|  |
| --- |

**Especificación de requisitos de software**

**Proyecto:**

**Video Juego – Cartas Mortal Kombat**

|  |  | 28/07/2025 |
| --- | --- | --- |

.

**Ficha del documento**

| **Fecha** | **Revisión** | **Autor** | **Verificado dep. Calidad.** |
| --- | --- | --- | --- |
|  |  |  |  |

*Tabla de contenido*

[Contenido](#_heading=h.1fob9te) 3

[1](#_heading=h.3znysh7) Introducción 4

[1.1Propósito](#_heading=h.2et92p0)

1.2 Alcance

1.3 Definiciones, acrónimos y abreviaturas

1.4 Referencias

1.5 Visión general.

[2 Descripción general 8](#_heading=h.z337ya)

[2.1 Perspectiva del producto 8](#_heading=h.1y810tw)

2.2 Funciones del producto [8](#_heading=h.2xcytpi)

2.3 Características de los usuarios [9](#_heading=h.3whwml4)

2.4 Restricciones

2.5 Suposiciones y dependencias

2.6 Requerimientos futuros [9](#_heading=h.qsh70q)

[3 Requisitos específicos 10](#_heading=h.5tqvktjg4afo)

[3.1 Requerimientos de interfaces externas: 10](#_heading=h.5tqvktjg4afo)

[3.2 Requerimientos Funcionales 10](#_heading=h.49x2ik5)

[3.3 Requerimientos No Funcionales](#_heading=h.2p2csry)

3.4 Restricciones de diseño

3.5 Atributos del sistema

3.6 Otros requisitos [34](#_heading=h.2p2csry)

4. Apéndices [41](#_heading=h.147n2zr)

[4.1 Diagrama UML de casos uso 41](#_heading=h.3o7alnk)

[4.2 Caracterización de casos de uso 42](#_heading=h.23ckvvd)

1. **Introducción**

***Abstract***

*Keywords:*

**Proyecto Tecnológico**, **Videojuegos**, **Bootcamp**

***Resumen***

El presente proyecto propone el desarrollo de un **videojuego** orientado a un estilo de juego tradicional por medio de cartas, para este caso, con la temática del reconocido videojuego **Mortal Kombat**. Esto se hará con el objetivo de culminar el Bootcamp en proceso.

El aplicativo incluirá ciertas funciones como elección de jugadores por partida, tiempos de respuesta y validación de jugada, distribución de cartas según resultado de competencia, entre otros.

* 1. **Planteamiento del problema**

A lo largo del Bootcamp se busca reforzar ciertas habilidades técnicas que no han sido claras o adoptadas por los aprendices y como prueba final se requiere llevar al cabo del desarrollo de un videojuego multijugador para demostrar que han adquirido las competencias faltantes.

* 1. **Alcance**

Se desea crear un juego de cartas denominado “Mortal Kombat” para Web, que será utilizado por usuarios con un interés en partidas de combate temporales donde tendrán que tomar en cuentas las decisiones y estrategias que debes seleccionar para poder ganar la partida. En el sistema tendrán personajes aleatorios, con su nivel de habilidad y poder, donde se permitirá combatir en un juego de cartas con demás personas..

* 1. **Propósito**

Este documento tiene como propósito dar a conocer el funcionamiento general del proyecto en desarrollo (Videojuego de Cartas con Temática de Mortal Kombat) que está dirigido a los instructores encargados del Bootcamp y los usuarios finales.

* 1. **Justificación**

El proyecto "Videojuego – Cartas Mortal Kombat" surge como la evidencia técnica para comprobar que el tiempo invertido en el Bootcamp se aprovechó de manera eficiente, logrando el objetivo inicial el cual proponía reforzar ciertas habilidades ausentes en los aprendices.

* 1. **Objetivo General**

Diseñar un juego de cartas con temática de Mortal Kombat en donde los jugadores juegan sus cartas por turno. Cada carta representa un combatiente de Mortal Kombat el cual va está ligado a ciertas habilidades generales diferenciando con el nivel de experticia en cada habilidad.

Inicialmente se iniciará con una elección de tipo de usuario para posteriormente dar inicio al juego. Se tendrá un máximo de siete jugadores y un mínimo de dos jugadores, a cada uno se le darán 8 cartas aleatoriamente seleccionadas que tendrá que ir jugando a lo largo de la partida. Finalmente se definirá como ganador al jugador que obtenga todas las cartas de su contrincante o el jugador que tenga la mayor cantidad de cartas en el tiempo límite.

* 1. **Objetivos específicos**
* Diseñar una juego que permita el registro de jugadores para competir de manera divertida.
* Documentar el desarrollo del proyecto, incluyendo manuales técnicos y guías de uso.
* Crear un sistema de base de datos estructurada que almacene información detallada de cada recurso.
* Desarrollar una interfaz accesible y fácil de usar para la realización del juego mediante la web.
* Garantizar el acceso al juego con cualquier persona conectada a internet

* 1. **Personal involucrado**

| **Nombre** | **Ingrid Yulissa Medina Esquivel** |
| --- | --- |
| **Rol** | **Líder** |
| **Categoría Profesional** | **Desarrollador** |
| **Responsabilidad** | **Desarrollador de Backend y Frontend** |
| **Información de contacto** | **3166054004** |
| **Nombre** | **Isabella Carrera Cabrera** |
| **Rol** | **Coordinador** |
| **Categoría Profesional** | **Desarrollador** |
| **Responsabilidad** | **Desarrollador de Frontend** |
| **Información de contacto** | **3202693215** |
| **Nombre** | **Rubén Felipe Tovar Avilés** |
| **Rol** | **Monitor** |
| **Categoría Profesional** | **Desarrollador** |
| **Responsabilidad** | **Desarrolladora de Backend y Frontend** |
| **Información de contacto** | **3138717041** |

* 1. **Definiciones, acrónimos y abreviaturas**

| ***Nombre*** | ***Descripción*** |
| --- | --- |
| **RF** | Requerimiento Funcional. |
| **RNF** | Requerimiento No Funcional. |

* 1. **Referencias**

| **Título del Documento** | **Referencia** |
| --- | --- |
| Como jugar | https://www.nhfournier.es/como-jugar/ |
| El mortal | https://bebeamordor.com/juego-de-mesa-el-mortal-atomo/ |
| Suposiciones | https://es.smartsheet.com/content/project-assumptions |
| Pokemon | https://www.pokemon.com/static-assets/content-assets/cms2-es-es/pdf/trading-card-game/rulebook/par\_rulebook\_es.pdf |

1. **Descripción general**

El aplicativo en cuestión estará orientado al combate entre diferentes personajes del videojuego Mortal Kombat, la representación de los personajes se realizará mediante cartas en donde cada una de estas tendrán un grupo de habilidades las cuales tendrán diferentes valores según el personaje establecido en la carta.

Se jugará con un máximo de 7 jugadores y un mínimo de dos jugadores por partida, a cada jugador le corresponderá un grupo de 8 cartas mezcladas aleatoriamente que a lo largo de la partida (10 minutos) deberá ir seleccionando para entrar en combate.

Finalmente ganará el jugador que acumule todas las cartas de las partidas y el que tenga la mayor cantidad a culminar el tiempo establecido

* 1. **Perspectiva del producto**

El producto final permitirá fomentar la competitividad y el pensamiento ágil mediante un ambiente divertido y sano. De la misma manera tiene como objetivo ser un aplicativo entregable como evidencia final del Bootcamp.

**Características Principales del Producto:**

1. **Competición**: Los jugadores podrán competir entre ellos para elegir su carta ganadora
2. **Acceso Fácil**: Desde cualquier computador conectado a internet se podrá acceder sin problemas implementando un registro simple

**Características de los usuarios**

| **Nombre de usuario** | ID |
| --- | --- |
| **Formación** | Creador |
| **Actividades** | Creación y Control de la Partida |

| **Nombre de usuario** | ID |
| --- | --- |
| **Formación** | Usuario |
| **Actividades** | Jugador de la Partida |

* 1. **Restricciones**
* El jugador debe seleccionar una carta por ronda
* Si se acaba el tiempo y no se ha tirado la carta perderá el turno
* Si se acaba la partida y hay una persona con un número mayor de cartas que los demás esa persona ganara
* Si la persona pierde sus cartas quedara eliminado

**Suposiciones:**

* Que todos los que participe sepan jugar cartas
* Que todos los jugadores sabrán de las reglas básicas del juego y podría realizar ciertas acciones
* Que los que sean participe tengan conocimiento previo del juego Mortal Kombat
* Que el juego atraerá a una cantidad de personas interesadas a juegos de cartas
* Que el jugador le atraiga la interfaz del juego

**Dependencias:**

* Depende de que un a persona gana, para empezar la otra partida
* Depende de un tiempo de respuesta por parte del jugador si o pierde
* Depende de haya una sala para poder realizar la partida
* Depende de que haya cartas para poder jugar.

1. **Requisitos específicos**

* Se implementará un apartado en donde el usuario (jugador) deberá elegir un como de acceso ya sea como creador de partida o se unirá a una como participante.
* Como creador de partida deberá elegir la cantidad de jugadores que se unirán a la partida y crear su nombre de usuario. De la misma manera, este se encargará de iniciar la partida.
* Como participante deberá crear su nombre de usuario y esperar el inicio de partida.
* Antes de empezar con la partida, el sistema mezclará las cartas y las repartirá de una en una a cada jugador.
* Una vez iniciada, el sistema elegirá la habilidad con la cual se debe combatir.
* Cada jugador en sus respectivos turnos deberá elegir su carta teniendo en cuenta la habilidad en juego y un tiempo límite de elección. En caso de que se agote el tiempo, el jugador en cuestión perderá su turno.
* Se irán lanzando la carta por turno sin ser reveladas, una vez el último jugador haya lanzado su carta, el sistema elegirá la carta ganadora y anunciará el ganador del encuentro.
* Para la finalización del juego, se elegirá ganador el jugador que consiga todas las cartas o que tenga la mayor cantidad de cartas dentro del tiempo de partida (10 min).
  1. **Requisitos comunes de las interfaces**
* **Requisitos Generales de Interfaz**
  + **Interfaz Responsive:** Compatible con computadores
  + **Diseño Intuitivo:** Uso de iconos, botones y colores que guíen al usuario sin necesidad de capacitación avanzada.
  + **Carga Rápida:** Optimización de imágenes y código para tiempos de respuesta ágiles.
  + **Accesibilidad:** Botones grandes, textos legibles y compatibilidad con lectores de pantalla.
* **Interfaz del Creador (Crear y Configura la Partida)**
  + **Pantalla Principal:**
    - Opción de Mezclar cartas y repartir
    - Iniciar Partida
* **Interfaz del Jugador**
  + **Pantalla Principal:**
    - Lista de sus cartas asignadas
    - Opción de lanzamiento de carta según turno asignado
    - Apilamiento de Cartas
    - Elección de ganador

Interfaces de hardware:

Es la interacción física con la computadora, por lo tanto, puede distinguirse entre la interfaz de hardware y la interfaz de software, son dispositivos utilizados para ingresar,procesar y entregar datos.

1. Hardware de procesamiento.

sistema, que digita,calcula y resuelve las operaciones necesarias para su funcionamiento como cpu, memoria RAM, y etc.

1. Hardware de entrada.

introduce datos al sistema por el usuario u operador, o proveniente de redes y sistemas de telecomunicaciones como el mouse, micrófonos, teclados y etc

1. Hardware mixto.

cumples las funciones de entrada y salida a la vez como la pantalla táctil,USB, disco duro

Interfaces de software:

destinada a entregar información acerca de los procesos y herramientas de control, a través de lo que el usuario observa habitualmente en la pantalla

1. software de sistema.

es el funcionamiento del sistema operativo se incorpora al sistema antes de que el usuario acceda a él como windows 10, APIs, bases de datos

Interfaces de comunicación:

1. las interfaces de comunicación Define protocolos de red o comunicación ejemplo Ip, FTP,

* 1. **Requisitos funcionales**

**Gestión de Sesión del Jugador**

| Identificador:  RF | | Nombre:  **Gestión de Sesión del Jugador** |
| --- | --- | --- |
| Tipo:  Requerimiento Funcional | Requerimiento que lo utiliza o especializa: | |
| Prioridad de desarrollo:  Alta | Documento de visualización asociado: | |
| Entrada:  Elección de tipo de Usuario (Creador, Participante) | Salida:  Funcionalidades según tipo de Usuario | |
| Descripción:  El sistema debe permitir al usuario elegir un rol de acceso (creador de partida o participante).  El sistema debe permitir al usuario crear un nombre de usuario. | | |
| Manejo de situaciones anormales:  En caso de no poder seleccionar una forma de sesión se mostrará una alerta | | |
| Criterios de aceptación:  El usuario deberá por de distinguir entre si es participante o creador | | |

**Creación de Partida (Rol Creador)**

| Identificador:  RF | | Nombre:  **Creación de Partida (Rol Creador)** |
| --- | --- | --- |
| Tipo:  Requerimiento funcional | Requerimiento que lo utiliza o especializa | |
| Prioridad de desarrollo  Alto | Documento de visualización asociado: | |
| Entrada:  Diferentes opciones de cantidad de sala | Salida:  Sala limitada a cierta cantidad de jugadores | |
| Descripción:   * El creador de partida debe poder seleccionar la cantidad de jugadores que se unirán (entre 2 y 7 jugadores, según la presentación) * El creador de partida debe poder iniciar la partida | | |
| Manejo de situaciones anormales:  En caso de no seleccionar una cantidad se dejará una cantidad preestablecida de dos jugadores | | |
| Criterios de aceptación:  Restringir el acceso una vez se ha llegado al límite | | |

**Unión a Partida (Rol Participante)**

| Identificador:  RF | | Nombre:  **Unión a Partida (Rol Participante)** |
| --- | --- | --- |
| Tipo:  Requerimiento Funcional | Requerimiento que lo utiliza o especializa | |
| Prioridad de desarrollo  Alto | Documento de visualización asociado: | |
| Entrada:  Código de invitación | Salida:  Unión a la partida | |
| Descripción:   * El participante debe poder unirse a una partida existente. * El participante debe esperar el inicio de la partida por parte del creador. | | |
| Manejo de situaciones anormales:  Se verificará cada código para restringir y validar los accesos | | |
| Criterios de aceptación:  Se validará cada código de acceso para un respectivo control | | |

**Preparación de la Partida**

| Identificador:  RF | | Nombre:  **Preparación de la Partida** |
| --- | --- | --- |
| Tipo:  Requerimiento Funcional | Requerimiento que lo utiliza o especializa | |
| Prioridad de desarrollo  Media | Documento de visualización asociado: | |
| Entrada:  Una baraja de 58 cartas | Salida:  Una baraja de 58 cartas mezcladas aleatoriamente | |
| Descripción:   * El sistema debe mezclar las cartas de los personajes de Mortal Kombat antes de repartirlas. * El sistema debe repartir 8 cartas a cada jugador al inicio de la partida | | |
| Manejo de situaciones anormales:  Se lanzará una alerta en caso de no poder realizar la mezcla | | |
| Criterios de aceptación:  Repartición de cartas completamente aleatoria | | |

**Desarrollo del Juego (Turnos)**

| Identificador:  RF | | Nombre:  **Desarrollo del Juego (Turnos)** |
| --- | --- | --- |
| Tipo:  Requerimiento Funcional | Requerimiento que lo utiliza o especializa | |
| Prioridad de desarrollo  Bajo | Documento de visualización asociado: | |
| Entrada:  Carta del Jugador | Salida:  Anuncio de jugador con la habilidad mas alta | |
| Descripción:   * El sistema debe elegir una habilidad (característica) con la que se debe combatir en cada ronda. * Cada jugador, en su turno, debe seleccionar una carta de su mano. * El sistema debe implementar un tiempo límite para la elección de la carta en cada turno. * Si el tiempo límite se agota, el jugador perderá su turno. * Las cartas lanzadas por los jugadores no deben ser reveladas hasta que todos los jugadores hayan lanzado su carta. * Una vez que el último jugador haya lanzado su carta, el sistema debe revelar todas las cartas jugadas en esa ronda. * El sistema debe determinar la carta ganadora de la ronda basándose en la habilidad en juego y comparar las características del modelo * El sistema debe anunciar al ganador de cada encuentro (ronda). | | |
| Manejo de situaciones anormales:  Se ejecutará una detención y reinicio de partida en caso de que falle algún proceso | | |
| Criterios de aceptación:  Una buena experiencia de juego | | |

**Finalización del Juego**

| Identificador:  RF | | Nombre:  **Finalización del Juego** |
| --- | --- | --- |
| Tipo:  Requerimiento Funcional | Requerimiento que lo utiliza o especializa | |
| Prioridad de desarrollo  Bajo | Documento de visualización asociado: | |
| Entrada:  Jugadores Posibles | Salida:  Jugador con la mayor cantidad de cartas | |
| Descripción:   * El juego debe finalizar cuando un jugador consiga todas las cartas. * El juego debe finalizar después de un tiempo límite de partida () * Si el tiempo de partida se agota, el ganador será el jugador con la mayor cantidad de cartas. * El sistema debe anunciar al ganador final del juego. | | |
| Manejo de situaciones anormales:  Se ejecutará una detención y reinicio de partida en caso de que falle algún proceso | | |
| Criterios de aceptación:  Elección de un Ganador | | |

**Gestión de Cartas**

| Identificador:  RF | | Nombre:  **Gestión de Cartas** |
| --- | --- | --- |
| Tipo:  Requerimiento Funcional | Requerimiento que lo utiliza o especializa | |
| Prioridad de desarrollo  Bajo | Documento de visualización asociado: | |
| Entrada: | Salida: | |
| Descripción:   * Las cartas deben tener plasmados diferentes modelos/personajes * Cada carta debe estará representada con un personaje de Mortal Kombat y los valores de sus habilidades generales * Cada carta debe contener características comparables | | |
| Manejo de situaciones anormales:  Prediseños cargados para asegurar la vista de las cartas | | |
| Criterios de aceptación:  Una vista legible y agradable de la presentación de la carta | | |

* 1. **Requisitos no funcionales**

| **Identificador:** 001 | | **Nombre:**  Interfaz accesible | |
| --- | --- | --- | --- |
| **Tipo:** Usabilidad | **Requerimiento que lo utiliza o especializa:** | | **¿Crítico?**  si |
| **Prioridad de desarrollo**  Alta | **Documento de visualización asociado:**  Diseño de interfaz de usuario | | |
| **Entrada:**  Interacción del usuario con la aplicación | **Salida:**  Experiencia de usuario fluida y sin complicaciones | | |
| **Descripción:**  La aplicación debe tener una interfaz sencilla y fácil de usar para una experiencia de juego dinamica | | | |
| **Manejo de situaciones anormales:**  En caso de errores, se mostrará sugerencias para corregirlos | | | |
| **Criterios de aceptación:**  El 90% de los usuarios deben poder jugar con facilidad | | | |

| **Identificador:** 002 | | **Nombre:** Respuesta rápida | |
| --- | --- | --- | --- |
| **Tipo:**  Rendimiento | **Requerimiento que lo utiliza o especializa:** | | **¿Crítico?**  si |
| **Prioridad de desarrollo**  **Alta** | **Documento de visualización asociado:**  Diagramas de flujo de procesos | | |
| **Entrada:**  Solicitudes de escaneo y consultas | **Salida:**  Respuestas en menos de 3 segundos | | |
| **Descripción:**  El sistema debe responder en menos de 3 segundos a las acciones de los jugadores | | | |
| **Manejo de situaciones anormales:**  Mensajes de Alerta durante las partidas | | | |
| **Criterios de aceptación:**  El tiempo de respuesta no debe superar los 3 segundos en el 95% de las solicitudes. | | | |

| **Identificador:** 003 | | **Nombre:** Documentación | |
| --- | --- | --- | --- |
| **Tipo:**  Mantenibilidad | **Requerimiento que lo utiliza o especializa:** | | **¿Crítico?**  No |
| **Prioridad de desarrollo**  Media | **Documento de visualización asociado:**  Manuales técnicos y de usuario | | |
| **Entrada:**  Cambios en la documentación | **Salida:**  Documentación Nueva | | |
| **Descripción:**  Proporcionar documentación técnica y de usuario | | | |
| **Manejo de situaciones anormales:**  No Aplica | | | |
| **Criterios de aceptación:**  No Aplica | | | |

**3.4 Restricciones de Diseño**

1.|No utilizar imágenes oficiales del juego base

2. Para el desarrollo de la UI se utilizara

2.1 Lenguajes

* TypeScript
* JS

2.2 Frameworks

* Angular
* Framework

2.3 Librerías

* SweetAlert
* Animation.css

3. El juego será compatible con (chrome,firefox,edge)

4. Sonido y música para dar un ambiente acogedor.

5. Para el prototipado se utilizará

* Figma
* PlanUML
* Draw.io