

**Actividad 6 : Aplicativo de arquitectura de software**

Jefferson Alexander Rojas Silva

**100099657**

Fabio Arturo Antolínez Sastoque

**100100005**

Facultad de ingeniería, Corporación Universitaria iberoamericana

202331: Arquitectura de software

Sr. José Castro

Abril de 2023

## **Aplicativo de arquitectura de software**

### **Introducción**

¡Bienvenidos a nuestra presentación del proyecto de la aplicación de ventas! En este proyecto, hemos desarrollado una aplicación de ventas en línea que ofrece una experiencia de compra fluida y eficiente para nuestros clientes. Nuestra aplicación se basa en una arquitectura de tres capas, con una interfaz de usuario atractiva y fácil de usar en el front-end, una lógica de negocio robusta en el back-end y una base de datos eficiente para gestionar los datos de los productos y clientes.

Nuestro objetivo principal con esta aplicación es proporcionar a nuestros clientes una plataforma segura y conveniente para comprar productos en línea. Hemos implementado medidas de seguridad para proteger los datos de nuestros clientes y asegurar una comunicación segura con el servidor. También hemos diseñado una interfaz de usuario intuitiva y responsive que se adapta a diferentes dispositivos, para que nuestros clientes puedan acceder a la aplicación desde cualquier lugar y en cualquier momento.

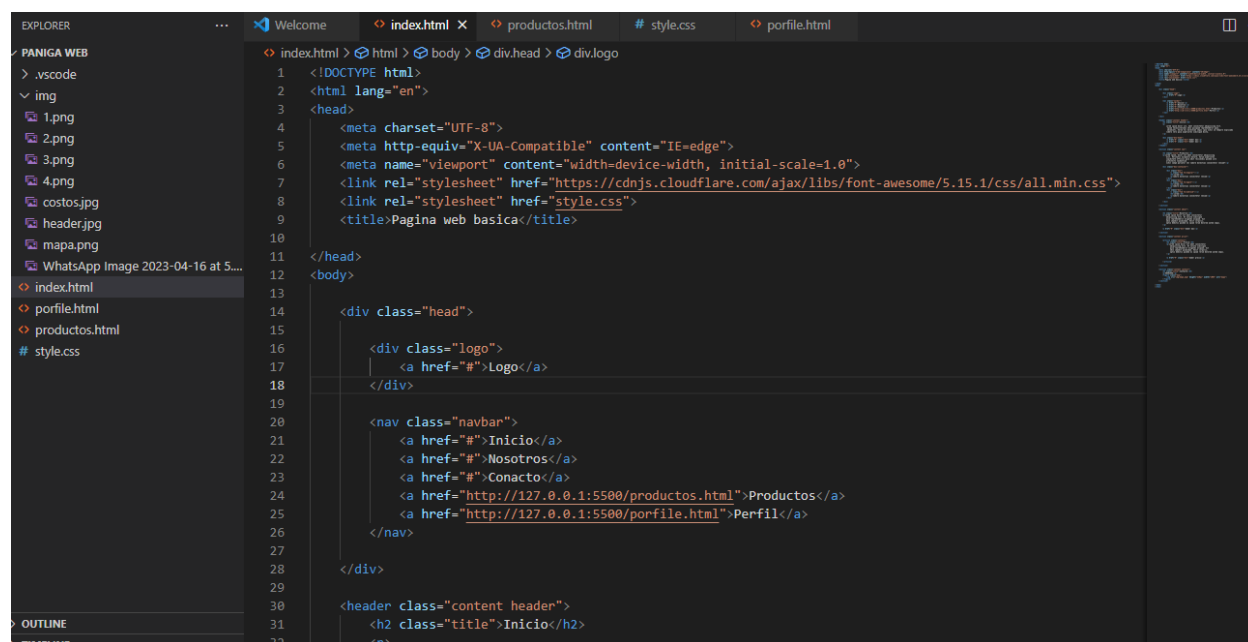
En esta presentación, exploraremos la arquitectura que hemos utilizado en nuestra aplicación, presentaremos un modelo del producto y proporcionaremos una descripción técnica detallada de cómo funciona nuestra aplicación. Estamos emocionados de compartir con ustedes este proyecto y esperamos que disfruten de la presentación. ¡Vamos a sumergirnos en los detalles técnicos de nuestra aplicación de ventas!

### **Arquitectura:**

Para desarrollar nuestra aplicación, hemos utilizado una arquitectura de tres capas, también conocida como arquitectura cliente-servidor. Las tres capas son:

Capa de presentación (Front-end): En esta capa utilizamos HTML, CSS y JavaScript para crear la interfaz de usuario de la aplicación. La interfaz es intuitiva y fácil de usar, con una navegación clara y accesible. Hemos implementado un diseño responsive que se adapta a diferentes dispositivos, como computadoras de escritorio, tablets y teléfonos móviles, para asegurarnos de que nuestros clientes puedan acceder a la aplicación desde cualquier dispositivo.

Capa de lógica de negocio (Back-end): En esta capa utilizamos tecnologías como PHP y MySQL para manejar la lógica de negocio de la aplicación. Aquí es donde se realizan todas las operaciones relacionadas con los datos de los productos, como la búsqueda, la gestión de inventario, el procesamiento de pedidos y el cálculo de precios. También implementamos medidas de seguridad para proteger los datos de nuestros clientes, como la encriptación de contraseñas y la prevención de ataques cibernéticos.



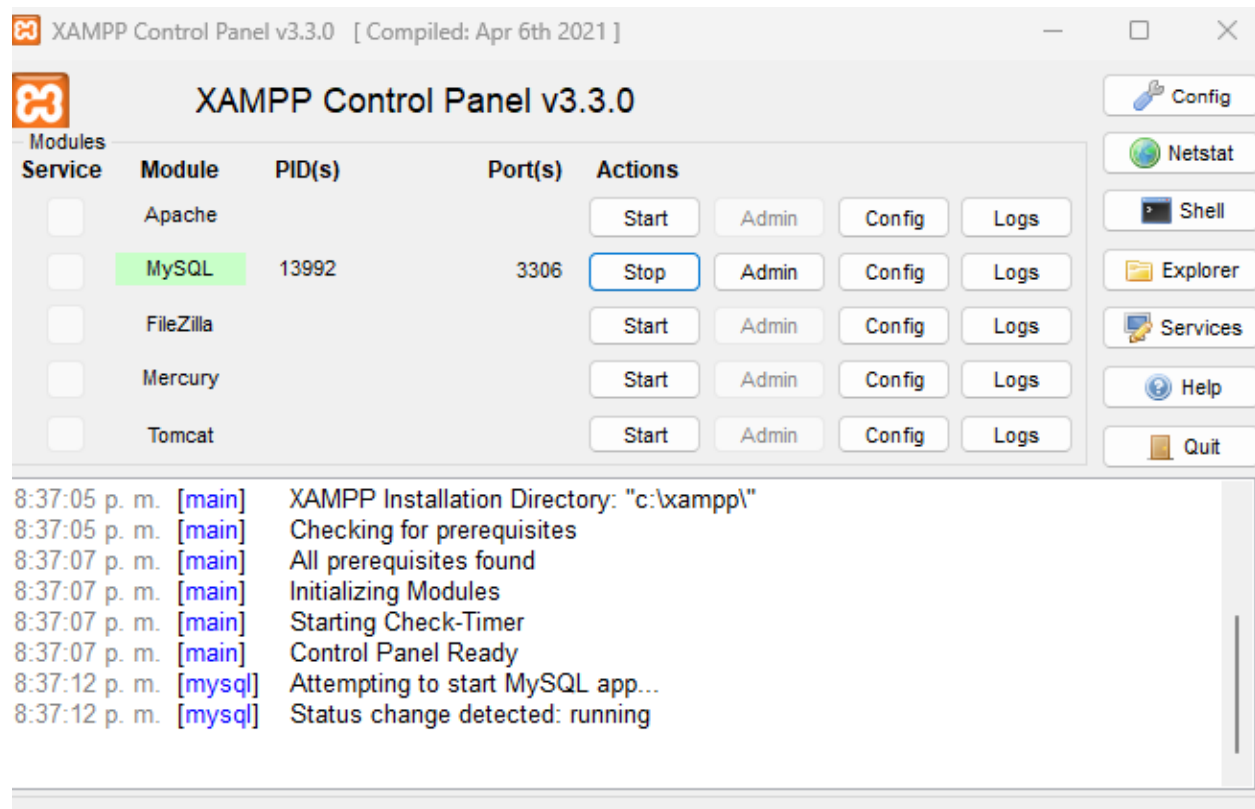
```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta http-equiv="X-UA-Compatible" content="IE=edge">
6    <meta name="viewport" content="width=device-width, initial-scale=1.0">
7    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/font-awesome@5.15.1/css/all.min.css">
8    <link rel="stylesheet" href="style.css">
9    <title>Pagina web basica</title>
10
11 </head>
12 <body>
13
14   <div class="head">
15
16     <div class="logo">
17       <a href="#">Logo</a>
18     </div>
19
20     <nav class="navbar">
21       <a href="#">Inicio</a>
22       <a href="#">Nosotros</a>
23       <a href="#">Contacto</a>
24       <a href="http://127.0.0.1:5500/productos.html">Productos</a>
25       <a href="http://127.0.0.1:5500/profile.html">Perfil</a>
26     </nav>
27
28   </div>
29
30   <header class="content header">
31     <h2 class="title">Inicio</h2>
32     <p>

```

Capa de datos (Base de datos): Utilizamos una base de datos MySQL para almacenar la información sobre nuestros productos, clientes, pedidos y otros datos relevantes para la

aplicación. La base de datos está diseñada para ser eficiente y escalable, lo que nos permite gestionar grandes volúmenes de datos y asegurar un rendimiento óptimo de la aplicación.



### Herramientas utilizadas:

**HTML:** Lenguaje de marcado para la creación de la estructura y contenido de la interfaz de usuario de la aplicación.

**CSS:** Lenguaje de estilos para la presentación visual de la interfaz de usuario, aplicando diseños, colores y estilos.

**JavaScript:** Lenguaje de programación para la implementación de la lógica del cliente en el front-end de la aplicación, como la interacción del usuario y la validación de formularios.

**PHP:** Lenguaje de programación del lado del servidor para la implementación de la lógica de negocio, procesamiento de datos y comunicación con la base de datos.

MySQL: Sistema de gestión de bases de datos relacional para almacenar y gestionar la información de los productos, clientes y pedidos.

**Modelo del producto:**

Nuestra aplicación de ventas permite a los clientes explorar y comprar una amplia gama de productos en línea. Los clientes pueden navegar por categorías de productos, realizar búsquedas, ver detalles de los productos, añadir productos al carrito de compra y realizar pedidos. También ofrecemos funciones de registro de usuarios y autenticación para proporcionar una experiencia personalizada y segura a nuestros clientes. Además, los clientes pueden ponerse en contacto con nosotros a través de la sección de contacto para realizar consultas o aclarar dudas.

**Descripción técnica del producto:**

La aplicación está construida utilizando tecnologías web estándar, incluyendo HTML5, CSS3, JavaScript, PHP y MySQL. Utilizamos frameworks y librerías populares como Bootstrap y jQuery para agilizar el desarrollo y garantizar una apariencia y funcionamiento consistentes en toda la aplicación. Además, implementamos medidas de seguridad, como la validación de datos del usuario para prevenir ataques de inyección de SQL y cross-site scripting (XSS), y utilizamos HTTPS para asegurar la comunicación segura entre el cliente y el servidor.

En resumen, nuestra aplicación de ventas utiliza una arquitectura de tres capas con una interfaz de usuario atractiva y fácil de usar en la capa de presentación, una lógica de negocio robusta en la capa de lógica de negocio y una base de datos eficiente en la capa de datos. Ofrecemos una amplia gama de productos y funcionalidades, mientras nos aseguramos de mantener la

La aplicación consta de varios apartados:

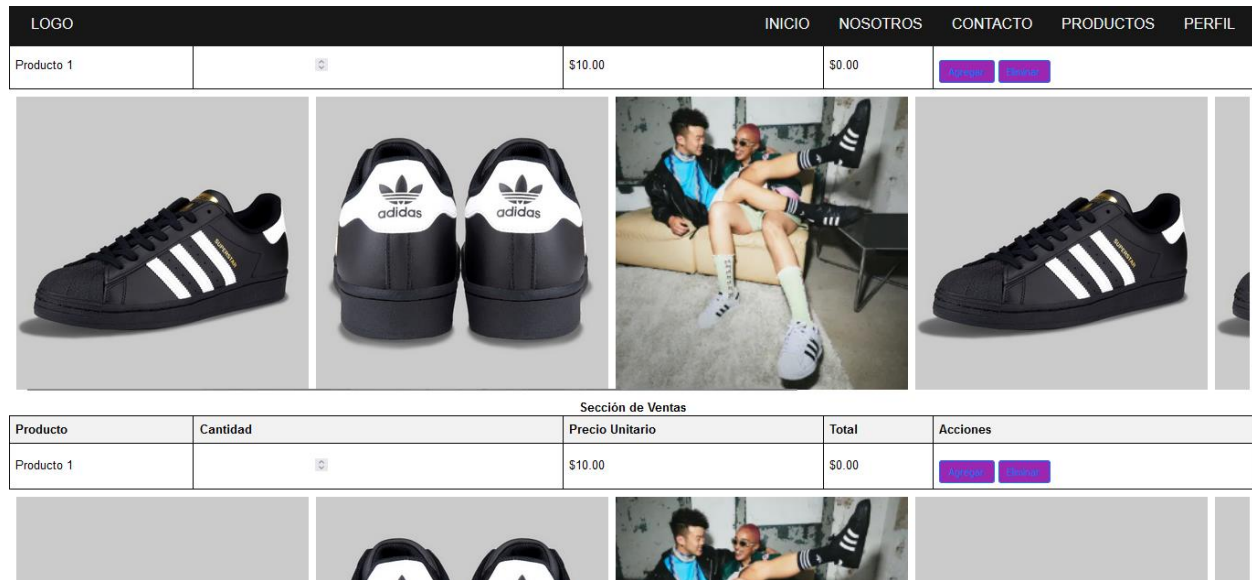
**Inicio:** Esta es la página principal de la aplicación de ventas. Aquí se muestra una vista general de los productos destacados, promociones o información relevante sobre la empresa.

**Nosotros:** En esta sección se brinda información sobre la empresa, su historia, misión, visión, equipo de trabajo y cualquier otro detalle relevante.

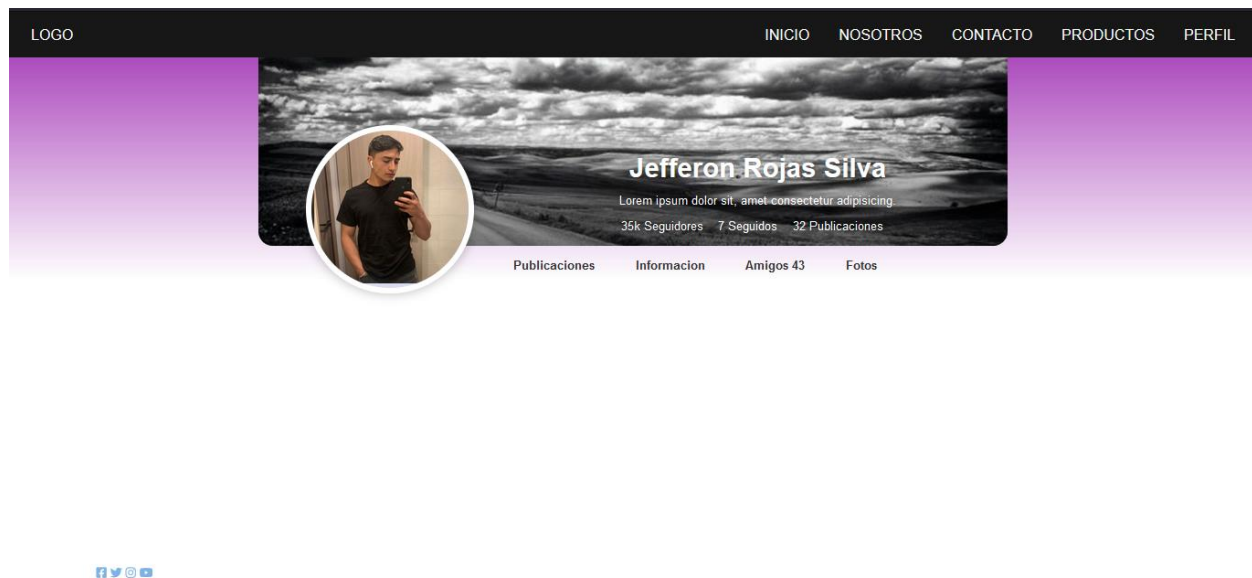


**Contacto:** En este apartado, los usuarios pueden ponerse en contacto con la empresa a través de un formulario de contacto o proporcionando información de contacto, como dirección de correo electrónico o número de teléfono. (en desarrollo).

**Productos:** Esta sección permite a los usuarios navegar y buscar productos disponibles para la venta. Los productos se organizan en categorías y se muestran con detalles como nombre, descripción, precio, imagen, etc. Los usuarios también pueden agregar productos a su carrito de compras desde esta sección. (en desarrollo).



Perfil: Esta sección permite a los usuarios registrados acceder a su perfil de usuario. Aquí pueden ver su historial de compras, actualizar su información de contacto, cambiar su contraseña y gestionar otras configuraciones de cuenta.



### Descripción/Código index:

Este código es una página web básica escrita en lenguaje de marcado HTML (Hypertext Markup Language). Proporciona una estructura y diseño simple para una página web con varias

secciones, como un encabezado, contenido principal y un pie de página. A continuación, se explica brevemente la estructura y funcionalidad del código:

Doctype y etiqueta html: La declaración `<!DOCTYPE html>` especifica la versión de HTML utilizada en el documento y la etiqueta `<html>` define el comienzo y el final del documento HTML.

Etiqueta head: La sección `<head>` contiene metadatos sobre el documento HTML, como la codificación de caracteres, la compatibilidad del navegador, y enlaces a hojas de estilo y archivos externos.

Etiqueta meta: Las etiquetas `<meta>` proporcionan información adicional sobre el documento HTML, como la codificación de caracteres y la configuración de la vista en dispositivos móviles.

Enlaces a hojas de estilo: Los enlaces `<link>` son utilizados para vincular hojas de estilo externas a la página HTML. En este caso, se vinculan dos archivos CSS, uno para la librería de iconos Font Awesome y otro para un archivo de estilo personalizado llamado "style.css".

Etiqueta title: La etiqueta `<title>` define el título de la página que se muestra en la barra de título del navegador.

Etiqueta body: La etiqueta `<body>` contiene todo el contenido visible de la página web, como encabezados, párrafos, enlaces, imágenes, etc.

Clases y elementos de estilo: Se utilizan clases y elementos HTML con clases para aplicar estilos personalizados a diferentes secciones de la página, como el encabezado, el contenido principal y las secciones de contacto.

Etiquetas de encabezado: Se utilizan las etiquetas de encabezado `<h1>`, `<h2>`, `<h3>` para definir los títulos y subtítulos de las diferentes secciones de la página.

Enlaces: Se utilizan las etiquetas `<a>` para crear enlaces a otras páginas o recursos externos.



Íconos: Se utilizan íconos de la librería Font Awesome mediante la clase "fab" y las clases específicas para cada ícono, como "fa-angular", "fa-apple" y "fa-android".

Elementos de párrafo: Se utilizan las etiquetas <p> para crear párrafos de texto.

Contenedores de contenido: Se utilizan divs con clases para crear contenedores de contenido, como el encabezado, las secciones de contenido y el pie de página.

En resumen, este código HTML crea una página web básica con varias secciones, estilos personalizados y enlaces a hojas de estilo externas y recursos externos, como íconos y una imagen de mapa.

### **Descripción/Código Perfil:**

El código HTML y CSS proporcionado parece ser una parte de una página web que muestra un perfil de usuario en una aplicación llamada "SkramApp". A continuación se proporciona una explicación detallada del código:

La etiqueta <!DOCTYPE html> define el tipo de documento y la versión de HTML que se utiliza en el documento.

La etiqueta <html lang="en"> define el comienzo del elemento HTML raíz del documento, y se establece el atributo lang en "en" para indicar que el lenguaje principal del documento es el inglés.

La etiqueta <head> contiene metaetiquetas y enlaces a hojas de estilo externas que se utilizan para dar estilo y configurar la página web. Dentro de la etiqueta <head>, se encuentran las siguientes etiquetas y atributos:

<meta charset="UTF-8">: Define la codificación de caracteres del documento como UTF-8, que es una codificación de caracteres universal que admite la mayoría de los caracteres y símbolos utilizados en varios idiomas.

`<meta http-equiv="X-UA-Compatible" content="IE=edge">`: Define el modo de compatibilidad del navegador para Internet Explorer.

`<meta name="viewport" content="width=device-width, initial-scale=1.0">`: Define la configuración de la vista en dispositivos móviles para que el ancho de la página se ajuste al ancho del dispositivo y se muestre correctamente en pantallas pequeñas.

`<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.1/css/all.min.css">`: Enlace a una hoja de estilo externa de la biblioteca Font Awesome, que proporciona iconos vectoriales escalables.

`<link rel="stylesheet" href="style.css">`: Enlace a una hoja de estilo externa llamada "style.css", que contiene reglas de estilo personalizadas para la página web.

La etiqueta `<title>` define el título de la página web que se muestra en la barra de título del navegador.

La etiqueta `<body>` contiene el contenido visible de la página web. Dentro de la etiqueta `<body>`, se encuentran varios elementos HTML que forman la estructura de la página, incluyendo:

Un `<div>` con la clase "head", que contiene la cabecera de la página web, que incluye un logotipo y una barra de navegación.

Dentro del `<div>` de clase "head", se encuentra otro `<div>` con la clase "logo", que contiene un enlace de anclaje `<a>` con el texto "logo".

También dentro del `<div>` de clase "head", se encuentra una etiqueta `<nav>` con la clase "navbar", que contiene varios enlaces de navegación que representan las diferentes páginas de la aplicación "SkramApp", como "inicio", "Nosotros", "contacto", "Productos" y "perfil".

Un segundo <body> sin etiqueta de cierre, lo cual es incorrecto y debería ser eliminado del código.

Dentro del segundo <body>, se encuentra un bloque de estilo CSS incorporado usando la etiqueta <style>. Este bloque de estilo contiene reglas de estilo para la apariencia del perfil de usuario, incluyendo el fondo, los colores, las fuentes, el diseño y la disposición de los elementos en la página.

### **Operación:**

**Inicio:** La página de inicio mostraría los productos destacados, promociones o novedades de la tienda. Los usuarios podrían ver la información de los productos, como imágenes, nombres y precios. Podrían hacer clic en los botones o enlaces para ver más detalles de los productos, agregarlos al carrito o realizar compras.

**Nosotros:** La página "Nosotros" mostraría información sobre la empresa o el equipo detrás de la aplicación de ventas. Podría incluir texto descriptivo, imágenes y/o videos que presenten la empresa, su misión, visión, valores y otros detalles relevantes.

**Contacto:** La página de contacto podría incluir un formulario que permita a los usuarios enviar mensajes a la empresa. Los usuarios podrían completar campos como nombre, dirección de correo electrónico y mensaje, y luego hacer clic en el botón de envío. El servidor recibiría los datos ingresados por el usuario, los procesaría y enviaría una notificación a la empresa con la información del mensaje.

**Productos:** La página de productos mostraría la lista de productos disponibles para la venta. Los usuarios podrían ver imágenes de los productos, sus nombres, descripciones y precios. Podrían hacer clic en los botones de agregar al carrito para añadir productos a su carrito de compra. El

servidor recibiría la solicitud del usuario y actualizaría el carrito de compra del usuario en su perfil.

**Perfil:** La página de perfil permitiría a los usuarios registrados acceder a su información personal. Podrían ver su historial de compras, actualizar sus datos personales, cambiar su contraseña, etc. El servidor verificaría la autenticación del usuario, accedería a la base de datos para obtener la información del perfil del usuario y mostraría los datos relevantes en la página. Los usuarios podrían actualizar su información y el servidor se encargaría de guardar los cambios en la base de datos.

En general, la operación de la aplicación implicaría la interacción del usuario con las diferentes páginas HTML, el envío de datos al servidor a través de formularios, la validación de la información en el servidor, la interacción con una base de datos para obtener o actualizar datos, y la actualización de la interfaz de usuario en consecuencia. Todo esto se lograría a través de la comunicación entre el cliente (navegador del usuario) y el servidor, siguiendo el flujo de la lógica de negocio implementada en el servidor y la presentación de la información en las páginas HTML.

#### **Objetivo de la aplicación:**

- Proporcionar una plataforma segura y conveniente para que los clientes puedan comprar productos en línea de manera eficiente y cómoda.
- Ofrecer una interfaz de usuario intuitiva y responsive que se adapte a diferentes dispositivos, para que los clientes puedan acceder a la aplicación desde cualquier lugar y en cualquier momento.
- Implementar medidas de seguridad para proteger los datos de los clientes, como la encriptación de contraseñas y la prevención de ataques cibernéticos.

- Gestionar eficientemente los datos de productos, clientes y pedidos a través de una base de datos, asegurando la integridad y confiabilidad de la información.
- Proporcionar funciones de búsqueda, gestión de inventario, procesamiento de pedidos y autenticación de usuarios para una experiencia de compra en línea personalizada y conveniente.

**Conclusiones:**

- En conclusión, nuestra aplicación de ventas es un proyecto bien estructurado que utiliza una arquitectura de tres capas para ofrecer una experiencia de compra en línea eficiente y segura. Hemos utilizado tecnologías web estándar como HTML, CSS, JavaScript, PHP y MySQL, junto con frameworks y librerías populares para agilizar el desarrollo y garantizar una interfaz de usuario atractiva y funcional.
- Nuestra aplicación permite a los clientes explorar y comprar una amplia gama de productos en línea, con funciones de búsqueda, gestión de inventario, procesamiento de pedidos y autenticación de usuarios para una experiencia personalizada y segura. Además, hemos implementado medidas de seguridad para proteger los datos de los clientes, como la encriptación de contraseñas y la prevención de ataques cibernéticos.
- En general, nuestro proyecto de aplicación de ventas ha sido diseñado pensando en la comodidad y seguridad de nuestros clientes, brindando una interfaz intuitiva y responsive, una lógica de negocio robusta y una base de datos eficiente. Estamos orgullosos del trabajo realizado y estamos entusiasmados con el potencial de esta aplicación para ofrecer una experiencia de compra en línea excepcional a nuestros clientes.

# Entidades

## Propuesta de aplicativo con arquitectura



Representa a los usuarios registrados en la aplicación, con atributos como nombre, dirección de correo electrónico, contraseña, historial de compras, etc.

Atributos:

- ID\_Usuario (Clave primaria)
- Nombre
- CorreoElectronico
- Contraseña
- HistorialCompras

1



Representa los productos disponibles para la venta, con atributos como nombre, descripción, precio, imagen, etc.

Atributos:

- ID Producto (Clave primaria)
- Nombre
- Descripción
- Precio
- Imagen
- Categoría

2



Representa el carrito de compras de un usuario, con atributos como el ID del usuario que lo posee y los productos agregados al carrito.

Atributos:

- ID\_CarritoCompra (Clave primaria)
- ID\_Usuario (Clave foránea a Usuario)
- Productos (Lista de productos en el carrito)

3



Representa los mensajes de contacto enviados por los usuarios a la empresa, con atributos como nombre, correo electrónico, mensaje, etc.

Atributos:

- ID\_Contacto (Clave primaria)
- Nombre
- CorreoElectronico
- Mensaje

4



Arquitectura de software

# Relaciones

## Propuesta de aplicativo con arquitectura

### RELACION COMPRA 1

Asocia un usuario con los productos que ha comprado, registrando la información de la compra.

Atributos:

- ID\_Compra (Clave primaria)
- ID\_Usuario (Clave foránea a Usuario)
- ID\_Producto (Clave foránea a Producto)
- Cantidad
- FechaCompra

1

### RELACIÓN PRODUCTO C ATEGORIA 2

Asocia un producto con una categoría específica.

Atributos:

- ID\_ProductoCategoría (Clave primaria)
- ID\_Producto (Clave foránea a Producto)
- ID\_Categoría (Clave foránea a Categoría)

2

### RELACIÓN CATEGORIA 3

Representa las categorías de los productos disponibles para la venta.

Atributos:

- ID\_Categoría (Clave primaria)
- Nombre

3



El MoCAD propuesto es una representación conceptual de las entidades y relaciones que podrían estar presentes en la aplicación de ventas.

## Bibliografía

Alicante, U. d. (s.f.). *Universitat d'Alacant*. Obtenido de Servicio de Informática SP.NET MVC3

Framework: <https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>

Huaman, W. C. (7 de Septiembre de 2018). *MEDIUM*. Obtenido de Los 10 patrones comunes de

arquitectura de software: <https://medium.com/@maniakhitoccori/los-10-patrones-comunes-de-arquitectura-de-software-d8b9047edf0b>

OPenWebinars. (24 de Agosto de 2022). *Arquitectura de software: Qué es y qué tipos existen*.

Obtenido de <https://openwebinars.net/blog/arquitectura-de-software-que-es-y-que-tipos-existen/>

Sommerville, I. (2011). *Ingeniería de software*. Pearson Educación.