



Instituto de Estudios Superiores El Lago

**DPTO. DE TECNOLOGÍAS
Y SISTEMAS DE LA INFORMACIÓN**

**DESARROLLO DE APLICACIONES
MULTIPLATAFORMA**

TRABAJO DE FIN DE GRADO

Sistema de Inventario y Venta

Autores: Ángel Bruno López Taday y Juan Carlos Otero Martínez

Tutor: José Luis Meiriño Seoane

Director: Juan José Rubio Carretero

Junio, 2020

Resumen

En los tiempos que corren actualmente todo propietario o emprendedor de un pequeño negocio se ve en la necesidad de implementar un sistema informático para controlar la actividad económica de este. Es por eso por lo que se crea automáticamente la necesidad de obtener softwares para hacer inventario, llevar un recuento de empleados y para vender los productos.

La necesidad de varios softwares de este tipo, que habitualmente son de pago, hace necesaria una inversión de capital en distintos programas y servidores, generando gastos considerados a todo pequeño negocio que dedica su actividad económica a la venta de productos. Uno de los problemas de este suceso son los pagos a distintas empresas por esos servicios, pero cabe destacar otro que viene totalmente ligado a este: la formación para el uso de los distintos programas.

En cuanto a estos programas, en su gran mayoría carecen de una dedicación por parte de los desarrolladores a crear un programa intuitivo y dedicado al caso de uso de un usuario con conocimientos medios en el campo de la informática. Por esta razón, estos softwares requieren de una enseñanza de uso para cada cliente que en muchos casos es bastante complicada y extensa, requiriendo incluso una experiencia con estas herramientas previa.

En el actual Trabajo de Fin de Grado se va a desarrollar una herramienta que unifica estos softwares, generando un sencillo programa capaz de obtener un control de stock, empleados y generar ventas, junto con sus facturas, tickets y documentación administrativa necesaria.

Por ende, unificadas las aplicaciones en una, se reducirán los costes a los pequeños comerciantes, que es el foco de venta principal, de varios pagos a un único pago. En cuanto al problema de la complejidad de uso de estos, se presenta un programa dedicado a un usuario el cuál no requiere formación previa en uso de terminales, si no una pequeña explicación de uso rápida, debido al diseño simple e intuitivo del aplicativo.

Esta herramienta se implementará utilizando la tecnología dedicada para el desarrollo gráfico de *JavaFX*, dividiendo el programa en varias pantallas de control e irá de la mano de la base de datos local de *MySQL* y un sistema de producción de facturas, tickets y reportes desarrollado con la tecnología de *TIBCO Jaspersoft Studio*. Poniendo así en práctica los conocimientos aprendidos con estas tecnologías y lenguajes a lo largo del Grado Superior.

Gracias al uso de *JavaFX* se dan resultados de rendimiento óptimo de la aplicación con un costo de procesamiento mínimo, capaz de soportar por un equipo con un sistema operativo básico con un JDK de Java 7u o posterior.

Por consiguiente, con el almacenamiento local de la base de datos de *MySQL*, se consigue reducir el costo del producto, pues no será necesario pagar un servidor de bases de datos y se utilizará un uso reducido de memoria del equipo.

Por último, el uso de *Jaspersoft* para generar los documentos administrativos que necesita cualquier negocio dedicado a la venta, reduce el coste de compra de otro, pues se entregan con la aplicación unos diseños predefinidos personalizables y la capacidad de almacenar estos localmente con formato de pdf en caso de que fuera necesario.

Índice general

Índice de Contenido

1	Tabla de Figuras.....	3
2	Definición del Sistema.....	1
2.1	Análisis de utilidad.	1
2.2	Proyección a nivel empresarial.	2
2.3	Identificación del alcance del sistema.....	3
2.4	Identificación del entorno tecnológico y tecnologías a utilizar.	4
2.4.1	JavaFX.....	4
2.4.2	NetBeans IDE.....	6
2.4.3	MySQL	6
2.4.4	Jaspersoft Studio	7
2.4.5	Librerías y jars necesarios para el desarrollo	8
2.5	Planificación de las fases de desarrollo del proyecto	9
3	Análisis del Sistema	11
3.1	Especificación de requisitos.	11
3.1.1	Medios utilizados	11
3.1.2	Requisitos mínimos	12
3.2	Especificación de estándares y normas de aplicación en el proyecto.	12
4	Diseño del Sistema	13
4.1	Diseño de interfaces de usuario.....	13
4.1.1	Login	13
4.1.2	Pantalla principal o Home	14
4.1.3	Conexión.....	16
4.1.4	Clientes.....	17
4.1.5	Ventas.....	21
4.1.6	Artículos	28
4.1.7	Proveedores	30
4.1.8	Devoluciones	33
4.1.9	Reportes	35

4.1.10	Control de Empleados	39
4.1.11	Recursos	43
4.1.12	Calculadora.....	46
4.2	Diseño del modelo de datos.....	47
4.3	Otros diagramas	48
4.3.1	Modelo de la Base de Datos de España	48
4.3.2	JavaDoc de la aplicación.....	48
5	Aspectos relevantes del sistema	49
6	Pruebas de accesibilidad y usabilidad	55
6.1	Usabilidad.....	55
6.2	Accesibilidad.....	56
7	Despliegue del proyecto.....	57
7.1	Despliegue Personal	57
7.2	Despliegue por Distribuidora	57
8	Problemas encontrados	58
9	Conclusiones.....	59

1 Tabla de Figuras

FIGURA 1-1 DIAGRAMA DE ELECCIÓN	3
FIGURA 1-2 LOGO JAVA FX	4
FIGURA 1-3 LOGO SCENE BUILDER	5
FIGURA 1-4 LOGO CSS3	5
FIGURA 1-5 LOGO NETBEANS IDE	6
FIGURA 1-6 LOGO MYSQL	6
FIGURA 1-7 LOGO MYSQL WORKBENCH	7
FIGURA 1-8 LOGO TIBCO JASPER SOFT STUDIO	7
FIGURA 1-9 LOGO MYSQL CONNECTOR	8
FIGURA 1-10 LOGO FONT AWESOME	8
FIGURA 1-11 LOGO JASPER REPORTS JAR	8
FIGURA 1-12 LOGO ITEXT JAR	8
FIGURA 1-13 LOGO APACHE COMMONS	8
FIGURA 1-14 LOGO MAIL JAR	8
FIGURA 1-15 DIAGRAMA DE GANTT	10
FIGURA 3-1 LOGIN	13
FIGURA 3-2 HOME ADMINISTRADOR	14
FIGURA 3-3 HOME USUARIO	15
FIGURA 3-4 DESCONEXIÓN 1	16
FIGURA 3-5 DESCONEXIÓN 2	17
FIGURA 3-6 DESPLEGABLE CLIENTES	17
FIGURA 3-7 REGISTRO DE CLIENTES	18
FIGURA 3-8 CONTROL DE CLIENTES	19
FIGURA 3-9 MODIFICACIÓN DE CLIENTE	20
FIGURA 3-10 DESPLEGABLE VENTAS	21
FIGURA 3-11 VENTA 1	21
FIGURA 3-12 VENTA 2	22
FIGURA 3-13 VENTA 3	22
FIGURA 3-14 VENTA 4	23
FIGURA 3-15 VENTA 5	23
FIGURA 3-16 PAGO 1	24
FIGURA 3-17 PAGO 2	24
FIGURA 3-18 PAGO 3	25
FIGURA 3-19 PAGO 4	25
FIGURA 3-20 PAGO 5	26
FIGURA 3-21 TIQUE EJEMPLO	27
FIGURA 3-22 ALERTA DE CAMBIO	27
FIGURA 3-23 DESPLEGABLE ARTÍCULOS	28
FIGURA 3-24 REGISTRO DE ARTÍCULOS	28
FIGURA 3-25 CONTROL DE ARTÍCULOS	29
FIGURA 3-26 MODIFICAR ARTÍCULO	29
FIGURA 3-27 DESPLEGABLE PROVEEDORES	30
FIGURA 3-28 REGISTRO PROVEEDORES	30
FIGURA 3-29 REGISTRO TIPO DE ARTÍCULO	31
FIGURA 3-30 CONTROL DE PROVEEDORES	32
FIGURA 3-31 DESPLEGABLE DEVOLUCIONES	33
FIGURA 3-32 DEVOLUCIÓN POR TIQUE	33

FIGURA 3-33 DEVOLUCIÓN POR ARTÍCULO	34
FIGURA 3-34 DESPLEGABLE REPORTES	35
FIGURA 3-35 REPORTES	35
FIGURA 3-36 REPORTE DE CLIENTES	36
FIGURA 3-37 REPORTE DE ARTÍCULOS	36
FIGURA 3-38 REPORTE DE DEVOLUCIONES	37
FIGURA 3-39 REPORTE DE PRODUCTOS	37
FIGURA 3-40 REPORTE DE PROVEEDORES	38
FIGURA 3-41 REPORTE DE VENTAS	38
FIGURA 3-42 REPORTE DE EMPLEADOS	39
FIGURA 3-43 DESPLEGABLE DE EMPLEADOS	39
FIGURA 3-44 REGISTRO DE EMPLEADO	40
FIGURA 3-45 CONFIRMACIÓN DE PUESTO DE EMPLEADO	40
FIGURA 3-46 REGISTRO DE CONTRASEÑA DE EMPLEADO/USUARIO	41
FIGURA 3-47 CONTROL DE EMPLEADOS	42
FIGURA 3-48 MODIFICACIÓN DE EMPLEADO	42
FIGURA 3-49 RECURSOS 1	43
FIGURA 3-50 RECURSO DE FACTURA	44
FIGURA 3-51 CÓDIGO POR ARTÍCULO	44
FIGURA 3-52 CÓDIGO DE BARRAS DE TODOS LOS ARTÍCULOS	45
FIGURA 3-53 RECURSO DE CORREO	45
FIGURA 3-54 CALCULADORA	46
FIGURA 3-55 MODELO ENTIDAD-RELACIÓN SISTEMA DE INVENTARIO	47
FIGURA 3-56 DIAGRAMA ENTIDAD-RELACIÓN ESPAÑA	48
FIGURA 3-57 JAVADOC DEL PROYECTO	48
FIGURA 4-1 PROFILE	49
FIGURA 4-2 HERRAMIENTA TELEMETRY	49
FIGURA 4-3 TELEMETRY EN EJECUCIÓN	50
FIGURA 4-4 HERRAMIENTA METHODS	50
FIGURA 4-5 METHODS EN EJECUCIÓN	51
FIGURA 4-6 HERRAMIENTA OBJECTS	51
FIGURA 4-7 OBJECTS EN EJECUCIÓN	52
FIGURA 4-8 HERRAMIENTA THREADS	52
FIGURA 4-9 HERRAMIENTA LOCKS AND THREADS	53
FIGURA 4-10 HERRAMIENTA SQL QUERYs	53
FIGURA 4-11 SQL QUERYs AL INICIALIZAR	54
FIGURA 4-12 SQL QUERYs EN EJECUCIÓN	54
FIGURA 5-1 INTERFAZ DE SISTEMA DE INVENTARIO Y VENTA	55
FIGURA 5-2 INTERFAZ LOGIN CON ERRORES	56
FIGURA 5-3 ALERTA DE INSERCIÓN	56

2 Definición del Sistema

El Sistema de Inventario y Venta mostrado en este Trabajo de Fin de Grado se extiende a una interfaz gráfica desarrollada con **JavaFX**, una base de datos local de **MySQL**, y un sistema de reportes en **Jaspersoft Studio**. A través de estas tres herramientas de desarrollo, se hace un sistema de control de inventario, capaz de llevar un control correctamente ordenado y esquematizado. En cuanto a la versatilidad de uso, se ha dividido el aplicativo varias pantallas en las cuales se diferencian claramente cada una de las estructuras de control de la empresa y se almacena para modificar estas estructuras de manera concurrente y sin generar problemas de pérdidas de datos importantes.

2.1 Análisis de utilidad.

En la actualidad, el almacenamiento de datos de manera segura, ordenada y con una accesibilidad sencilla prima en lo que a empresarialmente se refiere. Es por eso por lo que decidimos desarrollar una herramienta que disponga de las siguientes características y utilidades:

- Administración de clientes, artículos, proveedores y empleados
- Registro de ventas por cliente
- Registro de devoluciones de artículos
- Creación de facturas, tiques y reportes
- Generador de código de barras para los distintos artículos
- Generador de correos automáticos
- Calculadora

En primer lugar, a lo que se refiere a ventajas administrativas del negocio, encontramos unas herramientas para alojar en la base de datos toda la información necesaria del cliente para registrarlo, además de la posibilidad de modificar o eliminar al cliente en caso de que fuera necesario.

En cuanto a la administración de artículos, Sistema de Inventario y Venta dispone de un control de stock de artículos, organizados en una tabla y distinguidos por significativos como el código de barras, además de estar cada artículo unido a su proveedor, IVA por categoría de artículo y precios de venta y costo. De esta manera, la utilidad del aplicativo se extiende a un control de Stock exhaustivo de los productos que el local se dispone a vender.

Respecto a los proveedores, tal y como se dispone anteriormente, cada proveedor se registra con un tipo de artículo, el cual distribuirá, con un precio de venta al cliente y un Precio de Venta al Público (PVP), en algunos casos no obligatorio si no recomendado.

Todas estas características son, además, restringidas por permisos para establecer una jerarquía dentro de la empresa y de la propia aplicación. Las mencionadas estarán disponibles para los administradores del sistema y para los encargados/dueños del local que requiere del sistema. De esta manera, el administrador podría controlar los empleados, proveedores y demás datos que el usuario carece de permisos para acceder a ellos.

SISTEMA DE INVENTARIO

En cuanto a los permisos de los usuarios, el manejo de utilidades se reduce a: administración de ventas por cliente, registro de artículos y devoluciones, calculadora y los útiles necesarios. Además, este sistema de login con permisos, da a la herramienta la versatilidad de poder organizar por departamentos la tienda y no tener problemas de uso de herramientas indebidas a todos los empleados.

Si bien nos referimos a utilidad, nos debemos enfocar en el caso de uso del aplicativo, el caso de uso de un pequeño comercio que se dedica a la venta de todo tipo de artículos y que dispone de pocos recursos económicos para contratar varios servicios externos para la organización de la tienda.

Con la herramienta Sistema de Inventario y Venta evitamos que el dueño de este pequeño comercio tenga que disponer de un programa para su *Terminal de Punto de Venta (TPV)*, que irá unido a un servidor para alojar la base de datos (cobrado de manera independiente), un programa para hacer administrar el stock mediante un inventario, y un programa para administrar los empleados de la tienda. Siendo así la distribución de uso normal en cuanto a softwares dedicados a comercios, con Sistema de Inventario y Venta, conseguimos unificar todas esas características de los distintos softwares y, además, proveer de un almacenamiento local de la base de datos, el cuál si fuera necesario por tamaño o demanda, se podría alojar en un MySQL Server o cualquier otro alojamiento web.

Respecto a una de las últimas utilidades, ya marcadas en el *Resumen*, se encuentra la utilidad de uso para todo usuario sin necesidad de formaciones especiales ni el uso de herramientas que no estén dentro de los conocimientos de un uso normal. El diseño intuitivo y sencillo, a la par que detallado gráficamente de Sistema de Inventario y Venta, hace que el uso de las herramientas y diferentes ventanas del aplicativo esté al alcance de cualquier empleado, ya sea nuevo o con experiencia.

Con relación a esto último, desde el equipo de Sistema de Inventario y Venta, estuvimos planteando diferentes opciones para manejar los diferentes programas para originar facturas y documentos administrativos, pues sí que era lo más complicado, y finalmente elegimos Jaspersoft Studio como sistema de reportes y demás documentos, pues te permite realizar todo en formato pdf, formato familiar y que la mayoría de las personas suelen manejar.

2.2 Proyección a nivel empresarial.

Sistema de Inventario y Venta nace como una idea empresarial a manos de Ángel Bruno López Taday, quien ha desempeñado y actualmente sigue desempeñando, el arduo trabajo de reponedor en supermercados. Principalmente esta idea surge debido a la cantidad de programas usados para hacer inventarios, ventas y registros usados a lo largo de la vida de Ángel, los cuales él cree incompletos y con falta de compatibilidad entre ellos.

Por otro lado, Juan Carlos Otero Martínez, nace como colaborador de la aplicación debido al uso de diferentes sistemas de cobro (TPV's en su mayoría) e inventario a lo largo de su vida laboral en diferentes lugares de trabajo.

Ambos coinciden en la idea de que hay que mejorar esos sistemas por los cuales las empresas pagan cuantiosas cantidades de dinero y ofrecen muchas limitaciones. Es así como, se enfocan al caso de uso de una empresa pequeña como puede ser un negocio familiar, un bar o una tienda de ultramarinos, la cual decide abrir sus puertas o invertir en un sistema informático que les proporcione un control exhaustivo del negocio.

SISTEMA DE INVENTARIO

Al paso de varios locales y empresas, ven otro problema en su caso de uso: una tienda/empresa con ese tamaño suele disponer de presupuestos ajustados y sin posibilidad de pagar cuotas mensuales grandes. Vistos estos dos problemas, ven la necesidad de hacer llegar a los pequeños comerciantes un programa sencillo, intuitivo, que unifique sistemas para no tener que disponer de una gran cantidad de terminales en un negocio pequeño y que se pueda operar sobre cualquier información de la empresa en cualquier momento.

El tercer objetivo que ven claro a cumplir con Sistema de Inventario y Venta es la simplicidad en algo que puede resultar complicado como es en un registro de artículos o un control de Stock. Según han interactuado con distintos sistemas, han tenido que recibir formaciones, procesos de adaptación a los sistemas, de, incluso meses para hacer las tareas que ellos creían más sencillas.

2.3 Identificación del alcance del sistema.

Observando los problemas a cubrir, el alcance de sistema se identifica claramente con un objetivo claro: Desarrollar un software para tiendas pequeñas que sea completo, sutil, económico e intuitivo.

El diagrama para el inicio del proyecto fue desarrollado con los criterios de los lenguajes y herramientas aprendidas durante el curso. Con este diagrama, se deciden las tecnologías, lenguajes y se comienza a desarrollar con los NetBeans, MySQL Workbench y en el lenguaje de JavaFX.

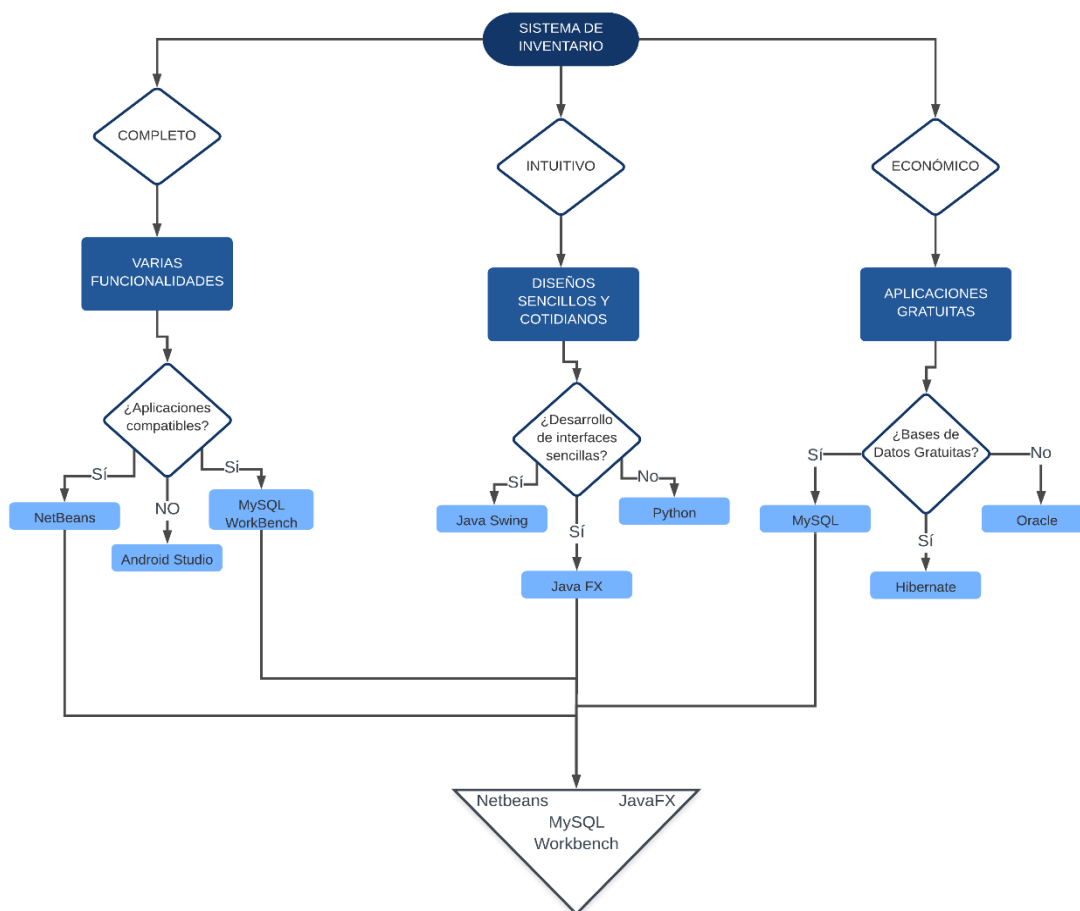


Figura 2-1 Diagrama de elección

2.4 Identificación del entorno tecnológico y tecnologías a utilizar.

El desarrollo del aplicativo se basa en el lenguaje Java, en el entorno de desarrollo gráfico de JavaFX, realizado en un IDE, en este caso NetBeans, para su desarrollo informático y desarrollado gráficamente con Scene Builder 2.0 para la modificación de los FXML y CSS para los estilos.

En cuanto a la base de datos, se desarrolla en un almacenamiento local, creado a través de MySQL en el software de diseño MySQL Workbench. Este irá unido al Java para el almacenamiento de datos y se complementará con la tecnología de reportes de Jaspersoft Studio para representar los documentos al usuario sobre las consultas a la base de datos.

2.4.1 JavaFX



Figura 2-2 Logo JavaFx

JavaFX es una familia de productos y tecnologías de Oracle Corporation para la creación de aplicaciones web que tienen las características y capacidades de aplicaciones de escritorio, incluyendo aplicaciones multimedia interactivas.

Las aplicaciones JavaFX pueden ser ejecutadas en una amplia variedad de dispositivos. En octubre de 2011 fue lanzada la versión 2.0. Para el desarrollo de aplicaciones JavaFX un lenguaje declarativo, tipado llamado JavaFX Script, además puede integrarse código Java en programas JavaFX. JavaFX es compilado a código Java, por lo que las aplicaciones JavaFX pueden ser ejecutadas en computadores con la máquina virtual de Java instalada (JRE), o celulares corriendo Java ME.

Investigando sobre esta tecnología de código abierto, innovadora y hecha para diseños gráficos modernos e intuitivos, investigamos sobre el uso de JavaFX con Java en el IDE de NetBeans, y tras acceder a su documentación de la página oficial (<https://openjfx.io/openjfx-docs/#install-javafx>) conseguimos, con los conocimientos previos de Java, empezar a desarrollar con JavaFX.

La tecnología que presenta para crear aplicaciones de escritorio está dividida en 2 grandes pilares: el desarrollo en Java con NetBeans y el desarrollo gráfico en Scene Builder 2.0. El desarrollo de Java, así como el desarrollo gráfico no supone un problema para todo el que haya trabajado en interfaces de Java anteriormente, ya sea Java2EE o Java Swing. Puesto que se asemeja mucho a lo estudiado y presenta un diseño más actual y manejable, elegimos esta tecnología.

2.4.1.1 Scene Builder



Figura 2-3 Logo Scene Builder

Scene Builder es una herramienta de diseño visual que permite a los usuarios diseñar rápidamente interfaces de usuario de aplicaciones JavaFX, sin codificación. Los usuarios pueden arrastrar y soltar componentes de la IU a un área de trabajo, modificar sus propiedades, aplicar hojas de estilo y el código FXML para el diseño que están creando se genera automáticamente en segundo plano. El resultado es un archivo FXML que luego se puede combinar con un proyecto Java al vincular la interfaz de usuario a la lógica de la aplicación.

Además, es una herramienta de diseño de interfaces de usuario sencilla, da un amplio flujo de trabajo para los programadores, pues es compatible con cualquier IDE, es una plataforma cruzada, pues es compatible con todos los Sistemas Operativos, es un editor visual de FXML, dejando atrás el editar picando código XML, ofrece una vista previa al trabajo a tiempo real y soporta CSS para los estilos de cada elemento que añadas al FXML.

2.4.1.2 CSS3



Figura 2-4 Logo CSS3

CSS (siglas en inglés de Cascading Style Sheets), en español «Hojas de estilo en cascada», es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado. Es muy usado para establecer el diseño visual de los documentos web, e interfaces de usuario. Junto con HTML y JavaScript, CSS es una tecnología usada por muchos sitios web para crear páginas visualmente atractivas, interfaces de usuario para aplicaciones web y GUIs para muchas aplicaciones móviles.

Así mismo, esta tecnología de lenguaje de marcado se compatibiliza a la perfección con Scene Builder para crear las interfaces gráficas más editables para el usuario. El lenguaje usado es igual que el CSS utilizado para los estilos web, pero con alguna que otra peculiaridad de Scene Builder. La integración de las hojas de estilo se puede hacer de manera gráfica, en la edición de la interfaz o con referencias a través del código XML de las interfaces FXML.

Las ediciones de estilos en CSS están realizadas con NetBeans, IDE que se muestra a continuación y se aplican a todos los elementos posibles del proyecto. El uso del CSS en Sistema de Inventario y Venta es vital, pues ofrece el estilo personal e innovador del aplicativo y la apariencia perfeccionista del mismo.

SISTEMA DE INVENTARIO

2.4.2 NetBeans IDE



Figura 2-5 Logo NetBeans IDE

NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. NetBeans IDE es un producto de código abierto con una gran base de usuarios, una comunidad en constante crecimiento. Sun Microsystems fundó el proyecto de código abierto NetBeans en junio de 2000 y continúa siendo el patrocinador principal de los proyectos (Oracle Corporation).

El NetBeans IDE soporta el desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones móviles). Entre sus características se encuentra un sistema de proyectos basado en Ant, control de versiones y refactoring.e java escritas para interactuar con las APIs de NetBeans y un archivo especial (manifest file) que lo identifica como módulo.

Siendo así, NetBeans es la aplicación que hemos utilizado y sobre la que hemos aprendido a programar, por lo que escogimos este IDE para desarrollar la aplicación, pues es totalmente compatible con todas las tecnologías a usar, incluso con **GitHub**, dando un manejo y una maniobrabilidad a la hora de programar 2 personas a la vez sin pisarnos muy buena. Con esta tecnología se prueba, desarrolla y genera la aplicación.

2.4.3 MySQL



Figura 2-6 Logo MySQL

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, y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web.

Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y los derechos de autor del código están en poder del autor individual, MySQL es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

MySQL es otro de los lenguajes, que, gracias a la formación durante el Grado Superior, hemos conseguido manejarlo íntegramente y por eso decidimos usarlo para la base de datos de Sistema de Inventario y Venta. Tras investigaciones sobre otros sistemas de gestión como MongoDB o Firebase y una prueba de nuestro sistema funcionando con Hibernate, optamos por el almacenamiento local y la sintaxis sencilla que muestra MySQL a través de su herramienta MySQLWorkbench. Una vez más, esta consigue compatibilidad con el código de Java gracias a un conector incluido en las librerías del proyecto, además de mostrar total compatibilidad con Jaspersoft para generar reportes mediante consultas a la base de datos local de MySQL.

SISTEMA DE INVENTARIO

2.4.3.1 MySQL Workbench



Figura 2-7 Logo MySQL Workbench

MySQL Workbench es una herramienta visual unificada para arquitectos de bases de datos, desarrolladores y DBA. MySQL Workbench proporciona modelado de datos, desarrollo de SQL y herramientas de administración integrales para la configuración del servidor, la administración de usuarios, la copia de seguridad y mucho más. MySQL Workbench está disponible en Windows, Linux y Mac OS X. Es el sucesor de DBDesigner 4 de fabFORCE.net, y reemplaza el anterior conjunto de software, MySQL GUI Tools Bundle.

Ofrece un diseño y administración visual de bases de datos, mediante la creación de modelos ER, ingeniería directa e inversa y cambios muy complejos facilitados al máximo. Es un software que ofrece herramientas visuales para el manejo de scripts SQL y un acceso instantáneo a los esquemas y objetos de la base de datos. Junto con estas características, cabe añadir que permite, a través de la consola visual administrar, consultar y perfeccionar el rendimiento e incluso migrar a otros servidores como PostgreSQL o SQL Server bases de datos, independientemente de su magnitud.

2.4.4 Jaspersoft Studio



Figura 2-8 Logo TIBCO Jaspersoft Studio

TIBCO Jaspersoft Studio es un software de edición para TIBCO JasperReports®. Le ayudará a diseñar y ejecutar plantillas de informes; construir consultas de informes; escribir expresiones complejas; componentes visuales de diseño como más de 50 tipos de gráficos, mapas, tablas, tablas cruzadas y visualizaciones personalizadas; y mucho más. Integra el servidor TIBCO JasperReports® para crear potentes flujos de trabajo de publicación de informes.

Con Jaspersoft Studio, puede acceder a diferentes tipos de fuentes de datos, incluidos big data, CSV, Hibernate, Jaspersoft Domain, JavaBeans, JDBC, JSON, NoSQL, TIBCO Spotfire® Information Links, XML o su propia fuente de datos personalizada.

Con esta herramienta dedicada a reportes de consultas, mediante la creación de reportes consultados a la base de datos de MySQL a través de un adaptador JDBC, conseguimos hacer un formato de tickets y facturas para el negocio en concreto. Principalmente creamos unas plantillas con un nombre comercial y unos datos ficticios para mostrar al cliente cómo se vería el diseño principal de los tickets y facturas. Este diseño es totalmente maleable al gusto del cliente, pudiéndose incluir más o menos datos y diferentes informaciones. Basándonos en el marco legal administrativo, hicimos consultas y seguimos la forma habitual que suelen tomar estos documentos para presentar a gestores/contables y luego archivar en libros de cuentas y/o suministrar al cliente. En adición a estos documentos, creamos un generador de código de barras para artículo por si fuera necesario proveer un código de barras en cualquier momento y artículo.

SISTEMA DE INVENTARIO

2.4.5 Librerías y jars necesarios para el desarrollo

Figura 2-9 Logo MySQL Connector

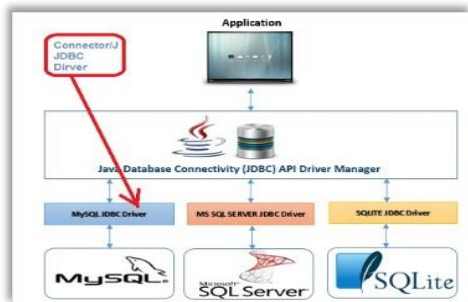


Figura 2-10 Logo Font Awesome

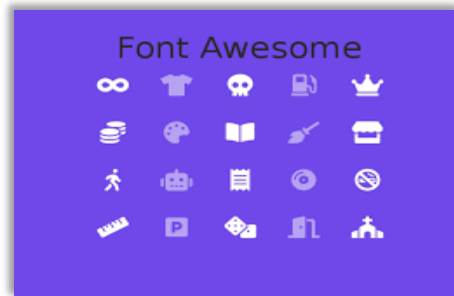


Figura 2-11 Logo JasperReports jar

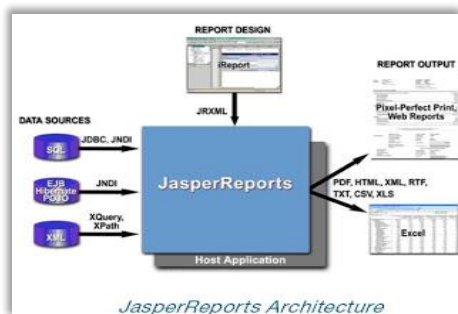


Figura 2-12 Logo iText jar



Figura 2-13 Logo Apache Commons



Figura 2-14 Logo Mail jar



-MySQL Connector J: MySQL Connector/J es el controlador JDBC oficial para MySQL con Java. MySQL Connector/J es compatible con todas las versiones de MySQL.

-Fontawesomefx-8.9: Librería destinada al uso de iconos y útiles gráficos para complementar los FXML en Scene Builder.

-Jasperreports-6.3.0: Jar necesario para el uso de reportes en de Jaspersoft en aplicaciones Java con NetBeans.

-iText-5.0.5: Jar necesario para generar los códigos de barras junto con la tecnología de Jaspersoft en forma de etiqueta y en formato pdf.

-Apache Commons Collections: Esta colección proporciona una clase de utilidades para cada interfaz de colecciones principales. Las colecciones utilizadas son: commons.collections-3.2.1, commons-beanutils-1.8.3, commons-digester-2.1, commons-javaflow y commons-logging-1.1.

SISTEMA DE INVENTARIO

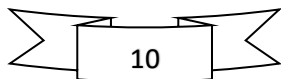
-*Mail Jar*: Este software de Oracle permite, mediante un jar en las librerías del proyecto, generar un sistema de correo dentro del aplicativo.

2.5 Planificación de las fases de desarrollo del proyecto

La duración total del proyecto ha sido, finalmente, de 192 horas, bastante repartidas, con un inicio el 27 de marzo de 2020 y una finalización el día 14 de junio de 2020. El proyecto ha superado bastante las horas previstas (55) y ha ido incrementando la magnitud del proyecto a la par.

La mejor manera de representación es el siguiente Diagrama de Gantt que muestra las horas y duración total de este proyecto.

Figura 2-15 Diagrama de Gantt



3 Análisis del Sistema

3.1 Especificación de requisitos.

3.1.1 Medios utilizados

Antes de la especificación de requisitos para Sistema de Inventario, creemos conveniente nombrar los medios utilizados para crear el mismo y especificar alguna que otra tecnología que no haya sido nombrada para crear el aplicativo.

3.1.1.1 *Hardware*

Esta sección está dividida pues somos 2 integrantes en el equipo:

Juan Carlos Otero Martinez

- Ordenador Pc Sobremesa con 8GB de memoria RAM, 225GB de SSD, 1TB de disco HDD, procesador Intel Core i5-9400 2.90GHz, tarjeta gráfica NVIDIA GeForce GTX 1650 y Windows 10 Pro.
- Monitor HP 24fw de 24 pulgadas.

Ángel Bruno López Taday

- Ordenador portátil AsusTek Computer GL553VD con 24GB de RAM, procesados Intel Core i7-7700 2.80GHz, tarjeta gráfica NVIDIA GeForce GTX 1050 y Windows 10.

3.1.1.2 *Software*

- **JavaFX** como lenguaje de programación para la implementación del aplicativo.
- **XML** como lenguaje de programación para la implementación gráfica del aplicativo.
- **CSS 3** como lenguaje de marcado para personalizar los estilos del aplicativo.
- **MySQL** como lenguaje del motor de la base de datos.
- **Hibernate** para el inicio del almacenamiento de la base de datos en Hibernate.
- **Java JDK 1.8u y 1.7u** como máquina virtual de Java para ejecutar la aplicación.
- **Discord** como sistema de comunicación entre los componentes del equipo.
- **GitHub** como sistema de almacenamiento mediante repositorios para el trabajo conjunto del código.
- **Jaspersoft Studio** para el diseño y la creación de reportes del aplicativo.
- **Scene Builder** para el diseño gráfico de la aplicación.
- **MySQL Workbench 8.0 CE** como software de diseño, creación y control de la base de datos. Además, utilizado para generar el esquema de Entidad-Relación de la base de datos.

SISTEMA DE INVENTARIO

- **NetBeans IDE** como IDE para programar y desarrollar el aplicativo. También utilizado para pruebas de rendimiento y errores del programa.
- **Herramientas de Microsoft Office** para realizar la documentación del programa.
- **Lucid chart** como software online para realizar el diagrama de elección inicial.

3.1.2 Requisitos mínimos

Para que la aplicación funcione correctamente y no tenga errores de rendimiento hemos establecido unos requisitos mínimos:

3.1.2.1 *Hardware*

- Memoria RAM 4GB
- Tarjeta gráfica virtual Intel HD Graphics Family
- Conexión Ethernet
- Disco de almacenamiento HDD de 250GB para almacenar los documentos
- Monitor de 14 pulgadas
- Teclado y ratón

3.1.2.2 *Software*

- Instalación del software MySQL Workbench para crear las bases de datos
- Instalación de Java JDK igual o superior a la versión 8.0.111
- Instalación de plugin de Java FX para el IDE
- Programa para la lectura de archivos pdf
- Correo profesional configurado

3.2 Especificación de estándares y normas de aplicación en el proyecto.

La codificación del proyecto está desarrollada con el estándar UTF-8, dado por defecto por el IDE NetBeans y por ser la codificación por defecto en cuanto a desarrollo de JavaFX y el XML de los FXML.

En cuanto a fuentes, la utilizada para el sistema en general es la fuente System, dada por el CSS por defecto en Scene Builder, a diferentes píxeles de tamaño. Otra fuente utilizada es la fuente Dominique, descargada de <https://www.dafont.com/dominique2.font> para realizar el Logo del equipo.

Finalmente, una de las normas más importantes para tener en cuenta es la del correo. El correo profesional para utilizar en la aplicación ha de ser Gmail, no puede tener un servicio de limpieza de correo automática y debe tener permitido el acceso a aplicaciones poco seguras desde Gmail.

Para que tenga acceso a los correos de la aplicación y el correo sea válido el usuario debe de ir a la *configuración de la cuenta Gmail* → *Seguridad* → Permitir el *acceso de aplicaciones poco seguras*.

4 Diseño del Sistema

4.1 Diseño de interfaces de usuario

El diseño de interfaces de usuario se divide en varias partes. Por una parte, tenemos la pantalla de login, que ocupa el inicio de la aplicación y, por otro lado, están las ventanas para el uso del usuario.

4.1.1 Login

El diseño del login es minimalista, sencillo y con varios aspectos a comentar. Pese a ser una ventana simple y con poco desarrollo gráfico, es de las más importantes. En la mitad izquierda de la izquierda, el nombre, el logo y los nombres de desarrolladores de la aplicación, y en la mitad derecha nos encontramos con dos campos, el usuario y la contraseña. Estos campos son comprobados mediante una consulta a la base de datos a la tabla de login y se realizan varias acciones.

En primer lugar, se comprueba en la base de datos si el usuario existe en la lista de empleados y si está registrado en la tabla de login, es decir, si tiene permiso para acceder al sistema. En consecuencia, aquí se bifurca la aplicación en 2 planos distintos debido a los permisos del usuario que accede, el plano del administrador, que visualiza todas las características y ventanas de la aplicación para utilizarlas a su gusto, y el plano del usuario, que sólo posee unas características limitadas y solo visualiza las ventanas de venta y artículos.

Seguido a esto, se comprueba en la base de datos la contraseña introducida para el usuario, dando seguridad al sistema y no permitiendo que cualquier persona ajena al sistema consiga entrar. Estas contraseñas están almacenadas en la tabla login de la base de datos y, para un futuro se aumentará la seguridad haciendo un sistema de cifrado de contraseñas para evitar ataques al sistema.



Figura 4-1 Login

SISTEMA DE INVENTARIO

4.1.2 Pantalla principal o Home

Como se ha explicado anteriormente, esta pantalla se verá afectada por los permisos del usuario que accede al sistema, de esta manera enfocaremos las explicaciones de manera dividida. Cabe a destacar que estas 2 opciones, siempre pueden ampliarse y modificarse por departamentos u otro tipo de organización al gusto del cliente.

4.1.2.1 Pantalla Principal -Administrador

Es la pantalla más completa de todas y dispone todas las características y ventanas opciones desarrolladas. Como se puede apreciar, se inicia en la ventana Venta por defecto, y esto ocurre apostando pues principalmente el sistema está pensado para vender productos. Además, se conserva la parte de las ventanas para navegar por las distintas en cualquier momento de manera fácil.

La ventana principal está personalizada a cada usuario, a pesar de que sea administrador y por ello tenga una interfaz diferente, también tiene nuestro toque personal de poner el nombre del usuario en todo momento.

Como vemos, las ventanas principales son: Clientes, Ventas, Artículos, Proveedores, Devolución, Reportes y Empleado. Por otro lado, están las ventanas secundarias o recursos, que es la ventana de 'Conexión', que sirve para desconectarse y logearse con otro usuario en cualquier momento, la ventana de 'Recursos' que dispone varias herramientas para el usuario y con el mismo propósito de ofrecer recursos al usuario, la ventana de la 'Calculadora'.

The screenshot displays the 'Home Administrator' interface of the 'INVENTARIO' system. At the top, there's a navigation bar with a 'Conexión' button and the user's name 'Juan Carlos Toledo' and the date '13-06-2020'. Below this is a main menu with buttons for 'Clientes', 'Ventas', 'Articulos', 'Proveedores', 'Devolucion', 'Reportes', 'Empleado', 'Recursos', and 'Calculadora'. The 'Ventas' window is currently active, showing a form for 'Nº Factura' (20807), 'Artículo' (Selecciona Artículo), 'Precio' (0,00), and 'Cantidad' (Cantidad). Below the form are buttons for 'Añadir', 'Eliminar', 'Pago', and 'Cancelar'. A table with columns 'Nº', 'Nombre Artículo', 'Cantidad', 'Precio', and 'Total' is displayed, showing 'Tabla sin contenido'. A 'Total' field is at the bottom right.

Figura 4-2 Home Administrador

SISTEMA DE INVENTARIO

4.1.2.2 Pantalla Principal-Empleado

Como se puede apreciar en la siguiente ilustración, el formato de la anterior es el mismo, incluso con nuestro toque personal de incluir el nombre del empleado, sin embargo, con menos opciones y con menos 'permisos'. Es en este punto de inflexión dónde nos centramos puesto que consideramos importante resaltar y diferenciar a los usuarios por permisos, puesto que normalmente en una empresa de este tipo no deseas que un empleado nuevo pueda modificar información importante como la agenda de proveedores o de clientes.

Por ende, el usuario/empleado dispondrá de los Recursos y la calculadora como el Administrador, pues están hechos pensados más para la venta que la administración, la disponibilidad de Desconectarse y conectarse en cualquier momento, la opción de realizar ventas y la opción de consultar y modificar el stock.

El diseño de la parte superior se conserva al navegar por las distintas pantallas para acceder a cualquiera de las ventanas en cualquier momento de manera fácil.

Nº	Nombre Artículo	Cantidad	Precio	Total
Tabla sin contenido				

Figura 4-3 Home Usuario

SISTEMA DE INVENTARIO

4.1.3 Conexión

Esta ventana ha sido desarrollada en el último proceso de desarrollo de la aplicación, puesto que es cuando se ha visto la necesidad de probar la conexión desde otros usuarios. A pesar de ser una ventana simple, marca un antes y un después en la aplicación, puesto que al acceder a ella puedes cerrar una conexión y no abrir otra hasta que el usuario se loguea. Cabe destacar la gran importancia de este cierre de conexión, puesto que proporciona un cese de tráfico de conexiones con la base de datos y no saturar de conexiones abiertas la base de datos.

Esta ventana está pensada para usar cuando el empleado/administrador deje de utilizar el sistema cierre su conexión, aunque sea 1 minuto, puesto que no consume muchos recursos y se puede acceder de nuevo rápidamente al sistema.

Como se puede apreciar, es una clara fusión de la ventana de login con la ventana Principal. Tras varios diseños, nos decidimos por este sutil diseño, que muestra como el usuario se dispone a desconectarse, pero no da esa sensación de que se ha cerrado la aplicación.

Tras decidir el diseño, jugamos un poco con la gamificación de la interfaz y decidimos que el usuario tuviera qué escribir sus datos y pulsar el botón Salir para la desconexión. De esta manera, el usuario no finaliza la conexión por error y si tiene que seguir una venta, por ejemplo, puede hacerlo.

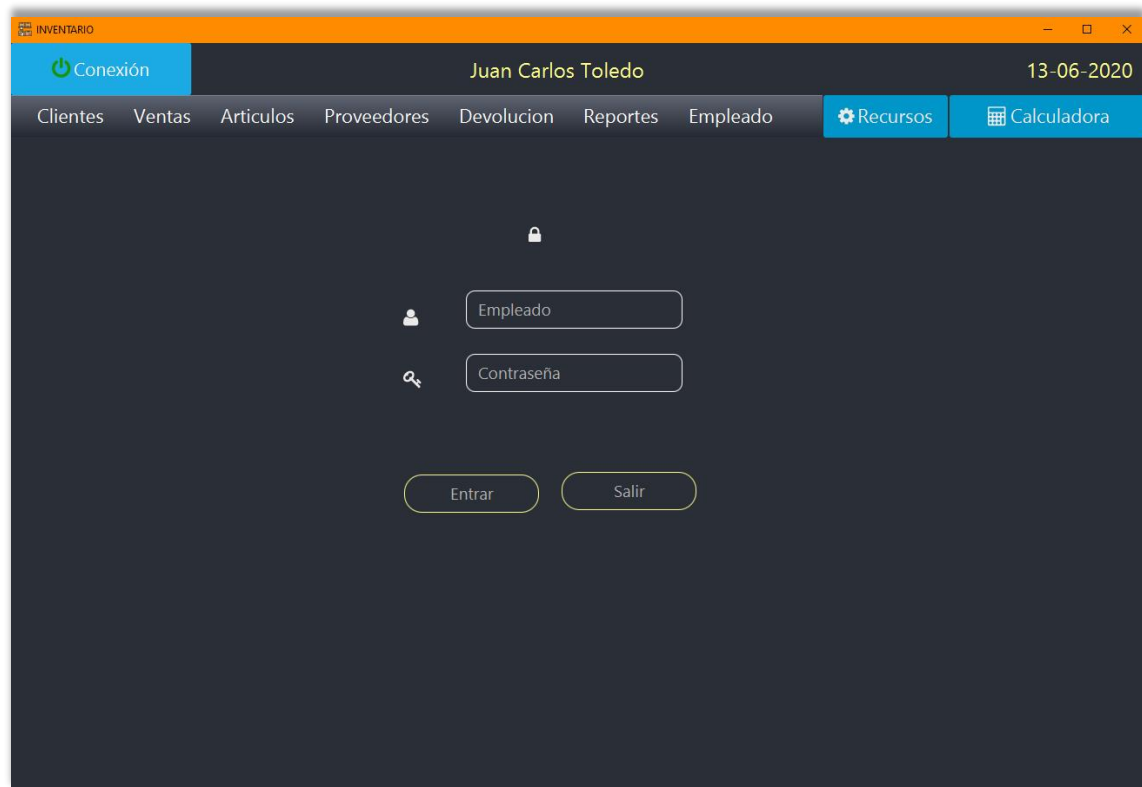


Figura 4-4 Desconexión 1

SISTEMA DE INVENTARIO

Una vez el usuario ha decidido desconectarse realmente, sale la Ventana de desconexión, que es un guiño a esta, pero con un poco más de simplicidad. En esta sólo se permite lo que consideramos básico: la conexión nuevamente, el acceso a la calculadora y cerrar o minimizar el programa.

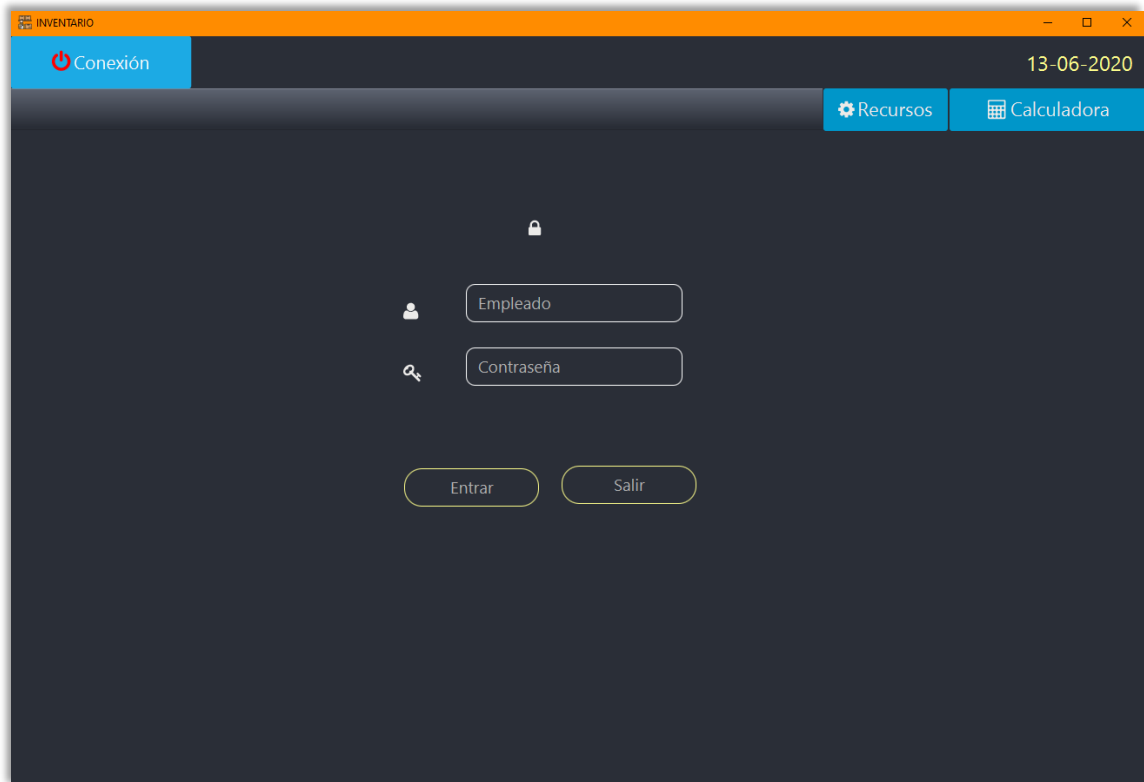


Figura 4-5 Desconexión 2

4.1.4 Cientes

Esta ventana dispone un control absoluto de la lista de clientes, teniendo una ventana, la principal, de registro de Clientes y otra de control, que permite modificar o eliminar datos del cliente. Así es como se ve el desplegable que te deja navegar entre las 2 ventanas diferentes:

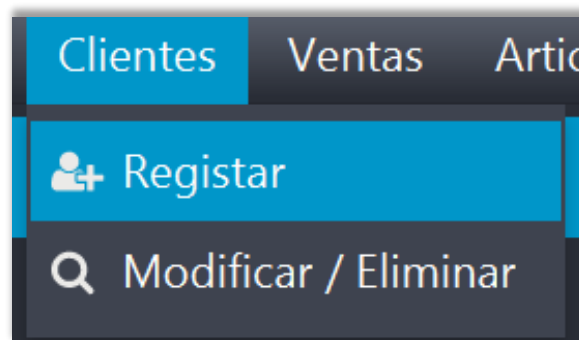


Figura 4-6 Desplegable Clientes

SISTEMA DE INVENTARIO

4.1.4.1 Registro de Clientes

En esta ventana se registran todos los datos personales del Cliente para mantener una personalización de la venta y la información de cada uno de los Clientes de la cuenta. De esta manera, podemos tener a los Clientes que necesiten consultar sus compras, datos o cualquier otra operación realizada en el comercio en cualquier momento sin necesidad de consultas complejas.

Como se puede observar, para hacer de manera efectiva el registro está el botón Guardar y, por otro lado, disponemos un botón de Cancelar, por si en cualquier momento no deseamos registrar finalmente al cliente. Este botón reiniciará el registro limpiando los campos.

En este registro se valida cada campo a introducir para que de esta manera no se tengan datos corruptos o de poca validez en la base de datos.

El control de campos se hace de la siguiente manera: el tipo de documento se controla con unas opciones recogidas en el Combo box a la izquierda del número, el número de documento es validado para que disponga 8 valores numéricos y 1 valor alfabético, se comprueban tanto el campo Teléfono como el campo de Código Postal para que sean numéricos, se valida el email para que este pertenezca a un servidor de correo válido, se comprueban los campos Nombre, Apellidos y Dirección para que sean alfabéticos, y por último, se validan las provincias y ciudades válidas con dos Combo box que consultan a una base de datos de todas las provincias y ciudades de España para que no haya errores. Cabe decir que actualmente el producto está pensado para una distribución nacional, por lo que se toma por defecto que el país es España. Todas estas validaciones son modificables a gusto del cliente.

The screenshot shows the 'Registro de Clientes' window. The form includes the following fields and controls:

- Documento:** A dropdown menu labeled 'Tipo' followed by a text input field.
- Nombre:** A text input field.
- Apellidos:** A text input field.
- Teléfono:** A text input field.
- E-mail:** A text input field.
- Pais:** A dropdown menu currently showing 'España'.
- Provincia:** A dropdown menu labeled 'Seleccionar provincia'.
- Ciudad:** A dropdown menu labeled 'Seleccionar ciudad'.
- Dirección:** A text input field.
- Codigo Postal:** A text input field.
- Nº Empleado:** A text input field currently showing '1'.

At the bottom of the form are two buttons: 'Guardar' (with a save icon) and 'Cancelar' (with a close icon).

Figura 4-7 Registro de Clientes

SISTEMA DE INVENTARIO

4.1.4.2 Control de Clientes

Por otro lado, pero formando una parte importante en cuanto a la agenda de clientes, nos encontramos con la ventana de Control de Clientes. En esta ventana se muestra una tabla con el listado de clientes registrados y nos permite modificar y eliminar cada uno de ellos de una manera intuitiva y sencilla.

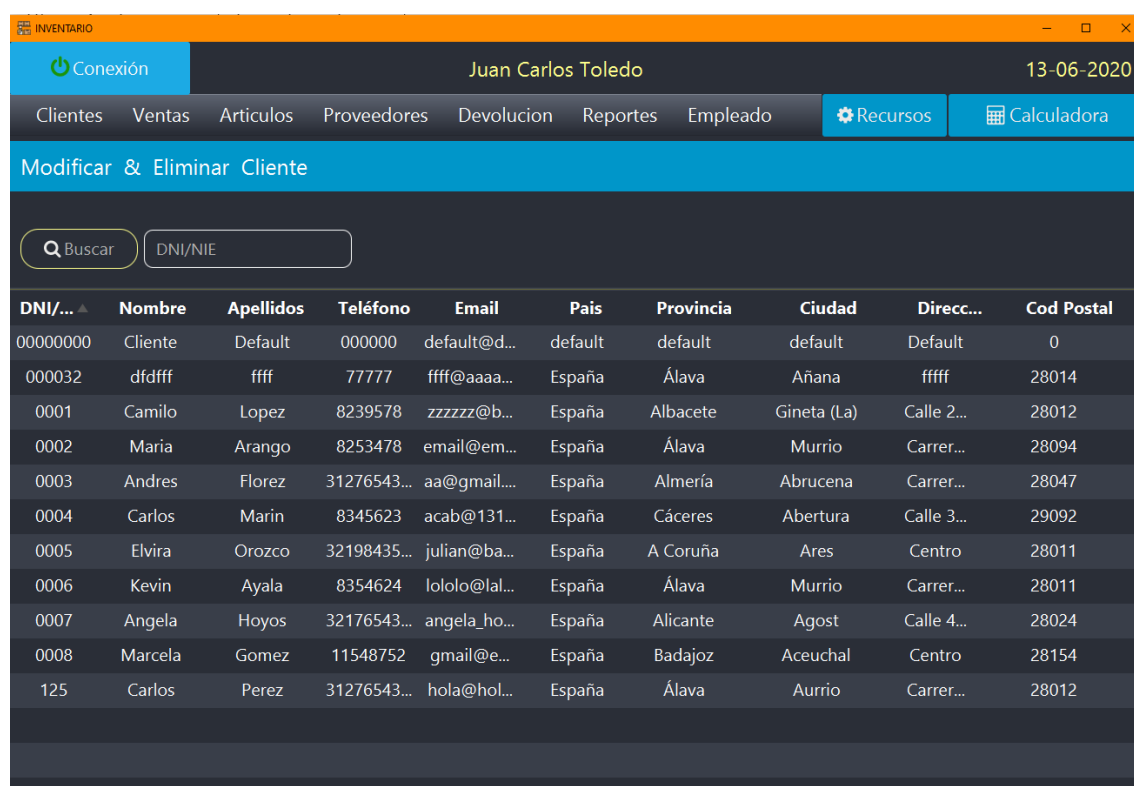
Los controles que encontramos en esta ventana la colman de versatilidad y utilidad.

Por un lado, tenemos un campo de búsqueda de clientes por identificador, es decir, el documento del cliente, que busca el cliente según se va escribiendo y por si esto no funcionara, consta de un botón buscar para hacerlo manualmente.

En la parte inferior a este tenemos la tabla, que consta con todos los datos de los clientes y nos permite acceder a ellos de una manera fácil.

La tabla visible a continuación ha sido rellenada con clientes ficticios insertados la mayoría desde la base de datos para el ejemplo. De esta manera se puede mostrar la tabla en una situación normal de un comercio con varios clientes.

Esta tabla contiene muchos elementos de gamificación: permite hacer scroll por ella para ver el listado completo de clientes, ordenar el listado por cada uno de los campos de manera ascendente y descendente (pinchando una en el campo que se quiera para hacerlo de manera ascendente y con otro clic después de este para hacerlo en sentido descendente), se puede ampliar cualquier campo según se quiera para ver la información completa y por último, pero no por ello menos importante, controlamos la selección de clientes con un doble clic en la fila.



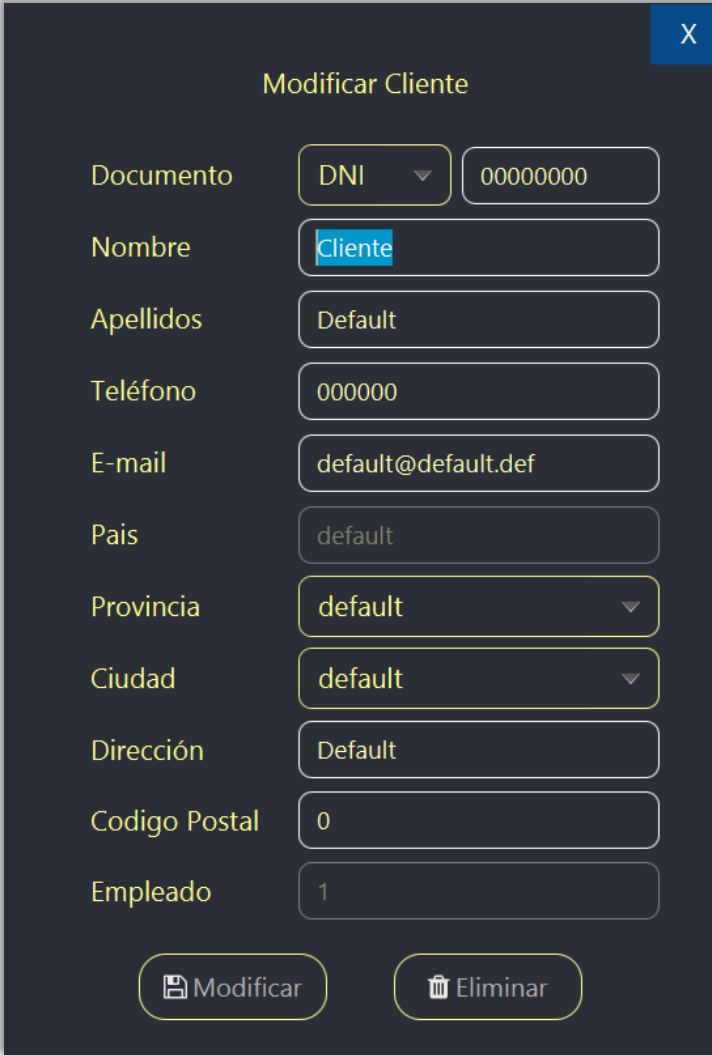
DNI/...	Nombre	Apellidos	Teléfono	Email	País	Provincia	Ciudad	Direcc...	Cod Postal
00000000	Cliente	Default	000000	default@d...	default	default	default	Default	0
000032	dfdfdf	ffff	77777	ffff@aaa...	España	Álava	Añana	fffff	28014
0001	Camilo	Lopez	8239578	zzzzz@b...	España	Albacete	Gineta (La)	Calle 2...	28012
0002	María	Arango	8253478	email@em...	España	Álava	Murrio	Carrer...	28094
0003	Andres	Florez	31276543...	aa@gmail...	España	Almería	Abrucena	Carrer...	28047
0004	Carlos	Marin	8345623	acab@131...	España	Cáceres	Abertura	Calle 3...	29092
0005	Elvira	Orozco	32198435...	julian@ba...	España	A Coruña	Ares	Centro	28011
0006	Kevin	Ayala	8354624	lololo@lal...	España	Álava	Murrio	Carrer...	28011
0007	Angela	Hoyos	32176543...	angela_ho...	España	Alicante	Agost	Calle 4...	28024
0008	Marcela	Gomez	11548752	gmail@e...	España	Badajoz	Aceuchal	Centro	28154
125	Carlos	Perez	31276543...	hola@hol...	España	Álava	Aurrio	Carrer...	28012

Figura 4-8 Control de Clientes

SISTEMA DE INVENTARIO

La característica de controlar dónde hace clic el usuario nos permite acceder a la información de este abriendo una ventana para modificar los datos y/o eliminar el cliente. En adición a esta utilidad de dar el acceso a la información para modificar y eliminar comprueba si estos campos son introducidos correctamente o no. En el ejemplo, vemos que el cliente Default tiene unos campos que desde la aplicación no se pueden introducir, puesto que ningún campo está bien.

Estos campos se controlan de igual manera que en el registro: el DNI ha de ser válido, se verifica que los valores de los campos teléfono y código postal sean numéricos, que se elija una provincia y una ciudad válida y que el correo tenga una extensión válida.



Modificar Cliente

Documento: DNI 00000000

Nombre: Cliente

Apellidos: Default

Teléfono: 000000

E-mail: default@default.def

Pais: default

Provincia: default

Ciudad: default

Dirección: Default

Código Postal: 0

Empleado: 1

Modificar Eliminar

Figura 4-9 Modificación de Cliente

SISTEMA DE INVENTARIO

4.1.5 Ventas

En la pantalla de Ventas, uno de los pilares más importantes de la aplicación, sólo disponemos de una pestaña, la de Registrar Ventas, pues ya hay otras para el resto de operaciones.



Figura 4-10 Desplegable Ventas

4.1.5.1 Ventana de venta

En esta ventana para registrar ventas, como vemos, disponemos de un número de factura, el cual se nos genera automáticamente cada vez que iniciamos una venta, y es el identificador único de cada venta. Además, los artículos se añadirán a la tabla derecha y se irán apilando en manera de lista.

Figura 4-11 Venta 1

SISTEMA DE INVENTARIO

La disposición de la ventana está hecha de una manera lógica: Escoger un artículo y la cantidad del artículo que se desea y darle al botón de añadir.

INVENTARIO

Conexión Juan Carlos Toledo 13-06-2020

Cientes Ventas Artículos Proveedores Devolucion Reportes Empleado Recursos Calculadora

Venta

Nº Factura 89108

Artículo Camisetas

Precio 363

Cantidad 2

Añadir Eliminar

Pago Cancelar

Nº	Nombre Artículo	Cantidad	Precio	Total
Tabla sin contenido				

Total

Figura 4-12 Venta 2

El artículo se añadirá en forma de lista, con su precio por unidad indicado al elegir y en la tabla una vez añadido, y el precio de la cantidad de ese producto que se añade a la venta.

INVENTARIO

Conexión Juan Carlos Toledo 13-06-2020

Cientes Ventas Artículos Proveedores Devolucion Reportes Empleado Recursos Calculadora

Venta

Nº Factura 89108

Artículo Selecciona Artículo

Precio 0,00

Cantidad Cantidad

Añadir Eliminar

Pago Cancelar

Nº	Nombre Artículo	Cantidad	Precio	Total
1	Camisetas	2	363.0	726.0

Total 726

Figura 4-13 Venta 3

SISTEMA DE INVENTARIO

Seguido a esto, añadimos un artículo y lo seleccionamos en la tabla y vemos cómo cambia de manera automáticamente el total (sin IVA) de la compra situado en el campo de abajo a la derecha. Seleccionamos el artículo y pulsamos el botón eliminar y este se eliminará de la compra.

The screenshot shows the 'Venta' window in the 'INVENTARIO' application. The window has a dark blue header with the title 'INVENTARIO' and a user profile 'Juan Carlos Toledo' dated '13-06-2020'. Below the header is a navigation bar with tabs: 'Clientes', 'Ventas', 'Articulos', 'Proveedores', 'Devolucion', 'Reportes', 'Empleado', 'Recursos', and 'Calculadora'. The 'Ventas' tab is active. The main area is divided into two sections. On the left, there are input fields for 'Nº Factura' (89108), 'Articulo' (a dropdown menu), 'Precio' (0,00), and 'Cantidad' (a dropdown menu). Below these fields are four buttons: 'Añadir', 'Eliminar', 'Pago', and 'Cancelar'. On the right, there is a table with the following columns: 'Nº', 'Nombre Articulo', 'Cantidad', 'Precio', and 'Total'. The table contains two rows: '1 Camisetas 2 363.0 726.0' and '2 Camisas 2 605.0 1210.0'. The second row is highlighted in yellow. At the bottom right, there is a 'Total' label and a value of '1936'.

Nº	Nombre Articulo	Cantidad	Precio	Total
1	Camisetas	2	363.0	726.0
2	Camisas	2	605.0	1210.0

Figura 4-14 Venta 4

Una vez eliminado el producto de la tabla, es decir, de la lista de la venta, hacemos clic en el botón pago y procederemos a acceder a la ventana para pagar la venta.

The screenshot shows the 'Venta' window in the 'INVENTARIO' application after removing one item. The window layout is the same as in Figure 4-14. The table now only contains one row: '1 Camisetas 2 363.0 726.0'. The total value at the bottom right is now '1210'.

Nº	Nombre Articulo	Cantidad	Precio	Total
1	Camisetas	2	363.0	726.0

Figura 4-15 Venta 5

SISTEMA DE INVENTARIO

4.1.5.2 Ventana de pago

En esta ventana, se ve el precio sin IVA de la compra y, además, el precio total de la compra una vez añadido el porcentaje de IVA que le corresponde a cada artículo según su tipo de artículo.

Metodo de Pago

Importe:

ID Transacción: 89108

N° Empleado: 1

Forma de Pago: EFECTIVO

Fecha: 13-06-2020

IVA	Base Imponible	Cuota
4%	0.0€	0.0€
10%	0.0€	0.0€
21%	210.0€	1000.0€
Total sin IVA	210.0 €	1000.0 €
Total		1210 €
Importe		€
Por entregar		0.0 €

Pagar

Registrar más

Anular

Figura 4-16 Pago 1

Esta ventana de pago es interactiva, pues va cambiando según escribes el importe que ha pagado el cliente. De esta manera, tratándose de Efectivo, seleccionado en el Combo box de opciones, sale la alerta en verde de que faltan por entregar 10€.

Metodo de Pago

Importe: 1200

ID Transacción: 89108

N° Empleado: 1

Forma de Pago: EFECTIVO

Fecha: 13-06-2020

IVA	Base Imponible	Cuota
4%	0.0€	0.0€
10%	0.0€	0.0€
21%	210.0€	1000.0€
Total sin IVA	210.0 €	1000.0 €
Total		1210 €
Importe		1200 €
Por entregar		10.0 €

Pagar

Registrar más

Anular

Figura 4-17 Pago 2

SISTEMA DE INVENTARIO

Este valor va cambiando, contando incluso con que el cliente pague de más, entonces pondrá la cantidad de efectivo que hay que retirar de la caja para dar de cambio al cliente.

INVENTARIO

— □ ×

Conexión

Juan Carlos Toledo

13-06-2020

Cientes

Ventas

Articulos

Proveedores

Devolucion

Reportes

Empleado

Recursos

Calculadora

Metodo de Pago

Importe

1215

ID Transacción

89108

Nº Empleado

1

Forma de Pago

ELECTIVO

Fecha

13-06-2020

IVA	Base Imponible	Cuota
4%	0.0€	0.0€
10%	0.0€	0.0€
21%	210.0€	1000.0€
Total sin IVA	210.0 €	1000.0 €
Total		1210 €
Importe		1215 €
A retirar		5.0 €

Pagar

Registrar más

Anular

Figura 4-18 Pago 3

Los botones situados a la derecha en la ventana de pago dan una versatilidad a esta pantalla, pues cada uno sirve para una cosa. El de pagar finaliza la venta y genera un tique, un botón que sirve para ampliar la venta más si se desea, un botón que anula la venta y el botón de la impresora. El botón de Registrar más se refiere a productos, pues se pulsa y lleva a la pantalla anterior:

[illegible]

Figura 4-19 Pago 4

SISTEMA DE INVENTARIO

En caso de que no se quiera añadir nada más o sí, sólo habría que darle a pago nuevamente. El botón de anular anula la venta y genera otra factura y venta nueva, y el de impresora, como vemos en la siguiente imagen, señala con una X en rojo que no se imprimirá el tique de venta.

Metodo de Pago

Importe: 1210

ID Transacción: 89108

N° Empleado: 1

Forma de Pago: EFFECTIVO

Fecha: 13-06-2020

IVA	Base Imponible	Cuota
4%	0.0€	0.0€
10%	0.0€	0.0€
21%	210.0€	1000.0€
Total sin IVA		210.0 €
Total		1210 €
Importe		1210 €
A retirar		0.0 €

Pagar

Registrar más


Anular

Figura 4-20 Pago 5

SISTEMA DE INVENTARIO

4.1.5.3 Tique de compra o factura simplificada

Para este ejemplo, como es obvio, se busca mostrar todas las funcionalidades de la ventana, por lo que se imprime el tique. El tique se abre en un archivo pdf, el cual se puede imprimir, guardar o exportar a otro tipo de archivos. Este es el formato de tique que tiene Sistema de Inventario y Venta en un principio, pero es totalmente modificable.


J&A S.L.

Dirección: Calle Orense 64
Tefl: 999999999
Nº Operación: 89108
13/6/20
FACTURA SIMPLIFICADA

Cand.	Articulo	Precio C/ud (€)	Importe(€)
2	Camisetas	363.0	726

Detalle (€)	TOTAL SIN IVA:	600.0 €
	TOTAL IVA:	1210.00 €

Juan Carlos
Le atendio:

Figura 4-21 Tique Ejemplo

A pesar de decirlo en la venta, por si eso fallara, nosotros hemos decidido introducir una alerta con el cambio a recibir o a entregar. Una vez mostrado el cambio, la venta se registra automáticamente a la base de datos con todos los datos de la venta.

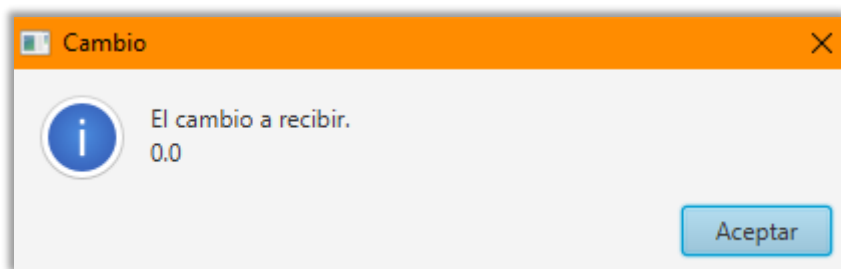


Figura 4-22 Alerta de Cambio

SISTEMA DE INVENTARIO

4.1.6 Artículos

La ventana de Artículos posee un desplegable con 2 opciones: el registro de artículos y la ventana de consulta y control de stock.

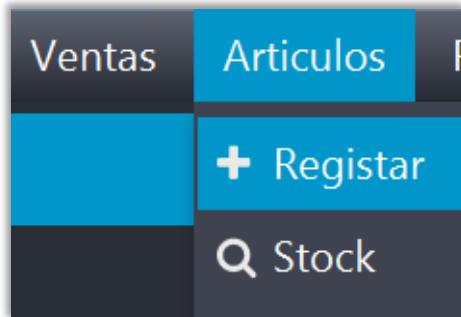


Figura 4-23 Desplegable Artículos

4.1.6.1 *Registro de Artículos*

La ventana de Registro dispone varios campos, todos controlados para no insertar datos corruptos a la base de datos. En un principio, se dan opciones de nombres de artículos los cuales distribuyen los proveedores. De esta manera se controla que sólo se registren artículos con el nombre correcto. Además de comprobar cada formato correspondiente a cada campo, se controla el tipo de artículo que es y el proveedor que lo lleva para registrar así el % de IVA imponible para su categoría y el precio de costo de este. Una vez rellenado el formulario, se da al botón guardar si se desea registrar en la base de datos e incluirlo en el stock de la tienda, y a cancelar si se desean limpiar los campos a como se ven ahora.

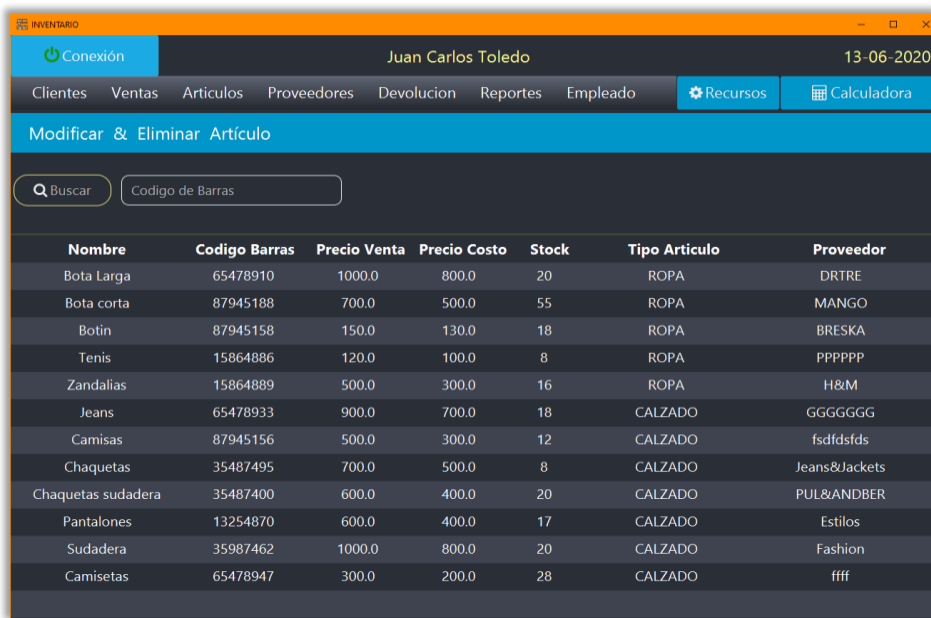
A screenshot of a web application window titled 'INVENTARIO'. The interface has a dark blue header with a 'Conexión' status icon and the user name 'Juan Carlos Toledo' and date '13-06-2020'. Below the header is a navigation bar with tabs: 'Clientes', 'Ventas', 'Articulos' (active), 'Proveedores', 'Devolucion', 'Reportes', 'Empleado', 'Recursos' (with a gear icon), and 'Calculadora' (with a calculator icon). The main content area is titled 'Artículo' and contains a form with the following fields: 'Nombre Artículo' (dropdown menu showing 'Articulo'), 'Tipo Artículo' (dropdown menu showing 'Tipo de Artículo'), 'Precio Venta' (text input), 'Proveedor' (dropdown menu showing 'Proveedor'), 'Precio Costo' (text input), 'Fecha Ingreso' (text input showing '13-06-2020'), 'Stock' (text input), and 'Codigo Barras' (text input). At the bottom of the form are two buttons: 'Guardar' (with a save icon) and 'Cancelar' (with an 'x' icon).

Figura 4-24 Registro de Artículos

SISTEMA DE INVENTARIO

4.1.6.2 Control de Stock

En esta ventana se controla el stock, pudiendo modificar información errónea de los artículos registrados, si fuera necesario. Sigue el mismo formato que la ventana de Clientes, incluso con el mismo funcionamiento: una tabla que se puede ordenar y agrandar al gusto, se puede buscar el artículo por código de barras y si se da doble clic a una fila de artículo, se abre una ventana para modificar los datos del producto o eliminarlo de la lista de stock si fuera necesario.



Nombre	Codigo Barras	Precio Venta	Precio Costo	Stock	Tipo Artículo	Proveedor
Bota Larga	65478910	1000.0	800.0	20	ROPA	DRTRE
Bota corta	87945188	700.0	500.0	55	ROPA	MANGO
Botin	87945158	150.0	130.0	18	ROPA	BRESKA
Tenis	15864886	120.0	100.0	8	ROPA	PPPPPP
Zandalias	15864889	500.0	300.0	16	ROPA	H&M
Jeans	65478933	900.0	700.0	18	CALZADO	GGGGGGG
Camisas	87945156	500.0	300.0	12	CALZADO	fsdfdsfds
Chaquetas	35487495	700.0	500.0	8	CALZADO	Jeans&Jackets
Chaquetas sudadera	35487400	600.0	400.0	20	CALZADO	PUL&ANDBER
Pantalones	13254870	600.0	400.0	17	CALZADO	Estilos
Sudadera	35987462	1000.0	800.0	20	CALZADO	Fashion
Camisetas	65478947	300.0	200.0	28	CALZADO	ffff

Figura 4-25 Control de Artículos

Mediante la selección del producto la pantalla de modificación o supresión del producto se rellena con los datos y se superpone sobre esta ventana, de manera que hasta que no se cierre esta ventana no se continúa con otra operación. Una vez hecha la operación se registra en la base de datos y se actualiza la tabla de control de Stock.



Modificar Artículo

Nombre: Camisas

Codigo Barras: 87945156

Precio Venta: 500.0

Precio Costo: 300.0

Stock: 12

Articulo: CALZADO

Proveedor: fsdfdsfds

Fecha Ingreso: 2020-10-20

Modificar Eliminar

Figura 4-26 Modificar Artículo

SISTEMA DE INVENTARIO

4.1.7 Proveedores

Esta ventana tiene 3 desplegables, cada uno con unas características para generar una agenda de proveedores correctamente y que la inserción de artículos en la ventana **Artículos** no sea corrupta.

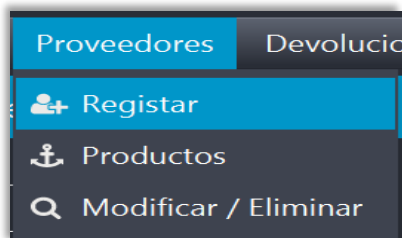


Figura 4-27 Desplegable Proveedores

4.1.7.1 Registro de Proveedores

Siguiendo el formato de las anteriores, esta ventana permite registrar los datos de los proveedores correctamente mediante la comprobación de campos. El tipo de documento, junto con la provincia y la ciudad se controla con el mismo sistema de Combo box que se lleva a cabo en el **registro de clientes**, así como el resto de los campos.

La peculiaridad de este registro sucede con el Combo box de Productos, que puede parecer extraño, pero tras varias pruebas y comprobaciones, dimos con que la mejor manera de registrar los productos que suministra un proveedor es registrando antes el tipo de Artículo con la siguiente ventana. De esta forma no habrá errores a la hora de escribir el producto.

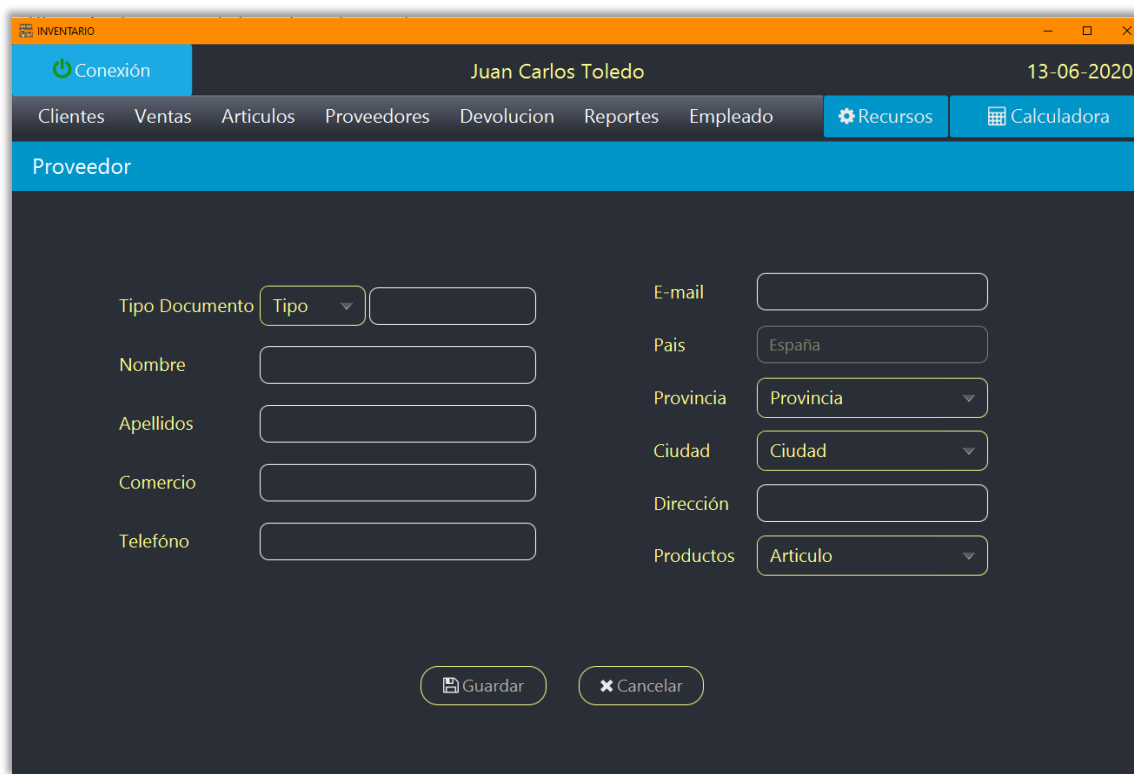
Una imagen de una ventana de software titulada 'INVENTARIO'. La barra superior es naranja y contiene un botón 'Conexión' con un icono de enchufe, el nombre de usuario 'Juan Carlos Toledo' y la fecha '13-06-2020'. Debajo hay una barra de navegación con pestañas: 'Clientes', 'Ventas', 'Articulos', 'Proveedores' (seleccionada), 'Devolucion', 'Reportes', 'Empleado', 'Recursos' (con un icono de engranaje) y 'Calculadora' (con un icono de calculadora). El título de la ventana principal es 'Proveedor'. El formulario contiene campos para: 'Tipo Documento' (un desplegable con 'Tipo' seleccionado), 'Nombre', 'Apellidos', 'Comercio', 'Teléfono', 'E-mail', 'País' (con 'España' seleccionado), 'Provincia' (un desplegable con 'Provincia' seleccionado), 'Ciudad' (un desplegable con 'Ciudad' seleccionado), 'Dirección' y 'Productos' (un desplegable con 'Artículo' seleccionado). En la parte inferior hay dos botones: 'Guardar' con un icono de disco y 'Cancelar' con un icono de X.

Figura 4-28 Registro Proveedores

SISTEMA DE INVENTARIO

4.1.7.2 Registro de Productos de Proveedores

Este es el epicentro en cuanto a registro de artículos. En esta sección se registran los artículos, con el tipo de artículo que es (parafarmacia, ropa, cosméticos...), se le introduce el % de IVA que lleva ese tipo de artículo impuesto según ley y el nombre de este artículo.

Registrando así el artículo evitamos que haya errores en registros de productos que suministra un proveedor o registros de artículos en el stock, los cuáles no están registrados. De momento hemos decidido poner esta ventana aquí, puesto que sería el paso previo para registrar un proveedor en caso de que se quiera, y consideramos que es también el paso anterior para registrar un artículo de un nuevo proveedor. Por ende, primero se pasa por esta ventana, seguidamente por el **registro de proveedor** que provee este artículo y por último por el **registro de artículos** si se quisiera registrar una entrada de este artículo en Stock.

The screenshot shows a web application window titled 'INVENTARIO'. The top navigation bar is orange and contains a 'Conexión' button with a power icon, the user name 'Juan Carlos Toledo', and the date '13-06-2020'. Below this is a dark grey menu bar with buttons for 'Clientes', 'Ventas', 'Articulos', 'Proveedores', 'Devolucion', 'Reportes', 'Empleado', 'Recursos' (with a gear icon), and 'Calculadora' (with a calculator icon). The main content area has a blue header labeled 'Articulo'. It is divided into two columns by a vertical line. The left column is titled 'Tipo de Artículo' and contains two input fields: 'Tipo Articulo' and 'IVA'. The right column is titled 'Registrar Articulo' and contains two input fields: 'Nombre Articulo' and 'Codigo de Barras' (which has the value '424490333' pre-filled). At the bottom center, there is a button with a floppy disk icon and the text 'Registrar'.

Figura 4-29 Registro Tipo de Artículo

SISTEMA DE INVENTARIO

4.1.7.3 Control de Proveedores

Al igual que las otras ventanas de control, presenta una tabla con el listado de proveedores, con las mismas características, una búsqueda por el documento del proveedor y una ventana para modificar datos del proveedor o suprimirlo de la lista.

Documento	Nombre	Apellido	Comercio	Teléfono	E-mail	País	Dirección	Producto
00004-4	Camilo	Muñoz	Jeans&Jackets	3219843...	hh@adolf.reich	España	Centro	Chaquetas
00001-1	Carlos	Maya	Estilos	11548752	mason@illumina...	España	Calle 12-12	Pantalones
00006-6	Carlos	sanchez diaz	jeans&&jeans	8213423	eeee@iiii.es	España	calle 45 N 23-23	Pantalones
00006-7	Carlos	sanchez diaz	PUL&ANDBER	8213423	eeee@iiii.es	España	calle 45 N 23-23	Chaquetas su
00006	Carlos	sanchez diaz	PPPPPP	8213423	eeee@iiii.es	España	calle 45 N 23-23	Tenis
2222	cccc	ddfff	ffff	6684756	uuuu@jjjj.es	España	calle dddd	Camisetas
2221	cccc	ddfff	DRTRE	6684756	uuuu@jjjj.es	España	calle dddd	Bota Larga
2226	cccc	ddfff	GGGGGGG	6684756	uuuu@jjjj.es	España	calle dddd	Jeans
43434	fsdfsdfsdfs	fsdfsdfs	fsdfsdfs	123458	rafa@nadal.ten	España	calle fsdfsdfs	Camisas
43435	fsdfsdfsdfs	fsdfsdfs	MANGO	123458	rafa@nadal.ten	España	calle fsdfsdfs	Bota Corta
43437	fsdfsdfsdfs	fsdfsdfs	BRESKA	123458	rafa@nadal.ten	España	calle fsdfsdfs	Botin
00002-2	Marino	Burbano	Sports	3217654...	kkkkk@lanlan.kkk	España	Centro	Zapatos

Figura 4-30 Control de Proveedores

SISTEMA DE INVENTARIO

4.1.8.2 Registro de Devoluciones por Artículo

En el diseño de esta ventana se busca la factura al igual que en el *anterior* método. La gran diferencia de este registro de devolución al otro es que funciona por artículo seleccionado en la tabla. Esta tabla se rellena al igual que la de la ventana anterior, pero con el producto que selecciones de la factura, se carga el código de barras en su campo y tienes que poner la cantidad que quieres devolver. De esta manera, puedes devolver todos los artículos que tengan el nombre o parte de ellos, por si la devolución fuera parcial.

[illegible]

Figura 4-33 Devolución por Artículo

SISTEMA DE INVENTARIO

4.1.9 Reportes

Esta sección obtiene mucho peso en la aplicación, pues sugiere mucha versatilidad al ofrecer reportes según varios campos de búsqueda y ofrece reportes que pueden servir para análisis económico de la empresa.

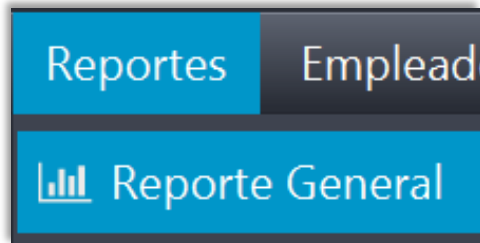


Figura 4-34 Desplegable Reportes

Como se puede apreciar en la siguiente imagen se permiten hacer reportes para todo tipo de información acerca del comercio. Esto permite obtener informes a tiempo real sobre cualquiera de estas opciones, pudiendo guardarlos en pdf, imprimirlos o exportarlos en diversos formatos.

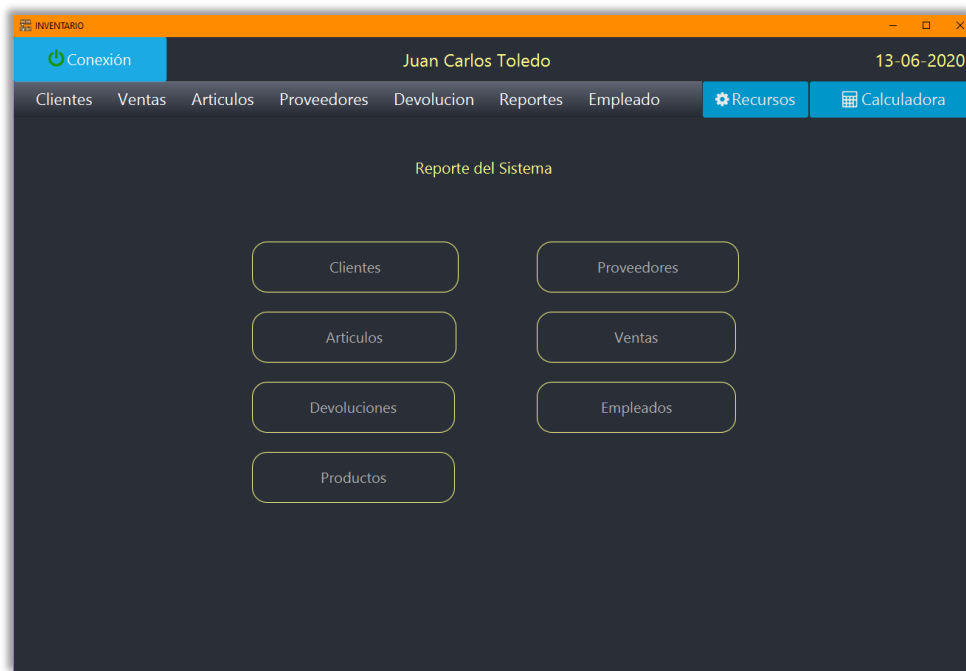


Figura 4-35 Reportes

SISTEMA DE INVENTARIO

4.1.9.1 Reporte de Clientes

Muestra un reporte de todos los clientes y sus datos a tiempo real, mostrando además la fecha del momento en el que se genera el reporte.



Clientes


Tip	Documento	Nombre	Apellidos	Telefono	E-mail	Pais	Ciudad	Localidad	Direccion	CP
DNI	125	Carlos	Perez	312765432	hola@hola	España	Álava	Aurrio	Carrera 23	28012
DNI	000032	dddfff	ffff	77777	ffff@aaaaa	España	Álava	Añana	ffff	28014
DNI	0001	Camilo	Lopez	8239578	zzzzzz@bb	España	Albacete	Gineta (La)	Calle 23 N	28012
NIE	0002	Maria	Arango	8253478	email@ema	España	Álava	Murrio	Carrera 5 N	28094
DNI	0003	Andres	Florez	312765432	ss@lenin.	España	Almería	Abrucena	Carrera 23	28047
DNI	0004	Carlos	Marin	8345623	acab@1312	España	Cáceres	Abertura	Calle 3 N	29092
DNI	0005	Elvira	Orozco	321984354	julian@bast	España	A Coruña	Ares	Centro	28011
DNI	0006	Kevin	Ayala	8354624	lololo@lalal	España	Álava	Murrio	Carrera 23	28011
DNI	0007	Angela	Hoyos	321765430	lalalal@lolol	España	Alicante	Agost	Calle 4 N	2328024
NIE	0008	Marcela	Gomez	11548752	gmail@ema	España	Badajoz	Aceuchal	Centro	28154

sábado 13 junio 2020 Page 1 of 1

Figura 4-36 Reporte de Clientes

4.1.9.2 Reporte de Artículos

Muestra un reporte de todos los artículos y sus peculiaridades a tiempo real, mostrando además la fecha del momento en el que se genera el reporte.



Articulos

Nº	Articulo	PrecioV(€)	PrecioC(€)	Stock	StockMin	StockMax	Tip. Articulo	IVA	Comercio	CodigoBarras	Fecha
1	Bota Larga	1000.00	800.00	20	5	30	ROPA	21	DRTRE	65478910	11/3/12
2	Bota corta	700.00	500.00	55	5	30	ROPA	21	MANGO	87945188	10/11/09
3	Botin	150.00	130.00	18	5	30	ROPA	21	BRESKA	87945158	9/1/12
4	Tenis	120.00	100.00	8	5	30	ROPA	21	PPPPPP	15864886	10/2/10
5	Zandalias	500.00	300.00	16	5	30	ROPA	21	H&M	15864889	11/1/12
6	Jeans	900.00	700.00	18	5	30	CALZADO	21	GGGGGGG	65478933	10/2/20
7	Camisas	500.00	300.00	12	5	30	CALZADO	21	fsdfdsfds	87945156	10/20/20
8	Chaquetas	700.00	500.00	8	5	30	CALZADO	21	Jeans&Jacke	35487495	11/2/09
9	Chaquetas	600.00	400.00	20	5	30	CALZADO	21	PUL&ANDBE	35487400	10/23/11
10	Pantalones	600.00	400.00	17	5	30	CALZADO	21	Estilos	13254870	10/23/11
11	Sudadera	1000.00	800.00	20	5	30	CALZADO	21	Fashion	35987462	10/23/12
12	Camisetas	300.00	200.00	28	5	30	CALZADO	21	ffff	65478947	10/3/10
14	Pantalones	250.00	120.00	25	5	30	ROPA	21	jeans&&jeans	13254870	6/13/20

sábado 13 junio 2020 Page 1 of 1

Figura 4-37 Reporte de Artículos

SISTEMA DE INVENTARIO

4.1.9.3 Reporte de Devoluciones

Muestra un reporte de todas las devoluciones, ya sean por tique o por artículo a tiempo real, mostrando la fecha de devolución, el dinero devuelto y quién realizó la devolución.



Devoluciones


Cod. Factura	Articulo	Motivo	Cantidad	Total(€)	Fecha	Empleado
9910966	Botin	pequeño	1	1500.00	1/14/13	Juan Carlos

sábado 13 junio 2020 Page 1 of 1

Figura 4-38 Reporte de Devoluciones

4.1.9.4 Reporte de Productos

Muestra un reporte de todos los productos que existen en el comercio con menos detalle que en artículos, mostrando además la fecha del momento en el que se genera el reporte.



Productos

N°	Articulo	Codigo de Barras
1	Pantalon	13254870
2	Zapatos	47922285
3	Sudadera	35987462
4	Chaquetas	35487495
5	Polos	99632582
6	Camisetas	65478947
7	Camisas	87945156
8	Zandalias	15864889
9	Chaquetas sudadera	35487400
10	Bota Larga	65478910
11	Bota Corta	87945188
12	Tenis	15864886
13	Jeans	65478933
14	Botin	87945156

sábado 13 junio 2020 Page 1 of 1

Figura 4-39 Reporte de Productos

SISTEMA DE INVENTARIO

4.1.9.5 Reporte de Proveedores

Genera un reporte de todos los proveedores que suministran al comercio con todos los datos.



Proveedores


Tip.	Document	Nombre	Apellidos	Comercio	Telefono	Email	País	Ciudad	Localidad	Dirección	Artículo
DNI	43437	fsdfsdfsdf	fsdfsdfs	BRESKA	123458	rafa@nada	España	Melilla	Melilla	calle	Botín
DNI	2226	cccc	ddfff	GGGGGG	6684756	uuuu@jjj.	España	Melilla	Melilla	calle dddd	Jeans
CIF	00006	Carlos	sanchez	PPPPPP	8213423	eeee@iiii.	España	Melilla	Melilla	calle 45 N	Tenis
DNI	43435	fsdfsdfsdf	fsdfsdfs	MANGO	123458	rafa@nada	España	Melilla	Melilla	calle	Bota Corta
DNI	2221	cccc	ddfff	DRTRE	6684756	uuuu@jjj.	España	Melilla	Melilla	calle dddd	Bota Larga
CIF	00006-7	Carlos	sanchez	PUL&AND	8213423	eeee@iiii.	España	Melilla	Melilla	calle 45 N	Chaquetas
CIF	00005-8	Marlon	Brandom	H&M	8354624	gggg@tttt.	España	Melilla	Melilla	Calle 34 B	Zandalias
DNI	43434	fsdfsdfsdf	fsdfsdfs	fsdfsdfs	123458	rafa@nada	España	Melilla	Melilla	calle	Camisas
DNI	2222	cccc	ddfff	ffff	6684756	uuuu@jjj.	España	Melilla	Melilla	calle dddd	Camisetas
CIF	00006-6	Carlos	sanchez	jeans&jea	8213423	eeee@iiii.	España	Melilla	Melilla	calle 45 N	Pantalones
CIF	00005-F	Marlon	Brandom	Bellas	8354624	gggg@tttt.	España	Melilla	Melilla	Calle 34 B	Polos
NIE	00004-4	Camilo	Muñoz	Jeans&Jac	321984354	hh@adolf.	España	Melilla	Melilla	Centro	Chaquetas
NIE	00003-A	Paola	Sanchez	Fashion	7517454308888	@hh.	España	Melilla	Melilla	Veraneras	Sudadera

sábado 13 junio 2020 Page 1 of 2

Figura 4-40 Reporte de Proveedores

4.1.9.6 Reporte de Ventas

Genera un informe de las ventas realizadas en el comercio, mostrando el número de factura correspondiente, con la forma de pago, la fecha, el total de la operación y el empleado que realizó la venta.



Ventas

Nº Factura	Forma Pago	Fecha	Total(€)	Empleado
9910966	EFFECTIVO	13/1/13	1076.90	Juan Carlos
89108	EFFECTIVO	13/6/20	1210.00	Juan Carlos

sábado 13 junio 2020 Page 1 of 1

Figura 4-41 Reporte de Ventas

SISTEMA DE INVENTARIO

4.1.9.7 Reporte de Empleados

Genera un reporte detallado de los empleados que hay en el comercio registrados.



Empleados

CodigoEmpleado	tipo	DNI_NIE	Nombre	Apellidos	Email	Puesto
1	DNI	99999999A	Juan Carlos	Toledo	hola@mundo.com	Administrador
2	NIE	D8888888B	Angel Bruno	Lopez	hola@mundo.es	Usuario

sábado 13 junio 2020 Page 1 of 1

Figura 4-42 Reporte de Empleados

4.1.10 Control de Empleados

La ventana de Empleados ofrece 2 opciones similar a las ventanas de control de clientes y de artículos. Una ventana para registrar empleados y concederle distintos permisos y otra para el control de los empleados.

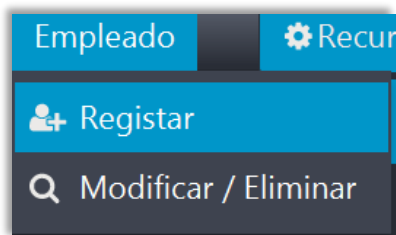


Figura 4-43 Desplegable de Empleados

SISTEMA DE INVENTARIO

4.1.10.1 Registro de Empleados

En el registro de empleados, al igual que en las anteriores ventanas, se comprueban los campos para evitar datos corruptos en la base de datos. Además, hay 3 peculiaridades en el registro de empleados: el código de empleado es generado aleatoriamente cada vez que se inicia la pantalla y es posible cambiarlo a gusto del empleado, el puesto ofrece varias opciones que van unidas a los permisos que se conceden al empleado para utilizar el sistema y el botón Registrar, cuando ha pasado los filtros de campos y se ha verificado la inserción, abre una alerta la cual asegura que se desea dar el puesto a ese empleado y seguido a esta, se abre la ventana que pide la contraseña del usuario, dejando que el usuario escoja su propia contraseña desde un inicio.

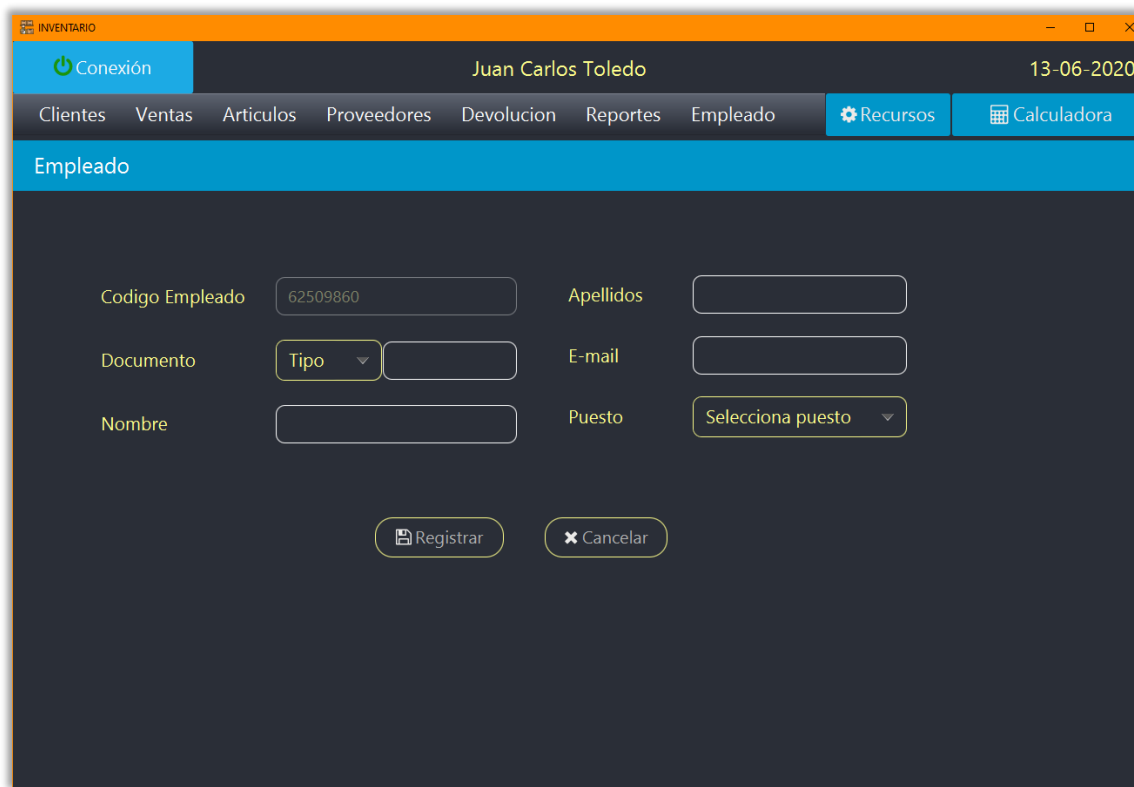


Figura 4-44 Registro de Empleado

Esta alerta confirma que el puesto seleccionado para el empleado a registrar no se ponga por error y da paso a la ventana de contraseña en caso de que se afirme y vuelve a la ventana de registro de empleados en caso de que se niegue la pregunta.

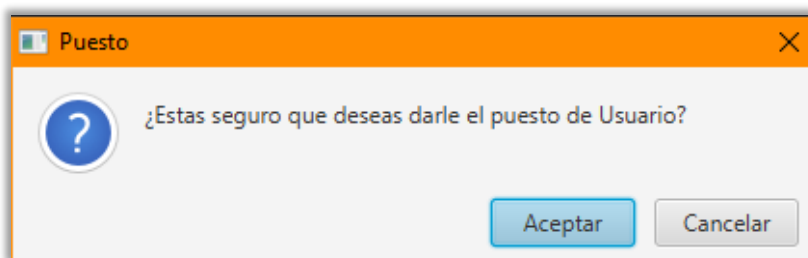



Figura 4-45 Confirmación de Puesto de Empleado

SISTEMA DE INVENTARIO

En esta ventana que se superpone a todas se registra la base de datos la contraseña que el Administrador decida poner al empleado o directamente la que el empleado quiera. En ambos campos se toma el formato de ocultar la contraseña mediante unos puntos por carácter de la contraseña, se pide que la contraseña tenga, al menos 1 mayúscula y un mínimo de 8 caracteres y se comprueba si ambas son las mismas.

Una vez comprobados los campos, la contraseña queda registrada en la base de datos y enlazada al empleado.



El formulario, titulado "Registrar Contraseña", se presenta sobre un fondo oscuro. Incluye tres campos de entrada: "Empleado" con el valor "14486055", "Contraseña" y "Repetir Contraseña", ambos vacíos. En la parte inferior hay dos botones: "Guardar" con un ícono de disco y "Cancelar" con un ícono de X.

Empleado	14486055
Contraseña	
Repetir Contraseña	



 Guardar  Cancelar

Figura 4-46 Registro de Contraseña de Empleado/Usuario

SISTEMA DE INVENTARIO

4.1.10.2 Control de Empleados

En esta ventana, al igual que el resto de las ventanas de control, se emplea la tabla para el listado de empleados, un campo de búsqueda automática por código de empleado y si se hace doble clic sobre un empleado se pueden modificar datos o eliminar de la base de datos.

Nº Empleado	Documento	Nombre	Apellido	E-mail	Puesto
1	9999999A	Juan Carlos	Toledo	hola@mundo.com	Administrador
2	D8888888B	Angel Bruno	Lopez	hola@mundo.es	Usuario
14486055	12458965L	Fernando	Simón	fernando.s@gmail.com	Usuario

Figura 4-47 Control de Empleados

Al igual que el resto de las ventanas, esta ventana permite modificar datos del empleado fácilmente.

Modificar Empleado

Nº Empleado: 14486055

Documento: DNI 12458965L

Nombre: Fernando

Apellidos: Simón

E-mail: fernando.s@gmail.com

Puesto: Usuario

Modificar Eliminar

Figura 4-48 Modificación de Empleado

SISTEMA DE INVENTARIO

4.1.11 Recursos

La ventana recursos es una de las que más puede utilizarse a menudo, pues los recursos que contiene son muy útiles.

En la parte izquierda los botones para: generar el último ticket por si este no se hubiera impreso, un botón para generar una factura, un generador de códigos de barras por si fuera necesario, un botón que envía un correo al destinatario que escribas con una plantilla de correo establecida por la empresa y, por último, el botón de los engranajes, que abre la información que se encuentra a la derecha, configuración de la conexión a la base de datos y el correo, con botones para comprobar que funcionan.

Figura 4-49 Recursos 1

4.1.11.1 Último Tique

Este botón da utilidad para cuando el sistema o la propia impresora de tiques fallas y no lo saca, entonces lo pulsas y te genera el último tique procesado.

SISTEMA DE INVENTARIO

4.1.11.2 Factura

En esta ventana flotante que se carga a la derecha, se destina un formato para imprimir esa factura. Con los campos factura y DNI se busca la factura a la base de datos y se rellena la tabla con los datos de la factura, y si el usuario quisiera, le da al botón imprimir y se genera el documento para imprimir en tamaño A4.

Artículo	Cantidad	Precio	Total
Camisetas	2	363.0	726.0

Figura 4-50 Recurso de Factura

4.1.11.3 Generados de Códigos

Esta herramienta permite al usuario crear códigos de barras con forma de etiqueta para cada uno de los artículos cargados en el Combo box de opciones o generar códigos de barras para todos los artículos y guardarlos en formato pdf en el terminal.

Artículo: Bota Larga

Generar CodeBar Generar Todo

Figura 4-51 Código por Artículo

SISTEMA DE INVENTARIO

Como se puede observar, se genera una etiqueta con el precio real de cada producto por unidad y con un código de barras aleatorio.



Figura 4-52 Código de Barras de Todos los Artículos

4.1.11.4 Correo empresarial

El correo empresarial sirve para que el dueño pueda enviar un correo con la cuenta configurada, respetando la plantilla empresarial y pudiendo adjuntar todo tipo de archivos.

The image shows a screenshot of an email composition interface. The header fields are filled with the following information:

- De:** funakyangel@gmail.com
- Para:** correo electronico
- Asunto:** asunto

Below the header fields, there is a large empty text area for the email body. To the right of the text area, there is a button labeled "Adjunto" with a paperclip icon. At the bottom right of the interface, there is a button labeled "Enviar" with a paper plane icon.

Below the email body text area, the following text is visible:

Comercial 4 Cantos S.L.
No responda a este mensaje.

Figura 4-53 Recurso de Correo

SISTEMA DE INVENTARIO

4.1.12 Calculadora

Por último, la utilidad que puede obtener un gran uso es la de la calculadora. Esta ventana emergente se superpone a la actual y permite al usuario realizar cálculos básicos con los botones numéricos del teclado y obtener los resultados al instante, por grande que sea la operación. La Calculadora presenta un modelo minimalista pero sutil.

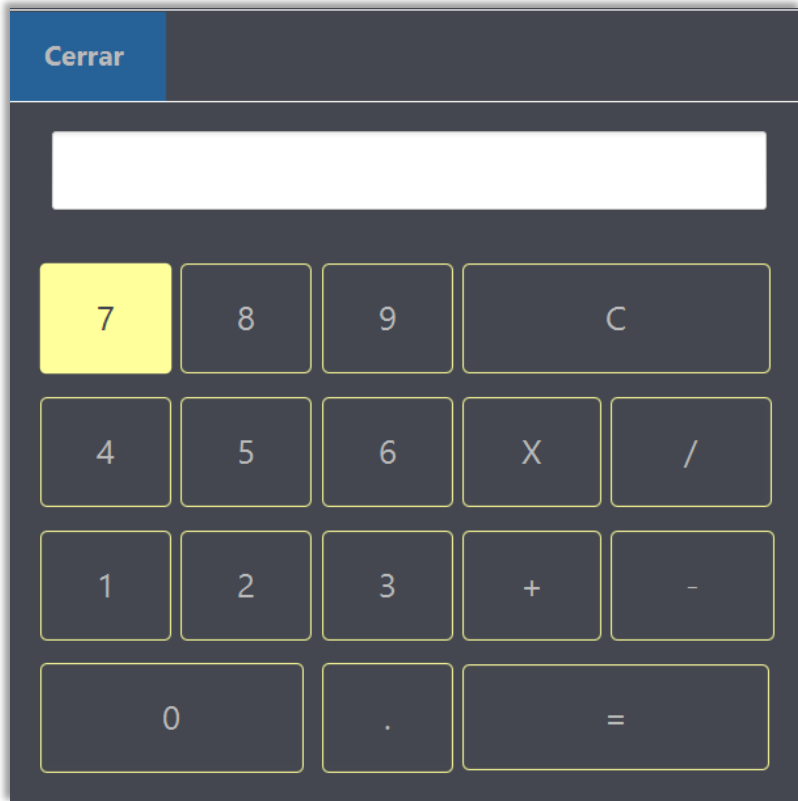


Figura 4-54 Calculadora

SISTEMA DE INVENTARIO

4.2 Diseño del modelo de datos.

El modelo de base de datos relacional ha sido modificado varias veces hasta conseguir un modelo final que determina una base de datos fiable, completa, estructurada y segura. Este modelo entidad-relación de la base de datos de MySQL muestra varias tablas pensadas y hechas a conciencia.

Para empezar, controlamos los artículos agrupándolos por tipos de artículos, los cuales son añadidos unidos al proveedor que los suministra y enlazados con la tabla producto. Estos artículos forman parte de una factura, ya sea porque se vende, que entonces entra en la factura a través del id del artículo por detalle factura y se especifica cómo ha sido pagada la venta en forma de pago, o porque se devuelve, que entonces entra en devolución por el id mediante la tabla detalle factura.

Todo este movimiento de artículos y facturas está controlado por los empleados que realizan estas transacciones, los cuales disponen de un puesto de empleado y un usuario y una contraseña para el login inicial del sistema. Además, para determinar los tipos de documentos de los empleados, los proveedores y de los clientes hemos creado la tabla tipo de documento, acabando así con las inserciones repetitivas de valores en las 3 tablas.

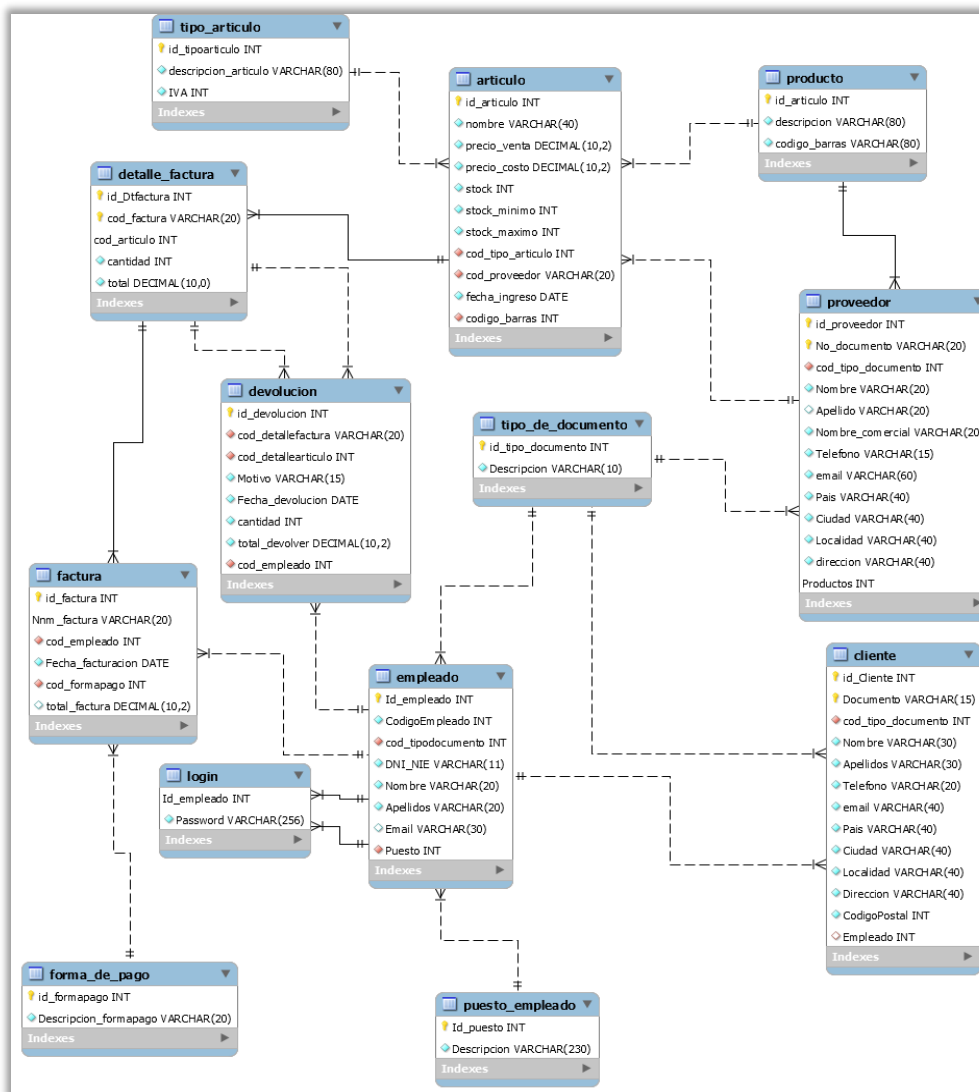


Figura 4-55 Modelo Entidad-Relación Sistema de Inventario

SISTEMA DE INVENTARIO

4.3 Otros diagramas

4.3.1 Modelo de la Base de Datos de España

El siguiente modelo de base de datos concuerda con la base de datos de España. Mediante esta base de datos conseguimos controlar los datos sobre todos los municipios y provincias que hay en España. La base de datos va conjunta a nuestro proyecto, pues de esta manera rellenamos los Combo box que pertenecen a cada una de las tablas y manejamos que no se introduzcan datos erróneos, puesto que, según elijas provincia, tienes los municipios que pertenecen a esa provincia solamente, así no se puede corromper la base de datos de Sistema de Inventario y Venta con direcciones erróneas.

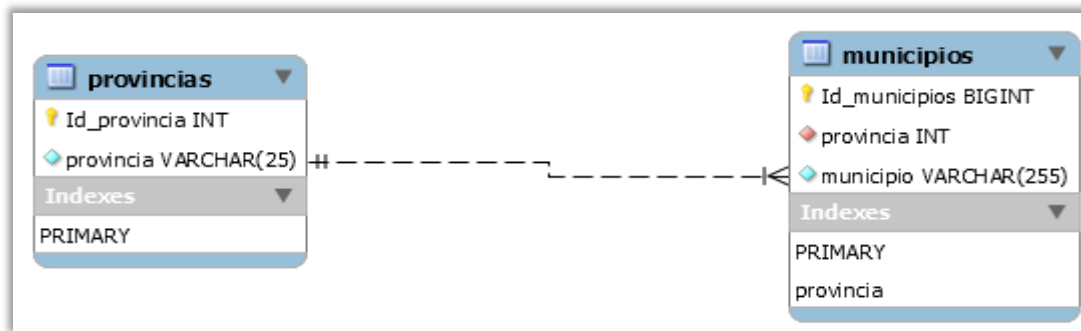


Figura 4-56 Diagrama Entidad-Relación España

4.3.2 JavaDoc de la aplicación

El JavaDoc de la aplicación viene incluido en los archivos de la esta y, además, queda adjuntada la imagen de cómo se ve y el zip en la entrega.

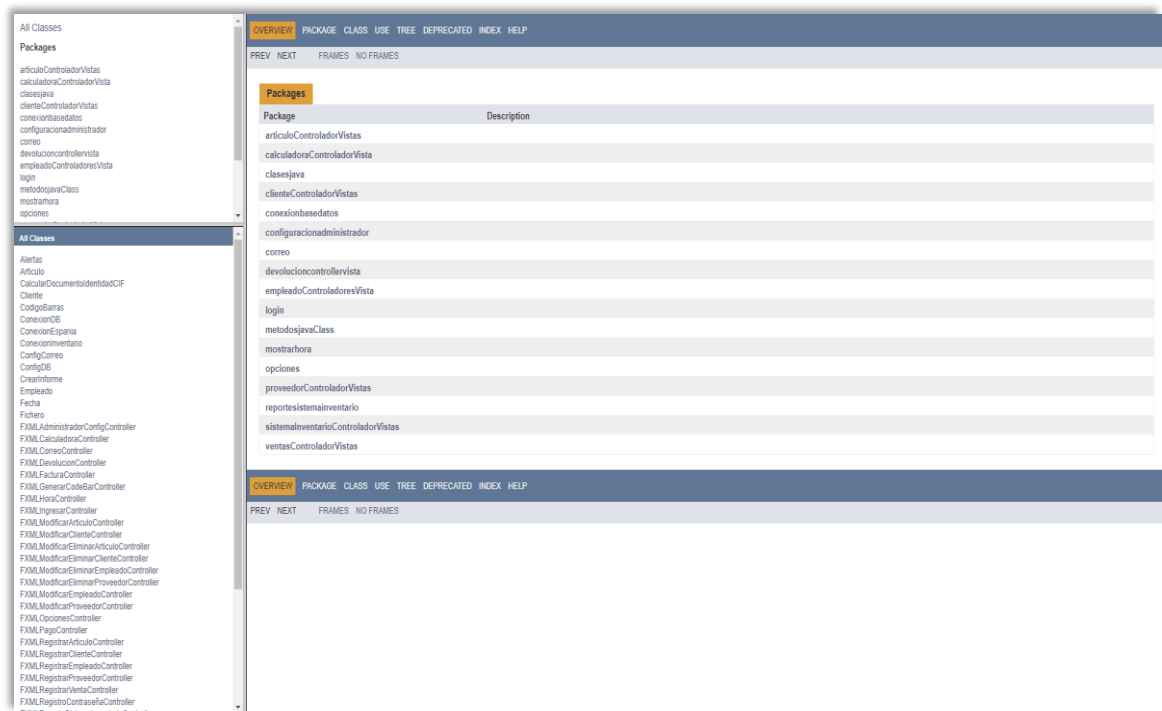


Figura 4-57 JavaDoc del Proyecto

5 Aspectos relevantes del sistema

Para concretar el rendimiento y uso de recursos de la aplicación hemos usado las herramientas que NetBeans IDE nos ofrece. De esta manera, sabemos el rendimiento de la aplicación y los aspectos más relevantes de esta.

La herramienta profile nos permite monitorizar y analizar cada componente de nuestro programa.

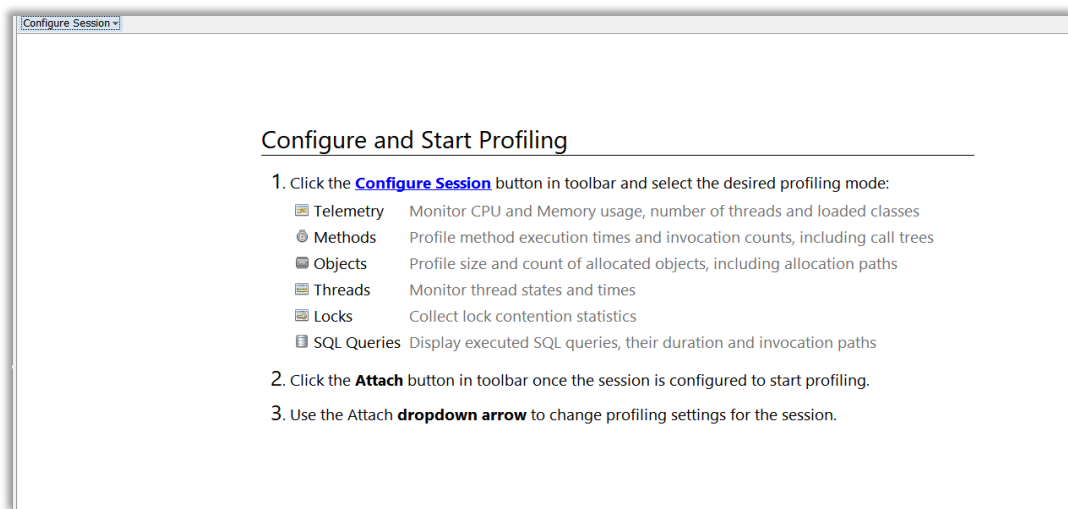


Figura 5-1 Profile

TELEMETRY

Nos permite monitorizar y analizar el rendimiento de la aplicación. Indica cuánto de memoria y CPU utiliza, el tiempo de ejecución de hilos y de las clases que existen al iniciar el programa.

Inicializamos el aplicativo y analizamos las gráficas:

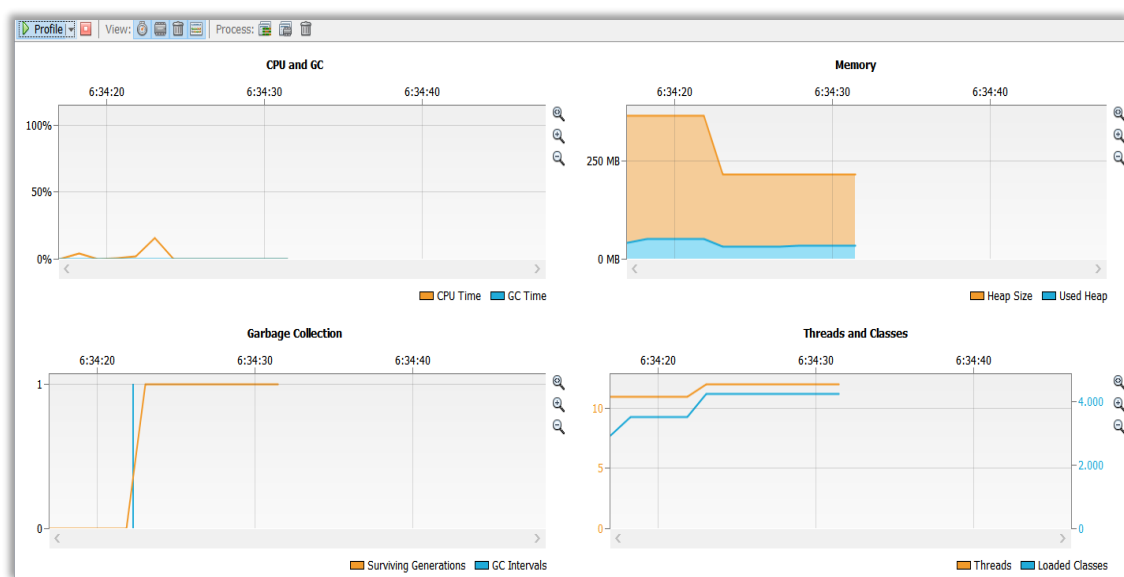


Figura 5-2 Herramienta Telemetry

SISTEMA DE INVENTARIO

Cuando utilizamos el programa el consumo de la CPU no sobrepasa el 50%, el consumo de la memoria no supera los 250MB, la recolección de basura del sistema se mantiene en un índice 2 y subiendo a 3 cuando comenzamos a realizar procesos en el sistema y, por último, en la ventana de hilos y clases se nos indica cuantos hilos se están ejecutando. A la vez, al correr la aplicación vemos que no usa una cantidad excesiva de recursos, si no que pide muy pocos recursos para ejecutarse, lo que indica la optimización conseguida en el aplicativo.



Figura 5-3 Telemetry en Ejecución

METHODS

En esta parte de profile nos indica que métodos se inician, su tiempo de ejecución y el tiempo de CPU que utiliza todo esto en milisegundos.

Name	Total Time	Total Time (CPU)
Thread-7	192.306 ms (100%)	7.295 ms (100%)
WindowsNativeRunLoopThread	189.769 ms (100%)	189.550 ms (100%)
Reference Handler	169.151 ms (100%)	11,4 ms (100%)
mysql-gf-abandoned-connection-cleanup	115.020 ms (100%)	10,8 ms (100%)
QuantumRenderer-0	7.199 ms (100%)	594 ms (100%)
main	759 ms (100%)	180 ms (100%)
JavaFX-Launcher	618 ms (100%)	0,0 ms (-%)
Thread-9	508 ms (100%)	508 ms (100%)
Finalizer	0,0 ms (-%)	0,0 ms (-%)
Prism Font Disposer	0,0 ms (-%)	0,0 ms (-%)

Figura 5-4 Herramienta Methods

SISTEMA DE INVENTARIO

Procedemos a realizar pruebas en el programa, realizamos un registro en el sistema y así conseguimos analizar qué el tiempo de ejecución de cada método y el tiempo de CPU utilizado son rápidos como podemos observar en la imagen.

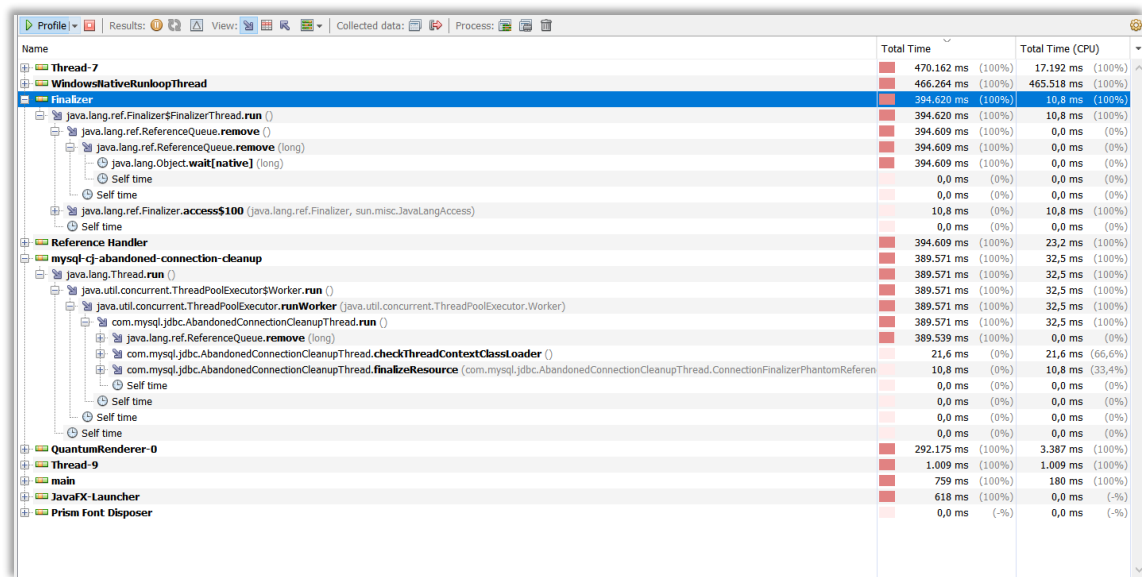


Figura 5-5 Methods en Ejecución

OBJECTS

En este apartado nos indica qué objetos y librerías son utilizados al inicializar el sistema. Además, nos indica los bytes utilizados y la vida útil del objeto. Tal y como se puede apreciar en la imagen, uno de los objetos utilizados es el seleccionado, el conector de MySQL.

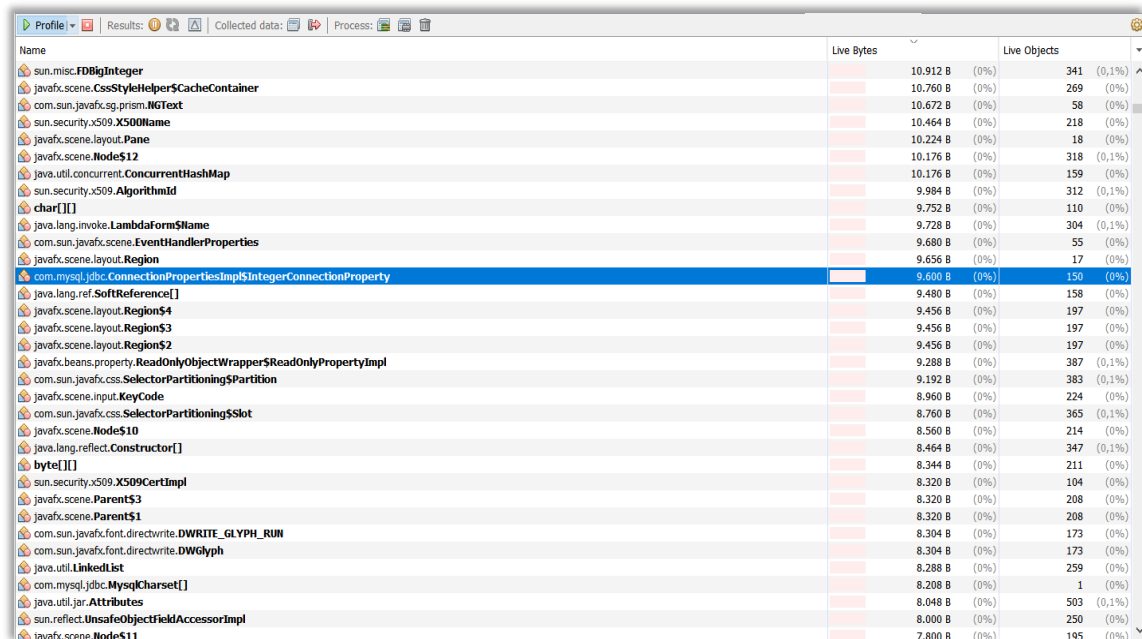


Figura 5-6 Herramienta Objects

SISTEMA DE INVENTARIO

Para probar el análisis de esta herramienta, generamos un reporte. En este instante, nos aparece qué librería estamos utilizando (en este ejemplo JasperReports), los bytes utilizados y la vida útil del objeto.

java.awt.GraphicsCallback\$PaintCallback	16 B	(0%)	1	(0%)
com.sun.javaafx.application.PlatformImpl\$\$Lambda\$32.1915503092	16 B	(0%)	1	(0%)
net.sf.jasperreports.web.util.WebResourceHandlersExtensionRegistryFactory\$1	16 B	(0%)	1	(0%)
javafx.scene.Scene\$1	16 B	(0%)	1	(0%)
javafx.scene.Scene\$3	16 B	(0%)	1	(0%)
javafx.scene.Scene\$2	16 B	(0%)	1	(0%)
com.sun.javaafx.scene.control.skin.MenuBarSkin\$\$Lambda\$239.368642876	16 B	(0%)	1	(0%)
java.lang.Double[]	16 B	(0%)	1	(0%)
net.sf.jasperreports.engine.util.StyleUtil	16 B	(0%)	1	(0%)
javafx.scene.image.WritablePixelFormat\$ByteBgraPre	16 B	(0%)	1	(0%)
javafx.scene.control.TableView\$TableViewArrayListSelectionModel\$\$Lambda\$225.1857451879	16 B	(0%)	1	(0%)
com.sun.javaafx.scene.layout.region.BorderImageWidthConverter	16 B	(0%)	1	(0%)
java.text.DontCareFieldPosition\$1	16 B	(0%)	1	(0%)
com.sun.javaafx.util.TempState\$1	16 B	(0%)	1	(0%)
com.sun.scenario.animation.AnimationPulse\$PulseData\$\$Lambda\$199.8805220	16 B	(0%)	1	(0%)
com.sun.javaafx.application.LauncherImpl\$\$Lambda\$54.460332449	16 B	(0%)	1	(0%)
sun.awt.imane.ImaneWatched\$1 link	16 B	(0%)	1	(0%)

Figura 5-7 Objects en Ejecución

THREADS

En este apartado nos indica el tipo de hilos que se están ejecutando cuando se inicializa o durante la ejecución del sistema. Para saber que está haciendo cada hilo tiene el color verde, que afirma que este hilo se está ejecutando, el amarillo que señala que el hilo está en espera, el morado que indica que el hilo estaba en suspensión, el naranja indica que el hilo se sitúa a la espera de ejecución, y el color rojo nos indica la monitorización del sistema. El tiempo se representa en milisegundos (ms).

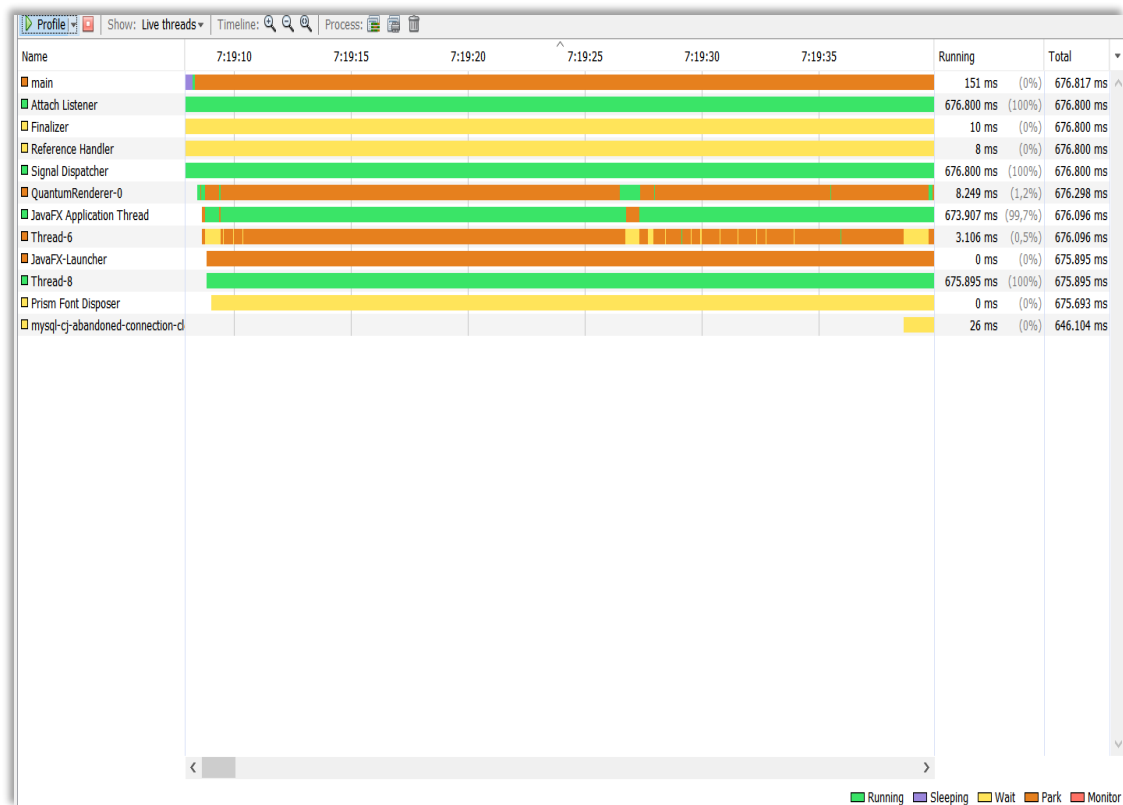
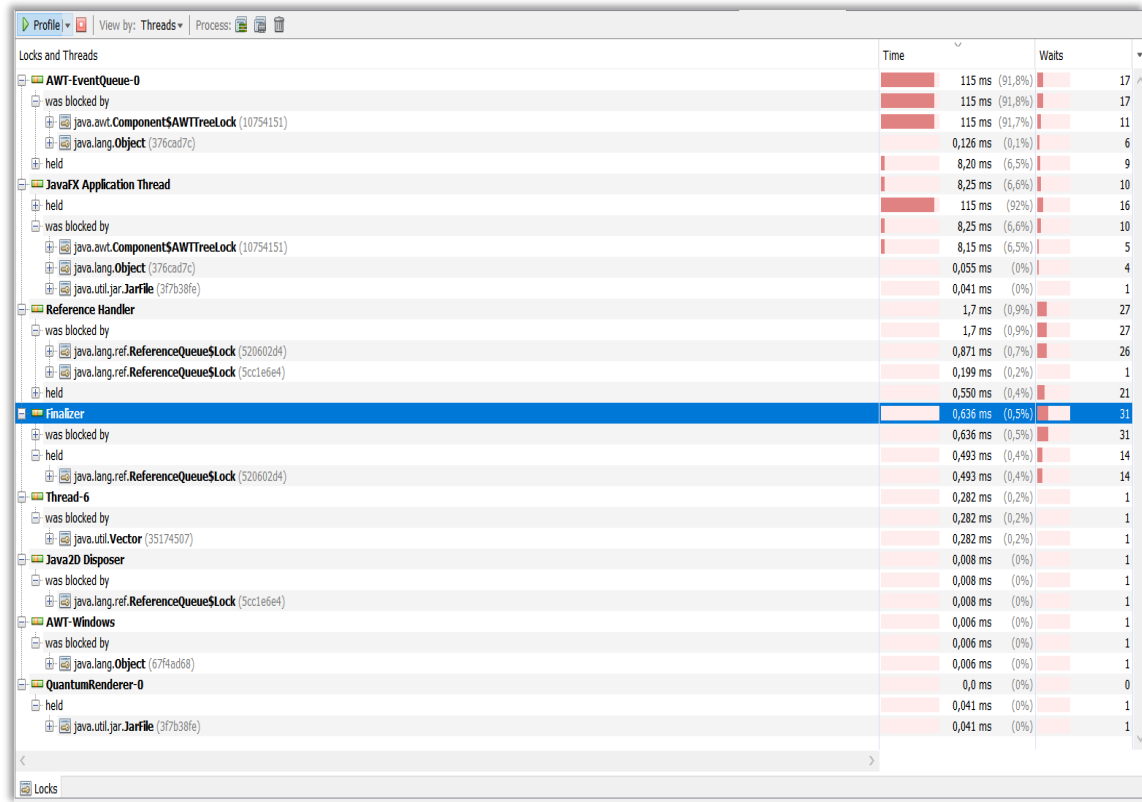


Figura 5-8 Herramienta Threads

SISTEMA DE INVENTARIO

LOCKS AND THREADS

En esta herramienta se nos indica los hilos se han ejecutado y los que han terminado su ejecución. También se nos indica el tiempo que ha transcurrido hasta cerrarse el hilo.

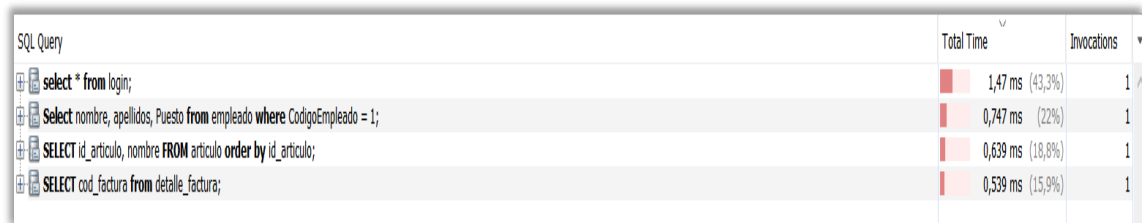


Thread Name	Time	Waits
AWT-EventQueue-0	115 ms (91,8%)	17
was blocked by	115 ms (91,8%)	17
java.awt.Component\$AWTTreeLock (10754151)	115 ms (91,7%)	11
java.lang.Object (376cad7c)	0,126 ms (0,1%)	6
held	8,20 ms (6,5%)	9
JavaFX Application Thread	8,25 ms (6,6%)	10
held	115 ms (92%)	16
was blocked by	8,25 ms (6,6%)	10
java.awt.Component\$AWTTreeLock (10754151)	8,15 ms (6,5%)	5
java.lang.Object (376cad7c)	0,055 ms (0%)	4
java.util.jar.JarFile (3f7b38fe)	0,041 ms (0%)	1
Reference Handler	1,7 ms (0,9%)	27
was blocked by	1,7 ms (0,9%)	27
java.lang.ref.ReferenceQueue\$Lock (520602d4)	0,871 ms (0,7%)	26
java.lang.ref.ReferenceQueue\$Lock (5cc1e6e4)	0,199 ms (0,2%)	1
held	0,550 ms (0,4%)	21
Finalizer	0,636 ms (0,5%)	31
was blocked by	0,636 ms (0,5%)	31
held	0,493 ms (0,4%)	14
java.lang.ref.ReferenceQueue\$Lock (520602d4)	0,493 ms (0,4%)	14
Thread-6	0,282 ms (0,2%)	1
was blocked by	0,282 ms (0,2%)	1
java.util.Vector (35174507)	0,282 ms (0,2%)	1
Java2D Disposer	0,008 ms (0%)	1
was blocked by	0,008 ms (0%)	1
java.lang.ref.ReferenceQueue\$Lock (5cc1e6e4)	0,008 ms (0%)	1
AWT-Windows	0,006 ms (0%)	1
was blocked by	0,006 ms (0%)	1
java.lang.Object (67f4ad68)	0,006 ms (0%)	1
QuantumRenderer-0	0,0 ms (0%)	0
held	0,041 ms (0%)	1
java.util.jar.JarFile (3f7b38fe)	0,041 ms (0%)	1

Figura 5-9 Herramienta Locks And Threads

SQL QUERYs

Aquí podemos ver cuantas sentencias SQL estamos ejecutando cuando iniciamos el sistema, a la par, nos indica tiempo de ejecución de cada una y cuántas veces han sido ejecutadas. Al inicio del sistema tenemos estas querys y su utilización.



SQL Query	Total Time	Invocations
select * from login;	1,47 ms (43,3%)	1
Select nombre, apellidos, Puesto from empleado whereCodigoEmpleado = 1;	0,747 ms (22%)	1
SELECT id_articulo, nombre FROM articulo order by id_articulo;	0,639 ms (18,8%)	1
SELECT cod_factura from detalle_factura;	0,539 ms (15,9%)	1

Figura 5-10 Herramienta SQL Querys

SISTEMA DE INVENTARIO

Procedemos a navegar en el sistema. Cada vez que cargamos una nueva ventana se ejecuta una sentencia SQL, ya que en cada ventana tenemos que obtener datos de la base de datos. Como podemos apreciar en la imagen, se nos muestran el número de sentencias que han sido ejecutadas en el sistema y el número de veces que han sido ejecutadas.

SQL Query	Total Time	Invocations
<code>SELECT id_proveedor, No_documento, cod_tipo_documento, Nombre, Apellido, Nombre_comercial, Telefono, email, Pais, Ciudad, Localidad, direccion, producto.descripcion FROM proveedor INNER JOIN producto ON proveedor.id_proveedor = producto.id_proveedor</code>	1,97 ms (9,7%)	2
<code>select * from login;</code>	1,96 ms (9,6%)	2
<code>SELECT id_articulo, nombre FROM articulo order by id_articulo;</code>	1,72 ms (8,5%)	3
<code>SELECT cod_factura FROM detalle_factura;</code>	1,46 ms (7,2%)	3
<code>SELECT id_Cliente, documento, cod_tipo_documento, Nombre, Apellidos, Telefono, Pais, Ciudad, Localidad, Direccion, CodigoPostal, email, Empleado FROM cliente WHERE Documento LIKE ('%') OR Nombre LIKE ('%')</code>	1,37 ms (6,7%)	2
<code>SELECT sistema.inventario.factura.`Nm_factura`, sistema.inventario.factura.`Fecha_facturacion`, sistema.inventario.factura.`form_pago`, sistema.inventario.factura.`total_factura`, sistema.inventario.factura.`id_cliente` FROM sistema.inventario.factura</code>	1,6 ms (5,2%)	1
<code>SELECT Nm_factura FROM factura WHERE id_factura = (select max(id_factura) from factura)</code>	1,1 ms (5%)	1
<code>SELECT Id_provincia, provincia FROM provincias</code>	0,956 ms (4,7%)	1
<code>SELECT articulo.nombre, sum(detalle_factura.cantidad), ((articulo.precio_venta * (tipo_articulo.IVA / 100)) + articulo.precio_venta) as precio_iva, sum(detalle_factura.total), factura.total_factura FROM articulo INNER JOIN detalle_factura ON articulo.id_articulo = detalle_factura.id_articulo</code>	0,912 ms (4,5%)	1
<code>SELECT Nm_factura FROM factura WHERE Nm_factura = '41492'</code>	0,811 ms (4%)	1
<code>SELECT Nm_factura FROM factura WHERE Nm_factura = '4'</code>	0,793 ms (3,9%)	1
<code>SELECT Nm_factura FROM factura WHERE Nm_factura = '4149'</code>	0,776 ms (3,8%)	1
<code>SELECT nombre, apellidos, Puesto FROM empleado WHERE CodigoEmpleado = 1;</code>	0,747 ms (3,7%)	1
<code>SELECT * FROM tipo_articulo</code>	0,611 ms (3%)	1
<code>SELECT id_tipo_documento, Descripcion FROM tipo_de_documento</code>	0,608 ms (3%)	1
<code>SELECT Nm_factura FROM factura WHERE Nm_factura = '41'</code>	0,604 ms (3%)	1
<code>SELECT nombre, apellidos, Puesto FROM empleado WHERE CodigoEmpleado = 2;</code>	0,540 ms (2,6%)	1
<code>select * from login where id_empleado=1;</code>	0,528 ms (2,6%)	1
<code>SELECT Id_empleado, CodigoEmpleado, cod_tipodocumento, DNI_NIE, Nombre, Apellidos, Email, Puesto, puesto_empleado.Descripcion FROM empleado INNER JOIN puesto_empleado ON empleado.puesto = puesto_empleado.puesto</code>	0,527 ms (2,6%)	1
<code>select producto_id_articulo, producto.descripcion, producto.codigo_barras FROM producto INNER JOIN proveedor ON producto.id_articulo=proveedor.Productos group by producto.id_articulo;</code>	0,520 ms (2,5%)	1
<code>SELECT Nm_factura FROM factura WHERE Nm_factura = '414'</code>	0,512 ms (2,5%)	1
<code>SELECT nombre, apellidos, Puesto FROM empleado WHERE CodigoEmpleado = 0;</code>	0,374 ms (1,8%)	1

Figura 5-11 SOL Ouerys al Inicializar

Si deseamos saber dónde se utiliza cada sentencia, solo pinchamos en una de ella y nos aparece en qué clase y en que método ha sido utilizado o ejecutado otra vez en el sistema. Nos aparecerá una línea perpendicular en el punto en el que ha sido ejecutado.

SQL Query	Total Time	Innovations
<pre> sistemaInventarioControladorVistas.FXMLSistemaInventarioController.registrarVentas () sistemaInventarioControladorVistas.FXMLSistemaInventarioController.permisosSistemas (int) sistemaInventarioControladorVistas.FXMLSistemaInventarioController.recibirIdentificador (int) login.FXMLIngresarController.sistemaInventario (java.lang.event.ActionEvent) login.FXMLIngresarController.ingresarSystem (java.lang.event.ActionEvent) login.FXMLIngresarController.ingresarSistema (java.lang.event.ActionEvent) sun.reflect.NativeMethodAccessorImpl.invoke[native] (java.lang.reflect.Method, Object, Object[]) sun.reflect.NativeMethodAccessorImpl.invoke[native] (java.lang.reflect.Method, Object, Object[]) </pre>	1,00 ms (12,8%) 0,833 ms (6,4%) 0,833 ms (6,4%) 0,833 ms (6,4%) 0,833 ms (6,4%) 0,833 ms (6,4%) 0,833 ms (6,4%) 0,833 ms (6,4%)	2 1 1 1 1 1 1
<pre> SELECT id_provincia, provincia FROM provincias SELECT id_articulo, nombre FROM articulo order by id_articulo; select * from login; SELECT id_proveedor,No_documento, cod_tipo_documento, Nombre,Apellido, Nombre_comercial, Telefono, email, Pais, Ciudad, Localidad, direccion, producto.descripcion from proveedor inner join p SELECT articulo.id_articulo,Nombre, articulo.precio_venta, articulo.precio_costo, articulo.stock, articulo.stock_maximo, tipo_articulo.descripcion_articulo, proveedor.Nomb SELECT id_tipo_documento, Descripcion from tipo_de_documento </pre>	1,63 ms (12,5%) 1,49 ms (11,5%) 1,8 ms (8,3%) 1,6 ms (8,2%) 1,1 ms (7,8%) 0,978 ms (7,5%)	2 2 1 1 1 2
<pre> com.mysql.jdbc.PreparedStatement.executeQuery () com.conexionbaseDatos.ConexionInventario.sSQL (String) metodosjavaClass.LlenarCombos.LlenarComboBox (java.util.collections.ObservableList, java.awt.scene.control.ComboBox, String) proveedorControladorVistas.FXMLRegistrarProveedorController.initialize (java.net.URL, java.util.ResourceBundle) javafx.fxml.FXMLLoader.loadImpl (java.io.InputStream, Class) javafx.fxml.FXMLLoader.loadImpl (Class) javafx.fxml.FXMLLoader.load () metodosjavaClass.VentanaRootPane.mostrarVentana (String, java.awt.scene.layout.AnchorPane) sistemaInventarioControladorVistas.FXMLSistemaInventarioController.registrarProveedor () sun.reflect.NativeMethodAccessorImpl.invoke[native] (java.lang.reflect.Method, Object, Object[]) clienteControladorVistas.FXMLRegistrarClienteController.initialize (java.net.URL, java.util.ResourceBundle) </pre>	0,978 ms (7,5%) 0,978 ms (7,5%) 0,978 ms (7,5%) 0,489 ms (3,7%) 0,489 ms (3,7%) 0,489 ms (3,7%) 0,489 ms (3,7%) 0,489 ms (3,7%) 0,489 ms (3,7%) 0,489 ms (3,7%) 0,489 ms (3,7%)	2 2 2 1 1 1 1 1 1 1
<pre> select producto.id_articulo, producto.descripcion, producto.codigo_barras from producto inner join proveedor on producto.id_articulo=proveedor.Productos group by producto.id_articulo; Select id_Cliente, Documento,cod_tipo_documento, Nombre, Apellidos, Telefono, Pais, Ciudad, Localidad, Direccion, CodigoPostal, email, Empleado from cliente where Documento LIKE ("%") or Nom Select nombre, apellidos, Puesto from empleado where CodigoEmpleado = 1; SELECT id_empleado, CodigoEmpleado, cod_tipodocumento, DNI_NIE , Nombre, Apellidos, Email,Puesto, puesto_empleado.Descripcion FROM empleado INNER JOIN puesto_empleado ON empleado.Pue </pre>	0,862 ms (6,6%) 0,759 ms (5,8%) 0,669 ms (5,1%) 0,652 ms (5%)	1 1 1 1

Figura 5-12 SQL Querys en Ejecución

6 Pruebas de accesibilidad y usabilidad

6.1 Usabilidad

El sistema está enfocado para que cualquier usuario pueda manejar el Sistema de Inventario y Venta. Su navegación no es compleja, el entendimiento de cada ventana es intuitivo para el manejo del sistema. Nos hemos enfocado en estas características para el desarrollo del sistema:

1. Facilidad de aprendizaje: Cualquier usuario podrá utilizar el programa y realizar operaciones básicas.
2. Facilidad y eficiencia de uso: La rapidez con la que se pueden realizar las tareas y aumento de la productividad del usuario.
3. Facilidad de recordar cómo funciona: Hace que el usuario pueda recordar cómo funciona y se utiliza el programa en un futuro.
4. Frecuencia de gravedad de errores: Controlamos los errores del sistema mediante alertas. Se notifica al usuario cada vez que un error sucede.

En esta imagen vemos lo versátil e intuitivo que resulta el programa para el usuario.

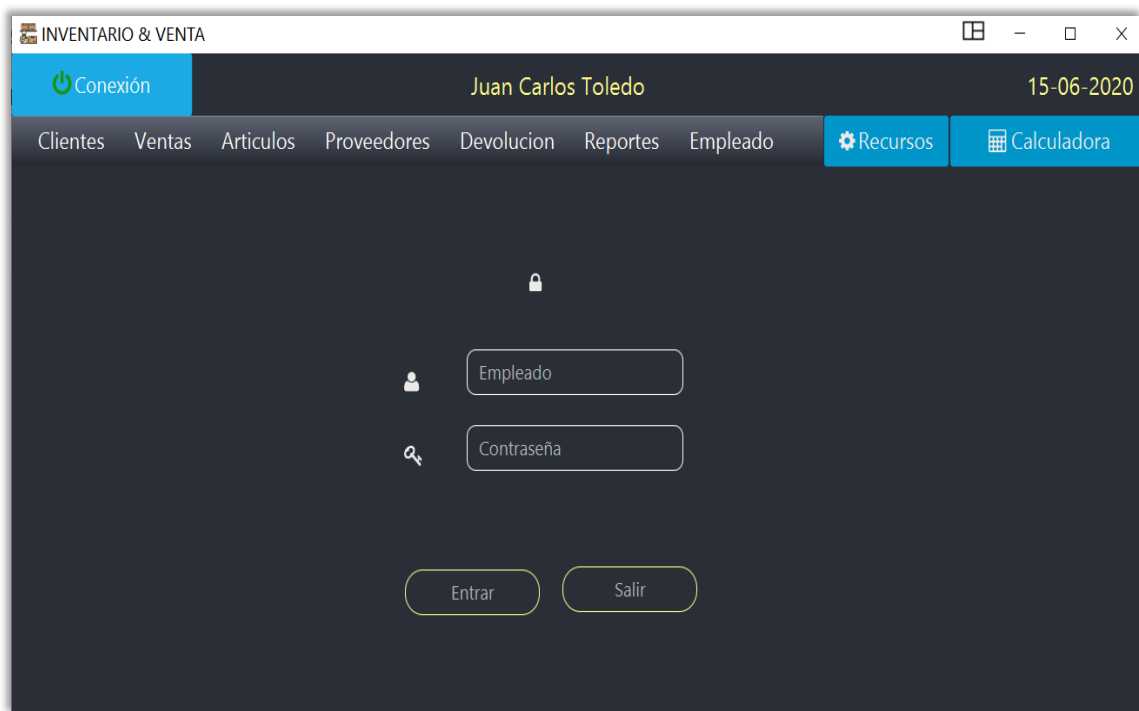


Figura 6-1 Interfaz de Sistema de Inventario y Venta

SISTEMA DE INVENTARIO

6.2 Accesibilidad

Está diseñado con el fin de que todos los usuarios puedan percibir, comprender, navegar e interactuar con el sistema.

La accesibilidad al sistema es sencilla para que cualquier usuario pueda utilizarlo. En el momento de que se realice cualquier operación en el programa, le indicamos con cuadro de diálogo si se ha realizado correctamente o, en caso contrario, le indicamos donde está el error que ha cometido el usuario.

En la imagen nos indica en donde se sitúa y cuál es el error.



Figura 6-2 Interfaz Login con Errores

En la siguiente imagen nos indica que se ha registrado correctamente. Este formato es el que siguen todas las alertas necesarias de la aplicación. Es un formato sencillo, pero muestra al usuario toda la información de error necesaria.

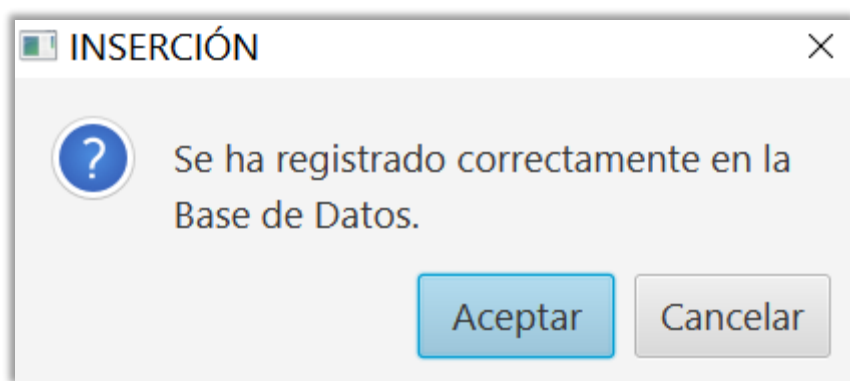


Figura 6-3 Alerta de Inserción

7 Despliegue del proyecto

En muchos casos el proyecto finaliza en la fase de construcción y pruebas, tras la entrega y aceptación del producto por parte del cliente. En otros sin embargo es preciso influir sobre el comportamiento del cliente y de los usuarios del producto para que éstos lo adopten. Esta fase es típica de proyectos internos de cambio en una organización (como, por ejemplo, en un proyecto de rediseño de procesos, reingeniería, implantación de solución informática tipo ERP, etc.) en los que el personal de la organización debe aceptar y aprender a manejar el nuevo producto. Los objetivos fundamentales de esta fase son:

- Conseguir que el producto sea utilizado por los usuarios dándoles el apoyo y la formación que precisen.
- Hay que asegurar que los beneficios alcanzados gracias al proyecto se mantengan una vez el equipo de proyecto se retire y finaliza el proyecto. Por ejemplo, los indicadores de un proceso de negocio rediseñado deben mantenerse en el tiempo.

Por ende, para la distribución del proyecto Sistema de Inventario y Venta consideramos dos posibles vías de despliegue.

7.1 Despliegue Personal

En un primer lugar, el despliegue de la aplicación se ha pensado en distribuir mediante un jar firmado de manera personal por los desarrolladores. Para llevar a cabo esta forma de despliegue, hemos pensado, en un futuro, formar una empresa conjunta por parte de los dos integrantes del proyecto y distribuir el software personalmente a cada uno de los clientes. Esta forma de distribución es muy común en el sector de programadores iniciándose en el sector informático y creemos que puede ser una forma rentable, aunque más limitada, de desplegar el producto.

7.2 Despliegue por Distribuidora

Como segunda opción, contemplamos vender el producto a una distribuidora de softwares que distribuya el aplicativo. Un cliente potencial sobre el que nos hemos fijado son las distribuidoras de TPV's de alquiler, pues constan de softwares propios para incluir en sus terminales y distribuir a empresas pequeñas que no pueden disponer de su propio sistema de inventario y cobro ni de sus propios terminales. Esta opción de despliegue, consideramos que es bastante válida, pues es un producto que, pese a que el cliente principal son los pequeños comercios, el cliente indirecto pueden ser este tipo de empresas distribuidoras, las cuales comparten perfil de cliente con el proyecto.

En cuanto a riesgo económico, esta opción es más rentable pues sería la distribuidora quién cargaría con el coste de la baja demanda del producto, sin embargo, también supondría una pérdida personal por parte de nosotros al vender el proyecto y dejar de ser propio.

8 Problemas encontrados

A la hora de desarrollar la aplicación hemos encontrado varios problemas de estructura y rendimiento. En un principio la estructura de la base de datos era muy simple, pero hemos encontrado problemas en cuanto a modificación y eliminación de objetos almacenados en la base de datos. Principalmente estos problemas se basan en la legalidad que el sistema debe cumplir. Este sistema está desarrollado para administrar información delicada refiriéndose tanto a la información personal de los clientes, empleados y proveedores como a la información logística de la empresa.

Por un lado, el problema de almacenar la información personal de personas ajenas a la empresa implica que para ello los registrados en el sistema tengan que aceptar legalmente el uso de su información por la ley de privacidad de datos. Esto complicaría el uso de la información de los clientes en un caso de uso real, puesto que tendría que reducirse la clientela del comercio de cualquier persona que quiera comprar a clientes específicos que tengan un consumo habitual.

En otro orden de problemas, los problemas que hemos tenido al enfocar esto como una base de datos modificable por el administrador, ha sido que hay datos que una vez registrados no pueden ser modificados o suprimidos. El claro ejemplo del problema lo tuvimos con las facturas, pues lícitamente estas facturas una vez son registradas no pueden modificarse ni eliminarse, por lo que tuvimos que hacer un cambio a última hora de la manera de administrarlas. En un principio, pensamos en un sistema de devolución que realiza una factura distinta para el cliente con los artículos que ha devuelto de la compra y después la factura registrada anteriormente sería modificada, pero al ver que este pensamiento era ilógico y lícito, cambiamos el sistema de registro de devoluciones tal y cómo la ley promulga.

En cuanto a las conexiones a la base de datos nos surgían problemas, pues cada uno de los integrantes del equipo disponíamos de usuarios y contraseñas diferentes, las cuales debíamos de cambiar manualmente a bajo nivel cada vez que deseábamos actualizar el proyecto, además de mostrar un problema posible para los clientes. Para corregir este problema hicimos un archivo de configuración para configurar y testear la conexión a la base de datos antes empezar la aplicación por primera vez que luego queda almacenado y toma los datos por defecto en futuras conexiones. Así mismo solucionamos la configuración del correo profesional del usuario, un archivo de datos con los datos del correo electrónico que se configura en la misma ventana que la conexión y persiste hasta que el usuario desee cambiarla.

Otro gran problema que encontramos fue al tener conexiones abiertas sin la necesidad de estar abiertas, esto nos produjo una sobrecarga de conexiones en varias ocasiones, además de producir el choque en el tráfico de datos enviado a la base de datos, haciendo inserciones repetidas o inserciones imposibles por estar duplicadas. Esto rápidamente nos llevó a realizar una investigación sobre la estructura de conexiones que procesaba el aplicativo y a corregir los errores que nos surgían debido a esto.

9 Conclusiones

Respecto a lo que a nuestra opinión se refiere, Sistema de Inventario y Venta es una aplicación muy potente que se ha desarrollado en poco tiempo lo conlleva dos aspectos que nos conciernen. Tras el paso del tiempo desarrollando el software, se nos han ocurrido implementaciones y mejoras para el aplicativo que desde un principio parecían imposibles lo que ha extendido el proyecto más de lo pensado. Esto ha hecho que Sistema de Inventario y Venta sea completamente tal y como esperábamos pero que queden en mente varias mejoras para implementar en un futuro y que sentimos no presentar actualmente. A pesar de eso, consideramos que la aplicación cumple los requisitos que nos habíamos propuesto cumplir y funciona con bastante velocidad para la cantidad de datos y operaciones que desarrolla.

Otros aspectos que, objetivamente, hemos observado en la aplicación son los siguientes: las interfaces las consideramos a nuestro gusto, pero son mejorables, tanto en estilo como en funcionalidades, la base de datos es estable y segura, pero sí que cambiaríamos el alojamiento por el de un servidor como SQL Server y cifraríamos datos para aumentar su seguridad, los diseños de los documentos están correctos, pero cabe la opción de que en un futuro se mejoren y se personalicen por cliente, y por último, el sistema nos parece bastante completo, de hecho, más completo que otro sistema que hasta ahora hayamos probado, pero sí que consideramos necesario implementarlo más y ampliar funcionalidades.

Como mejoras futuras tenemos en mente: migración de la base de datos a un servidor web, implementación táctil de la aplicación, cambio de diseño de algunas partes del sistema y mejora estética en los reportes.