

Finanzas SDMT + Prefactura - Informe de Ejecución de Pruebas de Extremo a Extremo

Cliente: CVDEX / Ikusi PMO **Proyecto:** Financial Planning - Módulo de Entrega de Servicios Finanzas **Fecha del Informe:** 6 de diciembre de 2025 **Ambiente de Pruebas:** Desarrollo (AWS us-east-2) **Preparado Por:** Equipo de Automatización de QA **Versión:** 1.0

Resumen Ejecutivo

Este documento presenta los resultados completos de la ejecución de pruebas de extremo a extremo para los módulos Finanzas SDMT (Herramientas de Gestión de Entrega de Servicios) y Prefactura. Las pruebas se realizaron en el ambiente de desarrollo para validar funcionalidad, integración, seguridad y cumplimiento con los estándares CVDEX y los requisitos de Ikusi PMO.

Estado General

Resumen de Ejecución de Pruebas:

- **Casos de Prueba Totales:** 22
- **Aprobados:** 18
- **Fallidos:** 0
- **Bloqueados:** 0
- **Pendientes de Verificación Manual:** 4
- **Tasa de Aprobación:** 81.8%

Hallazgos Clave

Fortalezas:

- Todos los endpoints de API son funcionales y responden correctamente
- El flujo de autenticación y autorización es seguro
- Integridad de datos mantenida en todas las operaciones CRUD
- Infraestructura AWS (Lambda, DynamoDB, S3) configurada correctamente
- Distribución CloudFront sirve los activos de UI correctamente

Áreas que Requieren Atención:

- 4 casos de prueba requieren verificación manual en ambiente de producción
- Comportamiento del caché de CloudFront necesita monitoreo
- Se recomienda pruebas de rendimiento bajo carga

Recomendaciones

1. Completar verificación manual de casos de prueba pendientes en producción 2. Implementar pruebas automatizadas de rendimiento 3. Programar auditorías de seguridad regulares 4. Establecer monitoreo continuo para endpoints de API

1. Alcance y Objetivos

1.1 Alcance de Pruebas

Este esfuerzo de pruebas cubre:

Módulos Probados:

- Gestión de Catálogo de Costos SDMT
- Procesamiento de Prefactura (Pre-factura)
- Gestión de Proyectos
- Pronóstico y Planificación de Presupuesto
- Cambios y Reconciliación
- Procedimientos de Handoff

Componentes Técnicos:

- Frontend: React 19 SPA servido vía CloudFront
- Backend: Funciones AWS Lambda con API Gateway
- Base de Datos: Tablas AWS DynamoDB
- Almacenamiento: AWS S3 para documentos y artefactos
- Autenticación: AWS Cognito User Pools
- Autorización: Amazon Verified Permissions (AVP)

Tipos de Pruebas:

- Pruebas Funcionales
- Pruebas de Integración
- Pruebas de Seguridad
- Pruebas de Contratos de API
- Flujos de Usuario de Extremo a Extremo

1.2 Ambiente de Pruebas

Detalles de Infraestructura:

| Componente | Configuración | Endpoint/ID | |-----|-----|-----| | API Gateway | HTTP API |
`m3g6am67aj.execute-api.us-east-2.amazonaws.com` | | CloudFront | Distribución |
`d7t9x3j66yd8k.cloudfront.net` | | Cognito Pool | User Pool | `us-east-2_FyHLtOhY` | | Cognito Client | Web
Client | `dshos5iou44tuach7ta3ici5m` | | Región AWS | Principal | `us-east-2` | | Stage | Ambiente | `dev` |

Credenciales de Prueba:

- Usuario: `finanzas-test-user`
- Grupos: `SDT` (Equipo de Entrega de Servicios)
- Método de Autenticación: Cognito Hosted UI (OAuth2)

2. Casos de Prueba y Matriz de Trazabilidad

Caso de Prueba 1: Verificación de Salud de API

Prioridad: Crítica **Tipo:** Prueba de Humo **Requisito:** API-001

Pasos de Prueba: 1. Enviar solicitud GET al endpoint `/health` 2. Verificar respuesta HTTP 200 3. Validar estructura de respuesta

Resultado Esperado: ``json { "ok": true, "service": "finanzas-sd-api", "stage": "dev" } ``

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/01-health-check.log`

Caso de Prueba 2: Recuperación de Lista de Proyectos

Prioridad: Alta **Tipo:** Funcional **Requisito:** PROJ-001

Pasos de Prueba: 1. Autenticar con Cognito 2. Enviar solicitud GET a `/projects?limit=50` 3. Verificar respuesta HTTP 200 4. Validar que se devuelve array de proyectos

Resultado Esperado:

- HTTP 200
- Array de objetos de proyecto con campos `id` y `name`
- Al menos 1 proyecto devuelto

Resultado Real: APROBADO **Proyectos Encontrados:** 3 **Evidencia:** Ver `testing-evidence/02-projects-list.log`

Caso de Prueba 3: Catálogo de Rubros - Recuperar Ítems

Prioridad: Alta **Tipo:** Funcional **Requisito:** CAT-001

Pasos de Prueba: 1. Para cada proyecto descubierto en CP2 2. Enviar solicitud GET a `/projects/{projectId}/rubros` 3. Verificar respuesta HTTP 200 4. Validar estructura de array de rubros

Resultado Esperado:

- HTTP 200
- Array de objetos rubro
- Cada rubro tiene campos requeridos: `id`, `name`, `category`, `unitCost`

Resultado Real: APROBADO **Rubros Recuperados:** 71 en todos los proyectos **Evidencia:** Ver `testing-evidence/03-rubros-catalog-*.log`

Caso de Prueba 4: Catálogo de Rubros - Crear Ítem

Prioridad: Alta **Tipo:** Funcional **Requisito:** CAT-002

Pasos de Prueba: 1. Preparar nuevo payload de rubro con identificador CLI 2. Enviar solicitud POST a `/projects/{projectId}/rubros` 3. Verificar respuesta HTTP 200/201 4. Validar que se devuelve el rubro creado

Resultado Esperado:

- HTTP 200 o 201
- Respuesta contiene rubro creado con ID asignado
- Campos de timestamp poblados

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/04-rubros-create-*.log`

Caso de Prueba 5: Catálogo de Rubros - Verificar Creación

Prioridad: Alta **Tipo:** Funcional **Requisito:** CAT-003

Pasos de Prueba: 1. Después de CP4, enviar solicitud GET a `/projects/{projectId}/rubros` 2. Buscar en respuesta el rubro recién creado por identificador CLI 3. Verificar que aparece en la lista

Resultado Esperado:

- Rubro recién creado aparece en el catálogo
- Todos los campos coinciden con el payload de creación

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/05-rubros-verify-*.log`

Caso de Prueba 6: Pronóstico - Ventana de 6 Meses

Prioridad: Alta **Tipo:** Funcional **Requisito:** FCST-001

Pasos de Prueba: 1. Para cada proyecto 2. Enviar solicitud GET a `/projects/{projectId}/plan/forecast?months=6` 3. Verificar respuesta HTTP 200 4. Validar estructura de datos de pronóstico

Resultado Esperado:

- HTTP 200
- Array de pronóstico con 6 entradas mensuales
- Cada mes tiene campos `period`, `projected`, `actual`

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/06-forecast-6m-*.log`

Caso de Prueba 7: Pronóstico - Ventana de 12 Meses

Prioridad: Alta **Tipo:** Funcional **Requisito:** FCST-002

Pasos de Prueba: 1. Para cada proyecto 2. Enviar solicitud GET a `/projects/{projectId}/plan/forecast?months=12` 3. Verificar respuesta HTTP 200 4. Validar estructura de datos de pronóstico

Resultado Esperado:

- HTTP 200
- Array de pronóstico con 12 entradas mensuales

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/07-forecast-12m-*.log`

Caso de Prueba 8: Reconciliación - Listar Facturas

Prioridad: Media **Tipo:** Funcional **Requisito:** RECON-001

Pasos de Prueba: 1. Para cada proyecto 2. Enviar solicitud GET a `/projects/{projectId}/reconciliation/invoices` 3. Verificar respuesta HTTP 200 4. Validar estructura de array de facturas

Resultado Esperado:

- HTTP 200
- Array de objetos de factura
- Cada factura tiene `id`, `date`, `amount`, `status`

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/08-reconciliation-*.log`

Caso de Prueba 9: Seguimiento de Cambios - Listar Cambios

Prioridad: Media **Tipo:** Funcional **Requisito:** CHG-001

Pasos de Prueba: 1. Para cada proyecto 2. Enviar solicitud GET a `/projects/{projectId}/changes` 3. Verificar respuesta HTTP 200 4. Validar estructura de array de cambios

Resultado Esperado:

- HTTP 200
- Array de objetos de cambio
- Cada cambio tiene `id`, `description`, `status`, `createdAt`

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/09-changes-*.log`

Caso de Prueba 10: Documentación de Handoff

Prioridad: Media **Tipo:** Funcional **Requisito:** HO-001

Pasos de Prueba: 1. Para cada proyecto 2. Enviar solicitud GET a `/projects/{projectId}/handoff` 3. Verificar respuesta HTTP 200 4. Validar estructura de datos de handoff

Resultado Esperado:

- HTTP 200
- Objeto handoff con `status`, `documents`, `approvals`

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/10-handoff-*.log`

Caso de Prueba 11: Flujo de Autenticación - Login de Cognito

Prioridad: Crítica **Tipo:** Seguridad **Requisito:** AUTH-001

Pasos de Prueba: 1. Navegar a Cognito Hosted UI 2. Ingresar credenciales de prueba 3. Completar flujo de autorización OAuth2 4. Verificar token JWT recibido

Resultado Esperado:

- Redirección exitosa a URL de callback
- Token JWT válido en localStorage
- Token incluye claims requeridos (sub, cognito:groups)

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/11-auth-flow.log`

Caso de Prueba 12: Autorización - Acceso a Endpoint Protegido

Prioridad: Crítica **Tipo:** Seguridad **Requisito:** AUTH-002

Pasos de Prueba: 1. Intentar acceder a endpoint protegido sin token 2. Verificar HTTP 401 No Autorizado 3. Acceder al mismo endpoint con token válido 4. Verificar HTTP 200 y datos devueltos

Resultado Esperado:

- Sin token: HTTP 401
- Con token válido: HTTP 200

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/12-authorization.log`

Caso de Prueba 13: Integridad de Datos DynamoDB - Tabla Rubros

Prioridad: Alta **Tipo:** Validación de Datos **Requisito:** DB-001

Pasos de Prueba: 1. Consultar tabla de rubros de DynamoDB directamente 2. Verificar que los datos coinciden con respuestas de API 3. Verificar consistencia de datos

Resultado Esperado:

- Registros de DynamoDB coinciden con respuestas de API
- Sin datos huérfanos o corruptos
- Timestamps válidos

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/13-dynamodb-rubros.log`

Caso de Prueba 14: Integridad de Datos DynamoDB - Tabla Proyectos

Prioridad: Alta **Tipo:** Validación de Datos **Requisito:** DB-002

Pasos de Prueba: 1. Consultar tabla de proyectos de DynamoDB 2. Verificar que todos los proyectos tienen campos requeridos 3. Verificar integridad referencial

Resultado Esperado:

- Todos los proyectos tienen estructura válida
- Sin campos requeridos faltantes
- Relaciones de clave foránea intactas

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/14-dynamodb-projects.log`

Caso de Prueba 15: Almacenamiento de Documentos S3

Prioridad: Media **Tipo:** Integración **Requisito:** S3-001

Pasos de Prueba: 1. Subir documento de prueba vía API 2. Verificar que bucket S3 contiene el archivo 3. Descargar documento y verificar integridad

Resultado Esperado:

- Documento subido exitosamente a S3
- Archivo accesible vía URL firmada
- Contenido coincide con carga original

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/15-s3-upload.log`

Caso de Prueba 16: Rendimiento de Funciones Lambda

Prioridad: Media **Tipo:** Rendimiento **Requisito:** PERF-001

Pasos de Prueba: 1. Ejecutar cada endpoint de API 10 veces 2. Medir tiempos de respuesta 3. Verificar todas las respuestas bajo 3 segundos

Resultado Esperado:

- Tiempo de respuesta promedio < 1 segundo
- Percentil 95 < 2 segundos
- Sin timeouts

Resultado Real: APROBADO **Métricas:**

- Promedio: 456ms
- P95: 1.2s
- P99: 1.8s

Evidencia: Ver `testing-evidence/16-lambda-performance.log`

Caso de Prueba 17: Entrega de UI por CloudFront

Prioridad: Alta **Tipo:** Integración **Requisito:** CF-001

Pasos de Prueba: 1. Acceder a UI vía distribución CloudFront 2. Verificar que todos los activos estáticos cargan 3. Verificar encabezados de caché

Resultado Esperado:

- HTTP 200 para todos los activos
- Tipos MIME correctos
- Encabezados de caché configurados correctamente

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/17-cloudfront-ui.log`

Caso de Prueba 18: Limitación de Tasa de API

Prioridad: Media **Tipo:** Seguridad **Requisito:** SEC-001

Pasos de Prueba: 1. Enviar solicitudes rápidas a API 2. Verificar que la limitación de tasa se activa 3. Confirmar HTTP 429 devuelto cuando se excede

Resultado Esperado:

- Límite de tasa aplicado después del umbral
- HTTP 429 con mensaje apropiado
- Encabezado Retry-After presente

Resultado Real: APROBADO **Evidencia:** Ver `testing-evidence/18-rate-limiting.log`

Caso de Prueba 19: UI de Producción - Verificación Manual

Prioridad: Alta **Tipo:** Manual **Requisito:** UI-001

Pasos de Prueba: 1. Acceder a URL de producción vía navegador 2. Navegar por todos los módulos 3. Verificar renderizado y funcionalidad de UI

Resultado Esperado:

- Todas las páginas cargan correctamente
- Navegación funciona
- Datos se muestran correctamente

Resultado Real: PENDIENTE DE VERIFICACIÓN MANUAL **Evidencia:** A capturarse durante validación de producción

Caso de Prueba 20: API de Producción - Verificación Manual

Prioridad: Alta **Tipo:** Manual **Requisito:** UI-002

Pasos de Prueba: 1. Abrir DevTools del navegador 2. Monitorear solicitudes de red en producción 3. Verificar que todas las llamadas a API tienen éxito

Resultado Esperado:

- Todas las solicitudes de API devuelven 200/201
- Sin errores de consola
- Encabezados de autenticación presentes

Resultado Real: PENDIENTE DE VERIFICACIÓN MANUAL **Evidencia:** A capturarse durante validación de producción

Caso de Prueba 21: Compatibilidad entre Navegadores

Prioridad: Media **Tipo:** Manual **Requisito:** UI-003

Pasos de Prueba: 1. Probar UI en Chrome, Firefox, Safari, Edge 2. Verificar renderizado consistente 3. Probar todas las características interactivas

Resultado Esperado:

- Comportamiento consistente entre navegadores
- Sin bugs específicos de navegador

Resultado Real: PENDIENTE DE VERIFICACIÓN MANUAL **Evidencia:** A capturarse durante pruebas de compatibilidad

Caso de Prueba 22: Pruebas de Carga

Prioridad: Media **Tipo:** Manual **Requisito:** PERF-002

Pasos de Prueba: 1. Simular 100 usuarios concurrentes 2. Monitorear tiempos de respuesta de API 3. Verificar errores o degradación

Resultado Esperado:

- Sistema maneja carga concurrente
- Sin degradación significativa de rendimiento
- Tasa de error < 0.1%

Resultado Real: PENDIENTE DE VERIFICACIÓN MANUAL **Evidencia:** A programarse para sesión de pruebas de carga

3. Matriz de Trazabilidad

ID Prueba	Requisito	Módulo	Prioridad	Estado	Aprobado/Fallido					
CP-01	API-001	API Core	Crítica	Completo	APROBADO	CP-02	PROJ-001	Proyectos		
Alta	Completo	APROBADO	CP-03	CAT-001	Catálogo	Alta	Completo	APROBADO	CP-04	
CAT-002	Catálogo	Alta	Completo	APROBADO	CP-05	CAT-003	Catálogo	Alta	Completo	
APROBADO	CP-06	FCST-001	Pronóstico	Alta	Completo	APROBADO	CP-07	FCST-002	Pronóstico	
Alta	Completo	APROBADO	CP-08	RECON-001	Reconciliación	Media	Completo	APROBADO	CP-09	
CHG-001	Cambios	Media	Completo	APROBADO	CP-10	HO-001	Handoff	Media	Completo	
APROBADO	CP-11	AUTH-001	Seguridad	Crítica	Completo	APROBADO	CP-12	AUTH-002	Seguridad	
Crítica	Completo	APROBADO	CP-13	DB-001	Base de Datos	Alta	Completo	APROBADO	CP-14	
DB-002	Base de Datos	Alta	Completo	APROBADO	CP-15	S3-001	Almacenamiento	Media	Completo	
APROBADO	CP-16	PERF-001	Rendimiento	Media	Completo	APROBADO	CP-17	CF-001	CloudFront	
Alta	Completo	APROBADO	CP-18	SEC-001	Seguridad	Media	Completo	APROBADO	CP-19	
UI-001	UI Manual	Alta	Pendiente	PENDIENTE	CP-20	UI-002	UI Manual	Alta	Pendiente	
PENDIENTE	CP-21	UI-003	UI Manual	Media	Pendiente	PENDIENTE	CP-22	PERF-002	Rendimiento	
Media	Pendiente	PENDIENTE								

4. Paquete de Evidencia

Todos los logs de ejecución de pruebas, capturas de pantalla y documentación de apoyo se almacenan en el directorio `testing-evidence/`:

Logs de Pruebas de API

- `01-health-check.log` - Verificación de endpoint de salud
- `02-projects-list.log` - Recuperación de proyectos
- `03-rubros-catalog-*.log` - Operaciones de catálogo por proyecto
- `04-rubros-create-*.log` - Operaciones de creación
- `05-rubros-verify-*.log` - Verificación de ítems creados
- `06-forecast-6m-*.log` - Datos de pronóstico de 6 meses
- `07-forecast-12m-*.log` - Datos de pronóstico de 12 meses
- `08-reconciliation-*.log` - Operaciones de reconciliación
- `09-changes-*.log` - Seguimiento de cambios
- `10-handoff-*.log` - Documentación de handoff

Logs de Pruebas de Seguridad

- `11-auth-flow.log` - Validación de flujo de autenticación
- `12-authorization.log` - Verificaciones de autorización
- `18-rate-limiting.log` - Verificación de limitación de tasa

Logs de Pruebas de Base de Datos

- `13-dynamodb-rubros.log` - Validación de tabla rubros de DynamoDB
- `14-dynamodb-projects.log` - Validación de tabla proyectos de DynamoDB

Logs de Pruebas de Integración

- `15-s3-upload.log` - Operaciones de almacenamiento S3
- `16-lambda-performance.log` - Métricas de rendimiento
- `17-cloudfront-ui.log` - Entrega por CloudFront

Capturas de Pantalla

- Las capturas de pantalla se capturarán durante las fases de verificación manual

5. Hallazgos y Recomendaciones

5.1 Hallazgos Principales

****Hallazgos Positivos:****

1. ****Infraestructura de API Robusta**** - Todas las pruebas automatizadas de API pasan exitosamente - Tiempos de respuesta cumplen requisitos de rendimiento - Manejo de errores es comprensivo
2. ****Autenticación Segura**** - Integración de Cognito funcionando correctamente - Tokens JWT validados apropiadamente - Acceso no autorizado bloqueado apropiadamente
3. ****Integridad de Datos**** - Operaciones de DynamoDB mantienen consistencia - Sin corrupción de datos observada - Integridad referencial mantenida
4. ****Confiabilidad de Infraestructura**** - Funciones Lambda funcionan bien - CloudFront entrega activos correctamente - Operaciones de almacenamiento S3 estables

⚠️ **Áreas de Mejora:**

1. ****Cobertura de Pruebas Manuales**** - 4 casos de prueba requieren verificación manual - Ambiente de producción necesita validación completa - Pruebas entre navegadores pendientes
2. ****Pruebas de Rendimiento**** - Pruebas de carga aún no realizadas - Pruebas de estrés recomendadas - Validación de escalabilidad necesaria
3. ****Monitoreo y Observabilidad**** - Monitoreo mejorado recomendado - Umbrales de alerta deben definirse - Agregación de logs para troubleshooting

5.2 Recomendaciones

Acciones Inmediatas:

1. ****Completar Verificación Manual**** - Programar sesión de validación de producción - Realizar pruebas entre navegadores - Conducir pruebas de aceptación de usuario
2. ****Pruebas de Rendimiento**** - Ejecutar pruebas de carga con 100+ usuarios concurrentes - Identificar cuellos de botella de rendimiento - Optimizar endpoints lentos si se encuentran
3. ****Documentación**** - Actualizar documentación de usuario - Crear guías de troubleshooting -

Documentar problemas conocidos y soluciones alternativas

Mejoras a Largo Plazo:

1. ****Monitoreo Automatizado**** - Implementar dashboards de CloudWatch - Configurar alertas automatizadas - Crear pruebas de monitoreo sintético
2. ****Pruebas Continuas**** - Integrar pruebas en pipeline CI/CD - Automatizar pruebas de regresión - Agregar pruebas de contrato de API
3. ****Fortalecimiento de Seguridad**** - Auditorías de seguridad regulares - Pruebas de penetración - Escaneo de vulnerabilidades

6. Firma del Cliente

Este informe documenta los resultados de ejecución de pruebas para los módulos Finanzas SDMT + Prefectura. Las pruebas demuestran que el sistema cumple con los requisitos funcionales y está listo para despliegue en producción pendiente completar las pruebas de verificación manual.

Preparado Por:

[illegible]**Revisado Por:**

Nombre: _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ Título: Líder Técnico Fecha:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ Firma: _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Aprobado Por:

Nombre: _____ Título: Gerente de Proyecto / PMO Fecha: _____
 _____ Firma: _____

Aceptación del Cliente:

Nombre: ____________________ Organización: CVDEX / Ikusi Título:
____________________ Fecha: ____________________ Firma:

— — —

7. Apéndice

7.1 Inventario de Endpoints de API

Endpoint	Método	Autenticación Requerida	Propósito	----- ----- ----- -----
/health	GET	No	Verificación de salud del servicio	/projects GET Sí Listar todos los proyectos
/projects/{id}/rubros	GET	Sí	Obtener rubros del proyecto	/projects/{id}/rubros POST Sí Crear rubro
/projects/{id}/plan/forecast	GET	Sí	Obtener datos de pronóstico	
/projects/{id}/reconciliation/invoices	GET	Sí	Listar facturas	/projects/{id}/changes GET Sí Listar cambios
/projects/{id}/handoff	GET	Sí	Obtener estado de handoff	

7.2 Tablas DynamoDB

| Nombre de Tabla | Clave de Partición | Clave de Ordenamiento | Propósito | |-----|-----|-----|
-----|-----| | `finanzas-rubros-dev` | `projectId` | `rubroId` | Catálogo de rubros | | `finanzas-projects-dev` | `id` | - | Metadatos de proyectos | | `finanzas-rubros-taxonomia-dev` | `category` | `subcategory` |
Taxonomía de rubros | | `finanzas-audit-log-dev` | `entityId` | `timestamp` | Pista de auditoría | | `finanzas-allocation-rules-dev` | `projectId` | `ruleId` | Reglas de asignación |

7.3 Buckets S3

| Bucket | Propósito | Acceso | |-----|-----|-----| | `finanzas-documents-dev` | Almacenamiento de documentos | Privado | | `finanzas-ui-assets-dev` | Activos estáticos de UI | CloudFront |

7.4 Ejemplo de Solicitud/Respuesta de API

Solicitud: ```bash curl -X GET \ 'https://m3g6am67aj.execute-api.us-east-2.amazonaws.com/dev/health' \ -H 'Accept: application/json' ```

Respuesta: ```json { "ok": true, "service": "finanzas-sd-api", "stage": "dev", "timestamp": "2025-12-06T00:49:00.000Z" } ```

7.5 Ambiente de Ejecución de Pruebas

Versiones de Software:

- Node.js: v20.19.6
- npm: 10.8.2
- AWS CLI: 2.x
- Playwright: 1.57.0

Información del Sistema:

- SO: Ubuntu 22.04 LTS
- Memoria: 16GB
- CPU: 4 núcleos

7.6 Acrónimos y Definiciones

- ****SDMT****: Herramientas de Gestión de Entrega de Servicios
- ****AVP****: Amazon Verified Permissions
- ****JWT****: JSON Web Token
- ****CRUD****: Crear, Leer, Actualizar, Eliminar
- ****API****: Interfaz de Programación de Aplicaciones
- ****UI****: Interfaz de Usuario
- ****S3****: Simple Storage Service (AWS)
- ****DynamoDB****: Servicio de Base de Datos NoSQL de AWS
- ****Lambda****: Servicio de Computación Serverless de AWS
- ****CloudFront****: Red de Entrega de Contenido de AWS
- ****Cognito****: Servicio de Autenticación de AWS

Fin del Informe

*Este documento está preparado para propósitos de auditoría y reembolso y contiene datos precisos de

ejecución de pruebas recopilados el 6 de diciembre de 2025.*