Gestión de datos

1er Cuatrimestre

**FRBA COMMERCE**

**ESTRATEGIA**

|  |  |
| --- | --- |
| Nombre y Apellido | Legajo |
| Joel Melamed | 146.804-2 |
| Ana Estevez |  |
| Victoria Cabrera |  |



# INDICE

ESTRATEGIA

Para comenzar, el proyecto fue subdivido en 4 proyectos:

- Clases:

En este proyecto se encuentran todas las clases del sistema. Cada clase se asemeja con cada tabla de la base. Cabe destacar que el manejo de estas clases fue realizado a conveniencia, donde pueden haber diferencias entre la clase y su entidad en la BD, como por ejemplo, en campos que son claves foráneas (relaciones con otras tablas), en la clase, directamente, el atributo pasa a ser la entidad relacionada en sí.

- Conexion:

En este proyecto se encuentra una clase llamada SQLHelper, el cual se encarga de realizar todas las acciones que tengan que ver con la BD, parseando los parámetros, y dando a quien programe la aplicación una interfaz más amigable, evitando sentencias poco declarativas de la libreria encargada de esta acción. Es, como bien lo dice su nombre, un helper.

- Excepciones:

Este proyecto contiene todos los tipos de excepciones que entiende y maneja la aplicación, ellas son:

- La entidad buscada no se encuentra.

- No hay datos encontrados.

- ErrorConsulta: Excepción para cualquier tipo de stored procedure o instrucción SQL que falle.

- Entidad existente.

-Bad insert: excepción para fallo de inserts.

- FRBA Commerce:

En este proyecto se encuentra la UI del sistema. Contiene todos los ABM's, historiales, estadísticas, Inicio de sesión, y demás funcionalidades del sistema que el usuario final podrá manejar.

-Utilidades:

En este proyecto creamos algunas clases que nos son de gran utilidad en el proyecto:

- Encryptor: se encarga de encriptar texto en el algoritmo SHA256

- Validator: Es la clase encargada de cualquier tipo de validacion de campos ingresados por el usuario: nulidad del campo, si es numérico, si es decimal, etc.

- Manejador de combos.

- Manejador de dialogos: Crea un formulario nuevo con un solo campo de texto a completar. Sirve, por ejemplo, para el cambio de clave.

## **Modelo de Aplicación**

La aplicación se divide en tres capas: Conexión, clases, interfaz del usuario. El usuario accederá visiblemente a la interfaz. Esta misma, se comunica con las clases, las cuales se encargarán, a su vez, de comunicarse con la capa de conexión, para acceder a la tablas de datos de la base y obtener/modificar/insertar datos.

Consideraciones y decisiones generales

Creación de tablas y migración de datos:

* Definimos DNI como usuario para los clientes, y CUIT para las empresas
* Definimos la clave admin (encriptada bajo el algoritmo SHA256) como default para todos los usuarios (salvo el administrador)
* Los IDs de las tablas son autonuméricos
* Definimos que las publicaciones van a migrarse todas como FINALIZADAS en su estado ya que su fecha de vencimiento ya ocurrió.
* Definimos que, como no hay un ABM de funcionalidades en la aplicación, las decidimos nosotros y las aplicamos mediante un enum en la clase Funcionalidades

Sobre la aplicación: