



Universidad Nacional Mayor de San Marcos

Universidad del Perú. Decana de América

Facultad de Ingeniería de Sistemas e Informática

Escuela Profesional de Ingeniería de Sistemas

Implementación de una solución tecnológica bajo la metodología Scrum para la autogestión, monitoreo y control de los consumos de las unidades de los clientes flota de una empresa comercializadora de combustible

TRABAJO DE SUFICIENCIA PROFESIONAL

Para optar el Título Profesional de Ingeniero de Sistemas

AUTOR

Fred Anthony MORALES CABEZAS

ASESOR

Mg. César Augusto ALCÁNTARA LOAYZA

Lima, Perú

2022



Reconocimiento - No Comercial - Compartir Igual - Sin restricciones adicionales

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

Usted puede distribuir, remezclar, retocar, y crear a partir del documento original de modo no comercial, siempre y cuando se dé crédito al autor del documento y se licencien las nuevas creaciones bajo las mismas condiciones. No se permite aplicar términos legales o medidas tecnológicas que restrinjan legalmente a otros a hacer cualquier cosa que permita esta licencia.

Referencia bibliográfica

Morales, F. (2022). *Implementación de una solución tecnológica bajo la metodología Scrum para la autogestión, monitoreo y control de los consumos de las unidades de los clientes flota de una empresa comercializadora de combustible*. [Trabajo de suficiencia profesional de pregrado, Universidad Nacional Mayor de San Marcos, Facultad de Ingeniería de Sistemas e Informática, Escuela Profesional de Ingeniería de Sistemas]. Repositorio institucional Cybertesis UNMSM.

Metadatos complementarios

Datos de autor	
Nombres y apellidos	FRED ANTHONY MORALES CABEZAS
Tipo de documento de identidad	DNI
Número de documento de identidad	47006639
URL de ORCID	https://orcid.org/0000-0001-6279-8490
Datos de asesor	
Nombres y apellidos	César Augusto Alcántara Loayza
Tipo de documento de identidad	DNI
Número de documento de identidad	09132297
URL de ORCID	https://orcid.org/0000-0003-3435-4555
Datos del jurado	
Presidente del jurado	
Nombres y apellidos	Augusto Parcemon Cortez Vásquez
Tipo de documento	DNI
Número de documento de identidad	08634618
Miembro del jurado 1	
Nombres y apellidos	Norberto Antonio Osorio Beltrán
Tipo de documento	DNI
Número de documento de identidad	08799230
Datos de investigación	
Línea de investigación	No aplica
Grupo de investigación	No aplica
Agencia de financiamiento	Sin financiamiento

Ubicación geográfica de la investigación	País: Perú Departamento: Lima Provincia: Lima Distrito: Cercado de Lima Jr. Carlos Amezaga No. 375 Universidad Nacional Mayor de San Marcos Latitud: -12.0564232 Longitud: -77.0843327
Año o rango de años en que se realizó la investigación	2022
URL de disciplinas OCDE	2.02.04 -- Ingeniería de sistemas y comunicaciones https://purl.org/pe-repo/ocde/ford#2.02.04



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
Escuela Profesional de Ingeniería de Sistemas

Acta Virtual de Sustentación
del Trabajo de Suficiencia Profesional

Siendo las 19:00 horas del día 17 de agosto del año 2022, se reunieron virtualmente los docentes designados como Miembros de Jurado del Trabajo de Suficiencia Profesional, presidido por el Mg. Cortez Vásquez Augusto Parcemon (Presidente), Ing. Osorio Beltrán Norberto Antonio (Miembro) y el Mg. Alcántara Loayza César Augusto (Miembro Asesor), usando la plataforma Meet (<https://meet.google.com/zhf-yrtc-sfc>), para la sustentación virtual del Trabajo de Suficiencia Profesional intitulado: **“IMPLEMENTACIÓN DE UNA SOLUCIÓN TECNOLÓGICA BAJO LA METODOLOGÍA SCRUM PARA LA AUTOGESTIÓN, MONITOREO Y CONTROL DE LOS CONSUMOS DE LAS UNIDADES DE LOS CLIENTES FLOTA DE UNA EMPRESA COMERCIALIZADORA DE COMBUSTIBLE”**, por el Bachiller **Morales Cabezas Fred Anthony**; para obtener el Título Profesional de Ingeniero de Sistemas.

Acto seguido de la exposición del Trabajo de Suficiencia Profesional, el Presidente invitó al Bachiller a dar las respuestas a las preguntas establecidas por los miembros del Jurado.

El Bachiller en el curso de sus intervenciones demostró pleno dominio del tema, al responder con acierto y fluidez a las observaciones y preguntas formuladas por los señores miembros del Jurado.

Finalmente habiéndose efectuado la calificación correspondiente por los miembros del Jurado, el Bachiller obtuvo la nota de 18 (**DIECIOCHO**).

A continuación el Presidente de Jurados el Mg. Cortez Vásquez Augusto Parcemon, declara al Bachiller **Ingeniero de Sistemas**.

Siendo las 20:00 horas, se levantó la sesión.

Presidente

Mg. Cortez Vásquez Augusto Parcemon

Miembro

Ing. Osorio Beltrán Norberto Antonio

Miembro Asesor

Mg. Alcántara Loayza César Augusto



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
Universidad del Perú, DECANA DE AMÉRICA
FACULTAD DE INGENIERIA DE SISTEMAS E INFORMATICA
Escuela Profesional de Ingeniería de Sistemas

INFORME DE EVALUACIÓN DE ORIGINALIDAD

1. Facultad de Ingeniería de Sistemas e Informática
2. Escuela Profesional de Ingeniería de Sistemas
3. Autoridad académica que emite el informe de originalidad
Directora(e) de la EPIS
4. Apellidos y Nombres de la autoridad académica
Dra. Luzmila Elisa Pró Concepción
5. Operador del programa informático de similitudes
Dra. Luzmila Elisa Pró Concepción
6. Documento evaluado
Título de pregrado: "Implementación de una Solución Tecnológica bajo La Metodología Scrum para la Autogestión, Monitoreo y Control de los Consumos de las Unidades de los Clientes Flota de una Empresa Comercializadora de Combustible"
7. Autor del documento
Bach. Morales Cabezas, Fred Anthony
8. Fecha de recepción del documento **10/09/2022**
9. Fecha de aplicación del programa informático de similitudes **10/09/2022**
10. Software utilizado
 - Turnitin
11. Configuración del programa detector de similitudes
 - Excluye textos entrecomillados
 - Excluye bibliografía
 - Excluye cadenas menores a 40 palabras
12. Porcentaje de similitudes según programa detector de similitudes **9 (nueve) %**
13. Fuentes originales de las similitudes encontradas
Se adjunta en el anexo 1
14. Observaciones
15. Calificación de originalidad
 - Documento cumple criterios de originalidad, sin observaciones ☒
 - Documento cumple criterios de originalidad, con observaciones ☐
 - Documento no cumple criterios de originalidad ☐
16. Fecha de informe **18/09/2022**



Firmado digitalmente por PRO
CONCEPCION Luzmila Elisa FAU
20148092202 soft
Motivo: Soy el autor del documento
Fecha: 14.11.2022 23:02:26 -05:00

Firma de evaluador
Dra. Luzmila E. Pró Concepción
Directora (e) de la EPIS



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
Universidad del Perú, DECANA DE AMÉRICA
FACULTAD DE INGENIERIA DE SISTEMAS E INFORMATICA
Escuela Profesional de Ingeniería de Sistemas

ANEXO 1

Fuentes originales de las similitudes encontradas

hdl.handle.net	4%
docs.microsoft.com	1%
josecuellar.net	1%
Submitted to Universidad Cesar Vallejo	1%
Submitted to Corporación Universitaria Remington	<1%
www.ibm.com	<1%
Submitted to Universidad Camilo José Cela	<1%
idoc.pub	<1%
cybertesis.unmsm.edu.pe	<1%
Submitted to Universidad de Granada	<1%
openwebinars.net	<1%



UNMSM

Firmado digitalmente por PRO
CONCEPCIÓN Luzmila Elisa FAU
20148092282 soft
Motivo: Soy el autor del documento
Fecha: 14.11.2022 23:02:06 -05:00

Firma de evaluador

Dra. Luzmila E. Pró Concepción

Director (e) de la EPIS

DEDICATORIA

A mi familia por todo el apoyo brindado,
sobre todo en esta etapa para alcanzar un
logro tan importante para mi desarrollo
profesional.

AGRADECIMIENTOS

Agradezco a Dios por siempre estar a mi lado, brindándome la fortaleza para seguir adelante.

A mi asesor César Alcántara por sus apreciables consejos brindados en el desarrollo del Trabajo de Suficiencia Profesional.

A todas aquellas personas con las que tuve la oportunidad de trabajar y saber apreciar sus experiencias y consejos.

UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
FACULTAD DE INGENIERIA DE SISTEMAS E INFORMATICA
ESCUELA PROFESIONAL DE INGENIERIA DE SISTEMAS

**IMPLEMENTACIÓN DE UNA SOLUCIÓN TECNOLÓGICA BAJO LA
METODOLOGÍA SCRUM PARA LA AUTOGESTIÓN, MONITOREO Y
CONTROL DE LOS CONSUMOS DE LAS UNIDADES DE LOS CLIENTES
FLOTA DE UNA EMPRESA COMERCIALIZADORA DE COMBUSTIBLE**

Autor: Morales Cabezas, Fred Anthony
Asesor: Alcántara Loayza, César Augusto
Título: Trabajo de Suficiencia Profesional para optar el Título Profesional
de Ingeniero de Sistemas
Fecha: Julio 2022

RESUMEN

El presente trabajo de suficiencia profesional expone la implementación de una solución tecnológica para la gestión, monitoreo y control de los consumos de combustible de las unidades vehiculares que pertenecen a clientes corporativos de una Empresa Comercializadora de Combustible (ECC), que utiliza una red de estaciones de servicio para abastecer de combustible a estas unidades vehiculares. Para ello, se desarrolló una plataforma web cuya arquitectura está conformada por dos capas, backend (API REST) y frontend (Angular), donde se reutilizaron algunos componentes de los sistemas que actualmente maneja la ECC. El proyecto se desarrolló bajo la metodología scrum adaptada. Esta solución permitió que la carga administrativa (gestión de vehículos) que realizaban los asesores comerciales de la ECC pase a manos de la gestión de los propios clientes. Con ello, se logró reducir en 3 horas diarias la carga administrativa de cada asesor comercial, así como el aumento de confianza con los clientes al permitirles tener el control de la gestión de su flota vehicular.

Palabras clave: Gestión de Flota, Venta de Combustible, Scrum, API REST, Empresa Comercializadora de Combustible.

NATIONAL MAJOR UNIVERSITY OF SAN MARCOS
FACULTY OF SYSTEMS ENGINEERING AND INFORMATICS
PROFESSIONAL SCHOOL OF SYSTEMS ENGINEERING

**IMPLEMENTATION OF A TECHNOLOGICAL SOLUTION UNDER THE
SCRUM FRAMEWORK FOR THE SELF-MANAGEMENT, MONITORING
AND CONTROL OF THE CONSUMPTION OF THE UNITS OF THE FLEET
CUSTOMERS OF A FUEL TRADING COMPANY**

Author: Morales Cabezas, Fred Anthony
Advisor: Alcántara Loayza, César Augusto
Title: Work of Professional Sufficiency to opt for the Professional Title of
Systems Engineer
Date: July 2022

ABSTRACT

The present work of professional sufficiency exposes the implementation of a technological solution for the management, monitoring and control of fuel consumption of vehicle units belonging to corporate clients of a Empresa Comercializadora de Combustible (ECC), which uses a network of service stations to supply fuel to these vehicle units. For this purpose, a web platform was developed whose architecture consists of two layers, backend (API REST) and frontend (Angular), where some components of the systems currently managed by the ECC were reused. The project was developed under the adapted scrum methodology. This solution allowed the administrative burden (vehicle management) performed by the commercial advisors of the ECC to be transferred to the management of the clients themselves. With this, the administrative burden of each commercial advisor was reduced by 3 hours per day, as well as the increase of trust with the clients by allowing them to have control over the management of their vehicle fleet.

Keywords: Fleet Management, Fuel Sales, Scrum, API REST, Fuel Trading Company.

TABLA DE CONTENIDO

RESUMEN.....	v
INTRODUCCIÓN.....	1
CAPÍTULO I - TRAYECTORIA PROFESIONAL	2
CAPÍTULO II - CONTEXTO EN EL QUE SE DESARROLLÓ LA EXPERIENCIA	8
2.1. EMPRESA - ACTIVIDAD QUE REALIZA	8
2.2. VISIÓN	8
2.3. MISIÓN	8
2.4. ORGANIZACIÓN DE LA EMPRESA.....	8
2.5. ÁREA, CARGO Y FUNCIONES DESEMPEÑADAS	10
2.6. EXPERIENCIA PROFESIONAL REALIZADA EN LA ORGANIZACIÓN	12
CAPÍTULO III - ACTIVIDADES DESARROLLADAS	13
3.1. SITUACIÓN PROBLEMÁTICA.....	13
3.1.1. DEFINICIÓN DEL PROBLEMA.....	14
3.2. SOLUCIÓN	14
3.2.1. OBJETIVOS	15
3.2.2. ALCANCE.....	16
3.2.3. ETAPAS Y METODOLOGÍA	16
3.2.4. FUNDAMENTOS UTILIZADOS	17
3.2.4.1. CONTROL DE COMBUSTIBLE EN LA GESTIÓN DE FLOTAS.....	17
3.2.4.2. SCRUM	18
3.2.4.3. SCRUM POKER CARDS (AGILE)	22
3.2.4.4. API REST	22
3.2.4.5. DOMAIN DRIVEN DESIGN (DDD).....	24

3.2.4.6.	NET FRAMEWORK	25
3.2.4.7.	VISUAL STUDIO	25
3.2.4.8.	SQL SERVER	26
3.2.4.9.	SWAGGER USER INTERFACE (SWAGGER UI).....	27
3.2.4.10.	GIGYA CONNECTOR	28
3.2.4.11.	ANGULAR	28
3.2.5.	<i>IMPLEMENTACIÓN DE LAS ÁREAS, PROCESOS, SISTEMAS Y BUENAS PRÁCTICAS</i>	29
3.3.	EVALUACIÓN	83
3.3.1.	<i>BENEFICIO PARA LOS CLIENTES DE LA ECC</i>	84
3.3.2.	<i>BENEFICIO PARA LA ECC</i>	87
CAPÍTULO IV. REFLEXIÓN CRÍTICA DE LA EXPERIENCIA		89
CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES		90
5.1.	CONCLUSIONES	90
5.2.	RECOMENDACIONES.....	91
5.3.	FUENTES DE INFORMACIÓN	92
5.4.	GLOSARIO	94
ANEXOS		95
<i>Anexo 1. Vista de los módulos de la Plataforma</i>		<i>95</i>
<i>Anexo 2. Laboratorio de pruebas donde se emula el funcionamiento del Punto de Venta.....</i>		<i>96</i>
<i>Anexo 3. Conformidad del asistente de calidad respecto a la funcionalidad Solicitar Tarjeta Física de la Plataforma</i>		<i>97</i>
<i>Anexo 4. Documento Parte del release 5 de la Plataforma.....</i>		<i>98</i>
<i>Anexo 5. Documento Parte del release 5 del BackOffice</i>		<i>99</i>
<i>Anexo 6. Documento de creación de objetos en el BackOffice</i>		<i>100</i>
<i>Anexo 7. Documento técnico ágil de release 5</i>		<i>101</i>
<i>Anexo 8. Evidencia ejecución de los scripts de rollback del release 5.....</i>		<i>102</i>

<i>Anexo 9. Rollback del release 5 de la Plataforma.....</i>	<i>103</i>
<i>Anexo 10. Documento TAG del release 5</i>	<i>104</i>
<i>Anexo 11. Conformidad del equipo de soporte respecto a la capacitación de las funcionalidades de la Solución desplegadas en el del release 5.....</i>	<i>105</i>

ÍNDICE DE TABLAS

Tabla 1. Experiencia profesional	2
Tabla 2. Educación superior	5
Tabla 3. Cursos y capacitaciones.....	6
Tabla 4. Fases y procesos del proyecto	16
Tabla 5. Procesos Scrum	21
Tabla 6. Equipo inicial de la célula 1	41
Tabla 7. Equipo de la célula 1 luego de la reconfiguración	42
Tabla 8. Listado de releases desplegados en el proyecto	45
Tabla 9. Tiempo ocupado aproximado por el asesor comercial en la atención de los clientes que tenían a cargo.....	87

ÍNDICE DE FIGURAS

Figura 1. Organigrama de la ECM en las oficinas de Perú	9
Figura 2. Organigrama de la ECC	10
Figura 3. Roles scrum.....	19
Figura 4. Scrum Poker Cards, una herramienta para puntuar historias de usuario.....	22
Figura 5. API REST para proporcionar datos del servidor al usuario de una aplicación	23
Figura 6. Ejemplo de una arquitectura basado en DDD	24
Figura 7. Imagen del IDE de Visual Studio.....	26
Figura 8. Ventana de consultas de SQL Server	27
Figura 9. Interfaz de usuario de Swagger	28
Figura 10. Elementos de un componente en Angular.....	29
Figura 11. Organización de los equipos del proyecto modernización de EES.....	32
Figura 12. Posibles planteamientos respecto al inicio de la construcción de la solución	34
Figura 13. Primera versión del backlog del producto.....	35
Figura 14. Flujo de la venta de combustible para el Servicio para Flotas antes de la implantación del proyecto	39
Figura 15. Cambios de alcance realizados en el proyecto	43
Figura 16. Especificaciones de la estructura del sprint en el proyecto.....	44
Figura 17. Cronograma global de la implementación del proyecto hasta el primer cambio de alcance.....	46
Figura 18. Cronograma global de la implementación del proyecto desde el primer cambio de alcance hasta el segundo cambio de alcance.....	46
Figura 19. Cronograma global de la implementación del proyecto desde el segundo cambio de alcance hasta la finalización del proyecto.....	47
Figura 20. Historias de usuario detalladas junto a los puntos de historia.....	49
Figura 21. Sprint backlog del sprint 21	50
Figura 22. Vista preliminar del tablero con las tareas del equipo	51
Figura 23. Backlog del producto con las historias de usuario refinadas	52

Figura 24. Esquemas de base de datos del proyecto Modernización de EES	53
Figura 25. Arquitectura general de la solución.....	55
Figura 26. Arquitectura del servicio API REST de la Plataforma y el Servicio de Comunicación.....	56
Figura 27. Swagger como documentación del API REST de la Plataforma	58
Figura 28. Estructura del proyecto de la Plataforma en Visual Studio.....	59
Figura 29. Vista inicial de la Plataforma para el perfil Administrador	60
Figura 30. Vista inicial de la Plataforma para el perfil Asesor donde visita la información de un cliente prepago	60
Figura 31. Vista inicial de la Plataforma para el perfil Asesor donde visita la información de un cliente crédito	61
Figura 32. Opción Reglas Comerciales en la Plataforma.....	61
Figura 33. Pantalla inicio de la opción Reglas Comerciales	62
Figura 34. Pantalla de creación de una Regla Comercial	62
Figura 35. Reglas de Frecuencia.....	63
Figura 36. Reglas de Producto.....	63
Figura 37. Regla de Tope por Consumo Diario.....	64
Figura 38. Regla por Estación de Servicio	64
Figura 39. Captura de la placa del vehículo y lectura de QR en el Punto de Venta.....	69
Figura 40. Validación de la regla por frecuencia en el Punto de Venta	70
Figura 41. Validación de la Regla de Producto en el Punto de Venta.....	71
Figura 42. Notificación que llega a la Plataforma cuando se trata de consumir un tipo de producto no autorizado	72
Figura 43. Validación de la Regla de Frecuencia por Días en el Punto de Venta.....	73
Figura 44. Validación de la Regla de Frecuencia por horario en el Punto de Venta.....	74
Figura 45. Evaluación de la implementación de las historias de usuario del sprint por parte del product owner y los stakeholders	75
Figura 46. Listado de actividades del pase a producción	77
Figura 47. Flujo de la venta de combustible para el Servicio para Flotas después de la implantación de la solución	80

Figura 48. Resultado de la encuesta respecto a la pregunta 1	85
Figura 49. Resultado de la encuesta respecto a la pregunta 2	85
Figura 50. Resultado de la encuesta respecto a la pregunta 3	86
Figura 51. Resultado de la encuesta respecto a la pregunta 4	86

INTRODUCCIÓN

Los asesores comerciales de una empresa que por razones de confidencialidad denominaremos Empresa Comercializadora de Combustible (en adelante ECC) envían a sus clientes corporativos un reporte acerca del consumo diario de combustible de sus vehículos, al final del día. Como resultado de la revisión del reporte, los clientes veían comportamientos sospechosos de consumo de combustible, por lo que tenían que realizar otra gestión para aplicar algún mecanismo de control.

Ante esta problemática, la ECC vio una oportunidad para mejorar el servicio concretándose en la implementación de una solución tecnológica que permitiera a sus clientes gestionar, monitorear y controlar el consumo de combustible de su flota; y, a su vez, disminuir la carga administrativa de los asesores comerciales (envío de reportes). El presente trabajo se enfoca en la capa backend de uno de los artefactos de solución: la Plataforma.

En el Capítulo I se detalla la experiencia profesional del autor, su formación académica y los conocimientos adquiridos en los últimos años.

En el Capítulo II se detalla la información de la empresa consultora donde laboró el autor, misión, visión, estructura organizacional, función del cargo que ocupó y una descripción de la experiencia profesional adquirida.

En el Capítulo III se analiza la situación de los problemas que impulsaron el desarrollo del proyecto, el establecimiento de objetivos, la estructura de la metodología utilizada, participación del autor en el desarrollo del proyecto, y los beneficios que trajo la implantación de la solución para los clientes y la ECC.

En el Capítulo IV se realiza una reflexión crítica sobre la experiencia del autor en el desarrollo del proyecto, describiendo la situación presentada, los problemas encontrados y una comparación con otros trabajos similares.

En el Capítulo V se exponen las conclusiones, recomendaciones, fuentes de información y glosario, así como los términos mencionados en el informe.

CAPÍTULO I - TRAYECTORIA PROFESIONAL

El autor del presente Trabajo de Suficiencia Profesional es bachiller de la carrera de Ingeniería de Sistemas e Informática. Cuenta con más de 3 años de experiencia en análisis y desarrollo de software, casi 1 año involucrado en proyectos relacionados al procesamiento masivo de datos. Se destaca por su dedicación, responsabilidad, investigación, trabajo en equipo, capacidad analítica y resolución de problemas, aportando con su trabajo al área y al logro de objetivos de la compañía. En la Tabla 1 se aprecia la experiencia profesional del autor.

Tabla 1. *Experiencia profesional*

Cargo: Desarrollador Datio (Scala) Semi Senior	
Empresa: Indra (diciembre 2021 – actualidad)	
Cliente	BBVA
Proyecto	Implementación de lógica en campos de tablas de procesamientos masivos
Funciones	• Implementación de lógica para campos de tablas de procesamiento masivo. • Actualización del diccionario de la metadata de los procesamientos. • Creación de mallas para la ejecución de procesos de reproceso de información. • Realizar el despliegue de soluciones con las herramientas de integración continua (IC). • Trabajo realizado bajo la metodología scrum.

Tecnologías	Git, Jira, Bitbucket, scala, java, maven, IntelliJ IDE, Jupyter Notebook, jenkins
Cargo: Data Engineer Scala Empresa: FactorIT (julio 2021 – diciembre 2021)	
Cliente	INTERBANK
Proyecto	Migración del framework de ingesta de datos
Funciones	• Revisión del actual framework mediante comandos linux a través de SSH, para la captura de requerimientos • Apoyo en el análisis, diseño, implementación y despliegue del nuevo framework de ingesta de datos • Documentación del nuevo modelo de datos • Corrección de incidencias • Trabajo realizado bajo la metodología scrum
Tecnologías	Git, bitbucket, scala, java, maven, IntelliJ IDE, Teradata, Oracle, hive, DBeaver, FileZilla, shell bash, MobaXterm, Notepad++
Cargo: Analista técnico Python Empresa: Experis IT Perú (marzo 2020 – mayo 2021)	
Cliente	CITIBANK
Proyecto	Migración de reportes regulatorios
Funciones	• Migración y optimización de reportes regulatorios SBS (en macros) a Python y KNIME • Implementación de bots para la automatización de la elaboración de reportes regulatorios • Apoyo en la elaboración de reportes contables
Tecnologías	VBA, Excel, Python, Jupyter Notebook, KNIME, Automation Anywhere

Empresa: Consultora multinacional que ofrece soluciones de negocio	
Cargo: Programador técnico .net (diciembre 2019 – marzo 2020)	
Cliente	LA POSITIVA
Proyecto	Soporte de aplicaciones
Funciones	• Corrección de incidencias • Propuesta de mejoras en el sistema Affinity • Trabajo realizado bajo la metodología scrum
Tecnologías	Visual Basic, c#, Sql Server (2008 R2), Informix, Toad For Oracle, Visual Studio (2008, 2010, 2013)
Cargo: Programador Backend .net (julio 2018 – diciembre 2019)	
Cliente	EMPRESA COMERCIALIZADORA DE COMBUSTIBLE
Proyecto	Creación de Portal Web para la autogestión de los clientes flota
Funciones	• Apoyo en el análisis, diseño e implementación del Portal Flotas, cuyo objetivo es brindar control en el abastecimiento de combustible para los clientes flota • Implementación y optimización de API REST en backend • Implementación y optimización de script's SQL • Trabajo realizado bajo la metodología scrum: planning, daily, reunión de sincronización del team, review, retrospectiva • Documentación Ágil del sistema • Elaboración de pase a producción backend • Corrección de incidencias
Tecnologías	C#, NUnit, API REST, TFS, SQL Server, git, GitLab, patrón MVP, Jenkins, SonarQube, Visual Studio
Cargo: Solutions Assistant (marzo 2017 – julio 2018)	

Cliente	EMPRESA COMERCIALIZADORA DE COMBUSTIBLE
Proyecto	Soporte de aplicaciones
Funciones	• Atención de incidencias • Propuestas de mejora de aplicaciones • Capacitación a usuarios en el uso de los sistemas
Tecnologías	Visual Basic, SQL Server, TFS, C#, Oracle, Visual Studio, Remedy

Cargo: Practicante profesional de sistemas

Empresa: Universidad Nacional Mayor de San Marcos, Convenio N° 176-PNICP-PIAP-2015 (abril 2017 – febrero 2018)

Proyecto	Sistema verificación de firmas manuscritas en planillones para la inscripción de partidos políticos
Funciones	• Apoyo en diversas tareas como implementación de algoritmos de pre-procesamiento, documentación del sistema y otros que indique el Coordinador General del Convenio
Tecnologías	Java, Spring, Maven, Git, API Weka para Machine Learning, HTML, CSS, Javascript

Fuente: Elaboración propia

Por el lado académico, el autor completó con éxito la carrera profesional de Ingeniería de Sistemas obteniendo el grado de bachiller en el año 2017, como se muestra en la Tabla 2.

Tabla 2. *Educación superior*

Centro de estudios: Universidad Nacional Mayor de San Marcos	
Escuela	Ingeniería de Sistemas

Grado	Bachiller
Periodo	2012 - 2017

Fuente: Elaboración propia

Por último, el autor ha seguido una serie de cursos y capacitaciones que le han permitido continuar desarrollándose profesionalmente en la línea de carrera de su interés, tal como se aprecia en la Tabla 3.

Tabla 3. *Cursos y capacitaciones*

BS GRUPO	Diplomado en Certified Big Data Applications Developer 2017 II LIMA: R, Python, Hadoop, Spark y Scala (incompleto)	Octubre 2018
Udemy	Machine Learning A-Z TM: Hands-On Python & R In Data Science	Marzo 2021
	Python for Data Science and Machine Learning Bootcamp	Setiembre 2020
Cognitive Class	- Python for Data Science and Machine Learning - Python for Data Science - Data Analysis with Python - Spark Fundamentals I - Spark Fundamentals II - Spark MLlib - Big Data 101 - Hadoop 101 - MapReduce and YARN - Moving Data into Hadoop - Accessing Hadoop Data Using Hive - Introduction to Data Science - Data Science Hands-On with Open Source Tools - Data Science Methodology	2019

	• Data Visualization with Python	
SCRUMstudy	• Scrum Master Certified. • Scrum Fundamentals Certified.	Octubre 2018
HUNDRED	• ASP.NET MVC Fundamentals. • Taller de Java. • Capacitación de Desarrollo de Software SAP ABAP.	Junio 2016

Fuente: Elaboración propia

CAPÍTULO II - CONTEXTO EN EL QUE SE DESARROLLÓ LA EXPERIENCIA

2.1. EMPRESA - ACTIVIDAD QUE REALIZA

La organización donde laboró el autor y que por razones de confidencialidad denominaremos Empresa Consultora Multinacional (en adelante ECM) desarrolla programas que apoyan la estrategia comercial, el mantenimiento de aplicaciones tecnológicas y la subcontratación.

2.2. VISIÓN

La ECM aspira a ser una empresa global que sobresalga en términos éticos y emocionales, impulsada por valores donde cualquier sueño pueda hacerse realidad.

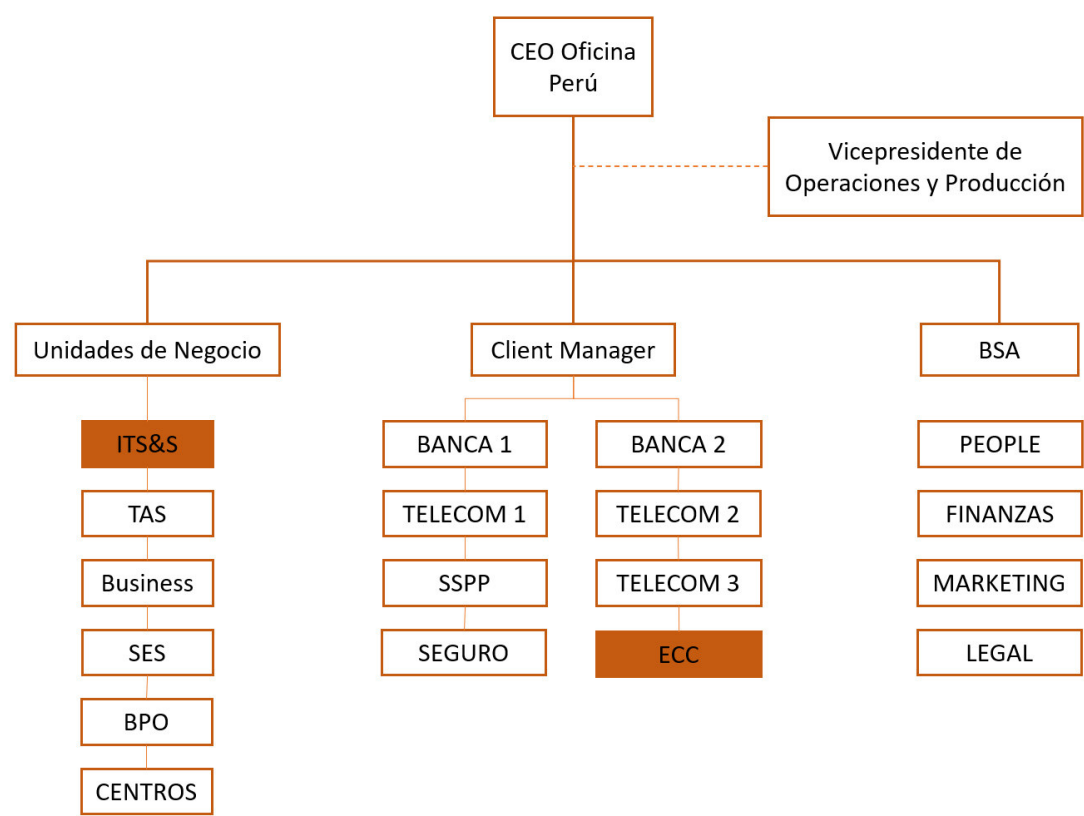
2.3. MISIÓN

La ECM tiene la misión de brindar servicios innovadores de la más alta calidad para ayudar a los clientes a crecer y mejorar.

2.4. ORGANIZACIÓN DE LA EMPRESA

El autor del presente trabajo de suficiencia profesional formó parte de la unidad ITS&S (Information Technology Solutions and Services) y estuvo asignado al cliente ECC. La ECM es una organización internacional con sucursales en varios países del mundo. En la Figura 1 se muestra la estructura organizacional de las oficinas en Perú.

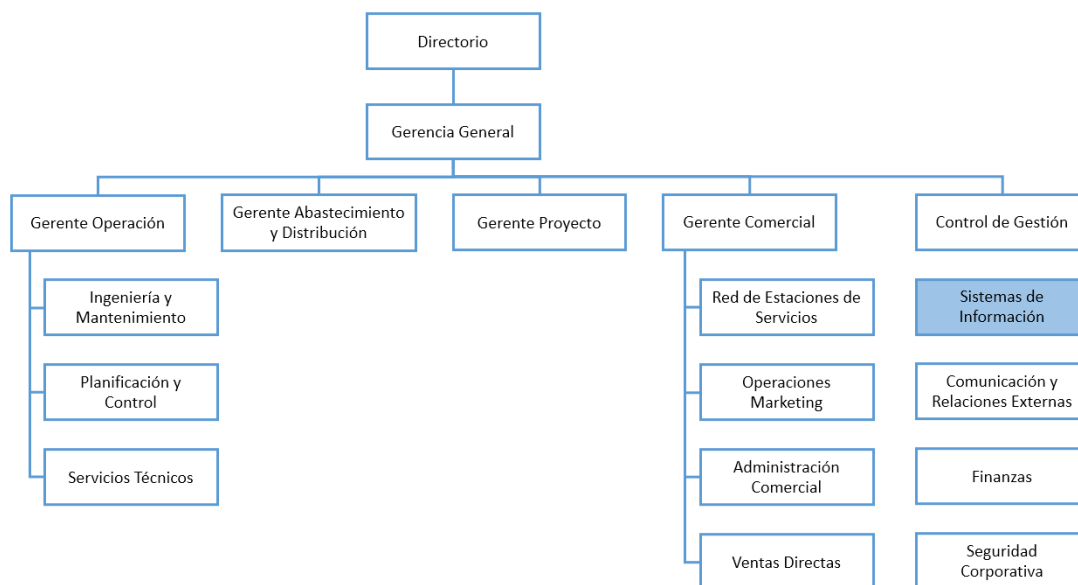
Figura 1. Organigrama de la ECM en las oficinas de Perú



Fuente: Memoria descriptiva de la ECM, 2017.

La estructura organizacional de la ECC estaba compuesta por cuatro gerencias y cinco unidades de soporte, como se visualiza en la Figura 2. El autor del presente informe estuvo laborando en las oficinas de la unidad de Sistemas de Información.

Figura 2. *Organigrama de la ECC*



Fuente: Memoria Anual de la ECC, 2021.

2.5. ÁREA, CARGO Y FUNCIONES DESEMPEÑADAS

El autor de este informe laboró para los siguientes clientes dentro de la ECM:

Cliente Empresa Comercializadora de Combustible (ECC)

Proyecto: Soporte de aplicaciones operativas

Área: Servicio Técnico

Cargo: Asistente de soporte de aplicaciones

Funciones:

- Apoyo a la unidad de Servicios Técnicos en la atención de incidencias de las aplicaciones de la unidad Ingeniería y Mantenimiento (Sistema Cargadero, Sistema de Producción y Libro de Relevos) y de la unidad Planificación y Control (Sistema de Laboratorio, Sistema Evaluación de Contratistas, Sistema Obligación de Contratistas y Sistema de Seguridad Corporativa)
- Adquisición de conocimientos de los procesos de negocios, tanto a nivel técnico como a nivel funcional, que soportan a la unidad de Ingeniería y Mantenimiento.
- Propuestas de mejora en los aplicativos.

- Elaboración y actualización de manuales de usuario.
- Capacitación a los usuarios sobre el uso de los sistemas.

Proyecto: Portal Flotas

Área: Ventas Directas

Cargo: Programador backend .net

Funciones:

- Apoyo en la captura y priorización de requerimientos indicados por el product owner (jefe de la unidad de Operaciones Marketing) en coordinación con los asesores comerciales (unidad de Ventas Directas).
- Implementación de módulos en backend, bajo estándares de desarrollo.
- Revisión de código backend a los desarrollos de otros miembros del equipo.
- Elaboración y seguimiento del pase a producción en backend, que involucra componentes, scripts de base de datos y el rollback, según lineamientos.
- Presentación de las funcionalidades de la Plataforma al product owner, stakeholders y clientes corporativos de la ECC.
- Capacitación de las funcionalidades de la Plataforma al equipo de soporte.
- Atención de incidencias de la Plataforma.

Cliente La Positiva

Área: Comercial

Cargo: Soporte de aplicaciones

Funciones:

- Atención de incidencias del sistema Affinity (Afiliaciones, Desafiliaciones y Anulaciones)
- Implementación de mejoras en el sistema Affinity

2.6. EXPERIENCIA PROFESIONAL REALIZADA EN LA ORGANIZACIÓN

Durante el tiempo en que el autor prestaba sus servicios a la ECC, tuvo la oportunidad de participar en dos proyectos, desde soporte de aplicaciones en la unidad Servicio Técnico hasta la implementación de una plataforma para la unidad Ventas Directas. Su experiencia se resume a los siguientes puntos:

- Corrección de incidencias de los sistemas operativos de la empresa.
- Visita a las instalaciones de la empresa para la adquisición conocimientos de los procesos de negocio.
- Orientación a los usuarios internos y externos en el uso de los sistemas de la empresa.
- Participación de la fase Agile Inception del proyecto para la adquisición de conocimientos acerca del Servicio para Flotas y el planteamiento de posibles soluciones tecnológicas para mejorarlo.
- Implementación y optimización de módulos de la solución.
- Elaboración y optimización de scripts de base de datos para los módulos de la solución.
- Participación en reuniones de sincronización de componentes con otros equipos.
- Presentación de los incrementos de la solución a los interesados del proyecto (product owner y stakeholders).
- Responsable de la mejora de la documentación de componentes de la solución.
- Elaboración y seguimiento de pases a producción de la solución.
- Elaboración del documento técnico ágil de los incrementos de la solución.
- Capacitación de las funcionalidades de la solución al equipo de soporte.

CAPÍTULO III - ACTIVIDADES DESARROLLADAS

3.1. SITUACIÓN PROBLEMÁTICA

La ECC ofrece una amplia gama de combustibles líquidos a la industria, minería, gasolineras independientes y otros negocios, y proporciona un asesoramiento técnico en el transporte, almacenamiento, distribución y suministro de acuerdo con las necesidades específicas de calidad, seguridad y protección del medio ambiente de cada cliente.

La empresa tiene una moderna red de EES distribuidos en todo el país. Las estaciones permiten, entre otras actividades, abastecer de combustible a las unidades vehiculares del público en general y de los clientes corporativos con los cuales la ECC mantiene una alianza estratégica.

Entre los productos y servicios que ofrece la ECC a los clientes corporativos, que en adelante lo denominaremos como clientes, existe el Servicio para Flotas, donde se ofrece promociones y descuentos respecto a la compra de combustible. Cuando se contrata este servicio, la ECC entrega tantas tarjetas flota (banda magnética) como el número de vehículos que tenga los clientes.

Con la tarjeta, los conductores se acercaban a cualquiera de las EES para abastecer de combustible al vehículo que tengan asignado. En este contexto, los clientes elaboraban un presupuesto que estimaba la cantidad de recursos económicos que destinaban a su flota para que sea utilizado en un tiempo determinado.

En ocasiones, los clientes veían comportamientos sospechosos de consumo de combustible de su flota, por lo que solicitaron a la ECC que se les envíe un reporte acerca del consumo realizado por su flota al final del día. Estos reportes eran

elaborados desde el BackOffice (sistema interno de la ECC) y enviados por los asesores comerciales de la unidad de Ventas Directas que dependía de la Gerencia Comercial.

En base al reporte, los clientes verificaban que había consumos de combustible injustificados, generándoles pérdida de dinero y asumiendo que se trataría de robo. Entre las posibles modalidades de robo, podemos destacar los siguientes (COMSATEL, 2019):

- Drenado de combustible: mediante una manguera se drena el combustible del tanque del vehículo y se coloca en un recipiente.
- Consumo de combustible inferior al facturado: se efectúa al momento de comprar el combustible, donde se pide que una parte sea colocada en una cubeta o en otro vehículo.

Dicha presunción condujo al cliente a buscar formas de controlar el consumo de combustible de su flota.

3.1.1. DEFINICIÓN DEL PROBLEMA

Falta de controles que impidan el consumo injustificado de combustible de las flotas vehiculares de los clientes corporativos.

3.2. SOLUCIÓN

Desde la Gerencia Comercial se planteó la implementación de una solución tecnológica que permitiera a los clientes gestionar, monitorear y controlar el consumo de combustible de su flota vehicular. Esta solución permitiría a los clientes crear controles que restrinjan el abastecimiento injustificado de combustible y reduciría la carga administrativa de los asesores comerciales.

En la parte técnica, la solución involucraría la creación y la modificación de los siguientes artefactos tecnológicos:

Artefactos nuevos

- *Plataforma*, una solución web que permitiría a los clientes gestionar, monitorear y controlar el consumo de combustible de su flota.
- *Servicio de Comunicación*, interface de comunicación entre el Punto de Venta y la Plataforma.

Artefactos modificados

- *BackOffice*, desarrollado sobre la plataforma de la ECC y cuyo usuario principal es interno (por ejemplo, los asesores comerciales) y que tendría todas las funcionalidades de la Plataforma.
- *Punto de Venta*, módulo ubicado en las EES y que es utilizado para despachar el combustible, y debería soportar los controles creados desde el BackOffice y la Plataforma.

3.2.1. OBJETIVOS

Objetivo General

Desplegar en el entorno productivo y de forma incremental las funcionalidades de una plataforma tecnológica que permita a los clientes gestionar, monitorear y controlar el consumo de combustible de su flota vehicular en el mediano plazo, de modo que se eviten mayores pérdidas.

Objetivos Específicos

- Revisar los posibles controles de consumo de combustible.
- Implementar los controles de consumo de combustible en el BackOffice.
- Habilitar las funcionalidades de gestión, monitoreo y control de consumos de combustible en la Plataforma para clientes corporativos.
- Implementar un servicio para comunicar el Punto de Venta con la Plataforma.
- Modificar el módulo del Punto de Venta para validar los controles al momento de

realizar la venta de combustible.

3.2.2. ALCANCE

El proyecto se enfoca en la implementación de una solución tecnológica que se ofrece a los clientes corporativos quienes cuenten con una flota vehicular.

Esta solución incluye un alcance nacional abarcando las 137 estaciones de servicio en donde se ofrece el Servicio para Flotas.

3.2.3. ETAPAS Y METODOLOGÍA

El proyecto implementación de una solución tecnológica fue desarrollado bajo la metodología de trabajo scrum adaptado a la ECC, donde se tuvo las fases, procesos y salidas que se muestra en la tabla 4.

Tabla 4. *Fases y procesos del proyecto*

Fase	Proceso	Salida
Agile Inception	Explicación general del contexto	Sincronización de los participantes
	Simulación de la implantación de la solución	Supuestos de la implantación de la solución
	Elaboración del backlog del producto	Backlog del producto
Inicio	Definición del proyecto	Documento de inicio del proyecto
	Formación del equipo	Presentación del equipo
	Planificación del lanzamiento	Plan de cronograma del proyecto
Planificación y estimación (Sprint planning)	Estimación de historias de usuario	Historias de usuario puntuadas
	Creación del sprint backlog	Listado de historias de usuario en el sprint
Implementación (Desarrollo del sprint)	Refinamiento del backlog	Backlog del producto actualizado
	Funcionalidades de la Plataforma	Actualización del Documento TAG de funcionalidades
	Generación de entregables	Scripts de base de datos

	Pruebas funcionales	Listado de componentes en backend
		Versionado Notificación de conformidad de historias de usuario Notificación de observaciones de las historias de usuario
Revisión (Sprint review)	Demostración a los interesados	Calificación de la funcionalidad de las historias de usuario Listado de observaciones de las historias de usuario
Lanzamiento	Generación del release	Documento Parte Documento técnico ágil Evidencia ejecución de los scripts de rollback Scripts de rollback Documento TAG de funcionalidades
	Pase a producción	Actualización del documento de actividades de despliegue
Evaluación del scrum team (Sprint retrospective)	Agradecimientos de los miembros del equipo	Listado de miembros destacados
	Identificación de buenas prácticas	Listado de buenas prácticas
	Reporte de bloqueos	Listado de bloqueantes
	Aspectos a mejorar	Listado de actividades a mejorar

Fuente: Elaboración propia

3.2.4. FUNDAMENTOS UTILIZADOS

3.2.4.1. CONTROL DE COMBUSTIBLE EN LA GESTIÓN DE FLOTAS

La gestión de flotas consiste en gestionar y organizar vehículos comerciales, permitiendo al gestor de flota hacer seguimiento de la ubicación, el comportamiento y el estado de los vehículos, con el fin de incrementar la eficiencia y la productividad, a la vez que mejora la seguridad de los conductores y los vehículos (MOVILDATA, 2022).

Como el combustible es uno de los gastos básicos de la flota vehicular, es

importante que el gestor de flotas realice un seguimiento al control de consumo de combustible día a día, por lo tanto, los dos indicadores más importantes, en este sentido, son (Redson, 2021):

- Cantidad de combustible consumido por los vehículos (de forma individual como colectiva)
- El rendimiento por litro de combustible, en kilómetros.

Para conocer la eficiencia en el consumo de combustible, se divide los kilómetros recorridos por el vehículo en un período de tiempo dado por el consumo total de la flota; conocer este rendimiento ayudará a comprender si el consumo de combustible de la flota es más o menos eficiente (Redson, 2021).

3.2.4.2. SCRUM

Es una metodología que ofrece la entrega temprana de resultados de forma rápida, flexible y eficaz en todo el proyecto. Su aplicación conlleva un esfuerzo de colaboración para crear un nuevo producto o servicio, o cualquier otro resultado. La clave de esta metodología radica en el desarrollo de ciclos de trabajos cortos (o Sprints) lo que permite tomar decisiones en fases tempranas del proyecto y, por consiguiente, un bajo costo ante un cambio (SCRUMstudy™, 2013).

Roles scrum

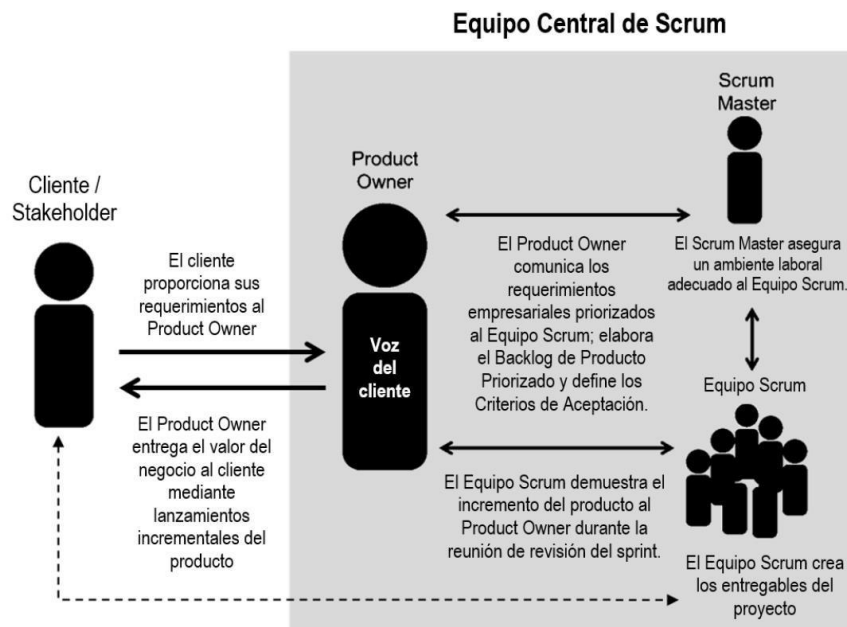
En la metodología scrum es importante la participación activa de las personas que cuyos roles se definen a continuación (SCRUMstudy™, 2013):

- *Product Owner*: es el responsable de comunicar al Equipo Scrum los requerimientos de la solución que se desea implantar en la organización a la que representa, para ello debe contar con un conocimiento sólido del sector de negocio en donde se va aplicar el proyecto, caso contrario se apoya de los Stakeholders para tener una visión más clara y amplia de dicho sector, por lo que su facilidad de comunicación es importante (SCRUMstudy™, 2013).

- *Scrum Master*: encargado de resolver los obstáculos que impidan el desarrollo de las tareas de los miembros del Equipo Scrum. Mantiene al equipo alineados a los procesos scrum y sus entregables (SCRUMstudy™, 2013).
- *Equipo Scrum*: o scrum team, encargado de la implementación de las historias de usuario durante el spring trabajando como una sola unidad, donde internamente se apoyan entre sí. (SCRUMstudy™, 2013).
- *Stakeholders*: incluye a los clientes, usuarios y patrocinadores. Brindan información para el desarrollo del proyecto y en ocasiones puede participar en varios procesos scrum (SCRUMstudy™, 2013).

En la siguiente figura se grafica la interacción de los roles en un proyecto Scrum.

Figura 3. Roles scrum



Fuente: A Guide to the SCRUM BODY OF KNOWLEDGE (SBOK™ Guide). Tomado de SCRUMstudy™, 2013

Ceremonias scrum

- *Agile Inception*: es un conjunto de actividades que permiten a los equipos ágiles empezar el desarrollo de un producto de forma coherente y efectiva (Rasmusson,

2010).

- *Sprint*: es todo el trabajo necesario para la implementación de la solución, incluye las ceremonias Sprint Planning, Daily Scrums, Sprint Review y Sprint Retrospective. Son eventos que pueden durar 1 mes o menos. Un nuevo sprint empieza luego de la finalización del sprint anterior (Schwaber y Sutherland, 2020).
- *Sprint Planning*: reunión donde el scrum team estima el trabajo para implementar las historias de usuario seleccionadas del backlog del producto por el product owner, en el sprint (Schwaber y Sutherland, 2020).
- *Daily Scrum*: es un evento diario que puede durar 15 minutos aproximadamente, donde cada miembro indica lo que ha revisado ayer, lo que hará hoy y si presenta algún impedimento que obstaculice su compromiso (Schwaber y Sutherland, 2020).
- *Sprint Review*: reunión donde el scrum team presenta el resultado del sprint a los interesados del proyecto (product owner y stakeholders). El product owner puede optar por aceptar o rechazar los resultados, dependiendo si se cumplan con los criterios de aceptación (Schwaber y Sutherland, 2020).
- *Sprint Retrospective*: el scrum team identifica los supuestos que los llevaron por mal camino y exploran sus orígenes, analizando qué salió bien durante el sprint, qué problemas se detectaron y cómo se resolvieron (o no) dichos problemas. Con este evento se termina el sprint (Schwaber y Sutherland, 2020).

Artefactos Scrum

- *Product Backlog*: lista de historias de usuario priorizadas por el producto owner para incrementar las funcionalidades del producto. Aquellas historias que el scrum team puede dar por finalizado dentro de un Sprint se consideran preparados para ser seleccionados en el Sprint Planning (Schwaber y Sutherland, 2020).
- *Sprint Backlog*: lista de historias de usuario seleccionadas del backlog del producto a fin de implementarla en el sprint. Esta selección lo realiza el product owner en coordinación con el scrum team (Schwaber y Sutherland, 2020).

- *Incremento*: aumenta las funcionalidades del producto. Cada incremento se suma a todos los incrementos anteriores y se revisa minuciosamente, lo que garantiza que todos los incrementos funcionen juntos (Schwaber y Sutherland, 2020).

Procesos Scrum

Los procesos Scrum se agrupan en fases, tal y como se muestra en la tabla 5.

Tabla 5. *Procesos Scrum*

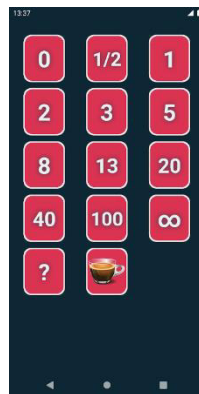
Fase	Procesos fundamentales
Inicio	1. Crear la visión del proyecto 2. Identificar al Scrum Master y Stakeholder(s) 3. Formar Equipos Scrum 4. Desarrollar épica(s) 5. Crear el Backlog Priorizado del Producto 6. Realizar la planificación de lanzamiento
Planificación y estimación	7. Crear historias de usuario 8. Estimar historias de usuario 9. Comprometer historias de usuario 10. Identificar tareas 11. Estimar tareas 12. Crear el Sprint Backlog
Implementación	13. Crear entregables 14. Realizar Daily Standup 15. Refinar el Backlog Priorizado del Producto
Revisión y retrospectiva	16. Demostrar y validar el sprint 17. Retrospectiva del sprint
Lanzamiento	18. Enviar entregables 19. Retrospectiva del proyecto

Fuente: A Guide to the SCRUM BODY OF KNOWLEDGE (SBOK™ Guide). Tomado de SCRUMstudy™, 2013

3.2.4.3. SCRUM POKER CARDS (AGILE)

Es una herramienta poderosa para realizar el planning poker, que es una técnica de estimación en Scrum. Permite que las reuniones aburridas se vuelvan divertidas (Scrum Poker Cards, 2022).

Figura 4. *Scrum Poker Cards, una herramienta para puntuar historias de usuario*



Fuente: Scrum Poker Cards. Tomado de Google Play, 2022

Los símbolos que se visualizan en la aplicación indican la complejidad de la historia de usuario, siendo 0 que indica ninguna complejidad y 100 la de mayor complejidad. El símbolo ∞ se refiere a que se conoce el contexto de la historia de usuario, pero no puede definir claramente las tareas involucradas. El símbolo “?” se refiere a que no se conoce el contexto de la historia de usuario y mucho menos se puede definir las tareas involucradas. El símbolo de la taza hace referencia a que el participante está demasiado cansado para continuar y necesita un descanso (Appvizer, 2020).

3.2.4.4. API REST

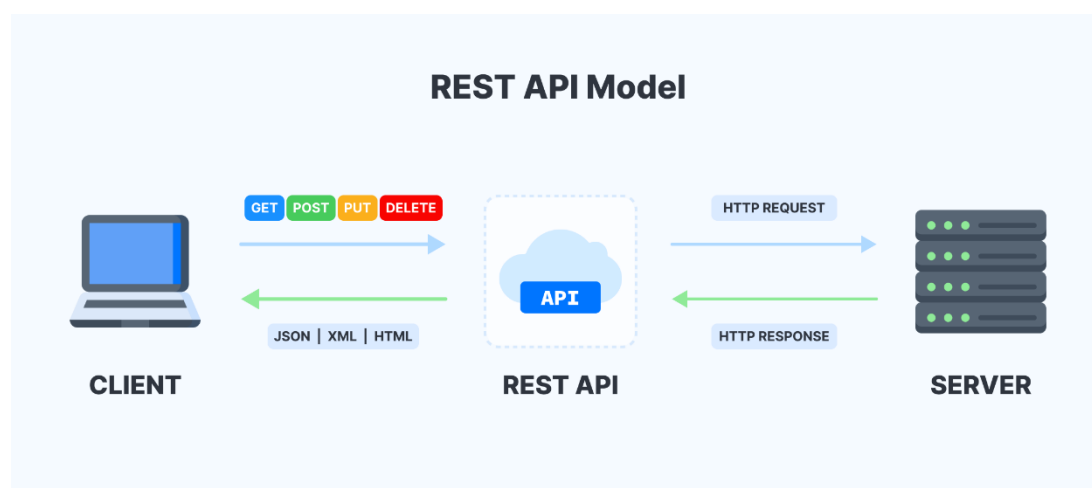
Una API (Application Programming Interfaces) es un grupo de reglas que determinan cómo las aplicaciones o dispositivos se conectan y comunican entre sí. Una API REST es una API que se basa en los principios de diseño REST, un estilo arquitectónico, también conocido como transferencia de estado representacional (IBM, 2022).

Una API permite a una aplicación o servicio acceder a un recurso dentro de otra aplicación o servicio. La aplicación o servicio que realiza la petición se denomina cliente y la aplicación o servicio que recibe la petición se denomina servidor (IBM, 2022).

La información que el servidor entrega al cliente puede estar prácticamente en cualquier formato, entre ellos JavaScript Object Notation (JSON), HTML, XLT, Python, PHP o un texto sin formato. JSON es el más utilizado porque es legible tanto por los seres humanos como por las máquinas, a su vez que es independiente de un lenguaje de programación (IBM, 2022).

Las API REST se comunican mediante solicitudes HTTP para realizar funciones generales en base de datos, como crear, leer, actualizar y eliminar datos en recursos. Por ejemplo, una API REST usaría una petición GET para recuperar el dato, una petición POST para crearlo, una petición PUT para actualizarlo y una petición DELETE para eliminarlo (IBM, 2022).

Figura 5. *API REST para proporcionar datos del servidor al usuario de una aplicación*



Fuente: ¿Qué es la API REST y en qué se diferencia de otros tipos? Tomado de AppMaster, 2022

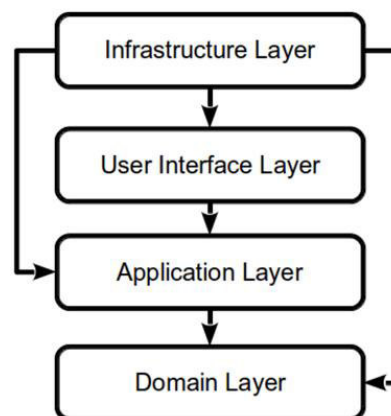
3.2.4.5. DOMAIN DRIVEN DESIGN (DDD)

Traducido al español como Diseño Dirigido por el Dominio, es un conjunto de patrones que permiten crear software en base a una mejor interpretación de los términos del negocio, ayudando a mejorar los procesos que tienen que ver con la construcción del código (Evans, 2004).

Uno de los principios de este diseño es la creación de un lenguaje ubicuo, que pueda ser entendido por el negocio y por los desarrolladores. La idea es que este lenguaje no sea documentado, sino sea incorporado dentro del mismo código, conllevando a que los nombres de variables, métodos, etc. acaben reflejando mejor el entendimiento del dominio del negocio (Evans, 2004).

Una de las arquitecturas más utilizadas es Layers N-Tier, donde se realiza la separación de responsabilidades en capas particionando la complejidad según su responsabilidad. En la siguiente figura podemos ver un ejemplo de esta arquitectura por capas (Cuéllar, 2016).

Figura 6. *Ejemplo de una arquitectura basado en DDD*



Fuente: Domain-Driven Design. Arquitectura. Tomado de Cuéllar, 2016

- *Infrastructure Layer*: Capa superior que provee/implementa todos los servicios de bajo nivel, que depende de interfaces definidas en los componentes de alto nivel (*User Interface Layer*, *Application Layer*, *Domain Layer*).
- *User Interface Layer*: Capa de presentación al usuario que contiene componentes

para la visualización de datos. No tiene lógica de negocio siendo el cliente directo de la *Application Layer*.

- *Application Layer*: Contiene la lógica de aplicación mediante servicios. Orientada a casos de uso de los usuarios, recibiendo los parámetros desde *User Interface Layer*. Cliente directo de la *Domain Layer*.
- *Domain Layer*: Contiene la lógica de negocio mediante servicios y modelado de dominio. Además, contiene las interfaces necesarias de acceso a datos (respositorios).

3.2.4.6. NET FRAMEWORK

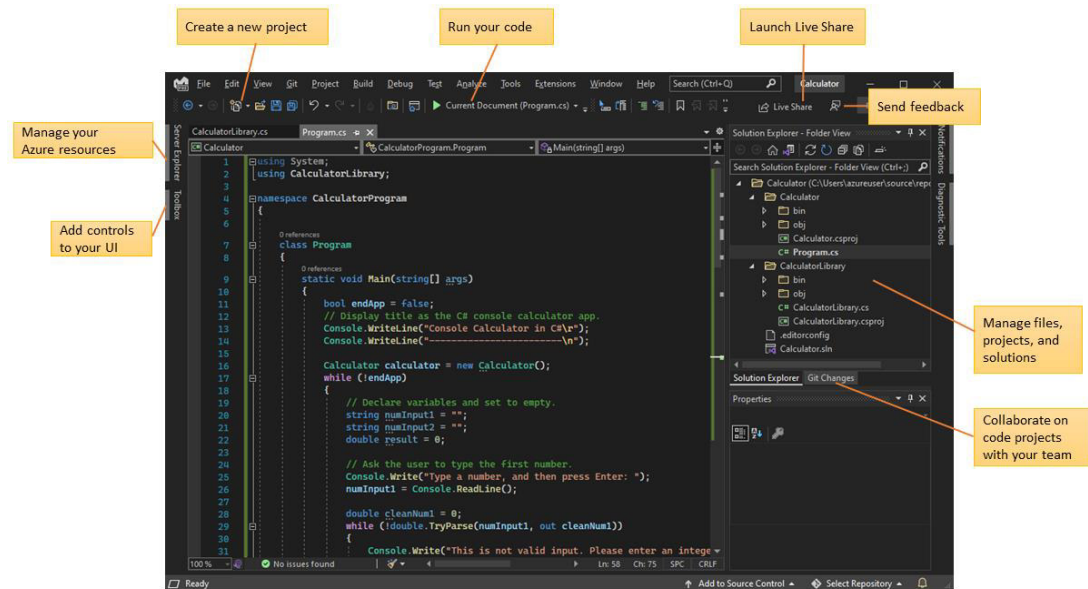
Es un conjunto de tecnologías que proporciona Microsoft para una sencilla programación orientada a las redes e internet, y que es independiente de la plataforma hardware donde se encuentre instalado. Brinda soluciones de código que son usados por los programas (mediante librerías dll), y que gestiona programas escritos para este Framework (Microsoft, 2021).

Este conjunto de tecnologías brinda un entorno de desarrollo que usa un software sencillo, aumentando la seguridad de los programas y reduciendo las vulnerabilidades. Para programar en .NET existen varios lenguajes de programación (F#, C++/CLI, etc.) de los cuales destacan C# y Visual Basic (Microsoft, 2021).

3.2.4.7. VISUAL STUDIO

Un entorno de desarrollo integrado (IDE) es un programa con muchas características que soportan muchos aspectos del desarrollo de software. Visual Studio es un IDE con un panel de inicio creativo que se puede usar para editar, depurar y compilar código y, luego, publicar una aplicación. Lejos del editor y del depurador estándar que proporcionan la mayoría de IDE, Visual Studio cuenta con compiladores, herramientas de finalización de código, diseñadores gráficos y muchos aspectos más para facilitar el proceso de desarrollo de software (Microsoft, 2022).

Figura 7. Imagen del IDE de Visual Studio



Fuente: Le damos la bienvenida al IDE de Visual Studio. Tomado de Microsoft, 2022

En la figura anterior se visualiza el IDE de Visual Studio con las ventanas principales y su funcionalidad (Microsoft, 2022):

- En la parte superior derecha del Explorador de soluciones, se puede ver y gestionar los archivos de código y navegar por ellos. El Explorador de soluciones ayuda a agrupar los archivos en soluciones y proyectos.
- La ventana del editor muestra el contenido del archivo donde se podrá editar el código o diseñar una interfaz de usuario, como una ventana con botones y cuadros de texto.
- En la parte inferior derecha de Git Changes (Cambios de Git), se puede hacer seguimiento de elementos de trabajo y compartir código con otros usuarios a través de tecnologías de control de versiones (Git y GitHub).

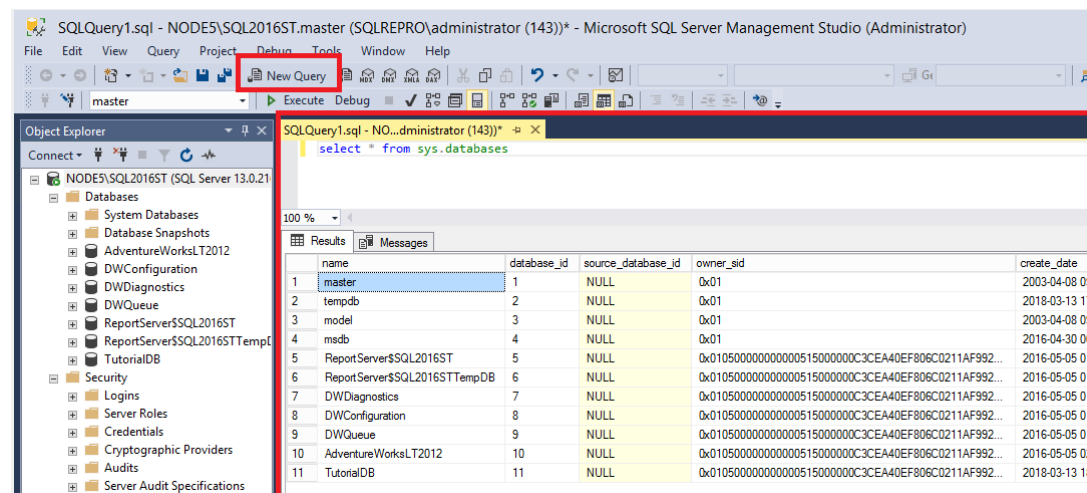
3.2.4.8. SQL SERVER

Es un sistema de gestión de base de datos relacional de Microsoft, que actúa como un servidor que brinda servicio para aplicaciones cliente que pueden funcionar ya sea en el mismo ordenador o en otro ordenador. Tiene las siguientes características

(nextech, 2021):

- Soporte de transacciones.
- Escalabilidad, estabilidad y seguridad.
- Soporte de procedimientos almacenados.
- Incluye un entorno gráfico de administración que permite el uso de comandos de lenguaje de definición de datos (DDL) y lenguaje de manipulación de datos (DML).
- Permite trabajar en modo cliente-servidor.
- Permite administrar información de otros servidores de datos.

Figura 8. Ventana de consultas de SQL Server



Fuente: Componentes y configuración de SQL Server Management Studio. Tomado de Microsoft, 2021

3.2.4.9. SWAGGER USER INTERFACE (SWAGGER UI)

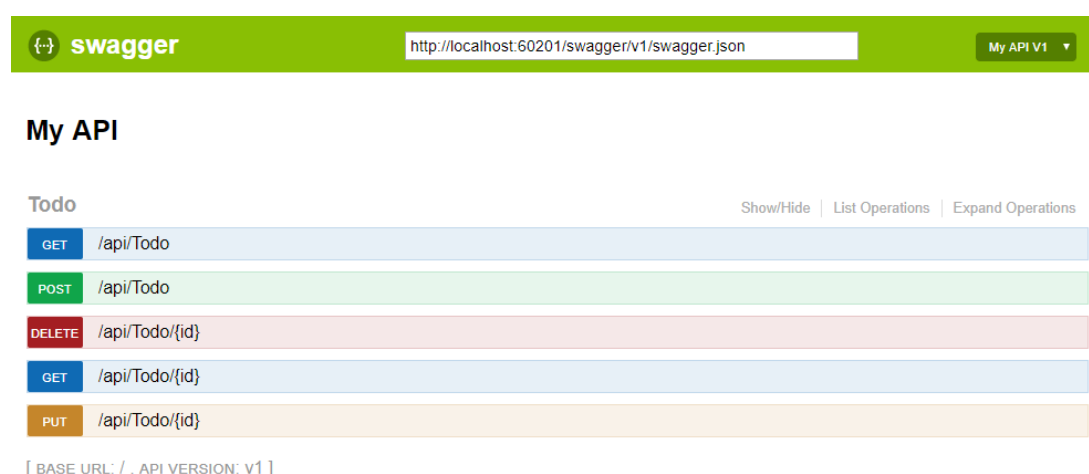
Una API pierde su sentido si no es accesible y si no cuenta con una documentación que ayude a entenderla. Uno de los mayores problemas de las APIs es que tienen documentación poco útil. Ante ello nace Swagger cuyo objetivo es estandarizar el vocabulario que utilizan las APIs (chakray, 2022).

Swagger es una serie de reglas, especificaciones y herramientas que ayudan

a documentar las APIs. De esta manera, podemos realizar documentación que sea realmente útil, que todo el mundo lo entienda (chakray, 2022).

Swagger UI permite que cualquier persona, ya sea el equipo de desarrollo o los consumidores finales, visualice e interactúe con los recursos de la API sin tener desarrollada ninguna lógica de implementación. Permite probar y documentar API REST (swagger, 2022).

Figura 9. *Interfaz de usuario de Swagger*



Fuente: Documentación de la API web de ASP.NET Core con Swagger/OpenAPI. Tomado de Microsoft, 2022

3.2.4.10. GIGYA CONNECTOR

Gigya era una empresa de software que proporcionaba gestión de acceso e identidad de clientes y soluciones de gestión de datos de consentimiento. En 2017, SAP anunció la adquisición de Gigya. Hoy, las capacidades de software de Gigya están integradas en las soluciones de datos de clientes de SAP. Esta cartera ayuda a las empresas a aumentar los ingresos al ofrecer experiencias de cliente personalizadas y confiables a escala (SAP, 2022).

3.2.4.11. ANGULAR

Es una plataforma de desarrollo construida en TypeScript. Como plataforma,

Angular incluye (Angular, 2022)

- Un marco que incluye componentes para crear aplicaciones web escalables.
- Un conjunto de bibliotecas bien integradas que cuentan con una amplia variedad de funciones, incluido el enrutamiento, la gestión de formularios, la comunicación cliente-servidor y más.
- Un conjunto de herramientas de codificación que ayudan a desarrollar, compilar, probar y actualizar el código.

Con Angular, se aprovecha una plataforma que puede escalar desde proyectos de un solo desarrollador hasta aplicaciones de nivel empresarial. Angular está diseñado para que la actualización sea lo más sencilla posible, por lo que los desarrolladores pueden aprovechar los últimos desarrollos con el mínimo esfuerzo. Angular cuenta con un grupo diverso de más de 1,7 millones de desarrolladores, autores de bibliotecas y creadores de contenido.

Figura 10. *Elementos de un componente en Angular*



Fuente: Componentes en Angular. Tomado de Medium, 2019

3.2.5. *IMPLEMENTACIÓN DE LAS ÁREAS, PROCESOS, SISTEMAS Y BUENAS PRÁCTICAS*

Antes de iniciar con el desarrollo de esta sección, a continuación, se explica el contexto donde se desarrolló el proyecto.

Contexto del proyecto

El autor de este informe trabajó en coordinación con la unidad de Sistemas de Información como programador backend desde julio del 2018 hasta diciembre del 2019, en el proyecto implementación de una solución tecnológica para la gestión y control de consumo de combustible para la unidad de Ventas Directas que dependía de la Gerencia Comercial de la ECC.

Este proyecto formaba parte de un proyecto mayor cuyo objetivo era la modernización de las estaciones de servicio (en adelante EES) de la ECC a través de la modificación de los procesos de negocio, aplicando tecnologías de vanguardia y utilizando scrum como marco de trabajo.

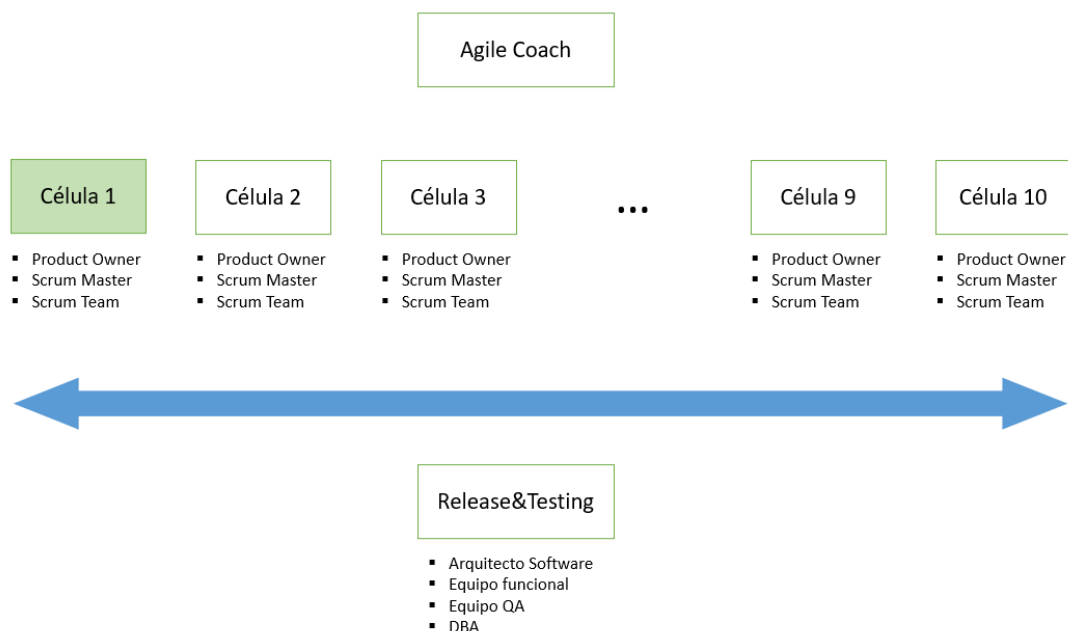
A medida que el proyecto de modernización avanzaba, la ECM contrataba nuevos perfiles requeridos para cada subproyecto. Aproximadamente en 4 meses, se formaron equipos con funciones muy específicas, los cuales se explica a continuación:

- *Agile coach*, quien brindaba lineamientos de gestión de recursos a los scrum masters y dirigía la reunión de Scrum of Scrums (SoS), entre otras actividades.
- *Equipos (o células)*, que en total fueron diez y cada uno estaba conformado por
 - Product Owner (PO): brindaba detalle de las historias de usuario, los priorizaba, verificaba su funcionamiento y los juntaba en un release para su puesta en producción.
 - Scrum Master (SM): facilitaba los recursos/herramientas al scrum team, gestionaba la resolución de impedimentos, dirigía ceremonias de scrum y generaba métricas de evaluación del equipo.
 - Scrum Team (ST): equipo encargado de la implementación de las historias de usuario, las analizaba para ver su factibilidad, las puntuaba, las aclaraba en coordinación con el PO, las gestionaba para su pase a producción, las corregía si hubiese incidencias, entre otras actividades.

- *Equipo Release&Testing*, quien brindaba apoyo a todas las células y estaba integrado por
 - Arquitecto Software: elaboraba la arquitectura de los proyectos software en backend para cada célula, generaba el versionamiento de cada uno de estos, indicaba lineamientos de desarrollo (backend y base de datos), realizaba pases a preproducción / producción de los componentes en backend en coordinación con el scrum team.
 - Equipo funcional: brindaba información acerca del negocio, coordinaba con el arquitecto para la generación de proyectos software, entre otras actividades.
 - Equipo QA: encargado de realizar pruebas de software, automatizarlas mediante la creación de pipelines, indicar lineamientos de pruebas de calidad, entre otras actividades.
 - Administrador de base de datos (DBA): brindaba lineamientos de desarrollo en base de datos, identificaba y notificaba mejoras en los scripts, realizaba pases a preproducción / producción de los scripts de base de datos en coordinación con el scrum team, entre otras actividades.

En la figura 11 se visualiza la organización de los equipos dentro del proyecto modernización de EES.

Figura 11. Organización de los equipos del proyecto modernización de EES



Fuente: Elaboración propia.

El autor de este informe realizó sus actividades en la célula 1 con el perfil programador backend. El objetivo de esta célula era la creación de una plataforma web donde los clientes de la ECC pudieran gestionar y controlar el consumo de combustible de su flota vehicular desde la comodidad de su pc, tablet o smartphone con acceso a internet.

Luego de explicar el contexto donde se desarrolló el proyecto, y según los procesos indicados en la sección 3.2.3 Etapas y Metodología, a continuación, se detalla cada una de las fases y procesos desarrollados para la implementación del proyecto haciendo hincapié en aquellas actividades donde el autor estuvo más involucrado con el rol de programador backend: implementación de la Plataforma.

I. Fase Agile Inception

Fueron una serie de reuniones dirigidas por un Agile Coach, donde se brindaba capacitación acerca que cómo se gestiona el trabajo desde la metodología scrum, explicando cada ceremonia y las dinámicas que se debería realizar en cada una de estas. Entre los participantes podemos mencionar:

- Por el lado de la ECC
 - Gisela, jefa de la unidad Operaciones Marketing
 - Richard y otros, Asesores comerciales de la unidad Ventas Directas
 - Jefes de estaciones de servicio, de la unidad Ventas Directas
 - Marcela, Analista de Sistemas de la unidad Sistemas de Información

- Por el lado de la ECM
 - José, Agile Coach
 - Giovanni, Gestor de Proyectos
 - Jorge y otros, analistas programadores full-stack
 - Raúl y otros, programadores backend
 - Viviana y otros, analistas y asistentes de Calidad
 - Manuel, Arquitecto Software
 - Luis, Administrador de Base de Datos

Una vez finalizado la capacitación, el Agile Coach invitada a que los participantes de la ECC comentaran acerca de qué se trataría el proyecto, desde la problemática hasta los beneficios que traería la implantación de una posible solución tecnológica.

I.1. Explicación general del contexto

En este proceso, los participantes por el lado de la ECC explicaron los servicios que se brindaba en esta empresa, entre ellos el Servicio para Flota, el cual mencionaron que necesitaban una solución tecnológica que pudiera mejorar dicho servicio.

Con esta información, los participantes por el lado de la ECM realizaron una serie de preguntas acerca del tipo de solución que sería apropiado para mejorar el Servicio para Flotas.

El objetivo de estas reuniones fue que todos estuvieran enterados de las

I.2. Simulación de la implantación de la solución

A continuación, se muestra una de las dinámicas realizadas en este proceso, donde se aprecia el planteamiento de los objetivos, roles y habilidades, reglas y actividades, y valores morales, entorno a la solución.

34

I.3. Elaboración del backlog del producto

La interacción entre los participantes durante la Agile Inception hizo que se recabara información acerca de la problemática y cómo una solución tecnológica podría ayudar a los clientes a controlar el consumo de combustible de su flota. Toda esta información se tradujo a historias de usuario, creándose la primera versión del backlog del producto.

A continuación, en la figura 13 se aprecia una vista preliminar de la primera versión del backlog del producto.

Figura 13. *Primera versión del backlog del producto*

Releases	Nro. Sprint	Fecha Inicio	Fecha Fin	Historias de Usuario
release #1 (Impactos: Back-office,)	1	13-Ago-18	24-Ago-18	Implementación del código QR
release #2 (Impactos: back-office,)	2	27-Ago-18	7-Set-18	Reglas Comerciales (por frecuencia, producto) en back-office y
	3	10-Set-18	21-Set-18	Impresión de notas despacho del conductor y de los centros de coste, precio en NDES
	4	24-Set-18	5-Oct-18	Listar tarjetas (Portal)
	5	9-Oct-18	19-Oct-18	Actualización de estados de tarjetas (Portal)
	6	22-Oct-18	2-Nov-18	Reporte de consumos (Portal) Reporte de saldos (Portal)
	7	5-Nov-18	16-Nov-18	Login (Portal) Integración con .com Registro del usuario administrador de flotas (back-office)
				Mejoras a listar tarjetas, reporte de saldos y consumos
				Resúmenes en las cabeceras en formulario de tarjetas y reporte de consumos
				Acceso a usuario con más de una empresa y/o más de un perfil (Portal)
				Asociación masiva de reglas por frecuencia y producto a tarjetas flotas (Portal)
				Integración del login con Gigya
				Listar reglas comerciales (formulario)
				Mejoras al formulario de tarjetas
	8	19-Nov-18	30-Nov-18	Resumen de cantidades en Reglas comerciales (cabecera - portal)
				Crear nueva regla comercial (por frecuencia y producto)
				Consulta de reglas comerciales (Portal)
				Editar reglas comerciales (Portal)
				Activa/inactivar reglas comerciales (Portal)
Release #3 (Impactos: Portal)	9	3-Dic-18	14-Dic-18	Reglas comerciales

Fuente: Extraído del proyecto

Con la información obtenida hasta este momento, ya se iba planteando el cómo sería las especificaciones para las siguientes fases y procesos del proyecto.

II. Fase de inicio

El proyecto surgió como una necesidad de la unidad de Ventas Directas para mejorar el Servicio para Flotas ya que los clientes venían reportando consumo injustificado de combustible de su flota. Además, había el riesgo de terminar los contratos con los clientes debido a la competencia que ya contaba con mecanismos que podían controlar el consumo de combustible.

Antes del desarrollo del proyecto, el flujo del Servicio para Flotas era de la siguiente manera:

a. **Elección de modalidad de pago:** el cliente pactaba el contrato con la ECC para solicitar el Servicio para Flotas. Aquí el cliente podía elegir entre tres modalidades de pago:

- Crédito directo, donde se realiza una previa evaluación de capacidad de pago.
- Prepago, donde el cliente realiza el abono a una cuenta.
- Al contado, donde se accede a descuentos por volumen.

b. **Establecer tipos de saldo:** Cerrado el contrato, para el caso de las modalidades crédito directo y prepago, se establece dos tipos de saldo

- Saldo cliente: monto total que tiene el cliente en el Servicio para Flotas.
- Saldo tope: monto máximo que puede acumular la tarjeta flota. No puede ser mayor al Saldo cliente.

c. **Distribuir saldos:** la ECC reparte el saldo cliente entre los saldos tope de las tarjetas de acuerdo a las especificaciones del cliente indicadas en el contrato. Por ejemplo: tenemos al cliente X que tiene una flota de tres vehículos, y solicita el Servicio para Flotas de acuerdo a las siguientes especificaciones:

- Saldo cliente: S/. 1000
- Saldo tope para el vehículo A: S/. 100
- Saldo tope para el vehículo B: S/. 200
- Saldo tope para el vehículo C: S/. 300

El saldo total destinado para los vehículos asciende a S/. 600, este monto se descuenta del saldo cliente, por lo que al final el saldo cliente disponible es de S/. 400. Este saldo puede ser distribuido nuevamente entre los saldos de las tarjetas de los vehículos, pero hasta donde su saldo tope lo permita o hasta donde el saldo cliente lo permita.

Por ejemplo, un conductor abasteció al vehículo A en dos oportunidades y el saldo disponible es de S/. 20,

- Abastecimiento 1: S/. 40
- Abastecimiento 2: S/. 40
- Saldo disponible de la tarjeta del vehículo A: $100 - (40 + 40) = \text{S/. 20}$

El conductor desea abastecer nuevamente al vehículo A con una compra de S/. 40 pero solo cuenta con un saldo disponible de S/. 20. Ante ello, solicita a su empresa (cliente X) una recarga de S/. 20. La empresa, al ver que tiene un saldo cliente disponible de S/. 400, desea abonarle todo este saldo. Sin embargo, cuando se comunica con un asesor comercial para que realicen el abono, el asesor le indica que solo puede agregar un monto máximo de S/. 80 ya que el tope de la tarjeta del vehículo A es de S/. 100.

saldo disponible vehículo A (20) + recarga máxima (80) = saldo tope vehículo A
(100)

En caso el saldo cliente sea insuficiente, el cliente podrá agregar el monto que desee solicitando al asesor comercial la generación de una nota de crédito.

La tarjeta flota es una tarjeta física de banda magnética que tiene, entre otros, información de la placa del vehículo.

- d. **Entregar tarjeta:** la ECC imprime las tarjetas (es un proceso que puede demorar dos semanas aproximadamente) y se los entrega al cliente.
- e. **Distribuir tarjetas:** El cliente puede distribuir las tarjetas entre los conductores de su flota o que estas sean repartidas y utilizadas en las EES donde acudirá su flota.

Existe dos tipos de tope que se aplica al saldo de las tarjetas:

- Reinicial, donde el saldo se reinicia a su saldo tope cada 30 días.

- Acumulable, donde el saldo se acumula pasado los 30 días, no se reinicia.

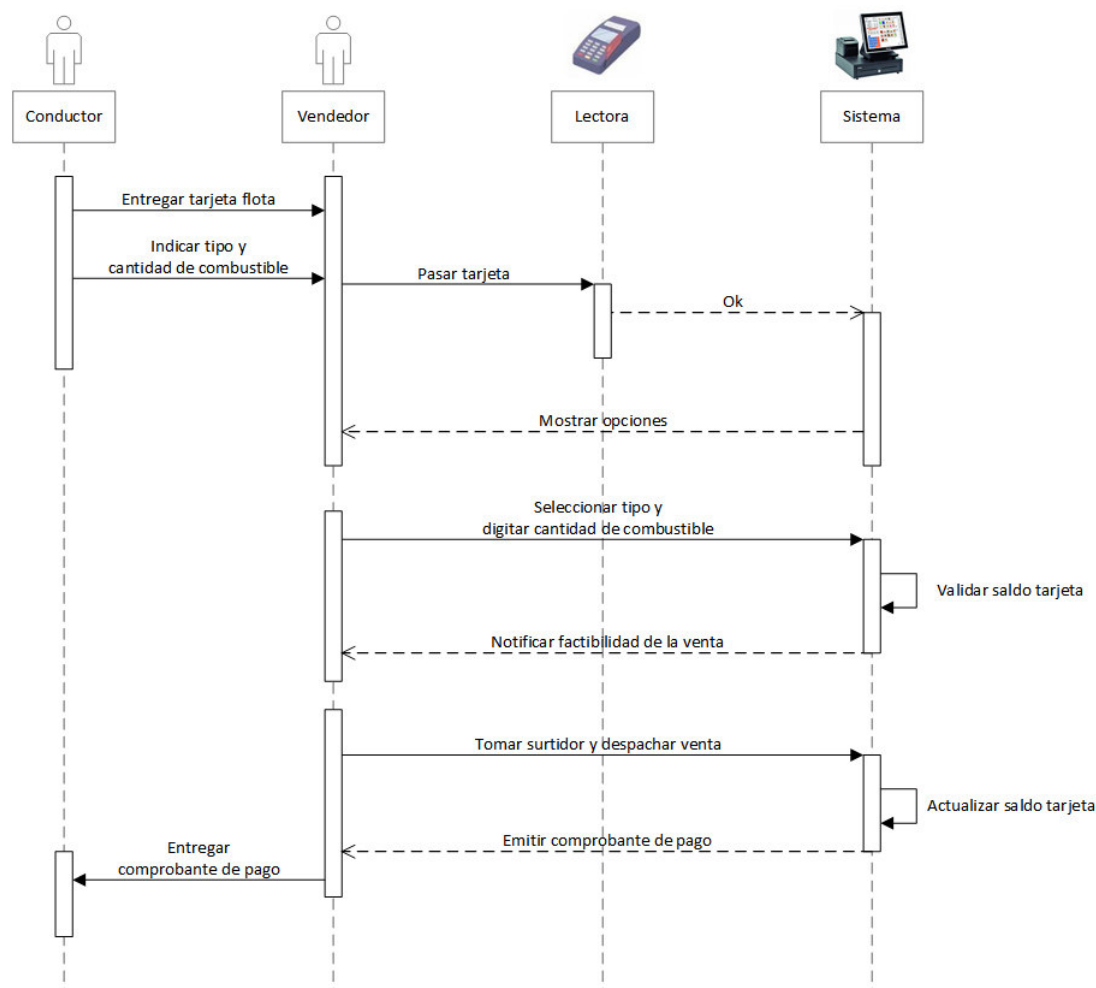
Hubo algunos clientes que parte del sueldo que les pagaba a sus choferes lo hacían con combustible, en esos casos el tipo de tope debería ser acumulable, caso contrario se perdería el dinero.

f. **Realizar compra de combustible:** Los conductores se acercaban a las EES para abastecerse de combustible de la siguiente manera:

1. El conductor entrega su tarjeta flota al vendedor o este busca la tarjeta con la placa del vehículo en un estuche que maneja por cliente.
2. El conductor indica el tipo y la cantidad de combustible a abastecerse.
3. El vendedor verifica que la placa del vehículo sea la misma que la placa impresa en la tarjeta.
4. El vendedor pasa la tarjeta por la lectora, digita el tipo y la cantidad de combustible solicitado.
5. El sistema valida que la tarjeta cuente con el saldo suficiente.
6. El sistema notifica al vendedor la factibilidad de la venta.
7. El vendedor toma el surtidor y despacha la cantidad de combustible solicitado.
8. El sistema corta el suministro de combustible cuando se llega a la cantidad solicitada.
9. El sistema actualiza el saldo de la tarjeta y emite el comprobante de pago.
10. El vendedor entrega el comprobante al conductor.

En la figura 14 se grafica los puntos mencionados anteriormente.

Figura 14. *Flujo de la venta de combustible para el Servicio para Flotas antes de la implantación del proyecto*



Fuente: Elaboración propia

- g. **Solicitar recarga de saldo:** el cliente solicitaba la recarga de saldo para sus tarjetas flota a los asesores.
- h. **Solicitar reporte de consumo diario:** el cliente solicitaba el envío de reportes de consumo diario de su flota a los asesores.
- i. **Enviar reporte:** El asesor comercial envía los reportes de consumos diarios de las flotas a los clientes al final del día.
- j. **Revisar reporte:** El cliente revisaba los reportes y veía consumos de combustible inusuales, lo que motivó a desarrollar formas de controlar dichos consumos (por ejemplo, la instalación de un medidor en el taque).

De acuerdo a la situación actual del Servicio para Flotas y con el fin de

mejorarla, la unidad de Ventas Directas solicitó a la Gerencia Comercial la posibilidad de mejorar dicho servicio en el sentido de permitirle al cliente controlar los consumos de combustible de su flota. La Gerencia Comercial propuso habilitar una solución tecnológica en donde los asesores y clientes crearían reglas que restringieran el consumo injustificado de combustible la flota.

Con ello, empezaron las primeras reuniones de la Gerencia Comercial con las unidades Operaciones Marketing y Ventas Directas para definir las funcionalidades que tendría la solución.

La Gerencia Comercial solicitó apoyo a la unidad Sistemas de Información para que se encargue del proyecto. Dicha unidad solo se encargaba de dar mantenimiento a la operatividad de las aplicaciones que tenían a cargo, mas no se dedicaban a la implementación de nuevas funcionalidades, por lo esta unidad coordinó reuniones con la ECM para la definir el proyecto, la formación del equipo y la planificación del lanzamiento.

II.1. Definición del proyecto

De acuerdo a la información obtenida de la situación actual del Servicio para Flotas, se realizó la definición del proyecto, donde se detalló el alcance de la solución, los integrantes y sus roles, un cronograma general de las actividades a realizar y la participación de la ECM.

II.2. Formación y reconfiguración del equipo de proyecto

El equipo encargado del proyecto fue la célula 1, donde los primeros 4 meses estuvo conformado por los integrantes que se visualiza en la tabla 6.

Tabla 6. *Equipo inicial de la célula 1*

Nombre	Rol	Perfil	Empresa
Gisela	Product Owner	Jefe Operaciones Marketing	ECC
Giovanni	Scrum master	Gestor de proyecto	ECM
Jorge	Scrum Team	Analista programador full-stack	ECM
Juan	Scrum Team	Analista programador full-stack	ECM
Raúl	Scrum Team	Programador backend	ECM
Fred (autor)	Scrum Team	Programador backend	ECM
David	Scrum Team	Programador full-stack	ECM
Igor	Scrum Team	Programador full-stack	ECM
Viviana	Scrum Team	Analista de calidad	ECM

Fuente: Elaboración propia

Después de los 4 meses, el equipo pasó a reconfigurarse por los integrantes que se muestra en la tabla 7.

Tabla 7. *Equipo de la célula 1 luego de la reconfiguración*

Nombre	Rol	Perfil	Empresa
Gisela	Product Owner	Jefe Operaciones Marketing	ECC
Celia	Scrum master	Gestor de proyecto	ECM
Raúl	Scrum Team	Analista programador backend	ECM
Fred (autor)	Scrum Team	Programador backend	ECM
Igor	Scrum Team	Programador backend	ECM
Martha	Scrum Team	Diseñador	ECM
Billy	Scrum Team	Maquetador	ECM
Álvaro	Scrum Team	Programador frontend	ECM
Ana	Scrum Team	Programador frontend	ECM
Omar	Scrum Team	Analista de Calidad	ECM
Alessandro	Scrum Team	Asistente de Calidad	ECM

Fuente: Elaboración propia

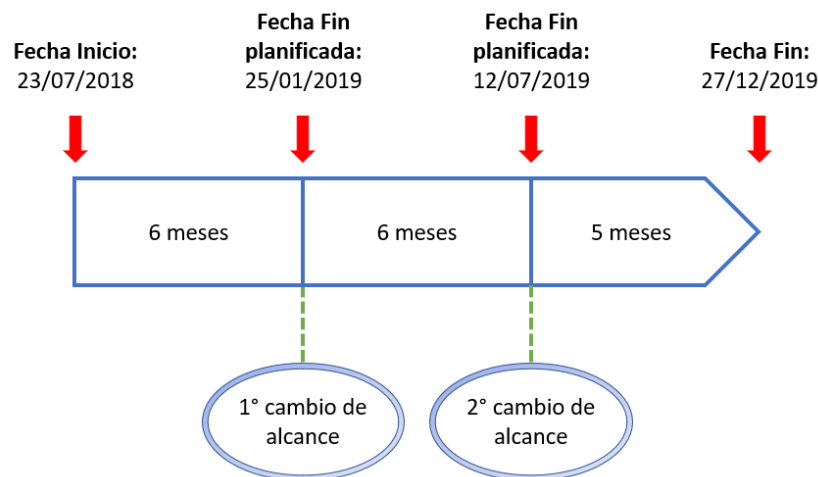
II.3. Planificación del lanzamiento

El proyecto implementación de una solución tecnológica inició el 23 de julio del 2018 con una planificación inicial de 6 meses, sin embargo, se actualizó en dos oportunidades la fecha de finalización debido a la decisión del product owner de

- terminar con los pendientes que aún había en el backlog del producto y los
- nuevos requerimientos que se iban agregando al backlog de acuerdo al feedback que recibía el product owner de parte de los clientes cuando estos utilizaban la Plataforma

culminando finalmente el 27 de diciembre del 2019, tal como se aprecia en la figura 15.

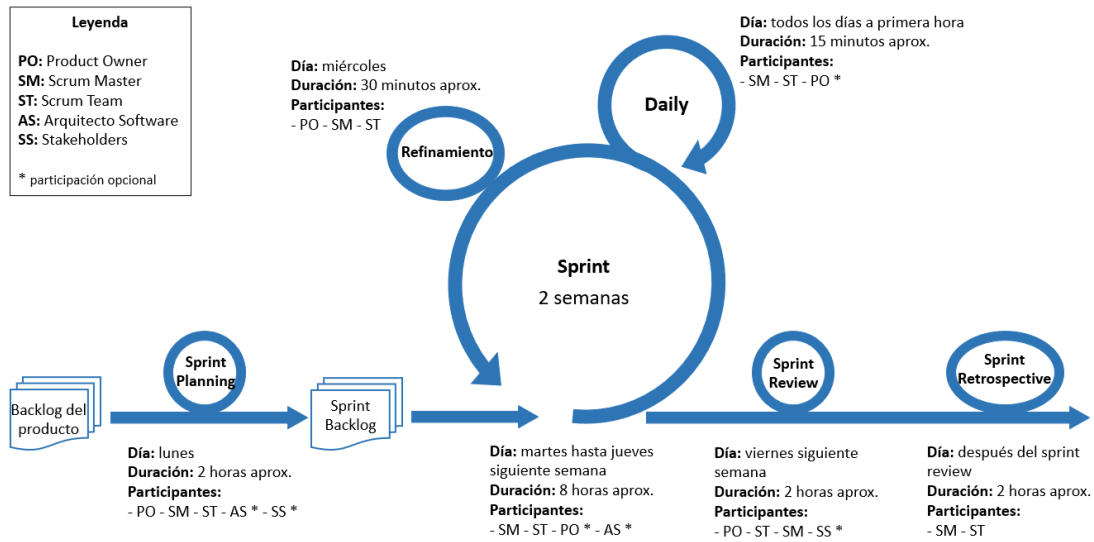
Figura 15. *Cambios de alcance realizados en el proyecto*



Fuente: Elaboración propia

Respecto a los sprints, tuvieron la estructura que se muestra en la siguiente figura.

Figura 16. Especificaciones de la estructura del sprint en el proyecto



Fuente: Elaboración propia

Los pases a producción se realizaban en base a los releases que contenían uno o más sprints. En el proyecto se desplegaron 10 releases que sumaron en total 35 sprints, tal como se aprecia en la tabla 8.

Tabla 8. *Listado de releases desplegados en el proyecto*

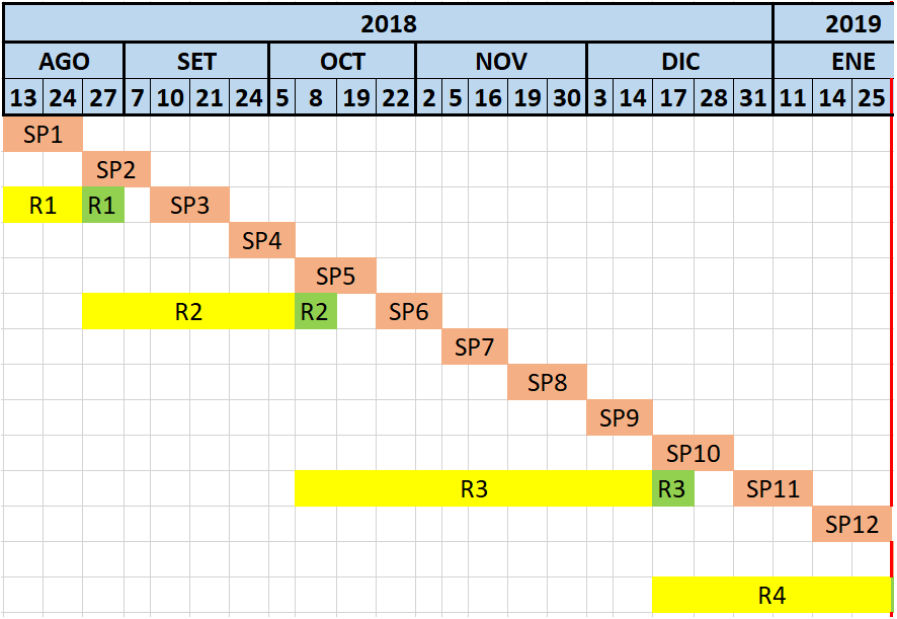
# Release	Intervalo de sprints	Cantidad de sprints	Rango de fecha
Release 1	1° Sprint	1 sprint	13/08/18 - 24/08/18
Release 2	2° Sprint - 4° Sprint	3 sprints	27/08/18 - 05/10/18
Release 3	5° Sprint - 9° Sprint	5 sprints	08/10/18 - 14/12/18
Release 4	10° Sprint - 12° Sprint	3 sprints	17/12/18 - 25/01/19
Release 5	13° Sprint - 17° Sprint	5 sprints	28/01/19 - 05/04/19
Release 6	18° Sprint - 21° Sprint	4 sprints	08/04/19 - 31/05/19
Release 7	22° Sprint - 24° Sprint	3 sprints	03/06/19 - 12/07/19
Release 8	25° Sprint - 28° Sprint	4 sprints	15/07/19 - 06/09/19
Release 9	29° Sprint - 31° Sprint	3 sprints	09/09/19 - 18/10/19
Release 10	32° Sprint - 35° Sprint	4 sprints	21/10/19 - 13/12/19

Fuente: Elaboración propia

En cada pase a producción se incrementaba la funcionalidad del producto, ya sea de la Plataforma, BackOffice, Servicio de Comunicación o el Punto de Venta; con ello se agregaba valor al manejo del control del consumo de combustible de su flota.

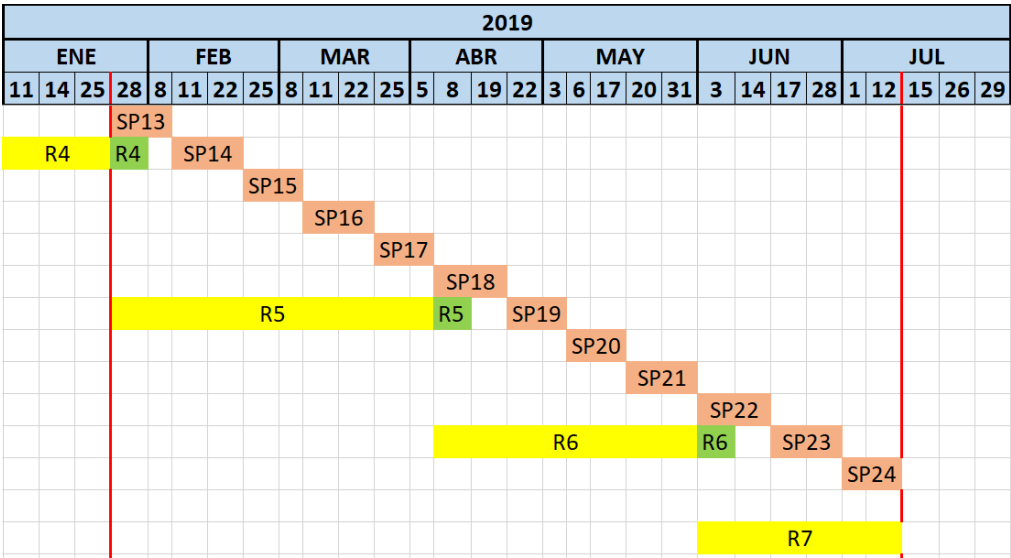
A continuación, se muestra tres figuras que corresponden al cronograma global de la implementación del proyecto junto con los dos cambios de alcance realizados durante su ejecución, además se muestra la distribución de los sprints (naranja), releases (amarillo) y los pases a producción (verde).

Figura 17. Cronograma global de la implementación del proyecto hasta el primer cambio de alcance



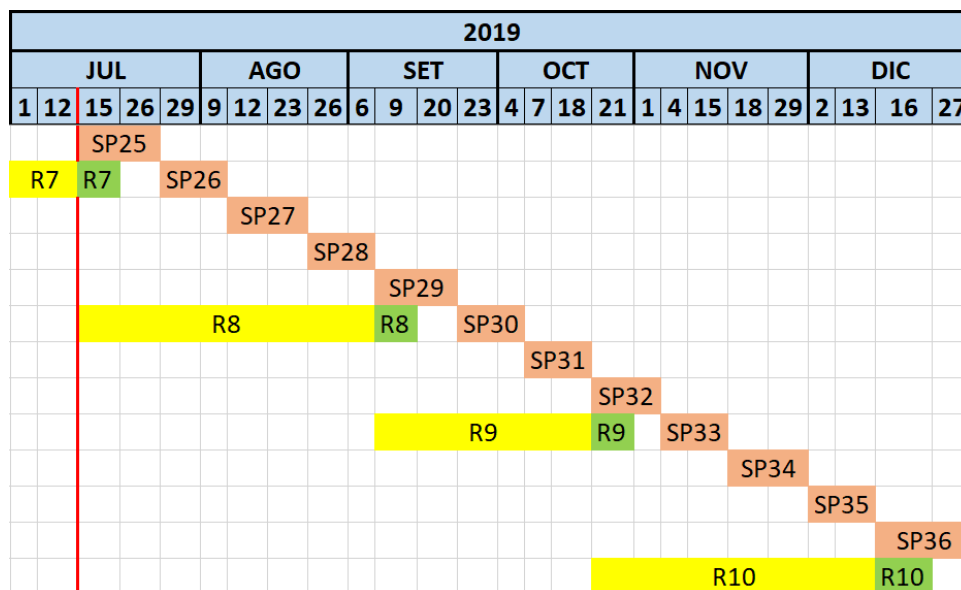
Fuente: Plan de cronograma del proyecto

Figura 18. Cronograma global de la implementación del proyecto desde el primer cambio de alcance hasta el segundo cambio de alcance



Fuente: Plan de cronograma del proyecto

Figura 19. *Cronograma global de la implementación del proyecto desde el segundo cambio de alcance hasta la finalización del proyecto*



Fuente: Plan de cronograma del proyecto

Los pases a producción se realizaban en plena ejecución del sprint (en verde), como se aprecia en las figuras 17, 18 y 19; ocasionando que el rendimiento del scrum team en dichos sprints fueran menores en comparación a otros sprints donde no hubo pase.

En el último sprint n° 36, que fue del 16 al 27 de diciembre del 2019, se realizó el último pase a producción (release n° 10). Luego, en la segunda semana del sprint, se realizaron actividades de soporte para toda la solución, tanto la atención de incidencias como la capacitación al equipo de soporte de la ECC.

Las siguientes fases corresponden a la estructura del sprint que se menciona en la figura 8, donde se explicará, de forma general, las tareas realizadas en los 36 sprints.

III. Fase de planificación y estimación

O también denominado Sprint Planning, esta fase tuvo las siguientes especificaciones:

- Día: lunes, cada 2 semanas
- Duración: 2 horas aproximadamente
- Participantes: product owner, scrum master, scrum team, eventualmente el arquitecto software y los stakeholders

III.1. Estimación de las historias de usuario

En este proceso el product owner priorizó las historias de usuario que se implementarían en el sprint, luego explicó y detalló cada una de estas incluyendo los criterios de aceptación. En base a ello, el scrum team discutió internamente la complejidad de las historias. Si la historia era compleja, la consideraban como una épica y procedían a dividirla en historias de usuario, luego estas se puntuaban en base a la escala 1, 2, 3, 5, 8 y 13, utilizando, en varias ocasiones, la aplicación Scrum Poker Cards.

Luego se tenía el backlog del producto con la puntuación de las historias de usuario, una vista previa se muestra en la siguiente figura.

Figura 20. *Historias de usuario detalladas junto a los puntos de historia*

Epic	Feature	Título Descriptivo	Sprint	Release	Descripción	Criterios de Aceptación	Comentarios	Story Points
Gestión de Tarjetas	Gestión de Promo Cards	Generación de códigos QR a partir de las Tarjetas Promo Card que existen en [Redacted]	Sprint 1	Release 1	Como Asesor Comercial de [Redacted], Necesito generar las códigos QR de las Promo Card (ya existentes) que yo seleccione para permitir a las flotas de nuestros Clientes utilizarlos para obtener descuentos sin necesidad de emitir tarjetas físicas (Promo Cards).	- La generación se debe realizar masivamente en [Redacted], sobre las Promo Card seleccionadas, en una lista de tarjetas asociadas a un cliente específico, dicha lista debe permitir selección múltiple, debe tener la opción [Seleccionar Todas], [Deseleccionar Todas]. (Nota: estas tarjetas ya fueron generadas en [Redacted]) - Esta generación se realizará a demanda. (No es necesario programarse en el [Redacted], el vendedor de Playa deberá elegir [Otras Opciones] y seleccionar la opción [Identificación con QR]. - Luego debe proceder a la lectura del Código QR (usando el scanner) - Después al Vendedor de Playa debe ingresar la fecha de vencimiento de la tarjeta. - Se debe indicar qué días de la semana se puede abastecer. Por defecto se presentan todos los días marcados. - Se debe indicar el rango de horas: desde - hasta en formato 24 horas:" hh:mm", por defecto se debe precargar el siguiente rango: 00:00 - 23:59. - Se debe ingresar la cantidad (número con	Tareas Identificadas: - Modificación de La Interface de tarjetas para agregar nuevos Filtros. - Modelamiento de BD (Tarjetas, Facturas). - Modificar Procedimiento Almacenado de Tarjetas. - Agregar Selección Múltiple de Tarjetas. - Generación de imágenes QR.	8
Gestión de Tarjetas	Gestión de Promo Cards	Reconocimiento de Código QR	Sprint 1	Release 1	Como Cliente de [Redacted], Quiero usar el código QR de mi Promo Card para obtener descuentos al comprar en una estación de servicio [Redacted]	- Después al Vendedor de Playa debe ingresar la fecha de vencimiento de la tarjeta. - Se debe indicar qué días de la semana se puede abastecer. Por defecto se presentan todos los días marcados. - Se debe indicar el rango de horas: desde - hasta en formato 24 horas:" hh:mm", por defecto se debe precargar el siguiente rango: 00:00 - 23:59. - Se debe ingresar la cantidad (número con	Tareas Identificadas: - Agregar nueva opción en [Redacted] [Identificación con QR] - Añadir lógica de activar y desactivar botón - Crear Procedimiento Almacenado para obtener información de la Estación de Servicio. - Definir la cantidad máxima, el saldo no se acumula. El usuario Logueado, debe ingresar a la pantalla de [Reglas Comerciales] que se encuentra dentro de [Clientes], desde la pantalla [Reglas Comerciales], se podrán crear las [Reglas de Frecuencia]	20
Configuración	Gestión de reglas comerciales	Crear Reglas por Frecuencia-BackOffice	Sprint 2	Release 2	Como Asesor Comercial de [Redacted], Quiero crear Reglas de Frecuencia aplicadas para un Cliente específico, Para aplicar restricciones en los abastecimientos por Tarjeta (a las tarjetas asociadas a dicho Cliente).	- Después al Vendedor de Playa debe ingresar la fecha de vencimiento de la tarjeta. - Se debe indicar qué días de la semana se puede abastecer. Por defecto se presentan todos los días marcados. - Se debe indicar el rango de horas: desde - hasta en formato 24 horas:" hh:mm", por defecto se debe precargar el siguiente rango: 00:00 - 23:59. - Se debe ingresar la cantidad (número con	Tareas Identificadas: - Agregar nueva opción en [Redacted] [Identificación con QR] - Añadir lógica de activar y desactivar botón - Crear Procedimiento Almacenado para obtener información de la Estación de Servicio. - Definir la cantidad máxima, el saldo no se acumula. El usuario Logueado, debe ingresar a la pantalla de [Reglas Comerciales] que se encuentra dentro de [Clientes], desde la pantalla [Reglas Comerciales], se podrán crear las [Reglas de Frecuencia]	13

Fuente: Obtenido del proyecto

En los primeros sprints se realizaban estimaciones que no eran comprometidas al 100%, dado que 4 de 7 integrantes del scrum team recién empezaban a conocer el negocio y las tecnologías que se manejaban en la unidad de Ventas Directas de la ECC. Sin embargo, con el apoyo de los otros 3 integrantes, la curva de aprendizaje del equipo mejoró y en consecuencia aumentó el número de historias de usuario implementadas en posteriores sprints.

III.2. Creación del sprint backlog

A partir del backlog del producto puntuado, el producto owner seleccionaba las historias de usuario que mejor le convenían para su implementación en el sprint, dando origen al sprint backlog. A continuación, en la figura 21 se muestra una vista preliminar del sprint backlog del sprint 21.

Figura 21. *Sprint backlog del sprint 21*

Epic	Feature	Usuario	Entorno	Título Descriptivo	Sprint	Release	Descripción	Como debería estar escrito: Como Quiero Para	Criterios de Aceptación
Conductores	Gestión de conductores	Cliente	Portal	Modificar Conductores	21	6	Como gestor de flota, necesito editar/modificar los datos relacionados al conductor, para corregir la información registrada.	Como gestor de flota, necesito editar los datos de un conductor, para corregir la información registrada.	Todos los campos son editables desde la ficha del conductor. Luego de realizar la modificación el sistema deberá de solicitar la confirmación de los cambios y guardar la información del usuario que los realizó y fecha de modificación (auditoría)
Configuración	Pines	Cliente	Portal	Gestión de PINES (Generación)	21	6	Como gestor de flota, necesito generar PINES de seguridad para autorizar los despachos e identificar al conductor que abasteció.	Como gestor de flota, necesito generar un PIN de seguridad a cada conductor, para autorizar los despachos e identificar al conductor al momento de abastecer combustible.	Los PINES deben ser autogenerados por el sistema al momento del registro o creación del conductor.
Configuración	Pines	Cliente	Portal	Envío Automático de PIN	21	6	Como gestor de flota necesito que al momento de la generación del PIN, el sistema envíe de forma automática un correo con el PIN del conductor al correo predeterminado en la ficha		Una vez generado el PIN, el sistema debe de emitir un correo automático dirigido al gestor de la flota y al conductor en caso se haya registrado su email, con el código generado.
Vehículos	Gestión de Vehículos	Cliente	Portal	Registrar vehículos	21	6	Como gestor de flotas necesito crear un vehículo para darle de alta en la lista de vehículos de mi flota y esté autorizado para el abastecimiento	Como gestor de flotas necesito dar de alta los vehículos de mi flota para poder autorizar su abastecimiento de combustible	La ficha debe contar con los campos: placa, marca, tipo de vehículo, modelo y año de fabricación, capacidad de tanque, rendimiento por galón según fabrica, tipo de combustible. Debe existir un control de campos obligatorios: Placa. Debe controlar que la placa tenga el formato requerido (alfanumérico, y 6 dígitos o 7 dígitos) El sistema no deberá permitir signos como guiones y cambiar todo el texto ingresado a mayúsculas. Al crearse, el vehículo se creará por defecto en estado (activo)

Resumen

Backlog Pendiente

Sprint 21

Sprint 22

Sprint 23

Fuente: Obtenido del proyecto

Una vez culminados los procesos de esta fase, el scrum team procedía con el diseño técnico preliminar de las historias de usuario.

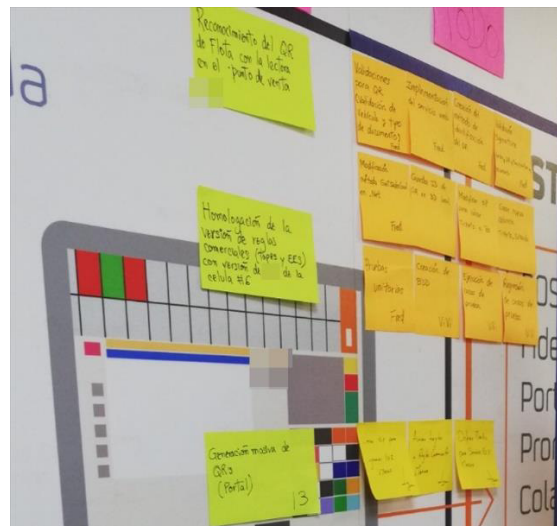
Diseño técnico preliminar de las historias de usuario

En esta actividad el scrum team realizaba un diseño técnico inicial que involucraría el desarrollo de las historias del sprint donde, eventualmente, participaba el arquitecto software. Luego, en consenso, se identificaba y asignaba las tareas a cada integrante del scrum team y estos estimaban una fecha de finalización.

Las tareas y sus responsables se escribían en unos post-its que eran colocados en un tablero kanban, donde el scrum master tenía un panorama del avance. Cuando algún integrante del equipo culminaba con alguna etapa de la tarea, el integrante

desplazaba el post-it de dicha tarea a lo largo del tablero. Un ejemplo del tablero que se manejó en el equipo aparece en la figura 22.

Figura 22. Vista preliminar del tablero con las tareas del equipo



Fuente: Obtenido del proyecto

La implementación del proyecto involucró la revisión de tecnologías de base de datos, backend y frontend, utilizando los lenguajes de programación sql, c# y angular, respectivamente.

IV. Fase de implementación

O también denominado el desarrollo del Sprint, fue una fase donde el scrum team realizó la codificación de las historias de usuario en base al sprint backlog. Si existía alguna duda funcional respecto a las historias, estas se aclaraban mediante una llamada telefónica al product owner o stakeholders. Esta fase tuvo las siguientes especificaciones:

- Día: desde el martes de la primera semana hasta el jueves de la segunda semana del sprint.
- Duración: 8 horas laborales al día.
- Participantes: scrum master, scrum team y, eventualmente, el arquitecto software y el product owner.

- Ceremonias:
 - Daily: se realizaba todos los días del sprint a primera hora, tuvo una duración de 15 minutos aproximadamente.
 - Refinamiento: los miércoles de la primera semana del sprint, tuvo una duración de 30 minutos aproximadamente.

Al inicio del día laboral, el equipo por el lado de la ECM se reunía en las dailys para indicar el avance de sus actividades. Luego, cada uno iba avanzando sus tareas a lo largo del sprint.

IV.1. Refinamiento del backlog

Hubo reuniones donde el equipo del lado de la ECM se reunía con el product owner para detallar las historias de usuario que se considerarían en el sprint backlog del siguiente sprint. Una vista preliminar del documento que contenía las historias refinadas se visualiza en la figura 23.

Figura 23. *Backlog del producto con las historias de usuario refinadas*

Epic	Feature	Título Descriptivo	Sprint	Release	Descripción	Criterios de Aceptación	Comentarios del Refinamiento
Conductores	Gestión de conductores	Modificar Conductores	21	6	Como gestor de flota, necesito editar/modificar los datos relacionados al conductor, para corregir la información registrada.	Todos los campos son editables desde la ficha del conductor. Luego de realizar la modificación el sistema deberá de solicitar la confirmación de los cambios y guardar la información del usuario que los realizó y fecha de modificación (auditoría)	El DNI no es editable por ser una llave. El PIN debe ser editable, solo números, 4 dígitos. MENSAJE: Se ha creado correctamente el conductor y se envió el PIN a los siguientes correos. Se debe guardar la información del usuario que los realizó y fecha de modificación (auditoría).
Configuración	Pines	Gestión de PINES (Generación)	21	6	Como gestor de flota, necesito generar PINES de seguridad para autorizar los despachos e identificar al conductor que	Los PINES deben ser autogenerados por el sistema al momento del registro o creación del conductor.	El PIN es un campo de 4 dígitos en la ficha conductor. Se mostrará bloqueado y vacío.
Configuración	Pines	Envío Automático de PIN	21	6	Como gestor de flota necesito que al momento de la generación del PIN, el sistema envíe de forma automática un correo con el PIN del conductor al correo predeterminado en la ficha	Una vez generado el PIN, el sistema debe de emitir un correo automático dirigido al gestor de la flota y al conductor en caso se haya registrado su email, con el código generado.	PENDIENTE DEL PO: Formato de correo. Se debe crear correctamente el conductor para posteriormente se envíe el correo al gestor y/o conductor. Cambiar la confirmación de creación de conductor. MENSAJE: Se ha creado correctamente el conductor y se envió el PIN a los siguientes correos.

Fuente: Extraído del proyecto

Respecto a la implementación del proyecto, se consideró las siguientes actividades (requisitos no funcionariales):

- Solicitud de accesos para el equipo por el lado de la ECM (las herramientas eran de la ECM a excepción del usuario de la ECC):
 - Confluence

- GitLab
 - Creación de usuario externo en la ECC
 - Grupo de trabajo en el Microsoft Teams
 - Jenkins
 - Sonar
- Revisión de la documentación de los lineamientos de desarrollo para base de datos y backend (alojados en el repositorio del Confluence).

A continuación, se listan los documentos de los lineamientos de desarrollo que manejaba la ECM:

- TC1_Estandares de desarrollo en SQL Server T-SQL.pdf
 - TC2_Estandares de desarrollo en C# 6.pdf
 - TC3_Manual de optimizacion de consultas SQL.pdf
 - Industrialización - Pruebas unitarias - Servicios - BD .Net.pdf
 - Lineamientos para Pases de Componente y scripts de base de datos V1.4.pdf
- Creación de un esquema para la base de datos del proyecto, realizado por el DBA.

A continuación, en la siguiente imagen se listan los esquemas creados por el DBA a nivel del proyecto de modernización, donde destacamos a la célula 1 cuyo nombre fue Júpiter.

Figura 24. *Esquemas de base de datos del proyecto Modernización de EES*



Fuente: Obtenido del proyecto

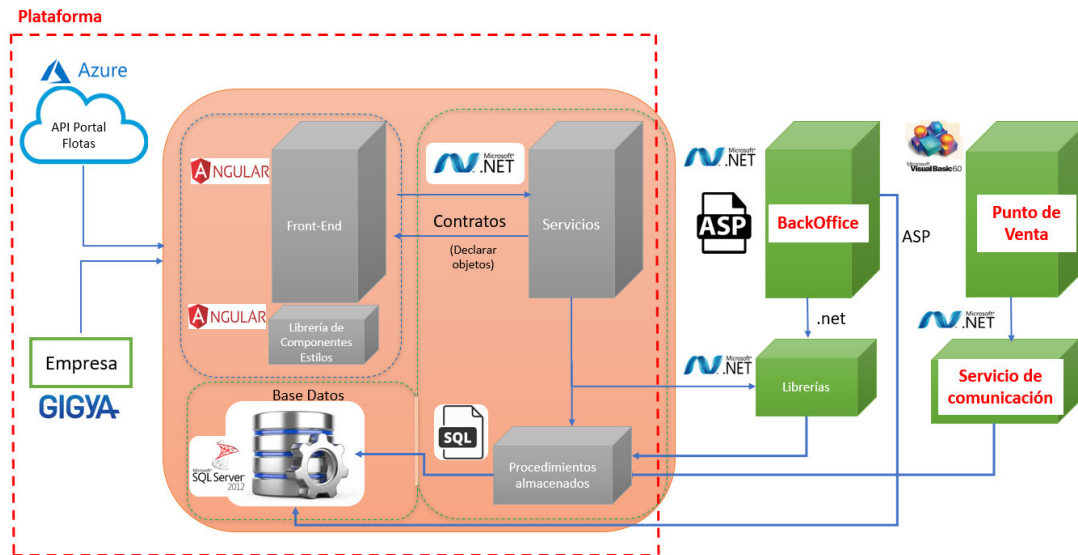
- Creación del proyecto base en c# .net para la Plataforma y el Servicio de Comunicación, y creación de módulos base en c# y visual basic .net para el BackOffice y el Punto de Venta, respectivamente, por parte del arquitecto software.
- Creación del proyecto base en angular versión 5 para la Plataforma, por parte del programador full-stack, de acuerdo al framework frontend de la ECM.
- Codificación de las historias de usuarios de la solución: Plataforma, Servicio de Comunicación, BackOffice y Punto de Venta.
- Despliegue de la solución en el entorno de desarrollo.
- Despliegue de la solución en el entorno preproductivo en coordinación con el equipo Release&Testing.
- Validación de la solución en el entorno preproductivo.
- Registro de componentes creados / modificados en la documentación del release.

Como se mencionó al inicio de esta sección, el autor se enfocó más en la implementación de la Plataforma con el perfil programador backend. A continuación, se expone el detalle de estos artefactos.

Arquitectura de la solución

La solución consistió en la creación de la Plataforma y el Servicio de Comunicación, y la modificación del BackOffice y el Punto de Venta. A continuación, se muestra la arquitectura general de la solución, donde se aprecia la comunicación entre los 4 artefactos que componen a la solución.

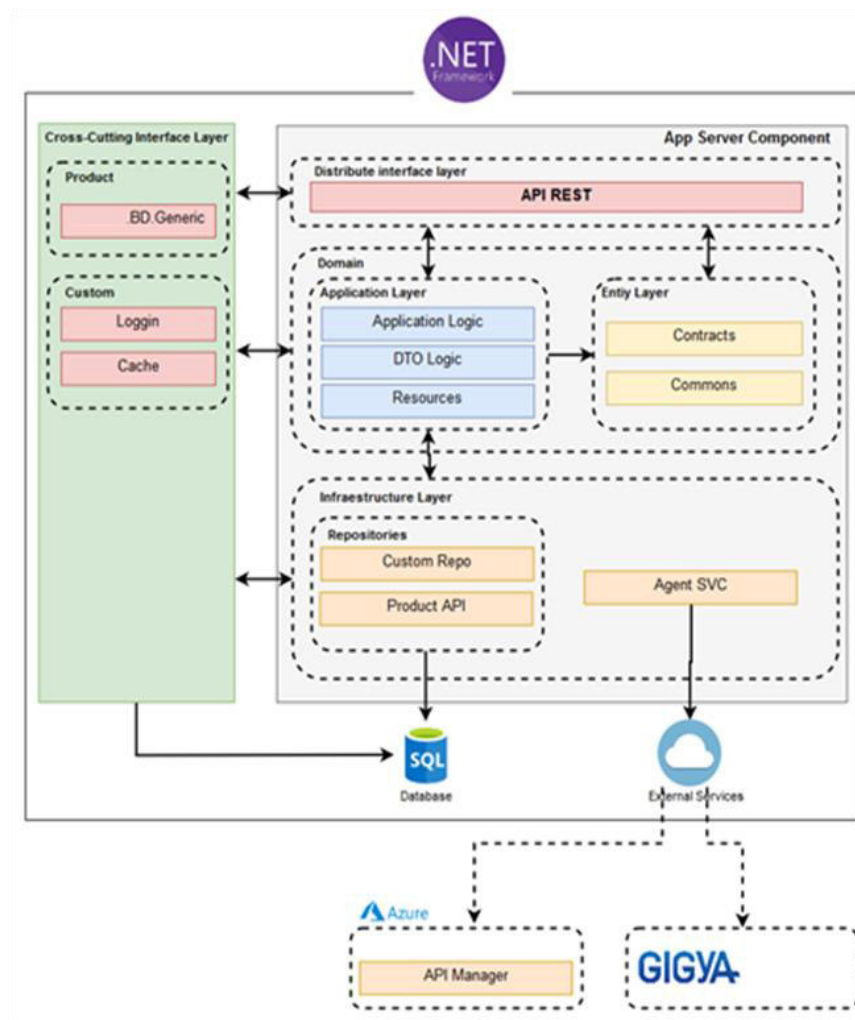
Figura 25. *Arquitectura general de la solución*



Fuente: Extraído del documento técnico ágil del proyecto.

Respecto al frontend, se reutilizó un framework de la ECM que estaba construido sobre angular versión 5 y fue personalizado para la Plataforma. Respecto al backend, por el lado de la Plataforma y el Servicio de Comunicación, estos consistieron en un servicio API REST construido en **c# .net** bajo la arquitectura **DDD** (Domain Driven Design) que consultaba una base de datos relacional **SQL Server**, tal y como se muestra en la figura 26.

Figura 26. Arquitectura del servicio API REST de la Plataforma y el Servicio de Comunicación



Fuente: Extraído del documento técnico ágil del proyecto.

Para la construcción de objetos de base de datos y backend se consideraron las siguientes tecnologías y framework:

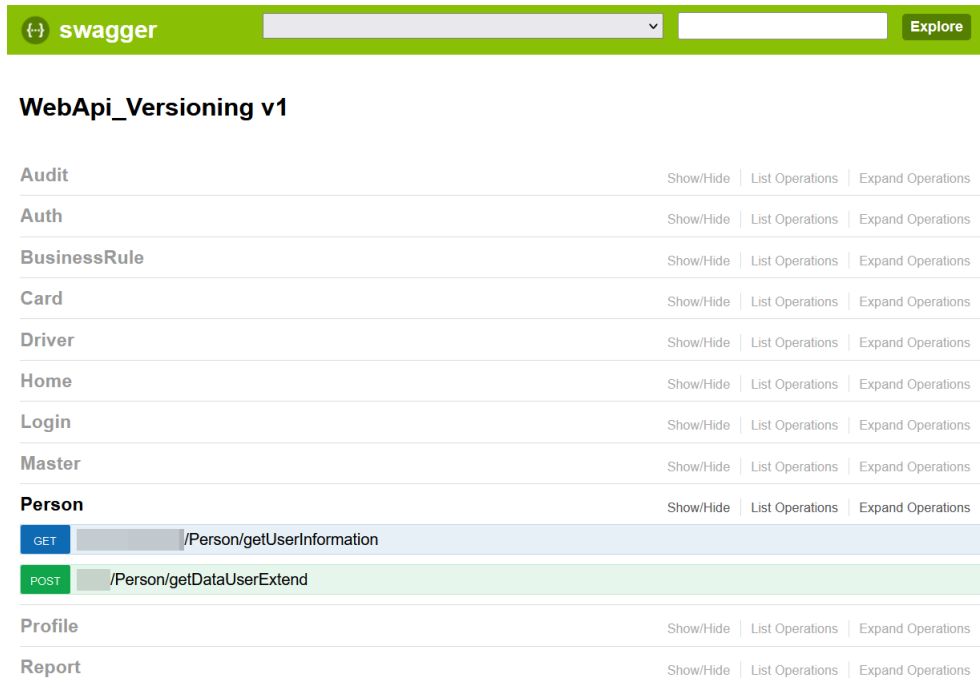
- Sql Server 2014
- Net Framework 4.5+
- Formato Rest (Json)
- Structure Map (Inyeccion de dependencia)
- Gigya connector

La estructura de la API estuvo conformada por los siguientes packages:

- API
 - Módulo Controllers: contuvo todas las clases que fueron expuestos a la Plataforma.
 - Módulo Handler: se almacenaron todos los interceptores de la aplicación.
 - Módulo Models: contenía las clases DTO de exposición de servicio.
 - Módulo DependencyResolution: se encargó de abstraer todas las interfaces. Implementó el patrón singleton, se basó en structure mapper.
- Domain:
 - Módulo Entity: capa de entidades de la aplicación, donde se generó los contratos que fueron expuestos hacia el frontend y también se generó las entidades de negocio o clases comunes que usó la Plataforma.
 - Módulo Logic
 - ❖ Logic: contuvo toda la codificación de las reglas, validaciones del servicio.
 - ❖ Repository: contuvo todos los métodos que accedieron a la base de datos.
 - ❖ Resources: contuvo todos los mensajes de validación del servicio.
 - ❖ Translator: contuvo la funcionalidad de mapear los campos del servicio con los campos externos del servicio.
- Infraestructure
 - Módulo Data: contuvo las clases que fueron contra la persistencia de datos (crear, actualizar, eliminar y listar).
 - Módulo ServicesAgent
 - ❖ ServicesAgent: se construyó todas las clases que se comunicaron con los servicios externos (como Gigya y servicios del BackOffice).

- ❖ DTO: contuvo los contratos utilizados por la capa ServicesAgent.
- Test: contuvo todas las pruebas unitarias por cada método que se implementó. Se utilizó Swagger UI para la documentación de la API.

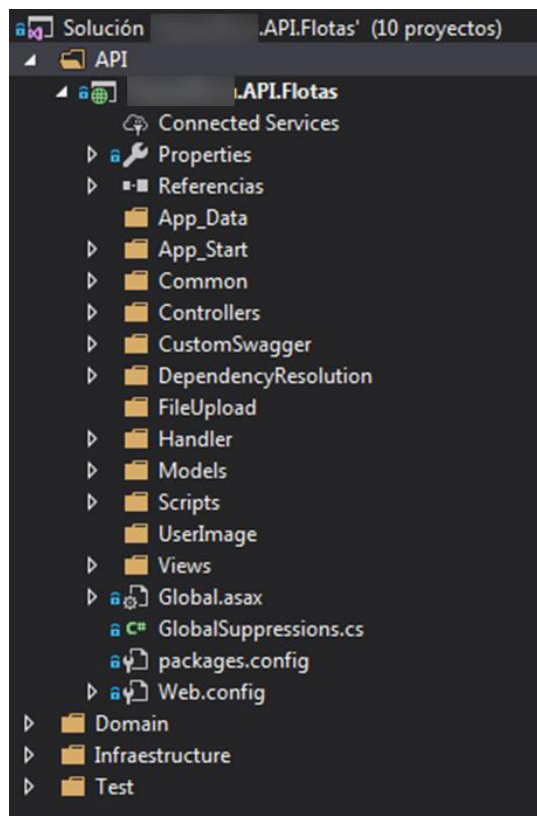
Figura 27. *Swagger como documentación del API REST de la Plataforma*



Fuente: Extraído del documento técnico ágil del proyecto.

A continuación, en la siguiente figura se visualiza los packages de la Plataforma creados en Visual Studio.

Figura 28. Estructura del proyecto de la Plataforma en Visual Studio



Fuente: Extraído del documento técnico ágil del proyecto.

IV.2. Funcionalidades de la Plataforma

Entre las funcionalidades que se desarrollaron para la Plataforma figuran los siguientes (el detalle de los módulos/pantallas de la Plataforma se visualiza en el Anexo 1):

1. **Dashboard:** mostraba información del histórico de consumos del cliente en un año y una comparación con el año anterior.
2. **Seguridad (Usuarios y Perfiles):** gestión de accesos para el asesor, cliente y los conductores.
3. **Administración:** gestión de registros de vehículos, conductores, centros de costo y reglas comerciales.
4. **Tarjetas:** gestión de registros de tarjetas, gestión de topes y saldos, y ver listado de programaciones de topes y saldos.
5. **Comercial:** gestión de reporte de consumo, reporte de saldo por tarjeta y reporte

de estado de cuenta.

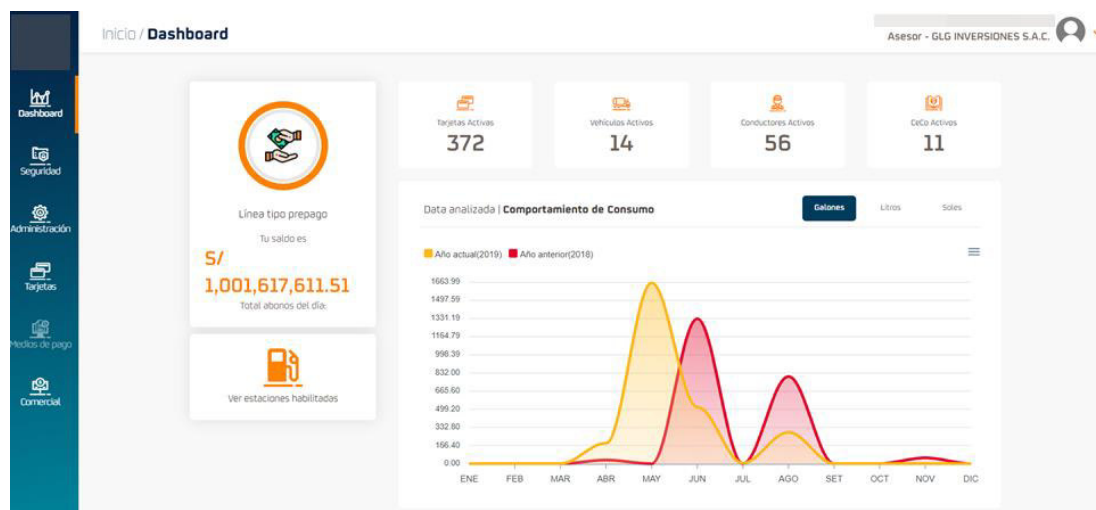
A continuación, se visualiza la primera vista de la Plataforma, tanto para el perfil Administrador como el perfil Asesor Comercial, donde este último puede ver la información de los clientes prepago y crédito, es decir, del perfil Gestor Flota.

Figura 29. Vista inicial de la Plataforma para el perfil Administrador



Fuente: Extraído del documento técnico ágil del proyecto.

Figura 30. Vista inicial de la Plataforma para el perfil Asesor donde visita la información de un cliente prepago



Fuente: Extraído del documento técnico ágil del proyecto.

Figura 31. Vista inicial de la Plataforma para el perfil Asesor donde visita la información de un cliente crédito

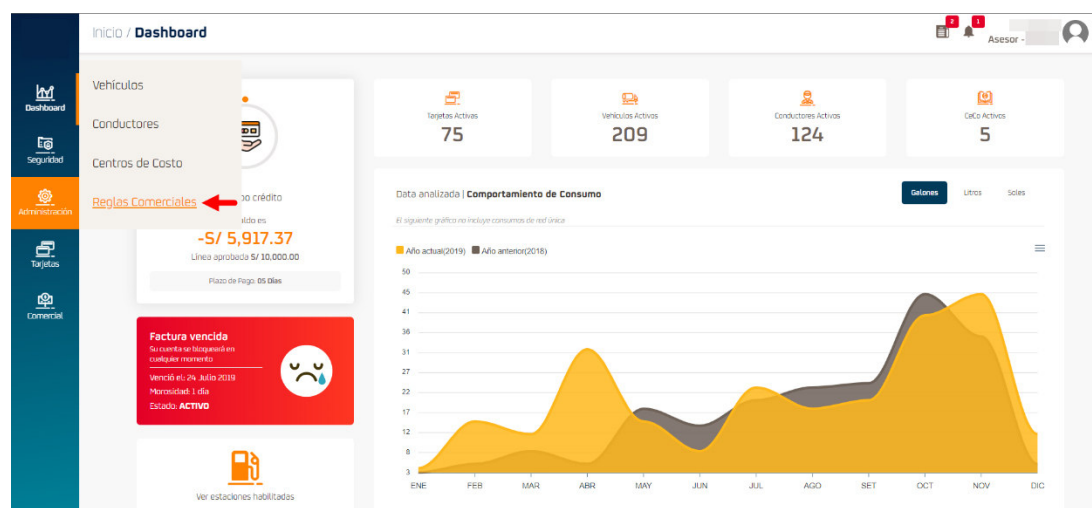


Fuente: Extraído del documento técnico ágil del proyecto.

A continuación, mencionaremos el detalle de algunas funcionalidades:

- **Reglas comerciales:** conjunto de controles configurables por el asesor comercial o el cliente corporativo a fin de restringir el consumo de combustible de la flota. Para ingresar a esta funcionalidad debemos ingresar a la opción Administración y luego Reglas Comerciales, tal como se indica en la figura 32. En la figura 33 se visualiza la pantalla de inicio de dicha opción, luego en la figura 34 se muestra la pantalla de creación de una Regla Comercial.

Figura 32. Opción Reglas Comerciales en la Plataforma



Fuente: Extraído del documento técnico ágil del proyecto.

Figura 33. Pantalla inicio de la opción Reglas Comerciales

Administración / **Reglas Comerciales**

Tienes **699** reglas registradas

218 Reglas Activas / 481 Reglas Inactivas

Nombre de la Regla: Seleccione Fecha de creación: Estado: **Buscar** **Limpiar**

10/493 Registros Encontrados **Primera** 1 2 3 Última

Crear regla

☐ ☐	Código	Nombre Regla	F. Creación	F. Modificación	Creado por	Estado
☐	535	Reglas UNITARIAS0002	30/11/2018		02934eiversp04	Activo
☐	534	Reglas JC30	30/11/2018		02934eiversp04	Activo
☐	533	prueba portal1	26/11/2018		02934PortalPeruFlota	Activo
☐	532	prueba portal1	26/11/2018		02934PortalPeruFlota	Activo
☐	531	prueba portal1	26/11/2018		02934PortalPeruFlota	Activo
☐	530	prueba portal1	26/11/2018		02934PortalPeruFlota	Activo
☐	529	prueba portal1	26/11/2018	29/11/2018	02934PortalPeruFlota	Inactivo
☐	528	prueba portal1	26/11/2018		02934PortalPeruFlota	Activo

Fuente: Extraído del documento técnico ágil del proyecto.

Figura 34. Pantalla de creación de una Regla Comercial

Administración / **Reglas Comerciales**

Crear Regla

Datos Generales

Código: Nombre de la Regla:

Nro. de código: Ejemplo - Regla Gasolina 90 NO fines de semana

Descripción de la Regla:

Ejemplo - Esta regla sólo aplica para gasolina de 90 y días de semana.

Reglas de Frecuencia

Días en los que se abastece: ☐ L ☐ M ☐ M ☐ J ☐ V ☐ S ☐ D

Horas en las que se abastece: 00:00 Hasta: 23:59 Frecuencia de Consumo: 0 vez/veces al día.

Reglas de Producto

Productos que se abastece: ☐ Tipo 1 ☐ Tipo 2 ☐ Tipo 3 ☐ Tipo 4 ☐ Tipo 5

Regla de Tope por Consumo Diario

Ingrese tope por Galones: Litros sólo para GLP: Ingrese tope por Soles: S/

Fuente: Extraído del documento técnico ágil del proyecto.

Entre los controles podemos mencionar:

- Reglas por Frecuencia
 - ❖ Días: listado de días que son permitidos para el consumo de combustible. Por defecto está permitido todos los días (lunes, martes, miércoles, jueves, viernes, sábado y domingo).
 - ❖ Horas: horario en el que se puede realizar el abastecimiento de combustible. Por defecto se indica desde la 00:00 hasta las 23:59 horas.
 - ❖ Frecuencia: número de veces al día en que se puede realizar el abastecimiento de combustible. Por defecto es libre.

Figura 35. *Reglas de Frecuencia*

Reglas de Frecuencia

Días en los que se abastece:

L M M J V S D

Horas en las que se abastece:

00:00 Hasta: 23:59

Frecuencia de Consumo:

0 vez/ veces al día.

Fuente: Extraído del documento técnico ágil del proyecto.

- Reglas de Producto: lista de productos habilitados para el abastecimiento de combustible. Por defecto se indica todos los productos.

Figura 36. *Reglas de Producto*

Reglas de Producto

Productos que se abastece: (*)

Tipo 1 Tipo 2 Tipo 3 Tipo 4 Tipo 5

Fuente: Extraído del documento técnico ágil del proyecto.

- Regla de Tope por Consumo Diario (por acumulación de compras al día)
 - ❖ Tope por galones: máxima cantidad de galones permitidos en un día. Aplicado para los productos Tipo 1, Tipo 2, Tipo 3 y Tipo 4.
 - ❖ Tope de litros: máxima cantidad de litros permitidos en un día. Aplicado para el producto Tipo 5.
 - ❖ Tope por Soles: máxima cantidad de gasto permitido en un día.

Figura 37. *Regla de Tope por Consumo Diario*

Fuente: Extraído del documento técnico ágil del proyecto.

- Regla por Estación de Servicio: lista de EES donde se podrá realizar el abastecimiento de combustible. Por defecto están habilitadas todas las estaciones.

Figura 38. *Regla por Estación de Servicio*

Fuente: Extraído del documento técnico ágil del proyecto.

Cada una de estas reglas eran creadas desde la Plataforma o el BackOffice y eran validadas en el Punto de Venta a través del Servicio de Comunicación.

IV.3. Generación de entregables

Cada miembro realizaba sus actividades que podrían involucrar la creación/modificación de componentes, luego estos eran registrados en el listado de objetos del release para posteriormente realizar el pase a producción.

Los entregables de la base de datos y del backend fueron los siguientes (extraídos del Documento Tag del Release 5):

- Objetos de base de datos:
 - Script de tipo de datos
 - ❖ CREATE_TABLE_[Marketing].[tGenericField].sql
 - ❖ CREATE_TABLE_[Marketing].[tProfileDetail].sql
 - ❖ Marketing.TYPE_CARDS_ID.sql
 - ❖ TYPE_CARDTYPENEW.sql
 - Script de creación de tablas
 - ❖ Create_Table_Perfiles_Portal.sql
 - ❖ Marketing.CARD_COUNTER.sql
 - ❖ Marketing.CARD_VIRTUAL.sql
 - ❖ Marketing.Conductores.sql
 - ❖ Marketing.VEHICULO.sql
 - ❖ Marketing.CONDUCTOR_VEHICULO.sql
 - ❖ Marketing.MARCA_VEHICULO.sql
 - ❖ Marketing.TERMS_AND_CONDITIONS.sql
 - ❖ Marketing.TIPO_VEHICULO.sql
 - ❖ Marketing.TYPE_TERMS_AND_CONDITIONS.sql
 - Modificación de tablas
 - ❖ AlterTables_Usuario_Perfil_Cliente.sql
 - ❖ ALTER_TABLE_ECC_PERSONA.sql
 - ❖ ALTER_TABLE_[ECC_USUARIO_CLIENTE].sql
 - Inserción de datos
 - ❖ InsertsTemplateEmail.sql
 - ❖ Insert_Data_Tabla_Perfiles_Portal.sql

- ❖ INSERT_Marketing.CARD_COUNTER.sql
- ❖ INSERT_Marketing.MARCA_VEHICULO.sql
- ❖ INSERT_Marketing.TIPO_VEHICULO.sql
- Actualización de datos
 - ❖ TABLE_FORMAS_JURIDICAS_UPDATE_ROW.sql
 - ❖ REGULARIZACION_USUARIOS_PORTAL.sql
- Script de creación y modificación de procedimientos almacenados
 - ❖ Marketing.uspDriverById.sql
 - ❖ Marketing.uspDriverByPlate.sql
 - ❖ Marketing.uspDriverFleetSummary.sql
 - ❖ Marketing.uspDriverJoinPlate.sql
 - ❖ Marketing.uspDriverValidationDocument.sql
 - ❖ Marketing.uspEnvioDeQRMasivoPorEmail.sql
 - ❖ Marketing.uspEnvioDeQRporEmail.sql
 - ❖ Marketing.uspExpireQR.sql
 - ❖ Marketing.uspGetAllProfiles.sql
 - ❖ Marketing.uspGetDataQR.sql
 - ❖ Marketing.uspGetListActions.sql
 - ❖ Marketing.uspGetListActionsByConfiguration.sql
 - ❖ Marketing.uspGetListDriverByCustomerFilters.sql
 - ❖ Marketing.uspGetListModules.sql
 - ❖ Marketing.uspGetListModulesByProfile.sql
 - ❖ Marketing.uspGetListPages.sql
 - ❖ Marketing.uspGetListPagesByConfiguration.sql
 - ❖ Marketing.uspGetListSubModules.sql
 - ❖ Marketing.uspGetListSubModulesByConfiguration.sql
 - ❖ Marketing.uspGetListUserByCustomerFilters.sql
 - ❖ Marketing.uspGetPhysicalCardRequestStatus.sql
 - ❖ Marketing.uspGetProfileKPI.sql
 - ❖ Marketing.uspGetProfileList.sql
 - ❖ Marketing.uspGetSalesCard_flotas.sql
 - ❖ Marketing.uspGetScheduleListByCustomFilter.sql
 - ❖ Marketing.uspGetSchuleKPI.sql
 - ❖ Marketing.uspGetUserById.sql

- ❖ Marketing.uspGetVirtualCard.sql
- ❖ Marketing.uspProfileCreate.sql
- ❖ Marketing.uspProfileGetById.sql
- ❖ Marketing.uspProfileStatusUpdate.sql
- ❖ Marketing.uspProfileUpdate.sql
- ❖ Marketing.uspRollBackDriver.sql
- ❖ Marketing.uspScheduleCardKPI.sql
- ❖ Marketing.uspScheduleCardList.sql
- ❖ Marketing.uspSetCardInContactosCliExtraData.sql
- ❖ Marketing.uspSetDriver.sql
- ❖ Marketing.uspSetGenerateQR.sql
- ❖ Marketing.uspSetPhysicalCardRequest.sql
- ❖ Marketing.uspSetUser.sql
- ❖ Marketing.uspSetVehicle.sql
- ❖ Marketing.uspUpdateDrivePortal.sql
- ❖ Marketing.uspUpdateStatusDriver.sql
- ❖ Marketing.uspUpdateStatusMileage.sql
- ❖ Marketing.uspUpdateStatusUser.sql
- ❖ Marketing.uspUpdateUserPortal.sql
- ❖ Marketing.uspUserFleetSummary.sql
- ❖ Marketing.uspUserJoinProfile.sql
- ❖ Marketing.uspUserValidationEmail.sql
- ❖ Marketing.uspValidacionTarjeta.sql
- ❖ Marketing.uspValidarEmisionDeCorreoQR.sql
- ❖ Marketing.uspValidarEmisionDeCorreoQRMasivo.sql
- ❖ Marketing.uspValidateChangeStatusCards.sql
- ❖ Marketing.uspVehicleDropDownList.sql
- ❖ Marketing.uspVehiclePlate.sql
- ❖ Marketing.uspVehicleValidationPlate.sql
- ❖ ECC_FLEET_CARD_GET.sql
- ❖ ECC_GetClientByUser.sql
- ❖ ECC_GetPerfilesuser.sql
- ❖ ECC_SetContactCli.sql
- ❖ ECC_INSERT_PERFIL_GESTOR.sql

- Programación de jobs
 - ❖ JOB - ECC - Inicializador de contador de tarjetas.sql
- Objetos del backend: en el Release 5 solo se desplegaron componentes de la Plataforma
 - Plataforma
 - ❖ ECC.API.Flotas.Data.dll
 - ❖ ECC.API.Flotas.dll
 - ❖ ECC.API.Flotas.Entity.dll
 - ❖ ECC.API.Flotas.Logic.dll
 - ❖ ECC.API.Flotas.Repository.dll
 - ❖ ECC.Api.Flotas.Resources.dll
 - ❖ ECC.API.Flotas.ServicesAgent.dll
 - ❖ ECC.API.Flotas.Translator.dll
 - Servicio de comunicación (solo lo mencionamos, formó parte de otros releases)
 - ❖ ECC.API.NotificadorTarjetasTpv.WebApi.dll

IV.4. Pruebas funcionales

Los programadores confirmaban a los analistas de calidad el despliegue de la historia de usuario en el entorno preproductivo para que se pueda dar inicio con las pruebas funcionales. Si existía algún desperfecto en las historias, inmediatamente se le notificaba al programador responsable para que pueda subsanarlo antes de la reunión con los interesados del proyecto.

Para las pruebas funcionales que se listan a continuación se necesitó desplegar en el ambiente preproductivo los artefactos Plataforma, Servicio de Comunicación y Punto de Venta, además de contar con acceso al laboratorio de pruebas de la ECC donde emulaban el funcionamiento de los surtidores (ver Anexo 2).

- **Reglas por Frecuencia:** se realizaron los siguientes pasos para la validación de esta regla.

En la Plataforma

1. Registrar una regla comercial estableciendo, entre otros, el número máximo de abastecimientos de combustible diario una cantidad igual a uno.
2. Registrar un vehículo.
3. Registrar una tarjeta tipo vehículo, seleccionando la placa del vehículo creado en el paso anterior.
4. Asociar la regla comercial creada en el paso 1 a la tarjeta creada en el paso anterior.
5. Generar el código QR de la tarjeta creada en el paso 3.

En el Punto de Venta

6. Digitar la placa del vehículo creado en el paso 2.
7. Capturar la información del QR a través de la lectora.

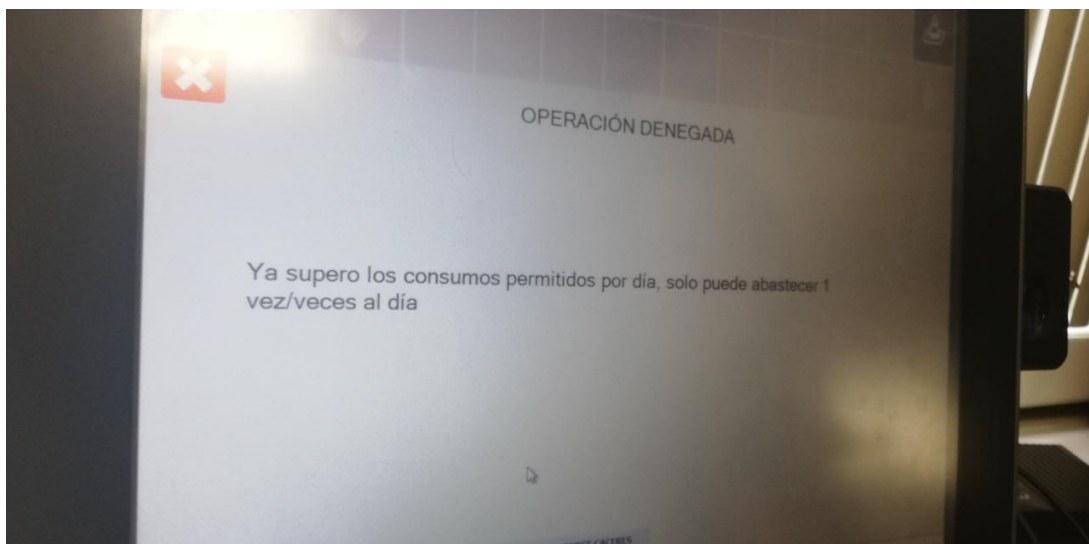
Figura 39. *Captura de la placa del vehículo y lectura de QR en el Punto de Venta*

The screenshot shows a mobile application interface with a light purple background. At the top, there is a header bar with a row of red icons representing different vehicle types. Below this, there are two input fields: 'Placa' (Plate) and 'Lectura de Código QR' (QR Code Reading). The 'Placa' field contains the text 'A1112D3'. To the right of these fields is a numeric keypad with buttons for digits 0-9, a decimal point, and a minus sign. There are also two buttons labeled 'DEL' and 'ENTR' (Enter). A red button labeled 'VALIDAR' (Validate) is positioned below the input fields. The interface is designed for capturing vehicle information and QR codes at a point of sale.

Fuente: Extraído del documento técnico ágil del proyecto.

8. Indicar la cantidad y el tipo de combustible a despachar.
9. Realizar el despacho de combustible y recepcionar el comprobante de pago (en este punto la tarjeta tiene una transacción realizada).
10. Realizar los pasos 6 al 8.
11. Cuando se intenta realizar el paso 9, el sistema deniega la operación mostrando el mensaje “Ya supero los consumos permitidos por día, solo puede abastecer 1 vez/veces al día”.

Figura 40. Validación de la regla por frecuencia en el Punto de Venta

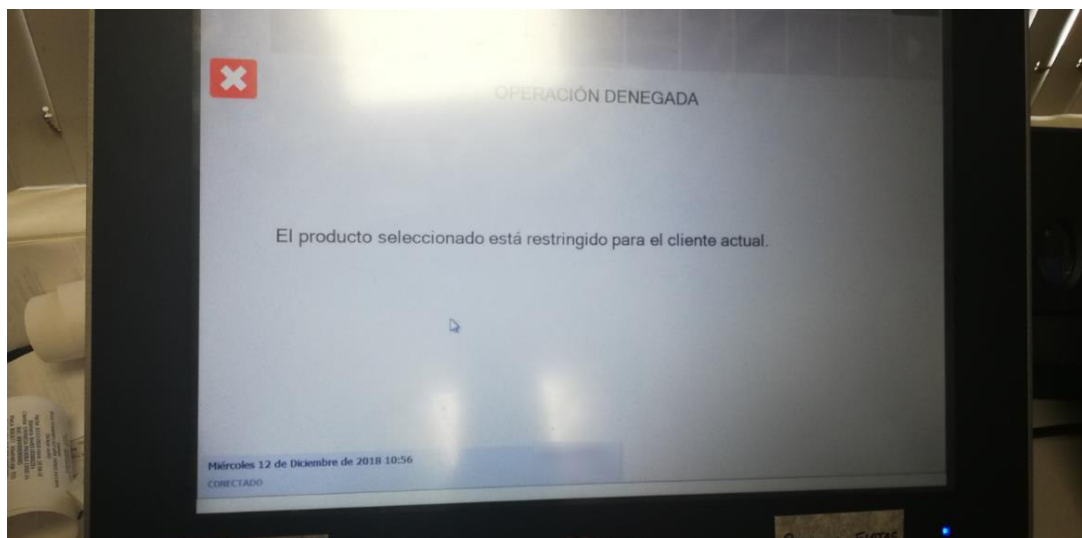


Fuente: Extraído del documento técnico ágil del proyecto.

De forma análoga se validó las otras reglas comerciales:

- **Regla de Producto:** cuando se configuraba una regla por producto en la Plataforma y se establecía que solo se podía consumir los productos Tipo 1, Tipo 2 y Tipo 3, luego cuando en el Punto de Venta se intentaba abastecer el combustible del Tipo 4, el sistema indicaba el mensaje que se visualiza en la figura 41.

Figura 41. *Validación de la Regla de Producto en el Punto de Venta*



Fuente: Extraído del documento técnico ágil del proyecto.

Luego llegaba en tiempo real una notificación a la Plataforma alertando al usuario que se estaba intentando realizar un consumo injustificado de combustible. La siguiente figura grafica lo anterior comentado. Las notificaciones incluían todas las alertas de consumo injustificado de combustible, así como mensajes de otros procesos (por ejemplo, proceso de carga masiva de vehículos a través de un archivo excel).

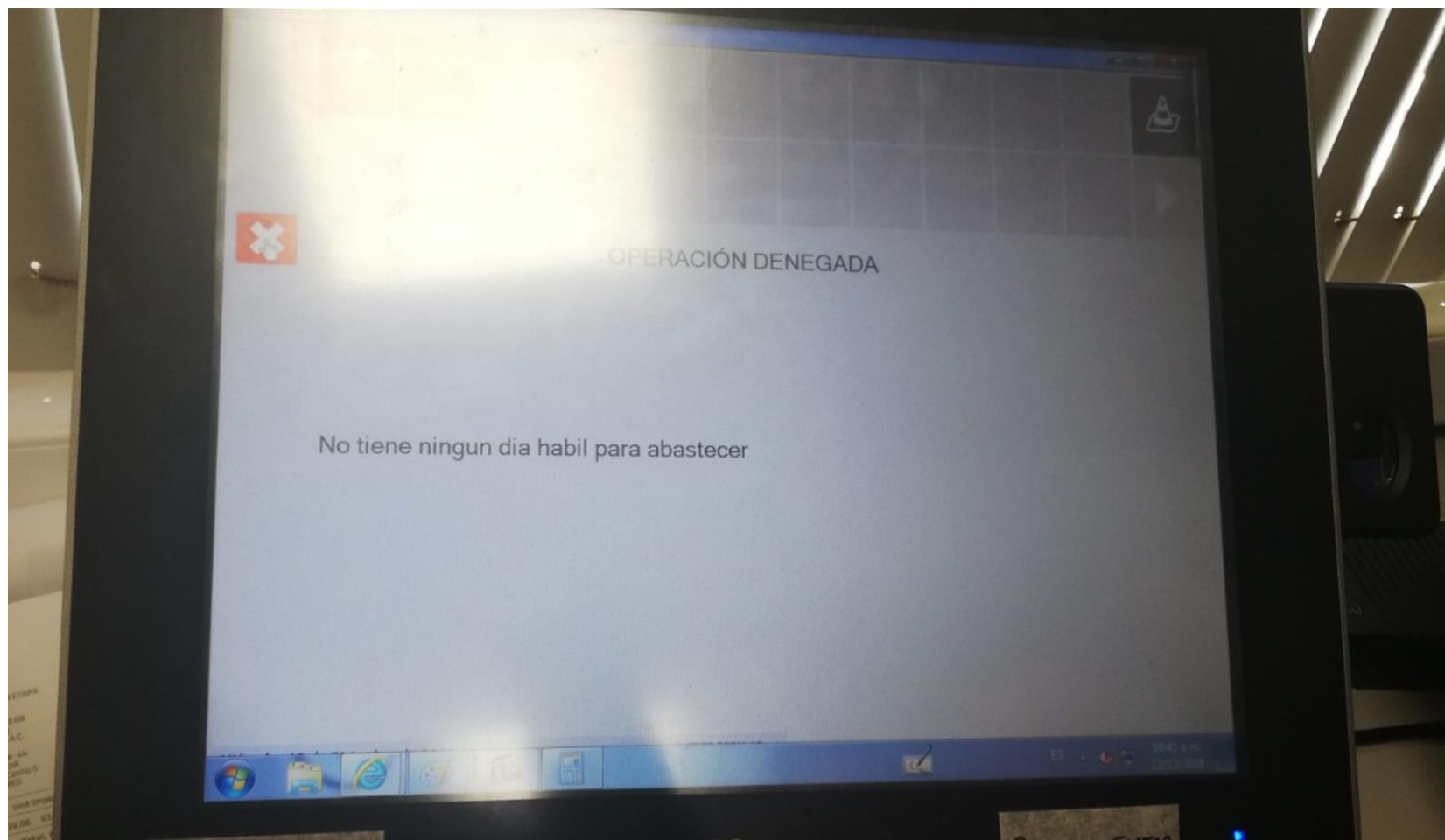
Figura 42. Notificación que llega a la Plataforma cuando se trata de consumir un tipo de producto no autorizado



Fuente: Extraído del documento técnico ágil del proyecto.

- **Reglas de Frecuencia por días habilitados:** cuando se tenía una regla de frecuencia en la Plataforma donde no se marcaba ningún día para el abastecimiento, luego cuando en el Punto de Venta se intentaba abastecer de combustible en cualquier día de la semana, el sistema indicaba el mensaje que se visualiza en la figura 43.

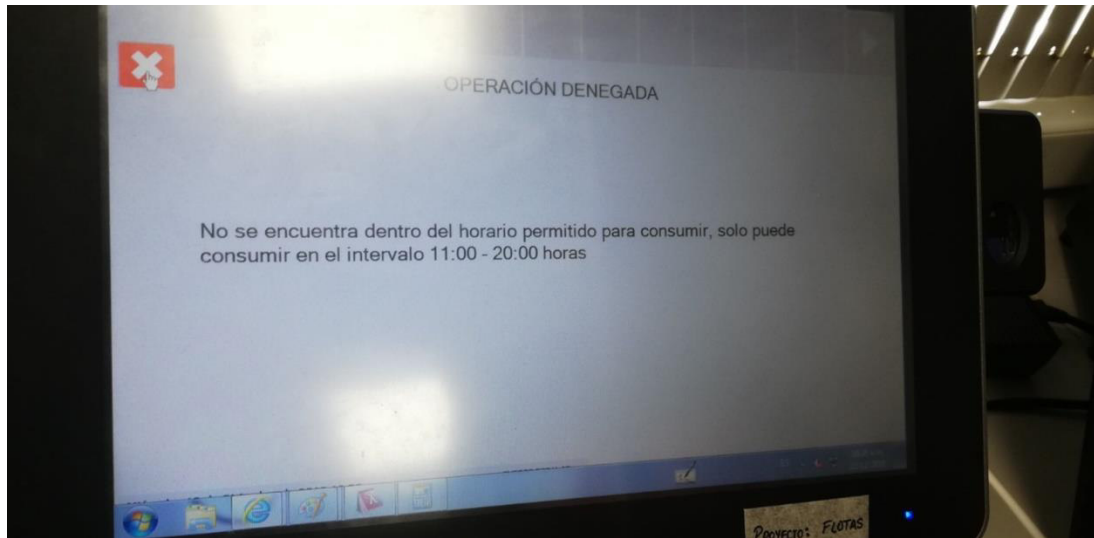
Figura 43. *Validación de la Regla de Frecuencia por Días en el Punto de Venta*



Fuente: Extraído del documento técnico ágil del proyecto.

- **Reglas de Frecuencia por horario:** cuando se tenía una regla de frecuencia en la Plataforma donde se establecía un horario para el abastecimiento desde las 11:00 hasta las 20:00 horas, luego cuando en el Punto de Venta se intentaba abastecerse de combustible fuera de este horario, el sistema indicaba el mensaje que se visualiza en la figura 44.

Figura 44. *Validación de la Regla de Frecuencia por horario en el Punto de Venta*



Fuente: Extraído del documento técnico ágil del proyecto.

Una vez que los analistas de calidad terminaban exitosamente con las pruebas de las historias de usuarios del sprint, notificaban el cierre del mismo al scrum master y a los otros miembros del scrum team (un ejemplo se muestra en el Anexo 3). Luego, se preparaba la presentación de las historias al product owner y stakeholders.

V. Fase de revisión

O también denominado el desarrollo del Sprint Review, en esta fase se realizó la demostración del funcionamiento de las historias del sprint backlog por parte del scrum team al product owner y a los stakeholders en el ambiente preproductivo.

V.1. Demostración a los interesados

Fueron reuniones donde el equipo por el lado de la ECM presentaba a los

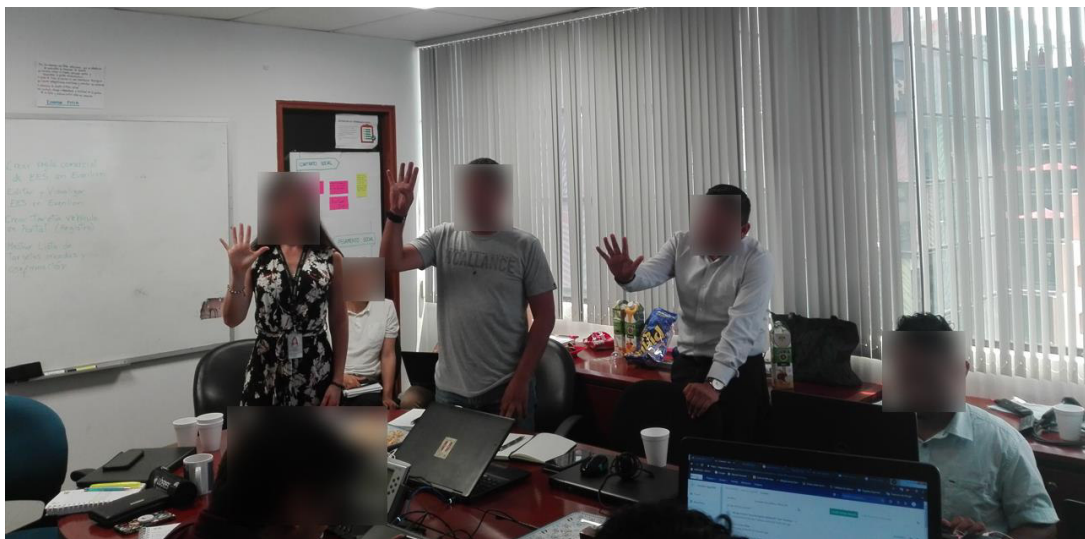
interesados (product owner y stakeholders) las historias de usuario desarrolladas en el sprint, donde se interactuaba con las funcionalidades de la solución en un entorno preproductivo.

Al finalizar la presentación, el equipo invitaba a los interesados a que realicen una puntuación del 1 al 5 sobre el funcionamiento de las historias presentadas, donde

1. Muy mala
2. Mala
3. Regular
4. Buena
5. Muy buena

En la mayoría de los casos los interesados indicaban una puntuación de 5, como se visualiza en la siguiente figura.

Figura 45. Evaluación de la implementación de las historias de usuario del sprint por parte del product owner y los stakeholders



Fuente: Extraído del Proyecto.

En los casos donde no se tenía una puntuación igual a 5, se les consultaba a los interesados el porqué de dicho puntaje y en base a ello el equipo tomaba acciones para mejorar. Otro aspecto importante fue que los interesados solicitaron

actualizaciones tanto a la definición de la historia de usuario como a los criterios de aceptación a medida que interactuaban con las funcionalidades. Estas actualizaciones se implementaron en el siguiente sprint.

VI. Fase de lanzamiento

En el proyecto se desplegaron 10 releases, donde cada release contenía uno a más sprints. El número de sprints en un release lo definió el product owner en coordinación con el scrum team.

VI.1. Generación del release

Para la elaboración del release se realizaron una serie de pasos que se listan a continuación:

- A. Elaboración del documento parte con las instrucciones detalladas y con las referencias correspondientes para la ejecución del pase a producción (el parte del release 5 se muestra en el Anexo 4, 5 y 6).
- B. Elaboración del documento técnico ágil (un ejemplo se muestra en el Anexo 7).
- C. Captura de evidencia de la ejecución de los de los scripts de rollback en el entorno de Desarrollo (un ejemplo se muestra en el Anexo 8).
- D. Elaboración de scripts de rollback (un ejemplo se muestra Anexo 9).
- E. Elaboración del documento donde envíen los TAG correspondientes a los Componentes de los proyectos (un ejemplo se muestra en el Anexo 10).

Hubo otros entregables que fueron del alcance del scrum master, por ejemplo, la conformidad de la capacitación del equipo de soporte que se muestra en el Anexo 11.

Estos pasos se realizaron en plena ejecución del sprint, donde 2

programadores del equipo (uno backend y el otro frontend), de forma rotativa, se encargaban de elaborar el release con proyección al pase a producción. Generalmente la elaboración y el pase se realizaba en la primera semana del sprint.

VI.2. Pase a producción

Consistió en el despliegue del release en el ambiente productivo. Para el caso de la Plataforma o el BackOffice, el pase se realizaba en las oficinas de la unidad de Sistemas de Información de la ECC. Para el caso del Servicio de Comunicación y el Punto de Venta, los pases se realizaban en las EES mediante la marcha blanca que contemplaba 1 o más EES piloto. A continuación, se visualiza el plan con las actividades involucradas en el pase del release 5, resaltando la participación del autor de este informe.

Figura 46. *Listado de actividades del pase a producción*

		Actividades	Equipo	Responsable	Fecha
Actividades Previas al día D	Comunicación	1. Enviar plan de despliegue	Proyecto	Cela Pardavé	19/06/2019
		2. Aprobación del plan de despliegue	TI/Negocio		20/06/2019
		3. Realizar transferencia a los equipos de Soporte (N1/N2) y TI , entrega de Documentación y capacitación	Proyecto	Raúl Bardales José Napa	13/06/2019
		4. Capacitación a formadores de	Proyecto (Gdc)	Walter Arellano	31/05/2019
		5. Comunicación a las áreas afectadas	TI/Negocio		D-2
	Preparar despliegue	6. Portal: Actualizar documentación parte	Proyecto	Fred Morales	21/06/2019
		7. Generar solicitud para ejecución de componentes	Proyecto	Fred Morales	21/06/2019
		8. Zippear archivos angular	Proyecto	José Napa	21/06/2019
		9. Homologar master con nueva rama	Proyecto	Fred Morales	25/06/2019
		10. Subir por SFTP al servidor componentes del parte del PORTAL y confirmar	Release & testing	Jorge López	26/06/2019
		11. Guardar Backup de la versión de los componentes (Previo al pase/Posterior al pase)	Proyecto / Release & testing	Jorge López	26/06/2019
		12. Portal: Realizar el pase a producción	Release & testing	Jorge López	D
Día D	Pase a producción	13. Generar QR Flota desde Portal	TI , Negocio, Proyecto	Gonzales, José Napa/Fred Morales	D
		14. Validar pase en TPV Laboratorio	TI , Negocio, Proyecto	Gonzales, José Napa/Fred Morales	D

Fuente: Extraído del proyecto.

La validación de la funcionalidad involucrada en los pases lo realizaban los asesores comerciales quienes accedían a la Plataforma o el BackOffice, donde seleccionaban un cliente de pruebas, creaban reglas comerciales y luego estas eran validadas en el Punto de Venta. Estas validaciones eran supervisadas por el analista de sistemas de la ECC, quien daba seguimiento y la conformidad del pase.

Por el lado del equipo de la ECM, el scrum team brindaba capacitación de las nuevas funcionalidades al equipo de soporte de la ECC.

Aproximadamente en dos meses, luego del despliegue release 2, ya se contaba con dos controles que restringían el consumo de combustible. Sin embargo, estos controles eran creados desde el BackOffice que era un sistema al que no tenían acceso los clientes, por lo que tenían que solicitar apoyo a los asesores comerciales de la ECC para aplicar estos controles a las tarjetas de los clientes.

Desde el inicio de la implementación del proyecto hasta el despliegue del release 4, que fueron seis meses aproximadamente, la Plataforma se habilitaba por primera vez para los clientes. Esta solución contaba con cinco controles y eran validadas en el Punto de Venta.

Luego de la implantación de toda la solución, el flujo del Servicio para Flotas se actualizó de la siguiente manera (tomaremos el flujo de venta indicado de la **Fase de inicio** y se hará una comparación con el flujo actual):

- a. **Elección de modalidad de pago:** se mantiene.
- b. **Establecer tipos de saldo:** se mantiene.
- c. **Distribuir saldos:** se mantiene.

La tarjeta flota es un registro que se puede crear desde la Plataforma o el BackOffice. El asesor comercial o el cliente puede optar por generar códigos QR (que caduca en 3 días como máximo) a partir de los registros de las tarjetas, para que puedan ser usadas en las EES, mientras se espera o no la entrega de las tarjetas físicas. Esta tarjeta se puede dividir en 3 tipos

- **Vehículo:** requiere el registro del vehículo en la Plataforma o el BackOffice, para luego asociarlo a la tarjeta. La tarjeta tiene impresa, entre otros, la placa del vehículo.
- **Conductor:** requiere el registro del conductor en la Plataforma o el BackOffice, para luego asociarlo a la tarjeta. La tarjeta tiene impresa, entre otros, el número de documento del conductor.

- **Pool:** la tarjeta puede tener impreso, entre otros, algún texto opcional.

Luego, el asesor comercial o cliente deberá crear el registro de la regla comercial con las especificaciones indicadas en el proceso *Funcionalidades de la Plataforma* de la **Fase de Implementación**, para luego asociarla a la tarjeta tipo vehículo o conductor.

- d. **Entregar tarjeta:** la ECC imprime las tarjetas de acuerdo a la solicitud del cliente.
- e. **Distribuir tarjetas:** el cliente puede distribuir las tarjetas o QRs entre los conductores de su flota o que las tarjetas sean repartidas y utilizadas en las EES donde acudirá su flota.

Cuando se crea el registro de la tarjeta en la Plataforma o el BackOffice, el asesor comercial o cliente podrá activar opcionalmente la función de Topes. Si lo activa, deberá ingresar de forma obligatoria los siguientes datos

- **Tope en Soles:** saldo tope de la tarjeta
- **Tipo de Control:** 15 días o 30 días
- **Tipo de Tope:** Reinicialable o Acumulable.

- f. **Realizar compra de combustible:** los conductores se acercaban a las EES para abastecerse de combustible de la siguiente manera, el texto resaltado en azul representa datos agregados / actualizados en el paso:

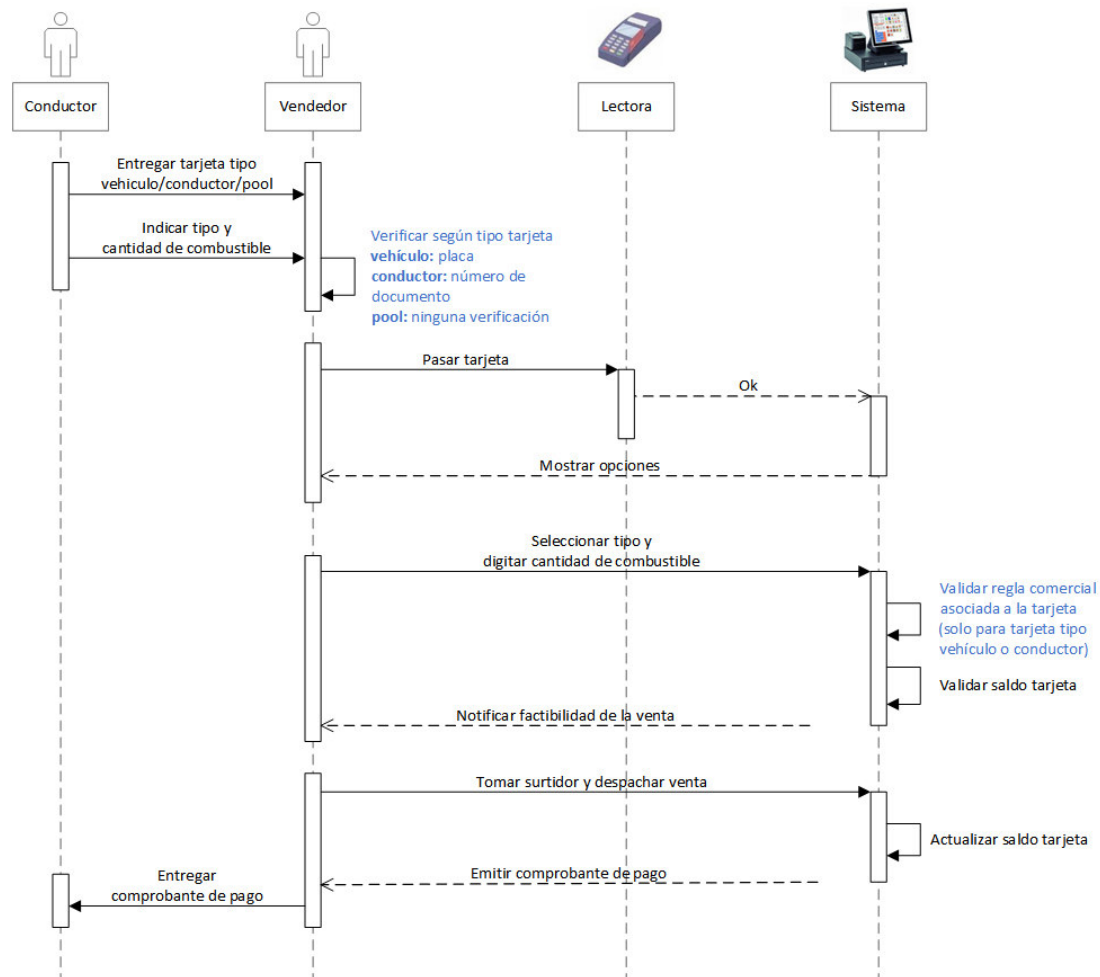
1. El conductor entrega su tarjeta flota al vendedor o este busca la tarjeta con la placa del vehículo o el número de documento en un estuche que maneja por cliente.
2. El conductor indica el tipo y la cantidad de combustible a abastecerse.
3. El vendedor verifica, según el tipo de tarjeta, que la placa del vehículo o el número de documento sea la misma que la placa o número de documento impresa en la tarjeta.
4. El vendedor pasa la tarjeta por la lectora, digita el tipo y la cantidad de combustible solicitado.
5. El sistema valida la regla comercial de la tarjeta para los tipos vehículo o

conductor y luego valida que cuente con el saldo suficiente.

6. El sistema notifica al vendedor la factibilidad de la venta.
7. El vendedor toma el surtidor y despacha la cantidad de combustible solicitado.
8. El sistema corta el suministro de combustible cuando se llega a la cantidad solicitada.
9. El sistema actualiza el saldo de la tarjeta y emite el comprobante de pago.
10. El vendedor entrega el comprobante al conductor.

En la figura 47 se grafica los puntos mencionados anteriormente.

Figura 47. *Flujo de la venta de combustible para el Servicio para Flotas después de la implantación de la solución*



Fuente: Elaboración propia

- g. **Solicitar recarga de saldo:** el asesor comercial o cliente realiza la programación de topes y saldos para la tarjeta en la Plataforma o el BackOffice, donde puede realizar las siguientes programaciones
- Incremento/Decremento de Saldo: afecta únicamente al saldo actual de la tarjeta.
 - Programación de Tope: afectará únicamente al saldo tope de la tarjeta. Se puede programar para cambiar el tope de la tarjeta hasta máximo 6 meses después del mes actual. Se puede elegir la primera o segunda quincena.
 - Cambio de Tipo de Control: el cambio de tipo de control modifica el tipo de 30 días a 15 días y viceversa.
 - Cambio de Tipo de Tope: solo modifica el tipo tope, de acumulable a reinicialable y viceversa.
- h. **Solicitar reporte de consumo diario:** el asesor comercial o cliente pueden visualizar los siguientes reportes en la Plataforma o el BackOffice
- Reporte de Consumo: listado de consumos realizados y total de importe facturado y no facturado.
 - Reporte de Saldo por Tarjeta: lista de tarjetas con información, entre otros, de topes y saldos.
 - Reporte de Estado de Cuenta: listado de facturas y total de deuda pendiente (deuda vencida + deuda por vencer). Tiene acceso el cliente crédito.
- i. **Enviar reporte:** poco a poco los asesores comerciales dejan de enviar de reportes de consumos diarios de las flotas a los clientes al final del día.
- j. **Revisar reporte:** paulatinamente el cliente deja de revisar los reportes de consumos diarios para dedicarse más a la información que se encuentra en los reportes de la Plataforma.

VII. Fase de evaluación del equipo

O también denominado sprint retrospective, donde participó el equipo por el lado de la ECM, dirigidas por el scrum master. Estas reuniones se realizaban para

dialogar acerca de los siguientes procesos:

VII.1. Agradecimientos de los miembros del equipo

Consistió en que cada integrante del equipo agradecía el apoyo que le había brindado otro integrante, ya sea dentro del equipo o parte del proyecto de modernización de las EES, a lo largo del sprint. Entre los agradecimientos destacaron:

- Arquitecto Software: por el apoyo brindado en la resolución de problemas técnicos y funcionales.
- Líder técnico del scrum team: por informar constantemente al equipo de las novedades respecto al flujo de trabajo, apoyo en la resolución de problemas técnicos y funcionales.
- Scrum master: por la buena presentación del equipo hacia el product owner y los stakeholders, por la gestión en los pases a producción.

VII.2. Identificación de buenas prácticas

Consistió en la identificación de actividades que ayudaban al equipo a mejorar su productividad, donde destacaron:

- Comunicación: el equipo trabajó en una gran habitación donde las consultas se realizaban cara a cara, con ello se ganaba rapidez en los tiempos de respuesta y todos estaban enterados de las actualizaciones.
- Documentación de componentes para el release: esta actividad se realizó cada vez que el programador culminaba con la implementación de las historias de usuario. Con ello se ganaba tiempo en la elaboración del release.
- Tiempo de entrega de las historias de usuario: las historias de usuario se desplegaron como máximo en la primera semana del sprint, esto permitió al analista de calidad ejecutar sus pruebas y notificar oportunamente a los programadores de alguna observación. Los programadores tuvieron hasta el jueves de la segunda semana para corregir las observaciones, con la finalidad de estabilizar la solución para la presentación de las historias de usuario al product

owner y stakeholders.

VII.3. Reporte de bloqueos

Consistió en la identificación de eventos que no permitieron avanzar con el desarrollo del sprint. Podemos destacar

- Ausencia del product owner en las reuniones del sprint planning, originando que el equipo seleccione las historias de usuario a realizar en el sprint.
- Tiempo de respuesta del arquitecto software, sobre consultas técnicas y funcionales.
- Entorno de trabajo donde se instaló el equipo (poco mantenimiento).

VII.4. Aspectos a mejorar

Consistió en la identificación de actividades en los que se necesitaba ajustes para mejorar la productividad del equipo. Entre ellas destaca

- Presentación del sprint review.
- Nivel de detalle en el documento ágil técnico.
- Participación de todo el equipo en el sprint planning.

Al término del sprint retrospective se tenía un plan de accionables a aplicarse en el siguiente sprint, donde uno o algunos de los integrantes del scrum team eran responsables de dar seguimiento a dichos accionables.

3.3. EVALUACIÓN

En esta sección se revisará los beneficios y costos que involucró la implantación de la Solución.

3.3.1. BENEFICIO PARA LOS CLIENTES DE LA ECC

Respecto a los resultados que originó la implantación de la Solución, se realizó una encuesta dirigida a 140 clientes que habían contratado el Servicio para Flotas de la ECC. Dicha encuesta fue enviada pasado aproximadamente 3 meses luego del despliegue del release 4. El formato de la encuesta fue la siguiente:

1. ¿Ha utilizado el Portal para controlar el consumo de combustible de su flota?
 - a) Si
 - b) No
2. Si su respuesta a la anterior pregunta fue Si, ¿A cuánto asciende el monto ahorrado mensualmente?
 - a) [S/ 0 - S/ 5,000 >
 - b) [S/ 5,000 - S/ 10,000 >
 - c) [S/ 10,000 - S/ 15,000 >
 - d) [S/ 15,000 - S/ 20,000 >
 - e) [S/ 20,000 - a más >
3. ¿Aún continúa solicitando el envío de reporte de consumo diario de su flota?
 - a) Si
 - b) No
4. Si su respuesta a la pregunta 1 fue Si, en general, ¿qué calificación le darías al Portal Flotas?
 - a) Muy mala
 - b) Mala
 - c) Regular
 - d) Buena
 - e) Muy buena

Los resultados de la encuesta se visualizan en los siguientes gráficos. De los 140 clientes encuestados, 74 (que representa 53%) estaban utilizando la Plataforma, tal como se aprecia en la figura 48. De los 74, 20 ahorraban en gasto de combustible un monto entre los S/ 10,000 y S/ 15,000; 14 ahorraban un monto entre los S/ 15,000 y S/ 20,000; y 16 ahorraban un monto de S/ 20,000 a más, según se aprecia en la figura 49.

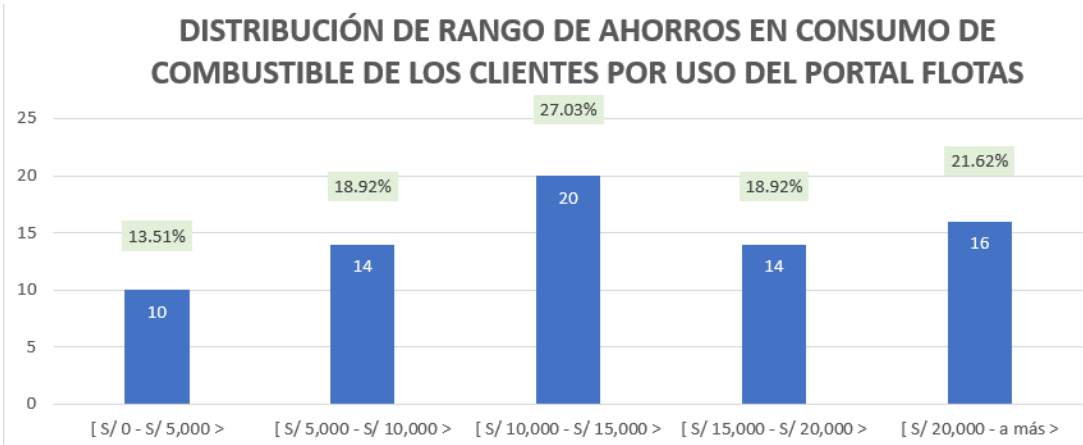
Figura 48. Resultado de la encuesta respecto a la pregunta 1

DISTRIBUCIÓN DE CLIENTES QUE USAN EL PORTAL FLOTAS



Fuente: Extraído del proyecto

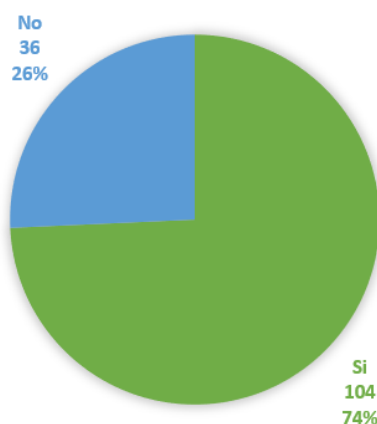
Figura 49. Resultado de la encuesta respecto a la pregunta 2



Fuente: Extraído del proyecto

Figura 50. Resultado de la encuesta respecto a la pregunta 3

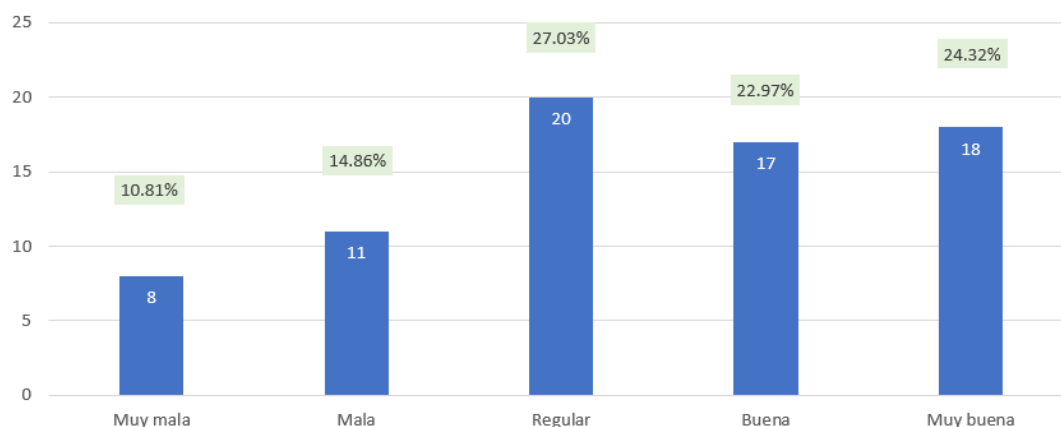
**DISTRIBUCIÓN DE CLIENTES QUE AÚN SOLICITAN
REPORTES DE CONSUMO DIARIO DE SU FLOTA**



Fuente: Extraído del proyecto

Figura 51. Resultado de la encuesta respecto a la pregunta 4

**DISTRIBUCIÓN DE CALIFICACIONES DE LOS CLIENTES RESPECTO
AL PORTAL FLOTAS**



Fuente: Extraído del proyecto

Respecto a la figura 50, 104 de 140 clientes, es decir, un 74% aún solicitan que se les envíe el reporte de consumo diario de su flota. Según el feedback que recibió el product owner de parte de los clientes, había algunos que a pesar que generaban controles desde la Plataforma, aún tenían cierta desconfianza respecto al funcionamiento y validación de los controles en el Punto de Venta.

En la figura 51 se puede apreciar que la mayoría de los clientes calificaron a la Plataforma de manera positiva. Con relación a las bajas calificaciones, estas guardan relación con la gráfica de la figura 49 ya que al parecer a los clientes si bien les estaba generando ahorro, estimaban que el monto ahorrado debería ser mayor.

3.3.2. BENEFICIO PARA LA ECC

Durante el despliegue del release 9, el producto owner realizó e informó al equipo sobre la disminución de horas en la carga administrativa de los asesores comerciales que había generado la implantación de la Solución. En la siguiente tabla se muestra el tiempo aproximado de atención de los clientes por asesor.

Tabla 9. *Tiempo ocupado aproximado por el asesor comercial en la atención de los clientes que tenían a cargo*

Antes de la implantación de la Solución							
Total clientes	Total asesores	Total clientes por asesor	Redondeo Total clientes por asesor	Por asesor			
				Envío reporte	Otras gestiones (recarga de saldo, consultas en general)	Total tiempo por cliente	Total tiempo por clientes a cargo
1500	158	9.49	9.50	00:10:00	00:25:00	00:35:00	05:32:30
Después de la implantación de la Solución							
Total clientes	Total asesores	Total clientes por asesor	Redondeo Total clientes por asesor	Por asesor			
				Envío reporte	Otras gestiones (recarga de saldo, consultas en general)	Total tiempo por cliente	Total tiempo por clientes a cargo
1500	158	9.49	9.50	00:05:00	00:10:00	00:15:00	02:22:30
							Tiempo ahorrado por asesor
							03:10:00

Fuente: Elaboración del product owner

Antes de la implantación de la Solución cada asesor tuvo a cargo 10 clientes aproximadamente, donde invertía 10 minutos en enviar el reporte de consumo diario y 25 minutos en realizar otras gestiones, por lo que el tiempo total fue de 35 minutos de atención de un cliente y 5 horas y media aproximadamente para la atención de los 10 clientes, diariamente.

Luego de la implantación de la Solución, cada asesor invertía 5 minutos en enviar los reportes, ya que algunos clientes dejaban de solicitarle dicho reporte; y de la misma forma sucedía con las otras gestiones, donde invertía 10 minutos, por lo que en total sumaba 15 minutos de atención de un cliente y 2 horas con 20 minutos aproximadamente para la atención de los 10 clientes que tuvieron a cargo. En consecuencia, se vio disminuida la carga administrativa de cada asesor en 3 horas aproximadamente, de forma diaria.

CAPÍTULO IV. REFLEXIÓN CRÍTICA DE LA EXPERIENCIA

El autor del presente trabajo de suficiencia profesional eligió este proyecto ya que fue la primera oportunidad donde pudo experimentar, de forma sólida y desde el inicio del proyecto, la aplicación de la metodología scrum y sus beneficios, experimentando las diversas ceremonias y dinámicas que se realizaron.

El desarrollo del proyecto permitió adquirir habilidades tanto técnicas como blandas. Por ejemplo, en lo técnico, entender la aplicación de la arquitectura DDD y cómo su uso permitía comprender más del negocio; respecto a las habilidades blandas, aumentó la habilidad de comunicación en la interacción con los integrantes del proyecto, tanto de la ECM como de la ECC, gracias a la participación en la daily, planning, refinamiento y exposición de las funcionalidades a los interesados del proyecto.

Otro aspecto importante fue la comunicación y coordinación, donde el equipo (product owner, scrum master y scrum team) se instaló en una mesa redonda donde todos podían comunicar rápidamente las novedades u observaciones que se presentaban en el desarrollo del sprint.

Por último y no menos importante, fue el manejo de la complejidad en la elaboración de los releases, ya que estos reunían componentes que habían sido creados con anterioridad (hasta 6 semanas antes) por lo que recuperarlos y mapearlos para el pase a producción fue una tarea compleja. Ante ello se establecieron una serie de procedimientos para que los programadores pudieran documentar los componentes, agilizando la elaboración de los releases.

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

La solución implantada de forma incremental permitió a la ECC mejorar el Servicio para Flotas, permitiendo a los clientes configurar reglas que controlaban el consumo de combustible, en la Plataforma. Esto originó una disminución en la carga administrativa de los asesores comerciales, reduciendo aproximadamente en 3 horas la atención a los clientes que realizaba cada asesor.

La mayoría de los clientes de la ECC calificaron de forma positiva a la Plataforma ya que les permitió controlar consumos injustificados de combustible, permitiendo reducir sus costos.

El despliegue de la solución de forma incremental permitió al equipo identificar requerimientos que o estaban mapeados en el backlog del producto con diferentes criterios de aceptación o fueron requerimientos nuevos, lo que al final permitió construir una solución adecuada a las necesidades de los clientes.

5.2. RECOMENDACIONES

Según lo comentado en el presente Trabajo de Suficiencia Profesional, se puede comentar las siguientes recomendaciones:

- Integrar un sistema de rastreo de vehículos que permita al gestor flotas ver el recorrido real de los vehículos y con ello saber el gasto de combustible consumido.
- Implantar un sensor en los tanques de los vehículos para saber la evolución del consumo de combustible.
- Crear un quinto artefacto de la solución y en base a la información identificada por los clientes sobre comportamientos inusuales de consumo de combustible, se podría construir un modelo de Machine Learning con aprendizaje supervisado donde el modelo se entrenaría con los datos de consumo de combustible justificado e injustificado, para finalmente, a través de un algoritmo de clasificación binaria (por ejemplo, redes neuronales) detectar si un consumo es normal o inusual.
- Aplicar continuamente una métrica de rendimiento en el scrum team (por ejemplo, burndwon chart) en base a la evolución de las historias de usuario entregadas en la duración del proyecto.

5.3. FUENTES DE INFORMACIÓN

- Angular. (2022). What is Angular? Recuperado de <https://angular.io>
- AppMaster. (2022). ¿Qué es la API REST y en qué se diferencia de otros tipos? Recuperado de <https://appmaster.io>
- Appvizer. (2020). Realiza estimaciones ágiles y precisas gracias al Planning Poker Recuperado de <https://www.appvizer.es>
- Chakray. (2022). SWAGGER Y SWAGGER UI: ¿QUÉ ES Y POR QUÉ ES IMPRESCINDIBLE PARA TUS APIS? Recuperado de <https://www.chakray.com>
- COMSATEL. (2019). ¿Conozca cuáles son las modalidades de robos de combustible y sepa cómo prevenirlos? Recuperado de <https://www.comsatel.com.pe>
- Cuéllar J. (2016). Domain-Driven Design. Arquitectura. Recuperado de <https://josecuellar.net>
- ECC. (2021). Memoria Anual.
- ECM. (2017). Memoria descriptiva.
- Evans E. (2004). Domain-Driven Design: Tackling Complexity in the Heart of Software. Addison-Wesley Professional.
- IBM. (2021). API REST. Recuperado de <https://www.ibm.com>
- Medium. (2019). Componentes en Angular. Recuperado de <https://medium.com>
- Microsoft. (2022a). Le damos la bienvenida al IDE de Visual Studio. Recuperado de <https://docs.microsoft.com>
- Microsoft. (2022b). Documentación de la API web de ASP.NET Core con Swagger/OpenAPI. Recuperado de <https://docs.microsoft.com>
- Microsoft. (2021a). Get started with .NET Framework. Recuperado de <https://docs.microsoft.com>
- Microsoft. (2021b). Componentes y configuración de SQL Server Management Studio. Recuperado de <https://docs.microsoft.com>
- Movildata. (2022). ¿Qué es la gestión de flotas? Recuperado de <https://movildata.com>
- Nextech. (2021). ¿Qué es SQL Server? y ¿Para qué sirve? Recuperado de <https://nextech.pe>
- Rasmusson, J. (2010). The Agile Samurai. How Agile Masters Deliver Great Software. Pragmatic Works Inc Programmers.
- Redson. (2021). Indicadores de gestión de flota vehicular: ¿cuáles son los más

importantes? Recuperado de <https://redson.pe>

SAP. (2022). What is Gigya? <https://www.sap.com>

Schwaber K. y Sutherland J. (2020). The Scrum Guide. The Definitive Guide to Scrum: The Rules of the Game.

Scrum Poker Cards. (2022). Scrum Poker Cards (Agile) Recuperado de <https://play.google.com>

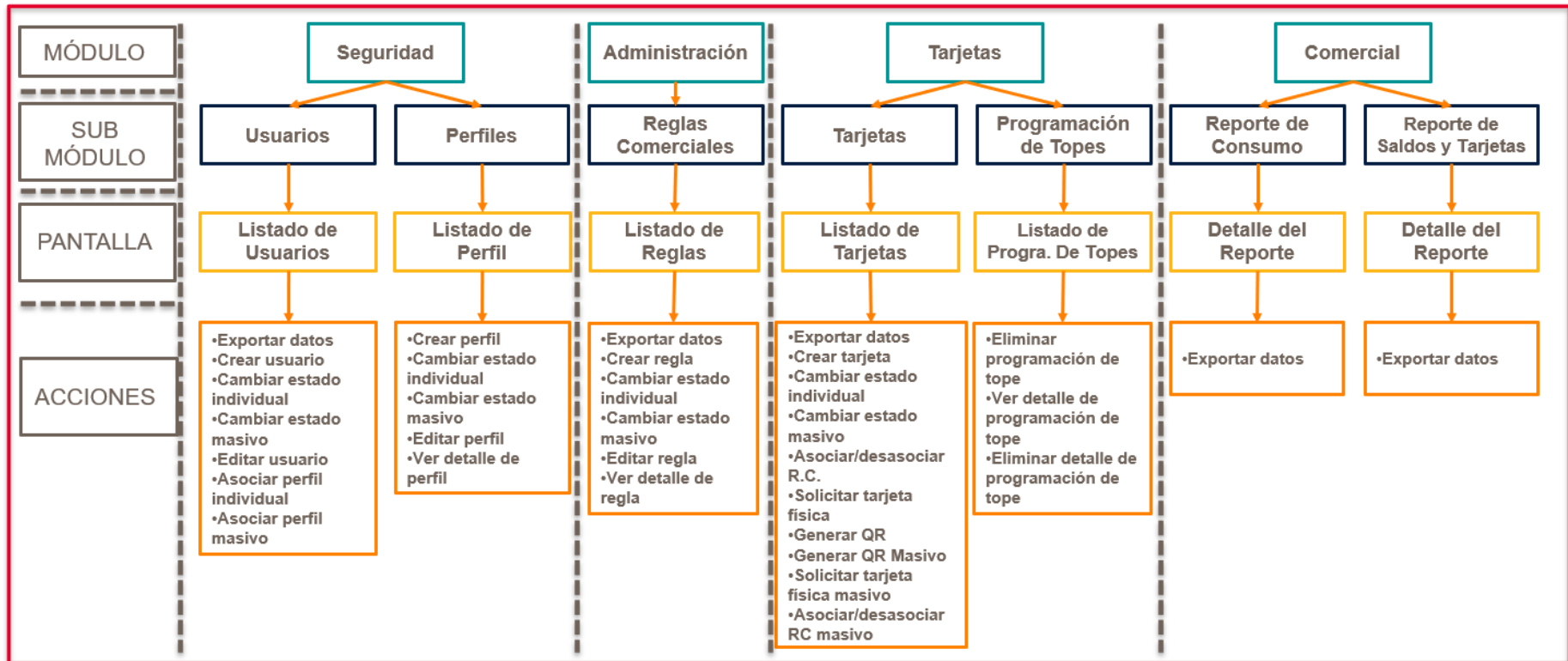
SCRUMstudy™. (2013). A Guide to the SCRUM BODY OF KNOWLEDGE (SBOK™ Guide). Phoenix, Arizona.

5.4. GLOSARIO

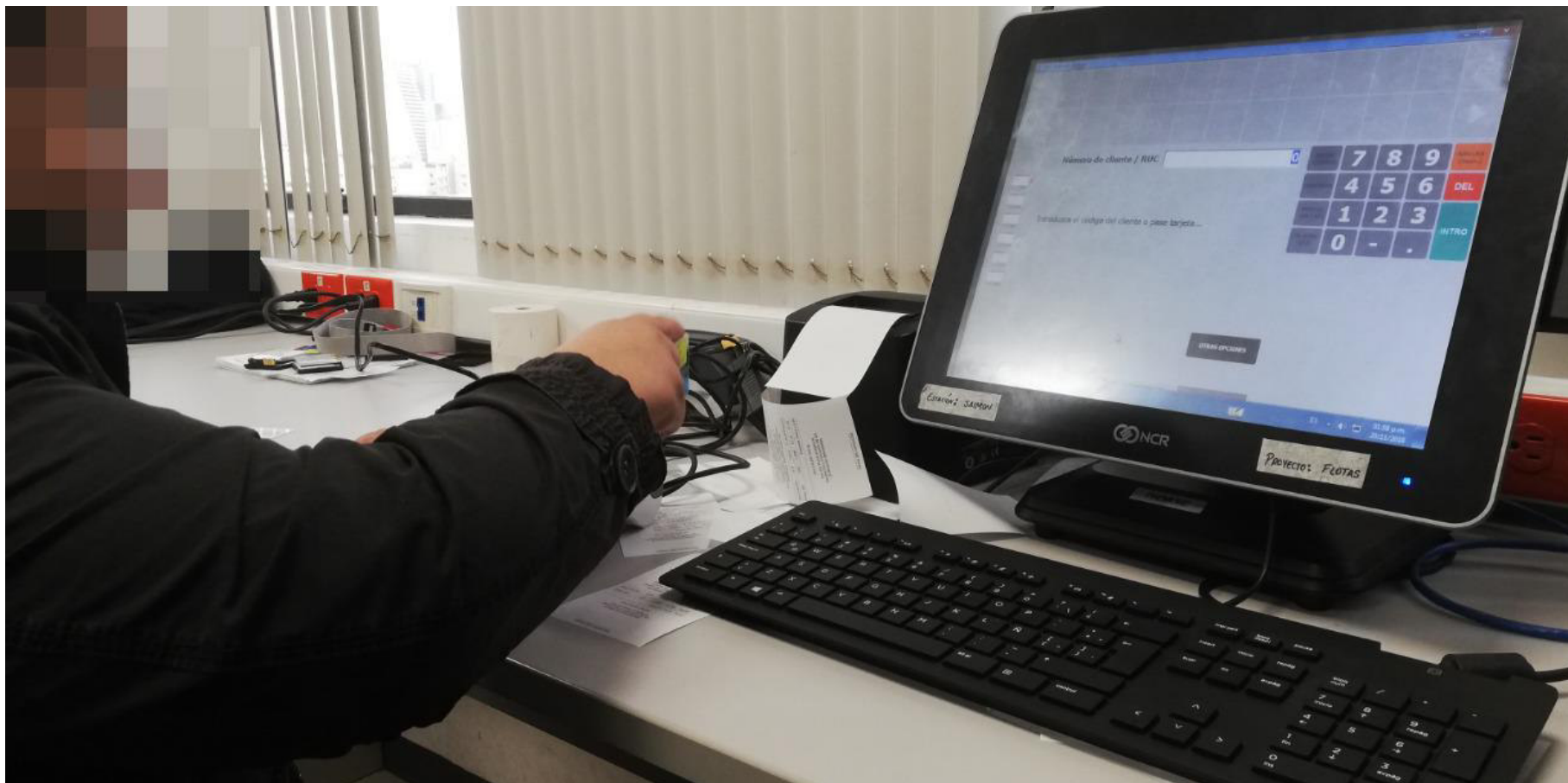
- **BackOffice:** Sistema que utilizan los asesores comerciales para generar los reportes de consumo de combustible de flotas vehiculares y permite dar de alta al Gestor de Flota en la Plataforma.
- **ECC:** Empresa Comercializadora de Combustible. Cliente en donde el autor adquirió la experiencia laboral para la realización del presente Trabajo de Suficiencia Profesional.
- **ECM:** Empresa Consultora Multinacional, donde el autor estuvo en la unidad ITS&S con el cargo programador backend para el cliente ECC.
- **EES:** Estaciones de servicio. Lugar de la ECC donde se realiza, entre otros, la venta de combustible.
- **Gestor de Flota:** Persona natural que gestiona la flota vehicular de la empresa en la que trabaja, tiene acceso a la Plataforma.
- **Plataforma:** Herramienta tecnológica que permite a los clientes corporativos realizar la gestión, monitoreo y control del consumo de combustible de su flota vehicular.
- **Punto de venta:** módulo que se encarga de registrar las ventas de combustible en las EES.
- **Regla comercial:** son una serie de controles que permite restringir el consumo de combustible de las flotas.
- **Servicio de Comunicación:** interface que permite la comunicación del punto de venta con la Plataforma, con la finalidad de validar las reglas comerciales.
- **Tipo de producto:** los productos Tipo 1, 2, 3, 4 y 5 corresponden a los productos efitec 98, efitec 95, efitec 90, Diesel y GLP.

ANEXOS

Anexo 1. Vista de los módulos de la Plataforma



Anexo 2. Laboratorio de pruebas donde se emula el funcionamiento del Punto de Venta



Anexo 3. Conformidad del asistente de calidad respecto a la funcionalidad Solicitar Tarjeta Física de la Plataforma



Mary Carmen

Susan Alison

+ 2

 2

18/06/2019

CC Mary Carmen

 TARJETA FISICA-REGRESIÓN.docx.xlsx
Archivo .xlsx

 TARJETA FISICA-REMEMBER.docx
Archivo .docx

Estimados,

Se realizó una regresión con solicitud de TARJETAS FISICAS desde el portal y la verificación del caso desde el BO, así mismo adjunto las evidencias del caso y el Excel con todos los casos de prueba realizados.

Saludos.

Atte:

Mary Carmen

Anexo 4. Documento Parte del release 5 de la Plataforma

DESCRIPCION TECNICA (PARTE)						
Se incluirá en este apartado la lista de ficheros y su cometido. (Instrucciones en cabecera de columnas)						
Responsable	Archivo			BBDD (si aplica)	Tipo	Descripción / Observaciones
FMORALCA		API/Flotas/bin/	.API.Flotas.Data.dll		ARCHIVO	
FMORALCA		API/Flotas/bin/	.API.Flotas.dll		ARCHIVO	
FMORALCA		API/Flotas/bin/	.API.Flotas.Entity.dll		ARCHIVO	
FMORALCA		API/Flotas/bin/	.API.Flotas.Logic.dll		ARCHIVO	
FMORALCA		API/Flotas/bin/	.API.Flotas.Repository.dll		ARCHIVO	
FMORALCA		API/Flotas/bin/	.Api.Flotas.Resources.dll		ARCHIVO	
FMORALCA		API/Flotas/bin/	.API.Flotas.ServicesAgent.dll		ARCHIVO	
FMORALCA		API/Flotas/bin/	.API.Flotas.Translator.dll		ARCHIVO	
CABECERA INSTRUCCIONES ARCHIVOS BBDD SONAR						

Anexo 5. Documento Parte del release 5 del BackOffice

DESCRIPCION TECNICA (PARTE)					
Se incluirá en este apartado la lista de ficheros y su cometido. (Instrucciones en cabecera de columnas)					
Responsable	Archivo	BBDD (si aplica)	Tipo	Descripción / Observaciones	
IMEDINAG	/bin/[REDACTED].Flotas.BackOffice.UI.dll		ARCHIVO	Proyecto	-app0204- rama : develop
IMEDINAG	/bin/descargar.aspx.435baa93.compiled		ARCHIVO	Proyecto	-app0204- rama : develop
IMEDINAG	/bin/Custom.[REDACTED].FlotasBO.dll		ARCHIVO	Proyecto	-app0204- rama : develop
IMEDINAG	/bin/bandejasolicitud.aspx.435baa93.compiled		ARCHIVO	Proyecto	-app0204- rama : develop
FMORALCA	/bin/[REDACTED].Business.Card.Lib.dll		ARCHIVO	Proyecto	-lib0205- rama : develop
CABECERA INSTRUCCIONES ARCHIVOS BBDD SONAI ...					

Anexo 6. Documento de creación de objetos en el BackOffice

ITEM	DESCRIPCIÓN	Hipervínculo	PÁGINA MÓVIL	MENÚ	ORDEN	SUBMENÚ	MOSTRAR EN MENÚ	ARCHIVO	BÚSQUEDA LATERAL
0J1	Bandeja Solicitud			Gestión Clientes	1	Formularios	SI	BandejaSolicitud.aspx	
Los siguientes objetos no hace falta crearlos, puesto que ya existen									
► Módulos comerciales Objetos de menú Accesos +									

Anexo 7. Documento técnico ágil de release 5

1



Documento Técnico Funcional Ágil

2

1. Objetivo.

2. Impacto Funcional.

3. Diagrama de la funcionalidad

4. Impacto Técnico

- Arquitectura
- Angular
- .Net
- Base Datos

5. Ubicación de Componentes

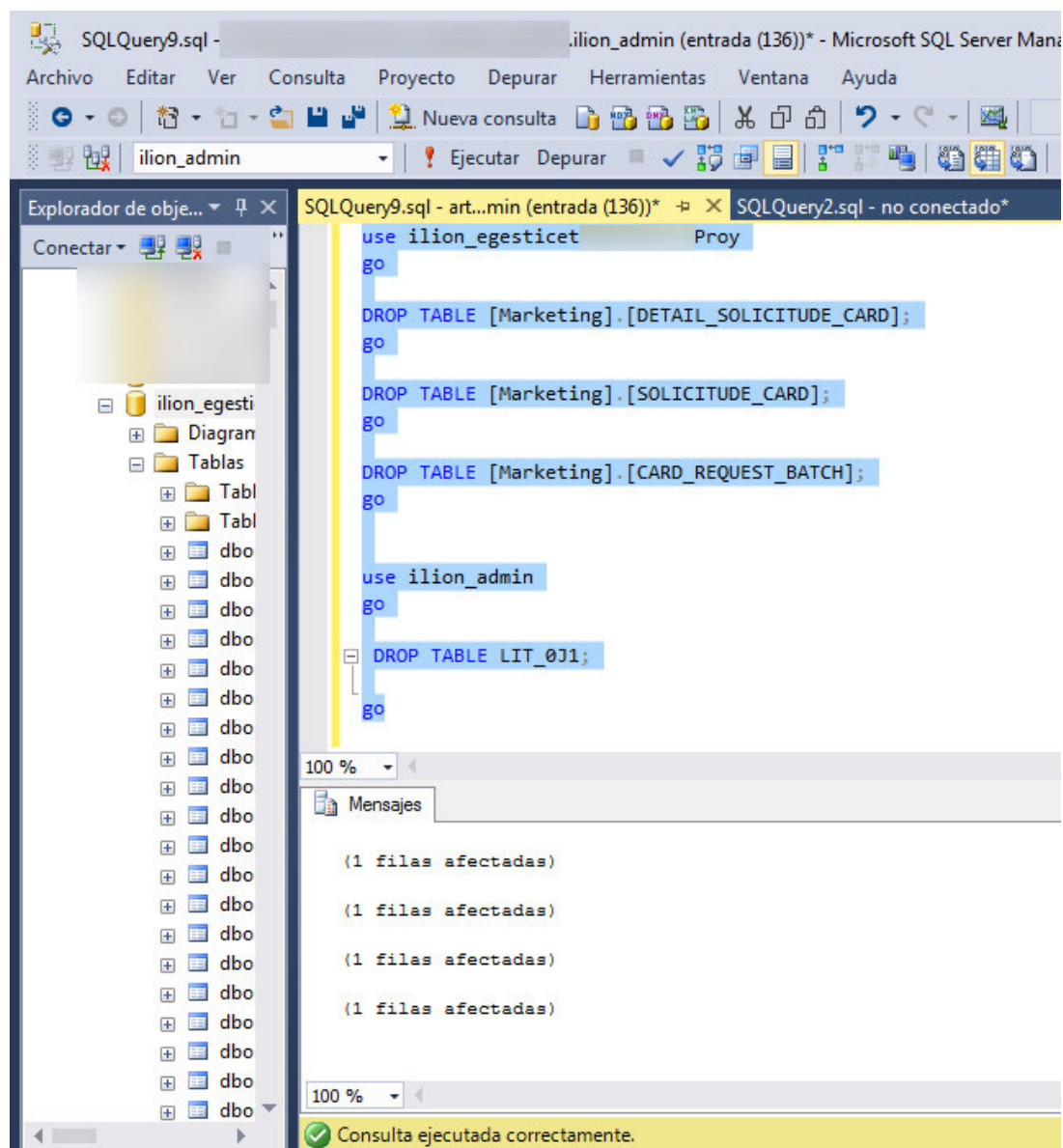
2



Documento Técnico Funcional Ágil

1

Anexo 8. Evidencia ejecución de los scripts de rollback del release 5



Anexo 9. Rollback del release 5 de la Plataforma

ROLLBACK SCRIPTS
<pre>/* ROLLBACK PROCEDIMIENTOS, JOB, UPDATE, INSERTS, ALTERS, TABLAS, TIPOS - CELULA 1 JUPITER */ /*===== -- NAME: Rollbacks -- AUTOR: C1 Jupiter -- CREATION: 21/06/2019 -- BBDD: ilion_egesticet [redacted] Proy, ilion_egesticet [redacted] -- DESCRIPTION: DROP DE PROCEDIMIENTOS, JOB, UPDATE, INSERTS, ALTERS, TABLAS, TIPOS -- ERRORES: - ===== */ --ROLLBACK PROCEDIMIENTOS USE ilion_egesticet [redacted] PROJ GO BEGIN TRANSACTION; BEGIN TRY DROP PROCEDURE [Marketing].[uspDriverById] DROP PROCEDURE [Marketing].[uspDriverByPlate] DROP PROCEDURE [Marketing].[uspDriverFleetSummary] DROP PROCEDURE [Marketing].[uspDriverJoinPlate] DROP PROCEDURE [Marketing].[uspDriverValidationDocument] DROP PROCEDURE [Marketing].[uspEnvioDeQRMasivoPorEmail] DROP PROCEDURE [Marketing].[uspEnvioDeQRporEmail] DROP PROCEDURE [Marketing].[uspExpireQR] DROP PROCEDURE [Marketing].[uspGetAllProfiles] DROP PROCEDURE [Marketing].[uspGetDataQR] DROP PROCEDURE [Marketing].[uspGetListActions]</pre>

Anexo 10. Documento TAG del release 5

GIT – Proyecto: [REDACTED]-app0208
Ruta del Proyecto: https://[REDACTED]-app0208.git
Branch: <u>branch-C1_R5</u>
Componente [REDACTED].API.Flotas
TAG: - Creación de tarjetas (conductor, vehículo, pool) - Generar QR Flota - Solicitar tarjeta física - Caducar QR (individual y masiva) - QR masivo: generación de varios QR en un solo evento. - Validación de cambio de estados de tarjeta. - Gestión de usuarios - Gestión de Perfiles - Asociar perfiles a usuarios - Mi perfil - Seleccionar empresa para el Rol Asesor - Validaciones programación - Programación topes de saldos - Programación de tipo de saldo - Incremento y decremento de saldos - Visualizar y anulación de programaciones - Anular tarjeta de programación - Crear conductores - Listar conductores - Pantalla de Programaciones - Programación de tipo tope

Anexo 11. Conformidad del equipo de soporte respecto a la capacitación de las funcionalidades de la Solución desplegadas en el del release 5



Alexander

Celia

Emmanuel

+ 6

18/06/2019

RE: C1 - Transferencia a N2 Portal V3 y BO

CC Emmanuel; Edinson; Jose; MARCELA; Jina; JUAN CARLOS; Fred Anthony Morales Cabezas

Buenas tardes Celia,

Por parte de N2 no presentamos observación de la transferencia brindara.

Quedo atento ante cualquier duda o consulta

Saludos,
Alexander
Correo:

De: Celia

Enviado el: lunes, 17 de junio de 2019 7:58 p. m.

Para: Alexander

CC: Emmanuel; Edinson; Jose; MARCELA; Jina; JUAN CARLOS; Fred Anthony Morales Cabezas

Asunto: C1 - Transferencia a N2 Portal V3 y BO

Alexander,

El 13 de Junio se realizó la transferencia a N2 y N1 del Release 5 de la C1-Jupiter, que corresponde al Portal V3 y BO.

Solicitamos la conformidad del mismo.

Saludos.