Meal Sharing: Banquet

Purpose: Similar to the way that ride-sharing allows people to obtain transit cheaper and more convenient than with a traditional taxi service, meal-sharing is intended to enable people to arrange group meals that will be cheap and convenient for customers while providing a reasonable income to the food preppers who put in a similar amount of effort to what it would have taken to feed themselves.

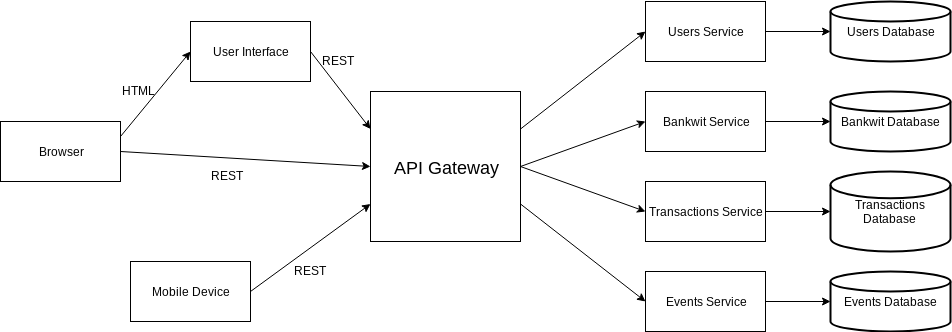
There is to consider the chicken and the egg problem. No customers will use the app if there aren’t already vendors, and no vendors will register and put effort towards making large elaborate meals if there isn’t already a customer base on the app. For this circumstance, I would advise taking advantage of locally owned food trucks and letting them use the app to advertise. Once a substantial customer base is using the app, pivot to encouraging regular folks to advertise backyard BBQ style events. Cohousing communities can also use the app to invite folks from neighboring communities for dinner for a small fee.

Our hope is to reduce the cost of meals as well as to reduce average food waste and bring communities closer together at the same time.

Functional requirements:

* Customers can create a profile and securely log in to it and out of it
* Customers can specify a price limit and a radius they are willing to travel
* Preppers can post an event on the app specifying the location as well as the start and end of availability and links to recipes on AllRecipes.com to show what’s on the menu.
* Preppers can specify the event as “A La Carte” or “Family Style”
  + A La Carte events are priced per menu item. Customers can load up their virtual “tray” with items, go to checkout, be shown the price and the estimated wait time, and choose if they want to make the purchase. They will then be issued a receipt which they can use to receive their food from the prepper.
  + Family Style events have one entry price. The customer chooses to purchase and is issued a QR code that can be used to gain access to the event.
* Preppers can set limits on how many of a particular item can be purchased in “A La Carte” events and limits on how many customers may purchase tickets for “Family Style” events depending upon how much food the prepper plans to make.
* Customers can make secure purchases using credit card, debit card, or PayPal
* Customers can securely save their payment methods to their accounts for future use
* Customers can pull up a record of their past transactions
* Admins can view the logs of events to keep the databases consistent

Architecture:

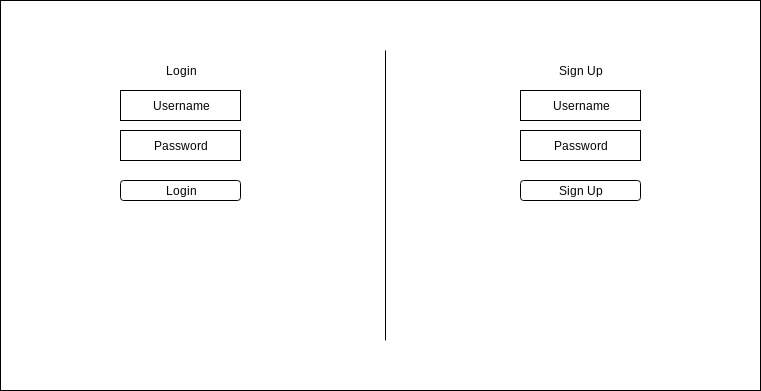


Databases:

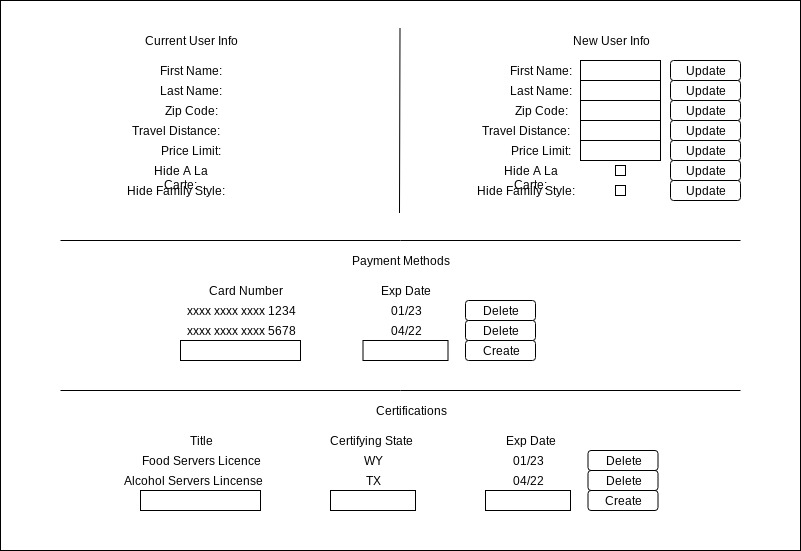
* Users
  + People
    - UID
    - email\_address
    - password
    - Admin (true/false)
    - First\_Name
    - Middle\_Name
    - Last\_Name
    - Home\_Address
    - Minimum\_price
    - Max\_price
    - Travel\_distance
    - Location (zip code)
    - Approves\_Family\_Style (true/false)
    - Approves\_A\_La\_Carte (true/false)
  + Payment methods
    - PaymentID
    - UID
    - Type
    - Card\_Number
    - Exp\_Date
    - CVV
    - email\_address
    - password
  + Certifications
    - CertID
    - UID
    - Title
    - Certifying\_State
    - Exp\_Date
* Transactions
  + Transaction\_Table
    - TransID
    - From\_UID
    - To\_UID
    - Banquet\_ID
    - DateTime
    - Price
    - Items/Description
    - Status (Pending/Received/Cancelled)
* Banquets
  + Instances
    - InstanceID
    - HostID
    - Instance\_Name
    - Address
    - Location (zip code)
    - Start\_DateTime
    - End\_DateTime
    - Type (A\_La\_Carte/Family\_Style)
    - Entry\_Price
    - Entry\_Limit
    - QR\_Code
  + Menu Items
    - ItemID
    - InstancesID
    - Item\_Name
    - Item\_Description
    - Stock\_Limit (yes/no)
    - In\_Stock
    - Number\_Claimed
    - Items\_Price
* Events
  + AggregateTable
    - AggregateID
    - AggregateType
    - AggregateStatus (Idle/Processing/Completed)
  + EventTable
    - EventId
    - AggregateID
    - EventType
    - EventData

Wireframes: Other than the login page, all pages will have a pane on the left that lists all other pages (except the events page) for easy navigation. The left pane will also have a logout option at the bottom.

* Login Page



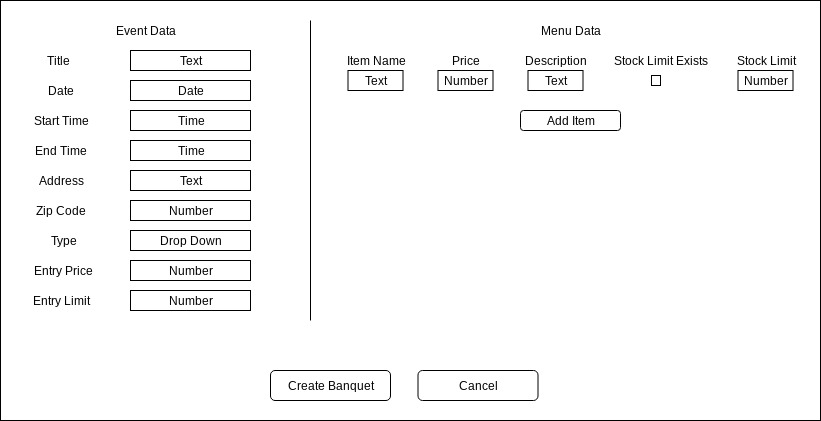
* Profile Page



* My Banquets page



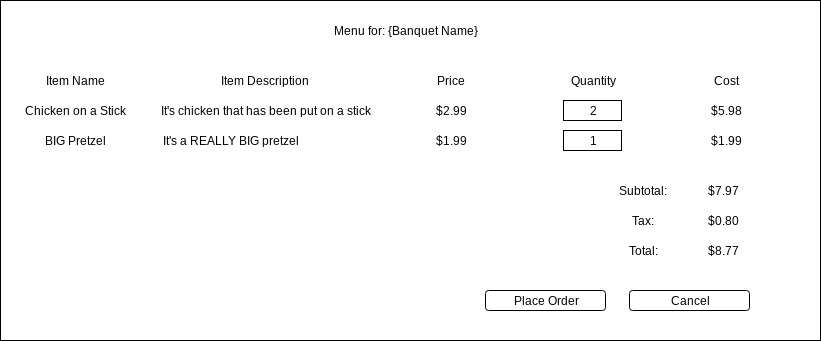
* New Banquet page



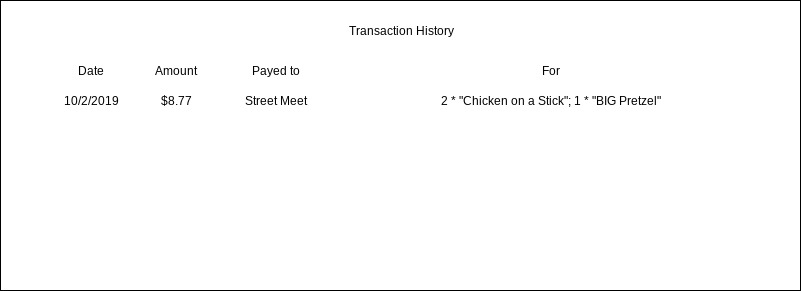
* Banquets in my area page



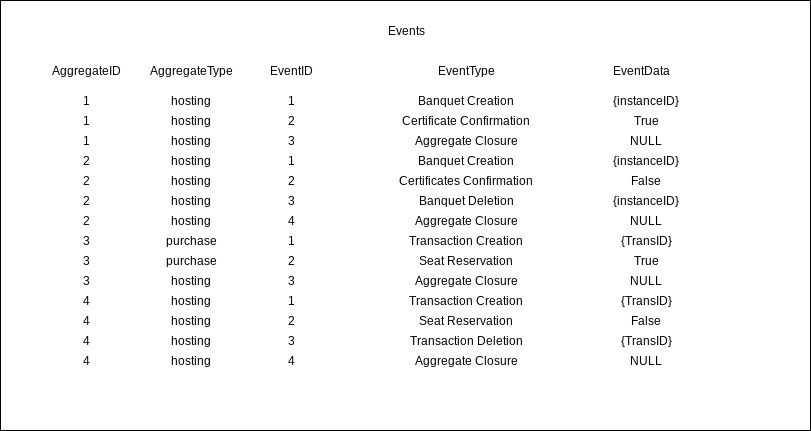
* Tray Page



* Transaction History page



* Events page



Use restful web services

Use react to reach out to restful web services