
Especificación de requisitos de software

Proyecto: KNIGHTS AND DEMONS (2025)

Revisión [1.1]

Ficha del documento

Fecha	Revisión	Autor	Verificado dep. calidad.
25/08/2025	1.1	Miranda, Guillermo Damián	Departamento de ingeniería e investigaciones Tecnológicas

Documento validado por las partes en fecha: [\[Fecha\]](#)

Por el cliente	Por la empresa suministradora
Fdo. D./ Dña [Nombre]	Fdo. D./Dña [Nombre]

Contenido

FICHA DEL DOCUMENTO	3
CONTENIDO	4
1	INTRODUCCIÓN 6
1.1	Propósito 4
1.2	Alcance 4
1.3	Personal involucrado 4
1.4	Definiciones, acrónimos y abreviaturas 5
1.5	Referencias 5
1.6	Resumen 6
2	DESCRIPCIÓN GENERAL 6
2.1	Perspectiva del producto 6
2.2	Funcionalidad del producto 6
2.3	Características de los usuarios 6
2.4	Restricciones 7
2.5	Suposiciones y dependencias 7
2.6	Evolución previsible del sistema 7
3	REQUISITOS ESPECÍFICOS 8
3.1	Requisitos funcionales 8
3.1.1	Configuración Inicial 10
3.1.2	Mecánica de Juego 10
3.1.3	Extensiones 10
3.2	Requisitos no funcionales 10
3.2.1	Requisitos de Rendimiento 11
3.2.2	Requisitos de Fiabilidad 11
3.2.3	Requisitos de Portabilidad 12
3.2.4	Requisitos de Usabilidad 12
4	APÉNDICES 12

1 Introducción

El presente documento tiene como propósito especificar los requisitos del software para el desarrollo de un juego en lenguaje ANSI C, inspirado en la mecánica del juego Knights & Demons (Kabuto Factory, 2012).

1.1 Propósito

El documento servirá como guía para el equipo de desarrollo durante la implementación, validación y verificación del producto, así como para el docente responsable de la asignatura Tópicos de Programación en el 2° cuatrimestre del año 2025.

1.2 Alcance

El producto a desarrollar es un juego de tablero basado en una matriz interactiva de una dimensión fija, utilizando una aplicación de consola.

El jugador podrá seleccionar casillas distintas, sin un límite de selección, las cuales, al ser activadas, modificarán su estado y el de sus casillas adyacentes (en forma de cruz, es decir la casilla inmediatamente arriba de la misma, a su derecha, a su izquierda y debajo de la misma, si correspondiese). El objetivo principal será unificar el tablero bajo un mismo estado antes de que se agote un determinado tiempo disponible, variable dependiendo del nivel de dificultad seleccionado.

Adicionalmente el juego deberá incluir distintos el uso de objetos llamados “pikas” para ejercer el cambio de estado de una casilla sin afectar a las demás y un medidor de estado de la partida, donde el jugador pueda reconocer el tiempo restante para completar el nivel.

1.3 Personal involucrado

Nombre	Darío Hirschfeldt
Rol	Supervisor del Proyecto.
Categoría profesional	Docente cargo de práctica.
Responsabilidades	Supervisar y coordinar al equipo de trabajo. Comunicar eficazmente requisitos de producto. Vínculo vía equipo de desarrollo y empresa.
Información de contacto	N/A
Aprobación	Acreditada

Nombre	Pablo Soligo
Rol	Docente cargo de Teoría
Categoría profesional	Magister en Desarrollos informáticos De aplicación espacial
Responsabilidades	Proveer las habilidades y competencias necesarias para el equipo de trabajo.
Información de contacto	N/A
Aprobación	Acreditada

Nombre	Guillermo Damián Miranda
Rol	Desarrollador
Categoría profesional	IT Software Developer
Responsabilidades	Software Developer, QA Analyst, Software Architect
Información de contacto	N/A
Aprobación	Acreditada

Nombre	Agustina Losada Martínez
--------	--------------------------

Rol	Desarrolladora
Categoría profesional	IT Software Developer
Responsabilidades	Software Developer, QA Analyst, Software Architect
Información de contacto	N/A
Aprobación	Acreditada

Nombre	Jean Pierre Larán
Rol	Desarrollador
Categoría profesional	IT Software Developer
Responsabilidades	Software Developer, QA Analyst, Software Architect
Información de contacto	N/A
Aprobación	Acreditada

1.4 Definiciones, acrónimos y abreviaturas

ANSI C: Estándar de lenguaje de programación C utilizado en el desarrollo.

Pikas: Recurso especial dentro del juego que permite invertir una casilla sin afectar a las demás.

SDL2: Biblioteca opcional para la implementación de gráficos, audio y control de dispositivos de entrada.

ERS: Especificación de Requisitos de Software (IEEE 830).

Knights: Se trata de uno de los posibles estados para las casillas de la matriz. Representa a un caballero de la época medieval.

Demons: Se trata de uno de los posibles estados para las casillas de la matriz. Representa a un demonio de la época medieval.

Tablero: Se referirá a la matriz donde se colocan las casillas.

1.5 Referencias

Referencia	Título	Ruta	Fecha	Autor
Documentación de requisitos	Archivo de Trabajo Práctico de la asignatura	N/A	19/08/2025	Hirschfeldt, Darío
Video	Gameplay del juego asociado	https://www.youtube.com/watch?v	07/05/2019	YT: "Araubi"
Manual	Manual del juego	https://spa2.speccy.org/Instructions/k/KnightsDemosDX.pdf	12/09/2014	Paul Cornell
Información General	Información específica y alternativa del juego	https://spectrumcomputing.co.uk/entry/28054/ZX-Spectrum/Knights_Demons	2015	Spectrum Computing
Librerías	Librería estándar de SDL2	https://www.libsdl.org	1998	Sam Lantinga

Manual	"El lenguaje de programación C"	https://pergamo.unlam.edu.ar/pergamo/documento.php?ui=1&recno=4443&id=PERGAMO.1.4443	1978	Brian Kernighan
--------	---------------------------------	---	------	-----------------

1.6 Resumen

El documento está organizado en **tres secciones principales**:

1. La Introducción describe el propósito, alcance y contexto del sistema.
2. La Descripción general detalla las características globales del producto, los tipos de usuarios, restricciones y dependencias.
3. Los Requisitos específicos enumeran y definen los requisitos funcionales y no funcionales que el sistema deberá cumplir, incluyendo tanto los mínimos obligatorios como los opcionales para ampliar la funcionalidad.

2 Descripción general

2.1 Perspectiva del producto

El videojuego se desarrollará como una práctica académica de la materia Tópicos de Programación de la universidad Nacional de la Matanza en ANSI C.

El sistema será un programa independiente que se ejecutará en computadoras personales, sin necesidad de conexión a internet, pero sí de archivos locales para su funcionamiento, entre ellos, archivos binarios, una aplicación de consola y archivos de texto.

La dinámica del juego se basa en una matriz donde el jugador interactúa con casillas que modifican su estado y el de las casillas adyacentes. El producto será una adaptación del juego Knights & Demons de año 2013 de Paul Cornell, manteniendo la mecánica principal pero con el agregado de requisitos específicos y opcionales establecidos por la cátedra.

2.2 Funcionalidad del producto

El sistema deberá proporcionar las siguientes funciones principales:

1. El sistema debe permitir seleccionar casillas dentro de una matriz, modificando su estado y el de las casillas adyacentes en cruz.
2. El sistema debe ofrecer al menos tres niveles de dificultad con variaciones en el tiempo y recursos disponibles.
3. El sistema debe ofrecer al menos tres configuraciones de tamaño de tablero.
4. El sistema debe ofrecer una mecánica de "pikas" que permitan realizar jugadas especiales, con la posibilidad de obtener pikas adicionales al completar tableros a favor de los Knights.
5. El sistema debe incluir un temporizador que determine el tiempo disponible para completar cada tablero.
6. El sistema debe incluir un medidor de estado de la partida que indique qué bando (Knights o Demons) va ganando.
7. El sistema debe ser creado utilizando estructuras de programación obligatorias las cuales son:
 - Arrays
 - Cadenas
 - Structs
 - Memoria dinámica
8. Opcionalmente, el sistema puede proveer interfaz gráfica con SDL2, control mediante mouse/teclado, sistema de guardado/carga de partida y sonido.

2.3 Características de los usuarios

El producto está dirigido a los siguientes usuarios:

Tipo de usuario	Jugador Casual
Formación	N/A
Habilidades	N/A
Actividades	Fines Recreativos

Tipo de usuario	Docentes Universitarios
Formación	Docencia
Habilidades	Supervisión y corrección
Actividades	Revisión del cumplimiento del Proyecto con los requisitos

2.4 Restricciones

El sistema está regido por las siguientes restricciones:

- El sistema deberá desarrollarse en lenguaje ANSI C, con el material provisto por la cátedra.
- Se deberán aplicar tópicos de programación obligatorios.
- El proyecto comprimido no deberá superar los 40 MB de tamaño (Supuesto por Plataforma de envío de trabajo práctico)
- La entrega se realizará en formato .zip.
- Se deberán cumplir las fechas límite de entrega establecidas (21/10/2025 para versión mínima, 25/11/2025 para versión con SDL2 y archivos).
- Se deberá entregar además una carátula desarrollada con información de los integrantes del equipo
- Está sujeto a cambios a implementar en la fecha de la defensa Oral.

2.5 Suposiciones y dependencias

El sistema se rige por las siguientes suposiciones y dependencias que si cambian podrían afectar a los requisitos:

- La biblioteca SDL2, en caso de usarse, deberá estar correctamente instalada en el entorno de ejecución.
- Se asume que el usuario final tiene conocimientos básicos de interacción con teclado y/o mouse.
- Se asume que el sistema se ejecutará en sistemas operativos compatibles con compiladores de ANSI C

2.6 Evolución previsible del sistema

Las futuras mejoras que se podrían agregar a futuro en el sistema son las siguientes:

- En algún caso se podría implementar más mecánicas similares a las “pikas” ejemplificando un objeto coleccionable que permita eliminar un casillero por completo de la matriz, eliminando la posibilidad de cambiar de estado para el usuario, pero también siendo innecesario de contabilizar para completar la matriz.
- En el juego original, el final del mismo variaba si lograbas completar mas matrices de Knights que de Demons. Esto es una mecánica que también podría ser agregable, inclusive con la posibilidad de generar un último nivel dependiendo de las “elecciones” de los niveles anteriores.

3 Requisitos específicos

Los siguientes requisitos son específicos y se detallan del sistema a desarrollar:

3.1 Requisitos Funcionales

En este apartado se explican y desarrollan los requisitos de funcionalidades

Número de requisito	RF-01				
Nombre de requisito	Dificultad				
Tipo	X	Requisito		Restricción	
Fuente del requisito	Cliente				
Prioridad del requisito	x	Alta/Esencial		Media/Deseado	Baja/ Opcional

Número de requisito	RF-02				
Nombre de requisito	Tableros				
Tipo	X	Requisito		Restricción	
Fuente del requisito	Cliente				
Prioridad del requisito	x	Alta/Esencial		Media/Deseado	Baja/ Opcional

Número de requisito	RF-03					
Nombre de requisito	Intercambio de casillas					
Tipo	X	Requisito		Restricción		
Fuente del requisito	Cliente					
Prioridad del requisito	x	Alta/Esencial		Media/Deseado		Baja/ Opcional

Número de requisito	RF-04				
Nombre de requisito	Uso de Pikas				
Tipo	X	Requisito		Restricción	
Fuente del requisito	Cliente				
Prioridad del requisito	x	Alta/Esencial		Media/Deseado	Baja/ Opcional

Número de requisito	RF-05				
Nombre de requisito	Pikas adicionales				
Tipo	<input checked="" type="checkbox"/>	Requisito	<input type="checkbox"/>	Restricción	

Fuente del requisito	Cliente				
Prioridad del requisito	<input checked="" type="checkbox"/>	Alta/Esencial	<input type="checkbox"/>	Media/Deseado	<input type="checkbox"/> Baja/ Opcional

Número de requisito	RF-06				
Nombre de requisito	Temporizador				
Tipo	X	Requisito		Restricción	
Fuente del requisito	Cliente				
Prioridad del requisito	x	Alta/Esencial		Media/Deseado	Baja/ Opcional

Número de requisito	RF-07				
Nombre de requisito	Medidor visible				
Tipo	X	Requisito		Restricción	
Fuente del requisito	Cliente				
Prioridad del requisito	x	Alta/Esencial		Media/Deseado	Baja/ Opcional

Número de requisito	RF-08				
Nombre de requisito	Mecánica de guardado				
Tipo	X	Requisito		Restricción	
Fuente del requisito	Cliente				
Prioridad del requisito		Alta/Esencial		Media/Deseado	X Baja/ Opcional

Número de requisito	RF-09				
Nombre de requisito	Carga de partidas guardadas				
Tipo	X	Requisito		Restricción	
Fuente del requisito	Cliente				
Prioridad del requisito		Alta/Esencial		Media/Deseado	x Baja/ Opcional

Número de requisito	RF-10				
Nombre de requisito	Interfaz Gráfica				
Tipo	<input type="checkbox"/>	Requisito	<input checked="" type="checkbox"/>	Restricción	
Fuente del requisito	Cliente				
Prioridad del requisito	<input type="checkbox"/>	Alta/Esencial	<input type="checkbox"/>	Media/Deseado	<input checked="" type="checkbox"/> Baja/ Opcional

Número de requisito	RF-11				
Nombre de requisito	Control mediante teclado y mouse				
Tipo	<input type="checkbox"/>	Requisito	<input checked="" type="checkbox"/>	X	Restricción
Fuente del requisito	Cliente				
Prioridad del requisito	<input type="checkbox"/>	Alta/Esencial	<input type="checkbox"/>	Media/Deseado	<input checked="" type="checkbox"/> x Baja/ Opcional

Número de requisito	RF-12				
Nombre de requisito	Efectos de sonido y música				
Tipo	<input type="checkbox"/>	Requisito	<input checked="" type="checkbox"/>	X	Restricción
Fuente del requisito	Cliente				
Prioridad del requisito	<input type="checkbox"/>	Alta/Esencial	<input type="checkbox"/>	Media/Deseado	<input checked="" type="checkbox"/> x Baja/ Opcional

3.1.1 Configuración inicial

- RF-01: El sistema deberá permitir seleccionar al menos tres niveles de dificultad.
- RF-02: El sistema deberá permitir seleccionar al menos tres tamaños de tablero.

3.1.2 Mecánica del juego

- RF-03: El sistema deberá implementar la mecaninca principal de intercambio de casillas en cruz, es decir, cuando el jugador selecciona una casilla, esta y sus adyacentes (arriba, abajo, izquierda, derecha) deberán cambiar de estado.
- RF-04: El sistema deberá permitir el uso de pikas para modificar una sola casilla sin afectar a las adyacentes.
- RF-05: El sistema deberá otorgar pikas adicionales al completar un tablero a favor del bando de los caballeros.
- RF-06: El sistema deberá incluir un temporizador de limite el tiempo disponible para resolver cada tablero.
- RF-07: El sistema deberá mostrar un medidor de estado de la partida, indicando cuál de los bandos (caballeros o demonios) va ganando.

3.1.3 Extensiones

- RF-08: El sistema deberá permitir guardar la partida para retomarla posteriormente.
- RF-09: El sistema deberá permitir cargar una partida guardada y continuar desde el estado previo.
- RF-10: El sistema deberá contar con una interfaz gráfica implementada con la biblioteca SDL2.
- RF-11: El sistema deberá permitir el control mediante el teclado y/o mouse.
- RF-12: El sistema deberá reproducir efectos de sonido o música durante la partida.

3.2 Requisitos no funcionales

En este apartado se detallan los requisitos no funcionales del sistema:

3.2.1 Requisitos de rendimiento

Se detallan los requisitos que afectan al rendimiento del sistema:

RNF-01: El sistema deberá ejecutarse en computadoras personales

3.2.2 Requisitos de Fiabilidad

Se detallan los requisitos que afectan a la fiabilidad del sistema:

RNF-02: El sistema debe ser entregado con todas sus funcionalidades obligatorias en la fecha estipulada.

RNF-03: De ser necesario, una segunda entrega con funcionalidades adicionales se debe entregar en la fecha estipulada.

3.2.3 Requisitos de Portabilidad

RNF-04: El sistema deberá ser compilable en entornos compatibles con ANSI C

RNF-05: La instalación de la biblioteca SDL2 será responsabilidad del usuario.

3.2.4 Requisitos de Usabilidad

RNF-07: La interfaz del sistema deberá ser comprensible para usuarios sin experiencia técnica avanzada.

4 Apéndices

Se detallan algunas herramientas de desarrollo a usar:

- **CodeBlocks:** Un IDE entorno de desarrollo por el cuál se va a compilar y realizar entornos de prueba.
- **Github:** Utilizado para el control de versiones y facilitar la comunicación entre el equipo de trabajo.