

## Actividad 8.4

### API REST

En esta actividad deberán agregar a la ya existente *API REST* vista en clase, soporte a transacciones del tipo extracción y depósito. Se debe agregar la capacidad de crear transacciones de ambos tipos (extracción y depósito), leer todas las transacciones dada una cuenta existente, leer una transacción en particular y, por último, poder eliminar una transacción.

Para realizar esto, se deberán agregar los *endpoints* pertinentes en la capa de *controller*, la lógica de negocio necesaria en la capa de *service*, y por último, la nueva entidad del dominio junto con su capa de *data access* (repositorio/DAO).

Las reglas de negocio son:

- No permitir depósitos de montos nulos o negativos
- No permitir extracciones mayores al saldo de la cuenta

### BDD

Se ha agregado una nueva *feature* en el archivo gherkin *bank\_account\_operations.feature*.

```
Bank account promo, get 10% extra in your $2000+ deposits, up  
to $500
```

Esta *feature* posee los siguientes escenarios:

1. *Successfully promo applied, cap not reached*
2. *Successfully promo applied, cap reached*
3. *Promo not applied*

Se deberá desarrollar en código las pruebas necesarias para pasar estos escenarios, así como también actualizar la lógica de negocio (capa de *service*) para cumplir con estas nuevas reglas y pasar las pruebas satisfactoriamente.

El trabajo se realizará a partir del código visto en clase ([repositorio](#)), el cual deberá ser *forkeado* (*fork* de git) en un repositorio personal del estudiante. Una vez terminado el trabajo de codificación, deberán grabar y adjuntar a la entrega un video no mayor a 2 minutos, en donde se demuestre y explique el funcionamiento de la API mediante la utilización de estos nuevos *endpoints* (con la ayuda de *swagger*, *postman* o *curl*), el código fuente generado y la ejecución de los test de BDD.