Screening Assignment

Understanding of Problem statement:

We need to design an application for a restaurant from where the user can order food items.

Breaking down the Problem Statement:

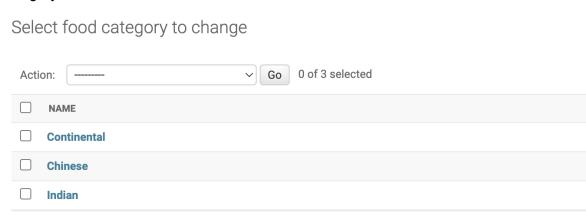
- 1. The user can choose from variety of Food Options(Chinese, Indian, Continental).
- 2. Corresponding Menu Items will be shown to user.
- 3. Now the user can see all those food Items, can filter those based on whether veg or non veg
- 4. User can select multiple items and place a order.
- 5. This order is then goes to the Kitchen (dashboard), The chef or manager can see the order, and assign a waiter to it, based on the waiter availability.
- 6. A preparation time is added in the order.
- 7. The order can be rejected if their are not enough ingredients or if chef doesnt want to entertain the order.
- 8. When a user places a order, its name, mobile number, table number and mode of payment are getting filled by the UI.
- 9. A database of waiter(staff user) is maintained in order to keep track of their availability and the track of which waiter is handling which order.
- 10. Once the order is placed, its get completed then can be rejected by the customer if something in food or it is raw or spoiled.

There would be three sections for it.

- 1. The user Section(As of now we are keeping this open ended, and just like walk in, you can open our webapp and can place order, no need to signup or login).
- 2. The staff user Section(A record of waiters)
- 3. The food items(A record of all the food items and their receipe and description)

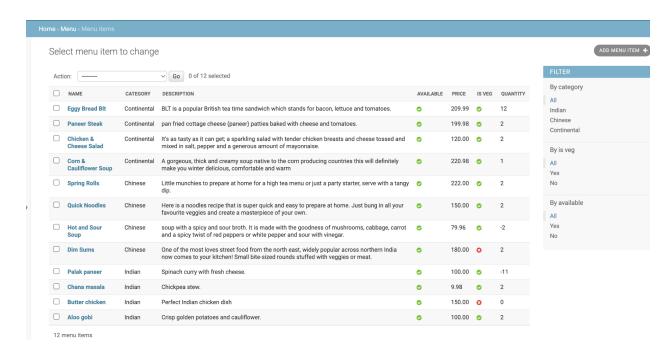
Models:

FoodCategory

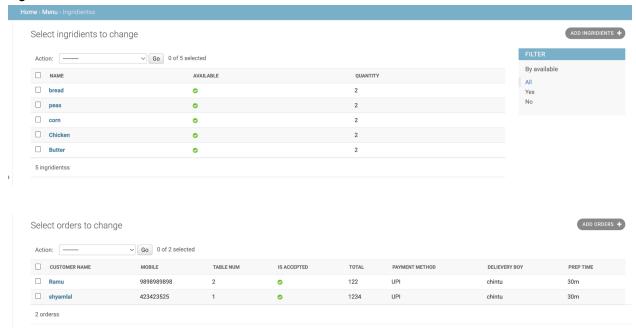


Menultem

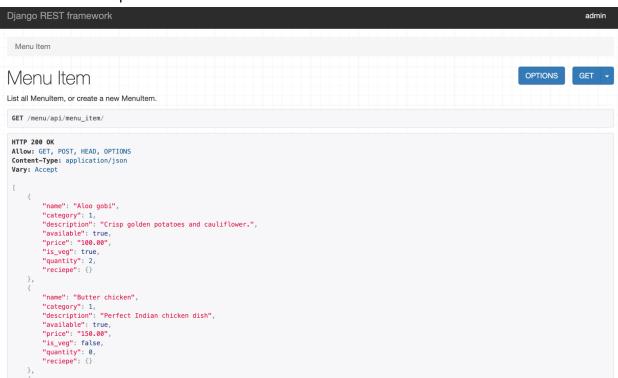
3 food categorys



Ingridients



MenuItem API Response



Ingredients API Response

```
Ingridients

Ingridients

Ingridients

Ingridients

Ingridients

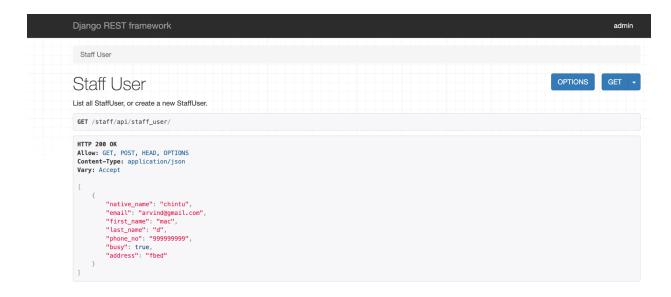
OPTIONS GET 

HTTP 200 OK
Allow: GET, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "name": "Butter",
    "description": "This is butter",
    "available": true,
    "quantity": 2
    },
    {
    "name": "Chicken",
    "description": "this is chicken",
    "available": true,
    "quantity": 2
    },
    {
    "name": "corn",
    "description": "corn",
    "available": true,
    "quantity": 2
    },
    {
    "name": "corn",
    "description": "corn",
    "description": "true,
    "quantity": 2
    },
    {
    "name": "corn",
    "description": "corn",
    "description": "corn",
    "available": true,
    "quantity": 2
    },
},
```

Order API Response

Staff User API Response



All these API's have been made with the Rest Framewrok and can be integrated with any frontend Framework, independent on any language.

How are we maintaining the Inventory for MenuItems availability from ingredients???

To tackle this problem I have used an attribute as recipe in the MenuItems, it's keys are the Ingridients and we can easily update the Inventory or if we are able to process the order using it.

How are we managing the right delievery???

Each order is associated with the customer name and table number, and a waiter, this will ensure the correct delivery of the order to right person.