

# **Proyecto Aduanas (DAS) Documento Arquitectura de Software**



Nombre: Sebastian Salgado  
Agustin Fernandez  
Carlos Navarrete  
Eduardo Uribe  
Curso: Ingeniería de Software  
Profesor: Ricardo Pino

## Identificación de Documento

Identificación	0.0.1
Proyecto	proyecto aduanas
Versión	1.5.3

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

<b>1. INTRODUCCIÓN</b>	<b>3</b>
1.1. CONTEXTO DEL PROBLEMA (General)	4
1.2. Propósito	4
1.3. Ámbito	4
1.4. Definiciones, acrónimos y abreviaciones	4
1.5. Resumen ejecutivo (General)	5
1.6. Arquitectura del sistema (General)	5
<b>2. VISIÓN DEL SISTEMA (General)</b>	<b>6</b>
2.1. Descripción general del sistema	6
2.2. Objetivos del sistema	6
2.3. Principales funcionalidades esperadas	6
2.4. Supuestos y dependencias	6
<b>3. ESTILOS Y PATRONES ARQUITECTÓNICOS (General)</b>	<b>7</b>
3.1. Estilo arquitectónico adoptado	7
3.2. Justificación del estilo según el contexto del sistema	7
3.3. Patrones de diseño aplicados	7
<b>4. MODELO 4 +1 Y VISTAS ARQUITECTÓNICAS</b>	<b>8</b>
4.1. VISTA DE ESCENARIO (General y salida vehículo o entrada vehículo)	8
4.1.1. Propósito (General)	8
4.1.2. Actores (General)	8
4.1.3. Diagrama general de casos de uso (General)	8
4.1.4. Diagrama de casos de uso específicos (salida vehículo)	9
4.1.5. Lista de casos de uso (salida vehículo)	9
4.1.6. Especificación de casos de uso (UN caso de uso principal de la salida vehículo)	10
4.2. VISTA LÓGICA (salida vehículo)	11
4.2.1. Propósito	11
4.2.2. Diagrama de clases	11
4.2.3. Descripción diagrama de clases	11
4.3. VISTA DE IMPLEMENTACIÓN/DESARROLLO (salida vehículo)	12
4.3.1. Propósito	12
Describir cómo se estructura el código fuente del sistema desde la perspectiva del desarrollador, y cómo se organizan los módulos y componentes que lo conforman.	12
4.3.2. Diagrama de componente	12
4.3.3. Descripción diagrama de componente	12
4.3.4. Diagrama de paquete	13
4.3.5. Descripción diagrama de paquete	13
4.4. VISTA DE PROCESOS (salida vehículo)	14
4.4.1. Propósito	14

Modelar el comportamiento dinámico del sistema, destacando la interacción concurrente entre procesos que participan en la entrada y salida de vehículos.	14
4.4.2. Diagrama de actividad	14
4.4.3. Descripción diagrama de actividad	14
4.5. VISTA FÍSICA (salida vehículo)	15
4.5.1. Propósito	15
Describir cómo se desplegará físicamente el sistema, incluyendo servidores, dispositivos de entrada (lectores OCR/RFID), red de comunicaciones y elementos externos.	15
4.5.2. Diagrama de despliegue	15
4.5.3. Descripción diagrama de despliegue	15
<b>5. REQUISITOS DE CALIDAD (General)</b>	<b>16</b>
5.1. Propósito	16
5.2. Atributos de calidad	16
5.3. Reglas y criterios de evaluación de calidad	17
<b>6. PRINCIPIOS DE DISEÑO APLICADOS</b>	<b>17</b>
6.1. Propósito	17
6.2. Principios de diseño	18
6.3. Diseño centrado en el usuario (UX/UI, prototipos, experiencia de usuario)	18
<b>7. PROTOTIPO</b>	<b>19</b>
7.1. Mockups	19
7.2. Herramientas de prototipado y su justificación	23
<b>8. EVALUACIÓN DE CALIDAD HEURÍSTICA DE NIELSEN</b>	<b>23</b>
8.1. Propósito	23
8.2. Lista de verificación	24
8.3. Análisis y métricas de resultados	25
<b>9. CONTROL DE VERSIONES</b>	<b>26</b>
9.1. Propósito	26
9.2. Control de versión utilizado	26
9.3. Justificación de herramientas de versionamiento	26
<b>10. CONCLUSIONES</b>	<b>28</b>
<b>11. BIBLIOGRAFÍA</b>	<b>29</b>

# 1. INTRODUCCIÓN

## 1.1. Contexto del Problema (General)

El Servicio Nacional de Aduanas de Chile enfrenta desafíos críticos en pasos fronterizos como por paso Los Libertadores, algunos de los mayores problemas observados son los siguientes:

- Procesos manuales en la revisión de vehículos y documentos.
- Falta de integración entre sistemas del SAG, PDI y Aduanas.
- Demoras en la validación de certificados zoosanitarios (CZE/CZI) para mascotas y mercancías.
- Entre otros.

## 1.2. Propósito

El propósito de nuestro proyecto es diseñar una nueva arquitectura de software para mejorar la eficiencia del control aduanero en fronteras, integrando actores como Aduanas, SAG, PDI y Carabineros, mediante una solución moderna basada en microservicios.

## 1.3. Ámbito

El sistema abarca la fiscalización vehicular, control documental, integración con otros servicios públicos y soporte a los procesos de revisión e inspección.

## 1.4. Definiciones, acrónimos y abreviaciones

ACRÓNIMO	DESCRIPCIÓN
<i>SAG</i>	Servicio Agrícola y Ganadero
<i>PDI</i>	Policía de Investigaciones
<i>CU</i>	Caso de Uso
<i>CZI / CZE</i>	Certificado Zoosanitario de Importación / Exportación

### 1.5. Resumen ejecutivo (General)

Este informe presenta la arquitectura del nuevo sistema aduanero, destacando una visión centrada en microservicios, con enfoque en rendimiento, interoperabilidad y escalabilidad. Se incluyen vistas arquitectónicas, requisitos de calidad, principios de diseño y casos de uso clave.

### 1.6. Arquitectura del sistema (General)

El sistema se basa en una arquitectura de microservicios organizada según el modelo 4+1:

- **Vista de Escenario:** Muestra la interacción entre actores como fiscalizadores, conductores y sistemas externos (SAG, PDI).
- **Vista Lógica:** Define los módulos funcionales como vehículos, inspección, usuarios y documentación.
- **Vista de Desarrollo:** Representa los microservicios desarrollados con Spring Boot y organizados por funcionalidad.
- **Vista de Procesos:** Describe los flujos de inspección, verificación documental e integración con terceros.
- **Vista Física:** Detalla el despliegue en contenedores Docker orquestados con Kubernetes y uso de bases de datos PostgreSQL y MongoDB.

## **2. VISIÓN DEL SISTEMA (General)**

### **2.1. Descripción general del sistema**

Plataforma moderna de gestión aduanera que permite la fiscalización automatizada, el seguimiento de vehículos y mercancías, y la integración de múltiples servicios del Estado en la frontera.

### **2.2. Objetivos del sistema**

- Optimizar el flujo de vehículos en las fronteras.
- Reducir tiempos de revisión.
- Integrar servicios de Aduanas, SAG, PDI y Carabineros.
- Asegurar cumplimiento legal y tributario.

### **2.3. Principales funcionalidades esperadas**

- Declaración electrónica de bienes.
- Gestión documental de vehículos y personas.
- Integración con sistemas externos (SAG, PDI).
- Registro y seguimiento de inspecciones.
- Generación automática de informes.

### **2.4. Supuestos y dependencias**

- Participación activa de funcionarios capacitados.
- Integración mediante API REST con SAG y PDI.
- Conectividad estable en los pasos fronterizos.

### 3. ESTILOS Y PATRONES ARQUITECTÓNICOS (General)

#### 3.1. Estilo arquitectónico adoptado

El estilo arquitectónico que adoptaremos en este proyecto es el de los microservicios.

#### 3.2. Justificación del estilo según el contexto del sistema

El sistema requiere alta escalabilidad, integración con otros servicios estatales y despliegue modular. El estilo de microservicios permite actualizaciones independientes, mayor tolerancia a fallos y escalamiento granular.

#### 3.3. Patrones de diseño aplicados

- **MVC (Modelo-Vista-Controlador):** Se usará para organizar el frontend del sistema de forma clara. Por ejemplo, los fiscalizadores interactuarán con una interfaz (Vista) que envía solicitudes a un Controlador, el cual procesa la lógica y se comunica con los modelos de datos, como registros de vehículos o inspecciones.
- **Repositorio:** Cada microservicio contará con su propia capa de repositorio para acceder a bases de datos. Esto permite cambiar o ajustar la base de datos sin afectar la lógica principal del sistema. Por ejemplo, el servicio de inspección podrá recuperar o guardar información sin preocuparse por cómo está almacenada.
- **Gateway API:** Todo el tráfico externo entrará por una puerta única que redirige las solicitudes al microservicio correspondiente. Esto facilita la seguridad, ya que se puede controlar todo el acceso desde un solo punto, además de simplificar la estructura del sistema para los usuarios y sistemas externos como el SAG.
- **Circuit Breaker:** Este patrón se aplicará para proteger el sistema cuando un servicio externo (como SAG o PDI) no esté disponible. En lugar de esperar indefinidamente, el sistema detecta la falla y responde con un mensaje claro o busca una alternativa temporal, manteniendo la estabilidad general.



## 4. MODELO 4 +1 Y VISTAS ARQUITECTÓNICAS

### 4.1. VISTA DE ESCENARIO (General y salida vehículo o entrada vehículo)

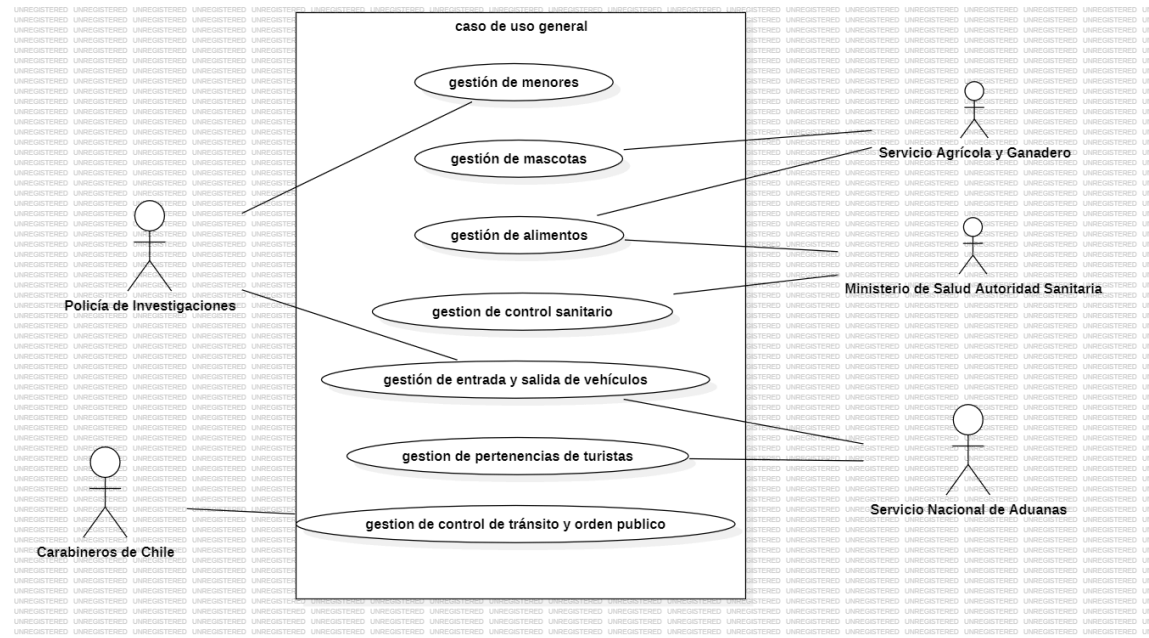
#### 4.1.1. Propósito (General)

Describir las interacciones clave entre los actores y el sistema durante el proceso de salida de vehículos desde el país, incluyendo las validaciones de identidad y documentos, los registros en el sistema, y el proceso de autorización final.

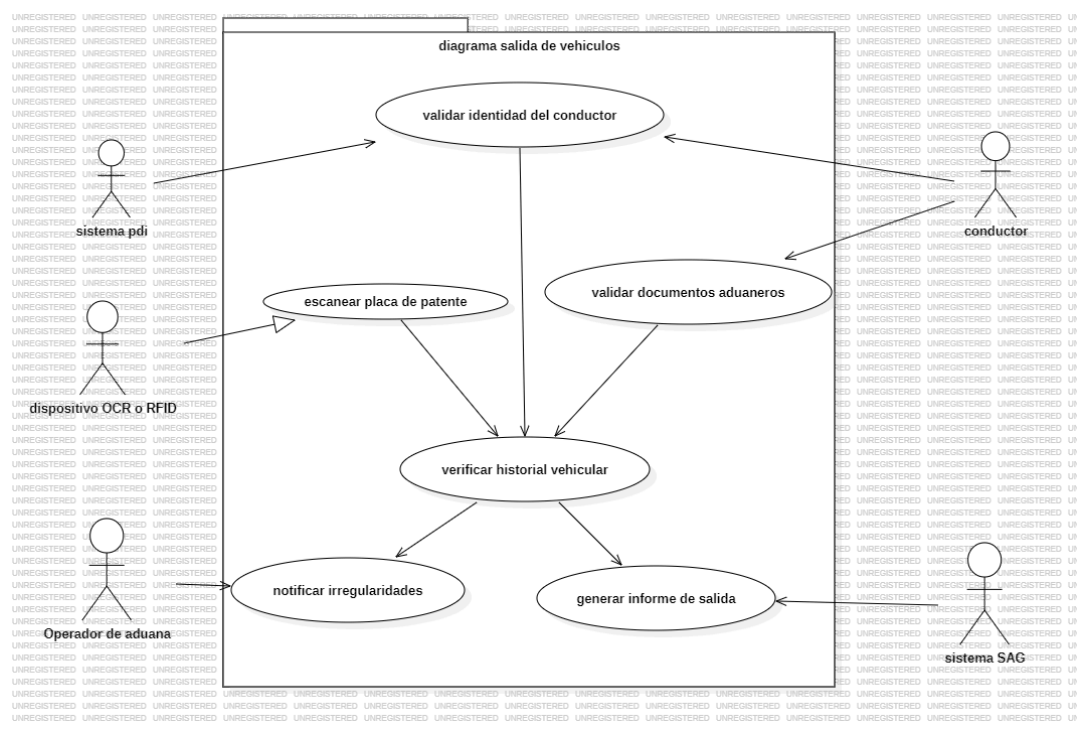
#### 4.1.2. Actores (General)

- PDI
- Carabineros de Chile
- Servicio agrícola y ganadero
- Servicio nacional de aduanas
- Ministerio de Salud/autoridad Sanitaria

#### 4.1.3. Diagrama general de casos de uso (General)



4.1.4. Diagrama de casos de uso específicos (salida vehículo)



4.1.5. Lista de casos de uso (salida vehículo)

Código	Nombre	Actores
CU-001-001	Completar Formulario	conductor
CU-002-001	Imprimir Formulario	conductor
CU-003-001	Actualizar Formulario	conductor
CU-004-001	Enviar formulario	conductor
CU-005-001	Guardar formulario	conductor
CU-006-001	Notificar irregularidades	operador de aduanas
CU-007-001	Generar informe de salida	sistema SAG
CU-008-001	Escanear placa de patente	dispositivo OCR O RPID
CU-009-001	Validar identidad del conductor	Sistema pdi, conductor
CU-010-001	Validar documentos aduaneros	conductor
CU-011-001	Validar historial vehicular	Sistema pdi, conductor, dispositivo OCR O RPID

#### 4.1.6. Especificación de casos de uso (UN caso de uso principal de la salida vehículo)

<b>Caso de uso</b>	<b>Enviar Formulario</b>	<b>Identificador: CU-004-001</b>
<b>Actores</b>	<b>Usuario (Fiscalizador), Sistema</b>	
<b>Tipo</b>	Primario	
<b>Referencias</b>	RF-014: Gestión de formularios de salida CU-001-001: Completar Formulario CU-005-001: Guardar Formulario	
<b>Precondición</b>	El formulario de salida ha sido completado correctamente y se encuentra validado. El usuario ha iniciado sesión en el sistema con los permisos necesarios.	
<b>Postcondición</b>	El formulario es enviado y registrado en el sistema. El estado del vehículo cambia a "Autorizado para salida". Se genera un comprobante digital o código de verificación.	
<b>Descripción</b>	Este caso de uso permite al usuario enviar un formulario previamente completado y validado para autorizar la salida de un vehículo desde una zona controlada. El sistema procesa y registra la información, generando una confirmación y actualizando el estado del vehículo.	
<b>Resumen</b>	El fiscalizador, tras verificar y completar los datos requeridos, envía el formulario al sistema para oficializar la salida del vehículo. El sistema valida, registra y genera una salida autorizada.	

#### CURSO NORMAL

<b>Nro.</b>	<b>Ejecutor</b>	<b>Paso o Actividad</b>
1	Usuario	Inicia sesión en el sistema.
2	Usuario	Accede al módulo de formularios de salida.
3	Usuario	Selecciona el formulario previamente completado.
4	Sistema	Valida que el formulario esté completo y sin errores.
5	Usuario	Confirma el envío del formulario.
6	Sistema	Registra el formulario como "enviado".
7	Sistema	Actualiza el estado del vehículo a "Autorizado para salida".
8	Sistema	Genera un comprobante o código de autorización.
9	Usuario	Visualiza y guarda el comprobante generado.

## CURSO ALTERNATIVO

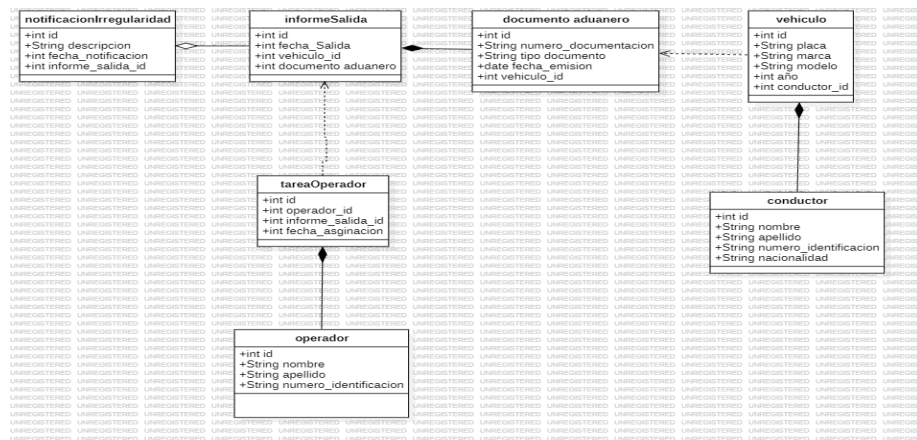
Nro.	Descripción de acciones alternas
1	Si el usuario no puede iniciar sesión, se muestra un mensaje de error y se cancela el proceso.
4	Si el formulario contiene errores o está incompleto, el sistema muestra advertencias y no permite el envío hasta corregirlos
6	Si ocurre un error al registrar el formulario (fallo del sistema o base de datos), se notifica al usuario y se sugiere intentar más tarde.
9	Si el comprobante no se puede generar, se da la opción de reintentar o recibirlo por correo institucional.

### 4.2. VISTA LÓGICA (salida vehículo)

#### 4.2.1. Propósito

La vista lógica tiene como objetivo modelar los elementos funcionales del sistema que intervienen en el proceso de entrada y salida de vehículos. Describe las clases principales, sus relaciones, atributos y métodos, facilitando una comprensión clara de la estructura del software desde el punto de vista funcional.

#### 4.2.2. Diagrama de clases



#### 4.2.3. Descripción diagrama de clases

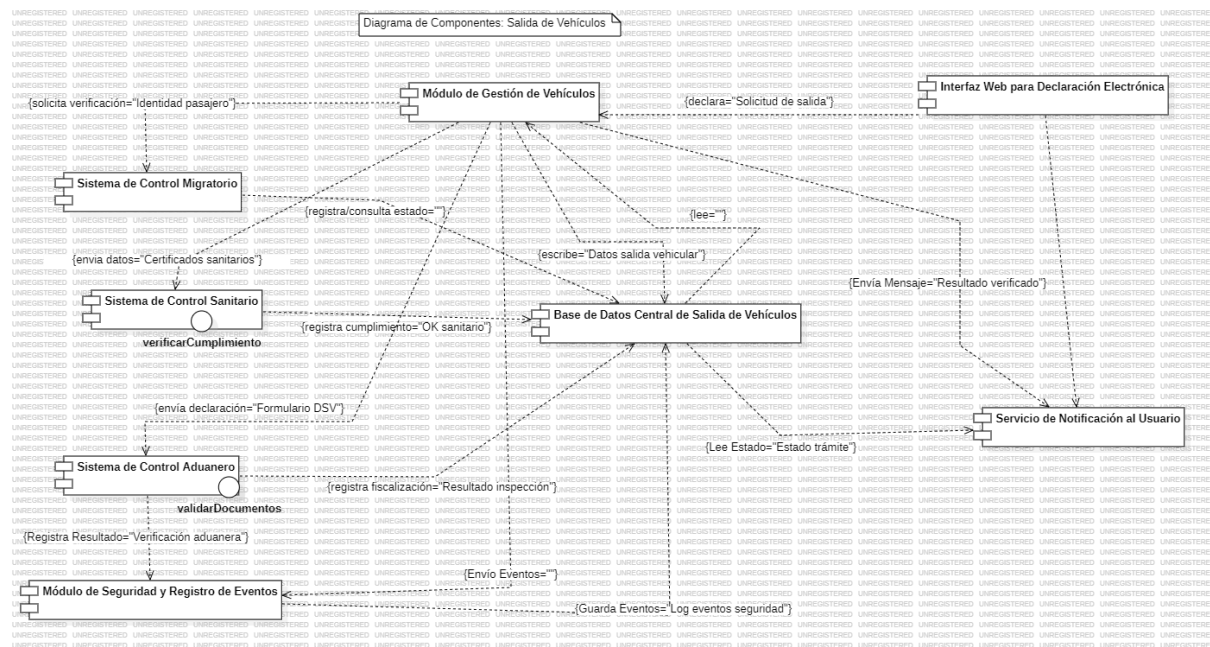
Este diagrama de clases modela los componentes lógicos de la base de datos para el proceso de control de salida vehicular en puntos fronterizos chilenos, específicamente para el Servicio Nacional de Aduanas. Se enfoca en representar las entidades clave, sus atributos, relaciones y cardinalidad para implementar una base de datos relacional robusta y normalizada.

## 4.3. VISTA DE IMPLEMENTACIÓN/DESARROLLO (salida vehículo)

### 4.3.1. Propósito

Describir cómo se estructura el código fuente del sistema desde la perspectiva del desarrollador, y cómo se organizan los módulos y componentes que lo conforman.

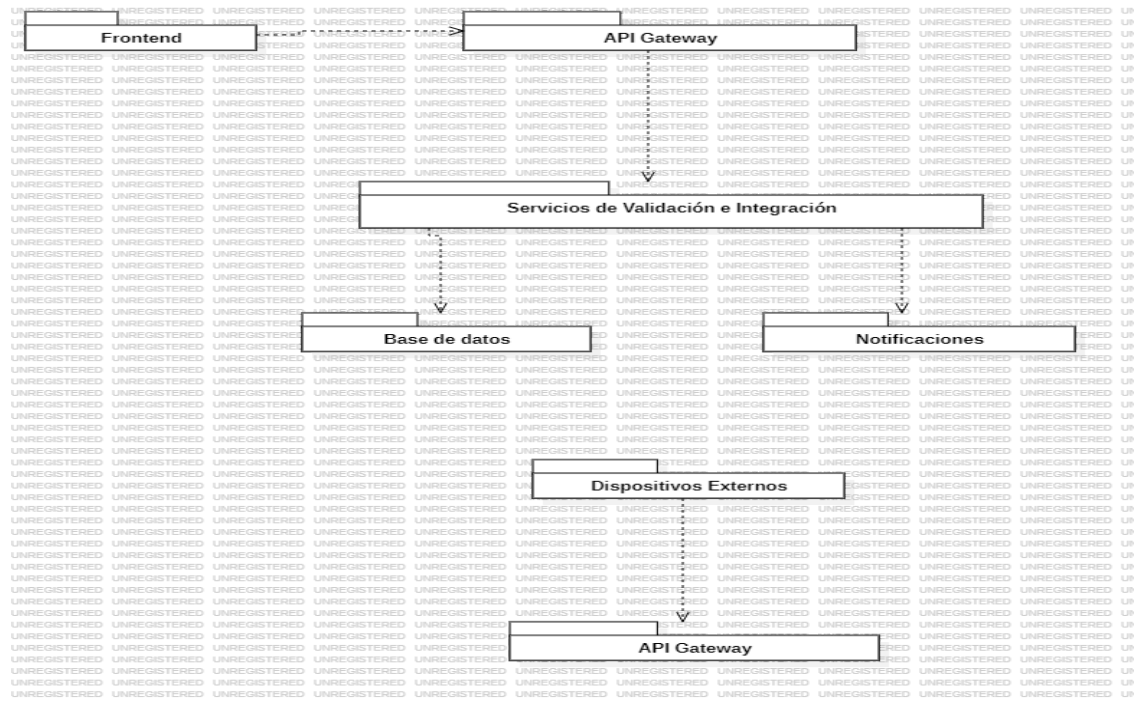
### 4.3.2. Diagrama de componente



### 4.3.3. Descripción diagrama de componente

El diagrama de componentes representa la estructura del sistema encargado de gestionar la salida de vehículos desde el país. En él se identifican los módulos principales, como la interfaz web, el módulo de gestión, los sistemas de control (migratorio, sanitario, aduanero y de seguridad), la base de datos central y el servicio de notificación. El diagrama muestra cómo estos componentes interactúan entre sí mediante el intercambio de información, reflejando el flujo lógico del proceso.

#### 4.3.4. Diagrama de paquete



#### 4.3.5. Descripción diagrama de paquete

El diagrama de paquetes muestra la organización lógica de los módulos en agrupaciones funcionales, facilitando la comprensión de la estructura del sistema. Cada paquete corresponde a un servicio o grupo funcional: API Gateway, servicios de validación e integración, notificaciones, frontend, bases de datos y dispositivos externos.

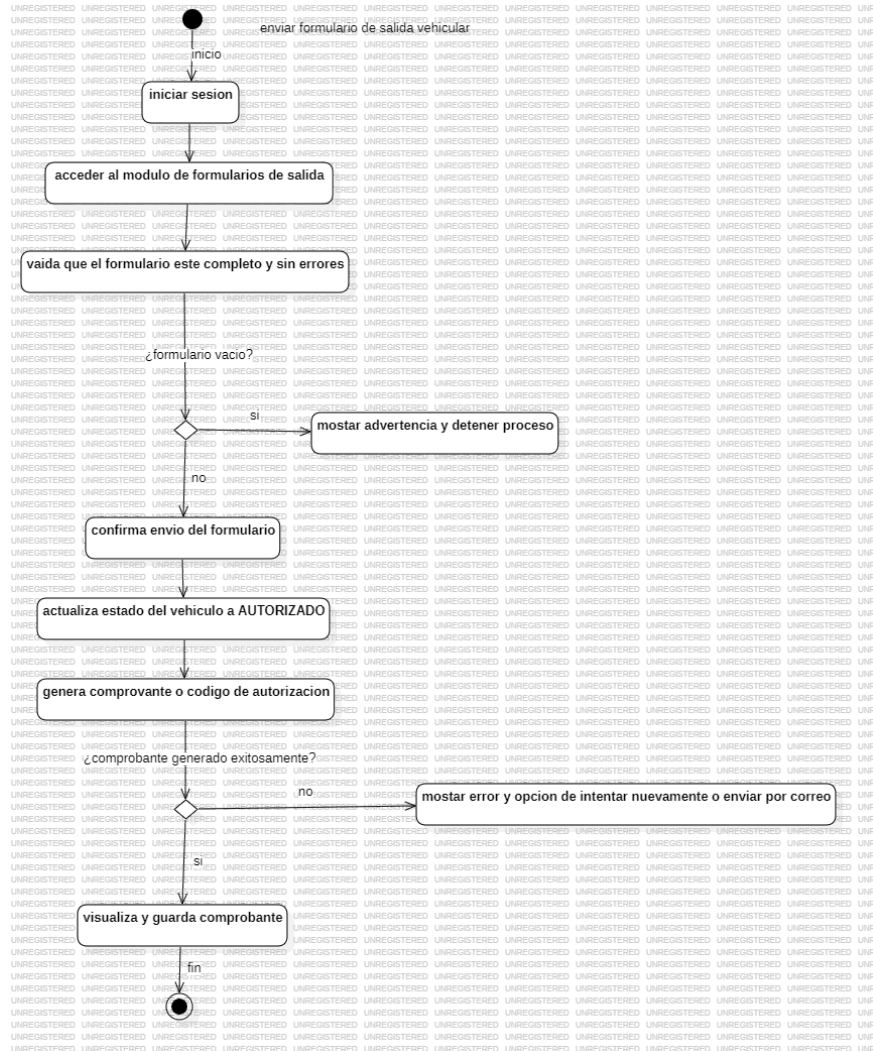


## 4.4. VISTA DE PROCESOS (salida vehículo)

### 4.4.1. Propósito

*Modelar el comportamiento dinámico del sistema, destacando la interacción concurrente entre procesos que participan en la entrada y salida de vehículos.*

### 4.4.2. Diagrama de actividad



### 4.4.3. Descripción diagrama de actividad

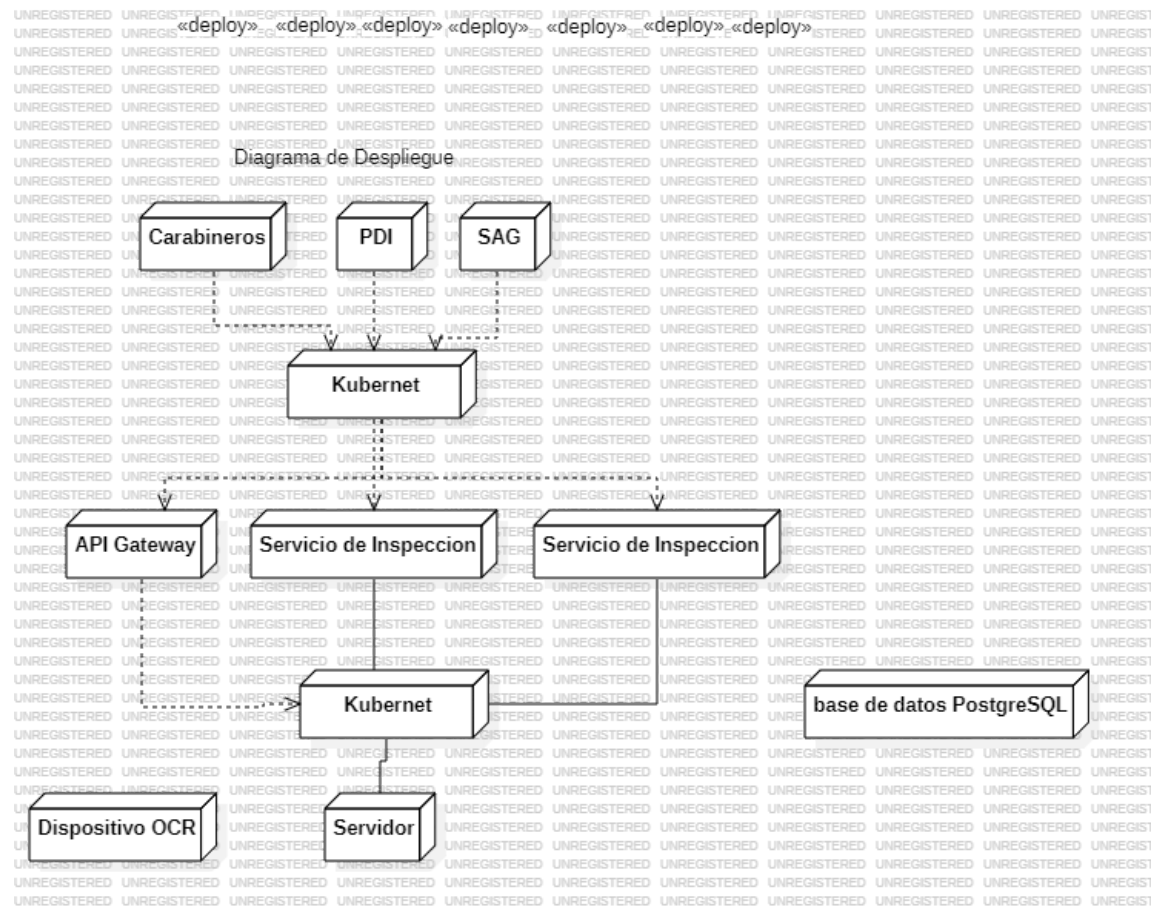
Es el caso de uso “enviar Formulario de salida vehicular”. Representa el flujo de acciones del fiscalizador desde que inicia sesión hasta que visualiza el comprobante de autorización. Incluye decisiones importantes como la validación del formulario y la generación exitosa del comprobante, con caminos alternativos en caso de errores.

## 4.5. VISTA FÍSICA (salida vehículo)

### 4.5.1. Propósito

*Describir cómo se desplegará físicamente el sistema, incluyendo servidores, dispositivos de entrada (lectores OCR/RFID), red de comunicaciones y elementos externos.*

### 4.5.2. Diagrama de despliegue



### 4.5.3. Descripción diagrama de despliegue

El sistema se despliega en un clúster de Kubernetes, donde cada microservicio (como inspección, documentación e integración externa) corre en contenedores Docker. Un API Gateway centraliza las solicitudes y redirige al servicio correspondiente.

El sistema se conecta con bases de datos PostgreSQL, y con sistemas externos como SAG, PDI y Carabineros mediante API REST. También se integran dispositivos físicos como lectores OCR/RFID para identificación vehicular en terreno.



## 5. REQUISITOS DE CALIDAD (General)

### 5.1. Propósito

El propósito de esta sección es definir los atributos de calidad que debe cumplir el sistema aduanero propuesto para garantizar un funcionamiento eficiente, seguro y centrado en el usuario. Estos atributos impactan directamente en la experiencia de los usuarios, la mantenibilidad del sistema y su integración con otros organismos estatales.

### 5.2. Atributos de calidad

ATRIBUTO DE CALIDAD	DESCRIPCIÓN	JUSTIFICACIÓN
Usabilidad	Interfaz intuitiva para fiscalizadores	Minimiza capacitación y errores
Accesibilidad	Cumplimiento WCAG AA	Inclusión de usuarios diversos
Rendimiento	Tiempo de respuesta < 2s	Flujo ágil en frontera
Seguridad	Autenticación con clave única y JWT	Protección de datos
Mantenibilidad	Servicios desacoplados	Cambios independientes

### 5.3. Reglas y criterios de evaluación de calidad

Atributo de Calidad	Cómo se evaluará	Herramientas o métodos que se usarán
Usabilidad	Se evaluará si el sistema es fácil de entender y usar para los funcionarios. Se buscará que las personas puedan realizar sus tareas sin confusión.	Pruebas con usuarios reales, encuestas simples de satisfacción (como SUS), observación directa.
Accesibilidad	Se comprobará que el sistema pueda ser utilizado por personas con diferentes capacidades (por ejemplo, con baja visión).	Herramientas automáticas como WAVE o Lighthouse, revisión manual con lista de verificación.
Rendimiento	Las pantallas principales deben cargarse en menos de 2 segundos para evitar demoras en el control.	Pruebas de carga con múltiples usuarios simulados, mediciones de tiempo de respuesta.
Seguridad	Se revisará que sólo las personas autorizadas puedan ingresar al sistema y que los datos estén protegidos.	Revisión de accesos, pruebas de seguridad (como escaneos de vulnerabilidades), autenticación por roles.
Mantenibilidad	Se medirá qué tan fácil es hacer cambios o arreglos sin dañar otras partes del sistema.	Pruebas después de aplicar cambios, revisión del código organizado por módulos.

## 6. PRINCIPIOS DE DISEÑO APLICADOS

### 6.1. Propósito

El propósito de esta sección es establecer los principios fundamentales de diseño utilizados en la construcción del sistema aduanero, con el fin de asegurar que sea robusto, mantenible, escalable y centrado en el usuario final. Estos principios guían tanto el diseño arquitectónico como el desarrollo del software.

## 6.2. Principios de diseño

PRINCIPIO	DESCRIPCIÓN	APLICACIÓN EN EL SISTEMA
<b>Cohesión</b>	Cada módulo o clase tiene una única responsabilidad bien definida.	Los servicios están diseñados para realizar tareas específicas y no múltiples funciones
<b>Acoplamiento</b>	Bajo acoplamiento entre microservicios	Cambios no afectan otros servicios
<b>Encapsulamiento</b>	Esconde lógica interna de servicios	Cada API gestiona su propio dominio
<b>Modularidad</b>	Separación en módulos funcionales	Servicios por tipo de acción (documentos, inspección)
<b>Abstracción</b>	Oculta detalles de infraestructura	Frontend no conoce implementación backend

## 6.3. Diseño centrado en el usuario (UX/UI, prototipos, experiencia de usuario)

El diseño de la interfaz se realizó siguiendo principios de usabilidad y accesibilidad, considerando a usuarios clave como funcionarios aduaneros, fiscalizadores del SAG y la PDI, y personal administrativo. Las decisiones de diseño incluyen:

- **Interfaces simples e intuitivas:** con menús claros, navegación jerárquica y lenguaje comprensible.
- **Acceso rápido a funciones críticas:** como revisión de documentos, historial de vehículos y generación de informes.
- **Sistema de ayuda contextual:** disponible en todas las vistas para apoyar a los usuarios durante el uso del sistema.
- **Prototipos interactivos evaluados por usuarios reales:** utilizando herramientas como Figma para simular escenarios reales de operación.
- **Consistencia visual:** respetando la identidad gráfica institucional de Aduanas de Chile.

Además, se consideró el entorno físico de operación (frontera, cabinas, dispositivos móviles), lo que motivó el diseño adaptativo (responsive) y compatible con pantallas táctiles y laptops robustas.

## 7. PROTOTIPO

**Link:** Además se subió el archivo del prototipo al ava.

<https://www.figma.com/proto/OAjMoOn6TpRwIUdIcKPCGQ/Prototipo-Salida-Vehicular-Aduanas?node-id=0-1&p=f&t=jDnJfna6i784oPkQ-0&scaling=min-zoom&content-scaling=fixed&page-id=0%3A1>

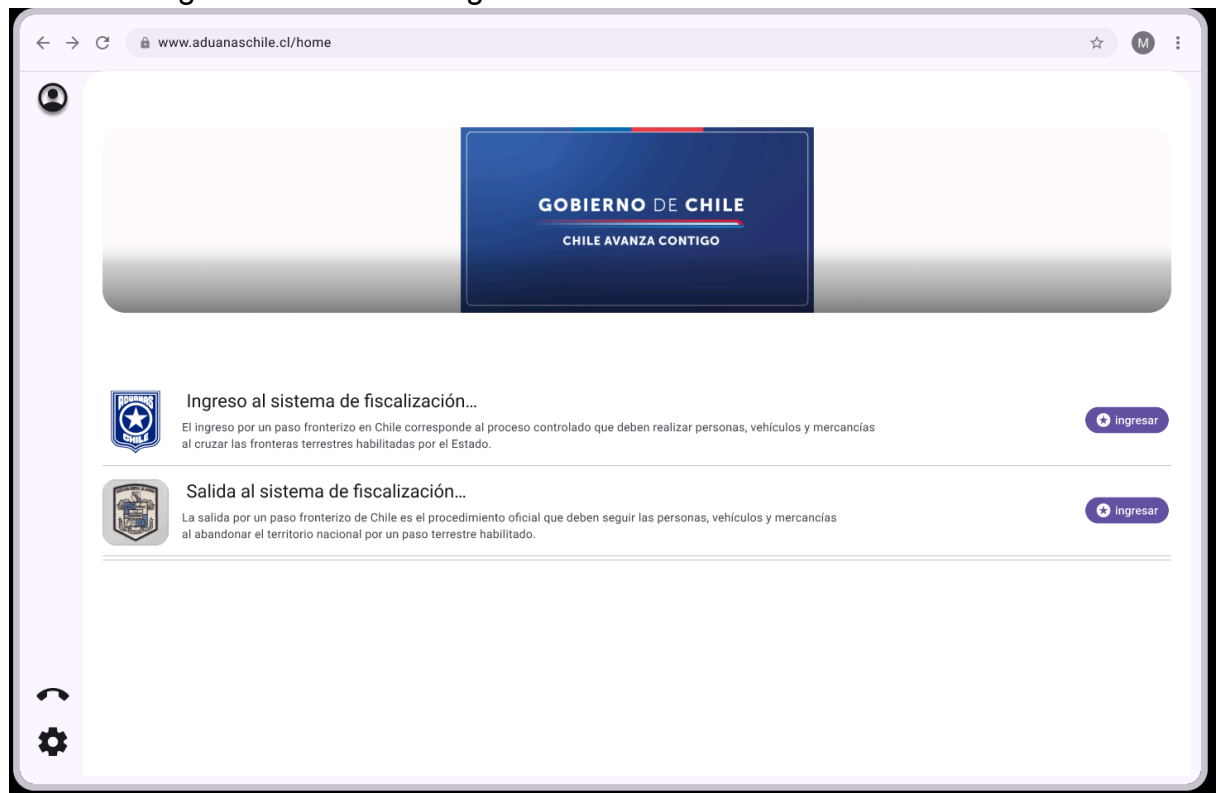
El propósito del prototipo es representar visualmente el sistema propuesto para la **gestión de la salida de vehículos**, permitiendo validar desde etapas tempranas el diseño, flujo y usabilidad del sistema. Este prototipo funcional simula la interacción del usuario con el sistema final, facilitando la detección temprana de mejoras antes del desarrollo e implementación técnica.

El prototipo se enfoca exclusivamente en el caso de uso asignado, considerando tanto **requisitos funcionales** (como el flujo de registro, inicio de sesión y salida del vehículo) como algunos **requisitos no funcionales** (como usabilidad, accesibilidad y claridad visual).

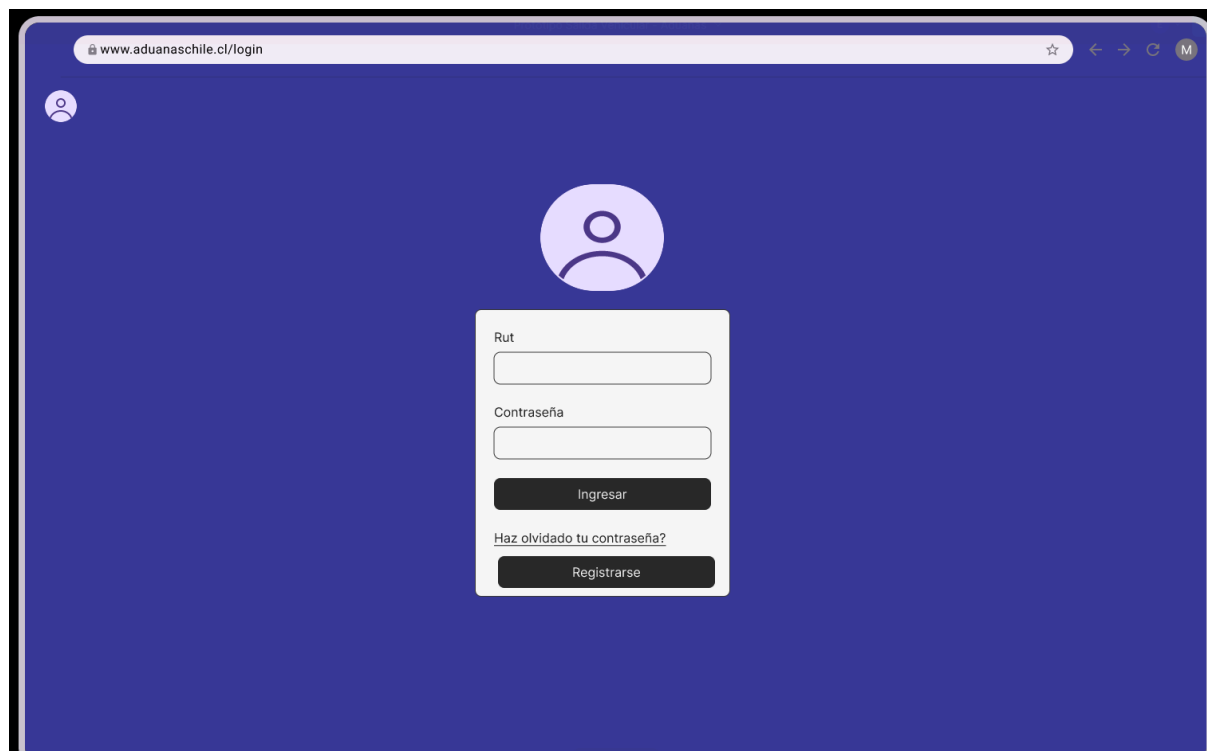
### 7.1. Mockups

A continuación, se describen las principales pantallas del prototipo desarrollado en Figma:

- **Portada:** pantalla inicial que da la bienvenida al sistema, con botones para dirigirse a los apartados importantes pero estos están bloqueados hasta que el usuario ingrese sesión o se registre.



- **Inicio de sesión:** permite que el conductor acceda al sistema con sus credenciales.



The image shows a web browser window with the address bar displaying "www.aduanaschile.cl/login". The page has a dark blue background. In the top left corner, there is a small white user icon. In the center, there is a large white user icon. Below this icon is a white login form. The form contains the following elements:

- A label "Rut" above a text input field.
- A label "Contraseña" above a text input field.
- A dark blue button labeled "Ingresar".
- A link labeled "Haz olvidado tu contraseña?".
- A dark blue button labeled "Registrarse".

- **Formulario de registro:** permite al conductor registrarse, con autocompletado de datos relevantes para facilitar la experiencia.

www.aduanaschile.cl/registro

### Datos del propietario

Nombres: miguel ignacio

Apellido paterno: muñoz

Apellido materno: sanchez

Genero: masculino

Fecha de nacimiento: 08/18/1985

Nacionalidad: chilena

Rut: 12.345.678-9

### Datos de contacto

Telefono de contacto: 912345678

Telefono adicional: 987654321

Correo electronico: figmatest@gmail.com

Correo adicional: testfigma@gmail.com

limpiar registrarse

### Datos direccion

Pais de residencia: chile

Region: biobio

Ciudad: chiguayante

Comuna: chiguayante

Calle: ohiggins

- **Panel de ajustes:** área donde el usuario puede modificar información personal o preferencias.

www.aduanaschile.cl/configuraciones

### Datos Personales

Cambiar foto

Nombre Completo: Miguel Muñoz

Correo Electrónico: figmatest@gmail.com

RUT: 12.345.678-9

Volver

### Credenciales de acceso

Contraseña Actual

Nueva contraseña

Confirmar nueva contraseña

Actualizar Contraseña

### Seguridad y sesiones

Autenticación de dos pasos

### Preferencias

Tema: ☒ (toggle)

Formato de hora: 24 horas

Tamaño de fuente: Mediano

### Notificaciones

- ☒ Recibir avisos de formularios
- ☒ Recibir notificaciones del sistema
- ☒ Avisos de cambios de contraseña o seguridad

### Acciones de la cuenta

Eliminar Cuenta

- **Sección de contacto:** permite al usuario contactar con soporte en caso de dudas o problemas.



### Información de contacto



Número de asistencia técnica:

+56 2 2345 6789



Teléfono central Aduanas:

+56 2 2222 3344



Correo de soporte accesible:

soporte.accesibilidad@aduanas.cl

Volver

"Este sistema está diseñado para garantizar la accesibilidad a todos los usuarios. Si necesitas asistencia adicional, por favor contáctanos."

- **Sistema de ayuda y FAQ:** proporciona orientación clara sobre cómo utilizar el sistema, incluyendo preguntas frecuentes.

### ¿PREGUNTAS FRECUENTES?

1. ¿Qué es un Formulario de Salida de Vehículos y para qué se utiliza?

El formulario se utiliza para registrar los vehículos que salen del país a través del paso fronterizo. Es obligatorio para fines de control y fiscalización aduanera.

2. ¿Qué documentos necesito para completar un formulario de salida?

Necesitas:  
• RUT del conductor  
• Patente del vehículo  
• Fecha y hora estimada de salida  
• Declaración jurada si llevas carga o mercancía especial

3. ¿Qué hago si cometí un error en un formulario ya enviado?

Debes comunicarte con el personal de Aduanas o asistencia técnica para solicitar la corrección. También puedes anular el formulario si no ha sido procesado aún.

4. ¿Qué significa el estado "Pendiente" en un formulario?

Significa que el formulario ha sido creado pero aún no ha sido validado o completado por el personal de fiscalización.

5. ¿Puedo crear un formulario para otro conductor?

Sí, siempre que tengas la información del conductor y estés autorizado para actuar en su representación. El sistema validará los datos ingresados.

6. ¿Qué debo hacer si el sistema no me deja ingresar un nuevo formulario?

Verifica que no tengas formularios anteriores pendientes o errores en los campos obligatorios. Si el problema persiste, contacta con soporte técnico.



Las pantallas fueron diseñadas siguiendo una estética simple, clara y centrada en el usuario, cumpliendo con principios de accesibilidad y usabilidad.

## 7.2. Herramientas de prototipado y su justificación

Se utilizó la herramienta **Figma** para el diseño del prototipo, debido a sus múltiples ventajas en proyectos de interfaz de usuario:

- **Trabajo colaborativo** en tiempo real entre los integrantes del equipo.
- **Historial de versiones** automático, ideal para registrar el avance del proyecto.
- **Prototipado interactivo**, que permite simular el comportamiento de la interfaz.
- **Diseño responsivo**, adaptable a distintos tamaños de pantalla.
- **Interfaz intuitiva** y sin necesidad de instalación, al estar basada en la nube.

La elección de Figma permitió mantener una comunicación clara entre los miembros del equipo y agilizó la iteración y mejora del prototipo de manera continua.

## 8. EVALUACIÓN DE CALIDAD HEURÍSTICA DE NIELSEN

### 8.1. Propósito

La evaluación heurística tiene como objetivo identificar problemas de usabilidad en la interfaz, basándose en los **10 principios de usabilidad de Nielsen**, para mejorar la experiencia del usuario antes de implementar el sistema.



## 8.2. Lista de verificación

Se realizó una evaluación utilizando la plantilla de calidad de software basada en Nielsen. El resultado fue el siguiente:

Nº	Principio de Nielsen	¿Se cumple?	Observaciones
1	Visibilidad del estado del sistema	✓	Al enviar formularios, se informa si fue exitoso o hubo error.
2	Correspondencia entre el sistema y el mundo real	✓	Se utilizan términos comunes, comprensibles por cualquier conductor.
3	Control y libertad del usuario	✓	El usuario puede corregir información antes de confirmar.
4	Consistencia y estándares	✓	Botones, tamaños y colores son coherentes entre pantallas.
5	Prevención de errores	✓	Se valida la patente antes de permitir registrar la salida.
6	Reconocimiento mejor que recuerdo	✓	El sistema guía al usuario y no exige recordar pasos anteriores.
7	Flexibilidad y eficiencia de uso	✓	La información del conductor se autocompleta, mejorando la eficiencia.
8	Diseño estético y minimalista	✓	Pantallas limpias, sin elementos innecesarios.
9	Ayudar a reconocer, diagnosticar y recuperarse de errores	✓	Se muestran mensajes claros ante errores y cómo resolverlos.
10	Ayuda y documentación	✓	FAQ accesible y sistema de ayuda contextualizado en el formulario.

### 8.3. Análisis y métricas de resultados

- **Total de heurísticas evaluadas:** 10
- **Total de heurísticas cumplidas:** 10/10
- **Problemas críticos identificados:** Ninguno
- **Grado de cumplimiento:** 100%

El análisis indica que el prototipo cumple con todos los principios de Nielsen, mostrando una alta calidad en términos de diseño de interfaz y experiencia de usuario. La estructura clara, validación de errores, ayudas visuales y consistencia del sistema contribuyen a una experiencia fluida para los conductores.

## 9. CONTROL DE VERSIONES

### 9.1. Propósito

El control de versiones tiene como objetivo llevar un registro organizado y trazable de los cambios realizados al prototipo durante su desarrollo. Permite identificar qué se modificó, cuándo, quién lo hizo y por qué, facilitando así la iteración, la mejora continua y la colaboración entre los miembros del equipo.

### 9.2. Control de versión utilizado

Se utilizó un control de versiones **basado en fecha y semántico**, adaptado al contexto de diseño en Figma:

- **Control basado en fecha:** aprovechando el historial automático que genera Figma por cada modificación, se puede visualizar con precisión el día y hora en que se hicieron los cambios.
- **Control semántico:** se asignaron nombres a versiones clave del prototipo utilizando una convención de versiones tipo **v1.0**, **v1.1**, etc., indicando versiones mayores, menores o finales. Esto facilitó el seguimiento del progreso y permitió identificar hitos importantes dentro del ciclo de diseño.

### 9.3. Justificación de herramientas de versionamiento

Se utilizó **Figma** como herramienta de prototipado, la cual incorpora un sistema interno de control de versiones que resultó suficiente para los fines del proyecto. Las principales razones para esta elección son:

- Registro automático de cada modificación con sello de tiempo y usuario responsable.
- Visualización cronológica de cambios con opción de restaurar versiones anteriores.
- Posibilidad de nombrar versiones relevantes y organizar los avances del equipo.
- Facilita la colaboración simultánea y el control de cambios sin requerir herramientas externas como Git.

Además, se creó una **tabla de control de versiones externa**, complementaria al historial de Figma, para documentar los cambios más significativos de manera estructurada. Esto permitió mantener una trazabilidad clara y comprensible para la evaluación.

Nº de versión	Fecha	Integrantes que modificaron	Descripción del cambio realizado	Motivo del cambio
v1.0	17-06-2025	Eduardo, Sebastián	Diseño de portada y login inicial	Inicio del prototipo
v1.1	17-06-2025	Agustín, Sebastián, Carlos	Pantallas de registro y mejora visual inicial	Completar flujo de usuario
v1.2	20-06-2025	Eduardo, Agustín, Sebastián	Inclusión de sección de ayuda y contacto	Mejorar soporte al usuario
v1.3	20-06-2025	Carlos, Sebastián	Revisión de validación de campos	Prevenir errores
v1.4	21-06-2025	Sebastián, Eduardo, Carlos	Ajustes estéticos y navegación general	Homogeneidad visual
v1.5	22-06-2025	Agustín, Carlos, Eduardo	Flujo de salida de vehículos agregado	Implementación del caso de uso

## 10. CONCLUSIONES

El desarrollo del prototipo funcional para el sistema de salida de vehículos permitió representar de forma clara y efectiva los requerimientos del caso asignado, abordando tanto aspectos funcionales como no funcionales.

Mediante el uso de Figma como herramienta de prototipado, se logró una interfaz intuitiva, accesible y centrada en el usuario, cumpliendo con principios de usabilidad establecidos por Nielsen. La implementación de un control de versiones estructurado y el uso del historial de Figma facilitaron la trazabilidad y mejora continua del prototipo.

En conjunto con la elaboración del plan de pruebas basado en la norma ISO/IEC 25000, se logró validar la calidad del diseño propuesto, asegurando una base sólida para una futura implementación del sistema.

## 11. BIBLIOGRAFÍA

Nielsen, J. (1995). *10 Heurísticas de Usabilidad para el Diseño de Interfaces de Usuario*. Nielsen Norman Group.

<https://www.nngroup.com/articles/ten-usability-heuristics/>

Figma Inc. (2024). *Herramienta de Diseño Figma*. <https://www.figma.com>

Git-scm.com. (2024). *Acerca del Control de Versiones*.

<https://git-scm.com/book/es/v2/Empezando-Acerca-del-Control-de-Versiones>

Gobierno de Chile. (2024). *Portal oficial del Gobierno de Chile*. <https://www.gob.cl>

Servicio Agrícola y Ganadero (SAG). (2024). *Sitio oficial del SAG*. <https://www.sag.cl>

Servicio Nacional de Aduanas. (2024). *Portal del Servicio Nacional de Aduanas de Chile*. <https://www.aduana.cl>

Spring. (2024). *Documentación oficial de Spring Boot*.

<https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>