



**Sistema de Administración de Información de Vehículos para la Empresa de
Transporte de Carga Terrestre del Sector Minero de Cundinamarca ISAMCO
SAS**

Ing. Diego Alejandro Poveda Sánchez – 10892217137

Ing. Jorge Alfonso García Espinosa – 10892218519

Universidad Antonio Nariño

Especialización en Ingeniería de Software

Facultad de Ingeniería de Sistemas

Bogotá D.C., Colombia

2022

**Sistema de Administración de Información de Vehículos para la Empresa de
Transporte de Carga Terrestre del Sector Minero de Cundinamarca ISAMCO**

SAS

Ing. Diego Alejandro Poveda Sánchez

Ing. Jorge Alfonso García Espinosa

Proyecto de grado presentado como requisito parcial para optar al título de:

Especialista en Ingeniería de Software

Director (a):

Ing. Dianalín Neme Prada

Ing. Iván Rodrigo Romero Flórez

Universidad Antonio Nariño

Especialización en Ingeniería de Software

Facultad de Ingeniería de Sistemas

Bogotá D.C., Colombia

2022

NOTA DE ACEPTACIÓN

El trabajo de grado titulado
*“Sistema de Administración de Información de
Vehículos para la Empresa de Transporte de Carga
Terrestre del Sector Minero de Cundinamarca
ISAMCO SAS”*,
Cumple con los requisitos para optar
Al título de
Especialista en Ingeniería de Software.

Firma del Tutor

Firma Jurado

Firma Jurado

Bogotá D.C., 16 de noviembre de 2022

Contenido

1.	RESUMEN.....	19
2.	ABSTRACT.....	21
3.	INTRODUCCIÓN.....	23
4.	FORMULACIÓN Y DESCRIPCIÓN DEL PROBLEMA	24
5.	OBJETIVO GENERAL	26
6.	OBJETIVOS ESPECIFICOS	27
7.	MARCO DE REFERENCIA	28
7.1	ESTADO DEL ARTE	28
7.1.1	<i>Sistemas en el Mercado Nacional.....</i>	28
7.1.2	<i>Sistemas en el Mercado Internacional.....</i>	32
7.2	COMPONENTE DE INNOVACIÓN	36
7.3	IMPACTO	38
7.4	MARCO TEÓRICO	39
7.4.1	<i>Minería de Carbón en Cundinamarca</i>	39
7.4.2	<i>Transporte de Carbón en Cundinamarca</i>	41
7.4.3	<i>Inclusión y Transformación Digital en Cundinamarca</i>	47
7.4.4	<i>Java.....</i>	48
7.4.5	<i>Spring Framework.....</i>	52
7.4.6	<i>Typescript</i>	54
7.4.7	<i>Vaadin.....</i>	55
7.4.8	<i>Computación en la nube</i>	56
7.4.9	<i>Arquitectura de Microservicios</i>	60
7.4.10	<i>Docker.....</i>	63
7.4.11	<i>PostgreSQL.....</i>	64
7.4.12	<i>Google App Engine Standard.....</i>	65
7.4.13	<i>Spring Cloud Gateway.....</i>	66
7.4.14	<i>Cloud SQL for PostgreSQL.....</i>	67
7.4.15	<i>Google Cloud Storage.....</i>	70
7.4.16	<i>Google Cloud Pub/Sub.....</i>	71
8.	DISEÑO METODOLÓGICO O METODOLOGÍA.....	73
8.1	SCRUM.....	73
8.1.1	<i>Roles</i>	74
8.1.2	<i>Eventos.....</i>	74
8.1.3	<i>Artefactos.....</i>	75
8.2	APLICACIÓN	76
9.	PROCESO DE SOFTWARE.....	78
9.1	REQUERIMIENTOS FUNCIONALES (HISTORIAS DE USUARIO)	78
9.1.1	<i>RF001: Administración de Conductores.....</i>	78
9.1.2	<i>RF002: Administración de Vehículos</i>	78
9.1.3	<i>RF003: Administración de Viajes y Trayectos.....</i>	78
9.1.4	<i>RF004: Registro de Novedades de Viajes y Trayectos.....</i>	79
9.1.5	<i>RF005: Gestión de Hoja de Vida de Vehículos.....</i>	79

9.1.6	RF006: Gestión de Alertas	79
9.1.7	RF007: Consulta de Reportes.....	80
9.2	REQUERIMIENTOS NO FUNCIONALES	80
9.2.1	RNF001: Comprobabilidad	80
9.2.2	RNF002: Confiabilidad	80
9.2.3	RNF003: Evolucionabilidad.....	81
9.2.4	RNF004: Modularidad	81
9.2.5	RNF005: Elasticidad	81
9.2.6	RNF006: Escalabilidad	81
9.2.7	RNF007: Facilidad de despliegue	82
9.3	DISEÑO Y ARQUITECTURA	82
9.3.1	Restricciones.....	82
9.3.2	Priorizaciones.....	82
9.3.3	Escenarios.....	83
9.3.4	Actores	83
9.3.5	Gerente	83
9.3.6	Administrativa.....	83
9.3.7	Conductor	84
9.3.8	Administrador del Sistema.....	84
9.4	DIAGRAMA DE DESPLIEGUE	85
9.5	CASO DE USO ARQUITECTURALMENTE RELEVANTE.....	87
9.6	DIAGRAMA DE SECUENCIA.....	91
9.7	DIAGRAMA DE CLASES	92
9.7.1	Vehículos	93
9.7.2	Ubicación Geográfica.....	93
9.7.3	Viajes	94
9.7.4	Empleados.....	94
9.7.5	Alertas.....	95
9.7.6	Minerales	95
9.8	ARQUITECTURA DE ALTO NIVEL.....	97
9.8.1	Visualización del Sistema	97
10.	CONSTRUCCIÓN.....	105
11.	PRUEBAS (PROCESO Y DE PRODUCTO)	112
12.	PRUEBAS DE SEGURIDAD.....	118
13.	INSTALACIÓN Y CONFIGURACIÓN.....	122
13.1	GOOGLE APP ENGINE	122
13.2	GOOGLE CLOUD SQL.....	124
14.	CONCLUSIONES.....	128
15.	REFERENCIAS BIBLIOGRÁFICAS	130

Lista de Figuras

<i>Ilustración 1 Framework Scrum (Scrum.org, 2020)</i>	74
<i>Ilustración 2 Diagrama de Contexto C4 (Autores, 2022)</i>	85
<i>Ilustración 3 Diagrama UML de Despliegue General (Autores, 2022)</i>	87
<i>Ilustración 4 Diagrama de Clases Consulta de Vehículos (Autores, 2022)</i>	89
<i>Ilustración 5 Diagrama de Clases Asignación de Vehículos (Autores, 2022)</i>	90
<i>Ilustración 6 Diagrama de Secuencia Asignación de Vehículos (Autores, 2022)</i>	91
<i>Ilustración 7 Diagrama de Secuencia Consulta de Vehículos (Autores, 2022)</i>	92
<i>Ilustración 8 Diagrama UML de Clases del Modelo de Negocio (Autores, 2022)</i>	96
<i>Ilustración 9 Diagrama de Contenedores C4 (Autores, 2022)</i>	97
<i>Ilustración 10 Pantalla de Login (Autores, 2022)</i>	100
<i>Ilustración 11 Pantalla de Menú Principal (Autores, 2022)</i>	101
<i>Ilustración 12 Pantalla de Dashboard (Autores, 2022)</i>	102
<i>Ilustración 13 Formulario de Vehículos (Autores, 2022)</i>	103
<i>Ilustración 14 Pantalla de Vehículos (Autores, 2022)</i>	104
<i>Ilustración 15 Formulario de Viajes (Autores, 2022)</i>	105
<i>Ilustración 16 Notificación por Correo Electrónico (Autores, 2022)</i>	105
<i>Ilustración 17 pom.xml (Autores, 2022)</i>	106
<i>Ilustración 18 Dockerfile (Autores, 2020)</i>	107
<i>Ilustración 19 Evaluación Sonar con errores (Autores, 2022)</i>	108
<i>Ilustración 20 Evaluación Sonar después de corrección (Autores, 2022)</i>	109
<i>Ilustración 21 Pull Request (Autores, 2022)</i>	110
<i>Ilustración 22 Pull Request Aprobado (Autores, 2022)</i>	111
<i>Ilustración 23 Hoja de Ruta en Jira (Autores, 2022)</i>	112
<i>Ilustración 24 Historia de Usuario en Jira (Autores, 2022)</i>	112
<i>Ilustración 25 Archivo POM con porcentaje de cubrimiento de pruebas unitarias (Autores, 2022)</i>	113
<i>Ilustración 26 Evaluación de cobertura de pruebas unitarias en Sonar (Autores, 2022)</i> ..	114
<i>Ilustración 27 Sprint Review en Confluence (Autores, 2022)</i>	115
<i>Ilustración 28 Login en ambiente de desarrollo para pruebas de seguridad (Autores, 2022)</i>	118
<i>Ilustración 29 Generalidades del reporte OWASP ZAP (Autores, 2022)</i>	119
<i>Ilustración 30 Reporte OWASP ZAP (Autores, 2022)</i>	120
<i>Ilustración 31 Detalles de seguridad encontrados por OWASP ZAP (Autores, 2022)</i>	121
<i>Ilustración 32 Panel App Engine (Autores, 2022)</i>	122
<i>Ilustración 33 Selección de Región para App Engine (Autores, 2022)</i>	122
<i>Ilustración 34 Selección de Recursos en App Engine (Autores, 2022)</i>	123
<i>Ilustración 35 Archivo de configuración para App Engine (Autores, 2022)</i>	123
<i>Ilustración 36 Ejecución comando gcloud para despliegue de contenedores en App Engine (Autores, 2022)</i>	124

Ilustración 37 Contenedor desplegado en App Engine (Autores, 2022)	124
Ilustración 38 Detalles del contenedor desplegado en App Engine (Autores, 2022)	124
Ilustración 39 Opción de creación de instancia SQL en GCP (Autores, 2022)	125
Ilustración 40 Elección de PostgreSQL como motor en Cloud SQL (Autores, 2022)	125
Ilustración 41 Configuración de Instancia PostgreSQL (Autores, 2022)	126
Ilustración 42 Estado de instancia Cloud SQL (Autores, 2022)	126
Ilustración 43 Detalles de instancia Cloud SQL (Autores, 2022)	126
Ilustración 44 Instancias ISAMCO en Cloud SQL (Autores, 2022)	127

Lista de tablas

<i>Tabla 1 Normatividad para transporte terrestre por carretera (PROCOLOMBIA, 2004)</i>	43
<i>Tabla 2 Configuraciones Permitidas y Usadas en el Transporte de Carbón en Colombia (PROCOLOMBIA, 2004)</i>	45
<i>Tabla 3 Estructura de Costos Operativos para una Tractomula (PROCOLOMBIA, 2004)</i>	47

Lista de Símbolos y Abreviaturas

Abreviaturas

Abreviatura Término

<i>RNDC</i>	Registro Nacional Despachos de Carga por Carretera
<i>UPME</i>	Unidad de Planeación Minero-Energética
<i>JVM</i>	Java Virtual Machine
<i>IDE</i>	Integrated Development Environment
<i>JCR</i>	Java Content Repository
<i>JSR</i>	Java Specification Requests
<i>i18n</i>	Internacionalización
<i>SpEL</i>	Spring Expression Language
<i>DAO</i>	Data Access Object
<i>JDBC</i>	Java Database Connectivity
<i>ORM</i>	Object-Relational Mapping
<i>XML</i>	Extensible Markup Language
<i>JMS</i>	Java Message Service
<i>JCA</i>	Java Cryptography Architecture
<i>JMX</i>	Java Management Extensions
<i>MVC</i>	Modelo Vista-Controlador
<i>DI</i>	Inyección de Dependencias
<i>JEE</i>	Java Platform, Enterprise Edition
<i>API</i>	Application Programming Interface
<i>IoC</i>	Inversión de Control
<i>EJB</i>	Enterprise JavaBean
<i>AOP</i>	Programación Orientada a Aspectos
<i>SPA</i>	Single Page Application
<i>DOM</i>	Document Object Model
<i>HTML</i>	HyperText Markup Language
<i>CPU</i>	Unidad de Procesamiento Central
<i>GPU</i>	Unidad de Procesamiento Gráfico
<i>TPU</i>	Unidad de Procesamiento Tensorial
<i>CI/CD</i>	Integración Continua y Entrega Continua

Dedicatoria

A nuestras familias.

Agradecimientos

Los autores del presente documento agradecen la ayuda y colaboración de las siguientes personas y empresas:

- ISAMCO SAS, y en su representación a Enrique Poveda, gerente general, Sandra Liliana González, asistente administrativa y todos los demás empleados y conductores de la empresa que nos colaboraron con la construcción del proyecto.
- Globant – Sistemas Colombia SAS, y Farmatodo de Colombia S.A. y en su representación a nuestros jefes y compañeros de trabajo, quienes nos permitieron los espacios necesarios para la consecución del proyecto y nos alentaron a continuar adelante en el cumplimiento de este objetivo profesional.

1. Resumen

ISAMCO SAS, empresa de explotación y transporte de mineral carbón ubicada en Cundinamarca, gracias a su crecimiento en los últimos años solicitó un sistema de información que le permita consolidar la información de los vehículos y viajes de forma ágil y certera. En búsqueda de suplir esta necesidad, se ha implementado un sistema de información web que ayuda a la empresa a cumplir esta meta y avanzar en su ingreso dentro de la cultura y transformación digital.

Aunque existen varias soluciones similares en el mercado nacional y extranjero, ISAMCO SAS decidió construir su propia solución, que se ajuste totalmente a sus necesidades y que pueda adaptar fácilmente según nuevas ideas o requerimientos del entorno minero se vayan presentando.

A nivel de innovación, la solución para ISAMCO SAS se basó en arquitectura de microservicios e infraestructura en la nube, para poder tener una capacidad de adaptación tecnológica, utilizando entornos de desarrollo y ejecución actuales, que permitan al sistema de información mantenerse vigente con el tiempo y evitar costes de adquisición y mantenimiento de infraestructura propia en sitio.

También, mediante la aplicación de la metodología Scrum, se logró mantener una retroalimentación permanente entre el equipo de desarrollo y la empresa, para ajustar los desarrollos con el fin que estos cumplieran los objetivos propuestos para la puesta en marcha de la prueba piloto productiva realizada para la verificación del sistema de información.

Toda la gestión de la metodología de desarrollo estuvo apoyada en herramientas disponibles en la nube para la administración y almacenamiento de la información del proyecto.

El sistema de información permitió a los principales roles de ISAMCO (Gerencia, Administrativo, Conductores y Sysadmin) realizar las tareas de gestión de vehículos y viajes de la forma ágil esperada, disminuyendo la tasa de errores y dando una visión general de los ajustes a futuro que se necesitan para dar el siguiente paso, correspondiente a la puesta en marcha total del sistema para toda la operación.

2. Abstract

ISAMCO SAS, a coal mining and transportation company located in Cundinamarca, thanks to its growth in recent years, requested an information system that allows it to consolidate the information on vehicles and trips in an agile and accurate manner. In search of solve this need, a web information system has been implemented that helps the company meet this goal and advance its entry into the culture and digital transformation.

Although there are several similar solutions in the national and foreign market, ISAMCO SAS decided to build its own solution, which fully fits its needs and can be easily adapted as new ideas or requirements of the mining environment arise.

At the level of innovation, the solution for ISAMCO SAS was based on architecture of microservices and infrastructure in the cloud, in order to have a technological adaptation capacity, using current development and execution environments, which allow the information system to remain current over time. and avoid costs of acquisition and maintenance of own infrastructure on site.

Also, through the application of the Scrum methodology, it was possible to maintain a permanent feedback between the development team and the company, to adjust the developments so that they met the objectives proposed for the start-up of the productive pilot test carried out for verification of the information system. All the management of the development methodology was supported by tools available in the cloud for the administration and storage of project information.

The information system allowed ISAMCO's main roles (Management, Administrative, Drivers and Sysadmin) to carry out vehicle and travel management tasks in the expected agile manner, reducing the error rate and giving an overview of the adjustments to future that are needed to take the next step, corresponding to the total start-up of the system for the entire operation.

3. Introducción

El presente documento expone el desarrollo de un proyecto de software para la gestión y administración de la información de los vehículos de la empresa ISAMCO SAS del sector transporte en el sector minero. El documento abarca la problemática a solucionar, el estado del arte respecto a tecnologías similares, una descripción del sector minero y de transporte y la propuesta tecnológica. Además, el documento describe la metodología utilizada durante el proceso de construcción del proyecto, los requerimientos funcionales y no funcionales, la arquitectura aplicada al sistema, la descripción de la construcción del mismo y la instalación y configuración.

4. Formulación y descripción del problema

ISAMCO SAS, es una empresa ubicada en el Municipio de Cucunubá, provincia de Ubaté, en el Departamento de Cundinamarca, que se dedica a la explotación y transporte de mineral carbón. La empresa ha presentado un crecimiento importante en los últimos 5 años, el cual se ha visto reflejado en la adquisición de vehículos y contratación de personal, incrementado el trabajo operativo que conlleva gestionar la información de los vehículos, conductores y viajes.

Hoy en día la gestión de la información relacionada a los vehículos se hace en diferentes herramientas ofimáticas como Microsoft Excel, Formularios de Google, además de plantillas y formularios impresos, haciendo que el trabajo operativo se incremente a medida que la flota de vehículos y la planta de personal crece. Actualmente el proceso de consolidar la información de los vehículos toma aproximadamente una semana, según lo reportado por la empresa, el cual realiza el personal Administrativo y lo debe hacer de manera meticulosa para evitar cometer errores.

ISAMCO SAS ha manifestado interés por hacer parte de una solución que permita resolver estos problemas, utilizando tecnologías de la información.

A partir de la identificación de la problemática de esta empresa se propone una solución que consiste en una plataforma tecnológica web que va a estar alojada en la nube

y podrá ser accedida desde cualquier dispositivo electrónico que tenga un navegador compatible.

La plataforma permitirá tener siempre a la mano información actualizada de los vehículos de la empresa, tal como:

- Hoja de vida del vehículo.
- Viajes y trayectos, tanto asignados como realizados.
- Ingresos generados por el vehículo por concepto de viajes.
- Gastos asociados a viajes, mantenimientos y otros.
- Impuestos.
- Seguros y pólizas.
- Mantenimientos.
- Conductores.
- Alertas de viajes, trayectos, mantenimientos y seguros.
- Carga y consulta de documentos relacionados con la carga a transportar.

5. Objetivo General

Desarrollar una plataforma tecnológica web que permita a ISAMCO SAS, empresa de transporte de carga del sector minero de Cundinamarca, gestionar la información de sus vehículos y los viajes que realizan.

6. Objetivos Especificos

Disminuir los tiempos empleados en la conciliación de información de transporte de carga para ISAMCO SAS.

Reducir el número de errores administrativos presentados durante el proceso de conciliación de información de los viajes y vehículos en ISAMCO SAS.

Centralizar la información de los vehículos y viajes, para ofrecer reportes que permitan tomar decisiones a la Gerencia de ISAMCO SAS.

Proporcionar a ISAMCO SAS notificaciones de vencimientos relacionados con sus vehículos o viajes, que permitan a las áreas administrativa y operativa adelantarse a estos eventos y evitar multas y/o sanciones.

Ayudar a ISAMCO SAS en la mejora de sus procesos de transporte mediante la inclusión y cultura digital.

7. Marco de Referencia

7.1 Estado del Arte

Actualmente existen las siguientes aplicaciones en el mercado para el transporte de carga:

7.1.1 Sistemas en el Mercado Nacional

SITCA: Es un software, de Bogotá, para compañías de transporte de carga terrestre, ya sea paquetero o a granel. Corresponde a una plataforma web donde los clientes pueden conservar un registro y control minucioso de sus procedimientos, órdenes de producción, contenedores, viajes, entre otras. El sistema cuenta con los siguientes módulos (GRUPO NW, 2022):

- Inspección de conflictos de terceros: Análisis con consentimiento de conductores y camiones.
- Gestión de clientes: Catálogo renovado sobre clientes.
- Comprobación de cuentas: Liquidación de tasas por trayecto, liquidación de concesionarios por peaje.
- Sistema de turnos: Creación de turnos para entradas reguladas.
- Gestión de reubicación de contenedores: Verificación de embarque y desembarque de lugares.

- Gestión de peajes: Lista de peajes y concesionarios con sus concernientes importes.
- Órdenes de Producción: Programación de futuras transacciones y viajes.
- Transmisión al RNDC: Papeles apropiadamente acreditados por el Ministerio de Transporte.

TRANSCARGA: El software de Syscom, sociedad con lugar en Bucaramanga, faculta a las empresas transportadoras inspeccionar su desarrollo operativo y contable correspondiente a sus transportes; comenzando en la orden de cargue hasta la liquidación y recaudación de los servicios que se proporcionen. A la par, proporciona la administración vigilada de empréstitos, remesas, documentos de carga, vencimientos y pagos, advierte con anticipación el cumplimiento de documentaciones tanto del conductor como del vehículo. El sistema cuenta con las siguientes funcionalidades (SYSCOM, 2022):

- Se acomoda a la regulación formulada por el Ministerio de Transporte que normaliza el servicio de transporte terrestre automotor de carga e instituye el empleo del formato único de manifiesto de carga y otras normatividades ceñidas a la resolución actual en vigor.
- Posee conexión en tiempo real con los servidores de Mintransporte, ahorrando al usuario reprocesos para la transferencia de datos a la institución.
- Emplea la información interrelacionada con los departamentos, ciudades, marcas, líneas de vehículos, modelos, colores, productos a transitar, tipos de carrocerías, unidades de medida, propiedades de la carga, unidades de empaque, requeridas por MinTransporte.

- Habilita la búsqueda de novedades en puntos de control de todos los vehículos, brindando un módulo de seguridad e interfaces con compañías de seguridad y parajes de control.
- Ofrece variados informes de remisiones, ejercicios de anticipo, despachos y seguridad, documentos, facturación, e informes financieros.
- Comprueba el cumplimiento de documentaciones de los vehículos y de los conductores mediante alertas instantáneas.

CLOUD FLEET: Con origen en Medellín, es una aplicación en la nube, planteada para optimar egresos y acrecentar la productividad de los vehículos. El sistema tiene las siguientes funcionalidades (CLOUD FLEET, 2022):

- Administra las restauraciones de los vehículos, implanta y elabora propios métodos de mantenimiento acorde con la operación.
- Inspecciona y examina el rendimiento del carburante utilizado. Revela anomalías en el empleo por vehículo o por conductor.
- Lista de chequeo, sobre la cual se crean automáticamente labores de mantenimiento.
- Exhibe los vehículos o equipos libres para operación. Asimismo, presenta qué proporción del tiempo han permanecido operativos.
- Gestiona el depósito de suministros interconectado con el área de mantenimiento y adquisiciones.
- Soporta el dominio de desplazamientos, recorridos, conductores, liquidación de anticipos.

SILOGCARGO: Proyecto con origen central en Cali, el cual suministra la interacción entre generadores de carga y transportadoras mediante roles adaptables en el sistema, agrupando los datos, computando el beneficio, lo cual protege de reprocesos, perfecciona y beneficia la operación. El sistema tiene las siguientes características (COLOMBIA SOFTWARE, 2022):

- Proyecta y mejora el transporte.
- Asiste a la elección del suministrador de transporte según el tipo de transporte o mecanismo más conveniente.
- Búsqueda de envíos en tiempo real.
- Optimización de carga, acorde con el itinerario y potencial del equipo.
- Simulación de costes y períodos de embarques.
- Gestión de reportes y estadísticas.
- Ratificación de facturas.
- Estado de entrega de mercadería.
- Exploración de Guías Digitalizadas.
- Hitos de Gestión.
- Facturación.
- Integración con terceros sistemas y aplicaciones.

TORRE DE CONTROL: Con centro en Bogotá y Medellín, bajo la perspectiva de contribuir al entorno del transporte y la logística, la empresa ControlLT facilita un medio informático con las siguientes características (CONTROLT, 2022):

- Integración de numerosos orígenes de información y asimismo se disminuyan los tiempos operativos de personal que hoy día están reuniendo

datos de forma manual, por medios digitales y habituales (llamadas, formatos en papel, etc).

- Más de 110 proveedores de GPS integrados.

7.1.2 Sistemas en el Mercado Internacional

PULPO (*México*): Es un sistema de administración de flotas en la nube, disponible 24/7, que brinda las siguientes características (PULPO, 2022):

- Información congregada en la nube. Los datos y documentos digitalizados continuamente a disposición de la empresa.
- Disponibilidad de los vehículos, programación de mantenimientos, alertas para que no se venza ningún oficio y listas de chequeo de los conductores.
- Reportes ejecutivos que reconocen vehículos ociosos, cuáles son los más ineficaces, los más contaminantes, aquellos que se emplean menos de lo que corresponderían.

CRISTAL (*Argentina*): Es un software para transporte logístico y sostenimiento de flotas, que brinda los módulos detallados a continuación (SOFTAR SRL, 2022):

- Proporciona el rastreo de las cargas trasladadas, entablando el procedimiento con la recopilación de la mercancía hasta la rendición de los recibos al origen. Permite efectuar trazabilidad por remisión o bulto a bulto para un mayor control. La carga se encamina conforme con la configuración del Manual de Servicios tanto por dependencias propias como por transportistas empleados.

- Formaliza el rastreo de los viajes de la empresa, inspeccionando la trazabilidad de este más allá del contenido. Registra planes de salida y llegada, gestiona múltiples recorridos para un mismo viaje, controla expensas y adelantos y liquida servicios a los clientes y conductores, tanto exclusivos como contratados.
- Administra los almacenes y el movimiento de la carga en los mismos. Habilita la capacidad de diseñar la disposición de los depósitos con las ubicaciones en variados niveles, administrando movimientos, abastecimiento, recolecciones y procesos que merecen ejecutarse a la mercadería acopiada.
- Mantenimiento de escuadras de vehículos y equipos para ejecutar el mantenimiento anticipado y corrector ya sea en talleres exclusivos o contratados, con repuestos y partes compradas o del depósito de la empresa donde se regula el inventario. Tiene también un módulo de gestión de llantas autónomo y de arrendamiento de vehículos y equipos.
- ERP añadido para la gestión de la contabilidad de la empresa, el cual contiene caja, bancos, suministradores y gestión contable. Este módulo se completa con todos los anteriores automáticamente, en caso de adquirir un solo software y evitar interfases extras.

EASY CARGO (*República Checa*): Es un software de embalaje de contenedores y furgones que tiene las siguientes características (BEE INTERACTIVE S.R.O., 2022):

- Sistematizaciones del plan de carga.
- Más de 10 000 productos de hasta 250 tipos distintos en un solo envío.

- Permite el uso de unidades imperiales (pulgadas, libras) o métricas (centímetros, kilogramos).
- Grupos de prelación que catalogan los productos de carga según el rumbo final.
- Editor manejable del plan de carga (conforme al lugar y la translación de cada compartimiento del plan de carga computado).
- Grupos no apilables y grupos fraccionados por una muralla virtual.
- Carga de módulos directamente a partir de un archivo de Microsoft Excel.
- Exportación del plan de carga a Microsoft Excel.
- Integración con SAP.
- Acomodar la capacidad de la carga.
- Base de datos de productos de carga.
- Configuración de limitaciones para el posicionamiento de mercancías de carga (no apilable, sin declive, no girar, permutar al centro de masa, etc.).
- La reseña se exhibe en cada producto llanamente en la visualización 3D.
- Declaración de vehículos, contenedores, remolques de 1, 2 y 3 ejes.
- Reseña de carga progresivamente.
- Resúmenes de impresión a partir de un plan de carga.

DIDTRANS (*España*): Es un sistema esbozado para cobijar los requisitos del cargador o agencia de transporte a través de las siguientes funcionalidades (DIGITAL DISSENY, 2022):

- Gestiona los datos de todos los que participan durante la actividad del transporte. A partir de los datos de cliente, incluso la gestión de flotas.

- Factura todas las cargas de transporte. Se permite imprimir o enviar por correo en diferentes formatos.
- Administra el transporte desde su establecimiento hasta su importe. Determina transportistas, conductores, vehículos, evaluaciones, trayectos. Imprime la documentación necesaria para todo tipo de transporte, cargas llenas, contenedores, grupajes y paquetería.
- Escanea y guarda los documentos concernientes, recibos de transporte, facturas de proveedores, contratos con clientes e, información de vehículos.
- Vigila los pagos y recaudaciones pendientes. Administra además las facturas receiptadas de todo tipo.

LISTMS (*México*): Es un software en la nube, para las corporaciones de transporte que solicitan un alto control de su operación y precisan una respuesta que los apoye a crecer.

Los siguientes son los módulos brindados en esta solución (LIS, 2022):

- Automatización de las técnicas de facturación.
- Control de los viajes prorrogados por facturar.
- Formación, búsqueda y descarga de documentos fiscales.
- Realización de notas de crédito y débito para clientes.
- Fraccionamiento de datos por cliente, unidad y filial.
- Gestión de cuentas por cobrar.
- Ordenación de estados de cuenta por cliente.
- Ejecución de proyecciones de cobranza.
- Incorporación de complementos de pagos.
- Creación de viajes, guías y cartas porte.

- Gestión de anticipos, carburantes y liquidaciones.
- Exámenes de lecturas de odómetro.
- Administración de los registros de periodos de servicio.
- Integración de GPS para el monitoreo de mercancía.
- Automatización de sueldos por km, comisión y viaje.
- Gastos de empresa verificables mediante anticipos.
- Utilidad de combustible por liquidación.
- Disminución de combustible por liquidación.
- Cálculo automático de sueldo a conductores.
- Alertas de mantenimientos preventivos.
- Advertencias en tiempo real.
- Aseguramiento del cumplimiento de mantenimientos preventivos.
- Alarmas de vencimientos.
- Control de registros de mantenimientos.
- Gestión de los gastos por vehículo.
- Caracterización de unidades con mantenimientos futuros o cumplidos.

7.2 Componente de Innovación

La propuesta de innovación consiste en proveer una solución tecnológica construida directamente para ISAMCO SAS, desplegada y disponible en infraestructura Cloud y accesible mediante una aplicación web, que permita desde cualquier dispositivo con

navegador compatible ingresar al sistema y utilizar las funcionalidades necesarias para la administración y seguimiento.

El mayor beneficio recae en ser un desarrollo a la medida, ya que el software se adaptará a los procesos de la empresa y no requerirá en fases iniciales que la empresa se acomode a los procesos establecidos por herramientas externas, lo cual permitirá a los usuarios dentro de ISAMCO SAS proveer retroalimentación de una forma más cercana con el equipo de construcción del sistema y ver reflejados sus requerimientos y observaciones en el producto resultante, generando así menos resistencia al cambio, percibiendo la herramienta como algo positivo que les permitirá tener mayor control de su trabajo.

Además, dado que el sistema ha sido pensado para ser construido en varias fases (la primera fase es la que se describe en el presente documento), ISAMCO SAS tendrá el tiempo suficiente para probar los resultados y refinar sus requerimientos adaptándose a su presupuesto y necesidades, motivo por el cual la empresa no seleccionó ninguna de las ofertas existentes en el mercado, puesto que fueron evaluadas como soluciones de costo elevado y que ofrecen varios módulos que no se utilizarían en el corto plazo por la empresa, lo que conllevaría a pagar por recursos que no serían utilizados, mientras que con la herramienta solicitada a la medida, ISAMCO SAS aprovechará todos los módulos que fueron requeridos para el control de su operación.

Tecnológicamente, la solución al ser diseñada para ejecutarse a través de una aplicación web y no en aplicaciones móviles, extiende el rango de usuarios que pueden

interactuar a través de ella, ya que no habría ningún tipo de restricción respecto al sistema operativo o dispositivo desde el cual se ejecute la operación, siempre y cuando este tenga la posibilidad de conectarse a internet. Esto también afecta positivamente a ISAMCO SAS, ya que no será necesario realizar grandes inversiones en infraestructura y en dispositivos para mejorar sus procesos con el uso de la herramienta.

Finalmente, las tecnologías seleccionadas para la construcción de la herramienta se encuentran entre las líderes del mercado, ofreciendo así, una solución robusta a nivel de los componentes del sistema, y permitiendo a la empresa obtener buen soporte y adaptabilidad a nuevos requerimientos y tecnologías que sean necesarias para construir módulos que expandan y mejoren la solución.

7.3 Impacto

Uno de los aspectos más importantes de la aplicación es que permitirá una mejora en los procesos administrativos de ISAMCO SAS, permitiendo enfocar estos esfuerzos en buscar nuevos objetivos que permitan el crecimiento y expansión de la empresa en el sector minero, al igual que extender oportunidades para involucrar otras empresas pertenecientes al mismo grupo de inversores y que se encuentran ligadas al sector minero, tanto en transporte como en extracción e ingresar a futuro en otros procesos relacionados con la minería y también proveer el servicio del sistema de información a otras empresas del mismo sector en la región de Cundinamarca y Boyacá.

De igual manera, es una oportunidad de brindar experiencias tecnológicas a personal que no está frecuentemente interactuando con este tipo de herramientas para su trabajo diario, ayudando a aumentar la cultura digital dentro del sector y de la población cundinamarqués.

La solución tecnológica también tiene un impacto ambiental positivo, el cual es generado al digitalizar parte de sus procesos y dejar de lado el uso de documentos impresos, disminuyendo el uso innecesario de papel.

7.4 Marco Teórico

7.4.1 Minería de Carbón en Cundinamarca

Con una productividad anualizada próxima a los 85 millones de toneladas, el carbón se formaliza como el producto minero que compone una notable contribución al PIB de Colombia, conviniendo entre 15 al 20% del PIB minero y entre 1,5 a 2% del PIB de la nación. El 90% de la obtención de carbón de Colombia es extraído del Cesar y La Guajira, y son de función térmica, y en gran medida son exportados a través de los puertos de Santa Marta y Puerto Bolívar. El demás 10% de la obtención es forjado por pequeños y medianos mineros en las regiones mineras en Córdoba, Cundinamarca, Boyacá, Santander, Norte de Santander, Antioquia, Valle y, Cauca. Los carbones de estas zonas también son de uso térmico, pero, en Santander, Norte de Santander, Boyacá y Cundinamarca se hallan también con aplicación siderúrgica de alta calidad que se consumen en el comercio interno y también

se exportan. La sustracción minera en estas tierras es subterránea y se utilizan las capas de las Formaciones. (Agencia Nacional de Minería, 2022)

En relación a las técnicas mineras en Cundinamarca, la Agencia Nacional de Minería ha declarado que “es incuestionablemente una muestra de minería bien elaborada para Colombia, un territorio donde gracias a su capacidad, y a la labor valiente de sus mandatarios, la minería es engranaje para el progreso, el avance, y la tranquilidad de sus colectividades”, y la Gobernación del Departamento ha mencionado que “Cien municipios del departamento tienen planes mineros en explotación y exploración, los capitales de las regalías están volcados en cuestiones tan importantes como la nutrición y el transporte colegial. Asimismo, con el este patrimonio se han desplegado proyectos agropecuarios y de infraestructura eléctrica”. Para el año 2017, Cundinamarca poseía aproximadamente de 978 títulos mineros efectivos que constituyen el 6% del departamento, del cual menos de un 3% está arbitrado. En esa dimensión se obtienen carbón, sal, hierro, materia prima para edificación, cales, cerámicas, esmeraldas, entre otros, transformándolo en uno de los departamentos con extraordinaria variedad respecto a recursos del subsuelo. Del mismo modo, el departamento colabora alrededor del 3% de la obtención de carbón del país, el 84% de sal, el 63% de hierro, el 29% de arcillas, el 25% de los materiales directos para construcción y el 3% de las calizas. (Agencia Nacional de Minería, 2017)

El Ministerio de Minas tiene en este momento con un programa llamado Agenda Carbón, proposición fundada por el Gobierno Nacional para continuar fructificando la prosperidad del carbón de Colombia de forma profesional, con los más altos modelos de

calidad y exigencia, admitiendo su valor para la seguridad energética y para la cimentación de infraestructura, por el requerimiento de este mineral en la producción de acero. El Ministerio comunicó que alrededor de 6.000 mineros se legalizaron para obrar bajo proyectos de minería legal. Esta cifra es escasamente un porcentaje del objetivo establecido por el gobierno, que es de 27.000 mineros legalizados para 2022. Asimismo, el Ministerio publicó la realización de un proyecto de infraestructura para la adiestramiento e invención de tecnologías en el sector minero. Este proyecto se llama Centro de Formación e Innovación Minero-energético y está localizado en el municipio de Guachetá. El proyecto fue revelado por la Gobernación de Cundinamarca y fue admitido en el marco de la citación abierta por el Ministerio de Minas para ofrecer capitales del SGR encaminados al impulso minero. (Auren, 2021)

7.4.2 Transporte de Carbón en Cundinamarca

El carbón habitualmente es trasladado desde la cantera en volquetas de 10t, doble troques de 20t y tractomulas de 40t y en demás situaciones por barcas, bandas transportadoras, cables aéreos y vías de tren. Es acarreado a los espacios de recolección, las plantas de beneficio, clientes internos y a los puertos de envío para su ulterior exportación. (UPME, 2005).

Las siguientes son las normativas que regulan el transporte terrestre del carbón por carretera en Colombia (PROCOLOMBIA, 2004):

Tipo de Norma	Número y año	Contenido
Decreto	2044 de 1988	Se dictan disposiciones sobre acarreo de productos especiales en vehículos de servicio público de transporte de carga.
Decreto	541 de 1994	Se reglamentan las escombreras del país incluyendo cargue, descargue, transporte, almacenamiento y disposición final de diversos materiales.
Decreto	173 de 2001	Se reglamenta el servicio público de transporte terrestre automotor de carga.
Ley	769 de 2002	Se expide el Código Nacional de Tránsito Terrestre.

Resolución	2499 de 2002	Establece la Ficha Técnica para el formato único de manifiesto de carga.
Resolución	5457 de 2002	Modifica el Decreto 2499 de 2002.
Decreto	1609 de 2002	Se reglamenta el manejo y transporte terrestre automotor de mercancías peligrosas por carretera.
Resolución	10799 de 2003	Fijan criterios de control de peso en algunos vehículos de carga.

Tabla 1 Normatividad para transporte terrestre por carretera (PROCOLOMBIA, 2004)

Entre los importantes actores del sector privado que intermedian en el transporte de carbón se pueden indicar los siguientes:

- Empresarios mineros y comercializadores de carbón: el primer eslabón en la cadena de transporte, ya que son quienes integran la categoría de los productores y expedidores de la carga de carbón y están citados a ejercer un papel significativo en los exámenes que deben realizarse a la operación.
- Empresas transportadoras: mantienen a su cargo las cuadrillas que conforman la flota empleada según el servicio que se suministre y el sistema

que se aplique. Hacen parte además de este grupo los conductores, los propietarios y las asociaciones privadas.

- Receptores de la carga de carbón: sean estos intermedios (comercializadores, puertos o patios de recolección) o finales (industrias compradoras, empresas transformadoras, centrales térmicas, etc.), quienes se establecen en el punto conclusivo de la cadena de transporte y logran llegar a ser concluyentes en el dominio de la actividad, como zona de cierre y declarantes del buen progreso y ejercicio del transporte de carbón.
- Agentes terceros: atañen a las empresas de mantenimiento, limpieza y aparcamiento para vehículos de transporte, quienes tienen del mismo modo gran realización en el cometido global ambiental de la labor del transporte de carbón. (PROCOLOMBIA, 2004)

Para 2001, según el Ministerio de Transporte, se movieron por carretera 76 Mt de carga, mediante el uso de 168.000 camiones, entre duros y articulados, de los cuales 72.240 eran de servicio privativo y 95.760 de sector público. Conforme con datos de Colfecar, la carga transportada creció para el 2002 a 87 Mt y para el 2003 a 91 Mt. (PROCOLOMBIA, 2004)

El potencial de las carreteras de Colombia se ha determinado de tal forma que el peso de los vehículos que circulen por ellas no exceda las 10 toneladas por eje y admitiendo esto y las disposiciones señaladas por el Instituto Nacional de Vías y el Ministerio de

Transporte, se tiene una categorización de las proporciones autorizadas para el transporte de carbón, exhibidas en la siguiente clasificación (PROCOLOMBIA, 2004):

Tipo de Vehículo	Configuración	Peso máximo (t) incluyendo el peso del vehículo.
Camión rígido de dos ejes	C2	16
Camión rígido de tres ejes	C3	28
Camión rígido de cuatro ejes	C4	36
Tracto mula de tres ejes	C2-S1	27
Tracto mula de cuatro ejes	C2-S2	32
Tracto mula de cinco ejes	C3-S2	48
Tracto mula de seis ejes	C3-S3	52
Camión y remolque de cuatro ejes	C2-R2	31
Camión y remolque de cinco ejes	C3-R2	44
Camión y remolque de seis ejes	C4-R2	48

Tabla 2 Configuraciones Permitidas y Usadas en el Transporte de Carbón en Colombia (PROCOLOMBIA, 2004)

En Cundinamarca se utilizan vehículos muy anticuados, tipo volqueta, los cuales habitualmente tienen carrocerías de estacas y son recargados empleando prolongaciones en madera. Asimismo, se usan doble troques para el transporte de las regiones mineras a los centros de recolección o consumo directo. Para los caminos de exportación hacia las costas Atlántica y Pacífica y hacia el Valle y Antioquia, se recurren a tractomulas de enorme capacidad tipo C3-S3 y C3-S2. (PROCOLOMBIA, 2004)

En relación con los costos de transporte adjuntos al comercio local y primordialmente con el transporte por carretera, la movilización del carbón en trayectos breves desde las minas hasta los acopios se hace en transporte privativo con precios de fletes que escasamente alcanzan los costos operativos. El transporte para amplias travesías es ejecutado por terceros distintos a los productores de carga, (constituidos por los mineros y comercializadores de carbón) quienes, en algunos casos, son empresas de transporte de carga apropiadamente instituidas. Los fletes cambian trascendentalmente, obedeciendo a cuál tipo de transportador se emplee, no obstante, hay que dar la razón al uso de empresas transportadoras puede favorecer a lograr mejoras medioambientales y evidentemente a componer la actividad del transporte en Colombia. Tradicionalmente el carbón es una mercadería que no se asegura, lo que hace que sus fletes sean mínimos en símil con otros tipos de productos que por sus naturales precios necesitan de un seguro para ser transportadas. En la distribución de costos del transporte, las variables que más determinan la constitución de los fletes son: combustibles, mantenimiento y ruedas, las cuales a su vez tienen una angosta correspondencia con propiedades de trazado y mantenimiento de vías. (PROCOLOMBIA, 2004)

La siguiente tabla expone la distribución de costos operativos representativa para el modo carretero (PROCOLOMBIA, 2004):

Concepto	Participación (%)
Salarios, prestaciones y comisiones	12.25
Combustibles	24.22

Llantas y neumáticos	15.25
Filtros y lubricantes	3.45
Mantenimiento y reparación	14.14
Peajes	12.88
Impuestos al vehículo	0.27
Seguros	8.68
Retefuente	1.64
Garajes y lavado	2.53
Otros	4.69
TOTAL	100.0

Tabla 3 Estructura de Costos Operativos para una Tractomula (PROCOLOMBIA, 2004)

7.4.3 Inclusión y Transformación Digital en Cundinamarca

Desde 2020, la secretaria de las TIC de Cundinamarca, bajo el liderazgo del Ministerio TIC, realiza un proceso de formación y acompañamiento a los 116 municipios del departamento, que tiene como objetivo alcanzar la Transformación Digital en sus regiones. En absoluta sintonía con la Política de Gobierno Digital, Cundinamarca toma el desafío de incluir las tecnologías para que los habitantes utilicen de forma adecuada los servicios que brindan las entidades. Según Andrés Roldan, director de Gobierno en Línea, para alcanzar los objetivos establecidos por el gobierno nacional y aceptadas en los planes de gobierno departamentales y municipales, desde el Ministerio TIC en labor vinculada con la secretaria de las TIC, mediante la estrategia ‘Transformación Digital para Todos’, se

ejecutará el acompañamiento técnico e individualizado para que se pueda prosperar hacia la madurez digital en las entidades. (Gobernación de Cundinamarca, 2020)

Para 2021, 7.031 habitantes de Cundinamarca se han facultado en fortalecimiento digital mediante las diversas preparaciones que está realizando la Gobernación en temas como Teletrabajo, Seguridad Cibernética, Datos Abiertos y el ABC en la ejecución de la Política de Gobierno Digital, son algunas de los entrenamientos con las que la Secretaría TIC ha arribado, tanto de forma presencial como virtual, a las entidades públicas y privadas de Cundinamarca. Los procesos se elaboran en unión con el Ministerio de las TIC, actuando en la disminución de las desigualdades, ya que gracias a las capacitaciones y la virtualidad se ha conseguido la cooperación de toda población sin distinción de género, raza o, condición social. (Suarez, 2021)

7.4.4 Java

Java es un lenguaje de programación y una plataforma informática distribuida por ocasión inicial en el año de 1995 por Sun Microsystems. Existen diversas aplicaciones y sitios web que no marcharán si no se tiene Java instalado y cada día se erigen más y más. Java es veloz, seguro y sólido. Desde laptops, centros de datos, consolas de videojuegos, súper computadoras, teléfonos inteligentes e incluso Internet, Java se encuentra en todas partes. (Oracle, 2022)

Dentro de sus características se encuentran (Carranza, 2021):

- Simplicidad: Java no posee las peculiaridades más imprecisas y menos empleadas de C y C++, lo que lo convierte más simple y natural de emplear.
- Orientado a objetos: esta dirección faculta el diseño del software para que los numerosos tipos de datos perduren fusionados a sus procedimientos.
- Distribuido e libre de la plataforma: tiene una extensa biblioteca estándar, aparte de sus herramientas para que los productos de software logren distribuirse. Al mismo tiempo, Java puede trabajar en cualquier hardware, haciéndolo portátil y flexible.
- Seguro y Multihilo: tiene una plataforma bastante segura, y asimismo certifica canales de comunicación de alta confiabilidad para resguardar la privacidad de los datos.

Java es el lenguaje más utilizado en las compañías que construyen su propio software. Desde su creación se ha encauzado esencialmente a solventar los inconvenientes concernientes al desarrollo de software, desde la perspectiva empresarial, contribuyendo un marco de trabajo más prolífico y que facultara crear software más mantenible y escalable.

Dentro de las ventajas que tiene construir software con Java, se encuentran (ARTECO, 2020):

- Es gratis: Actualmente, la plataforma OpenJDK es el apoderado de divulgar las ulteriores versiones de Java y de la máquina virtual imprescindible para ejecutar los programas compuestos en este lenguaje, manteniendo las pautas marcadas por la Java Community Process con más de 1200 integrantes, tales como Google, Adobe, Boeing, Oracle, Twitter, entre otros.

- **Tipado estático:** las variables sólo son de un tipo y éste no puede cambiar con el tiempo. Esta supuesta restricción, es una licencia para aplicaciones corporativas de gran importancia en donde colaboran equipos de programadores escribiendo cientos de miles de líneas de código. Las tareas de análisis de código, como las que efectúa el compilador, habilitan la detección una gran cantidad de errores durante la compilación, lo cual es invariablemente mejor que hallarlos durante la ejecución. Máxime, cuando la aplicación se está ejecutando en sistemas embebidos o pequeños dispositivos.
- **Portabilidad:** su compilación no forma código nativo, si no que reproduce código para ser interpretado en una máquina virtual, JVM. La Máquina Virtual de Java, es un intérprete capacitado para ejecutar aplicaciones compiladas en Java (y de otros lenguajes) abstrayéndose de las peculiaridades de cada plataforma, brindando al programador una original interfaz de programación. Esto promueve que los programas creados con Java sean autónomos y por tanto portables. Los binarios consiguientes de la compilación pueden ser ejecutados tal cual en Windows como Mac o Linux sin requisito de alterar ni una coma.
- **Rapidez:** desde la versión 6 de Java, la JVM contiene un compilador que compone código nativo en tiempo de ejecución de manera clara para el programador y el usuario. Por otra parte, durante esa compilación, se emplean técnicas de optimización más modernas del medio, y no sólo se fundamentan en cómo está escrito el código, sino también cómo ha sido la

última ejecución y cómo se supone estadísticamente que serán las subsiguientes iteraciones. Este conjunto de optimizaciones origina que Java posea tiempos equivalentes a programas escritos en C/C++.

- Gran ecosistema: Java ha exhibido una gran inserción en el comercio del desarrollo de software a nivel corporativo. Su madurez y fiabilidad ha estimulado que diversas empresas apunten a este lenguaje al momento de enfrentar sobresalientes inversiones en la construcción de sus propias plataformas de información. Como consecuencia de esa aceptación, varios programadores, empresas y organizaciones colaboran dinámicamente en la construcción de herramientas y librerías útiles (varias de ellas Open Source) para solucionar los inconvenientes usuales a los que debe confrontar el programador.
- Gran número de programadores: hallar programadores versados en Java es una labor más sencilla que encontrar a aquellos que dominan otros lenguajes menos establecidos.
- IDEs avanzados: Java ha predispuesto siempre del favor de acoger los IDEs más modernos del mercado, prácticamente su totalidad de uso gratuito favoreciendo a su colocación en las empresas. Estos IDEs facultan la compilación incremental, ayuda de escritura en código, refactorizaciones, exploración de documentación de API y diversas aplicaciones más.
- Continua evaluación: Desde que OpenJDK está al gobierno de Java y la JVM, el acuerdo del cuerpo que lo conforma es producir reiteradas versiones que integren los mejoramientos planteados por la comunidad mediante el

procedimiento Java Specification Requests (JSR), en donde se formulan, valúan e efectúan las solicitudes de nuevas características de Java o de la JVM, con el propósito de gozar de un marco especializado que se ajuste a las nuevas circunstancias y hardware que vaya brotando con el paso del tiempo. El propósito es que Java sea apto de resistir los retos tecnológicos tan inciertos como las nuevas arquitecturas web en la nube y asentados en contenedores virtualizados sin dejar de lado sus comienzos en sistemas embebidos y de tiempo real.

7.4.5 Spring Framework

Escrito originalmente por Rod Johnson, Spring Framework fue publicado en junio de 2003 bajo la licencia Apache 2.0, definiéndola, como una plataforma Java de código abierto. Erigiéndose en el framework más acreditado para Java empresarial, para la creación código fuente de alta productividad, ligero y reutilizable. Su propósito es generalizar, aligerar, manipular y solucionar los problemas que puedan ir brotando en el camino de la programación. Spring, brinda como parte importante el apoyo a infraestructura desde el nivel de aplicación, ofreciendo un modelo total para la configuración y la programación de aplicaciones empresariales Java, sin distinción en lo referente al despliegue de la plataforma. Todo esto conduce consigo una enorme utilidad, ya que faculta a los equipos de desarrollo para que puedan orientarse simplemente en la lógica empresarial que demanda la aplicación, estableciendo el proceso de una forma más corta, dinámica y efectiva, economizando líneas de código y esquivando tareas repetitivas. (Muradas, 2018)

Dentro de sus características podemos encontrar (Muradas, 2018):

- Tecnologías: inyección de dependencias, recursos, eventos, i18n, validación, enlace de datos, SpEL, conversión de tipo.
- Acceso a datos: JDBC, DAO, ORM, Marshalling XML.
- Integración: comunicación remota, JMS, JCA, JMX, correo electrónico, tareas, programación, caché.
- Pruebas: simulacro de objetos, TestContext, tests para Spring MVC, WebTestClient.
- Seguridad.
- Inyección de Dependencias: Al instante de construir una aplicación Java, las clases deben ser lo más autónomas posible de las otras clases, para acrecentar la probabilidad de poder reutilizarlas y probarlas libremente de las demás clases, mientras se examinan las unidades. Esencialmente la inyección de dependencias (DI) asiste en la unión y mantenimiento de estas clases.

Algunas de las ventajas de utilizar Spring Framework son (Muradas, 2018):

- Spring está constituido de forma modular. Sin importar la proporción de paquetes y clases que tenga la aplicación, solo debemos atender aquellos que requiramos para nuestro desarrollo y excluir el resto.
- Emplea las tecnologías que existen en el mercado, como diferentes frameworks ORM, JEE, temporizadores como Quartz y JDK, frameworks de registro y diferentes tecnologías de presentación.

- Probar una aplicación Spring es simple, ya que el código dependiente del ambiente se traspasa al framework. Asimismo, mediante el uso de JavaBeanstyle, se torna más sencillo la utilización de la inyección de dependencia para las pruebas, para lo que podemos usar datos “dummies” o “mocks”, para vislumbrar los resultados.
- El componente web del framework es un MVC bien planteado, que suministra una óptima opción para utilizar Struts u otros frameworks web como Thymeleaf o Vaadin.
- Proporciona una API para transcribir excepciones determinadas (por ejemplo, las producidas por JDBC, Hibernate o JDO) en excepciones sólidas y no verificadas.
- Los contenedores de Inversión de Control tienden a ser ligeros, en comparación con los Enterprise JavaBeans (EJB). Esto es perfecto para desarrollar y desplegar aplicaciones en dispositivos limitados de memoria y recursos.
- Permite separar la auditoría, el registro, la seguridad, las transacciones declarativas y, el almacenamiento en caché, de la lógica de negocio mediante la AOP (Programación Orientada a Aspectos).

7.4.6 Typescript

Es una tecnología de Microsoft para la construcción de aplicaciones Javascript a gran escala, robustas, que implementa particularidades en el lenguaje que permiten desarrollar herramientas más modernas para el desarrollo de aplicaciones. (Hernández, 2012)

TypeScript es un super conjunto de JavaScript. Indicamos que una tecnología es un super conjunto de un lenguaje de programación, cuando este puede interpretar programas de la tecnología, TypeScript en este caso, y del lenguaje del que es el super conjunto, JavaScript en este caso. Resumidamente, esto indica que los programas JavaScript son programas permitidos para TypeScript, sin importar que TypeScript sea un distinto lenguaje de programación. (Hernández, 2012)

El principal rasgo de TypeScript es el tipado estático, prestándose así a la implementación de herramientas de desarrollo más avanzadas, como (Hernández, 2012):

- Autocompletado de código.
- Recomendación de cuáles argumentos puede recibir una función.
- Recomendación de qué tipo de dato retorna una función.
- Auto documentación del código.
- Mejor análisis para la detección de errores.

7.4.7 Vaadin

Vaadin es un framework habilita la creación veloz de aplicaciones web en Java, sin necesidad de escribir HTML o JavaScript. (Vaadin Ltd., 2022) Vaadin facilita el proceso con una plataforma para el desarrollo de aplicaciones web integrada destinada a los

desarrolladores Java backends. (Vaadin Ltd., 2022) El framework corre del lado del servidor, por lo que gran parte de la lógica y la carga del trabajo sucede en el servidor. (Abellán, 2019)

Vaadin emplea Ajax del lado del cliente basado en GWT (Google Web Toolkit) para poder renderizar la interfaz de usuario en el navegador. Las aplicaciones Google Web Toolkit son escritas en Java y son traducidas automáticamente a JavaScript y HTML. GWT es muy conveniente para los desarrolladores porque le mantienen fuera de estas obligaciones relacionadas con la presentación, sin tener que instalar plugin alguno en el navegador para su labor. (Abellán, 2019)

Se pueden reconocer 4 componentes dentro de Vaadin:

- Componentes web: bloques visuales para formar la aplicación.
- Framework Java: Permite usar los componentes web desde el backend.
- Herramientas: Plugins Maven, librerías JavaScript, etc. El producto relevante se llama Designer, un diseñador visual para construir pantallas perfectas a base de componentes, haciendo estructuras de elementos web y creando enlaces entre las clases únicamente utilizando el ratón.
- Starters: Plantillas para la creación de nuevos proyectos. (Abellán, 2019)

7.4.8 Computación en la nube

Es la entrega de recursos de informática, comprendidos almacenamiento, procesamiento, redes, bases de datos, diagnósticos, inteligencia artificial y programas de software, por medio de Internet. Al exteriorizar estos recursos, las organizaciones pueden

suscribir a los activos de informática que precisan, cuando los requieran y sin adquirir o sostener una infraestructura física TI de manera local, facilitando recursos adaptables, rápida transformación y economías de escala. Para muchas compañías, el cambio a la nube está claramente conectado con la evolución de datos y TI. (Zettler, 2022)

Dentro de sus particularidades sobresalen (Zettler, 2022):

- Autoservicio bajo demanda: Los proveedores brindan APIs que los usuarios usan para evaluar recursos nuevos existentes cuando les sea necesario. Los equipos pueden automatizar el abastecimiento de la infraestructura forma simple con artefactos de infraestructura como código tales como Terraform o Ansible.
- Extenso acceso a la red: Significa una ventaja enorme al brindar hardware físico distribuido por todo el globo, facultando a las organizaciones suministrar de manera trascendental hardware establecido a la colocación.
- Congregación de recursos: Los recursos se reparten de manera dinámica, concedidos según sea solicitado. Puesto que el hardware de un host se suministra de dinámicamente y se reparten entre varios clientes, es optimizado de forma exhaustiva para su máximo uso.
- Veloz elasticidad: Las infraestructuras pueden aumentar y aminorar dinámicamente, facultando a los usuarios pedir que sus recursos aumenten de forma automática con las demandas de tráfico. La elasticidad puede acontecer en hardware individual, en el cual la adjudicación de recursos se acrecienta para incrementar los recursos libres, o en diferentes máquinas, en

donde los sistemas ascienden automáticamente a hardware ubicado en diversas redes.

- Evaluación del servicio: Los proveedores suministran métricas de uso minuciosas que se emplean para notificar sobre los gastos por uso.

Los beneficios que brinda la computación en la nube son (Zettler, 2022):

- Disminución de costos: Los equipos que utilizan este tipo de recursos no deben adquirir sus activos propios de hardware. Más allá de los costos de hardware, los proveedores realizan todo lo que está a su alcance para incrementar y perfeccionar la utilización del hardware. Asimismo, el hardware y demás recursos informáticos pasan a convertirse en productos, a la vez que los proveedores luchan por proporcionar excelentes resultados.
- Mayor escalabilidad: Ya que este tipo de computación es elástica por defecto, las empresas consiguen escalar los recursos por demanda, ofreciendo funciones para escalar de forma automática los equipos. Las aplicaciones, por tanto, pueden disminuir y acrecentar automáticamente los recursos de infraestructura en consecuencia de los picos de tráfico.
- Rendimiento optimizado: Brinda los mejores y más modernos recursos informáticos. Los usuarios pueden acceder al hardware más novedoso con CPU de múltiple núcleo, planteado para laboriosos trabajos de procesamiento en paralelo. Al mismo tiempo, los principales proveedores proporcionan hardware de GPU y TPU modernos para grandes misiones de procesamiento gráfico, inteligencia artificial y matrices.

- Mayor rapidez de ejecución: Los equipos suministrados pueden elaborar y entregar valor más velozmente a los usuarios. Los equipos de software ágil pueden valerse de este tipo de infraestructura para experimentar prontamente las máquinas virtuales más recientes con el objetivo de comprobar y aprobar conceptos originales, del mismo modo que, automatizar pruebas y las fases de ejecución.
- Seguridad fortalecida: Proporciona una infraestructura separada mediante el uso de firewalls que fortalecen la seguridad. Por otra parte, los proveedores proporcionan sinnúmero de componentes y conjuntos de técnicas de seguridad para asistir en la creación de sistemas seguros. El control de acceso es un inconveniente significativo en temas de seguridad, y la gran pluralidad de proveedores brindan instrumentos para restringir el acceso de los usuarios.
- Integración y entrega continuas: CI/CD es una práctica fundamental para DevOps que favorece el aumento de la velocidad de los equipos de desarrollo y acelera la presentación de los productos en el mercado. Cuando esta práctica está asentada en la nube, faculta a los equipos construir, valorar y ejecutar código de forma automática sin las preocupaciones concernientes a la administración o el sostenimiento de la infraestructura CI.
- Inspección integral y administración de incidencias: Brindan a los equipos la capacidad de enlazar sus sistemas de punta a punta, posibilitando la vigilancia de todos los segmentos del flujo de ejecución. La inspección integral es una facultad importante para las empresas que usen DevOps, dado que permite afrontar incidencias de forma rápida. Los proveedores

distribuyen métricas sobre el estado del sistema, por ejemplo, las concernientes a las aplicaciones y CPU de los servidores, memoria, número de solicitudes, tasa de fallas o tiempo promedio de respuesta.

7.4.9 Arquitectura de Microservicios

Los microservicios son una perspectiva de arquitectura de construcción de software en el cual este está erigido por servicios limitados y autónomos que se comunican entre sí por medio de API expuestas. Los dueños de los servicios son equipos no muy grandes y también autónomos. Este tipo de arquitecturas conciben aplicaciones fáciles de escalar y más ligeras para desarrollar, permitiendo a las organizaciones la innovación y celeridad sobre el tiempo de mercadeo para las características novedosas. (Amazon Web Services, Inc., 2022)

Con esta arquitectura, un sistema se construye con elementos autónomos que efectúan cada proceso como un servicio. Estos servicios realizan la comunicación por medio de una interfaz determinada mediante API's livianas. Los servicios se conciben para las competencias organizacionales donde cada servicio ejerce una única labor. Ya que son ejecutados de forma autosuficiente, cada servicio se puede renovar, ejecutar y escalar para complacer la demanda de funciones determinadas de un sistema. (Amazon Web Services, Inc., 2022)

Las principales propiedades de los microservicios son (Amazon Web Services, Inc., 2022):

- Autonomía: Cada servicio de esta arquitectura puede diseñarse, construirse, ejecutarse y escalar sin perjudicar el funcionamiento de los demás servicios,

ya que no es necesario compartir su código fuente o implementaciones entre servicios. Cualquier comunicación entre estos sucede a través de la API.

- Especialización: Cada servicio está esbozado para un grupo de competencias y se orienta en solucionar un problema concreto. Si los equipos de desarrollo contribuyen con más código fuente a un servicio durante el tiempo y el servicio se torna complicado, este se puede fraccionar en servicios más pequeños.

Los beneficios de emplear microservicios son (Amazon Web Services, Inc., 2022):

- Agilidad: Los microservicios promueven una distribución de equipos independientes y reducidos que se ocupan de los servicios, obrando a su vez en un entorno limitado y bien contenido, y están autorizados para producir autónoma y velozmente, disminuyendo los tiempos del proceso de desarrollo, favoreciendo de modo relevante el crecimiento de rendimiento en la organización.
- Escalado flexible: Los microservicios facultan a cada uno de los servicios para escalar de forma autónoma para cumplir con la demanda de la propiedad del sistema que resguarda, permitiendo a los equipos acomodarse a los requerimientos de infraestructura, calcular con exactitud los costos de la característica y conservar la disponibilidad en caso de que un servicio perciba un aumento en la demanda.
- Implementación simple: Los microservicios conceden la integración y la entrega continuas, facilitando a los equipos la capacidad de experimentar

nuevos conceptos y reestablecerlas en caso de que algo no funcione correctamente. El diminuto costo de los errores admite la realización de ensayos, posibilitando la modernización del código fuente y abreviando el tiempo de mercadeo de las nuevas particularidades.

- Libertad tecnológica: Las arquitecturas de microservicios no persiguen una dirección de "diseño único". Los equipos de desarrollo poseen la autonomía de preferir la herramienta que mejor les permita solucionar sus dificultades particulares.
- Reutilización del código: La segmentación del software en pequeños módulos y perfectamente descritos permite a los equipos de desarrollo utilizar funciones para diversos objetivos. Un servicio construido para una función establecida puede emplearse como componente de otra característica, permitiendo que una aplicación inicie independientemente, puesto que los desarrolladores pueden construir capacidades nuevas sin escribir código fuente desde cero.
- Resistencia: La autonomía del servicio acrecienta la defensa de un sistema a la presencia de errores. En una arquitectura monolítica, un error en un solo módulo provoca un error la aplicación completa, mientras que, con microservicios, en caso de presentarse un error en el servicio, las demás aplicaciones lo manipulan disminuyendo la operatividad sin paralizar todo el sistema.

7.4.10 Docker

Docker es una plataforma de software que permite construir, ensayar y ejecutar aplicaciones ágilmente, empackando software en elementos normalizados llamados contenedores que contienen todo lo indispensable para que el software funcione, abarcando librerías, herramientas de sistema, código y entornos de ejecución. Con Docker, es posible ejecutar y adaptar el escalamiento de las aplicaciones prontamente en cualquier tipo de entorno con la seguridad de entender que la ejecución del código se efectuará correctamente. (Amazon Web Services, Inc., 2022)

Docker permite suministrar código con mayor celeridad, estandarizando los procedimientos de los sistemas, trasladando el código fácilmente y economizando capital a las organizaciones, optimizando la utilización de recursos, produciendo un único objeto que puede ejecutarse confiablemente en cualquier infraestructura compatible. Docker tiene una fácil sintaxis y sencilla la cual contribuye a tener un absoluto control. La extensa aceptación por parte de la comunidad de desarrollo representa la existencia de un inmenso ambiente de instrumentos y programas preparados para su utilización con Docker. (Amazon Web Services, Inc., 2022)

Se pueden emplear los contenedores Docker como unidad de construcción fundamental en el momento de construir modernas aplicaciones y plataformas, ya que posibilita la elaboración y la realización de arquitecturas de microservicios distribuidos, la ejecución de código mediante integración y entrega continuas normalizadas, la formación

de sistemas de procesamiento de datos inmensamente escalables y la elaboración de entornos totalmente gestionados por sus desarrolladores. (Amazon Web Services, Inc., 2022)

7.4.11 PostgreSQL

PostgreSQL es una eficaz base de datos relacional de código abierto con 30 años de desarrollo eficiente que le ha otorgado una férrea popularidad debido a su confiabilidad, capacidad de funciones y alto rendimiento. (The PostgreSQL Global Development Group, 2022) Cimentado en POSTGRES, Versión 4.2, y construido en el Departamento de Ciencias de la Computación de la Universidad de California en Berkeley, este sistema fue precursor en muchas nociones que estuvieron disponibles en algunas bases de datos comerciales tiempo después. (The PostgreSQL Global Development Group, 2022)

PostgreSQL es proyecto de código abierto, heredero del código original de Berkeley, compaginable con el estándar SQL y el cual brinda muchas modernas particularidades, como por ejemplo: (The PostgreSQL Global Development Group, 2022)

- Consultas complejas.
- Llaves foráneas.
- Disparadores.
- Vistas actualizables.
- Integridad transaccional.

- Control de concurrencia multi-versión: Esta característica adiciona una imagen del estado a cada una de las transacciones, permitiendo que estas sean fortuitamente consistentes, entregando notables beneficios en rendimiento. (Dorantes, 2022)

Asimismo, el usuario puede extender PostgreSQL de muchas formas, por ejemplo, adicionando nuevos: (The PostgreSQL Global Development Group, 2022)

- Tipos de datos.
- Funciones.
- Operadores.
- Funciones agregadas.
- Métodos de índice.
- Lenguajes de procedimiento.

Y gracias a la licencia libre, cualquiera puede utilizar, alterar e intercambiar PostgreSQL gratuitamente para cualquier intención, tanto privado, comercial e incluso didáctico. (The PostgreSQL Global Development Group, 2022)

7.4.12 Google App Engine Standard

Google App Engine es un ecosistema de Google Cloud que faculta la creación de aplicaciones web que son renderizadas en el servidor. App Engine acepta lenguajes de programación bastante populares con una diversidad de herramientas para los desarrolladores. (Google Cloud, 2022)

Las características relevantes de App En son: (Google Cloud, 2022)

- **Lenguajes de programación populares:** Se pueden construir aplicaciones en Java, Ruby, C#, Node.js, Go, Python e incluso PHP. Esto habilita a los programadores para conservarse rápidos debido a la afinidad con lenguajes de desarrollo populares y la extensa diversidad de herramientas para desarrolladores.
- **Administración total:** Con un ambiente totalmente gestionado, los desarrolladores se pueden concentrar en el código, mientras que los inconvenientes de infraestructura se solventan en App Engine, librando a los desarrolladores con construcciones que no demandan configuración y también ofrece administración sin servidores.

El entorno estándar de App Engine está basado en instancias de contenedores que son ejecutados en la infraestructura propia de Google. Estos contenedores son configurados anticipadamente con uno de los diferentes ambientes de ejecución disponibles. El entorno estándar proporciona la compilación y la implementación de una aplicación para que sea ejecutada de forma confiable, inclusive, con pesadas acometidas y amplias porciones de datos. Las aplicaciones son ejecutadas en un ambiente de pruebas totalmente seguro, lo cual permite que el entorno estándar reparta las solicitudes entre distintos servidores y los escale para respetar las demandas de tráfico. La aplicación es ejecutada dentro de su entorno exclusivo que es seguro, confiable e independiente del hardware, sistema operativo o ubicación física del hardware. (Google Cloud, 2022)

7.4.13 Spring Cloud Gateway

Corresponde a un proyecto de Spring que suministra una librería para la construcción de un API Gateway sobre Spring WebFlux. Este proyecto abraiga como meta fundamental proveer una manera sencilla y segura de enrutar las API y ofrecerles características transversales, como: seguridad, monitoreo/métricas y resiliencia. (VMware, Inc., 2021)

Dentro de la lista de propiedades se puede notar: (VMware, Inc., 2021)

- Basado en Spring Framework 5, Project Reactor y Spring Boot 2.0
- Capaz de concordar rutas en cualquier característica de solicitud.
- Los predicados y los filtros son propios de las rutas.
- Composición de disyuntores.
- Integración Spring Cloud DiscoveryClient
- Predicados y filtros sencillos de programar.
- Restricción de tasa de solicitud.
- Reescritura de ruta.

7.4.14 Cloud SQL for PostgreSQL

Cloud SQL es el servicio de bases de datos relacionales de Google totalmente gestionado para MySQL, PostgreSQL y SQL Server con conjuntos de extensiones mejoradas, marcas de configuración y ambientes para desarrolladores. (Google Cloud, 2022)

Las funciones clave de Cloud SQL son: (Google Cloud, 2022)

- Completamente administrado: De manera automática garantiza que las bases de datos sean veraces, seguras y escalables para que las organizaciones sigan

trabajando sin complicaciones. Cloud SQL automatiza todas las labores referentes a copias de seguridad, replicación, parches y ampliaciones de capacidad. Asimismo, certifica una disponibilidad de más del 99.95% en todo el planeta.

- **Integrado:** Se puede acceder a las instancias de Cloud SQL desde prácticamente cualquier sistema, tales como, App Engine, Compute Engine, Kubernetes Engine y estaciones de trabajo. Cloud SQL brinda facultades de análisis y estadísticas mediante BigQuery para explorar directamente en las bases de datos.
- **Confiable:** Permite la configuración de la replicación y de las copias de seguridad sencillamente para resguardar la información. También se puede autorizar la conmutación por error de forma automática para que las bases de datos estén, en gran medida, disponibles. Los datos son cifrados automáticamente, y Cloud SQL acata los estándares SSAE 16, ISO 27001 y PCI DSS, además de la HIPAA.
- **Migraciones simples a Cloud SQL:** Database Migration Service (DMS) proporciona la migración de las bases de datos de producción a Cloud SQL con un tiempo de inactividad imperceptible. Esta propuesta sin servidores anula los obstáculos de abastecimiento, gestión e inspección manual de los recursos de migración. DMS utiliza las capacidades nativas de replicación de MySQL y PostgreSQL para aumentar la exactitud y fiabilidad de la migración. Conjuntamente, DMS está disponible sin costo extra para las migraciones nativas a Cloud SQL.

PostgreSQL es la base de datos relacional de código abierto puntera en la industria, la cual tiene un ambiente operativo y en aumento de desarrolladores y herramientas. Con Cloud SQL para PostgreSQL, los equipos de desarrollo pueden destinar menos esfuerzo a las operaciones de bases de datos de PostgreSQL y más a la construcción de aplicaciones. (Google Cloud, 2022)

Las aplicaciones fundamentales del servicio de Cloud SQL para PostgreSQL son: (Google Cloud, 2022)

- **Máxima compatibilidad:** Se puede continuar utilizando herramientas habituales, como pgAdmin, pgwatch2 y Percona. También se puede conseguir ayuda total para las diferentes versiones principales de PostgreSQL.
- **Observabilidad de la base de datos para DevOps:** Se puede solucionar ágilmente los inconvenientes de rendimiento de las bases de datos en Cloud SQL para PostgreSQL por medio de Insights, identificando el origen de las dificultades con paneles prediseñados y los planes de consultas visuales, accediendo rápidamente a las métricas y seguimientos de la base de datos mediante OpenTelemetry.
- **Mantenimiento en sus términos:** El mantenimiento acucioso y automatizado conserva las instancias parcheadas y constantes. Los administradores pueden recibir notificaciones con días de adelanto para que puedan proyectar mejor el mantenimiento y aplicación de controles.
- **Integraciones principales incorporadas:** Se puede utilizar Cloud SQL desde casi cualquier sistema, conectándose sencillamente desde Compute Engine,

Kubernetes Engine, Cloud Run y estaciones de trabajo, proporcionando flujos de cambios desde tus bases de datos de Oracle y MySQL a Cloud SQL con Datastream. También se pueden consultar los datos con Looker y realizar análisis con BigQuery para examinar las bases de datos.

- Migraciones rápidas, fáciles y seguras: Con el servicio de migración de bases de datos se puede realizar este procedimiento desde PostgreSQL u Oracle a partir de un ambiente local, Compute Engine u otros proveedores a Cloud SQL para PostgreSQL con un tiempo de inactividad exiguo.

7.4.15 Google Cloud Storage

Cloud Storage es el servicio de almacenamiento de objetos de Google para todo tipo de organizaciones, el cual guarda cualquier cantidad de datos y permite acceder a estos cuando el cliente lo requiera. (Google Cloud, 2022)

Sus características principales son: (Google Cloud, 2022)

- Migración de datos sencilla a tipos de almacenamiento más ahorrativas: Permite configurar los datos mediante la función de gestión del ciclo de vida de objetos (OLM), de tal forma que pasen de forma automática a tipos de almacenamiento de menor costo cuando las condiciones establecidas se cumplan. Por ejemplo, si los objetos han estado retenidos durante un tiempo explícito o si se almacena una versión más reciente de estos.
- Diferentes opciones de redundancia: Cloud Storage tiene una amplia lista de ubicaciones de almacenamiento donde las organizaciones pueden almacenar

sus datos con diferentes opciones de redundancia automática, pudiendo los clientes tipificar dónde y cómo almacenar los datos.

- **Fácil transferencia de datos a Cloud Storage:** El servicio de transferencia de Storage se distingue por un óptimo rendimiento, el cual brinda acceso en línea a Cloud Storage, con la escalabilidad y la rapidez precisas para la simplificación del proceso de transferencia de datos. Para que se pueda transferir datos sin conexión, el servicio Transfer Appliance suministra un servidor de almacenamiento que se une al centro de datos de la organización y, después, se remite a una ubicación de ingestión donde los datos son exportados a Cloud Storage.
- **Almacenamiento de archivado:** Con una poca latencia y la misma API de Cloud Storage, los tipos Archive y Coldline brindan un almacenamiento “en frío” que las organizaciones pueden usar para archivado de continuidad empresarial.
- **Tipos de almacenamiento para diferentes cargas de trabajo:** Las organizaciones pueden almacenar los datos en diferentes tipos de almacenamiento con el fin de disminuir costos sin perjudicar de forma negativa el rendimiento.

7.4.16 Google Cloud Pub/Sub

El servicio Pub/Sub de Google Cloud habilita a los servicios la posibilidad que se comuniquen entre sí asíncronamente, con latencias en promedio de 100 milisegundos, el cual se emplea para las integración, transferencia y distribución de datos, teniendo la misma

efectividad que un sistema dirigido a la mensajería para integración de servicios o como cola para la paralelización de tareas. (Google Cloud, 2022)

Pub/Sub facilita la construcción de sistemas publicadores y suscriptores de eventos. Los publicadores se manifiestan con los suscriptores de asincrónamente a través de la transmisión de eventos, enviándolos al servicio, sin convenir cómo o cuándo deben ser procesados. Posteriormente, los eventos son entregados a los servicios que reaccionan a estos. Esta integración de forma asíncrona acrecienta la elasticidad y la estabilidad integral de los sistemas de información. (Google Cloud, 2022)

8. Diseño metodológico o metodología

8.1 Scrum

Scrum es un paradigma de gestión que disminuye las dificultades en la construcción de productos para complacer los requerimientos de los clientes. La gerencia y los equipos de Scrum producen en conjunto en torno a los requisitos y tecnologías para proporcionar productos operando de forma incremental empleando el empirismo. Scrum es un marco de trabajo sencillo que impulsa la cooperación dentro de los equipos para alcanzar la construcción de productos complicados. Ken Schwaber y Jeff Sutherland son los autores de “La Guía Scrum” que especifica Scrum de manera comprensible y fácil. (Huambachano, 2017)

Scrum es un modelo inteligible, no una enorme recopilación de fragmentos y mecanismos forzosos determinados de forma prescriptiva. Scrum no es una técnica en sí, más bien, está asentado en un patrón empírico, con consideración por las personas y establecido en la organización autónoma de los equipos para enfrentarse con lo inadvertido y solventar dificultades enrevesadas, examinando y adecuando permanentemente. (Huambachano, 2017)

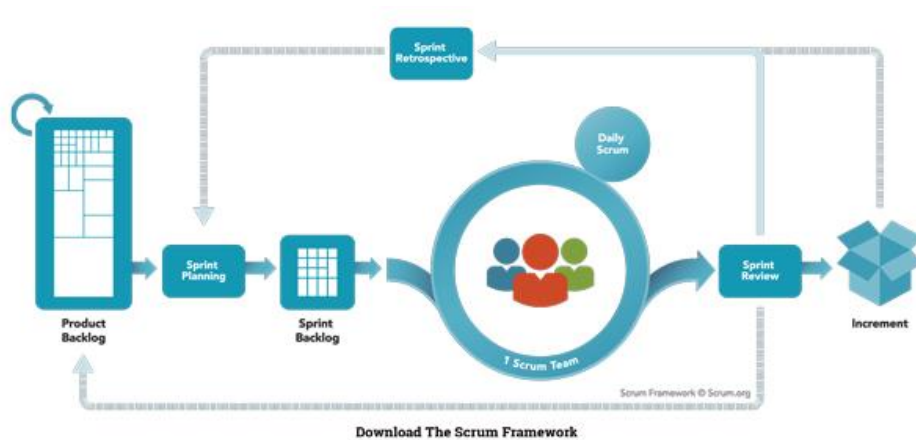


Ilustración 1 Framework Scrum (Scrum.org, 2020)

8.1.1 Roles

- Propietario del producto: Este integrante del equipo ejerce como vínculo entre el equipo de desarrollo y los clientes, además, es el encargado de asegurar que se hayan informado y convenido las expectativas respecto al producto acabado.
- Scrum master: Este participante del equipo interviene como facilitador, siendo el delegado de confirmar que se lleven a efecto las mejores prácticas de Scrum y que el plan pueda prosperar.
- Equipo de desarrollo: Este es el conjunto de individuos que contribuyen en agrupación para construir y ensayar entregas incrementales del producto terminal. (TechTarget, 2020)

8.1.2 Eventos

- Scrum diario: Es una reducida reunión que se realiza en la misma parte y hora todos los días, en la cual, el equipo comunica el trabajo que se concluyó el día antes y planea el trabajo a realizar en las futuras 24 horas. Este es el

instante en que los integrantes del equipo pueden conversar sobre algún inconveniente que pueda frenar la terminación del proyecto.

- Reunión de planificación de sprint: Un sprint alude al espacio de tiempo en el que se debe concluir el trabajo y, frecuentemente, es de 30 días. Todos cooperan en la constitución de los objetivos y, al término de esta, se debe formar al menos un incremento, en otras palabras, una fracción de software aprovechable.
- Revisión de sprint. Este es el instante de mostrar el incremento.
- Retrospectiva del sprint. Es una reunión llevada a cabo posteriormente a la finalización de un sprint, donde todos deliberan sobre el desarrollo del sprint. Asimismo, se puede brindar un entrenamiento de organización de equipos. Un objetivo trascendental de la retrospectiva es la mejora continua. (TechTarget, 2020)

8.1.3 Artefactos

- Aglomeración de productos: Describe a lo que permanece en la lista de “por hacer”. A lo largo de una sesión de organización de la aglomeración de productos, el equipo de desarrollo actúa con el propietario del producto para anteponer el trabajo que se ha atrasado. La aglomeración de productos puede acomodarse en el transcurso un procedimiento nombrado refinamiento de la aglomeración.
- Aglomeración de Sprint: Es una enumeración de labores que se deben acabar antes de que se puedan proporcionar los ítems del inventario de productos

seleccionados, los cuales se fragmentan en historias de usuarios establecidas en el tiempo.

- **Incremento de productos:** Representa lo que se ha conseguido durante un Sprint (todos los ítems de la aglomeración de productos), y también, lo que se ha elaborado a lo largo de todos los pasados Sprints. El incremento de productos evidencia cuánto se ha progresado.
- **Agotamiento:** Es una imagen de la porción de trabajo que aún se debe terminar. El gráfico cuenta con un eje Y (tarea) y un eje X (periodo), y explica una disposición a la baja, ya que la cantidad de actividades que permanecen por hacer con el periodo de tiempo se disminuye a cero. (TechTarget, 2020)

8.2 Aplicación

Para el proyecto se establecerán Sprints de 2 semanas, en los cuales se realizarán eventos de planeación, revisión, retrospectiva y refinamiento, además de la construcción del sistema. Se espera que, para los eventos de revisión al final del Sprint, un representante de ISAMCO SAS se reúna con el equipo de trabajo, ya sea de forma presencial o virtual, para presentar los incrementos logrados para su consideración, retroalimentación y aprobación.

La acumulación (backlog) del producto y de los diferentes Sprints, al igual que las diferentes historias de usuario, serán registradas mediante una herramienta web disponible en línea a través de internet (ej. Jira) para consulta tanto de la Gerencia de ISAMCO SAS como para el registro y seguimiento del equipo de trabajo.

Dado que el equipo de trabajo consiste únicamente en los autores del presente documento, el rol de propietario del producto recaerá en alguno los miembros según su disponibilidad para esta tarea del Sprint en desarrollo y podrá ser reasignado para el siguiente Sprint, como sea requerido. El rol de Scrum máster no tendrá un responsable específico, y las actividades correspondientes serán responsabilidad de los miembros del equipo de desarrollo.

9. Proceso de Software

9.1 Requerimientos Funcionales (Historias de Usuario)

9.1.1 RF001: Administración de Conductores

Como administrativo o conductor, quiero que el sistema permita gestionar la información de los conductores, para realizar una correcta administración de sus contratos, seguros, viajes y posteriores liquidaciones de comisiones.

9.1.2 RF002: Administración de Vehículos

Como administrativo, quiero que el sistema admita información de los diferentes vehículos pertenecientes a ISAMCO, permitiendo la asignación a conductores contratados, para la ejecución de viajes y trayectos correspondientes al transporte de carbón.

Como conductor, quiero que el sistema permita gestionar la administración de los mantenimientos, listas de chequeos y, seguros asociados a los vehículos asignados, para tener la información al día del estado de los vehículos.

9.1.3 RF003: Administración de Viajes y Trayectos

Como administrativo, quiero que el sistema habilite la planeación de viajes y trayectos a llevar a cabo por parte de los conductores utilizando los vehículos de ISAMCO para el transporte de carbón, para realizar la planeación adecuada del trabajo operativo de la empresa.

Como conductor, quiero que el sistema tenga la opción para realizar las correcciones necesarias sobre los registros de novedades, gastos e ingresos, para actualizar las novedades

generadas en caso de que se consideren necesarios y mantener la información correcta respecto a los viajes.

9.1.4 RF004: Registro de Novedades de Viajes y Trayectos

Como conductor, quiero que el sistema tenga la opción de novedades correspondientes a los diferentes viajes y trayectos realizados, admitiendo de estos también la inscripción de ingresos y gastos que hayan incurrido durante la ejecución de los trayectos, para mantener la información actualizada y proveer los datos correctos y necesarios para la posterior liquidación de honorarios.

9.1.5 RF005: Gestión de Hoja de Vida de Vehículos

Como conductor, quiero que el sistema provea las opciones necesarias para mantener actualizada la hoja de vida del vehículo asignado, permitiendo registrar novedades correspondientes a mantenimientos realizados, para tener la información de los vehículos asignados al día y reportar el estado de estos.

Como administrativo, quiero que el sistema permita realizar cualquier actualización y/o corrección sobre la hoja de vida que se consideren oportunos, para asegurar que la información de los vehículos sea la correcta y se encuentre actualizada.

9.1.6 RF006: Gestión de Alertas

Como administrativo, quiero que el sistema habilite la realización de programación de alarmas, según se considere importante, sobre diferentes cumplimientos de las distintas entidades del modelo del negocio que conlleven atributos de vencimientos, tales como seguros para conductores y vehículos, viajes, trayectos y, mantenimientos de vehículos, para

que los conductores y demás administrativos estén atentos a los eventos a realizar y eventualidades ocurridas.

Como conductor, quiero que el sistema notifique alertas sobre los diferentes eventos a realizar asociados con los viajes y los vehículos, para estar atento y listo para la realización de las actividades planeadas y cumplir con estas.

9.1.7 RF007: Consulta de Reportes

Como gerente general, quiero que el sistema genere reportes que permitan consultar información esencial respecto a la operación de los conductores y vehículos, la ejecución de los trayectos y viajes, y los datos de ingresos y egresos asociados a la ejecución operativa, para mantener la información precisa respecto a los resultados de la operación de la empresa y poder tomar las decisiones estratégicas necesarias para la mejora continua de la organización.

9.2 Requerimientos No Funcionales

9.2.1 RNF001: Comprobabilidad

El sistema deberá estar diseñado e implementado de tal forma que realizar pruebas automatizadas como manuales sea fácil y rápido, permitiendo detectar cualquier error a tiempo y habilitando al equipo técnico a realizar las correcciones necesarias sobre los contenedores correspondientes para minimizar su impacto.

9.2.2 RNF002: Confiabilidad

El sistema deberá proveer información confiable a la gerencia, administrativos y conductores de ISAMCO, ya que de los datos del sistema se obtendrán los cálculos e

informes necesarios para la gestión del negocio y la correcta liquidación de las operaciones realizadas.

9.2.3 RNF003: Evolucionabilidad

El sistema deberá contar con un diseño e implementación que le permita adaptarse ágilmente a nuevos requerimientos, ya sean por petición del negocio o para el cumplimiento de nuevas normativas, facultando así a ISAMCO mantener sus procesos actualizados en apoyo del sistema de información y en un futuro apoyar el crecimiento del negocio y ser referentes del mercado de carbón en la región.

9.2.4 RNF004: Modularidad

El sistema deberá poseer y conservar un diseño e implementación modular, cuyos contenedores representen los diferentes aspectos del negocio de ISAMCO, y que a su vez permita la adaptabilidad técnica y de negocio a cambios y nuevos requerimientos.

9.2.5 RNF005: Elasticidad

El sistema deberá contar con un diseño e implementación que reconozca la demanda de recursos técnicos y de infraestructura, con el fin de habilitar la gestión de estos ya sea de forma manual o automática, para satisfacer la experiencia de los diferentes usuarios del sistema.

9.2.6 RNF006: Escalabilidad

El sistema deberá contar con un diseño e implementación que se adecue al crecimiento de recursos técnicos y de infraestructura necesarios para soportar una mayor

operación del negocio, ayudando a mantener los objetivos presupuestales de ISAMCO para tal fin.

9.2.7 RNF007: Facilidad de despliegue

El sistema deberá disponer de un diseño e implementación configurado en los contenedores necesarios y, a su vez, estos configurados de tal forma que sea permitido su despliegue de forma fácil y rápida, sin interrumpir la operación del negocio, y que habilite la implantación de correcciones y nuevas funcionalidades de forma ágil.

9.3 Diseño y Arquitectura

9.3.1 Restricciones

1. El sistema deberá ser una aplicación web, que pueda ser accedida mediante internet por los usuarios utilizando un navegador de escritorio o móvil.
2. El sistema deberá proveer un API REST cuyas solicitudes y respuestas se encuentren en formato JSON, para que a futuro nuevas aplicaciones clientes y/o integraciones con terceros puedan realizarse de forma fácil y con el menor número de inconvenientes.
3. El sistema deberá contar con certificado digital para que todas sus transacciones expuestas a internet estén bajo el protocolo seguro HTTPS.

9.3.2 Priorizaciones

1. El sistema deberá estar disponible para piloto operativo a finales del segundo semestre de 2022.

2. El sistema deberá tener disponible para el piloto todas las operaciones que puedan ser soportadas desde un navegador web de escritorio.

3. El sistema deberá contar dentro de su diseño e implementación con las opciones necesarias para el cumplimiento de las normativas de custodia de información.

4. El sistema deberá tener disponible para el piloto el acceso a los módulos necesarios para el registro de información prioritaria mínima para la gestión del negocio, expuesta a través de reportes a consultar por parte de la gerencia general y personal administrativo.

9.3.3 Escenarios

A continuación, se describen los escenarios del sistema.

9.3.4 Actores

9.3.5 Gerente

Corresponde al gerente general de ISAMCO. Dado que su rol dentro de la empresa es netamente administrativo de alto nivel, sus principales escenarios estarán asociados con consulta de información (conductores, vehículos, viajes) y reportes requeridos para apoyar su toma de decisiones (gastos, ingresos, comisiones, viajes).

9.3.6 Administrativa

Recae sobre la asistente de la gerencia general de ISAMCO. Su rol es bastante importante, ya que es el puente tanto en la operación como en la gestión entre los conductores y la gerencia; esta función también se verá reflejada en el sistema, ya que será la delegada de la administración de las principales entidades del modelo del negocio (vehículos, viajes, trayectos, listas de chequeo, alarmas), buscando garantizar que la

información consignada corresponda a la realidad, ya que estos datos son los que alimentarán posteriormente los reportes consultados por la gerencia y también la generación de los datos utilizados para la liquidación de los conductores.

9.3.7 Conductor

Concierne a los conductores contratados por ISAMCO para la operación de los vehículos y la realización de los viajes para transporte de carbón. Los conductores son el principal rol operativo dentro del negocio de transporte y, por tanto, los alimentadores primordiales de información del sistema, ya que serán los encomendados en el registro y actualización de la información de su información personal, de su vehículo y sus listas de chequeo, de los viajes y sus trayectos, de los gastos e ingresos, remisiones y registros de báscula.

9.3.8 Administrador del Sistema

Concierne a un miembro del personal de ISAMCO encargado del soporte técnico de primer nivel para el sistema. Es el encargado de realizar las acciones de borrado lógico de las entidades del modelo dentro del sistema (viaje, conductor, vehículo) con el fin de certificar que sólo han sido eliminados por petición de la gerencia y/o administración y que no sean suprimidos por accidente.

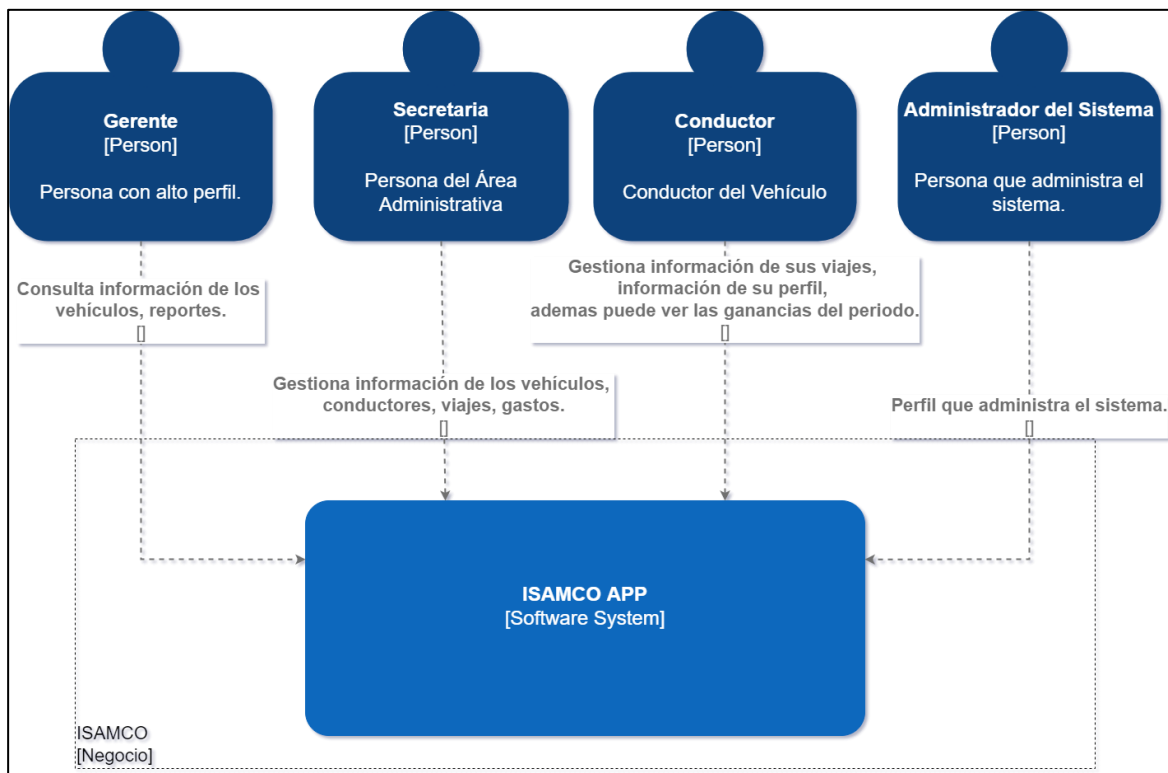


Ilustración 2 Diagrama de Contexto C4 (Autores, 2022)

9.4 Diagrama de Despliegue

Cada uno de los diferentes contenedores del frontend y del backend del sistema estarán empaquetados en archivos JAR par que a través de Google App Engine Standard sean agregados a contenedores Docker para su fácil configuración y despliegue.

En el caso del frontend, el contenedor estará construido mediante una imagen de OpenJDK y NodeJS, para la correcta ejecución de la interfaz Vaadin.

Para cada uno de los contenedores del backend, estarán fabricados con imágenes que contengan OpenJDK versión 11 y los archivos JAR ejecutables de cada proyecto, los cuales a su vez contienen las clases implementadas y sus correspondientes dependencias.

Cada uno de los contenedores Docker anteriormente mencionados serán generados y desplegados en la nube de Google Cloud Platform, mediante el uso de Google App Engine Standard.

En lo que corresponde al API expuesto del sistema, este será gestionado mediante con contenedor construido con Spring Cloud Gateway.

Para las notificaciones del sistema, estas son recibidas a través de un endpoint del contenedor correspondiente y colocadas en la cola de Publicación/Suscripción alojada en Pub/Sub. El mismo contenedor, a través de tareas programadas consultan los eventos de notificación y envía los mensajes correspondientes, garantizando así que ninguna notificación se quede sin enviar ante cualquier eventualidad del contenedor o el sistema en general.

Las bases de datos estarán desplegadas utilizando Cloud SQL for PostgreSQL dentro del entorno de Google Cloud Platform.

Finalmente, la gestión del almacenamiento de los documentos cargados y generados por el sistema se realizará mediante el uso de Google Cloud Storage.

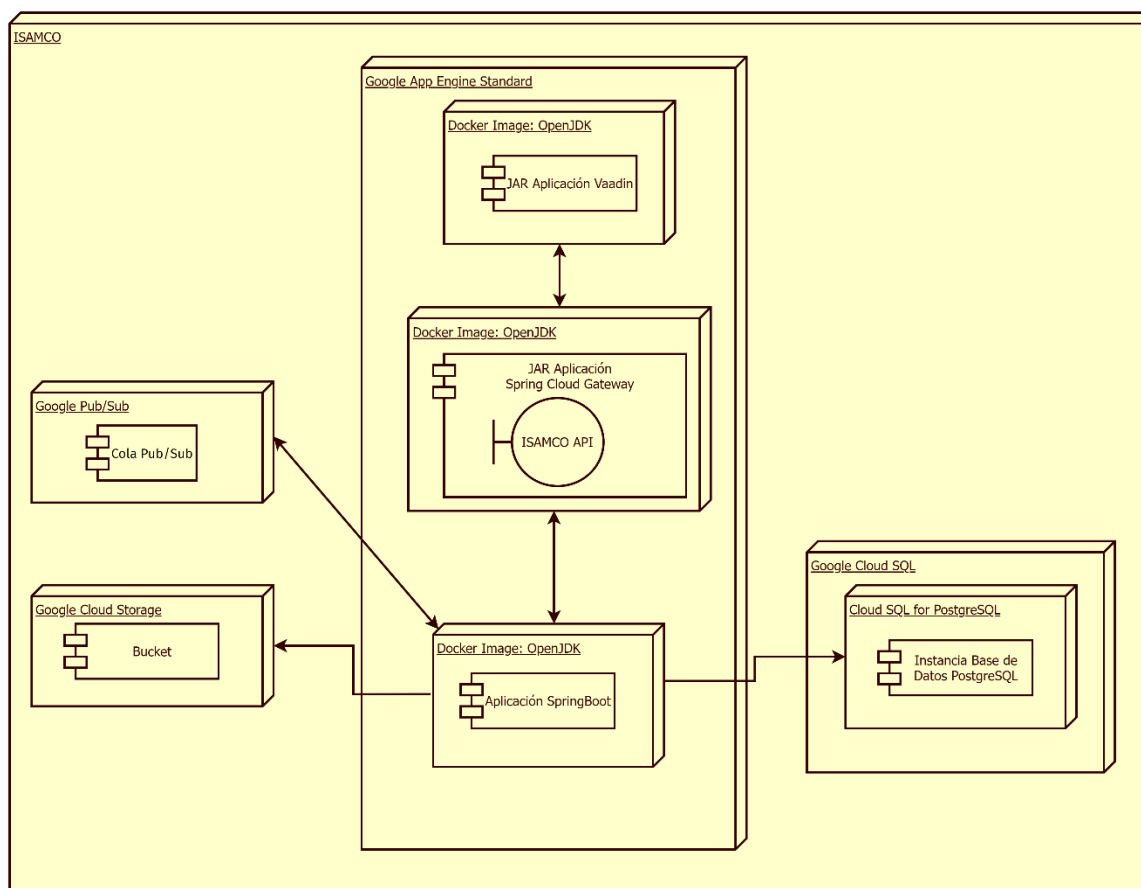


Ilustración 3 Diagrama UML de Despliegue General (Autores, 2022)

9.5 Caso de Uso Arquitecturalmente Relevante

Las funcionalidades arquitecturalmente relevantes del sistema para ISAMCO SAS, se componen de 2 casos de uso para la administración de vehículos: Asignación de Vehículos y Consulta de Vehículos.

El escenario general, está principalmente atendido por dos de los contenedores del sistema: isamco-cp-ui, contenedor de la interfaz gráfica e, isamco-cp-service, contenedor del backend encargado de la administración del parque vehicular del sistema.

A nivel general, el contenedor de UI activa una clase del paquete “view”, el cual a su vez llama una instancia de clase del paquete “service” que representa un cliente REST

encargado de la invocación y obtención de los resultados solicitados a la capa de negocio. El contenedor del backend, recibe la petición, mediante una clase del paquete “controller” que expone el servicio correspondiente y que invoca la instancia de una clase del paquete “service” la cual contiene la lógica de negocio necesaria para el procesamiento de la solicitud. Esta clase de lógica de negocio está constituida por una interfaz que precisa las funcionalidades sobre las cuales los controladores y las demás clases de negocio pueden acceder, y una clase “handler” de servicio que aplica la implementación de las reglas de negocio definidas. Esta clase, se apoya para sus operaciones persistentes en una instancia del paquete “handler”, la cual utiliza una instancia de clase del paquete “repository” para el acceso a la capa de base de datos y una instancia de clase del paquete “mapper” para realizar los procesos de transformación entre la entidad persistente y el objeto de dominio utilizado para recibir y enviar las peticiones y resultados desde el backend al frontend.

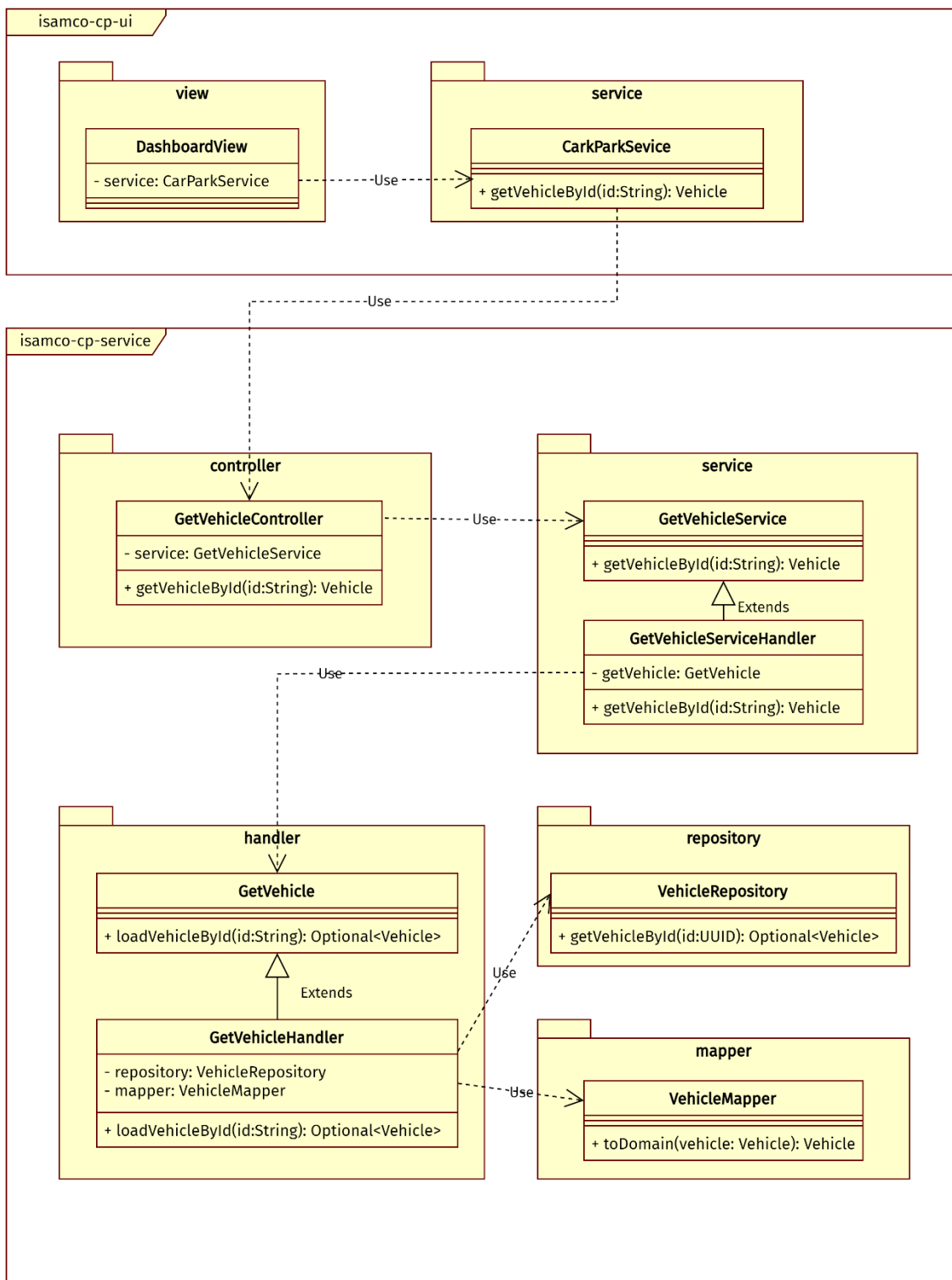


Ilustración 4 Diagrama de Clases Consulta de Vehículos (Autores, 2022)

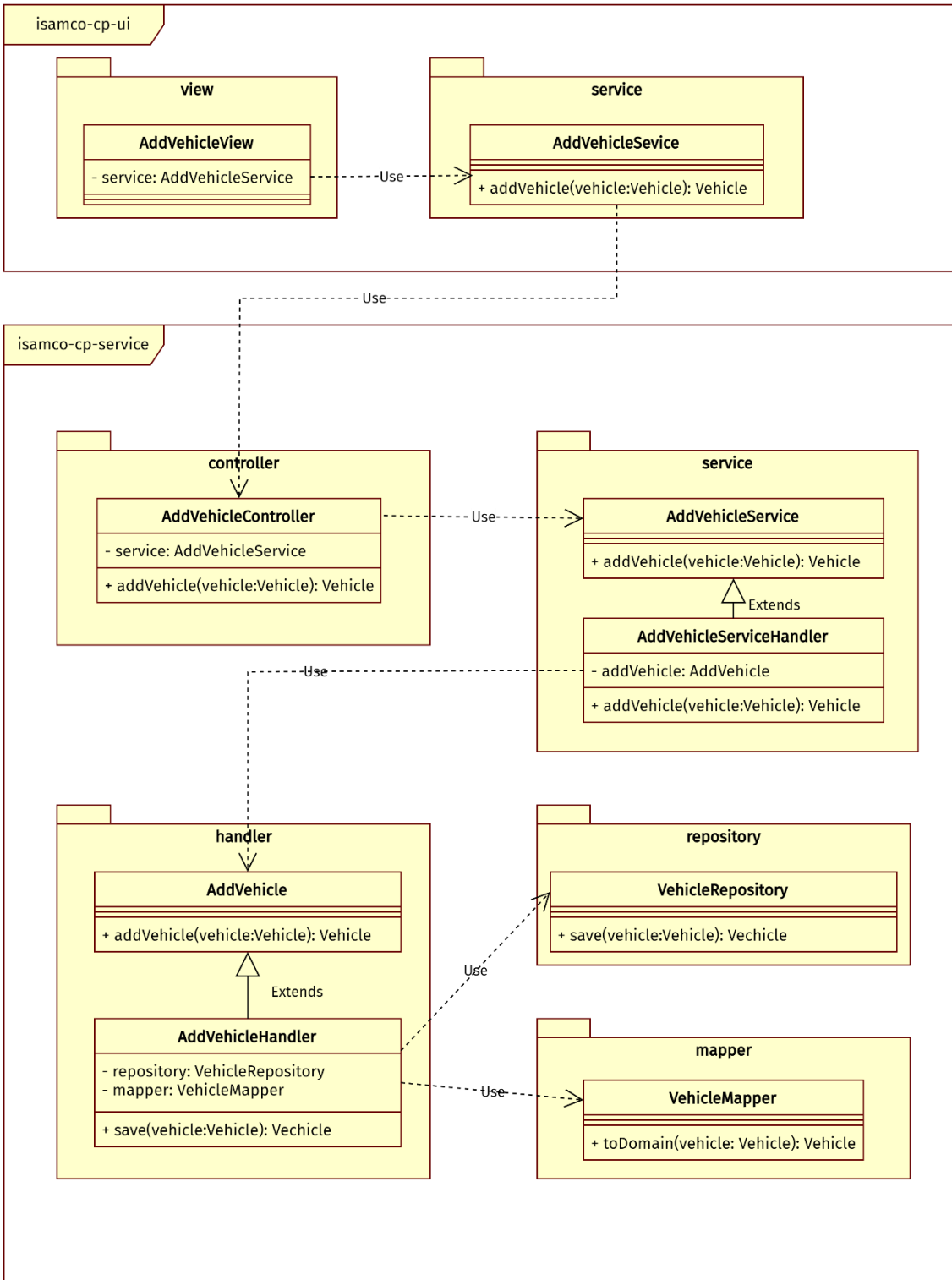


Ilustración 5 Diagrama de Clases Asignación de Vehículos (Autores, 2022)

9.6 Diagrama de Secuencia

Respecto a la secuencia definida para los escenarios del caso arquitecturalmente relevante, para el caso de la asignación de vehículos, la vista `AddVehicleView` invoca el método `addVehicle()` del servicio `AddVehicleService`; este a su vez, utiliza una petición REST vía HTTPS para comunicarse con el controlador del backend, `AddVehicleController`, petición recibida por su función `addVehicle()`, método POST. La petición es validada y enviada a la clase de lógica de negocio `AddVehicleService`, por medio del método `addVehicle()`, aplicando las reglas de negocio definidas para este escenario. La asignación de vehículo es persistida por la clase `AddVehicle` a través del método `save()` de la clase `VehicleRepository` y transformada la respuesta al objeto de dominio utilizando el método `toDomain()` de la clase `VehicleMapper`. El objeto de dominio es retornado por la misma vía de llamados hasta el controlador, el cual lo expone como un objeto JSON, el cual, a su vez, luego es transformado a un Value Object dentro del contenedor de la UI para la presentación de los resultados.

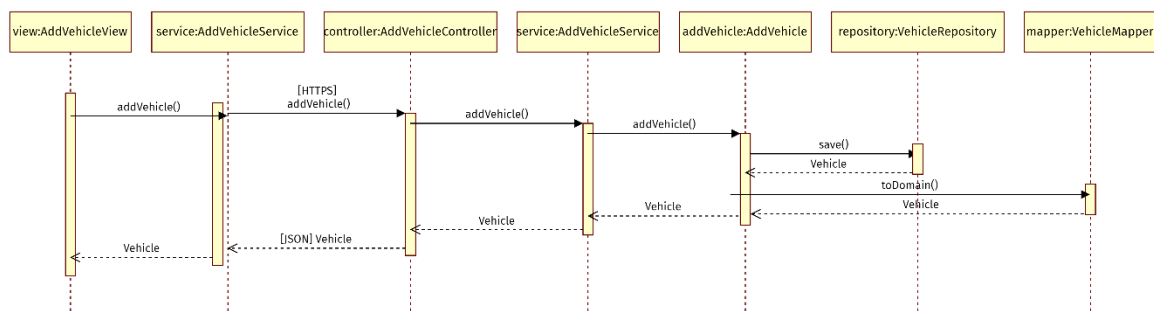


Ilustración 6 Diagrama de Secuencia Asignación de Vehículos (Autores, 2022)

Continuando con la “receta” de arquitectura definida para la construcción de los casos de uso, la consulta de vehículos está formada por el proceso encabezado por la vista `DashboardView`, la cual llama el método `getVehicleById()` del servicio `CarParkService`; este

a su vez, usa una solicitud REST vía HTTPS para transmitir la petición al controlador backend, `GetVehicleController`, recogida por su función `getVehicleById()`, método GET. Esta es legitimada y remitida a la clase de negocio `GetVehicleService`, método `getVehicleById()`, empleando las instrucciones de negocio determinadas. La consulta es realizada sobre la capa persistente por la clase `GetVehicle` a través del método `loadVehicleById()` de la clase `VehicleRepository` y convertida a una instancia de dominio aplicando el método `toDomain()` de la clase `VehicleMapper`. Dicha instancia del dominio es regresada por el mismo camino de invocaciones hasta llegar al controlador, el cual lo exterioriza en una representación JSON, que posteriormente es transmutada a un `Value Object` por contenedor de la UI para la publicación de los resultados.

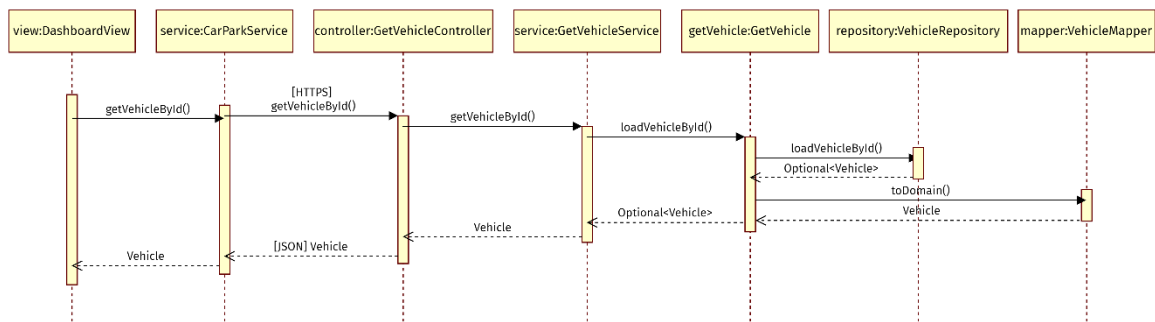


Ilustración 7 Diagrama de Secuencia Consulta de Vehículos (Autores, 2022)

9.7 Diagrama de Clases

La lógica de negocio será gestionada por los contenedores del backend, los cuales a su vez estarán constituidos por los diferentes módulos del dominio del negocio, los cuales se describen a continuación:

9.7.1 Vehículos

Comprende la información básica de los vehículos y su hoja de vida de mantenimientos. Contiene las siguientes clases del dominio:

1. MarcaVehiculo
2. ModeloVehiculo
3. Vehiculo
4. HojaVidaVehiculo
5. MantenimientoVehiculo
6. ListaChequeoVehiculo
7. ItemChequeoVehiculo
8. ChequeoVehiculo
9. RespuestaChequeoVehiculo
10. SeguroObligatorio
11. Poliza

9.7.2 Ubicación Geográfica

Abarca la información correspondiente a las ubicaciones de origen y destino de los viajes y trayectos a realizar por los vehículos. Comprende las siguientes clases del dominio:

1. TipoUbicacion
2. Ciudad
3. Departamento
4. Ubicación

9.7.3 Viajes

Incluye la información de los viajes y los trayectos que los componen, al igual que el registro de todos los documentos y novedades generados durante el proceso. Involucra las siguientes clases del dominio:

1. Viaje
2. Trayecto
3. TipoEgresoTrayecto
4. EgresoTrayecto
5. TipoIngresoTrayecto
6. IngresoTrayecto
7. TipoEventoTrayecto
8. EventoTrayecto
9. TipoDocumentoCarga
10. DocumentoCarga

9.7.4 Empleados

Encierra toda la información correspondiente a los empleados administrativos y conductores asociados a ISAMCO. Implica las siguientes clases del dominio:

1. Empresa
2. Contrato
3. Empleado
4. Conductor

9.7.5 Alertas

Corresponde a la información resultado de la parametrización de las diferentes alarmas a notificar a los usuarios por vencimientos. Contiene las siguientes clases del dominio:

1. Alerta
2. AlertaVehiculo
3. AlertaTrayecto

9.7.6 Minerales

Administra la información de los minerales transportados por los vehículos de ISAMCO. Ocupa las siguientes clases del dominio:

1. Mineral

9.8 Arquitectura de Alto Nivel

9.8.1 Visualización del Sistema

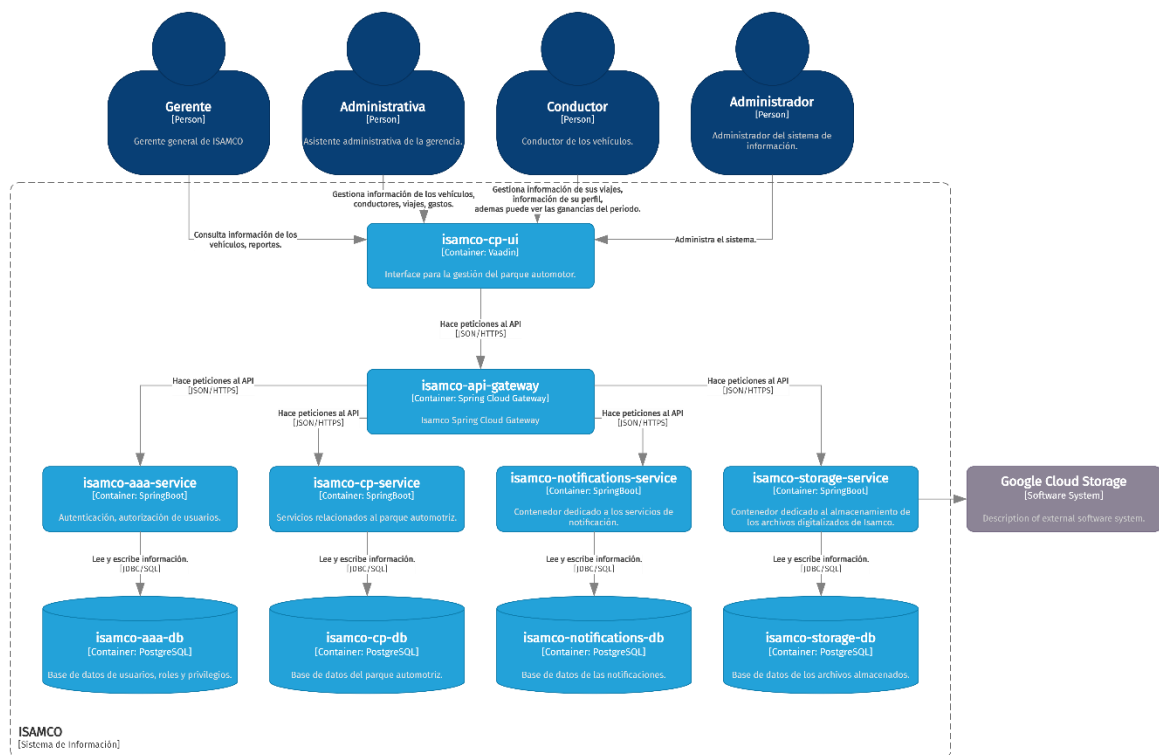


Ilustración 9 Diagrama de Contenedores C4 (Autores, 2022)

El sistema estará compuesto por los siguientes contenedores:

- **Aplicación Cliente Web:** Este contenedor corresponderá a una aplicación web que servirá como cliente para recibir las peticiones de los diferentes usuarios del sistema y a su vez, presentará las respuestas a dichas peticiones, sean exitosas o de error. Este contenedor recibirá peticiones y respuestas mediante el protocolo HTTPS y únicamente tendrá conexión del lado del sistema con el API Gateway.

- **API Gateway:** Este contenedor recaerá sobre un servicio tercero administrado que expondrá la API del sistema a internet para que los clientes puedan acceder a los datos, la lógica de negocio y la funcionalidad del backend. Para el sistema de ISAMCO, el API Gateway únicamente recibirá peticiones de la aplicación cliente web, mediante el protocolo HTTPS y recibirá y retornará información en formato JSON.
- **Aplicaciones Backend:** Son un conjunto de contenedores que en su totalidad conforman el backend del sistema, encargado del procesamiento de los datos y la lógica de negocio de ISAMCO para la gestión de conductores, vehículos y viajes para el transporte de carbón. Estos contenedores no estarán expuestos a internet y únicamente ofrecerán sus funcionalidades al exterior mediante microservicios expuestos a través del API Gateway. Cada uno de estos contenedores recibirá peticiones y retornará resultados en formato JSON. De igual forma, cada uno de los contenedores administrará su información en su propia base de datos. El conjunto de contenedores del backend estará compuesto por:
 - **Contenedor AAA:** Gestionará la sesión de los usuarios, la autorización y autenticación de estos. También administrará toda la información y datos relacionados con los empleados administrativos y los conductores asociados a ISAMCO.
 - **Contenedor de Vehículos:** Gestionará toda la información y procesos de administración de los vehículos de ISAMCO, a los viajes y los trayectos

efectuados por los conductores en sus vehículos asignados para el transporte de carbón.

- **Contenedor de Notificaciones:** Gestionará toda la configuración y parametrización de las diferentes notificaciones asociadas a vencimientos de seguros, mantenimientos, viajes y trayectos y de su correspondiente envío a los usuarios destinatarios.
- **Contenedor de Almacenamiento:** Gestionará el almacenamiento persistente de los diferentes documentos pertenecientes a los diferentes procesos de información soportados por el sistema.
- **Sistema de Almacenamiento:** Contenedor externo correspondiente a un servicio de almacenamiento de objetos para la persistencia y consulta de los documentos gestionados por el contenedor de almacenamiento del sistema.
- **Bases de datos:** Contenedor de persistencia de la información gestionada por cada uno de los contenedores del backend. La relación entre contenedores de backend y contenedores de bases de datos será de 1:1. Estos contenedores únicamente tendrán comunicación con su contenedor de backend correspondiente.

Dentro de los módulos implementados para el sistema de información de ISAMCO, se encuentran los siguientes:

ISAMCO
Módulo de acceso

Usuario •
test1

Contraseña •
.... 

Ingresar

[Recuperar contraseña](#)

Ilustración 10 Pantalla de Login (Autores, 2022)

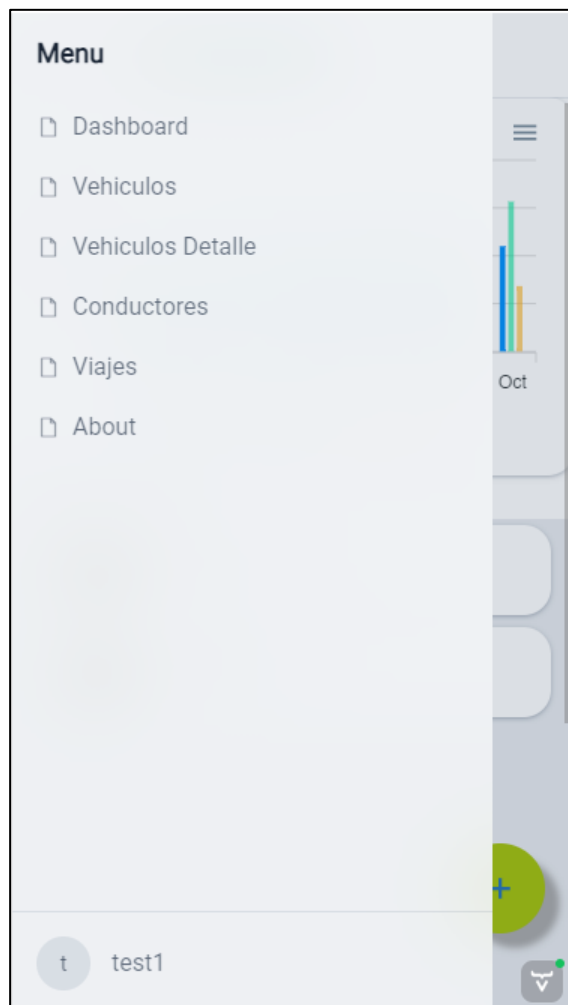


Ilustración 11 Pantalla de Menú Principal (Autores, 2022)

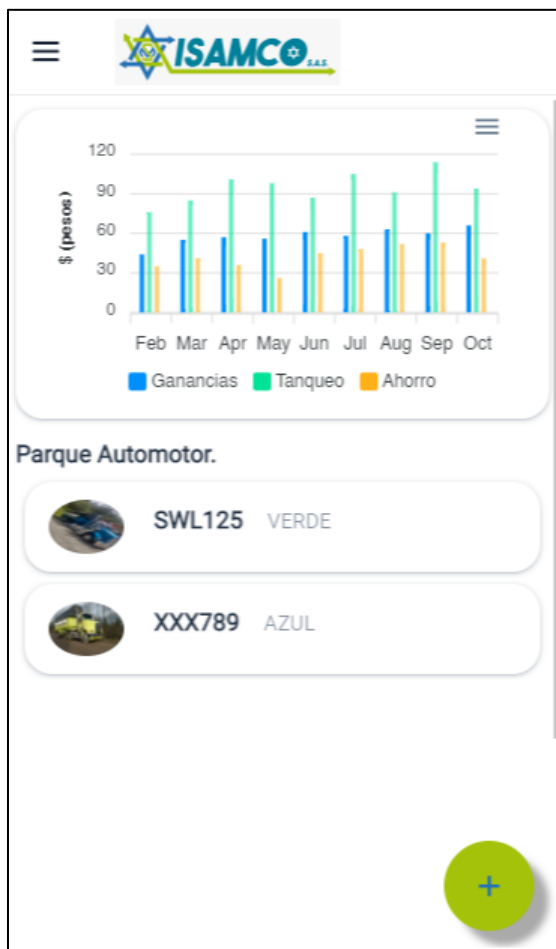
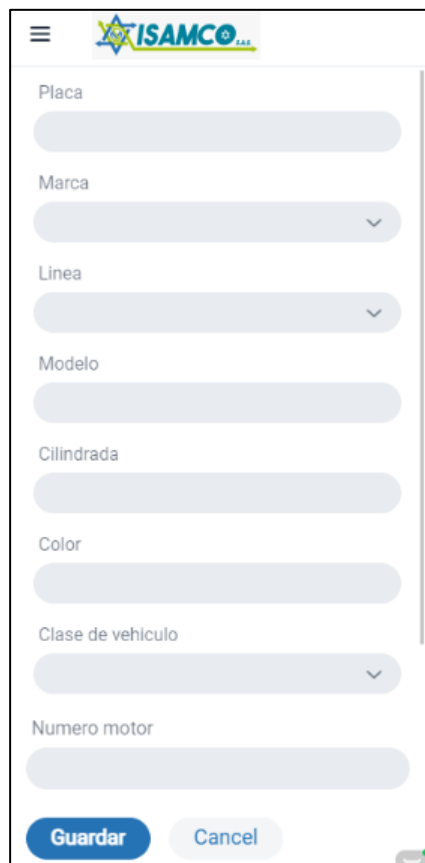


Ilustración 12 Pantalla de Dashboard (Autores, 2022)



The image shows a mobile application interface for a vehicle registration form. At the top left, there is a hamburger menu icon. To its right is the logo for 'ISAMCO S.A.' which consists of a stylized star and the company name. Below the header, the form contains several input fields, each with a label and a corresponding input area:

- Placa:** A text input field.
- Marca:** A dropdown menu with a downward arrow.
- Linea:** A dropdown menu with a downward arrow.
- Modelo:** A text input field.
- Cilindrada:** A text input field.
- Color:** A text input field.
- Clase de vehiculo:** A dropdown menu with a downward arrow.
- Numero motor:** A text input field.

At the bottom of the form, there are two buttons: a blue button labeled 'Guardar' and a light gray button labeled 'Cancel'. The entire form is enclosed in a thin black border.

Ilustración 13 Formulario de Vehículos (Autores, 2022)

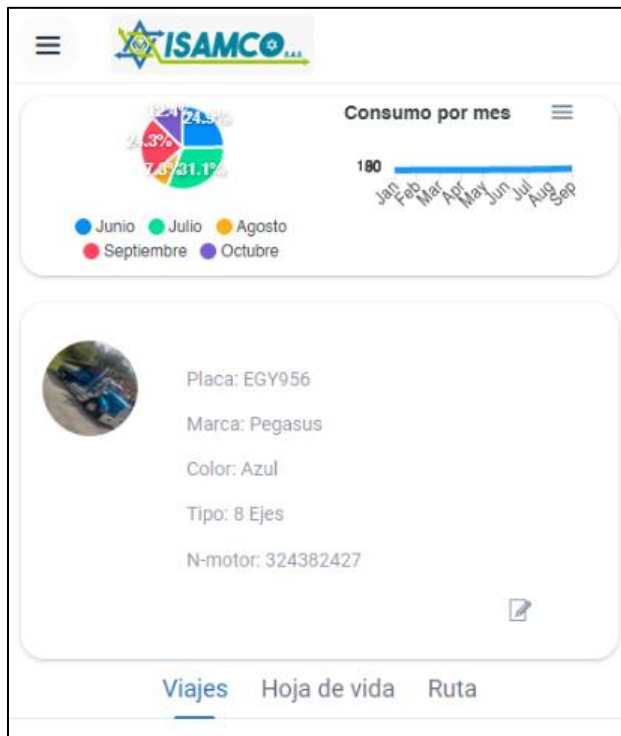


Ilustración 14 Pantalla de Vehículos (Autores, 2022)

The image shows a mobile application interface for creating a new trip. At the top, there is a menu icon and the ISAMCO S.A.S. logo. The title is 'Nuevo Viaje'. Below the title, there are several input fields: 'Origen', 'Destino', 'Fecha origen' (with a calendar icon), 'Fecha destino' (with a calendar icon), 'Odo 1', 'Odo 2', and 'Manifiesto'. At the bottom, there are two buttons: 'Guardar' (blue) and 'Cancelar' (light blue).

Ilustración 15 Formulario de Viajes (Autores, 2022)

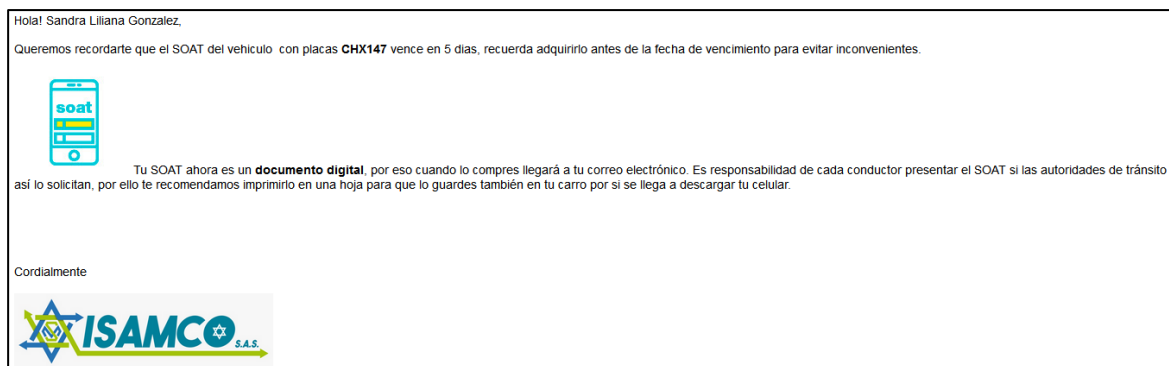
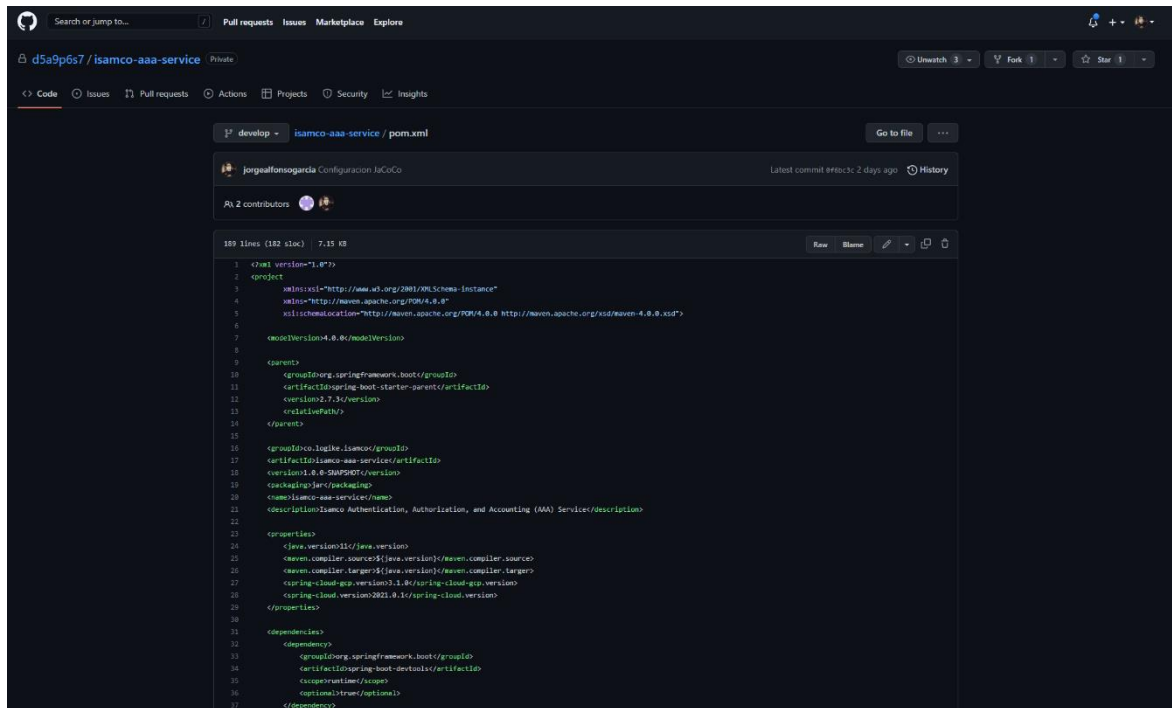


Ilustración 16 Notificación por Correo Electrónico (Autores, 2022)

10. Construcción

Para cada uno de los proyectos respectivos de los contenedores, se utilizaron proyectos Java, gestionados a través de archivos POM de Apache Maven, para la definición de dependencias y asignación de pasos de pruebas y construcción del ejecutable. La

principal dependencia utilizada para todos los proyectos fue Spring Boot versión 2. Cada uno de los contenedores se construyó como un archivo JAR, ejecutable y autocontenido, o sea, que el servidor de aplicaciones se encuentra dentro del JAR, a la vez que la misma aplicación.



```

1 <?xml version="1.0"?>
2 <project
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xmlns="http://maven.apache.org/POM/4.0.0"
5   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
6
7   <modelVersion>4.0.0</modelVersion>
8
9   <parent>
10    <groupId>org.springframework.boot</groupId>
11    <artifactId>spring-boot-starter-parent</artifactId>
12    <version>2.7.3</version>
13    <relativePath>
14    </parent>
15
16    <groupId>com.isamco</groupId>
17    <artifactId>isamco-aaa-service</artifactId>
18    <version>1.0.0-SNAPSHOT</version>
19    <packaging>jar</packaging>
20    <name>isamco-aaa-service</name>
21    <description>isamco Authentication, Authorization, and Accounting (AAA) Service</description>
22
23    <properties>
24      <java.version>11</java.version>
25      <maven.compiler.source>${java.version}</maven.compiler.source>
26      <maven.compiler.target>${java.version}</maven.compiler.target>
27      <spring-cloud-gcp.version>3.1.0</spring-cloud-gcp.version>
28      <spring-cloud.version>2021.0.1</spring-cloud.version>
29    </properties>
30
31    <dependencies>
32      <dependency>
33        <groupId>org.springframework.boot</groupId>
34        <artifactId>spring-boot-starter</artifactId>
35        <scope>runtime</scope>
36        <optional>true</optional>
37      </dependency>

```

Ilustración 17 pom.xml (Autores, 2022)

Para la ejecución dentro del entorno de Google App Engine, y también para el proceso de desarrollo y pruebas locales, cada uno de los contenedores posee un archivo Dockerfile a partir de una imagen del OpenJDK disponible en Docker, y desde esta se toma la copia del ejecutable construido como el punto de entrada de ejecución.

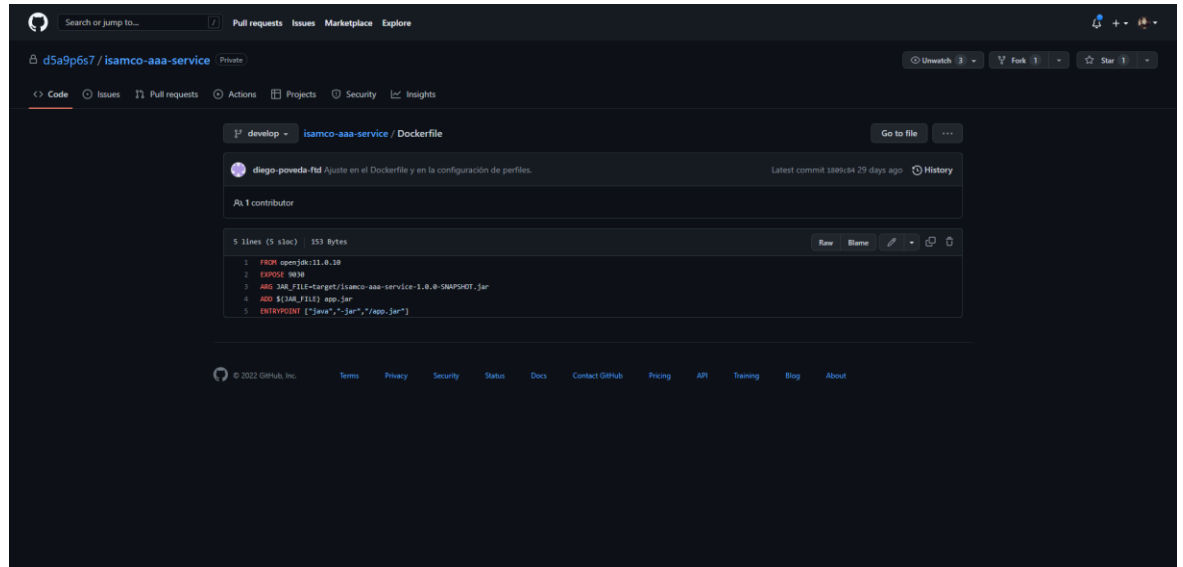


Ilustración 18 Dockerfile (Autores, 2020)

Con el fin de buscar refrendar la calidad del código fuente de cada uno de los proyectos, se utilizó la herramienta Sonar para la validación estática de código, obtención de posibles errores, y así poder corregirlos y lograr la validación de cada uno de los desarrollos efectuados.

The screenshot displays the SonarQube interface for a project named 'isamco-aaa-service'. The top navigation bar includes links for Projects, Issues, Rules, Quality Profiles, Quality Gates, and Administration. A search bar is present on the right. The main content area is divided into several sections:

- QUALITY GATE STATUS:** A green box indicates 'Passed' with the note 'All conditions passed.'
- MEASURES:**
 - New Code:** 2 Bugs (Reliability: C), 0 Vulnerabilities (Security: A), 2 Security Hotspots (Security Review: E, 0.0% Reviewed), 1h 47min Debt (Maintainability: A), and 11 Code Smells (Maintainability: A).
 - Overall Code:** 66.8% Coverage on 304 Lines to cover (Unit Tests: 54), 0.0% Duplications on 1.3k Lines (Duplicated Blocks: 0).
- ACTIVITY:** A section for 'Choose graph type' (set to 'Issues') with a message: 'There isn't enough data to generate an activity graph.'

At the bottom, a yellow warning banner states: 'Embedded database should be used for evaluation purposes only. The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.'

Footer text: 'SonarQube™ technology is powered by SonarSource SA. Community Edition - Version 9.7 (build 61563) - LGPLv3 - Community - Documentation - Plugins - Web API'

Ilustración 19 Evaluación Sonar con errores (Autores, 2022)

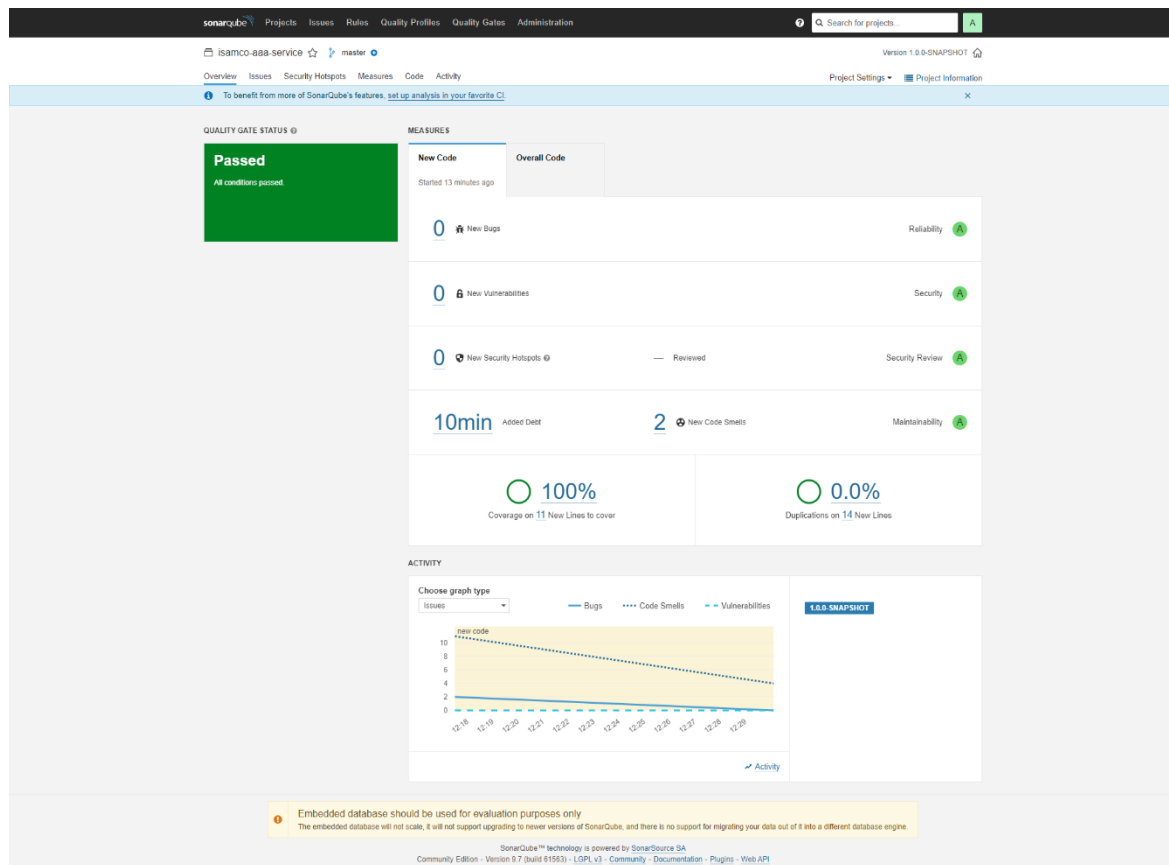


Ilustración 20 Evaluación Sonar después de corrección (Autores, 2022)

El proceso de desarrollo estuvo centralizado en repositorios privados de GitHub, sobre los cuales el equipo de desarrollo realizaba un “fork”, creaba una rama, según el desarrollo a realizar, aplicando el estándar de GitFlow, y luego solicitando un Pull Request, que era revisado y aprobado por el otro miembro del equipo de desarrollo.

Search or jump to... Pull requests Issues Marketplace Explore

d5a9p67 / isamco-aaa-service Private

Code Issues Pull requests Actions Projects Security Insights

[ISAMCO-23] AAA - PRUEBAS UNITARIAS #1

Open Jorgealfonsogarcia wants to merge 3 commits into d5a9p67:develop from Jorgealfonsogarcia:feature/unit-testing

Conversation Commits Checks Files changed +1,411 -330

Jorgealfonsogarcia commented now

TIQUETE JIRA

ISAMCO-23
AAA - PRUEBAS UNITARIAS

DESCRIPCIÓN

- Actualización y corrección de pruebas unitarias.

Jorgealfonsogarcia added 3 commits 10 hours ago

- Actualización de unit tests.
- Actualización nombres unit tests.
- Configuracion JACO

Jorgealfonsogarcia added the `enhancement` label now

Jorgealfonsogarcia requested a review from d5a9p67 now

Add more commits by pushing to the `feature/unit-testing` branch on `Jorgealfonsogarcia/isamco-aaa-service`

Review requested
Review has been requested on this pull request. It is not required to merge. [Learn more](#) [Show all reviewers](#)

1 pending reviewer

This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request You can also open this in GitHub Desktop or view command line instructions.

Write Preview

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Close pull request Comment

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

ProTip! Add `.patch` or `.diff` to the end of URLs for Git's plaintext views.

© 2022 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About

Ilustración 21 Pull Request (Autores, 2022)

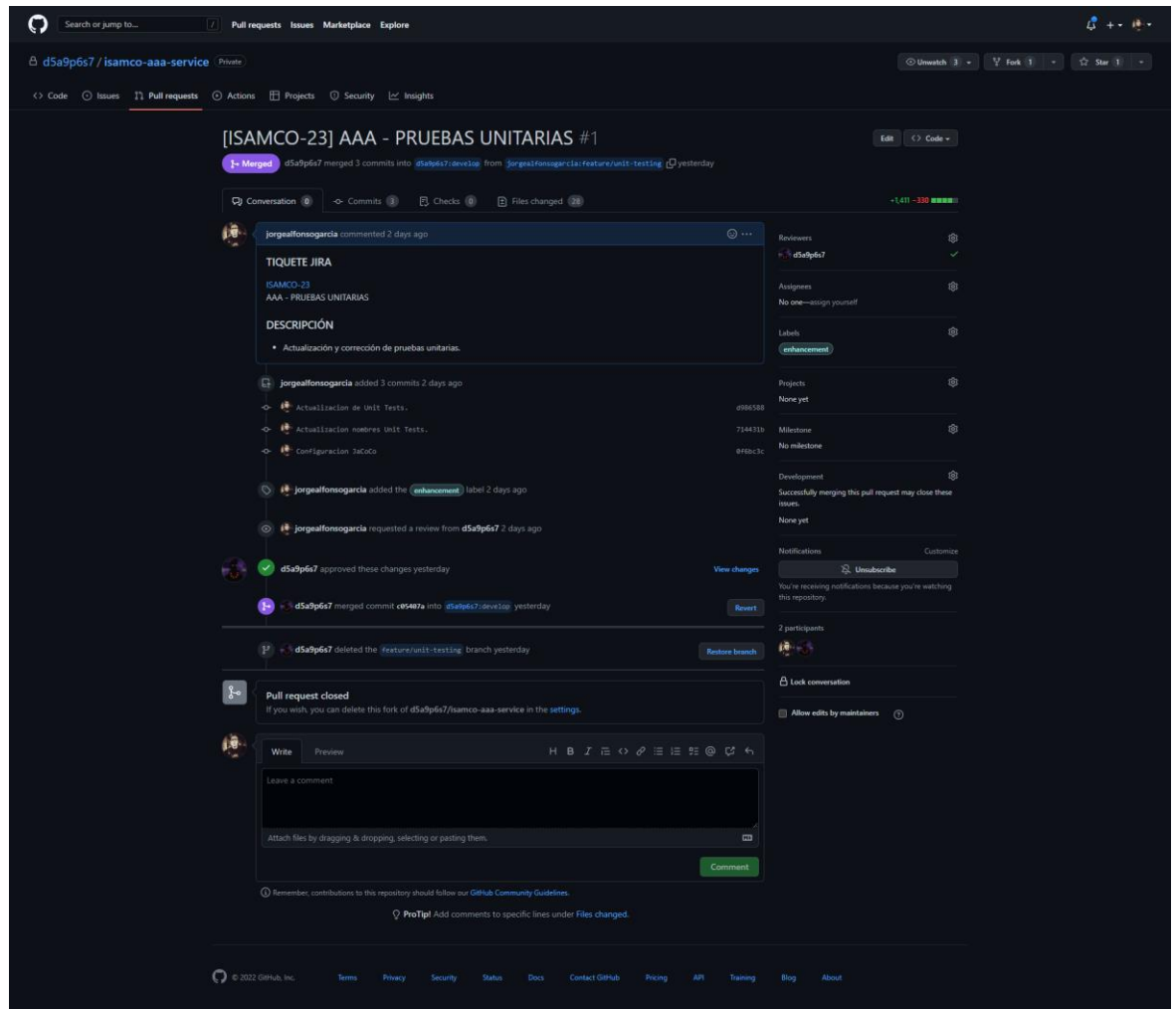


Ilustración 22 Pull Request Aprobado (Autores, 2022)

Todos los desarrollos estaban definidos según lo establecido en Jira, herramienta utilizada para la gestión del proyecto, con base a la metodología Scrum. Allí, se definió una serie de épicas, correspondientes a los diferentes casos de uso y requerimientos funcionales del sistema, de los cuales se desprendían una serie de historias de usuario, que inicialmente quedaban consignadas en el backlog y luego eran asignadas en los diversos sprints de trabajo.

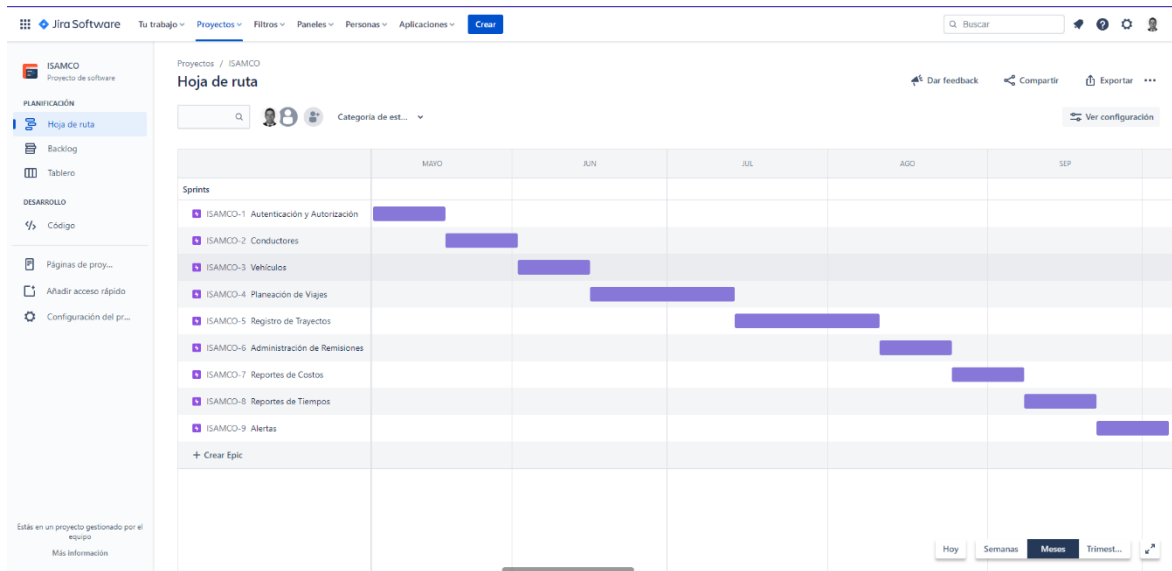


Ilustración 23 Hoja de Ruta en Jira (Autores, 2022)

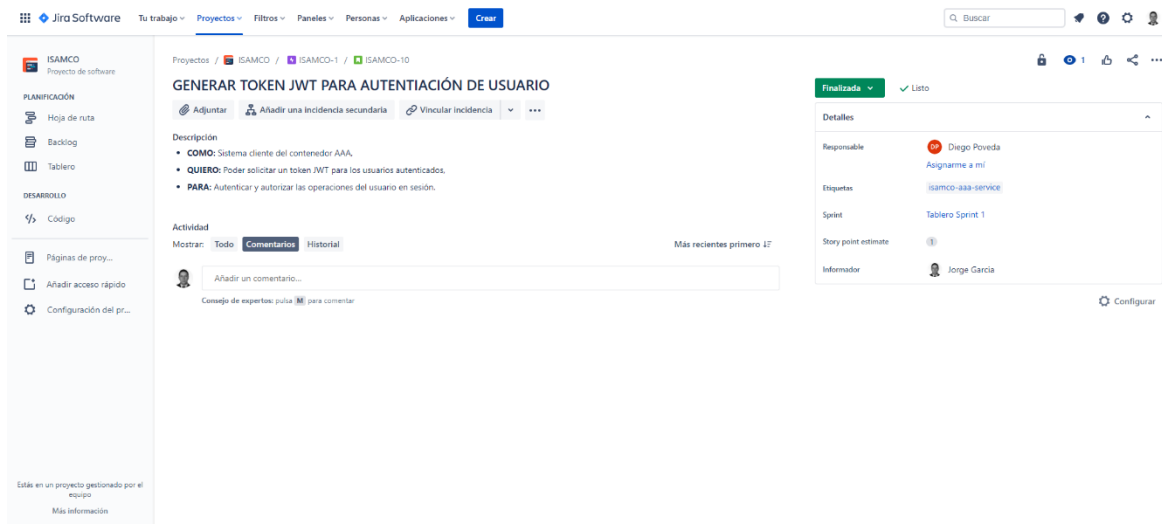


Ilustración 24 Historia de Usuario en Jira (Autores, 2022)

11. Pruebas (Proceso y de Producto)

Para las pruebas unitarias, se utilizó Junit5 con Mockito, para la simulación de las dependencias de cada una de las unidades de prueba. Con el fin de garantizar que los diferentes escenarios de cada uno de los contenedores tuviesen pruebas unitarias, se estableció un porcentaje de cobertura de pruebas del 90%, la cual fue configurada en el

archivo POM de cada proyecto y validada a través del servicio de Sonar. Las pruebas unitarias fueron realizadas excluyendo clases de entidades, value objects, configuración y clase principal de ejecución, solo para concentrarse donde la lógica de negocio se encuentra en cada uno de los proyectos, con el fin de asegurar que las reglas implementadas, corresponden a los requerimientos solicitados para el sistema.

```

131 <build>
132 <plugins>
133 <plugin>
134 <groupId>org.springframework.boot</groupId>
135 <artifactId>spring-boot-maven-plugin</artifactId>
136 <version>${project.parent.version}</version>
137 </plugin>
138 <plugin>
139 <groupId>org.jacoco</groupId>
140 <artifactId>jacoco-maven-plugin</artifactId>
141 <version>0.8.8</version>
142 <configuration>
143 <excludes>
144 <exclude>**/co/logike/ismco/aaa/application/model/*</exclude>
145 <exclude>**/co/logike/ismco/aaa/config/*</exclude>
146 <exclude>**/co/logike/ismco/aaa/domain/exception/*</exclude>
147 <exclude>**/co/logike/ismco/aaa/domain/model/*</exclude>
148 <exclude>**/co/logike/ismco/aaa/infrastructure/entities/*</exclude>
149 </excludes>
150 </configuration>
151 <executions>
152 <execution>
153 <goals>
154 <goal>prepare-agent</goal>
155 </goals>
156 </execution>
157 <execution>
158 <id>report</id>
159 <phase>prepare-package</phase>
160 <goals>
161 <goal>report</goal>
162 </goals>
163 </execution>
164 <execution>
165 <id>jacoco-check</id>
166 <goals>
167 <goal>check</goal>
168 </goals>
169 <configuration>
170 <rules>
171 <rule>
172 <element>BUNDLE</element>
173 <limits>
174 <limit>
175 <counter>LINE</counter>
176 <value>COVEREDRATIO</value>
177 <minimum>0.90</minimum>
178 </limit>
179 </limits>
180 </rule>
181 </rules>
182 </configuration>
183 </executions>
184 </executions>
185 </plugin>
186 </plugins>
187 </build>
188
189 </project>

```

Ilustración 25 Archivo POM con porcentaje de cubrimiento de pruebas unitarias (Autores, 2022)

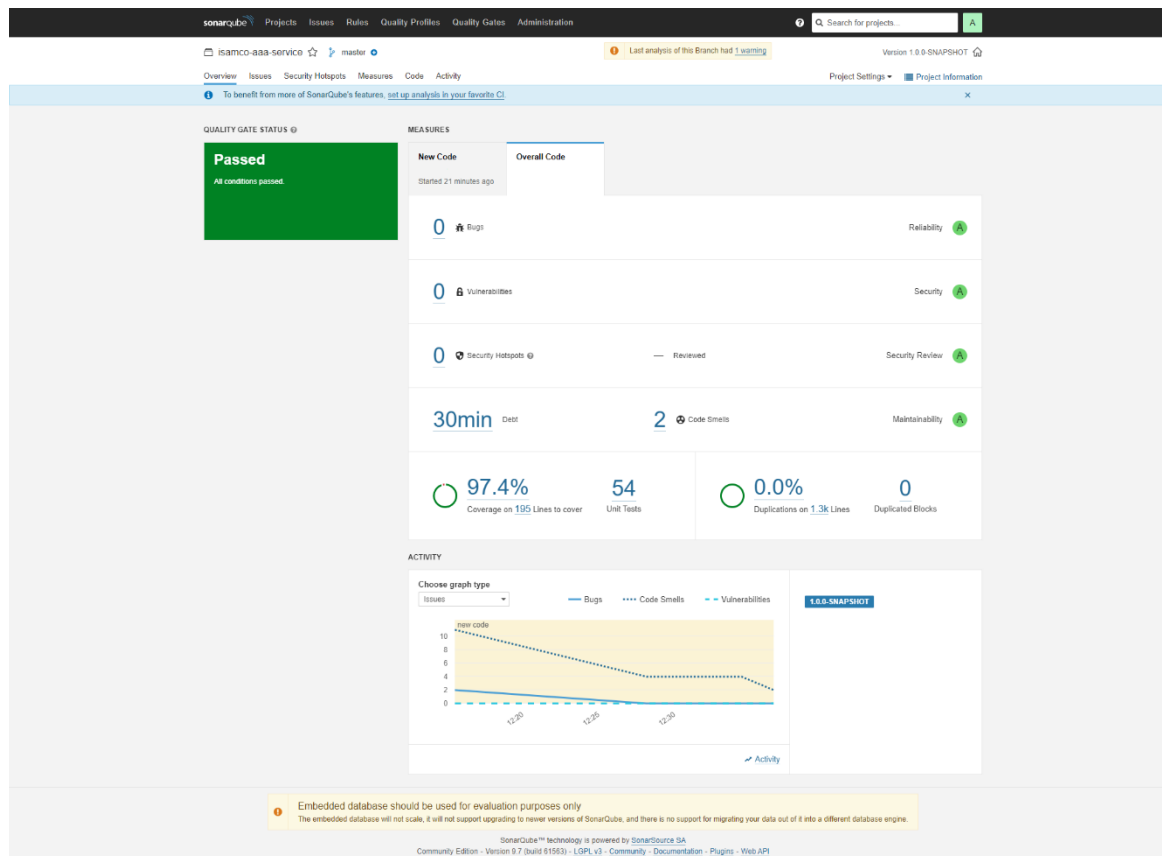


Ilustración 26 Evaluación de cobertura de pruebas unitarias en Sonar (Autores, 2022)

Las pruebas a nivel de producto se realizaron al final de cada sprint de desarrollo, mediante visita presencial o videollamada con la gerencia general y el personal administrativo de ISAMCO, en cumplimiento de las ceremonias de revisión de Scrum. Durante estas reuniones, se tomaba nota de las observaciones y retroalimentación del personal de ISAMCO, a través de un documento en Confluence, para poder ser consultado vía internet por todos los involucrados.

¿Tu equipo necesita acceder a más funciones de Confluence? [Inicia una versión de prueba gratuita del plan Standard.](#)

Confluence [Página principal](#) [Reciente](#) [Espacios](#) [Gente](#) [Aplicaciones](#) [Plantillas](#) [Crear](#) [Compartir](#)

ISAMCO [enlace de JIRA](#) [Compartir](#)

SPRINT REVIEW [ISAMCO-1] Autenticación y Autorización

Creado por Jorge Garcia

Fecha
Mayo 21 de 2022

Jira Epic
[ISAMCO-1: Autenticación y Autorización](#) [TAREAS POR HACER](#)

Participantes
ISAMCO

- Enrique Poveda
- Sandra González.

Equipo de Desarrollo

- Jorge García
- Diego Poveda

Actividades

- Revisión de historias de usuario correspondientes a autenticación y autorización.
- Presentación del módulo de autenticación y autorización.
- Retroalimentación.

Tareas

- Se debe ajustar el estilo de presentación de los módulos.
- Debe completarse el texto del Habeas Data tras el login de usuarios.

[+ Añadir etiqueta](#)

🗨️ Sé la primera persona en añadir una reacción

[Inicio rápido](#)

Ilustración 27 Sprint Review en Confluence (Autores, 2022)

Finalmente, cuando el sistema de información se encontró listo para la prueba piloto acordada, se realizó esta actividad en las instalaciones de ISAMCO, comenzando por una capacitación del uso de la herramienta, tanto al personal administrativo, como a una muestra del personal operativo (conductores) que harían parte del piloto. Este ejercicio fue bastante enriquecedor para todas las partes involucradas en el proyecto, ya que es la primera actividad de la organización en pro de adentrarse en procesos de transformación digital y extender la cultura digital entre sus empleados, en una región y un negocio donde estas operaciones no se habían llevado a cabo con el apoyo de herramientas tecnológicas fabricadas a la medida. La retroalimentación de parte de ISAMCO fue bastante positiva, ya que como en toda prueba piloto, el sistema no estuvo exento de presentar fallos, el sistema de información se acercó

bastante a los requerimientos del negocio y los errores presentados, algunos en los cálculos de conciliación, permitieron definir los ajustes necesarios para que la puesta en marcha del sistema de información de forma oficial sea un éxito.

Con ser de la presentación de algunos errores, ISAMCO concluyó lo siguiente de parte de la capacitación y el ejercicio piloto:

- La conciliación manual de toda la información asociada a las operaciones de transporte de mineral carbón conllevaba un tiempo de 5 días al personal administrativo. Con el estado actual de la herramienta, dado que es necesario realizar verificaciones de los resultados emitidos por el sistema de forma inicial, el personal administrativo considera que puede reducirse a un día, con expectativa de reducirlo a futuro a cuestión de horas.
- El proceso de registro y conciliación manual de información estuvo reportando desde el área operativa y administrativa a la gerencia un promedio de 16 errores por mes, entre lo observado a lo largo del tiempo de construcción del proyecto, tiempo en el cual la empresa monitoreó esta situación. Con lo consolidado a través de la herramienta durante la prueba piloto, y teniendo en cuenta que algunos fallos fueron detectados, ya sea por error del planteamiento del requerimiento original donde faltaron algunas reglas de negocio o por error de programación, ISAMCO consideró que esto se puede tomar una reducción a 4 errores. El objetivo tanto de ISAMCO como del equipo de desarrollo es reducirlos a 0 para el lanzamiento oficial de la herramienta y considerar a partir de allí, cualquier inconsistencia de los cálculos obtenidos como un control de cambios.

- Gracias a las notificaciones recibidas por el personal de ISAMCO que participó en la prueba piloto, el personal administrativo y operativo de la empresa manifestó que sienten que ahora pueden tener más presentes las diversas eventualidades que se pueden presentar durante un viaje y los vencimientos asociados a los diferentes aspectos técnicos de los vehículos, evitando así a futuro cualquier inconveniente o penalidad que se pudiese presentar.

12. Pruebas de Seguridad

La verificación de seguridad web se realizó utilizando la herramienta OWASP ZAP, la cual fue aplicada sobre un entorno en Google App Engine utilizado para pruebas de desarrollo. A través de esta herramienta se ejecutó un escaneo automatizado sobre la URL expuesta del ambiente de desarrollo, la cual lanzó miles de peticiones intentando ataques de diversos tipos, tales como Path Transversal, Inclusión Remota de Archivos, Cross Site Scripting, Fallas por Inyección SQL, entre otras.

Gracias al uso de un framework de interfaz gráfica seguro como Vaadin, la herramienta de OWASP ZAP no encontró ninguna falla de seguridad de alto nivel. La herramienta de análisis solo encontró 7 fallas de riesgo medio y bajo.

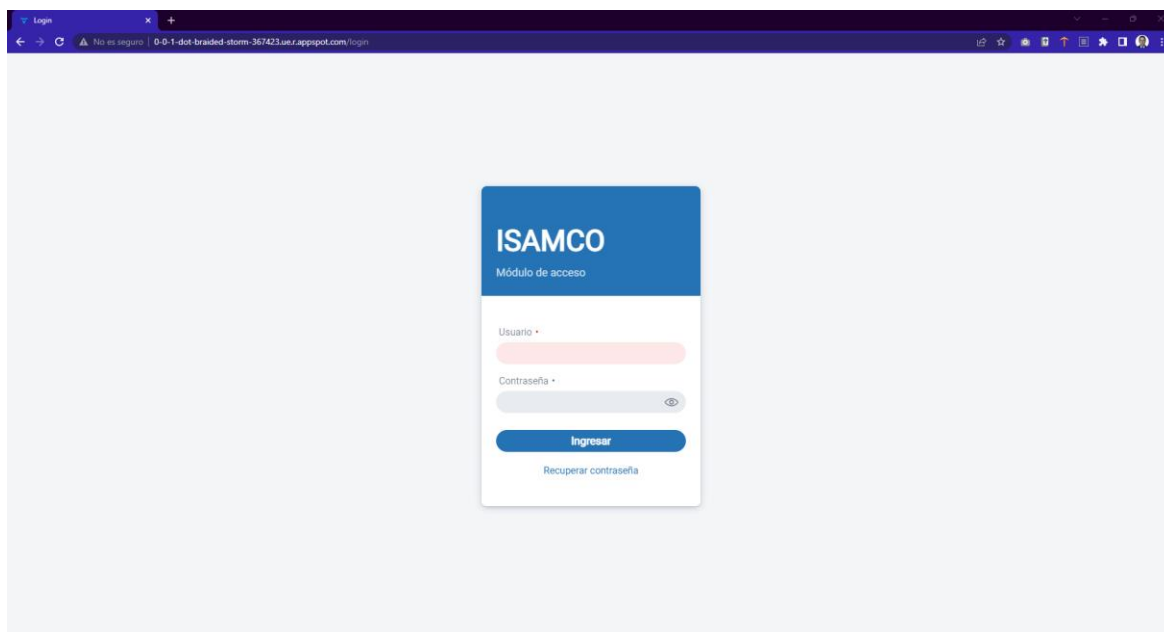


Ilustración 28 Login en ambiente de desarrollo para pruebas de seguridad (Autores, 2022)

About this report

Report parameters

Contexts

No contexts were selected, so all contexts were included by default.

Sites

The following sites were included:

- `http://0-0-1-dot-braided-storm-367423.ue.r.appspot.com`

(If no sites were selected, all sites were included by default.)

An included site must also be within one of the included contexts for its data to be included in the report.

Risk levels

Included: `Alto`, `Medio`, `Bajo`, `Informativo`

Excluded: `None`

Confidence levels

Included: `User Confirmed`, `Alto`, `Medio`, `Bajo`

Excluded: `User Confirmed`, `Alto`, `Medio`, `Bajo`, `Falso positivo`

Ilustración 29 Generalidades del reporte OWASP ZAP (Autores, 2022)

Summaries

Alert counts by risk and confidence

This table shows the number of alerts for each level of risk and confidence included in the report.

(The percentages in brackets represent the count as a percentage of the total number of alerts included in the report, rounded to one decimal place.)

		Confidence				Total
		Confirmed	Alto	Medio	Bajo	
Risk	Alto	0 (0,0 %)	0 (0,0 %)	0 (0,0 %)	0 (0,0 %)	0 (0,0 %)
	Medio	0 (0,0 %)	1 (14,3 %)	0 (0,0 %)	1 (14,3 %)	2 (28,6 %)
	Bajo	0 (0,0 %)	0 (0,0 %)	1 (14,3 %)	1 (14,3 %)	2 (28,6 %)
	Informativo	0 (0,0 %)	0 (0,0 %)	3 (42,9 %)	0 (0,0 %)	3 (42,9 %)
	Total	0 (0,0 %)	1 (14,3 %)	4 (57,1 %)	2 (28,6 %)	7 (100%)

Ilustración 30 Reporte OWASP ZAP (Autores, 2022)

Alert counts by alert type

This table shows the number of alerts of each alert type, together with the alert type's risk level.

(The percentages in brackets represent each count as a percentage, rounded to one decimal place, of the total number of alerts included in this report.)

Alert type	Risk	Count
Content Security Policy (CSP) Header Not Set	Medio	2 (28,6 %)
Hidden File Found	Medio	4 (57,1 %)
Cookie without SameSite Attribute	Bajo	3 (42,9 %)
Divulgación de la marca de hora - Unix	Bajo	16 (228,6 %)
Divulgación de información - Comentarios sospechosos	Informativo	5 (71,4 %)
Modern Web Application	Informativo	2 (28,6 %)
User Agent Fuzzer	Informativo	72 (1.028,6 %)
Total		7

Ilustración 31 Detalles de seguridad encontrados por OWASP ZAP (Autores, 2022)

Tras revisar lo indicado por la herramienta, encontramos que los riesgos de tipo bajo e informativo corresponden a mensajes y comentarios generados tanto por Vaadin como por GCP al no detectar las peticiones de OWASP ZAP como provenientes de un navegador conocido.

Al no encontrar riesgos de alto nivel que comprometan la seguridad de la aplicación y de la información que esta contiene, consideramos que el sistema de información se encontró listo para el piloto productivo.

13. Instalación y Configuración

13.1 Google App Engine

Para instalar los contenedores de aplicación en GCP App Engine, se debe ingresar al panel de este servicio en el administrador cloud de Google.

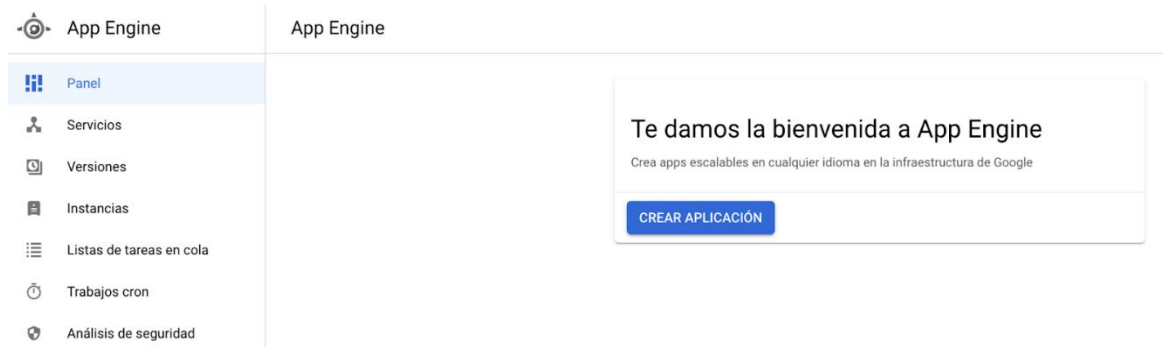


Ilustración 32 Panel App Engine (Autores, 2022)

Luego, se debe seleccionar una región en la cual se ejecutarán los contenedores. En este caso, se seleccionó US-East 1.

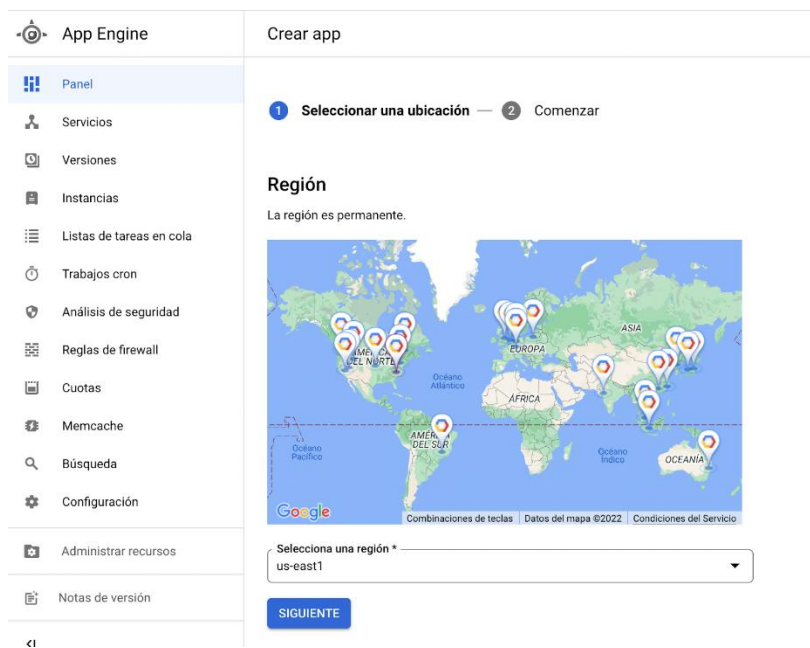


Ilustración 33 Selección de Región para App Engine (Autores, 2022)

Tras este paso, se seleccionan los recursos, correspondientes al lenguaje del entorno de programación, que es Java para este proyecto, y el ambiente Estándar de App Engine.

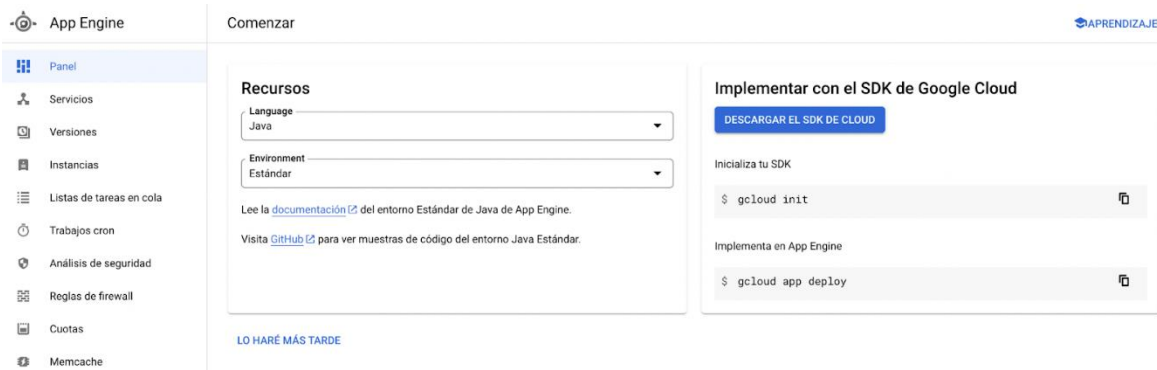


Ilustración 34 Selección de Recursos en App Engine (Autores, 2022)

Cada uno de los contenedores posee un archivo `app.yaml` con la configuración necesaria para el posterior despliegue en App Engine. En este archivo se indica el tipo de instancia a utilizar (F2), la escalabilidad y el perfil de ejecución.

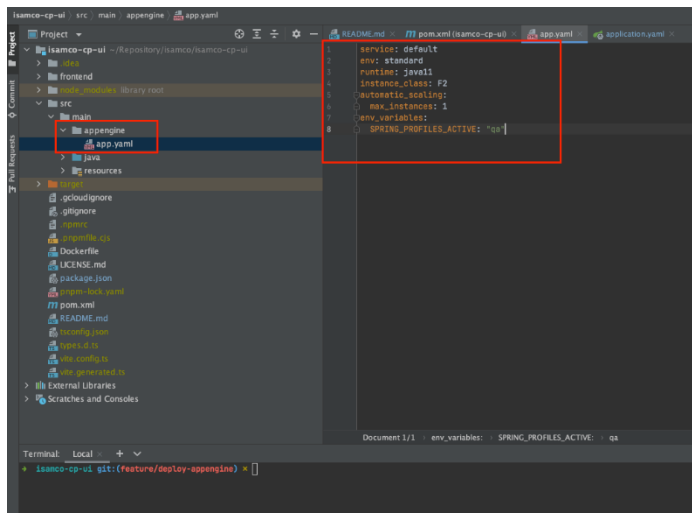


Ilustración 35 Archivo de configuración para App Engine (Autores, 2022)

Con los anteriores pasos ejecutados, se procede a enviar el contenedor al entorno correspondiente en App Engine a través del SDK de Google Cloud, utilizando el comando:

```
gcloud app deploy --appyaml=src/main/appengine/app.yaml --no-promote --
```

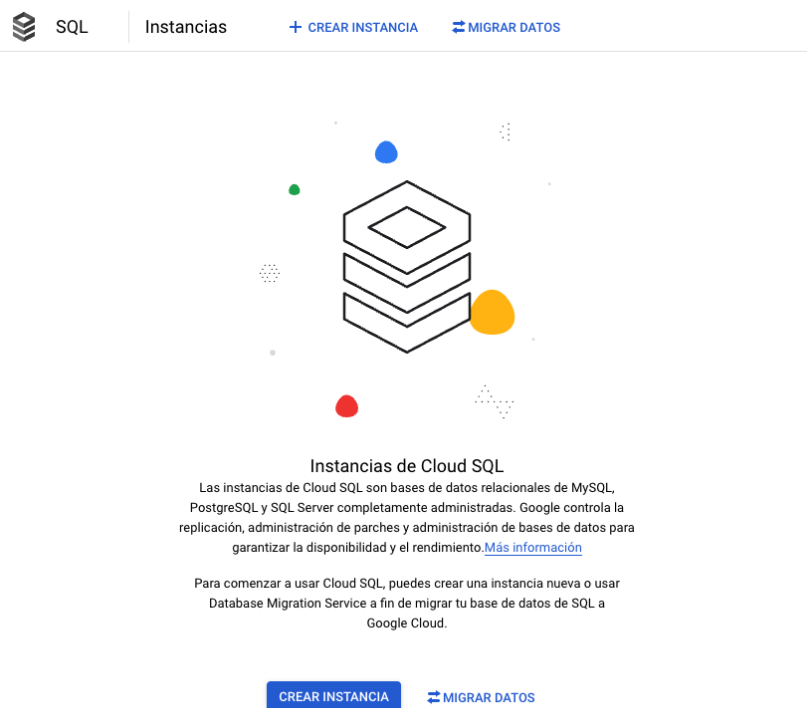



Ilustración 39 Opción de creación de instancia SQL en GCP (Autores, 2022)

En el proceso de creación de instancia, PostgreSQL debe ser seleccionado como el motor de base de datos.



Ilustración 40 Elección de PostgreSQL como motor en Cloud SQL (Autores, 2022)

A continuación, se procede con la configuración de la instancia. Se indica si la base de datos será de producción o desarrollo, la región (que debe ser la misma de App Engine), la disponibilidad y las características de la infraestructura.

← Crea una instancia de PostgreSQL

Elige una configuración para comenzar

Estas opciones de configuración sugeridas completarán previamente el formulario que a continuación se muestra. Puedes personalizarlo más adelante según sea necesario.

Production
Optimized for the most critical workloads. Highly available, performant, and durable.

Development
Performant but not highly available, while reducing cost by provisioning less compute and storage.

▼ DETALLES DE CONFIGURACIÓN

Elige la región y la disponibilidad zonal

Para obtener un mejor rendimiento, mantén tus datos cerca de los servicios que los necesitan. La región es permanente, mientras que la zona se puede cambiar en cualquier momento.

Región
us-east1 (Carolina del Sur)

Disponibilidad zonal

Zona única
Si se suspende el servicio, no se aplica la conmutación por error. No se recomienda esta opción para la producción.

Varias zonas (con alta disponibilidad)
La conmutación por error automática se aplica a otra zona en la región que seleccionaste. Esta opción se recomienda para las instancias de producción. Aumenta el costo.

▼ ESPECIFICAR ZONAS

Personaliza tu instancia

También puedes personalizar las opciones de configuración de instancias más adelante

▼ MOSTRAR OPCIONES DE CONFIGURACIÓN

CREAR INSTANCIA CANCELAR

Resumen

Región	us-east1 (Carolina del Sur)
Versión de la base de datos	PostgreSQL 14
CPU virtuales	4 CPU virtual(es)
Memoria	26 GB
Almacenamiento	100 GB
Capacidad de procesamiento de la red (MB/s)	1,000 de 2,000
Capacidad de procesamiento del disco (MB/s)	Lectura: 48.0 de 240.0 Escritura: 48.0 de 240.0
IOPS	Lectura: 3,000 de 15,000 Escritura: 3,000 de 15,000
Conexiones	IP pública
Copia de seguridad	Automatizada
Disponibilidad	Varias zonas (con alta disponibilidad)
Recuperación de un momento determinado	Habilitada

Ilustración 41 Configuración de Instancia PostgreSQL (Autores, 2022)

Tras la creación de la instancia, se puede verificar el estado de esta en el módulo de administración de Cloud SQL.

SQL Instancias + CREAR INSTANCIA MIGRAR DATOS MOSTRAR PANEL DE INFORMACIÓN

Filtro Ingresar el nombre o el valor de la propiedad

ID de instancia	Tipo	Dirección IP pública	Dirección IP privada	Nombre de la conexión con la instancia	Alta disponibilidad	Ubicación	Almacenamiento usado	Etiquetas	Acciones
isamco-cp-db	PostgreSQL 14	35.196.40.165		branded-storm-367423-us-east1-isamco-cp-db	AGREGAR	us-east1-c	68 MB de 100 GB		

Ilustración 42 Estado de instancia Cloud SQL (Autores, 2022)

En el módulo de administración se puede consultar los detalles de la instancia.

SQL Descripción general EDITAR REPORTAR EXPORTAR REINICIAR DETENER BORRAR CLONAR 1 hora 4 horas 5 días 30 días Personalizar

INSTANCIA PRINCIPAL

- Descripción general
- Estadísticas del sistema
- Estadísticas de consultas
- Conexiones
- Usuarios
- Base de datos
- Copias de seguridad
- Redes
- Operaciones

Uso de CPU

→ Ir a Estadísticas de consultas para obtener información más detallada sobre las consultas y el rendimiento

Conectarse a esta instancia

Dirección IP pública
25.196.40.165

Dirección IP privada
35.196.189.159

Nombre de la conexión
branded-storm-367423-us-east1-isamco-cp-db

¿Necesitas ayuda para conectarte?
Consulta la documentación para conocer los diversos métodos de conectarte a tu instancia. Más información

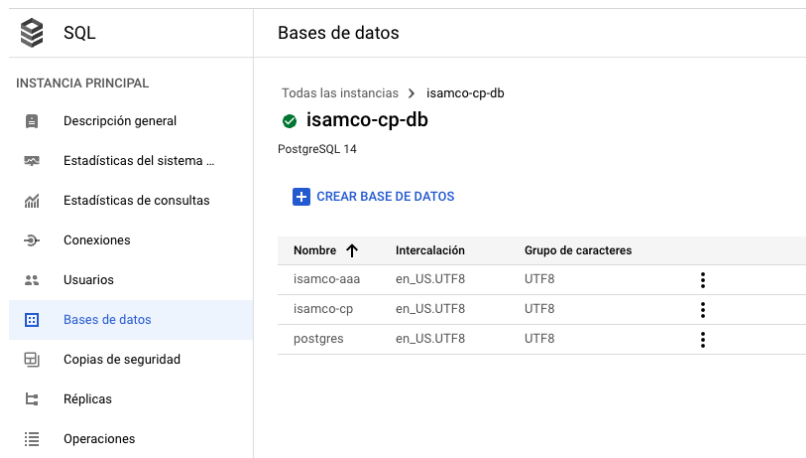
Configuración

CPU virtuales	Memoria	Almacenamiento de SSD
4	61.4 GB	100 GB

- La versión de la base de datos es PostgreSQL 14.4
- El aumento automático de almacenamiento está inhabilitado
- Las copias de seguridad automáticas están inhabilitadas
- La recuperación de un momento determinado está inhabilitada
- La protección contra la eliminación de instancia está inhabilitada
- Usa solo en us-east1-c

Ilustración 43 Detalles de instancia Cloud SQL (Autores, 2022)

Finalmente, se puede acceder a las diferentes instancias creadas para cada uno de los servicios de la aplicación.



The screenshot shows the AWS Cloud SQL console interface. On the left is a navigation menu with options like 'Descripción general', 'Estadísticas del sistema...', 'Estadísticas de consultas', 'Conexiones', 'Usuarios', 'Bases de datos', 'Copias de seguridad', 'Réplicas', and 'Operaciones'. The 'Bases de datos' option is selected. The main panel displays 'Bases de datos' and shows a list of instances under the heading 'Todas las instancias > isamco-cp-db'. A specific instance 'isamco-cp-db' is highlighted, showing it is a PostgreSQL 14 instance. Below this, there is a '+ CREAR BASE DE DATOS' button and a table listing existing instances.

Nombre ↑	Intercalación	Grupo de caracteres	
isamco-aaa	en_US.UTF8	UTF8	⋮
isamco-cp	en_US.UTF8	UTF8	⋮
postgres	en_US.UTF8	UTF8	⋮

Ilustración 44 Instancias ISAMCO en Cloud SQL (Autores, 2022)

14. Conclusiones

- Gracias al uso del sistema de información creado para ISAMCO se determinó que este permite la reducción de las conciliaciones de la información asociada del transporte de carga que originalmente tomaba un tiempo de 5 días al personal administrativo, a 1 día con apoyo de la herramienta.
- Mediante las reglas de negocio establecidas en el sistema de información, se disminuyeron la presentación de errores administrativos durante el proceso de conciliación de 16 en promedio, a 4 con el uso de la herramienta.
- La gerencia general de ISAMCO manifestó que, gracias al uso de los reportes brindados por el sistema de información, consolidando la información de vehículos y viajes, hay una mejor perspectiva para la toma de decisiones a futuro de las operaciones de la compañía.
- Las áreas administrativa y operativa de ISAMCO declaran tener más presentes las diferentes eventualidades asociadas a los vehículos y viajes programados, lo que a futuro permitirá adelantarse a situaciones que implique algún tipo de inconveniente o penalidad.
- El proceso de capacitación realizado para el piloto productivo permitió involucrar a personal de la compañía en procesos digitales de los cuales nunca habían tenido oportunidad de participar, adentrándose en la nueva cultura y transformación digital a la que apunta la organización.
- La arquitectura de microservicios permitió realizar un proceso de construcción del sistema de información requerido de una forma más fácil de distribuir entre los integrantes del equipo de desarrollo, lo que, a su vez,

permitió centrarse en un problema a solucionar al tiempo, y utilizar tecnologías de última generación, habilitando así que la organización pueda tener su sistema de información en la nube, ahorrando la implantación de costosa infraestructura en sitio.

15. Referencias Bibliográficas

- Abellán, P. (24 de 5 de 2019). *Introducción a Vaadin Flow*. Obtenido de Tribalyte Technologies: <https://tech.tribalyte.eu/blog-introduccion-vaadin-flow>
- Agencia Nacional de Minería. (2017). *Cundinamarca, ejemplo de minería bien hecha*. Obtenido de https://www.anm.gov.co/?q=cundinamarca_ejemplo_de_mineria_bien_hecha_principal
- Agencia Nacional de Minería. (2022). *Carbón*. Obtenido de <https://www.anm.gov.co/sites/default/files/DocumentosAnm/carbon.pdf>
- Amazon Web Services, Inc. (2022). *Amazon API Gateway*. Obtenido de AWS: <https://aws.amazon.com/es/api-gateway/>
- Amazon Web Services, Inc. (2022). *¿Qué es Docker?* . Obtenido de Contenedores de Docker: <https://aws.amazon.com/es/docker/>
- Amazon Web Services, Inc. (2022). *¿Qué son los microservicios?* Obtenido de AWS: <https://aws.amazon.com/es/microservices/>
- Amazon Web Services, Inc. (2022). *Amazon Aurora*. Obtenido de AWS: <https://aws.amazon.com/es/rds/aurora/>
- Amazon Web Services, Inc. (2022). *Amazon Elastic Container Service (Amazon ECS)*. Obtenido de AWS: <https://aws.amazon.com/es/ecs/>
- Amazon Web Services, Inc. (2022). *Amazon S3*. Obtenido de AWS: <https://aws.amazon.com/es/s3/>
- ARTECO. (2020). *Por qué DEBES usar JAVA para programar tus proyectos*. Obtenido de Artecó: <https://www.arteco-consulting.com/por-que-debes-usar-java/>
- Auren. (2021). *Agenda de Carbón y Transformación Minera en Cundinamarca*. Obtenido de <https://auren.com/co/blog/agenda-de-carbon-y-transformacion-minera-en-cundinamarca/>
- BEE INTERACTIVE S.R.O. (2022). *EASY CARGO*. Obtenido de <https://www.easycargo3d.com>
- Carranza, A. (2021). *Conoce qué es Java y ¡diseña aplicaciones móviles de ensueño!* Obtenido de Crehana: <https://www.crehana.com/co/blog/desarrollo-web/que-es-java/>

- CLOUD FLEET. (2022). *CLOUD FLEET*. Obtenido de <https://cloudfleet.com>
- COLOMBIA SOFTWARE. (2022). *SILOGCARGO*. Obtenido de <https://www.colombiasoftware.net/silogcargo/>
- CONTROLT. (2022). *Torre de control: Innovación para el transporte*. Obtenido de <https://controlt.com.co/torre-de-control-innovacion-para-el-transporte/>
- DIGITAL DISSENY. (2022). *DIDTRANS*. Obtenido de <https://www.didtrans.com>
- Dorantes, C. (2022). *PostgreSQL: qué es, cómo funciona y cuáles son sus ventajas*. Obtenido de Platzi: <https://platzi.com/blog/que-es-postgresql/>
- Gobernación de Cundinamarca. (2020). *Transformación Digital para Todos*. Obtenido de [Cundinamarca.gov.co](https://www.cundinamarca.gov.co): <https://www.cundinamarca.gov.co/Home/prensa/asnoticiasprensa/transformacion+digital+para+todos>
- Gonçalves, M. J. (2021). *¿Qué es Angular y para qué sirve?* Obtenido de Blog de Hiberus Tecnología: <https://www.hiberus.com/crecemos-contigo/que-es-angular-y-para-que-sirve/>
- Google Cloud. (2022). *Base de datos administrada de Cloud SQL para PostgreSQL*. Obtenido de Google Cloud: <https://cloud.google.com/sql/postgresql>
- Google Cloud. (2022). *Cloud SQL para PostgreSQL, MySQL y SQL Server*. Obtenido de Google Cloud: <https://cloud.google.com/sql>
- Google Cloud. (2022). *Cloud Storage*. Obtenido de Google Cloud: <https://cloud.google.com/storage?hl=es>
- Google Cloud. (2022). *Entorno estándar de App Engine*. Obtenido de Google Cloud: <https://cloud.google.com/appengine/docs/standard>
- Google Cloud. (2022). *Plataforma de aplicación de App Engine*. Obtenido de Google Cloud: <https://cloud.google.com/appengine>
- GRUPO NW. (2022). *SITCA*. Obtenido de SITCA: <https://www.sitca.co>
- Hernández, U. (2012). *Qué es TypeScript*. Obtenido de CódigoFacilito: <https://codigofacilito.com/articulos/typescript>
- Huambachano, J. F. (2017). *¿Qué es Scrum?* Obtenido de Scrum.org: <https://www.scrum.org/resources/blog/que-es-scrum>

- LIS. (2022). *LISTMS*. Obtenido de <https://lis.com.mx/sistema-administracion-flotillas-transporte>
- Muradas, Y. (2018). *Qué es Spring Framework y por qué usarlo*. Obtenido de OpenWebinars.net: <https://openwebinars.net/blog/conoce-que-es-spring-framework-y-por-que-usarlo/>
- Oracle. (2022). *¿Qué es la tecnología Java y para qué la necesito?* Obtenido de https://www.java.com/es/download/help/whatis_java.html
- PROCOLOMBIA. (2004). *Guía Ambiental Transporte de Carbón*. Obtenido de https://www1.upme.gov.co/siame/Guiasambientales/Transporte_carbon.pdf
- PULPO. (2022). *PULPO*. Obtenido de <https://www.getpulpo.com>
- Quality Devs. (2019). *¿Qué es Angular y para qué sirve?* Obtenido de Quality Devs: <https://www.qualitydevs.com/2019/09/16/que-es-angular-y-para-que-sirve/>
- Scrum.org. (2020). *What is Scrum?* Obtenido de Scrum.org: <https://www.scrum.org/resources/what-is-scrum>
- SOFTAR SRL. (2022). *CRISTAL*. Obtenido de <https://www.softwarecristal.com>
- Suarez, L. C. (2021). *Cundinamarca fortalece los procesos de transformación digital*. Obtenido de OTV Television: <https://otvtelevision.com/cundinamarca-fortalece-los-procesos-de-transformacion-digital/>
- SYSCOM. (2022). *TRANSCARGA*. Obtenido de <https://www.syscom.com.co/transcarga>
- TechTarget. (2020). *Scrum*. Obtenido de ComputerWeekly: <https://www.computerweekly.com/es/definicion/Scrum>
- The PostgreSQL Global Development Group. (2022). *1. What Is PostgreSQL?* Obtenido de PostgreSQL Documentation: <https://www.postgresql.org/docs/14/intro-what-is.html>
- The PostgreSQL Global Development Group. (2022). *PostgreSQL*. Obtenido de <https://www.postgresql.org/>
- UPME. (2005). *La Cadena del Carbón*. Obtenido de http://www.upme.gov.co/Docs/Cadena_carbon.pdf
- Vaadin Ltd. (2022). *Vaadin - An open platform for building web apps in Java*. Obtenido de Vaadin.com: <https://vaadin.com/>

Vaadin Ltd. (2022). *Vaadin Flow Quick Start*. Obtenido de Vaadin.com:
<https://vaadin.com/docs/latest/guide/quick-start>

VMware, Inc. (2021). *Spring Cloud Gateway*. Obtenido de Spring.io:
<https://spring.io/projects/spring-cloud-gateway>

Zettler, K. (2022). *¿Qué es la computación en la nube? Visión general de la nube*.
Obtenido de Atlassian: <https://www.atlassian.com/es/continuous-delivery/principles/cloud-computing>