Fit Legacy App - Documento de Especificación Técnica **MVP**

📌 Información General

Nombre del Proyecto: Fit Legacy - Fitness Tracking App

Versión: 1.0 (MVP) Fecha: Octubre 2025

Plataforma: Mobile (Android/iOS)



🎯 Visión del Producto

Fit Legacy es una aplicación de fitness gamificada que guía al usuario a través de un "Camino del Legado" visual e interactivo, acompañado por "Legacito", una mascota virtual que reacciona emocionalmente al progreso del usuario. La app combina entrenamiento físico con elementos de reflexión y conocimiento.



🖀 Arquitectura del Sistema

Stack Tecnológico Recomendado

Frontend

- Framework: React Native o Flutter
- Estado: Redux/Context API (React) o Riverpod (Flutter)
- Navegación: React Navigation / Flutter Navigator
- Animaciones: Lottie, Animated API
- UI Components: Custom components basados en diseño Figma

Backend

- BaaS: Xano o Firebase
- Autenticación: Firebase Auth o Xano Auth
- Base de Datos: Xano DB / Firestore
- **API:** RESTful endpoints
- Storage: Cloud Storage para assets

Herramientas

- **Versionado:** Git + GitHub/GitLab CI/CD: GitHub Actions / Bitrise
- **Testing:** Jest + React Native Testing Library
- **Analytics:** Firebase Analytics
- **Crash Reporting:** Firebase Crashlytics

📱 Estructura de Pantallas

1. Módulo de Autenticación

1.1 Login Screen

Ruta: /login

Elementos UI:

- Logo de Fit Legacy
- Campo: Email/Username
- Campo: Password (con toggle show/hide)
- Botón: "Login"
- Link: "¿Olvidaste tu contraseña?"
- Link: "¿No tienes cuenta? Sign up"

Validaciones:

- Email válido
- Password mínimo 8 caracteres
- Manejo de errores (credenciales incorrectas)

Estados:

- Loading durante autenticación
- Error state con mensaje
- Success → Redirect a Camino del Legado

1.2 Sign Up Screen

Ruta: /signup

Elementos UI:

- Campos: Nombre, Email, Password, Confirmar Password
- Botón: "Sign up"
- Link: "¿Ya tienes cuenta? Login"

Validaciones:

- Todos los campos obligatorios
- Email único
- Passwords coinciden
- Términos y condiciones

1.3 Reset Password Screen

Ruta: /reset-password

Elementos UI:

- Campo: Email
- Botón: "Enviar link de recuperación"
- Mensaje de confirmación

2. Módulo del Camino del Legado (Core)

2.1 Camino del Legado Screen

Ruta: /legacy-path

Componente Principal: Scroll Vertical Infinito

Estructura:



```
[Header con Progress]

↓

[Legacito Flotante]

↓

[Lista de Nodos]

- Nodo 1 (Entrenamiento)

- Nodo 2 (Refuerzo)

- Nodo 3 (Reflexión)

- Nodo 4 (Descanso)

- Nodo 5 (Maestría)

↓

[Carga Progresiva]
```

Elementos UI:

- Header sticky con título "Tu Camino hacia el Legado"
- Barra de progreso global
- Sección "Comienza aquí" con frase motivacional
- Nodos visuales con:
 - Ícono/Avatar según tipo
 - Estado visual (activo/completado/bloqueado)
 - o Nombre del nodo
 - Frase descriptiva
 - Línea conectora entre nodos
- Legacito animado (posición contextual)

Interacciones:

- Tap en nodo activo → Abre detalle del nodo
- Tap en nodo bloqueado → Mensaje "Completa el anterior"
- Tap en nodo completado → Muestra resumen
- Pull to refresh → Actualiza estado desde backend
- Scroll infinito → Carga más nodos al llegar al 80%

2.2 Componente: Nodo

Props:



typescript

```
interface Nodo {
  id: string;
  tipo: 'entrenamiento' | 'refuerzo' | 'reflexion' | 'descanso' | 'maestria';
  estado: 'activo' | 'completado' | 'bloqueado';
  titulo: string;
  frase: string;
  etapa: string;
  orden: number;
  icono: string;
  colorFondo: string;
  contenido: ContenidoNodo;
}
```

Estados Visuales:

- Activo: Iluminado, frase motivacional visible, animación pulsante
- Completado: Animación de validación, checkmark, frase trascendental
- Bloqueado: Icono de candado, opacidad reducida, frase estoica

2.3 Componente: Legacito

Props:



typescript

```
interface Legacito {
  estado: 'feliz' | 'motivado' | 'pensativo' | 'celebrando';
  frase: string;
  posicion: 'top' | 'middle' | 'bottom';
  animacion: LottieAnimation;
}
```

Comportamiento:

- Cambia expresión según progreso del usuario
- Muestra frases contextuales
- Animaciones de entrada/salida suaves
- Gestos interactivos (tap para mensaje aleatorio)

3. Módulo de Trivia

3.1 Trivia Modal

Tipo: Modal overlay sobre Camino del Legado

Elementos UI:

- Header con ícono de conocimiento
- Título: "Trivia de Conocimiento"
- Pregunta: "¿Cuál es el tiempo de descanso recomendado entre series para hipertrofia muscular?"
- 4 Opciones de selección:
 - A: 30-45 segundos
 - B: 60-90 segundos
 - C: 2-3 minutos
 - D: 5+ minutos
- Botón: "Saltar" (amarillo)
- Botón: "Confirmar" (amarillo con negro)

Flujo:

- 1. Usuario selecciona una opción
- 2. Opción se destaca visualmente
- 3. Usuario confirma
- 4. Se valida respuesta:
 - Correcta → Animación de éxito + continúa al nodo
 - o Incorrecta → Muestra respuesta correcta + opción de reintentar

Estados:

- Sin selección (Confirmar disabled)
- Con selección (Confirmar enabled)
- Validando (Loading)
- Resultado (Success/Error)

4. Módulo de Nodos Específicos

4.1 Nodo de Entrenamiento

Contenido:

- Nombre del ejercicio
- GIF o video demostrativo
- Series y repeticiones
- Indicaciones técnicas
- Timer integrado
- Botón "Marcar como completado"

4.2 Nodo de Refuerzo

Contenido:

- Ejercicio complementario
- Enfoque muscular
- Técnica
- Botón de completado

4.3 Nodo de Reflexión

Contenido:

- Pregunta reflexiva
- Campo de texto libre
- Botón "Guardar reflexión"

4.4 Nodo de Descanso

Contenido:

- Timer de descanso
- Tips de recuperación
- Hidratación reminder

4.5 Nodo de Maestría

Contenido:

- Resumen de etapa
- Estadísticas de progreso
- Insignia desbloqueada
- Mensaje de Legacito

📱 Estructura de Datos

Modelo de Usuario



```
json
```

```
"id": "user_123",
"nombre": "Juan Pérez",
"email": "juan@example.com",
"fechaRegistro": "2025-10-07T12:00:00Z",
"progreso": {
 "nodoActual": "node 005",
 "nodosCompletados": ["node_001", "node_002", "node_003", "node_004"],
 "etapaActual": "refuerzo",
 "porcentajeTotal": 25
"estadisticas": {
 "diasConsecutivos": 7,
 "totalNodosCompletados": 4,
 "totalReflexiones": 2,
 "triviaCorrectas": 3
"configuracion": {
 "notificaciones": true,
 "recordatorios": "08:00"
```

Modelo de Nodo



Modelo de Etapa



}

```
"id": "etapa 001",
"nombre": "Entrenamiento",
"descripcion": "Fase inicial de construcción",
"color": "#ffd700",
"nodos": ["node_001", "node_002", "node_003"],
"objetivo": "Completar 3 entrenamientos básicos"
```



API Endpoints

Autenticación



```
POST /api/auth/register
Body: { nombre, email, password }
Response: { userId, token }
POST /api/auth/login
Body: { email, password }
Response: { userId, token, user }
POST /api/auth/reset-password
Body: { email }
Response: { message }
```

Nodos



```
GET /api/nodos
Query: ?offset=0&limit=10&userId={userId}
Response: { nodos: [], hasMore: boolean }
GET /api/nodos/{nodeId}
Response: { nodo: {...} }
POST /api/nodos/{nodeId}/completar
Body: { userId, tiempo, resultado }
Response: { success, siguienteNodo }
```

Progreso



```
GET /api/progreso/{userId}
Response: { progreso: {...}, estadisticas: {...} }
POST /api/progreso/actualizar
Body: { userId, nodoId, accion }
Response: { progreso actualizado }
```

Trivia



POST /api/trivia/validar Body: { userId, nodoId, respuesta } Response: { correcta: boolean, explicacion }



锅 Sistema de Diseño

Paleta de Colores

Principales:

- Primary: #FFD700 (Amarillo dorado Legacito)
- Secondary: #1a1a2e (Azul oscuro Fondo)
- Accent: #ff6b6b (Rojo Estados de error)
- Success: #4ecdc4 (Verde agua Completado)

Estados:

- Activo: Amarillo brillante con glow
- Completado: Verde con animación de check
- Bloqueado: Gris oscuro con candado

Textos:

- Heading: Amarillo/Blanco
- Body: Gris claro
- Frases motivacionales: Amarillo

Tipografía

- **Primary Font:** Montserrat (Bold para títulos)
- Secondary Font: Open Sans (Regular para body)
- Sizes:
 - H1: 24px
 - H2: 20px
 - o Body: 16px
 - o Caption: 14px

Iconografía

- Biblioteca: Lucide Icons o Material Icons
- Custom: Iconos de Legacito y nodos específicos

🖸 Flujos de Usuario

Flujo de Onboarding



- 1. Usuario abre la app
- 2. Splash screen (Logo + Legacito)
- 3. Login/Signup screen
- 4. [Si signup] → Completar perfil básico
- 5. Intro animada: "Tu Camino hacia el Legado"
- 6. Tutorial interactivo (opcional, skip disponible)
- 7. Llega a Camino del Legado (primer nodo activo)

Flujo de Completar Nodo



- 1. Usuario tap en nodo activo
- 2. Abre detalle del nodo
- 3. Usuario completa la actividad
- 4. Aparece Trivia de Conocimiento
- 5. Usuario responde trivia
- 6. [Si correcta] → Nodo marcado como completado
- 7. [Si incorrecta] → Opción de reintentar
- 8. Animación de validación
- 9. Siguiente nodo se desbloquea
- 10. Legacito celebra con mensaje
- 11. Regresa a Camino del Legado actualizado

Flujo de Carga Progresiva



- 1. Usuario scrollea en Camino del Legado
- 2. Al llegar al 80% del scroll
- 3. Indicador de carga aparece
- 4. Backend envía siguiente batch de nodos
- 5. Nodos se añaden al final del scroll
- 6. Usuario puede continuar scrolleando

Optimizaciones y Rendimiento

Scroll Infinito

- Virtualización: Renderizar solo nodos visibles + buffer
- Lazy loading: Cargar imágenes bajo demanda
- Batch size: 10 nodos por petición
- Cache: Guardar últimos 50 nodos en memoria

Animaciones

- 60 FPS target: Usar animaciones nativas cuando sea posible
- Lottie optimizado: Archivos JSON < 100KB
- Reduce motion: Respetar preferencias de accesibilidad

Assets

- Imágenes: WebP format, múltiples resoluciones
- Videos: Streaming progresivo, calidades adaptativas
- Fonts: Subset de caracteres necesarios



Unit Tests

- Validación de formularios
- Lógica de negocio
- Transformación de datos

Integration Tests

- Flujo de autenticación completo
- API calls y respuestas
- Actualización de estado

E2E Tests

- Flujo completo de onboarding
- Completar un nodo end-to-end
- Carga progresiva de nodos

Testing Manual

- Diferentes dispositivos (iOS/Android)
- Diferentes tamaños de pantalla
- Conexión lenta/offline
- Estados edge cases



Plan de Deployment

Pre-launch Checklist

Técnico:

- Todas las funcionalidades del MVP implementadas Testing completado (95% coverage mínimo) Performance optimizado (< 3s tiempo de carga) Assets finales integrados Analytics configurado
- Crash reporting activo

Contenido:

- 5 nodos iniciales con contenido completo ■ 15+ frases de Legacito
- 5+ trivias validadas
- Instrucciones de ejercicios revisadas

Legal:

- Términos y condiciones
- Política de privacidad
- Permisos de app configurados

Despliegue

Beta Testing:

- 1. TestFlight (iOS) / Internal Testing (Android)
- 2. Grupo de 20-50 beta testers
- 3. 2 semanas de testing
- 4. Recoger feedback y fix bugs críticos

Launch:

- 1. Enviar a App Store y Google Play
- 2. Preparar marketing materials
- 3. Comunicado de prensa
- 4. Launch en redes sociales

Métricas de Éxito (MVP)

KPIs Principales

- **Retención D1:** > 40%
- **Retención D7:** > 20%
- Nodos completados promedio: > 3
- Tiempo promedio en app: > 15 min/sesión
- Tasa de conversión signup: > 60%

Métricas Secundarias

- Crash-free rate: > 99%
- ANR rate: < 0.5%
- Tiempo de carga inicial: < 3s
- Trivias completadas correctamente: > 60%

Roadmap Post-MVP

Versión 1.1 (Post-Launch +1 mes)

- Sistema de notificaciones push
- Recordatorios personalizados
- Social sharing
- Bug fixes de usuarios

Versión 1.2 (+2 meses)

- Más nodos y etapas
- Sistema de logros e insignias
- Perfil de usuario expandido
- Estadísticas detalladas

Versión 2.0 (+4 meses)

- Modo multijugador / Comunidad
- Planes personalizados con IA
- Integración con wearables

Modo offline completo



Roles y Responsabilidades

Desarrollo

- Frontend Developer: Implementar UI/UX según Figma
- Backend Developer: APIs, database, lógica de negocio
- QA Engineer: Testing y aseguramiento de calidad

Diseño

- UI/UX Designer: Refinamiento de diseños, assets
- Motion Designer: Animaciones de Legacito

Contenido

- Content Creator: Escribir frases, trivias, instrucciones
- Fitness Expert: Validar ejercicios y rutinas

Gestión

- Product Manager: Coordinación, priorización, decisiones
- Project Manager: Timeline, milestones, comunicación

Recursos y Referencias

Documentación

- Diseños en Figma: [Link al proyecto]
- Repositorio de código: [GitHub repo]
- Wiki del proyecto: [Confluence/Notion]

APIs y SDKs

- Firebase Documentation
- Xano Documentation
- React Native Documentation

Inspiración

- Duolingo (gamificación)
- Nike Training Club (fitness)
- Habitica (progreso visual)



Checklist de Desarrollo MVP

Semana 1-2: Setup y Autenticación

- Inicializar proyecto
- Configurar Git y CI/CD

•	Setup de Firebase/Xano
•	Implementar Login
•	Implementar Signup
•	Implementar Reset Password
•	Testing de autenticación
Semana 3-4: Camino del Legado	
•	Diseñar estructura de componentes
•	Implementar scroll infinito
•	Crear componente Nodo
•	Implementar estados visuales
•	Integrar con backend
•	Animaciones de transición
•	Testing de scroll y performance
Semana 5: Legacito	
•	Integrar animaciones Lottie
•	Sistema de frases contextuales
•	Comportamiento reactivo
•	Gestos interactivos
•	Testing de animaciones
Semana 6: Trivia y Nodos	
•	Implementar modal de trivia
	Lógica de validación
	Detalle de nodos (5 tipos)
	Contenido de ejemplo
	Testing de flujo completo
Semana 7: Backend y APIs	
•	Endpoints de nodos
•	Endpoints de progreso
•	Carga progresiva
•	Sincronización de estado
•	Testing de APIs
Semana 8: Polish y Testing	
•	Optimizaciones de performance
• [Refinamiento de UX
•	Testing integral
•	Fix de bugs
•	Preparar para beta
,	

Contingencias y Riesgos

Riesgos Técnicos

Riesgo: Performance degradado con muchos nodos Mitigación: Implementar virtualización desde el inicio

Riesgo: Animaciones pesadas

Mitigación: Optimizar Lottie files, usar animaciones nativas

Riesgo: Backend caído

Mitigación: Implementar modo offline con cache

Riesgos de Negocio

Riesgo: Baja retención de usuarios

Mitigación: A/B testing de onboarding, gamificación más fuerte

Riesgo: Competencia directa

Mitigación: Enfocarse en experiencia única de Legacito



& Contacto y Soporte

Product Owner: [Nombre] **Tech Lead:** [Nombre]

Email de soporte: support@fitlegacy.com

Documento creado: Octubre 2025 Última actualización: [Fecha]

Versión: 1.0