

GESTIÓN DE PROYECTO - MUNANI APP V2

Sistema E-commerce de Barritas Nutritivas - Offline-First

2. CICLO DE VIDA DEL PROYECTO

2.1. INICIO - Definir Objetivo y Alcance

Objetivo del Proyecto

Desarrollar una aplicación móvil empresarial de e-commerce para la comercialización de barritas nutritivas "Munani", con capacidad offline-first que permita gestionar inventario, ventas, compras, transferencias y pedidos de clientes en múltiples ubicaciones (tiendas y almacenes), garantizando sincronización automática bidireccional con el servidor cuando haya conexión a internet.

Alcance del Proyecto

Incluye:

- Sistema de autenticación con 4 roles (Admin, Gerente Tienda, Gerente Almacén, Cliente)
- Gestión completa de productos y variantes
- Control de inventario multi-ubicación con alertas
- Sistema de transferencias entre ubicaciones con aprobación
- Gestión de compras a proveedores
- Registro de ventas con ajuste automático de inventario
- Carrito de compras para clientes con revisión de pedidos
- Sistema de reportes analíticos (ventas, compras, transferencias)
- Sincronización offline-first bidireccional
- Base de datos local (Isar) y remota (Supabase)
- Seguridad implementada según OWASP Top 10
- Aplicación móvil multiplataforma (Android/iOS)

Límites:

- Sistema de facturación electrónica (SUNAT)
- Integración con pasarelas de pago automáticas
- Sistema de delivery tracking en tiempo real
- Aplicación web (solo móvil)
- Sistema de notificaciones push
- Chat en tiempo real
- Geolocalización de entregas

Actores Involucrados (Stakeholders)

Rol	Nombre	Responsabilidades	Tiempo Dedicación
-----	--------	-------------------	-------------------

Rol	Nombre	Responsabilidades	Tiempo Dedicación
Sponsor	Danae Revollo	<ul style="list-style-type: none"> • Proveer financiamiento • Aprobar presupuesto • Validar entregables • Definir prioridades de negocio • Tomar decisiones estratégicas 	10% (Reuniones semanales)
Project Manager / Team Lead	David	<ul style="list-style-type: none"> • Planificación del proyecto • Gestión de equipo y recursos • Seguimiento de cronograma • Gestión de riesgos • Comunicación con sponsor • Desarrollo backend y sincronización • Arquitectura del sistema 	100% (Full-time)
Developer Full-Stack	Jonas	<ul style="list-style-type: none"> • Desarrollo frontend (UI/UX) • Implementación de features • Testing de componentes • Documentación técnica • Code reviews 	100% (Full-time)
Developer Full-Stack	Daniel	<ul style="list-style-type: none"> • Desarrollo backend (API) • Base de datos (Supabase) • Seguridad (OWASP) • Integración y deployment • Code reviews 	100% (Full-time)

⌚ Objetivos SMART del Proyecto

Objetivo	SMART
Specific	Desarrollar app móvil e-commerce offline-first para Munani con 13 módulos funcionales <ul style="list-style-type: none"> • 13 features completas • 100% funcionalidad offline
Measurable	<ul style="list-style-type: none"> • <2s tiempo de respuesta • 95% cobertura de tests críticos • 0 vulnerabilidades OWASP críticas
Achievable	Equipo de 3 desarrolladores experimentados, stack tecnológico probado (Flutter, Supabase, Isar)
Relevant	Soluciona problema real de gestión multi-ubicación sin depender de internet permanente
Time-bound	16 semanas (4 meses) dividido en 8 sprints de 2 semanas cada uno

2.2. PLANIFICACIÓN - Cronograma, Fases, Presupuestos, Recursos

Cronograma General (16 Semanas - 8 Sprints)

Mes 1 (Semanas 1-4): Fundamentos y Core

 └ Sprint 1 (Sem 1-2): Setup + Auth + Core

- └ Configuration del proyecto
- └ CI/CD pipeline
- └ Arquitectura base
- └ Sistema de autenticación
- └ Base de datos local/remota

 └ Sprint 2 (Sem 3-4): Products + Locations

- └ CRUD de productos y variantes
- └ Gestión de tiendas y almacenes
- └ Sistema de permisos por rol
- └ UI básica

Mes 2 (Semanas 5-8): Inventory y Operaciones

 └ Sprint 3 (Sem 5-6): Inventory + Transfers

- └ Control de inventario multi-ubicación
- └ Alertas de stock
- └ Sistema de transferencias
- └ Workflow de aprobación

 └ Sprint 4 (Sem 7-8): Purchases + Suppliers

- └ Gestión de proveedores
- └ Compras con items
- └ Recepción de compras
- └ Ajuste automático de inventario

Mes 3 (Semanas 9-12): Ventas y Clientes

 └ Sprint 5 (Sem 9-10): Sales + Customers

- └ Registro de ventas
- └ Gestión de clientes
- └ Integración con inventario
- └ Numeración automática

 └ Sprint 6 (Sem 11-12): Cart + Orders

- └ Carrito de compras cliente
- └ Upload de comprobantes
- └ Revisión de pedidos por gerentes
- └ Workflow completo de pedidos

Mes 4 (Semanas 13-16): Reportes y Finalización

 └ Sprint 7 (Sem 13-14): Reports + Polish

- └ Reportes de ventas/compras/transferencias
- └ Gráficos y visualizaciones
- └ Optimización de performance
- └ Refinamiento de UI/UX

 └ Sprint 8 (Sem 15-16): Testing + Deploy

- └ Testing integral (E2E)
- └ Corrección de bugs críticos

- └── Documentación final
- └── Capacitación de usuarios
- └── Deployment a producción

Calendario de Sprints

Sprint	Semanas	Inicio	Fin	Entregable Principal
Sprint 1	1-2	2025-01-13	2025-01-26	Sistema de autenticación + Core funcional
Sprint 2	3-4	2025-01-27	2025-02-09	CRUD Productos + Locations
Sprint 3	5-6	2025-02-10	2025-02-23	Inventario + Transferencias
Sprint 4	7-8	2025-02-24	2025-03-09	Compras + Proveedores
Sprint 5	9-10	2025-03-10	2025-03-23	Ventas + Clientes
Sprint 6	11-12	2025-03-24	2025-04-06	Carrito + Sistema de Pedidos
Sprint 7	13-14	2025-04-07	2025-04-20	Reportes + Optimización
Sprint 8	15-16	2025-04-21	2025-05-04	Testing + Deploy

Recursos del Proyecto

Recursos Técnicos:

-  3 laptops de desarrollo (existentes)
-  2 dispositivos Android de prueba (existentes)
-  1 dispositivo iOS de prueba (existente)
-  Cuenta Supabase Pro
-  Herramientas de gestión: Trello/Jira (Free)

Recursos de Conocimiento:

-  Documentación Flutter oficial
-  Documentación Supabase
-  Clean Architecture guidelines
-  OWASP Security guidelines

Gestión de Riesgos

Matriz de Riesgos Identificados:

ID	Riesgo	Probabilidad	Impacto	Severidad	Mitigación	Contingencia

ID	Riesgo	Probabilidad	Impacto	Severidad	Mitigación	Contingencia
R01	Retrasos en desarrollo por complejidad técnica	Media (40%)	Alto	 Alta	<ul style="list-style-type: none"> • Pair programming • Spike de investigación • Refinamiento continuo 	<ul style="list-style-type: none"> • Reducir alcance no crítico • Extender 1 sprint
R02	Conflictos de sincronización offline compleja	Media (30%)	Alto	 Media-Alta	<ul style="list-style-type: none"> • Strategy de last-write-wins • Timestamps consistentes • Testing exhaustivo 	<ul style="list-style-type: none"> • Implementar resolución manual • Logs detallados
R03	Cambios de alcance por sponsor	Media (35%)	Medio	 Media	<ul style="list-style-type: none"> • Reuniones semanales • Backlog priorizado • Definition of Done clara 	<ul style="list-style-type: none"> • Agregar sprint adicional • Cobro de change orders
R04	Problemas de performance en dispositivos viejos	Baja (20%)	Medio	 Baja-Media	<ul style="list-style-type: none"> • Testing en dispositivos reales • Optimización continua • Lazy loading 	<ul style="list-style-type: none"> • Definir requisitos mínimos • Modo light
R05	Pérdida de desarrollador clave	Baja (15%)	Alto	 Media	<ul style="list-style-type: none"> • Documentación continua • Code reviews • Pair programming 	<ul style="list-style-type: none"> • Contratar reemplazo urgente • Redistributions tasks
R06	Vulnerabilidades de seguridad descubiertas	Media (25%)	Alto	 Media-Alta	<ul style="list-style-type: none"> • OWASP guidelines • Security reviews • Penetration testing 	<ul style="list-style-type: none"> • Hotfix inmediato • Parche de seguridad

ID	Riesgo	Probabilidad	Impacto	Severidad	Mitigación	Contingencia
R07	Problemas con servicios de Supabase	Baja (10%)	Alto	● Baja-Media	<ul style="list-style-type: none"> • Plan Pro con SLA • Monitoreo 24/7 • Backup automático 	<ul style="list-style-type: none"> • Failover a Supabase self-hosted • Migrar a Firebase
R08	Rechazo de app stores (Play/App Store)	Baja (15%)	Medio	● Baja	<ul style="list-style-type: none"> • Seguir guidelines estrictas • Pre-review de políticas 	<ul style="list-style-type: none"> • Corrección y resubmit • Distribución enterprise

Plan de Respuesta a Riesgos:

Estrategias:

- Evitar:** Cambiar plan para eliminar riesgo
- Mitigar:** Reducir probabilidad o impacto
- Transferir:** Pasar riesgo a terceros (seguro, proveedor)
- Aceptar:** Asumir el riesgo y preparar contingencia

Impacto de cambios:

Si ALCANCE ↓ (se reduce): → TIEMPO ↓ (menos desarrollo) → COSTO ↓ (menos horas) → CALIDAD = (enfoque en lo esencial)

Si ALCANCE ↑ (se aumenta): → TIEMPO ↑ (más desarrollo) → COSTO ↑ (más horas/recursos) → CALIDAD = (riesgo si no se ajustan otros)

Gestión de Cambios de Alcance (Change Management):

Cambio Solicitado	Evaluación	Impacto	Decisión
"Agregar notificaciones push"	+2 semanas	+\$3,200	X Fuera de presupuesto
"Integrar facturación SUNAT"	+4 semanas	+\$6,400	X Cambio mayor, fase 2
"Mejorar UX de carrito"	+1 semana	+\$1,600	<input checked="" type="checkbox"/> Dentro de contingencia
"Agregar dashboard gráfico"	Ya incluido	\$0	<input checked="" type="checkbox"/> En alcance original

⚙ CALIDAD (Centro del Triángulo)

Definición: El grado en que el producto cumple con los requisitos y expectativas.

> **🟡 REGLA DE ORO:** La CALIDAD NUNCA es negociable. No importa qué vértice se ajuste (tiempo, costo, alcance), la calidad debe mantenerse constante y alta.

Atributos de Calidad para Munani App:

Atributo	Descripción	Métrica	Mínimo Aceptable
Funcionalidad	Cumple todos los requisitos	User stories completadas	100%
Confiabilidad	Funciona consistentemente	Uptime, crashes	99% uptime, <1% crash rate
Usabilidad	Fácil de usar	SUS score	≥70/100
Eficiencia	Respuesta rápida	Tiempo de respuesta	<2 segundos
Mantenibilidad	Fácil de mantener	Code coverage, tech debt	≥80%, <15% debt
Seguridad	Protege datos	Vulnerabilidades	0 críticas, 0 altas
Portabilidad	Funciona en múltiples dispositivos	Dispositivos soportados	Android 7+, iOS 12+

Proceso de Aseguramiento de Calidad:

🏛 Escenarios de Balanceo del Triángulo

Escenario 1: 🚨 **Crisis - Sponsor pide adelantar lanzamiento 2 semanas**

Situación: Se necesita lanzar en 14 semanas en lugar de 16.

Análisis de Impacto:

TIEMPO: 16 semanas → 14 semanas (-12.5%)

Opciones: _____

A) AUMENTAR COSTO (horas extra / más recursos) ┌─ Costo adicional: +\$3,200 (horas extra) ┌─ Mantener alcance: 100% └─ Mantener calidad:

B) REDUCIR ALCANCE (MVP) ┌─ Costo: Sin cambio ┌─ Reducir alcance: -20% (postergar reportes) └─ Mantener calidad:

C) RECHAZAR (recomendado si compromete calidad) ┌─ Explicar impacto al sponsor ┌─ Proponer alternativa: Lanzamiento por fases └─ Fase 1: Core features (12 sem) Fase 2: Reportes avanzados (4 sem)

Decisión Recomendada: Opción C (Lanzamiento por fases)

Razón: Mantiene calidad sin aumentar costo ni comprometer features críticas.

Escenario 2: 💵 **Recorte de Presupuesto - Reducción del 15%**

Situación: Sponsor reduce presupuesto a \$39,542 (-\$6,978)

Análisis de Impacto:

COSTO: \$46,520 → \$39,542 (-15%)

Opciones: _____

A) REDUCIR ALCANCE (priorizar) |— Eliminar: Carrito de clientes (-20%) |— Tiempo: 16 semanas (sin cambio)
└ Calidad: (enfoque en core)

B) EXTENDER TIEMPO (reducir velocidad) |— Pasar de 3 a 2 desarrolladores |— Tiempo: 16 → 22 semanas (+37.5%)
└ Calidad: (mismo estándar)

C) NEGOCIAR (pago por fases) |— Fase 1: \$25,000 (core features) |— Fase 2: \$14,520 (features avanzadas)
Sponsor paga según cashflow _____

Decisión Recomendada: Opción A (Reducir alcance a MVP)

Razón: Entrega valor rápido, calidad alta, posibilidad de Fase 2.

Escenario 3: 💼 **Expansión de Alcance - "Aregar módulo de facturación SUNAT"**

Situación: Sponsor solicita agregar integración con SUNAT.

Análisis de Impacto:

ALCANCE: 13 módulos → 14 módulos (+7.7%)

Estimación del cambio: _____ • Complejidad: Alta (API SUNAT compleja) • Tiempo adicional: +3 semanas • Costo adicional: +\$4,800 (3 semanas × 3 devs × 40h × \$20/h)

Opciones: _____

A) APROBAR CHANGE ORDER (aumentar tiempo y costo) |— Tiempo: 16 → 19 semanas (+18.7%) |— Costo: \$46,520 → \$51,320 (+10.3%)
└ Alcance: +Facturación SUNAT └ Calidad: (sin compromiso)

B) POSTERGAR A FASE 2 (recomendado) |— Entregar proyecto original en 16 semanas |— Facturación SUNAT como proyecto separado |— Presupuesto adicional: \$8,000 - \$12,000 |— Timeline: +1-2 meses post-lanzamiento

C) RECHAZAR (si no es crítico para negocio) |— Enfoque en alcance original |— Evaluación de prioridad vs. impacto |— Alternativa: Facturación manual/externa —

Decisión Recomendada: Opción B (Postergar a Fase 2)

Razón: Evita scope creep, mantiene timeline y calidad del proyecto original.

Reportes de Avance

Reporte Semanal al Sponsor:

Formato de Email:

Asunto: Munani App - Reporte Semana #X

Hola Danae,

Resumen semanal del proyecto Munani App:

COMPLETADO ESTA SEMANA: • [Lista de historias de usuario completadas] • [Métricas de progreso]

EN PROGRESO: • [Lista de tareas actuales]

PRÓXIMA SEMANA: • [Lista de tareas planeadas]

BLOQUEOS/RIESGOS: • [Ninguno / Descripción de bloqueos]

MÉTRICAS: • Progreso: XX% completado • Presupuesto: \$XX gastado de \$XX (XX%) • Cronograma: En tiempo / X días de retraso

DEMO: [Link a video de demo si aplica]

Saludos, David - Project Manager

Dashboard de Proyecto (Actualización Diaria):

MUNANI APP - DASHBOARD

Progreso General: 60% (210/350 SP)

Sprint Actual: Sprint 5 - Ventas y Clientes |— Progreso Sprint: 80% (40/50 SP)

⌚ Cronograma: |— Inicio: 2025-01-13 |— Hoy: 2025-03-15 (Semana 10 de 16) |— Fin Estimado: 2025-05-04

💰 Presupuesto: |— Total: \$46,520 |— Gastado: \$28,912 (62%) |— Restante: \$17,608

Quality Gates: |— Code Coverage: 84% |— Bugs Críticos: 0 |— Vulnerabilidades: 0 |— Performance: <2s

⚠ Riesgos Activos: 2 (1 media, 1 baja)

👤 Documento preparado por: Equipo Munani App V2

📅 Última actualización: 2025-01-12

✉️ Contacto: david@munaniapp.com

⚠ CONFIDENCIAL - Este documento contiene información propietaria del proyecto Munani App V2.