

**[Sistema aduana]
(DAS) Documento Arquitectura de Software
Versión 1.3.0**

Identificación de Documento

Identificación	Sistema Aduanas Chile - Argentina
Proyecto	Automatización sistema aduanero
Versión	1.3

Documento mantenido por	
Fecha de última revisión	
Fecha de próxima revisión	

Documento aprobado por	
Fecha de última aprobación	

Historia de Revisiones

Fecha	Versión	Descripción	Autor

Tabla de contenidos

1. INTRODUCCIÓN	5
1.1. Contexto del problema	5
1.2. Propósito	5
1.3. Ámbito	5
1.4. Definiciones, acrónimos y abreviaciones	6
1.5. Resumen ejecutivo	6
1.6. Arquitectura del sistema	6
2. VISIÓN DEL SISTEMA	7
2.1. Descripción general del sistema	7
2.2. Objetivos del sistema	7
2.3. Principales funcionalidades esperadas	8
2.4. Supuestos y dependencias	9
3. ESTILOS Y PATRONES ARQUITECTÓNICOS	10
3.1. Estilo arquitectónico adoptado	10
3.2. Justificación del estilo según el contexto del sistema	10
3.3. Patrones de diseño aplicados	10
4. MODELO 4 + 1 Y VISTAS ARQUITECTÓNICAS	10
4.1. Vista de escenario	10
4.1.1. Propósito	10
4.1.2. Actores	10
4.1.3. Diagrama general de casos de uso	10
4.1.4. Diagrama de casos de uso específicos	10
4.1.5. Lista de casos de uso	10
4.1.6. Especificación de casos de uso	10
4.2. Vista lógica	10
4.2.1. Propósito	10
4.2.2. Diagrama de clases	10
4.2.3. Descripción diagrama de clases	10
4.3. Vista de implementación/desarrollo	10
4.3.1. Propósito	10
4.3.2. Diagrama de componente	10
4.3.3. Descripción diagrama de componente	10
4.3.4. Diagrama de paquete	10
4.3.5. Descripción diagrama de paquete	10
4.4. Vista de procesos	10
4.4.1. Propósito	10
4.4.2. Diagrama de actividad	10

4.4.3. Descripción diagrama de actividad	10
4.5. Vista física	10
4.5.1. Propósito	10
4.5.2. Diagrama de despliegue	10
4.5.3. Descripción diagrama de despliegue	10
5. REQUISITOS DE CALIDAD	10
5.1. Propósito	10
5.2. Atributos de calidad	10
5.3. Reglas y criterios de evaluación de calidad	10
6. PRINCIPIOS DE DISEÑO APLICADOS	10
6.1. Propósito	10
6.2. Principios de diseño	11
7. PROTOTIPO	11
7.1. Propósito	11
7.2. Mockups (imágenes con una breve descripción)	11
7.3. Justificar herramientas de prototipado	11
8. EVALUACIÓN CALIDAD HEURÍSTICA DE NIELSEN	11
8.1. Propósito	11
8.2. Lista de verificación	11
8.3. Análisis y métricas de resultados	11
9. CONTROL DE VERSIONES	11
9.1. Propósito	11
9.2. Control de versión utilizado (justificar tipo de control de versión - fecha - semántica o secuencial)	11
9.3. Justificar herramientas de versionamiento	11
10. CONCLUSIONES	11
11. BIBLIOGRAFÍA	11

1. INTRODUCCIÓN

1.1. Contexto del problema

El Paso Fronterizo Los Libertadores, ubicado en la Región de Valparaíso, constituye uno de los principales accesos terrestres a Chile desde la República Argentina. El ingreso al país requiere una revisión exhaustiva, tanto de la documentación de los viajeros como de la inspección física de los vehículos. Actualmente, estos procesos se ejecutan de forma manual o a través de sistemas fragmentados, lo que conlleva importantes deficiencias operativas, entre ellas: largos tiempos de espera, duplicidad de tareas, baja trazabilidad de las acciones realizadas y una experiencia insatisfactoria para los usuarios.

Esta situación no solo afecta la eficiencia del proceso aduanero, sino que también limita la capacidad del Estado para ejercer un control eficaz, seguro y transparente sobre los flujos vehiculares y de personas que ingresan al país por esta vía terrestre.

1.2. Propósito

El propósito de este proyecto es diseñar e implementar una plataforma digital integral que permita al Servicio Nacional de Aduanas gestionar de forma eficiente, segura y trazable todos los trámites relacionados con el ingreso de vehículos en los pasos fronterizos. El sistema busca automatizar procesos manuales, optimizar los tiempos de atención, reducir errores humanos y garantizar el cumplimiento de la normativa vigente.

Asimismo, se propone integrar de manera efectiva a los actores relevantes mediante interoperabilidad técnica con sistemas externos tales como el Servicio Agrícola y Ganadero (SAG), la Policía de Investigaciones de Chile (PDI) y el Servicio de Impuestos Internos (SII), asegurando la validación documental y de identidad en tiempo real.

1.3. Ámbito

El sistema propuesto abarca el proceso completo de ingreso de vehículos al territorio nacional por pasos fronterizos, iniciando con el pre-registro en línea por parte del ciudadano, seguido por la verificación documental, el control físico del vehículo, la validación automatizada con organismos externos y la posterior emisión de la autorización o el rechazo del ingreso, en función del cumplimiento de los requisitos legales y sanitarios vigentes.

1.4. Definiciones, acrónimos y abreviaciones

ACRÓNIMO	DESCRIPCIÓN
SAG	Servicio Agrícola Ganadero
PDI	Policía de Investigaciones
API	Application Programming Interface
SUS	System Usability Scale

1.5. Resumen ejecutivo

El presente documento describe el diseño de un sistema informático orientado a modernizar el proceso de control aduanero en el Paso Internacional Los Libertadores. La solución propuesta contempla el pre-registro electrónico de vehículos, la validación automática de documentación, la integración con bases de datos nacionales e internacionales, la inspección digital asistida y la emisión electrónica de autorizaciones de ingreso.

Este sistema prioriza atributos clave como la eficiencia operativa, la seguridad de la información, la interoperabilidad con servicios externos y la trazabilidad de las acciones realizadas, contribuyendo así a fortalecer la gestión fronteriza del país.

1.6. Arquitectura del sistema

El sistema ha sido concebido bajo una arquitectura monolítica, en la cual todos los componentes funcionales están integrados en una única aplicación desplegable. Esta elección permite una implementación inicial más simple, con menores requerimientos de infraestructura y un control centralizado de todos los módulos.

Para su diseño se emplea el patrón de arquitectura Modelo-Vista-Controlador (MVC), el cual garantiza una separación clara entre los niveles de presentación, lógica de negocio y acceso a datos. Esto favorece el mantenimiento, la evolución del sistema y la colaboración entre equipos de desarrollo.

Estructura general del sistema:

- Vista: Interfaz web accesible para ciudadanos y funcionarios aduaneros.
- Controlador: Encargado de gestionar las solicitudes del usuario, redirigiéndolas a los servicios correspondientes.
- Modelo: Responsable del procesamiento de la lógica de negocio y de la interacción con la base de datos.

2. VISION DEL SISTEMA

2.1. Descripción general del sistema

El sistema desarrollado para el Servicio Nacional de Aduanas de Chile tiene como objetivo principal digitalizar y centralizar el proceso de declaración anticipada de ingreso de vehículos en el Paso Internacional Los Libertadores. Este punto fronterizo constituye un nodo estratégico en la relación bilateral con Argentina.

La plataforma, construida bajo una arquitectura monolítica, estará disponible para ciudadanos y funcionarios aduaneros, permitiendo reemplazar los formularios físicos por registros digitales verificables. Los módulos funcionales internos del sistema gestionan los trámites relacionados con:

- Ingreso temporal o permanente de vehículos;
- Transporte de productos no considerados equipaje;
- Ingreso de mascotas, respaldado por certificados emitidos por el SAG.

La plataforma se integra con servicios externos mediante APIs, lo que posibilita la validación automática, en tiempo real, de la información proveniente de entidades como la PDI, el Registro Civil y el SAG. Toda la información es almacenada en una base de datos centralizada y se mantiene disponible para su consulta, fiscalización y auditoría por parte del personal autorizado del Servicio Nacional de Aduanas.

El diseño contempla los siguientes principios:

- Accesibilidad multiplataforma, mediante navegadores web modernos;
- Seguridad y trazabilidad, con control de acceso basado en roles;
- Escalabilidad funcional, habilitando la incorporación futura de nuevos trámites y su implementación en otros pasos fronterizos.

2.2. Objetivos del sistema

Objetivo general

Diseñar e implementar un sistema informático para el Servicio Nacional de Aduanas de Chile, que permita gestionar, validar y centralizar digitalmente las declaraciones de ingreso al país realizadas por ciudadanos, reduciendo los tiempos de atención, mejorando el control documental y garantizando la interoperabilidad con organismos externos.

Objetivos específicos

- a) Permitir el registro anticipado de los siguientes formularios:
 - Declaración de ingreso de mascotas (CZE);
 - Ingreso temporal de vehículos;
 - Transporte de productos no considerados equipaje.
- b) Digitalizar el proceso de validación y autorización, eliminando la dependencia de formularios físicos.
- c) Integrarse mediante APIs con los siguientes sistemas externos:
 - Registro Civil: Verificación de propiedad del vehículo;
 - PDI: Validación del estatus migratorio del ciudadano;
 - SAG: Validación del Certificado Zoosanitario de Exportación.
- d) Implementar trazabilidad completa de todas las acciones del sistema para fines de auditoría y control institucional.

2.3. Principales funcionalidades esperadas

El sistema debe proveer herramientas funcionales para una gestión integral de las declaraciones de ingreso, tanto desde el punto de vista del ciudadano como del funcionario aduanero.

Funciones del Ciudadano:

- Registro y autenticación segura en la plataforma.
- Consulta del estado de solicitudes ingresadas.
- Edición y envío de formularios en caso de observaciones o rechazo.

Funciones del Funcionario de Aduana:

- Acceso autenticado desde estaciones institucionales.
- Panel de control con filtros para gestionar formularios por estado o tipo.
- Visualización de información y documentos adjuntos.
- Gestión de estado del formulario: Aprobado, Rechazado o Pendiente.

Funciones del Sistema (automatizadas):

- Registro automático de acciones de usuarios y cambios de estado.

- Validación de vigencia y autenticidad de los documentos ingresados.
- Conexión en tiempo real con servicios externos vía API.
- Gestión segura de archivos adjuntos (subida, almacenamiento y visualización).

2.4. Supuestos y dependencias

Supuestos

- Los usuarios cuentan con acceso a internet y utilizan navegadores actualizados desde dispositivos móviles o de escritorio.
- Los ciudadanos poseen credenciales válidas para autenticarse en el sistema.
- Los formularios y documentos requeridos serán ingresados de forma anticipada.
- Los funcionarios operan desde estaciones de trabajo conectadas a la red institucional.

Dependencias

a) Servicios externos:

- API del Registro Civil: Validación de propiedad vehicular.
- API de la PDI: Validación migratoria.
- API del SAG: Validación de certificados zoosanitarios.

b) Infraestructura técnica:

- Base de datos centralizada del Servicio Nacional de Aduanas.
- Servidor monolítico correctamente desplegado y configurado.
- Mecanismos de autenticación y control de acceso operativos y seguros.

3. ESTILOS Y PATRONES ARQUITECTÓNICOS

3.1. Estilo arquitectónico adoptado

Se adopta una arquitectura monolítica, en la cual todos los componentes del sistema se integran en una única aplicación desplegable. Esta arquitectura facilita el desarrollo y despliegue inicial del sistema, asegurando una gestión centralizada del código y una menor complejidad en la infraestructura. El sistema está diseñado bajo el patrón

Modelo-Vista-Controlador (MVC), promoviendo una clara separación de responsabilidades entre la presentación, la lógica de negocio y el acceso a datos.

Se utiliza un estilo arquitectónico basado en microservicios, que permite escalabilidad, despliegue independiente de módulos, tolerancia a fallos y mayor mantenibilidad.

3.2. Justificación del estilo según el contexto del sistema

Dado que el sistema se gestionará de forma centralizada y operará en entornos controlados como pasos fronterizos, se ha optado por una arquitectura monolítica, que ofrece simplicidad en el despliegue, facilidad de mantenimiento en sus primeras etapas, y control centralizado sobre todos los módulos funcionales del sistema.

3.3. Patrones de diseño aplicados

- MVC (Modelo-Vista-Controlador): para la separación de lógica de presentación y negocio en la interfaz web.
- Repositorio: para el manejo de persistencia en capas desacopladas.
- Circuit Breaker: para manejar la resiliencia ante fallos de servicios externos.
- API Gateway: para gestionar de forma unificada el acceso a los microservicios.

4. MODELO 4 + 1 Y VISTAS ARQUITECTÓNICAS

4.1. Vista de escenario

4.1.1.

4.1.2. Propósito

El propósito es describir detalladamente los distintos escenarios de interacción entre los actores y el sistema, para identificar los flujos de trabajo típicos, las decisiones clave que se toman, las condiciones previas requeridas, las respuestas esperadas del sistema y las posibles excepciones que pueden ocurrir durante la operación del sistema aduanero.

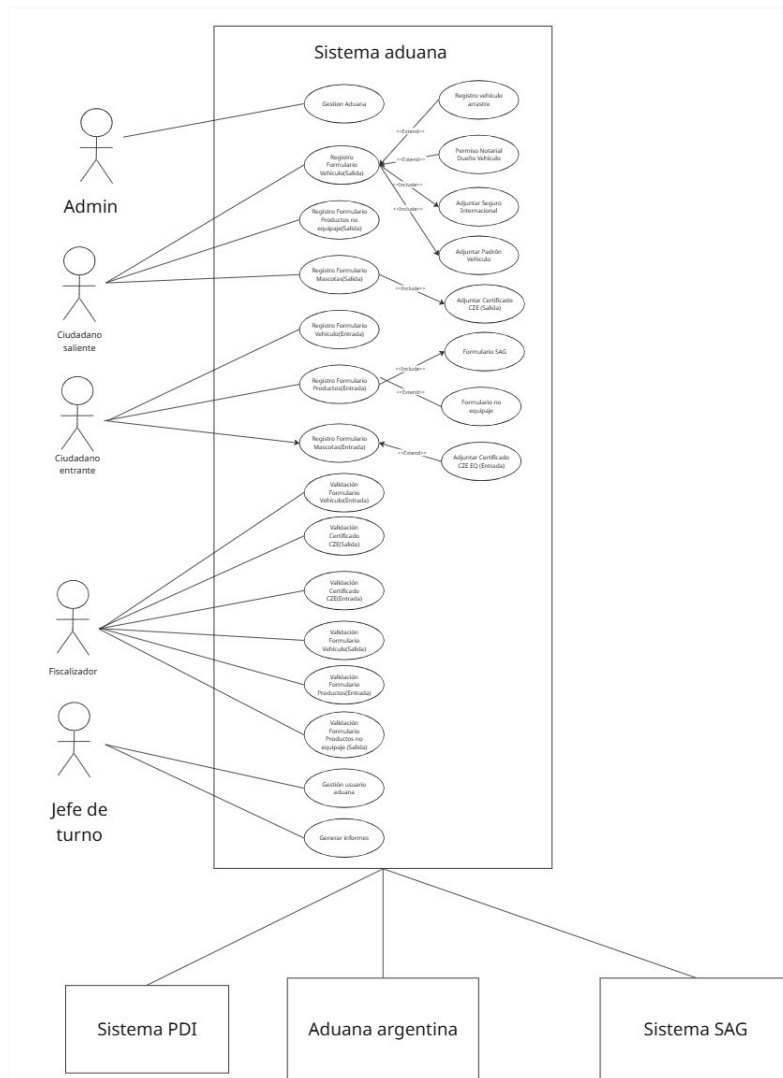
Esta vista permite comprender el comportamiento del sistema frente a distintas situaciones reales en los pasos fronterizos, asegurando una cobertura funcional completa en la definición y posterior implementación del software.

4.1.3. Actores

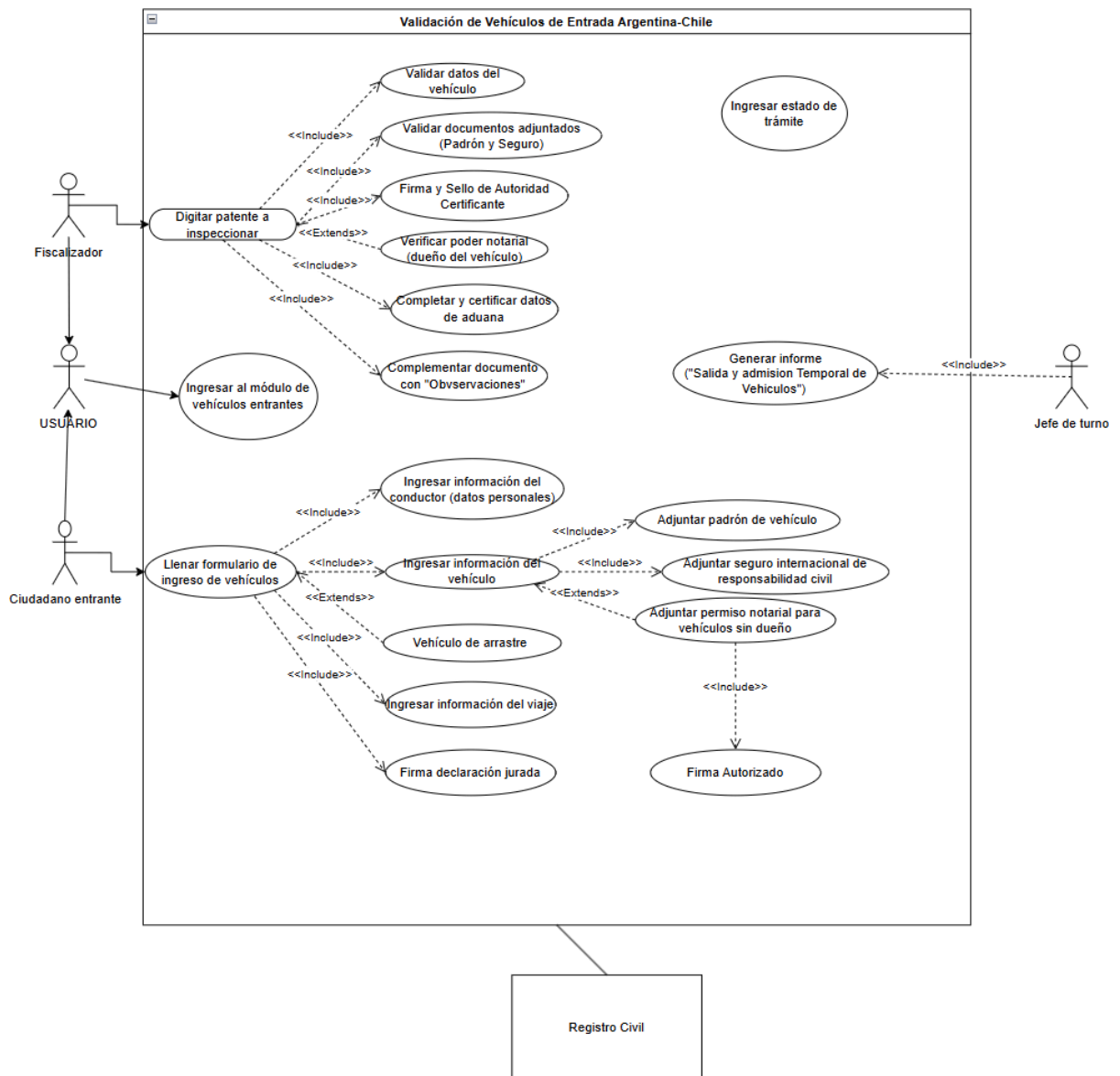
En el sistema general, los actores corresponden a:

- Administrador
- Ciudadano entrante
- Ciudadano saliente
- Fiscalizador
- Jefe de turno

4.1.4. Diagrama general de casos de uso



4.1.5. Diagrama de casos de uso específicos



4.1.6. Lista de casos de uso

Código	Nombre	Actores
CU-001	Ingreso al módulo de vehículos (entrada)	Usuario
CU-002	Completar formulario de entrada	Ciudadano
CU-003	Ingresar información del conductor (datos personales)	Ciudadano
CU-004	Ingresar vehículo de arrastre (si aplica)	Ciudadano

CU-005	Ingresar información del viaje	Ciudadano
CU-006	Firmar declaración jurada	Ciudadano
CU-007	Adjuntar padrón de vehículo (si aplica)	Ciudadano
CU-008	Adjuntar seguro de responsabilidad civil	Ciudadano
CU-009	Adjuntar permiso notarial para vehículos que no viajen con el dueño (si aplica)	Ciudadano
CU-0010	Digitar patente a inspeccionar	Agente Aduana
CU-0011	Validar datos del vehículo	Agente Aduana
CU-0012	Validar documentos adjuntados	Agente Aduana
CU-0013	Firma y sello de autoridad certificante	Agente Aduana
CU-0014	Completar y certificar datos de aduana	Agente Aduana
CU-0015	Complementar con "Observaciones"	Agente Aduana
CU-0016	Ingresar estado del trámite	Agente Aduana
CU-0017	Generar reportes estadísticos	Jefe de Turno
CU-0018	Validaciones externas de documentos	Registro Civil

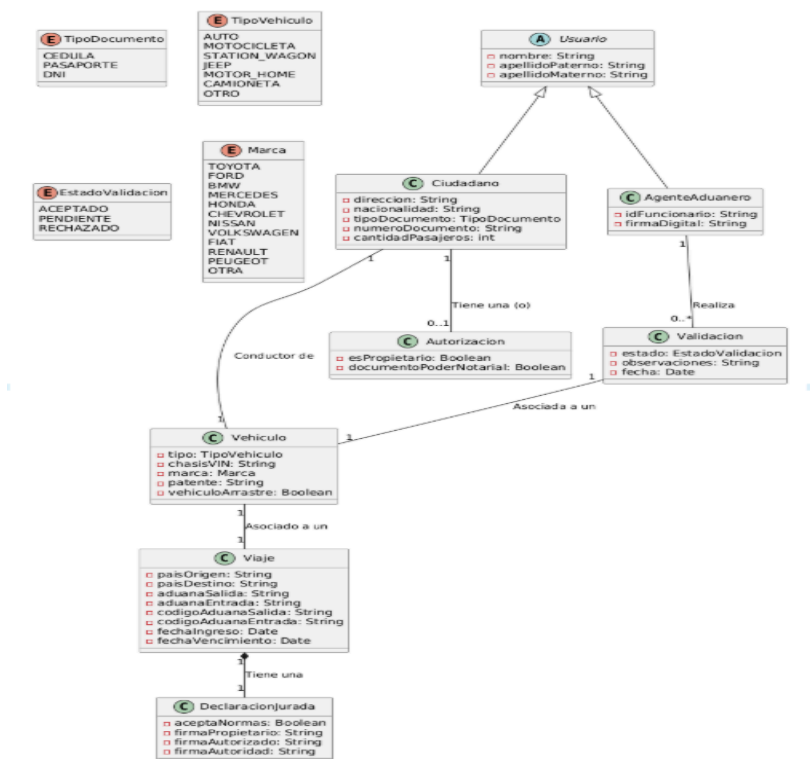
4.1.7. Especificación de casos de uso

4.2. Vista lógica

4.2.1. Propósito

La vista lógica tiene como objetivo modelar la estructura interna del sistema desde una perspectiva orientada a objetos. Esta vista permite definir las entidades principales, sus atributos y relaciones entre ellas, brindando una comprensión clara de cómo se organiza la lógica de negocio dentro del software. Es fundamental para garantizar la cohesión del sistema, facilitar la reutilización de componentes y establecer una base sólida para el desarrollo e implementación del código.

4.2.2. Diagrama de clases



4.2.3. Descripción diagrama de clases

El diagrama de clases representa la estructura principal del sistema de ingreso de ciudadanos a Chile. Muestra las entidades más importantes y cómo se relacionan entre sí.

- **Ciudadano** es la clase principal. Puede tener autorización si no es el dueño del vehículo. También está relacionado con un vehículo que usará para entrar al país.
- El **vehículo** está conectado con un viaje, que contiene los datos del recorrido, como país de origen, aduana y fechas. Cada viaje está ligado a una declaración jurada.
- Un agente aduanero es quien realiza las validaciones, donde se define si el ingreso es aceptado, pendiente o rechazado.
- La clase **usuario** permite reutilizar datos como nombre y apellidos en el sistema.
- Además, hay listas definidas para los tipos de documento, tipos de vehículo, marcas y estados de validación, lo que estandariza los datos y evita errores.

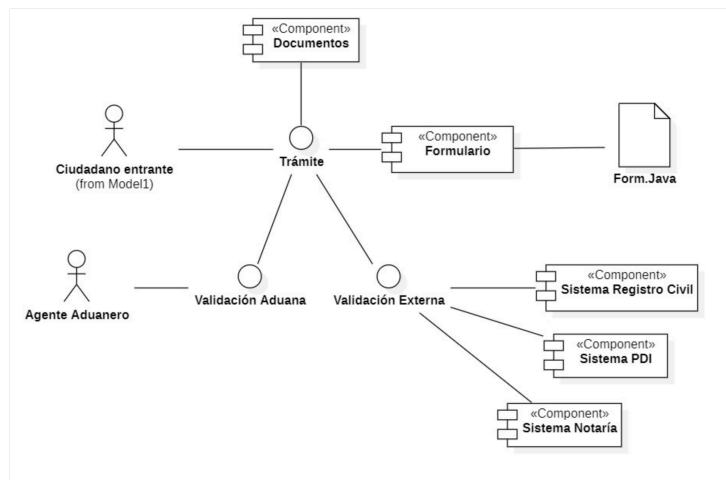
En resumen, el diagrama organiza de forma clara cómo interactúan ciudadanos , vehículos agentes y documentos en el sistema fronterizo.

4.3. Vista de implementación/desarrollo

4.3.1. Propósito

El objetivo de esta vista es representar y poder visualizar cómo se va a estructurar el sistema a nivel de componentes de software. Esto permite identificar los módulos principales, sus responsabilidades y cómo interactúan entre sí, asegurando un desarrollo organizado y mantenible.

4.3.2. Diagrama de componente



4.3.3. Descripción diagrama de componente

Actores del Sistema

- Ciudadano Entrante: Usuario que inicia el trámite digital mediante la carga de documentos y el llenado de formularios.
- Agente Aduanero: Funcionario encargado de revisar, validar y autorizar el trámite dentro del sistema de la aduana.

Componentes del Sistema

- Trámite: Componente central que coordina las acciones del usuario y la validación del proceso. Se relaciona con:
- Documentos: Componente encargado de gestionar la carga y verificación de los documentos subidos por el usuario.
- Formulario: Módulo que representa la lógica del formulario web utilizado para iniciar el trámite. Está vinculado a un archivo de código fuente específico (Form.Java).

Validaciones

- Validación Aduana: Etapa que involucra al agente aduanero, quien revisa los antecedentes ingresados por el ciudadano.
- Validación Externa: Fase de validación automatizada que interactúa con los siguientes sistemas:
- Sistema Registro Civil: Verifica la identidad del ciudadano y antecedentes del vehículo.
- Sistema PDI: Consulta sobre antecedentes penales o restricciones migratorias.
- Sistema Notaría: Valida documentos

4.3.4. Diagrama de paquete

4.3.5. Descripción diagrama de paquete

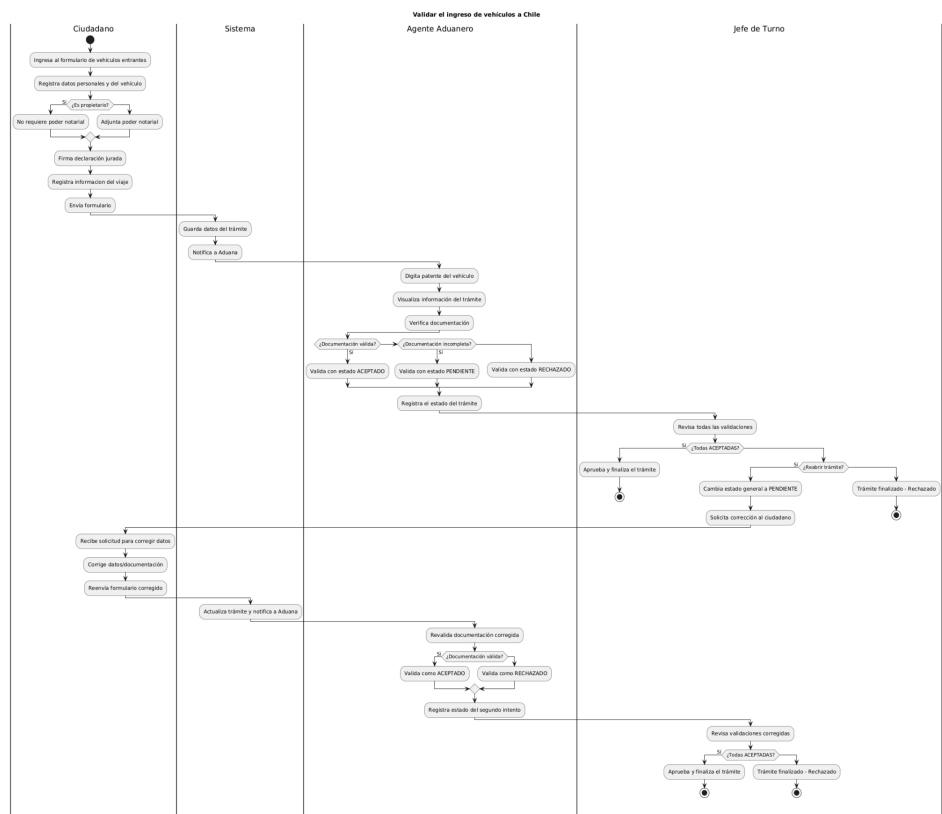
4.4. Vista de procesos

4.4.1. Propósito

La vista de procesos representa el flujo dinámico del sistema para el proceso de entrada de vehículos. Permite visualizar las actividades concurrentes y secuenciales, desde la llegada del vehículo al paso

fronterizo hasta su autorización o rechazo, identificando los responsables y decisiones clave. Esta vista es esencial para entender cómo los distintos módulos y actores interactúan en tiempo real.

4.4.2. Diagrama de actividad



4.4.3. Descripción diagrama de actividad

El diagrama de actividad representa el flujo de acciones y decisiones involucradas en el proceso de validación del ingreso de vehículos a Chile a través de un paso fronterizo. Está dividido en cuatro carriles (swimlanes) que corresponden a los actores principales del proceso: Ciudadano, Sistema, Agente Aduanero y Jefe de Turno

1. Ciudadano

- Inicia el proceso completando un formulario en línea con los datos personales, del vehículo y del viaje.
- Si se requiere poder notarial, adjunta el documento correspondiente.
- Firma la declaración jurada y envía el formulario.
- En caso de observaciones, corrige la información y vuelve a enviar el formulario corregido.

2. Sistema

- Guarda los datos del trámite y notifica a la Aduana.
- Más adelante, actualiza el estado del trámite si se recibe una corrección.

3. Agente Aduanero

- Digita la patente del vehículo para visualizar la información.
- Verifica la documentación recibida.
- Si la documentación es válida, el trámite se valida con estado “ACEPTADO”; si hay inconsistencias, se asigna estado “PENDIENTE” o “RECHAZADO”.
- En caso de corrección, reevalúa la documentación y registra el resultado del segundo intento.

4. Jefe de Turno

- Revisa todas las validaciones realizadas por los agentes.
- Si todas son aceptadas, aprueba y finaliza el trámite.
- Si hay errores, cambia el estado a “PENDIENTE” y solicita correcciones al ciudadano.

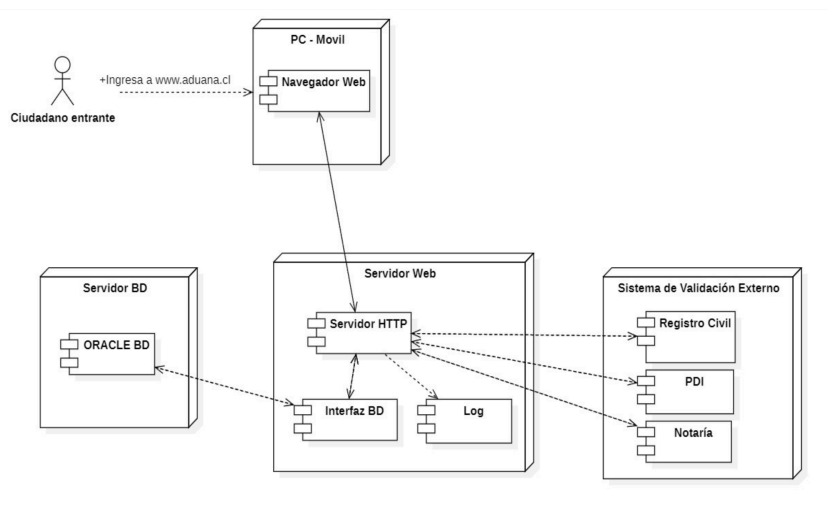
- En una segunda revisión, decide entre aprobar o rechazar el trámite según el cumplimiento de los requisitos.

4.5. Vista física

4.5.1. Propósito

La vista física representa la arquitectura de infraestructura y despliegue físico del sistema. Es útil para visualizar los nodos de hardware (servidores, estaciones de trabajo, dispositivos móviles) y su interacción con el software, garantizando que el diseño sea viable en el entorno operativo real del paso fronterizo.

4.5.2. Diagrama de despliegue



4.5.3. Descripción diagrama de despliegue

Actor Principal

- Ciudadano Entrante: Accede al sistema a través del sitio web oficial (www.aduana.cl) desde un dispositivo cliente.

Cliente (PC/Móvil)

- El ciudadano utiliza un navegador web en un dispositivo (PC o móvil) para interactuar con el sistema. Este navegador se comunica directamente con el servidor web de la aduana.

Servidor Web

Componente central de la aplicación. Incluye:

- Servidor HTTP: Recibe las peticiones del cliente y coordina las

operaciones internas.

- Interfaz BD: Permite la conexión con la base de datos para almacenar o recuperar información del trámite.
- Log: Registra eventos del sistema para trazabilidad y auditoría.

Servidor de Base de Datos

- Alojado de forma independiente, contiene una base de datos Oracle que gestiona toda la información del proceso de validación de ingreso de vehículos.

Sistema de Validación Externo

- Se establece comunicación con plataformas de verificación oficiales:
- Registro Civil: Validación de identidad del ciudadano.
- PDI (Policía de Investigaciones): Validación migratoria y antecedentes del vehículo o del ciudadano.
- Notaría: Verificación de poderes notariales adjuntos al trámite.

5. REQUISITOS DE CALIDAD

5.1. Propósito

Establecer atributos de calidad que aseguren la confiabilidad, usabilidad, seguridad y eficiencia del sistema, siguiendo principios de la Ingeniería de Software.

5.2. Atributos de calidad

ATRIBUTO DE CALIDAD	DESCRIPCIÓN	JUSTIFICACIÓN
Usabilidad	Interfaz clara, adaptada al perfil del usuario.	Facilita la operación rápida por parte del personal aduanero y ciudadanos.
Accesibilidad	Cumple normas WCAG para personas con discapacidades.	Garantiza el uso inclusivo del sistema para todos los usuarios.
Rendimiento	Respuestas rápidas, especialmente en validaciones externas.	Disminuye los tiempos de espera en el paso fronterizo.

Mantenibilidad	Arquitectura modular, principios SOLID.	Permite actualizar el sistema sin afectar otras funciones.
Seguridad	Control de acceso, cifrado de datos, hashing.	Protege datos sensibles y cumple normativa de protección de datos.
Portabilidad	Compatible con distintos navegadores y dispositivos.	Asegura el funcionamiento tanto en PC como en otros dispositivos.

5.3. Reglas y criterios de evaluación de calidad

- Usabilidad: Puntuación ≥ 70 en test de usabilidad SUS (System Usability Scale) y Tasa de error $< 5\%$ en tareas críticas.

Para evaluar estos criterios, se realizan pruebas de usabilidad con los usuarios, evaluaciones heurísticas (Nielsen) y encuestas SUS.

- Accesibilidad: Cumplimiento de criterios WCAG.

Para evaluar la accesibilidad se realizan pruebas manuales de navegación por teclado y contraste. Además de, validar con Lighthouse (Herramienta de Google para desarrolladores)

- Rendimiento: Tiempo de carga de interfaz < 3 segundos y Tiempo de respuesta de servicios API < 1 segundo.

Para evaluar los criterios de rendimiento, se hará uso de herramientas como Postman para pruebas de APIs y la herramienta de Google para desarrolladores en la sección de performance.

- Mantenibilidad: Cobertura de pruebas $\geq 80\%$ y Documentación técnica actualizada

Para garantizar el cumplimiento de los criterios, se realizarán revisiones exhaustivas de código entre pares y se llevará un control de versiones con la herramienta Git.

- Seguridad: Autenticación segura y cifrado de datos en tránsito y en reposo (TLS/SSL)

Para evaluar los criterios de seguridad, se implementarán pruebas de penetración manuales y automáticas.

- Portabilidad: Correcta ejecución en ambientes local, staging y producción. Contenerización con Docker

Para dar cumplimiento a los criterios, se utilizará la herramienta Docker y Docker Compose y se realizarán pruebas de despliegue en distintos entornos, además de, hacer simulaciones en los contenedores.

6. PRINCIPIOS DE DISEÑO APLICADOS

6.1. Propósito

El objetivo de los siguientes principios es guiar la creación del sistema para que sea claro, eficiente, mantenible, fácil de entender y escalar, tanto para los desarrolladores como para los usuarios. Basado en estos principios,

6.2. Principios de diseño

PRINCIPIO	DESCRIPCIÓN	APLICACIÓN EN EL SISTEMA
Cohesión	Cada módulo o clase tiene una única responsabilidad bien definida.	Los servicios como el trámite en línea para ciudadanos, la de validación aduana y externa se ocupan de tareas concretas y específicas.
Bajo acoplamiento	Los módulos están conectados mínimamente para facilitar el mantenimiento.	El componente Formulario está desacoplado del backend; las validaciones externas se hacen a través de interfaces bien definidas con los sistemas externos.
Abstracción	Ocultar detalles internos y muestra solo lo necesario para el uso del sistema.	El usuario no interactúa con la lógica interna de validación, solo con el formulario y carga de documentos.
Encapsulamiento	Protege los datos y funciones internas de un componente.	Cada sistema externo (Registro Civil, PDI, Notaría) se trata como una “caja negra” con la que se interactúa mediante servicios.
Modularidad	El sistema se divide en componentes independientes y reutilizables.	Componentes como Formulario, Documentos y Trámite permiten reutilización y pruebas aisladas.

7. PROTOTIPO

7.1. Propósito

El propósito del prototipado es validar la experiencia del usuario antes del desarrollo completo del sistema, permitiendo identificar mejoras en la interfaz, detectar errores

tempranos y asegurar que los flujos del sistema respondan de forma efectiva a las necesidades de los usuarios clave: ciudadanos y funcionarios aduaneros.

7.2. Mockups (imágenes con una breve descripción)

- **Inicio de sesión:** Muestra campos de ingreso para RUT y contraseña - Clave única.
- **Panel del ciudadano:** Interfaz donde el ciudadano puede ver el estado de sus trámites, editar formularios rechazados y subir documentación adicional.
- **Formulario de ingreso vehicular:** Contiene campos para ingresar los datos del conductor, del vehículo, del viaje, y la opción para adjuntar documentos requeridos.
- **Panel del funcionario aduanero:** Vista con filtros para revisar formularios según estado (pendiente, aprobado, rechazado), visualizar documentos, y emitir resoluciones.
- **Resumen del trámite y firma electrónica:** Visualiza todos los datos ingresados y permite al ciudadano firmar la declaración jurada electrónicamente.

7.3. Justificar herramientas de prototipado

Se utilizó HTML como herramienta principal de prototipado. Fue empleada desde las etapas iniciales por su facilidad para construir interfaces interactivas, simular flujos de navegación y validar la experiencia del usuario directamente en el navegador. HTML permitió diseñar pantallas responsivas, accesibles y funcionales sin necesidad de una implementación final, lo que facilitó detectar mejoras tempranas en la usabilidad del sistema.

Posteriormente, se utilizó este mismo prototipo navegable como base para realizar la evaluación heurística, permitiendo comprobar de forma práctica la estructura, distribución de componentes y el comportamiento de la interfaz antes de su desarrollo completo.

8. EVALUACIÓN CALIDAD HEURÍSTICA DE NIELSEN

8.1. Propósito

Evaluar la usabilidad del sistema a partir de las 10 heurísticas de usabilidad definidas por Jakob Nielsen, con el fin de detectar problemas de diseño que dificulten la experiencia del usuario final y proponer mejoras antes de su desarrollo completo.

8.2. Lista de verificación

Nº	Principio de Usabilidad de Nielsen	Criterio de Evaluación	¿Se cumple? (✓/X)	Observaciones / Evidencia	Gravedad del problema
1	Visibilidad del estado del sistema	¿El sistema informa claramente al usuario de lo que está ocurriendo (cargas, acciones)?	✓	En el formulario ciudadano se indica "Paso 1 de 3 ". El jefe de turno ve claramente el estado de cada trámite.	0
2	Correspondencia entre el sistema y el mundo real	¿La terminología y flujos se relacionan con el lenguaje y lógica del usuario?	✓	Se usan términos, íconos e imágenes que el usuario reconoce fácilmente (PDF, logos institucionales, Clave Única), lo que facilita la comprensión.	1
3	Control y libertad del usuario	¿El usuario puede deshacer/repetir acciones fácilmente?	X	No hay botones evidentes para volver, cancelar o deshacer acciones en varias vistas. El usuario podría sentirse atrapado en algunas pantallas.	2
4	Consistencia y estándares	¿Se mantiene un diseño coherente entre pantallas, botones y mensajes?	✓	Las tres vistas (ciudadano, fiscalizador, jefe) mantienen el mismo estilo visual y estructura.	0
5	Prevención de errores	¿El diseño evita que ocurran errores antes de que sucedan?	✓	JavaScript realiza validaciones previas antes de enviar formularios y evita errores comunes. Aunque se usa alert(), cumple la función. Se recomienda migrar a validaciones visuales más integradas.	1
6	Reconocimiento mejor que recuerdo	¿Las opciones y funciones son visibles sin que el usuario deba recordar información?	✓	El sistema muestra los datos del ciudadano y el vehículo directamente tras buscarlos, sin tener que recordarlos.	0

7	Flexibilidad y eficiencia de uso	¿Permite atajos o personalización para usuarios avanzados?	✓	El formulario está dividido por pasos. El jefe puede actuar por fila, pero no hay atajos de teclado ni funciones para usuarios expertos.	1
8	Diseño estético y minimalista	¿La interfaz evita información innecesaria o ruido visual?	✓	Las interfaces son limpias, con buena jerarquía visual y sin elementos innecesarios.	0
9	Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores	¿Los mensajes de error son claros, comprensibles y ofrecen solución?	✓	Se validan campos con JS y se usan alertas para errores. Aunque funcionales, podrían mejorarse con mensajes visibles en la interfaz.	1
10	Ayuda y documentación	¿Existe ayuda accesible, clara y orientada a la tarea cuando el usuario lo necesita?	✗	No se observa ayuda ni instrucciones dentro del sistema.	1

8.3. Análisis y métricas de resultados

El análisis de usabilidad basado en los 10 principios de Nielsen muestra un 80% de cumplimiento, con 8 principios cumplidos y solo 2 no cumplidos: "Control y libertad del usuario" y "Ayuda y documentación". La gravedad promedio de los problemas es baja (0.7), con la mayoría clasificados como leves. Se destaca una buena visibilidad del estado del sistema, uso de terminología familiar y consistencia visual en todas las vistas.

Las validaciones mediante JavaScript ayudan a prevenir errores, aunque podrían mejorarse con mensajes más integrados visualmente. El sistema presenta un diseño estético y minimalista, con funciones visibles que evitan la sobrecarga cognitiva. Sin embargo, se identifican oportunidades de mejora en ofrecer mayor control al usuario (botones de cancelar o retroceso) y en proporcionar ayuda o instrucciones contextuales que orienten al usuario cuando lo necesite. En resumen, el sistema tiene una base sólida de usabilidad, pero requiere ajustes puntuales para alcanzar un estándar más accesible, eficiente y centrado en el usuario.

9. CONTROL DE VERSIONES

9.1. Propósito

Asegurar la trazabilidad, organización y control de los cambios realizados durante el desarrollo del sistema, facilitando el trabajo en equipo, la detección de errores, la recuperación de versiones anteriores y el seguimiento de entregables.

9.2. Control de versión utilizado (justificar tipo de control de versión - fecha - semántica o secuencial)

Se adoptó un esquema de control de versiones semántico (SemVer), el cual sigue el formato MAJOR.MINOR.PATCH, donde:

- MAJOR se incrementa ante cambios incompatibles con versiones anteriores,
- MINOR ante nuevas funcionalidades retrocompatibles,
- PATCH para correcciones de errores o mejoras menores.

9.3. Justificar herramientas de versionamiento

Se utiliza Git como sistema de control de versiones distribuido, ya que permite un desarrollo colaborativo eficiente debido a su amplio soporte y su capacidad para gestionar ramas, fusiones y auditorías de cambios.

10. CONCLUSIONES

El desarrollo del sistema de digitalización para el ingreso de vehículos y personas en el paso fronterizo Los Libertadores responde directamente a la necesidad de modernizar y optimizar procesos críticos dentro del servicio nacional de aduanas. Actualmente, las tareas se realizan de forma manual o con herramientas fragmentadas, lo que genera una experiencia poco eficiente tanto para los ciudadanos como para los funcionarios encargados del control. Este proyecto propone una solución integral, capaz de automatizar procesos clave, como por ejemplo; reducir tiempos de atención, y facilitar el cumplimiento normativo mediante tecnologías digitales.

Durante el desarrollo del proyecto, se abordaron los distintos niveles de diseño y arquitectura del sistema, Asimismo se definieron casos de uso representativos del flujo real de trabajo en aduanas, contemplando actores como el ciudadano, el agente aduanero y el jefe de turno, lo que aseguró una cobertura funcional alineada con las necesidades del proceso.

El prototipo del sistema, diseñado en HTML funcional, permitió validar anticipadamente la experiencia de usuario y realizar ajustes antes de una implementación final. A través

de esta interfaz, los ciudadanos pueden realizar sus trámites de forma más intuitiva, subir documentación relevante, consultar el estado de su solicitud y firmar electrónicamente una declaración jurada, todo en un entorno centralizado y trazable.

Por otro parte, la evaluación de calidad basada en las heurísticas de Nielsen permitió detectar fortalezas del diseño, como la consistencia visual y visibilidad del estado del sistema, así como también identificar aspectos que pueden ser mejorados, como el control de errores o la ayuda contextual al usuario, Esta revisión temprana fortalece la usabilidad y accesibilidad de la plataforma, pensando en un uso inclusivo eficiente.

En cuanto a la gestión del proyecto, se utilizaron herramientas de control de versiones como Git para el desarrollo técnico y Google Docs para el trabajo documental, permitiendo un trabajo colaborativo ordenado, con registro de cambios y contralor sobre versiones oficiales de cada entrega. Además se hizo uso de Bootstrap para facilitar el diseño responsivo de las interfaces y JavaScript para lograr de forma eficientemente la interacción real del usuario dentro del sistema.

En resumen, el sistema propuesto no solo soluciona problemáticas técnicas y operativas, sino que también demuestra una aplicación concreta de los principios de la ingeniería de software. Aporta eficiente, seguridad y escalabilidad a un proceso clave del estado chileno, y deja sentadas las bases para una futura expansión hacia otros pasos fronterizos del país. Este trabajo refleja una visión centrada en el usuario, alineada con estándares de calidad y buenas prácticas de desarrollo.

11. BIBLIOGRAFÍA

12. ANEXOS

12.1. Planilla de requerimientos

Validación vehículos:

Usuario ciudadano:

- El sistema debe permitir llenar los formularios correspondientes de ingreso de vehículos con la información del conductor, vehículo, vehículo de arrastre (Opcional) e Información del viaje (Anexo 4).

Usuario Agente aduanero:

- Digitar la patente del vehículo a inspeccionar, validando los datos del vehículo y del dueño quien ya debería estar autorizado por PDI, en caso que el vehículo no viaje con su dueño, verificar la existencia de la del poder notarial tramitado por el dueño del vehículo para autorizar al conductor.
- El sistema debe permitir cambiar estado ha aceptado (cumple con todos los requisitos), pendiente (Falta de algún documento) o rechazado (no cumple con los requisitos).

Validación productos:

- El sistema debe permitir llenar el formulario (Declaración Jurada SAG) correspondiente para el acceso de productos. (Anexo 3).

- El agente debe ingresar el identificador del usuario y acceder al formulario SAG declarado.

- Validación mascotas:

- El sistema debe permitir a los ciudadanos poder adjuntar el certificado del CZE de las mascotas que quieran ingresar al país.

- Si el reingreso de la mascota es inferior a 60 días, se aprueba el reingreso. Si el reingreso de la mascota es superior a 60 días, se exige el certificado equivalente al certificado CZE.

- ## 12.2. Carta Gantt

ID	FASE / TAREA	DURACION EN HRS.	FEMCO	FTEMPO	COSTO CLP	NOL	S1	S2	S3	S4	S5	JN	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
A Análisis y Respuestas																											
1.1	Análisis de prioridades / Planificación inicial de hardware	4	01-06-2025	06-04-2025	\$237,350	Análisis QA																					
1.2	Revisión de normativas (SAC, PDI, Alumnos)	18	05-06-2025	06-04-2025	\$218,340	Análisis QA																					
1.3	Evaluación de estándares curriculares e análisis de roles	32	07-06-2025	06-04-2025	\$715,980	Desarrollador Full Stack																					
1.4	Comunicación con el cliente para el levantamiento de requisitos	16	15-06-2025	11-06-2025	\$208,454	Desarrollador Full Stack																					
1.5	Documentación y validación de requerimientos funcionales	40	12-06-2025	15-04-2025	\$1,902,200	Desarrollador Front End																					
1.6	Documentación y validación de requerimientos no funcionales	40	18-06-2025	19-04-2025	\$1,902,200	Desarrollador Front End																					
B Mantenimiento del Sistema																											
2.1	Diseño de arquitectura del sistema	40	20-04-2025	23-04-2025	\$902,440	Desarrollador Back End																					
2.2	Modelado de base de datos relacional (DBE) y normalización	40	24-04-2025	28-04-2025	\$902,440	Desarrollador Back End																					
2.3	Diseño UML de interfaces gráficas basadas en la identidad empresarial	40	28-04-2025	01-05-2025	\$1,202,440	Desarrollador Front End																					
2.4	Planificación de migración con sistemas existentes	32	02-05-2025	05-05-2025	\$715,980	Desarrollador Full Stack																					
2.5	Definición de reglas de negocio y políticas de seguridad	16	08-05-2025	07-05-2025	\$384,320	Desarrollador Back End																					
C Pruebas																											
3.1	Desarrollo de módulos de información documental / declaraciones conductor/investigativo; validación de menús de edición, producción, extracción/imprimir, acciones	80	08-05-2025	15-05-2025	\$1,924,480	Desarrollador Back End																					
3.2	Implementación de autorización y control de usuarios	40	16-05-2025	20-05-2025	\$902,440	Desarrollador Back End																					
3.3	Integración de módulos legacy que requiera actualizaciones (PDF/Evt)	64	21-05-2025	26-05-2025	\$1,420,096	Desarrollador Full Stack																					
3.4	Interoperación con sistemas externos (SACS, PDI, Argentina)	64	27-05-2025	31-05-2025	\$1,420,096	Desarrollador Full Stack																					
D Puesta en Funcional por módulo																											
4.1	Puesta en funcionamiento - Módulo de inicio	32	02-06-2025	05-06-2025	\$436,480	Análisis QA																					
4.2	Puesta en funcionamiento - Login y Screen de Inicio	32	05-06-2025	08-06-2025	\$436,480	Análisis QA																					
4.3	Puesta en funcionalidad - Módulos de gestión	40	09-06-2025	12-06-2025	\$545,600	Análisis QA																					
4.4	Puesta en estabilidad con usuarios reales	40	13-06-2025	16-06-2025	\$545,600	Análisis QA																					
E Implementación																											
5.1	Despliegue progresivo del sistema en fases iterativas	40	17-06-2025	21-07-2025	\$894,880	Desarrollador Full Stack																					
5.2	Verificación de funcionamiento en entorno real	24	23-06-2025	26-06-2025	\$977,440	Desarrollador Back End																					
5.3	Ajustes finales post -despliegue	16	25-06-2025	28-06-2025	\$308,454	Desarrollador Full Stack																					
F Capacitación y Mantenimiento																											
6.1	Diseño de manuales y materiales de capacitación	40	30-06-2025	03-07-2025	\$1,902,200	Desarrollador Front End																					
6.2	Ejecución de sesiones de capacitación por perfil de usuario	32	04-07-2025	08-07-2025	\$715,980	Desarrollador Full Stack																					
6.3	Implementación de actual. de acompañamiento post - implementación	32	09-07-2025	11-07-2025	\$715,980	Desarrollador Full Stack																					
6.4	Puesta en funcionalidad de soporte técnico	16	14-07-2025	15-07-2025	\$308,454	Desarrollador Full Stack																					
6.5	Mantenimiento correctivo (bug o registra errores)	16	14-07-2025	18-07-2025	\$308,454	Desarrollador Full Stack																					
6.6	Planificación de actualización	16	14-07-2025	01-08-2025	\$337,454	Desarrollador Full Stack																					

ID	FASE / TAREA	DURACION HRS	F.INICIO	F.TERMINO	COSTO CLP [1]	ROL
1	Análisis y Requisitos					
1.1	Análisis del problema y flujos actuales de frontera	24	01-04-2025	04-04-2025	\$327.360	Analista QA
1.2	Revisión de normativas (SAG, PDI, Aduanas)	16	05-04-2025	06-04-2025	\$218.240	Analista QA
1.3	Identificación de stakeholders y análisis de roles	32	07-04-2025	09-04-2025	\$715.969	Desarrollador Full Stack
1.4	Reunión con el cliente para el levantamiento de requisitos	16	10-04-2025	11-04-2025	\$358.464	Desarrollador Full Stack
1.5	Documentación y validación de requerimientos funcionales	40	12-04-2025	15-04-2025	\$1.002.200	Desarrollador Front End
1.6	Documentación y validación de requerimientos no funcionales	40	16-04-2025	19-04-2025	\$1.002.200	Desarrollador Front End
2	Diseño del Sistema					
2.1	Diseño de arquitectura del sistema	40	20-04-2025	23-04-2025	\$962.440	Desarrollador Back End
2.2	Modelado de base de datos relacional (DER y normalización)	40	24-04-2025	28-04-2025	\$962.440	Desarrollador Back End
2.3	Diseño UI/UX de interfaces gráficas basadas en la identidad empresarial	48	28-04-2025	01-05-2025	\$1.202.640	Desarrollador Front End
2.4	Planificación de integración con sistemas externos	32	02-05-2025	05-05-2025	\$715.968	Desarrollador Full Stack
2.5	Definición de reglas de negocio y políticas de seguridad	16	06-05-2025	07-05-2025	\$384.320	Desarrollador Back End
3	Desarrollo					
3.1	Desarrollo de módulos de automatización documental y declaraciones (conductor/vehículo/viaje, validación de menores de edad, productos animales/vegetales, mascotas)	80	08-05-2025	15-05-2025	\$1.924.880	Desarrollador Back End
3.2	Implementación de autenticación y control de usuarios	40	16-05-2025	20-05-2025	\$962.440	Desarrollador Back End
3.3	Desarrollo del módulo de reportes automatizados (PDF/Excel)	64	21-05-2025	26-05-2025	\$1.429.056	Desarrollador Full Stack
3.4	Integración con sistemas externos (SAG, PDI, Argentina)	64	27-05-2025	31-05-2025	\$1.429.056	Desarrollador Full Stack
4	Pruebas					
4.1	Pruebas de funcionalidad por módulo	32	02-06-2025	05-06-2025	\$436.480	Analista QA
4.2	Pruebas de rendimiento, carga y tiempo de respuesta	32	05-06-2025	08-06-2025	\$436.480	Analista QA
4.3	Pruebas de seguridad (acceso, roles, confidencialidad)	40	09-06-2025	12-06-2025	\$545.600	Analista QA
4.4	Pruebas de usabilidad con usuarios reales	40	13-06-2025	16-06-2025	\$545.600	Analista QA
5	Implementación					
5.1	Despliegue progresivo del sistema en pasos fronterizos piloto	40	17-06-2025	21-07-2025	\$894.960	Desarrollador Full Stack
5.2	Verificación de funcionamiento en entorno real	24	23-06-2025	25-06-2025	\$577.464	Desarrollador Back End
5.3	Ajustes finales post - despliegue	16	25-06-2025	28-06-2025	\$358.464	Desarrollador Full Stack
6	Capacitación y Mantenimiento					
6.1	Diseño de manuales y materiales de capacitación	40	30-06-2025	03-07-2025	\$1.002.200	Desarrollador Front End
6.2	Ejecución de sesiones de capacitación por perfil de usuario	32	04-07-2025	08-07-2025	\$715.968	Desarrollador Full Stack
6.3	Asistencia inicial al uso (acompañamiento post - implementación)	32	09-07-2025	11-07-2025	\$715.968	Desarrollador Full Stack
6.4	Planificación de soporte técnico	16	14-07-2025	15-07-2025	\$358.464	Desarrollador Full Stack
6.5	Mantenimiento correctivo (bugs o mejoras mínimas)	16	16-07-2025	18-07-2025	\$358.464	Desarrollador Full Stack
6.6	Planificación de mejoras evolutivas	24	21-07-2025	23-07-2025	\$537.696	Desarrollador Full Stack
		976 [2]			\$21.081.481 [3]	

La Carta Gantt presenta el cronograma detallado del proyecto de automatización del sistema aduanero para el Paso Los Libertadores, abarcando desde el 1 de abril hasta el 23 de julio de 2025. Organizada en 6 fases clave (Análisis, Diseño, Desarrollo, Pruebas, Implementación y Capacitación), visualiza la duración, responsables y secuencia de 26 tareas críticas, con un presupuesto total de \$21.081.481 CLP. Los hitos principales incluyen la integración con sistemas externos (SAG/PDI) y el despliegue piloto en frontera. Las barras horizontales representan el avance temporal, destacando dependencias y roles asignados (Full Stack, QA, Front/Back End)