Đánh giá điểm thực hành:

* Chuyên cần, tham gia đầy đủ, đúng giờ, nghiêm túc hoàn thành bài theo mẫu
* Hoàn thành bài tập theo yêu cầu đúng hạn.
* Làm báo cáo đầy đủ theo mẫu, link bài tập nộp lên Git/GitHub nộp đúng hạn: trước trước 23h59, thứ 5, tuần học.

Yêu cầu:

* Báo cáo trên assigment: (30 %)  
  (Chụp ảnh kết quả các câu dán vào file theo mẫu sau:)
* Nộp code bài lên link Git/GitHub. (30%)
* Điểm trắc nghiệm nếu có bài trắc nghiệm cuối mỗi buổi (30%)
* Điểm danh (10%). – assigment.

**Ví dụ:**

**Bài thực hành 01 – Tuần 4, ngày 21/10/2021**

* Link đã nộp bài lên Git/GitHub

<https://github.com/nguyentien090200/TKXDPM.KSTN20211.20181276.NguyenQuocTien>

* Nội dung báo cáo kết quả thực hành

## 3.3. NỘI DUNG CHI TIẾT

### 3.3.1. Bắt đầu với Git/GitHub

GitHub là một trong những dịch vụ cung cấp kho lưu trữ cho phần mềm quản lý phiên bản phân tán Git (distributed version control).

Hướng dẫn sử dụng Git/GitHub:

<https://rogerdudler.github.io/git-guide/index.vi.html>

<https://o7planning.org/vi/10283/huong-dan-su-dung-github-voi-github-desktop>

<https://git-scm.com/about>

### 3.3.2. Làm quen Astah UML

* Hướng dẫn sử dụng Astah UML:

<https://astah.net/support/astah-pro/user-guide/>

* Hướng dẫn sử dụng Astah UML với biểu đồ trình tự:

<https://astah.net/support/astah-pro/user-guide/sequence-diagram/>

<https://www.youtube.com/embed/Qi2CsTY4LSk>

* Hướng dẫn sử dụng Astah UML với biểu đồ giao tiếp:

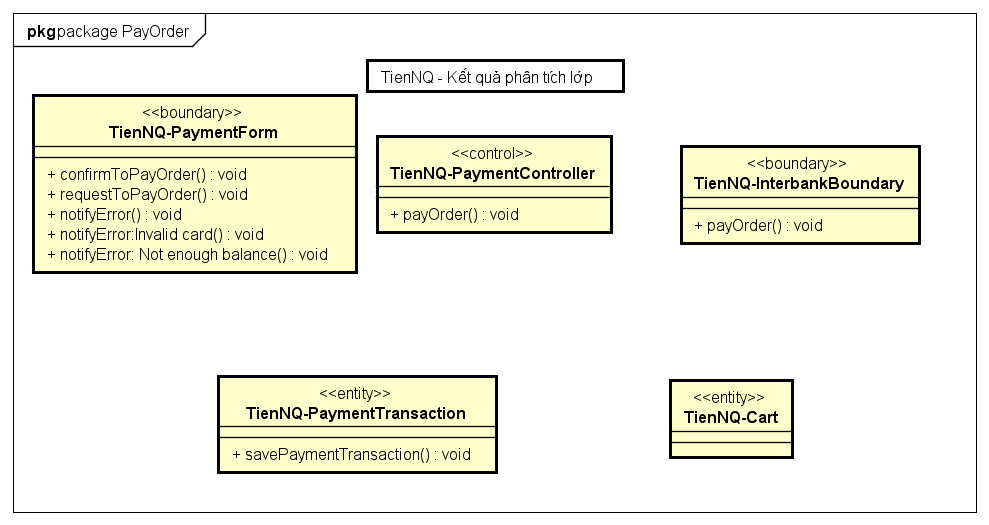
<https://astah.net/support/astah-pro/user-guide/communication-diagram/>

<https://www.uml-diagrams.org/communication-diagrams.html>

### 3.3.3. Thiết kế kiến trúc cho use case “Pay Order”

1. **Phân tích lớp**: Tìm các lớp (class) từ các hành vi trong use case.

**Kết quả là**:

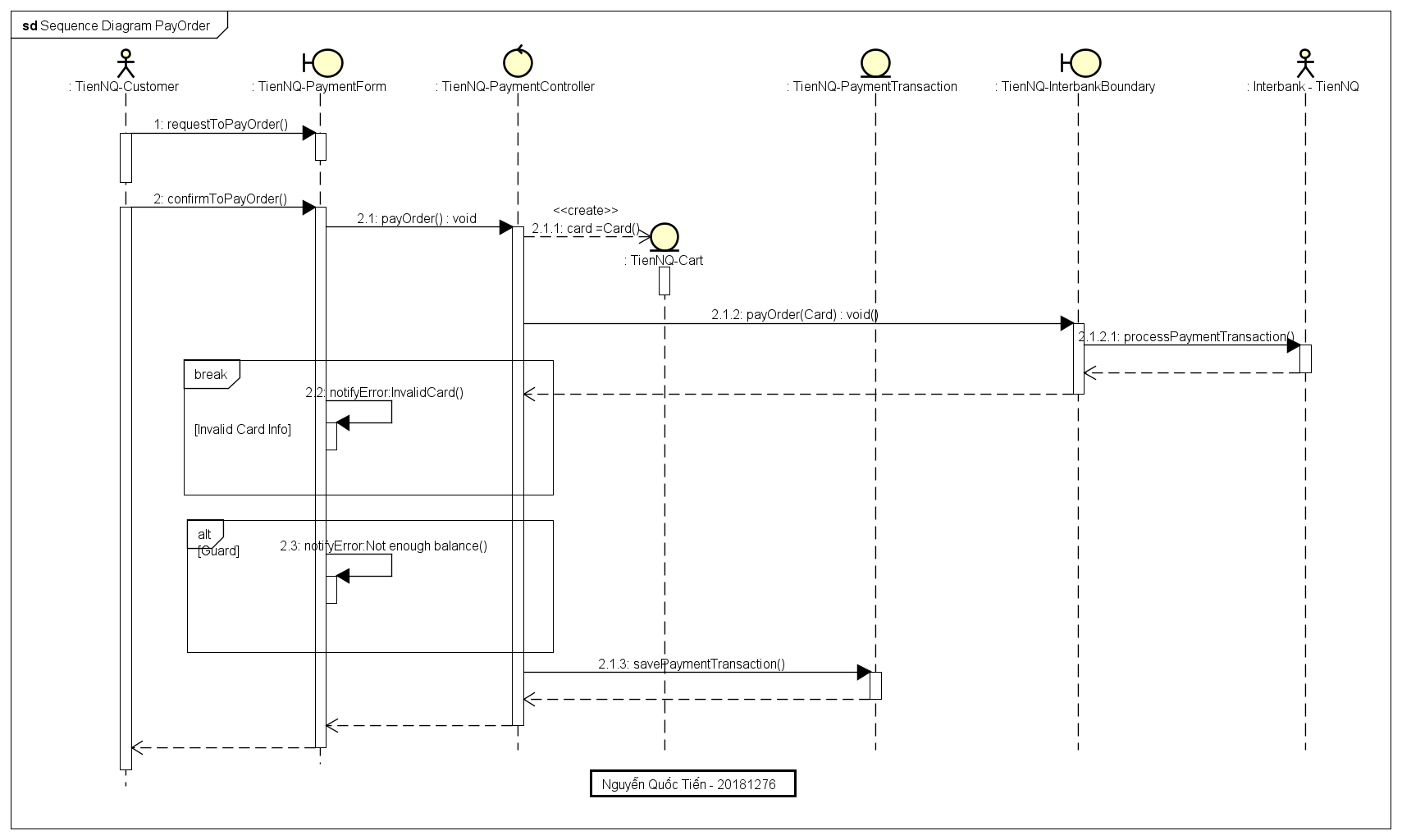


Hình 1.1 Biểu đồ lớp phân tích PayOrder

1. **Phân phối hành vi trong use case tới các lớp.** Phân bổ trách nhiệm tới các lớp và mô hình hóa mối quan hệ giữa các lớp bằng cách sử dụng biểu đồ tương tác (interaction diagram). Chúng ta có thể sử dụng biểu đồ trình tự (sequence diagram) **hoặc/và** biểu đồ giao tiếp (communication diagram).

***Biểu đồ trình tự***

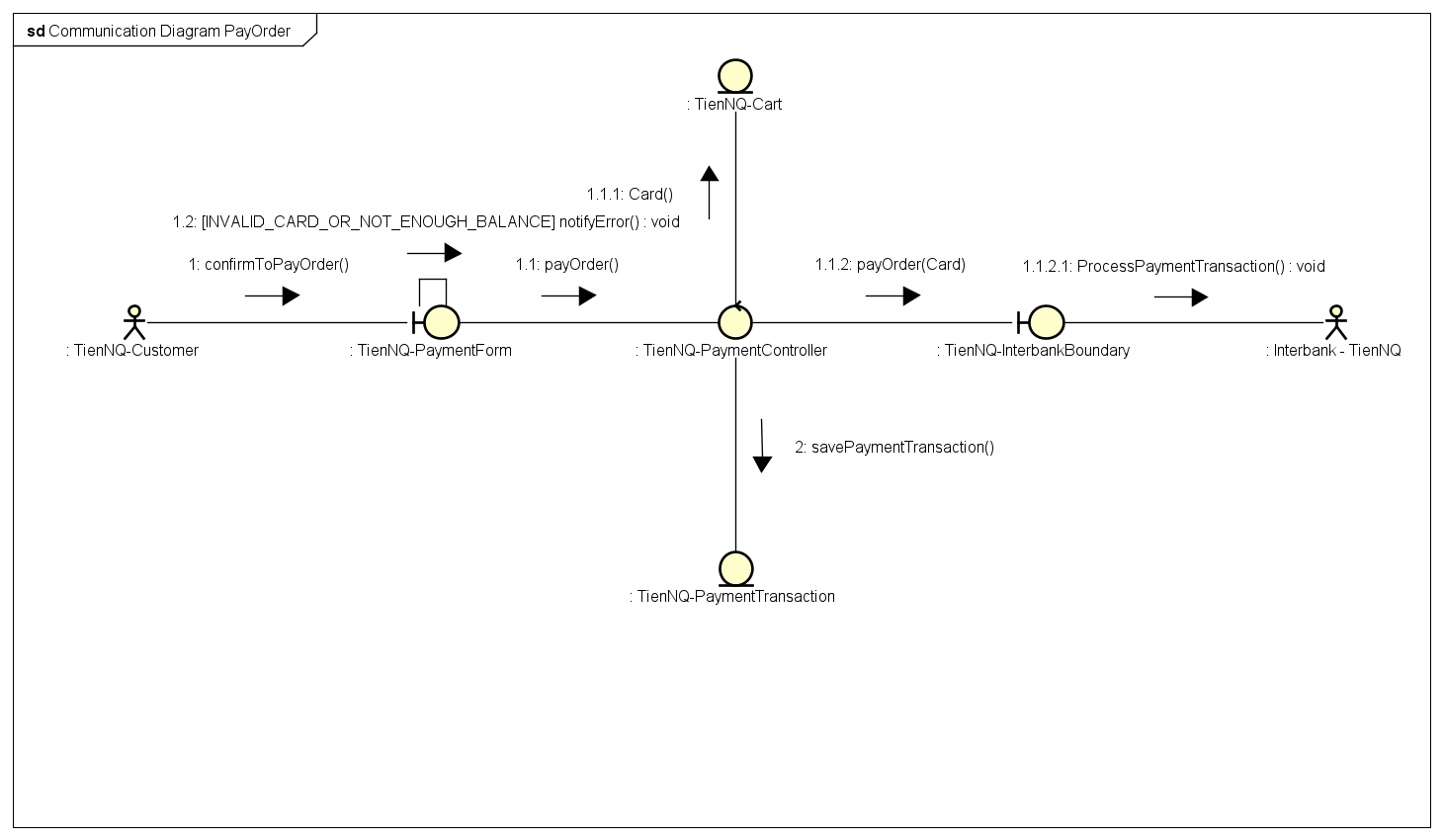
***Kết quả là:***

******

Hình 1.2 Biểu đồ trình tự PayOrder

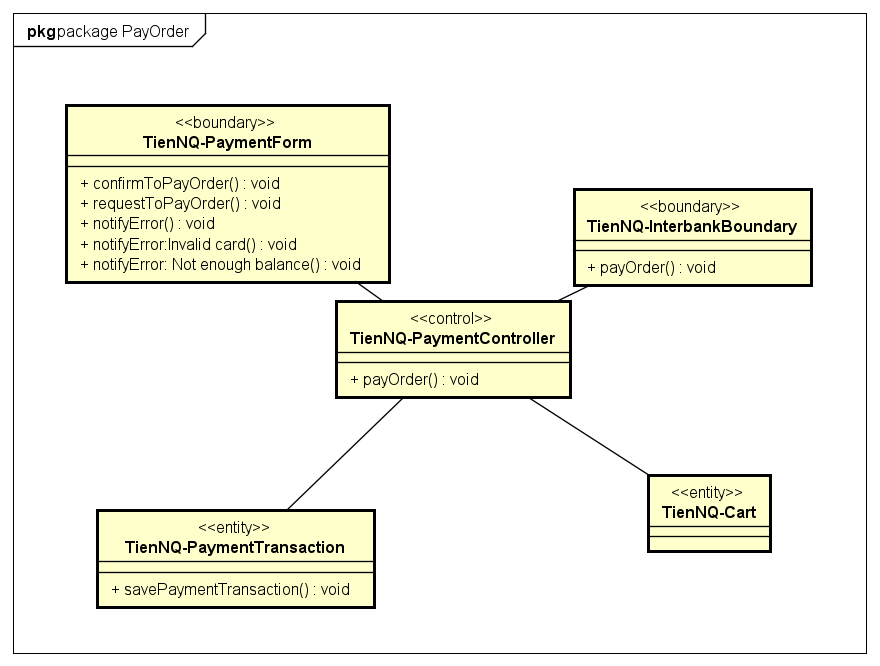
***Biểu đồ giao tiếp***

***Kết quả là:***

******

Hình 1.3 Biểu đồ giao tiếp PayOrder

**c) Biểu đồ lớp phân tích (analysis class diagram)**

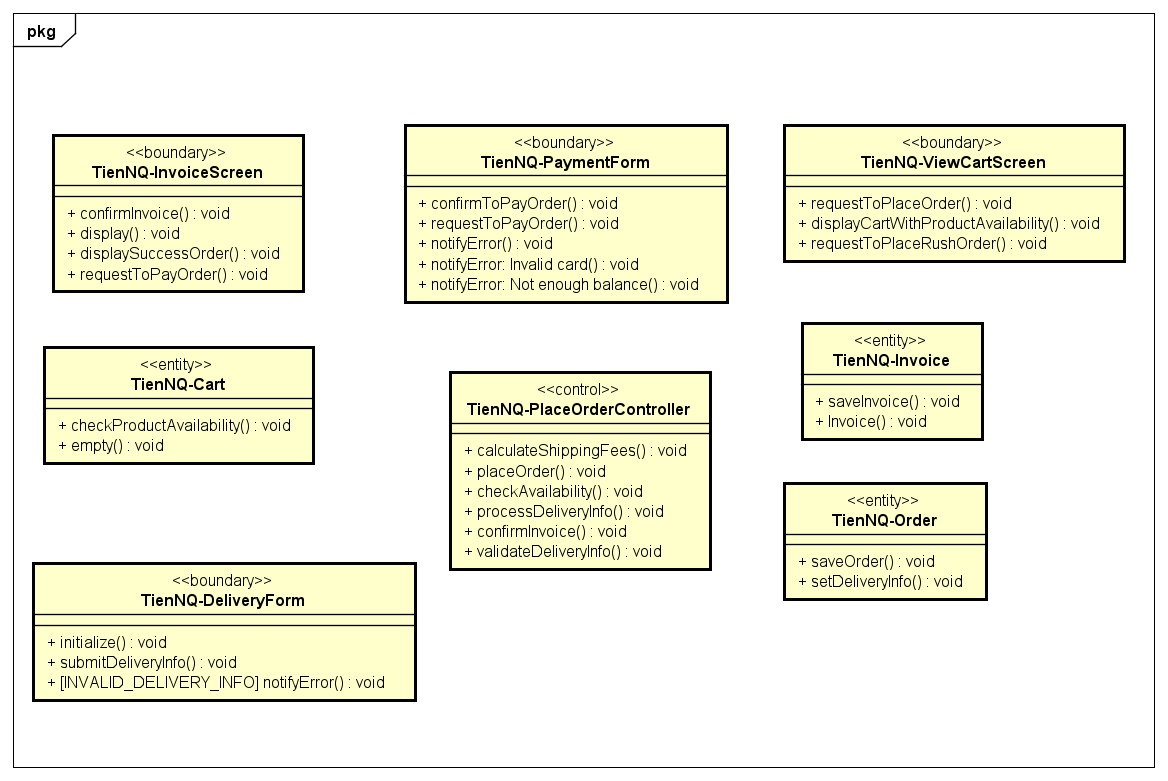
****

Hình 1.4 biểu đồ lớp phân tích PayOrder

### 3.3.4. Thiết kế kiến trúc cho use case “Place Order”

**a) Phân tích lớp**: Tìm các lớp (class) từ các hành vi trong use case.

**Kết quả là**:

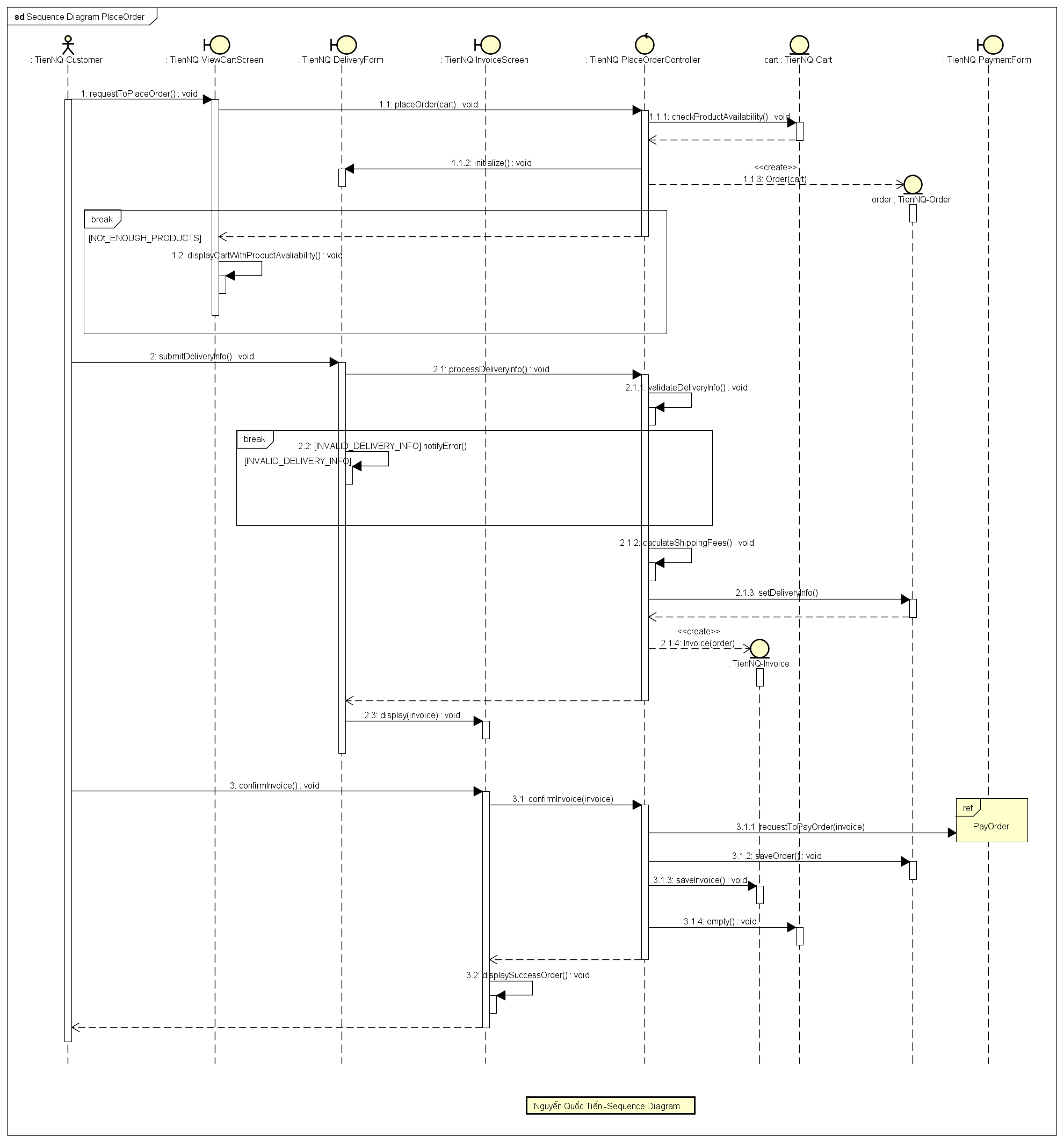


Hình 2.1 Biểu đồ lớp phân tích PlaceOrder

**b) Phân phối hành vi trong use case tới các lớp.** Phân bổ trách nhiệm tới các lớp và mô hình hóa mối quan hệ giữa các lớp bằng cách sử dụng biểu đồ tương tác (interaction diagram). Chúng ta có thể sử dụng biểu đồ trình tự (sequence diagram) **hoặc/và** biểu đồ giao tiếp (communication diagram).

***Biểu đồ trình tự***

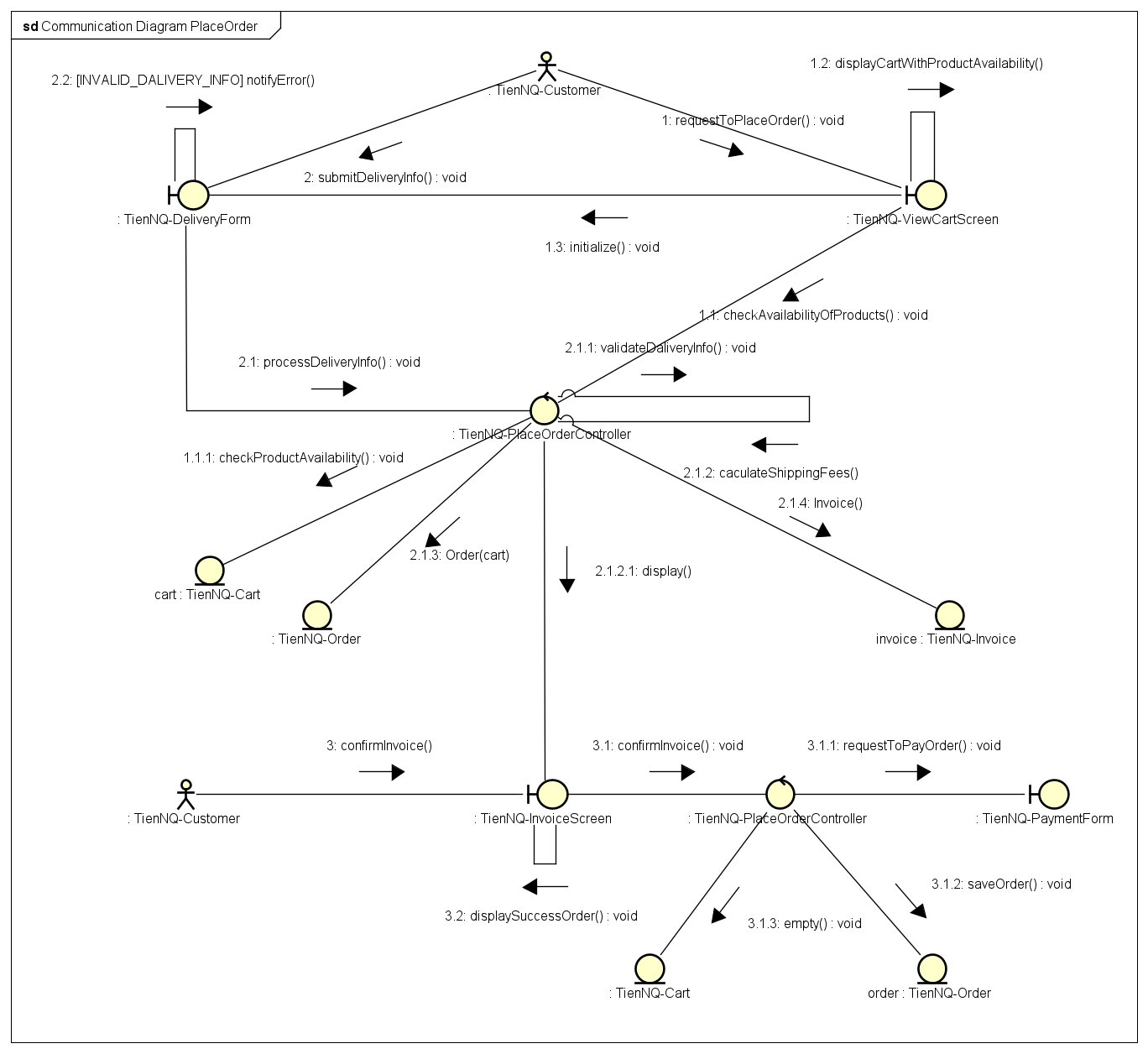
***Kết quả là:***



Hình 2.2 Biểu đồ trình tự PlaceOrder

***Biểu đồ giao tiếp***

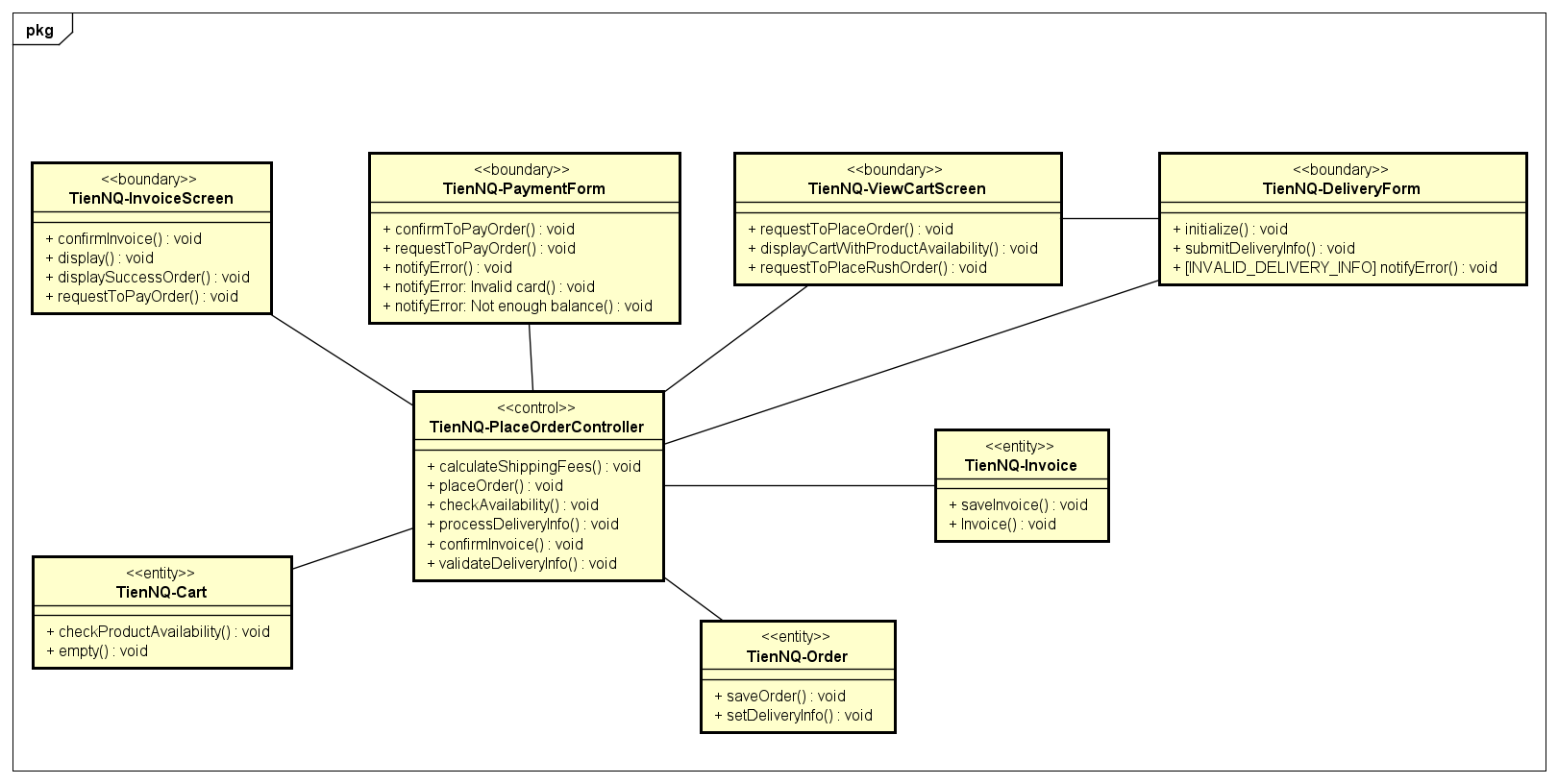
***Kết quả là:***

******

Hình 2.3 Biểu đồ giao tiếp PlaceOrder

**c) Biểu đồ lớp phân tích (analysis class diagram)**

**Kết quả là**:

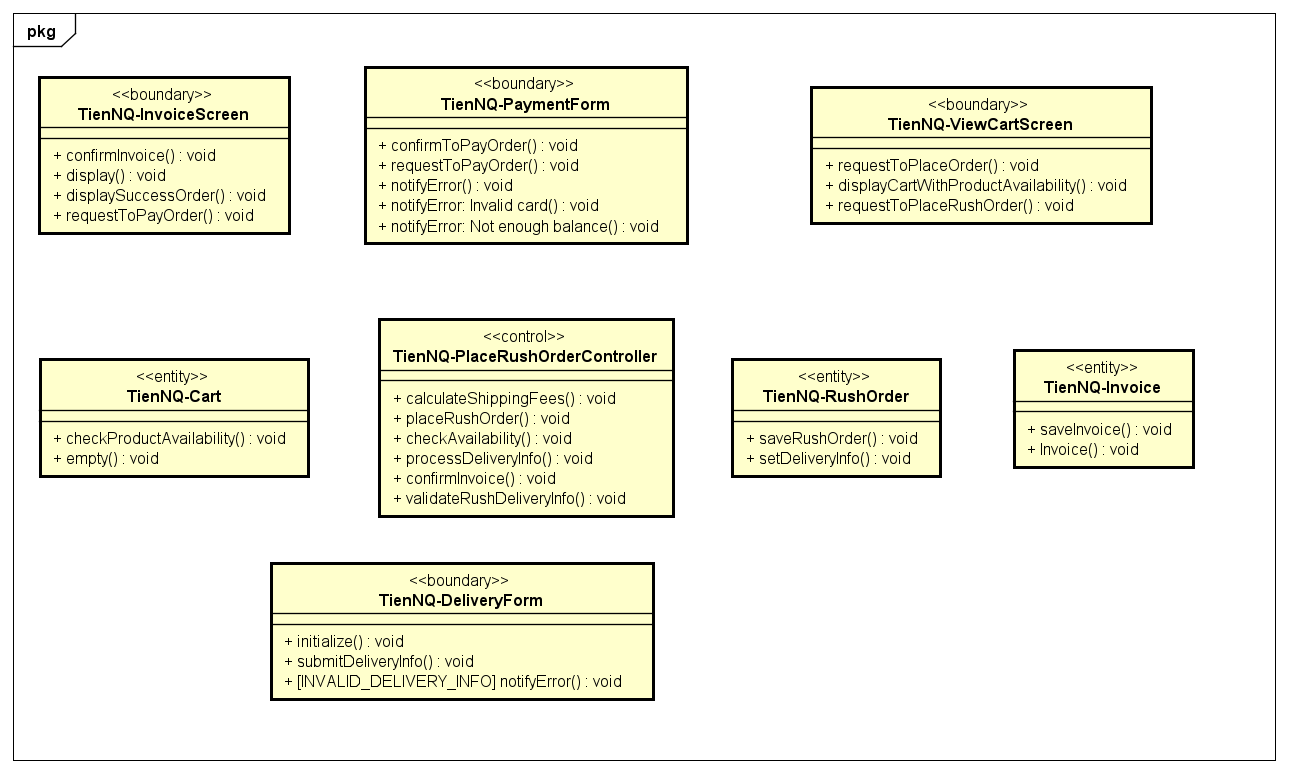
****

Hình 2.4 biểu đồ lớp phân tích PlaceOrder

### 3.3.5. Thiết kế kiến trúc cho use case “Place Rush Order”

**a) Phân tích lớp**: Tìm các lớp (class) từ các hành vi trong use case.

**Kết quả là**:

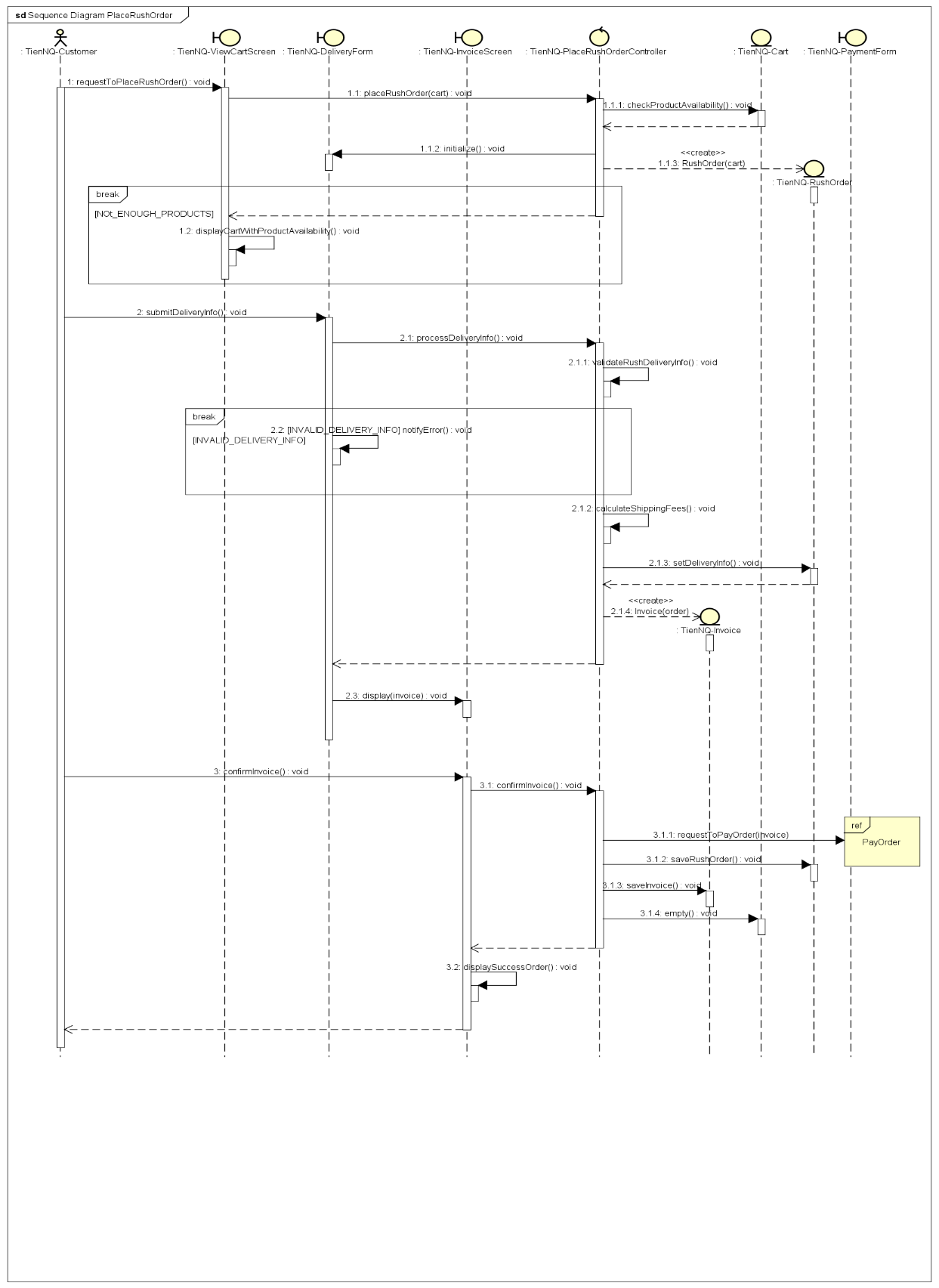


Hình 3.1 Biểu đồ lớp phân tích Place Rush Order

**b) Phân phối hành vi trong use case tới các lớp.** Phân bổ trách nhiệm tới các lớp và mô hình hóa mối quan hệ giữa các lớp bằng cách sử dụng biểu đồ tương tác (interaction diagram). Chúng ta có thể sử dụng biểu đồ trình tự (sequence diagram) **hoặc/và** biểu đồ giao tiếp (communication diagram).

***Biểu đồ trình tự***

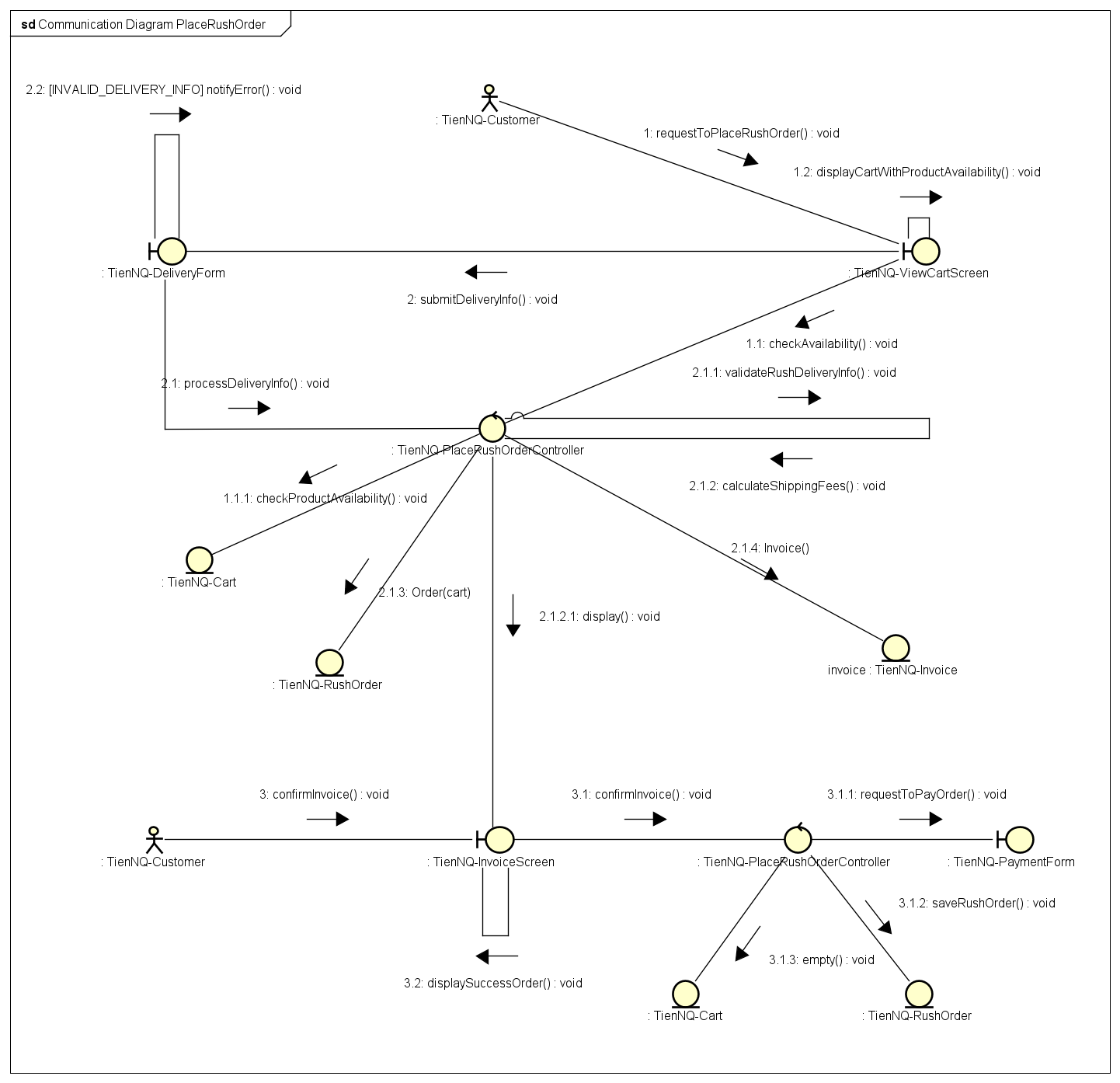
***Kết quả là:***



Hình 3.2 Biểu đồ trình tự Place Rush Order

***Biểu đồ giao tiếp***

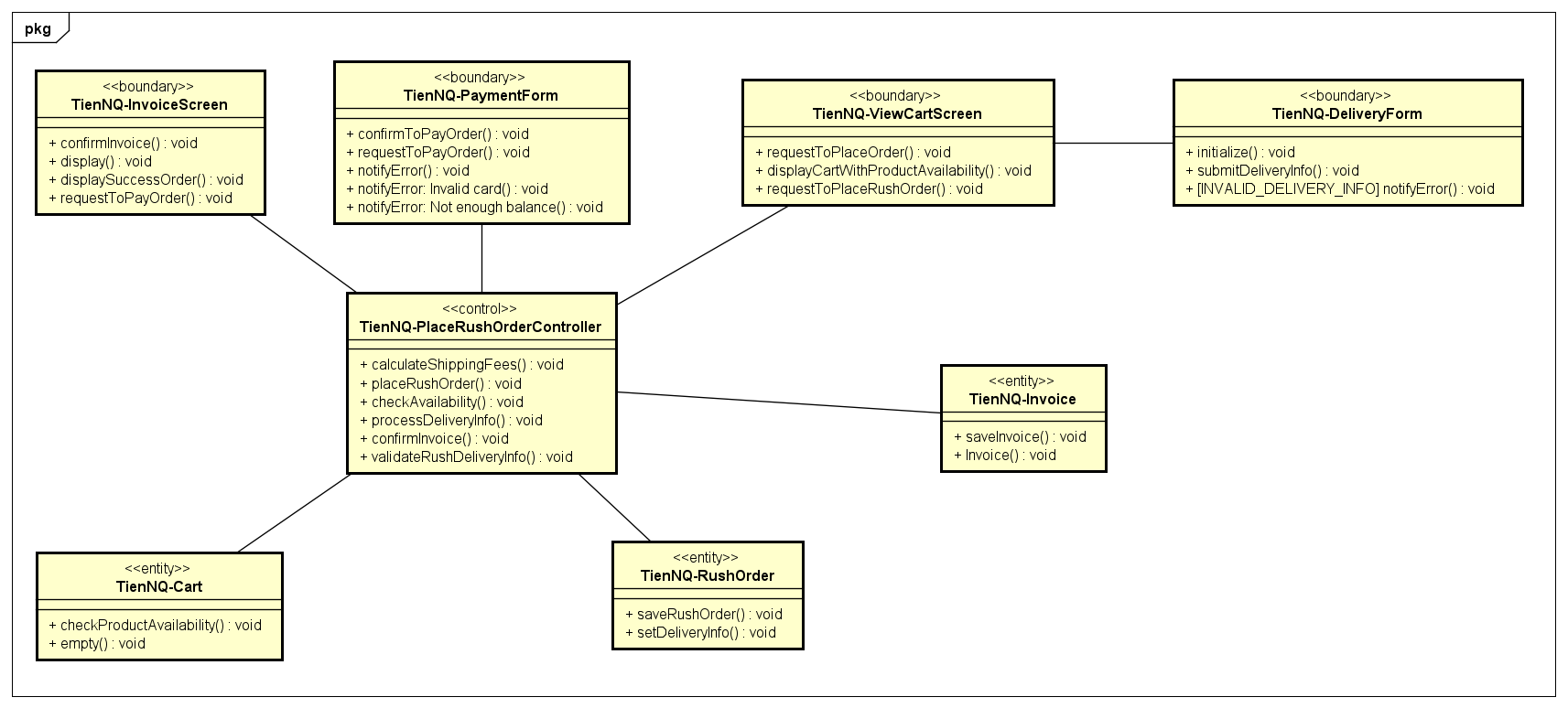
***Kết quả là:***

******

Hình 3.3 Biểu đồ giao tiếp Place Rush Order

**c) Biểu đồ lớp phân tích (analysis class diagram)**

***Kết quả là:***

****

Hình 3.4 biểu đồ lớp phân tích Place Rush Order

**Bài thực hành 02 – Tuần 6, ngày 2/11/2021**

**4. BÀI THỰC HÀNH SỐ 02 – THIẾT KẾ GIAO DIỆN**

**4.3. NỘI DUNG CHI TIẾT**

**4.3.1. Thiết kế giao diện người dùng (User Interface Design)**

Boundary Class được sử dụng để mô hình hình hoá tương tác giữa một hệ thống và môi trường xung quanh. Do đó, chúng có thể được sử dụng để nắm bắt các yêu cầu trên giao diện người dùng. Sự tương tác giữa con người và hệ thống có thể thông qua những loại User Interface (UI) khác nhau như Batch Interface, Command-line Interface (CLI) và Graphical User Interface (GUI). Trong phần này, chúng ta sẽ sử dụng GUI để minh hoạ thiết kế UI từng bước.

**4.3.1.1. Chuẩn hoá cấu hình màn hình**

**Display**

Số lượng màu được hỗ trợ: 16,777,216 màu

Độ phân giải: 1366 x 768 pixels

**Screen**

Vị trí của của button: Ở dưới cùng (theo chiều dọc) và ở giữa (theo chiều ngang) của khung.

Vị trí của message: Ở giữa trung tâm khung màn hình

Vị trí của screen title: Title đặt ở góc trên bên trái của màn hình.

Sự nhất quán trong hiển thị chữ số: dấu phẩy để phân cách hàng nghìn và chuỗi chỉ bao gồm các ký tự, chữ số, dấu phẩy, dấu chấm, dấu cách, dấu gạch dưới và ký hiệu gạch nối.

**Control**

Kích thước text: medium size (24px). Font: Segoe UI. Color: #000000

Xử lý check input: Nên kiểm tra xem input có empty hay không. Tiếp theo, kiểm tra xem input có đúng format hay không.

Dịch chuyển màn hình: Không có các khung chồng lên nhau. Các màn hình được tách biệt. Tuy nhiên, hướng dẫn sử dụng được xem như là 1 popup message vì màn hình chính ở dưới sẽ không thể thao tác trong khi màn hình hướng dẫn sử dụng đang được hiển thị. Ban đầu khi app khởi chạy thì màn hình splash screen (màn hình chớp) sẽ được hiện lên và sau đó màn hình đầu tiên (Home Screen) sẽ xuất hiện.

Thứ tự các màn hình trong hệ thống:

1. Splash screen (first screen)

2. Home screen

3. View cart screen – xem các sản phẩm trong giỏ hàng

4. Delivery form – Điền thông tin giao hàng

5. Invoice screen – Xem chi tiết order

6. Payment form – Điền thông tin thanh toán

7. Result screen

**Nhập input từ bàn phím**

Sẽ không có phím tắt. Có các button quay lại để quay lại các màn hình trước đó. Ngoài ra button “X” nằm ở thanh tiêu đề bên phải để đóng screen

**Error**

Một thông điệp sẽ được hiện lên để thông báo cho người dùng biết vấn đề đang gặp phải là gì.

**4.3.1.2. Tạo các ảnh màn hình**

Tạo ảnh màn hình hoặc bản mockup có thể được thực hiện bằng cách sử dụng các ứng dụng như https://moqups.com/, Figma, InVision Studio, Paint, Adobe Graphic design software, Adobe XD hay Scene Builder…

Các hình ảnh màn hình được tạo ra dưới đây được tạo ra bởi Scene Builder

Text, whiteboard

Description automatically generated

Graphical user interface

Description automatically generated

A picture containing text, screenshot, monitor

Description automatically generated

A picture containing text, screenshot, monitor

Description automatically generated

A picture containing text, screenshot, monitor, computer

Description automatically generated

A picture containing text, screenshot, monitor

Description automatically generated

**4.3.1.3. Tạo các dịch chuyển màn hình**

**Diagram

Description automatically generated**

**4.3.1.4. Mô tả các màn hình**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| AIMS Software | | Date of creation | Approved by | Reviewed by | Person in charge |
| Screen specification | View cart screen | 2/11/2021 |  |  | Nguyễn Quốc Tiến |
|  | | Control | Operation | Function | |
| Area for  displaying the subtotal | Initial | Display the subtotal | |
| Area for display items in the cart | Initial |  | |
| Place order button | Click | Display the Delivery form | |
| Delete button | Click | Remove the item from the cart | |

**Định nghĩa các trường thuộc tính**

|  |  |
| --- | --- |
| Screen name | View cart |
| View cart | Number of digits (bytes) | Type | Field attribute | Remark |
| Media title | 50 | Numeral | Blue | Left-justified |
| Price | 20 | Numeral | Blue | Right justified |
| Subtotal | 20 | Numeral | Blue | Left-justified |

**4.3.2. Thiết kế giao diện hệ thống (System Interface Design)**

**4.3.2.1. Tìm ra các subsystem**

Có thể thấy, InterbankBoundary class trong Analysis Class Diagram cung cấp các dịch vụ phức tạp liên quan đến giao tiếp giữa AIMS Software và Interbank. Ngoài ra, nó cũng là một External System Interface, nó độc lập, gắn chặt với actor Interbank. Do đó, ta cần chuyển InterbankBoundary từ một Analysis Class thành một Subsystem.

Diagram

Description automatically generated

**4.3.2.2. Thiết kế interface cho subsystem**

Dựa vào trách nhiệm chính của một hệ thống thanh toán, chúng ta có thể xác định được interface cho Subsystem như sau:

A picture containing text

Description automatically generated

**4.3.2.3. Thiết kế Subsystem**

**Distribute subsystem behavior to subsystem elements**

**Diagram

Description automatically generated**

**Diagram

Description automatically generated**

**Diagram

Description automatically generated**

**Document subsystem elements**

A picture containing diagram

Description automatically generated

**Describe subsystem dependencies**

Diagram

Description automatically generated

**Checkpoints**

**Diagram

Description automatically generated**

**4.3.3. Thiết kế giao diện Place Rush Order (Place Rush Order Interface Design)**

A picture containing text, screenshot, monitor, computer

Description automatically generated

Ở màn hình shipping ta thay đổi phần shipping intructions sao cho người dùng chỉ có thể chọn 1 trong hai cách thức chuyển là Normal shipping (vận chuyển thường) và Quick shipping (vận chuyển nhanh).