**Câu 1:**

Diagram

Description automatically generated

1. Giải thích thành phần và quan hệ của DFD 1

* Thực thể: Customer, Kitchen, Supplier, Manager
* Tiến trình: Order Food, Generate Reports, Order Inventory
* Data store: Order Food, Generate Reports, Order Inventory
* Hoạt động của một nhà hàng
  + Mỗi lần Customer thực hiện order, order được xử lý sau đó:
    - Tạo Bill cho Customer
    - Gửi order đến cho Kitchen.
    - Lưu order vào Database
    - Cập nhật thông tin chi tiết trong Kho hàng
  + Định kỳ, dữ liệu từ Kho hàng và order sẽ được lấy ra để tạo và gửi cho Manager.
  + Manager thực hiện nhập thêm hàng cho Kho hàng:
    - Order hàng từ các Supplier
    - Cập nhật thông tin chi tiết trong Kho hàng

**Câu 2:**

* 1. Draw in sheet of paper
  2. Explain entity, controller....method in the diagramactor: Khách hàng

giao diện: TimkiemSachForm, ChitietSachForm, GiohangFrm

phần control: SachCtr, GiohangCtr

các thực thể: Sach, Giohang, Khachhang

* 1. What is still not suitable to UML convention ?
* Hướng mũi tên trả về thường ở dạng < -------
* Tổng độ dài của Message phải bằng của Reply Message (trên hình thiếu phần reply từ giao diện về cho Khachhang

Chart

Description automatically generated

Câu 3,

1. *Draw two diagrams in a sheet of paper*
2. *Explain notations, flow…*

|  |  |
| --- | --- |
| Diagram  Description automatically generated | Hoạt động |
|  | Trạng thái bắt đầu |
|  | Trạng thái kết thúc |
|  | Thanh đồng bộ hoá |
|  | Chuyển tiếp |
|  | Quyết định |
| Order Process | Khi nhận được order sẽ diễn ra song song 2 hoạt động:   * Hoạt động 1: Điền order. Nếu order cần vận chuyển nhanh thì sắp xếp order vào loại vận chuyển qua đêm * Hoạt động 2: Gửi lại hoá đơn và nhận thanh toán   Hoàn thành 2 hoạt động trên, đóng order |
| A process for student enrollment in a university | Khi sinh viên muốn ghi danh vào trường, sinh viên điền vào form ghi danh   * Nếu form điền đúng hoặc sai sót không đáng kể, sinh viên sẽ ghi danh vào trường thành công * Nếu form điền chưa đúng, sinh viên thực hiện điền lại hoặc nhờ đến sự hỗ trợ để điền form   Sau khi ghi danh vào trường thành công, sinh viên tiếp tục thực hiện 2 hoạt động song song:   * Hoạt động 1: Tham gia vào bài thuyết trình giới thiệu tổng quan về trường đại học * Hoạt động 2: Đăng ký vào các khoá học và thực hiên thanh toán học phí   Sau khi hoàn thành 2 hoạt động, kết thúc |

1. *Draw them in VP*

Diagram

Description automatically generated

Diagram

Description automatically generated

Bài 4

1. *Draw the diagram in a sheet paper*
2. *Add attributes into classes*
3. *Explain relationships (why? And why?)*

|  |  |
| --- | --- |
| Diagram  Description automatically generated | Customer và Order có quan hệ Association vì đây là 2 lớp độc lập có liên quan đến nhau. 1 Customer có thể có nhiều Order |
| Diagram  Description automatically generated | Order và OrderDetail có quan hệ Aggregation vì chúng là tổng thể - bộ phận. OrderDetail là bộ phận của Order. 1 Order gồm nhiều OrderDetail |
| Diagram  Description automatically generated | Item và OrderDetail có quan hệ Association vì đây là 2 lớp độc lập có liên quan đến nhau. 1 Item có thể thuộc nhiều OrderDetail |
| Diagram  Description automatically generated | Payment và Order có quan hệ Association vì đây là 2 lớp độc lập có liên quan đến nhau. 1 Order có thể có nhiều Payment khác nhau |
| Diagram  Description automatically generated | Payment với (Cash, Check, Credit) là quan hệ kế thừa vì Payment có có hình thức (Cash, Check, Credit) và 3 hình thức này kết thừa những thuộc tính chung của Payment |

1. *Draw the diagram with additional attributes in VP*

Diagram

Description automatically generated

**Ex5.**

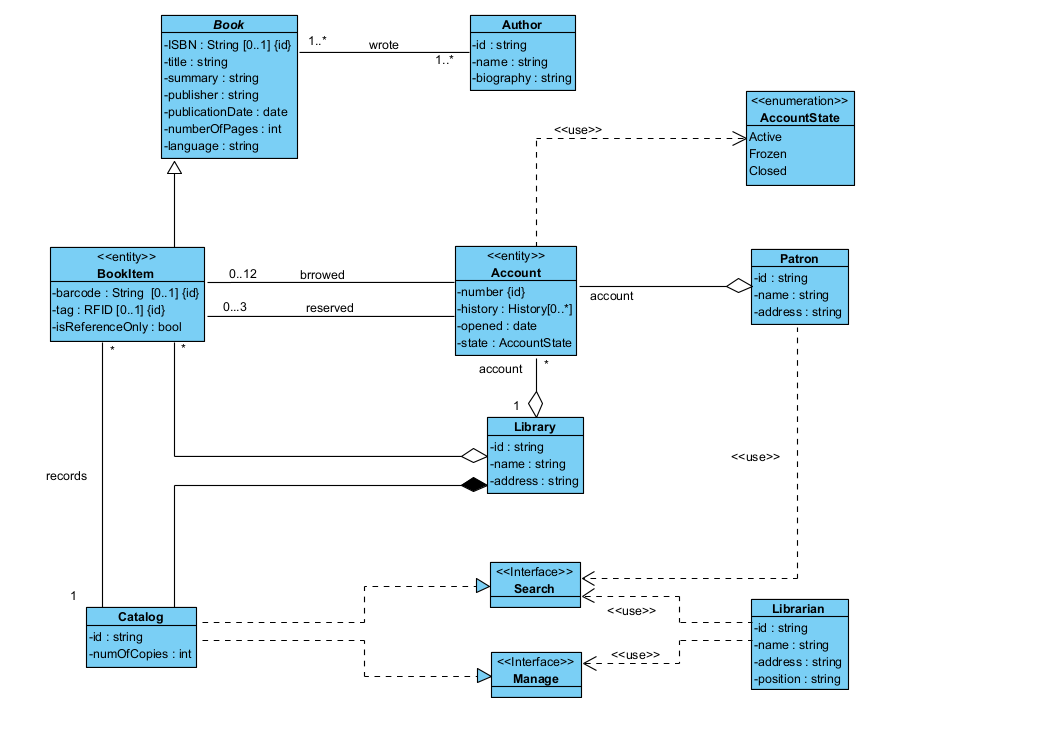
Diagram

Description automatically generated

1. *Draw the diagram in a sheet of paper*
2. *Add attributes*
3. *Explain relationships (why? And why?)*

|  |  |
| --- | --- |
| Book - Author | Association vì Book và Author là 2 lớp độc lập có quan hệ Author viết 1 số quyển sách nào đó.  1 Author có thể viết nhiều Book và 1 Book có thể được viết bởi nhiều Author |
| Book - BookItem | Generalization vì Lớp BookItem là kế thừa của lớp Book (thuộc tính và phương thức) |
| BookItem - Account | Association vì BookItem và Account là 2 lớp độc lập có quan hệ Account mượn/trả BookItem.  1 Account có thể mượn tối đa 2 BookItem/1 lần  1 Account có thể giữ tối đa 3 BookItem |
| Account - Patron | Aggregation vì Account là 1 bộ phận của Patron  1 Patron có 1 Account |
| BookItem – Library | Aggregation vì BookItem là 1 bộ phận của Library  1 Library có nhiều BookItem |
| BookItem – Catalog | Association vì BookItem và Catalog là 2 lớp độc lập có quan hệ Catalog ghi lại hồ sơ về BookItem.  1 Catalog ghi hồ sơ của nhiều BookItem |
| Library - Catalog | Composition vì Catalog là 1 bộ phận của Library và Library điều khiển sự tồn tại của Catalog  1 Library tồn tại 1 Catalog |
| Account - Library | Aggregation vì Account là 1 bộ phận của Library  1 Library có nhiều Account |
| Account - AccountState | Usage dependency vì lớp Account sẽ sử dụng 1 trong các giá trị của enum AccountState để xác định cho thuộc tính state |
| Patron - Search | Usage dependency vì lớp Patron có thể thực hiện 1 trong các method trong interface Search |
| Librarian - Search | Usage dependency vì lớp Librarian có thể thực hiện 1 trong các method trong interface Search |
| Librarian - Manage | Usage dependency vì lớp Librarian có thể thực hiện 1 trong các method trong interface Manage |
| Catalog - Search | Realization vì Catalog implements các phương thức từ interface Search |
| Catalog - Manage | Realization vì Catalog implements các phương thức từ interface Manage |

1. *Draw the diagram with additional attributes in VP*



**Ex6.**

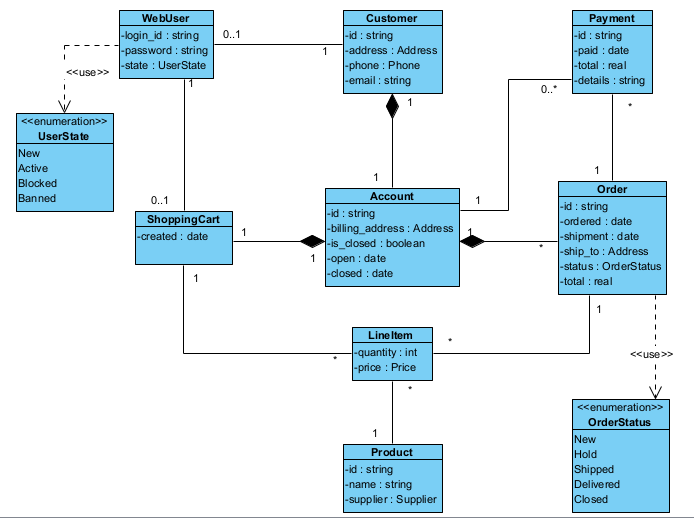
Diagram

Description automatically generated

1. *Add attributes into classes*
2. *Explain relationships (why? And why?)*

|  |  |
| --- | --- |
| Customer – WebUser | Association vì Customer và WebUser là 2 lớp độc lập có quan hệ.  1 Customer có thể có nhiều nhất 1 WebUser |
| WebUser – ShoppingCart | Association vì ShoppingCart và WebUser là 2 lớp độc lập có quan hệ.  1 WebUser có thể tạo nhiều nhất 1 ShoppingCart |
| ShoppingCart – LineItem | Association vì ShoppingCart và LineItem là 2 lớp độc lập có quan hệ.  1 ShoppingCart có nhiều LineItem |
| LineItem – Product | Association vì ShoppingCart và LineItem là 2 lớp độc lập có quan hệ  1 Product có thể thuộc nhiều LineItem |
| LineItem - Order | Association vì Order và LineItem là 2 lớp độc lập có quan hệ  1 Order có nhiều LineItem |
| Order – Payment | Association vì Order và Payment là 2 lớp độc lập có quan hệ vs nhau.  1 Order có nhiều Payment |
| Account - Payment | Association vì Account và Payment là 2 lớp độc lập có quan hệ vs nhau.  1 Account có thể thực hiện 0-nhiều Payment |
| Account – Customer | Composition vì Account là 1 bộ phận của Customer và Customer điều khiển sự tồn tại của Account  1 Customer có 1 Account |
| Account – ShoppingCart | Composition vì ShoppingCart là 1 bộ phận của Account và Account điều khiển sự tồn tại của ShoppingCart  1 Account có 1 ShoppingCart |
| Account – Order | Composition vì Order là 1 bộ phận của Account và Account điều khiển sự tồn tại của Order  1 Account có nhiều Order |
| WebUser – UserState | Usage dependency vì lớp WebUser sẽ sử dụng 1 trong các giá trị của enum UserState để xác định cho thuộc tính state |
| Order – OrderState | Usage dependency vì lớp Order sẽ sử dụng 1 trong các giá trị của enum OrderState để xác định cho thuộc tính state |

1. *Draw the diagram with additional attributes in VP*



**Ex7.**

1. *Compare three models and give your comments*

Diagram

Description automatically generated

Diagram

Description automatically generatedFigure: logical data model

Diagram

Description automatically generated

Figure: physical data model

|  |  |  |
| --- | --- | --- |
| Conceptual Model | Logical Data Model | Physical Data Model |
| Mô tả bức tranh tổng quan hoạt động hệ thống với:   * Các đối tượng tham gia vào mô hình kinh doanh * Mối quan hệ giữa các đối tượng | Bản mô tả chi tiết từ Conceptual Model:   * Bổ sung tên các thuộc tính của các đối tượng * Định nghĩa thêm các thực thể thể hiện cho hoạt động, giao dịch giữa các đối tượng tham gia vào hệ thống | Bản vẽ thiết kế cho cơ sở dữ liệu quan hệ dựa trên Logic Data Model:   * Xác định kiểu dữ liệu, độ dài, null … cho từng cột * Xác định khoá chính, khoá ngoại cho từng bảng * Tuân thủ quy tắc đặt tên cột, tên bảng… * Chuyển quan hệ nhiều-nhiều thành cặp quan hệ 1-nhiều |

1. *Draw them in on a sheet o paper and rewrite your comments*
2. *Draw them in VP*

Diagram

Description automatically generated