



UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA
La Universidad Católica de Loja

FACULTAD DE INGENIERIA Y ARQUITECTURA

CARRERA DE INGENIERÍA EN SISTEMAS

***Dashboard para la visualización de Indicadores y Métricas
de calidad de servicio en el departamento de Interservice
en Banco Internacional***

Trabajo de titulación previo a la obtención del título de:

INGENIERO EN INFORMÁTICA

Autor: Gómez Toro, Eduardo Rodrigo

Director: Morocho Yunga, Juan Carlos

SAN RAFAEL

2024



Esta versión digital, ha sido acreditada bajo la licencia Creative Commons 4.0, CC BY-NY-SA: Reconocimiento-No comercial-Compartir igual; la cual permite copiar, distribuir y comunicar públicamente la obra, mientras se reconozca la autoría original, no se utilice con fines comerciales y se permiten obras derivadas, siempre que mantenga la misma licencia al ser divulgada. <http://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>

2024

Aprobación del director del Trabajo de Titulación

Loja, 06 de enero de 2024

Magíster,
Juan Carlos Morocho Yunga
Director de la carrera de Informática
Ciudad.-

De mi consideración:

Me permito comunicar que, en calidad de director del presente Trabajo de Titulación denominado: Dashboard para la visualización de Indicadores y Métricas de calidad de servicio en el departamento de Interservice en Banco Internacional, realizado por Eduardo Rodrigo Gómez Toro ha sido orientado y revisado durante su ejecución, así mismo ha sido verificado a través de la herramienta de similitud académica institucional, y cuenta con un porcentaje de coincidencia aceptable. En virtud de ello, y por considerar que el mismo cumple con todos los parámetros establecidos por la Universidad, doy mi aprobación a fin de continuar con el proceso académico correspondiente.

Particular que comunico para los fines pertinentes.

Atentamente,

Director: Mgtr. Juan Carlos Morocho Yunga

C.I.:

Correo electrónico: jcmorocho@utpl.edu.ec

Declaración de autoría y cesión de derechos

Yo, Eduardo Rodrigo Gómez Toro, declaro y acepto en forma expresa lo siguiente:

Ser autor (a) del Trabajo de Titulación denominado: Dashboard para la visualización de Indicadores y Métricas de calidad de servicio en el departamento de Interservice en Banco Internacional, de la carrera de Ingeniería en sistemas, específicamente de los contenidos comprendidos en: (se debe colocar los nombres de los capítulos elaborados en el Trabajo de Titulación), siendo Mgtr. Juan Carlos Morocho Yunga, director (a) del presente trabajo; también declaro que la presente investigación no vulnera derechos de terceros ni utiliza fraudulentamente obras preexistentes. Además, ratifico que las ideas, criterios, opiniones, procedimientos y resultados vertidos en el presente trabajo investigativo, son de mi exclusiva responsabilidad. Eximo expresamente a la Universidad Técnica Particular de Loja y a sus representantes legales de posibles reclamos o acciones judiciales o administrativas, en relación a la propiedad intelectual de este trabajo.

Que la presente obra, producto de mis actividades académicas y de investigación, forma parte del patrimonio de la Universidad Técnica Particular de Loja, de conformidad con el artículo 20, literal j), de la Ley Orgánica de Educación Superior; y, artículo 91 del Estatuto Orgánico de la UTPL, que establece: "Forman parte del patrimonio de la Universidad la propiedad intelectual de investigaciones, trabajos científicos o técnicos y tesis de grado que se realicen a través, o con el apoyo financiero, académico o institucional (operativo) de la Universidad", en tal virtud, cedo a favor de la Universidad Técnica Particular de Loja la titularidad de los derechos patrimoniales que me corresponden en calidad de autor/a, de forma incondicional, completa, exclusiva y por todo el tiempo de su vigencia.

La Universidad Técnica Particular de Loja queda facultada para ingresar el presente trabajo al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública, en cumplimiento del artículo 144 de la Ley Orgánica de Educación Superior.

.....

Autor: Eduardo Rodrigo Gómez Toro

C.I.: 1713491965

Correo electrónico: ergomez@outlook.com

Dedicatoria

Quiero dedicar este trabajo a mi padre, Santos Gómez, y a mi madre, Emilda Toro, quienes han sacrificado sus vidas para ver a sus hijos e hijas crecer de manera completa. También agradezco a mis hermanos y hermanas, quienes brindaron un apoyo desinteresado y afectuoso para hacer realidad uno de mis sueños más preciados. A mi esposa, Johanna, le doy las gracias por su amor incondicional, que me ha permitido crecer como persona y como profesional. Mis hijos, Mateo y Renata, junto a mi esposa, son el mayor tesoro de mi vida, y me motivan a seguir adelante cada día. Finalmente, agradezco a todas las personas que, de una forma u otra, me han apoyado en los momentos buenos y malos, y me han animado a seguir adelante.

Agradecimiento

En primer lugar, quiero expresar mi gratitud a Dios por todas las bendiciones que he recibido, por su amor incondicional a pesar de mis numerosos defectos y mi obstinación, y por permitirme superar este nuevo desafío.

También quiero agradecer a mis padres por inculcarme mis valores fundamentales y por cuidar siempre de mi educación.

Además, deseo mostrar mi aprecio al Banco Internacional, en particular a Karina, por brindarme la oportunidad de llevar a cabo esta labor y por su respaldo.

Al Ingeniero Juan Carlos Morocho Yunga, mi director, quien, con sus conocimientos, experiencia, paciencia y estímulo, ha sido fundamental para que lograra completar esta tesis.

Además, quiero reconocer a todas las personas que han sido parte de mi vida personal y profesional. Les agradezco por su amistad, consejos, apoyo, aliento y compañía en los momentos más desafiantes. Algunas de ellas están a mi lado en este momento, mientras que otras permanecen en mis recuerdos y en mi corazón. No importa dónde se encuentren; quiero expresar mi gratitud por ser una parte importante de mi vida, por todo lo que me han brindado y por las bendiciones que han aportado.

Índice de contenido

<i>Aprobación del director del Trabajo de Titulación</i>	<i>II</i>
<i>Declaración de autoría y cesión de derechos</i>	<i>III</i>
<i>Dedicatoria</i>	<i>V</i>
<i>Agradecimiento.....</i>	<i>VI</i>
<i>Índice de contenido</i>	<i>VII</i>
<i>Resumen</i>	<i>1</i>
<i>Abstract</i>	<i>2</i>
<i>Introducción</i>	<i>3</i>
<i>Capítulo uno.....</i>	<i>5</i>
<i>Definición y alcance del problema.....</i>	<i>5</i>
1.1 Antecedentes	5
1.2 Justificación, Importancia y alcance.....	5
1.3 Objetivo General	6
1.4 Objetivos Específicos	6
<i>Capítulo dos</i>	<i>8</i>
<i>Marco teórico.....</i>	<i>8</i>
2.1 Técnicas de segmentación y agrupación	9
2.1.1 Gestión de datos	9
2.1.2 Técnicas de Modelado	10
2.1.2.1 Modelo Relacional.....	12
2.1.2.2 Modelo Jerárquico	12
2.1.2.3 Modelo de Red.....	12
2.1.2.4 Modelo de base de datos Orientado a objetos.....	13
2.1.2.5 Modelo Entidad – Relación.....	13
2.1.2.6 Esquema de estrella	14
2.1.3 Inteligencia de negocios.....	14

2.1.4	Indicadores KPI	15
2.1.5	Software BI para Empresas	16
2.2	Metodologías de desarrollo de software	22
2.2.1	Métodos de extracción de datos	22
2.2.1.1	Metodología KDD	23
2.2.1.2	Metodología SEMMA	25
2.2.1.3	Metodología CRISP-DM	26
2.2.1.4	Metodología Hefestos	28
2.2.2	Metodologías de desarrollo de software	29
2.2.2.1	Metodología SCRUM	30
2.2.2.2	Programación Extrema (XP)	31
2.2.2.3	Metodología Data-Driven	32
2.2.3	Metodologías de Data Warehouse o almacenamiento de datos	33
	Capítulo tres	36
	Análisis y diseño	36
3.1	Metodología de desarrollo	36
3.2	Necesidad de Negocio	37
3.3	Requerimientos funcionales	37
3.3.1	Introducción	39
3.3.1.1	Propósito	39
3.3.1.2	Ámbito del sistema	39
3.3.1.3	Referencias	39
3.3.1.4	Visión general del documento	40
3.3.2	Descripción General	40
3.3.2.1	Perspectiva del producto	40
3.3.2.2	Funciones del producto	40
3.3.2.3	Características del usuario	40

3.3.2.4	<i>Restricciones</i>	41
3.3.2.5	<i>Suposiciones y Dependencias</i>	41
3.3.2.6	<i>Requisitos futuros</i>	41
3.3.2.7	<i>Requisitos específicos</i>	41
3.3.2.8	<i>Interfaces</i>	41
3.3.2.9	<i>Interfaz de usuario</i>	41
3.3.2.10	<i>Interfaz de hardware y software</i>	41
3.3.2.11	<i>Funciones</i>	42
3.3.2.12	<i>Requerimientos funcionales</i>	42
3.3.2.13	<i>Atributos del sistema</i>	45
3.4	Arquitectura del almacén de datos (DWH)	45
3.5	Diseño de Base de datos	46
3.6	Tecnologías utilizadas para el almacén de datos	48
3.6.1	<i>SQL Server</i>	48
3.6.2	<i>Extracción, transformación y carga ETL</i>	49
3.6.3	<i>Power BI</i>	50
Capítulo cuatro		52
Implementación		52
4.1	Origen de datos	52
4.2	Extracción de datos	54
4.2.1	Conexión	54
4.2.2	Procedimientos almacenados	55
4.2.3	Fechas de carga	57
4.2.4	Calendarización	58
4.2.5	Extracción	59

4.3	Limpieza de datos	69
4.4	Tablas de dimensiones	71
4.5	Integración de datos	73
4.6	Carga Inicial	78
4.7	Visualización	78
4.8	Validación	81
	Conclusiones	84
	Recomendaciones	86
	Referencias	87

Índice de tablas

No se encuentran elementos de tabla de ilustraciones.

Índice de figuras

Figura 1	Representación gráfica del proceso KDD.....	24
Figura 2	Representación gráfica de la metodología CRISP-DM	28
Figura 3	Arquitectura de la extracción y almacenamientos de datos	46
Figura 4	Diagrama de base de datos.....	47
Figura 5	Tablas de la base de datos sd_Interservice	48
Figura 6	Orígenes de datos.....	54
Figura 7	Link Server al servidor de Remedy.....	55
Figura 8	Listado de procedimientos almacenados	56
Figura 9	Configuración del periodo de ejecución del JOB.....	59
Figura 10	Proceso de extracción de las fuentes de datos	60
Figura 11	Sentencia SQL con la que se extrae la fecha de procesamiento.....	61
Figura 12	Proceso de extracción de datos de las órdenes de trabajo hacia la base de datos del repositorio.....	61
Figura 13	Sentencia SQL de extracción de registros de tipo órdenes de trabajo	62
Figura 14	Proceso de extracción de datos de Incidencias hacia la base de datos de repositorio	62
Figura 15	Sentencia SQL de extracción de registros de tipo incidencias.....	63
Figura 16	Proceso de extracción de datos de órdenes de trabajo atendidos por flujo hacia la base de datos de repositorio.....	64
Figura 17	Sentencia SQL de extracción de registros de tipo órdenes de trabajo por flujo	65
Figura 18	Proceso de extracción de datos de requerimientos ingresados en el SAR hacia la base de datos de repositorio.....	66
Figura 19	Sentencia SQL de extracción de registros desde el CRM.....	67
Figura 20	Proceso de extracción de datos de órdenes de trabajo e incidentes que se han puesto en estado pendiente hacia la base de datos de repositorio	68
Figura 21	Sentencia SQL de extracción de registros de los logs de órdenes de trabajo o incidentes en estado pendiente.....	69
Figura 22	Sentencia SQL para la regularización de los registros creados por procesos automáticos.	70
Figura 23	Sentencia SQL para regularizar los registros creados a través del buzón del call center ..	70
Figura 24	Sentencia SQL para regularizar el canal en los registros creados desde el portal WEB....	71
Figura 25	Sentencia SQL para insertar nuevos registros en la tabla de dimensión de usuarios	72
Figura 26	Sentencia SQL para insertar registros en la tabla de dimensiones de agencia	72

Figura 27 Sentencia SQL para insertar registros en la tabla de dimensiones de grupos.....	73
Figura 28 Sentencia SQL para insertar registros en la tabla de dimensiones de estados	73
Figura 29 Proceso de integración de datos	74
Figura 30 Tablero de Monitoreo	80
Figura 31 Tablero de indicadores	80
Figura 32	81
Figura 33 Comparación de un mes entre reporte de Remedy con gráfico del tablero de indicadores agrupados por grupo resolutor.....	82
Figura 34 Comparativa de cantidad de registros de un mes y un grupo resolutor entre el reporte de Remedy y el tablero de indicadores.....	83

Resumen

En el Departamento de Interservice en Banco Internacional se atienden requerimientos de diferentes áreas a nivel nacional, la ausencia de indicadores de gestión y métricas de calidad de servicio dificultan la toma de decisiones sobre la distribución de carga de trabajo y asignación de tareas, al igual que no se tiene indicadores claves de rendimiento (KPI, *Key Performance Indicators*), para poder medir la efectividad y eficiencia de los Acuerdos de nivel de servicio (SLA, *Service Level Agreement*), en la resolución de requerimientos.

El presente proyecto de TT propone el desarrollo de un Dashboard con la que se utilizará todas sus ventajas al momento de la creación de gráficos para que se visualice los indicadores, métricas e indicadores claves de rendimiento (KPI, *Key Performance Indicators*).

Facilite el análisis y el reporte de datos de una forma sencilla y eficiente, mediante representación visual, la alimentación de la información se realizará a través de un proceso de Extracción, Transformación y Carga (ETL, *Extract, Transform, Load*), que permitirá extraer y consolidar datos de múltiples fuentes de información a una Base de Datos, esto ayudará a proporcionar un contexto histórico sobre la evolución del departamento en su gestión y resolución de requerimientos, a la toma de decisiones gerenciales basados en datos, aumentar la productividad y facilitar el trabajo en equipo y a la evaluación de los objetivos anuales con los que se les va medir a todos los analistas.

Palabras clave: Dashboard, Power BI, Métricas.

Abstract

In the Interservice Department at Banco Internacional, requirements from different areas are met at the national level, the absence of management indicators and service quality metrics makes it difficult to make decisions about the distribution of workload and assignment of tasks, as well as the lack of key performance indicators (KPIs, KPIs, etc.). Key Performance Indicators), to be able to measure the effectiveness and efficiency of Service Level Agreements (SLAs) in the resolution of requirements.

This TT project proposes the development of a Dashboard that will use all its advantages when creating graphs to visualize indicators, metrics and Key Performance Indicators (KPIs).

Facilitate the analysis and reporting of data in a simple and efficient way, through visual representation, the feeding of the information will be carried out through an Extract, Transform and Load (ETL) process, which will allow extracting and consolidating data from multiple sources of information to a Database, This will help provide historical context on the evolution of the department in its management and resolution of requirements, to make management decisions based on data, to increase productivity and facilitate teamwork and to the evaluation of the annual objectives against which all analysts will be measured

Keywords: Dashboard, Power BI, Métricas

Introducción

El presente proyecto de TT propone el desarrollo de un Dashboard en Power BI con la que se utilizará todas sus ventajas al momento de la creación de gráficos para que se visualice los indicadores, métricas e indicadores claves de rendimiento (KPI, Key Performance Indicators), que facilite el análisis y el reporte de datos de una forma sencilla y eficiente, mediante representación visual, la alimentación de la información se realizará a través de un proceso de Extracción, Transformación y Carga (ETL, Extract, Transform, Load), que nos permitirá extraer y consolidar datos de múltiples fuentes de información a una Base de Datos, esto ayudará a proporcionar un contexto histórico sobre la evolución del departamento en su gestión y resolución de requerimientos, a la toma de decisiones gerenciales basados en datos, aumentar la productividad y facilitar el trabajo en equipo y a la evaluación de los objetivos anuales con los que se les va medir a todos los analistas.

Business Intelligence (BI) es una tecnología informática que se enfoca en la organización y estructuración de grandes cantidades de datos con el propósito de transformarlos en información valiosa para facilitar la toma de decisiones en una empresa. El BI tiene como principal objetivo apoyar y mejorar el proceso de toma de decisiones dentro de una organización, ya que esto es crucial para su éxito y supervivencia. Esta tesis está dividida en varios capítulos que se desarrollan de manera progresiva para lograr los objetivos planteados en este trabajo.

Capítulo 1, se aborda el tema de estudio, se presenta la formulación del problema, se proporciona la razón que respalda la investigación y se delinean los propósitos que se persiguen.

Capítulo 2, en el marco teórico se introducen los precedentes y se detallan los variados términos vinculados al campo de estudio, los cuales serán fundamentales para respaldar el avance de la investigación. Así, se establece la base para la propuesta de resolución del problema previamente planteado en el capítulo precedente.

Capítulo 3, se describe la metodología para el desarrollo del Proyecto de investigación, así como la metodología de desarrollo, búsqueda de información y se describe la solución al problema.

Capítulo 4, en esta sección, se aborda la ejecución del proyecto, respondiendo a la interrogante sobre cómo se llevará a cabo. De manera específica, se detallan todos los pasos requeridos para alcanzar los objetivos establecidos en este trabajo de tesis.

Capítulo 5, se presentan las conclusiones y recomendaciones de este trabajo. Finalmente, se presentan las referencias bibliográficas utilizadas durante el desarrollo de la tesis y los anexos.

Capítulo uno

Definición y alcance del problema

En este capítulo, se analizan y delimitan los aspectos clave del problema, como sus causas, consecuencias, contexto y objetivos. Es fundamental definir con precisión el problema para poder desarrollar estrategias efectivas de solución. Esto implica recopilar información relevante, realizar investigaciones, analizar datos y consultar a expertos en el tema. Al establecer el alcance del problema, se determina el ámbito y los límites de la investigación o proyecto, así como los recursos y tiempo disponibles. Una definición y alcance claros del problema son fundamentales para orientar y guiar el trabajo posterior en la búsqueda de soluciones efectivas.

1.1 Antecedentes

En el Departamento de Interservice en Banco Internacional se atienden requerimientos de diferentes áreas a nivel nacional, el problema que se tiene es que no se dispone de indicadores de gestión y métricas de calidad de servicio, esto dificulta al momento de la toma de decisiones sobre la distribución de carga de trabajo y asignación de tareas, al igual que no se tiene indicadores claves de rendimiento (KPI, Key Performance Indicators), para poder medir la efectividad y eficiencia de los acuerdos de nivel de servicio (SLA, Service Level Agreement), en la resolución de requerimientos.

La herramienta que se utiliza actualmente no dispone de un visualizador de indicadores o métricas, solo se puede extraer un reporte con el detalle de los requerimientos atendidos por rango de fechas, con esta información se está realizando actualmente un dashboard en Excel que se alimenta mensualmente de forma manual.

1.2 Justificación, Importancia y alcance

La razón para elegir este tema de estudio se fundamenta en la necesidad del departamento de Interservice en el Banco Internacional de contar con indicadores y medidas de calidad de servicio. Estos indicadores son esenciales para respaldar las decisiones de

gestión y facilitar la asignación de tareas al personal, considerando la carga laboral y el tiempo requerido para resolver solicitudes.

Estas métricas ayudan a la evaluación del personal en el cumplimiento de algunos de los objetivos anuales de los empleados que se mide a través del cumplimiento de los SLA en cada requerimiento y la cantidad de requerimientos atendidos.

Es muy importante contar con esta información ya que nos ayuda a ver de forma gráfica el estado actual del departamento en cuanto a, el incremento que ha tenido en requerimientos con un historial de 2 años que nos puede ayudar a presionar a las áreas dueñas de los productos en la solución a los problemas que se presentan con más frecuencia.

Hasta el momento, estas fuentes de datos se utilizan como el fundamento para crear el panel de control, a través del cual la gerencia tomará decisiones oportunas en función de las necesidades del departamento.

El objetivo de esta investigación es implementar el panel de control como una herramienta estratégica para mejorar el proceso de toma de decisiones y la evaluación de los resultados.

1.3 Objetivo General

La finalidad de este proyecto es introducir un Dashboard como una herramienta estratégica y evaluar cómo influye en el proceso de toma de decisiones en el departamento de Interservice en el Banco Internacional. El propósito es supervisar y aumentar la eficiencia en el cumplimiento de los Acuerdos de Nivel de Servicio (SLA) para los requerimientos, así como asegurar una distribución equitativa de la carga laboral.

1.4 Objetivos Específicos

Examinar la situación actual del departamento de Interservice a través de reuniones de recopilación de datos con la subgerencia, con el propósito de definir las necesidades del departamento y evaluar el estado actual de la gestión de información.

Evaluar las capacidades de las principales herramientas de Business Intelligence y las metodologías de desarrollo líderes en el mercado, comparando sus características y

beneficios clave, con el objetivo de determinar cuáles se adecuan a los requisitos de Interservice - Banco Internacional.

Crear y poner en marcha el panel de control utilizando las metodologías apropiadas que faciliten la recopilación efectiva de datos e implementación de la herramienta seleccionada.

Verificar la mejora en el proceso de toma de decisiones de Interservice, basándonos en los requisitos iniciales, a través de la realización de pruebas de funcionamiento de los paneles de control y el análisis de los resultados.

Capítulo dos

Marco teórico

En este capítulo explicamos sobre el marco teórico, en el ámbito de gestión de datos abarca una serie de temas fundamentales para comprender y aplicar eficientemente las estrategias y técnicas relacionadas con la gestión de datos. Se exploran conceptos como la organización y estructura de los datos, la importancia de la calidad y la integridad de los datos, dentro de las metodologías de desarrollo, se analizan enfoques y modelos que guían el proceso de desarrollo de sistemas y aplicaciones. Estas metodologías proporcionan marcos de trabajo, etapas y técnicas para planificar, diseñar, implementar y mantener sistemas de gestión de datos de manera efectiva. Algunas metodologías ampliamente utilizadas son SCRUM, enfoques ágiles y enfoques en cascada, mientras que las técnicas de modelado, por otro lado, son herramientas y enfoques que permiten representar visualmente los datos y las relaciones entre ellos. Estas técnicas incluyen el modelado de datos a través de diagramas entidad-relación, diagramas de flujo de datos y diagramas UML. El modelado de datos es esencial para comprender la estructura y el flujo de la información, lo que facilita el diseño y la implementación de sistemas de gestión de datos eficientes.

En resumen, el marco teórico abarca temas cruciales como la gestión de datos, metodologías de desarrollo y técnicas de modelado. Estos conocimientos son fundamentales para abordar de manera efectiva los desafíos y oportunidades relacionados con la gestión de datos en diversos contextos organizativos.

2.1 Técnicas de segmentación y agrupación

2.1.1 Gestión de datos

Tomando como referencia (Piattini Velthuis et al., 2018) el manejo y protección de los datos como un recurso valioso es esencial para cualquier empresa. En la actualidad, las empresas se enfrentan a desafíos estratégicos cada vez más complejos y necesitan una mayor cantidad de datos para lograr sus objetivos.

Sin embargo, el problema recurrente en muchas empresas es la falta de conexión entre los datos. Esta falta de orden dificulta obtener una visión completa y global de todas las áreas del negocio. Por ejemplo, adquirir una visión de 360 grados sobre un cliente se vuelve complicado debido a razones comunes como las siguientes:

- Hay deficiencias en la comunicación entre las aplicaciones.
- La calidad de los datos es inferior a la deseada.
- La información no ofrece la confiabilidad que se espera.

La gestión de datos es el marco en el que se enquistan todas las demás funciones. Su objetivo principal es supervisar los datos de toda la empresa, departamento por departamento, aplicación por aplicación y área por área. En términos visuales, se puede comparar con la estructura gubernamental de un país, donde las demás áreas son los diferentes ministerios.

La tarea Data Governance implica:

- La creación de pautas, políticas y procedimientos que regulen la utilización, desarrollo y administración de los datos en el ámbito corporativo.
- La identificación de nuevos cargos y obligaciones en la estructura organizacional.
- La utilización de la tecnología apropiada como apoyo.

El Data Governance es esencial para el funcionamiento de todas las demás funciones. Su objetivo principal es asegurar la accesibilidad de los datos, lo cual se logra mediante la aplicación de principios específicos que definen cómo se debe garantizar dicha accesibilidad.

En resumen, el Gobierno de datos implica la planificación, supervisión y control de todos los aspectos relacionados con los datos. La gobernanza de datos establece el marco dentro del cual se llevan a cabo todas las actividades relacionadas con los datos, y busca lograr los siguientes objetivos:

- Garantizar que los datos satisfacen las necesidades o requerimientos.
- Disminuir los gastos relacionados con la administración y seguridad de los datos.
- Administrar y cultivar los datos como un recurso valioso para la empresa.

2.1.2 Técnicas de Modelado

Según (Deborah Henderson, 2022), un modelo es una colección de herramientas conceptuales que se utilizan para representar datos, sus interrelaciones, su significado y las restricciones de consistencia asociadas. Este modelo presenta las siguientes características:

- Consiste en examinar los elementos relevantes para una organización y cómo se relacionan entre sí
- Implica la identificación y registro de los recursos de datos utilizados en el negocio.
- El modelado se enfoca en la pregunta "qué" en lugar de "cómo", ya que se centra en la representación y comprensión de los datos en lugar de su procesamiento.
- Es un desafío considerable y requerido, pero es una actividad esencial que solo se domina a través de la experiencia.

Asimismo, un modelo de base de datos representa la estructura lógica de la base de datos, incluyendo las relaciones y restricciones que definen cómo se almacenan y acceden los datos. Los modelos de base de datos específicos se crean siguiendo las reglas y conceptos de un modelo de datos más amplio adoptado por los diseñadores. En la mayoría de los casos, estos modelos de datos se representan mediante un diagrama de base de datos correspondiente.

Hay muchos tipos de modelos de bases de datos. Algunos de los más comunes incluyen:

- Modelo relacional.
- Modelo de red.
- Modelo jerárquico
- Modelo entidad – relación.
- Modelo de base de datos orientado a objetos.
- Modelo de documentos.
- Esquema de estrella.

La selección de la forma en que se describe una base de datos se basa en varios elementos, siendo el factor principal la adecuación del sistema de gestión de bases de datos que se emplea a un modelo específico. La mayoría de los sistemas de gestión de bases de datos están diseñados en torno a un modelo de datos particular y exigen que los usuarios se adhieran a ese modelo, aunque algunos sistemas pueden ser versátiles y admitir varios modelos.

Además, en las diversas fases del proceso de diseño de bases de datos, se emplean varios modelos. Los modelos conceptuales de alto nivel son más apropiados para definir las relaciones entre los datos desde una perspectiva humana. En contraste, los modelos lógicos basados en registros se aproximan más a la manera en que los datos se almacenan en el servidor.

Tomando como referencia(Deborah Henderson, 2022), la selección de un modelo de datos también está influenciada por la alineación de las prioridades con las ventajas ofrecidas por un modelo específico de base de datos, ya sea que esas prioridades estén relacionadas con la velocidad, la reducción de costos, la facilidad de uso u otros aspectos.

Según (IBM, 2019) los modelos más frecuentemente empleados son los que se describen a continuación:

2.1.2.1 Modelo Relacional

Tomando como referencia a (IBM, 2019), el modelo relacional, que es el más comúnmente utilizado, organiza los datos en tablas, también conocidas como relaciones. Cada tabla está compuesta por columnas y filas. Cada columna representa un atributo de la entidad, como el precio, el código postal o la fecha de nacimiento. El conjunto de posibles valores que puede tomar un atributo se denomina dominio. Se selecciona un atributo o combinación de atributos específicos como clave primaria, que puede ser referenciada en otras tablas como clave externa.

Cada fila, también conocida como tupla, contiene datos sobre una instancia específica de la entidad, por ejemplo, un empleado en particular.

Dentro de la base de datos, las tablas pueden ser normalizadas, lo que implica cumplir con las reglas de normalización que hacen que la base de datos sea flexible, adaptable y escalable. Al estar normalizada, cada parte de los datos se divide en unidades atómicas, es decir, se divide en las partes más pequeñas y útiles posibles.

2.1.2.2 Modelo Jerárquico

El modelo jerárquico estructura los datos en un formato de árbol, donde cada registro está vinculado a un elemento raíz único. Los registros del mismo nivel se organizan en un orden específico que se emplea como la disposición física de almacenamiento de la base de datos. Este modelo resulta útil para representar diversas relaciones del mundo real.

Durante las décadas de 1960 y 1970, los Sistemas de Gestión de Información de IBM utilizaron ampliamente este modelo. No obstante, en la actualidad, su uso es poco común debido a ciertas ineficiencias operativas que presenta.

2.1.2.3 Modelo de Red

El modelo de red evoluciona a partir del modelo jerárquico y posibilita relaciones de muchos a muchos entre registros interconectados, lo que significa que varios registros pueden servir como registros principales. Este modelo se construye utilizando conjuntos de registros relacionados, basados en la teoría matemática de conjuntos. Cada grupo está formado por un registro principal, también llamado propietario, y uno o más registros

secundarios. Un registro secundario puede formar parte de varios grupos, lo que permite que este modelo represente relaciones complejas.

En la década de 1970, tras recibir una definición formal por parte de la Conference on Data Systems Languages (CODASYL), el modelo de red ganó una gran popularidad.

2.1.2.4 Modelo de base de datos Orientado a objetos

Este modelo postula que una base de datos se compone de una serie de elementos de software reutilizables, denominados objetos, que cuentan con funciones y métodos interconectados. Hay diversos tipos de bases de datos orientadas a objetos:

- Una base de datos multimedia incluye elementos multimedia, como imágenes, que no pueden almacenarse en una base de datos relacional convencional.

- Una base de datos de hipertexto posibilita la vinculación de cualquier objeto con cualquier otro. Se utiliza para organizar colecciones de datos diversos, aunque no es la elección óptima para realizar análisis numéricos.

- El modelo de base de datos orientado a objetos se reconoce como el modelo más avanzado de base de datos posrelacional, ya que no se restringe exclusivamente a tablas, aunque las incluye. Estos modelos también se denominan modelos de base de datos híbridos.

2.1.2.5 Modelo Entidad – Relación

Este modelo registra las conexiones entre entidades del mundo real de una manera parecida al modelo de red, pero no está directamente ligado a la configuración física de la base de datos. En su lugar, se emplea con regularidad en la fase de diseño conceptual de una base de datos.

En este modelo, las personas, lugares y cosas que se almacenan como datos se llaman entidades, cada una de las cuales tiene atributos que forman su dominio en conjunto. La cardinalidad, es decir, las relaciones entre las entidades, también se representa en diagramas.

Una manera típica de representar estas relaciones es a través del diagrama de entidad-relación, en el cual una tabla central de datos se enlaza con diversas tablas dimensionales.

2.1.2.6 Esquema de estrella

Según (Iseminger, 2019) el diseño de esquema de estrella es una metodología de modelado consolidada que ha sido ampliamente aceptada en los almacenes de datos relacionales. Esta técnica implica que los diseñadores clasifiquen las tablas del modelo como dimensiones o hechos.

Las tablas de dimensiones representan entidades empresariales, que pueden ser productos, personas, lugares, conceptos e incluso el tiempo. En un esquema de estrella, la tabla de dimensiones más común y consistente es la tabla de dimensiones de fecha. Esta tabla de dimensiones incluye una columna (o varias columnas) que sirven como identificador único, así como columnas descriptivas que proporcionan más información sobre la entidad representada.

Las tablas de hechos son capaces de almacenar observaciones o eventos, que pueden incluir pedidos de ventas, existencias, tasas de cambio, temperaturas, entre otros. Estas tablas de hechos contienen columnas de clave de dimensiones que están relacionadas con las tablas de dimensiones, así como columnas numéricas de medidas.

Las columnas que funcionan como claves en las dimensiones son las que establecen la estructura dimensional de la tabla de hechos, mientras que los datos de las claves en dimensiones definen el nivel de detalle de dicha estructura. Por ejemplo, consideremos una tabla de hechos diseñada para almacenar objetivos de ventas, que cuenta con dos columnas de clave de dimensiones: "Date" y "ProductKey". Es fácil comprender que esta tabla tiene dos dimensiones. Sin embargo, la granularidad no puede determinarse sin tener en cuenta los valores de las claves de dimensiones. En este ejemplo, si los valores almacenados en la columna "Date" representan el primer día de cada mes, la granularidad de la tabla estaría a nivel de mes y producto.

2.1.3 Inteligencia de negocios

La inteligencia de negocios, también conocida como BI por sus siglas en inglés (Business Intelligence), se refiere a un conjunto de metodologías, procesos, aplicaciones y

tecnologías utilizadas para recopilar, analizar y presentar información relevante y útil para la toma de decisiones en una organización. El objetivo principal de la inteligencia de negocios es convertir los datos empresariales en conocimiento accionable que permita a las empresas tomar decisiones informadas y estratégicas.

La inteligencia de negocios implica la recopilación y consolidación de datos de diversas fuentes internas y externas de la organización, la transformación de esos datos en información significativa a través de técnicas de análisis y modelado, y la presentación de esos resultados de manera clara y comprensible para los usuarios finales, como gerentes, ejecutivos y otros profesionales de la empresa.

Al utilizar herramientas y tecnologías especializadas, como sistemas de gestión de bases de datos, aplicaciones de análisis de datos, visualización de datos y cuadros de mando, la inteligencia de negocios ayuda a identificar patrones, tendencias, relaciones y oportunidades en los datos, lo que permite a las organizaciones tomar decisiones estratégicas, optimizar sus operaciones, mejorar el rendimiento y obtener una ventaja competitiva en el mercado (García-Jiménez et al., 2021).

2.1.4 Indicadores KPI

Los indicadores clave de rendimiento (KPIs, por sus siglas en inglés, Key Performance Indicators) son una herramienta utilizada para medir el éxito de un negocio mediante el uso de métricas objetivas y pautas cuantificables. Estos indicadores pueden ser empleados para evaluar el estado general de una organización, comparar el rendimiento actual con el plan de negocio, o hacer un seguimiento del progreso de una actividad específica, meta, departamento o producto. Su propósito es proporcionar una forma concreta de medir y monitorear el desempeño empresarial, permitiendo a las organizaciones tomar decisiones informadas y realizar ajustes necesarios para alcanzar sus objetivos (Martins, 2022).

En el mundo actual, las actividades de las empresas generan una gran cantidad de información que comúnmente se almacena en bases de datos o, de manera más simple, en hojas de cálculo de Excel.

Como es evidente, la inteligencia de negocios (BI) utiliza indicadores clave de rendimiento (KPIs) para analizar y evaluar el estado actual de la empresa y establecer un plan de acción. Estos indicadores proporcionan información valiosa para la toma de decisiones, por lo que es importante asegurarse de que los procesos críticos se estén monitoreando.

Aunque existen KPIs universales que se aplican a todas las organizaciones, cada empresa puede establecer aquellos que consideren más importantes para evaluar su actividad. Estos KPIs pueden analizar aspectos estratégicos, procesos, económicos, entre otros; en definitiva, evalúan si la empresa está cumpliendo con los objetivos establecidos.

Según (Porras Blanco, 2017), los KPIs en sí mismo no son metas ni objetivos, si no, que son una forma de medir el correcto funcionamiento de la organización para la consecución de sus objetivos.

2.1.5 Software BI para Empresas

Existen varios softwares de inteligencia empresarial (BI) disponibles en el mercado para ayudar a las empresas en la gestión, análisis y visualización de sus datos. Algunos de los software más populares y ampliamente utilizados incluyen:

Tableau, es una plataforma líder en visualización de datos y análisis empresarial. Proporciona herramientas poderosas para explorar, analizar y compartir datos de manera intuitiva y efectiva. Con Tableau, los usuarios pueden conectarse a diversas fuentes de datos, crear visualizaciones interactivas y paneles de control personalizados, y compartir sus descubrimientos con otros colaboradores.

Tableau ofrece una amplia gama de capacidades, incluyendo la creación de gráficos y visualizaciones interactivas, la generación de informes y cuadros de mando, el análisis avanzado de datos mediante funciones estadísticas y cálculos personalizados, y la colaboración en tiempo real. Su interfaz amigable y su enfoque basado en arrastrar y soltar hacen que sea accesible para usuarios de todos los niveles de experiencia, desde principiantes hasta expertos en análisis de datos.

Además, Tableau es compatible con una amplia variedad de fuentes de datos, como bases de datos relacionales, hojas de cálculo, archivos CSV, servicios web, entre otros. También ofrece opciones de integración con otras herramientas y plataformas, lo que facilita la incorporación de Tableau en el ecosistema tecnológico existente de una organización.

En resumen, Tableau es una solución completa de visualización y análisis de datos que ayuda a las empresas a obtener una comprensión más profunda de sus datos y a tomar decisiones basadas en información sólida y visualmente atractiva¹.

Pentaho, es una plataforma de inteligencia empresarial (BI) de código abierto que ofrece una amplia gama de herramientas para el análisis de datos, la integración de datos y la generación de informes. Proporciona un conjunto completo de capacidades de BI que incluyen extracción, transformación y carga (ETL), generación de informes, análisis de datos, minería de datos y paneles de control interactivos.

La plataforma Pentaho permite a las organizaciones acceder y combinar datos de diferentes fuentes, como bases de datos relacionales, hojas de cálculo, servicios web y sistemas de archivos, para crear flujos de datos integrados y limpios. Utilizando su interfaz gráfica de arrastrar y soltar, los usuarios pueden diseñar y ejecutar procesos de ETL para extraer, transformar y cargar datos en un almacén de datos o en un repositorio centralizado.

Además del ETL, Pentaho ofrece herramientas para la creación de informes y análisis de datos. Los usuarios pueden crear informes personalizados con una amplia variedad de formatos y estilos, incluyendo tablas, gráficos y visualizaciones interactivas. También pueden realizar análisis ad hoc y explorar los datos para descubrir patrones, tendencias y relaciones.

Pentaho es conocido por su enfoque de código abierto, lo que significa que su código fuente está disponible libremente y puede ser modificado y personalizado según las necesidades de cada organización. Además, la plataforma ofrece una comunidad activa de desarrolladores y usuarios que contribuyen con mejoras y brindan soporte adicional.

¹ Salesforce, Tableau (2023), <https://www.tableau.com>

En resumen, Pentaho es una plataforma de inteligencia empresarial flexible y de código abierto que ayuda a las organizaciones a gestionar, analizar y visualizar sus datos de manera efectiva. Ofrece una amplia gama de herramientas para el ETL, generación de informes y análisis de datos, lo que la convierte en una solución completa para las necesidades de BI de una empresa.²

QlikView, es una herramienta de inteligencia empresarial (BI) desarrollada por Qlik. Es reconocida por sus capacidades de análisis visual y descubrimiento de datos. QlikView permite a los usuarios explorar y analizar datos de manera flexible, generar visualizaciones interactivas y colaborar en tiempo real.

Con QlikView, los usuarios pueden cargar datos de múltiples fuentes y crear asociaciones entre ellos para descubrir patrones, tendencias y relaciones ocultas. La herramienta utiliza una tecnología llamada "asociación de datos" que permite a los usuarios navegar libremente por los datos sin restricciones predefinidas.

QlikView ofrece una interfaz intuitiva y amigable que facilita la creación de informes y paneles de control personalizados. Los usuarios pueden arrastrar y soltar elementos visuales para construir visualizaciones interactivas, como gráficos, tablas y mapas, que les ayuden a comprender mejor los datos y comunicar insights de manera efectiva.

Además, QlikView brinda capacidades de colaboración, lo que permite a los usuarios compartir y colaborar en tiempo real en los análisis y descubrimientos realizados. Esto fomenta el trabajo en equipo y la toma de decisiones basada en datos dentro de la organización.

En resumen, QlikView es una potente herramienta de BI que ofrece capacidades avanzadas de análisis visual y descubrimiento de datos. Es utilizada por muchas organizaciones para obtener insights accionables y mejorar la toma de decisiones basada en datos.³

² Hitachi Vantara, Pentaho (2023), <https://htchivantara.is/436Zy6K>

³ QlikTech International AB, QlikView (2023), <https://www.qlik.com/es-es/products/qlikview>

Jet Reports es una herramienta de inteligencia empresarial (BI) diseñada específicamente para integrarse con Microsoft Dynamics ERP, como Dynamics 365 Business Central y Dynamics NAV. Jet Reports permite a las empresas generar informes financieros y análisis de datos utilizando los datos almacenados en sus sistemas ERP.

Con Jet Reports, los usuarios pueden acceder y extraer datos de su sistema ERP de manera rápida y sencilla. La herramienta proporciona un conjunto de funciones y plantillas predefinidas que permiten a los usuarios crear informes financieros, como estados de resultados, balances generales y análisis de ventas, entre otros.

La interfaz de Jet Reports es amigable y se integra directamente con Excel, lo que facilita a los usuarios familiarizados con Excel utilizar la herramienta sin necesidad de aprender un nuevo entorno. Los usuarios pueden aprovechar las funcionalidades de Excel, como fórmulas, tablas dinámicas y gráficos, para analizar y presentar los datos extraídos de su sistema ERP.

Jet Reports también ofrece capacidades de programación y automatización, lo que permite a los usuarios programar la generación de informes periódicos y establecer actualizaciones automáticas de datos. Esto ahorra tiempo y garantiza que los informes estén siempre actualizados con la información más reciente.

En resumen, Jet Reports es una herramienta de BI especializada en integrarse con sistemas ERP de Microsoft Dynamics. Permite a los usuarios generar informes financieros y análisis de datos utilizando los datos almacenados en su sistema ERP, y se integra con Excel para una experiencia de usuario familiar.⁴

Google Analytics es una herramienta de análisis web desarrollada por Google que permite a las empresas y propietarios de sitios web obtener información detallada sobre el rendimiento de su sitio. Con Google Analytics, los usuarios pueden recopilar y analizar datos de tráfico, comportamiento del usuario y conversiones en su sitio web.

⁴ Insight Software, Jet Reports (2023), <https://insightsoftware.com/es/jet/jet-reports/>

La herramienta proporciona métricas y estadísticas clave que ayudan a comprender cómo los visitantes interactúan con el sitio, qué páginas visitan, cuánto tiempo pasan en el sitio, de dónde provienen y qué acciones realizan, como realizar una compra o completar un formulario.

Google Analytics utiliza un código de seguimiento que se coloca en cada página del sitio web para recopilar datos. Estos datos se presentan en un panel de control intuitivo que ofrece una variedad de informes y gráficos personalizables. Los usuarios pueden ver información detallada sobre el tráfico del sitio, fuentes de tráfico, palabras clave utilizadas para encontrar el sitio, datos demográficos de los visitantes y muchas otras métricas relevantes.

Además, Google Analytics ofrece funciones avanzadas, como el seguimiento de objetivos y embudos de conversión, que permiten a los usuarios medir y analizar el rendimiento de las conversiones en su sitio web. Esto es especialmente útil para comprender el comportamiento del usuario y optimizar el sitio para mejorar las tasas de conversión.

Google Analytics también se integra con otras herramientas y servicios de Google, como Google Ads, lo que permite a los usuarios realizar un seguimiento del rendimiento de sus campañas publicitarias y evaluar el retorno de la inversión.

En resumen, Google Analytics es una herramienta de análisis web que proporciona información detallada sobre el rendimiento de un sitio web. Permite a los usuarios recopilar y analizar datos de tráfico, comportamiento del usuario y conversiones, lo que ayuda a comprender y optimizar el rendimiento del sitio.⁵

Microsoft Power BI es una plataforma de inteligencia empresarial (BI) que permite a las organizaciones recopilar, analizar y visualizar datos de manera eficiente. Power BI se compone de varios componentes que trabajan juntos para brindar capacidades completas de BI.

⁵ Google, Google Analytics (2023), <https://marketingplatform.google.com/intl/es/about/analytics/>

La plataforma incluye Power BI Desktop, una aplicación de escritorio que permite a los usuarios crear informes interactivos y paneles de control utilizando una amplia gama de fuentes de datos. Power BI Desktop proporciona herramientas de modelado de datos, capacidades de transformación de datos y opciones de visualización avanzadas.

Power BI también ofrece servicios en la nube, como Power BI Service y Power BI Mobile, que permiten a los usuarios compartir, colaborar y acceder a los informes y paneles de control desde cualquier dispositivo y en cualquier momento. Power BI Service es una plataforma basada en la nube que permite la publicación de informes, la programación de actualizaciones automáticas de datos y la creación de paneles de control interactivos. Power BI Mobile es una aplicación móvil que permite a los usuarios acceder a los informes y paneles de control desde dispositivos móviles.

Además, Power BI se integra con una amplia gama de fuentes de datos, incluidas bases de datos, servicios en la nube, archivos de Excel y aplicaciones de terceros. Esto permite a los usuarios conectar y combinar datos de diversas fuentes para obtener una visión completa de su negocio.

Power BI también ofrece capacidades de análisis avanzado, como el uso de consultas de lenguaje natural y la generación automática de insights a partir de datos. Los usuarios pueden hacer preguntas en lenguaje natural y obtener respuestas instantáneas en forma de visualizaciones interactivas.

En resumen, Microsoft Power BI es una plataforma de inteligencia empresarial que permite a las organizaciones recopilar, analizar y visualizar datos de manera efectiva. Ofrece herramientas de creación de informes, capacidades de modelado de datos, servicios en la nube y capacidades de análisis avanzado para ayudar a las organizaciones a tomar decisiones informadas y obtener una visión completa de sus datos empresariales.⁶

⁶ Microsoft, Power BI (2023), <https://powerbi.microsoft.com/es-es/>

Estos son solo algunos ejemplos de software de BI para empresas, y la elección del software adecuado dependerá de las necesidades y requisitos específicos de cada organización. Es importante evaluar las características, funcionalidades, costo y soporte de cada opción antes de tomar una decisión.

2.2 Metodologías de desarrollo de software

La sección 2.2 aborda el fascinante mundo de las metodologías de desarrollo de software, un componente esencial en la creación de soluciones efectivas en el ámbito de la tecnología de la información. Estas metodologías proporcionan una estructura organizada y sistemática para guiar el proceso de construcción y diseño de proyectos. En esta sección, exploraremos diversas metodologías que abarcan desde enfoques tradicionales hasta los más ágiles y flexibles, cada uno con sus propias ventajas y aplicaciones específicas. A medida que profundicemos en esta sección, desentrañaremos cómo estas metodologías influyen en la planificación, ejecución y entrega exitosa de proyectos de desarrollo, resaltando su impacto en la gestión de datos y técnicas de modelado, aspectos fundamentales en la construcción de soluciones informáticas robustas y eficientes.

2.2.1 Métodos de extracción de datos

El éxito de un proyecto de software a menudo depende en gran medida de su metodología de trabajo. Esta metodología proporciona una guía para que los miembros del equipo sepan qué tareas deben realizar y cuándo deben llevarlas a cabo.

La extracción de datos se refiere al proceso de obtener información específica y relevante de una fuente de datos. Consiste en seleccionar y recopilar datos de diferentes fuentes, como bases de datos, archivos, páginas web u otras fuentes de información, para su posterior análisis o uso en un contexto determinado.

Durante el proceso de extracción de datos, se aplican técnicas y herramientas para acceder a la fuente de datos, extraer los datos requeridos y transformarlos en un formato adecuado para su posterior procesamiento. Esto puede incluir la eliminación de datos

innecesarios, la limpieza de datos erróneos o duplicados, y la integración de datos de múltiples fuentes en un único conjunto de datos coherente.

La extracción de datos es una parte fundamental en muchas áreas, como la investigación científica, el análisis de negocios, la toma de decisiones y el desarrollo de aplicaciones. Permite obtener información valiosa y significativa a partir de grandes volúmenes de datos, lo que facilita la comprensión y el aprovechamiento de la información para diversos propósitos.

2.2.1.1 Metodología KDD

Según (León Guzmán, 2019), la Extracción de conocimiento se encuentra estrechamente relacionada con el proceso de descubrimiento de conocimiento en bases de datos (*Knowledge Discovery in Databases, KDD*). Este proceso se enfoca en la identificación de conocimiento valioso e información relevante dentro de conjuntos de datos almacenados en repositorios de información. A diferencia de un proceso automático, el *KDD* es un proceso iterativo que implica una exploración exhaustiva de grandes volúmenes de datos con el fin de identificar relaciones significativas. A través de este proceso, se extrae información de calidad que permite establecer conclusiones basadas en las relaciones o modelos encontrados en los datos.

Las etapas del proceso *KDD* se dividen en 5 y son:

1. La **Selección de datos** implica identificar las fuentes de datos relevantes y el tipo de información requerida para el análisis. En esta etapa, se extraen los datos pertinentes de una o varias fuentes de datos.
2. El **Preprocesamiento** implica la preparación y limpieza de los datos extraídos de diferentes fuentes para que sean manejables y aptos para las etapas siguientes. Se aplican diversas estrategias para abordar datos faltantes, inconsistentes o fuera de rango, y se crea una estructura de datos adecuada para la transformación posterior.
3. La **Transformación** consiste en realizar un tratamiento preliminar de los datos, incluyendo transformaciones y la generación de nuevas variables basadas en

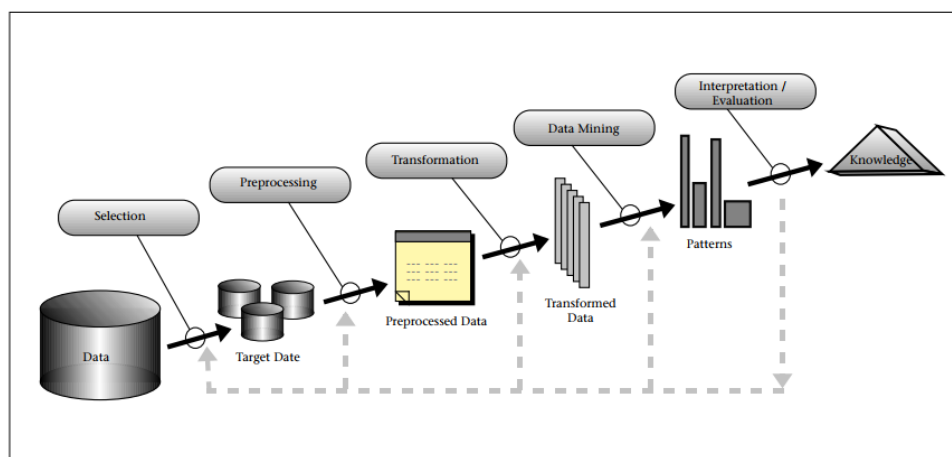
las existentes. Se aplican operaciones como agregación o normalización para consolidar los datos de manera que sean adecuados para la siguiente fase.

4. El **Data Mining** es la etapa central del proceso, donde se aplican métodos inteligentes con el objetivo de descubrir patrones previamente desconocidos, útiles, comprensibles y relevantes que se encuentran ocultos en los datos.
5. En la fase de **Interpretación y Evaluación**, se reconocen los patrones identificados y se valoran a través de métricas específicas. Se establece la importancia de los patrones descubiertos y se lleva a cabo una valoración de los resultados alcanzados.

Junto con estas etapas, frecuentemente se incorpora una fase inicial que implica analizar las necesidades de la organización y definir el problema. En esta etapa, se establecen los objetivos para el proceso de extracción de datos. También es común incluir una etapa final en la que los resultados se integran en el contexto del negocio para tomar decisiones y realizar acciones comerciales.

Figura 1

Representación gráfica del proceso KDD



Fuente: Koo, J. (2017). Minería de datos y KDD. [Diagrama]. Blogspot. <https://bit.ly/3RMfPug>

2.2.1.2 Metodología SEMMA

La metodología fue desarrollada por SAS Institute y se define como un proceso que involucra la selección, exploración y modelado de grandes volúmenes de datos con el objetivo de descubrir patrones de negocio previamente desconocidos. El nombre de esta metodología es un acrónimo que representa las cinco fases fundamentales del proceso.

El proceso comienza con la extracción de una muestra de la población que se utilizará en el análisis. El objetivo de esta etapa es seleccionar una muestra que sea representativa del problema que se está estudiando. La representatividad de la muestra es crucial, ya que, si no se cumple, el modelo y los resultados obtenidos no serán válidos. El método más habitual para obtener una muestra es a través de la selección aleatoria, donde cada miembro de la población tiene igual probabilidad de ser seleccionado. Este método de muestreo se conoce como muestreo aleatorio simple.

La metodología SEMMA indica que se debe asignar un nivel de confianza a cada muestra considerada en el análisis del proceso.

Después de haber elegido una muestra que sea representativa de la población bajo estudio, siguiendo la metodología SEMMA, es necesario llevar a cabo un análisis exploratorio de la información disponible. Esto se hace con el propósito de simplificar el problema y mejorar la eficiencia del modelo. Para lograr este objetivo, se recomienda emplear herramientas de visualización y técnicas estadísticas que permitan identificar las relaciones entre las variables. De esta manera, se busca identificar las variables explicativas que se utilizarán como entradas en el modelo.

La fase tres de la metodología involucra la transformación de los datos, basándose en la información recopilada durante la etapa de exploración. Esto se hace con el fin de garantizar que los datos estén en el formato adecuado y preparados para su inserción en el modelo.

Una vez que las variables de entrada se han configurado en el modelo y tienen el formato adecuado para aplicar la técnica de modelado, se procede a la fase de análisis y modelado de los datos. El propósito de esta etapa es establecer una relación entre las

variables que proporcionan explicaciones y las variables de interés, lo que permite realizar inferencias sobre sus valores con un nivel de confianza definido.

Las técnicas empleadas para el modelado de datos abarcan enfoques estadísticos convencionales, tales como análisis discriminante, procedimientos de agrupamiento y análisis de regresión, además de métodos basados en datos como redes neuronales, técnicas adaptativas, lógica difusa, árboles de decisión, reglas de asociación y computación evolutiva.

En última instancia, la fase conclusiva del proceso abarca la evaluación de los resultados a través de la valoración de la eficacia del modelo o modelos, en relación con otros enfoques estadísticos o a través de la aplicación a nuevas muestras de población.

2.2.1.3 Metodología CRISP-DM

Según (Coronel Valenzuela, 2019), la metodología CRISP-DM se estructura en cuatro niveles de abstracción, dispuestos en una jerarquía de tareas que abarcan desde el nivel más amplio hasta situaciones más particulares.

A un nivel más general, el proceso se divide en seis fases, y cada fase se subdivide en varias tareas generales de segundo nivel. Estas tareas generales se desglosan en tareas específicas que detallan las acciones necesarias para situaciones particulares. Por ejemplo, si en el segundo nivel se encuentra la tarea general "limpieza de datos", en el tercer nivel se especifican las tareas que deben llevarse a cabo para casos concretos, como la limpieza de datos numéricos o la limpieza de datos categóricos. El cuarto nivel comprende todas las acciones, decisiones y resultados relacionados con el proyecto de Minería de Datos específico.

La metodología CRISP-DM organiza el proceso de un proyecto de Minería de Datos en seis etapas que se relacionan entre sí y se llevan a cabo de manera iterativa durante la ejecución del proyecto.

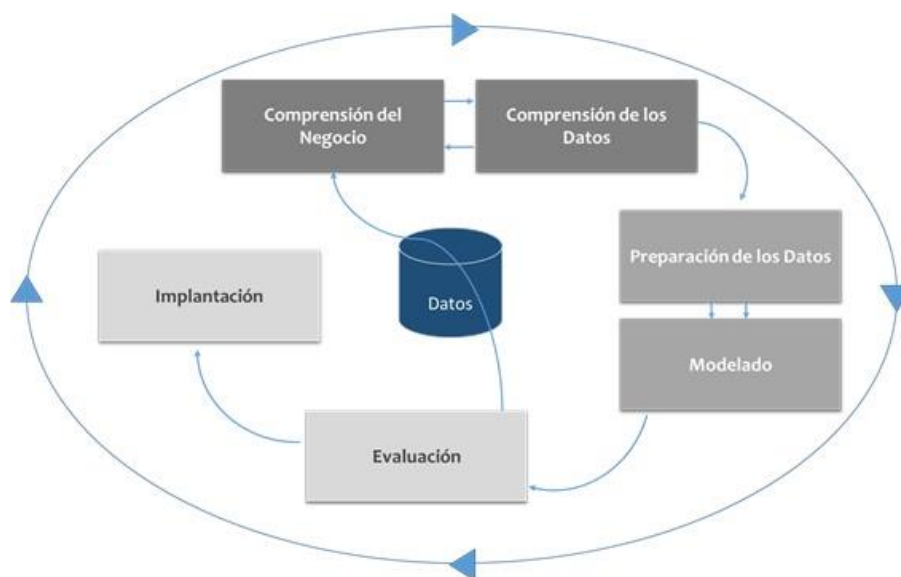
1. **Comprensión del Negocio**, se enfoca en entender los propósitos y requerimientos del proyecto desde una perspectiva empresarial. Luego, se traduce ese conocimiento en una definición de una solución de minería de datos y un plan preliminar para alcanzar los objetivos comerciales.

2. **Comprensión de los Datos**, comienza con la recopilación inicial de datos y continúa con actividades específicas destinadas a adquirir familiaridad con los datos. Esto se hace para identificar posibles problemas de calidad de los datos, obtener una primera impresión de los datos y detectar subconjuntos que puedan ser interesantes para formular hipótesis sobre información oculta.
3. **Preparación de los Datos**, engloba todas las actividades destinadas a crear los datos definitivos a partir de los datos brutos. Las tareas de preparación de datos a menudo se realizan en varias ocasiones y en diversos órdenes. Estas tareas abarcan la selección de atributos, la estructuración de tablas y registros, así como la transformación y depuración de los datos para su uso en las herramientas de modelado. Por lo general, esta fase implica el mayor esfuerzo en el proyecto.
4. **Modelado**, implica la elección y aplicación de diversas técnicas de modelado, ajustando sus parámetros para lograr los valores óptimos. Por lo general, hay varias técnicas disponibles para abordar un mismo problema de minería de datos. Algunas de estas técnicas tienen requisitos particulares en cuanto a la estructura de los datos, lo que a veces implica regresar a la fase de Preparación de Datos.
5. **Evaluación**, después de haber construido uno o más modelos que parecen ser apropiados desde una perspectiva de análisis de datos, es crucial llevar a cabo una evaluación exhaustiva del modelo y de los pasos seguidos en su implementación. Esto se realiza para asegurarse de que el modelo cumple con los objetivos comerciales. El objetivo principal es determinar si hay algún aspecto empresarial que no se ha abordado de manera suficientemente detallada. Al concluir esta fase, se debe tener la certeza de que los objetivos de negocio han sido alcanzados.
6. La fase de **Implantación** no marca el final del proyecto, ya que la creación del modelo, por lo general, no es suficiente. Incluso si el propósito del modelo es

adquirir un mayor entendimiento de los datos, los conocimientos obtenidos deben ser organizados y presentados de una manera que permita a los clientes utilizarlos. La fase de despliegue puede variar en complejidad, desde la generación de informes hasta la implementación de un proceso repetible de minería de datos, dependiendo de los requisitos. En muchos casos, será el cliente y no el analista de datos quien lleve a cabo los pasos de instalación. No obstante, incluso si el analista no lidera la transferencia, es esencial para que el cliente comprenda con anticipación las medidas que deben tomar para aprovechar realmente los modelos desarrollados.

Figura 2

Representación gráfica de la metodología CRISP-DM



Fuente: Diaz, M. (2016). Metodología CRISP-DM. [Diagrama]. Blogspot. <https://bit.ly/3S8YPP5>

2.2.1.4 Metodología Hefestos

HEFESTO es una metodología que se basa en una investigación exhaustiva, la comparación de metodologías existentes y experiencias personales en la creación de almacenes de datos. Es importante destacar que HEFESTO está en constante desarrollo y se ha tenido en cuenta el valioso aporte de aquellos que han utilizado esta metodología en

diferentes países y para diversos propósitos. La retroalimentación recibida ha sido considerada como un valor añadido para su mejora continua.

El concepto fundamental es tener un entendimiento claro de cada paso que se llevará a cabo, evitando caer en la monotonía de seguir un método estrictamente sin comprender completamente lo que se está haciendo ni por qué se está haciendo.

La construcción e implementación de un almacén de datos (DW) puede ser adaptada de manera efectiva a cualquier ciclo de vida de desarrollo de software. Sin embargo, es importante tener en cuenta que, para ciertas fases, las acciones a realizar pueden variar significativamente. Es crucial evitar el uso de metodologías que requieran largas reuniones de requerimientos y análisis, un desarrollo monolítico que consuma mucho tiempo o despliegues prolongados. El objetivo es entregar una primera implementación que satisfaga parte de las necesidades, demostrando así las ventajas del DW y motivando a los usuarios.

La metodología HEFESTO se puede incorporar en cualquier ciclo de vida que cumpla con la condición mencionada anteriormente.

Con el propósito de lograr una comprensión completa de cada paso o fase, se llevará a cabo la implementación en una empresa real para demostrar los resultados esperados y ejemplificar cada concepto de manera práctica.

2.2.2 Metodologías de desarrollo de software

Tomando como referencia a (Morales Carrillo et al., 2022), las metodologías ágiles son aquellas que posibilitan ajustar el enfoque de trabajo a las particularidades del proyecto, logrando así flexibilidad y una respuesta rápida para adaptar el desarrollo del proyecto a las circunstancias específicas del entorno.

Las metodologías ágiles se pueden describir como un conjunto de actividades y enfoques utilizados para gestionar proyectos. Son métodos de desarrollo en los cuales tanto los requisitos como las soluciones evolucionan a lo largo del tiempo, a través del trabajo colaborativo de equipos multidisciplinarios que se destacan por poseer las siguientes características:

- Desarrollo evolutivo y flexible.
- Autonomía de los equipos.
- Planificación.
- Comunicación.

Existen diversas metodologías de desarrollo de software que se utilizan para guiar y estructurar el proceso de creación de software. Estas metodologías proporcionan enfoques, prácticas y herramientas para planificar, diseñar, implementar y mantener proyectos de desarrollo de software de manera efectiva. Algunas de las metodologías más comunes incluyen:

2.2.2.1 Metodología SCRUM

SCRUM es una metodología ágil de gestión de proyectos que se enfoca en el desarrollo de productos de manera iterativa e incremental. Se basa en la colaboración y la comunicación constante entre los miembros del equipo para lograr resultados efectivos.

En SCRUM, los proyectos se dividen en ciclos llamados "*sprints*", que son períodos de tiempo definidos y cortos (generalmente de 1 a 4 semanas). Durante cada sprint, se seleccionan y planifican las tareas que se abordarán y se trabaja en ellas de manera intensiva.

El equipo de desarrollo se organiza en roles claros, como el Scrum Master, que es responsable de facilitar el proceso y eliminar obstáculos, y el *Product Owner*, que representa los intereses del cliente y define los requisitos del producto.

SCRUM se caracteriza por su enfoque en la transparencia, la inspección y la adaptación. A través de reuniones regulares, como las reuniones diarias de seguimiento (*daily stand-ups*) y las reuniones de revisión de sprint (*sprint reviews*), se evalúa el progreso, se ajustan las prioridades y se toman decisiones informadas para guiar el desarrollo del producto.

Esta metodología fomenta la colaboración y la retroalimentación constante, lo que permite a los equipos adaptarse rápidamente a los cambios y entregar incrementos de valor

al cliente de manera iterativa. SCRUM es ampliamente utilizado en el desarrollo de software, pero también se puede aplicar en otros contextos y proyectos.

2.2.2.2 Programación Extrema (XP)

La Programación Extrema (XP, por sus siglas en inglés, *Extreme Programming*) es una metodología ágil de desarrollo de software que se centra en la entrega rápida y continua de software de alta calidad. XP se basa en una serie de prácticas y valores que promueven la colaboración, la adaptabilidad y la mejora constante.

En XP, el trabajo se organiza en iteraciones cortas y frecuentes, llamadas "iteraciones" o "*sprints*", que suelen tener una duración de una a tres semanas. Durante cada iteración, el equipo se enfoca en desarrollar funcionalidades completas y probadas, lo que permite una entrega temprana y continua de valor al cliente.

Las principales prácticas de XP incluyen la programación en parejas, donde dos desarrolladores trabajan juntos en el mismo código; las pruebas unitarias automatizadas, que garantizan la calidad del software; la integración continua, que implica la integración frecuente de las contribuciones de los desarrolladores en un repositorio compartido; y la refactorización, que consiste en mejorar constantemente el diseño y la estructura del código.

Además, XP promueve una comunicación abierta y continua entre los miembros del equipo y los *stakeholders*, lo que ayuda a alinear las expectativas y a garantizar la satisfacción del cliente. También se enfatiza en la simplicidad, evitando la sobre ingeniería y priorizando la entrega de funcionalidades de valor.

La Programación Extrema se basa en una serie de valores fundamentales, como la retroalimentación, la comunicación, la simplicidad, el coraje y el respeto. Estos valores guían las decisiones y acciones del equipo a lo largo del proyecto.

En resumen, XP es una metodología ágil que se centra en la entrega rápida, frecuente y continua de software de calidad, a través de prácticas como la programación en parejas, las pruebas automatizadas y la integración continua, respaldada por valores como la retroalimentación y la comunicación constante.

2.2.2.3 Metodología Data-Driven

La metodología Data-Driven, también conocida como enfoque basado en datos, se refiere a la forma en que las organizaciones toman decisiones y desarrollan estrategias basándose en el análisis de datos y evidencias concretas. Esta metodología se basa en la idea de que los datos son una fuente de conocimiento objetivo y confiable que puede guiar y respaldar la toma de decisiones (Morales Carrillo et al., 2022).

En la metodología Data-Driven, el proceso comienza por identificar las preguntas o problemas a resolver y determinar qué tipo de datos son necesarios para abordarlos. Luego, se recopilan, procesan y analizan los datos relevantes utilizando técnicas y herramientas estadísticas y de análisis de datos. Este análisis ayuda a extraer información valiosa, identificar patrones, tendencias y relaciones entre variables, y generar conocimientos accionables.

Una vez obtenidos los resultados del análisis, se utilizan para respaldar la toma de decisiones informadas. Las decisiones basadas en datos suelen ser más objetivas y fundamentadas, ya que se apoyan en evidencias concretas en lugar de suposiciones o intuiciones.

La metodología Data-Driven promueve el uso de datos en todas las áreas de una organización, desde la planificación estratégica y la gestión de proyectos hasta la mejora de procesos y la optimización de la experiencia del cliente. Ayuda a identificar oportunidades de mejora, a predecir resultados y a evaluar el impacto de las decisiones tomadas.

Es importante destacar que la implementación efectiva de la metodología Data-Driven requiere contar con sistemas adecuados de recopilación, almacenamiento y análisis de datos, así como con personal capacitado en análisis de datos y en interpretación de resultados.

En resumen, la metodología Data-Driven se basa en el análisis de datos para respaldar la toma de decisiones informadas y mejorar los procesos y resultados en una organización. Utiliza técnicas y herramientas de análisis de datos para extraer conocimientos accionables y confiables, y promueve una cultura basada en datos en toda la organización.

2.2.3 Metodologías de Data Warehouse o almacenamiento de datos

Tomando como referencia a (Coronel Valenzuela, 2019), las metodologías de almacenamiento de datos comprenden un conjunto de tareas en común tales como el análisis de requisitos comerciales, el diseño, arquitectura e implementación de datos. Para el análisis de requisitos comerciales, se utilizan técnicas como entrevistas, lluvia de ideas y sesiones JAD.

La entrevista, en el ámbito de la investigación cualitativa, constituye una valiosa herramienta para obtener información. Se caracteriza como una conversación con un propósito específico, distinto del mero acto de platicar. Es un recurso técnico que adopta la estructura de un diálogo informal.

La técnica de lluvia de ideas se emplea en contextos de trabajo en grupo con el propósito de originar nuevas ideas o resolver problemas específicos. En la actualidad, esta técnica se utiliza de manera amplia en reuniones laborales y debates.

El enfoque JAD (*Joint Application Design* o Diseño Conjunto de Aplicaciones) constituye tanto una técnica para establecer requisitos como un método para diseñar interfaces de usuario. Su objetivo es optimizar la participación de los usuarios, agilizar el proceso de desarrollo y elevar la precisión de las especificaciones. Es especialmente adecuado para sistemas personalizados y eficaz en la recolección de requisitos de manera eficiente.

El análisis de requisitos comerciales se utiliza para obtener las necesidades de los usuarios previstos del almacén de datos. Las preguntas comerciales son preguntas analíticas o de apoyo a las decisiones que suelen plantear los gerentes. Una vez que se obtienen todas las preguntas comerciales, se priorizan pidiendo a los usuarios que califiquen las preguntas o estimando el riesgo asociado con las soluciones necesarias para las preguntas. A continuación, se crea un modelo conceptual de muy alto nivel (también conocido como modelo de datos de área temática) de la solución para cada una de las preguntas comerciales.

La tarea de diseño de datos incluye el modelado y la normalización de datos. Las dos técnicas de modelado de datos más populares para el almacenamiento de datos son:

Modelado Entidad-Relacional. - El modelado Entidad-Relacional que sigue el proceso de diseño de base de datos OLTP estándar, comenzando con un diseño conceptual entidad relación (ER), traduciendo el esquema ER en un esquema relacional y luego normalizando el esquema relacional.

Modelo dimensional. - Un modelo dimensional se compone de una tabla de hechos y varias tablas de dimensiones.

Estructura de estrella. - La característica estructura en forma de estrella de la representación física de un modelo dimensional se denomina esquema de unión en estrella, o simplemente esquema en estrella.

Esquema de copo de nieve. - Un modelo dimensional puede extenderse a un esquema de copo de nieve, eliminando los atributos de cardinalidad baja en las dimensiones y

colocándolos en tablas separadas, que se vinculan a la tabla de dimensiones con claves artificiales. En el ámbito OLAP, las consultas de soporte de decisiones pueden requerir una agregación y uniones importantes.

Para mejorar el rendimiento, la desnormalización generalmente se promueve en un entorno de almacenamiento de datos. La arquitectura es un plano que permite la comunicación, la planificación, el mantenimiento, el aprendizaje y la reutilización. Incluye diferentes áreas como diseño de datos, diseño técnico y diseño de infraestructura de hardware y software. La filosofía de diseño de arquitectura tiene su origen en la estrategia de diseño de esquemas de las bases de datos OLTP.

Existen varias estrategias para el diseño de esquemas, como de arriba hacia abajo, de abajo hacia arriba, de adentro hacia afuera y mixtas. Las filosofías de diseño de la arquitectura del almacén de datos se pueden clasificar en términos generales en diseño de almacén de datos para toda la empresa y diseño de centro de datos.

Las metodologías de desarrollo y el análisis y diseño de software son dos áreas fundamentales en el mundo de la ingeniería de software. Estas disciplinas se centran en

proporcionar enfoques estructurados y eficientes para la creación de soluciones de software robustas y efectivas. Las metodologías de desarrollo son marcos de trabajo que guían el proceso de creación de software desde la concepción hasta la implementación y mantenimiento, asegurando la calidad y la entrega exitosa del producto. Por otro lado, el análisis y diseño de software se concentra en comprender los requisitos del sistema, modelar su estructura y funcionalidad, y diseñar soluciones técnicas que satisfagan las necesidades del usuario. En conjunto, estas disciplinas desempeñan un papel esencial en la creación y mejora continua de aplicaciones y sistemas informáticos que impulsan nuestra sociedad digital.

Capítulo tres

Análisis y diseño

Este capítulo se centra en el análisis y diseño de la extracción de datos, con un énfasis en la identificación de fuentes de datos pertinentes y la creación de una estructura adecuada para el almacenamiento y procesamiento de datos. Se aborda la visualización de estos datos en Power BI, destacando la importancia de seleccionar el tipo de visualización apropiado según la naturaleza de los datos y los objetivos de análisis.

3.1 Metodología de desarrollo

Emplear una metodología de desarrollo de software ofrece numerosas ventajas y utilidades para asegurar la eficiencia, calidad y éxito en la creación de aplicaciones y sistemas.

Para el desarrollo del proyecto, se emplea una metodología llamada Hefestos, la cual fue seleccionada debido a su compatibilidad con las particularidades de la institución.

Hefestos se ha establecido como una metodología sólida para la construcción de un Data Warehouse (DW) desde cero, y destaca por su enfoque pragmático que permite su adaptación a diferentes ciclos de vida de software. Esta metodología se centra en el análisis exhaustivo de los requisitos y fuentes de datos para la implementación efectiva del DW.

La metodología Hefesto se utiliza tanto en el diseño del Data Warehouse como en el Data Mart, y se destaca por su claridad y comprensión. Está estructurada en fases que permiten identificar los objetivos y los resultados esperados. Esto significa que la estructura del Data Warehouse se adapta de manera ágil y sencilla, siguiendo los requisitos establecidos por los usuarios (William George et al., 2021).

En la metodología de Hefestos, se logra minimizar la resistencia al cambio por parte de los usuarios finales al involucrarlos en cada etapa del proceso. Esto permite determinar el comportamiento y las funcionalidades que se incorporarán en el diseño del Data Warehouse. Los modelos conceptuales y lógicos que se implementan son de fácil comprensión y análisis.

Además, la metodología y el ciclo de vida que se elijan se mantienen independientes entre sí, lo que garantiza una mayor flexibilidad y adaptabilidad en el desarrollo del proyecto.

3.2 Necesidad de Negocio

El departamento requiere contar con datos y estadísticas precisas para evaluar el rendimiento de los usuarios resolutores. En la actualidad, se carece de información detallada en este aspecto, lo que ha llevado a realizar movimientos de personal basados en percepciones y de manera empírica. Sin embargo, estos enfoques no han generado los resultados esperados.

Por esta razón, es necesario implementar un panel de control que registre las estadísticas y métricas de calidad de servicio. Este panel proporcionará información histórica sobre cada oferta de servicio atendidas, los tiempos de cumplimiento y la cantidad de requerimientos gestionados por cada resolutor y departamento.

Estos datos son útiles para que las gerencias de cada departamento puedan revisar sus procesos internos y asignación de tareas a los resolutores, ya que disponen con la información necesaria para realizar evaluaciones. Además, este panel de control sirve para medir los objetivos internos de cada área.

Asimismo, es necesario contar con un panel de control que facilite el monitoreo en tiempo real y proporcione de manera rápida toda la información relevante. Esto permite identificar de manera sencilla qué equipo de trabajo tiene una carga alta de requerimientos, qué usuarios tienen casos próximos a vencer o que ya han vencido sin ser atendidos. Estos datos son de gran ayuda para mejorar los tiempos de respuesta hacia los usuarios finales y, para fortalecer la imagen del departamento de Interservice ante las otras dependencias de la institución.

3.3 Requerimientos funcionales

La técnica utilizada en el levantamiento de requerimientos funcionales es las entrevistas ya que son las más utilizadas en el proceso de desarrollo de software y en la gestión de proyectos en general. Esta técnica implica la interacción directa entre el equipo de desarrollo y los grupos de interés, como usuarios finales, clientes, gerentes y expertos en

dominio, con el objetivo de recopilar información detallada y precisa sobre las necesidades, expectativas y objetivos del proyecto.

Para lograr una implementación exitosa, es fundamental adquirir un profundo conocimiento de la empresa, su funcionamiento interno, los servicios y productos que ofrece, así como su base de clientes y segmentos. Es especialmente importante comprender las expectativas y objetivos que se esperan alcanzar con la implementación de un sistema de indicadores y métricas de calidad de servicio.

A lo largo del proyecto ha habido un total de cinco reuniones en la empresa además de distintas llamadas para aclarar algunos requerimientos levantados durante las reuniones.

La primera reunión con la subgerencia del área a la cual se le realizó una presentación con la idea para implementar los indicadores de calidad y que beneficios se podría obtener de estos.

En la segunda reunión con la gerencia del departamento, se concretó la idea inicial y se obtuvo la autorización para asignar recursos de tiempo y acceso a diversas fuentes de información para el proyecto.

Las siguientes reuniones fueron con los custodios de la información a la que requerimos acceso para extraer la información y alimentar la base de datos, se socializa las necesidades del área en la construcción de los Dashboards para determinar la forma en la que se tendrá acceso a los datos sin comprometer cualquier degradación en el servicio que está en producción.

Se realiza reuniones de entendimiento de las estructuras de las bases y de la atomicidad de sus registros, en la que luego de la revisión de los datos proporcionados por los custodios, se detectaron varias incongruencias como registros duplicados, error en nombres, cambios de estados por error de la aplicación o manipulación directa en la base de datos.

La última reunión se la realiza con la gerencia del área, se revisa todos los requerimientos y se cierra el alcance inicial con el compromiso de entrega de dos Dashboards en Power BI.

1. El primero debe contener un tablero con indicadores y KPI de toda la herramienta (Remedy), con la que se atienden las diferentes órdenes de trabajo e incidentes.
2. El segundo debe ser un tablero de monitoreo para los grupos resolutores que pertenecen directamente al área de Interservice en la que refleje la carga de trabajo, tiempos de resolución y asignación por analista de todos los requerimientos e incidentes del día.

3.3.1 Introducción

3.3.1.1 Propósito

Mediante este documento, detallaremos los requisitos que el sistema, encargado de suministrar datos al panel de control destinado al departamento de Interservice, deberá satisfacer. Todos los requisitos que se describen a continuación se derivan de un proceso de relevamiento de procesos realizado en la institución. Este proceso determinó qué información debe ser considerada para cada uno de estos requisitos.

3.3.1.2 Ámbito del sistema

El sistema se describirá como los indicadores de Interservice y se enfocará en mejorar la eficiencia en la gestión de la información relacionada con órdenes de trabajo e incidentes dentro del departamento de Interservice. Este sistema permitirá la automatización del proceso de extracción de registros desde las fuentes de datos utilizadas por el almacén de datos.

La principal ventaja que se obtendrá mediante el desarrollo e implementación de este sistema es el acceso a información precisa, respaldada, unificada y almacenada en una base de datos con una estructura de diseño adaptada a las necesidades del departamento.

3.3.1.3 Referencias

Este capítulo se ha elaborado siguiendo las pautas establecidas en el Estándar IEEE 830. Este estándar proporciona una estructura para especificar los requisitos que el sistema debe cumplir de acuerdo con los requisitos de la empresa.

3.3.1.4 Visión general del documento

En este capítulo, se proporcionará una descripción detallada de cómo operará el sistema, incluyendo una presentación visual y una descripción textual de todas las funciones que se espera que el sistema realice. También se abordarán posibles restricciones o limitaciones del sistema, y se justificará su existencia. Además, se identificarán las partes interesadas que interactuarán directamente con el sistema.

3.3.2 Descripción General

3.3.2.1 Perspectiva del producto

El sistema se concentra en la administración de órdenes de trabajo e incidentes, donde los analistas son responsables de dar respuesta a todas las solicitudes, y no requerirá interactuar ni depender de otros programas que puedan estar en funcionamiento en la computadora.

3.3.2.2 Funciones del producto

El sistema facilitará la gestión de:

- Extracción de registros de los orígenes de datos
- Guardar los registros extraídos en un repositorio
- Limpiar los datos
- Insertar los requerimientos en el almacén de datos
- Generar los indicadores.
- Visualización de indicadores a través de dashboards

3.3.2.3 Características del usuario

El sistema estará manejado por el gerente, subgerente y líderes de equipo del departamento interservice.

Gerente

Nivel de formación académica: Título de 3r nivel
Experiencia técnica: Intermedia

Subgerente.

Nivel de formación académica: Universitaria
Experiencia técnica: Intermedia

Líderes de equipo

Nivel de formación académica: Universitaria
Experiencia técnica: Intermedia

Dado el enfoque de diseño del sistema, que se centra en la facilidad de uso, no es necesario que los usuarios tengan un alto nivel de conocimientos en sistemas. No obstante, se proporcionará una demostración de cómo funciona el sistema como una guía para su correcta utilización.

3.3.2.4 Restricciones

Debido a las limitación en licencias de Power BI, con los lideres de equipo se entrega el archivo de indicadores para que ellos lo abran en su laptop, esta aplicación se conecta a la base de datos para sincronizar y actualizar los indicadores.

3.3.2.5 Suposiciones y Dependencias

El sistema está desarrollado para su funcionamiento y compatibilidad con Windows

3.3.2.6 Requisitos futuros

Cualquier sistema tiene el potencial de mejorar y expandirse. Por lo tanto, el sistema podría ser mejorado para incluir métricas a nivel de clientes y empresas que presentan requerimientos, así como proporcionar un mayor nivel de detalle sobre los tiempos de atención por parte de los usuarios resolutores.

3.3.2.7 Requisitos específicos

3.3.2.8 Interfaces.

3.3.2.9 Interfaz de usuario.

Con el fin de garantizar la usabilidad del sistema, la interfaz estará compuesta por una variedad de elementos, que incluyen ventanas, botones, campos de texto, etiquetas de texto, tablas, imágenes, íconos, listas de selección y menús desplegables, entre otros.

3.3.2.10 Interfaz de hardware y software.

Cuando mencionamos hardware, estamos haciendo referencia a los elementos físicos necesarios para el funcionamiento adecuado del sistema. En este contexto, necesitamos una computadora que esté equipada con todos sus dispositivos periféricos.

La computadora debe cumplir con características básicas como:

- Procesador: 2.0 GHz o superior
- Memoria Ram: 8 Gb.
- Espacio de disco duro: 1 Gb.

3.3.2.11 *Funciones.*

3.3.2.12 *Requerimientos funcionales.*

Los requerimientos funcionales para el tablero de indicadores son los siguientes:

RF1. Se requiere contar con un filtro para poder seleccionar un rango de fechas, esta selección debe ser tipo calendario.

RF2. Se requiere que el tablero cuente con un segmento de datos con los tipos de requerimientos atendidos.

RF3. Se requiere un segmento de datos en el cual se pueda seleccionar uno a varios grupos resolutores.

RF4. Gráfico en el cual se dibuje de forma mensual la cantidad de requerimientos atendidos y el porcentaje de SLA Cumplido, este gráfico deberá ser afectado por el rango de fechas seleccionado al igual que el tipo de requerimientos y grupo resolutor.

RF5. Gráfico en la cual se dibuje de forma diaria los últimos 10 días con la cantidad de requerimientos atendidos, este gráfico deberá ser afectado por el tipo de requerimiento y grupo resolutor.

RF6. Gráfico que muestre el total de tickets atendidos por cada resolutor y su porcentaje respectivo, este gráfico deberá ser afectado por el rango de fechas seleccionado al igual que el tipo de requerimientos y grupo resolutor.

RF7. Se requiere un gráfico que muestre el total de ticket atendidos por estado, este gráfico deberá ser afectado por el rango de fechas seleccionado al igual que el tipo de requerimientos y grupo resolutor.

RF8. Gráfico con el número de requerimientos solicitados por departamento, este gráfico deberá ser afectado por el rango de fechas seleccionado al igual que el tipo de requerimientos y grupo resolutor.

RF9. Gráfico que muestre el total de requerimientos atendidos agrupados por el grupo resolutor, en este gráfico también se debe mostrar el porcentaje del SLA cumplido por grupo resolutor, este gráfico deberá ser afectado por el rango de fechas seleccionado al igual que el tipo de requerimientos y grupo resolutor.

RF10. Gráfico en el que se muestre el total de requerimientos atendidos por Oferta de servicio, este gráfico deberá ser afectado por el rango de fechas seleccionado al igual que el tipo de requerimientos y grupo resolutor.

RF11. Gráfico que muestre un mapa de calor por provincias donde se visualice la cantidad de requerimientos ingresados, este gráfico deberá ser afectado por el rango de fechas seleccionado al igual que el tipo de requerimientos y grupo resolutor.

RF12. Tabla con el listado total de requerimientos ingresados por agencia, este gráfico deberá ser afectado por el rango de fechas seleccionado al igual que el tipo de requerimientos y grupo resolutor.

RF13. Gráfico con el total del SLA promedio de toda la herramienta de Remedy, este gráfico debe ser afectado por el rango de fechas seleccionadas al igual que el tipo de requerimientos y grupo resolutor.

RF14. Tabla con el total de ofertas de servicio incumplido, este gráfico deberá ser afectado por el rango de fechas seleccionado al igual que el tipo de requerimientos y grupo resolutor.

RF15. Tabla de datos agrupados por usuario resolutor con el total de casos gestionados, el total de casos en estado pendiente, el porcentaje del SLA de cumplimiento de servicio y el porcentaje correspondiente a los casos puestos en pendientes versus el total gestionado, este gráfico deberá ser afectado por el rango de fechas seleccionado al igual que el tipo de requerimientos y grupo resolutor.

RF16. Membretes que muestre el total de tickets ingresados, el total de tickets gestionados, el porcentaje que corresponde la cantidad de tickets gestionados versus lo ingresado, la cantidad de usuarios resolutores, el número de tickets promedio que se atiende por usuario resolutor, la cantidad de tickets que estuvieron en pendiente, estos membretes deberán ser afectados por el rango de fechas seleccionado al igual que el tipo de requerimientos y grupo resolutor.

RF17. Gráfico que muestre la cantidad y porcentaje de tickets ingresados por los diferentes canales,

La segunda necesidad del departamento es un tablero de monitoreo que permita visualizar las alertas de los requerimientos pendientes o vencidos, específicamente de los grupos resolutores que incluyen a Soporte Banca Empresas, Soporte Banca Personas, Operaciones Contables, Secretaría General y los Técnicos de Interservice.

Los requerimientos funcionales para el tablero de monitoreo son los siguientes:

RF1. Estadísticas por grupos resolutores del área, en las que se pueda tener el total de requerimientos atendidos en los últimos 30 días, el total de casos que se han puesto en pendiente, el total de usuarios resolutores y su SLA.

RF2. Gráfico con el total de requerimientos mensuales de los últimos 13 meses

RF3. Gráfico con el total de requerimientos diarios de los últimos 30 días.

RF4. Gráfico con el porcentaje de casos puestos en pendientes por los grupos resolutores.

RF5. Gráfico con el SLA promedio de todos los grupos resolutores.

RF6. Gráfico con los SLA por grupos resolutores para ir validando el cumplimiento de sus objetivos.

RF7. Gráfico con los motivos de los casos enviados a pendiente.

RF8. Gráfico con la cantidad de tickets que están aún sin resolver agrupados por estados.

RF9. Gráfico con la cantidad de requerimientos que están vencidos y aún no se han entregado la solución agrupados por usuario resolutor.

RF10. Gráfico con la cantidad de requerimientos que están próximos a vencer agrupados por resolutor.

3.3.2.13 *Atributos del sistema*

El sistema operará de manera precisa, ya que su programación garantiza un funcionamiento sin errores en todos los registros y procesos. Se recomienda realizar el mantenimiento del sistema cada seis meses o anualmente para verificar su correcto desempeño, y si el departamento lo requiere, se podrán agregar nuevas funcionalidades.

Dado que el sistema está diseñado para ser compatible con el sistema operativo Windows, podrá instalarse en cualquier computadora que cumpla con los requisitos mínimos de funcionamiento.

3.4 Arquitectura del almacén de datos (DWH)

La definición de una arquitectura de aplicación es crucial en el proceso de desarrollo de software, ya que establece las bases estructurales y funcionales de cómo se diseña y construye la aplicación. La arquitectura de la aplicación es esencialmente el plan maestro que guía la organización y el diseño de los componentes, módulos y sistemas que componen la aplicación final.

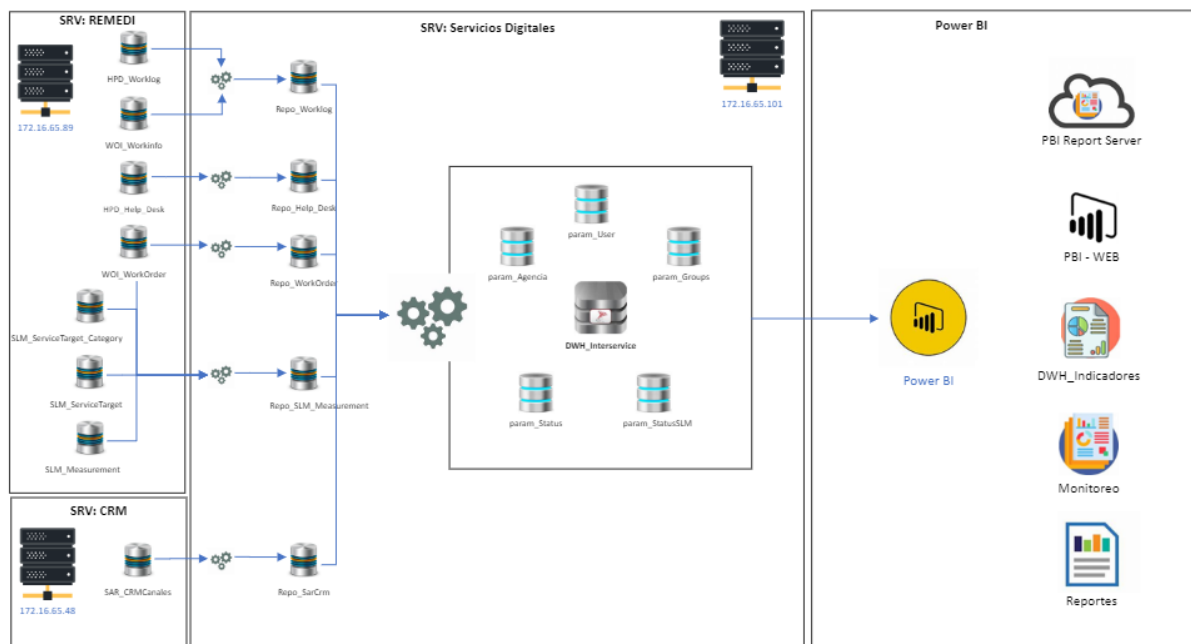
La base de datos para el DWH está diseñada en Microsoft SQL Server, esto debido a que el banco ya cuenta con licencias y los orígenes de datos se encuentran en bases de Datos en Microsoft SQL Server.

La visualización de los dashboards se lo realiza desde la herramienta Power BI y Power BI Report Server para lo cual el departamento de Interservice se hace cargo del costo de 5 licencias de Power BI, sobre Power BI Reporte Server el banco ya cuenta con licencias para todos los usuarios.

La figura 3 muestra la arquitectura definida para la extracción de los registros de los orígenes de datos y su almacenamiento de datos en el repositorio en donde se limpia los datos y se genera la información que alimenta al almacén de datos.

Figura 3

Arquitectura de la extracción y almacenamientos de datos



Fuente: Elaborado por el autor.

3.5 Diseño de Base de datos

El diseño de la base de datos fue realizado utilizando el esquema de estrella, este es un enfoque de modelado maduro ampliamente adoptado por muchos proyectos de DW, se requiere que se clasifiquen las tablas del modelo como dimensiones o hechos.

La tabla de hechos almacena todos los tickets registrados estos pueden ser las órdenes de trabajo o las incidencias reportadas a través de cualquiera de las ofertas de servicio, las tablas de hechos creadas para la DWH son:

- DWH_Interservice
- DWH_Monitor

Las tablas de dimensiones están compuestas por:

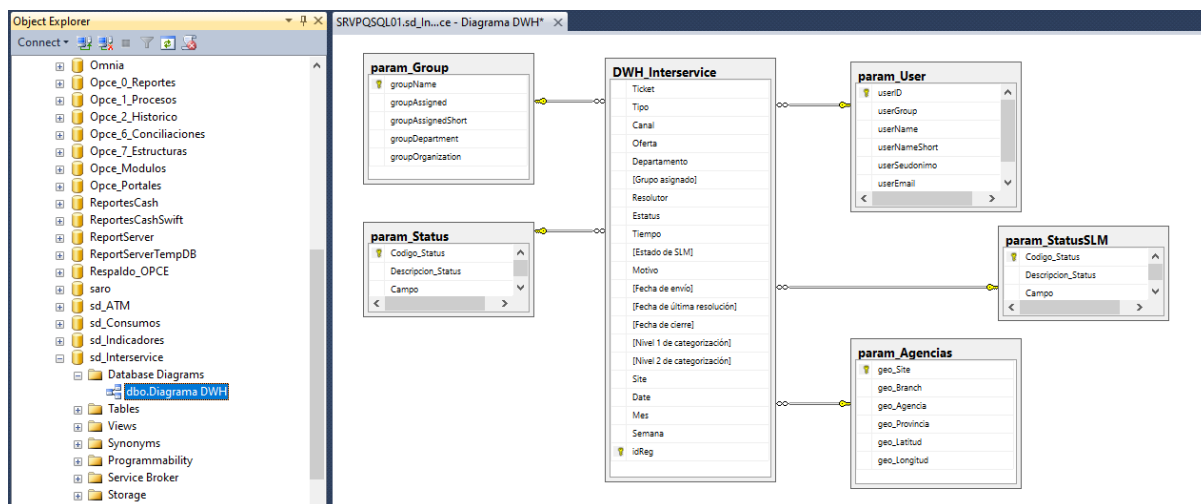
- parm_User
- parm_Agencias
- parm_Group

- parm_Status
- parm_StatusSLM

La figura 4 se muestra el esquema de base de datos del almacén de datos.

Figura 4

Diagrama de base de datos



Fuente: Elaborado por el autor.

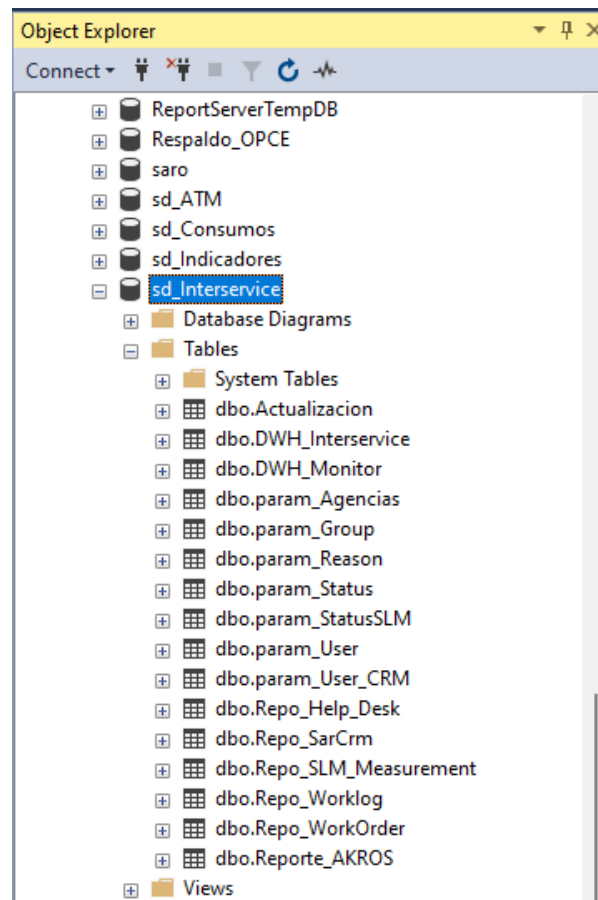
Adicional se tiene un repositorio de los registros extraídos de la base de datos de los servidores en caso de requerir algún tipo de reproceso se lo realizará directamente de estos repositorios evitando así consumir recursos de los servidores de producción.}

Los repositorios son:

- repo_Help_Desk
- repo_WorkOrder
- repo_SarCRM
- repo_Worklogs
- repo_SLM_Measurement

Figura 5

Tablas de la base de datos *sd_Interservice*



Fuente: Elaborado por el autor.

3.6 Tecnologías utilizadas para el almacén de datos

La construcción de un almacén de datos (data warehouse) puede llevarse a cabo utilizando una variedad de herramientas y tecnologías. A continuación, se mencionan algunas de las herramientas comunes que se utilizan en este proceso:

3.6.1 SQL Server

SQL Server es un sistema de gestión de bases de datos relacional desarrollado por Microsoft. Ofrece una amplia gama de servicios y herramientas para el almacenamiento, administración y manipulación de datos. SQL Server es conocido por su capacidad de manejar grandes volúmenes de datos y por su rendimiento, seguridad y confiabilidad.

Proporciona un lenguaje de consulta potente llamado Transact-SQL (T-SQL) que permite realizar consultas, modificaciones y manipulaciones de datos en la base de datos. SQL Server también incluye características adicionales como la capacidad de implementar y administrar sistemas de inteligencia de negocios, integración con otras plataformas y servicios en la nube, y herramientas de análisis y generación de informes.

3.6.2 Extracción, transformación y carga ETL

ETL es un método de integración de datos que involucra tres pasos fundamentales: extraer, transformar y cargar. Este enfoque se utiliza comúnmente para combinar datos provenientes de diversas fuentes y construir un almacén de datos. Durante el proceso de ETL, los datos se extraen de un sistema de origen, se transforman en un formato adecuado para su análisis, y finalmente se cargan en un almacén de datos u otro sistema. Existe también una variante llamada ELT (extraer, cargar, transformar), que consiste en enviar el procesamiento directamente a la base de datos con el objetivo de mejorar el rendimiento.

El proceso de Extracción, Transformación y Carga (ETL) se encarga de transferir datos desde los sistemas de origen a las bases de datos del Data Warehouse. Es responsabilidad clave garantizar la consistencia de los datos en el Data Warehouse, y esto se logra mediante una correcta programación de los procesos ETL. En ocasiones, un proceso ETL puede hacer uso de una base de datos auxiliar, y sus principales funciones son limpiar, depurar y estandarizar los datos, especialmente cuando provienen de diferentes fuentes. Las plataformas de Business Intelligence (BI) ofrecen herramientas y servicios integrados para llevar a cabo las funciones de ETL (Coronel Valenzuela, 2019).

Los procesos ETL se encargan de cargar datos desde las fuentes hacia los almacenes de datos, teniendo en cuenta posibles cambios en los datos a lo largo del tiempo. Los almacenes de datos son principalmente utilizados para la lectura de la información, a diferencia de las bases de datos que permiten la modificación de los datos.

Estos procesos desempeñan un papel crucial al recolectar y posicionar los datos en un lugar que sirve como fuente principal de conocimiento para aquellos interesados en realizar análisis. Aquellos encargados de diseñar y mantener estos procesos ETL, así como

presentar la información a través de herramientas OLAP, tienen la responsabilidad de generar toda la información necesaria para el análisis. Por lo tanto, el diseño y mantenimiento adecuado de estos procesos ETL son fundamentales para el éxito de proyectos que involucren los Data Warehouse.

3.6.3 Power BI

Power BI es una herramienta de Microsoft que permite a los usuarios analizar y compartir información de manera rápida y sencilla.

Esta herramienta permite a los usuarios recopilar datos de diferentes fuentes, como bases de datos, archivos de Excel, servicios web y otras aplicaciones. Una vez que los datos estén disponibles, los usuarios pueden usar visualizaciones para comprender mejor los patrones y tendencias en los datos.

Power BI también permite a los usuarios compartir sus análisis con otros usuarios y colaboradores, lo que hace que sea una herramienta excelente para equipos de trabajo. Esta herramienta se puede utilizar para una variedad de aplicaciones, como el análisis de datos, el seguimiento de indicadores clave de rendimiento, la creación de informes y la toma de decisiones.

La herramienta Power BI ofrece una variedad de ventajas. En primer lugar, es una herramienta de análisis de datos de alto rendimiento con herramientas de visualización intuitivas. Esto permite a los usuarios hacer un seguimiento de los datos en tiempo real y obtener información significativa de los datos.

También permite a los usuarios compartir sus análisis con otros usuarios, lo que facilita la colaboración y el intercambio de información. Además, la herramienta se puede utilizar fácilmente en cualquier dispositivo, lo que permite a los usuarios acceder a sus datos desde cualquier lugar.

Además, Power BI ofrece una variedad de características que lo hacen aún más útil. Por ejemplo, ofrece una función de almacenamiento en la nube, lo que permite a los usuarios guardar y compartir sus datos con otros usuarios. Esta herramienta también permite a los

usuarios crear informes personalizados con información relevante para sus necesidades. Finalmente, Power BI ofrece una variedad de herramientas de gráficos y visualización para ayudar a los usuarios a comprender mejor sus datos.

En resumen, Power BI es una herramienta de análisis de datos útil que permite a los usuarios comprender mejor sus datos. Esta herramienta ofrece una variedad de características útiles, como la capacidad de compartir datos con otros usuarios, crear informes personalizados, almacenar datos en la nube y utilizar herramientas de visualización para mejorar la comprensión de los datos. Esta herramienta es fácil de usar y ofrece una variedad de ventajas para los usuarios que deseen analizar y compartir datos de manera rápida y sencilla.

Capítulo cuatro

Implementación

En este capítulo proporciona una guía completa para implementar una solución de extracción de datos y visualización en Power BI. Ofrece los conocimientos necesarios para aprovechar al máximo esta potente herramienta, asegurando una implementación exitosa y la creación de informes interactivos y visualmente atractivos.

4.1 Origen de datos

Para el desarrollo de la solución se requiere de 8 fuentes de datos, 7 de ellas son de la aplicación de Remedy y una de la aplicación del CRM, cada aplicación se encuentra en servidores distintos.

Un **CRM** o Customer Relationship Management es la gestión de la relación con el cliente, para mejorar la retención de clientes, aumentar las ventas cruzadas y ventas adicionales, y brindar un servicio más personalizado al permitir el manejo de requerimientos.

Desde el servidor de CRM se extraen los registros de la base de datos siguiente:

- SAR_CRMCanales

SAR_CRMCanales, esta base de datos guarda todos los tickets pertenecientes a requerimientos de clientes que son ingresados por el oficial de cuenta a través de las agencias a nivel nacional desde la herramienta del CRM.

Remedy, es una aplicación de mesa de servicio que facilita el manejo de incidentes, órdenes de trabajo y niveles de servicio donde el rol de cliente son todos los empleados de la institución financiera y los resolutores son los usuarios que se encargan de atender y dar respuesta a las órdenes de trabajo o incidencias.

Desde el servidor de Remedy las bases de datos que se extraen son:

- HPD_Help_Desk
- WOI_WorkOrder
- HPD_Worklog
- WOI_Workinfo

- SLM_Measurement

HPD_Help_Desk, esta tabla guarda todos los requerimientos de tipo Incidente, este tipo se define de acuerdo con la oferta de servicio que ocupe el cliente ya que a través de estas ofertas de servicio se reportan errores o incidencias dentro de los sistemas del banco.

La estructura de esta tabla está comprendida por 328 campos de los cuales extraeremos datos de 25 campos.

WOI_WorkOrder, esta tabla guarda todos los requerimientos tipo Orden de Trabajo, este tipo se define de acuerdo con la oferta de servicio que ocupe el cliente ya que a través de estas ofertas de servicio nos solicitan reportes, revisión de información, ajustes contables, etc.

La estructura de esta tabla está comprendida por 66 campos de los cuales los que vamos a necesitar para nuestra solución son 26 campos.

HPD_Worklog, esta tabla guarda todos los logs de cambios de estados de los requerimientos de tipo Incidentes, en esta tabla se guarda toda la trazabilidad de los usuarios y departamentos por cada uno de los estados por el cual se cambió cada requerimiento.

La estructura de esta tabla está comprendida por 45 campos de los cuales los que vamos a necesitar para nuestra solución 25 campos.

WOI_Workinfo, esta tabla guarda todos los logs de cambios de estados y comentarios que se ponen en los Requerimientos tipo Orden de Trabajo.

La estructura de esta tabla está comprendida por 45 campos de los cuales los que vamos a necesitar para nuestra solución 25 campos.

SLM_Measurement, en esta tabla se registran todas las interacciones en las incidencias u ordenes de trabajo que están comprendidas por tareas, es decir está comprendida por varias soluciones por parte de dos o más departamentos.

La figura 6 presenta las fuentes de datos que se utilizarán para la construcción del almacén de datos.

Figura 6*Orígenes de datos*

Fuente: Elaborado por el autor.

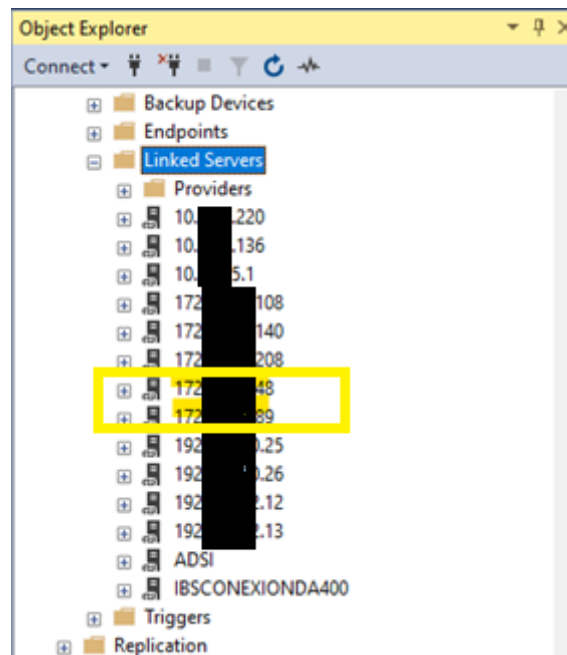
4.2 Extracción de datos

4.2.1 Conexión

La conexión a los servidores donde se encuentra el origen de datos se realiza desde el servidor SQL Server a través de Linked Server a los dos servidores donde se encuentran los orígenes de datos que se utilizará en la construcción del almacén de datos como se muestra en la figura 7.

Figura 7

Link Server al servidor de Remedy



Fuente: Elaborado por el autor.

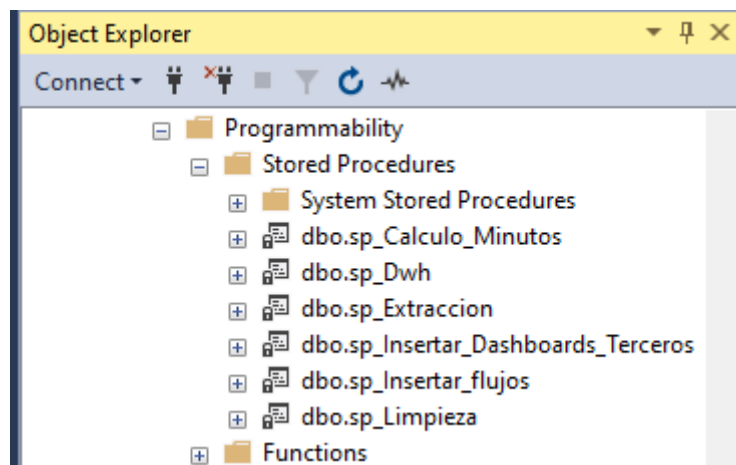
4.2.2 Procedimientos almacenados

Los procedimientos almacenados permiten centralizar la lógica de la base de datos, lo que mejora el modularidad y el rendimiento. Los procedimientos almacenados también permiten a los desarrolladores escribir instrucciones SQL complejas y optimizadas en una sola ubicación, en lugar de escribir códigos redundantes en todos los programas de aplicaciones.

Se realiza el desarrollo de 6 procedimientos almacenados que se muestran en la figura 8, que abarcan todo el proceso ETL con la que extrae datos de las diferentes fuentes, transformar los datos para su uso en el sistema, realiza una limpieza de los datos y cargar los datos en los repositorios finales.

Figura 8

Listado de procedimientos almacenados



Fuente: Elaborado por el autor.

sp_Extraccion.: este procedimiento almacenado realiza la búsqueda y extracción de registros en cada una de las fuentes de datos filtrando por fecha y hora de creación y/o modificación, almacenando los registros en tablas temporales, estos registros que constan en las tablas temporales son buscados y eliminados de las tablas definitivas para asegurar que los nuevos registros insertados en las tablas definitivas sean con los cambios más recientes evitando posibles duplicaciones de registros en las bases definitivas.

sp_Calculo_Minutos: este procedimiento almacenado realiza el cálculo de minutos transcurridos en el que un ticket pasó de un estado abierto a un estado cerrado o resuelto, esté cálculo se divide en dos procesos,

1. Como primer proceso se toma todos los tickets que se abrieron y cerraron en la misma fecha en la que solo se resta la fecha de cierre de la fecha de apertura obteniendo los minutos transcurridos
2. Si el ticket no fue cerrado en la misma fecha de la apertura se realiza la resta de la fecha de cierre de la fecha de apertura considerando que el horario de atención es de 8 horas diarias y de lunes a viernes, es decir si el caso fue abierto un viernes y cerrado

un lunes no serán considerados los días sábado y domingo, y el viernes se tomará solo hasta las 17:30 y el lunes desde las 08:30.

sp_Limpieza: este procedimiento almacenado realiza la limpieza de los registros en las que por falla del aplicativo o por alguna modificación directa a la base de datos se guardaron de forma errada la información.

sp_DWH: este procedimiento almacenado realiza la consolidación de las bases de incidentes, ordenes de trabajo, CRM y control de pendientes en una sola estructura de datos ([sd_Interservice].[dbo].[DWH_Interservice]), ya que de esta base se alimenta la dashboard de indicadores.

sp_insertar_flujos: este procedimiento almacenado carga todos los tickets con modalidad de flujos de trabajo en la base DWH_Interservice, que es la base de la cual se alimenta el dashboard de indicadores.

Sp_insertar_dashboard_terceros: en este procedimiento almacenado se realiza un truncado de la tabla Reporte_AKROS y se insertan todos los tickets atendidos por el departamento técnico, esto debido a que ellos necesitan la información para armar sus propios dashboards e indicadores de todo lo que ellos atienden para control interno de su departamento

4.2.3 Fechas de carga

Las fechas de carga están definidas para que en cada ejecución la fecha inicial sea 30 minutos antes de la fecha y hora actual, esto solo de lunes a viernes, adicional si la hora de ejecución es menor a las 9 am la fecha inicial será 3 días menos a la fecha actual esto para cubrir los tickets ingresados en fines de semana o fuera de horario laboral, las fechas de carga se guardan en la tabla [sd_Interservice].[dbo].[Actualizacion].

La sentencia que cubre dicho propósito es la siguiente:

```
if (Select datepart(dw,getdate())) < 6
begin
    if (Select datepart(hour,getdate()) ) < 9
    begin
        update [sd_Interservice].[dbo].[Actualizacion]
```

```

        set FechaInicial = dateadd(day,-3,getdate())
        , FechaFinal = getdate()
    end
    else
    begin
        update [sd_Interservice].[dbo].[Actualizacion]
        set FechaInicial = dateadd(minute,-30,getdate())
        , FechaFinal = getdate()
    end
end

```

4.2.4 Calendarización

La ejecución de los procedimientos almacenados esta calendarizado para que se ejecute de forma automática de lunes a viernes en horarios de 08:30 a 18:00 con una frecuencia de 30 minutos.

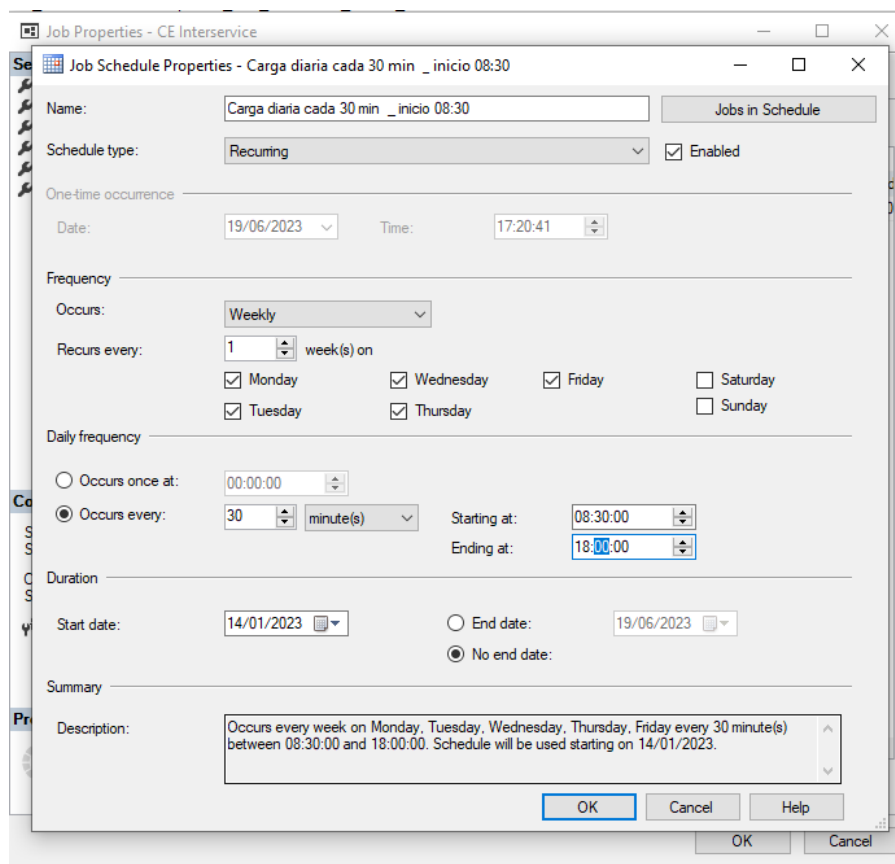
La calendarización se basa en el horario de los analistas que dan atención a los diferentes requerimientos ya que solo se labora de lunes a viernes en horarios de 08:30 a 17:30, salvo cualquier eventualidad que se requiera atención inmediata.

Con esta frecuencia se garantiza que la información en reportes, tableros y monitor estén actualizados y los líderes de equipos dispongan de la información relevante para la toma de decisiones en relación con la distribución de trabajo de los analistas que están a su cargo.

En la figura 9 se muestra la pantalla de configuración de la calendarización de la ejecución automática de este proceso.

Figura 9

Configuración del periodo de ejecución del JOB



Fuente: Elaborado por el autor

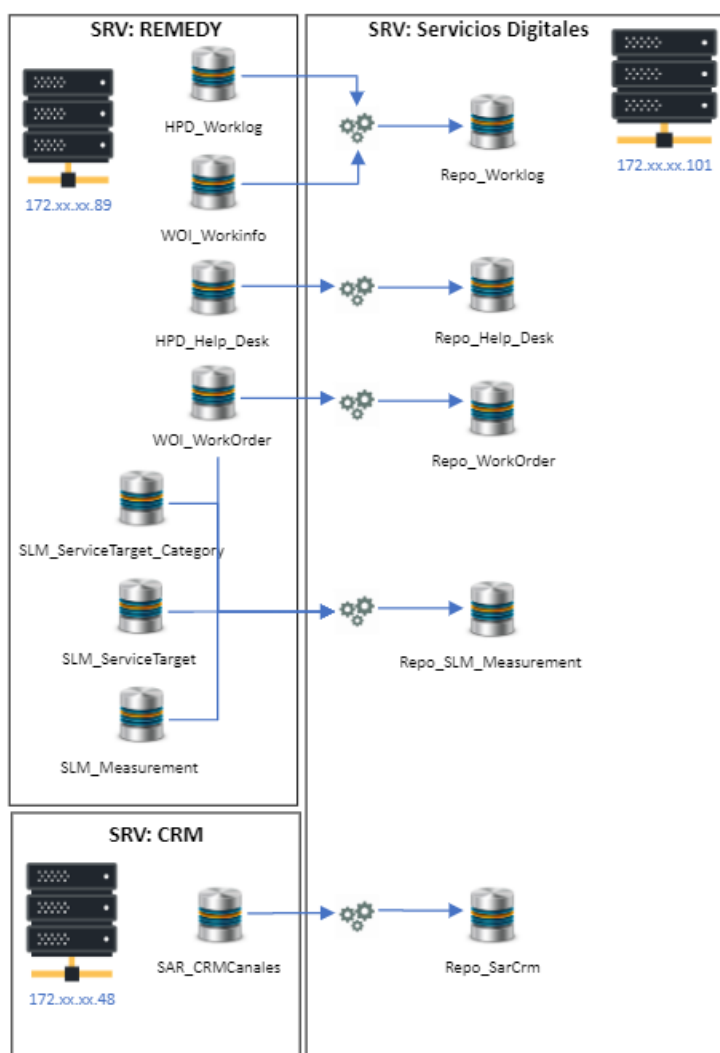
4.2.5 Extracción

Para la extracción de los registros se realiza utilizando el procedimiento almacenado de nombre “**sp_Extraccion**”, este procedimiento almacenado está compuesto de 6 tareas internas en las que extrae todos los registros de los orígenes de datos que cumplan con la condición que la fecha de creación o de modificación de sus registros estén dentro de las fecha de inicio y fecha fin que se encuentra en la tabla [sd_Interservice].[dbo].[Actualizacion], realiza la creación de tablas temporales donde se guarda todos los registros extraídos de los orígenes de datos, se aplica una conversión de datos en los campos de tipo fechas ya que las bases del servidor de Remedy trabaja con los campos fechas tipo numérico y en las tablas de DWH está definido como tipo datetime (Fecha y hora).

Una vez que se carga los registros en las tablas temporales se insertan en las tablas de repositorio controlando la integridad de los datos para evitar duplicaciones en sus registros, este proceso se lo realiza para en caso de que se requiera realizar algún reproceso de información ya no se deba descargar de los servidores donde se encuentra los orígenes de datos si no directamente de los repositorios guardados en el servidor de Servicios digitales, así como se muestra en la figura 10.

Figura 10

Proceso de extracción de las fuentes de datos



Fuente: Elaborado por el autor.

Como Primera tarea tenemos la extracción de las fechas de carga con las que se consultarán en los diferentes orígenes de datos.

La sentencia que cubre dicho propósito es la siguiente:

Figura 11

Sentencia SQL con la que se extrae la fecha de procesamiento

```
-----
-- Extrae las fechas de carga
-----

Declare    @Registros int
           , @FechaInicial datetime
           , @Fechafinal  datetime

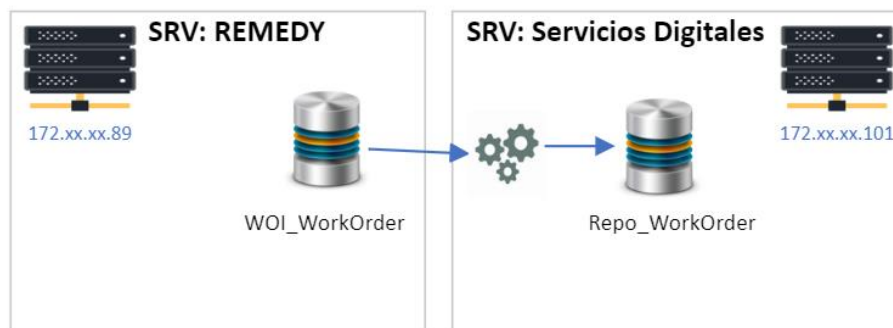
set @FechaInicial  = (Select top 1 Fechainicial from [sd_Interservice].[dbo].[Actualizacion])
set @Fechafinal    = (Select top 1 Fechafinal   from [sd_Interservice].[dbo].[Actualizacion])
```

Fuente: Elaborado por el autor.

La segunda tarea es la extracción de todos los tickets de tipo ordenes de trabajo que hayan sido creadas o modificadas dentro de las fechas inicial y final que se definieron en la tarea 1, posterior a la extracción de los registros serán respaldados en la tabla de repositorio.

Figura 12

Proceso de extracción de datos de las órdenes de trabajo hacia la base de datos del repositorio



Fuente: Elaborado por el autor

La sentencia que cubre dicho propósito es la siguiente:

Figura 13

Sentencia SQL de extracción de registros de tipo órdenes de trabajo

```

-----
-- Extrae registros Ordenes de Trabajo
-----
IF EXISTS (SELECT * FROM sysobjects WHERE type = 'U' AND name = 'WorkOrder_temp')
BEGIN
    DROP TABLE sd_Interservice.dbo.WorkOrder_temp
END

SELECT
    [Work_Order_ID]
    ,[Canal] = 'Interservice'
    ,[Summary]
    ,[Detailed_Description]
    ,[Site]
    ,[Customer_Department]
    ,[Customer_Organization]
    ,[Names] = Last_Name+' '+First_Name
    ,[Requestor_ID] = case when [Requestor_ID] is null then lower(Requestor_By_ID)
                        else lower([Requestor_ID])
                        end
    ,[Submit_Date] = DATEADD(HOUR,-5,DATEADD(SECOND, [Submit_Date], '1970-01-01'))
    ,[Last_Modified_Date] = DATEADD(HOUR,-5,DATEADD(SECOND, [Last_Modified_Date], '1970-01-01'))
    ,[Completed_Date] = DATEADD(HOUR,-5,DATEADD(SECOND, [Completed_Date], '1970-01-01'))
    ,[Customer_Internet_E_mail]
    ,[Business_Service]
    ,[ASGRP]
    ,[ASLOGID] = lower([ASLOGID])
    ,[Request_Assignee]
    ,[email_asignado]
    ,[Priority]
    ,[Status]
    ,[Status_Reason]
    ,[SLM_Status]
    ,[Categorization_Tier_1]
    ,[Categorization_Tier_2]
    ,[Categorization_Tier_3]
    ,[Tiempo] = 0
into [sd_Interservice].[dbo].[WorkOrder_temp]
FROM [172.16.65.89].[ARSystem].[dbo].[WOI_WorkOrder]
where DATEADD(HOUR,-5,DATEADD(SECOND, [Submit_Date], '1970-01-01')) Between @FechaInicial and @Fechafinal
or DATEADD(HOUR,-5,DATEADD(SECOND, [Last_Modified_Date], '1970-01-01')) Between @FechaInicial and @Fechafinal

Delete from [sd_Interservice].[dbo].[Repo_WorkOrder]
where exists (Select 1 from [sd_Interservice].[dbo].[WorkOrder_temp]
              where [WorkOrder_temp].[Work_Order_ID] = [Repo_WorkOrder].[Work_Order_ID])

Insert into [sd_Interservice].[dbo].[Repo_WorkOrder]
Select * FROM [sd_Interservice].[dbo].[WorkOrder_temp]

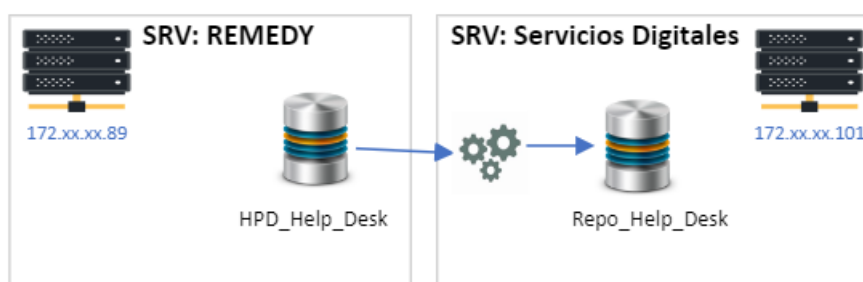
```

Fuente: Elaborado por el autor.

La tercera tarea es la extracción de todos los tickets de tipo incidencias que hayan sido creadas o modificadas dentro de las fechas inicial y final que se definieron en la tarea 1, posterior a la extracción de los registros serán respaldados en la tabla de repositorio.

Figura 14

Proceso de extracción de datos de Incidencias hacia la base de datos de repositorio



Fuente: Elaborado por el autor.

La sentencia que cubre dicho propósito es la siguiente:

Figura 15

Sentencia SQL de extracción de registros de tipo incidencias

```

-----
-- Extrae registros de tipo Incidencias
-----
IF EXISTS (SELECT * FROM sysobjects WHERE type = 'U' AND name = 'Help_Desk_temp')
BEGIN
    DROP TABLE sd_Interservice.dbo.Help_Desk_temp
END

SELECT
    [Incident_Number]
    ,[Canal] = 'Interservice'
    ,[Description]
    ,[Organization]
    ,[Department]
    ,[Site]
    ,[Name] = Last_Name+' '+First_Name
    ,[Customer_Login_ID] = Lower([Customer_Login_ID])
    ,[Internet_E_mail]
    ,[Priority]
    ,[Assigned_Support_Organization]
    ,[Assigned_Group]
    ,[Assignee_Login_ID] = Lower([Assignee_Login_ID])
    ,[Assignee]
    ,[email_asignado]
    ,[Status]
    ,[Status_Reason]
    ,[Submit_Date] = DATEADD(HOUR,-5,DATEADD(SECOND, [Submit_Date],'1970-01-01'))
    ,[Last_Resolved_Date] = DATEADD(HOUR,-5,DATEADD(SECOND, [Last_Resolved_Date],'1970-01-01'))
    ,[Closed_Date] = DATEADD(HOUR,-5,DATEADD(SECOND, [Closed_Date],'1970-01-01'))
    ,[SLM_Status]
    ,[Categorization_Tier_1]
    ,[Categorization_Tier_2]
    ,[Categorization_Tier_3]
    ,[Tiempo] = 0
INTO sd_Interservice.dbo.Help_Desk_temp
FROM [172.16.65.89].[ARSystem].[dbo].[HPD_Help_Desk]
where DATEADD(HOUR,-5,DATEADD(SECOND, [Submit_Date],'1970-01-01')) Between @FechaInicial and @Fechafinal
or DATEADD(HOUR,-5,DATEADD(SECOND, [Last_Resolved_Date],'1970-01-01')) Between @FechaInicial and @Fechafinal

Delete from [sd_Interservice].[dbo].[Repo_Help_Desk]
where exists (Select 1 from [sd_Interservice].[dbo].[Help_Desk_temp]
             where [Help_Desk_temp].Incident_Number = [Repo_Help_Desk].Incident_Number)

Insert into [sd_Interservice].[dbo].[Repo_Help_Desk]
Select * FROM [sd_Interservice].[dbo].[Help_Desk_temp]

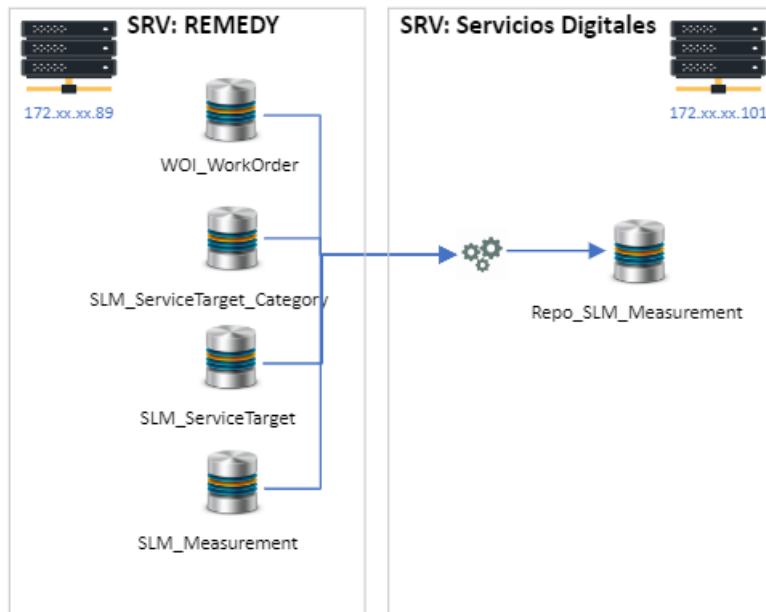
```

Fuente: Elaborado por el autor.

La cuarta tarea es la extracción de todos los tickets de tipo ordenes de trabajo atendidos por flujos que hayan sido creadas o modificadas dentro de las fechas inicial y final que se definieron en la tarea 1, posterior a la extracción de los registros serán respaldados en la tabla de repositorio.

Figura 16

Proceso de extracción de datos de órdenes de trabajo atendidos por flujo hacia la base de datos de repositorio



Fuente: Elaborado por el autor.

La sentencia que cubre dicho propósito es la siguiente:

Figura 17

Sentencia SQL de extracción de registros de tipo órdenes de trabajo por flujo

```

-----
-- Extrae registros de tipo Ordenes de trabajo x Flujo
-----
IF EXISTS (SELECT * FROM sysobjects WHERE type = 'U' AND name = 'SLM_Measurement_temp')
BEGIN
    DROP TABLE sd_Interservice.dbo.SLM_Measurement_temp
END

SELECT distinct
    [SLM_Measurement].[Request_ID]
    , [Work_Order_ID]
    , [Canal] = 'Interservice'
    , [Summary]
    , [Detailed_Description]
    , [Site]
    , [Customer_Department]
    , [Customer_Organization]
    , [Name] = Last_Name+' '+First_Name
    , [Requestor_ID] = Lower([Requestor_ID])
    , [Submit_Date] = DATEADD(HOUR,-5,DATEADD(SECOND, [OverallStartTime],'1970-01-01'))
    , [Last_Modified_Date] = DATEADD(HOUR,-5,DATEADD(SECOND, [OverallStopTime],'1970-01-01'))
    , [Completed_Date] = DATEADD(HOUR,-5,DATEADD(SECOND, [Completed_Date],'1970-01-01'))
    , [Customer_Internet_E_mail]
    , [Business_Service]
    , [Grupo_Resolutor] = Cat_InstanceName
    , [Usuario_Resolutor] = [SLM_Measurement].Last_Modified_By
    , [Request_Assignee] = null
    , [email_asignado]
    , [WOI_WorkOrder].[Priority]
    , [Status] = 'W'+convert(char(1),[WOI_WorkOrder].[Status])
    , [Status_Reason]
    , [SLM_Status]
    , [Categorization_Tier_1]
    , [Categorization_Tier_2]
    , [Categorization_Tier_3]
    , [Tiempo] = (DATEPART(HOUR,convert(datetime, right(uptimechar,8)))*60)+(DATEPART(MINUTE,convert(datetime, right(uptimechar,8)))*1)

INTO sd_Interservice.dbo.SLM_Measurement_temp
FROM [172.16.65.89].[ARSystem].[dbo].[WOI_WorkOrder]
INNER JOIN [172.16.65.89].[ARSystem].[dbo].[SLM_Measurement]
    on WOI_WORKORDER.Work_Order_ID = SLM_Measurement.ApplicationUserFriendlyID
join [172.16.65.89].[ARSystem].[dbo].[SLM_ServiceTarget]
    on SLM_ServiceTarget.InstanceID = SLM_Measurement.SVTInstanceID
INNER JOIN [172.16.65.89].[ARSystem].[dbo].[SLM_ServiceTarget_Category] AS SLM_SERVICETARGET_CATEGORY
    ON SLM_SERVICETARGET_CATEGORY.CategoryId = SLM_ServiceTarget.CategoryId
where (
    DATEADD(HOUR,-5,DATEADD(SECOND, [OverallStartTime],'1970-01-01')) Between @FechaInicial and @FechaFinal
    or DATEADD(HOUR,-5,DATEADD(SECOND, [OverallStopTime],'1970-01-01')) Between @FechaInicial and @FechaFinal
)
and RIGHT(Title,5) = 'Flujo'
and [SLM_Measurement].Last_Modified_By not in ('AR_ESCALATOR','Remedy Application Service')
order by [Work_Order_ID]

delete from sd_Interservice.dbo.Repo_SLM_Measurement
where exists (select 1 from sd_Interservice.dbo.SLM_Measurement_temp where SLM_Measurement_temp.[Request_ID] = Repo_SLM_Measurement.[Request_ID] )

insert into sd_Interservice.dbo.Repo_SLM_Measurement
select * from sd_Interservice.dbo.SLM_Measurement_temp

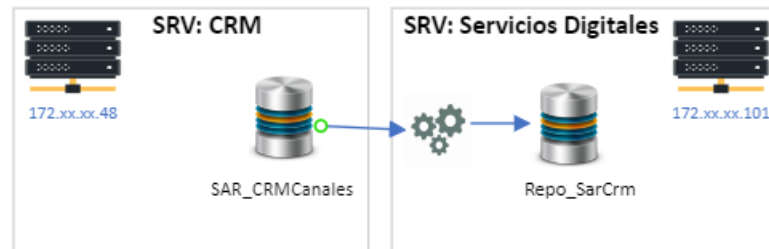
```

Fuente: Elaborado por el autor.

La quinta tarea es la extracción de todos los tickets atendidos a través de la herramienta de CRM que hayan sido creadas o modificadas dentro de las fechas inicial y final que se definieron en la tarea 1, posterior a la extracción de los registros serán respaldados en la tabla de repositorio.

Figura 18

Proceso de extracción de datos de requerimientos ingresados en el SAR hacia la base de datos de repositorio



Fuente: Elaborado por el autor.

La sentencia que cubre dicho propósito es la siguiente:

Figura 19

Sentencia SQL de extracción de registros desde el CRM

```

----- Insertar registros de CRM
-----
IF EXISTS (SELECT * FROM sysobjects WHERE type = 'U' AND name = 'SarCrM_temp')
    BEGIN
        DROP TABLE sd_Interservice.dbo.SarCrM_temp
    END

SELECT *
INTO sd_Interservice.dbo.SarCrM_temp
FROM [Respaldo_OPCE].[dbo].[SAR_CRMCanalesElectronicos]
WHERE convert(datetime,FechaAlta) Between @FechaInicial and @FechaFinal
or convert(datetime,Solucion) Between @FechaInicial and @FechaFinal

delete from sd_Interservice.dbo.Repo_SarCrM
where exists (Select 1 from sd_Interservice.dbo.SarCrM_temp where SarCrM_temp.Reclamo =Repo_SarCrM.Reclamo )

insert into sd_Interservice.dbo.Repo_SarCrM
SELECT [reclamo]
,[Tipo]
,[SubTipo]
,[Motivo]
,[clIdentificacion]
,[IdCliente]
,[Apellido]
,[Nombre]
,[Banca]
,[FechaAlta]
,[UsuarioAlta]
,[NSectorAlta]
,[Solucion]
,[DEstado]
,[Vencimiento]
,[dias]
,[UsuarioCambioEstado]
,[UsuarioDerivado]
,[NSector]
,[Observacion]
,[Respuesta]
,[NumeroOficio]
,[Favorable]
,[Costo]
,[ValorRespuesta]
,[NumeroCajeroAutomatico]
,[Areaquerechaza]
,[DescripcióndelComercio]
,[DescripcionTransSistema]
,[SBrucEmpresasCash]
,[ObservacionInicial]
,[Campo]
,[ReclamoActual]
FROM [sd_Interservice].[dbo].[SarCrM_temp]

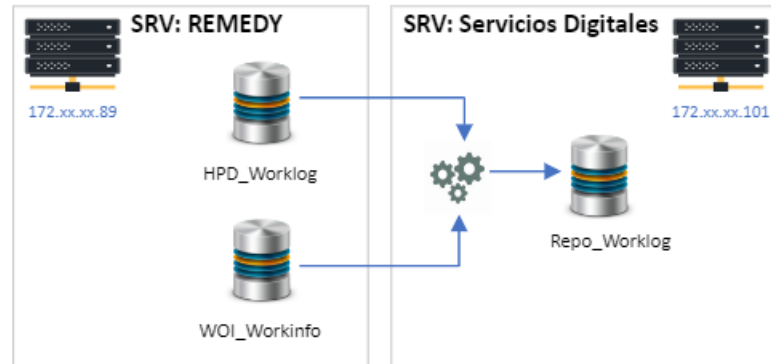
```

Fuente: Elaborado por el autor.

La sexta tarea es la extracción de todos los tickets de tipo ordenes de trabajo O incidente que hayan sido puesta en estado pendiente por el analista resolutor dentro de las fechas inicial y final que se definieron en la tarea 1, posterior a la extracción de los registros serán respaldados en la tabla de repositorio.

Figura 20

Proceso de extracción de datos de órdenes de trabajo e incidentes que se han puesto en estado pendiente hacia la base de datos de repositorio



Fuente: Elaborado por el autor.

La sentencia que cubre dicho propósito es la siguiente:

Figura 21

Sentencia SQL de extracción de registros de los logs de órdenes de trabajo o incidentes en estado pendiente

```

-----
-- Insertar Logs Incidentes (Tickets en estado Pendientes)
-----
IF EXISTS (SELECT * FROM sysobjects WHERE type = 'U' AND name = 'Worklog_temp')
BEGIN
    DROP TABLE sd_Interservice.dbo.Worklog_temp
END

Select
    [Incident_Number]
    ,[Work_Log_ID]      = [Incident_Number]+'-'+[Work_Log_ID]
    ,[Submitter]       = lower([Submitter])
    ,[Submit_Date]     = DATEADD(HOUR,-5,DATEADD(SECOND, [Submit_Date],'1970-01-01'))
    ,[Status Marked]   = 'Pending'
    ,[Reason]          = substring(Detailed_Description,CHARINDEX('Reason:',Detailed_Description,1)+8,50)
    ,[Detailed_Description]
    ,[logDepartment]   = null
into sd_Interservice.dbo.Worklog_temp
FROM [172.16.65.89].[ARSystem].[dbo].[HPD_Worklog]
where Description = 'Status Change Reason'
and DATEADD(HOUR,-5,DATEADD(SECOND, [Submit_Date],'1970-01-01')) Between @FechaInicial and @FechaFinal
and Detailed_Description like 'Status Marked: Pending%'
union
Select
    [Work_Order_ID]
    ,[Work_Log_ID]      = [Work_Order_ID]+'-'+[Work_Log_ID]
    ,[Submitter]       = lower([Submitter])
    ,[Submit_Date]     = DATEADD(HOUR,-5,DATEADD(SECOND, [Submit_Date],'1970-01-01'))
    ,[Status Marked]   = 'Pending'
    ,[Reason]          = substring(Detailed_Description,CHARINDEX('Reason:',Detailed_Description,1)+8,50)
    ,[Detailed_Description]
    ,[logDepartment]   = null
FROM [172.16.65.89].[ARSystem].[dbo].[WOI_Workinfo]
where Description = 'Status Change Reason'
and DATEADD(HOUR,-5,DATEADD(SECOND, [Submit_Date],'1970-01-01')) Between @FechaInicial and @FechaFinal
and Detailed_Description like 'Status Marked: Pending%'

delete from sd_Interservice.dbo.Repo_Worklog
where exists (select 1 from sd_Interservice.dbo.Worklog_temp
              where Worklog_temp.[Work_Log_ID] = Repo_Worklog.[Work_Log_ID] )

insert into sd_Interservice.dbo.Repo_Worklog
select * from sd_Interservice.dbo.Worklog_temp

```

Fuente: Elaborado por el autor.

4.3 Limpieza de datos

El proceso de limpieza se ejecuta sobre los nuevos registros insertados en los repositorios Repo_WorkOrder, Repo_Help_Desk y Repo_SLM_Measurement ya que en estas se guardó los tickets que fueron atendidos por Ordenes de Trabajo, Incidencias y Ordenes por Flujos de trabajo extraídas de los orígenes de datos.

Como primera limpieza se regulariza sobre los registros antes extraídos los campos de Canal, oferta de trabajo, departamento y organización de todos los requerimientos que se

crean por procesos automáticos como son la campaña de reversos para incentivo digital, y la generación de usuarios temporales como se observa en la figura 22.

La sentencia que cubre dicho propósito es la siguiente:

Figura 22

Sentencia SQL para la regularización de los registros creados por procesos automáticos.

```
update sd_Interservice.dbo.Repo_WorkOrder
set   [Canal]           = 'Robot'
,     [Summary]         = 'Reversos incentivos canales'
,     [Customer_Department] = 'INTERSERVICE'
,     [Customer_Organization] = 'VICEPRESIDENCIA DE OPERACIONES Y TECNOLOGIA'
where Summary           like '%RV: Reversos Incentivo%'
or    Summary           like '%RV: Generación Usuarios%'
and ( Submit_Date between @FechaInicial and @FechaFinal
      or Last_Modified_Date between @FechaInicial and @FechaFinal)
```

Fuente: Elaborado por el autor.

En la segunda limpieza de datos se regulariza los registros que son creados a través del correo electrónico, esto debido a que se tienen un buzón de soporte de banca empresas a los que los clientes envían su requerimiento enviando un correo a este buzón, este buzón a su vez remite el correo a un servicio de herramienta Interservice para que se cree un ticket de tipo Orden de Trabajo,

La sentencia que cubre dicho propósito es la siguiente:

Figura 23

Sentencia SQL para regularizar los registros creados a través del buzón del call center

```
update sd_Interservice.dbo.Repo_WorkOrder
set   [Canal]           = 'Buzón'
,     [Summary]         = 'Error o Soporte Plataforma Banca Online Empresas'
,     [Customer_Department] = 'CLIENTE EXTERNO'
,     [Customer_Organization] = 'VICEPRESIDENCIA DE OPERACIONES Y TECNOLOGIA'
where [Requestor_ID]     = 'binsoportempresas'
and   [Business_Service] <>'Banca Online Empresas'
and ( Submit_Date between @FechaInicial and @FechaFinal
      or Last_Modified_Date between @FechaInicial and @FechaFinal)
```

Fuente: Elaborado por el autor.

Posterior a estas se realiza la corrección de los campos de canal de todos los tickets ingresados a través de una aplicación WEB que consume los servicios de la Herramienta Interservice y que la utilizan de manera exclusiva los clientes de Banca Empresas para el ingreso de requerimientos de tipo ordenes de trabajo o de tipo incidente.

La sentencia que cubre dicho propósito es la siguiente:

Figura 24

Sentencia SQL para regularizar el canal en los registros creados desde el portal WEB

```

update sd_Interservice.dbo.Repo_WorkOrder
set    [Canal]                = 'Web'
,      [Customer_Department] = 'CLIENTE EXTERNO'
where  [Requestor_ID]        = 'binsoportempresas'
and (  Submit_Date between @FechaInicial and @Fechafinal
      or Last_Modified_Date between @FechaInicial and @Fechafinal)

update sd_Interservice.dbo.Repo_Help_Desk
set    [Canal]                = 'Web'
,      [Department]           = 'CLIENTE EXTERNO'
where  [Customer_Login_ID]    = 'binsoportempresas'
and (  Submit_Date between @FechaInicial and @Fechafinal
      or Last_Resolved_Date between @FechaInicial and @Fechafinal)

```

Fuente: Elaborado por el autor.

Estas regularizaciones son necesarias para poder identificar correctamente la vía por la cual son ingresados los tickets, debido a que esta información alimenta un indicador de derivación ya que un objetivo del departamento es que los clientes ya no utilicen el buzón de soporte y utilicen la aplicación WEB para el reporte de sus requerimientos.

4.4 Tablas de dimensiones

Antes de proceder con la integración de datos, se carga la información de las tablas de dimensiones ya que al ser una base de datos relacionada no permitirá insertar algún registro que no exista en las tablas de dimensiones.

En tabla de dimensiones de usuarios a diferencia de las otras tablas de dimensiones se realiza un proceso adicional al de cargar los nuevos registros y es el de actualizar los campos de grupo resolutor de los registros ya existentes, este proceso se realiza debido a que existen cambios de departamento o de funciones de los usuarios resolutores.

La sentencia que cubre dicho propósito es la siguiente:

Figura 25

Sentencia SQL para insertar nuevos registros en la tabla de dimensión de usuarios

```

INSERT INTO [sd_Interservice].[dbo].[param_User]
SELECT
    [idUser]
    ,[Groupname]
    ,[Names]
    ,[NameShort]
    ,[Seudonimo] = null
    ,[Email]
FROM [172.16.65.89].[ARSystem].[dbo].[Users]
WHERE NOT EXISTS (Select 1 from sd_Interservice.dbo.param_User where userID = idUser )

update [sd_Interservice].[dbo].[param_User]
set [userGroup] = [groupName]
from [172.16.65.89].[ARSystem].[dbo].[Users]
where userID = idUser

```

Fuente: Elaborado por el autor.

Para la tabla de dimensiones de agencias se realiza la inserción de los nuevos registros creado en el origen de datos, a estos no se aplican actualizaciones sobre los registros anteriores debido a que una agencia no puede cambiar de ubicación o de nombre.

La sentencia que cubre dicho propósito es la siguiente:

Figura 26

Sentencia SQL para insertar registros en la tabla de dimensiones de agencia

```

insert into sd_Interservice.dbo.param_Agencias
Select distinct
    [geo_site]          = [Site]
    ,geo_Branch        = [codSite]
    ,geo_agencia       = [Site]
    ,geo_Provincia     = [locations]
    ,geo_Latitud       = [latitud]
    ,geo_Longitud      = [Longitud]
from [172.16.65.89].[ARSystem].[dbo].[Branch]
where not exists (Select 1 from sd_Interservice.dbo.param_Agencias where [geo_Site] = [Site] )

```

Fuente: Elaborado por el autor.

La tabla de dimensiones de grupos de resolutores se realiza la inserción de los nuevos registros creado en el origen de datos, a estos no se aplican actualizaciones sobre los registros anteriores debido a que un grupo no debe ser renombrado solo creado o inactivado.

La sentencia que cubre dicho propósito es la siguiente:

Figura 27

Sentencia SQL para insertar registros en la tabla de dimensiones de grupos

```

insert into sd_Interservice.dbo.param_Group
Select
    groupName          = [Assigned_Group]
  , assigGroup         = [Assigned_Group]
  , groupnameShort    = null
  , groupDepartment   = [Customer_Department]
  , groupOrganization = [Assigned_Support_Organization]
from [172.16.65.89].[ARSystem].[dbo].[ASGRP]

```

Fuente: Elaborado por el autor.

La tabla de dimensiones para estados y estados de atención se realiza la inserción de los registros creados, al ser estados ya definidos en la plataforma se realiza por una sola ocasión esta carga ya que sus registros no varían en el tiempo.

La sentencia que cubre dicho propósito son las siguientes:

Figura 28

Sentencia SQL para insertar registros en la tabla de dimensiones de estados

```

insert into sd_Interservice.dbo.param_Status
Select
    Codigo_Status      = [Status]
  , Descripcion_Status = [Status_Reason]
  , campo              = null
from [172.16.65.89].[ARSystem].[dbo].[service_status]

insert into sd_Interservice.dbo.param_StatusSLM
Select
    Codigo_Status      = [SLM_Status]
  , Descripcion_Status = [Description]
  , campo              = null
from [172.16.65.89].[ARSystem].[dbo].[SLM_ServiceTarget_Category]

```

Fuente: Elaborado por el autor.

4.5 Integración de datos

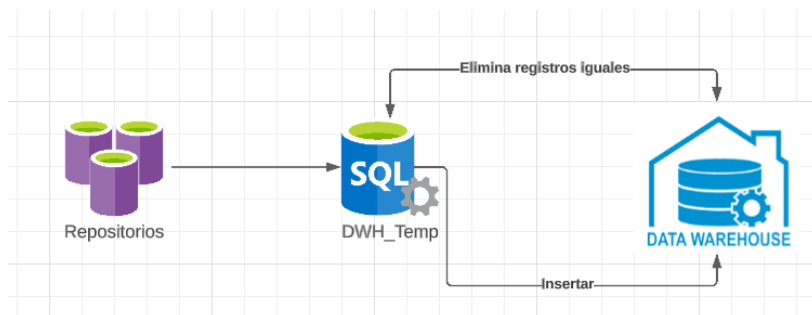
Con los registros extraídos de los orígenes de datos y almacenados en los repositorios, se procede a alimentar el DWH del cual tomará la información para los tableros del Dashboard.

Primeramente, se extrae de las bases repositorios que contienen los tickets de tipo orden de trabajo, incidencias y ordenes de trabajo por flujo para insertarlas en una tabla

temporal, posterior a esto se elimina de la DWH todos los registros que existen en la tabla temporal para asegurar que los nuevos registros que se inserten en esta tabla sean los más recientes y evitar duplicación de algún ticket.

Figura 29

Proceso de integración de datos



Fuente: Elaborado por el autor.

Para la extracción de los registros se realiza utilizando el procedimiento almacenado de nombre “**sp_Dwh**”, este procedimiento almacenado recopila información de 4 repositorios con registros que se encuentren dentro del rango de fechas definido al inicio del proceso.

Estos registros son almacenados en una base temporal debido a que antes de insertar en la base definitiva son depurados para asegurar la no duplicidad de registros y a la vez asegura insertar los registros con los últimos cambios.

La sentencia con la que se inserta los registros desde los repositorios hacia la base temporal es la siguiente:

```
INSERT INTO [sd_Interservice].[dbo].[DWH_Interservice_temp]
SELECT
    [Ticket]                = Incident_Number
    ,[Tipo]                 = 'Incidencia'
    ,[Canal]
    ,[Oferta]               = [Description]
    ,[Departamento]        = [Department]
    ,[Usuario Solicitante]  = [Customer_Login_ID]
    ,[Grupo asignado]       = [Assigned_Group]
    ,[Resolutor]           = [Assignee_Login_ID]
    ,[Estatus]              = 'I'+Convert(char,Status)
    ,[Tiempo]               = case when tiempo = 0 then
datediff(MINUTE,[Submit_Date],[Closed_Date]) else tiempo end
    ,[Estado de SLM]        = [SLM_Status]
    ,[Motivo]               = null
```

```

    ,[Date] = convert(date,[Submit_Date])
    ,[Fecha de envío] = [Submit_Date]
    ,[Fecha de última resolución] = [Last_Resolved_Date]
    ,[Fecha de cierre] = [Closed_Date]
    ,[Nivel 1 de categorización] = [Categorization_Tier_1]
    ,[Nivel 2 de categorización] = [Categorization_Tier_2]
    ,[Site]
    ,[Cliente] = '88888888888888'
FROM [sd_Interservice].[dbo].[Repo_Help_Desk]
where (Submit_Date between @FechaInicial and @Fechafinal
or Last_Resolved_Date between @FechaInicial and @Fechafinal)
union
SELECT
    [Ticket] = [Work_Order_ID]
    ,[Tipo] = 'Orden de Trabajo'
    ,[Canal]
    ,[Oferta] = [Summary]
    ,[Departamento] = [Customer_Department]
    ,[Usuario Solicitante] = [Requestor_ID]
    ,[Grupo asignado] = [ASGRP]
    ,[Resolutor] = [ASLOGID]
    ,[Estatus] = 'W'+Convert(char,Status)
    ,[Tiempo] = case when tiempo = 0 then
datediff(MINUTE,[Submit_Date],[Completed_Date]) else tiempo end
    ,[Estado de SLM] = [SLM_Status]
    ,[Motivo] = null
    ,[Date] = convert(date,[Submit_Date])
    ,[Fecha de envío] = [Submit_Date]
    ,[Fecha de última resolución] = [Last_Modified_Date]
    ,[Fecha de cierre] = [Completed_Date]
    ,[Nivel 1 de categorización] = [Categorization_Tier_1]
    ,[Nivel 2 de categorización] = [Categorization_Tier_2]
    ,[Site]
    ,[Cliente] = '88888888888888'
FROM [sd_Interservice].[dbo].[Repo_WorkOrder]
where (Submit_Date between @FechaInicial and @Fechafinal
or Last_Modified_Date between @FechaInicial and @Fechafinal)
union
SELECT
    [Ticket] = 'CRM-'+rtrim(convert(char,reclamo))
    ,[Tipo] = 'SAR'
    ,[Canal] = 'CRM'
    ,[Oferta] = SubTipo
    ,[Departamento] = Banca
    ,[Usuario Solicitante] = null
    ,[Grupo asignado] = Departamento
    ,[Resolutor] = UsuarioDerivado
    ,[Estatus] = case Destado when 'Reclamo Atendido' then
'C2'

    when 'Rechazados Asignados' then 'C2'

    else 'C1'

end
    ,[Tiempo] = case dias when 0 then 1440 else dias * 1440 end

```

```

,[Estado de SLM]           = case when Solucion > Vencimiento then 3 else 1 end
,[Motivo]                  = null
,[Date]                    = convert(date,FechaAlta)
,[Fecha de envío]         = convert(datetime, FechaAlta)
,[Fecha de última resolución] = Solucion
,[Fecha de cierre]        = Vencimiento
,[Nivel 1 de categorización] = Motivo
,[Nivel 2 de categorización] = null
,[Site]                   = NSectorAlta
,[Cliente]                 = '88888888888888'
FROM sd_Interservice.dbo.Repo_SarCrm
left join sd_Interservice.dbo.param_User_CRM
on UsuarioDerivado = BIN
WHERE (    convert(datetime, FechaAlta) between @FechaInicial and @FechaFinal
        or Solucion between @FechaInicial and @FechaFinal)
and      exists (select 1 from [sd_Interservice].[dbo].[param_User_CRM] where
[param_User_CRM].[BIN] = Repo_SarCrm.[UsuarioDerivado])

```

Una vez insertado los registros en la base temporal se realiza un proceso de eliminado de registros en la tabla definitiva de los registros que se encuentran en la tabla temporal, esto para garantizar que se insertarán registros únicos y los más actuales, ya que en la tabla temporal se encuentran los registros que fueron creados o modificados en el rango de fechas señalados al inicio del proceso.

La sentencia con la cual se realiza dicho proceso es al siguiente:

```

DELETE FROM [sd_Interservice].[dbo].[DWH_Interservice]
WHERE EXISTS (
    SELECT 1
    FROM [sd_Interservice].[dbo].[DWH_Interservice_temp]
    WHERE
    [DWH_Interservice_temp].[Ticket] = [DWH_Interservice].[Ticket]
)

```

Una vez depurado los registros, se realiza la inserción de todos los registros de la base temporal a la base definitiva, la misma se lo realiza con la siguiente sentencia.

```

INSERT INTO [sd_Interservice].[dbo].[DWH_Interservice]
SELECT
    [Ticket]
    ,[Tipo]
    ,[Canal]
    ,[Oferta]
    ,[Departamento]
    ,[Usuario Solicitante]
    ,[Grupo asignado]
    ,[Resolutor]
    ,[Estatus]
    ,[Tiempo]
    ,[Estado de SLM]
    ,[Motivo]
    ,[Date]
    ,[Fecha de envío]
    ,[Fecha de última resolución]
    ,[Fecha de cierre]
    ,[Nivel 1 de categorización]
    ,[Nivel 2 de categorización]
    ,[Site]
    ,[Cliente] = '88888888888888'
FROM [sd_Interservice].[dbo].[DWH_Interservice_temp]

```

Para el tablero de monitoreo se realiza el borrado completo de la base y la inserción de los registros de los últimos 30 días, ya que este tablero es de monitoreo interno del departamento no se requiere de todos los campos por tal motivo no se diseñó en un esquema de estrella y la misma toma datos de las tablas de dimensiones a través de uniones.

La sentencia que cubre dicho propósito es la siguiente:

```

TRUNCATE TABLE [sd_Interservice].[dbo].[DWH_Monitor]
INSERT INTO [sd_Interservice].[dbo].[DWH_Monitor]
SELECT
    [Ticket]
    ,[Tipo]
    ,[Canal]
    ,[Oferta]
    ,[Departamento]
    ,[Grupo asignado] = pGr.groupAssignedShort
    ,[Resolutor] = pus.userNameShort
    ,[Estatus] = pst.Descripcion_Status
    ,[Tiempo]
    ,[Estado de SLM] = pslm.Descripcion_Status
    ,[Motivo]
    ,[Fecha de envío]
    ,[Fecha de cierre]
from [sd_Interservice].[dbo].[DWH_Interservice] as DWH
left join [sd_Interservice].[dbo].[param_Group] as pGr
on DWH.[Grupo asignado] = Pgr.groupName

```

```

left join [sd_Interservice].[dbo].[param_User]      as pUs
      on DWH.Resolutor = pUs.userID
left join [sd_Interservice].[dbo].[param_Status]   as pSt
      on DWH.Estatus = pSt.Codigo_Status
left join [sd_Interservice].[dbo].[param_StatusSLM] as pSlm
      on DWH.[Estado de SLM] = pSlm.Codigo_Status
where convert(date,[Fecha de envío]) >= convert(date,getdate()-30)
and   pGr.groupDepartment ='INTERSERVICE'
and   Tipo <> 'Control'

```

4.6 Carga Inicial

Para la carga inicial de registros al DWH se realizó un script en la que se define una fecha de inicio y una fecha de fin para que los procedimientos almacenados se ejecuten dentro de un bucle en la que la fecha se vaya incrementando de forma diaria hasta que sea igual a la fecha final.

La sentencia que cubre dicho propósito es la siguiente:

```

Declare @fini datetime, @ffin datetime
set @fini      = convert(date,'2022-01-05')
set @ffin      = convert(date,getdate()-1)
Select @fini , @ffin

while @fini <= @ffin
begin
    update [sd_Interservice].[dbo].[Actualizacion]
    set Fechalnicial = convert(datetime,@fini)
    ,      Fechafinal = convert(datetime,dateadd(day,1,@fini))

    execute [sd_Interservice].[dbo].[sp_Extraccion]
    execute [sd_Interservice].[dbo].[sp_Limpieza]
    execute [sd_Interservice].[dbo].[sp_Dwh]
    execute [sd_Interservice].[dbo].[sp_Insertar_flujos]

    set @fini      = dateadd(day,1,@fini)
end

```

4.7 Visualización

Los dashboards se visulizan a través de la herramienta de Power BI y Power BI Report Server, los tableros se actualiza con una frecuencia de 30 minutos, esta información ayuda a los líderes de cada equipo de trabajo a tomar decisiones rápidamente cuando se vea que

algún usuario resolutor esta sobrecargado de tickets o con tickets que están próximos a vencer.

El diseño del dashboard es un aspecto fundamental en la creación de una interfaz efectiva para la presentación de indicadores y datos. Aquí hay algunas consideraciones clave que se toma en cuenta para el diseño de los dashboards.

Claridad y simplicidad: Manteniendo el diseño lo más claro y simple posible. Evitando el desorden visual y asegúrate de que los elementos importantes sean fácilmente identificables.

Diseño responsivo: Asegurar de que el diseño del dashboard sea responsivo, es decir, que se adapte correctamente a diferentes dispositivos y tamaños de pantalla, como computadoras de escritorio, tablets y teléfonos móviles.

Organización lógica: Agrupa los elementos de manera lógica y coherente, usando paneles o secciones para separar diferentes conjuntos de datos o métricas.

Selección de colores adecuada: Elegir una paleta de colores que sea atractiva y que también permita una fácil lectura de los datos. Evita colores demasiado brillantes o contrastantes que puedan distraer.

Uso de gráficos efectivos: Seleccionamos gráficos y visualizaciones adecuadas para representar los datos. Los gráficos de barras, gráficos circulares, gráficos de líneas y mapas ajustando a la naturaleza de los datos.

Jerarquía de información: Destacar las métricas más importantes y colocar elementos clave en la parte superior o en lugares prominentes del dashboard.

Interactividad: Agregar elementos interactivos, como filtros o controles deslizantes, para permitir a los usuarios explorar los datos por sí mismos.

El diseño del dashboard debe centrarse en brindar una experiencia de usuario intuitiva y efectiva, permitiendo a los usuarios acceder y comprender rápidamente la información que necesitan para tomar decisiones informadas, así como lo podemos ver en las figuras 30 y 31.

Figura 30

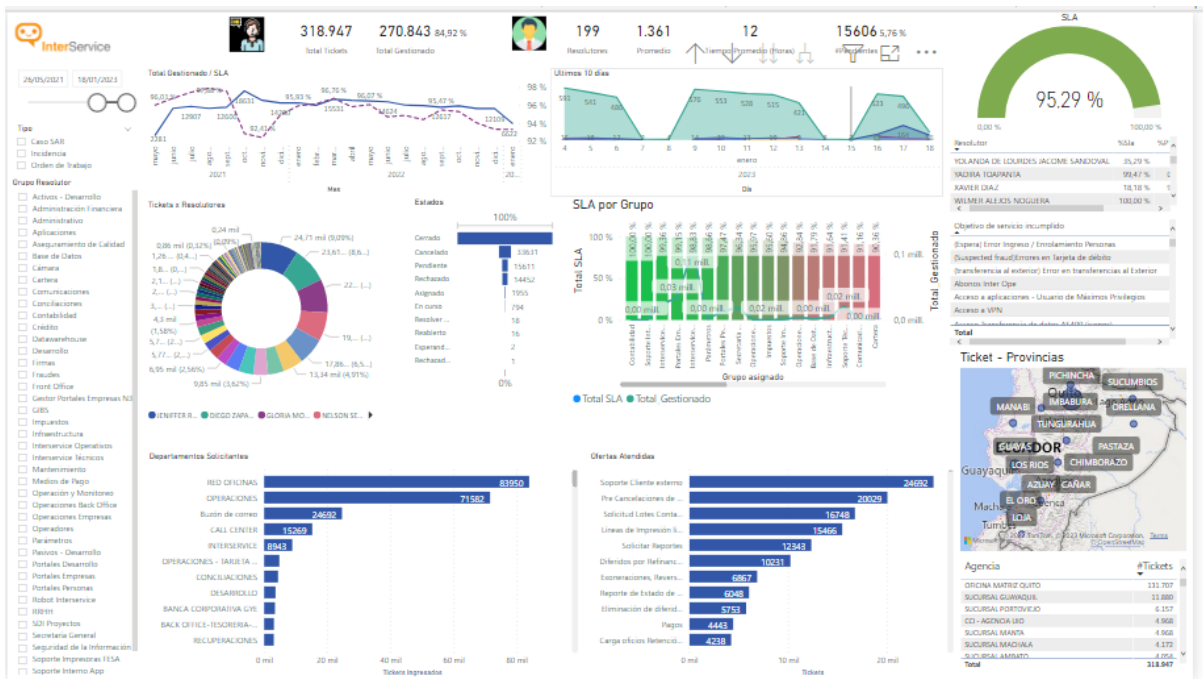
Tablero de Monitoreo



Fuente: Elaborado por el autor.

Figura 31

Tablero de indicadores



Fuente: Elaborado por el autor.

4.8 Validación

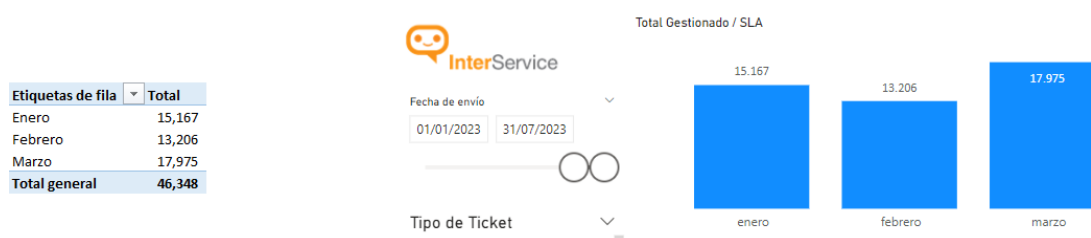
El proceso de validación se la realiza en conjunto con la Gerencia, subgerencia y líderes de equipos del departamento de Interservice.

Para el proceso de validación se extrajo un reporte de la herramienta de Remedy de todas las órdenes de trabajo e incidentes reportados en los meses de enero a marzo del 2023 y la contrastamos con la información del Dashboard de indicadores en conjunto con la Subgerencia del departamento Interservice

La primera validación se realiza obteniendo el total de cada mes en la que la agrupamos la cantidad de registros por mes del reporte extraído de la plataforma de Remedy y comparamos con el grafico de totales del Dashboard de indicadores.

Figura 32

Validación de totales mensuales de Remedy con Tablero de Indicadores



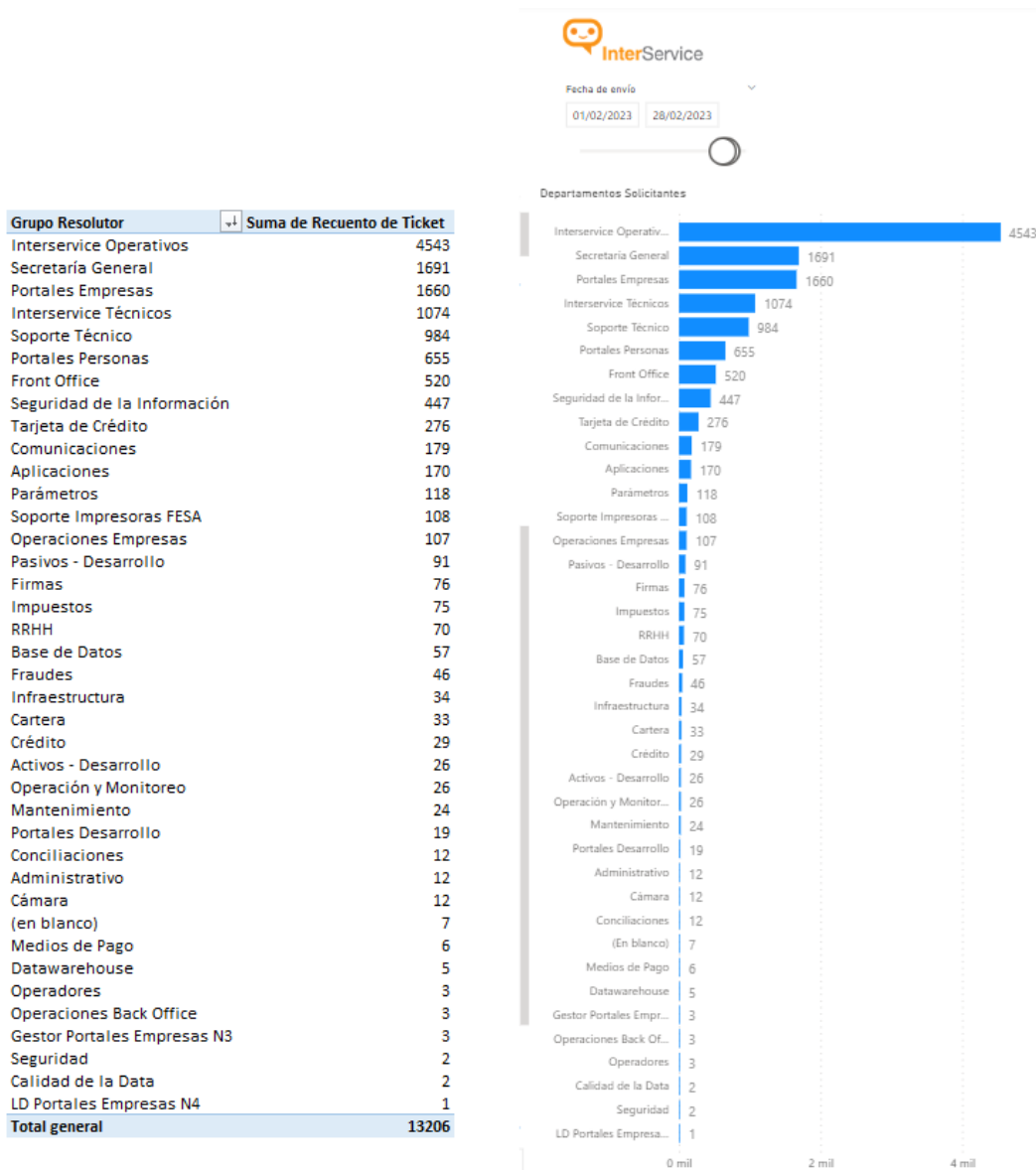
Fuente: Elaborado por el autor.

La figura 32 muestra los valores del conteo mensual del reporte cuadran con los totales mensuales dibujados en el Dashboard de indicadores con lo que se garantiza que el proceso que extrae los registros de las fuentes de información lo hace de forma correcta y completa.

En la segunda validación del reporte obtenido previamente de la herramienta de Remedy filtramos la información de un mes y agrupamos por grupo resolutor y comparamos con el gráfico del Dashboard de indicadores aplicando los mismos filtros de mes que para este caso fue el mes de febrero del 2023 y validamos los totales.

Figura 33

Comparación de un mes entre reporte de Remedy con gráfico del tablero de indicadores agrupados por grupo resolutor



Fuente: Elaborado por el autor.

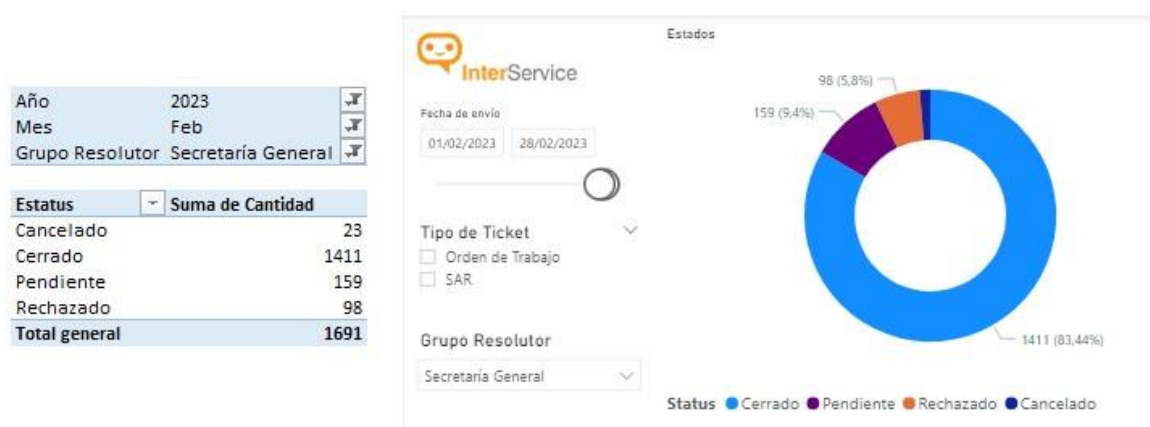
La figura 33 muestra la cantidad de registros agrupados por grupo resolutor coinciden entre el reporte de Remedy y el gráfico de totales por grupo resolutor del tablero de indicadores con lo que se garantiza que los registros extraídos y su agrupación es correcto.

Para finalizar las validaciones de la información extraída de la herramienta de Remedy se agrupa la información del mes de febrero del 2023 y filtrando los registros de un solo grupo

resolutor y agrupado por estados que para esta validación tomamos el grupo de Secretaría General y la comparamos con la información del tablero de indicadores filtrando el mismo mes y grupo resolutor.

Figura 34

Comparativa de cantidad de registros de un mes y un grupo resolutor entre el reporte de Remedy y el tablero de indicadores



Fuente: Elaborado por el autor.

La Figura 34 muestra la comparativa de la cantidad de registros del mes por un grupo resolutor versus el reporte de Remedy con lo que podemos confirmar que al agrupar por estados los requerimientos de un departamento de un mes en específico concuerdan con la información extraída de la herramienta Remedy.

Con todas estas validaciones en la que se compara los reportes extraídos de la herramienta de Remedy y la información que se está presentando en el Dashboard de indicadores se ha corroborado en conjunto con el equipo de validación conformado por la Gerencia, la subgerencia y los líderes de equipo que lo que se muestra en el Dashboard está correcto.

Conclusiones

En las conclusiones de este proyecto de elaboración de un dashboard, es importante destacar los logros significativos y los hallazgos clave que se han obtenido en el proceso de diseño y desarrollo de esta herramienta de visualización de datos. A lo largo de este proyecto, se ha trabajado en la creación de un dashboard que tiene como objetivo proporcionar a los usuarios una plataforma efectiva para monitorear y analizar indicadores clave de rendimiento. Se ha puesto un énfasis particular en la claridad del diseño, la usabilidad y la capacidad de proporcionar información valiosa de manera accesible. En esta sección de conclusiones, se resumirán los aspectos destacados del proyecto, los desafíos superados y se discutirá la importancia de este dashboard en el contexto de la toma de decisiones empresariales. Además, se considerarán posibles mejoras futuras y la continuación del monitoreo y la actualización de este valioso recurso de visualización de datos.

La metodología Hefesto aplicada en este proyecto, se enfoca en un diseño estructurado del Data Warehouse. Esto garantiza una comprensión sencilla de los requerimientos de los usuarios, a fin de aprovechar los datos de manera efectiva y eficiente. También, ayuda a crear una base de datos OLTP mediante el análisis de relaciones en las necesidades de los usuarios, lo que facilita la explotación de esta a través de los tableros dashboard.

El DWH fue creado utilizando el método de modelo estrella, lo que facilitará el mantenimiento en el futuro gracias a la simplicidad de su estructura al momento de realizar modificaciones.

La implementación de los tableros en Power BI para el Área de Interservice permitió obtener indicadores en tiempo real, lo que ayuda a los jefes de cada equipo a mantener un control continuo sobre los resolutores a su cargo en los tiempos de respuesta de la atención en los tickets.

La socialización del dashboard de monitoreo a los resolutores en Interservice ayuda a disminuir la mala práctica de colocar los casos en estado pendiente deteniendo así su SLA de atención para poder cerrar el caso fuera del SLA y que la herramienta le marque como

atendido a tiempo, como podemos observar en la imagen desde su socialización en octubre del 2022 se evidencia un descenso en un 64% en enero del 2023.

Recomendaciones

El desarrollo de un dashboard en Power BI para la presentación de indicadores es una tarea fundamental en el entorno empresarial actual. En un mundo cada vez más impulsado por los datos, contar con un conjunto efectivo de indicadores y presentarlos de manera clara y accesible puede marcar la diferencia en la toma de decisiones informadas y estratégicas.

Una vez concluido el trabajo y basados en los resultados obtenidos se recomienda lo siguiente:

Aumentar las capacidades del hardware del equipo donde se encuentra el tablero de monitoreo debido a que muchas veces se cuelga y su reinicio tarda mucho tiempo en iniciar la herramienta de Power BI.

Con el fin de evitar posibles errores en la visualización de la información en los tableros, se recomienda no realizar actualizaciones masivas en los estados de los tickets, de ser necesario realizar esta actualización masiva se recomienda que también actualicen el campo fecha de modificación para que de esta forma el proceso ETL tome estos registros modificados y los actualice en el DWH.

Que se asigne un responsable que cuenta con las habilidades necesarias para monitorear el proceso de extracción y carga de registros al DWH para tener siempre el sistema activo y con información actualizada.

Establecer un plan de mantenimiento regular para garantizar que el dashboard siga siendo efectivo a medida que evolucionen los datos y las necesidades del departamento Interservice.

Asegurarse de tener acceso a fuentes de datos confiables y actualizadas. Los procesos de limpieza y preparación de los datos son importantes antes de su análisis.

Referencias

- William George, P. P., Güere Salazar, F. V., López Cuadros, D., & Cruz García, A. (2021). Modelo de toma de decisiones implementado con BI para la gerencia de ventas en una comercializadora de alimentos. *Llamkasun*, 2. <https://doi.org/10.47797/llamkasun.v2i4.72>
- Morales Carrillo, J., Cedeño Valarezo, L., Cajape Bravo, J. S., & Ormaza Calderón, J. G. (2022). Metodologías de desarrollo de software y su ámbito de aplicación: Una revisión sistemática. *Revista Ibérica De Sistemas e Tecnologías De Informação*.
- Velásquez, S. M., Vahos Montoya, J. D., Gómez Adasme, M. E., Restrepo Zapata, E. J., Pino, A. A., & Londoño Marín, S. (2019). Una revisión comparativa de la literatura acerca de metodologías tradicionales y modernas de desarrollo de software. *Revista CINTEX*, 24(2). <https://doi.org/10.33131/24222208.334>
- Coronel Valenzuela, M. (2019). Metodología ETL para el procesamiento de datos en repositorios de proyectos. <https://bit.ly/44ikN6y>
- León Guzmán, E. (2019, July). Metodologías. <https://bit.ly/3JGK9TA>
- Galán Amador, M. (2018, October). Guía metodológica para diseños de investigación. <https://bit.ly/3XzHctV>
- Sociedad, U. Y., Díaz Vásquez, R. A., Lenin, J., Espinoza, A., Antonio, M., & Cabrera, C. (2022). Power bi como herramienta de apoyo a la toma de decisiones. *Universidad y Sociedad*, 14(S3).
- Ayala, J., Ortiz, J., Guevara, C., & Maya, E. (2018). Herramientas de Business Intelligence (BI) modernas, basadas en memoria y con lógica asociativa. *Revistapuce*. <https://doi.org/10.26807/revpuce.v0i106.144>
- Microsoft. (2019). Power BI | Herramientas de BI para la visualización de datos interactivos. 2019.
- Porras Blanco, M. (2017, September 29). KPI's ¿Qué son, para qué sirven y por qué y cómo utilizarlos? *Logicalis*. <https://bit.ly/44omgYV>

- Martins, Julia. (2022, August 16). Qué es un KPI, para qué sirve y cómo utilizarlo en tu proyecto. ASSANA. <https://bit.ly/443TVaG>
- García-Jiménez, A. de-J., Aguilar-Morales, N., Hernández-Triano, L., & Lancaster-Díaz, E. (2021). La inteligencia de negocios: herramienta clave para el uso de la información y la toma de decisiones empresariales. *Revista de Investigaciones Universidad Del Quindío*, 33(1). <https://doi.org/10.33975/riuq.vol33n1.514>
- Viteri-Cevallos, C. J., & Murillo-Párraga, D. Y. (2021). Inteligencia de Negocios para las Organizaciones. *Revista Arbitrada Interdisciplinaria Koinonía*, 6(12). <https://doi.org/10.35381/r.k.v6i12.1291>
- Castellnou, R. (2022, January 27). Inteligencia empresarial: Transformación de datos a decisiones . <https://bit.ly/3CSlq9X>
- Llamas, J. (2022). Inteligencia de negocio. <https://bit.ly/3NTxTSs>
- Iseminger, D. (2019). Descripción de un esquema de estrella e importancia para Power BI. <https://bit.ly/3Pzp55i>
- IBM. (2019, June 11). ¿Qué es el modelado de datos? <https://bit.ly/3JAuS6V>
- Piattini Velthuis, Ma., Cabellero Muñoz, I., & Gómez Carretero, A. (2018). Calidad de datos (Primera). Grupo Editorial RA-MA. <https://bit.ly/440RfKN>
- Deborah Henderson. (2022). Guía del conocimiento para la gestión de datos (Technics Publications, Ed.; 2022nd ed., Vol. 2).