

ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE (SRS)

UBISOFT RELEASE TRACKER

SEGÚN ESTÁNDAR IEEE 830

Documento: Requerimientos De la Web

Versión: 1.0

Fecha: 20 de Agosto, 2025

Tabla de Contenidos

1. Introducción
 - 1.1 Propósito
 - 1.2 Alcance
 - 1.3 Definiciones, Acrónimos y Abreviaturas
 - 1.4 Referencias
 - 1.5 Visión General del Documento
2. Descripción General
 - 2.1 Perspectiva del Producto
 - 2.2 Funciones del Producto
 - 2.3 Características de los Usuarios
 - 2.4 Restricciones
 - 2.5 Supuestos y Dependencias
3. Requisitos Específicos
 - 3.1 Requisitos de Interfaces Externas
 - 3.2 Requisitos Funcionales
 - 3.3 Requisitos de Rendimiento
 - 3.4 Restricciones de Diseño
 - 3.5 Atributos de Calidad del Software
 - 3.6 Otros Requisitos

1. INTRODUCCIÓN

1.1 Propósito

En el presente documento se especifican los requisitos de software para el sistema **Ubisoft Release Tracker**, una aplicación web diseñada para proporcionar información actualizada sobre próximos lanzamientos de videojuegos del desarrollador Ubisoft, incluyendo contadores de tiempo en tiempo real hasta las fechas de lanzamiento.

Este documento está dirigido a:

- Desarrolladores del sistema
- Testers del sistema
- Stakeholders del proyecto
- Usuarios técnicos y no técnicos

1.2 Alcance

Ubisoft Release Tracker es una aplicación web frontend que permite a los usuarios:

- Visualizar próximos lanzamientos de juegos de Ubisoft
- Monitorear contadores de tiempo en tiempo real hasta fechas de lanzamiento
- Filtrar y buscar juegos según múltiples criterios
- Acceder a la aplicación desde dispositivos móviles y de escritorio

Beneficios esperados:

- Centralización de información de lanzamientos de Ubisoft
- Experiencia de usuario mejorada para planificación de compras
- Accesibilidad universal para todos los tipos de usuarios
- Interfaz responsive para múltiples dispositivos

1.3 Definiciones, Acrónimos y Abreviaturas

Término	Definición
API	Application Programming Interface - Interfaz de programación de aplicaciones
ARIA	Accessible Rich Internet Applications - Especificación para accesibilidad web
CSS	Cascading Style Sheets - Hojas de estilo en cascada
DOM	Document Object Model - Modelo de objetos del documento
HTML	HyperText Markup Language - Lenguaje de marcado de hipertexto
HTTP	HyperText Transfer Protocol - Protocolo de transferencia de hipertexto
HTTPS	HTTP Secure - HTTP seguro
JavaScript	Lenguaje de programación para desarrollo web
JSON	JavaScript Object Notation - Notación de objetos JavaScript
PWA	Progressive Web Application - Aplicación web progresiva
RAWG	Servicio API para información de videojuegos
REST	Representational State Transfer - Transferencia de estado representacional
SPA	Single Page Application - Aplicación de página única
SRS	Software Requirements Specification - Especificación de requisitos de software
UI	User Interface - Interfaz de usuario
URT	Ubisoft Release Tracker - Nombre del sistema
UX	User Experience - Experiencia de usuario
WCAG	Web Content Accessibility Guidelines - Pautas de accesibilidad del contenido web

1.4 Referencias

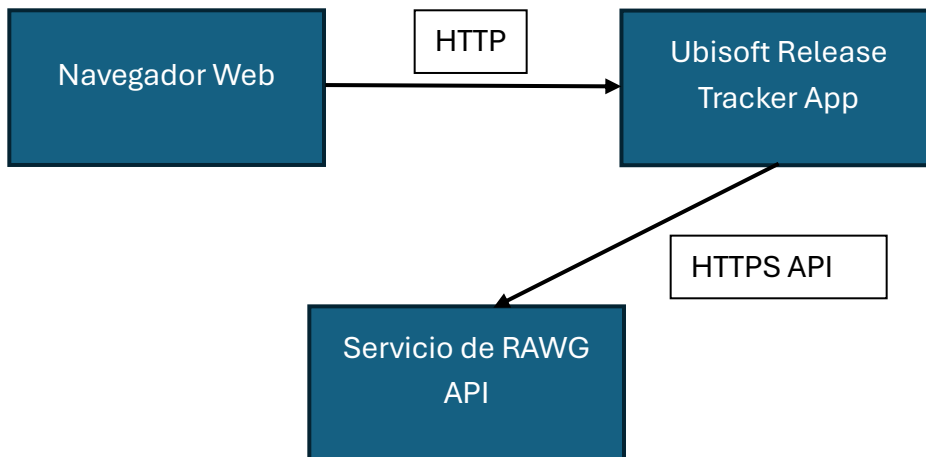
- IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications
- Web Content Accessibility Guidelines (WCAG) 2.1 Level AA
- RAWG Video Games Database API Documentation
- MDN Web Docs - Web APIs and Standards
- W3C HTML5 Specification
- W3C CSS3 Specification
- ECMAScript 2023 Language Specification

2. DESCRIPCIÓN GENERAL

2.1 Perspectiva del Producto

URT es una aplicación web independiente que funciona como cliente frontend consumiendo servicios de la API RAWG. No requiere instalación y es accesible desde cualquier navegador web moderno.

2.1.1 Interfaces del Sistema



2.1.2 Interfaces de Usuario

- **Interfaz web responsiva:** adaptable a dispositivos móviles y de escritorio
- **Navegación por teclado:** La web es navegable desde el teclado de cualquier dispositivo

- **Compatibilidad con lectores de pantalla:** Esta función tiene el objetivo de que personas con discapacidad visual, pueda aprovechar también las funciones de la web a través de lectores de pantalla

2.1.3 Interfaces de Hardware

- **Dispositivos móviles:** Smartphones y tablets con navegadores web
- **Dispositivos de escritorio:** Computadoras con navegadores modernos
- **Dispositivos asistidos:** Compatibilidad con lectores de pantalla, navegación por teclado

2.1.4 Interfaces de Software

- **Navegadores soportados:** Chrome 90+, Firefox 88+, Safari 14+, Edge 90+
- **API externa:** RAWG Games Database API v1.0

2.2 Funciones del Producto

2.2.1 Funciones Principales

F1. Visualización de Lanzamientos

- Mostrar lista de próximos juegos de Ubisoft
- Presentar información en formato de tarjetas
- Ordenar por fecha de lanzamiento

F2. Contadores de Tiempo Real

- Calcular tiempo restante hasta lanzamiento
- Actualizar automáticamente cada segundo
- Mostrar formato días/horas/minutos/segundos

F3. Sistema de Filtrado

- Filtrar por plataforma (PC, PlayStation, Xbox, Nintendo)
- Filtrar por género de juego
- Filtrar por rango temporal

F4. Búsqueda de Contenido

- Buscar juegos por título
- Búsqueda insensible a mayúsculas/minúsculas

F5. Carga Progresiva

- Implementar paginación de resultados
- Botón "Cargar más" para contenido adicional
- Mantener filtros durante carga progresiva

2.2.2 Funciones Secundarias

F6. Gestión de Estado

- Mostrar indicadores de carga
- Manejar estados de error

F7. Configuración de Usuario

- Ajustar días de búsqueda hacia adelante

2.3 Características de los Usuarios

2.3.1 Gamer Casual

- **Descripción:** Usuario ocasional que juega en tiempo libre
- **Nivel técnico:** Básico a intermedio
- **Frecuencia de uso:** Semanal
- **Dispositivos:** Principalmente móvil
- **Necesidades:** Información rápida y clara sobre lanzamientos

2.3.2 Entusiasta del Gaming

- **Descripción:** Jugador dedicado, fanático de Ubisoft
- **Nivel técnico:** Intermedio a avanzado
- **Frecuencia de uso:** Diaria
- **Dispositivos:** PC y móvil
- **Necesidades:** Información detallada, filtros avanzados

2.3.3 Creador de contenido

- **Descripción:** Creador de contenido sobre videojuegos
- **Nivel técnico:** Intermedio a avanzado
- **Frecuencia de uso:** Diaria

- **Dispositivos:** PC principalmente
- **Necesidades:** Información completa, recordatorios, planificación

2.3.4 Usuario con Discapacidad Visual

- **Descripción:** Usuario que utiliza tecnologías asistivas
- **Nivel técnico:** Variable
- **Frecuencia de uso:** Variable
- **Dispositivos:** PC con lector de pantalla
- **Necesidades:** Accesibilidad completa, navegación por teclado

2.4 Restricciones

2.4.1 Restricciones Regulatorias

- **C1.** Cumplimiento con términos de uso de RAWG API

2.4.2 Restricciones de Hardware

- **C2.** Funcionalidad en dispositivos con mínimo 1GB RAM
- **C3.** Compatibilidad con pantallas desde 320px de ancho
- **C4.** Funcionamiento en conexiones de internet lentas (3G+)

2.4.3 Restricciones de Software

- **C5.** Implementación usando solo tecnologías frontend (HTML5, CSS3, JavaScript)
- **C5.** No requiere instalación de plugins o software adicional
- **C7.** Compatibilidad con navegadores modernos (últimas 2 versiones principales)

2.4.4 Restricciones de Interfaz

- **C8.** Dependencia de disponibilidad de RAWG API
- **C9.** Límites de tasa de la API RAWG (requests por minuto)

2.5 Supuestos y Dependencias

2.5.1 Supuestos

- **A1.** Los usuarios tienen acceso a internet estable
- **A2.** La API RAWG mantendrá su formato actual de respuesta

- **A3.** Los navegadores soportarán estándares web modernos
- **A4.** Los datos de juegos de Ubisoft estarán disponibles en RAWG

2.5.2 Dependencias

- **D1.** Disponibilidad y confiabilidad de RAWG API
- **D2.** Mantenimiento de clave API válida para RAWG

3. REQUERIMIENTOS.

3.1. Requerimientos Funcionales

Requerimiento	
ID	RF1
Descripción	El sistema debe mostrar una cuadrícula con los próximos juegos, cada uno en una tarjeta que incluya imagen, título, fecha, plataformas y géneros.
Prioridad	Alta
Criterios de aceptación	Al cargar la página se muestran al menos 20 tarjetas; cada tarjeta contiene imagen, título y fecha; las tarjetas están ordenadas por fecha (más próximas primero).
Mapeo a HU	HU-001

Campo	
ID	RF2
Descripción	El sistema debe mostrar en cada tarjeta una cuenta regresiva en tiempo real hasta la fecha de lanzamiento.
Prioridad	Alta
Criterios de aceptación	El contador muestra días/hh/mm/ss y se actualiza cada segundo; al alcanzar la fecha muestra “Disponible” y deja de decrementar.

Campo	
Mapeo a HU	HU-002

Campo	
ID	RF3
Descripción	El sistema debe ser capaz de filtrar los juegos por plataforma, género y rango temporal seleccionado por el usuario.
Prioridad	Media
Criterios de aceptación	Los filtros se aplican inmediatamente en el cliente; se pueden combinar múltiples filtros; los filtros se mantienen al cargar más resultados.
Mapeo a HU	HU-003, HU-005

Campo	
ID	RF4
Descripción	El sistema debe ser capaz de buscar juegos por título con retroalimentación en tiempo real mientras el usuario escribe.
Prioridad	Media
Criterios de aceptación	Búsqueda insensible a mayúsculas; filtrado en vivo con debounce (~300 ms); resultados visibles sin recargar la página.
Mapeo a HU	HU-004

Campo	Valor
ID	RF5
Descripción	El sistema debe ofrecer un botón "Cargar más" para obtener la siguiente página de resultados e incorporar los nuevos juegos sin reemplazar los existentes.
Prioridad	Media
Criterios de aceptación	El botón aparece solo si existen más páginas; evita duplicados por ID; incrementa la página actual y añade nuevos elementos al grid.
Mapeo a HU	HU-011

Campo	
ID	RF6
Descripción	El sistema debe permitir al usuario establecer recordatorios locales (prototipo) mediante un botón "Recordarme" en cada tarjeta.
Prioridad	Baja
Criterios de aceptación	Al pulsar "Recordarme" aparece confirmación y el recordatorio queda persistido en localStorage; funciona sin backend.
Mapeo a HU	HU-007

Campo	
ID	RF7
Descripción	El sistema debe mostrar mensajes de estado claros (cargando, error, vacío) y proporcionar opciones de reintento para errores recuperables.
Prioridad	Alta
Criterios de aceptación	Diferencia errores de red/CORS, 401/403 (autenticación) y 429 (rate limit); muestra mensaje adecuado y botón "Reintentar" cuando corresponda.
Mapeo a HU	HU-012

Campo	
ID	RF8
Descripción	El sistema debe ser accesible: usar roles ARIA apropiados, aria-live para actualizaciones y permitir navegación por teclado.
Prioridad	Alta
Criterios de aceptación	aria-live presente en mensajes de estado; role="list" y role="listitem" en contenedores; tabulación funcional y foco visible; skip link operativo.
Mapeo a HU	HU-008, HU-009

Campo	
ID	RF9
Descripción	El sistema debe guardar en caché local la respuesta de la primera página para mejorar tiempos de carga en accesos posteriores.
Prioridad	Media
Criterios de aceptación	Cache con TTL (p. ej. 4 horas); si existe cache válida, se muestra inmediatamente en la carga inicial; opción de refrescar fuerza recarga desde la API.
Mapeo a HU	—

Campo	
ID	RF10
Descripción	El sistema debe permitir configurar el rango temporal (días hacia adelante) mediante un campo numérico con validación.
Prioridad	Baja
Criterios de aceptación	El campo acepta valores en el rango 30–1095; las peticiones usan dates=desde,hasta calculadas a partir del valor; el valor persiste en la sesión.
Mapeo a HU	HU-013

3.2. Requerimientos No Funcionales.

Campo	
ID	RNF1
Descripción	El sistema debe ofrecer tiempos de carga inicial y render razonables en dispositivos móviles para una buena experiencia de usuario.
Categoría	Rendimiento
Métrica	LCP < 2.5s; FCP < 1.5s; TBT < 300ms
Verificación	Ejecutar Lighthouse en modo Mobile y comprobar las métricas objetivo.

Campo	
ID	RNF2
Descripción	El sistema debe manejar fallos de red mostrando mensajes informativos y permitiendo reintento sin recargar la página.
Categoría	Disponibilidad / Conectividad
Métrica / Umbral	Reintento manual inmediato; la UI no debe colapsar por un fallo de red.
Verificación	Simular offline en DevTools y comprobar comportamiento y botón de reintento.

Campo	
ID	RNF3
Descripción	El sistema debe cumplir criterios de accesibilidad WCAG 2.1 Nivel AA en contraste, navegación por teclado y soporte para lectores de pantalla.
Categoría	Accesibilidad
Métrica / Umbral	Contrast ratio $\geq 4.5:1$ para texto normal; navegación completa por teclado; aria-live presente.
Verificación	Ejecutar Lighthouse Accessibility y WAVE; pruebas con NVDA/VoiceOver.

Campo	
ID	RNF4
Descripción	El sistema debe proteger la RAWG API key en producción; la clave no debe exponerse en el frontend y debe usarse mediante proxy/función serverless.
Categoría	Seguridad / Privacidad
Métrica / Umbral	RAWG_KEY almacenada en servidor/env vars; ausencia de clave en repositorio público.
Verificación	Revisar repo por secretos; probar proxy serverless que inyecta la clave desde variables de entorno.

Campo	
ID	RNF5
Descripción	El sistema debe ser compatible con navegadores modernos desktop y mobile (Chrome, Firefox, Safari, Edge).
Categoría	Compatibilidad
Métrica / Umbral	Pruebas exitosas en al menos 2 navegadores representativos por familia.
Verificación	Pruebas manuales o en BrowserStack en los navegadores objetivo.

Campo	
ID	RNF6
Descripción	El sistema debe ser responsive y ofrecer una interfaz usable en dispositivos táctiles, con objetivos táctiles mínimos y tipografía legible.
Categoría	Usabilidad / UI
Métrica / Umbral	Objetivo: puntuación de usabilidad alta (>90 Lighthouse); botones ≥44px; texto legible sin zoom.
Verificación	Probar en dispositivos reales y Lighthouse; comprobar tamaños táctiles.

Campo	
ID	NFR-7
Descripción	El sistema debe ser fiable ante respuestas parciales y evitar estados inconsistentes o duplicados al paginar.
Categoría	Fiabilidad
Métrica / Umbral	Duplicados por ID = 0 tras múltiples cargas; no errores JS por respuestas parciales.
Verificación	Ejecutar test de carga: múltiples loadMore y verificar unicidad de elementos en DOM.

Campo	
ID	RNF8
Descripción	El sistema debe escalar razonablemente en el cliente mediante paginación para manejar conjuntos de datos grandes sin degradación significativa.
Categoría	Escalabilidad
Métrica / Umbral	Mantener TTI razonable (ej. $\leq 5s$) con ~200 items en memoria.
Verificación	Pruebas de rendimiento local con dataset grande y medición de TTI/TTFB.

Campo	
ID	RNF9
Descripción	El sistema debe ser mantenible: código modular, documentado y fácil de trasladar parte de la lógica al backend.
Categoría	Mantenibilidad
Métrica / Umbral	Presencia de README y comentarios clave; estructura modular (HTML/CSS/JS separadas).
Verificación	Revisión de repositorio para comprobar documentación y estructura de archivos.

Campo	Valor
ID	RNF10
Descripción	El sistema debe cumplir con los términos de uso de RAWG y respetar límites de rate; debe mostrar atribución si RAWG lo exige.
Categoría	Legal / Políticas
Métrica / Umbral	No exceder cuotas; adherencia al TOS de RAWG.
Verificación	Revisar TOS RAWG y logs de peticiones; añadir atribución visible si es requerida.