

CSE 3241 Project Checkpoint 01 – Entities and Relationships

Names: Jeanie Chen, Yuanzhe Li, Zhe Luan, Haoyan Xiang

Date: 5/21/2017

In a **NEATLY TYPED** document, provide the following:

1. Based on the requirements given in the project overview, list the entities to be modeled in this database. For each entity, provide a list of associated attributes.

i.e. entities: associated attributes (attribute1, attribute2, ...)

1. book: published date, ISBN, edition, category(**name**), price, title, sales_amount, in_stock_amount.

2. author: name(first_name, last_name), phone_number

3. publisher: name, phone_number

4. customer: name, email, phone number, **credit_card_num**,

5. address(street, zipcode)

2. Based on the requirements given in the project overview, what are the various relationships between entities? (For example, "CUSTOMER entities purchase BOOK entities").

1. BOOK entities are written by AUTHOR entities.

2. PUBLISHER entities publish BOOK entities.

3. CUSTOMER entities purchase BOOK entities.

4. BOOK entities recommend BOOK entities.

5. PUBLISHER entities have AUTHOR entities.

3. Propose at least two additional entities that it would be useful for this database to model beyond the scope of the project requirements. Provide a list of possible attributes for the additional entities and possible relationships they may have with each other and the rest of the entities in the database. Give a brief, one sentence rationale for why adding these entities would be interesting/useful to the stakeholders for this database project.

i.e. entities: associated attributes (attribute1, attribute2, ...)

1. transaction_history: transaction_id, status(is_Sold/ is_Rented(dates(rent_date, return_data), returned))

2. course: course_number, post_course_num

CUSTOMER entities generate TRANSACTION_HISTORY entities.

COURSE entities post COURSE entities.

COURSE entities use BOOK entities.

TRANSACTION_HISTORY entities are related to COURSE entities.

TRANSACTION_HISTORY entity is related to BOOK_HAS_TRANSACTION entity.

BOOK_HAS_TRANSACTION entity is related to BOOK entity.

BOOK entity is related to COURSE_USE_BOOK entity.

COURSE_USE_BOOK entity is related to COURSE entity.

Reason for adding these entities:

The TRANSACTION_HISTORY entity records the book transaction history of each customer. Based on the transaction history of a customer, the system could recommend other books to the customer which he/she might be interested in as well. Also, it is useful for the administrator and the customer to review the customer account status.

This database system will be mainly focusing on servicing college students. Customers(possibly college students) would get recommendations of similar textbooks they might need whenever they purchased a textbook from the bookstore. The recommendations are given based on the information of the textbooks that will be used in a course. And that's why we want to add the COURSE entity which includes the information of its course number and courses which have this one as the prerequisite.

4. Give at least four examples of some informal queries/reports that it might be useful for this database might be used to generate. Include one example for each of the additional entities you proposed in question 3 above.
 1. A customer can find his or her favorite author's books that are available in the bookstore.
 2. A customer can check the price of a book.
 3. A transaction history can provide the administrator the in-stock of a book.
 4. A course can show customer other courses he or she may take and give recommendation of books he or she may purchase, the textbook of that course, for example.
5. Suppose we want to add a new publisher to the database. How would we do that given the entities and relationships you've outlined above? Given your above description, is it possible to add a new publisher to your database without knowing the title of any books they have published? If not, revise your model to allow for publishers to be added as separate entities.

We can add a new instance of the PUBLISHER entity with the name and phone number of this new publisher. Yes. We can add a new publisher to the DB without knowing the title of books they published.

6. Determine at least three other informal update operations and describe what entities would need to have attributes altered and how they would need to be changed given your above descriptions. Include one example for each of the additional entities you proposed in question 3 above.
 1. Customer rented a book : (1) book -> in_stock_amount decrement; (2) generates a new transaction_history to this customer: give a transaction_id to this record of transaction_history, mark type as rent, mark status as is_Rented as well as its rented and return dates, record the book's ISBN.
 2. Customer bought a book: (1) book -> sales_amount increment; (2) book -> in_stock_amount decrement; (3) generates a new transaction_history to this customer: give a transaction_id to this record of transaction_history, mark type as buy, mark status as is_Sold, records the book's ISBN.
 3. Book sold out: (1) set book -> in_stock_amount to zero
 4. Add a newly published book to stock: (1) add a new instance of the BOOK entity and set its attributes according to its information. (2) update the in_stock_amount based on how many this new book get into the stock.

7. Provide an ER diagram for your database. Make sure you include all of the entities and relationships you determined in the questions above ***INCLUDING the entities for question 3 above***, and remember that ***EVERY*** entity in your model needs to connect to another entity in the model via some kind of relationship.