

Laboratory #1

SOFTWARE REQUIREMENTS

Laura Ximena Gordillo Maldonado, Maria Valentina Molina Florez, Daniel Arturo Moreno Rincón
Software-Engineering I
2019-I

DESCRIPCIÓN DEL SISTEMA

El cliente necesita un sistema de software para una organización bancaria, una aplicación con usuarios y gestión de cuentas de ahorros. El sistema debe tener un único administrador y un conjunto de usuarios, cada usuario tiene una cuenta de ahorros asociada en la que puede enviar dinero a otra cuenta y revisar su saldo. Además el administrador es el único que puede crear usuarios, cuentas de ahorro y es el único que puede modificar una cuenta de ahorro agregando o descontando dinero, teniendo en cuenta que el administrador no puede ser un usuario.

REQUISITOS FUNCIONALES

Historias de uso:

No.	Rol	Característica/Funcionalidad	Razón/Resultado
US001	Como administrador	Quiero que el sistema disponga de solamente un actor con este rol.	Con el fin de tener control absoluto sobre las funcionalidades y evitar percances respecto a seguridad.
US002	Como usuario	Quiero tener una cuenta de ahorros asociada.	Con el fin de poder consignar, transferir y enviar dinero a otras cuentas.
US003	Como administrador	Quiero tener capacidad de agregar o descontar del saldo de cualquier cuenta de ahorros.	Con el fin de que nadie más pueda realizar cambios directos en las cuentas de ahorros.
US004	Como administrador	Quiero no tener las facultades del rol de usuario.	Con el fin de que exista integridad en las funciones del sistema.

Casos de uso:

Caso de Uso	Enviar Dinero			No.:	UC0001
Descripción	Enviar dinero de la cuenta de ahorros de un usuario, a la cuenta de ahorros de otro usuario.				
Actor(es)	Usuario 1: usuario que envía el dinero. Usuario 2: usuario que recibe el dinero.				
Pre-condiciones	El usuario 1 debe poseer una cuenta de ahorros en el banco. El usuario 2 debe poseer una cuenta de ahorros en el banco. El usuario 1 debe estar registrado en el sistema. El usuario 2 debe estar registrado en el sistema. El usuario 1 debe estar autenticado en el sistema.				
Flujo Normal	<div><div>1.</div><div>El usuario 1 ingresa el número de cuenta del usuario 2.</div></div> <div><div>2.</div><div>El usuario 1 ingresa la cantidad de dinero a enviar.</div></div> <div><div>3.</div><div>El usuario 1 autoriza el envío de dinero.</div></div> <div><div>4.</div><div>El sistema verifica que el saldo de la cuenta del usuario 1 sea igual o superior a la cantidad de dinero a enviar.</div></div> <div><div>5.</div><div>El sistema resta la cantidad de dinero enviada, al saldo de la cuenta del usuario 1.</div></div> <div><div>6.</div><div>El sistema suma la cantidad de dinero enviada, al saldo de la cuenta del usuario 2.</div></div> <div><div>7.</div><div>El sistema notifica al usuario 1 la transacción satisfactoria.</div></div>				
Flujo Alterno	<div><div>1.</div><div>El usuario 1 ingresa el número de cuenta del usuario 2.</div></div> <div><div>2.</div><div>El usuario 1 ingresa la cantidad de dinero a enviar.</div></div> <div><div>3.</div><div>El usuario 1 autoriza el envío de dinero.</div></div> <div><div>4.</div><div>El sistema verifica que el saldo de la cuenta del usuario 1 sea igual o superior a la cantidad de dinero a enviar.</div></div> <div><div>5.</div><div>El sistema notifica al usuario 1 que su saldo es insuficiente para el monto de la transacción.</div></div>				
Autor(es):	lgordillom mmolinaf damorenor	Fecha:	30.04.2019	Versión:	1.0

Caso de Uso	Revisar Saldo			No.:	UC002
Descripción	Mostrar saldo disponible de la cuenta de ahorros de un usuario.				
Actor(es)	Usuario: revisa el saldo en su cuenta de ahorros.				
Pre-condiciones	El usuario debe poseer una cuenta de ahorros en el banco. El usuario debe estar registrado en el sistema.				
Flujo normal	<ol style="list-style-type: none"> 1. El usuario consulta su saldo. 2. El sistema muestra el saldo disponible en la cuenta al usuario. 				
Flujo alterno					
Autor(es):	lgordillom mmolinaf damorenor	Fecha:	05.05.2019	Versi ón:	1.0

Caso de Uso	Crear usuario			No.:	UC003
Descripción	El administrador crea un usuario nuevo.				
Actor(es)	Administrador: crea una cuenta				
Pre-condiciones	Administrador debe estar autenticado.				
Flujo normal	<ol style="list-style-type: none"> 1. El administrador ingresa la cedula del nuevo usuario. 2. El sistema valida que la cédula no se encuentre registrada. 3. El administrador asigna una contraseña al nuevo usuario. 4. El administrador ingresa el nombre completo del nuevo usuario. 5. El sistema registra el nuevo usuario. 				
Flujo alterno	<ol style="list-style-type: none"> 1. El administrador ingresa la cedula del nuevo usuario. 2. El sistema valida que la cédula no se encuentre registrada. 3. El sistema informa que no se puede crear el usuario porque ya está registrado. 				
Autor(es):	lgordillom mmolinaf damorenor	Fecha:	05.05.2019	Versi ón:	1.0

Caso de Uso	Crear cuenta de ahorro			No.:	UC004
Descripción	El administrador crea una nueva cuenta de ahorros.				
Actor(es)	Administrador: Crea cuenta de ahorro Usuario: se le asocia la cuenta de ahorro				
Pre-condiciones	Usuario debe estar registrado en el sistema Administrador debe estar autenticado				
Flujo normal	<ol style="list-style-type: none"> 1. El administrador crea la cuenta de ahorros. 2. El administrador asigna el número de la cuenta. 3. El sistema valida que el número de cuenta esté disponible. 4. El administrador asocia la cuenta de ahorros al usuario. 				
Flujo alterno	<ol style="list-style-type: none"> 1. El administrador crea la cuenta de ahorros. 2. El administrador asigna el número de la cuenta. 3. El sistema valida que el número de cuenta esté disponible. 4. El sistema notifica que el número de cuenta no está disponible. 				
Autor(es):	Igordillom mmolinaf damorenor	Fecha:	05.05.2019	Versión:	1.0

Caso de Uso	Iniciar sesión			No.:	UC005
Descripción	Permitir el ingreso al software				
Actor(es)	Usuario				
Pre-condiciones	Usuario : debe estar registrado en el sistema				
Flujo normal	<ol style="list-style-type: none"> 1. Usuario abre la aplicación 2. Usuario ingresa su identificación y contraseña 3. Se revisa que la información sea correcta 4. Usuario inicia sesión 				
Flujo alterno	<ol style="list-style-type: none"> 1. Usuario abre la aplicación 2. Usuario ingresa su identificación y contraseña 3. Se revisa que la información sea correcta 4. Información incorrecta 5. Iniciar sesión de nuevo 				
Autor(es):	Igordillom	Fecha:	05.05.2019	Versi	1.0

	mmolinaf damorenor			ón:	
--	-----------------------	--	--	-----	--

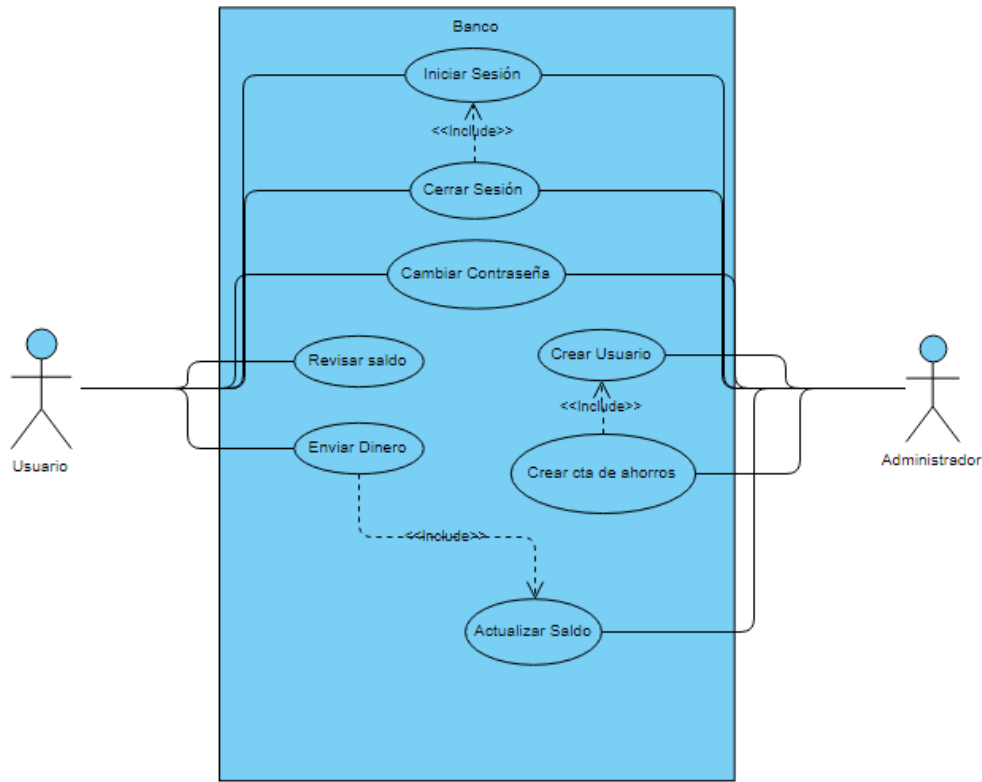
Caso de Uso	Cerrar sesión			No.:	UC006
Descripción	Permitir cerrar sesion en el sistema				
Actor(es)	Usuario				
Pre-condiciones	Usuario: está registrado en el sistema Usuario: ha iniciado sesión en el sistema				
Flujo normal	<ol style="list-style-type: none"> 1. Usuario decide cerrar sesión 2. Se confirma si desea cerrar sesión 3. Se cierra sesion del usuario 				
Flujo alterno	<ol style="list-style-type: none"> 1. Usuario decide cerrar sesión 2. Se confirma si desea cerrar sesión 3. No se cierra sesion del usuario 				
Autor(es):	Igordillom mmolinaf damorenor	Fecha:	05.05.2019	Versi ón:	1.0

Caso de Uso	Actualizar saldo			No.:	UC0006
Descripción	Actualizar saldo relacionado a la cuenta de ahorros de un usuario.				
Actor(es)	Administrador : modifica saldo de usuario.				
Pre-condiciones	Administrador debe estar autenticado. Usuario debe estar registrado en el sistema. Usuario tiene cuenta de ahorros asociada.				
Flujo normal	<ol style="list-style-type: none"> 1. El administrador hace ingreso del número de cuenta a modificar. 2. El administrador especifica si va a ingresar o retirar del saldo del número de cuenta especificada. 3. El sistema valida si es posible realizar dichos cambios en la cuenta especificada. 4. El sistema efectúa los cambios en la cuenta correspondiente. 				
Flujo alterno	<ol style="list-style-type: none"> 1. El administrador hace ingreso del número de cuenta a modificar. 				

	<ol style="list-style-type: none"> El administrador especifica si va a ingresar o retirar del saldo del número de cuenta especificada. El sistema valida si es posible realizar dichos cambios en la cuenta especificada. El sistema informa que no se puede modificar la cuenta correspondiente. 				
Autor(es):	Igordillom mmolinaf damorenor	Fecha:	05.05.2019	Versión:	1.0

Caso de Uso	Cambiar contraseña		No.:	UC0007	
Descripción	El sistema solicita al usuario cambio de contraseña				
Actor(es)	Usuario: cambia su contraseña				
Pre-condiciones	Usuario debe estar registrado en el sistema				
Flujo normal	<ol style="list-style-type: none"> 1. Usuario inicia sesión en el sistema. 2. El sistema le informa al usuario que debe cambiar contraseña. 3. El sistema solicita contraseña actual. 4. El sistema solicita nueva contraseña. 5. El sistema solicita de nuevo nueva contraseña. 6. El sistema verifica que las contraseñas nuevas sean iguales, y que la contraseña nueva no sea igual a la anterior. 7. El sistema cambia la contraseña. 8. El sistema informa al usuario que debe iniciar sesión de nuevo. 				
Flujo alterno	<ol style="list-style-type: none"> 1. Usuario inicia sesión en el sistema. 2. El sistema le informa al usuario que debe cambiar contraseña. 3. El sistema solicita contraseña actual. 4. El sistema solicita nueva contraseña. 5. El sistema solicita de nuevo nueva contraseña. 6. El sistema verifica que las contraseñas nuevas sean iguales, y que la contraseña nueva no sea igual a la anterior. 7. El sistema notifica que la contraseña no es válida. 8. El sistema solicita que el usuario lo intente de nuevo. 				
Autor(es):	Igordillom mmolinaf damorenor	Fecha:	05.05.2019	Versi ón:	1.0

Diagrama de Casos de Uso



REQUISITOS NO FUNCIONALES

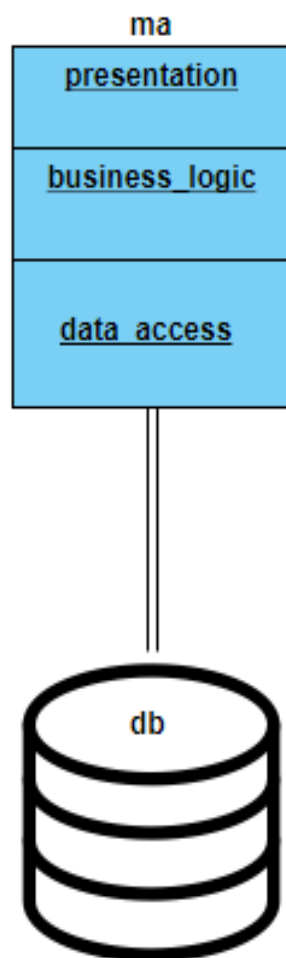
1. El sistema de software debe presentar una paleta de colores basada en azul y blanco, los cuales son parte de la identidad visual de la entidad bancaria.
2. El sistema de software debe almacenar del usuario por lo menos: identificación (Cedula), nombre y una clave de acceso.
3. El sistema de software debe permitir únicamente claves de acceso que estén compuestas de 6 caracteres alfanuméricos (alfabeto inglés).
4. El sistema de software debe registrar para cada administrador una identificación (Cedula) junto con una clave de acceso.
5. El sistema de software debería estar en actividad ininterrumpida en el horario estipulado por la entidad bancaria (Lunes a Viernes de 6:00am a 10:00pm).
6. El sistema de software debería ser capaz de soportar una concurrencia de hasta diez mil usuarios simultáneamente.
7. El sistema de software debe ser de tipo **aplicación móvil**.
8. El sistema de software debe ser construido para operar en sistema operativo **Android**.
9. El sistema de software debe ser implementado en lenguaje de programación **Java**.
10. El sistema de software debe hacer uso de una base de datos **relacional**.

Laboratory #2

SOFTWARE DESIGN

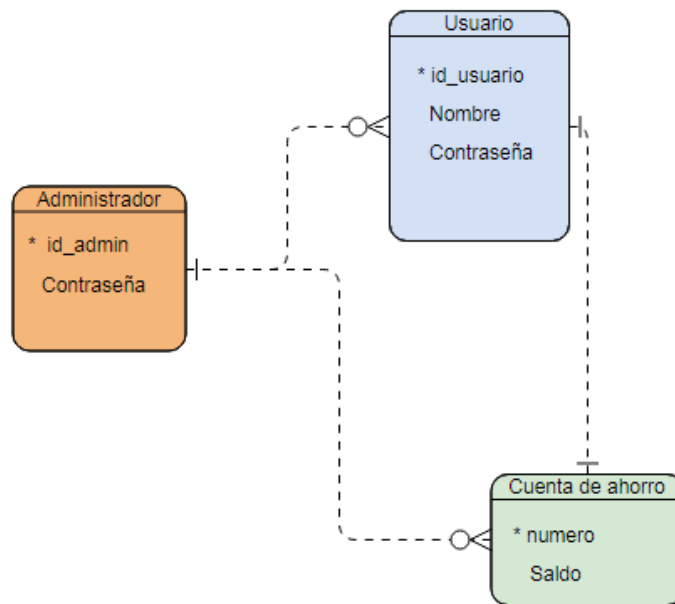
Laura Ximena Gordillo Maldonado, Maria Valentina Molina Florez, Daniel Arturo Moreno Rincón
Software-Engineering I
2019-I

i. Diseño Arquitectónico

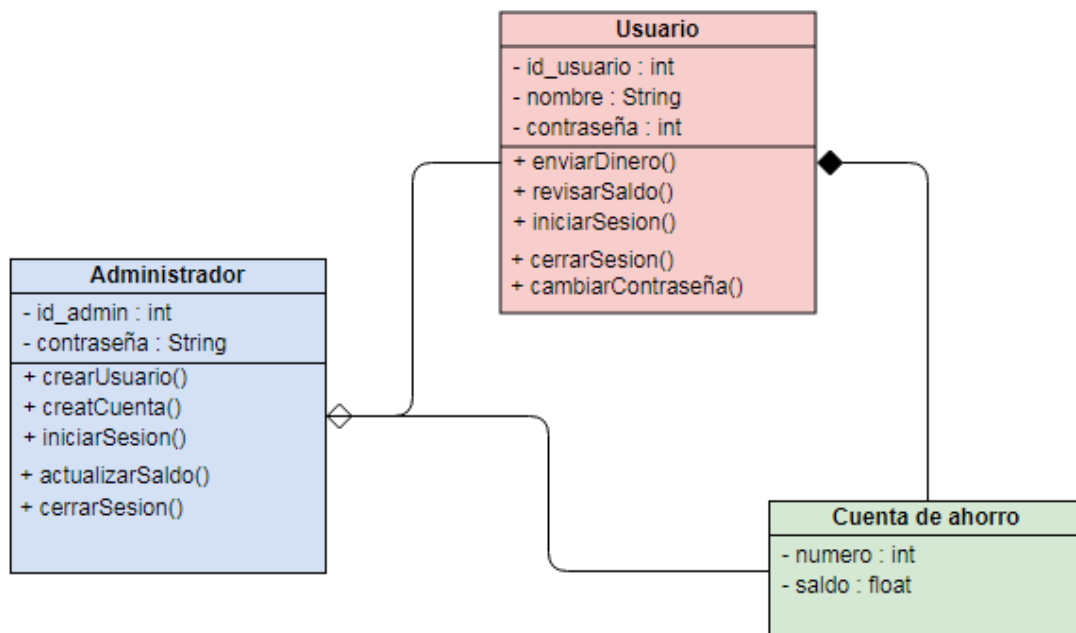


ii. Diseño Detallado

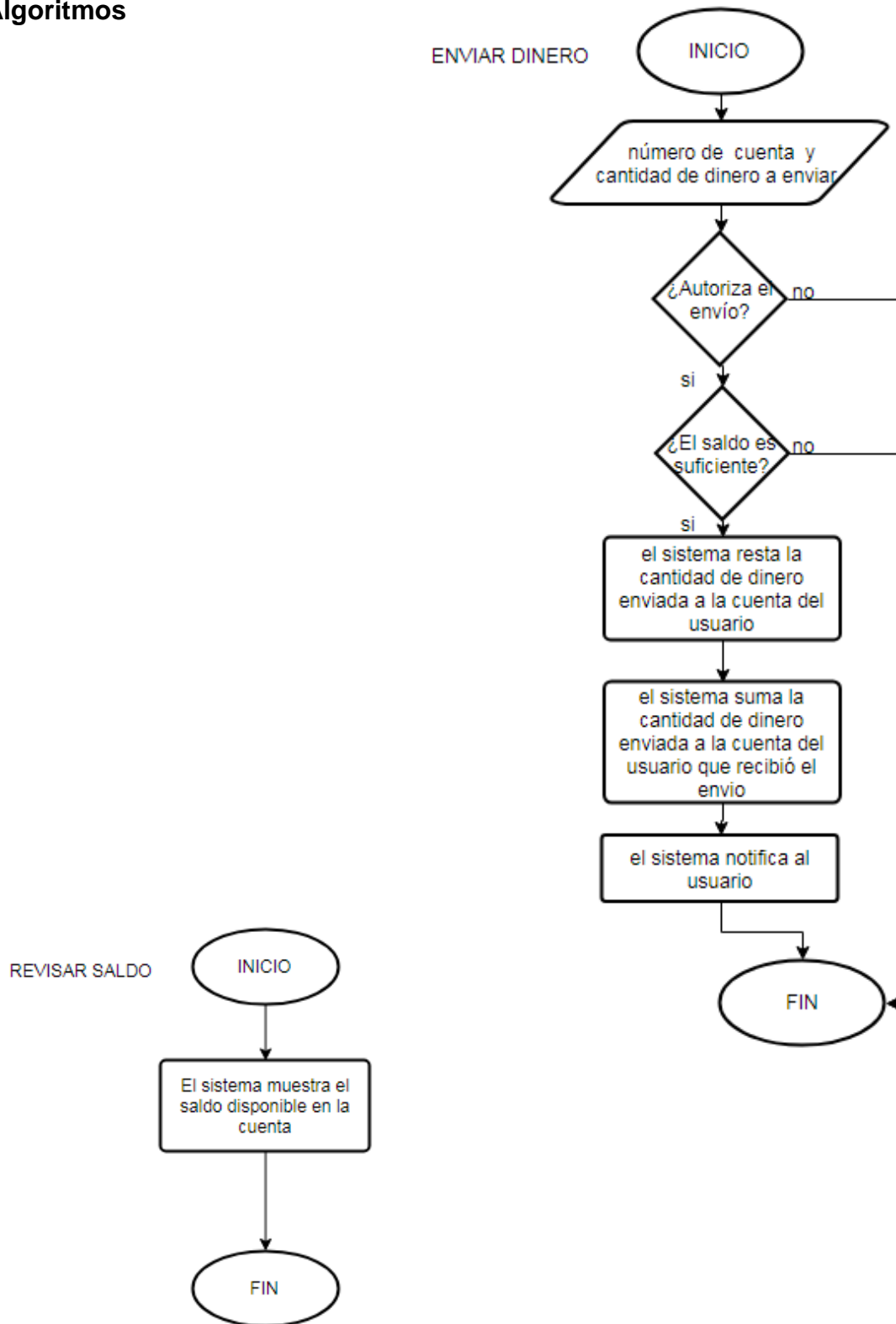
a. Datos



b. Modelado de Clases

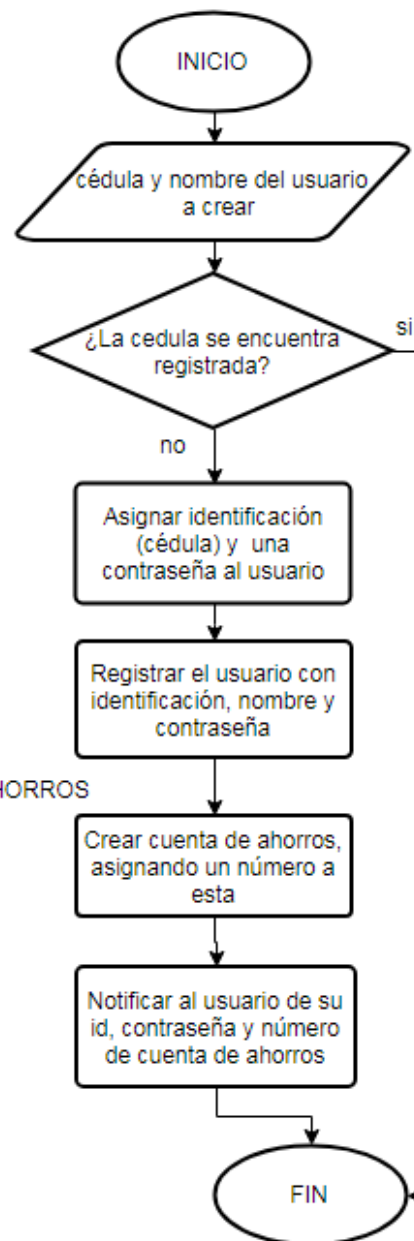


c. Algoritmos

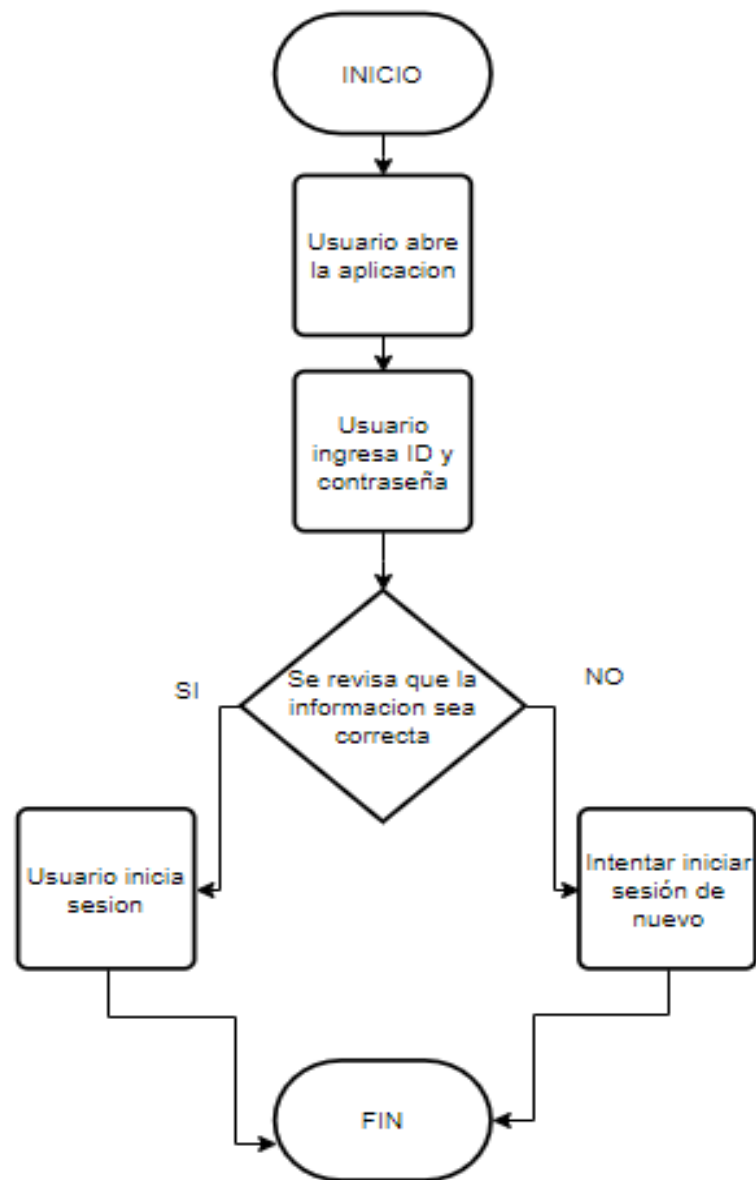


CREAR USUARIO

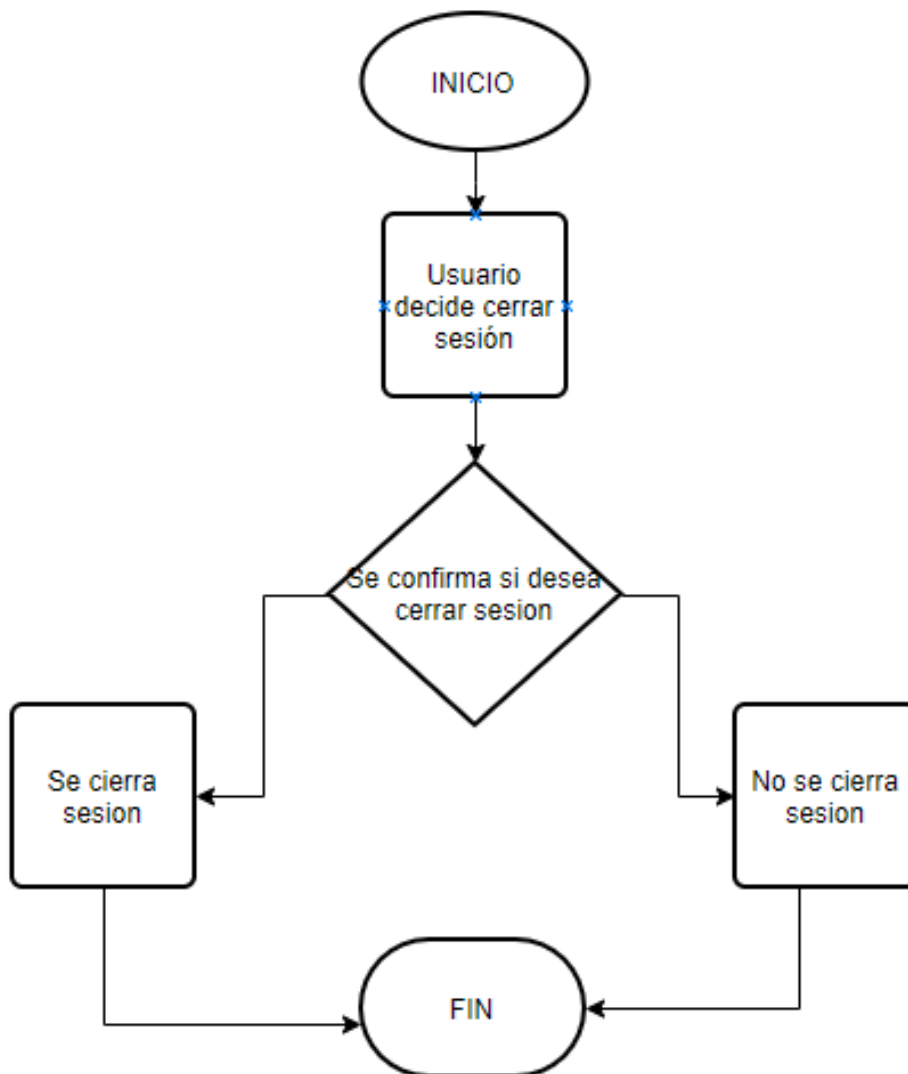
CREAR CUENTA DE AHORROS



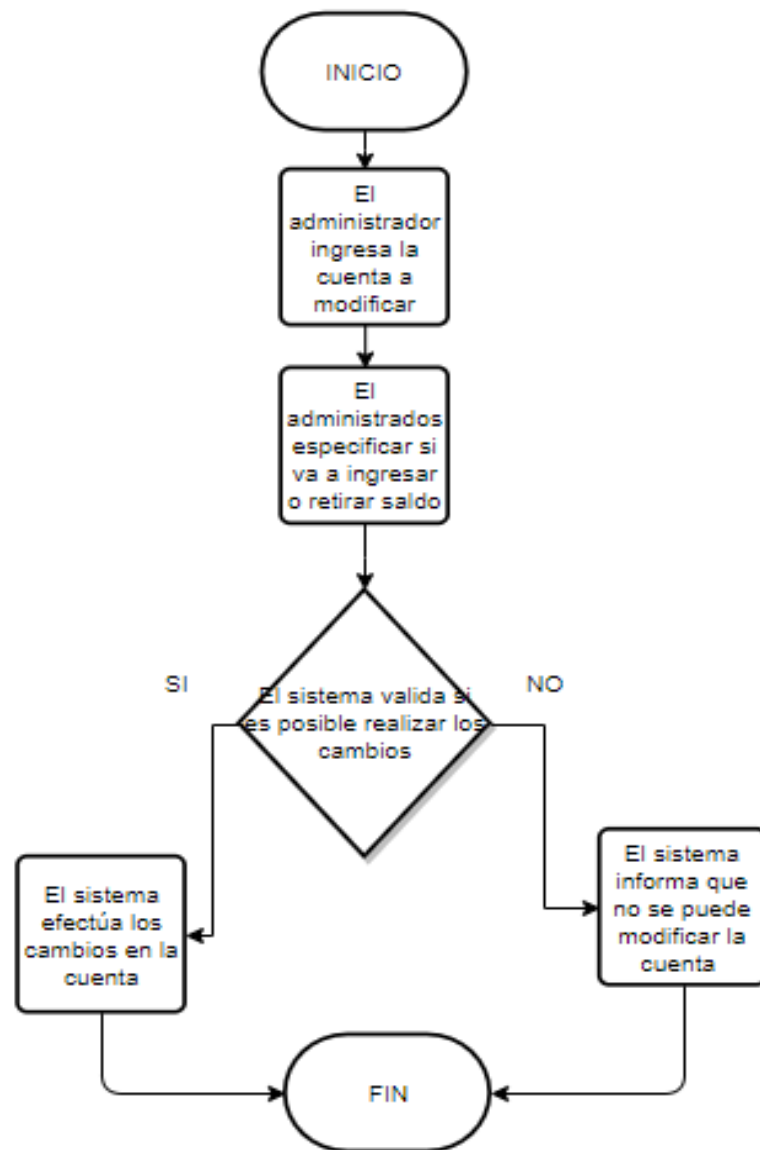
INCIAR SESION



CERRAR SESIÓN



ACTUALIZAR SALDO



CAMBIAR CONTRASEÑA

