}

**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 | * **Daniel Gálvez** * **Fabian Bravo** |
| Rut | **20.908.333-7**  **21.492.646-6** |
| 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 | **CanchaYa** |
| Área (s) de desempeño(s) | **El Proyecto APT CanchaYa aborda principalmente el área de desarrollo e integración de sistemas informáticos, incorporando también la gestión de proyectos tecnológicos y el aseguramiento de la calidad del software. Estas áreas forman parte del campo laboral del Ingeniero en Informática, ya que demandan la capacidad de diseñar, implementar y mantener soluciones tecnológicas seguras y escalables que optimicen procesos organizacionales.** |
| Competencias | **El proyecto permite aplicar diversas competencias del perfil de egreso, entre las que destacan:**   * **Levantamiento y análisis de requerimientos para ofrecer propuestas de solución informática ajustadas a las necesidades del cliente.** * **Desarrollo full-stack y construcción de programas de diversa complejidad utilizando técnicas y buenas prácticas de codificación.** * **Diseño e implementación de modelos de datos y arquitecturas sistémicas que soporten los procesos de negocio.** * **Integración de sistemas externos (pasarela de pagos, notificaciones), asegurando compatibilidad e interoperabilidad.** * **Seguridad de sistemas computacionales, aplicando estándares y controles para proteger la información.** * **Pruebas de certificación y aseguramiento de calidad en productos y procesos de software.** * **Gestión de proyectos informáticos, con planificación de fases, cronogramas, roles y riesgos.** |

|  |
| --- |
| **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 CanchaYa busca solucionar la fragmentación y los procesos manuales en la gestión de reservas deportivas, una problemática común en centros deportivos de Chile y de muchas otras regiones. Actualmente, gran parte de las reservas se manejan mediante llamadas telefónicas, mensajes informales o registros en papel, lo que genera ineficiencia, errores y baja trazabilidad. La plataforma digital propuesta automatiza estos procesos, entregando disponibilidad en tiempo real, pagos en línea, notificaciones y reportes administrativos.**  **Esta solución es altamente relevante en el campo laboral de la Ingeniería en Informática, ya que integra competencias claves como desarrollo full-stack, integración de servicios externos, seguridad informática, aseguramiento de calidad y gestión de proyectos. Además, impacta directamente a usuarios de distintos grupos etarios que utilizan instalaciones deportivas, así como a administradores de recintos que requieren mejorar la eficiencia de sus operaciones. El aporte de valor de CanchaYa consiste en ofrecer un sistema robusto, seguro y escalable que optimiza procesos y mejora la experiencia tanto de los clientes como de los administradores.** |
| Descripción del Proyecto APT | **El objetivo de CanchaYa es desarrollar una plataforma web compuesta por dos aplicaciones interconectadas:**   * **Aplicación principal para usuarios finales: permite registro, autenticación, consulta de disponibilidad, reserva de canchas, pagos en línea mediante una pasarela externa, notificaciones automáticas y consulta de historial.** * **Aplicación administrativa (Backoffice): incluye al menos ocho mantenedores (CRUD) para gestionar canchas, horarios, tarifas, promociones, usuarios, operadores y reservas; reportería con filtros y exportación a PDF/XLS; control de accesos por roles y auditoría de acciones.**   **La propuesta se aborda mediante una arquitectura en capas con Django y Oracle, diseño responsivo, validación exhaustiva de formularios, seguridad en sesiones y contraseñas, e integración con servicios REST externos.** |
| Pertinencia del proyecto con el perfil de egreso | **El proyecto se alinea directamente con el perfil de egreso de Ingeniería en Informática, ya que requiere:**   * **Analizar y levantar requerimientos para comprender las necesidades del cliente.** * **Diseñar e implementar arquitecturas y modelos de datos que soporten procesos de negocio.** * **Desarrollar soluciones de software full-stack utilizando buenas prácticas de codificación.** * **Integrar sistemas externos como la pasarela de pagos y los servicios de notificación.** * **Aplicar seguridad informática mediante autorización por roles y buenas prácticas OWASP.** * **Ejecutar pruebas y aseguramiento de calidad en todas las etapas del ciclo de vida del software.** * **Gestionar el proyecto con cronograma, roles, entregables y análisis de riesgos.**   **Todas estas competencias son necesarias para abordar la problemática de digitalizar la gestión de reservas deportivas, demostrando pertinencia formativa y profesional.** |
| Relación con los intereses profesionales | **Como equipo, nuestros intereses profesionales se centran en el desarrollo web full-stack, la integración de servicios externos, la arquitectura de sistemas y la seguridad informática. CanchaYa refleja estos intereses, ya que combina la construcción de un sistema completo (frontend y backend) con la aplicación de criterios de seguridad, reportería avanzada y prácticas de gestión de proyectos. Además, la experiencia adquirida fortalece nuestra proyección profesional al alinearse con las demandas actuales del mercado laboral, como la automatización de procesos, la interoperabilidad de sistemas y la mejora de la experiencia de usuario.** |
| Factibilidad de desarrollo del Proyecto APT | **El proyecto CanchaYa es factible de implementar en el marco de la asignatura, dado que el alcance definido corresponde a un MVP realista que puede desarrollarse dentro de la duración del semestre (11/08/2025 – 05/12/2025). El calendario académico otorga un margen suficiente para abordar las fases de planificación, análisis, diseño, desarrollo, pruebas y cierre, considerando las horas asignadas a la asignatura y el trabajo complementario del equipo fuera del aula.**  **En cuanto a materiales requeridos, el desarrollo se apoya en herramientas disponibles y dominadas por el equipo: Django como framework backend, Oracle como gestor de base de datos, Bootstrap/HTML/CSS para la interfaz responsiva y una API REST para la integración de pagos. Todas estas tecnologías cuentan con documentación extensa, versiones gratuitas o académicas y soporte en línea, por lo que no requieren inversión adicional en licencias. El hardware disponible (computadores personales y acceso a internet) es suficiente para ejecutar los entornos de desarrollo y pruebas.**  **Factores externos que facilitan el desarrollo:**  **Acceso a sandbox y entornos de prueba de la pasarela de pagos, lo que permite integrar y validar sin costos adicionales.**  **Documentación oficial y ejemplos ampliamente disponibles de Django, Oracle y Bootstrap, lo que reduce tiempos de aprendizaje.**  **Apoyo del cuerpo docente, que brinda retroalimentación técnica y metodológica durante las etapas del proyecto.**  **Experiencia previa adquirida en asignaturas anteriores relacionadas con bases de datos, desarrollo web, seguridad y gestión de proyectos.**  **Trabajo colaborativo en equipo con roles definidos y comunicación fluida, lo que facilita la distribución de tareas y el control de avances.**  ***Factores externos que podrían dificultar el desarrollo y sus soluciones:***   * **Disponibilidad del servicio de la pasarela de pagos, que puede presentar caídas o limitaciones en entornos de prueba. → Mitigación: pruebas tempranas, uso de entornos de prueba alternativos y planificación de un fallback simulado.** * **Posibles cambios de requerimientos durante el desarrollo. → Mitigación: mantener comunicación constante con el docente y registrar formalmente cambios para ajustarlos en el cronograma.** * **Sobrecarga de trabajo académico en paralelo con otras asignaturas. → Mitigación: organización semanal con herramientas de gestión, priorización de entregables y trabajo distribuido por sprint.** * **Complejidad en la integración de módulos backend y frontend. → Mitigación: desarrollo incremental, pruebas de integración tempranas y revisiones de código cruzadas.** |

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 | **Desarrollar e implementar una plataforma web compuesta por un módulo de reservas y un backoffice administrativo, integrados con pagos en línea, notificaciones automáticas, reportes exportables y medidas de seguridad informática.** |
| Objetivos específicos | * **Levantar y validar requerimientos funcionales y no funcionales (ERS) y definir la arquitectura del sistema.** * **Desarrollar funcionalidades acompañado de un diseño frontend alineado con la imagen de la empresa** * **Construir el módulo de reservas con disponibilidad en tiempo real.** * **Integrar una pasarela de pagos en línea con validación de seguridad y webhook.** * **Implementar notificaciones automáticas por correo electrónico.** * **Desarrollar el backoffice con al menos ocho CRUD y dos reportes exportables (PDF/XLS).** * **Aplicar medidas de seguridad en contraseñas, sesiones y autorización por roles.** * **Ejecutar pruebas unitarias, de integración y de aceptación para asegurar la calidad del software.** * **Documentar y desplegar el sistema en un entorno productivo con acta de cierre.** |

|  |
| --- |
| **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 |
| **El proyecto se abordará utilizando una metodología incremental basada en fases:**  **Fase de Planificación/Definición: levantamiento de requerimientos, acta de constitución, cronograma y gestión de riesgos.**  **Fase de Análisis–Diseño–Desarrollo: modelado de la arquitectura y BD, construcción de módulos principales, integración de pagos y reportería.**  **Fase de Pruebas–Despliegue–Cierre: ejecución de pruebas, correcciones, documentación final, despliegue en entorno productivo y acta de aceptación.**  **Roles del equipo:**   * **Jefe de Proyecto: coordina el plan, gestiona riesgos y asegura cumplimiento de entregables.** * **Desarrollador Frontend/UX: diseña e implementa interfaces responsivas y accesibles.** * **Desarrollador Backend/Seguridad: construye la lógica del negocio, integra la pasarela de pagos e implementa medidas de seguridad.** * **Arquitecto de Servicios e Integración: define la arquitectura, modelo de datos y garantiza interoperabilidad de sistemas.** * **QA (Quality Assurance): diseña y ejecuta pruebas (unitarias, integración, aceptación) y verifica calidad en los entregables.** |

|  |
| --- |
| **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** | **Acta de constitución del proyecto** | **Documento formal que define objetivos, alcance, roles, entregables y riesgos iniciales del proyecto***.* | **Permite establecer los lineamientos del proyecto** |
| **Avance** | **Carta gantt** | **Cronograma con fases, tareas, responsables y tiempos estimados.** | **Permite visualizar y controlar la planificación temporal del proyecto.** |
| **Avance** | **Planilla de Requerimientos** | **Registro detallado de requerimientos funcionales y no funcionales, actores y estado** | **Garantiza que el sistema satisfaga las necesidades de los usuarios y del negocio.** |
| **Avance** | **Documento de Casos de Uso Extendidos** | **Descripción estructurada de escenarios CU001–CU041, con actores, flujos y excepciones.** | **Asegura trazabilidad entre requerimientos y funcionalidades específicas del sistema.** |
| **Avance** | **Mockups** | **Prototipos de interfaz gráfica responsiva y navegables** | **Facilita la validación temprana con los usuarios y reduce riesgos de rediseño tardío.** |
| **Avance** | **Matriz EDT** | **Desglose jerárquico de fases, actividades y entregables con responsables.** | **Sirve como guía de control del alcance y facilita la asignación de tareas por rol.** |
| **Avance** | **Informe ERS** | **Documento formal que integra RF, RNF, casos de uso, restricciones y criterios de aceptación.** | **Consolida toda la información de análisis y permite la trazabilidad hacia el diseño y pruebas.** |
| **Avance** | **Minutas de Reunión y Avances de Sprint** | **Registros periódicos de reuniones, acuerdos, dificultades y avances iterativos.** | **Evidencian la gestión colaborativa y el seguimiento del proyecto en cada sprint.** |
| **Avance** | **Código Fuente Versionado** | **Desarrollo de la aplicación (frontend, backend y seguridad) en repositorio con control de versiones.** | **Evidencia tangible del producto implementado y permite trazabilidad de cambios.** |
| **Avance** | **Reportes del Sistema (PDF/XLS)** | **Reportes generados con filtros y exportación desde el backoffice.** | **exportación desde el backoffice.**  **Demuestran el cumplimiento de los requerimientos de reportería definidos.** |
| **Avance** | **Integración con Pasarela de Pagos** | **Documentación y prueba de integración con API externa de pagos, incluyendo validaciones de seguridad.** | **Refleja un componente crítico del proyecto y su factibilidad en condiciones reales.** |
| **Finales** | **Plan y Reporte de Pruebas** | **Documento con casos de prueba ejecutados (unitarias, integración, aceptación) y sus resultados.** | **Garantiza calidad del software y cumplimiento de requisitos.** |
| **Finales** | **Manuales de Usuario y Administrador** | **Documentos explicativos para la operación de la aplicación principal y el backoffice.** | **Aseguran usabilidad, capacitación y adopción de la solución.** |
| **Finales** | **Acta de cierre** | **Documento formal de aceptación final del proyecto.** | **Da término oficial al proyecto y valida la satisfacción de objetivos y entregables.** |

|  |
| --- |
| **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 |
| Gestión de Proyectos | Presentación inicial del proyecto | Presentar objetivos, alcance y plan preliminar al equipo y stakeholders | Presentación PowerPoint, acta inicial | 1 día | Fabián Bravo | Requiere preparación previa de información; coordinación con el equipo |
| Gestión de Proyectos | Elaborar Acta de Constitución | Redactar acta con objetivos, alcance, entregables, riesgos y recursos | Plantilla de acta, documentación previa | 2 días | Fabián Bravo | Puede requerir validación con cliente |
| Gestión de Proyectos | Elaborar Carta Gantt | Diseñar cronograma del proyecto con actividades, dependencias y tiempos | Microsoft Project | 2 días | Fabián Bravo | Requiere conocer todas las actividades y su duración |
| Análisis de Requisitos | Revisión y consolidación de requerimientos | Consolidar requerimientos funcionales y no funcionales | Documentación preliminar, reuniones con stakeholders | 5 días | Fabián Bravo | Riesgo de información incompleta si los stakeholders no colaboran |
| Análisis de Requisitos | Casos de uso (redacción y validación) | Crear y validar diagramas y descripciones de casos de uso | Plantilla de casos de uso, diagramas | 5 días | Fabián Bravo | Coordinar con equipo de desarrollo |
| Análisis de Requisitos | Criterios de aceptación y priorización | Definir criterios de aceptación y priorizar requerimientos | Reuniones con cliente, planilla de requerimientos | 2 días | Fabián Bravo | Acordar prioridades con stakeholders |
| Diseño de Sistema | Arquitectura en capas (presentación, negocio, datos) | Definir arquitectura del sistema en capas | Diagramas, herramientas de modelado | 5 días | Daniel Gálvez | Coordinar con backend y frontend |
| Diseño de Sistema | Modelo de datos Oracle (DDL, constraints) | Crear estructura de base de datos con tablas, relaciones y restricciones | Oracle, SQL Developer | 3 días | Daniel Gálvez | Validar consistencia de integridad referencial |
| Diseño de Sistema | Seguridad | Definir roles, permisos y enmascaramiento de datos sensibles | Documentación de seguridad, herramientas de modelado | 2 días | Daniel Gálvez | Considerar protección de información sensible |
| Diseño de Sistema | Wireframes y prototipos UI | Diseñar prototipos de interfaz y flujos de navegación | Figma, Adobe XD | 3 días | Daniel Gálvez | Validar usabilidad con equipo interno |
| Desarrollo Backend | Autenticación y perfiles | Implementar registro, login, manejo de sesiones y perfiles | Django, Oracle | 8 días | Daniel Gálvez | Validar seguridad y consistencia de datos |
| Desarrollo Backend | Disponibilidad y reservas | Implementar control de bloques horarios y reservas | Django, Oracle | 8 días | Daniel Gálvez | Coordinar con tabla DisponibilidadCancha |
| Desarrollo Backend | Pagos e integración REST | Integrar pagos electrónicos y validar transacciones | API Mercado Pago, Postman | 6 días | Daniel Gálvez | Validar correctamente webhook/IPN |
| Desarrollo Backend | Notificaciones | Implementar envío de correos y alertas | Django, SMTP | 6 días | Daniel Gálvez | Revisar envío oportuno y formato |
| Desarrollo Backend | Historial de reservas | Mostrar reservas pasadas, activas y canceladas | Django, Oracle | 3 días | Daniel Gálvez | Validar consultas y filtros |
| Desarrollo Backend | Perfil de usuario (ver/editar) | Permitir actualizar información personal | Django, ReactJS | 3 días | Daniel Gálvez | Validar consistencia de datos |
| Desarrollo Frontend | Validaciones front/back (formularios) | Implementar validaciones de datos en frontend y backend | ReactJS, Django | 4 días | Daniel Gálvez | Prevenir ingreso de datos incorrectos |
| Desarrollo Integración | Integración App Principal (ensamble de módulos) | Integrar todos los módulos desarrollados de la aplicación principal | ReactJS, Django | 5 días | Daniel Gálvez / Fabián Bravo | Coordinar que funcionalidades trabajen juntas |
| Desarrollo Backend | Canchas y Horarios | CRUD canchas y horarios | Django, Oracle | 6 días | Fabián Bravo | Validar creación automática de bloques horarios |
| Desarrollo Backend | Tarifas y Promociones | CRUD tarifas y promociones | Django, Oracle | 5 días | Fabián Bravo | Coordinar con módulo de reservas |
| Desarrollo Backend | Usuarios finales | CRUD usuarios | Django, Oracle | 5 días | Fabián Bravo | Validar roles y permisos |
| Desarrollo Backend | Administradores | CRUD administradores | Django, Oracle | 5 días | Fabián Bravo | Coordinar seguridad y acceso |
| Desarrollo Backend | Operadores | CRUD operadores | Django, Oracle | 5 días | Fabián Bravo | Roles con permisos limitados |
| Desarrollo Backend | Equipamiento deportivo | CRUD equipamiento | Django, Oracle | 5 días | Daniel Gálvez / Fabián Bravo | Control de inventario |
| Desarrollo Backend | Reservas (admin) | CRUD reservas desde backoffice | Django, Oracle | 6 días | Daniel Gálvez / Fabián Bravo | Integrar con disponibilidad de canchas |
| Desarrollo Backend | Roles/Perfiles (seguridad backoffice) | Asignar roles y permisos administrativos | Django | 4 días | Daniel Gálvez / Fabián Bravo | Control granular de acceso |
| Desarrollo Integración | Integración App Secundaria | Ensamblar todos los módulos de backoffice | Django, ReactJS | 5 días | Daniel Gálvez / David Espinosa | Coordinar comunicación con DB compartida |
| Desarrollo Backend | Reporte de ingresos | Generar y mostrar reportes de ingresos | Django, Oracle | 4 días | Daniel Gálvez / Fabián Bravo | Validar cálculos y filtros |
| Desarrollo Backend | Exportación ingresos a PDF | Generar reporte de ingresos en PDF | Django, librerías PDF | 2 días | Daniel Gálvez / Fabián Bravo | Verificar formato y consistencia |
| Desarrollo Backend | Exportación ingresos a XLS | Generar reporte de ingresos en Excel | Django, librerías XLS | 2 días | Daniel Gálvez / Fabián Bravo | Revisar compatibilidad con Excel |
| Desarrollo Backend | Reporte de ocupación | Generar reportes de ocupación de canchas | Django, Oracle | 4 días | Daniel Gálvez / Fabián Bravo | Validar filtros de fechas y canchas |
| Desarrollo Backend | Exportación ocupación a PDF | Exportar reporte ocupación a PDF | Django, librerías PDF | 2 días | Daniel Gálvez / Fabián Bravo | Verificar consistencia de datos |
| Desarrollo Backend | Exportación ocupación a XLS | Exportar reporte ocupación a Excel | Django, librerías XLS | 2 días | Daniel Gálvez / Fabián Bravo | Revisar compatibilidad con Excel |
| Seguridad | Hardening seguridad | Asegurar el sistema contra vulnerabilidades | Herramientas de seguridad, documentación | 4 días | Daniel Gálvez / Fabián Bravo | Revisar configuración de roles y permisos |
| Rendimiento | Rendimiento | Optimizar tiempos de respuesta y consultas | Pruebas de carga, herramientas de monitoreo | 3 días | Daniel Gálvez / Fabián Bravo | Monitorear performance durante pruebas |
| Disponibilidad | Disponibilidad / manejo de errores | Implementar manejo de errores y disponibilidad | Django, Oracle | 2 días | Daniel Gálvez / Fabián Bravo | Control de errores y logs |
| Escalabilidad | Escalabilidad (pruebas básicas de carga) | Validar rendimiento ante múltiples usuarios | Pruebas de carga | 2 días | Daniel Gálvez / Fabián Bravo | Detectar cuellos de botella |
| Compatibilidad | Responsivo y compatibilidad | Verificar funcionamiento en diferentes dispositivos y navegadores | Navegadores, dispositivos móviles | 3 días | Daniel Gálvez / Fabián Bravo | Ajustar diseño responsivo |
| Privacidad | Privacidad y cumplimiento | Garantizar protección de datos personales | Documentación legal, configuración DB | 3 días | Daniel Gálvez / Fabián Bravo | Cumplir con normativa vigente |
| Mantenibilidad | Mantenibilidad y modularidad | Código modular, documentado y organizado | IDE, documentación | 2 días | Daniel Gálvez / Fabián Bravo | Facilitar futuras modificaciones |
| Integridad de datos | Integridad de datos (transacciones/constraints) | Validar consistencia de datos y transacciones | Oracle, Django | 3 días | Daniel Gálvez / Fabián Bravo | Evitar corrupción o pérdida de datos |
| QA | Plan de pruebas | Definir plan de pruebas unitarias, integración y aceptación | Plantilla de plan de pruebas | 3 días | Daniel Gálvez / Fabián Bravo | Coordinar con equipo de desarrollo |
| QA | Pruebas unitarias | Probar módulos individualmente | Ambiente de test, casos de prueba | 4 días | Daniel Gálvez / Fabián Bravo | Detectar errores en cada módulo |
| QA | Pruebas de integración | Probar interacción entre módulos | Ambiente de test, casos de prueba | 4 días | Daniel Gálvez / Fabián Bravo | Verificar comunicación entre backend y frontend |
| QA | Pruebas de aceptación (UAT) | Validar sistema con criterios de aceptación | Ambiente de test, casos de prueba | 3 días | Daniel Gálvez / Fabián Bravo | Coordinar con cliente |
| QA | Corrección de defectos | Arreglar errores detectados en pruebas | IDE, documentación de errores | 4 días | Daniel Gálvez / Fabián Bravo | Coordinar con QA y frontend |
| QA | Cierre de pruebas (hito) | Concluir fase de pruebas y generar informe | Documentación de pruebas | 1 día | Daniel Gálvez / Fabián Bravo | Revisar que todos los módulos estén validados |
| Implementación | Migración/carga inicial de datos | Cargar datos iniciales en base de datos | Oracle, scripts SQL | 2 días | Daniel Gálvez / Fabián Bravo | Verificar consistencia de datos |
| Implementación | Manuales (usuario, admin, técnico) | Redactar manuales de usuario, administrador y técnico | Documentación, plantillas | 4 días | Daniel Gálvez / Fabián Bravo | Revisar claridad y detalle |
| Implementación | Capacitación administradores/operadores | Entrenar a usuarios internos en uso del sistema | Sistema en pruebas, guías | 2 días | Daniel Gálvez / Fabián Bravo | Coordinar disponibilidad de participantes |
| Implementación | Go-Live (hito) | Desplegar sistema en producción | Sistema final, servidor local | 1 día | Fabian Bravo | Verificar correcto funcionamiento del sistema |
| Cierre | Cierre del proyecto (hito) | Generar acta final y entregar documentación completa | Acta, manuales, sistema implementado | 1 día | Fabian Bravo | Confirmar aceptación del cliente |

|  |
| --- |
| **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** | | |
|  | **S1** | **S2** | **S3** | **S4** | **S5** | **S6** | **S7** | **S8** | **S9** | **S10** | **S11** | **S12** | **S13** | **S14** | **S15** | **S16** | **S17** | **S18** |
| **Presentación inicial del proyecto** | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Elaboración Carta Gantt** | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Planilla de requerimientos** |  | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Casos de Uso extendidos** |  |  | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **ERS (Especificación de Requerimientos de Software)** |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **EDT (Estructura de Desglose de Trabajo)** |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Mockups del sistema completo** |  |  | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Acta constitución del proyecto** |  | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Definición proyecto APT (Español)** |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Definición proyecto APT (Inglés)** |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Autoevaluaciones Fase 1 y Competencias** |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Documento de arquitectura (DAS)** |  |  |  |  | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  |  |
| **Documento proceso de negocio (TO-BE)** |  |  |  |  | **X** | **X** |  |  |  |  |  |  |  |  |  |  |  |  |
| **Matriz RACI** |  |  |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |  |
| **Matriz de riesgo** |  |  |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |  |
| **Diccionario de datos** |  |  |  |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |
| **Minuta de reuniones** |  |  |  |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |  |
| **Matriz control de cambios** |  |  |  |  |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |
| **PPT presentación avance del proyecto** |  |  |  |  |  |  |  | **X** |  |  |  |  |  |  |  |  |  |  |
| **Desarrollo 100% del sistema** |  |  |  |  |  |  |  |  | **X** | **X** | **X** | **X** | **X** |  |  |  |  |  |
| **Script de base de datos** |  |  |  |  |  |  |  |  | **X** | **X** |  |  |  |  |  |  |  |  |
| **Documento avance desarrollo APT (Español)** |  |  |  |  |  |  |  |  |  | **X** | **X** |  |  |  |  |  |  |  |
| **Documento avance desarrollo APT (Inglés)** |  |  |  |  |  |  |  |  |  | **X** | **X** |  |  |  |  |  |  |  |
| **Informe final proyecto APT (Español)** |  |  |  |  |  |  |  |  |  |  | **X** | **X** |  |  |  |  |  |  |
| **Informe final proyecto APT (Inglés)** |  |  |  |  |  |  |  |  |  |  | **X** | **X** |  |  |  |  |  |  |
| **Pauta de reflexión Fase 2** |  |  |  |  |  |  |  |  |  |  |  | **X** | **X** |  |  |  |  |  |
| **Matriz de control de cambios** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |  |
| **Minuta de reuniones** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |  |
| **Plan de pruebas** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |  |
| **Matriz pruebas de base de datos** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |  |
| **Planilla registro defectos** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |  |
| **Verificación de alcances** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |  |
| **Plan de capacitación** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |  |
| **Manual de usuario** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |  |
| **Informe cierre del proyecto** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |
| **Documento PTY4478 Fase 2 – Informe final (Español)** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |
| **Documento PTY4478 Fase 2 – Informe final (Inglés)** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** |  |
| **PPT presentación final del proyecto** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **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-1)