**CATÁLOGO WEB "BLOOM" – TALENTO TECH**

Luis Alberto Quino Manrique  
CC. 1075305271

Ministerio de Tecnologías de la Información y las Comunicaciones

Corporación Unificada Nacional de Educación Superior

[luisalbertoquino@gmail.com](mailto:luisalbertoquino@gmail.com)

2025

**INTRODUCCIÓN**

El presente proyecto tiene como objetivo desarrollar una plataforma web integral para Bloom, empresa especializada en joyería femenina que ofrece productos elegantes y asequibles para el público femenino. La plataforma servirá como un catálogo web administrable que conectará los productos ofrecidos por la empresa con sus clientes, facilitando la visualización de productos, procesamiento de pedidos y gestión de contenidos.

Para el desarrollo del backend, se empleará Laravel, un potente framework PHP que permitirá crear una API robusta, segura y escalable. La base de datos estará gestionada mediante MySQL, asegurando un almacenamiento eficiente y seguro de la información. En el frontend, se utilizará Angular con TypeScript para crear una interfaz de usuario dinámica y reactiva, junto con Tailwind CSS para implementar un diseño moderno y totalmente personalizable. Esta combinación de tecnologías permitirá desarrollar una aplicación web con un alto rendimiento, excelente experiencia de usuario y fácil mantenimiento.

El proyecto incluirá las siguientes etapas:

1. Diseño de interfaz: Creación de mockups para definir la estructura y flujo de la plataforma.
2. Desarrollo backend: Implementación de controladores, modelos y migraciones en Laravel para gestión de productos, categorías y entradas del blog.
3. Desarrollo frontend: Creación de componentes Angular con TypeScript y estilizados con Tailwind CSS.
4. Integración con base de datos: Conexión segura mediante Eloquent ORM de Laravel para almacenar datos de productos, categorías, entradas y configuraciones del sitio.

**MOCKUPS**

**Mockup 1: Estructura de Navegación (Navbar)**

Este mockup define la estructura de navegación del catálogo web Bloom. El diseño prioriza la accesibilidad y la experiencia de usuario.

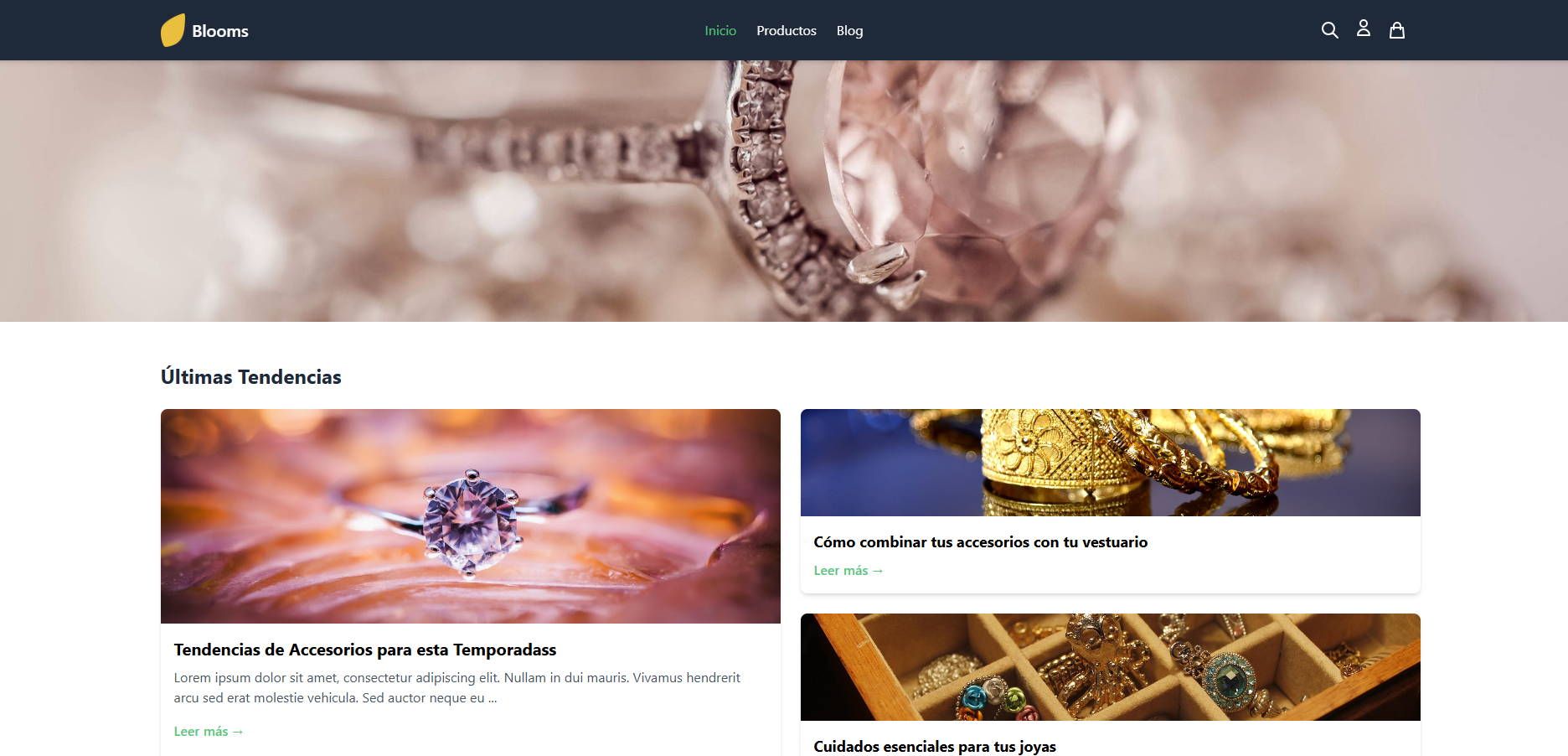
El navbar está organizado en tres secciones principales:

* **Izquierda**: Logo de Bloom
* **Centro**: Enlaces de navegación (Inicio, Productos, Blog)
* **Derecha**: Iconos funcionales (Búsqueda, Perfil/Login, Carrito)

****

**Mockup 2: Página de Inicio (Home)**

El siguiente mockup presenta la estructura principal de la página de inicio de Bloom, organizada en secciones visualmente atractivas. La parte superior incluye un banner destacado con promociones o novedades, seguido por tres entradas o noticias del blog. Continúa con un listado horizontal de categorías de productos, seguido por un extracto de productos destacados y un botón llamativo "Ver todos los productos". La página finaliza con el footer que contiene información de contacto y enlaces importantes.



Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

**Mockup 3: Visualización de Productos**

Este mockup muestra la sección de productos de Bloom, presentando una galería de cards con diseño elegante y minimalista. En la parte superior hay un buscador y un selector para organizar los productos. Los productos se muestran paginados para facilitar la navegación. Cada card de producto incluye:

* Imagen del producto
* Nombre del producto
* Precio
* Cuatro iconos que aparecen al hacer hover:
  1. Ver detalles (redirige a página individual del producto)
  2. Vista previa rápida (icono de rayo, abre popup)
  3. Añadir al carrito
  4. Comprar ahora (conecta con WhatsApp)

Interfaz de usuario gráfica, Sitio web

El contenido generado por IA puede ser incorrecto.

**Mockup 4: Vista Detallada de Producto**

En este mockup se presenta la página de detalle de producto individual, con una estructura que maximiza la presentación del producto:

* Imágenes del producto en alta calidad
* Información detallada (nombre, precio, descripción)
* Selector de cantidad
* Botones de acción (Añadir al carrito, Comprar ahora)
* Productos relacionados en la parte inferior

Interfaz de usuario gráfica, Sitio web

El contenido generado por IA puede ser incorrecto.

**Mockup 5: Vista Previa Rápida (Popup)**

Este mockup muestra el popup que aparece al hacer clic en el icono de rayo. Incluye:

* Imagen del producto
* Información básica (nombre, precio)
* Descripción breve
* Selector de cantidad
* Opciones de acción (Añadir al carrito, Comprar ahora)

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

**Mockup 6: Blog**

El mockup del blog presenta un diseño elegante que muestra las entradas más recientes:

* Banner superior con título "Blog"
* Artículos organizados en formato de cards
* Cada card incluye: imagen destacada, título, extracto y botón "Leer más"
* Sistema de paginación en la parte inferior

Interfaz de usuario gráfica, Sitio web

El contenido generado por IA puede ser incorrecto.

**Mockup 7: Panel Administrativo (Dashboard)**

Este mockup despliega la interfaz administrativa para el dueño del negocio:

* Sidebar izquierdo con:
  + Logo de Bloom
  + Sección "Inicio" seleccionada (muestra resumen estadístico)
  + Sección "Categorías" (gestión CRUD)
  + Sección "Productos" (gestión CRUD)
  + Sección "Entradas" (gestión CRUD)
  + Sección "Opciones de página" (personalización)

El contenido principal muestra el dashboard con estadísticas relevantes como:

* Productos más vendidos
* Categorías más populares
* Entradas más leídas
* Ventas recientes

Interfaz de usuario gráfica, Sitio web

El contenido generado por IA puede ser incorrecto.

**Mockup 8: Gestión de Productos (Admin)**

Este mockup muestra la interfaz de administración de productos:

* Tabla con listado de productos (columnas: ID, Imagen, Nombre, Precio, Categoría, Acciones)
* Botones de acción (Editar, Eliminar, Ver)
* Botón para agregar nuevo producto
* Formulario de edición/creación de producto

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

**Mockup 9: Gestión de Categorías (Admin)**

Este mockup presenta la interfaz para la gestión de categorías:

* Tabla con listado de categorías (columnas: ID, Nombre, Descripción, Productos asociados, Acciones)
* Botones de acción (Editar, Eliminar)
* Botón para agregar nueva categoría
* Formulario de edición/creación de categoría

Tabla

El contenido generado por IA puede ser incorrecto.

**Mockup 10: Personalización del Sitio (Admin)**

Este mockup muestra la sección de personalización del sitio:

* Opciones para cambiar el banner principal
* Selector de esquema de colores
* Configuración de información de contacto (WhatsApp)
* Configuración de redes sociales
* Opciones de visualización de productos

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

**ANÁLISIS DE DISEÑO**

**Análisis de Diseño: Frontend (Angular con TypeScript y Tailwind CSS)**

**1. Elementos Clave del Diseño**

**a) Navbar (Componente Angular)**

* **Logo Bloom**:
  + Diseño elegante con tipografía moderna
  + Implementado como componente reutilizable
  + Clases Tailwind: font-playfair text-2xl text-rose-400 font-bold
* **Menú de Navegación**:
  + Implementado con Angular Router para navegación fluida sin recargas
  + Transiciones suaves mediante clases Tailwind: transition-all duration-300
  + Diseño responsive con menú hamburguesa en móviles
* **Iconos Funcionales**:
  + Implementados con biblioteca Angular Material o Heroicons
  + Animaciones sutiles en hover con Tailwind: hover:scale-110 transition-transform
  + Estado de carrito sincronizado mediante Angular Services

**b) Página de Inicio (Componentes Angular)**

* **Banner Principal**:
  + Implementado como Slideshow con Angular ngFor y ngClass
  + Imágenes optimizadas con lazy loading
  + Clases Tailwind: relative overflow-hidden rounded-lg shadow-xl h-96
* **Sección de Blog**:
  + Consumo de API Laravel mediante Angular HttpClient
  + Grid responsive con Tailwind: grid grid-cols-1 md:grid-cols-3 gap-6
* **Categorías**:
  + Implementado como componente carousel con control de eventos touch
  + Clases Tailwind: flex overflow-x-auto snap-x snap-mandatory
* **Productos Destacados**:
  + Componente Angular reutilizable para cards de producto
  + Sistema de rating con estrellas implementado con ngTemplateOutlet
  + Hover effects con Tailwind: group-hover:opacity-100 absolute top-0 left-0 bg-black bg-opacity-40

**c) Estilo Visual Implementado con Tailwind CSS**

* **Configuración de Tema Personalizado**:
  + Extensión del tema Tailwind en tailwind.config.js para incluir colores corporativos
  + Variables CSS personalizadas para mantener consistencia
* **Estructura Responsiva**:
  + Sistema de Grid y Flexbox mediante clases nativas de Tailwind
  + Breakpoints personalizados para optimizar la experiencia en todos los dispositivos
* **Animaciones y Transiciones**:
  + Implementadas mediante clases de transición de Tailwind
  + Estados hover/focus/active consistentes en toda la aplicación

**Análisis de Diseño: Backend (Laravel y MySQL)**

**1. Estructura de la Aplicación Laravel**

* **Modelos Eloquent**:
  + Product (con relaciones a Category)
  + Category (con relación hasManyThrough a productos)
  + BlogPost (para gestión del blog)
  + User (para autenticación y administración)
  + SiteConfig (para opciones de personalización)
* **Controladores API**:
  + ProductController (CRUD de productos)
  + CategoryController (CRUD de categorías)
  + BlogController (CRUD de entradas)
  + AuthController (autenticación mediante JWT o Sanctum)
  + SettingsController (gestión de configuración del sitio)

**2. Endpoints de API**

* **Productos**:
  + GET /api/products - Listado paginado con filtros
  + GET /api/products/{id} - Detalle de producto
  + POST /api/products - Crear producto (protegido)
  + PUT /api/products/{id} - Actualizar producto (protegido)
  + DELETE /api/products/{id} - Eliminar producto (protegido)
* **Categorías**:
  + Endpoints similares para CRUD de categorías
* **Blog**:
  + Endpoints para gestión de entradas del blog
* **Autenticación**:
  + POST /api/login - Inicio de sesión
  + POST /api/logout - Cierre de sesión
  + GET /api/me - Información del usuario

**3. Sistema de Base de Datos**

* **Migraciones Laravel**:
  + Definición del schema usando el sistema de migraciones
  + Relaciones establecidas con claves foráneas
* **Seeders**:
  + Datos de ejemplo para desarrollo y pruebas
* **Optimización**:
  + Índices en campos de búsqueda frecuente
  + Eager loading para evitar N+1 queries

Tabla

El contenido generado por IA puede ser incorrecto.

**CONCLUSIONES Y TRABAJO FUTURO**

El catálogo web Bloom ha sido desarrollado exitosamente utilizando Angular, Laravel, TypeScript y Tailwind CSS, cumpliendo con los objetivos principales de crear una plataforma administrable para la venta de joyas. La aplicación cuenta con todas las funcionalidades esenciales para la operación del negocio, incluyendo la navegación intuitiva, visualización de productos, sistema de blog y panel administrativo.

**Características implementadas:**

* Catálogo de productos con paginación y filtrado
* Vista detallada de productos y vista previa rápida
* Sistema de carrito de compras con integración a WhatsApp
* Blog para artículos y noticias
* Panel administrativo con gestión CRUD completa
* Diseño responsive para todo tipo de dispositivos

**Mejoras futuras:**

1. Elementos de marketing:
   * Implementación de barra de anuncios en la parte superior del navbar
   * Desarrollo de ventanas emergentes para promociones especiales
   * Integración de botón de chat en la esquina inferior derecha
2. Experiencia de usuario mejorada:
   * Sistema de recomendación de productos basado en historial de navegación
   * Implementación de wishlist para usuarios registrados
   * Envío de notificaciones sobre nuevos productos o promociones
3. Optimización técnica:
   * Mejora del rendimiento mediante lazy loading y optimización de imágenes
   * Implementación de PWA (Progressive Web App) para experiencia móvil mejorada
   * Integración con plataformas de analítica para seguimiento de comportamiento de usuarios

**ACCESO AL PROYECTO**

Demo en línea

El proyecto está actualmente desplegado y puede visitarse en: URL: <https://store.invite-art.com/>

Se recomienda solo explorar la aplicación sin realizar cambios en la configuración o contenido.

Instrucciones para clonar y configurar el proyecto BLOOM

1. Clonar el repositorio

git clone https://github.com/luisalbertoquino/bloom.git

cd bloom

2. Backend (Laravel)

2.1. Instalar dependencias

cd backend

composer install

2.2. Configurar entorno

cp .env.example .env

php artisan key:generate

2.3. Crear Base de Datos

\* Crear manualmente una base de datos llamada: bloom\_db

\* Editar el archivo .env para configurar los datos de conexión:

DB\_CONNECTION=mysql

DB\_HOST=127.0.0.1

DB\_PORT=3306

DB\_DATABASE=bloom\_db

DB\_USERNAME=root

DB\_PASSWORD=tu\_password

2.4. Migrar Base de Datos

php artisan migrate --seed

Las credenciales de administrador se crearán automáticamente durante la ejecución del seeder.

2.5. Crear enlace simbólico para archivos públicos

php artisan storage:link

2.6. Iniciar servidor de desarrollo

php artisan serve

3. Frontend (Angular)

3.1. Instalar dependencias

cd ../frontend

npm install

3.2. Iniciar servidor Angular

npm start

Una vez completados estos pasos, el backend estará disponible en <http://127.0.0.1:8000> y el frontend en <http://localhost:4200>.