



UNIVERSIDAD  
PRIVADA  
DEL NORTE

# **FACULTAD DE INGENIERÍA**

CARRERA DE INGENIERÍA DE SISTEMAS COMPUTACIONALES

- **TÍTULO :** Sistema de Gestión de  
Ventas de comida del restaurante “Mechita”
  
- **AUTOR(ES):**
  - Amable Ricse, Fher
  - Velásquez Navarro, Raúl
  - Sánchez Rodríguez, Javier
  
- **CURSO :** Modelamiento y Análisis de Software
  
- **DOCENTE :** Jorge Alfredo , Guevara Jimenez

## **IDENTIFICACIÓN DEL PROBLEMA:**

Hoy en día, muchas empresas de trabajo como farmacias, hospitales, bodegas y/o restaurantes, ven la necesidad de implementar un sistema para así poder manejar mejor su negocio, atender satisfactoriamente a los clientes y poder llevar una mejor organización de la empresa asegurando un futuro viable.

Actualmente, la cevichera “mechita” ubicado en el callao, no cuenta con un sistema de pedidos que permita agilizar mejor sus ventas a sus clientes, ya que el personal solo cuenta con un mesero y al momento de que los clientes llaman para poder reservar una mesa, esté ocupado atendiendo a los clientes del restaurante y no pueda contestar la reservación, por tal motivo se vio la necesidad de crear un software donde se sistematizará el control de pedidos para que el cliente pueda reservar su sitio, pedir los diferentes tipos de platos que sirven en el restaurante y así poder estar satisfecho con el servicio.

Para cumplir con nuestro objetivo de entregarle al restaurante este software funcionando, haremos un análisis y diseño del sistema; utilizando la información pertinente para así desarrollar un eficaz y eficiente trabajo.

## **PLANTEAMIENTO DE SOLUCIÓN:**

Nuestro proyecto contará con un sistema, en el cual el cliente pedirá el tipo de plato que desee y reservará un sitio para poder ser atendido correctamente. Los datos ingresados, serán enviados directamente a la base de datos para ser visualizados, luego por el personal del restaurante, así mismo el cliente podrá cancelar o editar su pedido en un tiempo determinado, dándole un mejor uso y orden a las acciones que tome.

A continuación se mencionará dichas acciones que tendrá nuestro proyecto:

- Facilitar la consulta simple de los diferentes platos y precios que dispone el local.
- Permitir la elección de platos y lugar de mesa para su respectiva atención.
- Facilitar al personal atender los pedidos por medio de este sistema de una manera más eficiente y ordenada.
- Reducir los errores en la captura de información de los procedimientos de venta y servicio.

### **JUSTIFICACIÓN:**

La presente investigación se enfocará en generar una mejora y desarrollo en los procesos que se produce en un restaurante, así para poder alcanzar los objetivos propuestos; ergo es fundamental implementar un sistema informático de ventas y servicios que ofrezcan efectividad y eficiencias al lugar.

La implementación de este sistema de gestión de información de ventas y servicios planteada en el restaurante, ofrecerá la oportunidad de obtener ventajas considerables, mejorar y aumentar la capacidad de organización del restaurante.

### **LIMITACIONES:**

- Editar el ingreso de datos, se tendrá que volver a cargar manualmente de nuevo.
- Incapacidad de atender óptimamente a los clientes, la cantidad y uso del tiempo son limitantes.
- Poco acceso a la información ya que no manejan un sistema.
- Contabilidad y administración sencilla.

## **OBJETIVO GENERAL**

- Diseñar un Sistema de Gestión de información de las ventas y servicios, que permita mejorar u optimizar los procedimientos y manejo de la base de datos del Restaurante.

## **OBJETIVOS ESPECIFICOS**

- Determinar los requerimientos administrativos, técnicos y operativos para la construcción del Sistema de Gestión de la Información de Venta y Servicio.
- Recolectar y digitalizar la información para ser incluida en el Sistema de Gestión de Información de Venta y Servicio.
- Identificar y solucionar los problemas existentes mediante el Sistema de Gestión de la Información de Venta y Servicio.
- Desarrollar el Sistema de Gestión de Información de Venta y Servicio en una plataforma u entorno, apoyado en una base de datos.
- Recolectar la información de los clientes para incluirla en el Sistema de Gestión de Información.
- Analizar la solicitud y necesidades del usuario para el diseño del Sistema de Gestión de la Información.
- Verificar y poner en funcionamiento el Sistema de Gestión de la Información.

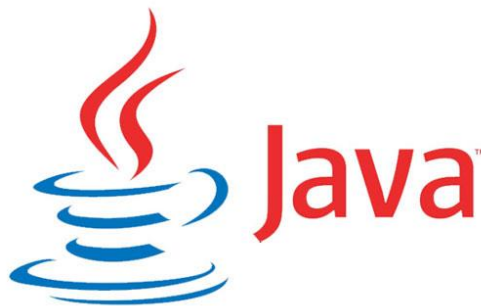
## MARCO TEÓRICO

El proyecto a elaborar requiere de un software donde por medio de un motor de base de datos y Lenguaje Java utilizando una IDE llamada **Netbeans** en su versión 8.2, lo cual pueda disminuir los problemas que tiene el restaurante y así poder ofrecer un mejor servicio a todo el personal, disminuyendo tiempo en consultas, búsquedas de datos de los clientes, donde el usuario pueda quedar satisfecho con la atención recibida en el Restaurante, agilizando la atención al cliente; este producto se entregará en un CD con el aplicativo con su respectivo manual de usuario. A continuación damos una descripción de cada uno de los instrumentos que se utilizaron para el Diseño.

1. **Netbeans**: es un entorno de desarrollo libre para lenguajes de programación como Java , c++, php, y otros ,está desarrollado en java y por eso requiere la instalación del JDK y la JVM para su funcionamiento, lo que permite este tipo de software es proporcionarle al programador herramientas para que el desarrollo de software sea lo menos complicado , además con esta IDE puedes diseñar interfaces de una manera mucho más fácil con solo arrastrar botones y demás a tu JFrame e incluso puedes incluirle plugin que mejoran la funcionalidad de la IDE.



2. **Lenguaje Java:** es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. (Wikipedia, 2018) Una de las ventajas significativas de Java sobre otros lenguajes de programación es que es independiente de la plataforma, tanto en código fuente como en binario. Esto quiere decir que el código producido por el compilador Java puede transportarse a cualquier plataforma (Intel, Motorola, Sparc, etc.) que tenga instalada una máquina virtual Java y ejecutarse.



3. **Bases de Datos:** Una base de datos es un archivo que respalda información con características comunes, la base de datos agrupa una gran cantidad de información para su posterior análisis y procesamiento, la utilidad de las bases de datos informatizadas radica en la simplificación de cálculos y la instantaneidad con que se pueden procesar, a diferencia de una base de datos manual, sería si no imposible bastante complicado procesar.



4. **DIAGRAMA UML:** El Lenguaje Unificado de Modelado (UML) es un lenguaje de modelado visual de propósito general que se utiliza para especificar, visualizar, construir y documentar los artefactos de un sistema software. (Wikipedia, 2018) Captura decisiones y conocimiento sobre sistemas que deben ser construidos. Se usa para comprender, diseñar, ojear, configurar, mantener y controlar la información sobre tales sistemas. Está pensado para ser utilizado con todos los métodos de desarrollo, etapas del ciclo de vida, dominios de aplicación y medios.

### **MODELO DE PROCESOS DE NEGOCIO**

**Roles:** Especifica los roles que participarán en el sistema.

**Evento:** Suceso que influye en el comportamiento de sistema.

**Proceso de negocio:** Es un conjugado de actividades diseñadas para producir una salida para un determinado cliente o mercado. Enfatiza cómo se realiza el trabajo.

**Información:** Los procesos de negocio utilizan información para completar sus actividades, éstas no se consumen en el proceso, pueden provenir de un proceso anterior.

**Recurso:** Es una entrada al proceso de negocio que es consumida durante el procesamiento, es decir solamente se utiliza una sola vez. Por ejemplo: materiales, productos, personas, información y servicios.



**Meta:** Es la razón por la cual la organización está realizando la actividad y deben ser definidas a las necesidades del negocio.

**Salida:** Una salida puede ser física como un informe o factura, una transformación de recursos en uno nuevo, como una agenda diaria y puede ser un resultado general de la empresa.

### **VENTAJAS DE LA SOLUCIÓN:**

- Mayor rapidez y uso en información.
- Mayor rapidez al momento de registrar los pedidos.
- Poder llevar un registro de las ventas realizadas.
- Mejora la presentación al usar tecnología sofisticada.
- Rapidez y eficiencia en el tiempo de atención.
- Mejora en la administración del lugar.
- Facilidad de registro de productos.

### **DESVENTAJAS DE LA SOLUCIÓN:**

- No aprovechar los recursos que ofrece el sistema.
- Hackers.
- Fallos del sistema.
- Mantenimiento al software constante.
- Capacitación al personal entrante.
- Crackers.
- Poca confianza para el uso del sistema establecido.

## MARCO METODOLÓGICO

Este proyecto se enmarca dentro de la Investigación para luego implementar la metodología de crear un sistema que comprende las fases de Análisis, Planificación, Contenido, Diseño, Programación Testeo.

- **Fase de análisis:** A la hora del desarrollo de una plataforma de escritorio se debe comenzar por dar respuesta a un grupo de interrogantes, que permitirán aclarar y definir de manera básica las expectativas existentes en torno al sistema. Todo proyecto debe contar con objetivos definidos, con los cuales puede medirse la efectividad de las acciones que se desarrollen.
- **Fase de planificación:** En ella se deben definir cuáles son los requerimientos técnicos para este fin, quiénes serán los miembros del equipo, seleccionar la estructura, planear un sitio bien organizado.
- **Fase de contenido:** El contenido de sistema, es toda la información que actualmente se posee y conoce, por lo cual se tendrá que crear desde ello. Es conveniente presentar un contenido acorde a los objetivos planteados para desarrollar el sistema, y que la información a mostrar cubra las necesidades e intereses del público.
- **Fase de diseño:** El diseño del sistema supone un trabajo extenso y detallado, puesto que abarca no sólo la interacción de múltiples elementos, como tecnología multimedia; sino, que abarca también su integración con una estructura lógica basada en el propósito del sistema. Representando una labor que va más allá de la simple necesidad e inquietudes de sus potenciales usuarios.

▪ **Fase de programación:**

Esta etapa corresponde a la utilización de las herramientas de programación Java. En este punto es primordial seleccionar los lenguajes de programación en los que se desarrollará el sistema.

• **Requisitos Funcionales**

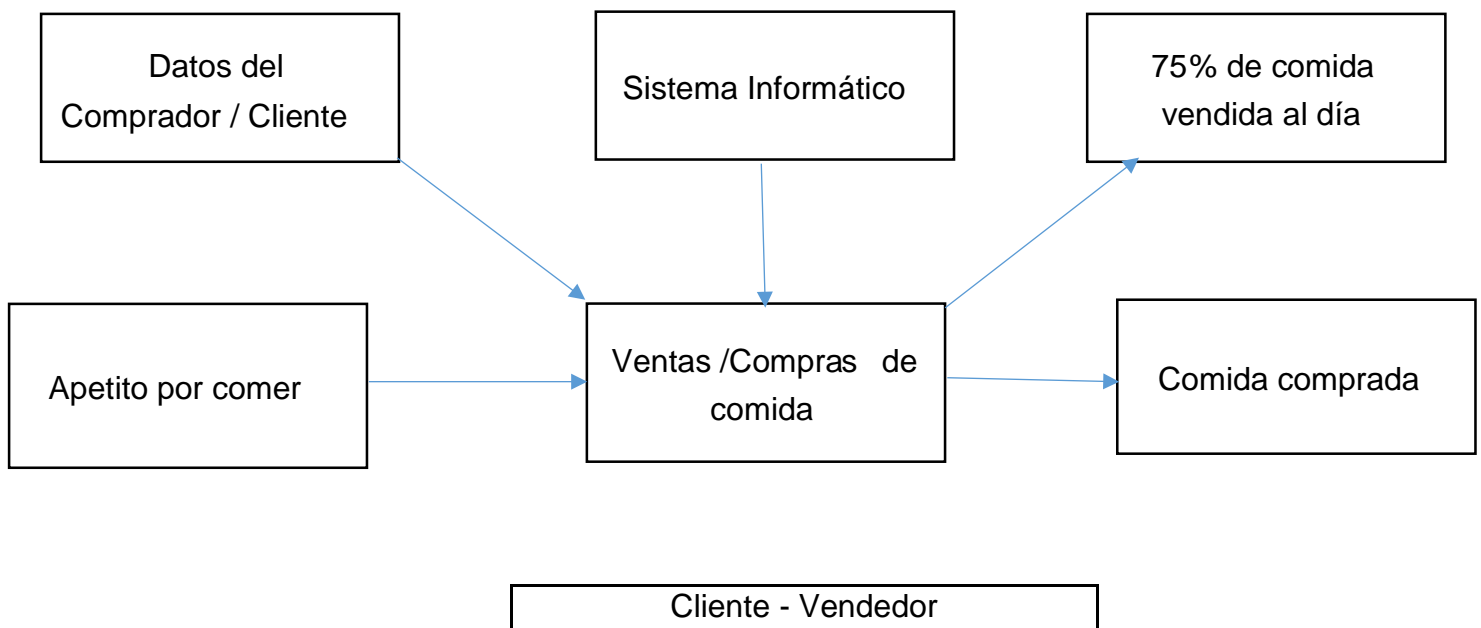
1. El sistema permite registrar detalles del pedido
2. El sistema permite registrar venta
3. El sistema permite emitir comprobante de pago por la venta
4. El sistema permite registrar datos del cliente
5. El sistema permite registrar la atención para el cliente
6. El sistema permite calcular las ventas del día y ganancias

• **Requisitos No Funcionales**

1. El sistema utiliza el lenguaje Java
2. El sistema utiliza la base de datos SQL Server
3. El sistema es para una plataforma de escritorio
4. El sistema implementará el paradigma POO

## MAPA DE PROCESOS

### 1. Sistema de Gestión de Ventas de comida



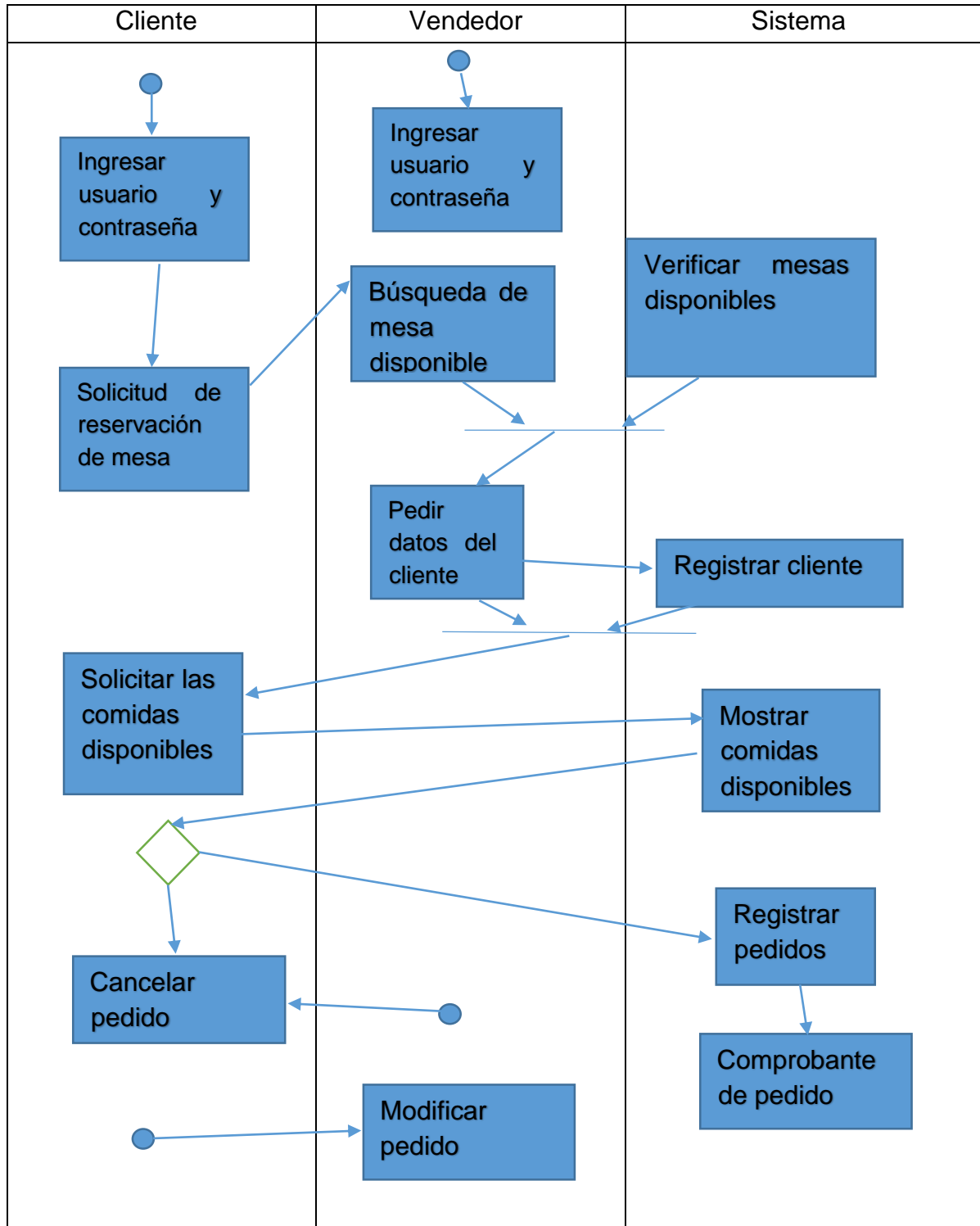
## ESPECIFICACION DE PROCESOS

Proceso	Entradas	Actividades	Salidas	Valor del cliente
Compra de comida	<ul style="list-style-type: none"> <li>☐ Reservación de mesa</li> <li>☐ Nombre de Cliente</li> <li>☐ N° Comida: - Ceviche -Sudado -Chicharrón</li> <li>☐ N° porción -Arroz -Chifle</li> <li>☐ Solicitud de compra</li> </ul>	<ul style="list-style-type: none"> <li>☐ Registrar cliente</li> <li>☐ Registrar Comida</li> <li>☐ Confirmación de solicitud</li> <li>☐ Pago de compra</li> </ul>	☐ Comida comprada	☐ Asegurar la satisfacción del consumo de la comida

## MODELO DE CASOS DE USOS



## DIAGRAMA DE ACTIVIDADES



## Lista Requisitos

Requisito funcional	Requisito no funcional	Requisito facilidad de uso
<ul style="list-style-type: none"> <li>• Ingresar usuario y contraseña</li> <li>• Registrar cliente</li> <li>• Registro de pedido</li> <li>• Reservación de mesa</li> <li>• Emitir Comprobante de pago</li> </ul>	<ul style="list-style-type: none"> <li>• El sistema utiliza el lenguaje Java</li> <li>• El sistema utiliza la base de datos SQL Server</li> <li>• El sistema es para una plataforma de escritorio</li> <li>• El sistema implementará el paradigma POO</li> </ul>	<ul style="list-style-type: none"> <li>• Interfaz gráfica intuitiva y fácil de usar</li> <li>• Realizar el programa en 4 pasos</li> <li>• Facilidad de registro de productos y pedidos</li> </ul>

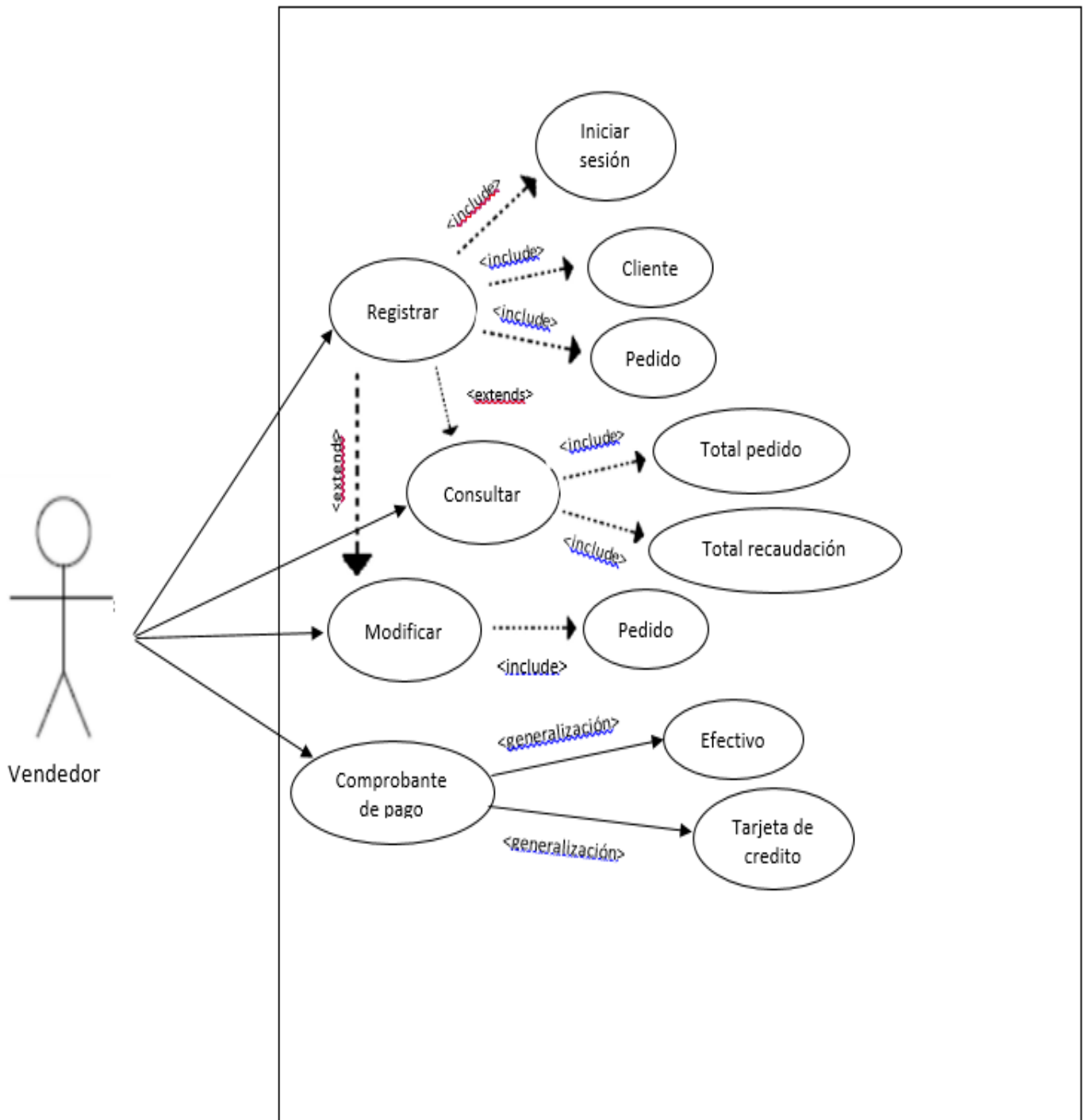


### Matriz de trazabilidad de requisitos

Enunciado de requisito	Caso de uso	Prototipo o de software	Criterio de aceptación	Estado	Responsable	Fecha INI	Fecha FIN
El sistema permite registrar cliente	Registrar Cliente	Prototipo registrar pedido	<ul style="list-style-type: none"> <li>Ingreso de datos del cliente</li> </ul>	Iniciado	RVN	24/04	27/04
El sistema permite registrar pedido	Registrar pedido		<ul style="list-style-type: none"> <li>Ingreso de datos de los pedidos</li> </ul>	Pendiente	FAR		
El sistema utiliza la base de datos SQL Server		SQL Server	<ul style="list-style-type: none"> <li>Dar registrar</li> </ul>	Pendiente	JSR	24/04	27/04
El sistema utiliza Neatbeans			<ul style="list-style-type: none"> <li>Verificar pedido</li> </ul>	Pendiente	JSR	24/04	27/04
			<ul style="list-style-type: none"> <li>Cancelar pedido</li> </ul>	Iniciado	JSR	24/04	27/04
			Instalar SQL Verificar conexión	Pendiente	FAR	24/04	27/04
		Netbeans (SDR)		Iniciado		24/04	27/04

**CODIGO FUENTE:**

## Casos de usos relacionados:



## Especificación de casos de usos:

Actores	Vendedor.
Descripción	Permite registrar los pedidos.
Disparador	Necesidad de registrar un nuevo pedido.
Pre-condiciones	Usuario registrado como vendedor.
Post-condiciones	Pedido registrado.
Flujo Normal	<ul style="list-style-type: none"> <li>Se valida su usuario (vendedor)</li> <li>El usuario ingresa al menú de pedidos</li> <li>El usuario ingresa los datos del pedido.               <ol style="list-style-type: none"> <li>Fuente:                   <ol style="list-style-type: none"> <li>Cantidad:</li> <li>Precio:</li> </ol> </li> <li>Bebida:                   <ol style="list-style-type: none"> <li>Cantidad:</li> <li>Precio:</li> </ol> </li> <li>Porción:                   <ol style="list-style-type: none"> <li>Cantidad:</li> <li>Precio:</li> </ol> </li> </ol> </li> <li>El usuario completa los campos.</li> <li>El sistema confirma los datos ingresados.</li> <li>El sistema guarda los datos en la base de datos.</li> <li>El sistema muestra los datos guardados en una tabla de pedidos.</li> <li>Finaliza el caso de uso.</li> </ul>
Flujo Alternativo	<ul style="list-style-type: none"> <li>El usuario cancela sin terminar.               <ul style="list-style-type: none"> <li>El sistema pregunta al usuario si desea abandonar.</li> </ul> </li> <li>Finaliza el caso de uso.</li> </ul>
Prioridad	Alto.

Actores	Vendedor
Descripción	Permite registrar los clientes
Disparador	Necesidad de registrar un nuevo cliente
Pre-condiciones	Usuario registrado como vendedor
Post-condiciones	Cliente registrado
Flujo Normal	<ul style="list-style-type: none"> <li>• Se valida su usuario (vendedor)</li> <li>• El usuario ingresa al menú de cliente.</li> <li>• El usuario ingresa los datos del cliente.               <ol style="list-style-type: none"> <li>1. Nombre:</li> <li>2. Apellido Paterno:</li> <li>3. Apellido Materno:</li> <li>4. DNI:</li> <li>5. e-mail:</li> <li>6. Teléfono:</li> </ol> </li> <li>• El usuario completa los campos.</li> <li>• El sistema confirma los datos ingresados.</li> <li>• El sistema guarda los datos en la base de datos.</li> <li>• El sistema muestra los datos guardados en una tabla de clientes.</li> <li>• Finaliza el caso de uso.</li> </ul>
Flujo Alternativo	<ul style="list-style-type: none"> <li>• El usuario no completa los campos.               <ul style="list-style-type: none"> <li>▪ El sistema muestra un mensaje de error.</li> </ul> </li> <li>• Finaliza el caso de uso.</li> </ul>
Prioridad	Alto

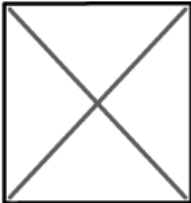
Actores	Vendedor.
Descripción	Permite consultar total de pedido y recaudación.
Disparador	Necesidad de consultar el total de pedido y recaudación.
Pre-condiciones	Usuario registrado como vendedor.
Post-condiciones	Se obtiene los datos que han sido registrados en la base de datos
Flujo Normal	<ul style="list-style-type: none"> <li>• Se valida el usuario (vendedor)</li> <li>• El usuario ingresa al menú de total de pedido y recaudación.</li> <li>• El sistema muestra los datos solicitados.</li> <li>• Finaliza el caso de uso.</li> </ul>
Flujo Alternativo	<ul style="list-style-type: none"> <li>• El sistema no muestra información de los datos guardados previamente.</li> <li>• Finaliza el caso de uso.</li> </ul>
Prioridad	Alto.

Actores	Vendedor
Descripción	Permite modificar los pedidos.
Disparador	Necesidad de modificar los pedidos.
Precondiciones	Campos llenados completamente.
Postcondiciones	Pedidos modificados.
Flujo Normal	<ul style="list-style-type: none"> <li>• Se valida su usuario (vendedor)</li> <li>• El usuario ingresa al menú de pedido.</li> <li>• El usuario modifica los datos ingresados previamente.</li> <li>• El sistema confirma los datos ingresados.</li> <li>• El sistema guarda los datos en la base de datos.</li> <li>• El sistema muestra los datos guardados en una tabla de clientes.</li> <li>• Finaliza el caso de uso.</li> </ul>
Flujo Alternativo	<ul style="list-style-type: none"> <li>• Al seleccionar modificar, no aparece ningún dato que se ingresó.</li> <li>• Finaliza el caso de uso.</li> </ul>
Prioridad	Medio

Actores	Vendedor
Descripción	Permite elegir el tipo de pago del pedido.
Disparador	Necesidad de emitir un tipo de comprobante de pago.
Precondiciones	Usuario registrado como vendedor.
Postcondiciones	Se obtiene los datos del pedido para su respectivo pago.
Flujo Normal	<ul style="list-style-type: none"> <li>• Se valida el usuario (cliente)</li> <li>• El vendedor ingresa al menú de ventas para seleccionar el tipo de documento de pago.</li> <li>• El vendedor ingresa los datos de la venta.</li> <li>• El sistema muestra los datos y los montos totales de la venta.</li> <li>• El sistema emite el detalle del comprobante de pago de la venta.</li> <li>• Finaliza el caso de uso.</li> </ul>
Flujo Alternativo	<ul style="list-style-type: none"> <li>• El sistema no muestra información de los datos guardados previamente.</li> <li>• Finaliza el caso de uso.</li> </ul>
Prioridad	Alto.

## PROTOTIPO DE SOFTWARE





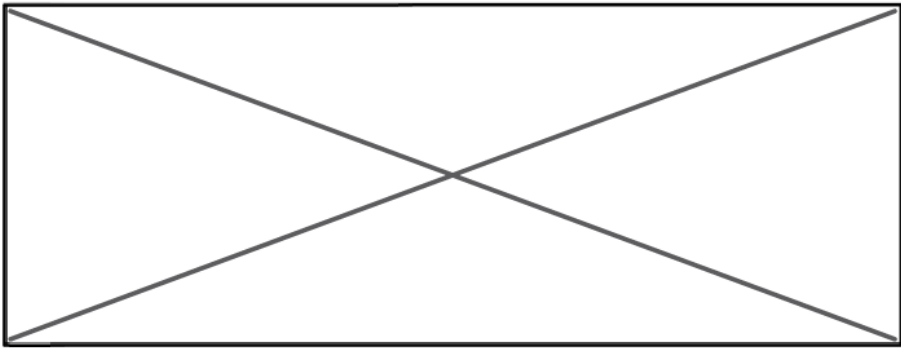
Login - Restaurante "Mechita"



USUARIO


CONTRASEÑA

Restaurante "Mechita"

CLIENTE	PRODUCTO	PEDIDOS	TOTAL DE PEDIDOS Y RECAUDACION
			
			



**Pedidos - Restaurante "Mechita"**



FUENTE  CANTIDAD  PRECIO

BEBIDA  CANTIDAD  PRECIO

**Registrar** PORCION  CANTIDAD  PRECIO

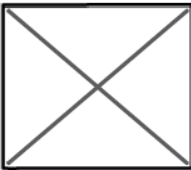



TABLA DE PEDIDOS

**Buscar**

**Editar** **Eliminar**

FUENTE	CANTIDAD	PRECIO	BEBIDA	CANTIDAD	PRECIO	PROCION	CANTIDAD	PRECIO

**Cliente - Restaurante "Mechita"**



NOMBRE

APELLIDO PATERNO

APELLIDO MATERNO

DNI

E-MAIL

TELEFONO

**Registrar**

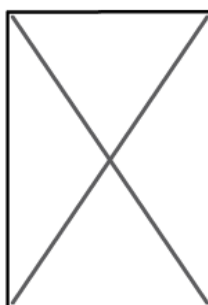



TABLA CLIENTE

**Buscar**

CODIGO	NOMBRE	APELLIDO PATERNO	APELLIDO MATERNO	DNI	DIRECCION	TELEFONO

Restaurante "Mechita" - Comprobante de pago



MESERO

FECHA DE EMISION

CLIENTE

TIPO DE PAGO

TOTAL

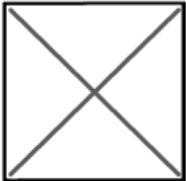


TABLA COMPROBANTE DE PAGO

CODIGO	MESERO	FECHA DE EMISION	CLIENTE	TIPO DE PAGO	TOTAL

Total de pedidos y Recaudacion - Restaurante "Mechita"

FUENTE	CANTIDAD

FUENTE	CANTIDAD	PRECIO	TOTAL

## REFERENCIAS

Wikipedia. (12 de Abril de 2018). *Wikipedia*. Obtenido de [https://es.wikipedia.org/wiki/Java\\_\(lenguaje\\_de\\_programaci%C3%B3n\)](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n))

Wikipedia. (6 de Abril de 2018). *Wikipedia*. Obtenido de [https://es.wikipedia.org/wiki/Lenguaje\\_unificado\\_de\\_modelado](https://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado)