

- Plan de Pruebas Detallado
 - Sistema de Gestión de Flota PepsiCo Chile
 - 1. Información del Documento
 - 2. Introducción
 - 2.1 Propósito
 - 2.2 Alcance
 - 2.3 Objetivos de Calidad
 - 3. Estrategia de Pruebas
 - 3.1 Enfoque General
 - 3.2 Metodología
 - 3.3 Criterios de Entrada
 - 3.4 Criterios de Salida
 - 3.5 Criterios de Suspensión
 - 4. Tipos de Pruebas
 - 4.1 Pruebas Unitarias
 - 4.2 Pruebas de Integración
 - 4.3 Pruebas Funcionales
 - Módulo de Autenticación (4 casos)
 - Módulo de Gestión de Ingresos (7 casos)
 - Módulo de Gestión de Vehículos (4 casos)
 - Módulo de Órdenes de Trabajo (10 casos)
 - Módulo de Inventario (6 casos)
 - Módulo de Reportes (7 casos)
 - Módulo de Notificaciones (4 casos)
 - Módulo de Administración (5 casos)
 - 4.4 Pruebas de Rendimiento
 - 4.5 Pruebas de Seguridad
 - 4.6 Pruebas de Usabilidad
 - 4.7 Pruebas de Compatibilidad
 - 5. Pruebas de Aceptación de Usuario (UAT)
 - 5.1 Objetivo
 - 5.2 Participantes
 - 5.3 Escenarios de Aceptación
 - 5.4 Criterios de Aceptación Global
 - 6. Gestión de Defectos
 - 6.1 Clasificación de Severidad
 - 6.2 Prioridad

- 6.3 Estados de Defecto
- 6.4 Herramienta de Gestión
- 7. Ambientes de Prueba
 - 7.1 Configuración de Ambientes
 - 7.2 Datos de Prueba
 - 7.3 Requisitos de Infraestructura
- 8. Recursos y Responsabilidades
 - 8.1 Equipo de Pruebas
 - 8.2 Capacitación Requerida
- 9. Cronograma de Pruebas
 - 9.1 Fases del Proyecto
 - 9.2 Plan Detallado por Sprint
- 10. Automatización de Pruebas
 - 10.1 Estrategia de Automatización
 - 10.2 Herramientas de Automatización
 - 10.3 Casos a Automatizar
 - 10.4 Integración Continua
- 11. Métricas y Reportes
 - 11.1 Métricas Clave
 - 11.2 Reportes
 - 11.3 Dashboard de Pruebas
- 12. Riesgos y Mitigación
 - 12.1 Riesgos Identificados
 - 12.2 Contingencias
- 13. Criterios de Calidad y Aceptación
 - 13.1 Definición de "Hecho" (Definition of Done)
 - 13.2 Criterios de Aceptación Global
- 14. Entregables
 - 14.1 Documentos
 - 14.2 Artefactos
- 15. Glosario
- 16. Referencias
- 17. Aprobaciones
- 18. Control de Versiones

Plan de Pruebas Detallado

Sistema de Gestión de Flota PepsiCo Chile

1. Información del Documento

Campo	Valor
Proyecto	Sistema de Gestión de Ingreso de Vehículos
Cliente	PepsiCo Chile
Fecha de Creación	15/10/2025
Versión	1.0
Estado	Aprobado
Responsables	Joaquín Marín, Benjamin Vilches
Institución	Duoc UC

2. Introducción

2.1 Propósito

Este documento define la estrategia, alcance, recursos, cronograma y casos de prueba para el Sistema de Gestión de Flota PepsiCo Chile. El objetivo es garantizar que el sistema cumpla con todos los requerimientos funcionales y no funcionales especificados.

2.2 Alcance

El plan de pruebas cubre:

- 10 Requerimientos Funcionales (RF-01 a RF-10)
- 7 Requerimientos No Funcionales (RNF-01 a RNF-07)

- 43 Casos de Uso
- 8 Módulos Funcionales
- 20 Tablas de Base de Datos
- 30+ Endpoints de API REST

2.3 Objetivos de Calidad

- **Funcionalidad:** 100% de RF implementados correctamente
 - **Cobertura de Código:** Mínimo 80% en Backend, 70% en Frontend
 - **Rendimiento:** Tiempo de respuesta < 3 segundos
 - **Disponibilidad:** 99.9% uptime
 - **Seguridad:** Cero vulnerabilidades críticas
 - **Usabilidad:** System Usability Scale (SUS) > 80
-

3. Estrategia de Pruebas

3.1 Enfoque General

Se aplicará un enfoque de pruebas **multi-nivel** que incluye:

1. **Pruebas Unitarias:** Validación de componentes individuales
2. **Pruebas de Integración:** Validación de interacción entre componentes
3. **Pruebas Funcionales:** Validación de casos de uso completos
4. **Pruebas de Sistema:** Validación del sistema completo
5. **Pruebas de Aceptación:** Validación con usuarios finales

3.2 Metodología

- **Desarrollo Dirigido por Pruebas (TDD):** Para componentes críticos
- **Integración Continua (CI/CD):** Ejecución automática de pruebas
- **Pruebas de Regresión:** Automatizadas en cada despliegue
- **Pruebas Exploratorias:** Manuales para descubrir casos edge

3.3 Criterios de Entrada

Para iniciar las pruebas:

- Código fuente disponible en repositorio
- Ambiente de pruebas configurado
- Base de datos con datos de prueba cargados
- Documentación técnica completa
- Casos de prueba revisados y aprobados

3.4 Criterios de Salida

Para considerar las pruebas completas:

- 100% de casos de prueba críticos ejecutados
- 95% de casos de prueba de prioridad alta ejecutados
- 90% de casos de prueba de prioridad media ejecutados
- Tasa de éxito de pruebas > 95%
- Cero defectos críticos o de bloqueo abiertos
- Máximo 5 defectos menores abiertos

3.5 Criterios de Suspensión

Las pruebas se suspenderán si:

- Más del 30% de casos de prueba fallan
- Se encuentran 3+ defectos críticos sin resolver
- El ambiente de pruebas está inestable por > 24 horas
- Cambios mayores en requerimientos requieren re-planificación

4. Tipos de Pruebas

4.1 Pruebas Unitarias

Objetivo: Validar el funcionamiento correcto de funciones, métodos y componentes individuales.

Alcance:

- Servicios de Backend (100% de servicios críticos)
- Controladores de API
- Componentes de React
- Funciones utilitarias
- Validaciones de esquemas

Herramientas:

- Backend: Jest + Supertest
- Frontend: Vitest + React Testing Library

Responsable: Desarrolladores

Cobertura Objetivo: 80% Backend, 70% Frontend

Ejemplos de Casos:

ID	Componente	Caso de Prueba	Resultado Esperado
UT-001	AuthService.login()	Login con credenciales válidas	Retorna token JWT válido
UT-002	AuthService.login()	Login con credenciales inválidas	Retorna error 401
UT-003	VehicleService.create()	Crear vehículo con datos válidos	Vehículo creado en BD
UT-004	VehicleService.create()	Crear vehículo con patente duplicada	Retorna error 409
UT-005	WorkOrderService.assign()	Asignar mecánico disponible	OT actualizada con mecánico
UT-006	SparePartService.requestPart()	Solicitar repuesto con stock	Stock decrementado
UT-007	SparePartService.requestPart()	Solicitar repuesto sin stock	Error de stock insuficiente
UT-008	validateRut()	RUT válido chileno	Retorna true

ID	Componente	Caso de Prueba	Resultado Esperado
UT-009	validateRut()	RUT inválido	Retorna false
UT-010	calculateWorkDuration()	Calcular duración sin pausas	Duración correcta

4.2 Pruebas de Integración

Objetivo: Validar la interacción correcta entre componentes, capas y sistemas externos.

Alcance:

- API REST con Base de Datos
- Frontend con Backend
- Autenticación y Autorización (JWT + RBAC)
- Prisma ORM con PostgreSQL
- Almacenamiento de archivos

Herramientas:

- Postman/Newman para APIs
- Jest para integración Backend
- Cypress para integración Frontend-Backend

Responsable: Desarrolladores y QA

Ejemplos de Casos:

ID	Integración	Caso de Prueba	Resultado Esperado
IT-001	Auth + DB	Login completo	Token generado y sesión creada
IT-002	API + RBAC	Acceso con permisos insuficientes	Error 403 Forbidden
IT-003	Frontend + API	Crear OT desde UI	OT creada y visible en dashboard

ID	Integración	Caso de Prueba	Resultado Esperado
IT-004	OT + Inventory	Solicitar repuesto desde OT	Stock actualizado y relación creada
IT-005	Upload + Storage	Subir foto de vehículo	Archivo guardado y URL retornada
IT-006	Notifications	Asignar OT genera notificación	Notificación creada para mecánico
IT-007	Reports + DB	Generar reporte de flota	Datos correctos desde múltiples tablas
IT-008	Vehicle Entry + Photos	Registrar ingreso con fotos	Ingreso y fotos almacenadas
IT-009	Work Order + Status	Cambiar estado de OT	Historial de estados actualizado
IT-010	Audit + All Actions	Acción crítica ejecutada	Log de auditoría creado

4.3 Pruebas Funcionales

Objetivo: Validar que los casos de uso se ejecuten correctamente de extremo a extremo.

Alcance: 43 Casos de Uso del sistema

Herramientas:

- Cypress para pruebas E2E
- Playwright (alternativa)
- Checklist manual para flujos críticos

Responsable: QA Team

Casos de Prueba por Módulo:

Módulo de Autenticación (4 casos)

ID	Caso de Uso	Escenario de Prueba	Prioridad
FT-001	CU-001: Iniciar Sesión	Usuario ingresa credenciales válidas	Crítica
FT-002	CU-001: Iniciar Sesión	Usuario ingresa credenciales inválidas	Alta
FT-003	CU-002: Cerrar Sesión	Usuario cierra sesión activa	Alta
FT-004	CU-003: Recuperar Contraseña	Usuario solicita reseteo de contraseña	Media

Módulo de Gestión de Ingresos (7 casos)

ID	Caso de Uso	Escenario de Prueba	Prioridad
FT-010	CU-010: Registrar Ingreso	Guardia registra ingreso completo	Crítica
FT-011	CU-010: Registrar Ingreso	Intento de ingreso con patente inválida	Alta
FT-012	CU-011: Tomar Fotografías	Subir 4 fotos del vehículo	Crítica
FT-013	CU-011: Tomar Fotografías	Intento de subir archivo no imagen	Media
FT-014	CU-012: Registrar Salida	Guardia registra salida completa	Crítica
FT-015	CU-012: Registrar Salida	Intento de salida con OT pendiente	Alta
FT-016	CU-027: Agregar Documentos	Subir PDF de documentación	Media

Módulo de Gestión de Vehículos (4 casos)

ID	Caso de Uso	Escenario de Prueba	Prioridad
FT-020	CU-013: Buscar Vehículo	Búsqueda por patente exacta	Alta

ID	Caso de Uso	Escenario de Prueba	Prioridad
FT-021	CU-013: Buscar Vehículo	Búsqueda con filtros múltiples	Media
FT-022	CU-014: Registrar Vehículo	Crear vehículo nuevo	Alta
FT-023	CU-016: Ver Historial	Consultar historial completo	Media

Módulo de Órdenes de Trabajo (10 casos)

ID	Caso de Uso	Escenario de Prueba	Prioridad
FT-030	CU-020: Crear OT	Recepcionista crea OT completa	Crítica
FT-031	CU-020: Crear OT	Crear OT sin descripción (validación)	Alta
FT-032	CU-021: Asignar Mecánico	Jefe asigna mecánico disponible	Crítica
FT-033	CU-021: Asignar Mecánico	Asignar a mecánico con carga completa	Media
FT-034	CU-022: Actualizar Estado	Mecánico cambia estado a "en progreso"	Crítica
FT-035	CU-022: Actualizar Estado	Secuencia completa de estados	Alta
FT-036	CU-023: Solicitar Repuesto	Solicitud con stock disponible	Alta
FT-037	CU-023: Solicitar Repuesto	Solicitud con stock insuficiente	Alta
FT-038	CU-025: Registrar Pausa	Mecánico pausa OT con motivo	Media
FT-039	CU-026: Finalizar Pausa	Mecánico resume OT pausada	Media

Módulo de Inventario (6 casos)

ID	Caso de Uso	Escenario de Prueba	Prioridad
FT-040	CU-024: Entregar Repuesto	Asistente entrega repuesto solicitado	Alta
FT-041	CU-060: Actualizar Inventario	Ajuste manual de stock	Media
FT-042	CU-061: Registrar Ingreso	Ingreso de nuevos repuestos	Media
FT-043	CU-062: Ajustar Stock	Corrección de inventario	Media
FT-044	CU-063: Ver Movimientos	Consultar historial de movimientos	Baja
FT-045	Alertas de Stock	Sistema genera alerta por stock mínimo	Alta

Módulo de Reportes (7 casos)

ID	Caso de Uso	Escenario de Prueba	Prioridad
FT-050	CU-029: Ver Dashboard	Supervisora accede a dashboard	Alta
FT-051	CU-030: Reporte de Flota	Generar reporte con filtros	Alta
FT-052	CU-031: Reporte de Desempeño	Generar por mecánico y período	Media
FT-053	CU-032: Reporte de Inventario	Reporte de stock actual	Media
FT-054	CU-033: Reporte de Costos	Análisis de costos por vehículo	Media
FT-055	CU-034: Exportar PDF	Exportar cualquier reporte a PDF	Media
FT-056	CU-035: Exportar Excel	Exportar reporte a Excel	Media

Módulo de Notificaciones (4 casos)

ID	Caso de Uso	Escenario de Prueba	Prioridad
FT-060	CU-040: Recibir Notificación	Usuario recibe notificación en tiempo real	Alta
FT-061	CU-041: Marcar como Leída	Marcar notificación individual	Baja
FT-062	CU-042: Marcar Todas Leídas	Marcar todas de un usuario	Baja
FT-063	CU-043: Eliminar Notificación	Eliminar notificación específica	Baja

Módulo de Administración (5 casos)

ID	Caso de Uso	Escenario de Prueba	Prioridad
FT-070	CU-050: Gestionar Usuarios	Admin lista todos los usuarios	Alta
FT-071	CU-051: Crear Usuario	Admin crea usuario nuevo	Alta
FT-072	CU-052: Editar Usuario	Admin modifica datos de usuario	Alta
FT-073	CU-053: Desactivar Usuario	Admin desactiva usuario	Alta
FT-074	CU-054: Gestionar Roles	Admin asigna permisos a rol	Alta

4.4 Pruebas de Rendimiento

Objetivo: Validar que el sistema cumple con los requerimientos de rendimiento (RNF-02).

Alcance:

- Tiempo de respuesta de endpoints críticos
- Capacidad de carga concurrente
- Tiempo de carga de páginas

- Consultas de base de datos

Herramientas:

- Apache JMeter
- k6 (Grafana)
- Lighthouse (performance web)

Responsable: QA + DevOps

Escenarios de Carga:

ID	Escenario	Usuarios	Duración	Objetivo
PT-001	Carga normal	20 concurrentes	30 min	< 3s respuesta
PT-002	Carga pico	50 concurrentes	15 min	< 5s respuesta
PT-003	Estrés	100 concurrentes	10 min	Sistema estable
PT-004	Resistencia	10 concurrentes	4 horas	Sin degradación

Endpoints Críticos a Medir:

Endpoint	Método	Objetivo Tiempo	Carga
POST /api/auth/login	POST	< 500ms	100 req/min
GET /api/vehicles?search=	GET	< 2s	50 req/min
POST /api/work-orders	POST	< 1s	30 req/min
GET /api/work-orders/:id	GET	< 500ms	100 req/min
PATCH /api/work-orders/:id/status	PATCH	< 800ms	80 req/min
GET /api/dashboard/stats	GET	< 3s	20 req/min
POST /api/uploads/photo	POST	< 5s	10 req/min
GET /api/reports/fleet	GET	< 3s	10 req/min

4.5 Pruebas de Seguridad

Objetivo: Validar que el sistema cumple con los requerimientos de seguridad (RNF-04).

Alcance:

- Autenticación y autorización
- Inyección SQL
- Cross-Site Scripting (XSS)
- Cross-Site Request Forgery (CSRF)
- Manejo de sesiones
- Encriptación de contraseñas
- Control de acceso basado en roles (RBAC)

Herramientas:

- OWASP ZAP
- Burp Suite (Community)
- npm audit / Snyk
- SonarQube

Responsable: Security Team + QA

Casos de Prueba de Seguridad:

ID	Categoría	Caso de Prueba	Resultado Esperado
ST-001	Autenticación	Acceso sin token	Error 401
ST-002	Autenticación	Token expirado	Error 401
ST-003	Autenticación	Token malformado	Error 401
ST-004	Autorización	Acceso con rol insuficiente	Error 403
ST-005	Autorización	Modificar recurso de otro usuario	Error 403
ST-006	SQL Injection	Enviar payload SQL en búsqueda	Sin ejecución SQL
ST-007	XSS	Enviar script en campo de texto	Script escapado

ID	Categoría	Caso de Prueba	Resultado Esperado
ST-008	CSRF	Request sin token CSRF	Rechazado
ST-009	Passwords	Verificar hash bcrypt	Nunca texto plano
ST-010	Session	Logout invalida token	Token inválido después
ST-011	Rate Limiting	100 requests en 1 minuto	Bloqueado temporalmente
ST-012	File Upload	Subir archivo .exe	Rechazado
ST-013	Audit	Acción administrativa	Log creado
ST-014	HTTPS	Acceso HTTP	Redirect a HTTPS
ST-015	Headers	Verificar security headers	Headers presentes

4.6 Pruebas de Usabilidad

Objetivo: Validar que el sistema es fácil de usar y cumple con los requerimientos de UX (RNF-03).

Alcance:

- Navegación intuitiva
- Claridad de formularios
- Mensajes de error comprensibles
- Responsividad en dispositivos móviles
- Accesibilidad (WCAG 2.1 AA)

Herramientas:

- Manual testing con usuarios reales
- System Usability Scale (SUS) questionnaire

- Google Lighthouse (accessibility)
- Hotjar / FullStory (opcional)

Responsable: UX Team + QA

Criterios de Evaluación:

ID	Criterio	Objetivo
UT-001	System Usability Scale (SUS)	Score > 80
UT-002	Tasa de éxito en tareas	> 90%
UT-003	Tiempo promedio por tarea	Dentro de benchmark
UT-004	Errores de usuario	< 5%
UT-005	Satisfacción del usuario	> 4/5

Tareas de Prueba con Usuarios:

ID	Tarea	Actor	Tiempo Esperado
UST-001	Registrar ingreso de vehículo completo	Guardia	< 5 minutos
UST-002	Crear orden de trabajo	Recepcionista	< 3 minutos
UST-003	Asignar mecánico a OT	Jefe Taller	< 1 minuto
UST-004	Actualizar estado de OT desde móvil	Mecánico	< 30 segundos
UST-005	Solicitar repuesto	Mecánico	< 1 minuto
UST-006	Buscar vehículo por patente	Cualquiera	< 30 segundos
UST-007	Generar y exportar reporte	Supervisora	< 2 minutos
UST-008	Consultar dashboard en tablet	Jefe Taller	Inmediato

4.7 Pruebas de Compatibilidad

Objetivo: Validar que el sistema funciona en diferentes navegadores y dispositivos (RNF-06).

Alcance:

- Navegadores: Chrome, Firefox, Safari, Edge
- Dispositivos: Desktop, Tablet, Mobile
- Sistemas Operativos: Windows, macOS, iOS, Android

Herramientas:

- BrowserStack
- Manual testing en dispositivos físicos

Responsable: QA Team

Matriz de Compatibilidad:

Navegador	Versión	Desktop	Tablet	Mobile	Prioridad
Chrome	Últimas 2	✓	✓	✓	Crítica
Firefox	Últimas 2	✓	✓	✓	Alta
Safari	Últimas 2	✓	✓	✓	Alta
Edge	Últimas 2	✓	-	-	Media

Resoluciones a Probar:

Dispositivo	Resolución	Orientación
Desktop HD	1920x1080	Horizontal
Desktop FHD	2560x1440	Horizontal
Laptop	1366x768	Horizontal
Tablet iPad	768x1024	Vertical/Horizontal
Mobile Android	360x640	Vertical
Mobile iPhone	375x667	Vertical

5. Pruebas de Aceptación de Usuario (UAT)

5.1 Objetivo

Validar que el sistema cumple con las expectativas del cliente y usuarios finales antes de la puesta en producción.

5.2 Participantes

- Guardias de Acceso (2 personas)
- Recepcionistas de Taller (2 personas)
- Jefes de Taller (2 personas)
- Mecánicos (3 personas)
- Asistentes de Repuestos (1 persona)
- Supervisora de Flota (1 persona)
- Gerente de Flota (1 persona)

5.3 Escenarios de Aceptación

Escenario 1: Flujo Completo de Ingreso y OT

Paso	Actor	Acción	Criterio de Aceptación
1	Guardia	Registra ingreso de vehículo PAT-123	Código generado, fotos subidas
2	Recepcionista	Crea OT para reparación de motor	OT creada con prioridad alta
3	Jefe Taller	Asigna mecánico Juan Pérez	Mecánico asignado, notificado
4	Mecánico	Actualiza estado a "en progreso"	Estado visible en dashboard
5	Mecánico	Solicita repuesto "filtro aceite"	Solicitud creada

Paso	Actor	Acción	Criterio de Aceptación
6	Asistente	Entrega repuesto	Stock actualizado
7	Mecánico	Completa OT con evidencias	OT marcada como completada
8	Jefe Taller	Valida y aprueba OT	OT aprobada
9	Guardia	Registra salida de vehículo	Salida registrada

Escenario 2: Gestión de Inventario

Paso	Actor	Acción	Criterio de Aceptación
1	Asistente	Consulta stock de repuesto X	Stock actual visible
2	Sistema	Genera alerta de stock mínimo	Notificación enviada
3	Asistente	Registra ingreso de 50 unidades	Stock actualizado correctamente
4	Asistente	Consulta movimientos	Historial visible

Escenario 3: Generación de Reportes

Paso	Actor	Acción	Criterio de Aceptación
1	Supervisora	Accede a dashboard	Métricas actualizadas
2	Supervisora	Genera reporte de flota del mes	Reporte generado con datos correctos
3	Supervisora	Exporta a Excel	Archivo descargado
4	Gerente	Consulta reporte de costos	Datos financieros precisos

5.4 Criterios de Aceptación Global

El sistema será aceptado si:

- ✓ Todos los escenarios críticos se completan sin errores
- ✓ Al menos 95% de funcionalidades funcionan correctamente
- ✓ Usuarios dan aprobación formal
- ✓ No hay defectos críticos o bloqueantes abiertos

- ✓ Documentación de usuario completa y aprobada
- ✓ Capacitación de usuarios completada

6. Gestión de Defectos

6.1 Clasificación de Severidad

Severidad	Descripción	Tiempo de Resolución	Ejemplo
Crítica	Sistema no funciona, bloquea trabajo	24 horas	Login no funciona, BD caída
Alta	Funcionalidad importante no funciona	3 días	No se puede crear OT
Media	Funcionalidad menor no funciona	1 semana	Reporte no exporta a Excel
Baja	Problema cosmético, workaround existe	2 semanas	Error de alineación UI

6.2 Prioridad

Prioridad	Criterio
P1	Afecta producción, sin workaround
P2	Afecta funcionalidad importante
P3	Afecta funcionalidad secundaria
P4	Mejora o cambio menor

6.3 Estados de Defecto

Nuevo → Asignado → En Progreso → Resuelto → Verificado → Cerrado



6.4 Herramienta de Gestión

- **GitHub Issues** con labels:
 - **bug** - defecto
 - **critical** / **high** / **medium** / **low** - severidad
 - **P1** / **P2** / **P3** / **P4** - prioridad
 - **testing** - encontrado en pruebas
 - **production** - encontrado en producción

7. Ambientes de Prueba

7.1 Configuración de Ambientes

Ambiente	Propósito	URL	Base de Datos	Datos
Desarrollo	Desarrollo activo	localhost:5173	dev_pepsico	Datos sintéticos
Testing	Pruebas QA	test.pepsico.local	test_pepsico	Datos de prueba
Staging	Pre-producción	staging.pepsico.local	staging_pepsico	Copia de producción
Producción	Usuarios finales	app.pepsico.cl	prod_pepsico	Datos reales

7.2 Datos de Prueba

Usuarios de Prueba:

Rol	RUT	Usuario	Contraseña	Permisos
Administrador	11111111-1	admin@test.cl	Test1234	Todos

Rol	RUT	Usuario	Contraseña	Permisos
Jefe Taller	22222222-2	jefe@test.cl	Test1234	Gestión OT
Mecánico	33333333-3	mecanico@test.cl	Test1234	OT asignadas
Guardia	44444444-4	guardia@test.cl	Test1234	Ingresos
Asistente	55555555-5	asistente@test.cl	Test1234	Inventario

Vehículos de Prueba:

Patente	Tipo	Marca	Modelo	Año	Región
AAAA-11	Camión	Chevrolet	NPR	2020	RM
BBBB-22	Camión	Chevrolet	NPR	2021	RM
CCCC-33	Camioneta	Toyota	Hilux	2019	V
DDDD-44	Auto	Suzuki	Swift	2022	RM

Repuestos de Prueba:

Código	Nombre	Categoría	Stock	Precio
REP-001	Filtro de aceite	Lubricantes	50	15000
REP-002	Pastillas de freno	Frenos	30	45000
REP-003	Batería 12V	Eléctricos	10	120000
REP-004	Aceite motor 15W40	Lubricantes	100	25000

7.3 Requisitos de Infraestructura

Backend:

- Node.js 20.x
- PostgreSQL 15+
- 2 GB RAM mínimo
- 10 GB almacenamiento

Frontend:

- Servidor web estático

- 1 GB RAM mínimo

Herramientas:

- Git / GitHub
- Docker (opcional)
- Postman
- Cypress

8. Recursos y Responsabilidades

8.1 Equipo de Pruebas

Rol	Nombre	Responsabilidades	Dedicación
QA Lead	Por asignar	Planificación, coordinación, reportes	100%
QA Tester 1	Por asignar	Ejecución de pruebas funcionales	100%
QA Tester 2	Por asignar	Ejecución de pruebas de integración	100%
Performance Tester	Por asignar	Pruebas de carga y rendimiento	50%
Security Tester	Por asignar	Pruebas de seguridad	50%
Desarrolladores	Equipo Dev	Pruebas unitarias, corrección de defectos	30%
Product Owner	Cliente	Validación UAT, aprobación	20%

8.2 Capacitación Requerida

Persona	Capacitación	Duración
QA Team	Cypress E2E Testing	2 días

Persona	Capacitación	Duración
QA Team	API Testing con Postman	1 día
Performance Tester	JMeter/k6	2 días
Security Tester	OWASP Top 10	1 día

9. Cronograma de Pruebas

9.1 Fases del Proyecto

Fase	Duración	Actividades de Prueba
Sprint 1-2	4 semanas	Pruebas unitarias de módulos base
Sprint 3-4	4 semanas	Pruebas de integración API
Sprint 5-6	4 semanas	Pruebas funcionales E2E
Sprint 7	2 semanas	Pruebas de rendimiento y seguridad
Sprint 8	2 semanas	Pruebas de regresión completas
Sprint 9	2 semanas	UAT con usuarios finales
Pre-Producción	1 semana	Smoke testing en staging
Producción	Continuo	Monitoreo y pruebas de humo

9.2 Plan Detallado por Sprint

Sprint 1-2: Fundamentos

- Semana 1-2: Configurar ambientes de prueba
- Semana 2-3: Desarrollar casos de prueba
- Semana 3-4: Pruebas unitarias de autenticación y usuarios
- Objetivo: 80% cobertura en módulos completados

Sprint 3-4: Integración

- Semana 5-6: Pruebas de API de vehículos e ingresos

- Semana 7-8: Pruebas de integración OT e inventario
- Objetivo: Todos los endpoints validados

Sprint 5-6: Funcionalidad E2E

- Semana 9-10: Pruebas funcionales de flujos críticos
- Semana 11-12: Pruebas de reportes y notificaciones
- Objetivo: 95% casos de uso validados

Sprint 7: Rendimiento y Seguridad

- Semana 13: Pruebas de carga
- Semana 14: Auditoría de seguridad
- Objetivo: Cumplir RNF-02 y RNF-04

Sprint 8: Regresión

- Semana 15-16: Ejecución completa de suite de regresión
- Objetivo: 100% suite automatizada ejecutada

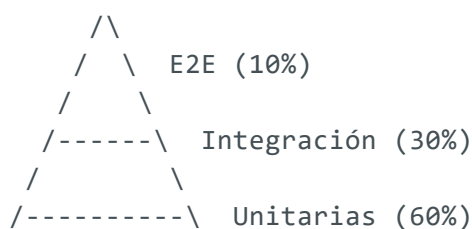
Sprint 9: UAT

- Semana 17-18: Pruebas con usuarios finales
- Objetivo: Aprobación formal del cliente

10. Automatización de Pruebas

10.1 Estrategia de Automatización

Pirámide de Pruebas:



10.2 Herramientas de Automatización

Tipo	Herramienta	Justificación
Unitarias Backend	Jest	Ecosistema Node.js, rápido
Unitarias Frontend	Vitest	Compatible con Vite, moderno
Integración API	Supertest + Jest	Pruebas de endpoints
E2E	Cypress	Fácil de usar, buen soporte React
Rendimiento	k6	Scripting en JavaScript, CI-friendly
Seguridad	OWASP ZAP	Estándar industria, gratuito
CI/CD	GitHub Actions	Integrado con repositorio

10.3 Casos a Automatizar

Prioridad Alta (Automatizar primero):

- Login y autenticación
- CRUD de vehículos
- Creación y actualización de OT
- Búsquedas principales
- APIs críticas

Prioridad Media:

- Gestión de inventario
- Reportes
- Notificaciones
- Gestión de usuarios

No automatizar (Manual):

- Pruebas de usabilidad
- Validación visual de diseño
- Casos exploratorios
- UAT

10.4 Integración Continua

Pipeline CI/CD:

```
# .github/workflows/test.yml
name: Tests

on: [push, pull_request]

jobs:
  test-backend:
    runs-on: ubuntu-latest
    steps:
      - Checkout code
      - Install dependencies
      - Run unit tests (Jest)
      - Run integration tests
      - Upload coverage to Codecov

  test-frontend:
    runs-on: ubuntu-latest
    steps:
      - Checkout code
      - Install dependencies
      - Run unit tests (Vitest)
      - Run E2E tests (Cypress)
      - Upload coverage

  security-scan:
    runs-on: ubuntu-latest
    steps:
      - Run npm audit
      - Run Snyk scan
      - Run OWASP dependency check
```

11. Métricas y Reportes

11.1 Métricas Clave

Métrica	Objetivo	Fórmula
Cobertura de Código	Backend: 80%, Frontend: 70%	(Líneas ejecutadas / Total líneas) × 100
Tasa de Éxito	> 95%	(Tests pasados / Total tests) × 100

Métrica	Objetivo	Fórmula
Densidad de Defectos	< 1 por funcionalidad	Defectos encontrados / Funcionalidades
Tasa de Detección	> 90% antes de UAT	Defectos en testing / Total defectos
Tiempo de Resolución	Críticos < 24h	Tiempo entre apertura y cierre
Cobertura de Requerimientos	100%	RF testeados / Total RF

11.2 Reportes

Reporte Diario (Durante Sprints):

- Tests ejecutados hoy
- Tests pasados/fallados
- Nuevos defectos encontrados
- Defectos resueltos
- Bloqueos

Reporte Semanal:

- Progreso contra plan
- Métricas de calidad
- Top 5 defectos críticos
- Cobertura de código actualizada
- Riesgos identificados

Reporte de Fin de Sprint:

- Resumen ejecutivo
- Todas las métricas
- Comparación con sprint anterior
- Funcionalidades validadas
- Pendientes para siguiente sprint

Reporte Final (Pre-Producción):

- Resumen completo del ciclo de pruebas
- Cobertura final de todos los tipos de prueba
- Lista de defectos conocidos en producción
- Recomendaciones
- Sign-off de aprobación

11.3 Dashboard de Pruebas

Herramientas:

- Codecov para cobertura de código
- GitHub Projects para tracking
- Grafana + k6 para rendimiento
- SonarQube para calidad de código

Métricas en Dashboard:

- Cobertura de código (backend/frontend)
- Tests ejecutados por día
- Tendencia de defectos
- Tiempo de respuesta de APIs
- Estado de ambientes

12. Riesgos y Mitigación

12.1 Riesgos Identificados

ID	Riesgo	Probabilidad	Impacto	Mitigación
R-01	Retrasos en desarrollo afectan pruebas	Alta	Alto	Buffer de 1 semana en cronograma
R-02	Falta de datos de prueba realistas	Media	Medio	Generador de datos sintéticos
R-03	Ambiente de pruebas inestable	Media	Alto	Infraestructura en Docker

ID	Riesgo	Probabilidad	Impacto	Mitigación
R-04	Recursos de QA insuficientes	Baja	Alto	Automatización agresiva
R-05	Defectos críticos en UAT	Media	Crítico	Pruebas exhaustivas previas
R-06	Cambios de requerimientos tardíos	Media	Alto	Proceso de control de cambios
R-07	Problemas de rendimiento en producción	Media	Crítico	Pruebas de carga completas
R-08	Vulnerabilidades de seguridad	Baja	Crítico	Auditoría de seguridad externa

12.2 Contingencias

- **Plan B para UAT:** Si usuarios no disponibles, equipo interno simula roles
- **Ambiente de respaldo:** Segundo ambiente de testing disponible
- **Recursos externos:** Contrato con consultora QA si necesario
- **Extensión de timeline:** Aprobación de 2 semanas adicionales si crítico

13. Criterios de Calidad y Aceptación

13.1 Definición de "Hecho" (Definition of Done)

Una funcionalidad se considera completa cuando:

- ✓ Código implementado y revisado (code review)
- ✓ Pruebas unitarias escritas y pasando (> 80% cobertura)
- ✓ Pruebas de integración ejecutadas exitosamente
- ✓ Pruebas funcionales E2E pasando
- ✓ Sin defectos críticos o bloqueantes abiertos
- ✓ Documentación técnica actualizada

- ✓ Aprobación de Product Owner

13.2 Criterios de Aceptación Global

El sistema completo será aceptado para producción si:

- ✓ 100% de requerimientos funcionales implementados y validados
 - ✓ 100% de casos de uso críticos funcionando
 - ✓ 95% de casos de uso de alta prioridad funcionando
 - ✓ Cobertura de código: Backend > 80%, Frontend > 70%
 - ✓ Tasa de éxito de pruebas automatizadas > 95%
 - ✓ Cero defectos de severidad crítica abiertos
 - ✓ Máximo 5 defectos de severidad alta abiertos (con workaround)
 - ✓ Todos los RNF cumplidos y validados
 - ✓ Pruebas de rendimiento exitosas (< 3s respuesta)
 - ✓ Auditoría de seguridad pasada (cero críticos)
 - ✓ UAT completada con aprobación formal de cliente
 - ✓ Documentación de usuario completa
 - ✓ Capacitación de usuarios realizada
 - ✓ Plan de rollback documentado
 - ✓ Monitoreo en producción configurado
-

14. Entregables

14.1 Documentos

- ✓ Plan de Pruebas Detallado (este documento)
- ✓ Casos de Prueba Automatizados (código en /tests)
- ✓ Casos de Prueba Manuales (spreadsheet)
- ✓ Reporte de Cobertura de Código
- ✓ Reporte de Pruebas de Rendimiento
- ✓ Reporte de Auditoría de Seguridad
- ✓ Reporte de UAT con firmas
- ✓ Lista de Defectos Conocidos
- ✓ Reporte Final de Pruebas

14.2 Artefactos

- Scripts de pruebas automatizadas
- Datos de prueba y scripts de seeding
- Colecciones de Postman
- Scripts de carga (JMeter/k6)
- Videos de ejecución de pruebas E2E
- Evidencias de UAT (screenshots, videos)

15. Glosario

Término	Definición
API	Application Programming Interface - Interfaz de comunicación entre sistemas
CI/CD	Continuous Integration / Continuous Deployment - Integración y despliegue continuo
E2E	End-to-End - Pruebas de extremo a extremo
JWT	JSON Web Token - Token de autenticación
OT	Orden de Trabajo
RBAC	Role-Based Access Control - Control de acceso basado en roles
RF	Requerimiento Funcional
RNF	Requerimiento No Funcional
TDD	Test-Driven Development - Desarrollo dirigido por pruebas
UAT	User Acceptance Testing - Pruebas de aceptación del usuario
UI/UX	User Interface / User Experience - Interfaz y experiencia de usuario

16. Referencias

Documento	Ubicación
Especificación de Requerimientos	Contexto proyecto/analisis-diseño/Especificación de Requerimientos del Sistema.docx
Casos de Uso Detallados	Contexto proyecto/analisis-diseño/Casos_de_Uso_Detallados.docx
Arquitectura del Sistema	Contexto proyecto/analisis-diseño/Arquitectura_del_Sistema.docx
Modelo de Datos	backend/prisma/schema.prisma
API Endpoints	Contexto proyecto/analisis-diseño/API_Endpoints_Especificación.docx
Diagrama de Casos de Uso	docs/casos-uso/Diagrama_Casos_de_Uso_Simplificado.md

17. Aprobaciones

Rol	Nombre	Firma	Fecha
QA Lead	_____	_____	//2025
Tech Lead	_____	_____	//2025
Product Owner	_____	_____	//2025
Cliente PepsiCo	_____	_____	//2025

18. Control de Versiones

Versión	Fecha	Autor	Cambios
0.1	10/10/2025	Equipo QA	Borrador inicial
0.5	12/10/2025	QA Lead	Casos de prueba detallados
1.0	15/10/2025	Equipo Completo	Versión final aprobada

Este plan de pruebas es un documento vivo y será actualizado según sea necesario durante el ciclo de vida del proyecto.