

Proyecto Kigo - Self Check-In AI

Sistema inteligente de auto registro multimodal para comunidades cerradas

Tipo de documento	Especificación de Requerimientos de Software (ERS) para FEPRO 2026
Empresa proponente	Kigo
Objetivo de entrega	Definir un MVP funcional y demostrable en un ciclo de 14 a 16 semanas

Resumen ejecutivo

Kigo Self Check-In AI es un sistema de auto registro sobre dispositivo Android en modo kiosko para visitantes y proveedores, aplicable a comunidades residenciales y corporativas. El sistema permitirá al usuario registrarse mediante interfaz táctil o asistencia por voz, capturar fotografía personal e identificación, validar reglas básicas de acceso y generar una solicitud o autorización para ingreso. El uso de IA deberá aportar valor real al flujo, por ejemplo en guía conversacional por voz, extracción de datos, validación de calidad de evidencia y apoyo contextual durante el registro. El resultado esperado al cierre de FEPRO 2026 es un prototipo funcional, integrable con APIs de Kigo y listo para una demo industrial.

1. Definición del proyecto

1.1 Título del reto

Kigo Self Check-In AI: sistema inteligente de auto registro multimodal para comunidades cerradas.

1.2 Descripción del problema

En comunidades residenciales y corporativas, el proceso de registro de visitantes y proveedores suele depender de personal de recepción o caseta, lo que provoca tiempos de espera, captura manual de datos, inconsistencias en el registro y poca estandarización entre sitios.

Además, muchos accesos ocurren en horas pico o en momentos de baja disponibilidad operativa, lo que incrementa filas, errores de captura y fricción para el visitante. En varios casos, la evidencia del registro queda incompleta o dispersa, dificultando la trazabilidad operativa y la mejora de la experiencia de acceso.

Kigo busca explorar un sistema de auto registro más completo y avanzado, capaz de operar en un dispositivo Android tipo kiosko, combinando experiencia táctil y experiencia guiada por voz con apoyo de IA, así como captura de fotografía e identificación oficial. La solución deberá priorizar una experiencia de usuario sobresaliente, intuitiva y de muy baja fricción para personas no entrenadas.

1.3 Objetivo general

Desarrollar un MVP funcional de kiosko de auto registro para Android que permita registrar visitantes y proveedores mediante interfaz táctil o asistencia por voz, capturar evidencia básica, aplicar reglas de negocio y generar una solicitud o autorización de acceso integrable con el ecosistema Kigo, con una experiencia de uso simple, intuitiva y demostrable en campo.



1.4 Alcance del MVP

- Aplicación Android en modo kiosk para tablet o dispositivo equivalente proporcionado por Kigo.
- Flujo principal de auto registro para visitantes y proveedores en contexto residencial o corporativo.
- Selección del modo de interacción: por botones o por voz.
- Captura de datos mínimos del visitante y del anfitrión, residente, colaborador o destino de visita.
- Toma de fotografía del visitante y captura de imagen del documento de identidad.
- Validaciones básicas de negocio con apoyo de APIs o reglas configurables.
- Generación de resultado de registro: solicitud enviada, acceso autorizado o registro pendiente de revisión.
- Panel básico o vista de consulta para revisar registros realizados y su estatus.

1.5 Oportunidades de uso de IA

- Asistente conversacional por voz para guiar el flujo paso a paso y resolver dudas frecuentes del visitante.
- OCR o extracción inteligente de datos desde la identificación para reducir captura manual y errores.
- Validación de calidad de evidencia para detectar fotos borrosas, documentos incompletos o capturas inutilizables y pedir reintento.
- Soporte multilinguaje o traducción contextual para visitantes que prefieran otro idioma.
- Analítica de patrones de uso del kiosk para identificar abandonos, puntos de fricción y oportunidades de mejora operativa.



2. Requerimientos funcionales priorizados (MoSCoW)

Prioridad	Módulo	Requerimiento
Must	Inicio y flujo	Permitir iniciar el proceso de auto registro desde una pantalla principal clara y operable en modo kiosko.
Must	Modalidad de interacción	Permitir que el usuario elija entre captura por interfaz táctil o asistencia guiada por voz.
Must	Captura de datos	Registrar nombre, motivo de visita, anfitrión o destino, y datos mínimos definidos por la operación.
Must	Evidencia visual	Tomar fotografía del visitante y capturar imagen del documento de identidad.
Must	Validación	Consultar reglas básicas de acceso mediante APIs, listas o configuraciones entregadas por Kigo.
Must	Resultado	Generar un resultado visible del proceso: aprobado, enviado a revisión o rechazado con mensaje claro.
Must	Bitácora	Guardar los registros generados con fecha, hora, evidencia y estatus.
Should	Asistencia IA	Incluir una experiencia guiada por voz con IA para acompañar el registro de principio a fin.
Should	Extracción de datos	Extraer campos básicos del documento capturado para reducir captura manual.
Should	Notificaciones	Notificar al residente, anfitrión o responsable cuando se genere una nueva solicitud.
Should	Validación de calidad	Detectar imágenes borrosas, incompletas o de baja calidad y solicitar una nueva captura cuando sea necesario.
Should	Configuración	Permitir parametrizar el flujo para contexto residencial o corporativo sin reconstruir la app.
Could	Asistente contextual	Ofrecer ayuda contextual con IA para explicar campos, aclarar instrucciones o detectar atascos en el flujo.
Could	QR o ticket	Generar un comprobante visual, QR o ticket digital para continuidad del flujo de acceso.
Could	Analítica	Mostrar métricas básicas de uso del kiosko, tiempos de registro y causas de abandono.
Won't	Hardware industrial	No incluye fabricación de kiosko físico, torniquete, impresora ni barrera en el alcance del MVP.
Won't	Verificación biométrica avanzada	No incluye validación biométrica forense, equivalencia legal de identidad ni certificación gubernamental.



3. Requerimientos no funcionales

Usabilidad: La solución debe ofrecer una experiencia de usuario sobresaliente, súper fácil e intuitiva, legible y operable por usuarios no entrenados en menos de 3 minutos para el flujo principal.

Rendimiento: La navegación entre pantallas debe ser fluida y las interacciones principales deben responder en tiempos razonables para demo y pruebas de campo.

Conectividad: Debe contemplarse comportamiento controlado ante fallas de red, mostrando mensajes claros y evitando pérdida silenciosa de información.

Seguridad y privacidad: El manejo de datos personales e imágenes debe alinearse con prácticas de protección de datos definidas por Kigo y con ambientes de prueba autorizados.

Mantenibilidad: El código debe estar estructurado por módulos y documentado para permitir continuidad técnica posterior.

Escalabilidad funcional: El diseño debe facilitar la futura incorporación de nuevos flujos, validaciones y tipos de visitantes sin rehacer el núcleo de la aplicación.

IA aplicada: Las funciones de IA deben ser explicables a nivel de producto, activarse solo cuando aporten valor real al flujo y contar con rutas de respaldo manual cuando corresponda.

4. Restricciones y entorno técnico

Plataforma objetivo	Android para tablet o dispositivo equivalente en modo kiosko.
Tipo de solución	Prototipo funcional de software. El hardware físico será provisto por Kigo o simulado en ambiente de pruebas.
Backend preferido	De preferencia Go. También se acepta Node.js o Python para servicios de apoyo, integración o lógica complementaria del MVP.
UX esperada	La propuesta debe priorizar claridad extrema, mínima fricción, accesibilidad básica y una experiencia apta para auto servicio sin capacitación previa.
Integración	Consumo de APIs o servicios de prueba proporcionados por Kigo para registro, validación o consulta.
Frontend preferido	De preferencia Flutter, Kotlin también es aceptado. Se aceptan propuestas equivalentes compatibles con Android si la justificación técnica es clara.
Datos de prueba	Se utilizarán datos, imágenes y ambientes autorizados por Kigo, evitando uso indebido de información sensible en producción.
Idioma sugerido	Español como idioma base del MVP, con posibilidad de contemplar inglés como mejora.



5. Entregables técnicos esperados

- Código fuente en repositorio Git con instrucciones de instalación y ejecución.
- MVP funcional ejecutable en Android o demostrable sobre el dispositivo asignado.
- Documentación técnica breve: arquitectura, flujos principales, dependencias y decisiones relevantes.
- Manual de uso o guía rápida para operación del prototipo.
- Presentación final y demo del caso de uso seleccionado.

6. Recursos proporcionados por Kigo

- Tablet Android o equipo equivalente para ejecución del prototipo.
- Acceso a APIs, endpoints o sandbox necesarios para el flujo de registro y consulta.
- Reglas de negocio básicas, casos de uso de referencia y criterios de operación.
- Acompañamiento técnico y de producto durante el programa, con seguimiento semanal.
- Datos y ejemplos de prueba autorizados para construir y validar el MVP.
- Canal de Slack para comunicación asíncrona y seguimiento semanal.

7. Cronograma sugerido de desarrollo

Fase	Periodo	Actividad
I	Semanas 1-2	Kick-off, entendimiento del problema, definición de alcance, arquitectura y backlog inicial.
II	Semanas 3-5	Diseño UX del kiosko, prototipo navegable y definición de integración.
III	Semanas 6-10	Desarrollo del núcleo funcional: registro, captura de evidencia y bitácora.
IV	Semanas 11-13	Integración de asistencia por voz, funciones de IA priorizadas, mejoras de usabilidad y validaciones adicionales.
V	Semanas 14-16	Pruebas, corrección de bugs, documentación, pitch y demo final.

8. Calendario de mentoría y seguimiento

- Semana 1: reunión de kick-off con presentación oficial del reto, contexto de negocio y resolución de dudas iniciales.
- Seguimiento semanal: sesiones breves, presenciales o remotas, para revisión de avances, decisiones de arquitectura y desbloques.
- Check-point de pre-entrega: revisión integral del MVP, feedback de demo y ajustes finales.
- Canal asíncrono: uso de Slack para dudas rápidas, acuerdos y seguimiento entre sesiones.



9. Criterios de evaluación desde la perspectiva industrial

Criterio	Qué se evaluará
Funcionalidad	Qué tanto el MVP resuelve el problema descrito y ejecuta el flujo principal de forma coherente.
Calidad del código	Claridad, modularidad, mantenibilidad y buenas prácticas de desarrollo.
Innovación	Uso valioso de voz, IA, diseño multimodal y experiencia de auto servicio, evitando complejidad innecesaria.
Presentación	Capacidad del equipo para explicar valor técnico y de negocio con una demo sólida.
Impacto	Viabilidad de implementación y potencial de mejora en operación y experiencia de usuario.

10. Consideraciones finales y exclusiones

Aunque la visión del producto contempla aplicabilidad en residencial y corporativo, el equipo deberá proponer un flujo principal suficientemente claro y demostrable dentro del tiempo del programa.

El proyecto debe priorizar experiencia de uso, robustez del flujo, calidad de evidencia operativa y uso de IA con valor real por encima de integraciones complejas no esenciales para el MVP.

Quedan fuera del alcance de esta ERS la fabricación del kiosko físico, integraciones industriales definitivas, certificaciones regulatorias y cualquier tratamiento de datos en producción no autorizado por Kigo.

Nombre recomendado para presentación pública: Kigo Self Check-In AI

Alternativa formal: Sistema inteligente de auto registro multimodal para comunidades cerradas.

