



UNIVERSIDAD PARTICULAR DE CHICLAYO



**FACULTAD DE ARQUITECTURA Y URBANISMO, INGENIERIAS,
ARTE & DISEÑO**

ESCUELA PROFESIONAL DE INGENIERIA INFORMATICA Y DE SISTEMAS

TESIS

**“Implementación del Sistema de Gestión de Ventas de platos a la
carta en VB.NET consumiendo un servicio API Rest para el
Restaurante-Cevicheria ‘Flor de Mar’ de la ciudad de Chiclayo”**

**PARA OPTAR EL TÍTULO PROFESIONAL DE:
INGENIERO INFORMÁTICO Y DE SISTEMAS**

PRESENTADO POR:

Bach. Erick Vera Oliva

ASESOR

Ing. Wilton Carpio Campos.

(ORCID: 000-0002-2504-3539)

Pimentel – Perú, 2021

DEDICATORIA

A Dios por haberme dado la vida, por ser mi guía y mi fortaleza para llegar hasta este momento de mi formación profesional. A mis padres Walter Vera Julca y Josefa Oliva Vera que siempre han confiado en mí y siempre me han brindado su apoyo incondicional para lograr cada meta propuesta. A mis hermanas Fiorella, Yesica, Nicol por siempre estar dispuestas a ayudarme en todo momento. A mi esposa Gianella por acompañarme en todo momento y a mi hijo Kaleth por alegrar mis días y ser el motor y motivo que me impulsa a seguir adelante.

AGRADECIMIENTO

A Dios que siempre me fortalece y con sus bendiciones me ha permitido llegar a culminar esta etapa de mi vida.

A mis padres que siempre me han brindado apoyo incondicional durante toda mi formación profesional y me han dado ánimos para seguir adelante.

A mis hermanas por ser quienes me motivan a seguir con mis metas trazadas.

Agradezco a mi alma mater por albergarme toda la carrera universitaria y centrarme de conocimientos y valores. Asimismo a mis docentes universitarios que con un granito de arena aportaron en mi formación profesional.

DECLARACIÓN DE AUTENTICIDAD

Yo, Bach. Erick Vera Oliva, identificado con DNI 70891382 egresado de la Escuela de Ingeniería Informática y de Sistemas de la Facultad de Arquitectura y Urbanismo, a efectos de cumplir con las disposiciones vigentes consideradas en el reglamento de Grados y Títulos de la Universidad Particular de Chiclayo, declaro bajo juramento que el informe del trabajo de suficiencia profesional denominada:

“Implementación del Sistema de Gestión de Ventas de platos a la carta en VB.NET consumiendo un servicio API Rest para el Restaurante-Cevicheria ‘Flor de Mar’ de la ciudad de Chiclayo”

Que:

1. El informe es de mi autoría.
2. He respetado y seguido los lineamientos de diseño, reglamento y parámetros establecidos para el desarrollo del informe.
3. El informe no ha sido plagiado, pues no ha sido objeto de publicaciones, ni presentada anteriormente para obtener un grado académico previo o título profesional.
4. Los datos presentados son resultado de visitas, entrevistas y consultas y que éstos no han sido falseados, pues los resultados contribuirán a conocer la realidad problemática con respecto a la investigación.

En tal sentido asumo la responsabilidad que corresponda ante cualquier falsedad, ocultamiento u omisión tanto de los documentos como de información aportada por lo cual me someto a lo dispuesto en las normas académicas de la Universidad de Chiclayo.

Pimentel, septiembre de 2021

El autor

PRESENTACIÓN

Señores Miembros del Jurado:

De conformidad y cumpliendo lo estipulado en el Reglamento de Grados y Títulos de la Facultad de Arquitectura y Urbanismo e Ingenierías de la Universidad Particular de Chiclayo, para optar el Título Profesional de Ingeniero Informático y de Sistema, someto a vuestra consideración el presente informe Titulado:

“Implementación del Sistema de Gestión de Ventas de platos a la carta en VB.NET consumiendo un servicio API Rest para el Restaurante-Cevicheria ‘Flor de Mar’ de la ciudad de Chiclayo”

Gracias

ÍNDICE GENERAL

DEDICATORIA.....	2
AGRADECIMIENTO.....	3
DECLARACIÓN DE AUTENTICIDAD	4
PRESENTACIÓN	5
ÍNDICE GENERAL	6
ÍNDICE DE FIGURAS	8
ÍNDICE DE TABLAS	10
RESUMEN	12
ABSTRACT	13
CAPITULO I: INTRODUCCIÓN.....	15
1.1. Realidad Problemática	15
1.2. Formulación del Problema	15
1.3. Hipótesis	15
1.4. Objetivos.....	16
1.4.1. Objetivo General	16
1.4.2. Objetivos Específicos.....	16
CAPITULO II: BASES TEÓRICAS	18
2.1. Antecedentes de la Investigación.....	18
2.2. Marco Teórico	22
2.3. Definición de términos.....	29
CAPÍTULO III: MARCO METODOLÓGICO	33
3.1. Variables.....	33
3.2. Operacionalización de las variables:	33
3.3. Metodología	34
3.4. Población, muestra y muestreo.....	34

3.5.	Técnicas e instrumentos de recolección de datos.	34
3.6.	Métodos de análisis de datos	35
3.7.	Aspectos éticos	35
CAPITULO IV: RESULTADOS		37
CAPITULO V: DISCUSIÓN		43
CAPÍTULO VI: CONCLUSIONES.....		45
CAPÍTULO VII: RECOMENDACIONES		47
CAPÍTULO VIII: PROPUESTA TECNOLÓGICA		49
8.1.	Generalidades.....	49
8.2.	Análisis.	52
8.3.	Diseño.....	57
8.4.	Implementación.....	88
8.5.	Presupuesto.....	101
CAPITULO IX: BIBLIOGRAFIA		103
ANEXO		107

ÍNDICE DE FIGURAS

Imagen No. 1 ¿Los Pedidos realizados son atendidos en forma inmediata?	
Resultados	37
Imagen No. 2 ¿Recibe los detalles de los productos cuando lo solicita? Resultados	38
Imagen No. 3 ¿Tuvo inconvenientes respecto a su pago total en caja? Resultados	39
Imagen No. 4 ¿Los datos de la comanda contenía exactamente los productos con sus respectivos precios? Resultados	40
Imagen No. 6 Modelo del negocio.	52
Imagen No. 7 Diagrama de Casos de uso - Almacén	53
Imagen No. 8 Diagrama de Casos de uso -Control de de área con sus respectivas mesas	54
Imagen No. 9 Diagrama de casos de uso – gestión de ventas.	55
Imagen No. 10 Diagrama de casos de uso - Gestión de cobranzas	56
Imagen No. 11 Modelo E-R 1/8	57
Imagen No. 12 Modelo E-R 2/8	58
Imagen No. 13 Modelo E-R 3/8	59
Imagen No. 14 Modelo E-R 4/8	60
Imagen No. 15 Modelo E-R 5/8	61
Imagen No. 16 Modelo E-R 6/8	62
Imagen No. 17 Modelo E-R 7/8	63
Imagen No. 18 Modelo E-R 8/8	64
Imagen No. 19 Formulario Acceso al Sistema.....	88
Imagen No. 20 Formulario Menú Principal.	89
Imagen No. 21 Formulario Registro General de Productos.	90
Imagen No. 22 Formulario de características del producto.....	91
Imagen No. 23 Formulario Ficha Otros.	92
Imagen No. 24 Formulario Registro general de personas.	93
Imagen No. 25 Formulario para agregar o modificar registro de persona.	94
Imagen No. 26 Formulario distribución de secciones.	95
Imagen No. 27Formulario para Asignación de mesas.	96
Imagen No. 28Formulario de Registro de venta según mesa.	97
Imagen No. 29 Formulario de gestión de comanda.	98

Imagen No. 30 Formulario selección de ítems para venta.	99
Imagen No. 31 Formulario para gestión de pago según comanda.	100

ÍNDICE DE TABLAS

Tabla 1 Operacionalización de las variables	33
Tabla 2 Tabla: ctapagar_abono.....	65
Tabla 3 Tabla: almacen.....	66
Tabla 4 Tabla 3: categoria_producto	66
Tabla 5 Tabla 4: comanda_mesa	67
Tabla 6 Tabla 5: configuración	68
Tabla 7 Tabla 6: cuadre_caja.....	68
Tabla 8 Tabla 7: ctas_cobrar.....	69
Tabla 9 Tabla 8: cta_pagar.....	70
Tabla 10 Tabla 9: datosempresa.....	71
Tabla 11 Tabla 10: ctapagar_detalle_abono	71
Tabla 12 Tabla 11: comanda_detalle	72
Tabla 13 Tabla 12: docpago_ctacobrar_detalle_impreso	73
Tabla 14 Tabla 13: detalles_ingreso_productos	74
Tabla 15 Tabla 14: documento_salida_detalle	74
Tabla 16 Tabla 15: ctapagar_detpago.....	75
Tabla 17 Tabla 16: docpago_ctacobrar	75
Tabla 18 Tabla 17: estadomesa	76
Tabla 19 Tabla 18: facturacion_electronica_documentoinformado	76
Tabla 20 Tabla 19: marca_producto.....	77
Tabla 21 Tabla 20: medio_pago.....	77
Tabla 22 Tabla 21: docpago_ctacobrar_mediopago.....	78
Tabla 23 Tabla 22: rol	78
Tabla 24 Tabla 23: mesa.....	79
Tabla 25 Tabla 24: persona.....	79
Tabla 26 Tabla 25: Persona	80
Tabla 27 Tabla 26: seccion_mesa.....	81
Tabla 28 Tabla 27: producto.....	81
Tabla 29 Tabla 28: documento_ingreso_producto.....	82
Tabla 30 Tabla 29: documento_salida_producto	83
Tabla 31 Tabla 30: seriesdoc_emitir.....	84
Tabla 32 Tabla 31: stock_producto_almacen	84

Tabla 33	Tabla 32: sucursal	85
Tabla 34	Tabla 33: sucursal_pc_registrado	85
Tabla 35	Tabla 34: tipo_doc_identidad	85
Tabla 36	tipodoc_comprobantepago	86
Tabla 37	Tabla 36: tiposervicio_compra	86
Tabla 38	Tabla 37: tipotransaccion	86
Tabla 39	Tabla 38: tipo_movimiento_mercaderia	87
Tabla 40	Recursos y Presupuestos	101
Tabla 41	Recuperación.	101

RESUMEN

El presente trabajo de investigación tiene como objetivo Implementar el Sistema de Gestión de Ventas de platos a la carta en VB.NET consumiendo un servicio API Rest para el Restaurante-Cevicheria 'Flor de Mar' de la ciudad de Chiclayo". Es una investigación de tipo aplicada; cuyo diseño de investigación es el no experimental. La población a considerar son los clientes que visitaron al negocio la primera semana de mayo y cuya muestra corresponde a 30 clientes con un muestreo no probabilístico. Se aplicó la técnica de la encuesta con el instrumento cuestionario. Entre las principales conclusiones tenemos que el postgresql cumplió con las expectativas respecto a un alto performance en la gestión de una base de datos como almacenamiento, acceso, seguridad y respaldo de datos. La otra conclusión es saber que los servicios que ofrece Windows en integración con .net a través de una Api Rest, permitió brindar mayor seguridad a la aplicación y a los datos que utiliza.

Palabras clave: Gestión de ventas, Api, Rest, .Net, Postgresql.

ABSTRACT

The present research work aims to Implement the Sales Management System for à la carte dishes in VB.NET, consuming an API Rest service for the 'Flor de Mar' Restaurant-Cevicheria in the city of Chiclayo". It is an applied research; whose research design is non-experimental. The population to be considered are customers who visited the business the first week of May and whose sample corresponds to 30 customers with a non-probability sampling. The survey technique was applied with the questionnaire instrument. Among the main conclusions we have that the postgresql fulfilled the expectations regarding a high performance in the management of a database such as storage, access, security and data backup. The other conclusion is to know that the services offered by Windows in integration with .net through an Api Rest, allowed to provide greater security to the application and the data it uses.

Keyword: Sales management, Api, Rest, .Net, Postgresql.

CAPÍTULO I

INTRODUCCIÓN

CAPITULO I: INTRODUCCIÓN

1.1. Realidad Problemática

El Restaurant-Cevichería “Flor de Mar” tiene dos años de funcionamiento y que ofrece platos derivados del pescado, así como otros productos propios de un restaurant. El restaurant-cevichería tiene como local un inmueble de dos pisos. Los dos pisos son ocupados para el negocio. En la primera planta se encuentra el área de recepción y caja, también está dos ambientes pequeños para familia y la cocina. En el segundo piso se encuentra mesas de atención para clientes.

Al inicio, el número de personas que frecuentaban el negocio era muy reducido. Luego, por muchos motivos y estrategias de mercado el número de clientes fue aumentando. El personal de atención anota en una comanda los pedidos del cliente y luego lo lleva a caja. En caja el responsable lo anota en forma manual los datos en el comprobante solicitado, factura o boleta. En muchos de los casos lo anota en forma global. Al finalizar un periodo no contaban con la información necesaria que les permita conocer en forma exacta el detalle de las ventas, así como el detalle de los productos con mayor demanda y menor demanda.

1.2. Formulación del Problema

¿La Implementación del Sistema de Gestión de Ventas de platos a la carta en VB.NET consumiendo un servicio API Rest para el Restaurante-Cevicheria ‘Flor de Mar’ de la ciudad de Chiclayo brindará con mayor exactitud el detalle de los platos consumidos durante el periodo febrero a junio 2021?

1.3. Hipótesis

La Implementación del Sistema de Gestión de Ventas de platos a la carta en VB.NET consumiendo un servicio API Rest para el Restaurante-Cevicheria ‘Flor de Mar’ de la ciudad de Chiclayo permitirá conocer con

mayor exactitud el detalle de los platos consumidos durante el periodo febrero a junio 2021.

1.4. Objetivos

1.4.1. Objetivo General

Implementar el Sistema de Gestión de Ventas de platos a la carta en VB.NET consumiendo un servicio API Rest para el Restaurante-Cevicheria 'Flor de Mar' de la ciudad de Chiclayo para brindar con mayor exactitud el detalle de los platos consumidos durante el periodo febrero a junio 2021.

1.4.2. Objetivos Específicos

- a. Determinar los requerimientos funcionales mediante la herramienta visual UML.
- b. Implementar la aplicación que permita utilizar servicios API
- c. Gestionar la base de datos que ofrezca un alto rendimiento en el almacenamiento, acceso, seguridad y respaldo de datos.
- d. Utilizar un servicio Api Rest para brindar mayor seguridad a la aplicación y a los datos.

Capítulo II.

BASES TEÓRICAS

CAPITULO II: BASES TEÓRICAS

2.1. Antecedentes de la Investigación

A. Nombre de proyecto: “Implementación De Unaaplicación Móvil Para La Empresa Gourmedi Eirl-Huaraz; 2018”.

Año: 2018

Lugar: Universidad Católica Los Angeles de Chimbote

Autor: Katherin Paola Gargate Leon

Resumen:

“La presente tesis fue desarrollada bajo la línea de investigación: Implementación de las Tecnologías de la Información y Comunicación (TIC) para la mejora continua de la calidad en las organizaciones del Perú, de la escuela profesional de Ingeniería de Sistemas de la Universidad Católica los Ángeles de Chimbote. Tuvo como objetivo: Realizar la implementación una aplicación móvil para la empresa GOURMEDI EIRL, que mejore la gestión del servicio y el control en los pedidos a domicilio – Huaraz; 2019. El tipo de la investigación es descriptiva y con un enfoque cuantitativo, teniendo un diseño de tipo no experimental y de corte transversal, tomando como población a 173 y la muestra es 120 clientes que realizaron pedidos a la empresa del 2018, el instrumento para la recolección de datos fue el cuestionario mediante la técnica de la encuesta, obteniéndose como resultado para la dimensión 01: Situación actual de la empresa, un 70.00% determinó que NO están satisfechos con el proceso actual de pedidos, respecto a la dimensión 02: Necesidad de implementar una aplicación móvil, el 60.00% concluyo indicando que SI tiene un alto nivel de funcionabilidad. Estos resultados coinciden con la hipótesis quedando demostrada y aceptada. La investigación queda debidamente justificada de realizar la implementación de una aplicación móvil para la empresa GOURMEDI EIRL. El alcance de la investigación contempla desarrollo de aplicaciones móviles y el servicio de delivery.” (Gargate,2018)

B. Nombre de proyecto: “Sistema Informático De Control De Ventas Para El Restaurant Milagritos, Casma”.

Año: 2018

Lugar: Universidad San Pedro.

Autor: Jeri Joel Bernuy Paz

Resumen:

“La presente investigación tuvo como propósito el desarrollo de un sistema informático de control de ventas para el restaurant Milagritos de Casma, para que de esta manera pueda ayudar al personal operativo, clientes y a la administración del Restaurant; mediante la sistematización en los procesos que desarrolla el Restaurant, permitiendo llevar un mayor control, suministración y administración de cada uno de los procesos (producción, control de insumos y ventas) enfocado a las comandas, órdenes de compras, reservaciones y documentos de venta. Para el desarrollo del sistema se hizo uso de herramientas como Microsoft Visual Studio 2010 Express, es una versión gratuita, y puede ser usada en empresas pequeñas y medianas ya que cuenta con lo necesario para poder administrarlas; y para el manejo de la base de datos se usó del SQL Server 2010 Express, ésta versión es gratuita, y permite la construcción de aplicaciones robustas, ofreciendo una sencilla pero potente base de datos. Por lo tanto la empresa no tuvo que gastar en licencia para éstos productos, ya que además de ser gratuitos le brindarán lo necesario para el manejo de su empresa. Gracias a las herramientas utilizadas y a la presente investigación mencionada se pudo lograr tener un resultado favorable donde se presenta una solución informática para el control de ventas en el restaurant Milagritos, Casma donde ahora le permite tener un mejor control de sus ventas y lograr obtener una gran variedad de información generado por reportes rápidamente que son útiles para el administrador del restaurante. Así poder ver los logros y llevar una buena gestión del negocio.” (Bernuy, 2018)

C. Nombre de proyecto: “Análisis, Diseño E Implementación De Una Guía Gastronómica Para La Administración Y Ubicación De Restaurantes En Entorno Web”.

Año: 2015

Lugar: Universidad Católica del Perú.

Autor: Christian Pérez Ortiz

Resumen:

“En el actual contexto peruano, la gastronomía peruana es un mercado en constante movimiento tanto de alzas como de bajas. Esto quiere decir que, así como se van formando nuevos negocios de restaurantes, también van desapareciendo otros. Asimismo, algunos de estos permiten agilizar su ubicación a través de portales web, de tal forma que facilita esta actividad para los consumidores. Frente a este contexto, aparece el problema de la dificultad de la ubicación de restaurantes. Por un lado, la distribución de portales web a lo largo de Internet, desemboca en una falta de centralización, lo cual no permite a los consumidores ubicar fácilmente un restaurante que se acomode a sus necesidades. Por otro lado, no todos los restaurantes cuentan con un portal web propio, lo cual genera una falta de medios de comunicación entre los restaurantes y clientes. En este proyecto de fin de carrera, se brindará una propuesta de solución tanto al problema de descentralización de los restaurantes en Internet como a la falta de medios de comunicación entre restaurantes y consumidores. Para desarrollar esta solución se abarcará el análisis, diseño e implementación de un sistema de información en base a metodologías y procedimientos de ingeniería de software. Asimismo, se realizará el desarrollo de algoritmos para brindar soporte a algunas funcionalidades del sistema”. (Pérez, 2015)

D. Nombre de proyecto: “Desarrollo De Una Api Rest Con Sus Aplicaciones Web Y Móvil Para La Venta De Ropa Online De La Empresa Roosman”.

Año: 2017

Lugar: Universidad Central del Ecuador.

Autores: Caina Aysabucha, Darwin Rodolfo.

Santos Hernández, Wagner David.

Serrano Parreño, Jairo Adrián.

Resumen

“Al implementar sistemas informáticos orientados a automatizar el proceso de venta de una empresa, se aumenta el alcance de la misma generando mayores ganancias, además de permitir la expansión de la empresa en otros mercados del país. Para esto se analizó los procesos que se deseaba automatizar, mejorar e implementar. Una vez que se definió estos requerimientos, se desarrolló un API REST con el cual se maneja toda la lógica de negocio, permitiendo realizar una aplicación Web y una aplicación móvil con lo cual se busca impulsar el crecimiento de la empresa. Para el almacenamiento de datos se utilizó bases de datos no relacionales (NoSQL) y para el desarrollo Web y aplicación móvil se manejó AngularJS, permitiendo la reutilización de código para ambos desarrollos y al establecer un core común, la escalabilidad del sistema es mucho más simple.”. (Caina y otros, 2017)

2.2. Marco Teórico

2.2.1. Lenguaje de Programación Microsoft Visual Basic .NET 2010 (Duncan, 2012)

Duncan no dice que:

“Net es una plataforma para el desarrollo de aplicaciones, que integra múltiples tecnologías que han ido apareciendo en los últimos años como: ASP.NET, ADO.NET, LINQ, WPF, Silverlight, junto con el potente entorno integrado de desarrollo Visual Studio que permite desarrollar múltiples tipos de aplicaciones como:

- *Aplicaciones de línea de comando.*
- *Servicios de Windows.*
- *Aplicaciones de escritorio con Windows Forms o WPE.*
- *Aplicaciones web con el framework ASP.NET o Silverlight.*
- *Aplicaciones distribuidas SOA mediante servicios web.*
- *Aplicaciones para dispositivos móviles con Windows Mobile.* “(Duncan, 2010)

También nos hace referencia a su entorno integrado de desarrollo:

“Visual Studio es un entorno integrado de desarrollo (IDE) compartido y único para todos los lenguajes .NET. El entorno proporciona acceso a todas las funcionalidades del .NET Framework, así como a muchas otras funcionalidades que hacen que el desarrollo de aplicaciones sea más ágil.
(Duncan, 2010)

2.2.2. REST.

La Escuela de Negocios de la Innovación y los Emprendedores (IEBS) el cual tiene mas de 13 años al servicio de los diseñadores, programadores, empresarios, analistas y a todos aquellos que se involucre en la creación e innovación de un negocio define en su portal web (<https://www.iebschool.com/blog/que-es-api-rest-integrar->

negocio-business-tech/) la terminología REST:

“API es un conjunto de reglas y especificaciones que las aplicaciones pueden seguir para comunicarse entre ellas. Para que lo entendamos, el uso de una API es el mecanismo más útil para conectar dos softwares entre sí, de esta manera, podemos garantizar el intercambio de mensajes o datos en formato estándar.

Lo que se pretende con esta herramienta, es simplificar el trabajo de los programadores y desarrolladores puesto que para crearlas o programarlas, no necesitan picar código desde cero, sino hacer solo modificaciones dependiendo el uso que se le va a dar y el tipo de sitio dónde se quiera implementar. Una de las principales funciones de la API es que pueden utilizarse para comunicarse con un sistema operativo que tenga bases de datos o protocolos de comunicaciones”. (IEBS, 2018).

En el libro de Enrique Amodeo nos indica un poco mas claro porque o cuando debemos utilizar REST en nuestras plataformas o aplicaciones: *“¿Queremos diseñar un sistema de servicios web? Pues lo más sensato es ver como funciona la web, e investigar el único caso de éxito de interoperabilidad a escala global que existe en la actualidad. ¿Cuál es tal caso de éxito? ¿Acaso existe? Sí, y es ni más ni menos que la World Wide Web. Sí querido lector, las páginas web que navegamos a diario son un caso real de interoperabilidad a escala global, sólo es necesario usar un navegador decente que entienda los formatos disponibles para cada página (HTML, GIF, JPEG, códecs de video, etc) para poder consumirlas. No importa que el fabricante del navegador y el autor de la página no hayan interactuado en la vida, o que los autores de las páginas sean totalmente independientes entre si”. (Amodeo, 2013).*

2.2.3. API.

El término API es uno de los mas empleados por hoy a nivel de los desarrolladores de software o aplicaciones y el cual se ha convertido en una pieza clave de la reusabilidad de piezas de código. La

empresa REDHAT, muy conocido de hace mas de 30 años por sus productos como sistemas operativos basados en el Kernel, publica en su portal web (<https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>) la definición del término API en forma muy concreta el cual a la letra nos dice: *“Una API es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones. API significa interfaz de programación de aplicaciones. Las API permiten que sus productos y servicios se comuniquen con otros, sin necesidad de saber cómo están implementados. Esto simplifica el desarrollo de las aplicaciones y permite ahorrar tiempo y dinero. Las API le otorgan flexibilidad; simplifican el diseño, la administración y el uso de las aplicaciones, y proporcionan oportunidades de innovación, lo cual es ideal al momento de diseñar herramientas y productos nuevos (o de gestionar los actuales). A veces, las API se consideran como contratos, con documentación que representa un acuerdo entre las partes: si una de las partes envía una solicitud remota con cierta estructura en particular, esa misma estructura determinará cómo responderá el software de la otra parte”*. (Redhat, 2019).

ABC el cual es un diario español que existe desde el año 1903, también publicó en su portal web (<https://www.abc.es/tecnologia/consultorio/20150216/abci--201502132105.html>) una sección que explica la definición del API y para qué sirve. En este texto recopilado podemos encontrar que nos indica lo siguiente: *“...son un conjunto de comandos, funciones y protocolos informáticos que permiten a los desarrolladores crear programas específicos para ciertos sistemas operativos. Las API simplifican en gran medida el trabajo de un creador de programas, ya que no tiene que «escribir» códigos desde cero. Estas permiten al informático usar funciones predefinidas para interactuar con el sistema operativo o con otro programa.*

Desde el show HowStuffWorks señalan que le usuario nunca ve las API en pleno proceso de trabajo, pero sí detalles de sus acciones. La API es una interfaz o rostro que sólo da la cara al software. Es

decir, el el usuario no ve eso. Con las API, las aplicaciones como Facebook, Twitter, y Candy Crush se pueden comunicar entre ellas sin que el usuario tenga que intervenir o incluso, percatarse...”. (ABC, 2015).

2.2.4. NET.

En el sitio web del portal líder en formación de Emagister (<https://www.emagister.com/blog/que-es-el-net-para-que-sirve/>), el cual lleva mas de 20 años en el mercado y se ubica en mas de 14 países a nivel mundial el cual ofrece **oferta formativa de** cursos, formación profesional, grados, másteres, postgrados, oposiciones y cursos subvencionados, entre diferentes universidades y centros nos comenta sobre el NET:

“Se trata de una plataforma para el desarrollo de software que fue lanzada por Microsoft con la finalidad de fusionar su amplio catálogo de productos, que va desde sus múltiples sistemas operativos hasta herramientas de desarrollo. Desde un punto de vista tecnológico, lo que se pretendía con la creación de .NET era poder desarrollar aplicaciones y sistemas que fueran independientes de la arquitectura física y del sistema operativo sobre el que se ejecutaran. La repercusión fue muy grande, ya que Microsoft estableció un estándar de intercambio de información entre sus productos llamado ‘XML’, que posiblemente a muchos de vosotros os suene. (Emagister, 2020) Desde el punto de vista comercial, .NET podría interpretarse como la alternativa por parte de Microsoft en el sector de los desarrollos web para competir con la plataforma Java de Oracle Corporation, así como con los diversos framework basados en PHP. (Emagister, 2020)

Al tratarse de una plataforma de propósito general, se pueden realizar tanto desarrollos web, como programas de escritorio o aplicaciones para dispositivos móviles. Además, favorece el desarrollo en multiplataforma, como por ejemplo el que una misma aplicación pueda correr indistintamente en los diferentes sistemas operativos móviles como IOS, Android o Windows Phone

garantizando la comunicación entre los diferentes dispositivos”.
(Emagister, 2020)

2.2.5. PostgreSQL.

En el portal de la misma industria del gestor en mención, nos señala aspectos relacionados con este gestor de base de datos:

“PostgreSQL es un potente sistema de base de datos relacional de objetos de código abierto que utiliza y amplía el lenguaje SQL combinado con muchas características que almacenan y escalan de forma segura las cargas de trabajo de datos más complicadas. Los orígenes de PostgreSQL se remontan a 1986 como parte del proyecto POSTGRES en la Universidad de California en Berkeley y tiene más de 30 años de desarrollo activo en la plataforma central. PostgreSQL se ha ganado una sólida reputación por su arquitectura probada, confiabilidad, integridad de datos, conjunto de características robustas, extensibilidad y la dedicación de la comunidad de código abierto detrás del software para brindar soluciones innovadoras y de alto rendimiento de manera consistente. PostgreSQL se ejecuta en todos los principales sistemas operativos, ha sido compatible con ACID desde 2001 y tiene potentes complementos como el popular extensor de base de datos geoespacial PostGIS. No es de extrañar que PostgreSQL se haya convertido en la base de datos relacional de código abierto elegida por muchas personas y organizaciones”. (Postgresql, 2020)

2.2.6. Sistema.

En el sitio web <https://www.significados.com/sistema/> encontramos una explicación al respecto:

“Un sistema es un conjunto de elementos relacionados entre sí que funciona como un todo. Si bien cada uno de los elementos de un sistema puede funcionar de manera independiente, siempre formará

parte de una estructura mayor. Del mismo modo, un sistema puede ser, a su vez, un componente de otro sistema". (Significados, 2020)

2.2.7. Gestión de la información.

La Universidad de Ciencias Médicas de Santiago de Cuba publica en su portal web la definición del término Gestión de información el cual muchas veces lo reducimos en la palabra gestión cuando se trata de sistemas informáticos. En este portal <https://instituciones.sld.cu/toximed/2017/04/16/que-es-gestion-de-la-informacion/> nos dice lo siguiente:

“Es la denominación convencional de un conjunto de procesos por los cuales se controla el ciclo de vida de la información, desde su obtención (por creación o captura), hasta su disposición final (su archivo o eliminación). Tales procesos también comprenden la extracción, combinación, depuración y distribución de la información a los interesados. El objetivo de la gestión de la información es garantizar la integridad, disponibilidad y confidencialidad de la información. En el contexto de las organizaciones, la gestión de la información se puede identificar como la disciplina que se encargaría de todo lo relacionado con la obtención de la información adecuada, en la forma correcta, para la persona indicada, al coste adecuado, en el momento oportuno, en el lugar apropiado y articulando todas estas operaciones para el desarrollo de una acción correcta. En este contexto, los objetivos principales de la Gestión de la Información son: maximizar el valor y los beneficios derivados del uso de la información, minimizar el coste de adquisición, procesamiento y uso de la información, determinar responsabilidades para el uso efectivo, eficiente y económico de la información y asegurar un suministro continuo de la información. (Infomed, 2020).

2.2.8. Ventas.

RDSTATION, empresa constituida en 2011 con la finalidad de apoyay a todos los tipos de negocio el como aprovechar los

beneficios del Marketing Digital, nos ilustra respecto al término ventas y el cual tiene mucha aplicabilidad en el momento de desarrollar aplicaciones:

“Un proceso de ventas es nada más que un modelo de reglas, acciones y actividades, utilizadas por todos los involucrados del área comercial y que contenga las siguientes características:

Educativo: capaz de ser enseñado a los involucrados, pudiendo ser utilizado por cualquiera del área;

Medible: tiene condiciones de crecer de forma uniforme, soportando el aumento de demanda;

Previsible: posee indicadores claros para el análisis de resultados e incluye expectativas claras de resultados para ser cumplidos en el período”. (Rdstation, 2020)

2.2.9. Plato a la carta

ECURED, el cual es una enciclopedia colaborativa en la red cubana en idioma español el cual surge de la intención de crear y difundir el conocimiento para todos, en su portal [https://www.ecured.cu/Carta_\(men%C3%BA_de_restaurante\)](https://www.ecured.cu/Carta_(men%C3%BA_de_restaurante)), nos explica respecto al término carta y por tal podemos deducirlo la expresión plato a la carta. ECURED, nos señala claramente lo siguiente:

“La carta del restaurante es uno de los instrumentos más importantes dentro del negocio gastronómico o (carta de precio) ya que es la pieza a través de la cual el cliente que ya nos ha elegido seleccionará qué va a consumir: el menú es el órgano de comunicación entre el restaurante y el cliente y por lo tanto debe promover un encuentro entre las preferencias o gustos del cliente y la oferta del local” (Ecured, 2020).

Por lo tanto, el plato a la carta es simplemente un plato preparado a solicitud del comensal, cliente o persona y no en forma masiva cómo lo es un plato de menú.

2.3. Definición de términos

Base de datos: Conjunto de registros de estructura homogénea que almacena datos y su acceso como su manipulación se realiza mediante un gestor que lo regula y brinda los mecanismos de acceso.

Carta: En un negocio de alimentos es la lista o relación de platos o alimentos, así como bebidas y relacionados con sus respectivos precios que brinda un negocio para el consumo de los mismos.

Consumir: El uso, disfrute o servirse de cierta cosa, material o inmaterial mediante un pago económico para satisfacer necesidades o gustos.

Hipótesis: Suposición que se realiza a partir de unos datos o situación que permite iniciar una investigación.

Implementar: Desarrollar un componente físico o lógico que permita satisfacer ciertos requerimientos o alcanzar objetivos planteados.

Investigación aplicada:

Es un tipo de diseño de investigación que busca resolver un problema específico o brindar soluciones innovadoras a problemas que afectan a un individuo, grupo o sociedad. A menudo se lo conoce como un método científico de investigación o investigación contractual porque implica la aplicación práctica de métodos científicos a problemas cotidianos.

Gestión: conjunto de acciones, trámites u operaciones que se llevan a cabo para lograr objetivos o satisfacer alguna necesidad.

Negocio: Ocupación, actividad o trabajo que se realiza para obtener un beneficio, especialmente el que consiste en realizar

operaciones comerciales, comprando y vendiendo mercancías o servicios

Plato: Respecto a un restaurant o similar es una porción de alimento preparado o cocinado que se sirve para su consumo.

Problema: Es un tema o hecho que requiere ser investigado para luego ofrecer propuestas, soluciones o alternativas de mejora.

Problemática: Conjunto de incidentes que no permiten lograr o alcanzar metas u objetivos y por el cual podemos comprender que existe un problema a solucionar.

Restaurant: Lugar o establecimiento de servicio en el cual se paga por la comida y bebida para ser consumidas en el mismo local o para llevarla (delivery).

Requerimiento: Necesidad o petición de una cosa que se considera necesaria para lograr alcanzar un objetivo o solucionar un problema o necesidad.

Servicio: Trabajo que realiza una persona, conjunto de personas o negocio para otra persona, personas o instituciones.

Sistema: Conjunto de elementos o componentes integrados y relacionados para cumplir con un objetivo de existencia.

Venta: Operación que se realiza entre dos instancias y el cual requiere el pago por la recepción de un objeto o servicio.

Variable:

Es aquella característica o propiedad que se asume ser la causa del fenómeno o problema estudiado.

Variable dependiente:

Es algo que depende de otros factores. Por ejemplo, el puntaje de una prueba podría ser una variable dependiente porque

podría cambiar dependiendo de varios factores, como cuánto estudió, cuánto durmió la noche antes de tomar la prueba o incluso cuánta hambre tenía cuando la tomó. Por lo general, cuando busca una relación entre dos cosas, está tratando de averiguar qué hace que la variable dependiente cambie de la forma en que lo hace.

. Variable independiente:

Es una variable que está aislada y no cambia por las otras variables que está tratando de medir. Por ejemplo, la edad de alguien puede ser una variable independiente. Otros factores (como lo que comen, cuánto van a la escuela, cuánta televisión ven) no van a cambiar la edad de una persona. De hecho, cuando busca algún tipo de relación entre variables, está tratando de ver si la variable independiente causa algún tipo de cambio en las otras variables, o variables dependientes.

Capítulo III:

MARCO METODOLÓGICO

CAPÍTULO III: MARCO METODOLÓGICO

3.1. Variables.

a) Variable Independiente: Sistema de Gestión de Ventas de platos a la carta en VB.NET consumiendo un servicio API Rest.

b) Variable dependiente: Detalle de los platos consumidos.

Implementar el Sistema de Gestión de Ventas de platos a la carta en VB.NET consumiendo un servicio API Rest para el Restaurante-Cevicheria 'Flor de Mar' de la ciudad de Chiclayo para brindar con mayor exactitud el detalle de los platos consumidos durante el periodo febrero a junio 2021.

3.2. Operacionalización de las variables:

Tabla 1 Operacionalización de las variables

Variable	Definición conceptual	Dimensiones	Indicadores
Dependiente	Detalle de los platos consumidos.	Recursos económicos	Incremento de ingresos
		Tiempo	Escala de tiempo
Independiente	Sistema de Gestión de Ventas de platos a la carta en VB.NET consumiendo un servicio API Rest	Accesibilidad	Seguridad. Interés.

Fuente: propia autoría

3.3. Metodología

3.3.1. Tipo de estudio:

Investigación aplicada. El Consejo Nacional de Ciencia, tecnología e Innovación tecnológica nos indica en su portal web que la investigación aplicada o de carácter tecnológico es otro tipo de investigación científica que conduce a la transformación material de las sociedades en el mundo. Esta se divide en investigación sustantiva que llega a ser plasmada en prototipos, y la investigación operativa que tiene que ver con los sistemas y enlaces virtuales y físicos que han experimentado un vertiginoso desarrollo en las ciencias de la comunicación e información. (Concytec,2020).

3.3.2. Diseño de investigación

Se ha considerado el diseño de investigación no experimental. Fernández Collado y Baptista Lucio nos señala que el diseño de investigación es no experimental cuando se basa en categorías, conceptos, variables, sucesos, comunidades o contextos que se dan sin la intervención directa del investigador, es decir; sin que el investigador altere el objeto de investigación. En la investigación no experimental, se observan los fenómenos o acontecimientos tal y como se dan en su contexto natural, para después analizarlos (Fernández & Baptista, 2014).

3.4. Población

En el portal oficial de la oficina de estadística de Chile, el término “población” se refiere al conjunto de elementos que se quiere investigar, estos elementos pueden ser objetos, acontecimientos, situaciones o grupo de personas (INE, 2021). La población a considerar son los clientes que han sido registrados en la primera quincena de mayo. Se ha considerado 30 clientes.

3.5. Técnicas e instrumentos de recolección de datos.

Se aplicó la encuesta mediante un enlace de Google forms. Según Pardinás la encuesta es un sistema de preguntas que tiene

como finalidad obtener datos para una investigación. También resulta ser un eficaz auxiliar en la observación científica (Pardinas, 1991).

3.6. Métodos de análisis de datos

El análisis de datos consiste en la realización de las operaciones a las que el investigador someterá los datos con la finalidad de alcanzar los objetivos del estudio (Juárez, 2012); para tal efecto se utilizó el software Microsoft excel para este fin.

3.7. Aspectos éticos

En la Universidad de Miami, específicamente en el Programa CITI nos señala que los principios éticos de la investigación son universales pero su aplicación requiere la adaptación a las condiciones locales sanitarias, socioculturales y económicas. La disponibilidad de los recursos necesarios para garantizar los principios no es universal y los procedimientos que se usan para asegurar que los estudios de investigación se ejecuten éticamente pueden no ser óptimos. Independientemente de las limitaciones locales, estos principios éticos deben guiar la conducta de quienes participen en la planeación, realización y patrocinio de la investigación con seres humanos. (citiprogram, 2021)

Capítulo IV:

RESULTADOS

CAPITULO IV: RESULTADOS

Se analizan los datos antes y después de la implementación:

Pregunta 1: ¿Cree que la recepción de sus pedidos personalizada?

RESPUESTA	PRE IMPLEM.	% PRE IMPLEM.	POST IMPLEM.	% POST IMPLEM.
SI	6	20.00%	28	93.33%
NO	19	63.33%	0	0.00%
A VECES	5	16.67%	2	6.67%
TOTAL	30	100.00%	30	100.00%

Tabla No. 1 ¿Los pedidos, son atendidos en forma inmediata? Resultados.

Fuente: Propia Autoría.

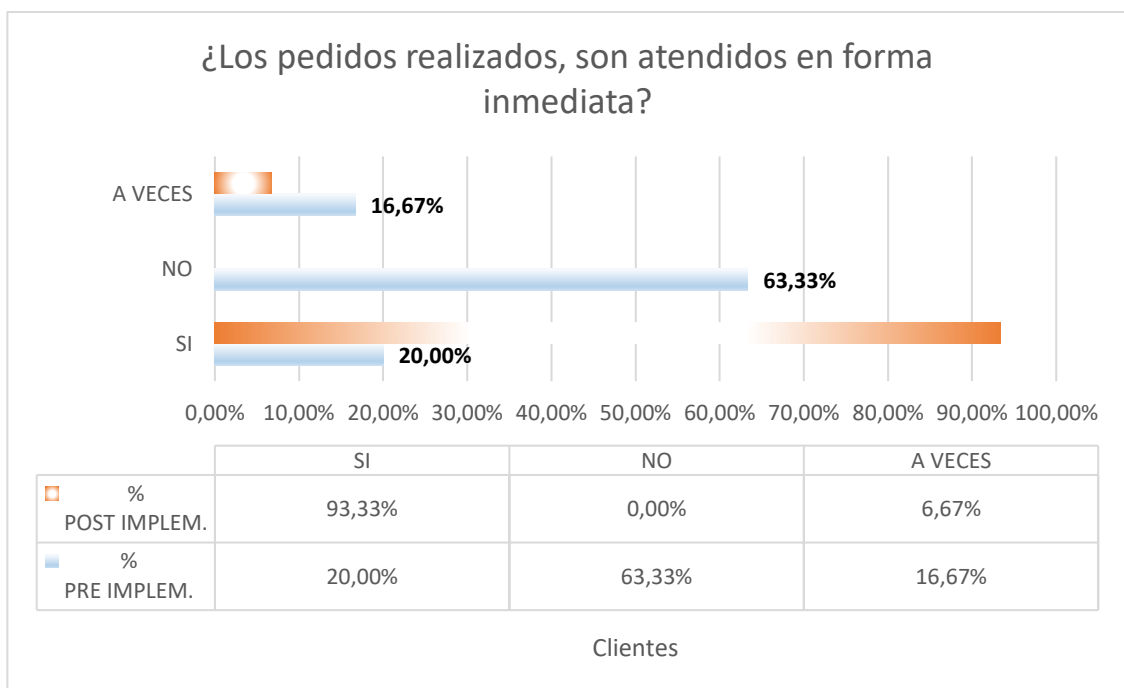


Imagen No. 1 ¿Los Pedidos realizados son atendidos en forma inmediata? Resultados

Fuente: Propia Autoría.

Pregunta 2: ¿Recibe los detalles de los productos cuando lo solicita?

ITEM	PRE IMPLEM.	% PRE IMPLEM.	POST IMPLEM.	% POST IMPLEM.
SI	12	40.00%	27	90.00%
NO	9	30.00%	1	3.33%
A VECES	9	30.00%	2	6.67%
TOTAL	30	100.00%	30	100.00%

Tabla No. 2 ¿Recibe los detalles de los productos cuando lo solicita?Resultados

Fuente: Propia Autoría.

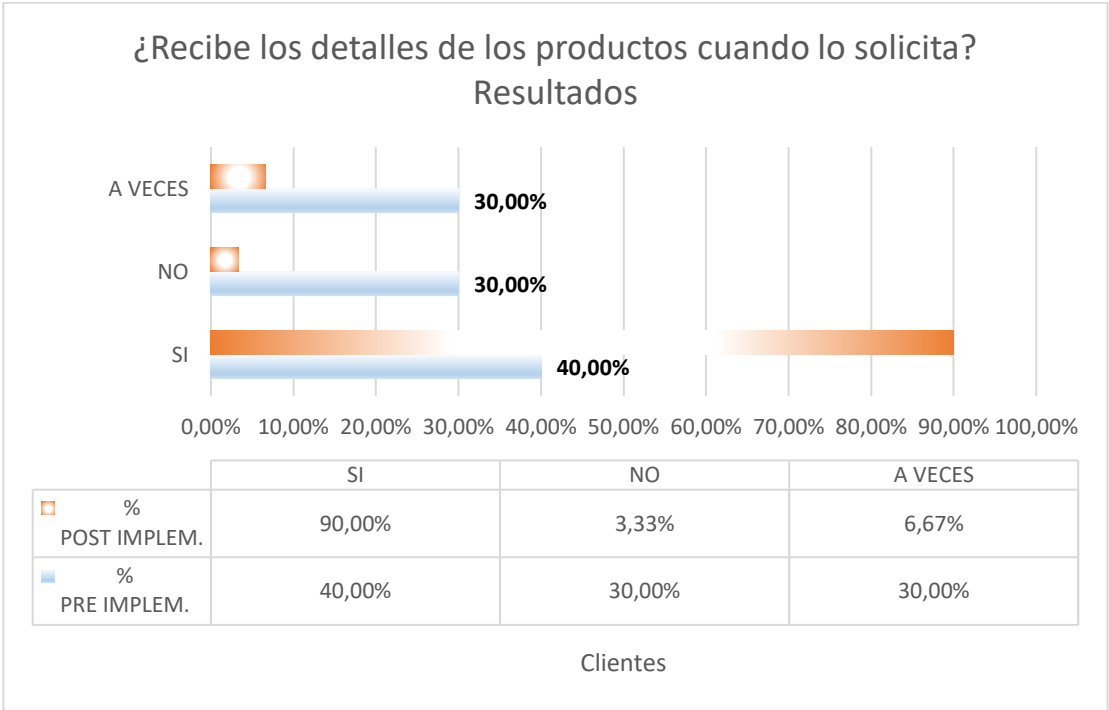


Imagen No. 2 ¿Recibe los detalles de los productos cuando lo solicita? Resultados

Fuente: Propia Autoría.

Pregunta 3: ¿Tuvo inconvenientes respecto a su pago total en caja?

ITEM	CANTIDAD PREIMPLEM.	% PREIMPLEM.	CANTIDAD POSTIMPLEM.	% POSTIMPLEM.
SI	12	40.00%	1	3.33%
NO	11	36.67%	28	93.33%
A VECES	7	23.33%	1	3.33%
TOTAL	30	100.00%	30	100.00%

Tabla No. 3 ¿Tuvo inconvenientes respecto a su pago total en caja? Resultados

Fuente: Propia Autoría.

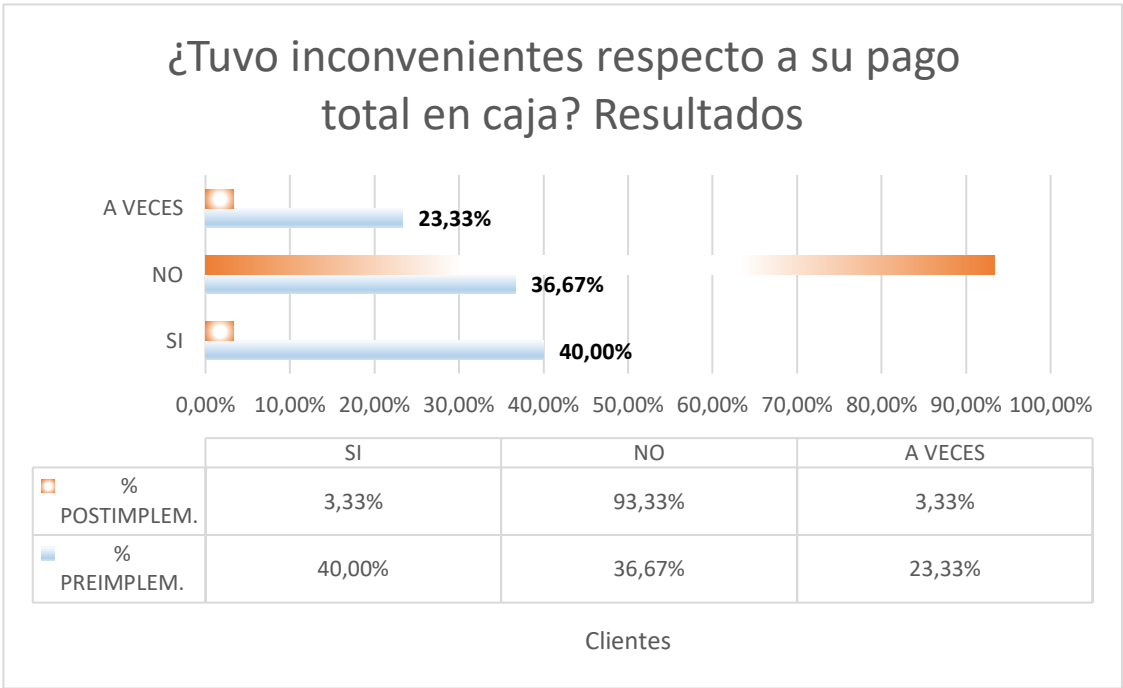


Imagen No. 3 ¿Tuvo inconvenientes respecto a su pago total en caja? Resultados

Fuente: Propia Autoría.

Pregunta 4: ¿Los datos de la comanda contenía exactamente los productos con sus respectivos precios?

ITEM	CANTIDAD PREIMPLEM.	% PREIMPLEM.	CANTIDAD POSTIMPLEM.	% POSTIMPLEM.
SI	14	46.67%	27	90.00%
NO	10	33.33%	1	3.33%
CASI SIEMPRE	6	20.00%	2	6.67%
TOTAL	30	100.00%	30	100.00%

Tabla No. 4 ¿Los datos de la comanda contenía exactamente los productos con sus respectivos precios? Resultados

Fuente: Propia Autoría.

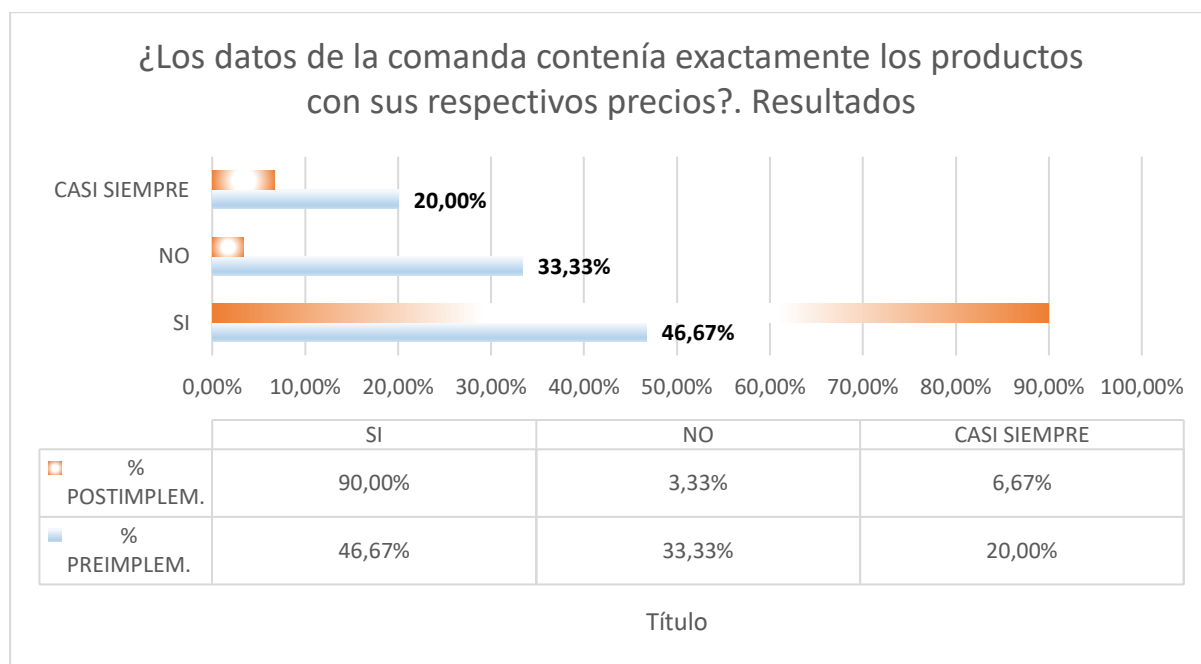


Imagen No. 4 ¿Los datos de la comanda contenía exactamente los productos con sus respectivos precios? Resultados

Fuente: Propia Autoría.

Pregunta 5: ¿Al momento de cancelar ud recibió el comprobante de pago correspondiente?

ITEM	CANTIDAD PREIMPLEM.	% PREIMPLEM.	CANTIDAD POSTIMPLEM.	% POSTIMPLEM.
SI	22	73.33%	30	100.00%
NO	8	26.67%	0	0.00%
TOTAL	30	100.00%	30	100.00%

Tabla No. 5 ¿Al momento de cancelar ud recibió el comprobante de pago correspondiente? Resultados

Fuente: Propia Autoría.

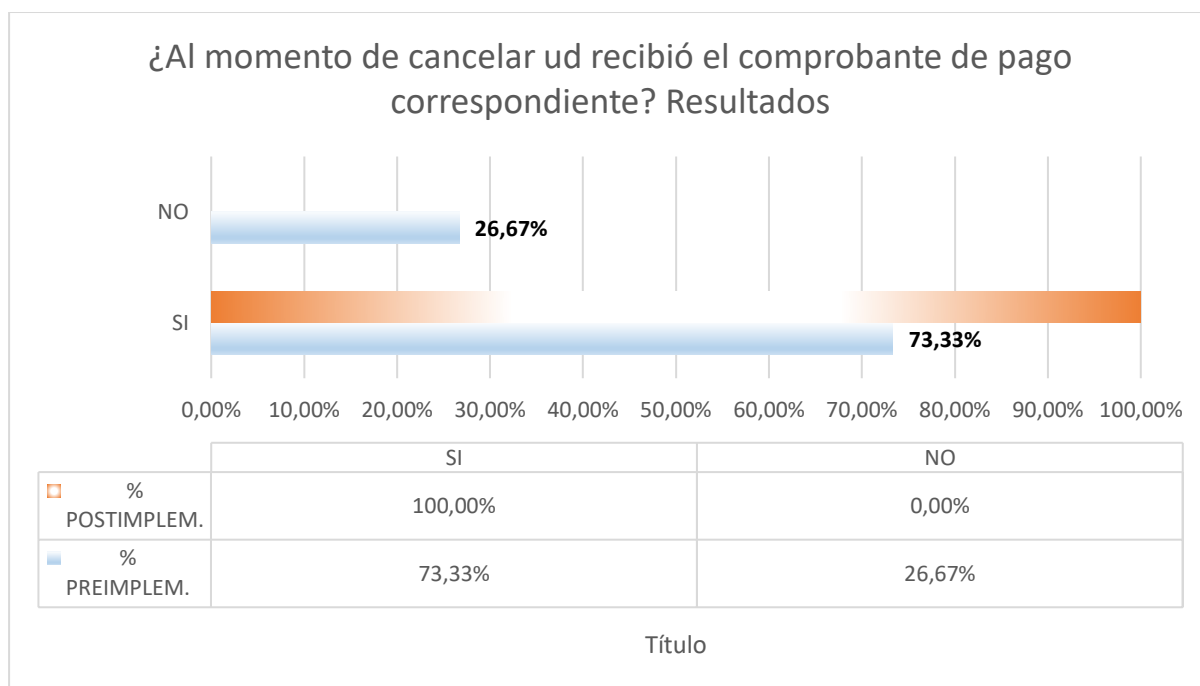


Tabla No. 6 ¿Al momento de cancelar ud recibió el comprobante de pago correspondiente? Resultados

Fuente: Propia Autoría.

Capítulo V: DISCUSIÓN

CAPITULO V: DISCUSIÓN

Los resultados obtenidos han sido muy significativos y valederos para la propietaria y administradora de este negocio.

El uso de los dispositivos como las tablets en la atención de los clientes permite una atención mucho mas precisa respecto a la información de las características, precios y disponibilidad de los productos.

La aplicación brindó los resultados esperados respecto a la atención de pedidos, cobranzas de los pedidos, comanda de los productos solicitados, facilidad de cambio de mesa por algún motivo sin originar incomodidad en el cambio, así como la facilidad para el pago de lo consumido en cualquiera de los tipos y escenarios existentes.

En el mismo análisis de sistemas se determina los requerimientos funcionales que debe implementar en la aplicación en concordancia con las generalidades indicadas en el capítulo de la propuesta tecnológica. Con esto se logra conocer el stock de los productos terminados y determinar las salidas y entradas del almacén.

En el diseño, se debe diseñar y construir las tablas, relaciones y restricciones para poder brindar los datos que se requieran para los reportes, así como para las mismas transacciones y así se cumple con las expectativas respecto a la actualización y protección de datos junto con la manipulación del mismo.

Lo que se solicita es lo que se cobra y lo que se solicita es lo que existe. Por lo tanto, nuestra base de datos ofrece siempre la integridad y consistencia de datos.

Capítulo VI:

CONCLUSIONES

CAPÍTULO VI: CONCLUSIONES

1. Se determinó mediante el uso de las especificaciones de la herramienta visual UML los requerimientos funcionales en el modelo del negocio y de objetos.
2. Los Frameworks permitió aprovechar los servicios Api en la aplicación que se implementó.
3. El postgresql cumplió con las expectativas respecto a un alto performance en la gestión de una base de datos como almacenamiento, acceso, seguridad y respaldo de datos.
4. Los servicios que ofrece Windows en integración con .net a través de una Api Rest, permitió brindar mayor seguridad a la aplicación y a los datos que utiliza.

Capítulo VII

RECOMENDACIONES

CAPÍTULO VII: RECOMENDACIONES

1. Velar por el fiel cumplimiento de las especificaciones funcionales brindadas por el modelado UML para el buen funcionamiento del sistema.
2. El sistema operativo debe siempre estar actualizado para que exista compatibilidad en la funcionalidad de las APIs.
3. El administrador del gestor de la base de datos debe siempre validar el rendimiento de la base de datos.
4. Se debe actualizar, según lo requiera, las funciones propias del .net framework y del Api Rest.

Capítulo VIII

PROPUESTA TECNOLÓGICA

CAPÍTULO VIII: PROPUESTA TECNOLÓGICA

8.1. Generalidades.

El sistema propuesto busca mejorar y atender en el menor tiempo posible a los clientes que visitan a la cevichería. Para ello propongo una atención mas personalizada respecto a la atención de pedidos, así como la simplificación y sencillez en la forma de pago.

Respecto a una atención mas personalizada respecto a la atención de pedidos me refiero que el cliente será atendido en la misma mesa con una Tablet para verificar los platos y productos terminados existentes, así como los precios actualizados.

Cuando me refiero a la simplificación y sencillez en la forma de pago no referimos a la facilidad del pago por mesa, mesas, persona o personas.

El sistema informático comprende los siguientes módulos:

A. Gestión de clientes.

Los clientes a registrar son naturales o jurídicos. El primero se identifica por DNI y el segundo por RUC.

B. Gestión de platos, insumos y productos terminados.

Constituido por todo lo que se compra para el almacén o se vende al cliente. Se personaliza en esta opción el plato a presentar como imagen, detalles, precios, etc.

C. Gestión de mesas y áreas del local.

Permite que el sistema organice de mejor forma las mesas por el número de pisos o áreas.

D. Gestión de ventas.

Se Visualiza todas las mesas por áreas y por cada mesa se visualiza sus estados. Cada estado se manifiesta por sus respectivos colores. Los estados existentes son lo siguientes: Disponible, Ocupado sin pre cuenta, ocupado con pre cuenta, en limpieza.

También se visualiza por cada mesa ocupada su comanda y por cada comanda se puede agregar productos o platos adicionales. Por último, permite el cambio de mesa, trasladando su cuenta de una mesa a otra.

E. Cobranzas.

El cliente puede pagar la cuenta de una sola mesa, de varias mesas o dividir la cuenta de la mesa entre varias personas. Si el cliente no está registrado en la base de datos el mismo sistema lo busca en la Reniec (cliente natural) o en la Sunat (cliente jurídico)

F. Almacén.

Registra todos los movimientos de ingresos y egresos del almacén como la compra de insumos o productos terminados, salidas por uso de insumos o por venta.

G. Reportes

Permite obtener los reportes primordiales para la gestión de la gerencia como es el reporte de ganancias por semana o por mes, de existencias, de ventas, de compras, etc.

8.2. Análisis.

8.2.1. Modelo del Negocio.

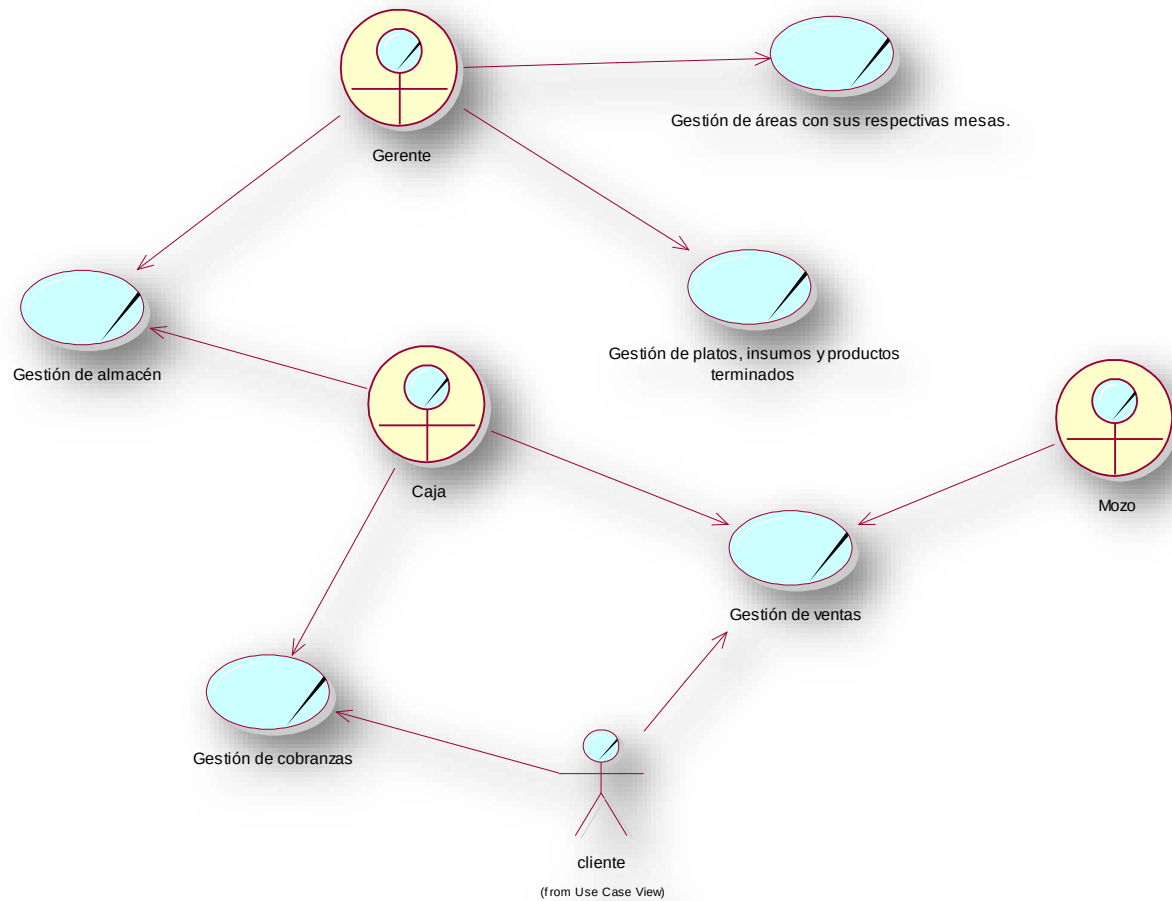


Imagen No. 5 Modelo del negocio.

Fuente: Propia Autoría.

8.2.2. Modelo de casos de uso.

8.2.2.1. Control de Almacén

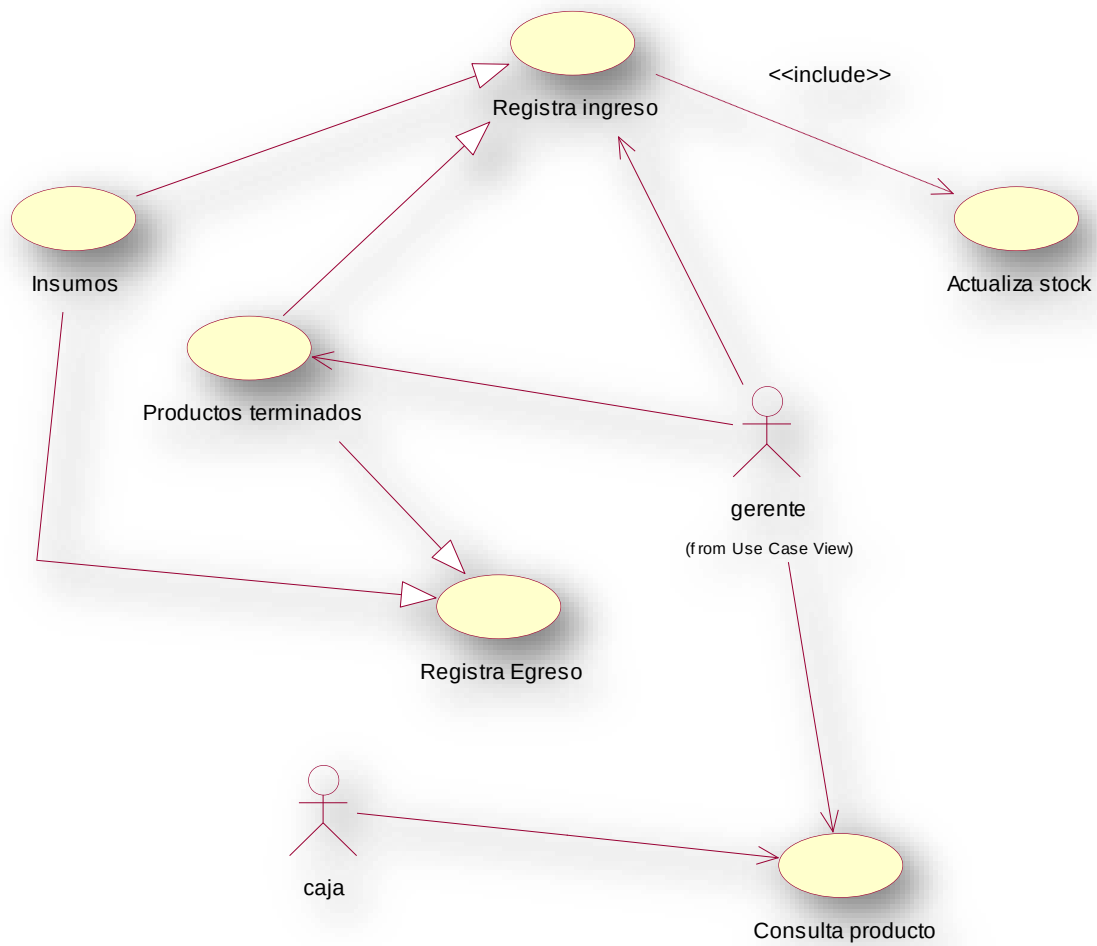


Imagen No. 6 Diagrama de Casos de uso - Almacén

Fuente: Propia Autoría

8.2.2.2. Control de área con sus respectivas mesas.

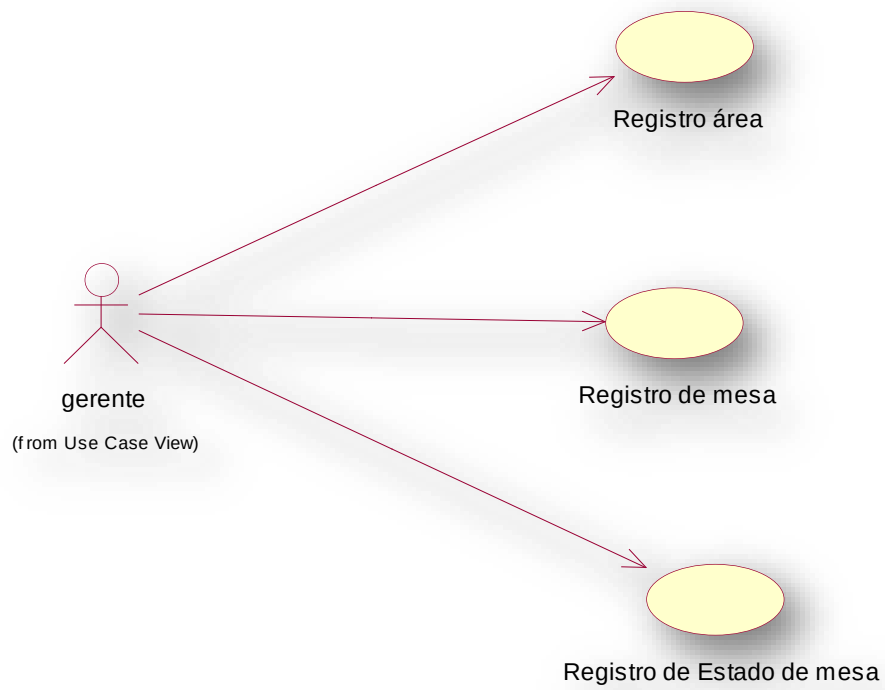


Imagen No. 7 Diagrama de Casos de uso -Control de de área con sus respectivas mesas

Fuente: Propia Autoría

8.2.2.3. Gestión de ventas.

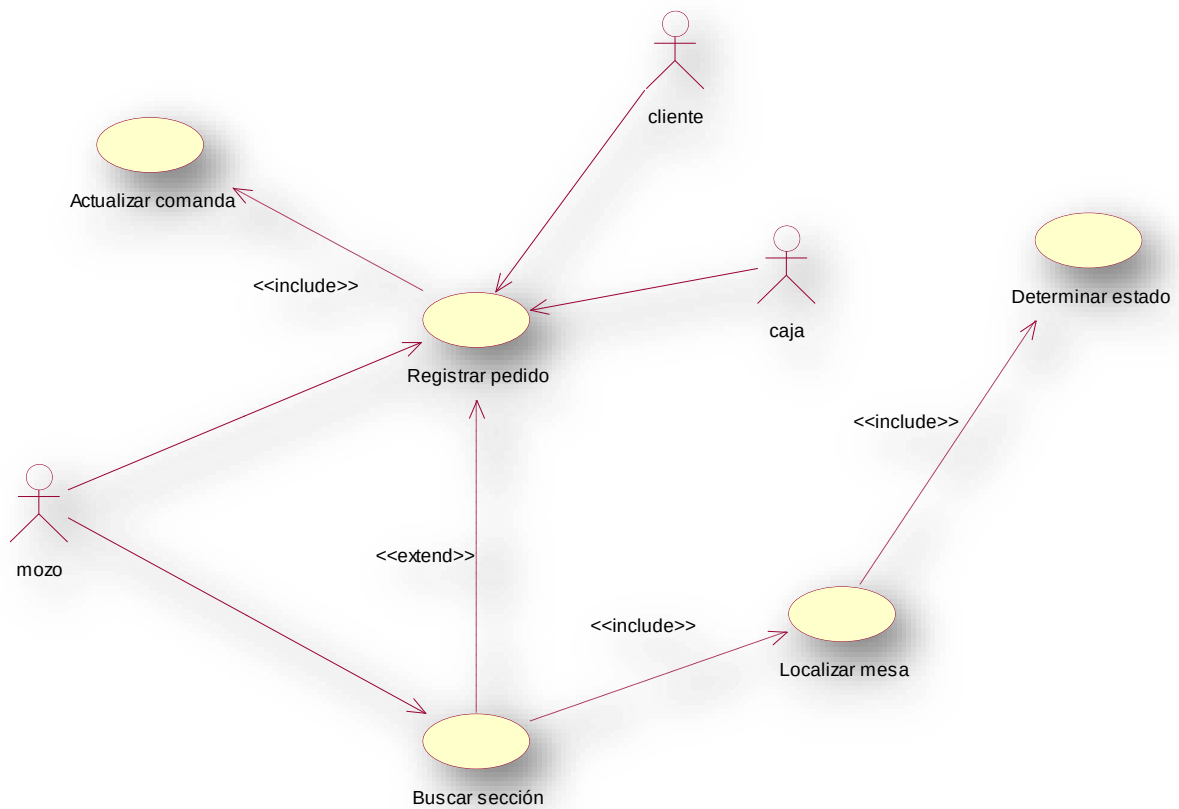


Imagen No. 8 Diagrama de casos de uso – gestión de ventas.

Fuente: Propia Autoría

8.2.2.4. Gestión de cobranzas.

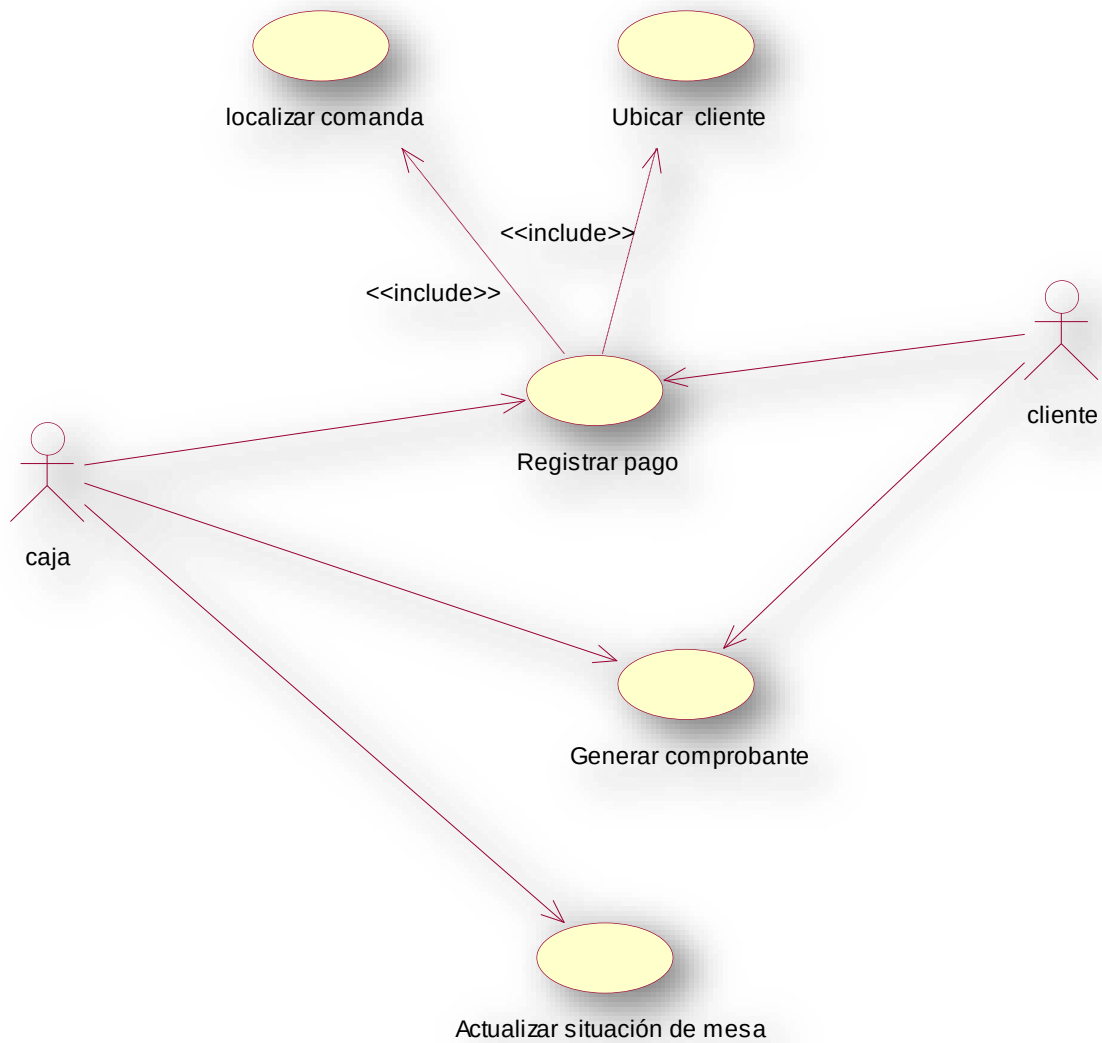


Imagen No. 9 Diagrama de casos de uso - Gestión de cobranzas

Fuente: Propia Autoría

8.3. Diseño

4.3.1. Base de Datos.

A. Modelo Entidad Relación

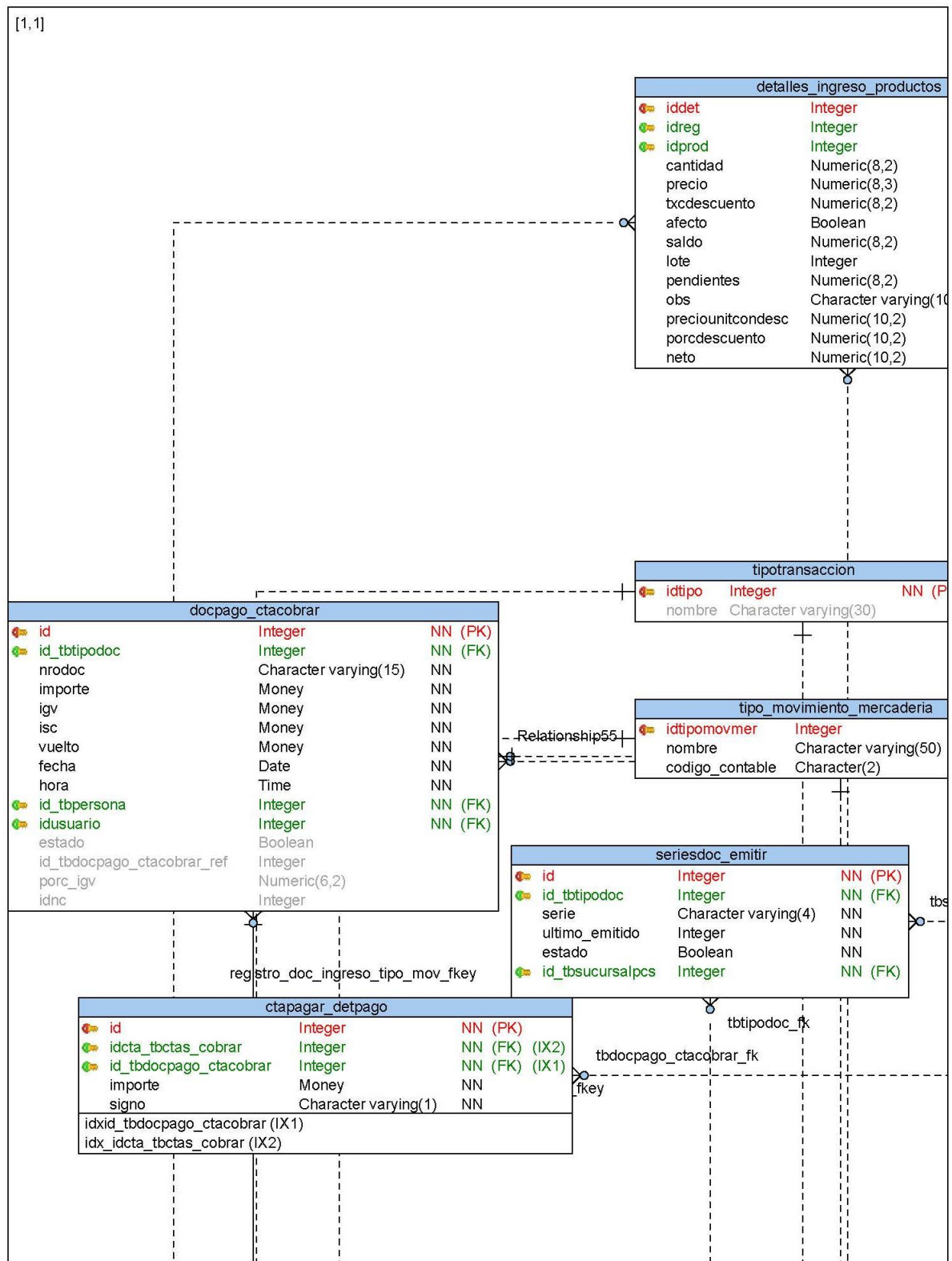


Imagen No. 10 Modelo E-R 1/8

Autor: Propia Autoría.

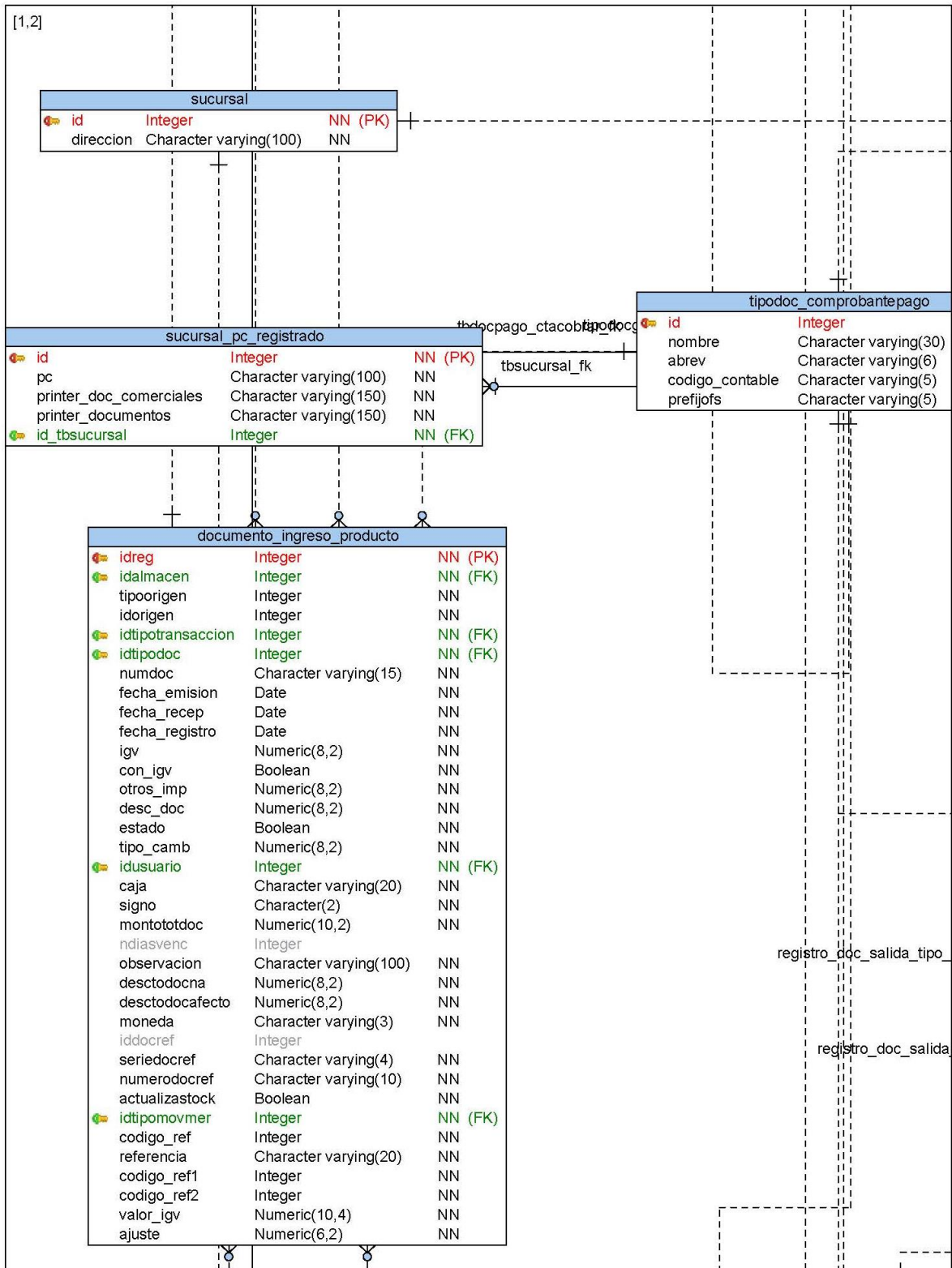


Imagen No. 11 Modelo E-R 2/8

Autor: Propia Autoría.

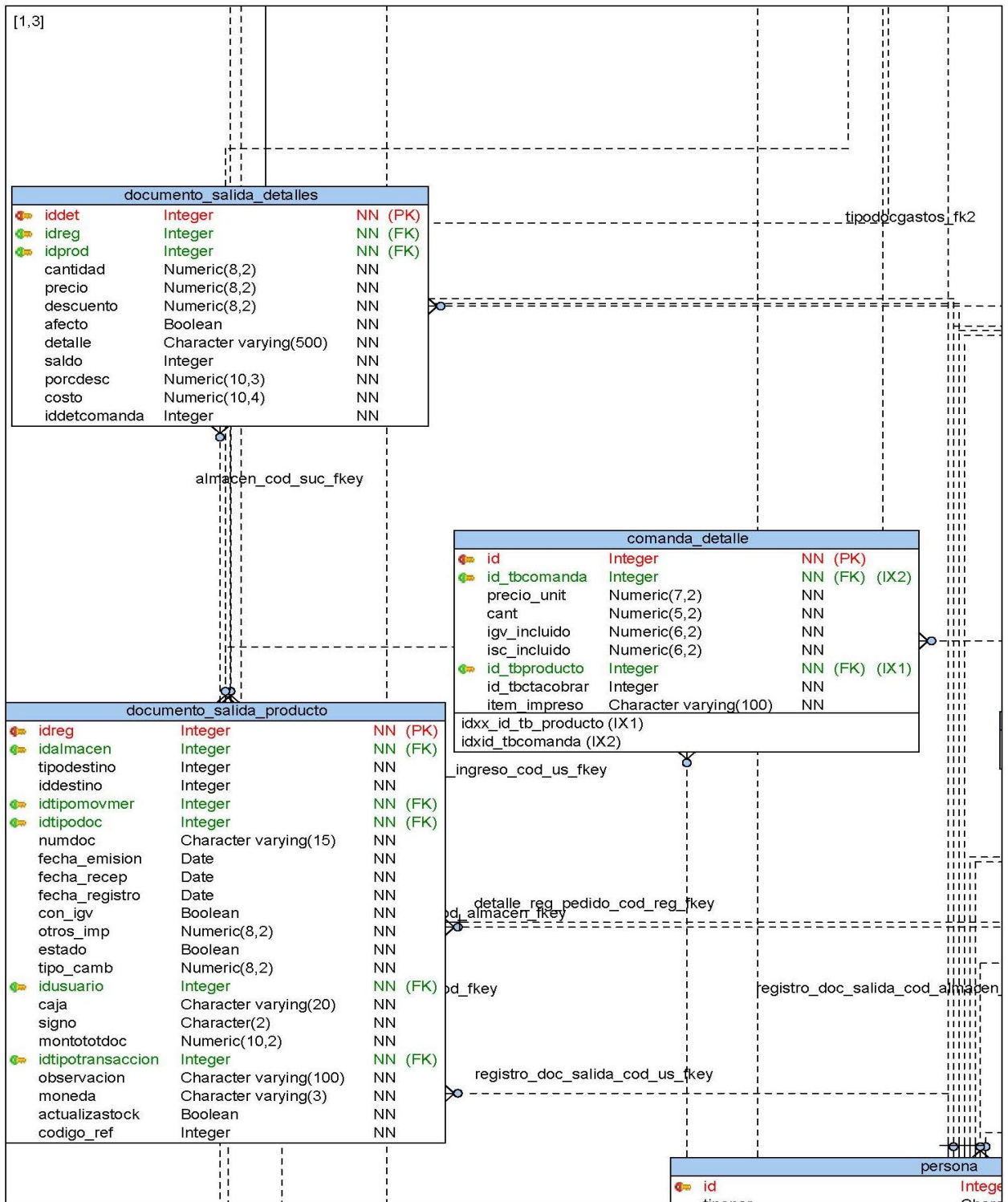


Imagen No. 12 Modelo E-R 3/8

Autor: Propia Autoría.

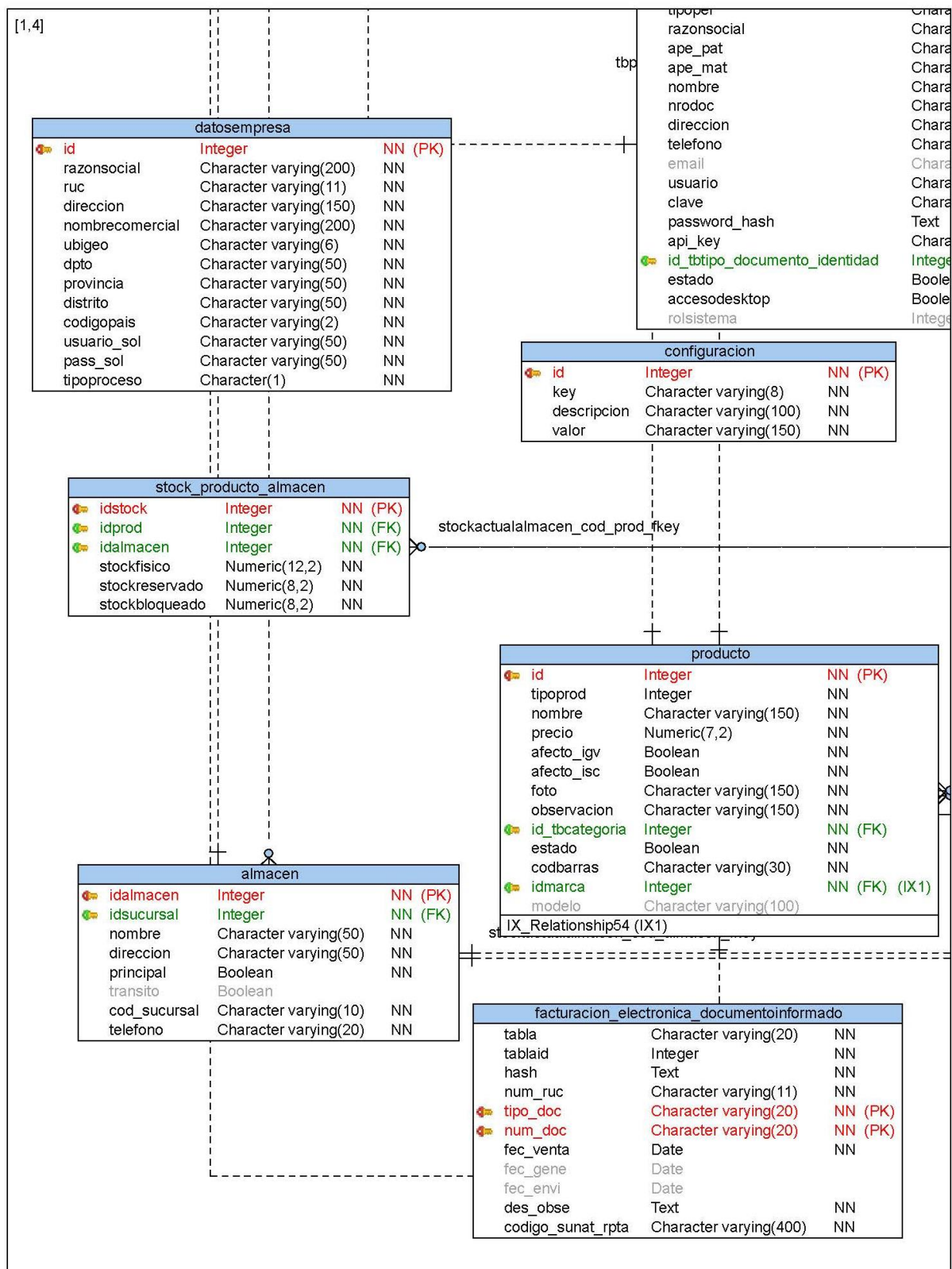


Imagen No. 13 Modelo E-R 4/8

Autor: Propia Autoría.

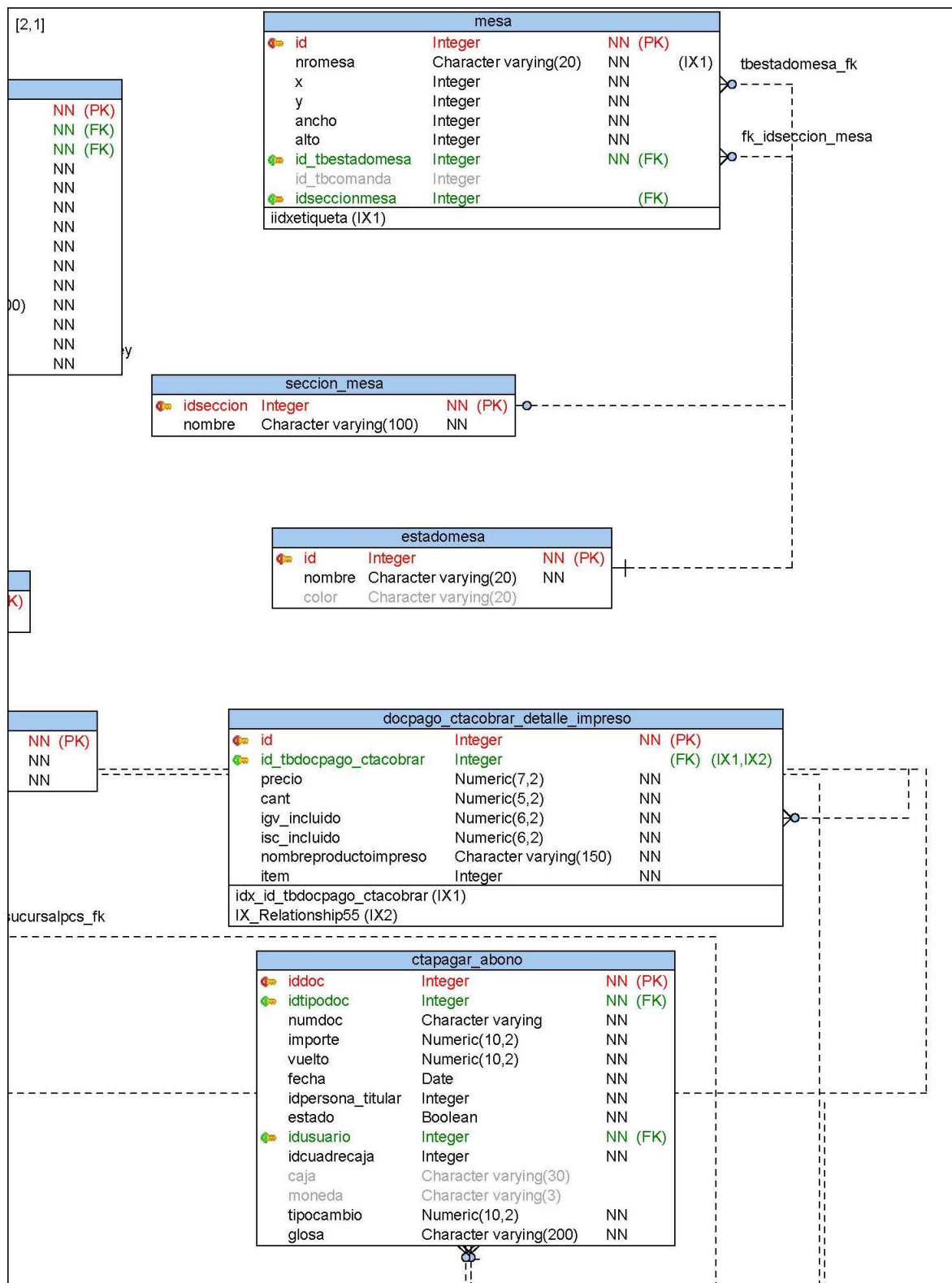


Imagen No. 14 Modelo E-R 5/8

Autor: Propia Autoría

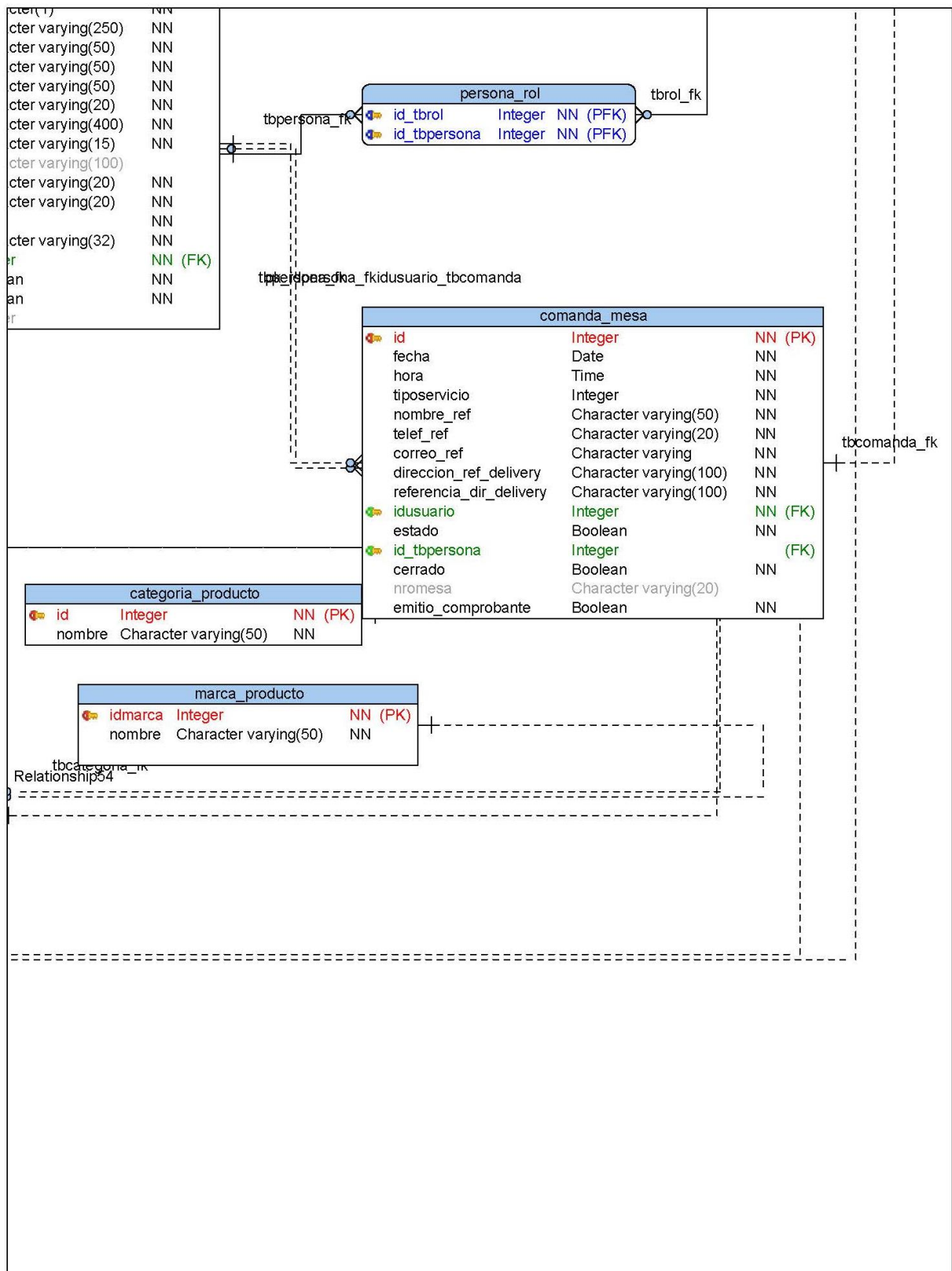


Imagen No. 17 Modelo E-R 8/8

Autor: Propia Autoría

4.3.2. Diccionario de Datos.

Tabla1: ctapagar_abono
<pre>CREATE TABLE "public"."ctapagar_abono" ("iddoc" Integer DEFAULT nextval('tbabono_ctapagar_iddoc_seq'::regclass) NOT NULL, "idtipodoc" Integer NOT NULL, "numdoc" Character varying NOT NULL, "importe" Numeric(10,2) NOT NULL, "vuelto" Numeric(10,2) NOT NULL, "fecha" Date NOT NULL, "idpersona_titular" Integer NOT NULL, "estado" Boolean NOT NULL, "idusuario" Integer NOT NULL, "idcuadrecaja" Integer NOT NULL, "caja" Character varying(30), "moneda" Character varying(3), "tipocambio" Numeric(10,2) DEFAULT 0 NOT NULL, "glosa" Character varying(200) DEFAULT ''::character varying NOT NULL) WITH (autovacuum_enabled=true); ALTER TABLE "public"."ctapagar_abono" ADD CONSTRAINT "tbdocpago_ctapagar_pkey" PRIMARY KEY ("iddoc");</pre>

Tabla 2 Tabla: ctapagar_abono

Autor: Propia Autoría

Tabla 2: almacen

```
CREATE TABLE "public"."almacen"
(
  "idalmacen" Integer DEFAULT
nextval('tbalmacen_idalmacen_seq'::regclass) NOT NULL,
  "idsucursal" Integer NOT NULL,
  "nombre" Character varying(50) NOT NULL,
  "direccion" Character varying(50) NOT NULL,
  "principal" Boolean NOT NULL,
  "transito" Boolean,
  "cod_sucursal" Character varying(10) DEFAULT "::character varying NOT
NULL,
  "telefono" Character varying(20) DEFAULT "::character varying NOT
NULL
)
WITH (
  autovacuum_enabled=true);
ALTER TABLE "public"."almacen" ADD CONSTRAINT
"tbalmacen_pkey" PRIMARY KEY ("idalmacen");
```

Tabla 3 Tabla: almacen

Autor: Propia Autoría

Tabla 3: categoria_producto

```
CREATE TABLE "public"."categoria_producto"
(
  "id" Integer DEFAULT nextval('tbcategoria_id_seq'::regclass) NOT
NULL,
  "nombre" Character varying(50) NOT NULL
)
WITH (
  autovacuum_enabled=true);

ALTER TABLE "public"."categoria_producto" ADD CONSTRAINT
"tbcategoria_pk" PRIMARY KEY ("id");
```

Tabla 4 Tabla 3: categoria_producto

Autor: Propia Autoría

Tabla 4: comanda_mesa

```
CREATE TABLE "public"."comanda_mesa"
(
  "id" Integer DEFAULT nextval('tbcomanda_id_seq'::regclass) NOT
  NULL,
  "fecha" Date NOT NULL,
  "hora" Time NOT NULL,
  "tiposervicio" Integer NOT NULL,
  "nombre_ref" Character varying(50) DEFAULT "::character varying NOT
  NULL,
  "telef_ref" Character varying(20) DEFAULT "::character varying NOT
  NULL,
  "correo_ref" Character varying DEFAULT "::character varying NOT
  NULL,
  "direccion_ref_delivery" Character varying(100) DEFAULT "::character
  varying NOT NULL,
  "referencia_dir_delivery" Character varying(100) DEFAULT "::character
  varying NOT NULL,
  "idusuario" Integer NOT NULL,
  "estado" Boolean DEFAULT true NOT NULL,
  "id_tbpersona" Integer,
  "cerrado" Boolean DEFAULT false NOT NULL,
  "nromesa" Character varying(20),
  "emitio_comprobante" Boolean DEFAULT false NOT NULL
)
WITH (
  autovacuum_enabled=true);
COMMENT ON COLUMN "public"."comanda_mesa"."tiposervicio" IS '1
= Mesa
2 = Para llevar
3 = Delivery';

ALTER TABLE "public"."comanda_mesa" ADD CONSTRAINT
"tbcomanda_pk" PRIMARY KEY ("id");
```

Tabla 5 Tabla 4: comanda_mesa

Autor: Propia Autoría

Tabla 5: configuracion

```
CREATE TABLE "public"."configuracion"
(
  "id" Integer DEFAULT nextval('tbconfiguracion_id_seq'::regclass) NOT
  NULL,
  "key" Character varying(8) DEFAULT "::character varying NOT NULL,
  "descripcion" Character varying(100) DEFAULT "::character varying NOT
  NULL,
  "valor" Character varying(150) DEFAULT "::character varying NOT
  NULL
)
WITH (
  autovacuum_enabled=true);

ALTER TABLE "public"."configuracion" ADD CONSTRAINT
"tbconfiguracion_pk" PRIMARY KEY ("id");
```

Tabla 6 Tabla 5: configuración

Autor: Propia Autoría

Tabla 6: cuadro_caja

```
CREATE TABLE "public"."cuadro_caja"
(
  "id" Integer DEFAULT nextval('tbcuadro_caja_id_seq'::regclass) NOT
  NULL,
  "Fechainicio" Date NOT NULL,
  "horainicio" Time NOT NULL,
  "fechatermino" Date,
  "horatermino" Time,
  "dineroefectivo" Money DEFAULT 0.00 NOT NULL,
  "id_tbpersona" Integer DEFAULT nextval('tbpersona_id_seq'::regclass),
  "estado" Boolean DEFAULT true NOT NULL
)
WITH (
  autovacuum_enabled=true);

CREATE INDEX "IX_Relationship53" ON "public"."cuadro_caja"
("id_tbpersona");

ALTER TABLE "public"."cuadro_caja" ADD CONSTRAINT
"tbcuadro_caja_pk" PRIMARY KEY ("id");
```

Tabla 7 Tabla 6: cuadro_caja

Autor: Propia Autoría

Tabla 7: ctas_cobrar

```
CREATE TABLE "public"."ctas_cobrar"
(
  "idcta" Integer DEFAULT nextval('tbctas_cobrar_idcta_seq'::regclass)
  NOT NULL,
  "tiposervicio" Integer DEFAULT 1 NOT NULL,
  "id_tbtipodoc" Integer NOT NULL,
  "numdoc" Character varying(15) DEFAULT ">::character varying NOT
  NULL,
  "fecha" Date NOT NULL,
  "fvenc" Date,
  "nroletra" Integer DEFAULT 0 NOT NULL,
  "importe" Money DEFAULT 0 NOT NULL,
  "mora" Money DEFAULT 0.00 NOT NULL,
  "saldo" Money DEFAULT 0 NOT NULL,
  "igv" Money DEFAULT 0 NOT NULL,
  "isc" Money DEFAULT 0 NOT NULL,
  "idusuario" Integer NOT NULL,
  "estado" Boolean DEFAULT true NOT NULL,
  "cerrado" Boolean DEFAULT false NOT NULL,
  "incluir_reg_ventas" Boolean DEFAULT false NOT NULL,
  "idempresa" Integer,
  "idreferencia_origencta" Integer DEFAULT 0 NOT NULL,
  "glosa" Character varying(300) DEFAULT ">::character varying NOT
  NULL,
  "id_tbpersona" Integer NOT NULL,
  "formapago" Integer DEFAULT 2 NOT NULL
)
WITH (
  autovacuum_enabled=true);
COMMENT ON COLUMN "public"."ctas_cobrar"."tiposervicio" IS '1 =
Servicio venta comida. Comanda.
2 = Otros servicios. Para hoteles: Lavanderia, Cochera, ';
COMMENT ON COLUMN "public"."ctas_cobrar"."formapago" IS '1 =
Contado
2 = Credito;

ALTER TABLE "public"."ctas_cobrar" ADD CONSTRAINT
"tbctas_cobrar_pk" PRIMARY KEY ("idcta");

ALTER TABLE "public"."ctas_cobrar" ADD CONSTRAINT
"idx_unico_documento" UNIQUE
("tiposervicio","id_tbtipodoc","numdoc");
```

Tabla 8 Tabla 7: ctas_cobrar

Autor: Propia Autoría

Tabla 8: cta_pagar

```
CREATE TABLE "public"."cta_pagar"
(
  "idcta" Integer DEFAULT nextval('tbctas_pagar_idcta_seq'::regclass) NOT
  NULL,
  "idtipodoc" Integer NOT NULL,
  "numdoc" Character varying(15),
  "fecha" Date NOT NULL,
  "fvenc" Date NOT NULL,
  "nroletra" Integer,
  "idpersona" Integer NOT NULL,
  "moneda" Character varying(6) NOT NULL,
  "importe" Numeric(10,2) DEFAULT 0 NOT NULL,
  "saldo" Numeric(10,2) DEFAULT 0 NOT NULL,
  "igv" Numeric(10,2) DEFAULT 0 NOT NULL,
  "cambio" Numeric(10,2) DEFAULT 0 NOT NULL,
  "idalmacen" Integer,
  "idusuario" Integer NOT NULL,
  "caja" Character varying(20),
  "cerrado" Boolean NOT NULL,
  "estado" Boolean NOT NULL,
  "incluir_reg_compras" Boolean NOT NULL,
  "afectacaja" Boolean NOT NULL,
  "referencia" Character varying(5),
  "idref1" Integer,
  "idcuadrecaja" Integer,
  "idtiposervicio" Integer NOT NULL,
  "emitido" Boolean NOT NULL,
  "glosa" Character varying(300),
  "mgrabado" Numeric(10,2) DEFAULT 0 NOT NULL,
  "mnograbado" Numeric(10,2) DEFAULT 0 NOT NULL,
  "montorh" Numeric(10,2) DEFAULT 0 NOT NULL,
  "misc" Numeric(10,2) DEFAULT 0 NOT NULL,
  "otros" Numeric(10,2) DEFAULT 0 NOT NULL,
  "rta4cat" Numeric(10,2) DEFAULT 0 NOT NULL,
  "idtipotransaccion" Integer NOT NULL
  WITH ( autovacuum_enabled=true);
COMMENT ON COLUMN "public"."cta_pagar"."referencia" IS
'C=Compra';
COMMENT ON COLUMN "public"."cta_pagar"."idref1" IS 'ID del
Documento Compra';
ALTER TABLE "public"."cta_pagar" ADD CONSTRAINT
"tbctas_pagar_pkey" PRIMARY KEY ("idcta");
```

Tabla 9 Tabla 8: cta_pagar

Autor: Propia Autoría

Tabla 9: datosempresa

```
CREATE TABLE "public"."datosempresa"
(
  "id" Integer DEFAULT nextval('tbdatosempresa_id_seq'::regclass) NOT
  NULL,
  "razonsocial" Character varying(200) NOT NULL,
  "ruc" Character varying(11) NOT NULL,
  "direccion" Character varying(150) NOT NULL,
  "nombrecomercial" Character varying(200) NOT NULL,
  "ubigeo" Character varying(6) NOT NULL,
  "dpto" Character varying(50) NOT NULL,
  "provincia" Character varying(50) NOT NULL,
  "distrito" Character varying(50) NOT NULL,
  "codigopais" Character varying(2) NOT NULL,
  "usuario_sol" Character varying(50) NOT NULL,
  "pass_sol" Character varying(50) NOT NULL,
  "tipoproceso" Character(1) NOT NULL
)
WITH (
  autovacuum_enabled=true);

ALTER TABLE "public"."datosempresa" ADD CONSTRAINT
"tbdatosempresa_pkey" PRIMARY KEY ("id");
```

Tabla 10 Tabla 9: datosempresa

Autor: Propia Autoría

Tabla 10: ctapagar_detalle_abono

```
CREATE TABLE "public"."ctapagar_detalle_abono"
( "id" Integer DEFAULT nextval('tbdetabono_ctapagar_id_seq'::regclass)
  NOT NULL,
  "iddoc" Integer NOT NULL,
  "idcta" Integer NOT NULL,
  "importe" Numeric(10,2) DEFAULT 0 NOT NULL,
  "signo" Character(1) NOT NULL,
  "referencia" Character varying(5),
  "idref" Integer)
WITH ( autovacuum_enabled=true);

ALTER TABLE "public"."ctapagar_detalle_abono" ADD CONSTRAINT
"tbdetpago_ctapagar_pkey" PRIMARY KEY ("id");
```

Tabla 11 Tabla 10: ctapagar_detalle_abono

Autor: Propia Autoría

Tabla 11: comanda_detalle

```
CREATE TABLE "public"."comanda_detalle"
(
  "id" Integer DEFAULT nextval('tbdetalle_comanda_id_seq'::regclass)
  NOT NULL,
  "id_tbcomanda" Integer NOT NULL,
  "precio_unit" Numeric(7,2) DEFAULT 0.00 NOT NULL,
  "cant" Numeric(5,2) DEFAULT 0.00 NOT NULL,
  "igv_incluido" Numeric(6,2) DEFAULT 0.00 NOT NULL,
  "isc_incluido" Numeric(6,2) DEFAULT 0.00 NOT NULL,
  "id_tbproducto" Integer NOT NULL,
  "id_tbtacobrar" Integer DEFAULT 0 NOT NULL,
  "item_impreso" Character varying(100) DEFAULT "::character varying"
  NOT NULL
)
WITH (
  autovacuum_enabled=true);
CREATE INDEX "idx_id_tb_producto" ON "public"."comanda_detalle"
("id_tbproducto");
CREATE INDEX "idx_id_tbc_comanda" ON "public"."comanda_detalle"
("id_tbcomanda");
ALTER TABLE "public"."comanda_detalle" ADD CONSTRAINT
"pk_det_comanda" PRIMARY KEY ("id");
```

Tabla 12 Tabla 11: comanda_detalle

Autor: Propia Autoría

Tabla 12: docpago_ctacobrar_detalle_impreso

```

CREATE TABLE "public"."docpago_ctacobrar_detalle_impreso"
( "id" Integer DEFAULT
nextval('tbdetalle_docpago_ctacobrar_impreso_id_seq'::regclass) NOT
NULL,
  "id_tbdocpago_ctacobrar" Integer DEFAULT
nextval('tbdocpago_ctacobrar_id_seq'::regclass),
  "precio" Numeric(7,2) DEFAULT 0.00 NOT NULL,
  "cant" Numeric(5,2) DEFAULT 0.00 NOT NULL,
  "igv_incluido" Numeric(6,2) DEFAULT 0.00 NOT NULL,
  "isc_incluido" Numeric(6,2) DEFAULT 0.00 NOT NULL,
  "nombreproductoimpreso" Character varying(150) DEFAULT "::character
varying NOT NULL,
  "item" Integer DEFAULT 0 NOT NULL)
WITH ( autovacuum_enabled=true);
CREATE INDEX "idx_id_tbdocpago_ctacobrar" ON
"public"."docpago_ctacobrar_detalle_impreso" ("id_tbdocpago_ctacobrar");
CREATE INDEX "IX_Relationship55" ON
"public"."docpago_ctacobrar_detalle_impreso" ("id_tbdocpago_ctacobrar");
ALTER TABLE "public"."docpago_ctacobrar_detalle_impreso" ADD
CONSTRAINT "tbdetalle_docpago_ctacobrar_impreso_pk" PRIMARY
KEY ("id");

```

Tabla 13 Tabla 12: docpago_ctacobrar_detalle_impreso

Autor: Propia Autoría

Tabla 13: detalles_ingreso_productos

```

CREATE TABLE "public"."detalles_ingreso_productos"
(
  "iddet" Integer DEFAULT nextval('tbdetalle_reg_ing_iddet_seq'::regclass)
NOT NULL,
  "idreg" Integer NOT NULL,
  "idprod" Integer NOT NULL,
  "cantidad" Numeric(8,2) NOT NULL,
  "precio" Numeric(8,3) NOT NULL,
  "txcdescuento" Numeric(8,2) NOT NULL,
  "afecto" Boolean NOT NULL,
  "saldo" Numeric(8,2) NOT NULL,
  "lote" Integer NOT NULL,
  "pendientes" Numeric(8,2) DEFAULT 0 NOT NULL,
  "obs" Character varying(100) DEFAULT "::character varying NOT NULL,
  "preciounitcondesc" Numeric(10,2) DEFAULT 0 NOT NULL,
  "porcdescuento" Numeric(10,2) DEFAULT 0 NOT NULL,
  "neto" Numeric(10,2) DEFAULT 0 NOT NULL
)
WITH (
  autovacuum_enabled=true)

```

```

;

ALTER TABLE "public"."detalles_ingreso_productos" ADD
CONSTRAINT "tbdetalle_reg_ing_pkey" PRIMARY KEY ("iddet")
;

CREATE TRIGGER "trtbdetalle_reg_ing"
AFTER DELETE OR INSERT OR UPDATE
ON "public"."detalles_ingreso_productos"
FOR EACH ROW
EXECUTE PROCEDURE "public"."trftbdetalle_reg_ing"()
;

```

Tabla 14 Tabla 13: detalles_ingreso_productos

Autor: Propia Autoría

Tabla 14: documento_salida_detalles

```

CREATE TABLE "public"."documento_salida_detalles"
(
  "iddet" Integer DEFAULT
nextval('tbdetalle_reg_salida_iddet_seq'::regclass) NOT NULL,
  "idreg" Integer NOT NULL,
  "idprod" Integer NOT NULL,
  "cantidad" Numeric(8,2) NOT NULL,
  "precio" Numeric(8,2) NOT NULL,
  "descuento" Numeric(8,2) NOT NULL,
  "afecto" Boolean NOT NULL,
  "detalle" Character varying(500) DEFAULT '::character varying NOT
NULL,
  "saldo" Integer DEFAULT 0 NOT NULL,
  "porcdesc" Numeric(10,3) DEFAULT 0 NOT NULL,
  "costo" Numeric(10,4) DEFAULT 0 NOT NULL,
  "iddetcomanda" Integer DEFAULT 0 NOT NULL
)
WITH (
  autovacuum_enabled=true);
COMMENT ON COLUMN
"public"."documento_salida_detalles"."iddetcomanda" IS 'id del detalle de
la comanda del producto que no sea a la carta. ';
ALTER TABLE "public"."documento_salida_detalles" ADD
CONSTRAINT "tbdetalle_reg_salida_pkey" PRIMARY KEY ("iddet");

```

Tabla 15 Tabla 14: documento_salida_detalles

Autor: Propia Autoría

Tabla 15: ctapagar_detpago

```

CREATE TABLE "public"."ctapagar_detpago"
(
  "id" Integer DEFAULT nextval('tbdetpago_ctapagar_id_seq'::regclass)
  NOT NULL,
  "idcta_tbctas_cobrar" Integer NOT NULL,
  "id_tbdocpago_ctacobrar" Integer NOT NULL,
  "importe" Money DEFAULT 0.00 NOT NULL,
  "signo" Character varying(1) DEFAULT '+'::character varying NOT NULL
)
WITH (
  autovacuum_enabled=true);
CREATE INDEX "idxid_tbdocpago_ctacobrar" ON
"public"."ctapagar_detpago" ("id_tbdocpago_ctacobrar");
CREATE INDEX "idx_idcta_tbctas_cobrar" ON
"public"."ctapagar_detpago" ("idcta_tbctas_cobrar");
ALTER TABLE "public"."ctapagar_detpago" ADD CONSTRAINT
"tbdetpago_ctapagar_pk" PRIMARY KEY ("id");

```

Tabla 16 Tabla 15: ctapagar_detpago

Autor: Propia Autoría

Tabla 16: docpago_ctacobrar

```

CREATE TABLE "public"."docpago_ctacobrar "
( "id" Integer DEFAULT nextval('tbdocpago_ctacobrar_id_seq'::regclass)
  NOT NULL,
  "id_tbtipodoc" Integer NOT NULL,
  "nrodoc" Character varying(15) NOT NULL,
  "importe" Money DEFAULT 0.00 NOT NULL,
  "igv" Money DEFAULT 0.00 NOT NULL,
  "isc" Money DEFAULT 0.00 NOT NULL,
  "vuelto" Money DEFAULT 0.00 NOT NULL,
  "fecha" Date NOT NULL,
  "hora" Time NOT NULL,
  "id_tbpersona" Integer NOT NULL,
  "idusuario" Integer NOT NULL,
  "estado" Boolean DEFAULT true,
  "id_tbdocpago_ctacobrar_ref" Integer,
  "porc_igv" Numeric(6,2) DEFAULT 18,
  "idnc" Integer)
WITH ( autovacuum_enabled=true);
ALTER TABLE "public"."docpago_ctacobrar" ADD CONSTRAINT
"tbdocpago_ctacobrar_pk" PRIMARY KEY ("id");

```

Tabla 17 Tabla 16: docpago_ctacobrar

Autor: Propia Autoría

Tabla 17: estadomesa

```

CREATE TABLE "public"."estadomesa"
(
  "id" Integer DEFAULT nextval('tbestadomesa_id_seq'::regclass) NOT
  NULL,
  "nombre" Character varying(20) NOT NULL,
  "color" Character varying(20) DEFAULT "::character varying
)
WITH (
  autovacuum_enabled=true);
COMMENT ON TABLE "public"."estadomesa" IS '1 = Disponible
2 = Ocupado / Sin Orden Comanda
3 = Ocupado / Con Comanda.
4. = Reservado';
ALTER TABLE "public"."estadomesa" ADD CONSTRAINT
"tbestadomesa_pk" PRIMARY KEY ("id");

```

Tabla 18 Tabla 17: estadomesa

Autor: Propia Autoría

Tabla 18: facturacion_electronica_documentoinformado

```

CREATE TABLE "public"."facturacion_electronica_documentoinformado"
(
  "tabla" Character varying(20) DEFAULT "::character varying NOT NULL,
  "tablaid" Integer NOT NULL,
  "hash" Text DEFAULT "::text NOT NULL,
  "num_ruc" Character varying(11) DEFAULT "::character varying NOT
  NULL,
  "tipo_doc" Character varying(20) DEFAULT "::character varying NOT
  NULL,
  "num_doc" Character varying(20) DEFAULT "::character varying NOT
  NULL,
  "fec_venta" Date NOT NULL,
  "fec_gene" Date,
  "fec_envi" Date,
  "des_obse" Text DEFAULT "::character varying NOT NULL,
  "codigo_sunat_rpta" Character varying(400) DEFAULT "::character
  varying NOT NULL)
WITH (
  autovacuum_enabled=true);
ALTER TABLE "public"."facturacion_electronica_documentoinformado"
ADD CONSTRAINT "pk_fact" PRIMARY KEY ("tipo_doc","num_doc");

```

Tabla 19 Tabla 18: facturacion_electronica_documentoinformado

Autor: Propia Autoría

Tabla 19: marca_producto

```
CREATE TABLE "public"."marca_producto"
(
  "idmarca" Integer DEFAULT nextval('tbmarca_idmarca_seq'::regclass)
  NOT NULL,
  "nombre" Character varying(50) NOT NULL
)
WITH (
  autovacuum_enabled=true);

ALTER TABLE "public"."marca_producto" ADD CONSTRAINT
"marca_pkey" PRIMARY KEY ("idmarca");
```

Tabla 20 Tabla 19: marca_producto

Autor: Propia Autoría

Tabla 20: medio_pago

```
CREATE TABLE "public"."medio_pago"
(
  "id" Integer DEFAULT nextval('tbmediopago_id_seq'::regclass) NOT
  NULL,
  "nombre" Character varying(50) NOT NULL
)
WITH (
  autovacuum_enabled=true);
ALTER TABLE "public"."medio_pago" ADD CONSTRAINT
"tbmediopago_pk" PRIMARY KEY ("id");
```

Tabla 21 Tabla 20: medio_pago

Autor: Propia Autoría

Tabla 21: docpago_ctacobrar_mediopago

```
CREATE TABLE "public"."docpago_ctacobrar_mediopago"
(
  "id_tbmediopago" Integer NOT NULL,
  "id_tbdocpago_ctacobrar" Integer NOT NULL,
  "importe" Money DEFAULT 0.00 NOT NULL,
  "observacion" Character varying(100) DEFAULT "::character varying
NOT NULL
)
WITH (
  autovacuum_enabled=true);
ALTER TABLE "public"."docpago_ctacobrar_mediopago" ADD
CONSTRAINT "tbmediopago_tbdocpago_ctacobrar_pk" PRIMARY KEY
("id_tbmediopago","id_tbdocpago_ctacobrar");
```

Tabla 22 Tabla 21: docpago_ctacobrar_mediopago

Autor: Propia Autoría

Tabla 22: rol

```
CREATE TABLE "public"."rol"
(
  "id" Integer DEFAULT nextval('tbrol_id_seq'::regclass) NOT NULL,
  "nombre" Character varying(50) NOT NULL
)
WITH (
  autovacuum_enabled=true);
ALTER TABLE "public"."rol" ADD CONSTRAINT "tbrol_pk"
PRIMARY KEY ("id");
```

Tabla 23 Tabla 22: rol

Autor: Propia Autoría

Tabla 23: mesa

```
CREATE TABLE "public"."mesa"
(
  "id" Integer DEFAULT nextval('tbmesa_id_seq'::regclass) NOT NULL,
  "nromesa" Character varying(20) NOT NULL,
  "x" Integer DEFAULT 0 NOT NULL,
  "y" Integer DEFAULT 0 NOT NULL,
  "ancho" Integer DEFAULT 0 NOT NULL,
  "alto" Integer DEFAULT 0 NOT NULL,
  "id_tbestadomesa" Integer NOT NULL,
  "id_tbcomanda" Integer,
  "idseccionmesa" Integer
)
WITH (
  autovacuum_enabled=true);
CREATE UNIQUE INDEX "iidxetiqueta" ON "public"."mesa"
("nromesa");
ALTER TABLE "public"."mesa" ADD CONSTRAINT "tbmesa_pk"
PRIMARY KEY ("id");
```

Tabla 24 Tabla 23: mesa

Autor: Propia Autoría

Tabla 24: persona

```
CREATE TABLE "public"."persona_rol"
(
  "id_tbro" Integer NOT NULL,
  "id_tbpersona" Integer NOT NULL
)
WITH (
  autovacuum_enabled=true);
ALTER TABLE "public"."persona_rol" ADD CONSTRAINT
"tbrol_persona_pk" PRIMARY KEY ("id_tbro","id_tbpersona");
```

Tabla 25 Tabla 24: persona

Autor: Propia Autoría

Tabla 25: Persona

```
CREATE TABLE "public"."persona"
(
  "id" Integer DEFAULT nextval('tbpersona_id_seq'::regclass) NOT NULL,
  "tipoper" Character(1) NOT NULL,
  "razonsocial" Character varying(250) DEFAULT "::character varying NOT
NULL,
  "ape_pat" Character varying(50) DEFAULT "::character varying NOT
NULL,
  "ape_mat" Character varying(50) DEFAULT "::character varying NOT
NULL,
  "nombre" Character varying(50) DEFAULT "::character varying NOT
NULL,
  "nrodoc" Character varying(20) DEFAULT "::character varying NOT
NULL,
  "direccion" Character varying(400) DEFAULT "::character varying NOT
NULL,
  "telefono" Character varying(15) DEFAULT "::character varying NOT
NULL,
  "email" Character varying(100) DEFAULT NULL::character varying,
  "usuario" Character varying(20) DEFAULT "::character varying NOT
NULL,
  "clave" Character varying(20) DEFAULT "::character varying NOT
NULL,
  "password_hash" Text DEFAULT "::text NOT NULL,
  "api_key" Character varying(32) DEFAULT "::character varying NOT
NULL,
  "id_tbtipo_documento_identidad" Integer NOT NULL,
  "estado" Boolean DEFAULT true NOT NULL,
  "accesodesktop" Boolean DEFAULT false NOT NULL,
  "rolsistema" Integer DEFAULT 0
)
WITH (
  autovacuum_enabled=true);
COMMENT ON COLUMN "public"."persona"."rolsistema" IS '0=Sin
acceso al sistema
1=Nivel Operador.
2=Nivel Administrador';
ALTER TABLE "public"."persona" ADD CONSTRAINT "tbpersona_pk"
PRIMARY KEY ("id");
```

Tabla 26 Tabla 25: Persona

Autor: Propia Autoría

Tabla 26: seccion_mesa

```
CREATE TABLE "public"."seccion_mesa"
( "idseccion" Integer DEFAULT
nextval('tbseccionmesa_idseccion_seq'::regclass) NOT NULL,
"nombre" Character varying(100) NOT NULL)
WITH (
autovacuum_enabled=true);
ALTER TABLE "public"."seccion_mesa" ADD CONSTRAINT
"tbseccionmesa_pkey" PRIMARY KEY ("idseccion");
```

Tabla 27 Tabla 26: seccion_mesa

Autor: Propia Autoría

Tabla 27: producto

```
CREATE TABLE "public"."producto"
( "id" Integer DEFAULT nextval('tbproducto_id_seq'::regclass) NOT NULL,
"tipoprod" Integer DEFAULT 1 NOT NULL,
"nombre" Character varying(150) NOT NULL,
"precio" Numeric(7,2) DEFAULT 0.00 NOT NULL,
"afecto_igv" Boolean DEFAULT true NOT NULL,
"afecto_isc" Boolean DEFAULT false NOT NULL,
"foto" Character varying(150) DEFAULT "::character varying NOT NULL,
"observacion" Character varying(150) DEFAULT "::character varying NOT
NULL,
"id_tbcategoria" Integer NOT NULL,
"estado" Boolean DEFAULT true NOT NULL,
"codbarras" Character varying(30) DEFAULT "::character varying NOT
NULL,
"idmarca" Integer DEFAULT nextval('tbmarca_idmarca_seq'::regclass) NOT
NULL,
"modelo" Character varying(100) DEFAULT "::character varying)
WITH ( autovacuum_enabled=true);
COMMENT ON COLUMN "public"."producto"."tipoprod" IS '1 = Producto
Terminado.
2 = Servicio'
;
COMMENT ON COLUMN "public"."producto"."idmarca" IS '1 = NO
DEFINIDO';
CREATE INDEX "IX_Relationship54" ON "public"."producto" ("idmarca");
ALTER TABLE "public"."producto" ADD CONSTRAINT "tb_producto_pk"
PRIMARY KEY ("id");
```

Tabla 28 Tabla 27: producto

Autor: Propia Autoría

Tabla 28: documento_ingreso_producto

```
CREATE TABLE "public"."documento_ingreso_producto"
( "idreg" Integer DEFAULT nextval('tbregistro_doc_ingreso_idreg_seq'::regclass)
NOT NULL,
  "idalmacen" Integer NOT NULL,
  "tipoorigen" Integer NOT NULL,
  "idorigen" Integer NOT NULL,
  "idtipotransaccion" Integer NOT NULL,
  "idtipodoc" Integer NOT NULL,
  "numdoc" Character varying(15) NOT NULL,
  "fecha_emision" Date NOT NULL,
  "fecha_recep" Date NOT NULL,
  "fecha_registro" Date NOT NULL,
  "igv" Numeric(8,2) NOT NULL,
  "con_igv" Boolean NOT NULL,
  "otros_imp" Numeric(8,2) NOT NULL,
  "desc_doc" Numeric(8,2) NOT NULL,
  "estado" Boolean NOT NULL,
  "tipo_camb" Numeric(8,2) NOT NULL,
  "idusuario" Integer NOT NULL,
  "caja" Character varying(20) NOT NULL,
  "signo" Character(2) NOT NULL,
  "montototdoc" Numeric(10,2) DEFAULT 0 NOT NULL,
  "ndiasvenc" Integer,
  "observacion" Character varying(100) DEFAULT "::character varying NOT NULL,
  "desctodocna" Numeric(8,2) DEFAULT 0 NOT NULL,
  "desctodocafecto" Numeric(8,2) DEFAULT 0 NOT NULL,
  "moneda" Character varying(3) DEFAULT "::character varying NOT NULL,
  "iddocref" Integer,
  "seriedocref" Character varying(4) DEFAULT "::character varying NOT NULL,
  "numerodocref" Character varying(10) DEFAULT "::character varying NOT NULL,
  "actualizastock" Boolean DEFAULT true NOT NULL,
  "idtipomovmer" Integer DEFAULT 1 NOT NULL,
  "codigo_ref" Integer DEFAULT 0 NOT NULL,
  "referencia" Character varying(20) DEFAULT "::character varying NOT NULL,
  "codigo_ref1" Integer DEFAULT 0 NOT NULL,
  "codigo_ref2" Integer DEFAULT 0 NOT NULL,
  "valor_igv" Numeric(10,4) DEFAULT 18 NOT NULL,
  "ajuste" Numeric(6,2) DEFAULT 0 NOT NULL)
WITH ( autovacuum_enabled=true);
ALTER TABLE "public"."documento_ingreso_producto" ADD CONSTRAINT
"registro_doc_ingreso_pkey" PRIMARY KEY ("idreg");
CREATE TRIGGER "trreg_doc_ingreso"
AFTER DELETE OR INSERT OR UPDATE
ON "public"."documento_ingreso_producto" FOR EACH ROW
EXECUTE PROCEDURE "public"."trfreg_doc_ingreso"()
```

Tabla 29 Tabla 28: documento_ingreso_producto

Autor: Propia Autoría

Tabla 29: documento_salida_producto

```
CREATE TABLE "public"."documento_salida_producto"
(
  "idreg" Integer DEFAULT
nextval('tbregistro_doc_salida_idreg_seq'::regclass) NOT NULL,
  "idalmacen" Integer NOT NULL,
  "tipodestino" Integer NOT NULL,
  "iddestino" Integer NOT NULL,
  "idtipomovmer" Integer NOT NULL,
  "idtipodoc" Integer NOT NULL,
  "numdoc" Character varying(15) NOT NULL,
  "fecha_emision" Date NOT NULL,
  "fecha_recep" Date NOT NULL,
  "fecha_registro" Date NOT NULL,
  "con_igv" Boolean NOT NULL,
  "otros_imp" Numeric(8,2) NOT NULL,
  "estado" Boolean NOT NULL,
  "tipo_camb" Numeric(8,2) NOT NULL,
  "idusuario" Integer NOT NULL,
  "caja" Character varying(20) NOT NULL,
  "signo" Character(2) NOT NULL,
  "montototdoc" Numeric(10,2) DEFAULT 0 NOT NULL,
  "idtipotransaccion" Integer NOT NULL,
  "observacion" Character varying(100) DEFAULT "::character varying
NOT NULL,
  "moneda" Character varying(3) DEFAULT "::character varying NOT
NULL,
  "actualizastock" Boolean DEFAULT true NOT NULL,
  "codigo_ref" Integer DEFAULT 0 NOT NULL
)
WITH (
  autovacuum_enabled=true)
;
COMMENT ON COLUMN
"public"."documento_salida_producto"."iddestino" IS 'Cuando es para
cocina indicar el ID de "SALIDA POR VENTA POR COMANDA"'
;

ALTER TABLE "public"."documento_salida_producto" ADD
CONSTRAINT "tbregistro_doc_salida_pkey" PRIMARY KEY ("idreg")
;
```

Tabla 30 Tabla 29: documento_salida_producto

Autor: Propia Autoría

Tabla 30: seriesdoc_emitir

```
CREATE TABLE "public"."seriesdoc_emitir"
(
  "id" Integer DEFAULT nextval('tbseriesdoc_emitir_id_seq'::regclass) NOT
  NULL,
  "id_tbtipodoc" Integer NOT NULL,
  "serie" Character varying(4) DEFAULT "::character varying" NOT NULL,
  "ultimo_emitido" Integer DEFAULT 0 NOT NULL,
  "estado" Boolean DEFAULT true NOT NULL,
  "id_tbsucursalpcs" Integer NOT NULL
)
WITH (
  autovacuum_enabled=true)
;

ALTER TABLE "public"."seriesdoc_emitir" ADD CONSTRAINT
"tbseriesdoc_emitir_pk" PRIMARY KEY ("id")
;
```

Tabla 31 Tabla 30: seriesdoc_emitir

Autor: Propia Autoría

Tabla 31: stock_producto_almacen

```
CREATE TABLE "public"."stock_producto_almacen"
(
  "idstock" Integer DEFAULT
  nextval('tbstockactualalmacen_idstock_seq'::regclass) NOT NULL,
  "idprod" Integer NOT NULL,
  "idalmacen" Integer NOT NULL,
  "stockfisico" Numeric(12,2) NOT NULL,
  "stockreservado" Numeric(8,2) NOT NULL,
  "stockbloqueado" Numeric(8,2) NOT NULL
)
WITH (
  autovacuum_enabled=true)
;

ALTER TABLE "public"."stock_producto_almacen" ADD CONSTRAINT
"tbstockactualalmacen_pkey" PRIMARY KEY ("idstock");
```

Tabla 32 Tabla 31: stock_producto_almacen

Autor: Propia Autoría

Tabla 32: sucursal

```
CREATE TABLE "public"."sucursal"
( "id" Integer DEFAULT nextval('tbsucursal_id_seq'::regclass) NOT
NULL,
"direccion" Character varying(100) NOT NULL)
WITH ( autovacuum_enabled=true);
ALTER TABLE "public"."sucursal" ADD CONSTRAINT "tbsucursal_pk"
PRIMARY KEY ("id");
```

Tabla 33 Tabla 32: sucursal

Autor: Propia Autoría

Tabla 33: sucursal_pc_registrado

```
CREATE TABLE "public"."sucursal_pc_registrado"
( "id" Integer DEFAULT nextval('tbsucursalpcs_id_seq'::regclass) NOT
NULL,
"pc" Character varying(100) NOT NULL,
"printer_doc_comerciales" Character varying(150) DEFAULT "::character
varying NOT NULL,
"printer_documentos" Character varying(150) DEFAULT "::character
varying NOT NULL,
"id_tbsucursal" Integer NOT NULL)
WITH ( autovacuum_enabled=true);
ALTER TABLE "public"."sucursal_pc_registrado" ADD CONSTRAINT
"tbsucursalpcs_pk" PRIMARY KEY ("id");
```

Tabla 34 Tabla 33: sucursal_pc_registrado

Autor: Propia Autoría

Tabla 34: tipo_doc_identidad

```
CREATE TABLE "public"."tipo_doc_identidad"
( "id" Integer DEFAULT
nextval('tbtipo_documento_identidad_id_seq'::regclass) NOT NULL,
"nombre" Character varying(30) NOT NULL,
"longitud" Integer NOT NULL,
"cod_sunat" Integer NOT NULL)
WITH (
autovacuum_enabled=true);
ALTER TABLE "public"."tipo_doc_identidad" ADD CONSTRAINT
"tbtipo_documento_identidad_pk" PRIMARY KEY ("id");
```

Tabla 35 Tabla 34: tipo_doc_identidad

Autor: Propia Autoría

Tabla 35: tipodoc_comprobantepago

```
CREATE TABLE "public"."tipodoc_comprobantepago"
( "id" Integer DEFAULT nextval('tbtipodoc_id_seq'::regclass) NOT NULL,
  "nombre" Character varying(30) NOT NULL,
  "abrev" Character varying(6) DEFAULT "::character varying" NOT NULL,
  "codigo_contable" Character varying(5) DEFAULT "::character varying"
  NOT NULL,
  "prefijos" Character varying(5) DEFAULT "::character varying" NOT
  NULL)
WITH ( autovacuum_enabled=true);
ALTER TABLE "public"."tipodoc_comprobantepago" ADD
CONSTRAINT "tbtipodoc_pk" PRIMARY KEY ("id");
```

Tabla 36 tipodoc_comprobantepago

Autor: Propia Autoría

Tabla 36: tiposervicio_compra

```
CREATE TABLE "public"."tiposervicio_compra"
( "id" Integer DEFAULT nextval('tbtiposervicio_compras_id_seq'::regclass)
  NOT NULL,
  "nombre" Character varying(30) NOT NULL,
  "abrev" Character varying(5))
WITH ( autovacuum_enabled=true);
ALTER TABLE "public"."tiposervicio_compra" ADD CONSTRAINT
"tbtiposervicio_compras_pkey" PRIMARY KEY ("id");
```

Tabla 37 Tabla 36: tiposervicio_compra

Autor: Propia Autoría

Tabla 37: tipotransaccion

```
CREATE TABLE "public"."tipotransaccion"
( "idtipo" Integer DEFAULT
  nextval('tbtipotransaccion_idtipo_seq'::regclass) NOT NULL,
  "nombre" Character varying(30))
WITH ( autovacuum_enabled=true);

ALTER TABLE "public"."tipotransaccion" ADD CONSTRAINT
"tbtipotransaccion_pkey" PRIMARY KEY ("idtipo")
```

Tabla 38 Tabla 37: tipotransaccion

Autor: Propia Autoría

Tabla 38: tipo_movimiento_mercaderia

```
CREATE TABLE "public"."tipo_movimiento_mercaderia"
(
  "idtipomovmer" Integer DEFAULT
nextval('tipo_movimiento_mercaderia_idtipomovmer_seq'::regclass) NOT
NULL,
  "nombre" Character varying(50) NOT NULL,
  "codigo_contable" Character(2) NOT NULL
)
WITH (
  autovacuum_enabled=true)
;

ALTER TABLE "public"."tipo_movimiento_mercaderia" ADD
CONSTRAINT "tipo_movimiento_mercaderia_pkey" PRIMARY KEY
("idtipomovmer")
;
```

Tabla 39 Tabla 38: tipo_movimiento_mercaderia

Autor: Propia Autoría

8.4. Implementación.

1. Acceso al sistema.



CEVICHERIA FLOR DE MAR

Usuario: SOPORTE

Clave: *****

login Salir

¿Olvidó Clave?

Sistema Via Online (flordelmar.com)

Imagen No. 18 Formulario Acceso al Sistema

Fuente: Propia autoría.

Descripción: Mediante este formulario se accesa al sistema mediante el tipo de usuario y la clave que nos brinda al administrador del sistema. En el caso de que no se recuerde la contraseña se cuenta con la opción “¿olvidó clave?” que permite la recuperación mediante su correo electrónico.

2. Menú principal.

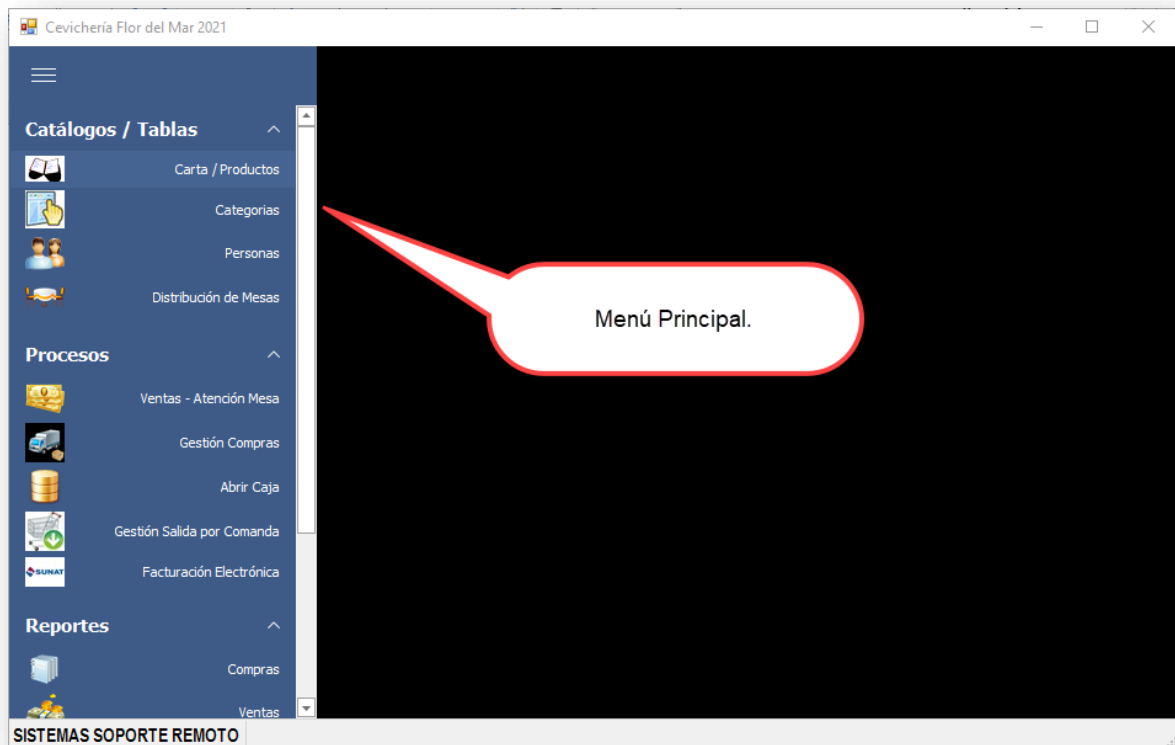


Imagen No. 19 Formulario Menú Principal.

Fuente: Propia autoría.

Descripción: Una vez que el sistema permita el acceso, después de la validación de las credenciales de acceso, nos muestra el formulario del menú principal que contiene las opciones clasificadas en tres principales paneles de acciones:

- a. Catálogos o tablas que permiten la actualización de los datos registrados en la base de datos.
- b. Procesos que registran los movimientos o transacciones que se realizan en la cevichería.
- c. Reportes que se generan en forma impresa de los datos almacenados de los catálogos y los que son resultados de los procesos.

3. Registro general de productos.

Botones para realizar operaciones como

Opciones de búsqueda

Comentarios de cantidad de registros encontrados

Registros encontrados

id	nombre	P.Vta	estado
1	Arroz Blanco	3.00	☒
2	ARROZ CON LANGOSTINOS. Deliciosa selección de langostinos frescos a base de una salsa madre	35.00	☒
3	ARROZ CON MARISCOS. Esquisita mezcla de mariscos del día a base de una salsa madre de pimienta	30.00	☒
4	Camote	3.00	☒
5	Cancha	3.00	☒
6	CEVICHE DE MARISCOS. Mezcla de mariscos (mariscos, pulpo, calamar y langostinos). Acompañado	42.00	☒
7	CEVICHE DE PESCADO. Pescado del día acompañado de choclo, cancha, yuca y camote.	30.00	☒
8	CEVICHE DE PULPO. Ceviche clásico a base de pulpo. Acompañado de choclo, cancha y camote.	40.00	☒
9	CEVICHE FLOR DE MAR. Combinación de ajíes. Generosa variedad de pescados y mariscos. Acompañado	38.00	☒
10	CEVICHE MIXTO ESPECIAL. Pescados y Mariscos de estación acompañado de choclo, cancha y camote	25.00	☒
11	CHAUFU DE LANGOSTINOS. Langostinos selectos salteados al wok con arroz y salsa oriental.	38.00	☒
12	CHAUFU DE MARISCOS. Fusión de nuestra cocina marina con la cocina oriental, al wok	30.00	☒
13	CHAUFU DE PESCADO. Deliciosos trozos de pescado del día acompañados de una salsa oriental, al wok	35.00	☒
14	CHICHARRÓN DE CALAMAR. Aros de calamar con yuca frita y salsa.	40.00	☒
15	CHICHARRÓN DE PESCADO. pescado del día acompañado de yuca frita y nuestra salsa.	30.00	☒
16	CHICHARRÓN MIXTO. Crujientes mariscos de estación con yuca frita salsa.	28.00	☒
17	Choclo	3.00	☒
18	CHORITOS A LA CHALACA	18.00	☒
19	COCA COLA 1 LT REGULAR / ZERO	7.00	☒
20	CUZQUEÑA	9.00	☒
21	DE PESCADO Y PULPO. Tradicional ceviche acompañado de tiras láminas de pulpo	49.00	☒
22	INCA KOLA 1 LT REGULAR / ZERO	7.00	☒
23	JALEA DE PESCADO. Pescado del día, cebolla, tomate y culantro en salsa criolla acvichada.	32.00	☒
24	JALEA MIXTA. Mariscos, moluscos y pescados apunados	30.00	☒
25	JARRA DE CHICHA	12.00	☒
26	JARRA DE MARACUYA	17.00	☒

Imagen No. 20 Formulario Registro General de Productos.

Fuente: Propia autoría.

Descripción: En este formulario se visualizan los productos de la cevichería.

Se cuenta con una barra de opciones el cual está conformado por los botones que permiten realizar las operaciones de agregar nuevo producto, modificar sus datos y eliminarlos. También se muestra la barra de búsqueda el cual se puede hacer por producto o categoría. En este panel se puede suspender temporalmente un producto.

4. Características del producto

Registro General de Producto(s)...

Nuevo - F2 Modificar - F3 Eliminar - F4 Cerrar

Producto: Categoria: [Cualquiera] 100 Ver

44 registros visualizados. Total de coincidencias encontradas 44 segun criterios de busqueda.

id	producto	estado
19	40 COCA COLA 1 LT RE	☒
20	42 CUZQUEÑA	☒
21	9 DE PESCADO Y PUL	☒
22	41 INCA KOLA 1 LT RE	☒
23	23 JALEA DE PESCADO	☒
24	24 JALEA MIXTA Maris	☒
25	44 JARRA DE CHICHA	☒
26	46 JARRA DE MARACU	☒
27	45 JARRA LIMONADA	☒
28	33 LECHE DE TIGRE	☒
29	31 PARIHUELA DE Cabr	☒
30	26 PESCADO APANADO	☒
31	25 PESCADO DORADO.	☒
32	16 PICANTE DE LANGO	☒
33	15 PICANTE DE MARIS	☒
34	17 PICANTE DE PULPO.	☒
35	18 PICANTE SABOR & S	☒
36	43 PILSEN	☒
37	19 PULPO AL OLIVO. Delicioso pulpo fresco bañado con muestra salsa especial al olivo.	☒
38	28 Ronda Marina de Pescado Flor del Mar. Ceviche de Pescado. Chaufa de Pescado y Chicharrón de Pesc	☒

Datos Principales Otros

Tipo: * PLATO

Nombre: * SECO DE CABRITO CON ARROZ Y FREJOL

Categoria: * Arroces

Marca: * NO ESPECIFICADO

Cod.Barras:

Descripcion:

Precio: S/ 18.00

☒ Afecto IGV? ☐ Afecto ISC?

☒ Activo? Grabar Cancelar

Características del producto.

Imagen No. 21 Formulario de características del producto.

Fuente: Propia autoría.

Descripción: En este formulario observamos los datos principales y otros respecto al producto que ofrece la cevichería. Mediante este formulario podemos además especificar si se encuentra o no afecto al IGV (impuesto general a las ventas) y al ISC (impuesto selectivo al consumo). Por último, podemos especificar si se encuentra o no disponible para solicitud del cliente. El código de barras se aplica a los productos terminados como gaseosas, cerveza, etc.

5. Otros datos del producto.

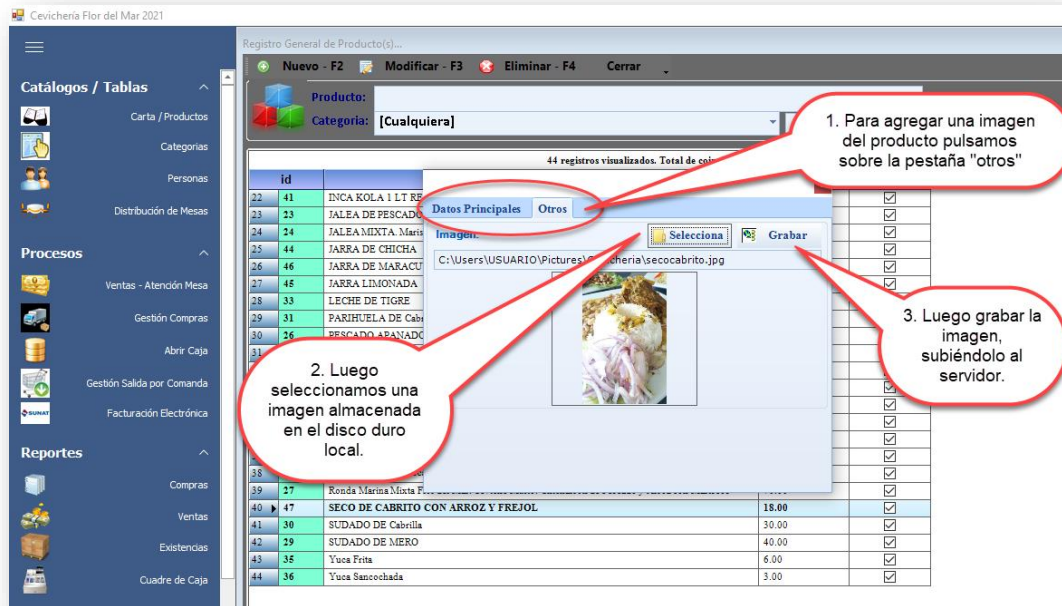


Imagen No. 22 Formulario Ficha Otros.

Fuente: Propia autoría.

Descripción: En esta sección se agrega la imagen del producto el cual requiere que el usuario pulse click en “Selecciona” para ubicar la unidad, carpeta y archivo imagen a subir. Una vez ubicado el archivo de la imagen se procede a pulsar el botón “Grabar” para subirlo al servidor y disponerlo en forma inmediata.

6. Registro general de personas.

Registro General de Persona(s)...

Nuevo - F2 Modificar - F3 Eliminar - F4 Cerrar Registro de Usuario Sistema

Persona: [Cualquiera] Rol: [Cualquiera] 200 Ver

ID	tipo	nombres	nrodoc
1	N	SISTEMAS SOPORTE REMOTO	DNI 42774472
2	N	VARIOS	DNI 00000000
3	J	SALIDA POR PEDIDO COMANDA	
4	J	CUBAS CRUZ ELVES ORLANDO	
5	J	COLEGIO MANUEL PARDO	
6	N	TEJADA QUEVEDO FERNANDO MIGUEL	DNI 16687937
7	J	TEJADA QUEVEDO FERNANDO MIGUEL	RUC 10166879375
8	J	TECNOLOGIA Y SERVICIOS GENERALES EMANUEL E.I.R.L.	RUC 20600080882
9	J	COLEGIO SAN AGUSTIN	RUC 20103560757
10	J	COLEGIO SANTA MARIA REYNA	RUC 20137382303
11	J	SERVICIOS SOCIALES SOLIDARIOS ABATE PIERRE	RUC 20543772586
12	N	ALBINES AAGAZCO MARY DEL PILAR	DNI 47870590
13	N	CRUZADO PEÑA ALVARO ALBERTO	DNI 47268030
14	N	CUBAS CRUZ MARIA ARMANDINA	DNI 42714472
15	J	JIREH LOGISTICA INTEGRAL SOCIEDAD ANONIMA - JIREH LOG	RUC 20600549180
16	J	RESTAURANT CEBICHERIA CABO BLANCO E.I.R.L.	
17	J	ANDINO SERVICIOS Y MONTAJES INDUSTRIALES S.A.C.	
18	J	G.I. INDUSTRIA PERU S.A.C.	
19	J	PETROLEOS DEL PERU PETROPERU SA	
20	J	AVICOLA EVA CECILIA EMPRESA INDIVIDUAL DE RESPONSABILIDAD	
21	J	SODIMAC PERU S.A.	RUC 20589230724
22	N	PERALTA QUEVEDO DEMETRIO	DNI 03823403
23	N	PERALTA ARCA GABRIEL ANTONIO	DNI 43007914
24	N	JIMENEZ PALOMINO HEBERT ADALBERTO	DNI 03891866
25	J	AJEPER S.A.	RUC 20331061655
26	J	APTIM PERU S.A.C.	RUC 20601961009
27	N	ANGELES CAMPOS VICTOR ANDRES	DNI 08878277
28	J	COLEGIO SAN AGUSTIN DE VILLA E.I.R.L.	RUC 20556874981

Opciones de búsqueda

Botones para agregar, modificar o eliminar registros. Adicionalmente para definir los usuarios del sistema.

Personas encontradas (clientes, proveedores, usuarios del sistema)

Imagen No. 23 Formulario Registro general de personas.

Fuente: Propia autoría.

Descripción: Este formulario visualiza las personas naturales (con DNI) y jurídicas (con RUC) que han sido registradas. Similar al formulario del registro general de productos se cuenta con los botones para agregar, modificar y eliminar personas y usuarios del sistema. Encontramos opciones de búsqueda de personas o por rol.

7. Formulario para agregar o modificar registro de persona.

Formulario para agregar o modificar un nuevo registro (persona). En el caso de los clientes estos se agregan automáticamente al momento del pago de lo consumido en la ventana de ventas.

ID	tipo	nombres	
64	87	N	PERALTA ARCA LUIS JAVIER
65	77	N	PERALTA QUEVEDO DEMETRIO
66	118	N	PEREZ CALISAN JOSE MERCEDES
67	74	J	PEREZ CALISAN JOSE MERCEDES
68	93	N	PEREZ CALISAN JOSE MERCEDES
69	69	J	PEREZ CALISAN JOSE MERCEDES
70	135	N	PEREZ CALISAN JOSE MERCEDES
71	107	N	PEREZ CALISAN JOSE MERCEDES
72	3	N	PEREZ CALISAN JOSE MERCEDES
73	56	N	PEREZ CALISAN JOSE MERCEDES
74	1	N	PEREZ CALISAN JOSE MERCEDES
75	76	N	PEREZ CALISAN JOSE MERCEDES
76	100	N	PEREZ CALISAN JOSE MERCEDES
77	82	N	PEREZ CALISAN JOSE MERCEDES
78	80	N	PEREZ CALISAN JOSE MERCEDES
79	81	N	PEREZ CALISAN JOSE MERCEDES
80	128	N	PEREZ CALISAN JOSE MERCEDES
81	149	N	PEREZ CALISAN JOSE MERCEDES
82	2	N	PEREZ CALISAN JOSE MERCEDES
83	136	N	PEREZ CALISAN JOSE MERCEDES
84	112	N	PEREZ CALISAN JOSE MERCEDES
85	124	N	PEREZ CALISAN JOSE MERCEDES
86	122	N	PEREZ CALISAN JOSE MERCEDES

Datos Principales

Datos Generales...

Tipo Persona: Jurídica

Documento: RUC

Número: 20600080882

Persona Jurídica...

Razon Social: TECNOLOGIA Y SERVICIOS GENERALES EMANUEL E.I.R.L.

Dirección Principal

E-Mail:

Telefono:

Telefono:

Estado: [icon] Grabar [icon] Cancelar [icon]

Imagen No. 24 Formulario para agregar o modificar registro de persona.

Fuente: Propia autoría.

Descripción: Este formulario permite agregar o modificar un nuevo registro de personas, así como conocer detalles del mismo. En el caso de los clientes que han realizado consumo en el momento de pagar se reagistran sus datos automáticamente a la base de datos.

8. Distribución de secciones.

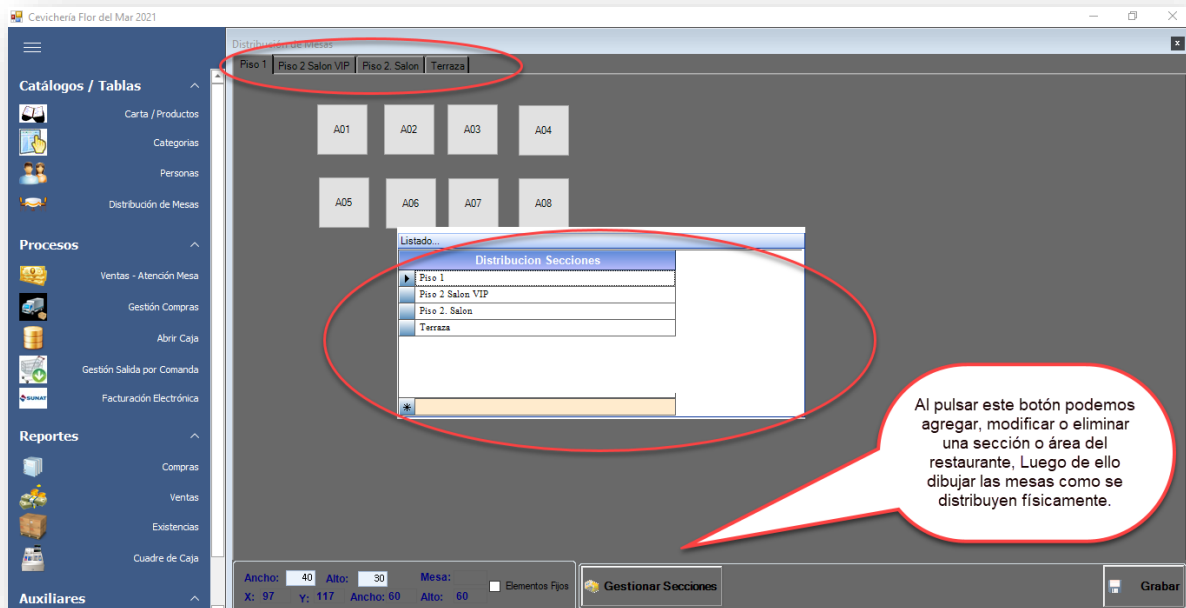


Imagen No. 25 Formulario distribución de secciones.

Fuente: Propia autoría.

Descripción: El presente formulario permite realizar la distribución de secciones utilizando el botón “Gestionar Secciones” el cual permite agregar, modificar o eliminar una sección o área del restaurant. Por cada sección se puede agregar o eliminar las mesas.

9. Distribución de Mesas.

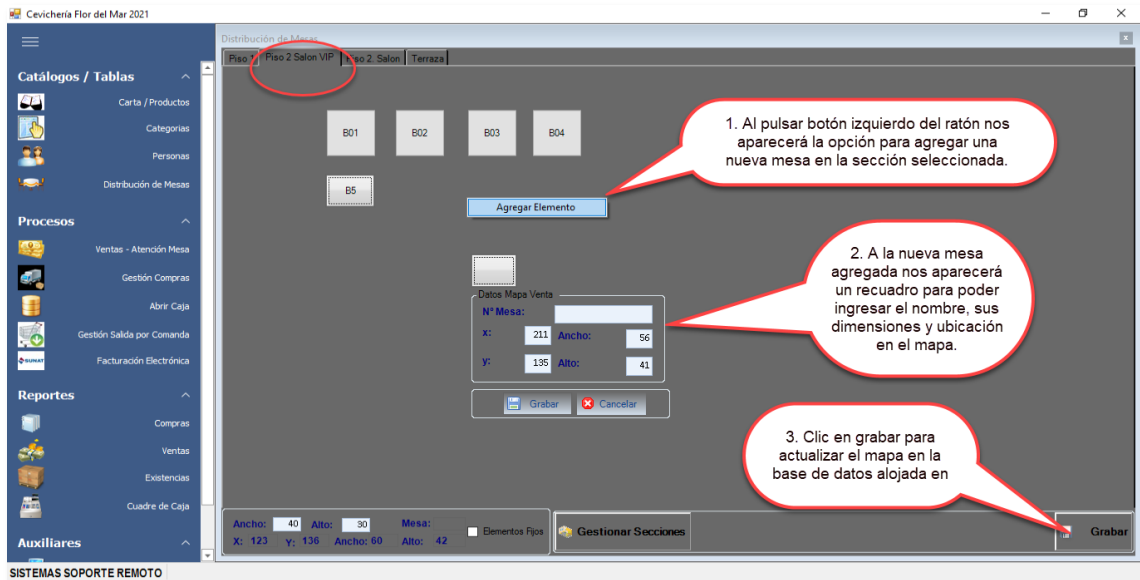


Imagen No. 26 Formulario para Asignación de mesas.

Fuente: Propia autoría.

Descripción: En este formulario también permite gestionar las mesas que se ubicarán en la sección correspondiente. El botón “Agregar elemento” que aparece al pulsar botón izquierdo del mouse, permite ubicar una nueva mesa indicando previamente sus nombre, dimensiones y ubicación en el mapa. Finalmente se presiona el botón grabar.

10. Registro de ventas.

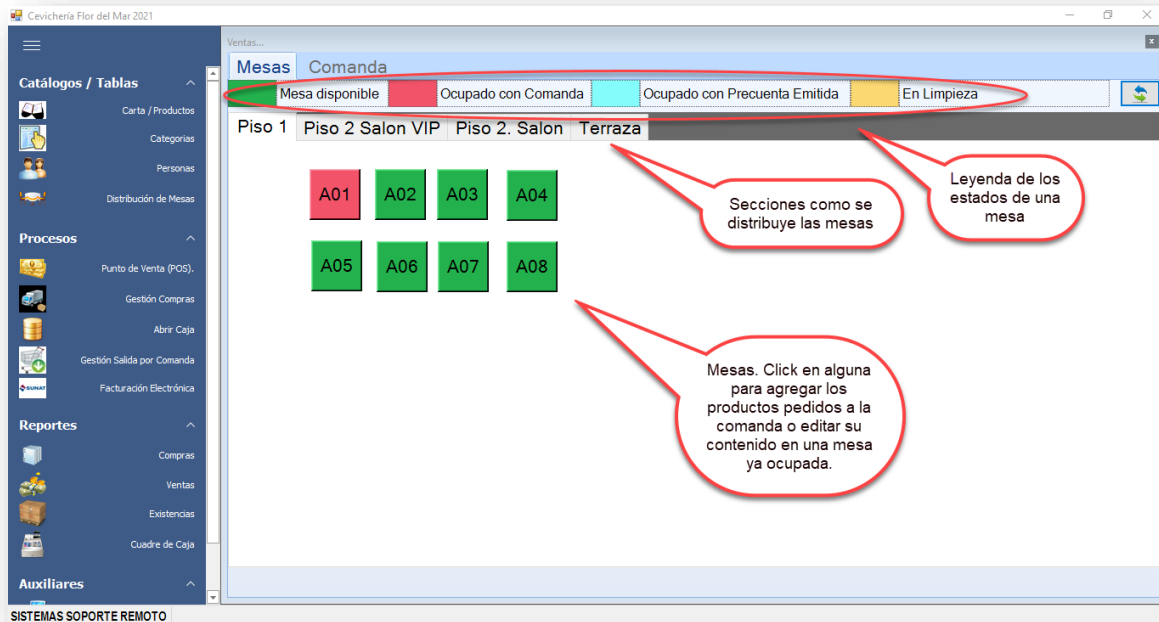


Imagen No. 27 Formulario de Registro de venta según mesa.

Fuente: Propia autoría.

Descripción: Para realizar una venta debemos agregar a la comanda los productos pedidos o editar su contenido en una mesa ya ocupada.

11. Comanda

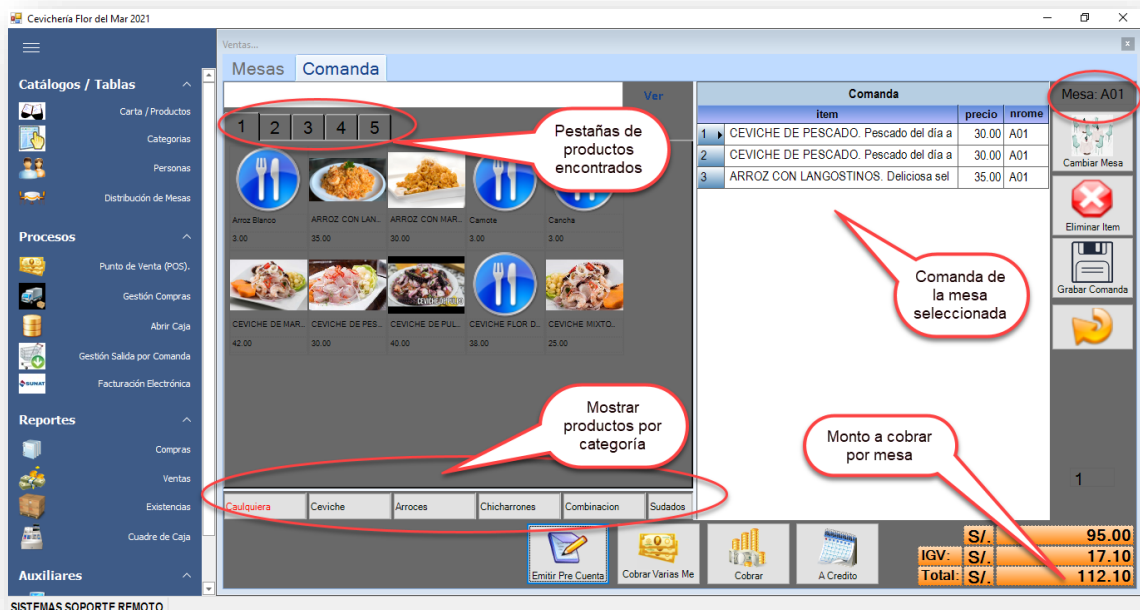


Imagen No. 28 Formulario de gestión de comanda.

Fuente: Propia autoría.

Descripción: En este formulario nos permite ubicar los productos agrupadas por las pestañas correspondientes, pero también es posible listarlos por categorías seleccionando la opción correspondiente en la barra indicada. Se selecciona los productos que solicita cada mesa. Al conjunto de platos, bebidas u otros del pedido se llama comanda. En esta vista se muestra el costo del pedido con y sin IGV.

12. Selección ítems para concretar venta.

Selección de ítems para concretar venta...

item	precio	nrome
ARROZ CON LANGOSTINO	35.00	A01
CEVICHE DE PESCADO. Pe	30.00	A01
CEVICHE DE PESCADO. Pe	30.00	A01

Items seleccionados

itemImprimir	precio	nromesa

Luego de pulsar el botón "Cobrar" elegimos los ítems a cobrar.

S/. 0.00 Siguiente

Emitir Pre Cuenta Cobrar Varias Mesas Cobrar A Crédito

Mesa: A01

Cambiar Mesa Eliminar ítem Grabar Comanda

S/. 95.00
IGV: S/. 17.10
Total: S/. 112.10

Imagen No. 29 Formulario selección de ítems para venta.

Fuente: Propia autoría.

Descripción: Mediante este formulario visualizamos los ítems consumidos pendientes antes de concretar el pago. Una vez pulsado el botón cobrar seleccionamos los ítems que se incluirán en la cobranza.

13. Registro de pago de comanda.

Consumido: 95.00 Descuento: 0.00 Total: 95.00
IGV: 17.10 A Pagar: 112.10

Formas de pago

VISA 112.10
MasterCard 0.00
Canje Nota Credito 0.00

comprobante a emitir

Boleta Factura Ticket Provisional

S/ 95.00
IGV: S/ 17.10
Total: S/ 112.10

Imagen No. 30 Formulario para gestión de pago según comanda.

Fuente: Propia autoría.

Descripción: mediante este formulario de la comanda podemos generar el ticket provisional para hacer de conocimiento el pago a realizar el cliente. El cliente al formalizar el pago, se selecciona el tipo de pago (efectivo, simple o múltiple) así como el tipo de tarjeta si es que el pago se realiza mediante una de estas.

8.5. Presupuesto

A. Recursos y presupuestos

CLASIFICACIÓN	ITEM	CANTIDAD	TIEMPO	TOTAL
EQUIPO DE DESARROLLO	ASESOR	1	5	S/ 1,200.00
	DESARROLLADOR	1	2 meses	S/ 1,800.00
SUBTOTAL				S/ 3,000.00
DOCUMENTACIÓN	FOTOCOPIAS	300		S/ 15.00
	IMPRESIONES	200		S/ 140.00
	ESPIRALADOS	6		S/ 27.00
	EMPASTADOS	5		S/ 125.00
SUBTOTAL				S/ 307.00
OTROS	PASAJES		3 meses	S/ 220.00
	CHUWI Hi10 X,10.1inch Tablet PC,Windows10	5		S/ 7,000.00
	PAQUETE DE DATOS		4 meses	S/ 180.00
SUBTOTAL				S/ 7,400.00
TOTAL				S/ 10,707.00

Tabla 40 Recursos y Presupuestos.

Fuente: Propia autoría

A. Recuperación

Monto venta antes prom.	S/ 18,000.00	INCREMENTO	S/ 5,500.00
Monto venta ahora prom.	S/ 23,500.00	RECUPERACIÓN	S/ 1,650.00

MES	COSTO A RECUPERAR	RECUPERACIÓN	SALDO A RECUPERAR
1	S/ 10,707.00	S/ 1,650.00	S/ 9,057.00
2	S/ 9,057.00	S/ 1,650.00	S/ 7,407.00
3	S/ 7,407.00	S/ 1,650.00	S/ 5,757.00
4	S/ 5,757.00	S/ 1,650.00	S/ 4,107.00
5	S/ 4,107.00	S/ 1,650.00	S/ 2,457.00
6	S/ 2,457.00	S/ 1,650.00	S/ 807.00
7	S/ 807.00	S/ 1,650.00	-S/ 843.00

Tabla 41 Recuperación.

Fuente: propia autoría

Capítulo IX:

REFERENCIAS BIBLIOGRÁFICAS

CAPITULO IX: BIBLIOGRAFIA

ABC (2015). www.abc.es/tecnologia/consultorio/20150216/abci--201502132105.html. Recuperado el 23 de noviembre de 2020.

Amodeo, E. (2013). Principios de Diseño de Apis Rest. Editorial LeanPub. Madrid. España.

Araya P. (2012). Administración de SGBD Postgresql. Editorial Open libra. Colombia.

Carrillo Perez, I. (2009). Metodología De Desarrollo De Software. Edición 2. México.

Ceballos Sierra, F. (2008) Enciclopedia De Microsoft Visual Basic. Editorial Ra- Wa Editorial. 2da. Edición. Perú.

Celaya, J. (2008). La empresa en la web 2.0. Barcelona: Gestión 2000.

Citiprogram (2021). El estándar confiable en investigación, ética y capacitación en cumplimiento. Extraído de <https://about.citiprogram.org/> el 21 de noviembre de 2021.

Concytec (2020). Acceso Libre a la Investigación Científica para su Innovación. Extraído de https://alicia.concytec.gob.pe/vufind/Record/UNIS_5b55a9811d9ab27b8e45c193546b0187 el 18 de noviembre de 2021.

Coronel, C. (2012). Base De Datos, Implementacion Y Administracion. Editorial Language Learning. Novena Edición. México.

Date, C. (2006). Introducción a los sistemas de base de datos. Editorial Pearson Educación. México.

Duncan Mackenzle, K. (2012). "Aprendiendo Visual Basic .Net En 21 Lecciones Avanzadas", Editorial Pearson Educación, México

Ecured (2020). Ecured.cu/Carta_(men%C3%BA_de_restaurante).
Publicado el 26 de noviembre de 2020.

Emagister (2020). www.emagister.com/blog/que-es-el-net-para-que-sirve/. Recuperado el 24 de noviembre de 2020.

Fernández C. & Baptista F. (2014). Metodología de la Investigación. Editorial Mcgraw-Hill / Interamericana Editores, S.A. Sexta Edición. México.

Ginesta Gibert, M. & Mora Pérez, O. (2009), Base de Datos PostgreSQL, España: Autoedición.

IBM. API REST. <https://developer.ibm.com/es/articles/ws-restful>. Consultado el 17 de octubre de 2020.

IEBS (2018). www.iebschool.com/blog/que-es-api-rest-integrar-negocio-business-tech/. Recuperado el 25 de noviembre de 2020.

Infomed (2020). instituciones.sld.cu/toximed/2017/04/16/que-es-gestion-de-la-informacion. Recuperado el 25 de noviembre de 2020.

INE (2021). ¿Qué es población? Extraído de <https://www.ine.cl/ine-ciudadano/definiciones-estadisticas/poblacion/que-es-poblacion> el 20 de abril de 2021.

Jiménez, L (2012), Introducción a las Tecnologías Web, América Latina: Le-mos.

Juárez, P. (2012). Análisis y diseño de sistemas de información. Extraído de <http://patriciajuarezpz.blogspot.com/2012/10/analisis-de-datos-cuantitativos.html> el 21 de noviembre de 2021.

Lardièrre L. (2019) PostgreSQL. Administración y explotación de sus bases de datos. Editorial Epsilon. España.

López, L. (2004). Punto Cero. Carrera de Ciencias de la Comunicación Social - Universidad Católica Boliviana San Pablo. Cochabamba. Bolivia

Luján, S (s.f.). Programacion De Aplicaciones Web: Historia, Principios Basicos Y Clientes Web, Editorial Club Universitario, San Vicente. Madrid.

Orbegozo Arana, B. (2015). Curso Practico Avanzado De Postgresql: La Base De Datos Más Potente. Altaria. Madrid.

Pardinas, F. (1991). Metodología y Técnicas de Investigación en Ciencias Sociales. Editorial Siglo XXI. Bogotá.

Pimienta, R. (2000). Encuestas probabilísticas vs. no probabilísticas. Universidad Autónoma Metropolitana Unidad Xochimilco. Distrito Federal de México. México.

Postgresql (2020). <https://www.postgresql.org/about>. Recuperado el 25 de noviembre de 2020.

Puerta Gonzales, J. (2015). Desarrollo de una API para la descripción y gestión de Servicios Web REST. http://repositori.uji.es/xmlui/bitstream/handle/10234/156006/TFM_2014_puertaJ.pdf?sequence=1&isAllowed=y. Recuperado el 25 de noviembre de 2020.

Redhat (2019). www.redhat.com/es/topics/api/what-are-application-programming-interfaces. Recuperado el 26 de noviembre de 2020.

Rdstation (2020). <https://www.rdstation.com/es/ventas/>. Recuperado el 20 de noviembre de 2020.

Redhat (2019). www.redhat.com/es/topics/api/what-are-application-programming-interfaces. Recuperado el 26 de noviembre de 2020.

Significados (2020). www.significados.com/sistema. Recuperado el 23 de noviembre de 2020.

Rumbaugh J., Jacobson I., Booch G. (2009) “El Lenguaje Unificado De Modelado Manual De Referencia”, Editorial Pearson Educación, Madrid

Villarino Marzo, J. (2018). La privacidad en el entorno del cloud computing. Madrid, Editorial Reus. Recuperado de <https://elibro.net/es/ereader/udch/121544?page=27>.

ANEXO

-- Create foreign keys (relationships) section -----

```
ALTER TABLE "public"."ctapagar_abono"  
ADD CONSTRAINT "tbdocpago_ctapagar_fk"  
FOREIGN KEY ("idusuario")  
REFERENCES "public"."persona" ("id")  
ON DELETE NO ACTION  
ON UPDATE NO ACTION  
;
```

```
ALTER TABLE "public"."ctapagar_abono"  
ADD CONSTRAINT "tbdocpago_ctapagar_fk1"  
FOREIGN KEY ("idtipodoc")  
REFERENCES "public"."tipodoc_comprobantepago" ("id")  
ON DELETE NO ACTION  
ON UPDATE NO ACTION  
;
```

```
ALTER TABLE "public"."almacen"  
ADD CONSTRAINT "almacen_cod_suc_fkey"  
FOREIGN KEY ("idsucursal")  
REFERENCES "public"."sucursal" ("id")  
ON DELETE NO ACTION  
ON UPDATE NO ACTION  
;
```

```
ALTER TABLE "public"."comanda_mesa"  
ADD CONSTRAINT "pk_idpersona_fkidusuario_tbcomanda"  
FOREIGN KEY ("idusuario")  
REFERENCES "public"."persona" ("id")  
MATCH FULL  
ON DELETE NO ACTION  
ON UPDATE NO ACTION  
;
```

```
ALTER TABLE "public"."comanda_mesa"  
ADD CONSTRAINT "tbpersona_fk"  
FOREIGN KEY ("id_tbpersona")  
REFERENCES "public"."persona" ("id")  
MATCH FULL  
ON DELETE SET NULL  
ON UPDATE CASCADE  
;
```

```
ALTER TABLE "public"."ctas_cobrar"  
ADD CONSTRAINT "fk_idempresa_idpersona"  
FOREIGN KEY ("idempresa")
```

```

REFERENCES "public"."persona" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."ctas_cobrar"
ADD CONSTRAINT "fk_idpersona_idusuario"
FOREIGN KEY ("idusuario")
REFERENCES "public"."persona" ("id")
MATCH FULL
ON DELETE NO ACTION
ON UPDATE CASCADE
;

ALTER TABLE "public"."ctas_cobrar"
ADD CONSTRAINT "tbpersona_fk"
FOREIGN KEY ("id_tbpersona")
REFERENCES "public"."persona" ("id")
MATCH FULL
ON DELETE RESTRICT
ON UPDATE CASCADE
;

ALTER TABLE "public"."ctas_cobrar"
ADD CONSTRAINT "tbtipodoc_fk"
FOREIGN KEY ("id_tbtipodoc")
REFERENCES "public"."tipodoc_comprobantepago" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."cta_pagar"
ADD CONSTRAINT "tbctas_pagar_fk"
FOREIGN KEY ("idpersona")
REFERENCES "public"."persona" ("id")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;

ALTER TABLE "public"."cta_pagar"
ADD CONSTRAINT "tbctas_pagar_fk1"
FOREIGN KEY ("idusuario")
REFERENCES "public"."persona" ("id")
ON DELETE NO ACTION
ON UPDATE NO ACTION

```

;

```
ALTER TABLE "public"."cta_pagar"  
ADD CONSTRAINT "tbctas_pagar_fk4"  
FOREIGN KEY ("idtipodoc")  
REFERENCES "public"."tipodoc_comprobantepago" ("id")  
ON DELETE NO ACTION  
ON UPDATE NO ACTION
```

;

```
ALTER TABLE "public"."cta_pagar"  
ADD CONSTRAINT "tbctas_pagar_fk3"  
FOREIGN KEY ("idtiposervicio")  
REFERENCES "public"."tiposervicio_compra" ("id")  
ON DELETE NO ACTION  
ON UPDATE NO ACTION
```

;

```
ALTER TABLE "public"."ctapagar_detalle_abono"  
ADD CONSTRAINT "tbdetpago_ctapagar_fk1"  
FOREIGN KEY ("iddoc")  
REFERENCES "public"."ctapagar_abono" ("iddoc")  
ON DELETE NO ACTION  
ON UPDATE NO ACTION
```

;

```
ALTER TABLE "public"."ctapagar_detalle_abono"  
ADD CONSTRAINT "tbdetpago_ctapagar_fk"  
FOREIGN KEY ("idcta")  
REFERENCES "public"."cta_pagar" ("idcta")  
ON DELETE NO ACTION  
ON UPDATE NO ACTION
```

;

```
ALTER TABLE "public"."comanda_detalle"  
ADD CONSTRAINT "tbcomanda_fk"  
FOREIGN KEY ("id_tbcomanda")  
REFERENCES "public"."comanda_mesa" ("id")  
MATCH FULL  
ON DELETE CASCADE  
ON UPDATE CASCADE
```

;

```
ALTER TABLE "public"."comanda_detalle"  
ADD CONSTRAINT "tbproducto_fk"  
FOREIGN KEY ("id_tbproducto")  
REFERENCES "public"."producto" ("id")
```

```

MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."detalles_ingreso_productos"
ADD CONSTRAINT "detalle_reg_ing_cod_pres_prod_fkey"
FOREIGN KEY ("idprod")
REFERENCES "public"."producto" ("id")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;

ALTER TABLE "public"."detalles_ingreso_productos"
ADD CONSTRAINT "detalle_reg_ing_cod_reg_fkey"
FOREIGN KEY ("idreg")
REFERENCES "public"."documento_ingreso_producto" ("idreg")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;

ALTER TABLE "public"."documento_salida_detalles"
ADD CONSTRAINT "tbdetalle_reg_pedido_cod_pres_prod_fkey"
FOREIGN KEY ("idprod")
REFERENCES "public"."producto" ("id")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;

ALTER TABLE "public"."documento_salida_detalles"
ADD CONSTRAINT "detalle_reg_pedido_cod_reg_fkey"
FOREIGN KEY ("idreg")
REFERENCES "public"."documento_salida_producto" ("idreg")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;

ALTER TABLE "public"."ctapagar_detpago"
ADD CONSTRAINT "tbctas_cobrar_fk"
FOREIGN KEY ("idcta_tbctas_cobrar")
REFERENCES "public"."ctas_cobrar" ("idcta")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."ctapagar_detpago"

```

```

ADD CONSTRAINT "tbdocpago_ctacobrar_fk"
FOREIGN KEY ("id_tbdcpago_ctacobrar")
REFERENCES "public"."docpago_ctacobrar" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."docpago_ctacobrar"
ADD CONSTRAINT "fk_idusuario_pk_idpersona_tbtacobrar"
FOREIGN KEY ("idusuario")
REFERENCES "public"."persona" ("id")
MATCH FULL
ON DELETE NO ACTION
ON UPDATE NO ACTION
;

ALTER TABLE "public"."docpago_ctacobrar"
ADD CONSTRAINT "tbpersona_fk"
FOREIGN KEY ("id_tbpersona")
REFERENCES "public"."persona" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."docpago_ctacobrar"
ADD CONSTRAINT "tbtipodoc_fk"
FOREIGN KEY ("id_tbtipodoc")
REFERENCES "public"."tipodoc_comprobantepago" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."docpago_ctacobrar_mediopago"
ADD CONSTRAINT "tbdocpago_ctacobrar_fk"
FOREIGN KEY ("id_tbdcpago_ctacobrar")
REFERENCES "public"."docpago_ctacobrar" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."docpago_ctacobrar_mediopago"
ADD CONSTRAINT "tbmediopago_fk"
FOREIGN KEY ("id_tbmediopago")

```



```

REFERENCES "public"."medio_pago" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."mesa"
ADD CONSTRAINT "tbestadomesa_fk"
FOREIGN KEY ("id_tbestadomesa")
REFERENCES "public"."estadomesa" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."mesa"
ADD CONSTRAINT "fk_idseccion_mesa"
FOREIGN KEY ("idseccionmesa")
REFERENCES "public"."seccion_mesa" ("idseccion")
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT VALID
;

ALTER TABLE "public"."persona"
ADD CONSTRAINT "tbtipo_documento_identidad_fk"
FOREIGN KEY ("id_tbtipo_documento_identidad")
REFERENCES "public"."tipo_doc_identidad" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."producto"
ADD CONSTRAINT "tbcategoria_fk"
FOREIGN KEY ("id_tbcategoria")
REFERENCES "public"."categoria_producto" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."documento_ingreso_producto"
ADD CONSTRAINT "registro_doc_ingreso_cod_almacen_fkey"
FOREIGN KEY ("idalmacen")
REFERENCES "public"."almacen" ("idalmacen")
ON DELETE NO ACTION

```

```
ON UPDATE NO ACTION
;
```

```
ALTER TABLE "public"."documento_ingreso_producto"
ADD CONSTRAINT "registro_doc_ingreso_cod_us_fkey"
FOREIGN KEY ("idusuario")
REFERENCES "public"."persona" ("id")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;
```

```
ALTER TABLE "public"."documento_ingreso_producto"
ADD CONSTRAINT "tipodocgastos_fk1"
FOREIGN KEY ("idtipodoc")
REFERENCES "public"."tipodoc_comprobantepago" ("id")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;
```

```
ALTER TABLE "public"."documento_ingreso_producto"
ADD CONSTRAINT "registro_doc_ingreso_tipo_mov_fkey"
FOREIGN KEY ("idtipotransaccion")
REFERENCES "public"."tipotransaccion" ("idtipo")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;
```

```
ALTER TABLE "public"."documento_ingreso_producto"
ADD CONSTRAINT "tbregistro_doc_ingreso_idtipomovmer_fkey"
FOREIGN KEY ("idtipomovmer")
REFERENCES "public"."tipo_movimiento_mercaderia" ("idtipomovmer")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;
```

```
ALTER TABLE "public"."documento_salida_producto"
ADD CONSTRAINT "registro_doc_salida_cod_almacen_fkey"
FOREIGN KEY ("idalmacen")
REFERENCES "public"."almacen" ("idalmacen")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;
```

```
ALTER TABLE "public"."documento_salida_producto"
ADD CONSTRAINT "registro_doc_salida_cod_us_fkey"
FOREIGN KEY ("idusuario")
REFERENCES "public"."persona" ("id")
```

```

ON DELETE NO ACTION
ON UPDATE NO ACTION
;

ALTER TABLE "public"."documento_salida_producto"
ADD CONSTRAINT "tipodocgastos_fk2"
FOREIGN KEY ("idtipodoc")
REFERENCES "public"."tipodoc_comprobantepago" ("id")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;

ALTER TABLE "public"."documento_salida_producto"
ADD CONSTRAINT "registro_doc_salida_tipo_mov_fkey"
FOREIGN KEY ("idtipotransaccion")
REFERENCES "public"."tipotransaccion" ("idtipo")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;

ALTER TABLE "public"."documento_salida_producto"
ADD CONSTRAINT "registro_doc_salida_idtipomovmer_fkey"
FOREIGN KEY ("idtipomovmer")
REFERENCES "public"."tipo_movimiento_mercaderia" ("idtipomovmer")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;

ALTER TABLE "public"."persona_rol"
ADD CONSTRAINT "tbpersona_fk"
FOREIGN KEY ("id_tbpersona")
REFERENCES "public"."persona" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."persona_rol"
ADD CONSTRAINT "tbrol_fk"
FOREIGN KEY ("id_tbrol")
REFERENCES "public"."rol" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."seriesdoc_emitir"

```

```

ADD CONSTRAINT "tbsucursalpcs_fk"
FOREIGN KEY ("id_tbsucursalpcs")
REFERENCES "public"."sucursal_pc_registrado" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."seriesdoc_emitir"
ADD CONSTRAINT "tbtipodoc_fk"
FOREIGN KEY ("id_tbtipodoc")
REFERENCES "public"."tipodoc_comprobantepago" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."stock_producto_almacen"
ADD CONSTRAINT "stockactualalmacen_cod_almacen_fkey"
FOREIGN KEY ("idalmacen")
REFERENCES "public"."almacen" ("idalmacen")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;

ALTER TABLE "public"."stock_producto_almacen"
ADD CONSTRAINT "stockactualalmacen_cod_prod_fkey"
FOREIGN KEY ("idprod")
REFERENCES "public"."producto" ("id")
ON DELETE NO ACTION
ON UPDATE NO ACTION
;

ALTER TABLE "public"."sucursal_pc_registrado"
ADD CONSTRAINT "tbsucursal_fk"
FOREIGN KEY ("id_tbsucursal")
REFERENCES "public"."sucursal" ("id")
MATCH FULL
ON DELETE CASCADE
ON UPDATE CASCADE
;

ALTER TABLE "public"."cuadre_caja"
ADD CONSTRAINT "Relationship53"
FOREIGN KEY ("id_tbpersona")
REFERENCES "public"."persona" ("id")
ON DELETE NO ACTION

```

```

        ON UPDATE NO ACTION
    ;

    ALTER TABLE "public"."producto"
    ADD CONSTRAINT "Relationship54"
    FOREIGN KEY ("idmarca")
    REFERENCES "public"."marca_producto" ("idmarca")
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
    ;

    ALTER TABLE "public"."docpago_ctacobrar_detalle_impreso"
    ADD CONSTRAINT "Relationship55"
    FOREIGN KEY ("id_tbdocpago_ctacobrar")
    REFERENCES "public"."docpago_ctacobrar" ("id")
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
    ;

-- Associating sequence with table column section section -----
-----

    ALTER SEQUENCE "public"."tbabono_ctapagar_iddoc_seq" OWNED BY
    "public"."ctapagar_abono"."iddoc"
    ;
    ALTER SEQUENCE "public"."tbalmacen_idalmacen_seq" OWNED BY
    "public"."almacen"."idalmacen"
    ;
    ALTER SEQUENCE "public"."tbcategoria_id_seq" OWNED BY
    "public"."categoria_producto"."id"
    ;
    ALTER SEQUENCE "public"."tbcomanda_id_seq" OWNED BY
    "public"."comanda_mesa"."id"
    ;
    ALTER SEQUENCE "public"."tbconfiguracion_id_seq" OWNED BY
    "public"."configuracion"."id"
    ;
    ALTER SEQUENCE "public"."tbctas_cobrar_idcta_seq" OWNED BY
    "public"."ctas_cobrar"."idcta"
    ;
    ALTER SEQUENCE "public"."tbctas_pagar_idcta_seq" OWNED BY
    "public"."cta_pagar"."idcta"
    ;
    ALTER SEQUENCE "public"."tbcuadre_caja_id_seq" OWNED BY
    "public"."cuadre_caja"."id"
    ;
    ALTER SEQUENCE "public"."tbdatosempresa_id_seq" OWNED BY

```

```

"public"."datosempresa"."id"
;
ALTER SEQUENCE "public"."tbdetabono_ctapagar_id_seq" OWNED BY
"public"."ctapagar_detalle_abono"."id"
;
ALTER SEQUENCE "public"."tbdetalle_comanda_id_seq" OWNED BY
"public"."comanda_detalle"."id"
;
ALTER SEQUENCE
"public"."tbdetalle_docpago_ctacobrar_impreso_id_seq" OWNED BY
"public"."docpago_ctacobrar_detalle_impreso"."id"
;
ALTER SEQUENCE "public"."tbdetalle_reg_ing_iddet_seq" OWNED BY
"public"."detalles_ingreso_productos"."iddet"
;
ALTER SEQUENCE "public"."tbdetalle_reg_salida_iddet_seq" OWNED BY
"public"."documento_salida_detalles"."iddet"
;
ALTER SEQUENCE "public"."tbdetpago_ctapagar_id_seq" OWNED BY
"public"."ctapagar_detpago"."id"
;
ALTER SEQUENCE "public"."tbdocpago_ctacobrar_id_seq" OWNED BY
"public"."docpago_ctacobrar"."id"
;
ALTER SEQUENCE "public"."tbestadomesa_id_seq" OWNED BY
"public"."estadomesa"."id"
;
ALTER SEQUENCE "public"."tbmarca_idmarca_seq" OWNED BY
"public"."marca_producto"."idmarca"
;
ALTER SEQUENCE "public"."tbmediopago_id_seq" OWNED BY
"public"."medio_pago"."id"
;
ALTER SEQUENCE "public"."tbmesa_id_seq" OWNED BY
"public"."mesa"."id"
;
ALTER SEQUENCE "public"."tbpersona_id_seq" OWNED BY
"public"."persona"."id"
;
ALTER SEQUENCE "public"."tbproducto_id_seq" OWNED BY
"public"."producto"."id"
;
ALTER SEQUENCE "public"."tbregistro_doc_ingreso_idreg_seq" OWNED
BY "public"."documento_ingreso_producto"."idreg"
;
ALTER SEQUENCE "public"."tbregistro_doc_salida_idreg_seq" OWNED
BY "public"."documento_salida_producto"."idreg"

```

```

;
ALTER SEQUENCE "public"."tbrol_id_seq" OWNED BY "public"."rol"."id"
;
ALTER SEQUENCE "public"."tbseccionmesa_idseccion_seq" OWNED BY
"public"."seccion_mesa"."idseccion"
;
ALTER SEQUENCE "public"."tbseriesdoc_emitir_id_seq" OWNED BY
"public"."seriesdoc_emitir"."id"
;
ALTER SEQUENCE "public"."tbstockactualalmacen_idstock_seq" OWNED
BY "public"."stock_producto_almacen"."idstock"
;
ALTER SEQUENCE "public"."tbsucursal_id_seq" OWNED BY
"public"."sucursal"."id"
;
ALTER SEQUENCE "public"."tbsucursalpcs_id_seq" OWNED BY
"public"."sucursal_pc_registrado"."id"
;
ALTER SEQUENCE "public"."tbtipo_documento_identidad_id_seq"
OWNED BY "public"."tipo_doc_identidad"."id"
;
ALTER SEQUENCE "public"."tbtipodoc_id_seq" OWNED BY
"public"."tipodoc_comprobantepago"."id"
;
ALTER SEQUENCE "public"."tbtiposervicio_compras_id_seq" OWNED BY
"public"."tiposervicio_compra"."id"
;
ALTER SEQUENCE "public"."tbtipotransaccion_idtipo_seq" OWNED BY
"public"."tipotransaccion"."idtipo"
;
ALTER SEQUENCE
"public"."tipo_movimiento_mercaderia_idtipomovmer_seq" OWNED BY
"public"."tipo_movimiento_mercaderia"."idtipomovmer"
;

-- Grant permissions section -----

GRANT "pg_read_all_settings" TO "pg_monitor"
;
GRANT "pg_read_all_stats" TO "pg_monitor"
;
GRANT "pg_stat_scan_tables" TO "pg_monitor"
;
GRANT CREATE ON SCHEMA "public" TO "postgres"
;
GRANT USAGE ON SCHEMA "public" TO "postgres"
;

```

```
GRANT CREATE ON SCHEMA "public" TO "public"  
;  
GRANT USAGE ON SCHEMA "public" TO "public"
```