HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

School of Information and communications technology

System EcoBikeRental

Môn học: <Phân tích thiết kế và xây dựng phần mềm phần mềm>

<Nhóm 6>

<Trần Văn Đạo,Võ Hồng Sang>

*Hanoi,* *<tháng 1, 2021>*

Table Content

[1 System Architecture and Architecture Design 2](#_Toc59358740)

[1.1 Architectural Patterns 2](#_Toc59358741)

[1.2 Interaction Diagrams 2](#_Toc59358742)

[1.3 Analysis Class Diagrams 4](#_Toc59358743)

[1.4 Unified Analysis Class Diagram 6](#_Toc59358744)

[2 Detailed Design 7](#_Toc59358745)

[2.1 User Interface Design 7](#_Toc59358746)

[2.2 Data Modeling 11](#_Toc59358747)

[2.2.1 Conceptual Data Modeling 11](#_Toc59358748)

[2.2.2 Database Design 11](#_Toc59358749)

[2.3 Class Design 13](#_Toc59358750)

[2.3.1 General Class Diagram 13](#_Toc59358751)

[2.3.2 Class Diagrams 14](#_Toc59358752)

[2.3.3 Class Design 14](#_Toc59358753)

# System Architecture and Architecture Design

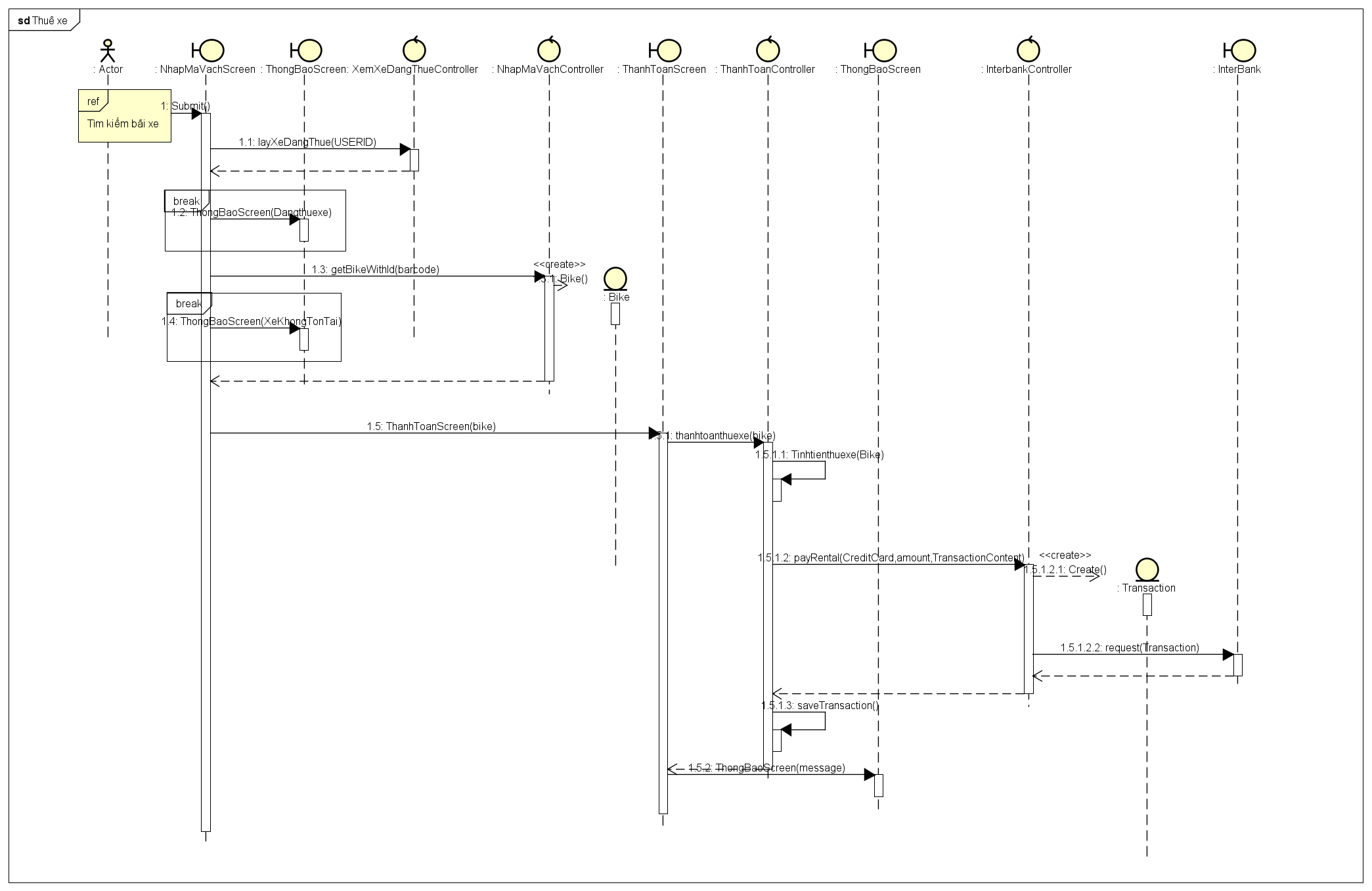
## Architectural Patterns

- Mô hình được lựa chọn là mô hình 3 lớp

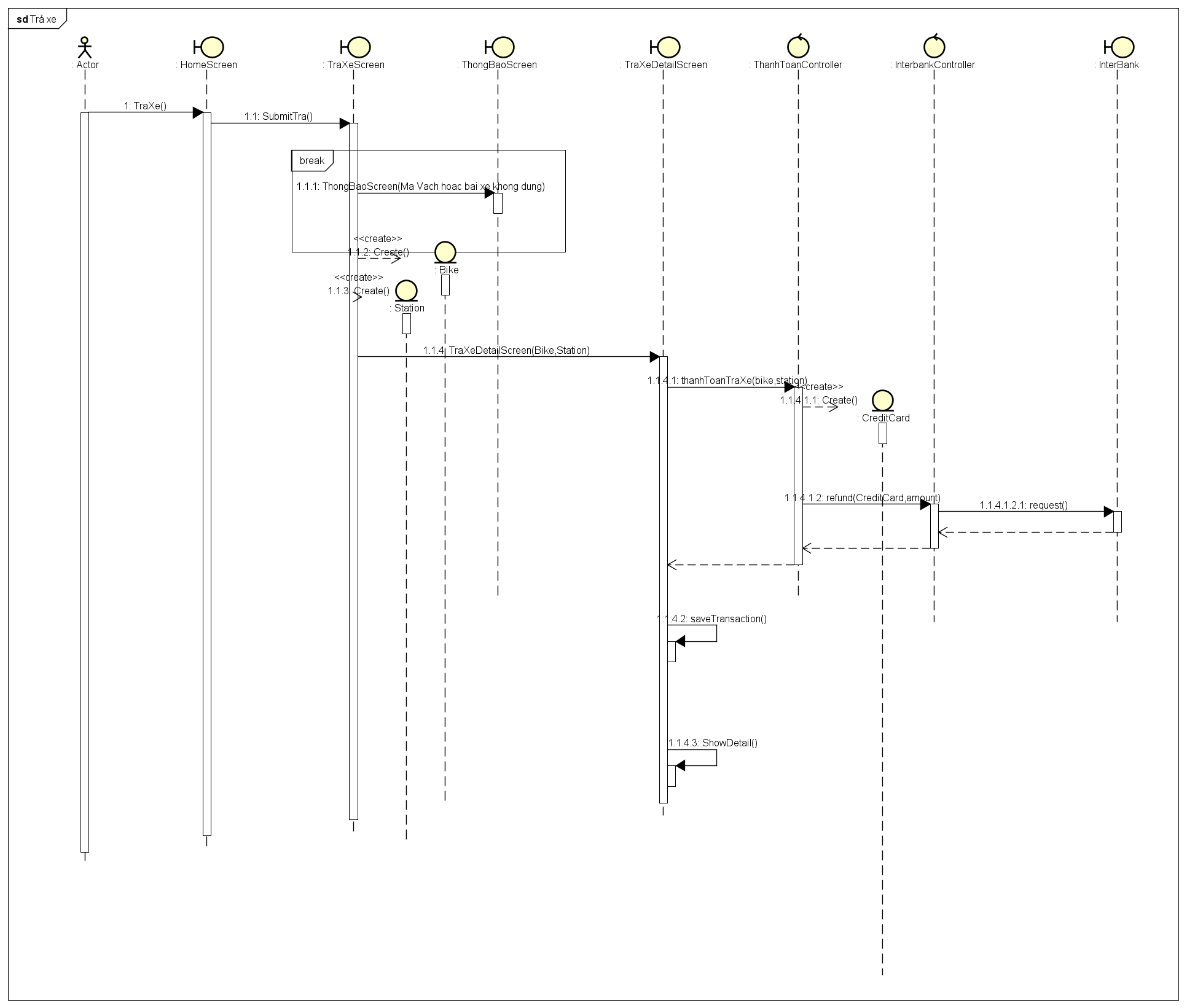
- Ứng dụng đòi hỏi có sự tách biệt 3 phần: giao diện, xử lý nghiệp vụ, giao tiếp với hệ quản trị CSDL để người viết có thể dễ dàng quản lý ứng dụng của mình khi có bug xảy ra với 1 số thành phần xử lý nghiệp vụ không mong muốn. Ngoài ra với mô hình này nó còn tạo ra 1 không gian làm việc rất tốt để người thiết kế giao diện, lẫn người lập trình có thể làm việc chung với nhau 1 cách dễ dàng.

- Khả năng tái tạo cao: Khi ứng dụng bất chợt yêu cầu thay đổi hệ quản trị CSDL hoặc chuyển ứng dụng từ window application sang web application, việc xây dựng lại ứng dụng từ đầu rất tốn nhiều thời gian và chi phí. Vì vậy với mô hình này ra đời sẽ giải quyết vấn đề này.

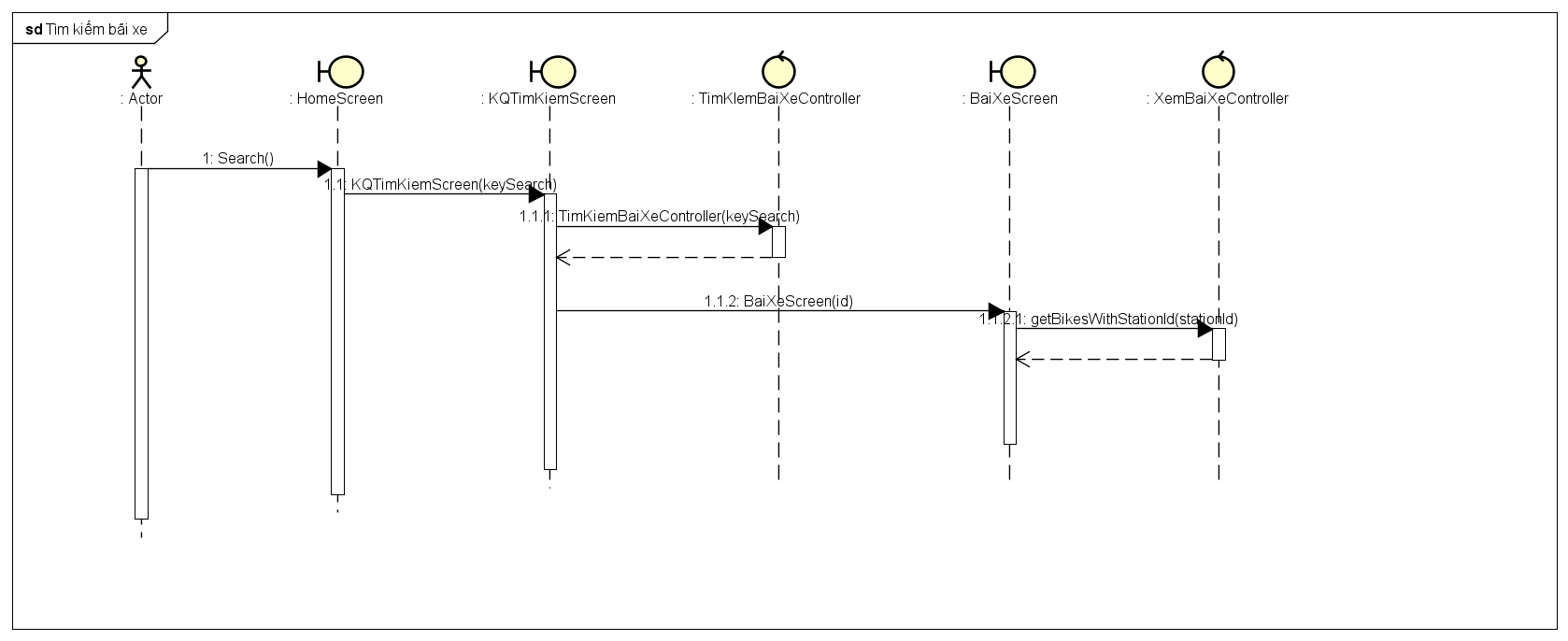
## Interaction Diagrams

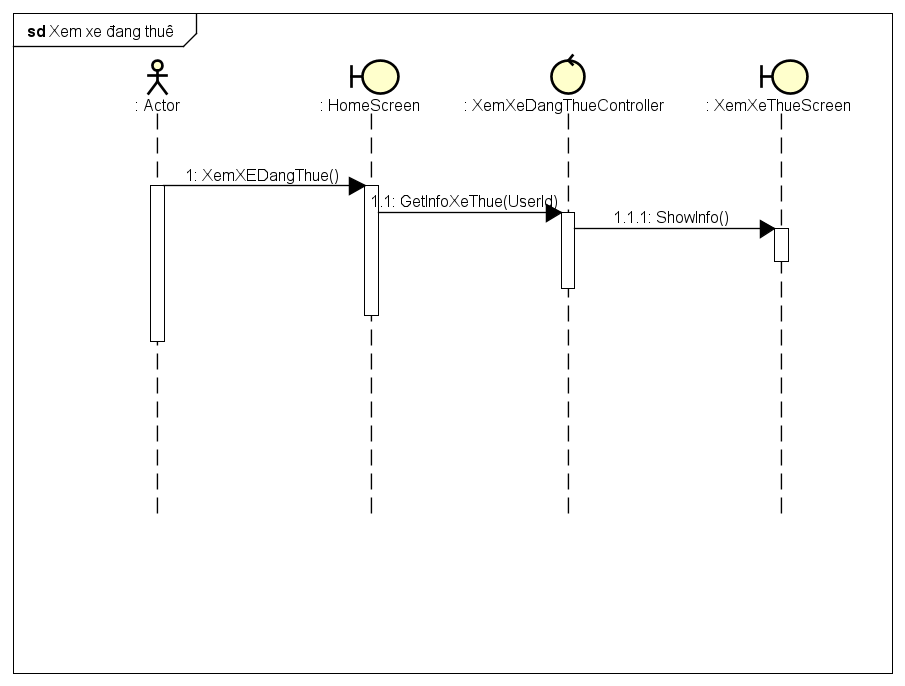


Thuê xe



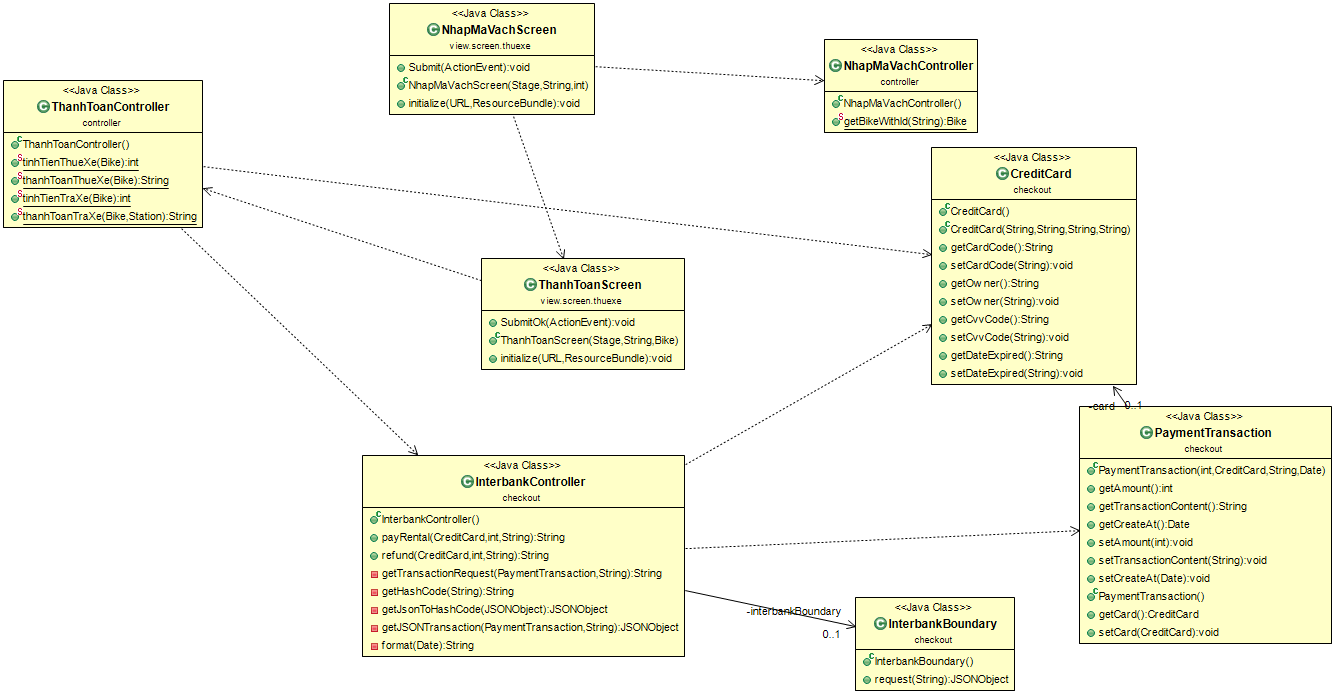
Trả xe



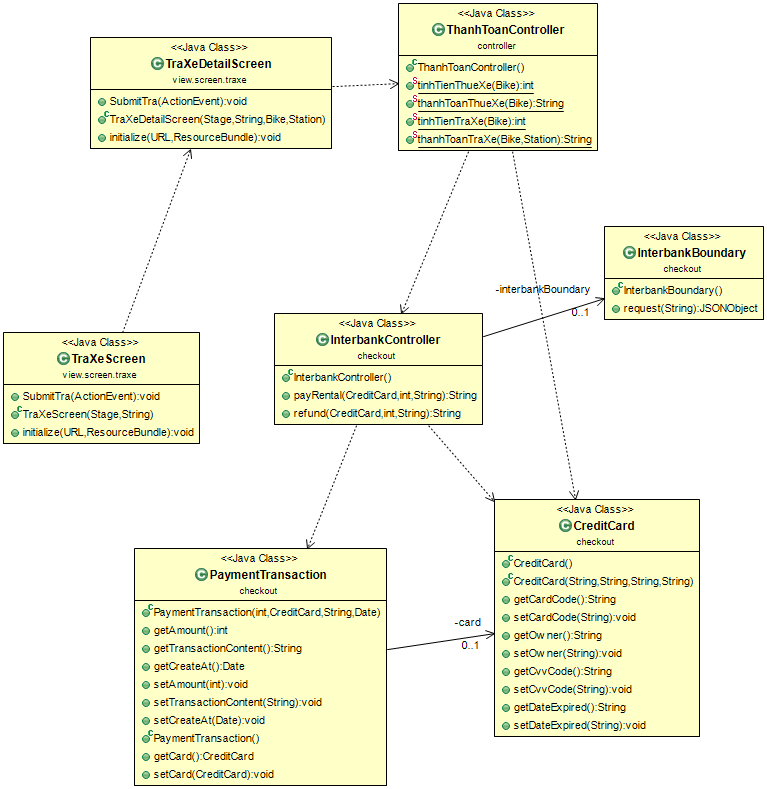
Tìm kiếm bãi xe

Xem xe đang thuê

## Analysis Class Diagrams

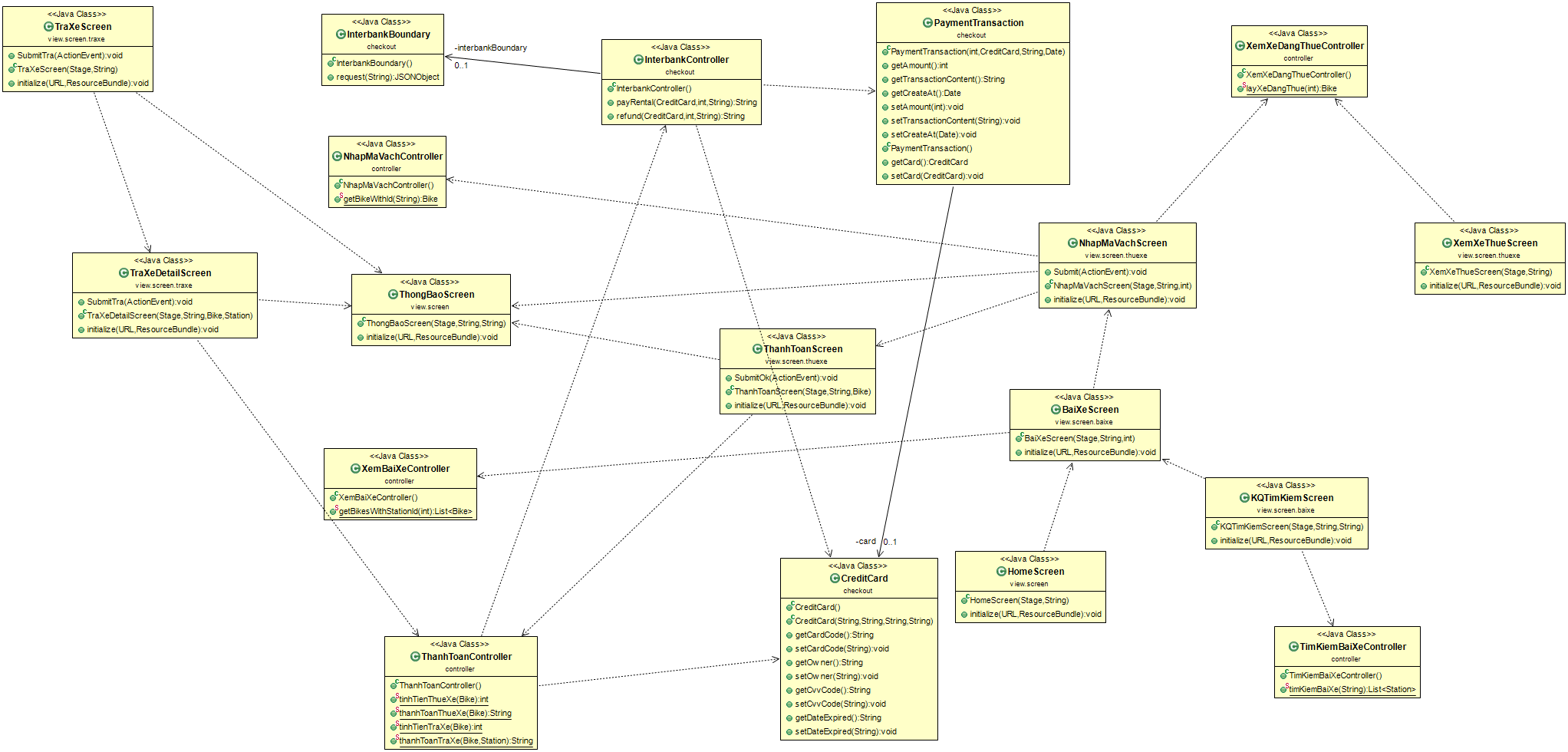


Thuê xe



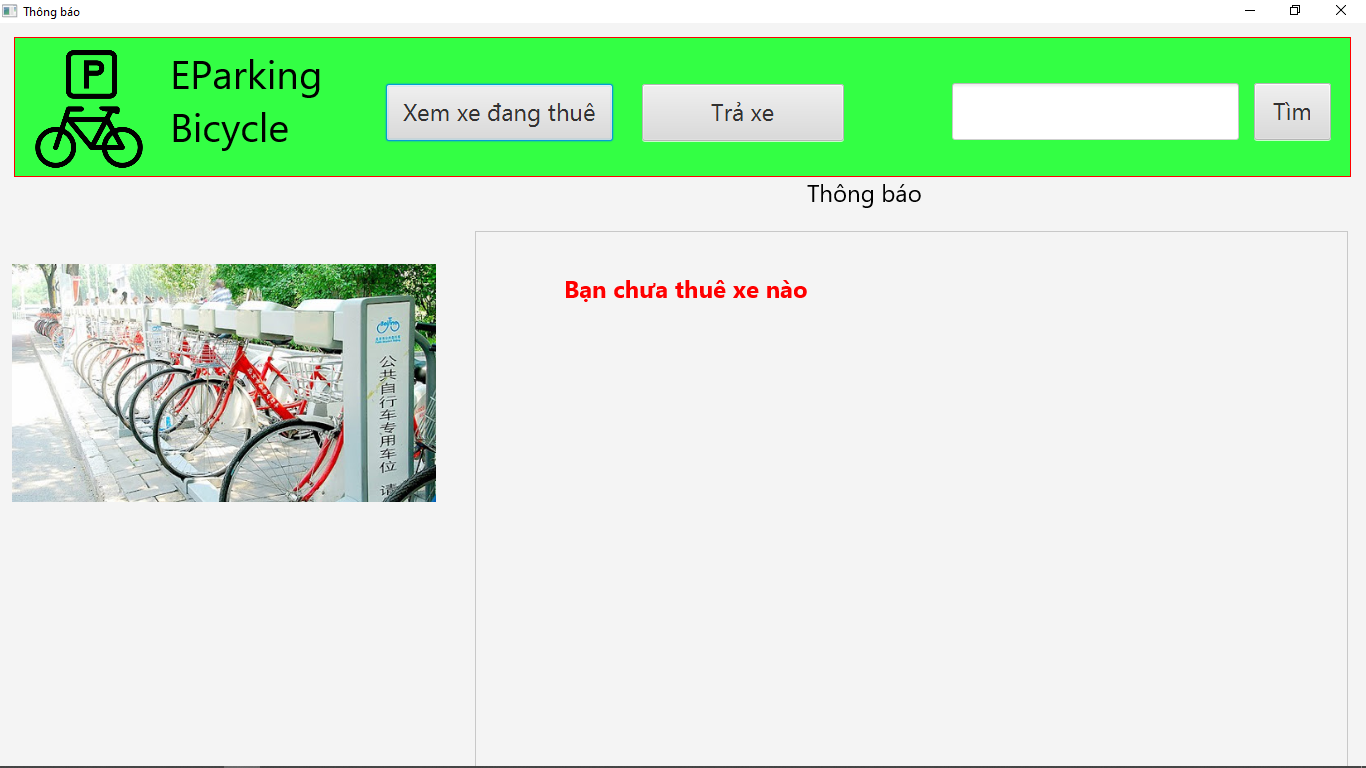
Trả xe

## Unified Analysis Class Diagram

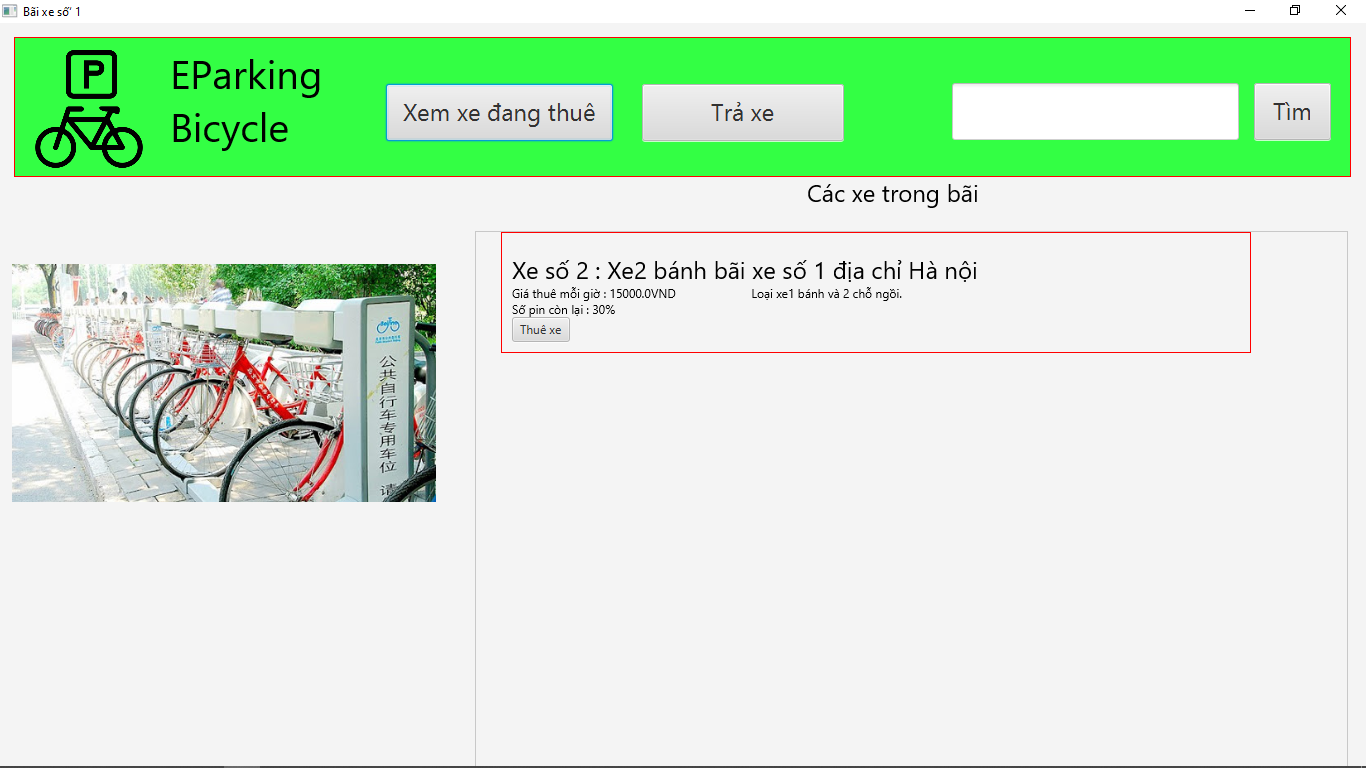


# Detailed Design

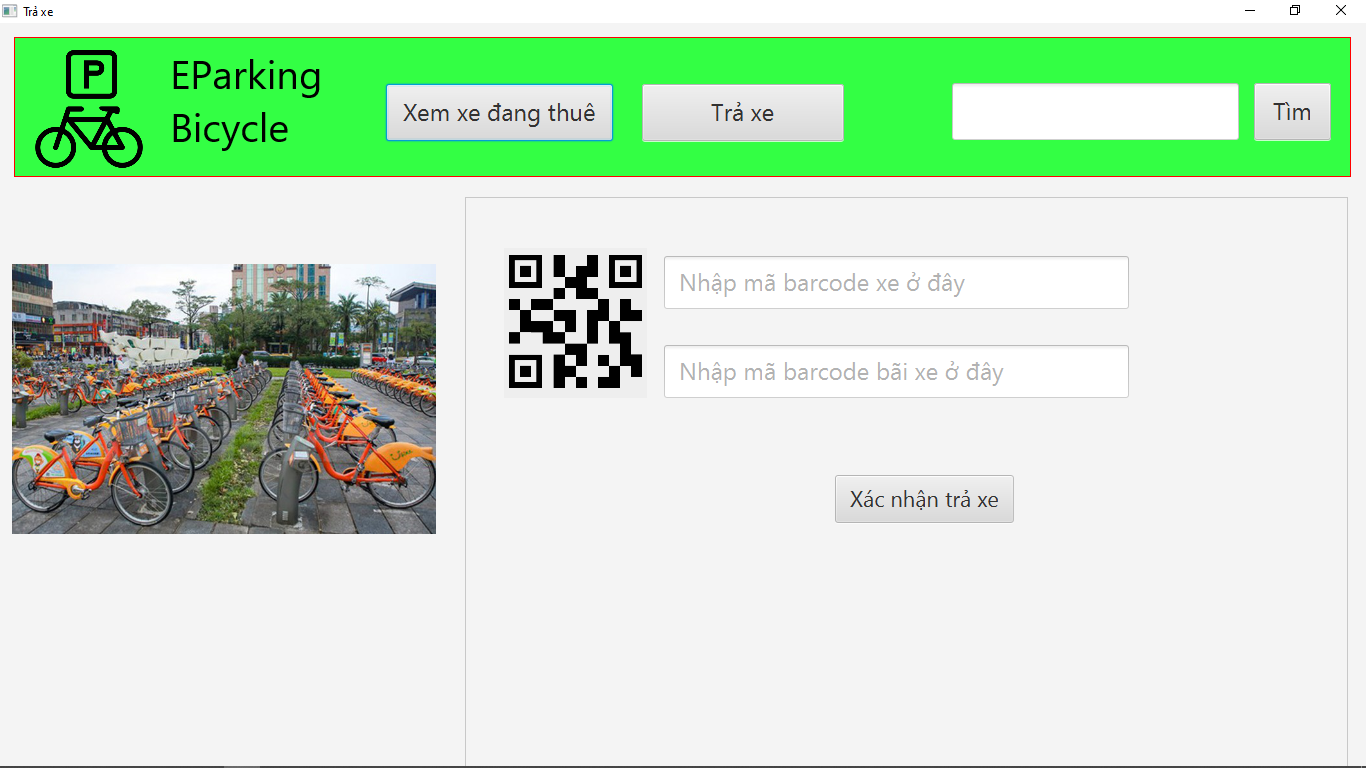
## User Interface Design

**

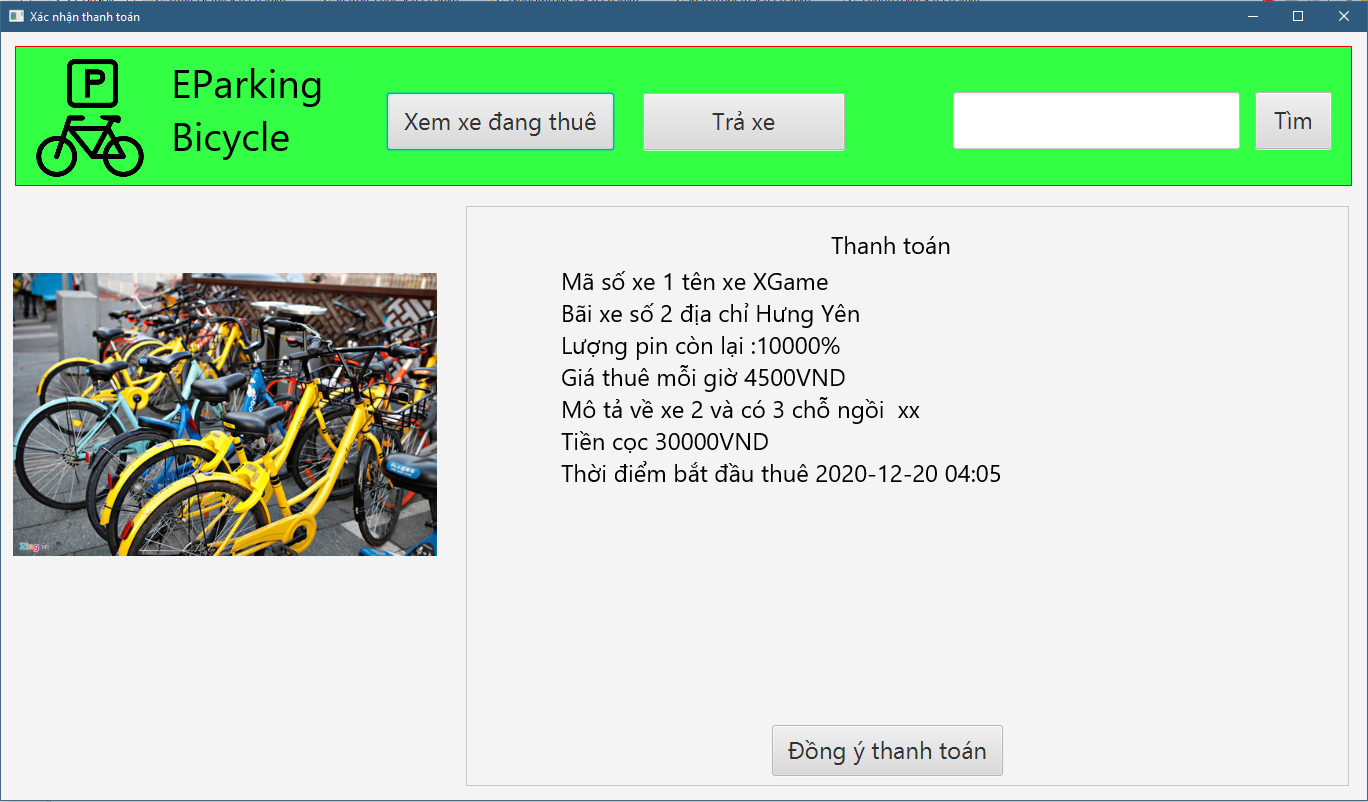
*Màn hình thông báo*

**

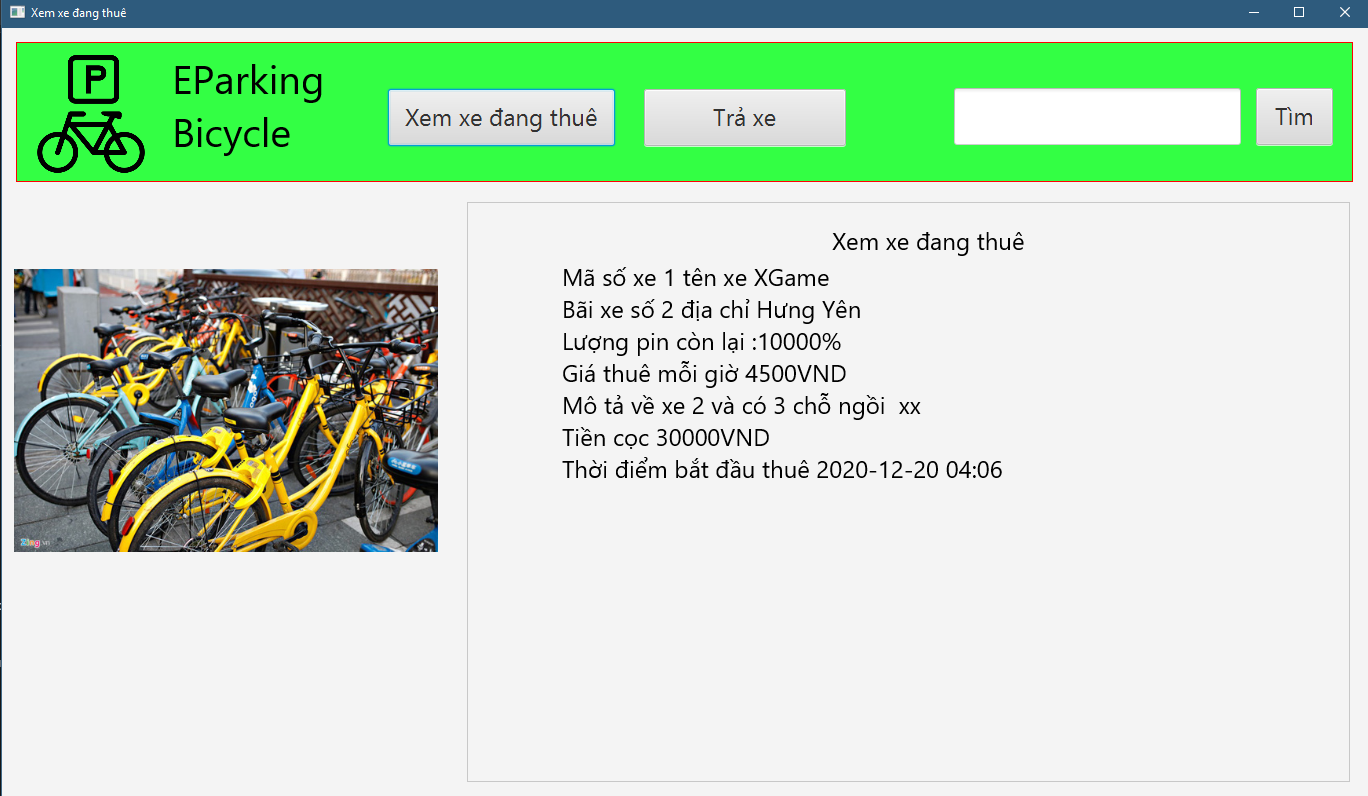
*Màn hình xem xe trong bãi*

**

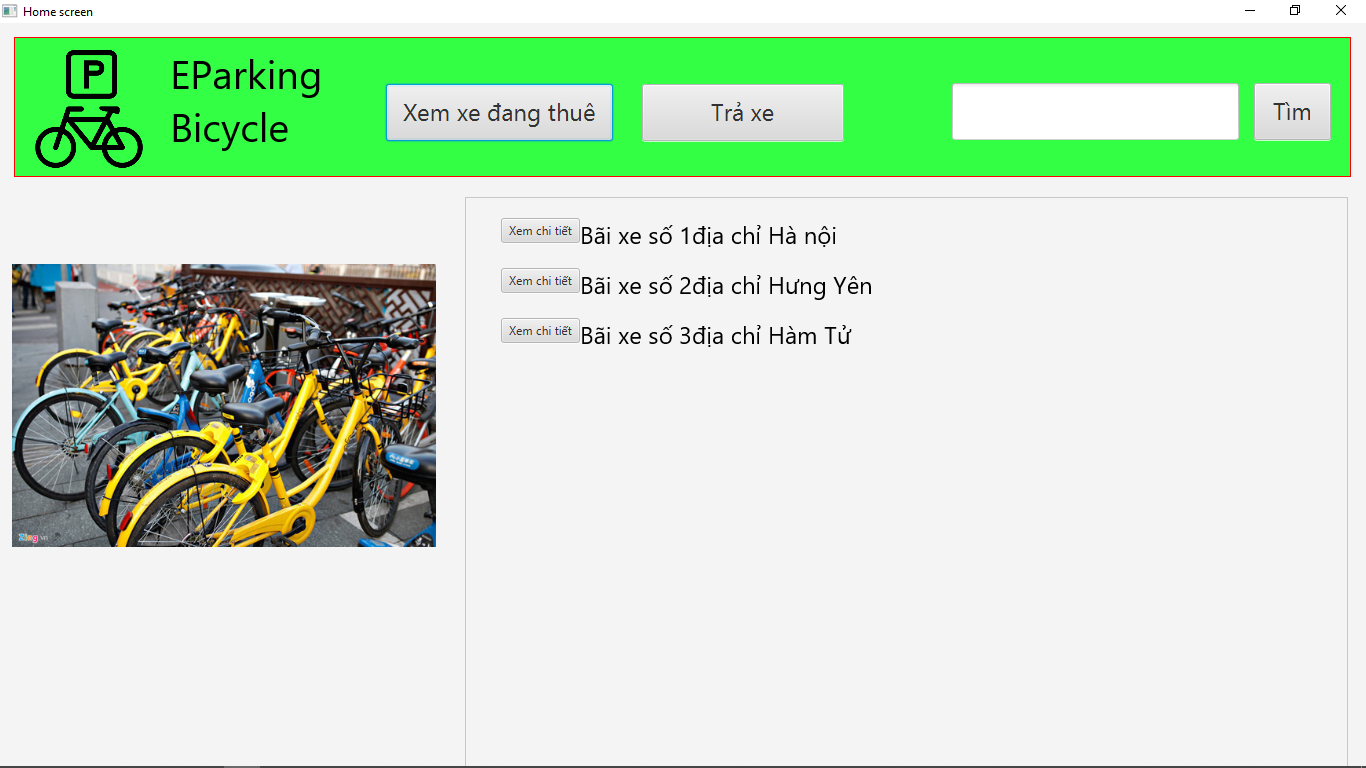
*Màn hình Trả Xe*

**

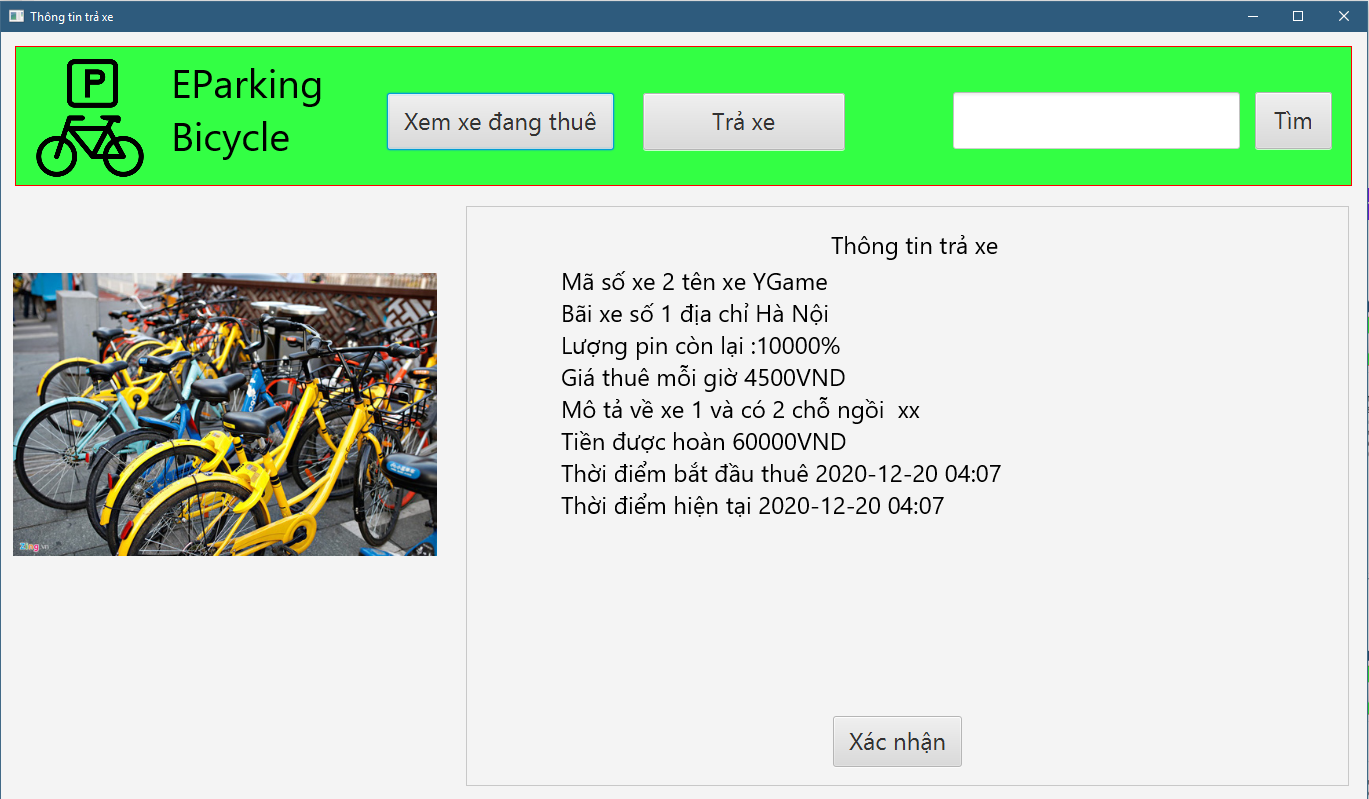
*Màn hình xác nhận thanh toán*

**

*Màn hình xem xe đang thuê*

**

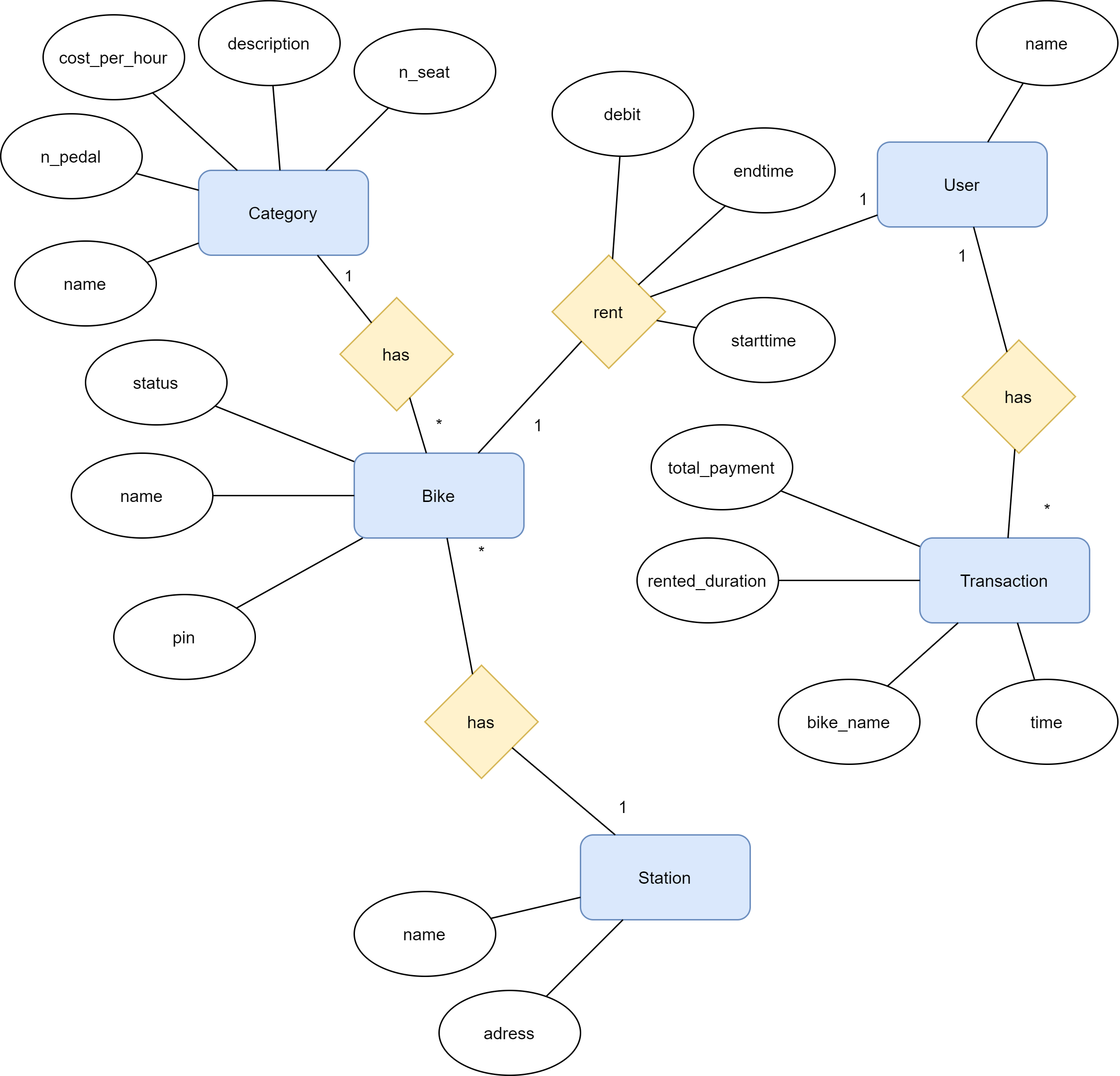
*Màn hình chính*

**

*Màn hình thông tin trả xe*

## Data Modeling

### Conceptual Data Modeling



### Database Design

#### Database Management Systems

*Hệ quản trị dữ liệu được dung là MySQL*

#### Logical Data Model

#### Physical Data Model

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *#* | *Pk* | *FK* | *Tên* | *Kiểu dữ liệu* | *Giá trị mặc định* | *Mô tả* |
| 1 | x |  | bikeId | Int | Null | ID của xe |
| 2 |  |  | bikeName | Varchar(255) | Null | Tên xe |
| 3 |  |  | pin | Float | Null | Phần trăm pin còn lại |
| 4 |  |  | status | Int | Null | Trạng thái thuê của xe |
| 5 |  | x | CategoryId | Int | Null | Loại xe |
| 6 |  |  | Price | Int | Null | Giá của xe |
| 7 |  | x | stationId | int | Null | Vị trí của xe |

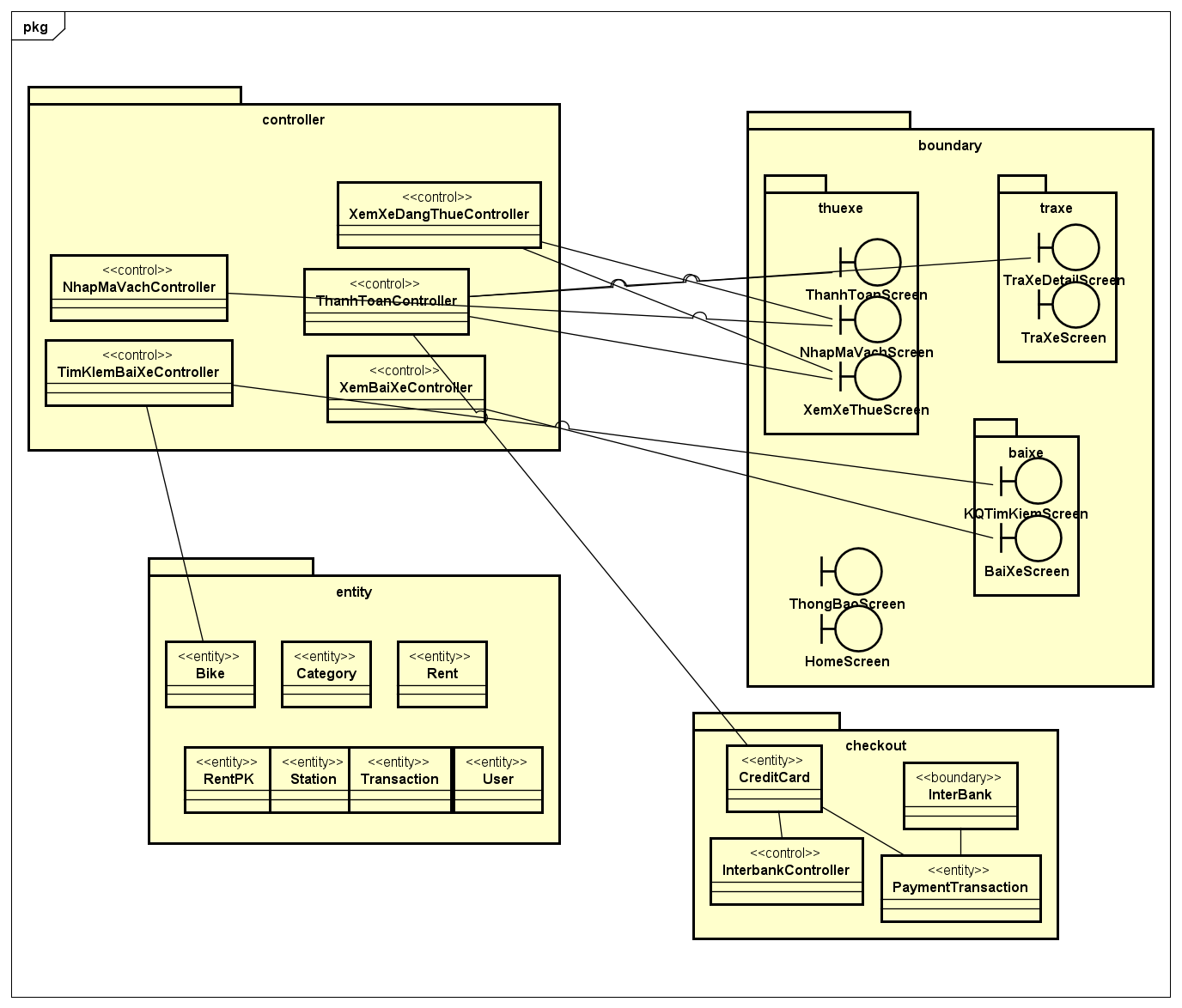
Table Bike

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *#* | *Pk* | *FK* | *Tên* | *Kiểu dữ liệu* | *Giá trị mặc định* | *Mô tả* |
| 1 | x |  | CategoryId | Int | Null | Id của loại xe |
| 2 |  |  | name | Varchar(255) | Null | Tên của loại xe |
| 3 |  |  | Description | Varchar(255) | Null | Miêu tả về loại xe |
| 4 |  |  | Cost\_per\_hour | Int | Null | Giá thuê xe trên giờ |
| 5 |  |  | N\_pedal | Int | Null | Số bàn đạp |
| 6 |  |  | N\_seats | Int | Null | Số chỗ ngồi |

Table Category

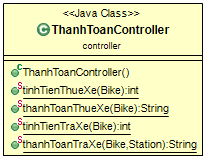
## Class Design

### General Class Diagram



### Class Design

#### Class ThanhToanController



**Operation**

|  |  |  |  |
| --- | --- | --- | --- |
| *#* | *Name* | *Return type* | *Description (purpose)* |
| 1 | tinhTienThueXe | int | Tính tiền thuê xe |
| 2 | thanhToanThueXe | String | Gửi thông tin thuê xe cho InterBankController |
| 3 | tinhTienTraXe | Int | Tính tiền trả xe |
| 4 | thanhToanTraXe | String | Gửi thông tin trả xe cho InterBankController |

*Parameter*:

* bike:Bike xe đang được thanh toán
* station: Station bến xe đang được thanh toán

*Exception*:

* Không

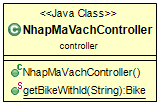
**Method**

* Không

**State**

* Không

#### Class NhapMaVachController



**Operation**

|  |  |  |  |
| --- | --- | --- | --- |
| *#* | *Name* | *Return type* | *Description (purpose)* |
| 1 | getBikeWithId | Bike | Lấy thông tin xe từ Database qua ID |

*Parameter*:

* String:String - id của xe

*Exception*:

* Mã Id của xe không có trong database

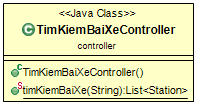
**Method**

* Không

**State**

* Không

#### Class TimKiemBaiXe



**Operation**

|  |  |  |  |
| --- | --- | --- | --- |
| *#* | *Name* | *Return type* | *Description (purpose)* |
| 1 | timKiemBaiXe | List<Station> | Tìm kiếm bãi xe |

*Parameter*:

* *String:String – tên của bãi xe*

*Exception*:

* String nhập vào không có dữ liệu phù hợp trong Database

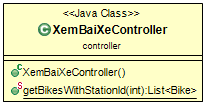
**Method**

* Không

**State**

* Không

#### Class XemBaiXeController



**Operation**

|  |  |  |  |
| --- | --- | --- | --- |
| *#* | *Name* | *Return type* | *Description (purpose)* |
| 1 | getBikéWithStationId | List<Bike> | Hiện các xe có thể thuê trong bãi |

*Parameter*:

* Int:int – id của bãi xe

*Exception*:

* Không

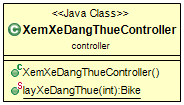
**Method**

* Không

**State**

* Không

#### Class XemXeDangThue



**Operation**

|  |  |  |  |
| --- | --- | --- | --- |
| *#* | *Name* | *Return type* | *Description (purpose)* |
| 1 | layXeDangThue | Bike | Lấy thông tin xe đang thuê |

*Parameter*:

* Int:int – id của xe

*Exception*:

* Không

**Method**

* Không

**State**

* Không

# Design Considerations

## Coupling and Cohesion

* Nhóm đang Control coupling. Module controller điều khiển các thao tác trong module boundary

## Design Principles

## Design Patterns