

Proyecto - Banco



Programación II
Pregrado
2024-0
Ciencia de la Computación
Laboratorio 1.02

Febrero 2024

I. ENUNCIADO DEL PROYECTO

La idea del proyecto es que puedan simular los servicios que brinda un Banco.

Requerimientos funcionales para el proyecto:

1. El programa debe mostrar un menú listando los servicios que ofrece el Banco, puede ser:
 - a. **Lista de clientes**, donde se puede agregar más clientes
 - b. **Cuentas de ahorro**, asociadas a tarjetas de débito.
 - c. **Prestamos de dinero**.
 - d. Depósitos a plazo fijo.
 - e. Créditos Hipotecarios.
 - f. Cuentas de crédito, asociadas a tarjetas de crédito

Obligatorio es a, b y c

2. **Lista de clientes:** El programa debe permitir :
 - a. Buscar, agregar, eliminar clientes
 - b. Listar los servicios que el cliente tiene con el Banco
3. **Cuentas de ahorro:** El cliente puede tener una o varias cuentas, cada cuenta asociada a una tarjeta de débito.
 - a. El cliente puede solicitar bloquear la tarjeta de débito y solicitar otra.
 - b. Se debe poder **registrar los ingresos de dinero**, puede ser: Descripción, monto, fecha, cuenta destino.
 - c. Se debe poder visualizar todos los ingresos de dinero.
 - d. El cliente puede solicitar cancelar su cuenta de ahorro.
 - e. El programa debe listar todas las cuentas de ahorro incluso las canceladas.
4. **Préstamos de dinero:** El cliente puede solicitar un préstamo de dinero.
 - a. El programa debería mostrar el cronograma de pagos, incluyendo información de los intereses cobrados.
 - b. El programa debe permitir cancelar el préstamo en cualquier momento.
 - c. El programa debe listar todos los préstamos realizados al cliente, incluyendo los pagados.

5. Cuentas de crédito: El cliente puede tener una cuenta de crédito, y esa cuenta asociada a una tarjeta de crédito
 - a. El cliente puede solicitar bloquear la tarjeta de crédito y solicitar otra.
 - b. Una tarjeta de crédito debe tener: monto límite, fecha de corte, y fecha de pago.
 - c. El cliente puede solicitar cancelar su cuenta de crédito.
6. **El programa debe imprimir reportes**, por ejemplo:
 - a. Reporte actual del saldo de cada cuenta.
 - b. Reporte de los préstamos realizados
7. Opcional: Que el programa pueda guardar toda la información en archivos y después pueda leer ese archivo y seguir registrando.
8. Puede incluir las clases y funcionalidades que desee.
9. El programa debe mostrar mensajes de errores adecuados y debe permitir volver a ingresar otro dato.

II. PROGRAMACIÓN / REQUISITOS TÉCNICOS

Se debe programar usando C++, en modo consola, usando Programación Orientada a Objetos.

Se debe usar **clases**, punteros, arreglos, matrices, vectores, **relaciones entre clases**(asociación, agregación, composición, herencia), **polimorfismo por herencia** y sobrecarga de operadores, use su creatividad para usar esos temas.

III. EJEMPLOS (Solo de referencia, pueden crear su propio menú)

Menú

1. Clientes
 2. Cuentas de ahorro
 3. Prestamos de dinero
 4. Salir
- Opción: 1

1. Clientes

- 1.1 Buscar cliente
 - 1.2 Agregar cliente
 - 1.3 Regresar
- Opción: 1

1.1 Buscar cliente

Ingresar DNI: 34981234

====

Servicios

Cuenta de ahorro: 0011-938485 Monto S/: 3500.00 Activa

Tarjeta débito asociada: 3456-1234-2345-1234 Venc: 12/27

Prestamos:

Fecha	Monto S/	Nro	Cuota S/	Interes	Estado
12/02/2023	5000.00	12	500.00	20%	Activo

1 Regresar

9 Eliminar Cliente

Opción: 1

1. Clientes

1.1 Buscar cliente

1.2 Agregar cliente

1.3 Regresar

Opción: 2

1.2 Agregar cliente

Ingresar DNI: 85761234

Ingresar Nombre: Mario

Ingresar Fecha Nacimiento:

Cliente Registrado

Menú

1 Clientes

2 Cuentas de ahorro

3 Préstamos de dinero

4 Salir

Opción: 3

3 Préstamos de dinero

Ingresar DNI: 85761234

====

Prestamos:

Fecha	Monto S/	Nro	Cuota S/	Interes	Estado
12/02/2023	5000.00	12	500.00	20%	Activo

1 Regresar

2 Pagar préstamo

IV. ENTREGA DEL PROYECTO

Los grupos deben ser de máximo 5 alumnos.

La entrega del Proyecto es por gradescope, hasta el mismo día de la exposición.

Subir a Gradescope un informe del Proyecto en PDF, conteniendo:

- Título del Proyecto
- Integrantes
- Antecedentes (contexto, motivación, el porque),
- Fundamento Teórico (definiciones de conceptos),
- Métodos y Desarrollo (diagrama de clases, descripción de algunos métodos, y características de programación) y
- Conclusiones.

Use Diagrama de clases y algunas capturas de la ejecución del código

Subir a Gradescope el código del proyecto, solo archivos .h, .cpp, .txt y el cmakefile

V. CRITERIOS DE EVALUACIÓN:

Criterio	EXCELENTE	ADECUADO	MÍNIMO	INSUFICIENTE
Software	Diseña y elabora el software para lograr una solución adecuada al problema planteado. El software debe ser ordenado, claro y óptimo (8 pts)	Diseña y elabora el software para lograr una solución adecuada al problema planteado. El software es solo funcional (5 pts)	Diseña el software para lograr una solución adecuada al problema planteado. El software no se concluye adecuadamente (3 pts)	No logra el diseño ni la implementación correcta del software (1 pts)
Presentación escrita (informe)	El informe contiene las secciones de Antecedentes, Fundamento Teórico, Métodos y Desarrollo y Conclusiones. Estas últimas, adecuadamente formuladas (7 pts)	El informe contiene las secciones de Antecedentes, Fundamento Teórico, Métodos y Desarrollo, pero no pone énfasis en las conclusiones (5 pts)	El informe contiene menos de la mitad de las secciones estipuladas, incluyendo conclusiones (3 pts)	El informe contiene menos de la mitad de las secciones estipuladas, sin incluir conclusiones (1 pts)
Presentación oral	El alumno presenta el proyecto en forma adecuada y responde a las preguntas del profesor en forma lógica y coherente (5 pts)	El alumno presenta el proyecto en forma adecuada y pero no responde a todas las preguntas del profesor en forma lógica y coherente (3 pts)	El alumno no presenta el proyecto en forma adecuada pero responde a las preguntas del profesor en forma lógica y coherente (2 pts)	El alumno no presenta el proyecto en forma adecuada ni responde a las preguntas del profesor en forma lógica y coherente. O no se presenta a la

				presentacion oral (0 pts)
--	--	--	--	------------------------------