**INFORME TÉCNICO DEL PROYECTO APT**

***Sistema de Monitoreo Hospitalario con Integración Helpnex***

**Estudiantes:** Brian Arenas

Bruno Garrido

Jhon Herrera

**Carrera:** Ingeniería en Informática  
**Asignatura:** Capstone  
**Año:** 2025

**ÍNDICE**

1. Abstract
2. Descripción del Proyecto y Relevancia
3. Ajustes a la Propuesta
4. Metodología de Trabajo
5. Evidencias del Avance
6. Desarrollo de Ingeniería
7. Conclusiones y Reflexiones Individuales

**1. Abstract**

El Proyecto APT tiene como propósito desarrollar una plataforma de monitoreo hospitalario capaz de visualizar y controlar en tiempo real el estado de los equipos tecnológicos conectados al sistema Helpnex.  
La solución busca optimizar la gestión tecnológica hospitalaria, permitiendo detectar fallas, generar alertas y supervisar los dispositivos de infraestructura crítica desde una interfaz unificada.

El desarrollo se basa en la metodología ágil Scrum, aplicando herramientas modernas como Python (Django/FastAPI), PostgreSQL y API REST, bajo un enfoque modular y escalable.  
El proyecto representa una integración entre la ingeniería informática y la gestión hospitalaria, generando una herramienta de alto impacto para la eficiencia operativa del entorno de salud.

**2. Descripción del Proyecto y Relevancia**

El Sistema de Monitoreo Hospitalario Helpnex es una plataforma tecnológica diseñada para centralizar el estado y rendimiento de los equipos tecnológicos de un hospital, tales como cámaras, puntos de red, controles de acceso y llamados de enfermería.

El sistema permite al personal técnico detectar fallas en tiempo real, emitir alertas automáticas y generar reportes de rendimiento, lo que reduce los tiempos de respuesta y mejora la continuidad de los servicios críticos.  
Su relevancia radica en que responde a una necesidad real de los entornos hospitalarios: contar con sistemas inteligentes e integrados que apoyen la toma de decisiones y la gestión eficiente de recursos tecnológicos.

**3. Ajustes a la Propuesta**

Durante el desarrollo del proyecto se realizaron ajustes importantes derivados de las dificultades técnicas, los factores facilitadores y la retroalimentación recibida.

Dificultades identificadas:

* Problemas de compatibilidad entre la API Helpnex y la base de datos PostgreSQL durante la sincronización de datos.
* Inconsistencias en los formatos de respuesta JSON de la API, lo que exigió pruebas adicionales.
* Necesidad de ampliar el tiempo de prueba en los módulos de monitoreo y alertas.

**Factores facilitadores:**

* Experiencia previa en programación backend y modelado de bases de datos.
* Acceso a documentación técnica actualizada de Helpnex.
* Buen nivel de comunicación y coordinación con el equipo de trabajo.

**Ajustes realizados:**

* Se implementó un módulo de simulación de datos para continuar con el desarrollo mientras se resolvían las integraciones reales.
* Se refactorizó el modelo entidad-relación, agregando nuevas relaciones para mejorar la trazabilidad de alertas.
* Se priorizó la fase de pruebas y optimización dentro de los últimos sprints, extendiendo su duración para asegurar estabilidad del sistema.

Estos cambios permitieron mantener el cumplimiento del cronograma y garantizar la calidad del producto final.

**4. Metodología de Trabajo**

La metodología utilizada es Scrum, una metodología ágil que permite una planificación flexible y adaptable a las necesidades del proyecto.

El trabajo se organizó en 9 sprints de 2 semanas cada uno, cubriendo una duración total de 18 semanas.  
Cada sprint contempló planificación, desarrollo, revisión y retrospectiva, promoviendo la mejora continua.

**Estructura de trabajo:**

|  |  |  |
| --- | --- | --- |
| **Fase** | **Actividad principal** | **Entregable** |
| Sprint 1 | Definición del alcance, backlog y requerimientos | Documento inicial de planificación |
| Sprint 2 | Diseño del modelo ER y arquitectura del sistema | Diagrama entidad-relación |
| Sprint 3 | Configuración de entorno y base de datos | Base de datos funcional |
| Sprint 4 | Desarrollo de módulo de autenticación | Login operativo |
| Sprint 5 | Dashboard de monitoreo | Panel con datos de prueba |
| Sprint 6 | Integración API Helpnex | Sincronización parcial |
| Sprint 7 | Módulo de alertas | Generación y gestión de alertas |
| Sprint 8 | Pruebas finales y ajustes | Informe de pruebas |
| Sprint 9 | Documentación y entrega final | Informe técnico y presentación |

Scrum resultó pertinente al requerir una constante adaptación técnica y validación funcional, características esenciales en proyectos de ingeniería informática aplicada.

**5. Evidencias del Avance**

Durante la ejecución del proyecto se generaron múltiples evidencias que demuestran el progreso y cumplimiento de los objetivos:

* Documento de requerimientos: Contiene épicas, historias de usuario y requisitos funcionales/no funcionales.
* Modelo entidad-relación (MER): Diagrama estructurado de base de datos, validado en Oracle Data Modeler.
* Prototipo del dashboard web: Pantalla principal con visualización de dispositivos y alertas.
* Bitácora de Dailies y Sprints: Seguimiento semanal de actividades, dificultades y avances.

Estas evidencias fueron utilizadas para validar la evolución del proyecto y la coherencia entre los entregables y la planificación original.

**6. Desarrollo de Ingeniería**

El desarrollo del sistema combina componentes de software, base de datos e integración API.  
Los módulos principales incluyen:

* Autenticación de usuarios y control de roles.
* Dashboard dinámico con actualización de estados de dispositivos en tiempo real.
* Gestión de alertas automáticas, generadas a partir de eventos Helpnex.
* Reportes y analítica, orientados a la gestión y supervisión del sistema.
* Configuración y sincronización con los endpoints de la API Helpnex.

La base de datos relacional fue construida en PostgreSQL, con tablas de usuarios, dispositivos, áreas, alertas y registros de sincronización.  
El sistema se diseñó bajo una arquitectura modular, que permite escalar el proyecto y agregar nuevas funcionalidades sin afectar la estructura principal.

El uso de Python (Django/FastAPI) para el backend garantiza compatibilidad con API REST, mientras que HTML, CSS y JavaScript ofrecen una interfaz intuitiva y adaptable a distintos dispositivos.

**7. Conclusiones y Reflexiones Individuales**

El desarrollo del Proyecto APT permitió integrar de manera práctica conocimientos de ingeniería de software, redes y gestión tecnológica, aplicándolos a un entorno real de alta exigencia como es el hospitalario.  
La aplicación de Scrum facilitó la gestión del tiempo y la priorización de tareas, asegurando entregas funcionales en cada sprint.

Entre los principales aprendizajes se destacan la integración de APIs, el diseño de sistemas escalables y la resolución de problemas técnicos mediante iteracióncontinua.  
Asimismo, se fortalecieron habilidades de trabajo colaborativo, documentación técnica y análisis de procesos.

Desde una perspectiva personal, el proyecto me permitió confirmar mi interés por el desarrollo backend, la administración de sistemas tecnológicos y la automatización de infraestructura crítica.  
A futuro, me gustaría continuar perfeccionándome en áreas como ciberseguridad, automatización y monitoreo inteligente, aplicando la experiencia obtenida en este proyecto.

En conclusión, el proyecto Helpnex no solo cumple los objetivos académicos, sino que representa un aporte real y aplicable al ámbito profesional del Ingeniero en Informática, promoviendo la innovación, la eficiencia y la gestión tecnológica avanzada.