



# Tecnológico de Monterrey

Instituto Tecnológico y de Estudios Superiores de Monterrey - Campus Santa Fe

Bases de Datos Avanzadas

Ariel Lucien

Adrian Biller - A01018940

Edgar Garcia - A01021730

Enrique Tayabas - A01019108

**Descripción del Proyecto y Manual de usuario**

10 de mayo del 2018

# Manual de usuario MiComidita

1. Antes que nada para correr la aplicación antes que nada hay que tener instalado react-native Android Studio o Xcode y node.js, una vez que se tenga instalado estos componentes.
2. Una vez que estén instalados los componentes lo siguiente que se hará es acceder al siguiente link para descargar la aplicación.
  - a. GitHub de la aplicación: <https://github.com/Biller17/RestaurantApp>
  - b. Api Base de datos: <https://github.com/henrykalex/RestaurantAPI>

## Instrucciones para correr la aplicación

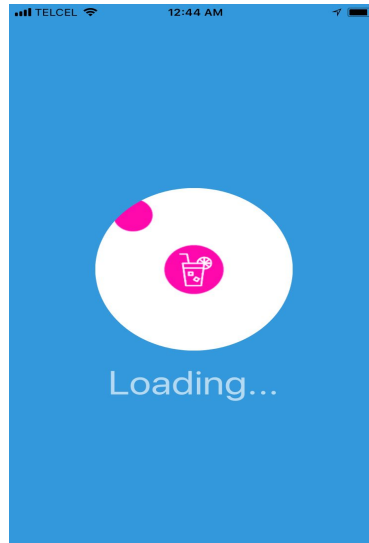
3. Una vez instalados los componentes mencionados en el punto 1. Lo siguiente que se hará es abrir la terminal y ingresar un sencillo comando.

```
Edgars-MacBook-Pro% npm i
```

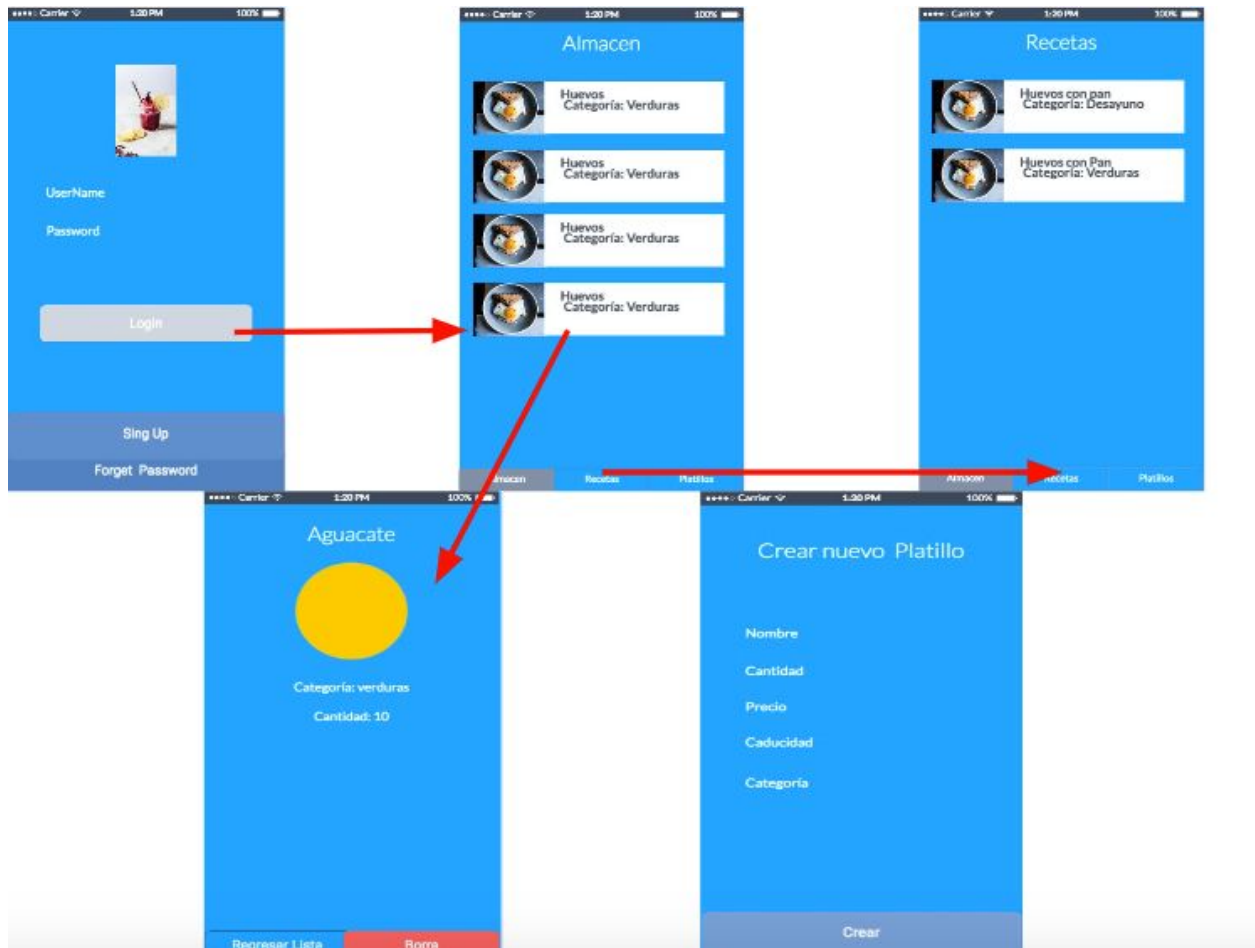
4. El cual instalará los componentes necesarios (paquetes) para correrlo.
5. Una vez instalados los componentes lo siguiente, es correr la aplicación con este sencillo comando.

```
Edgars-MacBook-Pro% react-native run-ios
```

- a. El comando anterior puede variar dependiendo de los instalado Xcode (ios) o Android Studio(Android)
6. Una vez puesto el comando pasado la aplicación se ejecutará y desplegará un dispositivo virtual donde se ejecutará la aplicación.
  7. Si todo está perfectamente instalado se ejecutará la aplicación de manera correcta y aparecera lo siguiente:



8. Algunas de las pantallas son las siguientes:



**Instrucciones para correr dockerfile**

```
$ docker-compose up
6
$ sudo docker-compose up
```

# MiComidita

**MiComidita**, es un aplicación móvil hecha en react native.El restaurante miComidita S.A. cuenta con 2 almacenes:

- Materias primas
- Productos Elaborados

Donde dicha base de datos será manejada mediante neo4.js que es un software orientados a grafos implementados en java.

## Objetivo Principal

EL objetivo principal de este aplicación es crear una aplicación que se encargue de verificar los productos, así como su fecha de caducidad entre otros aspectos que detallaremos más adelante, ya que con ello podemos crear platillos (agregarlos) ver el inventarios de ellos y si es que se pueden realizar dichos platillos.

## Descripción técnica de producto

El restaurante o sistema se basará en los siguiente

### CRUD de Materias Primas

- Productos Frescos
- Carne
- Lácteos
- Alimentos congelados
- Pan, cereal, arroz y pastas
- Básicos y misceláneos: hierbas, especias, azúcar, harina, polvos para hornear, etc.
- Otros: todo lo que no puedas catalogar en lo anterior

### CRUD de recetas

- Productos y cantidades necesarias para la receta. Ejemplo

- Salsa barbacoa (400 ml)
  - 1 cebolla
  - 2 dientes de ajo
  - 400 ml de tomate triturado
  - 90 ml de vinagre
  - 2 cucharadas de miel
  - 2 cucharadas de salsa de mostaza
  - Pizca Sal
  - Pizca de pimienta blanca molida
  - Aceite de oliva
- Costo de la receta (calculado a partir de los productos y cantidades que contiene)

#### CRUD de productos elaborados

- Sólo se produce a partir de recetas disponibles
- Sólo se produce si hay materia prima disponible
- El inventario de materia prima disponible se reduce según la receta

#### Consultas:

- Materia prima disponible (por producto).
- Materia prima y productos elaborados próximos a caducar
- Costo de pérdidas por productos que han caducado
- Productos elaborados disponibles (por producto elaborado)
- Costo total (e individual) de ambos almacenes

#### Restricciones

- Utilizar una base de datos orientada a grafos
- Elegir entre las siguientes plataformas para el front end: web, escritorio, móvil, etc.
- Al menos 20 recetas
- Es necesario considerar la caducidad de los productos (los productos que han caducado se eliminan automáticamente del almacén (cualquiera de los dos))

#### Contenido de la plataforma

- La aplicación contará con un login con autenticación básica. El cual permitirá hacer registro y inicio de sesión.
- Una vez que se acceda aparecerán 3 secciones, en las cuales aparecerá la parte de inventario de productos.

- ◆ Una vez que seleccione el producto desplegará una pantalla de información la cual dirá aspectos generales ella así como la fecha, cantidad, entre otros aspectos.
- En la segunda pestaña vendrá un agregado de receta, donde se pasara la misma acción que la pasada, con el agregado de cardview.
- En la tercera sección está el almacén donde se verá la información de los productos y las cardview para agregar o pedir inventario al almacén

### **Utilidad**

La aplicación mi MiComidita está diseñada preferentemente a personas que manejen un restaurante con el cual puedan hacer un pedido de platillo e ingredientes, para surtir su restaurante. Se tiene un completo control de inventarios y te dice los productos disponibles que se pueden tener y que platillos se pueden, realizar

### **Alcance**

- Impulsar una aplicación que permita el control de comida
- Creación de productos para verificar cuánto inventario se tiene en el restaurante
- Control en aspectos de inventario para surtir productos

### **Que se logró y que se agregara**

En concreto se puede decir, que a lo largo del curso se pudieron aplicar conocimientos de javascript y react-native aunque en la clase vimos react. Sin embargo en lo personal a lo largo del curso lo importante es aplicar react, donde pudimos realizar una app desde cero, con una base de datos orientada a grafos.

### **Logros**

- Se logró hacer un login con registro.
- Agregación de productos.
- Verificación de inventario.

- Creación de Recetas
- Verificación de campos
- Diseños básico y algunas animaciones
- Conexión a una base de datos orientada a grafos
- Acceso a la plataforma en el caso de que no este registrado
- Responsividad para varios dispositivos móviles (Iphone).

## Que falto

- ❖ Está en proceso el envío de correos ya que para react (web) es mucho más sencillo la implementación de correos.
- ❖ Cargar en la base de datos las imágenes con lo url de cada producto.