**Guía2. Desarrollo Proyecto APT**

**Asignatura Capstone**

|  |
| --- |
| **1. Resumen avance Proyecto APT** |
| A continuación, encontrarás distintos campos que deberás completar con la información solicitada. |

|  |  |
| --- | --- |
| Resumen de avance proyecto APT | *Durante la Fase 2 del proyecto* ***“Calendario Plazos Judiciales”*** *se avanzó en: análisis y levantamiento de requisitos; diseño de arquitectura; modelo de datos; implementación del* ***motor de cálculo de plazos*** *(hábiles, corridos, feriados, suspensiones); desarrollo de interfaz para registro de causas y calendario; autenticación y resguardo de datos sensibles; y sistema de alertas/recordatorios. Se ejecutaron ajustes técnicos puntuales (p. ej.,* ***migración de MySQL a PostgreSQL*** *y de Trello a GitHub Projects) sin cambiar el objetivo general.* |
| Objetivos | ***Sin ajustes*** *respecto a Fase 1.*  *Se mantienen: (1) desarrollar un software que* ***automatice la gestión de plazos****; (2) registrar causas y vincularlas al Poder Judicial; (3) calcular plazos según normativa; (4) alertas y recordatorios; (5) gestión de suspensiones/notificaciones; (6) almacenamiento seguro de información; (7) personalización de horarios.* |
| Metodología | ***Sin ajustes****: metodología ágil* ***SCRUM****, trabajo por* ***sprints*** *con entregables funcionales (análisis, diseño, desarrollo, pruebas y documentación). Pertinente para validar con usuarios, adaptarse a cambios y mantener entregas incrementales.* |
| Evidencias de avance | ***Código fuente funcional (backend y frontend).***  ***Base de datos operativa con entidades clave (usuarios, causas, plazos, tribunales).***  ***Interfaz: formularios y calendario de plazos.***  ***Pruebas BDD (Gherkin/Behave) y unitarias; reportes de ejecución.***  ***Documentación técnica y README de instalación/uso.***  ***Justificación: evidencian implementación de requisitos y calidad por metodología ágil, pruebas automatizadas, control de versiones y validación normativa del motor de plazos.*** |

|  |
| --- |
| **2. Monitoreo del Plan de Trabajo** |
| Examina cuidadosamente tu plan de trabajo, enfocándote especialmente en la columna de estado de avance y ajustes. |

| **Competencia o unidades de competencias** | **Actividades** | **Recursos** | **Duración de la actividad** | **Responsable** | **Observaciones** | **Estado de avance** | **Ajustes** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Desarrollo de software y programación | Levantamiento de requisitos | Reuniones, documentación legal | 1 semana | Jairo | Información completa obtenida | Completado | Ninguno |
| Análisis y diseño de sistemas | Diseño de base de datos y lógica de plazos | Herramientas UML, MySQL Workbench | 2 semanas | Matías | Validado con normativa vigente | Completado | Migración a PostgreSQL |
| Gestión de proyectos TI | Organización de trabajo en Trello (SCRUM) | Trello, metodología SCRUM | 1 semana | Benjamín | Coordinación efectiva lograda | Completado | Migración a GitHub Projects |
| Diseño de interfaz | Desarrollo del frontend e interfaz | Figma, HTML, CSS, JavaScript | 2 semanas | Benjamín | Interfaz responsiva implementada | Completado | Mejoras UX adicionales |
| Lógica de negocio | Implementación del motor de cálculo | Django, Python, API feriados | 3 semanas | Equipo | Casos reales validados exitosamente | Completado | Optimización de algoritmos |
| Seguridad de información | Seguridad de credenciales | bcrypt, JWT, Django Auth | 1 semana | Matías | Buenas prácticas implementadas | Completado | Ninguno |
| Pruebas de software | Pruebas funcionales | Behave, Gherkin, unittest | 2 semanas | Matías | Casos con feriados incluidos | En curso | Ampliación de cobertura |
| Documentación técnica | Elaboración del manual de usuario | Markdown, capturas de pantalla | 1 semana | Equipo | Explicaciones claras incluidas | En curso | Versión final pendiente |
| Comunicación efectiva | Preparación de presentación final | Diapositivas, demo funcional | 1 semana | Benjamín | Ensayos programados | No iniciado | Programado para Fase 3 |

|  |
| --- |
| **3. Ajustes a partir del monitoreo** |
| Profundiza en las observaciones de tu plan de trabajo. Analiza las actividades planificadas y señala qué aspectos facilitaron u obstaculizaron la ejecución del plan. Plantea cómo abordaste y/o abordarás los obstáculos. Por último, señala los ajustes que realizaste al plan de trabajo a partir de este análisis. |

|  |
| --- |
| **Factores que facilitaron:**   * Experiencia técnica previa del equipo en desarrollo web y backend. * Acceso a documentación legal completa y actualizada. * Uso de herramientas modernas: Django, PostgreSQL, GitHub Projects. * Metodología ágil SCRUM que permitió adaptabilidad y entregas continuas.   **Factores que dificultaron:**   * Complejidad normativa chilena (distintos tipos de plazos y excepciones). * Limitaciones de acceso a APIs del Poder Judicial. * Coordinación horaria del equipo durante sprints.   **Acciones tomadas:**   * Validación de reglas con apoyo de profesional del área legal. * Implementación de *web scraping* como alternativa temporal a la API. * Reuniones semanales fijas y comunicación vía Discord.   **Ajustes realizados:**   * Migración de base de datos a PostgreSQL para mayor estabilidad. * Integración del tablero a GitHub Projects para mejor control de versiones. * Mejoras UX en interfaz basadas en feedback inicial. * Extensión del periodo de pruebas para cobertura completa. |

|  |
| --- |
| **Ajustes realizados:**  Base de datos: se migró de MySQL a PostgreSQL, debido a su mejor integración con Django y mayor estabilidad en entornos de desarrollo.  Gestión del proyecto: se reemplazó Trello por GitHub Projects, permitiendo vincular tareas directamente al código y gestionar incidencias.  Interfaz de usuario: se añadieron mejoras de UX y responsividad, basadas en retroalimentación de usuarios de prueba.  Motor de cálculo: se optimizaron algoritmos para manejar grandes volúmenes de datos y reducir tiempos de ejecución.  **Justificación:**  Estos ajustes se realizaron para mejorar la calidad técnica y el rendimiento del sistema sin alterar los objetivos planteados. Las nuevas herramientas ofrecieron una mejor integración y eficiencia en la colaboración del equipo. |

|  |
| --- |
| **Actividades retrasadas:**   * **Pruebas funcionales:** iniciadas, pero extendidas una semana adicional para incorporar más casos de prueba y validar excepciones complejas. * **Manual de usuario:** en desarrollo; se completará junto con las pruebas finales del sistema.   **Motivos del retraso:** La ampliación del alcance de pruebas y la validación de resultados con normativa procesal aumentó el tiempo estimado de esta fase.  **Estrategias para avanzar:**   1. Asignar más horas a la validación de pruebas automatizadas y de usuario. 2. Dividir la documentación entre todos los integrantes del equipo. 3. Comenzar anticipadamente la preparación de la presentación final.   **Conclusión:** No existen actividades sin iniciar que afecten el cumplimiento general del cronograma; el desarrollo se mantiene dentro de los plazos proyectados. |