

How I Approached The Problem Statement.

Feature/Functionality List

- Landing Page
- Authentication System for both Customer and Restaurant (Login, Signup).
- Restaurant Register/Signup
 - Restaurant Name
 - Address of Restaurant
 - Opening & Closing Time
- Customer Register/Signup
 - Name
 - Email
 - Mobile Number
 - Password
- List of All Restaurants
- Restaurant Login
 - Restaurant Admin can perform the following operations for the food items:
 - Add food items
 - Item name, quantity, price, description, picture/placeholder for the item.
 - Update food Items
 - Item name, quantity, price, description, picture/placeholder for the item.
 - Delete food items
 - Change the Availability/Stock of food.
 - Accept the Order
- Customer Login
 - See full Food items list.
 - Select Food item, quantity and add to cart.
 - Checkout
 - Confirm and Place Order.
 - Past Orders
- Make a proper Database Schema design.

My Approach

First i read the above statement carefully twice, after that i started working on the project as follows:

1. Gathered Resources (images, font).

2. For Design Inspiration (checked few websites and templates).

Once, my mind was ready to act, i setted folder and started vscode.

3. I went throught the problem statement and covered the first feature **Landing Page**.

4. Likewise, i went through 2nd listed feature **Authentication System for both Customer and Restaurant (Login, Signup)**.

After reading that i realized i need to work on **backend**, so i setted everything up and created the endpoint and logic of **Login & Signup** and along the way i created other basic logic as well such as **add, delete, update, delete, resetpassword etc**.

5. To make the **Endpoints** work i created **database scheme** and connected my api with local mongodb collection.

6. After testing all the endpoint through **thunderclient** i went back to **frontend** and designed **Login and Signup** for **Customer and Restaurant**.

7. Next i worked with **Restaurant dashboard** so, i could **add food** and have data to be displayed in **menu page** of restaurant.

8. Along the way, i made next section on landing page which was **list of all restaurants** as in card format.

9. Next, i worked with **Menu page** where **all food items** of particular restaurant were **displayed**.

10. Since so far all was good i went back to **Restaurant Dashboard** to **create rest** of the **features** like **edit, delete Food items** and even **display it** there.

11. I even made sure that **Restaurant** could **change** the **Availability/Stock of food** as mentioned on the problem statement.

Real Challenge

For me the real deal was to **create a way for customer to add product into cart** with **desired number of food quantity**, since i did not knew any state management, **to complete** this feature i **spended some time** on **researching & learning Redux Toolkit**.

12. Finally i was **able to cover** the **feature to select food items, quantity and add to cart**.

13. **Features remaind** were **accepting orders by restaurant** and **checkout for Customer**.

14. Accepting orders was easily done and for **checkout** i **did some documentation reading** of **Stripe Payment** and **watched some tutorial** and **completed it** (faced various issues but managed it somehow).

15. **Final step** was, i **pushed the project to github**, then after **hosted it** on **render.com**, **worked on mobile responsiveness** and **created some documentation, screenshot pdf** along with **small working video of the project**.