Reservation System

Contents

[Microservices: 2](#_Toc65253548)

[1. Merchants Service: Manage merchants, each merchant has 1 admin, controll overall restaurants of merchants 2](#_Toc65253549)

[ Database Table: 2](#_Toc65253550)

[2. Auth Service: used to manage merchant’s admin, staff,… 2](#_Toc65253551)

[ Database Table: 2](#_Toc65253552)

[3. Customer Service: store and manage member of the restaurant. 2](#_Toc65253553)

[ Database Table: 2](#_Toc65253554)

[4. Restaurants Service: be used to manage restaurants, tables and staffs of the restaurant 2](#_Toc65253555)

[ Database Table: 2](#_Toc65253556)

[5. Table Service: be used to get, set table status, available or served. 3](#_Toc65253557)

[ Databasse 3](#_Toc65253558)

[6. Reservation Service: manage user’s bookings. 4](#_Toc65253559)

[7. Checkin Service: be used to track customer visit time. 4](#_Toc65253560)

[ Database: 4](#_Toc65253561)

[8. Checkout Service: be used to track when the customer leaves restaurant. 4](#_Toc65253562)

[ Database: 4](#_Toc65253563)

[Operation Flow 5](#_Toc65253564)

[Customer Login: 5](#_Toc65253565)

[Staff Login: 5](#_Toc65253566)

[Customer select the restaurant: 6](#_Toc65253567)

[Table status change: 6](#_Toc65253568)

# Microservices:

## Merchants Service: Manage merchants, each merchant has 1 admin, controll overall restaurants of merchants

### Database Table:

#### **Merchants**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| id | Uuid |  |
| companyName | Varchar2(200) |  |
| logo | Varchar2(40) | Name of the image |
| adminId | Uuid | UserId, with admin role |

## Auth Service: used to manage merchant’s admin, staff,…

### Database Table:

#### **Users**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| id | Uuid |  |
| firstName | Varchar2(40) |  |
| lastName | Varchar2(40) |  |
| userName | Varchar2(20) |  |
| encryptedPassword | Varchar2(40) |  |
| Email | Varchar2(40) |  |
| role | Jsonb |  |

## Customer Service: store and manage member of the restaurant.

### Database Table:

#### **Customers**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| Id | Uuid |  |
| firstName | Varchar(40) |  |
| lastName | Varchar(40) |  |
| email | Varchar(100) |  |
| encryptedPassword | Varchar(40) |  |

## Restaurants Service: be used to manage restaurants, tables and staffs of the restaurant

### Database Table:

#### **Restaurants**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| id | Uuid |  |
| name | Varchar2(40) |  |
| logo | Varchar2(40) | Image path |
| merchantId | uuid |  |

#### **Tables**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| id | Uuid |  |
| displayedName | Varchar2(40) | Name of restaurant |
| minSit | Integer | Minimum sit required when booking |
| maxSit | Integer | Maximum sit required when booking |
| Location | Uuid | Location of the table, restaurant staff should implement this option |
| restaurantId | Uuid | Where the table belong to |

#### **Staffs**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| id | Uuid |  |
| Name | Varchar2(40) | Nick name or fullname,… |
| Position | Varchar2(40) |  |
| merchantId | Uuid |  |
| restaurantId | Uuid |  |
| userId | Uuid | Information for staff to login into the app |

#### **Locations**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| Id | Uuid |  |
| Name | Varchar2(40) |  |
| Image1 | Varchar2(40) |  |
| Image2 | Varchar2(40) |  |
| Image3 | Varchar2(40) |  |
| Image4 | Varchar2(40) |  |
| createdBy | Uuid | Staff who created this info |

## Table Service: be used to get, set table status, available or served.

### Databasse

#### **TableStatus**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| Id | Uuid |  |
| tableId | Uuid |  |
| CustomerId | Uuid | Who books this table |
| RestaurantId | Uuid |  |
| bookingId | Uuid |  |
| checkinId | Uuid |  |
| numberOfSit | Integer |  |
| Status | Boolean | True = served, False = available |

## Reservation Service: manage user’s bookings.

#### Database:

#### **Bookings**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| Id | Uuid |  |
| customerId | Uuid | Who make a booking for the table |
| merchantId | Uuid |  |
| restaurantId | Uuid |  |
| tableId | Uuid |  |
| numberOfSits | Integer |  |
| estVisitTime | Datetime | The time that the customer will come |
| bookingCode | Varchar2(4) | Customer keep this code to work with staff when come to the restaurant |

## Checkin Service: be used to track customer visit time.

### Database:

#### **CheckIn**

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| Id | Uuid |  |
| bookingId | Uuid |  |
| customerId | Uuid | Who make a booking for the table |
| merchantId | Uuid |  |
| restaurantId | Uuid |  |
| tableId | Uuid |  |
| numberOfSits | Integer |  |
| checkInTime | Datetime | The moment that the customer checks in |

## Checkout Service: be used to track when the customer leaves restaurant.

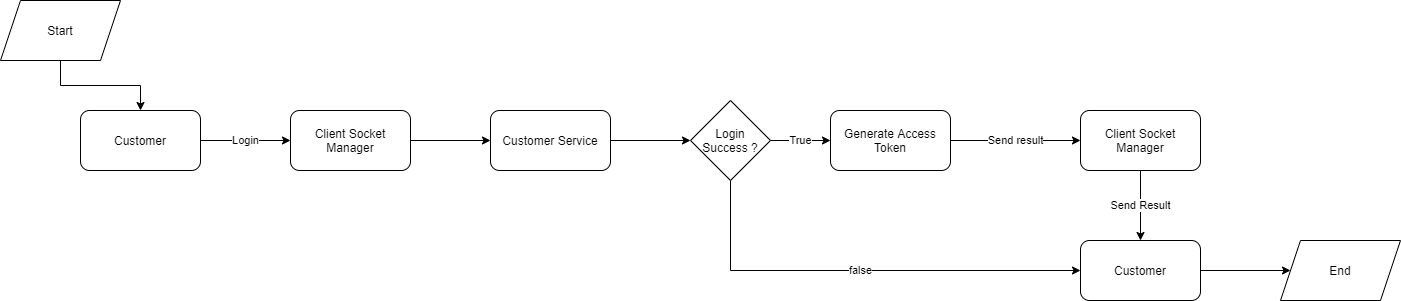
### Database:

#### CheckOut

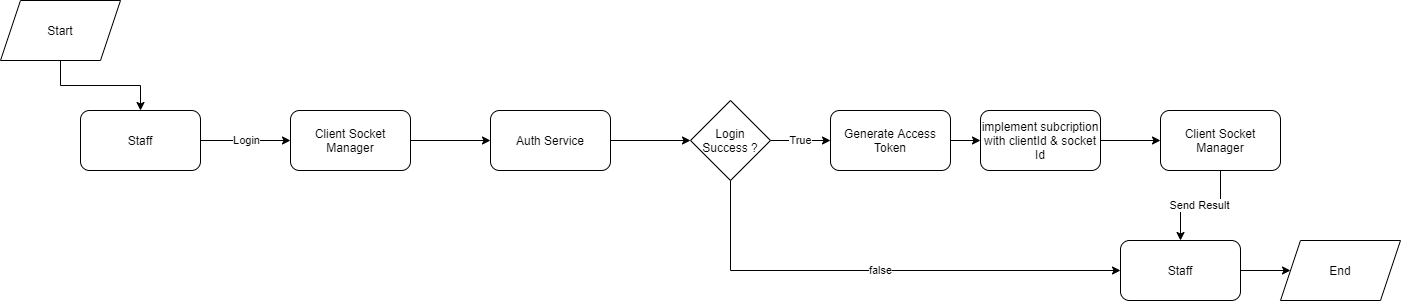
|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| Id | Uuid |  |
| bookingId | Uuid |  |
| checkInId | Uuid |  |
| customerId | Uuid | Who make a booking for the table |
| merchantId | Uuid |  |
| restaurantId | Uuid |  |
| tableId | Uuid |  |
| numberOfSits | Integer |  |
| checkOutTime | Datetime | The moment that the customer leaves restaurant |

# Operation Flow

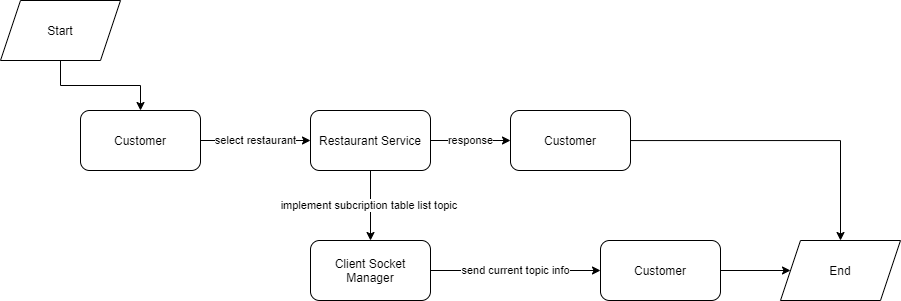
## Customer Login:



## Staff Login:



## Customer select the restaurant:



## Table status change:

* Staff changes the table status
* Customer books the table.

