PROJECT STEP 2 - FINAL

Boba Best-Teas

Garrick Chan
Leonel Garay

PEER FEEDBACK

Mitchell Hein

Great details in your description. You painted a very clear picture of what problem this DB is meant to solve. 200-400 calls per day, this tea shop must be popping! Your variable naming is very consistent, as well as your entity names, good job. I think the way you structured your relationships is well done. Whenever you want to build out a receipt to show an order for a customer, you will be able to go from Customer to their Order and then look in the Drinks_Instances table and do a query based on every drink with that Order's ID. You will then be able to use the Base Drink and Size for that Drink Instance to determine the cost per item and total it up, after using the many to many table to total the toppings as well.

I see 5 entities that all represent a single idea (and then adding in the relationship table for your many to many), and they each have datatypes and constraints and describe the relationships, which all are well done. You may want to consider making one of the drinks size costs in Base_Drinks NOT NULL. If you have all 3 sizes as null for a Base Drink, then it will cause an error when calculating the order total. I would maybe consider how you would implement that as well, as you may want constraints for certain Base Drinks on the front end. For instance, if the Medium Cost for a Base Drink is null, a customer should not be able to choose that size at checkout. You could also make all of the Size Costs NOT NULL, but I personally would choose 1 size that you know will always be an option in every drink. Topping Cost should also be NOT NULL. Even if a topping is free, it should still have a Cost of 0.

The ER diagram looks great, but I am not sure if the words are necessary. I have not seen that before in an ER diagram, and it can be hard to figure out which direction the word is supposed to describe the relationship. Other than that though, it looks great!

The Schema is also well done. I like how you structured it to all flow in 1 direction to the right, it makes it very easy to read and understand!

Questions I did not directly address in the above review:

- *Does the outline of entity details describe the purpose of each, list attribute datatypes and constraints and describe relationships between entities?*

The outline does address all of the necessary details and relationships with good detail!

- *Are 1:M relationships correctly formulated? Is there at least one M:M relationship?*

The M:M relationship between Drink Instances and Toppings is implemented well with the intersection table, well done!

- *Is there consistency in a) naming between overview and entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?*

The naming is all consistent between Table names, using caps and underscores. The attributes use camelcasing. The attribute names are abbreviated, but are consistent. I personally prefer the longer descriptive name over the abbreviation, but definitely nothing wrong with the approach you took. Great work!

PROJECT STEP 2 - FINAL

Boba Best-Teas

Garrick Chan
Leonel Garay

Veronica Jo

Hi Team Honir, love this idea! My manager at work is OBSESSED with boba so having a database like this would have been helpful when making orders for our team (yes, we've made team boba runs).

Does the overview describe what problem is to be solved by a website with DB back end?

- Yes, the project aims to keep track of a store's boba orders received through phone calls and aims to eventually allow for online ordering as well.

Does the overview list specific facts?

- Yes, the overview lists specific ranges for number of calls per day and cost per order.

Are at least four entities described and does each one represent a single idea to be stored as a list?

- Yes, there are 5 entities described with each representing a single idea.

Does the outline of entity details describe the purpose of each, list attribute datatypes and constraints and describe relationships between entities?

- Yes, the entities are clear in describing their purpose.
- The datatypes are all well though out, but there are some attributes where constraints are not specified when they should, such as:
  - Entity attributes that are PKs are not specified as NOT NULL
  - FKs as an attribute in an entity should be made NOT NULL
    - For example, in Orders, the cid attribute should be NOT NULL because an order should always be associated with a customer.
  - As Mitchell mentioned before me, I think the cost attributes should also be NOT NULL, especially if all base drinks are offered in all sizes.
- Yes, the relationships described between the entities were clear.

Are 1:M relationships correctly formulated? Is there at least one M:M relationship?

- Yes, all 1:M relationships were correctly formulated and there is at least one M:M relationships.
- I had a question about the 1:1 relationship between Customers and Orders. Does the database only track orders made within a single day? I ask because a customer could have made more than one order over the course of a week, month, or even a day (it's happened, people LOVE boba), which would make the relationship 1:M instead.

Is there consistency in a) naming between overview and entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

- The entity names were not explicitly used in the overview, but the process of a customer making an order was described in which you could match some of the entity names to the process step.
- Yes, entity names are plural and attributes names are singular.
- Use of capitalization was consistent in the entities, but not in the attributes.
  - Camel case was used in some of the attributes, but not others.
    - For example, in the Base_Drinks entity, dName (abbreviation for "drink name") versus scost (abbreviation for "small cost") instead of sCost.

PROJECT STEP 2 - FINAL

Boba Best-Teas

Garrick Chan
Leonel Garay

Laura Jones

I love this idea. You can see the real world application for this database. I love Boba and many small boba tea shops could use this.

- **Does the overview describe what problem is to be solved by a website with DB back end?**

Yes, it's clear why the Boba Tea place could use the website with this database backend. It would make ordering more accurate, faster, and easier for both the customer and the shop if they don't have to order over the phone and worry about miscommunication.

- **Does the overview list specific facts?**

  Yes it lists the amount of calls/orders, the general cost of the orders and the average daily numbers the shop gets in sales.
- **Are at least four entities described and does each one represent a single idea to be stored as a list?**

  Yes, the entities are clear and there are more than four.
- **Does the outline of entity details describe the purpose of each, list attribute datatypes and constraints and describe relationships between entities?**

  Yes it's very pretty descriptive. One thing I noticed is that there it does not specify not NULL for the primary key and forgein key for the entities. These should be not null as they are important for creating unique identifiers.
- **Are 1:M relationships correctly formulated?**
  Is there at least one M:M relationship?

  Yes the 1:M relationships are correctly formulated and there is a M:M relationship. I'm not sure why the order customer is a 1:1 relationship. I understand there can only be 1 customer to an order, but I would think a customer could order more than once in a day.
- **Is there consistency in a) naming between overview and entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?**

The entities are consistent in naming and the attributes are consistent in naming, but the entities and attributes are not consistent together. I also believe that attributes can only be one single string with no spaces. Those would need to be edited to be consistent with the entities camel case and being one single string. overall nice work!

PROJECT STEP 2 - FINAL

Boba Best-Teas

Garrick Chan
Leonel Garay

Una Lee

Hi Team Honir! I think boba combinations is a really fun application of what we are learning.

- **Does the overview describe what problem is to be solved by a website with DB back end?** Yes! Their website will allow a small Boba Tea shop to keep track of orders digitally rather than by hand, which has been slowing them down.
- **Does the overview list specific facts?** Yes! The overview provides specific ranges for daily order volumes.
- **Are at least four entities described and does each one represent a single idea to be stored as a list?** Yes! They outline 5 entities, which each represent a single idea to be stored as a list (lists of customers, orders, drinks, base drink options, and toppings).
- **Does the outline of entity details describe the purpose of each, list attribute datatypes and constraints and describe relationships between entities?** The outline does include all of these things. I only have a few small suggestions. The entity descriptions for drink_instances, toppings, and base_drinks are focused on the process by which a new drink_instance. I actually found this more confusing than just thinking about what each entity represents and how they are related.
- **Are 1:M relationships correctly formulated? Is there at least one M:M relationship?** The 1:M relationships are correctly formulated. I understand the rationale for customers and orders having a 1:1 relationship, but a 1:M relationship may provide more functionality. The outline does a good job explaining how they will be implementing the M:M relationship between drink_instances and toppings.
- **Is there consistency in a) naming between overview and entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?** I think there is some inconsistency between using underscores and camelcase (for example. Base_Drinks vs. dName vs. scost). I also find some of the use of acronyms hard to follow but that may just be me!

Hope this was helpful! Let me know if you have any questions.

PROJECT STEP 2 - FINAL

Boba Best-Teas

Garrick Chan
Leonel Garay

Actions Based on Feedback

ACTION	REVIEWER
Added NOT NULL constraint to drink size.	Hein
Added NOT NULL constraint to topping cost.	Hein
Renamed all entities to use underscores instead of CamelCase (e.g., tid to topping_id) and all lowercase so that naming is more descriptive and unified.	Hein, Jo, Jones, and Lee
Removed words on ER diagram connections.	Hein
Updated ERD and schema to reflect new naming convention.	Hein
Updated all Primary and Foreign Keys to include more constraints (UNIQUE, AUTO_INCREMENT, NOT NULL, etc.) within the parentheses (previously they were on the description).	Jo and Jones
1:1 customers - orders relationship changed to 1:M.	Jo, Jones, and Lee

PROJECT STEP 2 - FINAL

Boba Best-Teas

Garrick Chan
Leonel Garay

Project Outline and Database Outline

A small Boba Tea shop has been taking orders over the phone during the COVID-19 Pandemic, they receive between 200-400 calls per day with each order placed being anywhere between $6-$40 so in a slow day they make a minimum of $1,200. Their current system requires them to talk to each customer over the phone, write down their order, repeat the order back to the customer, then the written order is taken to the kitchen for an employee to start working on it. This slows them down and makes customers wait on their phone queue for too long before being able to place an order. The store wants to be able to input the orders on a computer instead of having to write them down and eventually give customers the option to do it online themselves and have the staff making the drinks being able to see the orders as soon as they are placed. With this system, ordering drinks and customizing each drink with toppings will be easier than ever for both staff and customers.

PROJECT STEP 2 - FINAL

Boba Best-Teas

Garrick Chan
Leonel Garay

Entities

customers: Keeps track of the customer using the following attributes:

- Customer ID (customer_id - INT, PK, NOT NULL, AUTO INCREMENT, UNIQUE)
- First Name (first_name - VARCHAR, NOT NULL): First name of the customer.
- Last Name (last_name - VARCHAR, NOT NULL): Last name of the customer.
- Phone Number (phone_number - VARCHAR): Using this format ###-###-####. Phone numbers must be unique, no two customers can have the same number.

orders: Tracks the customers to which the orders belong.

- Order ID (order_id - INT, PK, NOT NULL, AUTO INCREMENT, UNIQUE)
- Customer (customer_id - INT, FK, NOT NULL): ID of the customer who made this order.

drinks_instances: When a customer orders a particular drink from base_drinks, a drink instance is created. This instance is then connected to any toppings the customer wants to add. This allows a customer to order multiple of the same base drink, each with different toppings or sizes.

- Instance ID (instance_id - INT, PK, NOT NULL, AUTO INCREMENT, UNIQUE)
- Order ID (order_id - INT, FK, NOT NULL): References the order this instance is part of.
- Drink ID (drink_id - INT, FK, NOT NULL): References the base drink.
- Size (size - INT, NOT NULL): 0 for Small, 1 for Medium, and 2 for Large.

toppings: Toppings can be added to drinks_instances

- Topping ID (topping_id - INT, PK, NOT NULL, AUTO INCREMENT, UNIQUE)
- Topping Name (topping_name - VARCHAR, NOT NULL): The name of the topping.
- Cost (total_cost, DECIMAL [4,2], NOT NULL): Price of the topping.

base_drinks: Tracks drinks on the menu and the cost of each, based on size.

- Drink ID (drink_id - INT, PK, NOT NULL, AUTO INCREMENT, UNIQUE)
- Name (drink_name - VARCHAR, NOT NULL): The name of the base drink.
- Small Cost (small_cost - DECIMAL [4,2], NOT NULL): Price of a small drink.
- Medium Cost (medium_cost - DECIMAL [4,2], NOT NULL): Price of a medium drink.
- Large Cost (large_cost - DECIMAL [4,2], NOT NULL): Price of a large drink.

PROJECT STEP 2 - FINAL

Boba Best-Teas

Garrick Chan
Leonel Garay

Relationships

A 1:M relationship between customers and orders. An order made by exactly one customer, and a customer can place many orders.

A 1:M relationship between orders and drinks_instances. An order is composed of at least one drink instance, and any given drink instance belongs to exactly one order.

A M:M relationship between drinks_instances and toppings. Drinks instances can have 0 or more toppings. Toppings can be in 0 or more drinks instances. A table called drinks_toppings, containing foreign keys to drinks_instances and toppings, is used to achieve the M:M relationship between drinks_instances and toppings.

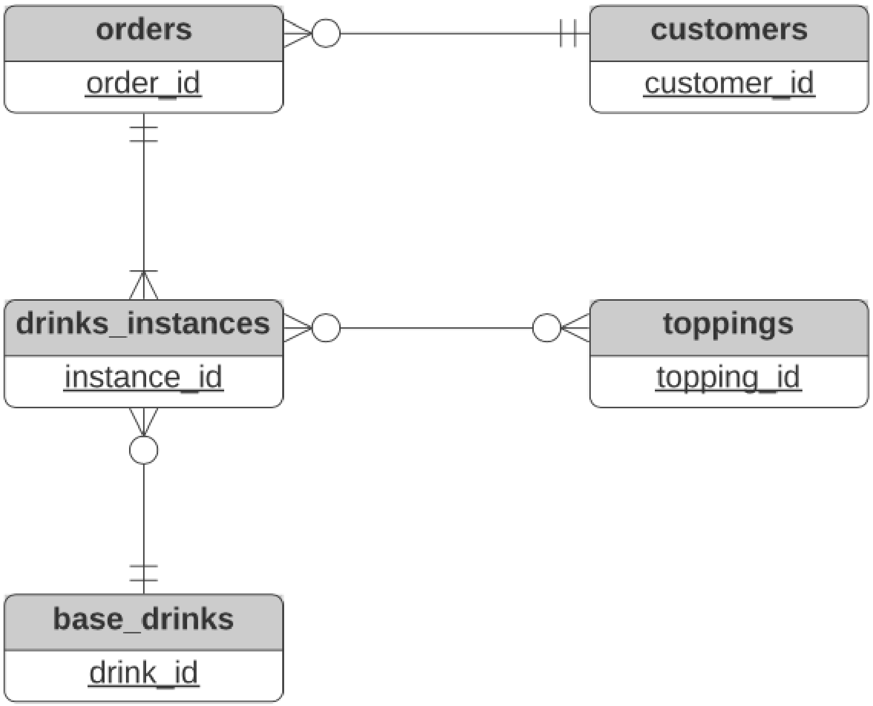
A 1:M relationship between base_drinks and drinks_instances. Any given drink instance is based on exactly one base drink. A base drink can be the basis of 0 or more drink instances.

PROJECT STEP 2 - FINAL

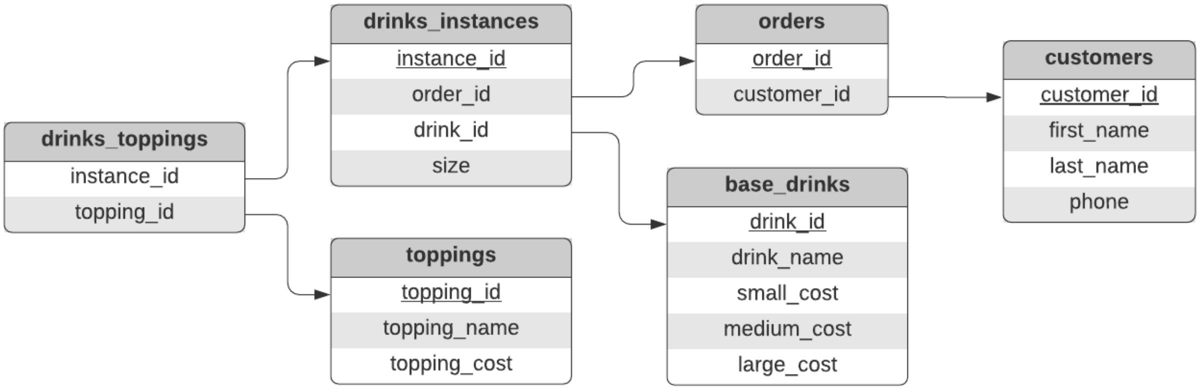
Boba Best-Teas

Garrick Chan
Leonel Garay

Entity-Relationship Diagram



Schema



drinks_toppings is used to achieve the M:M relationship between drinks_instances and Toppings

PROJECT STEP 2 - FINAL

Boba Best-Teas

Garrick Chan
Leonel Garay

DRAFT (ed Submission)

Fixes Requested

“For your M:M relationship between Order and Drink, I was unable to find an entity named Drink. I see one named Drinks but you also list a 1:M relationship between Order and Drinks so I was not sure which one was which. Is an entity missing? At this moment, although you list 4 relationships, right now I only see three since there is no Drink entity (I only see Drinks). Also, for your M:M relationship, provide a quick explanation of how it will be implemented. You will need to describe the intersection table you will use, give it a name (drinkOrders?), and what foreign keys it will have in it (PK of Drink and PIK or Order.)”

Solutions

- Removed the Employees entity and its relationship with the Orders entity.
- Added two new entities: Toppings and Drinks_Instances.
- Changed entity name from Drinks to Base_Drinks.
- Created new relationships: Drinks_Instances to Base_Drinks, and Drinks_Instances to Toppings.
- A table called Drinks_Toppings is used to achieve the M:M relationship between Drinks_Instances and Toppings.

PROJECT STEP 2 - FINAL

Boba Best-Teas

Garrick Chan
Leonel Garay

Project Outline and Database Outline

A small Boba Tea shop has been taking orders over the phone during the COVID-19 Pandemic, they receive between 200-400 calls per day with each order placed being anywhere between $6-$40 so in a slow day they make a minimum of $1,200. Their current system requires them to talk to each customer over the phone, write down their order, repeat the order back to the customer, then the written order is taken to the kitchen for an employee to start working on it. This slows them down and makes customers wait on their phone queue for too long before being able to place an order. The store wants to be able to input the orders on a computer instead of having to write them down and eventually give customers the option to do it online themselves and have the staff making the drinks being able to see the orders as soon as they are placed. With this system, ordering drinks and customizing each drink with toppings will be easier than ever for both staff and customers.

PROJECT STEP 2 - FINAL

Boba Best-Teas

Garrick Chan
Leonel Garay

Entities

Customers: Keeps track of the customer using the following attributes:

- Customer ID (cid - INT, PK): An auto-incrementing integer that is unique to a customer.
- First Name (fName - VARCHAR, Not Null): First name of the customer.
- Last Name (lName - VARCHAR): Last name of the customer.
- Phone Number (phone - VARCHAR): Using this format ###-###-####. Phone numbers must be unique, no two customers can have the same number.

Orders: Tracks the Customers to which Orders belong

- Order ID (oid - INT, PK): An auto-incrementing integer that is unique to an order.
- Customer (cid - INT, FK): The id of the customer who made this order.

Drinks_Instances: When a customer orders a particular drink from Base_Drinks, a drink instance is created. This instance is then connected to any Toppings the customer wants to add.

- Instance ID (iid - INT, PK): An auto-incrementing integer that is unique to a drink instance.
- Order ID (oid - INT, FK): References the order this instance is part of.
- Drink ID (drid - INT, FK): References the base drink.
- Size (size - INT, Not Null): 0 for Small, 1 for Medium, and 2 for Large.

Toppings: Toppings can be added to Base_Drinks, each increases the cost of the base drink.

- Topping ID (tid - INT, PK): An auto-incrementing integer that is unique to a topping.
- Topping Name (tName - VARCHAR, Not Null): The name of the topping.
- Cost (tcost, DECIMAL [4,2]): Price of the topping.

Base_Drinks: Tracks drinks on the menu and the cost of each, based on size.

- Drink ID (drid - INT, PK): An auto-incrementing integer that is unique to a base drink.
- Name (dName - VARCHAR, Not Null): The name of the base drink.
- Small Cost (scost - DECIMAL [4,2]): Price of a small drink.
- Medium Cost (mcost - DECIMAL [4,2]): Price of a medium drink.
- Large Cost (lcost - DECIMAL [4,2]): Price of a large drink.

PROJECT STEP 2 - FINAL

Boba Best-Teas

Garrick Chan
Leonel Garay

Relationships

A 1:1 relationship between Customers and Orders. An order made by exactly one customer, and a customer can make at most one order at a time.

A 1:M relationship between Orders and Drinks_Instances. An order is composed of at least one drink instance, and any given drink instance belongs to exactly one order.

A M:M relationship between Drinks_Instances and Toppings. Drinks Instances can have 0 or more Toppings. Toppings can be in 0 or more Drinks Instances. A table called Drinks_Toppings, containing foreign keys to Drinks_Instances and Toppings, is used to achieve the M:M relationship between Drinks_Instances and Toppings.

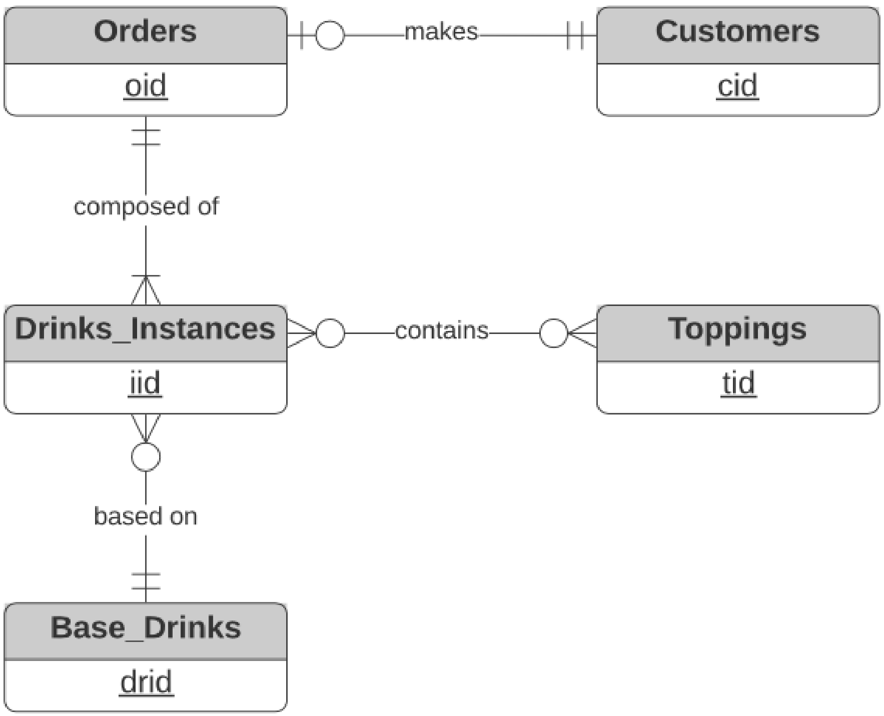
A 1:M relationship between Base_Drinks and Drinks_Instances. Any given drink instance is based on exactly one base drink. A base drink can be the basis of 0 or more drink instances.

PROJECT STEP 2 - FINAL

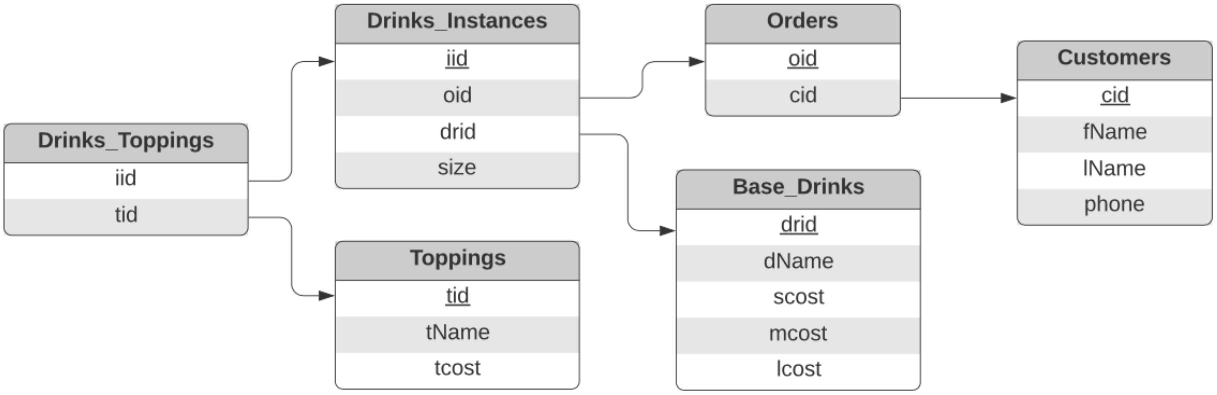
Boba Best-Teas

Garrick Chan
Leonel Garay

Entity-Relationship Diagram



Schema



Drinks_Toppings is used to achieve the M:M relationship between Drinks_Instances and Toppings