

SISTEMA PUNTO DE VENTA (POS)

RESTOBAR “LA COMARCA”

Trabajo de Titulación para optar al
título de Técnico Universitario en
Informática.

Alumnos:

Álvaro Sebastián Godoy García

Elliott Alejandro Briones Ureta

Profesor Guía:

Carlos Alten López

Resumen

Keywords: Punto de Venta, PHP Laravel, Restaurant, HTML & Javascript, MySQL

El presente informe abarca la creación de un Sistema de punto de venta (P.O.S.) el cual tiene por objetivo satisfacer de manera segura y eficiente las clásicas problemáticas que se encuentran en un restaurante; a su vez también, satisfacer las necesidades del cliente, **Restobar La Comarca**, el cual se encuentra ubicado en la calle Maipú en la comuna de Quillota.

Con el fin de lograr un mejor desarrollo sin un mayor gasto económico, el backend del sistema propuesto fue desarrollado en PHP junto al framework Laravel en su versión 5.8 complementándolo con distintas librerías y paquetes que son ofrecidos por la misma comunidad del anterior mencionado. Por otro lado, el frontend fue desarrollado en Bootstrap 4. También, una base de datos alojada localmente con MySQL/MariaDB.

Capítulo 1: Encontraremos la información necesaria respecto al cliente; datos del dueño del local comercial, su tipo de organización, sus problemáticas, necesidades, requerimientos, y finalmente, la propuesta de sistema.

Capítulo 2: Aquí se hallará el detalle del sistema propuesto, el ambiente en el que será desarrollado, el ambiente físico necesario para desarrollarse, el detalle de la implementación de la base de datos con sus respectivas tablas relacionales, y finalmente el diseño y programación del sistema junto a los distintos niveles de acceso que posee cada tipo de usuario.

Índice de contenidos

Introducción.	1
CAPÍTULO 1.	2
ASPECTOS RELEVANTES DEL DISEÑO LÓGICO	2
1.1 Descripción de la organización.	3
1.2 Descripción de la situación actual.	4
1.2.1 Procesos	5
1.3 Problemas detectados.	6
1.4 Descripción Del Sistema Propuesto.	7
1.4.1 Objetivo Principal	7
1.4.2 Objetivos específicos	7
1.4.3 Beneficios del sistema	7
1.4.4 Descripción general de la solución propuesta	8
1.4.5 Estructura funcional del sistema	9
1.4.6 Descripción de fórmulas y cálculos empleados	10
1.4.7 Información que se maneja	12
1.4.7.1 Entidades de información	12
1.4.7.1-Figura 1: Modelo de proporcionados para Spatie/Laravel-permissions	
Fuente: Spatie/Laravel-Permissions Migración	12
1.4.8 Estructura de códigos	13
1.4.8-Figura 1: Tabla de estructura de código	
Fuente: Elaboración propia	14
1.4.9 Condicionamiento de diseño	15
CAPÍTULO 2.	
MEDIO AMBIENTE COMPUTACIONAL Y DESCRIPCIÓN DE ARCHIVOS.	16
2.1 Descripción del ambiente computacional	17
2.1.1 Configuración del sistema	17
2.1.2 Software utilizado	19
2.2 Descripción de archivos	22
Resumen tipos de datos	21
2.3 Diagrama de Menú	36
2.4 Listado de programas	37
Conclusión.	39
Bibliografía.	40

Índice de tablas

Tabla 1-1: Tabla de estructura de código	14
Tabla 2-1: Requisitos mínimos para el uso del sistema.	17
Tabla 2-2: Equipos utilizados para el desarrollo del sistema.	18
Tabla 2-3: Requisitos óptimos para el funcionamiento del servidor.	18
Tabla 2-4: Tipos de datos en MySQL	21
Tabla 2-5: Representa la tabla CHECKOUTS de la base de datos.	22
Tabla 2-6: Representa la tabla CATEGORIES de la base de datos.	23
Tabla 2-7: Representa la tabla BILLS de la base de datos.	23
Tabla 2-8: Representa la tabla de intersección entre BILLS y PAYMENTS de la base de datos.	24
Tabla 2-9: Representa la tabla MENUS de la base de datos.	24
Tabla 2-10: Representa la tabla INVOICES de la base de datos.	25
Tabla 2-11: Representa la tabla TABLES de la base de datos.	25
Tabla 2-12: Representa la tabla PRODUCTS de la base de datos.	26
Tabla 2-13: Representa la tabla de intersección entre MENUS y PRODUCTS de la base de datos.	27
Tabla 2-14: Representa la tabla de intersección entre MENUS y PREPARATIONS de la base de datos.	27
Tabla 2-15: Representa la tabla RAWMS de la base de datos.	28
Tabla 2-16: Representa la tabla ORDERS de la base de datos.	29
Tabla 2-17: Representa la tabla de intersección entre MENUS y ORDERS de la base de datos.	29
Tabla 2-18: Representa la tabla PAYMENTS de la base de datos.	30
Tabla 2-19: Representa la tabla PREPARATIONS de la base de datos.	30
Tabla 2-20: Representa la tabla de intersección entre PREPARATIONS y RAWMS de la base de datos.	31
Tabla 2-21: Representa la tabla PROMOTIONS de la base de datos.	31
Tabla 2-22: Representa la tabla TIPS de la base de datos.	32
Tabla 2-23: Representa la tabla PROVIDERS de la base de datos.	33
Tabla 2-24: Representa la tabla REPORTS de la base de datos.	34
Tabla 2-25: Representa la tabla USERS de la base de datos.	35
Tabla 2-26: Tabla representativa de los permisos de usuario.	35
Tabla 2-27: Listado de programas.	37

Índice de figuras

Figura 1-1: Imagen de referencia de comandas.	4
Figura 1-2: Sumatoria del precio de las órdenes.	10
Figura 1-3: Sumatoria de ingresos.	11
Figura 1-4: Sumatoria de egresos.	11
Figura 1-5: Sumatoria propinas.	11
Figura 1-6: Modelo de proporcionados para Spatie/Laravel-permissions.	12
Figura 1-7: Modelo Propuesto para el sistema.	13
Figura 2-1: Diagrama de menú.	36

Introducción

Satisfacer necesidades es el pilar fundamental de la tecnología en todo contexto posible. En el presente informe, se darán a conocer las problemáticas y necesidades existentes en el Restobar La Comarca y como la tecnología (desarrollo de software) brindará las soluciones pertinentes para lograr satisfacer las ya mencionadas.

En este local, desde su existencia hasta la fecha de este informe, ha carecido de soluciones tecnológicas para menguar sus problemas; desde pedidos que se pierden, mezclan y no se realizan hasta inclusive la pérdida de dinero por mala organización.

Hemos propuesto solucionar sus problemáticas y/o necesidades con un sistema que cumpla las funciones de un “POS” (Punto de venta, por su traducción del inglés¹), el cual se presentará y funcionará a través de una plataforma web no solo con el fin de ayudar a la administración del restaurante, sino también a su organización económica.

Este informe consta en: principalmente el análisis de problemáticas correspondientes al local comercial y las necesidades a satisfacer. Luego se realizará el diseño de la solución y todo lo que conlleva esta (tipo de solución, software y hardware a utilizar, entre otros), y finalmente los detalles en el modelamiento de bases de datos.

¹ Point of Sale: es un sistema que, en un establecimiento comercial, permite gestionar tareas relacionadas con las ventas

CAPÍTULO 1.
ASPECTOS RELEVANTES DEL DISEÑO LÓGICO

1.1 Descripción de la organización.

El presente sistema está pensado para el hoy conocido Restobar “La Comarca”, ubicado en Maipú #168, Quillota. Este local fue creado por Gustavo Rojas y su familia en el año 2011 inicialmente como un bar. El cual se dedicó únicamente a la venta de cervezas artesanales en el centro de la ciudad de Quillota desde su inicio hasta 2016.

Durante sus primeros años, el local funcionó con una patente de Cervezas, la cual únicamente le permitía vender la bebida anteriormente mencionada.

En los últimos 2 años, se ha debido adaptar a nuevos clientes que ya no solo buscan disfrutar únicamente una cerveza, sino también ir acompañado de sus amigos y/o familiares. Por lo tanto, ha sido necesaria la incorporación de una patente de restaurante y venta de alcoholes durante el último año; esta les permite funcionar hoy en día como Restobar “La Comarca”.

Algunos de sus servicios principales y en donde quieren destacar son:

- Venta de bebidas alcohólicas y tragos.
- Venta de comida durante el día y la noche.
- Brindar un buen servicio al cliente.

1.2 Descripción de la situación actual.

Al acercarse a la caja podemos encontrarnos con los siguientes escenarios:

- El restaurante cuenta con un sistema de *comandas*² para escribir y tomar los pedidos de cada mesa que se atiende en el local.
- En un cuaderno se encuentran corcheteadas las facturas de las compras de los insumos bebestibles y de las materias primas.

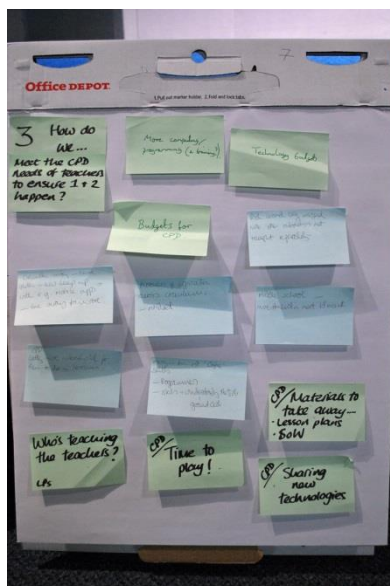


Figura 1-1: Imagen referencial.)

Fuente: <https://www.flickr.com/photos/15879717@N00/6240094596>

(CC BY-NC-SA)

- En una de las paredes de este local podemos encontrar una pequeña pizarra donde se encuentran las comandas adheridas y divididas en una cuadrícula según las mesas existentes y correspondientes a cada pedido.
- En otras ocasiones cuando llega un grupo grande de personas que desean consumir, se juntan dos mesas, realizando así pedidos a una sola cuenta.
- Si los clientes lo desean, pueden dejar el 10% opcional de su cuenta como propina.
- En variadas ocasiones cuando se requiere llamar a los clientes a consumir en el local, se aplican promociones y descuentos. Para esto el administrador de turno hace un descuento directamente en el resumen final de la cuenta. El dueño del local sólo permite un descuento o promoción, y no desea cambiar esa política.
- Cuando un cliente pregunta por precios, quien lo atiende, se dirige a una lista de productos en una hoja de cálculo Excel que posee sus valores actualizados. Si la

² Comanda: Es un soporte en papel donde se anota el pedido de un cliente.

hoja de cálculo no se encuentra disponible, se procede a leer la carta que posee los datos y precios de la última vez que fue impresa.

1.2.1 Procesos

Este local ejerce distintas funciones y procesos para llevar a cabo la atención al cliente y su organización, dentro de estos podemos destacar los más relevantes.

- ➔ Atención de una mesa: Cuando una mesa debe ser atendida, el mesero debe dirigirse a su posición y consultar qué es lo que el cliente desea consumir. El mesero deberá escribir todo en una comanda. Esta debe ser llevada a la cocina o bar y realizar una copia para la caja del local; será guardada y adjuntada a una posición de la cuadrícula que corresponde al número de mesa.
- ➔ Cambio de mesa: Cuando un cliente desea cambiarse de mesa, se transfieren las órdenes a una nueva cuadrícula (nueva mesa) y la anterior queda liberada.
- ➔ Propinas:
Cada cuenta debe tener solo un mesero asignado, por lo que la propina de esta corresponderá al mesero asociado.
- ➔ Cierre de Caja: Al final del día laboral quien está administrando en ese turno, debe hacer los cálculos y encuadramientos correspondientes para el día trabajado. Se calculan las propinas acumuladas para cada mesero y se pagan.

1.3 Problemas detectados.

- **Desconocimiento del stock de insumos y materias primas.** Se desconoce el abastecimiento de ciertos insumos y materias primas en el local, por lo tanto, el mesero no sabe con exactitud si estarán disponibles para ofrecer a un cliente.
- **Extenso tiempo de espera entre cocina/bar y la llegada de la orden.** Cuando un mesero anota una orden en una comanda, esta demora mucho tiempo en llegar a la barra/cocina y ser informada a la caja. Así, aumentando los tiempos de espera para los clientes.
- **Desconocimiento de los precios actualizados.** En diversas ocasiones los precios se encuentran sin actualizar, acción que el restaurante realiza en muy pocas ocasiones.
- **Pérdida de información e inconsistencia de dinero (vulnerabilidad del papel).** Los pedidos son anotados en papel, en caso que éste sea dañado o adulterado, se pierde toda la información y datos que estaban en él.
- **Inconsistencia de datos al cambiar la mesa de clientes.** Cuando hay cambios de mesas los pedidos tienden a ser confundidos y con ello los datos de estos pueden perderse y/o cruzarse.

1.4 Descripción Del Sistema Propuesto.

1.4.1 Objetivo Principal

Crear un sistema informático para el restobar “La Comarca” que permita mejorar su productividad realizando el cálculo de ingresos diarios (de propinas y pedidos) y gestionando mesas, productos, propinas, pedidos y usuarios para agilizar la atención de sus clientes.

1.4.2 Objetivos específicos

- Llevar un registro seguro y preciso de mesas, dinero, órdenes, propinas, entre otros.
- Llevar un control de precios y stock (registrando insumos y materias primas).
- Mantener informado a barra/cocina.
- Facilitar la creación y aplicación de promociones/descuentos.
- Facilitar la gestión de mesas, órdenes, insumos, entre otros.

1.4.3 Beneficios del sistema

El sistema propuesto permitirá que cada mesero pueda registrar pedidos en el sistema, así, agilizar y permitir cálculos rápidos de estos para su resumen final.

Al reducir inconsistencias, tener datos rápidos de los productos disponibles y lograr una gestión de mesas, ordenes, cuentas, entre otros; el cliente se verá beneficiado por los tiempos de atención que se le ofrecerán.

Gracias a una funcionalidad de propinas, los funcionarios se verán beneficiados por el rápido cálculo de los anteriormente mencionados.

1.4.4 Descripción general de la solución propuesta

Se realizará una aplicación web de ventas (**POS**), en el cual cada usuario ingresará con un usuario y contraseña para verificar que trabaja allí; validado esto, la aplicación permitirá administrar mesas, stock, propinas, entre otras. Además, deberá generar órdenes, las cuales deben ser comunicadas a la caja, cocina y/o barra. También permitirá imprimir un recibo con el resumen de cuenta para cada mesa. El sistema tiene por uno de sus objetivos agilizar tanto la atención al cliente como el movimiento de dinero en el local.

Según el tipo de usuario, se podrá acceder a distintas funcionalidades, el mesero por ejemplo podrá crear órdenes y generar estados de cada mesa, entre otros. Por otro lado el administrador será un usuario con privilegios superiores, el cual podrá gestionar la totalidad del sistema, exceptuando aquellos permisos exclusivos para desarrolladores y/o soporte de este.

1.4.5 Estructura funcional del sistema

A continuación, se describe la estructura funcional del sistema propuesto junto con sus entradas y salidas:

- **Mantenedor de usuarios:** Privilegio otorgado a un usuario *administrador*³ con el fin de agregar, eliminar, modificar y listar usuarios.
 - **Entradas:** Código de usuario, nombre y contraseña los cuales se almacenan en la base de datos.
 - **Salida:** Mensaje de confirmación.
- **Inicio de sesión:** El usuario ya registrado podrá iniciar sesión y utilizar el sistema en su totalidad según el rol que le corresponda.
 - **Entradas:** Código de usuario y contraseña, para el chequeo de su existencia.
 - **Salidas:** Sesión iniciada con el sistema listo para su uso.

³ Se le llama administrador al encargado del restaurante quien tiene acceso de mayor nivel en el sistema.

- **Mantenedor de mesas:** El usuario podrá agregar, listar, modificar y/o eliminar mesas en caso de que estas cambien.
 - **Entradas:** Mediante Interfaz de usuario podrán gestionar las mesas del sistema.
 - **Salidas:** Visualización de mesas que se encontrarán disponibles a partir de las entradas.
- **Mantenedor de órdenes:** Un usuario podrá crear, modificar, listar y eliminar órdenes.
 - **Entradas:** Código de mesa, Código del producto y cantidad (en caso de modificar y crear).
 - **Salidas:** Confirmación visual de la orden para posterior entrega correspondiente, ya sea en cocina o bar.

- **Pago de la cuenta:** Permite imprimir el recibo de una cuenta para ser entregada al cliente.
 - **Entrada:** Código de mesa.
 - **Salidas:** Recibo (Detalle de cuenta, monto total a pagar y propina opcional).
- **Pago de propinas:** Permite que al momento de cerrar la caja se pueda calcular y pagar las propinas a cada mesero.
 - **Entradas:** Código de usuario.
 - **Salidas:** Monto total de propinas a pagar.
- **Mantenedor de stock:** Permite llevar un registro de los insumos y materias primas que hay actualmente en el restaurante (agregar, modificar o eliminar productos).
 - **Entradas:** Nombre, descripción, cantidad, precio.
 - **Salidas:** Mensaje de confirmación y actualización de los datos

1.4.6 Descripción de fórmulas y cálculos empleados

- De momento se tendrán fórmulas simples.
 - Sumatoria de órdenes o pedidos (precios) donde, **n** corresponde a la cantidad de órdenes, **x** corresponde al precio de la orden e **i** actúa como subíndice tomando los valores desde **1** hasta llegar a **n**.

$$\sum_{i=1}^n x_i$$

Figura 1-2: Sumatoria del precio de las órdenes.

Fuente: Elaboración propia.

- Sumatoria de ingresos donde, **n** corresponde a la cantidad de órdenes, **x** corresponde al valor del ingreso e **i** actúa como subíndice tomando los valores desde **1** hasta llegar a **n**.

$$\sum_{i=1}^n x_i$$

Figura 1-3: Sumatoria de ingresos.

Fuente: Elaboración propia

- Sumatoria de egresos donde, **n** corresponde a la cantidad de egresos, **x** corresponde al valor del egreso e **i** actúa como subíndice tomando los valores desde **1** hasta llegar a **n**.

$$\sum_{i=1}^n x_i$$

Figura 1-4: Sumatoria de egresos.

Fuente: Elaboración propia

- Cálculo de IVA.

$$\blacksquare \text{ valorIva} = \text{precio} * 1.19$$

- Cálculo de Propinas por mesero.

$$\blacksquare \text{ propina} = \text{totalCuenta} * 0.10$$

- Sumatorias de Propinas por mesero donde, **n** corresponde a la cantidad de propinas, **x** corresponde al valor de la propina e **i** actúa como subíndice tomando los valores de **1** a **n**.

$$\sum_{i=1}^n x_i$$

Figura 1-5: Sumatoria propinas.

Fuente: Elaboración propia

1.4.7 Información que se maneja

1.4.7.1 Entidades de información

Se poseen 2 modelos para el manejo de datos, uno que corresponde netamente a los datos como tal, y el otro modelo el cual corresponde a los permisos del sistema. Los cuales están modelados de la siguiente manera:

- Modelo de Permisos de Usuarios y Roles propuestos para el Sistema. Proporcionado para funcionar de una manera compatible con Spatie/Permissions⁴
- Modelo de Datos del Sistema propuesto para el desarrollo del sistema.

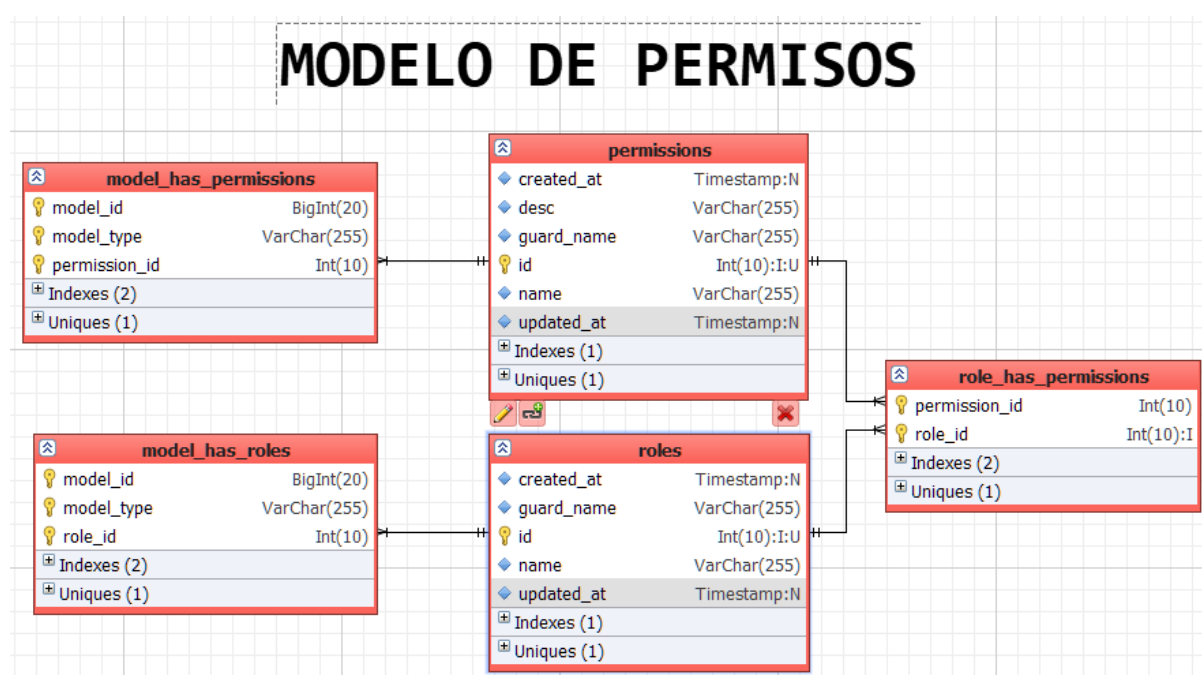


Figura 1-6: Modelo de proporcionados para Spatie/Laravel-permissions.

Fuente: Elaboración propia.

⁴ Spatie - Administración de permisos y roles en Laravel Framework. (2018). spatie/laravel-permission. Recuperado de <https://github.com/spatie/laravel-permission>

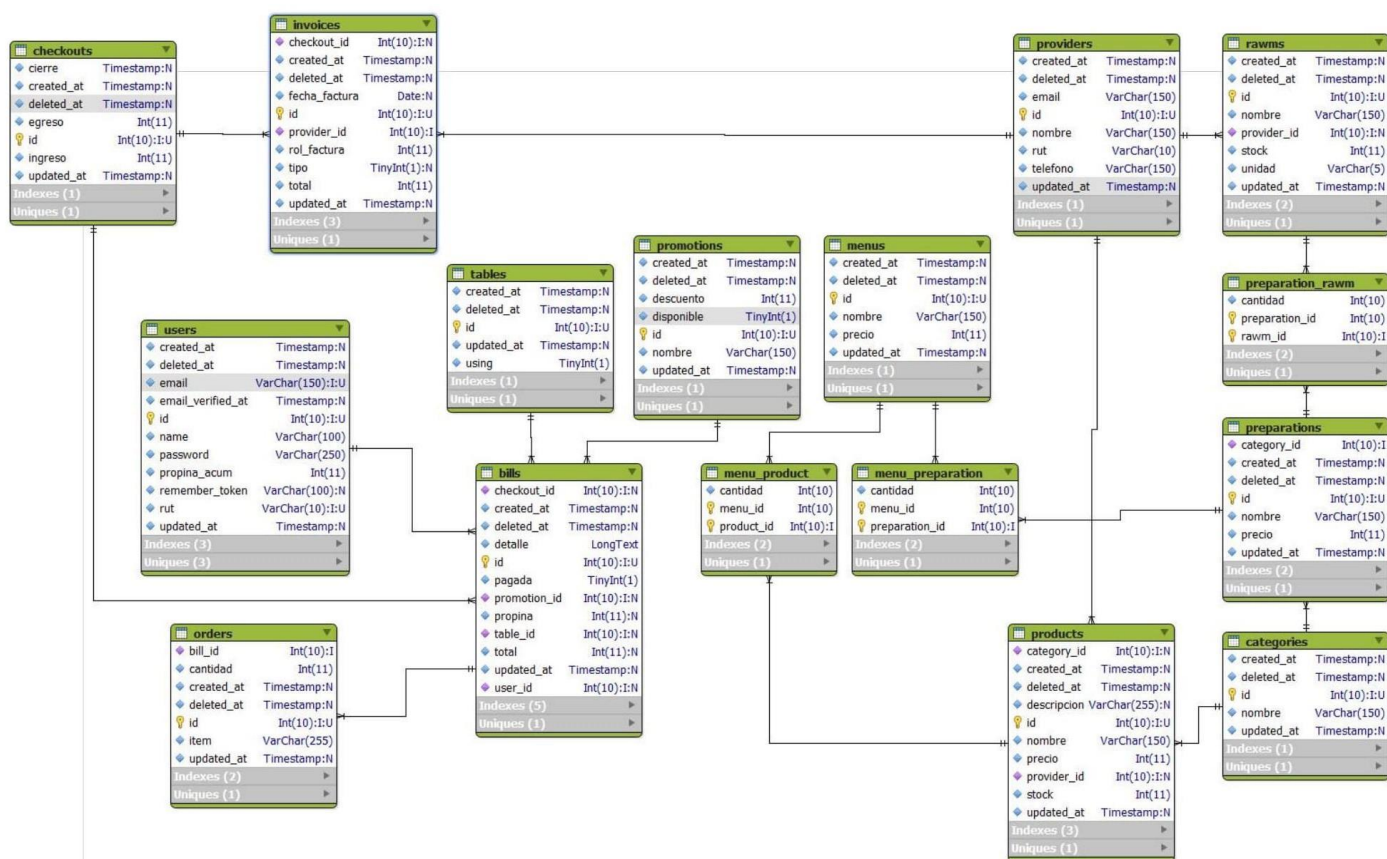


Figura 1-7: Modelo Propuesto para el sistema.

Fuente: Elaboración propia.

1.4.8 Estructura de códigos

A continuación, se presentan las tablas de la base de datos con su respectiva clave primaria y una breve descripción de estas.

TABLA	CLAVES (PK/FK)	TIPO	OBSERVACIÓN
checkouts	id	INT	Corresponde a una Caja
categories	id	INT	AutoIncrementable, corresponde a la id de una categoría
bills	id	INT	AutoIncrementable, corresponde a una cuenta

invoices	id	INT	AutoIncrementable, corresponde a una factura
products	id	INT	AutoIncrementable, corresponde a un producto
menus	id	INT	AutoIncrementable, corresponde a un menú
tables	id	INT	Número Correlativo, corresponde a una mesa
rawms	id	INT	Número Correlativo, corresponde a una materia prima
orders	id	INT	AutoIncrementable, corresponde a una orden
payments	id	INT	AutoIncrementable, corresponde a un pago
preparations	id	INT	Número Correlativo, corresponde a una preparación
promotions	id	INT	Número Correlativo, corresponde a una promoción.
tips	id	INT	Propinas
providers	id	INT	Id de proveedor
reports	id	INT	Id de reporte de caja
users	id	INT	Corresponde al id de un usuario

Tabla 1-1: Tabla de estructura de código

Fuente: Elaboración propia

1.4.9 Condicionamiento de diseño

Para el sistema propuesto se utilizará una computadora en modalidad de servidor, que funcione como alojamiento local del sistema web, puesto que éste no estará hospedado en internet.

El equipo que sirva este sistema, debe poseer por lo menos un procesador dual core y 4GB de RAM para condiciones mínimas (aproximadamente 8 a 15 usuarios utilizando el sistema). Sin embargo, todo dependerá del alcance que el cliente desee darle.

El lenguaje en el que se desarrollará este sistema será PHP Junto a Laravel Framework, una base de datos en MySQL/MariaDB y añadiendo librerías y complementos que aporten al desarrollo de este proyecto.

El usuario no tendrá un acceso físico ni virtual a este equipo⁵, si no, a través del sistema propuesto.

Para el desarrollo de la aplicación se utilizará una computadora con mínimo Procesador *Intel Core i3* y 6GB de RAM.

⁵ Accesos virtuales tales como shell, ssh, telnet, entre otros. Y físicamente con el hardware del servidor.

CAPÍTULO 2.
MEDIO AMBIENTE COMPUTACIONAL
Y DESCRIPCIÓN DE ARCHIVOS.

2.1 Descripción del ambiente computacional

2.1.1 Configuración del sistema

El sistema requerirá de distintos componentes para su funcionamiento, por el lado del usuario y por el lado del servidor.

Usuario (Requisitos mínimos)	
Procesador	Intel Pentium Dual-Core 1.30Ghz en adelante
Tipo y capacidad de almacenamiento	- 2GB RAM - 32 GB Almacenamiento
Tipo y número de pantallas e impresoras.	- 1 monitor 1280x720 - 3 Impresoras térmicas
Sistema Operativo	- Windows 7 o Superior - Linux Ubuntu/Fedora/Centos etc o superior - MacOS 10 o Superior
Periféricos	- Mouse - Teclado - Audio (Notificaciones)

Tabla 2-1: Requisitos mínimos para el uso del sistema.

Fuente: Elaboración propia

Desarrolladores	
Procesador	- Intel Core i3 4005U - Intel Core i5-7300HQ
Tipo y capacidad de almacenamiento	- 6GB RAM/ 12GBRAM - 1TB/512GB de almacenamiento
Sistema Operativo	- Windows 10 - Linux Ubuntu.
Periféricos	- Mouse - Teclado - Audio

Tabla 2-2: Equipos utilizados para el desarrollo del sistema.

Fuente: Elaboración propia

Servidor (Sugerido ⁶)	
Procesador	Intel Celeron N3160 1.6Ghz O Superior
Tipo y capacidad de almacenamiento	HDD 500 GB 5400 RPM SATA 3 - Mínimo 4GB RAM - 500 GB HDD - 5GB Espacio mínimo requerido para software.
Sistema Operativo	- Linux Debian / Ubuntu Server
Otros	- Ethernet - I/O USB - SSH / SFTP

Tabla 2-3: Requisitos óptimos para el funcionamiento del servidor.

Fuente: Elaboración propia

⁶ Sugerencia basada en minipc cotizado <https://www.pcfactory.cl/producto/28254-gear-mini-pc-nuc-1i-celeron-n3160-4gb-500gb>

2.1.3 Software utilizado

Listado de software utilizado para el desarrollo del sistema POS propuesto.

- Sistema operativo: Linux Debian 9 64bits.
- Base de datos: MySQL/MariaDB 10.4.
 - MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base de datos de código abierto más popular del mundo. MariaDB es una bifurcación en modalidad Open Source.
- Editor de textos: Visual Studio Code.
 - Abreviado como VSCode, es un editor de texto con soporte incorporado para JavaScript, TypeScript y Node.js, tiene un rico ecosistema de extensiones para otros lenguajes (como C ++, C #, Java, Python, **PHP**, Go) y tiempos de ejecución (como .NET y Unity).
- Laravel 5.8.
 - Framework de PHP, muy utilizado en la última década gracias a los avances de PHP5 y PHP7. Laravel intenta eliminar el “*dolor*” del desarrollo al facilitar las tareas comunes utilizadas en la mayoría de los proyectos web, evitando el código spaghetti y permitiendo una fácil mantención.
- Javascript y JQuery.
 - Javascript: Es un lenguaje de programación orientado a objetos. Principalmente utilizado en Sitios y Aplicaciones web debido a su fácil manejo y manipulación de objetos DOM/HTML.
 - JQuery: jQuery es una biblioteca de JavaScript rápida, pequeña y rica en funciones. Hace que cosas como el desplazamiento y la manipulación de documentos HTML, el manejo de eventos, la animación y Ajax sean mucho más simples con una API fácil de usar que funciona en una multitud de navegadores.
- HTML5 - CSS3.
 - HTML5: Última versión de HTML. Contiene un conjunto más amplio de tecnologías que permite a los sitios Web y a las aplicaciones ser más diversas y de gran alcance.
 - CSS3: Última evolución del lenguaje de CSS. Trae consigo muchas novedades altamente esperadas.
- Framework Bootstrap.

- Bootstrap es un kit de herramientas de código abierto para desarrollar con HTML, CSS y JS; Permitiendo crear proyectos receptivos para dispositivos móviles en la web con la biblioteca de componentes front-end más popular del mundo.
- PHP7.
 - Es un lenguaje de programación de propósito general de código del lado del servidor originalmente diseñado para el pre procesamiento de texto plano
- Composer.
 - Es un administrador de dependencias para PHP que nos permite descargar paquetes desde un repositorio para agregarlo a nuestro proyecto.
- Git.
 - Es un software de control de versiones que nos permiten hacer todo tipo de modificaciones en nuestro código, y hacer que sea más fácil la administración de las distintas versiones de cada producto desarrollado.
- PhpMyAdmin.
 - PHPMyAdmin es un software de código abierto, diseñado para manejar la administración y gestión de bases de datos MySQL a través de una interfaz gráfica de usuario.
- Complementos y Librerías de Laravel
 - Spatie/laravel-permissions
 - Implementado para laravel, “laravel-permissions” corresponde a un paquete creado por Spatie que le permite administrar los permisos y roles de usuario en una base de datos.
 - LaravelCollective/HTML
 - Es un conjunto de elementos para Formularios HTML tales como Labels, Botones, Listas de Formulario, Radio Buttons, Checkboxes, etc. Por otro lado cuenta con Anotaciones, Remote SSH, Iron Queue y Command Bus para otro tipo de tareas.
 - Laracasts/Generators:
 - Complemento para generación de migraciones y tablas de intersección.
 - freshworkstudio/ChileanBundle:
 - Complemento para validación de RUT, Verificación y formateo.

2.2 Descripción de archivos

Como ya se mencionó, se utilizará una base de datos en MySQL. Cada tabla de la base de datos y sus relaciones han sido representadas de la siguiente forma.

Resumen tipos de datos

Distintos tipos de datos existentes en MySQL, siendo los más importantes y lo que se darán uso, los siguientes:

Tipo	Descripción
VARCHAR(n)	Carácter de largo variable, de máximo n.
CHAR(n)	Carácter de largo n.
DATE	Válido para almacenar una fecha con año, mes y día. Formato: 'AAAA-MM-DD'.
TIMESTAMP	Valores en el rango de '1970-01-01 00:00:01' UTC a '2038-01-19 03:14:07' UTC.
SMALLINT	Numérico, ocupación de 2 bytes con valores entre -32768 y 32767 o entre 0 y 65535.
MEDIUMINT	Numérico, ocupación de 3 bytes con valores entre -8388608 y 8388607 o entre 0 y 16777215.
TINYINT	Numérico, ocupación de 1 bytes con valores entre -128 y 127 o entre 0 y 255.
BOOLEAN/BIT	Número entero con valor 0 o 1.
INT	Sin signos entre 0 y 4294967295.

Tabla 2-4: Tipos de datos en MySQL

Fuente: Elaboración propia

TABLA	CHECKOUTS	
CLAVES	PK: ID FK: No posee clave foránea.	
DESCRIPCIÓN	Corresponde a las cajas registradoras disponibles en el local.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Indica el número correspondiente a la caja que se están utilizando en el sistema. Autoincrementable.
apertura	TIMESTAMP	Corresponde a la hora en que se abrió la caja para ser operada.
cierre	TIMESTAMP	Corresponde a la hora en que se cerró la caja. * Al cerrarse la caja, se guarda un reporte de los movimientos. * Al abrirse la caja, se sobrescriben las horas de apertura y cierre.

Tabla 2-5: Representa la tabla CHECKOUTS de la base de datos.

Fuente: Elaboración propia

TABLA	CATEGORIES	
CLAVES	PK: ID FK: No posee clave foránea.	
DESCRIPCIÓN	Corresponde al tipo de preparación que se ofrece en el lugar.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Indica un identificador único para una categoría.
NOMBRE	VARCHAR(150)	Corresponde al nombre de la categoría.

Tabla 2-6: Representa la tabla CATEGORIES de la base de datos.

Fuente: Elaboración propia.

TABLA	BILLS	
CLAVES	PK: ID FK: USER_ID referencia a la tabla USERS	
DESCRIPCIÓN	Corresponde a la cuenta de un cliente, está asociada a un mesero, mesa o grupo de mesas.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Corresponde a un número de la cuenta. Autoincrementable
TOTAL	INT(11)	Corresponde al total de la cuenta, es calculado en base a las órdenes.
FECHA	TIMESTAMP	Indica a la hora que se registró el pedido en el sistema.
PAGADA	TINYINT	Indica si la cuenta fue paga en su totalidad o no. (0 o 1)
USER_ID	INT(10)	Corresponde al usuario (funcionario) que atendió y/o está atendiendo esa cuenta.

Tabla 2-7: Representa la tabla BILLS de la base de datos.

Fuente: Elaboración propia

TABLA	BILL_PAYMENT	
CLAVES	PK: BILL_ID + PAYMENT_ID FK: BILL_ID referencia a la tabla BILLS PAYMENT_ID referencia a la tabla PAYMENTS	
DESCRIPCIÓN	Corresponde a los pagos que recibe esa cuenta. Útil en el caso de que se pague una cuenta por separado.	
CAMPO	TIPO	DESCRIPCIÓN
BILL_ID	INT(10)	Corresponde a la clave foránea cuenta.
PAYMENT_ID	INT(10)	Corresponde a la id del pago, autoincrementable.

Tabla 2-8: Representa la tabla de intersección entre BILLS y PAYMENTS de la base de datos.

Fuente: Elaboración propia

TABLA	MENUS	
CLAVES	PK: ID FK: No posee clave foránea.	
DESCRIPCIÓN	Corresponde a un conjunto de Insumos y/o Preparaciones.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Indica un identificador único.
NOMBRE	VARCHAR(150)	Nombre del Menú.
PRECIO	INT(11)	Precio del Menú.

Tabla 2-9: Representa la tabla MENUS de la base de datos.

Fuente: Elaboración propia

TABLA	INVOICES	
CLAVES	PK: ID FK: PROVIDER_ID referencia a la tabla PROVIDERS REPORT_ID referencia a la tabla REPORTS	
DESCRIPCIÓN	Corresponde a las compras facturadas hechas por el local.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Corresponde a un número identificador. Autoincrementable.
ROL_FACTURA	INT(11)	Indica el rol de cada factura
FECHA	TIMESTAMP	Corresponde a la fecha indicada en la factura
TOTAL	INT(11)	Indica el monto de la factura.
PROVIDER_ID	INT(10)	Es el identificador de un proveedor que fue registrado en el sistema.
REPORT_ID	INT(10)	Corresponde al id del reporte en el cual quedó registrado, con fines de conocer los egresos en una fecha indicada por el usuario.

Tabla 2-10: Representa la tabla INVOICES de la base de datos.

Fuente: Elaboración propia

TABLA	TABLES	
CLAVES	PK: ID FK: No posee clave foránea.	
DESCRIPCIÓN	Una mesa disponible en el lugar.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Correlativo a la mesa.
USING	TINYINT	Indica si está en uso por algún cliente. (1 o 0)

Tabla 2-11: Representa la tabla TABLES de la base de datos.

Fuente: Elaboración propia

TABLA	PRODUCTS	
CLAVES	PK: ID FK: PROVIDER_ID referencia a la tabla PROVIDERS CATEGORY_ID referencia a la tabla CATEGORIES	
DESCRIPCIÓN	Es un producto que bien puede estar envasado y no afecta al stock de Materias Primas.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Indica un identificador único.
NOMBRE	VARCHAR(150)	Nombre del insumo registrado
PRECIO	INT(11)	Precio de venta del insumo
STOCK	INT(11)	Cantidad disponible
PROVIDER_ID	INT(10)	Corresponde al identificador del proveedor de la materia prima indicada.
CATEGORY_ID	INT(10)	Indica la categoría de un insumo

Tabla 2-12: Representa la tabla PRODUCTS de la base de datos.

Fuente: Elaboración propia

TABLA	MENU_PRODUCT	
CLAVES	PK: MENU_ID + PRODUCT_ID FK: MENU_ID referencia a la tabla MENUS PRDUCT_ID referencia a la tabla PRODUCTS	
DESCRIPCIÓN	Corresponde a los insumos que existen en un menú ofrecido.	
CAMPO	TIPO	DESCRIPCIÓN
MENU_ID	INT(10)	Corresponde al menu a vender
PRODUCT_ID	INT(10)	Corresponde a los insumos contenidos en el menú.
QTY	INT(10)	Indica la cantidad que es ofrecida en el menú.

Tabla 2-13: Representa la tabla de intersección entre MENUS y PRODUCTS de la base de datos.

Fuente: Elaboración propia

TABLA	MENU_PREPARATION	
CLAVES	PK: MENU_ID + PREPARATION_ID FK: MENU_ID referencia a la tabla MENUS PREPARATION_ID referencia a la tabla PREPARATIONS	
DESCRIPCIÓN	Corresponde a las preparaciones que existe en un menú	
CAMPO	TIPO	DESCRIPCIÓN
MENU_ID	INT(10)	Corresponde al menu a vender
PREPARATION_ID	INT(10)	Corresponde a las preparaciones

Tabla 2-14: Representa la tabla de intersección entre MENUS y PREPARATIONS de la base de datos.

Fuente: Elaboración propia

TABLA	RAWMS	
CLAVES	PK: ID FK: PROVIDER_ID referencia a la tabla PROVIDERS	
DESCRIPCIÓN	Materias primas que componen una preparación.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Correlativo a la materia prima.
NOMBRE	VARCHAR(150)	Nombre de la materia prima.
STOCK	INT(11)	Cantidad disponible
UNIDAD	VARCHAR(5)	Indica la unidad de medida de la materia prima (KG,GR,L).
PROVIDER_ID	INT(10)	Id del proveedor de la materia prima

Tabla 2-15: Representa la tabla RAWMS de la base de datos.

Fuente: Elaboración propia

TABLA	ORDERS	
CLAVES	PK: ID FK: BILL_ID desde BILLS PROMOTION_ID referencia a la tabla PROMOTIONS	
DESCRIPCIÓN	Reemplaza a la antigua comanda utilizada por los meseros.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Correlativo autoincrementable
FECHA	TIMESTAMP	corresponde a la fecha en que se tomó la orden
ESTADO	VARCHAR(15)	Indicará si la orden se encuentra activa, suspendida o cancelada por el cliente.
BILL_ID	INT(10)	Apunta hacia la cuenta donde se está registrando la orden
PROMOTION_ID	INT(10)	Indica si la orden posee alguna promoción.

Tabla 2-16: Representa la tabla ORDERS de la base de datos.

Fuente: Elaboración propia

TABLAS	MENU_ORDER	
CLAVES	PK: ORDER_ID + MENU_ID FK: ORDER_ID referencia a la tabla ORDERS MENU_ID referencia a la tabla MENUS	
DESCRIPCIÓN	Corresponden a los menu que hay en una orden	
CAMPO	TIPO	DESCRIPCIÓN
ORDER_ID	INT(10)	Corresponde a la id de la orden registrada.
MENU_ID	INT(10)	Corresponde al menu a vender

Tabla 2-17: Representa la tabla de intersección entre MENUS y ORDERS de la base de datos.

Fuente: Elaboración propia

TABLAS	PAYMENTS	
CLAVES	PK: ID FK: CHECKOUT_ID referencia a la tabla CHECKOUTS REPORT_ID referencia a la tabla REPORTS	
DESCRIPCIÓN	Corresponde a un pago, pueden ser 1 o muchos.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Identificador autoincrementable
TOTAL	INT(10)	Monto pagado.
TIPO	VARCHAR(15)	Si es efectivo, crédito o débito.
DATE	TIMESTAMP	Fecha en que se hizo el pago.
CHECKOUT_ID	INT(10)	La caja donde se realizó el pago.
REPORT_ID	INT(10)	Donde será registrado para su reporte.

Tabla 2-18: Representa la tabla PAYMENTS de la base de datos.

Fuente: Elaboración propia

TABLAS	PREPARATIONS	
CLAVES	PK: ID FK: CATEGORY_ID referencia a la tabla CATEGORIES	
DESCRIPCIÓN	Corresponde a un pago, pueden ser 1 o muchos.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Id de la preparación
NOMBRE	VARCHAR(150)	Nombre de la preparación
PRECIO	INT(11)	Precio de la preparación
CATEGORY_ID	INT(10)	Categoría de la preparación

Tabla 2-19: Representa la tabla PREPARATIONS de la base de datos.

Fuente: Elaboración propia

TABLAS	PREPARATION_RAWM	
CLAVES	PK: PREPARATION_ID + RAWM_ID FK: PREPARATION_ID referencia a la tabla PREPARATIONS RAWM_ID referencia a la tabla RAWMS	
DESCRIPCIÓN	Son las materias primas que son utilizadas en la elaboración de una preparación.	
CAMPO	TIPO	DESCRIPCIÓN
PREPARATION_ID	INT(10)	Corresponde al id de la preparación.
RAWM_ID	INT(10)	Corresponde a la materia prima utilizada.
CANTIDAD	INT(11)	Cantidad a utilizar.

Tabla 2-20: Representa la tabla de intersección entre PREPARATIONS y RAWMS de la base de datos.

Fuente: Elaboración propia

TABLAS	PROMOTIONS	
CLAVES	PK: ID FK: No posee clave foránea.	
DESCRIPCIÓN	Indica una promoción que puede ser ofrecida a un cliente.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Correlativo autoincrementable.
NOMBRE	VARCHAR(150)	Nombre de la promoción aplicable a la orden.
DESCUENTO	INT(11)	Porcentaje de descuento.
DISPONIBLE	TINYINT	Indica si se encuentra disponible para su uso

Tabla 2-21: Representa la tabla PROMOTIONS de la base de datos.

Fuente: Elaboración propia

TABLAS	TIPS	
CLAVES	PK: ID FK: USER_ID desde USERS BILL_ID referencia a la tabla BILLS CHECKOUT_ID referencia a la tabla CHECKOUTS	
DESCRIPCIÓN	Corresponde a una propina registrada desde una cuenta.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Correlativo autoincrementable.
MONTO	INT(11)	Monto de la propina registrada.
USER_ID	INT(10)	Funcionario que recibe propina
BILL_ID	INT(10)	Cuenta de origen.
CHECKOUT_ID	INT(10)	Donde fue recibida.

Tabla 2-22: Representa la tabla TIPS de la base de datos.

Fuente: Elaboración propia

TABLAS	PROVIDERS	
CLAVES	PK: ID FK: No posee clave foránea.	
DESCRIPCIÓN	Indica a un proveedor del local comercial, registrado en una base de datos.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Correlativo autoincrementable.
NOMBRE	VARCHAR(150)	Nombre del proveedor
RUT	VARCHAR(10)	Rol único Tributario
TELEFONO	VARCHAR(150)	Contacto
EMAIL	VARCHAR(150)	Contacto

Tabla 2-23: Representa la tabla PROVIDERS de la base de datos.

Fuente: Elaboración propia

TABLAS	REPORTS ⁷	
CLAVES	PK: ID FK: No posee clave foránea.	
DESCRIPCIÓN	Indica un reporte de ingresos y egresos.	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Correlativo autoincrementable.
FECHA	TIMESTAMP	Indica fecha en que fue generado el reporte
INGRESOS	INT(11)	Muestra resumen de ingresos
EGRESOS	INT(11)	Muestra resumen de egresos
APERTURA	TIMESTAMP	A qué hora fue abierta la caja.
CIERRE	TIMESTAMP	A qué hora fuera cerrada la caja.

Tabla 2-24: Representa la tabla REPORTS de la base de datos.

Fuente: Elaboración propia

⁷ Los resúmenes son generados cuando se cierra una caja, considerando su fecha de apertura y su fecha de cierre. Se adquiere información desde Pagos y Facturas.

TABLAS	USERS	
CLAVES	PK: ID FK: No posee clave foránea.	
DESCRIPCIÓN	Corresponde a un usuario del sistema	
CAMPO	TIPO	DESCRIPCIÓN
ID	INT(10)	Correlativo autoincrementable.
PASSWORD	VARCHAR(250)	Clave alfanumérica, se almacena como un hash
NAME	VARCHAR(100)	Funcionario que recibe propina
RUT	VARCHAR(10)	Rol Único, usado para iniciar sesión.
EMAIL	VARCHAR(150)	Corresponde al correo del usuario

Tabla 2-25: Representa la tabla USERS de la base de datos.

Fuente: Elaboración propia

Permisos de usuarios (id correspondientes):

ID	ROL	DESCRIPCIÓN
1	SUPERADMIN	Rol designado únicamente a los desarrolladores del sistema y/o quienes están calificados para hacer soporte y mantenimiento de este.
2	ADMINISTRADOR	Asignado a los administradores del sistema, quienes tienen privilegios para manejar la totalidad del sistema, exceptuando aquellos que son únicamente para el rol SUPERADMIN.
3	CAJERO	Posee permisos para administrar lo relacionado a una caja (movimientos de dinero, entre otros).
4	MESERO	Permisos para crear, ver y eliminar órdenes, mesas entre otros.

Tabla 2-26: Tabla representativa de los permisos de usuario.

Fuente: Elaboración propia

2.3 Diagrama de Menú

El siguiente esquema representa la barra de menú del sistema, por el cual se podrá navegar entre las distintas funcionalidades que este posee.

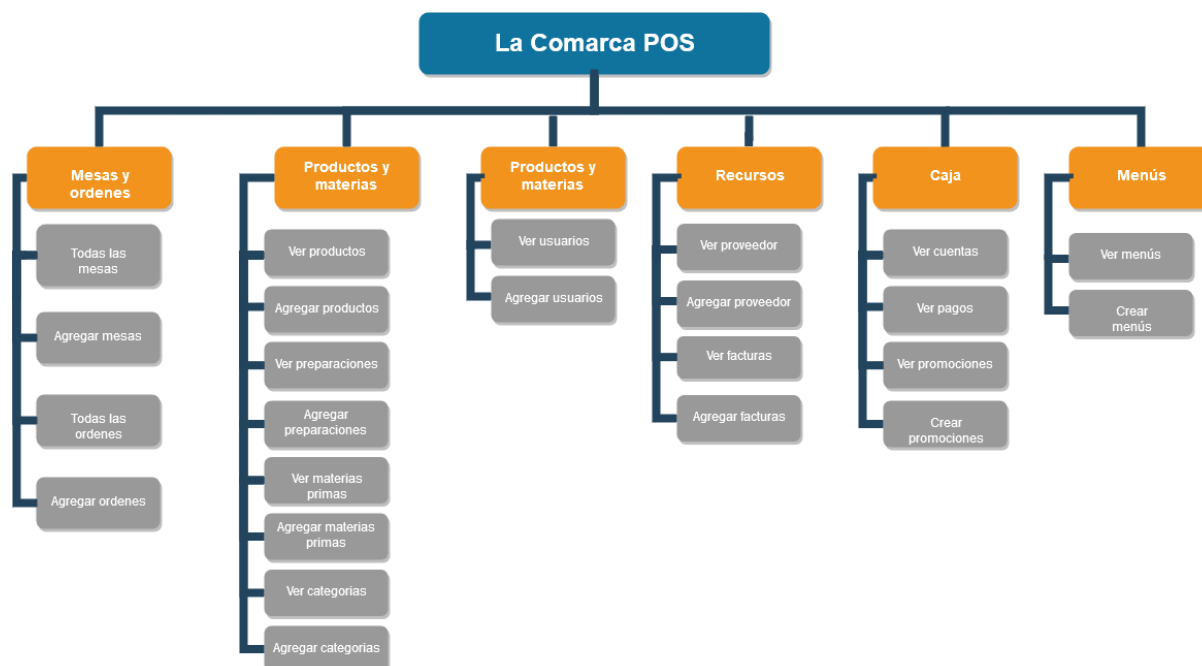


Figura 2-1: Diagrama de menú.

Fuente: Elaboración propia

2.4 Listado de programas

Presentación de los programas del sistema, en otras palabras las funcionalidades que este puede realizar.

	Nombre del programa	Descripción del programa
1.-	Inicio de sesión	Le permite al usuario ingresar al sistema con su RUT y contraseña.
2.-	Registrar usuario	Permite registrar un usuario en la base de datos sin roles ni permisos hasta que un administrador se los asigna en el gestor de usuarios.
3.-	Recuperar contraseña o Cambiar de Contraseña.	Permite recuperar la contraseña a través de correo electrónico y/o cambiarla desde el perfil del usuario.
4.-	Gestor de usuarios	Permitirá crear, modificar y/o eliminar usuarios del sistema.
5.-	Mantener de Productos/Insumos	Mantenedor que permite crear, modificar y/o eliminar productos del sistema.
6.-	Mantenedor de Preparaciones	Mantenedor que permite crear, modificar y/o eliminar preparaciones basadas en materias primas.
7.-	Mantenedor de Menús	Mantenedor que permite crear, modificar y/o eliminar menús basados en preparaciones y/o productos
8.-	Mantenedor de Materias Primas	Mantenedor que permite crear, modificar y/o eliminar materias primas del sistema.
9.-	Mantenedor de Promociones	Permite crear, modificar y/o eliminar promociones del sistema, las cuales se aplicaran a una cuenta.
10.-	Mantenedor de Propinas	Mantenedor de propinas para gestionar pagos a trabajadores. (Agregar propinas desde una cuenta o pagarlas antes del cierre de caja).
11.-	Mantenedor de Ordenes	Permitirá operaciones básicas tales como crear, eliminar y/o modificar ordenes de clientes.
12.-	Mantenedor de Cuentas	Mantenedor que permite crear (abrir), modificar (agregar ordenes o cerrar cuentas) y/o eliminar cuentas (borrado lógico/físico).
13.-	Imprimir una cuenta (estado, cierre de cuenta, cuenta final y/o detalle de cuenta)	Funcionalidad que permite la impresión física de una cuenta en papel.
14.-	Generar reportes diarios de caja.	Se generan al momento de cerrar una caja, crea un reporte sobre como funcionó el movimiento de dinero.
15.-	Mantenedor de Facturas	Se pueden registrar facturas en el sistema para ser contabilizadas cómo egresos en los reportes diarios mencionados anteriormente en esta tabla.
16.-	Mantenedor de Mesas	Este mantenedor permite organizar las mesas disponibles en el local comercial. Agregar, modificar y/o eliminar una mesa. También podrá hacer intercambio de mesas.
17.-	Mantenedor de Categorías	Cada producto o preparación tendrá su categoría asociada, este mantenedor podrá agregar,

18.-		modificar y/o eliminar una categoría del sistema.
	Pago de Cuentas	Un usuario con permisos específicos podrá realizar el pago de una cuenta en caja. Se imprimirá un detalle de cuenta de forma física.
19.-	Notificaciones de Ordenes	Una notificación sonora y visual es enviada para indicar que existe una orden disponible. Se podrá quitar la orden de la vista una vez servida y confirmada.

Tabla 2-27: Listado de programas.

Fuente: Elaboración propia.

Conclusión

Este proyecto surge de la necesidad de adaptarse e innovar tecnológicamente, así también por la búsqueda de una mayor confianza y solidez en términos de consistencia de datos. Para lo cual se propuso y se desarrolló un sistema web local; Este fue basado en conocimientos investigados y adquiridos a lo largo de nuestra carrera. Así, gracias a lo anteriormente mencionado se logró un desarrollo óptimo, eficaz, seguro y económico.

Uno de los integrantes conocía este Restobar y sus distintas problemáticas (puesto que estuvo presente en el lugar antes de iniciar el desarrollo). A este local comercial se le ofreció el desarrollo de un sistema para satisfacer aquellas necesidades que fueran importantes y urgentes. Con el fin de que todo lo que se realizaba en papel pasase a ser digital, así teniendo como ventajas:

- Datos almacenados en una base de datos estable gracias a MariaDB.
- Rapidez en la comunicación de pedidos, en el cálculo de precios, en el cálculo de propinas, una mayor organización, producción de informes y facturas con datos completamente acertados.

Para el desarrollo del sistema fue necesaria estar en constante comunicación con el cliente para que nos detallara sus problemáticas, las soluciones que necesitaba y las que se le podía brindar, avances de este, y como tal, llevarle la idea de lo que es un POS, el cual fue nuestro desarrollo final en conclusión.

No se han especificado nuevos requerimientos para futuros o funcionalidades para agregarle al sistema, pero nos encontramos abiertos a atender nuevas necesidades que el cliente requiere satisfacer.

Bibliografía

- @rimorsoft. (s.f.). Roles y permisos en Laravel con spatie/laravel-permission. Recuperado de <https://rimorsoft.com/roles-y-permisos-en-laravel-con-spatie-laravel-permission>
- JS Foundation. (s.f.). jQuery API Documentation. Recuperado de <https://api.jquery.com/>
- Laravel. (2019). Laravel - The PHP Framework For Web Artisans. Recuperado de <https://laravel.com/docs/5.8/releases>
- Mark Otto, Jacob Thornton, and Bootstrap contributors. (s.f.). Introduction. Recuperado de <https://getbootstrap.com/docs/4.3/getting-started/introduction/>
- Microsoft. (2019). Documentation for Visual Studio Code. Recuperado de <https://code.visualstudio.com/docs>
- Mozilla MDN. (2019a, 11 mayo). What is JavaScript? Recuperado 8 noviembre, 2019, de https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/Qu%C3%A9_es_JavaScript
- Mozilla MDN. (2019b, 24 abril). HTML5. Recuperado de <https://developer.mozilla.org/es/docs/HTML/HTML5>
- Mozilla MDN. (2019c, 4 octubre). HTML. Recuperado de <https://developer.mozilla.org/es/docs/Web/HTML>
- Mozilla MDN. (2019d, 25 julio). CSS3. Recuperado de <https://developer.mozilla.org/es/docs/Archive/CSS3>
- Mozilla MDN. (2019e, 10 septiembre). CSS. Recuperado de <https://developer.mozilla.org/es/docs/Web/CSS>
- MySQL. (2019a). The CHAR and VARCHAR Types. Recuperado de <https://dev.mysql.com/doc/refman/8.0/en/char.html>
- MySQL. (2019b). The DATE, DATETIME, and TIMESTAMP Types. Recuperado de <https://dev.mysql.com/doc/refman/8.0/en/datetime.html>
- MySQL. (2019c). Numeric Types. Recuperado de <https://dev.mysql.com/doc/refman/8.0/en/numeric-types.html>
- Nube Colectiva. (2019, 7 junio). Que es Laravel Collective y cómo implementarlo en Laravel. Recuperado de <http://blog.nubecollectiva.com/que-es-laravel-collective-y-como-implementarlo-en-laravel/>
- Rodríguez Patiño, Eduardo. (2016, 13 junio). ¿Qué es Composer y cómo se usa en PHP? Recuperado de <https://anexsoft.com/que-es-composer-y-como-se-usa-en-php>

Spatie. (2019). spatie/laravel-permission. Recuperado de <https://github.com/spatie/laravel-permission>

The PHP Group. (s.f.). PHP: ¿Qué es PHP? - Manual. Recuperado de <https://www.php.net/manual/es/intro-what-is.php>

Debian. (2018). Debian -- Documentation. Recuperado de <https://www.debian.org/doc/>

MariaDB Foundation. (2018, 23 mayo). About MariaDB - MariaDB.org. Recuperado de <https://mariadb.org/about/>

PhpMyAdmin contributors. (s.f.). Phpmyadmin Documentation. Recuperado de <https://www.phpmyadmin.net/docs/>

W3.org. (2019, 22 enero). CSS Snapshot 2018. Recuperado de <https://www.w3.org/TR/css-2018/>

Git. (s.f.). About - Git. Recuperado de <https://git-scm.com/about>