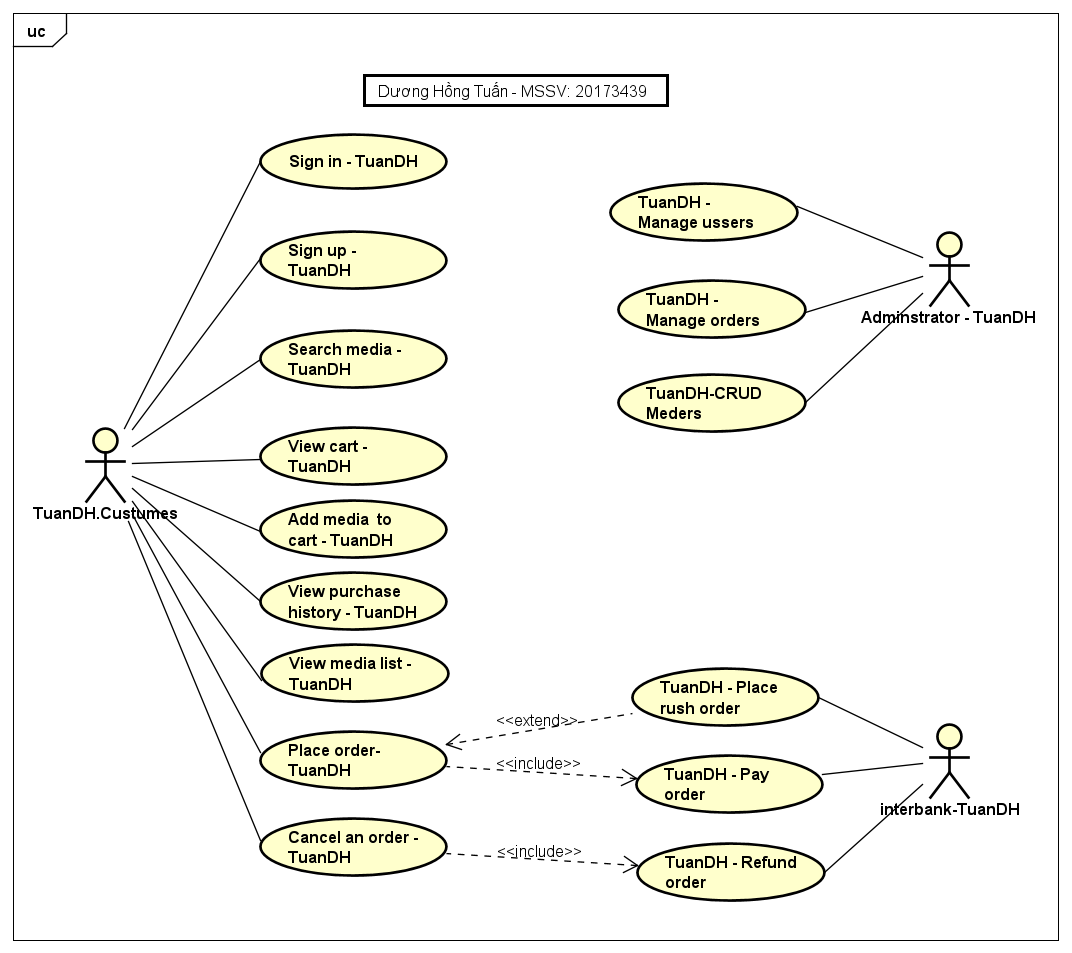
Đánh giá điểm thực hành:

* 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.

Use case tổng quan



**Ví dụ:**

**Bài thực hành 01 – Tuần 4, ngày**

* Link đã nộp bài lên Git/GitHub.
* 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>

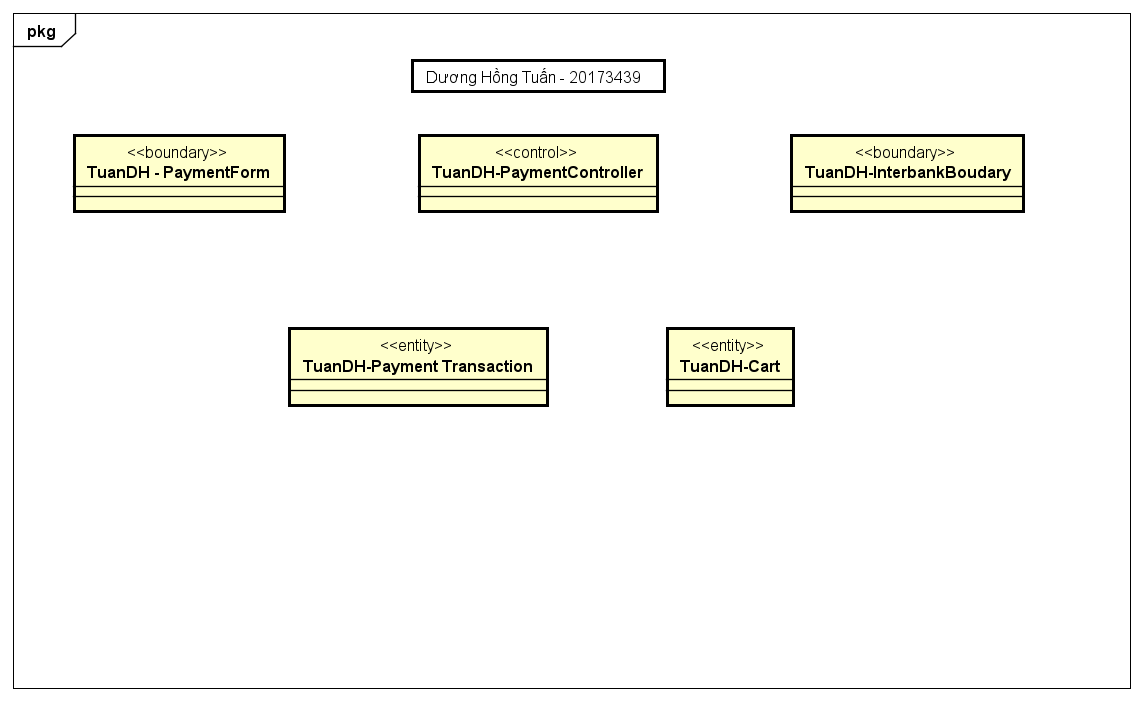
### 3.3.3. Thiết kế kiến trúc cho use case “Pay Order”

Chụp ảnh kết quả các bước đã thực hiện của sinh viên theo mẫu

**Phần này sẽ mô tả từng bước thiết kế kiến trúc cho use case “Pay Order”. Tất cả kết quả trong bài thực hành này được nộp vào thư mục “ArchitecturalDesign” trên repository cá nhân của sinh viên.**

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

**Kết quả là**:

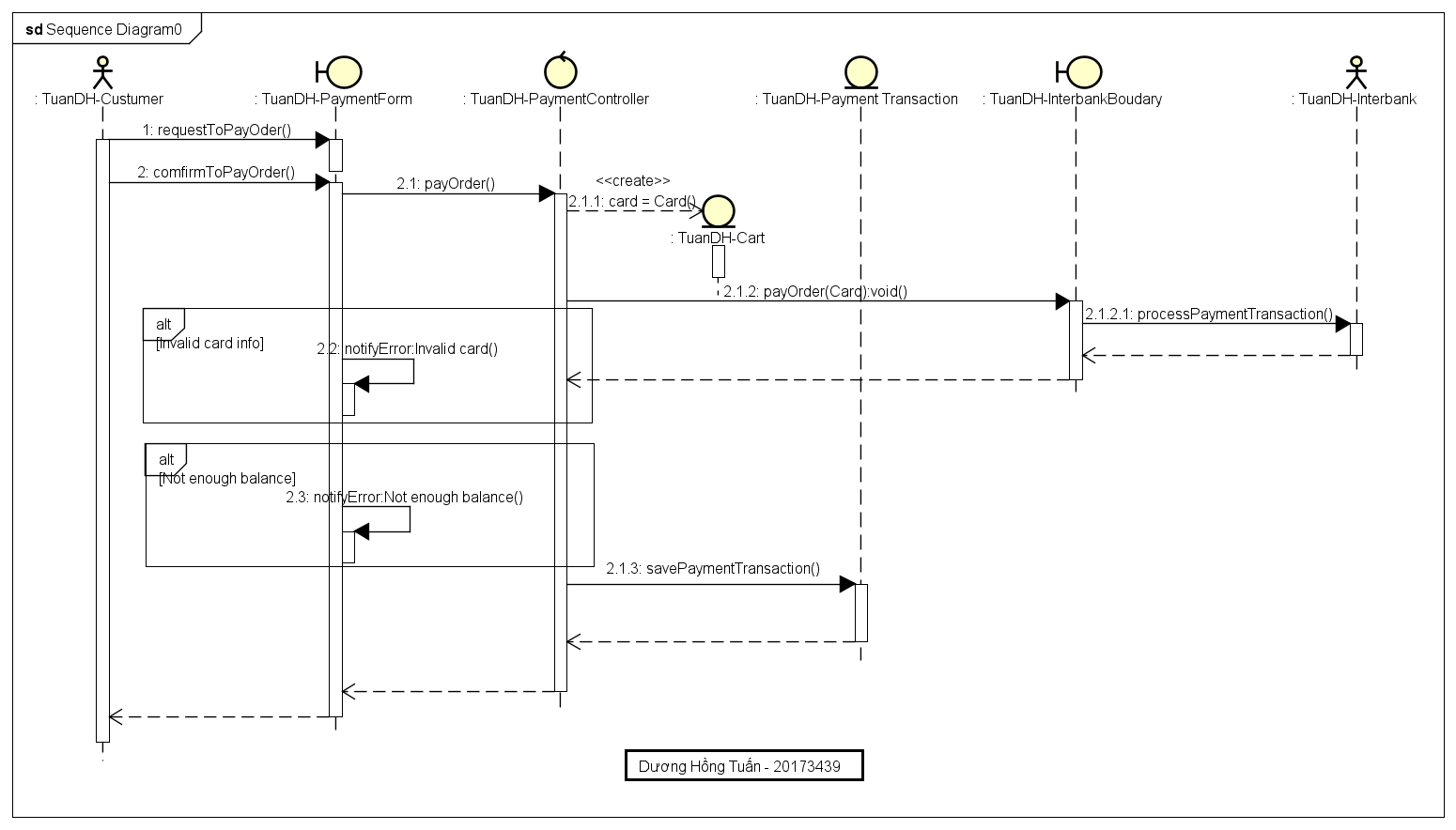


Hình 1 Biểu đồ lớp phân tích PayOrder

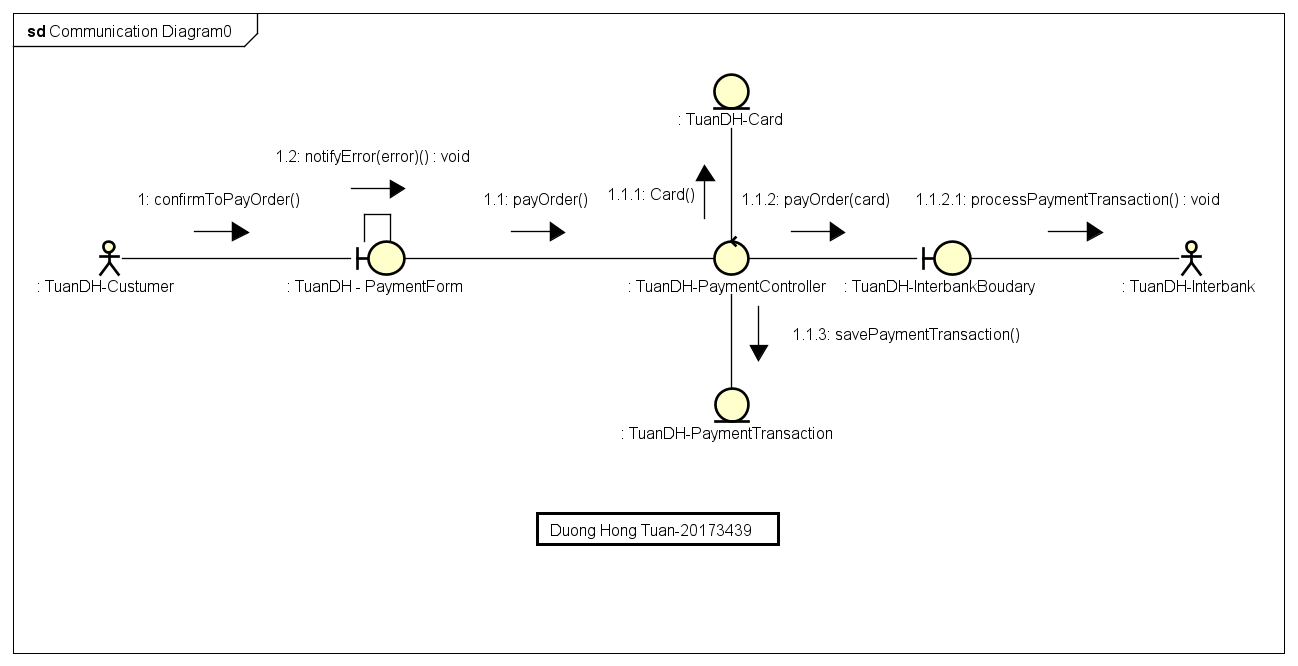
1. **Phân phối hành vi trong use case tới các lớp.** Phân bổ trách nhiệm tới các lớp và mô hình hóa mối quan hệ giữa các lớp bằng cách sử dụng biểu đồ tương tác (interaction diagram). Chúng ta có thể sử dụng biểu đồ trình tự (sequence diagram) **hoặc/và** biểu đồ giao tiếp (communication diagram).

***Biểu đồ trình tự***

***Kết quả là:***



Hình 2 Biểu đồ trình tự PayOrder

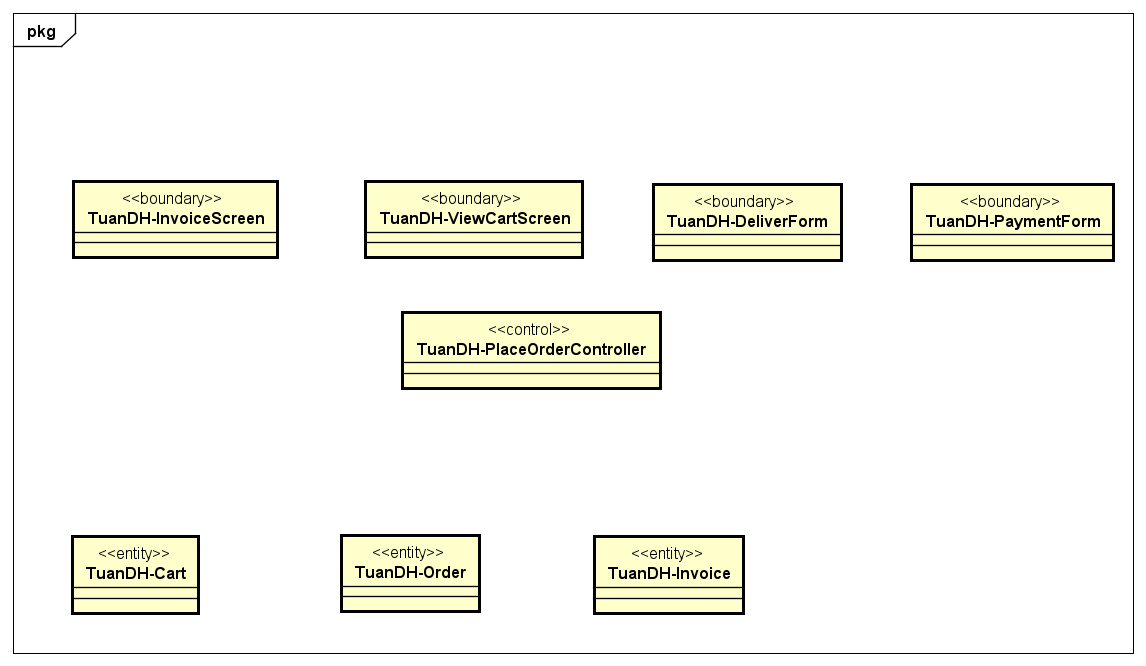


Hình 3 Biểu đồ Giao Tiếp PayOrder

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

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

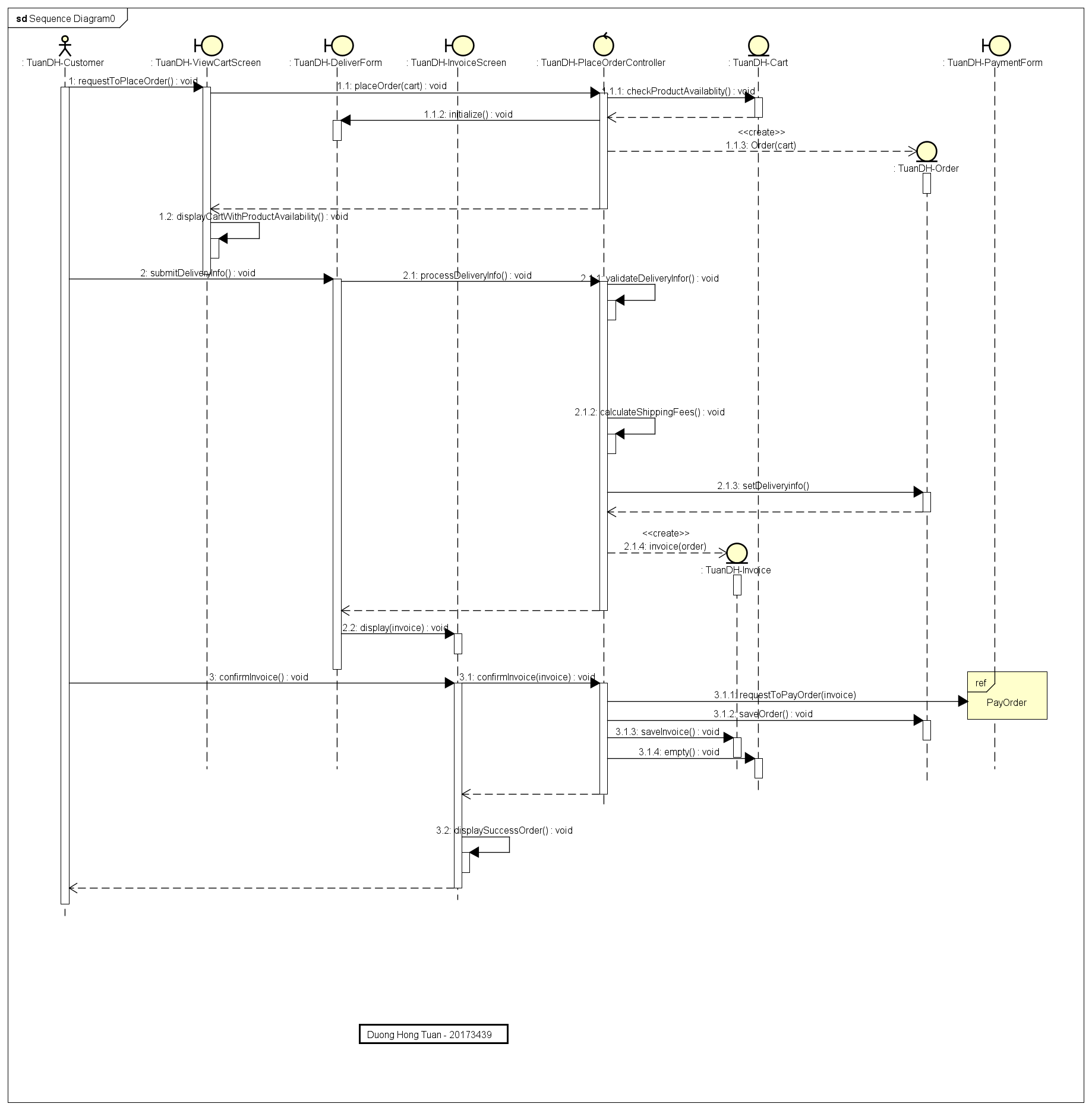
**Kết quả là**:



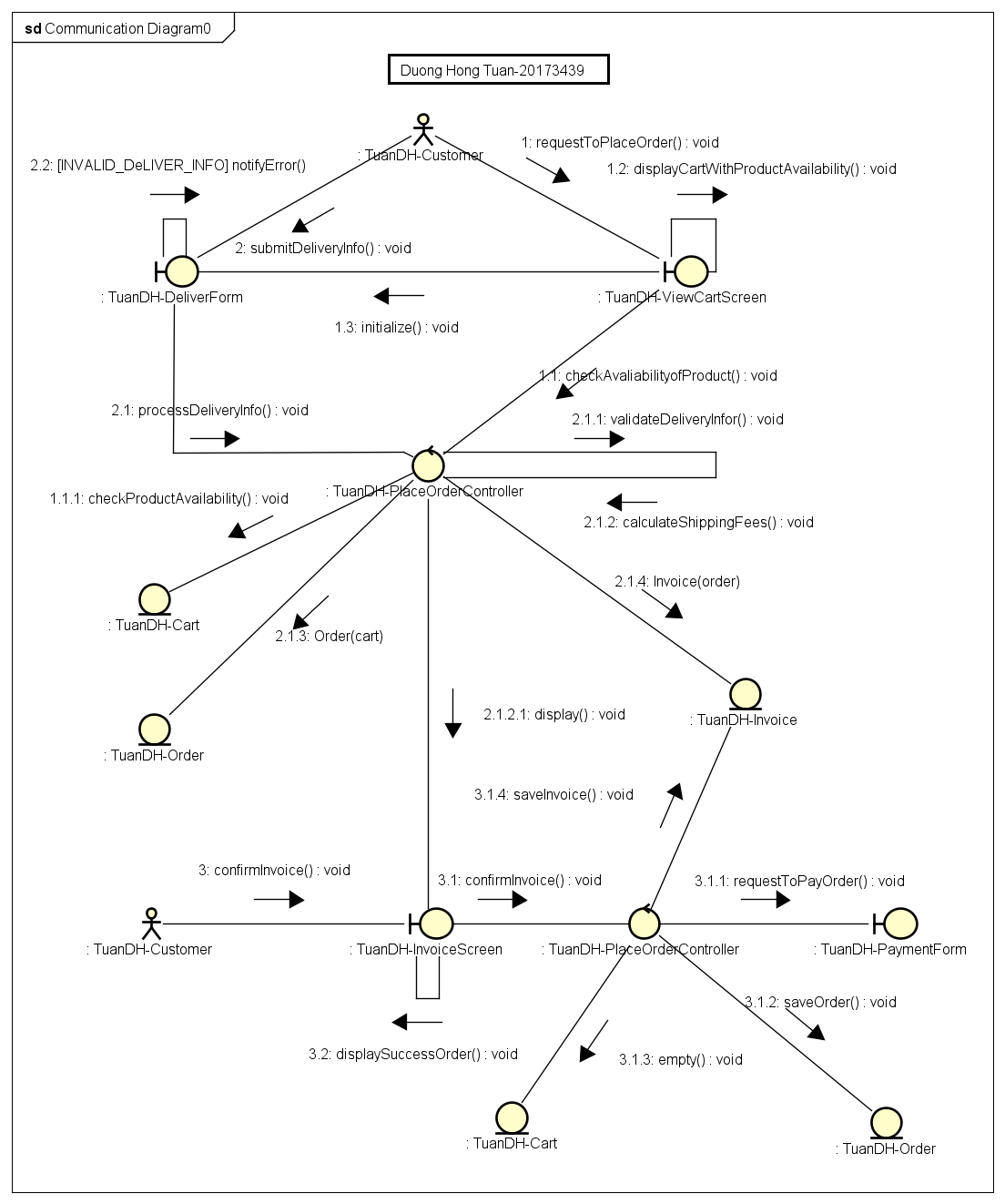
Hình 4 Biểu đồ lớp phân tích PayOrder

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***

******

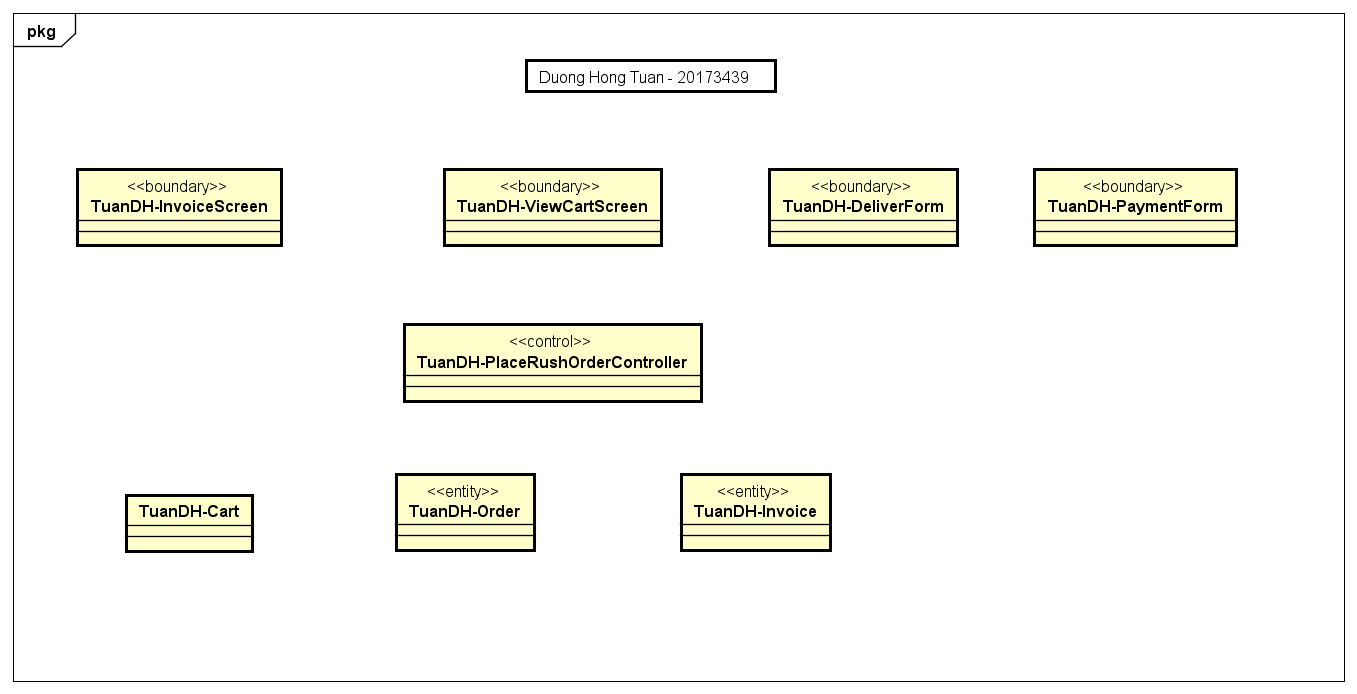
***Biểu đồ giao tiếp***

******

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

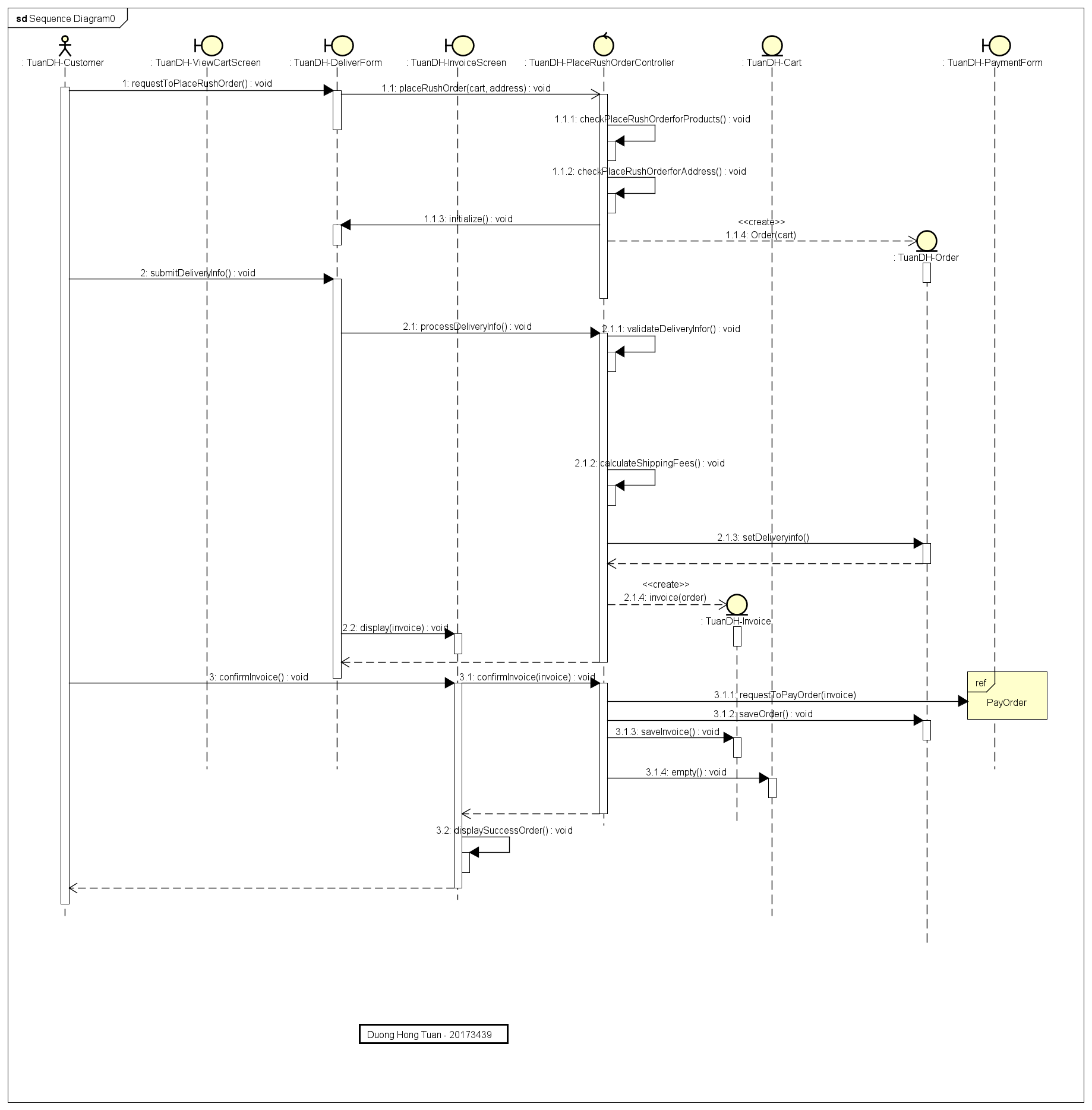
1. **Phân tích lớp**: Tìm các lớp (class) từ các hành vi trong use case.

**Kết quả là**:

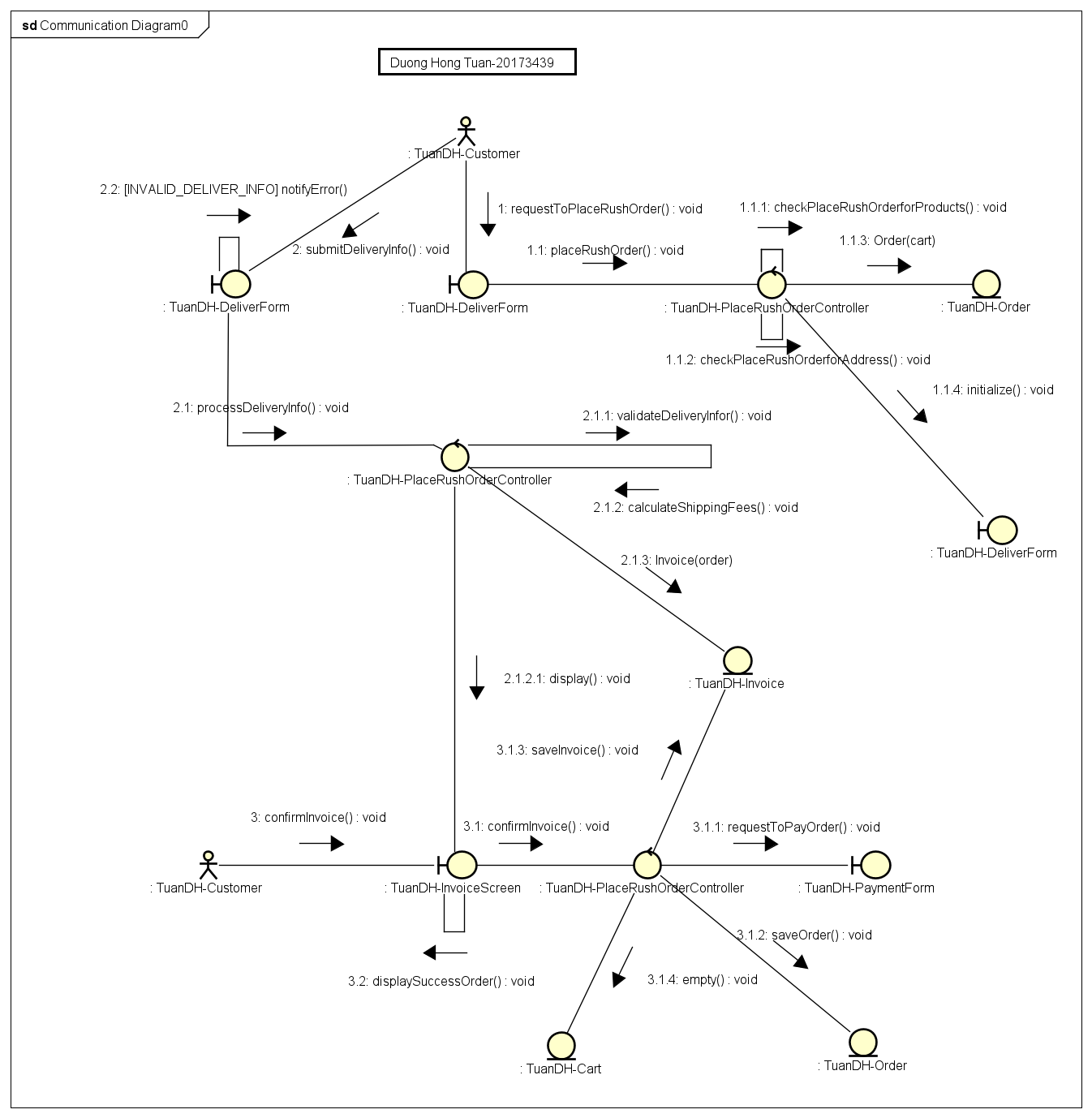


1. **Phân phối hành vi trong use case tới các lớp.** Phân bổ trách nhiệm tới các lớp và mô hình hóa mối quan hệ giữa các lớp bằng cách sử dụng biểu đồ tương tác (interaction diagram). Chúng ta có thể sử dụng biểu đồ trình tự (sequence diagram) **hoặc/và** biểu đồ giao tiếp (communication diagram).

***Biểu đồ trình tự***

******

***Biểu đồ giao tiếp***



### **3.5 Biểu đồ lớp phân tích gộp theo những cập nhật của biểu đồ lớp phân tích cho use case “Place Order” cùng biểu đồ lớp phân tích của use case “Place Rush Order”**

