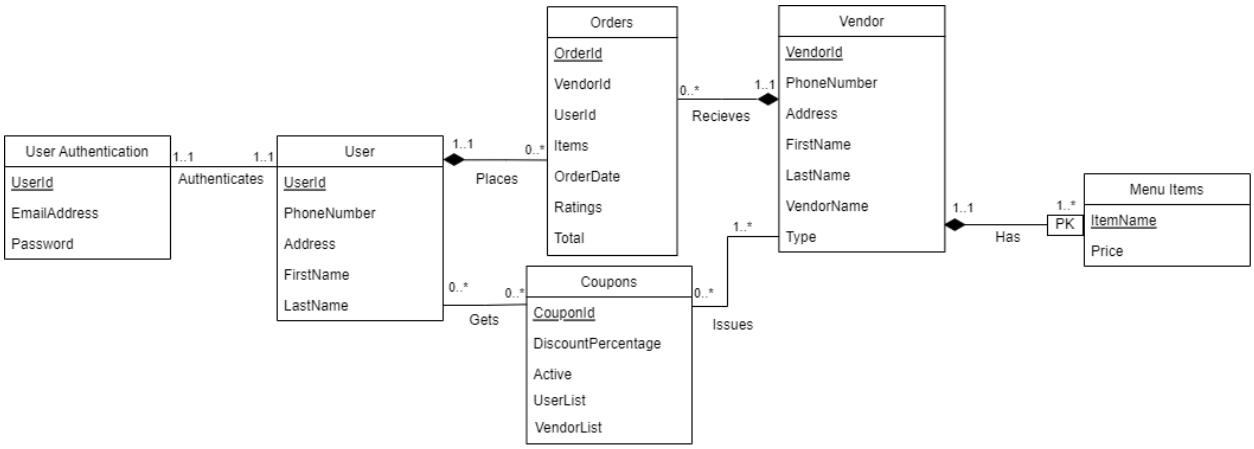
Stage2

UML Diagram:



Assumptions:

- The database has Users. Each is uniquely identified by the UserId, and also has typical user attributes, that is, phone number, address, first name, last name.
- The User Authentication information is used to authenticate each user and is uniquely identified by the UserId. It has attributed email address and password, which are commonly used by Users to log into their account in most websites.
- Each Vendor is uniquely identified by the VendorId. It also has as attributes the Vendor's phone number, the Address, FirstName, LastName, VendorName, Type (the type of the Vendor's cuisine).
- Each Order is uniquely identified by its OrderId. It has attributes VendorId, UserId, Items (the items contained in that order), OrderDate, Ratings (a number rating the quality of service the User received from the Vendor), and Total (the total price of the order).
- Each Coupon is uniquely identified by its CouponId, and has attributes DiscountPercentage, Active (whether this Coupon is active or has expired), UserList (this is a list that contains all the Users that got this Coupon) and VendorList (this is a list that contains all the Vendors that are currently issuing this Coupon).
- Each Menu Items is identified by its item name and Vendor and has attributes ItemName and Price.

Description:

- Each User (where User is the customer in our application) is authenticated by exactly one set of User-Authentication information, and User-Authentication information concerns exactly one User.
- Each User can place multiple (that is, zero or more) Orders, and an Order must always come from exactly one User.
- Each User can get zero or more Coupons, and each Coupon can be received by zero or more Users.
- Each Vendor (where Vendor is the service provider in our application) can receive multiple Orders, but each Order must be given to exactly one Vendor.
- Each Vendor can issue multiple Coupons but each Coupon must have at least one Vendor that is related to.
- Each Vendor can have at least one Menu Item, but exactly one of the available Menu Items (for example, steak).

Logical Design (Relational Schema):

User Authentication(UserId: INT [PK], EmailAddress: VARCHAR(255), password: VARCHAR(255))

User(UserId: INT [PK] [FK to User_Authentication.UserId], PhoneNumber: INT, Address: VARCHAR(255), FirstName: VARCHAR(255), LastName: VARCHAR(255))

Orders(OrderId: INT [PK], VendorId: INT [FK to Vendor.VendorId], UserId: INT [FK to User.UserId], Ratings: INT, Items: [VARCHAR(255)], OrderDate: DATE, Total: REAL)

Coupons(CouponId: VARCHAR(255) [PK], DiscountPercentage: INT, Active: BOOLEAN, UserList: [INT], VendorList: [INT])

Vendor(VendorId: INT [PK], PhoneNumber: INT, Address: VARCHAR(255), FirstName: VARCHAR(255), LastName: VARCHAR(255), VendorName: VARCHAR(255), Type: VARCHAR(255))

Menu Items(ItemName: VARCHAR(255), Price: REAL)