

# Reporte de Pruebas de Aceptación

Sistema de Control de Asistencia GeoFace

Fecha de verificación: 07/11/2025

Requisitos Funcionales

14 / 14

Cumple

Requisitos No Funcionales

5 / 5

Cumple

## Resumen Ejecutivo

Se realizaron pruebas de aceptación sobre 14 requisitos funcionales y 5 requisitos no funcionales del sistema GeoFace. El resultado global indica que todas las validaciones planificadas fueron aprobadas, asegurando el cumplimiento de las capacidades clave solicitadas por el cliente.

## Requisitos Funcionales

RF-001 Autenticar usuario en el sistema

Escenario verificado: Administrador inicia sesión con credenciales válidas y accede al dashboard correspondiente.

Resultado esperado: La sesión se establece en menos de 2 segundos y el estado global refleja su rol.

Evidencia: Video Login\_Admin.mp4 / Logs auth\_controller (2025-11-07)

Responsable: QA - JOrge Briceño

Estado: Aprobado

RF-002 Gestionar sedes con perímetros

Escenario verificado: Administrador crea una nueva sede con coordenadas reales y la visualiza en la lista general.

Resultado esperado: La sede queda disponible para asignaciones y se persisten los campos obligatorios en Firestore.

Evidencia: Captura RF002\_sede\_creada.png / Registro en colección sedes

Responsable: QA - Brayar Lopez

Estado: Aprobado

RF-003

**Gestionar empleados**

Escenario verificado: Se registra un empleado, se asigna a una sede y se valida la unicidad de DNI y correo.

Resultado esperado: El empleado aparece activo en la tabla, sin duplicados y con datos consistentes.

Evidencia: Checklist RF003\_empleado.xlsx / Captura formulario con validaciones

Responsable: QA - JOrge Briceño

**Estado: Aprobado**

RF-004

**Registrar datos biométricos**

Escenario verificado: El sistema almacena tres capturas faciales y marca al empleado como listo para reconocimiento.

Resultado esperado: Se generan URLs válidas en Storage y se actualiza el flag hayDatosBiometricos.

Evidencia: Carpeta evidencias/biometrico/emp\_001/, log Storage upload

Responsable: QA - Brayar Lopez

**Estado: Aprobado**

RF-005

**Configurar URLs de API**

Escenario verificado: Se registra una URL base HTTPS y el sistema deriva los endpoints de identificación y sincronización.

Resultado esperado: Los endpoints son persistidos, validados y la sincronización queda habilitada.

Evidencia: Captura RF005\_config.png / Documento app\_config/settings

Responsable: QA - Jorge Briceño

**Estado: Aprobado**

RF-006

**Marcar asistencia con reconocimiento facial**

Escenario verificado: Empleado con datos biométricos válidos marca entrada dentro de la geocerca y luego registra salida.

Resultado esperado: Se crea el registro de asistencia con coordenadas y evidencia fotográfica.

Evidencia: Video RF006\_asistencia.mov / Documento firestore/asistencias

Responsable: QA - Brayar Lopez

**Estado: Aprobado**

RF-007

**Visualizar detalle de asistencia diaria**

Escenario verificado: Empleado revisa su historial del día y puede distinguir asistencias completas e incompletas.

Resultado esperado: El detalle presenta horas, geolocalización y estado de cada marca.

Evidencia: Capturas Aplicación móvil - pantalla historial

Responsable: QA - JOrge Briceño

**Estado: Aprobado**

RF-008

**Dashboard de monitoreo (Empleado)**

Escenario verificado: Empleado autenticado visualiza su estado actual (entrada/salida) y atajos a acciones clave.

Resultado esperado: La interfaz cambia al marcar entrada/salida y mantiene consistencia tras refrescar.

Evidencia: GIF RF008\_dashboard.gif / Logica Provider refresh()

Responsable: QA - Brayar Lopez

**Estado: Aprobado**

RF-009

**Generar reportes detallados**

Escenario verificado: Administrador filtra asistencias por sede y mes y genera estadísticas de tardanzas y ausencias.

Resultado esperado: El resumen muestra totales consistentes y permite exportar a PDF.

Evidencia: Reporte generado: reportes/reporte\_agosto.pdf

Responsable: QA - Jorge Briceño

**Estado: Aprobado**

RF-010

**Cambiar contraseña de usuario**

Escenario verificado: Admin autenticado ingresa contraseña actual y nueva, recibiendo confirmación y reautenticación.

Resultado esperado: La contraseña se actualiza en Firebase Auth y se notifica al usuario.

Evidencia: Registro firebaseAuth.changePassword / SnackBar confirmación

Responsable: QA - Brayar Lopez

**Estado: Aprobado**

RF-011

**Exportar reportes a PDF**

Escenario verificado: Administrador exporta un reporte mensual y el archivo mantiene formato corporativo.

Resultado esperado: Se genera PDF descargable con todas las métricas y logos de la empresa.

Evidencia: Archivo pdf/export\_2025-11.pdf

Responsable: QA - Jorge Briceño

**Estado: Aprobado**

RF-012

**Gestionar usuarios administradores**

Escenario verificado: Superadmin crea, edita y desactiva administradores asegurando que no accedan cuando están inactivos.

Resultado esperado: El listado refleja cambios en tiempo real y se respetan los roles.

Evidencia: Captura RF012\_admins.png / Registro usuarios colección

Responsable: QA - Brayar Lopez

**Estado: Aprobado**

RF-013

**Asignar credenciales a empleados**

Escenario verificado: Administrador genera usuario @geoface.com para un empleado y fuerza cambio de contraseña en primer login.

Resultado esperado: El empleado recibe credenciales y se revierte la sesión del admin por seguridad.

Evidencia: Log creación FirebaseAuth / Correo generado automáticamente

Responsable: QA - Jorge Briceño

**Estado: Aprobado**

RF-014

**Sincronizar datos faciales con API externa**

Escenario verificado: Administrador ejecuta la sincronización y recibe confirmación del endpoint remoto.

Resultado esperado: La API responde 200 OK y se muestra notificación de éxito.

Evidencia: Captura Postman sync-database.png / Log consola HTTP 200

Responsable: QA - Brayar Lopez

**Estado: Aprobado**

## Requisitos No Funcionales

Código	Nombre	Criterio de aceptación	Medición/Evidencia	Estado
RNF-001	Rendimiento	El sistema debe responder en menos de 2 segundos para operaciones comunes (login, consulta, marcación). ☑ Mediciones con cronómetro y logs automáticos durante 10 ejecuciones consecutivas en ambiente QA.	Promedio login 1.4 s, registro asistencia 1.8 s, carga dashboard 1.2 s (Logs 2025-11-07).	Aprobado
RNF-002	Seguridad	Comunicación cifrada entre clientes y backend, autenticación con Firebase y roles segregados. ☑ Validación de certificados HTTPS, revisión de reglas Firestore y pruebas de cuentas inactivas.	Todas las URLs usan https://, reglas Firestore restringen acceso y usuario inactivo es rechazado.	Aprobado
RNF-003	Disponibilidad	El servicio debe operar de 8:00 a 18:00 con mantenimiento planificado fuera del horario. ☑ Monitoreo de uptime con Firebase Status y registro manual durante 5 días hábiles.	Disponibilidad 99.2% en semana 44 (logs cloud functions) - sin caídas en horario laboral.	Aprobado
RNF-004	Portabilidad	Compatibilidad con Android 8.0+ y diseño adaptable a pantallas; preparada para futuro soporte iOS. ☑ Pruebas en dispositivos/emuladores 5.5" y 6.7" Android; verificación de build iOS en Flutter.	APK probado en Pixel 3a (Android 12) y Samsung A21 (Android 10); flutter build ios --no-tree-shake-icons exitoso.	Aprobado
RNF-005	Mantenibilidad	Código modular con buenas prácticas Flutter, documentación y separación de responsabilidades. ☑ Revisión de arquitectura MVC/Provider, comentarios en controladores y cumplimiento de lint.	Ejecución de flutter analyze sin errores; documentación en cabeceras y tests unitarios por módulo.	Aprobado