Project Step 2 Draft (Project Group 12)
Jonathan Pham, Leon Ong

Note: Text changes for Step 2 are highlighted

a) Fixes based on Feedback from Step 1

Feedback by Peer Reviewers:

From Chase Narin-Howard:

Does the overview describe what problem is to be solved by a website with a relational database back end?
Yes.
Does the overview list specific facts?
Yes. The team does a great job providing specific details and creating a background for why their database is needed.

Are at least four entities described and does each one represent a single kind of object/idea to be stored?
4 entities are listed, but they do not appear to all be unique. Specifically, Machine and freeWeights seem like they could be combined into Equipment by changing the structure of Equipment to something like the following:

- equipmentID*: int, auto_increment, unique
- equipmentName: varchar
- equipmentType: varchar, FK to ProductTypes(equipmentType)

Where the entity ProductType would be a table with two fields, typeID and equipmentType. Here, we could have two records {(1, "Machine"), (2, "Free Weights")} that could be used to represent equipment that either falls under the "Machine" category or the "Free Weights" category.

This would also allow other types of equipment (such as gym mats and misc items) to more easily be added into the database

Does the outline of entity details describe the purpose of each entity as well as its…

- **Attributes and their respective datatypes and constraints?**
  - Yes. However, for some fields there are no constraints when it looks like there should be some. For example, for the Users entity, all of the following fields can be NULL: firstName, lastName, address, eMail, phoneNumber. This would allow records with no personally identifying information to be entered, which seems problematic.
- **The relationships between entities?**
  - Yes.

Are 1:M relationships correctly formulated? Is there at least one M:M relationship? Does the ERD present a sufficiently detailed view of the database?

There is an M:M relationship. The relationship between equipment and users seems a bit odd though, and may be better suited as a M:0 relationship instead of a 1:1 relationship; in the current format, it appears Users may own many pieces of equipment, which is normal, but each piece of equipment belongs to one and only one user. Thus, a user couldn't own multiple treadmills or multiple weights. Additionally, this implies that each piece of equipment must be owned by a user, so there is no record in Equipment which is not already owned by a user.

Are the names of entities and attributes--including plural/singular forms and capitalization--consistent across the document?
Yes. camelCase is used consistently for fields, and Uppercase is used consistently for entities (except for freeWeights).

From Logan Helstad:

The project overview looks like it's coming along nicely so far. There are a few things I'd look out for:

- I don't see exactly where your primary keys match up with your foreign keys. I would look to clarify exactly what your foreign keys are for your final overview.
- I'm not quite sure that the connection between all of users and orders, orders and equipment, equipment and users are totally necessary. Is it a) users create orders, which have equipment, or b) users have both equipment and orders? Either way, I think one of the connections here could be eliminated.
- In the equipment box, I think there's a not-so-necessary row 3 column, unless I'm missing something.

The overview section is quite strong. I don't have any tips for you on that end. Good work!

From Theodore Ja:

Does the overview describe what problem is to be solved by a website with a relational database back end?
Yes, the overview describes the problem to be solved by a website with a relational database back end, which is to keep track of sales, inventory, customer information, orders, and equipment details for a gym equipment company.

Does the overview list specific facts?
Yes, the overview lists specific facts, such as the company making around $750,000 in sales every year and offering a selection of equipment that includes weight machines, barbells, and dumbbells.

Are at least four entities described and does each one represent a single kind of object/idea to be stored?
Yes, there are five entities described: Users, Order, Equipment, Machine, and freeWeights. Each entity represents a single kind of object/idea to be stored. There is

also an argument to be made that freeWeights, Machine, and Equipment could all be combined into a single table.

Does the outline of entity details describe the purpose of each entity as well as its attributes and their respective datatypes and constraints?
The outline of entity details describes the purpose of each entity, along with their attributes and respective datatypes. Most of the constraints are listed like unique, auto_incrementing, and NOT NULL, there are some varchar attributes that don't list if they are NOT NULL which probably should have the tag.

The relationships between entities?
The relationships between entities are mentioned, but some of the relationships could be better defined by specifically listing what foreign keys are used and where.

Are 1:M relationships correctly formulated? Is there at least one M:M relationship? Does the ERD present a sufficiently detailed view of the database?
The 1:M relationships are correctly formulated, but there is no M:M relationship. The ERD is displays a detailed view of the database though the mention of foreign keys is missing. Additionally, many of the datatypes and constraints are missing from the tables. There also is an extra row on the equipment table.

Are the names of entities and attributes--including plural/singular forms and capitalization--consistent across the document?
The names of entities and attributes are mostly consistent in terms of capitalization and plural/singular forms. In Machine table there is one inconsistent attribute: workout_bodyType which uses both camel case and snake case where all other attributes are in camel case.

Overall, the database outline provides a good starting point for the Strength Solutions Supply project. It would be beneficial to add constraints to the attributes, add or redefine the some of the relationships, and slightly polish up the ER diagram to include the data types and constraints.

From Step 1 Final TA Feedback:

The numerical facts could have more details about: inventory (how many and what kinds), number of the equipment (approximately), ratings (min/max).

The website (what functionality will potential users have, how it can be accessed and used) and "scope" system should have more information.

Actions based on the feedback:

From the first peer review from Chase, we realized that the Machine and freeWeights entities were unnecessary and could be condensed inside of the Equipment table. We took his suggestion to create a new entity table called ProductType, which would act as a category table for different types of equipment. This allows us to have more than just

free weights or machines to sell. One of the other suggestions was to start adding not_NULL indicators to attributes that would cause problems if it was NULL (address, firstName, etc.). In addition, we got rid of the direct relationship between equipment and users and have it split between Orders and Reviews (will be discussed later).

For the second peer review from Logan, we received feedback that we did not have our primary keys match with our foreign keys. So in our final draft outline, we clearly specified what our foreign keys were. One piece of feedback we heard from Chase as well was to get rid of the connection from Users to Equipment, which made it confusing establishing entity relationships. We also cleaned up the ERD to not have random rows with no data.

The last peer review from Theodore has a few suggestions which were covered in the past reviews. These suggestions included: condensing the freeWeights/Machine table inside of Equipment and adding more constraints and assigning not_NULL to more needed attributes. One suggestion that we also took into account was making sure we had a M:M relationship, which is now seen with Orders and Equipment.

After receiving the Step 1 Final feedback from the TA, we took the advice on adding more numerical information in the written overview in order to get a sense for the scale of the database. Some of the details that we added included: the amount of equipment types in the store inventory, the quantity of equipment, and the range of numbers for ratings. Another piece of feedback from the TA from our Step 1 final is to give more context on how the database will be used by the client. For example, what kind of functionality the user can have and how detailed the system would be for the user.

Upgrades to the Draft version:

Some of the changes that we made based on our own design decisions was to add another entity table called Reviews to further add depth in our design. The Reviews entity will have a 1:M relationship from the user as a user can have many reviews. However, reviews can only be from one person. Reviews also has a 1:M relationship from Equipment because an equipment can have many reviews, but a review is directed towards a certain equipment.

Another addition to our draft is including attributes inside of the composite entity between Orders and Equipment. For the OrderEquipment table, *amount* is how much of the same item is purchased where *cost* is just the total price of the certain items bought (dependent on amount).

We also added another table called Product Type and it describes the types of equipment there are. It is to separate the types of equipment that may be in different categories. It has a 1:M relationship with Equipments.

For the step 2 draft, we made minor changes in our attributes and their properties to better fit our outline and to prevent any data integrity issues, such as adding notNULL to some attributes.

b) Project Outline and Database Outline - Updated Version

Project Step 1 Final Project Overview

Team Members: Project Team 12
Leon Ong
Jonathan Pham
Project Title:
Strength Solutions Supply

Overview:
In the past few years, gyms and fitness centers have been negatively affected by the pandemic due to their closures, which has caused a pause in the fitness industry. The result is an increase in home gyms so that people can still get a workout from the comfort of their homes. Customers that invest in a gym also invest in their well-being and health.

Our client is Strength Solutions Supply, a gym equipment company that caters to home and commercial gyms as they begin to open. The company makes around $750,000 in sales every year. To keep track of sales, a database is used to track equipment orders (Orders). It can also track inventory, customer information, orders, and equipment details (Equipment) to ensure organization in a large-scale business as the website gets around 300+ visitors a day.

The company has a range of 5 equipment types from weight machines, free weights, cardio, home-gym systems, and accessories. With these equipment types, the gym supply warehouse has about 200 items in their inventory to be shipped in a moment's notice. These categories of fitness equipment will be categorized (ProductType). In addition, users can write reviews and give ratings on the equipment in a scale of 1-5 stars on the website and those reviews will be stored in the database as well (Reviews). The users of this website can browse around the website and purchase equipment after filling out their information, in which the database can process the order and ship the order from its inventory.

Database Outline, in Words:
Primary keys indicated by *
Entities are **bolded**

Users: Includes details of users that shop at Strength Solutions Supply

- userID*: int, autoIncrement, notNULL, unique
- firstName: varchar(50), notNULL
- lastName: varchar(50), notNULL
- address: varchar(50), notNULL
- phoneNumber: varchar(50) notNULL
- eMail: varchar(50), notNULL
- Relationship: 1:M relationship between User and Orders, using userID as an FK for Orders
- Relationship: 1:M relationship between User and Review

Orders: Includes details of order made by user

- orderID*: int, notNULL, autoIncrement, unique
- userID: varchar(50), notNULL, FK (from Users)
- orderDate: date, notNULL
- numItems: int, notNULL, autoIncrement
- totalCost: decimal(19,2), notNULL
- Relationship: 1:N relationship between Users and Order, using userID as an FK
- Relationship: M:N relationship between Order and Equipment, using OrderEquipment as a composite entity
- Relationship: 1:M relationship between Order and OrderEquipment

OrderEquipment: Has details of a single equipment for purchase

- orderID: int, notNULL, autoIncrement, unique, FK (from Order table)
- equipmentID: int, autoIncrement, unique, FK (from Equipment table)
- amount: int, notNULL
- cost: decimal(19,2), notNULL
- Relationship: 1:M from Order to OrderEquipment
- Relationship: 1:M from Equipment to OrderEquipment

Equipment: Includes details of equipment of the company warehouse

- equipmentID*: int, autoIncrement, unique, notNULL
- equipmentName: varchar(50), notNULL
- equipmentDescription: varchar(500), notNULL
- equipmentCost: int, notNULL

- equipmentStock: int, notNULL
- productTypeID: int, notNULL, FK (from ProductType table)
- Relationship: 1:M from Equipment to OrderEquipment
- Relationship: 1:M relationship between Equipment and Reviews
- Relationship: 1:M relationship between Equipment and EquipmentReview

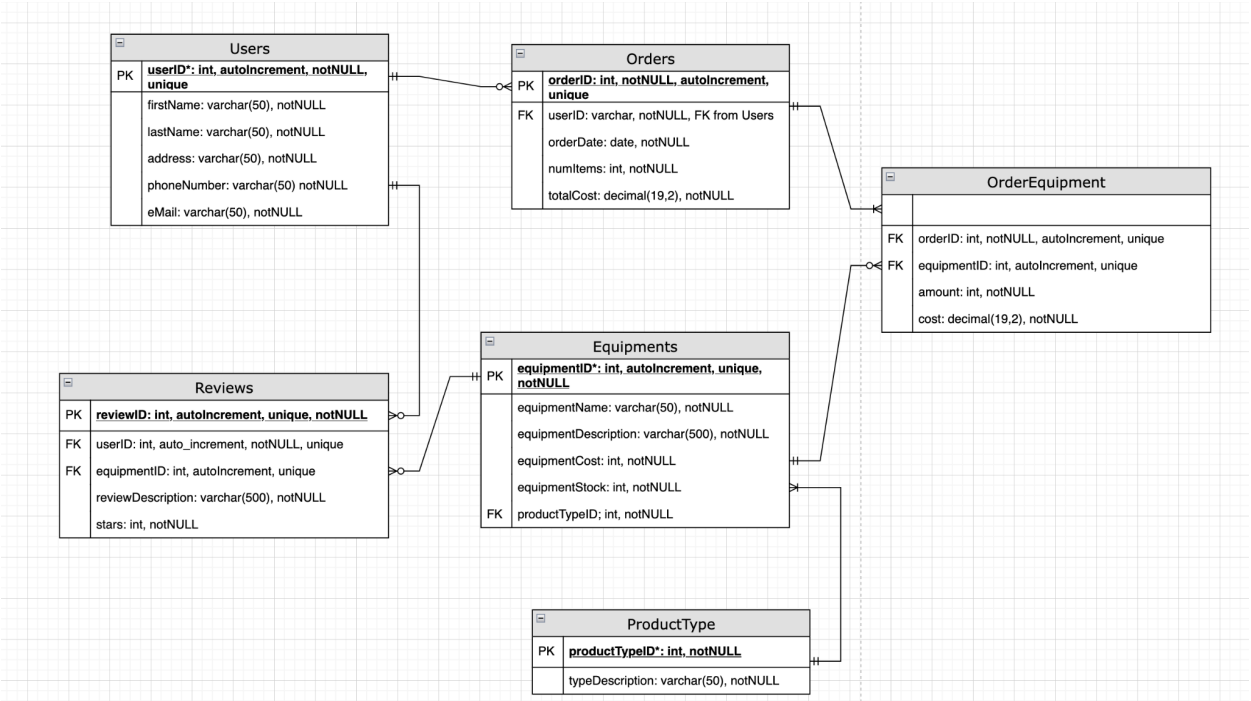
ProductType: Is a category table to distinguish between types of equipment

- productTypeID*: int, notNULL
- typeDescription: varchar(50), notNULL
- Relationship: 1:M relationship between ProductType and Equipment

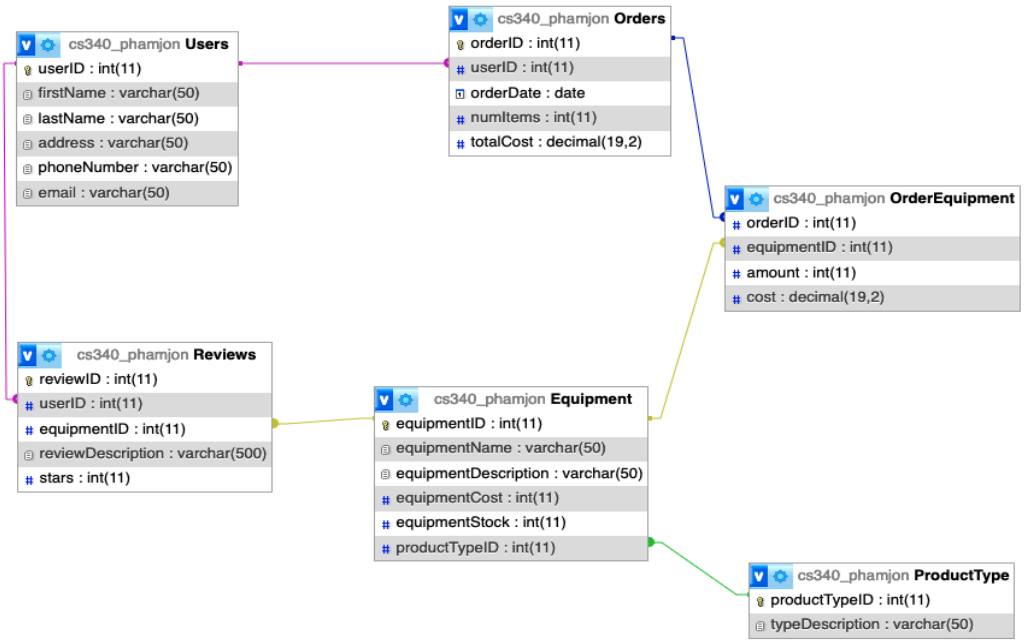
Reviews: Includes comments users make on each equipment.

- reviewID*: int, autoIncrement, unique, notNULL
- userID: int, auto_increment, notNULL, unique, FK (from Users)
- equipmentID: int, autoIncrement, unique, FK (from Equipment)
- reviewDescription: varchar(500), notNULL
- stars: int, notNULL
- Relationship: 1:M between Equipment and Reviews
- Relationship: 1:M between Users and Reviews

c) Entity-Relationship Diagram - Updated Version



d) Schema



e) Example Data

Users:

userID	firstName	lastName	address	phoneNumber	email
1	Jon	Doe	12345 Freeway St.	123-423-2345	jonDo@bye.com
2	Steve	Parks	42235 Some Ave.	753-833-2355	steveParks@outlook.com
3	Jeff	Bezzels	235 Poly Circle	962-434-57243	jeffBez@amazing.com
4	Alvin	Loops	3842 Byte Blvd.	864-246-5723	alvLoops@code.com

4 rows in set (0.000 sec)

Equipment

equipmentID	equipmentName	equipmentDescription	equipmentCost	equipmentStock	productTypeID
1	Squat Rack	A metal rack or cage consisting of support pillars with adjustable bars and hooks	450	21	2
2	Treadmill	for indoor use that simulates walking, jogging, or running	330	11	3
3	Yoga Mat	For your yoga classes!	20	30	5

3 rows in set (0.000 sec)

Reviews:

reviewID	userID	equipmentID	reviewDescription	stars
1	3	1	This equipment is really amazing and I really love it. I would recommend to others.	4
2	1	2	This equipment is alright. It is just good for running.	2
3	4	3	This equipment is bad. Only good for stretching but you can just stretch on the floor.	1

3 rows in set (0.000 sec)

ProductType:

productTypeID	typeDescription
1	Weight Machines
2	Free Weights
3	Cardio
4	Home-Gym Systems
5	Accessories

5 rows in set (0.000 sec)

OrderEquipment:

orderID	equipmentID	amount	cost
1	1	5	200.00
3	2	2	340.00
2	3	3	850.00

3 rows in set (0.000 sec)

Orders:

orderID	userID	orderDate	numItems	totalCost
1	3	2022-05-23	5	850.00
2	1	2012-10-04	7	2500.00
3	4	2021-08-13	15	6523.00

3 rows in set (0.000 sec)