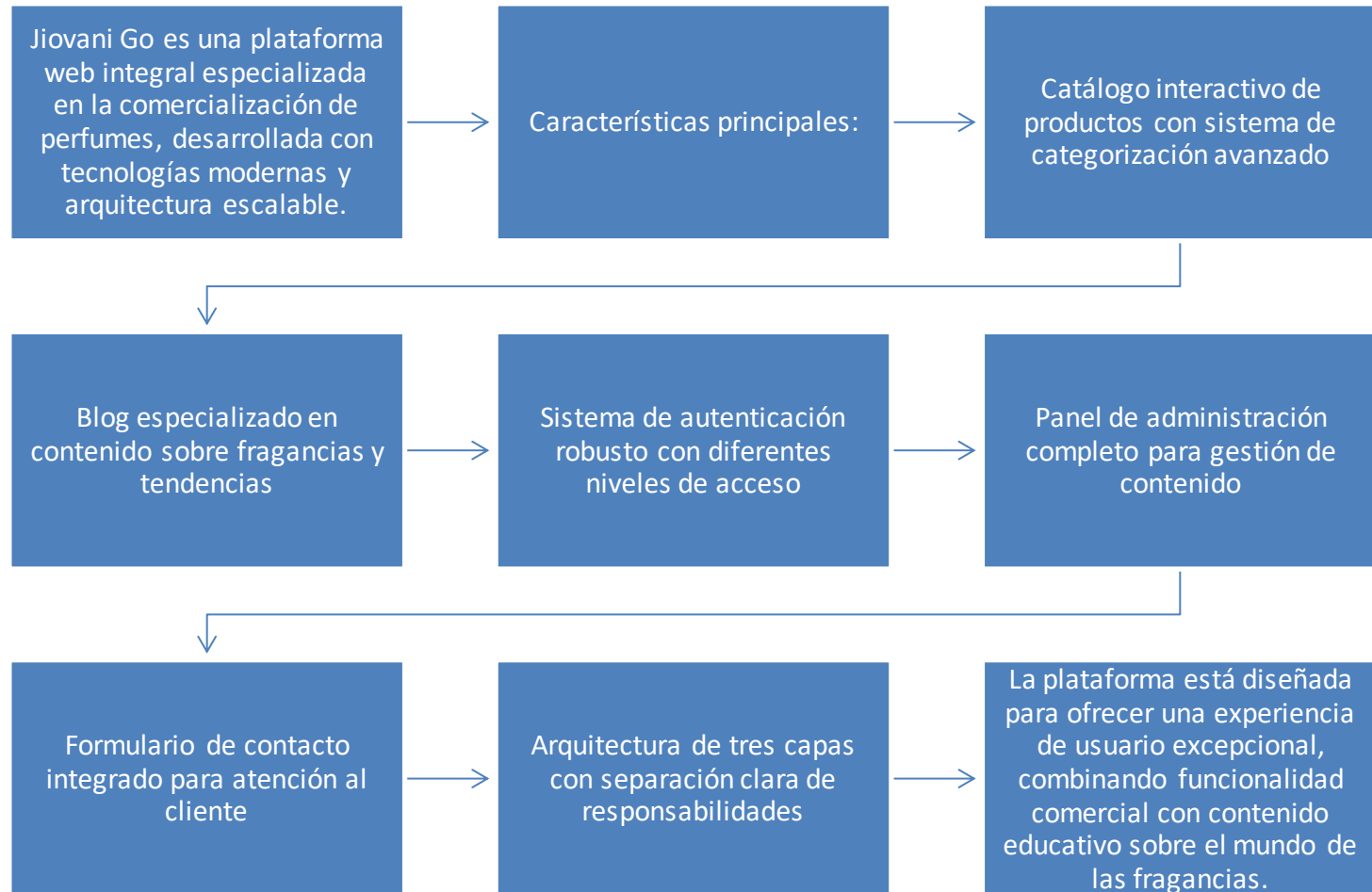


Plataforma Web Jiovani Go

Sistema de Comercialización de Fragancias

Presentación del Proyecto Final

Descripción del Proyecto



Objetivos

Objetivo General

- Desarrollar una plataforma web robusta y escalable que permita la comercialización efectiva de fragancias, integrando funcionalidades de e-commerce con un sistema de gestión de contenido especializado.

Objetivos específicos:

- Implementar un sistema de autenticación y autorización con múltiples roles de usuario
- Crear un catálogo de productos dinámico con capacidades de búsqueda y filtrado avanzadas
- Desarrollar un blog integrado para educación y marketing de contenidos
- Establecer una arquitectura de tres capas que garantice mantenibilidad y escalabilidad
- Implementar patrones de diseño que faciliten futuras expansiones del sistema
- Asegurar una experiencia de usuario fluida y responsive en todos los dispositivos

Metodología Scrum Aplicada

Visión del Producto: Plataforma líder en comercialización de fragancias premium con experiencia de usuario excepcional.



Product Backlog: Funcionalidades priorizadas incluyendo autenticación, catálogo, blog, administración y optimizaciones.



Sprints Implementados:

Sprint 1: Configuración del entorno y arquitectura base

Sprint 2: Sistema de autenticación y gestión de usuarios

Sprint 3: Desarrollo del catálogo de productos y categorías

Sprint 4: Implementación del blog y gestión de contenido



Ceremonias Scrum:

Sprint Planning: Definición de objetivos y selección de historias de usuario

Daily Stand-ups: Seguimiento diario del progreso y resolución de impedimentos

Sprint Review: Demostración de funcionalidades completadas

Sprint Retrospective: Mejora continua del proceso de desarrollo

Alcances del Proyecto

Plataforma Jiovani Go - Estado Actual de Implementación

FRONTEND PUBLICO - 100% OPERATIVO

- Catalogo completo: 24 productos con filtros por genero y familia olfativa
- Paginas de detalle especializadas con notas olfativas y concentración
- Blog educativo "El Diario de un Aroma" con contenido publicado
- Sistema de contacto con formulario funcional y mapa integrado
- Diseño responsive y navegación intuitiva completamente operativa

BACKEND Y ADMINISTRACION - 95% FUNCIONAL

- API REST completa con Node.js, Express y MongoDB optimizada
- Autenticación JWT segura y sistema de permisos granular
- Panel administrativo funcional con gestión CRUD de productos
- Sistema de usuarios, roles y gestión de contenido del blog
- Middleware de seguridad y validación de datos implementado

ESTADO GENERAL Y LIMITACIONES

- Progreso: 85% completado según especificaciones de alcance inicial
- Todas las funcionalidades core de vitrina digital operativas y listas
- Arquitectura escalable preparada para futuras expansiones planificadas
- Limitación: Carrito de compras visual únicamente, sin procesamiento de pagos
- Pendiente: Sistema de métricas y optimizaciones SEO para siguiente fase

Funcionalidades Pendientes

OPTIMIZACIONES TECNICAS PENDIENTES

- Implementacion de sistema de metricas basicas para administradores
- Optimizacion avanzada de SEO
Mejoras en el editor de blog con vista previa en tiempo real
- Implementacion de cache del lado cliente para mejor performance
- Validaciones adicionales de seguridad en formularios publicos

MEJORAS DE EXPERIENCIA DE USUARIO

- Optimizacion de carga de imagenes
Implementacion de breadcrumbs para mejor navegacion
- Sistema de busqueda inteligente con sugerencias automaticas
- Mejoras en la accesibilidad para usuarios con discapacidades
- Optimizacion especifica para dispositivos moviles

DOCUMENTACION Y TESTING FINAL

- Documentacion tecnica completa para desarrolladores
- Manual de usuario para administradores del sistema
- Testing exhaustivo de compatibilidad entre navegadores
- Validacion final de rendimiento bajo carga simulada
- Documentacion de despliegue y configuracion de produccion

Plan de Finalización

FASE 1 - COMPLETAR FUNCIONALIDADES CORE

Finalización del sistema de métricas básicas en panel administrativo
Implementación de optimizaciones SEO con meta tags dinámicos
Mejoras en editor de blog con funcionalidades de vista previa
Testing exhaustivo de todas las funcionalidades implementadas
Validación de rendimiento y optimización de consultas de base de datos

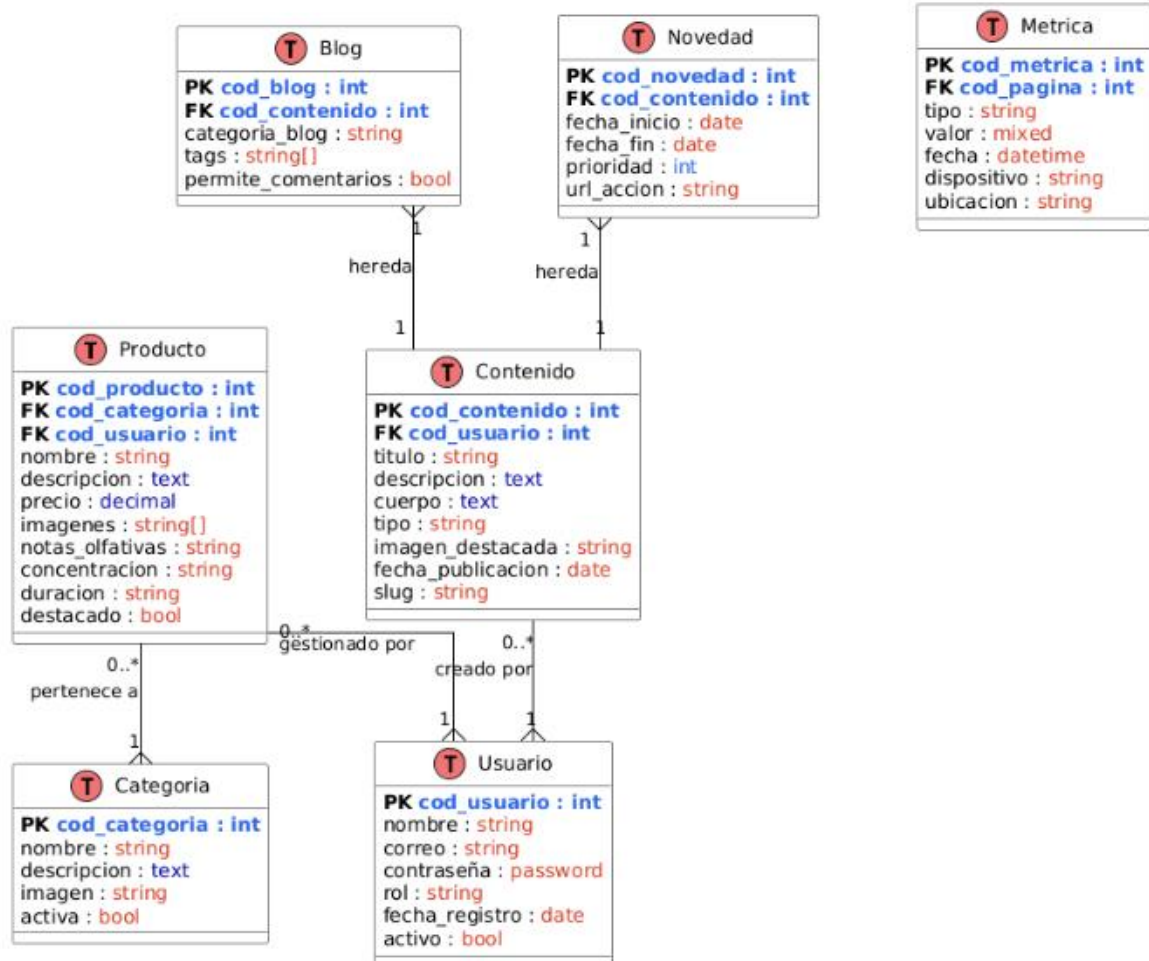
FASE 2 - OPTIMIZACION Y PULIMIENTO

Implementación de mejoras de experiencia de usuario identificadas
Optimización de performance y tiempos de carga
Validaciones adicionales de seguridad en todos los formularios
Testing de compatibilidad entre diferentes navegadores y dispositivos
Corrección de errores menores y refinamiento de interfaz

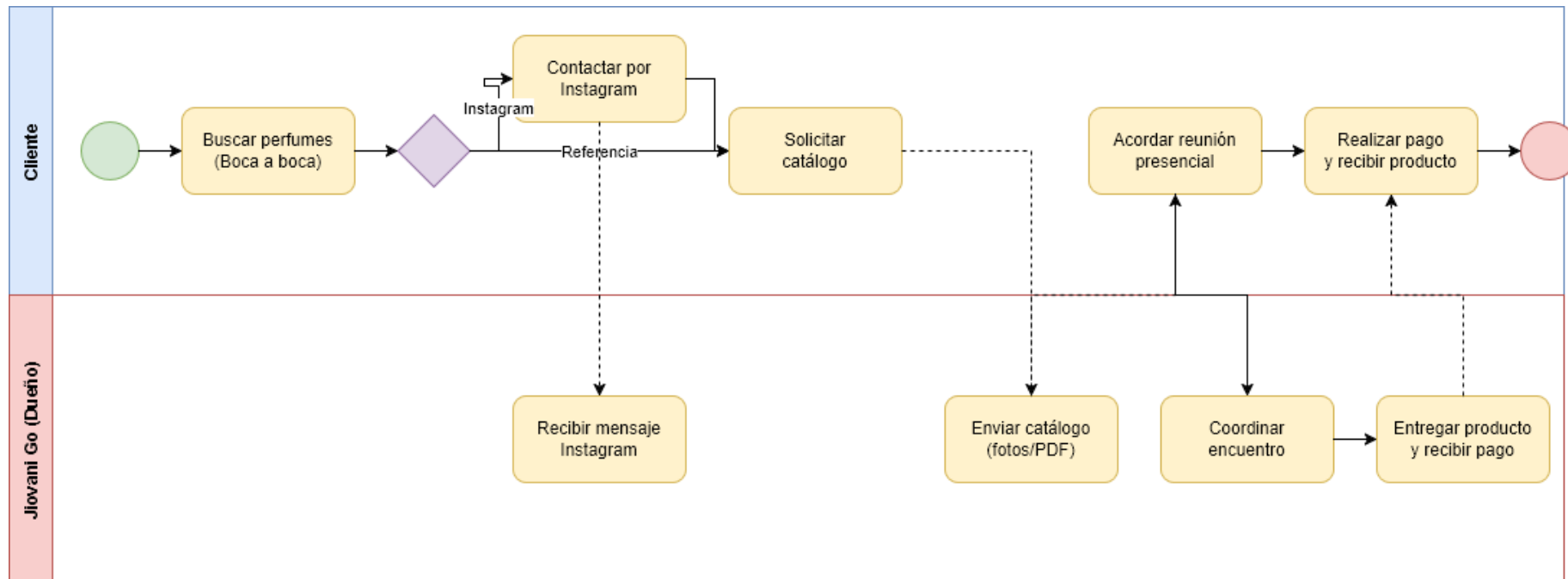
FASE 3 - DOCUMENTACION Y ENTREGA

Elaboración de documentación técnica completa para mantenimiento
Creación de manual de usuario para administradores del sistema
Configuración final del entorno de producción con medidas de seguridad
Entrega de credenciales y transferencia de conocimiento al cliente
Validación final con cliente y cierre formal del proyecto

Diagrama de Base de Datos



Diagramas BPMN



Diagramas BPMN Mejorado

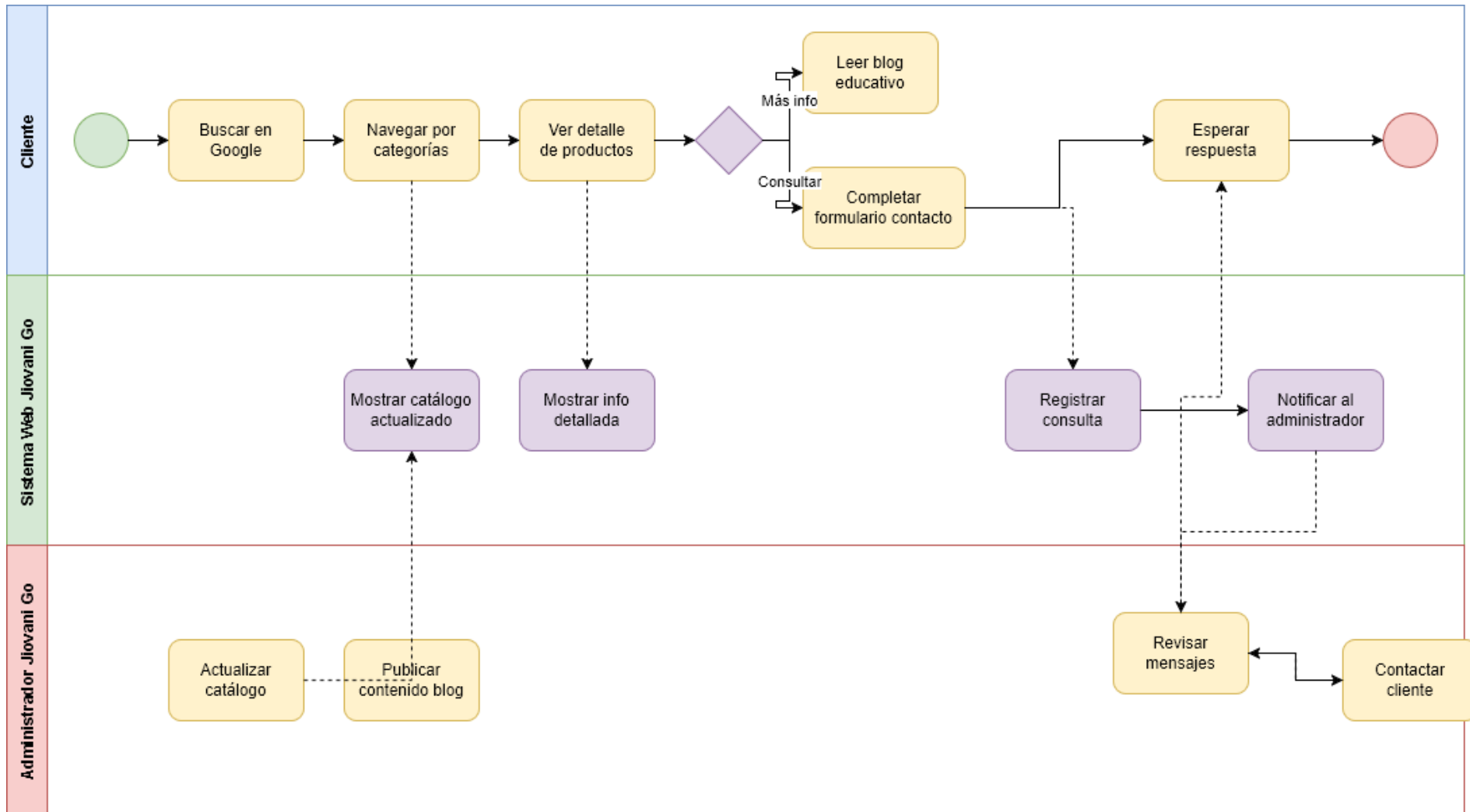


Diagrama de Casos de Uso

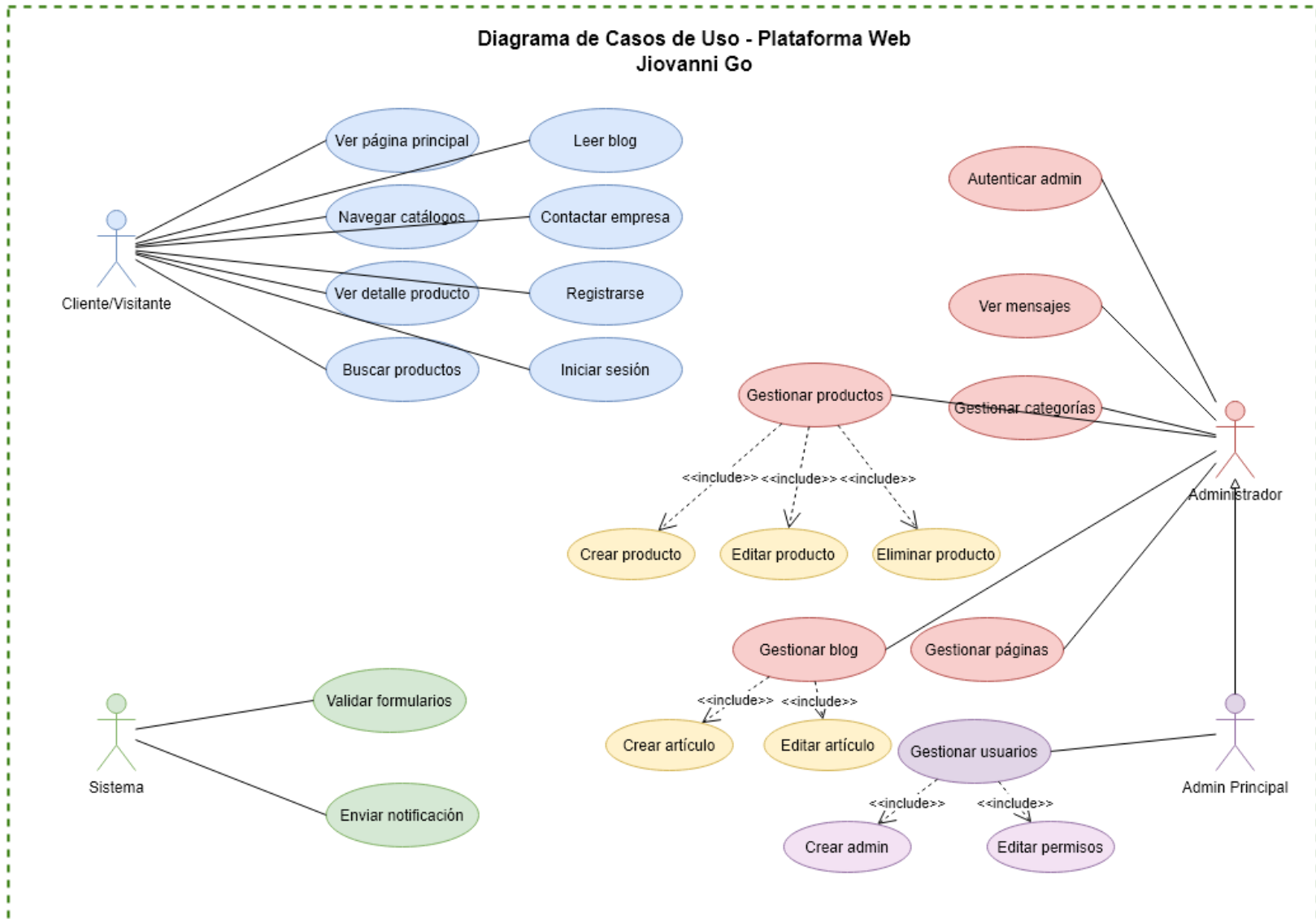
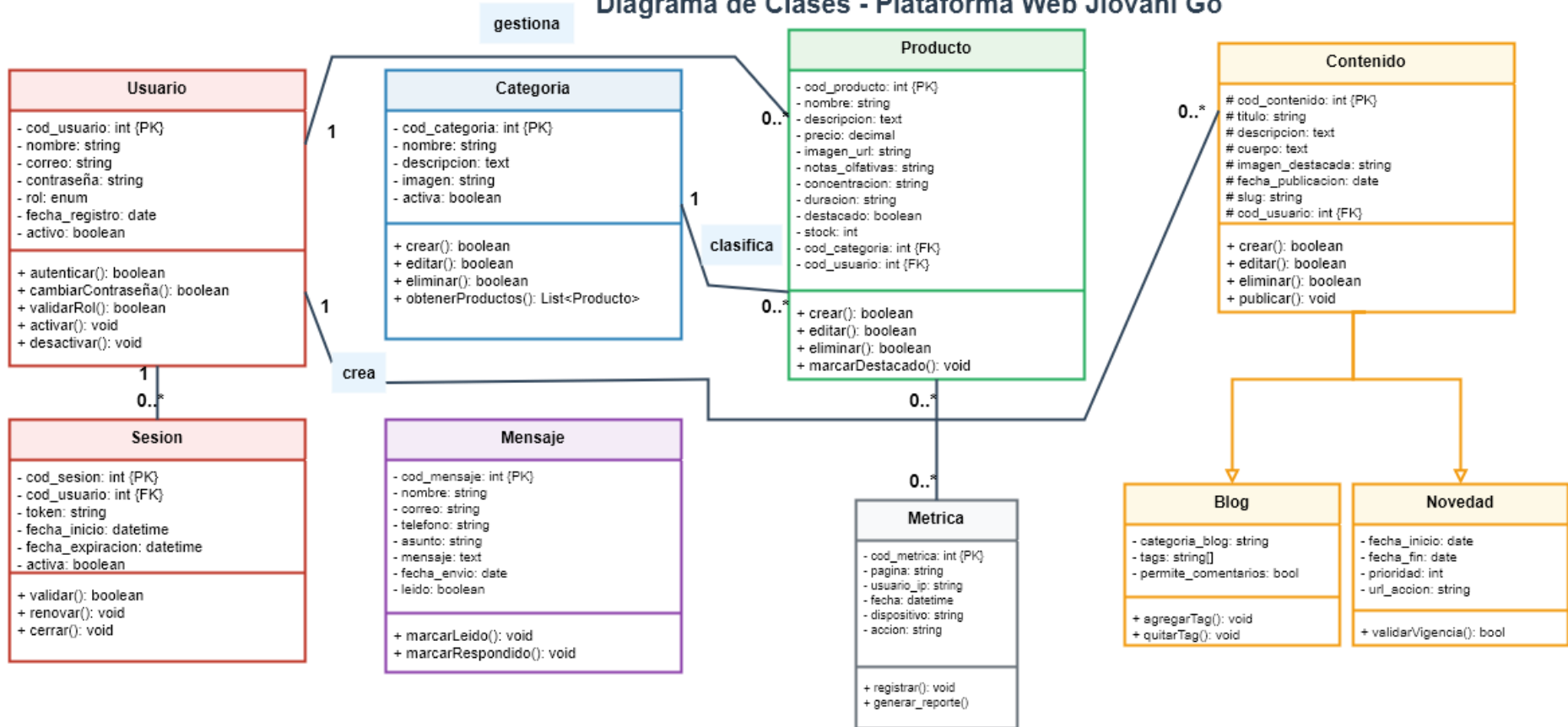


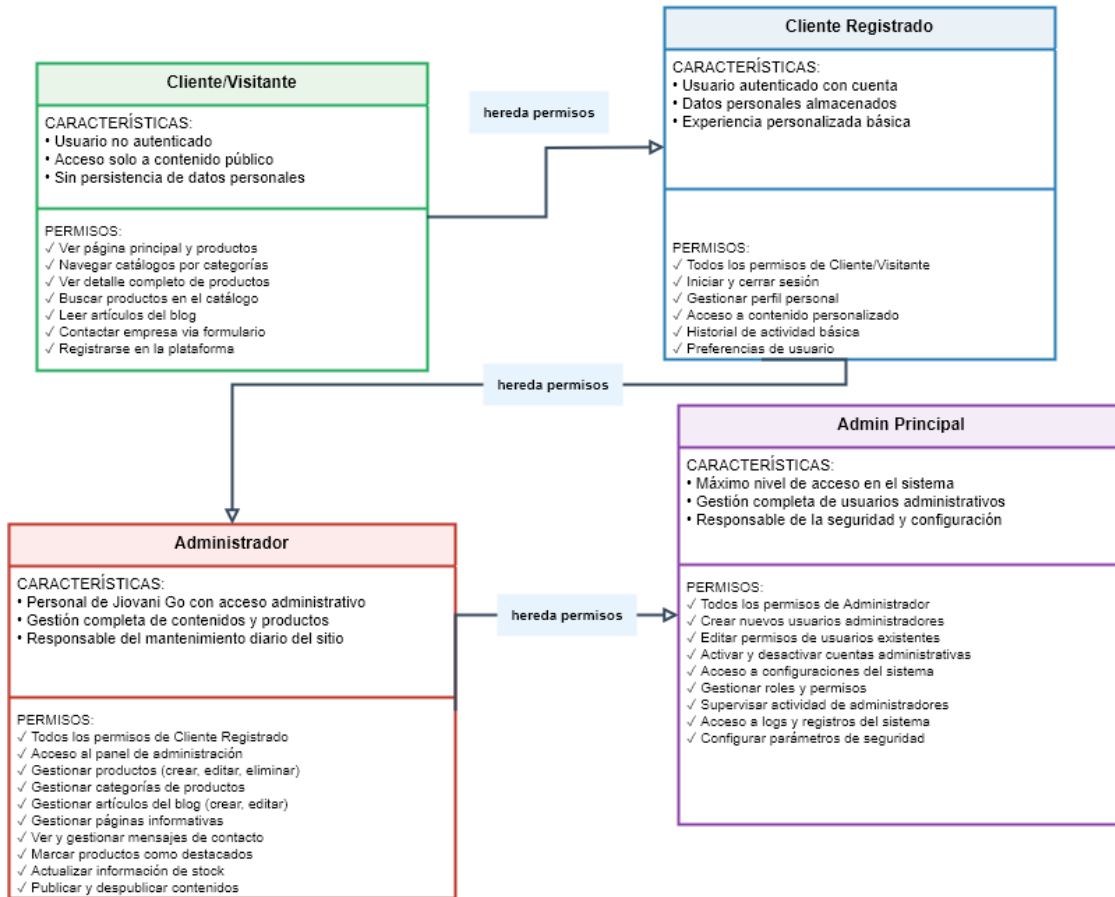
Diagrama de Clases

Diagrama de Clases - Plataforma Web Jiovani Go



Herencia de Roles y Permisos

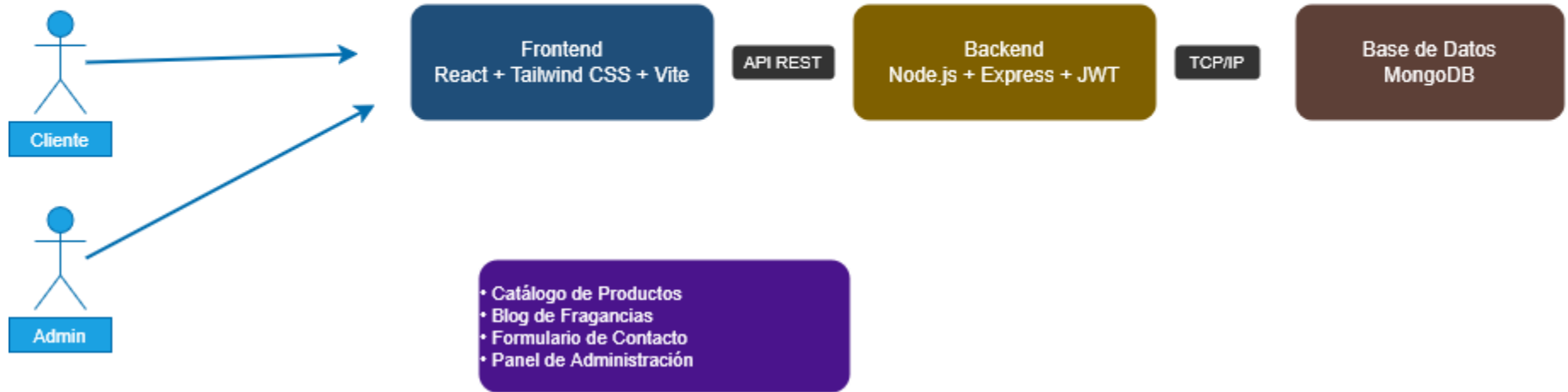
Diagrama de Roles y Permisos - Plataforma Web Jiovani Go



MATRIZ DE PERMISOS POR FUNCIONALIDAD

FUNCIONALIDAD	VISITANTE	REGISTRADO	ADMIN	ADMIN PRIN.
Ver productos y catálogo	✓	✓	✓	✓
Leer blog y contenidos	✓	✓	✓	✓
Contactar empresa	✓	✓	✓	✓
Iniciar sesión	X	✓	✓	✓
Gestionar productos	X	X	✓	✓
Gestionar blog	X	X	✓	✓
Ver mensajes contacto	X	X	✓	✓
Gestionar páginas	X	X	✓	✓
Gestionar categorías	X	X	✓	✓
Gestionar usuarios admin	X	X	X	✓
Configurar sistema	X	X	X	✓

Arquitectura del Sistema



Patrones de Diseño

Patrones Implementados

- **MVC (Model-View-Controller):**
 - Separación clara entre datos, lógica y presentación con MongoDB, Express y React
 - Controladores especializados para gestión de productos, usuarios y contenido
- **Repository Pattern:**
 - Abstracción de acceso a datos con interfaces consistentes para operaciones CRUD
 - Facilita testing y mantenimiento del código de base de datos
- **Middleware Pattern:**
 - Autenticación JWT y validación de datos centralizadas
 - Sistema de logging y manejo de errores implementado

Patrones Pendientes de Implementación:

- **Service Layer Pattern:**
 - Encapsulación de lógica de negocio compleja para reutilización entre controladores
- **Observer Pattern:**
 - Sistema de notificaciones automáticas y alertas para administradores

Stack Tecnológico



Frontend:

- React 18 - Framework principal para UI
- Tailwind CSS - Framework de estilos utilitarios
- Vite - Build tool y desarrollo rápido
- Lucide React - Iconografía moderna

Backend:

- Node.js - Runtime de JavaScript
- Express.js - Framework web minimalista
- JWT - Autenticación y autorización
- Bcrypt - Hashing de contraseñas
- Cors - Manejo de políticas de origen cruzado

Base de Datos:

- MongoDB - Base de datos NoSQL
- Mongoose - ODM para modelado de datos

Herramientas de Desarrollo:

- Git - Control de versiones
- ESLint - Linting de código
- Prettier - Formateo de código
- Postman - Testing de API

Conclusión

Logros Alcanzados:

- Desarrollo exitoso de una plataforma web funcional especializada en fragancias
- Implementación de arquitectura escalable y mantenible
- Sistema de autenticación robusto con roles granulares
- Base sólida para futuras expansiones comerciales

Lecciones Aprendidas:

- La metodología Scrum facilita el desarrollo iterativo y la adaptación a cambios
- La separación en capas mejora significativamente la mantenibilidad del código
- La planificación detallada de la base de datos es crucial para el rendimiento
- Las pruebas continuas reducen los errores en producción

Próximos pasos:

- Implementar sistema de métricas y analytics básicos para administradores
- Optimizar la experiencia de usuario en dispositivos móviles
- Completar optimizaciones SEO para mejor posicionamiento web