

DOCUMENTACIÓN TÉCNICA Y FUNCIONAL

Sistema NovelasApp

Versión 1.0

Generado el: 01/04/2025 18:16:56

ÍNDICE DE CONTENIDOS

1. INTRODUCCIÓN

- 1.1 Descripción General
- 1.2 Propósito y Alcance
- 1.3 Objetivos del Sistema

2. DOCUMENTACIÓN FUNCIONAL

- 2.1 Módulo de Usuarios
- 2.2 Módulo de Novelas
- 2.3 Módulo de Capítulos
- 2.4 Módulo de Géneros
- 2.5 Módulo de Interacción
- 2.6 Módulo de Imágenes
- 2.7 Perfiles de Usuario
- 2.8 Flujos de Trabajo
- 2.9 Interfaces de Usuario
- 2.10 Requisitos No Funcionales

3. DOCUMENTACIÓN TÉCNICA

- 3.1 Arquitectura del Sistema
- 3.2 Stack Tecnológico
- 3.3 Modelos y Base de Datos
- 3.4 Estructura del Código
- 3.5 API y Puntos de Entrada
- 3.6 Seguridad
- 3.7 Implementación de Características
- 3.8 Configuración y Despliegue

1. INTRODUCCIÓN

1.1 Descripción General

NovelasApp es una plataforma web especializada para la creación, publicación y lectura de novelas en línea. Desarrollada con Laravel 10, el sistema permite a los escritores compartir sus obras por capítulos y a los lectores descubrir y seguir historias según sus géneros preferidos.

1.2 Propósito y Alcance

El propósito principal de NovelasApp es proporcionar un espacio donde:

- Los escritores puedan publicar sus novelas capítulo a capítulo
- Los lectores puedan descubrir y disfrutar de nuevas historias
- Se fomente la comunidad literaria mediante interacciones básicas
- Se mantenga un catálogo organizado por géneros literarios

1.3 Objetivos del Sistema

- Facilitar la creación y publicación de contenido literario
- Ofrecer una experiencia de lectura limpia y agradable
- Proporcionar herramientas para organizar y gestionar novelas
- Permitir la interacción básica entre usuarios y contenidos
- Establecer un sistema de categorización eficiente

2. DOCUMENTACIÓN FUNCIONAL

Documentación Funcional - Sistema NovelasApp

1. Introducción al Sistema

1.1 Descripción General

NovelasApp es una plataforma web especializada para la creación, publicación y lectura de novelas en línea. El sistema permite a los escritores compartir sus obras por capítulos y a los lectores descubrir y seguir historias según sus géneros preferidos.

1.2 Propósito y Alcance

El propósito principal de NovelasApp es proporcionar un espacio donde:

- Los escritores puedan publicar sus novelas capítulo a capítulo
- Los lectores puedan descubrir y disfrutar de nuevas historias
- Se fomente la comunidad literaria mediante interacciones básicas
- Se mantenga un catálogo organizado por géneros literarios

1.3 Objetivos del Sistema

- Facilitar la creación y publicación de contenido literario
- Ofrecer una experiencia de lectura limpia y agradable
- Proporcionar herramientas para organizar y gestionar novelas
- Permitir la interacción básica entre usuarios y contenidos
- Establecer un sistema de categorización eficiente

2. Funcionalidades del Sistema

2.1 Módulo de Usuarios

2.1.1 Registro e Inicio de Sesión

- ****Registro de usuario****: Permite a visitantes crear una cuenta proporcionando nombre, correo electrónico y contraseña
- ****Inicio de sesión****: Autentica usuarios mediante correo electrónico y contraseña
- ****Recuperación de contraseña****: Permite restablecer contraseña vía email

- **Recordar sesión**: Opción para mantener la sesión activa entre visitas

2.1.2 Gestión de Perfil

- **Ver perfil**: Visualización de datos personales y estadísticas
- **Editar perfil**: Actualización de datos personales
- **Cambiar contraseña**: Modificación de credenciales de acceso
- **Eliminar cuenta**: Opción para dar de baja la cuenta

2.2 Módulo de Novelas

2.2.1 Gestión de Novelas

- **Creación de novela**: Formulario para introducir título, descripción, portada y géneros
- **Listado de novelas**: Vista con todas las novelas del usuario
- **Edición de novela**: Modificación de datos existentes
- **Eliminación de novela**: Opción para eliminar obra completa
- **Estadísticas**: Visualización de datos de lectura y engagement

2.2.2 Exploración de Novelas

- **Catálogo general**: Listado de todas las novelas disponibles
- **Filtrado por género**: Visualización de novelas por categoría
- **Búsqueda por texto**: Localización de novelas por palabras clave
- **Novelas destacadas**: Selección de obras con mayor interacción

2.3 Módulo de Capítulos

2.3.1 Gestión de Capítulos

- **Creación de capítulo**: Editor de texto enriquecido para nuevo contenido
- **Listado de capítulos**: Vista de todos los capítulos de una novela
- **Edición de capítulo**: Modificación de contenido existente
- **Eliminación de capítulo**: Opción para eliminar contenido
- **Ordenación de capítulos**: Sistema de numeración automática

2.3.2 Lectura de Capítulos

- **Visualización de capítulo**: Interfaz de lectura limpia
- **Navegación entre capítulos**: Botones para capítulo anterior/siguiente
- **Interacción**: Opciones para guardar y dar "me gusta"
- **Capítulos guardados**: Lista de capítulos marcados para lectura posterior

2.4 Módulo de Géneros

- **Gestión de géneros**: Administración de categorías literarias
- **Asignación a novelas**: Relación de novelas con múltiples géneros
- **Exploración por género**: Navegación por categorías

2.5 Módulo de Interacción

- **Me gusta**: Opción para marcar capítulos favoritos
- **Guardar capítulo**: Función para guardar capítulos para lectura posterior
- **Capítulos guardados**: Listado de capítulos guardados por el usuario

2.6 Módulo de Imágenes

- **Subida de portadas**: Carga de imágenes para novelas
- **Imágenes en capítulos**: Inserción de imágenes en el contenido
- **Gestión de imágenes**: Organización y eliminación de recursos

3. Perfiles de Usuario

3.1 Tipos de Usuario

3.1.1 Usuario Anónimo (No Registrado)

Capacidades:

- Navegar por novelas y capítulos públicos
- Ver detalles básicos de las novelas
- Leer capítulos publicados
- Registrarse en la plataforma

Limitaciones:

- No puede crear contenido
- No puede interactuar con el contenido existente
- No tiene acceso a funciones de guardar o dar "me gusta"

3.1.2 Usuario Registrado (Autor/Lector)

Capacidades:

- Todas las funciones del usuario anónimo
- Crear y gestionar sus propias novelas
- Gestionar capítulos de sus novelas
- Dar "me gusta" a capítulos

- Guardar capítulos para lectura posterior
- Visualizar sus contenidos guardados
- Gestionar su perfil

****Limitaciones**:**

- No puede modificar contenido de otros usuarios
- No tiene acceso a la gestión de géneros

3.1.3 Administrador

****Capacidades**:**

- Todas las funciones del usuario registrado
- Gestión de géneros literarios
- Acceso a estadísticas generales
- Gestión de usuarios y contenido

****Limitaciones**:**

- No definidas (acceso completo)

4. Flujos de Trabajo

4.1 Creación de una Nueva Novela

1. El usuario inicia sesión en el sistema
2. Accede a la sección "Mis Novelas" en el dashboard
3. Hace clic en "Crear nueva novela"
4. Completa el formulario con título, descripción y géneros
5. Opcionalmente, sube una imagen de portada
6. Envía el formulario para crear la novela
7. Es redirigido a la página de detalles de la novela creada
8. Desde allí, puede empezar a crear capítulos

4.2 Publicación de un Capítulo

1. El usuario accede a una de sus novelas
2. Hace clic en "Añadir capítulo"
3. Introduce título y número de capítulo
4. Utiliza el editor para escribir el contenido
5. Opcionalmente, añade imágenes al contenido
6. Guarda el capítulo (como borrador o publicado)
7. Es redirigido a la vista de capítulos de la novela

4.3 Lectura y Guardado de Capítulos

1. El usuario navega por el catálogo o búsqueda
2. Selecciona una novela de interés
3. Visualiza los capítulos disponibles
4. Elige un capítulo para leer
5. Lee el contenido del capítulo
6. Puede dar "me gusta" o guardar el capítulo
7. Navega al siguiente capítulo o vuelve al listado

5. Interfaces de Usuario

5.1 Diseño General

La interfaz de NovelasApp sigue un diseño moderno y limpio, utilizando TailwindCSS para la maquetación y estilos. Los elementos visuales clave incluyen:

- Navegación intuitiva con menú principal y submenús contextuales
- Paneles de información con tarjetas para cada novela
- Formularios con validación en tiempo real
- Editor de texto enriquecido para creación de contenido
- Diseño responsivo adaptable a diferentes dispositivos

5.2 Pantallas Principales

5.2.1 Dashboard

Panel central desde donde los usuarios acceden a:

- Listado de sus novelas
- Opción para crear nueva novela
- Acceso a capítulos guardados
- Estadísticas básicas de sus obras

5.2.2 Listado de Novelas

Muestra las novelas disponibles con:

- Imagen de portada
- Título y descripción breve
- Géneros asociados
- Autor
- Estadísticas básicas (capítulos, lecturas)

5.2.3 Editor de Capítulos

Interfaz para crear y editar capítulos con:

- Campo para título
- Editor TinyMCE para contenido enriquecido
- Herramientas para formateo de texto e inserción de imágenes
- Opciones para guardar como borrador o publicar

5.2.4 Visualización de Capítulo

Interfaz de lectura con:

- Título del capítulo
- Contenido formateado
- Navegación entre capítulos
- Botones para interacción (me gusta, guardar)

6. Requisitos No Funcionales

6.1 Rendimiento

- Tiempo de carga de página inferior a 3 segundos
- Soporte para almacenamiento de imágenes optimizado
- Capacidad para manejar al menos 100 usuarios concurrentes

6.2 Seguridad

- Autenticación segura con Laravel Breeze
- Protección contra ataques CSRF
- Validación de datos en formularios
- Almacenamiento seguro de contraseñas con hash

6.3 Usabilidad

- Interfaz intuitiva y accesible
- Diseño responsivo para diferentes dispositivos
- Mensajes de error claros y orientativos
- Ayudas contextuales en formularios complejos

6.4 Escalabilidad

- Estructura modular para facilitar expansión
- Base de datos optimizada para crecimiento

- Arquitectura preparada para añadir nuevas funcionalidades

3. DOCUMENTACIÓN TÉCNICA

Documentación Técnica - Sistema NovelasApp

1. Arquitectura del Sistema

1.1 Visión General

NovelasApp es una aplicación web desarrollada con Laravel que permite a los usuarios leer, escribir y compartir novelas online. La plataforma proporciona una interfaz intuitiva para los escritores, permitiéndoles publicar sus obras por capítulos, mientras los lectores pueden descubrir, guardar y dar me gusta a las novelas que les interesen.

1.2 Arquitectura Técnica

El sistema sigue una arquitectura Model-View-Controller (MVC) implementada a través del framework Laravel 10:

- **Capa de Presentación**: Implementada con Laravel Blade y TailwindCSS
- **Capa de Negocio**: Controladores Laravel organizados por funcionalidad
- **Capa de Datos**: Modelos Eloquent con relaciones definidas
- **Capa de Persistencia**: Base de datos MySQL para almacenamiento

Diagrama de Arquitectura

...

??

? Cliente (Navegador) ?

??

?

??

? Laravel Application ?

? ?????????????? ?????????????? ?????????????? ?

? ? Views ? ? Controllers? ? Models ? ?

? ? (Blade) ?????? (HTTP) ?????? (Eloquent) ? ?

? ?????????????? ?????????????? ?????????????? ?

??

?

? MySQL Database ?

??

...

1.3 Stack Tecnológico

- **Backend**: PHP 8.1, Laravel 10
- **Frontend**: HTML5, CSS3, JavaScript, TailwindCSS, Alpine.js
- **Base de Datos**: MySQL
- **Autenticación**: Laravel Breeze
- **Editor de Texto**: TinyMCE para edición de capítulos
- **Almacenamiento**: Sistema de archivos local para imágenes

2. Modelos y Base de Datos

2.1 Diagrama Entidad-Relación

...

??

? Users ? ? Novelas ? ? Capítulos ?

??

? id ? ? id ? ? id ?

? name ? ? titulo ? ? titulo ?

? email ???1??<>? user_id ???1??<>? novela_id ?

? password ? ? ? descripcion? ? ? contenido ?

? ... ? ? ? portada ? ? ? numero ?

???????????????????? ? ? ... ? ? ? ... ?

? ????????????????? ? ?????????????????

? ? ? ?

? ? ? ?

? ? ? ?

? ????????????????? ? ?????????????????

? ? NovelaGenero ? ?CapituloImagen?

? ????????????????? ? ?????????????????

? ? novela_id ? ? ?capitulo_id?

? ? genero_id ? ? ? path ?

? ????????????????? ? ?????????????????

? ? ?

? ? ?

? ????????????????? ?

???>? Generos-2??? Documentación Técnica y Funcional - Página {PAGE_NUM} de {PAGE_COUNT}

????????????????

? id ?
? nombre ?
? ... ?
?????????????
...

2.2 Descripción de Tablas Principales

Users

Almacena información de usuarios, incluyendo credenciales de autenticación y datos personales.

Novelas

Contiene información sobre las novelas, incluyendo título, descripción y referencia al autor.

Capítulos

Almacena capítulos individuales asociados a cada novela, con su contenido y numeración.

Generos

Catálogo de géneros literarios disponibles para clasificar novelas.

Tablas Pivot

- ****novela_genero****: Relación muchos a muchos entre novelas y géneros
- ****capitulo_guardado****: Registro de capítulos guardados por usuarios
- ****capitulo_like****: Registro de "me gusta" en capítulos

2.3 Relaciones Clave

- ****Usuario ? Novelas****: Un usuario puede tener muchas novelas (1:N)
- ****Novela ? Capítulos****: Una novela contiene muchos capítulos (1:N)
- ****Novela ? Géneros****: Una novela puede tener múltiples géneros, y un género puede estar asociado a múltiples novelas (N:M)
- ****Capítulo ? Imágenes****: Un capítulo puede contener múltiples imágenes (1:N)

3. Estructura del Código

3.1 Organización de Directorios

...
novelas-app/
??? app/
? ??? [Http](#)NovelasApp - Documentación Técnica y Funcional - Página {PAGE_NUM} de {PAGE_COUNT}
? ? ??? Controllers/ # Controladores de la aplicación

? ? ??? Middleware/ # Middleware de autenticación y permisos
 ? ? ??? Requests/ # Validación de formularios
 ? ??? Models/ # Modelos Eloquent
 ? ??? Providers/ # Proveedores de servicios
 ??? config/ # Configuración de la aplicación
 ??? database/
 ? ??? migrations/ # Migraciones de la base de datos
 ? ??? seeders/ # Datos iniciales
 ??? public/ # Archivos accesibles públicamente
 ? ??? storage/ # Imágenes y archivos subidos
 ??? resources/
 ? ??? css/ # Estilos CSS/TailwindCSS
 ? ??? js/ # Código JavaScript
 ? ??? views/ # Vistas Blade
 ? ??? auth/ # Vistas de autenticación
 ? ??? layouts/ # Plantillas de diseño
 ? ??? novelas/ # Vistas de novelas
 ? ??? capitulos/ # Vistas de capítulos
 ? ??? publico/ # Vistas de acceso público
 ??? routes/ # Definición de rutas
 ? ??? web.php # Rutas principales de la aplicación
 ??? storage/ # Almacenamiento de la aplicación
 ...

3.2 Patrones de Diseño Implementados

- **Repository Pattern**: Abstracción de la capa de datos
- **Service Layer**: Lógica de negocio separada de los controladores
- **Factory Method**: Creación de modelos
- **Middleware**: Filtrado de peticiones HTTP

4. API y Puntos de Entrada

4.1 Rutas Principales

Método	URI	Controlador	Función
GET	/	PublicoController	inicio
GET	/novelas	NovelaController	index
GET	/novelas/create	NovelaController	create
POST	/novelas	NovelaController	store
GET	/novelas/{novela}/capitulos	CapituloController	index

Método	URI	Controlador	Función	
GET	/	PublicoController	inicio	Página principal
GET	/novelas	NovelaController	index	Listar novelas del usuario
GET	/novelas/create	NovelaController	create	Formulario de creación
POST	/novelas	NovelaController	store	Guardar nueva novela
GET	/novelas/{novela}/capitulos	CapituloController	index	Listar capítulos

| POST | /capitulos/{capitulo}/like | CapituloInteraccionController@toggleLike | Dar/quitar "me gusta" |

4.2 Middlewares

- ****auth****: Verificación de autenticación
- ****verified****: Verificación de email
- ****admin****: Restricción para administradores

5. Seguridad

5.1 Autenticación y Autorización

- Autenticación basada en Laravel Breeze
- Sistema de roles básico (usuario/administrador)
- Políticas de autorización por recurso

5.2 Protección de Datos

- Validación de formularios en servidor
- Protección CSRF en todos los formularios
- Almacenamiento seguro de contraseñas con bcrypt
- Sanitización de entrada de datos

5.3 Manejo de Sesiones

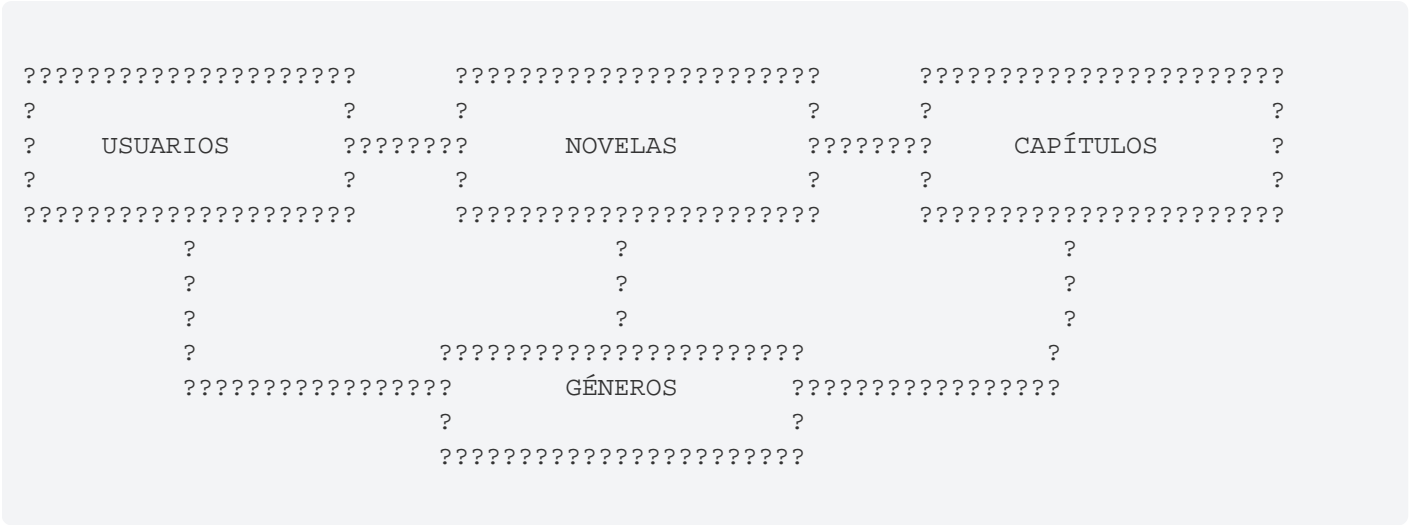
- Sesiones con tiempo de expiración
- Rotación de tokens CSRF
- Opción "recordarme" para prolongar sesiones

ANEXOS

A.1 Glosario de Términos

Término	Definición
Novela	Obra literaria narrativa de cierta extensión, compuesta por capítulos.
Capítulo	División de una novela que agrupa un conjunto específico de eventos dentro de la misma.
Género	Categoría que clasifica las novelas según su temática y características.
Laravel	Framework de PHP utilizado para el desarrollo de la aplicación.
MVC	Patrón de diseño Modelo-Vista-Controlador implementado en la aplicación.
Middleware	Mecanismo de filtrado de peticiones HTTP en Laravel.
Blade	Motor de plantillas de Laravel utilizado para las vistas.
Eloquent	ORM (Object-Relational Mapping) utilizado por Laravel para interactuar con la base de datos.

A.2 Diagrama del Sistema



A.3 Acerca de la Documentación

Este documento ha sido generado automáticamente mediante el sistema de documentación integrado en NovelasApp. La información presentada corresponde al análisis detallado del sistema en su estado actual al momento de la generación (01/04/2025 18:16:56).

Para actualizar esta documentación, utilice el comando Artisan:

```
php artisan documentation:profesional
```

Para cualquier consulta sobre el sistema o su documentación, contacte con el administrador del sistema.