



**Universidad Don Bosco**

**FACULTAD DE INGENIERÍA**

**ESCUELA DE INGENIERÍA EN COMPUTACIÓN**

**Lenguajes Interpretados en el Cliente 02T**

**Fase 1 Proyecto de Cátedra**

**Elaborado en Agosto de 2024**

**Integrantes:**

Roberto José Barahona Alemán BA221212

José Ángel Cuellar Flores CF200830

Luis Antonio Méndez Parada MP220885

Priscila Lisseth Osorio Escobar OE210860

Némesis Alejandra Valencia Rivera VR211067

**DOCENTE**

Ingeniera Karens Medrano

**Fecha**

**Domingo 1 de Septiembre de 2024**

**LINK DE REPOSITORIO:**

[https://github.com/VR211067/PROYECTO\\_LIC01T.git](https://github.com/VR211067/PROYECTO_LIC01T.git)

## ÍNDICE

<b>INTRODUCCIÓN.....</b>	<b>3</b>
<b>PERFIL DEL PROYECTO.....</b>	<b>4</b>
<b>DISEÑO UX/UI.....</b>	<b>9</b>
<b>LÓGICA DE PROYECTO.....</b>	<b>18</b>
<b>ARQUITECTURA DE SOFTWARE.....</b>	<b>30</b>
<b>HERRAMIENTAS A UTILIZAR.....</b>	<b>32</b>
<b>CRONOGRAMA.....</b>	<b>34</b>
<b>COSTO.....</b>	<b>35</b>
<b>FUENTES DE CONSULTA.....</b>	<b>37</b>

# INTRODUCCIÓN

En el panorama empresarial actual, la eficacia en la gestión de catálogos de productos se ha convertido en un factor determinante para el éxito y la competitividad de las empresas. Las organizaciones se enfrentan a desafíos significativos al intentar implementar y mantener catálogos de productos digitales que sean a la vez atractivos, funcionales y adaptables a sus necesidades específicas. Entre estos desafíos se encuentran la complejidad en la creación de interfaces visualmente atractivas, la facilidad de navegación para los usuarios y la capacidad de personalización según las características particulares de cada empresa. Además, la gestión de múltiples usuarios con diferentes niveles de acceso y opciones de personalización a menudo resulta un proceso complicado sin las herramientas adecuadas.

La fase 1 del proyecto "Portal Web Personalizado de Catálogo de Productos Online Adaptable a Diversas Empresas" se centra en abordar estos desafíos mediante el desarrollo de una plataforma web integral. Esta plataforma permitirá a las empresas de todos los tamaños e industrias crear, gestionar y personalizar sus catálogos de productos de manera sencilla y eficiente. El objetivo es proporcionar una solución flexible y escalable que facilite la administración de catálogos, mejore la experiencia del usuario y permita una adaptación continua a las necesidades cambiantes del mercado.

Esta primera fase se dedicará a la conceptualización y diseño inicial del portal, estableciendo los fundamentos sobre los cuales se construirán las funcionalidades avanzadas en fases posteriores del proyecto. A través de un análisis detallado de los requisitos y la identificación de las características clave, esta etapa inicial sentará las bases para una solución robusta y eficaz, orientada a optimizar la gestión del catálogo de productos en el entorno digital.

# PERFIL DEL PROYECTO

## Nombre del Proyecto

Portal Web Personalizado de Catálogo de Productos Online Adaptable a Diversas Empresas

## Problema a resolver

En el entorno empresarial actual, la gestión eficaz del catálogo de productos es esencial para cualquier empresa que quiera destacar en un mercado altamente competitivo. Sin embargo, muchas organizaciones enfrentan desafíos importantes al intentar implementar y mantener catálogos de productos efectivos y actualizados en plataformas digitales. Estos desafíos incluyen la complejidad de crear un catálogo que sea visualmente atractivo, fácil de navegar y adaptable a las necesidades específicas de cada negocio. Además, la gestión centralizada de múltiples usuarios con diferentes niveles de acceso y opciones de personalización es un aspecto crítico que muchas veces resulta difícil de gestionar sin las herramientas adecuadas.

El proyecto propuesto es una plataforma web para crear, gestionar y personalizar catálogos de productos para empresas de todos los tamaños e industrias.

La plataforma proporciona una solución integral que aborda los siguientes puntos clave:

- *Flexibilidad y Personalización:* la plataforma está diseñada para ser altamente personalizable, lo que permite a cada empresa adaptar el diseño y la estructura de su catálogo de productos a sus necesidades específicas de marca e industria.
- *Gestión eficiente de catálogos:* facilita la gestión y las actualizaciones centralizadas de productos, garantizando que la información sea precisa y esté siempre actualizada. Los administradores cuentan con herramientas avanzadas para agregar, modificar y eliminar productos de forma rápida y sencilla.
- *Disponibilidad y seguridad:* Proporciona un potente sistema de gestión de usuarios con diferentes niveles de acceso y permisos para garantizar la seguridad y privacidad de la información empresarial confidencial.
- *Interfaz intuitiva y experiencia de usuario:* la plataforma proporciona a los usuarios finales una interfaz intuitiva y amigable que facilita la navegación y la búsqueda de productos, mejorando así la experiencia del cliente y aumentando las tasas de conversión. Compatibilidad y escalabilidad: creada con tecnologías modernas como los frameworks de JavaScript, la plataforma es altamente compatible y se puede ampliar fácilmente para satisfacer las necesidades cambiantes de su negocio a medida que crece.
- *Soporte y mantenimiento continuos:* Brindar soporte técnico continuo y actualizaciones periódicas para garantizar que la plataforma esté siempre optimizada y en línea con las últimas tendencias y tecnologías del mercado.

## Objetivo general

Desarrollar e implementar la operación de una plataforma web innovadora y adaptable que permita a empresas de diversos sectores crear, gestionar y personalizar eficientemente sus catálogos de productos, mejorando así la visibilidad de sus productos en el mercado digital y optimizando la experiencia del usuario final.

## Metas:

Las metas esperadas como equipo son las siguientes:

- Aumentar la visibilidad online.
- Incrementar la conversión de visitantes a clientes
- Aumentar la interacción de consultas y contactos
- Optimizar el catálogo de productos
- Cumplir los estándares de seguridad y privacidad
- Promover la tienda física.

## Resultados esperados

Los resultados esperados para nuestra página web son los siguientes:

- Exposición de productos: Mostrar de manera clara y atractiva los productos ofrecidos y disponibles en categorías para facilitar la navegación al usuario,
- Información Detallada: Proporcionar descripciones completas de los productos ofrecidos con especificaciones técnicas y detalles relevantes.
- Imágenes de buena calidad: Utilizar imágenes de alta calidad de nuestros trabajos realizados anteriormente para ser utilizados como ejemplos.
- Fácil navegación: Garantizar una navegación intuitiva y fácil de usar para el usuario tales como un menú y opciones de búsqueda.
- Feedback y opiniones: Incluir una sección donde los clientes puedan dejar reseñas o comentarios para poder tener la confianza de futuros clientes.
- Actualización: Mantener el catálogo actualizado con productos nuevos o con productos que ya proveemos.

## Metodología

La metodología que se implementará será: Kanban

Kanban es una metodología que traerá beneficios en el proyecto entre los cuales podemos mencionar:

- Una visualización más clara ya que permite ver el estado en que está el proyecto en desarrollo y facilita el poder identificar las tareas que están pendientes y en proceso de realizarlas.
- Control del flujo de trabajo ayudan a evitar una sobrecarga de tareas y ayuda a mantener un enfoque en las más importantes.
- La revisión periódica ayudará a la eficiencia del equipo a tener una mejora continua.
- Implementar kanban ayudará a gestionar el trabajo de forma más eficiente pudiendo reducir tiempo y asegurar una mejora en cada avance del proyecto.

Entre los roles que se considerarán asignar se encuentran:

- **Product Owner:** Némesis Valencia, encargada de definir y priorizar las tareas del backlog del proyecto, asegurándose de que el equipo esté trabajando en las cosas más importantes para los stakeholders. Su responsabilidad incluye recopilar requisitos, mantener la visión del proyecto y gestionar las expectativas del cliente.
- **Diseñador UI/UX:** José Cuellar y Priscila Osorio, se encargarán del diseño de la interfaz y la experiencia del usuario. Su trabajo incluye la creación de wireframes, prototipos y diseño visual. Debe asegurarse de que la página sea atractiva, intuitiva y fácil de usar.
- **Desarrollador Front-End:** Roberto Barahona, responsable de convertir los diseños del UI/UX en código. Trabaja principalmente con JavaScript para construir la parte visual de la página web, asegurándose de que sea responsiva y funcional en diferentes dispositivos y navegadores.
- **Desarrollador Back-End:** Luis Mendez, y Némesis Valencia encargados de la lógica del servidor, base de datos y la integración de APIs., asegura que el sitio web tenga la funcionalidad necesaria, incluyendo la gestión de usuarios, la seguridad y la optimización del rendimiento del servidor.
- **Facilitador Kanban:** Priscila Osorio, se encarga de gestionar el flujo de trabajo y mejorar continuamente el proceso. Facilita las reuniones semanales y revisiones, asegurando que el equipo esté alineado y trabajando de manera eficiente.

### Implementación en Kanban

- **Tablero Kanban:** Crear un tablero con columnas como "Backlog", "Por hacer", "WIP", "En revisión" y "Completado".
- **Tarjetas de tareas:** Cada tarea del proyecto se representará con una tarjeta que se mueve a través de las columnas del tablero.
- **Límites de trabajo en progreso (WIP):** Establecer límites en la cantidad de trabajo que puede estar en progreso simultáneamente para evitar la sobrecarga.
- **Reuniones semanales:** Realizar reuniones breves para revisar el progreso y resolver problemas.
- **Revisiones y retrospectivas:** Al final de cada avance, realizar revisiones para evaluar el trabajo completado y retrospectivas para identificar áreas de mejora.

### Área geográfica y Beneficiarios del proyecto

El proyecto se llevará a cabo en empresas del Área de la Capital de la localidad de San Salvador. San Salvador está ubicado en la zona central del país El Salvador. Su elevación se encuentra entre 600 y 1000 metros sobre el nivel del mar. Limita al norte con los municipios de Nejapa, Mejicanos, Cuscatancingo, y Ciudad Delgado, al este con Soyapango y San Marcos, al sur con Panchimalco y también con San Marcos, y al oeste con Antiguo Cuscatlán y Santa Tecla.

Relevancia para el Proyecto: Área de la Capital de San Salvador ha sido seleccionada para este proyecto debido a su ubicación estratégica en una zona céntrica clave del país.

### **Beneficiarios:**

El proyecto tiene como objetivo proporcionar una colección organizada y actualizada de productos y servicios que proporcione cada empresa. Este catálogo facilitará el acceso a productos de calidad, generando ahorro de tiempo.

Los principales beneficiarios serán las empresas que deseen incorporar un servicio en línea para que sus clientes puedan tener una mejor experiencia.

## **Equipo Participante**

### **Gerencia**

- Roberto Alemán
- Angel Cuellar
- Luis Méndez
- Priscila Osorio
- Némesis Valencia

## **Posibles Fuentes de Financiamiento**

### **Internas**

- Capital Propio
- Ganancias generadas por la empresa

### **Externas**

- Bancos
- Inversionistas
- Organizaciones no gubernamentales
- Venta de Acciones

## **Factores críticos de éxito**

- Falta de colaboración y compromiso del equipo

Riesgo: la falta de compromiso de parte de los miembros puede llevar a un trabajo no desigual y generar problemas entre el equipo.

Propuesta para solventarlo: definir roles y responsabilidades claras para cada miembro. También inculcar un ambiente positivo y motivador.

- Problemas de seguridad

Riesgo: Vulnerabilidades de seguridad en el sitio web pueden resultar en ataques cibernéticos, pérdida de datos o compromisos de información sensible de los productos.

Propuesta para solventarlo: Implementar prácticas de desarrollo seguro, realizar auditorías de seguridad y pruebas constantemente.

- Problemas de compatibilidad con navegadores y dispositivos

Riesgo: El sitio web puede no funcionar correctamente en todos los navegadores y dispositivos, lo que afecta la experiencia del usuario.

Propuesta para solventarlo: Realizar pruebas múltiples en diferentes navegadores y dispositivos desde las primeras etapas del desarrollo. Utilizar frameworks y bibliotecas que faciliten la compatibilidad y la adaptabilidad.

- Falta de habilidades o experiencia en el equipo de desarrollo

Riesgo: Que el equipo de desarrollo no cuente con las habilidades necesarias puede producir un sitio web de baja calidad.

Propuesta para solventarlo: Asegurarnos de que el equipo de desarrollo tenga la experiencia y las habilidades requeridas. De lo contrario capacitar a los miembros con la habilidad que necesiten y estar actualizándose constantemente.

- Problemas de comunicación dentro del equipo

Riesgo: La falta de comunicación clara y efectiva entre los miembros puede llevar a malentendidos, errores y retrasos.

Propuesta para solventarlo: Establecer canales de comunicación claros y regulares. Realizar reuniones de seguimiento frecuentes y documentar todas las decisiones y cambios importantes.



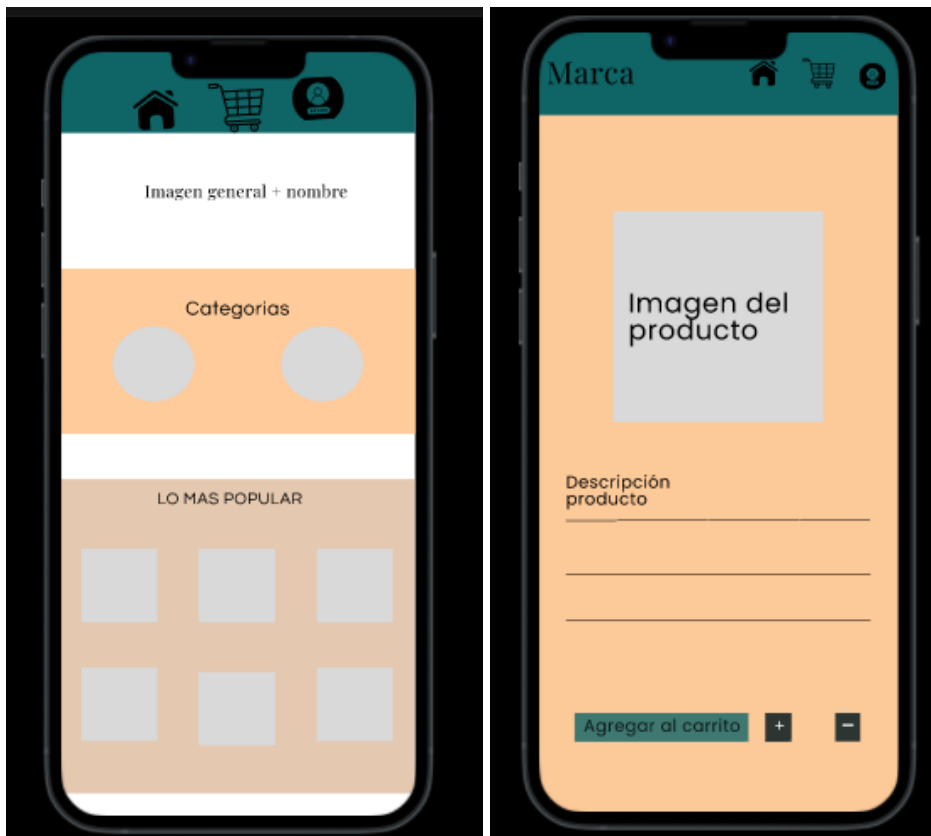
# DISEÑO UX/UI

## Investigación del Usuario

- **Perfiles de Usuario:**
  - **Administrador:** Usuario con acceso completo al sistema. Puede gestionar todos los aspectos del portal, incluyendo la administración de usuarios, modificaciones de inventario y configuración general.
  - **Empleado:** Usuario con permisos limitados. Puede revisar perfiles, actualizar inventarios y verificar feedback, pero no tiene acceso completo a todas las funciones del sistema.
  - **Usuario Final:** Usuario con permisos básicos para crear y modificar su perfil, visualizar categorías e inventarios, y hacer pedidos, aunque no puede realizar compras directamente.
- **Objetivos de Usuario:**
  - **Administrador:** Eficiencia en la gestión de usuarios y productos. Necesita herramientas intuitivas para modificar el inventario, gestionar perfiles y revisar la configuración general del sistema.
  - **Empleado:** Facilidad en la actualización de inventarios y revisión de feedback. Necesita una interfaz sencilla para realizar estas tareas de manera eficiente.
  - **Usuario Final:** Acceso fácil para crear y gestionar su perfil, explorar productos y realizar pedidos sin complicaciones.
- **Arquitectura de la Información**
  - **Contenido y Estructura:**
    - **Productos:** Información detallada sobre cada producto, incluyendo nombre, descripción, imagen y precio.
    - **Categorías:** Agrupaciones de productos que facilitan la navegación (por ejemplo, Electrónica, Ropa, Accesorios).
    - **Descripciones:** Textos que proporcionan detalles sobre cada producto, incluyendo características y beneficios.
    - **Imágenes:** Visuales de alta calidad que muestran los productos desde diferentes ángulos.
    - **Precios:** Información sobre el costo de cada producto.
- **Jerarquía de Información:**
  - **Categorías Principales:** Las secciones principales del catálogo (por ejemplo, Electrónica, Ropa, etc.).
  - **Subcategorías:** Detalles dentro de las categorías principales (por ejemplo, dentro de Electrónica: Teléfonos, Computadoras, etc.).

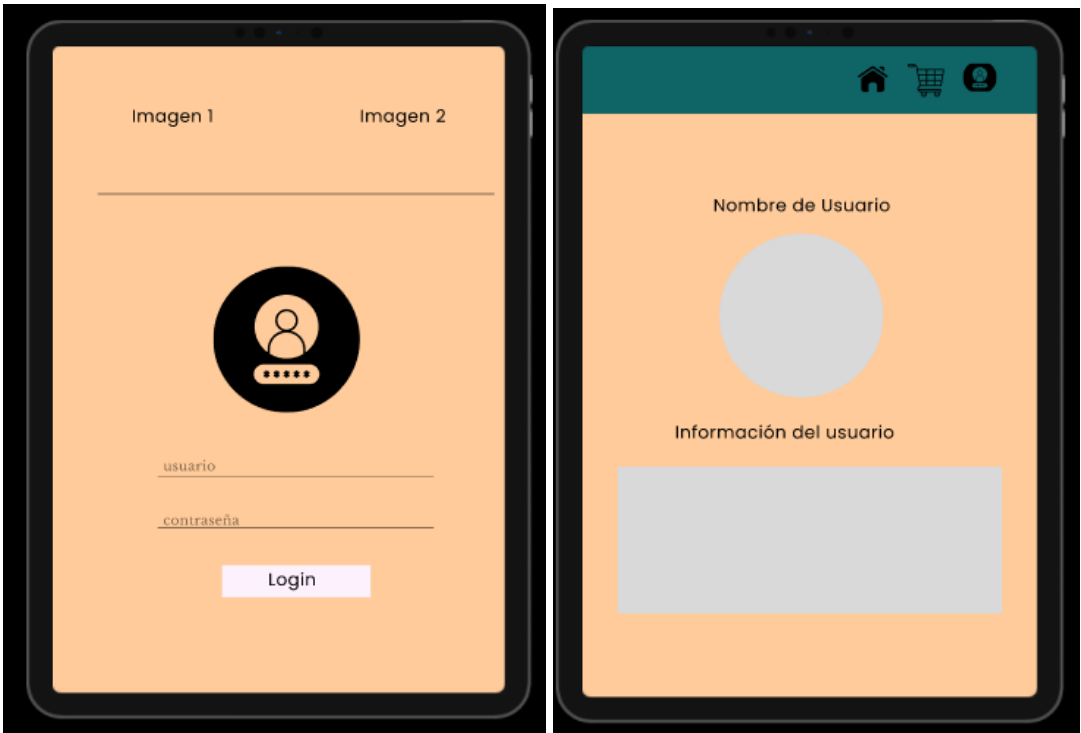
# MOCKUPS

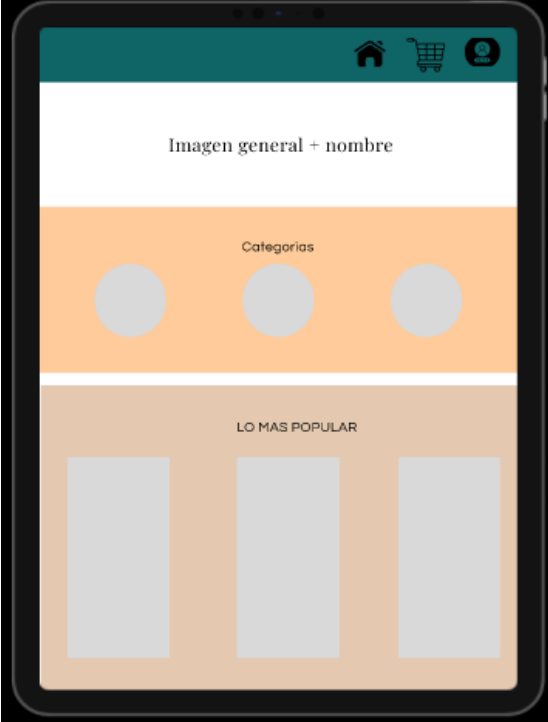
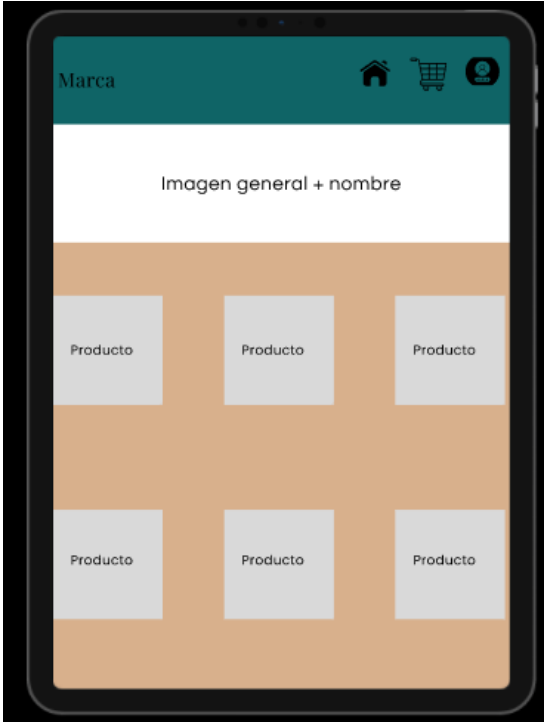
## Móvil



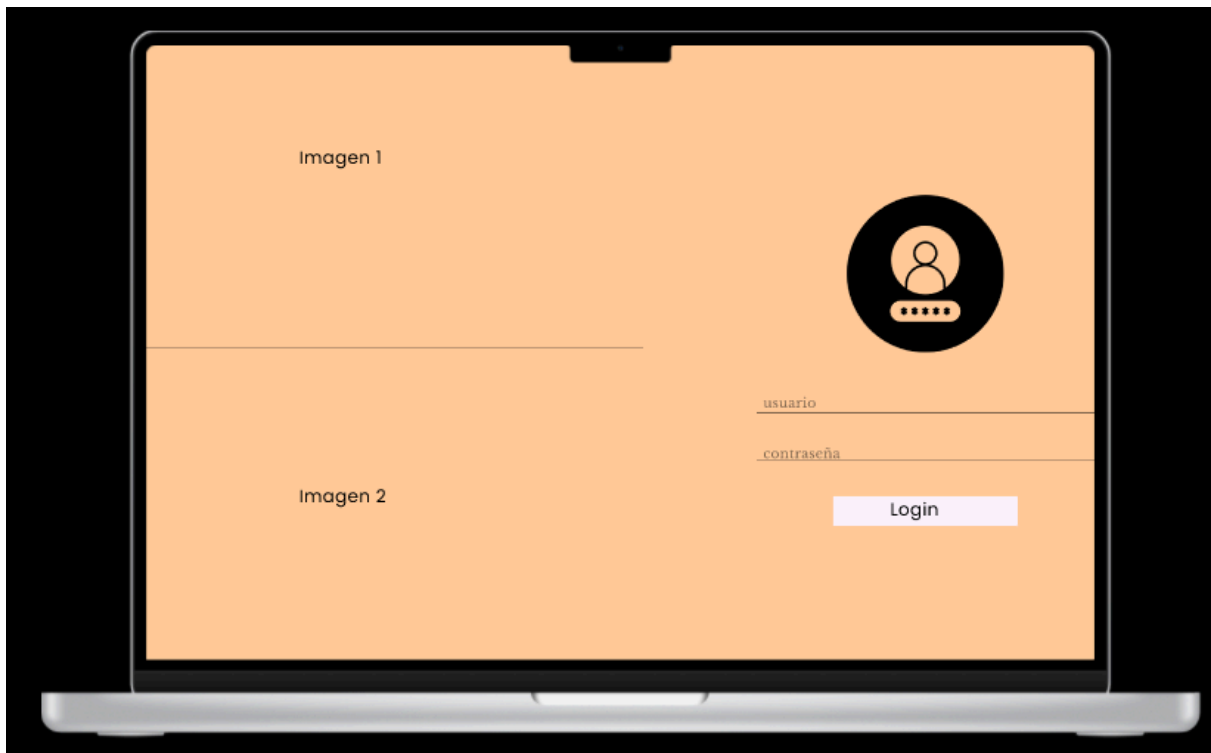


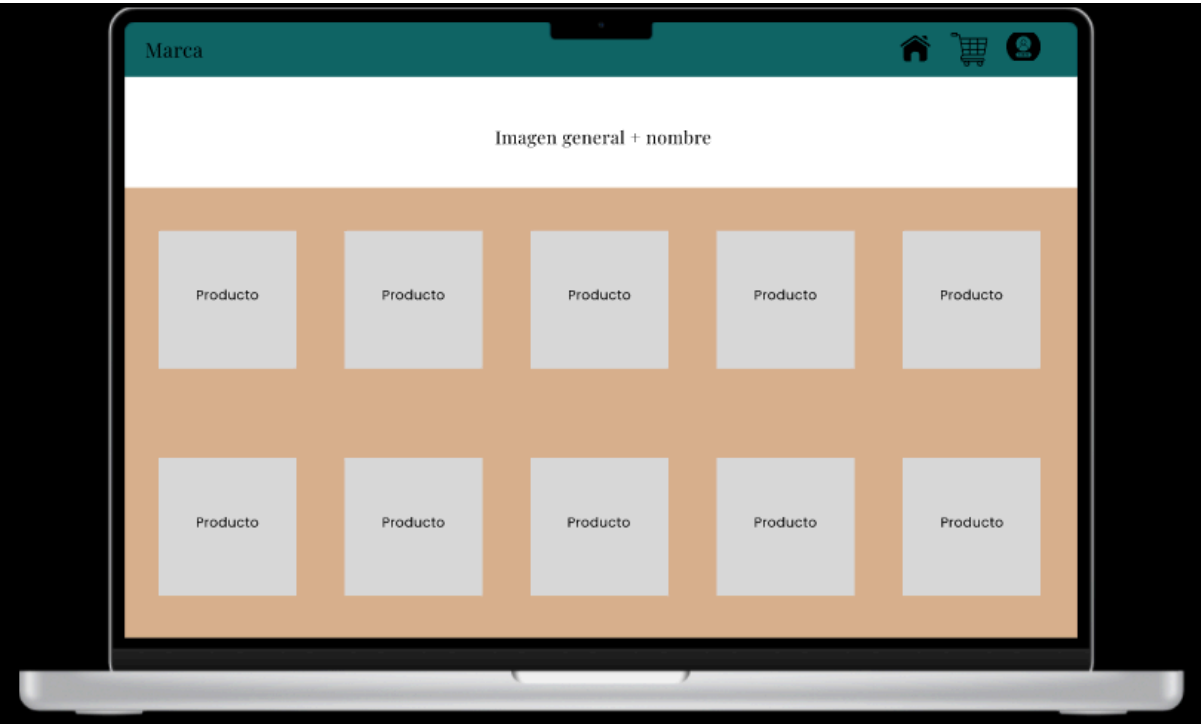
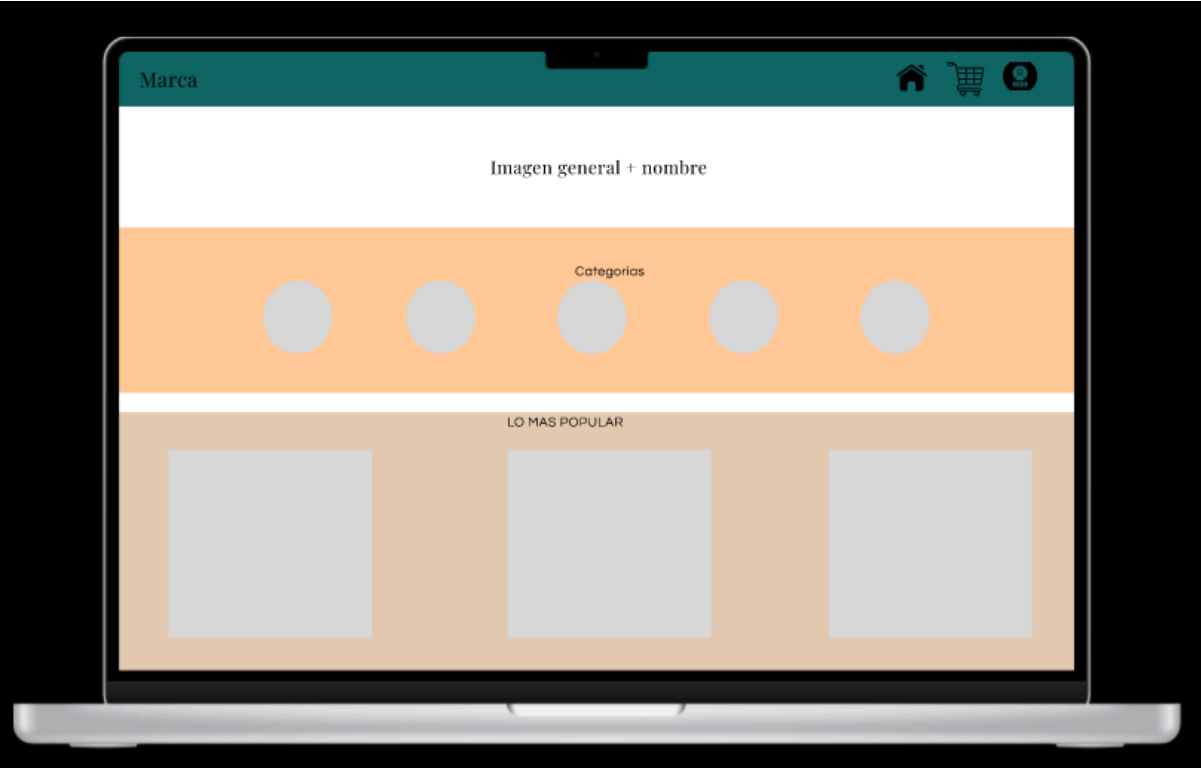
**TABLET**

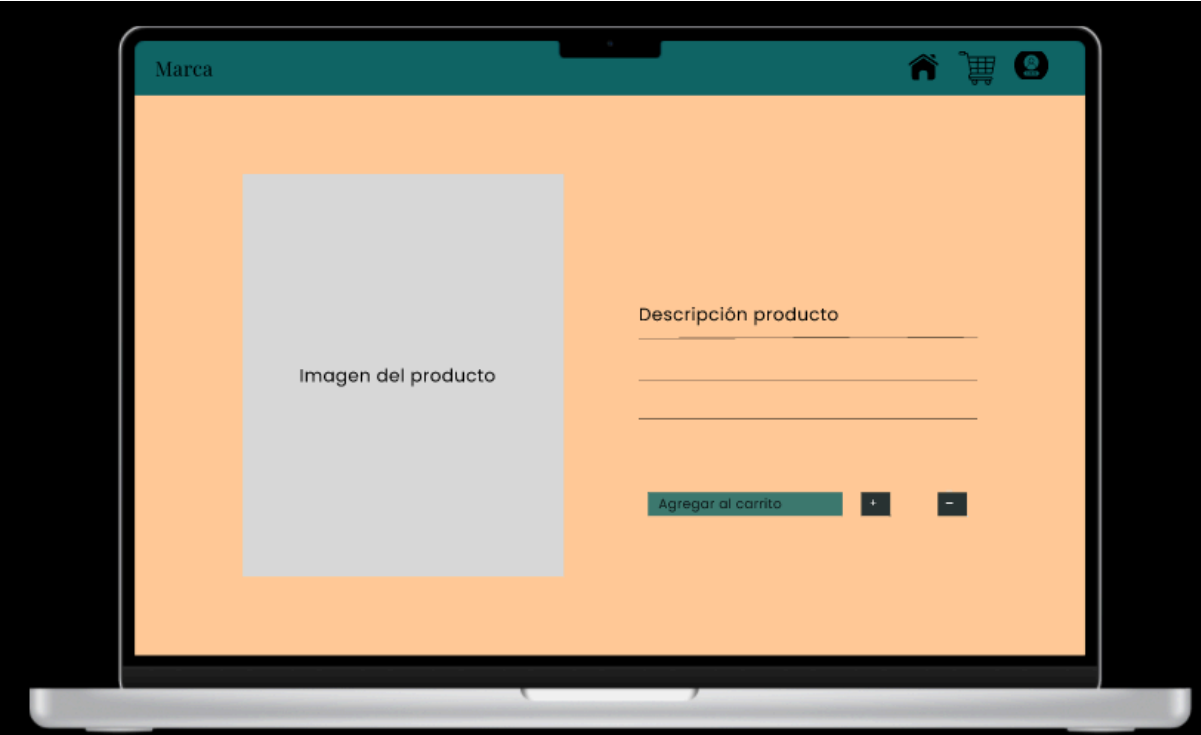




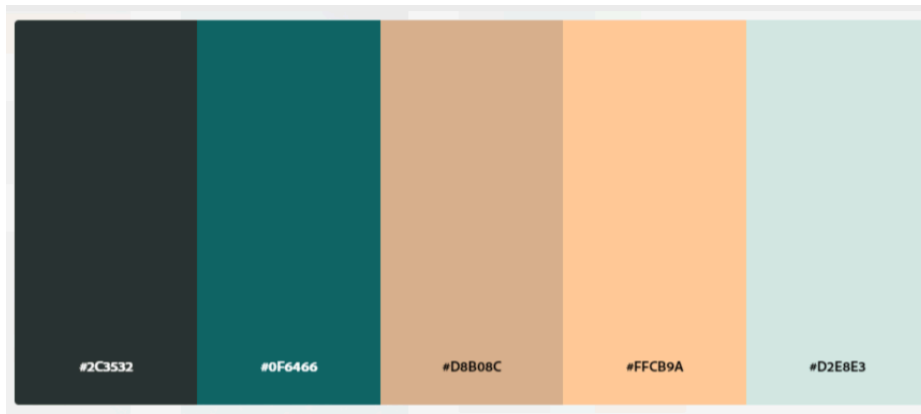
## DESKTOP







- **Paleta de colores:**



- **Iconografía**



- **Tipografía:**

Título	Playfair Display	“Fase 1”
Párrafos	Questrial	“Fase 1”
Botones	Poppins	“Fase 1”
Otros textos	Libre Baskerville	“Fase 1”

- **Imágenes:**

Se pretende mantener una experiencia de usuario equilibrada y eficiente, las imágenes utilizadas en la página web tendrán un tamaño optimizado. Estas imágenes serán de tamaño suficiente para mostrar el producto de manera clara, pero no tan grandes como para distraer de otros elementos importantes de la página, como la información textual o la acción a realizar.

El objetivo es asegurar que las imágenes complementan el objetivo de la página manteniendo un diseño visual limpio y esencial.



- **Navegador Web:**

El sitio web implementará dos tipos de navegación para asegurar una experiencia de usuario fluida y adaptable:

- **Navegación Vertical:** Este tipo de navegación se mostrará en un panel lateral, con enlaces alineados verticalmente. Su función es proporcionar acceso claro y directo a las distintas secciones del sitio
- **Navegación Oculta:** Además, se utilizará una navegación oculta, que estará representada por un ícono (generalmente tres líneas) que se ubicará en la parte superior de la pantalla. Al hacer clic en este ícono, el menú se desplegará, mostrando las opciones de navegación disponibles.

- **Diseño de Pantallas:**

Este diseño será responsivo para que al visualizar con cualquier dispositivo la experiencia del usuario sea agradable garantizando que todos los elementos, incluidas las imágenes, se adapten a cualquier dispositivo en el que se visualice la página. De esta manera, se asegura una experiencia de usuario fluida y agradable.

# LÓGICA DE PROYECTO

- **Visión General:**

El proyecto es una plataforma web para crear, gestionar y personalizar catálogos de productos para diversas empresas. Utilizando Express.js para el backend, la plataforma permitirá a las empresas configurar y mantener catálogos adaptados a sus necesidades específicas, mientras que MySQL gestionará los datos de productos, usuarios y configuraciones.

- **Componentes Principales:**

- Frontend: Interfaz de usuario construida con tecnologías web modernas (HTML, CSS, JavaScript). El frontend interactúa con el backend a través de una API RESTful para mostrar y manipular los datos del catálogo.
- Backend: Implementado con Express.js y Node.js, gestionará la lógica de negocio, la autenticación de usuarios y la comunicación con la base de datos MySQL.
- Base de Datos: MySQL se utilizará para almacenar información sobre productos, usuarios, empresas y configuraciones del catálogo. MySQL se gestionará mediante el paquete mysql2 o sequelize para facilitar las operaciones CRUD.

- **Gestión de Productos:**

- Endpoints CRUD:
  - Crear: POST /products para agregar nuevos productos.
  - Leer: GET /products para obtener una lista de productos y GET /products/:id para obtener detalles de un producto específico.
  - Actualizar: PUT /products/:id para modificar un producto existente.
  - Eliminar: DELETE /products/:id para eliminar un producto.
- Visualización: Los productos se presentarán en una interfaz atractiva con opciones para buscar, filtrar y ordenar.

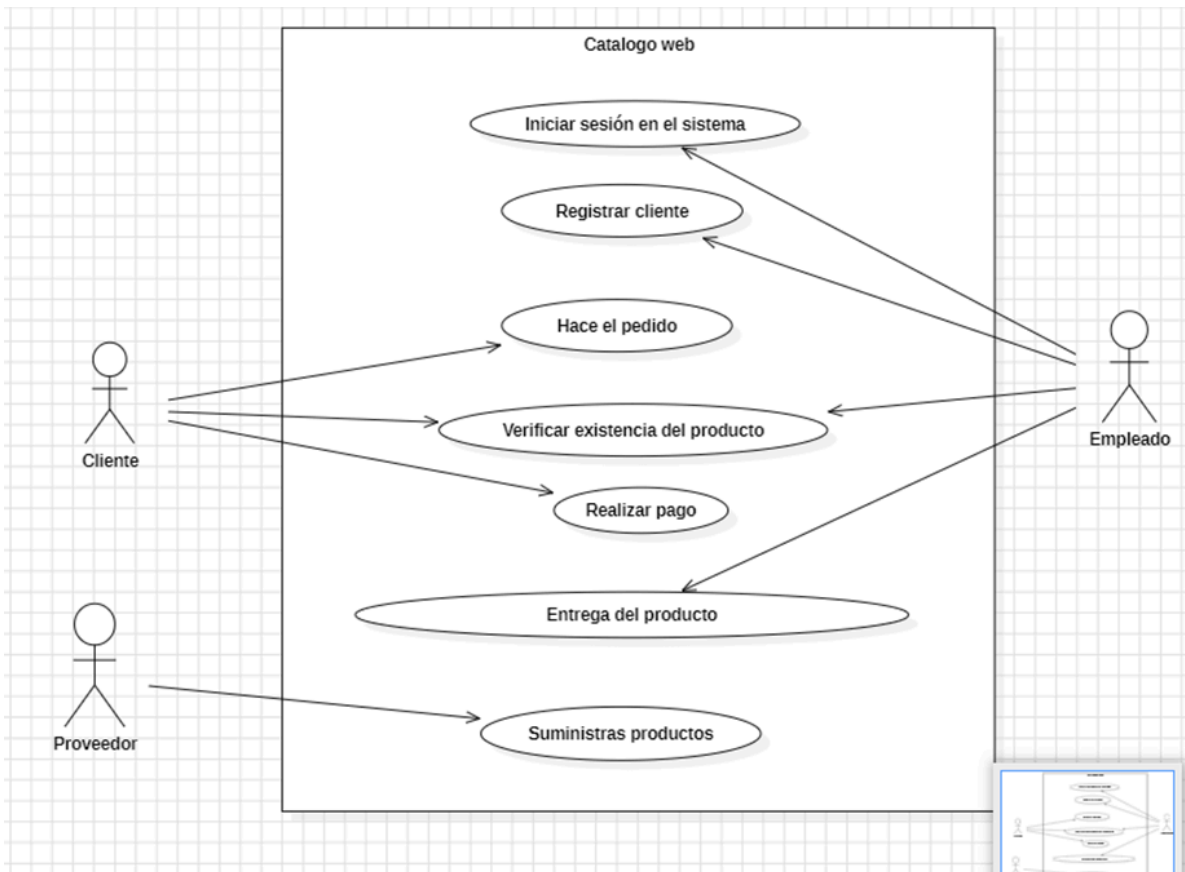
- **Gestión de Empresas y Personalización:**

- Adaptabilidad: Permitir la personalización del catálogo para cada empresa, incluyendo la carga de logotipos, la selección de colores y temas.
- Configuraciones: Los administradores podrán ajustar la estructura del catálogo y otras configuraciones desde un panel de administración.

- **Gestión de Usuarios:**

- Roles y Permisos:
  - Administradores: Gestionan productos, usuarios y configuraciones.
  - Gestores de Productos: Añaden y actualizan productos.
  - Usuarios Finales: Acceden y visualizan el catálogo.
- Autenticación:
  - Implementación de JWT (JSON Web Tokens) para manejar el inicio de sesión y la autorización.
  - Endpoints de Autenticación:
    - POST /auth/login para el inicio de sesión.

- POST /auth/register para el registro de nuevos usuarios.
- **Interfaz de Usuario y Experiencia (UI/UX):**
  - Diseño Responsivo: Asegurar que la plataforma funcione correctamente en distintos dispositivos.
  - Navegación Intuitiva: Implementar una navegación clara y fácil de usar para mejorar la experiencia del usuario.
- **Seguridad y Mantenimiento:**
  - Protección de Datos
  - Validaciones y Sanitización:
  - Manejo de Errores
- **Servidor IIS**
- **Diagramas UML**

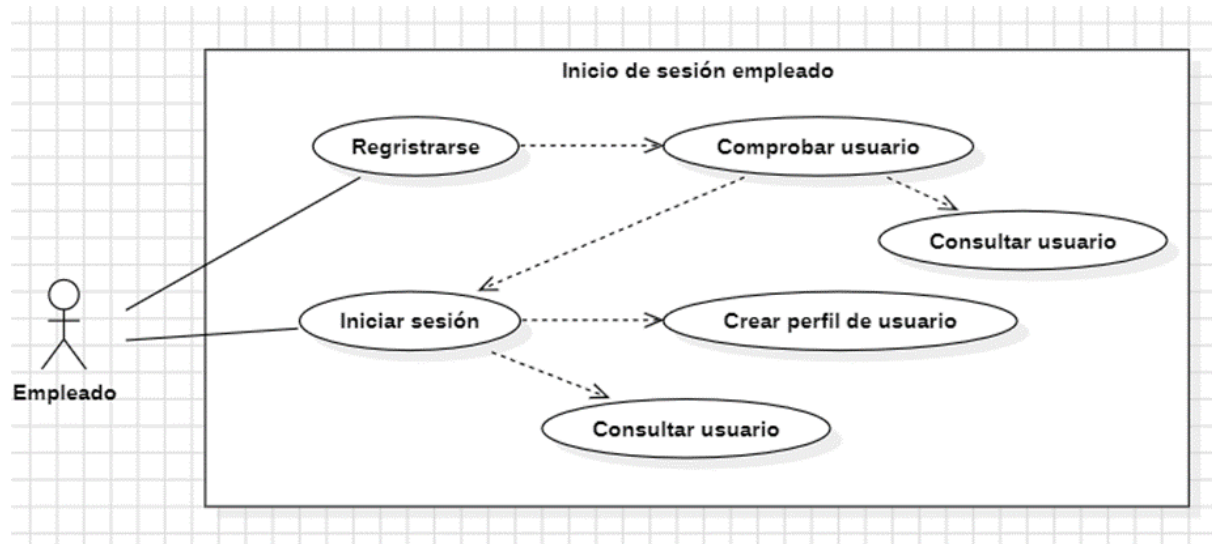


### **Iniciar sesión**

**Actor: Empleado**

**Caso de uso:**

- Iniciar sesión
- Registrar usuario
- Consultar usuario
- Eliminar usuario
- Actualizar catálogo

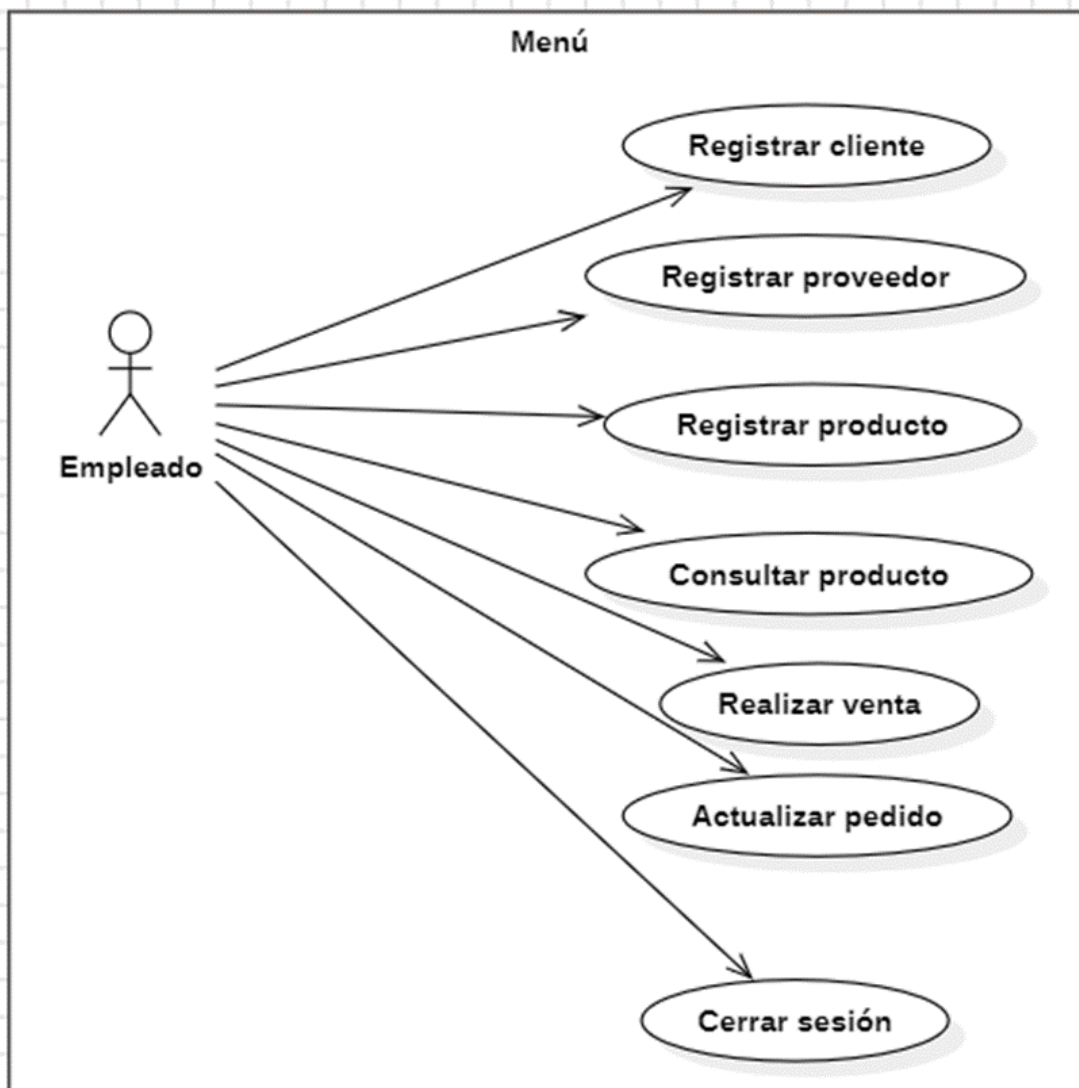


## Menú

**Actor: Empleado**

**Caso de uso:**

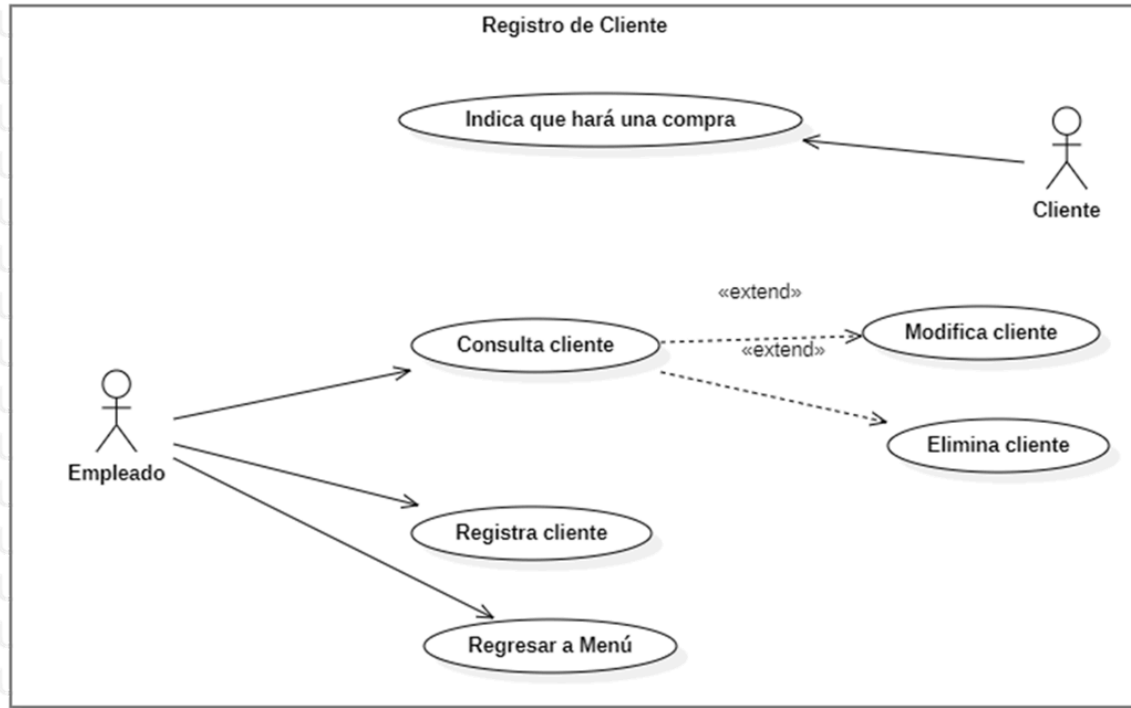
- Iniciar sesión
- Registrar usuario
- Consultar usuario
- Eliminar usuario
- Actualizar pedido



**Actor: Empleado y Cliente.**

**Caso de uso:**

- Registrar cliente
- Consultar cliente
- Eliminar cliente
- Modificar cliente
- Regresar al menú

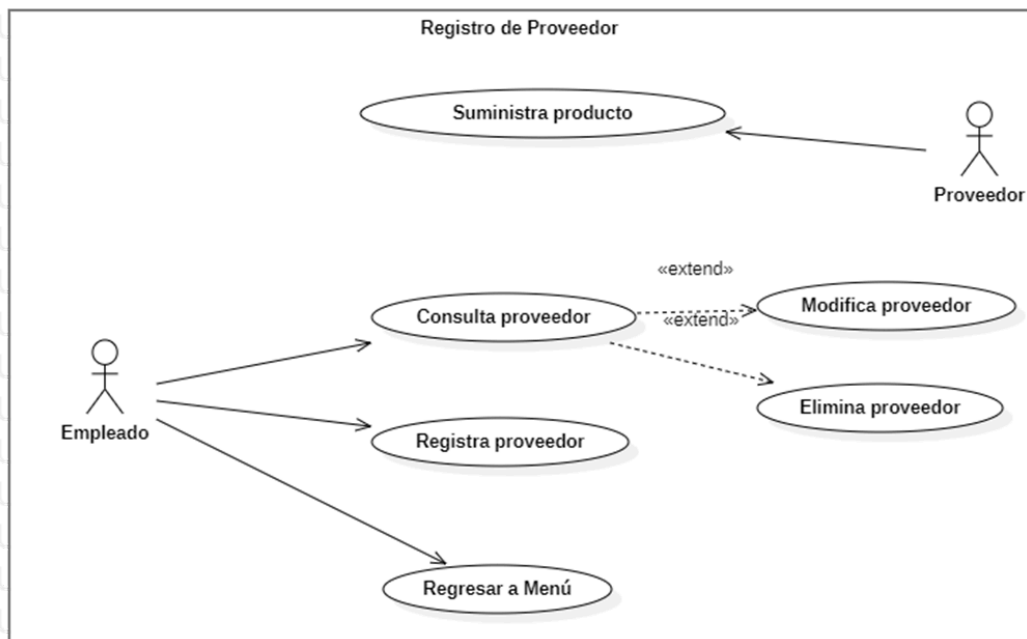


## Registrar Proveedor

**Actor: Empleado y Proveedor.**

**Caso de uso:**

- Registrar proveedor
- Consultar proveedor
- Eliminar proveedor
- Modificar proveedor.
- Regresar al menú

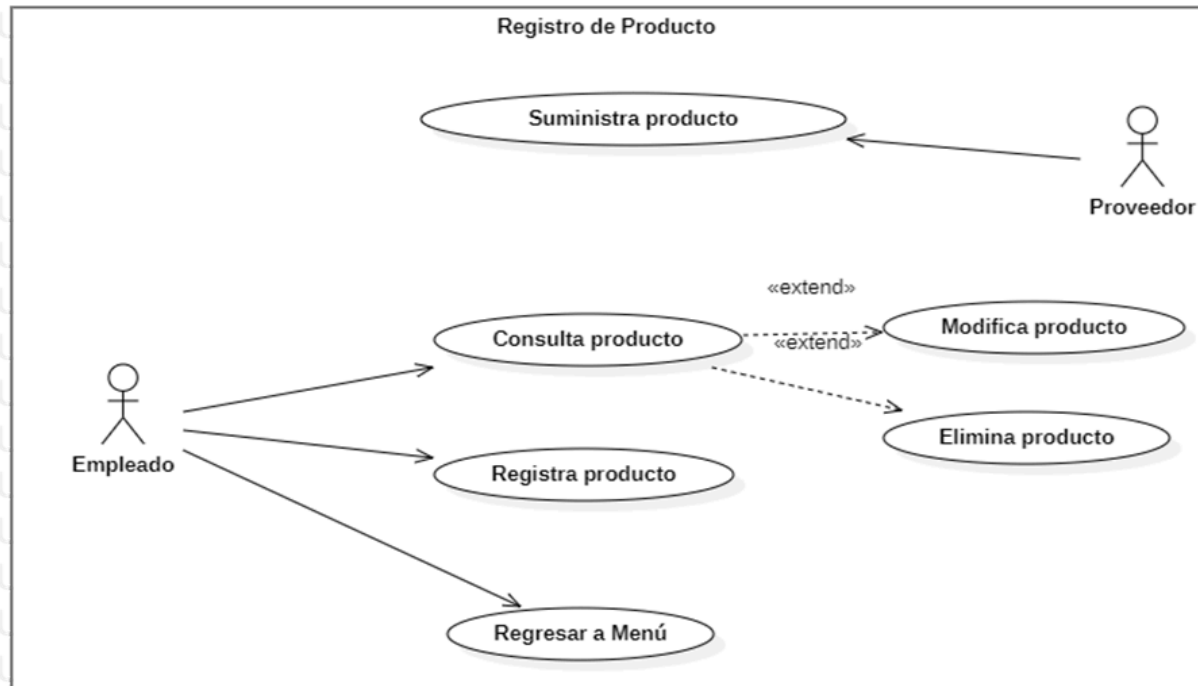


## Registrar Producto

**Actor: Empleado y Proveedor.**

**Caso de uso:**

- Registrar producto
- Consultar producto
- Eliminar producto
- Modificar producto
- Regresar al menú

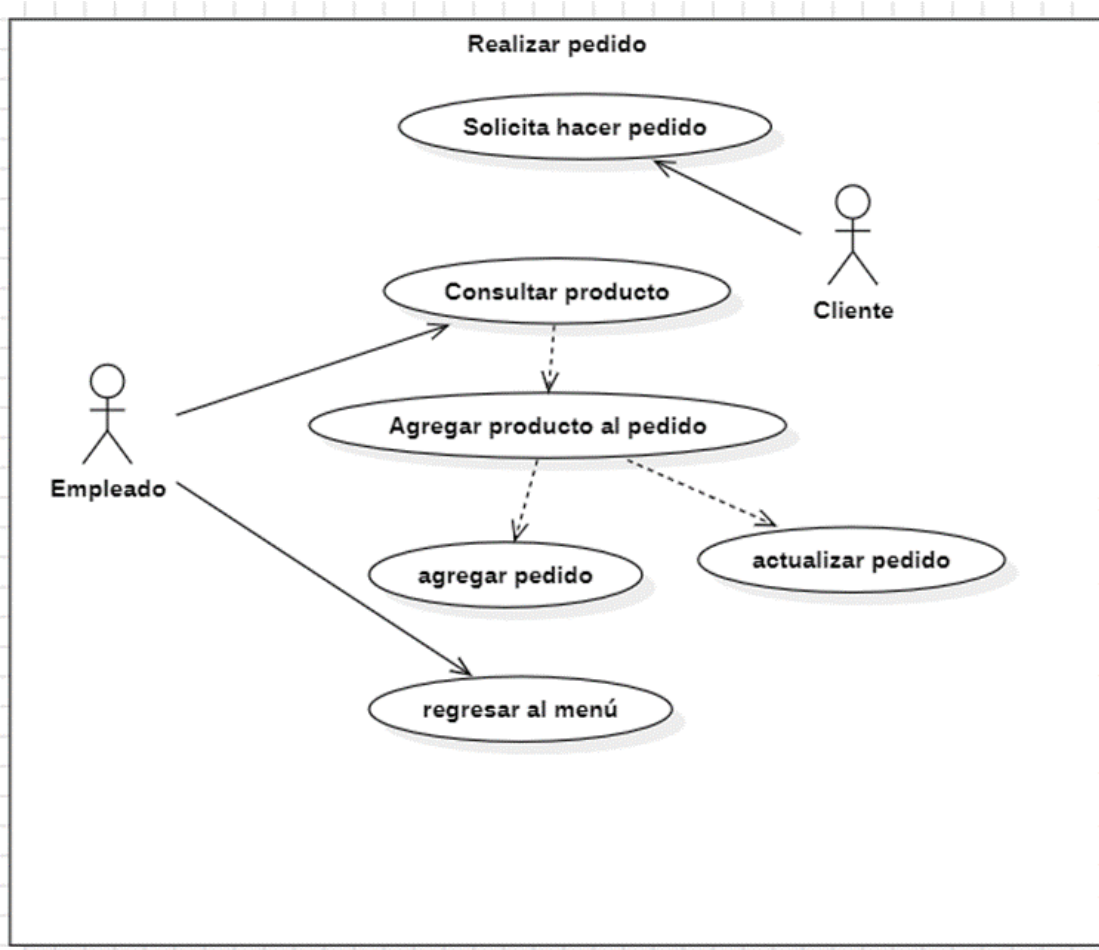


## Realizar Pedido

**Actor: Empleado y Cliente.**

**Caso de uso:**

- Crear nueva factura
- Agregar venta
- Eliminar venta
- Regresar al menú



### Subsistemas de análisis:

#### Subsistema de acceso

Dentro de este subsistema los usuarios validaron sus perfiles previamente creados o se crearán dichos perfiles en caso de que no estén registrados.

Si el empleado o gerente ingresa su usuario y su contraseña, el sistema buscará en la base de datos los registros de dichas credenciales y en caso de no existir se notificará que las credenciales ingresadas son incorrectas por lo que debe proceder a registrarse.

Al registrarse, deberá llenar una serie de campos los cuales después de registrarse pueden ser modificados y consultados.

En caso de que las credenciales sean válidas, el usuario será redirigido al menú y tendrá acceso a todas las tareas que permitirá realizar el sistema, luego de finalizar las tareas puede cerrar sesión



### **Subsistema de Registro de clientes:**

Cuando un cliente indique que realizará una compra, el empleado que previamente inició sesión procede a entrar a la interfaz de registros de cliente, en la cual procederá a consultar la existencia del cliente en el registro de clientes que se tiene.

Si el cliente ya ha sido registrado procede a salir de la interfaz y se redirige al menú para ingresar al próximo subsistema que es “Realizar una venta”.

En el caso que el cliente no esté registrado, el usuario selecciona la opción de “agregar cliente” y procede a llenar los campos requeridos con la información brindada por el cliente.

Si el cliente ya existía en el registro, pero solicita actualizar su información el usuario busca al cliente por el ID que el sistema genera automáticamente y selecciona al cliente y procede a modificar la información para luego guardar el registro y poder utilizar la nueva información.

Si se necesita eliminar por completo la información de un cliente, el usuario debe seleccionar la opción de eliminar cliente.

### **Subsistema de Registro de Proveedor:**

Cuando se va a adquirir productos de un proveedor, el empleado inicia sesión en el sistema y procede a ingresar a la interfaz de registro de proveedor, y consulta la existencia del proveedor dentro del registro existente de proveedores, en caso de que el proveedor ya exista debe regresar al menú e ingresar a la interfaz de registrar producto y en el caso que el proveedor necesite actualizar su información el usuario procede a modificar la información existente.

Si el usuario no existe en el registro de proveedores, el usuario procede a seleccionar la opción “Agregar proveedor” y guarda la información en la base de datos.

Si se necesita eliminar por completo la información de un proveedor, el usuario consulta a dicho proveedor por medio de su ID y selecciona la opción de “Eliminar proveedor”

### **Subsistema de Registro de Producto:**

Cuando se va a adquirir productos de un proveedor, el empleado que previamente inició sesión procede a entrar a la interfaz de registro de producto, en la cual procederá a consultar la existencia del producto en el registro de productos que se tiene.

Si el producto ya ha sido registrado procede a salir de la interfaz y se redirige al menú para ingresar al próximo subsistema que es “Consultar inventario”.

En el caso que el producto no esté registrado, el usuario selecciona la opción de “agregar producto” y procede a llenar los campos requeridos con la información brindada por el proveedor.

Si el producto ya existía en el registro, pero se solicita actualizar su información el usuario busca al producto por el ID que el sistema genera automáticamente y selecciona al producto y procede a modificar la información para luego guardar el registro y poder utilizar la nueva información.

Si se necesita eliminar por completo la información de un producto, el usuario debe seleccionar la opción de eliminar producto.

## Análisis de clases

El diagrama que integra todo nuestro sistema aparece los actores que van a intervenir a lo largo de la vida útil del sitio web los cuales detallaremos con sus propios métodos y atributos.

•**Clase Empleado:** Esta contendrá los objetos datos del empleado relativos a cada uno de los permitidos para el acceso.

Empleado
+id_empleado +nombre +apellido +fecha_nacimiento +fecha_inicio_trabajo +correo_electronico +usuario +contraseña +telefono_1 +telefono_2 +dui +salario +nit +direccion +Attribute1
+registrar_pedido() +registrar_compra() +editar_compra() +entregar_pedidos() +actualizar_productos() +recibir_productos()

### Atributos:

- id\_empleado
- nombre
- apellido
- fecha\_nacimiento
- fecha\_inicio\_trabajo
- correo\_electronico
- usuario
- contraseña
- telefono\_1
- telefono\_2
- dui
- salario
- nit
- dirección

**Clase Clientes:** Permitirá poder almacenar los datos del cliente para poder llevar su registro en la empresa

Cliente
+id_cliente +nombre_cli +fecha_nacimiento +correo_electronico +usuario +contraseña +telefono_1 +telefono_2 +dirección +Attribute1
+consultar_pro() +hacer_pedido() +cancelar_pedido() +agregar_datos()

### Atributos:

- id\_cliente
- nombre\_cli
- fecha\_nacimiento
- correo\_electronico
- usuario
- contraseña
- telefono\_1
- telefono\_2
- dirección

**Clase Proveedores:** Permitirá tener a la mano la lista de proveedores con los que ya se cuenta para saber cuál elegir.

proveedor
+id_cliente +nombre_contacto +apellido_contacto +fecha_nac +nombre_empresa +telefono_contacto +telefono_empresa +correo_empresa +correo_contacto +dirección_empresa +Attribute1
+consultar_producto() +suministrar_productos() +entregar_productos()

**Atributos:**

- id\_cliente
- nombre\_contacto
- apellido\_contacto
- fecha\_nac
- nombre\_empresa
- telefono\_contacto
- telefono\_empresa
- correo\_empresa
- correo\_contacto
- dirección\_empresa

**Clase Producto:** Esta permitirá llevar un control de los productos para un mejor manejo a la hora de seleccionarlos.

Producto
+id_producto +nombre_producto +descripción_producto +cantidadTotal +precio_compra +precio_venta_unidad +fecha_vencimiento +Attribute1

**Atributos:**

- id\_producto
- nombre\_producto
- descripcion\_producto
- cantidadTotal
- precio:compra
- precio\_venta\_unidad
- fecha\_vencimiento

**Clase Pedido:** Para un control de historial de pedidos

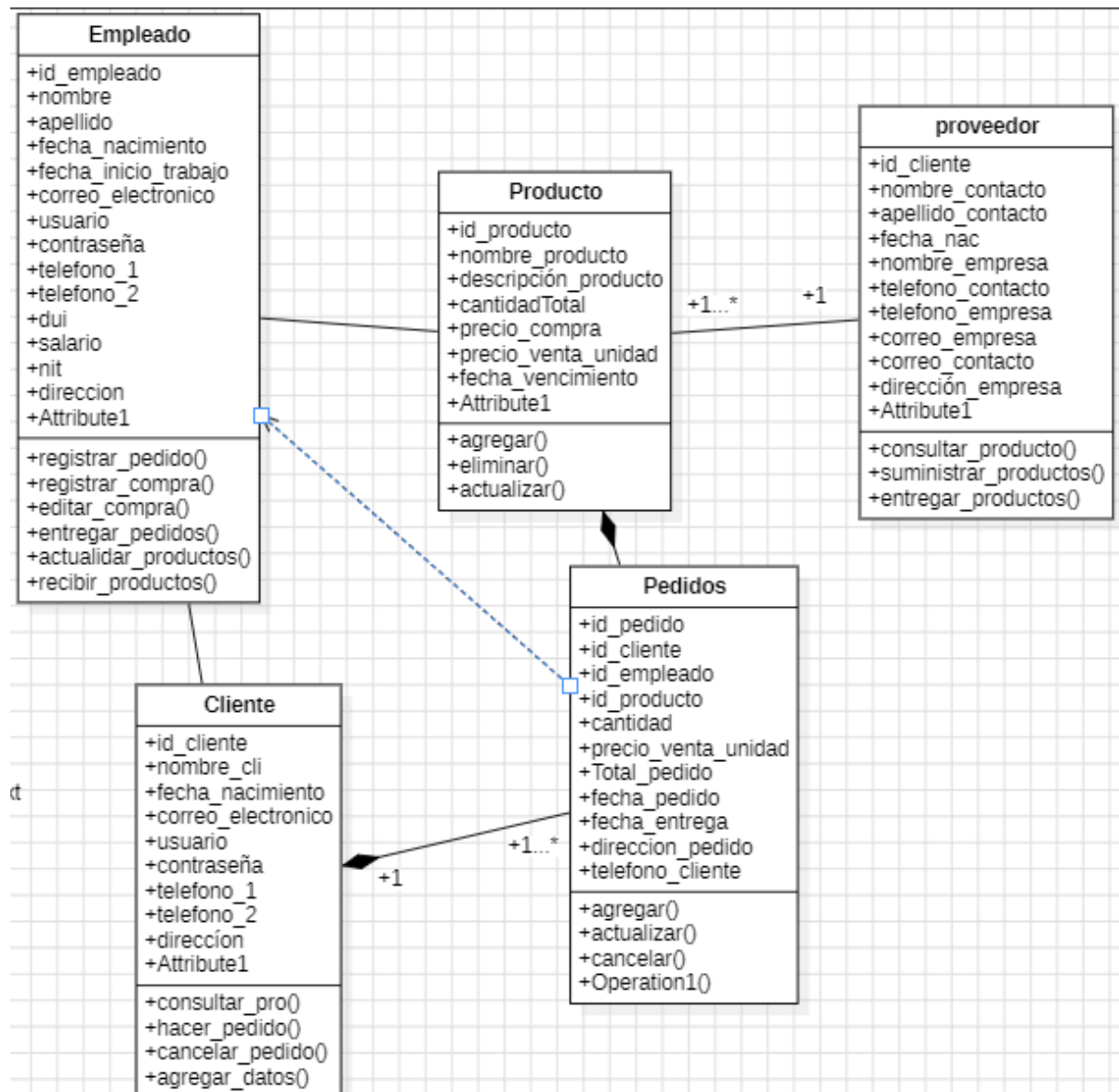
Pedidos
+id_pedido +id_cliente +id_empleado +id_producto +cantidad +precio_venta_unidad +Total_pedido +fecha_pedido +fecha_entrega +direccion_pedido +telefono_cliente
+agregar() +actualizar() +cancelar() +Operation1()

**Atributos:**

- id\_pedidos
- id\_cliente
- id\_empleado
- id\_producto
- cantidad
- precio\_venta\_unidad
- Total\_pedido
- fecha\_pedido
- fecha\_entrega
- direccion\_pedido
- telefono\_cliente

## Asociaciones y Agregaciones

- La clase empleado tiene una relación de asociación con la clase producto ya que permite que se acceda a ver los detalles.
- La clase Producto está asociada con la clase proveedores con el fin de poder detallar el producto, está relacionada con la clase pedidos ya que depende la existencia del producto que el pedido se lleve a cabo..
- La clase empleado está asociada de manera dependiente a la clase pedido para tener comprobante sobre el registro en la empresa y poder registrar dicho pedido.
- La clase empleado está asociada con la clase cliente para poder consultar cualquier producto o pedido.
- La clase pedido está asociada a la clase cliente para poder registrar el cliente que quiere dichos productos.
- la clase pedido está relacionada de manera dependiente de la clase producto ya que si no hay producto no hay pedido que registrar.



# ARQUITECTURA DE SOFTWARE

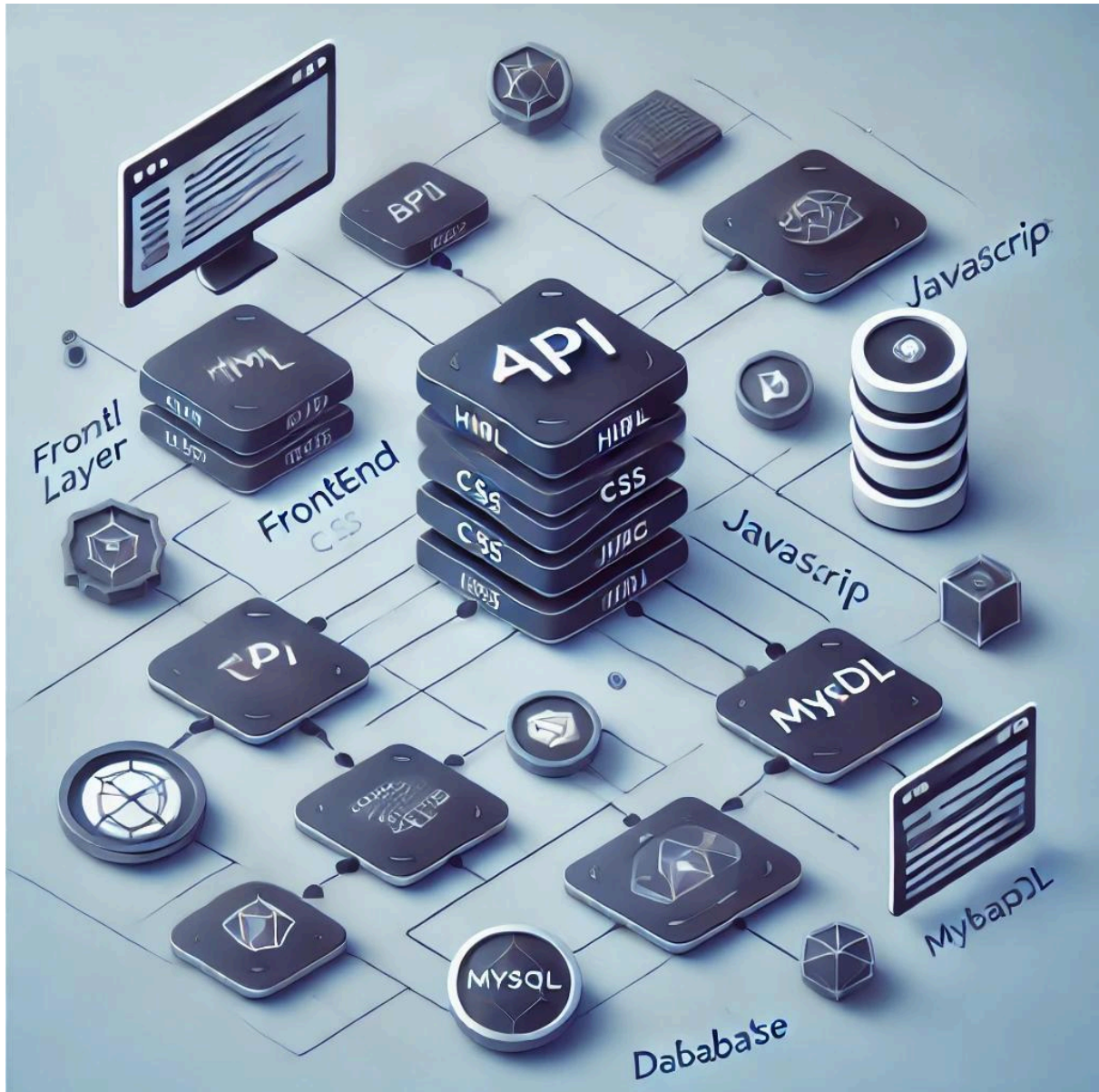
Para desarrollar una página web que gestione el inventario de una empresa, incluyendo productos, empleados, proveedores y clientes, es crucial tener un diseño bien estructurado. El objetivo es crear un sistema de gestión que permita a una empresa administrar su inventario de productos, junto con la información de empleados, proveedores y clientes. El sistema debe ser capaz de realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) para cada uno de estos elementos para eso iremos detallando el software con diagramas UML.

El siguiente caso de uso engloba el funcionamiento del sistema mientras se efectúa una venta, este caso de uso lo llamaremos Caso de Contexto en el cual los actores son: Empleado, Proveedor y Cliente.

El diagrama de arquitectura de software muestra la estructura básica de una aplicación web utilizando varias tecnologías clave:

1. Frontend (HTML, CSS, JavaScript): Esta es la parte de la aplicación con la que interactúan los usuarios. El frontend se encarga de la presentación y la experiencia del usuario. HTML define la estructura del contenido, CSS se encarga del estilo y el diseño visual, y JavaScript añade interactividad y lógica en el navegador.
2. API (RESTful o GraphQL): La API actúa como un intermediario entre el frontend y el backend. Es el punto de comunicación donde el frontend envía solicitudes para obtener o enviar datos, y el backend responde con la información solicitada. En el diagrama, la API se muestra como una capa independiente que conecta ambos extremos de la aplicación.
3. Backend (Express.js y Node.js): Esta es la lógica del servidor donde se maneja la lógica de la aplicación, la gestión de usuarios, la autenticación, y las interacciones con la base de datos. Express.js es un marco de trabajo para Node.js que facilita la creación de aplicaciones web robustas y rápidas.
4. Base de Datos (MySQL): MySQL es el sistema de gestión de bases de datos relacional utilizado para almacenar datos de manera estructurada. En el diagrama, MySQL está conectado al backend, lo que permite que la aplicación almacene y recupere datos de forma eficiente.

Este diagrama ilustra cómo estos componentes se conectan entre sí para crear una aplicación web funcional y escalable.



# HERRAMIENTAS A UTILIZAR

## CI/CD JENKINS Y GIT

### 1. Uso de Git para Control de Versiones y Colaboración

Se utilizará Git para gestionar todo el código fuente de la plataforma web. Cada desarrollador tendrá su propia copia del repositorio, lo que permite trabajar en paralelo en diferentes características.

### 2. Automatización con Jenkins

**Integración Continua (CI):** Se configura Jenkins para que cada vez que se suban cambios a Git se active un pipeline de Jenkins. Este pipeline podría incluir pasos como:

- Compilación del Proyecto
- Ejecución de Pruebas
- Análisis de Calidad de Código

**Entrega Continua (CD):** Se configuran pipelines de entrega continua en Jenkins para que, una vez aprobados los Pull Request, el código se despliegue automáticamente en un entorno de prueba o incluso en producción.

### 3. Gestión Centralizada de Usuarios

Control de Acceso en Jenkins: Se usará Jenkins para gestionar quién tiene acceso a ejecutar, modificar, y monitorear los pipelines. Se pueden definir roles y permisos, asegurando que solo las personas autorizadas puedan desplegar cambios o modificar la configuración.

## Cookies y tokens JWT

Usaremos lo JWT principalmente para la autenticación y la autorización, por ejemplo:

### 1. Proceso de Inicio de Sesión

Cuando un usuario intenta iniciar sesión, envía sus credenciales, el servidor valida las credenciales y, si son correctas, genera un JWT que incluye información relevante sobre el usuario. El servidor firma el JWT y lo envía de vuelta al usuario. Este token se almacena en el cliente, generalmente en cookies o en el almacenamiento local del navegador.

### 2. Validación de Peticiones

Cada vez que el usuario realiza una solicitud al servidor, envía el JWT en el encabezado de la solicitud HTTP; el servidor recibe el JWT, lo descifra y valida la firma para asegurarse de que



no ha sido manipulado. Si el token es válido, el servidor permite al usuario acceder a los recursos solicitados.

## **Vuex**

### **1. Gestión de Autenticación y Autorización**

**Almacenar Información del Usuario:** Se puede utilizar Vuex para almacenar la información del usuario autenticado, como su JWT, roles, y permisos. Esto permite que cualquier componente de la aplicación acceda a esta información sin necesidad de pasarla explícitamente a través de props.

**Autenticación:** Al iniciar sesión, la acción de autenticación en Vuex podría realizar una solicitud a la API de autenticación, recibir el JWT y almacenarlo en el estado de Vuex.

**Verificación de Permisos:** Se puede usar getters para verificar si un usuario tiene los permisos necesarios para realizar ciertas acciones

### **2. Gestión de Catálogos de Productos**

**Almacenamiento de Datos del Catálogo:** Se puede utilizar Vuex para almacenar y gestionar los datos de los catálogos de productos, asegurándose de que los cambios realizados en un componente se reflejan inmediatamente en todos los componentes que consumen esos datos.

**Sincronización con la API:** Las acciones de Vuex pueden gestionar la sincronización del estado de los catálogos con la API del backend.

## **VUE.js**

En el contexto de la plataforma web para crear, gestionar y personalizar catálogos de productos, Vue.js se puede utilizar de diversas maneras para mejorar la interacción del usuario, la modularidad del código y la eficiencia general del desarrollo.

### **1. Construcción de la Interfaz de Usuario (UI)**

Vue.js permite crear componentes reutilizables que encapsulan HTML, CSS y JavaScript en un solo archivo (`.vue`) y también facilita la composición de la interfaz de usuario a partir de estos componentes reutilizables.

### **2. Interactividad y Reactividad**

Vue.js proporciona un binding de datos bidireccional, lo que significa que cualquier cambio en el modelo de datos se refleja automáticamente en la vista y viceversa. Esto es ideal para formularios donde los usuarios pueden editar productos en el catálogo y ver los cambios en tiempo real.

### 3. Interacción con APIs

Vue.js se integra bien con bibliotecas como Axios para hacer solicitudes HTTP. Se puede usar Axios para interactuar con la API backend, cargar productos desde la base de datos, agregar nuevos productos, o actualizar productos existentes.

## CRONOGRAMA

Para este trabajo se llevarán 3 meses los cuales hemos planificado con el siguiente cronograma:

	Julio	Agosto	Septiembre	Octubre
<b>Fase 1 Planificación y requisitos</b>				
Reunión inicial	17/07/2024			
Aprobación de la idea de proyecto	19/07/2024			
Definición de objetivos				
Creación de sitemap				
Control del proyecto		14/08/2024		
Diseño UI, selección de estilo				
Mockups, feedback				
Diseño final				
Desarrollo frontend(páginas principales)				
Desarrollo backend(servidor, BD)				
Implementación de funcionalidades				
Pruebas(tester)				
Control del proyecto				
<b>Fase final</b>				
Pruebas de usabilidad y rendimiento(tester)				

Corrección de errores				
Revisión final				
Lanzamiento, monitoreo				
Control del proyecto				17/10/2024

Para este proyecto las reuniones serán constantes y se harán mínimo dos veces a la semana para tener una comunicación clara, precisa y ordenada.

## COSTO

Concepto	Costo Mensual	Total (3 meses)
Mano de Obra		
Desarrollador Front-End	\$ 1.000,00	\$ 3,000.00
Desarrollador 1 Back-End	\$ 1.500,00	\$ 4,500.00
Desarrollador 2 Back-End	\$ 1.500,00	\$ 4,500.00
Diseñador 1 UX/UI	\$ 1.000,00	\$ 3,000.00
Diseñador 2 UX/UI	\$ 1.000,00	\$ 3,000.00
<b>Infraestructura y Servicios</b>		\$ -
Computadoras (5)		\$ 7,500.00
Licencias de Software	\$ 300,00	\$ 900.00
Servidor Web y Base de Datos	\$ 150,00	\$ 450.00
APIs	\$ 300,00	\$ 900.00
Servicios de Nube	\$ 100,00	\$ 300.00
<b>Costos Operativos</b>		\$ -
Electricidad	\$ 100,00	\$ 300.00
Agua	\$ 50,00	\$ 150.00
Internet	\$ 150,00	\$ 450.00

Alquiler de Oficina	\$ 500,00	\$ 1,500.00
Mobiliario y Suministros		\$ 1,500.00
Mantenimiento y Limpieza	\$ 200,00	\$ 600.00
<b>Subtotal</b>		<b>\$ 32,550.00</b>
Fondo de Contingencia (10%)		\$ 3,255.00
<b>Total</b>		<b>\$ 35,805.00</b>

El presupuesto muestra el valor mensual que se ocupara y el total de los tres meses que se planea trabajar, al considerar todos los factores necesarios, desde mano de obra hasta gastos operativos y gastos de contingencia podemos asegurar que el proyecto se mantenga dentro del presupuesto establecido.

## FUENTES DE CONSULTA

- Cruz, A. L. (2024, 22 marzo). *La importancia del diseño UX/UI en el mundo digital*. Eniun. <https://www.eniun.com/importancia-diseno-ux-ui-mundo-digital/>
- B, G., & B, G. (2024, 23 agosto). *¿Cuál es el precio de crear una página web en 2024?* Tutoriales Hostinger. <https://www.hostinger.es/tutoriales/precio-pagina-web>
- Bigredes. (2023, 1 diciembre). *¿Cuál es el precio de diseño de páginas web en El Salvador?: Comparativa 5 Diseñadores 2024* » BigRedes. *BigRedes*.  
<https://bigredes.com/precio-de-diseno-de-paginas-web-en-el-salvador/>
- Domeka, I. (2024, 6 marzo). *Qué es un mockup y cómo hacer uno en 7 pasos*. *IdeandoAzul - Diseño Web Estratégico y Branding Para Negocios Digitales*.  
<https://ideandoazul.com/branding/mockup/>
- Ospina, D. (2023, 21 abril). *Mockup web: ¿Qué es y cómo diseñarlo? | Diseño web*. *Marketeros Agencia*.  
<https://www.marketerosagencia.com/blog/diseno-web/mockup-web-que-es/>
- *Arquitectura de software: ¿Qué es y qué tipos hay?* (s. f.). Gluo.  
<https://www.gluo.mx/blog/arquitectura-de-software-que-es-y-que-tipos-hay>
- *Cómo diseñar una arquitectura de software: consejos y prácticas recomendadas*. (2021, 8 abril). Lucidchart.  
<https://www.lucidchart.com/blog/es/como-disenar-una-arquitectura-de-software>