**INFORME FINAL DEL PROYECTO APT**

**Sistema de Gestión de Incidencias IT**

**1. INFORMACIÓN BÁSICA**

**Nombre del proyecto**

**Sistema de Gestión de Incidencias IT**

**Área(s) de desempeño**

* **Gestión de Servicios TI:** Diseño e implementación de procesos de soporte técnico.
* **Desarrollo de Sistemas de Información:** Creación de software para la automatización de flujos de trabajo.
* **Administración de Infraestructura Tecnológica:** Configuración y mantenimiento de bases de datos y servidores.
* **Gestión de Proyectos Tecnológicos:** Aplicación de metodologías ágiles para la planificación y ejecución.

**Competencias**

1. **Diseño e implementación de soluciones informáticas:** Desarrollo de un sistema web completo con arquitectura escalable.
2. **Administración de servicios y recursos tecnológicos:** Gestión de bases de datos PostgreSQL y APIs RESTful.
3. **Gestión de proyectos tecnológicos:** Uso de Scrum para planificación, ejecución y entrega iterativa.
4. **Aplicación de metodologías ágiles:** Implementación de ceremonias Scrum (Sprint Planning, Daily Meetings, Retrospectivas).

**2. CONTENIDOS DEL INFORME FINAL**

**1. Relevancia del proyecto APT**

**Problema abordado**

En empresas medianas, la gestión de incidencias TI se realiza mediante métodos informales (correos electrónicos, planillas Excel y mensajería instantánea), lo que genera:

* Falta de trazabilidad en el ciclo de vida de las incidencias.
* Tiempos de respuesta prolongados.
* Imposibilidad de medir el desempeño del equipo de soporte.

**Relevancia para el campo laboral**

* **Demanda en el mercado:** Las empresas buscan soluciones escalables para optimizar sus departamentos de TI.
* **Competencias técnicas:** El proyecto integra desarrollo full-stack, gestión de bases de datos y metodologías ágiles.
* **Alineación con tendencias:** Digitalización de procesos operativos y enfoque en experiencia del usuario.

**Contexto de implementación**

* **Ubicación:** Santiago, Región Metropolitana, Chile.
* **Características del entorno:** Empresas con infraestructura TI emergente y altos costos operativos en soporte técnico.
* **Sector objetivo:** Empresas de servicios financieros, logística y retail con equipos internos de TI.

**Población impactada**

* **Usuarios finales:** Personal que reporta incidencias (mejora en tiempos de respuesta y transparencia).
* **Técnicos de soporte:** Reducción de carga administrativa y enfoque en resolución de problemas.
* **Supervisores de TI:** Acceso a métricas para la toma de decisiones estratégicas.

**Aporte de valor**

* **Eficiencia operacional:** Reducción del 40% en el tiempo de resolución de incidencias.
* **Optimización de recursos:** Distribución automatizada de carga de trabajo.
* **Data-driven:** Generación de reportes para identificar problemas recurrentes.

**2. Objetivos**

**Objetivo general**

Desarrollar un sistema web que gestione el ciclo completo de incidencias TI, desde el reporte hasta la resolución, mejorando la eficiencia del soporte técnico en un 40%.

**Objetivos específicos**

1. **Implementar un sistema de autenticación basado en roles** (usuario, técnico, supervisor).
2. **Digitalizar el proceso de creación, asignación y seguimiento de tickets.**
3. **Proporcionar un dashboard en tiempo real con métricas de desempeño.**
4. **Garantizar escalabilidad técnica para futuras integraciones.**

**3. Metodología**

**Metodología utilizada**

**Scrum:**

* **Sprints de 2 semanas:** Iteraciones enfocadas en entregas incrementales.
* **Ceremonias clave:**
  + *Sprint Planning:* Definición de tareas y estimaciones.
  + *Daily Meetings:* Seguimiento diario de avances y bloqueos.
  + *Sprint Review:* Demostración de funcionalidades completadas.
  + *Retrospectiva:* Identificación de mejoras en el proceso.

**Fases del proyecto**

1. **Fase de planificación:** Análisis de requisitos y diseño de arquitectura.
2. **Fase de desarrollo:** Configuración de BD, desarrollo backend/frontend e integración.
3. **Fase de pruebas:** Validación de funcionalidades y calidad del código.
4. **Fase de cierre:** Documentación y entrega del producto mínimo viable (MVP).

**Pertinencia metodológica**

* **Flexibilidad:** Permite ajustes ante imprevistos técnicos.
* **Enfoque colaborativo:** Fomenta la comunicación entre roles técnicos y no técnicos.
* **Entrega de valor continuo:** Cada sprint aporta funcionalidades tangibles.

**4. Desarrollo**

**Etapas y actividades**

1. **Análisis de requisitos:**
   * Entrevistas con usuarios potenciales.
   * Definición de user stories y criterios de aceptación.
2. **Diseño técnico:**
   * Arquitectura de base de datos (PostgreSQL).
   * Diseño de APIs RESTful y esquema de autenticación JWT.
3. **Implementación:**
   * Backend: Node.js/Express para lógica de negocio.
   * Frontend: React.js para interfaz de usuario.
4. **Pruebas:**
   * Validación de integración frontend-backend.
   * Pruebas de usabilidad y compatibilidad cross-browser.

**Facilitadores**

* **Comunicación diaria:** Daily Meetings para alineación del equipo.
* **Especialización de roles:** Benjamin (backend), Juan (frontend), Diego (gestión).
* **Herramientas colaborativas:** GitHub, Google Workspace y Slack.

**Dificultades y soluciones**

1. **Bloqueos técnicos:**
   * *Problema:* Acceso a servidor de base de datos.
   * *Solución:* Gestión proactiva con el administrador de infraestructura.
2. **Subestimación de tiempos:**
   * *Problema:* Complejidad en autenticación por roles.
   * *Solución:* Buffer del 20% en estimaciones futuras.
3. **Integración frontend-backend:**
   * *Problema:* Inconsistencias en contratos de APIs.
   * *Solución:* Reuniones técnicas y documentación detallada.

**Ajustes realizados**

* **Estimaciones:** Incremento del 20% en horas planificadas.
* **Procesos técnicos:** Implementación de reuniones de sincronización semanales.
* **Calidad:** Revisiones de código obligatorias antes de integración.

**5. Evidencias**

**Evidencias adjuntas**

1. **Repositorio de código:** [Enlace a GitHub] con commits organizados por funcionalidades.
2. **Documentación técnica:** Manual de instalación y arquitectura del sistema.
3. **Registros de gestión:**
   * Sprint Backlog y planificación.
   * Registros de Daily Meetings y Retrospectivas.
4. **Demostración funcional:** Video del flujo completo de gestión de tickets.

**6. Intereses y proyecciones profesionales**

**Reflexión sobre intereses profesionales**

* **Confirmación de intereses:** Gestión de proyectos tecnológicos y liderazgo de equipos ágiles.
* **Nuevos descubrimientos:** Valor de la experiencia de usuario (UX) en soluciones técnicas.

**Proyecciones laborales**

* **Roles目标:** Scrum Master, Product Owner o Technical Project Manager.
* **Sectores de interés:** Startups tecnológicas y empresas en procesos de transformación digital.
* **Habilidades a desarrollar:**
  + Especialización en frameworks ágiles (Scrum, Kanban).
  + Gestión de productos digitales (Product Management).

**Conclusión**

El proyecto **Sistema de Gestión de Incidencias IT** demostró la viabilidad de aplicar metodologías ágiles y tecnologías modernas para resolver problemas operativos en entornos empresariales. Los resultados validaron no solo las compet técnicas del equipo, sino también la importancia de la gestión colaborativa y la adaptabilidad ante desafíos imprevistos.