

**Sistema de Entrada de Vehículos a Chile (Aduana)
(DAS) Documento Arquitectura de Software
Versión 1.0**

Identificación de Documento

Identificación	
Proyecto	Sistema de Entrada de Vehículos a Chile (Aduana)
Versión	1.0.0
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 (GENERAL)	5
1.2. PROPÓSITO	5
1.3. ÁMBITO	5
1.4. DEFINICIONES, ACRÓNIMOS Y ABREVIACIONES	5
1.5. RESUMEN EJECUTIVO (GENERAL)	6
1.6. ARQUITECTURA DEL SISTEMA (GENERAL)	6
2. VISIÓN DEL SISTEMA (GENERAL)	6
2.1. DESCRIPCIÓN GENERAL DEL SISTEMA	6
2.2. OBJETIVOS DEL SISTEMA	6
2.3. REQUISITOS FUNCIONALES Y NO FUNCIONALES	7
2.4. SUPUESTOS Y DEPENDENCIAS	7
3. ESTILOS Y PATRONES ARQUITECTÓNICOS (GENERAL)	7
3.1. ESTILO ARQUITECTÓNICO ADOPTADO	7
3.2. JUSTIFICACIÓN DEL ESTILO SEGÚN EL CONTEXTO DEL SISTEMA	8
4. MODELO 4 +1 Y VISTAS ARQUITECTÓNICAS	8
4.1. VISTA DE ESCENARIO (GENERAL Y SALIDA VEHÍCULO O ENTRADA VEHÍCULO)	8
4.1.1. <i>Propósito (General)</i>	8
4.1.2. <i>Actores (General)</i>	8
4.1.3. <i>Diagrama general de casos de uso (General)</i>	9
4.1.4. <i>Diagrama de casos de uso específicos (salida vehículo o entrada vehículo)</i>	9
4.1.5. <i>Lista de casos de uso</i>	10
4.1.6. <i>Especificación de casos de uso (UN caso de uso principal de la salida vehículo/entrada vehículo)</i>	10
4.2. VISTA LÓGICA (SALIDA VEHÍCULO O ENTRADA VEHÍCULO)	12
4.2.1. <i>Propósito</i>	12
4.2.2. <i>Diagrama de clases</i>	12
4.2.3. <i>Descripción diagrama de clases</i>	12
4.3. VISTA DE IMPLEMENTACIÓN/DESARROLLO (SALIDA VEHÍCULO O ENTRADA VEHÍCULO)	14
4.3.1. <i>Propósito</i>	14
4.3.2. <i>Diagrama de componente</i>	14
4.3.3. <i>Descripción diagrama de componente</i>	14
4.3.4. <i>Diagrama de paquete</i>	14
4.3.5. <i>Descripción diagrama de paquete</i>	15
4.4. VISTA DE PROCESOS (SALIDA VEHÍCULO O ENTRADA VEHÍCULO)	15
4.4.1. <i>PROPÓSITO</i>	15
4.4.2. <i>DIAGRAMA DE ACTIVIDAD</i>	15
4.4.3. <i>DESCRIPCIÓN DIAGRAMA DE ACTIVIDAD</i>	16
4.5. VISTA FÍSICA (SALIDA VEHÍCULO O ENTRADA VEHÍCULO)	16

4.5.1. <i>Propósito</i>	16
4.5.2. <i>Diagrama de despliegue</i>	16
4.5.3. <i>Descripción diagrama de despliegue</i>	17
5. REQUISITOS DE CALIDAD (GENERAL)	17
5.1. PROPÓSITO	17
5.2. ATRIBUTOS DE CALIDAD	17
5.3. <i>Reglas y criterios de evaluación de calidad</i>	18
6. PRINCIPIOS DE DISEÑO APLICADOS	18
6.1. <i>Propósito</i>	18
6.2. PRINCIPIOS DE DISEÑO (POR EJEMPLO: ABSTRACCIÓN, ACOPLAMIENTO, COHESIÓN, ENCAPSULAMIENTO, MODULARIDAD)	19
7. PROTOTIPO	19
7.1. PROPÓSITO	19
7.2. MOCKUPS (IMÁGENES CON UNA BREVE DESCRIPCIÓN)	19
7.3. JUSTIFICAR HERRAMIENTAS DE PROTOTIPADO	24
8. EVALUACIÓN DE CALIDAD HEURÍSTICA DE NIELSEN	24
8.1. PROPÓSITO	24
8.2. LISTA DE VERIFICACIÓN	24
8.3. ANÁLISIS Y MÉTRICAS DE RESULTADOS	25
9. CONTROL DE VERSIONES	26
9.1. PROPÓSITO	26
9.2. CONTROL DE VERSIÓN UTILIZADO (JUSTIFICAR EL TIPO DE CONTROL DE VERSIÓN UTILIZADO (FECHA, SEMÁNTICA O SECUENCIAL)	26
9.3. JUSTIFICAR HERRAMIENTAS DE VERSIONAMIENTO	26
10. CONCLUSIONES	27
11. BIBLIOGRAFÍA	27
12. ANEXOS	28
12.1. REPOSITORIO GITHUB DEL PROYECTO	28
12.2. CAPTURA DE PANTALLA DEL REPOSITORIO	28
12.3. PLANILLA DE REQUERIMIENTOS	28
12.4. CARTA GANTT	29
12.5. PROTOTIPO FIGMA	29

1. INTRODUCCIÓN

1.1. Contexto del Problema (General)

La Aduana chilena enfrenta problemas críticos al momento del ingreso de vehículos en pasos fronterizos. Las largas filas, la validación manual de documentos y la falta de interoperabilidad entre organismos como SAG, PDI y Registro Civil provocan demoras, riesgos de error humano y mal uso de recursos. Se requiere modernizar el sistema para asegurar eficiencia, trazabilidad y seguridad.

1.2. Propósito

El propósito de este proyecto es diseñar un sistema informático web que permita a los ciudadanos declarar previamente su ingreso al país con vehículos, productos o mascotas, y que al mismo tiempo permita a los técnicos aduaneros validar dicha información de forma digital e integrada con servicios externos del Estado.

1.3. Ámbito

El sistema está enfocado exclusivamente en el ingreso terrestre de vehículos a Chile. No contempla salida ni tránsito, ni validación de personas. Incluye funcionalidades para ciudadanos, técnicos aduaneros y jefes operativos, abarcando los módulos de validación de vehículos, productos no equipaje, mascotas (perros/gatos), e informes estadísticos.

1.4. Definiciones, acrónimos y abreviaciones

ACRONIMO	DESCRIPCION
<i>CZE</i>	Certificado Zoosanitario de Exportación, requerido para ingreso de mascotas
<i>SAG</i>	Servicio Agrícola y Ganadero, entidad que regula el ingreso de productos y animales
<i>PDI</i>	Policía de Investigaciones, entidad que verifica autorización de ingreso
<i>API</i>	Application Programming Interface, permite integración con servicios externos
<i>MVC</i>	Modelo Vista Controlador, patrón arquitectónico usado en el sistema
<i>DAO</i>	Data Access Object, patrón para separar lógica de acceso a datos
<i>BD</i>	Base de Datos, sistema de almacenamiento de la información del sistema

1.5. Resumen ejecutivo (General)

El proyecto busca mejorar la eficiencia en el proceso de ingreso de vehículos al país mediante un sistema informático que digitaliza el formulario de declaración, automatiza validaciones documentales y genera informes. Utiliza tecnologías web, arquitectura modular, y se integra con APIs externas como Registro Civil, SAG y PDI.

1.6. Arquitectura del sistema (General)

(ej. vista de escenario, vista lógica, vista de desarrollo, vista de proceso, vista física)

La arquitectura se basa en el modelo 4+1 de vistas:

- Vista de Escenario: Modela los casos de uso del sistema, centrados en el ciudadano y el técnico aduanero.
- Vista Lógica: Representa las entidades del sistema y sus relaciones (clases como Ciudadano, Vehículo, Producto, etc.).
- Vista de Desarrollo: Muestra cómo está organizado el código en paquetes y componentes reutilizables.
- Vista de Proceso: Describe el flujo de actividades desde el ingreso de datos hasta la validación y notificación del estado.
- Vista Física: Detalla cómo se despliega el sistema en servidores, con integración a servicios externos.

2. VISIÓN DEL SISTEMA (General)

2.1. Descripción general del sistema

El sistema está diseñado como una plataforma web que permite a los ciudadanos que ingresan a Chile declarar previamente los vehículos, productos no considerados equipaje y mascotas que llevarán consigo. El sistema procesa esta información, valida los documentos mediante integración con APIs externas y entrega una interfaz para que los técnicos de aduana puedan verificar los antecedentes y aprobar o rechazar el ingreso. También incluye una interfaz para que el jefe operativo genere informes estadísticos sobre los ingresos registrados.

2.2. Objetivos del sistema

- Disminuir los tiempos de espera en los pasos fronterizos.
- Mejorar la trazabilidad y control documental de vehículos, productos y

mascotas que ingresan al país.

- Reducir el trabajo manual del personal aduanero mediante automatización de procesos.
- Integrar a través de servicios web las validaciones con Registro Civil, PDI y SAG.

2.3. Requisitos funcionales y no funcionales

- Formulario de ingreso de vehículos (con o sin vehículo de arrastre).
- Subida de documentación obligatoria (padrón, seguro internacional, poder notarial).
- Declaración de productos no equipaje y validación con SAG.
- Adjuntar y validar el certificado CZE para mascotas (perros y gatos).
- Interfaz de revisión técnica para validar cada ingreso.
- Generación automática de informes estadísticos para el jefe operativo.

2.4. Supuestos y dependencias

- El ciudadano debe contar con Clave Única para autenticarse en el sistema.
- Toda la documentación cargada debe tener una antigüedad máxima de 10 días.
- El sistema se conecta en tiempo real con servicios externos: Registro Civil (validación de datos vehiculares), PDI (autorizaciones de ingreso), y SAG (productos y mascotas).
- Se asume que los dispositivos utilizados en frontera tienen conexión estable a internet.

3. ESTILOS Y PATRONES ARQUITECTÓNICOS (General)

3.1. Estilo arquitectónico adoptado (ej. monolítico, microservicios, SOA, capas)

El sistema utiliza un estilo en capas (n-tier) combinado con el patrón Modelo-Vista-Controlador (MVC). Esta elección permite separar de manera lógica la interfaz de usuario (presentación), la lógica de negocio (controladores y validaciones) y el acceso a datos (modelo y DAO). También se estructura como una arquitectura modular, donde cada componente maneja una funcionalidad específica.

3.2. Justificación del estilo según el contexto del sistema

El enfoque en capas facilita la escalabilidad del sistema, ya que cada capa puede modificarse, mantenerse o escalarse de forma independiente. El uso de MVC permite desarrollar una plataforma web clara, organizada y de fácil mantenimiento. Dado que el sistema interactúa con múltiples actores (ciudadano, técnico, jefe operativo) y con servicios externos, esta arquitectura garantiza separación de responsabilidades, reutilización de componentes y desacoplamiento entre vistas y lógica de negocio.

3.3. Patrones de diseño aplicados (ej. patrón MVC, repositorio, etc.)

- MVC (Modelo-Vista-Controlador): estructura principal para separar lógica de presentación, lógica de control y modelo de datos.
- DAO (Data Access Object): patrón para el acceso a datos, separando consultas SQL del resto de la lógica.
- Fachada (Facade): utilizado para encapsular la comunicación con APIs externas (Registro Civil, PDI, SAG), simplificando su integración y desacoplando el sistema de su implementación.
- Singleton (opcional para servicios de acceso único como conexión a base de datos).

4. MODELO 4 +1 Y VISTAS ARQUITECTÓNICAS

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

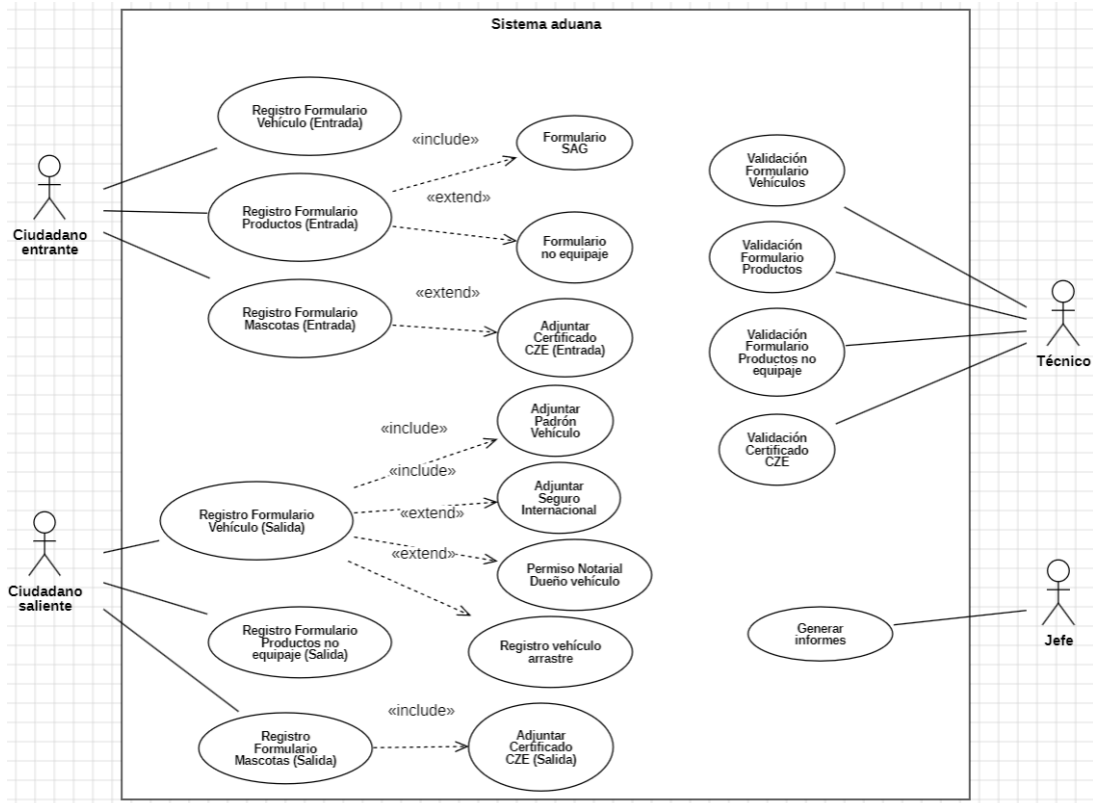
4.1.1. Propósito (General)

Representar las interacciones entre los usuarios del sistema (ciudadano, técnico en revisión y jefe operativo) con los principales procesos funcionales relacionados con el ingreso de vehículos a Chile. Esta vista permite comprender el alcance del sistema y delimitar los roles y funcionalidades.

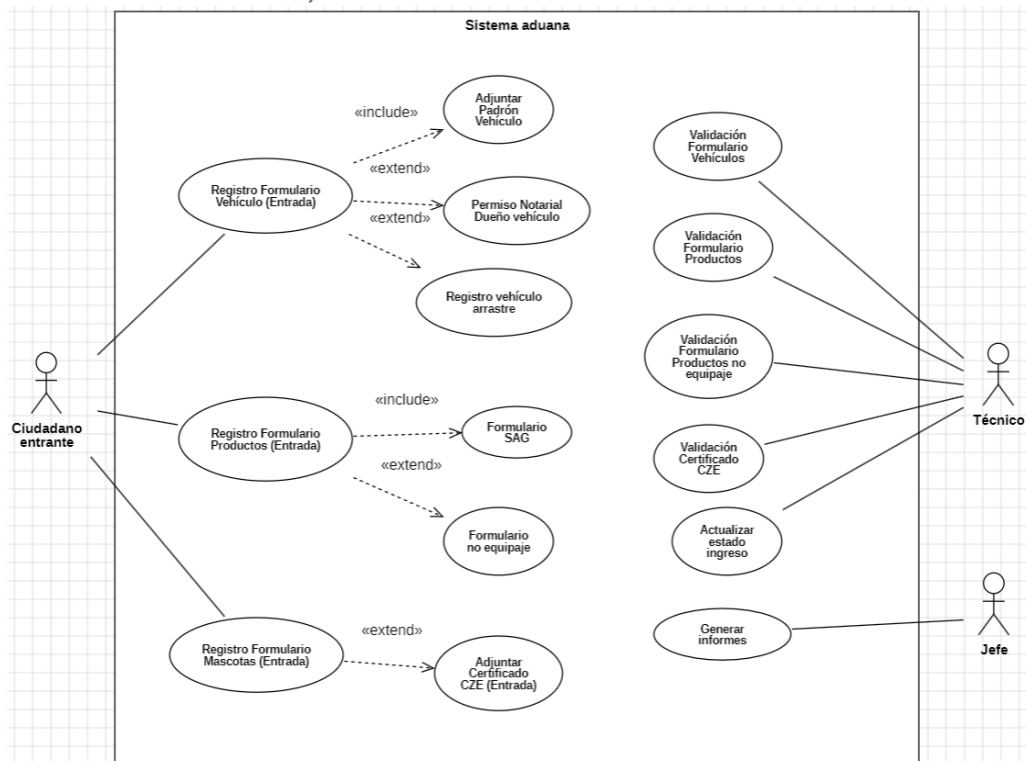
4.1.2. Actores (General)

- Ciudadano que ingresa a Chile: Declara información del vehículo, productos y mascotas mediante formulario web.
- Técnico en revisión no invasiva: Valida los datos ingresados por el ciudadano y decide si se aprueba o rechaza el ingreso.
- Jefe Operativo: Genera informes estadísticos a partir de los registros procesados.

4.1.3. Diagrama general de casos de uso (General)



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



4.1.5. Lista de casos de uso (salida vehículo o entrada vehículo)

Código	Nombre	Actores
CU-001-001	Registrar formulario vehículo (entrada)	Ciudadano
CU-001-002	Adjuntar padrón vehículo	Ciudadano
CU-001-003	Permiso notarial dueño del vehículo (si aplica)	Ciudadano
CU-001-004	Registro vehículo arrastre	Ciudadano
CU-001-005	Registrar formulario productos (entrada)	Ciudadano
CU-001-006	Registrar formulario SAG	Ciudadano
CU-001-007	Registrar formulario no equipaje	Ciudadano
CU-001-008	Registrar formulario mascotas (entrada)	Ciudadano
CU-001-009	Adjuntar certificado CZE (entrada)	Ciudadano
CU-001-010	Validar formulario vehículos	Técnico
CU-001-011	Validar formulario productos	Técnico
CU-001-012	Validar formulario productos no equipaje	Técnico
CU-001-013	Validar certificado CZE	Técnico
CU-001-014	Actualizar estado de ingreso	Técnico
CU-001-015	Generar informes	Jefe

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

Caso de Uso	Registrar entrada de vehículo	Identificador: CU-001-001
Actores	Ciudadano, Técnico	
Tipo	Primario	
Referencias	Ley de Aduanas, validación PDI, validación Registro Civil	
Precondición	El ciudadano debe estar autenticado y contar con Clave Única	
Postcondición	El sistema guarda la solicitud con su estado inicial como "Pendiente de validación"	
Descripción	El ciudadano declara los datos del vehículo, sube la documentación correspondiente y completa la información del viaje	
Resumen	Este caso permite registrar el vehículo antes de cruzar la frontera, enviando los datos al sistema para su validación técnica posterior	

CURSO NORMAL

Nro.	Ejecutor	Paso o Actividad
1	Ciudadano	Ingresa al sistema con Clave Única.
2	Sistema	Valida las credenciales del usuario.
3	Ciudadano	Completa formulario con datos del vehículo, conductor y viaje.
4	Ciudadano	Adjunta padrón, seguro internacional, y poder notarial si aplica.
5	Sistema	Verifica que los documentos no superen los 60 días de antigüedad.
6	Sistema	Registra el ingreso como solicitud "Pendiente de revisión técnica".

7	Técnico	Accede al formulario y visualiza la solicitud.
8	Técnico	Valida documentos e información declarada.
9	Técnico	Cambia el estado a Aceptado, Pendiente o Rechazado.

CURSO ALTERNATIVO

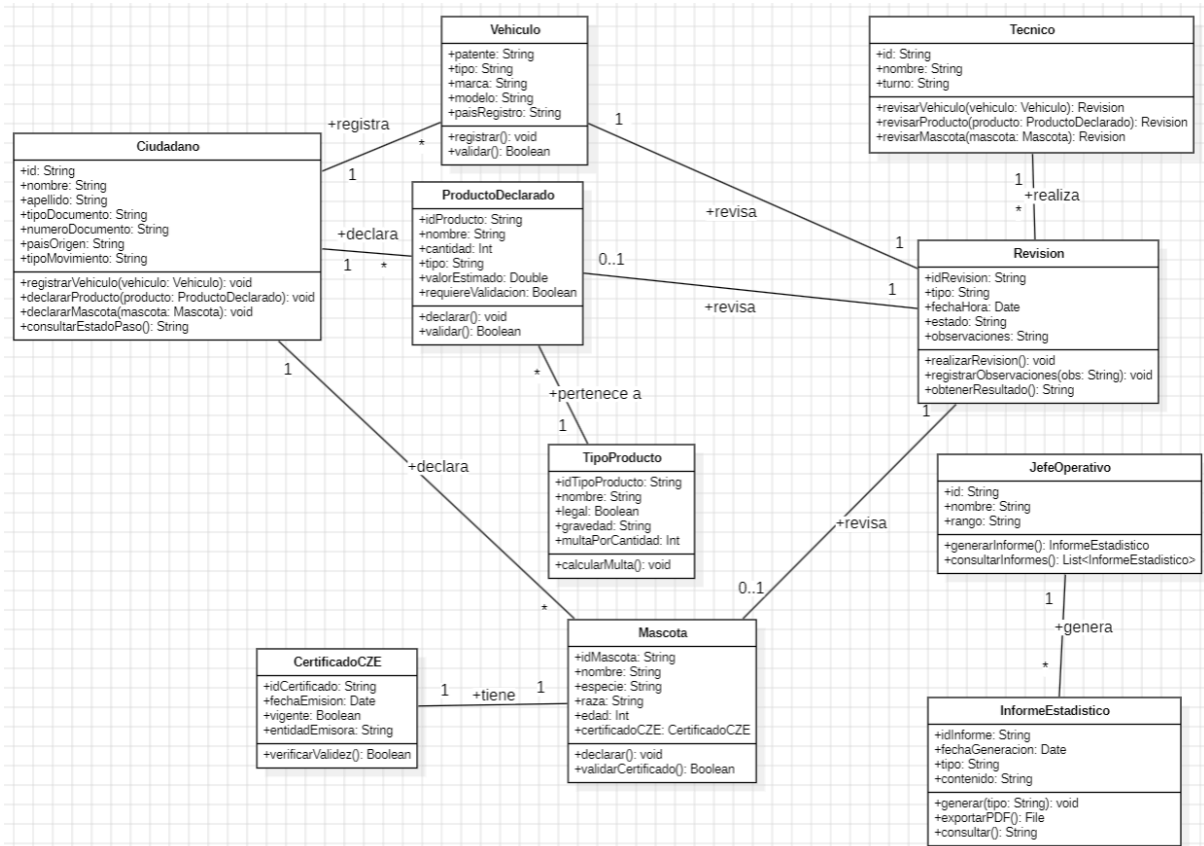
Nro.	Descripción de acciones alternas
4	Si el ciudadano no cuenta con el poder notarial, el sistema marca la solicitud como Incompleta.
5	Si el documento tiene más de 60 días, el sistema bloquea el envío y solicita uno actualizado.
8	Si el técnico detecta inconsistencias, puede dejar observaciones y marcar como "Pendiente".

4.2. VISTA LÓGICA (salida vehículo o entrada vehículo)

4.2.1. Propósito

La vista lógica tiene como objetivo representar la estructura interna del sistema mediante clases, atributos, métodos y relaciones entre objetos. Esto permite visualizar cómo se organizan los datos, cómo interactúan las entidades del dominio y cómo se construye la base para el desarrollo orientado a objetos.

4.2.2. Diagrama de clases



4.2.3. Descripción diagrama de clases

El sistema se organiza a partir de entidades clave del dominio aduanero, representadas mediante clases con relaciones claras y responsabilidades bien definidas:

- Ciudadano actúa como entidad principal que registra o declara elementos a ingresar: Vehículo, ProductoDeclarado y Mascota. Posee métodos para gestionar estas declaraciones y consultar el estado de su solicitud.
- Vehículo, ProductoDeclarado y Mascota son las entidades que requieren validación previa al ingreso. Cada una contiene atributos específicos de

identificación y métodos de registro y validación. Además, ProductoDeclarado se asocia a un TipoProducto, el cual define reglas legales y de multa por exceso.

- Mascota está relacionada con un CertificadoCZE, el cual debe validarse antes del ingreso. Este certificado contiene información sobre vigencia, fecha y entidad emisora.
- Técnico realiza la Revisión de los objetos declarados. Esta clase registra la observación, resultado y estado de la revisión, y se relaciona con los elementos revisables mediante una asociación polimórfica.
- Revisión encapsula la lógica del proceso de inspección, registrando fecha, estado y observaciones, y es el vínculo funcional entre técnico y objeto declarado.
- JefeOperativo es responsable de generar un InformeEstadístico, consolidando los datos obtenidos a partir de las revisiones. Este informe contiene métricas y estadísticas relevantes del sistema.

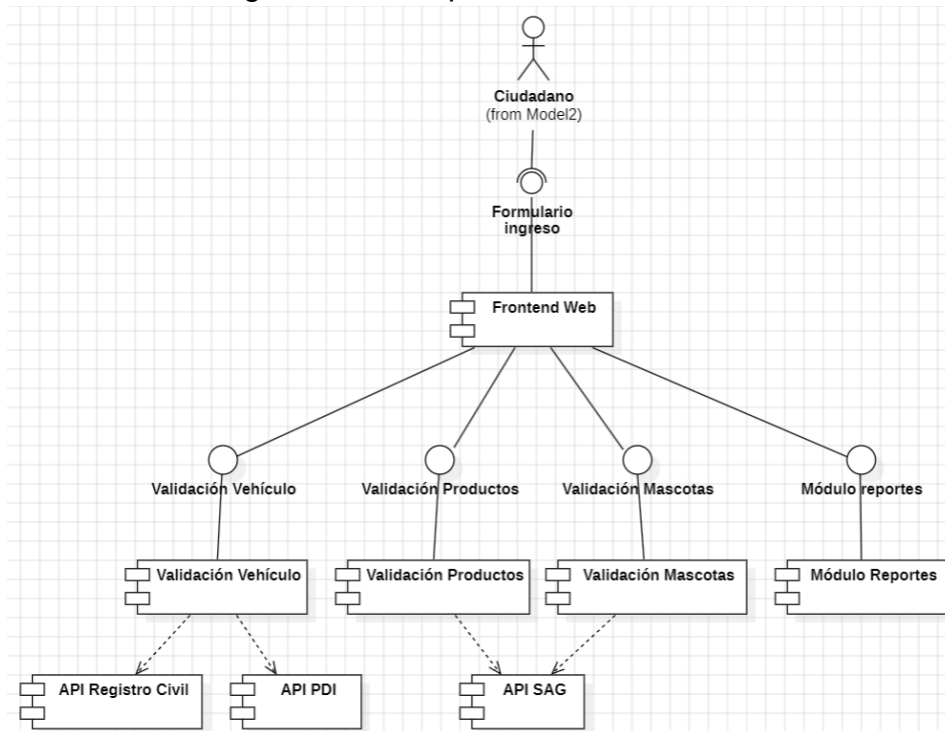
Cada clase incorpora atributos fundamentales y métodos clave como validar(), declarar(), registrar(), revisar(), y generarInforme(), lo que garantiza una alta cohesión, encapsulamiento adecuado y claridad funcional en el diseño orientado a objetos del sistema.

4.3.VISTA DE IMPLEMENTACIÓN/DESARROLLO (salida vehículo o entrada vehículo)

4.3.1. Propósito

Representar la organización física del software a través de sus componentes funcionales. Permite modelar cómo se dividen los módulos del sistema y cómo se comunican a través de interfaces, facilitando el trabajo modular y la reutilización.

4.3.2. Diagrama de componente

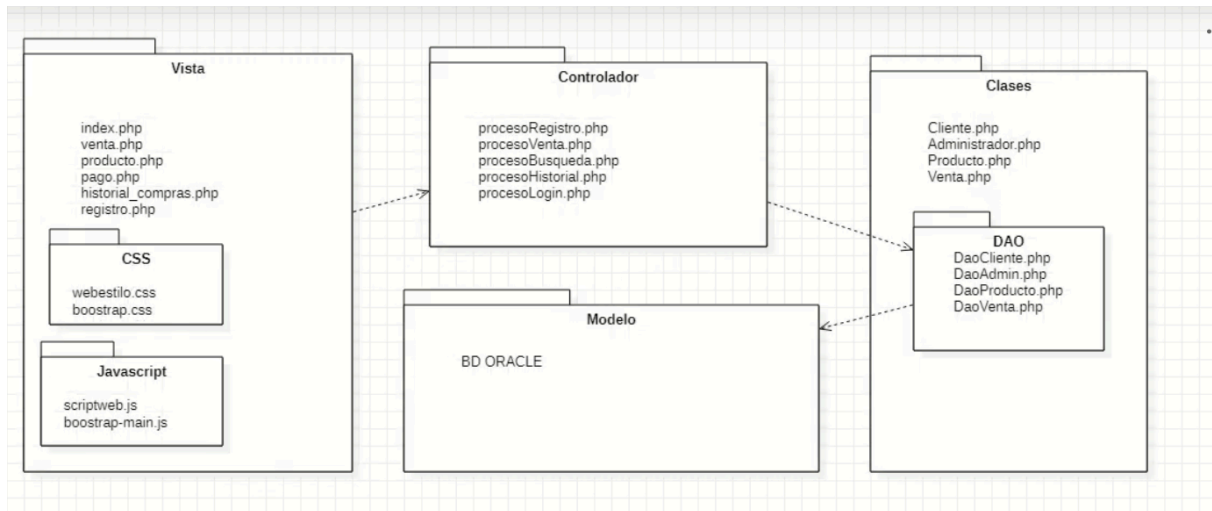


4.3.3. Descripción diagrama de componente

El sistema se estructura en torno al componente principal “Frontend Web”, que centraliza la interacción con el ciudadano a través del formulario de ingreso. Desde allí, se delegan funciones a componentes especializados según el tipo de trámite: Validación de Vehículo, Productos, Mascotas y Reportes.

Cada uno de estos módulos se comunica con APIs externas específicas (Registro Civil, PDI, SAG) mediante componentes de fachada que encapsulan las llamadas, desacoplando así el sistema interno de los servicios externos. Esta arquitectura favorece la modularidad, facilita la integración de nuevas validaciones y mejora el mantenimiento del sistema. La presencia de un módulo de reportes permite consolidar información relevante para seguimiento o estadísticas.

4.3.4. Diagrama de paquete



4.3.5. Descripción diagrama de paquete

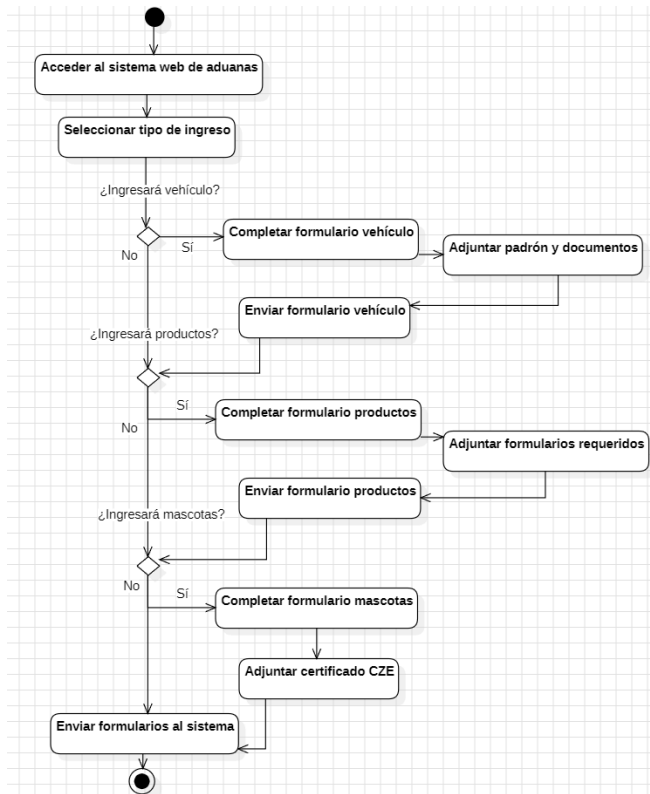
El sistema está estructurado siguiendo el patrón de arquitectura Modelo-Vista-Controlador (MVC), lo que favorece una clara separación de responsabilidades. La vista incluye archivos PHP relacionados con la interfaz del usuario, así como archivos CSS y JavaScript que definen el estilo y la interacción del sistema. El controlador se compone de scripts PHP responsables de manejar las solicitudes del usuario y coordinar la lógica entre vista y modelo. El modelo, conectado a la base de datos Oracle, se encarga de la persistencia de los datos. Finalmente, se incluyen paquetes de clases que representan las entidades del sistema (Cliente, Administrador, Producto y Venta), junto con sus respectivos objetos de acceso a datos (DAO). Esta organización modular mejora el mantenimiento, la escalabilidad y permite una integración eficiente de nuevas funcionalidades a futuro.

4.4. VISTA DE PROCESOS (salida vehículo o entrada vehículo)

4.4.1. Propósito

Modelar el flujo dinámico del sistema. Esta vista permite identificar el orden en que ocurren las actividades, la participación de cada actor, las decisiones condicionales y la secuencia de validaciones que atraviesa cada declaración.

4.4.2. Diagrama de actividad



4.4.3. Descripción diagrama de actividad

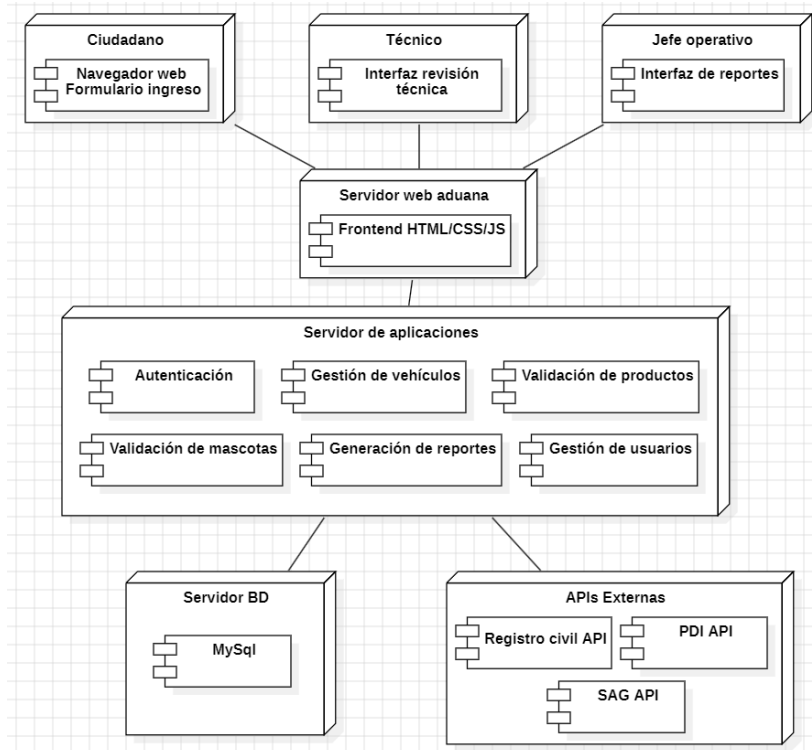
El diagrama inicia con el acceso al sistema web de Aduanas y la selección del tipo de ingreso por parte del usuario. A continuación, se presentan decisiones secuenciales donde el sistema consulta si se ingresarán vehículos, productos o mascotas. Si el usuario indica que sí a alguna de estas opciones, se habilitan los formularios correspondientes junto con la opción de adjuntar los documentos necesarios: padrón del vehículo, formularios requeridos o certificado CZE, según corresponda. Tras completar cada formulario y adjuntar la documentación, se procede al envío individual de cada uno. Finalmente, una vez completadas todas las secciones aplicables, el sistema permite enviar todos los formularios al sistema para su revisión.

4.5. VISTA FÍSICA (salida vehículo o entrada vehículo)

4.5.1. Propósito

La vista física tiene como finalidad mostrar cómo se despliega el sistema en una infraestructura tecnológica concreta, identificando los distintos nodos, servidores y conexiones entre componentes físicos y lógicos. Esta vista es esencial para comprender la distribución real del sistema, su escalabilidad, rendimiento y puntos de integración con servicios externos.

4.5.2. Diagrama de despliegue



4.5.3. Descripción diagrama de despliegue

El sistema se ejecuta como una aplicación web distribuida. El ciudadano accede al formulario desde cualquier dispositivo conectado, que se comunica con el servidor web. Este redirige las peticiones al backend, donde se ejecutan los procesos de validación y lógica de negocio.

La base de datos almacena de forma segura todas las entidades del sistema, mientras que la comunicación con APIs externas permite validar documentos oficiales (como el padrón del vehículo o el certificado CZE) sin intervención manual.

Además, el técnico aduanero y el jefe operativo acceden desde terminales internos conectados a la misma aplicación, pero con interfaces específicas y control de roles.

5. REQUISITOS DE CALIDAD (General)

5.1. Propósito

Esta sección define los atributos de calidad esperados del sistema, en línea con el modelo ISO/IEC 25010. Establece cómo se medirá y garantizará la calidad del producto software, considerando factores como usabilidad, rendimiento, seguridad, mantenibilidad, entre otros.

5.2. Atributos de calidad (por ejemplo: Usabilidad, Accesibilidad (WCAG), Rendimiento, Mantenibilidad, Seguridad Portabilidad)

ATRIBUTO DE CALIDAD	DESCRIPCIÓN	JUSTIFICACIÓN
Usabilidad	El sistema debe ser intuitivo, fácil de usar y comprensible para cualquier usuario.	El sistema será utilizado por ciudadanos comunes, por lo tanto debe minimizar la curva de aprendizaje.
Rendimiento	El tiempo de respuesta debe ser inferior a 2 segundos en operaciones clave.	Las validaciones deben ser rápidas para evitar demoras y acumulación de personas en frontera.
Seguridad	Se requiere autenticación, validación de usuarios y protección de datos.	El sistema trata con datos personales sensibles (RUT, certificados, etc.), por lo que debe ser seguro.
Mantenibilidad	La arquitectura debe permitir actualizaciones y corrección de errores fácilmente.	El sistema puede evolucionar (nuevas reglas o validaciones), por lo que debe ser fácilmente modificable.
Portabilidad	El sistema debe ser accesible desde distintos dispositivos (PC y móviles).	Se busca que el ciudadano pueda acceder al formulario desde su hogar o celular antes de viajar.

5.3. Reglas y criterios de evaluación de calidad

Cómo se medirá el cumplimiento de cada atributo (ej. tiempo de carga < 2 seg, puntuación Nielsen de usabilidad, cumplimiento WCAG nivel AA, etc.).

Herramientas o métodos que se utilizarán(pruebas de carga, pruebas heurísticas, inspecciones, validación con usuarios, etc)

- Usabilidad: Evaluada mediante pruebas heurísticas y validación con usuarios. Se aplicarán principios de Nielsen para verificar claridad, consistencia y visibilidad del estado del sistema.
- Rendimiento: Se realizarán pruebas de carga con simulación de múltiples solicitudes simultáneas. Se medirá el tiempo de respuesta, el tiempo de procesamiento de formularios y generación de reportes.
- Seguridad: Se asegurará que todas las conexiones sean HTTPS, los documentos estén protegidos, y el acceso esté restringido por rol. Se evaluará mediante pruebas de vulnerabilidad y control de acceso.
- Mantenibilidad: Se verificará la modularidad del código, el cumplimiento de principios SOLID, y la existencia de documentación técnica.
- Portabilidad: Se harán pruebas de compatibilidad en navegadores modernos (Chrome, Firefox, Edge) y en dispositivos móviles Android/iOS.

6. PRINCIPIOS DE DISEÑO APLICADOS

6.1. Propósito

El objetivo de esta sección es evidenciar cómo se aplicaron principios fundamentales de la Ingeniería de Software en el diseño del sistema, con el fin de lograr una arquitectura robusta, mantenible y de calidad, que permita escalar el sistema y adaptarse a cambios futuros.

6.2. Principios de diseño (por ejemplo: abstracción, acoplamiento, cohesión, encapsulamiento, modularidad)

PRINCIPIO	DESCRIPCIÓN	APLICACIÓN EN EL SISTEMA
Cohesión	Cada módulo o clase tiene una única responsabilidad bien definida.	Los servicios están diseñados para realizar tareas específicas y no múltiples funciones
Bajo acoplamiento	Las clases y componentes están lo menos dependientes entre sí.	Los módulos (vehículos, productos, mascotas) están desacoplados y se comunican por interfaces.
Encapsulamiento	La lógica interna de cada clase está protegida del acceso externo directo.	La validación y manipulación de datos se hace a través de métodos definidos, protegiendo los atributos.
Modularidad	El sistema está dividido en componentes reutilizables y autónomos.	Las funcionalidades están separadas por paquetes y componentes: autenticación, revisión, reportes, etc.
Abstracción	Se ocultan detalles internos y se exponen solo funcionalidades relevantes.	Se aplican interfaces para declarar mascotas, productos y vehículos sin revelar la implementación técnica.


7. PROTOTIPO


7.1. Propósito

El propósito del prototipo es representar de manera visual y funcional la interfaz del sistema antes de su implementación definitiva. A través del prototipado, se pueden validar ideas de diseño, identificar posibles problemas de usabilidad y obtener retroalimentación temprana, lo que reduce riesgos y costos en etapas posteriores del desarrollo. Este proceso permite simular la interacción del usuario con el sistema y evaluar la estructura, el flujo y la estética de las pantallas propuestas.

7.2. Mockups (imágenes con una breve descripción)

1. Menú principal: Pantalla de inicio donde el usuario puede iniciar sesión con su cuenta y contraseña de clave única, para luego poder elegir entre registrar una nueva solicitud de ingreso o consultar el estado de solicitudes anteriores.





Menú principal

RUN

Value

Clave única

Value

Iniciar sesión

[Recupera tu clave única](#)


[Solicita tu clave única](#)

Menú principal

Solicitud de ingreso >

Ver solicitudes >

2. Formulario de Ingreso: Formulario inicial donde el usuario ingresa sus datos personales como RUN, número de documento, nombre, apellido y país de origen.



Formulario de Ingreso

RUN

Value

Tipo documento

Seleccionar

Número de documento

Value

Nombre

Value

Apellido

Value


País origen

Seleccionar

Continuar

Atrás

3. Datos del vehículo: Sección donde el usuario puede ingresar información del vehículo, como patente, chasis, marca, modelo, pasajeros y condición del conductor.



Datos vehículo

Patente

Chasis/VIN

Marca

Modelo

Pasajeros


Vehículo de arrastre
☐ Sí ☒ No

Condición del conductor
☐ Propietario ☒ Autorizado

[Continuar](#)

[Atrás](#)

4. Archivos y declaraciones: Sección donde el usuario puede subir documentos requeridos, como el padrón del vehículo, acceder al formulario SAG y registrar mascotas que viajan.



Archivos y declaraciones

Autorización vehículo
[Subir](#) autorizacion_vehiculo.png

Padrón vehículo
[Subir](#) padron_vehiculo.png

Formulario SAG
[Ir al formulario](#) ✓


Mascotas

Nombre	Tipo
Max	Perro Editar

[Enviar solicitud](#)

[Atrás](#)

5. Registro de mascota: Formulario específico para ingresar información de mascotas, incluyendo nombre, especie, raza, edad y certificado CZE.



Registro de mascota

Nombre

Especie

Raza

Edad

Certificado CZE
[Subir](#) certificado_cze.png

[Atrás](#)

6. Confirmación de envío: Pantalla que informa al usuario que su solicitud fue enviada correctamente, permitiéndole continuar con el proceso.



Solicitud enviada
con éxito

Continuar

7. Consulta de solicitudes: Vista que muestra el historial de solicitudes ingresadas por el usuario, incluyendo su fecha, hora y estado actual (aceptado, pendiente o denegado).



Solicitudes

ID	Fecha	Hora	Estado
1	15/06/2025	12:00	Denegado
2	20/06/2025	9:30	Aceptado
3	30/06/2025	18:00	Pendiente

Atras

8. Login: Pantalla de inicio de sesión donde el técnico o jefe debe ingresar su nombre de usuario y contraseña para acceder al sistema.



LOGIN

Usuario	<input type="text"/>
Value	
Contraseña	<input type="password"/>
Value	
<input type="button" value="Iniciar sesión"/>	

9. Listado de Solicitudes: Vista principal del sistema tras iniciar sesión, donde se muestran todas las solicitudes realizadas. Cada registro incluye el RUN del solicitante, su nombre, fecha y hora de solicitud, estado actual (Aceptado, Denegado o Pendiente) y un botón para visualizar más detalles.

Solicitudes					
RUN	Nombre	Fecha	Hora	Estado	
15.458.245-1	Juan Carlos Bottoque	1/10/2025	11:15	Derogado	Ver
15.458.245-1	Juan Carlos Bottoque	20/06/2025	9:30	Aceptado	Ver
15.458.245-1	Juan Carlos Bottoque	20/06/2025	18:50	Pendiente	Ver

10. Detalle de Solicitud: Interfaz dirigida al técnico, presenta toda la información relevante de una solicitud: datos del solicitante y vehículo, archivos adjuntos (padrón y formulario SAG), mascotas asociadas y opciones para aprobar, rechazar o dejar en revisión.

ID solicitud

1

Nº Documento solicitante

15.458.245-1

RUN solicitante

15.458.245-1

Nombre solicitante

Juan Carlos

Apellido solicitante

Bottoque

País origen

Chile

Pariente vehículo

Conductor

VIN

YV00H4LJ05H4N109190

Marca

Sungyoung

Modelo

Sant

Pasajeros

4

Vehículo de arrendo

No

Condición del conductor

Automático

Autenticación vehículo



Padrón vehículo



Formulario SAG

[Ver formulario](#)

Mascotas

Nombre

Max

Tipo

Perrito

Estado

Derogado

[Aprobar](#)
[Rechazar](#)
[Revisar](#)

11. Detalle de Mascota: Ventana emergente que despliega los datos completos de una mascota: nombre, especie, raza, edad y espacio para cargar o visualizar el certificado CZE.

Nombre mascota

Max

Especie mascota

Perrito

Raza mascota

Tanque chileno

Edad mascota

3

Certificado CZE



12. Pantalla de Informes: Vista que permite al jefe consultar informes por año y mes. Muestra una tabla con los registros disponibles y un botón “Ver” para acceder al informe detallado del periodo seleccionado.



13. Informe Mensual: Pantalla que presenta un resumen estadístico de las solicitudes procesadas en el mes seleccionado, incluyendo solicitudes aceptadas, rechazadas, pendientes, causas de rechazo y rendimiento operativo diario.

Informe Junio 2025		
RESUMEN Total de solicitudes registradas 100 Solicitudes aceptadas 45 (45.0%) Solicitudes rechazadas 55 (55.0%) Solicitudes pendientes 0 (0.0%) SOLICITUDES PENDIENTES Solicitudes pendientes de nuevo análisis 0 (0.0%) SOLICITUDES RECHAZADAS Solicitudes rechazadas por causas de rechazo 55 (55.0%) Solicitudes rechazadas por causas de rechazo 0 (0.0%) Solicitudes rechazadas por causas de rechazo 0 (0.0%)	SOLICITUDES REGISTRADAS Solicitudes registradas 100 Documento de identidad verificado 10 Documento de identidad no verificado 12 Otros de rechazo previo 4 Otros 16 RECHAZADO Solicitudes rechazadas por causas de rechazo 17 Otros causas rechazadas 15 (causas de rechazo) Otros causas rechazadas 10 (causas de rechazo) Otros causas rechazadas 10 (causas de rechazo)	RENDIMIENTO OPERATIVO Rendimiento operativo 100 Solicitudes diarias aceptadas (por día 1000) 10.2 Solicitudes diarias rechazadas (por día 1000) 10.8 Rendimiento promedio de solicitud 1.7 días hábiles

7.3. Justificar herramientas de prototipado

Se utilizó Figma como herramienta principal para el diseño de prototipos debido a su capacidad de trabajo colaborativo en línea, su interfaz intuitiva y sus potentes funciones de diseño y prototipado interactivo. Figma permite crear mockups de alta fidelidad, compartílos fácilmente con otros miembros del equipo y recibir comentarios en tiempo real. Además, su enfoque basado en la nube evita problemas de compatibilidad y facilita la iteración constante en el diseño.

8. EVALUACIÓN DE CALIDAD HEURÍSTICA DE NIELSEN

8.1. Propósito

El propósito de esta evaluación es identificar fortalezas y debilidades en la interfaz del sistema, aplicando los diez principios heurísticos de usabilidad definidos por Jakob Nielsen. Esta metodología permite detectar problemas potenciales que podrían afectar la experiencia del usuario, estimar su nivel de severidad y proponer recomendaciones concretas de mejora. Su aplicación contribuye a asegurar una interacción eficiente, intuitiva y satisfactoria con el sistema, incluso antes de realizar pruebas con usuarios reales.

8.2. Lista de verificación

Para llevar a cabo la evaluación, se utilizó una tabla de verificación basada en los principios de Nielsen. En ella, cada criterio fue evaluado considerando si se cumple (✓) o no (X), acompañado de observaciones que justifican la decisión y un nivel de gravedad estimado para cada posible problema identificado. Esta lista permite una revisión sistemática y estructurada de la interfaz, asegurando que se consideren todos los aspectos fundamentales de la usabilidad.

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)?	✓	El sistema notifica cuando una solicitud ha sido enviada y aprobada.	Baja
2	Correspondencia entre el sistema y el mundo real	¿La terminología y flujos se relacionan con el lenguaje y lógica del usuario?	✓	La terminología utilizada (vehículo, revisión, mascota) se adapta al lenguaje del usuario común.	Baja
3	Control y libertad del usuario	¿El usuario puede deshacer/repertir acciones fácilmente?	✓	El usuario puede modificar entradas antes del envío definitivo y cancelar operaciones.	Baja
4	Consistencia y estándares	¿Se mantiene un diseño coherente entre pantallas, botones y mensajes?	✓	Se mantiene un diseño visual coherente en todas las interfaces del sistema web.	Baja
5	Prevención de errores	¿El diseño evita que ocurran errores antes de que sucedan?	X	No se evidencian mensajes preventivos ante posibles errores de ingreso de datos incorrectos.	Alta
6	Reconocimiento mejor que recuerdo	¿Las opciones y funciones son visibles sin que el usuario deba recordar información?	✓	El sistema muestra menús desplegables y opciones visibles sin tener que memorizar códigos.	Baja
7	Flexibilidad y eficiencia de uso	¿Permite atajos o personalización para usuarios avanzados?	X	No hay funcionalidades avanzadas como autocompletado o accesos rápidos para usuarios frecuentes.	Media
8	Diseño estético y minimalista	¿La interfaz evita información innecesaria o ruido visual?	✓	Interfaz limpia con distribución clara de elementos. No hay elementos visuales innecesarios.	Baja
9	Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores	¿Los mensajes de error son claros, comprensibles y ofrecen solución?	X	No se identifican mensajes claros de error en caso de fallas o datos incorrectos.	Alta
10	Ayuda y documentación	¿Existe ayuda accesible, clara y orientada a la tarea cuando el usuario lo necesita?	X	El prototipo no incluye una sección de ayuda o documentación accesible que oriente al usuario sobre cómo utilizar el sistema.	Media

8.3. Análisis y métricas de resultados

La evaluación heurística realizada con base en los 10 principios de usabilidad de Nielsen permitió identificar fortalezas y debilidades en la interfaz del sistema. Cada principio fue examinado mediante criterios de evaluación, observaciones de la interfaz y una categorización de la gravedad de los problemas detectados. A continuación, se resumen los principales hallazgos:

- **Cumplimiento parcial:** De los 10 principios evaluados, 6 fueron cumplidos satisfactoriamente (✓) y 4 presentaron deficiencias (X). Esto sugiere una interfaz con una base sólida en usabilidad, pero con áreas específicas que deben ser reforzadas para mejorar la experiencia del usuario.
- **Tipo y gravedad de los problemas:** Los problemas identificados incluyen dos de gravedad alta, uno de gravedad media y uno de gravedad baja. Si bien ninguno de ellos impide el funcionamiento del sistema, sí pueden afectar la comprensión, eficiencia y autonomía del usuario, especialmente en contextos menos guiados o con errores de ingreso de datos.
- **Áreas de mejora observadas:**

- Incorporar validaciones preventivas antes del envío de formularios, para reducir la posibilidad de errores por parte del usuario y mejorar la robustez del sistema.
- Agregar funcionalidades que aumenten la eficiencia para usuarios frecuentes, como autocompletado de datos, atajos o preferencias personalizadas.
- Diseñar mensajes de error más claros y orientativos, que indiquen no solo qué falló, sino también cómo corregirlo, fomentando una recuperación efectiva.
- Implementar una sección de ayuda o guía contextual integrada, que oriente al usuario durante el uso del sistema, especialmente en tareas menos intuitivas o críticas.
- Métrica cualitativa general: La interfaz del sistema evidencia un diseño mayormente coherente, claro y funcional, con puntos destacables como la consistencia visual, el uso de lenguaje comprensible y la retroalimentación básica sobre acciones exitosas. No obstante, fortalecer la prevención de errores, la asistencia al usuario y la eficiencia para usuarios avanzados resultará clave para alcanzar un mayor estándar de usabilidad.

9. CONTROL DE VERSIONES

9.1. Propósito

El propósito del control de versiones en un proyecto de software es llevar un seguimiento organizado y estructurado de los cambios realizados a lo largo del desarrollo. Permite identificar con precisión qué modificaciones se han efectuado, cuándo se hicieron, quién las realizó y por qué. Esta práctica es fundamental para mantener la integridad del proyecto, facilitar el trabajo colaborativo, recuperar versiones anteriores en caso de errores y documentar la evolución del sistema.

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

Para este proyecto se utilizó un control de versión semántico, basado en el esquema MAJOR.MINOR.PATCH, donde cada número refleja un tipo de cambio específico: cambios mayores que afectan la compatibilidad, mejoras menores que agregan funcionalidades sin romper lo existente, y parches que corrigen errores. Este sistema fue elegido por su claridad y por ser ampliamente adoptado en proyectos de desarrollo, lo cual facilita la comprensión de las actualizaciones tanto para el equipo como para terceros.

9.3. Justificar herramientas de versionamiento

Se utilizó Git como herramienta de control de versiones, debido a su eficacia, flexibilidad y capacidad para trabajar de forma distribuida. Git permite llevar un historial detallado de los cambios, facilitar la colaboración entre distintos integrantes del equipo, gestionar ramas para el desarrollo paralelo y resolver conflictos de manera eficiente. La herramienta fue utilizada en conjunto con la plataforma GitHub, que proporciona una interfaz amigable para la gestión de repositorios, seguimiento de incidencias y revisión de código, optimizando así el flujo de trabajo y asegurando la trazabilidad del desarrollo.

10. CONCLUSIONES

El desarrollo del sistema para la entrada de vehículos a Chile demuestra cómo la Ingeniería de Software, apoyada en estándares y buenas prácticas de modelamiento UML, puede resolver problemas reales de eficiencia y control en procesos críticos como el aduanero. Mediante el uso de diagramas de casos de uso, clases, componentes, paquetes, despliegue y actividades, fue posible representar con claridad tanto la funcionalidad esperada como la estructura interna y externa del sistema.

Se logró una arquitectura modular, segura y fácilmente mantenible, que permite integrar servicios externos (como SAG, PDI y Registro Civil) y mejorar la experiencia del usuario final. Asimismo, se aplicaron principios sólidos de diseño (como cohesión, bajo acoplamiento y modularidad), y se definieron atributos de calidad que garantizan usabilidad, rendimiento y confiabilidad.

Además, se diseñó una interfaz intuitiva y responsiva, que contempla pantallas como login, listado de solicitudes y visualización detallada de cada solicitud, incluyendo los antecedentes del solicitante, sus mascotas, el padrón del vehículo y los formularios requeridos. Este diseño busca facilitar el trabajo de los funcionarios y reducir errores administrativos.

La incorporación de enlaces al formulario SAG y la visualización directa de certificados (como el CZE para mascotas) permiten una verificación rápida y eficiente de los requisitos exigidos por normativa, agilizando la toma de decisiones por parte del personal autorizado.

Finalmente, el enfoque adoptado promueve una solución escalable, que podría extenderse a otros puntos de control fronterizo o integrarse con nuevas instituciones, fortaleciendo la trazabilidad y transparencia de los procesos asociados al ingreso de personas, vehículos y mascotas al país.

11. BIBLIOGRAFÍA

Pressman, R. S. (2010). *Ingeniería del Software. Un enfoque práctico* (7ma Ed.). McGraw-Hill.

ISO/IEC 25010:2011. *Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE)*.

Kruchten, P. (1995). *Architectural Blueprints—The “4+1” View Model of Software Architecture*.

Documentación oficial de SAG, PDI y Registro Civil de Chile.

12. ANEXOS

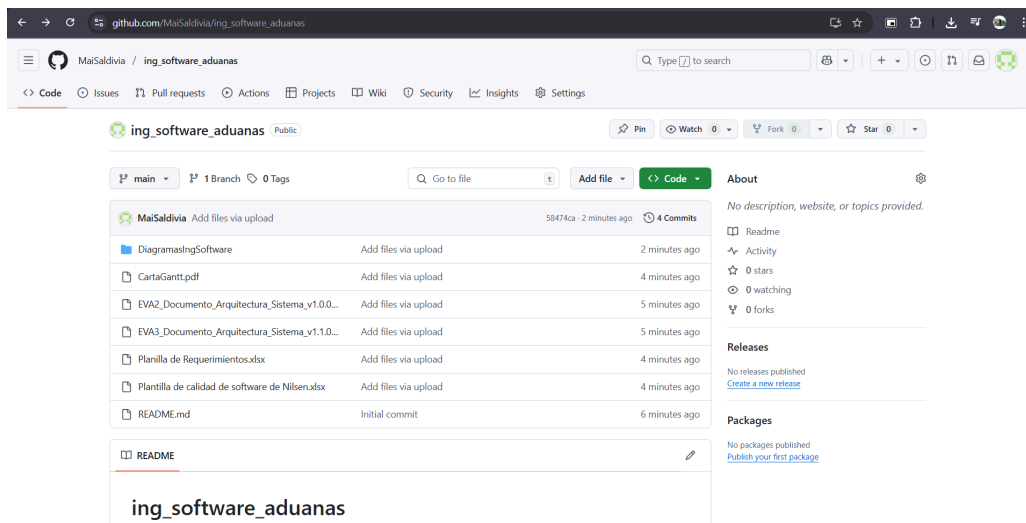
12.1. Repositorio GitHub del Proyecto

Durante el desarrollo del sistema, se utilizó un repositorio en GitHub para almacenar los documentos generados a lo largo del semestre. Esto permitió mantener un control de versiones y asegurar la trazabilidad del trabajo.

Enlace al repositorio: https://github.com/MaiSaldivia/ing_software_aduanas

12.2. Captura de pantalla del repositorio

A continuación, se muestra una imagen del repositorio en GitHub donde se puede visualizar la estructura de carpetas y archivos subidos:



12.3. Planilla de requerimientos

Ámbito Funcional	[R-N°]	[Nombre del Requerimiento]	Tipo Requerimiento	Actores y Roles Relacionados	[Descripción corta del requerimiento]	Estado
Gestión de Vehículos	RF-01	Registro Entrada/Salida de Vehículo	Funcional Usuario	Ciudadano, Funcionario Aduanas	Registrar vehículos que ingresan o salen del país por aduanas.	Análisis
Gestión de Productos	RF-02	Declaración de Productos en Formulario	Funcional Usuario	Ciudadano, Funcionario Aduanas	Permite declarar productos acompañantes o en el vehículo.	Análisis
Gestión de Mascotas	RF-03	Ingreso de Mascotas con Certificado	Funcional Usuario	Ciudadano, Funcionario Aduanas	Permite declarar mascotas con el certificado correspondiente.	Análisis
Documentación Vehicular	RF-04	Adjuntar Documentación Vehicular	Funcional Usuario	Ciudadano, Funcionario Aduanas	Adjuntar documentación como padrón, seguro y permiso notarial.	Análisis
Certificación CZE	RF-05	Adjuntar Certificado CZE	Funcional Usuario	Ciudadano, Técnico	Cargar el Certificado CZE obligatorio para validar ingreso.	Análisis
Validación de Trámites	RF-06	Validación de Formularios	Funcional Sistema	Técnico	Validar que los formularios estén completos y correctos.	Análisis
Registro de Ingreso	RF-07	Actualizar Estado de Ingreso	Funcional Sistema	Técnico	Actualizar el estado del ciudadano tras validaciones.	Análisis
Formularios de Declaración	RF-08	Formulario SAG y No Equipaje	Funcional Usuario	Ciudadano, Técnico	Formulario complementario de productos SAG o no equipaje.	Análisis
Control de Técnicos	RF-09	Validación de Técnicos	Funcional Sistema	Técnico	El técnico valida formularios y adjuntos recibidos.	Análisis
Reportes Estadísticos	RF-10	Generación de Informes	Funcional Sistema	Jefe Operativo	Permite generar reportes y estadísticas del sistema.	Análisis
Seguridad y Acceso	RNF-01	Gestión de Usuarios y Accesos	No Funcional	Todos los Usuarios	Control de acceso y autenticación de usuarios.	Levantamiento
Interfaz de Usuario	RNF-02	Interfaz Intuitiva y Accesible	No Funcional	Todos los Usuarios	Diseño intuitivo con ayuda y navegación clara.	Levantamiento
Rendimiento del Sistema	RNF-03	Respuesta Ágil del Sistema	No Funcional	Todos los Usuarios	Tiempo de respuesta bajo en validaciones y registros.	Levantamiento
Adaptabilidad	RNF-04	Adaptación a Dispositivos y Navegadores	No Funcional	Usuarios Finales	Interfaz adaptable a distintos navegadores y pantallas.	Levantamiento
Compatibilidad	RNF-05	Integración con Plataformas Externas	No Funcional	Sistema de Migraciones, Aduana	Permitir interoperabilidad con sistemas de migración.	Levantamiento
Disponibilidad	RNF-06	Alta Disponibilidad del Servicio	No Funcional	Todos los Usuarios	El sistema debe estar disponible 24/7 sin interrupciones.	Levantamiento
Portabilidad	RNF-07	Acceso desde Dispositivos Móviles	No Funcional	Pasajeros, Funcionarios	Sistema accesible desde teléfonos o tablets.	Levantamiento
Escalabilidad	RNF-08	Escalabilidad de Consultas Simultáneas	No Funcional	Administradores del Sistema	Manejo de grandes volúmenes sin pérdida de rendimiento.	Levantamiento
Integración de Sistemas	RNF-09	Interoperabilidad con Sistemas Externos	No Funcional	Sistema Externo, Aduana	Integración fluida entre plataformas aduaneras y externas.	Levantamiento
Mantenibilidad	RNF-10	Facilidad de Mantenimiento	No Funcional	Equipo de Desarrollo	El sistema debe facilitar su mantenimiento y ajustes.	Levantamiento

12.4. Carta Gantt

ID	FASE / TAREA / ACTIVIDAD	DUR. HRS	F.INICIO	F.TERMINO	COSTO	ROL	S1	S2	S3	S4	S5	S6
1	Planificación - Definición de alcance	26	Semana 1	Semana 1	260000	Gestor de Proyecto						
2	Planificación - Asignación de recursos	26	Semana 2	Semana 2	260000	Gestor de Proyecto						
3	Planificación - Planificación inicial	26	Semana 3	Semana 3	260000	Gestor de Proyecto						
4	Análisis y Diseño - Diseño de mockups	40	Semana 3	Semana 3	400000	Analista de Sistemas						
5	Análisis y Diseño - Modelo de datos	40	Semana 4	Semana 4	400000	Analista de Sistemas						
6	Análisis y Diseño - Arquitectura de software	40	Semana 5	Semana 5	400000	Analista de Sistemas						
7	Desarrollo Backend - Estructura base de datos	66	Semana 6	Semana 6	660000	Desarrolladores Backend						
8	Desarrollo Backend - Lógica de negocios	66	Semana 7	Semana 7	660000	Desarrolladores Backend						
9	Desarrollo Backend - API de formulario	66	Semana 8	Semana 8	660000	Desarrolladores Backend						
10	Desarrollo Frontend - Formulario digital	66	Semana 8	Semana 8	660000	Desarrolladores Frontend						
11	Desarrollo Frontend - UI adaptable	66	Semana 9	Semana 9	660000	Desarrolladores Frontend						
12	Desarrollo Frontend - Consulta de formularios	66	Semana 10	Semana 10	660000	Desarrolladores Frontend						
13	Seguridad - Normativas	26	Semana 11	Semana 11	260000	Especialista en Seguridad						
14	Seguridad - Permisos	26	Semana 12	Semana 12	260000	Especialista en Seguridad						
15	Seguridad - Auditoría	26	Semana 13	Semana 13	260000	Especialista en Seguridad						
16	Pruebas y Validación - Pruebas funcionales	40	Semana 14	Semana 14	400000	QA Tester						
17	Pruebas y Validación - Compatibilidad	40	Semana 15	Semana 15	400000	QA Tester						
18	Pruebas y Validación - Corrección de bugs	40	Semana 16	Semana 16	400000	QA Tester						
19	Implementación y Capacitación - Despliegue	40	Semana 18	Semana 18	400000	Equipo de Proyecto						
20	Implementación y Capacitación - Capacitación a usuario	40	Semana 19	Semana 19	400000	Equipo de Proyecto						
21	Implementación y Capacitación - Soporte técnico	40	Semana 20	Semana 20	400000	Equipo de Proyecto						

12.5 Prototipo Figma

Enlace al prototipo:

<https://www.figma.com/design/EfUk5QJM8B3XoiXaPFJJRI/Prototipo-Solicitudes-de-ingreso?node-id=0-1&p=f&t=s1liV5eMzXSGFbe0-0>