêu cầu tạo ra một bảng mới với tên là Username của người dùng (Chỉ với tài khoản mới là “người dùng”).
* Nếu người dùng đăng ký thành công, sẽ có bảng thông báo “Bạn đã đăng ký thành công” và sẽ có tùy chọn quay lại màn hình đăng nhập để đăng nhập.

A screenshot of a computer

Description automatically generated

A screenshot of a phone

Description automatically generated

1. **Màn hình giao diện chính**

* Với hai loại tài khoản khác nhau “Người dùng” và “Nhà quản lý” nên giao diện sử dụng chính của hai loại tài khoản này cũng sẽ có sự khác biệt rõ rệt.

1. **Giao diện sử dụng của tài khoản “Người dùng”**

A screenshot of a computer

Description automatically generated

* Đây là giao diện bắt đầu của tài khoản “Người dùng”. Chúng ta sẽ có 3 nút ở bên cạnh trái của giao diện ứng với 3 chức năng chính: Đặt chuyến, Xem các chuyến bay đã đặt, cài đặt tài khoản.
* Đầu tiên là cơ chế đặt chuyến bay:

A screenshot of a computer

Description automatically generated

* Để có thể tiến hành đặt chuyến, trước tiên người dùng phải chọn điểm đến và điểm xuất phát:

A screenshot of a computer

Description automatically generatedA close up of a box

Description automatically generated

Dựa trên các điểm đến đã chọn, chương trình sẽ loại trừ các điểm đi không khả thi, ví dụ như “Điểm đến” là Hà Nội thì điểm xuất phát không thể là Quảng Ninh được, do đó trường hợp này sẽ bị loại.

* Sau khi chọn được điểm đến và điểm xuất phát, người dùng sẽ sử dụng nút search để tìm kiếm chuyến bay phù hợp với yêu cầu của mình, bảng phía dưới sẽ hiển thị một loại các chuyến bay phù hợp với yêu cầu của người dùng, nó sẽ tự động loại trừ các chuyến bay đã quá hạn bằng cách so sánh thời gian khởi hành với thời gian hiện tại và chỉ đưa những chuyến bay khả dụng, có thể đặt vé được:

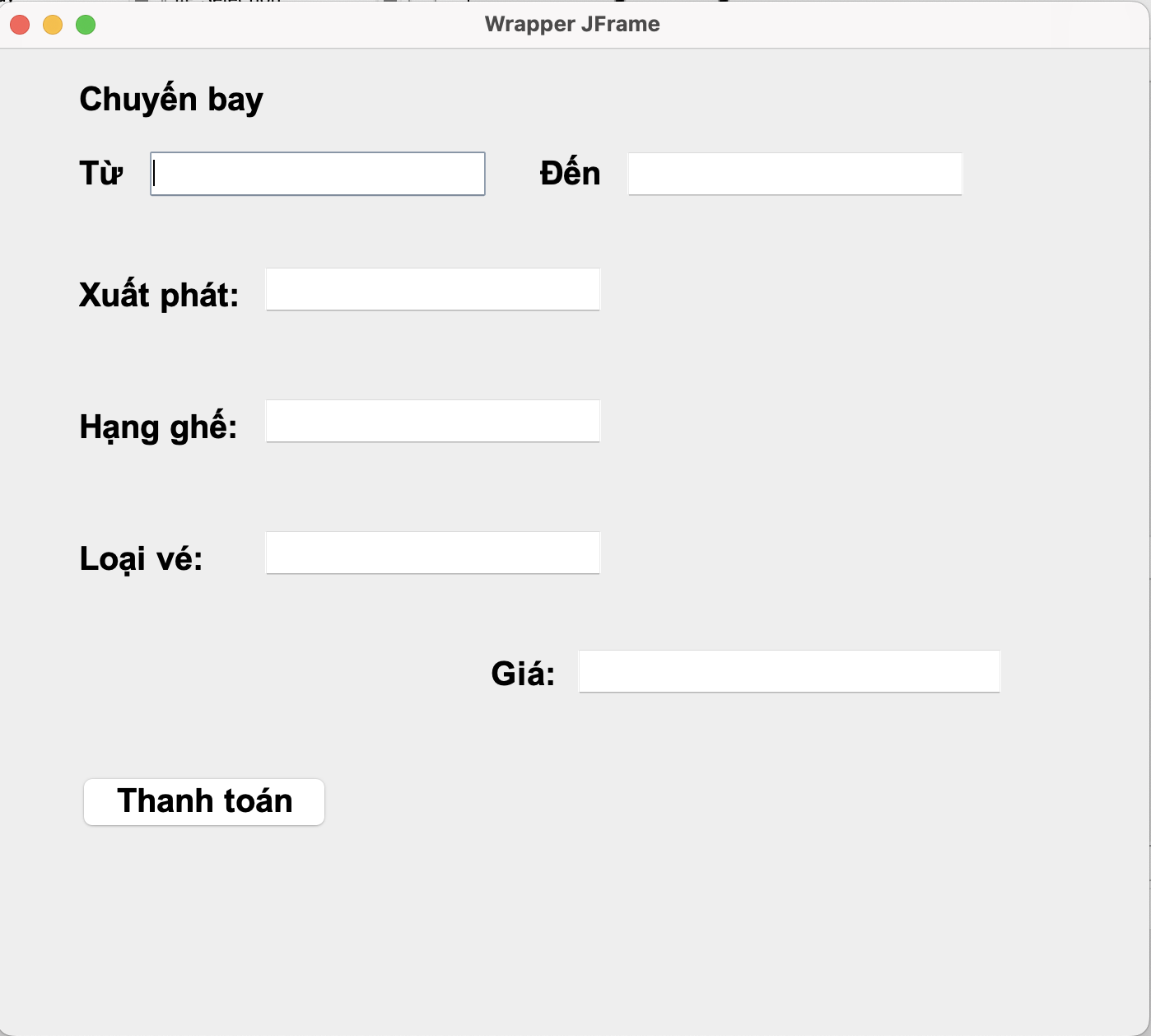
A screenshot of a computer

Description automatically generated

* Để chọn được chuyến mình muốn, người dùng sẽ bấm thẳng vào dòng chứa chuyến muốn đặt trong bảng, hệ thống sẽ tự ghi lại mọi thông tin và chuyển tiếp đến Panel tiếp theo.
* Và để xác nhận chuyến bay muốn chọn, các bạn bấm vào nút “Chọn chuyến” phía dưới để đến bước tiếp theo.



* Để hoàn thiện được việc đặt chuyến, người dùng phải cung cấp đầy đủ thông tin cá nhân và loại vé cũng như hạng ghế mình muốn đặt.
* Ở các trường thông tin yêu cầu người dùng nhập sẽ có các cơ chế kiểm tra để tránh người dùng nhập sai hoặc thiếu thông tin. Ví dụ như số điện thoại phải đủ 10 số theo chuẩn của Việt Nam, hay căn cước cũng phải đủ 12 số…
* Sau khi điền đầy đủ các trường thông tin và bấm “Xác nhận”, dữ liệu người dùng sẽ được đóng gói và chuyển đến panel cuối cùng để thanh toán trước khi đẩy lên cở sở dữ liệu.



* Panel thanh toán này sẽ có tác dụng giúp người dùng kiểm tra lại thông tin đã chọn một lần nữa, các TextField sẽ thể hiện thông tin mà người dùng đã nhập trước đó.
* Ở Panel này dựa trên những lựa chọn của người dùng về điểm đi điểm đến cũng như loại vé và hạng ghế, thì chương trình sẽ tính toán ra số tổng số tiền mà người dùng cần phải thanh toán.

A screen shot of a computer code

Description automatically generated

* Khi bấm “Thanh toán” dữ liệu sẽ được đẩy lên cở sử dữ liệu. Không gian dữ liệu của người dùng sẽ lưu các trường sau:

A screenshot of a chat

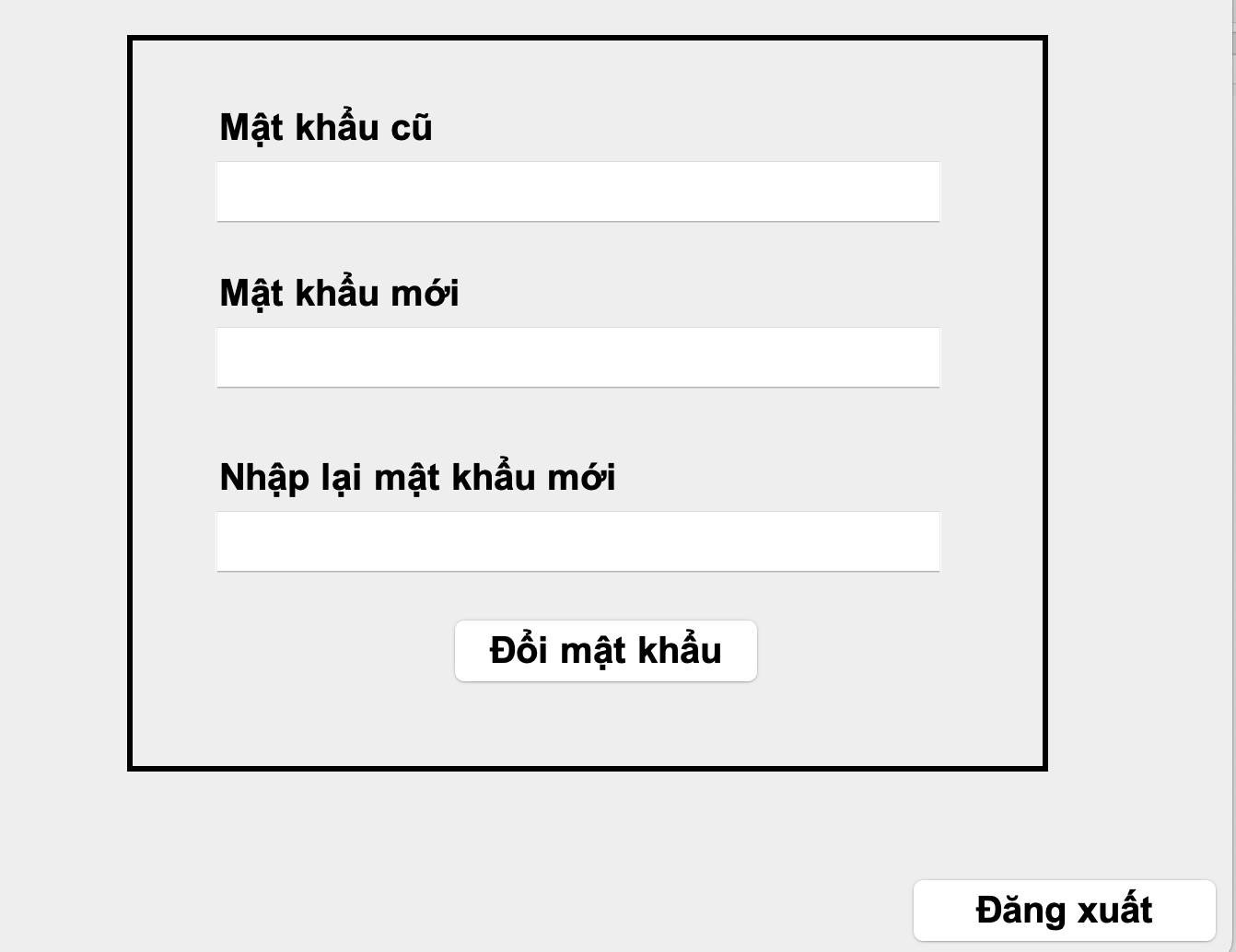
Description automatically generated

* Còn không gian dữ liệu của chuyến bay người dùng đã đăng ký sẽ lưu trữ thông tin cá nhân của người dùng.
* Thứ hai là cách xem các chuyến bay đã đặt:

A screenshot of a computer

Description automatically generated

* Nút “Xem chuyến” sẽ kích hoạt các truy vấn đến cơ sở dữ liệu và tự động tải về toàn bộ các chuyến bay mà người dùng đã đặt, sau đó hiển thị trên “Danh sách chuyến bay”.
* Để huỷ chuyến, người dùng chỉ cần bấm chọn vào chuyến muốn huỷ tương tự như cách làm khi đặt chuyến, hệ thống sẽ ghi lại mã chuyến mà bạn muốn huỷ và tạo truy vấn đến cơ sử dữ liệu và xoá chuyến bay có mã hiệu như bạn đã chọn trong không gian lưu trữ của chính bạn đồng thời xoá luôn thông tin cá nhân của bạn ra khỏi không gian lưu trữ của chuyến bay đó.
* Thứ 3 là giao diện tài khoản:



* Giao diện tài khoản sẽ hỗ trợ tính năng đổi mật khẩu là chính, để đổi mật khẩu các bạn sẽ phải nhập đúng được mật khẩu cũ và nhập mật khẩu hai lần để tránh trường hợp gõ nhầm ký tự.
* Bên cạnh đó, phía dưới sẽ là nút “Đăng xuất” để thoát khỏi chương trình và quay trở lại màn hình đăng nhập ban đầu.

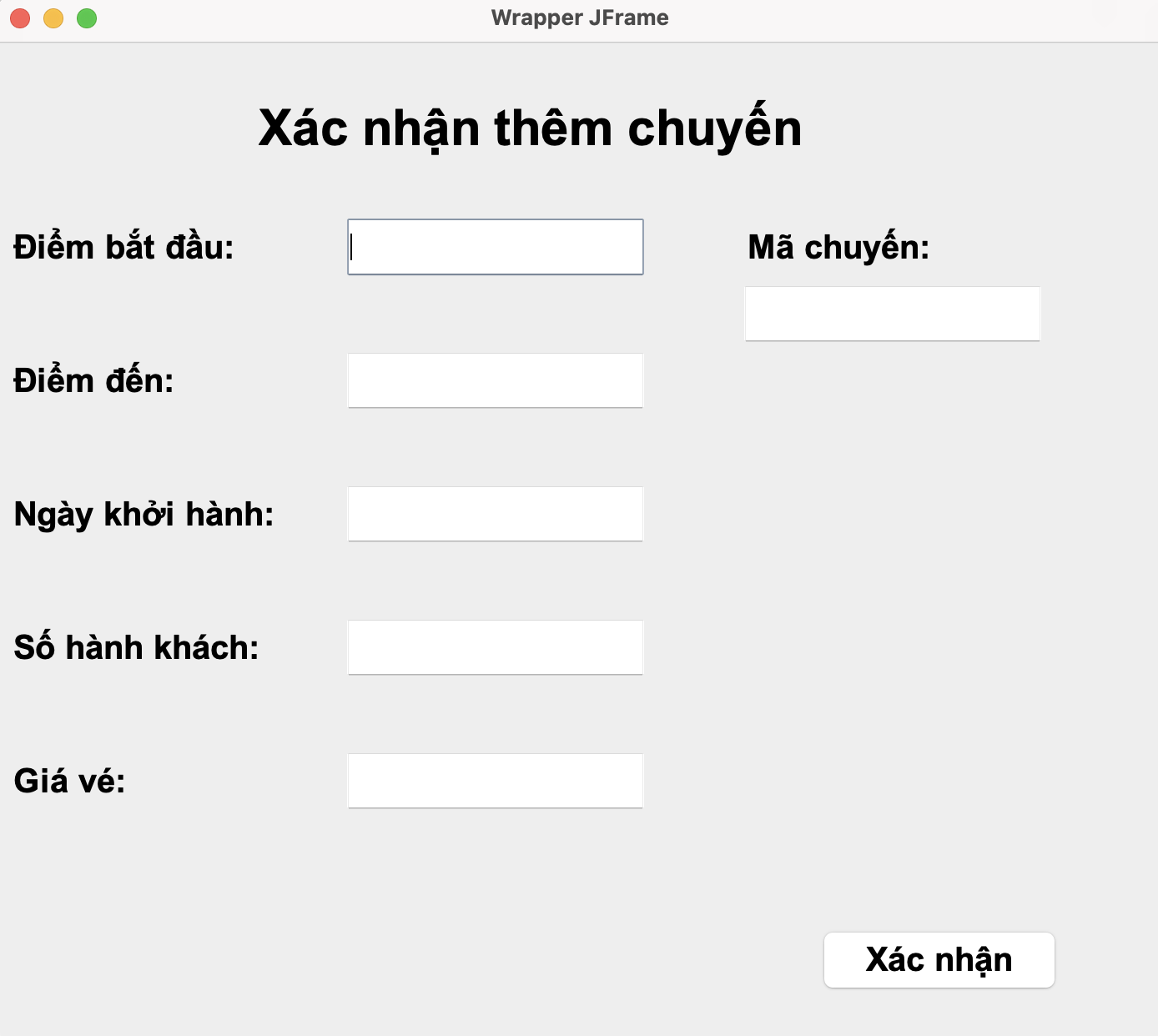
1. **Giao diện sử dụng của tài khoản “Người quản lý”**

* Sự khác biệt của Tài khoản “Người quản lý” chủ yếu nằm ở tính năng thêm chuyến

A screenshot of a computer

Description automatically generated

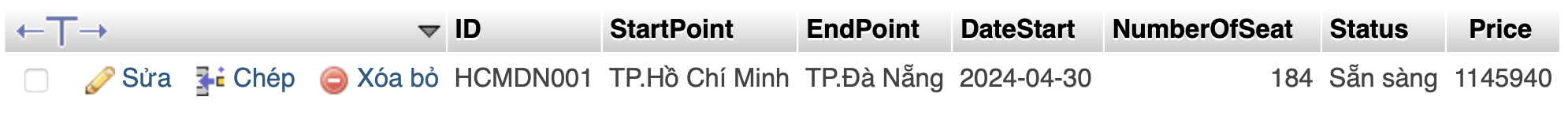
* Nhà quản lý sẽ có công việc thêm vào Database các chuyến bay có sẵn, đã được lên lịch trước để người dùng lựa chọn. Thế nên giao diện thêm chuyến sẽ được biến đổi lại để phục vụ cho tác vụ này.
* Trong các Combox để chọn cũng sẽ có sẵn các bước kiểm tra, tránh trường hợp ngoại lệ tương tự như với tài khoản người dùng: Kiểm tra điểm đi điểm đến, kiểm tra ngày tháng năm của chuyến được thêm xem đã quá ngày hiện tại chưa.
* Khi bấm thêm chuyến, các dữ liệu sẽ được đóng gói và trong mội đối tượng Flight và chuyển tiếp đến Panel Xác nhận.



* Chương trình sẽ phân giải gói vừa nhận được từ Panel trước để điền vào các TextField và dùng các dữ liệu do quản trị viên chọn để tự tính giá và tạo ra mã chuyến.
* Để lấy được giá và mã chuyến, chương trình sẽ tạo ra một truy xuất đến bảng bảng Price, với điều kiện tìm kiếm là điểm đến và điểm đi. Kết quả nhận được chúng ta sẽ có mã chuyến bay và giá của chuyến bay đó:



* Sau khi lấy được mã chuyến, chúng ta sẽ truy xuất vào bảng ListFlight, kiểm tra xem đã có bao nhiêu chuyến được thêm vào trước đó, nếu số chuyển là 15 thì chuyến mới được thêm vào sẽ được mang mã là 016, mã này sẽ ghép với mã chuyến để thành mã chuyến cuối cùng.
* Khi đã có được mã chuyến cuối cùng và giá việc còn lại của chúng ta chỉ còn là đẩy dữ liệu lên cơ sở dữ liệu, cụ thể là bảng ListFlight và kết quả sẽ được như thế này:



* Thêm chuyến là tính năng có mang sự khác biệt lớn nhất giữa tài khoản “Người dùng” và tài khoản “Người quản lý”. Các tính năng khác như xem chuyến hay đổi mật khẩu có cơ chế hoạt động tương tự nên chúng tôi sẽ không trình bày lại nữa.

# CHƯƠNG 4: CÁC TÍNH NĂNG CÓ THỂ ĐƯỢC BỔ SUNG THÊM

1. **Tính năng đổi mật khẩu**

* Hiện tại nhóm đang nghiên cứu về API cho phép gửi tin nhắn SMS cho người dùng. Hệ thống sẽ tạo một mã số gồm 6 chữ và gửi nó đến số điện thoại người dùng đã đăng ký từ trước. Người dùng khi nhận được mã số này sẽ nhập lại vào một TextField, nếu mã số đúng, hệ thống sẽ cho người dùng đặt lại mật khẩu mới.

1. **Tính năng thông báo**

* Cùng với việc sử dụng API cho phép gửi tin nhắn, chúng tôi cũng sẽ xây dụng tính năng thông báo về các chuyến bay sắp đến hạn cho người dùng chuẩn bị theo các giai đoạn: trước 1 ngày, trước 1 tuần…
* Tin nhắn có thể được gửi qua nhiều đường khác nhau, có thể là SMS theo số điện thoại đã được cung cấp hoặc qua Gmail đã được cung cấp.

# DANH MỤC TÀI LIỆU THAM KHẢO

1. Kênh Youtube Bro Code: học về Java Swing, các thiết kế giao diện trong Java;
2. Kênh Youtube Trần Văn Diệp: học về cách kết nối cơ sở dữ liệu.