

Assignment 08

Data modelling

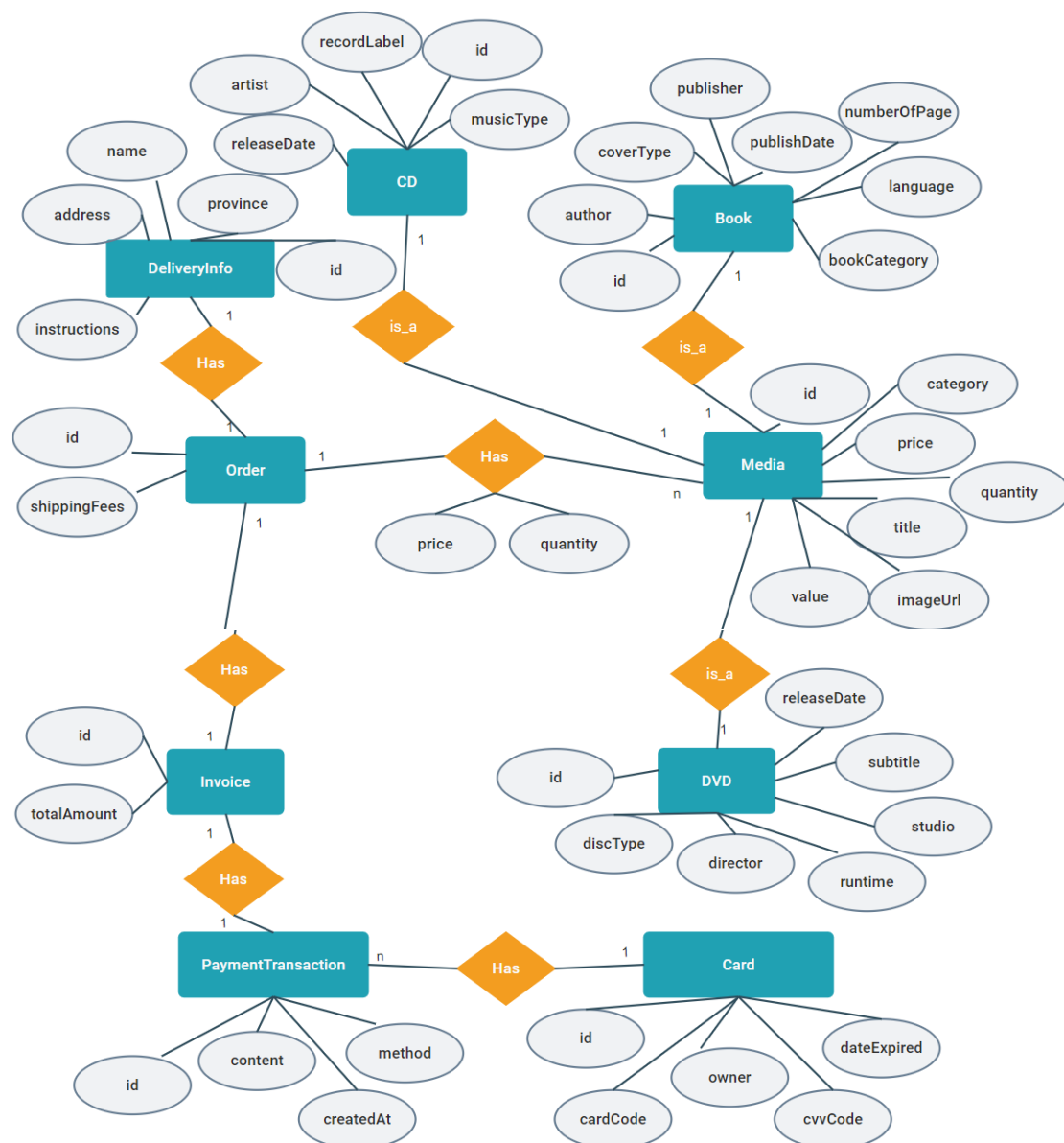
Mục đích và nội dung

5.3.2. Mô hình hóa dữ liệu (Data Modeling)

5.3.2.1. Mô hình khái niệm (Conceptual Data Model)

Mô hình khái niệm là một mô hình dữ liệu có tính trừu tượng cao, trừu tượng hóa các biểu hiện tự nhiên (natural expression) mà không phụ thuộc vào bất kỳ hệ thống quản trị dữ liệu (database management system – DBMS) nào như PostgreSQL, SQLite, Microsoft Access, hoặc MongoDB. Một mô hình khái niệm có thể được thể hiện bởi biểu đồ thực thể-liên kết (Entity-Relationship diagram – ER diagram). Sau đây là một ví dụ cho AIMS Software.

AIMS System ERD



5.3.2.2. Thiết kế cơ sở dữ liệu

Trong phần này, ta cần phải xác định sự lựa chọn hệ thống quản trị dữ liệu (DBMS) và mô tả DBMS.

Ví dụ, ta sẽ chọn SQLite 3.7.2 là DBMS cho Case Study vì:

- SQLite là một DBMS mã nguồn mở.

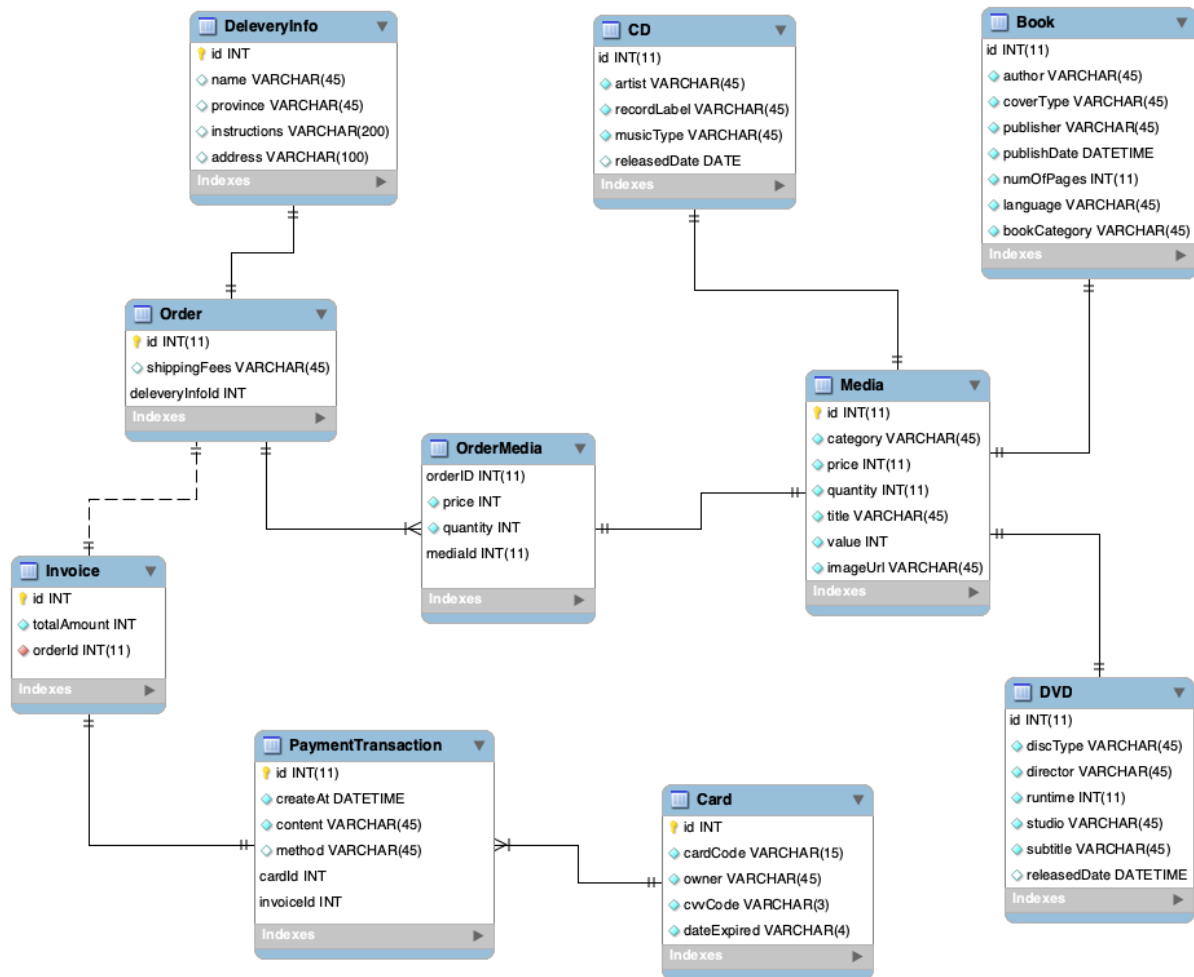
- SQLite không chỉ nhẹ, nhanh, phổ biến, không cần cài đặt, có độ tin cậy cao, đầy đủ tính năng, và là một engine cơ sở dữ liệu SQL; ngoài ra còn ổn định, không phụ thuộc vào nền tảng, tương thích ngược và hỗ trợ lâu dài.
- Có thể kết nối với Java (<https://github.com/xerial/sqlite-jdbc>)

Tuy nhiên, SQLite không thể đảm đương trách nhiệm của một cơ sở dữ liệu ở máy chủ trong thực tế (hệ thống thương mại điện tử) như MySQL hay PostgreSQL vì SQLite lưu dữ liệu của người dùng tại chính thiết bị.

Chúng ta vẫn sử dụng SQLite cho Case Study bởi project mẫu sẽ chỉ chạy local mà vẫn có thể minh họa đầy đủ cho người học.

a) Mô hình dữ liệu logic (Logical data model)

Từ mô hình khái niệm (được thể hiện bởi biểu đồ ER) trong phần trước, ta có thể thiết kế mô hình dữ liệu logic tương thích với DBMS đã chọn (SQLite). Sau đây là mô hình dữ liệu logic ví dụ.



b) Mô hình dữ liệu vật lý (Physical data model)

Trong phần này, chúng ta sẽ thiết kế chi tiết cho từng phần tử trong biểu đồ trên. Ví dụ, trong biểu đồ cơ sở dữ liệu quan hệ (RDBMS), ta thiết kế chi tiết cho từng bảng và ràng buộc.

Chú thích:

PK: Primary Key

FK: Foreign Key

Media

#	PK	FK	Column Name	Data type	Mandatory	Description
1.	x		id	Integer	Yes	ID, auto increment
2.			category	VARCHAR(45)	Yes	Media type, e.g., CD, DVD

#	PK	FK	Column Name	Data type	Mandatory	Description
3.			price	Integer	Yes	Current price
4.			quantity	Integer	Yes	Number of products
5.			title	VARCHAR(45)	Yes	Product name
6.			value	Integer	Yes	Value of the product
7.			imageUrl	VARCHAR(45)	Yes	Product image path

▪ CD

#	PK	FK	Column Name	Data type	Mandatory	Description
1.		x	id	Integer	Yes	ID, same as ID of Media of which type is CD
2.			artist	VARCHAR(45)	Yes	Artist's name
3.			recordLabel	VARCHAR(45)	Yes	Record label
4.			musicType	VARCHAR(45)	Yes	Music genres
5.			releasedDate	DATE	No	Release date

▪ Book

#	PK	FK	Column Name	Data type	Mandatory	Description
1.		x	id	Integer	Yes	ID, same as ID of Media of which type is Book
2.			author	VARCHAR(45)	Yes	Author
3.			coverType	VARCHAR(45)	Yes	Cover type
4.			Publisher	VARCHAR(45)	Yes	Publishing house
5.			publishDate	DATETIME	Yes	Date of publishing
6.			numOfPages	Integer	Yes	Page number
7.			language	VARCHAR(45)	Yes	Language

#	PK	FK	Column Name	Data type	Mandatory	Description
8.			bookCategory	VARCHAR(45)	Yes	Book category

▪ DVD

#	PK	FK	Column Name	Data type	Mandatory	Description
1.		x	id	Integer	Yes	ID, same as ID of Media of which type is DVD
2.			discType	VARCHAR(45)	Yes	Disc type
3.			director	VARCHAR(45)	Yes	Director
4.			runtime	Integer	Yes	Duration
5.			studio	VARCHAR(45)	Yes	Manufacturer
6.			subtitle	VARCHAR(45)	Yes	Subtitles
7.			releasedDate	DATETIME	Yes	Release date
8.			filmType	VARCHAR(45)	Yes	Genres

▪ Card

#	PK	FK	Column Name	Data type	Mandatory	Description
1.	x		id	Integer	Yes	ID, auto increment
2.			cardCode	VARCHAR(45)	Yes	Card code
3.			owner	VARCHAR(45)	Yes	Cardholders
4.			cvvCode	VARCHAR(3)	Yes	CVV code
5.			dateExpired	VARCHAR(4)	Yes	Expiration date

▪ DeliveryInfo

#	PK	FK	Column Name	Data type	Mandatory	Description
1.	X		id	Integer	Yes	ID, auto increment

2.			name	VARCHAR(45)	Yes	Receiver name
3.			province	VARCHAR(45)	Yes	Provinces
4.			instructions	VARCHAR(200)	No	Delivery instructions
5.			address	VARCHAR(100)	Yes	Delivery address

▪ Order

#	PK	FK	Column Name	Data type	Mandatory	Description
1.	X		id	Integer	Yes	ID
2.			shippingFees	VARCHAR(45)	Yes	Shipping fee
3.		X	deliveryInfoId	Integer	Yes	Delivery Info ID

▪ OrderMedia

#	PK	FK	Column Name	Data type	Mandatory	Description
1.		X	mediaID	Integer	Yes	Media ID
2.		X	orderId	Integer	Yes	Order ID
3.			price	Integer	Yes	Selling price
4.			quantity	Integer	Yes	Number

▪ Invoice

#	PK	FK	Column Name	Data type	Mandatory	Description
1.	x		id	Integer	Yes	ID
2.			totalAmount	Integer	Yes	Total
3.		x	orderId	Integer	Yes	Order ID

▪ PaymentTransaction

#	PK	FK	Column Name	Data type	Mandatory	Description
1.	x		id	Integer	Yes	ID
2.			createAt	DATETIME	Yes	Date of creation
3.			content	VARCHAR(45)	Yes	Transaction contents
4.			method	VARCHAR(45)	Yes	Payment methods
5.		x	cardId	Integer	Yes	ID of used card
6.		x	invoiceId	Integer	Yes	Invoice ID

Cuối cùng, ta cần có database script. Với những công cụ thiết kế database chuyên nghiệp và các plugins, ta có thể tự động tạo database script trực tiếp từ mô hình dữ liệu logic.

```

BEGIN;
CREATE TABLE "aims"."Media"(
  "id" INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
  "category" VARCHAR(45) NOT NULL,
  "price" INTEGER NOT NULL,
  "quantity" INTEGER NOT NULL,
  "title" VARCHAR(45) NOT NULL,
  "value" INTEGER NOT NULL,
  "imageUrl" VARCHAR(45) NOT NULL
);
CREATE TABLE "aims"."CD"(
  "id" INTEGER PRIMARY KEY NOT NULL,
  "artist" VARCHAR(45) NOT NULL,
  "recordLabel" VARCHAR(45) NOT NULL,
  "musicType" VARCHAR(45) NOT NULL,
  "releasedDate" DATE,
  CONSTRAINT "fk_CD_Media1"
    FOREIGN KEY("id")
      REFERENCES "Media"("id")
);
CREATE TABLE "aims"."Book"(
  "id" INTEGER PRIMARY KEY NOT NULL,
  "author" VARCHAR(45) NOT NULL,
  "coverType" VARCHAR(45) NOT NULL,
  "publisher" VARCHAR(45) NOT NULL,
  "publishDate" DATETIME NOT NULL,
  "numOfPages" INTEGER NOT NULL,
  "language" VARCHAR(45) NOT NULL,
  "bookCategory" VARCHAR(45) NOT NULL,

```



```
        CONSTRAINT "fk_Book_Media1"
            FOREIGN KEY("id")
            REFERENCES "Media"("id")
    );
CREATE TABLE "aims"."DeleveryInfo"(
    "id" INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
    "name" VARCHAR(45),
    "province" VARCHAR(45),
    "instructions" VARCHAR(200),
    "address" VARCHAR(100)
);
CREATE TABLE "aims"."Card"(
    "id" INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
    "cardCode" VARCHAR(15) NOT NULL,
    "owner" VARCHAR(45) NOT NULL,
    "cvvCode" VARCHAR(3) NOT NULL,
    "dateExpired" VARCHAR(4) NOT NULL
);
CREATE TABLE "aims"."DVD"(
    "id" INTEGER PRIMARY KEY NOT NULL,
    "discType" VARCHAR(45) NOT NULL,
    "director" VARCHAR(45) NOT NULL,
    "runtime" INTEGER NOT NULL,
    "studio" VARCHAR(45) NOT NULL,
    "subtitle" VARCHAR(45) NOT NULL,
    "releasedDate" DATETIME,
    CONSTRAINT "fk_DVD_Media1"
        FOREIGN KEY("id")
        REFERENCES "Media"("id")
);
CREATE TABLE "aims"."Order"(
    "id" INTEGER NOT NULL,
    "shippingFees" VARCHAR(45),
    "deleveryInfoId" INTEGER NOT NULL,
    PRIMARY KEY("id","deleveryInfoId"),
    CONSTRAINT "fk_Order_DeleveryInfo1"
        FOREIGN KEY("deleveryInfoId")
        REFERENCES "DeleveryInfo"("id")
);
CREATE INDEX "aims"."Order.fk_Order_DeleveryInfo1_idx" ON "Order"
    ("deleveryInfoId");
CREATE TABLE "aims"."OrderMedia"(
    "orderId" INTEGER NOT NULL,
    "price" INTEGER NOT NULL,
    "quantity" INTEGER NOT NULL,
    "mediaId" INTEGER NOT NULL,
    PRIMARY KEY("orderId","mediaId"),
    CONSTRAINT "fk_ordermedia_order"
        FOREIGN KEY("orderId")
        REFERENCES "Order"("id"),
    CONSTRAINT "fk_OrderMedia_Media1"
        FOREIGN KEY("mediaId")
        REFERENCES "Media"("id")
);
```

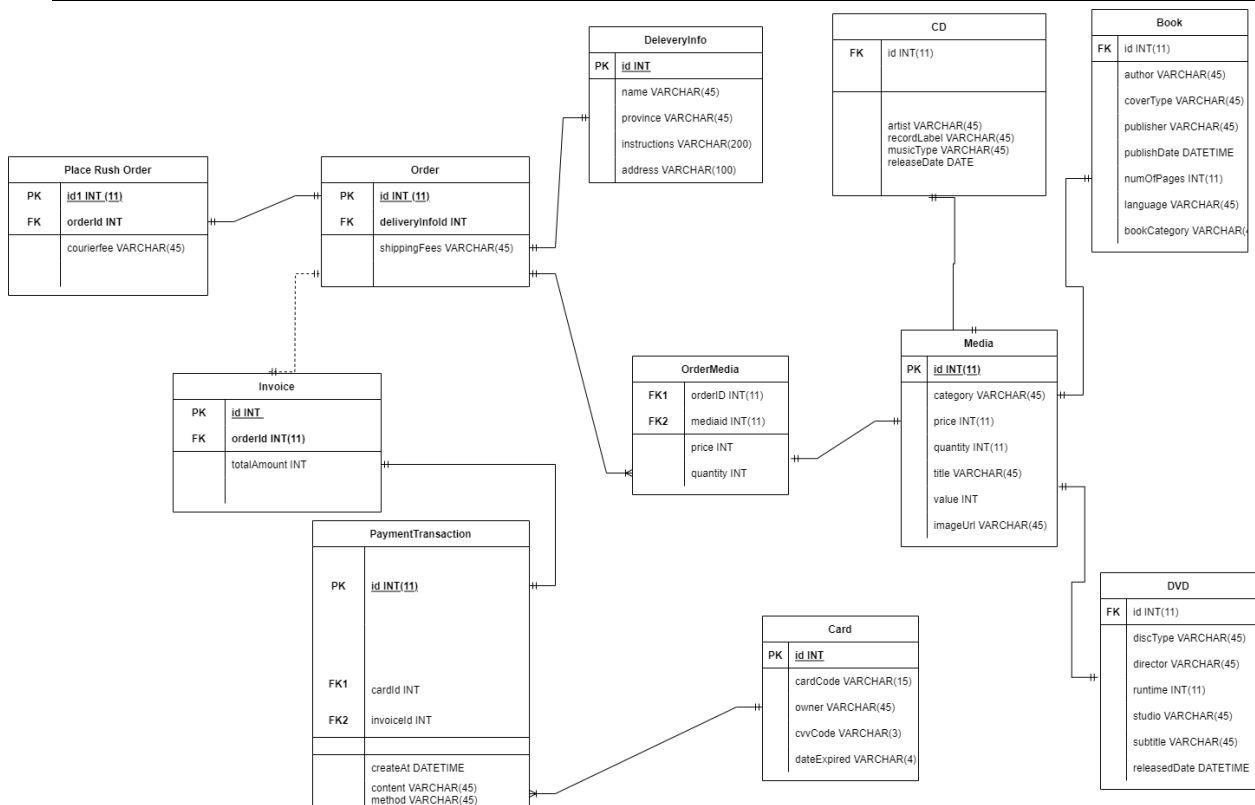
```

CREATE INDEX "aims"."OrderMedia.fk_ordermedia_order_idx" ON "OrderMedia"
("orderId");
CREATE INDEX "aims"."OrderMedia.fk_OrderMedia_Media1_idx" ON "OrderMedia"
("mediaId");
CREATE TABLE "aims"."Invoice"(
    "id" INTEGER PRIMARY KEY NOT NULL,
    "totalAmount" INTEGER NOT NULL,
    "orderId" INTEGER NOT NULL,
    CONSTRAINT "fk_Invoice_Order1"
        FOREIGN KEY("orderId")
        REFERENCES "Order"("id")
);
CREATE INDEX "aims"."Invoice.fk_Invoice_Order1_idx" ON "Invoice"
("orderId");
CREATE TABLE "aims"."PaymentTransaction"(
    "id" INTEGER NOT NULL,
    "createAt" DATETIME NOT NULL,
    "content" VARCHAR(45) NOT NULL,
    "method" VARCHAR(45),
    "cardId" INTEGER NOT NULL,
    "invoiceId" INTEGER NOT NULL,
    PRIMARY KEY("id","cardId","invoiceId"),
    CONSTRAINT "fk_PaymentTransaction_Card1"
        FOREIGN KEY("cardId")
        REFERENCES "Card"("id"),
    CONSTRAINT "fk_PaymentTransaction_Invoice1"
        FOREIGN KEY("invoiceId")
        REFERENCES "Invoice"("id")
);
CREATE INDEX "aims"."PaymentTransaction.fk_PaymentTransaction_Card1_idx"
ON "PaymentTransaction" ("cardId");
CREATE INDEX
    "aims"."PaymentTransaction.fk_PaymentTransaction_Invoice1_idx" ON
    "PaymentTransaction" ("invoiceId");
COMMIT;

```

5.4. BÀI TẬP

Hãy thiết kế lớp chi tiết và mô hình hóa dữ liệu cho Use case “Place Rush Order”.



+ Place Rush Order

#	PK	FK	Column Name	Data type	Mandatory	Description
1.	X		Id1	Integer	Yes	ID
2.			courierFee	VARCHAR(45)	Yes	Courier fee
3.		X	orderId	Integer	Yes	Order ID

```

CREATE TABLE "aims"."Place Rush Order"(
    "id1" INTEGER NOT NULL,
    "courierFee" VARCHAR(45),
    "orderId" INTEGER NOT NULL,
    PRIMARY KEY("id1"),
    CONSTRAINT "fk_PlaceRushOrder_Order"
        FOREIGN KEY("orderId")
        REFERENCES "Order"("id")
);
  
```

Bài tập nhóm:

Thực hiện Thiết kế cơ sở dữ liệu (Database Design) cho bài tập lớn môn học

- Các bước thực hiện:
 - 1. Xây dựng mô hình dữ liệu mức khái niệm (sơ đồ thực thể-liên kết ERD).
 - 2. Xây dựng mô hình dữ liệu logic.
 - 3. Đặc tả mô hình dữ liệu vật lý.
- Yêu cầu nộp bài:
 - Nộp bài vào thư mục 03-DetailedDesign trên thư mục Google Drive mà thầy đã tạo
(https://drive.google.com/drive/folders/18or3uP5H3xr69Pkuidj_5eLQo__X1hew?usp=sharing)
 - Tổng hợp các kết quả thiết kế bước đầu ở trên viết tiếp vào tài liệu SRS
 - Các tệp tin Astah

HẾT