

“AÑO DE LA UNIVERSALIZACIÓN DE LA SALUD”



**Universidad  
Continental**

**EAP DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

**SISTEMA DE VENTA DE PRODUCTOS PARA LA EMPRESA  
FUNERARIA RODRÍGUEZ**

**INTEGRANTES:**

- . Arauco Medrano Merly
- . Condori Ochoa Brady
- . Cordova Limaylla Sergio

**DOCENTE:**

Guevara Jiménez Jorge

**PERIODO ACADEMICO:**

2020-I

HUANCAYO - 2020

## **AGRADECIMIENTOS**

A nuestros maestros quienes nos impartieron nuevos conocimientos, por adaptarse al nuevo método de enseñanza y la paciencia que presentan ante los estudiantes, por lo cual se logró culminar de manera satisffecha el presente proyecto

### **DEDICATORIA**

A nuestros padres que siempre nos brindan su apoyo constante e incondicional, por avernós inculcados los valores, su enseñanza de buenas costumbres y lo más importante que es para nosotros la responsabilidad.

## INDICE

|  |           |
|--|-----------|
| <b>AGRADECIMIENTOS</b>                                 | <b>1</b>  |
| <b>DEDICATORIA</b>                                     | <b>2</b>  |
| <b>ÍNDICE DE GRÁFICOS</b>                              | <b>5</b>  |
| <b>ÍNDICE DE TABLAS</b>                                | <b>5</b>  |
| <b>CERTIFICADO DE REDACCIÓN, ORTOGRAFÍA Y ESTILO</b>   | <b>5</b>  |
| <b>ADECLARACIÓN DE AUTENTIFICIDAD</b>                  | <b>7</b>  |
| <b>RESUMEN EJECUTIVO</b>                               | <b>8</b>  |
| <b>ABSTRACT</b>  | <b>9</b>  |
| <b>CAPITULO I</b>                                      | <b>10</b> |
| <b>INTRODUCCIÓN</b>                                    | <b>10</b> |
| <b>1.1. PLANTEAMIENTO Y JUSTIFICACIÓN DEL PROBLEMA</b> | <b>10</b> |
| <b>1.2. DESCRIPCION GENERAL DE LA EMPRESA</b>          | <b>10</b> |
| <b>1.3. SITUACIÓN ACTUAL</b>                           | <b>11</b> |
| <b>1.4. PROBLEMAS</b>                                  | <b>11</b> |
| <b>1.5. JUSTIFICACIÓN</b>                              | <b>12</b> |
| <b>1.6. OBJETIVOS</b>                                  | <b>12</b> |
| • OBJETIVO GENERAL                                     | 12        |
| • OBJETIVOS ESPECIFICOS                                | 13        |
| <b>1.7. METODOLOGÍA DE LA INGESTIGACIÓN</b>            | <b>13</b> |
| <b>1.8. PASOS DE LA ELABORACIÓN DEL PROYECTO</b>       | <b>13</b> |
| <b>1.9. LIMITACIONES</b>                               | <b>14</b> |
| <b>1.10. ALCANCES DE LA INVESTIGACIÓN</b>              | <b>14</b> |
| <b>CAPITULO II</b>                                     | <b>15</b> |
| <b>MARCO TEÓRICO Y CONTEXTUAL</b>                      | <b>15</b> |
| <b>2.1. TECNICAS USADAS</b>                            | <b>15</b> |
| 2.1.1. Mapa de procesos                                | 15        |
| 2.1.2. Patrón de diseño modelo-vista-controlador       | 16        |
| 2.1.3. Procedimientos almacenados                      | 17        |
| 2.1.4. Desarrollo del ciclo de vida del software       | 17        |
| 2.1.5. Gestión de la configuración del Software (GUS)  | 18        |
| 2.1.6. Método FURPS                                    | 19        |
| 2.1.7. Evaluación CMMI                                 | 20        |

|   |           |
|---|-----------|
| <b>2.2. HERRAMIENTAS DEL MODELADO Y DISEÑO DE SISTEMAS</b>                              | <b>21</b> |
| 2.2.1. Herramientas del Modelado y Diseño de Sistemas                                   | 21        |
| 2.2.2. Framework de desarrollo: GitHub  | 21        |
| 2.2.3. Plataforma de Desarrollo   | 22        |
| 2.2.4. Motor de Base de Datos   | 22        |
| <b>2.2.5. Lenguaje de Programación</b>  | <b>23</b> |
| <b>2.2.6. Técnicas Temáticas</b>  | <b>24</b> |
| <b>CAPITULO III</b>   | <b>25</b> |
| <b>INGENIERIA DEL PRODUCTO O DE DESARROLLO DE LA INVESTIGACIÓN</b>                      | <b>25</b> |
| <b>3.1. MAPA DE PROCESOS</b>  | <b>25</b> |
| • <b>DIAGRAMA DE FLUJO INTEGRADO DE CADA FLUJO DE TRABAJO DE LA FUNERARIA RODRÍGUEZ</b> | <b>26</b> |
| <b>3.3 INVENTARIO DE REQUERIMIENTOS</b>   | <b>26</b> |
| <b>3.4 DESARROLLO INCREMENTAL</b>   | <b>29</b> |
| <b>3.5 GESTIÓN DEL PRODUCTO Y PROYECTO</b>  | <b>30</b> |
| • Planificación de Unidades de Programación   | 30        |
| • Procedimientos Almacenados  | 31        |
| • Modelo Lógico de Base de Datos  | 32        |
| • Plan de Prueba  | 33        |
| • Diagrama de Despliegue  | 34        |
| <b>3.6 DESARROLLO DEL CICLO DE VIDA DEL SOFTWARE</b>                                    | <b>35</b> |
| • Diseño de Prototipos  | 35        |
| <b>3.7 GESTIÓN DE LA CONFIGURACIÓN</b>  | <b>38</b> |
| • Código fuente en GitHub   | 38        |
| <b>3.8 EVIDENCIAS DEL CÓDIGO FUENTE EN CLASES MVC</b>                                   | <b>40</b> |
| <b>3.9 EJECUCIÓN DEL PLAN DE PRUEBAS</b>  | <b>45</b> |
| <b>CAPITULO IV</b>  | <b>53</b> |
| <b>CONCLUSIONES Y RECOMENDACIONES</b>   | <b>53</b> |
| <b>4.1. CONCLUSIONES</b>  | <b>53</b> |
| <b>4.2. RECOMENDACIONES</b>   | <b>54</b> |
| <b>ÍNDICE DE ABREVIATURAS Y SIGLAS</b>  | <b>55</b> |
| <b>APÉNDICES</b>  | <b>56</b> |
| <b>APÉNDICES DE FOTOS</b>   | <b>56</b> |
| <b>REFERENCIAS BIBLIOGRAFICAS</b>   | <b>59</b> |

## ÍNDICE DE GRÁFICOS

|   |    |
|---|----|
| FIGURE 1 INTERRELACIÓN ENTRE LOS ELEMENTOS DEL PATRÓN MVC .....                             | 17 |
| FIGURE 2 ACTIVIDADES DEL DESARROLLO DE SOFTWARE REPRESENTADO EN EL MODELO INCREMENTAL ..... | 18 |
| FIGURE 3 TIPOS DE REQUERIMIENTOS FURPS .....  | 20 |
| FIGURE 4 LOGOTIPO DE GITHUB.....  | 21 |
| FIGURE 5 LOGO DE VISUAL STUDIO .....  | 22 |
| FIGURE 6 LOGO DE SQL SERVER.....  | 23 |
| FIGURE 7 LOGO DEL LENGUAJE DE PROGRAMACIÓN C# .....   | 23 |
| FIGURE 8 ARQUITECTURA CLIENTE-SERVIDOR.....   | 24 |
| FIGURE 9 DIAGRAMA DE PROCESOS COMPLETO.....   | 25 |
| FIGURE 10 DIAGRAMA DE FLUJO INTEGRADO .....   | 26 |
| FIGURE 11 MODELO INCREMENTAL.....   | 29 |
| FIGURE 12 INVENTARIO DE REQUERIMIENTOS .....  | 30 |
| FIGURE 13 PROCEDIMIENTOS ALMACENADO .....   | 31 |
| FIGURE 14 MODELO LÓGICO DE LA BASE DE DATOS.....  | 32 |
| FIGURE 15 CUADRO DE PLANIFICACIÓN DE PRUEBAS.....   | 33 |
| FIGURE 16 DIAGRAMA DE ARQUITECTURA TI -HW Y SW POR ENTORNO.....                             | 34 |
| FIGURE 17 DIAGRAMA DE ARQUITECTURA DE SOFTWARE .....  | 34 |
| FIGURE 18 PAGINA DE REGISTRO DE USUARIO.....  | 35 |
| FIGURE 19 INICIO DE SESIÓN DEL SOFTWARE.....  | 36 |
| FIGURE 20 PAGINA DE BÚSQUEDA DE USUARIO .....   | 36 |
| FIGURE 21 PAGINA DE EMISIÓN DE BOLETA DE VENTA Y FACTURA.....                               | 37 |
| FIGURE 22 PAGINA DE BÚSQUEDA PARA EL REPORTE DE STOCK .....                                 | 37 |
| FIGURE 23 PROYECTO EN EL GITHUB .....   | 38 |
| FIGURE 24 VISTA EN CAPAS .....  | 38 |
| FIGURE 25 CÓDIGO DEL CASO DE USO REGISTRAR PRODUCTO .....                                   | 39 |
| FIGURE 26 MODELO-REGISTRAR PRODUCTO.....  | 40 |
| FIGURE 27 PROCEDIMIENTO ALMACENADO REGISTRAR PRODUCTO .....                                 | 40 |
| FIGURE 28 VISTA – INTERFAZ REGISTRAR PRODUCTO.....  | 42 |
| FIGURE 29 CÓDIGO EN GITHUB DE LA INTERFAZ REGISTRAR PRODUCTO.....                           | 43 |
| FIGURE 30 CONTROLADOR – REGISTRAR PRODUCTO .....  | 44 |

## ÍNDICE DE TABLAS

|   |    |
|---|----|
| TABLE 1 CUADRO DE REQUERIMIENTOS FUNCIONALES .....    | 27 |
| TABLE 2 CUADRO DE REQUERIMIENTOS NO FUNCIONALES ..... | 28 |
| TABLE 3 EJECUCIÓN DE PRUEBAS.....                     | 45 |

## CERTIFICADO DE REDACCIÓN, ORTOGRAFÍA Y ESTILO

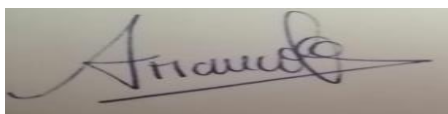
Huancayo, 23 de mayo del 2020

A quien corresponda.

Expedimos el presente certificado a fin de informarle que el proyecto titulado “Implementación de un sistema de ventas, control de empleados, productos y stock para la funeraria Rodríguez de Huancayo del 2020”, el equipo de trabajo realizó la revisión correspondiente a la redacción y ortografía del presente documento.

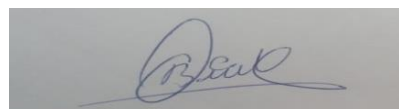
El trabajo consistió principalmente en la revisión por capítulos y su contenido y se empleó el estilo ISO. Todo que podemos decir es, que se trabajó de manera responsable, honesta y honrada.

Atentamente.



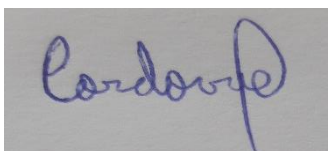
Arauco Medrano Merly

70041970



Condori Ochoa Brady

76531131



Córdova Limaylla Sergio

72889213

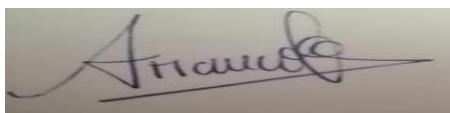
### ADECLARACIÓN DE AUTENTIFICIDAD

Los integrantes de proyecto de desarrollo en calidad de estudiantes de la Escuela Académica profesional de Ingeniería de Sistemas e Informática, declaramos que:

1. Los contenidos de este de proyecto de software, son absolutamente originales, auténticos, grupal y de exclusiva responsabilidad legal y académica de los autores.
2. El proyecto que ponemos en consideración para la evaluación no ha sido presentado anteriormente para obtener algún grado académico o título, ni ha sido publicado en ningún sitio web.

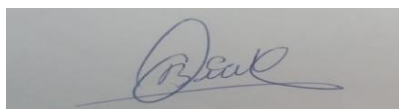
Asimismo, nos hacemos responsables ante la universidad o terceros de cualquier irregularidad o daño que pudiera ocasionar por incumplimiento de lo declarado.

Huancayo, 23 de mayo del 2020



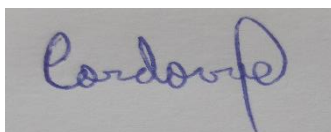
Arauco Medrano Merly

70041970



Condori Ochoa Brady

76531131



Córdova Limaylla Sergio

72889213



## RESUMEN EJECUTIVO

El proyecto Sistema de Venta de Productos para la empresa Funeraria Rodríguez consiste en la implementación de un sistema de escritorio, siendo su desarrollo a partir del uso de herramientas de software accesible.

El desarrollo del proyecto parte de la problemática y la situación actual de la empresa, el proceso para solicitar producto o servicio funerario de productos se ha convertido en algo engorroso y genera mucha pérdida de tiempo, ya que para realizar este proceso se realiza de manera manual, el registro de su stock es mediante un cuaderno. Lo cual genera riesgo de pérdida de información.

Es por esa razón nosotros decidimos implementar una solución ante este problema, un sistema de venta, mediante esta, los usuarios del sistema podrán encontrar rápido el producto que el cliente necesita, puedan registrar la venta de manera automática, cotizar los precios de los productos que requiere el cliente más rápido. El proyecto tiene como objetivo de automatizar el proceso de venta, registro de productos y reportes de las ventas.

El sistema fue realizado mediante el uso de los lenguajes de programación c#, para poder crear los prototipos de las interfaces se utilizó Balsamiq Backup. Además del uso de un gestor de base de datos en SQL Server para almacenar a los clientes que se registren, los productos en stock, las ventas realizadas.

## ABSTRACT

The Product Sales System project for the Funeralary Rodríguez company consists of the implementation of a desktop system, its development being based on the use of accessible software tools.

The development of the project starts from the problems and the current situation of the company, the process to request product or funeral service of products has become somewhat cumbersome and generates a lot of time loss, since to carry out this process it is done manually, the registration of your stock is through a notebook. Which generates risk of loss of information.

It is for this reason that we decided to implement a solution to this problem, a sales system. Through this, system users can quickly find the product that the customer needs, can register the sale automatically, quote the prices of the products that requires the fastest client. The project aims to automate the sales process, product registration and sales reports.

The system was made using the c # programming languages, in order to create the prototypes of the Balsamiq Backup interfaces. In addition to the use of a database manager in SQL Server for customers who register, products in stock, sales made.

## **CAPITULO I**

### **INTRODUCCIÓN**

#### **1.1. PLANTEAMIENTO Y JUSTIFICACIÓN DEL PROBLEMA**

El desarrollo del proyecto parte de la problemática y la situación actual de la empresa funeraria Rodríguez, el modo de revisar sus productos en stock en un cuaderno, o realizar las ventas de manera manual, realizar la boleta de venta o factura de manera manual, esto hace que genere insatisfacción en los clientes porque genera. ¿Cómo implementar un sistema de ventas para automatizar el proceso de ventas, registro de productos en la Funeraria Rodríguez del año 2020?

Es por ello que nosotros decidimos elaborar un sistema que ayude a los usuarios de la empresa para atender a sus clientes de manera más rápida, que tengan un registro de sus ventas y de sus clientes más frecuentes para promover descuentos y promociones. Esta empresa cuenta con una gran cantidad de clientes ya sean públicas y privadas, lo cual hace necesario la elaboración de un sistema que cumpla con todas las necesidades del cliente, seguridad en su información sin el riesgo de ser perdido.

#### **1.2. DESCRIPCION GENERAL DE LA EMPRESA**

- Nombre: Funeraria Rodríguez
- RUC: 10427951737
- Ubicación: Jirón Junín 360, Huancayo 12000
- Teléfono: 964735777

### Misión

Somos una empresa que se compromete en satisfacer las necesidades de nuestros clientes en integridad y seriedad La funeraria Rodríguez se enorgullece en ofrecer un servicio de alta calidad moral y sensibilidad humana, para brindarles consuelo en los momentos más difíciles

### Visión

Ser una empresa reconocida en la calidad de sus productos y servicios.

## 1.3. SITUACIÓN ACTUAL

Actualmente la empresa Funeraria Rodríguez se dedica a la venta de ataúdes y brindar servicios funerarios, estas ventas y servicios prestados son realizados manualmente llevando un registro en un cuaderno, incluyendo los productos en stock, esto retrasa los procesos de búsqueda de cada producto solicitado y corre riesgo de pérdida de la información.

## 1.4. PROBLEMAS

Los problemas encontrados en los procesos de la empresa Funeraria Rodríguez son los siguientes:

- **Registro de clientes**

Quando el encargado de ventas de la empresa realiza una venta, pide los datos del cliente para registrar sus datos de manera manual en una Boleta de ventas o factura; esta causa retraso en la atención a los clientes cuando existe bastante concurrencia por parte de los clientes.

- **Registro de Producto**

En el proceso de abastecimiento el comisionado realiza el registro del stock y las características del producto, en consecuencia, de esto la Empresa no tiene conocimiento de los productos con las que cuenta ni la cantidad.

- **Consultar Stock**

Cuando la empresa realiza la venta de un ataúd, el encargado debe buscar el registro en un cuaderno y anotar la venta para verificar cuantos productos le quedan en stock.

### **1.3.1 Propuesta de Cambio en el Proceso Principal**

- **Registro de Clientes**

Mediante el sistema que se propone implementar, el usuario del sistema podrá registrar al cliente cuando realice su compra de forma automática, y sus datos serán almacenados en una base de datos.

- **Registro de Productos**

El usuario del sistema podrá registrar en la base de datos todos los productos que ofrece, así como sus características y la cantidad en stock para que al momento de venderlos sean buscados por nombre o características.

### **1.5. JUSTIFICACIÓN**

La implementación de un sistema de información que nos ayudaría con el proceso de venta y reabastecimientos para tener una buena disponibilidad de la información en cualquier momento, y ser congruentes con los objetivos trazados por la empresa, con un mayor dinamismo en la implementación de los procesos, de esta manera lograr un mayor grado de satisfacción de los empleados y de la empresa.

### **1.6. OBJETIVOS**

- **OBJETIVO GENERAL**

Implementar un sistema de ventas, control de empleados, productos y stock para la funeraria Rodríguez de Huancayo del 2020

- **OBJETIVOS ESPECIFICOS**

- Implementar la meta 1: Venta Local.
- Implementar la meta 2: Venta por Teléfono.
- Implementar la meta 3: Consulta de Información.

### **1.7. METODOLOGÍA DE LA INVESTIGACIÓN**

Referente a la metodología de la investigación acerca de la empresa Funeraria Rodríguez, fue indispensable usar los siguientes tipos de metodologías:

- Metodología Exploratoria: Realizamos un seguimiento determinado para conocer el contexto y los procedimientos de las actividades de dicha empresa.
- Metodología análisis Crítico: Nos permitió analizar la situación actual de la empresa.
- Metodología Inductiva: Nos nuestras conclusiones generales y están hechas a partir de hechos particulares, acumulando datos que reafirmen nuestra postura.

### **1.8. PASOS DE LA ELABORACIÓN DEL PROYECTO**

Para la elaboración de un proyecto de software es necesario seguir diversas pautas para lograr su éxito entre las cuales se encuentran:

- ✓ Buscar y elegir una empresa que tenga problemas con sus procesos de negocio.
- ✓ Obtener toda la información necesaria para la definir los procesos de negocio y determinar el problema.
- ✓ Realizar su mapa de procesos actual y nuevo
- ✓ Diseñar el flujo de procesos actuales y nuevo
- ✓ Lista de requerimientos funcionales
- ✓ Los casos de uso
- ✓ Planificación de unidades de programación que nos ayudara a elaborar el software
- ✓ Diseño del modelo lógico de la base de datos
- ✓ Diseño de los prototipos
- ✓ Gestión de configuración en el GitHub
- ✓ Creación de la base de datos

- ✓ Lista de procedimientos almacenados según el plan de unidades de programación
- ✓ Plan de pruebas por incremento por cada caso de uso
- ✓ Entrega del software

### **1.9. LIMITACIONES**

- Limitación del tiempo
- Limitación de recursos
- Limitación de la ubicación de la empresa
- Limitación de la ubicación de los desarrolladores

### **1.10. ALCANCES DE LA INVESTIGACIÓN**

El sistema de escritorio está orientado para los usuarios que son encargados de llevar todo el control de las cuentas y ventas de los productos. Se entregará un sistema informático completamente funcional, compuesto de una serie de módulos que ayudará a los procesos principales de la empresa, como son el registro de clientes, productos y administración de los productos.

## CAPITULO II

### MARCO TEÓRICO Y CONTEXTUAL

#### 2.1. TECNICAS USADAS

##### 2.1.1. Mapa de procesos

Un proceso es un conjunto de actividades y recursos interrelacionados que transforman elementos de entrada en elementos de salida aportando valor añadido para el cliente o usuario. Los recursos pueden incluir: personal, finanzas, instalaciones, equipos técnicos, métodos, etc.

El propósito que ha de tener todo proceso es ofrecer al cliente / usuario un servicio correcto que cubra sus necesidades, que satisfaga sus expectativas, con el mayor grado de rendimiento en coste, servicio y calidad.

Un procedimiento es la forma específica de llevar a término un proceso o una parte del mismo. Los resultados deseados en los procesos dependen de los recursos, la habilidad y motivación del personal involucrado en el mismo, mientras los procedimientos son sólo una serie de instrucciones elaboradas para que las siga una persona o conjunto de personas.

Un mapa de procesos es un diagrama de valor; un inventario gráfico de los procesos de una organización. (Macías, 2007)

Los procesos pueden ser calificados:

- **Procesos claves:** son aquellos directamente vinculados a los bienes producidos o a los servicios que se prestan y, en consecuencia, orientados al cliente/usuario. Centrados en aportar valor, su resultado es percibido directamente por el cliente o usuario. Por lo general, en la ejecución de estos procesos intervienen varias áreas funcionales y son los que emplean los mayores recursos. (Macías, 2007)
- **Procesos Estratégicos:** Son aquellos establecidos por la alta dirección para definir cómo opera el negocio y cómo se crea valor. Constituyen el soporte de la toma de decisiones relacionadas con la planificación, las estrategias y las mejoras en la organización. También proporcionan directrices y límites al



resto de los procesos. Ejemplos de procesos estratégicos son la comunicación interna, la comunicación con el cliente, el marketing, el diseño, la revisión del sistema, la planificación estratégica, el diseño de planes de estudios, entre otros. (Macías, 2007)

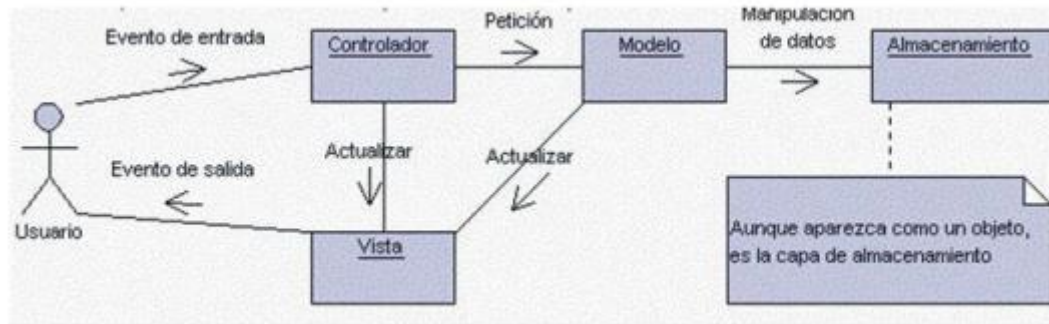
- **Procesos de Soporte:** Son aquellos que sirven de soporte a los procesos claves y a los procesos estratégicos. En muchos casos, estos procesos son determinantes para conseguir los objetivos de los procesos dirigidos a cubrir las necesidades y expectativas de los clientes o usuarios. Ejemplos de procesos de apoyo o soporte: formación, compras, auditorías internas, informática, etc. (Macías, 2007)

### 2.1.2. Patrón de diseño modelo-vista-controlador

El patrón Modelo-Vista-Controlador (MVC) surge con el objetivo de reducir el esfuerzo de programación, necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos, a partir de estandarizar el diseño de las aplicaciones. El patrón MVC es un paradigma que divide las partes que conforman una aplicación en el Modelo, las Vistas y los Controladores, permitiendo la implementación por separado de cada elemento, garantizando así la actualización y mantenimiento del software de forma sencilla y en un reducido espacio de tiempo. A partir del uso de frameworks basados en el patrón MVC se puede lograr una mejor organización del trabajo y mayor especialización de los desarrolladores y diseñadores

Un patrón común para el código MVC es:

- El controlador recibe una petición y busca alguna información en una base de datos.
- El controlador crea un modelo con la información y la adjunta a la vista.
- La vista es generada y mostrada en el navegador del usuario.
- El usuario presiona un botón o envía un formulario, lo que envía una nueva solicitud al controlador y el ciclo se repite. (Yenisleidy, 2012)



*Figure 1 Interrelación entre los elementos del patrón MVC*

### 2.1.3. Procedimientos almacenados

Un procedimiento almacenado es un grupo de una o más instrucciones Transact-SQL o una referencia a un método de Common Runtime Language (CLR) de Microsoft .NET Framework. Los procedimientos se asemejan a las construcciones de otros lenguajes de programación, porque pueden:

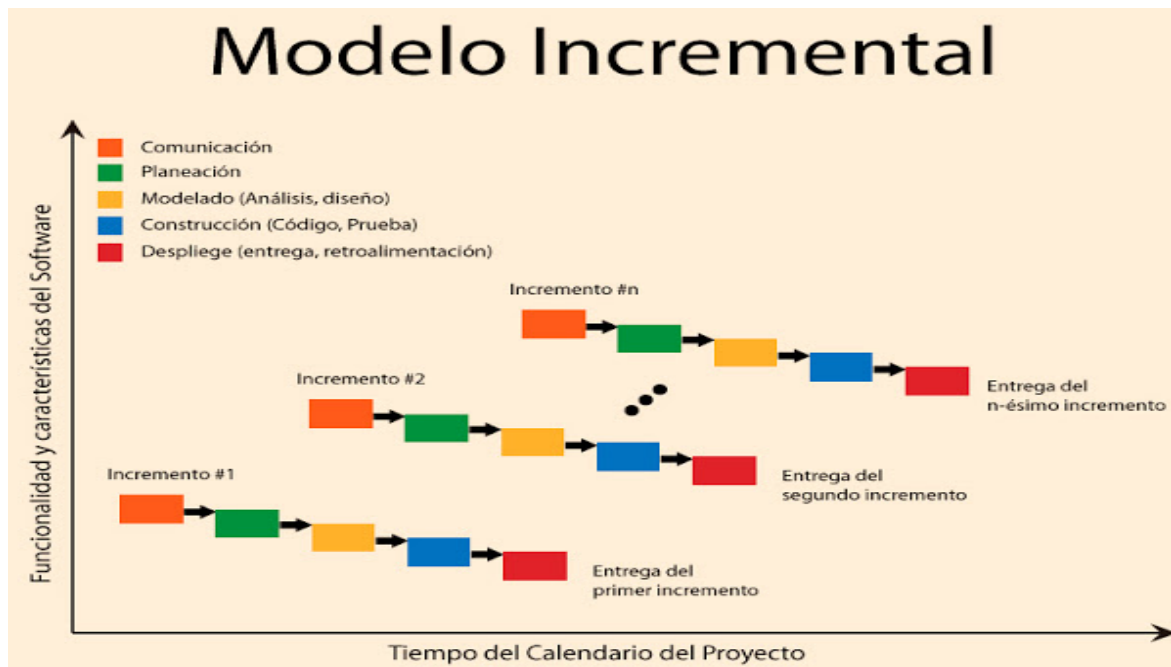
- Aceptar parámetros de entrada y devolver varios valores en forma de parámetros de salida al programa que realiza la llamada.
- Contener instrucciones de programación que realicen operaciones en la base de datos. Entre otras, pueden contener llamadas a otros procedimientos.
- Devolver un valor de estado a un programa que realiza una llamada para indicar si la operación se ha realizado correctamente o se han producido errores, y el motivo de estos. (MICROSOFT, 2019)

### 2.1.4. Desarrollo del ciclo de vida del software

El ciclo de vida de un software especifica el enfoque general del desarrollo, indicando los procesos, actividades y tareas que se van a realizar y en qué orden, y los productos que se van a generar, los que se van a entregar al cliente y en qué orden se van a entregar. (Sevilla, 2013)

**Modelo Incremental:** El modelo incremental combina elementos del modelo en cascada con la filosofía interactiva de construcción de prototipos. Se basa en la filosofía de construir incrementando las funcionalidades del programa.

Este modelo aplica secuencias lineales de forma escalonada mientras progresa el tiempo en el calendario. Cada secuencia lineal produce un incremento del software. Cuando se utiliza un modelo incremental, el primer incremento es a menudo un producto esencial, sólo con los requisitos básicos. Este modelo se centra en la entrega de un producto operativo con cada incremento. Los primeros incrementos son versiones incompletas del producto final, pero proporcionan al usuario la funcionalidad que precisa y también una plataforma para la evaluación. (Sevilla, 2013)



*Figure 2 Actividades del desarrollo de software representado en el Modelo Incremental*

#### 2.1.5. Gestión de la configuración del Software (GUS)

La Gestión de Configuración de Software (GUS) forma parte de los procesos que intervienen en el desarrollo de software. Son muchas las definiciones existentes sobre esta disciplina. Todo especialista que la ha definido ha aportado nuevos puntos de vista, así como tareas a tener en cuenta. En ocasiones, pudiera existir diferencias dentro de estos conceptos en cuanto a nombres o prioridades entre tareas a realizar, sin embargo, todos señalan la

importancia de esta disciplina. Roger S. Pressman la definió de la siguiente manera:

“El arte de coordinar el desarrollo de software para minimizar la confusión se denomina gestión de configuración. La gestión de configuración es el arte de identificar, organizar y controlar las modificaciones que sufre el software que construye un equipo de programación. La meta es maximizar la productividad minimizando los errores”. (Morejón, 2015)

#### **2.1.6. Método FURPS**

El modelo FURPS incluye, además de los factores de calidad y los atributos, restricciones de diseño y requerimientos de implementación, físicos y de interfaz. Una limitación de este modelo de calidad es que no tiene en cuenta la portabilidad de los productos software que se estén considerando, factor digno de consideración en función de las exigencias actuales que recaen sobre el proceso de desarrollo del software.

El modelo FURPS establece cinco características como factores de calidad que son los que le dan nombre:

- Functionality (Funcionalidad).
- Usability (Usabilidad).
- Reliability (Confiabilidad).
- Performance (Prestación) y
- Supportability (Soporte).

| Sigla | Tipo de Requerimiento             |                         | Descripción  |
|-------|-----------------------------------|-------------------------|--|
| F     | Funtional                         | <b>Funcional</b>        | Características, capacidades y algunos aspectos de seguridad                                 |
| U     | Usability                         | <b>Facilidad de Uso</b> | Factores Humanos (interacción), ayuda, documentación   |
| R     | Reliability <small>Furps+</small> | <b>Fiabilidad</b>       | Frecuencia de fallos, capacidad de recuperación de un fallo y grado de previsión             |
| P     | Performance                       | <b>Rendimiento</b>      | Tiempos de respuesta, productividad, precisión, disponibilidad, uso de los recursos          |
| S     | Supportability                    | <b>Soporte</b>          | Adaptabilidad, facilidad de mantenimiento, internacionalización, facilidad de configuración. |
| +     | Plus                              | <b>Implementación</b>   | Limitación de recursos, lenguajes y herramientas, hardware                                   |
|       |                                   | <b>Interfaz</b>         | Restricciones impuestas para la interacción con sistemas externos (no es GUI)                |
|       |                                   | <b>Operaciones</b>      | Gestión del sistema, pautas administrativas, puesta en marcha                                |
|       |                                   | <b>Empaquetamiento</b>  | Forma de distribución  |
|       |                                   | <b>Legales</b>          | Licencia, derechos de autor, etc.  |

Figure 3 Tipos de requerimientos FURPS

### 2.1.7. Evaluación CMMI

La evaluación CMMI es un examen de uno o más procesos por un calificado equipo de profesionales con un modelo de referencia evaluación como base para determinar las fortalezas y debilidades de la organización.

Las evaluaciones requieren una planificación. Al planear una evaluación de su organización, determinar el alcance de la unidad orgánica, que las disciplinas que incluyen, si el equipo de evaluación estará compuesto por miembros internos o externos a la organización, los proyectos que se han incluido, a las personas a ser entrevistadas y el tipo o clase de evaluación es necesario. (Anonimo, 2017)

## 2.2. HERRAMIENTAS DEL MODELADO Y DISEÑO DE SISTEMAS

Las herramientas que se usaron para el desarrollo de la aplicación son las siguientes:

### 2.2.1. Herramientas del Modelado y Diseño de Sistemas

- **Bizagi**

Bizagi Modeler es una aplicación basada en el estándar Business Process Model and Notation (BPMN) que te brinda la posibilidad de documentar y modelar procesos de negocio fácilmente. (Bizagi, 2019)

- **Balsamiq Mockups**

Balsamiq Mockups es una aplicación para crear maquetas para interfaces gráficas para usuario. Le permite al diseñador diagramar widgets pre contruidos utilizando un editor WYSIWYG (Balsamiq, 2019)

### 2.2.2. Framework de desarrollo: GitHub

Es una plataforma de desarrollo inspirada en la forma en que trabajas. Desde el código abierto hasta el negocio, puede alojar y revisar códigos, administrar proyectos y crear software junto a millones de otros desarrolladores. (GitHub, 2019)



*Figure 4 Logotipo de GitHub*

### 2.2.3. Plataforma de Desarrollo

- **Visual Studio**

Es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para Windows, Linux y macOS. Es compatible con múltiples lenguajes de programación, tales como C++, C#, Visual Basic .NET, F#, Java, Python, Ruby y PHP, al igual que entornos de desarrollo web, como ASP.NET MVC, Django, etc., a lo cual hay que sumarle las nuevas capacidades en línea bajo Windows Azure en forma del editor Monaco. (Microsoft, 2019)



*Figure 5 Logo de Visual Studio*

### 2.2.4. Motor de Base de Datos

- **SQL Server**

Es un sistema de gestión de base de datos relacional, desarrollado por la empresa Microsoft.

El lenguaje de desarrollo utilizado (por línea de comandos o mediante la interfaz gráfica de Management Studio) es Transact-SQL (TSQL), una implementación del estándar ANSI del lenguaje SQL, utilizado para manipular y recuperar datos (DML), crear tablas y definir relaciones entre ellas (DDL). (Microsoft, 2019)



*Figure 6 Logo de SQL Server*

#### **2.2.5. Lenguaje de Programación**

- **C#**

C# es un lenguaje elegante, con seguridad de tipos y orientado a objetos que permite a los desarrolladores crear una gran variedad de aplicaciones seguras y sólidas que se ejecutan en .NET Framework. Puede usar C# para crear aplicaciones cliente de Windows, servicios web XML, componentes distribuidos, aplicaciones cliente-servidor, aplicaciones de base de datos y muchas, muchas más cosas. Visual C# proporciona un editor de código avanzado, prácticos diseñadores de interfaz de usuario, un depurador integrado y muchas otras herramientas que facilitan el desarrollo de aplicaciones basadas en el lenguaje C# y .NET Framework. (Microsoft, 2019)



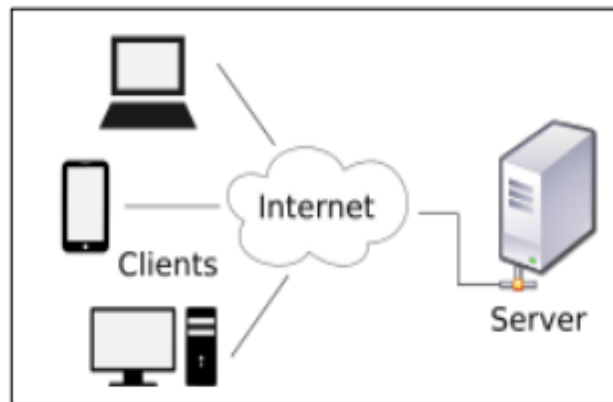
*Figure 7 Logo del  
lenguaje de programación C#*



### 2.2.6. Técnicas Temáticas

- **Hardware**

El sistema que se va a realizar debe poseer la capacidad técnica de manejar una cantidad de datos, por lo que se requiere de un computador servidor (arquitectura cliente-servidor) que soporte el almacenamiento de base de datos con todos los registros de la empresa Biogen JY. Por lo que la empresa no deberá adquirir ningún sistema computacional nuevo para el desarrollo, ya que existe suficiente hardware para llevar a cabo el proyecto.



*Figure 8 Arquitectura Cliente-Servidor*

- **Recursos Humanos**

Para la realización de este proyecto se cuenta con tres alumnos que posee las

siguientes competencias académicas:

- Lenguaje de programación: C# y visual Basic.
- Modelamiento de Software: Lenguaje de modelado UML.
- Base de Datos: SQL

## CAPITULO III

### INGENIERIA DEL PRODUCTO O DE DESARROLLO DE LA INVESTIGACIÓN

#### 3.1. MAPA DE PROCESOS

Mapa de procesos de la Empresa Rodríguez

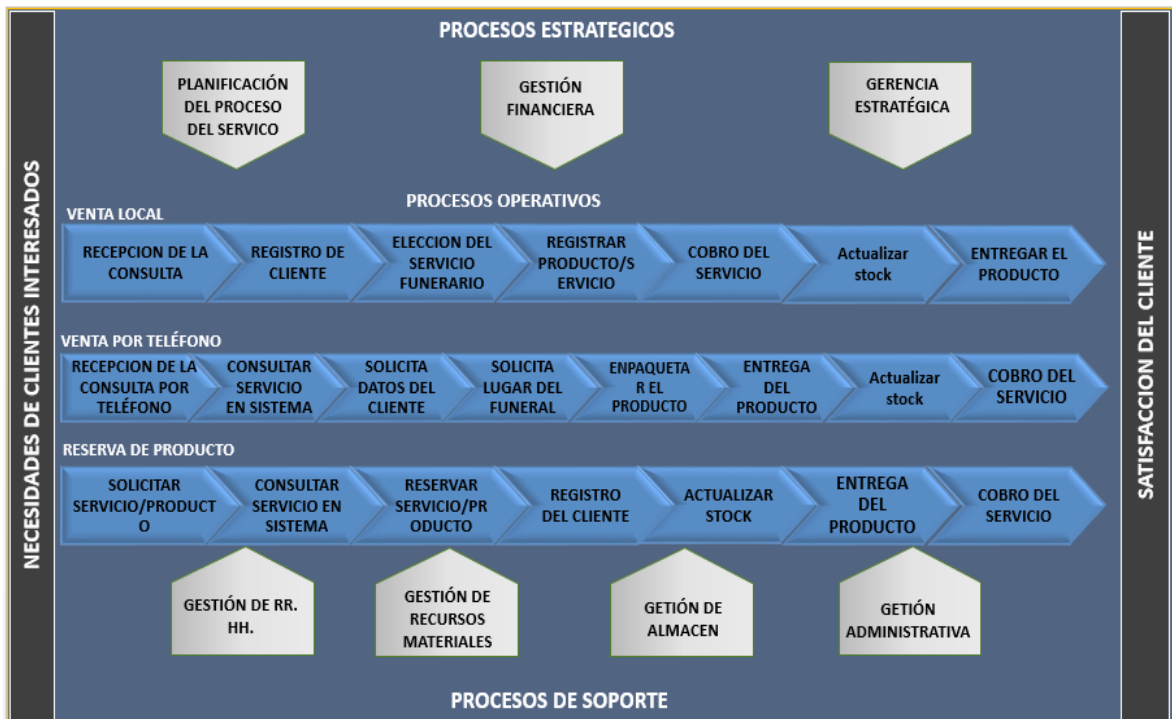


Figure 9 Diagrama de Procesos Completo

- DIAGRAMA DE FLUJO INTEGRADO DE CADA FLUJO DE TRABAJO DE LA FUNERARIA RODRÍGUEZ**

El flujograma de procesos es la representación gráfica de los principales procesos. Representa los flujos de trabajo paso a paso, de negocio y operacionales de los componentes de la empresa funeraria Rodríguez.

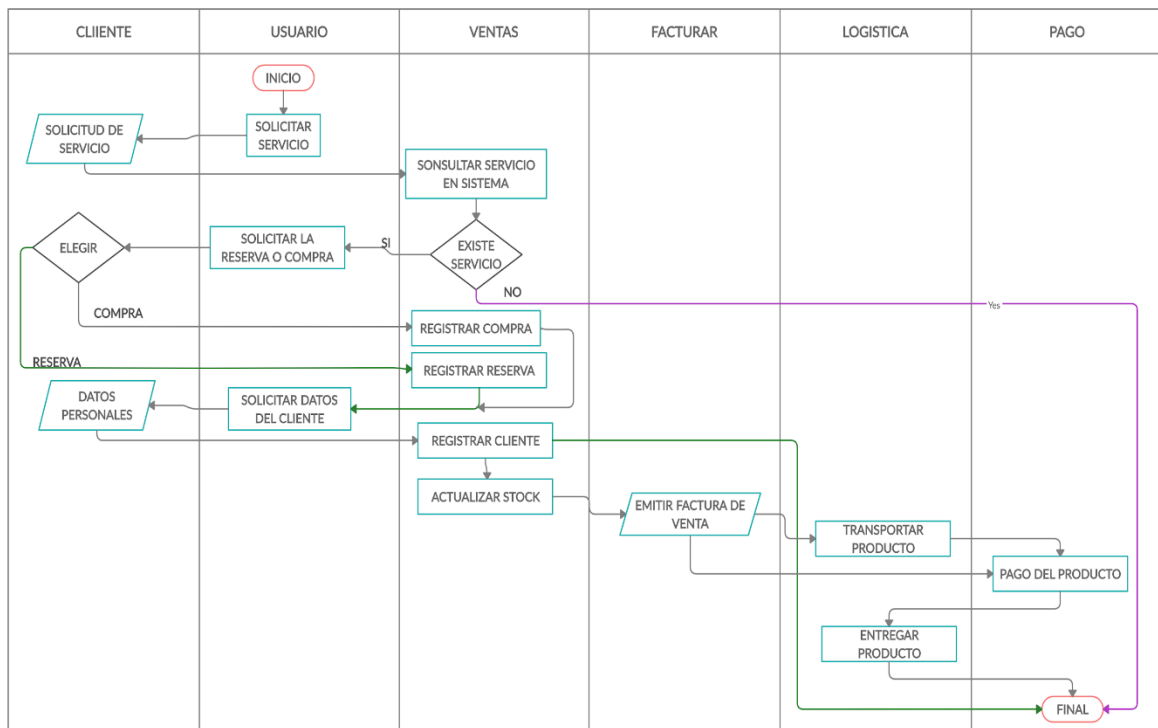


Figure 10 Diagrama de flujo integrado

### 3.3 INVENTARIO DE REQUERIMIENTOS

Hacer el inventario de requerimientos nos ayudó construir el sistema que resuelve las necesidades del usuario, de los servicios que se espera que la página web proporcione y de las restricciones bajo las cuales debe funcionar.

- CUADRO DE REQUERIMIENTOS FUNCIONALES

*Table 1 Cuadro de requerimientos funcionales*

| ID | REQUERIMIENTOS FUNCIONALES  | CASOS DE USO                               |
|----|---|--|
| 01 | Permite el registro de los usuarios para que puedan acceder al sistema    | Registrar usuario                          |
| 02 | Permite que el usuario accede al Sistema con un Id y una contraseña.      | Autenticación                              |
| 03 | Permite el registro del cliente como persona natural o una empresa.       | Registrar cliente                          |
| 04 | Permite el registro de productos con sus características.                 | Registrar producto                         |
| 05 | Permite el registro de los servicios con sus descripciones                | Registrar servicio                         |
| 06 | Permite editar los campos de los productos                                | Modificar producto por compra de proveedor |
| 07 | Permite editar las columnas del servicio                                  | Modificar servicio                         |
| 08 | Permite realizar una venta por teléfono                                   | Realizar venta por teléfono                |
| 09 | Permite emitir una boleta de venta  | Emitir boleta de venta                     |
| 10 | Permite emitir una factura  | Emitir factura                             |
| 11 | Permite buscar a los clientes que fueron registrados por su nombre        | Consulta de cliente por nombre             |
| 12 | Permite buscar a los clientes que fueron registrados por su número de dni | Consulta de cliente por número de DNI      |
| 13 | Permite buscar a los clientes que fueron registrados por su razón social  | Consulta de cliente por razón social       |
| 14 | Permite buscar a los clientes que fueron registrados por su número de ruc | Consulta de cliente por número de ruc      |

|    |  |   |
|----|--|---|
| 15 | Permite buscar a los productos registrados por su nombre           | Consulta de producto por nombre           |
| 16 | Permite buscar a los productos registrados por el tipo de material | Consulta de producto por tipo de material |
| 17 | Permite buscar a los productos registrados por su precio           | Consulta de producto por precio           |
| 18 | Permite buscar a los usuarios registrados por su número de dni     | Consulta de usuario por número de dni     |
| 19 | Permite ver todas las ventas realizadas                            | Reporte ventas                            |
| 20 | Permite ver los productos en stock y sin stock.                    | Reporte stock                             |
| 21 | Permite ver los productos más vendidos.                            | Reporte top                               |

- CUADRO DE REQUERIMIENTOS NO FUNCIONALES

*Table 2 Cuadro de requerimientos no funcionales*

|    | REQUERIMIENTOS NO FUNCIONALES           |
|----|---|
| 01 | Disponibilidad 24 x 7                   |
| 02 | Acceder fácilmente al sistema           |
| 03 | Plataforma modalidad Escritorio         |
| 04 | Lenguaje de programación C# Versión 5.0 |
| 05 | Se puede usar en otros dispositivos     |
| 06 | Uso de la Base de datos SQL SERVER 2016 |
| 07 | Balsamiq mockups versión 3.5            |

### 3.4 DESARROLLO INCREMENTAL

#### MODELO INCREMENTAL DE PROCESOS DE SOFTWARE

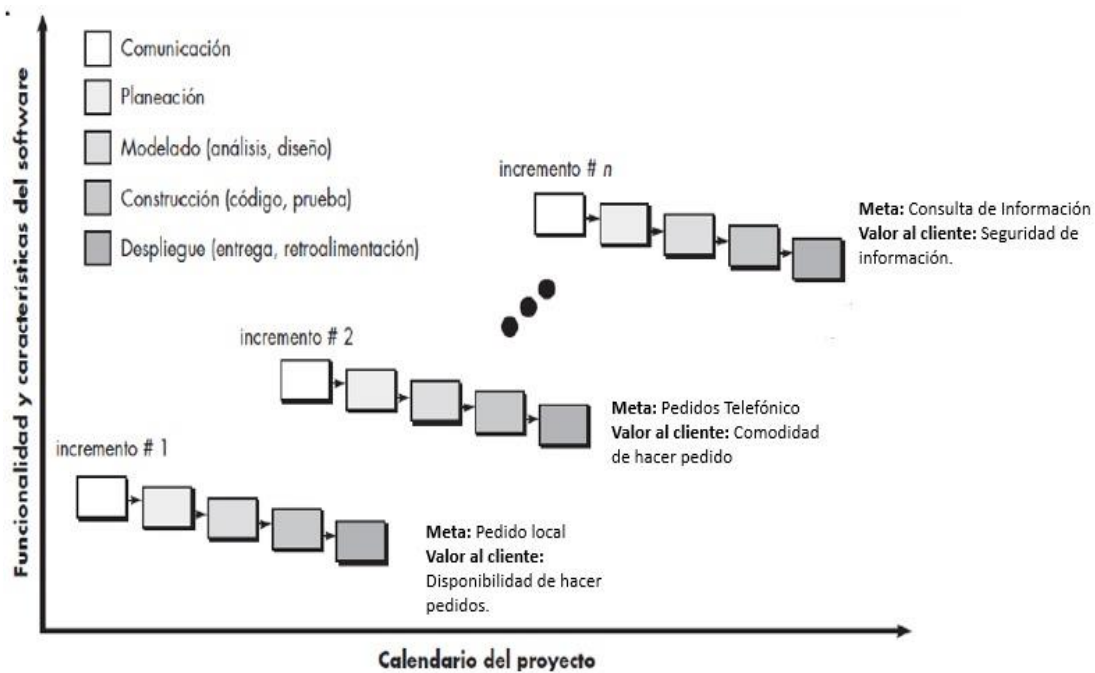


Figure 11 Modelo Incremental

### 3.5 GESTIÓN DEL PRODUCTO Y PROYECTO

- Planificación de Unidades de Programación

Inventario de requerimientos de la funeraria “Rodriguez”

| Requerimiento          | Incremento | Tipo Clase  | Nombre Clase           | Responsable |                              |
|------------------------|------------|-------------|------------------------|-------------|------------------------------|
| Registrar Usuario      | 1          | Modelo      | M_registra_usuario     | SCL         | SCL: Sergio Cordova Limaylla |
|                        | 1          | Vista       | V_registra_usuario     | BCO         | BCO: Brady Condori Ochoa     |
|                        | 1          | Controlador | C_registra_usuario     | MAM         | MAM: Merly Arauco Medrano    |
| Autenticacion          | 1          | Modelo      | M_autenticacion        | BCO         |                              |
|                        | 1          | Vista       | V_autenticacion        | SCL         |                              |
|                        | 1          | Controlador | C_autenticacion        | MAM         |                              |
| Emitir Boleta de Venta | 1          | Modelo      | M_emitir_boleta_ventas | MAM         |                              |
|                        | 1          | Vista       | V_emitir_boleta_ventas | BCO         |                              |
|                        | 1          | Controlador | C_emitir_boleta_ventas | SCL         |                              |
| Reporte de Ventas      | 1          | Modelo      | M_reporte_ventas       | SCL         |                              |
|                        | 1          | Vista       | V_reporte_ventas       | MAM         |                              |
|                        | 1          | Controlador | C_reporte_ventas       | BCO         |                              |
| Reporte stock          | 1          | Modelo      | M_reporte_stock        | MAM         |                              |
|                        | 1          | Vista       | V_reporte_stock        | BCO         |                              |
|                        | 1          | Controlador | C_reporte_stock        | SCL         |                              |
| Reporte top            | 1          | Modelo      | M_reporte_top          | BCO         |                              |
|                        | 1          | Vista       | V_reporte_top          | SCL         |                              |
|                        | 1          | Controlador | C_reporte_top          | MAM         |                              |
| Registrar Producto     | 1          | Modelo      | M_registrar_producto   | MAM         |                              |
|                        | 1          | Vista       | V_registrar_producto   | SCL         |                              |
|                        | 1          | Controlador | C_registrar_producto   | BCO         |                              |

Figure 12 Inventario de Requerimientos

- **Procedimientos Almacenados**

Con el propósito de mejorar el manejo de información de la base de datos y su relación con el patrón MVC, se presenta la lista de procedimientos almacenados por tabla y ACME (agregar, consultar, modificar, eliminar)

| ID | TABLA    | INCREMENTO | LISTA DE PROCEDIMIENTOS ALMACENADOS | RESPONSABLE |
|----|----------|------------|-------------------------------------|-------------|
| 1  | usuario  | 1          | SP_A_Tabla_Usuario                  | SCL         |
| 2  |          | 1          | SP_C_Tabla_Usuario                  | SCL         |
| 3  |          | 1          | SP_M_Tabla_Usuario                  | BCD         |
| 4  |          | 1          | SP_E_Tabla_Usuario                  | MAM         |
| 5  | cliente  | 1          | SP_A_Tabla_Cliente                  | SCL         |
| 6  |          | 1          | SP_C_Tabla_Cliente                  | BCD         |
| 7  |          | 1          | SP_M_Tabla_Cliente                  | MAM         |
| 8  |          | 1          | SP_E_Tabla_Cliente                  | MAM         |
| 9  | venta    | 1          | SP_A_Tabla_Venta                    | SCL         |
| 10 |          | 1          | SP_C_Tabla_Venta                    | BCD         |
| 11 |          | 1          | SP_M_Tabla_Venta                    | BCD         |
| 12 |          | 1          | SP_E_Tabla_Venta                    | MAM         |
| 13 | producto | 1          | SP_A_Tabla_Producto                 | SCL         |
| 14 |          | 1          | SP_C_Tabla_Producto                 | BCD         |
| 15 |          | 1          | SP_M_Tabla_Producto                 | MAM         |
| 16 |          | 1          | SP_E_Tabla_Producto                 | BCD         |
| 17 | boleto   | 1          | SP_A_Tabla_Boleto                   | BCD         |
| 18 |          | 1          | SP_C_Tabla_Boleto                   | MAM         |
| 19 |          | 1          | SP_M_Tabla_Boleto                   | SCL         |
| 20 |          | 1          | SP_E_Tabla_Boleto                   | SCL         |

Figure 13 Procedimientos Almacenado



- **Modelo Lógico de Base de Datos**

*Modelo Entidad- Relación SQL Server*



Activar Wind

*Figure 14 Modelo Lógico de la Base de Datos*

- **Plan de Prueba**

- **Cuadro de planificación de pruebas del primer incremento**

| Caso de uso            | Datos de entrada normal   | Resultado esperado normal | Datos de entrada anómala                 | Resultado esperado anómalo | 1 | 2 | 3 |
|------------------------|---|---------------------------|--|----------------------------|---|---|---|
| Registrar usuario      | . DNI<br>. Nombre<br>. Apellido Paterno<br>. Apellido Materno<br>. Teléfono<br>. E-mail<br>. Contraseña | Usuario registrado        | DNI con más de 8 dígitos                 | Error de registro          |   | X |   |
| Autenticación          | . Dni<br>. contraseña   | Inicio exitoso            | Inicio incorrecto de nombre o contraseña | Error de ingreso           |   | X |   |
| Emitir Boleta de Venta | . Número de venta<br>. Productos<br>. DNI<br>. Cliente<br>. Total a pagar                               | Mostrar Boleta            | Nombre del producto erróneo              | Producto no encontrado     |   |   | X |
| Reporte de Ventas      | . Fecha<br>. Hora<br>. Número de ventas realizadas  | Reporte Exitoso           | Inicio de letras o símbolos en lugar de  | Reporte fallido            |   |   | X |
| Reporte Stock          | . Id del producto<br>. Nombre del producto  | Stock mostrado            | Nombre del producto erróneo              | Reporte fallido            |   | X |   |
| Reporte Top            | . Nombre de producto  | Mostrar top               | Nombre del producto erróneo              |                            |   |   | X |
| Registrar producto     | . Id<br>. Nombre del producto<br>. Stock  | Producto registrado       | Id erróneo                               | Registro fallido           |   | X |   |

*Figure 15 Cuadro de planificación de pruebas*

- **Diagrama de Despliegue**

- Diagrama de arquitectura TI -HW y SW por entorno

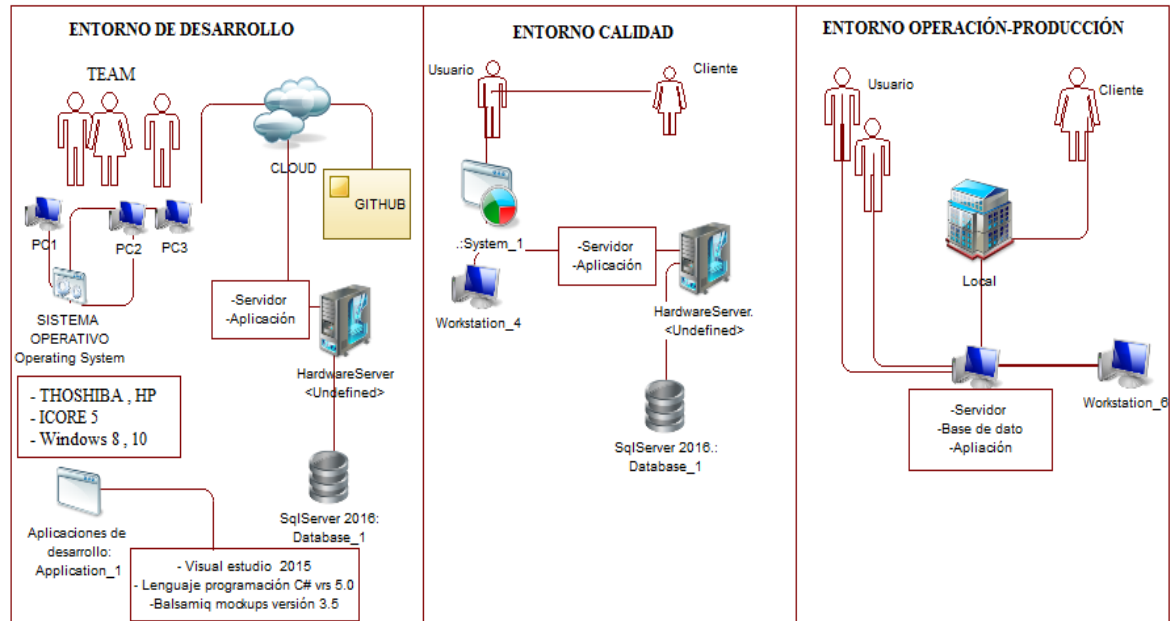


Figure 16 Diagrama de arquitectura TI -HW y SW por entorno

- Diagrama de arquitectura de Software

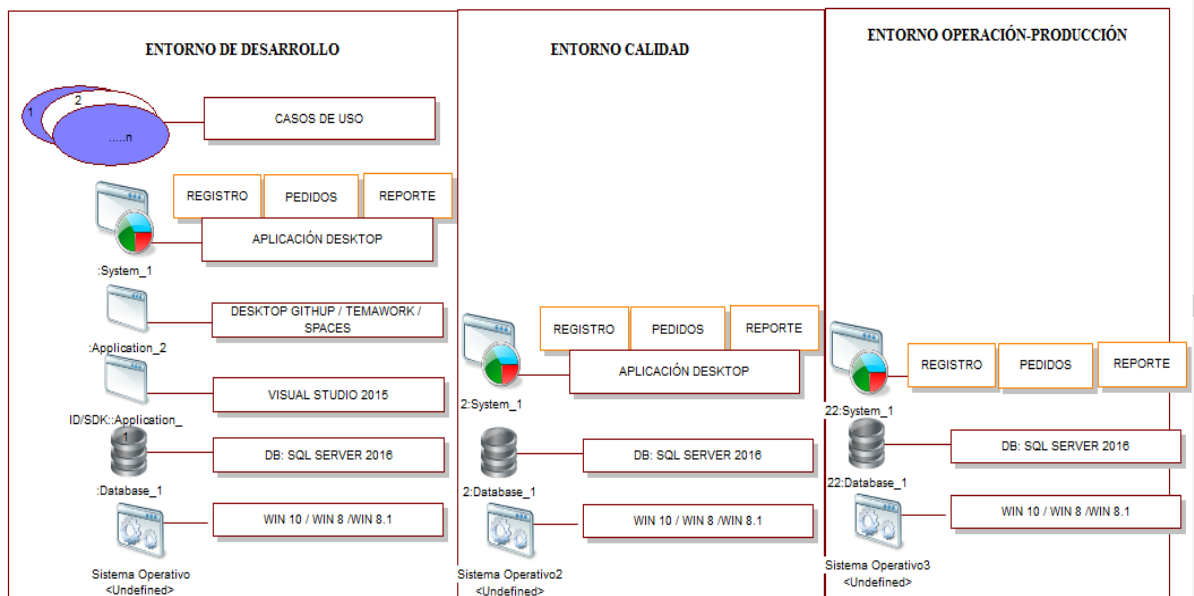


Figure 17 Diagrama de arquitectura de Software

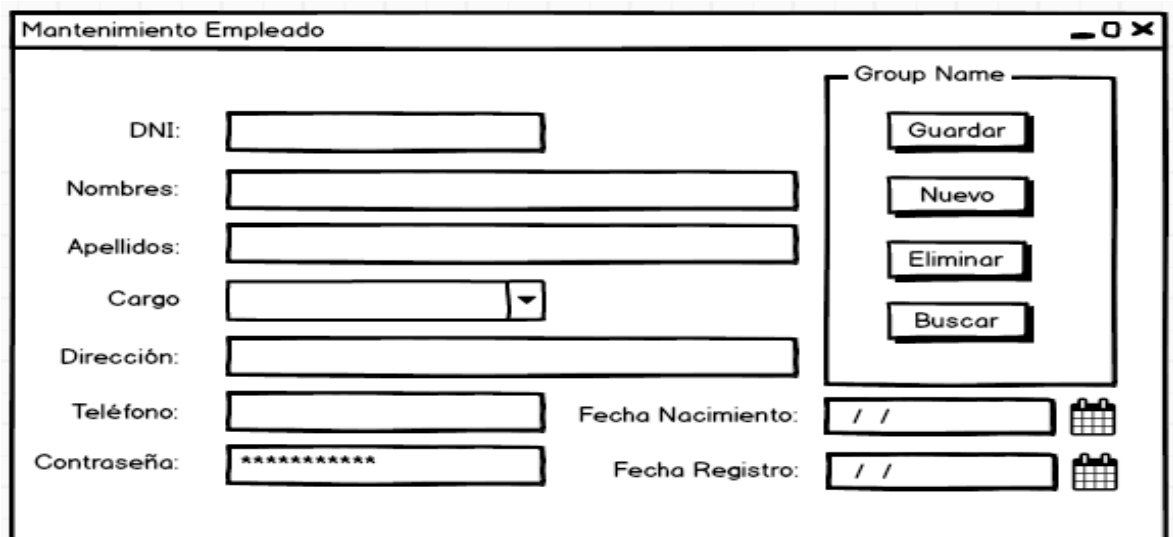
### 3.6 DESARROLLO DEL CICLO DE VIDA DEL SOFTWARE

- **Diseño de Prototipos**

Los prototipos son un mapa para la elaboración del producto de parte del equipo de desarrollo, ya que se utiliza como un bosquejo de cómo se debe de incluir el software al entorno real.

#### Diseño de prototipo por incremento

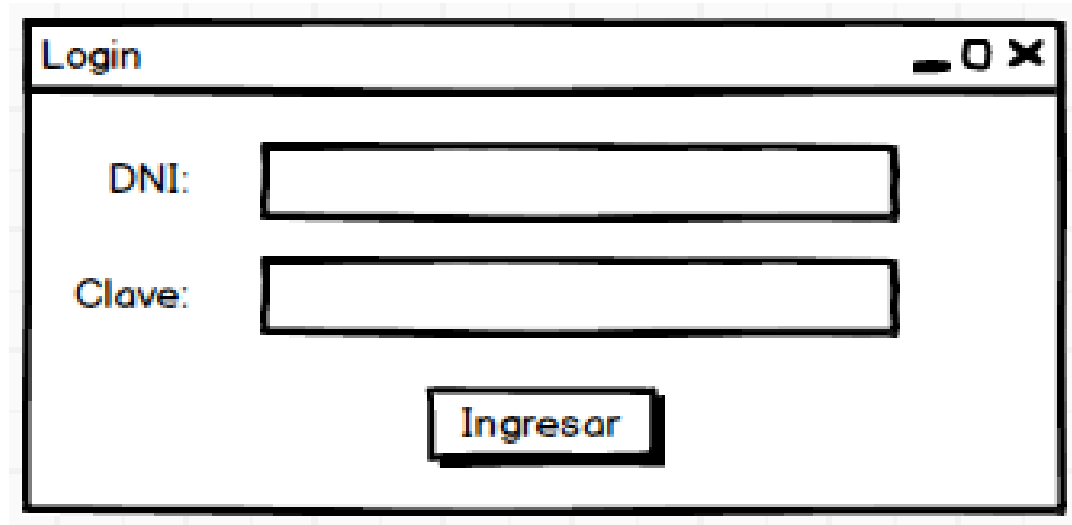
- **Registrar usuario:**



The image shows a screenshot of a web application window titled "Mantenimiento Empleado". The window contains a form for user registration. The form fields are arranged in two columns. The left column contains: "DNI:" with a text input field; "Nombres:" with a text input field; "Apellidos:" with a text input field; "Cargo" with a dropdown menu; "Dirección:" with a text input field; "Teléfono:" with a text input field; and "Contraseña:" with a text input field containing asterisks. The right column contains: "Fecha Nacimiento:" with a date input field (//) and a calendar icon; and "Fecha Registro:" with a date input field (//) and a calendar icon. To the right of the form fields is a vertical box labeled "Group Name" containing four buttons: "Guardar", "Nuevo", "Eliminar", and "Buscar".

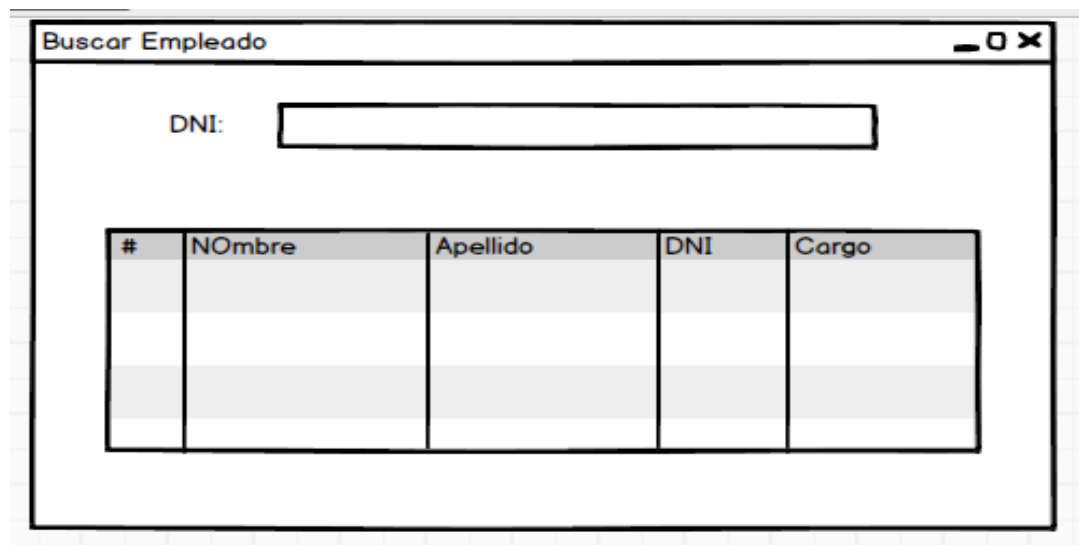
Figure 18 Pagina de registro de usuario

- Autenticación:



A screenshot of a software login window titled "Login". It features two input fields: "DNI:" and "Clave:". Below these fields is a button labeled "Ingresar". The window has a standard title bar with minimize, maximize, and close buttons.

Figure 19 Inicio de sesión del software

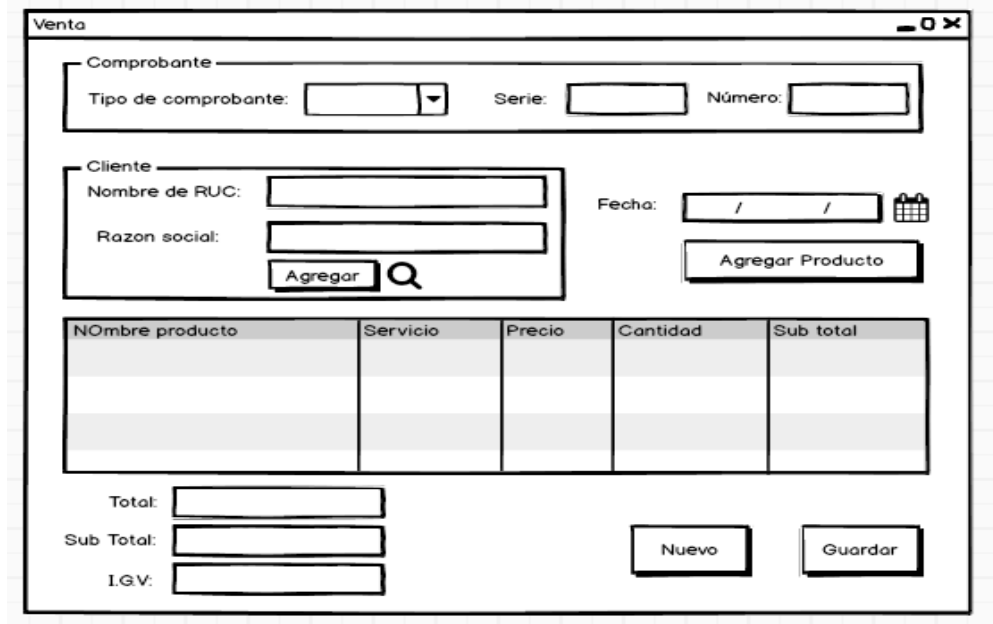


A screenshot of a software window titled "Buscar Empleado". It contains a "DNI:" input field and a table with five columns: "#", "NOmbre", "Apellido", "DNI", and "Cargo". The table has three empty rows for data entry.

| # | NOmbre | Apellido | DNI | Cargo |
|---|--------|----------|-----|-------|
|   |        |          |     |       |
|   |        |          |     |       |
|   |        |          |     |       |



Figure 20 Pagina de búsqueda de usuario

- Emitir boleta o factura de venta:



**Venta**

Comprobante:   
 Tipo de comprobante:  Serie:  Número:

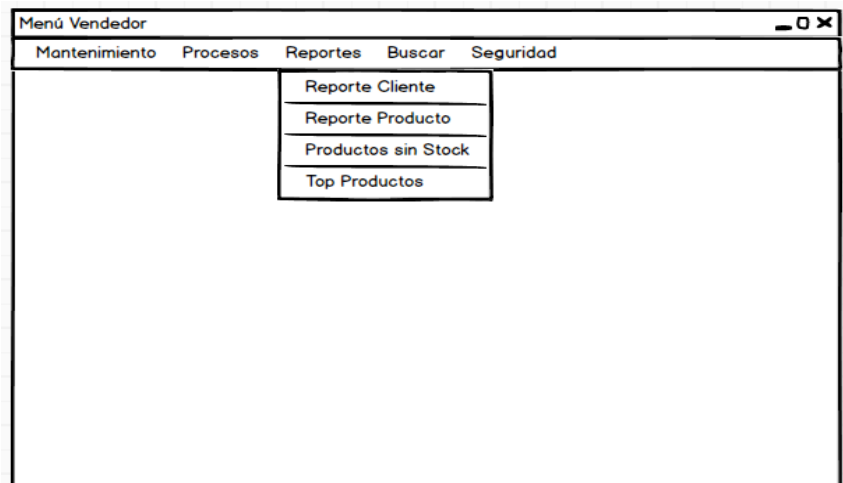
Cliente:   
 Nombre de RUC:  Fecha:     
 Razon social:    
 

| NOmbre producto | Servicio | Precio | Cantidad | Sub total |
|-----------------|----------|--------|----------|-----------|
|                 |          |        |          |           |
|                 |          |        |          |           |
|                 |          |        |          |           |

Total:    
 Sub Total:    
 I.G.V:

Figure 21 Pagina de emisión de boleta de venta y factura

- Reporte ventas - reportes stock - reporte Top



**Menú Vendedor**

Mantenimiento Procesos Reportes Buscar Seguridad

Reporte Cliente   
 Reporte Producto   
 Productos sin Stock   
 Top Productos

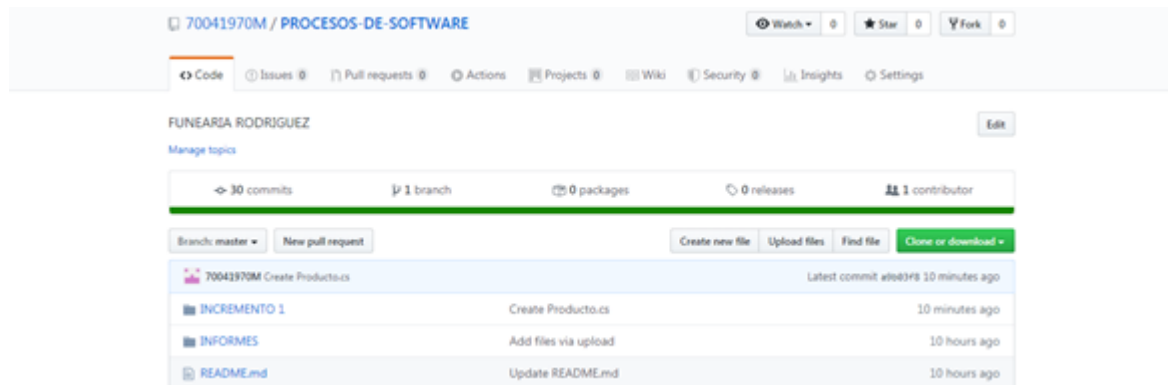
Figure 22 Pagina de búsqueda para el reporte de stock

### 3.7 GESTIÓN DE LA CONFIGURACIÓN

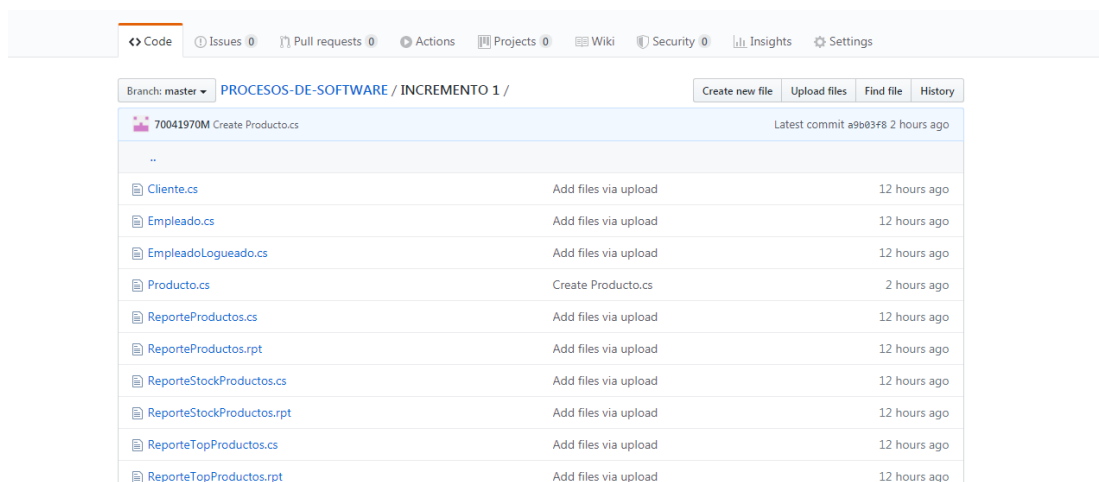
La gestión de la configuración es un identificador de versiones y cambios abordados a lo largo del ciclo de vida del software, esto supone que el software a lo largo del tiempo ha adquirido nuevas funcionalidades y se ha quedado con las que satisfacen los requerimientos del gerente general de la empresa, estas configuraciones están registradas en el GitHub.

- **Código fuente en GitHub**

Evidencias del software y documentación en GITHUB y código fuente



*Figure 23 Proyecto en el GitHub*



*Figure 24 Vista en capas*

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security 0 Insights Settings

Branch: master PROCESOS-DE-SOFTWARE / INCREMENTO 1 / Producto.cs Find file Copy path

70041970M Create Producto.cs a9b83f8 2 hours ago 1 contributor

137 lines (102 sloc) 3.44 KB Raw Blame History

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace Rodriguez.Entitites
8  {
9      class Producto
10     {
11
12         private Int16 idproducto;
13
14         public Int16 Idproducto
15         {
16             get { return idproducto; }
17             set { idproducto = value; }
18         }
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33     {
34         get { return categoria; }
35         set { categoria = value.ToUpper(); }
36     }
37
38     private string nombre;
39
40     public string Nombre
41     {
42         get { return nombre; }
43         set { if (String.IsNullOrEmpty(value)) throw new Exception("El valor de nombre no puede ser nulo ni vacio");
44             else if (value.Length > 50) throw new Exception("la longitud de nombre no puede superar los 50 caracteres");
45             else nombre = value.ToUpper();
46         }
47     }
48
49
50
51     private string diseño;
52
53     public string Diseño
54     {
55         get { return diseño; }
56         set {
57             if (String.IsNullOrEmpty(value)) throw new Exception("El valor de diseño no puede ser nulo ni vacio");
58             else if (value.Length > 50) throw new Exception("la longitud de diseño no puede superar los 50 caracteres");
59             else diseño = value.ToUpper();
60         }
61     }
62
63
64     private string material;

```

Figure 25 Código del caso de uso Registrar Producto



### 3.8 EVIDENCIAS DEL CÓDIGO FUENTE EN CLASES MVC

- **Modelo:** Es el responsable de mantener y gestionar los datos de la aplicación

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace Rodriguez.Entitites
8  {
9      class Producto
10     {
11
12         private Int16 idproducto;
13
14         public Int16 Idproducto
15         {
16             get { return idproducto; }
17             set { idproducto = value; }
18         }
19
20
21
22         private Int16 stock;
23
24         public Int16 Stock
25     {
26         get { return stock; }
27         set { stock = value; }
28     }
29
30     private string categoria;
31
32     public string Categoria
33     {
34         get { return categoria; }
35         set { categoria = value.ToUpper(); }
36     }
37
38     private string nombre;
39
40     public string Nombre
41     {
42         get { return nombre; }
43         set { if (String.IsNullOrEmpty(value)) throw new Exception("El valor de nombre no puede ser nulo ni vacio");
44             else if (value.Length > 50) throw new Exception("la longitud de nombre no puede superar los 50 caracteres");
45
46             else nombre = value.ToUpper();
47         }
48     }
49
50
51     private string diseño;
52
53     public string Diseño
```

Figure 26 Modelo-Registrar Producto

```
CREATE PROCEDURE [dbo].[sp_producto_insertar]
    @nombrepro varchar(50)
    ,@diseño    varchar(50)
    ,@material  varchar(50)
    ,@descripcion varchar(150)
    ,@precio    decimal(10,2)
    ,@tamaño    varchar(20)
    ,@stockpro  tinyint
    ,@categoriapro varchar(20)
    ,@color     varchar(20)
    ,@idproducto smallint OUT
AS
INSERT INTO
    producto
    (
        nombrepro
        ,diseño
        ,material
        ,descripcion
        ,precio
        ,tamaño
        ,stockpro
        ,categoriapro
        ,color
    )
VALUES(
    @nombrepro
```

Figure 27 Procedimiento Almacenado Registrar Producto

- **Vista:** Es la capa responsable del interfaz gráfico de usuario y la detección de eventos sobre los componentes.



The image shows a software interface for registering a product. It features a dark background with light gray input fields and buttons. The fields are arranged in a vertical list on the left, with labels in Spanish. The 'Stock' field is a numeric spinner set to 15. On the right, a box titled 'Operaciones:' contains four buttons: 'Guardar' (Save), 'Nuevo' (New), 'Eliminar' (Delete), and 'Buscar' (Search). The 'Nuevo' button is highlighted with a red border.


| Field Label | Field Type      | Value / Content |
|-------------|-----------------|-----------------|
| Categoría   | Dropdown        | [Empty]         |
| Stock       | Numeric Spinner | 15              |
| Nombre      | Text            | [Empty]         |
| Diseño      | Text            | [Empty]         |
| Material    | Text            | [Empty]         |
| Tamaño      | Dropdown        | [Empty]         |
| Color       | Dropdown        | [Empty]         |
| Precio      | Text            | [Empty]         |
| Descripción | Text            | [Empty]         |

Operaciones:




- Guardar
- Nuevo
- Eliminar
- Buscar

*Figure 28 Vista – Interfaz Registrar Producto*

Branch: master [PROCESOS-DE-SOFTWARE / INCREMENTO 1 / registrar\\_producto\\_interfaz](#) [Find file](#) [Copy path](#)

 70041970M Create registrar\_producto\_interfaz e18ec92 3 minutes ago

1 contributor

244 lines (181 sloc) | 7.29 KB [Raw](#) [Blame](#) [History](#)   

```

1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel;
4  using System.Data;
5  using System.Drawing;
6  using System.Linq;
7  using System.Text;
8  using System.Threading.Tasks;
9  using System.Windows.Forms;
10
11 using Rodriguez.BusinessLogicLayer;
12 using Rodriguez.Utilities;
13 using Rodriguez.Entitites;
14
15 namespace Rodriguez.PresentationLayer
16 {
17     public partial class FrmMantenimientoProducto : Form
18     {
19         public FrmMantenimientoProducto()
20         {
21             InitializeComponent();
22         }
23
24         private void btnGuardar_Click_1(object sender, EventArgs e)
25         {
26             try
27             {
28                 Int16 id;
29                 Int16.TryParse(txtIdProd.Text, out id);
30
31                 ///---creamos un objeto cliente
32                 Producto producto = new Producto();
33
34                 ///--- llenar de datos al objeto cliente
35                 producto.Categoria = cmbCategoria.Text;
36
37                 producto.Nombre = txtNombreProd.Text;
38                 producto.Diseño = txtDiseño.Text;
39                 producto.Material = txtMaterial.Text;
40                 producto.Descripcion = txtDescripcionProd.Text;
41                 // convertidor
42
43                 Int16 stock;
44                 Int16.TryParse(nudStock.Text, out stock);

```

*Figure 29 Código en GitHub de la interfaz Registrar Producto*

- **Controlador:** Es quien hace corresponder la interacción del usuario con los posibles cambios en el modelo.

```
72         else
73         {
74
75             //----actualizado
76
77             producto.Idproducto = id;
78             Int32 resp = productoBLL.actualizar(producto);
79
80             MessageBox.Show("PRODUCTO ACTUALIZADO SATISFACTORIAMENTE", "Información", MessageBoxButtons.OK, MessageBoxIcon.Informat
81         }
82
83     }
84
85
86     catch (Exception ex)
87     {
88
89         MessageBox.Show(ex.Message + "-" + ex.StackTrace.ToString());
90     }
91
92 }
93
94 private void btnNuevo_Click(object sender, EventArgs e)
95 {
96     DialogResult result = MessageBox.Show("¿Desea registrar nuevo Producto?", "Nuevo Registro", MessageBoxButtons.YesNo, MessageBox
97     if (result == DialogResult.Yes)
```

*Figure 30 Controlador – Registrar Producto*

### 3.9 EJECUCIÓN DEL PLAN DE PRUEBAS

*Tabla 3 Ejecución de pruebas*

| CASO DE USO  | REGISTRAR USUARIO  |
|--|--|
| Datos normales   | Resultado Esperado   |
|  |  |
| Datos anormales  | Resultado obtenido   |

### DNI incompleto

Menu Administrador  
Mantenimiento Proceso Reportes Seguridad Buscar

Mantenimiento Empleado

DNI: 7004197

Nombres: milagros

Apellidos: medrano

Cargo: VENDEDOR

Direccion: av. huancavelica

Teléfono: 984563215

Fecha Nacimiento: 17/07/1996

Contraseña: 1017905

Fecha Registro: 23/05/2020

Operaciones:  
Guardar  
Nuevo  
Eliminar  
Buscar

Menu Administrador  
Mantenimiento Proceso Reportes Seguridad Buscar

Mantenimiento Empleado

DNI: 7004197

Nombres: milagros

Apellidos: medrano

Cargo: VENDEDOR

Direccion: av. huancavelica

Teléfono: 984563215

Fecha Nacimiento: 17/07/1996

Contraseña: 1017905

Fecha Registro: 23/05/2020

Operaciones:  
Guardar  
Nuevo  
Eliminar  
Buscar

El valor de DNI tiene formato erroneo

Aceptar

### CASO DE USO

### AUTENTICACIÓN

### Datos normales

Resultado Esperado: Ingresar al sistema

Login

DNI: 40712186

Clave: \*\*\*\*\*

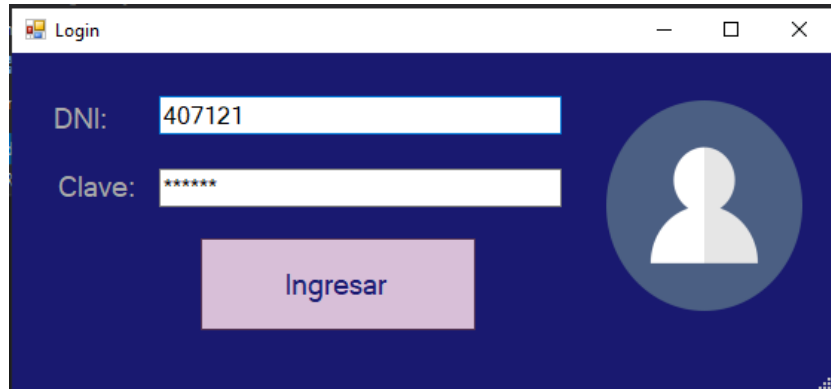
Ingresar

Menu Administrador  
Mantenimiento Proceso Reportes Seguridad Buscar

DNI: 40712186 APELLIDO: RODRIGUEZ SUAREZ NOMBRE: CARLOS ABRAHAM CARGO: ADMINISTRADOR

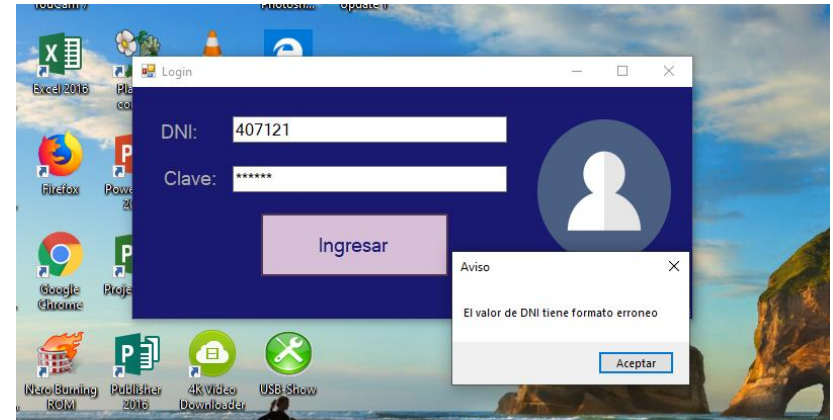
### Datos anormales

Resultado Obtenido



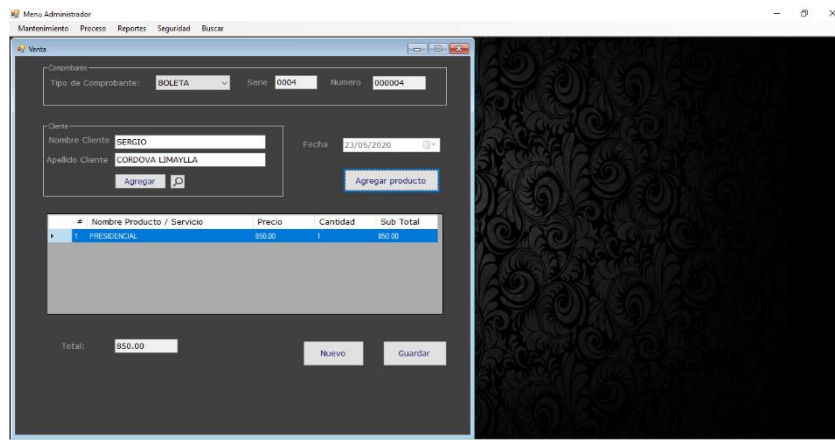
**CASO DE USO**

**Datos normales**

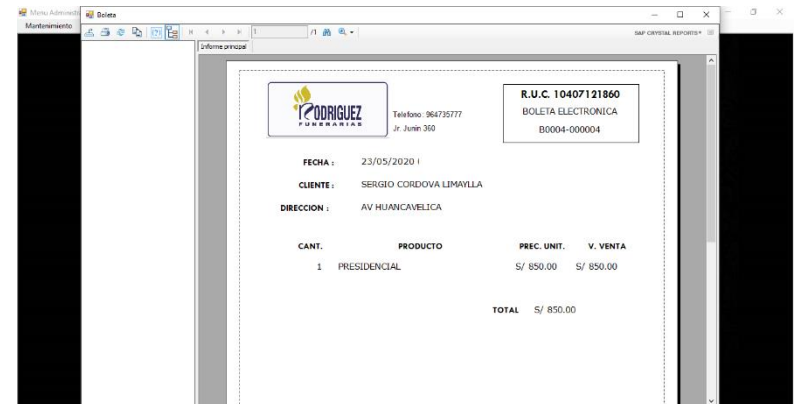


**EMITIR BOLETA DE VENTA**

**Resultado esperado**

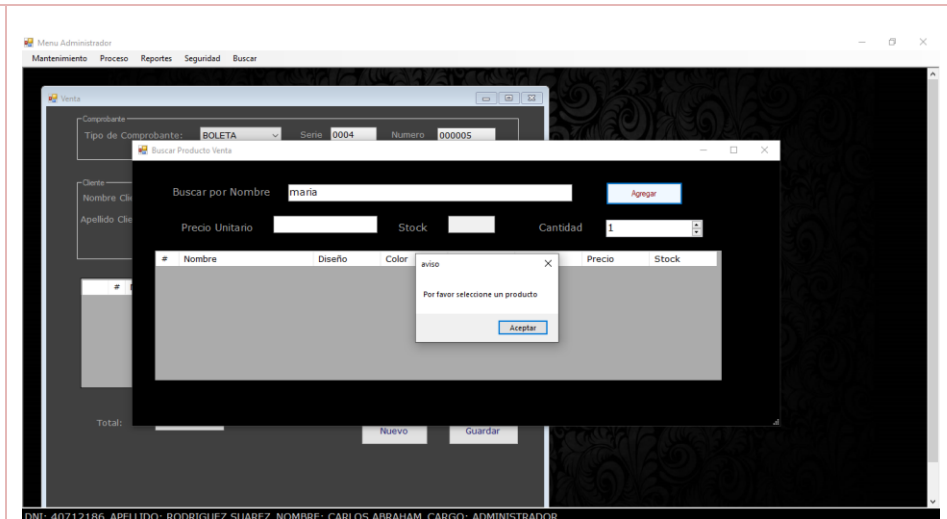
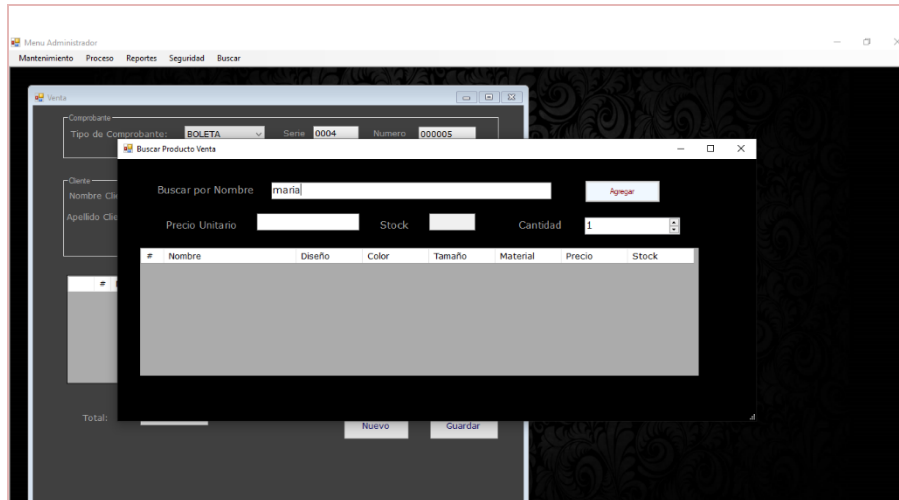


**Datos anormales**



**Resultado Obtenido**





## CASO DE USO

**Datos normales:** Búsqueda por fecha

## REPORTE DE VENTAS

Resultado esperado: Muestra las ventas realizadas




| NOMBRE       | DISEÑO   | MATERIAL | TAMAÑO | PRECIO   |
|--------------|----------|----------|--------|----------|
| LADY DIANA   | COFRE    | CAOBA    | ADULTO | 700.00   |
| ONIX         | PIRAMIDE | MARMOL   | --     | 250.00   |
| PRESIDENCIAL | BIBLIA   | PINO     | ADULTO | 850.00   |
| OSIRIS       | ABOMBADO | CERAMICA | --     | 250.00   |
| ESPAÑOL      | JAPONES  | NORDEX   | NIÑO   | 1,000.00 |
| EGIPCIO      | PIRAMIDE | CERAMICA | --     | 150.00   |

**Datos anormales: Fechas donde no se realizaron ventas**



Resultado Obtenido: No se encuentra ningun registro de ventas en la fecha establecida

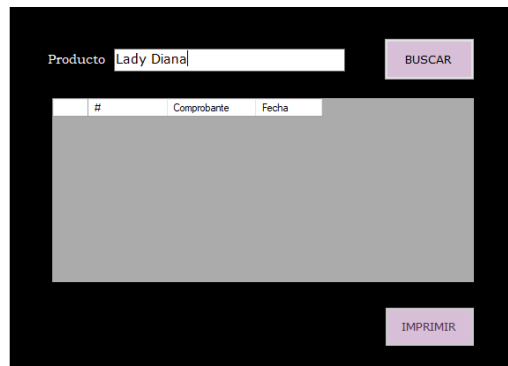
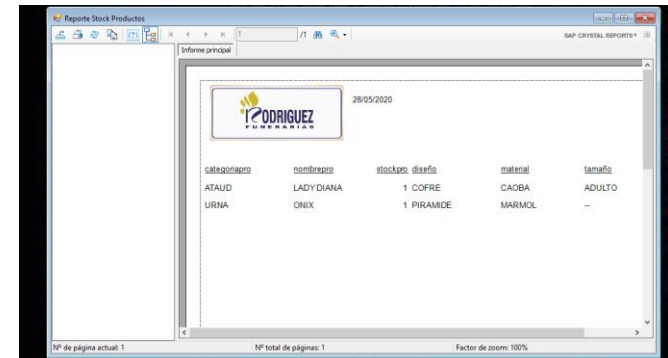


**CASO DE USO**

Reporte Stock

**Datos normales**

Resultado esperado

**Datos anormales:** Nombre de producto no registrado

Resultado Obtenido: No se obtiene registro en a busqueda

Producto

Extra grande

BUSCAR

| # | Comprobante | Fecha |
|---|-------------|-------|
|---|-------------|-------|

IMPRIMIR

Producto

Extra grande

BUSCAR

| # | Comprobante | Fecha |
|---|-------------|-------|
|---|-------------|-------|

IMPRIMIR

## CASO DE USO

Reporte Top

### Datos normales: Búsqueda por fecha

Resultado esperado: Lista de productos mas vendidos por fechas

The screenshot shows a web application window titled "Listar Productos por Fecha". At the top, there are two date input fields: "Desde: 23/05/2020" and "Hasta: 25/05/2020". To the right of these fields is a button labeled "BUSCAR". Below the date fields is a table with the following columns: "#", "Nombre", "Diseño", "Material", "Categoría", "Color", and "Cantidad". The table body is currently empty. At the bottom right of the application area is a button labeled "IMPRIMIR".

Listar Productos por Fecha

Desde:

23/05/2020

Hasta:

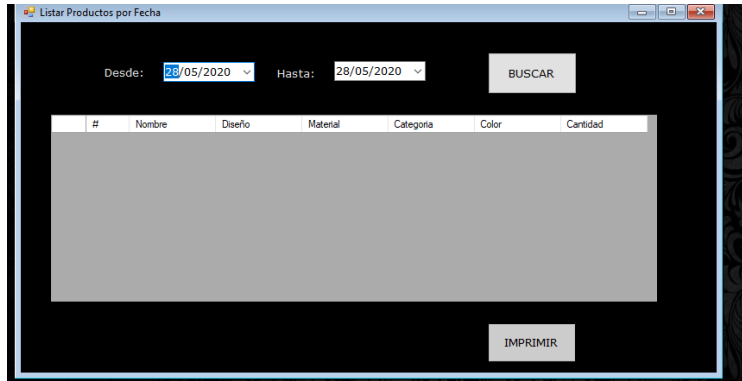
25/05/2020

BUSCAR

| #   | Nombre       | Diseño | Material | Categoría | Color  | Cantidad |
|-----|--------------|--------|----------|-----------|--------|----------|
| ▶ 0 | PRESIDENCIAL | BIBLIA | PINO     | ATAUD     | MARRON | 1        |

IMPRIMIR

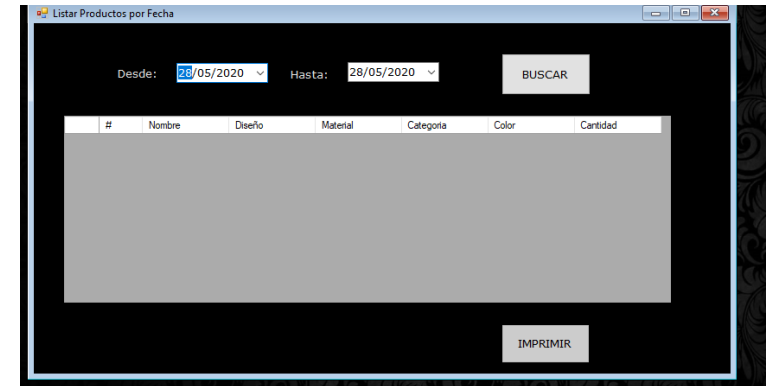
**Datos anormales:** Ingresar fechas donde no se realizo ninguna venta



**CASO DE USO**

**Datos normales**

**Resultado Obtenido:** No muestra ningun produto top



**Registrar Producto**

**Resultado esperado**

The screenshot shows the 'Mantenimiento Producto' form with the following data:
 

- Categoría: ATAUD (dropdown)
- Stock: 7 (spinner)
- Nombre: LADY DIANA II (text)
- Diseño: Antiguo (text)
- Material: Caoba (text)
- Tamaño: ADULTO (dropdown)
- Color: MARRON (dropdown)
- Precio: 850 (text)
- Descripción: (empty text area)
- Operaciones: Guardar, Nuevo, Eliminar, Buscar (buttons)

**Dato anormales**

The screenshot shows the 'Mantenimiento Producto' form with the same data as the previous one. A modal dialog box titled 'Información' is displayed in the center, containing the message 'PRODUCTO REGISTRADO SATISFACTORIAMENTE' and an 'Aceptar' button.

**Resultado obtenido**

The screenshot shows the 'Mantenimiento Producto' form with the following data:
 

- Categoría: ATAUD (dropdown)
- Stock: 7 (spinner)
- Nombre: y (text)
- Diseño: s (text)
- Material: a (text)
- Tamaño: ADULTO (dropdown)
- Color: MARRON (dropdown)
- Precio: 0 (text)
- Descripción: (empty text area)
- Operaciones: Guardar, Nuevo, Eliminar, Buscar (buttons)

The screenshot shows the 'Mantenimiento Producto' form with the same data as the previous one. A modal dialog box titled 'Información' is displayed in the center, containing the message 'PRODUCTO NO REGISTRADO' and an 'Aceptar' button.

## **CAPITULO IV**

### **CONCLUSIONES Y RECOMENDACIONES**

#### **4.1. CONCLUSIONES**

1. La implementación del primer entregable incluye las necesidades registradas en los requerimientos funcionales de la empresa, para el registro de pedidos, permitiendo almacenar la información en una base de datos.
2. El desarrollo de ciclo de vida del software se enfoca en el modelo incremental, al entregarle el primer incremento, el cliente podrá hacer uso del sistema, dando posibilidad de mejorar el software en una siguiente versión.
3. A nivel de proyecto, a razón de tener la capacidad de manejar la calendarización de la elaboración del producto, el proyecto puede ser evaluado para que se vuelva un buen recurso, no solo en la empresa trabajada, sino también luego de la consolidación del proyecto, este podría generar más interés en otros sectores de negocio establecidos en la ciudad de Huancayo.
4. El sistema permite la consulta de los precios de los productos que están en ofertas o promociones haciendo más eficaz el proceso de cotización. Por lo tanto, dicho sistema constituye una herramienta importante para la gestión y toma de decisiones de la empresa.
5. Durante la construcción del Sistema se realizaron pruebas para garantizar la calidad del funcionamiento, la seguridad, el rendimiento y la interfaz gráfica basándose en un plan de pruebas enfocado por iteraciones.

#### **4.2. RECOMENDACIONES**

1. Para complementar y facilitar la gestión en la empresa, el sistema en un futuro puede ser complementado o asociado a otros sistemas como el de gestión de proveedores, que permita un registro más detallado de materiales y productos utilizados de terceros.
2. El proyecto puede ser ampliado logrando que su alcance sea aún mayor y poder almacenar la información en la nube, es necesario que el sistema se adecue a mejores niveles de seguridad que aseguren que la información no sea corrompida o sea eliminada por terceros.
3. Si bien por inherencia, las tareas que realiza el programa lo vuelven más flexible a los cambios y por ellos puede ser llevado a otros sectores manufactureros para un mejoramiento en la organización que se use esto puede requerir más tiempo para su desarrollo.
4. Es recomendable que, en un futuro, el sistema se conecte a un sistema web que permita las ventas online y se puede manejar desde un servidor web.
5. Se recomienda que todos los productos que se almacenen coincidan con el catálogo de productos que ofrece la empresa, dando prioridad al nombre del producto.

**ÍNDICE DE ABREVIATURAS Y SIGLAS****C****CMMI**

Capability Maturity Model Integration (Modelo de Madurez de Capacidades de Integración)

**G****GUS**

(Gestión de Configuración de Software).

**M****MVC**

Model-View-Controller (Modelo-Vista-Controlador)

**SDLC**

Systems Developed Life Cycle (Ciclo de vida de desarrollo de software)

**SQL**

Structured Query Language (lenguaje de consulta estructurada)

**U****UML**

Unified Modeling Language (Lenguaje Unificado de Modelado)



## APÉNDICES

### APÉNDICES DE FOTOS

- Integrantes del proyecto



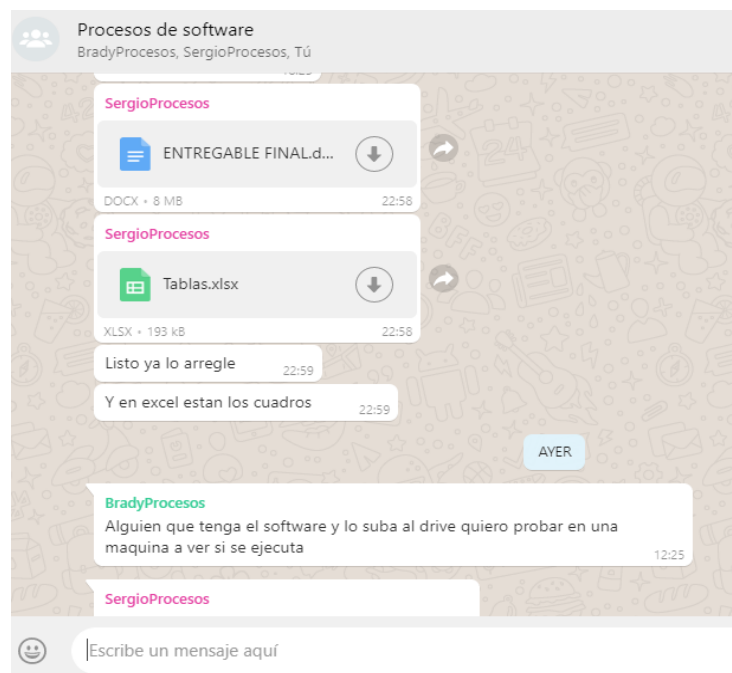
Condori Ochoa Brady



Córdova Limaylla Sergio



Arauco Medrano Merly



Grupo de WhatsApp con los  
integrantes del grupo

<> Código
Cuestiones 00
Solicitudes de extracción 00
Comportamiento
Proyectos 00
Seguridad 0

Percepciones
Configuraciones

Opciones
Administrar acceso
Webhooks
Notificaciones
Integraciones
Implementar claves
Misterios
Comportamiento

### Quien tiene acceso







REPOSITORIO PRIVADO
Solo aquellos con acceso a este repositorio pueden verlo.
Gestionar

ACCESO DIRECTO
2 tienen acceso a este repositorio.
2 invitaciones.

### Administrar acceso

Invita a un colaborador

☒ 2 miembros seleccionados ...
Tipo ▾

|                                     |  |  |   |
|-------------------------------------|--|--|---|
| <input checked="" type="checkbox"/> |  <b>70041970</b><br>En espera de la respuesta de 70041970                           | Invitación pendiente  |  |
| <input checked="" type="checkbox"/> |  <b>SergioCordovaLimaylla</b><br>En espera de la respuesta de SergioCordovaLimaylla | Invitación pendiente  |  |

## Trabajo remoto por medio de GitHub

**REFERENCIAS BIBLIOGRAFICAS**

**Anonimo. 2017.** *Aprender CMMI.* 2017.

**GitHub. 2019.** *¿Qué es GitHub?* 2019.

**Macías, García Manuel. 2007.** *GUIA PARA LA IDENTIFICACIÓN Y ANÁLISIS DE PROCESOS.* Cadiz : s.n., 2007.

**Manuel, Macías García. 2017.** *GUIA PARA LA IDENTIFICACION Y ANALISIS DE PROCESOS.* CADIZ : s.n., 2017.

**Microsoft. 2019.** *Introducción al lenguaje c#.* 2019..

**MICROSOFT. 2019.** *PROCEDIMIENTOS ALMACENADOS.* 2019.

**Microsoft. 2019.** *Visual Basic.* 2019.

**Morejón, Manuel. 2015.** *Érase una vez Kubernetes.* 2015.

**Sevilla, Universidad de. 2013.** *El ciclo de vida del Software.* sevilla : s.n., 2013.

**Yenisleidy, Romero Fernandez. 2012.** *El patrón Modelo-Vista-Controlador (MVC).* La Habana : Revista Telematica, 2012.