

# PROYECTO 1º TRIMESTRE

## PORTFOLIO CRIPTOMONEDAS



Miguel García Holgado

2ºASIR

Asignatura: GBD

**INDICE**

1. Introducción al Proyecto.....3

2. Clases y esquemas.....4

3. Herencias.....6

## 1. Introducción al Proyecto

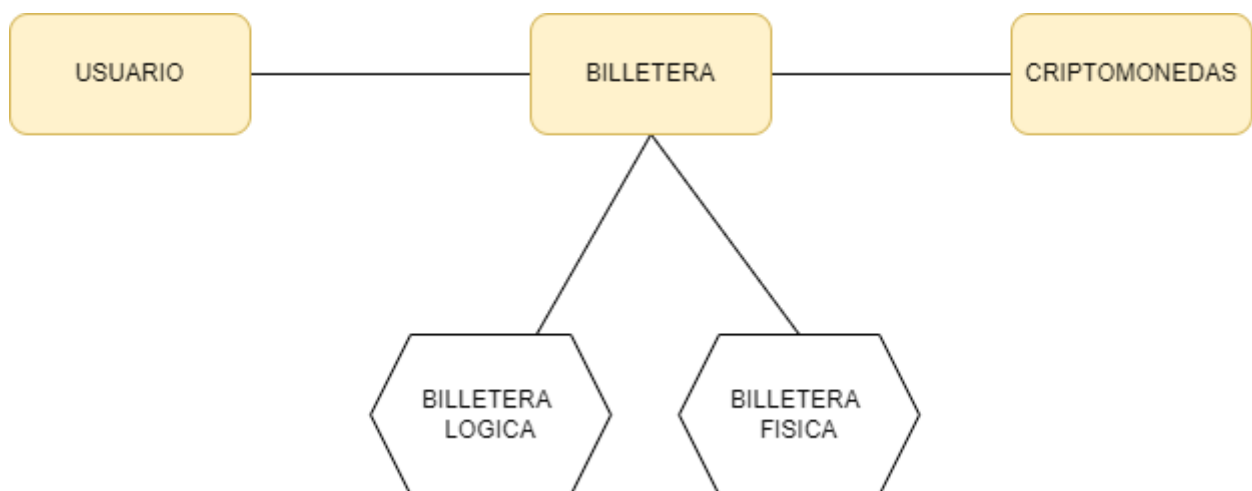
Este proyecto trata sobre una cartera de criptomonedas en la que podemos usarla mediante un previo inicio de sesión o registro en la API.

Para el uso de la aplicación primero tendremos un menú que nos pregunta si queremos logearnos o registrarnos, seguido de eso tendremos el siguiente menú que te pregunta sobre tus portfolios activos, te pregunta cosas como crear uno, buscarlo, eliminarlo etc..

Después en ese mismo (Menu para los portfolios) tenemos una opción que nos dice “Acceder a un portfolio”, esto sirve para poder entrar al portfolio elegido y añadirle, eliminarle ver tus criptomonedas etc..

Una cosa a destacar es que cada usuario tiene un apartado para el en la base de datos, cada usuario tienes sus propios portfolios y en cada portfolio tiene sus propias criptomonedas.

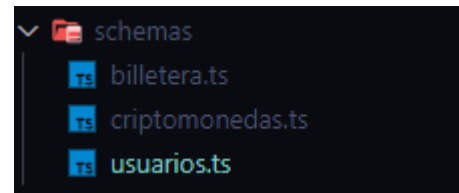
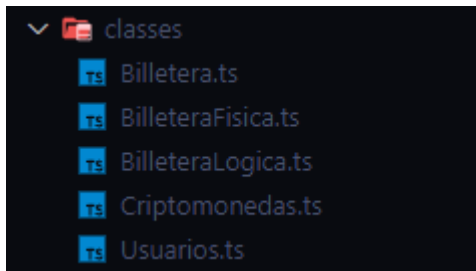
La estructura principal es la siguiente:



Mis herencias son billetera lógica y billetera física, la diferencia entre una y la otra es que la lógica no tiene numero de intentos y la física si fallas tu contraseña 10 veces se elimina todo lo que esta dentro.

## 2. Clases y esquemas

En mi proyecto he usado las siguientes clases y los siguientes schemas:



MI CLASE USUARIOS:

```
protected idUser: number;  
protected Nombre: string;  
protected Password: string;  
protected DNI: string;  
protected Telefono: number;  
protected Billeteras: Array<Billetera>;
```

## MI CLASE BILLETERA

```
protected idWallet: number;  
protected NombreWallet: string;  
protected TipoWallet: string;  
protected Password: string;  
protected InversionInic: number;  
protected Monedas: Array<Monedas>;
```

## MI CLASE CRIPTOMONEDAS

```
protected _id: number;  
protected Nombre: string;  
protected Siglas: string;  
protected qty: number;  
protected Token: number;  
protected Capitalizacion: number;
```

### 3. Herencias

**Definición de herencia:** significa que se pueden crear nuevas clases partiendo de clases existentes, que tendrá todas los atributos y los métodos de su 'superclase' o 'clase padre' y además se le podrán añadir otros atributos y métodos propios.

Mis herencias en este proyecto son las siguientes:

- **Billetera Física:** esta billetera se caracteriza por tener una contraseña la cual si falla un numero de veces borra todo el contenido de su interior.
- **Billetera Lógica:** esta billetera tiene un PIN auto generado para que el usuario entre a su portfolio.