Denormalization

Denormalization MenuItems Class: The MenuItems class can have the MenuCategories, SpiceLevels, MeatEntrees, and subsequently, Meats classes absorbed into it as text attributes.

This would require the creation of database triggers to validate the input value for these text attributes at the time the records are updated. These triggers would ensure that the records are filled with only acceptable values.

Also, an additional acceptable value, 'N/A', for the Meats attribute should be included in the list as this will allow us to minimize the number of null values included in our data.

Additionally, the MenuCategory attribute can include a code at the beginning to denote meat entrees along with the name of the entree, i.e. 'ME - Chop Suey'. This code could denote that 'N/A' is not a valid entry for these items by way of a trigger.

Denormalization Customer Account: The customer account is denormalized by having the redundant data in the customer table since a customer is not defined by the contact name, email, phone number. This means a contact for a customer may be in many different customer accounts. This should be ok since the customer may have the same contact, but be considered a different customer.

This means that the Miming's account has to hold data integrity through the structure of relationship between a customer and account. A customer is defined through the customer name attribute and the address of the customer and the account is identifiable through the account ID.

The customer and account should be identifiable meaning there should only be one account per customer which will help restrict customers to one and only one account.

There is still some data validation to be implemented with triggers that we cannot do on the structure such as creating a customer with a duplicate company name and company department and having the customer fill out all the information for a company to keep the account either a company or individual account and no in betweens