



# DOCUMENTO FUNCIONAL DEL PROYECTO

ALUMNO/GRUPO: \_\_\_\_\_

---

## 1. Introducción y contexto

*Objetivo:* explicar de qué trata el proyecto y por qué es necesario.

Describe:

- El problema o necesidad que se quiere resolver.

La necesidad que se ha podido observar es la combinación de dos géneros de videojuegos que no se han complementado nunca.

- Quién será el usuario o cliente final.

El usuario o cliente final será cualquier tipo de persona que quiera disfrutar del videojuego brindado por parte del proyecto.

- Qué solución se propone y con qué propósito.

El proyecto propone la creación de un nuevo género de videojuegos nunca antes visto con el propósito de inspirar más videojuegos con este género.

*Ejemplo:*

Se desarrollará una aplicación web para gestionar las reservas del gimnasio del centro. Permitirá a los usuarios consultar y reservar espacios desde el móvil.

---

## 2. Análisis de requisitos

### 2.1. Requisitos funcionales (RF)

*Qué debe hacer el sistema.*

Enumera las funciones principales, numeradas como RF1, RF2, etc.

Código	Descripción del requisito funcional
RF1	La página web contendrá la descarga del videojuego.
RF2	Los usuarios podrán jugar al videojuego en el ordenador.
RF3	El videojuego se podrá ejecutar desde un .exe.

---

## 2.2. Requisitos no funcionales (RNF)

*Cómo debe comportarse el sistema.*

Incluye aspectos como rendimiento, seguridad, compatibilidad o facilidad de uso.

Código	Descripción del requisito no funcional
RNF1	La página web podrá registrar usuarios
RNF2	El videojuego tendrá en cuenta los records del usuario.

---

## 2.3. Restricciones

*Condiciones o limitaciones del proyecto.*

- Lenguajes o tecnologías obligatorias:  
Html, Css, C#, Wordpress, phpMyAdmin, Unity
  - Recursos disponibles (tiempo, equipo, materiales):  
6 meses, 2 personas, Ordenadores.
  - Dependencias o limitaciones técnicas:
- 

## 3. Análisis de usuarios y roles

*Objetivo:* identificar quién usará el sistema y qué podrá hacer.

Rol	Descripción	Permisos principales
Programador	Creador y jefe de las gestiones del proyecto	Acceso al código fuente del videojuego y a la página web.
Usuario	Acceso a la página web y a la descarga y ejecución del videojuego.	Instalación y uso del videojuego.
Visitante	Acceso a la página web.	Sin permisos.

## 4. Casos de uso / Escenarios de uso

*Objetivo:* mostrar cómo interactúan los usuarios con el sistema.

Selecciona de tres a cinco casos principales y descríbelos brevemente.

Código	Nombre del caso de uso	Actor principal	Descripción	Resultado esperado
CU1	Registrar usuario	Visitante	El usuario introduce sus datos y se crea una cuenta.	Usuario registrado correctamente.
CU2	Iniciar sesión	Usuario	Introduce credenciales y accede al sistema.	Acceso concedido.
CU3	Descargar el juego	Usuario	Se descarga el juego en su dispositivo.	Juega al juego en su ordenador de forma local.

## 5. Modelo de datos o estructura de la información

*Objetivo:* representar la información que gestionará el sistema.\*

Incluye las entidades principales (tablas u objetos) y las relaciones entre ellas.

Ejemplo:

- Entidades: [Usuario](#), [Descarga el juego](#), [Desarrollador](#).
- Relaciones: [Un usuario puede descargar varios archivos de instalación del juego, cada archivo corresponde a un solo juego.](#)

(Opcional: añadir diagrama ER o tabla resumen de campos.)

---

## 6. Diseño de la interfaz

*Objetivo:* visualizar la estructura y navegación del sistema antes de desarrollarlo.\*

Incluye bocetos o capturas de las pantallas principales y una breve descripción de su función.

Para cada pantalla, indica:

### **Pantallas principales:**

#### **1. Página de inicio**

**Función:** Presentar el proyecto y mostrar información principal.

**Casos relacionados:** Visualización general (sin CU concreto).

#### **2. Pantalla de registro**

**Función:** Permitir a un visitante crear una cuenta.

**Casos:** CU1.

#### **3. Pantalla de inicio de sesión**

**Función:** Acceso al perfil del usuario.

**Casos:** CU2.

#### **4. Página de descarga**

**Función:** Ofrecer el archivo .exe para descargar el videojuego.

**Casos:** CU3.

---

## 7. Planificación técnica

*Objetivo:* planificar el desarrollo del proyecto.\*

Indica las tecnologías y herramientas que se utilizarán, y cómo se organizará el trabajo.

### Lenguajes y frameworks:

- HTML, CSS para la web.
- Wordpress como gestor de contenido.
- C# y Unity para el videojuego.

### Base de datos:

- MySQL gestionado mediante phpMyAdmin.

### Herramientas de diseño o edición:

- Canva, Figma o Photoshop para diseño visual.
- Unity Editor para el videojuego.

### Reparto de tareas:

- **Persona 1:** Desarrollo del videojuego en Unity.
- **Persona 2:** Creación y gestión de la página web.

### Cronograma (resumen):

- Mes 1: Diseño del videojuego y la web.
  - Mes 2–3: Desarrollo del videojuego.
  - Mes 4: Desarrollo de la web.
  - Mes 5: Integración, pruebas y base de datos.
  - Mes 6: Revisión, optimización y presentación final.
-

## 8. Análisis de riesgos

*Objetivo:* identificar posibles problemas y cómo se afrontarán.\*

### 8.1. Identificación de riesgos

- Falta de tiempo o mala planificación.
- Problemas técnicos o incompatibilidades.
- Pérdida de datos por fallos en el equipo.
- Abandono o baja de un miembro del grupo.
- Cambios en los requisitos del proyecto.

### 8.2. Valoración y respuesta

Riesgo	Probabilidad	Impacto	Plan de prevención
Falta de tiempo	Alta	Alta	Dividir tareas y crear entregas intermedias.
Problemas técnicos	Media	Media	Probar tecnologías antes de programar.
Pérdida de datos	Baja	Alta	Realizar copias de seguridad semanales.
Abandono de un miembro	Baja	Alta	Documentar todo el proceso para facilitar continuidad.

### 8.2. Valoración y respuesta

Clasifica cada riesgo según su probabilidad e impacto, e indica cómo se mitigará.

Riesgo	Probabilidad	Impacto	Plan de prevención o contingencia
Falta de tiempo	Alta	Alta	Dividir tareas y fijar entregas intermedias.
Problemas técnicos	Media	Media	Probar tecnologías antes de programar.
Pérdida de datos	Baja	Alta	Hacer copias de seguridad semanales.

---

## 9. Validación y criterios de éxito

*Objetivo:* definir cómo sabremos que el proyecto funciona correctamente.\*

### **Criterios de aceptación:**

- El usuario puede registrarse, iniciar sesión y descargar el juego.
- El videojuego se ejecuta correctamente en formato .exe.
- La web funciona en distintos navegadores.
- Los récords del usuario se guardan correctamente.

### **Pruebas previstas:**

- Pruebas funcionales de todos los casos de uso.
- Pruebas de usuario para comprobar la buena experiencia.
- Pruebas de rendimiento del videojuego en distintos equipos.

### **Indicadores de calidad:**

- Tiempo de carga reducido.
  - Ausencia de errores críticos.
  - Usuarios capaces de descargar y ejecutar el juego sin problemas.
- 

## 10. Conclusión

*Objetivo:* cerrar el análisis y preparar la siguiente fase.\*

Resume las decisiones principales tomadas durante el análisis:

Este documento define de forma clara las funciones, tecnologías y limitaciones del proyecto. Se establece la creación de un videojuego innovador, acompañado de una página web funcional para su distribución.

El valor principal del proyecto es la creación de un nuevo género que combina mecánicas nunca antes unidas, aportando originalidad al ámbito del desarrollo de videojuegos.

**Próximos pasos:**

- Preparar el entorno de desarrollo.
- Crear la estructura del repositorio y la base de datos.
- Empezar la implementación de los casos de uso prioritarios.