

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN 1**

**MÔN NHẬP MÔN CÔNG NGHỆ PHẦN MỀM**

---



**BÁO CÁO**  
**HỆ THỐNG QUẢN LÝ CỬA HÀNG CHO THUÊ XE**  
**Ô TÔ**

**Lớp học phần: N08**

**Số thứ tự nhóm: 02**

**Modul: Nhận trả xe từ khách hàng**

**Chủ đề: Báo cáo tổng hợp**

Trịnh Ngọc Anh

MSSV: B22DCAT023

Trương Gia Bách

MSSV: B22DCAT024

**Lê Văn Minh**

**MSSV: B22DCCN533**

Nguyễn Huy Trung

MSSV: B22DCAT307

***Hà Nội – 2025***

# MỤC LỤC

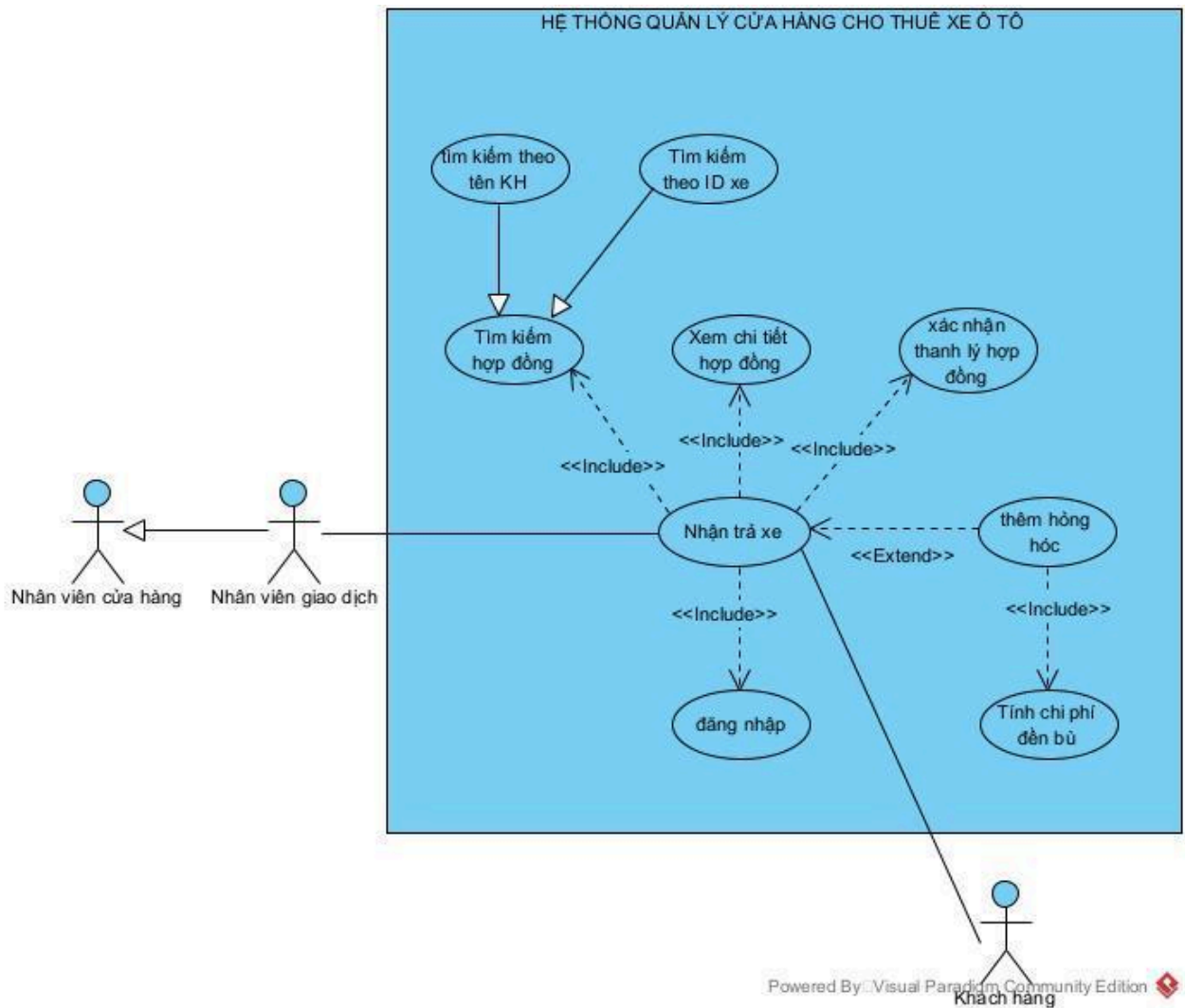
1. Biểu đồ UseCase chi tiết cho modul và mô tả các UC của modul.....	3
2. Kịch bản chuẩn của modul.....	4
3. Biểu đồ thực thể pha phân tích của modul.....	6
4. Biểu đồ lớp đầy đủ pha phân tích của modul.....	9
5. Biểu đồ tuần tự pha phân tích của modul.....	11
6. Biểu đồ thiết kế lớp thực thể của modul.....	15
7. Biểu đồ thiết kế CSDL của modul.....	15
8. Thiết kế giao diện và biểu đồ lớp thiết kế chi tiết đầy đủ của modul.....	17
9. Biểu đồ tuần tự pha thiết kế của modul.....	26
10. Test plan và test case chuẩn cho test hộp đen của modul.....	31

## 1. Biểu đồ UseCase chi tiết cho modul và mô tả các UC của modul

Quy trình nhận trả xe bao gồm việc nhân viên thực hiện thanh lý hợp đồng thuê xe với khách hàng, kiểm tra các thông tin liên quan đến hợp đồng, tình trạng xe, các khoản thanh toán và xử lý các hỏng hóc nếu có.

### Mô tả các use case:

- **Giao diện đăng nhập:** Giao diện cho phép nhân viên đăng nhập vào hệ thống.
- **Giao diện tìm kiếm hợp đồng:** Giao diện này cho phép nhân viên chọn chức năng tìm kiếm hợp đồng cần thanh lý với khách hàng. Hệ thống sẽ hiển thị danh sách các hợp đồng thuê xe chưa thanh lý để nhân viên tìm kiếm.
- **Giao diện chi tiết hợp đồng:** Use case này cho phép nhân viên xem thông tin chi tiết của hợp đồng thuê xe, bao gồm:
  - Thông tin khách hàng
  - Danh sách xe thuê (có thể nhiều xe)
  - Ngày thuê, ngày trả xe
  - Đơn giá thuê, tổng tiền thuê
  - Tiền cọc đã nhận, số tiền còn lại cần thanh toán
  - Danh sách các tài sản đảm bảo
- **Giao diện thêm hỏng hóc (nếu có):** Use case này cho phép nhân viên kiểm tra tình trạng xe sau khi khách trả. Nếu có hỏng hóc, nhân viên có thể tìm kiếm và chọn các hỏng hóc từ danh sách có sẵn hoặc thêm mới các hỏng hóc cần đền bù. Hệ thống sẽ tính toán tổng số tiền thanh toán bao gồm cả chi phí đền bù (nếu có).
- **Giao diện xác nhận hóa đơn Thanh lý hợp đồng:** Use case này cho phép nhân viên xác nhận thanh lý hợp đồng sau khi kiểm tra đầy đủ thông tin và số tiền thanh toán. Hệ thống sẽ lưu kết quả vào cơ sở dữ liệu, in hóa đơn để nhân viên và khách hàng ký xác nhận, thực hiện thanh toán, bàn giao lại tài sản đảm bảo và hoàn tất quá trình nhận trả xe.



## 2. Kịch bản chuẩn của modul

Scenario	Nhận trả xe
Actor(s)	Nhân viên giao dịch, khách hàng
Pre-condition	Nhân viên giao dịch cần có tài khoản của nhân viên giao dịch
Post-condition	Nhân viên giao dịch thanh lý hợp đồng thành công
Main events	<ol style="list-style-type: none"> <li>Nhân viên giao dịch A đăng nhập vào hệ thống với username: A, password: 123456.</li> <li>Hệ thống hiện giao diện của nhân viên giao dịch, nhân viên A click vào chức năng "Thanh lý hợp đồng". Nhân viên A muốn thực hiện quy trình nhận lại xe từ khách hàng B, người đang đến cửa hàng để trả xe.</li> </ol>

3. Hệ thống hiển thị giao diện danh sách các hợp đồng thuê xe chưa thanh lý.

ID hợp đồng	Tên KH	Ngày thuê	Ngày trả dự kiến	Xe đang thuê
1234	Nguyễn Văn B	01/03/2025	13/03/2025	Toyota Vios
1235	Hà Văn A	24/02/2025	01/03/2025	Mazda 3
1236	Lê Văn C	03/03/2025	19/03/2025	Hyundai

4. Nhân viên A hỏi khách hàng B về hợp đồng thuê xe cần thanh lý để có thể tìm kiếm.
5. Khách hàng B cung cấp thông tin hợp đồng, xác nhận đang trả xe theo hợp đồng số 1234.
6. Nhân viên A nhập ID hợp đồng và click vào nút “tìm kiếm hợp đồng”.
7. Hệ thống hiển thị ra hợp đồng cần tìm, nhân viên bấm chọn vào nút xem chi tiết thanh lý hợp đồng, bao gồm:

KH	Xe	Ngày thuê	Ngày trả dự kiến	Đơn giá	Thành tiền	Tiền cọc	Tài sản đảm bảo	Còn thiếu
Nguyễn Văn B	Toyota Vios (tình trạng tốt không hỏng hóc)	01/03/2025	13/03/2025	3.000.000 đ	12.000.000 đ	3.000.000 đ	Xe máy vision	9.000.000 đ

8. Nhân viên A ra khu vực giao nhận xe cùng khách hàng B, đối chiếu với thông tin tình trạng xe khi nhận. Kiểm tra tổng thể xe theo các tiêu chí:
- Hình dáng tổng thể, màu sơn.
  - Kính chắn gió, đèn pha, đèn hậu, gương chiếu hậu.
  - Nội thất: Ghế, bảng điều khiển, điều hòa.
  - Động cơ, lốp xe, phanh, nhiên liệu còn lại.
9. Nếu xe có hỏng hóc hoặc vi phạm, nhân viên A:
- Chụp ảnh hiện trạng xe, ghi chú lỗi phát sinh.

	<ul style="list-style-type: none"><li>○ Chọn hoặc nhập lỗi phát sinh vào hệ thống.</li><li>○ Xác định mức phạt cho từng lỗi.</li><li>○ Hệ thống tự động tính tổng số tiền bồi thường.</li></ul> <p>10. Nhân viên A thông báo tổng số tiền khách hàng cần thanh toán bao gồm:</p> <ul style="list-style-type: none"><li>○ Số tiền thuê xe còn lại.</li><li>○ Tiền phạt do hỏng hóc (nếu có).</li><li>○ Tổng số tiền cần thanh toán sau khi trừ tiền đặt cọc.</li></ul> <p>11. Khách hàng B xác nhận số tiền cần thanh toán và tiến hành thanh toán.</p> <p>12. Nhân viên A click vào nút "Xác nhận thanh lý hợp đồng".</p> <p>13. Hệ thống lưu thông tin thanh lý hợp đồng, in hóa đơn thanh lý hợp đồng bao gồm:</p> <table><tr><th>Mã hợp đồng</th><th>Tên KH</th><th>Xe đã thuê</th><th>Ngày thuê</th><th>Ngày trả</th><th>Đơn giá / ngày</th><th>Tiền cọc</th><th>Tổng tiền</th><th>Phương thức thanh toán</th><th>Phí phát sinh</th></tr><tr><td>1234</td><td>Nguyễn Văn B</td><td>Toyota Vios</td><td>01/03 /2025</td><td>13/03 /2025</td><td>1.000 .000đ</td><td>3.000 .000đ</td><td>12.000 .000 đ</td><td>CK</td><td>Không</td></tr></table> <p>14. Nhân viên A giao hóa đơn cho khách hàng B để ký xác nhận.</p> <p>15. Nhân viên A trả lại tài sản đảm bảo cho khách hàng B.</p> <p>16. Nhân viên A hoàn tất quy trình, xác nhận xe đã được trả, kết thúc hợp đồng trong hệ thống, nhấn nút “Hoàn tất”. Hệ thống đánh dấu hợp đồng đã hoàn tất và loại hợp đồng số 1234 khỏi danh sách hợp đồng chưa thanh lý.</p>	Mã hợp đồng	Tên KH	Xe đã thuê	Ngày thuê	Ngày trả	Đơn giá / ngày	Tiền cọc	Tổng tiền	Phương thức thanh toán	Phí phát sinh	1234	Nguyễn Văn B	Toyota Vios	01/03 /2025	13/03 /2025	1.000 .000đ	3.000 .000đ	12.000 .000 đ	CK	Không
Mã hợp đồng	Tên KH	Xe đã thuê	Ngày thuê	Ngày trả	Đơn giá / ngày	Tiền cọc	Tổng tiền	Phương thức thanh toán	Phí phát sinh												
1234	Nguyễn Văn B	Toyota Vios	01/03 /2025	13/03 /2025	1.000 .000đ	3.000 .000đ	12.000 .000 đ	CK	Không												
Exception	<p>1. Hệ thống báo sai tài khoản mật khẩu đăng nhập.</p> <p>6. Không tìm thấy kết quả.</p> <p>9. Nhân viên A click vào nút bổ sung chi phí cho các phần hỏng hóc.</p> <p>11. Khách hàng B không đồng ý với mức đền bù hỏng hóc.</p>																				

### 3. Biểu đồ thực thể pha phân tích của modul

Hệ thống quản lý thông tin về hợp đồng thuê xe, thông tin khách hàng thuê xe và các hoạt động liên quan đến quá trình nhận xe trả về từ khách hàng. Hệ thống cho phép nhân viên giao dịch thực hiện các nghiệp vụ liên quan đến thanh lý hợp đồng thuê xe. Khi khách hàng đến trả xe, nhân viên giao dịch chọn chức năng thanh lý hợp đồng, hệ thống hiển thị danh sách các hợp đồng thuê xe chưa được thanh lý. Nhân viên tìm và chọn hợp đồng tương ứng với xe mà khách hàng muốn trả. Hệ thống sau đó hiển thị chi tiết thông tin hóa đơn thanh toán bao gồm thông tin khách hàng, thông tin các xe thuê, ngày thuê, ngày trả, đơn giá thuê, tổng số tiền thuê, số

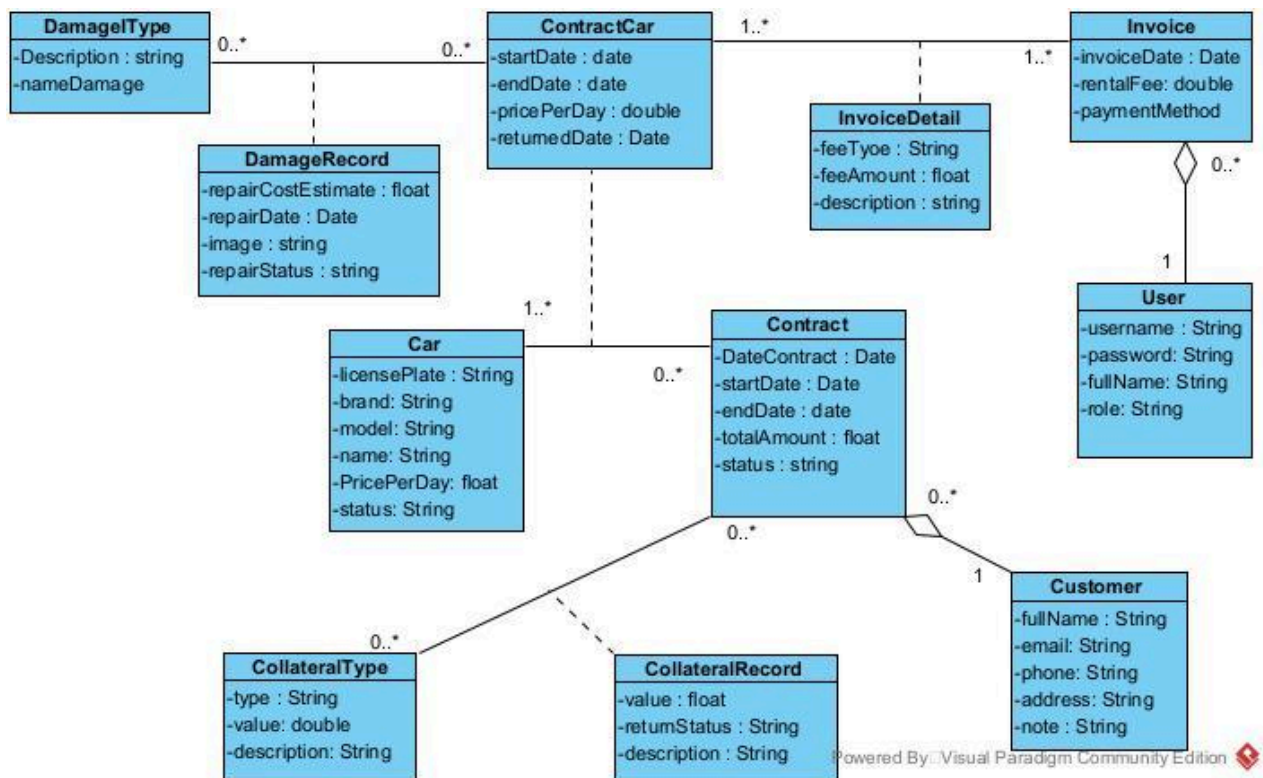
tiền đặt cọc đã nhận, số tiền còn lại phải thanh toán, cùng danh sách các tài sản đảm bảo. Nếu phát hiện hỏng hóc, nhân viên giao dịch có thể tìm kiếm hoặc nhập mới các hỏng hóc cần bồi thường, hệ thống tự động tính tổng số tiền khách hàng cần thanh toán. Sau khi xác nhận thanh toán, nhân viên giao dịch nhận xác nhận thanh lý hợp đồng, hệ thống sẽ lưu kết quả, in hóa đơn thanh lý hợp đồng để nhân viên và khách hàng ký xác nhận. Khách hàng hoàn tất thanh toán, nhận lại tài sản đảm bảo, và nhân viên giao dịch xác nhận đã nhận lại xe, hoàn tất quy trình thanh lý hợp đồng.

Như vậy, ta có các danh từ và các phân tích như sau:

- **Hệ thống**: danh từ chung chung → loại.
- **Thông tin**: danh từ chung chung → loại.
- **Hợp đồng**: là đối tượng xử lý của hệ thống → là một lớp thực thể: Contract.
- **Khách hàng**: là đối tượng xử lý của hệ thống → là một lớp thực thể: Customer.
- **Nhân viên giao dịch**: không phải là đối tượng xử lý trực tiếp của hệ thống, nhưng bị quản lý cùng với các nhân viên khác theo kiểu người dùng trực tiếp của phần mềm → đề xuất là một lớp thực thể chung: User.
- **Xe**: là đối tượng xử lý của hệ thống → là một lớp thực thể: Car.
- **Hóa đơn**: là đối tượng xử lý của hệ thống → là một lớp thực thể: Invoice.
- **Tiền thuê xe**: trừu tượng, chung chung → loại (có thể làm thuộc tính của Invoice).
- **Tiền đặt cọc**: trừu tượng, chung chung → loại (có thể làm thuộc tính của Invoice).
- **Hỏng hóc**: là đối tượng xử lý của hệ thống → là một lớp thực thể: DamageType.
- **Phí bồi thường**: trừu tượng, chung chung → loại (có thể làm thuộc tính của DamageItem).
- **Tài sản đảm bảo**: là đối tượng xử lý của hệ thống → là một lớp thực thể: CollateralType.

Vậy chúng ta thu được các lớp thực thể ban đầu là:

Contract, Customer, User, Car, Invoice, DamageType, CollateralType.



Biểu đồ lớp thực thể pha phân tích

Quan hệ giữa các lớp thực thể trong hệ thống "Nhận trả xe từ khách" được xác định như sau:

- Một **Customer** có thể có nhiều **Contract**, nhưng một **Contract** chỉ thuộc về một **Customer** duy nhất. Do đó, quan hệ giữa **Customer** và **Contract** là **1-n**.
- Một **Contract** có thể bao gồm nhiều **Car**, và một **Car** có thể tham gia vào nhiều **Contract** khác nhau tại thời điểm khác nhau. Vì vậy, quan hệ giữa **Contract** và **Car** là **n-n** và cần có lớp trung gian **ContractCar** để theo dõi lịch sử thuê xe.
- Mỗi **Invoice** có thể có một hoặc nhiều xe (trường hợp khách hàng trả trước 1 vài xe hoặc toàn bộ xe, 1 xe có thể có nhiều loại hóa đơn.. Vì vậy, quan hệ giữa **ContractCar** và **Invoice** là **n-n**, cần có lớp trung gian **InvoiceDetail** để theo dõi.
- Một **Car** có thể có nhiều hỏng hóc, một loại hỏng hóc có thể gặp ở nhiều xe. Vì vậy, quan hệ giữa **ContractCar** và **DamageType** là **n-n**. cần có lớp trung gian **DamageRecord** để theo dõi.
- Một **Contract** có thể có nhiều **Collateral** (tài sản đảm bảo), 1 loại **Collateral** có thể gặp ở nhiều **Contract**. Quan hệ giữa **Contract** và **Collateral** là **n-n**. cần có lớp trung gian **CollateralRecord** để theo dõi.
- Một **User** (nhân viên giao dịch) có thể xử lý nhiều **Invoice**, nhưng một **Invoice** chỉ do một **User** xử lý. Do đó, quan hệ giữa **User** và **Invoice** là **1-n**.

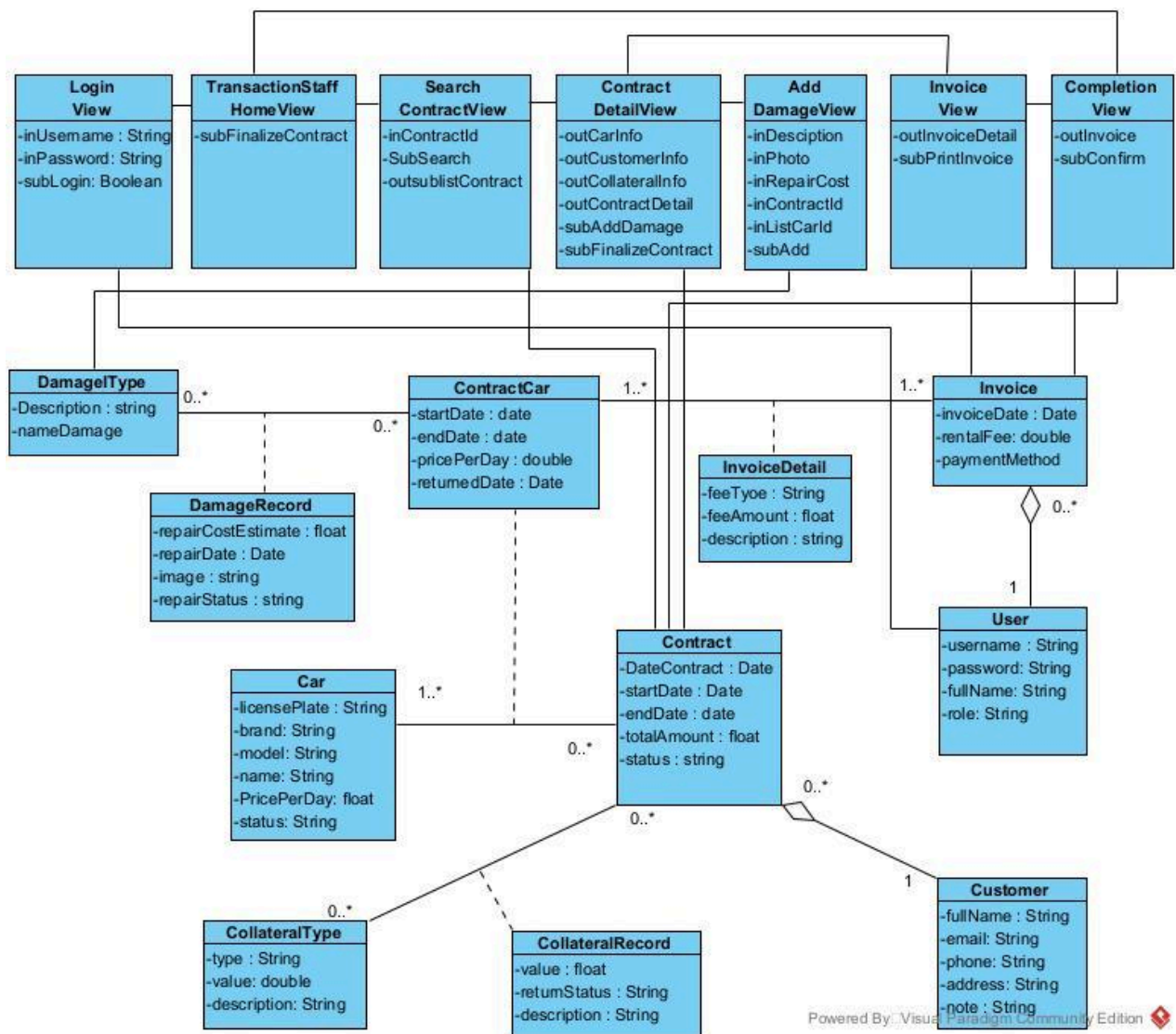


#### **4. Biểu đồ lớp đầy đủ pha phân tích của modul**

Lặp lại các bước sau cho đến khi hết GD chính

B1: Mỗi GD chính  $\rightarrow$  1 lớp biên

- Nhập dữ liệu  $\rightarrow$  in\_ <Tên>
- Hiện dữ liệu  $\rightarrow$  out\_ <Tên>
- Điều khiển/submit  $\rightarrow$  sub\_ <Tên>
- Kết hợp 3 loại trên



Biểu đồ lớp đầy đủ pha phân tích

B2: Mỗi hoạt động vào/ra dữ liệu với hệ thống → Đề xuất hàm

1. Hàm kiểm tra đăng nhập
  - Tên: checkLogin()
  - Input: username, password (thuộc lớp User)
  - Output: boolean
  - Lớp chủ thể: User
2. Hàm tìm hợp đồng theo ID
  - Tên: searchByContractId()
  - Input: contractId (thuộc lớp Contract)
  - Output: Contract
  - Lớp chủ thể: Contract
3. Hàm lấy chi tiết hợp đồng
  - Tên: getContractById()
  - Input: id (thuộc lớp Contract)

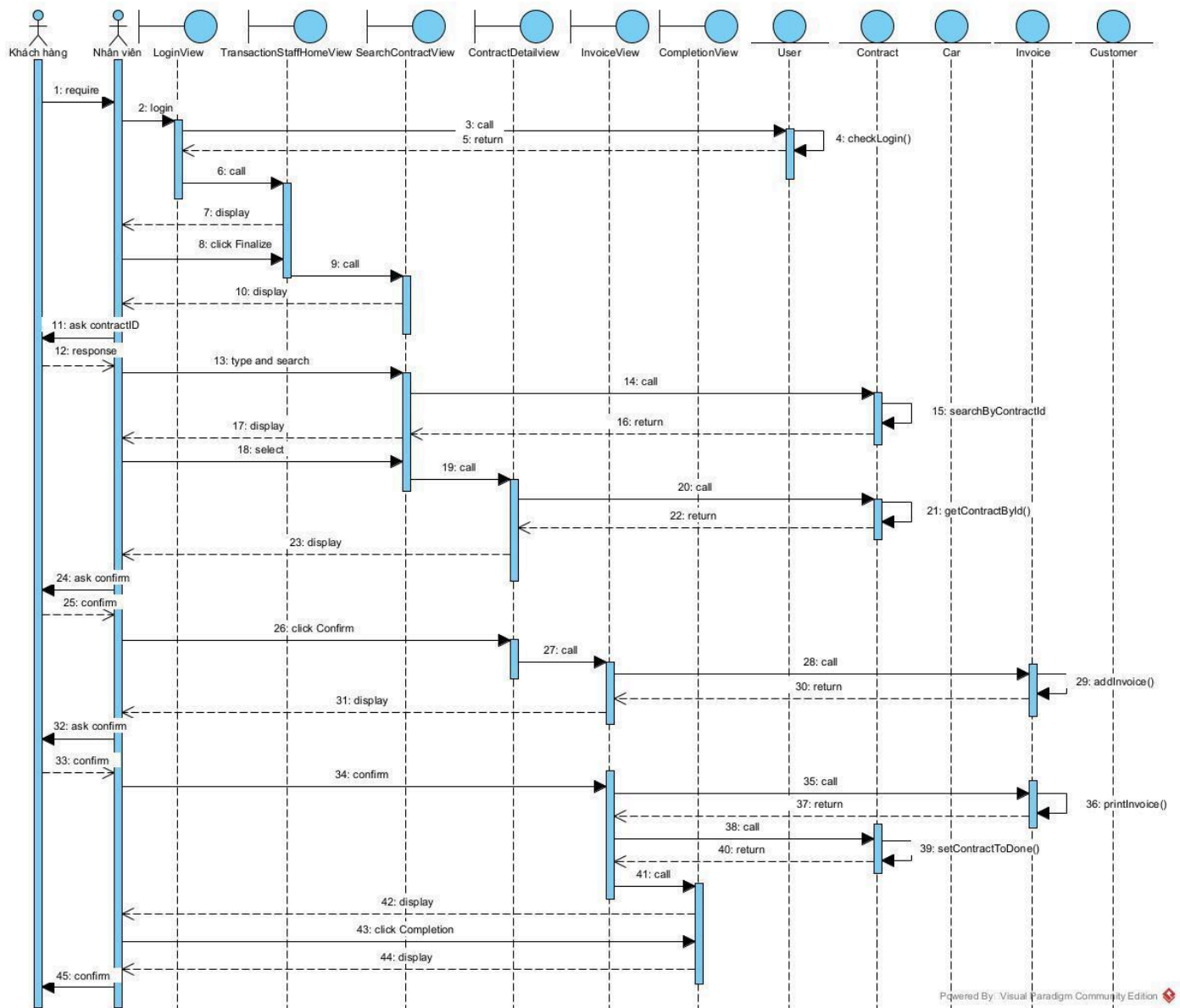
- Output: Contract
- Lớp chủ thể: Contract
- 4. Hàm in hóa đơn
  - Tên: printInvoice()
  - Input: không có đầu vào rõ ràng trong sơ đồ
  - Output: Không xác định rõ, giả định là boolean hoặc in ra trực tiếp
  - Lớp chủ thể: Invoice
- 5. Hàm thêm báo cáo hư hỏng
  - Tên: addNew()
  - Input: reportDate, description, repairCost, contractId (thuộc lớp DamageReport)
  - Output: Không rõ ràng, giả định là boolean hoặc void
  - Lớp chủ thể: DamageReport
- 6. Hàm tìm báo cáo hư hỏng theo hợp đồng
  - Tên: searchByContractCarId()
  - Input: contractCarId
  - Output: Danh sách DamageReport
  - Lớp chủ thể: DamageReport
- 7. Hàm cập nhật trạng thái hợp đồng hoàn thành
  - Tên: setContractToDone()
  - Input: Không rõ ràng trong sơ đồ, có thể là contractId hoặc không cần nếu gọi trên đối tượng
  - Output: Không xác định rõ, có thể là boolean hoặc void
  - Lớp chủ thể: Contract
- 8. Hàm thêm hóa đơn
  - Tên: addInvoice()
  - Input: Không rõ ràng trong sơ đồ, nhưng thuộc về lớp Invoice
  - Output: Không xác định rõ, giả định là boolean hoặc void
  - Lớp chủ thể: Invoice

## **5. Biểu đồ tuần tự pha phân tích của modul**

1. Khách hàng đến cửa hàng yêu cầu trả xe thanh lý hợp đồng
2. Nhân viên giao dịch đăng nhập trên LoginView
3. Lớp LoginView gọi sang lớp User
4. Lớp User thực hiện hàm checkLogin()
5. Lớp User trả kết quả về lớp LoginView

6. Lớp LoginView gọi lớp TransactionStaffHomeView
7. Lớp TransactionStaffHomeView hiển thị cho nhân viên
8. Nhân viên click chức năng "Thanh lý hợp đồng"
9. Lớp TransactionStaffHomeView gọi lớp SearchContractView
10. Lớp SearchContractView hiển thị cho nhân viên
11. Nhân viên hỏi khách hàng về mã hợp đồng
12. Khách hàng trả lời cho nhân viên
13. Nhân viên nhập mã hợp đồng và tìm trên giao diện SearchContractView
14. Lớp SearchContractView gọi lớp Contract
15. Lớp Contract thực hiện hàm searchByContractId()
16. Lớp Contract trả kết quả về cho lớp SearchContractView
17. Lớp SearchContractView hiển kết quả cho nhân viên
18. Nhân viên chọn hợp đồng cần thanh lý tìm được
19. Lớp SearchContractView gọi lớp ContractDetailView
20. Lớp ContractDetailView gọi lớp Contract
21. Lớp Contract thực hiện hàm getContractById()
22. Lớp Contract trả kết quả về lớp ContractDetailView
23. Lớp ContractDetailView hiển thị thông tin chi tiết hợp đồng
24. Nhân viên kiểm tra thông tin và xác nhận lại với khách hàng
25. Khách hàng xác nhận đúng hợp đồng của mình
26. Nhân viên xác nhận hoàn tất kiểm tra xe và chọn "xác nhận thanh lý"
27. Lớp ContractDetailView gọi lớp InvoiceView
28. Lớp InvoiceView gọi lớp Invoice
29. Lớp Invoice thực hiện hàm addInvoice()

30. Lớp Invoice trả kết quả việc tạo hóa đơn về cho lớp InvoiceView
31. Lớp InvoiceView hiển thị thông tin hóa đơn cho nhân viên
32. Nhân viên đọc lại thông tin hóa đơn và yêu cầu khách hàng xác nhận
33. Khách hàng đồng ý với thông tin thanh toán
34. Nhân viên xác nhận in hóa đơn, thanh lý hợp đồng
35. Lớp InvoiceView gọi lớp Invoice
36. Lớp Invoice thực hiện hàm printInvoice() để in hóa đơn
37. Lớp Invoice trả kết quả thành công về lớp InvoiceView
38. Lớp InvoiceView gọi lớp Contract
39. Lớp Contract thực hiện hàm setContractToDone().
40. Lớp Contract trả kết quả về lớp InvoiceView
41. Lớp InvoiceView gọi lớp CompletionView
42. Lớp CompletionView hiển thị cho nhân viên
43. Nhân viên click xác nhận hoàn tất trên CompletionView
44. Lớp CompletionView hiển thị thông báo hoàn tất cho nhân viên
45. Nhân viên báo khách hàng hoàn tất thanh lý



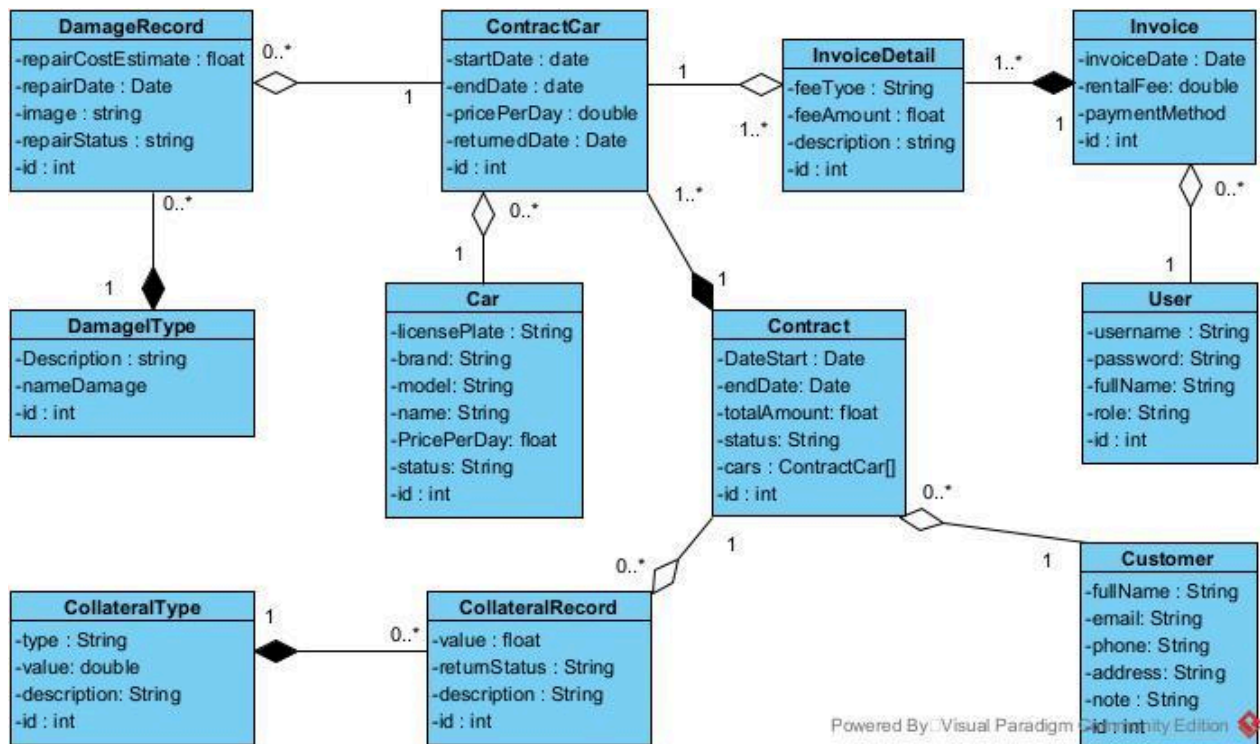
Powered By: Visual Paradigm Community Edition

Biểu đồ tuần tự pha phân tích của modul

## 6. Biểu đồ thiết kế lớp thực thể của modul

Trong phần ta sử dụng biểu đồ lớp thực thể từ pha phân tích, các bước tiến hành như sau:

- Bổ sung kiểu dữ liệu cho các thuộc tính đã có
- Bổ sung thuộc tính id cho các lớp không kế thừa từ lớp khác.
- Chuyển đổi các quan hệ association thành nhiều quan hệ thành phần. Quan hệ ContractCar + DamageType -> DamageRecord chuyển thành quan hệ: ContractCar là thành phần của DamageRecord, DamageRecord là thành phần của DamageType. Quan hệ Contract + Car -> ContractCar chuyển thành quan hệ: Car là thành phần của ContractCar, ContractCar là thành phần của Contract. Contract + CollateralType -> CollateralRecord chuyển thành quan hệ: Contract là thành phần của CollateralRecord, CollateralRecord là thành phần của CollateralType.
- Bổ sung các thuộc tính tương ứng với quan hệ thành phần.



Biểu đồ thiết kế lớp thực thể cho module Nhận trả xe

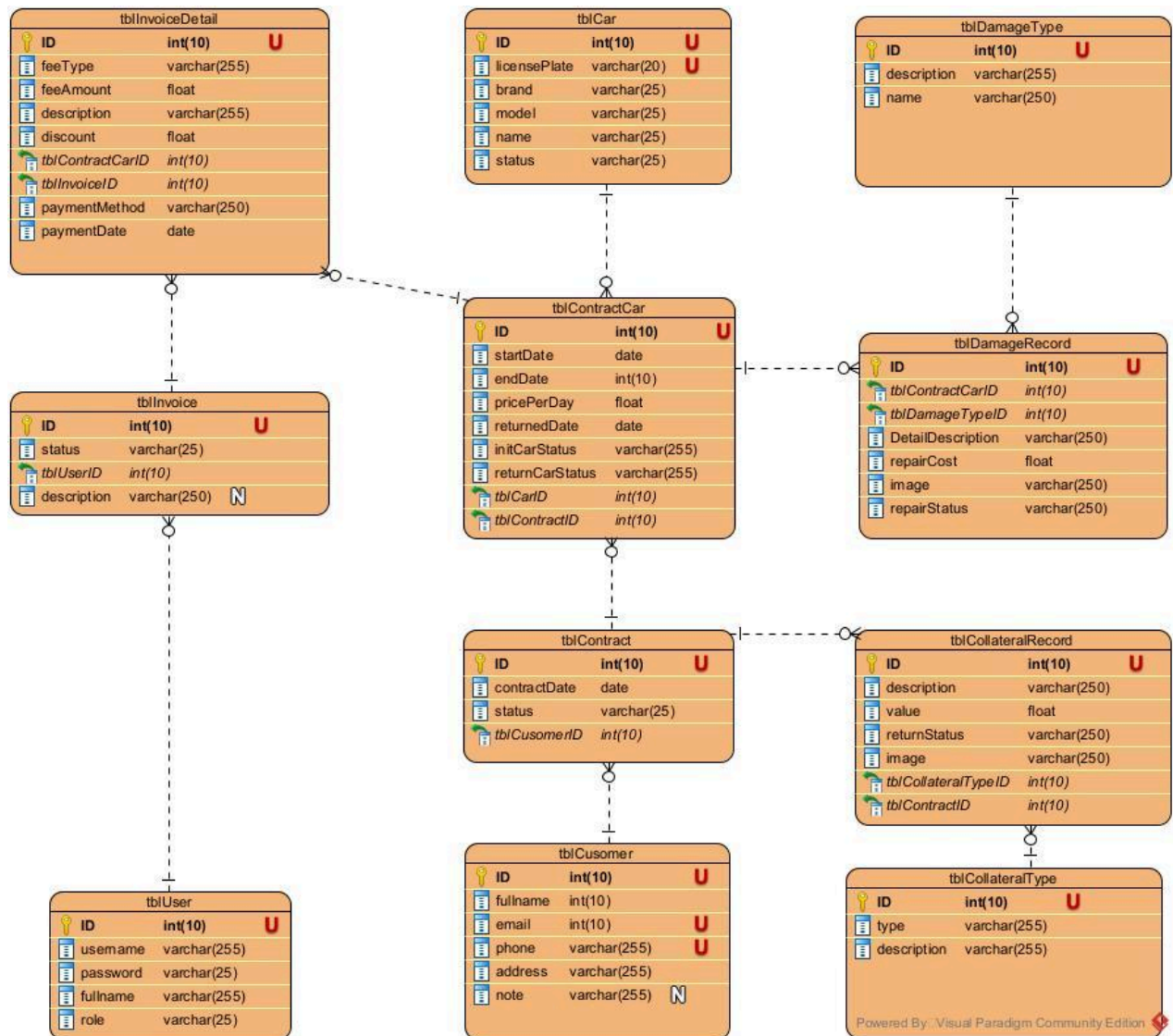
## 7. Biểu đồ thiết kế CSDL của modul

Trong phần này ta sử dụng biểu đồ thiết kế lớp thực thể ở phần trên để thực hiện thiết kế cơ sở dữ liệu, mỗi lớp thực thể sẽ tương đương với một bảng trong cơ sở dữ liệu. Từ đó chúng ta có thể đề xuất các bảng dữ liệu như sau:

- tblCar: lưu các thông tin về ô tô, bao gồm: id, biển số, thương hiệu, tên, thể hệ xe, và trạng thái xe.



- tblContract: lưu các thông tin về hợp đồng, bao gồm: id, Ngày tạo hợp đồng, ngày bắt đầu thuê, ngày trả xe, ngày trả xe thực tế ID khách hàng, trạng thái hợp đồng .



- tblContractCar: lưu thông tin liên kết giữa hợp đồng và ô tô, bao gồm: id, ID ô tô, và ID hợp đồng.
- tblUser: lưu các thông tin về người dùng hệ thống, bao gồm: id, tên đăng nhập, mật khẩu, họ tên đầy đủ, và vai trò.
- tblCustomer: lưu các thông tin về khách hàng, bao gồm: id, họ tên đầy đủ, email, số điện thoại, địa chỉ, và ghi chú.
- tblDamageRecord: lưu các thông tin chi tiết về hư hỏng của mỗi xe, bao gồm: id, mô tả hư hỏng, chi phí sửa chữa ước tính, thông tin ô tô trong hợp đồng, và ID báo cáo hư hỏng.
- tblDamageType: lưu các loại hư hỏng phổ biến hay gặp bao gồm các thông tin: tên hỏng hóc, mô tả chung về hỏng hóc, id của loại hỏng hóc đó



- tblCollateralRecord: lưu các thông tin chi tiết về tài sản thế chấp của mỗi hợp đồng, bao gồm: id, mô tả tài sản, giá trị của tài sản, thông tin hợp đồng liên quan, và ID.
- tblCollateralType: lưu các loại tài sản thế chấp phổ biến hay gặp bao gồm các thông tin: tên loại tài sản, mô tả chung về tài sản, id.
- tblInvoice: lưu thông tin chung về hóa đơn, bao gồm: id, trạng thái hóa đơn chung trong hợp đồng (đã thanh lý hoàn toàn hoặc còn đang xử lý), mô tả thêm.
- tblInvoiceDetail: lưu các thông tin chi tiết về các loại hóa đơn trong 1 hợp đồng (hóa đơn hư hỏng, hóa đơn trả trước 1 phần xe, hóa đơn thanh lý toàn bộ hợp đồng....), bao gồm: id, ngày lập hóa đơn, loại phí tương ứng với loại hóa đơn, phương thức thanh toán, chi phí khác, ID hợp đồng xe liên quan, ngày thanh toán, số tiền giảm giá.

Quan hệ giữa các bảng được mô tả như sơ đồ trên:

- Bảng tblCar quan hệ 1-n với bảng tblContractCar.
- Bảng tblContractCar quan hệ 1-n với bảng tblDamageRecord.
- Bảng tblContractCar quan hệ 1-n với bảng tblInvoiceDetail.
- Bảng tblContract quan hệ 1-n với bảng tblContractCar.
- Bảng tblContract quan hệ 1-n với bảng tblCollateralRecord.
- Bảng tblCustomer quan hệ 1-n với bảng tblContract.
- Bảng tblDamageType quan hệ 1-n với bảng tblDamageRecord .
- Bảng tblUser quan hệ 1-n với bảng tblInvoice.
- Bảng tblInvoice quan hệ 1-n với bảng tblInvoiceDetail.
- Bảng tblCollateralType quan hệ 1-n với bảng tblCollateralRecord.

## **8. Thiết kế giao diện và biểu đồ lớp thiết kế chi tiết đầy đủ của modul**

## Đăng Nhập Hệ Thống Cho Thuê Xe

Tên đăng nhập:

Mật khẩu:

Đăng Nhập

*Giao diện đăng nhập*

## Trang Chủ Nhân Viên

Thanh lý Hợp Đồng

Hủy

### Thông Tin Tài Khoản

**Nhân viên:** Nguyễn Văn A

**Chức vụ:** Nhân viên tiếp nhận

*Giao diện sau khi đăng nhập cho nhân viên*

# Tìm Kiếm Hợp Đồng

Mã hợp đồng:

Nhập mã hợp đồng

Tìm Kiếm

## Kết quả tìm kiếm

Mã HĐ	Khách hàng	Xe	Ngày bắt đầu	Ngày kết thúc	Trạng thái	Thao tác
HD001	Trần Văn B	Toyota Camry - 30A-12345	15/04/2025	20/04/2025	Đang thuê	<div>Chi tiết</div>
HD002	Lê Thị C	Honda Civic - 30A-67891	10/04/2025	18/04/2025	Đã trả	<div>Chi tiết</div>

*Giao diện tìm kiếm phục vụ cho việc thanh lý hợp đồng*

## Chi Tiết Hợp Đồng

Mã hợp đồng:

HD001

Khách hàng:

Trần Văn B

Ngày bắt đầu:

15/04/2025

Ngày kết thúc:

20/05/2025

Thông tin xe:

Biển số	Hãng xe	Mẫu xe	Giá thuê/ngày	Tiền đặt cọc	Ngày trả thực tế
30A-12345	Toyota	Camry	1.500.000 VNĐ	15.000.000 VNĐ	

### Danh sách hư hỏng

Xe	Mô tả	Chi phí ước tính	Tình trạng	Hình ảnh ban đầu
Camry - 30A-12345	Xước cánh cửa bên trái	500.000 VNĐ	Chưa sửa chữa	toyota_camry.png

Thêm Hư Hỏng

Kết Thúc Hợp Đồng

Quay Lại

*Giao diện chi tiết hợp đồng*

## Thêm Hư Hỏng

---

Mã hợp đồng:

HD001

Xe:

Toyota Camry - 30A-12345

Trạng thái sửa chữa:

Chưa sửa chữa

Mô tả hư hỏng:

Nhập mô tả chi tiết về hư hỏng

Chi phí ước tính:

Nhập chi phí ước tính

Hình ảnh hư hỏng:

Chọn tệp

Không có tệp nào được chọn Có thể chọn nhiều hình ảnh (tối đa 5MB/ảnh)

Lưu Hư Hỏng

Hủy

*Giao diện thêm hỏng hóc*

## Hóa Đơn Thanh Toán

Mã hợp đồng:

HD001

Ngày lập hóa đơn:

20/04/2025

Khách hàng:

Trần Văn B

Xe:

Toyota Camry - 30A-12345

Thời gian thuê:

36 ngày

Người tạo hóa đơn:

Nguyễn Văn A

Loại hóa đơn:

Thanh toán kết thúc hợp đồng

### Chi tiết thanh toán

Tiền thuê xe:	18.000.000 VNĐ
Chi phí hư hỏng:	500.000 VNĐ
Tiền đặt cọc:	-15.000.000 VNĐ
Phương thức thanh toán:	Tiền mặt
<b>Tổng cộng:</b>	<b>3.000.000 VNĐ</b>

Xác Nhận Thanh Toán

Quay Lại

*Giao diện hóa đơn*

## Hoàn Tất Hợp Đồng

---

Mã hợp đồng:

HD001

Khách hàng:

Trần Văn B

### Thông tin kết thúc

Hợp đồng đã được hoàn tất thành công.

Thời gian hoàn tất: 20/04/2025 15:30

Trạng thái: Đã thanh toán và trả xe

Về Trang Chủ

*Giao diện xác nhận thành công*

### Biểu đồ lớp thiết kế chi tiết đầy đủ của modul

*Tầng giao diện:* LoginFrm là giao diện đăng nhập. StaffHomeFrm là giao diện chính của nhân viên. SearchContractFrm là giao diện tìm kiếm hợp đồng. ContractDetailFrm là giao diện hiển thị chi tiết hợp đồng. InvoiceFrm là giao diện tạo và xác nhận hóa đơn. AddDamageFrm là giao diện ghi nhận hư hỏng xe (nếu có). CompleteFrm là giao diện xác nhận hoàn tất quá trình trả xe.

*Tầng điều khiển:* ContractDAO cần để thực hiện tìm kiếm hợp đồng và cập nhật trạng thái hợp đồng. DamageDAO cần để ghi nhận thông tin hư hỏng xe (nếu có). InvoiceDAO cần để tạo và lưu hóa đơn. UserDAO cần để kiểm tra đăng nhập của nhân viên. Các lớp này đều kế thừa từ lớp DAO để điều khiển truy nhập chung vào cơ sở dữ liệu.

*Tầng thực thể:* Cần các lớp Car, Contract, DamageReport, DamageItem, Invoice, Customer, Collateral, và User. Lớp Contract cần có các lớp thành phần ContractCar, DamageReport, Collateral, và Customer. Lớp DamageReport cần có lớp thành phần DamageItem. Lớp Invoice cần có lớp thành phần User, Lớp ContractCar cần có lớp thành phần Car, Lớp DamageItem cần có lớp thành phần Car.

#### a) Kiểm tra đăng nhập nhân viên:

**Chức năng:** checkLogin()

**Mô tả:** Xác thực thông tin đăng nhập của nhân viên để truy cập hệ thống.

**Tham số vào:**

- checkLogin(User u): Đối tượng User chứa thông tin đăng nhập.

**Tham số ra:**

- checkLogin(User u): User: Trả về thông tin đầy đủ của nhân viên nếu đăng nhập thành công.

**Tối ưu hóa:** Chỉ truy cập database một lần trong UserDao để kiểm tra thông tin đăng nhập. Sau khi đăng nhập thành công, lưu thông tin User vào bộ nhớ tạm (session) để tránh truy vấn lại database khi cần thông tin như role hay fullName.

**b) Tìm kiếm hợp đồng thuê xe:**

**Chức năng:** searchContract() (liệt kê trong ContractDAO)

**Mô tả:** Tìm kiếm hợp đồng thuê xe dựa trên mã hợp đồng (contractID).

**Tham số vào:**

- searchContract(int contractID): Mã hợp đồng

**Tham số ra:**

- searchContract(): Contract: Trả về duy nhất một hợp đồng nếu có, bao gồm thông tin liên quan như startDate, endDate, totalAmount, status, cars (danh sách xe), customer, và collateral.

**Tối ưu hóa:**

- Truy vấn database một lần duy nhất trong ContractDAO để lấy danh sách hợp đồng và các thông tin liên quan (**JOIN** với Car, Customer, Collateral).
- Lưu danh sách hợp đồng vào bộ nhớ tạm để sử dụng cho các bước tiếp theo

**c) Thêm báo cáo hư hỏng (khi nhận hoặc trả xe):**

**Chức năng:** addDamage() (liệt kê trong DamageDAO)

**Mô tả:** Ghi nhận thông tin hư hỏng của xe (nếu có) khi khách hàng nhận hoặc trả xe, tạo một báo cáo hư hỏng (DamageReport).



**Tham số vào:**

- addDamage(int: ContractID): ID của đối tượng Contract (đã lấy từ bước searchContract())

**Tham số ra:**

- addDamage(): DamageReport[]: Trả về danh sách các hỏng hóc mới được thêm.

**Tối ưu hóa:**

- Sử dụng thông tin xe từ đối tượng Contract đã có, không cần truy vấn lại cơ sở dữ liệu.
- Thực hiện một truy vấn duy nhất trong DamageDAO để lưu DamageReport và các DamageItem liên quan.
- Lưu danh sách DamageReport[] vào bộ nhớ tạm để sử dụng trong bước tạo hóa đơn (invoiceConfirm()) phục vụ cho việc tính toán chi phí sửa chữa, tránh truy vấn lại.

**d) Lưu hóa đơn và cập nhật trạng thái (khi xác nhận)**

**Chức năng:** confirmInvoiceAndUpdateStatus() (có trong InvoiceDAO)

**Mô tả:** Khi nhân viên bấm "Confirm" trên InvoiceFrm, lưu hóa đơn Invoice vào database với trạng thái "Đã thanh toán", đồng thời cập nhật trạng thái của Contract (ví dụ: "Hoàn thành") và Car (ví dụ: "Available").

**Tham số vào:**

- +confirmInvoiceAndUpdateStatus(int : idContract, in : invoice) : Mã hợp đồng (idContract) và đối tượng Invoice (đã tạo từ bước trước).

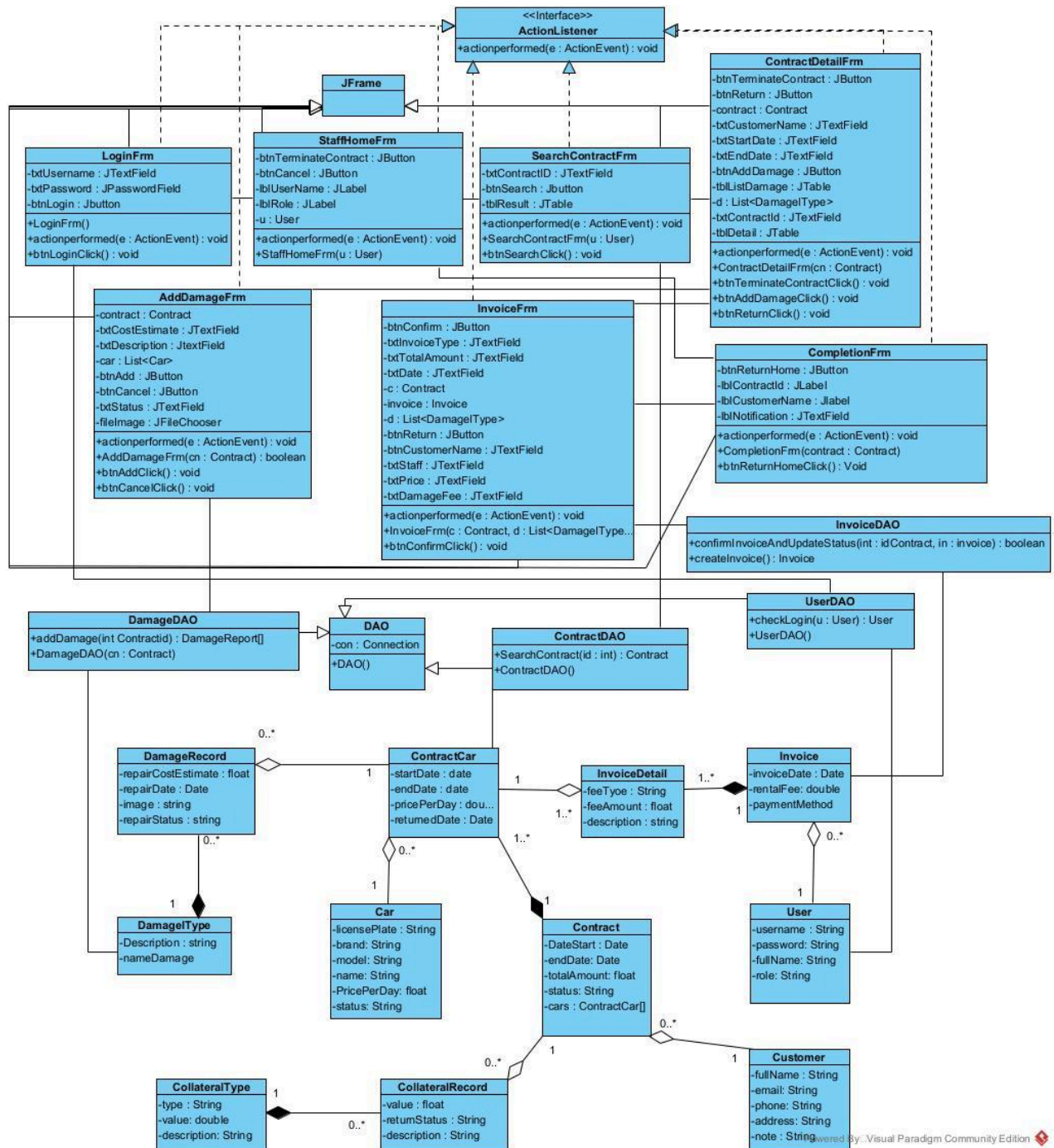
**Tham số ra:**

- confirmInvoiceAndUpdateStatus(): boolean: Trả về true nếu tất cả thao tác thành công, false nếu thất bại.

**Tối ưu hóa:**

- Sử dụng một giao dịch (transaction) duy nhất trong database:
  - Lưu hóa đơn (Invoice) với trạng thái "Đã thanh toán".
  - Cập nhật Contract.status thành "Hoàn thành".

- Cập nhật Car.status thành "Available" cho tất cả xe liên quan đến hợp đồng.



Biểu đồ lớp thiết kế chi tiết chức năng nhận trả xe từ khách hàng

## 9. Biểu đồ tuần tự pha thiết kế của modul

Chi tiết hoạt động của chức năng được mô tả như sau:

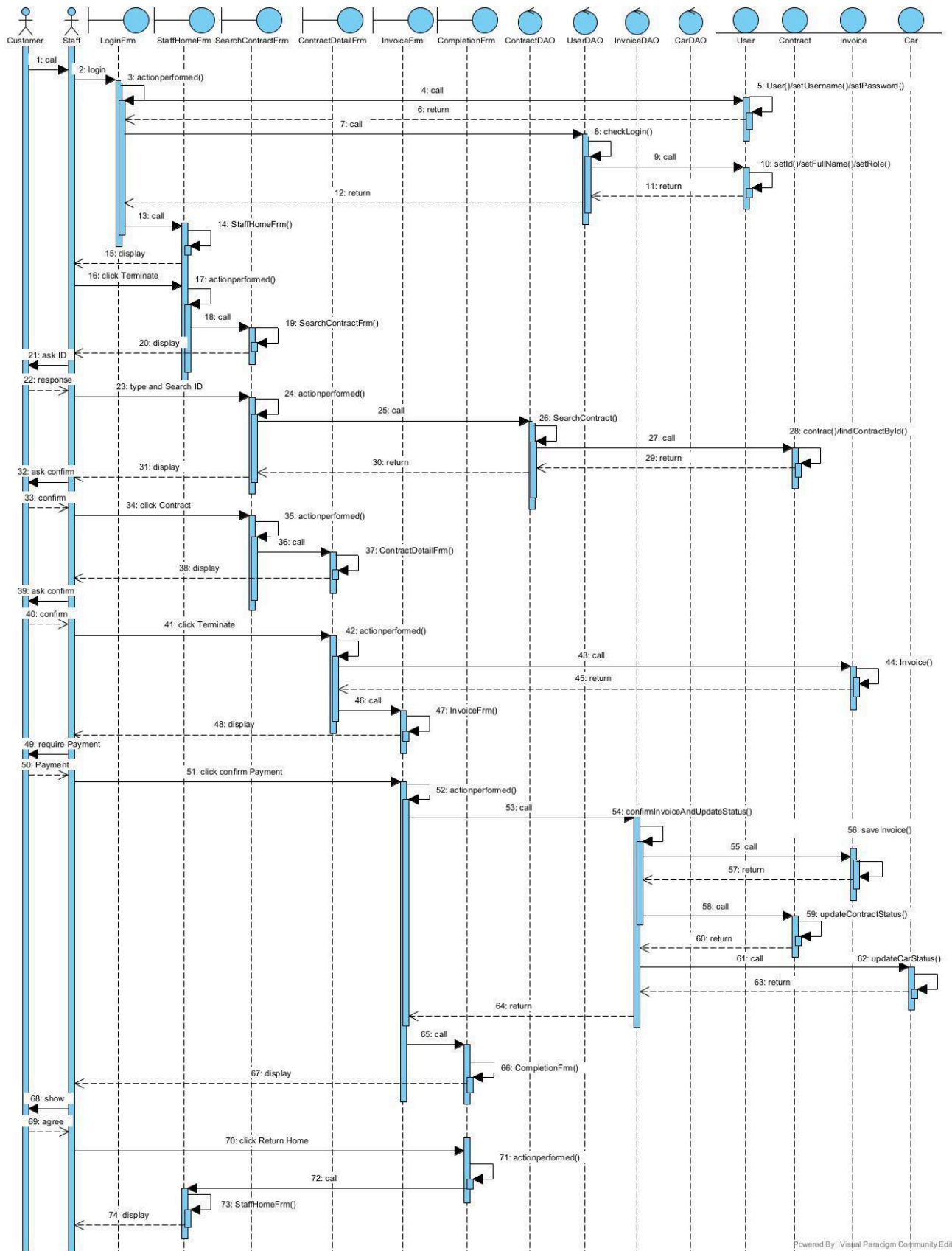
1. Khách hàng đến cửa hàng trả ô tô và yêu cầu thanh lý hợp đồng

2. Nhân viên nhập tên đăng nhập, mật khẩu và click đăng nhập trên giao diện LoginFrm.
3. Hàm actionPerformed() của lớp LoginFrm được gọi.
4. Hàm actionPerformed() gọi lớp User để đóng gói thông tin đăng nhập.
5. Lớp User đóng gói thông tin vào thực thể User (username, password).
6. Lớp User trả đối tượng User về cho phương thức actionPerformed().
7. Phương thức actionPerformed() gọi phương thức checkLogin() của lớp UserDAO.
8. Phương thức checkLogin() kiểm tra thông tin đăng nhập trong cơ sở dữ liệu.
9. Phương thức checkLogin() gọi lớp User để đóng gói bổ sung thuộc tính fullName, role.
10. Lớp User gọi các hàm setFullName(), setRole() để đóng gói bổ sung.
11. Lớp User trả kết quả về cho phương thức checkLogin().
12. Phương thức checkLogin() trả kết quả về cho phương thức actionPerformed().
13. Phương thức actionPerformed() gọi lớp StaffHomeFrm.
14. Phương thức StaffHomeFrm() được gọi.
15. Giao diện nhân viên hiển thị.
16. Nhân viên click chọn chức năng thanh lý hợp đồng.
17. Phương thức actionPerformed() được kích hoạt.
18. Phương thức actionPerformed() gọi lớp SearchContractFrm.
19. Phương thức SearchContractFrm() được gọi.
20. Giao diện tìm kiếm hợp đồng hiển thị.
21. Nhân viên yêu cầu khách hàng cung cấp thông tin ID hợp đồng cần thanh lý
22. Khách hàng cung cấp mã hợp đồng.
23. Nhân viên nhập mã hợp đồng và click tìm kiếm.
24. Phương thức actionPerformed() được kích hoạt.
25. Phương thức actionPerformed() gọi phương thức searchContract() của lớp ContractDAO.
26. Phương thức searchContract() tìm hợp đồng theo mã hợp đồng.
27. Phương thức searchContract() gọi lớp Contract để đóng gói kết quả tìm kiếm.
28. Lớp Contract thực hiện đóng gói vào thực thể Contract, findContractById() được thực thi.
29. Lớp Contract trả đối tượng về cho phương thức searchContract().
30. Phương thức searchContract() trả kết quả về cho phương thức actionPerformed() của lớp SearchContractFrm.
31. Phương thức actionPerformed() hiển thị danh sách hợp đồng cho nhân viên.
32. Nhân viên đưa ra hợp đồng tìm được cho khách hàng xác nhận.
33. Khách hàng xác nhận đúng là hợp đồng của mình.

34. Nhân viên click vào xem chi tiết hợp đồng
35. Phương thức actionPerformed() được kích hoạt.
36. Phương thức actionPerformed() gọi lớp ContractDetailFrm.
37. Phương thức ContractDetailFrm() được gọi.
38. Giao diện chi tiết hợp đồng hiển thị cho nhân viên.
39. Nhân viên cho khách hàng xem thông tin chi tiết của hợp đồng
40. Khách hàng xem và đồng ý với thông tin
41. Nhân viên click nút thanh lý hợp đồng.
42. Phương thức actionPerformed() được kích hoạt.
43. Phương thức actionPerformed() gọi đến lớp Invoice để đóng gói 1 đối tượng Invoice
44. Lớp Invoice thực hiện đóng gói thực thể Invoice và lưu vào bộ nhớ tạm.
45. Lớp Invoice trả kết quả về cho phương thức actionPerformed().
46. Phương thức actionPerformed() gọi lớp InvoiceFrm
47. Phương thức InvoiceFrm() được gọi.
48. Giao diện hóa đơn hiển thị.
49. Nhân viên cho khách hàng xem thông tin hóa đơn và yêu cầu khách hàng thanh toán
50. Khách hàng thực hiện thanh toán.
51. Nhân viên click nút xác nhận thanh toán trên giao diện InvoiceFrm.
52. Phương thức actionPerformed() được kích hoạt.
53. Phương thức actionPerformed() gọi phương thức confirmInvoiceAndUpdateStatus() của lớp InvoiceDAO.
54. Phương thức confirmInvoiceAndUpdateStatus() thực hiện thêm thông tin hóa đơn mới và cập nhật trạng thái của Contract, Car liên quan.
55. Phương thức confirmInvoiceAndUpdateStatus() gọi lớp Invoice
56. Lớp Invoice thêm đối tượng Invoice được khởi tạo trước đó vào cơ sở dữ liệu, saveInvoice() được thực thi.
57. Lớp Invoice trả kết quả về Phương thức confirmInvoiceAndUpdateStatus()
58. Phương thức confirmInvoiceAndUpdateStatus() gọi lớp Contract.
59. Lớp Contract thực hiện cập nhật trạng thái hợp đồng hiện tại, updateContractStatus() được thực thi.
60. Lớp Contract trả kết quả về Phương thức confirmInvoiceAndUpdateStatus()
61. Phương thức confirmInvoiceAndUpdateStatus() gọi lớp Car.
62. Lớp Car thực hiện cập nhật trạng thái xe hiện tại, updateCarStatus()() được thực thi.
63. Lớp Car trả kết quả về Phương thức confirmInvoiceAndUpdateStatus()
64. Phương thức confirmInvoiceAndUpdateStatus() trả kết quả về cho phương thức actionPerformed()

- 65. Phương thức `actionPerformed()` gọi lớp `CompletionFrm`.
- 66. Phương thức `CompletionFrm()` được gọi.
- 67. Giao diện xác nhận thành công hiển thị.
- 68. Nhân viên thông báo cho khách hàng đã hoàn tất.
- 69. Khách hàng đồng ý và ra về.
- 70. Nhân viên click nút quay trở lại trang chủ.
- 71. Phương thức `actionPerformed()` được.
- 72. Phương thức `actionPerformed()` gọi lớp `StaffHomeFrm`.
- 73. Phương thức `StaffHomeFrm()` được gọi.
- 74. Giao diện nhân viên hiển thị lại.

**Biểu đồ tuần tự:**



## 10. Test plan và test case chuẩn cho test hộp đen của modul

### a. Lập kế hoạch test

Các trường hợp phải test cho modul ‘nhận trả xe từ khách’:

TT	Chức năng	Các trường hợp cần kiểm thử
1	Nhận trả xe từ khách hàng	Khách hàng thanh lý hợp đồng, trả toàn bộ xe, không có hỏng hóc, hợp đồng chưa quá hạn
2		Khách hàng thanh lý hợp đồng, trả toàn bộ xe, không có hỏng hóc, hợp đồng đã quá hạn
3		Khách hàng thanh lý hợp đồng, trả toàn bộ xe, có hỏng hóc, hợp đồng chưa quá hạn
4		Khách hàng thanh lý hợp đồng, thuê 2 xe nhưng chỉ trả 1 xe trước, không có hỏng hóc, hợp đồng chưa quá hạn
5		Khách hàng thanh lý hợp đồng, thuê 2 xe nhưng chỉ trả 1 xe trước, có hỏng hóc, hợp đồng đã quá hạn

Bước 1: Cơ sở dữ liệu trước khi test:

- tblUser:

id	name	username	password	role
1	Nguyễn Văn A	nguyenvana	123456	seller

- **tblCar:**

ID	licensePlate	brand	model
1	51H-1234	Toyota	Camry
2	32G-4322	Lamborghini	2.0

- **tblCustomer:**

ID	fullName	email	address	phone	note
1	Nguyễn Văn B	an@gmail.com	Ha Noi	123456	N

- **tblContract:**

ID	contractDate	status	tblCustomerID
1	2025-05-08	In Progress	1

- **tblContractCar:**

ID	startDate	endDate	price PerDay	return CarStatus	returned Date	tblCarID	tblContractID
1	2025-05-09	2025-05-20	500.000đ	Đã trả trước	2025-05-15	1	1
2	2025-05-09	2025-05-20	2.000.000đ	Chưa trả	null	2	1

- **tblCollateralType:**

ID	type	description
1	CCCD	Công chứng
2	Tiền mặt	Tối thiểu 1.000.000đ

- **tblCollateralRecord:**



ID	value	description	returnStatus	tblCollateralTypeID	tblContractID
1	14.000.000đ	Cash	Chưa hoàn trả	2	1

● **tblInvoice:**

ID	UserID	invoiceStatus	description
1	1	Chưa thanh lý hoàn toàn	Đã trả trước 1 xe

● **tblInvoiceDetail:**

ID	feeType	payment method	description	discount	tblContractCarID	tblInvoiceID
1	Trả một phần	Tiền mặt	Trả trước	Không	1	1

● **tblDamageType:**

ID	description	name
1	Tối thiểu 500.000đ	xước sơn
2	Tối thiểu 1.000.000đ	Hỏng lốp
3	Tối thiểu 2.000.000đ	Hỏng đèn

● **tblDamageRecord:**

ID	description	repairCost	tblDamageTypeID	tblContractCarID	image	repairStatus

Bước 2: Các bước thực hiện

Các bước thực hiện	Kết quả mong đợi																		
1. Khách hàng (B) đến cửa hàng mang xe (51H-1234) để trả và thanh lý hợp đồng với mã hợp đồng "1"	Giao diện cửa hàng nhận thông tin khách hàng, hiển thị yêu cầu thanh lý hợp đồng																		
2. Nhân viên (id=1) nhập username "nguyenvana" và password "123456", click Đăng nhập	Giao diện đăng nhập thành công: <ul style="list-style-type: none"><li>- Thanh lý hợp đồng</li><li>- Cancel</li></ul>																		
3. Nhập ID hợp đồng = 1, click tìm kiếm	Giao diện tìm kiếm hợp đồng, ID = 1 <table><tr><th>ID hợp đồng</th><th>Tên KH</th><th>Ngày thuê</th><th>Ngày trả dự kiến</th><th>Xe</th><th>Trạng thái</th></tr><tr><td>1</td><td>Nguyễn Văn B</td><td>2025-05-09</td><td>2025-05-20</td><td>Toyota , Lamborghini</td><td>Đang thuê</td></tr></table>	ID hợp đồng	Tên KH	Ngày thuê	Ngày trả dự kiến	Xe	Trạng thái	1	Nguyễn Văn B	2025-05-09	2025-05-20	Toyota , Lamborghini	Đang thuê						
ID hợp đồng	Tên KH	Ngày thuê	Ngày trả dự kiến	Xe	Trạng thái														
1	Nguyễn Văn B	2025-05-09	2025-05-20	Toyota , Lamborghini	Đang thuê														
4. Nhân viên click nút xem chi tiết thanh lý hợp đồng	Giao diện chi tiết hợp đồng: <table><tr><th>KH</th><th>Xe</th><th>Ngày thuê</th><th>Ngày trả dự kiến</th><th>Đơn giá</th><th>Thành tiền</th><th>tổng Tiền cọc</th><th>Tài sản đảm bảo</th><th>còn thiếu</th></tr><tr><td>Nguyễn Văn B</td><td>Toyota</td><td>2025-05-09</td><td>2025-05-20</td><td>500.000 đ</td><td>6.000.000 đ</td><td>14.000.000 đ</td><td>cash</td><td>0đ</td></tr></table>	KH	Xe	Ngày thuê	Ngày trả dự kiến	Đơn giá	Thành tiền	tổng Tiền cọc	Tài sản đảm bảo	còn thiếu	Nguyễn Văn B	Toyota	2025-05-09	2025-05-20	500.000 đ	6.000.000 đ	14.000.000 đ	cash	0đ
KH	Xe	Ngày thuê	Ngày trả dự kiến	Đơn giá	Thành tiền	tổng Tiền cọc	Tài sản đảm bảo	còn thiếu											
Nguyễn Văn B	Toyota	2025-05-09	2025-05-20	500.000 đ	6.000.000 đ	14.000.000 đ	cash	0đ											

	<table><tr><td>Nguyễn Văn B</td><td>Lamborghini</td><td>2025-05-09</td><td>2025-05-20</td><td>2.000.000đ</td><td>24.000.00đ</td><td>14.000.00đ</td><td>cash</td><td>12.000.00đ</td></tr></table>	Nguyễn Văn B	Lamborghini	2025-05-09	2025-05-20	2.000.000đ	24.000.00đ	14.000.00đ	cash	12.000.00đ											
Nguyễn Văn B	Lamborghini	2025-05-09	2025-05-20	2.000.000đ	24.000.00đ	14.000.00đ	cash	12.000.00đ													
5. Nhân viên kiểm tra và phát hiện hỏng hóc, click nút thêm hỏng hóc	<div>Giao diện thêm hỏng hóc</div> <table><tr><td>CarID</td><td>Mô tả</td><td>Hình ảnh</td><td>Chi phí đền bù</td><td>ContractID</td></tr><tr><td>2</td><td>Xước sơn</td><td>URL</td><td>500.000đ</td><td>1</td></tr></table>	CarID	Mô tả	Hình ảnh	Chi phí đền bù	ContractID	2	Xước sơn	URL	500.000đ	1										
CarID	Mô tả	Hình ảnh	Chi phí đền bù	ContractID																	
2	Xước sơn	URL	500.000đ	1																	
6. Nhân viên click nút xem chi tiết thanh lý hợp đồng	<div>Giao diện chi tiết hóa đơn:</div> <table><tr><td>Mã hợp đồng</td><td>Tên KH</td><td>Xe đã thuê</td><td>Ngày thuê</td><td>Ngày trả</td><td>Đơn giá / ngày</td><td>Tiền cọc</td><td>Tổng tiền</td><td>Phương thức thanh toán</td><td>Phí phát sinh hỏng hóc</td></tr><tr><td>1</td><td>Nguyễn Văn B</td><td>Lamborghini</td><td>2025-05-09</td><td>2025-05-10</td><td>2.000.000</td><td>14.000.00đ</td><td>10.500.00đ</td><td>Cash</td><td>500.000đ</td></tr></table>	Mã hợp đồng	Tên KH	Xe đã thuê	Ngày thuê	Ngày trả	Đơn giá / ngày	Tiền cọc	Tổng tiền	Phương thức thanh toán	Phí phát sinh hỏng hóc	1	Nguyễn Văn B	Lamborghini	2025-05-09	2025-05-10	2.000.000	14.000.00đ	10.500.00đ	Cash	500.000đ
Mã hợp đồng	Tên KH	Xe đã thuê	Ngày thuê	Ngày trả	Đơn giá / ngày	Tiền cọc	Tổng tiền	Phương thức thanh toán	Phí phát sinh hỏng hóc												
1	Nguyễn Văn B	Lamborghini	2025-05-09	2025-05-10	2.000.000	14.000.00đ	10.500.00đ	Cash	500.000đ												
7. Nhân viên click xác nhận thanh toán	Hiện thị thông báo thanh toán thành công																				

Bước 3: Cơ sở dữ liệu sau khi test:

● **tblContract:**

ID	contractDate	status	tblCustomerID
1	2025-05-08	Completed	1

● **tblContractCar:**

ID	startDate	endDate	price PerDay	return CarStatus	returned Date	tblCarID	tblContractID
1	2025-05-09	2025-05-20	500.000đ	Đã trả	2025-05-15	1	1
2	2025-05-09	2025-05-20	2.000.000đ	Đã trả	2025-05-20	2	1

● **tblCollateralRecord:**

ID	value	description	returnStatus	tblCollateralTypeID	tblContractID
1	14.000.000đ	Cash	Đã hoàn trả	2	1

● **tblInvoice:**

ID	UserID	invoiceStatus	description
1	1	Đã thanh lý hoàn toàn	Đã trả toàn bộ xe

● **tblInvoiceDetail:**

ID	feeType	payment method	description	discount	tblContractCarID	tblInvoiceID
1	Trả một phần	Tiền mặt	Trả trước	Không	1	1
2	Thanh lý hợp đồng	Tiền mặt	Thanh lý hợp đồng hoàn tất	Không	2	1

● **tblDamageRecord:**

ID	description	repairCost	tblDamageTypeID	tblContractCarID	image	repairStatus
1	Xước ít	500.000đ	1	2	url	chưa sửa