**S.I.G.P.D.**

Ingeniería en Software

Dreams Magic Wizard

| **Rol** | **Apellido** | **Nombre** | **C.I** | **Email** |
| --- | --- | --- | --- | --- |
| Coordinador | Espinosa | Jonathan | 4.563.848-6 | jonathan.espinosa@estudiante.ceibal.edu.uy |
| Sub-Coordinador | Estévez | Leandro | 4.438.718-9 | lelo.estevez24@gmail..com |
| Integrante 1 | Perdomo | Martin | 4.986.528-7 | martin.perdomo@estudiante. ceibal.edu.uy |
| Integrante 2 | Uval | German | 5.476.329-6 | valentin.uval.barboza@gmail. com |
| Integrante 3 | Kramarchuk | Robert | 4.828.299-3 | kramarchuk108@gmail.com |

**Docente: Aranda, Juan Andres**

# 

# **Índice**

[**Índice 1**](#_36my1mejhy0h)

[**Empleo de técnicas vistas en el curso 3**](#_bv83duhi2fec)

[Uso de la técnica de observación 3](#_sqcklz4sgzsm)

[Revisión de registros 3](#_205iistgntsr)

[Entrevistas 5](#_ofgp5eefknuq)

[**Cuestionarios 5**](#_2mp55rtfekvj)

[Datos generales 5](#_g2im9svv2s6f)

[Requisitos funcionales 6](#_suias4jt4cwt)

[Usabilidad y experiencia 7](#_pbdytz9tdsjh)

[Aspectos visuales 7](#_fas68qoa4kjg)

[Comentarios adicionales 8](#_yqa71y8wrhsa)

[**Requerimientos funcionales, no funcionales: 8**](#_t4dqyhoz3drz)

[1. Requerimientos Funcionales (RF) 8](#_n69624i7ca7g)

[Modo Seguimiento: 8](#_u5l2x9r7qtk9)

[Modo Juego Digital (Versión Local): 9](#_dfannre29sn6)

[Administración: 9](#_547qrayi8j7b)

[2. Requerimientos No Funcionales (RNF) 9](#_dcntr3cf26ow)

[Usabilidad: 9](#_prlg5kmxnavc)

[Rendimiento: 9](#_mbbvt01ic113)

[Seguridad: 9](#_j1sam8i2jv9i)

[Compatibilidad: 10](#_gvrb1ukg2gya)

[Mantenibilidad: 10](#_wvno10qkv7g4)

[3. Requerimientos de Alcance y Limitaciones 10](#_ovhdanlx1h79)

[4. Requerimientos Técnicos 10](#_3imgay5oz60s)

[Stack Tecnológico: 10](#_czwej0fp83jn)

[Entorno de Producción: 10](#_le18pmm54uy1)

[5. Requerimientos de Documentación 11](#_xn4l7pgfhkon)

[6. Criterios de Aceptación 11](#_z169l3xpo3tw)

[Resumen Ejecutivo 11](#_uorxwh6p54ub)

[1. Requisitos Funcionales 12](#_vwynq2r7y8d8)

[Funcionalidades esenciales del juego digitalizado: 12](#_c4c668od922v)

[Gestión de Partidas: 12](#_q2bv5vbw17oi)

[Lógica del Juego: 12](#_j5d10e5wsw9t)

[Dado de Colocación: 12](#_57b7xyxpdhj)

[Colocación de Dinosaurios: 12](#_m5nau2buzgh4)

[Puntuación Automática: 12](#_tplymq4vc3fy)

[Interfaz de Usuario: 12](#_opi9kp8stg9p)

[2. Requisitos No Funcionales 13](#_ltr0bn4ygsqo)

[Aspectos técnicos y de calidad: 13](#_6u5rrj18gz23)

[Rendimiento: 13](#_76t8rj45i41f)

[Usabilidad: 13](#_kssy05f5nxeh)

[Seguridad: 13](#_jh5pouyc0we4)

[Tecnologías: 13](#_olcj17y9i71b)

[3. Límites del Sistema 13](#_kegfjfgnmk93)

[Qué NO incluye el juego digitalizado: 13](#_i75uaxebpflr)

[4. Alcance 14](#_4z3v7n5sizcl)

[Cobertura del proyecto: 14](#_rtwrrvd55583)

[Modo Digitalizado: 14](#_i9dx2q759isu)

[Exclusiones: 14](#_9soh02rpfmqx)

[5. Reglas Específicas para 2 Jugadores 14](#_84eckvtzo7bb)

[Adaptaciones clave: 14](#_ddwbr5onioea)

[Paso de Dinosaurios: 14](#_cj8yf0npsh2d)

[Dado de Colocación: 14](#_15m7c0rhlv76)

[Ejemplo de Flujo: 14](#_o5dgrz5mo63u)

[6. Comparativa: 2 Jugadores vs. 3+ Jugadores 15](#_78w46fe457pz)

[Aspecto 15](#_izecabthlraw)

[Conclusión 15](#_v8e52204d6o)

[Ejemplo de validación para 2 jugadores: 15](#_c2mpkk1ejcfg)

[**Factibilidad Operativa 15**](#_am621uzen53u)

[Factibilidad Técnica 16](#_lf7140u0f92j)

[Requisitos mínimos de hardware para desarrollo y uso: 17](#_3g2sbsl3vn2)

[**Factibilidad Legal (Uruguay) 18**](#_9u1cppjtotq1)

[a) Propiedad Intelectual 18](#_g7h2dzvfqzec)

[b) Protección de Datos Personales 18](#_zexjeqowfqx1)

[c) Actividad Comercial y Registro 18](#_azw82wqph28n)

[d) Licencias de Software y Uso de Herramientas 19](#_1qej22aj71qw)

[**Propuesta de Modelo Organizativo para Dream Magic Wizard: Enfoque Descentralizado y Democrático 19**](#_ptt2hatvs82)

[**Implementación del Modelo de Desarrollo Tradicional en el Proyecto "Draftosaurus" 20**](#_o4ie35mfb7y7)

[1. Recolección y Análisis de Requisitos 20](#_9fi49v3j8dlt)

[2. Diseño del Sistema 20](#_6d93svnduglk)

[3. Implementación 20](#_kppc0ooxfd7a)

[4. Pruebas 21](#_i3ne0cor7lw9)

[5. Mantenimiento y Documentación 21](#_i6vzy3lv2svs)

### 

### 

### 

# **Empleo de técnicas vistas en el curso**

Durante el desarrollo del preanálisis y la etapa de relevamiento de información para el proyecto, se aplicaron diversas técnicas aprendidas en el curso, entre las cuales destaca:

## **Uso de la técnica de observación**

Se aplicó la observación no participante para relevar requerimientos del juego Draftosaurus, con foco en usuarios de 8 años o más.

El objetivo fue identificar comportamientos, necesidades, dificultades y preferencias de los jugadores durante una partida del juego de mesa físico, como insumo para el diseño de la versión digital.n:

Aspectos observados:

* Tiempo que tarda cada jugador en tomar decisiones.
* Necesidad de ayuda de un adulto o reglas confusas.
* Interacción entre jugadores (colaboración, competencia, diálogo).
* Interés visual por los dinosaurios y el tablero.
* Reacciones emocionales (aburrimiento, entusiasmo, frustración, concentración).
* Comentarios espontáneos de los jugadores.

## **Revisión de registros**

* Se examinaron documentos internos como manuales, políticas, reglas del juego.
* **Análisis realizado:**

Requerimientos cuantitativos y cualitativos

**Análisis Cuantitativo**

Número de partidas jugadas durante las sesiones de prueba.

Tiempo promedio por turno o partida (identifica posibles cuellos de botella o demoras).

Frecuencia de errores o dudas al aplicar las reglas.

Cantidad de jugadores por sesión (individual, en pareja, grupo).

Cantidad de veces que un jugador necesitó ayuda externa.

Porcentaje de niños que completaron la partida sin asistencia.

Resultados de cuestionarios cerrados (por ejemplo: “¿Te resultó fácil entender el juego?” Sí: 70%, No: 30%).

Utilidad: estos datos permiten optimizar tiempos de juego, simplificar reglas, ajustar dificultad y priorizar funcionalidades clave en la versión digital.

**Análisis Cualitativo**

Este análisis se enfoca en aspectos más subjetivos, basados en percepciones, opiniones, comportamientos y experiencias.

Comentarios espontáneos de los niños durante el juego (“¡Me gusta este dinosaurio!”, “No entiendo esta parte”).

Actitudes observadas: entusiasmo, distracción, frustración, cooperación.

Dificultades narradas por los adultos que explican el juego.

Lenguaje utilizado por los jugadores para describir reglas o jugadas.

Sugerencias o deseos expresados: “Estaría bueno que los dinos se movieran en la pantalla” o “Que tenga sonidos cuando colocas uno”.

Evaluación de la experiencia visual y estética: qué les gusta más del diseño.

* Utilidad: este análisis es clave para mejorar la experiencia de usuario, la interfaz, la narrativa, y la ambientación del juego digital..

# **Entrevistas**

Se realizaron entrevistas semiestructuradas, combinando preguntas abiertas y cerradas.

Objetivo: Obtener información rica en detalles y enfocada en los procesos y expectativas del usuario.

Ejemplo de preguntas abiertas: “¿Cómo reaccionaría su sector al nuevo sistema propuesto?”

#### 

# **Cuestionarios**

Se diseñaron cuestionarios con preguntas mixtas (abiertas y cerradas).

Utilizados para obtener datos de una mayor cantidad de personas de forma rápida y ordenada.

En algunos casos, se usaron cuestionarios online para facilitar la recolección.

—-------------------------------------------------------------------------------------------------------------

Formulario:  
Formulario de Relevamiento de Requerimientos

Juego: Draftosaurus (versión digital)  
 Edad objetivo: 8 años o más

### **Datos generales**

1. **Nombre del entrevistado:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. **Rol:**
   * ☐ Niño/a
   * ☐ Padre/Madre
   * ☐ Docente
   * ☐ Otro: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Edad del jugador/a (si aplica):** \_\_\_\_\_\_ años
2. **¿Tiene experiencia previa con juegos de mesa?**
   * ☐ Sí
   * ☐ No
   * Si respondió "Sí", ¿cuáles?: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### **Requisitos funcionales**

1. ¿Qué dispositivos utilizan habitualmente para jugar?
   * ☐ PC
   * ☐ Tablet
   * ☐ Celular
   * ☐ Otro: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. ¿Preferiría una versión del juego...?
   * ☐ Para un solo jugador
   * ☐ Para varios jugadores (modo multijugador)
   * ☐ Ambas
3. ¿Qué le gustaría que se pueda hacer en el juego digital? (Marcar todas las que correspondan)
   * ☐ Jugar contra la computadora
   * ☐ Jugar con amigos en línea
   * ☐ Jugar con otros en el mismo dispositivo
   * ☐ Ver estadísticas o logros
   * ☐ Personalizar dinosaurios o tableros
   * ☐ Aprender sobre dinosaurios
   * ☐ Otro: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### **Usabilidad y experiencia**

1. ¿Qué tan fácil debe ser el uso del juego?
   * ☐ Muy fácil (todo con imágenes)
   * ☐ Intermedio (mezcla de texto e imágenes)
   * ☐ Completo (instrucciones más largas y explicativas)
2. ¿Qué tipo de sonidos o música preferiría que tenga el juego?
   * ☐ Relajante y suave
   * ☐ Divertida y dinámica
   * ☐ Sin música
   * ☐ No sabe / No contesta
3. ¿Cuál de estas opciones cree que ayudaría a aprender más rápido a jugar?

* ☐ Tutorial interactivo
* ☐ Video explicativo
* ☐ Instrucciones escritas con dibujos
* ☐ Que otro jugador me enseñe

### **Aspectos visuales**

1. ¿Qué estilo de diseño preferiría para los dinosaurios y el entorno?

* ☐ Realista
* ☐ Caricaturesco / dibujo animado
* ☐ Minimalista / simple
* ☐ Otro: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. ¿Cuáles de estos colores cree que serían más atractivos para los niños? (Marcar hasta 3)

* ☐ Verde
* ☐ Azul
* ☐ Naranja
* ☐ Violeta
* ☐ Amarillo
* ☐ Rojo
* ☐ Otro: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### 

### **Comentarios adicionales**

1. ¿Hay alguna funcionalidad que le gustaría ver en el juego digital y que no haya sido mencionada?
2. ¿Algo que debamos tener en cuenta para hacerlo más divertido o accesible para niños/as de 8 años?

# **Requerimientos funcionales, no funcionales:**

### **1. Requerimientos Funcionales (RF)**

#### **Modo Seguimiento:**

RF-01: Registrar jugadores (nombre, ID único).

RF-02: Ingresar dinosaurios colocados en tableros por jugador.

RF-03: Validar automáticamente restricciones del dado de colocación (ej: "zona boscosa").

RF-04: Calcular puntuación por recinto según reglas del juego (ej: Bosque de la Semejanza = puntos por dinosaurios de misma especie).

RF-05: Mostrar resumen de puntuación final y determinar ganador.

#### 

#### **Modo Juego Digital (Versión Local):**

RF-07: Simular mecánicas base del juego (2 rondas, 6 turnos cada una).

RF-08: Generar dinosaurios aleatorios y distribuirlos entre jugadores.

RF-09: Implementar lanzamiento virtual del dado y aplicar restricciones.

RF-10: Interfaz drag-and-drop para colocar dinosaurios en recintos.

RF-11: Validar reglas de recintos en tiempo real (ej: Pradera del Amor = 5 pts por parejas).

RF-12: Sistema de turnos automatizado (paso de dinosaurios entre jugadores).

#### **Administración:**

RF-13: Gestionar perfiles de usuarios (registro, inicio de sesión).

RF-14: Configurar partidas (número de jugadores, dificultad).

### **2. Requerimientos No Funcionales (RNF)**

#### **Usabilidad:**

RNF-01: Interfaz responsive (móvil: 320px-576px, tablet: 577px-992px, escritorio: ≥993px).

RNF-02: Diseño intuitivo con iconografía acorde al tema (dinosaurios/parques).

#### **Rendimiento:**

RNF-03: Tiempo de respuesta <2 segundos para cálculos de puntuación.

RNF-04: Soporte para 2-5 jugadores simultáneos en modo digital.

#### **Seguridad:**

RNF-05: Protección contra SQL Injection en formularios PHP.

RNF-06: Autenticación básica para administradores.

#### **Compatibilidad:**

RNF-07: Funcionamiento en navegadores modernos (Chrome, Firefox, Edge).

RNF-08: Entorno de producción: Servidor GNU/Linux con LAMP.

#### **Mantenibilidad:**

RNF-09: Código modular (POO) documentado con estándar PHPDoc.

RNF-10: Control de versiones con Git (repositorio en GitHub).

### **3. Requerimientos de Alcance y Limitaciones**

Alcance Incluido:

Soporte para modo base de Draftosaurus (sin expansiones).

Localización en español/inglés.

Exportación de resultados en PDF.

Limitaciones Excluidas:

No incluye modo multijugador en línea (solo local).

No soporta personalización avanzada de reglas.

### **4. Requerimientos Técnicos**

#### **Stack Tecnológico**:

Frontend: HTML5, CSS3, JavaScript (Bootstrap 5.3 + jQuery para interacciones).

Backend: PHP 8.x (arquitectura MVC).

Base de Datos: MySQL/MariaDB (normalizada hasta 3FN).

Herramientas: XAMPP 8.x, Visual Studio Code, Git.

#### **Entorno de Producción:**

Servidor Apache en GNU/Linux (Ubuntu/CentOS).

Configuración manual de LAMP (Linux, Apache, MySQL, PHP).

### **5. Requerimientos de Documentación**

Manual de Usuario: Instrucciones para jugadores y administradores.

Manual Técnico: Instalación, estructura de BD, API (si aplica).

Diagramas UML: Casos de uso, clases, secuencia (coordinación con Ingeniería de Software).

### **6. Criterios de Aceptación**

CA-01: La puntuación se calcula correctamente en todos los recintos.

CA-02: El dado aplica restricciones válidas en cada turno.

CA-03: La interfaz se adapta a dispositivos móviles sin errores.

CA-04: Los datos de partidas guardadas persisten tras reiniciar la aplicación.

### **Resumen Ejecutivo**

Este documento cubre todos los requisitos para cumplir con los objetivos del proyecto, alineados a las rúbricas de evaluación del ITI. Los RF y RNF priorizan funcionalidades clave del juego físico (modo seguimiento) y su versión digital, garantizando viabilidad técnica y experiencia de usuario óptima.

Nota: Los requisitos están sujetos a ajustes durante las entregas parciales (hitos del 14/07, 15/09, y 03/11/2025).

## 

## **1. Requisitos Funcionales**

### **Funcionalidades esenciales del juego digitalizado:**

### **Gestión de Partidas:**

Crear partidas locales con 2 a 6 jugadores.

Configurar parámetros iniciales (tableros, modo de paso de dinosaurios).

#### **Lógica del Juego:**

#### **Dado de Colocación:**

Generar restricciones aleatorias (ej: "zona izquierda", "recinto vacío").

Aplicar restricciones a todos los jugadores (en 2 jugadores, nadie está exento).

#### **Colocación de Dinosaurios:**

Validar reglas de recintos (ej: "Prado de la Diferencia" solo acepta especies distintas).

Implementar drag and drop para colocar dinosaurios.

#### **Puntuación Automática:**

Calcular puntos por recinto, dinosaurios en el río (+1) y T-Rex (+1 por recinto).

#### **Interfaz de Usuario:**

Mostrar tableros interactivos por jugador.

Panel de control con dado, historial de turnos y puntuación en tiempo real.

## **2. Requisitos No Funcionales**

### **Aspectos técnicos y de calidad:**

### **Rendimiento:**

Tiempo de respuesta <1 segundo en acciones críticas (lanzar dado, calcular puntuación).

#### **Usabilidad:**

Diseño responsivo (móvil, tablet, escritorio).

Accesibilidad (contrastes de color, tamaños de texto ajustables).

#### **Seguridad:**

Validación backend de reglas para evitar trampas.

Consultas SQL parametrizadas (prevención de inyección).

#### **Tecnologías:**

Frontend: HTML5, CSS3, Bootstrap 5.3, JavaScript.

Backend: PHP 8.x + MySQL/MariaDB.

Arquitectura en 3 capas (presentación, negocio, datos).

## **3. Límites del Sistema**

### **Qué NO incluye el juego digitalizado:**

Multijugador online: Solo modo local (mismo dispositivo).

Expansiones: Solo el modo base (sin variantes como Marina o Aéreo).

## **4. Alcance**

#### **Cobertura del proyecto:**

#### **Modo Digitalizado:**

Réplica fiel del juego físico con todas sus reglas.

Soporte para 2-6 jugadores (con adaptaciones para 2 jugadores).

#### **Exclusiones:**

No es un reemplazo del juego físico (no incluye escaneo de componentes reales).

No genera informes estadísticos avanzados (solo puntuación final).

## **5. Reglas Específicas para 2 Jugadores**

### **Adaptaciones clave:**

#### **Paso de Dinosaurios:**

(Descarte Directo):

Los dinosaurios no colocados se descartan y se roban nuevos en la siguiente ronda.

#### **Dado de Colocación:**

Afecta a ambos jugadores (en partidas de 2, nadie está exento).

#### **Ejemplo de Flujo:**

Jugador A lanza el dado (ej: "recinto vacío").

Ambos eligen 1 dinosaurio de su mano y lo colocan en un recinto vacío (o en el río si no es posible).

Los dinosaurios restantes se pasan al jugador fantasma o se descartan.

## **6. Comparativa: 2 Jugadores vs. 3+ Jugadores**

### **Aspecto**

| **Elemento** | **2 Jugadores** | **3+ Jugadores** |
| --- | --- | --- |
| Paso de Dinosaurios | Jugador fantasma o descarte directo | Pasan al jugador de la izquierda |
| Dado | Afecta a ambos jugadores | No afecta al lanzador |
| Estrategia | Más control (menos aleatoriedad) | Mayor incertidumbre (los dinosaurios giran) |

## **Conclusión**

El juego digitalizado debe:

Implementar las reglas estándar con adaptaciones claras para 2 jugadores.

Garantizar usabilidad en dispositivos pequeños.

Limitarse al modo base (sin expansiones).

#### **Ejemplo de validación para 2 jugadores:**

Si un jugador no puede colocar un dinosaurio por la restricción del dado:

- El sistema fuerza la colocación en el río (+1 punto).

- Muestra un mensaje claro: "¡Debes colocar en el río por la restricción del dado!".

## **Factibilidad Operativa**

Con el objetivo de garantizar la calidad, escalabilidad y mantenibilidad del proyecto, se implementará una metodología robusta en cuanto a la arquitectura del software, basada en buenas prácticas de desarrollo, patrones de diseño y una estructura modular que permita futuras mejoras y adaptaciones.

Asimismo, se elaborará una guía de usuario clara y accesible, compuesta por instrucciones simples y precisas para jugar, de forma que cualquier persona pueda comprender fácilmente el funcionamiento del juego.

Se priorizarán los principios de accesibilidad, usabilidad e intuición, con el fin de que la experiencia del usuario sea fluida, amigable y sin barreras innecesarias. El diseño de la interfaz tendrá en cuenta:

* El uso de colores contrastantes y tipografías legibles.
* Navegación simple y directa.

Todo esto contribuirá a una experiencia inclusiva, profesional y centrada en el usuario, alineada con los valores de *Dreams Magic Wizard*.

## **Factibilidad Técnica**

En el desarrollo de nuestro proyecto se priorizará la optimización del uso de memoria, con el objetivo de minimizar el consumo de recursos sin comprometer la confiabilidad ni la disponibilidad de la información. Asimismo, al tratarse de una aplicación web, se garantiza una alta compatibilidad con una amplia variedad de dispositivos, incluyendo computadoras, teléfonos móviles y otros equipos que cuenten con un navegador compatible con JavaScript. Esto permitirá un acceso flexible y eficiente al sistema desde múltiples plataformas.

### **Requisitos mínimos de hardware para desarrollo y uso:**

SO: Windows 10.

Procesador: Intel Core i3 (8va generación o superior) o equivalente.

RAM: 8 GB mínimo (16 GB recomendado para desarrollo).

Almacenamiento: 10 GB de espacio disponible.

Pantalla: Resolución mínima 1366x768 (Full HD recomendada).

Conectividad: Acceso a internet estable para pruebas y despliegue.

**Navegador**  **Versión mínima recomendada**

Google Chrome 61+

Mozilla Firefox 60+

Microsoft Edge (Chromium) 79+

Safari (macOS/iOS) 11+

Opera 48+

Android WebView 61+

## **Factibilidad Legal (Uruguay)**

El desarrollo de la aplicación *Draftosaurus* es legalmente viable dentro del marco normativo de la República Oriental del Uruguay. La legislación nacional permite y fomenta la creación de software y aplicaciones digitales, tanto con fines educativos como comerciales.

A continuación se detallan los principales aspectos legales relevantes:

### **a) Propiedad Intelectual**

Los contenidos originales desarrollados en el marco del proyecto (código fuente, diseños, logotipos, documentación, etc.) están amparados por la **Ley Nº 9.739 sobre Derechos de Autor**, por lo que el grupo puede registrar la propiedad intelectual si lo desea, garantizando así la protección de sus creaciones.

### **b) Protección de Datos Personales**

En caso de que la aplicación recoja o procese información personal de los usuarios, se deberá cumplir con la **Ley Nº 18.331 sobre Protección de Datos Personales y Acción de Habeas Data**, que establece los principios de consentimiento, finalidad, seguridad y acceso por parte del titular de los datos.

### **c) Actividad Comercial y Registro**

Si en el futuro se desea comercializar la aplicación, se podrá registrar una forma jurídica adecuada, como una **Sociedad de Responsabilidad Limitada (SRL)** o una **Unipersonal**, y realizar la inscripción correspondiente ante:

* Dirección General Impositiva (DGI)
* Banco de Previsión Social (BPS)
* Ministerio de Trabajo y Seguridad Social (MTSS)
* Agencia para el Desarrollo del Gobierno de Gestión Electrónica y la Sociedad de la Información y del Conocimiento (AGESIC), si corresponde por uso de servicios digitales.

### **d) Licencias de Software y Uso de Herramientas**

El desarrollo se realizará utilizando herramientas y librerías de código abierto o con licencias compatibles con el uso educativo y/o comercial, respetando las condiciones de uso definidas por sus respectivos autores.

# **Propuesta de Modelo Organizativo para Dream Magic Wizard: Enfoque Descentralizado y Democrático**

Consideramos que el modelo organizativo más adecuado para nuestro equipo en Dream Magic Wizard es el **Descentralizado Democrático**. Este enfoque se basa en el reconocimiento de que todos los integrantes del equipo poseen talentos y habilidades únicas, que pueden complementarse y potenciarse mutuamente.

En muchas ocasiones, un miembro del equipo puede guiar a otro que enfrenta dificultades en una determinada tarea, generando un ambiente de colaboración activa y aprendizaje mutuo. Además, al contar con roles de coordinación distribuidos, se fomenta la participación equitativa en la toma de decisiones, permitiendo que cada integrante aporte desde su área de fortaleza.

Este modelo también facilita la distribución eficiente de tareas, ya que permite derivar responsabilidades específicas a quienes poseen mayor conocimiento o experiencia en cada aspecto del proyecto. De este modo, se puede avanzar de manera ordenada y cohesiva hacia una primera entrega sólida y representativa del trabajo colectivo.

# **Implementación del Modelo de Desarrollo Tradicional en el Proyecto "Draftosaurus"**

El modelo de desarrollo tradicional, también conocido como modelo en cascada, se caracteriza por una secuencia lineal y estructurada de fases. Cada etapa debe completarse completamente antes de avanzar a la siguiente, permitiendo una planificación y documentación exhaustiva. En el desarrollo del proyecto Draftosaurus, este modelo ha sido implementado con el objetivo de asegurar una construcción ordenada y predecible del sistema.

## **1. Recolección y Análisis de Requisitos**

En esta primera etapa, se identificaron y documentaron los requisitos funcionales y no funcionales del sistema Draftosaurus. Se generaron formularios de consultas para obtener información relevante para el desarrollo del proyecto, los tipos de acciones disponibles, el número de jugadores y las condiciones de victoria. Esta información permitió establecer las bases del sistema a desarrollar.

## **2. Diseño del Sistema**

A partir de los requisitos definidos, se elaboró el diseño del sistema, incluyendo la arquitectura general y la interfaz de usuario. Se priorizó la modularidad y la escalabilidad, contemplando tanto el flujo del juego como la gestión de turnos, dados, tableros y dinosaurios.

## **3. Implementación**

En esta etapa se comenzó con la construcción inicial del sistema, priorizando el desarrollo de la estructura base del proyecto y el diseño visual de la interfaz. Se utilizó **HTML5** y **CSS3** con el apoyo del framework **Bootstrap** para lograr un diseño responsivo, moderno y coherente con la temática del juego. También se incorporaron las primeras funciones básicas utilizando **JavaScript**, aunque su implementación aún se encuentra en una fase inicial.

Hasta el momento, no se ha avanzado en la lógica del juego ni en la conexión con bases de datos, ya que el foco ha estado en el diseño de la interfaz de usuario y en sentar las bases estructurales del sistema para facilitar futuras integraciones.

## **4. Pruebas**

Se realizaron pruebas básicas sobre las tres secciones desarrolladas hasta el momento: la página de **inicio**, el **tablero** y la sección de **rankings**. Estas pruebas se enfocaron principalmente en la correcta visualización, la navegación entre páginas y el comportamiento general de los elementos de la interfaz. Dado que el sistema aún se encuentra en una fase temprana de desarrollo, no se llevaron a cabo pruebas sobre lógica de juego ni validaciones funcionales avanzadas.

## **5. Mantenimiento y Documentación**

Aunque el modelo en cascada no contempla iteraciones, se ha considerado una fase posterior de mantenimiento para corregir errores detectados durante las pruebas finales y realizar mejoras menores. Se documentó todo el desarrollo, incluyendo el manual de juego, facilitando futuras updates.