

TRABAJO FIN DE CICLO



CICLO DE GRADO SUPERIOR EN DESARROLLO DE APLICACIONES MULTIPLATAFORMA

Proyecto de desarrollo de aplicaciones multiplataforma

PackageLog

Autor: Javier Baena Santa-Cruz

Tutor: Miguel Ángel Martín Martínez

Año: 2022

ÍNDICE

1. Introducción, justificación y objetivos	4
2. Resumen	5
3. El Proyecto Software	6
4. Solución	7
4.1. Fase de preanálisis	7
4.2. Fase de análisis	8
4.3. Fase de diseño	9
4.4. Fase de desarrollo	10
4.4.1. Parte 1 Login y registro	11
4.4.2. Parte 2 Entradas, Buscador y Tracking	12-17
4.5. Fase Posdesarrollo	18
4.6. Fase Pruebas	18
4.7. Fase de documentación	19
5. Simulación de prueba	19
6. Conclusiones y mejoras	20
7. Bibliografía	21
8. Anexo	22

ÍNDICE DE FIGURAS

Figura 4.1. - Logo PackageLog	7
Figura 4.2. - Diagrama de casos de usos	10
Figura 4.3. - Login y Registro	11
Figura 4.4. - Entradas	12
Figura 4.5. - Tracking	13
Figura 4.6. - Buscador y cambio de estado	14
Figura 4.7. - Diagrama flujo de datos	15
Figura 4.8. - Vehículos	16
Figura 4.9. - Alta Vehículos	16
Figura 4.10. - Gestión de carga	16
Figura 4.11. - Clasificación	17

1. Introducción, justificación y objetivos

El proyecto viene motivado por el aumento de empresas de transporte y el incremento masivo de paquetería dada la pandemia actual que vivimos. Esto ha supuesto una renovación en las empresas para adaptarse a la nueva situación, con escenarios como el de los confinamientos, restricciones y demás limitaciones personales. Muchas empresas apostaron por dar un fuerte empujón a sus negocios online, lo que hizo que algunas de estas, tales como MRW, ASM o Correos Express tuvieran exceso de trabajo. Estas se han visto desbordadas con un número de envíos que apenas pueden asumir, a raíz de esto se han ido creando muchas pymes dedicadas al transporte. Aquí es donde entra en juego nuestro proyecto, creando una aplicación para una empresa pequeña que haga los repartos a nivel local.

1.1. Objetivos

El objetivo de este proyecto es el de realizar una aplicación cuyas funcionalidades serán:

- Control de usuario para acceder a la aplicación.
- Creación de usuarios.
- Módulo de configuración de BBDD.
- Módulo para la creación de envíos.
- Módulo para la búsqueda y modificación de envíos.
- Módulo para ver la trazabilidad de los envíos.
- Módulo para la gestión de vehículos y control de paquetería.
- Módulo para la clasificación de los paquetes en vehículos.

2. Resumen

El trabajo intenta recoger los funcionamientos internos (intranet) y externos (de cara al cliente) de una empresa de paquetería/logística, para ello se ha realizado una aplicación de escritorio desarrollada en java.

Dentro de la aplicación se tienen tres partes bien diferenciadas, aunque de forma interna todas están relacionadas.

La primera parte, que es la más genérica, se encarga del control de usuarios. Esta parte no tiene nada de la lógica característica propiamente dicha de la empresa, simplemente permite registrarse como usuario y si ya se tiene un usuario registrado, permite loguearse y acceder a la aplicación.

Una vez dentro de la aplicación hay 5 pestañas de trabajo, que agrupan a las otras dos partes que se han mencionado anteriormente.

Las pestañas de entradas y tracking: estas ventanas se encargan de la parte de la empresa a la que pueden acceder los clientes como tal. Sus funciones son limitadas, pues un cliente no puede modificar parámetros internos de la empresa. En esta parte se recogen las funcionalidades de dar de alta un envío con los datos que se quieran para, después, con el número de seguimiento ir a la pestaña de tracking y ver los diferentes estados por los que ha pasado el envío.

Las pestañas Buscador, vehículos y clasificación: en esta parte es donde se recoge el funcionamiento a nivel interno de la empresa.

La pestaña de Buscador, simula la PDA de los operarios. En ella se puede introducir un número de seguimiento de un envío para encontrar los datos de este (si los hubiera). Además, se pueden cambiar estados a los envíos con los diferentes estados por los que va a pasar el paquete.

La pestaña vehículos permite modificar todo lo que está relacionado con la flota de vehículos de reparto de la empresa. Desde la aplicación, se puede dar de alta una furgoneta, editar su ruta de reparto, darla de baja y gestionar los envíos que hay en su interior.

Finalmente se encuentra la parte de clasificación que es la encargada de repartir los paquetes según el CP de destino en los diferentes vehículos.

De forma resumida queda explicada cada parte de la aplicación. Hay que tener en cuenta que existe gran cantidad de lógica interna, motivo por el cual todas las partes están interconectadas entre sí, de este modo todo encaja perfectamente y es lo más fiel posible a la lógica de negocio.

3. El proyecto Software

Una vez estudiadas las necesidades empresariales y la lógica de negocio se decidió que lo más rentable era desarrollar una aplicación en java.

Se ha utilizado el entorno de desarrollo eclipse y el lenguaje de programación Java para la implementación de la parte lógica del proyecto.

Para la parte visual se ha hecho uso de la librería JavaSwing que permite utilizar ventanas, cuadros de dialogo y otros componentes que se necesitaban de una manera más amigable.

Para conectar con la base de datos se valoraron diferentes tecnologías, JPA JDBC, MyBattis, pero al final se decidió utilizar JDBC. MyBattis no se había estudiado en ningún modulo y aunque era una solución idónea se descartó. JPA permitía una interfaz más amigable que JDBC, pero como hay muchos procesos interconectados y era posible que hubiera que rehacer partes de la base de datos según las necesidades de desarrollo se decidió finalmente JDBC, que permite hacer cambios y rehacer el código de manera mucho más simple.

La base de datos es una base de datos relacional con lenguaje SQL en MySql. Se ha usado PHPmyAdmin como interfaz gráfica para desarrollarla.

Además, se han usado otros programas adyacentes que han permitido optimizar el desarrollo, como por ejemplo Excel, donde se han implementado diferentes hojas para guardar datos de base de datos y así poder adjuntar un resumen de los envíos que se han creado y no tener que memorizar números de seguimiento ni matriculas. También las pruebas de aceptación de la fase Alpha del proyecto se guardaron en Excel para que fuera más fácil ver los errores.

4. Solución

4.1. Fase de preanálisis

La primera fase del proyecto se basó en hacer un estudio de las posibles necesidades que tuviera el mercado para poder buscar un desarrollo que no solo fuera interesante a nivel de desarrollar usando los conocimientos aprendidos, sino que también representara una solución real a un problema actual.

Se plantearon varias opciones, pero después de recabar datos hubo una que destacó. Los datos de correos express, la empresa privada de paquetería más grande de España, mostraban lo siguiente.

El récord de paquetes diarios que había manejado la empresa había sido durante un blackfriday con unos 380.000 envíos. Durante la pandemia, con todo el país confinado en su casa, la gente pedía por internet muchos más productos que de costumbre y los datos de correos express mostraban que diariamente se estaban moviendo entre 500.000 y 600.000 paquetes.

Esto supone prácticamente triplicar la carga de un día normal antes de la pandemia y casi duplicar la del mejor día de su historia. En este punto se sabía que, aunque los sistemas informáticos podían aguantar la carga de trabajo, para los repartidores estaba siendo un gran sobreesfuerzo.

Es así como nace la empresa a la que se le ha dado el nombre de PackageLog, empezando a trabajar en un proyecto que se pueda presentar.

Logo Corporativo:



Figura 4.1. - Logo PackageLog

4.2. Fase de análisis

La idea de PackageLog estaba clara, ahora solo había que estudiar bien la manera más optima y eficiente de llevar a cabo el proyecto. Se hizo una fase de análisis en la se estudiaron los requisitos y necesidades que podría tener la empresa, y se vio que en función de la cantidad de paquetes que se fuese a recibir se podían tener dos tipos de alcance.

Alcance para reparto de pocos paquetes. Se vio que otras empresas del sector en sus almacenes y delegaciones eran los mismos operarios los que recibían un camión con todos los paquetes y luego con una PDA leían los códigos de barras para saber dónde iban y ellos mismos los metían en furgonetas para el reparto.

Este alcance es útil si la carga de paquetes es pequeña, sin embargo, se vio que en cuanto el volumen de paquetería subía, el coste humano y de tiempo para mantener este sistema es inasumible.

Por otra parte, también estaba el alcance suponiendo que la empresa fuera a repartir una gran carga de paquetes. En este caso la subida de paquetes en furgonetas se hace en una maquina clasificadora o Sorter, donde se recogen todos los paquetes y los coloca en diferentes rampas en las que están las furgonetas en función de donde vaya cada paquete.

Finalmente, después de un buen análisis se decidió lo siguiente: las dos opciones de alcance no eran excluyentes, sino que una podía ser una evolución de la otra. Es decir, mientras que la empresa vaya creciendo se tendrá implementado el primer alcance y se irá desarrollando el segundo. Cuando el segundo alcance esté desarrollado ya se podrá trabajar con un gran volumen de mercancía si fuese necesario. Bajo esta premisa se comienza la fase de diseño.

4.3. Fase de diseño

Una vez se tenía clara la idea a nivel de lógica de negocio había que ver cuál era la mejor manera de implementarla. Se pasó el proyecto al servicio informático y desde ahí se planteó dos posibles soluciones. Una aplicación web o una aplicación de escritorio, con sus ventajas y desventajas cada una.

Las ventajas que planteaba la aplicación web era su escalabilidad. Se planteaba con un servidor central lo que permitía hacer llamadas desde cualquier máquina de la empresa, permitiendo ampliar la empresa a nivel nacional si se desease en algún momento. Por otra parte, si se caía la red, la empresa quedaría totalmente paralizada, un riesgo totalmente innecesario para una empresa tan pequeña.

En cuanto a la aplicación de escritorio se planteó con una base de datos en un servidor local en un almacén. Esto permite no depender de la red externa solo de la intranet. Luego bastaría con instalar el programa con un ejecutable en cada ordenador del almacén y configurar la conexión de base de datos.

Para ayudar con el desarrollo de las ventanas se usó la librería de javaSwing, muy útil para este tipo de aplicaciones.

En cuanto a la base de datos se decidió usar el motor mysql Workbench, que permite una interfaz amigable para el uso de sentencias sql.

Se usó git para llevar un control de versiones.

Y por último se pensó dividir el proyecto en tres partes bien diferenciadas. Las dos primeras completas serían el primer alcance del que se ha hablado en el apartado anterior y la tercera parte, una vez acabada, haría que completáramos el segundo alcance. Estas partes se dividen de la siguiente manera:

1. Control de acceso y registro.
2. Utilidades del cliente e intranet básica.
3. Intranet avanzada.

Desarrollo 1 y 2 completan alcance sin clasificador.

Desarrollo 1, 2 y 3 completan el alcance total.

Se entrará en más detalle de estas partes en la fase de desarrollo, que viene a continuación.

4.4. Fase de desarrollo

Después de la fase de análisis los objetivos a alcanzar en el desarrollo están claros. Se harán las tres fases de forma secuencial para ir completando los objetivos de alcance en orden, aunque desde el primer momento se va a desarrollar con el objetivo de completar el alcance número dos.

De casos de uso se ha realizado un pequeño diagrama para que sirva de guía y de este modo el empleado sepa qué hacer en función de lo que necesite. Se adjunta foto del esquema borrador.

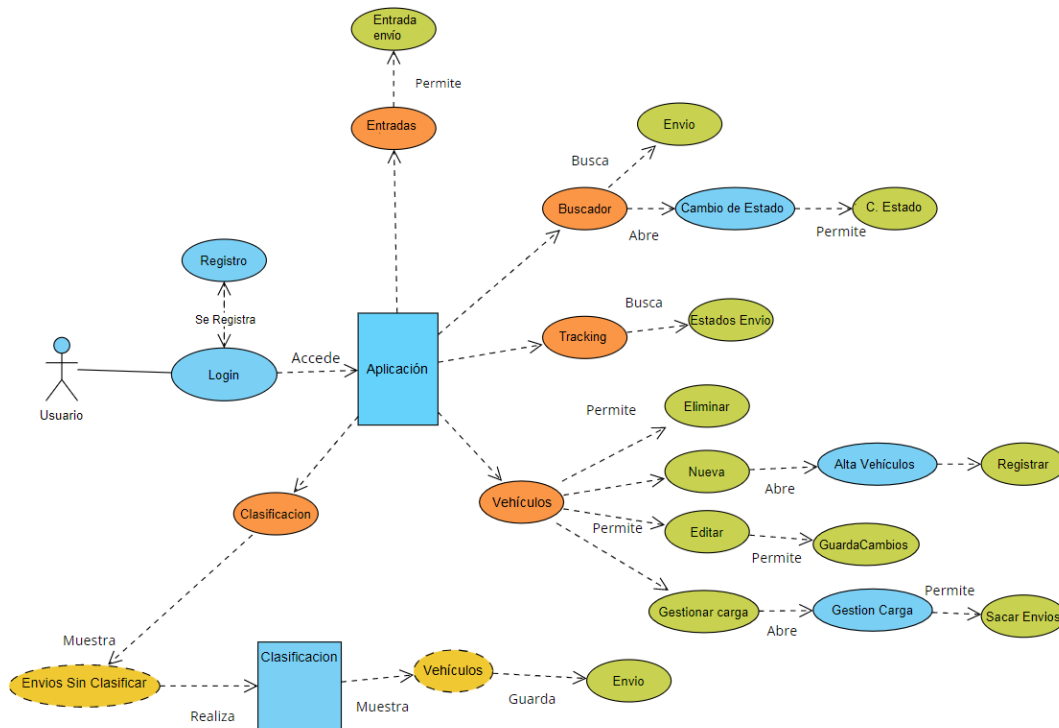


Figura 4.2. – Diagrama de casos de usos

Leyenda:

Azul: pantallas o ventanas modales de la aplicación Naranja: Pestañas de la ventana principal (aplicación) Amarillo: combobox con datos

Verde: botones y procesos que van a realizar

Nota: se va a pasar a explicar las 3 partes del desarrollo, no se va a entrar en excesivo detalle en su funcionamiento o cómo usarlo pues en el anexo se adjunta una guía básica de uso.

4.4.1. Parte 1 Login y registro

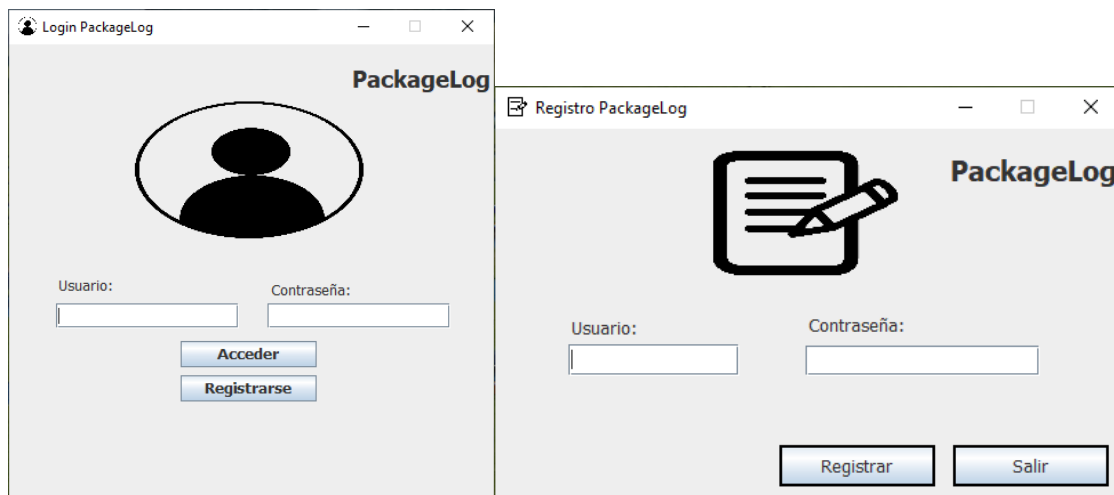


Figura 4.3. – Login y Registro

El primer paso del desarrollo ha sido implementar una base de datos, a pesar de tener un buen análisis de las necesidades de la base de datos, no se sabía con exactitud cuáles iban a ser los datos exactos que iba a necesitar cada tabla. Por este motivo la base de datos se fue incrementando en función de las necesidades del desarrollo. En esta primera parte se creó una tabla de usuarios para poder guardar cada persona que se registrase en la aplicación.

Después se inició el proyecto en java descargando la librería de java swing y lo primero que se desarrolló fue la clase para la conexión con la base de datos y poder empezar a trabajar. La aplicación lleva intrínseca esta conexión a la base de datos con estos parámetros que habrá que configurar.

```
url="jdbc:mysql://localhost:3306/packageLog?serverTimezone=UTC"; usuario="root";  
contraseña="";
```

Se empezó a desarrollar la ventana de Login, que era bastante sencilla, junto con la de registro, que tienen un comportamiento similar. La ventana de Login se diseñó pequeña e intuitiva, tiene dos espacios para introducir usuario y contraseña; al hacer click en el botón aceptar hace una búsqueda en la base de datos: si el usuario y la contraseña son correctos abre la ventana principal de la aplicación, si no son correcto da un mensaje de error.

Al dar al botón registrar se abre la ventana de registro, muy similar a la de Login, que permite registrarte con usuario y contraseña. Si se han pasado todas las validaciones necesarias se cierra y vuelve a abrir la ventana de Login con un mensaje de que todo ha ido bien.

A nivel interno la parte visual es independiente de la lógica, solo cogen y muestran datos. La lógica intermedia está introducida en dos clases llamadas serviceLogin y servicePrincipal. Son las que hacen todas las comprobaciones intermedias, hacen cálculos y llaman a la capa de datos DAO para poder operar correctamente.

Aunque las pruebas exhaustivas no se realizaron hasta tener el programa completo a cada parte implementada se le realizaron pruebas básicas de funcionamiento, conexión correcta a base de datos, guardado y muestra de datos,... Evidentemente, el funcionamiento tenía que ser el correcto.

4.4.2. Parte 2 Entradas, Buscador y Tracking.

Una vez se ha completado la parte de Login para poder entrar en la aplicación se empezó a desarrollar la lógica interna de la aplicación. En primer lugar, se implementaron nuevas tablas en la base de datos (envíos, estados) donde se guardan los datos de los paquetes que se dan de entrada.

La primera pestaña implementada fue la de Entradas. Esta pestaña tiene varias cajas de texto donde se indica que es necesario introducir los datos del envío. También están indicados los campos obligatorios. Se desarrolló un control tanto de los campos que se pueden pasar como de las excepciones por si hubiera fallo.

El proceso de grabación de envío es el siguiente: el programa tiene una lógica para generar un numero de envío único, una vez lo genera guarda los datos en la base de datos y si todo ha ido bien, le asigna un primer estado (Entrada en el sistema); en la pantalla nos devuelve excepciones por si fallara algo. Algunos ejemplos de estos controles son no meter letras en un campo numérico o que los campos estén dentro de ciertos rangos.

Una vez pasadas todas las validaciones, desde esta pestaña se puede grabar un envío. el número de envío por si fuese necesario trabajar con él.

PackageLog

Entradas Buscador Tracking Vehículos Clasificación

Introduce los datos para dar entrada a un envío

Localidad Origen:

CP Origen:*

Peso:*

Volumen:*

Destinatario:*

Direccion:*

Localidad Destino:*

CP Destino:*

Observaciones:

Cliente: Javier

Dar entrada

Figura 4.4. - Entradas

Una vez está desarrollada la parte de entradas se pasa a la pestaña de tracking. Aquí es donde los clientes pueden hacer el seguimiento virtual de sus envíos. Para ello se diseñó una sencilla pestaña donde se introduce el número de envío de alguno de los paquetes, se hace click en a buscar y en un cuadro de texto te sale una lista ordenada de todos los estados por los que ha pasado el envío. Por ejemplo, un envío recién creado solo tendrá el primer estado, generado automáticamente al darle entrada.

La lógica interna es sencilla. Con el número de seguimiento se llama al service, que accede a los datos, y saca todos los estados asociados a ese envío ordenados por fecha decreciente.



Figura 4.5. - Tracking

Por último, se desarrolló el programa del buscador de los operarios. Esta permite leer el número de envío de los paquetes. Para saber sus datos, basta con meter el número de envío en el buscador y, si está en el sistema accederá a los datos de este y los mostrará por pantalla. Con este buscador los operarios aparte de leer los bultos pueden ir asignando estados según vayan operando con ellos. Basta con dar click al botón de cambiar estado una vez haya salido el bulto. Se implementó una ventana modal en la que permite elegir una serie de estados en los envíos y asignarlos como si se fuese un repartidor sacando un pedido a reparto o entregándolo en una casa. Ahora mismo solo hay una restricción en la ventana modal y es que si un pedido ha sido marcado como entregado no se le pueden asignar más estados.

Para ver que todos los estados se han asignado bien, basta con ir a la ventana de tracking volver a buscar el envío y ver que todo sale correctamente.

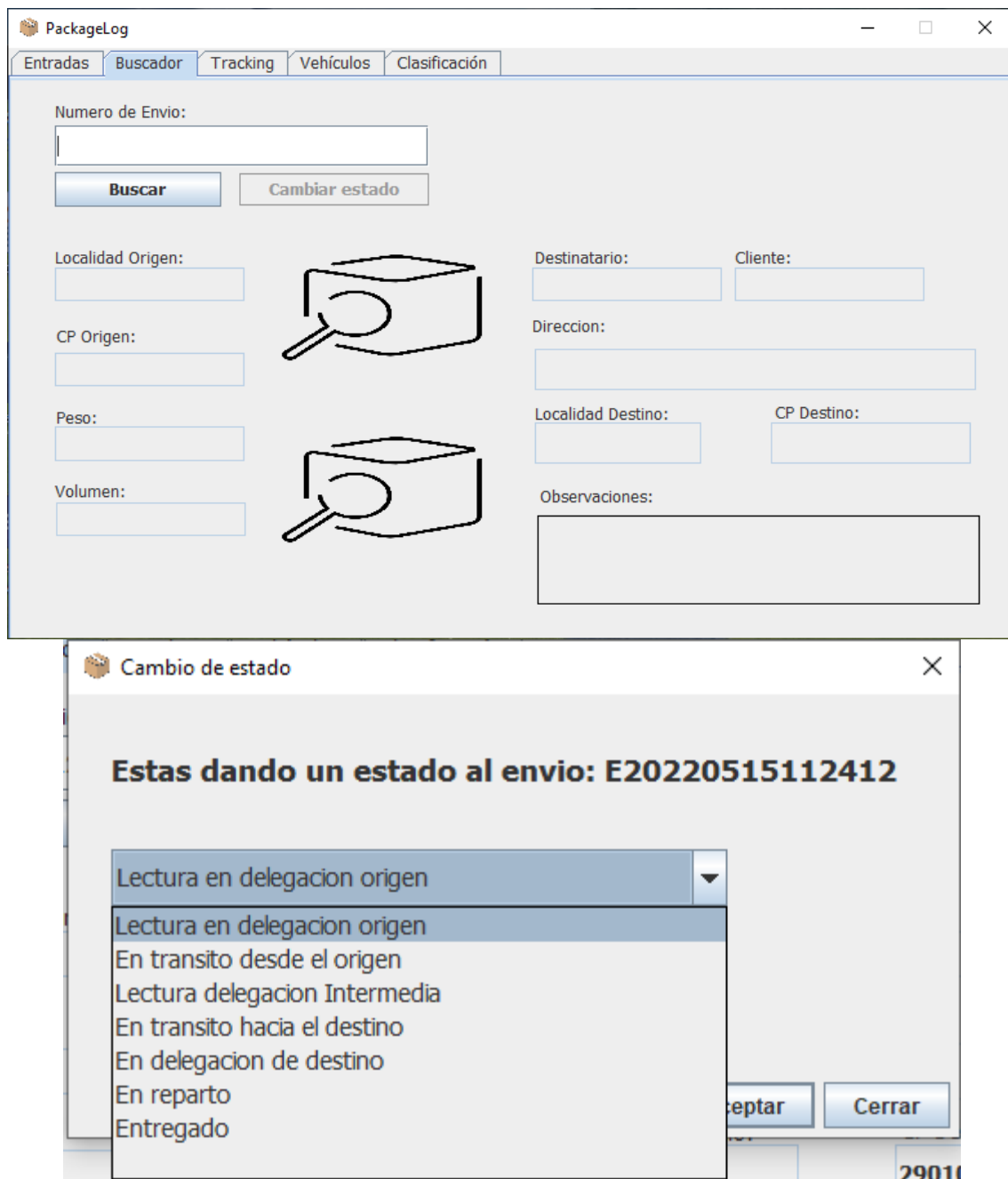


Figura 4.6. – Buscador y cambio de estado

Parte 3 ventana Principal con pestañas vehículos y clasificación

Al acabar la fase dos se hicieron unas pruebas básicas de funcionamiento, finalizando con éxito. Fue entonces cuando se empezó a implementar la fase 3. El desarrollo independiente de esta fase no era especialmente complicado, pero lo difícil era coordinar todo el programa en su conjunto. En primer lugar, se realizó un pequeño diagrama que representaría el flujo de datos hacia la base de datos e internamente en la aplicación.

Se adjunta el diagrama:

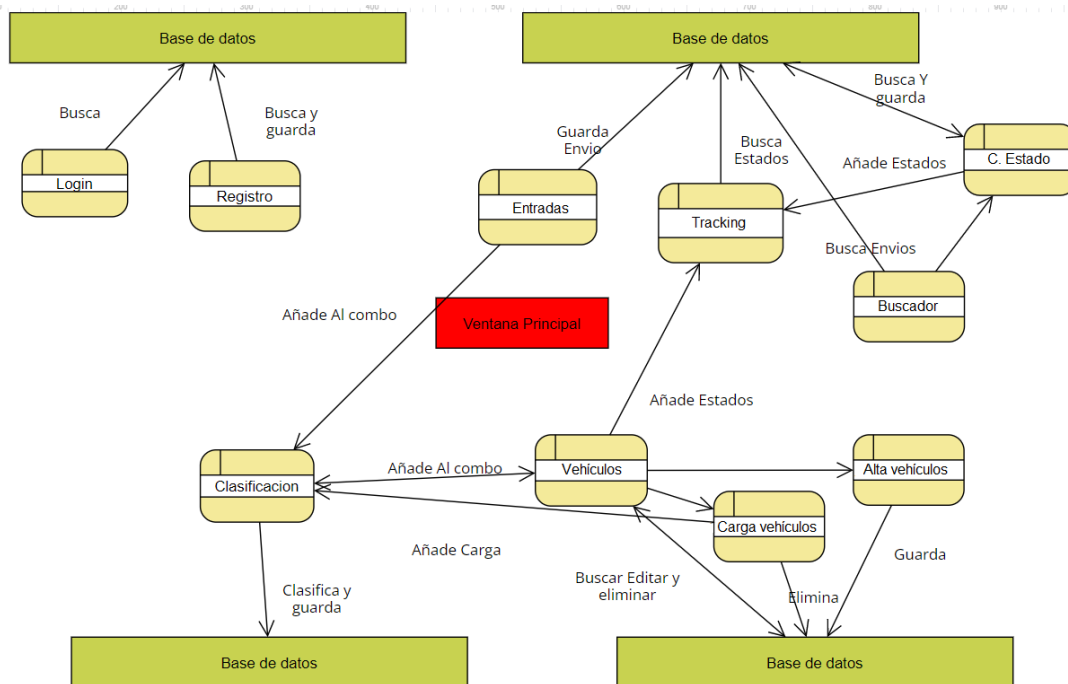


Figura 4.7. – Diagrama flujo de datos

Una vez se tenía más o menos claro el flujo de los datos y como tenían que estar interconectadas las partes del programa (siguiendo siempre una lógica de negocio), se añadieron dos tablas más a la base de datos, cargavehículos y vehículos, desde donde se controla la lógica de clasificación para el reparto de los paquetes.

Se implementó la pestaña de vehículos, que tiene una gran cantidad de funcionalidades. Lo primero que permite es buscar una matrícula, por si quieres acceder a un vehículo ya creado. Si encuentras un vehículo se mostrarán todos sus datos, los bultos que tiene guardados dentro, y también permite editar su ruta de reparto. Además, se implementó una ventana modal para manejar la carga de los vehículos. Si el usuario desea bajar un bulto de un vehículo, basta con ir a la parte de gestionar la carga y seleccionar el bulto desde la ventana modal para sacarlo.

También se desarrolló la opción de crear un. Nuevo vehículo a través de otra ventana modal en la que se introduce la matricula (no podía existir en el sistema), la ruta de reparto y el programa la da de alta en el servicio.

Por otra parte, se implementó un botón de eliminar vehículo por si hubiera que dar de baja alguna, sin que esto afecte al funcionamiento de la empresa.

PackageLog

Entradas Buscador Tracking **Vehículos** Clasificación

Matricula

Buscar Editar Nueva

Ruta Desde:

Ruta Hasta:

Fecha Alta:

Fecha Baja:

Carga de envios:

Guardar Cambios Gestionar Carga Eliminar Vehículo

Figura 4.8. - Vehículos

Alta Vehículos



Matricula



Ruta Desde

Ruta Hasta

Registrar Cancelar

Figura 4.9. – Alta Vehículos

Gestion de carga

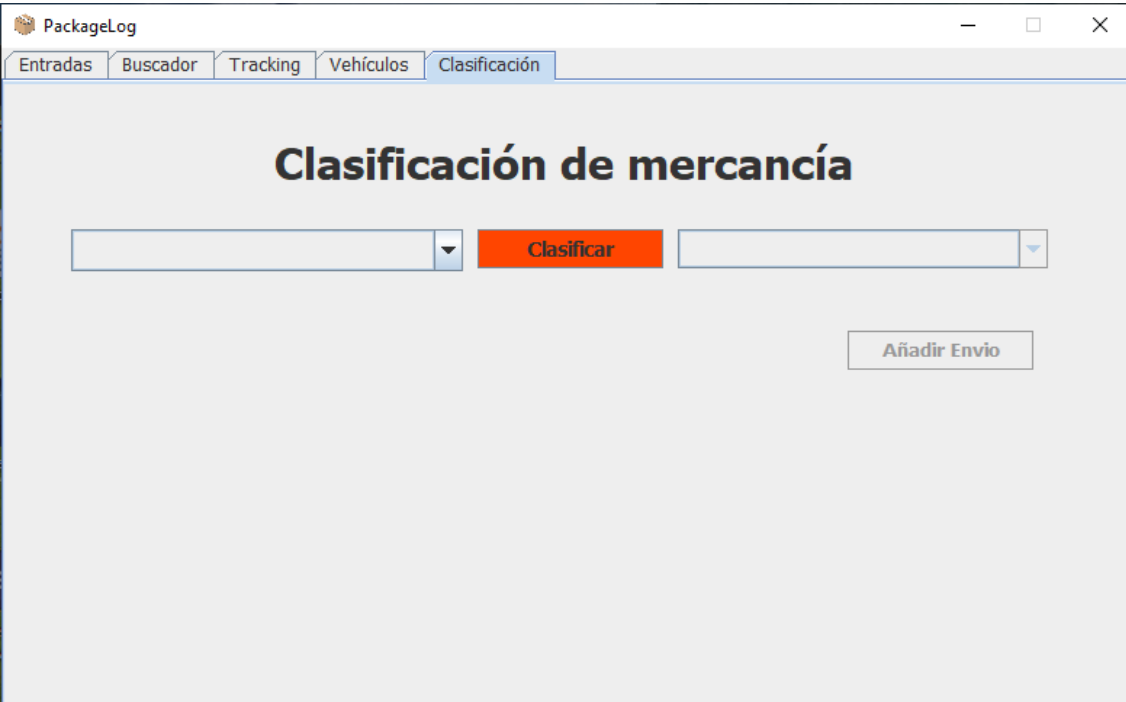
Sacar Salir

Figura 4.10. – Gestión de carga

Una vez desarrollada la parte de los vehículos, solo falta el proceso de clasificación que simula la maquina sorter. Esta máquina se ha simulado en la pestaña de clasificación, es sencilla, pero tiene mucha lógica interna. Al acceder a ella esta precargado un combo en tiempo real con los envíos que se encuentran listos para ser clasificados, es decir, envíos que no hayan sido entregados ni estén subidos en otro vehículo. Se selecciona el número de envío que se quiera clasificar y se tiene que activar la maquina clasificadora; una vez la máquina acabe el trabajo, dirá qué vehículos tienen la ruta de reparto del envío seleccionado, escogemos una de los vehículos y se hace click en añadir envío para así guardarlo dentro.

La lógica interna de esta parte se pensó de la siguiente manera: el combo con los envíos se recarga automáticamente cada vez que se realiza una acción que afecta a los envíos, como por ejemplo dar de alta un envío, dar un estado nuevo o meterlo en un vehículo.

Después, el proceso de clasificación coge el envío que se ha seleccionado y usa el código postal de destino para compararlo con las rutas de reparto de los vehículos, es decir, el código postal de destino tiene que estar en el rango ruta desde ruta hasta del vehículo. Una vez hace la comparativa, carga el segundo combo con los vehículos que cumplen este criterio que no estén dadas de baja. Y finalmente mete el envío en el vehículo.



The screenshot shows a web application window titled 'PackageLog'. It has a navigation bar with tabs: 'Entradas', 'Buscador', 'Tracking', 'Vehículos', and 'Clasificación'. The 'Clasificación' tab is active. The main content area is titled 'Clasificación de mercancía'. Below the title, there is a form with two dropdown menus, a red 'Clasificar' button, and a 'Añadir Envío' button.

Figura 4.11. - Clasificación

4.5. Fase Posdesarrollo

En este punto se tenía la lógica funcional pero no se habían conectado todas las piezas entre sí, sino que todas funcionaban bien de manera individual. Por este motivo, se usó una fase entera de desarrollo en representar fielmente la lógica de negocio interconectando las partes y poniendo las restricciones de uso necesarias para evitar errores.

Las restricciones están bien detalladas en el anexo de guía básica de uso. Es necesario hacer hincapié en las interconexiones que se detallan a continuación, pues no fueron fáciles de implementar y son las que permiten al programa actuar como una empresa real de logística.

Ventana Entradas: al dar de alta un envío se le asigna automáticamente un primer estado y se recarga el combo de envíos disponibles para la máquina de clasificación. Ventana tracking: esta ventana solo muestra información, por lo que no genera ninguna conexión directa con otras ventanas solo las recibe.

Ventana Buscador y modal cambio de estado: no se puede dar como entregado un envío que está en una furgoneta. Si se entrega un envío no se puede volver a darle estados y automáticamente será sacado del combo de clasificación. Solo se puede dar incidencia si encuentra el envío.

Ventanas vehículos, altas vehículos y gestión de carga: solo se puede acceder a los botones de control si se encuentra un vehículo. Un vehículo no se puede eliminar si tiene envíos dentro. Un nuevo vehículo no puede tener la misma matrícula que otra. La ruta tiene que ser del código postal más pequeño al más grande. Si se saca un envío de un vehículo, se le asigna un estado de bajado y vuelve al combo de clasificación.

Ventana Clasificación: El combo siempre se recarga en directo después de cualquier acción que le pueda afectar. No se van a cargar en los combos ni envíos entregados ni envíos que estén en otras furgonetas ni furgonetas que estén dadas de baja. Con todas las ventanas funcionando correctamente se procedió a la fase de pruebas.

4.6. Fase Pruebas

Una vez terminado el desarrollo se decidió seguir con las pruebas y correcciones. Primero se realizaron unas pruebas en fase Alpha y se vieron los posibles errores y mal funcionamiento. Una vez corregidos se hizo la misma batería de pruebas en fase beta para dejar el proyecto cerrado, a la espera de las pruebas en los usuarios. Estas baterías de pruebas se resumieron de modo intuitivo en un Excel llamado "pruebas de aceptación", que se adjunta en el anexo.

En esta primera batería se encontraron algunos errores y disfuncionalidades como pantallas en las que no se podía volver atrás, comboboxs mal coordinados con los datos reales o errores en la visualización de elementos. Se detallan todos por colores en el Excel adjunto.

Posteriormente se realizó una fase de pruebas en usuarios tanto nativos digitales como no para ver posibles mejoras en la usabilidad.

Al final de esta fase se exportó el proyecto y se guardó en un ejecutable de Windows para poder instalarlo en los ordenadores.

4.7. Fase de documentación.

Durante el desarrollo se elaboraron los diagramas, el Excel y la guía básica de usuario.

También se extrajeron las sentencias sql necesarias para crear la base de datos y también, si el cliente quisiera no empezar todo de cero y ahorrar tiempo, se adjunta otro script donde se pueden inicializar algunas de las tablas.

Una vez acabado el desarrollo se empezó a elaborar esta memoria de trabajo como un resumen de todo el trabajo elaborado.

5. Simulación de prueba

Aunque se hará una presentación del proyecto explicando su uso se detallará en unas pocas líneas el ciclo de vida básico de un bulto a través de la aplicación.

Se inicia en la pantalla de Login, en la que se tiene hacer un registro. Una vez hecho el registro se podrá entrar en la pantalla principal.

Aquí se accede a la ventana de entradas, donde introduciendo los datos correctamente, se podrá dar de alta un envío, obteniendo así su número de seguimiento.

Como operarios, en la ventana de Buscador se puede buscar este envío para ver sus datos y darle estados, como en reparto, en tránsito etc.

En la ventana de seguimiento se puede meter el número de envío para ver todos los estados por los que ha pasado.

En la ventana de clasificación se puede introducir el envío en los vehículos.

En la ventana de vehículos se hacen todas las operaciones crud con la flota de vehículos, así como gestionar los envíos que tenga dentro.

El ciclo de vida termina cuando el envío adquiere el estado entregado.

6. Conclusiones y mejoras

El trabajo representa fielmente lo que podría ser una empresa de transportes. Se han conseguido reproducir a pequeña escala las funcionalidades principales de la lógica de negocio que tienen las empresas de paquetería. Además, el programa es sólido y fluido.

El nivel de conexión interna de los elementos y las diferentes restricciones hacen que la experiencia sea muy cercana a manejar cualquier aplicación de intranet de este tipo de empresas.

En cuanto a las mejoras, sobre lo que hay implementado apenas se puede mejorar nada pues el funcionamiento es sólido, por lo que todas las mejoras que se implementen serán nuevos desarrollos añadidos y pequeños evolutivos que se hagan sobre el programa.

Quizás una posible mejora es la asignación de roles a los usuarios, de tal manera que en función de si tenías el rol de cliente o el de operario se activasen unas u otras pestañas tras el login.

También se podría implementar el tipo de vehículo que podamos dar de alta, se entiende que solo son furgonetas, pero se podrían agregar motos y trailers y agregar una nueva columna en la base de datos.

Como mejora importante se podría desarrollar una aplicación móvil desde la que pudiéramos acceder a la pestaña buscador y se pudieran hacer los cambios de estados automáticamente "picando" la referencia que estuviese como QR o código de barras

7. Bibliografía

Manual MySQL WorkBench: <https://dev.mysql.com/doc/workbench/en/>

Pequeña guía de java Swing: <https://guru99.es/java-swing-gui/>

Base de datos Correos Express

Ejemplos de Jdialog: <https://jnjsite.com/java-swing-3-jugando-con-un-jframe-un-jdialog-y-un-jinternalpane/>

8. Anexo

En la entrega del proyecto se adjuntan los siguientes elementos repartidos en varias carpetas:

- Documento del trabajo (documento actual).
- Carpeta Code donde está el código fuente de la aplicación.
- Jar listo para ejecutar.
- Exe portable para Windows.
- Carpeta BBDD donde se encuentra el script: "Creación_BBDD" que se usa para crear la base de datos.
- Excel de pruebas de aceptación que se realizaron en fase pruebas (Docs).
- Guía de uso básica donde se entra en detalle más en el funcionamiento (Docs).