



**Diseño, implementación y evaluación MyFEPS de un  
modelo de Business Intelligence para empresas Pymes.**

**Universidad de Belgrano  
Facultad de Tecnología e Información  
Año 2022**

**Carrera:** Licenciatura en Sistemas de Información

**Autor:**

- *Duplaa, Sacha Ignacio*
  - *Matrícula 501-11201*
  - *ID P000147200*
  - [sacha.duplaa@comunidad.ub.edu.ar](mailto:sacha.duplaa@comunidad.ub.edu.ar)

**Tutora:** *Lic. María Belén Piedrabuena*

[belenpiedrabuena@comunidad.ub.edu.ar](mailto:belenpiedrabuena@comunidad.ub.edu.ar)

**Experto técnico:** *Mg. Paula María Angeleri*

[paula.angeleri@comunidad.ub.edu.ar](mailto:paula.angeleri@comunidad.ub.edu.ar)

## **Agradecimientos**

Quiero agradecer a mi tutora Lic. María Belén Piedrabuena por acompañarme durante el desarrollo de mi tesina de grado.

Al profesor Jorge Luis Ceballos quien ha aportado de sus conocimientos en testing y riesgos del software para ayudarme a realizar este trabajo.

Al Mg. Sergio Aguilera, por su ayuda constante y apoyo a lo largo de la carrera.

Al PhD. Alejandro Mitaritonna, por sus consejos y por su aporte crítico que motivó la mejora continua de este trabajo.

A la profesora Paula Angeleri por participar como experta técnica tanto en este trabajo final de carrera como a lo largo de mi formación profesional durante la carrera.

A todos mis profesores quienes a lo largo de estos años me guiaron e instruyeron.

A mis compañeros con los que compartí tiempo dentro y fuera de la universidad, formé recuerdos y anécdotas y que en su día a día me ayudaron a llegar aquí.

Y por último y más que a nadie, a mi familia, que siempre estuvo ahí dando aliento y su apoyo incondicional.

## Índice

Resumen .....	9
Abstract .....	9
1 Introducción.....	10
1.1 Formulación del problema.....	10
1.2 Justificación del trabajo .....	11
1.3 Objetivos.....	12
1.3.1 Objetivo general.....	12
1.3.2 Objetivos específicos.....	12
1.3.3 Objetivos personales.....	13
1.4 Elaboración de hipótesis.....	14
1.5 Metodología .....	15
1.5.1 Justificación de la elección.....	15
1.6 Alcance y limitaciones del trabajo.....	16
1.6.1 Alcance.....	16
1.6.2 Limitaciones.....	17
2 Marco teórico.....	16
2.1 Business Intelligence.....	17
2.1.1 ¿Qué es el Business Intelligence?.....	17
2.1.2 ¿Cómo funciona?.....	17
2.2 Selección de la herramienta de BI.....	17
2.2.1 Arquitectura de power BI.....	18
2.3. Data Warehouse.....	19
2.3.1 Características de un Data Warehouse.....	20
2.4 Data Mart.....	21
2.4.1 ¿Qué es un datamart?.....	21
2.4.2 Ejemplo de un datamart.....	21

2.5 Modelado dimensional.....	21
2.5.1 Elaboración del Data Mart Ventas.....	21
2.6 ETL.....	22
2.6.1 Integración de datos.....	23
2.6.2 Extracción.....	23
2.6.3 Transformación.....	23
2.6.4 Carga.....	23
2.7 Cubo OLAP.....	23
2.8 Indicadores clave de rendimiento (KPI).....	24
2.9 Riesgos .....	24
2.9.1 Definición de riesgo .....	24
3 Desarrollo.....	25
3.1 Planificación del proyecto.....	25
3.2 Análisis de requerimientos.....	26
3.2.1 Especificación de requisitos funcionales y no funcionales.....	26
3.3 Determinar los objetivos del negocio.....	27
3.4 Análisis de riesgos.....	27
3.4.1 Escala de riesgos .....	27
3.4.2 Ponderación de probabilidades de ocurrencias .....	28
3.4.3 Matriz de riesgos del proyecto .....	28
3.4.4 Matriz de riesgos del producto .....	29
3.5 Especificar la solución .....	30
3.6 Identificar los orígenes de datos .....	30
3.6.1 Análisis de la base de datos .....	31
3.6.2 Estrategia para la unificación de datos .....	32
3.7 Desarrollo del Data Warehouse .....	33
3.7.1 Descripción del Data Warehouse .....	33

3.7.2	Justificación de la elección del modelo estrella .....	33
3.7.3	Objetivos y alcance .....	33
3.7.4	Objetivo principal .....	33
3.7.5	Objetivos generales.....	33
3.7.6	Arquitectura del Data Warehouse.....	34
3.7.6.1	Diseño físico del Data Warehouse.....	35
3.7.6.2	Estructura y tipo de datos de las tablas.....	35
3.8	Proceso ETL.....	38
3.8.2	Elección de la herramienta de ETL.....	38
3.8.3	Librerías utilizadas para el desarrollo del proceso ETL .....	38
3.8.4	Explicación del proceso ETL .....	39
3.8.5	Detalle programación de funciones del ETL .....	47
3.9	Creación del cubo OLAP .....	48
3.9.1	Conexión con el Data Warehouse .....	48
3.9.2	Vistas de datos .....	49
3.9.3	Selección de dimensiones .....	50
3.9.4	Generación del cubo .....	50
3.9.5	Procesado y compilación del cubo .....	51
3.10	Desarrollo del informe en PowerBI .....	53
3.10.1	Orígenes de datos .....	53
3.10.1.1	Capa modelo de datos .....	53
3.10.1.2	Capa modelo .....	54
3.10.1.3	Capa de datos .....	55
3.10.1.4	Capa informe .....	56
3.10.2	Estructura general de las vistas .....	56
3.10.3	Vista ventas por región .....	57
3.10.3.1	Gráfico Variación de ventas según cantidad y facturación por sucursal.....	58

3.10.3.2	Gráfico Ventas anuales.....	58
3.10.3.3	Gráfico Mapa .....	59
3.10.4	Vista productos más vendidos.....	59
3.10.4.1	Gráfico TOP 5 Productos con mayor volumen de ganancias.....	59
3.10.4.2	Gráfico Variación de ganancias anuales por marca .....	60
3.10.4.3	Gráfico KPI Cantidad de productos vendidos .....	60
3.10.5	Vista Clientes.....	60
3.10.5.1	Gráfico KPI Porcentaje de compradores por género.....	60
3.10.5.2	Gráfico Top 5 clientes por facturación.....	61
3.10.5.3	Gráfico variación de facturación anual por provincia .....	61
3.10.6	Tarjetas utilizadas .....	61
3.10.7	Filtros utilizados .....	62
3.10.8	Medidas utilizadas .....	62
3.10.9	Explicación de las medidas .....	63
4.0	Proceso de evaluación ágil Dashboard de ventas - MyFeps .....	64
4.1	Establecer el propósito de la evaluación .....	64
4.2	Identificar el producto a evaluar .....	64
4.3	Identificar requerimientos de calidad .....	65
4.3.1	Definición de los stakeholders.....	65
4.3.2	Definición de las características, subcaracterísticas,métricas y atributos.....	67
4.4	Definir recursos y equipo para la evaluación.....	71
4.5	Establecer ponderación.....	72
4.6	Preparar recursos e infraestructura.....	73
4.7	Elaborar casos de prueba.....	73
4.7.1	Plan de pruebas.....	74
4.7.2	Especificación de la evaluación.....	74
4.8	Obtener y adecuar recursos para la evaluación.....	77

4.9 Realizar las pruebas.....	77
4.10 Medir los atributos requeridos por las métricas.....	83
4.10.1 Volcado de valores en la Planilla de cálculo de G Software.....	86
4.11 Resultados de la evaluación.....	87
5.0 Conclusiones.....	87
5.1 Conclusiones personales.....	87
5.2 Futuras líneas de investigación.....	88
Referencias bibliográficas: .....	89
Glosario .....	91
Anexos.....	92

## Índice de Figuras

Figura 1: Administración del proyecto de BI.

Figura 2: Microsoft PowerBI Gartner Quadrant 2022.

Figura 3: Arquitectura de PowerBI.

Figura 4: Data Warehouse

Figura 5: Modelo estrella del Data Mart ventas.

Figura 6: Proceso ETL.

Figura 7: Estructura de un cubo OLAP.

Figura 8: Gestión, evaluación y tratamiento del riesgo

Figura 9: Diseño físico del Data Warehouse

Figura 10: Base de datos

Figura 11: ETL En Python 3

Figura 12: Conexión al DW

Figura 13: Driver de conexión

Figura 14: Selección de vistas de datos

Figura 15: Selección de atributos

Figura 16: Creación del cubo

Figura 17: Procesado del cubo

Figura 18: Comprobación de la cantidad de registros del cubo y data mart

Figura 19: Capa Obtener conjunto de datos PowerBI

Figura 20: Capa Modelo PowerBI

Figura 21: Capa de Datos PowerBI

Figura 22: Capa de Informe PowerBI.

Figura 23: Mockup estructura de tableros

Figura 24: Vista ventas por región

Figura 25: Leyenda del gráfico variación de ventas según cantidad y facturación por sucursal

Figura 26: Vista Productos más vendidos

Figura 27: Tablero Clientes.

Figura 28: Curva de usabilidad

Figura 29: Resultados del cuestionario satisfacción de los usuarios.

Figura 30: Planilla de cálculo de G Software

## **Índice de Tablas**

Tabla 1: Planificación del proyecto

Tabla 2: Análisis de riesgos del proyecto

Tabla 3: Análisis de riesgos del producto

Tabla 4: Dimensión clientes

Tabla 5: Dimensión productos

Tabla 6: Dimensión Sucursales

Tabla 7: Dimensión Tiempo

Tabla 8: Fact Ventas

Tabla 9: Tabla de funciones - ETL

Tabla 10: Tarjetas utilizadas en PowerBI.

Tabla 11: Filtros utilizados en PowerBI.

Tabla 12: Medidas utilizadas en PowerBI.

Tabla 13: Modelo de evaluación ágil MyFEPS.

Tabla 14: Grupo Evaluador.

Tabla 15: Escala de ponderación MyFEPS Ágil.

Tabla 16: Resultado Cuestionario pesos MyFEPS Ágil.

Tabla 17: Grupo de usuarios finales.

Tabla 18: Especificación de los casos de prueba.

Tabla 19: MyFEPS Ágil. Correctitud de datos - 04.1.1.A.

Tabla 20: MyFEPS Ágil. Correctitud de procesos - 04.2.1.A.

Tabla 21: MyFEPS Ágil. Usabilidad objetiva. En el aprendizaje 17.1.4

Tabla 22: MyFEPS Ágil. Ayuda para la Prevención de errores en el ingreso manual de información 17.3.3.U

Tabla 23: MyFEPS Ágil. Velocidad en el Acceso a las funciones. 17.4.4.U

Tabla 24: MyFEPS Ágil. Evaluación promedio de usuarios en lo relativo al grado de conocimiento de las operaciones del sistema 17.6.1.U

Tabla 25: MyFEPS Ágil. Características.

Tabla 26: MyFEPS Ágil. Subcaracterísticas.

Tabla 27: MyFEPS Ágil. Atributos.

## **Resumen.**

En este trabajo de investigación, se desarrollará e implementará un modelo de Business Intelligence diseñado para una empresa Pyme con un rango de empleados que va desde 100 personas como mínimo hasta 200 como máximo.

El Business Intelligence (BI) [1] es un conjunto de técnicas de manipulación de datos utilizadas en empresas con un principal objetivo, el cual consiste en transformar información en conocimiento para la toma de decisiones basadas en datos.

En este trabajo de tesina, se presentarán 2 (dos) principales problemas de la compañía, siendo (1) la información descentralizada y (2) la infraestructura TI.

La situación actual que presenta la empresa es que no cuenta con una infraestructura tecnológica adecuada para sustentar el negocio. La empresa trabaja con mucho volumen de información que es crucial para su correcto funcionamiento. Dicha información contiene datos como la facturación histórica desde que la empresa puso inicio a sus actividades. Toda esta información no se encuentra consolidada; es decir, que los datos no están representados de manera unificada y relacionada, por lo que existe un problema de inconsistencia de información y esto puede ser contraproducente para la compañía, dado que si un dato se pierde, puede representar una pérdida importante de sumas de dinero, perjudicando a la compañía y a sus empleados.

## **Abstract**

In this research work, a Business Intelligence model designed for a SME company with a range of employees ranging from 100 people at least to 200 at most will be developed and implemented.

Business Intelligence (BI) is a set of data manipulation techniques used in companies with a main objective, which is to transform information into knowledge for decision-making based on data.

In this thesis work, 2 (two) main problems of the company will be presented, being (1) the decentralized information and (2) the IT infrastructure.

The current situation presented by the company is that it does not have an adequate technological infrastructure to support the business. The company works with a large volume of information that is crucial for its correct operation. This information contains data such as

historical billing since the company started its activities. All this information is not consolidated; that is, the data is not represented in a unified and related way, so there is a problem of information inconsistency and this can be counterproductive for the company, given that if data is lost, it can represent a significant loss of sums of money, harming the company and its employees.

## **1 - Introducción**

Hoy en día la transformación digital está en constante cambio y crecimiento. Es por ello que cada día salen al mercado nuevas tecnologías y herramientas que nos facilitan la vida, tanto a los desarrolladores como a los usuarios finales.

En la actualidad, la mayoría de grandes empresas cuentan con un modelo de Business Intelligence (BI) que les permite realizar análisis de datos. Al implementar estas técnicas de manipulación de datos en la organización, se ayuda tanto a los gerentes de la empresa como a los diversos miembros de los equipos que analizan métricas utilizando los datos generados por el modelo BI.

En conclusión, el BI, sirve para recopilar, organizar y analizar datos, transformándolos en importantes fuentes de información y conocimiento para la posterior toma de decisiones.

### **1.1 Formulación del problema**

La empresa Marketbris es una empresa Pyme que está ubicada en la Ciudad Autónoma de Buenos Aires Argentina y se encuentra en expansión por toda la región. La actividad principal de la empresa, se basa en la venta de productos. El mayor problema de la empresa es que existe una gran cantidad de información que no está relacionada entre sí. Marketbris informó que al unificar los datos hay pérdida de algunos registros. Esto es debido a que los procesos

utilizados para cruzar los datos, se hacen de manera manual y se realizan, en mayor medida, en planillas de cálculo en Excel (y en ocasiones en papel).

En cuanto a la infraestructura de la empresa, se cuenta a nivel de software, con una aplicación web programada en php denominada "marketbris sales". Esta aplicación, corre sobre un en un servidor (Windows server 2008 R2) que está administrado por la empresa. La función principal de esta aplicación es permitir al usuario descargar un informe de ventas semanal.

Este sistema se ejecuta localmente y no cuenta con ninguna medida de seguridad para acceder y proteger la información, por lo que cualquier usuario puede acceder tanto al sistema como a la propia base de datos.

El sistema cuenta con una base de datos que presenta una serie de errores importantes en su funcionamiento. Además, no cuenta con un correcto modelado de datos, lo que imposibilita relacionar tablas directamente debido a su estructura. Cabe agregar que la base de datos en sí no tiene ningún tipo de seguridad, por lo que es fácil para un atacante clonar la base de datos y obtener acceso sin restricciones. La suma de todos estos factores, conllevan a que la dirección no disponga de la información clara y necesaria para tomar decisiones y sacar el máximo partido a su modelo de negocio.

## **1.2 Justificación del trabajo**

Hoy en día, uno de los principales problemas en las empresas consiste en la administración y el manejo de la información. Implementar un modelo de Business Intelligence (BI) es la mejor solución a este problema, dado que la empresa actualmente cuenta con una infraestructura que no es adecuada para su negocio, procesos y tareas manuales que podrían ser automatizadas vía software. Además de todo lo mencionado anteriormente, tampoco existe un tablero que contenga reportes donde se puedan visualizar estadísticas de los datos que administra la empresa para poder explotarlos y transformarlos en información útil y de valor.

Todos estos eventos, conllevan a que la empresa no responda de manera óptima ocasionando una serie de problemas que perjudican el funcionamiento de la misma. Por lo tanto, se decidió Implementar un modelo de Business Intelligence (BI) con el objetivo de solucionar esta problemática.

## **1.3 Objetivos**

En este apartado, se detallarán los objetivos generales y específicos que se van a abordar durante el transcurso de este trabajo final de carrera.

### **1.3.1 Objetivo general.**

El objetivo de este trabajo consiste en desarrollar e implementar un modelo de Business intelligence dirigido a empresas pymes, cuya componente principal es facilitar la toma de decisiones a través de los datos pertenecientes a la compañía.

El desarrollo del modelo se llevará a cabo en varias etapas: hacer un análisis de relevamiento del negocio para determinar los objetivos generales y específicos del mismo para luego dar origen a un modelo dimensional que va a ser diseñado para el negocio con el fin de transformar la información en conocimiento. Posteriormente, se hará la integración de las diversas fuentes de datos para poder hacer los cruces de tablas con el fin de darle una trazabilidad al dato, los

procesos ETL, la creación del cubo y, finalmente, un tablero compuesto por 3 (tres) vistas que presentan estadísticas de los datos.

### **1.3.2. Objetivos específicos.**

- Análisis de requerimientos.
- Definir los requisitos funcionales (RF) y los requisitos no funcionales (RNF).
- Determinar los objetivos del negocio.
- Análisis de riesgos.
- Identificar las fuentes de datos.
- Diseñar la arquitectura del Data Warehouse.
- Elaboración del Data Mart Ventas.
- Integración de datos.
- Proceso ETL.
- Diseñar las métricas y KPI'S.
- Diseño del cubo OLAP.
- Diseño del tablero para la toma de decisiones.
- Creación de las vistas para el tablero.
- Evaluación del tablero usando el Framework MyFEPS

### **1.3.3 Objetivos personales.**

Mi objetivo personal es poder aplicar los conocimientos adquiridos a lo largo de mi carrera de grado. Más en concreto, de las siguientes materias:

Base de datos 1.

Base de datos 2.

Programación 1.

Programación 2.

Programación 3.

Ingeniería de Software IV. Testing y Calidad.

Habilitación profesional A. Proyecto de Sistemas.

Habilitación profesional B. Proyecto de Sistemas.

Trabajo final de carrera.

Habiendo mencionado todo lo anterior, como objetivo personal propongo lo siguiente:

- Utilizar un lenguaje de programación para el desarrollo del ETL.
- Utilizar diversos orígenes de datos para lograr un mayor grado de realismo en el proyecto.
- Utilizar la herramienta PowerBI para desarrollar la solución.

## **1.4 Elaboración de hipótesis.**

La hipótesis a verificar es que es posible asegurar que se cumplan la calidad, integridad y disponibilidad de los datos, para mejorar los procesos de negocio de una empresa Pyme haciendo uso de un modelo de Business Intelligence a través de los distintos KPI (Key Performance Indicator) con el fin de otorgar valores medibles (cualitativos y cuantitativos) a la gerencia para la toma de decisiones.

(Nota: los datos obtenidos para realizar este trabajo final de carrera, son públicos y no se tomaron datos de la empresa).

Además, se considera la evaluación de las funcionalidades, la usabilidad y el grado de satisfacción del usuario del tablero, a través del modelo MyFEPS [2][2.1] en su versión MyFEPS Ágil [2][2.1][2.2].

## 1.5 Metodología.

Para realizar este trabajo final de carrera, se utilizará la metodología Ralph Kimball [3] para el desarrollo de este proyecto de Business Intelligence y, posteriormente, se utilizará MyFEPS para medir la usabilidad y grado de satisfacción de los usuarios que hayan participado en la encuesta de software del tablero realizado en este trabajo final de carrera.

### 1.5.1 Justificación de la elección.

Como se mencionó anteriormente, para abordar este proyecto, se utilizarán 2 (dos) metodologías: Ralph Kimball & MyFEPS, esté último, en su versión denominada: MyFEPS Ágil. Se utilizará la metodología Ralph Kimball para el diseño y la implementación del Data Warehouse. Esta metodología es adecuada para el diseño y la implementación de data warehouses pequeños y medianos.

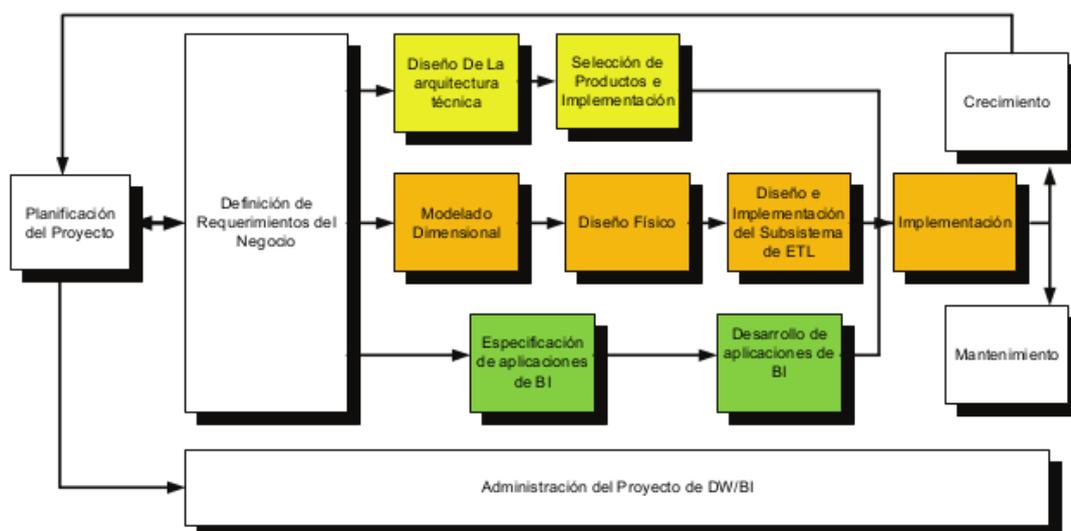


Figura 1: Administración del proyecto de BI.

Fuente: <https://docplayer.es/689468-Enfoques-de-desarrollo-dw-kimball-inmon.html>

Esta metodología, propone una visión de negocio combinada con una visión tecnológica. Esta metodología está basada en 4 (cuatro) principios básicos:

- **Centrarse en el negocio:** Es necesario hacer un análisis de requerimientos para obtener un entendimiento del negocio para, posteriormente, implementar una solución de BI.
- **Construir una infraestructura de información adecuada:** Se debe diseñar una base de datos con las características óptimas para generar los datamarts que nos permitirán explotar la información.
- **Realizar entregas en incrementos significativos:** Esta metodología implica que se deben hacer entregas en plazos acordados con la organización, similar a otras metodologías ágiles para desarrollar software.
- **Ofrecer la solución completa:** Este principio nos indica que se debe entregar un diseño del Data Warehouse funcional y que cumpla con las expectativas de la organización. Además, se deberán realizar las entregas con las aplicaciones y/o herramientas con las que se realizaron los informes, planes de capacitación y soporte, etc.

Concluida la explicación de la metodología de Ralph Kimball, se utilizará la metodología de MyFEPS Ágil para medir y evaluar atributos tales como la correctitud de datos y procesos, usabilidad del tablero y el grado de satisfacción de los usuarios.

“MyFEPS es un Framework de Calidad de Software (Modelo, Proceso y Herramientas) que brinda apoyo al proceso de evaluación de software en su totalidad. Está basado en normas internacionales (ISO/IEC 9216, ISO/IEC 14598, e ISO/IEC 25000 SQUARE.) y tiene un enfoque distinto al modelo de calidad ISO 9000” [2][2.1][2.2].

## **1.6 Alcance y limitación del trabajo.**

En este apartado se detalla el nivel de alcance y limitaciones que impidan realizar el proyecto.

### **1.6.1 Alcance.**

La empresa Marketbris está interesada en actualizarse a nivel tecnológico con el fin de poder explotar todos sus recursos de información.

En primera instancia, se buscará relacionar toda la información de las ventas, clientes y sucursales que tiene la empresa. En segundo lugar, se buscará integrar toda la información digital en una base de datos unificada para luego, explotar los datos de una manera estratégica utilizando una herramienta de Business Intelligence, con el fin de otorgar información consolidada para mejorar la toma de decisiones.

### **1.6.2 Limitaciones**

Además del alcance de la solución BI que se detalla en el análisis y diseño de la solución, se ha definido utilizar la metodología Ralph Kimball [3] para su desarrollo y, posteriormente, se utilizará MyFEPS para su evaluación. No es propósito de este trabajo determinar la metodología óptima, siendo Kimball uno de los gurúes del Business Intelligence. Por otra parte se usa MyFEPS por ser una herramienta gratuita y validada. Respecto de la herramienta para implementar la solución BI se usará Power BI Desktop, por ser de mi interés, ser gratuita, y conocida en el mercado.

Asimismo, se limitará la evaluación de la performance de este modelo BI dado que la empresa cuenta con un servidor local, el cual cuenta con un hardware suficiente para este proyecto, la única limitación que se presenta es la imposible escalabilidad del servidor y, por motivos de presupuestos, este no será actualizado o reemplazado por otro. Además, la compañía, de momento, tampoco está dispuesta a contratar un servicio en cloud que sería una solución posiblemente menos costosa, más segura y escalable al mismo tiempo (como futura líneas de investigación y oportunidad de mejora, se planteará una solución cloud utilizando Microsoft Azure).

## **2. Marco teórico.**

En este apartado, se procede a detallar todos y cada uno de los conceptos utilizados a lo largo de este trabajo final de carrera con la finalidad de introducir al lector en la materia.

## **2.1 Business Intelligence.**

### **2.1.1 ¿Qué es el Business Intelligence?**

Históricamente, la Inteligencia de negocios (desde ahora en más, BI), es una rama dentro de la informática que se encarga de manipular la información de una compañía con el fin de generar conocimiento de importancia para el negocio.

Se busca analizar datos, utilizando técnicas de manipulación, extracción de datos y visualización de los mismos.

Estos sistemas de BI no cuentan con bases de datos transaccionales OLTP, sino que cuentan con bases de datos multidimensional OLAP.

Hoy en día, en efectos prácticos, la disciplina del BI "moderno", requiere contar con una visión integral de todos los datos del negocio con el fin de recolectar la información y hacer un uso de la misma para poder generar un impacto en el negocio.

### **2.1.2 ¿Cómo funciona?**

Los negocios y organizaciones tienen metas, objetivos y preguntas pero con la diferencia de que estas, no saben la respuesta.

Para encontrar esa respuesta, se necesita hacer una trazabilidad a cada meta, recopilando los datos necesarios para luego poder analizarlos y determinar qué acciones deberán llevarse a cabo con el fin de alcanzar esas metas. El BI, además de responder a las preguntas, permite formular nuevas preguntas o reformular las preguntas anteriores para poder identificar nuevos insights<sup>1</sup>.

## **2.2 Selección de la herramienta de BI**

Para este trabajo de investigación, se abordará la herramienta Power BI Desktop, desarrollada por Microsoft. Esta herramienta es utilizada para hacer análisis de datos con el fin de explotar esa información lo mayor posible para poder brindar un informe descriptivo a la o las personas y/o entidades que quieran visualizar esos datos. La principal función de Power BI consiste en permitir al usuario visualizar los datos en tableros, asimismo, permite la creación de tableros y vistas, entre otras cosas.

Además, Power BI, es de uso gratuito y actualmente está catalogada como herramienta de BI líder año 2022, situándose por encima de otras tales como QlikView y Tableau.

---

<sup>1</sup> Un insight, en el análisis de datos, hace referencia a aquellos datos que aportan información valiosa para la organización.



Figura 2: Microsoft Power BI Gartner Quadrant 2022

Fuente: <http://blog.bismart.com/microsoft-power-bi-lider-del-cuadrante-magico-de-gartner-2022>

### 2.2.1 Arquitectura de Power BI

La arquitectura de Power BI permite conectar con distintas fuentes de datos para la obtención de datos con el fin de explotarlos. Las fuentes de datos pueden encontrarse en los siguientes formatos; archivos de hoja de cálculo (.xlsx), texto plano (.txt), separados por coma (.csv), JSON. Además, esta herramienta nos permite conectar con bases de datos relacionales y multidimensionales, ya sea on premise o en cloud.

Power BI está disponible en las siguientes plataformas:

- Desktop: para el sistema operativo windows.
- Mobile: para tablets y smartphones con sistemas Android e iOS
- Cloud: compatible con google chrome, mozilla firefox y opera.

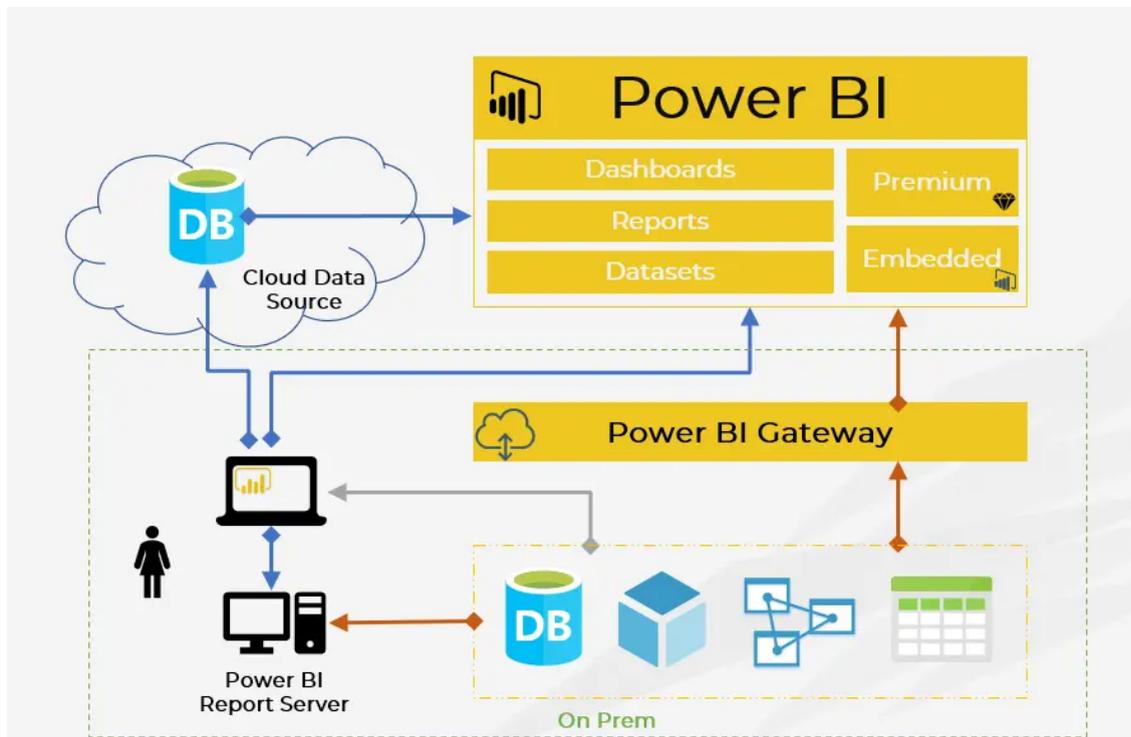


Figura 3: Arquitectura de Power BI

Fuente: <https://qmetrix.com.au/power-bi-architecture-and-security/>

## 2.3 Data Warehouse

Un data warehouse [3] o almacén de datos, es un repositorio centralizado de datos integrados por una o más fuentes de datos. Los data warehouses almacenan datos consolidados que luego son utilizados para hacer sus respectivos análisis.

“Un almacén de datos (data warehouse, DW) según Inmon (Inmon 02, Imhoff & Galemno 03), es una colección de datos orientada a un determinado ámbito (empresa, organización, etc.), integrado, no volátil y variable en el tiempo, que ayuda a la toma de decisiones en la entidad en la que se utiliza. Se trata, sobre todo, de un historial completo de la organización, más allá de la información transaccional y operacional, almacenado en una base de datos diseñada para favorecer el análisis y la divulgación eficiente de datos (especialmente con herramientas OLAP, de procesamiento analítico en línea). Por otra parte, Kimball la define como “una copia de los datos transaccionales estructurados específicamente para consultas y análisis” [3][3.1].

### 2.3.1 Características de un Data Warehouse

A continuación se presentaran las caracterisitcas que compone un Data Warehouse, Según Bill Inmon<sup>2</sup>:

- Orientado a temas: Los datos en la base de datos están organizados de manera tal que todos los elementos de datos relativos al mismo evento u objeto del mundo real queden unidos entre sí.
- Variable en el tiempo: Los cambios producidos en los datos a lo largo del tiempo quedan registrados para que los informes que se puedan generar reflejen esas variaciones.
- No volátil: La información no se modifica ni se elimina, una vez almacenado un dato, este se convierte en información de *sólo lectura*, y se mantiene para futuras consultas.
- Integrado: La base de datos contiene los datos de todos los sistemas operacionales de la organización, y dichos datos deben ser consistentes.

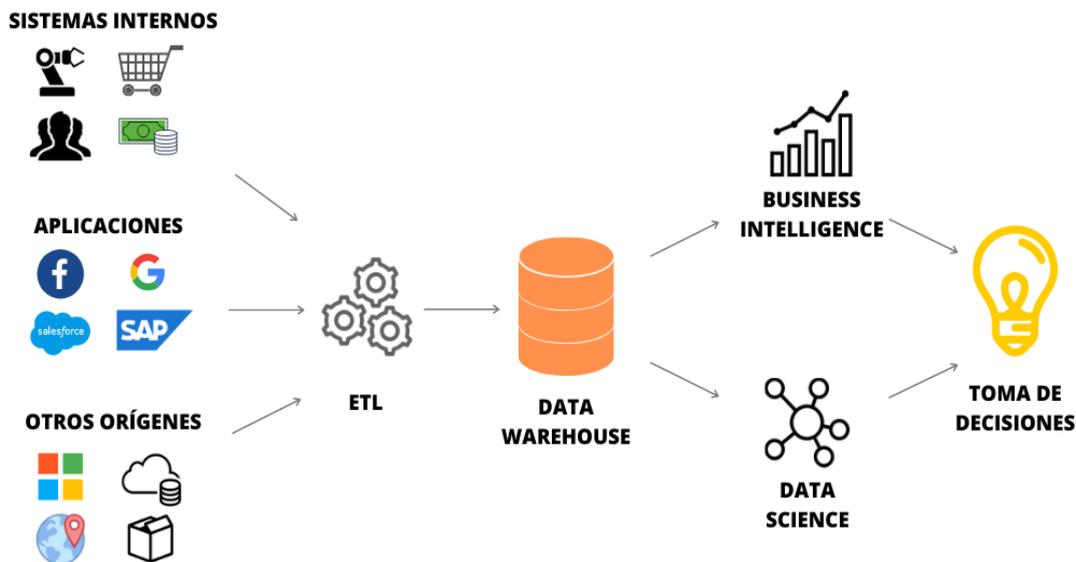


Figura 4: Data Warehouse

Fuente: <https://www.mistralbs.com/expertos-data-warehouse/>

<sup>2</sup> Bill Inmon (1945) es considerado como el padre de los data warehouses.

## **2.4 Data mart**

### **2.4.1 ¿Qué es un Data Mart?**

Un data mart es un módulo específico de un datawarehouse focalizado en un tema o un área de negocio dentro de una organización. Estos, se pueden entender como subconjuntos de datos de un DW con la finalidad de ayudar a que un área específica dentro del negocio pueda tomar mejores decisiones.

### **2.4.2 Ejemplo de un Data Mart**

Supongamos que hay una compañía que se dedica a la compra y venta de máquinas industriales. Dentro del datawarehouse, podemos tener un datamart llamado Finanzas, en donde se va a almacenar toda la información relacionada a las finanzas, de esta manera, se logra compactar y unificar el volumen de datos de una subárea específica de la compañía.

## **2.5 Modelado dimensional**

Un modelo dimensional, es un diseño lógico que posee una estructura que sirve para almacenar datos en un data warehouse.

El modelado dimensional, no implica necesariamente una base de datos relacional, ya que también puede estar en una base de datos multidimensional.

Su diseño puede ser de 3 (tres) tipos diferentes: Estrella, Copo de nieve o Híbrido, siendo este último una composición de la estructura estrella y copo de nieve.

### **2.5.1 Elaboración del Data Mart Ventas**

Para este proyecto se utilizará el Modelado de estrella, dado que este, a diferencia del copo de nieve, consume menos espacio en disco y es más performante en términos de tiempo de procesamiento y ejecución.

Este modelo de tipo estrella, está compuesto por una tabla de hechos (facts) y cinco tablas de dimensiones (dims).

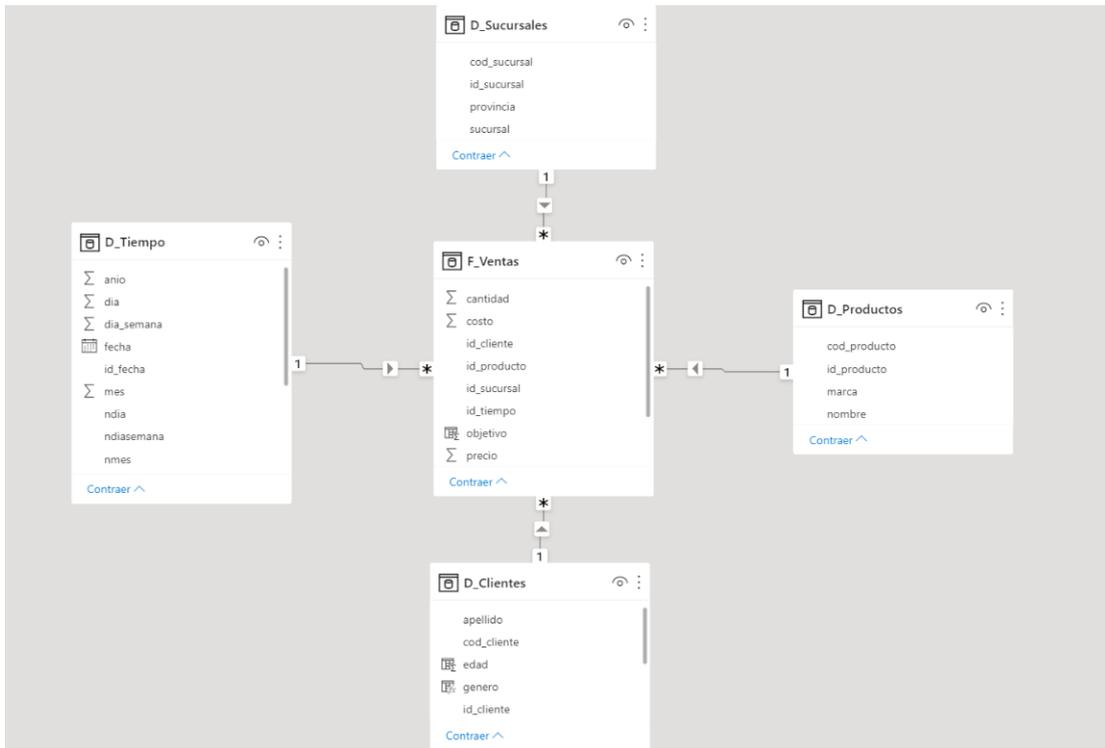


Figura 5: Modelo estrella del Data Mart ventas

Fuente: imagen extraída de mi propio desarrollo.

## 2.6 ETL

Extraer, transformar y cargar (ETL) es el proceso que permite a las organizaciones mover datos desde múltiples fuentes de datos con la finalidad de limpiarlos, procesarlos y formatearlos para luego, cargarlos en un repositorio de almacenamiento; sea una base de datos, un data mart, un archivo excel/csv/txt, etc.

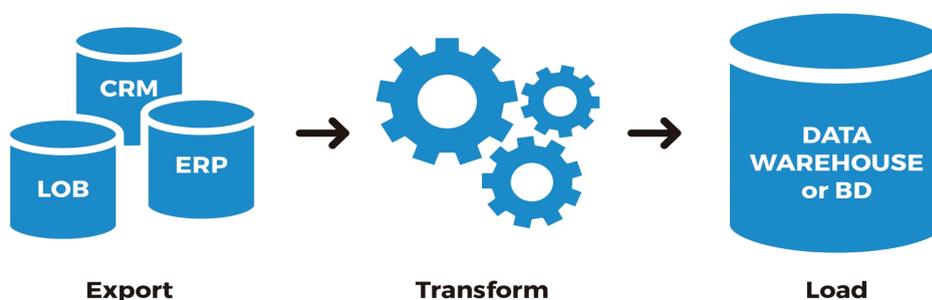


Figura 6: Proceso ETL

Fuente: [https://cdn.goconqr.com/en/p/32411029?canonical=true&frame=true&no\\_cache=true](https://cdn.goconqr.com/en/p/32411029?canonical=true&frame=true&no_cache=true)

### **2.6.1 Integración de datos**

La integración de datos es un proceso en el cual consiste unificar varias fuentes de información en un solo conjunto. La primera premisa que hay que considerar en este aspecto es identificar las diversas fuentes de datos.

En el proceso de integración, los datos suelen provenir de diversas fuentes de datos y en distintos formatos. Por ejemplo, podemos extraer información desde archivos de tipo Excel, JSON, Txt, CSV, etc para luego consolidar dicha información en un solo archivo o fuente de datos.

### **2.6.2 Extracción.**

La fase de ingesta o extracción de los datos, consiste en extraer los datos de una fuente de origen. Esta fuente puede ser: un sistema ERP, un archivo de texto plano, una base de datos, archivos enriquecidos tales como pdf, etc.

### **2.6.3 Transformación.**

Durante esta fase, se procede a realizar transformaciones a los datos para su posterior tratamiento. Ya sea aplicando filtros, convirtiéndolos a un tipo de dato que sea soportado por el sistema destino que es donde se va a realizar la carga de estos.

### **2.6.4 Carga de datos**

Esta fase es la final, los datos que ya fueron previamente tratados en la fase de la transformación, se cargan en un repositorio de datos, dicho repositorio generalmente suele ser un datawarehouse.

## **2.7 Cubo OLAP**

Un cubo de datos OLAP (Online Analytical Processing) es una base de datos multidimensional en que se almacena, de manera física, una medida agregada por una serie de dimensiones de un conjunto de datos en un vector multidimensional. [4]

El cubo OLAP permite consumir, en tiempo real, información de la base de datos multidimensional. Esta base de datos fue desarrollada por Microsoft y es denominada como "Analysis services" [4][4.1].

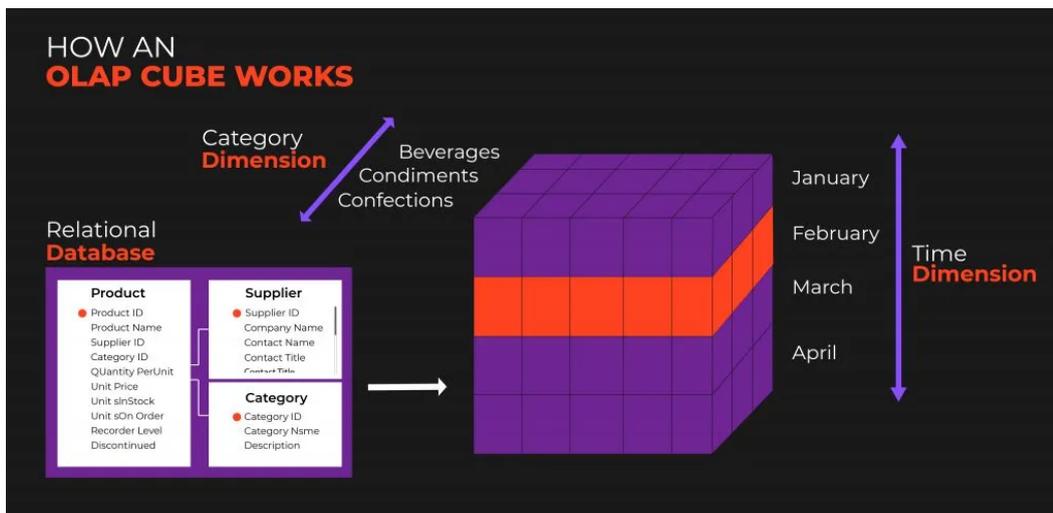


Figura 7: Estructura de un cubo OLAP

Fuente:

<https://i0.wp.com/smartboost.com/wp-content/uploads/2020/06/OLAP-Blog-image-01-1024x497.jpg>

## 2.8 Indicadores clave de rendimiento (KPI)

Un KPI (indicador clave de rendimiento/performance) es una medida del nivel de rendimiento de un proceso. El valor del KPI es un valor numérico calculable y está relacionado con un objetivo fijado previamente y normalmente se expresa en valores porcentuales.

Los KPI tienen como objetivo principal medir valores. Dichos valores medibles pueden ser tales como la medición de un servicio, una actividad, un diagnóstico, etc.

## 2.9 Riesgos.

En este trabajo final de carrera, se identificarán y evaluarán los posibles riesgos que puedan generar un impacto a lo largo del proyecto.

### 2.9.1 Definición de riesgo:

Según la definición de la norma ISO 31000, un riesgo es el “efecto de la incertidumbre sobre el logro de los objetivos” [5].

A continuación se conceptualiza la relación de gestión, evaluación y tratamiento de riesgos

## Introducción al Análisis y Evaluación de Riesgos

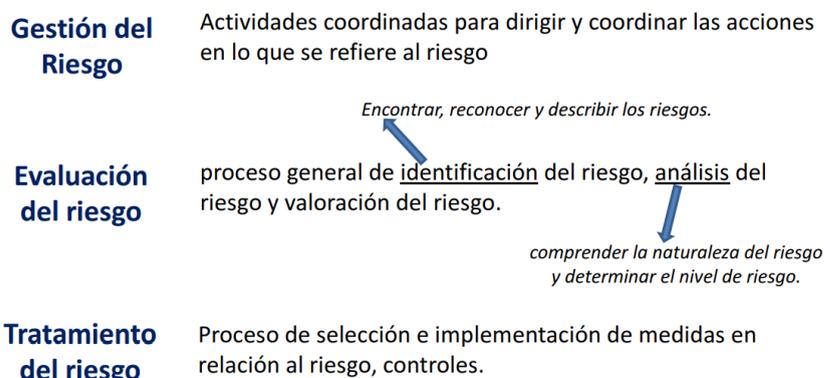


Figura 8: Gestión, evaluación y tratamiento del riesgo

Fuente: Ingeniería de SW IV - Testing y Calidad - Introducción al Análisis de riesgos pág 49.

## 3. Desarrollo

### 3.1 Planificación del proyecto

Durante este proyecto, se detallan las fases y actividades a lo largo del desarrollo.

Actividad	Inicio actividad	Fin actividad
Análisis de requerimientos	10/11/2021	10/11/2021
Especificación de requisitos	10/11/2021	11/11/2021
Determinar los objetivos del negocio	11/11/2021	12/11/2021
Análisis de riesgos	19/11/2021	20/11/2021
Desarrollo del Data warehouse	14/12/2021	19/12/2021
Integración de datos	19/12/2021	24/12/2021
Proceso ETL	25/01/2022	27/01/2022

Desarrollo de las métricas y KPI's	02/02/2022	06/02/2022
Desarrollo de Dashboards (tableros)	06/02/2022	20/02/2022
Evaluación del producto de software utilizando MyFeps ágil	25/02/2022	10/04/2022

Tabla 1: Planificación del proyecto

### 3.2 Análisis de requerimientos

Durante la fase de análisis se llevaron a cabo las primeras entrevistas con el cliente para conocer, principalmente lo que el cliente desea.

Asimismo, gracias a las entrevistas pudimos entender cómo funciona el negocio, cuáles son sus métricas, cuál es su objetivo principal y los recursos tecnológicos que el cliente dispone.

Para este caso de uso, se utilizó la técnica de entrevista con preguntas abiertas, semiabiertas y cerradas para la obtención de requisitos funcionales y no funcionales.

En base a lo conversado con el cliente y a las necesidades del negocio, se realizó un estudio de mercado donde se logró determinar un conjunto de tecnologías y herramientas que satisfacen las necesidades del cliente.

El cliente nos contó que no quiere invertir demasiado dinero en infraestructura TI, por lo que mi solución fue brindarle lo que el cliente desee con herramientas open source (en su mayoría). A continuación, se detallan los requisitos que se lograron extraer de las entrevistas con el cliente.

#### 3.2.1 Especificación de requisitos funcionales (RF) y requisitos no funcionales (RNF)

RF: El sistema debe presentar una interfaz gráfica estandarizada para todas las vistas.

RF: El sistema debe ser capaz de filtrar las ventas anuales y por provincia.

RF: El sistema debe presentar un mapa regional mostrando las sucursales disponibles con puntos representados en diferentes colores.

RF: El sistema debe mostrar por cada vista al menos 2 (dos) visualizaciones con estadísticas descriptivas.

RF: El sistema debe presentar, en la barra lateral izquierda, al menos tres botones de desplazamiento de menús.

RF: El sistema debe ser capaz de mostrar un gráfico de anillo que diferencie a los clientes por género, siendo este hombre o mujer.

RF: El sistema debe mostrar la cantidad y porcentaje de clientes cuyo género sea hombre o mujer.

RF: El sistema debe ser capaz de permitir al usuario elegir uno o varios clientes.

RNF: La herramienta de explotación de datos debe ser Power BI en su versión desktop.

RNF: El sistema debe ser capaz de sincronizar los filtros con los gráficos.

RNF: La paleta de colores deberá ser uniforme en todas las vistas.

RNF: El modelo de datos para el datamart debe ser estrella, sin jerarquías.

RNF: El servidor de la base de datos (SQL Server) debe contar con al menos un usuario administrador y una contraseña.

RNF: El tiempo de respuesta con el datamart ventas debe ser menor a 5 (cinco) segundos

RNF: El sistema debe ser capaz de establecer una conexión con el cubo multidimensional en un tiempo menor a 5 segundos.

RNF: El lenguaje de programación para el ETL deberá ser Python3 y TL-SQL.

### **3.3 Determinar los objetivos del negocio.**

Esta etapa es la inicial y consiste en determinar lo que impulsa al negocio; es decir, aquél atributo principal que hace que el negocio funcione y/o tenga éxito. En este caso el atributo principal son las ventas de productos.

### **3.4 Análisis de riesgos**

En esta etapa, se hace un análisis de riesgos del proyecto y del producto, con el fin de mitigarlos para reducir la probabilidad de ocurrencia de los mismos.

A continuación, se define la escala numérica y la ponderación de la probabilidad de un riesgo con los siguientes valores:

#### **3.4.1 Escala de riesgos**

En este apartado se define la escala numérica de riesgos siendo:

Crítico: 5

Alto: 4

Medio: 3

Bajo: 2

Insignificante: 1

### 3.4.1 Ponderación de Probabilidades de ocurrencias

En este apartado se define la escala de ponderación siendo:

Casi segura: 5

Probable: 4

Moderada: 3

Improbable: 2

Muy remota: 1

### 3.4.3 Matriz de riesgos del proyecto

En este apartado se detalla la matriz de riesgos que se han detectado a lo largo del proyecto.

El valor total, en puntos, de un riesgo, está dado por el producto entre el impacto del riesgo y la probabilidad o frecuencia de ocurrencia del mismo. La escala utilizada en este TFC es de 1 a 100, siendo 1 el valor más bajo y 100 el valor más alto.

Amenazas o eventos de riesgos.	Impacto	Probabilidad o frecuencia	Valor de riesgo	Decisión
Mala comunicación del cliente	Crítico: (5) (No establecer un entendimiento de lo que el cliente desee, no se podrían especificar los requerimientos)	Casi segura: (5) (las sucesivas iteraciones lo hacen poco probable)	5x5 = 25	Aceptar el riesgo
Especificación de requisitos.	Crítico: (5) (Si no especifica bien un requisito, lo más probable es que el sistema no cumpla con lo que el cliente solicitó.)	Probable: (4) (Establecer un requisito ambiguo o mal especificado ocurre de manera frecuente.)	4x5 = 20	Controlar: se debe verificar el requisito con el cliente para poder validarlo
Integración de datos.	Alto: (4). (Si los orígenes de datos están mal especificados, la información puede que no sea la esperada por el cliente.	Moderado (3): (El cliente suele tener conocimiento de los orígenes de datos.)	4x3 = 12	Controlar: se debe verificar con el cliente que las fuentes de datos sean las correctas
Seguridad en las bases de datos.	Crítico: (5). (No establecer una contraseña para un	Muy remota: (1). (se define un usuario	5x1 = 5	Controlar: se debe establecer un control que verifique los

	grupo de usuarios der las bases de datos, compromete a la información de la compañía)	administrador con una clave alfanumérica)		permisos y las contraseñas de un grupo de usuarios de las bases de datos.
Seguridad en infraestructura	Crítico: (5). (Si el servidor no dispone de una contraseña, entonces es un blanco fácil para atacantes)	Muy remota: (1). (Es obligatorio en los sistemas operativos orientados a servidores introducir una clave con ciertas especificaciones de caracteres y longitudes)	5x1 = 5	Controlar: se debe establecer un control que verifique la seguridad de las claves y los cortafuegos del servidor se encuentren habilitados.
Disponibilidad en infraestructura	Crítico: (5). (El servidor no se encuentre disponible)	bajo: (2). (En los sistemas on premise, la probabilidad de que un servidor con recursos suficientes falle, es baja)	5x2 = 10	Aceptar el riesgo.

Tabla 2: Análisis de riesgos del proyecto

### 3.4.4 Matriz de riesgos del producto

En este apartado se detalla la matriz de riesgos del producto donde se han identificado las siguientes amenazas o riesgos que se han detectado a lo largo del desarrollo del producto.

Amenazas o eventos de riesgos.	Impacto	Probabilidad o frecuencia	Valor de riesgo	Decisión
Mal diseño del tablero	Alto: (4). (no exista la suficiente información para proporcionar lo que el cliente solicitó)	Moderada: (3). (la constante interacción con el cliente hace que no sea tan probable que ocurra este suceso)	4x3 = 12	Controlar: verificar que el diseño cumple con lo solicitado por el cliente.
Datos incorrectos en el tablero	Crítico: (5). (Los datos que presenta el tablero sean incorrectos)	Improbable: (2). (Las fuentes de información y los datos que estas presentan, fueron validadas por el	5x2 = 10	Aceptar el riesgo.

		cliente)		
Mal diseño del datawarehouse	Crítico: (5). (Un mal diseño del datawarehouse implica problemas de data y performance)	improbable: (2). (El diseño de facts y dims fue validado con el cliente. Es por ello que no sea tan probable que ocurra este suceso.)	5x2 = 10	Aceptar el riesgo.
flujo ETL errado.	Alto: (4). (El flujo de datos a través del ETL no es correcto.)	Probable: (4). (Es probable que la pureza de los datos no llegue 100% correcta a destino en la mayoría de casos)	4x4 = 16	Controlar: verificar por qué el flujo de datos no llega a destino. Corregir posibles errores.

Tabla 3: Análisis de riesgos del producto

### 3.5 Especificar la solución

Tras haber estado en contacto con el cliente y haber especificado los requisitos funcionales y no funcionales del sistema, se procede a detallar la solución propuesta para este trabajo final de carrera.

Esta solución de Business Intelligence, está basada, principalmente en ofrecer datos de importancia al negocio para que, desde la gerencia, se puedan tomar decisiones lo más precisamente posible. La arquitectura de este producto es de cliente-servidor.

Este producto está compuesto por un proceso ETL que se encarga de integrar los datos provenientes de diversas fuentes de información con el objetivo de disponibilizar los datos en el datawarehouse y, más en concreto, en el datamart de ventas. Una vez que la información se encuentre disponible en el datamart, es necesario emplear una herramienta de BI que sea capaz de consumir y explotar dicha información del mismo datamart o bien, del cubo OLAP que va a ser generado a partir de ese datamart, con el objetivo de brindar tableros que contengan la información disponible, en tiempo real, al usuario final.

### 3.6 Identificar los orígenes de datos.

Durante esta fase, se determinarán los orígenes de datos que se requerirán para el proyecto.

A la hora de realizar la ingesta de datos, es una buena práctica conocer los sistemas de información que utiliza el negocio realizando entrevistas con el cliente para identificar las fuentes de información. Estas fuentes, idealmente son sistemas de información que generan

datos y estos pueden venir en distintos formatos y/o extensiones de archivos, tales como .txt, JSON, csv (mayormente vienen de este tipo), xlsx, etc. Para este caso de uso, se cuentan con más de un origen de datos; la base de datos relacional BDVentas y un dataset con extensión .csv.

En la base de datos relacional, se posee información sobre los productos, vendedores, ventas con su detalle, clientes y sucursales, mientras que en el segundo origen, se encuentran los clientes actuales del negocio y en el tercero, las sucursales donde opera el negocio.

### 3.6.1 Análisis de la base de datos.

Para este proyecto, se tomó esta base de datos para aumentar un grado de realismo al mismo, ya que la base de datos principalmente se encuentra en español y, además, posee distintos errores que se analizarán en este trabajo y que, posteriormente, se brindará, como oportunidad de mejora, una solución para reparar los mismos y poder disponibilizar mejor la información en la base de datos.

A continuación, se mostrará el diagrama de entidad relación (DER) de la base de datos en MySQL Server.

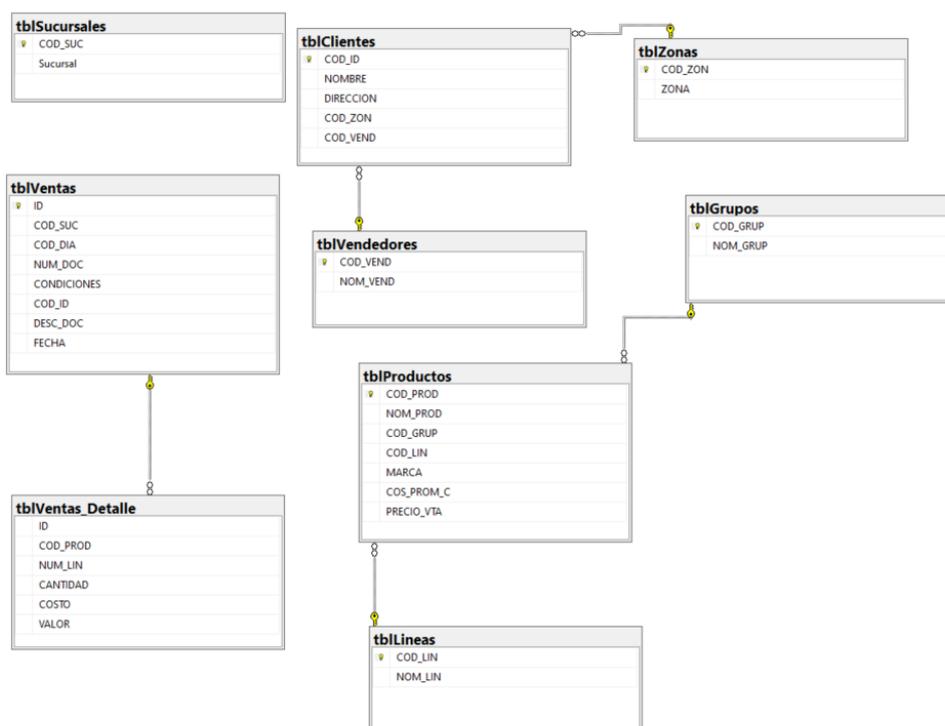


Figura 10: Base de datos

Fuente: imagen extraída de mi propio desarrollo.

La base de datos BDVentas, presenta varios errores en medida; grandes, medianos y pequeños.

A simple vista, es posible visualizar errores notables de modelado de datos, entre ellos; tablas sin un tipo de relación alguna, incluso teniendo campos definidos como claves primarias (PK) y campos que no están siendo utilizados.

Un claro ejemplo de ello se puede encontrar en la tabla de ventas (tblVentas). Esta tabla posee un campo denominado COD ID; representa a los clientes que hicieron una compra. Dicho campo, como es de esperar, se puede encontrar en la tabla tblClientes como PK (primary key) existe una relación establecida en esas tablas por ese campo.

### **3.6.2 Estrategia para la unificación de datos**

Para hacer más dinámico e interesante este proyecto académico, se tomará, de la base de datos relacional, la tabla tblProductos para luego, insertar cada registro tomado de la tabla productos en la futura dimensión productos. Se tomará la columna COD\_ID de la tabla clientes de la base de datos relacional y se tomarán los clientes del dataset clientes.csv [7] para crear un dataset que contenga el COD\_ID de un cliente sumado a las columnas seleccionadas del dataset clientes.csv, con el fin de que cada cliente esté asociado a un código de venta mediante el campo COD\_ID para, posteriormente, poder insertar a los clientes que hayan realizado una compra en el data mart Ventas. Además del dataset mencionado anteriormente, se van a crear 5 (cinco) sucursales que se insertarán posteriormente en la D\_Sucursal.

Adicionalmente, me gustaría hacer mención a que se pudo haber tomado de ejemplo toda la tabla clientes de la base de datos relacional evitando la creación del dataset que posteriormente se inserta en la D\_Clientes. El motivo de elegir la segunda opción es para representar un proyecto con un grado más de realismo, siendo que en los proyectos de BI, al recopilar los orígenes de datos para luego unificarlos, estos no suelen estar consolidados en un una sola tabla o archivo, sino que existe una relación

indirecta entre los mismos que hay que detectar para luego unificarlos y lograr que quede la información procesada, relacionada y disponibilizada de manera correcta.

## **3.7 Desarrollo del datawarehouse**

En este apartado, se detalla la solución de la arquitectura utilizada para este datawarehouse.

### **3.7.1 Descripción del DW**

El Datawarehouse que se propone en este trabajo de investigación, está diseñado con el fin de explotar información de los distintos datamarts de la empresa (en este caso, ventas) y está diseñado para empresas Pymes de 100 hasta 200 empleados y cuya actividad principal sean las ventas.

### **3.7.2 Justificación de la elección del Modelo Estrella**

Para este Data Warehouse, se optó por la arquitectura de modelo estrella, ya que idealmente este DW está pensado, principalmente, para un negocio donde no es necesario tener conocimiento total del dato. Es por ello que no se optó por un esquema híbrido o copo de nieve, dado que para este caso no es necesario contar con más jerarquías para obtener mayor granularidad del dato; como podría ser una dimensión "Productos" que tenga un campo denominado "categoría" y que este haga referencia a otra tabla que se llame "subcategoría". Sin embargo, este mismo puede ser modificado de tal manera que si el cliente desearía incorporar nuevas tablas al mismo, el sistema lo permitiría sin problema alguno.

### **3.7.3 Objetivos y alcance**

Este datawarehouse tendrá un solo datamart con el fin de mostrar las ventas del negocio.

### **3.7.4 Objetivo principal**

Conocer el número y el volumen de ventas que realizó el negocio en un período de tiempo histórico.

### **3.7.5 Objetivos generales**

En esta sección se detalla un listado de ítems con los objetivos generales y específicos que se desea conocer.

- Número y facturación de ventas por región.
- Variación de ventas según cantidad y facturación por sucursal.
- Ventas anuales: por región o en total.
- Conocer la facturación y volumen de ventas de cada sucursal.

- Top N productos con mayor volumen de ganancias.
- Conocer la marca y el producto más vendido.
- Cantidad de productos vendidos esperada vs objetivo.
- Variación de ganancias anuales por marca.
- Conocer la ubicación de los clientes para saber poder determinar aquellos clientes que más compras efectúan y de qué regiones provienen.
- Conocer el promedio de edad de los clientes y también la edad de cada cliente en particular.
- Conocer el top N clientes que realizaron más compras.

### 3.7.6 Arquitectura del DW

La arquitectura de este DW está compuesta por 1 (una) tabla central de hechos (fact table) y 4 (cuatro) tablas de dimensiones (dim tables).

Para identificar las tablas de hechos, se utilizó la nomenclatura F\_NombreDeTablaHecho, donde F se refiere a Fact, seguido de un guión bajo y el nombre de la tabla. Donde el nombre de la tabla está compuesto por la primera letra en mayúscula y el resto de letras en minúsculas. Gramática BNF para armar una tabla de hechos:

```
<hechos> ::= F_ <Lmayusc> <Lminusc>  
<Lmayusc> ::= A|B|C|...|Z  
<Lminusc> ::= a|b|c|...|z <letra> | A|B|C|...|Z | a|b|c|...|z
```

De manera similar, para identificar las tablas dimensiones

D\_NombreDeTablaDimension.

donde D se refiere a Dimension, seguido de un guión bajo y el nombre de la tabla.

Gramática BNF para armar una tabla de dimensiones:

```
<dimensiones> ::= D_ <Lmayusc> <Lminusc>  
<Lmayusc> ::= A|B|C|...|Z  
<Lminusc> ::= a|b|c|...|z <letra> | a|b|c|...|z
```

Es una buena práctica mantener los nombres de las tablas en plural siempre que se hable de una entidad con múltiples entidades

### 3.7.6.1 Diseño físico del Data Warehouse.

En este apartado, se muestra el diseño físico del datawarehouse realizado en MySQL Server 2021.

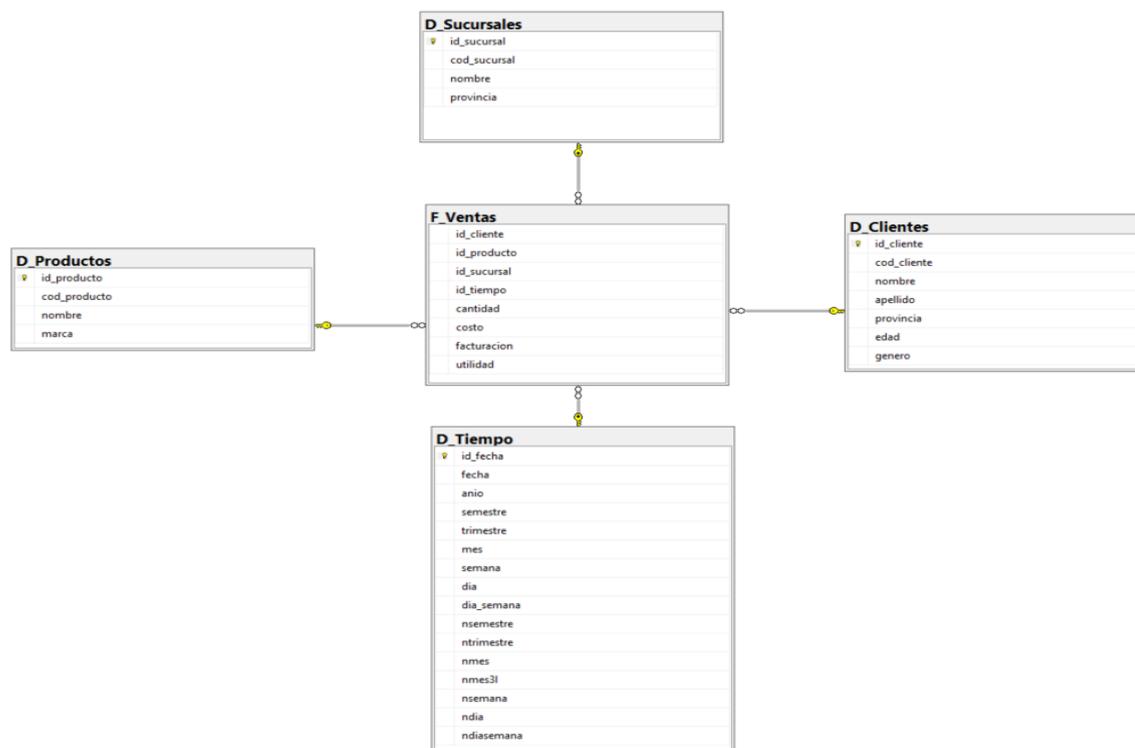


Figura 9: Diseño físico del Data Warehouse

Fuente: imagen extraída de mi propio desarrollo.

### 3.7.6.2 Estructura y tipo de datos de las tablas.

A continuación, se presentará el diseño de cada tabla con sus respectivos atributos, tipos de datos y llaves (PK y FK).

D_Clientes	
id_cliente	INT IDENTITY (1,1) NOT NULL PRIMARY KEY
cod_cliente	VARCHAR (7) NOT NULL
nombre	VARCHAR(50) NOT NULL
apellido	VARCHAR(50) NOT NULL
provincia	VARCHAR(50) NOT NULL

edad	INT NOT NULL
genero	VARCHAR(10) NOT NULL

Tabla 4: Dimension Clientes

D_Productos	
id_producto	INT NOT NULL PRIMARY KEY
cod_producto	VARCHAR(50) NOT NULL
nombre	VARCHAR(50) NOT NULL
marca	VARCHAR(50) NOT NULL

Tabla 5: Dimensión Productos

D_Sucursales	
id_sucursal	INT NOT NULL PRIMARY KEY
cod_sucursal	VARCHAR(50) NOT NULL
nombre	VARCHAR(50) NOT NULL
provincia	VARCHAR(50) NOT NULL

Tabla 6: Dimensión Sucursales

En la dimensión tiempo, es posible que uno o más campos no se utilicen en el reporte final, debido a que no es necesario usar todos los campos. El motivo de la creación de todos estos campos consiste en brindarle al cliente, si decide, la posibilidad de tener una mayor granularidad del dato haciendo uso de los mismos en posteriores reportes.

D_Tiempo	
id_tiempo	INT NOT NULL PRIMARY KEY
fecha	DATETIME NOT NULL
dia	SMALLINT NOT NULL
mes	SMALLINT NOT NULL
anio	SMALLINT NOT NULL

semestre	SMALLINT NOT NULL
trimestre	SMALLINT NOT NULL
mes	SMALLINT NOT NULL
semana	SMALLINT NOT NULL
día	SMALLINT NOT NULL
día_semana	SMALLINT NOT NULL
nsemestre	VARCHAR(15) NOT NULL
nmes	VARCHAR(15) NOT NULL (representa el nombre del mes)
nsemana	VARCHAR(15) NOT NULL nombre de la semana anual (ej semana 4)
ndía	VARCHAR(15) NOT NULL (representa el nombre del día)
ndiasemana	VARCHAR(15) NOT NULL (representa el nombre del día de la semana)

Tabla 7: Dimensión Tiempo

F_Ventas	
id_cliente	INT NOT NULL FK (referencia a D_Clientes)
cod_producto	INT NOT NULL FK (referencia a D_Productos)
cod_sucursal	INT NOT NULL FK (referencia a D_Sucursal)
id_tiempo	INT NOT NULL FK (referencia a D_Tiempo)
cantidad	INT NOT NULL
costo	FLOAT NOT NULL
facturación	FLOAT NOT NULL
utilidad bruta	FLOAT NOT NULL

Tabla 8: Fact Ventas

### **3.8 Proceso ETL.**

Durante esta etapa, se integrarán los datos que van a ser obtenidos a través de las fuentes de información identificadas en la etapa previa.

Para poder integrar las fuentes de información, se tienen que realizar procesos de extracción, transformación y carga de datos (ETL).

Actualmente, en los proyectos de Business Intelligence, existen diversas herramientas y lenguajes capaces de realizar estos procedimientos. No entraré en detalle mencionar todas las del mercado pero sí daré un enfoque desde mi punto de vista personal y profesional.

Las herramientas de ETL, son programas que corren por detrás muchos servicios y capas de abstracción. Si bien cumplen la función principal que se mencionó anteriormente, demoran mucho tiempo en realizar operaciones. Estas herramientas son utilizadas acorde a un proyecto.

#### **3.8.2 Elección de la herramienta de ETL**

Para este caso, utilizaré el lenguaje de programación Python en su versión 3.6 Este lenguaje tiene librerías que son capaces de facilitarnos el desarrollo de procesos ETL. Además, al trabajar directamente con el lenguaje en sí, el tiempo de ejecución y operaciones, a diferencia de otras herramientas de ETL como SSIS, Alteryx y Pentaho, es considerablemente menor y además nos permite escribir código fuente escalable. Por esa razón se ha decidido utilizar este lenguaje para desarrollar el ETL.

#### **3.8.3 Librerías utilizadas para el desarrollo del proceso ETL**

En esta sección se detallarán las librerías utilizadas para el desarrollo de este proceso ETL:

Pandas: es una librería que nos permite manipular datos en diferentes extensiones: .csv, xlsx, json, etc.

Pyodbc: es la librería que nos permite establecer una conexión con la base de datos mediante el conector de SQL server denominado ODBC [8]. Esta librería es capaz de extraer datos de la base como también manipularlos.

La aplicación que ejecuta el proceso ETL, cuenta con 5 archivos en extensión .py. Entre ellos se encuentran: librerías.py, funciones.py, db.py, orígenes.py, main.py

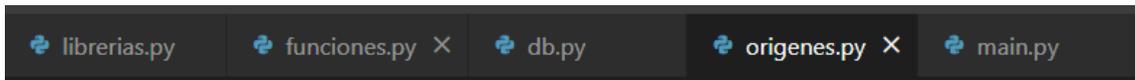


Figura 11: ETL en Python 3

Fuente: imagen extraída de mi propio desarrollo.

### 3.8.4 Explicación del proceso ETL

A continuación, se explicará el código fuente desarrollado.

librerias.py: En este módulo se importan todas las librerías utilizadas

```
import pandas as pd
import pyodbc
import datetime
import os
from logging import exception
```

db.py

```
from librerias import pyodbc
server = 'DESKTOP-TS5PPCK'
database = 'DW_Ventas'
username = 'admin'
password = 'tesinaduplaa2021'
connDW=pyodbc.connect('driver={SQLSERVER};server='+server+';database='+database+';UID='+username+';PWD='+ password')
```

En este módulo, se importa la librería pyodbc y se utiliza el método pyodbc.connect() para conectar con el datawarehouse.

quers.py

```
#Ventas
productos = "SELECT NOM_PROD, COD_PROD, MARCA FROM tblProductos"

#Creates
D_Clientes = "CREATE TABLE dbo.D_Clientes ( id_cliente INT IDENTITY(1,1) NOT NULL,
cod_cliente VARCHAR(7), nombre VARCHAR(50) NOT NULL, apellido VARCHAR(50) NOT
NULL, provincia VARCHAR(50) NOT NULL, edad INT NOT NULL, genero VARCHAR(10),
CONSTRAINT [PK_Clientes_id_cliente] PRIMARY KEY NONCLUSTERED (id_cliente) )"
```

```
D_Productos = "CREATE TABLE dbo.D_Productos ( id_producto INT NOT NULL
IDENTITY(1,1), cod_producto VARCHAR(50) NOT NULL, nombre VARCHAR(255) NOT NULL,
marca VARCHAR(255), CONSTRAINT [PK_Productos_id_producto] PRIMARY KEY
NONCLUSTERED (id_producto) )"

D_Sucursales = "CREATE TABLE D_Sucursales(id_sucursal INT NOT NULL
IDENTITY(1,1),cod_sucursal VARCHAR(50) NOT NULL,nombre VARCHAR(50) NOT
NULL,provincia VARCHAR(50) NOT NULL, CONSTRAINT [PK_Sucursal_id_sucursal]
PRIMARY KEY NONCLUSTERED (id_sucursal) )"

F_Ventas = "CREATE TABLE F_Ventas(id_cliente INT NOT NULL, id_producto INT NOT
NULL, id_sucursal INT NOT NULL, id_tiempo INT NOT NULL, cantidad FLOAT NOT NULL,
costo FLOAT NOT NULL,precio FLOAT NOT NULL, utilidad FLOAT NOT NULL
CONSTRAINT [FK_F_id_cliente] FOREIGN KEY (id_cliente) REFERENCES
D_Clientes(id_cliente), CONSTRAINT [FK_F_cod_producto] FOREIGN KEY (id_producto)
REFERENCES D_Productos(id_producto), CONSTRAINT [FK_F_cod_sucursal] FOREIGN
KEY (id_sucursal) REFERENCES D_Sucursales(id_sucursal), CONSTRAINT
[FK_F_id_tiempo] FOREIGN KEY (id_tiempo) REFERENCES D_Tiempo(id_fecha) )"

D_Tiempo = "CREATE TABLE dbo.D_Tiempo(id_fecha INT NOT NULL, fecha DATETIME NOT
NULL, anio SMALLINT NOT NULL, semestre SMALLINT NOT NULL, trimestre SMALLINT NOT
NULL, mes SMALLINT NOT NULL, semana SMALLINT NOT NULL, dia SMALLINT NOT NULL,
dia_semana SMALLINT NOT NULL, nsemestre VARCHAR(15) NOT NULL, ntrimestre
VARCHAR(15) NOT NULL,nmes VARCHAR(15) NOT NULL, nmes3l VARCHAR(15) NOT
NULL, nsemana VARCHAR(15) NOT NULL, ndia VARCHAR(15) NOT NULL, ndiasemana
VARCHAR(15) NOT NULL,CONSTRAINT [PK_Tiempo_id_fecha] PRIMARY KEY
NONCLUSTERED (id_fecha) )"

lista_tablas = [D_Clientes, D_Productos, D_Sucursales, D_Tiempo, F_Ventas]

#Inserts

insert_Clientes = "INSERT INTO D_Clientes (cod_cliente, nombre, apellido, provincia) VALUES
(?,?,?,?)"

insert_sucursal = "INSERT INTO D_Sucursal (cod_sucursal, nombre, provincia) VALUES (02,
Sucursal 504 - CABA, Buenos Aires - CABA), (04, Sucursal 211 - La Plata (LP), Buenos Aires -
GBA ),(03, Sucursal 512 - Arroyito, Córdoba), (06, Sucursal 212 - Rosario), (13, Sucursal 775 -
Chubut, Chubut), (07, Sucursal 1001 - Salta, Salta)"

insert_productos = "INSERT INTO D_Productos (cod_producto, nombre, marca)
VALUES(?,?,?)"3
```

<sup>3</sup> el carácter "?" representa los marcadores de posición para los parámetros.

En este módulo se encuentran declaradas las variables que, posteriormente, se le pasarán por parámetro al método execute del objeto cursor para la creación del datamart.

origenes.py

```
from librerias import pyodbc, pd, exception
from queries import productos
server = 'DESKTOP-TS5PPCK'
database = 'BDVentas'
df=pd.read_csv("https://datasets.datos.mincyt.gob.ar/dataset/06ae9728-c376-47bd-9c41-fbdca68707c6/resource/8ab77b16-f1a8-4d3f-b664-67becf83a9b9/download/personas.csv"
", sep=";")

conn_dbventas =
pyodbc.connect('driver={SQLSERVER};server='+server+';database='+database+';trusted_connections=YES')

#Se añade la columna cod_cliente al dataframe para insertar los datos de COD_ID de la tabla
clientes de la base de datos

df = df.assign(cod_cliente=' ')
df["sexo_id"]=df["sexo_id"].map(lambda x: "M" if x==1 else "H")

#reacomodamos las columnas del df clientes
df_clientes = df['cod_cliente','nombre','apellido','provincia','sexo_id','edad','edad','provincia']

#Borramos posibles registros duplicados en las columnas
df_clientes = df.drop_duplicates(subset=['cod_cliente','nombre','apellido','provincia','sexo_id','edad','edad'])

try:
    cursor = conn_dbventas.cursor()
    query = cursor.execute(productos)
    data = cursor.fetchall()
    df_productos = pd.read_sql(data, conn_dbventas)
    cursor.close()
except exception as e:
    print(f"Error: '{e}'")
```

En este módulo se obtienen los datos del dataframe mencionado anteriormente y la tabla productos de la BDVentas.

- En primer lugar, se agrega la columna cod\_cliente en el dataframe luego, poder insertar los datos al datamart.
- En segundo lugar, se acomodan las columnas del dataframe.
- En tercer lugar, se realiza una eliminación de duplicados (si existen) para que no existan registros duplicados posteriormente.
- Finalmente, se establece una conexión con la base de datos BDVentas para generar un dataframe, pasándole como parámetro al objeto cursor la variable productos que contiene la siguiente query con los campos especificados para generar el dataframe.

**NOTA:** El método fetchall() se encarga de extraer los registros de la query ejecutada en el cursor previamente para poder, en conjunto con pandas, generar el dataframe df\_productos para usarlo posteriormente.

funciones.py

```
from queries import *
from db import *
from origenes import df_clientes, df_productos
from librerias import *

def cargarTablas(lista_tablas, conn):
    try:
        cursor = connDWH.cursor()
        for tabla in lista_tablas:
            cursor.execute(tabla)
            cursor.commit()

        cursor.close()
    except exception as e:
        print(f"Error: '{e}'")

def insertarClientes(df, conn):
    try:
```

```
cursor = connDW.cursor()
cursor.execute("TRUNCATE TABLE D_Clientes")
for i, row in df_clientes.iterrows():
    cursor.execute(insert_Clientes, row.codigo, row.nombre, row.apellido, row.localidad)

connDW.commit()
cursor.close()
except exception as e:
    print(f"Error: '{e}'")

def insertarSucursal(query, conn):
    try:
        cursor = connDW.cursor()
        cursor.execute("TRUNCATE TABLE D_Sucursales")
        cursor.execute(insert_sucursal)
        cursor.commit()
        cursor.close()
    except exception as e:
        print(f"Error: '{e}'")

def insertarProductos(df, conn, query):
    try:
        cursor = connDW.cursor()
        cursor.execute("TRUNCATE TABLE D_Productos")
        for i,row in df_productos.iterrows():
            cursor.execute(insert_productos, row.COD_PROD, row.NOM_PROD, row.MARCA)
        cursor.commit()
        cursor.close()
    except exception as e:
        print(f"Error: '{e}'")

#Carga el script que llena la Dim tiempo y la F ventas

def cargarScript(conn):
    cursor = connDW.cursor()
    try:
        for archivo in os.listdir():
            if archivo.endswith('.sql'):
                fd = open(archivo, 'r')
```

```
sqlFile = fd.read()
fd.close()
sqlCommands = sqlFile.split(';')
for command in sqlCommands:
    try:
        cursor.execute(command)
    except OperationalError, msg:
        print("Error: ", msg)
cursor.commit()
cursor.close()
except exception as e:
    print(e)
```

El módulo funciones.py, contiene el código fuente de las funciones desarrolladas.

loadDW.sql

Este script se encarga de poblar la dimensión tiempo como también el data mart ventas.

```
GO
DECLARE @FechaDesde as datetime, @FechaHasta as datetime
DECLARE @FechaAAAAMMDD int
DECLARE @Año as smallint, @Semestre as smallint, @Trimestre smallint, @Mes smallint
DECLARE @Semana smallint, @Dia smallint, @DiaSemana smallint
DECLARE @NSemestre varchar(15), @NTrimestre varchar(15), @NMes varchar(15)
DECLARE @NMes3l varchar(15)
DECLARE @NSemana varchar(15), @NDia varchar(15), @NDiaSemana varchar(15)

BEGIN TRANSACTION
    SELECT @FechaDesde = CAST('20070101' AS Datetime)
    SELECT @FechaHasta = CAST(CAST(YEAR(GETDATE()+2 AS CHAR (4)) + '1231' AS
datetime)
    WHILE (@FechaDesde <= @FechaHasta) BEGIN
    SELECT @FechaAAAAMMDD = YEAR (@FechaDesde) *10000+
        MONTH (@FechaDesde) *100+
        DATEPART (dd, @FechaDesde)
    SELECT @Año = DATEPART (yy, @FechaDesde)
    SELECT @Semestre = MONTH(GETDATE())/6+1
    SELECT @Trimestre = DATEPART(qq, @FechaDesde)
    SELECT @Mes = DATEPART (m, @FechaDesde)
    SELECT @Semana = DATEPART (wk, @FechaDesde)
```

```
SELECT @Dia = RIGHT('0' + DATEPART (dd, @FechaDesde), 2)
SELECT @DiaSemana = DATEPART (DW, @FechaDesde)
SELECT @NMes = DATENAME (mm, @FechaDesde)
SELECT @NMes3l = LEFT (@NMes, 3)
SELECT @NSemestre = concat('Semestre' ,@SEMESTRE)
SELECT @NTrimestre = 'T' + CAST (@Trimestre as CHAR (1)) + '/' + RIGHT(@Año, 2)
    SELECT @NSemana = 'Sem' +CAST(@Semana AS CHAR(2)) + '/' +
RIGHT(RTRIM(CAST(@Año AS CHAR(4))), 2)
SELECT @NDia = CAST (@Dia as CHAR(2)) + ' ' + RTRIM (@NMes)
SELECT @NDiaSemana = DATENAME(dw, @FechaDesde)
INSERT INTO DW_VENTAS.dbo.D_Tiempo
(
id_fecha,
fecha,
anio,
semestre,
trimestre,
mes,
semana,
dia,
dia_semana,
nsemestre,
ntrimestre,
nmes,
nmes3l,
nsemana,
ndia,
ndiasemana
)
VALUES
(
@FechaAAAAMMDD,
@FechaDesde,
@Año,
@Semestre,
@Trimestre,
@Mes,
@Semana,
```

```
@Dia,  
@DiaSemana,  
@NSemestre,  
@NTrimestre,  
@NMes,  
@NMes3l,  
@NSemana,  
@NDia,  
@NDiaSemana  
)  
SELECT @FechaDesde = DATEADD(DAY, 1, @FechaDesde)  
END  
COMMIT TRANSACTION  
GO  
INSERT INTO DW_VENTAS.dbo.F_Ventas  
SELECT dc.id_cliente, dp.id_producto, ds.id_sucursal, dt.id_fecha,  
SUM(tvd.CANTIDAD) AS CANTIDAD,  
SUM(tvd.CANTIDAD*tvd.COSTO) AS COSTO,  
SUM(tvd.CANTIDAD*tvd.VALOR) AS FACTURACION,  
SUM(tvd.CANTIDAD*tvd.VALOR-tvd.CANTIDAD*tvd.COSTO) AS [UTILIDAD BRUTA]  
FROM BDVentas.dbo.tblVentas tv  
INNER JOIN  
    DW_VENTAS.dbo.D_Clientes dc ON tv.COD_ID = dc.cod_cliente COLLATE  
DATABASE_DEFAULT  
INNER JOIN  
    DW_VENTAS.dbo.D_Productos dp ON tvd.COD_PROD = dp.cod_producto COLLATE  
DATABASE_DEFAULT  
INNER JOIN  
    DW_VENTAS.dbo.D_Sucursales ds ON tv.COD_SUC = ds.cod_sucursal COLLATE  
DATABASE_DEFAULT  
INNER JOIN  
    DW_VENTAS.dbo.D_Tiempo dt ON tv.FECHA = dt.FECHA  
GROUP BY dc.id_cliente, dp.id_producto, ds.id_sucursal, dt.id_fecha;
```

Finalmente, el módulo main se encarga de llamar a todas las funciones con sus respectivos parámetros.

main.py

```
from funciones import *

def main():
    try:
        cargarTablas(lista_tablas)
        insertarClientes(df_clientes, connDW)
        insertarProductos(df_productos, connDW)
        insertarSucursal(D_Sucursales, connDW)
        cargarScript(connDW)
    except exception as e:
        print("No se pudo ejecutar proceso ETL. Error: ", e)
```

### 3.8.5 Detalle programación de funciones del ETL

En este apartado se definen las funciones del ETL y su descripción.

Función	Descripción
cargarTablas(lista_tablas)	Recibe como parámetro la lista de tablas que contienen las queries para armar el datamart.
insertarClientes(df, conn)	Recibe como parámetro el df_clientes de orígenes.py y la conexión al DW para poder poblar el dataframe en el datamart.
insertarProductos(df_productos, connDW)	Recibe como parámetro el df_pdocutos (creado en orígenes.py) y la conexión al DW.
insertarSucursal(D_Sucursales, connDW)	Recibe como parámetro la query D_Sucursales y la conexión al DW. En este caso, a diferencia de los anteriores, se carga es una query con las sucursales del negocio.
cargarScript(connDW)	Esta función recibe como parámetro la conexión al DW y su funcionamiento es el siguiente: identifica los archivos .sql en un directorio y abre el archivo "loadDW.sql" en modo de lectura para que este pueda ser ejecutado. El script loadDW.sql, se encarga de poblar la dimensión tiempo como también el datamart entero (en este caso, el datamart ventas).
main()	Función principal del programa que hace las llamadas al resto de funciones.

Tabla 9: Tabla de funciones - ETL

### 3.9 Creación del Cubo OLAP

En esta sección, se detalla paso a paso la creación del cubo OLAP.

Habiendo aclarado en el marco teórico lo que implica un cubo OLAP y su funcionamiento, quisiera hacer una aclaración antes de comenzar con esta sección; los KPIS utilizados en este trabajo final de carrera, se implementará directamente en el reporte de Power BI para corroborar su correcto funcionamiento, con el fin de implementarlas en un futuro en el cubo OLAP. Habiendo aclarado lo anterior, a continuación, se presenta la creación del Cubo OLAP en Visual Studio 2019.

#### 3.9.1 Conexión con el Data Warehouse

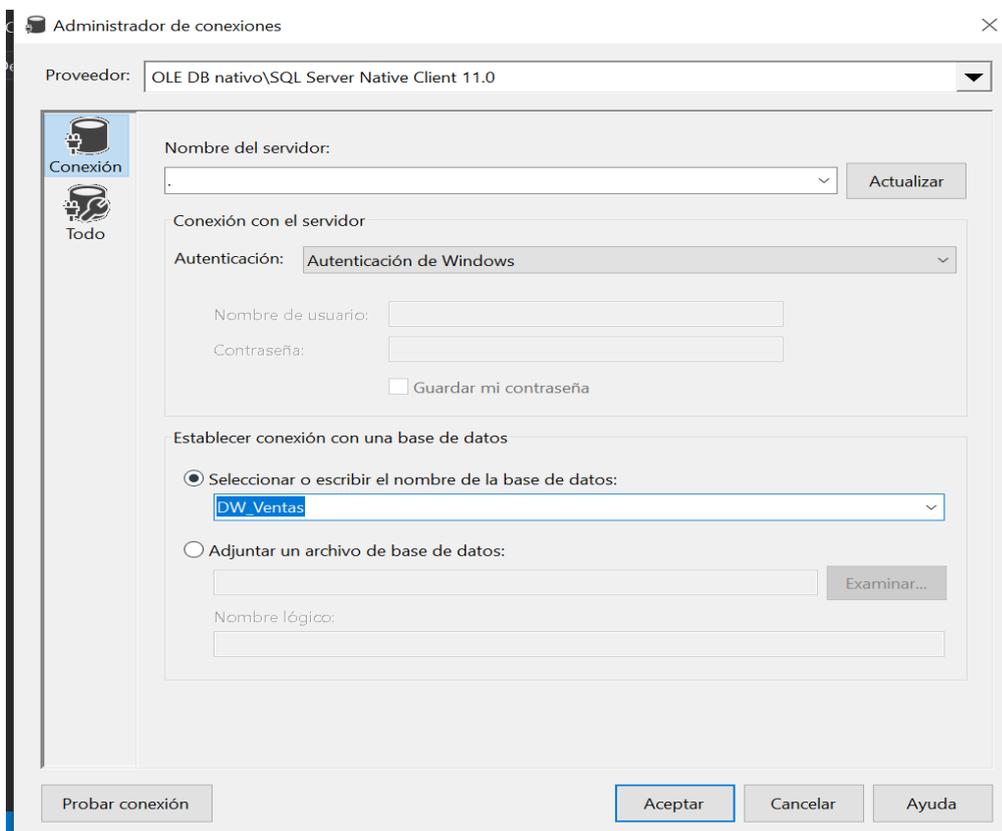


Figura 12: Conexión al DW

Fuente: imagen extraída de mi propio desarrollo.

Cada vez que se instala una conexión, se crea un objeto con una cadena de conexión. Internamente, dicha cadena, tiene las propiedades del objeto que se las pasa al driver de conexión que utiliza Windows para establecer la conexión con las bases de datos. El driver se

denomina ODBC (Open Database Connectivity), es el mismo conector que fue utilizado anteriormente en el proceso ETL.

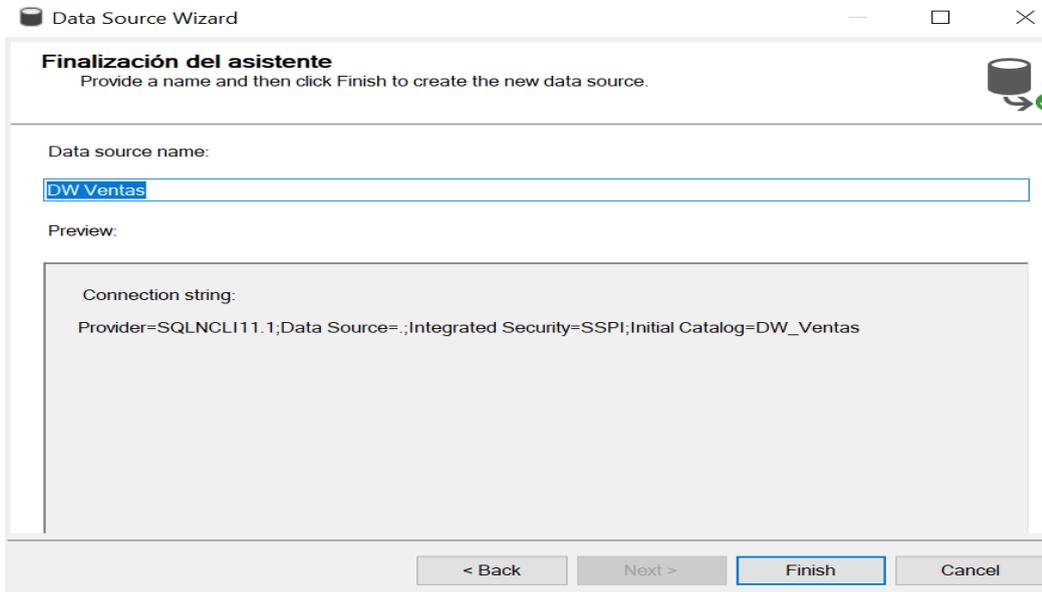


Figura 13: Driver de conexión

Fuente: imagen extraída de mi propio desarrollo.

### 3.9.2 Vistas de datos

En esta sección se eligen las tablas y/o vistas que deseemos agregar al cubo para consumirlas.

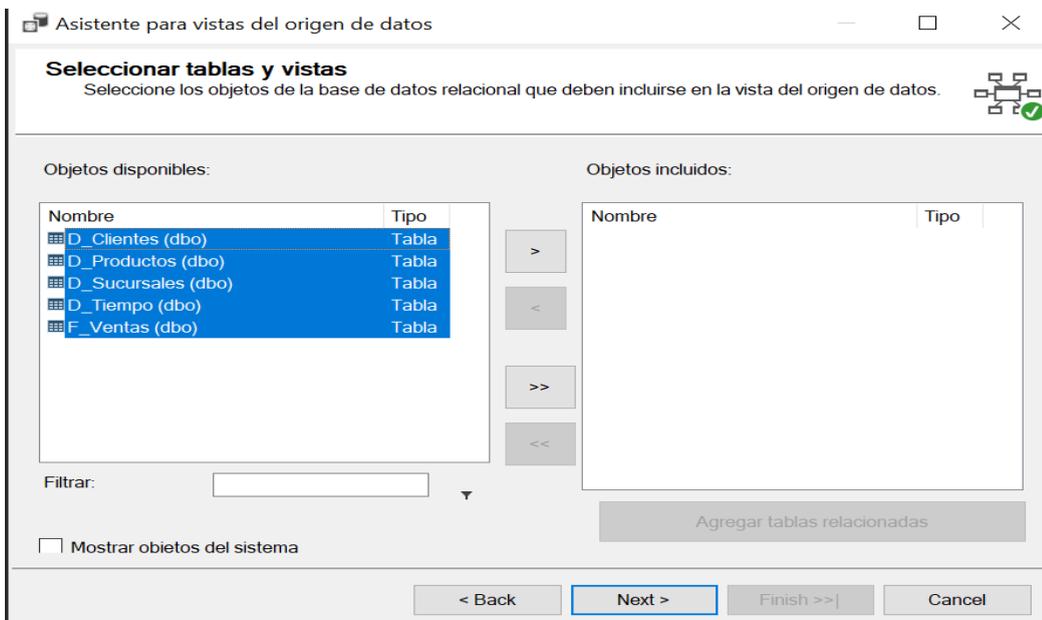


Figura 14: Selección de vistas de datos

Fuente: imagen extraída de mi propio desarrollo.

### 3.9.3 Selección de dimensiones

En esta sección, se crean las dimensiones correspondientes haciendo uso de los atributos que se deseen para la creación del cubo.

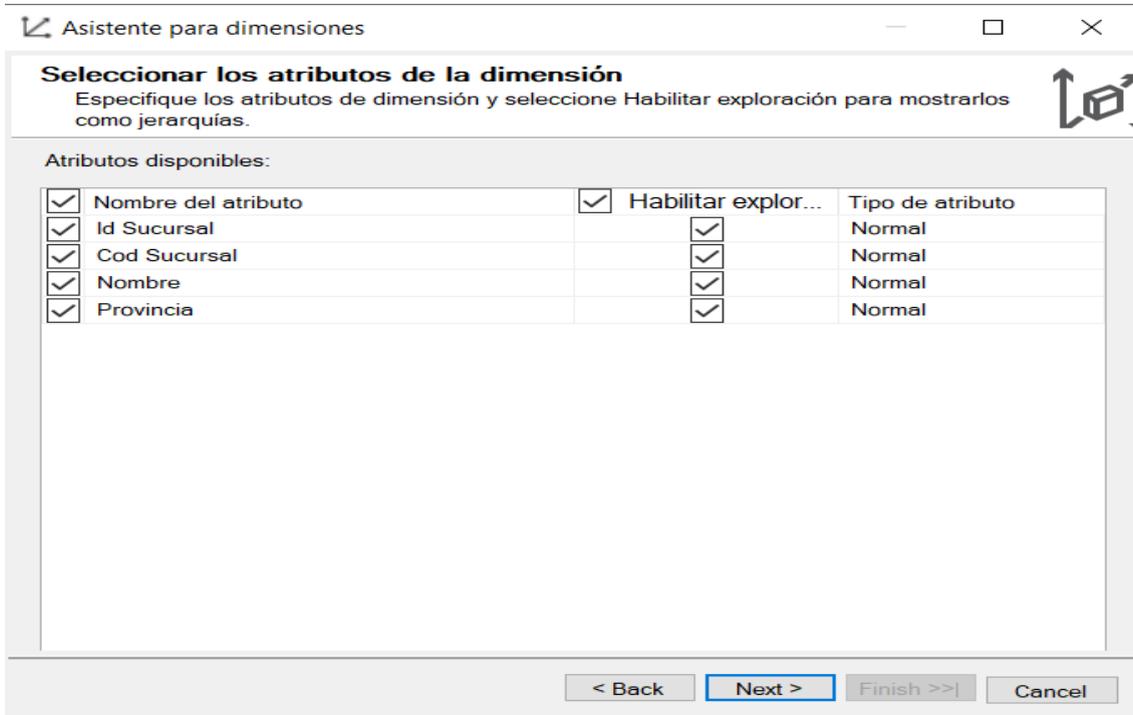


Figura 15: Selección de atributos

Fuente: imagen extraída de mi propio desarrollo.

Una vez seleccionadas todas las dimensiones que van a ser cargadas en el cubo, es posible generar el cubo para las dimensiones seleccionadas.

### 3.9.4 Generación del cubo

Para crear el cubo, se debe elegir la base de datos correspondiente con su o sus respectivas tablas de Facts.

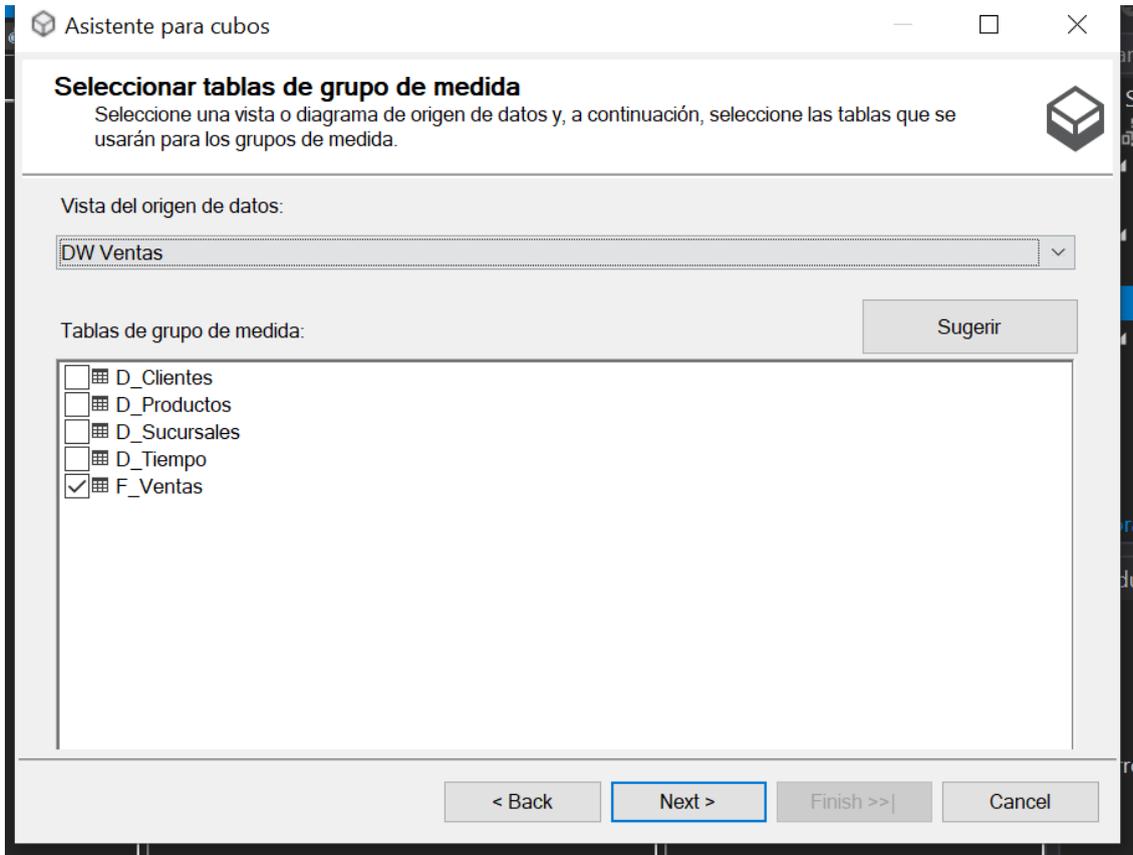


Figura 16: Creación del cubo

Fuente: imagen extraída de mi propio desarrollo.

### 3.9.5 Procesado y compilación del cubo

En primera instancia, para generar el cubo es necesario procesarlo. Este proceso, implica la lectura de las dimensiones y las facts asociadas al cubo.

Para comprobar si no hubo pérdida de datos, simplemente basta con hacer un COUNT(\*) en SQL Server a cada tabla Fact para conocer la cantidad de filas que existen en la tabla y compararla con la cantidad que se procesó en el cubo.

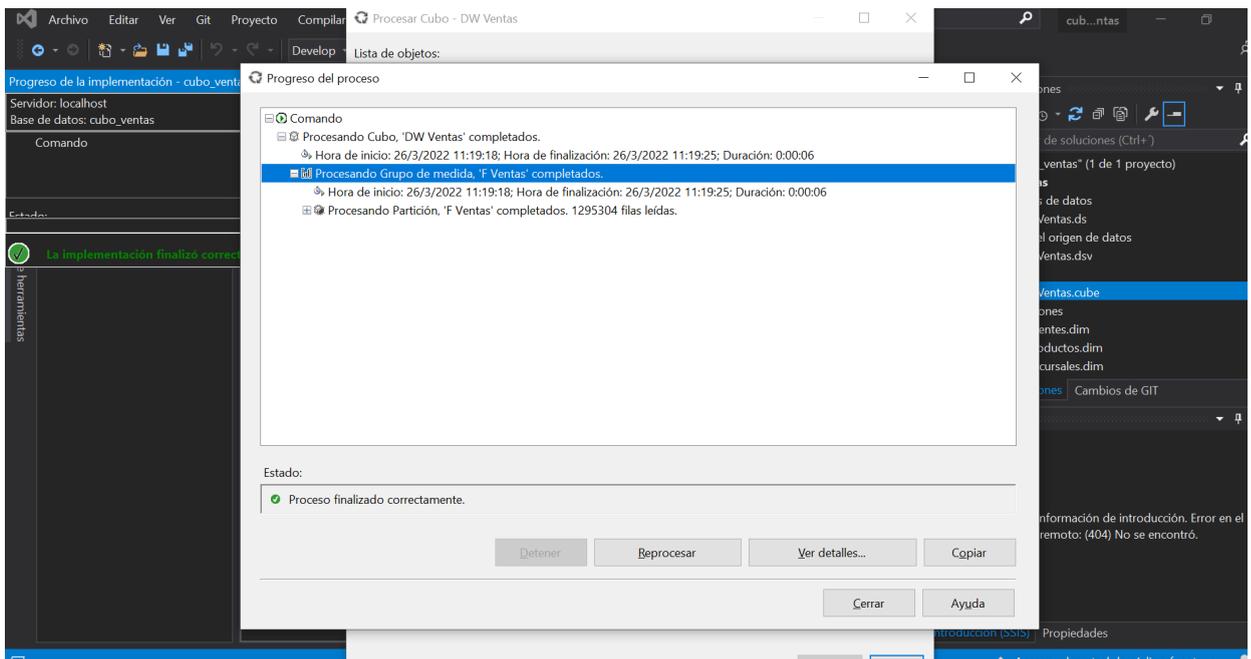


Figura 17: Procesado del cubo

Fuente: imagen extraída de mi propio desarrollo.



Figura 18: Comprobación de la cantidad de registros del cubo y datamart

Fuente: imagen extraída de mi propio desarrollo.

La cantidad de registros en el cubo y en la consulta que apunta a la F\_Ventas son iguales para una misma fecha, con lo cual, significa que el cubo no ha sufrido la pérdida de registros.

### **3.10 Desarrollo del informe en PowerBI**

En esta etapa, se procederá a detallar el armado del informe utilizando la herramienta Power BI Desktop. La creación del tablero consta en la confección de tres vistas mencionadas anteriormente. Este informe no pudo haberse realizado previamente sin contar con todos los datos transformados, unificados y consolidados en el data mart como fuente de origen.

#### **3.10.1 Orígenes de datos**

La obtención de datos para armar el informe puede ser desde el cubo olap como desde el propio data mart. Las ventajas de consumir los datos del cubo es que la información se actualiza en tiempo real, mientras que la desventaja principal es que no es posible crear métricas desde Power BI si los datos son consumidos desde el cubo. Las ventajas de consumir datos desde el datamart, son varias frente al cubo, entre ellas; creación de más tablas, creación de métricas y KPIS, creación de campos, etc. Es fundamental poder probar las métricas y KPIS en el informe para corroborar su correcto funcionamiento con el fin de, a futuro, implementarlas directamente en el cubo para un procesamiento en tiempo real. A continuación, se detallan los orígenes de datos tomados desde la herramienta PowerBI. La estructura general que tiene el tablero de PowerBI está compuesta por 3 capas: Datos, Modelo e Informe.

##### **3.10.1.1 Capa modelo de datos.**

En primera instancia, se debe seleccionar de donde se va a obtener el conjunto de datos que se utilizará para la creación de las vistas.

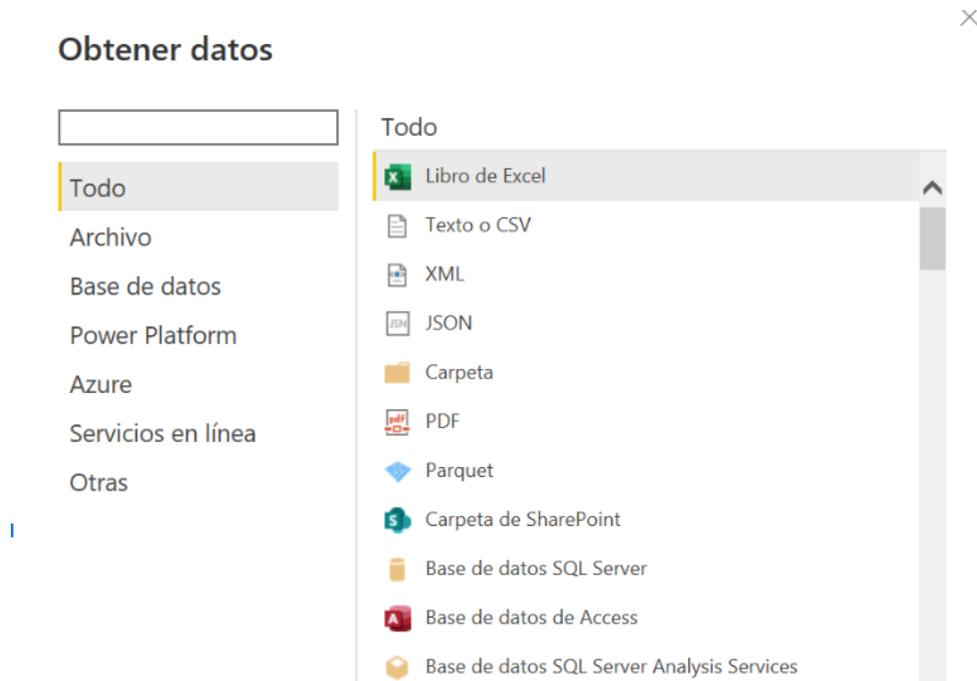


Figura 19: Capa Obtener conjunto de datos PoweBI

Fuente: imagen extraída de mi propio desarrollo.

### 3.10.1.2 Capa modelo.

Una vez especificada la o las fuentes de los orígenes de datos, se procede a hacer una serie de verificaciones en la capa Modelo. Las verificaciones sobre el conjunto de datos que comúnmente que suelen hacerse son las siguientes:

- Relaciones de tablas (dependiendo del conjunto de datos, estas pueden estar definidas o no)
- Cardinalidad entre relaciones<sup>4</sup>: si la cardinalidad es la misma que en el modelo de origen y en caso de no existir, crearla manualmente.
- La cantidad de tablas cargadas en PowerBI coincide con el conjunto de datos.
- Los nombres de las tablas sean correctos.

<sup>4</sup> En PowerBI la cardinalidad está denotada de 1 a 1 o 1 a N, siendo N representado por el símbolo \*.

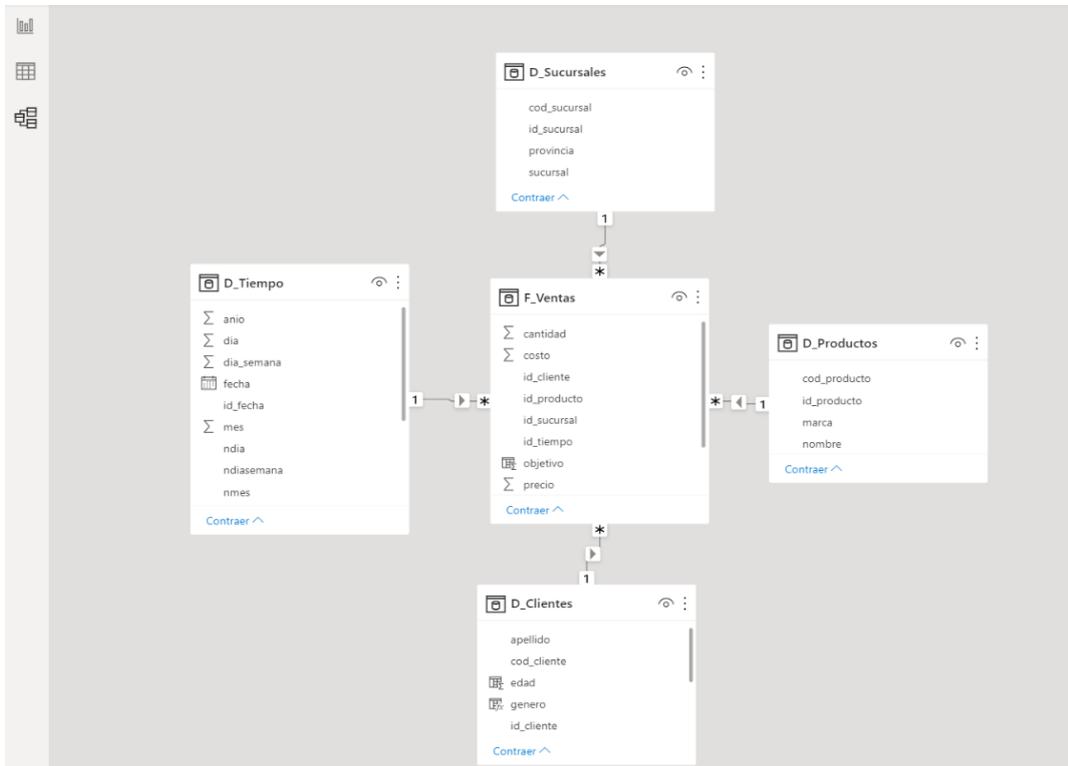


Figura 20: Capa Modelo PowerBI

Fuente: imagen extraída de mi propio desarrollo.

### 3.10.1.3 Capa de datos.

En la capa Datos, se muestran las tablas con sus respectivos datos. Además, esta capa permite realizar operaciones con los datos y crear medidas utilizando Power Query [11].

id_cliente	id_producto	id_sucursal	id_tiempo	cantidad	costo	precio	utilidad	tiempo	objetivo
49	14720	2	20130805	1	13,7	\$16	2,74	5/8/2013 00:00:00	8,27726110350341
49	14720	2	20150111	1	13,7	\$16	2,74	11/1/2015 00:00:00	28,65412514191112
49	14720	2	20131116	1	13,7	\$16	2,74	16/11/2013 00:00:00	6,98592351749539
49	14720	2	20130812	1	13,7	\$16	2,74	12/8/2013 00:00:00	5,54896145593375
49	14720	2	20131001	1	13,7	\$16	2,74	1/10/2013 00:00:00	14,8030481720343
49	14720	2	20150323	1	13,7	\$16	2,74	23/3/2015 00:00:00	17,9043489228934
49	14720	2	20150203	1	13,7	\$16	2,74	3/2/2015 00:00:00	6,81059648282826
49	14720	2	20130818	1	13,7	\$16	2,74	18/8/2013 00:00:00	19,7354096034542
49	14720	2	20131113	1	13,7	\$16	2,74	13/11/2013 00:00:00	0,99140057573095
49	14720	2	20130809	1	13,7	\$16	2,74	9/8/2013 00:00:00	14,7672315943055
49	14720	2	20130808	1	13,7	\$16	2,74	8/8/2013 00:00:00	28,5857292101718
49	14720	2	20131207	1	13,7	\$16	2,74	7/12/2013 00:00:00	23,793502474267
49	14720	2	20130807	1	13,7	\$16	2,74	7/8/2013 00:00:00	4,13449887419119
49	14720	2	20131220	1	13,7	\$16	2,74	20/12/2013 00:00:00	20,7766440021805
49	14720	2	20150116	1	13,7	\$16	2,74	16/1/2015 00:00:00	3,00093749770895
49	14720	2	20131005	1	13,7	\$16	2,74	5/10/2013 00:00:00	12,1491753053851
49	14720	2	20130727	1	13,7	\$16	2,74	27/7/2013 00:00:00	10,1073117577471
49	14720	2	20130920	1	13,7	\$16	2,74	20/9/2013 00:00:00	28,3802473638207
49	14720	2	20130806	1	13,7	\$16	2,74	6/8/2013 00:00:00	16,977695482783
49	14720	2	20130903	1	13,7	\$16	2,74	3/9/2013 00:00:00	18,9312432473525
49	14720	2	20131212	1	13,7	\$16	2,74	12/12/2013 00:00:00	19,6504394710064

Figura 21: Capa de Datos PowerBI

Fuente: imagen extraída de mi propio desarrollo.

### 3.10.1.4 Capa informe.

La capa Informe nos permite crear los reportes/vistas a partir de un conjunto de datos. Esta capa ofrece, además de lo mencionado anteriormente, crear medidas y aplicar transformaciones de datos utilizando Power Query.

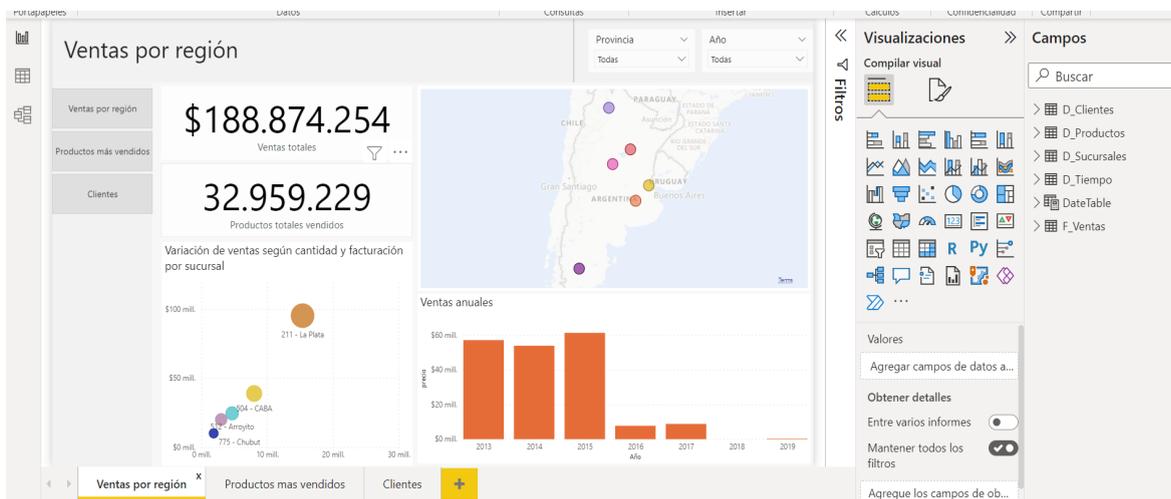


Figura 22: Capa de Informe PowerBI

Fuente: imagen extraída de mi propio desarrollo.

### 3.10.2 Estructura general de las vistas.

Para este caso de uso, se definió una estructura uniforme para poder visualizar las vistas de una manera amigable para el usuario final.

Esta estructura está compuesta por dos barras; una en el lateral izquierdo y otra en el borde superior, siendo estas dos cubiertas por el siguiente código de color: hex #F2F2F2 242 242 242

En el lateral izquierdo, se crearon 3 (tres) accesos directos que al hacer clic sobre alguno de ellos, llevan hacia otra vista.

El borde superior, separa el nombre del tablero y la cantidad de filtros (se explica su funcionamiento más adelante) mediante una barra divisoria.

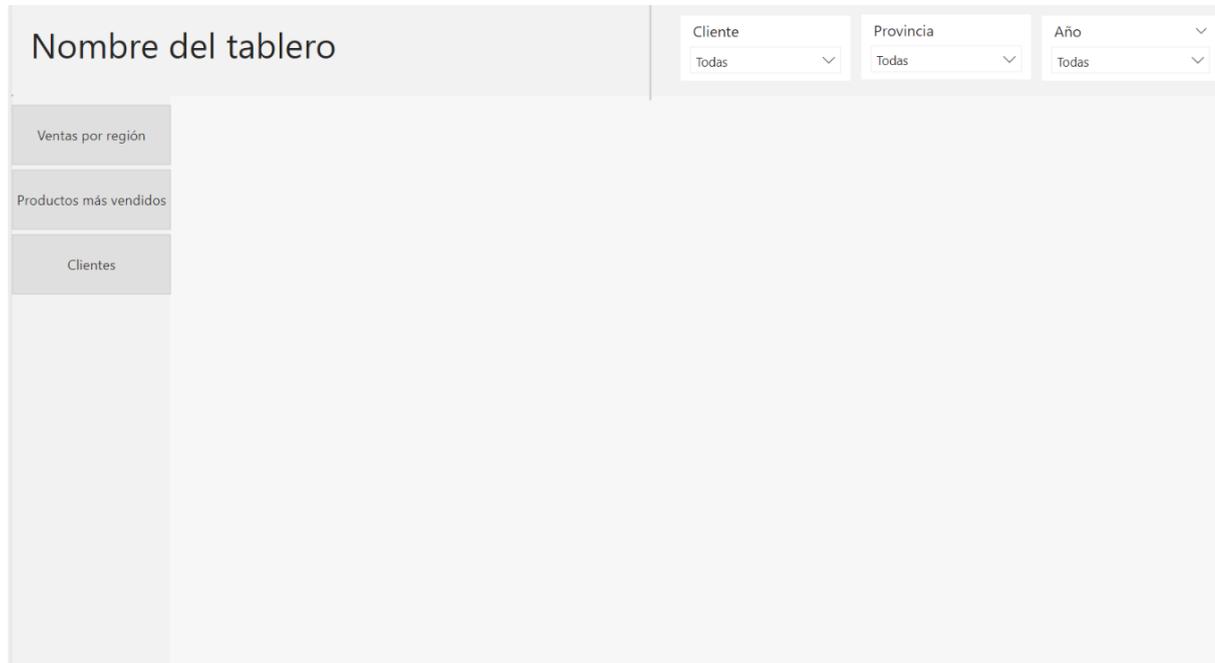


Figura 23: Mockup estructura de tableros

Fuente: imagen extraída de mi propio desarrollo.

### 3.10.3 Vista ventas por región

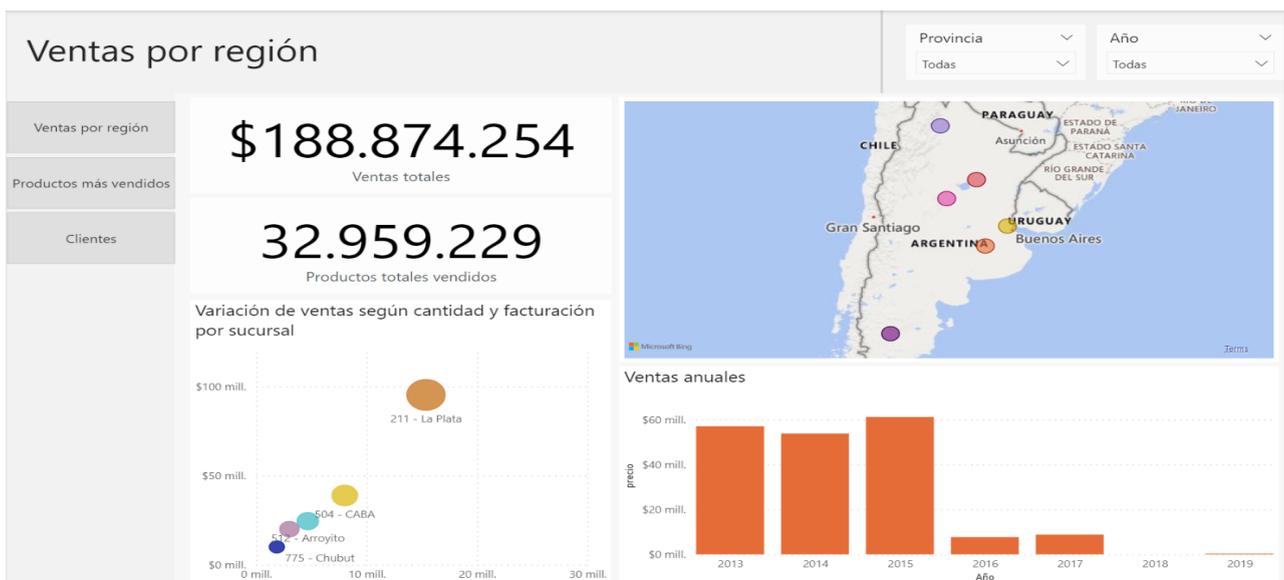


Figura 24: Vista ventas por región

Fuente: imagen extraída de mi propio desarrollo.

### 3.10.3.1 Gráfico Variación de ventas según cantidad y facturación por sucursal.

- Este gráfico se denomina gráfico de dispersión o, en inglés, scatterplot.
- El eje X, representa la cantidad de unidades vendidas y el eje Y la facturación total de las ventas.
- Cada círculo corresponde a una sucursal.
- El tamaño del círculo representa la facturación por sucursal.

Al posicionar el cursor sobre una sucursal, se despliega una leyenda donde se pueden ver los datos de la misma.

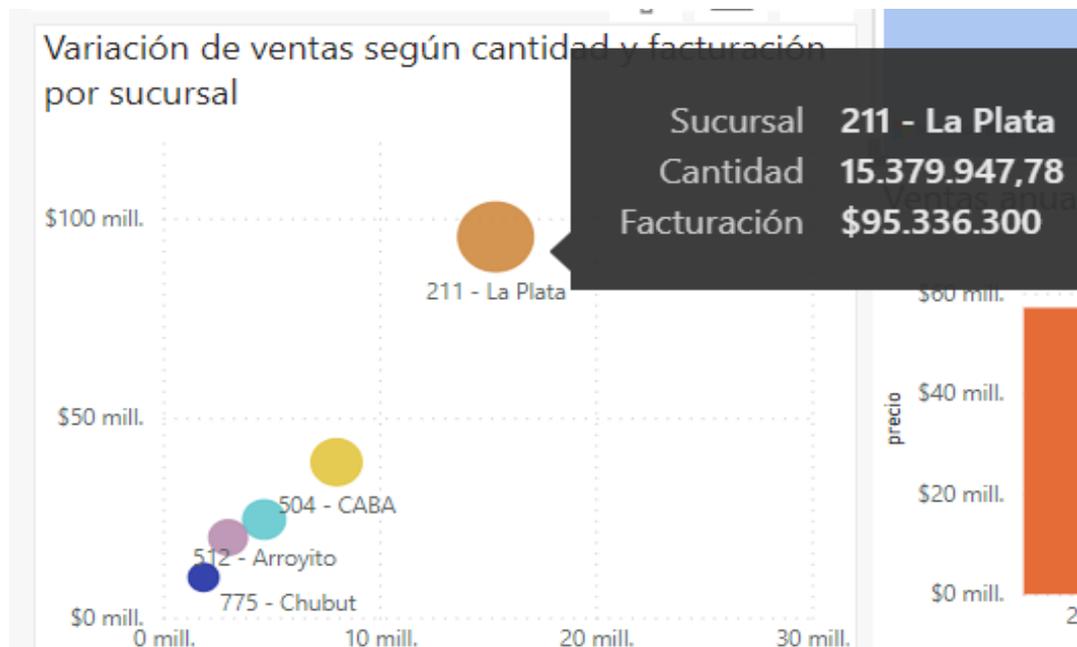


Figura 25: Leyenda del gráfico variación de ventas según cantidad y facturación por sucursal

Fuente: imagen extraída de mi propio desarrollo.

### 3.10.3.2 Gráfico Ventas anuales

Este gráfico es denominado como gráfico de barras.

- Su función es representar la facturación anual.
- En el eje X se encuentra el año mientras que en el eje Y la facturación.

### 3.10.3.3 Gráfico Mapa

Este gráfico muestra un mapa mundial donde se pueden ver las sucursales disponibles como puntos de colores.

**NOTA:** es importante establecer un estándar en la paleta de colores, ya que si no existe una uniformidad de colores, el usuario no va a poder asociar cada color con cada sucursal o lo que corresponda.

### 3.10.4 Vista Productos más vendidos



Figura 26: Vista Productos más vendidos

Fuente: imagen extraída de mi propio desarrollo.

#### 3.10.4.1 Gráfico TOP 5 Productos con mayor volumen de ganancias

Este gráfico es denominado como gráfico de tortas o en inglés, pie graph.

Su función es representar el porcentaje y la ganancia obtenida en base a la cantidad de productos vendidos.

Para hacer más representativo el análisis, se agregó una leyenda indicando de que lo que se está analizando son productos y una paleta de colores para poder diferenciarlos. Esta paleta de colores utiliza los mismos colores que en el tablero de ventas por región.

### 3.10.4.2 Gráfico Variación de ganancias anuales por marca

Este gráfico muestra un histórico de las ganancias anuales según la marca de producto. Este gráfico está pensado para mostrar un histórico, es por ello que no está sincronizado con los filtros "Provincia y Años" utilizados, al igual que en Ventas por región, también en este tablero.

### 3.10.4.3 Gráfico KPI Cantidad de productos vendidos

Este gráfico muestra la cantidad de productos vendidos en un año y la cantidad esperada que se planea vender para ese año. Este gráfico cuenta con un KPI que mide el grado de satisfacción de la cantidad de ventas que se espera alcanzar anualmente o en base a la suma de 2 o más años.

Si las ganancias superan o igualan al objetivo, entonces este se satisface y automáticamente, cambia a color verde indicando las cantidades y el porcentaje de satisfacción. Si se selecciona más de un año, hace la suma de los N años y lo compara con la suma de los M objetivos para ver si se satisface o no.

### 3.10.5 Vista Clientes

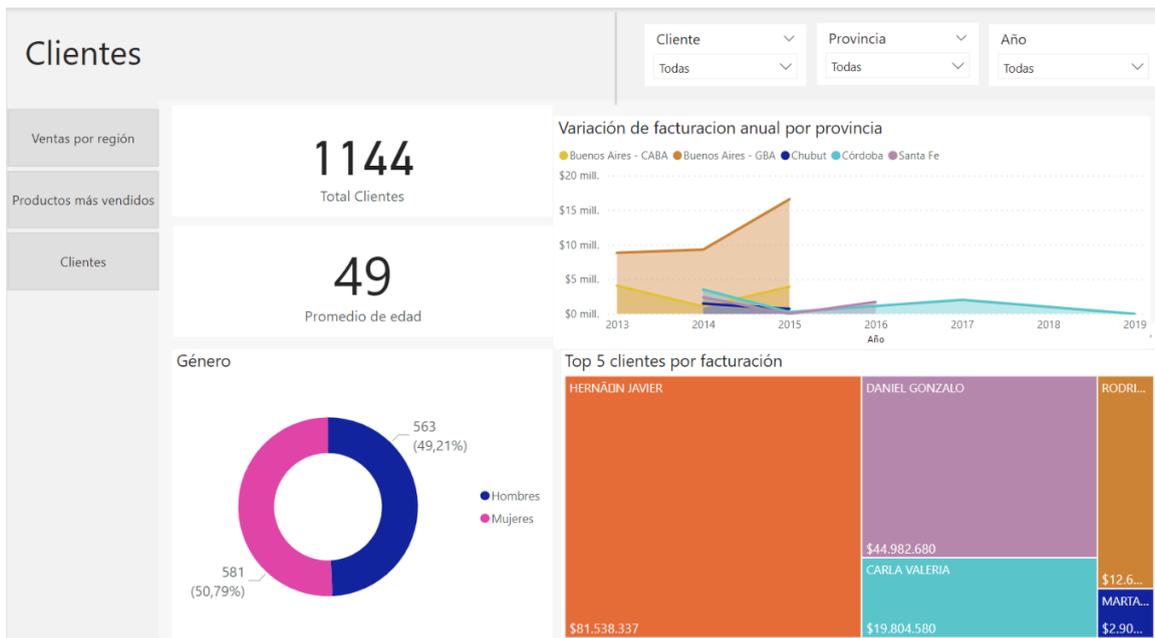


Figura 27: Vista Clientes

Fuente: imagen extraída de mi propio desarrollo.

### 3.10.5.1 Gráfico KPI Porcentaje de compradores por género.

Este KPI sirve para medir el porcentaje y cantidad de clientes cuyo género sea hombre o mujer.

### 3.10.5.2 Gráfico Top 5 clientes por facturación

Este gráfico muestra a los clientes con mayor facturación y los clasifica por región.

### 3.10.5.3 Gráfico Variación de facturación anual por provincia

Este gráfico muestra la cantidad de clientes que realizaron compras y la cantidad de ganancias anual por provincia.

### 3.10.6 Tarjetas utilizadas.

En este apartado se presentan las tarjetas con su descripción y su impacto en cada vista.

Tarjeta	Descripción	Impacta en vista
Ventas totales	Muestra en pantalla la cantidad de ventas totales por año.	3.10.3
Productos totales vendidos	Muestra en pantalla la cantidad de productos vendidos por año	3.10.3
Marca más vendida	Esta tarjeta muestra la marca de producto más vendida en un período de tiempo o en un histórico.	3.10.4
Producto más vendido	Esta tarjeta muestra el producto más vendido históricamente o por un período de tiempo.	3.10.4
Total Clientes	Muestra el total de los clientes que se hayan seleccionado	3.10.5
Promedio de edad	Muestra el promedio de edad de los clientes. Si se selecciona un solo cliente, muestra la edad del mismo.	3.10.5

Tabla 10: Tarjetas utilizadas en PowerBI.

### 3.10.7 Filtros utilizados

En este apartado se detallan los filtros, su funcionamiento e impacto en cada vista. Los filtros Provincia y año, están sincronizados con todos los gráficos y las tarjetas. Esto quiere decir que al seleccionar algún filtro y aplicarlo, los gráficos y tarjetas sufrirán un impacto, mientras que el filtro Clientes, solamente existe en la vista cliente y, por lo tanto, su impacto es únicamente para esa vista.

Filtro	Descripción	Impacta en vista	Sincronización con gráfico
Provincia	Permite seleccionar una o muchas provincias	3.10.3  3.10.4  3.10.5	3.10.3.1 3.10.3.3  3.10.4.1
Año	Permite filtrar datos por uno o más años. En caso de existir un año donde el negocio no estuvo operativo, aparecerá un cartel en pantalla indicando que no hubo registros de ventas para ese año.	3.10.3   3.10.4	3.10.3.1 3.10.3.2 3.10.3.3  3.10.4.1 3.10.4.2 3.10.4.3
Clientes	Permite seleccionar uno o más clientes.	3.10.5	3.10.5.1 3.10.5.2

Tabla 11: Filtros utilizados en PowerBI

### 3.10.8 Medidas utilizadas

En este apartado se detalla una tabla con las medidas utilizadas en PowerBI. Para elaborar las métricas en PowerBI, se hizo uso del lenguaje de programación DAX [9].

ID	Nombre de la medida	Fórmula	Impacta en vista
1	Facturación	Sum(F_Ventas[facturacion])	3.10.3 3.10.4 3.10.5
2	Productos totales vendidos	Sum(F_Ventas[cantidad])	3.10.3
3	Comprobar cantidad vendida anual	if(sum(F_Ventas[cantidad])>0,"", "N o hay registros de ventas este año")	3.10.4
4	Total Clientes	COUNTROWS(D_Clientes)	3.10.5
5	Total Hombres	CALCULATE([Total Clientes], D_Clientes[genero] = "H")	3.10.5
6	Total Mujeres	CALCULATE([Total Clientes], D_Clientes[genero] = "M")	3.10.5

Tabla 12: Medidas utilizadas en PowerBI

### 3.10.9 Explicación de las medidas.

A continuación se explica el funcionamiento de cada una de las medidas abordadas para este trabajo final de carrera.<sup>5</sup>

1: Representa la suma de las ventas totales de la Fact ventas.

2: Suma la cantidad de productos vendidos y muestra el total.

3: Evalúa si la suma de la cantidad de productos vendidos es mayor que cero. Si la condición es cierta, devuelve la cantidad total de productos vendidos. Si la condición es falsa, devuelve un texto indicando que no hubo registros de ventas en el año en cuestión. Esta medida se sincroniza con el filtro "Año".

4: Cuenta la cantidad de filas en la tabla D\_Clientes y devuelve el total de filas.

5: Devuelve el total de clientes cuyo género sea hombre.

6: Devuelve el total de clientes cuyo género sea mujer.

### 4.0 Proceso de evaluación ágil Dashboard de ventas - MyFeps.

En esta fase se va a evaluar, como se detalló en el apartado 1.5.1, las características: usabilidad objetiva, grado de satisfacción del usuario del tablero y correctitud mostrado anteriormente. Es necesario introducir al lector en la materia. Es por ello que se procederá a desarrollar la evaluación ágil paso a paso.

Proceso Ágil	Id viejo	Nombre	Predecesor Ágil
1	2	Establecer el Propósito de la Evaluación	
2	3	Identificar el Producto a Evaluar	1
3	4	Identificar Los Requerimientos de Calidad (Car, SubCar, Métricas)	2
4	14	Definir Recursos y Equipo para la Evaluación	3
5	17	Establecer Ponderación	3

<sup>5</sup> El número de la medida representa el ID de la tabla de medidas descrita en el apartado 3.10.8

6	21	Preparar Recursos e Infraestructura	4
7	24	Elaborar casos de Prueba	6
8	26	Obtener y Adecuar Recursos para la Evaluación	7
9	30	Realizar las Pruebas	8
10	35	Medir los Atributos requeridos por las Métricas	5; 10
11	39	Analizar y Concluir el Grado de Calidad	10

Tabla 13: Modelo de evaluación ágil MyFEPS

#### 4.1 Establecer el propósito de la evaluación.

En este trabajo final de carrera, se buscará evaluar, mediante el uso de métricas, el producto de software “Dashboard de ventas” con el objetivo de poder determinar el grado de calidad del software.

En el Business Intelligence y en el análisis de datos, es indispensable:

- 1- Contar con la calidad del artefacto confiable a través de la característica correctitud.
- 2- contar con dashboards que sean fáciles de utilizar a través de la característica usabilidad objetiva para la mayoría de los usuarios a la hora de interactuar con ellos.
- 3- Alcanzar un grado de satisfacción de los usuarios.

#### 4.2 Identificar el producto a evaluar

El producto de software a evaluar es denominado como “Dashboard Ventas”. Este producto de software fue desarrollado, con la herramienta Power BI Desktop, con el propósito de brindar un análisis descriptivo de datos y resultar de fácil uso para los usuarios.

### **4.3 Identificar requerimientos de calidad**

En este apartado se va a dividir en dos partes:

- 1 - Definir los stakeholders que van a ser partícipes del proyecto.
- 2 - Definir y desarrollar los atributos y métricas. Esto es muy importante, ya que se deberá seleccionar las métricas con sus características y subcaracterísticas con el objetivo de cubrir todos los requisitos de calidad del software de la evaluación.

#### **4.3.1 Definición de los Stakeholders**

En este apartado, tal como se detalla en el framework MyFEPS apartado 4.3 - ítem 1. Se definirán los stakeholders que participarán a la hora de evaluar al producto de software. Para realizar estas pruebas, se decidió tomar una muestra de 5 usuarios. Este criterio, está basado en lo expuesto por Jakob Nielsen y Tom Landauer: “Las pruebas de usabilidad elaboradas son un desperdicio de recursos. Los mejores resultados provienen de probar no más de 5 usuarios y ejecutar tantas pruebas pequeñas como pueda pagar.” [12]. Este criterio, está basado en una fórmula que muestra la cantidad de problemas de usabilidad encontrados en una prueba de usabilidad con N usuarios:

$$N(1-(1-L)^n)$$

“donde  $N$  es el número total de problemas de usabilidad en el diseño y  $L$  es la proporción de problemas de usabilidad descubiertos al probar un solo usuario. El valor típico de  $L$  es 31 %, promediado en una gran cantidad de proyectos que estudiamos. Trazar la curva para  $L = 31\%$  da el siguiente resultado” [12]

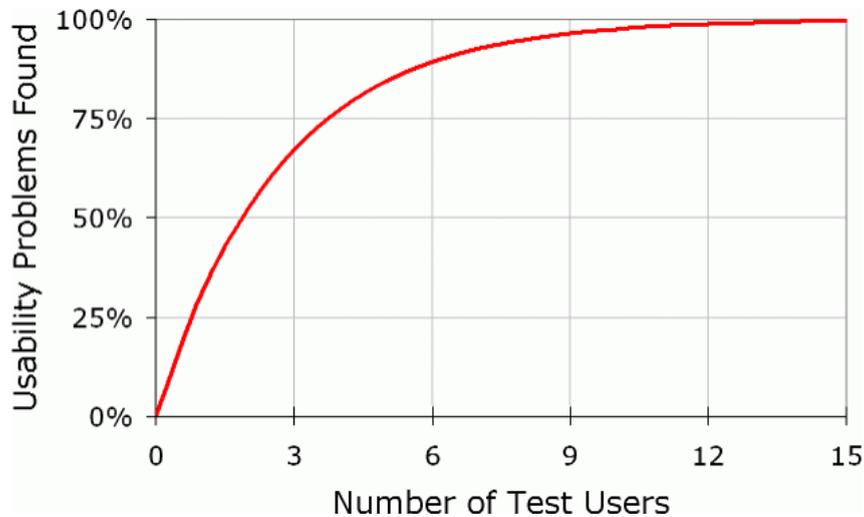


Figura 28: Curva de usabilidad de Jakob Nielsen

Fuente: <https://media.nngroup.com/media/editor/alertbox/20000319-user-testing-diminshing-returms-curve.gif>

Nielsen explica, según la figura 28 que, a medida que se incremente el número de usuarios, hay una mayor probabilidad de encontrar problemas de usabilidad, dado que si se excede la cantidad de cinco usuarios, el observador va a notar comportamientos similares entre los usuarios; patrones de comportamiento equivalentes, errores al ejecutar ciertas funcionalidades, etc. La gráfica muestra que, si se toma una muestra de quince usuarios, la probabilidad de encontrar problemas de usabilidad, es del 100% y, además, al contratar más usuarios, implica un mayor gasto presupuestario. En cambio, si la prueba se hace con cinco usuarios, en la primera iteración, la probabilidad de encontrar problemas de usabilidad, es del 85%, lo que da margen para corregir los problemas de usabilidad encontrados en la primera iteración para luego, corregirlos y pasar a la segunda iteración. “El segundo estudio con 5 usuarios descubrirá la mayor parte del 15 % restante de los problemas de usabilidad originales que no se encontraron en la primera ronda de pruebas. (Todavía quedará el 2 % de los problemas originales; tendrán que esperar hasta que se identifique el tercer estudio).” [12].

A continuación, se define el grupo de Stakeholders:

- Mauro Fibonacci: director general de marketbris. Este actor, es el principal stakeholder que se encargará de responder el cuestionario.

- Diego Lezcano: Gerente de ventas. De gran relevancia a la hora de evaluar los análisis de las ventas, ya que cuenta con toda la información del negocio.
- María Diorio: Analista de Marketing JR. Este actor está encargado de armar las campañas de ventas.
- Araceli Letroye: Analista de datos. Este sujeto es ajeno a la organización, pero tiene un fuerte entendimiento de la materia.
- Darío Nieves: Analista Comercial SR: Es el encargado de elaborar reportes para clientes e informar a la gerencia general.

#### 4.3.2 Definición de las características, subcaracterísticas, métricas y atributos.

Una vez definidos los stakeholders que van a participar en la evaluación de software, se procederá a definir las características y subcaracterísticas con sus métricas y atributos correspondientes [13].

##### Característica Básica: Correctitud (3.4)

- ✓ Sub-característica 3.4.1: Correctitud de datos.
  - Atributo 04.1.1.A: El número de errores en los datos detectados en uso en un periodo dado.
    - o Métrica: en un período preestablecido se colectan:
      - a. Nro de datos erróneos provistos por el sistema: NDE
      - b. Nro de datos no erróneos provistos por el sistema: NDNE
$$M = 1 - \frac{NDE}{(NDE + NDNE)}$$
- ✓ Sub-característica 3.4.2: Correctitud de procesos.
  - Atributo 04.2.1.A: El número de errores en los datos detectados en uso en un periodo dado.
    - o Métrica: en un período preestablecido se colectan:
      - a. Nro de datos erróneos provistos por el sistema: NPE
      - b. Nro de procesos no erróneos provistos por el sistema: NNE
$$M = 1 - \frac{NPE}{(NPE + NNE)}$$

**NOTA:** Se actualizaron las fórmulas de las métricas en los atributos 04.2.1.A y 04.2.1.A. El motivo de esta actualización, se debe a que al no estar colocado el siguiente término [ 1 - ... ] siendo 1 el 100%, - la diferencia y los puntos ... la fórmula en su estado original, entonces el grado de calidad para ese atributo va a tender siempre a 0. Por ejemplo, si NPE = 2 y NNE = 24, entonces  $M = 2/26$  y por lo tanto se obtiene un GCAT de 0.07 que sería inaceptable. Es por eso que se decidió agregar dicho término, siendo este el total menos la diferencia entre NPE/NPE + NNE.

Característica Básica: Grado de satisfacción del usuario (3.13)

- ✓ Sub-característica 3.13.2: En el Acceso a las funciones.
  - Atributo 13.2.1.A Facilidad de Acceso a las Funciones percibido en entorno de uso.
    - Métrica: se realizarán entrevistas a los usuarios consultándoles lo accesible que le resultó el sistema. Los mismos responderán con alguno de los siguientes valores: 0, 0.25, 0.50, 0.75 y 1, siendo 0 el valor donde el usuario expresó que le fue difícil acceder a las funciones y 1 el valor donde el usuario expresó que le fue fácil acceder a las funciones. Finalmente se obtendrá un valor promedio del grado de facilidad del acceso a las funciones (GFAi según nomenclatura MyFEPS) percibido por el usuario con la siguiente fórmula:

$$M = \frac{SUM(GFAi)}{Cant. Usuarios}$$

- ✓ Sub-característica 13.3.3: En la Comprensión de las salidas del sistema.
  - Atributo 13.3.1.A: Grado de comprensión de las salidas del sistema percibido en entorno de uso.
    - Métrica: se realizarán entrevistas a los usuarios consultándoles si comprendieron las salidas mostradas por el sistema. Los mismos responderán con alguno de los siguientes valores: 0, 0.25, 0.50, 0.75 y 1, siendo 0 el valor donde el usuario expresó que no comprendió las salidas y 1 el valor donde el usuario expresó que comprendió las salidas. Finalmente se obtendrá un valor promedio del grado de comprensión de las salidas del sistema (GCSi según

nomenclatura MyFEPS) percibido por el usuario con la siguiente fórmula:

$$M = \frac{SUM(GSi)}{Cant. Usuarios}$$

- ✓ Sub-característica 3.13.6: Satisfacción total.
  - Atributo 13.6.1.A: Grado de satisfacción percibido en entorno de uso.
    - Métrica: se realizarán entrevistas a los usuarios consultándoles cuán satisfactorio fue usar el sistema. Los mismos responderán con alguno de los siguientes valores: 0, 0.25, 0.50, 0.75 y 1, siendo 0 el valor donde el usuario expresó que no fue satisfactorio y 1 el valor donde el usuario expresó que fue satisfactorio. Finalmente se obtendrá un valor promedio del grado de conocimientos del sistema (GSi según nomenclatura MyFEPS) percibido por el usuario con la siguiente fórmula:

$$M = \frac{SUM(GSi)}{Cant. Usuarios}$$

Característica Básica: Usabilidad objetiva (3.17)

- ✓ Sub-característica 3.17.1: En el aprendizaje
  - Atributo 17.1.4.U: Tiempo promedio de aprendizaje intuitivo.
    - Métrica: se realizarán una lista de tareas organizacionales (LTOE nomenclatura MyFEPS) a ejecutar y definiendo el tiempo óptimo en que deberían ejecutarse (TiLTOE en la nomenclatura de MyFEPS), con estos parámetros se tomará el tiempo en que tarda c/usuario en ejecutar las tareas. El valor final de la métrica quedará definido por el siguiente cálculo:

*Si Media < TiLTOE entonces Valoración = 1*

*Sino Valoración =  $\frac{TiLTOE}{Media}$*

- ✓ Sub-característica 3.17.3: En la entrada manual de información.
  - Atributo 17.3.3.U: Ayuda para la prevención de errores en el ingreso de manual de información.

- o Métrica: para calcular esta métrica se evaluará, a cada usuario, la cantidad de campos de datos que logran completar con éxito (TACE nomenclatura MyFEPS) y la cantidad de campos de datos que no logran completar con éxito (nTACE en la nomenclatura de MyFEPS), obteniendo una sumatoria de ambas variables. El valor final de la métrica quedará definido por el siguiente cálculo:

$$Valoración = \frac{Sum(TACE)}{nSum(TACE)+Sum(TACE)}$$

- ✓ Sub-característica 3.17.4: En el acceso a las funciones.

- Atributo 17.4.1: Velocidad en el acceso a las funciones.

- o Métrica: se calculará definiendo todas las funciones implementadas (NFTot nomenclatura MyFEPS) y el tiempo promedio de óptimo de ejecución de ellas (MLTF nomenclatura MyFEPS). Luego se obtiene el tiempo promedio de ejecución real (MediaNFI en la nomenclatura de MyFEPS). Por último, el valor de la métrica se obtendrá de la siguiente fórmula:

$$Valoración = \frac{MLTF}{MediaNFI}$$

- ✓ Sub-característica 3.17.6: En el conocimiento del sistema en su totalidad.

- Atributo 17.6.1.U: Evaluación promedio de usuarios en lo relativo al grado de conocimiento de las operaciones del sistema.

- o Métrica: se tomará una evaluación objetivo a los usuarios teniendo en cuenta el total de operaciones conocibles del sistema (TOS nomenclatura MyFEPS), se realizará una sumatoria de las notas (SumT nomenclatura MyFEPS) obtenidas durante la ejecución del sistema. Por último, el valor de la métrica se obtendrá de la siguiente fórmula:

$$Valoración = \frac{\frac{SumT}{cant. usuarios}}{100}$$

#### 4.4 Definir recursos y equipo para la evaluación.

En este apartado se definen los recursos, herramientas y el grupo evaluador para poder llevar adelante la evaluación.

Usuarios finales: los actores (stakeholders) que interactuaron con el sistema con sus roles definidos en el apartado 4.3.1

Infraestructura: Plataforma de sistema operativo – Microsoft Windows 10 Home 21H1

Herramientas tecnológicas: Una notebook con mouse y teclado integrados.

Software: Microsoft Power BI Desktop (Free versión)

Hardware: Intel(R) Core (TM) i5-10210U CPU @ 1.60GHz 2.11 GHz

El grupo evaluador está conformado por los siguientes actores:

Rol o especialidad	Cantidad de recursos	Tareas	Recurso
Analista de QA.	1	<ul style="list-style-type: none"> <li>- Preparar los casos de pruebas.</li> <li>- Elaborar la documentación de los casos de pruebas.</li> </ul>	Ignacio Sacha Duplaa
Ingeniero de software.	1	<ul style="list-style-type: none"> <li>- Documentar el proceso de evaluación.</li> <li>- Elegir y definir las métricas.</li> <li>- Llevar un registro de los resultados obtenidos en la encuesta.</li> <li>- Analizar resultados.</li> </ul>	Ignacio Sacha Duplaa
Project manager (PM).	1	<ul style="list-style-type: none"> <li>- Organiza al equipo evaluador y a los recursos humanos.</li> </ul>	Ignacio Sacha Duplaa
Tester.	1	<ul style="list-style-type: none"> <li>- Gestionar el armado de los ambientes de prueba.</li> <li>- Ejecuta los casos de prueba.</li> </ul>	Ignacio Sacha Duplaa

Tabla 14: Grupo evaluador.

#### 4.5 Establecer ponderación

En esta etapa, se definen los criterios de aprobación para dicha evaluación, con el fin de determinar si la evaluación fue exitosa o no en función del valor alcanzado por cada métrica definida en el ítem 4.3 apartado 2, el grado de calidad de Subcaracterísticas, el grado de calidad de características y el grado de calidad que obtuvo el producto de software.

Se detalla la escala de valores con su valor definible y peso asociado:

ID encuesta	Valor Definible	Peso Asociado
4	Muy importante	1
3	Importante	0.75
2	Medianamente importante	0.50
1	Algo importante	0.25
0	Totalmente sin importancia	0

Tabla 15: Escala de ponderación MyFEPS Ágil.

A continuación, se muestra una tabla con los resultados obtenidos de los cuestionarios de la ponderación de pesos que completaron los stakeholders definidos en el apartado 4.3.1

Características	Puntuación Total	StakeHolders Media
<b>3.4 Correctitud</b>	<b>4,50</b>	<b>0,90</b>
<b>3.4.1 Correctitud de datos</b>	<b>4,50</b>	<b>0,90</b>
04.1.1.A Número de errores en los datos detectados en un período dado	<b>4,00</b>	<b>0,80</b>
<b>3.4.2 Correctitud de procesos</b>	<b>4,00</b>	<b>0,80</b>
04.1.2.A Número de errores en los datos detectados en un período dado	<b>3,50</b>	<b>0,70</b>
<b>3.13 Satisfacción de los usuarios</b>	<b>4,50</b>	<b>0,90</b>
<b>3.13.2 En el acceso a las funciones</b>	<b>4,25</b>	<b>0,85</b>
13.2.1.A Facilidad de Acceso a las Funciones percibido en entorno de uso	<b>3,75</b>	<b>0,75</b>
<b>3.13.3 En la Comprensión de las salidas del sistema</b>	<b>4,00</b>	<b>0,80</b>
13.3.1.A Grado de comprensión de las salidas del sistema percibido en entorno de uso	<b>4,00</b>	<b>0,80</b>
<b>3.13.6. Satisfacción total</b>	<b>4,25</b>	<b>0,85</b>
13.6.1.A Grado de Satisfacción percibido en entorno de uso	<b>4,00</b>	<b>0,80</b>

<b>3.17 Usabilidad Objetiva</b>	<b>4,75</b>	<b>0,95</b>
<b>3.17.1 En el Aprendizaje</b>	<b>4,75</b>	<b>0,95</b>
17.1.4 Tiempo promedio de Aprendizaje intuitivo	<b>4,75</b>	<b>0,95</b>
<b>3.17.3 En la Entrada Manual de Información</b>	<b>3,75</b>	<b>0,75</b>
3.17.3.3.U Ayuda para la Prevención de Errores en el ingreso manual de Información	<b>3,50</b>	<b>0,70</b>
<b>3.17.3 En el Acceso a las Funciones</b>	<b>4,25</b>	<b>0,85</b>
17.4.1.U Velocidad en el Acceso a las Funciones	<b>4,25</b>	<b>0,85</b>
<b>3.17.6 En el conocimiento del sistema en su totalidad</b>	<b>4,25</b>	<b>0,85</b>
17.6.1.U Evaluación Promedio de usuarios en lo relativo al grado de conocimiento de las Operaciones del Sistema	<b>4,00</b>	<b>0,80</b>

Tabla 16: Resultado Cuestionario pesos MyFEPS Ágil.

#### 4.6 Preparar recursos e infraestructura.

En este apartado se definen los 5 (cinco) recursos que van a realizar las pruebas. Estos sujetos, poseen un dominio de tecnología, clasificado en una escala de: nulo, bajo, básico, intermedio y avanzado. Este grupo de recursos realizarán los casos de prueba especificados posteriormente en el ítem 4.7.1. con la infraestructura suministrada por Ignacio Sacha Duplaa.

Nombre	Apellido	Dominio tecnológico
Guillermo	Ferrari	intermedio
Monica	Istariz	Básico
Julian	Bedoya	intermedio
Manuel	De la vega	Avanzado
Maximiliano	Calderón	Básico

Tabla 17: Grupo de usuarios finales.

#### 4.7 Elaborar casos de prueba

En este apartado se definen los casos de prueba que van a ser otorgados a los usuarios finales para que puedan completar la evaluación.

En primera instancia, se elaborará la especificación de la prueba que describe genéricamente la prueba a realizar y luego se detallan los casos de prueba a tomando en cuenta las métricas para determinar los resultados de la evaluación:

- NDE: Número de datos erróneos provistos por el sistema.
- NDNE: Número de datos no erróneos provistos por el sistema.
- NPE: El número de procesos erróneos provistos por el sistema.
- NNE: El número de procesos no erróneos provistos por el sistema.
- LTOE: define las tareas organizacionales que deben llevar a cabo los usuarios evaluados a través del Sistema.
- TiLTOE: Tiempos óptimos esperados de ejecución de tareas.
- nTACE: Número de controles que el usuario no completa con éxito.
- NFTot: Total de funciones implementadas en el sistema.
- GCAT: Valor de la medición del atributo que determina el grado de calidad.

#### **4.7.1 Plan de pruebas.**

1. Abrir reporte de ventas.
2. Diferenciar gráficos de tarjetas.
3. Interactuar con los botones de acción que llevan a las distintas hojas de cada tablero.
4. Filtrar las ventas que se realizaron en Buenos Aires entre el año 2013 a 2015.
5. Obtener género del cliente, total de clientes y promedio de edad.
6. Obtener los cinco productos con más ventas en la Santa Fe del año 2014
7. Obtener el total de productos vendidos del KPI "Cantidad de productos vendidos"
8. Identificar a los 5 clientes con mayor facturación históricamente.

#### **4.7.2 Especificación de la evaluación.**

En este apartado, se especificarán y se elaborarán los casos de prueba desarrollados en el apartado 4.7.1

Caso de prueba	Especificación de la prueba	Descripción del caso de prueba
1. Abrir el reporte de ventas.	El usuario selecciona el ícono de PowerBI para ejecutar el programa. Luego, elige el reporte de ventas y lo selecciona.	1- El usuario se posiciona sobre el ícono de PowerBI en el escritorio y luego hace doble click sobre el ícono de la aplicación. 2- Una vez que la aplicación haya terminado de cargar, aparecerá una pantalla donde el usuario podrá visualizar un listado de reportes disponibles. 3- El usuario tendrá que hacer click sobre el reporte "Reporte de ventas.pbix".  <b>Resultado esperado:</b> Que el usuario pueda abrir el reporte de ventas.
2. Diferenciar gráficos de tarjetas	El usuario selecciona una tarjeta y en la sección visualizaciones debe aparecer el ícono correspondiente a la tarjeta. El usuario selecciona un gráfico y en la sección de visualizaciones debe aparecer el ícono correspondiente a un gráfico.	1- El usuario se posiciona en la solapa Ventas por región y luego selecciona alguna de las dos tarjetas: Ventas o Productos totales vendidos. 2- Una vez que el usuario haya seleccionado alguna tarjeta, debe poder visualizar en la sección "Visualizaciones" que aparezca sombreado el ícono "tarjeta". 3- Posteriormente, selecciona alguno de los tres gráficos: variación de ventas, ventas anuales o mapa y debe observar que en la sección "visualizaciones" aparezca sombreado el ícono del gráfico que haya seleccionado.  <b>Resultado esperado:</b> que el usuario pueda identificar los elementos que sean gráficos y los elementos que sean tarjetas.
3. Interactuar con los botones de acción que llevan a las distintas hojas de cada tablero.	El usuario selecciona un botón de acción del menú lateral izquierdo y este debe ser re dirigido hacia otra solapa, en caso de elegir una solapa distinta a la actual.	1- El usuario elige cualquiera de las 3 solapas disponibles y deberá posicionarse sobre el menú de acción. 2- Luego deberá mantener presionada la tecla Ctrl y, mientras la tecla está presionada, el usuario deberá hacer click sobre un elemento dentro del menú de acción para que pueda ser re dirigido hacia otra pestaña  <b>Resultado esperado:</b> que el usuario sea redirigido hacia otra pestaña haciendo uso del menú de acción.

<p>4. Filtrar las ventas que se realizaron en Buenos Aires entre el año 2013 a 2016.</p>	<p>El usuario selecciona el filtro Provincia y elige Buenos aires, luego selecciona el filtro Año y elige un rango de fechas.</p>	<p>1- El usuario selecciona la solapa "Ventas por región" y luego selecciona la barra de despliegue del filtro. Por defecto, en la barra dice "Todas" haciendo referencia a que todas las provincias se encuentran seleccionadas por defecto. 2- El usuario deberá hacer click sobre "Todas" para que no quede seleccionada ninguna. 3- Finalmente, el usuario deberá elegir "Buenos Aires" y deberá aplicar el mismo funcionamiento para filtro de año pero, a diferencia que, para seleccionar dos o más fechas, es necesario que el usuario mantenga la tecla Ctrl mientras hace click sobre los años 2013 y 2016.</p> <p><b>Resultado esperado:</b> El usuario pueda realizar un filtrado de ventas por año y región y pueda visualizar los el impacto generado en la vista.</p>
<p>5. Obtener género del cliente, total de clientes y promedio de edad.</p>	<p>El usuario selecciona uno o más clientes y deberá poder visualizar su género, edad y, si selecciona más de un cliente, el promedio de edad.</p>	<p>1- El usuario se posiciona en la solapa "Clientes" y se posiciona con el cursor sobre el filtro Cliente. 2- Posteriormente, este deberá seleccionar la ventana desplegable del filtro, que por defecto está en "Todas", y seleccionar un cliente aleatorio. 3- Luego, manteniendo presionada la tecla Ctrl, el usuario deberá elegir cuatro clientes más de manera aleatoria y deberá poder visualizar lo siguiente: género del cliente, edad/promedio edad y total de clientes.</p> <p><b>Resultado esperado:</b> Obtener género de un cliente, total de clientes y promedio de edad.</p>
<p>6. Obtener los cinco productos con más ventas en la Santa Fe del año 2014</p>	<p>El usuario selecciona el filtro provincia, eligiendo como provincia Santa Fe y luego seleccionando el año 2014 en el filtro Año.</p>	<p>1- El usuario se posiciona sobre la solapa "Productos más vendidos" y luego sobre el filtro "Provincia". 2- Una vez posicionado sobre el filtro provincia, este hará click sobre el botón de la flecha hacia abajo y tendrá que seleccionar "Santa Fe". 3- Posteriormente, el usuario deberá repetir el mismo procedimiento para el filtro año, eligiendo como fecha 2014.</p>

		<b>Resultado esperado:</b> El usuario pueda identificar los 5 Productos con mayor volumen de ganancias en santa fe, año 2014.
7. Obtener el total de productos vendidos del KPI "Cantidad de productos vendidos"	El usuario selecciona el filtro año y elige un año o un rango de años (siendo este consecutivo o no consecutivo) y deberá identificar la cantidad de productos vendidos y la cantidad vendida esperada para esa fecha o ese rango de fechas.	1- El usuario se posiciona sobre la solapa "Productos más vendidos" y luego sobre el filtro "Año". 2- En segundo lugar, debe seleccionar un año o un rango de años. 3- El usuario deberá visualizar un número (en el caso de seleccionar un solo año) y un área (en caso de seleccionar un rango de fechas) pintada de rojo o verde  <b>Resultado esperado:</b> Obtener un valor numérico pintado de verde o rojo sobre el KPI Cantidad de productos vendidos.
8. Identificar a los 5 clientes con mayor facturación históricamente.	El usuario selecciona la solapa de Clientes y elige todos los clientes, provincias y años.	1- El usuario se posiciona sobre la solapa "Clientes". 2- El usuario tendrá que seleccionar la opción "Todas" en los siguientes filtros: Cliente, Provincia y Año. 3- Finalmente, el usuario deberá poder visualizar cuadrados con diferente volumen dentro del gráfico "TOP 5 Clientes por facturación".  <b>Resultado esperado:</b> visualizar los 5 clientes que más facturación le generaron a la empresa.

Tabla 18: Especificación de los casos de prueba.

#### 4.8 Obtener y adecuar recursos para la evaluación.

En este apartado se consultó la disponibilidad de horarios para los recursos y se organizó un cronograma con las actividades de la evaluación que van a ser realizadas por estos mismos.

#### 4.9 Realizar las pruebas

En este apartado, del sistema (actores) van a realizar la ejecución de los casos de prueba. Estos mismos, se encuentran detallados en el ítem 4.7. Finalmente, por cada usuario realizó los casos de prueba indicados, se le presentó el cuestionario para que pueda completarlo, además, si algún usuario por algún motivo no podía completar el cuestionario, como vía

alternativa, el mismo iba a ser enviado a su casilla de email para que lo complete posteriormente.

**Correctitud**

(3.4.1) Correctitud de datos

(código 04.1.1.A) Número de errores en los datos detectados en un período dado

Caso de prueba	NDE	NDNE
1	0	2
2	0	1
3	0	4
4	0	3
5	0	2
6	0	4
7	0	1
8	0	1

Tabla 19: MyFEPS Ágil. Correctitud de datos - 04.1.1.A

Casos de prueba evaluados: 8

$$\text{SUM(NDE)} = 0$$

$$\text{SUM(NDNE)} = (2+1+4+3+2+4+1+1) = 18$$

$$M = 1 - \frac{0}{(0 + 18)}$$

$$M = 1$$

$$\text{GCAT} = 1$$

(3.4.2) Correctitud de procesos

(código 04.2.1.A) Número de errores en los procesos detectados en uso en un período dado.

NPE: Número de procesos erróneos provistos por el sistema

NNE: Número de procesos no erróneos provistos por el sistema

Caso de prueba	NPE	NNE
1	1	3
2	0	3
3	0	2
4	0	3
5	0	3
6	0	4
7	1	3
8	0	3

Tabla 20: MyFEPS Ágil. Correctitud de procesos - 04.2.1.A

Casos de prueba evaluados: 8

$$\text{SUM(NPE)} = (1+0+0+0+0+0+1+0) = 2$$

$$\text{SUM(NNE)} = (3+3+2+3+3+4+3+3) = 24$$

$$M = 1 - \frac{2}{(2+24)}$$

$$M = 1 - 0.07$$

$$M = 0.93$$

$$\text{GCAT} = 0.93$$

### Usabilidad objetiva

(3.17.1) En el Aprendizaje

(código 17.1.4) Tiempo promedio de Aprendizaje intuitivo

**NOTA:** el tiempo (expresado en minutos) que se detalla en el resultado de la tabla, corresponde a todos los casos de prueba que fueron ejecutados en el apartado 4.7.2

Usuario	Resultado (en minutos)
Usuario 1	19,12
Usuario 2	12,56
Usuario 3	13,55

Usuario 4	16,57
Usuario 5	14,12

Tabla 21: MyFEPS Ágil. Usabilidad objetiva. En el aprendizaje 17.1.4

LTOE = 5

TiLTOE = 20 (es el tiempo óptimo que se considera para esta prueba)

Suma = (19,12 + 12,56 + 13,55 + 16,57 + 12,12) = 75,92

Media = (75.92 / 5) = 15,18

De acuerdo con la primera condición para obtener la valoración:

*Si Media < TiLTOE entonces Valoración = 1*

Evaluamos: 15,18 < 20. Como la condición es verdadera, entonces la valoración es de 1

Valoración = 1

GCAT = 1

(3.17.3) En la entrada manual de la información

(código 3.17.3.3.U) Ayuda para la Prevención de Errores en el ingreso manual de Información

Usuario	TACE	nTACE
Usuario 1	3	0
Usuario 2	5	1
Usuario 3	2	2
Usuario 4	5	2
Usuario 5	4	1

Tabla 22: MyFEPS Ágil. Ayuda para la Prevención de errores en el ingreso manual de información 17.3.3.U

SUM(TACE) = (3+5+2+5+4) = 19

SUM(nTACE) = (0+1+2+2+1) = 6

Para obtener la valoración, se utiliza la siguiente fórmula descrita en el apartado 4.3

$$Valoración = \frac{Sum(TACE)}{nSum(TACE)+Sum(TACE)}$$

$$Valoración = \frac{19}{(6+19)}$$

$$Valoración = 0,76$$

$$GCAT = 0,76$$

(3.17.3) En el Acceso a las Funciones

(código 17.4.1.U) Velocidad en el Acceso a las Funciones

Usuario	Tiempo promedio (MLFT)	Tiempo real (MediaNFI)
Función 1	1	2
Función 2	1	3
Función 3	1	4
Función 4	2	2
Función 5	1	3
Función 6	1	1
Función 7	2	2
Función 8	3	2
Función 9	1	3
Función 10	1	3
Función 11	1	2
Función 12	2	1
Función 13	1	2
Función 14	3	1
Función 15	1	2

Tabla 22:MyFEPS Ágil. Velocidad en el Acceso a las funciones. 17.4.4.U

NFTot = 15

MLTF = 29

MediaNFI = 33

$$\text{Valoración} = \frac{MLTF}{\text{MediaNFI}}$$

$$\text{Valoración} = \frac{29}{33}$$

$$\text{Valoración} = 0,87$$

GCAT = 0,87

### En el conocimiento del sistema en su totalidad

17.6.1.U Evaluación Promedio de usuarios en lo relativo al grado de conocimiento de las Operaciones del Sistema

TOS = 50

Cant usuarios = 5

Usuario	Total de operaciones ejecutadas
Usuario 1	35
Usuario 2	41
Usuario 3	38
Usuario 4	46
Usuario 5	42

Tabla 24: MyFEPS Ágil. Evaluación promedio de usuarios en lo relativo al grado de conocimiento de las operaciones del sistema 17.6.1.U

$$\text{SUMT} = (35 + 41 + 38 + 46 + 42) = 202$$

$$\text{Valoración} = \frac{\left(\frac{\text{SumT}}{\text{cant. usuarios}}\right)}{100}$$

$$\text{Valoración} = \frac{\left(\frac{202}{5}\right)}{100}$$

$$\text{Valoración} = 0,40$$

GCAT = 0,40

Las preguntas que se les suministraron a los usuarios para completar el cuestionario, fueron extraídas del siguiente documento: MyFEPS - Descripción de Atributos y Métricas 2.7 v2019-03-13 Blanco - página 19 y 20 [15].

Haciendo uso de dicho cuestionario, se obtuvieron los siguientes valores:

Característica Subcaracterística Atributo	Métrica	M
<b>3.13 Satisfacción de los usuarios</b>		
<b>3.13.2 En el acceso a las funciones</b>		
13.2.1.A Facilidad de Acceso a las Funciones percibido en entorno de uso	Preguntar a N usuarios: ¿De 0 a 100 cuan fácil le resulta acceder a las funciones más usadas? : GFAi $M = \text{SUM}(\text{GFAi}) / N$	<b>82</b>
<b>3.13.3 En la comprensión de las salidas del sistema</b>		
13.3.1.A Grado de comprensión de las salidas del sistema.	Preguntar a N usuarios: ¿De 0 a 100 cuan confortable le resulta trabajar con el sistema? : GCSI $M = \text{SUM}(\text{GCSI}) / N$	<b>80</b>
<b>3.13.6 En satisfacción total</b>		
13.6.1.A Grado de Satisfacción percibido en entorno de uso	Preguntar a N usuarios: ¿De 0 a 100 cuan satisfactorio le resulta el sistema? : GSi	<b>86</b>

Figura 29: Resultados del cuestionario satisfacción de los usuarios.

#### 4.10 Medir los atributos requeridos por las métricas.

En este apartado se va a detallar el puntaje obtenido del cuestionario pesos de las características y subcaracterísticas. Una vez que los usuarios hayan concluido las pruebas descritas en el ítem 4.9, se procederá a calcular las métricas seleccionadas para cada característica utilizando el modelo suministrado por MyFeps denominado “Planilla de cálculo de G Software”, en la cual, se volcarán las mediciones y las ponderaciones para obtener el grado final de calidad del producto de software.

#### Características

Característica	Muy Importante	Importante	Medianamente Importante	Algo Importante	Totalmente sin Importancia
Correctitud		0,90			
Usabilidad (Objetiva)		0,95			
Satisfacción de los Usuarios		0,90			

Tabla 25: MyFEPS Ágil. Características.

**Subcaracterísticas**

<b>Correctitud</b>					
<i>Sub Característica</i>	<i>Muy Importante</i>	<i>Importante</i>	<i>Medianamente Importante</i>	<i>Algo Importante</i>	<i>Totalmente sin Importancia</i>
<i>Correctitud de datos</i>		0,90			
<i>Correctitud de procesos</i>		0,80			
<b>Promedio</b>		<b>0,85</b>			

<b>Usabilidad (Objetiva)</b>					
<i>Sub Característica</i>	<i>Muy Importante</i>	<i>Importante</i>	<i>Medianamente Importante</i>	<i>Algo Importante</i>	<i>Totalmente sin Importancia</i>
<i>En el Aprendizaje</i>		0,95			
<i>En la Entrada Manual de Información</i>		0,75			
<i>En el Acceso a las Funciones</i>		0,85			
<i>En el conocimiento del sistema en su totalidad</i>		0,85			
<b>Promedio</b>		<b>0,86</b>			

<b>Satisfacción de usuarios</b>					
<i>Sub Característica</i>	<i>Muy Importante</i>	<i>Importante</i>	<i>Medianamente Importante</i>	<i>Algo Importante</i>	<i>Totalmente sin Importancia</i>
<i>En el acceso a las funciones</i>		0,85			
<i>En la Comprensión de las salidas del sistema</i>		0,80			
<i>Satisfacción total</i>		0,85			
<b>Promedio</b>		<b>0,83</b>			

Tabla 26: MyFEPS Ágil. Subcaracterísticas

**Atributos**

<b>Correctitud</b>					
Atributos	Muy Importante	Importante	Medianamente Importante	Algo Importante	Totalmente sin Importancia
Número de errores en los datos detectados en un período dado		0,80			
Número de errores en los procesos detectados en un período dado		0,70			

<b>Satisfacción de los Usuarios</b>					
Atributos	Muy Importante	Importante	Medianamente Importante	Algo Importante	Totalmente sin Importancia
Facilidad de Acceso a las Funciones percibido en entorno de uso		0,75			
Grado de comprensión de las salidas del sistema percibido en entorno de uso		0,80			
Grado de Satisfacción percibido en entorno de uso		0,80			

<b>Usabilidad (Objetiva)</b>					
Atributos	Muy Importante	Importante	Medianamente Importante	Algo Importante	Totalmente sin Importancia
Tiempo promedio de Aprendizaje intuitivo		0,95			
Ayuda para la Prevención de Errores en el ingreso manual de Información		0,70			
Velocidad en el Acceso a las Funciones		0,85			

Evaluación Promedio de usuarios en lo relativo al grado de conocimiento de las Operaciones del Sistema		0,80			
--	--	------	--	--	--

Tabla 27: MyFEPS Ágil. Atributos

#### 4.10.1 Volcado de valores en Planilla de cálculo de G Software

Una vez obtenidos los valores de las características, subcaracterísticas y atributos, se procede a volcar dichos valores en la planilla de cálculo para determinar el grado de calidad del producto de software.

					Producto: Dashboard de ventas							
					PSC	PSSC	PAT	GCAT	$\frac{\text{SUM}(\text{PAT}(i) \cdot \text{GCAT}(i))}{\text{SUM}(\text{PAT}(i))}$	$\frac{\text{SUM}(\text{PSC}(i) \cdot \text{GCSC}(i))}{\text{SUM}(\text{PSC}(i))}$	GC	Grado de CALIDAD Del Producto
Básica CB	Sub	Sub-sub	Código Atributo	Descripción Atributo	peso C	peso SC	Peso SSC	Peso Atributo	Grado de Calidad: Es la Medición del Atributo	GCSC	Grado de Calidad de la SC	Grado de Calidad de la CB
					0,90							0,84
<b>Correctitud</b>												0,96706
	1.			Correctitud de datos		0,90					1	
			04.1.1.A.	Número de errores en los datos detectados en un período dado				0,80	1			
	2.			Correctitud de procesos		0,80					0,93	
			04.2.1.A.	Número de errores en los datos detectados en un período dado				0,70	0,93			
<b>Usabilidad (Objetiva)</b>												0,75928
	1.			Aprendizaje		0,95					1	
			17.1.4	Tiempo promedio de Aprendizaje Intuitivo				0,95	1			
	2.			En la Entrada Manual de Información		0,75					0,76	
			3.17.3.3.U	Ayuda para la Prevención de Errores en el ingreso manual de Información				0,70	0,76			
	3.			En el Acceso a las Funciones		0,85					0,87	
			3.17.3.3.U	Velocidad en el Acceso a las Funciones				0,85	0,87			
	4.			En el conocimiento del sistema en su totalidad		0,90					0,4	
			17.6.1.U	Evaluación Promedio de usuarios en lo relativo al grado de conocimiento de las Operaciones del Sistema				0,80	0,4			
<b>Satisfacción de los Usuarios</b>												0,8103
	1.			En el acceso a las funciones		0,85					0,8	
			13.2.1.A	Facilidad de Acceso a las Funciones percibido en entorno de uso				0,75	0,82			
	2.			En la Comprensión de las salidas del sistema		0,80					0,8	
			13.3.1.A	Grado de comprensión de las salidas del sistema percibido en entorno de uso				0,80	0,8			
	3.			Satisfacción total		0,85					0,86	
			13.6.1.A	Grado de Satisfacción percibido en entorno de uso				0,8	0,86			

Figura 30: Planilla de cálculo de G Software

Fuente: imagen extraída de mi propio desarrollo.

#### **4.11 Resultados de la evaluación.**

Una vez realizado el volcado de valores en la Planilla de cálculo, se obtuvieron los siguientes resultados:

- Correctitud: 0.96
- Usabilidad Objetiva: 0.75
- Satisfacción de los usuarios: 0.81

Dados los resultados obtenidos, es posible concluir que estamos ante un software que es robusto en cuanto a correctitud de datos y procesos, dado que el grado de calidad obtenido para este atributo es por demás de aceptable.

Además de ser correcto en su funcionamiento, está pensado para ser utilizable por la mayoría de usuarios que no necesariamente cuenten con conocimientos intermedios y avanzados en informática y/o en el análisis de datos. Siendo esta última componente, más específica de un área y no tan en general, por lo que en materia de usabilidad, es aceptable.

Finalmente, el grado final de calidad los resultados obtenidos en las encuestas fueron positivos y el grado final de calidad del producto es de 0.84. Es por ello que, es posible concluir que el software “Dashboard de ventas” es un producto con un grado de calidad en términos generales.

#### **5.0 Conclusiones**

Habiendo concluido con el análisis y con el grado final de calidad del software en el apartado 4.11 (“Resultados de la evaluación”), se demostró, haciendo uso de MyFEPS Ágil, que el producto de software desarrollado en esta ocasión, cumple en su funcionamiento, es fácil de utilizar y, por lo tanto, logró captar la atención de los usuarios finales (además de los stakeholders), siendo esta herramienta gran utilidad para el análisis de datos dentro de una compañía pyme que no maneje excesivos volúmenes de datos.

#### **5.1 Conclusiones personales.**

A lo largo del presente trabajo final de carrera, he aplicado todos los conocimientos adquiridos durante mi carrera de grado, tanto técnicos como funcionales, más en concreto aquellos relacionados con: programación, bases de datos, calidad de software e ingeniería de software. Además de aplicar dichos conocimientos, he profundizado y adquirido nuevos conocimientos durante el desarrollo de este proyecto, lo cual es un logro importante para mi desarrollo

personal y profesional. No obstante, habiendo realizado el trabajo final de carrera, he de decir que logré finalizar con mis estudios universitarios, obteniendo el título de Licenciado en Sistemas de Información.

## **5.2 Futuras líneas de investigación**

Como futuras líneas de investigación en base a este trabajo final de carrera, se propone lo siguiente:

- Replicar el proyecto utilizando una tecnología cloud como Microsoft Azure o Amazon Web Services (AWS).
- Desarrollar el ETL utilizando entornos cloud de Azure o AWS tales como Databricks.
- Expandir el dashboard y evaluar posibles oportunidades de mejora.

## Referencias bibliográficas

[1] Tableau. "Business Intelligence: What It Is, How It Works, Its Importance, Examples, & Tools". Visitada en: Sep. 24, 2021. [Online]. Accesible:

<https://www.tableau.com/learn/articles/business-intelligence>

[2] MyFeps: Sitio Oficial. Visitada en: Sep. 24, 2021 [Online]. Accesible:

<https://sites.google.com/a/comunidad.ub.edu.ar/myfeps/>

[2.1] Paula Angeleri, Rolando Titiosky, Jorge Ceballos. Framework de Evaluación de Productos Software. Visitada en: Sep. 24, 2021 [Online]. Accesible:

[http://sedici.unlp.edu.ar/bitstream/handle/10915/54072/Documento\\_completo.pdf-PDFA.pdf?sequence=1](http://sedici.unlp.edu.ar/bitstream/handle/10915/54072/Documento_completo.pdf-PDFA.pdf?sequence=1)

[2.2] Rolando Titiosky, Paula Angeleri, MyFEPS Proceso de evaluación Ágil de productos de software, Universidad de Belgrano – Versión 11-07-2021.

[3] Microsoft. Data warehousing in Microsoft Azure. Visitada en Sep. 24, 2021 [Online].

Accesible:

<https://docs.microsoft.com/en-us/azure/architecture/data-guide/relational-data/data-warehousing>

[3.1] Gustavo R. Rivadera: La metodología de Kimball para el diseño de almacenes de datos (Data warehouses). Visitada en: Sep. 24, 2021. Accesible:

<http://ucasal.edu.ar/hm/ingenieria/cuadernos/archivos/5-p56-rivadere-formateado.pdf>

[4] Wikipedia. OLAP Cube. Visitada en: Oct 1, 2021 [Online]. Accesible:

<https://es.wikipedia.org/wiki/OLAP>

[4.1] Microsoft. Analysis Services Azure. Visitada en: Oct 1, 2021 [Online]. Accesible:

<https://azure.microsoft.com/es-es/services/analysis-services/>

[5] Jorge Luis Ceballos. Ingeniería de SW IV - Testing y Calidad. Introducción a la Calidad y Riesgos 2020-08-04, Pág 49.

- [6] Denis Guido. Diseño Implementación de Data Warehouse con SQL Server 2019. Visitada Oct 23, 2021 [ONLINE]. Accesible:  
<https://www.udemy.com/course/disenio-e-implementacion-de-datawarehouse-con-sql-server-2019/>
- [7] Portal de información de ciencia y tecnología argentino. Dataset personas Nov 10, 2021 [ONLINE]. Accesible:  
<https://datasets.datos.mincyt.gob.ar/dataset/personal-de-ciencia-y-tecnologia/archivo/8ab77b16-f1a8-4d3f-b664-67becf83a9b9>
- [8] Microsoft. Python SQL Driver - pyodbc. Visitada en: Dic 20, 2021 [Online]. Accesible:<https://docs.microsoft.com/en-us/sql/connect/python/pyodbc/python-sql-driver-pyodbc?view=sql-server-ver15>
- [9] Microsoft. Apply DAX basics in Power BI Desktop. Visitada en Marzo 20, 2022 [Online]. Accesible:  
<https://docs.microsoft.com/en-us/power-bi/transform-model/desktop-quickstart-learn-dax-basics>
- [10] datadata. Cómo usar Power BI desde cero. Visitada en: Marzo 20, 2022 [Online]. Accesible: <https://www.youtube.com/watch?v=pwJuFbyhZFE>
- [11] Microsoft. The Power Query user interface. Visitada en: Marzo 20, 2022 [Online]. Accesible:  
<https://docs.microsoft.com/en-us/power-query/power-query-ui>
- [12] Nielsen Norman Group. Why You Only Need to Test with 5 Users. Visitada en: Marzo 21, 2022 [Online]. Accesible:  
<https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>
- [13] María Victoria Balin, Evaluando la Accesibilidad de un sitio E-Learning aplicando el Framework MyFEPS, Tesis de grado ingeniería en informática. Universidad de Belgrano - 2019
- [14] Agustín Ventura. Planilla de Cálculo de G Software Evaluado - Versión 03-07-2013.
- [15] Amos Sorgen, Rolando Titiosky, Martín Santi, Paula Angeleri, 3 - MyFEPS - Descripción de Atributos y Métricas 2.7 v2019-03-13 Blanco - páginas 19 y 20.

[16] ISO 25000/25010, Usabilidad. Visitada en: Abril 1, 2022. [Online] Accesible:

<https://iso25000.com/index.php/normas-iso-25000/iso-25010/23-usabilidad#:~:text=Capacidad%20del%20producto%20software%20para,Capacidad%20para%20reconocer%20su%20adecuaci%C3%B3n.>

## Glosario

1. BI: Business Intelligence.
2. OLTP: Online Transaction Processing. Son bases de datos transaccionales orientadas al procesamiento de transacciones.
3. OLAP: Online Analytical Processing. Son bases de datos multidimensionales orientadas a consultas analíticas de datos.
2. KPI: Key performance indicator.
3. Usabilidad: Es la “Capacidad del producto de software para ser entendido, aprendido, usado y resultar atractivo para el usuario, cuando se usa bajo determinadas condiciones” según la norma ISO 25010 [16].
4. Dashboard: Conjunto de visualizaciones que representan gráficos en PowerBI.
5. Cloud: Son servicios suministrados por el uso de una red de servidores que están en internet y se caracterizan por la alta disponibilidad y la escalabilidad.
6. Riesgo: efecto de la incertidumbre de los objetivos.
7. Data Warehouse: es un repositorio centralizado de datos integrados por una o más fuentes de datos.
8. Data Mart: es un módulo específico de un datawarehouse focalizado en un tema o un área de negocio dentro de una organización.
9. ETL: es un proceso que se encarga de extraer datos de una o más fuentes, luego se aplican transformaciones a ese conjunto de datos para luego cargarlos en un repositorio de datos.
10. Dataset: Conjunto de datos.
- 11 Dashboard: Vista de un conjunto de datos.
12. Tarjetas Power BI: son objetos de Power BI que sirven para representar un valor cuantificable o un valor de una métrica.
13. NDE: Número de datos erróneos provistos por el sistema.
14. NDNE: Número de datos no erróneos provistos por el sistema.
15. NPE: El número de procesos erróneos provistos por el sistema.
16. NNE: El número de procesos no erróneos provistos por el sistema.
17. LTOE: define las tareas organizacionales que deben llevar a cabo los usuarios evaluados a través del Sistema.

18. TiLTOE: “El tiempo en que se espera que se ejecuten la lista de Tareas Organizacionales relacionadas con el Sistema”

19. nTACE: Número de controles que el usuario no completa con éxito.

20. NFTot: Total de funciones implementadas en el sistema.

21. GCAT: Valor de la medición del atributo que determina el grado de calidad.

## Anexos

En este apartado, se muestran las respuestas de los stakeholders y el cuestionario, suministrado por el modelo MyFEPS, del cual se hizo uso para este trabajo final de carrera.

### 1. Respuestas del cuestionario de pesos.

#### Respuestas Stakeholder 1

	Muy importante	importante	Medianamente importante	Algo importante	Totalmente sin importancia	Puntuación
<b>3.4 Correctitud</b>	x					1,00
<b>3.4.1 Correctitud de datos</b>	x					1,00
04.1.1.A Numero de errores en los datos detectados en un período dado	x					1,00
<b>3.4.2 Correctitud de procesos</b>	x					1,00
04.1.2.A Numero de errores en los procesos detectados en uso en un período dado		x				0,75
<b>3.13 Satisfacción de los usuarios</b>	x					1,00
<b>3.13.2 En el acceso a las funciones</b>	x					1,00
13.2.1.A Facilidad de Acceso a las Funciones percibido en entorno de uso		x				0,75
<b>3.13.3 En la Comprensión de las salidas del sistema</b>		x				0,75
13.3.1.A Grado de comprensión de las salidas del sistema percibido en entorno de uso		x				0,75
<b>3.13.6. Satisfacción total</b>		x				0,75
13.6.1.A Grado de Satisfacción percibido en entorno de uso		x				0,75
<b>3.17 Usabilidad Objetiva</b>		x				0,75
<b>3.17.1 En el Aprendizaje</b>	x					1,00
17.1.4 Tiempo promedio de Aprendizaje intuitivo	x					1,00
<b>3.17.3 En la Entrada Manual de Información</b>			x			0,50
3.17.3.3.U Ayuda para la Prevención de Errores en el ingreso manual de Información		x				0,75
<b>3.17.3 En el Acceso a las Funciones</b>		x				0,75
17.4.1.U Velocidad en el Acceso a las Funciones	x					1,00
<b>3.17.6 En el conocimiento del sistema en su totalidad</b>			x			0,50
17.6.1.U Evaluación Promedio de usuarios en lo relativo al grado de conocimiento de las Operaciones del Sistema		x				0,75

#### Respuestas Stakeholder 2:

	Muy importante	importante	Medianamente importante	Algo importante	Totalmente sin importancia	Puntuación
<b>3.4 Correctitud</b>		x				0,75
<b>3.4.1 Correctitud de datos</b>		x				0,75
04.1.1.A Numero de errores en los datos detectados en un período dado		x				0,75
<b>3.4.2 Correctitud de procesos</b>		x				0,75
04.1.2.A Numero de errores en los procesos detectados en uso en un período dado		x				0,75
<b>3.13 Satisfacción de los usuarios</b>		x				0,75
<b>3.13.2 En el acceso a las funciones</b>		x				0,75
13.2.1.A Facilidad de Acceso a las Funciones percibido en entorno de uso		x				0,75
<b>3.13.3 En la Comprensión de las salidas del sistema</b>		x				0,75
13.3.1.A Grado de comprensión de las salidas del sistema percibido en entorno de uso	x					1,00
<b>3.13.6. Satisfacción total</b>	x					1,00
13.6.1.A Grado de Satisfacción percibido en entorno de uso		x				0,75
<b>3.17 Usabilidad Objetiva</b>	x					1,00
<b>3.17.1 En el Aprendizaje</b>	x					1,00
17.1.4 Tiempo promedio de Aprendizaje intuitivo	x					1
<b>3.17.3 En la Entrada Manual de Información</b>	x					1,00
3.17.3.3.U Ayuda para la Prevención de Errores en el ingreso manual de Información		x				0,75
<b>3.17.3 En el Acceso a las Funciones</b>	x					1,00
17.4.1.U Velocidad en el Acceso a las Funciones	x					1,00
<b>3.17.6 En el conocimiento del sistema en su totalidad</b>	x					1,00
17.6.1.U Evaluación Promedio de usuarios en lo relativo al grado de conocimiento de las Operaciones del Sistema		x				0,75

**Respuestas Stakeholder 3:**

	Muy importante	importante	Medianamente importante	Algo importante	Totalmente sin importancia	Puntuación
<b>3.4 Correctitud</b>		x				0,75
<b>3.4.1 Correctitud de datos</b>		x				0,75
04.1.1.A Numero de errores en los datos detectados en un período dado			x			0,50
<b>3.4.2 Correctitud de procesos</b>		x				0,75
04.1.2.A Numero de errores en los procesos detectados en uso en un período dado			x			0,50
<b>3.13 Satisfacción de los usuarios</b>	x					1,00
<b>3.13.2 En el acceso a las funciones</b>		x				0,75
13.2.1.A Facilidad de Acceso a las Funciones percibido en entorno de uso		x				0,75
<b>3.13.3 En la Comprensión de las salidas del sistema</b>	x					1,00
13.3.1.A Grado de comprensión de las salidas del sistema percibido en entorno de uso		x				0,75
<b>3.13.6. Satisfacción total</b>		x				0,75
13.6.1.A Grado de Satisfacción percibido en entorno de uso	x					1,00
<b>3.17 Usabilidad Objetiva</b>	x					1,00
<b>3.17.1 En el Aprendizaje</b>	x					1,00
17.1.4 Tiempo promedio de Aprendizaje intuitivo		x				0,75
<b>3.17.3 En la Entrada Manual de Informacion</b>			x			0,50
3.17.3.3.U Ayuda para la Prevención de Errores en el ingreso manual de Información			x			0,50
<b>3.17.3 En el Acceso a las Funciones</b>		x				0,75
17.4.1.U Velocidad en el Acceso a las Funciones			x			0,50
<b>3.17.6 En el conocimiento del sistema en su totalidad</b>		x				0,75
17.6.1.U Evaluación Promedio de usuarios en lo relativo al grado de conocimiento de las Operaciones del Sistema			x			0,50

**Respuestas Stakeholder 4:**

	Muy importante	importante	Medianamente importante	Algo importante	Totalmente sin importancia	Puntuación
<b>3.4 Correctitud</b>		x				1,00
<b>3.4.1 Correctitud de datos</b>		x				1,00
04.1.1.A Numero de errores en los datos detectados en un período dado		x				0,75
<b>3.4.2 Correctitud de procesos</b>		x				0,75
04.1.2.A Numero de errores en los procesos detectados en uso en un período dado		x				0,75
<b>3.13 Satisfacción de los usuarios</b>	x					1,00
<b>3.13.2 En el acceso a las funciones</b>	x					1,00
13.2.1.A Facilidad de Acceso a las Funciones percibido en entorno de uso		x				0,75
<b>3.13.3 En la Comprensión de las salidas del sistema</b>		x				0,75
13.3.1.A Grado de comprensión de las salidas del sistema percibido en entorno de uso		x				0,75
<b>3.13.6. Satisfacción total</b>		x				0,75
13.6.1.A Grado de Satisfacción percibido en entorno de uso		x				0,75
<b>3.17 Usabilidad Objetiva</b>	x					1,00
<b>3.17.1 En el Aprendizaje</b>		x				0,75
17.1.4 Tiempo promedio de Aprendizaje intuitivo	x					1,00
<b>3.17.3 En la Entrada Manual de Informacion</b>		x				0,75
3.17.3.3.U Ayuda para la Prevención de Errores en el ingreso manual de Información		x				0,75
<b>3.17.3 En el Acceso a las Funciones</b>	x					1,00
17.4.1.U Velocidad en el Acceso a las Funciones		x				0,75
<b>3.17.6 En el conocimiento del sistema en su totalidad</b>	x					1,00
17.6.1.U Evaluación Promedio de usuarios en lo relativo al grado de conocimiento de las Operaciones del Sistema	x					1,00

**Respuestas Stakeholder 5:**

	Muy importante	importante	Medianamente importante	Algo importante	Totalmente sin importancia	Puntuación
<b>3.4 Correctitud</b>	x					1,00
<b>3.4.1 Correctitud de datos</b>	x					1,00
04.1.1.A Numero de errores en los datos detectados en un período dado	x					1,00
<b>3.4.2 Correctitud de procesos</b>		x				0,75
04.1.2.A Numero de errores en los procesos detectados en uso en un período dado		x				0,75
<b>3.13 Satisfacción de los usuarios</b>		x				0,75
<b>3.13.2 En el acceso a las funciones</b>		x				0,75
13.2.1.A Facilidad de Acceso a las Funciones percibido en entorno de uso		x				0,75
<b>3.13.3 En la Comprensión de las salidas del sistema</b>		x				0,75
13.3.1.A Grado de comprensión de las salidas del sistema percibido en entorno de uso		x				0,75
<b>3.13.6. Satisfacción total</b>	x					1,00
13.6.1.A Grado de Satisfacción percibido en entorno de uso		x				0,75
<b>3.17 Usabilidad Objetiva</b>	x					1,00
<b>3.17.1 En el Aprendizaje</b>	x					1,00
17.1.4 Tiempo promedio de Aprendizaje intuitivo	x					1,00
<b>3.17.3 En la Entrada Manual de Informacion</b>	x					1,00
3.17.3.3.U Ayuda para la Prevención de Errores en el ingreso manual de Información		x				0,75
<b>3.17.3 En el Acceso a las Funciones</b>		x				0,75
17.4.1.U Velocidad en el Acceso a las Funciones	x					1,00
<b>3.17.6 En el conocimiento del sistema en su totalidad</b>	x					1,00
17.6.1.U Evaluación Promedio de usuarios en lo relativo al grado de conocimiento de las Operaciones del Sistema	x					1,00

## 2. Respuestas de cuestionario de pesos.

### Respuestas Usuario 1:

Complete las siguientes preguntas:

1. (13.2.1.A) ¿De 0 a 100 cuan fácil le resulta acceder a las funciones más usadas?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GFAi
								X			80

2. (13.3.1.A) ¿De 0 a 100 cuan confortable le resulta trabajar con el sistema?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GCSi
								X			80

3. (13.6.1.A) ¿De 0 a 100 cuan satisfactorio le resulta el sistema?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GSI
									X		90

### Respuestas Usuario 2:

Complete las siguientes preguntas:

1. (13.2.1.A) ¿De 0 a 100 cuan fácil le resulta acceder a las funciones más usadas?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GFAi
							X				70

2. (13.3.1.A) ¿De 0 a 100 cuan confortable le resulta trabajar con el sistema?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GCSi
								X			80

3. (13.6.1.A) ¿De 0 a 100 cuan satisfactorio le resulta el sistema?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GSI
									X		90

### Respuestas Usuario 3:

Complete las siguientes preguntas:

1. (13.2.1.A) ¿De 0 a 100 cuan fácil le resulta acceder a las funciones más usadas?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GFAi
									X		90

2. (13.3.1.A) ¿De 0 a 100 cuan confortable le resulta trabajar con el sistema?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GCSi
								X			80

3. (13.6.1.A) ¿De 0 a 100 cuan satisfactorio le resulta el sistema?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GSI
								X			80

### Respuestas Usuario 4:

Complete las siguientes preguntas:

1. (13.2.1.A) ¿De 0 a 100 cuan fácil le resulta acceder a las funciones más usadas?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GFAi
									X		90

2. (13.3.1.A) ¿De 0 a 100 cuan confortable le resulta trabajar con el sistema?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GCSi
									X		90

3. (13.6.1.A) ¿De 0 a 100 cuan satisfactorio le resulta el sistema?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GSI
									X		90

### Respuestas Usuario 5:

Complete las siguientes preguntas:

1. (13.2.1.A) ¿De 0 a 100 cuan fácil le resulta acceder a las funciones más usadas?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GFAi
								X			80

2. (13.3.1.A) ¿De 0 a 100 cuan confortable le resulta trabajar con el sistema?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GCSi
							X				70

3. (13.6.1.A) ¿De 0 a 100 cuan satisfactorio le resulta el sistema?

0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%	GSI
								X			80

### Archivos para la entrega:

Los archivos que se adjuntan con el trabajo final de carrera, son los siguientes:

- 3 Planilla de Cálculo de G SoftwareEvaluado\_Dashboard de ventas Ignacio Sacha Duplaa.
- Cuestionario Satisfacción de Usuarios- Dashboard de Ventas - Ignacio Duplaa.
- Cuestionarios Pesos Dashboard de Ventas - Ignacio Duplaa.
- ETL Python (contiene el código fuente que se utilizó en el apartado 3.8).

Adicionalmente, se adjunta el enlace para poder descargar el proyecto. Dentro del enlace, se encuentra el Data Warehouse (DW\_Ventas.bak) y dashboard de ventas (Dashboard Ventas.pbix). El enlace está configurado para que todo usuario que cuente con un dominio @ub y que tenga acceso al enlace, pueda visualizarlo pero no editarlo y/o comentarlo.

Enlace al proyecto:

[https://drive.google.com/drive/folders/1neU3-G2iq9AuRzpe8QwnZS1HCEDAfJ\\_i?usp=sharing](https://drive.google.com/drive/folders/1neU3-G2iq9AuRzpe8QwnZS1HCEDAfJ_i?usp=sharing)