



OrderIt!

Problema.

Todos los restaurantes cuentan con un menu donde podemos ver los platillos y bebidas que estos ofrecen. Menús que cuentan con un mismo formato, es decir, en el encuentras el nombre del platillo y su precio, algunas veces unos te ofrecen una imagen o foto alusiva al producto, otras, alguna descripción.

Como **cliente**, quisiera saber que ingredientes contiene un platillo o ver la forma de este por medio de una foto/imagen.

Como **dueño** de un restaurante, quiero que mis clientes tengan mayor información de los platillos que les ofrezco. De igual manera, saber cuales son los platillos que mas les gustan y así tomar decisiones en beneficio de mi restaurante.



Solución.

Crear una WebApp que muestre el menú del restaurante que el cliente esta visitando, este menú contendrá una amplia información acerca de los platillos y bebidas que el restaurante ofrece, así como promociones y otros anuncios.

El cliente podrá generar ordenes, calificar el platillo ordenado y solicitar su cuenta.

El dueño/gerente podrá notar un incremento significativo en las ventas y de igual manera podrá tomar decisiones con base a la tendencia (calificación) de los platillos.

Ejecución.

Las herramientas ha utilizar para esta primera fase del desarrollo de este proyecto serán: **JavaScript**, **Python**, **Django**, **MariaDB** y **Docker**.



Dificultades.

La principal dificultad que existe en el desarrollo de este proyecto es la poca expertis que se tiene en algunas de las herramientas que se planean utilizar pero que conforme pase el tiempo se irán dominando.

Otra dificultad es el poco tiempo que se tiene para cumplir con la fecha de finalización del proyecto.



Conclusiones

OrderIt! busca innovar la forma en la que los clientes interactúan con los menús de comida, ofreciendo amplia información de los platillos y basar lo que quieren ordenar mas allá del nombre y el precio de un platillo. Además, busca ser una herramienta de ventas para el restaurante aumentado de manera significativa sus ventas diarias.



Mejora.

Se contempla el realizar la aplicación de manera nativa en React Native y cambiar el modelo de la base de datos, pasando así de un modelo relacional a uno no relacional (MongoDB).

Otra mejora que se busca hacer es la de añadir mas funciones a la aplicación, por ejemplo, corte de caja y la generación de pequeños reports de venta.