

**UNIVERSIDAD TECNOLÓGICA DE CHILE - INACAP**

SEDE SANTIAGO CENTRO

FACULTAD DE INGENIERÍA

CARRERA DE INGENIERÍA EN INFORMÁTICA

**""Arquitectura de Servicios para la Gestión de Garantías mediante Notificaciones  
Calendarizadas ""**

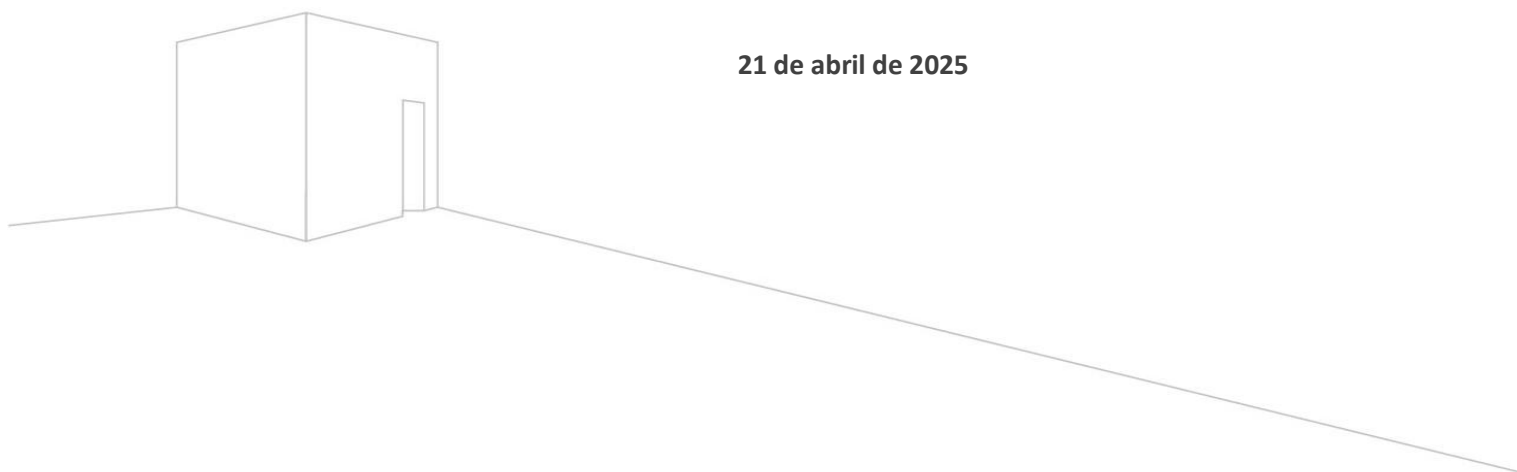
**Asignatura: Seminario de Grado – TIHI12**

**Sección: TIHI12**

**Académico guía: Ernesto Vivanco Tapia**

**Integrantes del equipo: Renzo Gomez León**

**21 de abril de 2025**



## Contenido

I.	Introducción .....	
II.	Identificación del Problema .....	
1.	Actualización y justificación del problema .....	
1.1.	Descripción de la organización .....	
2.	Justificación del problema .....	
2.1.	Relevancia del problema .....	
2.2.	Complejidad del problema .....	
III.	Definición del Proyecto .....	
1.	Marco Teórico .....	
2.	Solución tecnológica .....	
2.1.	Formulación .....	
2.2.	Alcance .....	
3.	Impacto de la solución .....	
3.1.	Proceso de negocio afectado. ....	
3.2.	Registro de Interesados. ....	
3.3.	Indicadores de gestión .....	
3.4.	Niveles de servicio. ....	
4.	Objetivos del proyecto .....	
4.1.	Objetivo General .....	
4.2.	Objetivo Específico. ....	
IV.	Metodología de Trabajo .....	
1.	Desarrollo de la solución .....	
1.1.	Duración y cronograma .....	
1.2.	Equipo de trabajo. ....	
1.3.	Plan de recursos. ....	
2.	Validación de la solución. ....	
2.1.	Validación de la funcionalidad. ....	
2.2.	Validación de la entrega de valor al negocio .....	
V.	Planificación General .....	
2.1.	Actividades y tareas .....	

2.2.	Responsables.....
2.3.	Asignación de costos.....
2.4.	Línea base de seguimiento (de acuerdo a las entregas de cada hito). ....
2.5.	Carta Gantt.....
VI.	Conclusiones .....
VII.	Referencias bibliográficas .....
VIII.	Anexos .....

## **I. Introducción**

En la actualidad, la gestión de garantías asociadas a productos tecnológicos presenta diversos desafíos derivados del uso predominante de comprobantes físicos, como boletas o facturas. Estos documentos, esenciales para hacer valer derechos como la garantía legal, tienden a extraviarse o deteriorarse con facilidad, lo que impide a los usuarios acceder a soluciones frente a fallos técnicos en dispositivos como teléfonos móviles, notebooks o computadores personales. A pesar del avance de las tecnologías digitales y la alta penetración de dispositivos móviles en Chile, no existen soluciones ampliamente adoptadas que permitan respaldar, organizar y monitorear de forma eficiente la vigencia de garantías en este tipo de bienes.

En respuesta a esta problemática, se plantea el diseño de una arquitectura basada en API REST para la gestión de garantías mediante notificaciones calendarizadas, que permita a los usuarios digitales registrar, almacenar y recibir alertas automatizadas relacionadas con el vencimiento de las garantías de sus equipos tecnológicos. Esta solución no busca únicamente ofrecer un sistema de almacenamiento, sino establecer una estructura técnica escalable, segura y programable que se diferencia de otras aplicaciones existentes por su enfoque en la trazabilidad, disponibilidad continua y automatización de alertas.

El presente informe describe el contexto del problema, la fundamentación técnica del enfoque propuesto, la planificación para su desarrollo e implementación, así como los métodos de validación considerados para asegurar su funcionalidad, desempeño e impacto real en los usuarios del entorno tecnológico.

## **II. Identificación del Problema**

## 1. Actualización y justificación del problema

### 1.1. Descripción de la organización

#### 1.1.1. Antecedentes

El presente proyecto surge como una iniciativa tecnológica orientada a resolver una problemática común en el ámbito de los productos tecnológicos: la dificultad de los usuarios para conservar, organizar y gestionar comprobantes de garantía de equipos como teléfonos móviles, notebooks o computadores personales. Estos documentos, esenciales para ejercer derechos de postventa, suelen deteriorarse, extraviarse o quedar desorganizados, lo que genera una pérdida directa de beneficios legales.

La idea nace desde la experiencia del equipo de desarrollo al identificar que, pese al uso extendido de dispositivos móviles y acceso a internet, no existen soluciones digitales especializadas que ofrezcan un respaldo confiable, automatizado y calendarizado de garantías, especialmente en el sector de bienes tecnológicos. Esta carencia representa una oportunidad para diseñar una solución técnica robusta, centrada en la arquitectura basada en API REST, que permita establecer un control eficiente del ciclo de vida de las garantías.

#### 1.1.2. Diagnóstico

El análisis preliminar del entorno digital revela que, a pesar de la alta penetración de tecnologías móviles y el avance en infraestructura digital, no existe una solución

ampliamente adoptada que permita centralizar, respaldar y monitorear garantías asociadas a dispositivos tecnológicos de alto valor. Esto provoca que muchos usuarios pierdan sus derechos por no contar con medios válidos para acreditar sus compras o condiciones de garantía.

Los informes del SERNAC reflejan que una proporción importante de los reclamos por garantías no reconocidas están vinculados a la ausencia de comprobantes legibles o a la dificultad de acreditar condiciones específicas de compra. Este contexto evidencia la necesidad de una solución que no solo almacene datos, sino que los gestione activamente mediante alertas calendarizadas y principios de trazabilidad digital, características propias de una arquitectura basada en API REST bien definida.

## 1.2. Descripción del problema

Actualmente, los usuarios de productos tecnológicos continúan recibiendo comprobantes físicos al momento de adquirir equipos como notebooks, teléfonos o accesorios electrónicos. Estos documentos son esenciales para validar las condiciones de garantía, sin embargo, al estar en formato papel, se vuelven vulnerables al deterioro, pérdida o ilegibilidad con el paso del tiempo. La imposibilidad de presentar estos respaldos implica, en numerosos casos, la pérdida de derechos establecidos por la ley.

A pesar de los avances en digitalización, gran parte de los usuarios carece de herramientas específicas que les permitan organizar, almacenar y gestionar de manera eficiente los plazos y condiciones de sus garantías. Esta situación genera frustración,

desprotección y dependencia de medios físicos poco confiables. En este contexto, la arquitectura basada en API REST calendarizadas representa una solución técnica viable que permite automatizar la gestión de garantías, reducir riesgos de pérdida de información, y fortalecer el ejercicio de los derechos del consumidor en el ámbito tecnológico.

## 2. Justificación del problema

### 2.1. Relevancia del problema.

La ausencia de soluciones especializadas que permitan gestionar eficientemente las garantías de productos tecnológicos afecta directamente a los usuarios, impidiéndoles ejercer derechos fundamentales como el cambio, la reparación o la devolución de equipos en mal estado. Esta situación no solo perjudica al consumidor final, sino que también genera ineficiencias en los procesos administrativos de empresas proveedoras y genera una sobrecarga en organismos reguladores debido a reclamos mal fundamentados o sin respaldo documental válido.

El desarrollo de una arquitectura basada en API REST calendarizadas para la gestión de garantías no solo responde a esta necesidad urgente, sino que permite establecer una solución técnica robusta, automatizada y escalable. Esta propuesta representa un avance significativo en el proceso de transformación digital, al ofrecer mecanismos concretos para asegurar la trazabilidad de las garantías, reducir la dependencia de documentos físicos y fomentar un uso más consciente y estratégico de la tecnología. Asimismo, promueve el empoderamiento del usuario mediante alertas oportunas que facilitan el ejercicio informado de sus derechos.

## 2.2. Complejidad del problema.

Aunque la solución propuesta puede parecer sencilla desde la perspectiva del usuario final, su implementación requiere la integración eficiente de múltiples componentes tecnológicos. Esto incluye el diseño de una arquitectura basada en API REST compuesta por módulos de notificación calendarizada, almacenamiento seguro en la nube, autenticación de usuarios, encriptación de datos, gestión de tiempos de garantía y monitoreo del estado de cada documento registrado.

A ello se suma la necesidad de garantizar una experiencia de usuario clara y confiable, considerando aspectos como la interfaz de interacción, la interoperabilidad con futuras soluciones de validación comercial, la capacidad de escalar el sistema según la demanda, y la continuidad operativa mediante servidores espejo y respaldos automáticos. Todo esto exige una planificación metodológica rigurosa, que combine ingeniería de software, buenas prácticas de diseño de sistemas distribuidos, y un enfoque centrado en la sostenibilidad tecnológica.

### III. Definición del Proyecto.

#### 1. Marco Teórico



El desarrollo del proyecto se sustenta en diversos pilares tecnológicos que conforman la base de una arquitectura moderna, segura y escalable orientada a la gestión eficiente de garantías en productos tecnológicos.

#### Arquitectura basada en API REST:

Este enfoque permite construir sistemas modulares donde cada componente cumple una función específica y puede comunicarse con los demás mediante interfaces bien definidas. En el contexto de este proyecto, la arquitectura basada en API REST permite gestionar alertas calendarizadas de garantías, almacenamiento seguro, autenticación de usuarios y notificaciones automatizadas de forma desacoplada y escalable.

#### Aplicaciones móviles como interfaz:

Las aplicaciones móviles no constituyen el eje central de la solución, pero sí cumplen un rol importante como canal de acceso. Permiten a los usuarios registrar productos, revisar alertas y visualizar el estado de sus garantías de manera intuitiva y accesible desde sus dispositivos. La Mobile Marketing Association (2011) destaca el valor de las apps para facilitar la gestión personal de información digital.

#### Seguridad informática:

La integridad, confidencialidad y disponibilidad de la información son fundamentales. Según Stolk (2013), la seguridad informática incluye prácticas como el cifrado de datos en tránsito y en reposo, la autenticación de múltiples factores y la realización de respaldos automáticos, todas ellas aplicadas en este proyecto para proteger los documentos de garantía registrados.

#### Cloud computing:

El modelo de servicios en la nube proporciona la infraestructura necesaria para almacenar documentos de forma redundante, con alta disponibilidad y acceso remoto. Según el NIST (2011), esta tecnología permite ofrecer servicios escalables bajo demanda, lo que resulta esencial para garantizar continuidad y recuperación ante fallos.

#### Gestión de garantías en productos tecnológicos:

En el contexto chileno, el ejercicio de la garantía legal depende de la presentación de comprobantes válidos. En productos tecnológicos como notebooks, smartphones o PCs, esta necesidad se vuelve crítica debido al valor y uso intensivo de dichos dispositivos. El sistema propuesto busca automatizar la gestión de estos respaldos y sus fechas clave, mejorando la trazabilidad y empoderando al usuario mediante recordatorios y control digital.

## 2. Solución tecnológica

### 2.1. Formulación

La solución propuesta consiste en una arquitectura basada en API REST calendarizadas orientada a la gestión de garantías de productos tecnológicos, tales como teléfonos móviles, notebooks o PCs. Esta arquitectura está compuesta por diversos módulos funcionales, diseñados para operar de forma desacoplada, segura y escalable. Entre sus capacidades principales se incluyen:

- Registro estructurado de productos tecnológicos y sus respectivas garantías.
- Almacenamiento digital seguro de boletas o comprobantes de compra.
- Programación de alertas calendarizadas antes del vencimiento de la garantía.
- Trazabilidad del ciclo de vida de cada garantía.

La implementación considera tecnologías como servicios REST en PHP, bases de datos MySQL, almacenamiento en la nube, cifrado de datos en tránsito y en reposo, servidores espejo para continuidad operativa, y mecanismos de autenticación segura.

## 2.2. Alcance

El alcance del proyecto incluye:

- Diseño e implementación de la arquitectura basada en API REST calendarizadas.
- Desarrollo de interfaces web y móvil básicas para registro y visualización de garantías.
- Integración de módulos de notificación y respaldo automático.
- Pruebas funcionales y de seguridad con usuarios en entorno controlado.
- Validación del sistema mediante indicadores de rendimiento y encuestas dirigidas.

## 3. Impacto de la solución.

### 3.1. Proceso de negocio afectado.

La arquitectura impacta directamente en la forma en que los usuarios gestionan sus garantías digitales, abordando procesos clave como:

- Registro y seguimiento digital de garantías.
- Disminución de la pérdida de comprobantes físicos.
- Prevención de vencimientos mediante notificaciones oportunas.
- Educación del consumidor en prácticas de respaldo y control digital.

### 3.2. Registro de Interesados.

- Usuarios finales: Personas que adquieren productos tecnológicos y requieren trazabilidad de sus garantías.
- Empresas proveedoras de tecnología: Potenciales integradores o validadores de sistemas de garantía digital.
- Entidades gubernamentales: SERNAC u organismos similares interesados en fortalecer el derecho del consumidor.
- Proveedores tecnológicos: Infraestructura cloud, servicios de autenticación, hosting y seguridad.
- Equipo de desarrollo: Encargado del diseño modular, programación, pruebas y mantenimiento.

### 3.3. Indicadores de gestión.

1. Cantidad de garantías registradas mensualmente.
2. Porcentaje de usuarios que configuran alertas personalizadas.
3. Número de notificaciones emitidas antes del vencimiento.
4. Tiempo promedio de respuesta de los servicios API (< 3 segundos).
5. Nivel de satisfacción de usuarios técnicos (>80%).

### 3.4. Niveles de servicio.

Elemento	Descripción
<b>Nombre del servicio</b>	Arquitectura de Servicios Calendarizados para la Gestión de Garantías Tecnológicas
<b>Gestor del servicio</b>	Líder de proyecto – Renzo Gomez
<b>Cliente</b>	Usuarios particulares con acceso a internet y propietarios de dispositivos tecnológicos (PCs, notebooks, móviles)
<b>Resultado esperado</b>	Gestión digital, trazable y automatizada de garantías, con alertas programadas
<b>Procesos apoyados</b>	Control de vigencia de garantías, respaldo documental, trazabilidad y alertas
<b>Criticalidad</b>	Alta – Servicio esencial para asegurar el ejercicio de derechos postventa en productos tecnológicos
<b>Activos esenciales</b>	Servidores cloud, base de datos, servicios API, interfaz web/móvil, sistema de notificaciones
<b>Horario disponible</b>	24/7 (excepto mantenciones programadas)
<b>Período de mantención</b>	Primer lunes de cada mes, entre 02:00 y 04:00 AM
<b>Disponibilidad esperada</b>	99.7% mensual
<b>Tiempo respuesta app</b>	< 3 segundos
<b>Continuidad de servicio</b>	RTO: 2 horas / RPO: 15 minutos (uso de servidores espejo y respaldos automáticos)
<b>Responsabilidades</b>	El proveedor mantiene la plataforma operativa y segura. El cliente debe usar contraseñas robustas y respetar los términos de uso.
<b>Seguridad TI</b>	Cifrado de datos en tránsito y reposo, doble autenticación, respaldos diarios automáticos

## 4. Objetivos del proyecto

### 4.1. Objetivo General.

Diseñar e implementar una arquitectura basada en API REST calendarizadas que permita a los usuarios gestionar de forma eficiente y automatizada las garantías asociadas a productos tecnológicos, mediante el registro estructurado de comprobantes, la generación

de alertas oportunas y el acceso seguro a la información, con el fin de fortalecer el ejercicio de derechos del consumidor y promover una transformación digital sostenible.

#### 4.2. Objetivo Específico.

- Diseñar una arquitectura modular que contemple servicios independientes para almacenamiento, notificación, autenticación y control de garantías.
- Implementar mecanismos de respaldo seguro en la nube para asegurar disponibilidad y recuperación ante fallos.
- Desarrollar un sistema de notificaciones calendarizadas que permita alertar al usuario sobre eventos críticos como el vencimiento de una garantía.
- Validar el funcionamiento y desempeño de la arquitectura mediante indicadores técnicos (KPIs) y retroalimentación de usuarios en pruebas controladas.

## IV. Metodología de Trabajo

### 1. Desarrollo de la solución

#### 1.1. Duración y cronograma.

La duración estimada del proyecto es de 13 semanas, distribuidas en cinco fases principales:

Inicio y análisis	Revisión de la problemática, definición del alcance, levantamiento de requerimientos técnicos, planificación general	2 semanas
-------------------	--	-----------

Diseño de arquitectura	Modelado de la arquitectura de servicios, definición de módulos (notificaciones, respaldo, seguridad), planificación de integración	2 semanas
Desarrollo de servicios	Implementación del backend (servicios de registro, notificación, almacenamiento, autenticación), pruebas unitarias por módulo	6 semanas (3 sprints)
Integración y pruebas	Integración de servicios, pruebas funcionales, validación de seguridad, monitoreo de desempeño (KPIs)	2 semanas
Validación y cierre	Pruebas con usuarios en entorno controlado, aplicación de encuestas, análisis de resultados y entrega final	1 semana

## 1.2. Equipo de trabajo.

Integrante	Rol en el equipo	Responsabilidades principales
Renzo Gomez	Líder de Proyecto	Coordinación general, relación con el docente guía, presentación del proyecto
Renzo Gomez	Arquitecto de Servicios	Diseño de la arquitectura del sistema, definición de módulos funcionales, configuración de entorno de desarrollo.
Renzo Gomez	Desarrollador Backend	Programación de servicios (notificación, almacenamiento, autenticación), integración con base de datos y nube.

Renzo Gomez	Documentador / QA	Redacción del informe, control de versiones, ejecución de pruebas funcionales, validación de KPIs técnicos.
-------------	-------------------	---

### 1.3. Plan de recursos.

Recurso	Descripción
Personal	1-3 integrantes del equipo de desarrollo
Equipamiento	Computadora personal con acceso a entorno de desarrollo backend (Visual Studio Code, Postman), conexión a internet estable.
Servicios contratados	Dominio y hosting web (2 servidores: principal y espejo)
Software	Visual Studio Code, React Native (CLI o Expo), Figma (prototipado), Trello (gestión ágil), GitHub (control de versiones)
Presupuesto estimado	CLP \$169.360

Elemento	Valor	Tipo de costo
Dominio	\$10.000 CLP	Fijo
Hosting (2 servidores)	\$80.000 CLP	Fijo
Publicación en Google Play	\$16.000 CLP	Fijo (USD 25)
Otros gastos operativos	\$63.360 CLP	Variable
Total	\$169.360 CLP	

## 2. Validación de la solución.

### 2.1. Validación de la funcionalidad.

- **Pruebas funcionales:** Verificar que cada componente del sistema (carga de documentos, clasificación, notificaciones) funcione correctamente según los requerimientos.
- **Usabilidad:** Se aplicarán pruebas con usuarios (preferentemente estudiantes o trabajadores familiarizados con herramientas digitales) para identificar mejoras en la interfaz.



- Seguridad: Se revisarán aspectos como autenticación, cifrado de datos, manejo de sesiones y recuperación de información mediante backups.

## 2.2. Validación de la entrega de valor al negocio.

- Encuestas de satisfacción: Aplicadas a usuarios beta mediante formularios digitales (Google Forms) para medir experiencia y percepción del sistema.
- Indicadores clave (KPI):
  - Porcentaje de funcionalidades implementadas exitosamente.
  - Tiempo de carga de la aplicación (< 3 segundos).
  - Frecuencia de uso de la funcionalidad de notificación.
  - Nivel de satisfacción del usuario (>80%).
- Juicio experto: Revisión del proyecto por parte del profesor guía y, eventualmente, profesionales externos.

## Evaluación del Éxito

- El éxito del proyecto se evaluará en función de:
  - El cumplimiento de los objetivos específicos planteados.
  - La recepción positiva de los usuarios y stakeholders.
  - El funcionamiento estable del sistema en pruebas reales.
  - La adherencia al cronograma y la correcta entrega de los productos definidos.

## V. Planificación General

### 1. Planificación temporal

#### 2.3. Actividades y tareas

Distribución de actividades por fases

Semana	Actividad	Responsable	Costo asociado*
1	Formación del equipo, asignación de roles	Todo el equipo	-
2	Levantamiento de requerimientos, análisis del problema	Líder de proyecto	-
3	Redacción del informe preliminar (Parte I), exposición inicial	Documentador	-
4-5	Diseño de interfaz (mockups, flujos), validación con usuarios	Todo el equipo	-
6-8	Desarrollo iterativo Sprint 1 y 2 (web y app móvil)	Desarrollador	\$40.000 CLP
9	Implementación de sistema de notificaciones y subida de archivos	Desarrollador	\$20.000 CLP
10	Pruebas funcionales y de seguridad	Documentador	-
11	Validación con usuarios finales (encuestas)	Líder de proyecto	-
12	Análisis de resultados y ajustes finales	Todo el equipo	-
13	Presentación final y entrega del informe completo	Líder de proyecto	-

#### 2.4. Responsables

Actividad	Responsable Principal
Análisis y planificación	Líder de proyecto
Diseño de interfaz y prototipos	Todo el equipo
Desarrollo técnico (app y web)	Desarrollador principal
Redacción de informe y documentación	Documentador / QA
Pruebas y validación	Documentador / QA
Presentación final	Líder de proyecto

## 2.5. Asignación de costos

Elemento	Valor	Tipo de costo
Dominio	\$10.000 CLP	Fijo
Hosting (2 servidores)	\$80.000 CLP	Fijo
Publicación en Google Play	\$16.000 CLP	Fijo (USD 25)
Otros gastos operativos	\$63.360 CLP	Variable
Total	\$169.360 CLP	

## 2.6. Línea base de seguimiento (de acuerdo a las entregas de cada hito).

Hito	Semana de Entrega	Entregable
Formación de equipo y planificación	Semana 1	Documento de planificación inicial
Análisis y diseño	Semana 3	Documento de requisitos técnicos y diseño de arquitectura
Desarrollo iterativo (Sprint 1 y 2)	Semana 8	Servicios funcionales: registro, almacenamiento, notificaciones
Implementación y pruebas	Semana 10	Sistema integrado con pruebas técnicas y de seguridad superadas
Validación con usuarios y ajustes	Semana 12	Resultados de pruebas controladas y encuestas aplicadas
Presentación final	Semana 13	Informe final, demo funcional y presentación del proyecto

## 2.7. Carta Gantt

La planificación del proyecto Recooba se organiza en base a las etapas del ciclo de desarrollo ágil y a los objetivos planteados. Las actividades se distribuyen en un periodo estimado de 13 semanas, que representa un semestre académico. A continuación, se presenta la descripción general del plan de trabajo:

Distribución de actividades por fases

Semana	Actividad	Responsable	Costo asociado*
1	Formación del equipo, asignación de roles	Todo el equipo	-
2	Levantamiento de requerimientos, análisis del problema	Líder de proyecto	-
3	Diseño de arquitectura, definición de módulos y flujos de servicios	Documentador	-
4-5	Implementación de servicios: registro de productos, almacenamiento, alertas	Todo el equipo	-
6-8	Desarrollo de funcionalidades secundarias, pruebas unitarias por módulo	Desarrollador	\$40.000 CLP
9	Integración de servicios, pruebas funcionales y de seguridad	Desarrollador	\$20.000 CLP
10	Pruebas con usuarios en entorno controlado	Documentador	-
11	Aplicación de encuestas, recolección y análisis de KPIs	Líder de proyecto	-
12	Ajustes finales, validación del sistema, revisión del informe	Todo el equipo	-
13	Presentación final y entrega oficial del proyecto	Líder de proyecto	-

## VI. Conclusiones

La arquitectura basada en API REST calendarizadas para la gestión de garantías tecnológicas surge a partir de una necesidad real y cotidiana: la dificultad que enfrentan los usuarios al intentar conservar y acceder a los comprobantes de garantía de sus dispositivos electrónicos, justo cuando más los necesitan. Esta situación, común en la vida moderna, fue el punto de partida para diseñar una solución que ofrezca control, trazabilidad y respaldo automatizado.

A lo largo del desarrollo del proyecto se definió una propuesta tecnológica concreta, basada en principios de modularidad, escalabilidad y automatización. Esta arquitectura no busca simplemente almacenar documentos, sino ofrecer una plataforma sólida que gestione de forma inteligente la vida útil de las garantías asociadas a dispositivos como notebooks, teléfonos y PCs.

El proyecto también refleja un compromiso con la transformación digital responsable, fomentando el uso eficiente de los recursos, la reducción del papel físico, y el empoderamiento de los usuarios en el ejercicio de sus derechos. Al integrar servicios como notificaciones calendarizadas, respaldo seguro en la nube y trazabilidad digital, se aporta valor tanto desde lo técnico como desde lo humano.

Esta propuesta abre la puerta a nuevas posibilidades: futuras integraciones con proveedores tecnológicos, validación automática de garantías, y la expansión del sistema hacia otros dominios donde la gestión calendarizada de eventos críticos pueda marcar una diferencia. Más que una aplicación, este trabajo representa una visión técnica con impacto real, pensada desde la ingeniería y con empatía hacia los usuarios finales.

## **VII. Referencias bibliográficas**

Aalbers, H. (2017). \*Una introducción a Cloud Computing\*. Madrid.

Association, M. M. (2011). \*Libro Blanco de Apps\*. Obtenido de <http://www.mmaspain.com/wp-content/uploads/2015/09/Libro-Blanco-Apps.pdf>

GFK Chile. (s.f.). \*Adimark\*. Obtenido de <http://www.adimark.cl>

Gobierno de Chile. (2016, marzo). \*Resumen 7ma encuesta de uso y acceso\*. Subtel. [http://www.subtel.gob.cl/wp-content/uploads/2016/05/Resumen\\_7ma\\_encuesta\\_de\\_uso\\_y\\_acceso.pdf](http://www.subtel.gob.cl/wp-content/uploads/2016/05/Resumen_7ma_encuesta_de_uso_y_acceso.pdf)

Gobierno del Estado de Tabasco. (s.f.). \*Manual de Seguridad Informática Básica\*. <http://dgtic.tabasco.gob.mx/sites/all/files/vol/dgtic.tabasco.gob.mx/fi/Manual%20de%20Seguridad%20Informatica%20Basica.pdf>

Gonzales, F. (2017, agosto 22). \*5 tipos de papel para imprimir\*. Natura Print.  
<https://imprentaonline-naturaprint.com/5-tipos-de-papel-imprimir/>

Grupo ASSA. (2015). \*Latam 4.0 - dBT in the Value Chain\*.  
<http://www.grupoassa.com/informes/Latam-40-dBT-in-the-Value-Chain.pdf>

Instituto Nacional de Estadísticas. (2016). \*Síntesis Censo\*.  
[http://www.censo2017.cl/wp-content/uploads/2017/01/pc2016\\_region-comuna-13122016.pdf](http://www.censo2017.cl/wp-content/uploads/2017/01/pc2016_region-comuna-13122016.pdf)

Instituto Nacional de Estándares y Tecnología (NIST). (2011). \*The NIST Definition of Cloud Computing\*. <https://csrc.nist.gov/publications/detail/sp/800-145/final>

López, A. G., Caycedo Espinel, C. G., & Madrinán Rivera, R. E. (s.f.). \*Comentarios al nuevo estatus del consumidor\*. Legis.

Merodio, J. (2010). \*Marketing en redes sociales\*. Madrid: Gestión 2000.

Olachea, O. (2014, septiembre 10). \*Conoce el significado de 10 colores y a qué público le hablan\*. Paredro. <https://www.paredro.com/conoce-el-significado-de-10-colores-y-a-que-publico-le-hablan/>

Real Academia Española. (s.f.). \*Diccionario de la lengua española\*. <http://dle.rae.es>

Revista Forbes. (2013, enero 17). \*10 habilidades más solicitadas en 2013\*. Universia Noticias. <http://noticias.universia.es/practicas-empleo/noticia/2013/01/17/994359/10-habilidades-competencias-solicitadas-2013.html>

SERNAC. (s.f.). \*Guía de alcances jurídicos para ejercer la garantía legal\*.  
<https://www.sernac.cl/wp-content/uploads/2012/12/guia-de-alcances-juridicos-para-ejercer-la-garantia-legal-sernac.pdf>

SERNAC. (2017). \*Comportamiento de Respuesta de Proveedores\*.  
[https://www.sernac.cl/wp-content/uploads/2017/09/2017.09.26-Comportamiento-de-Respuesta-de-Proveedores-2017Finalcon\\_acciones.pdf](https://www.sernac.cl/wp-content/uploads/2017/09/2017.09.26-Comportamiento-de-Respuesta-de-Proveedores-2017Finalcon_acciones.pdf)

SoloMarketing.es. (s.f.). \*Guía Social Media\*. <https://www.solomarketing.es>

Stolk, A. (2013). \*Técnicas de seguridad informática con software libre\*. Mérida: ESLARED.

Viñals, J. T. (2012). \*Del Cloud Computing al Big Data\*. Barcelona: Eureka Media SL.

## **VIII. Anexos**