



FACULTAD DE INGENIERÍAS Y CIENCIAS APLICADAS

DESARROLLO DE UNA APLICACIÓN DE FORECAST PARA YANBAL EC

AUTOR

Karla Lissette Mafla Mafla

AÑO

2020



FACULTAD DE INGENIERÍAS Y CIENCIAS APLICADAS

DESARROLLO DE UNA APLICACIÓN DE FORECAST PARA YANBAL EC

Trabajo de Titulación presentado en conformidad con los requisitos  
establecidos para optar por el título de Ingeniero en Sistemas de Computación  
e Informática

Profesor Guía

MSc. Verónica Fernanda Falconí Ausay

Autora

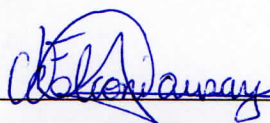
Karla Lissette Mafla Mafla

Año

2020

## **DECLARACIÓN DEL PROFESOR GUÍA**

"Declaro haber dirigido el trabajo, desarrollo de Aplicación de Forecast para Yanbal EC que apoye el proceso de estimación de joyas, a través de reuniones periódicas con el estudiante Karla Lissette Mafla Mafla, en el semestre 2020-2, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación"



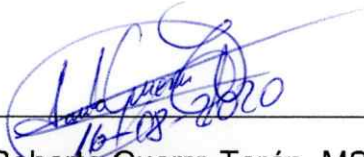
Verónica Fernanda Falconí Ausay

Magister en Ciencias de la Computación y Comercio Electrónico

C. I: 0502395270

## DECLARACIÓN DEL PROFESOR CORRECTOR

"Declaro haber corregido el trabajo, desarrollo de Aplicación de Forecast para Yanbal EC que apoye el proceso de estimación de joyas, a través de reuniones periódicas con el estudiante Karla Lissette Mafla Mafla, en el semestre 2020-2, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación"

A handwritten signature in blue ink, appearing to read 'Paulo Roberto Guerra Terán', is written over a horizontal line. The signature is stylized and includes the date '16/08/2020' written below it.

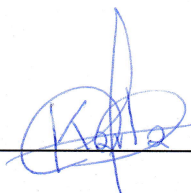
Ing. Paulo Roberto Guerra Terán, MSc.

Máster en Software y Sistemas

C. I: 1002856050

## DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

“Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.”

A handwritten signature in blue ink, appearing to be 'Karla', written over a horizontal line.

Karla Lissette Mafla Mafla

C. I: 1721357133

## **AGRADECIMIENTOS**

A mi familia, novio y docentes

por su apoyo y guía

## **DEDICATORIA**

A mis padres, mis abuelos, mis hermanos y Hernan que siempre me han apoyado para ser mejor cada día.

## RESUMEN

La empresa Yanbal Ecuador elabora productos usando ingredientes locales de Ecuador y los utiliza como medio de comercialización a nivel de todo el país con el método de venta directa por medio de sus consultoras y directoras las cuales promocionan los beneficios y eficacia de los productos Yanbal desde Joyería hasta maquillaje, para lo cual realizan estimaciones sobre la venta de cada producto el cual debe tener un precio y un mínimo de venta por cada campaña. La falta de información concreta sobre los resultados de los productos en los diferentes lanzamientos en especial para la categoría de Joyería Yanbal permitió que se desarrolle el sistema descrito en este documento, partiendo desde un marco teórico, pasando por un desarrollo hasta llegar a la validación del sistema. Gracias a este sistema los analistas de marketing podrán buscar, seleccionar y calcular la estimación de los productos que se lanzarán cada 4 semanas, facilitando el cálculo para la orden de unidades sobre cada producto en la categoría de Joyería permitiendo que los productos no queden en inventario y mejorando las estadísticas de venta sobre lanzamiento de nuevos productos. La solución fue desarrollada con el marco de trabajo Scrum, base de datos MySQL, como framework se utilizó Python con las librerías Thinker, CV2, panda.



## **ABSTRACT**

The company Yanbal Ecuador that produces products using local Ecuadorian ingredients and uses them as a mean of marketing throughout the country with the direct sales method through its consultants and directors who promote the benefits and efficacy of Yanbal products from Jewelry to makeup, for which they make estimates on the sale of each product which must have a price and a minimum sale for each campaign. The lack of concrete information on the results of the products in the different launches, especially for the Yanbal Jewellery category, allowed the system described in this document to be developed, starting from a theoretical framework, going through a development until reaching the validation of the system. Thanks to this system, marketing analysts will be able to search, select and calculate the estimate of the products that will be launched every 4 weeks, facilitating the calculation for the order of units on each product in the Jewellery category, allowing the products not to be in inventory. and improving sales statistics on new product launches. The solution was developed with the Scrum framework, MySQL database, as a framework Python was used with the Thinker, CV2, panda libraries.

# ÍNDICE

Introducción .....	1
Alcance.....	2
Justificación .....	5
Objetivo General.....	5
Objetivos específicos.....	5
1. CAPÍTULO 1. Marco teórico.....	6
1.1 Introducción.....	6
1.2 Yanbal Ecuador.....	6
1.3 Venta Directa .....	7
1.3.1 ¿Qué es Venta Directa?.....	7
1.3.2 Ventajas de la Venta Directa.....	8
1.3.2.1 Ventajas para la empresa .....	8
1.3.2.2 Ventajas para el cliente .....	8
1.3.3 Importancia de la Venta Directa .....	9
1.4 Herramientas y aplicaciones de Estimación en Yanbal EC.....	9
1.4.1 Excel .....	9
1.4.1.1 Ventajas para el Excel.....	10
1.4.2 Servidor de Archivos o File Server.....	10
1.4.2.1 Funcionamiento de un File Server.....	11
1.5 Marco de trabajo .....	11
1.5.1 Definición e historia .....	11
1.5.2 Equipo Scrum.....	12
1.5.3 Ciclo de vida de Scrum .....	14

1.6 Tecnologías de desarrollo .....	15
1.6.1 Framework de desarrollo.....	15
1.6.2 Lenguaje de programación.....	17
1.6.3 Base de datos .....	18
1.6.4 Repositorio de Versionamiento GitLab.....	19
<b>2. CAPÍTULO 2. Propuesta de Solución .....</b>	<b>19</b>
2.1 Introducción.....	19
2.2 Product Backlog .....	20
2.3 Historias de Usuario.....	23
2.4 Arquitectura de Sistema .....	26
2.5 Sprints.....	27
2.5.1 Flujo del primer Sprint .....	28
2.5.2 Flujo del segundo Sprint.....	29
2.5.3 Flujo del tercer Sprint .....	30
2.6 Primer Sprint .....	31
2.6.1 Sprint Planning.....	31
2.6.2 Sprint Backlog .....	31
2.6.3 Daily Scrum .....	32
2.6.4 Sprint Review .....	32
2.6.5 Código del primer Sprint.....	32
2.6.6 Capturas de pantalla del incremento del primer Sprint.....	33
2.6.7 Sprint Retrospective .....	34
2.7 Segundo Sprint .....	34
2.7.1 Sprint Planning.....	34
2.7.2 Sprint Backlog .....	34
2.7.3 Daily Scrum.....	35

2.7.4 Sprint Review .....	35
2.7.5 Código segundo Sprint.....	36
2.7.6 Capturas de pantalla del incremento del segundo Sprint.....	37
2.7.7 Sprint Retrospective .....	39
2.8 Tercer Sprint .....	39
2.8.1 Sprint Planning.....	39
2.8.2 Sprint Backlog .....	39
2.8.3 Daily Scrum.....	40
2.8.4 Sprint Review .....	40
2.8.5 Código segundo Sprint.....	40
2.8.6 Capturas de pantalla del incremento del segundo Sprint.....	43
2.8.7 Sprint Retrospective .....	44
2.9 Medición y Estimación.....	44
2.9.1 Gráfico Burn Up.....	45
2.10 Gráficos Burn Down .....	45
2.10.1 Gráfico Burn Down Primer Sprint.....	45
2.10.2 Gráfico Burn Down Segundo Sprint .....	46
2.10.3 Gráfico Burn Down Tercer Sprint .....	46
<b>3. CAPÍTULO 3. Validación de Cliente.....</b>	<b>47</b>
3.1 Introducción.....	47
3.2 Pruebas de aceptación del cliente .....	47
3.2.1 Demostración de las pruebas de aceptación del cliente .....	48
3.3 Cambios propuestos por cliente.....	51
3.3.1 Captura de cambios del cliente .....	52
<b>4. CAPÍTULO 4. Rendimiento Sistema .....</b>	<b>53</b>
4.1 Introducción.....	53

4.2 Pruebas unitarias .....	53
4.2.1 Demostración del cumplimiento de las pruebas unitarias .....	54
4.3 Pruebas de carga .....	56
<b>Conclusiones y recomendaciones .....</b>	<b>57</b>
Conclusiones.....	57
Recomendaciones .....	58
<b>Referencias .....</b>	<b>59</b>

## ÍNDICE DE TABLAS

TABLA 1: PRODUCT BACKLOG .....	21
TABLA 2: HISTORIA DE USUARIO NÚMERO 1 .....	24
TABLA 3: HISTORIA DE USUARIO NÚMERO 2 .....	24
TABLA 4: HISTORIA DE USUARIO NÚMERO 3 .....	25
TABLA 5: HISTORIA DE USUARIO NÚMERO 4 .....	25
TABLA 6: HISTORIA DE USUARIO NÚMERO 5 .....	26
TABLA 7: SPRINT BACKLOG DEL PRIMER SPRINT .....	31
TABLA 8: SPRINT BACKLOG DEL SEGUNDO SPRINT .....	35
TABLA 9: SPRINT BACKLOG DEL TERCER SPRINT .....	39
TABLA 10: PRUEBAS DE ACEPTACIÓN DEL CLIENTE .....	47
TABLA 11: CAMBIOS DEL CLIENTE .....	51
TABLA 12: PRUEBAS UNITARIAS. ....	53
TABLA 13: PRUEBAS DE CARGA .....	56

## INDICE DE FIGURAS

FIGURA 1: HOJAS DE CÁLCULO EXCEL .....	9
FIGURA 2: ESTRUCTURA DE FILE SERVER .....	11
FIGURA 3: CICLO DE VIDA DE SCRUM.....	15
FIGURA 4: COLABORACIÓN COMPONENTES MVC .....	16
FIGURA 5: ARQUITECTURA DE SISTEMA.....	26
FIGURA 6: FLUJO DE TRABAJO REALIZADO EN EL PRIMER SPRINT.....	28
FIGURA 7: FLUJO DE TRABAJO REALIZADO EN EL SEGUNDO SPRINT.....	29
FIGURA 8: FLUJO DE TRABAJO REALIZADO EN EL TERCER SPRINT .....	30
FIGURA 9: CÓDIGO FUENTE CARGA DE IMÁGENES.....	32
FIGURA 10: CÓDIGO FUENTE COMPARACIÓN DE IMÁGENES.....	33
FIGURA 11: IMÁGENES COMPARADAS POR SIMILITUD .....	33
FIGURA 12: IMÁGENES RESULTADO DE COMPARACIÓN .....	33
FIGURA 13: CAPTURA RESULTADO SIMILITUD DE COMPARACIÓN.....	34
FIGURA 14: CÓDIGO SELECCIÓN DE ARCHIVO Y COMPROBACIÓN DE EXISTENCIA .....	36
FIGURA 15: CÓDIGO INTEGRACIÓN INTERFAZ GRÁFICA BOTONES Y ACCIONES .....	36
FIGURA 16: CÓDIGO REGLA DE COMPARACIÓN SIMILITUD DE IMÁGENES PARA DESPLIEGUE.....	36
FIGURA 17: CÓDIGO PARA EL MANEJO DE VENTANA CON IMÁGENES DE JOYAS .....	37
FIGURA 18: CÓDIGO PARA SELECCIÓN DE IMÁGENES Y CHECKS CON ESTADO.....	37
FIGURA 19: EJECUTABLE PARA SELECCIÓN DE ARCHIVO .....	37
FIGURA 20: CARGA DE ARCHIVOS PROGRAMA SOLO PERMITE ARCHIVOS JPG.....	37
FIGURA 21: IMPRESIÓN DE MENSAJE PARA CONFIRMAR EXISTENCIA DE UN ARCHIVO. ....	38
FIGURA 22: IMPRESIÓN DE PORCENTAJE DE SIMILITUD POR COMPARACIÓN DE IMAGEN .....	38
FIGURA 23: MENSAJE CON CANTIDAD DE IMÁGENES DE MAYOR SIMILITUD .....	38
FIGURA 24: DESPLIEGUE DE IMÁGENES PARA SELECCIÓN Y PORCENTAJE DE SIMILITUD .....	38
FIGURA 25: CÓDIGO CONEXIÓN A BASE DE DATOS .....	40
FIGURA 26: CÓDIGO PARA TRANSFORMACIÓN DE IMAGEN DE JPG A GIF PARA LA SELECCIÓN .....	40
FIGURA 27: CÓDIGO PARA OBTENCIÓN DE DATOS Y ALMACENAMIENTO EN DATAFRAME .....	41
FIGURA 28: CÓDIGO PARA CREACIÓN DE TABLA Y MUESTRA DE DATOS EN INTERFAZ GRAFICA .....	41
FIGURA 29: CÓDIGO MANEJO CAMPOS INGRESADOS POR USUARIO .....	41
FIGURA 30: CÓDIGO BOTONES PARA CALCULO Y REGISTRO DE ESTIMACIÓN .....	41
FIGURA 31: CÓDIGO DE LA FUNCIÓN CALCULAR ESTIMACIÓN.....	42
FIGURA 32: CÓDIGO DE LA FUNCIÓN REGISTRAR ESTIMACIÓN.....	42
FIGURA 33: MODULO DE ESTIMACIÓN .....	43

FIGURA 34: VISTA DE MODULO ESTIMACIÓN Y SELECCIÓN.....	43
FIGURA 35: MENSAJE DE ESTIMACIÓN DEL BOTÓN CALCULO.....	43
FIGURA 36: <i>MENSAJE CONFIRMACIÓN DE GUARDADO DE DATOS E IMAGEN EN DB</i> .....	44
FIGURA 37: GRAFICO BURN UP FINAL.....	45
FIGURA 38: BURN DOWN PRIMER SPRINT .....	45
FIGURA 39: BURN DOWN SEGUNDO SPRINT .....	46
FIGURA 40: BURN DOWN TERCER SPRINT.....	46
FIGURA 41: VALIDACIÓN DE EXISTENCIA Y FORMATO .....	48
FIGURA 42: CONFIRMACIÓN DE IMÁGENES CON SIMILITUD.....	48
FIGURA 43: VISTA DE IMÁGENES SEGÚN RESULTADO DE COMPARACIÓN Y CHECK PARA ESTIMACION..	49
FIGURA 44: OBTENCIÓN DE INFORMACIÓN SOBRE IMÁGENES SELECCIONADAS .....	49
FIGURA 45: IMPRESIÓN DE CHECK Y MUESTRA EN TABLA DE DATOS RECUPERADOS .....	49
FIGURA 46: INGRESO DATOS PARA ESTIMACIÓN.....	50
FIGURA 47: IMPRESIÓN CALCULO .....	50
FIGURA 48: IMPRESIÓN REGISTRO ESTIMACIÓN .....	50
FIGURA 49: IMPRESIÓN DE NOMBRE IMAGEN Y PORCENTAJE DE SIMILITUD.....	52
FIGURA 50: IMPRESIÓN DE IMÁGENES SELECCIONADAS Y NOMBRE DE LAS IMAGENES.....	52
FIGURA 51: IMPLEMENTACIÓN DE COMBOBOX PARA CAMPAÑA Y AÑO .....	52
FIGURA 52: VALIDACIÓN DE IMÁGENES FORMATO JPG .....	54
FIGURA 53: VALIDACIÓN IMAGEN ÚNICA .....	54
FIGURA 54: VALIDACIÓN SELECCIÓN DE IMÁGENES SIMILARES.....	55
FIGURA 55: VALIDACIÓN SOLO ACEPTACIÓN NÚMEROS.....	55
FIGURA 56: VALIDACIÓN INGRESO A BASE DE DATOS .....	56
FIGURA 57: GRAFICO RESULTADO PRUEBAS DE CARGA .....	57



## **Introducción**

El presente trabajo muestra el desarrollo de un proyecto que busca mejorar la estimación de los productos para la categoría de Joyería de Yanbal Ecuador, además de generar un historial sobre los tipos de bisutería y los resultados al momento de lanzamiento de esta manera exponer los beneficios de contar con el historial sobre los productos esto permitiría que se dé una mejor posición dentro del Market Share a nivel de Ecuador y generaría un resultado más exacto en cuanto a las ventas.

Dentro de la fundamentación teórica se presenta un análisis del proceso de negocio y de las actividades realizadas en Yanbal, las tecnologías, la metodología de desarrollo y las herramientas usadas para la elaboración de este proyecto. Para finalizar se presentarán las conclusiones, las recomendaciones y las referencias.

## **Antecedentes**

Unique Yanbal, fue fundada en 1967 por Fernando Belmont, desde el primer día en la que se fundó la misión de esta empresa fue cambiar la vida de miles de mujeres, el éxito comercial de Yanbal en el pilar con los productos de belleza de altísima calidad, además de que para el nombre “Yanbal” Fernando se inspiró en el nombre su primera hija Janine Belmont por la cual el tomo la decisión de construir una empresa para poder empoderar a las mujeres del mundo, por medio de la venta de productos cosméticos con el modelo de la venta por comisiones a las vendedoras el crea a las consultoras, después de 10 años en los cuales Yanbal creció a ser una de las marcas más usadas en Perú la visión de Fernando comenzó a traspasar las fronteras y llegó a Ecuador en 1977 donde crearon su primera planta de cosméticos y bisutería desde este punto Yanbal siguió creciendo y atravesando fronteras.

Yanbal se encuentra en más de 10 países expandiendo su misión de empoderar a las mujeres, después de 50 años de llegar a Ecuador Yanbal Ecuador ya cuenta con más de 5 mil Directoras y 200 mil consultoras independientes usando el modelo de venta directa esta empresa depende de sus directoras y consultoras para realizar las ventas de sus productos ya que ellas promocionan

los productos Yanbal mediante demostraciones o exponiendo los beneficios de los productos Yanbal ya sea de los componentes nacionales con los que se realizan para promover la economía local del país hasta los colores únicos de sus maquillajes o sombras, los diseños de su Bisutería la cual es ampliamente estudiada según las tendencias del país y elaborada con materiales de alta calidad para tener una durabilidad mayor a las de las Joyas de otras marcas.

Yanbal aunque tiene 50 años en el mercado recabando tendencias y los gustos de los hombre y mujeres del Ecuador para realizar la venta y lanzamiento de productos óptimos, apreciando los resultados de las consultoras y directoras decide crear los niveles de consultoras y comienza a realizar los concursos para que las directoras y consultoras con el incremento de sus ventas ganen premios esto genera un gran impacto en la compra y venta de las diferentes categorías de productos Yanbal impactando más a las categorías de cuidado personal y Bisutería las cuales no generaban las cantidades necesarias de productos para la venta o se realizaban lanzamientos en épocas que los productos no eran aceptados por el cliente final, esto se debe a que no se cuenta con un historia de venta para el manejo de los productos ya sea lanzamientos o seguimientos en las categorías debido al hecho de que se trabaja con hojas Excel que están dispuestas a errores humanos ya que no se cuenta con ninguna aplicación para realizar el cálculo ni comparar los diferentes productos que serán utilizados para las campañas de esta manera se generara un estimado de venta y un manejo de lanzamiento para los productos según la época más adecuada.

### **Alcance**

Este proyecto de titulación propone desarrollar un sistema que constituya en un mecanismo para que los analistas/brands de Yanbal realicen la estimación/forecast para la categoría de Bisutería por este medio se obtendrá una mejor visión de la venta y aceptación de las joyas en los diferentes variantes ya sea por la campaña el tipo de joya la vista que se le da o el tipo de oferta, hay que tomar en cuenta que no son los actores más importantes para realizar la estimación ya que no existen collares o aretes iguales ya sea en diseño o colores se puede realizar comparaciones de manera que nos da una idea de los gustos

para los clientes finales de Yanbal a los que las directoras están haciendo promociones.

Este sistema permitirá cargar una imagen de la joya a estimar y realizara la búsqueda según la base de datos de los conjuntos con características más parecidas el usuario deberá seleccionar las imágenes que se usaran para la estimación y el sistema obtendrá la información de lanzamiento de las joyas la página de promoción, el tipo de exposición, la cantidad de unidades vendidas, promoción con la que se lanzó, campaña de lanzamiento y el precio de lanzamiento, según esta información el analista podrá seleccionar cuales son las joyas que se usaran para la estimación, después de generar la estimación el sistema guardara los datos generados para poder realizar la comparación de los mismos en los reportes de la aplicación así se podrá comparar el porcentaje de venta y aceptación de la joya por los usuarios.

En este proyecto se desarrollara un sistema que constituirá en una herramienta por la cual se realizaran los cálculos de venta, tendrá una interfaz gráfica con los módulos:

#### **A) Módulo de Estimación**

En este módulo el sistema permitirá generar los modelos de ventas sobre Joyas y productos de la categoría Bisutería de manera que se pueda tener estimados más reales con la comparación sobre información anterior y aplicación de fórmulas de estimación que serán proporcionados por Yanbal EC

En este módulo se contará con dos submódulos que permitirán al usuario realizar dos tipos de estimaciones:

- **Estimación Producto Nuevo**

Para realizar el cálculo de la venta de un producto sin tener una base para poder tener una predicción del rendimiento del mismo se tomará en cuenta con el uso de estimaciones con productos que tengan parecido y según la similitud de los mismos se les dará la importancia en la estimación.

- **Estimación Producto Relanzado**

Este módulo permitirá realizar la estimación de productos relanzado con información de ventas anteriores debido a que el producto en este módulo se puede realizar con la información de ventas anteriores ya que producto cuenta con una existencia anterior y puede ser calculado con más facilidad

## **B) Módulo de Datos**

Este módulo va a tener la información de todas las predicciones que el sistema haya realizado para poder tener un manejo de comparaciones con la venta real del producto en cuestión, contará con las imágenes de las bisuterías y colecciones las cuales usara el sistema para las predicciones.

## **C) Módulo de Reportes**

El módulo de reportes realizará comparaciones de las ventas reales vs predicciones para el manejo de estadísticas vs los modelos existentes, generar reportes sobre rendimiento del sistema desde el inicio de funcionamiento y la eficiencia del mismo, generará reportes de rendimientos de información, fallas registradas.

El módulo de reportes constara con un reporte especial para el administrados del sistema en el cual se detalla el uso de la herramienta según los usuarios.

Esta solución será implementada utilizando el lenguaje de programación Python y se desarrollará en el framework Spyder, contará con una base de datos MySQL y Scrum como marco de trabajo.

**Justificación**

El desarrollo de la aplicación permitirá optimizar el tiempo de estimación de las ventas que realizan los Analistas/Brand, quienes actualmente en cada campaña realizan estimaciones entre 30 a 60 productos de Bisutería, con un proceso prácticamente manual que involucra la búsqueda en la información registrada en la base de ventas de las imágenes de semejanza de los productos, calcular los totales de ventas y generar separadamente las estimaciones, Asegurar que las estimaciones sean más exactas para evitar falta de productos, generar mayor control del inventario y evitar pérdidas de valor de la venta después de su despliegue en la campaña.

Manejo de datos históricos sobre ventas de productos para generación de modelos más precisos y fiables para tener una mayor gestión de la información adquirida en los últimos 5 años de venta, obtención de rendimiento sobre las ventas para poder generar los datos de ganancia y pérdida campañales que se dan de los productos de bisutería de manera que se pueda optimizar el inventario para esta categoría.

**Objetivo General**

Desarrollar una aplicación de Forecast para ventas de productos de bisutería de Yanbal EC.

**Objetivos específicos**

- Realizar un análisis sobre la situación actual del departamento de Marketing para determinar los requerimientos del usuario.
- Desarrollar una aplicación para estimaciones del manejo de bisutería usando análisis de imágenes.
- Realizar las pruebas de funcionalidad sobre los modelos generados de proyecciones.

## **1. CAPÍTULO 1. Marco teórico**

### **1.1 Introducción**

En el capítulo, se detallarán los conceptos fundamentales que se tomarán para el desarrollo del sistema y la resolución del problema de esta manera se tendrá un concepto más claro de los conocimientos para generar una comprensión sencilla y altamente útil.

Se detallará la situación actual del Área de Marketing Yanbal y los medios actuales con los que se están realizando la estimación y los sistemas similares que han utilizado y los requerimientos que se han planteado por los analistas, después se describirá la metodología de desarrollo para la solución además de una descripción de las tecnologías a usar para el desarrollo de la aplicación y solventar los requerimientos planteados en este documento.

### **1.2 Yanbal Ecuador**

Unique Yanbal, fue fundada en 1967 por Fernando Belmont, desde el primer día en la que se fundó la misión de esta empresa fue cambiar la vida de miles de mujeres, el éxito comercial de Yanbal en el pilar con los productos de belleza de altísima calidad, además de que para el nombre “Yanbal” Fernando se inspiró en el nombre su primera hija Janine Belmont por la cual el tomo la decisión de construir una empresa para poder empoderar a las mujeres del mundo. (Yanbal, 2017)

Yanbal es una empresa con más de 50 años en el mercado y presencia en más de 10 países a nivel mundial, está dedicada a la venta directa de sus productos por lo que no tiene contacto con el cliente final sino con sus directoras y consultoras para realizar la venta sobre los productos de sus cuatro categorías principales maquillaje, bisutería, cuidado personal y perfumes estas categorías son manejadas para la venta campañal y se realizan estimaciones de los productos de igual manera por lo que cada campaña estos cambian la categoría. (Yanbal, 2017)

El manejo de la categoría esta realizado por un Brand pero a diferencia de las demás categorías bisutería no tiene productos iguales sino solo similares y estos son 30 nuevos diseños por cada campaña el encargado de la categoría debe

buscar de manera manual las joyas más parecidas para después buscar en un Excel con toda la historia de joyas los datos de la joya lo que propenso a errores ya sea humano o de los datos que son registrados de manera manual cada campaña. (Yanbal, 2017)

### **1.3 Venta Directa**

#### **1.3.1 ¿Qué es Venta Directa?**

“Es la definición para la comercialización de productos fuera del establecimiento o tienda y dando las demostraciones a los clientes finales sobre los productos en venta de manera que la vendedora o emprendedora da muestras personalizadas, lo que la distingue de las ventas por retail en supermercados o tiendas, los vendedores son personas que firman un acuerdo o licencia con la empresa para poder vender y promocionar los productos de la misma. (DIRECTA, 2019)

Dentro de la venta directa existe la venta multinivel es cuando un fabricante vende productos o servicios por medio de una red de personas las cuales promocionan un producto y están coordinadas en una sola red comercial la cual tiene un ingreso por medio de la venta de productos de alta calidad y costo con garantías de devolución. (DIRECTA, 2019)

Es una estrategia de las grandes empresas para evitar usar proveedores o distribuidores de manera que la fuerza de venta es decir las emprendedoras o vendedoras realizan la venta de los productos lo que ofrece una manera eficiente de comercializar los productos. (DIRECTA, 2019)

Se deben tener en cuenta 4 aspectos fundamentales para considerar realizar estrategia de venta directa en la empresa:

- El producto no existe o tiene un método de uso complicado.
- El número de potenciales clientes es alto
- El precio de venta del producto es alto.
- Se deben firmar documentos para realizar uso del producto

La venta directa permite generar ahorro en distribución de los productos para invertirlo en la calidad de los productos a producir, también se puede invertir ese dinero para dar más incentivos a las vendedoras y animarlas a vender más.” (DIRECTA, 2019)

### **1.3.2 Ventajas de la Venta Directa**

#### **1.3.2.1 Ventajas para la empresa**

Las principales ventajas que tienen las empresas que usan el modelo de venta directa son:

- **“Mayor influencia sobre el mensaje que da el producto:**

La empresa no usa a intermediarios para relacionarse con el cliente final lo que le permite enviar el mensaje sobre sus productos de una misma manera y que sea entendido con más facilidad. (Quiroa, n.d.)

- **Un enfoque en el sector de mercado al que se dirige el producto:**

Permite conocer a los clientes en los que la empresa se quiere enfocar para la compra y venta de sus productos lo que le da la facilidad de dar publicidad enfocada a ese sector del mercado en específico al que se acomodan los clientes. (Quiroa, n.d.)

- **Facilidad para medir resultados de los productos y vendedoras:**

Para las empresas no es sencillo medir los resultados de los gastos que se realizan para poder generar el producto contra la venta o rendimiento alcanzado de los mismos para poder saber si el resultado es positivo o negativo.” (Quiroa, s.f.)

#### **1.3.2.2 Ventajas para el cliente**

El cliente final tendrá los beneficios de la estrategia de venta directa las más importantes son:

- **“Productos de alta calidad a un precio más bajo:**

La empresa dimite en los costos como intermediarios o distribuidores lo que genera en causa que el producto tenga un precio más bajo y conveniente para los clientes ya que si hay muchos intermediarios el precio del producto se puede incrementar. (Quiroa, n.d.)



- **Ventas y demostraciones personalizadas:**

Las emprendedores y vendedoras se encargan de dar demostraciones de manera que se adaptan a las necesidades del cliente.” (Quiroa, s.f.)

### 1.3.3 Importancia de la Venta Directa

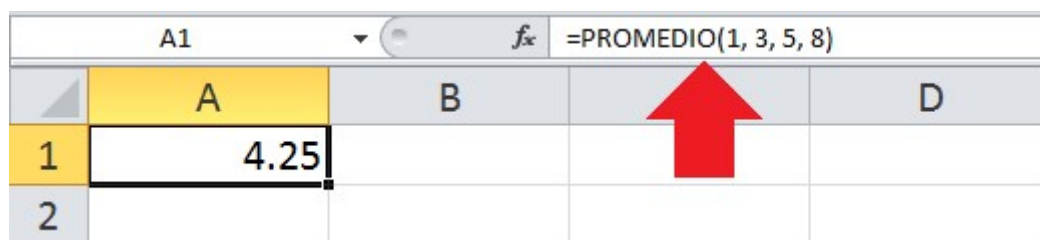
“La venta directa logra establecer una relación de confianza mutua entre sus vendedoras y emprendedoras lo que se da como un enlace que permite tener una relación semi directa a los clientes finales, esta estrategia de venta permite tener un punto de enfoque más personalizado lo que genera la oportunidad de demostrar y describir la importancia de los productos además de su beneficio y su forma de uso “ (Quiroa, s.f.)

## 1.4 Herramientas y aplicaciones de Estimación en Yanbal EC

### 1.4.1 Excel

Aplicación creada por la Corporación Microsoft es una herramienta que permite crear hojas de cálculo en las cuales se puede ingresar y generar formulas, es distinguido de los demás programas ya que permite trabajar con números, esto facilita el trabajo con números, permite generar reportes ya sea con gráficos o solo con tablas. (Microsoft, 2020)

El manejo de las hojas de cálculo electrónicas reemplaza a los sistemas de calculadoras digitales o en papel, fueron creadas para manejo de tareas contables, la primera versión de Excel en fue Multiplan que fue lanzado en 1982 después se lanzó Lotus 1-2-3 el cual se posiciono en el mercado de manera rápida por las hojas de cálculo que se utilizaban. (Microsoft, 2020)



	A	B	C	D
1	4.25			
2				

Figura 1: Hojas de Cálculo Excel

(Adaptado de Grupo Microsoft)

#### **1.4.1.1 Ventajas para el Excel**

El manejo de Excel sobre una hoja de papel o la calculadora de la computadora tiene más ventajas que solo realizar cálculos, las más conocidas son:

- Con el manejo de la inteligencia mejorada en Excel organiza y aprende los patrones del usuario, lo que permite crear una plantilla o hoja de cálculo desde cero usando las fórmulas actualizadas continuamente.
- Realizar gráficos en Excel permite presentar los datos de una forma más interactiva y convincente usando mini gráficos, tablas y formatos.
- Realizar trabajos compartidos los cuales siempre están actualizados, se generan autoguardados de manera automática, permite a los usuarios trabajar en tiempo real ya sea uno o más usuarios.
- Gestionar los permisos de acceso al documento para cada usuario desde permisos de edición, visualización y eliminación de información hasta conexiones a bases de datos. (Microsoft, 2020)

#### **1.4.2 Servidor de Archivos o File Server**

“Se lo define como un conjunto de servidores de una red la cual se pone a disponibilidad de los usuarios, este proporciona una opción de almacenamiento para los documentos con los soportes de datos, el cual está disponible solamente para los usuarios con permisos o autorizados.” (IONOS, 2018)

“Si el File server se encuentra conectado a internet se le puede dar el acceso a escritorios remotos no únicamente a repositorios de datos, estos repositorios son utilizados con todos los sistemas operativos como Linux, Windows o macOs que tienen compatibilidad con los dispositivos de la red, se los usa como servidores de respaldo ya sea de archivos o de programas que deben estar disponibles en todo momento.” (IONOS, 2018)

“Las ventajas de los File Servers para empresas son las siguientes:

- Organización sencilla de conjuntos de archivos.
- Visualización clara de la estructura
- Intercambio fácil de archivos
- Evitar errores por diferentes versiones de archivos

- Aligeramiento del dispositivo del usuario en cuanto al espacio del equipo.
- Protección y seguridad de datos internos.” (IONOS, 2018)

#### 1.4.2.1 Funcionamiento de un File Server

“Los protocolos de red son responsables de la comunicación entre el file server y las computadoras de los clientes: mientras que el protocolo SMB (Server Message Block), desarrollado por IBM, para los dispositivos Windows y macOS, los ordenadores con sistemas unixoides como las distribuciones de Linux utilizan el protocolo NFS (Network File System). Se puede combinar ambos tipos de protocolos sin problemas en una misma red, debe actualizar los clientes y servidores de archivos en Linux/Unix con una implementación del protocolo SMB en estos sistemas, por ejemplo, con el paquete de software Samba, disponible de forma gratuita.” (IONOS, 2018)

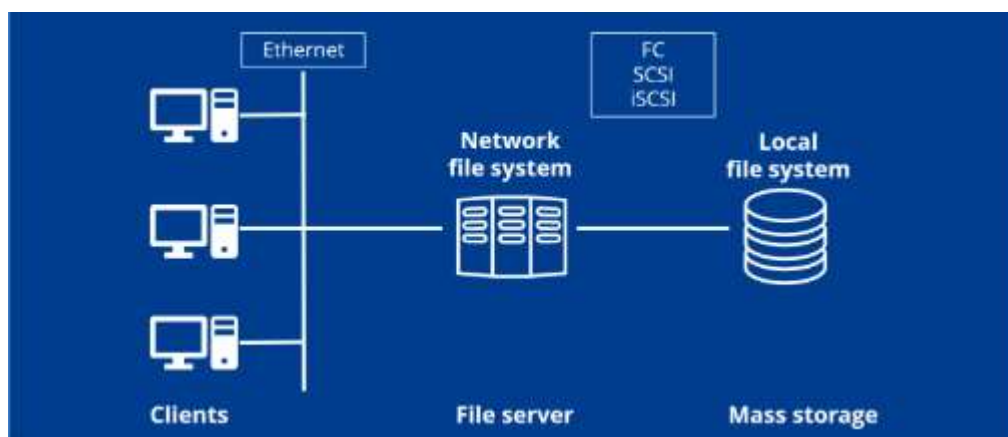


Figura 2: Estructura de File Server

(Adaptado de Grupo IONOS)

## 1.5 Marco de trabajo

El marco de trabajo que se utilizará para este proyecto será Scrum.

### 1.5.1 Definición e historia

“Marco para el desarrollo de productos complejos que entrega productos de un valor máximo de productividad y creatividad es un método ligero, fácil de entender y aplicar. (Schwaber & Sutherland, 2013)

Es un método que es usado desde los principios de los años 90 este no es una técnica para desarrollar productos el cual emplea diferentes procesos y técnicas,

muestra la eficiencia de las practicas sobre la gestión de desarrollo de los productos. (Schawaber & Sutherland, 2013)

La teoría Scrum está basada en el control de procesos empíricos, esto asegura que el conocimiento y experiencia viene de tomar decisiones basándose en lo que una persona conoce, Scrum tiene un enfoque incremental que permite realizar el control de riesgo y tener una mejor predictibilidad. (Schawaber & Sutherland, 2013)

Existen tres pilares que mantienen el control de los procesos estos son:

- **Transparencia**

Deben ser transparentes los aspectos más importantes para los responsables del resultado, estos aspectos son definidos con un estándar común de modo que todos los que observan compartan un entendimiento.

- **Inspección**

Los usuarios y responsables Scrum deben verificar con frecuencia el progreso hacia el objetivo, estas inspecciones no deben interferir con el progreso del proyecto.

- **Adaptación**

Los inspectores pueden determinar si un proceso se está desviando de su definición original lo que puede causar que el producto final no sea aceptable por lo que el proceso debe ser ajustado para adaptarse a las necesidades del producto final” (Schawaber & Sutherland, 2013)

### **1.5.2 Equipo Scrum**

“Los equipos de trabajo Scrum son multifuncionales y autoorganizados, estos equipos de trabajo elijen la mejor forma de llevar el trabajo sobre el proyecto y son dirigidos exclusivamente por los miembros del equipo, estos equipos entregan productos de manera incremental e interactiva lo que da más oportunidades de tener la retroalimentación de los usuarios” (Schawaber & Sutherland, 2013)

Los equipos de trabajo Scrum están conformados por:

- **Dueño de Producto (Product Owner)**

Es la persona responsable de dar el verdadero valor al producto y del trabajo al equipo de desarrollo Scrum, es el único que puede administrar la lista de producto (Producto Backlog) esto incluye: (Schawaber & Sutherland, 2013)

- Expresar los elementos del Product Backlog.
- Optimizar el trabajo realizado por el equipo de desarrollo.
- Asegurar el entendimiento del elemento de la lista por el equipo de desarrollo.

El dueño del producto es una única persona no un grupo de personas, este puede representar los deseos de un comité, para que el dueño del producto pueda trabajar de manera eficiente la organización debe respetar sus decisiones. (Schawaber & Sutherland, 2013)

- **Equipo de Desarrollo (Development Team)**

Consiste en los profesionales que desempeñaran el trabajo de desarrollar y entregar el incremento del producto y el producto Terminado, que se pondrá en un sprint a producción para demostración del usuario, estos son estructurados y se reparten las tareas de los sprint, estos equipos están caracterizados por: (Schawaber & Sutherland, 2013)

- Cuentan con todas las habilidades para crear un incremento del producto.
- No h áreas especializadas en las que están hace diferencia entre los títulos de los miembros del equipo, todos son desarrolladores sin importar el trabajo que realicen.
- Pueden tener habilidades y especialidades por las cuales están asignadas sus responsabilidades.

- **Scrum Master**

Responsable de asegurar que el Scrum y el proyecto sean entendidos esto lo realizan asegurándose de que el Equipo Scrum trabaja ajustándose y entendiendo las teorías y prácticas de la metodología Scrum.” (Schawaber & Sutherland, 2013)

### 1.5.3 Ciclo de vida de Scrum

El ciclo de vida Scrum es un ciclo predefinido en el cual se crea un desarrollo incremental del producto los cuales se llaman sprints que son de una duración constante, existen 5 etapas las cuales son: (Schawaber & Sutherland, 2013)

#### 1) Planificación Sprint:

El trabajo que se realizara en el sprint se planifica en la reunión de planificación de Sprint el plan de desarrollo se crea con todo el Equipo Scrum. (Schawaber & Sutherland, 2013)

La reunión tiene dos partes importantes, cada una es de la mitad de duración de tiempo sobre la reunión:

- ¿Qué se entregará como resultado del Sprint?
- ¿Cómo se realizará el trabajo acordado?

#### 2) Scrum Diario:

Es una reunión de todo el equipo scrum de máximo 15 minutos, la cual se utiliza para determinar el plan de trabajo de las 24 horas siguientes, esto se da debido a una inspección sobre los últimos avances del Scrum. (Schawaber & Sutherland, 2013)

#### 3) Desarrollo durante el Sprint:

Una vez que el Sprint está en curso se debe asegurar que:

- No se darán cambios que afecten al Sprint.
- No disminuya la calidad y se el alcance se logre con el equipo de desarrollo.

#### 4) Revisión del Sprint:

La revisión se lleva a cabo al final de cada Sprint, para poder tener una retroalimentación y adaptar el proyecto de ser necesario, el equipo y las partes interesadas ya sean usuarios o el dueño del producto colaboran con la revisión de lo desarrollado en el Sprint. (Schawaber & Sutherland, 2013)

#### 5) Retrospectiva del Sprint:

Es la inspección y creación de un plan para realizar mejoras en el siguiente Sprint el propósito del mismo es revisar como fue el desarrollo del Sprint anterior no solo desarrollo sino las relaciones,

procesos y las herramientas, ordenar e identificar lo que salió bien y cuáles son las mejoras que se pueden dar. (Schawaber & Sutherland, 2013)

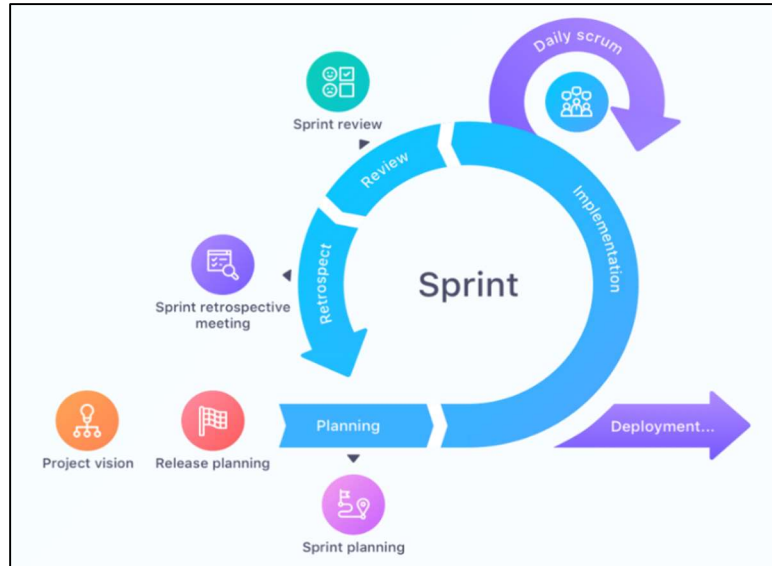


Figura 3: Ciclo de vida de Scrum

(Adaptado de Grupo Obsbusiness)

## 1.6 Tecnologías de desarrollo

### 1.6.1 Framework de desarrollo

“La definición de Framework se usa en muchos casos para el desarrollo y creación de sistemas no solamente para las aplicaciones , se refiere a una estructura de software es un conjunto de componentes intercambiables y personalizables dentro del desarrollo de un sistema, se lo puede tomar como una aplicación incompleta y fácil de configurar para agregar los componentes que permitirán construir la aplicación. (Gutierrez, 2019)

Los principales objetivos de un Framework son acelerar el desarrollo, generar y promover las prácticas de desarrollo con los patrones de diseño, permite centrarse en el problema que se desea solucionar. (Gutierrez, 2019)

La mayoría de Framework tienen una gran cantidad de usuarios, los que realizan desarrollos o liberan extensiones con funcionalidades específicas esto permite tener acceso a las funcionalidades sin desarrollarlas o usarlas como base para el sistema en desarrollo.” (Gutierrez, 2019)

Para el proyecto y su desarrollo se usará el Framework con el patrón Model-View-Controller o Modelo-Vista-Controlador (MVC) el cual separa los datos de un sistema, la lógica de control del sistema y la interfaz de usuario en componentes diferentes de esta forma si se modifica uno de los componentes tiene un mínimo impacto sobre los otros dos componentes. (Gutierrez, 2019)

“El modelo consta de tres componentes específicos los cuales se detallará a continuación:

- **Modelo:**

En esta capa se encuentran los datos del sistema los que el usuario espera ver.

- **Vista:**

Se encarga de generar la interfaz de usuario puede ser desde un archivo de Excel normal o una página HTML, este componente tiene el propósito de mostrar los datos en una forma entendible para el usuario.

- **Controlador:**

Este componente es el responsable del comportamiento y peticiones lógicas del usuario el cual construye un modelo y se lo envía al componente de vista para su visualización.” (Mestras, 2018)

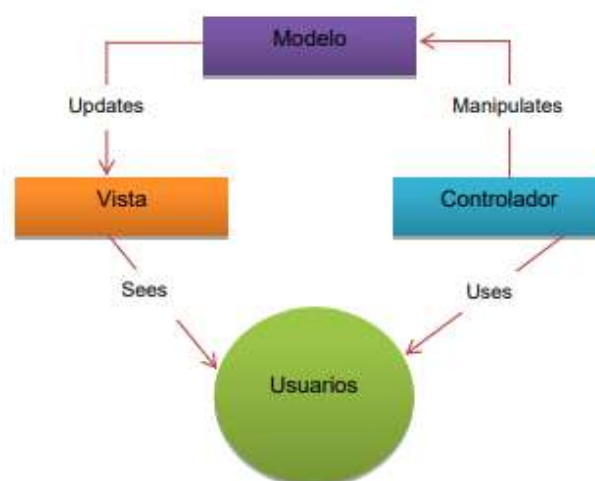


Figura 4: Colaboración componentes MVC

(Adaptado de Universidad Alejandro de Humboldt)



### **1.6.1.1 Spyder Anaconda**

“Spyder es un entorno escrito y diseñado en Python para analistas, científicos e ingenieros, da una funcionalidad de edición depuración y análisis de los perfiles de la herramienta ya sea inspección de los datos, ejecución o depuración. (Contributors, 2018)

Anteriormente llamado Pydee es un componente extensible con los complementos publicados, se puede utilizar como biblioteca para proporcionar widgets con la consola y de muy alto poder. (Contributors, 2018)

Las características generales de Spyder son:

- Editor multilenguaje, tiene la funcionalidad de analizar el código, división horizontal y vertical para visualización de análisis y ejecución.
- Ipython es un espacio que permite evaluar de manera inmediata el código escrito y viene con la integración a Matplotlib.
- Facilidad de explorar los archivos que se usa en el programa o se está generando antes de guardarlos.” (Contributors, 2018)

### **1.6.2 Lenguaje de programación**

Python es el lenguaje de programación seleccionado para el desarrollo de la solución, cuenta con estructuras de alto nivel y eficientes para un enfoque simple, pero a la vez efectivo de la programación orientada a objetos. (Foundation, 2020)

Lenguaje tipo scripting, listo para realizar cualquier tipo de programas de páginas hasta servidores de red, se define como lenguaje interpretador la definición de esto es que no se necesita compilar para poder ejecutar el programa. (Foundation, 2020)

Es un lenguaje multiplataforma se puede ejecutar e instalar en Windows, Mac, Unix es un tipo de lenguaje open source es decir gratuito, sencillo de aprender de alta velocidad de ejecución ya que tiene una gran cantidad de librerías, funciones y tipos de datos que ayudan a realizar instrucciones sin programas un código totalmente nuevo.” (Foundation, 2020)

### 1.6.3 Base de datos

“Una base de datos es una colección estructurada de datos. Puede ser cualquier cosa, desde una simple lista de compras hasta una galería de imágenes o la gran cantidad de información en una red corporativa. Para agregar, acceder y procesar los datos almacenados en una base de datos de computadora, se necesita un sistema de administración de base de datos como MySQL Server. Dado que las computadoras son muy buenas para manejar grandes cantidades de datos, los sistemas de administración de bases de datos desempeñan un papel central en la informática, como utilidades independientes o como parte de otras aplicaciones” (MySQL, 2018). Y cuenta con el respaldo de Oracle Corporation, ya que ellos son los encargados de su distribución, mantenimiento y desarrollo.

El sistema administrador de base de datos seleccionado para este proyecto es MySQL, el cual emplea el lenguaje SQL (Structured Query Language o lenguaje de consulta estructurada), este lenguaje es el lenguaje estandarizado más utilizado para acceder a las bases de datos y que está definido por el standard ANSI/ISO, el cual ha ido evolucionando desde su primera aparición en 1986.

MySQL es una base de datos relacional y al ser una base de este tipo los datos son almacenados en tablas separadas para evitar colocar todos los datos dentro de un solo dispositivo de almacenaje. Dentro de las estructuras de la base de datos existen ficheros físicos que están optimizados (tunning) para brindar una mejor velocidad y un mejor procesamiento de la información. Además de estos ficheros físicos existe también, el modelo lógico. El modelo lógico está formado por los objetos en la base de datos cómo, por ejemplo: tablas, vistas, filas y columnas, etc.

Una de las principales ventajas de usar MySQL es que es código abierto lo cual permite que cualquier usuario lo use y si desea modifique el software.

Otra de las principales ventajas de MySQL es su rapidez, escalabilidad, facilidad de uso, confiabilidad, la habilidad de ser ejecutado desde cualquier computadora, sea esta una computadora con altas o bajas prestaciones y su capacidad de integración con otras aplicaciones o servidores.

MySQL es usado, generalmente, en sistemas cliente-servidor, el cual permite que varios clientes se conecten al servidor de back-end, ya sea con la ayuda de herramientas o administrativas (DBMS), aplicaciones o programas de cliente, librerías o cualquiera de las herramientas de programación. (MySQL 2018)

#### **1.6.4 Repositorio de Versionamiento GitLab**

Nace como un repositorio Git para alojamiento de sistemas, un hosting de proyectos gestionado por el sistema de versiones Git, ofrece un conjunto más grande de servicios como gestión de proyectos, grupos o snippets. (Zimmer, 2020)

Los Snippets son conocidos como pedazos de códigos que te sirven varias partes del proyecto ya sea modificando el código base o usándolo como respaldo de las funcionalidades base del sistema. (Zimmer, 2020)

“Unas de las principales características de GitLab son:

- Tienes todo el servicio bajo tu control. Esto es muy importante para empresas que trabajan con código privado y sensible.
- Puedes instalarlo en cualquier plataforma UNIX si lo haces manualmente, pero para ciertas distribuciones de Linux, cuentas con paquetes de instalación que hacen que el proceso sea súper simple y rápido. Estos paquetes están disponibles para:
  - Ubuntu 12.04 y 14.04
  - Debian 7
  - CentOS 6 y 7
  - RedHat Enterprise Linux y similares (Oracle Linux, Scientific Linux).” (Zimmer, 2020)

## **2. CAPÍTULO 2. Propuesta de Solución**

### **2.1 Introducción**

En el capítulo, se detallará la solución propuesta con el objetivo de mejorar y optimizar la tarea de estimación para los productos de Yanbal EC, se demostrará el uso de la metodología Scrum como el marco de trabajo para el desarrollo de este proyecto, se describirá el Product Backlog y los Sprint realizados en el

desarrollo de la aplicación al completar cada uno de los Sprint se presentará el resultado incremental que se obtuvo del proyecto.

Se mostrará el diseño de la tabla para la base de datos y los diagramas para el manejo de archivos en la solución ya que estos se encontrarán en un File server almacenados para que pueda acceder el programa, pero los usuarios no podrán ingresar al repositorio de archivos.

## **2.2 Product Backlog**

Se detalla la tabla para las historias de usuarios la cual será denominada tabla 1, en esta se encontrarán los resultados obtenidos de los usuarios en cuanto a los requerimientos que debe cumplir el sistema.

La tabla 1 está compuesta de 5 columnas las cuales contienen:

- 1) El código único para la historia de usuario.
- 2) Descripción breve de la historia de usuario.
- 3) La prioridad asignada a cada una de las historias de usuario con la calificación alta, media, baja.
- 4) La estimación del esfuerzo requerido para completar el requerimiento o tarea para el manejo de esta estimación se utilizará número del 1 al 5.

Se tomará los numero con los siguientes significados:

1. Muy baja
  2. Baja
  3. Normal
  4. Importante
  5. Muy Importante
- 5) Se visualizará los resultados que permitirá verificar el cumplimiento de los requerimientos.

El total del esfuerzo realizado para el desarrollo de la aplicación se encuentra detallado y escrito en este documento, el resultado de la sumatoria para las historias de los usuarios es el que se encuentra presente en la columna 4 de la tabla 1 con un total de 21 puntos sobre las Historias de Usuario.

Tabla 1: Product Backlog

Id	Nombre	Descripción	Prioridad	Pre-estimación	Criterio de validación
1	Ingreso y Modificación de Joyas	Como administrador se debe ser capaz de ingresar, modificar Joyas dentro del File server y la aplicación. Debe ser capaz de seleccionar una joya para modificar y si se crea una nueva verificar que no exista en el sistema.	Alta	3	<ul style="list-style-type: none"> <li>• Se debe visualizar las actualizaciones y/o creaciones inmediatamente en el sistema.</li> <li>• Se visualiza la imagen ingresada de la joya.</li> <li>• Se verificará que no exista duplicidad al momento de la creación de registros.</li> </ul>
2	Comparación de Imágenes	Al ejecutar el programa se debe poder cargar una imagen para que se realice la comparación y obtener el mensaje de si se existen similares o no	Alta	3	<ul style="list-style-type: none"> <li>• Se mostrar la cantidad de Joyas parecidas que se encuentran en el sistema</li> </ul>
3	Consulta de estimación	El usuario visualiza la lista de imágenes con el porcentaje de similitud, las cuales se podrán seleccionar para la estimación.	Media	2	<ul style="list-style-type: none"> <li>• Vista de las imágenes</li> <li>• Selección de joyas a estimar</li> </ul>

4	Obtención de datos	Se obtendrán los datos de las joyas seleccionadas para la estimación desde la base de datos con los nombres de las imágenes con similitud.	Media	5	<ul style="list-style-type: none"> <li>• Usuario visualizara la información de las joyas seleccionadas para la estimación datos como venta total, unidades, campaña y año de lanzamiento</li> </ul>
5	Reporte de Estimación	El usuario deberá generar los estimados que se necesite para tener los promedios de venta, podrá guardar la estimación validando que no exista una previa y visualizara el rendimiento del sistema contra datos reales en un reporte generado a base del histórico y DataMark	Media	8	<ul style="list-style-type: none"> <li>• Cálculo de estimación de joyas con similitud.</li> <li>• Validación en histórico sobre estimación generada para registro.</li> <li>• Visualización de estimaciones generadas para comparación con datos reales.</li> </ul>

En la tabla 1 se detalla la pre-estimación realizada por el Analista Yanbal en este caso llamado Product Owner para el desarrollo de la aplicación, esta puede estar sujeta a cambios o variaciones dependiendo de los factores presentes al momento de realizar el desarrollo ya sean factores internos o externos al desarrollo pudiendo ser estas catástrofes naturales, ausencias por enfermedades o retrasos por parte de la empresa en el detalle de las historias de usuario.

### **2.3 Historias de Usuario**

Para la definición de los requerimientos del sistema se toma en cuenta las historias de usuario que detallan las funcionalidades que deberá tener el sistema las cuales se describirá utilizando tablas.

La tabla 2 a la tabla 6 estarán compuestas de 10 filas:

- ID que es el código único de la historia.
- Usuario es el cliente que genero la historia.
- Prioridad en negocio sistema numérico que nos permite tener la secuencia de implementación de las historias de usuario.
- Estimación esfuerzo que se dedicara a la historia de usuario por parte del desarrollador.
- Sprint asignado para el uso del producto owner para saber que esperar cada sprint.
- Responsable miembro del equipo de desarrollo que se hará cargo de implementar la historia de usuario.
- Descripción resumen de la historia que debe responder a tres preguntas:
  - ¿Quién se beneficia?
  - ¿Cuál es el beneficio?
  - ¿Qué se quiere?
- Criterio de aceptación cuáles serán las pruebas que deberá realizar el usuario para dar como cumplida la historia de usuario.

Tabla 2: Historia de Usuario Número 1

<b>Historia de usuario</b>	
<b>ID:</b> 01	<b>Usuario:</b> Israel Estrada
<b>Nombre de historia:</b> Ingreso y Modificación de Joyas	
<b>Prioridad en negocio:</b> 3	<b>Riesgo en desarrollo:</b> N/A
<b>Estimación:</b> 9 horas	<b>Sprint asignado:</b> 1
<b>Responsable:</b> Karla Mafla	
<b>Descripción:</b> Como administrador, se debe ingresar y modificar joyas dentro del file server y verificar que no exista en él sistema.	
<b>Criterios de aceptación:</b> La visualización de imágenes ingresadas y modificadas, verificación de que no exista duplicidad al momento de realizar una nueva estimación.	

Tabla 3: Historia de Usuario Número 2

<b>Historia de usuario</b>	
<b>ID:</b> 02	<b>Usuario:</b> Israel Estrada
<b>Nombre de historia:</b> Comparación de Imágenes	
<b>Prioridad en negocio:</b> 3	<b>Riesgo en desarrollo:</b> N/A
<b>Estimación:</b> 9 horas	<b>Sprint asignado:</b> 1
<b>Responsable:</b> Karla Mafla	
<b>Descripción:</b> Como usuario, al ejecutar el programa se debe poder adjuntar la imagen para comparación en formato jpg y recibir el mensaje de si existen imágenes similares.	
<b>Criterios de aceptación:</b> El despliegue de una lista de imágenes similares que se pueden seleccionar para la estimación.	



Tabla 4: Historia de Usuario Número 3

<b>Historia de usuario</b>	
<b>ID:</b> 03	<b>Usuario:</b> Israel Estrada
<b>Nombre de historia:</b> Consulta de Estimación	
<b>Prioridad en negocio:</b> 2	<b>Riesgo en desarrollo:</b> N/A
<b>Estimación:</b> 6 horas	<b>Sprint asignado:</b> 2
<b>Responsable:</b> Karla Mafla	
<b>Descripción:</b> Como usuario, tener la visualización de las imágenes con el detalle de similitud y el nombre de las imágenes para poder seleccionar las que se usaran para la estimación.	
<b>Criterios de aceptación:</b> Vista de las imágenes con los datos de cada imagen y posibilidad de seleccionar las imágenes que se necesite como un mínimo de una imagen seleccionada.	

Tabla 5: Historia de Usuario Número 4

<b>Historia de usuario</b>	
<b>ID:</b> 04	<b>Usuario:</b> Israel Estrada
<b>Nombre de historia:</b> Obtención de Datos	
<b>Prioridad en negocio:</b> 5	<b>Riesgo en desarrollo:</b> N/A
<b>Estimación:</b> 15 horas	<b>Sprint asignado:</b> 2
<b>Responsable:</b> Karla Mafla	
<b>Descripción:</b> Como usuario, envié de las imágenes seleccionadas para generar la vista de las imágenes y sus datos de venta.	
<b>Criterios de aceptación:</b> La obtención y carga de información desde el repositorio con el despliegue de los datos por joya.	

Tabla 6: Historia de Usuario Número 5

Historia de usuario	
<b>ID:</b> 05	<b>Usuario:</b> Israel Estrada
<b>Nombre de historia:</b> Reporte de Estimación	
<b>Prioridad en negocio:</b> 8	<b>Riesgo en desarrollo:</b> N/A
<b>Estimación:</b> 24 horas	<b>Sprint asignado:</b> 3
<b>Responsable:</b> Karla Mafla	
<b>Descripción:</b> Como usuario, generar modelos de estimación en base a los datos que se ingrese por cantidad, costo para generar el registro de la información y la imagen.	
<b>Criterios de aceptación:</b> Manejo de campos base para estimación con los botones para generar una estimación o guardar la estimación y la imagen que se está usando en el sistema	

## 2.4 Arquitectura de Sistema

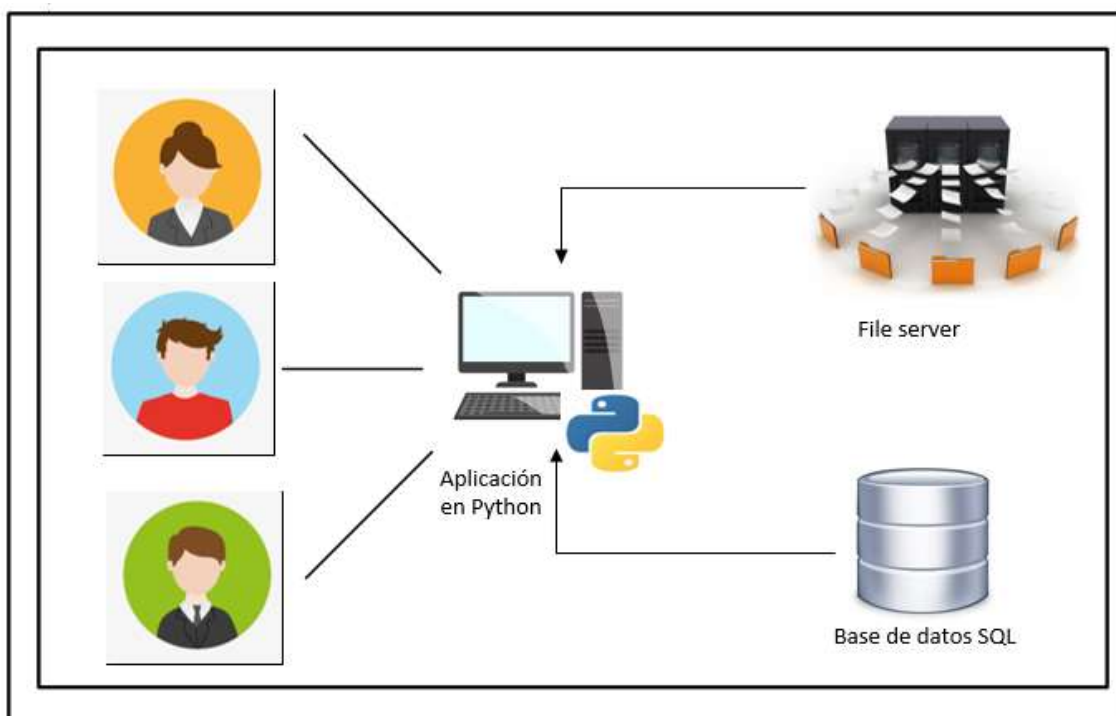


Figura 5: Arquitectura de Sistema

## 2.5 Sprints

Tomando en cuenta las historias de usuario y la prioridad de estas se decidió realizar 3 Sprints con una duración de 2 semanas cada uno.

**1. Primer Sprint:**

Compuesto por las historias de usuario con el identificador: 1,2. Estas historias de usuario tiene un total de 6 Puntos de Historia de Usuario.

**2. Segundo Sprint:**

Compuesto por las historias de usuario con identificador: 3,4. Estas historias de usuario tiene un total de 7 Puntos de Historia de Usuario.

**3. Tercer Sprint:**

Compuesto por las historias de usuario con identificador: 5. Estas historias de usuario tienen un total de 8 Puntos de Historia de Usuario,

En el primer Sprint contiene el manejo de las imágenes para la comparación y la carga de los archivos además del manejo de formato y dimensiones que se usaran para el sistema.

El segundo Sprint contiene el manejo y obtención de los datos sobre ventas de los productos y configuración de las fórmulas de estimación a usar en el sistema.

El tercer Sprint contiene el manejo de datos para analítica del sistema y generación de reportes además de un historial de estimaciones

### 2.5.1 Flujo del primer Sprint

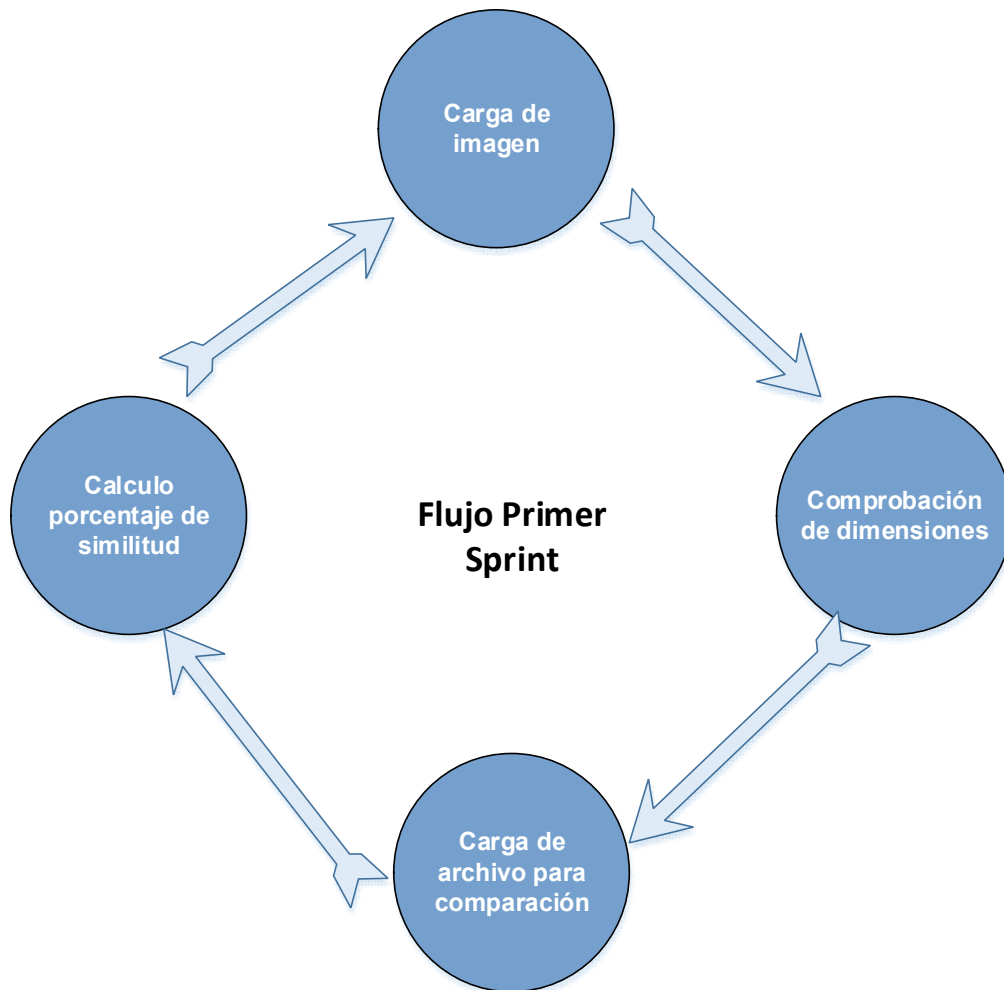


Figura 6: Flujo de trabajo realizado en el primer Sprint

## 2.5.2 Flujo del segundo Sprint

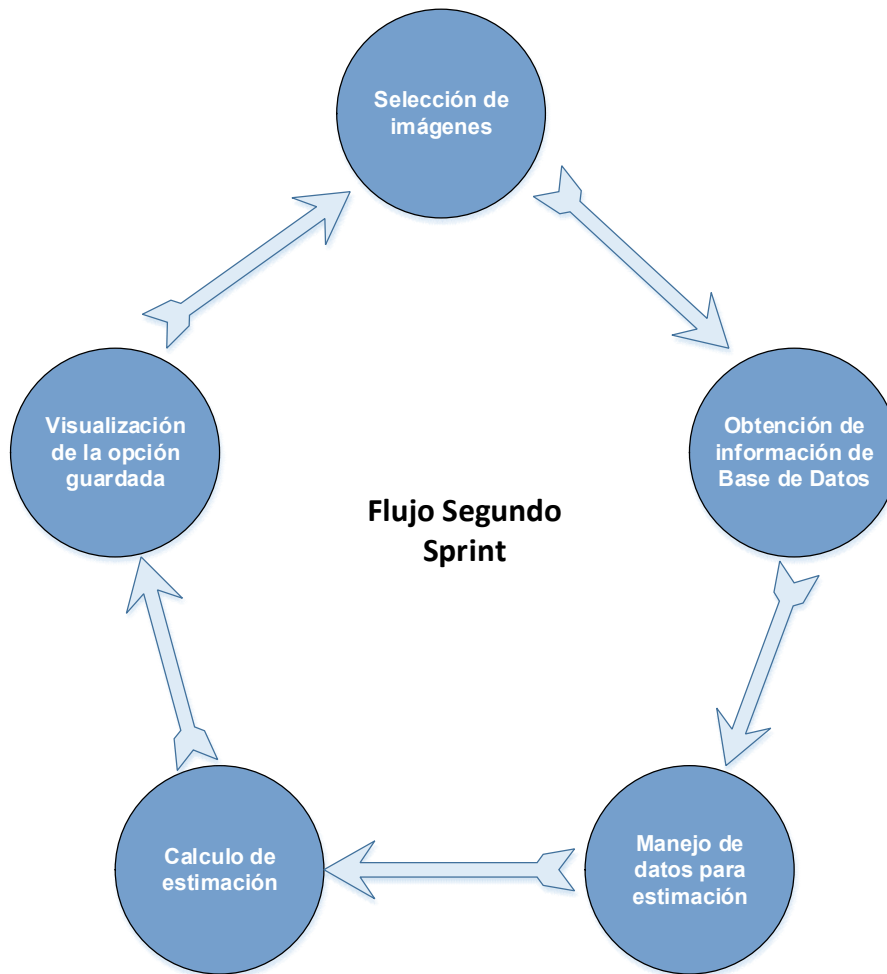


Figura 7: Flujo de trabajo realizado en el segundo Sprint

### 2.5.3 Flujo del tercer Sprint

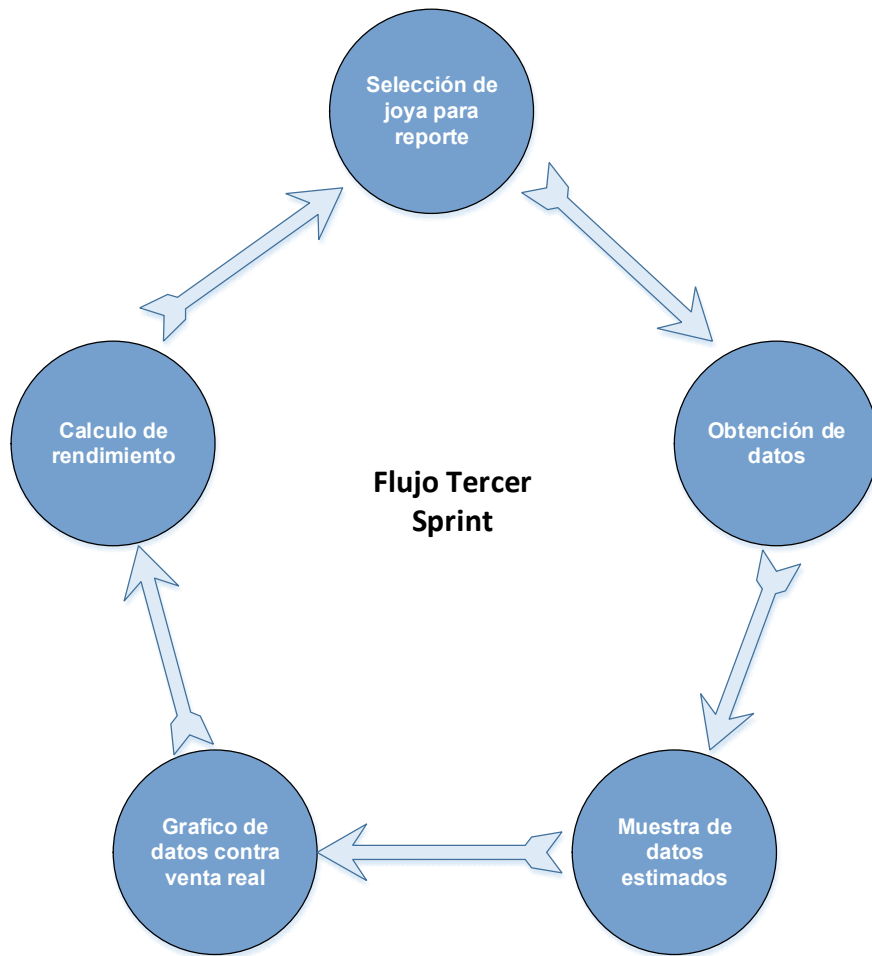


Figura 8: Flujo de trabajo realizado en el tercer Sprint

## 2.6 Primer Sprint

### 2.6.1 Sprint Planning

Se estima y selecciona las historias de usuario que van a ser desarrolladas en el primer Sprint, considerando las opiniones de todo el equipo, así como también permitir que los integrantes tengan más compromiso con el proyecto, para este primer Sprint se seleccionaron las historias de usuario 1, 2.

### 2.6.2 Sprint Backlog

En el Sprint Backlog del primer Sprint se encuentran listadas las historias de usuario con identificador 1, 2, y que están descritas en la tabla 1 ya que estas se encuentran agrupadas por actividades que permiten generar un incremento

A continuación, se muestra la tabla 7 correspondiente al Sprint Backlog del primer Sprint, en el cual se detalla las historias seleccionar para realizar en el primer Sprint.

Tabla 7: Sprint Backlog del primer Sprint

ID	Nombre	Descripción	Estado	Esfuerzo	Prioridad	Responsable
1	Ingreso y Modificación de Joyas	Como administración se debe ser capaz de ingresar, modificar Joyas dentro del File Server y la aplicación. Debe ser capaz de seleccionar una joya para modificar y si se crea una nueva verificar que no exista en el sistema.	Completo	3	Alta	Karla Mafla
2	Comparación de Imágenes	Al ejecutar el programa se debe poder cargar una imagen para que se realice la comparación y obtener el mensaje de si se encontraron similares o no	Completo	3	Alta	Karla Mafla

### 2.6.3 Daily Scrum

Cada día el equipo se reunió durante 15 minutos con el fin de responder a las 3 inquietudes, ¿Qué se hizo ayer?, ¿Qué se va a hacer hoy? y ¿Qué problemas se presentaron? Una vez respondidas las 3 interrogantes se procede a actualizar la gráfica Burn Down de Sprint, para poder observar el comportamiento y avance logrado.

En el primer Sprint no se presentaron demoras en el desarrollo durante la primera semana, durante la segunda semana se tuvo un retraso en la entrega del file server por parte de Yanbal por lo que las pruebas se retrasaron, pero se presentó el resultado incremental de las comparaciones entre imágenes son problemas

### 2.6.4 Sprint Review

Al finalizar este Sprint se llevó a cabo un Sprint Review, en donde revisó lo que se trabajó y el incremento presentado que aporta valor para el cliente, además se obtienen conclusiones, recomendaciones y experiencias que fueron compartidos con el equipo, y que servirán para los siguientes Sprints o proyectos futuros.

En el primer Sprint se obtuvo el “core” de la aplicación.

### 2.6.5 Código del primer Sprint

```

shift = cv2.xfeatures2d.SIFT_create()
kp_1, desc_1 = shift.detectAndCompute(original, None)
kp_2, desc_2 = shift.detectAndCompute(image_to_compare, None)

print("Puntos 1 Image", str(len(kp_1)))
print("Puntos 2 Image", str(len(kp_2)))

index_params = dict(algorithm=0, trees=5)
search_params = dict()

flann = cv2.FlannBasedMatcher(index_params, search_params)
matches = flann.knnMatch(desc_1, desc_2, k=2)

good_points = []
for m, n in matches:
    if m.distance < 0.6*n.distance:
        good_points.append(m)

number_keypoints = 0
if (len(kp_1) <= len(kp_2)):
    number_keypoints = len(kp_1)
else:
    number_keypoints = len(kp_2)

```

Figura 9: Código Fuente Carga de Imágenes



```

print("Puntos iguales", len(good_points))
print("Porcentaje de Similitud", len(good_points) / number_keypoints * 100, "%")

result = cv2.drawMatches(original, kp_1, image_to_compare, kp_2, good_points, None)
#cv2.imshow("Resultado", cv2.resize(result, None, fx = 0.4, fy=0.4))
cv2.imwrite("Iguales.jpg", result)

cv2.imshow("Original", original)
cv2.imshow("Duplicate", image_to_compare)
cv2.waitKey(0)
cv2.destroyAllWindows()

```

Figura 10: Código Fuente comparación de imágenes

### 2.6.6 Capturas de pantalla del incremento del primer Sprint



Figura 11: Imágenes Comparadas por Similitud

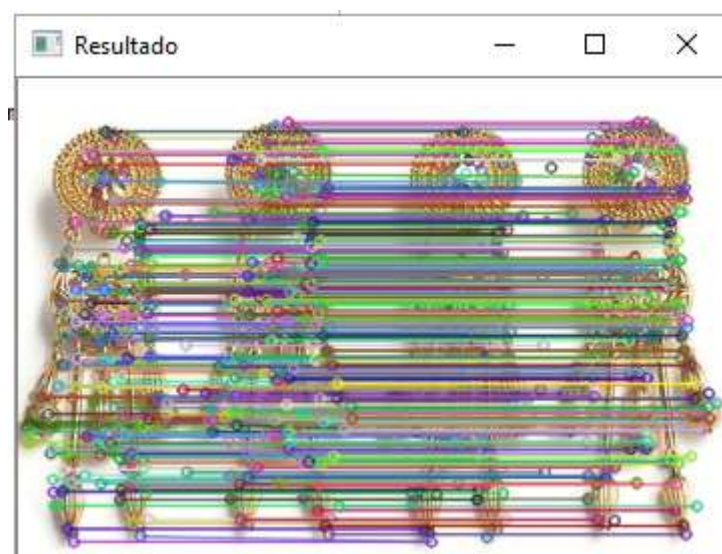


Figura 12: Imágenes resultado de Comparación

```

In [2]: runfile('D:/Users/kmafla/Documents/Proyectos de Automatizacion/
Pruebas/Titulacion/Ejecutable.py', wdir='D:/Users/kmafla/Documents/
Proyectos de Automatizacion/Pruebas/Titulacion')
Las imagenes tiene el mismo tamaño y canal
17752
Las imagenes no son iguales
Puntos 1 Image 460
Puntos 2 Image 451
Puntos iguales 376
Porcentaje de Similitud 83.37028824833703 %

```

Figura 13: Captura resultado similitud de Comparación

### 2.6.7 Sprint Retrospective

Después de realizar la carga de la imágenes se debe comprobar que tengan la misma cantidad de pixeles de lo contrario la comparación no será exacta y deben estar en formato JPG para poder ser comparadas en Python es mejor utilizar la librería OpenCv que LibCv ya que se realiza un reconocimiento de formas y puntos similares como se puede evidenciar en la figura 11 de los puntos compradas entre las imágenes de prueba.

## 2.7 Segundo Sprint

### 2.7.1 Sprint Planning

Aplicando la técnica Planning Poker se pudo estimar y seleccionar las historias de usuario que van a ser desarrolladas en el segundo Sprint, considerando las opiniones de todo el equipo, así como también permitir que los integrantes tengan más compromiso con el proyecto, para este primer Sprint se seleccionaron las historias de usuario 3, 4.

### 2.7.2 Sprint Backlog

En el Sprint Backlog del segundo Sprint se encuentran listadas las historias de usuario con identificador 3, 4 y que están descritas en la tabla 3 ya que estas se encuentran agrupadas por actividades que permiten generar un incremento

A continuación, se muestra la tabla 8 correspondiente al Sprint Backlog del primer Sprint, en el cual se detalla las historias seleccionar para realizar en el segundo Sprint.

Tabla 8: Sprint Backlog del segundo Sprint

ID	Nombre	Descripción	Estado	Esfuerzo	Prioridad	Responsable
3	Consulta de estimación	Despliegue de lista para las imágenes con el porcentaje de similitud, las cuales se podrán seleccionar para la estimación.	Completado	2	Media	Karla Mafla
4	Consulta sobre los datos de las joyas	Se obtendrán los datos de las joyas seleccionadas para la estimación desde la base de datos con los nombres de las imágenes con similitud.	Completado	5	Media	Karla Mafla

### 2.7.3 Daily Scrum

Cada día el equipo se reunió durante 15 minutos con el fin de responder a las 3 inquietudes, ¿Qué se hizo ayer?, ¿Qué se va a hacer hoy? y ¿Qué problemas se presentaron? Una vez respondidas las 3 interrogantes se procede a actualizar la gráfica Burn Down de Sprint, para poder observar el comportamiento y avance logrado.

En el segundo spring se encontró la dificultad con el manejo de las librerías Python ya que las versiones para realizar la ejecución del programa tenían varios requerimientos, manejo de archivos ya que se realizó la comprobación de que sea con formato jpg no presento dificultad en ejecución.

### 2.7.4 Sprint Review

Al finalizar este Sprint se llevó a cabo un Sprint Review, en donde revisó lo que se trabajó y el incremento presentado que aporta valor para el cliente, además se obtienen conclusiones, recomendaciones y experiencias que fueron compartidos con el equipo.

En el desarrollo del sistema se debió tener en cuenta los permisos para acceder a las carpetas ya que, si no se tenía el permiso se abría la carpeta con imágenes,

pero el programa daba error, en el manejo de la interfaz se tuvo que pedir otras imágenes ya que Python thinker solo permite imágenes gif.

El manejo de check se debe realizar las declaraciones de variables antes de iniciar el programa para poder generar los selectores, en el tamaño de las imágenes se usó la librería PI para cambiar el tamaño y calidad de la imagen.

### 2.7.5 Código segundo Sprint

```
def abrir_archivo_Catalogo():
    archivo_abierto=filedialog.askopenfilename(initialdir = "/",
        title = "Seleccione archivo",filetypes = (("Todos Los Archivos","*.jpg"),
        ("all files","*.*")))
    print(archivo_abierto)
    global Catalogo
    Catalogo=archivo_abierto;
    messagebox.showinfo('Message', 'Archivo de Joya Cargado Correctamente')
    return archivo_abierto;
```

Figura 14: Código selección de Archivo y comprobación de existencia

```
myFont1 = fon.Font(family='Courier', size=12, weight='bold')
imagenL=PhotoImage(file="portada2.gif")
lblImagen=Label(ventana,image=imagenL).place(x=0,y=0)

lbl2 = Label(ventana, text="Cargar el documento:",font=myFont1).place(x=90,y=190)

myFont = fon.Font(family='Courier', size=10, weight='bold')
btn1 = Button(ventana, text='Cargar Imagen JPG',command = abrir_archivo_Catalogo, width=20, height=2, bg='#FF6B33',

lbl3 = Label(ventana, text="Revisar Reporte de Joyas:",font=myFont1).place(x=70,y=280)
btn3 = Button(ventana, text='Ingresar a Reporte',command = abrir_archivo, width=20, height=2, bg='#FF6B33', fg='#17

myFont3 = fon.Font(family='Courier', size=8, weight='bold')
imagenL1=PhotoImage(file="Yanbal.gif")
lblImagen2=Label(ventana,image=imagenL1).place(x=360,y=370)
lbl3 = Label(ventana, text="Yanbal - 2020",font=myFont3).place(x=262,y=380)

ventana.mainloop()
```

Figura 15: Código Integración Interfaz Gráfica Botones y Acciones

```
number_keypoints = 0
if (len(kp_1) <= len(kp_2)):
    number_keypoints = len(kp_1)
else:
    number_keypoints = len(kp_2)

porcentaje=len(good_points) / number_keypoints * 100;
print("Similitud "+str(round(porcentaje,2))+"% img");
if porcentaje >80:
    contador=contador+1;
Imagenes.loc[i] = [nombre_corto]+[nombre_comparar]+[str(round(porcentaje,2))]
i=i+1;
```

Figura 16: Código regla de comparación similitud de imágenes para despliegue

```
def abrir_archivo():
    seleccion = tk.Toplevel(ventana)
    seleccion.title("REFY - Robot de Estimación Yanbal")
    seleccion.geometry("518x438+0+0")
    labelExample = tk.Label(seleccion, text = "Seleccione joyas para estimar:",font=myFont1).place
    btn4 = tk.Button(seleccion, text='Realizar Estimación',command = estimar, width=20, height=1,
```

Figura 17: Código para el manejo de ventana con imágenes de joyas

```
for i in range(0, len(Imagenes)):
    archivo = Image.open(str(Imagenes.iloc[i]["direccion"]))
    nueva = archivo.resize((100,100))
    nueva.save('imagenes_mostrar/'+str(Imagenes.iloc[i]["nombre"])+'.gif', 'gif')
    imagenL12=PhotoImage(file='imagenes_mostrar/'+str(Imagenes.iloc[i]["nombre"])+'.gif')
    #img1
    if i==0:
        lblImagen21=tk.Label(seleccion,image=imagenL12).place(x=20,y=50)
        chk1 = Checkbutton(seleccion, text=str(Imagenes.iloc[i]["nombre"]), variable=jj1).plac
        lbljy1 = Label(seleccion, text="Similitud: "+str(Imagenes.iloc[i]["Similitud"])+"%", fo
```

Figura 18: Código para selección de imágenes y checks con estado.

## 2.7.6 Capturas de pantalla del incremento del segundo Sprint

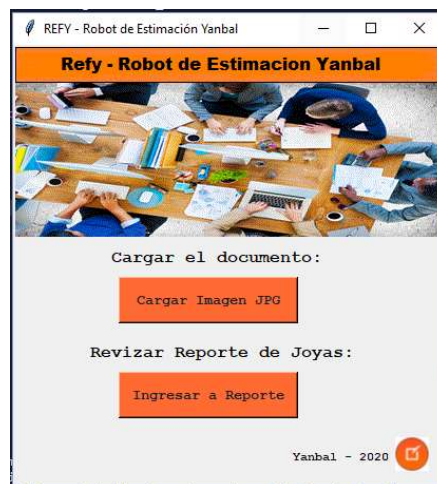


Figura 19: Ejecutable para Selección de Archivo

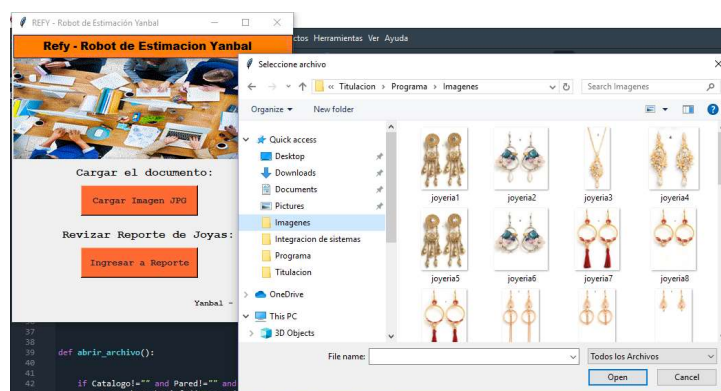


Figura 20: Carga de archivos programa solo permite archivos jpg



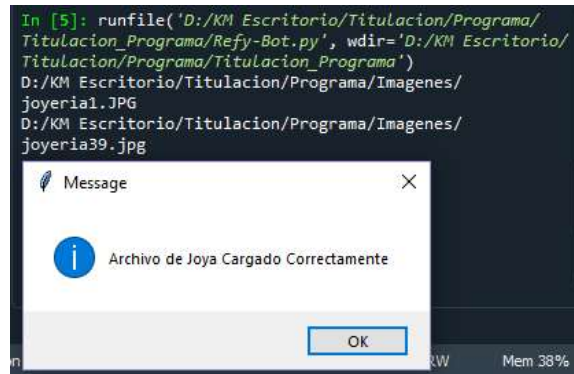


Figura 21: Impresión de mensaje para confirmar existencia de un archivo.

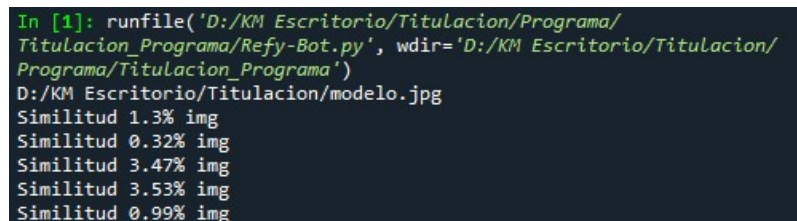


Figura 22: Impresión de porcentaje de similitud por comparación de imagen

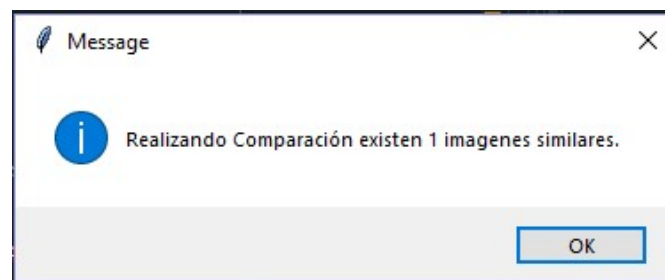


Figura 23: Mensaje con cantidad de imágenes de mayor similitud

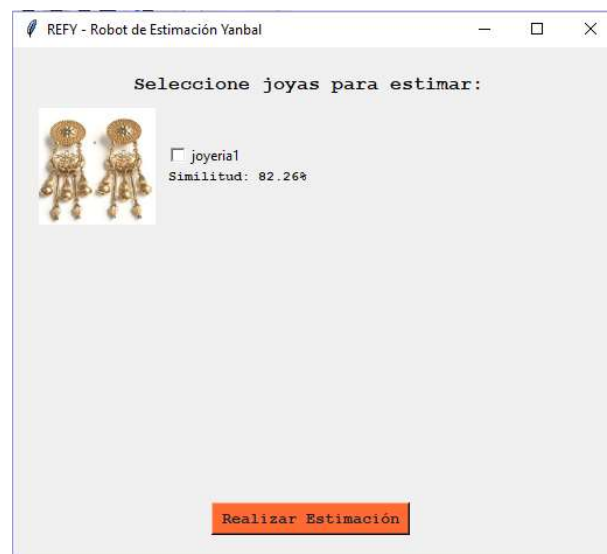


Figura 24: Despliegue de imágenes para selección y porcentaje de similitud

### 2.7.7 Sprint Retrospective

Para la carga de imágenes y consulta de similitud se realiza un recorrido de los archivos por lo que implemento que solo obtenga archivos jpg y se los transforma a gif sin afectar la calidad de la imagen. En el selector de imagen para el manejo de los check se tiene que transformar la variable entero o cadena de caracteres ya que si se la compara o imprime sin transformar los datos que muestra son coordenadas.

## 2.8 Tercer Sprint

### 2.8.1 Sprint Planning

Aplicando la técnica Planning Poker se pudo estimar y seleccionar las historias de usuario que van a ser desarrolladas en el segundo Sprint, considerando las opiniones de todo el equipo, así como también permitir que los integrantes tengan más compromiso con el proyecto, para este primer Sprint se seleccionaron las historias de usuario 5.

### 2.8.2 Sprint Backlog

En el Sprint Backlog del segundo Sprint se encuentran listadas las historias de usuario con identificador 5 y que están descritas en la tabla 4 ya que estas se encuentran agrupadas por actividades que permiten generar un incremento

A continuación, se muestra la tabla 9 correspondiente al Sprint Backlog del primer Sprint, en el cual se detalla las historias seleccionar para realizar en el segundo Sprint.

*Tabla 9: Sprint Backlog del Tercer Sprint*

ID	Nombre	Descripción	Estado	Esfuerzo	Prioridad	Responsable
5	Reporte de Estimación	El usuario deberá generar los estimados que se necesite para tener los promedios de venta, podrá guardar la estimación validando que no exista una previa y visualizara el rendimiento del sistema contra datos reales en un reporte generado a base del histórico y DataMark	Completado	8	Media	Karla Mafla

### 2.8.3 Daily Scrum

Cada día el equipo se reunió durante 15 minutos con el fin de responder a las 3 inquietudes, ¿Qué se hizo ayer?, ¿Qué se va a hacer hoy? y ¿Qué problemas se presentaron? Una vez respondidas las 3 interrogantes se procede a actualizar la gráfica Burn Down de Sprint, para poder observar el comportamiento y avance logrado.

Se realizó el manejo de datos para la estimación por medio de dos botones uno solo para hacer el cálculo y el otro para realizar el registro del forecast y el guardado de la imagen en el repositorio, se da al usuario la opción de poder volver a seleccionar las imágenes de las estimaciones sin tener que cerrar el programa acuerdo que se llega con el área de marketing.

### 2.8.4 Sprint Review

Al finalizar este Sprint se llevó a cabo un Sprint Review, en donde revisó lo que se trabajó y el incremento presentado que aporta valor para el cliente, además se obtienen conclusiones, recomendaciones y experiencias que fueron compartidos con el equipo.

En este sprint review se debió tener en cuenta para él envió de los datos al cálculo como es una ventana secundaria no se los declaro como datos globales se usó una función con parámetros y para la conexión a la base de datos se usó la clase pyodbc que sirve para conexiones de instancias exclusivas de Sql

### 2.8.5 Código segundo Sprint

```
#conexion base de datos
cnxn = pyodbc.connect("DRIVER={SQL Server};SERVER=DESKTOP-V920L86\SQLEXPRESS;DATABASE=Tesis;UID=sa;PWD=root")
cursor = cnxn.cursor()
```

Figura 25: Código conexión a base de datos

```
if i==0:
    archivo = Image.open(str(Imagenes.iloc[i]["direccion"]))
    nueva = archivo.resize((100,100))
    nueva.save('imagenes_mostrar/'+str(Imagenes.iloc[i]["nombre"])+'.gif', 'gif')
    imagenL10=PhotoImage(file='imagenes_mostrar/'+str(Imagenes.iloc[i]["nombre"])+'.gif')
    lblImagen21=tk.Label(seleccion, image=imagenL10).place(x=20,y=50)
    chk1 = Checkbutton(seleccion, text=str(Imagenes.iloc[i]["nombre"]), variable=jy1).place(x=130,y=80)
    lbljy1 = Label(seleccion, text="Similitud: "+str(Imagenes.iloc[i]["Similitud"])+"%", font=myFont3).place(x=130
```

Figura 26: Código para transformación de imagen de jpg a gif para la selección



```

for ex in range(0, len(Imagenes)):
    if ex==0 and jy1.get()==1:
        print(str(Imagenes.iloc[ex]["nombre"]));
        query1 = "select * from Historico where Nombre='"+str(Imagenes.iloc[ex]["nombre"])+"'";
        datos=cursor.execute(query1)
        for row in datos:
            Datos_Estimacion.loc[contador] = [row.Nombre]+[row.unidades]+[row.costo_unitario]+[row.venta]
            contador=contador+1;
        cnxn.commit()

```

Figura 27: Código para obtención de datos y almacenamiento en dataframe

```

for u in range(0, len(Datos_Estimacion)):
    io=u+1;
    # "Nombre", "Unidades_V", "Costo_C/U", "Venta_Total", "Campania", "Anio", "Exposicion"
    l = Label(estimar, text=str(Datos_Estimacion.iloc[u]["Nombre"]).strip(), relief=RIDGE)
    l.grid(row=io, column=0, sticky=NSEW)
    l1 = Label(estimar, text=str(Datos_Estimacion.iloc[u]["Unidades_V"]).strip(), relief=RIDGE)
    l1.grid(row=io, column=1, sticky=NSEW)
    l2 = Label(estimar, text=str(Datos_Estimacion.iloc[u]["Costo_C/U"]).strip(), relief=RIDGE)
    l2.grid(row=io, column=2, sticky=NSEW)
    l3 = Label(estimar, text=str(Datos_Estimacion.iloc[u]["Venta_Total"]).strip(), relief=RIDGE)
    l3.grid(row=io, column=3, sticky=NSEW)
    l4 = Label(estimar, text=str(Datos_Estimacion.iloc[u]["Campania"]).strip(), relief=RIDGE)
    l4.grid(row=io, column=4, sticky=NSEW)
    l5 = Label(estimar, text=str(Datos_Estimacion.iloc[u]["Anio"]).strip(), relief=RIDGE)
    l5.grid(row=io, column=5, sticky=NSEW)
    l6 = Label(estimar, text=str(Datos_Estimacion.iloc[u]["Exposicion"]).strip(), relief=RIDGE)
    l6.grid(row=io, column=6, sticky=NSEW)

```

Figura 28: Código para creación de tabla y muestra de datos en interfaz Grafica

```

txt1 = Entry(estimar, width=20)
txt1.place(x=15, y=170)

txt2 = Entry(estimar, width=15)
txt2.place(x=160, y=170)

txt3 = Entry(estimar, width=15)
txt3.place(x=260, y=170)

txt4 = Entry(estimar, width=15)
txt4.place(x=360, y=170)

cmb1=ttk.Combobox(estimar, values=["Baja", "Media", "Alta"])
cmb1.place(x=470, y=170)

```

Figura 29: Código manejo campos ingresados por usuario

```

camp=cmb1.get()
campindice=cmb1.current()

est1 = Button(estimar, text='Calcular Estimación', command = lambda: calculo(nombre_joya, txt1.get(), t
est2 = Button(estimar, text='Registrar Estimación', command = lambda: historial(nombre_joya, txt1.get(
estimar.mainloop()

```

Figura 30: Código botones para calculo y registro de estimación

```

calculo(nombre, costo, u_venta, campania, anio, exposicion):
unidadesvendidad=0;
ventatotal=0;
if nombre!="" and costo!="" and u_venta!="" and campania!="" and anio!="" and exposicion!="":
    if exposicion == "Baja":
        unidadesvendidad=round(float(u_venta)*0.85,0)
        ventatotal=round(float(costo)*unidadesvendidad,2)
        messagebox.showinfo('Estimacion', 'Las Unidades de vendeta seran: '+str(unidadesvendidad)+'
    if exposicion == "Media":
        unidadesvendidad=round(float(u_venta)*0.90,0)
        ventatotal=round(float(costo)*unidadesvendidad,2)
        messagebox.showinfo('Estimacion', 'Las Unidades de vendeta seran: '+str(unidadesvendidad)+'
    if exposicion == "Alta":
        unidadesvendidad=round(float(u_venta)*0.95,0)
        ventatotal=round(float(costo)*unidadesvendidad,2)
        messagebox.showinfo('Estimacion', 'Las Unidades de vendeta seran: '+str(unidadesvendidad)+'
else:
    messagebox.showinfo('Message', 'Falta un campo se encuentra vacio')

```

Figura 31: Código de la función Calcular Estimación

```

historial(nombre, costo, u_venta, campania, anio, exposicion):
unidadesvendidad=0;
ventatotal=0;
if nombre!="" and costo!="" and u_venta!="" and campania!="" and anio!="" and exposicion!="":
    if exposicion == "Baja":
        unidadesvendidad=round(float(u_venta)*0.85,0)
        ventatotal=round(float(costo)*unidadesvendidad,2)
        #messagebox.showinfo('Estimacion', 'Las Unidades de vendeta seran: '+str(unidadesvendidad)+' la
    if exposicion == "Media":
        unidadesvendidad=round(float(u_venta)*0.90,0)
        ventatotal=round(float(costo)*unidadesvendidad,2)
        # messagebox.showinfo('Estimacion', 'Las Unidades de vendeta seran: '+str(unidadesvendidad)+' la
    if exposicion == "Alta":
        unidadesvendidad=round(float(u_venta)*0.95,0)
        ventatotal=round(float(costo)*unidadesvendidad,2)
        #messagebox.showinfo('Estimacion', 'Las Unidades de vendeta seran: '+str(unidadesvendidad)+' la

    insert1="INSERT INTO Historico(Nombre, unidades, costo_unitario, venta, campania, anio, exposicion)VALUES
    datos=cursor.execute(insert1)
    cnxn.commit()

    archivo = Image.open(Catalogo)
    archivo.save('D:/KM Escritorio/Titulacion/Programa/Imagenes/'+nombre+'.jpg')
    messagebox.showinfo('Historial', 'Datos de Joya '+nombre+' Ingresados sin Problema.')
else:
    messagebox.showinfo('Message', 'Falta un campo se encuentra vacio')

```

Figura 32: Código de la función Registrar Estimación

## 2.8.6 Capturas de pantalla del incremento del segundo Sprint

REFY - Robot de Estimación Yanbal

NOMBRE PRODUCTO	COSTO C/U	UNIDADES VENDIDAS	VENTA TOTAL	CAMPAÑA	AÑO DE VENTA	EXPOSICIÓN
joyeria1	1275655	26.3	84552.5	C1	2019	alta
joyeria5	1247655	26.3	84542.5	C1	2019	alta

UNIDADES DE VENTA:   
 COSTO C/U:   
 CAMPAÑA:   
 AÑO DE VENTA:   
 EXPOSICIÓN:

Calcular Estimación Registrar Estimación

Figura 33: Modulo de estimación

REFY - Robot de Estimación Yanbal

Seleccione joyas para estimar:

joyeria1 Similitud: 82.26%

joyeria5 Similitud: 100.0%

Realizar Estimación

NOMBRE PRODUCTO	COSTO C/U	UNIDADES VENDIDAS	VENTA TOTAL	CAMPAÑA	AÑO DE VENTA	EXPOSICIÓN
joyeria1	1275655	26.3	84552.5	C1	2019	alta
joyeria5	1247655	26.3	84542.5	C1	2019	alta

UNIDADES DE VENTA:   
 COSTO C/U:   
 CAMPAÑA:   
 AÑO DE VENTA:   
 EXPOSICIÓN:

Calcular Estimación Registrar Estimación

Figura 34: Vista de modulo estimación y selección

REFY - Robot de Estimación Yanbal

NOMBRE PRODUCTO	COSTO C/U	UNIDADES VENDIDAS	VENTA TOTAL	CAMPAÑA	AÑO DE VENTA	EXPOSICIÓN
joyeria1	1275655	26.3	84552.5	C1	2019	alta
joyeria5	1247655	26.3	84542.5	C1	2019	alta

UNIDADES DE VENTA:   
 COSTO C/U:   
 CAMPAÑA:   
 AÑO DE VENTA:   
 EXPOSICIÓN:

Calcular Estimación Registrar Estimación

Estimacion

Las Unidades de vendeta seran: 18889.0 la venta total estimada: 43444.7

OK

Figura 35: Mensaje de estimación del botón calculo

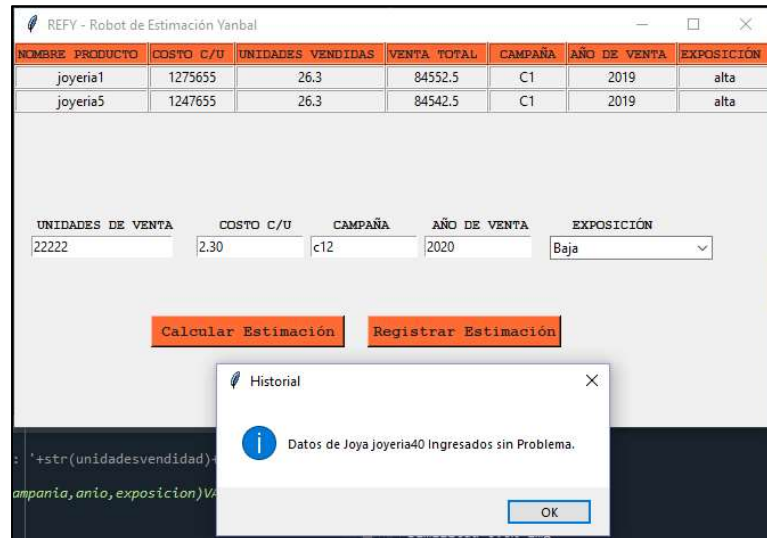


Figura 36: Mensaje confirmación de guardado de datos e imagen en DB

### 2.8.7 Sprint Retrospective

En este sprint se dio una mayor atención a las cadenas de conexión y el manejo de las imágenes ya que se obtuvo el problema que se bajaba la calidad de la imagen al ser guardada, se solucionó el problema guardando la imagen como puntos lo que da como resultado una nueva imagen y no una copia de la imagen seleccionada.

El manejo de ventanas secundarias y obtención de datos entre ventanas es algo que se debe tener cuidado ya que si la ventana está dentro de una función no se puede acceder a los campos específicos desde afuera de la función o desde otra función.

### 2.9 Medición y Estimación

Para realizar una medición (evaluación) del progreso en el desarrollo de la solución con el marco de trabajo Scrum se usarán gráficos Burn Down y Burn Up estos gráficos permiten visualizar el avance estimado contra el avance real que debe tener el equipo de trabajo.



## 2.9.1 Gráfico Burn Up

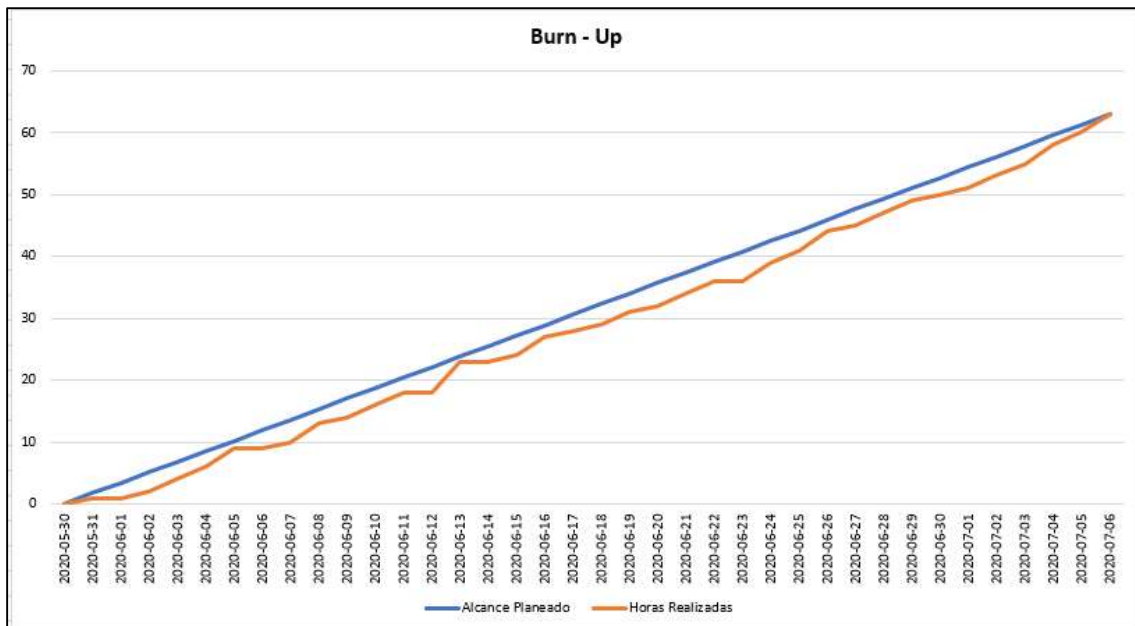


Figura 37: Grafico Burn Up Final

## 2.10 Gráficos Burn Down

### 2.10.1 Gráfico Burn Down Primer Sprint



Figura 38: Burn Down primer Sprint

### 2.10.2 Gráfico Burn Down Segundo Sprint

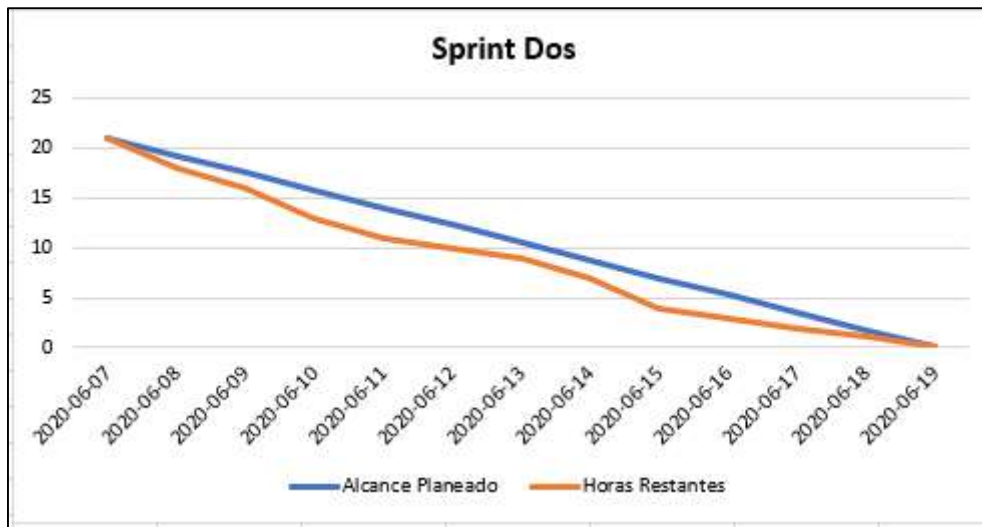


Figura 39: Burn Down segundo Sprint

### 2.10.3 Gráfico Burn Down Tercer Sprint

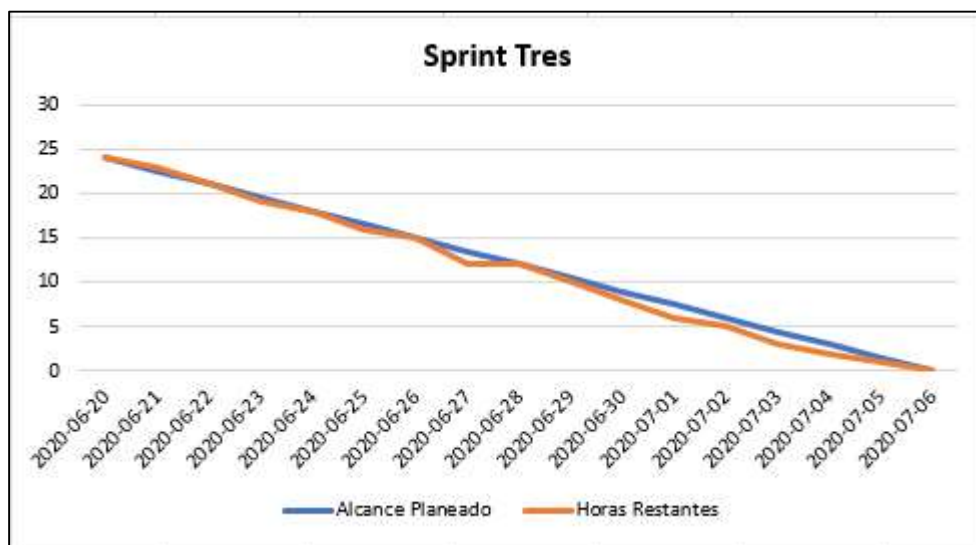


Figura 40: Burn Down tercer Sprint

### 3. CAPÍTULO 3. Validación de Cliente

#### 3.1 Introducción

En este capítulo, se expondrá las pruebas de validación del cliente para la aplicación de estimación joyería, para mejorar el rendimiento y la interactividad de la aplicación.

Las pruebas realizadas para validar la aplicación fueron, unitarias y de aceptación del cliente, estas pruebas se ejecutaron por el producto owner y a pedido de Yanbal por los analistas del área comercial los cuales dieron un feedback sobre cambios y mejoras a la aplicación los cuales serán documentados y presentados en este capítulo.

#### 3.2 Pruebas de aceptación del cliente

Las pruebas de aceptación del cliente y analistas están detalladas en la tabla 10 y representan a las historias de usuario detallada anteriormente. La tabla de pruebas contiene las siguientes columnas:

- ID: Código de prueba.
- Descripción: Detalle de lo que debe pasar en la prueba.
- Resultado: Resultado de la prueba si es completada con éxito “Correcto” en caso de un problema “Fallido”.

Tabla 10: Pruebas de aceptación del cliente

ID	Descripción	Resultado
1	Carga de información e imágenes sobre la joyería con validación de formato imagen en estilo jpg y validación de que sea un nombre único.	Correcto
2	Comparación de imagen con el repositorio histórico y obtención de mensaje con el número de imágenes que tienen similitud.	Correcto

3	Despliegue de pantalla con las imágenes similares por cada imagen debe aparecer el nombre, similitud y un check para estimación	Correcto
4	Captura de imágenes seleccionadas para estimación y búsqueda de datos en base por el nombre de la joya para despliegue posterior en una tabla.	Correcto
5	Manejo de datos para calculo y registro de estimación junto a la validación de registro en la información y mensajes de confirmación por calculo.	Correcto

### 3.2.1 Demostración de las pruebas de aceptación del cliente

#### Prueba 1

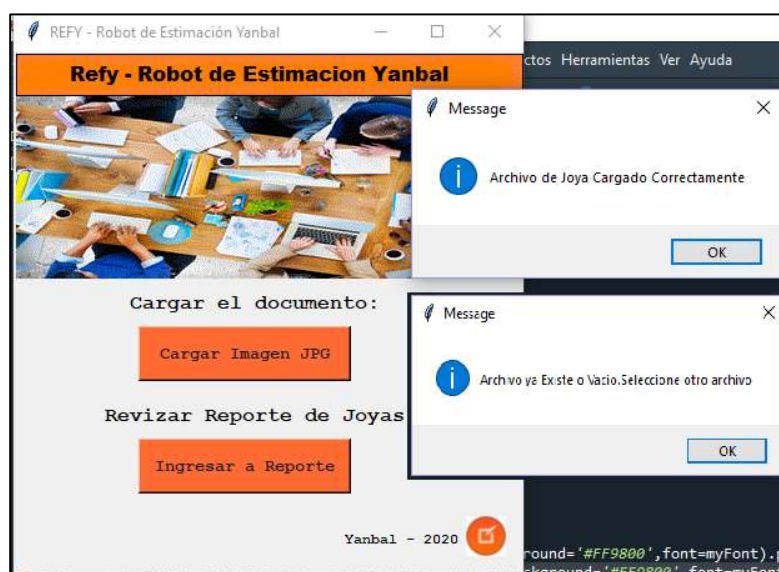


Figura 41: Validación de existencia y formato

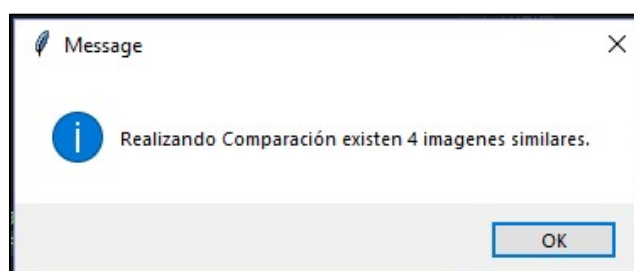


Figura 42: Confirmación de imágenes con similitud



## Prueba 2

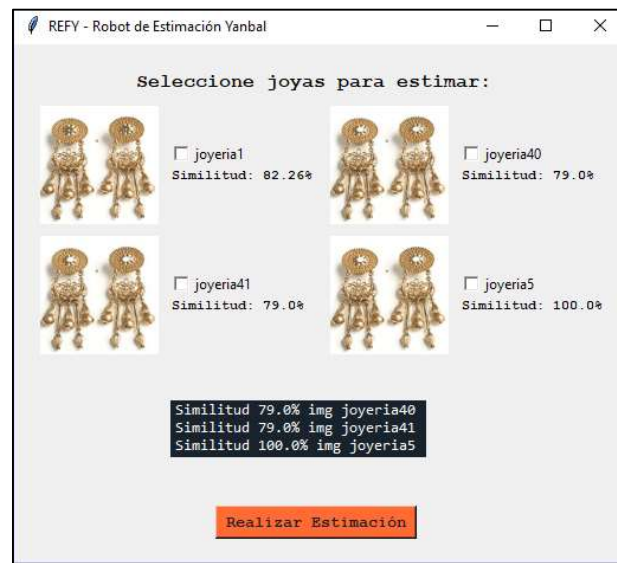


Figura 43: Vista de imágenes según resultado de comparación y check para estimacion

## Prueba 3



Figura 44: Obtención de Información sobre imágenes seleccionadas

## Prueba 4

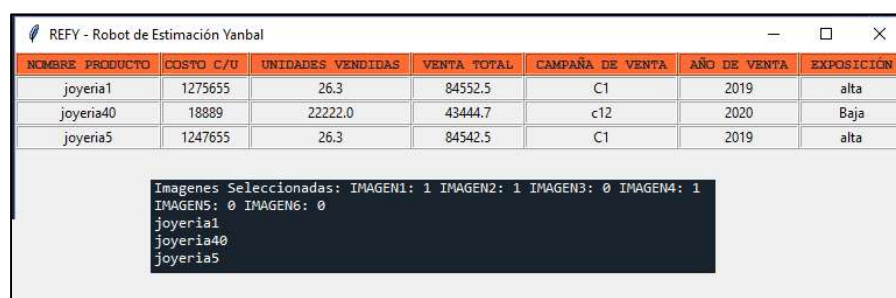


Figura 45: Impresión de check y muestra en tabla de datos recuperados

## Prueba 5

REFY - Robot de Estimación Yanbal

NOMBRE PRODUCTO	COSTO C/U	UNIDADES VENDIDAS	VENTA TOTAL	CAMPAÑA DE VENTA	AÑO DE VENTA	EXPOSICIÓN
joyeria1	1275655	26.3	84552.5	C1	2019	alta
joyeria40	18889	22222.0	43444.7	c12	2020	Baja
joyeria5	1247655	26.3	84542.5	C1	2019	alta

UNIDADES DE VENTA:

COSTO C/U:

CAMPAÑA DE VENTA:

AÑO DE VENTA:

EXPOSICIÓN:

Calcular Estimación Registrar Estimación

Figura 46: Ingreso datos para estimación

REFY - Robot de Estimación Yanbal

NOMBRE PRODUCTO	COSTO C/U	UNIDADES VENDIDAS	VENTA TOTAL	CAMPAÑA DE VENTA	AÑO DE VENTA	EXPOSICIÓN
joyeria1	1275655	26.3	84552.5	C1	2019	alta
joyeria40	18889	22222.0	43444.7	c12	2020	Baja
joyeria5	1247655	26.3	84542.5	C1	2019	alta

UNIDADES DE VENTA:

COSTO C/U:

CAMPAÑA DE VENTA:

AÑO DE VENTA:

EXPOSICIÓN:

Estimación

Las Unidades de vendeta seran: 2209366.0 la venta total estimada: 74013761.0

OK

Figura 47: Impresión Calculo

REFY - Robot de Estimación Yanbal

NOMBRE PRODUCTO	COSTO C/U	UNIDADES VENDIDAS	VENTA TOTAL	CAMPAÑA DE VENTA	AÑO DE VENTA	EXPOSICIÓN
joyeria1	1275655	26.3	84552.5	C1	2019	alta
joyeria40	18889	22222.0	43444.7	c12	2020	Baja
joyeria5	1247655	26.3	84542.5	C1	2019	alta

UNIDADES DE VENTA:

COSTO C/U:

CAMPAÑA DE VENTA:

AÑO DE VENTA:

EXPOSICIÓN:

Historial

Datos de Joya joyeria42 Ingresados sin Problema.

OK

Figura 48: Impresión Registro Estimación

### 3.3 Cambios propuestos por cliente

En las pruebas de usuario, el producto owner dio la aprobación los analistas de marketing que serían los usuarios principales de la aplicación pidieron que la aplicación tenga cambios adicionales que fueron realizados con éxito a continuación se detallarla la tabla con los cambios pedidos.

La tabla 11 de pruebas contiene las siguientes columnas:

- ID: Código de prueba.
- Descripción: Detalle de lo que debe pasar en la prueba.
- Aplicado: Muestra si el cambio se realizó con éxito “Exitoso” en caso de poder haberse aplicado “Fallido”.

Tabla 11: Cambios del cliente

ID	Descripción	Aplicado
1	Impresión de porcentaje de similitud con nombre de imagen en repositorio.	Exitoso
2	Combobox de campañas para estimación	Exitoso
3	Cambio en formula de estimación para que tenga porcentaje según campaña.	Exitoso
4	Combobox de año con carga de años hasta el año actual desde 2015	Exitoso
5	Impresión de los check seleccionados con el detalle de imágenes	Exitoso

### 3.3.1 Captura de cambios del cliente

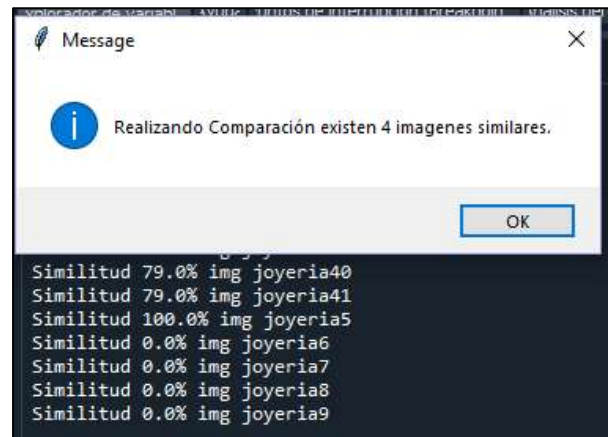


Figura 49: Impresión de nombre imagen y porcentaje de similitud

```

Imágenes Seleccionadas: IMAGEN1: 1 IMAGEN2: 0 IMAGEN3: 0 IMAGEN4: 1
IMAGEN5: 0 IMAGEN6: 0
joyeria1
joyeria5
  
```

Figura 50: Impresión de imágenes seleccionadas y nombre de las imágenes

UNIDADES DE VENTA	COSTO C/U	CAMPAÑA DE VENTA	AÑO DE VENTA	EXPOSICIÓN
<input type="text"/>	<input type="text"/>	<input type="text" value="v"/>	<input type="text" value="v"/>	<input type="text" value="v"/>
<input type="button" value="Calcular Estimación"/>		<input type="button" value="Registrar Estimación"/>		

Figura 51: Implementación de combobox para campaña y año

## 4. CAPÍTULO 4. Rendimiento Sistema

### 4.1 Introducción

En este capítulo, se expondrá las pruebas unitarias para la aplicación de estimación joyería, se realizarán mediciones de carga y estrés sobre el sistema las cuales permitirán determinar si existe una sobre exigencia en la aplicación

Los modelos que se usara en este capítulo permitirán examinar el código y rendimiento de sistema, esto permitirá que cuando el sistema sea desplegado tengo un manejo optimo y con menos errores.

### 4.2 Pruebas unitarias

La definición de pruebas unitarias se trata de demos o manejo de una parte específica del software para verificar que esté funcionando de manera correcta y en caso de que genere errores poder prevenir esos errores y mejorar el código, estas pruebas son realizadas por el lado técnico y no por los requerimientos de usuarios, al realizar estas pruebas estamos verificando y mejorando la aplicación de manera que los usuarios consigan la menor cantidad de errores.

En la siguiente tabla 12 se detallan las pruebas realizadas para verificar si un error esta validado y si está controlado o no:

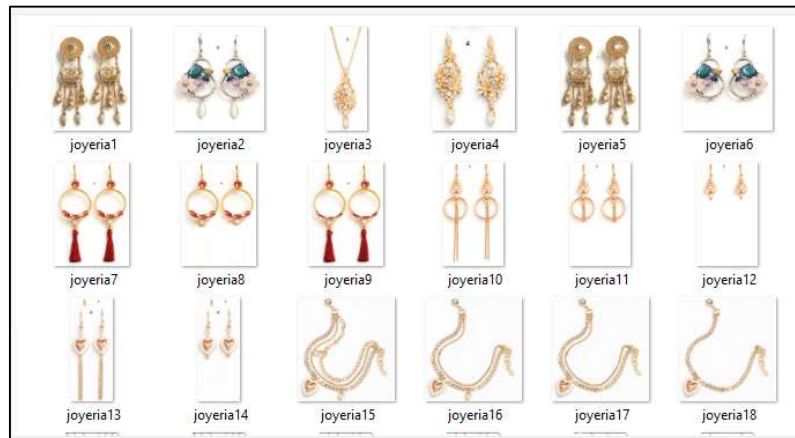
Tabla 12: Pruebas unitarias.

<b>Id</b>	<b>Detalle</b>	<b>Estado</b>
1	Carga de imágenes solo formato jpg	Controlado
2	Validación imagen única	Controlado
3	Validar selección de imágenes	Controlado
4	Validar campos de número y costo a valor número y no alfabético	Controlado
5	Validación de registro información base de datos	Controlado

### 4.2.1 Demostración del cumplimiento de las pruebas unitarias

A continuación, se muestran las imágenes de la ejecución y validación de las pruebas unitarias.

#### Prueba 1



```
In [1]: runfile('D:/KM Escritorio/Titulacion/Programa/  
Titulacion_Programa/Refy-Bot.py', wdir='D:/KM Escritorio/Titulacion/  
Programa/Titulacion_Programa')  
D:/KM Escritorio/Titulacion/Programa/joyeria44.JPG
```

Figura 52: validación de imágenes formato jpg

#### Prueba 2

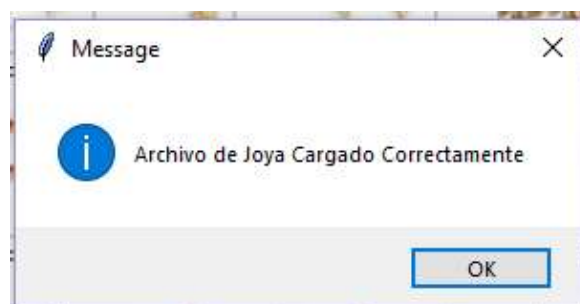
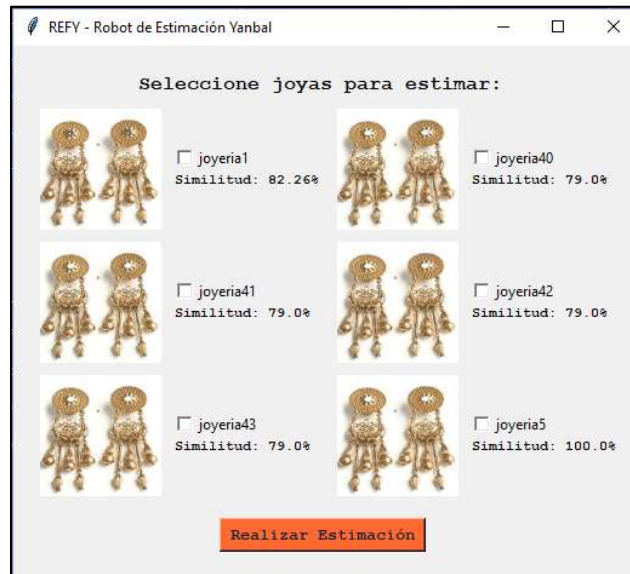


Figura 53: validación imagen única

### Prueba 3



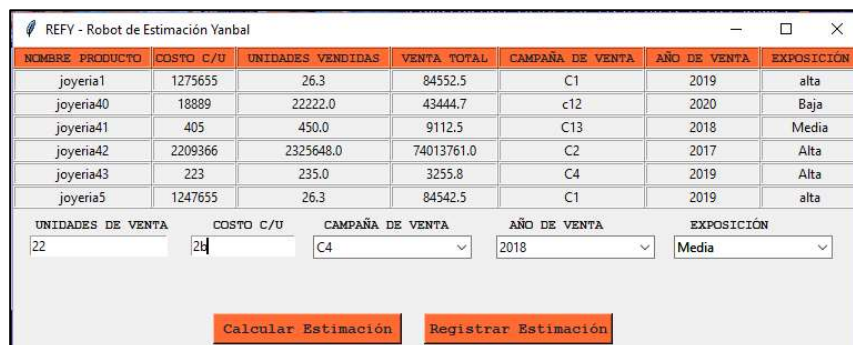
```

Imagenes Seleccionadas: IMAGEN1: 1 IMAGEN2: 1 IMAGEN3: 1 IMAGEN4: 1
IMAGEN5: 1 IMAGEN6: 1
joyeria1
joyeria40
joyeria41
joyeria42
joyeria43
joyeria5

```

Figura 54: validación selección de imágenes similares

### Prueba 4



```

activebackground='#FF9800',font=myFont).place(x=180,y=240)
File "D:\KM Escritorio\Titulacion\Programa\Titulacion_Programa\Refy-
Bot.py", line 360, in calculo
    ventatotal=round(float(u_venta)*unidadesvendidad,2)
ValueError: could not convert string to float: '2b'

```

Figura 55: validación solo aceptación números

## Prueba 5

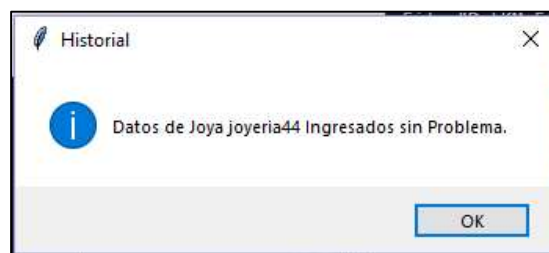


Figura 56: validación ingreso a base de datos

### 4.3 Pruebas de carga

Se realizarán pruebas de carga ya que el sistema desarrollado es una aplicación de escritorio y no hay cuellos de botella que se generen, por el sistema, pero se medirá las respuestas del sistema y la rapidez de reacción sobre dos tipos de ambientes uno directo a los computadores y otro a través de una conexión a máquina remota ya que estos son los dos medios en los cuales se colocará la aplicación para su uso.

Para obtener las estadísticas de pruebas se realizaron cargas continuas con nueve usuarios que serían los que generaran las peticiones al sistema y se medirán los tiempos de respuesta para cada usuario, esto permitirá generar una estadística y promedio de rendimiento del sistema que se dio como 350 ms para generar una respuesta.

En la tabla 8 se puede ver los resultados de las pruebas de carga realizadas por los usuarios.

Tabla 13: Pruebas de Carga

Prueba	Respuesta Remota	SLA	Usuario
Test 1	150	300	326
Test 2	203	300	256
Test 3	260	300	206
Test 4	154	300	102
Test 5	320	300	325
Test 6	532	300	126
Test 7	420	300	420
Test 8	321	300	436
Test 9	360	300	459



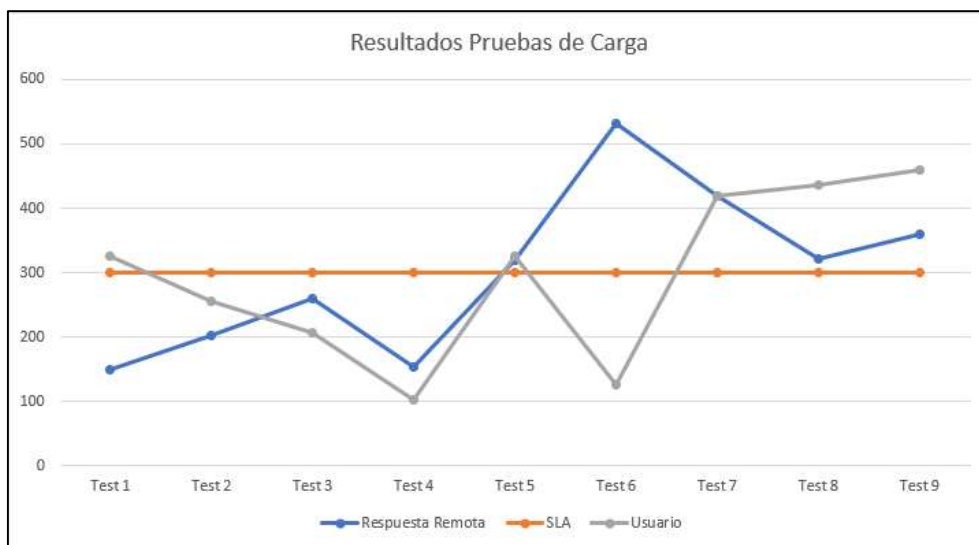


Figura 57: Grafico resultado pruebas de carga

## Conclusiones y recomendaciones

### Conclusiones

- Después de analizar el proceso de estimación y en base a las historias de los usuarios se determinó que el desarrollo de la aplicación brinda un gran apoyo y optimización de tiempo teniendo en cuenta que se está realizando las estimaciones que antes demoraban 3 horas en 20 min lo cual da una reducción del tiempo empleado en más de 100%.
- El sistema genera un rendimiento más alto sobre la información histórica de la venta de joyas y permite un rendimiento de la información mediante la comparación de las imágenes y la generación de reportes de estimación en base a las similitudes de las imágenes a analizar para la comparación.
- Después de realizar las pruebas funcionales y de carga de la aplicación se determinó que los usuarios tendrían mayor facilidad para realizar las estimaciones y disminuir los errores que se pueden causar por manejo humano.
- La metodología ágil Scrum permite realizar un desarrollo más eficaz y rápido sobre los sistemas generando un alcance funcional por cada sprint, de esta manera se proporciona un entregable para que los usuarios pueden probar, y al final del sprint recibir los comentarios y cambios al sistema.

- Python, es un lenguaje de programación que no solo se basa en objetos permite crear sistemas que aprenden sobre la ejecución y pueden identificar las similitudes entre objetos ya sean imágenes en vivo o fotos esto es lo que permite tener programas de reconocimiento o similitud.
- Anaconda es la plataforma donde se permite editar, probar el código de Python ya que normalmente se utiliza editores de texto para crear programas de py, para este desarrollo se utilizó la plataforma Spyder, la cual permite tener vista de las variables, errores y depuración del sistema.

### **Recomendaciones**

- Se recomienda para el uso correcto del sistema que se genere para las siguientes versiones un ayudante virtual que pueda gestionar las comparaciones y generar un documento con los cálculos.
- Se recomienda realizar la implementación de la aplicación en un módulo web para tener un mayor espacio de uso en pantalla y vistas y generar el manejo por usuarios.
- Para el manejo de pruebas de funcionalidad se recomienda una interfaz más amigable y atractiva para el uso de la aplicación.
- Se recomienda tener en cuenta el manejo masivo de la información y los datos de las imágenes.
- Generar acceso a la base de datos y manejo de sentencias sql con los stored procedure.
- Se recomienda que se genere un módulo para las áreas de perfumes y maquillajes que debe adaptarse a las características de las categorías y permita facilitar el trabajo de estimación.
- Se recomienda que la aplicación sea montada en un servidor ya que podrá ser usada de una manera más eficiente y permitirá a los usuarios tener el manejo de claves y sesiones para la difusión a los demás países de Yanbal.
- Se recomienda utilizar un robot el cual generaría las comparaciones y estadísticas sobre los reportes de venta contra el rendimiento de la campaña mensual, de esta manera se notificará sobre las estadísticas para impulsar el producto que son las más eficientes.

## Referencias

Contributors, s (2018). Spyder-ide. Recuperado 29 de abril de 2020 de: <https://www.spyder-ide.org>

Directa, A. D. (2019). Avd.es. Recuperado 25 de mayo de 2020 de: <https://avd.es/la-venta-directa/>

Foundation, P. S. (2020). Python.org. Recuperado 25 de mayo de 2020 de: <https://www.python.org/doc/>

Gutierrez, J. (2019). Isi, Recuperado 15 de mayo de 2020 de: [http://www.isi.us.es/~javierj/investigacion\\_ficheros/Framework.pdf](http://www.isi.us.es/~javierj/investigacion_ficheros/Framework.pdf)

IONOS. (2018). Digital Guide. Recuperado 18 de abril de 2020 de: <https://www.ionos.es/digitalguide/servidores/know-how/file-server/>

Mestras, J. (2018). Fdi. Recuperado 15 de abril de 2020 de: <https://www.fdi.ucm.es/profesor/jpavon/poo/2.14.MVC.pdf>

Microsoft. (2020). Microsoft. Recuperado 27 de mayo de 2020 de: <https://products.office.com/es/excel>

Quiroa, M. (2017). Economipedia. Recuperado 27 de mayo de 2020 de: <https://economipedia.com/definiciones/venta-directa.html>

Schawaker, K., & Sutherland, J. (2016). La Guía de Scrum. Recuperado 27 de 2020 de: <https://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-es.pdf>

Yanbal. (2017). Yanbal EC. Recuperado 15 de abril de 2020 de: <https://www.yanbal.com/ec>

Zimmer, F. (2020) GitLab. Recuperado 10 de mayo de 2020 de: <https://about.gitlab.com/>

Alejandro (Rodas de Paz), A. (2015). Python game programming by example. Lugar (Birmingham,UK): Packt Publishing Ltd.

Sebastian (Raschka), S. (2017). Python machine learning: machine learning and deep learning with python, scikit-learn, and tensorflow. Lugar (Birmingham,UK): Packt Publishing Ltd.

Nigel (George), N. (2017). Build your first site with Python and Django: go from zero knowledge to your own site using the easiest to learn programming language on the Internet. Lugar (Seattle,USA): GNW Independent Publishing.

Charles (Severance), C. (2009). Python para informaticos. Lugar (Mexico,D.F.,Mexico): Creative Common Attribution-NonCommercial-ShareAlike.

