Database Design

Project Design Document

Section 910 Group 9

Nick Mastroberardino

William Pierce

Benjamin Ramon

Seethalakshmi Senthilnathan

Ian Stephenson

Department of Computer Science and Engineering

Texas A&M University

09/15/2020

Table of Contents

1 Executive summary

Currently, there is a need for an application, from the Marketing department of the May's Business School, that can query business names, reviews and business service questions so students can come up with answers to these. The objective of the project is to provide an user friendly application that can query the data as needed. The goal of this project is to create a database and Graphical User Interface (GUI) that can go with it to allow users to easily access the information they want.

Our team has come up with the high-level design for the implementation of the database. The entities that we have decided are important for the efficient functioning of this database are Reviews, Users, Customers, Customer Questions, Businesses, Addresses, Contact Information, Answers, Social Media, Franchise and Logo. The business entity will contain information on all the business names. The social media entity will contain the social media usernames of prominent networking sites of the business. The logo entity contains the logo for the business. The franchise entity will contain franchises of each business, with the address and contact information for each franchise stored in the address and contact information entities, respectively. The customer entity will store customer information for each franchise. The reviews and customer questions entities will hold the information the customer wants to communicate to the business. The user entity will store information about the users of the database, and their answers to reviews and customer questions will be stored in the answers entity. The elements of the high-level design and the relationship between them are discussed in more detail in the document. This database will be implemented in SQL.

This design gives the users an easy way to access the reviews and questions for particular businesses and franchises. The end result of this project will be a useful application for students to view and make conclusions on a large dataset without having to sift through them manually. Another benefit of this design is that it stores the answers from the students and can be used by the faculty for future reference. Along with the GUI, this application will meet all the needs of the faculty of the Marketing department. If we receive funding from the department to continue this project, we will deliver the working application in three weeks, with intermediate products every week. We hope that you consider our project for the funding.

2 Introduction

2.1 Purpose of the Project

Our team is tasked with creating an interactive system where users can input names of businesses, services, restaurants or other items. They then gain access to customer reviews and business service questions. This project will improve user experience with accessing details on businesses with an user-friendly interface. The target audience for this project would be the students in the May's Business School that will be using the program within their class, along with the faculty at the school that have given us the task of designing and implementing this database.

2.2 Needs Statement

This program will allow the students to access the information about these businesses much easier, and it will be presented in a much more organized format than it was previously. It will also allow for much quicker transactions between the user and the database.

3 High Level Entity Design

The high level entity design we made as the following entities: Reviews, User, Customer, Customer Questions, Business, Address, Contact Information, Answers, Social Media, Franchise, Logo. We have broken down the details of each entity below.

The Review entity will contain all the reviews related to the franchises. The entity has attributes like the user name of the customer that left the review, a review ID, the review text, the date the review was posted and the rating given by the customer. This entity can be used to query reviews for a certain business. The entity will be the focus of queries in this database.

The User entity keeps an account of all users that interact with this database. The entity has the attributes username, user id, user email address and user phone number. The user information needs to be saved to keep log of who makes changes to the database. The user can make changes to the answers entity, and this entity can be used to track these changes. The entity is thus responsible for maintaining the durability and consistency of the data.

The Customer entity keeps track of information of customers that left a review on a particular business. The entity has the attributes customer name, customer id, customer email address and customer phone number. The customer information needs to be stored to give users or businesses a way to contact them. Customers are an important part of a business and therefore it is essential to keep track of the customer base. The entity will be used once answers for reviews or questions are created and need to communicated to the customer.

The Customer Questions entity exists to store all pertinent information regarding a given proposed question. This entity will maintain information such as the actual question, the user who asked it, the business it's associated with, and an identification number, and it will be connected to a franchise. The reason that this entity needs to be created is because customer questions will all need to be stored as discrete, individual pieces of text. For a given business, there can be many associated questions, and this entity will represent a single question, but the set can hold many questions that a franchise can pull from. There needs to be a way to store individual questions and this entity is the way to store them. The customer questions entity, as stated, will store individual questions. Each individual question entity will also have an associated business and customer. The question will connect with both the customer and the business, and businesses will be able to pull questions from the customer questions entity set.

The Business entity is a way to abstract each business. A business entity will maintain all the key information needed to identify the business, as well as have a way to get all the reviews, questions, and answers that are associated with it. The business will operate as a focal point for accessing information within the project, and the business name will be a primary access method to many other entities. The business entity has a crucial role in this system. It will be the primary maintainer for all information regarding a given business, as well as act as a kind of hub for the rest of the system. This entity will be needed to access reviews, questions, and answers. The role of the business entity is to store all the information about a given business. It will also act as a super class to the franchise, and thus have an important role in connecting to the rest of the system.

The Contact Information entity provides a way for the user to contact the franchise, which results in information regarding the business so that a user can be more intrigued by the franchise, and what it has to offer. We feel that a contact system is needed in a design like this because a user needs to have a way to communicate with the franchises, as well as creates a new object that allows a business to have multiple ways of contact, email, phone number, etc. Entity will store data regarding the contact information for a franchise, and will help connect a user to a business.

The Answers entity lets a user answer customer questions. This results in a faster response to the questions that other customers have. Since there would be customer reviews and questions, we felt that the other users should have the opportunity to respond and answer the questions of the other users. Entity will store a database for user answers to customer questions, and will also provide a way for users to provide feedback regarding a business.

The Social Media entity provides access to the social media accounts of the business. This way, it gives the user access to the social media pages in order to learn more about the business or potentially use it as a method to get in contact with them. When a user looks up a business and goes to the business' website, they almost always provide links to their social media pages, so we felt it would be applicable to include it in our database. The social media entity will contain the social media handles for the business for multiple platforms such as Facebook and Twitter along with the website for the business.

The Franchise entity would give us distinction between the different stores within a franchise if multiple were provided for the dataset. For instance, if there were two McDonald's, this is the object that would distinguish between the different stores to access franchise specific information. We felt this was necessary to include as a way to keep businesses of the same franchise together while also distinguishing between their store specific information such as address and contact information. The franchise entity will store information relevant to each specific store of a business such as the name of the business, its store number, and its rating.

The Address entity would give us the ability to store the physical locations of the different franchises including StreetNumber, StreetName, City, State, and ZipCode. To differentiate between franchises, the trait most often used is the different addresses. Because of this we felt that an address entity would be an important component. This will both allow us to store the physical location of a store while also granting us an easy way to identify different franchises.

The logo entity would hold the image of the logo that is associated with a business, allowing both the business and the franchise to access it, but being tied solely to the business. We felt that this would be an important entity specifically because we are composing this database for a marketing class. The information that a marketing student would most likely be interested in include the business traits that are designed to appeal to customers, which a logo is an example of. This entity will allow us to save the logo of a business, and the franchise can access it.

4 Low Level Entity Design

The interactions and relationships between the entities are described in detail below.

The Reviews entity has three relationships, one to the franchise, one to the user entity and another to the customer entity. The relationship to the franchise will allow us to relate the reviews to the particular franchise. This is a many-to-one relationship as the review can be related to only one franchise, but the franchise can have many reviews. The ratings from the reviews entity will also be used to calculate the total rating of the franchise. The relationship with the customer entity is to log which customer left which review, and it also a many-to-one relationship as a review can only be related to one customer, but the customer can leave many reviews. The relationship with the user entity is so the user can view these reviews and create answers to many of these reviews. This entity will be the focus queries from the user. This entity can be used to easily store reviews and retrieve information based on the business or franchise name. The entity can be used to quickly bring up the reviews related to the particular business, and can be used to calculate total rating easily. The risks with using this entity is we have assumed that we will get data for the attributes we defined. One assumption is that a rating is provided with each review in some numerical form that allows us to do calculations on it. Another difficulty with this part is the making unique IDs for each review when we could have a large amount of reviews.

The User entity has relationships with the business, franchises reviews, customer questions and answers entities. The user needs access to the business to be able to see the details such as the business' logo and franchises related to it. The user also needs access to view all details related to the franchises. The relationship with the reviews entity is so the users can view the reviews given to the business and they can come with answers that they can add to the answers entity. The user also needs to be able to view customer questions and write answers for these as well. All these relationships are many to one as the user can view many instances of each entity mentioned and multiple users can view any instance of these entities. The user information is important to keep track of the additions made to the answers entity. This will ensure that we have a log of who made what additions. It will also help with ensuring consistency and durability of the data. We are assuming that we will be able to get this information about the user and add it to the database.

The Customer entity is connected to the franchise, customer questions and reviews. The customer is connected to the franchise in an infinite-to infinite relationship, that exists to establish what franchises the customers have been in contact with. The relationship with reviews and customer questions is so that we can know which reviews and questions written by which customer. These are many-to-one relationships as the review or question can be written by only one customer but a customer can write multiple reviews and questions. As the focus of the database is to allow users to query reviews and let them come up with answers, it is important to have details about the people who wrote these reviews. It will be an easy way for users to connect customers with answers to their review and questions. The risk with this implementation is that we will provide access to this sensitive information to the users. The data provided might not have all these details either.

The Customer Questions entity has four relationships. First is the relationship with the answers, where a question will have one or more associated answers. Second, a question is related to a user in that many users can answer many questions. There is also a relationship to customers, where a customer can have many questions. Finally, there is a relationship between the user and the franchise, where many questions can be asked about and associated with a particular franchise. The thought process for defining this entity was so that we could maintain a central database that just contained questions. This will allow us to create an association between a question and a company, and from there we would be able to effectively pull all questions about a company from the questions database. There must be an effective way to store all the questions, and this database can store and retrieve all the questions easily. The benefits of this entity is that we will have a central place to access all the questions that exist and it will be searchable by the associated business. The potential risks that exist with this implementation is that there isn't always a clear way to get answer information.

The Business entity will have four relationships. First, a single business can have many franchises associated with it. Next, a business will have a one to one relationship with the social media entity. Third, a business will have another one to one relationship with a logo entity. This entity was defined as a general connection point between many other entities. The business is a more general version of the franchise, but the business name will be a key access point to most of the information that is stored. The reason that this entity is necessary is because the businesses will be a central point of the application, and it will be much like an interface for accessing information. The business entity has the benefit of interfacing with the rest of the database through the franchise entity. It is very clear exactly what makes up a business entity. The risk with the business entity is that it exists behind the franchise, so fragments in the franchise data would make it hard for the business to access all the information.

The Contact Information entity has a relationship with the franchise entity. This will be a one to one relationship between the two entities. We felt that this entity is better sought to belong to the franchise rather than the business. Since if a user were to contact a franchise, the franchise could just redirect them to the certain business in their specific location or some other method of service that allows them to be serviced. With everything going on in the world right now, a franchise's contact information is one of the most important things that a business can have. Also, a full database of contact information would result in a cleaner and easier way for a user to contact them. If a franchise does not have a form of contact, or a faulty way of communication, it can cause problems for a user. This can also lead to a faulty database with accessing empty values from tables.

The Answers entity belongs to the user entity, with an infinite to infinite relationship. This also has a one to infinite relationship with the user questions entity. We felt this belongs in our database because of the source of the information it can propose, as well as the overall cohesiveness this entity can have with its corresponding ones. This provides a faster response to a user question, since a member or owner of the franchise cannot always answer all the questions a user might have. Having an answers entity might not always be the best thing for our database. A lot can go wrong with having users answer other users questions. If they just don't answer at all, this will again create gaps in our database. And if they do answer, the answers might not be the answer that the user that posted the question was looking for.

The biggest relationship for the Social Media entity is between social media and business entity. Since the social media entity contains the handle for each of the different social media platforms as attributes and businesses only have one page per platform, the relationship is one to one. We decided to give it a relationship with businesses rather than franchises because, typically, it is the main business that runs social media pages, not each individual franchise. Having social media as a separate entity allows for more organization in the database. It keeps all of the pages together without crowding another entity with too many attributes. The biggest risk in this part of the database is that, because not every business has social media pages on every platform, there could be a lot of empty spaces within the table, leading to unclean data.

The Franchises entity contains many relationships, the biggest one being an infinite to one with businesses, since each business can potentially have many franchises. It also contains infinite to one relationships with reviews and user questions, as each franchise can have multiple of these associated with it. This entity contains a one to one relationship with address, and a one to one relationship with contact information. We thought it would be best to use franchises to keep track of this information, as these pieces of information tend to be directed towards a specific store, rather than the business as a whole. Having this entity allows us to keep track of the different franchises for each business without needing to sort through the many franchises of other businesses to find the specific store we want when retrieving data. The biggest risk involved with this entity is finding a way to keep track of the franchises in the table. We would either need to develop a system of making ids for the franchises or use a store number given in the data, but we will not know what method we will need to use until we see the data firsthand.

The Address entity has only one relationship, a one-to-one with the franchise. The address is an element of a franchise and because it is used to identify a franchise, there will be one for each franchise and it will not be connected to the larger business entity. We used an address to identify a franchise as that is the way that most people currently specify a franchise. For example, when saying “Let’s go to walmart”, if there are multiple stores nearby, one would have to specify the street name. We included the more minor elements like city, zip code, etc to help us be able to store all important elements of a franchise’s address. This will allow us to properly distinguish between franchises. We will also be able to store the locations of these franchises which can be very important when dealing with a marketing perspective. If an address is not provided or if a business is not physical, but rather online, we will have no way to distinguish it from other franchises, besides its ID. This is remedied by the fact that most businesses with online stores have no need for different franchises but are instead a single entity.

The Logo entity has a one-to-one relationship with businesses as each business will have its own logo. The logo also has an infinite-to-one (many-to-one) relationship with franchise as franchises might also need access to the logo component as they still use it. Since most logos apply to a business and the franchises incorporate the same logo, we felt safe assigning this to the business and letting the franchise access it. This will allow us to save memory by keeping the logo tied to the business rather than a new instance for each franchise. If a franchise has a different logo than the business logo, our system is not designed to handle that.

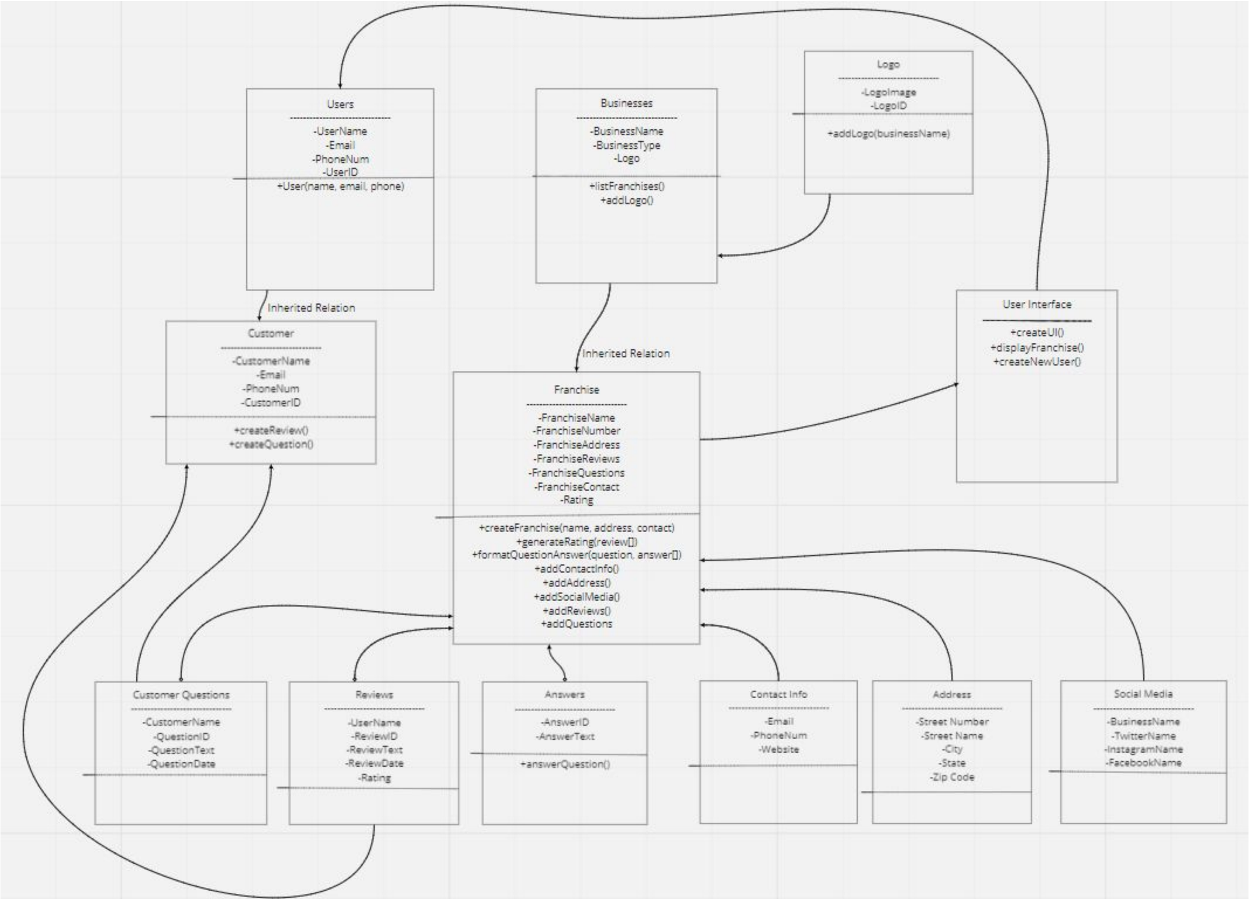

Figure 1: UML Diagram

The UML diagram above characterizes the classes and entities needed for our system. Each class has both private data members and some public methods. The diagram does not list out getters, setters, constructors, or other trivial functions. The clear central class of the system is the franchise, which gets information from all other classes to provide output to a user interface. There are also two instances of inheritance listed, business to franchise and user to customer.

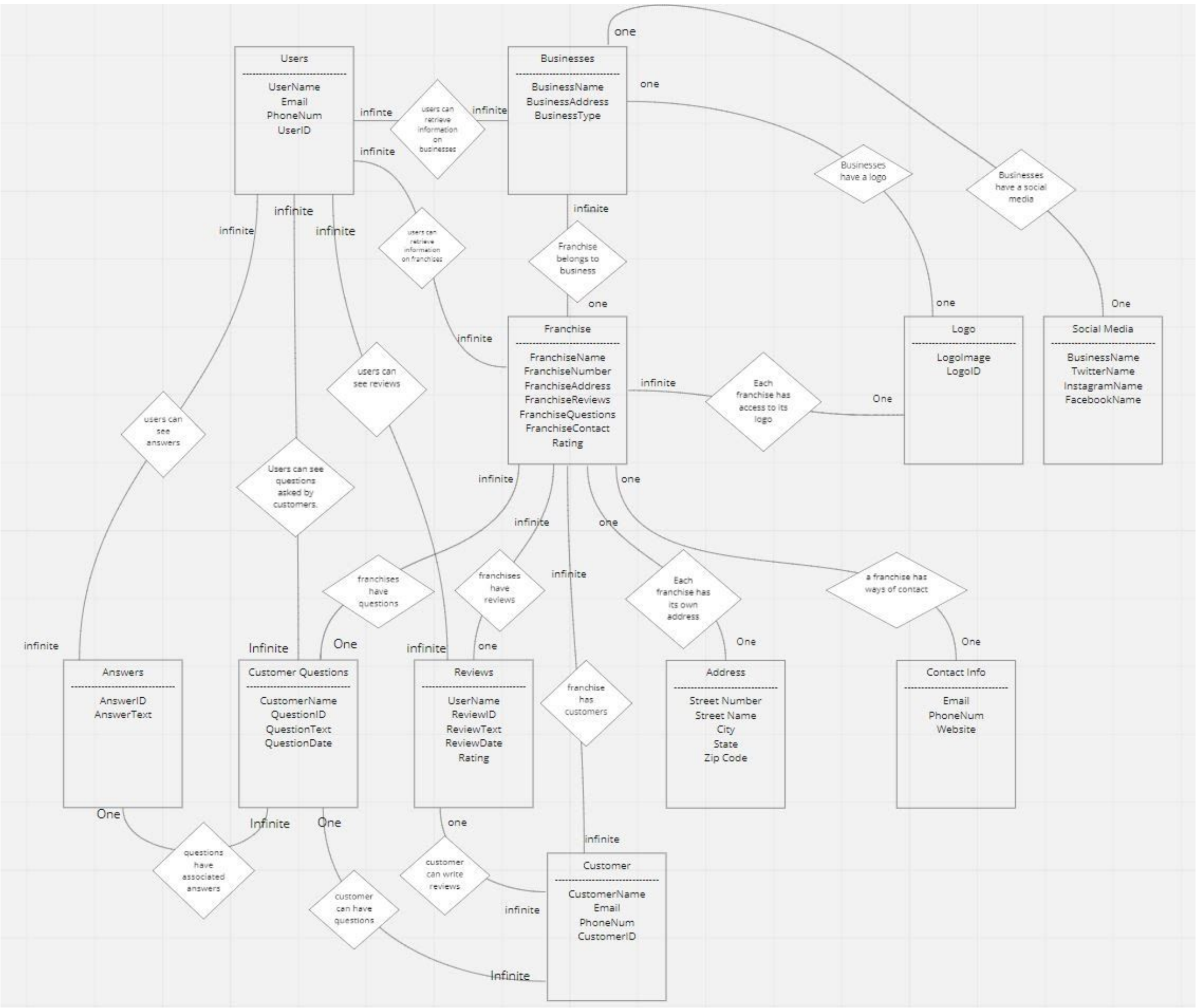

Figure 2: Entity Relationship Diagram

This diagram represents all of the relationships each of the entities have between each other and why the relationship exists. It also displays what kind of relationship each interaction is: one to one, infinite to one, or infinite to infinite.

5 Expectations

5.1 Benefits

To start with the list of benefits, we can start with the core of our system which is the benefits of franchise. This entity allows us to maintain a track of all of the different types of businesses that are obtained through data without having to have different types of objects for each different type of franchise or business when receiving data. Then going into the business entity, which provides a separate way of accessing information regarding a specific business within a franchise. This simplifies the information that would all be stored in a single franchise and breaks it down into a business entity.

We also have entities regarding the customer questions and answers. To start, the customer questions provide a great way of connecting many entities and linking their information together. The customer questions entity links to answers, user, and the customer. And the answers give benefits by allowing responses that users and customers can access. Also, the answers entity gives a way of answering the questions without having to contact the franchise.

Regarding the social media, contact information, addresses, and logo entities, their benefits remain cohesive. Since these are all recognizable traits of the company, we benefit from the entities by allowing us to have a distinguished set of information that differentiates the franchises from one another. The last entities are the users and reviews, the users allows us to keep track of who has made a contribution or has provided an answer to the queries brought up. The reviews entity gives the benefits of having an easy way to gather ratings and reviews on a business quickly.

5.2 Assumptions

For assumptions, regarding franchises and businesses, we are assuming that each business is part of a single franchise. And each franchise has multiple businesses, along with the general assumptions that each type of business will fit within our created entities.

To begin, our assumptions for the questions entity is brief. We are simply assuming that there is a full database with questions about the franchise and the business. Regarding the answers, we do not expect there to be answers to all of the questions, but most of them. This way there are not too many gaps in our database, and questions can be updated and added to when needed.

Regarding the social media entity, we are assuming that a franchise has at least one source of social media on at least one major platform. For contact information, the franchise should have all methods of contact, so a form of email, phone number, and a website. A logo for a franchise is also mandatory for recognition of a business. Lastly, every business should have a valid address that can be accessed by a user. For users and reviews, the assumptions are just that there is enough information for each entity in the database that will allow us to create correct assumptions regarding the franchise and what a user thinks of it.

5.3 Risks

The biggest risk we are facing is the fact that we know almost nothing about what the data we will receive will contain or how it will be formatted. Because of this, we designed our database in how we believed the data would be formatted. However, we will not know until we receive the data, and may need to make changes at that point.

The biggest example is through our user and customer entities. We are not sure whether or not we will be provided information about the users of the database or the customers that have provided reviews or asked questions. For this reason, we may be forced to change the layout of our database, and one of these entities may potentially be useless.

Another risk we will be facing is having unclean tables. We formed the entity attributes based on what information we believe will be provided based on commonly searched information. However, if these attributes are not provided in the data at all or some businesses do not have them, the tables we create could possibly be partially empty.

5.4 Issues

Our biggest issue designing this database was starting with no information about the data needed or how it would be used. We did our best to design a database around this fact, but we will not know how well our database fits the table until later in the design process. We are able to change the database slightly when we receive that data, but it would have been more beneficial to have all specifications beforehand in order to best structure the database around to data.

We also did not know how the data will be used in the class it is being designed for. The instructions stated that queries would be used as input for generating responses to reviews and answers. This does not give us much information as to how these queries would be used in the database, while telling us they will be. We would need to talk further with the client to make sure we understand fully how this data will be used to design the database in the most efficient way possible.