



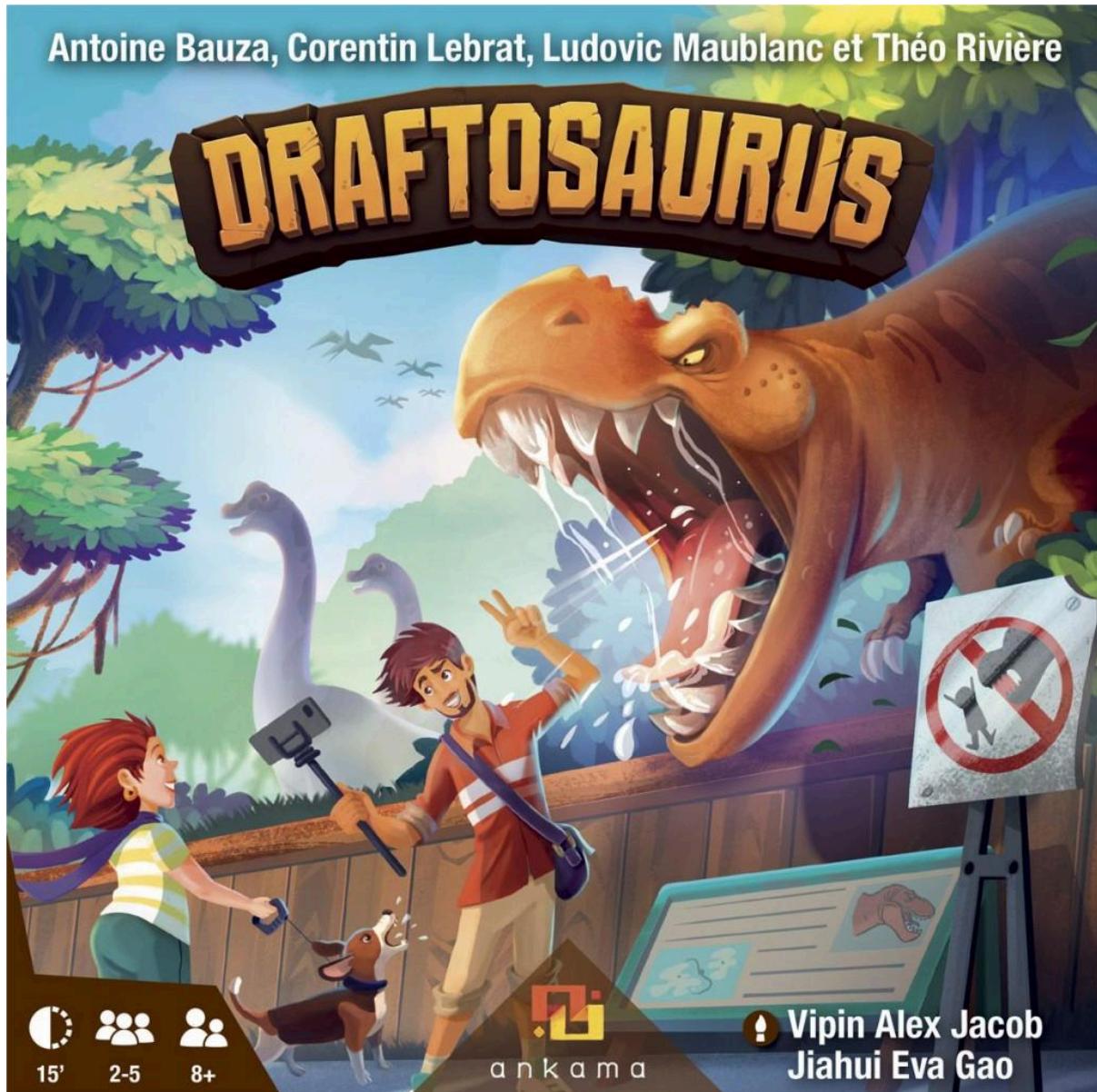
BALATECH

3ºMA

ITI

S.I.G.P.D

Ingenieria En Software



Descripción de S.I.G.P.D

El S.I.G.P.D es un proyecto diseñado para los estudiantes del bachillerato Tecnológico en tecnologías de la informática del IAE Montevideo.

Este proyecto consiste en desarrollar una versión digitalizada del juego de mesa conocido como Draftosaurus esta versión deberá permitir al usuario jugar de forma local una partida normal del mismo. Este proyecto busca la realización correcta del proyecto implementando los conceptos y habilidades aprendidas en las materias participantes de este proyecto. Este proyecto se realizará en grupos formados por 3 a 4 estudiantes por equipo que deberán comprometerse con el equipo.

Organización del proyecto

Diagrama de gantt

Definición Diagrama de Gantt

El Diagrama de Gantt es una herramienta de planificación y control utilizada en la gestión de proyectos, cuya finalidad principal es representar de manera gráfica y cronológica las actividades que deben desarrollarse para alcanzar los objetivos propuestos. Fue creado por Henry L. Gantt a comienzos del siglo XX y, desde entonces, se ha consolidado como uno de los instrumentos más difundidos y eficaces para la organización del trabajo en proyectos de diversa índole.

En términos de funcionamiento, el diagrama se estructura mediante un eje horizontal que representa el tiempo (expresado en días, semanas o meses) y un eje vertical que muestra las actividades o tareas del proyecto. Cada actividad se representa a través de una barra horizontal cuya posición indica la fecha de inicio y de finalización, mientras que su extensión refleja la duración prevista. Esta representación visual permite observar de forma clara la secuencia de tareas, la superposición de actividades y las dependencias entre ellas. Asimismo, es posible identificar hitos relevantes, plazos de entrega y períodos de mayor carga de trabajo.

La aplicación del Diagrama de Gantt en un proyecto resulta de gran utilidad, ya que facilita la planificación detallada y el seguimiento del avance. En primer lugar, permite distribuir de manera ordenada las actividades en el tiempo, lo que contribuye a optimizar los recursos humanos y técnicos disponibles. En segundo término, ofrece una visión global que favorece la comunicación entre los integrantes del equipo, ya que todos pueden visualizar de manera sencilla las responsabilidades



BALATECH

3°MA

ITI

asignadas y los plazos comprometidos. Finalmente, concretarán medidas correctivas.

En el caso de nuestro proyecto, la implementación de un Diagrama de Gantt es especialmente valiosa porque permite estructurar las diferentes etapas de desarrollo dentro de un cronograma definido. Esto asegura un control riguroso del tiempo, facilita la coordinación entre los integrantes y aumenta la probabilidad de cumplir con las fechas de entrega establecidas por la institución.

Este diagrama de gantt se realizo con el fin de mostrar las actividades hechas por el equipo con la fecha de cada una tambien asi mostrando los responsables de cada una de ellas.

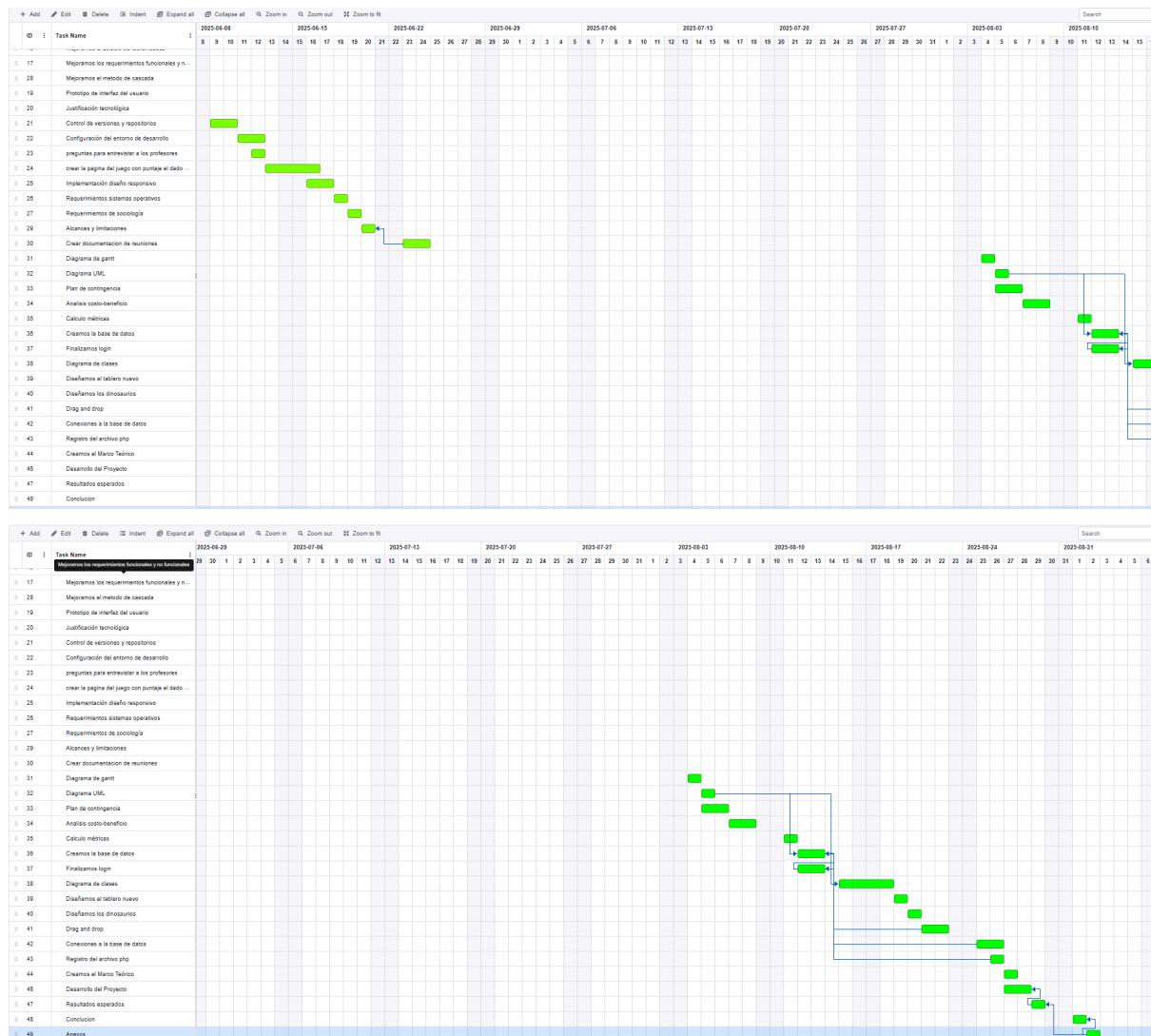




BALATECH

3ºMA

ITI



Implementacion de metodología

Aqui mostraremos la implementación requerida por el profesor de ingenieria en software

Ciclo de vida Incremental

Incremento 1:

-Aspectos técnicos y organizativos-



3ºMA

ITI

- Diseño inicial de interfaz de usuario (UI)
 - Maquetación en HTML5 y estilización en CSS3, aplicando principios de usabilidad
 - Prototipo navegable de las siguientes pantallas:
 - Página de bienvenida
 - Tablero del juego
 - Pantalla de reglas.
 - Diseño responsive
 - Elección y justificación del sistema de control de versiones en GitHub, creación del repositorio inicial y documentación de buenas prácticas de commits
-

-Planificación y estructura del proyecto-

- Organización del equipo, roles, normas internas, etc.
 - Diagrama de Gantt y cronograma tentativo
 - Carta de presentación del equipo
 - Prototipos gráficos del tablero y dado
 - Justificación tecnológica de las herramientas utilizadas como Bootstrap, GitHub, etc.).
-

-Primeros elementos funcionales del sistema-

- Registro manual de jugadores
-



3°MA

ITI

-Próximos incrementos-

- Lógica del juego completa
- Aplicación de restricciones al dado
- Validación automática de jugadas
- Interfaz en Ingles
- Instalación en entorno LAMP
- Agregar base de datos estable

Primera Entrega: 15/7/25

Roles integrantes del proyecto

Carta de Presentacion

Balatech
Prof.BRANDON CAIRÚS
Ingenieria en Software

Montevideo, 27 mayo de 2025

A continuación, los alumnos de 3°MA del turno matutino del ITI,nos complace presentarle Balatech, una iniciativa tecnológica nacida con la visión de transformar ideas en soluciones innovadoras y eficientes los correspondientes integrantes con sus roles son los siguientes:

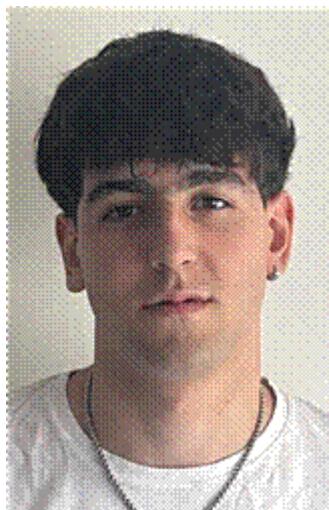


3ºMA

ITI

Rol	Nombre	Apellido	C.I	email	Teléfono
Coordinador	Luciano	Gonzalez	5.698.242 -6	lu.gonzafer@gmail.com	09856044 7
Sub-Coord	Emiliano	Moreira	5.669.126 -9	emilianomoreira@gmail. com	09846343 4
Integrante 1	Matias	Porta	5.708.353 -6	matipo33@gmail.com	09166551 8

Miembros Equipo:



Coordinador
Luciano Gonzalez

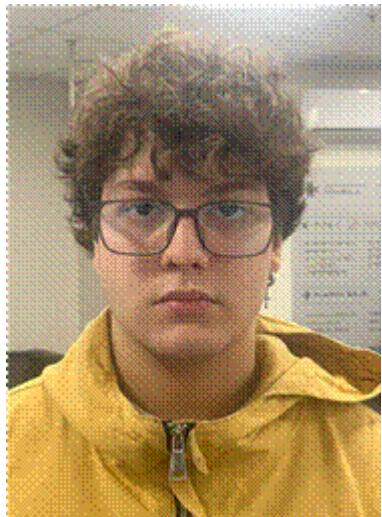
Firmas:

Coordinador
Luciano Gonzalez



3ºMA

ITI



Emiliano

Sub-Coordinador
Emiliano Moreira



Integrante 1
Matias Porta

Sub-Coordinador
Emiliano Moreira

Mp

Integrante 1
Matias Porta



3ºMA

ITI

Reglamento reuniones del grupo

- ° Llevar dudas, consultas, desacuerdos o imprevistos para plantear en la misma
- ° Respeto TOTAL entre los presentes
- ° No utilizar el celular mientras dictamos los planteamientos
- ° Se hablará única y exclusivamente sobre temas del proyecto
- ° Serán de entre 10 y 15 minutos
- ° Se deberá tener suma responsabilidad con la asistencia en caso de ausentarse deberá brindar la razón por la ausencia.

Horarios del grupo

El grupo llamado BalaTech conformado por Luciano Gonzalez, Matias Porta y Emiliano Moreira, tendrá una disposición horaria de entre 6 a 8 horas diarias ya que utilizamos 8 horas de sueño y las demás para las actividades normales del día al día.

Decidimos esa estimación horaria ya que consideramos que es apta para las necesidades de nuestro proyecto ya que creemos que es importante dedicarle la mayor cantidad de tiempo posible sin dejar de lado nuestra vida “normal” para poder llegar a hacer un mejor trabajo.

Horarios:

°Disponibles a partir de las: 8

°Disponibles hasta las:14-16hs

°Descanso de 30 minutos para almorzar a la hora que decida el integrante



3ºMA

ITI

Técnicas de relevamiento

En el correspondiente fragmento mostraremos dos puntos:

- ° Bibliografía
- ° Análisis Etnográfico

Bibliografía

Anaka y board game box. (2019). Draftosaurus. Diseñado por Antonie Bauza , Corentine Lebrat, Ludovic Maublanc y Théo Rivière

Juego de mesa (s.f) Draftosaurus.

<https://boardgamegeek.com/boardgame/269259/draftosaurus>

Anaka Games (s.f) Draftosaurus – Manual de Reglas

https://drive.google.com/file/d/138qY_aZfQ-RXYDA0j6HshSk-_1mmJlrG/view

Como Jugar Draftosaurus. Youtube.

<https://www.youtube.com/watch?v=-ZyFqRNkiAU>

CETP-UTU.(2025). Letra Del Proyecto De Egreso – BT- Tecnología de la información.

IAE Montevideo – I.S.B.O

Análisis etnográfico



3ºMA

ITI

La comunidad de usuarios observada está compuesta por gente desde la edad de 4 años hasta cualquier edad, principalmente lo juegan los adolescentes, adultos o gente aficionada por los juegos de mesa. Las partidas se desarrollan generalmente en entornos recreativos o en momentos familiares en busca de diversión o pasar el rato.

Para este análisis, se recurrió a contenidos de videos de la plataforma Youtube, simuladores del juego, experiencia de juego nuestra y de personas externas. También se tomaron en cuenta entrevistas informales con jugadores “frecuentes” también recurriendo a contenido de youtube como medio de información.

A través de juego previo entre nosotros y tras investigar a fondo, se identificaron los siguientes patrones:

- Los jugadores tienden a olvidar las reglas de ciertos recintos debido a la gran cantidad, especialmente los que requieren de condiciones específicas como por ejemplo exactamente 3 dinosaurios o único en su especie.
 - Las restricciones del dado generan dudas frecuentes, sobre todo a jugadores iniciantes o que jugaron poco
 - Hay tendencia a errores de conteo de puntos, especialmente en partidas rápidas o sin una supervisión rigurosa
 - Algunos jugadores prefieren ver visualmente el tablero y al lado los puntos, pero en juego físico no siempre facilita esa visión general y más cuando son muchos jugadores
 - En contextos sociales, la prioridad es la dinámica rápida y divertida, por lo que cualquier elemento que frene el juego como por ejemplo reglas concretas o conteo lento afecta negativamente la experiencia
-

Con base en estas observaciones, se identificaron varias necesidades clave que la aplicación debe resolver:



3ºMA

ITI

1-Validación automática de las reglas: Esto evitara errores a la hora de colocar dinosaurios en recintos inválidos.

2-Cálculo automático de puntuación: para reducir “discusiones”, hacer mas dinámico el juego y acelerar el proceso

3- Interfaz clara y visual: Que muestre el parque de cada jugador y su progreso sin necesidad de recordar manualmente cada recinto.

4- Registro del dato y advertencia si no se cumple

5-Opcionalidad de modo seguimiento para usar en partidas físicas y modo digital para jugar desde la app

Conclusiones:

Tras realizar el análisis etnográfico concluimos que la mayoría de jugadores de draftosaurus priorizan la fluidez, claridad y sencillez al momento de jugar. La aplicación S.I.G.P.D debe responder a esas expectativas, actuando como asistente inteligente que reduce el margen de error, que elimine la necesidad del cálculo manual y que mejore la experiencia general del juego, tanto para su versión física y digital.

Esto justifica el desarrollo del sistema propuesto, enfocado en automatizar procesos, validar reglas en tiempo real y acceder la puntuación para todos los jugadores.

Requerimientos

Requerimientos Funcionales

1.1- Funcionalidades del soporte para juego físico

Permitir a los jugadores ingresar manualmente los dinosaurios colocados en sus tableros



BALATECH

3ºMA

ITI

Aplicar automáticamente las restricciones impuestas por el dado, como:

Zona izquierda o derecha del tablero

Zona boscosa o rocosa

Recinto vacío o que no contenga T-Rex

Validar las reglas de cada recinto del parque:

Bosque de la Semejanza: sólo dinosaurios iguales, sin
huecos

Prado de la Diferencia: sólo dinosaurios distintos

Pradera del Amor: cuenta parejas de una misma especie

Trío Frondoso: solo válido si hay 3 dinosaurios

Rey de la Selva: 7 puntos si se tiene mayoría de una especie



3ºMA

ITI

Isla Solitaria: 7 puntos si es el único de su especie en todo el parque

Calcular la puntuación total, incluyendo bonificaciones:

1 punto por dinosaurio en el río

+1 punto por recinto con al menos un T-Rex

Determinar y mostrar automáticamente al jugador ganador

Proveer una interfaz de tablero digital para el seguimiento visual de las colocaciones de los dinosaurios

1.2- Funcionalidades del juego completo desde la app

Implementar la lógica de juego por rondas (2 rondas, 6 turnos cada una)

Simular la selección de dinosaurios aleatoriamente desde un “inventario” virtual

Gestionar el paso de dinosaurios entre jugadores

Permitir el lanzamiento del dado virtual y aplicar su restricción automáticamente



3ºMA

ITI

Validar en cada turno la legalidad de las acciones del jugador

Actualizar en tiempo real el estado del tablero de cada jugador

Calcular y mostrar la puntuación parcial y final por recinto y en general

Manejar múltiples perfiles de jugador (2 a 5 jugadores)

Generar un resumen final de partida (resultados, errores, puntajes, etc.)

Registrar historial de partidas para cada usuario (opcional)

1.3- Interacción del Usuario y Perfiles

Jugador:

Ingreso de jugadas

Visualización del tablero y puntuaciones



3ºMA

ITI

Participación en el juego o seguimiento

Administrador:

Alta, baja y modificación de usuarios

Configuración y mantenimiento del sistema

Control del backend y base de datos

Interfaz de ingreso manual de datos para jugadores físicos

Visualización de tableros personalizados y dinámicos por usuario

Requerimientos No Funcionales

1- Tecnología y Herramientas

Backend: PHP

Frontend: HTML5, CSS, JavaScript



3ºMA

ITI

Base de Datos: MySQL

Entorno de desarrollo: XAMPP durante el desarrollo, y servidor LAMP en entorno GNU/Linux para instalación final

Control de versiones: GitHub

Virtualización: VirtualBox o VMware Workstation 17

2.2- Infraestructura

El sistema deberá ser ejecutable en un servidor con sistema operativo Linux (Ubuntu o Debian)

Compatible con navegadores web modernos

Aplicación pensada para uso local, sin necesidad de conexión a Internet

2.3- Accesibilidad y Usabilidad

Interfaz amigable, intuitiva y de fácil navegación para usuarios no técnicos

Visualización clara del tablero, puntuaciones y opciones de juego



3ºMA

ITI

Adaptado para ser utilizado en pantallas medianas y chicas (PC, Laptop, Celular, etc)

Idioma: Español, pero que tenga posibilidad de extender a Inglés

2.4- Documentación y Mantenimiento

Manual de usuario para jugadores

Manual técnico para administradores del sistema

Código comentado y documentado

Instrucciones de instalación en entorno local

2.5- Seguridad y Estabilidad

Validación de datos en frontend y backend

Control de errores de entrada del usuario

Sistema robusto para evitar bloqueos por acciones no válidas

Base de datos alojada en servidor local, sin acceso externo



Alcance:

Valida que las reglas del juego sean las correctas

Calcula automáticamente la puntuación final

Implementación completa del juego para jugar localmente desde la web

Tiene lógica de selección y colocación de dinosaurios

Se ejecuta en una base de datos en MySQL con GNU o Linux

Limitaciones:

No reemplaza al juego físico ni es el juego completo, solo es una versión para un sitio web

Los usuarios deben ingresar manualmente los dinosaurios colocados

No hay modo online

No tiene nada diferente al juego original, no hay contenido extra ni reglas modificadas

Hecho con HTML, CSS y JavaScript

Isla Solitaria: 7 puntos si es el único de su especie en todo el parque

Calcular la puntuación total, incluyendo bonificaciones:

1 punto por dinosaurio en el río



3ºMA

ITI

+1 punto por recinto con al menos un T-Rex

Determinar y mostrar automáticamente al jugador ganador

Proveer una interfaz de tablero digital para el seguimiento visual de las colocaciones de los dinosaurios

Factibilidades

Factibilidad Técnica

Nuestro equipo de desarrollo cuenta con conocimientos previos y habilidades en tecnologías web HTML, CSS, JavaScript para el frontend, también tendremos conocimientos de PHP para la lógica del backend. Estas herramientas serán alineadas con los requerimientos definidos en la letra del proyecto S.I.G.P.D.

Además, el grupo tiene experiencia previa en el uso de entornos como XAMPP y GitHub, así como como de Java y Base de datos MySQL/MariaDB, lo que garantiza la capacidad técnica del grupo para desarrollar una aplicación funcional, tanto en su versión de seguimiento como en su modalidad de juego digitalizado. Por lo tanto, se concluye que la factibilidad técnica es ampliamente positiva

Factibilidad Operativa

Desde el punto de vista operativo, el sistema propuesto es totalmente viable ya que replica las reglas y mecánicas del juego Draftosaurus, conocidas y documentadas. Se prevé implementar dos modos de uso:

-Modo Seguimiento: En este modo los jugadores ingresarán las acciones realizadas y el sistema calculará automáticamente los puntajes y aplicará restricciones.

-Modo de juego digitalizado: Aquí se podrá jugar el juego completamente desde la aplicación, replicando el flujo del juego real.



3ºMA

ITI

La solución propuesta mejora la experiencia del usuario, reduce los errores de conteo y facilita la aplicación y reglas, lo que anticipa una aceptación favorable por parte de los usuarios finales. Se concluye que la factibilidad operativa es alta

Factibilidad Legal

Este proyecto se basa en la estructura del juego Draftosaurus, pero no pretende reproducir ni distribuir el producto original con fines comerciales. Se utilizaran las reglas oficiales del juego y se utilizaran los mismos elementos gráficos, incluyendo tablero, fichas y dado.

Se trata de un desarrollo educativo, sin ánimo de lucro, orientado a fines formativos. En este proyecto se mencionan los autores y enlaces oficiales en la bibliografía del proyecto. Por lo tanto no existen impedimentos legales para su realización dentro del contexto académico.

Factibilidad Económica

El proyecto no requiere de una inversión económica significativa, ya que todas las herramientas necesarias para su desarrollo son de uso libre y gratuito, como PHP, MySQL, Visual Studio Code, GitHub, entre otras

Los recursos físicos necesarios como computadoras y conexión a internet ya están disponibles por el centro educativos o por los integrantes del grupo. Este trabajo se basa en esfuerzo, tiempo y dedicación del equipo, sin requerimientos de financiamiento externo, compra de software, de programas o de hardware adicional. Por lo tanto la factibilidad económica es completamente viable

Definición de roles

Usuario

El usuario utiliza la pagina e aplicación con objetivos funcionales. Este mismo no tendrá acceso ni a la configuración ni administración del sistema, pero interactúa con las funcionalidades proporcionadas para cumplir sus objetivos.

Puede:



3ºMA

ITI

- Consultar información disponible.
- Realizar acciones básicas como jugar, visualizar reglas, girar dado, etc.
- Gestionar su propia cuenta (Crear perfil, editar perfil, cambiar contraseña, etc.).

Permisos:

- Acceso limitado a las funcionalidades disponibles.
- No puede modificar configuraciones globales ni acceder a datos de otros usuarios

Administrador

El administrador es responsable del control y el mantenimiento del sistema. Tiene privilegios extendidos para gestionar usuarios, contenido y configuraciones.

Responsabilidades:

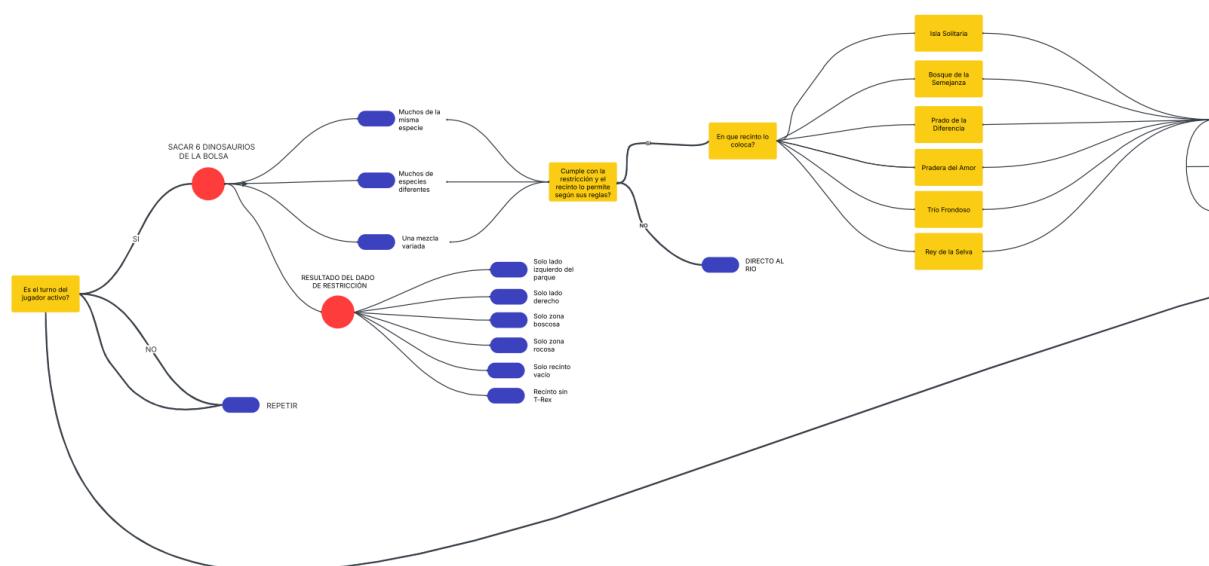
- Supervisar el correcto funcionamiento del sistema
- Gestiona el contenido del sistema
- Visualiza reportes, quejas, etc.

Permisos:

- Acceso completo a todas las funcionalidades del sistema.
- Modificación y configuración de parámetros internos.
- Capacidad de intervenir en cuentas de usuarios (eliminar cuentas o modificar algunas)



Lógica del sistema



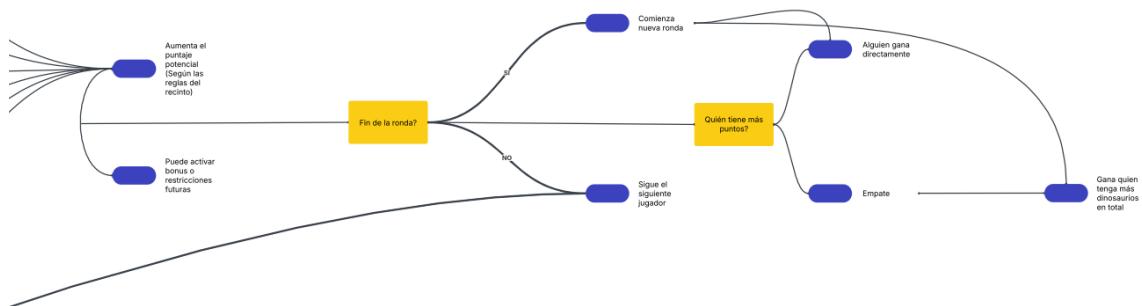


Tabla de decisiones

Como Jugar

Ronda 2	Sacar 6 dinosaurios nuevos	Mostrar en mano al jugador
luego de colocar	Quedan dinosaurios en mano	Pasar dinosaurio al jugador siguiente
Turno 6	No quedan dinosaurios	Finaliza la ronda



3ºMA

ITI

Finalización del juego

Rondas completadas	Dinosaurios colocados	Acción del sistema
2 rondas completas	12 dinosaurios en el tablero	Calcular puntaje total por recinto
2 rondas completas	hay dinosaurios en el rio	sumar un punto por cada uno
Hay T-REX en algun recinto		Sumar punto extra por cada uno
Hay empate de puntos	Jugador con mas dinosaurios gana	Determinar ganador

Software a utilizar

S.O a utilizar	Etapa de desarrollo	Acción
	Etapa de desarrollo local	Usar entorno XAMPP sobre windows
	Etapa final	Usar servidor GNU/Linux con LAMP

Lenguajes programacion

Función del sistema	Acción
Backend	PHP
Frontend	HTML, CSS Y JavaScript
validación de reglas	Lógica JavaScript

Gestor base de datos



3ºMA

ITI

Base de datos

Acción

Durante

MySQL y MariaDB en XAMPP

SEGUNDA ENTREGA INGENIERIA (S.I.G.P.D)

El presente documento Corresponde al área de Ingenieria en Software del proyecto de egreso de la ITI, Orientado al desarrollo de del Sistema Informático de Gestión de Partidas para Draftosaurus (S.I.G.P.D). Su objetivo principal es mantener las bases metodológicas y técnicas que guiaran la planificación y análisis, diseño e implementación de la solución propuesta, en concordancia con los lineamientos académicos y profesionales de la disciplina.

La ingeniería en software cumple un rol central en la organización y control del proyecto, ya que permite agregar un enfoque sistemático para transformar los requerimientos del juego Draftosaurus en un sistema web funcional y confiable. A través de este documento se detallan aspectos clave como diagramas de uso, planes de contingencia, análisis de costo-beneficio, cálculo de métricas, etc.

3ºMA

ITI

De esta forma el documento no solo constituye una guia técnica para el equipo sino que tambien funciona como una evidencia académica del cumplimiento de los criterios a logro exigidos por la asignatura. En el se integran las practicas, metodologías y herramientas estudiadas durante el año.

Definicion Diagramas UML

El Lenguaje Unificado de Modelado (UML, por sus siglas en inglés: *Unified Modeling Language*) es un estándar ampliamente utilizado en el ámbito del análisis y diseño de sistemas de software. Este mismo fue creado con el objetivo de proporcionar un lenguaje gráfico común que permita a desarrolladores, analistas, arquitectos de software y demás interesados en un proyecto, representar de manera clara y precisa la estructura, el comportamiento y la interacción de un sistema.

Un diagrama UML es una representación visual que utiliza un conjunto de símbolos, reglas y convenciones gráficas para modelar los diferentes aspectos de un sistema. Su principal ventaja radica en que permite comprender y comunicar ideas complejas de forma sencilla, facilitando tanto la documentación como la comunicación entre los distintos miembros del equipo de desarrollo.

Tipos de diagramas UML

Hay distintos tipos de diagramas UML como lo son, los diagramas de caso de uso, de navegabilidad, de actividad y de secuencias.

Diagrama de Caso de Uso

Este diagrama es uno de los elementos más representativos de los diagramas UML, utilizado para describir de forma conceptual las interacciones que existen entre un sistema y los actores externos que se relacionan con él. Este tipo de diagrama no se centra en los aspectos técnicos de la implementación, sino en mostrar qué



BALATECH

3ºMA

ITI

funcionalidades ofrece el sistema y quiénes son los usuarios o sistemas que harán uso de ellas. Su importancia radica en que establece un puente entre los requerimientos planteados y el diseño posterior, asegurando que se comprendan y documenten las expectativas de los diferentes interesados.

El propósito principal del diagrama de caso de uso es definir con claridad el alcance del sistema. Al representar gráficamente los actores y los casos de uso, se obtiene una visión global de las funciones que el sistema debe cumplir, lo que facilita la comunicación entre el equipo de desarrollo y los usuarios. A diferencia de otros diagramas más técnicos, este puede ser comprendido fácilmente por personas sin formación especializada, como clientes o responsables funcionales, lo que lo convierte en una herramienta idónea para la validación de requerimientos. Así mismo, el diagrama contribuye a organizar el diseño del sistema, ya que sirve de base para la elaboración de otros modelos más detallados, como los diagramas de actividades, de secuencia o de clases, que se construyen a partir de la información inicial contenida en los casos de uso.

La incorporación del diagrama de caso de uso en este proyecto genera múltiples beneficios. En primer lugar, mejora la comunicación con los actores involucrados, ya que permite expresar de manera clara qué hará el sistema antes de comenzar su construcción. En segundo lugar, asegura la estandarización del análisis gracias a que forma parte de un lenguaje formal como UML, reconocido a nivel internacional. En tercer lugar, constituye un soporte fundamental para el diseño posterior, dado que proporciona una estructura sólida desde la cual se pueden desarrollar modelos más complejos. Finalmente, su utilización previene errores en etapas tempranas del ciclo de desarrollo, ya que facilita la detección de inconsistencias, funciones faltantes o redundancias antes de que el sistema avance hacia fases más costosas de implementación.

Diagrama de Navegabilidad

En este caso diagrama de navegabilidad es una herramienta gráfica utilizada para representar de manera clara la estructura de navegación de un sistema, una aplicación o un sitio web. Su finalidad principal es mostrar las posibles rutas que puede seguir el usuario al interactuar con la interfaz, lo que lo convierte en un recurso valioso para comprender cómo se organiza la información y cómo se desplazará el usuario a través de las distintas pantallas o secciones. A diferencia de

otros diagramas orientados a la lógica interna del sistema, el diagrama de navegabilidad se enfoca en la experiencia del usuario, priorizando la facilidad de acceso y el orden en la interacción.

El objetivo de este diagrama es doble. Por un lado, permite comprender la estructura de navegación del sistema antes de su implementación, lo cual favorece la detección temprana de problemas como rutas innecesarias, flujos demasiado largos o pantallas inaccesibles. Por otro lado, contribuye a mejorar la experiencia de usuario, ya que al diseñar un recorrido lógico y consistente se garantiza que las acciones puedan realizarse de manera intuitiva y eficiente. Asimismo, este tipo de diagrama es útil en proyectos colaborativos, pues facilita la comunicación entre diseñadores, desarrolladores y otros miembros del equipo, asegurando que todos compartan una visión común sobre el funcionamiento del sistema.

Los beneficios de utilizar un diagrama de navegabilidad en nuestro proyecto múltiples. Este mismo nos permite visualizar cómo se desplaza el usuario dentro de la aplicación, facilita la comunicación interdisciplinaria, mejora la arquitectura de la información y apoya en la planificación de pruebas de usabilidad. También contribuye a identificar rutas redundantes o inconsistentes, optimizando así la estructura de navegación y reduciendo la complejidad del sistema. Su utilización resulta especialmente recomendable en fases iniciales del proyecto, durante un rediseño o cuando los flujos de interacción son particularmente complejos.

Existen diferentes tipos de navegación que pueden representarse mediante este diagrama. La navegación lineal corresponde a flujos paso a paso, como en encuestas o tutoriales. La navegación jerárquica organiza el recorrido de lo general a lo específico, como ocurre en menús desplegables. La navegación en red o matricial conecta todas las páginas entre sí, otorgando gran libertad al usuario, mientras que la navegación mixta combina los tipos anteriores para adaptarse a contextos más complejos.

El uso adecuado de un diagrama de navegabilidad requiere atender a buenas prácticas de modelado, tales como mantener la claridad visual, evitar cruces innecesarios de flechas, alinear correctamente los elementos y utilizar colores o leyendas para distinguir distintos tipos de páginas. Del mismo modo, es necesario evitar malas prácticas como generar recorridos circulares sin sentido, crear rutas sin retorno, dejar pantallas sin acceso o incorporar niveles jerárquicos excesivos que dificulten la interacción.

Diagrama de Actividad

Este diagrama pertenece al Lenguaje Unificado de Modelado (UML) que se utiliza para describir el flujo de actividades o procesos dentro de un sistema. Su función principal es mostrar la secuencia de pasos que se realizan, las decisiones que pueden surgir durante la ejecución y las posibles bifurcaciones o concurrencias de tareas. A diferencia de otros diagramas que representan la estructura estática del sistema, el diagrama de actividad se centra en el aspecto dinámico, es decir, en cómo se desarrollan los procesos a lo largo del tiempo.

Este diagrama resulta especialmente útil para modelar procedimientos que siguen un flujo definido, como la ejecución de un caso de uso, un proceso de negocio o un algoritmo específico. Al representarse mediante nodos y transiciones, permite observar con claridad qué actividades se llevan a cabo, en qué orden ocurren y bajo qué condiciones se avanza de una actividad a otra. De este modo, facilita la comprensión de procesos complejos, ya que los descompone en pasos visuales simples y comprensibles.

El propósito del diagrama de actividad es doble. Por un lado, busca mejorar la comunicación entre los miembros del equipo de desarrollo y los usuarios, mostrando de manera accesible cómo funciona un proceso sin necesidad de utilizar un lenguaje técnico complejo. Por otro lado, sirve como base de análisis para detectar inefficiencias, omisiones o redundancias en los flujos de trabajo antes de implementar la solución en el sistema. Su naturaleza gráfica lo convierte en una herramienta poderosa para validar procesos y para discutir alternativas de diseño en fases tempranas del proyecto.

La utilización de este diagrama en un proyecto genera beneficios significativos. Permite visualizar el comportamiento del sistema de manera ordenada, lo que facilita la detección de errores en etapas iniciales, reduciendo así los costos de corrección en fases posteriores. Además, otorga una visión integral de los procesos, lo que contribuye a la estandarización y documentación del proyecto, asegurando que todas las partes interesadas comprendan el funcionamiento del sistema. También ayuda a la planificación, ya que al representar flujos alternativos o actividades que pueden ejecutarse en paralelo, permite anticipar escenarios y optimizar el diseño del sistema.



Diagrama de Secuencia

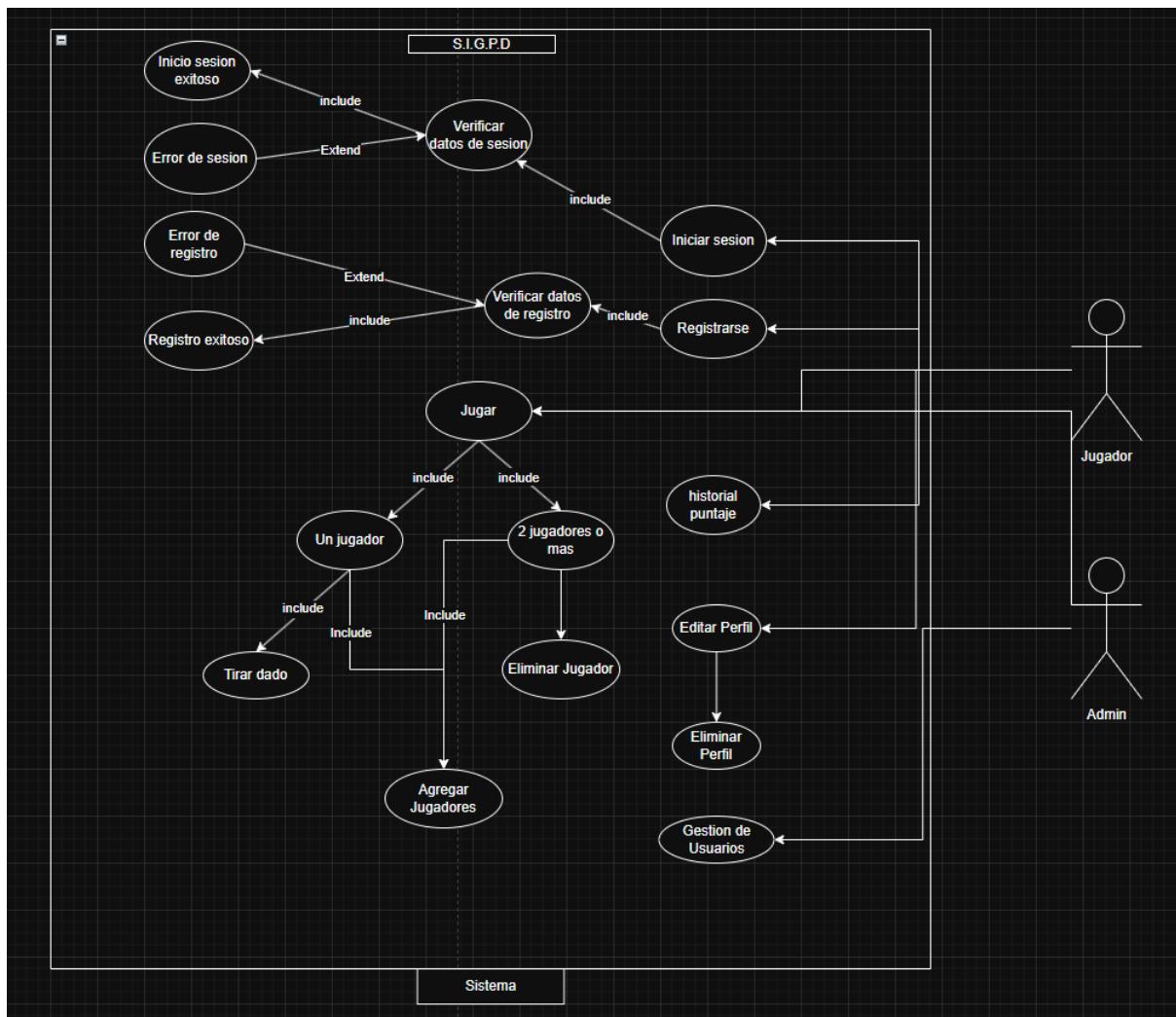
Este diagrama es un tipo de diagrama perteneciente al lenguaje de modelado unificado (UML) que se utiliza para representar la interacción dinámica entre los diferentes objetos o componentes de un sistema a lo largo del tiempo. Su estructura se basa en mostrar, de manera gráfica y ordenada, cómo se producen los mensajes y las llamadas entre los objetos participantes, señalando el orden en el que ocurren y el flujo de información que se intercambia. De esta forma, el diagrama de secuencia no solo describe los elementos involucrados en un proceso, sino que también ilustra de qué manera se comunican y colaboran entre sí para cumplir con una determinada funcionalidad del sistema.

La utilidad principal de este diagrama radica en que permite comprender con claridad la lógica de un caso de uso o de un escenario específico, ofreciendo una visión temporal y secuencial de la ejecución de las operaciones. Al mostrar de manera detallada cómo los objetos interactúan, se facilita la identificación de responsabilidades, dependencias y posibles problemas en la comunicación entre las partes del sistema. Asimismo, constituye una herramienta esencial para los desarrolladores y diseñadores de software, ya que les proporciona una representación visual del comportamiento esperado, lo cual facilita la validación de requerimientos y la detección temprana de inconsistencias.

El uso de diagramas de secuencia resulta beneficioso porque contribuye a una mayor claridad en el diseño y documentación del sistema, permitiendo tanto a los equipos de desarrollo como a los interesados externos comprender con facilidad cómo se llevará a cabo una funcionalidad concreta. Además, favorece la comunicación entre los distintos miembros de un proyecto, ya que estandariza la forma de expresar interacciones complejas mediante un lenguaje visual común. Desde el punto de vista académico y profesional, utilizar diagramas de secuencia no solo mejora la calidad del diseño, sino que también aporta rigor metodológico, reduce riesgos en la implementación y promueve un desarrollo de software más organizado y coherente con los objetivos planteados.

Diagramas

Diagrama de caso de Uso



Plantilla diagrama de caso de uso

Caso de base	Tipo de relación	Caso relacionado
Iniciar Sesión	Include	Verificar datos de sesión
Registrarse	include	Verificar datos de registro
Verificar datos de sesión	extend	Error de sesión / Inicio sesión exitoso

Plantilla diagrama de caso de uso

Caso de base	Tipo de relación	Caso relacionado
Verificar datos de registro	extend	Error de registro / Registro exitoso

Diagrama de Navegabilidad

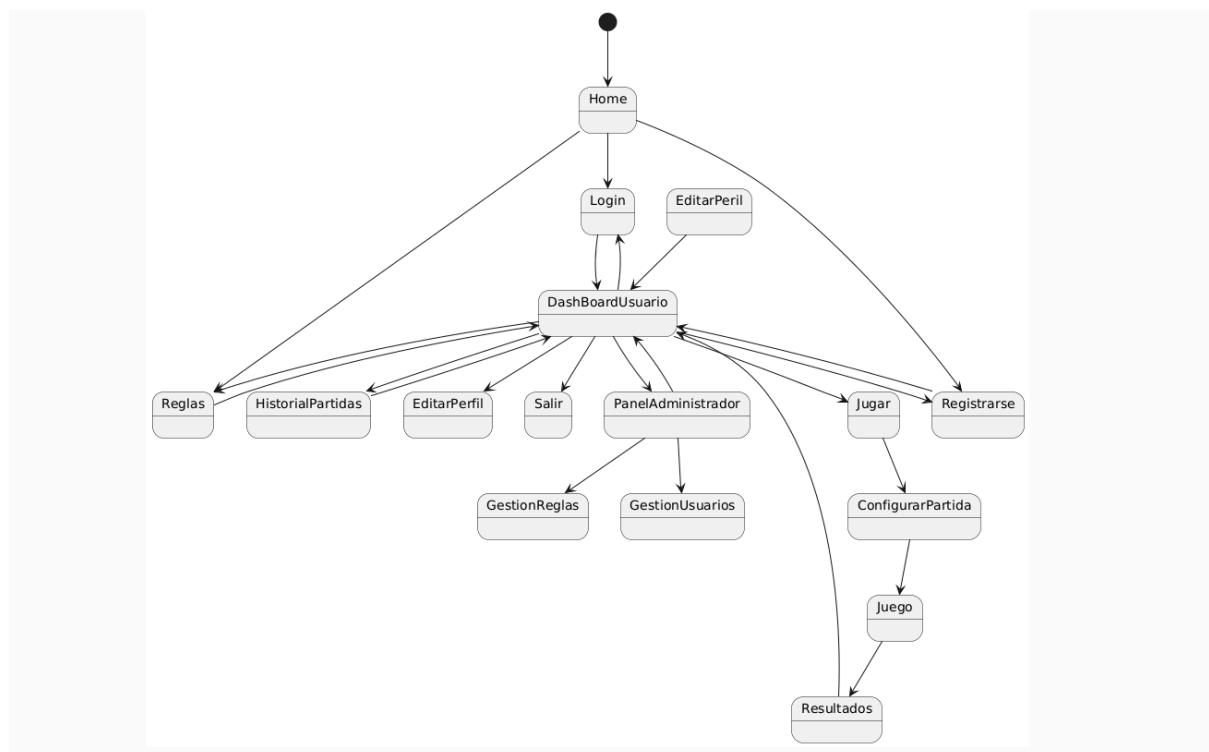


Diagrama de Clases

El diagrama de clases es una herramienta central dentro del Lenguaje Unificado de Modelado (*UML, Unified Modeling Language*), utilizada para representar la estructura de un sistema orientado a objetos. Este diagrama describe de forma visual las clases que forman parte del sistema, los atributos y métodos que las caracterizan, así como las relaciones que mantienen entre ellas, incluyendo la herencia, la asociación, la agregación o la composición. Gracias a esta representación, se obtiene una visión global y organizada de la arquitectura del software, lo que facilita la comprensión de su funcionamiento antes de proceder con la fase de implementación.

La incorporación de un diagrama de clases en el presente proyecto resulta particularmente valiosa, ya que permite estructurar de manera anticipada los elementos que lo conforman, asegurando que las funcionalidades y relaciones se definan con claridad antes de iniciar el desarrollo. Esto no solo ayuda a prever la lógica interna del sistema, sino que también proporciona una guía visual para todos los integrantes del equipo, de modo que se comparta una misma comprensión sobre los objetivos y la forma en que cada componente interactuará dentro del software.

Beneficios de implementar el mismo en S.I.G.P.D

Los beneficios de emplear un diagrama de clases en nuestro proyecto son múltiples. En primer lugar, ofrece claridad conceptual, puesto que facilita la identificación de las entidades principales y la forma en que se relacionan entre sí, evitando interpretaciones ambiguas. Además, contribuye a la prevención de errores, dado que permite detectar inconsistencias o redundancias en la fase de diseño, reduciendo así riesgos de fallos más adelante, cuando las correcciones serían más costosas. Otro beneficio importante es la comunicación efectiva, ya que el diagrama actúa como un lenguaje común entre analistas, programadores y demás participantes del proyecto, lo que garantiza que todos trabajen sobre una misma base de entendimiento.

Así mismo, este recurso promueve la reutilización de componentes, pues facilita la identificación de clases que pueden ser generalizadas y empleadas en diversas partes del sistema o incluso en proyectos futuros. También favorece una planificación estructurada, dado que permite definir de manera ordenada la lógica



3ºMA

ITI

interna del software, estableciendo una base sólida que orienta el desarrollo bajo principios de diseño orientado a objetos. Finalmente, el diagrama de clases cumple una función esencial como documentación formal del sistema, lo cual asegura que, una vez implementado, pueda mantenerse y ampliarse con mayor facilidad, sirviendo como referencia tanto en el presente como en etapas posteriores del ciclo de vida del proyecto.

En el caso específico de este proyecto, el diagrama de clases permitirá organizar de manera adecuada las entidades centrales que lo componen. Por ejemplo, se podrán representar claramente las clases principales junto con sus atributos y métodos correspondientes. Esto posibilitará una mejor definición de las relaciones jerárquicas y de dependencia entre los distintos componentes, garantizando un desarrollo coherente y bien estructurado.

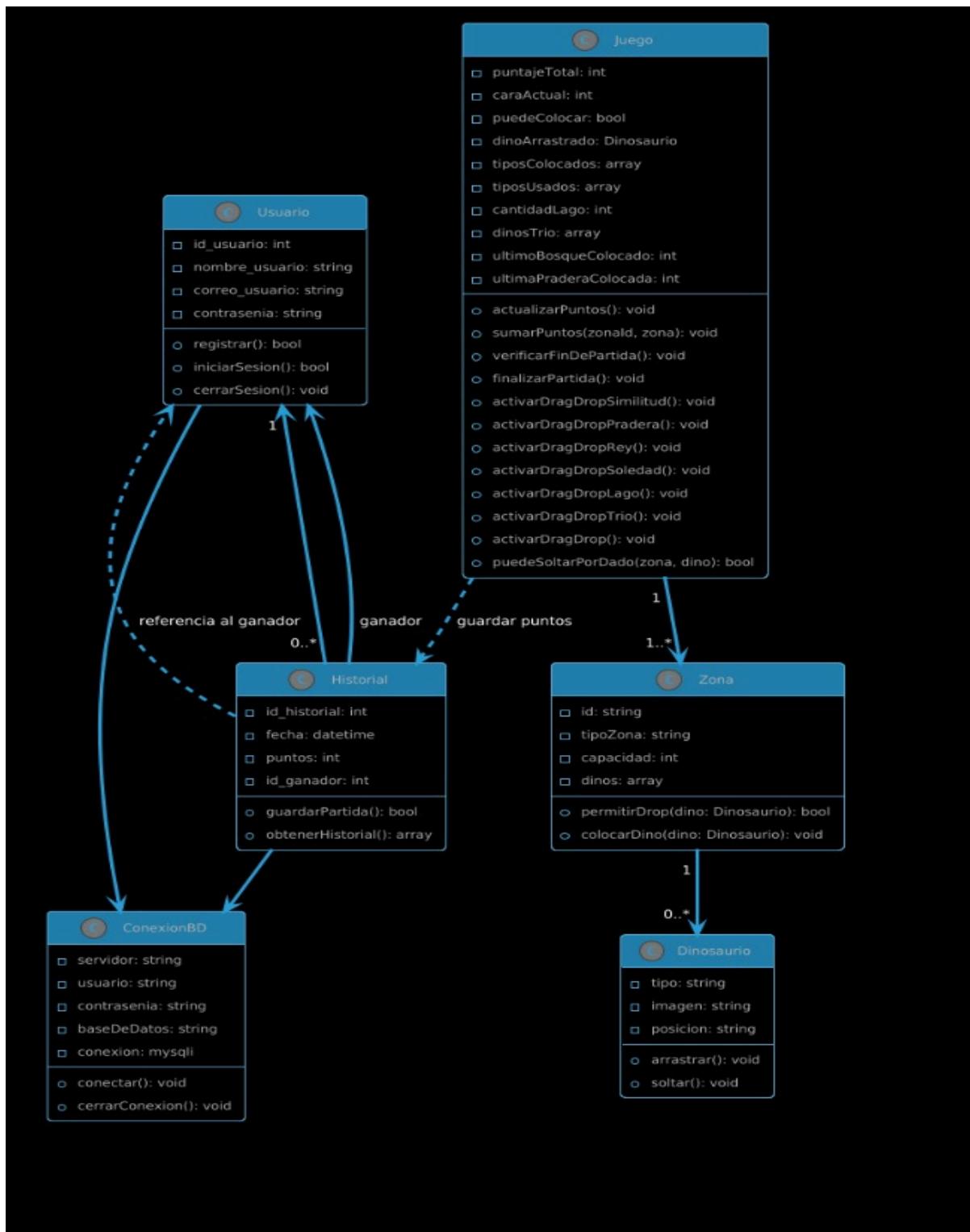
Diagrama de Clases UML



BALATECH

3ºMA

ITI





Definición gestión de riesgos

La gestión de riesgos es un conjunto de acciones y procedimientos que se diseñan con el objetivo de anticiparse a posibles problemas o imprevistos que puedan afectar el desarrollo o el funcionamiento de un proyecto. No se trata de suponer que las cosas van a salir mal, sino de reconocer que siempre existe la posibilidad de que aparezcan riesgos y por lo tanto, es mejor estar preparados con una respuesta clara y ordenada.

En el caso de nuestro proyecto, aplicar un plan de contingencia es especialmente útil porque el desarrollo involucra tanto software como infraestructura tecnológica que depende de distintos factores: la disponibilidad de los equipos personales e institucionales, la correcta configuración de entornos de desarrollo y servidores, la estabilidad del software libre que se utiliza y la organización del propio equipo. Un imprevisto en cualquiera de estas áreas, como la falla de un equipo, la pérdida de información, problemas en la configuración de la base de datos o incluso retrasos en las entregas por dificultades de coordinación, puede comprometer el avance del proyecto si no se tiene una respuesta planificada. Al implementar una gestión de riesgos, el equipo asegura una mayor capacidad de reacción frente a los riesgos. Por ejemplo, contar con copias de seguridad periódicas garantiza que, en caso de pérdida de datos, no sea necesario reiniciar todo el trabajo desde cero. Del mismo modo, prever equipos alternativos en caso de falla de un dispositivo personal, o definir cómo redistribuir tareas si un integrante no puede cumplir con su parte, son medidas que permiten mantener la continuidad del proyecto sin afectar de forma significativa el cronograma establecido.

El impacto de aplicar un plan de contingencia es doble. Por un lado, ofrece seguridad y tranquilidad al equipo, ya que se sabe de antemano cómo proceder si ocurre una situación inesperada. Esto evita improvisaciones y reduce el estrés en momentos críticos. Por otro lado, refuerza la confianza en la calidad del producto final, ya que los usuarios y la institución perciben que el sistema se desarrolló con criterios de seriedad y responsabilidad, contemplando no solo el funcionamiento normal sino también posibles escenarios adversos.

En definitiva, el gestionar los riesgos en nuestro proyecto nos sirve como una red de seguridad que protege el esfuerzo invertido y asegura que el S.I.G.P.D. pueda completarse y mantenerse en funcionamiento incluso frente a dificultades. Su impacto es positivo porque fortalece la fiabilidad del proyecto, mejora la gestión de

riesgos y contribuye a que los objetivos planteados se cumplan dentro de los plazos y condiciones previstas.

Enfoque de Gestión

La gestión de riesgos constituye un proceso sistemático orientado a identificar, evaluar y tratar aquellos eventos que puedan afectar negativamente el desarrollo de un proyecto. En el ámbito de la ingeniería de software y de los proyectos tecnológicos, existen diferentes enfoques para su aplicación, cada uno con características particulares.

Uno de los enfoques más comunes es el reactivo, el cual se centra en responder a los problemas una vez que estos se han materializado. Aunque resulta práctico en situaciones de bajo nivel de criticidad, presenta limitaciones importantes, ya que no permite anticiparse ni minimizar las consecuencias de los riesgos antes de que se manifiesten. En contraposición, el proactivo se orienta a la prevención. Este enfoque busca identificar de manera anticipada los posibles riesgos, evaluando su probabilidad de ocurrencia y su impacto, para posteriormente definir medidas de mitigación o planes de contingencia que reduzcan la posibilidad de que el riesgo se materialice o que, en caso de hacerlo, su impacto sea mínimo. Se considera el enfoque más recomendable en proyectos académicos y profesionales, pues promueve una gestión ordenada y reduce la incertidumbre.

Un tercer enfoque es el mixto o adaptativo, el cual combina la anticipación del enfoque proactivo con la capacidad de respuesta del reactivo. En este caso, se establecen planes de prevención y contingencia, pero al mismo tiempo se definen mecanismos de reacción rápida para aquellos riesgos que no hayan sido previstos en la fase inicial de planificación.

Para nuestro proyecto deliberamos que el enfoque más pertinente sería proactivo, complementado con elementos adaptativos. La naturaleza del proyecto, que involucra tanto la coordinación del equipo como la utilización de software libre y recursos tecnológicos compartidos, exige prever posibles escenarios de fallos técnicos, retrasos en la planificación o dificultades en la configuración de entornos. La aplicación de un enfoque proactivo permitirá identificar estos riesgos desde

etapas tempranas, establecer medidas preventivas y diseñar planes de contingencia específicos, garantizando así la continuidad del desarrollo.

El impacto de adoptar este enfoque en el proyecto se traduce en una mayor seguridad en la ejecución, una reducción de la incertidumbre y una mejora en la calidad final del producto, al integrar la gestión de riesgos como un componente esencial de la planificación y no como una reacción improvisada ante imprevistos.

Criterios para Clasificar

- 1) Probabilidad: Esto hace alusión a la posibilidad de que un riesgo pueda pasar dentro del proyecto S.I.G.P.D

Alta: 70% o mas

media: 50% hasta 70%

baja: 0% hasta 50%

- 2) Impacto: Esto muestra las consecuencias negativas que significa cada riesgo para el proyecto que se esta llevando a cabo

muy alto: Puede interferir completamente en el proyecto, dejarlo varado arreglando este inconveniente o dejarlo completamente inviable, esto tambien puede inferir en costos

alto: Puede generar grandes retrasos que compliquen el seguimiento del proyecto aunque estos son riesgos capaces de ser arreglados.

Medio: Estos son riesgos que no deberían de generar mucha complicación la hora de la realización el proyecto

bajo: impacta mínimamente con el proyecto suelen ser cosas pequeñas.

Riesgos encontrados



3ºMA

ITI

Riesgo: Perdida de datos de partida en curso

Descripción: El jugador esta teniendo una partida en el juego y cierra la pestaña o recarga por accidente la pagina, puede perder la configuración del juego y del puntaje.

Impacto: Muy alto

Probabilidad: 20%

Prevención:

- Utilizar un autoguardado de datos en la base datos cada vez que el usuario realice una acción importante como por ejemplo colocar un dinosaurio
- Recuperar automáticamente la sesión al volver a entrar

Acción de contingencia:

- Si se detecta que el usuario perdió la conexión, recargo la pestaña o se cerro se le mostrara un mensaje “desea continuar donde lo dejo?”
- Si no se pudo recuperar por fallo de guardado se le dará la opción de reiniciar desde la ultima configuración guardada

Responsable: Desarrollador Backend para la lógica del guardado y el Frontend para la interfaz de recuperación.

Riesgo: Caída de servidor en mitad de una partida

Descripción: El servidor donde se esta jugando la partida cae y los jugadores no podrán continuar con su partida ni podrán acceder al sistema

Impacto: Muy Alto

Probabilidad: 10%



3ºMA

ITI

Prevención:

- Usar hosting confiable
- Revisión 24/7 y respaldos automáticos de la base de datos

Acción de contingencia:

- Migrar temporalmente a un servidor y restaurar los datos de la ultima copia de seguridad

Responsable: Desarrollador Backend y Frontend

Riesgo: Error de validación de reglas

Impacto: Alto

Probabilidad: 20%

Prevención: Hacer prueba de integración para validar las reglas y revisar la lógica en caso de pruebas antes de liberar la versión

Acción de contingencia: Al detectar algún error, se registrara el bug y se aplicara un parche en el backend y se le notificara a los jugadores del problema y que se reiniciara la partida desde 0.

Responsable: Desarrollador Backend



3ºMA

ITI

Riesgo: Alta o baja de miembros en el equipo

Impacto: Alto

Posibilidad: 50%

Prevención: No existe una prevención para este riesgo, son cosas sin prevención ya que se puede dar por muchas características

Responsable: No hay ningún responsable

Acción de contingencia: No se encuentra ninguna acción de contingencia para algo que no podemos prevenir

Riesgo: Problema de compatibilidad entre navegadores

Impacto: Medio



3ºMA

ITI

Posibilidad: 30%

Prevención: Realizar prueba de compatibilidad en diferentes navegadores

Acciones de contingencia: Aplicar parches específicos para navegadores conflictivos

Responsable: Desarrollador Frontend

Riesgo: Brechas de seguridad en el sistema

Impacto: Muy Alto

Probabilidad: 5%

Prevención: Realizar pruebas de seguridad y aplicar protocolos de protección.

Acciones de contingencia: Aplicar parches de seguridad de forma inmediata

Responsable: Desarrollador Backend y Administrador de sistemas.

Conclusion Gestión de Riesgos

La gestión de riesgos en el proyecto S.I.G.P.D se consolida como un elemento fundamental para garantizar la continuidad, calidad y sostenibilidad del desarrollo. A lo largo del análisis realizado se ha demostrado que los riesgos, lejos de ser hechos aislados o imprevisibles, constituyen escenarios posibles que deben ser contemplados con antelación para reducir su impacto en los objetivos del proyecto.

El establecimiento de criterios claros de clasificación en términos de probabilidad e impacto permitió evaluar de manera ordenada cada situación potencial, diferenciando aquellas que podrían comprometer la viabilidad del sistema, como la



3ºMA

ITI

pérdida de datos de partidas en curso o la caída del servidor, de aquellas con repercusiones menores, como problemas de compatibilidad entre navegadores. Esta categorización, complementada con la identificación de responsables, medidas de prevención y planes de contingencia, posibilita una gestión estructurada y proactiva frente a los imprevistos.

Asimismo, la adopción de un enfoque proactivo, apoyado con elementos adaptativos, asegura que el equipo no solo se anticipa a los riesgos más probables y de mayor impacto, sino que también cuente con la capacidad de reacción ante eventualidades no previstas. De esta manera, el proyecto combina planificación anticipada con flexibilidad operativa, fortaleciendo su resiliencia y reduciendo la incertidumbre inherente al desarrollo de sistemas tecnológicos.

El impacto positivo de aplicar esta gestión radica en dos aspectos clave. Por un lado, fortalece la seguridad del equipo de trabajo, que dispone de un marco de acción claro para enfrentar cualquier situación crítica sin recurrir a improvisaciones. Por otro, refuerza la calidad y confiabilidad del producto final, ya que demuestra que el proyecto se desarrolló considerando no solo su funcionamiento óptimo, sino también los escenarios adversos que podrían afectar su ejecución.

Definición Análisis Costo-Beneficio

El análisis de costo-beneficio constituye una metodología fundamental dentro de la evaluación y gestión de proyectos, cuya finalidad es determinar la viabilidad y conveniencia de una iniciativa a partir de la relación existente entre los recursos invertidos y los beneficios generados. Este procedimiento no se limita a una

dimensión exclusivamente económica, sino que incorpora factores académicos, técnicos, sociales y profesionales que permiten una visión integral del valor aportado por el proyecto.

En el caso del presente trabajo el análisis de costo-beneficio adquiere una especial relevancia, dado que, si bien no requiere de una inversión monetaria significativa, sí demanda recursos críticos tales como tiempo, esfuerzo intelectual, competencias técnicas y soporte tecnológico. Por lo tanto, resulta pertinente examinar cada una de las etapas del análisis a fin de evidenciar la justificación y sostenibilidad del proyecto.

La primera etapa corresponde a la identificación del proyecto, donde se delimitan los objetivos, el alcance y las metas a alcanzar. En este contexto, la propuesta consiste en diseñar e implementar una aplicación web que permita trasladar la experiencia lúdica de *Draftosaurus* al entorno digital, facilitando la gestión y ejecución de partidas en línea. Esta definición inicial constituye la base para la estimación posterior de costos y beneficios.

La segunda etapa se centra en la identificación de costos y beneficios. Entre los costos más relevantes se incluyen la inversión de tiempo destinada al diseño, la programación y la verificación de la aplicación, así como los recursos tecnológicos asociados al desarrollo, tales como equipos de cómputo, software y posibles servicios de alojamiento web. A estos se añaden los costos de capacitación derivados de la adquisición de nuevas competencias técnicas necesarias para la correcta implementación del sistema, así como los costos de mantenimiento y soporte posteriores a su despliegue. Frente a estos costos, se reconocen beneficios de distinta índole: en el ámbito académico, el refuerzo de los conocimientos adquiridos durante la formación universitaria; en el plano práctico, la creación de una herramienta funcional de entretenimiento digital; en el ámbito profesional, la experiencia adquirida en el desarrollo de un sistema web complejo que favorece la empleabilidad y la proyección laboral; y finalmente, en el aspecto social y recreativo, la posibilidad de ofrecer a los usuarios un producto accesible y novedoso.

La tercera etapa es la valoración de costos y beneficios, en la cual se pondera la magnitud relativa de cada uno de los elementos identificados. Aunque los costos asociados al tiempo de dedicación y al esfuerzo intelectual son significativos, su

3ºMA

ITI

impacto resulta inferior frente a los beneficios alcanzados, que se manifiestan de manera más amplia y sostenible en el tiempo, especialmente en lo referente a la formación académica y la proyección profesional.

La siguiente fase corresponde a la comparación entre costos y beneficios, cuyo propósito es establecer si el proyecto es factible y recomendable. Del análisis realizado se desprende que los beneficios superan ampliamente a los costos, ya que la inversión de recursos se traduce en resultados tangibles e intangibles de gran valor, consolidando al proyecto como una propuesta viable y pertinente.

Finalmente, la toma de decisiones confirma la conveniencia de llevar adelante el proyecto. Los beneficios derivados de su ejecución, tanto en el ámbito educativo como en el profesional y recreativo, justifican plenamente la dedicación de recursos y el esfuerzo invertido en su desarrollo.

La inclusión de este análisis en el informe académico reviste una importancia metodológica considerable, ya que dota al proyecto de un sustento técnico que avala su pertinencia y viabilidad. En ausencia de este apartado, el trabajo podría percibirse como incompleto, carente de planificación y de fundamentos objetivos para su ejecución. Por el contrario, su incorporación demuestra que el proyecto se encuentra debidamente evaluado y que los beneficios obtenidos compensan con creces los recursos comprometidos.

En conclusión, el análisis de costo-beneficio aplicado al desarrollo de la aplicación web inspirada en *Draftosaurus* permite afirmar que la propuesta es factible y altamente recomendable. Si bien existen costos asociados a la inversión de tiempo, capacitación y mantenimiento, estos resultan menores en comparación con los beneficios académicos, prácticos, profesionales y sociales. El proyecto, por lo tanto, no solo cumple con los objetivos planteados, sino que además genera un impacto positivo que trasciende el plano educativo, aportando valor al campo del desarrollo tecnológico y de las aplicaciones interactivas.

Costos Iniciales

Los costos Iniciales serían las inversiones necesarias para llevar a cabo la fase de diseño y desarrollo.



3ºMA

ITI

Licencias: El proyecto utilizará un stack compuesto por JavaScript, PHP, MySQL, HTML y CSS. Todas estas tecnologías son de código abierto y de uso libre, por lo que no implican costos asociados en concepto de licencias. De igual forma, se emplearán herramientas auxiliares como Visual Studio Code, XAMPP y GitHub, que cuentan con versiones gratuitas y plenamente funcionales para los objetivos del proyecto, todo lo nombrado anteriormente es de carácter gratuito así que no necesitaremos invertir en demás aplicaciones, editores de código, etc.

Hardware: Para realizar este proyecto utilizaremos los equipos disponibles de cada integrante del equipo por lo tanto no hará falta invertir en equipos de hardware.

Instalación: Para la instalación contemplaremos los costos en base al tiempo de instalación y configuración de servidores creados en XAMPP, aunque estos no tendrán costo, si se toma en cuenta el tiempo de trabajo y se generara un presupuesto en base a la carga horaria.

Costos Operativos y de Mantenimiento

Operativos:

Equipos de la institución y personales: se utilizarán de manera conjunta para el desarrollo, pruebas y ejecución del sistema. Al ser recursos ya disponibles, no generan costos adicionales de operación. Energía eléctrica y conectividad: se consideran parte de los recursos provistos por la institución durante las horas de trabajo en laboratorio. En el caso de los equipos personales, estos gastos son absorbidos por cada integrante sin representar un costo específico para el proyecto.

Herramientas de software: todas las herramientas de desarrollo y gestión seleccionadas (Visual Studio Code, XAMPP, VirtualBox/VMware, GitHub) cuentan con versiones gratuitas, por lo que no existen costos operativos por licencias o suscripciones.



3ºMA

ITI

Mantenimiento:

Actualizaciones de software: tanto PHP, MySQL/MariaDB, como el resto de los componentes del stack son de código abierto y gratuitos, por lo que el mantenimiento implica únicamente la aplicación de actualizaciones disponibles sin costo.

Mantenimiento de hardware: al emplear equipos de la institución y personales, la responsabilidad de mantenimiento se reparte entre ambas partes. La institución se encarga de la infraestructura de laboratorio, mientras que cada integrante asume el cuidado y mantenimiento básico de sus equipos personales.

Soporte técnico y configuración: las tareas de mantenimiento preventivo y correctivo del entorno de desarrollo (configuración de servidor local, entornos virtualizados, respaldo de la base de datos) serán gestionadas por el equipo de proyecto, sin costos adicionales.

Para todo esto tendremos en cuenta únicamente la carga horaria ejercida para realizar el mantenimiento, actualizaciones de software, etc. En base a eso se le realizara un presupuesto

Cuantificación Costo y Beneficio

En el marco de la metodología de análisis de costo-beneficio, los beneficios que se derivan de la ejecución de un proyecto se clasifican en dos categorías fundamentales: beneficios tangibles y beneficios intangibles. Esta distinción resulta indispensable para una valoración integral, puesto que no todos los efectos positivos de un proyecto pueden expresarse en términos cuantitativos o económicos, aunque todos ellos aportan valor a la iniciativa.

Beneficios intangibles:

- Mejora en la experiencia del usuario: La aplicación facilita el conteo de puntos y la validación de reglas en Draftosaurus, reduciendo errores y aumentando la fluidez de las partidas, lo que se traduce en una experiencia de juego más agradable.
- Mejora en la accesibilidad del usuario: Al implementar un diseño responsive a la página en usuario podrá hacer uso de ella desde cualquier dispositivo que tenga, como móviles, tablets y computadoras. Mejorando la experiencia del usuario.
- Aporte educativo y de aprendizaje: El sistema no solo ayuda a los jugadores a comprender mejor las reglas, sino que también permite a los integrantes del proyecto aplicar conocimientos adquiridos en todas las materias requeridas para este proyecto, reforzando competencias profesionales.
- Accesibilidad y disponibilidad: Al tratarse de una aplicación web, el sistema es fácilmente accesible desde diferentes dispositivos, lo que amplía su alcance sin necesidad de infraestructura compleja.

Beneficios Tangibles:

- Reducción de errores en el conteo de puntos: El sistema automatiza los cálculos de puntuación, lo que elimina inconsistencias y posibles discusiones entre jugadores.
- Ahorro de tiempo durante las partidas: Al digitalizar el conteo y validación de reglas, se acorta el tiempo total de juego, permitiendo más partidas en menos tiempo.
- Escalabilidad del sistema: La estructura tecnológica utilizada permite que en el futuro se pueda ampliar el sistema a más funcionalidades (ejemplo: modalidad multijugador en línea) sin requerir inversiones iniciales adicionales.

Viendo estos beneficios demostramos que el proyecto no solo cumple con los requisitos planteados sino que también aporta un valor agregado tanto a los usuarios finales como al proceso formativo de los integrantes. Esto lo convierte en una propuesta viable, innovadora y con potencial de crecimiento futuro.



3ºMA

ITI

Análisis Costo-Beneficio

Concepto	Costo(USD)	Beneficio(USD)	Descripción
Licencias y Suscripciones	0USD	300USD	No se requiere el uso de suscripciones ya que utilizaremos programas de carácter libre sin costo
Soporte y Mantenimiento	5USD-20USD	0USD	Costos en base a horas invertidas en la corrección de errores u actualizaciones del sistema
Ahorro de tiempo	0USD	200USD	Automatización de tiempo y validación de reglas automáticamente para los jugadores
Hardware	100USD	0USD	por alguna falla en algún equipo que se necesite reponer o por mayor capacidad de servidor
Satisfacción del usuario	0USD	75USD	Incorporación diseño Responsivo
Escalabilidad del sistema	70USD	200USD	Si se quieren incorporar nuevas opciones como multijugador u otros
Accesibilidad del sistema	0USD	100USD	Es posible acceder desde cualquier dispositivo

Análisis Costo-Beneficio

Concepto	Costo(USD)	Beneficio(USD)	Descripción
Capacitación del equipo	250USD	0USD	Horas empeñadas en adquirir mas conocimientos en distintos campos y tambien cursos
Total	440USD	875USD	425USD

Total = 440 + 875 = 1315

ROI=400%435x100

ROI=98,86%

Por cada 1 dolar invertido se recuperaran su inversion mas un 98%

Definición Calculo de Métricas:

El cálculo de métricas es un proceso analítico mediante el cual se cuantifican y evalúan distintos aspectos del rendimiento de un sistema, con el fin de obtener información objetiva y verificable acerca de su funcionamiento. En el ámbito del desarrollo de software, las métricas se conocen como indicadores que permiten medir variables clave, tales como la eficiencia del código, la usabilidad de la aplicación, la estabilidad del sistema y la satisfacción del usuario. Estas mediciones no se limitan únicamente a la dimensión técnica, sino que también abarcan aspectos relacionados con la experiencia de los jugadores, la escalabilidad de la plataforma y la calidad de los procesos de gestión.

La utilidad principal del cálculo de métricas radica en que proporciona una base empírica para la toma de decisiones. En lugar de depender de percepciones subjetivas o de estimaciones aproximadas, el uso de métricas otorga la posibilidad de contar con información cuantitativa que permite identificar fortalezas y



3ºMA

ITI

debilidades dentro del proyecto. Esto se traduce en una mayor precisión a la hora de implementar mejoras, corregir fallos o rediseñar elementos que no cumplen con los objetivos planteados. Además, la comparación de métricas en distintos períodos posibilita evaluar la evolución del sistema, lo que contribuye a establecer tendencias y proyectar escenarios futuros de manera fundamentada.

Beneficios de la aplicación de esta área a S.I.G.P.D

Aplicado al proyecto que estamos realizando (S.I.G.P.D), el cálculo de métricas se vuelve particularmente beneficioso en la medida en que permite evaluar tanto la dimensión técnica del desarrollo como la experiencia de los usuarios que interactúan con la aplicación. Medir el tiempo de respuesta de las partidas, la estabilidad de la conexión, la fluidez de la interfaz y la frecuencia de uso por parte de los jugadores otorga una visión clara sobre el grado de satisfacción del público objetivo. Así mismo, estas métricas ofrecen un sustento sólido para justificar mejoras funcionales, optimizar recursos y asegurar que la plataforma cumpla con estándares de calidad. En este sentido, su incorporación no solo fortalece la gestión interna del proyecto, sino que también aumenta la probabilidad de consolidar un producto competitivo, atractivo y sostenible en el tiempo.



Cálculo de Métricas del Proyecto

Componentes del Sistema

Componente	Tipo	Complejidad	Puntos de Función (PF)
Registro	Entrada	Media	4
Login	Entrada	Baja	3
Base de datos	ILF	Media	7
Total PFSA	—	—	14

Factor de Ajuste (FA)

Característica	Valor (0-5)
Copias de seguridad	3
Interacción con otros sistemas	1
Distribución en varias computadoras	3
Rendimiento	3
Uso en entornos con carga alta	1
Método de ingreso de datos	3
Complejidad de consultas	1
Modificación de datos por usuario	3
Complejidad de procesos internos	3
Reusabilidad	2
Conversión e instalación	2
Adaptabilidad	3
Facilidad de cambio	3
Facilidad de uso	3
Suma FA	33

Cálculo de Function Points Ajustados (PFA)

PFSA	FA	PFA
14	33	13.7

Tercera Entrega Ingenieria

Foda Ponderado

Definición Foda Ponderado: El FODA ponderado es una metodología que nos permite evaluar de forma estructurada los factores externos e internos que influyen en el desarrollo de un sistema, este otorga además de un valor cuantitativo a cada uno de ellos para determinar su nivel de incidencia en el proyecto.

A diferencia del FODA tradicional, el FODA Ponderado no se limita únicamente a identificar las Fortalezas, Oportunidades, Debilidades y Amenazas, sino que introduce un enfoque mas analítico mediante la asignación de ponderaciones o valores numéricos. Estas ponderaciones reflejan el grado de importancia o el impacto que cada factor tiene sobre los objetivos del proyecto, permitiendo así priorizar las estrategias de acción y orientar la toma de decisiones en función de criterios objetivos y medibles.

En el contexto de nuestro proyecto, el FODA ponderado resulta particularmente útil para analizar el entorno técnico, organizacional y operativo. A través de este instrumento, es posible identificar las fortalezas internas del equipo de desarrollo, como el dominio de lenguajes de programación, el uso de metodologías ágiles o la correcta gestión de versiones, al mismo tiempo que se reconocen las debilidades que podrían afectar el rendimiento del grupo, como la falta de experiencia en determinadas tecnologías o limitaciones en los recursos disponibles.

Por otra parte, el análisis de las oportunidades permite detectar factores externos favorables, tales como la disponibilidad de herramientas de código abierto, la existencia de comunidades de soporte o el crecimiento de la demanda de soluciones digitales similares. En contrapartida, el estudio de las amenazas contribuye a prever posibles riesgos, como la aparición de competidores, cambios tecnológicos acelerados o problemas de compatibilidad con entornos de producción.

Fortalezas

Factor	P	C	VP = P x C
Conocimiento entero del equipo en lenguajes como html ,css, javascript, php	0,5	4	2
Utilizamos herramientas gratuitas(XAMPP, VS code, GitHub) y propios equipos	0,3	3	0,9
Creamos diseños intuitivos y amigables para el usuario	0,1	2	0,2
Buena capacidad para amoldarnos a otras herramientas o formas de trabajo	0,1	2	0,2

Valor Ponderado total fortalezas: 3,3

Oportunidades

Factor	P	C	VP = P x C
Posibilidad de escalar modos de juego multijugador y en linea	0,2	4	0,8
Adaptar el juego a otro idioma (Ingles)	0,3	4	1,2
Posibilidades de incluir Inteligencia artificial a nuestro juego	0,3	3	0,9
Apoyo de gente con experiencia como docentes	0,2	2	0,4

Valor Ponderado total Oportunidades: 3,3



BALATECH

3ºMA

ITI

Debilidades

Factor	P	C	VP = P x C
Mala coordinación entre integrantes	0,5	1	0,5
Trabajo en equipo no tan bueno	0,2	2	0,4
Falta de experiencia del equipo	0,2	3	0,6
Poca experiencia en metodologías ágiles de desarrollo como lo es Scrum	0,1	3	0,3

Valor Ponderado total Debilidades: 1,8

Amenazas

Factor	P	C	VP = P x C
Rotura de equipo sin repuesto	0,2	2	0,4
Cambios de requerimientos	0,2	2	0,4
Perdidas de datos	0,3	3	0,9
Riesgo de vulnerabilidades en seguridad	0,3	3	0,9

Valor Ponderado total Amenazas: 2,6

Plan de Testing:

Este plan de testing tiene como objetivo garantizar la calidad funcional, técnica y operativa de nuestro proyecto, desarrollado por nuestra empresa Balatech.

En el proceso de plan de testing podremos identificar errores y validar el correcto funcionamiento de todas las funcionalidades del sistema.

Objetivos del Plan de Testing:

- Verificar que el sistema cumpla con todos los requerimientos funcionales y no funcionales.
- Asegurar la estabilidad y el rendimiento del sistema en diferentes entornos
- Detectar errores de lógica, interfaz de usuario, base de datos, etc
- Garantizar una experiencia satisfactoria para el usuario

Alcance:

El plan de testing abarcara los siguientes puntos:

- Gestión de usuarios: Registro, inicio de sesión, modificación de perfil, mantenimiento de sesiones activas.
- Gestión de partidas: Creación, seguimiento, puntuación automática, cierre de partidas, historial de partidas.
- Validación de reglas: comprobar las condiciones pautadas por el dado y la colocación de dinosaurios de forma correcta.
- Interfaz gráfica: Funcionalidad y adaptabilidad del tablero y de los elementos visuales
- Base de datos: Integridad de datos y conexión segura

3ºMA

ITI

-Seguridad y validación: Control de formularios, manejo de errores y acceso restringido

Tipos de Prueba	
Tipo de prueba	Descripción
Pruebas funcionales	Verifica que cada modulo contengan los requerimientos definidos (ej: Validar puntos, validar usuario, conteo de puntos, etc.)
Pruebas de integración	Evaluan la comunicación entre componentes PHP -> Base de datos -> interfaz
Pruebas de interfaz	Analizan la usabilidad y la respuesta del sistema ante acciones del usuario
Pruebas de validación	Confirma que las reglas del juego se apliquen correctamente (ej: restricción de datos, validación de recintos)
Pruebas de rendimiento	Miden el tiempo de estabilidad y del sistema ante diferentes cargas de uso
Pruebas de seguridad	Comprueban el manejo de errores, validación de formularios y prevención de inyecciones SQL o accesos indebidos
Pruebas de compatibilidad	Evaluan el correcto funcionamiento del sistema en distintos navegadores y resoluciones de pantalla



3ºMA

ITI

Casos de prueba

ID	Caso de prueba	Descripción	Entrada	Resultado esperado
CP-01	Registro de usuario	Verificar que el sistema permita registrar el nuevo usuario correctamente	Formulario completo(mail, nombre, contraseña).	Usuario creado y mensaje de confirmación
CP-02	Inicio de sesión	Validar que los datos sean correctos para entrar al sistema	Usuario y contraseña válidos	Acceso concedido y redirección al menu
CP-03	Carga de Partida	Verificar que el sistema permita crear una nueva partida sin ningún problema	Datos de jugadores y modo de juego	Partida creada con tablero, dinosaurios, dado visible
CP-04	Colocación de dinosaurios	Confirmar que las colocaciones correspondan a la restricción del dado	Tipo de dinosaurio + recinto + restricción	El sistema acepta o rechaza la colocación según las reglas
CP-05	Calculo de puntuación	Comprobar que la puntuación final se calcule correctamente	Datos de partida finalizada	Resultado final mostrado con puntaje correcto
CP-06	Agrega partida al historial	Comprobar que los datos de la partida se hayan guardado correctamente en el historial	Datos de partida finalizada	Ganador + puntos + fecha y hora aparecen en el historial
CP-07	Validación de formularios	Probar el manejo de errores de entrada	Campos vacíos o incorrectos	Mensaje de error mostrado al usuario
CP-08	Prueba de seguridad SQL	Intentar ingresar códigos maliciosos o directamente a la BD	Entrada con caracteres SQL	Sistema rechaza la entrada y no compromete datos
CP-09	Compatibilidad	Probar interfaz en distintos navegadores	Abrir sistema en Chrome, etc.	Visualización y funcionamiento



3ºMA

ITI

Casos de prueba

ID	Caso de prueba	Descripción	Entrada	Resultado esperado
				correctos

Conclusión:

En conclusión este plan de testing permitirá asegurar la calidad de todo nuestro sistema antes de su entrega final, reduciendo el riesgo de errores en producción y garantizando que el producto cumpla con los estándares esperados por el cuerpo docente y el cliente final.

Pruebas Funcionales:

Las pruebas funcionales son un conjunto de verificaciones destinadas a comprobar que el software desarrollado cumple correctamente con los requisitos establecidos. Su propósito principal es garantizar que cada función del sistema opere de acuerdo con lo esperado, respondiendo adecuadamente a las acciones que realiza el usuario.

En el presente proyecto se aplicaron pruebas funcionales para confirmar que las operaciones principales, funciones sin errores y cumplan los objetivos pactados



3ºMA

ITI

Manual de Usuario

Introducción:

El presente manual de usuario tiene como objetivo guiar al usuario final en la correcta utilización del proyecto S.I.G.P.D desarrollado por Balatech.

Este sistema permite gestionar, registrar y administrar partidas partidas del juego Draftosaurus de forma digital, simplificando el control de puntuaciones, la validación de reglas y el seguimiento de jugadores en tiempo real, también permite el registro de usuario, login y modificación del mismo.

Requisitos de Software:

- Sistema operativo: windows 10 o superior
- Servidor local: Xampp o equivalente (Apache, PHP, MySql)
- Navegador compatible: Chrome, Mozilla Firefox o Microsoft Edge

Requisitos Hardware:



3ºMA

ITI

Acceso al Sistema:

1. Iniciar el servidor local (XAMPP) y asegurarse de que Apache y MySQL estén en ejecución
2. Abrir navegador y ingresar la siguiente dirección

<http://localhost/>(Ruta del archivo)

Registro de Usuario:

1. En la pantalla de inicio seleccionar el botón “registrarse”
2. Ingresar los campos obligatorios:
 - Nombre
 - Correo
 - Contraseña
3. Clickear botón Enviar.
4. El sistema mostrará un mensaje de que el registro fue válido

Empezar una partida:

1. Desde la pantalla de inicio clickear el botón Jugar
2. Elegir modo de juego
3. Clickear empezar partida
4. El sistema generara el tablero, dado y dinosaurios disponibles para cada jugador

Colocar un dinosaurio:

3ºMA

ITI

1. Clickear botón “dinosaurios”
2. Elegir el tipo de dinosaurio
3. Arrastrarlo
4. El sistema validará automáticamente las reglas del juego:
 - Si el movimiento es válido entonces colocara el dinosaurio en el recinto
 - Si no es válido entonces se mostrar un mensaje que diga que no es válido el movimiento

Consultar Historial de partidas:

1. En la pantalla principal clickear el botón “Historial de partidas”
2. El sistema mostrará una lista de partidas jugadas por el usuario logueado
3. El usuario podrá eliminar las partidas que desee

Soluciones a problemas comunes		
Problema	Causa Posible	Solución recomendada
No se puede acceder al sistema	Servidor Apache o MySql no iniciado	Verificar que XAMPP este en ejecución
Error al registrar usuario	Campos vacíos o formato invalido	Revisar los datos ingresados y volver a intentar
Dinosaurio no se coloca	Esta clickeando el dinosaurio y luego el recinto	Arrastrar el dinosaurio al recinto elegido en base a la restricción

Recomendaciones Generales:

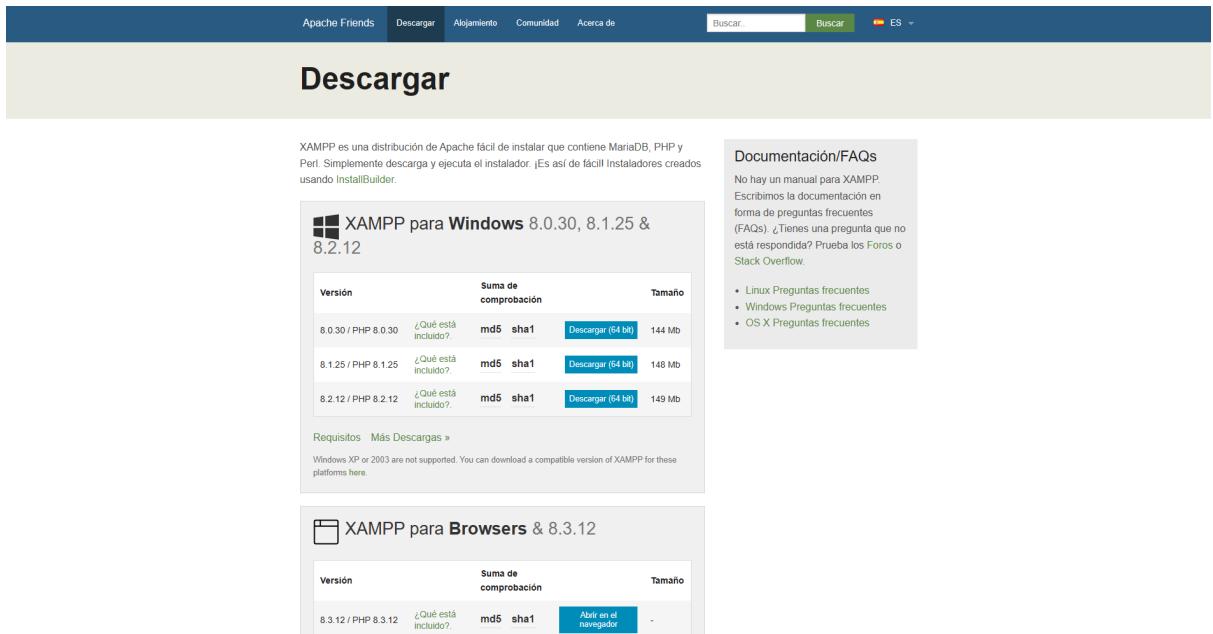
1. No cerrar el navegador mientras una partida este en curso
2. Evitar abrir múltiples pestañas del sistema simultáneamente

Conclusión:

El proyecto S.I.G.P.D ha sido diseñado para ofrecer una experiencia de gestión de partidas práctica, visual y confiable. Este manual proporciona las instrucciones necesarias para el uso correcto del sistema, garantizando que tanto usuarios nuevos como experimentados puedan aprovechar todas sus funcionalidades.

Manual de Instalación:

1. Instalar el programa XAMPP desde su página oficial:



The screenshot shows the Apache Friends website with a dark blue header bar containing links for Apache Friends, Descargar, Alojamiento, Comunidad, Acerca de, and search fields. Below the header, a large button labeled "Descargar" is prominently displayed. To its left, a text block explains what XAMPP is: "XAMPP es una distribución de Apache fácil de instalar que contiene MariaDB, PHP y Perl. Simplemente descarga y ejecuta el instalador. ¡Es así de fácil! Instaladores creados usando InstallBuilder." On the right, a sidebar titled "Documentación/FAQs" provides links to frequently asked questions for Linux, Windows, and OS X.

XAMPP para Windows 8.0.30, 8.1.25 & 8.2.12

Versión	Suma de comprobación	Tamaño
8.0.30 / PHP 8.0.30	md5 sha1	Descargar (64 bit) 144 Mb
8.1.25 / PHP 8.1.25	md5 sha1	Descargar (64 bit) 148 Mb
8.2.12 / PHP 8.2.12	md5 sha1	Descargar (64 bit) 149 Mb

[Requisitos](#) [Más Descargas »](#)

Windows XP or 2003 are not supported. You can download a compatible version of XAMPP for these platforms [here](#).

XAMPP para Browsers & 8.3.12

Versión	Suma de comprobación	Tamaño
8.3.12 / PHP 8.3.12	md5 sha1	Abrir en el navegador -



3ºMA

ITI

- Una vez dentro de la carpeta "htdocs" de XAMPP, clonar el repositorio con el comando "git clone"

<https://github.com/balatechING/ProgramacionFullStack.git>

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.26200.6899]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\matip>cd ..

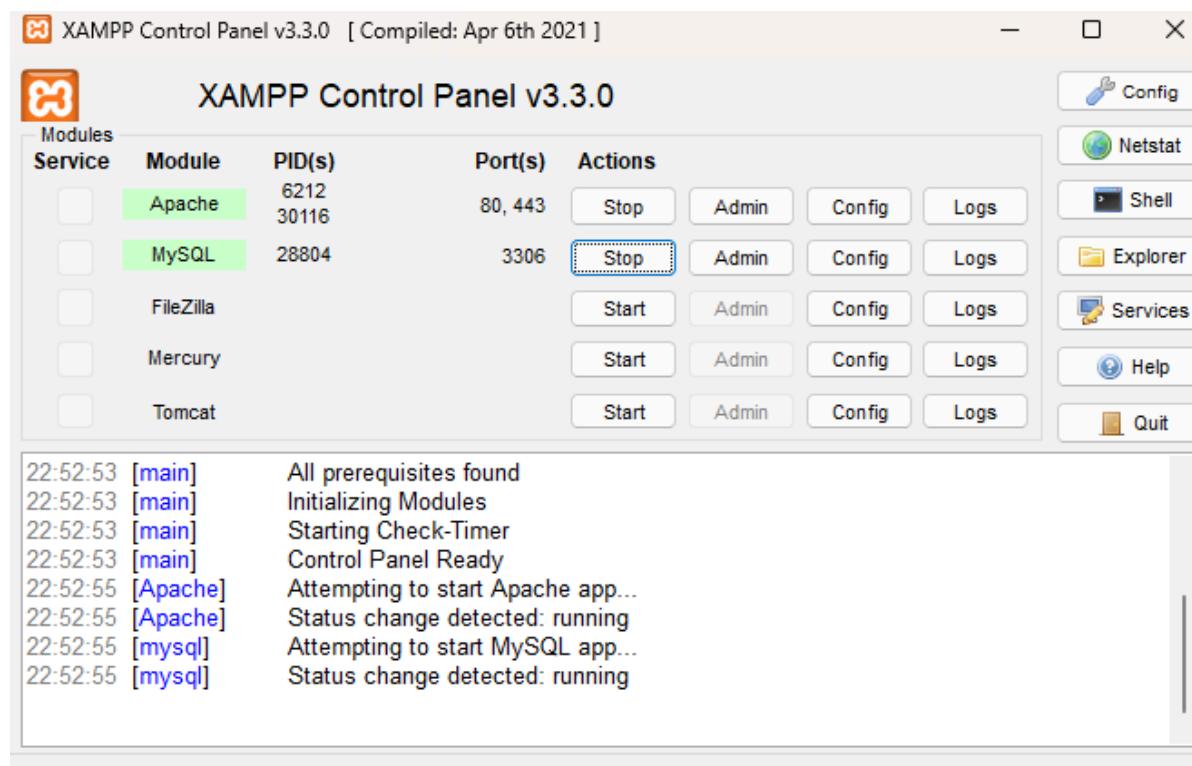
C:\Users>cd ..

C:\>cd xampp

C:\xampp>cd htdocs

C:\xampp\htdocs>git clone https://github.com/balatechING/ProgramacionFullStack.git
```

- Ya con todos los archivos del sistema descargados, iniciaremos Apache y MySQL desde XAMPP o la herramienta a preferencia del usuario





3ºMA

ITI

4. Acceder a phpMyAdmin presionando el botón llamado “Admin” que corresponde a MySQL dentro de XAMPP

XAMPP Control Panel v3.3.0 [Compiled: Apr 6th 2021]

Service	Module	PID(s)	Port(s)	Actions
	Apache	6212 30116	80, 443	Stop Admin Config Logs
	MySQL	28804	3306	Stop Admin Config Logs
	FileZilla			Start Admin Config Logs
	Mercury			Start Admin Config Logs
	Tomcat			Start Admin Config Logs

22:52:53 [main] All prerequisites found
22:52:53 [main] Initializing Modules
22:52:53 [main] Starting Check-Timer
22:52:53 [main] Control Panel Ready
22:52:55 [Apache] Attempting to start Apache app...
22:52:55 [Apache] Status change detected: running
22:52:55 [mysql] Attempting to start MySQL app...
22:52:55 [mysql] Status change detected: running

Admin

5. Hacer click donde dice “nueva” en la columna de la parte lateral izquierda para poder crear una nueva base de datos y luego clickear donde dice “importar” que se encuentra en la barra de opciones que podemos ver arriba

localhost/phpmyadmin/index.php?route=/server/databases

Bases de datos SQL Estado actual Cuentas de usuarios Exportar Importar Configuración Replicación Variables Juegos de caracteres Motores Complementos

Nueva

Crear base de datos

Nombre de la base de datos: utf8mb4_general_ci

Base de datos Coleamiento Acción

drafosaurus	utf8mb4_general_ci	• Seleccionar privilegios
information_schema	utf8_general_ci	• Seleccionar privilegios
mysql	utf8mb4_general_ci	• Seleccionar privilegios
performance_schema	utf8_general_ci	• Seleccionar privilegios
phpmyadmin	utf8_bin	• Seleccionar privilegios
test	latin1_swedish_ci	• Seleccionar privilegios

Total: 6

Nota: Activar aquí las estadísticas de la base de datos podría causar tráfico pesado entre el servidor web y el servidor MySQL.

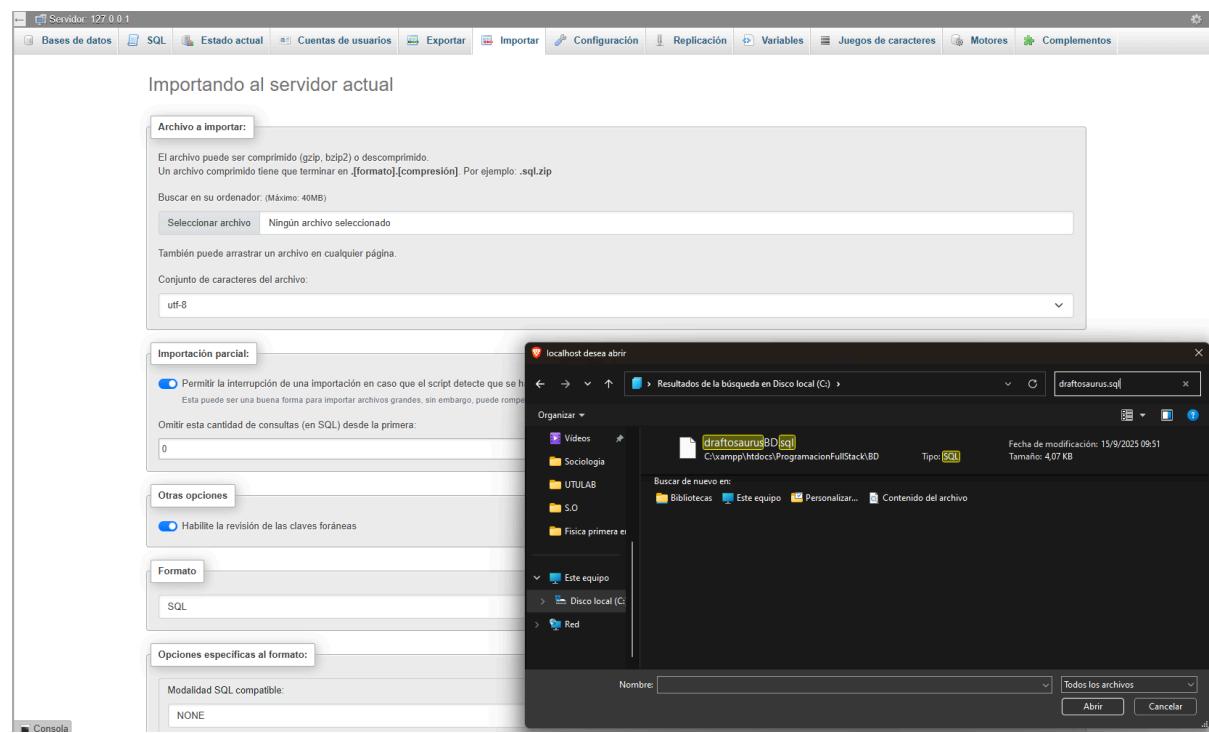
Activar estadísticas



3ºMA

ITI

6. Una vez dentro de la opcion “importar” hacemos click en “seleccionar archivo”, dirigirse a la ruta de la base de datos y seleccionar el archivo

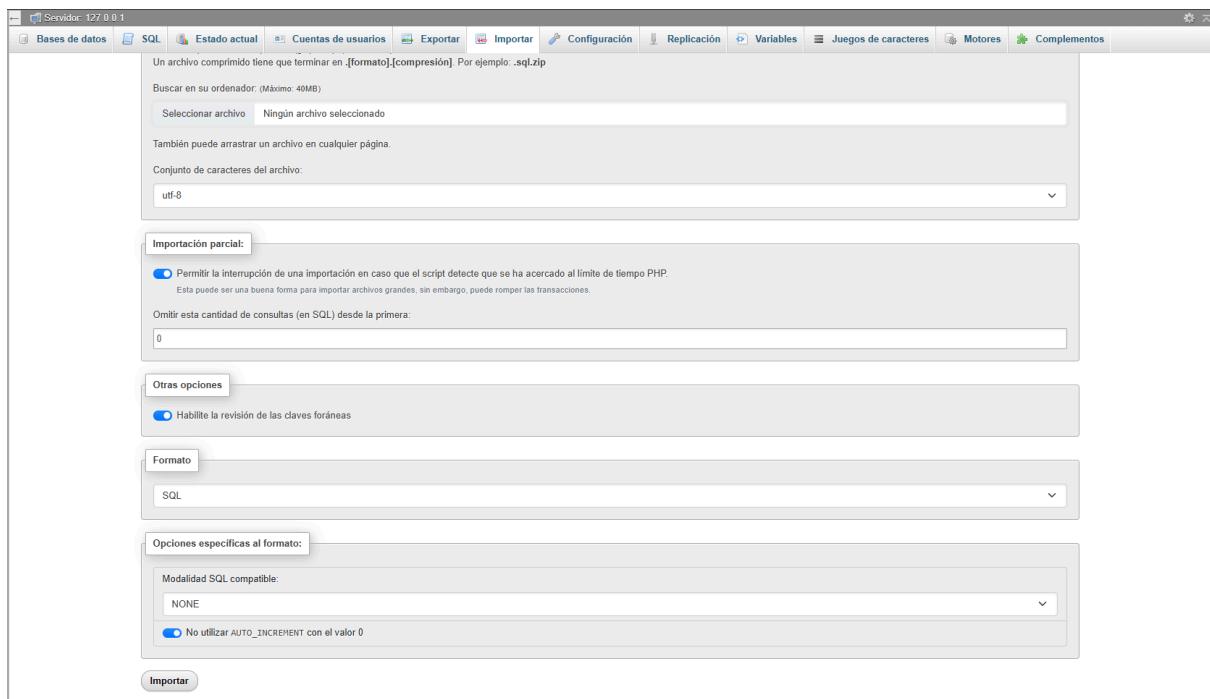




3ºMA

ITI

7. Dirigirse abajo del todo en la pagina y clickear el botón “importar”



Fin manual de instalación

Retrospectiva final del proyecto:

Individual:

Coordinador - Luciano Gonzalez: Como coordinador del proyecto, asumí un rol central en la organización, planificación y ejecución de las distintas etapas. Me encargué de todo lo relacionado con Ingeniería de Software, gran parte de la programación, parte del área de Emprendedurismo y la totalidad de UTU Lab.

Esta experiencia me permitió fortalecer mis habilidades técnicas y de liderazgo, comprendiendo la importancia de la coordinación, la comunicación y la gestión del tiempo. Aprendí a equilibrar las tareas técnicas con la toma de decisiones estratégicas y a mantener la cohesión del equipo.

En general, considero que mi desempeño fue positivo, ya que logré cumplir los objetivos propuestos y aportar de manera significativa al desarrollo del proyecto. Sin embargo, reconozco que debo mejorar en la delegación de tareas y en la distribución equitativa del trabajo dentro del equipo.



3ºMA

Sub-Coordinador - Emiliano Moreira

ITI

Integrante 1 - Matias Porta:

Como integrante del equipo, asumí principalmente un rol enfocado en el desarrollo y la programación del proyecto. Me encargué de implementar gran parte de la lógica y de asegurar el correcto funcionamiento del sistema, aplicando conocimientos previos de programación FullStack y adquiriendo nuevas habilidades a lo largo del proceso. Esta experiencia me permitió profundizar en el uso de distintas tecnologías y mejorar mi capacidad para resolver problemas técnicos de manera eficiente.

Durante el desarrollo del proyecto, surgieron algunas dificultades relacionadas con la comunicación y la coordinación entre ciertos miembros del grupo, lo que ocasionó demoras en la ejecución de algunas tareas. Sin embargo, estos desafíos sirvieron como aprendizaje para comprender la importancia del trabajo en equipo, la organización y la planificación conjunta para alcanzar los objetivos propuestos de manera más efectiva.

Grupal:

Durante el desarrollo del proyecto, el grupo enfrentó diversos desafíos relacionados con la comunicación y el trabajo en equipo. En algunas etapas se presentaron dificultades para coordinar tareas, mantener una comunicación constante y distribuir las responsabilidades de manera equilibrada. Estas situaciones generaron ciertos retrasos y malentendidos que exigieron un esfuerzo adicional de organización y colaboración.

A pesar de estos inconvenientes, el equipo logró adaptarse, mejorar la coordinación y fortalecer la cooperación entre algunos de los integrantes. Con el tiempo, se establecieron métodos de trabajo más claros y una comunicación más efectiva, lo que permitió avanzar de manera ordenada hacia los objetivos planteados.

Finalmente, gracias al compromiso y la dedicación de algunos miembros, se cumplieron los requerimientos pautados y se alcanzaron los resultados esperados.



BALATECH

3ºMA

ITI

El proyecto permitió al grupo adquirir experiencia práctica, desarrollar habilidades técnicas y mejorar la capacidad de trabajo conjunto, demostrando que la comunicación y la organización son claves para el éxito colectivo.