

Movie Website KarenFlix

**Harley Yefrey Cabrales Vargas
David Santiago Vera Mendez**

S1

PEDRO FELIPE GÓMEZ BONILLA

**CAMPUSLANDS
RUTA NODE.JS
BUCARAMANGA
2025**

Movie Website KarenFlix

1. SITUACIÓN PROBLEMA

En la actualidad, los fanáticos del cine, las series y el anime suelen utilizar múltiples plataformas para registrar, recomendar y calificar el contenido que consumen. Sin embargo, estas plataformas presentan varios inconvenientes: algunas no permiten una interacción real entre usuarios, otras restringen la creación de reseñas a perfiles verificados, y la mayoría no ofrecen un sistema transparente de ranking que combine calificaciones, reseñas y reacciones de la comunidad.

Esto genera una dificultad para los usuarios que desean encontrar recomendaciones confiables y organizadas de acuerdo con sus gustos, ya que la información se dispersa en diferentes aplicaciones o redes sociales. Por otra parte, los administradores carecen de herramientas simples para gestionar categorías, validar contenido y mantener la calidad de las reseñas publicadas.

Ante esta necesidad, surge la propuesta de **KarenFlix**, una plataforma web full-stack que permitirá a los usuarios registrarse, calificar, reseñar y rankear películas, animes y series geek, con un sistema de autenticación seguro, validaciones robustas, un ranking ponderado y una interfaz amigable que centralice toda la información en un solo lugar.

2. LEVANTAMIENTO DE REQUERIMIENTOS

El levantamiento de requerimientos del proyecto **KarenFlix** se realizó a través de entrevistas con usuarios potenciales y revisión de plataformas similares, con el fin de identificar sus necesidades, problemas y expectativas. Posteriormente, el equipo consolidó la información obtenida y la priorizó en sesiones de trabajo colaborativo, estableciendo la base para definir los requerimientos funcionales y no funcionales del sistema.

3. REQUERIMIENTOS

Gestión de usuarios

- Registro, inicio de sesión y autenticación mediante JWT.
- Roles: usuario y administrador.
- Los administradores pueden gestionar categorías y aprobar películas.

Gestión de películas y series

- CRUD de películas/series-
- Validación para evitar títulos repetidos.
- Atributos mínimos: título, descripción, categoría, estreno, podter.



Gestión de reseñas y ratings

- Los usuarios pueden crear, editar y eliminar reseñas.
- Cada reseña incluye: título, comentario, calificación numérica del 1 al 10.
- Los usuarios pueden dar like/dislike a reseñas de otros
- El sistema debe calcular un ranking ponderado de películas basado en calificaciones.

Ranking y listados

- Listado de películas con ordenamiento por popularidad y ranking.
- Filtrado por categoría.
- Vista de detalle con información y reseñas asociadas.

Seguridad y autenticación

- Implementación de JWT con passport-jwt y bcrypt.
- Rate limiting para prevenir abusos.
- Validaciones robustas con express-validator.

Gestión de base de datos

- MongoDB con transacciones para operaciones críticas.
- Consistencia en operaciones de reseñas con likes/dislikes.
- Validación de duplicadas

3.1. Requerimientos Funcionales

RF-01. El sistema permitirá el registro de usuarios con roles diferenciados de usuario y administrador.

RF-02. El sistema permitirá el inicio de sesión mediante autenticación con JWT usando correo y contraseña.

RF-03. El sistema permitirá a los administradores gestionar las categorías de contenido disponibles en la plataforma.

RF-04. Creación, visualización, actualización y eliminación de películas.

RF-05. El sistema validará que no existan títulos duplicados en la base de datos.

RF-06. El sistema permitirá a los usuarios crear reseñas con, comentario y calificación de 1 a 10

RF-07. El sistema permitirá a los usuarios editar y eliminar sus propias reseñas.

RF-08. El sistema permitirá a los usuarios dar like o dislike a reseñas de otros usuarios, pero no a las propias.



RF-09. El sistema calculará automáticamente rankings ponderados de contenido basado en calificaciones, likes/dislikes.

RF-10. El sistema proporcionará listados de contenido ordenados por popularidad y ranking.

RF-11. El sistema mostrará vistas detalladas de cada película, serie o anime con toda su información y reseñas asociadas.

RF-12. El sistema permitirá búsquedas de contenido por título.

3.2. Requerimientos No Funcionales

RNF-01. El sistema utilizará bcrypt para el hashado seguro de contraseñas antes de almacenarlas en la base de datos.

RNF-02. El sistema implementará autenticación mediante JWT con tokens que expiren automáticamente.

RNF-03. El sistema utilizará express-rate-limit para prevenir ataques de fuerza bruta y limitar peticiones por IP.

RNF-04. El sistema utilizará MongoDB como base de datos principal con el driver oficial de Node.js.

RNF-05. El sistema implementará transacciones MongoDB para operaciones críticas como creación de reseñas y actualización de rating.

RNF-06. El sistema documentará todos los endpoints mediante swagger-ui-express.

RNF-07. El sistema versionará la API siguiendo las convenciones de Semantic Versioning.

RNF-08. El sistema utilizará variables de entorno mediante dotenv para toda la configuración sensible.

RNF-09. El sistema será responsive y funcionará correctamente en dispositivos móviles y desktop.

4. HISTORIAS DE USUARIO CON CRITERIOS DE ACEPTACIÓN

1.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	RF-01	Actor	Usuario
NOMBRE DEL REQUERIMIENTO	Registro de usuarios		
Descripción			
Como usuario puedo registrarme en KarenFlix como usuario			

Funcionalidad	
Registro de usuarios con asignación de roles y permisos diferenciados.	
Criterios de aceptación	1. La contraseña debe estar en la base de datos hasheada. 2. El correo electrónico debe ser único en el sistema.
Restricciones	
El correo debe ser unico	

2.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	RF-02	Actor	Usuario
NOMBRE DEL REQUERIMIENTO	Inicio de sesión con JWT		
Descripción			
Como usuario puedo iniciar sesión con correo y contraseña para acceder a mi cuenta.			
Funcionalidad			
Autenticación segura mediante JWT con validación de credenciales.			
Criterios de aceptación	1. Se debe generar un token JWT tras autenticación exitosa. 2. Se debe mostrar error por credenciales incorrectas.		
Restricciones			
El token durará una hora			

3.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	RF-03	Actor	Administrador
NOMBRE DEL REQUERIMIENTO	Gestión de categorías		
Descripción			

Como administrador puedo asignarle categorias a las peliculas.	
Funcionalidad	
Herramientas administrativas para gestión de categorías de contenido.	
Criterios de aceptación	1. Debe existir mínimo 4 categorías predefinidas. 2. Solo administradores pueden gestionar categorías. 3. Los nombres de categorías deben ser únicos.
Restricciones	
Solo el admin puede gestionar las categorias	

4.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	RF-04	Actor	Administrador
NOMBRE DEL REQUERIMIENTO	CRUD de películas		
Descripción			
Como administrador puedo gestionar el catálogo de películas, series y animes.			
Funcionalidad			
Gestión completa del catálogo con validaciones de integridad.			
Criterios de aceptación	1. Todos los campos obligatorios deben completarse. 2. La categoría debe existir en el sistema. 3. El año debe ser un valor numérico válido.		
Restricciones			
Solo administradores pueden realizar estas operaciones.			

5.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL	RF-05	Actor	Sistema

REQUERIMIENTO:			
NOMBRE DEL REQUERIMIENTO	Validación de títulos únicos		
Descripción			
Como sistema debo validar que no existan títulos duplicados para mantener integridad del catálogo.			
Funcionalidad			
Validación automática de unicidad de títulos.			
Criterios de aceptación	1. La validación debe ser case-insensitive. 2. Se debe considerar año para títulos similares. 3. La validación debe realizarse en tiempo real.		
Restricciones			
No se permiten títulos idénticos independientemente del tipo de contenido.			

6.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	RF-06	Actor	Usuario
NOMBRE DEL REQUERIMIENTO	Creación de reseñas		
Descripción			
Como usuario puedo crear reseñas con título, comentario y calificación del 1 al 10.			
Funcionalidad			
Sistema de reseñas con calificación granular y contenido textual.			
Criterios de aceptación	1. La calificación debe estar entre 1.0 y 10 con decimales. 2. El título es obligatorio con máximo 100 caracteres. 3. Solo una reseña por usuario por contenido.		
Restricciones			
Los usuarios deben estar autenticados para crear reseñas.			

7.

HISTORIA DE USUARIO

Prioridad: Media			
CÓDIGO DEL REQUERIMIENTO:	RF-07	Actor	Usuario
NOMBRE DEL REQUERIMIENTO	Edición de reseñas propias		
Descripción			
Como usuario puedo editar o eliminar mis reseñas para corregir o actualizar información.			
Funcionalidad			
Gestión de reseñas propias con historial de cambios.			
Criterios de aceptación	1. Solo el autor puede editar sus reseñas. 2. Se debe mantener historial de cambios. 3. La eliminación requiere confirmación.		
Restricciones			
No se pueden editar reseñas después de 24 horas si tienen más de 10 likes.			

8.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	RF-08	Actor	Usuario
NOMBRE DEL REQUERIMIENTO	Sistema de likes y dislikes		
Descripción			
Como usuario puedo dar like o dislike a reseñas de otros usuarios pero no a las mías.			
Funcionalidad			
Sistema de votación social para reseñas.			
Criterios de aceptación	1. Solo un voto por reseña por usuario. 2. No se permite votar reseñas propias. 3. Se puede cambiar el voto de like a dislike.		
Restricciones			

Los usuarios deben estar autenticados para votar.

9.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	RF-09	Actor	Sistema
NOMBRE DEL REQUERIMIENTO	Cálculo de rankings ponderados		
Descripción			
Como sistema debo calcular rankings basados en calificaciones, likes y fechas de reseñas.			
Funcionalidad			
Algoritmo de ranking considerando múltiples factores.			
Criterios de aceptación	1. El algoritmo debe ponderar calificación, volumen y fecha. 2. Los likes/dislikes deben influir en el peso de reseñas. 3. El ranking debe actualizarse automáticamente.		
Restricciones			
El cálculo debe ejecutarse cada que se crea o actualiza una reseña.			

10.

HISTORIA DE USUARIO			
Prioridad: Media			
CÓDIGO DEL REQUERIMIENTO:	RF-10	Actor	Usuario
NOMBRE DEL REQUERIMIENTO	Listados ordenados		
Descripción			
Como usuario puedo ver listados ordenados por popularidad y ranking.			
Funcionalidad			

Múltiples vistas de contenido con diferentes ordenamientos.	
Criterios de aceptación	1. Debe ofrecer ordenamiento por ranking y popularidad. 2. Los listados deben implementar paginación. 3. Se debe mostrar información resumida.
Restricciones	
Máximo 50 y 30 elementos por página.	

11.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	RF-11	Actor	Usuario
NOMBRE DEL REQUERIMIENTO	Vista detallada de contenido		
Descripción			
Como usuario puedo acceder a vista detallada con toda la información y reseñas.			
Funcionalidad			
Página de detalles completa del contenido y reseñas.			
Criterios de aceptación	1. Debe mostrar toda la información del contenido. 2. Las reseñas deben ordenarse por relevancia. 3. Se debe mostrar calificación promedio.		
Restricciones			
Solo contenido aprobado tiene página de detalles pública.			

12.

HISTORIA DE USUARIO			
Prioridad: Media			
CÓDIGO DEL REQUERIMIENTO:	RF012	Actor	Usuario
NOMBRE DEL REQUERIMIENTO	Búsqueda de contenido		

Descripción	
Como usuario puedo buscar películas, series y animes por título y otros criterios para encontrar contenido específico.	
Funcionalidad	
Motor de búsqueda con múltiples criterios y filtros avanzados.	
Criterios de aceptación	1. Debe permitir búsqueda por título con autocompletado. 2. Los resultados deben mostrarse ordenados por relevancia. 3. Se debe incluir filtros por categoría, año y calificación.
Restricciones	
Solo se muestran resultados de contenido aprobado.	
El historial solo es visible para administradores y autor de la reseña.	
Criterios de aceptación	1. Se debe poder marcar reseñas como inapropiadas. 2. Se debe notificar al usuario sobre moderación. 3. Se debe mantener registro de acciones de moderación.
Restricciones	
Solo administradores pueden moderar contenido.	

13.

HISTORIA DE USUARIO			
Prioridad: Media			
CÓDIGO DEL REQUERIMIENTO:	RNF-01	Actor	Sistema
NOMBRE DEL REQUERIMIENTO	Seguridad de contraseñas		
Descripción			
Como sistema debo usar bcrypt para hashear contraseñas antes de almacenarlas.			
Funcionalidad			
Implementación de seguridad en almacenamiento de credenciales.			
Criterios de aceptación	1. Las contraseñas deben hashearse con bcrypt.		

Control de velocidad de peticiones por IP y usuario.	
Criterios de aceptación	1. Se debe limitar a 100 peticiones por minuto por IP. 2. Se debe implementar bloqueo temporal tras intentos fallidos. 3. Se debe excluir peticiones estáticas del rate limiting.
Restricciones	
Las APIs críticas deben tener límites más restrictivos.	

16.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	RNF-04	Actor	Sistema
NOMBRE DEL REQUERIMIENTO	Base de datos MongoDB		
Descripción			
Como sistema debo utilizar MongoDB con driver oficial para persistencia de datos.			
Funcionalidad			
Implementación de base de datos NoSQL con driver nativo.			
Criterios de aceptación	1. Se debe usar driver oficial de MongoDB para Node.js. 2. Se debe implementar conexión con pool de conexiones. 3. Se debe manejar errores de conexión adecuadamente.		
Restricciones			
No se debe usar Mongoose u otros ODMs.			

17.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	RNF-05	Actor	Sistema
NOMBRE DEL REQUERIMIENTO	Transacciones MongoDB		

Descripción	
Como sistema debo implementar transacciones para operaciones críticas de reseñas y likes.	
Funcionalidad	
Transacciones atómicas para mantener consistencia de datos.	
Criterios de aceptación	1. Las operaciones de reseñas deben ser transaccionales. 2. Se debe hacer rollback automático en caso de error. 3. Las transacciones deben completarse o fallar completamente.
Restricciones	
Solo operaciones críticas deben usar transacciones por rendimiento.	

18.

HISTORIA DE USUARIO			
Prioridad: Media			
CÓDIGO DEL REQUERIMIENTO:	RNF-06	Actor	Sistema
NOMBRE DEL REQUERIMIENTO	Documentación API con Swagger		
Descripción			
Como sistema debo documentar todos los endpoints usando swagger-ui-express.			
Funcionalidad			
Documentación automática e interactiva de la API.			
Criterios de aceptación	1. Todos los endpoints deben estar documentados. 2. Se debe incluir ejemplos de peticiones y respuestas. 3. La documentación debe ser accesible vía web.		
Restricciones			
La documentación debe actualizarse automáticamente con cambios de código.			

19.

HISTORIA DE USUARIO

Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	RNF-07	Actor	Sistema
NOMBRE DEL REQUERIMIENTO	Versionado de API		
Descripción			
Como sistema debo versionar la API siguiendo convenciones de Semantic Versioning.			
Funcionalidad			
Control de versiones para mantener compatibilidad y evolución de API.			
Criterios de aceptación	1. Se debe implementar versionado en URLs (/api/v1/). 2. Se debe mantener retrocompatibilidad entre versiones menores. 3. Se debe documentar cambios breaking en changelog.		
Restricciones			
Los cambios breaking solo pueden introducirse en versiones mayores.			

20.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	RNF-08	Actor	Sistema
NOMBRE DEL REQUERIMIENTO	Variables de entorno		
Descripción			
Como sistema debo usar variables de entorno para toda configuración sensible.			
Funcionalidad			
Gestión segura de configuración y secretos.			
Criterios de aceptación	1. Se debe usar dotenv para cargar variables. 2. Todas las credenciales deben estar en .env. 3. Se debe proveer .env.example con variables necesarias.		
Restricciones			

El archivo .env no debe versionarse en Git.

21.

HISTORIA DE USUARIO			
Prioridad: Alta			
CÓDIGO DEL REQUERIMIENTO:	RNF-09	Actor	Usuario
NOMBRE DEL REQUERIMIENTO	Diseño responsive		
Descripción			
Como usuario puedo acceder a la plataforma desde dispositivos móviles y desktop correctamente.			
Funcionalidad			
Interfaz adaptable a diferentes tamaños de pantalla.			
Criterios de aceptación	1. Se debe adaptar a pantallas de 320px a 1920px. 2. Los elementos deben ser táctiles en dispositivos móviles. 3. Se debe mantener usabilidad en todas las resoluciones.		
Restricciones			
El diseño debe ser mobile-first y progresivamente mejorado.			

5. METODOLOGÍA

Para el presente proyecto se va a utilizar el marco de trabajo SCRUM que implica el esfuerzo de colaboración para crear un producto nuevo, este se va a utilizar en conjunto con la metodología ágil Kanban ya que es la que se ajusta de una mejor manera a los requerimientos del proyecto permitiendo una mayor efectividad al realizar el software solicitado por el cliente.

El equipo de trabajo está seccionado de la siguiente manera.

Scrum Master: David Santiago Vera Mendez

Product Owner: Harley Jeffrey Cabrales Vargas

Equipo Scrum: Harley Jeffrey Cabrales Vargas, David Santiago Vera Mendez

6. EVIDENCIA DE PLANTEAMIENTO DE PLATAFORMA DE TRABAJO

Acá se debe documentar toda la evidencia de trabajo colaborativo con los siguientes elementos:

- Link del repositorio donde se evidencia el trabajo colaborativo con el correcto uso de roles, ramas y conventional commit.
- Evidencia (capturas) del tablero Scrum utilizado en todas las etapas donde se visualicen todas las tareas, requerimientos e historias de usuario documentadas allí con responsables, tiempos, priorización, y demás requerimientos que se indiquen para las tareas.

POR HACER 3	EN CURSO 2	EN REVISIÓN 1	FINALIZADO 34 ✓
Endpoints CRUD de Categorías KAN-32 = HV	Endpoint (POST/PUT) Like/Dislike de Reseña KAN-36 = HV	Diseño Responsive General KAN-17 = DM	hacer el front de registro KAN-2 ✓ = DM
Página de Gestión de Categorías (Admin) KAN-15 = DM	Sistema de Like/Dislike de Reseñas KAN-13 = DM	+ Crear	inicio de sesion KAN-3 ✓ = DM
Diseño del Esquema de Categoría KAN-31 = HV			acceso al sistema KAN-4 ✓ = DM
+ Crear			funcionalida de administrador KAN-5 ✓ = DM
			funcionalidad de usuario KAN-6 ✓ = DM
			vista de listados de contenido KAN-7 ✓ = DM
			Componente de Búsqueda por Título KAN-8 ✓ = DM
			Vista Detallada de Contenido KAN-9 ✓ = DM
			Sección de Reseñas en Detalle KAN-10 ✓ = DM
			Formulario de Creación de Reseñas KAN-11 ✓ = DM
			Componentes de Edición/Eliminación de Reseña KAN-12 ✓ = DM
			Página de Gestión de Contenido (Admin) KAN-14 ✓ = DM
			Formularios de