



**FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE COMPUTACIÓN Y SISTEMAS**

**PROCESO AUTOMÁTICO PARA EL PAGO DE
COMISIONES ESPECIALES A BROKERS DEL
SEGMENTO EMPRESAS DE RIMAC**

**PRESENTADO POR
AGUSTIN ALBERTO MEGO AGUILERA**

**ASESOR
GENER VICTOR ZAMBRANO LOLI**

**TRABAJO DE SUFICIENCIA PROFESIONAL
PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE
COMPUTACIÓN Y SISTEMAS**

**LIMA – PERÚ
2024**



CC BY-NC-ND

Reconocimiento – No comercial – Sin obra derivada

El autor sólo permite que se pueda descargar esta obra y compartirla con otras personas, siempre que se reconozca su autoría, pero no se puede cambiar de ninguna manera ni se puede utilizar comercialmente.

<http://creativecommons.org/licenses/by-nc-nd/4.0/>



UNIVERSIDAD DE SAN MARTÍN DE PORRES

FACULTAD DE INGENIERÍA Y ARQUITECTURA

ESCUELA PROFESIONAL DE INGENIERÍA DE COMPUTACIÓN Y SISTEMAS

PROCESO AUTOMÁTICO PARA EL PAGO DE COMISIONES
ESPECIALES A BROKERS DEL SEGMENTO EMPRESAS DE RIMAC

TRABAJO DE SUFICIENCIA PROFESIONAL

PARA OPTAR

EL TÍTULO PROFESIONAL DE INGENIERO DE COMPUTACIÓN Y SISTEMAS

AUTOR

MEGO AGUILERA, AGUSTIN ALBERTO

ASESOR

MTRO. ZAMBRANO LOLI, GENER VÍCTOR

ORCID: 0000-0001-5173-8337

Lima – Perú

2024

Dedicado a mi padre y a mi madre que siempre me alientan a seguir adelante.

Agradezco a Dios por todo lo que me ha dado y me sigue dando.

A mis padres y a mi familia, por el apoyo incondicional.

A mi casa de estudios, por formarme como el profesional que soy.

RESUMEN

Los procesos son conjuntos de tareas planificadas que forman la base de la gestión estratégica para muchas empresas. La tecnología ha avanzado de una forma exponencial en los últimos años, dotando de herramientas para que el hombre pueda realizar, de una mejor manera, sus tareas diarias. La automatización de procesos consiste en emplear recursos tecnológicos con el fin de mejorar los procesos empresariales, volviéndose cada vez más importante para el crecimiento sostenible de las empresas. El presente trabajo de suficiencia profesional se enfoca en la aplicación de la tecnología para automatizar un proceso de negocio que, hasta antes de implementar la solución, se realizaba de manera manual. Para este propósito, la solución emplea SCRUMBAN como metodología ágil, el cual mezcla las ceremonias de SCRUM y el tablero visual de KANBAN para mantener un orden en el flujo de trabajo.

Palabras clave: Proceso, Tarea, Automatización, Python, SCRUMBAN

ABSTRACT

Processes are sets of planned tasks that form the basis of strategic management for many companies. Technology has advanced exponentially in recent years, providing tools so that man can perform his daily tasks in a better way. Process automation consists of using technological resources in order to improve business processes, becoming increasingly important for the sustainable growth of companies. This professional proficiency work focuses on the application of technology to automate a business process that, until before implementing the solution, was carried out manually. For this purpose, the solution uses SCRUMBAN as an agile methodology, which mixes the SCRUM ceremonies and the KANBAN visual board to maintain order in the workflow.

Keywords: Process, Task, Automation, Python, SCRUMBAN

REPORTE DE TURNITIN

Reporte de similitud

NOMBRE DEL TRABAJO

PROCESO AUTOMÁTICO PARA EL PAGO
DE COMISIONES ESPECIALES A BROKE
RS DEL SEGMENTO EMPRESAS DE RIMA
C

AUTOR

AGUSTIN ALBERTO MEGO AGUILERA

RECuento de palabras

10333 Words

Recuento de caracteres

60388 Characters

Recuento de páginas

78 Pages

Tamaño del archivo

3.1MB

Fecha de entrega

Jun 26, 2024 9:09 AM GMT-5

Fecha del informe

Jun 26, 2024 9:10 AM GMT-5

● 4% de similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para cada base de datos.

- 3% Base de datos de Internet
- Base de datos de Crossref
- 3% Base de datos de trabajos entregados
- 1% Base de datos de publicaciones
- Base de datos de contenido publicado de Crossref

● Excluir del Reporte de Similitud

- Material bibliográfico
- Material citado
- Material citado
- Coincidencia baja (menos de 8 palabras)



Resumen

INTRODUCCIÓN

La solución del presente documento fue desarrollada durante el primer trimestre del presente año 2024. En este documento se evidencia las fases que se siguieron para lograr el desarrollo de un proceso automático para el pago de comisiones, entregando como herramientas finales diversos scripts y un reporte BI para la visualización de los resultados.

Es importante mencionar a RIMAC Seguros y Reaseguros, empresa líder en el mercado de seguros en Perú y que cuenta con mas de 100 años de experiencia, además de pertenecer al grupo empresarial más grande del Perú como es el Grupo BRECA, el cual cuenta con un portafolio variado de empresas entre los cuales destacan BBVA Perú, Minsur y RIMAC Seguros.

El desarrollo de la solución se dio bajo la metodología ágil SCRUMBAN, la cual ayudó a tener un orden y un mejor control de las tareas que se iban realizando en cada sprint. Al ser un híbrido entre SCRUM y KANBAN, SCRUMBAN utiliza las ceremonias de SCRUM para poder afinar las tareas de las historias de usuario planificadas al inicio de cada sprint, agregando el tablero visual de KANBAN como un artefacto importante en el desarrollo de la solución, ofreciendo un flujo de trabajo visual y muy entendible al momento de tener las dailys y reviews.

Debido al modelo ágil que predomina en la mayoría de las divisiones de RIMAC, las áreas pasaron a llamarse tribus, que a su vez están conformadas por squads y sus miembros pasaron a denominarse Business Developers, los cuales realizan tareas orientadas al objetivo que cada squad tiene y que, en el caso de RIMAC, se evalúan cada trimestre.

En el capítulo I se describe la experiencia profesional del autor, como también información relevante de la empresa donde se realiza el presente trabajo.

En el capítulo II se explica el problema central y el desarrollo de la solución, mostrando las diferentes fases del proyecto y como se logró cumplir con los objetivos.

ÍNDICE DE CONTENIDO

RESUMEN	4
ABSTRACT	5
REPORTE DE TURNITIN	6
INTRODUCCIÓN	7
ÍNDICE DE CONTENIDO	8
ÍNDICE DE FÍGURAS	10
ÍNDICE DE TABLAS	12
ÍNDICE DE ANEXOS	13
I. EXPERIENCIA PROFESIONAL	14
1.1. TRAYECTORIA PROFESIONAL	14
1.2. CONTEXTO EN EL QUE SE DESARROLLÓ EL TRABAJO DE SUFICIENCIA PROFESIONAL	15
1.2.1. DESCRIPCIÓN DE LA EMPRESA DONDE DESARROLLO LA EXPERIENCIA PROFESIONAL	15
1.2.2. VISIÓN	15
1.2.3. MISIÓN.....	15
1.2.4. ORGANIGRAMA	15
1.2.5. DESCRIPCIÓN DEL CARGO	16
1.2.6. SERVICIOS	16
1.3. CONTRIBUCIÓN A LA EXPERIENCIA PROFESIONAL.....	16
1.3.1. CONTRIBUCIÓN EN LA SOLUCIÓN DE PROBLEMAS QUE SE HAYAN PRESENTADO DURANTE SU ESTANCIA EN LA EMPRESA	16
1.3.2. CONTRIBUCIÓN EN TÉRMINOS DE LAS COMPETENCIAS Y HABILIDADES ADQUIRIDAS DURANTE SU FORMACIÓN PROFESIONAL	17
1.4. REFLEXIÓN CRÍTICA DE LA EXPERIENCIA	17
1.4.1. APORTES Y BENEFICIOS OBTENIDOS POR EL CENTRO LABORAL.....	17

1.4.2.	PLANTEAMIENTO DE MEJORAS PROPUESTAS A LA ORGANIZACIÓN....	17
II.	INFORME DEL PROYECTO DE ESPECIALIDAD	18
2.1.	METODOLOGÍA.....	18
2.1.1.	DESCRIPCIÓN DE LA METODOLOGÍA	18
2.1.2.	ORGANIZACIÓN DE LA METODOLOGÍA.....	19
2.2.	DESARROLLO	21
2.2.1.	SITUACIÓN PROBLEMÁTICA	21
2.2.2.	DISEÑO DE LA SOLUCIÓN	25
2.3.	CONCLUSIONES Y RECOMENDACIONES	67
2.3.1.	CONCLUSIONES	67
2.3.2.	RECOMENDACIONES.....	69
	REFERENCIAS	70
	ANEXOS.....	71

ÍNDICE DE FIGURAS

Figura 1 Organigrama resumido - RIMAC	15
Figura 2 EDT	19
Figura 3 Cronograma de actividades.....	20
Figura 4 Flujo automatizado	28
Figura 5 Ejemplo query: 1ra parte	33
Figura 6 Ejemplo query: 2da parte	34
Figura 7 Ejemplo query: 3ra parte	35
Figura 8 Importar librerías	36
Figura 9 Importar base renovable.....	37
Figura 10 Importar base de brokers	37
Figura 11 Función base renovable	38
Figura 12 Función metas de brokers	39
Figura 13 Archivo HTML para anexo.....	39
Figura 14 Función crear directorios	40
Figura 15 Función exportar anexo.....	41
Figura 16 Función exportar convenio	42
Figura 17 Función exportar convenio final.....	43
Figura 18 Importar librerías	44
Figura 19 Parámetros de fecha	45
Figura 20 Función crear carpeta	46
Figura 21 Almacenar querys en variables	47
Figura 22 Función exportar avance	48
Figura 23 Importar librerías	49
Figura 24 Parámetros de fecha	50
Figura 25 Función crear carpetas.....	51
Figura 26 Archivo HTML para avance PDF	52
Figura 27 Función exportar avance en PDF	54
Figura 28 Función envío masivo de correos.....	55
Figura 29 Portada reporte BI.....	56
Figura 30 Vista resumen reporte BI.....	57
Figura 31 Vista corredor reporte BI	58
Figura 32 Vista clientes nuevos reporte BI	59
Figura 33 Vista stock RRGG – TCA reporte BI.....	60
Figura 34 Vista stock VyS reporte BI.....	61

Figura 35 Correos masivos	62
Figura 36 Publicación reporte BI	63
Figura 37 Programar actualizaciones reporte BI.....	64
Figura 38 Reporte BI publicado.....	65
Figura 39 Tabla descargada de la vista resume	66

ÍNDICE DE TABLAS

Tabla 1 Participación de RIMAC en el mercado de seguros	21
Tabla 2 Esquema de comisiones	22
Tabla 3 Listado de tareas	26
Tabla 4 Listado de inputs.....	27
Tabla 5 Output seguimiento.....	29
Tabla 6 Output venta nueva.....	30
Tabla 7 Output stock.....	31

ÍNDICE DE ANEXOS

Anexo 1 Convenio programa de comisiones	71
Anexo 2 Avance programa de comisiones	77

I.EXPERIENCIA PROFESIONAL

1.1. Trayectoria profesional

RIMAC Seguros y Reaseguros (noviembre 2023 – actualidad)

Tribu Empresas

Cargo: Business Developer

Funciones:

- Automatizar procesos recurrentes que impliquen data.
- Elaborar querys según necesidad de negocio.
- Analizar y proponer indicadores clave requeridos por el negocio.
- Seguimiento a los OKRs del squad.

Los Portales (noviembre 2022 – octubre 2023)

Área de Excelencia Comercial

Cargo: Analista de Inteligencia de Negocios

Funciones:

- Desarrollar reportes comerciales y gerenciales en Power BI y QlikView.
- Analizar indicadores y proponer mejoras en los procesos comerciales.
- Implementar nuevas herramientas BI que ayuden a la gestión comercial.
- Participar en proyectos de implementación de sistemas y nuevas tecnologías.

Rom Outsourcing (enero 2022 – octubre 2022)

Área de Inteligencia de Negocios

Cargo: Analista de Inteligencia Comercial

Funciones:

- Desarrollar reportes comerciales en Power BI.
- Elaborar presentaciones gerenciales para la toma de decisiones.
- Crear y administrar base de datos.
- Implementar procesos ETL con SSIS para carga de datos masivos.

1.2. Contexto en el que se desarrolló el trabajo de suficiencia profesional

1.2.1. Descripción de la empresa donde desarrollo la experiencia profesional

RIMAC Seguros y Reaseguros es una compañía aseguradora y reaseguradora ubicada en Lima, Perú. Fue fundada en un inicio como Compañía de Seguros en 1896. En 1992, la Compañía Internacional de Seguros del Perú y la Compañía de Seguros RIMAC se fusionan, iniciando la marca RIMAC Seguros y Reaseguros. RIMAC es parte del grupo BRECA, conglomerado empresarial peruano que cuenta con más de 130 años de existencia, con operaciones en Perú y otros países de Latinoamérica.

1.2.2. Visión

“Ser una empresa socialmente responsable, centrada en el cliente y de clase mundial, líder nacional de seguros y salud “

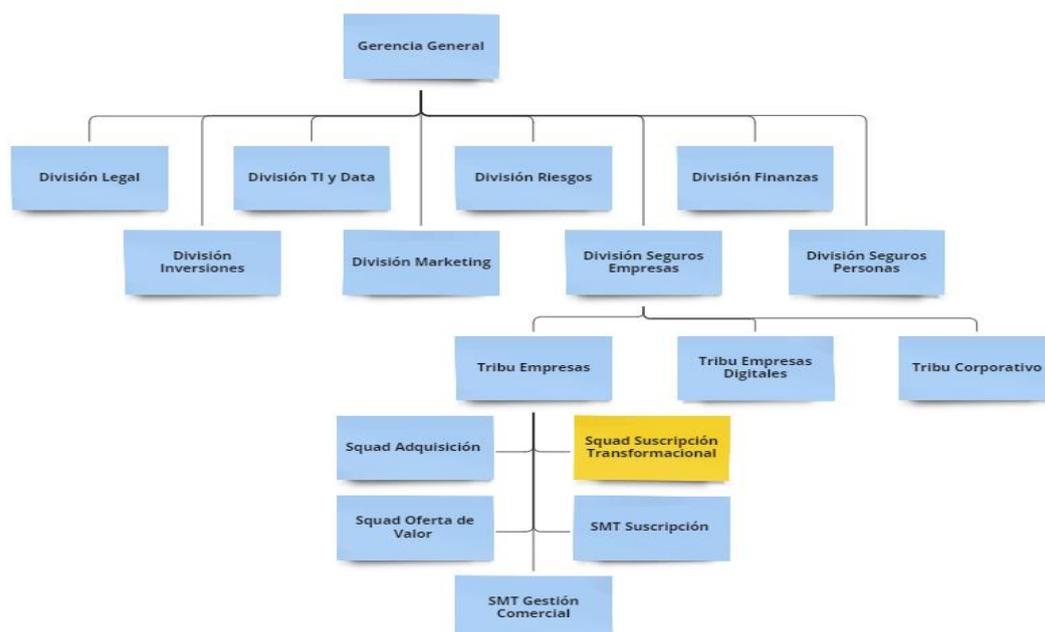
1.2.3. Misión

“Trabajar por un mundo con menos preocupaciones”

1.2.4. Organigrama

Figura 1

Organigrama resumido - RIMAC



Fuente: Elaboración propia

1.2.5. Descripción del cargo

El cargo que desempeño es el de Business Developer y pertenezco al squad de Suscripción Transformacional de la tribu empresas. A diferencia de los otros Business Developer de la tribu, mi posición se centra en disponibilizar data a las demás posiciones para la toma de decisiones efectiva o creación de nuevas oportunidades de negocio, ya sea para Business Developer, Product Owner, Analistas, Ejecutivos, etc. Además de esto, me encargo de la viabilidad, en un primer inicio, de las iniciativas que tengan algún componente tecnológico, como por ejemplo automatizaciones o creación de nuevos módulos en las plataformas que maneja RIMAC.

1.2.6. Servicios

RIMAC Seguros y Reaseguros brinda servicios de seguridad a los asegurados tanto para personas naturales como para personas jurídicas.

- Seguros Patrimoniales: Son todos los tipos de seguros que compensan algún daño patrimonial que el asegurado haya sufrido.
- Seguros de Salud: Seguros que brindan apoyo financiero para cubrir gran parte o la totalidad de gastos médicos.
- Seguros de Vida: Son tipos de seguros que protegen económicamente a los dependientes del asegurado en caso de fallecimiento.
- Seguros de Riesgos Laborales: Seguros destinados a la prevención y protección contra accidentes o enfermedades que se ocasionan en el lugar de trabajo.

1.3. Contribución a la experiencia profesional

1.3.1. Contribución en la solución de problemas que se hayan presentado durante su estancia en la empresa

Durante mis actividades en RIMAC, he podido contribuir con soluciones innovadoras a problemas de todo tipo, centrándome en funciones de extracción, transformación y visualización de datos para la toma de decisiones, además de automatizar procesos y tareas frecuentes ayudando así a la transformación tecnológica de la tribu, logrando reemplazar los procesos manuales por procesos automáticos.

1.3.2. Contribución en términos de las competencias y habilidades adquiridas durante su formación profesional

En RIMAC he podido participar en distintas iniciativas tecnológicas y no tecnológicas las cuales han enriquecido mi formación como profesional de ingeniería de computación y de sistemas, contribuyendo así a la mejora de competencias y habilidades propias de mi formación académica. Además, al estar participando en proyectos para áreas comerciales he podido potenciar y, en algunos casos, desarrollar habilidades sociales gracias a la interacción constante con ejecutivos de negocios.

1.4. Reflexión crítica de la experiencia

1.4.1. Aportes y beneficios obtenidos por el centro laboral

Mis aportes a la empresa se ven reflejados en los resultados obtenidos por las distintas decisiones en base a la data que pongo a disposición de los squads que conforman la tribu. Cada idea nueva que se presenta tiene que estar respaldada por los datos, es aquí donde mis habilidades y capacidades ayudan a dar forma a las propuestas que tienen como fin incrementar la participación de la empresa en el mercado, cumpliendo así las expectativas de mi rol. En lo que concierne a los beneficios obtenidos por el centro laboral, RIMAC me ofrece desempeñar mis actividades en un ambiente saludable y competitivo, el cual potencia mis habilidades técnicas y sociales al estar en contacto continuo con profesionales destacados en el rubro.

1.4.2. Planteamiento de mejoras propuestas a la organización

En el tiempo que tengo trabajando en la empresa, he planteado distintas mejoras a nivel de herramientas visuales las cuales se utilizan para la constante toma de decisiones, además de plantear y desarrollar soluciones que automatizan las tareas manuales, logrando así una disminución notable del trabajo operativo y que los usuarios se centren en idear nuevas formas de ingresos para la empresa.

II.INFORME DEL PROYECTO DE ESPECIALIDAD

2.1. Metodología

2.1.1. Descripción de la metodología

La división seguros empresas trabaja hace muchos años bajo un modelo ágil, en el cual las áreas se reestructuraron para dar paso a las tribus. En ese sentido, Mark Cruth (s.f) explica que las tribus son formadas por squads, estos son equipos interdisciplinarios y autónomos conformados por 6 a 12 miembros centrados en un área de funciones, cada squad tiene un objetivo único y la coordinación de estos forman una tribu.

En relación con el proyecto, al trabajar bajo este modelo ágil, los squads empleamos la metodología ágil SCRUMBAN para la gestión y desarrollo de actividades o proyectos que tengan como objetivo el crecimiento de la empresa. SCRUMBAN, según Atlassian (s.f), combina las mejores características de SCRUM y KANBAN en un marco híbrido de gestión de proyectos.

SCRUMBAN utiliza la estructura de SCRUM, respetando los eventos (sprint planning, daily, sprint review, sprint retrospective, sprint refinement) de este marco, pero incorporando el flujo de trabajo (pendiente, en curso, listo) visual de KANBAN, dando como resultado un método flexible para la gestión de proyectos de cualquier tamaño. Al ser un híbrido de dos metodologías, los artefactos varían de tal forma que al término de cada sprint no necesariamente habrá un incremento de producto que mostrar como si ocurre en SCRUM, es así como el tablero visual será un artefacto fundamental en esta metodología.

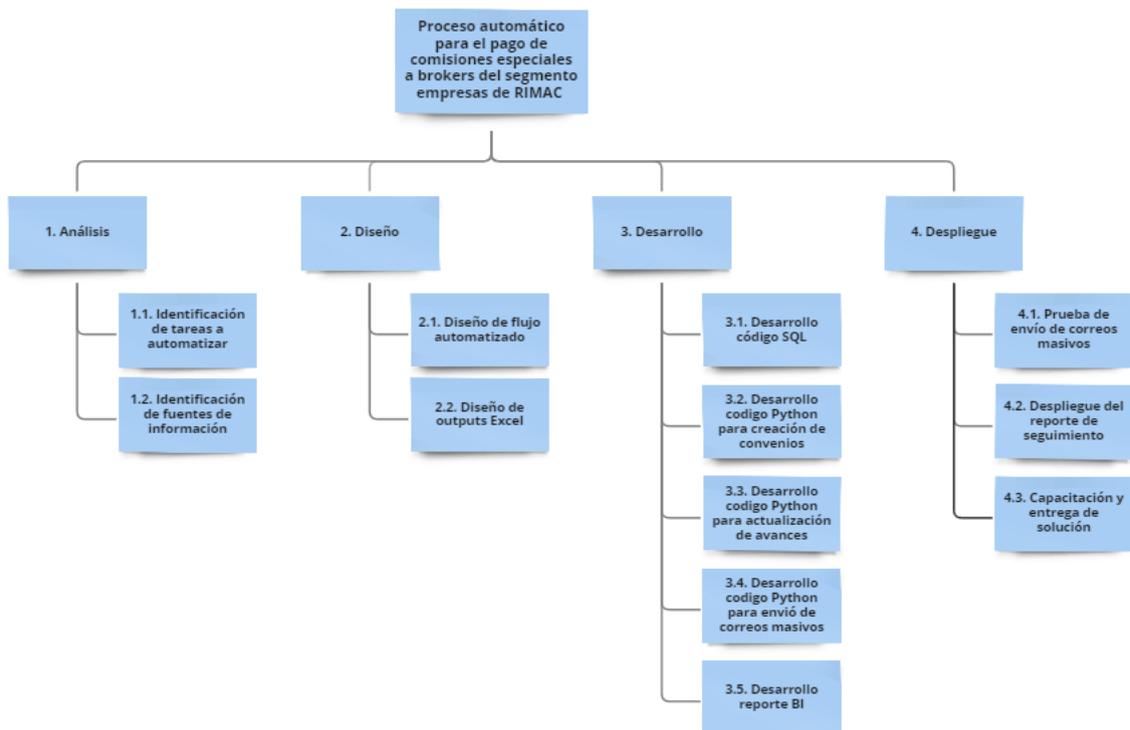
2.1.2. Organización de la metodología

2.1.2.1. EDT

Cada fase del EDT (Estructura de Descomposición del Trabajo) representa una HU (Historia de Usuario), a su vez, estas HU's pertenecen a una Épica que para este caso vendría a ser el proyecto. Cada HU tiene actividades que se realizarán dependiendo del cronograma.

Figura 2

EDT



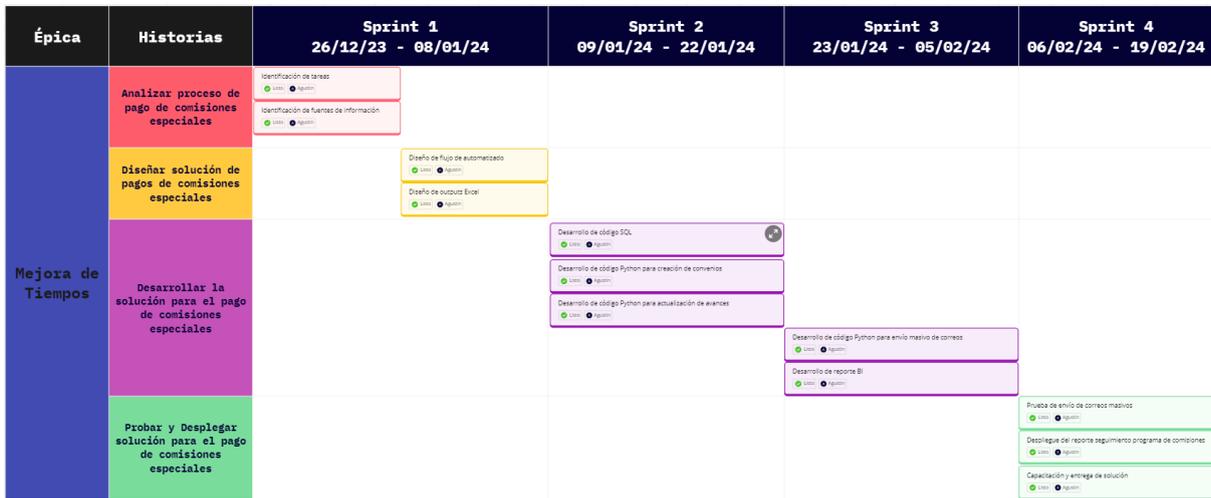
Fuente: Elaboración propia

2.1.2.2. Cronograma

Según el EDT, se distribuyeron las historias en 4 sprints, cada sprint tiene una duración de 2 semanas en las cuales se tendrán que trabajar las actividades planificadas. En total será 2 meses para el despliegue de la solución.

Figura 3

Cronograma de actividades



Fuente: Elaboración propia

2.2. Desarrollo

2.2.1. Situación problemática

2.2.1.1. Contextualización del desarrollo de la experiencia profesional

RIMAC es una empresa que lleva muchos años posicionada en los más alto del negocio de seguros y reaseguros, es por esto por lo que cada vez más brokers confían en nuestra experiencia y traen clientes nuevos para asegurar. Los brokers son un elemento esencial para que RIMAC pueda seguir creciendo como empresa y aumentar la participación de mercado.

Tabla 1

Participación de RIMAC en el mercado de seguros

	Empresas	Monto	Participación (%)	Porcentaje Acumulado
1	Rímac	1 979 445	28.06	28.06
2	Pacífico Seguros	1 639 651	23.24	51.30
3	Mapfre Perú	978 115	13.86	65.16
4	La Positiva Vida	502 662	7.13	72.29
5	La Positiva	401 621	5.69	77.98
6	Interseguro	417 248	5.91	83.90
7	Cardif	215 088	3.05	86.95
8	Protecta	208 863	2.96	89.91
9	Ohio National Vida	194 572	2.76	92.66
10	Vivir Seguros	148 430	2.10	94.77
11	Crece Seguros	102 352	1.45	96.22
12	Chubb Seguros	63 281	0.90	97.12
13	Secrex	53 434	0.76	97.87
14	AVLA Perú	50 285	0.71	98.59
15	Insur	39 237	0.56	99.14
16	Quálitas	35 178	0.50	99.64
17	Liberty Seguros	25 327	0.36	100.00
	Total	7 054 789	100	

Fuente: SBS, 2024

Es por esto por lo que para premiar a los brokers por traer clientes nuevos y cumplir con las renovaciones de sus carteras, nace este programa de comisiones especiales a brokers seleccionados.

A inicios de cada trimestre el squad de adquisición selecciona a los brokers que formarán parte de este programa, normalmente son brokers que tienen una prima alta por renovar y potencial para traer clientes nuevos. Una vez seleccionados se les envía un convenio, el cual contiene las bases del programa, su meta de stock (renovaciones), las cuentas a renovar, meta de venta nueva (clientes nuevos) y su esquema de bono. En las bases del programa se mencionan los productos que son considerados para el programa, todos estos productos son de Riesgos Patrimoniales del segmento empresas, los cuales están conformados por los ramos de RRGG (Riesgos Generales), TCA (Transportes, Cascos y Aviación) y VyS (Vehículos y SOAT).

Tabla 2

Esquema de comisiones

Esquema	Rango	Rango Orden	Prima	Bono
E - VIP	Rango 1	1	de 10,000.00 hasta 50,000.00	4 %
E - VIP	Rango 2	2	de 50,001.00 hasta 90,000.00	3 %
E - VIP	Rango 3	3	de 90,001.00 hasta 150,000.00	2 %
MyP - VIP	Rango 1	1	de 10,000.00 hasta 25,000.00	4 %
MyP - VIP	Rango 2	2	de 25,001.00 hasta 75,000.00	3 %
MyP - VIP	Rango 3	3	de 75,001.00 hasta 150,000.00	2 %
Base	Rango 1	1	de 10,000.00 hasta 150,000.00	2 %

Fuente: Elaboración propia

Cada broker cuenta con el apoyo de un ejecutivo comercial distribuido en alguno de los 5 equipos comerciales, por lo tanto, los brokers cuentan con una clasificación comercial de acuerdo con el ejecutivo.

Durante el trimestre, el broker tendrá que renovar las cuentas que se les ha indicado en el convenio, salvo algunas excepciones, se tendrá que renovar la totalidad de las cuentas asegurando el monto de la meta de stock. La otra condición que se deberá cumplir es la de traer al menos 1 cliente nuevo con un monto de prima igual o superior a los USD 10,000. Sobre esta última condición es que se pagará la comisión ya que el foco del programa es la venta nueva, mientras más clientes nuevos y mayor sea el monto de primas por venta nueva, mayor será la comisión que recibirá el broker al final del trimestre.

Cada cierre de trimestre se envía un correo a cada broker con los resultados en base a la producción que han obtenido a lo largo de los 3 meses, este correo contiene un PDF donde se detalla las cuentas que han renovado, las cuentas que faltaron renovar, las cuentas nuevas, el bono que le correspondería si gana y el porcentaje final que obtuvieron (meta real / meta total), si el porcentaje fuera igual o superior al 100% significará que ha ganado y lo que aparezca en la columna bono es la comisión que le corresponde, esta comisión debe ser pagada en un plazo máximo de 15 días calendario, por lo tanto no puede existir retrasos al momento de calcular las comisiones ya que si este proceso no llegara a estar del todo claro o no llegara a completarse, ocasionaría errores en los pagos y malestar en los brokers.

Para llegar al correo final tiene que pasar por un cálculo interno, el cual es parte fundamental del proceso de pago de comisiones. El cálculo de resultados se hace de forma manual y toma aproximadamente 1 semana para consolidar la información.

Dicho proceso tiene 2 resultados:

1. Correo al área de pagos con el listado de brokers ganadores con sus respectivas comisiones.
2. Correo a los brokers con los resultados del trimestre, sean o no ganadores del programa.

2.2.1.2. Descripción de la solicitud de trabajo a realizar

Al ser un proceso medianamente complejo y con una tasa alta de error humano por ser manual, se solicitó idear alguna forma de poder reducir el tiempo que se emplea actualmente para realizar el cálculo y los posibles errores humanos al momento de consolidar la información. Dicha solución debe de poder calcular avances cada dos semanas y ya no solo al cierre del trimestre, al igual que una opción para envío de correos masivos para comunicar estos avances y el resultado final, teniendo en cuenta la creación de los convenios y su almacenamiento en alguna ruta compartida. Además, se solicitó crear un reporte donde se muestren los estados de las cuentas de cada broker y el porcentaje de cumplimiento de estos.

2.2.1.3. Objetivos

Objetivo general

- Automatizar el proceso de pago de comisiones a brokers del segmento empresas para reducir el tiempo y los posibles errores que toma hacerlo de forma manual.

Objetivos específicos

- Desarrollar un script con Python que automatice el flujo de cálculo de comisiones para reducir el tiempo que toma hacerlo de forma manual.
- Desarrollar un reporte trimestral en Power BI para mostrar los resultados a las áreas involucradas en el proceso.
- Desarrollar un script con Python que envíe correos masivos para comunicar los resultados a los brokers.

2.2.1.4. Alcance

El alcance de la solución involucra diversas áreas y roles que, de manera directa o indirecta, se benefician con esta solución.

- Squad de adquisición: Encargados de comunicar los resultados y tener la visibilidad de las cuentas renovadas y cuentas ganadas.
- Área comercial: Encargados de dar soporte a los brokers en temas de renovación y cuentas ganadas.
- Área de pagos: Encargados de mandar las liquidaciones para efectuar el pago de las comisiones a los brokers ganadores.
- Brokers: Aliados estratégicos los cuales son premiados por cumplir con su meta de renovaciones y traer cuentas nuevas.

2.2.1.5. Limitaciones

- Acceso a la información: Se requiere tener acceso a GCP (Google Cloud Platform) para tener la información de la producción actualizada de las pólizas, además de los diversos data entry como: la segmentación actualizada de brokers con sus respectivos ejecutivos y el listado de correos para hacerles llegar el convenio, el avance semanal y el resultado final.
- Instalación de software: Se requiere la instalación de algunos programas y librerías para construir la solución, como: Visual Studio Code, Power BI, Python, Numpy, etc.
- Permisos de administrador: Al ser RIMAC una empresa grande tiene diversas restricciones, esto con el fin de salvaguardar los datos sensibles que se manejan. Al ejecutar procesos Python, nos aparecerá un flag que impedirá la ejecución de dicho proceso, por lo cual, se tiene que conversar con el área de ciberseguridad para que hagan una excepción y nos den permisos de administrador, logrando ejecutar el script sin problema.

- Reuniones con dueños de negocio: Al ser un proceso que actualmente lo trabaja otro squad, se requiere reuniones con los involucrados en el proceso para levantar el flujo de inicio a fin. Muchas veces se tendrá que esperar para poder concretar las reuniones.
- Conexión con Power BI: RIMAC tiene restricciones para integrar BigQuery (Servicio de GCP) y Power BI, esto por temas relacionados a la seguridad de la información; por lo cual, el script de Python tendrá que generar un archivo xlsx para ser consumido por Power BI y armar el reporte. No se contempló la opción de usar Looker Studio (Servicio de visualización de datos de Google) al ser una herramienta limitada, por más que se integre de forma nativa y que no haya restricciones de una conexión directa, por experiencias pasadas con usuarios similares, se decidió ir por Power BI.
- Tiempo de desarrollo: Se requiere una herramienta o herramientas que no tomen mucho tiempo en ser desarrolladas y que cumplan con las expectativas, el plazo máximo que se dio fue de 4 sprints que vendrían a ser 8 semanas.

2.2.1.6. Importancia y/o justificación

La solución no solo cuenta con beneficios en base a tiempos y posibles errores humanos, sino que ayudará a tener una mejor visibilidad de resultados trimestrales por brokers, los cuales mejoraran la toma de decisiones internas en base al reporte que se creará y que será compartido con las áreas involucradas con el fin de que todos miremos los mismos números al final de cada trimestre. Además, mejorará la comunicación que se tiene actualmente con los brokers, ya que estará automatizado y semanalmente les llegará un correo con los avances que han tenido, este correo estará sincronizado con los números del reporte, evitando falsas expectativas y malestar al momento de recibir los resultados finales.

2.2.2. Diseño de la solución

Para el desarrollo de la solución se consideró la siguiente estructura considerando las fases y las actividades a realizar para poder llegar a los entregables deseados y cumplir con los objetivos y expectativas.

2.2.2.1. Fase de análisis

En esta primera fase se realizó el levantamiento de las tareas que pasarían a ser automatizadas como también la identificación de las fuentes de información para ser posteriormente trabajadas en la fase de desarrollo.

2.2.2.1.1. Identificar tareas a automatizar

Se dio una reunión con el squad de adquisición, en esta reunión se describieron las tareas que se realizan cada trimestre para poder calcular las comisiones especiales de los brokers y la interacción que tienen con los stakeholders. En esta reunión se observó que no se contaba con un mapeo del proceso por lo que en un primer momento fue complicado llegar a entender dicho proceso y sus tareas. Debido a que se tenía una limitante de tiempo, se decidió no diagramar el proceso y considerarlo para después de la entrega del proyecto como una oportunidad de mejora para futuras integraciones. Al término de la reunión se consiguió un listado con todas las tareas que pasarían a ser automatizadas y cuales, por el momento, quedarían como manuales.

Tabla 3

Listado de tareas

Tarea	Estado	Periodicidad	Descripción
Creación de Convenios	Por automatizar	Cada inicio de trimestre	Crear los convenios a partir de la base renovable.
Envío de Convenios	Manual	Cada inicio de trimestre	Envío de convenios trimestrales a brokers.
Cálculo de Comisiones	Por automatizar	Cada fin de trimestre	Calculas las comisiones que le corresponde a cada broker a partir de la producción.
Actualización de Reporte BI	Por automatizar	Sin periodicidad	Actualización de reporte trimestral
Envío de Correos Masivos	Por automatizar	Cada fin de trimestre	Envío de avances a brokers

Fuente: Elaboración propia

2.2.2.1.2. Identificar fuentes de información

A partir de lo visto en la primera reunión con el squad de adquisición, se llevó a cabo una segunda reunión para identificar todos los inputs necesarios y sus orígenes. En esta reunión se detallaron todos los archivos necesarios para el proceso y los dueños de cada uno, algunos archivos eran propios del squad y tenían pleno control de estos, pero otros dependían de áreas que no necesariamente tenían algún interés en el proceso, sobre estos últimos se validó que estuvieran en alguna tabla del almacén de datos para extraerlos directamente del servicio. Una vez identificados, se clasificaron en dos posibles orígenes: archivos Excel o BigQuery. Al término de la reunión se consiguió un listado con todos los archivos con su clasificación y la ruta o nombre de la tabla.

Tabla 4

Listado de inputs

Input	Origen	Descripción
Base Renovable	Excel	Base con información de pólizas que se deberían renovar durante el trimestre.
Base Brokers	Excel	Base con información de brokers.
Base Producción Renovaciones	BigQuery	Base con la producción real de pólizas renovadas.
Base Producción Venta Nueva	BigQuery	Base con la producción real de pólizas nuevas.

Fuente: Elaboración propia

2.2.2.2. Fase de diseño

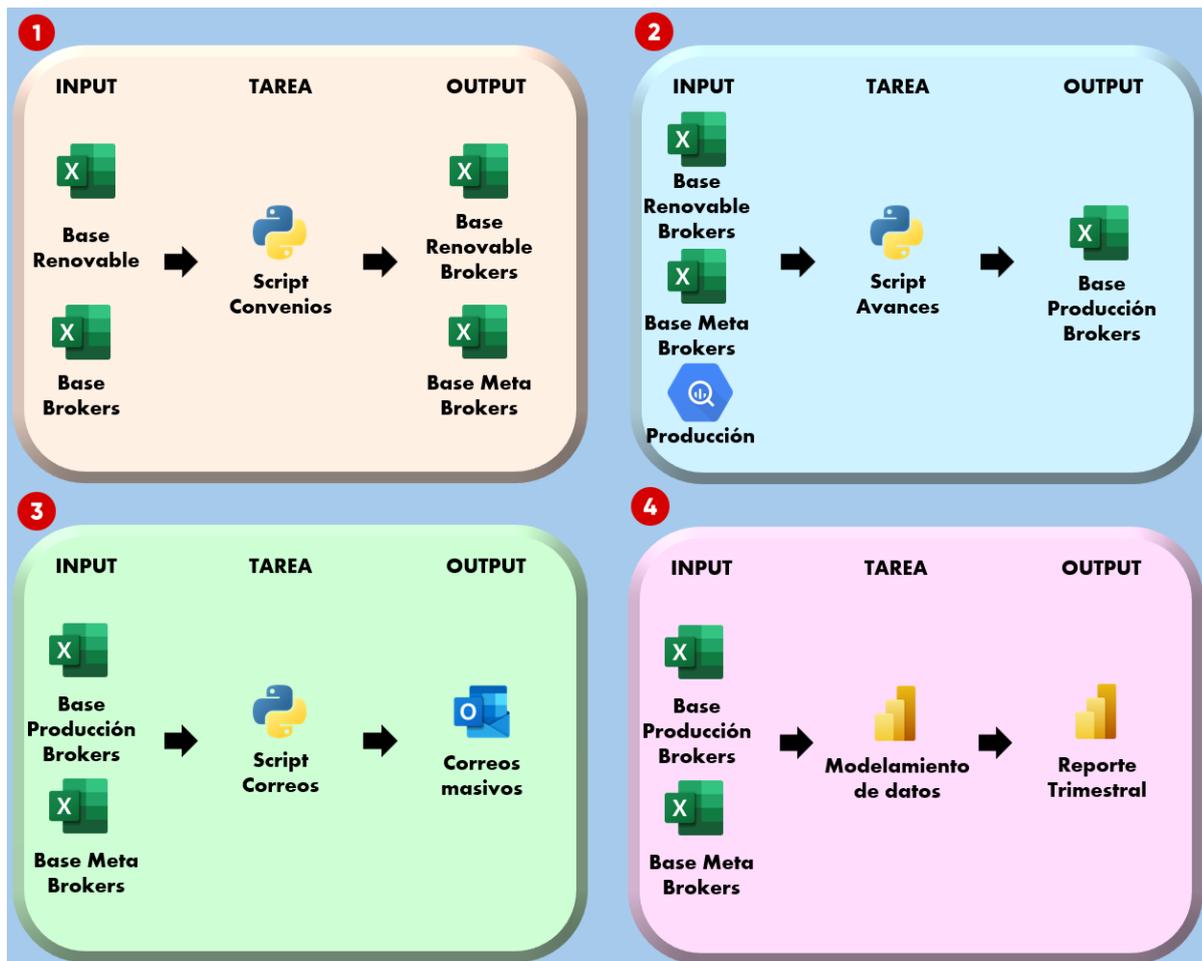
En esta fase se realizó el diseño de la solución a partir de lo levantado en la fase previa, con esto se diseñó un prototipo no funcional el cual integra todas las tareas

2.2.2.2.1. Diseño de flujo automatizado

Con las tareas identificadas en la fase anterior, se diseñó un flujo visual en el cual se observa el orden de las tareas a automatizar, además de los inputs necesarios para el proceso y los outputs que se pueden obtener al finalizar cada script.

Figura 4

Flujo automatizado



Fuente: Elaboración propia

2.2.2.2. *Diseño de outputs excel*

Teniendo el diseño macro del flujo de tareas, se diseñaron los outputs que se deberán obtener en cada etapa. Estos outputs son archivos Excel que servirán como inputs para ejecutar otra tarea, para actualizar el reporte BI o para enviar los correos masivos. Este diseño se basó en el Excel que manejaba el squad de Adquisición y, a partir de este, se hicieron los cambios correspondientes.

Tabla 5

Output seguimiento

Campo	Descripción
id_corredor	identificador unico del broker
corredor	nombre del broker
id_contratante	identificador unico de la cuenta
contratante	nombre de la cuenta
segmento	segmento empresarial de la cuenta
ramoagrupado	ramo (RRGG, TCA, VyS)
producto	producto de la poliza
numeropoliza	numero de poliza
iniciovigenciapoliza	inicio de la poliza
finvigenciapoliza	fin de la poliza
tipoventa	tipo de venta
ventanuevastock	subtipo de venta
primaneta	prima neta de la poliza
primacomercial	prima comercial de la poliza
esquema	esquema del broker
rango	rango de la cuenta
bono (%)	porcentaje según el ranto
bono (\$)	prima neta multiplicado por el porcentaje del rango

Fuente: Elaboración propia

Tabla 6*Output venta nueva*

Campo	Descripción
id_corredor	identificador unico del broker
corredor	nombre del broker
id_contratante	identificador unico de la cuenta
contratante	nombre de la cuenta
ramoagrupado	ramo (RRGG, TCA, VyS)
codigoproducto	codigo del producto
producto	producto de la poliza
numeropoliza_por_renovar	numero de poliza a renovar
iniciovigenciapoliza_por_renovar	inicio de vigencia de la poliza a renovar
finvigenciapoliza_por_renovar	fin de vigencia de la poliza a renovar
numeropoliza_renovacion	numero de poliza que se renovo
iniciovigenciapoliza_renovacion	inicio de vigencia de la poliza renovada
finvigenciapoliza_renovacion	fin de vigencia de la poliza renovada
primaneta_por_renovar	prima neta presupuestada
primaneta_renovacion	prima neta real
status	estado renovado o pendiente

Fuente: Elaboración propia

Tabla 7*Output stock*

Campo	Descripción
id_corredor	identificador unico del broker
corredor	nombre del broker
corredor_corto	nombre corto del broker
equipo_comercial	equipo al que pertenece el broker
tipo	tipo de esquema
esquema	esquema del broker
ejecutivo	nombre del ejecutivo
META_BR_RRGG	meta de RRGG
META_BR_TCA	meta de TCA
META_BR_VyS	meta de VyS
cuenta_foco_minima	meta de venta nueva
META_TOTAL_PREVIO	suma de todas las metas
META_TOTAL_FINAL	suma de todas las metas redondeada a miles
flag_br	flag si tien o no base renovable
DESPEGA_BR_RRGG	produccion real de RRGG
DESPEGA_BR_TCA	produccion real de TCA
DESPEGA_BR_VyS	produccion real de VyS
DESPEGA_FOCO_RRGG	produccion real de cuentas nuevas de RRGG
DESPEGA_FOCO_TCA	produccion real de cuentas nuevas de TCA
DESPEGA_FOCO_VyS	produccion real de cuentas nuevas de VyS
DESPEGA_TOTAL	suma de todas las producciones reales
despega_foco_rango_1	produccion real de cuentas nuevas de rango 1
despega_foco_rango_2	produccion real de cuentas nuevas de rango 2
despega_foco_rango_3	produccion real de cuentas nuevas de rango 3
despega_foco_total	suma de producciones reales de cuentas nuevas para todos los rangos
cuentas_foco_rango_1	numero de cuentas nuevas de rango 1
cuentas_foco_rango_2	numero de cuentas nuevas de rango 2
cuentas_foco_rango_3	numero de cuentas nuevas de rango 3
cuentas_foco_total	suma de cuentas nuevas
avance(%)	division de produccion total y meta total
cond_br	condicion si cumple o no con todas sus renovaciones
cond_foco	condicion si cumple o no con cuentas nuevas
cond_meta_total	condicion si cumple o no con su meta total
cond_final	condicion final
bono	bono correspondiente si su condicion final es 'Ganador'

Fuente: Elaboración propia

2.2.2.3. Fase de desarrollo

En esta fase se desarrollaron todas las herramientas necesarias para cumplir con las expectativas del proyecto según el diseño del flujo de tareas, es aquí donde empezamos con la parte técnica del proyecto.

2.2.2.3.1. Desarrollo de código SQL

Para extraer la producción real de los brokers se crearon varios queries a partir de la tabla de producción, la cual pertenece a la capa analítica del modelo de datos en bigquery (servicio de almacenamiento de datos de Google Cloud Platform), en esta tabla encontramos información de todas las pólizas emitidas por RIMAC en todos los riesgos, segmentos, ramos y productos. Para llegar al resultado esperado tomamos como referencia el feedback obtenido en las reuniones con el área encargada (Fase de Análisis) y transformamos esto en sentencias SQL filtrando solo la información que vamos a utilizar. Al ser información sensible, solo se mostrará uno de los queries trabajados con los nombres de columnas alterados.

Query pólizas nuevo y cross

Hacemos un SELECT con los campos que necesitamos extraer de la tabla PRODUCCION, con el comando GROUP BY agrupamos los campos y con el comando SUM sumamos los valores prima_comercial y derecho_emision además de otros cálculos necesarios, con el comando WHERE filtramos solo los registros que coincidan con las características requeridas para realizar el cálculo. Con el comando WITH almacenamos de forma temporal el resultado del primer del primer SELECT en la variable primas, con el fin de utilizarlo en un query posterior.

Figura 5

Ejemplo query: 1ra parte

```
WITH primas AS
(
  SELECT
    codigo_mes, canal,
    id_corredor, des_corredor,
    equipo_comercial, id_contratante,
    des_contratante, segmento_comercial,
    segmento_poliza, ramo,
    producto, id_poliza,
    numer_opoliza, inicio_vigencia_poliza,
    fin_vigencia_poliza, facultativa,
    fronting, tipoventa,
    SUM(prima_comercial) AS prima_comercial_ft_primas,
    SUM(derecho_emision) AS derecho_emision_ft_primas,
    (SUM(prima_comercial) - SUM(derecho_emision)) AS prima_neta_ft_primas,
    SAFE_DIVIDE((SUM(prima_comercial) - SUM(derecho_emision)), SUM(prima_comercial))
AS pct_prima_neta
  FROM `TABLA PRODUCCION`
  WHERE codigo_mes IN (202401, 202402, 202403)
  AND segmento_poliza = 'EMPRESAS'
  AND ramo IN ('RRGG', 'TCA', 'VyS')
  AND tipoventa IN ('NUEVO', 'CROSS')
  AND segmento_comercial IN ('GRAN EMPRESA', 'EMPRESARIAL')
  AND facultativa = 'N'
  AND fronting = 'N'
  AND cod_producto IN (1111, 2222, 3333, 4444, 5555, 6666, 7777, 8888)
  GROUP BY 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18
),
```

Fuente: Elaboración propia

Llamamos a la tabla temporal primas para agregar campos calculados en base a las lógicas requeridas. De la misma forma que el SELECT anterior, almacenamos en una tabla temporal el resultado del SELECT para su uso posterior.

Figura 6

Ejemplo query: 2da parte

```
primas_1 AS
(
SELECT
codigo_mes,
CONCAT(SUBSTRING(CAST(codigo_mes AS STRING), 1, 4), '-', SUBSTRING(CAST(codigo_mes
AS STRING), 5, 2)) AS periodoreal,
canal, id_corredor,
des_corredor, equipo_comercial,
id_contratante, des_contratante,
segmento_comercial, segmento_poliza,
ramo, producto,
id_poliza, numero_poliza,
inicio_vigencia_poliza, fin_vigencia_poliza,
facultativa, fronting,
tipoventa,
CASE
WHEN ramo ='Vys' THEN
CASE
WHEN tipoventa='LIBERACION' OR tipoventa='RETENCION' OR tipoventa='ANUL
MOROSIDAD' OR tipoventa='OTROS' THEN 'STOCK'
ELSE 'VENTA NUEVA'
END
ELSE
CASE
WHEN tipoventa='LIBERACION' OR tipoventa='RETENCION' OR tipoventa='ANUL
MOROSIDAD' OR tipoventa='FLOTANTE' THEN 'STOCK'
ELSE 'VENTA NUEVA'
END
END ventanuevastock,
prima_comercial_ft_primas,
derecho_emision_ft_primas,
prima_neta_ft_primas,
pct_prima_neta
FROM primas
ORDER BY id_contratante DESC, codigo_mes ASC, numero_poliza ASC
)
```

Fuente: Elaboración propia

Hacemos un último SELECT a la tabla temporal primas_1 con el fin de dar forma a los datos según las necesidades de la solución.

Figura 7

Ejemplo query: 3ra parte

```
SELECT
canal, id_corredor,
des_corredor, equipo_comercial,
id_contratante, des_contratante,
segmento_comercial, segmento_poliza,
ARRAY_AGG(ramo) AS ramo,
ARRAY_AGG(producto) AS producto,
ARRAY_AGG(id_poliza) AS id_poliza,
ARRAY_AGG(numero_poliza) AS numero_poliza,
ARRAY_AGG(inicio_vigencia_poliza) AS inicio_vigencia_poliza,
ARRAY_AGG(CASE
    WHEN fin_vigencia_poliza IS NOT NULL THEN fin_vigencia_poliza
    ELSE 'NULL'
END) AS fin_vigencia_poliza,
facultativa,
fronting,
ARRAY_AGG(tipoventa) AS tipoventa,
ARRAY_AGG(ventanuevastock) AS ventanuevastock,
SUM(prima_neta_ft_primas) AS prima_neta_ft_primas,
SUM(derecho_emision_ft_primas) AS derecho_emision_ft_primas,
SUM(prima_comercial_ft_primas) AS prima_comercial_ft_primas,
SUM(pct_prima_neta) AS pct_prima_neta_1,
SAFE_DIVIDE((SUM(prima_comercial_ft_primas) - SUM(derecho_emision_ft_primas)),
SUM(prima_comercial_ft_primas)) AS pct_prima_neta_2,
ARRAY_AGG(codigo_mes) AS codigo_mes,
ARRAY_AGG(periodoreal) AS periodoreal
FROM primas_1
GROUP BY
canal, id_corredor,
des_corredor, equipo_comercial,
id_contratante, des_contratante,
segmento_comercial, segmento_poliza,
facultativa, fronting;
```

Fuente: Elaboración propia

2.2.2.3.2. Desarrollo de código Python para creación de convenios

El primer documento que se les envía a los brokers es el convenio del programa, este documento PDF contiene: las bases, las pólizas de los clientes a renovar durante el trimestre y la meta total compuesta por la meta de stock (renovaciones) y la meta de venta nueva (clientes nuevos). Para las renovaciones utilizamos la BR (Base Renovable) que se nos compartió como un archivo xlsx, siendo este el primer input que trabajaremos con python. Esta base contiene todas las pólizas a renovar durante el año. Al ser información delicada, solo se mostrará código que no involucre datos o información sensible.

Importar librerías

El primer paso será importar las librerías necesarias. En la figura 8 observamos las librerías a utilizar para lograr la creación de convenios.

Figura 8

Importar librerías

```
import pandas as pd
import numpy as np
import os
import fnmatch
import shutil
from jinja2 import Environment, FileSystemLoader
import pdfkit
from docxtpl import DocxTemplate
from PyPDF2 import PdfWriter, PdfReader
from PyPDF2 import PdfMerger
# pd.options.display.float_format = '$ {:.2f}'.format
from itables import init_notebook_mode, show
import itables.options as opt
init_notebook_mode(all_interactive=True)
opt.maxBytes = 2**20
opt.maxRows = 0
```

Fuente: Elaboración propia

Importar base renovable

En la figura 9 observamos el código para importar el input base renovable desde la ruta indicada y el almacenamiento en variable, además le asignamos valores a las variables year y quarter. La ruta está alterada por seguridad.

Figura 9

Importar base renovable

```
year = '2024'
quarter = 'q1'

path_1 = r'd:\programa_comisiones'

file_1 = 'br' + '_' + year + '.xlsx'
path_and_file_1 = os.path.join(path_1, year, 'base_renovable', file_1)

br_ppto_total_v0 = pd.read_excel(path_and_file_1,\
                                sheet_name = 'data')
```

Fuente: Elaboración propia

Importar base de brokers

En la figura 10 observamos el código para importar el input brokers desde la ruta indicada y el almacenamiento en variable. Este input contiene datos de los brokers como su ejecutivo y equipo comercial. La ruta está alterada por seguridad.

Figura 10

Importar base de brokers

```
path_2 = r'd:\programa_comisiones'

file_2 = 'corredores' + '_' + year + '_' + quarter + '.xlsx'
path_and_file_2 = os.path.join(path_2, year, 'corredores', file_2)

df_v1 = pd.read_excel(path_and_file_2,\
                      sheet_name = 'dictionary')
```

Fuente: Elaboración propia

Función base renovable

Una vez importada la base renovable se hacen filtros para dejar las pólizas que coincidan con lo establecido en el programa; además, solo se consideran las pólizas correspondientes a los brokers que forman parte del programa y que pertenezcan al trimestre en cuestión. Los filtros que se aplican están dentro de la función `br_programa_forma_1`. En la figura 11 observamos el código para llamar a la función y los parámetros requeridos, los cuales son: la base renovable almacenada en la variable `br_ppto_2024_total_v0` y el listado de corredores participantes almacenado en la variable `corredorid_programa_list`. El resultado de la función es almacenado en la variable `br_programa_f1`. Al ser información sensible, solo se muestra el llamado a la función más no el código.

Figura 11

Función base renovable

```
br_programa_f1 = br_programa_forma_1(df = br_ppto_total_v0,\
                                   corredores = corredorid_programa_list)
```

Fuente: Elaboración propia

Función metas de brokers

Con la base renovable filtrada y almacenada en la variable `br_programa_f1` se arman las metas de stock por ramo (RRGG, TCA, VyS) y se le agrega 10,000 USD que será la meta de venta nueva, la suma de estas dos variables forma la meta total que será redondeado a miles; además, se agregará una columna llamada `flag_br` para diferenciar a los brokers que tienen base renovable y así crear su anexo en los pasos posteriores. En la figura 12 observamos el código para llamar a la función `br_programa_forma_2` y el parámetro requerido. El resultado de la función es almacenado en la variable `br_programa_f2`. Al ser información sensible, solo se muestra el llamado a la función más no el código.

Figura 12

Función metas de brokers

```
br_programa_f2 = br_programa_forma_2(df = br_programa_f1)
```

Fuente: Elaboración propia

Archivo HTML para anexo

Un primer output que conseguiremos será un anexo que contendrá una tabla con el listado de pólizas a renovar en formato PDF; si el broker no tiene pólizas a renovar, no tendrá dicho anexo. Para generar el anexo se creó un archivo HTML que será la plantilla que recibirá los datos de la variable `br_programa_f1` de la figura 13, es decir, recibirá la base renovable filtrada por broker.

Figura 13

Archivo HTML para anexo

```
<!DOCTYPE html>
<html>

<head lang="en">
  <meta charset="UTF-8">
  <title>{{ title }}</title>
</head>
  <h2> Anexo 2 </h2>
<body>
<table>
  {{ br_corredor_forma_1 }}
</table>
</body>
</html>
```

Fuente: Elaboración propia

Función crear directorios

Se creó una función llamada `directorio_equipo` (figura 14) que crea 5 directorios con los nombres de los equipos comerciales con los que se trabaja a partir del directorio raíz que se coloca, en estas carpetas se guardarán los anexos.

Figura 14

Función crear directorios

```
def directorio_equipo(parent_directory = str, directory_equipo = str):  
  
    directory_path = os.path.join(parent_directory, directory_equipo)  
  
    if os.path.exists(directory_path):  
        shutil.rmtree(directory_path)  
        os.mkdir(directory_path)  
    else:  
        os.mkdir(directory_path)  
  
    return print("Directory '%s' created" %directory_path)  
  
path_3 = r'd:\programa_comisiones\convenios'  
  
pdf_path_anexo_2 = os.path.join(path_3, year, quarter, 'pdf_anexo_2')  
print(pdf_path_anexo_2)  
  
directorio_equipo(parent_directory = pdf_path_anexo_2, directory_equipo =  
'equipo_1')  
directorio_equipo(parent_directory = pdf_path_anexo_2, directory_equipo =  
'equipo_2')  
directorio_equipo(parent_directory = pdf_path_anexo_2, directory_equipo =  
'equipo_3')  
directorio_equipo(parent_directory = pdf_path_anexo_2, directory_equipo =  
'equipo_4')  
directorio_equipo(parent_directory = pdf_path_anexo_2, directory_equipo =  
'equipo_5')
```

Fuente: Elaboración propia

Función exportar anexo

Una vez creado el archivo HTML y las carpetas de los equipos, pasamos a crear y exportar los anexos. En la figura 15 observamos el código para llamar a la función `exportar_anexo_2_pdf` y los parámetros requeridos, los cuales son: la base renovable filtrada, las metas de los brokers, la ruta del archivo HTML y la ruta raíz de las carpetas de los equipos comerciales. Para saber que broker tiene base renovable y en que carpeta guardar el PDF, usamos las columnas de `flag_br` y `equipo_comercial` de la tabla almacenada en la variable `br_programa_f2`.

Figura 15

Función exportar anexo

```
exportar_anexo_2_pdf(df1 = br_programa_f1,\n                    df2 = br_programa_f2,\n                    html_input_path = html_path,\n                    pdf_parent_directory = pdf_path_anexo_2)
```

Fuente: Elaboración propia

Función exportar convenio

Con los anexos exportados, creamos una función para importar un archivo Word (anexo _) que contiene la plantilla del convenio, pasándole los valores requeridos que se identifican por estar entre dos corchetes para crear un convenio por broker. Hacemos uso de la función crear directorios para crear una carpeta similar a la de anexos y usamos la librería DocxTemplate que nos da la opción de manipular una plantilla en formato Word para pasarle los datos obtenidos por la función metas de broker y que se encuentran almacenados en la variable br_programa_f2. Una vez ejecutada la función exportamos los convenios. Al ser información sensible, en la figura 16 solo se muestra el llamado a la función más no el código.

Figura 16

Función exportar convenio

```
pdf_path_convenio = os.path.join(path_3, year, quarter, 'pdf_convenio')
print(pdf_path_convenio)

directorio_equipo(parent_directory = pdf_path_convenio, directory_equipo =
'equipo_1')
directorio_equipo(parent_directory = pdf_path_convenio, directory_equipo =
'equipo_2')
directorio_equipo(parent_directory = pdf_path_convenio, directory_equipo =
'equipo_3')
directorio_equipo(parent_directory = pdf_path_convenio, directory_equipo =
'equipo_4')
directorio_equipo(parent_directory = pdf_path_convenio, directory_equipo =
'equipo_5')

exportar_convenio(df2 = br_programa_f2,\
                  word_input_path = word_path,\
                  pdf_parent_directory = pdf_path_convenio)
```

Fuente: Elaboración propia

Función exportar convenio final

Finalizando el script para crear convenios, utilizamos nuevamente la función crear directorios para crear una carpeta que reciba los convenios combinados con los anexos en un mismo archivo PDF por broker. Creamos la función `convenios_listos Equipos`, la cual recibirá la ruta de los convenios, la ruta de los anexos, la ruta previamente creada y los datos almacenados en la variable `br_programa_f2`, este último nos servirá para extraer el equipo comercial de cada broker y guardar los convenios en las carpetas creadas. Al ser información sensible, en la figura 17 solo se muestra el llamado a la función más no el código.

Figura 17

Función exportar convenio final

```
pdf_path_listos = os.path.join(path_3, year, quarter, 'pdf_listos')
print(pdf_path_convenio)

directorio_equipo(parent_directory = pdf_path_listos, directory_equipo =
'equipo_1')
directorio_equipo(parent_directory = pdf_path_listos, directory_equipo =
'equipo_2')
directorio_equipo(parent_directory = pdf_path_listos, directory_equipo =
'equipo_3')
directorio_equipo(parent_directory = pdf_path_listos, directory_equipo =
'equipo_4')
directorio_equipo(parent_directory = pdf_path_listos, directory_equipo =
'equipo_5')

convenios_listos_equipos(pdf_path_convenios = pdf_path_convenios,\
pdf_path_anexo = pdf_path_anexo,\
pdf_path_listos_parent_directory = pdf_path_listos,\
df2 = br_programa_f2)
```

Fuente: Elaboración propia

2.2.2.3.3. *Desarrollo de código Python para actualización de avances*

Iniciando la construcción del script, se importaron las librerías y se mapearon las rutas donde estarían alojados los Excel, además de agregar los queries para ser ejecutados y guardar los resultados en variables. Se crearon las funciones para realizar los cálculos y segmentar la producción según el esquema. Se implementaron las funciones para exportar los resultados consolidados en un archivo Excel, este archivo será el input para el reporte BI. El script para el envío masivo de correos con los resultados se trabajó en otra opción ya que esta parte del proceso no necesariamente se realiza semanalmente como si se realiza el avance de resultados.

Importar librerías

Como primer paso importamos las librerías. En la figura 18 observamos las librerías a utilizar para lograr la actualización de avances.

Figura 18

Importar librerías

```
import pandas as pd
import numpy as np
from numpy import nan
import os
import pandas_gbq
from tqdm import tqdm
from datetime import date, datetime, timedelta

pd.options.display.float_format = '$ {:.2f}'.format
```

Fuente: Elaboración propia

Parámetros de fecha

Cada vez que se vaya a realizar la actualización de avances se tendrá que actualizar las variables de fecha según el formato. En la figura 19 observamos las variables que luego se utilizarán para crear un archivo excel con el fin de conocer cuántas veces se han actualizado los avances. En la variable year colocamos el año actual, en quarter el trimestre actual, en file_yearmonth se coloca el año y mes actual, en fecha_descarga se coloca la fecha actual y en fecha_actualizacion se coloca la última fecha en que se actualizó la producción. La variable file_descarga_actualizacion será el nombre que tomará el archivo excel para identificarlo.

Figura 19

Parámetros de fecha

```
year = '2024'  
quarter = 'q2'  
file_yearmonth = '202405'  
fecha_descarga = '20240521'  
fecha_actualizacion = '20240514'  
  
file_descarga_actualizacion = fecha_descarga + "_" + fecha_actualizacion
```

Fuente: Elaboración propia

Función crear carpeta

Desarrollamos una función para crear una carpeta donde se guardarán todas las actualizaciones de los avances, pasándole como parámetros: la ruta del directorio y el file_yearmonth. Cada vez que se ejecute este script creará una carpeta en el directorio nombrándola con el año y mes insertados en los parámetros de fecha. En la figura 20 observamos la función y como llamamos a esta para la creación de la carpeta.

Figura 20

Función crear carpeta

```
def directorio(parent_directory = str, file_yearmonth = str

    directory_path = os.path.join(parent_directory, file_yearmonth)
    os.mkdir(directory_path)

    return print("Directory '%s' created" %directory_path)

path = r'd:\programa_comisiones\avance_resultados'

parent_directory_programa_produccion = os.path.join(path, year, quarter)

directorio(parent_directory = parent_directory_programa_produccion,
file_yearmonth = file_yearmonth)
```

Fuente: Elaboración propia

Almacenar resultados de queries en variables

Previamente se desarrollaron sentencias SQL las cuales daban como resultado la producción de los brokers para stock y venta nueva, para ejecutar estas sentencias y almacenarlas en variables utilizamos la librería `panda_gbq`. En la figura 21 observamos los parámetros que se le tiene que pasar a la función, los cuales son: la sentencia SQL y el `project_id`. Para visualizar el avance de la operación agregamos una barra visual. Lo que va dentro de la variable `sql` es la sentencia SQL de la figura 7 y en la variable `project_id` se coloca el id del proyecto GCP (Google Cloud Platform) en el cual se está trabajando. Al ser información confidencial no se muestra el `project_id`.

Figura 21

Almacenar queries en variables

```
sql = """
SELECT
*
FROM TABLA
"""

project_id = "NOMBRE_PROYECTO_GCP"

query_v1 = pandas_gbq.read_gbq(sql, \
                                project_id=project_id, \
                                progress_bar_type='tqdm_notebook')
```

Fuente: Elaboración propia

Función exportar avance

Previamente a la función para exportar el avance, se realizaron transformaciones con la librería `pandas`, la cual permite manipular datasets de una forma sencilla. Estas transformaciones se dan ya que los datos obtenidos por los queries tienen que cruzarse con archivos excel. Finalizando la transformación de datos, utilizamos el método `ExcelWriter`, de la librería `pandas`, para crear un archivo Excel que contenga las lógicas que se encuentran almacenadas en las variables observadas. Al ser información sensible solo se describe lo trabajado previamente a la ejecución de la función `exportar avance`, en la figura 22

observamos el código para exportar, en un archivo Excel, todas las tablas que se han trabajado en este script y que se encuentran almacenadas en variables.

Figura 22

Función exportar avance

```
path = r'd:\programa_comisiones\avances_resultados'

file = 'dashboard_programa_seguimiento' + '_' + file_descarga_actualizacion +
'.xlsx'

path_and_file = os.path.join(path, year, quarter, file_yearmonth, file)

with pd.ExcelWriter(path_and_file) as writer:
    seguimiento_v2.to_excel(writer, sheet_name='seguimiento',
index=False)
    seguimiento_resumen_v2.to_excel(writer,
sheet_name='seguimiento_resumen', index=False)
    seguimiento_unpivot_v2.to_excel(writer,
sheet_name='seguimiento_unpivot', index=False)
    stock_por_renovar_stock_renovacion_rrgg_tca_considerar_v3.to_excel(wri
ter, sheet_name='stock_rrgg_tca', index=False)
    stock_por_renovar_stock_renovacion_vys_considerar_v1.to_excel(writer,
sheet_name='stock_vys', index=False)
    foco_pivot_table_v3.to_excel(writer, sheet_name='foco_pivot_table',
index=False)
    gcp_nuevo_cross_ts_cuentas_productos_v10.to_excel(writer,
sheet_name='nuevo_cross_c_p', index=False)
    gcp_nuevo_cross_ts_cuentas_productos_polizas_v10.to_excel(writer,
sheet_name='nuevo_cross_c_p_po', index=False)
    stock_por_renovar_stock_renovacion_rrgg_tca_v9.to_excel(writer,
sheet_name='stock_rrgg_tca_comp', index=False)
    stock_por_renovar_stock_renovacion_vys_v5.to_excel(writer,
sheet_name='stock_vys_comp', index=False)
    foco_pivot_table_mes_e_ge_ra_v2.to_excel(writer,
sheet_name='foco_e_ge_ra', index=True)
    foco_pivot_table_mes_ts_ra_v2.to_excel(writer,
sheet_name='foco_ts_ra', index=True)
```

Fuente: Elaboración propia

2.2.2.3.4. Desarrollo de código Python para envío masivo de correos

Para comunicar los avances y los resultados finales se utiliza el ultimo output que se obtiene de la ejecución del script para la actualización de avances. Tomando este archivo como referencia, se crea un documento PDF por cada broker con los avances obtenidos hasta el corte y enviamos los correos de forma masiva.

Importar librerías

Empezamos importando las librerías necesarias. En la figura 23 observamos las librerías a utilizar para lograr el envío masivo de correos.

Figura 23

Importar librerías

```
import os
import pandas as pd
import html
from html.parser import HTMLParser
import win32com.client as win32
from jinja2 import Environment, FileSystemLoader
import pdfkit
import sutil
```

Fuente: Elaboración propia

Parámetros de fecha

Al igual que el código de actualización de avances, cada vez que se vaya a realizar el envío masivo se tendrá que actualizar las variables de fecha. En la figura 24 observamos las variables que luego se utilizarán para nombrar a la carpeta donde se guardarán los PDF con los avances hasta la fecha de corte. En la variable year colocamos el año actual, en quarter el trimestre actual, en file_yearmonth se coloca el año y mes actual, en fecha_inicio_campaña_f1 la fecha de inicio del trimestre actual, en fecha_descarga_campaña_f1 se coloca la fecha actual y por último en fecha_corte_campaña_f1 se coloca la última fecha de actualización de la producción. La variable file_descarga_corte será el nombre que tomará el archivo excel para identificarlo.

Figura 24

Parámetros de fecha

```
year = '2024'
quarter = 'q2'
file_yearmonth = '202405'
fecha_inicio_campaña_f1 = '20240401'
fecha_descarga_campaña_f1 = '20240521'
fecha_corte_campaña_f1 = '20240514'

fecha_corte_campaña_f2 = fecha_corte_campaña_f1[6:8] + '/' +
fecha_corte_campaña_f1[4:6] + '/' + fecha_corte_campaña_f1[0:4]
file_descarga_corte = fecha_descarga_campaña_f1 + "_" + fecha_corte_campaña_f1
```

Fuente: Elaboración propia

Función crear carpeta

Desarrollamos una función para crear una carpeta tomando la fecha_descarga y fecha_corte como nombre de la carpeta donde se guardarán los archivos PDF con los avances o resultados finales de los brokers participantes. Cada vez que se envíen los avances se creará una carpeta nueva, la cual guardará todos los archivos con el fin de llevar un control sobre lo que se está enviando en cada correo. En la figura 25 observamos la función y como llamamos a esta para la creación de la carpeta.

Figura 25

Función crear carpetas

```
def directorio(parent_directory = str, file_yearmonth = str,
file_descarga_corte = str):

    directory_path = os.path.join(parent_directory, file_yearmonth,
file_descarga_corte)

    if os.path.exists(directory_path):
        shutil.rmtree(directory_path)
        os.mkdir(directory_path)
    else:
        os.mkdir(directory_path)

    return print("Directory '%s' created" %directory_path)

path = r'd:\programa_comisiones\avance_correo_corredores'

parent_directory = os.path.join(path, year, quarter)
print(parent_directory)

directorio(parent_directory = parent_directory, \
    file_yearmonth = file_yearmonth, \
    file_descarga_corte = file_descarga_corte)
```

Fuente: Elaboración propia

Archivo HTML para avance en PDF

El output que conseguiremos será un PDF por broker con sus respectivos avances, para esto se necesita crear archivos HTML, los cuales serán las plantillas que recibirán los datos obtenidos del output de la actualización de avances. Se crearon un total de 3 plantillas ya que hay existen casuísticas propias del negocio, como por ejemplo los brokers que no tienen base renovable. En la figura 26 vemos la estructura HTML de la plantilla y las variables que serán reemplazadas por los datos de los brokers. En el anexo _ se observa un PDF con los datos alterados por ser información sensible.

Figura 26

Archivo HTML para avance PDF

```
<!DOCTYPE html>
<html>
<body>
<header class="my-header">
  <hgroup>
    <div class="left">
      <h1 class="site-title">
        </h1>
      </div>
      <p class="right">AVANCE CAMPA&Ntilde;A</p>
      <p class="date">{{year_quarter}}</p>
    </hgroup>
  </header>
  <p class="saludo_p1">&#161;Hola {{corredor}}!</p>
  <p class="saludo_p2">Queremos contarte todo lo que necesitas
saber sobre el avance de tu Campa&ntilde;a
      <strong>del {{fecha_inicio_campaña}} al
cierre de {{fecha_corte_campaña}}</strong></p>
  <hr class="linea_horizontal_2">
  <h3>Resumen de Avance: </h3>
  <table class="tabla_resumen">
    <tr class="fila_resumen">
      <th>Avance Total (%)</th>
      <th>Avance Total</th>
      <th>Meta Total</th>
      <th>Clientes Nuevos (#)</th>
    </tr>
    <tr>
      <th>{{porcentaje_avance_total}}</th>
      <th>{{prima_avance_total}}</th>
      <th>{{meta_total}}</th>
```

```

        <th>{{cuentas_foco_total}}</th>
    </tr>
</table>
    <link rel="stylesheet"
href="d:\avance_templates\style.css">
</table>
<br>
<table class="tabla_resumen">
    <tr class="fila_resumen">
        <th>Ramo</th>
        <th>Stock</th>
        <th>Nuevo</th>
    </tr>
    <tr>
        <th>Riesgos Generales</th>
        <th>{{prima_br_rrgg}}</th>
        <th>{{prima_foco_rrgg}}</th>
    </tr>
    <tr>
        <th>Transportes</th>
        <th>{{prima_br_trans}}</th>
        <th>{{prima_foco_trans}}</th>
    </tr>
    <tr>
        <th>Veh&iacute;culos</th>
        <th>{{prima_br_vehi}}</th>
        <th>{{prima_foco_vehi}}</th>
    </tr>
    <link rel="stylesheet"
href="d:\avance_templates\style.css">
</table>
<div class="container_cuentas_foco">
    <h2>Detalle de Clientes Nuevos</h2>
    {{ cuentas_foco_table }}
    <link rel="stylesheet"
href="d:\avance_templates\style.css">
</div>
<div class="container_renovaciones">
    <p style="page-break-before: always" ></p>
    <h2>Detalle de Renovaciones del Stock</h2>
    <h2>Riesgos Generales - Transportes</h2>
    {{ br_anexo2_rrgg_tca_table }}
    <link rel="stylesheet"

```

```
href="d:\avance_templates\style.css">
  </div>
  <link rel="stylesheet" href="d:\avance_templates\style.css">
</body>
</html>
```

Fuente: Elaboración propia

Función exportar avance en PDF

Para exportar los avances en formato PDF se creó la función `exportar_avance_programa_pdf`. Esta función recibirá como parámetros la ruta donde se encuentra el Excel que obtuvimos como output al finalizar el proceso de actualización de avance y la ruta donde está la plantilla HTML, además de los parámetros que creamos en el inicio para guardarlos en la carpeta correspondiente a las fechas establecidas. En la figura 27 se observa cómo se llama a la función con sus respectivos parámetros. Al ser información sensible, no se mostrará el código de la función.

Figura 27

Función exportar avance en PDF

```
exportar_avance_programa_pdf(excel_seguimiento = seguimiento_path,\
                             html_input_path = html_path,\
                             pdf_output_path = pdf_path,\
                             year = year,\
                             quarter = quarter,\
                             inicio_campaña_f1 = fecha_inicio_campaña_f1,\
                             corte_campaña_f1 = fecha_corte_campaña_f1,\
                             descarga_campaña_f1 = fecha_descarga_campaña_f1)
```

Fuente: Elaboración propia

Función envío masivo de correos

Finalmente se crea la función de envío masivo de correos para enviar los avances a cada broker a partir del directorio de correos que fue mapeado como input en la fase de análisis. La función tomará la ruta del excel donde se encuentran los correos de los contactos de cada broker, los correos de sus respectivos ejecutivos y los correos de los encargados de negocio; además de la ruta de los PDF previamente exportados para ser adjuntados en los correos. Al ser información sensible, la figura 28 solo muestra el llamado de la función.

Figura 28

Función envío masivo de correos

```
envio_masivo_correo(excel_correos = correos_path,\n                    pdf_output_path = pdf_path,\n                    corredorid_excluidos_list = corredorid_excluidos,\n                    year = year,\n                    quarter = quarter,\n                    inicio_campaña_f1 = fecha_inicio_campaña_f1,\n                    corte_campaña_f1 = fecha_corte_campaña_f1,\n                    descarga_campaña_f1 = fecha_descarga_campaña_f1)
```

Fuente: Elaboración propia

2.2.2.3.5. Desarrollo de reporte BI

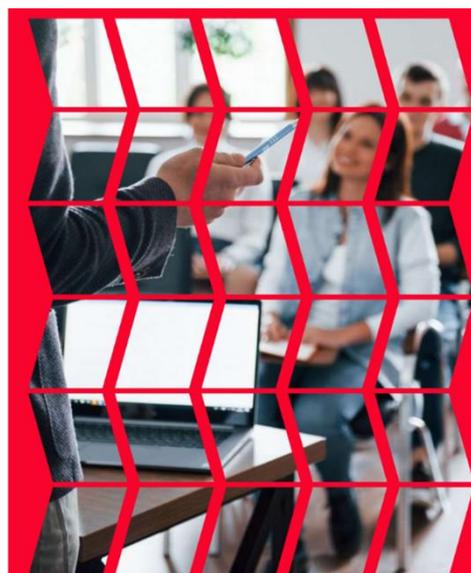
Para la visualización de los avances y resultados finales del programa de comisiones especiales, se desarrolló un reporte en Power BI con distintas vistas a partir del output generado proceso en Python.

Portada:

Se creó una portada mostrando la última fecha de actualización de los datos junto con 5 botones que sirven para navegar entre las vistas del reporte.

Figura 29

Portada reporte BI



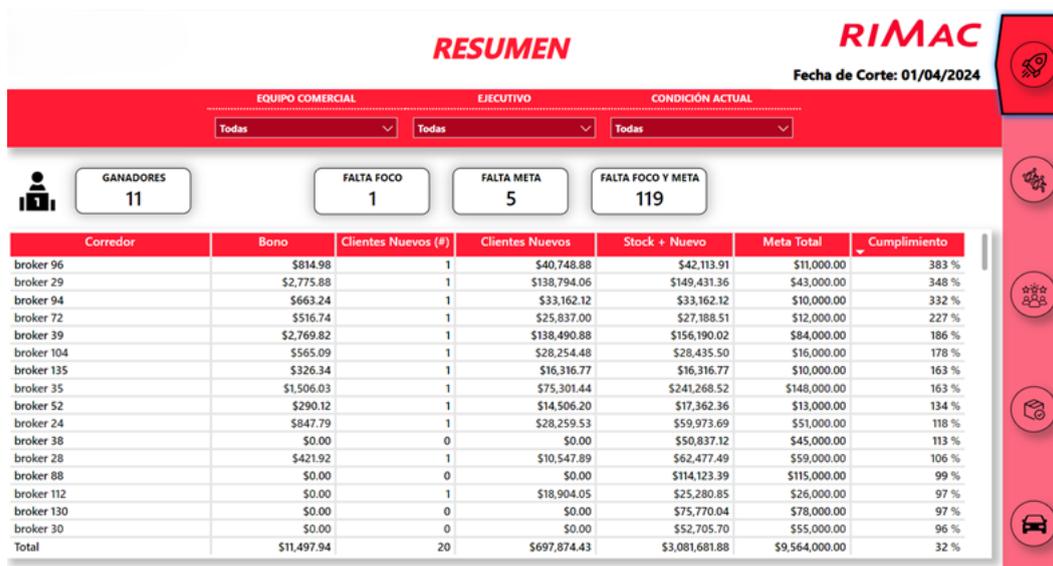
Fuente: Elaboración propia

Vista resumen:

La vista resumen muestra los resultados de los brokers a lo largo del trimestre, aquí se visualizará una tabla con el bono que le corresponde a los ganadores, el número de clientes nuevos, el monto en dólares por clientes nuevos, el monto real en dólares, la meta total en dólares y el porcentaje de cumplimiento. Además, esta vista cuenta con 3 filtros: equipo comercial, ejecutivo y condición actual; la condición actual muestra 4 posibles estados (1. Ganador, 2. Falta FOCO, 3. Falta META, 4. Falta FOCO y META) los cuales están también en forma de tarjetas visuales en esta misma vista, los cuales muestran la cantidad de brokers por estado.

Figura 30

Vista resumen reporte BI



Fuente: Elaboración propia

Vista corredor:

A diferencia de la vista resumen, la cual muestra los resultados totales, aquí veremos más detalle de la producción del broker, las condiciones de cumplimiento de meta y clientes nuevos, el esquema de bono, la prima neta por rango y el bono que tiene hasta la fecha. Esta vista solo cuenta con el filtro de corredor ya que el foco es mostrar el detalle por broker.

Figura 31

Vista corredor reporte BI



Fuente: Elaboración propia

Vista clientes nuevos:

La vista de clientes nuevos al igual que la vista por corredor y las siguientes vistas solo cuenta con el filtro de broker ya que el foco sigue siendo mostrar detalle por broker. En esta vista veremos el detalle de todos los clientes nuevos que el broker haya conseguido en el trimestre e información adicional como el número de póliza, el tipo de producto de la póliza, la prima neta, el rango del cliente, el porcentaje de bono dependiendo del rango del cliente y el bono que recibirá por cada cliente nuevo en base a la prima neta.

Figura 32

Vista clientes nuevos reporte BI



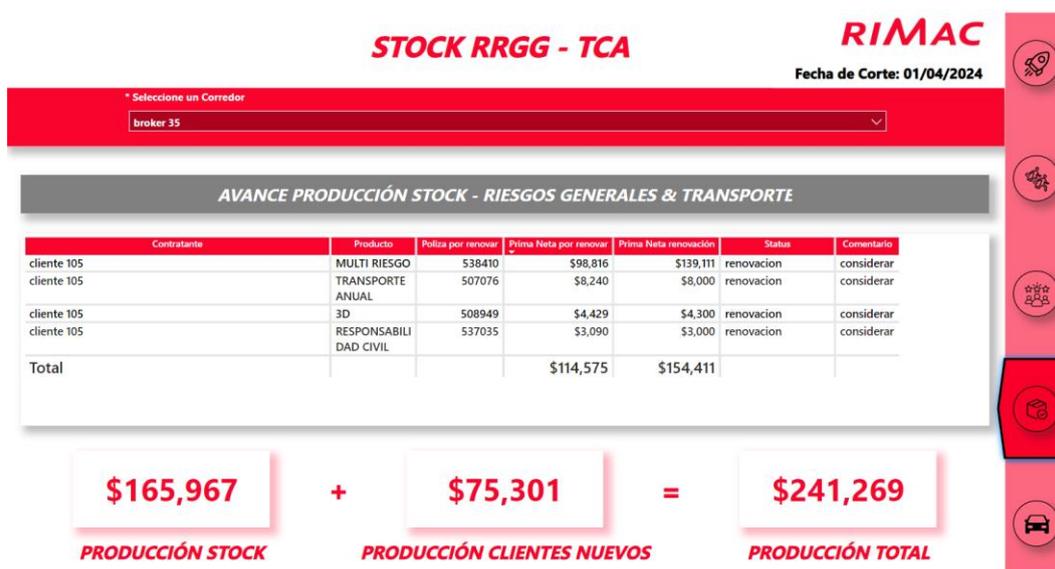
Fuente: Elaboración propia

Vista stock RRGG – TCA:

La vista de stock RRGG – TCA muestra los clientes renovados o por renovar de los ramos RRGG Y TCA según el broker que se filtre. Al igual que la vista anterior, muestra el producto, la póliza, el status y los montos de prima neta por renovar y post renovación. En esta tabla visual se muestran 2 columnas de prima neta, la primera hace referencia al monto por el cual la póliza se debería de renovar y la segunda columna hace referencia al monto real de la prima después que ocurre la renovación, con esto se puede detectar si hubo un upsell (incremento en el monto de la prima neta) o downsell (decremento en el monto de la prima neta).

Figura 33

Vista stock RRGG – TCA reporte BI



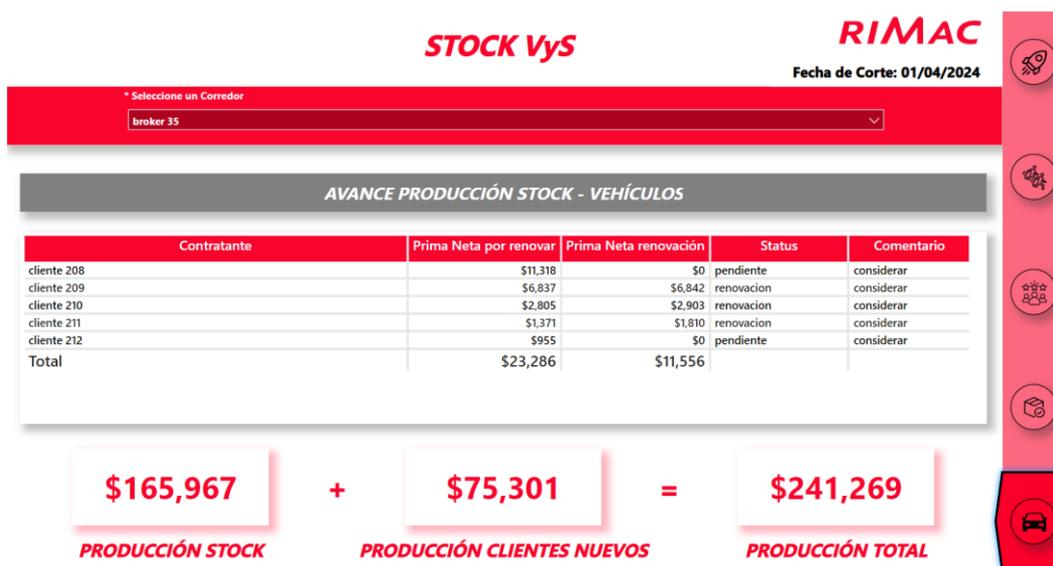
Fuente: Elaboración propia

Vista stock VyS:

La vista de stock VyS muestra los clientes renovados o por renovar del ramo VyS según el broker que se filtre. Tiene el mismo contenido que la vista anterior pero orientado al ramo en cuestión. En este ramo ocurre una casuística que hay que tener en cuenta, la cual es que las renovaciones de estas pólizas suelen cambiar el número de póliza, dificultando la trazabilidad de esta, a diferencia de los ramos de RRGG y TCA que mantiene el mismo número de póliza cuando se renueva. Es por esto por lo que la tabla visual solo contendrá las columnas de monto de prima por renovar, monto de prima real y el status de los clientes.

Figura 34

Vista stock VyS reporte BI



Fuente: Elaboración propia

2.2.2.4. Fase de pruebas y despliegue

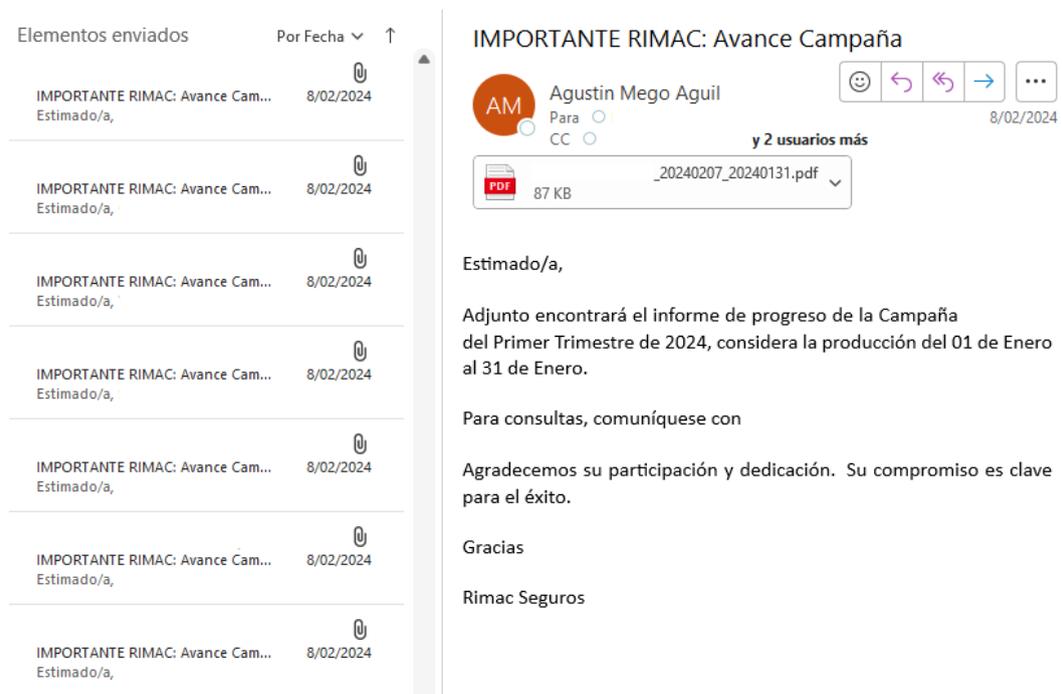
En esta última fase se realizaron las pruebas de las herramientas desarrolladas en la fase previa. Una vez terminada las pruebas se procedió con el despliegue del reporte en el workspace de la tribu en el servicio de Power BI para que los distintos usuarios implicados en este proceso tengan acceso y puedan autoservirse.

2.2.2.4.1. Prueba de envío de correos masivos

Para validar que la herramienta desarrollada para el envío de correos masivos funciona como se planificó, procedemos a realizar una prueba enviando el avance del programa. En la figura 35 observamos la bandeja de elementos enviados con los correos del programa y el PDF del avance adjuntado (**anexo 2**).

Figura 35

Correos masivos



Fuente: Elaboración propia

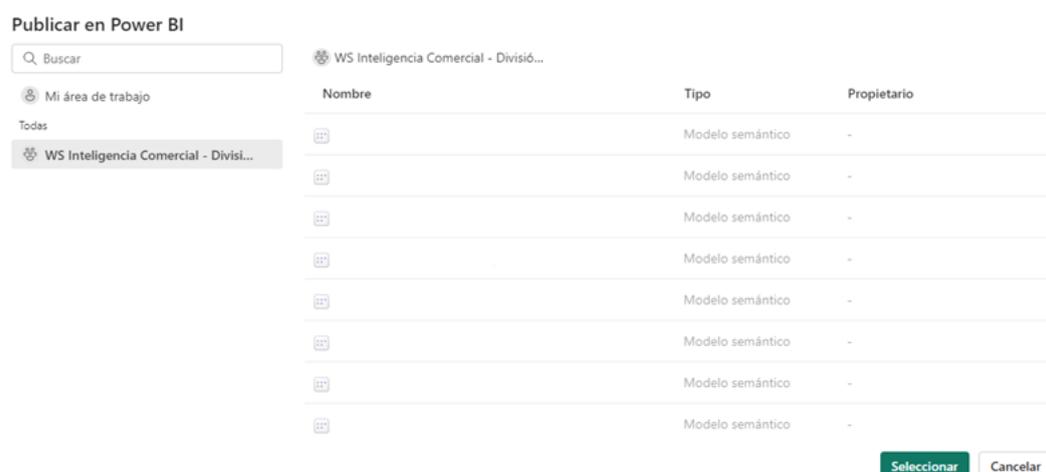
2.2.2.4.2. Despliegue del reporte seguimiento programa de comisiones

Una vez validadas las cifras del reporte y que coincidan con los números del squad de adquisición se procede a publicar el reporte en el workspace de la tribu, esto para que todos los usuarios que tengan acceso al área de trabajo puedan visualizarlo y hacer uso de la información brindada sin la necesidad de enviar correos solicitando avances o resultados del programa como tradicionalmente se estaba haciendo. Además, se programó para que se actualice todos los días a las 10 de la mañana para, de ser necesario, contar con el tiempo suficiente de ejecución del script de avances.

En la figura 36 observamos la publicación del reporte en el workspace de la tribu desde Power BI Desktop.

Figura 36

Publicación reporte BI



Fuente: Elaboración propia

En la figura 37 observamos la programación de actualización del reporte en el workspace de la tribu.

Figura 37

Programar actualizaciones reporte BI

Actualizar

Configurar una programación de actualización

Defina una programación de actualización de datos para importar datos del origen de datos al modelo semántico. [Más información](#)

Activar

Frecuencia de actualización

Diaria

Zona horaria

(UTC-05:00) Bogotá, Lima, Quito

Hora

10 00 a. m

[Agregar otra hora](#)

Fuente: Elaboración propia

En la figura 38 observamos el reporte publicado y disponible para todos los usuarios con acceso al workspace de la tribu.

Figura 38

Reporte BI publicado



Fuente: Elaboración propia

2.2.2.4.3. Capacitación y entrega de solución

Finalizadas las pruebas y el despliegue del reporte de seguimiento, se capacitó al squad de adquisición para que puedan ejecutar los scripts y obtener los cálculos de las comisiones de los brokers. Además, se tuvo una última capacitación que reunió al squad de adquisición, al área comercial y al área de pagos para mostrarles el reporte BI, donde se visualizó las funcionalidades y como ellos podrán descargar las tablas. Una vez terminadas las capacitaciones, se procedió a entregar las herramientas desarrolladas al squad de adquisición para que estos se conviertan en los owners del flujo automatizado.

Figura 39

Tabla descargada de la vista resume

seguimiento_resumen[CORREDOR]	[v_Avance_Mes_Medida]	[Sumbono]	[Sumcuentas_foco_total]	[SumFoco_US]	[SumDespega_US]	[Summeta_total]
broker 1	3.828537273	814.9776	1	40748.88	42113.91	11000
broker 2	3.484543256	2775.8812	1	138794.06	149835.36	43000
broker 3	3.367	673.4	2	33670	33670	10000
broker 4	3.316212	663.2424	1	33162.12	33162.12	10000
broker 5	2.295877333	1809.965	1	90498.25	137752.64	60000
broker 6	2.295709167	516.74	1	25837	27548.51	12000
broker 7	2.147202857	2769.8176	1	138490.88	180365.04	84000
broker 8	1.777218678	565.089577	1	28254.47885	28435.49885	16000
broker 9	1.632097917	1506.028835	1	75301.44176	241550.4918	148000
broker 10	1.631676531	326.3353063	1	16316.76531	16316.76531	10000
broker 11	1.627581931	626.806004	2	31340.3002	42317.1302	26000
broker 12	1.374935987	847.7857603	1	28259.52534	70121.73534	51000
broker 13	1.35337329	1116.776965	2	27919.42412	79849.02412	59000
broker 14	1.306163077	290.124	1	14506.2	16980.12	13000
broker 15	1.141250222	0	0	0	51356.26	45000
broker 16	1.134357179	300	1	15000	44239.93	39000
broker 17	1.125284	0	0	0	61890.62	55000
broker 18	1.11743665	0	0	0	50284.64923	45000
broker 19	1.07965	0	0	0	63699.35	59000
broker 20	1.001222348	0	0	0	115140.57	115000

Fuente: Elaboración propia

2.3. CONCLUSIONES Y RECOMENDACIONES

2.3.1. Conclusiones

- Respecto al primer objetivo específico, se logró desarrollar un script con Python que automatizó el cálculo de comisiones especiales como se observa en el punto **3.2.3.3. Desarrollo de código Python para actualización de avances** y en la **figura 18, figura 19, figura 20, figura 21 y figura 22**, pasando de tomar aproximadamente 3 días a 5 minutos y eliminando los errores humanos al calcular los resultados de cada broker.

Además, al estar automatizado, no se tendrá que esperar al final del trimestre para revisar los números obtenidos ya que se podrá ejecutar el flujo las veces que el squad crea conveniente para tener avances del programa, respetando la solicitud de tener avances cada dos semanas y logrando un mejor control del programa.

- Respecto al segundo objetivo específico, se cumplió con la creación del reporte trimestral en Power BI como se muestra en el punto **3.2.3.5. Desarrollo de reporte BI** y en la **figura 38**, el cual muestra de una manera sencilla y fácil de entender los resultados obtenidos por cada broker participante del programa durante los meses en cuestión. Este reporte se publicó en el workspace de la tribu con el fin de ser compartido con el área comercial y el área de pagos, logrando el soporte y pago puntual del bono a los ganadores. El reporte se programó para que se actualice automáticamente todos los días en la mañana a partir del output que se obtiene como resultado de la ejecución del flujo para la actualización de avances según el punto **3.2.3.3. Desarrollo de código Python para actualización de avances**.

- Respecto al tercer objetivo específico, se logró desarrollar un script con Python que automatizó el envío masivo de correos para la comunicación de avances y resultados finales a los brokers como se observa en el punto **3.2.4.1. Prueba de envío de correos masivos** y en la **figura 35**. Cada correo tiene adjunto un PDF personalizado (**anexo 2**), el cual muestra el resultado obtenido por cada corredor, coincidiendo con los números del reporte BI (**figura 38**). Esta forma de envío masivo no solo ayudó a tener una mejor comunicación de cara a los brokers, sino que también ayudó a reducir la carga operativa del squad de adquisición, logrando tener a la mano una herramienta potente y fácil de utilizar al momento de enviar los resultados.

- Respecto al objetivo general, se cumplió con creces la automatización del proceso de pago de comisiones a brokers del segmento empresas, logrando una reducción en el tiempo empleado y eliminando los errores humanos al momento de calcular los resultados como se comenta en la primera conclusión. Además, se fortaleció el proceso dando otras herramientas que complementan la solución, como son el caso del reporte BI para la visualización de resultados y el script para comunicar masivamente los avances a los brokers.

2.3.2. Recomendaciones

- Se recomienda integrar todas las automatizaciones de tareas o procesos a alguna plataforma o sistema que la empresa tenga con el fin que se pueda mantener en el tiempo y optimizar en la medida que se necesite con las mejores prácticas del desarrollo de software. En un primer momento cuando se planifica el desarrollo de la automatización, puede que no se plantee la integración de esta en alguna plataforma debido a varios factores como tiempos o costos, pero en la medida que la herramienta crezca y se requiera desarrollar nuevas funcionalidades, se debería considerar en alguna plataforma o sistema.

- Se recomienda automatizar la tarea de envío de convenios siguiendo el mismo formato y estructura que el envío de avances, el script se puede reutilizar y adaptar dependiendo del requerimiento del squad, esto ayudará a reducir la carga operativa y mejorar la eficiencia al momento de comunicar las bases del programa a los brokers participantes.

- Se recomienda capacitar a los usuarios no tecnológicos sobre nuevas tecnologías para que al momento de recibir herramientas que ayuden a automatizar tareas operativas, no se les sea complicado entender su funcionamiento y puedan interactuar con estas de una forma natural, reduciendo el desgaste generado por capacitaciones específicas de herramientas. Además, generará un enfoque tecnológico en el cual los usuarios propondrán mejoras en los procesos en base al conocimiento que estos tienen sobre los procesos en el que están involucrados.

REFERENCIAS

Mark Cruth. (s.f). *Descubre el modelo de Spotify*. Recuperado de:

<https://www.atlassian.com/es/agile/agile-at-scale/spotify>

Atlassian. (s.f). *Scrumban: domina dos metodologías ágiles*. Recuperado de:

<https://www.atlassian.com/es/agile/project-management/scrumban#:~:text=Scrumban%20es%20una%20metodolog%C3%A1gil,la%20complejidad%20de%20un%20proyecto.>

RIMAC. (s.f). *Sobre nosotros*. Recuperado de:

<https://www.rimac.com/nosotros>

ANEXOS

Anexo 1

Convenio programa de comisiones



Señores:
BROKER 1

Programa

¡Hola!

En el 2024 tenemos un gran reto por cumplir y eres uno de nuestros principales socios estratégicos para crecer juntos.

Este programa se ha diseñado para incentivar, a través de un Bono Trimestral la venta nueva de pólizas en Productos Patrimoniales.

Todos los detalles podrás encontrarlos en este documento.

Cordialmente,

1) DEFINICIONES

1. Cliente Nuevo:

Cliente que cumpla una sumatoria de primas netas anualizadas de pólizas nuevas emitidas entre el 01 de enero de 2024 hasta el 31 de marzo de 2024 dentro del rango de US\$ 10,000 a US\$ 150,000.

2. Meta Programa:

Es el monto de prima mínima requerida a la que debes llegar en el trimestre para calificar al Bono. Esta Meta también está conformada por las pólizas en el stock de RIMAC en los productos que forman parte del programa y cuya fecha de vencimiento (renovación) está entre el 01 de abril de 2024 hasta el 30 de junio de 2024 (Anexo 2).

2) CONDICIONES GENERALES

1. La producción del programa se mide en base a la prima neta anualizada RIMAC de pólizas emitidas y con vigencia entre el 01 de enero de 2024 hasta el 31 de marzo de 2024.

2. Participan los siguientes productos de RIMAC:

- Producto 1
- Producto 2
- Producto 3
- Producto 4
- Producto 5
- Producto 6
- Producto 7
- Producto 8
- Producto 9
- Producto 10
- Producto 11
- Producto 12
- Producto 13
- Producto 14
- Producto 15

3. Para ganar el Bono del Programa debes llegar a la Meta Programa:

US\$ 59,000
Meta Programa

4. Para el cálculo del monto a pagar, se considerará la prima neta anualizada de las pólizas nuevas de Clientes Nuevos que ingresen en el trimestre (del 01 de enero de 2024 hasta el 31 de marzo de 2024) con una prima entre US\$ 10,000 a US\$ 150,000.

5. El programa no está sujeto al cumplimiento de siniestralidad.

6. Solo califican las pólizas de Clientes Nuevos o Clientes del Stock que se mantengan vigentes por más de tres meses y se encuentren al día en sus pagos.
7. Se excluyen pólizas ~~fronting~~, coaseguro recibido, pólizas facultativas, cuentas del Estado y producción en Canales No Tradicionales (CNT).
8. El corredor deberá dar cumplimiento a las disposiciones del Reglamento de Supervisión y Control de los Corredores y Auxiliares de Seguros, aprobado por la Resolución SBS N° 809-2019, así como sus normas modificatorias y sustitutorias, y a su vez cumplirá con presentar la Declaración Jurada que como Anexo N° 01 forma parte del presente Convenio.

3) CÁLCULO DEL BONO

El cálculo del Monto a pagar se aplicará sobre la sumatoria de la **prima neta anualizada de cada Cliente Nuevo**:

$$\text{Prima de cada Cliente Nuevo} \times 2\% = \text{BONO DE CADA CLIENTE}$$

El monto por pagar será la sumatoria de todos los bonos de cada Cliente Nuevo.

RIMAC SEGUROS se ha comprometido al fiel cumplimiento de los Diez Principios establecidos en el Pacto Mundial de las Naciones Unidas, desarrollado en el Foro Económico Mundial, en Davos, Suiza, al cual se ha adherido.

En tal sentido, RIMAC Seguros considera fundamental el respeto por los derechos humanos, derechos laborales, la lucha contra la corrupción y la protección del medio ambiente, en virtud de lo cual se compromete a contribuir con la sociedad peruana implementando políticas de responsabilidad social.

Dentro de sus políticas frente a sus colaboradores, RIMAC Seguros desarrolla prácticas de empleo buscando los más altos estándares legales y éticos. Bajo esta consideración, RIMAC Seguros no contrata ni aprueba la contratación ilegal de personas que no cumplan con la edad mínima requerida por la ley peruana o la realización de trabajos forzados o bajo amenaza de que el personal sea sancionado, respeta la libertad de asociación y reconoce el derecho a la negociación colectiva de sus trabajadores, y rechaza toda discriminación en el empleo.

En caso de que RIMAC Seguros tome conocimiento y confirme que usted o la organización a la cual pertenece ha contratado o utiliza personal que no cumple con la edad mínima establecida por la ley peruana, o dicho personal realiza trabajo forzado o bajo amenaza, RIMAC Seguros adoptará las medidas legales correspondientes, perdiendo cualquier beneficio plasmado en el presente convenio.

Declaro haber leído detalladamente las condiciones del programa y estar de acuerdo con las metas y bonos establecidos. Adicionalmente, las condiciones generales y la política de responsabilidad social señalados en este documento.

Fecha: ____ / ____ / ____

BROKER 1

TRIBE LEAD

Nombre:

Empresa:

ANEXO N° 01
DECLARACIÓN JURADA

Mediante el presente documento, _____ con RUC / DNI N° _____, con domicilio en _____, distrito de _____, provincia y departamento de _____, (en caso de ser persona jurídica) debidamente representada por _____, identificado con DNI N° _____, según poderes inscritos en la Partida N° _____ del Registro de Personas Jurídicas de _____, **DECLARO BAJO JURAMENTO** que:

Conozco, acepto y me someto a dar cumplimiento a las disposiciones del Reglamento de Supervisión y Control de los Corredores y Auxiliares de Seguros, aprobado por Resolución SBS N° 809-2019, así como sus normas modificatorias o sustitutorias; mediante el cual nos hacemos responsables de cumplir con la normativa antes mencionada y declaramos que, para alcanzar con las metas propuestas por **RIMAC SEGUROS Y REASEGUROS**, desempeñaremos nuestras funciones con profesionalismo y sin afectar nuestra independencia.

Suscrito en la ciudad de Lima, el día ____ de ____ de 2024.

Nombres y apellidos:

DNI:

Anexo 2

	corredor	contratante	producto	poliza	fecinivig	fecfinvig	mes	primaneta
0	BROKER 1	CONTRATANTE 1	WEB VEHICULOS	999999	19/03/2023	19/03/2024	202403	39557
1	BROKER 1	CONTRATANTE 2	WEB VEHICULOS	999999	29/03/2023	29/03/2024	202403	9109

Anexo 2

Avance programa de comisiones



AVANCE CAMPAÑA

2024 Q1

¡Hola BROKER 1!

Queremos contarte todo lo que necesitas saber sobre el avance de tu Campaña **del 01/03/2024 al cierre de 31/03/2024**

Resumen de Avance:

Avance Total (%)	Avance Total	Meta Total	Clientes Nuevos (#)
78.42%	\$ 46,269.71	\$ 59,000.00	0

Ramo	Stock	Nuevo
Riesgos Generales	\$ 0.00	\$ 0.00
Transportes	\$ 0.00	\$ 0.00
Vehículos	\$ 46,269.71	\$ 0.00

Detalle de Clientes Nuevos

contratante	producto	numeropoliza	iniciovigenciapoliza	finvigenciapoliza	primaneta	rango	bono (%)	bono (\$)
-------------	----------	--------------	----------------------	-------------------	-----------	-------	----------	-----------

Detalle de Renovaciones del Stock

Vehículos

	contratante	primaneta por renovar	primaneta renovacion	status
0	CONTRATANTE 1	\$ 39,557.00	\$ 48,269.71	renovacion
1	CONTRATANTE 2	\$ 9,109.00		pendiente