HUST

ĐẠI HỌC BÁCH KHOA HÀ NỘI

HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

ONE LOVE. ONE FUTURE.

ĐẠI HỌC
BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY
OF SCIENCE AND TECHNOLOGY

AIMS: An Internet Media Store

ITSS SOFTWARE DEVELOPMENT - IT4549E

Instructor: Ph. D Nguyen Thi Thu Trang

Group 17

Đinh Việt Quang	20215235
Ngô Minh Quý	20215238
Trịnh Diễm Quỳnh	20210737
Hồ Nam Sơn	20215239
Lê Phú Tài	20210759

ONE LOVE. ONE FUTURE.

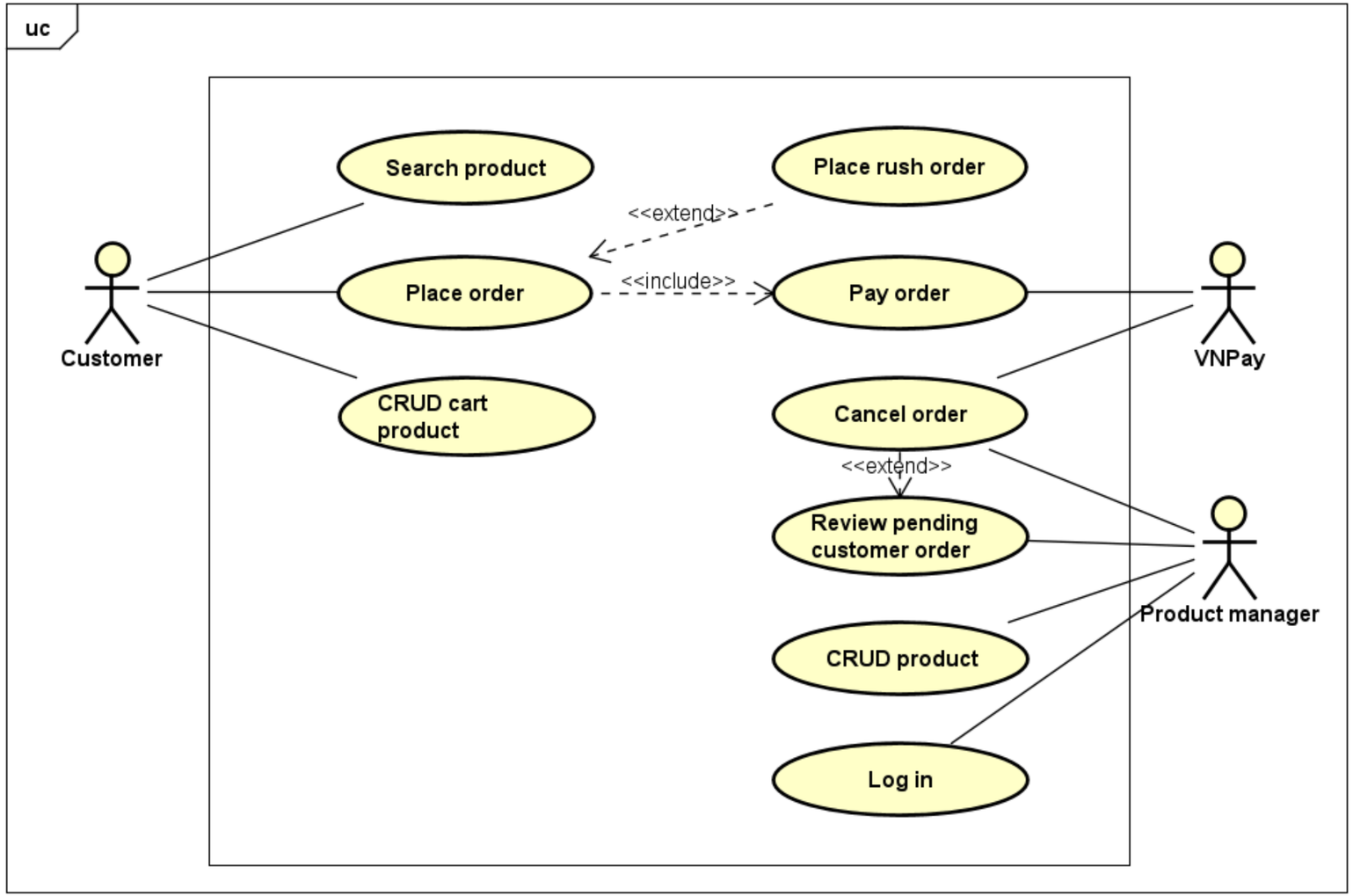
CONTENTS

1. SUMMARY OF COMPLETED FEATURES

We have successfully implemented these following features:

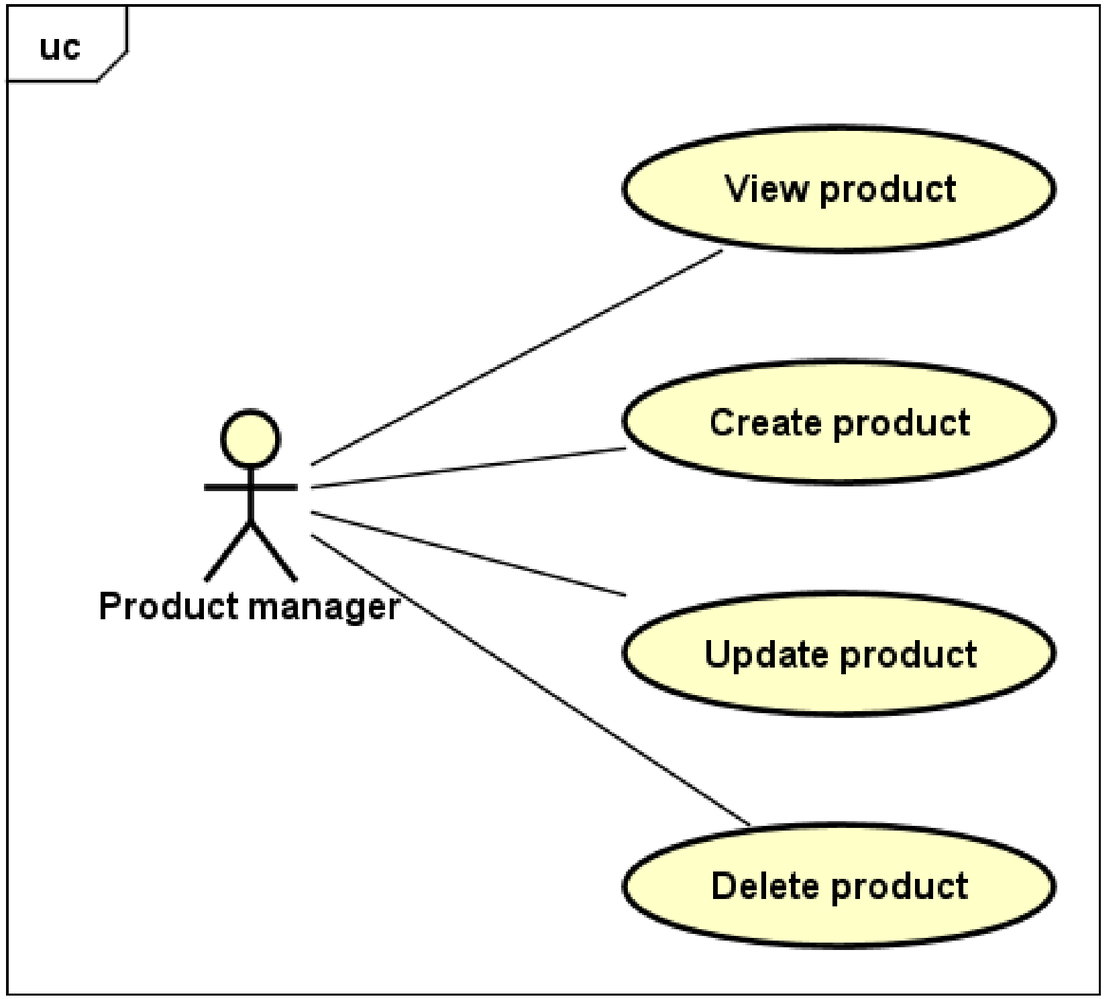
1. Search for products by keywords
2. Add products to cart
3. View cart
4. Place order/rush order
5. Make payment via VNPay
6. Login/Logout for manager
7. Manage products as admin
8. Manage orders as admin

Use case diagram



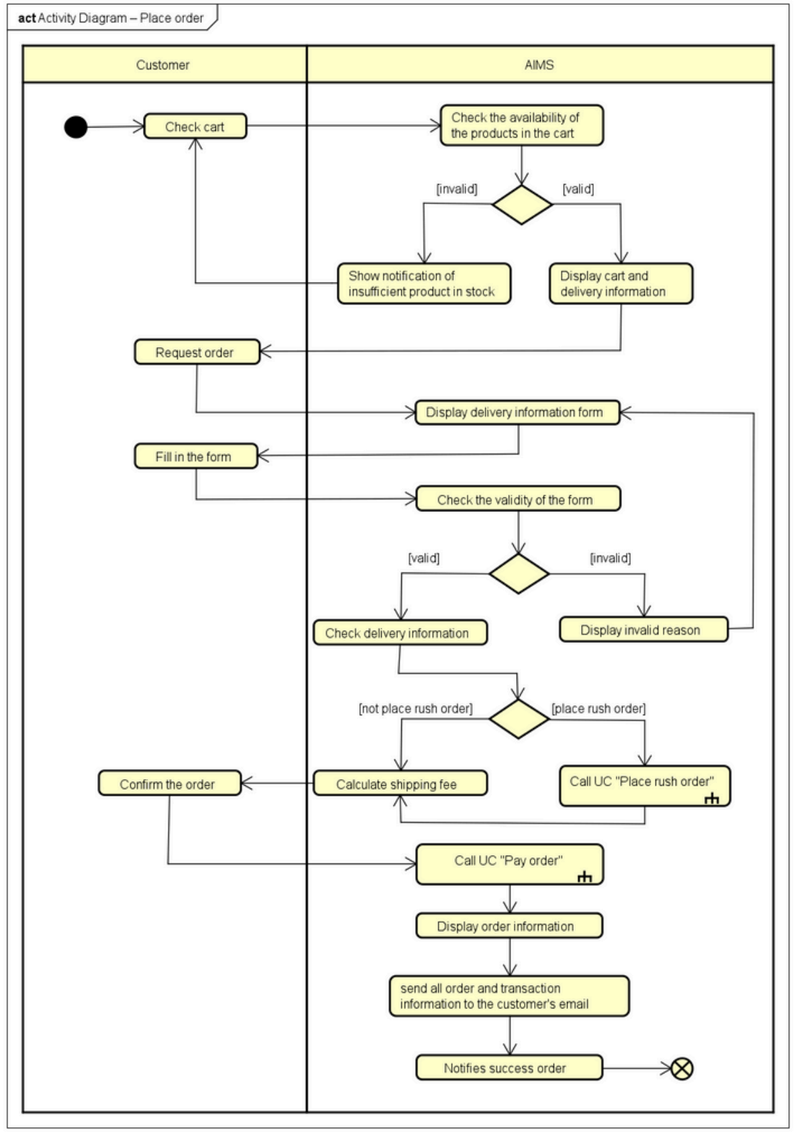
Use case diagram

- CRUD product



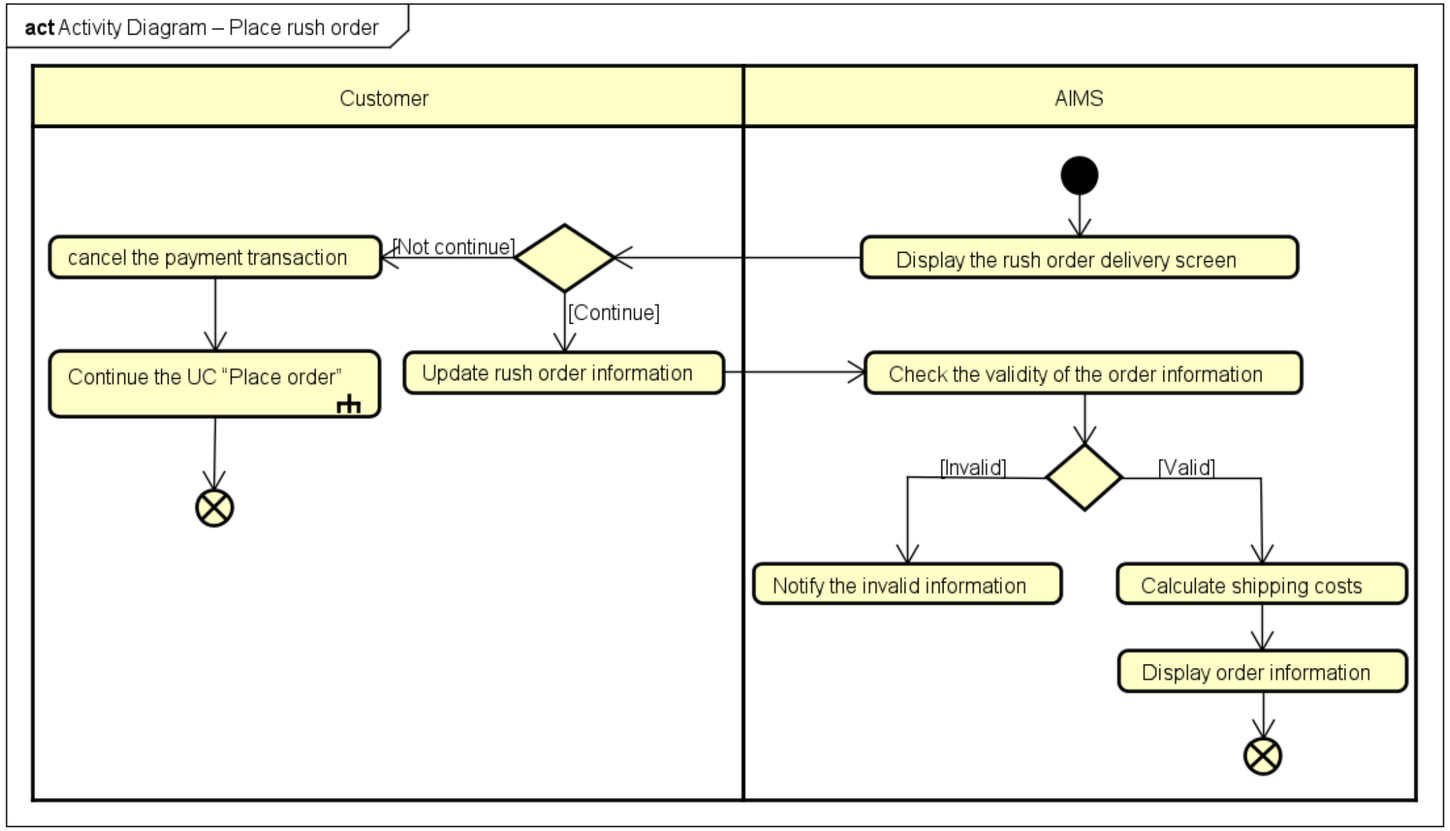
Activity diagram

- Place order



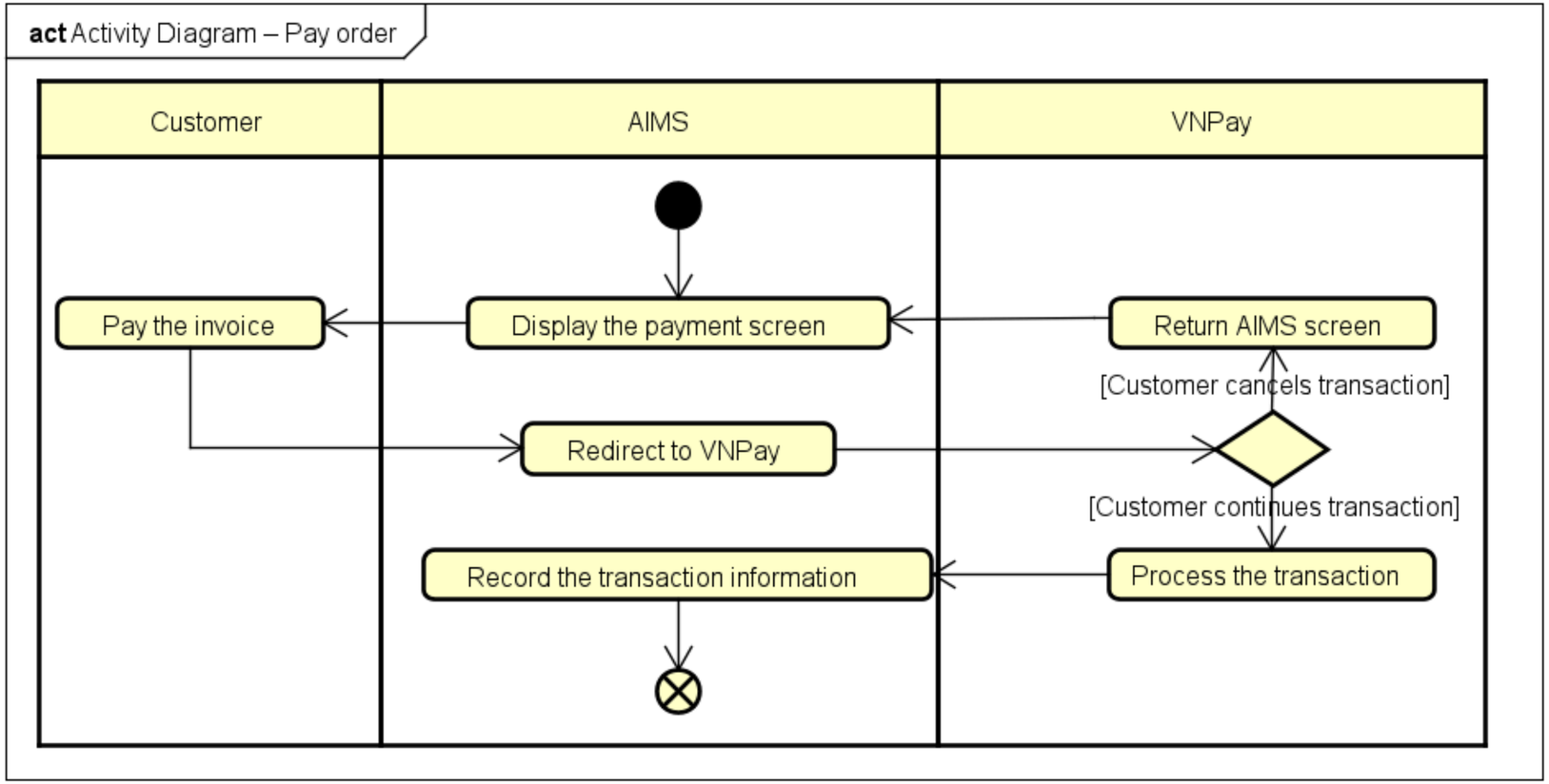
Activity diagram

- Place rush order



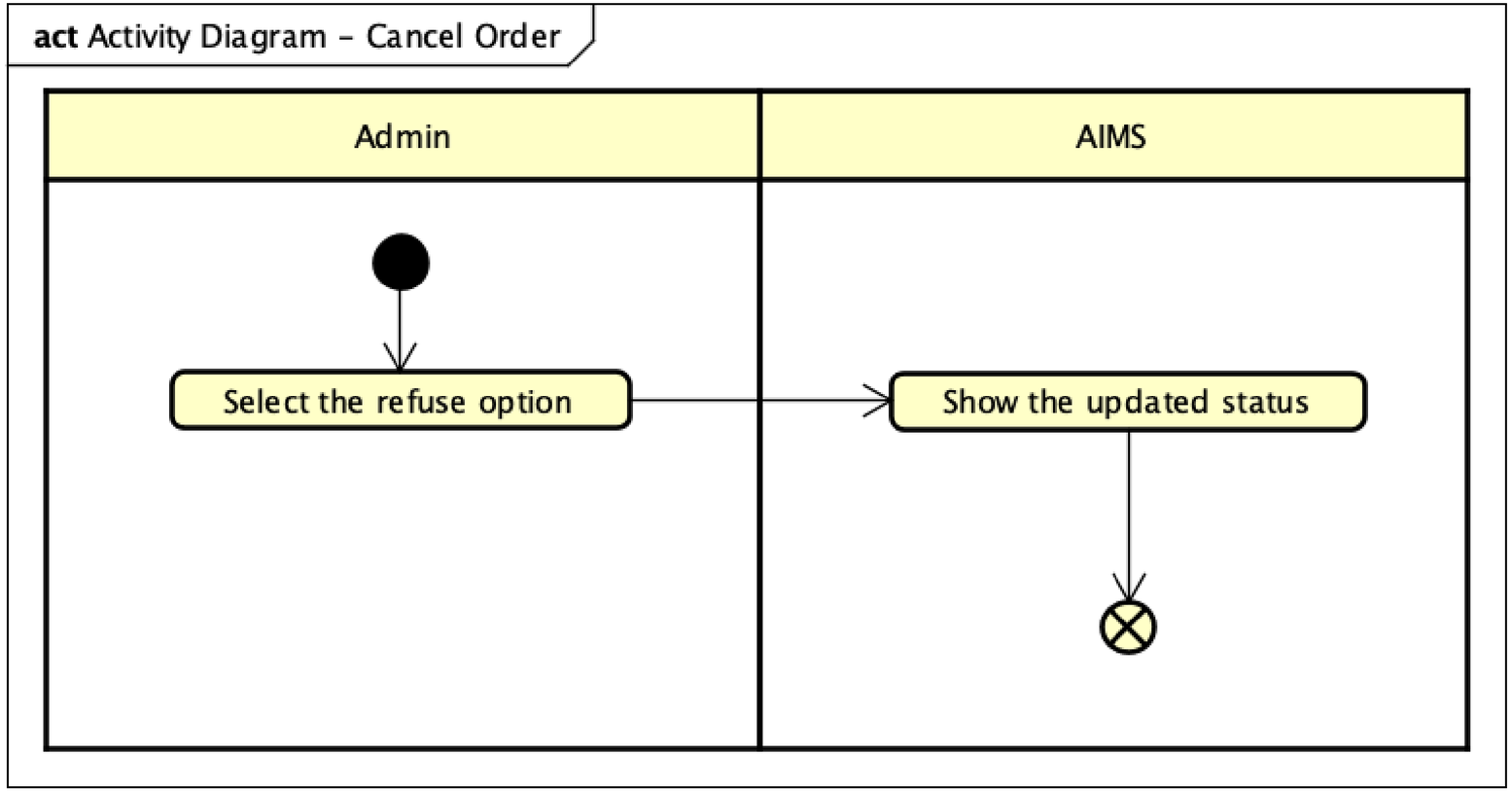
Activity diagram

- Pay order



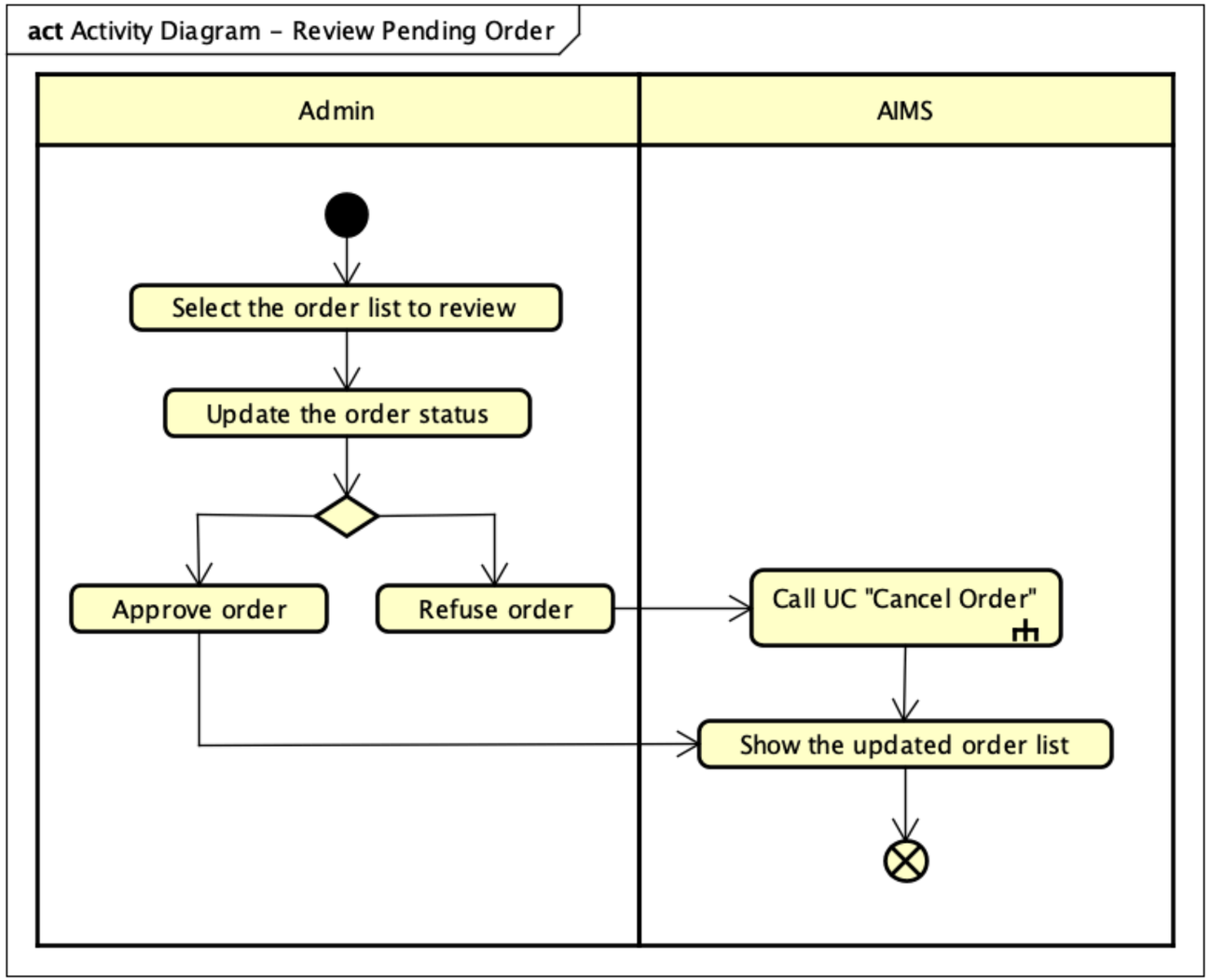
Activity diagram

- Cancel order



Activity diagram

- Review pending order



3. DETAILED DESIGN

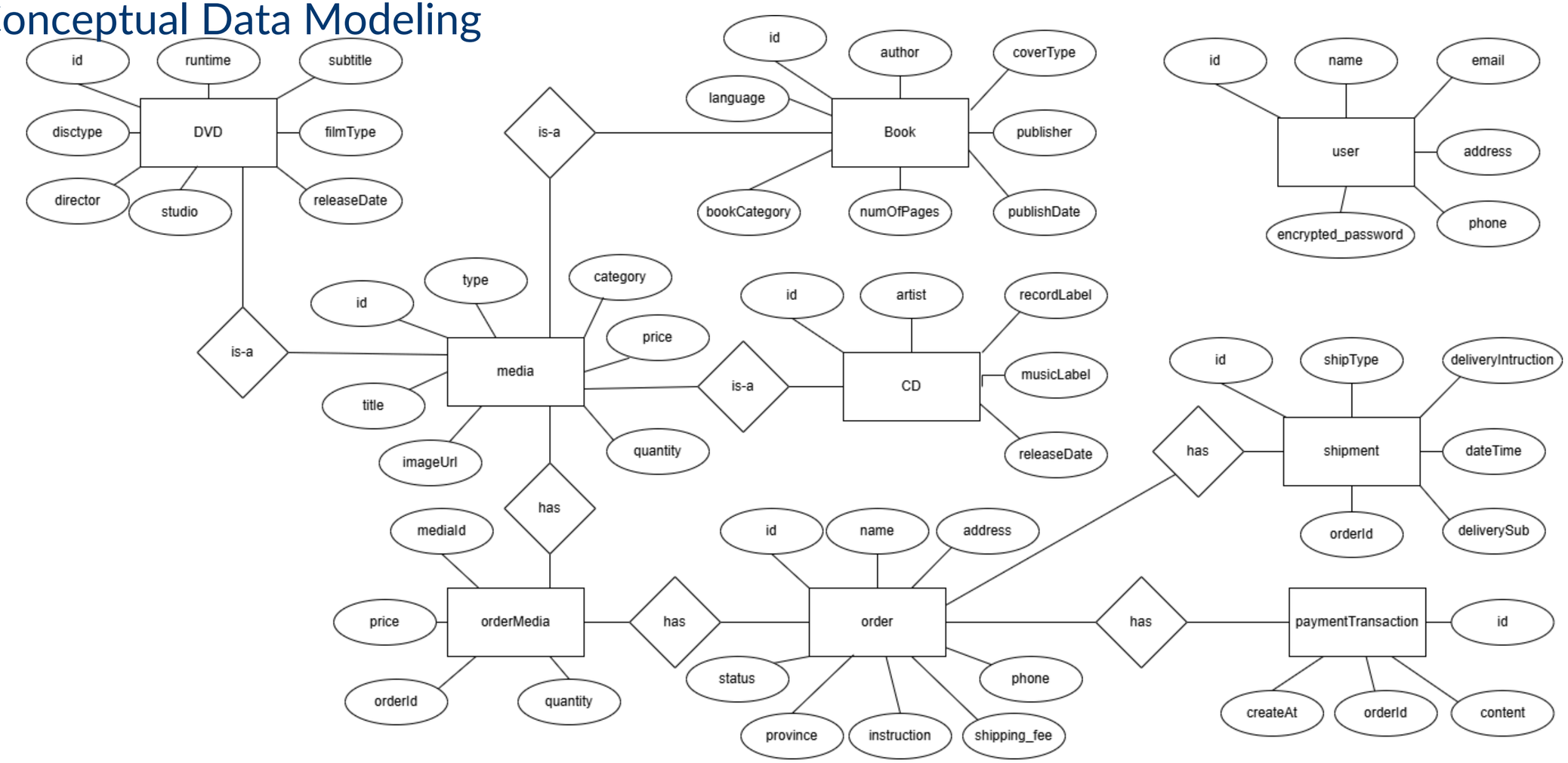
Interface design

- Screens in AIMS application: Home screen, Cart screen, Shipping screen, Login screen, Product manager dashboard, VNPay interface.
- Display: resolution 1366x768.
- Consistency of expressions: a limited range of valid characters for strings ([0-9],...).
- Control: input format checking functionality; each screen is separated and no stack framed exists.
- Error: error message with its details displayed while encountering such one.

3. DETAILED DESIGN

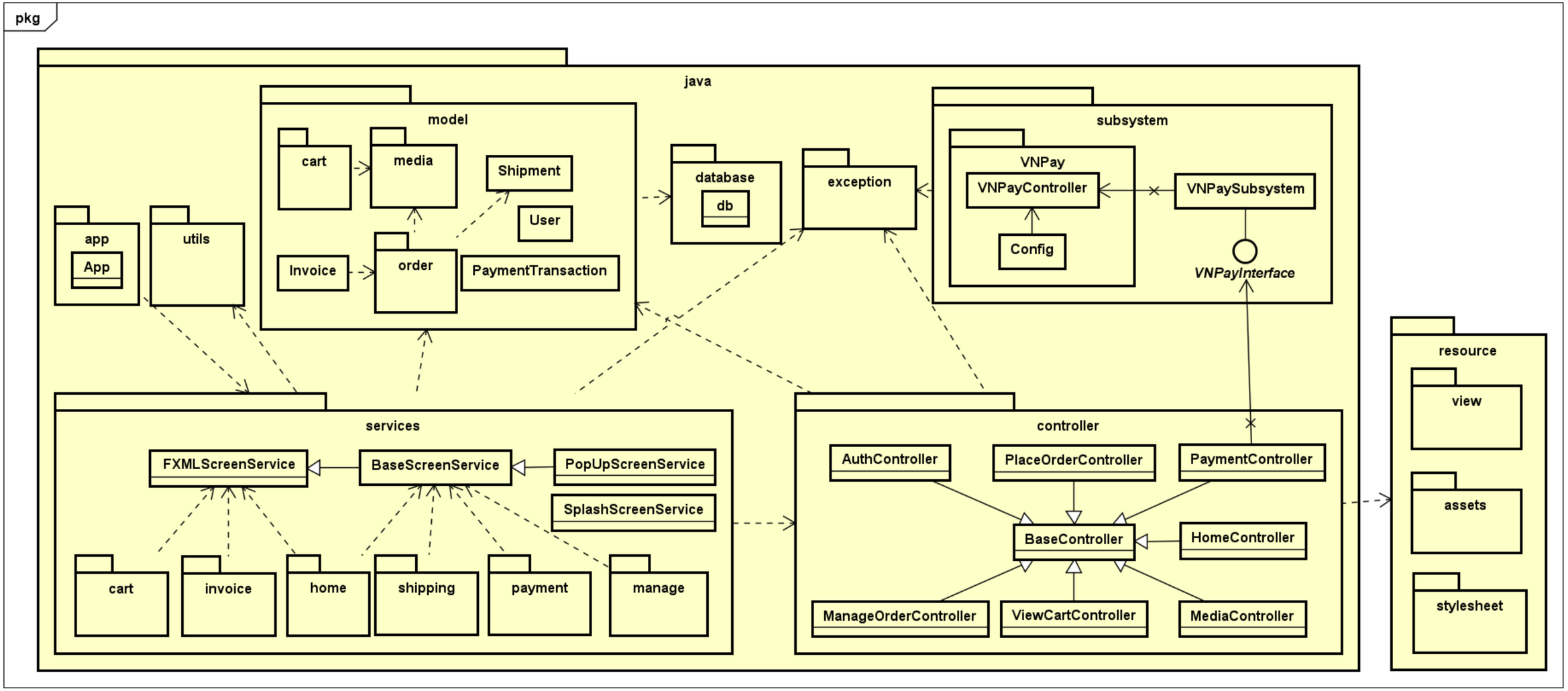
Data modelling

- Database management system: SQLite
- Conceptual Data Modeling



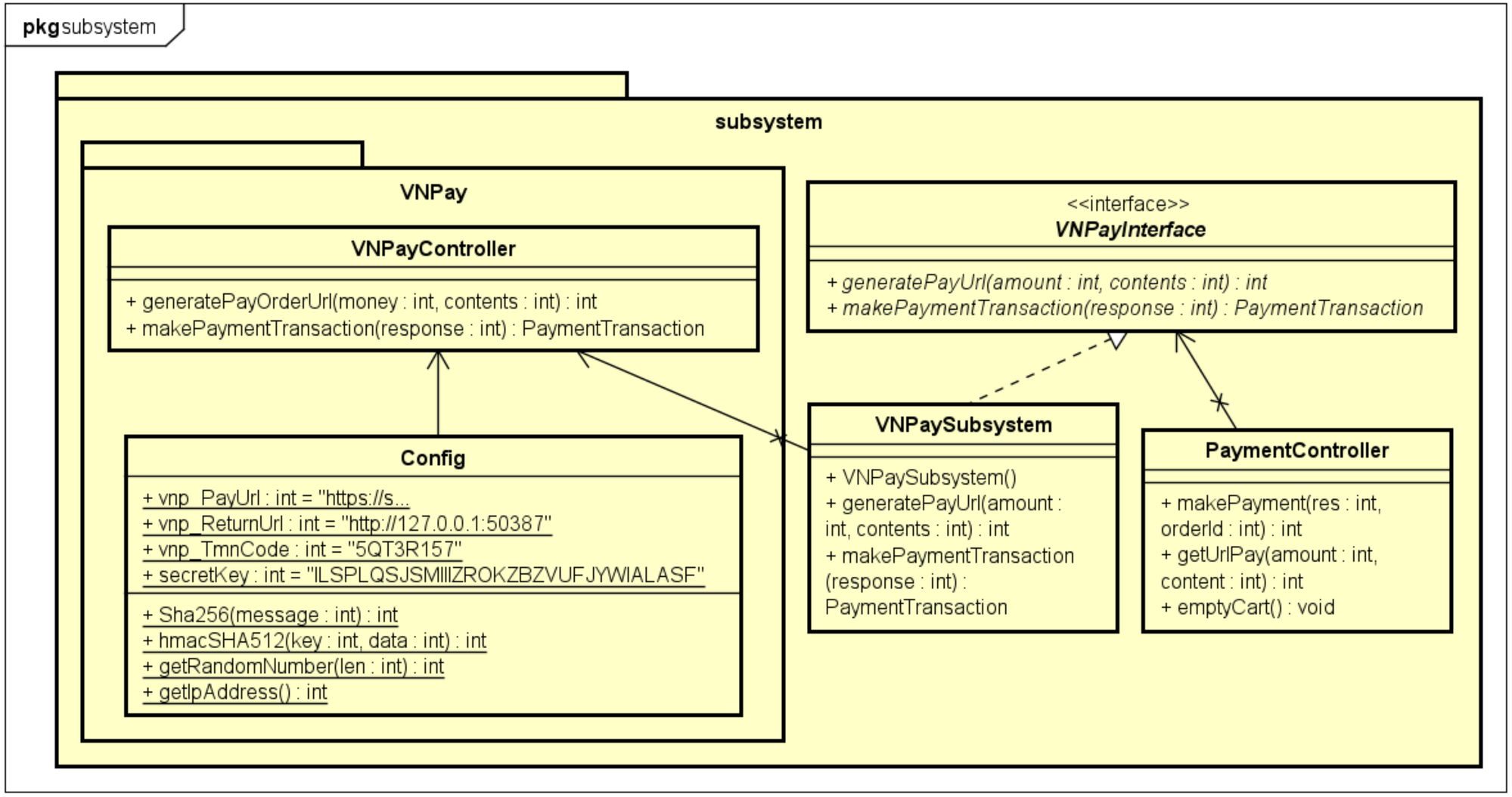
3. DETAILED DESIGN

General class diagram



3. DETAILED DESIGN

VNPay subsystem



4. DESIGN ASSESSMENT

Architectural strategies

- Programming languages and framework: Java, JavaFX for GUI with Maven.
- Architectural pattern: MVC (Model - View - Controller).
- Error handling: Ultilize exception package.
- Version control: Github workflow (develop, feature, release, hotfix).

Objectives

- Ensure the application development process and simplify dependency management.
- Enhance separation of concerns, code maintenance scalability.
- Code quality and consistency for streamline development process.

4. DESIGN ASSESSMENT

Coupling

- Our project does not violate common coupling.

- Our project violates stamp coupling at Controller module, where HomeController passes the Model object and ViewCartController passes the Cart object, but do not use all of its properties.

Cohesion

- Temporal cohesion: The App class performs the function of displaying Splash screen first and then displaying Home screen.

- Communicational cohesion: MediaController, ViewCartController, ManageCartController have different methods of performing functions but they all operate on the same Order, Cart data.

- Functional cohesion: Most of the modules in the project satisfies.

4. DESIGN ASSESSMENT

Design principle

- *Single Responsibility Principle* (SRP): some classes in controller package such as ManageOrderController or PlaceOrderController violate this principle because they perform many different functions for managing order or placing order.

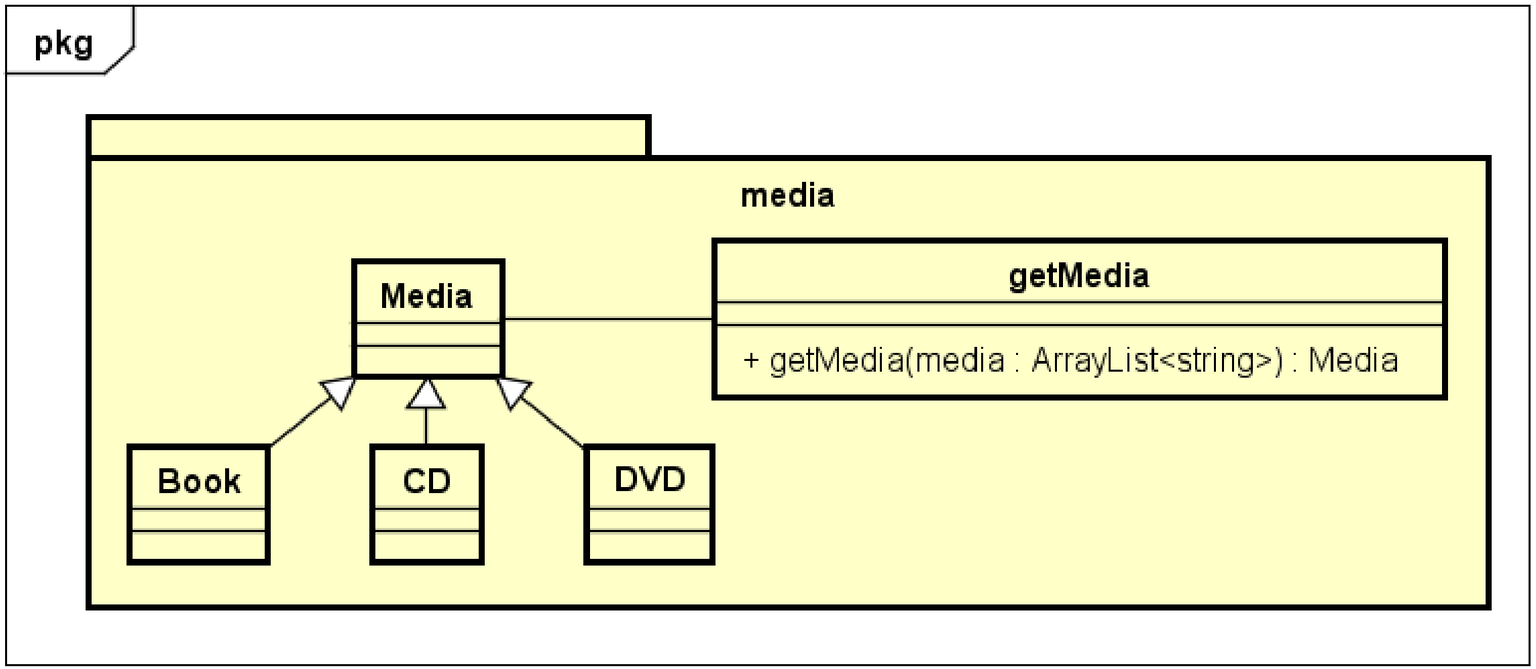
- *Open/Closed Principle* (OCP): FormScreenService inheritance for different kinds of Media.

- *Liskov Substitution Principle* (LSP): The inheritance hierarchy in the Media class follows this principle.

- *Interface Segregation Principle* (ISP): The project does not violate this principle because no interface has too many methods.

- *Dependency Inversion Principle* (DIP): screen display classes in services package depend quite strictly on several model classes, which leads to this principal violation.

Design pattern proposal - Factory



Benefits: reduce dependencies between modules, easy to add more subclasses media in the future.

HUST

hust.edu.vn fb.com/dhbkhn

THANK YOU FOR LISTENING!