Ingeniería en Desarrollo de Software

# Actividad 2 Programa 2 Banco Mexicano

## Parte 1

## Desarrollo de Aplicaciones Móviles IV

Ingeniería en Desarrollo de Software

TUTOR: Marco Alonso Rodríguez Tapia

ALUMNO: Homero Ramirez Hurtado

FECHA: 9 de Abril del 2025

Índice.

. Introducción.

. Descripción.

. Justificación.

. Desarrollo.

- Codificación.
- Prueba del Programa.

. Conclusión.

. Referencias.

## Introducción.

El desarrollo de aplicaciones bancarias es un aspecto crucial en la transformación digital del sector financiero. En este contexto, el Banco Mexicano busca implementar una solución eficiente para su banca en línea, permitiendo a los usuarios realizar operaciones esenciales de manera segura y rápida. Con esta aplicación, los clientes podrán depositar fondos, efectuar retiros, consultar su saldo y cerrar sesión de manera intuitiva.

Para garantizar un rendimiento óptimo y una experiencia de usuario fluida, la aplicación será desarrollada utilizando Swift, el lenguaje de programación de Apple diseñado específicamente para la creación de software en sus plataformas. Swift ofrece una sintaxis moderna y segura, optimizando la ejecución del código y minimizando errores, lo que es esencial para una aplicación bancaria donde la precisión y la fiabilidad son fundamentales.

Este proyecto no solo facilitará el acceso a los servicios financieros del banco, sino que también reforzará la seguridad y comodidad de sus clientes, permitiéndoles gestionar sus cuentas desde cualquier lugar a través de sus dispositivos iOS. Con esta solución, el Banco Mexicano se posiciona a la vanguardia de la innovación tecnológica en el sector bancario.

## Descripción.

El desarrollo de una aplicación bancaria en Swift permitirá a los usuarios gestionar sus cuentas de manera eficiente y segura. La estructura del programa se basa en un menú interactivo con cuatro opciones principales: depósito, retiro, consulta de saldo y salida. Estas funciones garantizarán que los clientes puedan realizar transacciones esenciales sin complicaciones.

Cuando el usuario seleccione la opción de depósito, el sistema solicitará la cantidad a ingresar y ofrecerá la posibilidad de realizar otra operación. Esto asegura flexibilidad en la administración de fondos. En el caso del retiro, la aplicación implementará medidas de control para verificar si el usuario tiene saldo disponible. Si el cliente ingresa por primera vez, recibirá una notificación indicando que su cuenta está vacía. En caso de que ya tenga dinero, el sistema validará que el monto solicitado sea accesible y, si no lo es, mostrará un mensaje de advertencia.

Esta funcionalidad no solo optimiza la seguridad financiera del usuario, sino que también mejora la experiencia bancaria digital. Al ofrecer interacción sencilla y decisiones automatizadas, el programa permitirá una gestión clara y eficaz del dinero, adaptándose a las necesidades modernas de la banca en línea.

## Justificación.

Replit es una excelente opción como entorno de desarrollo integrado (IDE) para la creación del programa de banca en línea en Swift. Su accesibilidad y versatilidad lo convierten en una herramienta ideal para desarrolladores que buscan una plataforma intuitiva y eficiente para escribir, ejecutar y probar código sin complicaciones.

Una de las principales razones para utilizar Replit es su capacidad de funcionar completamente en línea, eliminando la necesidad de instalaciones locales. Esto permite a los programadores acceder a sus proyectos desde cualquier dispositivo con conexión a Internet, facilitando el desarrollo colaborativo. Además, su compatibilidad con múltiples lenguajes de programación, incluido Swift, brinda una experiencia fluida para la creación de software.

Otra ventaja clave es su interfaz sencilla, que permite escribir y ejecutar código con rapidez. Replit también cuenta con un sistema de ejecución en tiempo real, lo que ayuda a detectar errores y optimizar el desarrollo. Gracias a su integración con Git y otras herramientas de control de versiones, es posible gestionar cambios de manera eficiente.

En definitiva, Replit es una solución práctica y accesible que simplifica el proceso de desarrollo de software, permitiendo una ejecución rápida y eficiente del proyecto de banca en línea sin comprometer la calidad y seguridad del código.

Desarrollo.

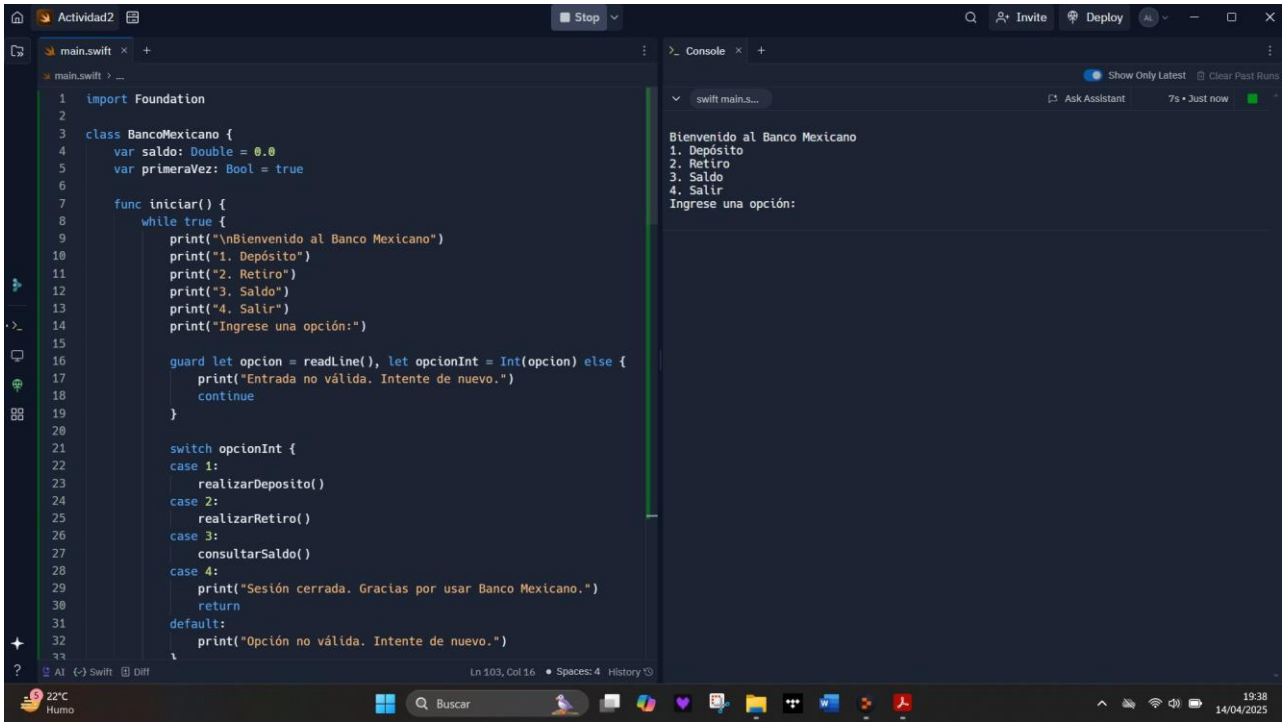
Codificación.

```
import Foundation


class BancoMexicano {
    var saldo: Double = 0.0
    var primeraVez: Bool = true


    func iniciar() {
        while true {
            print("\nBienvenido al Banco Mexicano")
            print("1. Depósito")
            print("2. Retiro")
            print("3. Saldo")
            print("4. Salir")
            print("Ingrese una opción:")


            guard let opcion = readLine(), let opcionInt = Int(opcion) else {
                print("Entrada no válida. Intente de nuevo.")
                continue
            }


            switch opcionInt {
            case 1:
```

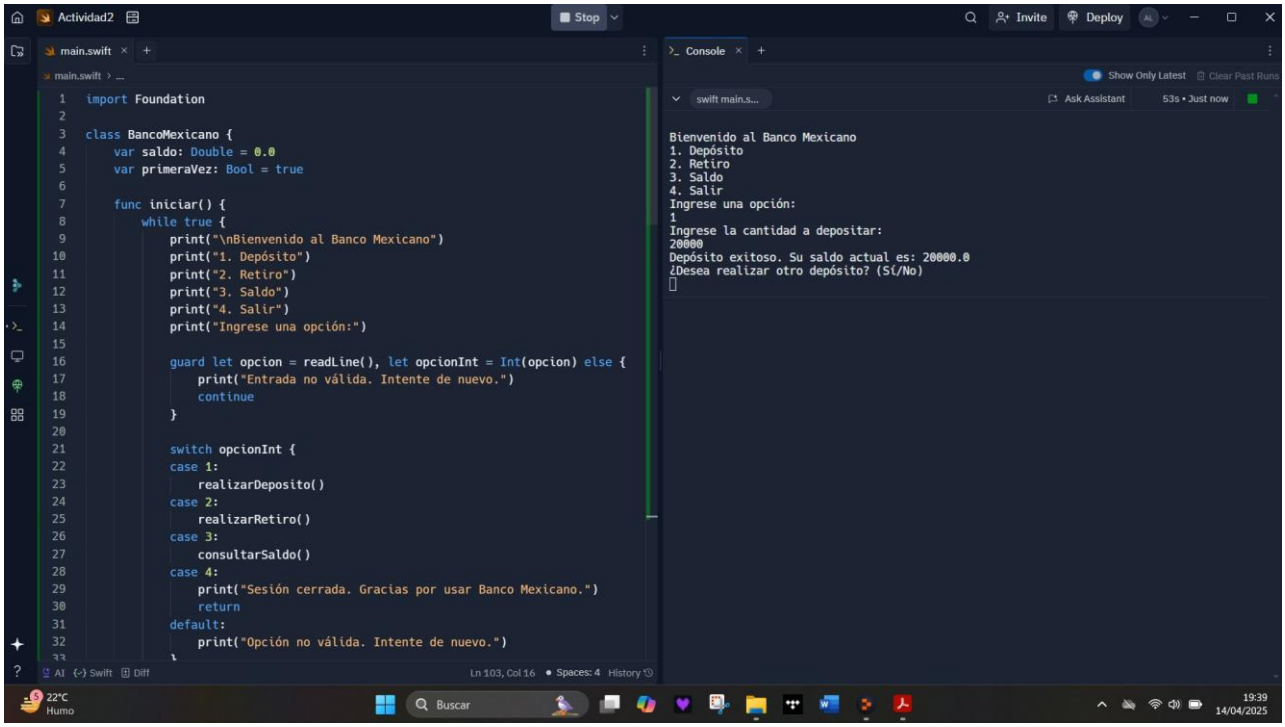
```
            realizarDeposito()
        case 2:
            realizarRetiro()
        case 3:
            consultarSaldo()
        case 4:
            print("Sesión cerrada. Gracias por usar Banco Mexicano.")
            return
        default:
            print("Opción no válida. Intente de nuevo.")
        }
    }
}

func realizarDeposito() {
    while true {
        print("Ingrese la cantidad a depositar:")

        guard let cantidad = readLine(), let cantidadDouble = Double(cantidad), cantidadDouble > 0
else {
            print("Cantidad no válida. Intente de nuevo.")
            continue
        }

        saldo += cantidadDouble
        print("Depósito exitoso. Su saldo actual es: \(saldo)")

        print("¿Desea realizar otro depósito? (Sí/No)")
        if let respuesta = readLine(), respuesta.lowercased() == "no" {
            preguntarOtraOperacion()
```

```
            break
        }
    }
}

func realizarRetiro() {
    if primeraVez {
        print("No cuenta con saldo en su primera operación.")
        primeraVez = false
        preguntarOtraOperacion()
        return
    }

    while true {
        print("Ingrese la cantidad a retirar:")

        guard let cantidad = readLine(), let cantidadDouble = Double(cantidad), cantidadDouble > 0
else {
            print("Cantidad no válida. Intente de nuevo.")
            continue
        }

        if cantidadDouble > saldo {
            print("Saldo insuficiente. Su saldo actual es: \(saldo)")
        } else {
            saldo -= cantidadDouble
            print("Retiro exitoso. Su saldo actual es: \(saldo)")
        }

        print("¿Desea realizar otro retiro? (Sí/No)")
```

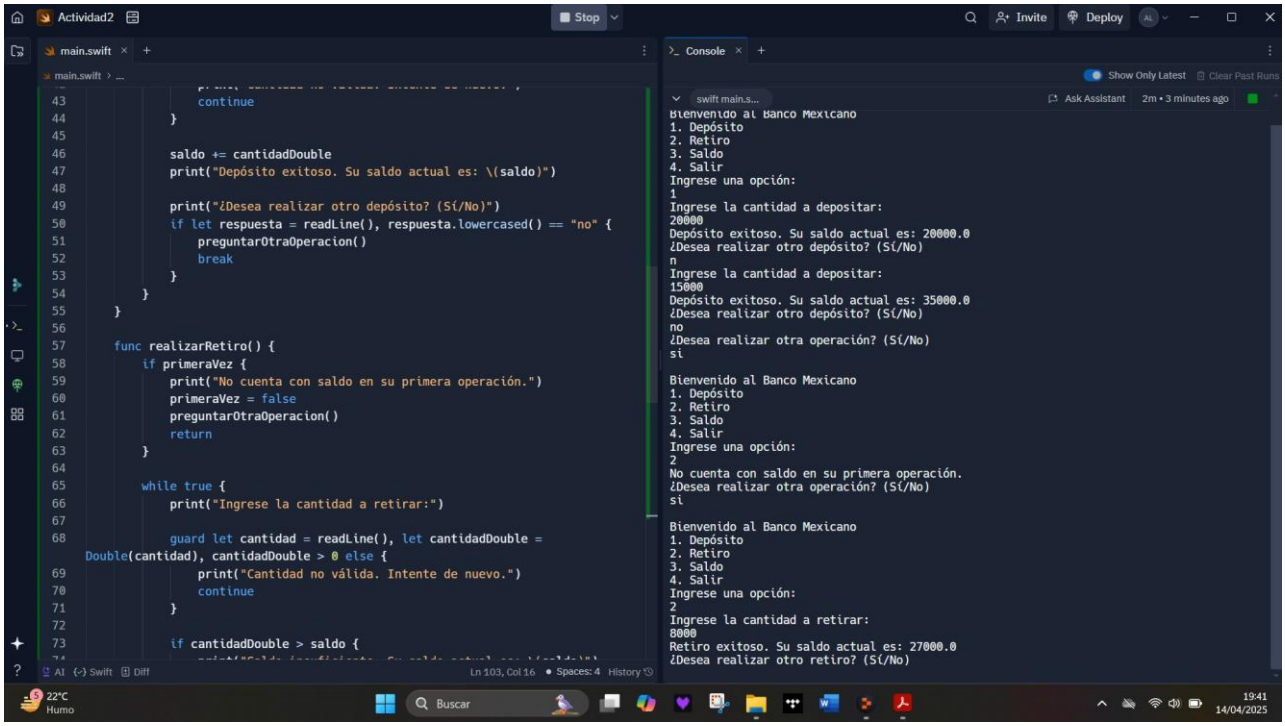
```
            if let respuesta = readLine(), respuesta.lowercased() == "no" {
                preguntarOtraOperacion()
                break
            }
        }
    }

    func consultarSaldo() {
        print("Su saldo actual es: \(saldo)")
        preguntarOtraOperacion()
    }

    func preguntarOtraOperacion() {
        print("¿Desea realizar otra operación? (Sí/No)")
        if let respuesta = readLine(), respuesta.lowercased() == "no" {
            print("Sesión cerrada. Gracias por usar Banco Mexicano.")
            exit(0)
        }
    }
}

let banco = BancoMexicano()
banco.iniciar()
```



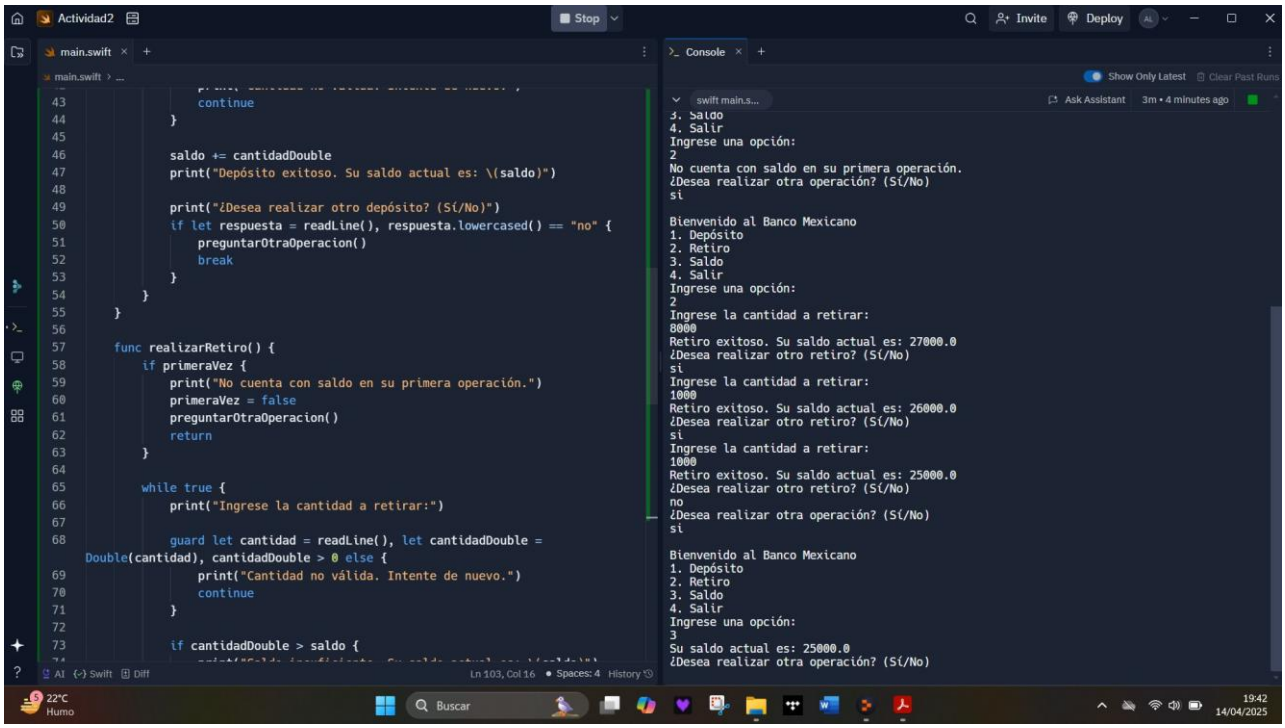

Menú.




Deposito.


Actividad2
Stop
Invite
Deploy
main.swift
Console
Show Only Latest
Clear Past Runs
Ask Assistant
2m • 3 minutes ago
continue
}
saldo += cantidadDouble
print("Depósito exitoso. Su saldo actual es: \(saldo)")
print("¿Desea realizar otro depósito? (Sí/No)")
if let respuesta = readLine(), respuesta.lowercased() == "no" {
preguntarOtraOperacion()
break
}
}
}
func realizarRetiro() {
if primeraVez {
print("No cuenta con saldo en su primera operación.")
primeraVez = false
preguntarOtraOperacion()
return
}
while true {
print("Ingrese la cantidad a retirar:")
guard let cantidad = readLine(), let cantidadDouble = Double(cantidad), cantidadDouble > 0 else {
print("Cantidad no válida. Intente de nuevo.")
continue
}
if cantidadDouble > saldo {
Bienvenido al Banco Mexicano
1. Depósito
2. Retiro
3. Saldo
4. Salir
Ingrese una opción:
1
Ingrese la cantidad a depositar:
20000
Depósito exitoso. Su saldo actual es: 20000.0
¿Desea realizar otro depósito? (Sí/No)
n
Ingrese la cantidad a depositar:
15000
Depósito exitoso. Su saldo actual es: 35000.0
¿Desea realizar otro depósito? (Sí/No)
no
¿Desea realizar otra operación? (Sí/No)
si
Bienvenido al Banco Mexicano
1. Depósito
2. Retiro
3. Saldo
4. Salir
Ingrese una opción:
2
No cuenta con saldo en su primera operación.
¿Desea realizar otra operación? (Sí/No)
si
Bienvenido al Banco Mexicano
1. Depósito
2. Retiro
3. Saldo
4. Salir
Ingrese una opción:
2
Ingrese la cantidad a retirar:
8000
Retiro exitoso. Su saldo actual es: 27000.0
¿Desea realizar otro retiro? (Sí/No)
Ln 103, Col 16
Spaces: 4
History
22°C
Humo
Buscar
19:41
14/04/2025

Retiro.


Actividad2
Stop
Invite
Deploy
main.swift
Console
Show Only Latest
Clear Past Runs
Ask Assistant
3m • 4 minutes ago
continue
}
saldo += cantidadDouble
print("Depósito exitoso. Su saldo actual es: \(saldo)")
print("¿Desea realizar otro depósito? (Sí/No)")
if let respuesta = readLine(), respuesta.lowercased() == "no" {
preguntarOtraOperacion()
break
}
}
}
func realizarRetiro() {
if primeraVez {
print("No cuenta con saldo en su primera operación.")
primeraVez = false
preguntarOtraOperacion()
return
}
while true {
print("Ingrese la cantidad a retirar:")
guard let cantidad = readLine(), let cantidadDouble = Double(cantidad), cantidadDouble > 0 else {
print("Cantidad no válida. Intente de nuevo.")
continue
}
if cantidadDouble > saldo {
3. Saldo
4. Salir
Ingrese una opción:
2
No cuenta con saldo en su primera operación.
¿Desea realizar otra operación? (Sí/No)
si
Bienvenido al Banco Mexicano
1. Depósito
2. Retiro
3. Saldo
4. Salir
Ingrese una opción:
2
Ingrese la cantidad a retirar:
8000
Retiro exitoso. Su saldo actual es: 27000.0
¿Desea realizar otro retiro? (Sí/No)
si
Ingrese la cantidad a retirar:
1000
Retiro exitoso. Su saldo actual es: 26000.0
¿Desea realizar otro retiro? (Sí/No)
si
Ingrese la cantidad a retirar:
1000
Retiro exitoso. Su saldo actual es: 25000.0
¿Desea realizar otro retiro? (Sí/No)
no
¿Desea realizar otra operación? (Sí/No)
si
Bienvenido al Banco Mexicano
1. Depósito
2. Retiro
3. Saldo
4. Salir
Ingrese una opción:
3
Su saldo actual es: 25000.0
¿Desea realizar otra operación? (Sí/No)
Ln 103, Col 16
Spaces: 4
History
22°C
Humo
Buscar
19:42
14/04/2025

Conclusión.

Finalizar la implementación de las opciones de **Depósito, Retiro y Saldo** en este proyecto de banca en línea ha sido un paso crucial en la creación de un sistema funcional y seguro. No solo ha sido un ejercicio técnico, sino también una experiencia enriquecedora que demuestra lo esencial que es la lógica y la estructura bien diseñada en el desarrollo de software. Cada línea de código escrita para estas funciones representa un avance hacia una aplicación eficiente, que permitirá a los usuarios administrar sus finanzas de manera sencilla y confiable.

Lo más gratificante de este proceso ha sido ver cómo cada funcionalidad cobra vida y responde de manera intuitiva a las necesidades del usuario. Validar el saldo antes de permitir un retiro, asegurarse de que los depósitos sean registrados correctamente y mostrar el saldo disponible en pantalla son detalles que, aunque parecen simples, marcan la diferencia en la experiencia de usuario.

Este avance refuerza la importancia de un código claro y bien estructurado, y sienta las bases para continuar mejorando la aplicación. Cada ajuste y optimización nos acerca más a un producto sólido, confiable y útil.

Referencias.

Tutoría 2.

Copilot.