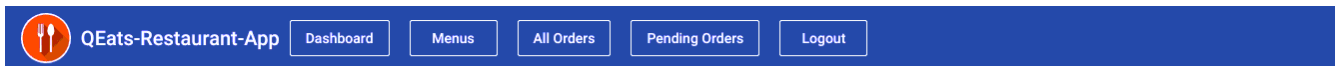


# QEats Restaurant App

## Prototype of a restaurant app of online food ordering service

### What is it?

- QEats Restaurant App help restaurant owners and managers in managing their menu (items, availability, prices), order preparation time, order status, order history, and order notifications (like the arrival of order and/or cancellation of orders).
- This application will provide restaurant owners (or managers) a seamless and easy-to-use interface for managing their restaurant over QEats platform.
- By providing intuitive and easy access for restaurant owners (or managers) will result in an increase in the number of restaurants over QEats.



Welcome tester!

Orders  
Pending

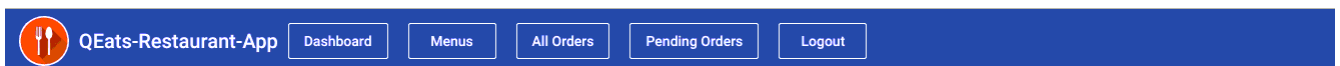
[More Details](#) →

Orders  
Pending

...

Items  
Running Out

...



Add new Item!



#2R06LKK7TM  
**Item Name:**Paneer Butter Masala  
Pizza  
**Item Price:**311  
**Attributes:** Sweet,Tasty

EDIT

SET QUANTITY

DELETE



#1B97J9LLDE  
**Item Name:**Zucchini, Corn, Spinach  
and Mushroom Pizza  
**Item Price:**348  
**Attributes:** Gravy,Tasty

EDIT

SET QUANTITY

DELETE



#BPFLMS07YW  
**Item Name:**Create Your Own Veg  
Pizza  
**Item Price:**411  
**Attributes:** Savory,Savory

EDIT

SET QUANTITY

DELETE



#MAPH89Q9G1  
**Item Name:**Just BBQ  
**Item Price:**504  
**Attributes:** Salty,Savory

EDIT

SET QUANTITY

DELETE



#T7KC5P4BBF  
**Item Name:**Minced Chicken Pizza  
**Item Price:**542  
**Attributes:** Gravy,Popular

EDIT

SET QUANTITY

DELETE



#VEYWVFM9MO  
**Item Name:**Butter Chicken Pizza  
**Item Price:**252  
**Attributes:** Salty,Tasty

EDIT

SET QUANTITY

DELETE

## All Orders

#5d09d00e7cb0318eb7d2544b

Customer Id : tester

☐ Seafood Grill Squared Burger

PACK

#5d09d0157cb0318eb7d2544c

Customer Id : tester

☐ Seafood Grill Squared Burger

PREPARE

## PENDING ORDERS

#5d09e39e7cb0318eb7d2544f

Customer Id : tester

- Seafood Grill Squared Burger

ACCEPT

REJECT

#5d09e49bbe33b92d365f44af

Customer Id : tester

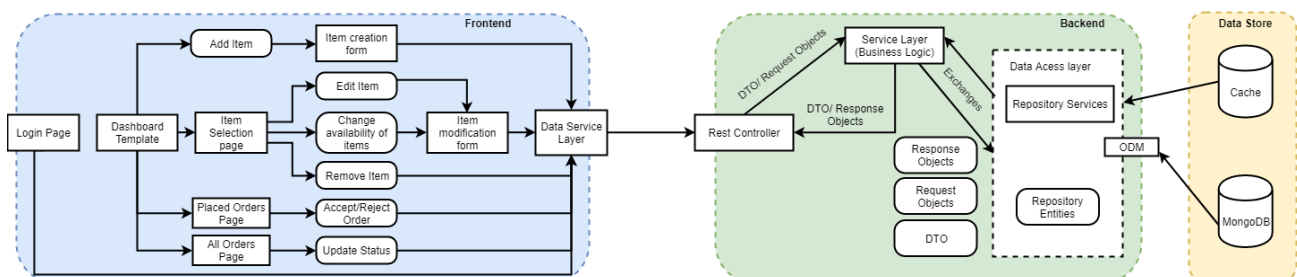
- LASSI

ACCEPT

REJECT

## Tech Stack

### Architecture:



## Development Stack:

### 1. Front-End

#### 1. Templates (Views) -

- HTML5
- CSS
- Angular-Material

#### 2. Service Layer

- Angular
- TypeScript

### 2. Back-End

#### 1. Rest Controllers, Exchanges, Services, Repository Services, Entities -

- Java
- Spring boot

### 3. Data Store

#### 1. Database -

- MongoDB

#### 2. Cache -

- Redis

### 4. Tools

- REST API
- JSON
- Postman
- JMeter
- IntelliJ

## This Version

---

Current version has these flows implemented:

- Add item to menu
- Remove item from menu
- Edit attributes of a particular item in the menu
- Check the quantity of items available
- Update the quantity of items available
- Accept/Reject incoming orders based on availability of items
- Update order status - PREPARING, PACKED, Out For Delivery (OFD)

## Testing Live Server:

The current backend is hosted on cloud VM at this address `54.158.255.17:8081`

Try sending `curl -X GET http://54.158.255.17:8081/qeats/v1/menu?restaurantId=18471268`

The current frontend is hosted at <https://q-eats-restaurant.firebaseio.com/> **Note: the frontend won't be able to send request to backend. Since browser prevents sending request from HTTPS to HTTP. Run your browser to `--allow-insecure-sources`**

## API contracts:

Following APIs are developed on top of current QEats backend:

1. Adding an item to the menu

`POST /qeats/v1/menu/item`

2. Removing an item from the menu

`DELETE /qeats/v1/menu/item`

3. Edit the item in the menu

`PUT /qeats/v1/menu/item`

4. Edit the quantity of items available

`PUT /qeats/v1/item/available`

5. Get the quantity of items available

`GET /qeats/v1/item/available?itemId=&restaurantId=`

6. Get all the placed orders

`GET /qeats/v1/orders/placed?restaurantId=`

7. Accept/Reject the placed orders

`PUT /qeats/v1/orders/placed`

8. Get all the orders (which are not yet finished and are accepted)

`GET /qeats/v1/orders?restaurantId=`

9. Update status of order

`PUT /qeats/v1/orders/placed`

10. User Login

`POST /qeats/v1/user/login`

11. Get the menu of a restaurant

`GET /qeats/v1/menu/item?restaurantId=`

The contracts and spec sheets of each endpoint can be found [here](#)

## Development Guide

---

The project is divided into two main folders namely:

- `/backend` for all the backend related code
- `/fronend` for all the frontend related code
- `/datastore` contains the database dump

## Installing dependencies:

### For backend development:

You'll need:

1. Java JDK

```
sudo apt install default-jdk
```

2. Java JRE

```
sudo apt install default-jre
```

3. Gradle

Code already has a `gradle-5.2.x` zip file

4. IntelliJ IDE

Head over: <https://www.jetbrains.com/idea/download>

### For frontend development:

You'll need:

1. NPM

```
sudo apt install npm
```

2. NodeJS v10 or later

```
sudo npm install n -g
```

 For the latest stable version: 

```
sudo n stable
```

 For the latest version: 

```
sudo n latest
```

3. Angular CLI

```
sudo npm install @angular/cli -g
```

### For datastore:

You'll need:

1. MongoDB

Install monogdb: 

```
sudo apt install mongodb
```

Run mongod: 

```
sudo service mongod start
```

Verify installation 

```
mongo --version
```

2. Redis

Install redis: 

```
sudo apt install redis-cli
```

Verify installation: 

```
redis-cli --version
```

## Running Server locally:

### Restore datastore:

1. Go to the `/datastore` directory
2. Type 

```
mongo restaurant-database --eval "db.dropDatabase()"
```

3. Then type `mongorestore --host localhost --db restaurant-database --gzip --archive=restaurant-db.gz`

## Backend Server:

From terminal:

1. Go to `/backend` folder
2. Type `./gradlew bootrun`
3. Server will start on `localhost:8081`

## Frontend Server:

From terminal:

1. Go to `/frontend` folder
2. Type `npm install`
3. Type `ng serve`
4. Go to `localhost:4200` in your favorite browser

Currently DB has data for this User's restaurant:

Username	Password	Restaurant ID
tester	Tester@123	18471268

## Contributors

1. Abdul Kadir ([@Abdulkadir98](#))
2. Chirag Mittal ([@mittalchirag](#))
3. Shashank Holla ([@shashankholla](#))

## Links

1. "Engineering Design Doc" [https://docs.google.com/document/d/1pZmBQZdeMnm6twxNAntwyEuARUKHNSYNCA0J\\_aFITzA/edit?usp=sharing](https://docs.google.com/document/d/1pZmBQZdeMnm6twxNAntwyEuARUKHNSYNCA0J_aFITzA/edit?usp=sharing)
2. "Product Requirement Doc" <https://docs.google.com/document/d/13eOloxcUo8GjkRQh-OxBezNKzWik4SnKTW8CB7gzXBQ/edit?usp=sharing>