}

**Resumen**

El proyecto ECOS es un marketplace desarrollado como respuesta al problema de la gestión informal de residuos y los vertederos ilegales en Puente Alto. Su objetivo principal es entregar a los ciudadanos una plataforma segura y confiable donde puedan publicar, donar o vender objetos en desuso, fomentando la reutilización y la economía circular. El proyecto aplica una metodología ágil híbrida que combina Scrum y Kanban, estructurada en cuatro sprints con entregables definidos. Los resultados esperados incluyen una solución tecnológica funcional que fortalezca la gestión comunitaria de residuos, valide la aplicación de las competencias del programa de Ingeniería en Informática y demuestre la factibilidad de abordar problemáticas sociales y ambientales mediante el desarrollo de software.

1. **PARTE I**

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

| Nombre estudiante | **Camila Hormazabal Quintulen/ Cristóbal Marín Palacios /Rodrigo Montalván Rivera /Annais Romero Díaz.** |
| --- | --- |
| Rut | **21.195.029-3/ 19.920.373-8/20.226.543-K/ 20.538.007-8** |
| Carrera | **Ingeniería en informática** |
| Sede | **Puente Alto** |

| **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 | ECOS |
| --- | --- |
| Área (s) de desempeño(s) | Desarrollo de software y aplicaciones web.  Gestión de proyectos informáticos.  Calidad de software y aseguramiento de la usabilidad. |
| Competencias | * Diseñar y generar soluciones de software innovadoras y de calidad, aplicando el ciclo de vida completo y siguiendo estándares y buenas prácticas de la industria. * Diseñar y adaptar procesos de ingeniería de requisitos, a través del uso de metodologías de vanguardia y estándares de la industria. * Evaluar y gestionar proyectos informáticos, aplicando metodologías ágiles e híbridas para cumplir con los requerimientos organizacionales. * Desarrollar proyectos de software innovadores para plataformas web, utilizando marcos de trabajo, herramientas de desarrollo y buenas prácticas de codificación. * Diseñar soluciones de software seguras y sostenibles, considerando aspectos de calidad, seguridad y escalabilidad. |

| **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 ECOS surge como respuesta a una problemática social y ambiental presente en la comuna de Puente Alto, debido a la acumulación de objetos en desuso que terminan en microbasurales o son gestionados de forma informal en redes sociales. Esta situación genera impactos negativos en el entorno urbano, riesgos sanitarios y desaprovechamiento de bienes que podrían ser reutilizados.  El tema lo escogimos porque vincula directamente la tecnología con un problema real de la comunidad, permitiendo aplicar la informática como herramienta para esta transformación social. Para nuestra carrera, esta problemática es relevante, ya que implica diseñar e implementar una solución tecnológica que incorpore ingeniería de requisitos, desarrollo de software, gestión de proyectos y control de calidad, todas competencias clave en el perfil de egreso.  La situación se ubica en Puente Alto, una de las comunas más pobladas del país, caracterizada por una alta generación de residuos y limitada capacidad de gestión local en este ámbito. Afecta directamente a los habitantes de la comuna, especialmente a familias que no cuentan con recursos para costear servicios privados de retiro, los cuales suelen tener convenios con grandes organizaciones y altos costos, inaccesibles para la mayoría de los hogares.  El aporte de valor de este proyecto es netamente social, ya que permite entregarle a la comunidad una plataforma formal que facilita la reutilización de objetos, reduciendo microbasurales. |
| --- | --- |
| Descripción del Proyecto APT | El proyecto consiste en el desarrollo de una aplicación web tipo marketplace que permita publicar, buscar, vender o donar objetos en desuso y que se le pueda dar una segunda vida. El sistema contará con registro de usuarios, roles diferenciados (administrador/usuario final), catálogo con filtros dinámicos, transacciones en entorno seguro (sandbox), notificaciones automáticas y generación de reportes.  Se espera lograr una solución tecnológica funcional que atienda la necesidad detectada en Puente Alto, impulsando la reutilización de bienes y mejorando la gestión comunitaria de residuos. El abordaje será mediante una metodología ágil híbrida (Scrum y Kanban), asegurando entregas iterativas, pruebas continuas y trazabilidad de requisitos. |
| Pertinencia del proyecto con el perfil de egreso | Nuestro proyecto se vincula directamente con las competencias del perfil de egreso ya que requiere diseñar y generar soluciones de software innovadoras y de calidad, aplicando el ciclo de vida completo.  Exige gestionar un proyecto informático con plazos y entregables definidos, aplicando buenas prácticas ágiles.  Implica adaptar procesos de ingeniería de requisitos y traducirlos en funcionalidades verificables.  Demanda la aplicación de pruebas de calidad y seguridad, garantizando usabilidad y confiabilidad en el software desarrollado. |
| Relación con los intereses profesionales | El proyecto ECOS se relaciona con nuestros intereses profesionales de tres maneras diferentes, una es el desarrollo de software ,ya que nos motiva crear soluciones tecnológicas desde cero y estructurarlas con buenas prácticas; También está la gestión de proyectos, donde en verdad disfrutamos planificar y organizar tareas, distribuyendo recursos y coordinando al equipo y por último está en resguardar la calidad del software, debido a que nos interesa asegurar que las soluciones entregadas sean seguras, confiables y usables para el usuario.  Al realizar este proyecto, estamos seguros de que como equipo fortaleceremos tanto nuestras competencias individuales como nuestra capacidad de trabajo colaborativo, lo que aportará a nuestro desarrollo profesional. |
| Factibilidad de desarrollo del Proyecto APT | El desarrollo del proyecto es factible dentro del semestre porque la duración es de aproximadamente 17 semanas, la cual permite organizar las fases en 4 sprints con entregables claros para las fechas solicitadas como entrega.  Se cuenta con horas asignadas y trabajo grupal, lo que facilita distribuir responsabilidades.  Se utilizarán tecnologías open source (Python/Django, MySQL, Bootstrap, GitHub), lo que elimina costos de licenciamiento.  Los factores externos que facilitan el desarrollo son el acceso a entornos de desarrollo gratuitos, disponibilidad de bibliografía, apoyo de la docente y metodologías ágiles previamente aplicadas en semestres anteriores.  Los factores que podrían dificultar son la curva de aprendizaje en Django/MySQL y la dependencia de servicios externos como SMTP y sandbox de pagos. Estos se abordarán mediante capacitación autónoma, pruebas previas de integración y diseño modular que permita mitigar errores sin afectar a todo el sistema. |

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 es desarrollar una aplicación web tipo marketplace, llamada ECOS, que permita a los ciudadanos publicar, buscar, donar y vender objetos en desuso, fomentando la reutilización y la economía circular, mediante un sistema seguro, accesible y confiable que cumpla con los requerimientos funcionales y no funcionales definidos. |
| --- | --- |
| Objetivos específicos | Levantar y documentar los requerimientos funcionales y no funcionales del sistema, asegurando trazabilidad y alineación con las necesidades del usuario.  Diseñar la arquitectura del sistema bajo un enfoque en capas (Modelo-Vista-Controlador), considerando estándares de calidad, seguridad y sostenibilidad.  Implementar los módulos principales de la aplicación web, incluyendo autenticación de usuarios, publicación de productos, catálogo con filtros, transacciones en entorno sandbox y generación de reportes exportables (PDF/XLS).  Desarrollar el módulo de administración para la gestión de usuarios, roles, categorías y políticas del sistema, garantizando un control en la plataforma.  Aplicar pruebas unitarias, de integración y de usabilidad para validar el correcto funcionamiento del sistema y asegurar un mínimo de 90% de casos exitosos.  Generar la documentación técnica y de usuario final, que incluya manuales, plan de capacitación y registros de defectos.  Cumplir con los hitos y plazos establecidos. |

| **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 el desarrollo de este proyecto APT ECOS, se implementará una metodología ágil híbrida, basada en los principios de SCRUM y complementados con herramientas de Kanban para la gestión visual de tareas. Este enfoque nos permitirá organizar el trabajo en iteraciones cortas (sprints), asegurar la retroalimentación continua y mantener la trazabilidad de cada avance.  El proyecto se dividirá en 4 sprints principales, con los entregables definidos para cada fase del proyecto:   * **Sprint 1:** Documentación inicial, levantamiento de requisitos, casos de uso, mockups y EDT. * **Sprint 2:** Documento de arquitectura, diccionario de datos, matrices de riesgos y RACI, y desarrollo de módulos base (login, registro y mantenedores iniciales). * **Sprint 3:** Implementación de funcionalidades principales (publicaciones, catálogo, transacciones sandbox, notificaciones y reportes). * **Sprint 4:** Ejecución de pruebas funcionales e integración, manual de usuario, plan de capacitación y cierre del proyecto.   **Métodos de trabajo**   * Realizar reuniones de planificación al inicio de cada sprint para definir las tareas, estimar el esfuerzo y priorizar entregables. * Daily meetings breves de 15 minutos para revisar avances, bloqueos y próximos pasos. * Revisiones de sprint para laexposición de avances y validación de los entregables parciales. * Retrospectiva del proceso interno de las asignaciones para identificar mejoras en la colaboración y la gestión del proyecto. * Para la gestión de tareasse usará un tablero Kanban digital en la plataforma Notion para visualizar el progreso de las actividades asignadas.   **Roles y responsabilidades**   * **Jefe de Proyecto (Annais Romero):** encargada de coordinar el cumplimiento de plazos, alcance y costos, también debe gestionar la comunicación con los interesados y supervisar la documentación. * **Rodrigo Montalván (PMO y QA/Tester):** encargado de asegurar el cumplimiento metodológico y trazabilidad del proyecto, además de diseñar y ejecutar pruebas de calidad (manuales y automatizadas). * **Analista programador (Cristóbal Marín):** encargado de implementar los módulos funcionales, gestionar la base de datos y desarrollar las pruebas unitarias. * **Diseñadora (Camila Hormazabal):** debe elaborar mockups, prototipos y velar por la experiencia del usuario y la usabilidad de la plataforma. |

| **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** | **Documento ERS** (Especificación de Requerimientos de Software) | Documento que detalla los requisitos funcionales y no funcionales del sistema ECOS, incluyendo casos de uso y diagramas. | Permite validar que el sistema responde a necesidades claras y verificables desde el inicio. |
| **Avance** | **Acta de constitución del proyecto** | Documento formal que establece los objetivos, alcance, roles y restricciones iniciales del proyecto. | Garantiza un marco oficial de inicio y alinea al equipo con los lineamientos del proyecto. |
| **Avance** | **Planilla de Requerimientos** | Registro detallado de requisitos funcionales y no funcionales en formato estructurado. | Facilita la trazabilidad de los requisitos y su control durante el desarrollo. |
| **Avance** | **Documento de Caso de Uso extendido** | Describe en detalle los flujos de interacción entre usuarios y sistema para cada funcionalidad clave. | Asegura claridad en el comportamiento esperado del sistema y reduce ambigüedades en desarrollo. |
| **Avance** | **Mockups del sistema completo** | Representaciones visuales de las pantallas principales y flujo de navegación del sistema. | Permiten validar usabilidad y experiencia de usuario antes de la implementación técnica. |
| **Avance** | **EDT-Costos** | Estructura de desglose del trabajo (EDT) junto con la estimación de costos asociados. | Ayuda a organizar el esfuerzo del proyecto y calcular la inversión necesaria. |
| **Avance** | **Roadmap** | Plan visual de hitos y entregables distribuidos en el tiempo. | Sirve como guía de planificación y control del progreso del proyecto. |
| **Avance** | **Documento proceso de negocio (TO-BE)** | Definición de los procesos mejorados que soportará la solución tecnológica. | Permite alinear el sistema con los procesos de negocio proyectados. |
| **Avance** | **Matriz RACI** | Matriz que define responsabilidades (Responsable, Aprobador, Consultado, Informado) por tarea. | Evita ambigüedades en los roles del equipo y asegura claridad en la gestión. |
| **Avance** | **Matriz de Riesgo** | Identificación y análisis inicial de riesgos asociados al proyecto. | Facilita la anticipación de problemas y la definición de estrategias de mitigación. |
| **Avance** | **Diccionario de Datos** | Documento que describe atributos, tipos de datos y relaciones principales de la base de datos. | Asegura consistencia en el modelado de información y facilita la implementación técnica. |
| **Avance** | **Minutas de reuniones** | Registro formal de acuerdos, compromisos y temas tratados en cada reunión del proyecto. | Mantiene trazabilidad de decisiones y facilita la comunicación dentro del equipo. |
| **Avance** | **Documento de avance de sprint** | Informe de actividades y resultados obtenidos en cada sprint (si se usa metodología ágil). | Permite evaluar el cumplimiento de objetivos parciales y ajustar la planificación. |
| **Final** | **Plan de pruebas** | Documento que define casos, escenarios y criterios de prueba del sistema. | Garantiza la calidad del software mediante validación planificada y sistemática. |
| **Final** | **Matriz de pruebas de base de datos** | Documento que evalúa la consistencia, integridad y rendimiento de la base de datos. | Asegura que la base de datos soporte correctamente las operaciones del sistema. |
| **Final** | **Planilla de registro de defectos** | Registro de errores encontrados durante las pruebas y su estado de resolución. | Permite gestionar y dar seguimiento al proceso de corrección de fallas. |
| **Final** | **Verificación de alcances** | Documento que compara entregables finales contra lo planificado en los requerimientos iniciales. | Asegura que el producto cumpla con lo comprometido en alcance y calidad. |
| **Final** | **Plan de capacitación** | Material y planificación para capacitar a los usuarios en el uso de la aplicación. | Facilita la adopción del sistema y asegura un uso adecuado por parte de los usuarios |
| **Final** | **Manual de usuario** | Documento práctico que guía a los usuarios en el uso de las funcionalidades del sistema. | Asegura autonomía de los usuarios finales y reduce la necesidad de soporte constante. |
| **Final** | **Matriz de control de cambios** | Registro formal de cambios en requisitos, alcance o cronograma. | Permite gestionar modificaciones y mantener la trazabilidad de decisiones. |
| **Final** | **Informe cierre del proyecto** | Documento que recopila resultados, aprendizajes y evaluación del proyecto. | Sirve como evidencia de cumplimiento y reflexión final del trabajo. |
| **Final** | **Informe Final Proyecto APT** | Documento académico integral en español e inglés que resume el proceso y resultados obtenidos. | Corresponde a la entrega formal que será evaluada por la institución. |
| **Final** | **PPT presentación de avance y presentación final** | Presentaciones en PowerPoint para comunicar los avances y resultados finales. | Facilitan la exposición ante la docente, el comité evaluador y otros interesados. |

| **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-0)** | **Observaciones** |
| Gestionar proyectos informáticos | **Acta de constitución** | Documento que formaliza el inicio del proyecto, define marco, roles y objetivos. | Word.  reuniones de equipo. | Sprint 1. | Annais Romero  Rodrigo Montalván | Documento base, dependiente de los acuerdos iniciales del equipo y contexto. |
| Diseñar y adaptar procesos de requisitos | **Requerimientos (RF/RNF)** | Levantamiento y documentación de requisitos funcionales y no funcionales. | Planilla de requisitos.  Excel. | Sprint 1. | Annais Romero  Rodrigo Montalván.  Cristóbal Marín. | Puede requerir ajustes tras validación con docente. |
| Diseñar soluciones de software | **ERS inicial** | Documento con especificación de requerimientos de software, casos de uso y RNF. | Word.  Guía ERS.  Excel. | Sprint 1. | Annais Romero  Cristóbal Marín | Documento muy Iterativo, se complementa en fases siguientes. |
| Diseñar soluciones de software | **Casos de uso extendidos** | Modelado detallado de interacciones entre actores y sistema. | Word. | Sprint 1. | Rodrigo Montalván | Se debe validar en conjunto con los requerimientos funcionales. |
| Diseñar soluciones de software  Desarrollar documentación técnica | **Mockups del sistema** | Prototipos visuales de pantallas y flujos principales. | Balsamiq | Sprint 1. | Cristóbal Marín  Camila Hormazabal | Facilita la validación temprana de la usabilidad del software propuesto. |
| Gestionar proyectos informáticos | **EDT - Costos** | Estructura de desglose de trabajo con estimación de costos. | Excel | Sprint 1. | Annais Romero | Base para la planificación financiera. |
| Diseñar soluciones de software innovadoras y de calidad | **Documento de arquitectura (DAS - 1° parte)** | Definición de la arquitectura técnica del sistema (capas, BD, integraciones). | Word.  UML,.  Lucidchart/Draw.io. | Sprint 2. | Cristóbal Marín  Annais Romero | Debe aprobarse antes del desarrollo. |
| Diseñar y adaptar procesos de ingeniería de requisitos | **Procesos TO-BE** | Modelado de procesos futuros que describen la operación ideal del sistema. | Bizagi | Sprint 2. | Rodrigo Montalván  Annais Romero  Cristóbal Marín  Camila Hormazabal | Se debe validar con la docente y retroalimentación del equipo. |
| Diseñar y generar soluciones que permitan resolver requerimientos de información | **Diccionario de datos** | Definición de entidades, atributos y reglas de la base de datos. | MySQL.  Excel. | Sprint 2. | Cristóbal Marín | Fundamental para la consistencia en el desarrollo posterior del software. |
| Evaluar y gestionar proyectos informáticos | **Matriz RACI** | Asignación de responsabilidades (Responsible, Accountable, Consulted, Informed). | Excel. | Sprint 2. | Annais Romero | Requiere consenso del equipo. |
| Evaluar y gestionar proyectos informáticos | **Matriz de riesgos** | Identificación, análisis y categorización de riesgos iniciales. | Excel. | Sprint 2. | Rodrigo Montalván | Documento activo, se actualizará durante el proyecto. |
| Desarrollar soluciones de software utilizando buenas prácticas de codificación | **Login/Registro + mantenedores iniciales (usuarios, roles, categorías)** | Desarrollo de los módulos básicos de autenticación y gestión inicial. | Django.  MySQL.  Bootstrap.  GitHub. | Sprint 2. | Cristóbal Marín | Base crítica para el resto de las funcionalidades. |
| Evaluar y gestionar proyectos informáticos | **Matriz de control de cambios** | Registro formal de cambios en requisitos, alcance y diseño. | Excel. | Sprint 2. | Cristóbal Marín  Camila Hormazabal | Controla la trazabilidad del proyecto. |
| Desarrollar soluciones de software innovadoras para plataformas web. | **Publicación y gestión de productos** | Implementación del módulo para crear, editar, pausar y reactivar publicaciones con fotos y atributos. | Django.  MySQL.  Bootstrap. | Sprint 3. | Cristóbal Marín  Camila Hormazabal | Validar reglas de negocio. |
| Diseñar y generar soluciones de software de calidad. | **Catálogo con filtros y búsqueda** | Desarrollo del catálogo con filtros dinámicos (categoría, comuna, precio, estado). | Django.  MySQL.  Bootstrap. | Sprint 3. | Cristóbal Marín | Requerimiento no funcional relacionado con el tiempo de respuesta que debe ser menor a 3 segundos. |
| Construir programas y rutinas de variada complejidad. | **Transacciones (ventas/donaciones con sandbox)** | Implementación del flujo de compra/donación con integración sandbox de pagos. | Django.  API Sandbox MercadoPago. | Sprint 3. | Cristóbal Marín | Riesgo de dependencia de servicio externo. |
| Diseñar soluciones de software seguras y confiables. | **Notificaciones por correo** | Configuración del envío automático de correos en eventos clave (registro, transacciones, cambios de estado). | Django.  SMTP.  servidor de correo. | Sprint 3. | Cristóbal Marín | Validar envío solo a cuentas registradas. |
| Diseñar y generar soluciones innovadoras de calidad. | **Mantenedores secundarios (estados, métodos de pago, políticas)** | CRUD de métodos de pago, estados de producto y políticas. | Django.  MySQL. | Sprint 3. | Cristóbal Marín  Camila Hormazabal | Mantener consistencia con BD y RF previos. |
| Diseñar soluciones de software apoyadas en datos. | **Reportes (productos vendidos, transacciones por período)** | Generación de reportes exportables (PDF/XLS) de publicaciones y transacciones. | Django.  Librerías reportlab/xlsxwriter. | Sprint 3. | Cristóbal Marín  Rodrigo Montalván | Validar los filtros y consistencia de datos. |
| Diseñar y generar soluciones que permitan resolver requerimientos de información. | **Script de base de datos** | Creación del script SQL para poblar datos iniciales y mantener integridad de la base. | MySQL. | Sprint 3. | Cristóbal Marín | Requiere de pruebas de integridad y relaciones. |
| Evaluar y gestionar proyectos informáticos. | **Verificación de alcances** | Validación de que los módulos implementados coincidan con el roadmap y requisitos definidos. | Checklist.  Minutas de reuniones. | Sprint 3. | Annais Romero  Rodrigo Montalván | Condición previa a QA en el Sprint 4 |
| Realizar pruebas de certificación de software | **Pruebas funcionales e integración** | Validación del funcionamiento global del sistema con distintos escenarios. | Selenium.  Unittest.  checklist. | Sprint 4. | Rodrigo Montalván. | Se espera que los casos de prueba sean mayor o igual al 90% exitosos. |
| Construir programas y rutinas de variada complejidad  Validar soluciones de datos | **Matriz de pruebas de BD.** | Documento que registra las pruebas realizadas sobre consultas y rutinas de la base de datos. | Excel.  MySQL. | Sprint 4. | Cristóbal Marín.  Rodrigo Montalván. | Asegura consistencia y rendimiento en BD. |
| Realizar pruebas de certificación de software | **Registro de defectos** | Documento de control donde se registran los errores encontrados durante QA. | Excel.  Planilla de defectos. | Sprint 4. | Rodrigo Montalvan.  Cristóbal Marín.  Camila Hormazabal. | Permite la trazabilidad y control de calidad. |
| Desarrollar documentación técnica y de usuario | **Manual de usuario más plan de capacitación** | Documentación final que orienta a los usuarios en el uso del sistema y plan de formación. | Word.  PowerPoint. | Sprint 4. | Annais Romero.  Camila Hormazabal. | Asegurar la adaptación del usuario al sistema. |
| Evaluar y gestionar proyectos informáticos | **Informe final APT 2.0 (Español/Inglés)** | Documento formal que consolida resultados, metodología, evidencias y aprendizajes. | Word. | Sprint 4. | Annais Romero.  Rodrigo Montalván. | Requiere revisión en equipo antes de la entrega. |
| Evaluar y gestionar proyectos informáticos  Comunicación efectiva | **PPT presentación final** | Presentación ejecutiva del proyecto para defensa ante comisión. | PowerPoint.  Canva. | Sprint 4. | Annais Romero.  Cristóbal Marín.  Rodrigo Montalván.  Camila Hormazabal. | Requiere ensayos previos y coordinación grupal. |

| **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** | **Sprint 1** | | | | **Sprint 2** | | | | **Sprint 3** | | | | | | | **Sprint 4** | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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** |
| Acta de constitución | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Requerimientos (RF/RNF) |  | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| ERS inicial |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Casos de uso extendidos |  |  | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Mockups del sistema |  |  | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |
| EDT - Costos |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Documento de arquitectura (DAS - 1° parte) |  |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |  |
| Procesos TO-BE |  |  |  |  | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  |
| Diccionario de datos |  |  |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |
| Matriz RACI |  |  |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |
| Matriz de riesgos |  |  |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |
| Login/Registro + mantenedores iniciales (usuarios, roles, categorías) |  |  |  |  |  |  | **X** | **X** |  |  |  |  |  |  |  |  |  |
| Matriz de control de cambios |  |  |  |  |  |  |  | **X** |  |  |  |  |  |  |  |  |  |
| Publicación y gestión de productos |  |  |  |  |  |  |  |  | **X** |  |  |  |  |  |  |  |  |
| Catálogo con filtros y búsqueda |  |  |  |  |  |  |  |  |  | **X** |  |  |  |  |  |  |  |
| Transacciones (ventas/donaciones con sandbox) |  |  |  |  |  |  |  |  |  |  | **X** | **X** |  |  |  |  |  |
| Notificaciones por correo |  |  |  |  |  |  |  |  |  |  |  | **X** |  |  |  |  |  |
| Mantenedores secundarios (estados, métodos de pago, políticas) |  |  |  |  |  |  |  |  |  |  |  | **X** | **X** |  |  |  |  |
| Reportes (productos vendidos, transacciones por período) |  |  |  |  |  |  |  |  |  |  |  |  | **X** | **X** |  |  |  |
| Script de base de datos |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |  |  |
| Verificación de alcances |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |  |
| Pruebas funcionales e integración |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |
| Matriz de pruebas de BD. |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |
| Registro de defectos |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |
| Manual de usuario más plan de capacitación |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** | **X** |
| Informe final APT 2.0 (Español/Inglés) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |
| PPT presentación final |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |

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-0)