

Project Step 3 Draft (HTML Interface, DML, & Revisions to Existing Deliverables)

Ya Zou & Maxwell Zimmer

a) Fixes based on Feedback from Step 2 Draft:

1. Review from Evan Hock:

- Description of issues:

There is only one issue that Evan has dressed as an issue he couldn't find a correspondence between the ER diagram's Categories.numCategories attribute and anything else in the document.

- Actions to solve the issues:

We deleted the numCategories in the ERD because it was a typo when we make the ERD.

2. Review from Charlie Chen:

- Description of issues:

Charlie pointed out that our DDL.sql doesn't have a CASCADE operation defined for any foreign key constraints, which might help by doing the following ALTER TABLE **Orders** ADD CONSTRAINT **Orders_ibfk_1** FOREIGN KEY (**customerID**) REFERENCES **Customers** (**customerID**) ON DELETE CASCADE and sample data shown in the PDF is not full inserted into the database.

- Actions to solve the issues:

We added the on delete cascade and fully implemented our sample data in the table

3. Review from Samantha Egge

- Description of issues:

She didn't see any CASCADE operations present and for the sample data there customerOrders doesn't have any example data, and the items table has all of the reviewID attributes set to NULL.

- Actions to solve the issues:

We added on delete cascade to our FK and implemented the example data in our database and also fixed reviewID to not null.

Review from Robert Long:

- Description of issues: he also pointed out that our DDL.sql file doesn't have any cascade operations
- Actions to solve the issues: already fixed it.

Grading Feedback from Instructor:

- Description of issues:

I don't believe I was asked about your outline as page 5 indicates (but please correct me if I'm wrong). I *do* think the outline should be for a specific use-case (either a particular shop or kind of shop). This will help guide your design somewhat. Please update this for future revisions. The current review lacks motivation for including "Reviews" and "Categories". How many reviews can a user write, for instance. And can items only belong to one Category?

ERD and Outline aren't consistent. For example, Customers does not have postalCode (in ERD) and Items has "stockAmount" instead of "numItems". Items FK to Categories should point to CategoryID, but it seems to be a varchar(50) and not an INT. The current design doesn't imply that a single Order row can be associated with multiple items. We would need an intersection table (composite entity) between Orders and Items to achieve this.

Also, consider what your primary keys should be for each relation. Making a distinct primary key (e.g., reviewID) has a very different effect on row participation than using the FKs as the foreign keys.

Remember to put Summary of Changes (reviews and how you acted on them) at the top

- Actions to solve the issues:

We decided to design a database for an Online electronic retailer called OregonTech and rewrote the project outline, we also changed the relationship between table Customers and Reviews, now is a customer can make 0 and many reviews, a review can be written one and only one customer. Yes, we have three categories (Phones, iPads, Laptops) in our database and each item has one and only one category to which it

belongs. We deleted postalCode from our outline and ERD. Also changed the stockAmount to numItems in the table Items. A FK(categoryID) is added to the Items table and we confirmed all ID in our ERD are int, we also added a composite table between Items and Categories. We also changed the order of the PDF.

b) Project Outline and Database Outline - Updated Version:

Team member names and the project title.

Team members: Ya Zou, Maxwell Zimmer.

Project title: Online Electronic Retailer.

Project Outline:

OregonTech is a fast-growing electronic online shop specializing in selling iPads, phones, and laptops. With an impressive annual sales revenue of \$15 million, the client requires a robust and efficient database-driven website with a relational database backend to handle their high volume of sales and product inventory. The system's main focus will be on recording Sales Orders for iPads, phones, and laptops to Customers, ensuring real-time tracking of inventory levels, and providing accurate stock information to both customers and the management team. The website must support a seamless and user-friendly shopping experience, allowing customers to easily browse through a wide range of electronic products, make purchases, and track their orders. As a result, the database system must be designed to handle a substantial number of daily transactions, ensure data integrity, and offer exceptional performance to cater to the demands of Tech Emporium's thriving online retail business.

Database Outline in Words:

1. **Customers:** Record the details of customers who do business with the online electronic shop.

Attributes:

- customerID: int, auto_increment, unique, NOT NULL, PK
- firstName: varchar(50)
- lastName: varchar(50)

- emailAddress: varchar(50)
- address: varchar(50)
- city: varchar(50)
- Relationship: A one-to-many (1:M) relationship between Customers and Orders, where the customerID serves as a foreign key (FK) in the Orders table, linking each order to the corresponding customer.

2. **Items:** Represents the electronic products available for sale on the online shop.

Attributes:

- itemID: PK, auto_increment. Int, unique
- reviewID FK2, int, NOT NULL
- categoryID FK3,int
- itemName varchar(50)
- price int (12)
- Relationship: M: N between orders and items, 1:N between reviews and items, M:1 between items and categories.

3. **Orders:** tracks details of customer orders placed on the online shop

Attributes:

- orderID: PK, int, auto-incement, NOT NULL
- customerID: FK1 ->customer table, not NULL
- orderDate: date
- creditCardNumb int(12)
- creditCardExpDate date
- numOrderedItmes int(12)
- pricePaid int
- Relationship: M: N between orders and items, 1: N between orders and customers

4. **Categories:** Divide electronic products into different categories for sale on the online shop.

Attributes:

- categoryID int, auto_increment, unique, not NULL, PK
- categoryName varchar(50)

- Relationship: M: 1 between categories and items.

5. Reviews: Records customer reviews and ratings for products

Attributes:

- reviewID: PK, int, auto-increment, not NULL.
- customerID: int not NULL, FK.
- overallRating: decimal(3,2), not NULL
- feedback: varchar(200).
- Relationship: 1:N between customers and reviews, M:1 between reviews and items.

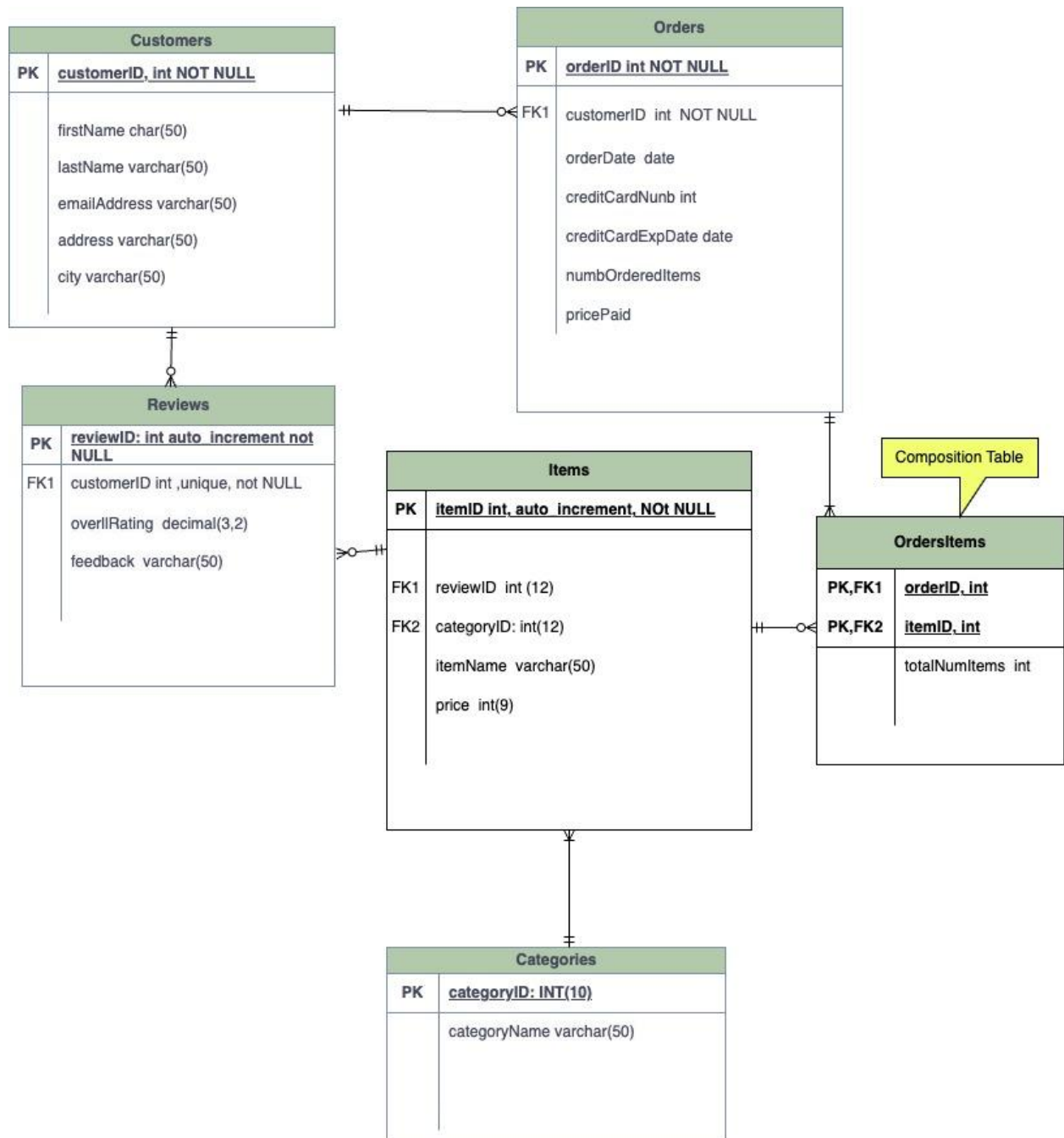
6. ordersItems: composition table, It keeps tracking our inventory

Attributes:

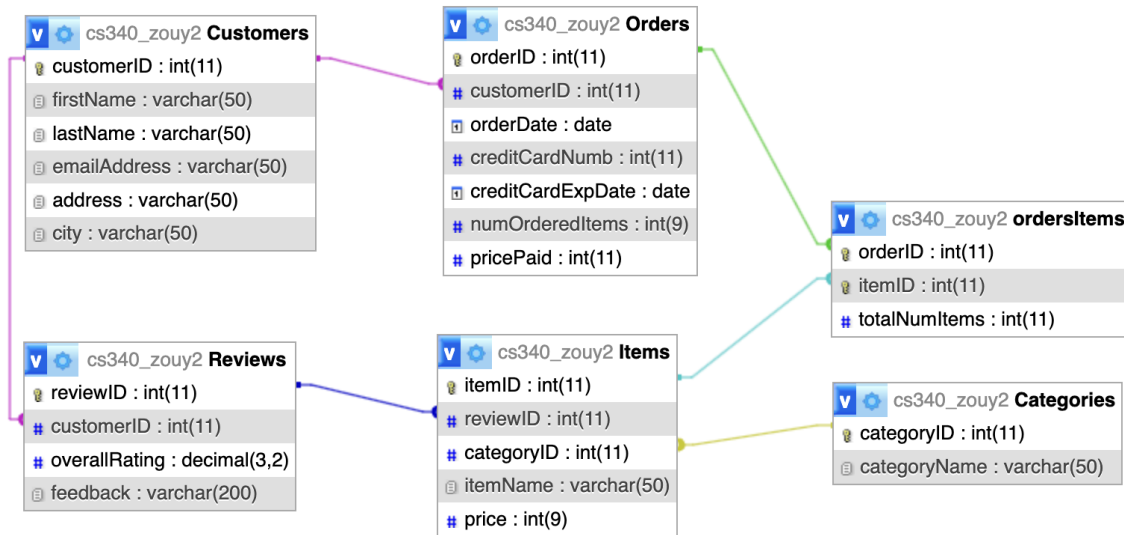
- orderID: int not NULL, FK,PK
- itemID: int NOT NULL ,FK,PK
- totalNumItems: int

Our design is normalized.

c) Entity-Relationship Diagram- Updated Version:



d) Schema



e) Example Data

| Customers | | | | | |
|-----------------------|-----------|----------|------------------------|----------------|-----------|
| customerID(PK) | firstName | lastName | emailAddress | adress | city |
| auto_increment | Jack | Spring | JackSpring23@hello.com | 123NW lane dr | Corvillas |
| auto_increment | Alice | Summer | AliceSummer@hello.com | sw ln dr 1256 | Salem |
| auto_increment | Iris | Winter | IrisWinter@hello.com | 420 Database C | Portland |
| auto_increment = 1000 | | | | | |

| Items | | | | |
|--------------------|------------------|---------------|----------|-------|
| ItemID (PK) | categoryID (FK1) | reviewID(FK2) | itemName | price |
| auto_increment | used select | used select | phone | 729 |
| auto_increment | used select | used select | ipad | 629 |
| auto_increment | used select | used select | laptop | 999 |
| auto_incremnt = 30 | | | | |

| Categories | |
|----------------------|--------------|
| categoryID(PK) | categoryName |
| auto_increment | lphones |
| auto_increment | lpads |
| auto_increment | Laptops |
| auto_increment = 500 | |

| Reviews | | |
|----------------------|---------------|-----------|
| reviewID(PK) | overallRating | feedback |
| auto_increment | 4.8 | excellent |
| auto_increment | 1.3 | bad |
| auto_increment = 100 | 3.3 | good |

| Orders | | | | | | |
|------------------------|--------------------------------|------------|----------------|----------------|--------------|-----------|
| OrderID (PK) | customerID(FK1) | orderDate | creditCardNumb | creditCardExpD | numOrderedIt | pricePaid |
| auto_increment | used select to find customerID | 2-2-2019 | 1234567 | 2-26-2021 | 2 | 1628 |
| auto_increment | used select to find customerID | 10-12-2018 | 1234568 | 4-13-2023 | 1 | 629 |
| auto_increment | used select to find customerID | 2-15-2017 | 1234569 | 9-7-2021 | 1 | 729 |
| auto_increment = 10000 | | | | | | |

| Composition_table | | |
|-------------------|---------------|------------|
| ordersItems | | |
| orderID(PK,FK) | itemID(FK,PK) | orderItems |
| used select | used slect | 43 |
| used select | used slect | 56 |
| used select | used slect | 96 |