**Universidad Nacional Mayor de San Marcos**

****

UNIVERSIDAD DEL PERÚ - DECANA DE AMÉRICA

**FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

**EAP:** INGENIERÍA DE SOFTWARE

**CURSO:** DISEÑO DE SOFTWARE

**DOCUMENTO DE ARQUITECTURA**

Sistema de Registro de Vehículos

**EQUIPO:**

RAMOS GERONIMO, Daniel Dante Mathew

MONZON INFANTE, Paolo Cesar

QUIROZ ARDILES, Sergio Daniel

LEÓN ROBLES, Illary Marcelo

RODAS PANAIFO, Jordan André

MARCELO MACAVILCA, Luis

RAMOS SIFUENTES, Marcelo Alfonso

**Lima, Perú**

**2023**

1. **Descripción general**

Este documento proporciona una visión general de la arquitectura del módulo de reclamos, solicitudes y quejas, que se desarrollará para satisfacer las necesidades de nuestra organización en la gestión eficiente de las interacciones de los clientes y la resolución de problemas. Este módulo permitirá a los usuarios registrar y proporcionar un seguimiento adecuado a los reclamos, solicitudes o quejas (de ahora en adelante RSQ para abreviar) de manera efectiva. Los principales objetivos de este documento son los siguientes:

* Guiar el desarrollo, facilitar la comprensión de la arquitectura y servir como referencia para el mantenimiento y futuras mejoras.
* Diseñar una arquitectura que permita el procesamiento eficiente de reclamos, quejas y solicitudes de los clientes.
* Garantizar la escalabilidad y la capacidad de respuesta del sistema a medida que crece la base de clientes.
* Integrarse con sistemas y servicios existentes en la organización cuando sea necesario.

1. **Drivers de la arquitectura**
   1. **Requerimientos funcionales**

| **RF - 01** | **Registro de RSQ**  Permitir el registro de RSQ mediante el llenado de campos que el usuario completará con la información que el cliente le proporcionará. El registro incluye la clasificación entre reclamo, solicitud y queja. | Operador | **Alta** |
| --- | --- | --- | --- |
| Media |
| Baja |
| **RF - 02** | **Edición de RSQ en el sistema**  El sistema debe permitir a los trabajadores del área modificar los RSQ ingresados, con el fin de especificar información pasada o añadir datos adicionales. Esto incluye cambiar el estado del RSQ (pendiente, derivado, concluido) para tener un correcto seguimiento del proceso. | Operador | **Alta** |
| Media |
| Baja |
| **RF - 03** | **Listar RSQ**  Mostrar todos los RSQ del sistema en una lista ordenada. | Operador | **Alta** |
| Media |
| Baja |
| **RF - 04** | **Detallar RSQ**  Permitir al usuario ver los detalles sobre estos, el producto o servicio involucrado, y cualquier otra información relevante. Además de permitir al usuario filtrar los reclamos según ciertos parámetros. | Operador | **Alta** |
| Media |
| Baja |
| **RF - 05** | **Filtrar la búsqueda de RSQ**  Permitir al operador especificar los parámetros de búsqueda para filtrar los RSQ mostrados en pantalla. | Operador | **Alta** |
| Media |
| Baja |
| **RF - 06** | **Atender RSQ**  Brindar la opción de atender y resolver el RSQ seleccionado de la lista de búsqueda, proporcionando alternativas de solución. | Operador | **Alta** |
| Media |
| Baja |
| **RF - 07** | **Derivar RSQ**  Permitir a los operadores del sistema, tras haber evaluado las solicitudes, derivarlas al área administrativa respectiva para que se encargue de continuar con su resolución. | Operador | **Alta** |
| Media |
| Baja |
| **RF - 08** | **Notificar a los clientes del resultado de los RSQ**  Permitir notificar a los clientes de la finalización de la atención del RSQ mediante un correo electrónico, en el que se incluya información detallada sobre las consideraciones más importantes sobre el asunto tratado. | Operador | **Alta** |
| Media |
| Baja |
| **RF - 09** | **Generar formato de impresión de resolución**  El sistema debe ayudar a generar un formato de impresión estandarizado que se pueda mandar a los clientes. | Operador | **Alta** |
| Media |
| Baja |
| **RF - 10** | **Completar campos de resolución de RSQ**  Permitir incluir información sobre la resolución del RSQ, llenando campos en los que se puede redactar y precisar los aspectos evaluados para justificar el resultado y los procesos realizados. | Operador | **Alta** |
| Media |
| Baja |
| **RF - 11** | **Archivar RSQ**  Permitir a los operadores, tras haber concluido el proceso de atención, archivar los RSQ para mantener un orden apropiado. | Operador | **Alta** |
| Media |
| Baja |

* 1. **Requerimientos no funcionales**

**RNF - 001: Rendimiento**

* + **Tiempo de respuesta:** El sistema proporciona tiempos de respuesta rápidos para asegurar la satisfacción del cliente.
  + **Escalabilidad:** El rendimiento se mantendrá aun cuando la capacidad de gestionar sufra un aumento en la carga de trabajo
  + **Disponibilidad:** El sistema estará disponible durante un alto porcentaje del tiempo, con un tiempo mínimo de inactividad programado.

**RNF - 002: Seguridad**

* + **Autenticación y autorización:** Debe garantizar la autenticación segura del usuario y limitar el acceso solo a usuarios autorizados.
  + **Cifrado:** Los datos confidenciales deben protegerse mediante cifrado durante su transmisión y almacenamiento.
  + **Auditoría:** Crear y mantener registros de auditoría para realizar un seguimiento de todas las actividades relacionadas con las quejas y la información de los clientes.

**RNF - 003: Fiabilidad**

* + **Recuperación de errores:** Debe poder recuperarse rápidamente de errores inesperados sin perder datos críticos.
  + **Resistencia a errores:** Debe ser resistente a errores para mantener la continuidad del servicio incluso en condiciones adversas.

**RNF - 004: Usabilidad**

* + **Diseño intuitivo:** La interfaz de usuario debe ser sencilla de utilizar y comprensible para los usuarios finales.
  + **Accesibilidad:** Los estándares de accesibilidad se deben cumplir para cualquier persona.

**RNF - 005: Interoperabilidad**

* + **Integración:** Debe poder integrarse con otros sistemas y servicios de telecomunicaciones, como bases de datos de clientes y sistemas de facturación.
  + **Compatibilidad:** Debe ser compatible con una variedad de plataformas y navegadores web.

**RNF - 006: Mantenimiento y escalabilidad**

* + **Facilidad de mantenimiento:** El sistema debe ser sencillo de mantener y actualizar sin causar interrupciones significativas en el servicio.
  + **Escalabilidad:** Debe ser escalable para adaptarse al crecimiento futuro y a nuevas funciones.

**RNF - 007: Cumplimiento normativo**

* + **Cumplimiento legal:** Se deben seguir todas las regulaciones y leyes relevantes relacionadas con la privacidad de datos y las comunicaciones.
  + **Estándares y regulaciones de la industria:** Debe cumplir con los estándares y regulaciones de la industria.

**RNF - 008: Rendimiento de la base de datos**

* + **Consultas eficientes:** Las consultas a bases de datos deben ser eficientes y no desperdiciar recursos.
  + **Recuperación de datos y respaldo:** Deben existir procedimientos adecuados de recuperación y respaldo de datos.

**RNF - 009: Gestión de la carga de trabajo**

* + **Planificación de capacidad:** Debe ser capaz de planificar y gestionar eficazmente la carga de trabajo para evitar cuellos de botella y garantizar un rendimiento óptimo.
  1. **Restricciones**

Dependencia del módulo de ventas, servicio de envío de correos y clientes para avanzar el código.

**Alcance**

**Lenguajes**

El equipo de trabajo del módulo de Reclamos, Solicitudes y Quejas se rige bajo los acuerdos que se han realizado con los desarrolladores de los demás módulos del sistema. En base a ello:

* El frontend del software se realizará con ReactJS (JavaScript)
* El módulo debe poseer una base de datos propia
* Para el manejo de la base de datos se usará PostgreSQL

Se utilizarán Python, JavaScript y ProgreSQL

**Personas**

Se cuenta con un equipo de trabajo de 4 personas para desarrollar el módulo de reclamos. Los roles de cada uno de los integrantes se especifica a continuación:

| **Módulo** | Reclamos, solicitudes y quejas |
| --- | --- |
| **Scrum master** | Guia Muñoz, Wilfredo |
| **Product owner** | León Robles, Illary Marcelo |
| **Equipo de desarrollo** | Álvarez More, Diego (DBA) |
| Ancaya Martínez, Benjamín (Diseñador UI/UX) |
| Guia Muñoz, Wilfredo (Tester) |
| Guzmán Neyra, Paulo Renato (Programador Backend) |
| León Robles, Illary Marcelo (Programador Frontend) |
| **Fecha de inicio** | 02/09/23 |
| **Fecha final** | 25/11/23 |

**Equipamiento o materiales**

El equipo necesario para desarrollar el módulo consiste en ordenadores, que se usarán para generar la documentación, programar los componentes y realizar las pruebas del software.

**Tiempo**

En cuanto al tiempo, se debe considerar las siguientes restricciones:

* **Programación:** Se cuenta con un plazo definido para la finalización de cada fase del proyecto. Para ello, el cronograma de actividades especifica la fecha de inicio y fin de cada actividad a desarrollar por cada iteración.
* **Seguimiento:** Una vez que el proyecto está en marcha, se debe observar tendencias y efectos en planes futuros, y comunicar estos hallazgos a todas los integrantes del proyecto, sobre todo los implicados en la realización de las partes observadas.
* **Control:** El equipo debe comunicar los resultados de cada fase del proyecto y avanzar en consecuencia. Se debe analizar los factores que contribuyen a ese resultado positivo para que pueda continuar y reproducirse. Si ha habido un descarrilamiento, el equipo debe saber cómo ocurrió y por qué, y tomar medidas para corregirlo para acciones futuras. Con este fin, el equipo de gestión tiene fechas definidas para realizar el control de las etapas pasadas y la evaluación de resultados.

* 1. **Preocupaciones/Riesgos**

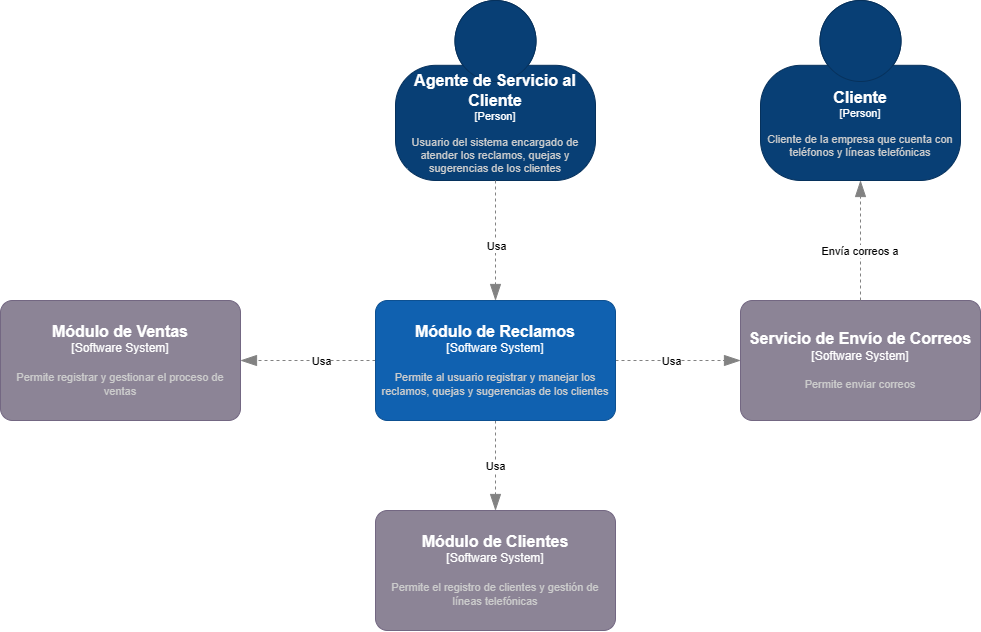
1. **Seguridad de datos y cumplimiento normativo:** Proteger la información y cumplir con las regulaciones para evitar problemas legales.
2. **Evolución, errores y fallas del sistema:** Asegurar que el sistema funcione rápidamente y sin errores.
3. **Interoperabilidad y escalabilidad:** Asegurar que el sistema se conecte con otros y crezca según sea necesario.
4. **Mantenimiento y costes inesperados:** Evitar costes inesperados y garantizar un mantenimiento óptimo y adecuado.
5. **Usabilidad y aceptación del usuario:** Garantizar que el sistema sea de fácil uso por parte de los usuarios.
6. **Reputación y ciberataques:** Proteger la imagen de la empresa y prevenir ciberataques.

Para mitigar estos riesgos, se deben implementar medidas de seguridad sólidas, se deben realizar pruebas exhaustivas e integrales.

1. **Diagrama de Contexto**

* Nombre del Sistema: Módulo de Reclamos
* Objetivo del Sistema: El Módulo de Reclamos es una aplicación diseñada para gestionar y registrar reclamos, solicitudes y quejas de los clientes.
* Usuarios:
  + Clientes
  + Asesor de RQS
* Tecnologías Utilizadas:
  + Frontend: React (JavaScript)
  + Backend: FastAPI (Python)
  + Base de Datos: PostgreSQL
* Resumen de Arquitectura:
  + Módulo Reclamos: Arquitectura Multinivel
  + Módulo Reclamos - Frontend: Arquitectura Basada en Componentes
  + Módulo Reclamos - Backend: Arquitectura en Capas

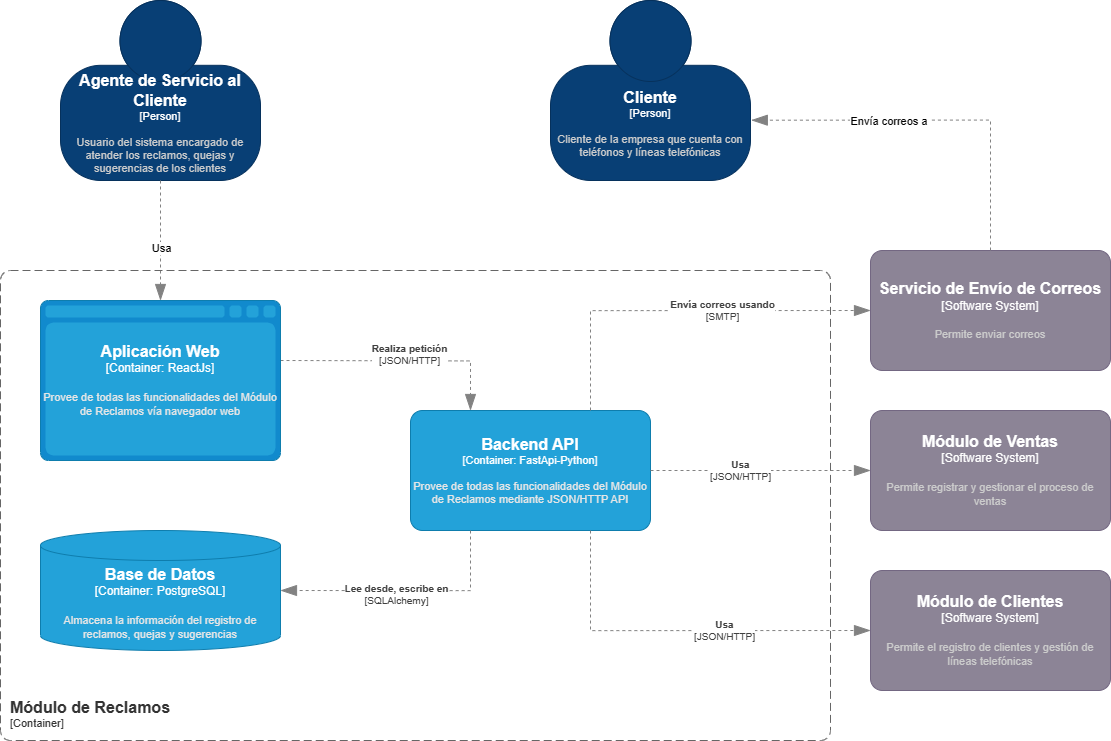
A continuación, se presenta el diagrama de contexto que ilustra la posición y las interacciones del Módulo de Reclamos en su entorno.



1. **Arquitectura General del Sistema**

El sistema Módulo de Reclamos sigue una arquitectura multinivel que organiza sus componentes en tres niveles distintos. Cada nivel tiene responsabilidades específicas y se comunica con los demás para proporcionar una funcionalidad completa y eficiente.

* Nivel de Presentación (Frontend): En este nivel, se encuentra la interfaz de usuario (UI) y la lógica relacionada con la presentación y la interacción del usuario. El frontend se encarga de proporcionar una experiencia de usuario atractiva e intuitiva.
* Nivel de Aplicación (Backend): El backend se encuentra en este nivel y comprende la lógica de negocio y la gestión de datos. Aquí se procesan las solicitudes del cliente, se ejecuta la lógica de la aplicación y se gestiona la interacción con la base de datos.
* Nivel de Datos (Base de Datos): La base de datos se sitúa en este nivel y se encarga de almacenar y recuperar información de la aplicación. Almacena los datos necesarios para el funcionamiento del sistema.

A continuación, se presenta un diagrama que ilustra la arquitectura multinivel del Módulo de Reclamos. Este diagrama mostrará cómo los componentes de cada nivel se relacionan y cooperan para brindar funcionalidades al usuario final.

1. **Diagrama de componentes - Backend API**

