

# Trabajo práctico 2: Granelco

Sistemas de Comunicaciones

Instituto La Salle Florida

Octubre de 2021

## 1. Objetivo

El presente trabajo práctico tiene el objetivo de poner en práctica los conceptos de programación orientada a objetos vistos en clase. Para ello, se propone una situación problemática muy similar a un caso de aplicación real, como motivación, el cual se debe resolver bajo un marco acotado pero amplio de conceptos troncales para la materia.

## 2. Motivación

Se desea diseñar una solución que modele el funcionamiento integral de un sistema de cajero automático **Granelco**.

Este sistema debe ser capaz de gobernar toda la lógica de funcionamiento, dejando de forma abstracta el aspecto físico de la máquina en manos de los alumnos de electromecánica.

Es decir, se propone programar una solución en Python que modele el funcionamiento del cajero, dejando la instancia de interfaz con la mecánica para otra instancia.

## 3. Desarrollo

El cajero automático Granelco debe ser capaz de administrar depósitos y extracciones de billetes de las cuentas de sus usuarios.

Los usuarios del sistema pueden ser personas físicas o personas jurídicas. En el caso de las personas físicas, se registran datos como Nombre, Apellido, DNI, Fecha de nacimiento, Profesión, Género, entre otros, mientras que en las personas jurídicas los datos a registrar suelen ser la Razón Social, CUIT, Fecha de inicio de actividades, Rubro, entre otros.

A su vez, existen las tarjetas de Granelco, las cuales están asociadas a un usuario y sus principales datos son: Número de tarjeta (16 dígitos numéricos), Fecha de Vencimiento y Código de Seguridad.

También, cada usuario tiene un PIN de 4 dígitos numéricos para poder acceder al Granelco.

Cada usuario tiene asociada una cuenta en ARS y otra en USD

Las cuentas tienen un saldo disponible, y movimientos de ingreso y egreso de dinero.

Cada movimiento que realiza el usuario consta de 3 datos elementales: fecha, monto (positivo para ingreso y negativo para egreso) y descripción.

Cuando el usuario se logea en un sistema Granelco, debe primero insertar su tarjeta, colocar su PIN y puede operar, es decir, puede depositar o extraer dinero en formato de billetes.

Los billetes admitidos por el Granelco son:

- En ARS: 10, 20, 50, 100, 200, 500 y 1000.
- En USD: 1,2,5,10,20,50 y 100.

El sistema debe contemplar un stock de billetes que debe ser definido mediante un archivo de configuración inicial y al ir operando, el sistema debe volcar en ese archivo como quedo el stock de billetes después de cada operación.

Todas las mañanas, los camiones de Prosecur llenan las máquinas Granelco con estas cantidades de billetes con 10 billetes de cada tipo.

### **3.1. Interfaz de usuario**

El sistema Granelco ofrece una interfaz de usuario sencilla con un menu por consola donde en primera instancia se invita al usuario a loggearse, pudiendo hacerlo de 2 formas: ingresando DNI/CUIT + PIN o ingresando la tarjeta y su PIN. Cómo estamos desarrollando una aplicación que simula una máquina física, para poder emular el ingreso de tarjeta, el

sistema deberá pedir que el usuario ingrese los 16 dígitos de su tarjeta.

Si el usuario ingresa de forma incorrecta su PIN 3 veces, el sistema Granelco deberá bloquear al usuario y el sistema debe arrojar un mensaje al usuario notificando esto e invitando a que se diriga a la sucursal a desbloquear su clave.

A su vez, una vez en el sistema, el usuario podrá consultar su saldo, sus últimos 5 movimientos, podrá depositar, extraer dinero, cambiar el PIN y desloguearse.

Toda la información del usuario y su cuenta debe quedar registrada en un archivo a modo de servidor en caso de futuros ingresos al sistema.

Si el usuario desea extraer una cifra que puede entregarse por el Granelco por falta de billetes, deberá arrojarse un mensaje indicando que no es posible completar la operación por falta de billetes.

### 3.2. Bases de datos

Se prevee que sistema Granelco, en su versión productiva, este conectado a un motor de base de datos PostgreSQL, pero en la instancia de desarrollo, se manejará la persistencia de los datos con archivos de texto plano.

Dado esto, se recomienda como parte de la solución, se contemple construir una clase Base De Datos, que tenga métodos simples y sirvan de interfaz entre los objetos que forman parte de esta aplicación y los archivos donde se guarda la información.

Esta clase debe permitir por ejemplo buscar en la base de archivos a una persona dado su DNI o CUIT, así como también escribir sobre estos mismos archivos la modificación de sus estados.

Se recomienda guardar en los archivos todos los atributos de la clase y escribir un parser que leyendo el archivo, permita construir en memoria el objeto en su totalidad.

Recordar que los usuarios tienen asociada una cuenta corriente, la cual también debe ser guardada en un archivo de texto con los movimientos realizados.

Queda a cargo de cada alumno investigar como hacer esto en Python.

Finalmente, el sistema Granelco debería permitir el ingreso al sistema en modo administrador (proporcionando un nombre de usuario y contraseña especiales que defina el programador) y que en ese modo administrador, se permita crear un usuario nuevo, des-

bloquear a usuarios con PIN bloqueados y que el mismo quede guardado en la base de datos, así como también simular que se hace el arqueo de caja y se resetea el stock de billetes.

Para probar el funcionamiento integral de la aplicación, se pide crear 5 usuarios con datos al azar que considere de interes, así como también sus respectivas tarjetas y demás.

### **3.3. Entregables**

Se debe armar y entregar no solo el código con la solución a este problema con sus archivos para funcionar, sino casos de uso con capturas de pantalla y el diagrama UML correspondiente.

Se deben entregar capturas de pantallas de todas las situaciones posibles de uso. A modo de ejemplo, como mínimo se pide:

- Persona física que retira dinero.
- Persona física que ingresa dinero.
- Idem anterior con personas jurídicas.
- Bloqueo de usuario por colocar 3 veces incorrecto su PIN.
- Modificación de PIN de un usuario.
- Visualización de últimos 5 movimientos de un usuario.
- Creación de un usuario.
- Arqueo de caja.
- Otro caso de uso de interés.