



Universidad Nacional de Rosario

Experimento Empírico

Requerimiento funcional

Autor: Stizza, Federico

Director: Cristiá, Maximiliano

Departamento de Ciencias de la Computación
Facultad de Ciencias Exactas, Ingeniería y Agrimensura
Av. Pellegrini 250, Rosario, Santa Fe, Argentina
7 de Agosto de 2023

1. Modalidad y plazo de entrega

En caso de utilizar *Replit* como entorno de desarrollo la solución deberá ser entregada por ese medio.

Caso contrario, por favor comprimir la solución en un archivo con su nombre, por ejemplo **Stizza-Federico.zip** y envíelo a mi dirección de correo electrónico: *federico.stizza@gmail.com*.

El plazo de entrega es de 1 semana desde que recibió este documento. Se estima que como máximo necesitará cuatro horas para resolver el problema.

Le solicito que contabilice el tiempo neto que utilizó para resolver el problema ya que en la encuesta de cierre habrá una pregunta relacionada.

2. Introducción

Deberá implementar **seis** operaciones básicas de un cajero automático.

Junto con este documento se le entregarán dos archivos que le servirán de base para empezar la solución.

Dichos archivos varían según el lenguaje que haya seleccionado y son:

1. Estado.cs (C#), estado.js (JavaScript) y estado.py (Python).
2. Programa.cs (C#), index.js (JavaScript) y main.py (Python).

Los archivos del primer punto contienen una clase llamada **Estado** que deberán completar de acuerdo a lo solicitado en la especificación. La misma provee una interfaz para guardar y cargar el estado en un archivo.

Los del segundo punto corresponden al punto de entrada del sistema, el cuál reúne todas las operaciones que debe implementar y se ejecuta de la siguiente manera:

C#	dotnet run -project Solucion.csproj operacion arg1 arg2 ...
JavaScript	node index.js operacion arg1 arg2 ...
Python	python3 main.py operacion arg1 arg2 ...

3. Consideraciones generales

- Las operaciones se deben poder ejecutar secuencialmente y deben persistir los cambios de estado según corresponda.
- Si le resulta útil para realizar pruebas puede modificar manualmente el estado que se encuentra en el archivo **estado.json**.
- Puede incluir en su solución librerías de terceros o paquetes que crea necesario para resolver el requerimiento.
- El programa que corresponde al punto de entrada de la solución no debe ser modificado, debe mantener los nombres de las operaciones y el orden de los argumentos según se indica en la siguiente tabla.

- Aquellas operaciones que *retornan* algo deben imprimirlo en pantalla utilizando la salida estándar.

Operación	Invocación
Extracción	extraccion dni clave monto
Cambio de Clave	clave dni actual nueva
Consulta de Saldo	saldo dni clave
Alta de Usuario	alta dni_administrador clave_administrador dni clave nombre sueldo saldo
Carga de Cajero	carga dni_administrador clave_administrador monto
Consulta de movimientos	movimientos dni_administrador clave_administrador dni_consulta desde hasta

3.1. Requerimiento

Los usuarios deben autenticarse con *número de documento* y *clave*.

Existe un *usuario especial* con el que se pueden realizar operaciones administrativas.

Los datos que se poseen de cada usuario son:

1. Número de documento
2. Nombre y apellido
3. Sueldo mensual

Las operaciones a implementar son las siguientes:

- (01) Extracción
- (02) Cambio de clave
- (03) Consulta de saldo
- (04) Alta de usuario
- (05) Carga cajero
- (06) Consulta movimientos

Todas las operaciones requieren autenticación con número de documento y clave.

Las operaciones (04), (05) y (06) están habilitadas para el **usuario administrador**.

Las operaciones de Extracción y Cambio de clave deben ser persistidos en un **histórico** de movimientos.

3.2. Operaciones

3.2.1. Extracción

Extrae el monto especificado de la cuenta del usuario autenticado.

El usuario solo puede realizar tres extracciones por día y el monto de cada una no puede superar la mitad de su sueldo.

Se debe contemplar que el cajero tenga dinero suficiente para entregarle al usuario.

3.2.2. Cambio de clave

Asignar la nueva clave especificada al usuario autenticado.

Modifica la clave existente si la nueva clave proporcionada posee al menos 8 caracteres y es una combinación de letras y números.

Los usuarios solo pueden modificar su clave una vez por mes.

3.2.3. Consulta de saldo

Retorna el saldo de la cuenta del usuario autenticado.

3.2.4. Alta usuario

Da de alta un nuevo usuario.

Cada cajero puede tener registrado en el sistema hasta 300 usuarios.

El saldo inicial de los usuarios debe ser igual al sueldo mensual declarado.

3.2.5. Carga de cajero

Acredita el monto especificado al cajero.

3.2.6. Consulta movimientos

Retorna los movimientos realizados por un usuario particular en el rango de fechas especificado.