

IS-410, Programación Orientada a Objetos Primer Periodo 2022

Informe Proyecto de Clase-Entrega Final

Elaborado por:

|  |  |
| --- | --- |
| **20171031672** | **Zeyris Lisbeth Maradiaga Cerrato** |

Catedrático: Erick Vladimir Reyes Marín

# Índice

[Índice 2](#_bookmark0)

[Introducción 3](#_bookmark1)

[Título y descripción detallada del proyecto 4](#_bookmark2)

[Definición de marca, logotipo y paleta de colores 5](#_bookmark3)

[Listado de módulos o formularios a desarrollar 7](#_bookmark4)

[Definición de tipos de usuario y niveles de acceso 12](#_bookmark5)

[Distribución de tareas 13](#_bookmark6)

[Repositorios de Github 14](#_bookmark7)

[Glosario 15](#_bookmark8)

[Landing Page 18](#_bookmark9)

[Aplicación para cliente 19](#_bookmark10)

[Aplicación para Repartidores 22](#_bookmark11)

[Aplicación de administradores 26](#_bookmark12)

[.................................................................................................................................................26](#_bookmark12)

[Mockups en figma 35](#_bookmark13)

[Video explicativo de la funcionalidad del proyecto en base a los mockups 35](#_bookmark14)

# Introducción

A continuación, presento el informe o documentación de mi proyecto de la clase de Programación Orientada a Objetos (IS-410), en este avance de mi informe, tocare los puntos de definición de detallada de mi proyecto, el rubro y tema asignado, definiré la marca, logotipo y paleta de colores de mi empresa y también enlistare los módulos o formularios a desarrollar, definiré los tipos de usuario y niveles de acceso de cada página.

Estaré llevando a cabo este proyecto de manera individual, por lo que la distribución de tareas en trello, Jira o similares, no será necesaria. También, realizare un glosario de al menos 10 términosrelacionados con el rubro de mi trabajo realizado hasta este punto. Luego, capturas de los mockups de mis páginas, landing page, aplicación de clientes, aplicación de motoristas, aplicación de administradores y como punto final, adjuntare.

En mi segundo avance de proyecto, estaré presentando tres (3) repositorios en gitHub, los cuales pertenecen a 3 proyectos, un (1) proyecto para Clientes, un (1) proyecto para Motoristas y un (1) proyecto para Administradores.

Dichos proyectos contienen la estructura HTML, CSS y JS que se usara desde ahora en adelante, contiene todas las vistas UI hechas en los mockups de figma y también contiene la lógica de JavaScript para poder navegar hacia la página.

También en este segundo avance, estaré presentando, 10 nuevos términos en mi vocabulario relacionados con el rubro de la programación vista o usada en esta entrega, también presentare fotos de los diseños vistos desde el navegador Google Chrome.

Cabe de destacar que los mockups fueron hechos de la siguiente manera:

* Página web Clientes: **Diseño para móviles, especializado en iPhone SE**
* Página web Motoristas: **Diseño para móviles, especializado en iPhone SE**
* Página web Administradores: **Diseño para escritorio, especializado en 1024x768**

Es muy importante que se tome en cuenta que estas fueron las dimensiones elegidas para la exactitud de los mockups en figma.

# Título y descripción detallada del proyecto

El rubro y tema asignado para este proyecto es, el desarrollo web, tanto como la parte del cliente (FrontEnd) como la parte del servidor (BackEnd), de una empresa de entregas de pedidos, la empresa está abarcando un rubro de deliveries y está operando a nivel nacional en la republica de Honduras.

El proyecto va consistir en el desarrollo de 3 páginas web en total.

La primera va ser la página destinada para los clientes, esta página será en la se definirán las categorías de productos en ventas, las empresas que venden sus productos y los productos en venta. El flujograma será simple, va consistir en una landing page, la cual al dar click en el go to action, se pasara a una página de registro de usuario, luego al entrar, se encontrara con una sección de categorías de compra, en donde el usuario va a elegir que desea comprar dependiendo en que categoría se encuentre, luego pasara a visualizar a todas las empresas / restaurantes / comerciantes, que están disponibles, al dar click a uno de estos, automáticamente pasara a una vista en donde el usuario podrá visualizar todos los productos de esta empresa / restaurante.

Luego, el usuario podrá hacer la compra de estos productos, y al continuar deberá de elegir su ubicación y pagar con tarjeta de crédito, hasta finalizar la compra. Dicha compra se hace por medio de un carrito de compras, el cual almacena todos los productos que el usuario selecciona para comprar.

La segunda página consiste en la página para repartidores, en donde los repartidores hacen login, y podrán ver detalladamente una lista o tabla de ordenes disponibles, los cuales ellos pueden entregar, al darle click a uno, podrán ver los detalles de esta orden y elegir si tomarla, o no. También tienen acceso a una sección de “Ordenes Tomadas” en donde pueden ver todas las ordenes que tienen en estado de “pendiente” o de “tomadas” las cuales aún no han sido entregadas, además de eso, el usuario puede cambiar el estado de todas las ordenes, como “Entregada”. El repartidor también tiene acceso a una sección de órdenes entregadas.

La tercera página consiste en la página para Administradores, los administradores no podrán hacer una cuenta, estos deben de ser agregados directamente en la base de datos.

Los administradores tienen la habilidad de agregar y eliminar empresas, productos, repartidores y órdenes. También tienen la habilidad de poner el estatus de una orden y también de asignarla a un repartidor.

# Definición de marca, logotipo y paleta de colores

Mi logotipo aún no está definido por lo que en la entrega final se va a mostrar.

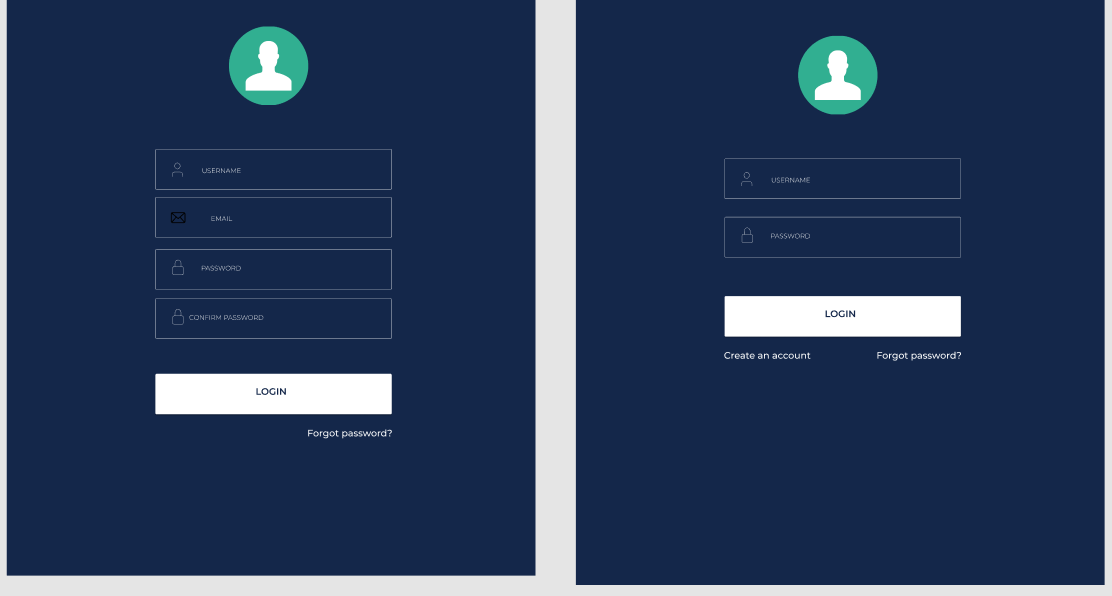
Paleta de Colores:



# Listado de módulos o formularios a desarrollar

1. Módulo de Registro

Este módulo de registro será para los clientes y los repartidores, es básicamente la forma en la que se crean las cuentas, ya sea para clientes o para repartidores.



1. Módulo de Pago

Este módulo será usado en la página de los clientes, sirve para obtener la información de pago del usuario.



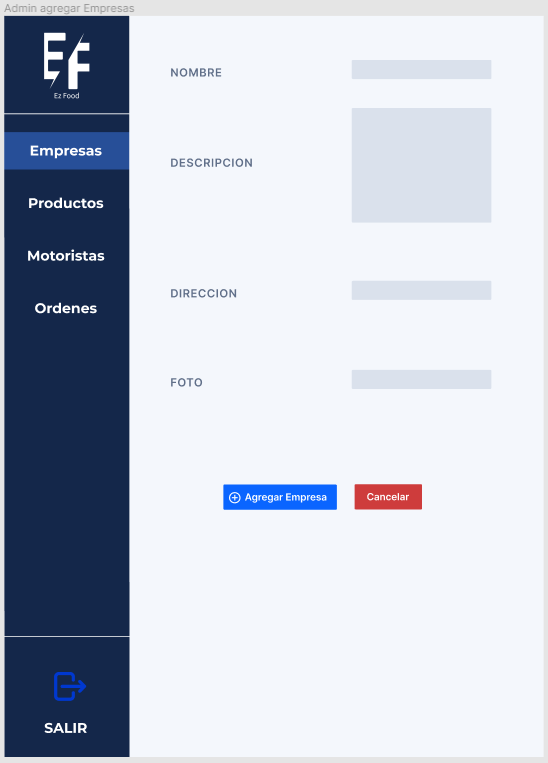
1. Módulo de ubicación

Este módulo es usado en la página de clientes, para obtener la información de la ubicación de destino de la orden.

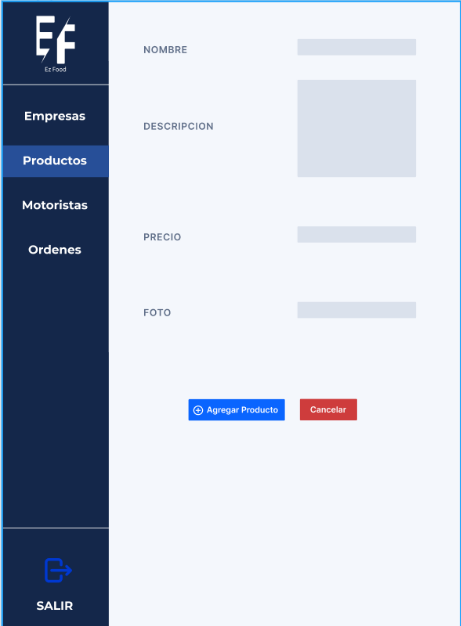


1. Modulo para agregar empresas

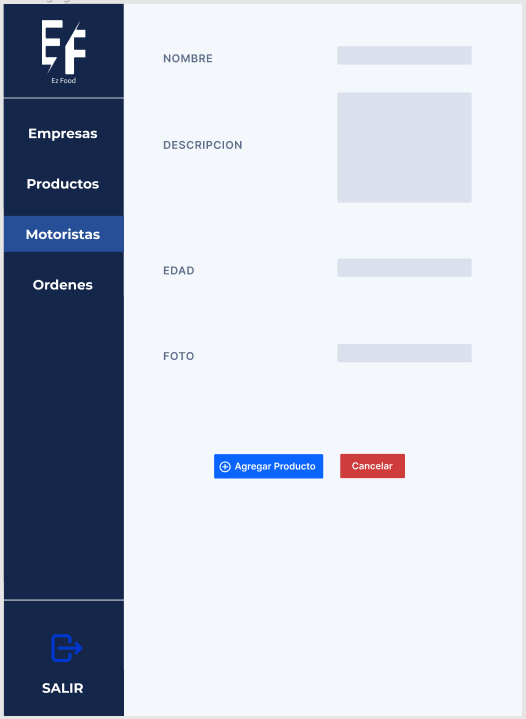
Este módulo es usado en la página administrativa para agregar empresas a la página de clientes.



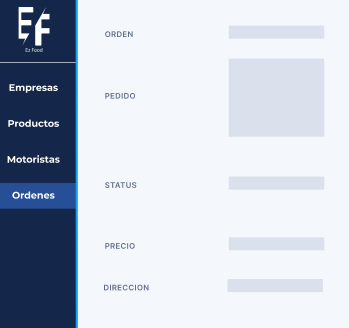
1. Modulo para agregar productos: Este módulo es usado en la página administrativa para agregar productos a la página de clientes.



1. Modulo para agregar Motoristas: Este módulo es utilizado en la página de administradores para agregar motoristas en la página para repartidores.



1. Modulo para asignar órdenes: Este módulo es utilizado en la página de administradores para modificar y agregar órdenes.



# Definición de tipos de usuario y niveles de acceso

Tipos de usuario

Los tipos de usuario serán 3:

* + Clientes: Este tipo de usuario, podrá acceder únicamente a la página para clientes, en la que se podrá crear una cuenta y registrar y únicamente realizar la compra de sus pedidos.
  + Repartidores: Este tipo de usuario, podrá acceder únicamente a la página para repartidores, en la que se podrá crear una cuenta y registrar, luego podrá observar las ordenes disponibles para hacer entregas, no puede modificar dichas órdenes.

Administrador: Este tipo de usuario, podrá acceder únicamente a la página de administradores, No puede crear una cuenta, debe de ser agregado en la base de datos, al loguearse podrá agregar y eliminar empras, productos, repartidores, ordenes.

Niveles de acceso

Los accesos se pueden representar básicamente en una jerarquía o flujo de programa.

1. Nivel 1: En este nivel, el acceso es únicamente en clientes, se producen las órdenes de entrega.
2. Nivel 2: En este nivel, el acceso es únicamente en repartidores, el repartidor crea una cuenta, y espera a la confirmación de los administradores, una vez activo, puede hacer login y ver las órdenes creadas en el acceso de Nivel 1.
3. Nivel 3: Este es el nivel con jerarquía más alta, los administradores tienen el rol más importante, ya que tienen acceso tanto al nivel 1 como al nivel 2, el nivel 3 puede manipular (agregar, eliminar y editar) el nivel 1 y 2, como agregando/eliminando empresas o agregando/eliminando repartidores.

# Distribución de tareas

Este proyecto está hecho de manera individual, por tanto, no se requiere realizar una distribución de tareas en Jira, Trello o similares.

# Repositorios de Github

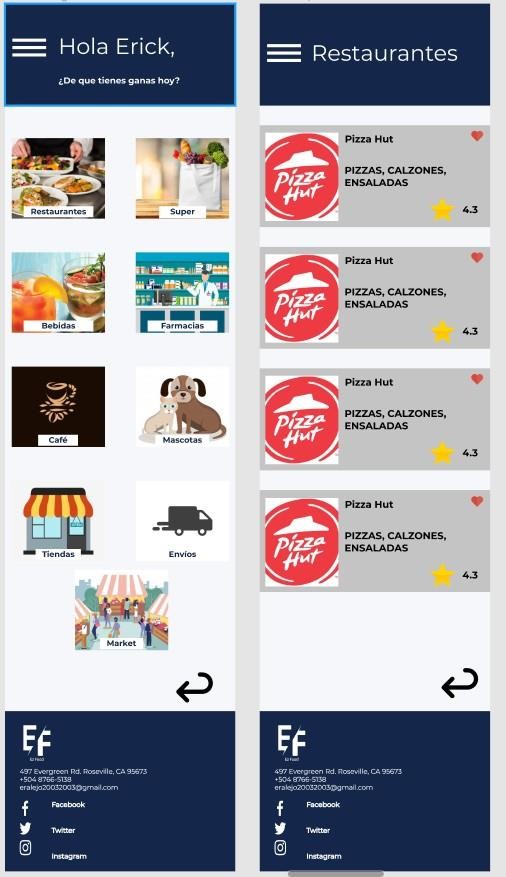
A continuación, por ahora no presentare los links de los repositorios en Github de mis proyectos ya que no he podido hacer una cuenta en github pero en la entrega final los adjuntaré todos y cada uno.

# Glosario

1. Mockup: Es un diseño de una página, modal o vista de una aplicación, pagina web, revista, libro, etc. Con el fin de representar un prototipo visual, el cual sirve como ejemplo de guía para el desarrollo de una aplicación.
2. Diseño gráfico: Es un arte, disciplina, pasión y profesión de expresar ideas por medio de arte visual, el diseño grafico combina el orden, la simetría y la combinación de colores para crear un ambiente de confort al usuario final.
3. Interfaz de usuario: Es el medio con el que el usuario se comunica con la aplicación.
4. Experiencia de usuario: Es el conjunto de conocimientos y técnicas que se aplican al interfaz de usuario con el fin de mejorarla y tener reseñas positivas de manera que todos los elementos en pantalla cumplan con una función específica y puntual, y sin desaprovechar espacio en pantalla ni sobrecargar espacio.
5. FrontEnd: Es una parte del desarrollo web, que hace referencia a la estructura visual con la interactúa el cliente final, contiene la estructura básica de etiquetas, los colores, imágenes y funcionalidades, como ser los botones, colores, fondos, logos, etc.
6. Pixeles: Unidad básica de una imagen digitalizada en pantalla a base de puntos de color o en escala de grises. Se usa como una medida estándar para el diseño gráfico y también para el desarrollo del frontEnd, suelen haber estándares de medida en pixeles para los desarrollos en figma.
7. Land Page: Es la página principal y suele ser la más llamativa, su objetivo es mostrar los elementos principales de la página web e incentivar al usuario a realizar una acción como una compra, llenar formularios, dar sugerencias, dar reseñas, etc.
8. Style Guide: Esta guía de estilos es la que contiene todos los estándares de diseño que se va usar en el desarrollo frontEnd, puede contener la paleta de colores, las dimensiones exactas de los componentes, el tamaño, negrita, cursiva, tipo de letra de los headings o texto y también puede contener componentes básicos como botones, text fields, cards, etc.
9. diseño responsivo: Este diseño hace referencia al diseño de una página web o una aplicación de escritorio, que se adecua a todos los tamaños del dispositivo en el que se encuentre.
10. Equilibrio visual: Es la técnica para repartir todo el peso visual en un determinado espacio, para conseguir un diseño simétrico lo que se debe de hacer es repartir todos los elementos en el viewport, de manera que no esté sobrecargado y tampoco mu vacío, además de tener los elementos necesarios en el momento necesario.
11. Pseudocódigo: es una forma de expresar los distintos pasos que va a realizar un programa, de la forma más parecida a un lenguaje de programación.
12. Algoritmo: es un conjunto de reglas definidas que permite solucionar un problema, de una determinada manera, mediante operaciones sistemáticas y finitas.
13. Programación: es el proceso de crear un conjunto de instrucciones que le dicen a una computadora como realizar algún tipo de tarea.
14. JavaScript: es un lenguaje de programación orientado al desarrollo web, el cual, a partir de la API de los navegadores como Google Chrome, accede a los elementos que el API le provee, los cuales denominaremos elementos DOM, y mediante la manipulación de estos elementos, se le agrega dinamismo a la página web.
15. HTML: es un lenguaje de programación de híper textos y etiquetas, orientado al desarrollo web, cuyo objetivo es el de definir estructuras básicas de las páginas web.
16. CSS: es una herramienta de lenguajes de programación de etiquetas, orientado al desarrollo web, cuyo objetivo es darle estilo, colores y propiedades de posicionamiento a las estructuras HTML de las páginas web.
17. Librería externa: Podemos decir que una librería externa es un pedazo de código a parte del nuestro, el cual, nosotros implementamos a nuestro proyecto con el fin de usar estas nuevas funcionalidades ya implementadas por alguien más.
18. Bootstrap: es una librería externa orientada al desarrollo de interfaces de usuario y experiencias de usuario, se usa mucho en el desarrollo web y esta consiste en muchas clases de CSS ya predefinidas.
19. leaflet: es una librería externa orientada al desarrollo web y de escritorio, su finalidad es proveer al usuario de un API el cual nos da varios atributos y funciones que nosotros podemos usar para poder generar un mapa y tomar la localización aproximada del usuario.
20. Test-driven development: el TDD es una técnica de programación el cual consiste en primero implementar ciertas pruebas a nuestro código, y luego realizar el mismo código, el cual primero que nada deberá de pasar las pruebas, y luego de haber pasado las pruebas, podremos re factorizar este mismo código para llegar a tener un diseño similar a los mockups.
21. Paleta de Colores: Es el conjunto de colores que forman el desarrollo de la página frontEnd, contiene sus colores primarios, secundarios, base y también sus variantes lighten.
22. Tipología Textual: Son las clasificaciones de los distintos tipos de texto que puede utilizar la página web, esta podría ser sans-serif, sans-simon, times new roman, etc.
23. CSS: Es un lenguaje de diseño grafico para definir la presentación de un contenido, define colores, posiciones, tamaños, etc.
24. RGB: Es la forma de representar los colores como sintaxis en CSS, las siglas RGB significan Reed, Green y Blue.

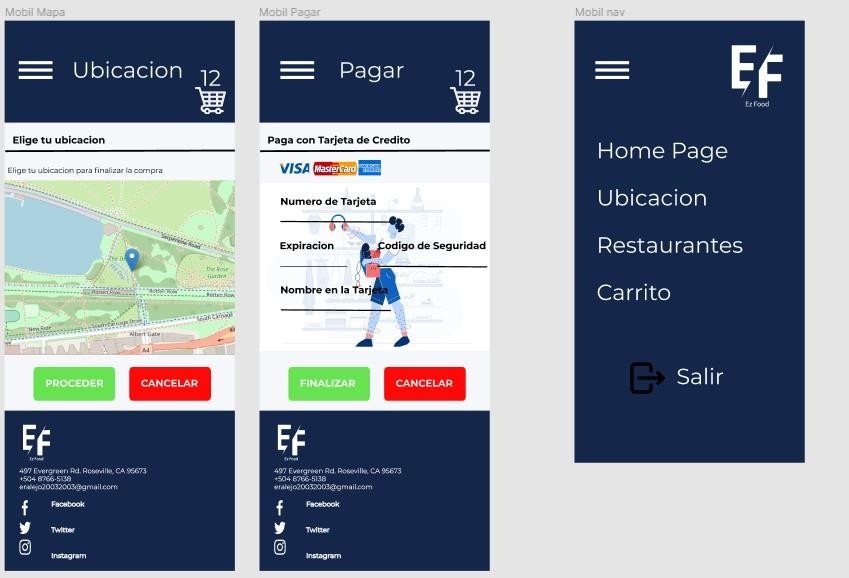
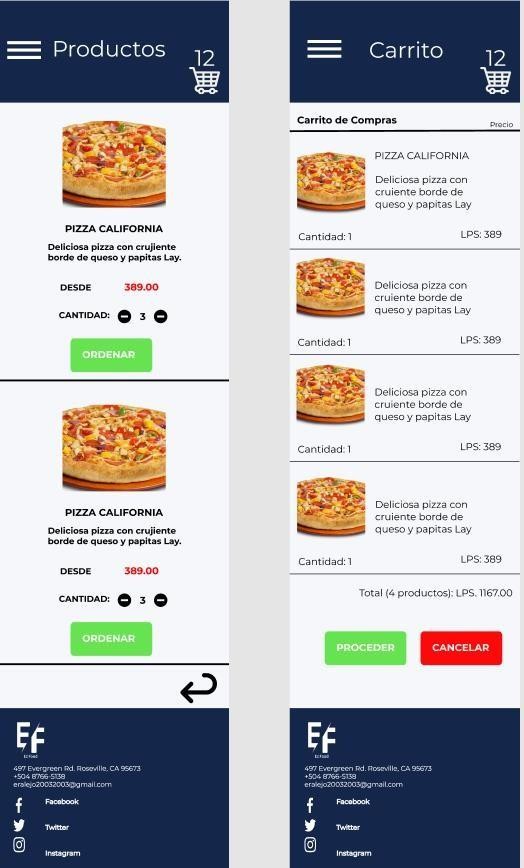


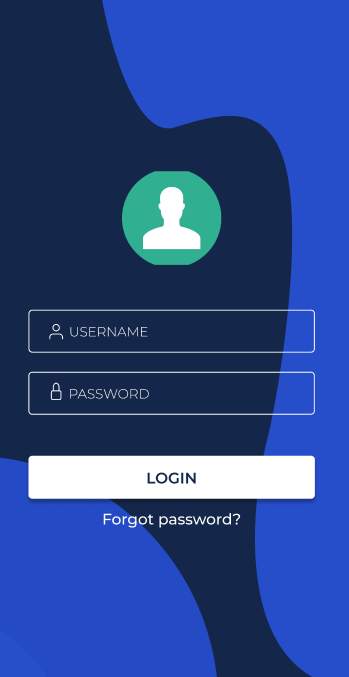
# Landing Page



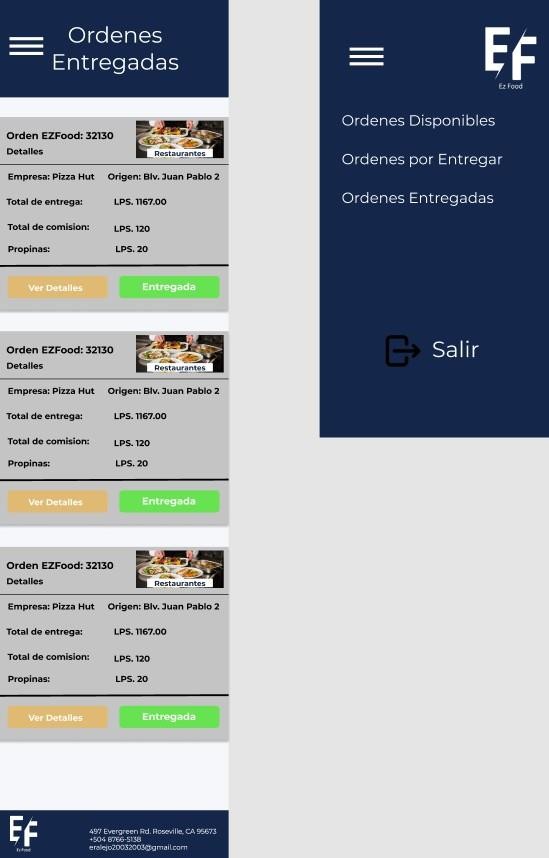
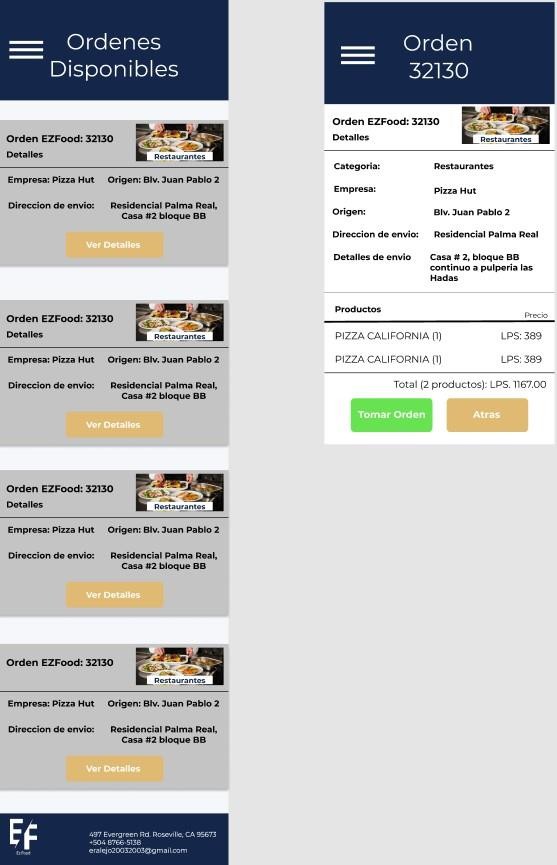
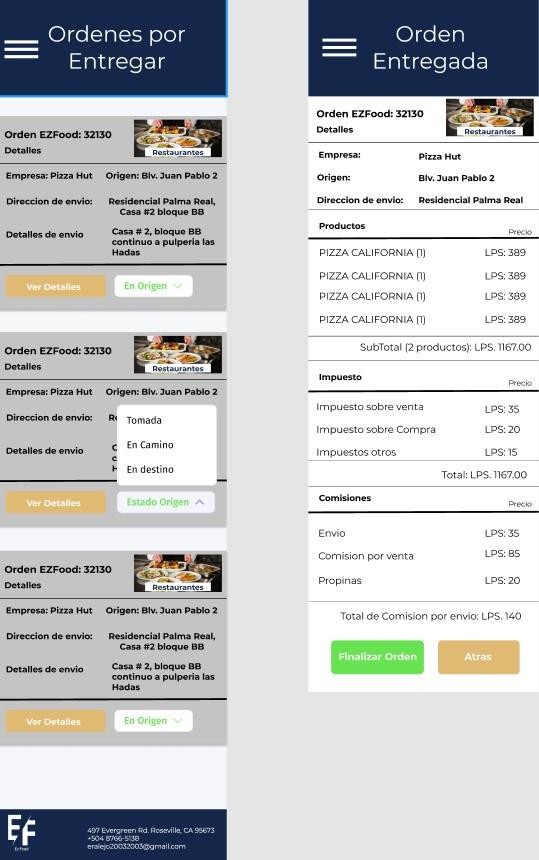
# Aplicación para cliente

Universidad Nacional Autónoma de Honduras | CIUDAD UNIVERSITARIA | Tegucigalpa M.D.C. Honduras C.A | [www.unah.edu.hn](http://www.unah.edu.hn/)

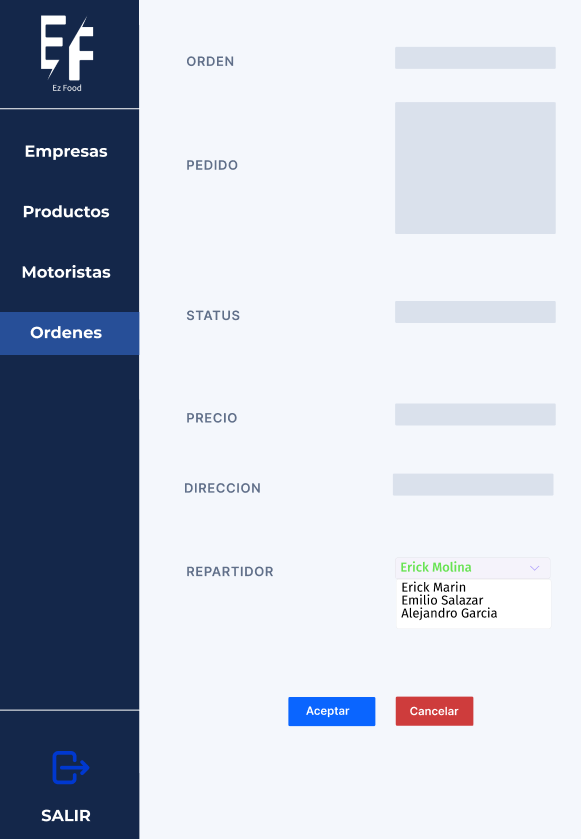
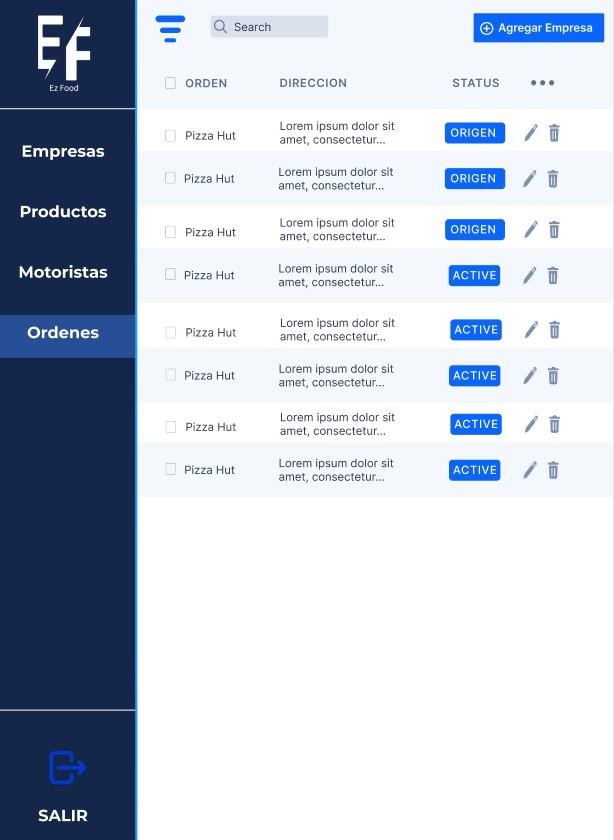
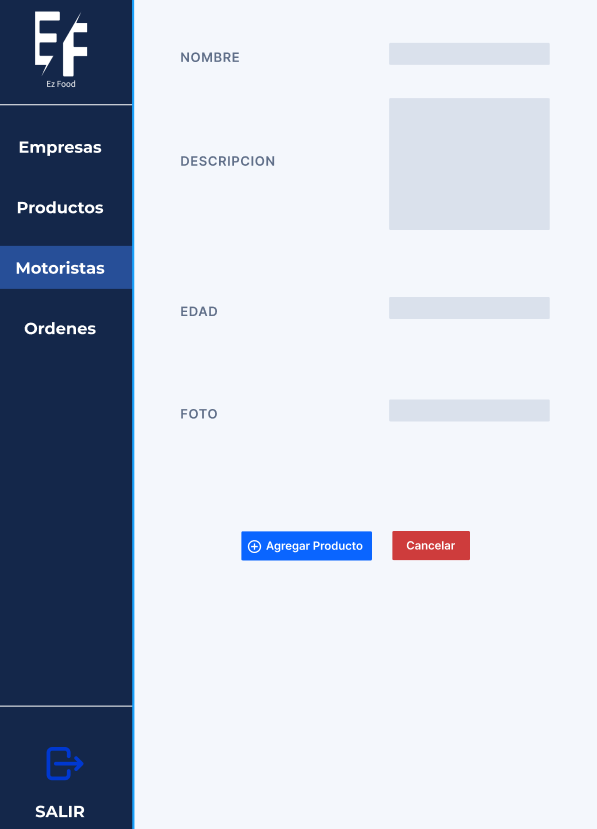
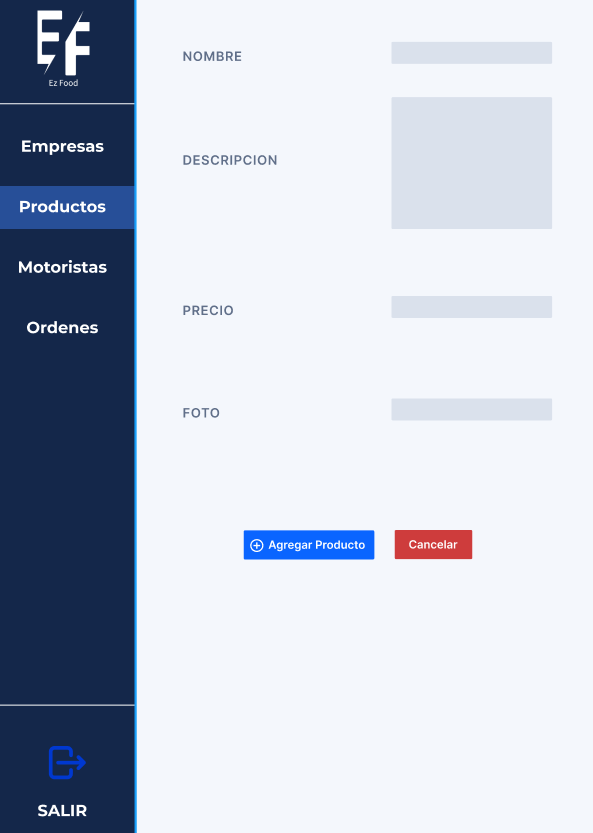
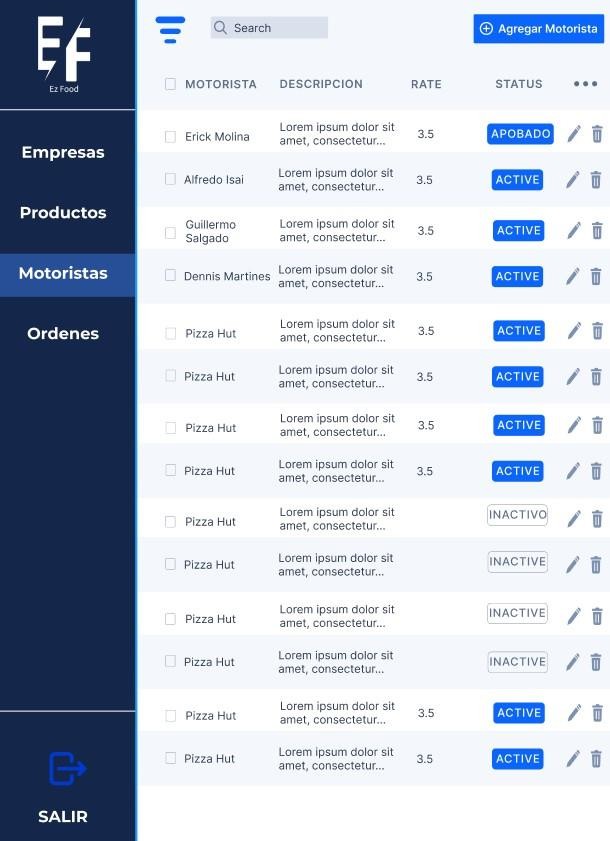
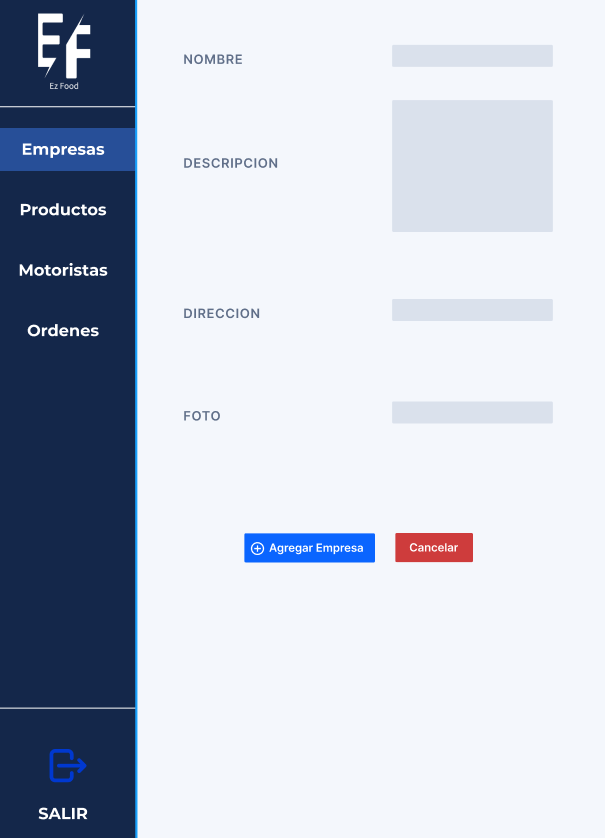
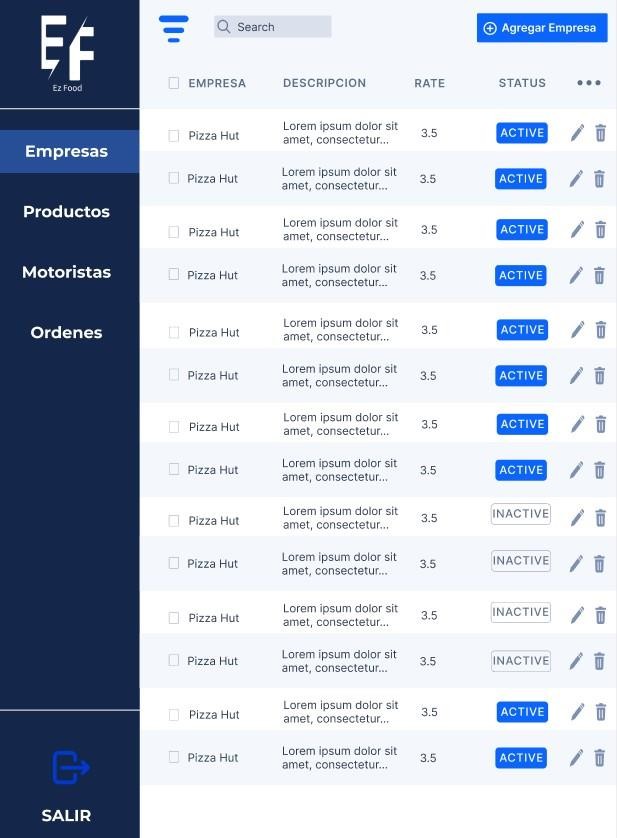
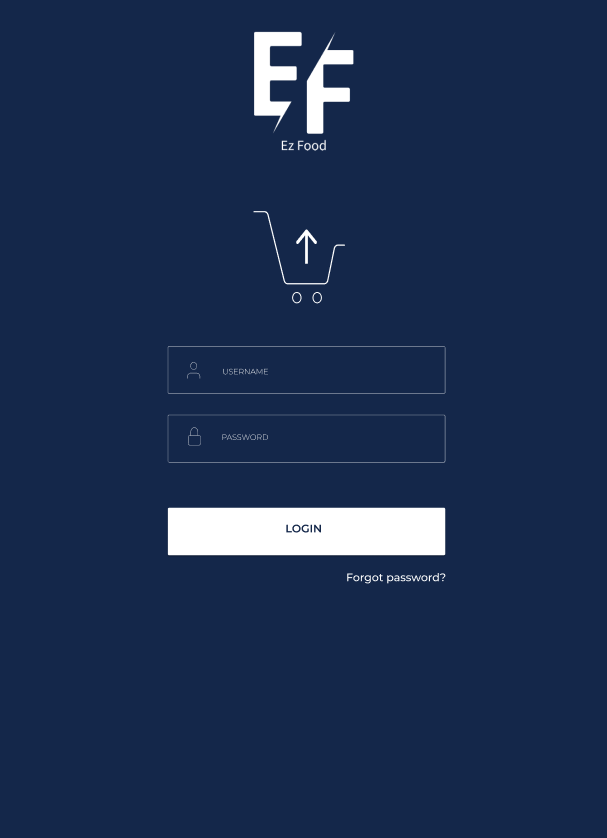




# Aplicación para Repartidores



# Aplicación de administradores



# Mockups en figma

Todos los Mockups encontrados en este informe se pueden visualizar en el siguiente link:

# Video explicativo de la funcionalidad del proyecto en base a los mockups

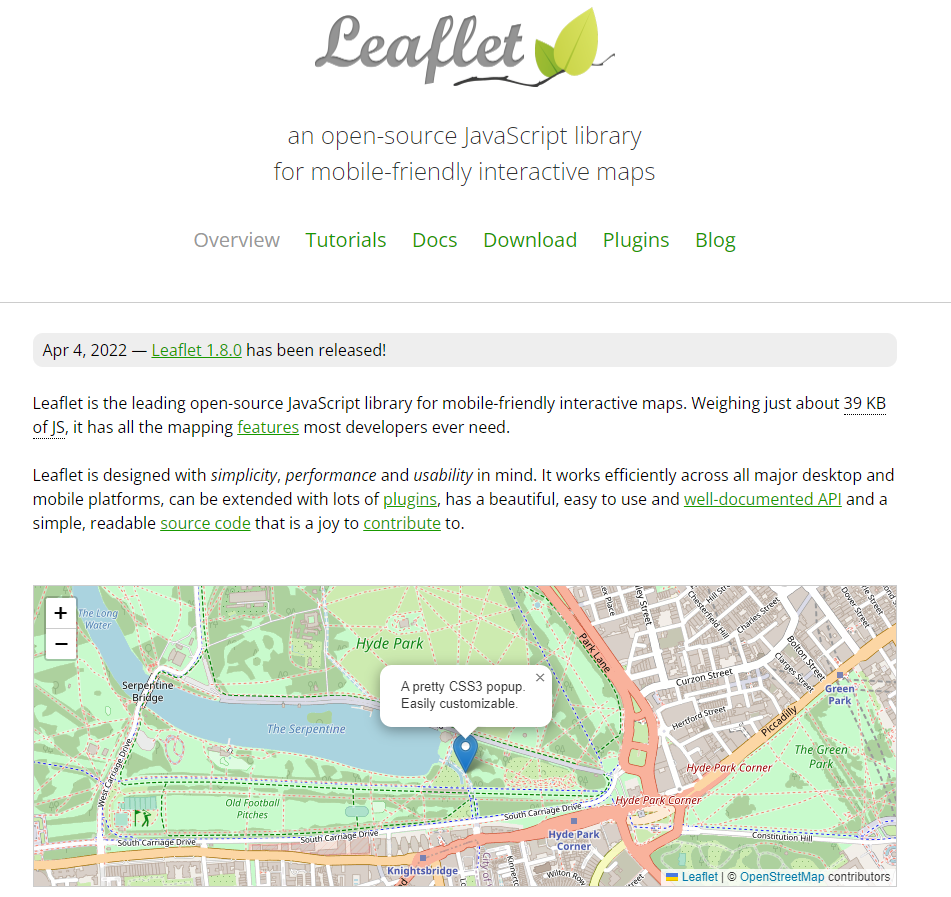
Aun no disponible.

**Herramientas utilizadas**

Las herramientas utilizadas en básico para la creación de los repositorios:

* Página Clientes (FrontEnd):
  + JavaScript
  + CSS
  + HTML
  + Leaflet
* Página Repartidores (FrontEnd):
  + JavaScript
  + CSS
  + HTML
* Página Administradores (FrontEnd):
  + JavaScript
  + CSS
  + HTML
* BackEnd:
  + PHP

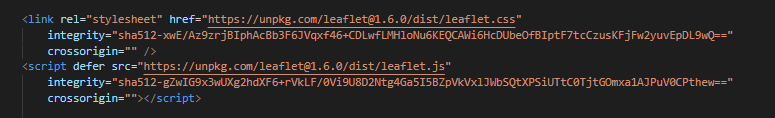
**Investigación nueva herramienta (Leaflet)**



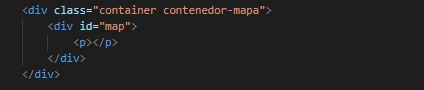
Leaflet es una liberaría para JavaScript la cual nos provee una API en base a google maps, en donde podemos utilizar varias herramientas para generar un mapa.

También podemos usar métodos que nos facilitan la ubicación del mapa, por ejemplo, podemos usar métodos que soliciten permiso al navegador para tomar la ubicación del usuario, y paso seguido, podemos renderizar el mapa justamente en la ubicación que se brindó del browser.

**Como funciona Leaflet**

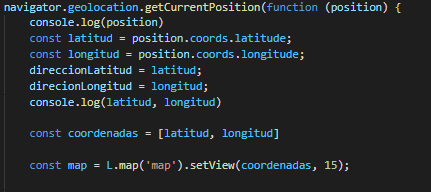
* Para que Leaflet funcione, empezaremos agregando su cdn a nuestro archivo index.html y luego, podremos usar la librería.
* Luego, debemos de colocar un <div> en el archivo html, con el especifico ‘id’ de ‘map’, de la siguiente forma <div id=”map”></div>, y de esta forma, Leaflet automáticamente detectara ese contenedor, y es donde el mapa será renderizado.

Captura tomada del código del repositorio de la aplicación de clientes



Captura tomada del código del repositorio de la aplicación de clientes

* Luego definimos las coordenadas de latitud y altitud con una función proveída por Leaflet “geolocation.getCurrentPosition” y luego, hacemos un setView de esas coordenadas en nuestro div del mapa y listo.



Captura tomada del código del repositorio de la aplicación de clientes

**Funcionalidades implementadas**

* Página de Clientes (FE):
  + Poder visualizar todas las categorías del .json del backend
  + Poder visualizar todas las empresas de cada categoría del .json del backend
  + Poder visualizar todos los productos de cualquier empresa del .json del backend
  + Poder crear una orden con cualquier producto.
* Página de Repartidores (FE):
  + Poder tomar una orden hecha desde el FE de clientes y cambiarle el estado de “origen” a “tomada” o “entregada” o “camino”.
  + Poder visualizar todas las ordenes disponibles o sin asignar.
  + Poder ver los detalles de las órdenes.
  + Poder finalizar las ordenes.
* Página de Admin (FE):
  + Poder ver todas las categorías
  + Poder ver todas las empresas
  + Poder ver todos los productos
  + Poder ver todas las ordenes
  + Poder ver todos los repartidores
  + Crear una empresa
  + Crear un producto
  + Asignar una orden
* BE:
  + Provee todos los endpoints para todos los repositorios del FE.

Las funcionalidades están más ampliamente explicadas en el video.

**Video del proyecto Final**