



Informe de Cierre: Sprint 2

Proyecto: Qualifast Buildings – Estación de Policía Inteligente

Período: 2 Semanas (Continuación Sprint 1)

Fecha de Revisión: 23 de Noviembre de 2025

Estado: Completado (100% de los objetivos alcanzados)

Equipo: 4 Integrantes

Dedicación total: ~160 horas (40h/persona)

🎯 Objetivo del Sprint

El foco principal de este segundo Sprint fue dotar de interactividad y dinamismo a la aplicación.

Mientras el Sprint 1 estableció la estructura, el Sprint 2 se centró en:

1. **Gestión de Entidades:** Implementar el CRUD para la gestión de presos.
 2. **Simulación IoT:** Crear un entorno de datos simulados para probar la aplicación sin hardware físico conectado.
 3. **Dashboard en Tiempo Real:** Visualizar datos y logs en la interfaz principal.
 4. **Integración de Cámara:** Añadir la vista y control del sistema de videovigilancia.
-

👤 Resumen de Esfuerzo y Equipo

La distribución de tareas se realizó por módulos funcionales para evitar conflictos en el control de versiones.

Rol	Miembro	Responsabilidad Principal en Sprint 2

Scrum Master	Nicolás Ramírez	Desarrollo del script de simulación (simulador_iot.py) y generación de datos aleatorios.
Product Owner	Álvaro López	Implementación completa del módulo de Presos (Modelo, Vista y Controlador).
Dev IoT / Backend	Fernando Fernández	Lógica de la cámara de vigilancia y conexión de su controlador.
QA / Dev	Daniel Vicente	Diseño del Dashboard principal, panel de logs y conexión de sensores visuales.



Backlog del Sprint y Tareas Realizadas

A continuación, se detallan las tareas completadas extraídas del tablero Trello, agrupadas por las nuevas Historias de Usuario (HU).

● HU-05: Gestión de Presos (CRUD)

Objetivo: Permitir a los agentes registrar y dar de baja detenidos en el sistema.

Estado	Tarea	Responsable	Entregable
✓	Implementar Modelo de Presos	Álvaro	Funciones add_preso, get_presos,

			delete_preso en manejador_datos.py.
✓	Diseñar Vista de Presos	Álvaro	Archivo vista_gestion_presos.py con listado y buscador.
✓	Conectar Controlador de Presos	Álvaro	Lógica en main.py para enlazar botones con el modelo.

● HU-06: Simulación y Datos de Sensores

Objetivo: Generar datos realistas de temperatura, humo y luminosidad para pruebas.

Estado	Tarea	Responsable	Entregable
✓	Crear Script de Simulación	Nicolás	Script simulador_iot.py ejecutándose en hilo paralelo.
✓	Generar Datos de Sensores	Nicolás	Escritura automática en sensores_log.json con timestamps reales.
✓	Simular Cámara	Nicolás	Generación de estados simulados para el dispositivo de cámara.

● HU-07: Dashboard de Monitorización

Objetivo: Centralizar la información en una pantalla principal intuitiva.

Estado	Tarea	Responsable	Entregable
✓	Implementar Modelo de Sensores	Daniel	Lectura y parseo del archivo JSON de logs.
✓	Diseñar Dashboard Principal	Daniel	Estructura visual en vista_dashboard_sensores.py (mapa y paneles).
✓	Implementar Panel de Log	Daniel	ListView lateral que muestra las últimas 10 lecturas.
✓	Conectar Controlador Sensores	Daniel	Botón "Refrescar" conectado a la lectura de datos nuevos.

● HU-08: Sistema de Videovigilancia

Objetivo: Integrar el control visual de la cámara de seguridad.

Estado	Tarea	Responsable	Entregable
✓	Crear Vista de Cámara	Fernando	Componente visual para el feed

			(simulado) de la cámara.
✓	Conectar Controlador Cámara	Fernando	Lógica de encendido/apagado y permisos de usuario.



Métricas del Sprint

- **Total de Tareas Planificadas:** 12
- **Total de Tareas Completadas:** 12
- **Tasa de éxito:** 100%
- **Deuda Técnica:** Baja. Se resolvió un bug crítico con el ListView antes del cierre.



Evidencias del Incremento (Demo)

El sistema ahora cuenta con las siguientes capacidades funcionales:

1. **Interactividad Real:** Al pulsar "Refrescar" o realizar una acción, la interfaz responde inmediatamente sin necesidad de reiniciar la app.
2. **Gestión de Presos:**
 - Se pueden añadir y editar presos con validación de nombre.
 - Se pueden buscar presos por nombre o ID en tiempo real.
 - Se pueden eliminar presos de la base de datos JSON.
3. **Visualización IoT:** El panel derecho muestra un log vivo de lo que está ocurriendo en la comisaría (simulado por el script de fondo).
4. **Mapa Interactivo:** Las puertas cambian de color (Rojo/Verde) según su estado en el JSON.
 - `do .update()`, asegurando que el control ya estuviera montado en la interfaz.



Aprobación

Product Owner: Álvaro López

Scrum Master: Nicolás Ramírez

Fecha de Cierre: 23 de Noviembre de 2025