

S.I.G.P.D.

Sistema Informático de Gestión de Partidas para Draftosaurus



Luis E. Fagúndez - Asistente Administrador de Informática
Gabriel L. Barboza - Asistente Administrador de Informática
Luis G. Gonella - Asistente Administrador de Informática

ÍNDICE

1. Introducción al documento	3
2. Marco Reglamentario	3
3. Situaciones puntuales	3
4. Entrega de avances de proyecto	4
5. Fechas importantes	4
6. Defensa del proyecto de egreso	4
7. Generalidades del proyecto	5
7.1. Introducción	5
Objetivo del Juego	5
Cómo se Juega	5
Preparación del Juego en modo base	5
Desarrollo del Juego	5
Los turnos de los jugadores	5
Restricciones del Dado de Colocación	6
Reglas de los Recintos y Puntuación	6
Fin del Juego y Puntuación Final	7
Resumen Rápido	7
7.2. Alcance del Proyecto	8
7.3. Organización del proyecto	8
Parámetros de Desarrollo	9
Usuarios del Sistema	9
8. Requerimientos generales del proyecto de egreso por parte del cuerpo docente	9
Evaluación de las competencias del centro	10
Criterios de logros para cada asignatura	10
8.1. CRITERIOS DE LOGRO PARA INGENIERÍA DEL SOFTWARE	10
8.2. CRITERIOS DE LOGRO PARA PROGRAMACIÓN FULL STACK	16
8.3. CRITERIOS DE LOGRO PARA ADMIN. DE SISTEMAS OPERATIVOS	30
8.4. CRITERIOS DE LOGRO PARA TUTORÍA DE PROYECTO UTULAB:	34
8.5. CRITERIOS DE LOGRO PARA EMPRENDEDURISMO Y GESTIÓN	37
8.6. CRITERIOS DE LOGRO PARA FÍSICA:	37
8.7 CRITERIOS DE LOGROS PARA SOCIOLOGÍA	39
9. Pautas obligatorias para la presentación del proyecto	43
10. ANEXOS	45

1. Introducción al documento

En esta instancia se desarrolla el contenido de dicho documento referente al desarrollo del proyecto de manera integradora, identificando en hitos parciales, como entregas de carácter de seguimiento y control del mismo, a solicitud o inquietudes por parte del alumnado o los correspondientes tutores.

2. Marco Reglamentario

El proyecto será realizado por grupos de alumnos, quedando a criterio del docente de Ingeniería del software y Tutoría de Proyecto UTULAB (como interlocutores válidos del cuerpo docente) la conformación de los mismos y el número de integrantes. Se sugiere un mínimo de 3 y un máximo de 4 integrantes, analizando las posibles situaciones de los casos excepcionales, la conformación de los grupos se realizará por conformidad de los alumnos.

Cada grupo de proyecto deberá elegir entre sus integrantes a un Coordinador de Proyecto que tendrá la representación oficial del grupo a los efectos de cualquier notificación, trámite o indicaciones técnicas. Se indicará también un segundo interlocutor Sub-Coordinador previendo circunstancias coyunturales que inhabilitaran la participación del Coordinador.

El grupo de proyecto deberá seleccionar un **Nombre de Empresa** (nombre fantasía) a los efectos de identificar con un nombre único al grupo, siendo esta acción obligatoria.

La presentación de las carpetas del proyecto se realizará siguiendo las pautas que oportunamente se suministrarán.

Cada grupo confirmará su integración oficial (acción obligatoria), entregando en medio impreso y digital a todos los actores (Dirección, Coordinación de Informática, Adscripción y cada uno de los docentes que integran el proyecto), con la firma de todos los integrantes, un documento en el cual se incluirá el nombre del Grupo de Proyecto, la nómina de los integrantes con C.I., nombre, e-mail de cada integrante, identificando al Coordinador del proyecto y su suplente. Esta gestión tendrá como plazo máximo una semana y media, a partir de la presentación oficial de la letra del proyecto en la Coordinación, y a los actores anteriormente citados, a efectos de facilitar lo anterior, en los anexos al final se encuentra un ejemplo de carta de presentación y acuse de recibo.

3. Situaciones puntuales

Si una parte minoritaria del grupo de proyecto se retirara del mismo, cederá los derechos sobre el **proyecto de egreso**, incluyendo el nombre de la **empresa fantasía**, al resto de los integrantes.

En todos los casos se deberá buscar el consentimiento de todos los integrantes del grupo.

La **presentación de las carpetas del proyecto** correspondientes a cada entrega se realizará en la fecha indicada y deberán ser acompañadas de un documento con copia, detallando el material entregado que permitirá el acuse de recibo por parte del receptor. El lugar de entrega será la Coordinación de Informática y deberá **realizarse en el turno correspondiente al grupo**.

Cada entrega se considerará un documento de avance del proyecto de egreso, por lo cual debe incluir la entrega anterior con las correcciones indicadas por el docente conformando un documento único e integrado dentro de cada asignatura.

Las carpetas de cada entrega deberán ajustarse a las directrices de formato establecidas en el anexo que acompaña el presente documento.

Las entregas parciales o finales, que a juicio docente sean copias de otro proyecto o de material de Internet o realizados en su totalidad con IA, o hayan sido realizadas por otros que no integren el grupo de proyecto, serán consideradas como **ilegítimas**, lo que configurará falta grave derivándose el caso a consideración del CAP, pudiéndose aplicar el Art. 34 Inc. D del estatuto del estudiante Acta Nro. 47 del CODICEN”

No existirán instancias en el curso para reclamos frente a la detección de proyectos copiados, independientemente de las causas que pudieran originarse.

Los cambios de equipos se podrán realizar únicamente hasta una semana pasada la segunda entrega de proyecto. Queda totalmente prohibido los cambios de equipos en el lapso de tiempo de la realización de la tercera entrega, salvo situaciones de fuerza mayor.

4. Entrega de avances de proyecto

La presentación de las carpetas del proyecto correspondientes a cada entrega se realizará en la fecha y horarios indicados por la Coordinación y deberán ser acompañadas de un documento con copia (**ACUSE DE RECIBO**), detallando el material entregado. En la copia se realizará el acuse de recibo por parte del receptor, que será devuelto al grupo. Aquellos grupos que no entreguen en fecha (1era y 2da entrega), tendrán otra oportunidad (DÍA DE ENTREGA ATRASADA) de fecha asignada por el instituto, pero este incumplimiento se considerará en forma negativa (se descontará puntos) en la evaluación de dicha entrega, significando esto una posible calificación negativa. No tendremos instancia de 3er avance de ENTREGA ATRASADA.

5. Fechas importantes

ENTREGA DE LOS REQUERIMIENTOS A LOS GRUPOS: **12/05/2025**

ENTREGA DE CONFORMACIÓN DE LOS GRUPOS HASTA: **22/05/2025**

La entrega física de la conformación se realizará en la Coordinación de Informática y las demás se realizarán de forma digital por CREA.

ENTREGAS DE AVANCE DE PROYECTO:

PRIMERA. ENTREGA: 14/07/2025

SEGUNDA. ENTREGA: 15/09/2025

TERCERA. ENTREGA: (Se realiza en dos partes)

- *Documentación Final del proyecto: 10/11/2025*
- *Instalación del proyecto: 12/11/2025*

En caso de fuerza mayor la entrega se realizará el próximo día hábil.

Primera y segunda entrega se realizará en formato virtual debiendo entregar copia de toda la documentación en la Coordinación de Informática mediante acuse de recibo.

La entrega final se realizará en formato físico en la Coordinación de Informática mediante acuse de recibo.

6. Defensa del proyecto de egreso

Se definirá la defensa de proyecto luego de la segunda entrega

7. Generalidades del proyecto

Draftosaurus es un juego de mesa familiar y estratégico diseñado por Antoine Bauza, Corentin Lebrat, Ludovic Maublanc y Théo Rivière. Fue publicado en 2019 por Ankama y Board Game Box. Es un juego rápido (10-15 minutos por partida) con mecánicas de draft y colocación de trabajadores, en el que los jugadores construyen su propio parque de dinosaurios con el objetivo de obtener la mayor cantidad de puntos.

Instrumentos que vienen en el juego base:

- 5 Tableros de parque (uno para cada jugador)
- 60 Dinosaurios de madera (en 6 colores diferentes, representando diferentes especies)
- 1 Dado de colocación
- 1 Bolsa de tela
- 1 Reglamento

Se adjunta un video en el cual se explica el paso a paso de como es una partida normal de este juego: [Enlace del video](#)

<https://www.youtube.com/watch?v=-ZyFqRNkiAU&feature=youtu.be>

Se buscará desarrollar una app de seguimiento que se detalla más abajo en el alcance del proyecto.

7.1. Introducción

Objetivo del Juego

Construir un parque de dinosaurios y colocarlos estratégicamente en los distintos recintos para obtener la mayor cantidad de puntos al final de la partida.

Cómo se Juega

El juego se desarrolla en **dos rondas**, y cada ronda tiene **seis turnos**.

Preparación del Juego en modo base

1. **Cada jugador toma un tablero del parque** y lo coloca frente a sí.
2. **Todos los dinosaurios se meten en la bolsa** y se mezclan bien.
3. **Decidan quién será el primer jugador** (puede ser al azar).

Desarrollo del Juego

El juego se desarrolla en **dos rondas**, cada una con **6 turnos**.

Los turnos de los jugadores

- 1) **Toma de Dinosaurios**
 - Cada jugador **toma 6 dinosaurios al azar** de la bolsa.
 - Los mantiene en su mano sin mostrarlos a los demás.
- 2) **Lanzamiento del Dado**

- El jugador activo **lanza el dado de colocación**.
 - El dado **impone una restricción para el resto de jugadores** en la colocación del próximo dinosaurio (el jugador activo no aplica esta restricción).
- 3) Elección y Colocación de un Dinosaurio**
- Todos los jugadores **eligen 1 dinosaurio de su mano** y lo colocan en su parque, respetando las reglas de colocación y la restricción del dado si corresponde.
- 4) Paso de dinosaurios restantes y cambio de jugador activo**
- Cada jugador **pasa los dinosaurios restantes** de su mano al jugador de su izquierda.
 - Se pasa el dado en sentido horario al nuevo jugador activo.
 - Se **repite el proceso desde el paso 2** hasta que se hayan colocado **6 dinosaurios**.
- 5) Fin de la Primera Ronda**
- Se repiten los mismos pasos sacando **6 dinosaurios nuevos de la bolsa** para una segunda ronda.

Restricciones del Dado de Colocación

El dado determina en qué zona del parque se debe colocar un dinosaurio en ese turno (no aplica para el jugador que lo lanza).

- **Zona izquierda o derecha** del parque.
- **Zona boscosa o de rocas**.
- **Recinto vacío**.
- **Recinto que no contenga un T-Rex**.

Si un jugador no puede colocar un dinosaurio en un recinto válido, debe colocarlo en el río.

Reglas de los Recintos y Puntuación

Cada recinto tiene sus propias reglas de colocación y puntúa de forma diferente al final del juego:

- 1. El Bosque de la Semejanza:**
 - Este recinto sólo puede albergar dinosaurios de la misma especie.
 - Debe ocuparse siempre de izquierda a derecha sin dejar espacios intermedios.
 - Al final de la partida, ganarás los puntos de victoria indicados según el número total de dinosaurios colocados en el recinto.
- 2. El Prado de la Diferencia:**
 - Este recinto sólo puede albergar dinosaurios de especies distintas.
 - Debe ocuparse siempre de izquierda a derecha sin dejar espacio intermedios.
 - Al final de la partida, ganarás los puntos de victoria indicados según el número de dinosaurios colocados en el recinto.

3. La Pradera del Amor:

- Este recinto puede albergar dinosaurios de todas las especies.
- Al final de la partida, conseguirás 5 puntos de victoria por cada pareja de dinosaurios de la misma especie que se encuentre en el recinto.
- Está permitido tener más de una pareja de la misma especie.
- Los dinosaurios que no formen pareja no suman puntos.

4. El Trío Frondoso:

- Este recinto puede albergar hasta 3 dinosaurios sin importar su especie.
- Al final de la partida, ganarás 7 puntos de victoria si hay exactamente 3 dinosaurios dentro del recinto.
- Si al final de la partida el recinto no alberga exactamente 3 dinosaurios, no te llevas ningún punto.

5. El Rey de la Selva:

- Este recinto puede albergar solo 1 dinosaurio.
- Al final de la partida ganarás 7 puntos de victoria si ningún jugador o jugadora tiene en su parque más dinosaurios que tú de esta especie.
- En caso de empate recibes igualmente los 7 puntos.

6. La Isla Solitaria:

- Este recinto puede albergar solo 1 dinosaurio.
- Al final de la partida otorga 7 puntos de victoria si es el único de su especie en tu parque.
- En caso contrario otorga 0 puntos.

Para entender y comprender más a fondo las reglas del juego, se recomienda leer las reglas desde el siguiente enlace: [Manual del juego](#)

https://drive.google.com/file/d/138qY_aZfQ-RXYDA0j6HshSk-1mmJlrG/view

Fin del Juego y Puntuación Final

Después de la segunda ronda (cuando cada jugador ha colocado 12 dinosaurios en su parque), se **suman los puntos** y se determina el ganador.

- **Se suman los puntajes de cada recinto**
- **Cada dinosaurio en el río, suma 1 punto.**
- **Cada recinto con al menos 1 T-Rex otorga 1 punto extra.**

El jugador con más puntos es el ganador.

En caso de empate, el que tenga más dinosaurios en su parque gana.

Resumen Rápido

- Tomar 6 dinosaurios al azar.
- Lanzar el dado y aplicar la restricción.
- Elegir y colocar un dinosaurio.
- Pasar los dinosaurios restantes.
- Repetir hasta colocar 6 dinosaurios (fin de la ronda).

- Hacer una segunda ronda con otros 6 dinosaurios.
- Contar los puntos y determinar al ganador.

7.2. Alcance del Proyecto

Desarrollar una aplicación de seguimiento para una partida de Draftosaurus que contemple todas las peculiaridades de una partida en modo base. Es importante aclarar que la aplicación de seguimiento no es el juego en sí, sino una herramienta capaz de hacer todos los cálculos que se mencionan arriba para mejorar la experiencia del usuario a la hora de disfrutar una partida con sus amigos.

- A. Debe ser una herramienta auxiliar para contar puntos y validar reglas.
- B. Puede permitir a los jugadores ingresar los dinosaurios colocados y calcular automáticamente la puntuación.
- C. Se debe implementar como una aplicación web con PHP, HTML y CSS, donde los jugadores ingresan manualmente los datos.

Desarrollar una experiencia de juego similar al juego real de manera local donde se contemple todo el desarrollo del juego en si usando las herramientas de desarrollo web.

- D. Implica programar el juego completo, permitiendo jugar partidas desde la aplicación usando el modo básico.
- E. Debe incluir la lógica de selección y paso de dinosaurios, colocación en recintos y validación de reglas.
- F. Se deberá de desarrollar con PHP en el backend y JavaScript en el frontend para manejar la interacción.

7.3. Organización del proyecto

El objetivo principal de este proyecto es desarrollar un software intuitivo y fácil de usar que permita gestionar y seguir el desarrollo de una partida de **Draftosaurus**. Este sistema no reemplazará el juego físico, sino que servirá como una herramienta auxiliar para facilitar la puntuación y la aplicación de reglas, mejorando la experiencia de los jugadores. Además, se trabajará en una versión digital del juego que permite jugar de manera local replicando las mecánicas originales.

Para el desarrollo del software, se utilizará **PHP 8.x** en el backend y **MySQL o MariaDB** como motor de base de datos. Se emplea **XAMPP en su versión 8.x** para el entorno de desarrollo, proporcionando todos los paquetes necesarios preinstalados. Sin embargo, para la implementación final, se configurará un sistema **LAMP** manualmente en el servidor que aloja la aplicación.

El producto final consistirá en un software funcional que cumpla con todos los requisitos del sistema, permitiendo a los usuarios seguir partidas en tiempo real o jugar digitalmente de manera local. Es importante destacar que el servidor de bases de datos deberá ejecutarse en un entorno basado en **GNU/Linux**, garantizando estabilidad y compatibilidad con la infraestructura establecida.

Parámetros de Desarrollo

Modo Seguimiento:

- Permitir a los jugadores registrar la colocación de dinosaurios en sus tableros.
- Aplicar automáticamente las reglas y restricciones del juego.
- Calcular la puntuación final y determinar al ganador.

Modo Juego Digitalizado:

- Implementar la dinámica de selección, paso y colocación de dinosaurios.
- Integrar la lógica de restricciones por dado y reglas de puntuación.
- Ofrecer una experiencia de juego fluida y cercana a la versión física.

Usuarios del Sistema

Los usuarios serán definidos en la **Especificación de Requerimientos**, considerando dos perfiles principales:

- **Jugadores:** Utilizarán la aplicación para el registro y cálculo de puntuación o para jugar en la versión digital.
- **Administradores:** Encargados de la configuración y mantenimiento del sistema, asegurando el correcto funcionamiento de la plataforma.

Este sistema proporcionará una herramienta confiable y accesible para mejorar la experiencia de juego de **Draftosaurus**, optimizando la gestión de partidas y ofreciendo una alternativa digital para disfrutar de este emocionante juego de mesa.

8. Requerimientos generales del proyecto de egreso por parte del cuerpo docente

Toda la propuesta deberá enmarcarse en una adecuada **gestión del proyecto**, incluyendo un **cronograma de ejecución** y el **detalle de las etapas de desarrollo**. La aplicación deberá contar con una documentación completa, abarcando tanto el funcionamiento del sistema como su código fuente. Además, el software será desarrollado en **dos idiomas (Español/Inglés)**, garantizando accesibilidad a una mayor audiencia.

La infraestructura del proyecto deberá cumplir con los requisitos técnicos solicitados. El servidor de bases de datos se implementará en un entorno **GNU/Linux**, dejando a criterio del equipo la elección de la distribución

Para garantizar un entorno de desarrollo compatible con la infraestructura de producción, se trabajará con las siguientes herramientas:

Ambiente de Desarrollo:

Herramientas utilizadas recomendadas:

- **Visual Studio Code o similar** como entorno de desarrollo.
- **XAMPP 8.X ó servidor escolar** para la gestión de PHP y MySQL en local.
- **VirtualBox 7.x o VMware Workstation 17** para la virtualización del servidor de pruebas.
- **GitHub.com** para el control de versiones y colaboración en equipo.

Este enfoque permitirá desarrollar una solución robusta y bien documentada, asegurando compatibilidad con la infraestructura final y facilitando su mantenimiento y escalabilidad en el futuro.

Evaluación de las competencias del centro

En el Instituto Superior Brazo Oriental, mediante acuerdo de sala se seleccionaron las competencias de

Criterios de logros para cada asignatura

Los criterios de logros que se indican a continuación son tentativos y no exhaustivos, cada docente indicará qué puntos serán relevantes para él, pudiendo agregar a la misma aquellos puntos que no estuvieran presentes o quitar de la misma aquellos que no se consideren oportunos, mientras sean coordinados en consenso por todo el equipo de docentes y avalado por la coordinación de informática.

8.1. CRITERIOS DE LOGRO PARA INGENIERÍA DEL SOFTWARE PUNTEO GENERAL DE REQUERIMIENTOS PARA PRIMERA ENTREGA

- Relevamiento.
 - Empleo de técnicas vistas durante el curso
 - Elaboración de formularios apropiados para la tarea anterior.
- Estudio de factibilidades.
- Definición de roles de usuario con permisos y privilegios.
- **Especificación de Requerimientos** (Funcionales, No Funcionales, Alcance y Limitaciones).
- Lógica de sistema
 - Árboles de decisión
- Implementación de metodología tradicional.
- Prototipado de la aplicación en coordinación con el Profesor de Fullstack (Propuesta y aplicación) (Mock-up y Wireframe).
- Carátula de la carpeta
- Documentación de inicio y planificación del proyecto:
 - Carta de presentación.
 - Nombre del Equipo, integrantes y sus roles.
 - Reglas del grupo.
 - Diagrama Gantt para la primera entrega.
 - Actas de reunión.
- Creación de un repositorio privado en GitHub.com con un usuario identificado por el nombre de su empresa.
 - Utilizar software de visualización de diagrama de Gantt (Gantt Project, TeamGantt, ClickUp, InstaGantt)

- Subir TODO el proyecto hasta la primera entrega al repositorio.

RÚBRICA DE EVALUACIÓN PARA PRIMERA ENTREGA

Criterios de evaluación	1 punto	4 puntos	7 puntos	10 puntos
Documentación	No presenta los documentos necesarios	Presenta un documento necesario para el proyecto	Presenta algunos documentos necesarios para el proyecto	Presenta todos los documentos necesarios para el proyecto
Relevamiento y estudio de factibilidad	No realiza este punto para la entrega	Realiza un relevamiento mínimo sin análisis.	Realiza un relevamiento y justifica las decisiones mediante análisis de factibilidad.	Realiza un relevamiento exhaustivo y justifica las decisiones mediante análisis de factibilidad
ESRE y Roles de usuario	No realiza este punto para la entrega.	Realiza un esbozo del ESRE y usuarios del sistema.	Realiza un esquema básico del ESRE y usuarios del sistema.	Realiza una exhaustiva aplicación de la IEEE830 para el ESRE y perfilado de usuarios del sistema.
Diagrama Gantt	No realiza este punto para la entrega.	Realiza un análisis mínimo de las tareas a realizar en el proyecto.	Realiza un análisis básico de las tareas a realizar en el proyecto.	Realiza un análisis exhaustivo de las tareas a realizar en el proyecto.
Árbol de decisiones	No realiza este punto para la entrega.	Realiza una estructura mínima del árbol de decisiones	Realiza una estructura básica del árbol de decisiones	Realiza una estructura exhaustiva del árbol de decisiones.
Ciclo de vida	No realiza este punto para la entrega	Selecciona un ciclo de vida sin justificar la decisión	Selecciona un ciclo de vida justificando algunas etapas.	Selecciona un ciclo de vida, justificando cada etapa correspondiente de forma correcta.

PUNTEO GENERAL DE REQUERIMIENTOS PARA SEGUNDA ENTREGA

- Diagrama Gantt para la segunda entrega.
- Diagrama UML (Casos de Uso) (Planilla y Diagrama)
- Diagrama de Clases
- Plan de contingencias.
- Análisis Costo-Beneficio.
- Cálculo de métricas del proyecto.
- Actas de reuniones (SOLO SEGUNDA ENTREGA).
- Demostración al cliente (video de no más de 2 min explicando el avance del producto).
- Subir TODO el proyecto al repositorio con las correcciones de la primera entrega.

RÚBRICA DE EVALUACIÓN PARA SEGUNDA ENTREGA

Criterios de evaluación	1 punto	4 puntos	7 puntos	10 puntos
Diagrama Gantt	No realiza este punto para la entrega.	Realiza un análisis mínimo de las tareas a realizar en el proyecto.	Realiza un análisis básico de las tareas a realizar en el proyecto.	Realiza un análisis exhaustivo de las tareas a realizar en el proyecto.
Diagrama Caso de uso	No realiza este punto para la entrega.	Realiza un diagrama básico de los casos de uso.	Realiza un diagrama extenso de los casos de uso	Realiza un diagrama completo de los casos de uso.
Planilla Caso de uso	No realiza este punto para la entrega	Realiza la minoría de los desarrollos de los casos de uso	Realiza la mayoría de los desarrollos de los caso de uso	Realiza todo el desarrollo de los casos de uso.
Diagrama de clases	No realiza este punto para la entrega.	Realiza un diagrama básico de clases.	Realiza un diagrama extenso de clase	Realiza un diagrama completo de clases
Plan de contingencias	No realiza este punto para la entrega.	Realiza un plan de contingencia básico, sin tener en cuenta las soluciones a los problemas	Realiza un plan de contingencia intermedio, teniendo en cuenta algunas soluciones a los problemas sin definir	Realiza un plan de contingencias completo, teniendo en cuenta todas las soluciones al problema
Análisis Costo-Beneficio	No realiza este punto para la entrega	Realiza un análisis básico de Costo-Beneficio, donde no se contempla tener	Realiza un análisis intermedio de Costo-Beneficio para hacer un cálculo aproximado	Realiza un análisis intermedio de Costo-Beneficio, para tener un cálculo correcto de

		ganancias en el proyecto	del costo del proyecto.	las ganancias en el proyecto
Métricas	No realiza este punto para la entrega.	Identifica los ítems básicos de métricas de PF, pero no resuelve el cálculo correspondiente.	Realiza la mayoría de los cálculos correctamente, con pequeñas imprecisiones.	Realiza todos los cálculos de métricas correctamente y justifica los resultados.
Video	No realiza este punto para la entrega.	Presenta un video que no cumple el tiempo o es confuso en la explicación.	Presenta un video de hasta 2 minutos, pero con explicaciones poco claras o incompletas.	Presenta un video de hasta 2 minutos, claro, conciso, mostrando los avances relevantes del producto.

PUNTEO GENERAL DE REQUERIMIENTOS PARA TERCERA ENTREGA

- Revisión de diagrama de clases
- Análisis F.O.D.A. ponderado
- Plan de testing
 - Pruebas funcionales
 - Tablas de decisión
 - Casos de uso – Casos de prueba
- Comparación métricas tamaño y función.
- Manual de usuario.
- Manual de instalación y mantenimiento de la aplicación.
- Actas de reuniones (PRIMERA, SEGUNDA Y TERCER ENTREGA).
- Documentación de Cierre de Proyecto.
 - Revisión de diagrama Gantt.
 - Demostración final al Cliente (video explicativo del funcionamiento de la aplicación de 2 min en español e inglés).
 - Retrospectiva final del Proyecto (Reflexión grupal e individual de aprendizajes)
 - Evaluación de la Experiencia del Cliente.
- Subir TODO el proyecto al repositorio con las correcciones de la primera y segunda entrega.

RÚBRICA DE EVALUACIÓN PARA TERCERA ENTREGA

Criterios de evaluación	1 punto	4 puntos	7 puntos	10 puntos
Diagrama de clases	No realiza este punto de entrega	Realiza un diagrama básico de clases. No corrigiendo las modificaciones de la anterior entrega	Realiza un diagrama extenso de clase. Realizando algunas modificaciones de la anterior entrega	Realiza un diagrama completo de clases. Realizando todas las modificaciones de la anterior entrega.

I.S.B.O. - IAE Montevideo - ITI CETP
 Proyecto de Egreso - BT - Tecnologías de la Información - 2025

FODA PONDERADO	No realiza este punto de entrega	Realiza un análisis FODA sin ponderación clara	Realiza un análisis FODA con ponderación parcial.	Realiza un FODA completo y ponderado, con análisis profundo.
Plan de testing	No realiza este punto de entrega	Realiza un plan mínimo de testng, testeando algunas funcionalidades	Realiza un plan de testing organizado, testeando casi todas las funcionalidad es del sistema	Realiza un plan de testing completo, testeando al completo todas las funcionalidades.
Manual de usuario	No realiza este punto de la entrega	Presenta un manual básico, incompleto o poco claro.	Presenta un manual casi completo pero con explicaciones vagas.	Presenta un manual completo, claro y bien estructurado.
Manual de instalación y mantenimiento	No realiza este punto.	Presenta un manual incompleto o poco claro.	Presenta un manual casi completo pero con detalles faltantes.	Presenta un manual completo, detallado y comprensible.
Documentación de cierre	No presenta actas.	Presenta pocas actas e incompletas	Presenta las actas pero incompletas	Presenta las actas correctamente detallada
Diagrama Gantt	No realiza este punto para la entrega.	Realiza un análisis mínimo de las tareas a realizar en el proyecto. Sin corregir los errores de las entregas anteriores	Realiza un análisis básico de las tareas a realizar en el proyecto. Corrigiendo algunos punteos de las entregas anteriores	Realiza un análisis exhaustivo de las tareas a realizar en el proyecto. Corrigiendo los punteos de las entregas anteriores
Video	No realiza este punto para la entrega.	Presenta un video que no cumple el tiempo o es confuso en la explicación.	Presenta un video de hasta 2 minutos, pero con explicaciones poco claras o incompletas.	Presenta un video de hasta 2 minutos, claro, conciso, mostrando los avances relevantes del producto.

Subida completa del proyecto al repositorio	No realiza este punto para la entrega.	Sube parte del proyecto pero está incompleto.	Sube el proyecto con la mayoría de las correcciones.	Sube todo el proyecto con todas las correcciones solicitadas.
---	--	---	--	---

REQUERIMIENTOS MÍNIMOS DE INGENIERÍA DE SOFTWARE:

- Estudio de Factibilidad
- Análisis Costo-Beneficio
- Especificación de Requerimientos
- Especificación de los casos de uso, necesarios para representar el sistema
- Diagrama de casos de uso y clases
- Manual de instalación y mantenimiento de la aplicación
- Validación del Sistema Casos de prueba (VyV de Requerimientos)
- Prototipado de la aplicación en coordinación con el Profesor de Fullstack
- Organización de la estructura de desarrollo.
- Actas de reuniones.
- Métricas.
- Creación de un repositorio privado en GitHub.com con un usuario identificado por el nombre de su empresa.
- Subir TODO el proyecto al repositorio.

8.2. CRITERIOS DE LOGRO PARA PROGRAMACIÓN FULL STACK

PRIMERA ENTREGA

1. Justificación Tecnológica

Elaborar un documento técnico estructurado que justifique la selección de tecnologías para el proyecto, incluyendo:

- Análisis de PHP como lenguaje backend, destacando sus ventajas específicas frente a alternativas
- Criterios de selección del sistema gestor de base de datos con justificación técnica
- Evaluación de frameworks frontend seleccionados (Bootstrap y complementarios)
- Selección fundamentada de herramientas para control de versiones y colaboración
- Recomendaciones de entornos de desarrollo con sus ventajas

Este documento debe relacionar explícitamente cada tecnología con los requerimientos específicos del juego Draftosaurus, demostrando cómo estas elecciones optimizan el desarrollo y funcionamiento del sistema.

2. Prototipo de Interfaz de Usuario

Desarrollar un prototipo visual completo con los siguientes elementos:

- Maquetación HTML5 implementando correctamente etiquetas semánticas
- Estilización mediante CSS3 con enfoque en usabilidad y experiencia de usuario
- Integración de Bootstrap u otros frameworks de diseño
- Sistema de navegación funcional entre secciones

El prototipo debe incluir, como mínimo, las siguientes pantallas:

- Página de inicio/bienvenida con presentación del juego
- Tablero de juego (sin funcionalidad) con visualización clara de dinosaurios disponibles
- Pantalla de resultados y sistema de puntuación final

Importante: Esta fase requiere únicamente la estructura visual y navegación, no la implementación de la lógica del juego.

3. Implementación de Diseño Responsivo

- Utilizar Bootstrap 5.3 como framework base por su sistema de grid y componentes prediseñados
- Se permite el uso de frameworks complementarios (responsabilidad del estudiante su implementación)
- El diseño debe garantizar funcionamiento óptimo en dispositivos móviles (320px - 576px)
- Todos los elementos de interfaz deben adaptarse correctamente según las dimensiones de pantalla
- Mantener coherencia visual y experiencia de usuario uniforme en todos los dispositivos

4. Modelado de Datos

4.1 Modelo Entidad-Relación (DER)

- Desarrollar un diagrama entidad-relación inicial (primera versión) que refleje fielmente la estructura del juego
- Incluir todas las entidades principales con sus atributos y relaciones

4.2 Esquema Relacional Normalizado

- Realizar la transformación del DER al modelo relacional (pasaje a tablas)
- Aplicar el proceso de normalización hasta la Tercera Forma Normal (3FN)
- Identificar y documentar claramente claves primarias y foráneas

- Registrar todas las decisiones de diseño con su justificación técnica

4.3 Restricciones No Estructurales

- Documentar exhaustivamente las restricciones no estructurales derivadas de las reglas del juego
- Especificar cómo estas restricciones serán implementadas en el sistema

5. Configuración del Entorno de Desarrollo

- Documentar detalladamente, paso a paso, el proceso de instalación y configuración del entorno
- Especificar los requisitos de software (servidor Apache, versión de PHP, MySQL/MariaDB)
- Incluir configuración de IDE (Visual Studio Code recomendado) y extensiones útiles
- Proporcionar instrucciones verificables para replicar el entorno

6. Control de Versiones y Repositorio

- Crear y estructurar repositorio inicial en GitHub siguiendo buenas prácticas
- Establecer y documentar convenciones de commits y mensajes
- Asegurar que el repositorio contenga:
 - Código fuente completo del prototipo
 - Documentación técnica organizada
 - Archivo README.md con instrucciones claras de instalación y uso

criterio	1 punto	4 puntos	7 puntos	10 puntos
1. Justificación Tecnológica	No presenta el documento o está incompleto.	Presenta la justificación con descripciones mínimas y sin vinculación clara con el proyecto.	Justifica las tecnologías seleccionadas con argumentos técnicos generales; la relación con el proyecto es parcial o poco explícita.	Justifica de forma técnica y detallada todas las tecnologías seleccionadas, vinculándolas directamente con los requerimientos específicos del juego.
2. Prototipo de Interfaz de Usuario	No presenta el prototipo o es irreconocible.	Presenta un prototipo incompleto o con errores importantes de estructura y navegación.	Presenta un prototipo funcional con navegación básica, etiquetas semánticas y uso adecuado de CSS y Bootstrap, aunque con detalles de usabilidad por mejorar.	Presenta un prototipo visual completo, con navegación funcional, etiquetas semánticas, diseño atractivo y experiencia de usuario bien lograda.

<p>3. Implementación de Diseño Responsivo</p>	<p>No presenta el diseño responsivo o es ineficaz.</p>	<p>Presenta un diseño que se adapta de forma parcial a pantallas pequeñas, sin coherencia visual clara.</p>	<p>Logra adaptabilidad aceptable con Bootstrap 5.3, aunque algunos elementos no se ajustan completamente a móviles o presentan errores.</p>	<p>Integra diseño responsivo efectivo en todas las pantallas, asegurando coherencia, adaptabilidad y buena experiencia en dispositivos móviles.</p>
<p>4.1 Modelo Entidad-Relación (DER)</p>	<p>No presenta el DER o es incorrecto.</p>	<p>Presenta un DER básico, con entidades principales pero relaciones o atributos mal definidos.</p>	<p>Presenta un DER correcto con entidades, atributos y relaciones relevantes, aunque con algunos errores menores.</p>	<p>Presenta un DER completo, claro y bien estructurado, reflejando fielmente la lógica del juego.</p>
<p>4.2 Esquema Relacional Normalizado</p>	<p>No presenta el esquema o está mal transformado.</p>	<p>Presenta el esquema relacional con errores de transformación o sin aplicar normalización.</p>	<p>Aplica transformación adecuada al modelo relacional con normalización hasta 2FN; incluye claves primarias y foráneas.</p>	<p>Realiza correctamente la transformación hasta 3FN, con claves bien identificadas y decisiones de diseño justificadas.</p>
<p>4.3 Restricciones No Estructurales</p>	<p>No se presentan las restricciones.</p>	<p>Menciona algunas restricciones sin detallar su implementación.</p>	<p>Documenta restricciones relevantes, aunque no todas están bien fundamentadas o especificadas.</p>	<p>Documenta exhaustivamente todas las restricciones del juego y explica cómo serán implementadas en el sistema.</p>

5. Configuración del Entorno de Desarrollo	No presenta la documentación o es inentendible.	Presenta pasos generales para configurar el entorno, sin verificabilidad o claridad.	Detalla correctamente la instalación del entorno y las herramientas, pero faltan instrucciones específicas para su replicación.	Documenta paso a paso el proceso completo, con requisitos, configuraciones e instrucciones claras y replicables.
6. Control de Versiones y Repositorio	No utiliza control de versiones o el repositorio es inadecuado.	Crea el repositorio en GitHub pero sin estructura clara ni buenas prácticas.	Organiza el repositorio con estructura básica, incluye parte del código y documentación; convenciones de commits poco claras.	Configura un repositorio bien estructurado, con código y documentación completa, README claro y buenas prácticas de commits documentadas.

SEGUNDA ENTREGA

1. Modelado de Datos

1.1 Correcciones y Avances del Modelo

- Implementar ajustes al DER según retroalimentación de la primera entrega
- Refinar el esquema relacional normalizado (3FN), garantizando eliminación de redundancias
- Documentar detalladamente las Restricciones No Estructurales (RNE) que representan reglas de negocio específicas

2. Implementación de Diseño Responsivo

- Continuar utilizando Bootstrap 5.3 como framework principal
- Permitido el uso de frameworks complementarios bajo responsabilidad del estudiante
- Expandir compatibilidad a:
 - Dispositivos móviles (320px - 576px)
 - Tabletas (577px - 992px)
- Garantizar adaptación coherente de todos los elementos de interfaz

3. Implementación Física de la Base de Datos

3.1 Sentencias DDL (Data Definition Language)

- Desarrollar scripts para creación de tablas con claves primarias y foráneas
- Implementar índices optimizados para consultas frecuentes

3.2 Gestión de Usuarios y Permisos

3.3 Datos de Prueba

- Crear scripts para inserción de registros iniciales de prueba
- Asegurar la reproducibilidad de estos scripts en diferentes entornos

4. Desarrollo de la Aplicación

Objetivo Principal: Implementar versión funcional básica centrada en experiencia individual, con diseño escalable para futura expansión multijugador.

Arquitectura de Software:

Implementar una estructura de tres capas tanto en frontend como en backend siguiendo los principios de la Programación Orientada a Objetos (POO):

- **Capa de Presentación:** Interfaz de usuario (HTML, CSS, JavaScript)
- **Capa de Negocio:** Lógica del juego, reglas y controladores
- **Capa de Datos:** Acceso y persistencia en base de datos
- Mantener separación clara de responsabilidades entre capas
- Documentar las clases y sus relaciones.

4.1 Funcionalidades a implementar

- **Tablero de juego personal:**
 - Renderizado dinámico mediante JavaScript
 - Interacción con elementos del juego (selección, movimientos)
 - Implementación de arrastrar y soltar (drag and drop) para colocar dinosaurios
 - Validación de reglas específicas del juego
- **Validación de Formularios:**
 - Verificación dual (frontend con JavaScript, backend con PHP)
 - Sistema de mensajes de error contextualizados
- **Comunicación con el Servidor (Fetch/Async-Await):**
 - Implementación de consultas asíncronas para:
 - Carga de dinosaurios disponibles
 - Guardado de progreso
- **Conexión a Base de Datos con DML:**
 - Operaciones CRUD mediante PHP y MySQL
 - Implementación de consultas preparadas (prevención de SQL Injection)
- **Formulario de Configuración:**
 - Personalización de parámetros iniciales
 - Sistema de persistencia de partidas

Criterio	1 punto	4 puntos	7 puntos	10 puntos
1.1 Correcciones y Avances del Modelo	No realiza correcciones al DER ni al esquema relacional.	Realiza algunas correcciones mínimas en el DER, pero sin mejorar la normalización o eliminar redundancias.	Realiza correcciones sustanciales al DER y mejora la normalización del esquema relacional, aunque aún con pequeñas redundancias.	Refina completamente el DER y el esquema relacional, garantizando eliminación de redundancias y detallando adecuadamente las restricciones no estructurales.

<p>2. Implementación de Diseño Responsivo</p>	<p>No se amplía la compatibilidad en dispositivos.</p>	<p>Amplía la compatibilidad a dispositivos móviles, pero con errores de adaptación y sin considerar tabletas.</p>	<p>Garantiza una buena adaptación en dispositivos móviles y tabletas, aunque algunos elementos no se ajustan correctamente.</p>	<p>Asegura una adaptación perfecta en móviles y tabletas, con una experiencia de usuario coherente y sin errores de visualización.</p>
<p>3. Implementación Física de la Base de Datos</p>	<p>No se desarrolla la implementación física de la base de datos.</p>	<p>Crea algunas tablas y relaciones pero sin optimización en índices o scripts para inserción de datos.</p>	<p>Desarrolla tablas y relaciones correctamente con claves primarias y foráneas, y mejora parcialmente la optimización de índices y datos de prueba.</p>	<p>Desarrolla tablas y relaciones completas, optimiza índices, y crea scripts de inserción de datos de prueba claros y reproducibles.</p>
<p>4. Arquitectura de Software</p>	<p>No presenta la estructura en capas ni la separación de responsabilidades.</p>	<p>Implementa una estructura básica de capas pero sin aplicar principios de la Programación Orientada a Objetos (POO) ni separación clara de responsabilidades.</p>	<p>Implementa la estructura en capas, sigue los principios de la POO, pero la separación de responsabilidades es parcial o inadecuada.</p>	<p>Implementa de manera efectiva la estructura en tres capas, mantiene una separación clara de responsabilidades entre cada capa.</p>

<p>4.1 Funcionalidades a Implementar</p>	<p>No presenta ninguna de las funcionalidades requeridas o están muy incompletas.</p>	<p>Implementa algunas funcionalidades del tablero o la validación de formularios, pero con errores o sin interacción completa.</p>	<p>Implementa la mayoría de las funcionalidades del tablero, validación de formularios y comunicación con el servidor, pero algunos aspectos son funcionalmente limitados o presentan errores.</p>	<p>Implementa todas las funcionalidades de manera correcta, incluyendo tablero de juego interactivo, validación de formularios completa y comunicación asíncrona con el servidor, con código limpio y eficiente.</p>
<p>4.2 Funcionalidad de Arrastrar y Soltar (Drag and Drop)</p>	<p>No se implementa la funcionalidad de arrastrar y soltar.</p>	<p>Implementa una funcionalidad de arrastrar y soltar básica pero con errores o sin validación de las reglas del juego.</p>	<p>Implementa arrastrar y soltar correctamente, pero con algunas restricciones o errores en la validación de las reglas del juego.</p>	<p>Implementa arrastrar y soltar de manera fluida, validando correctamente las reglas del juego y garantizando una experiencia de usuario óptima.</p>
<p>4.3 Validación de Formularios</p>	<p>No se implementa validación de formularios.</p>	<p>Implementa validación básica en el frontend pero sin considerar el backend o sin mensajes de error claros.</p>	<p>Implementa validación adecuada tanto en frontend como en backend, aunque algunos mensajes de error no son del todo claros.</p>	<p>Implementa validación eficiente en frontend y backend, con mensajes de error claros y contextualizados, garantizando seguridad y usabilidad.</p>

<p>4.4 Comunicación con el Servidor (Fetch/Async-Await)</p>	<p>No implementa comunicación asíncrona o presenta errores importantes.</p>	<p>Implementa algunas consultas asíncronas básicas, pero no funciona correctamente o no se usa de manera eficiente.</p>	<p>Implementa correctamente consultas asíncronas para carga de dinosaurios y guardado de progreso, aunque con errores menores o mejoras posibles.</p>	<p>Implementa consultas asíncronas de manera eficiente y fluida, con interacciones adecuadas para cargar dinosaurios y guardar progreso sin errores.</p>
<p>4.5 Conexión a la Base de Datos con DML</p>	<p>No se conecta a la base de datos ni se realizan operaciones CRUD.</p>	<p>Implementa operaciones CRUD básicas pero sin prevención de SQL Injection ni buenas prácticas en la conexión.</p>	<p>Implementa correctamente operaciones CRUD con prevención de SQL Injection, pero con algunos detalles de optimización faltantes.</p>	<p>Implementa operaciones CRUD completas, optimizadas y seguras, con consultas preparadas correctamente implementadas.</p>
<p>4.6 Formulario de Configuración</p>	<p>No se presenta el formulario de configuración.</p>	<p>Implementa un formulario de configuración básico pero sin sistema de persistencia de partidas.</p>	<p>Implementa un formulario de configuración funcional, pero con algunos errores en la persistencia de partidas o personalización de parámetros.</p>	<p>Implementa completamente el formulario de configuración, con personalización de parámetros y sistema de persistencia de partidas bien integrado.</p>

TERCERA ENTREGA

1. Modelado de Datos

1.1 Consolidación del Modelo

- Incorporar correcciones finales al MER según retroalimentación
- Presentar esquema relacional definitivo en 3FN
- Desarrollar script DDL final con todas las estructuras, relaciones y restricciones

1.2 Implementación Física Completa

- **Finalizar modelo físico incluyendo:**
 - Estructuras de almacenamiento optimizadas (tipos de datos)
 - Vistas para consultas frecuentes (ej: ranking de jugadores)
- **Gestión de acceso:**
 - Scripts para creación de usuarios con distintos niveles de privilegios
 - Implementación de roles diferenciados.
- **Políticas de mantenimiento:**
 - Estrategia documentada para backups (diarios/incrementales)

2. Implementación de Diseño Responsivo

- Uso avanzado de Bootstrap 5.3:
 - Sistema grid completamente adaptativo
 - Implementación coherente de componentes UI
 - Utilización óptima de utilidades de espaciado responsive
- **Compatibilidad Multiplataforma completa:**
 - Mobile-First (320px - 576px)
 - Tablet (577px - 992px)
 - Escritorio (≥993px)

3. Desarrollo de la Aplicación

Objetivo Central: Implementar versión multijugador completa con:

- Sistema de gestión de perfiles (registro e historial)
- Configuración flexible de partidas
- Seguimiento simultáneo de múltiples tableros
- Cálculo automático de puntuaciones
- Registro histórico de partidas y rankings

Arquitectura de Software:

- Consolidar y refinar la estructura de tres capas tanto en frontend como en backend:
 - **Capa de Presentación:** Interfaces responsivas y componentes reutilizables
 - **Capa de Negocio:** Implementación completa de reglas del juego, validaciones y lógica de negocio
 - **Capa de Datos:** Gestión optimizada de persistencia y recuperación de información
- Aplicar rigurosamente los principios de Programación Orientada a Objetos:
 - Encapsulamiento apropiado de datos
 - Herencia para elementos comunes del juego
 - Polimorfismo para manejar diferentes tipos de recintos y dinosaurios
 - Interfaces y clases abstractas donde sea pertinente
- Documentar el diseño arquitectónico con diagramas UML relevantes en coordinación con Ingeniería del Software

3.1 Requisitos Técnicos

- **Repositorio GitHub:**
 - Código organizado.

- Documentación técnica actualizada.
- **Stack Tecnológico:**
 - Frontend: JavaScript/Bootstrap
 - Backend: PHP
 - Persistencia: MySQL
 - Estricta arquitectura en 3 capas tanto en frontend como en backend, siguiendo principios POO

4. Material de Entrega

- USB identificado conteniendo:
 - Paquete portable con:
 - Máquina virtual configurada
 - Código fuente ejecutable
 - Documentación completa:
 - Manual de usuario
 - Credenciales de prueba
 - Identificación formal:
 - Carátula oficial del proyecto
 - Nombres completos de integrantes

Nota: El sistema deberá conectarse a una instancia MySQL local utilizando los scripts DDL/DML incluidos en la entrega.

Criterio	1 punto	4 puntos	7 puntos	10 puntos
1.1 Consolidación del Modelo	No se incorporan las correcciones del MER ni se presenta el esquema relacional.	Realiza algunas correcciones al MER, pero el esquema relacional no está completamente en 3FN o faltan algunas relaciones.	Presenta un esquema relacional en 3FN, pero algunas relaciones o restricciones no están completamente detalladas.	Refina completamente el MER, presentando un esquema relacional en 3FN, con todas las relaciones y restricciones detalladas, y un script DDL final completo.

<p>1.2 Implementación Física Completa</p>	<p>No se implementan estructuras físicas ni optimizaciones</p>	<p>Desarrolla algunas estructuras físicas, pero sin vistas ni una organización clara para la lógica del juego.</p>	<p>Implementa estructuras físicas y vistas, pero la gestión de acceso o la optimización de las consultas no están completamente desarrolladas.</p>	<p>Desarrolla una implementación física completa con estructuras de almacenamiento optimizadas, vistas para consultas frecuentes, y gestión de acceso con roles diferenciados y políticas de mantenimiento bien documentadas.</p>
<p>2. Implementación de Diseño Responsivo</p>	<p>No se implementa un diseño responsivo o es muy limitado.</p>	<p>Implementa diseño responsivo básico con el uso de Bootstrap, pero no se adapta adecuadamente a todas las plataformas.</p>	<p>Implementa un diseño responsivo que se adapta bien a móviles y tabletas, pero con algunos errores en la adaptación a escritorio.</p>	<p>Implementa un diseño completamente responsivo con el sistema grid de Bootstrap, adaptando correctamente la interfaz a dispositivos móviles, tabletas y escritorio, utilizando de forma óptima las utilidades de espaciado.</p>

<p>3. Desarrollo de la Aplicación</p>	<p>La aplicación no tiene funcionalidad multijugador o es incompleta.</p>	<p>Implementa algunas características multijugador, pero la gestión de perfiles, configuración de partidas, o el seguimiento de tableros no están completamente desarrollados.</p>	<p>Implementa la mayoría de las funcionalidades del sistema multijugador, pero algunos aspectos (como la gestión de perfiles o el cálculo de puntuaciones) tienen errores o limitaciones.</p>	<p>Implementa una versión completa y funcional del sistema multijugador, con gestión de perfiles, configuración de partidas, seguimiento simultáneo de tableros, cálculo automático de puntuaciones y registro histórico de partidas y rankings.</p>
<p>Arquitectura de Software</p>	<p>No se presenta una estructura clara en tres capas o principios de POO.</p>	<p>Implementa una estructura básica en tres capas, pero sin aplicar correctamente principios de POO (encapsulamiento, herencia, polimorfismo).</p>	<p>Implementa correctamente la estructura en tres capas y principios de POO, pero sin un enfoque consistente o adecuado en todos los elementos.</p>	<p>Consolida y refina la arquitectura en tres capas, aplicando rigurosamente los principios de POO (encapsulamiento, herencia, polimorfismo) y utilizando interfaces y clases abstractas donde sea necesario. Además, documenta el diseño con diagramas UML.</p>

3.1 Requisitos Técnicos	El código no está organizado ni documentado adecuadamente.	El código está parcialmente organizado en ramas, pero no sigue buenas prácticas de documentación.	El código está bien organizado en ramas y documentado parcialmente, pero con algunas áreas de mejora en la estructura.	El código está completamente organizado en ramas feature/ y se documenta de manera clara y completa. La documentación técnica está actualizada, y se siguen las buenas prácticas de desarrollo en todo el proyecto.
--------------------------------	--	---	--	---

LOGROS MÍNIMOS NECESARIOS PARA PROGRAMACIÓN FULL STACK

1. Documentación y Justificación Tecnológica

- Documento técnico con justificación de PHP como backend
- Criterios de selección del SGBD (MySQL/MariaDB)
- Evaluación de Bootstrap y frameworks complementarios
- Selección de herramientas para control de versiones
- Recomendaciones de entornos de desarrollo

2. Modelado de Datos

- Diagrama Entidad-Relación (DER) normalizado hasta 3FN
- Esquema relacional con claves primarias y foráneas claramente identificadas
- Documentación de restricciones no estructurales derivadas de las reglas del juego
- Scripts DDL para creación de tablas, índices y relaciones
- Scripts DML para inserción de datos de prueba

3. Implementación Física de la Base de Datos

- Creación de la base de datos con estructura normalizada
- Configuración de usuarios y permisos básicos
- Implementación de consultas preparadas para prevención de SQL Injection
- Conexión funcional entre la aplicación y la base de datos

4. Interfaz de Usuario

- Maquetación HTML5 con etiquetas semánticas
- Estilización mediante CSS3 enfocada en usabilidad
- Implementación de Bootstrap 5.3 para diseño responsivo
- Adaptación para dispositivos móviles (320px - 576px) y tabletas (577px - 992px)
- Sistema de navegación funcional entre secciones

5. Funcionalidades Mínimas del Juego (Versión Individual)

- Tablero de juego personal con renderizado dinámico (JavaScript)
- Implementación de arrastrar y soltar (drag and drop) para colocar dinosaurios
- Validación de reglas del juego en el tablero personal
- Formulario de configuración con personalización de parámetros iniciales
- Sistema de persistencia de partidas individuales

6. Arquitectura de Software

- Implementación de estructura de tres capas:
 - Capa de Presentación: Interfaz de usuario (HTML, CSS, JavaScript)
 - Capa de Negocio: Lógica del juego, reglas y controladores
 - Capa de Datos: Acceso y persistencia en base de datos
- Programación Orientada a Objetos con separación clara de responsabilidades

7. Validación y Comunicación con el Servidor

- Validación dual de formularios (frontend con JavaScript, backend con PHP)
- Sistema de mensajes de error contextualizados
- Implementación de consultas asíncronas (Fetch/Async-Await) para:
 - Carga de dinosaurios disponibles
 - Guardado de progreso
 - Actualización del tablero

8. Control de Versiones y Repositorio

- Repositorio GitHub estructurado con buenas prácticas
- Convenciones de commits y mensajes documentadas
- Archivo README.md con instrucciones claras de instalación y uso
- Organización del código fuente y documentación técnica

9. Entorno de Desarrollo

- Configuración documentada del entorno (Apache, PHP, MySQL/MariaDB)
- Instrucciones verificables para replicar el entorno
- Configuración recomendada de IDE (Visual Studio Code)

8.3. CRITERIOS DE LOGRO PARA ADMIN. DE SISTEMAS OPERATIVOS

PUNTEO GENERAL DE REQUERIMIENTOS PARA PRIMERA ENTREGA

- Relevamiento del S.O. para el servidor
- Manual de Instalación del S.O Servidor
- Instalación y Configuración de la red
- Gestión y administración de paquetes del Sistema Operativo elegido

RÚBRICA DE EVALUACIÓN PARA PRIMERA ENTREGA

Criterios de evaluación	1 punto	4 puntos	7 puntos	10 puntos
Relevamiento del S.O. para el servidor	No realiza este punto para la entrega	Realiza un relevamiento mínimo de los sistemas operativos eligiendo uno para ser utilizado en el servidor.	Realiza un relevamiento y justifica el Sistema Operativo a utilizar en el servidor, no incluye plan de soporte técnico claro para orientar al usuario final.	Realiza un relevamiento exhaustivo y justifica el Sistema Operativo a utilizar en el servidor, incluyendo planes de soporte técnico para el usuario final.
Manual de Instalación del S.O Servidor	No realiza este punto para la entrega	Realiza un manual básico de instalación con capturas de pantallas no tan claras para el usuario, no se contemplan todas las partes de la instalación ni se explica el soporte	Realiza un manual de instalación básico con capturas claras para el usuario final, no brinda manual de soporte.	Realiza un manual de instalación para el servidor Linux aplicado en el proyecto con soporte técnico empresarial (SUSE Linux Enterprise Server, Red Hat Enterprise Linux, CentOS) para el usuario final.
Instalación y Configuración de la red	No realiza este punto para la entrega	Instala el servidor pero deja la red en formato DHCP para ser configurado más adelante	Instala el servidor y configura la red, pero no contempla los puertos necesarios que se deben de modificar para ser visible por la red en base al proyecto.	Configura la red en el servidor Gnu/Linux de manera estática para ser visible fácilmente desde la red. Configura los puertos del servidor para enviar información a través de la red
Gestión y	No realiza	Instala solo los	Instala y configura	Instala y configura

administración de paquetes del S.O	este punto para la entrega	paquetes por defecto en el sistema operativo.	los paquetes básicos del sistema operativo, no logra configurar el sistema para realizar un despliegue de la aplicación	de manera eficiente los paquetes necesarios para que el servidor pueda ser utilizado para hacer un despliegue inicial de la aplicación
------------------------------------	----------------------------	---	---	--

PUNTEO GENERAL DE REQUERIMIENTOS PARA SEGUNDA ENTREGA

- Configuración del servicio SSH en el servidor
- Definición de medios de respaldo a largo plazo y alta disponibilidad
- Rutinas de backup y scripts correspondientes
- Primera Versión del script del servidor

RÚBRICA DE EVALUACIÓN PARA SEGUNDA ENTREGA

Criterios de evaluación	1 punto	4 puntos	7 puntos	10 puntos
Configuración del servicio SSH	No realiza este punto para la entrega	Instala el servicio SSH en el servidor pero no aplica las configuraciones básicas	Instala el servicio SSH cliente/servidor y configura los puertos básicos	Configura el servicio SSH en el servidor ajustado a los requerimientos, siendo utilizado para realizar conexiones de manera remota generando llaves públicas distribuidas en las conexiones
Definición de respaldos	No realiza este punto para la entrega	Define de manera teórica los respaldos que serán realizados en el servidor	Define de manera teórica los respaldos e implementa de manera básica los script de automatización.	Define medios de respaldo a largo plazo y alta disponibilidad de los datos para evitar pérdidas de información
Rutinas de Backups y Scripts básicos	No realiza este punto para la entrega	Define de manera teórica las rutinas de backups pero no implementa de manera correcta los scripts de automatización	Define de manera teórica las rutinas de backups y implementa scripts básicos de automatización	Programa rutinas de backup, y sus correspondientes scripts, configurados de manera automática para minimizar pérdidas de datos.
Primer script de	No realiza este punto	Implementa una base del script	Implementa varios scripts para	Realizar una primera versión del

administración del servidor en formato modular.	para la entrega	del servidor pero no cumple con los criterios de modularización	administrar el servidor pero no logra realizar una correcta conexión entre ellos.	script de administración del servidor donde se contemplan los scripts trabajados en clase. Desarrolla el script de manera modular en partes pequeñas y reutilizables, lo que facilita su mantenimiento, depuración y escalabilidad
---	-----------------	---	---	---

PUNTEO GENERAL DE REQUERIMIENTOS PARA TERCERA ENTREGA

- Script final del servidor
- Gestión y administración de contenedores para despliegue de aplicaciones
- Generación del servidor de respaldo

RÚBRICA DE EVALUACIÓN PARA TERCERA ENTREGA

Criterios de evaluación	1 punto	4 puntos	7 puntos	10 puntos
Script final del servidor	No realiza este punto para la entrega	Implementa el script final del servidor con la gestión de usuarios y grupos completa	Implementa el script final del servidor con la gestión de usuarios, grupos y redes completa, quedan parcialmente completos el resto de los scripts	Implementa el script final del servidor de manera modular con todos los script necesarios vistos en el curso: <ul style="list-style-type: none"> • Gestión de usuarios • Gestión de grupos • Gestión de respaldos • Gestión de redes • Gestión de Bases de datos • Gestión de Firewall • Gestión de Logs del Sistema Para poder administrar de manera eficiente todo lo relacionado a la infraestructura.

Gestión y administración de contenedores para despliegue de aplicaciones	No realiza este punto para la entrega	Realiza un despliegue básico de la aplicación en un solo contenedor, no maneja volúmenes de datos	Realiza un despliegue parcial en dos contenedores donde se contempla la aplicación y el volumen de datos, no se gestiona la red de manera eficiente.	Gestiona y administra los contenedores del proyecto usando la herramienta docker-compose para hacer un despliegue de la aplicación del proyecto de manera eficiente.
Generación del servidor de respaldo	No realiza este punto para la entrega	Realiza el manual de instalación del servidor de respaldos pero no lo implementa	Realiza el manual y la instalación del servidor de respaldo pero no lo configura	Generará un servidor de respaldo de datos que permite realizar un backup íntegro del sistema como sistema de contingencia ante pérdidas importantes.

LOGROS MÍNIMOS NECESARIOS PARA ADMINISTRACIÓN DE SISTEMAS OPERATIVOS:

- Realiza un manual de instalación para el servidor Linux aplicado en el proyecto con soporte técnico empresarial (SUSE Linux Enterprise Server, Red Hat Enterprise Linux, CentOS) para el usuario final.
- Configura la red en el servidor Gnu/Linux de manera estática para ser visible fácilmente desde la red. Configura los puertos del servidor para enviar información a través de la red.
- Configura el servicio SSH en el servidor ajustado a los requerimientos para ser utilizado para realizar conexiones de manera remota
- Realizar una primera versión del script de administración del servidor donde se contemplan los scripts trabajados en clase. Desarrolla el script de manera modular en partes pequeñas y reutilizables, lo que facilita su mantenimiento, depuración y escalabilidad
- Implementa el script final del servidor de manera modular con todos los script necesarios vistos en el curso:
 - Gestión de usuarios
 - Gestión de grupos
 - Gestión de respaldos
 - Gestión de redes
 - Gestión de Bases de datos
 - Gestión de Firewall
 - Gestión de Logs del Sistema

Para poder administrar de manera eficiente todo el servidor.

8.4. CRITERIOS DE LOGRO PARA TUTORÍA DE PROYECTO UTULAB:

PRIMERA ENTREGA:

Documentación obligatoria:

- Nombre del equipo (único en todas las materias).
- Portada
- Índice
- Introducción
- Objetivos generales y específicos
- Justificación del proyecto
- Logo del equipo (diseño propio).
- Breve descripción del juego (ficha técnica: nombre, temática, tipo).
- Cronograma inicial del proyecto (aunque sea tentativo).

Prototipos iniciales:

- Diseño del tablero de juego.
- Diseño del dado.
- Bocetos de UI/UX digital en herramientas gráficas. (Tablero invierno y verano)

RÚBRICA 1° ENTREGA – (máximo 10 puntos)

Criterios de evaluación:

Criterio	Excelente (2 pts)	Satisfactorio (1.5 pts)	Parcial (1 pt)	No cumple (0 pt)
Identidad del equipo (nombre + logo)	Nombre original y logo creativo, bien presentado.	Presentes pero simples.	Uno de los dos presentes o poco trabajado.	No entregó.
Ficha técnica del juego (nombre, temática, tipo)	Clara, completa, coherente.	Incompleta pero comprensible.	Poco clara.	No entregó.
Cronograma inicial	Realista, bien organizado.	Tentativo pero funcional.	Muy básico o sin fechas.	No entregó.
Prototipos iniciales (tablero y dado)	Presentes, creativos y funcionales.	Simple pero comprensibles.	Incompletos o mal presentados.	No entregó.
Pantallazos UI/UX (si aplica)	Claros, funcionales, con propuesta visual.	Presentes, básicos.	Muy preliminares o poco organizados.	No entregó.

Nota total = /10

SEGUNDA ENTREGA:

Carpeta parcial (impresa y firmada):

- Marco teórico
- Planificación y cronograma actualizado

- Desarrollo del proyecto (avance técnico y organizativo)
- Resultados esperados
- Conclusiones
- Anexos

Desarrollo técnico:

- Primeros diseños en 3D (fichas de dinosaurios).
- Diseño final tableros
- Merchandising inicial (remeras, tarjetas, etc. si aplica).
- Posibles enlaces, QR o capturas de funcionalidades ya desarrolladas.

RÚBRICA 2° ENTREGA – (máximo 10 puntos)

Criterios de evaluación:

Criterio	Excelente (2 pts)	Satisfactorio (1.5 pts)	Parcial (1 pt)	No cumple (0 pt)
Carpeta parcial teórica (con todos los apartados requeridos)	Organizada, bien redactada y están al día con el material.	Casi completa.	Varias secciones incompletas.	No entregó.
Planificación y cronograma actualizado	Claro, detallado y realista.	Aceptable, con algunos detalles.	Básico o sin actualización.	No entregó.
Desarrollo del proyecto (avance técnico)	Avance claro, con documentación.	Avance visible, pero poco claro.	Muy poco desarrollo.	Sin evidencia.
Diseños 3D preliminares + UI/UX mejorado	Claros y funcionales.	Presentes pero básicos.	Muy preliminares.	No entregó.
Merchandising y pruebas funcionales (QR, enlaces, etc.)	Bien presentado, coherente con la propuesta.	Presentes, pero simples.	Muy básico o incompleto.	No entregado

Nota total = /10

TERCERA ENTREGA:

3° ENTREGA FINAL

Carpeta final (impresa y firmada):

- Versión final y corregida de todos los apartados anteriores.
- Incorporar mejoras y observaciones recibidas en la segunda entrega.

Producto final:

- Diseño final 3D (tablero, fichas, dado).
- Diseño final UI/UX (pantallazos mejorados).
- Logo definitivo impreso.
- Merchandising completo.

RÚBRICA 3° ENTREGA FINAL (máximo 10 puntos)

I.S.B.O. - IAE Montevideo - ITI CETP
Proyecto de Egreso - BT - Tecnologías de la Información - 2025

Criterios de evaluación:

Criterio	Excelente (2 pts)	Satisfactorio (1.5 pts)	Parcial (1 pt)	No cumple (0 pt)
Carpeta final corregida y firmada	Muy completa, con mejoras claras.	Corregida con pocos errores.	Pocas mejoras.	No entregada.
Diseño final 3D (tablero, fichas, dado)	Completo, estético, funcional.	Presentes, con detalles a mejorar.	Incompletos o poco funcionales.	No entregado.
Diseño final UI/UX	Coherente, atractivo, funcional.	Presentes, básicos.	Muy preliminares o incoherentes.	No entregado.
Merchandising final	Creativo y bien ejecutado.	Presente, simple.	Muy básico.	No entregado.
Incorporación de mejoras sugeridas	Todas incorporadas correctamente.	Algunas incorporadas.	Mínimas mejoras.	No aplicó ninguna mejora.

Nota total = /10

LOGROS MÍNIMOS NECESARIOS PARA TUTORÍA DE PROYECTO UTULAB:

- Nombre del equipo.
- Logo definitivo.
- Diseño final 3D (tablero, fichas, dado).
- Carpeta final (impresa y firmada)

REQUISITOS MÍNIMOS PARA APROBAR (Obligatorios, sin puntaje)

Requisito	Cumple (✓)	No cumple (✗)
Nombre del equipo presente en todas las entregas		
Logo definitivo entregado		
Diseño final 3D del tablero, fichas y dado		
Carpeta final completa, impresa y firmada		

Si alguno de estos requisitos no se cumple, no se aprueba la unidad curricular.

8.5. CRITERIOS DE LOGRO PARA EMPRENDEDURISMO Y GESTIÓN

PRIMERA ENTREGA

- Nombre de la Empresa, logo y fundamentos (psicología del color, las formas y tipografía).
- Presentación, propósitos de la empresa y Objetivos Generales.
- Misión y Visión.
- Valores de la Organización.
- Habilidades de cada integrante del equipo (formato selfie emprendedora).

SEGUNDA ENTREGA

- Forma jurídica seleccionada y trámites de apertura.
- Ubicación de la empresa y Análisis del entorno.
- Estudio de Mercado, Matriz FODA.
- Plan de Marketing.

TERCERA ENTREGA

- Plan de recursos necesarios para puesta en marcha (humanos y materiales).
- Plan de recursos financieros (Estudio de propuestas de fortalecimiento y financiamiento para las Pymes).
- Viabilidad del Proyecto.
- Costos.

ANEXOS:

- Formulario de registro frente al Estado: BPS, DGI, BSE, etc.
- Contrato o Estatuto de la empresa.

LOGROS MÍNIMOS NECESARIOS PARA EMPRENDEDURISMO Y GESTIÓN:

- Los planteará el docente en clase en base a lo que logren avanzar en el proyecto.

8.6. CRITERIOS DE LOGRO PARA FÍSICA:

Dinoparque en Acción: Física en el Mundo de Draftosaurios

El proyecto busca integrar conocimientos de Física Mecánica en la construcción y análisis de un parque de dinosaurios inspirado en el juego Draftosaurios. A través del diseño de atracciones, sistemas de transporte y estructuras de seguridad, los estudiantes aplicarán conceptos fundamentales de la física a situaciones reales simuladas, promoviendo el pensamiento crítico, la creatividad y el trabajo colaborativo.

PRIMERA ENTREGA: Idea del proyecto y planificación inicial Rúbrica de Evaluación

I.S.B.O. - IAE Montevideo - ITI CETP
 Proyecto de Egreso - BT - Tecnologías de la Información - 2025

Criterio	Excelente (5)	Bueno (4)	Regular (3)	Insuficiente (1-2)
Claridad de la idea	La idea es clara, creativa y bien desarrollada	La idea es clara, con algunos detalles por mejorar	La idea está poco desarrollada o es confusa	La idea no está definida o no se entiende
Organización y planificación	Hay un cronograma o planificación detallada y coherente	Hay planificación, pero con detalles menores a ajustar	Falta organización o cronograma claro	No hay planificación evidente

SEGUNDA ENTREGA: Desarrollo del Parque y Diseño de Elementos
Rúbrica de Evaluación

Criterio	Excelente (5)	Bueno (4)	Regular (3)	Insuficiente (1-2)
Aplicación de conceptos físicos	Se aplican con precisión y creatividad	Se aplican con precisión básica	Se aplican con errores o faltan algunos	No hay aplicación clara
Originalidad y diseño	Diseño innovador, bien argumentado y visualmente claro	Diseño funcional y correcto	Diseño básico o poco detallado	Diseño ausente o muy confuso

TERCERA ENTREGA: Prototipos o Simulación Física
Rúbrica de Evaluación

Criterio	Excelente (5)	Bueno (4)	Regular (3)	Insuficiente (1-2)
Calidad del prototipo o simulación	Prototipo funcional y bien logrado, con explicación física clara	Prototipo funcional, con explicación suficiente	Prototipo con fallos o poco detallado	Prototipo ausente o muy incompleto
Razonamiento y cálculos físicos	Cálculos correctos, bien explicados y aplicados	Cálculos básicos, con algunas omisiones	Cálculos con errores importantes	Sin cálculos o sin justificación física

LOGROS MÍNIMOS NECESARIOS PARA FÍSICA:

Aplicar conceptos de cinemática, dinámica, energía y estructuras al diseño de elementos funcionales o simulados del parque de dinosaurios, reforzando la comprensión del movimiento y la mecánica en contextos lúdicos y aplicado

8.7 CRITERIOS DE LOGROS PARA SOCIOLOGÍA

El proyecto busca integrar conocimientos de las unidades curriculares de informática y sociología, en el cual se visualice el desarrollo de habilidades prácticas y su importancia en la inserción en el medio. A través de la investigación de campo, la recolección y análisis de datos, los estudiantes pueden comprender mejor las dinámicas sociales y problemáticas que les afectan. Permite fomentar el pensamiento crítico, al analizar la información desde diferentes perspectivas y tomar decisiones basadas en evidencias, lo que fortalecen su capacidad de análisis y reflexión. Dicho proyecto va a ser una oportunidad para involucrarse en la comunidad y contribuir en la misma. Se busca con el mismo, estimular la creatividad y la búsqueda de soluciones innovadoras a problemas concretos, en donde es esencial el trabajo colaborativo. De esta forma les permite enfrentarse a situaciones del mundo laboral en donde se requiere trabajar en equipo, gestionar proyectos y comunicarse asertivamente. Por lo tanto, va contribuir a preparar a los estudiantes para el futuro donde la tecnología y la sociedad están cada vez más interconectadas.

Competencias a trabajar durante el desarrollo del proyecto

- Pensamiento crítico.
- Pensamiento computacional.
- Iniciativa y orientación a la acción.
- Relación con los otros.
- Ciudadanía local, global y digital.

PRIMERA ENTREGA

- Identificación de integrantes del grupo.
- Planteo de los objetivos de investigación. Formular un objetivo general y dos específicos. (en forma de pregunta)
- Fundamentación de la importancia del proyecto. Relevancia de acuerdo al contexto actual y a la orientación elegida.
- Marco teórico: citar las fuentes bibliográficas, teorías y los sitios web que se van utilizando durante la realización del proyecto como base para el desarrollo del mismo.

PRIMERA ENTREGA: rúbrica de evaluación

I.S.B.O. - IAE Montevideo - ITI CETP
 Proyecto de Egreso - BT - Tecnologías de la Información - 2025

Criterio	Muy bueno (5)	Bueno (4)	Regular (3)	No cumple (1-2)
Planteo de los objetivos	Muy buen desarrollo y claridad de los mismos.	Buen planteamiento presentando algunas correcciones a realizar para mejorarlos.	No presentan claridad, y los mismos están confusos	Errores graves al plantarlos o no se realizan los mismos.
Fundamentación del proyecto	Muy buen desarrollo de la misma con coherencia y bien argumentada.	Presenta algunos detalles a mejorar.	Falta claridad y respaldo teórico.	No realiza una fundamentación correcta o la misma no existe.
Marco teórico	Muy buen desarrollo del mismo, en el cual consta todos los sitios web, bibliografía, antecedentes y teorías que se utilizaron para la elaboración del respaldo del proyecto.	Escaso desarrollo del mismo, en el cual no consta todos los sitios web, bibliografía, antecedentes y teorías que se utilizaron para la elaboración del respaldo del proyecto.	Falta desarrollo del mismo, en el cual consta errores en la búsqueda de los sitios web, bibliografía, antecedentes y teorías que se utilizaron para la elaboración del respaldo del proyecto	Errores graves en el desarrollo del mismo, ya que se presentaron incoherencias en la selección o no se presentó el marco teórico.

SEGUNDA ENTREGA

- Formulación de objetivos y resultados esperados.
- Elección de metodología de investigación. Aplicación de al menos una técnica de recolección de datos (encuesta y/o entrevista). Para las encuestas aclarar a quiénes se encuestará y qué se busca averiguar. Para las entrevistas explicar cuál fue el criterio adoptado para la elección de los informantes.
- Análisis de los resultados obtenidos en la aplicación de las herramientas metodológicas (encuestas y/o entrevistas)

- Identificación de limitaciones del proyecto, es decir aquellas dificultades con las que creen se van a encontrar a lo largo de la realización del proyecto (bibliografía, recursos materiales, organización, tiempos, implementación de la aplicación de las herramientas de recolección de datos etc.).
- Continuar citando las fuentes utilizadas y consultadas (las que forman el marco teórico).

SEGUNDA ENTREGA: rúbrica de evaluación

Criterio	Muy bueno (5)	Bueno (4)	Regular (3)	No cumple (1-2)
Formulación de los resultados esperados en relación de los objetivos.	Muy buen desarrollo de los mismos con coherencia.	Presenta algunos detalles a mejorar.	Los resultados están confusos y se presenta un escaso desarrollo de los mismos al explicarlos.	Los resultados no se relacionan con los objetivos o los mismos no aparecen.
Elección de la metodología de investigación y análisis de los resultados	Muy buen planteamiento de las técnicas de investigación social al igual que el análisis de sus resultados.	Presenta algunos detalles a mejorar en cuanto al planteamiento de técnicas o errores en el análisis de los resultados	Falta claridad en el planteamiento de las técnicas y los resultados presentan errores graves.	No se presentaron técnicas de investigación social ni resultados.
Identificación de limitaciones del proyecto	Muy buena visualización y planteamiento de las dificultades con las cuales se encontraron.	Escasa presentación al igual que su planteamiento	Se presentan algunos errores en el planteamiento.	No quedaron claras las dificultades planteadas o las mismas no eran correctas en relación con el proyecto.

Marco teórico (se continúa citando fuentes)	Muy buen desarrollo del mismo, en el cual consta todos los sitios web, bibliografía, antecedentes y teorías que se utilizaron para la elaboración del respaldo del proyecto.	Escaso desarrollo del mismo, en el cual no consta todos los sitios web, bibliografía, antecedentes y teorías que se utilizaron para la elaboración del respaldo del proyecto.	Falta desarrollo del mismo, en el cual constan algunos errores en la búsqueda de los sitios web, bibliografía, antecedentes y teorías que se utilizaron para la elaboración del respaldo del proyecto.	Se presentan errores graves, pues no se relaciona el marco teórico con el proyecto o el mismo no consta.
---	--	---	--	--

TERCERA ENTREGA

- Verificación del cumplimiento de los objetivos que se plantearon en la primera y segunda entrega, en la cual se señalan las dificultades con las cuales se encontraron durante el desarrollo del proyecto, Explicar si las mismas dificultades pudieron ser solucionadas y cómo lo lograron.
- Explicar la importancia para el ámbito social de la realización de dicho proyecto.
- Continuar citando las fuentes utilizadas y consultadas.

TERCERA ENTREGA: rúbrica de evaluación

Criterio	Muy bueno (5)	Bueno (4)	Regular (3)	No cumple (1-2)
Verificación y cumplimiento de objetivos y sus correspondientes problemas	Muy buen desarrollo de los mismos con muy buena claridad y justificación.	Buen desarrollo de la verificación y cumplimiento de los objetivos, aunque se podría mejorar su desarrollo.	Falta señalar algunos puntos del cumplimiento de los objetivos.	La idea no está definida o no se entiende

Justificación de la importancia social del proyecto	Muy buena justificación utilizando la aplicación de herramientas de sociología.	Aceptable la justificación, aunque debería haber profundizado con las herramientas de sociología.	Falta claridad al justificar y presenta algunos errores	Mal justificado ya que presenta incoherencias o la misma no se realiza
Cita de fuentes utilizadas para sustentar y fundamentar el proyecto (marco teórico)	Muy buen desarrollo de la misma.	Escaso desarrollo de la misma, en la cual no constan todas las fuentes más importantes relacionadas al proyecto.	Falta desarrollo de la misma, en el cual constan algunos errores de búsqueda.	Se presentan errores graves, pues no se relaciona el marco teórico con el proyecto o el mismo no consta.

REQUERIMIENTOS MÍNIMOS DE SOCIOLOGÍA

Mínimo (5 puntos):

- El docente de Sociología definirá los requerimientos mínimos en clase con sus alumnos en base al contenido que logren avanzar de las tres entregas.

9. Pautas obligatorias para la presentación del proyecto

COMPOSICIÓN DE LAS CARPETAS POR ASIGNATURA QUE SE DEBERÁ ENTREGAR

Carátula:

En esta se deberá leer en forma clara y respetando los formatos establecidos lo siguiente:

- Nombre del proyecto.
- Nombre del grupo.
- Nombre de cada uno de los integrantes del mismo.
- Nombre de los docentes de las materias (se sugiere pedirle con antelación el nombre completo del docente).
- Fecha de culminación del trabajo.
- Número de entrega (Primera, Segunda, Tercera).

Índice:

Se colocará inmediatamente después de la carátula, ocupando todas las hojas que sean necesarias, se deberá de respetar las normas APA.

Debe abarcar todo el contenido de la carpeta, incluyéndose a sí mismo y a los Anexos.

Desarrollo del proyecto:

En este espacio el grupo de proyecto deberá relatar con sus propias palabras, los objetivos a cumplir, de qué forma y mediante qué medios los realizará, presentando una argumentación técnica acorde a lo solicitado en la letra del proyecto.

Llevará los títulos y subtítulos que el grupo considere necesarios, de acuerdo a la temática que se esté tratando.

ASPECTOS FORMALES DEL TEXTO

Papel:

- Debe usarse tamaño A4 y escribirse por una sola cara de la hoja, con interlineado 1.5.
- El color y la calidad del papel deben facilitar la impresión y la lectura.

Márgenes:

Se solicita los siguientes márgenes siguiendo el estándar APA:

- Izquierdo 2.54 cm Derecho 2.54 cm
- Superior 2.54 cm Inferior 2.54 cm

Numeración:

- Las páginas deberán numerarse en forma consecutiva, comenzando con la página que contiene el índice. Dicha numeración será establecida sobre el ángulo inferior derecho de cada página.
- Si existe más de un Anexo, cada uno de ellos será nombrado como Anexo 1, Anexo 2,..., Anexo N.
- Las páginas de los Anexos tienen su propia numeración, comenzando en 1 para cada uno de ellos. Se enumeran entendiéndose por el número de Anexo.
- En Anexo deberá numerarse, con números romanos.

Tipo de Letra:

- Los títulos tendrán un tamaño de letra de 16 Negrita Subrayado Tipo Arial - Times New Roman
- Los subtítulos tendrán un tamaño de 14 Negrita Tipo Arial - Times New Roman
- Los párrafos tendrán un tamaño de letra de 12. Tipo Arial - Times New Roman
- El trabajo debe redactarse de manera impersonal, en lenguaje claro y lo más conciso posible; con una correcta sintaxis y ortografía.

Encabezado de página:

El encabezado de página debe incluir nombre del grupo, logo (centrado) y fecha de la elaboración del trabajo.

Pies de página:

El pie de página debe incluir el nombre del proyecto, nombre de la escuela (siglas), el grupo al que pertenece.

Referencias:

Se deberá de realizar citas en formato APA al pie de página y se deberá de citar al final la bibliografía/webgrafía y referencias bibliográficas.

Respaldo USB:

En la tercera entrega TODAS las entregas deberán de ser respaldadas en un **medio USB** y deberá ser entregado con las carpetas. Será responsabilidad del grupo guardar una copia para sus integrantes (SOLAMENTE UN MEDIO USB POR EQUIPO Y POR ENTREGA).

Carpeta:

La documentación deberá ser entregada en una carpeta POR ASIGNATURA, cuya carátula deberá cumplir lo especificado para la realización de ésta.

Al final de la carpetas se debe agregar una hoja con título "Hoja Testigo", la cual verifica lo entregado, como también las fotos de los integrantes para facilitar la gestión de corrección y recepción de la misma.

NO SE ACEPTARÁN:

- Hojas sueltas como parte de la entrega
- Carpetas que no cumplan con lo establecido
- Archivos editables, se debe entregar en formato PDF

10. ANEXOS

Las carátulas de las carpetas, los acuses de recibo, la máquina virtual para trabajar con sistemas operativos y el reglamento de pasaje de grado, estarán subidos a la web oficial de la Institución en la sección de proyectos:

- <https://isbo.utu.edu.uy/bt-tecnologias-de-la-informacion>