



UNIVERSIDAD  
PRIVADA  
DEL NORTE

# FACULTAD DE INGENIERÍA

---

CARRERA DE INGENIERÍA DE SISTEMAS COMPUTACIONALES

## **Sistema de ventas e inventario para mejorar el proceso comercial de "Maderera Kiba"**

### **Informe académico**

#### **Autor(es):**

Camila Tamara Cruz Diaz  
Jean Carlo Chirinos Avendaño  
Kenko Enrique Garcia Padilla  
Cesar Cornejo Turpo

#### **Curso:**

Modelamiento Y Análisis De Software

#### **Docente:**

Jorge Alfredo Guevara Jimenez

LIMA – PERÚ  
2020-2

## Contenido

<b>CAPÍTULO 1.</b>	<b>5</b>
1. Datos de la empresa.	5
2. Identificación del problema y enunciado del problema.	5
3. Planteamiento de la solución.	5
3.1. Como implementar la solución.	5
4. Justificación de la investigación.	6
5. Limitaciones de la investigación.	6
5.1. Limitación espacial:	6
5.2. Limitación de recursos:	6
5.3. Limitación temporal:	6
6. Ventajas de la solución.	6
7. Desventajas de la solución.	6
8. Objetivos del proyecto de investigación.	7
<b>CAPÍTULO 2.</b>	<b>8</b>
1. Marco Teórico.	8
1.1. Definición de software.	8
1.2. Modelo de proceso de software.	8
1.3. Notación UML.	8
1.4. Diagrama de caso de uso.	9
1.5. Diagramas de actividades.	9
1.6. Proceso de negocio.	10
1.7. GitHub	10
1.8. NetBeans	10
1.9. Java.	11
1.10. Especificación de casos de uso.	11
1.11. Requerimientos del sistema.	11
1.12. MySQL.	12
1.13. Relaciones entre casos de uso.	12
2. Marco metodológico.	13
2.1. Paso 1: Identificar la problemática.	13
2.2. Paso 2: Identificar los actores afectados.	13
2.3. Paso 3: Plantear una solución hacia la problemática.	13
2.4. Paso 4: Establecer las herramientas y medios a usar.	13
2.5. Paso 5: Establecer los diseños necesarios.	13
2.6. Paso 6 Desarrollar la propuesta de solución.	13
2.7. Paso 7: Codificar los interfaces en NetBeans.	14

2.8.	Paso 8: Implementar la base de datos en MySQL Workbench .....	14
2.9.	Paso 9: Probar el funcionamiento y eficacia del proyecto. ....	14
2.10.	Paso 10: Implementar el proyecto. ....	14
<b>CAPITULO 3. ....</b>		<b>15</b>
<b>1.</b>	<b>Implementación de la solución planteada. ....</b>	<b>15</b>
1.1.	Modelado del proceso de negocio. ....	15
1.2.	Especificación de proceso de negocio. ....	16
1.3.	Proceso de software. ....	17
1.4.	Diagrama de casos de uso relacionado. ....	17
1.5.	Especificación de casos de uso. ....	18
1.6.	Diagrama de actividades. ....	20
1.7.	Prototipos de software. ....	21
1.8.	Lista de requerimientos funcionales y no funcionales. ....	26
1.9.	Requerimientos no funcionales instalados. ....	27
<b>CAPITULO 4. ....</b>		<b>30</b>
<b>1.</b>	<b>Conclusiones y recomendaciones. ....</b>	<b>30</b>
1.1.	Conclusiones. ....	30
1.2.	Recomendaciones. ....	30
<b>2.</b>	<b>Referencias bibliográficas. ....</b>	<b>30</b>
<b>3.</b>	<b>Anexos. ....</b>	<b>31</b>
3.1.	Acta de aceptación del proyecto. ....	31
3.2.	Foto grupal. ....	32
3.3.	Repositorio en GitHub. ....	33
3.4.	Script de la versión 1 del software. ....	35
3.5.	Secuencia de funcionamiento de la versión 1. ....	36

## Índice de ilustraciones.

Ilustración 1 Modelado del proceso de negocio .....	15
Ilustración 2 Especificación de proceso de negocio .....	16
Ilustración 3 Proceso de software .....	17
Ilustración 4 Diagrama de casos de uso relacionado .....	17
Ilustración 5 Diagrama de actividades.....	20
Ilustración 6 Ventana de selección de usuario .....	21
Ilustración 7 Ventana de Login .....	21
Ilustración 8 Ventana de registro de proveedores .....	22
Ilustración 9 Ventana de registro de usuarios .....	22
Ilustración 10 Ventana de registro de pedidos .....	23
Ilustración 11 Ventana de registro de productos .....	23
Ilustración 12 Ventana de reporte de proveedores.....	24
Ilustración 13 Ventana de reporte de usuarios .....	24
Ilustración 14 Ventana de reporte de pedidos .....	25
Ilustración 15 Ventana de reporte de productos.....	25
Ilustración 16 Entorno de desarrollo de Kenko Garcia .....	27
Ilustración 18 Entorno de desarrollo de Cesar Cornejo.....	28
Ilustración 17 Entorno de desarrollo de Jean Carlos Chirinos.....	28

## Índice de tablas.

Tabla 1 Especificación de caso de uso registrar usuario. ....	18
Tabla 2 Especificación de caso de uso registrar producto. ....	19
Tabla 3 Lista de requerimientos funcionales y no funcionales. ....	26

## CAPÍTULO 1.

### 1. Datos de la empresa.

La empresa la cual ha sido seleccionada para brindar la solución tecnológica es **"Maderera Kiba SRL."**, Una MYPE con número de RUC 20414289534, ubicada en la Avenida Pachacutec N° 1945, Villa Maria del Triunfo. Cuyo representante legal Jorge Enrique Garcia Sandoval accedió a brindar la información requerida para este proyecto.

### 2. Identificación del problema y enunciado del problema.

En la Maderera "Kiba" se establece una falta de un sistema de ventas, lo que determina el uso de métodos tradicionales de administración de ventas e inventario. Esto se debe al escaso uso de nuevos modelos de negocio que son utilizados por los nuevos emprendimientos o empresas más grandes como herramienta fundamental a través de las tecnologías de información. Al no usar estos medios, se dificulta el orden y control de los productos que ofrece la empresa a sus clientes, además de ralentizar el proceso de ventas y de permitir el uso versátil de las herramientas informáticas que exige actualmente el mercado, lo cual disminuye el potencial de crecimiento de la empresa.

### 3. Planteamiento de la solución.

Existen diversos programas de gestión ya existentes, que tienen un costo medio, que permiten la administración de inventarios y ventas que funcionan de tanto de forma local como en la nube. Sin embargo, estos mismos no cubren las necesidades del negocio, ya que, se centra en una materia prima muy específica y que tiene muchas regulaciones gubernamentales. Por ello, se determinó que la manera optimizar dichos procesos de gestión es creando un software local conectado a una base de datos desde cero.

#### 3.1. Como implementar la solución.

Para implementar la solución, se estableció diversos diseños de prototipos con la herramienta Balsamiq, que presenta funciones principales, las cuales

son registrar, crear, consultar para luego llevarlo a cabo en forma de código a través de la IDE Apache NetBeans.

#### 4. Justificación de la investigación.

Actualmente, el desarrollo de las tecnologías de la información estructura nuevos modelos de negocio, en los cuales el software tiene un factor primordial como mejora efectiva de los métodos tradicionales.

La Maderera Kiba debe implementar un sistema de ventas el cual permita optimizar su proceso comercial mediante tecnologías de la información. Esta implementación ayudará a cambiar el esquema de negocio actual de la empresa y permitirá abrir las posibilidades de mejora ante sus competidores.

#### 5. Limitaciones de la investigación.

##### 5.1. Limitación espacial:

- El proyecto de investigación se desarrolla de forma remota a través de GitHub.

##### 5.2. Limitación de recursos:

- Se toma en cuenta los recursos de hardware disponibles para la implementación del proyecto.

##### 5.3. Limitación temporal:

- Se establece un plazo de tiempo estimado de tres meses.

#### 6. Ventajas de la solución.

- Favorece el manejo de información.
- Ofrece múltiples opciones de registro y consulta.
- Ofrece una base de datos confiable y segura para el inventariado de los productos.
- Optimiza el manejo de la cartera de clientes de la empresa.

#### 7. Desventajas de la solución.

- Es una solución local, por lo tanto, no tiene un soporte o respaldo en la nube inmediato.
- Es posible que el terminal presente fallas de hardware o del sistema operativo.

## 8. Objetivos del proyecto de investigación.

### 8.1. Objetivo general.

Implementar un sistema de ventas e inventarios local que optimice el proceso comercial de Maderera Kiba.

### 8.2. Objetivos específicos.

- Mejorar la eficiencia del proceso de inventariado.
- Reducir los tiempos de administración del stock de productos.
- Implementar un inventario que permita realizar consultas y modificaciones.
- Implementar un registro de usuarios que permita realizar consultas y modificaciones.

## **CAPÍTULO 2.**

### **1. Marco Teórico.**

#### **1.1. Definición de software.**

En la mayoría de los casos asociamos la definición de software a los programas que se pueden encontrar y visualizar dentro de una computadora y si bien este concepto que se tiene no es algo erróneo, la definición del mismo abarca algo más que eso. Sommerville (2005) considera que la definición apropiada para el término software abarca algo más amplio que solo ser un programa, puesto que, para él, son "todos los documentos asociados y la configuración de datos que se necesitan para hacer que estos programas [puedan operar] de manera correcta" (p.5).

#### **1.2. Modelo de proceso de software.**

Sommerville (2005) lo define como "una descripción simplificada de un proceso del software que presenta una visión de ese proceso" (p.8). Por lo que hay que tener en cuenta realizarlo de una forma que el cliente pueda entender lo que está destinado a hacerse de una forma eficaz. Estos modelos están basados en uno de los tres modelos generales del desarrollo de software, los cuales son: El enfoque de cascada, el desarrollo iterativo y la ingeniería del software basada en componentes (CBSE).

#### **1.3. Notación UML.**

La notación UML, cuyas siglas en inglés significan Unified Modeling Language, es un lenguaje de modelado de sistemas de software con mayor uso en la actualidad. Esta notación es la unificación de la técnica de modelado de objetos o OMT, Booch y la ingeniería de software orientada a objetos (OOSE). Para (Fowler, 1999) "el lenguaje de modelado es la notación gráfica (principalmente gráfica) de que se valen los métodos para expresar los diseños" (p.1). Por lo que esta notación nos permite visualizar y elaborar el sistema de una manera más gráfica y dinámica, siendo más



"amigable" para los ojos de los clientes e incluso, los mismos desarrolladores.

#### 1.4. Diagrama de caso de uso.

Este tipo de diagrama ayuda a representar la funcionalidad que tiene el sistema desde la vista del usuario. En ella se definen los actores, los límites y como su nombre lo dice, los casos. Los actores son las entidades que interactúan de forma externa con el sistema. Presentan nombres únicos y están vinculados a los casos de uso mediante conectores. Estos pueden estar relacionados con otros actores mediante el tipo de relación herencia. Los casos de uso describen las funciones que el sistema les proporciona a los actores externos. Según (Fowler, 1999) "[un caso de uso] se obtiene hablando con los usuarios habituales y analizando con ellos las distintas cosas que deseen hacer con el sistema" (p.49). Por lo que se podría decir que los casos de usos son los requerimientos del sistema mismo.

#### 1.5. Diagramas de actividades.

Una definición precisa es la que realizan Bruegge y Dutoit (2002) ya que describen a este tipo de diagrama como "un sistema desde el punto de vista de las actividades [...] [cuyas finalizaciones disparan] una transición hacia otra actividad" (p.28). Estos diagramas tienen características similares a los diagramas de flujo, puesto que se pueden emplear para representar los flujos de control y de datos. Los disparadores que inician al diagrama son representados mediante un círculo pintado. Por otro lado, los que hacen finalizar al diagrama, son representados por un círculo pintado que se encuentra dentro de otro círculo sin pintar.

Cada diagrama tiene estados de actividad representados mediante rectángulos con puntas redondeadas y estos pasan a otros por medio de transiciones. El flujo de control no siempre es secuencial, por lo que también se puede llegar a encontrar (y realizar) bifurcaciones de acuerdo a las necesidades. A su vez, existen actividades que requieren ser realizadas al mismo tiempo, para ello se emplea la división y unión, que son

representadas mediante una línea horizontal. Finalmente, todas las actividades deben de estar divididas en el diagrama conforme a los actores responsables de su realización, estas divisiones son denominadas calles.

#### 1.6. Proceso de negocio.

Según (Sparks, 2000) "Un proceso de negocio es una colección de actividades diseñadas para producir una salida específica para un cliente o un mercado en particular". Esto implica un fuerte énfasis en cómo se realiza el trabajo dentro de una organización, en contraposición con un enfoque del producto en qué se produce.

Especificación de proceso de negocio: Parafraseando (Molina, 2007) Es un diagrama de procesos que describirán un objetivo ligado al proceso de negocio original, de modo que los procesos de negocio se organizan jerárquicamente.

#### 1.7. GitHub

GitHub es una forja (plataforma de desarrollo colaborativo) para alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente de programas de ordenador. El software que opera GitHub fue escrito en Ruby on Rails. Desde enero de 2010, GitHub opera bajo el nombre de GitHub, Inc. Anteriormente era conocida como Logical Awesome LLC. El código de los proyectos alojados en GitHub se almacena típicamente de forma pública.

#### 1.8. NetBeans

NetBeans es un proyecto de código abierto de gran éxito con una gran base de usuarios, una comunidad en constante crecimiento.

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las API de NetBeans y un archivo especial (manifest file) que

lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software. El NetBeans IDE soporta el desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones móviles). Entre sus características se encuentra un sistema de proyectos basado en Ant, control de versiones y refactoring.

### 1.9. Java

Según (Schildt 2009) "Java es una mezcla de los mejores lenguajes de programación exitosos"(p3.). Como describe el autor, Java surge ante una necesidad, los lenguajes de programación existentes de la época dependían de una arquitectura específica para ser compilados, en cambio Java no, ya que esta se ejecuta en lo que se conoce como máquina virtual de Java. Este lenguaje se utiliza principalmente en la programación orientada a objetos debido a su flexibilidad y rendimiento.

### 1.10. Especificación de casos de uso.

Las especificaciones de los casos de uso describen el rol y objetivo de cada caso. Para (Bruegge & Dutoit, 2002) "Para describir un caso de uso se usa una plantilla compuesta por ocho campos. El nombre, los actores, las condiciones iniciales, el disparador, el flujo de eventos, las condiciones de salida, los requerimientos especiales y la prioridad"

### 1.11. Requerimientos del sistema.

Los requerimientos de un sistema se definen como funciones específicas para identificar las necesidades de un sistema. Según (Bruegge & Dutoit, 2002) "las definiciones de requerimientos del sistema especifican qué es lo que el sistema. Esto requiere consultar con los clientes del sistema y los usuarios finales".

### 1.12. MySQL.

Según (Oracle 2012) "El servicio de base de datos MySQL es un servicio de base de datos totalmente administrado para implementar aplicaciones nativas de la nube utilizando la base de datos de código abierto más popular del mundo."

### 1.13. Relaciones entre casos de uso.

Así como puede llegar a existir relaciones entre los actores, los casos también pueden llegar a relacionarse entre sí. Los tipos de relaciones que se tienen son la inclusión, la extensión y la generalización.

Las relaciones de inclusión se dan entre dos casos de usos cuando uno de ellos incluye al otro en su flujo de eventos. Se representan mediante la etiqueta <<include>> o <<incluye>> y una línea punteada.

La extensión es un tipo de relación en la que se cumple siempre y cuando una operación establecida previamente se ejecute en un caso, en pocas otras cuando se cumpla una condición. Al igual que la inclusión se representa mediante una línea punteada, sin embargo; lleva la etiqueta <<extend>> o <<extensión>>.

La generalización, al igual que los demás tipos de relaciones, ayuda a reducir la complejidad que un modelo puede llegar a tener. "Un caso de uso puede especializar a otro más general añadiendo más detalle" (Brueege & Dutoit, 2002, p.44). Se representa mediante una flecha blanca que apunta hacia el caso principal.

## 2. Marco metodológico.

### 2.1. Paso 1: Identificar la problemática.

Primero tendremos que identificar cual es el problema, además de examinar los posibles efectos negativos en ello, al que vamos a dar solución, este procedimiento debe ser flexible y generar numerosas opiniones para llevar a cabo la posible solución.

### 2.2. Paso 2: Identificar los actores afectados.

Luego de identificar la problemática, se debe identificar los actores implicados que serán afectados dependiendo de la solución, esto para adaptar el proyecto a las necesidades o requisitos que se necesitaran.

### 2.3. Paso 3: Plantear una solución hacia la problemática.

Al tener conocimiento del problema y las necesidades que requieren las personas involucradas se procede a plantear una posible solución, en este caso se realizara un proyecto de investigación grupal.

### 2.4. Paso 4: Establecer las herramientas y medios a usar.

Ya planteada la posible solución se establece que herramientas se van a usar y en que medios se realizaran ya sea la base de datos a usar, un editor de código fuente o software de ayuda.

### 2.5. Paso 5: Establecer los diseños necesarios.

Para empezar a trabajar el proyecto es necesario contar con unos diseños o "planos" para que sirvan de referencia, esto hace que la realización del proyecto sea eficiente.

### 2.6. Paso 6 Desarrollar la propuesta de solución.

Desde este paso se empieza a desarrollar el proyecto tomando en cuenta todos los parámetros de construcción y estándares admitidos que sean necesarios, además, se considerara las indicaciones de los pasos 4 y 5.

#### 2.7. Paso 7: Codificar los interfaces en NetBeans.

En este paso se realizará la codificación de la primera versión de la solución tecnológica, siguiendo las pautas establecidas en los pasos anteriores. Una vez terminado este paso se generará el archivo ejecutable de la versión.

#### 2.8. Paso 8: Implementar la base de datos en MySQL Workbench

Se realizará el modelado de la base de datos según los requerimientos del software.

#### 2.9. Paso 9: Probar el funcionamiento y eficacia del proyecto.

Una vez se concluye con el paso 6, 7 y 8, se procederá a realizar pruebas para verificar el adecuado funcionamiento y posibles errores del proyecto.

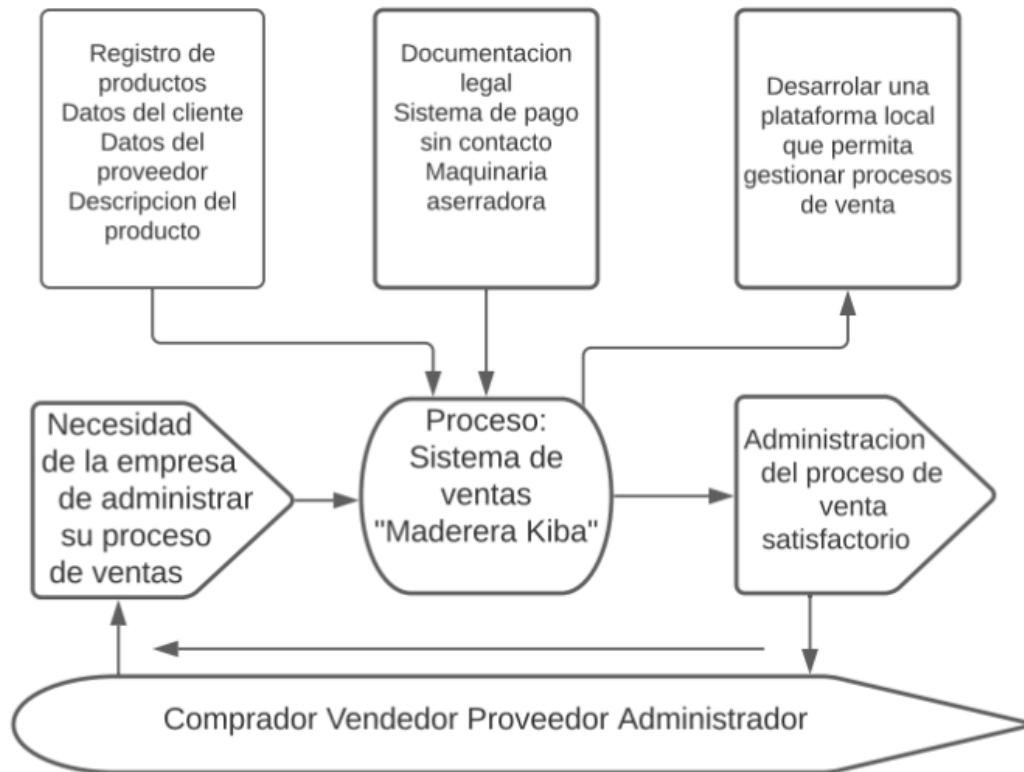
#### 2.10. Paso 10: Implementar el proyecto.

Finalmente, si se comprueba el correcto funcionamiento del proyecto en el paso 7 este estará listo para ser implementado y poder ser usado por el usuario.

### CAPITULO 3.

#### 1. Implementación de la solución planteada.

##### 1.1. Modelado del proceso de negocio.



*Ilustración 1 Modelado del proceso de negocio*

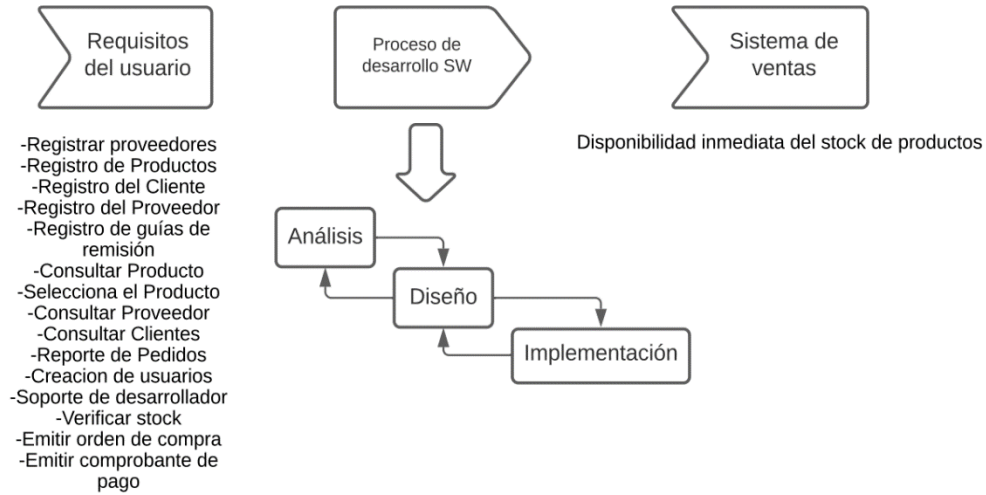
## 1.2. Especificación de proceso de negocio.

PROCESO	ENTRADA	ACTIVIDADES	SALIDA	VALOR PARA EL CLIENTE
Sistema de ventas "Maderera Kiba"	Necesidad de la empresa de administrar su proceso de ventas	Registrar proveedores Registrar productos Almacenar productos Registrar guías de remisión Ordenar los productos en el almacén Habilitar el producto para la venta Registrar clientes Registrar pedidos Verificar stock Emitir documentación legal Emitir comprobante de pago	Administración del proceso de venta satisfactorio	Implementación de una base de datos integrada y eficiente Disponibilidad inmediata del stock de productos Gestión efectiva de ingreso de productos Proceso de ventas rápido y confiable

*Ilustración 2 Especificación de proceso de negocio*

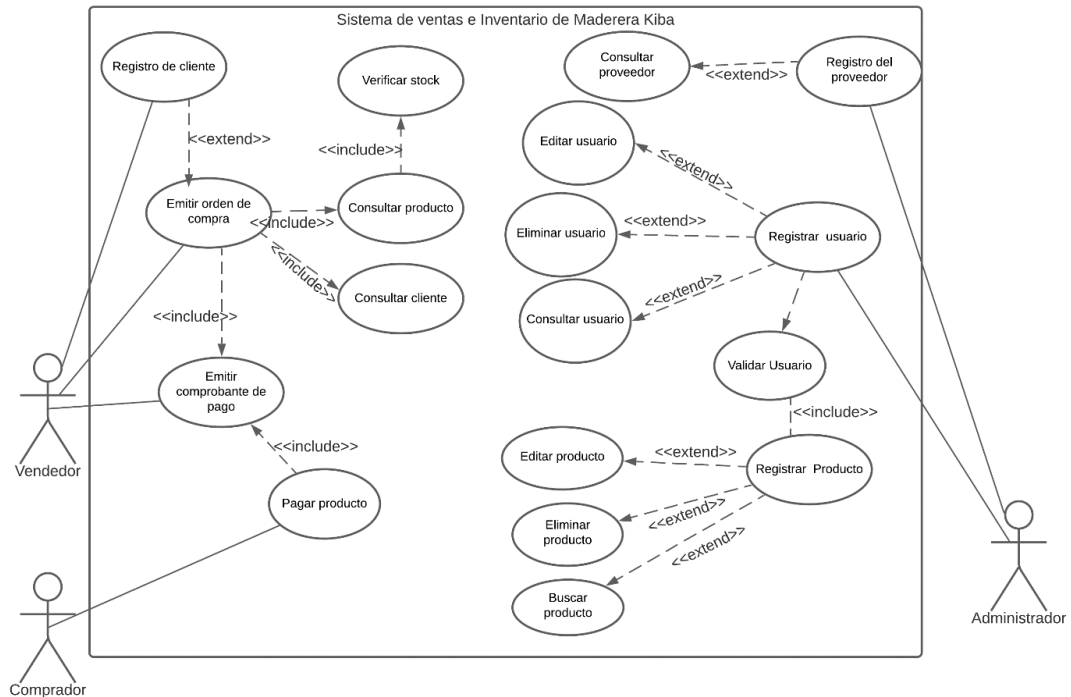


### 1.3. Proceso de software.



*Ilustración 3 Proceso de software*

### 1.4. Diagrama de casos de uso relacionado.



*Ilustración 4 Diagrama de casos de uso relacionado*

## 1.5. Especificación de casos de uso.

*Tabla 1 Especificación de caso de uso registrar usuario.*

ID	ITEM	DESCRIPCIÓN
1	Nombre corto	Registrar usuario
2	Actores	Administrador
3	Objetivo	Ingresar usuario nuevo
4	Disparador	Nuevo empleado de la empresa
5	Precondiciones	Validar usuario que realiza el registro
6	Post condiciones	Usuario registrado exitosamente
7	Escenario básico	<p>Escenario 1 : Validar usuario</p> <p>1.- Administrador ingresa nombre de usuario y contraseña 2.- Sistema valida nombre de usuario y contraseña.</p> <p>Escenario 2 : Editar usuario</p> <p>1.- Administrador selecciona usuario previamente registrado 2.- Administrador selecciona dato (nombre de usuario o contraseña) 3.- Administrador modifica la información del dato seleccionado 4.- Administrador confirma la modificación del dato del producto seleccionado</p> <p>Escenario 3 : Eliminar usuario</p> <p>1.- Administrador selecciona usuario previamente registrado 2.- Administrador selecciona usuario a eliminar 3.- Administrador confirma la eliminación del producto seleccionado</p> <p>Escenario 4 : Buscar usuario</p> <p>1.- Administrador ingresa un nombre de usuario previamente registrado</p>
8	Escenario alternativo	<ul style="list-style-type: none"> <li>• Si el nombre del usuario o la contraseña no es válido, se brinda opciones para recordar usuario o contraseña</li> <li>• Si los datos del usuario no están completos al registrar o editar, se solicita rellenar todos los campos.</li> <li>• Si se intenta registrar un usuario previamente registrado, se denegará el registro.</li> <li>• Si no se logra registrar, editar o eliminar el usuario, se envía un mensaje de error indicando la razón por la cual la operación falló.</li> <li>• Si no se encuentra el nombre de usuario ingresado, se solicita ingresar nuevamente la información</li> </ul>
9	Prioridad	Versión 1

Tabla 2 Especificación de caso de uso registrar producto.

ID	ITEM	DESCRIPCIÓN
1	Nombre corto	Registrar producto
2	Actores	Administrador
3	Objetivo	Ingresar producto nuevo
4	Disparador	Requerimiento de nuevo producto por parte del administrador
5	Precondiciones	Validar usuario que realiza el registro
6	Post condiciones	Producto registrado exitosamente
7	Escenario básico	<p>Escenario 1 : Validar usuario</p> <p>1.- Administrador ingresa nombre de usuario y contraseña 2.- Sistema valida nombre de usuario y contraseña.</p> <p>Escenario 2 : Editar producto</p> <p>1.- Administrador selecciona producto previamente registrado 2.- Administrador selecciona dato a editar del producto 3.- Administrador modifica la información del dato seleccionado 4.- Administrador confirma la modificación del dato del producto seleccionado</p> <p>Escenario 3 : Eliminar producto</p> <p>1.- Administrador selecciona producto previamente registrado 2.- Administrador selecciona producto eliminar 3.- Administrador confirma la eliminación del producto seleccionado</p> <p>Escenario 4 : Buscar producto</p> <p>1.- Administrador ingresa nombre o código de producto previamente registrado</p>
8	Escenario alternativo	<ul style="list-style-type: none"> <li>• Si el nombre del usuario o la contraseña no es válido, se brinda opciones para recordar usuario o contraseña</li> <li>• Si los datos del producto no están completos al registrar o editar, se solicita rellenar todos los campos.</li> <li>• Si se intenta registrar un producto previamente registrado, se denegará el registro.</li> <li>• Si no se logra registrar, editar o eliminar el producto, se envía un mensaje de error indicando la razón por la cual la operación falló.</li> <li>• Si no se encuentra el producto ingresado, se solicita ingresar nuevamente la información</li> </ul>
9	Prioridad	Versión 1

## 1.6. Diagrama de actividades.

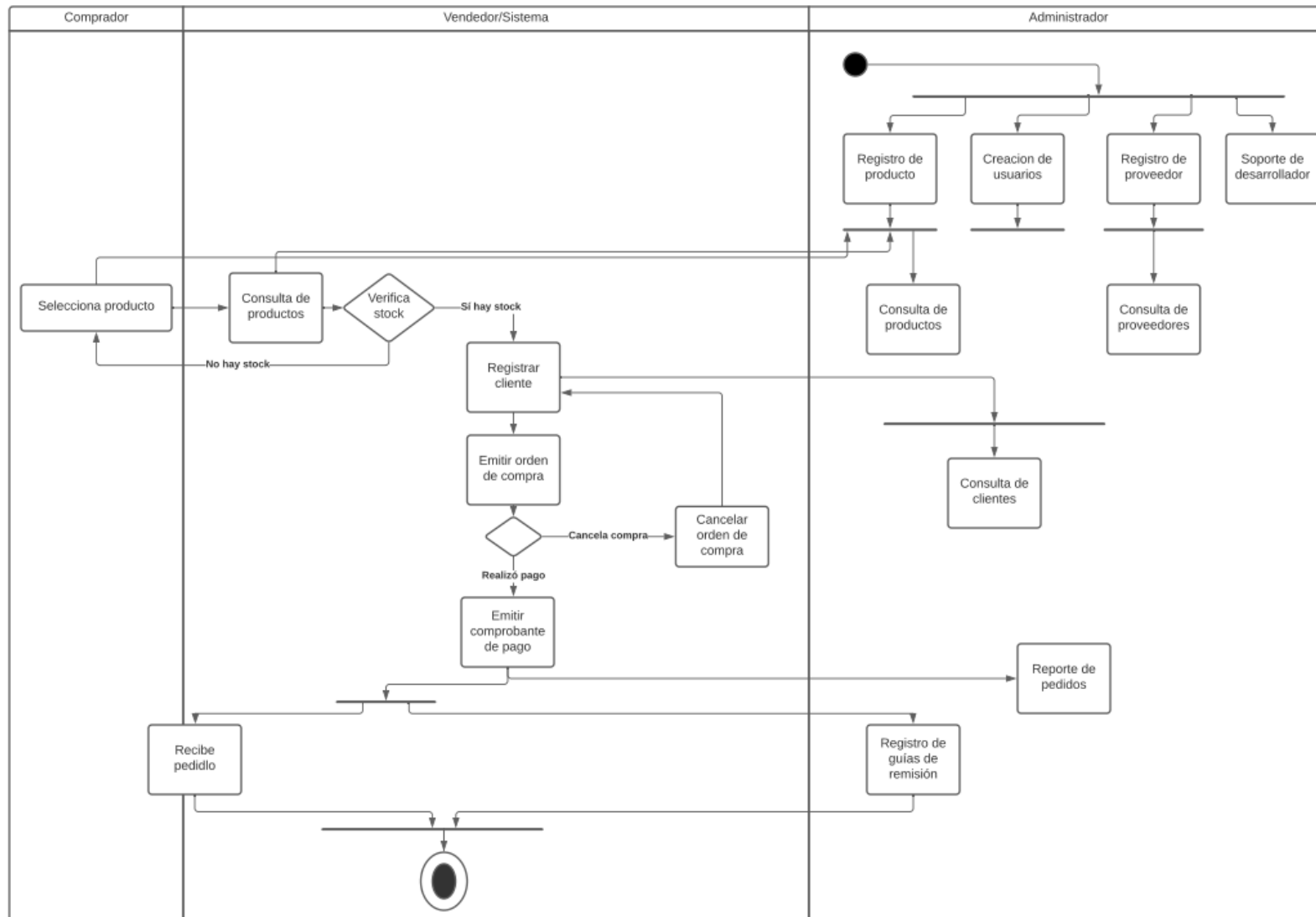


Ilustración 5 Diagrama de actividades

### 1.7. Prototipos de software.

El prototipo de la ventana de Login muestra un título "SISTEMA DE VENTAS 'Maderera Kiba'" en la parte superior. En el centro, el texto "LOGIN" está en mayúsculas. Debajo, hay dos campos de entrada: "Usuario :" y "Contraseña :". El campo de contraseña incluye un botón "Ver" con un icono de ojo. Abajo de los campos está un botón "Ingresar". En la esquina inferior derecha, hay una barra de navegación con un icono de retroceso y el texto "X".

*Ilustración 7 Ventana de Login*

El prototipo de la ventana de selección de usuario muestra un título "SISTEMA DE VENTAS 'Maderera Kiba'" en la parte superior. En el centro, el texto "BIENVENIDO" está en mayúsculas, seguido de un icono de sonrisa. Debajo, hay dos opciones de usuario: "Vendedor" con un icono de persona y "Administrador" con un icono de documento. En la esquina inferior derecha, hay una barra de navegación con un icono de retroceso y el texto "Menú".

*Ilustración 6 Ventana de selección de usuario*

SISTEMA DE VENTAS "Maderera Kiba"

Registros Reportes

Usuarios  
Proveedores  
Productos  
Pedidos

## Registro de Usuarios

Tipo de usuario : Administrador  
Vendedor

Numero de DNI

Contraseña :  ☐ Ver

Fecha de registro: 02 / 09 / 2020

[< Menú](#)

*Ilustración 9 Ventana de registro de usuarios*

SISTEMA DE VENTAS "Maderera Kiba"

Registros Reportes

Proveedores  
Usuarios  
Productos  
Pedidos

## Registro de Proveedores

Nombre/Razón Social:

DNI/RUC:

Dirección:

Fecha de registro: 02 / 09 / 2020

[< Menú](#)

*Ilustración 8 Ventana de registro de proveedores*

SISTEMA DE VENTAS "Maderera Kiba"

Registros Reportes

Productos  
Usuarios  
Proveedores  
Pedidos

### REGISTRO DE PRODUCTOS

C. Producto :

Nombre :

Procedencia :  ☐

Empresa de Transporte :  ☐

Proveedor :  ☐

Tipo de Madera :

Tipo de Troza :

Especie de Madera :

Cantidad :

Registrar

< Menú

Ilustración 11 Ventana de registro de productos

SISTEMA DE VENTAS "Maderera Kiba"

### REGISTRO DE PEDIDOS

Registros Reportes

Pedidos  
Usuarios  
Proveedores  
Productos

Q Código de producto

Stock :

Precio :

☒

DNI Cliente :  ☐  Eliminar

Cantidad :

Cliente	C. Producto	Producto	Cantidad	P. Unitario	Subtotal

Confirmar pedido Emitir comprobante

Total :

< Menú

Ilustración 10 Ventana de registro de pedidos

SISTEMA DE VENTAS "Maderera Kiba"

# REPORTE DE USARIOS

Reportes

Registros

Usuarios

Proveedores

Productos

Pedidos

Listar

Buscar

Eliminar

DNI Usuario	Contraseña	T. Usuario	F. Registro

Menú

Ilustración 13 Ventana de reporte de usuarios

SISTEMA DE VENTAS "Maderera Kiba"

# REPORTE DE PROVEEDORES

Reportes

Registros

Proveedores

Usuarios

Productos

Pedidos

Listar

Buscar

Eliminar

DNI/RUC	Nombre/RS	Dirección	F. Registro

Menú

Ilustración 12 Ventana de reporte de proveedores



[illegible]

*Ilustración 15 Ventana de reporte de productos*

[illegible]

*Ilustración 14 Ventana de reporte de pedidos*

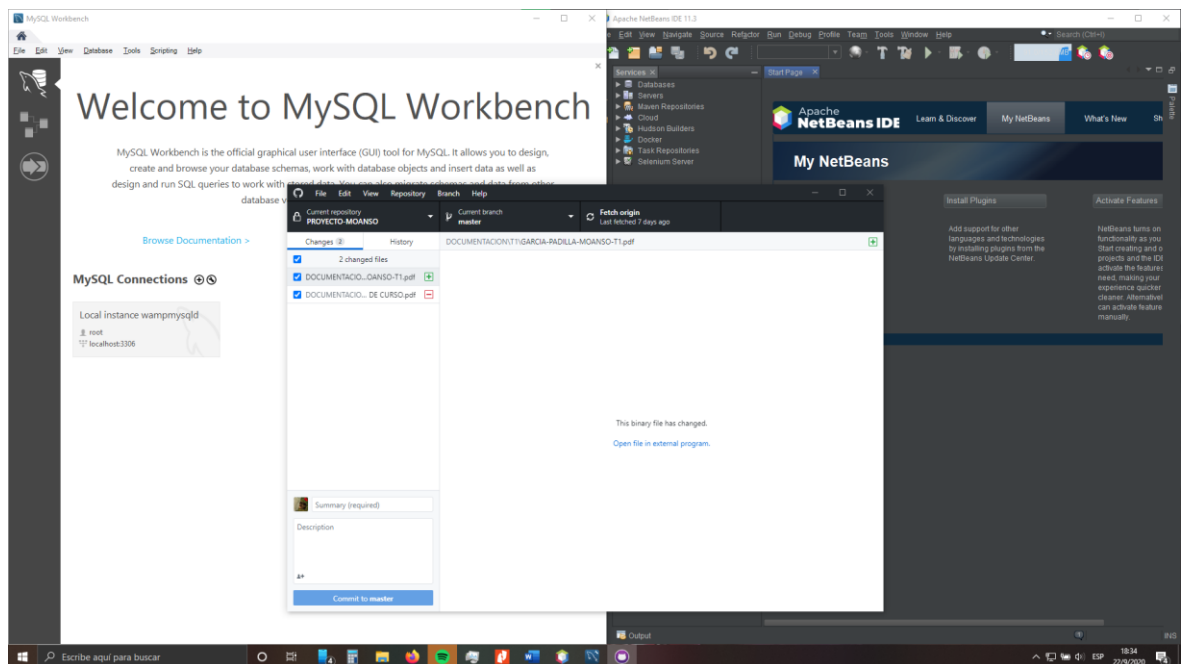
### 1.8. Lista de requerimientos funcionales y no funcionales.

*Tabla 3 Lista de requerimientos funcionales y no funcionales.*

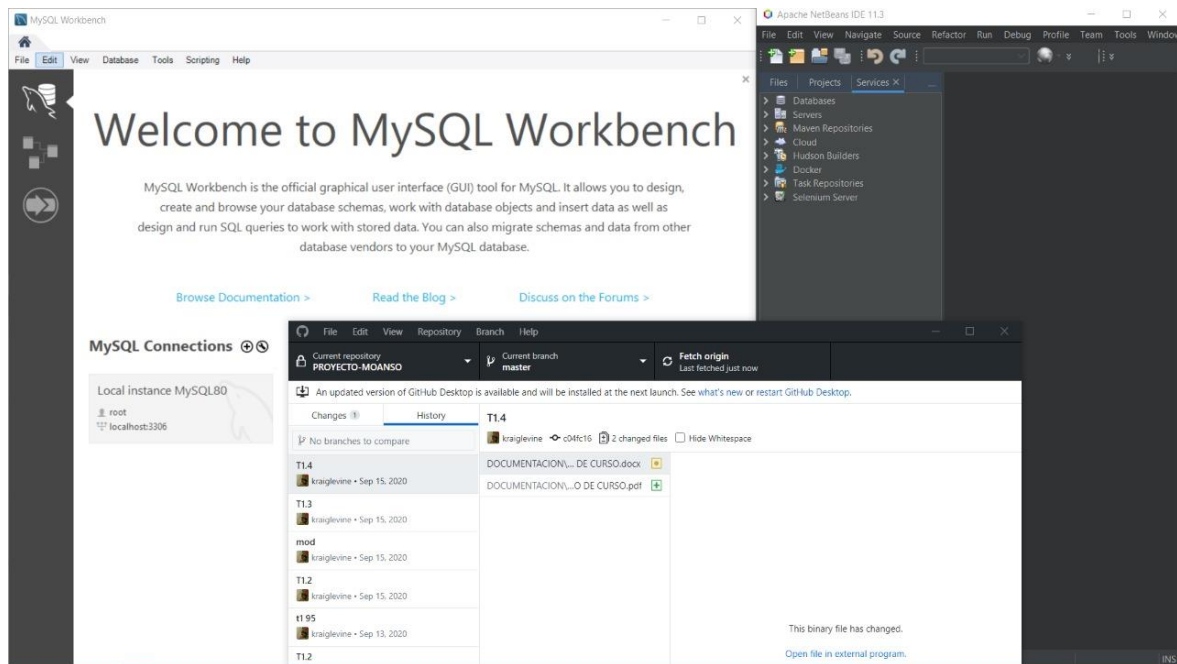
ID	Requisito	Caso(s) de uso	Prototipo	Versión	Instalación
1	Permite el registro de los administradores y vendedores	Creación de usuarios	Sí	1	No
2	Permite el registro de los productos que ingresan	Registro de producto	Sí	1	No
3	Permite el registro de los proveedores de los productos	Registro de proveedor	Sí	3	No
4	Permite la comunicación del usuario con el desarrollador	Soporte de desarrollador	No	3	No
5	Permite al usuario a buscar el producto solicitado	Selección de producto	Sí	2	No
6	Permite consultar la lista de productos registrados	Consulta de productos	Sí	1	No
7	Permite consultar la lista de los proveedores registrados	Consulta de proveedores	Sí	1	No
8	Permite consultar el stock de los productos registrados	Verificación de stock	Sí	1	No
9	Permite el registro de los clientes que realicen una compra	Registro de cliente	Sí	2	No
10	Permite consultar la lista de clientes registrados	Consulta de clientes	No	3	No
11	Permite el registro de las compras realizadas por el cliente	Emisión de orden de compra	Sí	3	No
12	Permite al cliente tener una constancia de la compra realizada	Emisión de comprobante de pago	Sí	3	No
13	Permite al cliente sustentar el traslado de su compra	Registro de guías de remisión	No	3	No

14	Permite consultar la lista de órdenes de compra registradas	Reporte de pedidos	Sí	3	No
15	Lenguaje de programación	Java	No	1	No
16	Framework de desarrollo	Netbeans	No	1	Sí
17	Base de datos	Mysql Workbench	No	1	Sí
18	Repositorio	GitHub Desktop	No	1	Sí

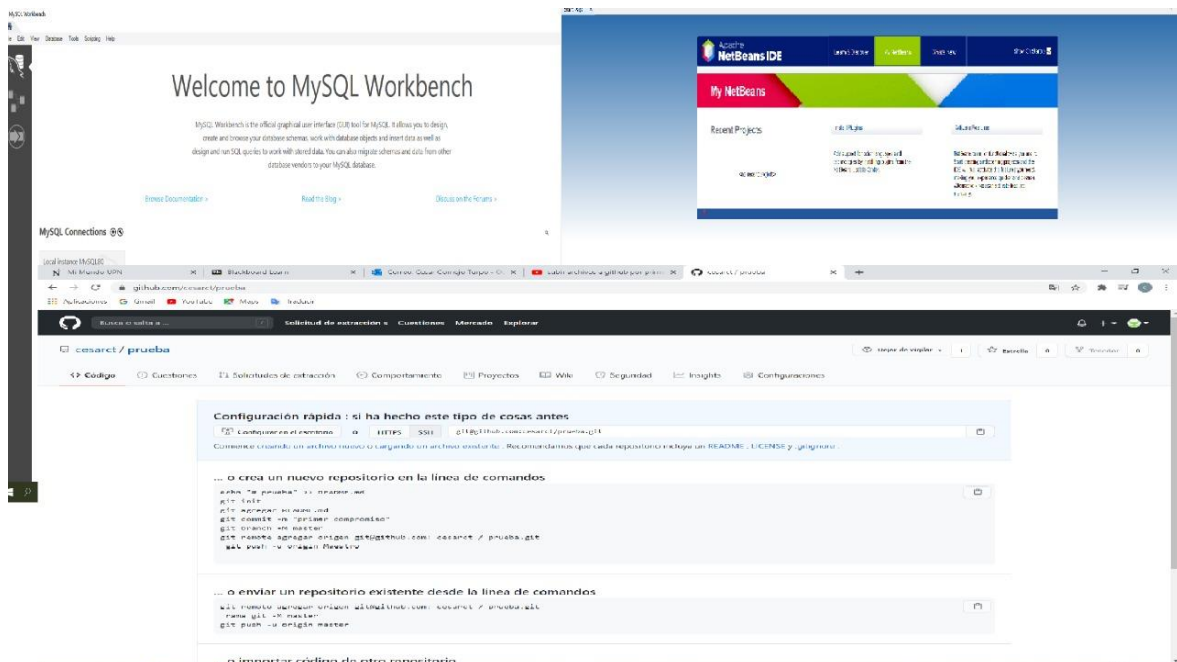
### 1.9. Requerimientos no funcionales instalados.



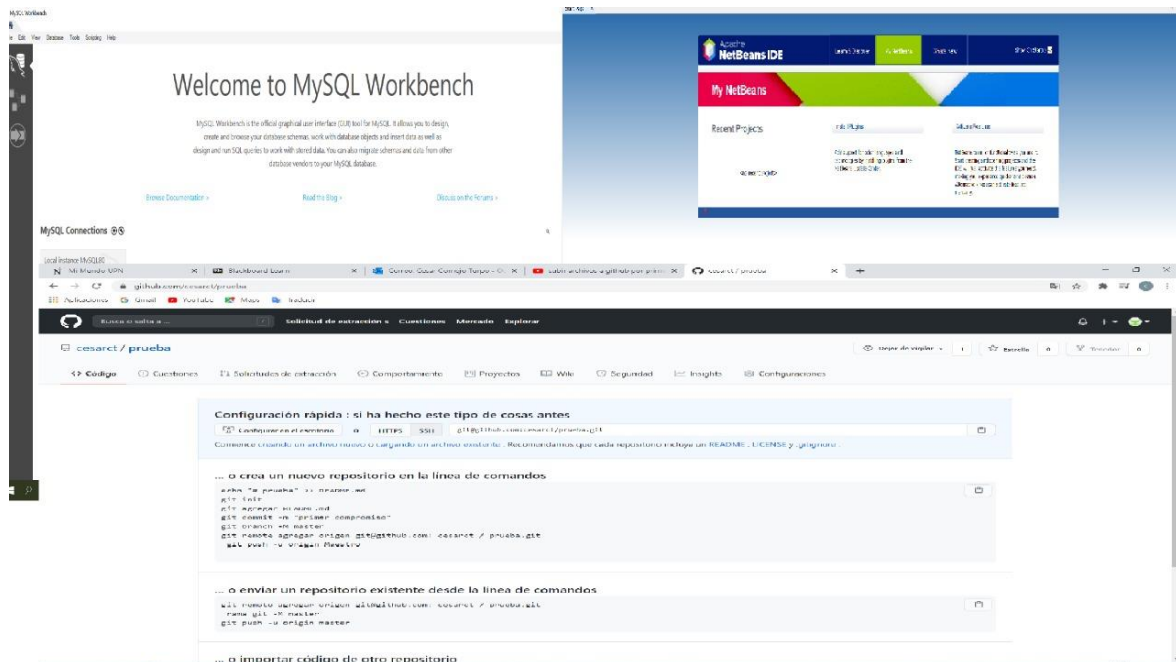
*Ilustración 16 Entorno de desarrollo de Kenko Garcia*



*Ilustración 18 Entorno de desarrollo de Jean Carlos Chirinos*



*Ilustración 17 Entorno de desarrollo de Cesar Cornejo*



## CAPITULO 4.

### 1. Conclusiones y recomendaciones.

#### 1.1. Conclusiones.

- La primera versión del software demuestra un correcto funcionamiento.
- El marco teórico se fundamenta en la utilización de IDE's como NetBeans y Workbench.
- El sistema facilita el trabajo eficaz del usuario y agiliza el proceso de registro de productos en comparación a los métodos escritos.

#### 1.2. Recomendaciones.

- Mejorar el diseño visual del interfaz del prototipo.
- Utilizar criterios de calidad en el desarrollo del software.
- Agregar mejoras y actualizaciones en caso el entorno lo requiera.

### 2. Referencias bibliográficas.

- Bruegge, B., & Dutoit, A. H. (2002). *Ingeniería de software orientado a objetos*. Mexico: Pearson Education.
- Fowler, M. (1999). *UML gota a gota*. Mexico: Pearson Education.
- Gimeno, J. M., & Gonzalez, J. L. (2011). *Introducción a Netbeans*. Lérida: Universitat de Lleida, Enginyeria Informàtica.
- GitHub Inc. (15 de Septiembre de 2020). *GitHub*. Obtenido de <https://github.com/about>
- Oracle. (2020). *¿Qué es Java?* Obtenido de [https://www.java.com/es/about/whatis\\_java.jsp](https://www.java.com/es/about/whatis_java.jsp)
- Schildt, H. (2009). *Java*. Mexico: McGraw-Hill Interamericana.
- Sommerville, I. (2005). *Ingeniería de software*. Pearson Education.

### 3. Anexos.

#### 3.1. Acta de aceptación del proyecto.



Kenko Garcia <kenkogarcia@gmail.com>

---

#### CARTA DE PRESENTACION

2 mensajes

Kenko Garcia <kenkogarcia@gmail.com>  
Para: madererakiba@hotmail.com

15 de septiembre de 2020, 19:22

Buenas noches estimado señor Garcia.

De acuerdo a lo conversado por teléfono, le envío la carta de presentación formal por parte de la UPN, así mismo le solicito autorice la inclusión del equipo de trabajo que conforma el proyecto, sus nombres detallados a continuación.

#### INTEGRANTES:

Camila Tamara Cruz Diaz

Jean Carlo Chirinos Avendaño

Cesar Cornejo Turpo

Quedo atento a su respuesta,

Saludos cordiales,

Kenko Garcia.

---

 Presentation-62896.pdf  
6K

---

Jorge Garcia Sandoval <madererakiba@hotmail.com>  
Para: Kenko Garcia <kenkogarcia@gmail.com>

15 de septiembre de 2020, 21:58

Correcto Kenko.

Brindare las facilidades necesarias para el desarrollo de su proyecto.

Gracias.

Atentamente, Jorge Garcia.

---

De: Kenko Garcia <kenkogarcia@gmail.com>  
Enviado: martes, 15 de setiembre de 2020 19:22  
Para: madererakiba@hotmail.com <madererakiba@hotmail.com>  
Asunto: CARTA DE PRESENTACION

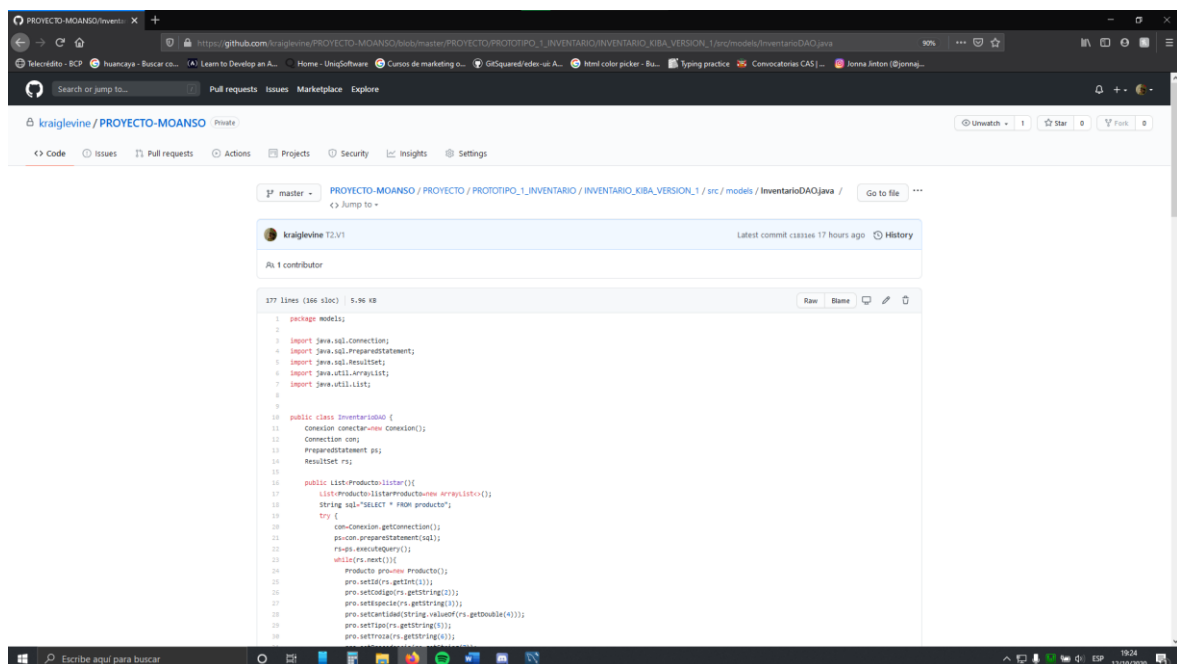
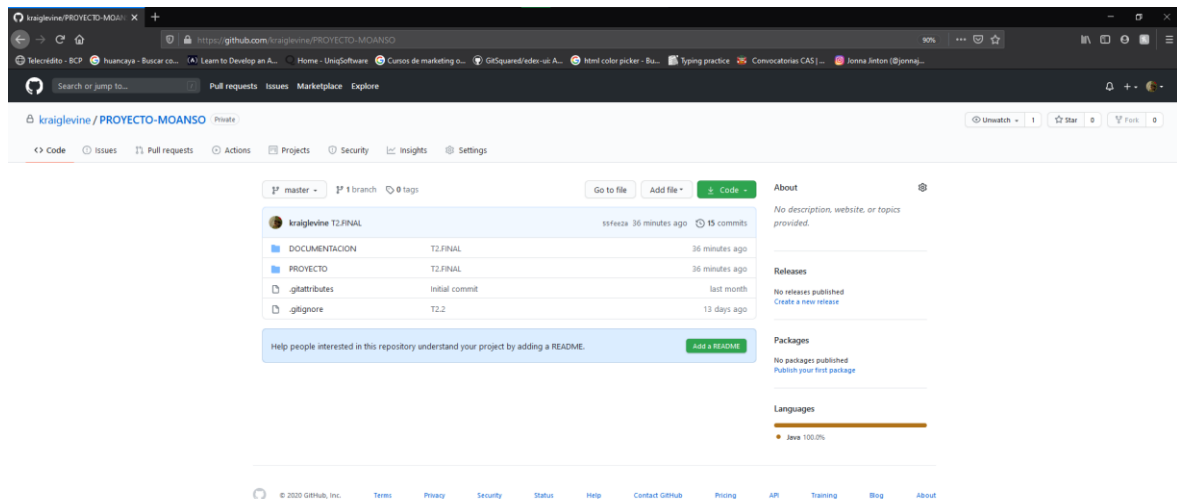
[El texto citado está oculto]

### 3.2. Foto grupal.





### 3.3. Repositorio en GitHub.



PROYECTO-MOANSO

https://github.com/kraiglevine/PROYECTO-MOANSO/blob/master/PROYECTO-BASE DE DATOS/MYSQL/inventario\_kiba\_usuario.sql

Go to file

master - PROYECTO-MOANSO / PROYECTO / BASE DE DATOS / MYSQL / inventario\_kiba\_usuario.sql

Go to file

kraiglevine T2.1

Latest commit 13 days ago

History

1 contributor

57 lines (56 lines) 2.1 KB

Raw Blame

```
1 -- MySQL dump 10.13 Distrib 5.0.28, for Win64 (x86_64)
2 --
3 -- host: localhost    database: inventario_kiba
4 --
5 -- Server version      5.0.28
6 --
7 /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
8 /*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
9 /*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
10 /*!40101 SET NAMES utf8 */;
11 /*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
12 /*!40103 SET TIME_ZONE='+00:00' */;
13 /*!40101 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
14 /*!40101 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
15 /*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
16 /*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;
17
18 --
19 -- Table structure for table 'usuario'
20 --
21
22 DROP TABLE IF EXISTS `usuario`;
23 /*!40101 SET @CHARACTER_SET_CLIENT = @@character_set_client */;
24 /*!40101 SET @COLLATION_CONNECTION = @@collation_connection */;
25 CREATE TABLE `usuario` (
26   `usr_id` int(11) NOT NULL,
27   `usr_nombres` varchar(45) NOT NULL,
28   `usr_apellidos` varchar(45) NOT NULL,
29   `usr_email` varchar(45) NOT NULL,
30   `usr_registro` date NOT NULL,
31   `usr_tipo` varchar(10) NOT NULL,
32   `inv_id` int(11) NOT NULL,
33   PRIMARY KEY (`usr_id`),
34   KEY `fk_usuario_inventario` (`inv_id`),
35   CONSTRAINT `fk_usuario_inventario` FOREIGN KEY (`inv_id`) REFERENCES `inventario` (`inv_id`)
36 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
37
38 -- Dumping data for table 'usuario':
39 --
40 --
41 --
42 --
43 --
44 --
45 --
46 --
47 --
48 --
49 --
50 --
51 --
52 --
53 --
54 --
55 --
56 --
57 --
```

### 3.4. Script de la versión 1 del software.

```

package controllers;

import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.List;
import javax.swing.JOptionPane;
import javax.swing.JTable;
import javax.swing.table.DefaultTableModel;
import modelo.InventarioDAO;
import modelo.Producto;
import vista.InventarioPrincipal;
import vista.Menu_Admin;

public class InventarioController implements ActionListener {
    DefaultTableModel modeloTabla=new DefaultTableModel();
    InventarioPrincipal vista=new InventarioPrincipal();
    InventarioDAO dao=new InventarioDAO();

    public InventarioController(InventarioPrincipal vista){
        this.vista=vista;
        this.vista.btnListar.addActionListener(this);
        this.vista.btnAgregar.addActionListener(this);
        this.vista.btnEliminar.addActionListener(this);
        this.vista.btnActualizar.addActionListener(this);
        this.vista.btnSalir.addActionListener(this);
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        if(e.getSource()==vista.btnListar){
            limpiarTabla();
            listarInventario(vista.tblInventario);
        }
        if(e.getSource()==vista.btnAgregar){
            agregarProducto();
            limpiarTabla();
            listarInventario(vista.tblInventario);
        }
        if(e.getSource()==vista.btnEliminar){
            vista.btnAgregar.setEnabled(false);
            vista.btnActualizar.setEnabled(false);
            vista.btnEliminar.setEnabled(true);
            int fila=vista.tblInventario.getSelectedRow();
            if(fila!=-1){
                JOptionPane.showMessageDialog(vista, "Debe seleccionar una fila...");
            }
        }
        if(e.getSource()==vista.btnActualizar){
            String nombre=(String)vista.tblInventario.getValueAt(fila, 0).toString();
            String nombre2=(String)vista.tblInventario.getValueAt(fila, 1);
            String apellido=(String)vista.tblInventario.getValueAt(fila, 2);
            String dni=(String)vista.tblInventario.getValueAt(fila, 3);
            String contrasena=(String)vista.tblInventario.getValueAt(fila, 4);
            String tipo=(String)vista.tblInventario.getValueAt(fila, 5);
        }
    }
}

```

```

-- Table structure for table `usuario`
--
DROP TABLE IF EXISTS `usuario`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8mb4 */;
CREATE TABLE `usuario` (
  `usr_id` int(11) NOT NULL AUTO_INCREMENT,
  `usr_nombre` varchar(45) NOT NULL,
  `usr_apellido` varchar(45) NOT NULL,
  `usr_dni` varchar(9) NOT NULL,
  `usr_contrasena` varchar(15) NOT NULL,
  `usr_tipo` varchar(30) NOT NULL,
  PRIMARY KEY (`usr_id`)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @@saved_cs_client */;

-- Dumping data for table `usuario`
--
LOCK TABLES `usuario` WRITE;
/*!40101 ALTER TABLE `usuario` DISABLE KEYS */;
INSERT INTO `usuario` VALUES (1, 'kenko', 'Garcia', '123', '123', 'Administrador'), (2, 'kenko', 'Garcia', '456', '456', 'Vendedor');
/*!40101 ALTER TABLE `usuario` ENABLE KEYS */;
UNLOCK TABLES;
/*!40101 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40101 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40101 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;

```

### 3.5. Secuencia de funcionamiento de la versión 1.

