

Tarea de Entrega Tema 2

Acceso a Datos



Elaboración de prototipo de la app de gestión de Easy Depot
(Proyecto DAM2°)

Por: Veronica Gonzalez Bravo

Índice

1 Introducción.....	3
2 Objetivos.....	4
3 Clases de datos.....	5
4 Excepciones.....	6
5 Explicación final.....	7

1 Introducción

En este apartado haremos una descripción breve de los objetivos de esta entrega.

El programa realizado durante esta entrega trata sobre un pequeño prototipo de la parte lógica del proyecto EasyDepot , un proyecto de Consignariado automatizado, o de cabinas automatizadas donde los clientes podrán reservar desde su app de móvil una cabina, recibirán un código para abrir el locker y posteriormente guardaran sus enseres dentro.

Este proyecto tendrá dos versiones distintas, la versión de móvil o cliente, que será la que se usara para reservar una cabina y la versión de escritorio o administrados, con la que podremos observar cada uno de los locales, su rendimiento y si es necesario abrir una de las cabinas para que el usuario en caso de algún problema pueda extraer sus objetos.

Citando el [documento del proyecto](#) , sección 3.1 página 21, en la parte referida al perfiles de administrador que va a ser el que use la app de escritorio, esta debe seguir las siguientes condiciones:

- a. **Visionado de rendimiento:** A diferencia del perfil anterior en este podremos observar el rendimiento de todo el franquiciado, independiente del empresario que lo maneje, puede ser global o específico, es decir puede observarse el rendimiento de toda la franquicia o de una instalación en concreto.
- b. **Listado de entidades:** Podrá observar el listado de entidades, desde clientes , reservas, consignas incluso cabinas en cada consigna , desde aqui podrá seleccionar una entidad concreta y observar cada perfil, si por ejemplo selecciona una consigna podra ver el rendimiento de esa consigna, si selecciona una reserva la información de esa reserva y si selecciona a un cliente las estadísticas de ese cliente.

La búsqueda de estas entidades serán realizada mediante los elementos de los mismos, como correos, nombre, números de teléfono, fechas, etc. Es importante que esta función de búsqueda esté bien trabajada.

- c. **Buzón de tickets:** A través de un servicio de tickets que tendrá el soporte el administrador podrá acceder a la información del cliente solicitante para

realizar una llamada, tras confirmar la identidad del usuario este podrá apuntar la incidencia y proceder con la resolución.

- d. **Apertura de cabinas:** El soporte podrá abrir y cerrar cabinas accediendo desde el ticket de soporte con el código de reserva o en su defecto yendo de manera manual a la consigna donde se encuentran los enseres del cliente.

Sin embargo y con la finalidad de cumplir los objetivos de esta entrega solo se hará la parte de visionado de rendimiento y listado de entidades(Apartado CRUD).

2 Objetivos

Para esta entrega deberemos cumplir los siguientes objetivos:

- Listar los clientes, las reservas y las consignas
 - Los clientes se listaran solo enseñando sus datos en una línea
 - Las reservas se listaran en un historial dentro del local donde se hayan realizado
 - Las consignas se listarán dentro del sistema de manera independiente, cada una con su lista de reservas principal.
- Modificar clientes, y reservas:
 - En referente a las consignas solo se podrá cambiar la dirección.
 - Se podrán modificar los datos del cliente como nombre apellidos, teléfono y email(si este no existe previamente)
 - Los datos de las reservas podrán modificarse en referente a la fecha de finalización (Estará implementado como una opción en el menu llamado cerrarReserva) y si se ha efectuado la retirada de los objetos correctamente. Se podra añadir una fecha de devolución de objetos , también tendrá una descripción de ticket en caso de que haya una incidencia y quedará en el historial.
- Visionado del rendimiento:
 - El sistema mostrará el rendimiento de las consignas y del franquiciado en general. Si seleccionamos un cliente también podremos ver las reservas que ha realizado.

- Por parte de una consigna las reservas que ha tenido y sus ingresos
- Por parte de un cliente las reservas que ha realizado y sus puntos de tienda
- Crear entidades, podremos crear clientes, consignas y reservas nuevas.
- Persistencia: Se efectuará carga y guardado de datos en dos archivos distintos con la finalidad de poder hacer un entorno de pruebas cómodo, tendrá un archivo llamado ficheros.txt donde se guardaran y cargarán siempre los datos.

3 Clases de datos

- Cliente, tendrá la siguiente configuración:
 - String nombre
 - String apellidos
 - String telefono
 - String email <Elemento clave>
 - String contraseña
 - Int numeroReservas
 - int puntosTienda
- Consigna, tendrá la siguiente configuración:
 - int id
 - int numeroReservas
 - List reservas
 - int ingresos
 - String direccion
- Reserva, tendrá la siguiente configuración:
 - String emailCliente
 - String id reserva (idConsigna-idpersonal)
 - String fechaInicio
 - String fechaExtracción
 - Boolean ReservaFinalizada
 - Boolean Incidencia
 - String descripciónIncidencia

4 Excepciones

Las excepciones a contemplar tendrán cuatro tipos importantes.

- **Críticas (Junit)**: se producirán durante la apertura y cierre de los archivos de guardado, si el sistema no encuentra los archivos o estos están dañados darán una excepción crítica que cerrará el sistema previo mensaje al usuario y no le dejarán interactuar con el menú principal.
- **Errores de carga de datos (Junit)**: Son errores dentro de lectura de datos aquellos que si no se proporciona la línea correctamente formateada para su lectura en csv no se imprimirán en pantalla pero se escribirán en un log singleton donde podrá verse la fecha del error, la línea donde se produjo el error y la razón del error. Cuando se produzca el error el cargador de datos ignorará la línea y no la meterá en el sistema, solo escribirá el error en el log (llamado log.txt).
- **Datos suministrados por el usuario**: Son aquellos en los que el usuario escribirá los datos para crear nuevos objetos o para buscarlos, en caso de que el usuario por ejemplo escriba el teléfono mal, escriba una letra en vez de un entero son ejemplos de errores de este tipo.
- **Ejemplos de errores**:
 - Añadir una reserva a una consigna que no existe
 - Añadir un elemento que ya existe
 - Que le falte o sobre una categoría a una línea del csv
 - Que no esté bien escrita o corrompida una línea del csv
 - Que no se encuentre un email
 - Escribir mal el teléfono

5 Explicación final

Para hacer las pruebas en el código deberá escribir el nombre del archivo “ficheros.csv” o “pruebaErrores.csv” para ver la escritura en el log, si quiere ver que pasa si no carga nada simplemente ponga lo que quiera.