

Reto Final

ECOFFE

1 Desarrollo de Aplicaciones Web

Laura Gil Óscar Ruiz David Torrubia

Índice

| | |
|-----------------------------------------------------|----|
| 1. Introducción | 3 |
| 2. Objetivos y requisitos | 6 |
| 2.1. Objetivos | 7 |
| 2.2. Requisitos | 8 |
| 3. Gestión | 10 |
| 3.1. Metodología ágil | 11 |
| 3.2. Estructura GIT | 13 |
| 3. Análisis y diseño | 14 |
| 3.1. Diagrama de casos de uso | 15 |
| 3.2. Diagrama de clases | 16 |
| 3.3. Prototipo visual en axure | 17 |
| 3.4. Secuencia de uso | 23 |
| 3.5. Formación y Orientación Laboral e Inglés | 26 |
| 4. Estudios propios | 28 |
| 5. Conclusiones | 31 |
| 7. Bibliografía | 33 |
| 8. Anexos | 35 |

Introducción

¿Qué es Ecoffe?



Cafetería ecológica
Santa Mónica, LA, EEUU



**Mínimo
impacto
medioambiental**



**Productos
de alta
calidad**



**Compromiso
social**

Somos una pequeña star up con sede localizada en Santa Mónica, Los Ángeles, Estados Unidos, fundada por 3 desarrolladores: David Torrubia, Óscar Ruiz y Laura Gil.

Nuestra empresa es una cadena de cafeterías ecológicas, debido a que utilizamos envases reciclados y biodegradables, medios de transporte que generan el mínimo impacto ambiental y promovemos el comercio local de la zona donde establecemos nuestras cafeterías haciendo acuerdos comerciales con agricultores locales.

Actualmente contamos con la oficina en Santa Mónica, Estados Unidos y una cafetería en Zaragoza, España. También en junio de 2022 tenemos previsto abrir otra cafetería en Madrid, España.

Servicios que ofrecemos

Tienda física



La ventaja de ser una empresa con tienda física es que se puede ofrecer una mejor atención al cliente y este lo desea.

La atención al cliente presencial permite un trato más cercano y cálido con el cliente.

Tienda online



Ser una empresa con tienda online a su vez permite un acceso más personalizado, cómodo y rápido a la información promocional de la empresa. También facilita el proceso de compra para el usuario y permite un servicio de reparto a domicilio para los clientes que prefieran no desplazarse hasta la tienda física.

Objetivos y requisitos

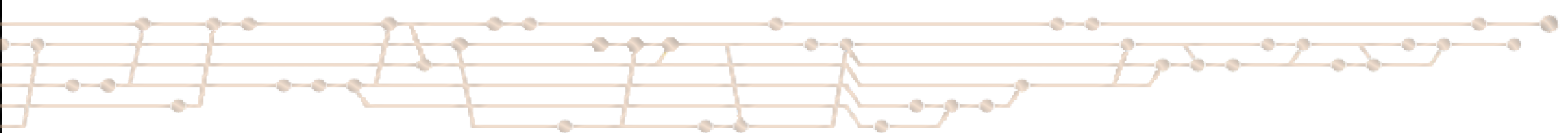
Objetivos

Objetivos principales:

1. Cumplir con los plazos de entrega del proyecto alcanzando todos los objetivos y requisitos principales propuestos y en la medida de lo posible con los secundarios también.
2. Facilitar en lo máximo posible el uso de la página web mediante la realización de casos de uso y secuencias funcionales.
3. Realizar un prototipo de axure que sirva como una guía fiable a la hora de maquetar la web y establecer la relación correcta entre las páginas.
4. Realizar una base de datos funcional y escalable para poder ampliar las funcionalidades de la página en un futuro.
5. Permitir listar y la visualización correcta de información de la base de datos.
6. Permitir el correcto registro y borrado de la información de la base de datos desde la página web.
7. Permitir la visualización correcta de los productos.
8. Permitir al usuario una compra satisfactoria.
9. Realizar una página web en inglés.
10. Ofrecer información sobre los valores de la empresa y ofertas de trabajo.

Objetivos secundarios:

1. Aprender a trabajar en equipo correctamente.
2. Aprender a planificar el desarrollo de una página web.
3. Aprender a realizar el desarrollo funcional de una página web.



Requisitos

Gestión

Metodología
ágil



Git

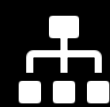


Trello



Github

Análisis



Diagramas:

- Casos de uso
- Clases
- Secuencia funcionalidad.

Base de datos

SQL

Base de datos
funcional

Prototipo

Prototipo axure

Desarrollo funcional

- Listado
- Detalle
- Registro
- Borrado

Requisitos principales:

- Planificación y gestión del proyecto mediante el uso de una metodología ágil como SRUM.
- Seguimiento del producto mediante Trello y Git.
- Almacenamiento del proyecto en un repositorio de Github.
- Realización de análisis previos al desarrollo funcional:
 - Diagrama de casos de uso.
 - Diagrama de clases.
 - Secuencia de funcionalidad
- Prototipo en axure.
- Base de datos funcional.
 - Consultas para mostrar listados.
 - Consultas de búsqueda de información.
 - Consultas de registro de información + plan de contingencia.
 - Consultas de borrado de información + plan de contingencia.
- Desarrollo funcional:
 - Implementación de funciones de listado, detalle, registro y borrado en la página web.

Requisitos

Los requisitos secundarios:

- Implantar un buscador o filtrado sobre los listados de datos.
- Realizar un diseño responsive.
- Desarrollo funcionalidad (asincronía) utilizando fetch en java script.
- Realización de la página en inglés.
- Determinar una estructura de funcionamiento de la empresa.
- Redacción de una oferta de trabajo.
- Redacción de un proceso de selección de empleados.
- Crear un CV en Europass de un candidato idóneo.

Gestión

Metodología ágil SCRUM

Metodología SCRUM:

La metodología SCRUM es una metodología de tipo RAD (Rapid Application Development), lo cual se caracteriza por la entrega rápida de componentes funcionales completos mediante una lista continua de entregables, cada uno llamado sprint. Cada sprint es revisado y evaluado por el equipo del proyecto y el cliente. El cliente adquiere un fuerte sentido de pertenencia, se pueden realizar cambios a tiempo, el diseño y desarrollo está centrado en el usuario y se puede producir una versión operativa desde una fase temprana del proyecto e ir completando mejorando y en sucesivas iteraciones.

Ventajas:

En un proyecto como este con unos requisitos definidos, unos objetivos cuantificables y un tiempo acotado (02/05/2022 - 30/05/2022) la metodología scrum nos ofrece la posibilidad de realizar planificaciones semanales con revisiones diarias y un mayor margen de maniobra ante posibles inconvenientes.

Planificación semanal del proyecto:

El Reto final / Proyecto Ecoffe está constituido en 4 sprints semanales y un sprint diario final.

En la primera semana se establecen tareas de planificación y estructuración del proyecto de alta importancia para el correcto desarrollo del proyecto, durante la segunda semana se plantea la completa creación de la base de datos y el comienzo de la inserción de scripts e implementación con la base del front en HTML y CSS. La tercera semana es una continuación de la segunda, dado que se realizan tareas de larga duración. El 4º sprint es una semana en la que se pulen posibles errores de los anteriores sprints, es decir, es un margen de error en el caso de que ocurran inconvenientes durante el desarrollo del proyecto; también es una semana de preparación de documentos a entregar y preparación de la presentación final.

En el último sprint de un día se realiza una última revisión del proyecto y se entrega.

Posteriormente se realizará una defensa del proyecto el 2 de Junio de 2022.

Sprints semanales

02-08
MAYO

SPRINT 1

Planificación y análisis

09-15
MAYO

SPRINT 2

Desarrollo de las tareas urgentes y de larga duración

16-22
MAYO

SPRINT 3

Continuación sprint 2 y desarrollo de las tareas sencillas y no urgentes

23-29
MAYO

SPRINT 4

Continuación sprint 3, implementación de mejoras y revisión general

30
MAYO

SPRINT 5

Revisión final y entrega del proyecto

Metodología ágil SCRUM

Planificación semanal del proyecto:

Anexo: Documento excel 'Planificacion_Ecoffe'

Historias de usuario:

Las tareas de la planificación han sido evaluadas mediante la realización de historias de usuario. En dichas historias de usuario se ha evaluado cada tarea según la prioridad, el riesgo y la valoración estimada por parte del cliente.

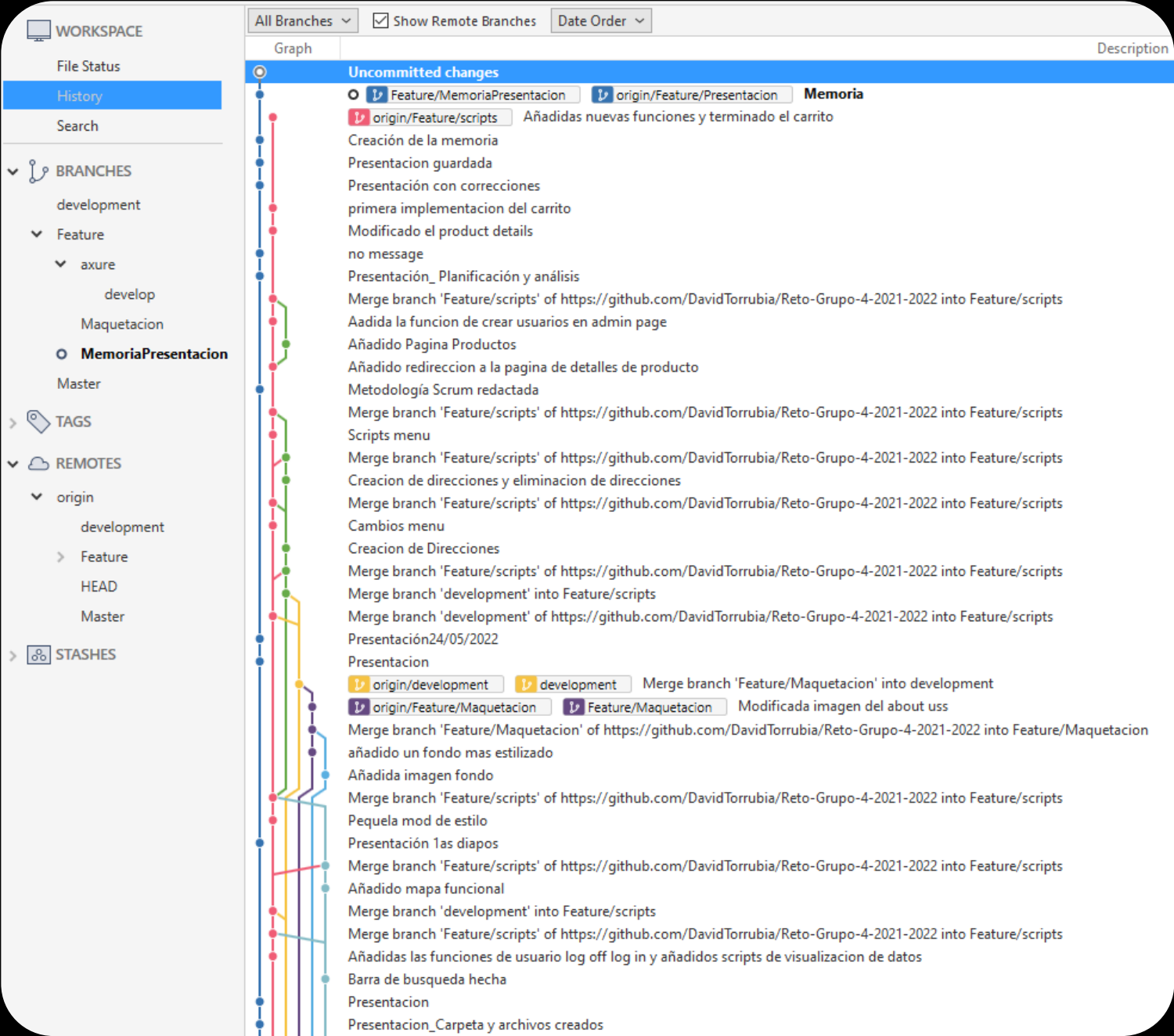
Anexo: Documento excel 'Historias de Usuario_Ecoffe'

Seguimiento del proyecto:

Durante el proyecto de Ecoffe se ha realizado un seguimiento de las tareas mediante la herramienta Trello y se han almacenado los datos en la excel de planificación mencionada anteriormente.

Anexo1: Documento excel , carpeta Entornos 'Planificacion_Ecoffe' 2a pestaña

Gestión GIT



Git:

La realización de las tareas ha sido organizada mediante la metodología gitflow, dicha metodología establece un punto de partida que se va ramificando realizando diferentes cambios que se van añadiendo a la rama principal una vez están acabados.

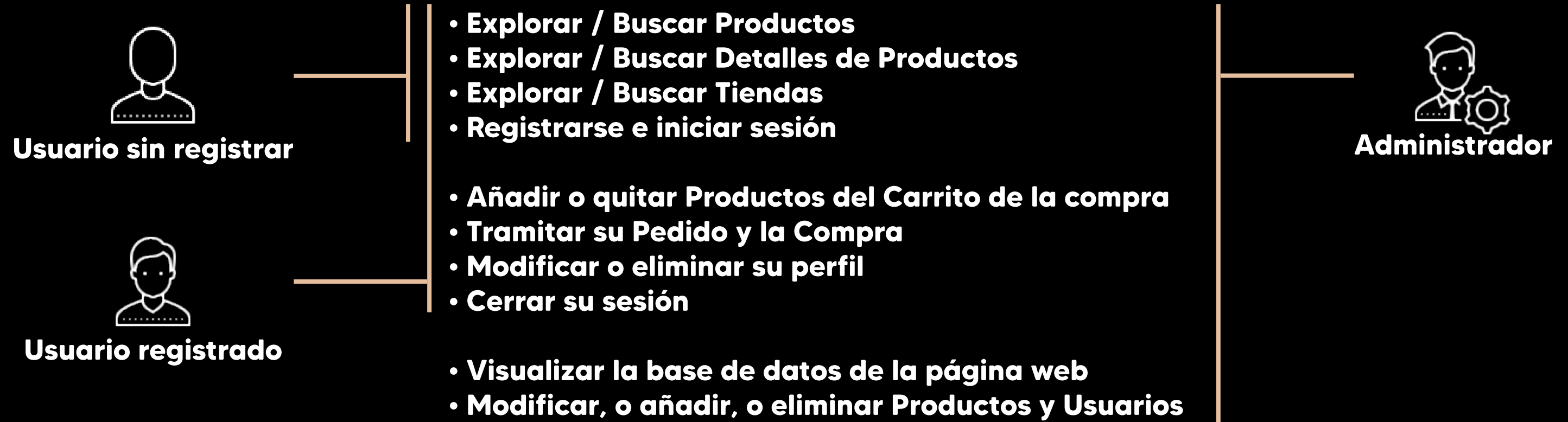
Esta metodología permite almacenar todos los cambios realizados y volver a pasos anteriores si es necesario.

La estructura de ramas que se ha seguido es la siguiente:

- Master
- Development
- Features:
 - Axure
 - Maquetación
 - Scripts
 - Presentación y Dossier

Análisis y diseño

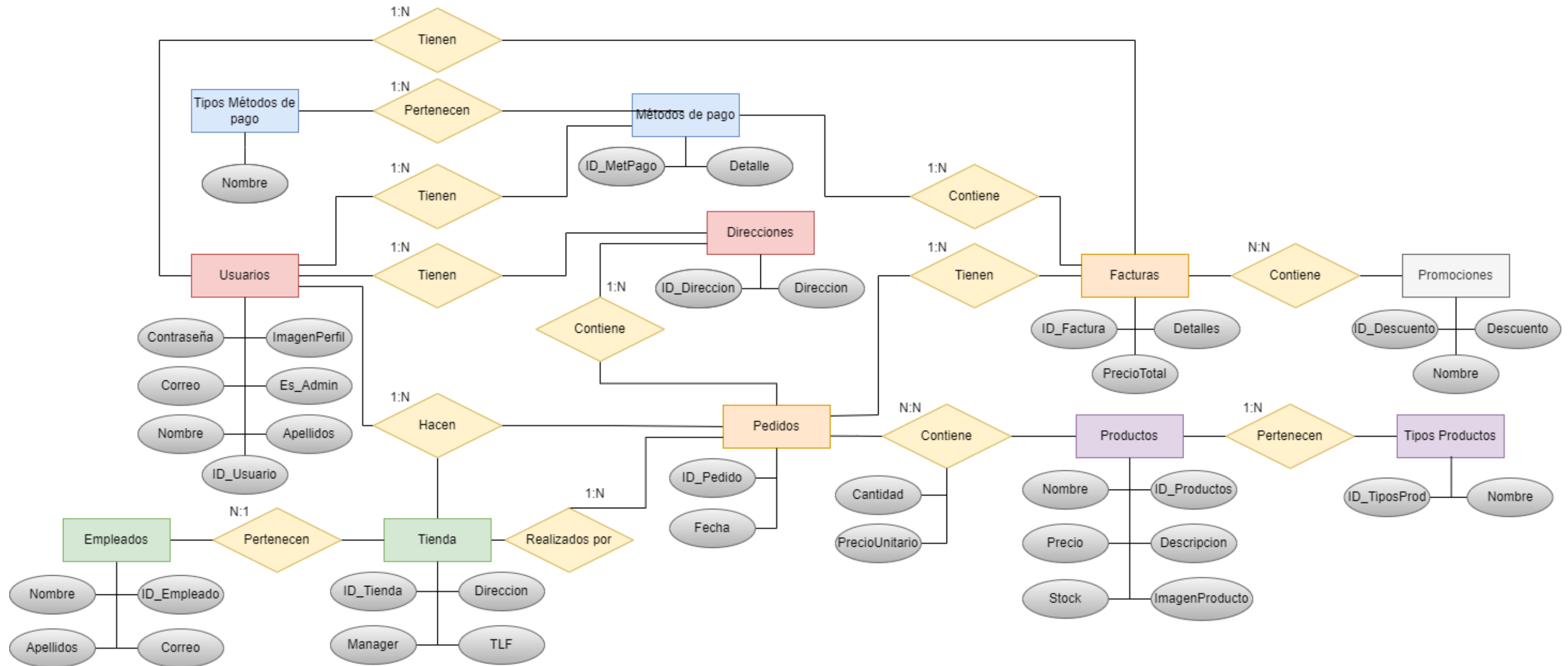
Diagrama de casos de uso



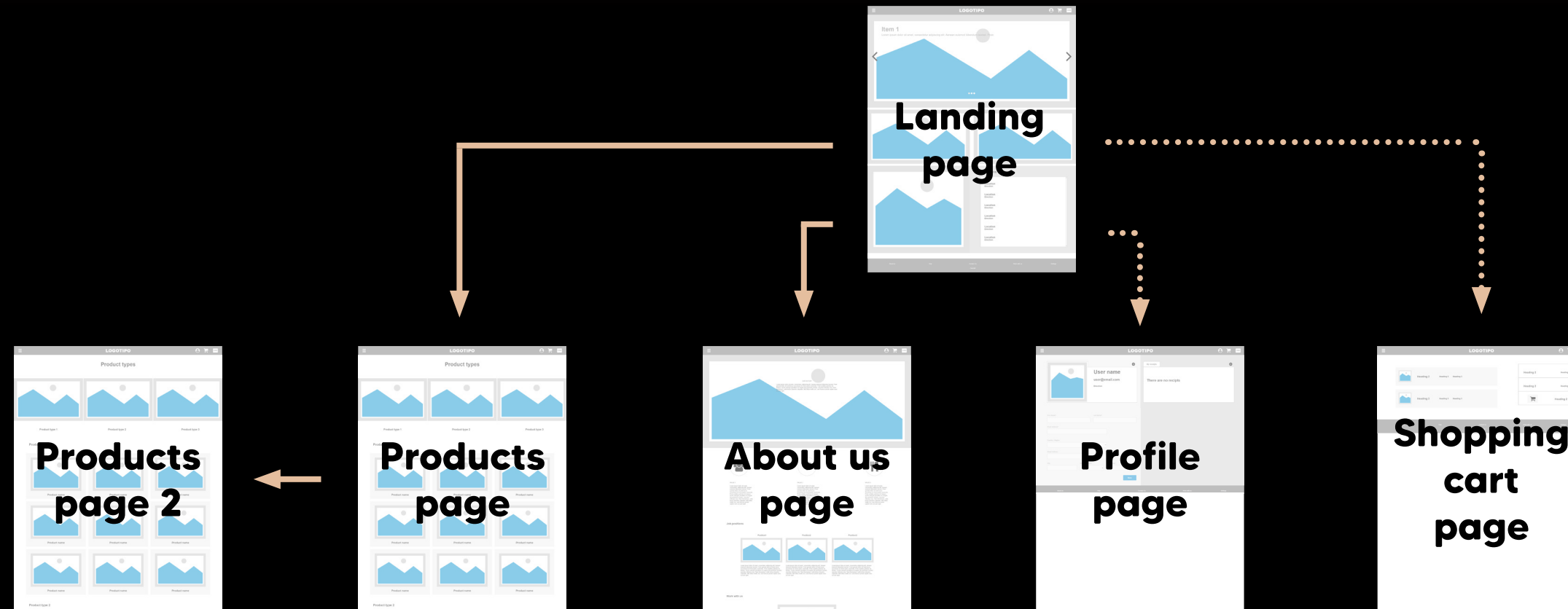
Realizando el diagrama de casos de uso, reflejamos que hay 3 tipos de usuarios:

- El usuario sin registrar es aquel usuario que entra por primera vez a la página y observa qué podemos ofrecer como empresa, se le permite el acceso a la búsqueda de productos, detalles, tiendas y la posibilidad de registrarse e iniciar sesión si lo desea.
- Al usuario registrado se le permiten los mismos accesos que el anterior, pero además tendrá un perfil que le permitirá poder realizar compras y pedidos online, también podrá modificar o eliminar su propio perfil y cerrar sesión.
- Los usuarios administradores de la página tendrán acceso a todo, sobretodo a la base de datos de la página, en la que podrán modificar, añadir o eliminar productos y usuarios como mínimo.

Diagrama de clases



Prototipo Axure



El prototipo realizado en axure consta de 5 páginas:

- **Landing page:** la página de inicio en la que se observan ofertas, anuncios, noticias de la empresa y localizaciones de las tiendas.
- **Products page:** se accede a través de un menú desplegable situado en el header de la landing page. Informa del menú de productos que contienen las cafeterías. Principalmente se observan los tipos de productos.
- **Products page 2:** se accede al clicar sobre una tipología de producto, y entonces muestra todos los productos de esa tipología.
- **About us:** se accede a través de un menú desplegable situado en el header de la landing page. Contiene información general de la empresa (mision, visión y valores), puestos de trabajo y vacantes.
- **Profile page:** Página a la que solo tiene acceso el usuario registrado. Se accede a través del header de la landing page en el icono del perfil, se rellena el login del desplegable, y si es correcto dará acceso. Contiene información personal del usuario registrado y los recibos de las compras.
- **Shopping cart page:** Página a la que solo tiene acceso el usuario registrado. Una vez realizado el log in, el usuario puede realizar la compra seleccionando productos y acceder a su carrito de la compra.

Prototipo Axure: Landing page



La página principal se compone de un header fijo que también está en el resto de las páginas.

Header:

El header se compone de un menú desplegable que da acceso al menú de productos y a la página about us.

En el centro se sitúa el logotipo que permite la vuelta a la página principal si el usuario no se encuentra en la página principal. Por último, a la derecha del header se encuentra el menú desplegable de usuario, el acceso al carrito de la compra y el cambio de idioma.

En el logotipo del usuario se encuentra un menú desplegable que permite iniciar sesión, registrarse como nuevo usuario y cerrar sesión; asimismo, dicho menú desplegable contiene dos formularios desplegables para que el usuario pueda completar los campos necesarios para el acceso, como su email y contraseña.

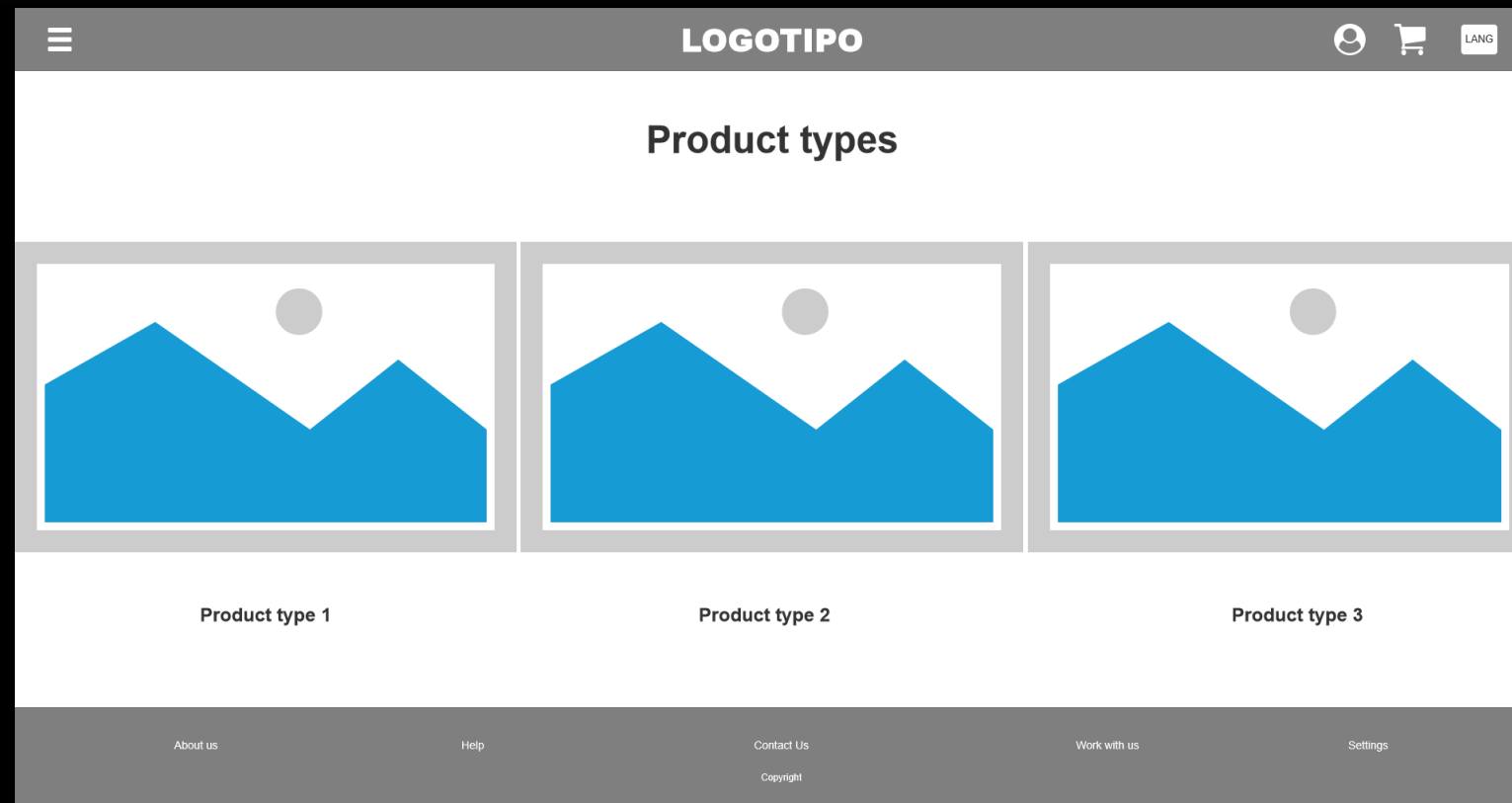
Footer:

El footer contiene otros accesos más directos a la página de about us y otras opciones secundarias para los usuarios como pedir ayuda para hacer consultas de manera más personalizada, que aún no está habilitado.

Body:

El cuerpo de la página se compone de un carrusel de anuncios principales, un poco más abajo, se sitúan los anuncios de las próximas aperturas y finalmente, abajo del todo se sitúa un mapa con las localizaciones de las tiendas. Dicha sección de localizaciones posteriormente se coloca en un desplegable para obtener una mayor limpieza visual.

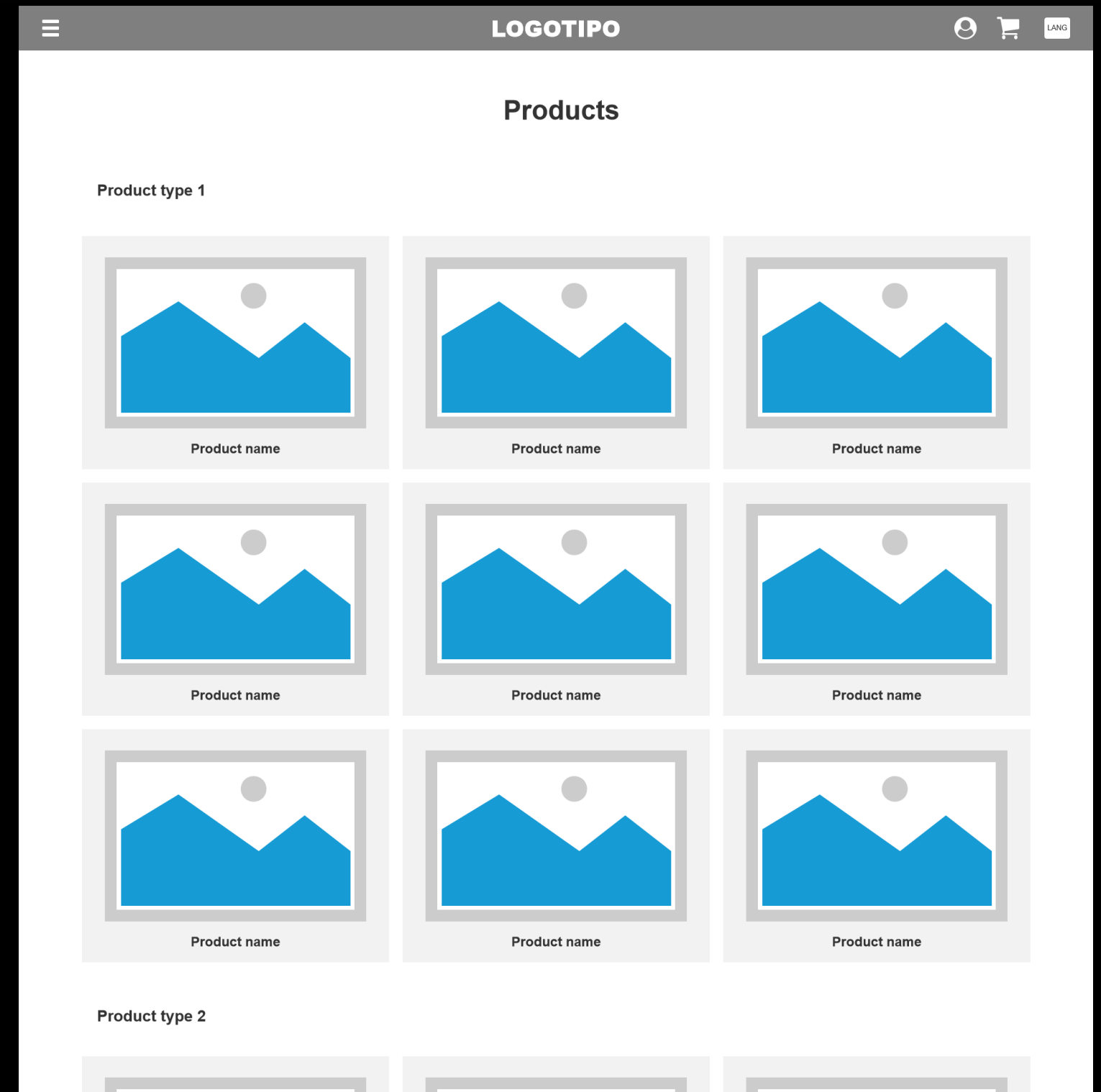
Prototipo Axure: Products page



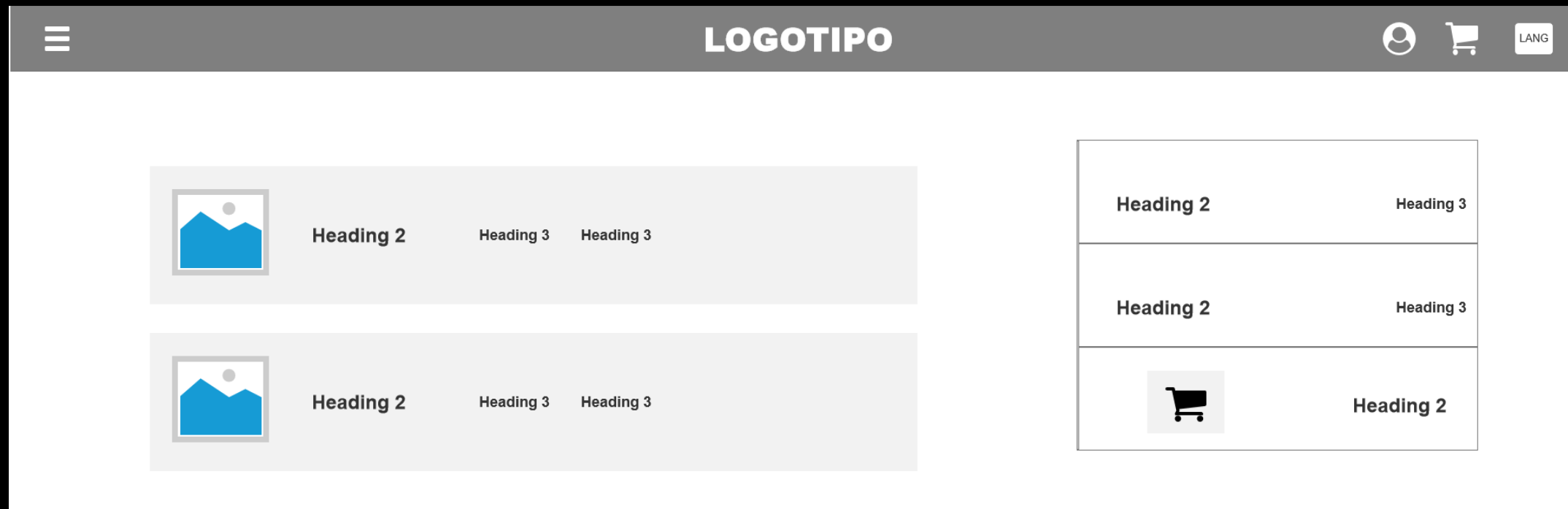
Las páginas de productos se componen de una matriz 3xN de tarjetas de productos.

En la 1a página de productos se visualiza la tipología de productos y en la segunda página, a la que se accede clicando sobre una tipología de productos, obtenemos la misma matriz pero con los productos de dicha tipología.

Posteriormente esta estructura se ha integrado en una sola página en la que al clicar una tipología se hace auto-scroll hasta donde se sitúan los productos de dicha tipología. También se ha incorporado un botón en la zona derecha abajo para volver a la zona de tipología de producto, de esta manera se facilita la vuelta atrás en las búsquedas



Prototipo Axure: Shoppingcart page



La página del carrito tiene una composición sencilla en la que principalment se ve un listado de las tarjetas de los productos a comprar, el nombre, el precio unitario y la cantidad de productos iguales. En el lado derecho se observa un pequeño resumen de la compra, como si fuese el recibo, en el que al lado del precio final se situa el botón de compra.

Posteriormente se han implantado botones de edición y eliminación de productos y un dsplgable en el botón de la compra en la que surgen las opciones de pago.

Prototipo Axure: Profile page

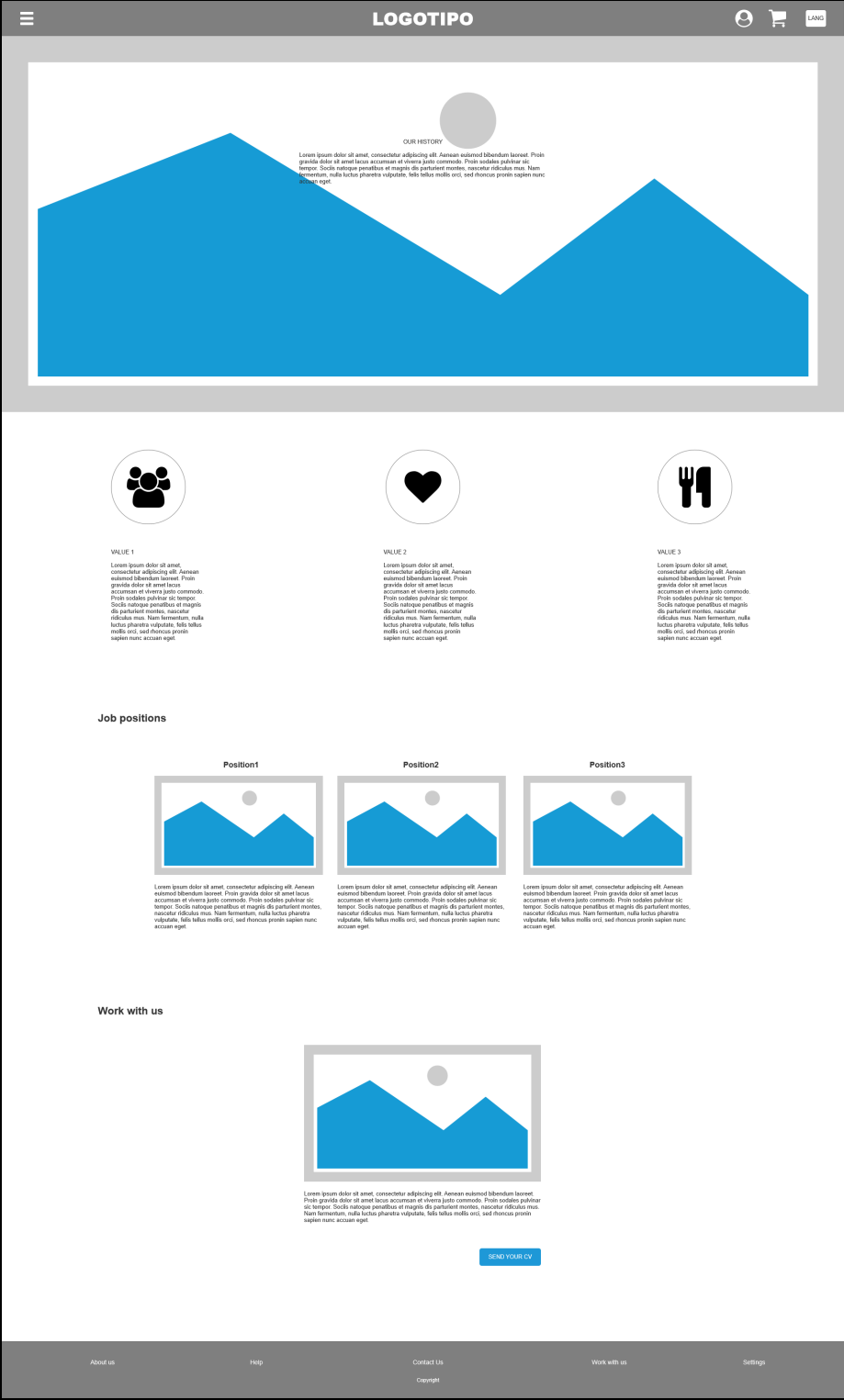
The prototype shows a user profile page with a header containing a menu icon, the logo 'LOGOTIPO', and user icons. The main content area is divided into two panels. The left panel, titled 'User data', contains a profile picture placeholder, the text 'User name', the email 'user@email.com', and a label 'Direction'. The right panel, titled 'My receipts', displays 'There are no receipts'. A settings modal is overlaid in the foreground, featuring input fields for 'First Name*', 'Last Name*', 'Email Address*', 'Country / Region', 'Street Address', 'City', 'State / Province', and 'ZIP / Postal', along with a 'Save' button.

La página de usuario se compone de dos partes, los datos del usuario a la izquierda y los recibos de las compras a la derecha.

En los datos del usuario principalmente se puede ver la foto de perfil, el nombre del usuario, el correo electrónico con el que se ha registrado y la dirección principal.
Al clicar en el símbolo de settings se despliega un formulario que permite al usuario modificar sus propios datos.

Análisis y diseño

Prototipo Axure: About us page



En la página About us está compuesta al inicio por un texto introductorio de nuestra empresa sobre una imagen de fondo, a continuación se objerva en tres columnas los valores de marca. Posteriormente se encuentran los puestos de trabajo qu estructuran la empresa también en 3 columnas, y finalmente un apartado con un breve texto y un botón que permite enviar el curriculum al usuario si desea trabajar con nosotros.

Posteriormente se añade la opción de ver ofertas de puestos vacantes.

Diagrama de secuencia de funcionalidad

Proceso de compra online de un usuario registrado

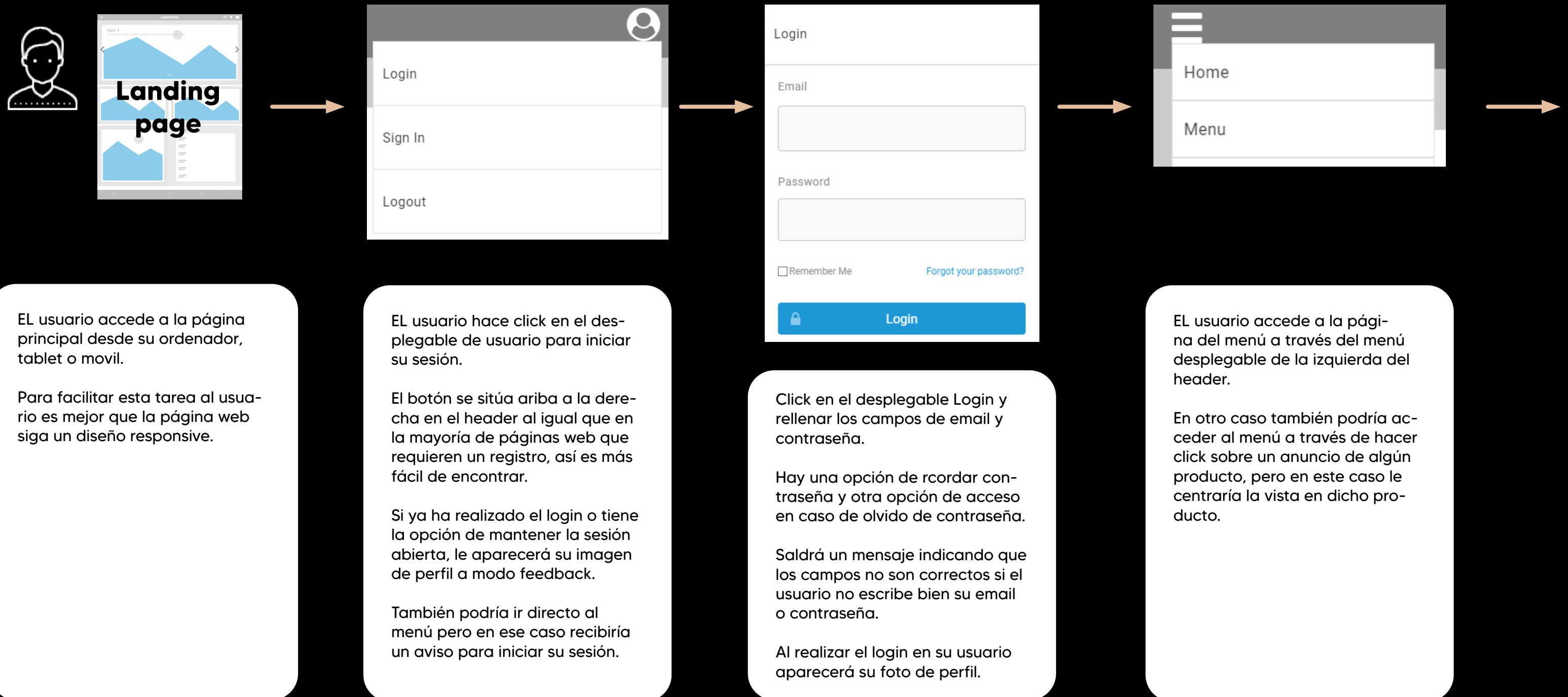


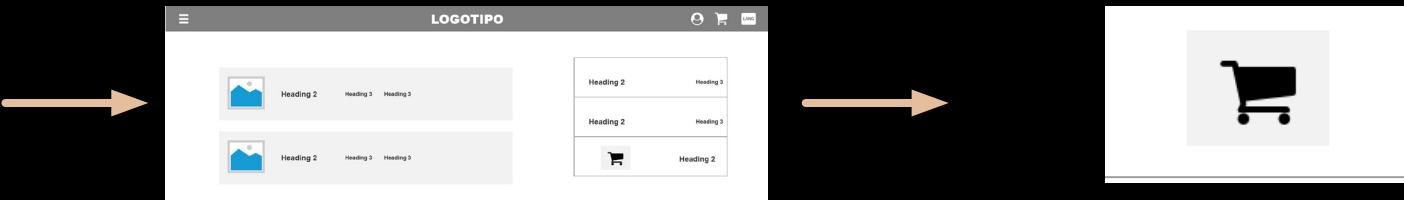
Diagrama de secuencia de funcionalidad

Proceso de compra online de un usuario registrado



Diagrama de secuencia de funcionalidad

Proceso de compra online de un usuario registrado



El usuario accede a lapágina del carrito de la compra y comprueba los datos de su pedido.

El usuario hace click en el botón de compra, rellena el desplegable del modo de pago y finaliza la compra con éxito.

Formación y Orientación Laboral

Estructura de la empresa



Managers



Camareros



Repartidores

La estructura mínima de la empresa en cada local estará constituida por:

- 1 Manager: La persona manager estará al cargo de la gestión del local además de realizar las tareas propias de un camarero como preparar pedidos, atención al cliente y mantenimiento del local. Cobrará 1600€ al mes.
- 2 Camareros: Los camareros realizarán las tareas de atención al cliente y preparación de pedidos. Han de ser mínimo 2 para al menos cubrir uno el horario de mañana y otro el horario de tarde. Cobrarán 1300€ al mes.
- 2 Repartidores: Los repartidores se encargarán de llevar el pedido al cliente si dicho cliente solicita el reparto a domicilio. También deberán de cobrar al cliente si solicita el pago en mano. Han de ser 2 repartidores para poder cubrir el turno de mañana y el de tarde. Cobrarán 1300€ al mes.

Formación y Orientación Laboral

Ejemplo de oferta de trabajo:

Bartender

Indefinite contract

Full-time

Work schedule: M 7:00 - 15:00 / A 15:00 - 23:00

Zaragoza, C/Violeta Parra 8

Madrid, C/Gran Vía 41

Description

Ecoffe is a company of coffeshops which is compromised with the natural enviroment. We ensure the minimal enviromental footprint, the best quality ingredients and a warm friendly treatment between clients, suppliers and employees.

Responsabilities:

Customer service
Order management
Keep work area clean

Requirements:

Kind and friendly attitude
Desire to learn
Good presence
English skills
Spanish skills

Proceso de selección:

Los currículums recibidos serán clasificados según el personal reclutador de talento de recursos humanos. Si los currículums recibidos superan los 100 por candidatura, se aplicará un filtrado previo mediante programas informáticos de empresas de reclutamiento subcontratadas.

A continuación se citarán a los candidatos de los currículums seleccionados en grupos de 10 personas y se les procederá a realizar una entrevista grupal en la que se les realizará una explicación del puesto o puestos vacantes, se les preguntará las preferencias en cuanto a vacantes y horarios, y finalmente se realizará una dinámica en grupo para observar qué actitud y roles toman los candidatos a la hora de trabajar en equipo.

Una vez pasada la primera entrevista se citarán a los candidatos que más encajen con el puesto de trabajo a una entrevista individual.

Si un candidato supera dicho proceso de selección será contratado y pasará a formar parte de nuestra empresa.

Estudios propios

Conexión de la base de datos con los scripts

La implementación de scripts en la página se ha realizado mediante PHP.

Para conectar los scripts con la base de datos y la maqueta de la página se ha creado un documento php llamado 'conexion'. En dicho documento se han creado los siguientes métodos para poder realizar dicha conexión.

Empezamos enlazando la página web con la base de datos; para ello creamos 4 variables (user, pass, server y db) donde almacenamos el usuario de la base de datos, la contraseña, el servidor donde esta alojada y el nombre de la base de datos.

```
<?php
error_reporting(0);

$user="root";
$pass="";
$server="localhost";
$db="reto";
$con= new mysqli($server,$user,$pass,$db);

if($conexion->connect_errno) {
    die("La conexion ha fallado" . $conexion->connect_errno);
}
```

Se hace una conexión con la base de datos y en caso de que de fallo muestra un mensaje diciendo que la conexión ha fallado. El código error_reporting(0) sirve para que no muestre avisos en la página.

Una vez este la base de datos conectada podemos hacer las consultas guardándolas en una variable y ejecutándolas con otra.

```
$sql = "SELECT * FROM PRODUCTOS WHERE ID_TIPOPROD = 1";
$result = mysqli_query($con, $sql) or die('Error');
```

Si hay más de una fila para esa variable, creamos otra para almacenar las filas.

```
while ($row = mysqli_fetch_assoc($result)) {
```


Implementación de scripts en la página con PHP

Para insertar datos maquetados como tarjetas, por ejemplo en el caso de los productos, se realiza de la siguiente manera:

Se ha cambiado la extensión de los documentos html por .php y se ha integrado el código de php utilizando las etiquetas **<?PHP (contenido) ?>**

```
<div class="Content-Products-Container-Item" <?PHP echo "id=P" . $row["ID_PRODUCTO"] ?> onclick="goto(<?php echo $row['ID_PRODUCTO']; ?>)">
   class="Content-Products-Container-Item-Image">
  <h2 class="Content-Products-Container-Item-Header"><?PHP echo $row["NOMBRE"] ?></h2>
  <h3 class="Content-Products-Container-Item-Precio"><?PHP echo $row["PRECIO"] ?>€</h3>
</div>
```

El método **echo** se utiliza para imprimir por pantalla.
La variable **row** se utiliza para definir un array o lista.

Para terminar de insertar los datos, se cierra el while y el try catch

```
$sql = "INSERT INTO USUARIOS (CORREO,CONTRASENA,ES_ADMIN) VALUES ('" . $email . "','" . $password . "', 0)";
$result = mysqli_query($con, $sql) or die('Error');
$sql = "COMMIT";
```

```
} catch (mysqli_sql_exception $ex) {
    $_SESSION['Error'] = 5;
    $sql = "ROLLBACK";
    $result = mysqli_query($con, $sql) or die('Error');
}
```

Conclusiones

Conclusiones

- El proceso de realización de una página web implica revisiones constantes y una buena comunicación entre los miembros del equipo, por ello el uso de la metodología scrum y un seguimiento de las tareas diario ha sido fundamental a la hora de desarrollar el proyecto.
- Los análisis de mercado, la observación de cómo interactúan con el usuario otras páginas web de cafeterías ha tenido un gran peso a la hora de diseñar nuestra página web. Se ha buscado facilitar al usuario la interacción con nosotros y a su vez obtener un modelo de página web escalable, para ser capaz de tener una mejora continua de ésta a medida que crece la empresa.
- La integración de la base de datos con las funcionalidades y la maquetación de la página web fue una tarea en la que se debería de haber dedicado más tiempo para estudiar diferentes opciones y poder escoger la mejor manera al comienzo del proyecto.

Futuras mejoras:

- Perfeccionar los mensajes de error a modo feedback para el usuario.
- Mejorar la Información a mostrar de los recibos.
- Creación de un usuario manager, encargado de la cafetería local con permisos a un registro de recibos y compras, para facilitar la gestión de la cafetería.
- Implementación de ofertas sobre los productos.
- Implementar una función de reserva de mesa en la cafetería.

Bibliografía

Bibliografía

La información consultada como apoyo para realizar el proyecto proviene de las siguientes páginas web.

<https://www.w3schools.com/>

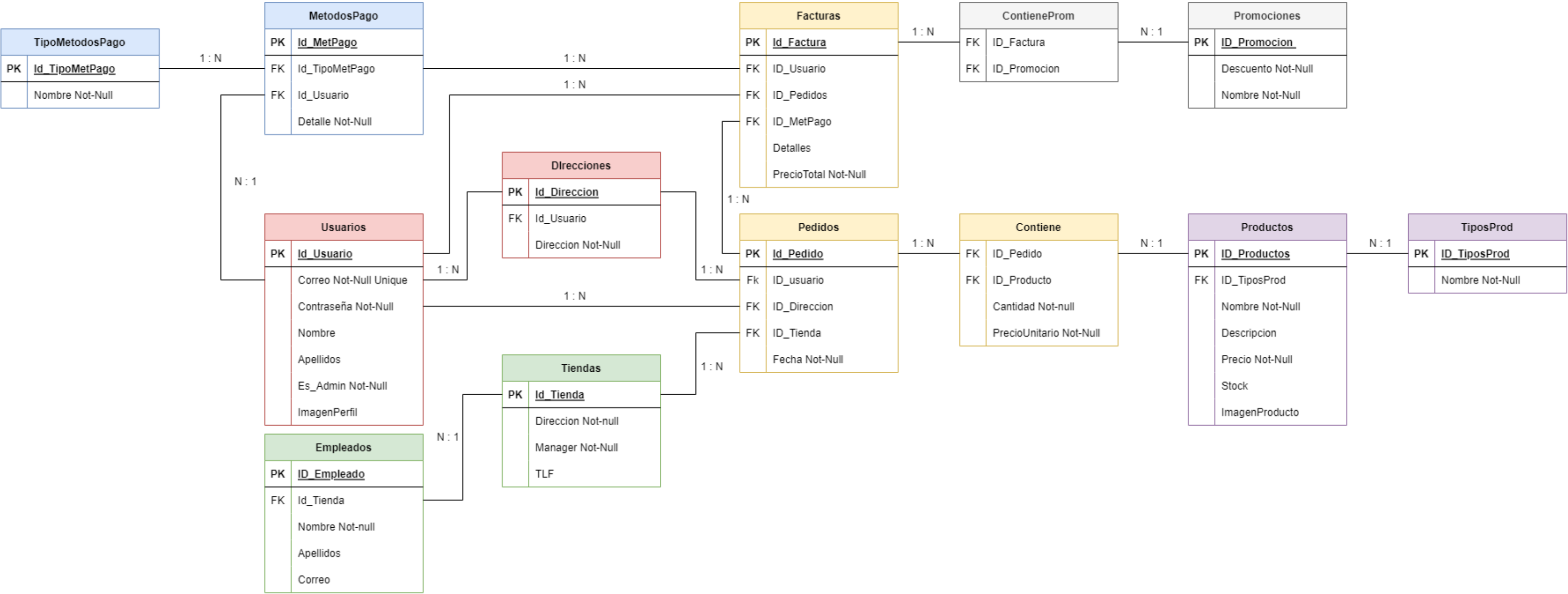
<https://es.stackoverflow.com/>

Anexos

Anexos

- Anexo 1: Documento excel , carpeta Entornos 'Planificacion_Ecoffe' 2a pestaña
- Anexo 2: Documento excel, carpeta Entornos 'Historias de Usuario'
- Anexo 3: Documento excel, carpeta Entornos 'Diagrama de casos de uso'
- Anexo 4: Documento excel, carpeta Entornos 'Diagramas de secuencia'
- Anexo 5: Documento png , carpeta Bases de datos 'Diagrama de clases'
- Anexo 6: Documento Axure, carpeta Prototipo Axure 'Prototipo_WebResponsive'
- Anexo 7: Documento png , carpeta Bases de datos 'Modelo relacional'
- Anexo 8: Documento excel , carpeta Bases de datos 'Diccionario de datos'
- Anexo 9: Documento de texto , carpeta Bases de datos 'CreaciónDbTotal'
- Anexo 10: Documento de texto , carpeta Historia Diaria 'Historia Diaria'

Anexo 4: Base de datos: Modelo relacional



Anexo 5: Base de datos: Diccionario de datos

| TABLA USUARIOS | | | | |
|----------------------------------------------------|----------------|--------|------------------|-------------------------------------------------------------|
| Diccionario de datos | Tipo | Tamaño | Restricciones | Funcion |
| ID_USUARIO | NUMBER | NA | PRIMARY KEY | Almacena el id que identifica a cada usuario |
| CORREO | VARCHAR2(60) | 60 | NOT NULL, UNIQUE | Almacena el correo de cada usuario |
| CONTRASENA | VARCHAR2(40) | 40 | NOT NULL | Almacena la contraseña de cada usuario |
| NOMBRE | VARCHAR2(45) | 45 | NA | Almacena el nombre de cada usuario |
| APELLIDOS | VARCHAR2(60) | 60 | NA | Almacena los apellidos de cada usuario |
| ES_ADMIN | NUMBER | NA | NOT NULL | Identifica si un usuario es administrador o no |
| IMAGENPERFIL | VARCHAR2(2048) | 2048 | NA | Almacena una url que direcciona a la imagen de cada usuario |
| Descripcion | | | | Fecha de creación |
| Tabla que almacenen la informacion de los usuarios | | | | 9/5/2022 |

| TABLA DIRECCIONES | | | | |
|------------------------------------------------------------|--------------|--------|---------------|-----------------------------------------------------------|
| Diccionario de datos | Tipo | Tamaño | Restricciones | Funcion |
| ID_DIRECCION | NUMBER | NA | PRIMARY KEY | Almacena el id que identifica a cada direccion almacenada |
| ID_USUARIO | NUMBER | NA | FOREIGN KEY | Identifica el usuario al que pertenece cada direccion |
| DIRECCION | VARCHAR2(70) | 70 | NOT NULL | Detalles de la direccion |
| Descripcion | | | | Fecha de creación |
| Tabla que almacena los datos de las direcciones de usuario | | | | 9/5/2022 |

| TABLA METODOSPAGO | | | | |
|-----------------------------------------------------|---------------|--------|---------------|------------------------------------------------------------|
| Diccionario de datos | Tipo | Tamaño | Restricciones | Funcion |
| ID_METPAGO | NUMBER | NA | PRIMARY KEY | Almacena el id que identifica a cada metodo de pago |
| ID_TIPOMETPAGO | NUMBER | NA | FOREIGN KEY | identifica el tipo de metodo de pago que le corresponde |
| ID_USUARIO | NUMBER | NA | FOREIGN KEY | Identifica el usuario al que pertenece cada metodo de pago |
| DETALLE | VARCHAR2(100) | 100 | NOT-NULL | Almacena los detalles del metodo de pago |
| Descripcion | | | | Fecha de creación |
| Tabla que almacena los datos de los metodos de pago | | | | 9/5/2022 |

Base de datos: Diccionario de datos

| TABLA TIPOSMETODOPAGO | | | | |
|--------------------------------------------------------------|--------------|--------|---------------|-------------------------------------------------------------|
| Diccionario de datos | Tipo | Tamaño | Restricciones | Funcion |
| ID_TIPOMETPAGO | NUMBER | NA | PRIMARY KEY | Almacena el id que identifica a cada tipo de metodo de pago |
| NOMBRE | VARCHAR2(40) | 40 | NOT-NULL | indica el nombre de cada tipo de metodo de pago |
| Descripcion | | | | Fecha de creación |
| Tabla que almacena los datos de los tipos de metodos de pago | | | | 9/5/2022 |

| TABLA PEDIDOS | | | | |
|---------------------------------------------|--------|--------|---------------|---------------------------------------------------|
| Diccionario de datos | Tipo | Tamaño | Restricciones | Funcion |
| ID_PEDIDO | NUMBER | NA | PRIMARY KEY | Almacena el id que identifica a cada pedido |
| ID_USUARIO | NUMBER | NA | FOREIGN KEY | Identifica el usuario al que pertenece el pedido |
| ID_DIRECCION | NUMBER | NA | FOREIGN KEY | Identifica a que direccion se entregara un pedido |
| ID_TIENDA | NUMBER | NA | FOREIGN KEY | Identifica que tienda ha procesado un pedido |
| FECHA | DATE | NA | NOT NULL | indica la fecha en la que se realizo un pedido |
| Descripcion | | | | Fecha de creación |
| Tabla que almacena los datos de los pedidos | | | | 9/5/2022 |

| TABLA CONTIENE | | | | |
|-----------------------------------------------------------------|--------|--------|---------------|---------------------------------------------------------------|
| Diccionario de datos | Tipo | Tamaño | Restricciones | Funcion |
| ID_PEDIDO | NUMBER | NA | FOREIGN KEY | identifica el pedido al que pertenece el producto relacionado |
| ID_PRODUCTO | NUMBER | NA | FOREIGN KEY | identifica el producto contenido por un pedido |
| CANTIDAD | NUMBER | NA | NOT NULL | indica la cantidad del producto adjunta al pedido |
| PRECIOUNITARIO | NUMBER | NA | NOT NULL | indica el precio al que se ha vendido un producto |
| Descripcion | | | | Fecha de creación |
| Tabla que relaciona los datos de las tablas productos y pedidos | | | | 9/5/2022 |

Base de datos: Diccionario de datos

| TABLA FACTURAS | | | | |
|----------------------------------------------|---------------|--------|---------------|-------------------------------------------------------|
| Diccionario de datos | Tipo | Tamaño | Restricciones | Funcion |
| ID_FACTURA | NUMBER | NA | PRIMARY KEY | Almacena el id que identifica a cada factura |
| ID_USUARIO | NUMBER | NA | FOREIGN KEY | Identifica el usuario al que le pertenece la factura |
| ID_PEDIDO | NUMBER | NA | FOREIGN KEY | Identifica el pedido relacionado a la factura |
| ID_METPAGO | NUMBER | NA | FOREIGN KEY | identifica el metodo de pago relacionado a la factura |
| DETALLE | VARCHAR2(100) | 100 | NA | Almacena los detalles extra de la factura |
| PRECIOTOTAL | NUMBER | NA | NOT NULL | Almacena el precio final de la factura |
| Descripcion | | | | Fecha de creación |
| Tabla que almacena los datos de las facturas | | | | 9/5/2022 |

| TABLA PRODUCTOS | | | | |
|-----------------------------------------------|----------------|--------|---------------|---------------------------------------------------------------|
| Diccionario de datos | Tipo | Tamaño | Restricciones | Funcion |
| ID_PRODUCTO | NUMBER | NA | PRIMARY KEY | Almacena el id que identifi a cada producto |
| ID_TIPOPROD | NUMBER | NA | FOREIGN KEY | Identifica el tipo de producto al que pertenece este producto |
| NOMBRE | VARCHAR2(45) | 45 | NOT NULL | Almacena el nombre de cada producto |
| DETALLE | VARCHAR2(100) | 100 | NA | Almacena los detalles de cada producto |
| PRECIO | NUMBER | NA | NOT NULL | Almacena el precio de cada producto |
| STOCK | NUMBER | NA | NA | Almacena el stock de cada producto |
| IMAGENPROD | VARCHAR2(2048) | 2048 | NA | Almacena una url con la imagen de cada producto |
| Descripcion | | | | Fecha de creación |
| Tabla que almacena los datos de los productos | | | | 9/5/2022 |

| TABLA TIPOSPROD | | | | |
|-------------------------------------------------------|--------------|--------|---------------|-------------------------------------------------------|
| Diccionario de datos | Tipo | Tamaño | Restricciones | Funcion |
| ID_TIPOPROD | NUMBER | NA | PRIMARY KEY | Almacena el id que identifica a cada tipo de producto |
| NOMBRE | VARCHAR2(40) | 40 | NOT-NULL | Almacena el nombre de cada tipo de producto |
| Descripcion | | | | Fecha de creación |
| Tabla que almacena los datos de los tipos de producto | | | | 9/5/2022 |

Base de datos: Diccionario de datos

| TABLA CONTIENEPROM | | | | |
|---------------------------------------------------------|--------|--------|---------------|-----------------------------------------|
| Diccionario de datos | Tipo | Tamaño | Restricciones | Funcion |
| ID_FACTURA | NUMBER | NA | FOREING KEY | Identifica la factura con una promocion |
| ID_PROMOCION | NUMBER | NA | FOREING KEY | Identifica la factura con una promocion |
| Descripcion | | | | Fecha de creación |
| Tabla que relaciona las tablas Promociones con facturas | | | | 9/5/2022 |

| TABLA PROMOCIONES | | | | |
|-------------------------------------------------|--------------|--------|---------------|---------------------------------------------------|
| Diccionario de datos | Tipo | Tamaño | Restricciones | Funcion |
| ID_PROMOCION | NUMBER | NA | PRIMARY KEY | Almacena el id que identifica a cada promocion |
| DESCUENTO | NUMBER | NA | NOT NULL | Almacena el descuento aplicado por cada promocion |
| NOMBRE | VARCHAR2(40) | 40 | NOT-NULL | Almacena el nombre de cada promocion |
| Descripcion | | | | Fecha de creación |
| Tabla que almacena los datos de las promociones | | | | 9/5/2022 |

| TABLA TIENDAS | | | | |
|---------------------------------------------|--------------|--------|---------------|-----------------------------------------------------|
| Diccionario de datos | Tipo | Tamaño | Restricciones | Funcion |
| ID_TIENDA | NUMBER | NA | PRIMARY KEY | Almacena el id que identifica a cada tienda |
| DIRECCION | VARCHAR2(70) | 70 | NOT NULL | Almacena la direccion donde se encuentra una tienda |
| MANAGER | NUMBER | NA | NA | Almacena el id_empleado del manager |
| TLF | VARCHAR2(11) | 11 | NA | Almacena el telefono de la tienda |
| Descripcion | | | | Fecha de creación |
| Tabla que almacena los datos de las tiendas | | | | 9/5/2022 |

| TABLA EMPLEADOS | | | | |
|-----------------------------------------------|--------------|--------|---------------|-------------------------------------------------|
| Diccionario de datos | Tipo | Tamaño | Restricciones | Funcion |
| ID_EMPLEADO | NUMBER | NA | PRIMARY KEY | Almacena el id que identifica a cada empleado |
| ID_TIENDA | NUMBER | NA | FOREING KEY | identifica a que tienda pertenece cada empleado |
| NOMBRE | VARCHAR2(40) | 40 | NA | Almacena el nombre de un empleado |
| APELLIDOS | VARCHAR2(60) | 60 | NA | Almacena los apellidos de un empleado |
| CORREO | VARCHAR2(60) | 60 | NA | Almacena el correo de un empleado |
| Descripcion | | | | Fecha de creación |
| Tabla que almacena los datos de los empleados | | | | 9/5/2022 |