}

**Guía1. Definición Proyecto APT**

**Asignatura Capstone**

1. **PARTE I**

|  |
| --- |
| **1. Antecedentes Personales** |
| A continuación, se presenta una tabla en la que debes completar la información solicitada. |

|  |  |
| --- | --- |
| Nombre estudiante | **Rafael Ancalipe y Bruno Roja** |
| Rut | 20.776.088-9 \_\_ 20.972.173-2 |
| Carrera | Ingeniería Informática |
| Sede | Padre Alonso de Ovalle |

|  |
| --- |
| **2. Descripción Proyecto APT** |
| En la descripción debes señalar brevemente el nombre de tu proyecto APT y las competencias del perfil de egreso que vas a poner en práctica. Si en tu carrera están definidas las áreas de desempeño, también menciona a qué áreas de desempeño está vinculado el proyecto. |

|  |  |
| --- | --- |
| Nombre del proyecto | Study Chess |
| Área (s) de desempeño(s) | **Área(s) de desempeño(s) abordadas por el Proyecto APT**   1. **Desarrollo e Ingeniería de Software**  * Justificación: diseño e implementación del frontend (React), componentes del tablero, lógica de negocio y pruebas del prototipo.  1. **Arquitectura de Sistemas y Computación en la Nube**  * Justificación: definición de la arquitectura cliente/servidor, decisiones de despliegue (hosting estático, funciones serverless) y comunicación (REST/WebSocket).  1. **Inteligencia Artificial / Sistemas de Análisis**  * Justificación: integración y uso de un motor de análisis (Stockfish) y diseño de la capa de interpretación que traduce evaluaciones en explicaciones pedagógicas.  1. **Bases de Datos y Persistencia**  * Justificación: diseño e implementación del almacenamiento de estudios y perfiles de usuario (Firestore u otra solución), import/export de PGN y manejo de datos de uso.  1. **Interacción Persona-Computador (UI/UX)**  * Justificación: diseño de la interfaz del tablero, historial anotado y paneles de análisis para que la información técnica sea comprensible y utilizable por aprendices.  1. **Seguridad Informática y Buenas Prácticas**  * Justificación: implementación de autenticación segura (Firebase Auth o equivalente), uso de HTTPS y consideraciones básicas de protección de datos de usuarios.  1. **Gestión de Proyectos y Metodologías Ágiles**  * Justificación: planificación por sprints, priorización de MVP, control de versiones y documentación del proceso, integrando prácticas de trabajo en equipo.  1. **Evaluación de Software y Usabilidad**  * Justificación: diseño y ejecución de pruebas funcionales y un piloto con usuarios para validar la utilidad pedagógica y la usabilidad de la plataforma. |
| Competencias | Competencias que aborda el Proyecto APT   1. **Diseñar y desarrollar sistemas de software**  * Cómo se evidencia: implementación del frontend en React, componentes del tablero y la lógica de negocio del análisis. Código en repositorio, entregable funcional (MVP) y documentación técnica.  1. **Arquitectura de sistemas y despliegue en la nube**  * Cómo se evidencia: definición y justificación de la arquitectura cliente/servidor, uso de hosting (Vercel/Netlify) y funciones serverless; diagrama de arquitectura y guía de despliegue.  1. **Aplicación de algoritmos e integración de motores de análisis**  * Cómo se evidencia: integración de Stockfish para evaluación por jugada, diseño de la capa de interpretación heurística y pruebas de consistencia entre evaluaciones.  1. **Gestión de bases de datos y persistencia de información**  * Cómo se evidencia: diseño e implementación del almacenamiento de estudios y perfiles de usuario (Firestore u otra DB), esquemas de datos y ejemplos de consultas/backup.  1. **Interacción persona-computador (UI/UX)**  * Cómo se evidencia: diseño y prototipado del tablero, historial anotado y paneles explicativos; pruebas de usabilidad y resultados del piloto sobre facilidad de uso.  1. **Seguridad informática y buenas prácticas de autenticación**  * Cómo se evidencia: implementación de autenticación segura (Firebase Auth u alternativa), uso de HTTPS, manejo básico de permisos y documentación de consideraciones de privacidad.  1. **Evaluación y validación de software**  * Cómo se evidencia: pruebas unitarias/funcionales, casos de prueba para análisis de partidas, y reporte del piloto (feedback cualitativo y métricas de uso).  1. **Gestión de proyectos y metodologías ágiles**  * Cómo se evidencia: planificación por sprints, backlog, control de versiones (Git), entregas incrementales y registro de actividades (issues, milestones).  1. **Comunicación técnica y documentación profesional**  * Cómo se evidencia: elaboración de la memoria de tesis, documentación técnica, manual de usuario y presentación para la defensa/propuesta del proyecto.  1. **Ética profesional y responsabilidad en el uso de datos**  * Cómo se evidencia: consideración documental sobre privacidad de usuarios, uso responsable de resultados automáticos (evitar recomendaciones dañinas) y citación de fuentes (Stockfish, librerías).  1. **Aprendizaje autónomo y actualización tecnológica**  * Cómo se evidencia: selección e integración de tecnologías actuales (React, WASM, Firebase), justificación de decisiones y plan de mejora/continuidad del proyecto. |

|  |
| --- |
| **3. Fundamentación Proyecto APT** |
| A continuación, se presentan distintos campos que debes completar con la información solicitada. Esta sección busca que describas en detalle tu proyecto y justifiques su relevancia y pertinencia. |

|  |  |
| --- | --- |
| Relevancia del proyecto APT | El proyecto **Study Chess** surge como una respuesta a la creciente necesidad de contar con herramientas digitales que faciliten el aprendizaje del ajedrez de manera interactiva y pedagógica. Si bien existen múltiples plataformas que permiten jugar en línea, pocas se enfocan realmente en el **estudio y la comprensión de las partidas**, aspecto clave para el progreso de jugadores principiantes e intermedios.  La problemática principal que se busca abordar es la **falta de acceso a un sistema de entrenamiento personalizado** que combine análisis computacional avanzado —a través de motores como Stockfish— con explicaciones pedagógicas adaptadas al nivel del usuario. En el contexto actual, los jugadores que desean mejorar dependen de entrenadores particulares o de plataformas que presentan evaluaciones técnicas difíciles de interpretar sin guía.  Este proyecto se sitúa en **Chile**, en un contexto educativo y tecnológico en expansión, donde el ajedrez ha sido reconocido como una herramienta de desarrollo cognitivo en colegios, universidades y talleres comunitarios. El impacto principal se dirige a **jóvenes y adultos principiantes o intermedios**, quienes buscan mejorar su nivel de juego y desarrollar habilidades como la toma de decisiones, la concentración y el pensamiento crítico.  La **relevancia para el campo laboral de la ingeniería informática** radica en que el proyecto integra diversas competencias profesionales: desarrollo web con React, integración de inteligencia artificial (motor Stockfish), uso de bases de datos en la nube (Firebase) y aplicación de metodologías ágiles. Esto lo convierte en un ejemplo práctico de cómo un profesional del área puede **diseñar soluciones tecnológicas innovadoras** que responden a problemáticas reales de la sociedad.  El aporte de valor del proyecto se refleja en dos dimensiones:   * **Educativa**, al entregar una herramienta accesible que fomenta el aprendizaje autónomo y la práctica del ajedrez como disciplina cognitiva. * **Profesional**, al demostrar la capacidad del egresado de aplicar conocimientos técnicos en el desarrollo de software orientado a la innovación y con impacto social. |
| Descripción del Proyecto APT | Objetivo general:  Desarrollar un prototipo funcional de plataforma web —Study Chess— que proporcione análisis de partidas con comentarios pedagógicos automatizados para facilitar el aprendizaje autónomo y la mejora técnica de jugadores de ajedrez.  Descripción breve del proyecto:  Study Chess consiste en una aplicación web que combina un tablero interactivo con un motor de análisis (Stockfish) y una capa de interpretación pedagógica. El sistema permite cargar o reproducir partidas, obtener evaluación por jugada, visualizar variantes alternativas y recibir explicaciones breves y accionables (p. ej. fallo táctico, pérdida de iniciativa, oportunidad de combinación). Además incorpora un sistema de usuario para guardar estudios y realizar seguimiento del progreso.  Cómo se abordará la problemática:  Para transformar las salidas técnicas del motor en retroalimentación útil se seguirá un enfoque práctico y modular:   * Integración del motor: usar Stockfish (WASM en cliente o servicio backend) para obtener evaluaciones y variantes por jugada. * Capa pedagógica: aplicar reglas heurísticas que detecten cambios relevantes en la evaluación (umbral de pérdidas/ganancias de material, caídas de evaluación por errores tácticos, cambios posicionales) y generen explicaciones en lenguaje natural orientadas al aprendizaje. * Interfaz y usabilidad: diseñar un tablero interactivo y un historial anotado que facilite la comprensión de la evolución de la partida y de las recomendaciones. * Validación: realizar pruebas funcionales y un piloto con usuarios (niveles principiantes–intermedio) para evaluar claridad y utilidad de las explicaciones y ajustar las reglas heurísticas. * Alcance práctico: priorizar un MVP (tablero, análisis por jugada, generación de comentarios y guardado de estudios) y dejar mejoras avanzadas (recomendadores ML, app nativa) como trabajo futuro. |
| Pertinencia del proyecto con el perfil de egreso | El Proyecto APT Study Chess se alinea directamente con el perfil de egreso de la carrera de Ingeniería Informática porque articula conocimientos técnicos, habilidades de diseño y desarrollo de software, capacidad de análisis y juicio profesional, y actitudes éticas y de aprendizaje continuo que el perfil exige. El perfil de egreso busca formar profesionales capaces de concebir, diseñar, implementar y evaluar soluciones informáticas completas que respondan a problemas reales; Study Chess constituye un caso práctico claro: la creación de una plataforma educativa que integra software cliente, servicios de análisis y criterios pedagógicos para resolver la problemática de la interpretación y uso didáctico del output de motores de ajedrez.  A continuación se explicita la relación entre el proyecto y las competencias seleccionadas, y por qué cada una es necesaria para abordar la problemática planteada:   1. Diseñar y desarrollar sistemas de software  * El proyecto exige la concepción y construcción de una aplicación web funcional (MVP) que incluya interfaz, lógica de negocio y módulos de análisis. Esta competencia es esencial para transformar la idea pedagógica en un sistema operativo, modular y mantenible. * Evidencia: repositorio con código, entregable desplegado y documentación técnica.  1. Arquitectura de sistemas y despliegue en la nube  * Decidir cómo se distribuyen las responsabilidades entre cliente y servidor (ejecución de Stockfish en WASM vs servicio), elegir hosting y funciones serverless son decisiones de arquitectura cruciales para la viabilidad técnica y el rendimiento del sistema.  1. Aplicación de algoritmos e integración de motores de análisis  * Integrar Stockfish y diseñar la capa que interpreta sus evaluaciones requiere comprensión de algoritmos, estructuras de datos (representación de tableros/partidas) y técnicas para traducir salidas numéricas en explicaciones. Es el núcleo técnico que permite resolver la carencia pedagógica detectada. * Evidencia: módulo de integración con Stockfish, tests de consistencia de evaluaciones y heurísticas de interpretación.  1. Gestión de bases de datos y persistencia de información  * Guardar estudios, perfiles y registros de uso exige un diseño de persistencia coherente que permita recuperar sesiones, analizar progreso y sostener la biblioteca de estudios del usuario. * Evidencia: esquema de Firestore/DB, ejemplos de consultas y backup.  1. Interacción persona-computador (UI/UX)  * La problemática central (hacer interpretables las salidas del motor) es, en gran parte, un problema de comunicación entre máquina y usuario; una interfaz bien diseñada es necesaria para que las explicaciones y los controles sean intuitivos y efectivos. * prototipos UI, pruebas de usabilidad y encuestas del piloto.  1. Seguridad informática y buenas prácticas  * La gestión de cuentas de usuario y datos requiere asegurar confidencialidad e integridad; además, buenas prácticas aumentan la confiabilidad del servicio. Aunque el proyecto no es crítico en términos de seguridad a gran escala, la competencia es necesaria para garantizar una entrega profesional. * Evidencia: uso de Firebase Auth/HTTPS y documentación de consideraciones de privacidad.  1. Evaluación y validación de software  * Validar que las explicaciones sean útiles y que el análisis sea coherente con el motor requiere métodos de prueba (unitarias, funcionales) y evaluación con usuarios reales. Esta competencia permite medir el cumplimiento del objetivo pedagógico. * Evidencia: suite de pruebas, protocolo del piloto y resultados del cuestionario.  1. Gestión de proyectos y metodologías ágiles  * Para mantener el alcance controlado (MVP), priorizar tareas y documentar el proceso de desarrollo durante la tesis, se requiere la competencia de gestión de proyectos. Permite entregar en tiempo y con calidad. * Evidencia: backlog, sprints planificados, registro de issues y entregas.  1. Comunicación técnica y documentación profesional  * La redacción de la memoria, manuales de usuario y documentación de diseño son esenciales para la evaluación académica y para que terceros puedan entender y reutilizar el sistema. * Evidencia: el propio informe de tesis, manual de usuario y documentación del repositorio.  1. Ética profesional y responsabilidad en el uso de datos  * interacción con datos de usuarios y la emisión de recomendaciones automáticas requieren una aproximación responsable (transparencia, limitaciones del sistema, consentimiento). Esta competencia asegura que la solución se implemente con criterios éticos. * Evidencia: secciones de privacidad y limitaciones en la documentación. |
| Relación con los intereses profesionales | Relación con los intereses profesionales  Mi interés profesional se centra en el desarrollo de soluciones de software orientadas a la enseñanza, análisis de datos y experiencia de usuario, con un enfoque en tecnologías web modernas, metodologías ágiles (Scrum) y buenas prácticas de seguridad informática. Además, me interesa la integración de inteligencia artificial para apoyar la toma de decisiones y el aprendizaje autónomo en entornos educativos o lúdicos.   1. El Proyecto APT Study Chess refleja directamente estos intereses profesionales en varios aspectos:  * Desarrollo de software e integración de tecnologías modernas: La construcción de la plataforma web en React, la integración de Stockfish para análisis de partidas y el uso de Firebase para autenticación y persistencia, me permite profundizar en tecnologías que forman parte de mi campo de interés profesional. * Aprendizaje automático y sistemas de recomendación: Aunque no se desarrolla un motor ML avanzado, la capa de interpretación heurística de las evaluaciones del motor me introduce al diseño de sistemas que transforman datos técnicos en retroalimentación útil para los usuarios, lo que conecta con mi interés en inteligencia artificial aplicada al aprendizaje.  1. Metodologías ágiles: La planificación y ejecución del proyecto siguiendo principios de Scrum (sprints, MVP, backlog y entregas incrementales) permite reforzar habilidades de gestión de proyectos y trabajo colaborativo, competencias clave en mi desarrollo profesional.  * Ciberseguridad y buenas prácticas: La implementación de autenticación segura y la consideración de privacidad de datos fortalecen mi conocimiento y experiencia en seguridad informática aplicada a proyectos reales.  1. Realizar este Proyecto APT contribuye a mi desarrollo profesional al permitirme:  * Aplicar de manera práctica los conocimientos adquiridos en la carrera, integrando desarrollo frontend, backend, bases de datos, análisis de datos y UI/UX. * Experimentar con metodologías ágiles y herramientas de gestión de proyectos, fortaleciendo mis capacidades como futuro Scrum Master o líder técnico. * Profundizar en la interacción entre inteligencia artificial y experiencia del usuario, un área de gran relevancia para proyectos educativos y de ciberseguridad. * Generar un prototipo funcional que evidencia competencias técnicas y de gestión, constituyendo un portafolio tangible que respalda mi perfil profesional. |
| Factibilidad de desarrollo del Proyecto APT | Factibilidad de desarrollo del Proyecto APT  El desarrollo del Proyecto APT Study Chess es factible considerando los recursos disponibles, el tiempo del semestre y las condiciones de la asignatura.   1. Duración del semestre y horas asignadas a la asignatura:  * La planificación del proyecto se ajusta al semestre académico y a las horas asignadas al taller de proyectos, organizando el trabajo en sprints semanales para priorizar el MVP (tablero interactivo, análisis por jugada, biblioteca de estudios y generación de explicaciones pedagógicas). Esto permite avanzar progresivamente y entregar resultados funcionales dentro del plazo.  1. Materiales requeridos:  * Software: React, Firebase, Node.js, Stockfish (WASM o servicio). * Hardware: computadora personal con capacidad de desarrollo web, acceso a internet para despliegue y pruebas en línea. * Recursos adicionales: documentación de librerías, ejemplos de partidas en formato PGN, herramientas de control de versiones (Git/GitHub). * Todos estos materiales son accesibles y no requieren inversión significativa.  1. Factores externos que facilitan el desarrollo:  * Acceso a bibliotecas y motores open-source (React, Stockfish, Firebase). * Entorno de aprendizaje ágil que permite asesorías y feedback continuo del docente y compañeros. * Disponibilidad de documentación y tutoriales en línea que agilizan la integración de tecnologías. * Factores externos que podrían dificultar el desarrollo y soluciones:  1. Latencia o limitaciones al ejecutar Stockfish en cliente: se puede optar por un servicio backend para análisis más pesado o ajustar la profundidad de cálculo.  * Falta de experiencia en integración de motores de análisis: mitigable mediante el estudio de ejemplos, documentación oficial y prototipado gradual. * Tiempo limitado por otras asignaturas: se organiza un cronograma de trabajo y sprints para priorizar las funcionalidades críticas del MVP. * Problemas de conexión a internet o despliegue: se planifica un entorno local de pruebas y respaldos, asegurando que el desarrollo no dependa completamente del hosting en línea. |

1. **PARTE II**

|  |
| --- |
| **4. Objetivos** |
| En este apartado debes definir objetivos generales y específicos del Proyecto APT. Es importante aclarar que los objetivos se deben plantear en forma clara, concisa y sin dar mayores explicaciones, es decir, deben entenderse por sí solos. Se sugiere redactarlos utilizando un verbo en infinitivo, pues ello obliga a precisar acciones concretas. |

|  |  |
| --- | --- |
| Objetivo general | El objetivo general del Proyecto APT *Study Chess* es desarrollar un prototipo funcional de plataforma web que permita a los usuarios estudiar y mejorar en ajedrez mediante análisis automatizado de partidas y retroalimentación pedagógica. La plataforma integrará un tablero interactivo, un motor de análisis (Stockfish) y una capa de interpretación que traduzca las evaluaciones técnicas en explicaciones comprensibles y accionables.  Este objetivo busca proporcionar una solución práctica a la problemática identificada: la dificultad de los jugadores para interpretar de manera útil las evaluaciones de motores de ajedrez y mejorar su aprendizaje autónomo. Al finalizar el proyecto, se evaluará en qué medida el prototipo cumple con la funcionalidad esperada, la calidad de las explicaciones pedagógicas y la usabilidad para los usuarios. |
| Objetivos específicos | 1. Implementar un tablero interactivo de ajedrez  * Diseñar e integrar un tablero web que permita arrastrar y soltar piezas, visualizar movimientos legales y reproducir partidas con control de velocidad.  1. Integrar un motor de análisis de partidas  * Conectar Stockfish para obtener evaluaciones numéricas y variantes alternativas por jugada, asegurando precisión y consistencia en los resultados.  1. Desarrollar una capa pedagógica de retroalimentación  * Traducir los cambios de evaluación del motor en explicaciones comprensibles y accionables, indicando errores tácticos, oportunidades estratégicas o recomendaciones de estudio.  1. Crear un sistema de almacenamiento y gestión de estudios  * Implementar una biblioteca donde los usuarios puedan guardar partidas, análisis y sesiones de estudio, con autenticación y recuperación de datos.  1. Validar la usabilidad y efectividad pedagógica del prototipo  * Realizar pruebas funcionales y un piloto con usuarios de nivel principiante a intermedio para evaluar claridad, utilidad y facilidad de uso de la plataforma.  1. Documentar el proyecto y preparar entregables  * Generar documentación técnica, manual de usuario y reporte de resultados del piloto que respalde el cumplimiento de los objetivos planteados. |

|  |
| --- |
| **5. Metodología** |
| En el siguiente apartado deberás describir la metodología, propia de tu disciplina, que utilizarás para resolver el proyecto APT antes descrito, incluyendo las etapas y métodos de trabajo. |

|  |
| --- |
| Descripción de la Metodología |
| Para abordar la problemática de transformar análisis técnico de motores de ajedrez en retroalimentación pedagógica comprensible, se seguirá una metodología ágil basada en Scrum, organizada en sprints semanales y entregas incrementales. Este enfoque permitirá priorizar funcionalidades críticas, ajustar el desarrollo según retroalimentación y asegurar que se cumpla el objetivo general del proyecto dentro del semestre.  Enfoque metodológico   1. Planificación y definición de alcance (Sprint 0)   Identificación de funcionalidades esenciales para el MVP: tablero interactivo, integración de Stockfish, retroalimentación pedagógica y biblioteca de estudios.  Asignación de tareas, estimación de tiempos y definición de criterios de aceptación.   1. Desarrollo iterativo (Sprints 1–N)   Implementación progresiva de módulos: frontend (tablero, historial de jugadas, panel de análisis), integración del motor Stockfish, generación de explicaciones automáticas y persistencia de datos.  Revisión y pruebas parciales al final de cada sprint para asegurar cumplimiento de criterios y ajustar prioridades.   1. Pruebas y validación   Pruebas funcionales unitarias y de integración para cada módulo.  Piloto con usuarios (niveles principiante e intermedio) para evaluar usabilidad y efectividad pedagógica de la retroalimentación.  Registro de observaciones, métricas de tiempo de uso y claridad de explicaciones para ajustar heurísticas y la interfaz.   1. Documentación y entrega   Elaboración de documentación técnica (arquitectura, instrucciones de despliegue, decisiones de diseño).  Generación de manual de usuario y reporte de resultados del piloto.  Preparación de la presentación final del prototipo.  Funciones, tareas y responsabilidades del equipo (si aplica)   * Integrante 1 – Desarrollo Frontend: responsable del tablero interactivo, paneles de análisis y UI/UX. * Integrante 2 – Integración y Lógica de Análisis: encargado de la integración con Stockfish, desarrollo de la capa pedagógica y generación de explicaciones. * Integrante 3 – Backend y Persistencia (opcional): gestión de la base de datos de usuarios, almacenamiento de estudios y autenticación. * Todos los integrantes: planificación de sprints, documentación técnica, pruebas funcionales, recopilación de feedback del piloto y preparación de la entrega final. |

|  |
| --- |
| **6. Evidencias** |
| A continuación, describe qué evidencias serán evaluadas en el informe de avance y en el informe final de tu proyecto APT. Estas evidencias deben ser acordadas con tu docente. Se entenderá por evidencia los productos que se desarrollen durante el proyecto y cuyo propósito sea visibilizar o documentar cómo se ha implementado el trabajo. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Tipo de evidencia**  **(avance o final)** | **Nombre de la evidencia** | **Descripción** | **Justificación** |
| Avance | Capturas del tablero interactivo | Fotografías del tablero en funcionamiento mostrando movimientos legales, historial de jugadas y controles de reproducción. | Permite evidenciar que la interfaz y la interacción básica con el tablero están implementadas correctamente. |
| Avance | Historial de jugadas y panel de análisis | Capturas mostrando la evaluación de jugadas y variantes generadas por el motor. | Demuestra que el motor de análisis está integrado y funcionando, paso clave para la retroalimentación pedagógica. |
| Avance | Capa pedagógica / Explicaciones automáticas | Capturas de Firestore mostrando registros de usuarios y partidas guardadas. | Evidencia que la persistencia de información y la gestión de usuarios están implementadas. |
| Avance | Repositorio GitHub (MVP inicial) | Link al repositorio con commits iniciales, estructura de carpetas y código comentado. | Permite evidenciar el progreso en el desarrollo, buenas prácticas de control de versiones y documentación de los primeros módulos. |
| Final | Plataforma web funcional (MVP completo) | Demo desplegada con tablero, análisis por jugada, retroalimentación pedagógica y biblioteca de estudios. | Muestra que el proyecto cumple con los objetivos generales y específicos planteados. |
| Final | |  | | --- | |  |  |  | | --- | | Capturas del prototipo final | | Fotografías de todas las funcionalidades integradas: tablero, análisis, biblioteca y perfil de usuario. | Evidencia completa de la implementación final y del cumplimiento de requisitos funcionales. |
| Final | Informe técnico y documentación | Memoria de tesis, manual de usuario, diagrama de arquitectura, explicación de decisiones de diseño y guía de despliegue. | Permite evaluar la correcta documentación del proyecto y la capacidad de comunicar el desarrollo técnico y pedagógico. |
| Final | Resultados del piloto | Registro de sesiones de prueba con usuarios, cuestionarios completados y métricas de uso. | Permite validar la usabilidad, la efectividad pedagógica y la pertinencia del proyecto frente a la problemática inicial. |
| Final | Repositorio GitHub actualizado | Repositorio con todos los commits finales, código completo y documentación. | Evidencia que todo el desarrollo está versionado, organizado y listo para revisión académica o futura continuidad del proyecto. |

|  |
| --- |
| **7. Plan de Trabajo** |
| En la siguiente tabla define la planificación de tu Proyecto APT de acuerdo a lo requerido. |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Plan de Trabajo Proyecto APT** | | | | | | |
| Competencia o unidades de competencias | Nombre de Actividades/Tareas | Descripción Actividades/Tareas | Recursos | Duración de la actividad | Responsable[[1]](#footnote-1) | Observaciones |
| Desarrollo y diseño de software | Implementación del tablero interactivo | Crear tablero web interactivo con arrastrar/soltar, movimientos legales y control de reproducción de partidas | React, librería de tablero, computadora personal | 2 semanas | Rafael Ancalipe | Posible ajuste de UI según pruebas de usabilidad; facilita aprendizaje práctico de React |
| Aplicación de algoritmos / Integración de motor | Integración de Stockfish y análisis por jugada | Conectar Stockfish (WASM o backend) y generar evaluaciones y variantes para cada jugada | Stockfish, Node.js, computadora personal | 2 semanas | Rafael Ancalipe y bruno rojo | Latencia de motor mitigable ajustando profundidad de análisis |
| Capa pedagógica y heurística | Desarrollo de retroalimentación pedagógica | Traducir evaluaciones numéricas en explicaciones comprensibles para el usuario | JavaScript, reglas heurísticas, ejemplos de partidas | 2 semanas | Rafael Ancalipe y bruno rojo | Ajustar explicaciones según feedback de piloto; facilita aprendizaje autónomo |
| Bases de datos y persistencia | Implementación de biblioteca de estudios y autenticación | Guardar partidas, análisis y perfiles de usuario en Firestore con autenticación segura | Firebase, Firestore, GitHub | 1 semana | Rafael Ancalipe y bruno rojo | Posibles problemas de permisos mitigables con buenas prácticas de Firebase |
| UI/UX y pruebas | Ajuste de interfaz y pruebas de usabilidad | Mejorar la usabilidad del tablero y paneles de análisis; realizar pruebas con usuarios | Prototipos, cuestionarios, testeo piloto | 1 semana | Rafael Ancalipe y bruno rojo | Facilita validación del MVP y retroalimentación de usuarios reales |
| Gestión de proyecto | Planificación por sprints y seguimiento | Organizar backlog, definir tareas por sprint, seguimiento de progreso y control de versiones | Trello/GitHub Issues, GitHub, reuniones de equipo | Durante todo el proyecto | Rafael Ancalipe y bruno rojo | Uso de Scrum facilita priorización y entrega incremental |
| Documentación y reporte | Elaboración de documentación técnica y manual de usuario | Crear memoria de proyecto, guía de despliegue, manual de usuario y reportes de piloto | Word/Google Docs, capturas, diagramas, GitHub | 1 semana | Rafael Ancalipe y bruno rojo | Facilita evaluación académica y futura continuidad del proyecto |
| |  | | --- | | Validación pedagógica |  |  | | --- | |  | | Piloto con usuarios y ajuste de heurísticas | Realizar prueba con jugadores principiantes/intermedios, recoger feedback y ajustar explicaciones automáticas | Computadora, cuestionarios, partidas PGN | 1 semana | Rafael Ancalipe y bruno rojo | Feedback crítico para garantizar efectividad pedagógica y cumplir objetivos |

|  |
| --- |
| **8. Carta Gantt** |
| Busca un formato de Carta Gantt que te acomode y organiza en este las actividades planificadas en el punto anterior considerando el periodo asignado para el desarrollo de tu Proyecto APT. Debes mantener la temporalidad del periodo académico en el desarrollo de las tres fases que contempla la Asignatura de Portafolio de Título. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Actividad** | **Fase 1** | | | | **Fase 2** | | | | | | | | | | | | **Fase 3** | | | |
| **S 1** | **S 2** | **S 3** | **S 4** | **S 5** | **S 6** | **S 7** | **S 8** | **S 9** | **S 10** | **S 11** | **S 12** | **S 13** | **S 14** | **S 15** | **S 16** | | **S 17** | **S 18** |
| Implementación tablero interactivo |  | **XX** | **XX** | **XX** | **XX** |  |  |  |  |  |  | **XX** |  |  |  |  | |  |  |
| **Integración de Stockfish** |  |  |  |  | **XX** | **XX** | **XX** | **XX** | **XX** |  |  |  |  |  |  |  | |  |  |
| **Desarrollo retroalimentación pedagógica** |  |  |  |  |  | **XX** | **XX** | **XX** | **XX** |  |  |  |  |  |  |  | |  |  |
| **Biblioteca de estudios y autenticación** |  |  |  |  |  |  | **XX** | **XX** | **XX** |  |  |  |  |  |  |  | |  |  |
| **Ajuste de interfaz y pruebas de usabilidad** |  |  |  |  |  |  |  |  | **XX** | **XX** | **XX** |  |  | **XX** |  |  | |  |  |
| **Gestión de proyecto (Scrum, seguimiento)** | **XX** | **XX** | **XX** | **XX** | **XX** | **XX** | **XX** | **XX** | **XX** | **XX** | **XX** | **XX** | **XX** | **XX** | **XX** | **XX** | | **XX** | **XX** |
| **Documentación técnica y reporte** |  |  |  |  |  |  |  |  |  |  |  |  |  | **XX** | **XX** | **XX** | | **XX** |  |
| **Piloto con usuarios y validación pedagógica** |  |  |  |  |  |  |  |  |  |  |  |  | **XX** | **XX** | **XX** | **XX** | |  |  |

1. En caso de que el Proyecto APT sea grupal, en esta columna deben indicar el nombre de los responsables de cada tarea o actividad. Esto posteriormente permitirá diferenciar la evaluación por cada integrante. [↑](#footnote-ref-1)