

LABORATORIO 2
hpintom,jcristanchor,mbarragani

Descripción del sistema:

El sistema a construir es una aplicación que permita a los usuarios de una entidad financiera la administración de cuentas de ahorro , incluyendo las actividades de realizar traspasos de dinero entre usuarios, verificar saldo e historial de transacciones.

Dentro del aplicativo también existe un administrador quien tiene facultades para la creación y administración de usuarios en la entidad bancaria.

Requerimientos funcionales:

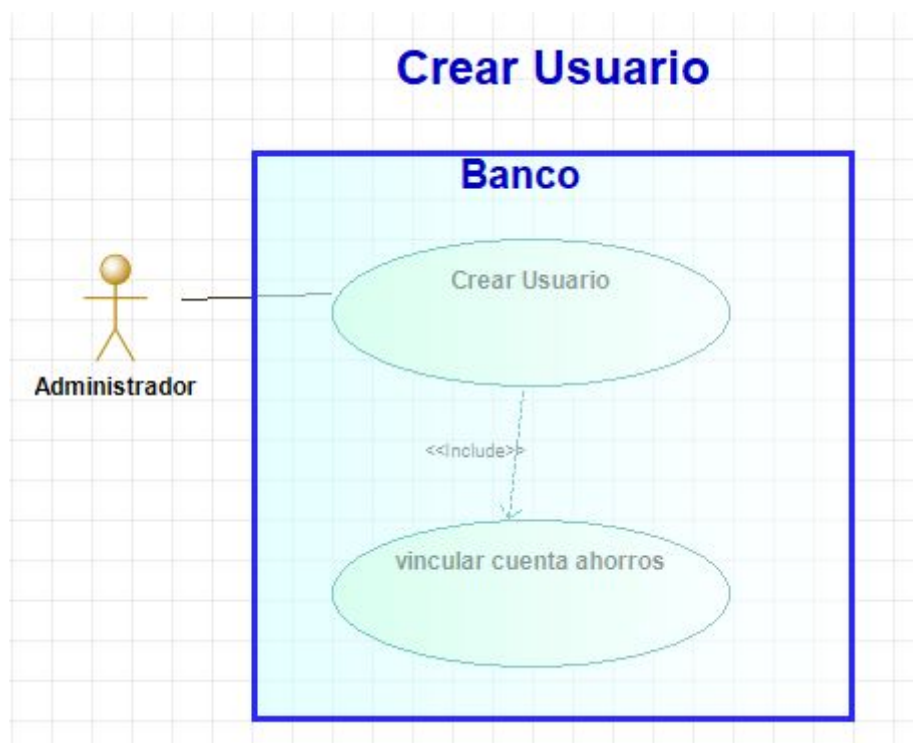
No.	Rol	Características/funcionalidades	Razón/Resultados
US0001	Como Administrador	Quiero modificar la cantidad de dinero en la cuenta de los usuarios	Con el fin de que un usuario pueda retirar dinero o transferir de su cuenta de ahorros.
US0002	Como Administrador	Quiero poder crear usuarios	Con el fin de que puedan usar una cuenta de ahorros
US0003	Como Administrador	Quiero poder modificar la información de los usuarios	Con el fin de mantenerla actualizada y segura
US0004	Como usuario	Quiero revisar mi saldo	Con el fin de saber cuánto dinero tengo en la cuenta de ahorros
US0005	Como usuario	Quiero ver mi historial de transacciones	Con el fin de saber cómo he manejado mi dinero
US0006	Como usuario	Quiero tener una identificación única asociada y una clave de acceso privada	Con el fin de garantizar la seguridad de mi cuenta

Caso de Uso	Modificar dinero en una cuenta de ahorros	No.:	UC0001
Descripción	Permitir el adicionar o sustraer un monto de una cuenta de ahorros de cualquier usuario.		
Actor(es)	Administrador: quien modificará el monto en la cuenta de ahorros del usuario en el sistema.		
Precondiciones	El usuario debe existir en el sistema.		

	El administrador debe tener la sesión iniciada.				
Flujo Normal	<ol style="list-style-type: none"> 1. El administrador elige al usuario.. 2. El administrador ejecuta la acción de adicionar o sustraer dinero del mismo. 3. El sistema verifica que el monto final del usuario sea válido. 4. El sistema actualiza la información del monto. 				
Flujo Alternativo	<ol style="list-style-type: none"> 1. El administrador elige al usuario.. 2. El administrador ejecuta la acción de adicionar o sustraer dinero del mismo. 3. El sistema verifica que el monto final del usuario sea válido. 4. El sistema notifica al administrador que hay un problema con la operación solicitada. 				
Autor(es):	hpintom	Fecha:	05.05.2019	Versión:	1.0

Caso de Uso	Crear Usuario	No.:	UC0002
Descripción	Crear un usuario con una contraseña, un identificador y asociarle una cuenta de ahorros		
Actor(es)	Administrador: quien creara el usuario en el sistema		
Precondiciones	El Administrador debe tener la cédula del usuario a crear		
Flujo Normal	<ol style="list-style-type: none"> 1. El administrador ingresa la Cédula 2. El administrador crea una contraseña al usuario de seis caracteres numericos. 3. El sistema le asocia un número de cuenta de ahorros 4. El sistema verifica que el número de la cuenta de ahorros sea diferente al de la cédula 5. El sistema verifica que la contraseña creada sea válida 6. El sistema informa al administrador de la creación exitosa del usuario 		

Flujo Alterno	<ol style="list-style-type: none"> 1. El administrador ingresa la Cédula 2. El administrador crea una contraseña al usuario 3. El sistema le asocia un número de cuenta de ahorros 4. El sistema verifica que el número de la cuenta de ahorros sea diferente al de la cédula. 5. El sistema verifica que la contraseña creada sea válida 6. El sistema verifica si el usuario no existe 7. El sistema informa al administrador que el usuario ya existe y termina el proceso 				
Autor(es):	mbarragani, hpintom	Fecha:	05.05.2019	Versión:	1.0

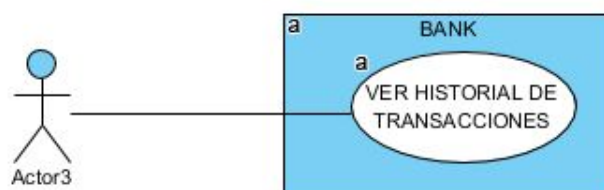


Caso de Uso	Modificar Usuario	No.:	UC0003
Descripción	Modificar la información de un usuario		
Actor(es)	Administrador: quien modificara la información del usuario en el sistema		
Precondiciones	El Administrador debe tener la id del usuario a modificar		
Flujo Normal	<ol style="list-style-type: none"> 1. El administrador ingresa el id del usuario. 2. El administrador selecciona qué información modificara 3. El administrador rellena el campo con la nueva información 4. El sistema verifica que la información ingresada sea válida 5. El sistema informa al administrador de una actualización de información exitosa 		

Flujo Alterno	<ol style="list-style-type: none"> 1. El administrador ingresa la id 2. El administrador selecciona qué información modificara 3. El administrador rellena el campo con la nueva información 4. El sistema verifica que la información ingresada sea válida 5. El sistema informa al administrador que hay un error en la información modificada 				
Autor(es):	mbarragani	Fecha:	05.05.2019	Versión:	1.0

Caso de Uso	Revisar mi saldo				No.:	UC0004
Descripción	Revisar la cantidad actual de dinero que tiene el usuario					
Actor(es)	Usuario 1: usuario que tiene el dinero y quiere ver cuánto es.					
Pre-condiciones	El usuario 1 debe poseer una cuenta de ahorros en el banco. El usuario 1 debe estar registrado en el sistema. El usuario 1 debe estar autenticado en el sistema.					
Flujo Normal	1. El usuario 1 puede visualizar su saldo que aparece en pantalla					
Flujo Alterno						
Autor(es):	jcristanchor	Fecha:	05.05.2019	Versión:	1.0	

VER HISTORIAL DE TRANSACCIONES



Caso de Uso	Ver historial de transacciones	No.:	UC0005
Descripción	ver las transacciones realizadas por el usuario		
Actor(es)	Usuario 1: usuario que podra ver su historial		
Pre-condiciones	El usuario 1 debe poseer una cuenta de ahorros en el banco. El usuario 1 debe estar registrado en el sistema. El usuario 1 debe estar autenticado en el sistema.		

Flujo Normal	<ol style="list-style-type: none"> 1. El usuario 1 hace llamada al historial 2. El sistema verifica que se haya realizado al menos una transacción 3. El sistema trae el historial de transacciones , incluyendo detalles como el destinatario de la transacción y el monto de la misma así como fecha y hora en la que se realizó 				
Flujo Alterno	<ol style="list-style-type: none"> 1. El usuario 1 hace llamada al historial 2. El sistema verifica que se haya realizado al menos una transacción 3. El sistema muestra que no ha existido ninguna transacción 				
Autor(es):	javergarav	Fecha:	30.04.2019	Versión:	1.0

Requerimientos no funcionales:

- El sistema de software debe asegurarse que el número de la cuenta de ahorros del usuario sea diferente al número de su cédula
- El sistema de software debe tener un administrador, entiéndase por qué debe tener una contraseña por defecto
- El sistema de software debe asegurar que el administrador no sea un usuario
- El sistema de software debe tener como colores principales el azul y el blanco
- El sistema de software debe estar disponible entre las 6 de la mañana a las 10 de la noche de lunes a viernes
- El sistema de software debe poder soportar la entrada de múltiples usuarios de forma concurrente

- El sistema de software debe almacenar la información encriptada en una base de datos
- El sistema de software debe asegurar que el usuario solo tenga una cuenta de ahorros asociada.
- El sistema de software debe usar un algoritmo asimétrico para cifrar la contraseña que se almacena en la base de datos.
- El sistema de software debe garantizar que un usuario tenga dinero suficiente para realizar una transacción
- El sistema de software debe notificar al usuario si tuvo una transacción exitosa o le falta dinero para realizar la transacción deseada
- El sistema de software debe darle un id y un número de cuenta de ahorros a cada usuario
- El sistema de software debe asegurar que el saldo mínimo es 0.
- Se debe desarrollar una aplicación nativa android.
- La aplicación móvil debe ser programada usando java.
- La base de datos debe ser relacional.

Diagrams

pedro | May 8, 2019

