

**FACULTAD DE INGENIERÍA**

Escuela Académico Profesional de Ingeniería de Sistemas e Informática

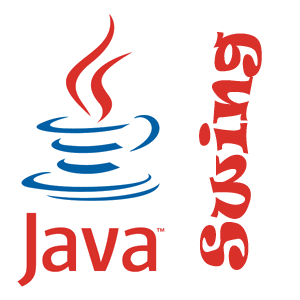
**PRODUCTO ACADÉMICO 03**

INFORME

**Programación visual y con eventos**

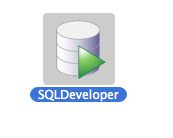
**Conexión a base de datos. API JDBC. Consulta SQL**

**Manipulación de Base de Datos. CRUD a tablas**

PROGRAMACIÓN ORIENTADA A OBJETOS

NRC 13967



**INTEGRANTES – GRUPO 12**:

* Chero Sojo José Casimiro
* Escobar Arcaya Yuri Alexander
* Mercado Guerrero Edward David

PERÚ

2021

**ÍNDICE**

[**INTRODUCCIÓN**](#_heading=h.30j0zll) **2**

[**ENUNCIADO**](#_heading=h.gridig7i3uof) **3**

[**PROGRAMA**](#_heading=h.3k6x3nn2zqdc) **6**

**Diseño de Interfaces de Usuario 6**

**Diseño de diagrama de clases 13**

**Diseño de base de datos 14**

**Construcción** **15**

**PRUEBAS 18**

[**CONCLUSIONES**](#_heading=h.92gna1b9pim0) **20**

[**RECOMENDACIONES**](#_heading=h.2jxsxqh) **20**

[**REFERENCIAS**](#_heading=h.z337ya) **20**

**INTRODUCCIÓN**

El siguiente trabajo es un informe de documentación del desarrollo de un software el cual simula a un cajero ATM, siendo la segunda versión con la implementación de conexión a una base de datos.

El informe ha sido dividido en 5 partes, las cuales consta de: enunciado, programa, pruebas correspondientes, conclusiones y bibliografía utilizada para la realización del presente producto académico 3.

El enunciado comprende los procesos y lineamientos que se debe llevar a cabo para realizar el cajero automático, aquí se define todas las pantallas que se debe diseñar para que el cliente pueda interactuar con la máquina, y permitiéndole realizar las operaciones comunes de un cajero automático.

El apartado de programa consta de 4 subtemas, dentro de los cuales se encuentra: En primer lugar, diseño de Interfaces de Usuario, donde se presentará la imagen del producto terminado, en segundo lugar se realizará el diagrama de clases el cual permitirá trazar la estructura del sistema, en se este será realizado con el software NetBeans 8.2, con la herramienta UML Diagrams Project, en tercera lugar el diseño de base de datos, aqui se presentará las tabla de base de datos, el cual está estructurado por filas y columnas agrupando los datos que se usará en el software a realizar, y finalmente la construcción, el cual es el script utilizado para realizar el trabajo encargado.

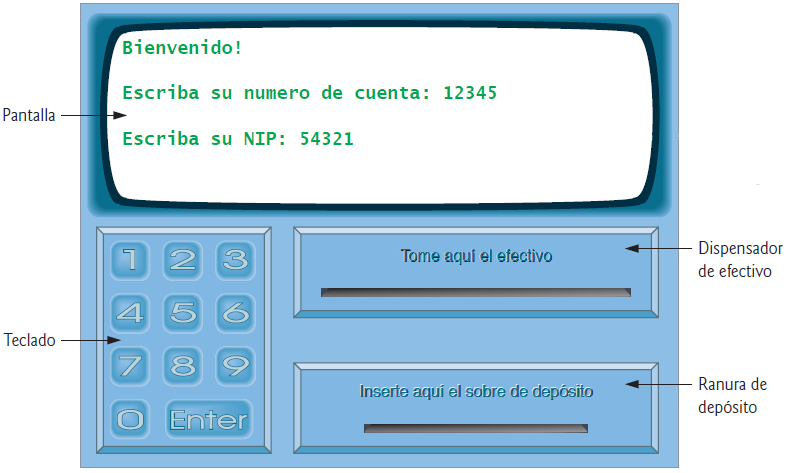
En pruebas se mostrará screenshots del software en funcionamiento, demostrando algunas funcionalidades que debe tener según el enunciado establecido.

Finalmente, el equipo dará a conocer las conclusiones y las referencias utilizadas para elaborar el presente informe.

**ENUNCIADO**

Desarrollar una **segunda versión del software** que simula un ATM.

La interfaz de usuario de cajero automático contiene los siguientes componentes: una pantalla que muestra mensajes al usuario, un teclado que recibe datos numéricos de entrada del usuario, un dispensador de efectivo que dispensa efectivo al usuario, y una ranura de depósito que recibe dinero del usuario.



Esta segunda versión deberá implementar la siguiente funcionalidad:

Cuando el Cliente se acerca al ATM se visualiza:

1. La pantalla muestra un mensaje de bienvenida y pide al usuario que introduzca un número de cuenta.

2. El usuario introduce un número de cuenta de cinco dígitos, mediante el uso del teclado.

3. En la pantalla aparece un mensaje, en el que se pide al usuario que introduzca su NIP (número de identificación personal) asociado con el número de cuenta especificado.

4. El usuario introduce un NIP de cinco dígitos mediante el teclado numérico.

5. Si el usuario introduce un número de cuenta válido y el NIP correcto para esa cuenta, la pantalla muestra el menú principal:

1. Ver mi saldo

2. Retirar efectivo

3. Depositar dinero

4. Salir

Después el usuario elegirá si desea realizar una transacción (oprimiendo 1, 2 o 3) o salir del sistema (oprimiendo 4).

Si el usuario selecciona la opción 1 para solicitar su saldo, la pantalla mostrará el saldo de esa cuenta bancaria. Para ello, el ATM deberá obtener el saldo de una base de datos.

Si usuario elige la opción 2 para hacer un retiro:

1. La pantalla muestra un menú que contiene montos de retiro estándar: S/ 20 (opción 1), S/ 40 (opción 2), S/ 60 (opción 3), S/ 100 (opción 4) y S/ 200 (opción 5). El menú también contiene una opción que permite al usuario cancelar la transacción.

2. El usuario introduce la selección del menú mediante el teclado numérico. Si el monto a retirar elegido es mayor que el saldo de la cuenta del usuario, la pantalla muestra un mensaje indicando esta situación y pide al usuario que seleccione un monto más pequeño. Entonces el ATM regresa al paso 1.

3. Si el monto a retirar elegido es menor o igual que el saldo de la cuenta del usuario (es decir, un monto de retiro aceptable), el ATM procede al paso 4. Si el usuario opta por cancelar la transacción (opción 6), el ATM muestra el menú principal y espera la entrada del usuario.

4. Si el dispensador contiene suficiente efectivo para satisfacer la solicitud, el ATM procede al paso 5. En caso contrario, la pantalla muestra un mensaje indicando el problema y pide al usuario que seleccione un monto de retiro más pequeño.

Después el ATM regresa al paso 1.

5. El ATM carga el monto de retiro al saldo de la cuenta del usuario en la base de datos del banco (es decir, resta el monto de retiro al saldo de la cuenta del usuario).

6. El dispensador de efectivo entrega el monto deseado de dinero al usuario.

Si usuario elige la opción 3 para hacer un depósito:

1. La pantalla muestra un mensaje que pide al usuario que introduzca un monto de depósito o que escriba 0 (cero) para cancelar la transacción.

2. El usuario introduce un monto de depósito o 0 mediante el teclado numérico.

3. Si el usuario especifica un monto a depositar, el ATM procede al paso 4. Si elige cancelar la transacción (escribiendo 0), el ATM muestra el menú principal y espera la entrada del usuario.

4. La pantalla muestra un mensaje indicando al usuario que introduzca un depósito en la ranura para depósitos.

5. Si la ranura de depósitos recibe el dinero dentro de un plazo de tiempo no mayor a 2 minutos, el ATM abona el monto del depósito al saldo de la cuenta del usuario en la base de datos (es decir, suma el monto del depósito al saldo de la cuenta del usuario).

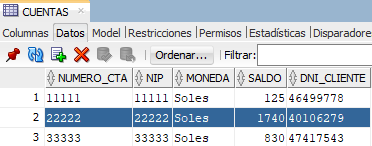
Una vez que el sistema ejecuta una transacción en forma exitosa, debe volver a mostrar el menú principal para que el usuario pueda realizar transacciones adicionales. Si el usuario elige salir del sistema, la pantalla debe mostrar un mensaje de agradecimiento y después el mensaje de bienvenida para el siguiente usuario.

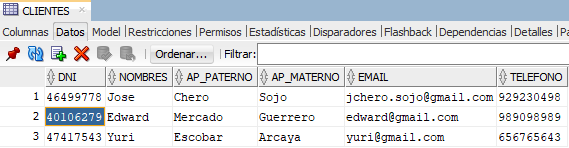
**PROGRAMA**

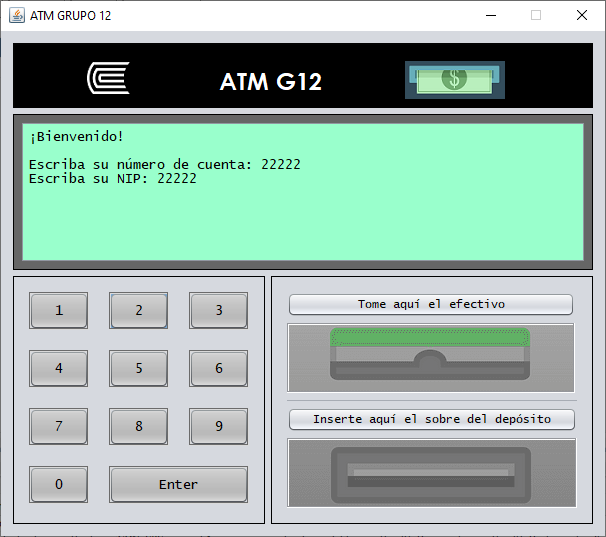
**Diseño de Interfaces de Usuario**

La pantalla de inicio de bienvenida y acceso al ATM donde se le pide al usuario validar sus credenciales de su cuenta bancaria introduciendo a través del teclado numérico su número de cuenta y NIP (número de identificación personal) de cinco dígitos cada uno. Las credenciales son validadas con registros almacenados en una base de datos.

En la figura 1 se muestra la validación de acceso al ATM con el número de cuenta = 22222 y NIP = 22222 interactuando 2 tablas : CUENTAS y CLIENTES.

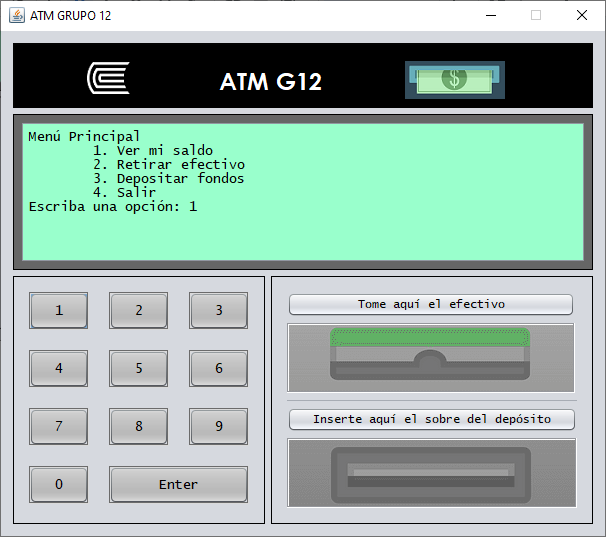




****

*Figura 1. Interfaz de usuario del cajero ATM*

En la figura 2 se muestra el Menú Principal con las 4 opciones principales [1] Ver mi saldo, [2]Retirar efectivo, [3]Depositar fondos y [4]Salir.

****

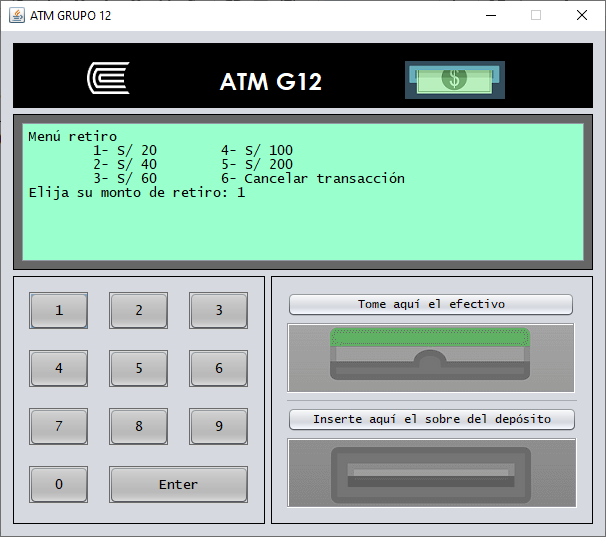
*Figura 2. Menú principal en la interfaz del cajero ATM*

En la figura 3 se selecciona [1]Ver mi saldo y se muestra en pantalla el Saldo Actual = S/ 1740.0 (de la tabla CUENTAS) correspondiente al Cliente con DNI = 40106279 (de la tabla CLIENTES).

****

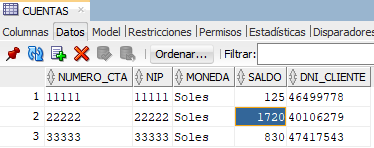
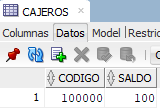
*Figura 3. Menú principal – mostrando “el saldo actual”*

En la figura 4 se muestra el Menú Retiro previamente al haber seleccionado [2]Retirar efectivo del Menú Principal. Se escoge retirar un monto predefinido por S/20 ingresando la opción [1- S/20]

****

*Figura 4. Menú retiro en la interfaz del cajero ATM*

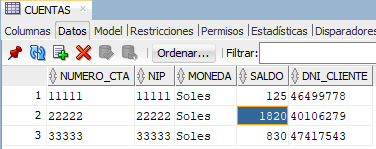
En la figura 5 se muestra el mensaje “Retire su efectivo”, lo cual indica que se ha concretado la transacción, para ello interactúa las tablas: CAJEROS (donde se verifica que exista dinero suficiente para atender operaciones de la transacción de retiro) y CUENTAS (donde se observa un nuevo saldo en la cuenta del usuario).



****

*Figura 5. Mensaje “Retire su efectivo” en la interfaz del cajero ATM*

En la figura 6 se muestra al usuario en Pantalla para que ingrese desde el teclado numérico el monto a depositar o digitar cero “0” para cancelar la transacción previamente al haber seleccionado [3]Depositar fondos del Menú Principal. Se ingresa un monto de S/100 para ser depositado a la cuenta bancaria del usuario identificado con el DNI = 40106279 y Número de Cuenta = 22222. Se efectúa la interacción de las tablas CUENTAS (donde se observa un nuevo saldo en la cuenta del usuario).

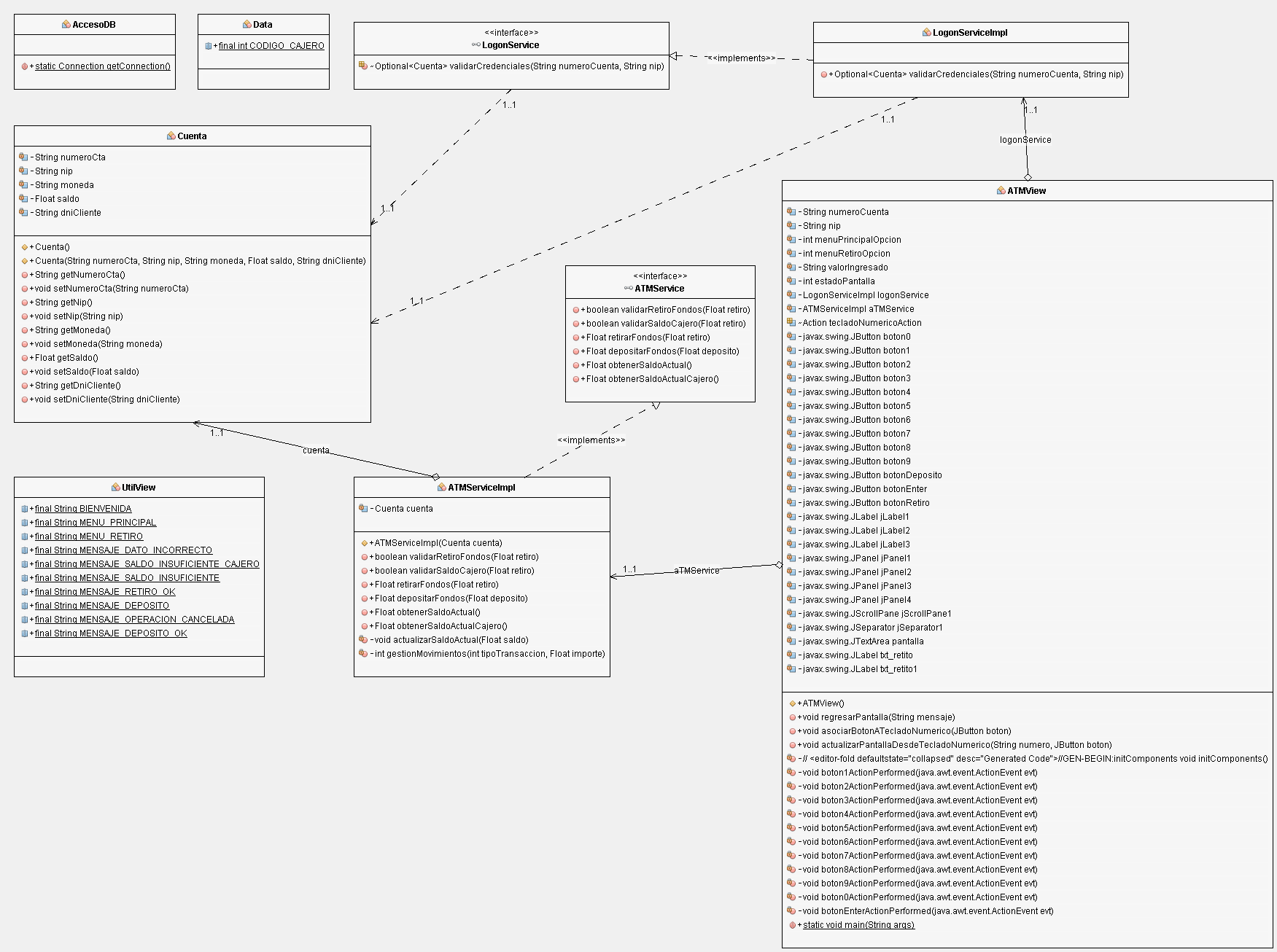


****

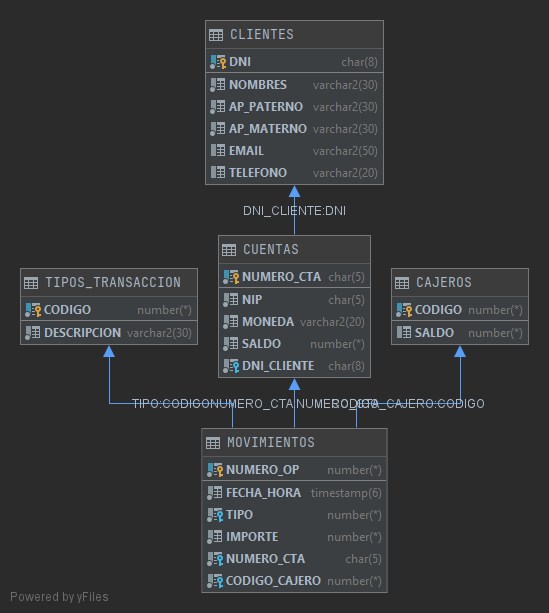
*Figura 6. Ingrese el monto a depositar de del proceso “Depositar fondos” (opción 3)*

**Diseño de diagrama de clases**

El diagrama de clases del proyecto se ha realizado con el software NetBeans 8.2, con la herramienta UML Diagrams Project.

****

**Diseño de base de datos**

****

**Construcción**

-- =============================================

-- CREACIÓN DEL USUARIO

-- =============================================

DECLARE

N INT;

COMMAND VARCHAR2(200);

BEGIN

COMMAND := 'DROP USER USER12 CASCADE';

SELECT COUNT(\*) INTO N

FROM DBA\_USERS

WHERE USERNAME = 'USER12';

IF ( N = 1 ) THEN

EXECUTE IMMEDIATE COMMAND;

END IF;

END;

/

ALTER SESSION SET "\_ORACLE\_SCRIPT"=true;

CREATE USER USER12 IDENTIFIED BY USER12;

GRANT CONNECT, RESOURCE TO USER12;

ALTER USER USER12 QUOTA UNLIMITED ON USERS;

GRANT CREATE VIEW TO USER12;

GRANT ALTER ANY INDEX TO "USER12";

GRANT CREATE ANY SEQUENCE TO "USER12";

GRANT CREATE ANY TRIGGER TO "USER12";

GRANT CREATE TRIGGER TO USER12;

GRANT ALTER ANY SEQUENCE TO "USER12";

-- =============================================

-- CONECTARSE A LA APLICACIÓN

-- =============================================

CONNECT USER12/USER12

create table CLIENTES

(

DNI CHAR(8) not null

constraint CLIENTES\_PK

primary key,

NOMBRES VARCHAR2(30) not null,

AP\_PATERNO VARCHAR2(30) not null,

AP\_MATERNO VARCHAR2(30) not null,

EMAIL VARCHAR2(50),

TELEFONO VARCHAR2(20)

)

/

create table CUENTAS

(

NUMERO\_CTA CHAR(5) not null

constraint CUENTAS\_PK

primary key,

NIP CHAR(5) not null,

MONEDA VARCHAR2(20) not null,

SALDO NUMBER not null,

DNI\_CLIENTE CHAR(8) not null

constraint CUENTAS\_CLIENTES\_FK

references CLIENTES

)

/

create table TIPOS\_TRANSACCION

(

CODIGO NUMBER not null

constraint TIPOS\_TRANSACCION\_PK

primary key,

DESCRIPCION VARCHAR2(30) not null

)

/

create table CAJEROS

(

CODIGO NUMBER not null,

SALDO NUMBER

)

/

create unique index CAJEROS\_CODIGO\_UINDEX

on CAJEROS (CODIGO)

/

alter table CAJEROS

add constraint CAJEROS\_PK

primary key (CODIGO)

/

create sequence MOVIMIENTOS\_SEQ

/

create table MOVIMIENTOS

(

NUMERO\_OP NUMBER not null

constraint MOVIMIENTOS\_PK

primary key,

FECHA\_HORA TIMESTAMP(6) not null,

TIPO NUMBER not null

constraint MOVIMIENTOS\_TIPOS

references TIPOS\_TRANSACCION,

IMPORTE NUMBER not null,

NUMERO\_CTA CHAR(5) not null

constraint MOVIMIENTOS\_CUENTAS\_FK

references CUENTAS,

CODIGO\_CAJERO NUMBER not null

constraint MOVIMIENTOS\_CAJEROS\_CODIGO\_FK

references CAJEROS

)

/

-- Disparador para actualizar el saldo del cajero en base a las transacciones sobre la tabla moviemientos

create trigger ACTUALIZAR\_SALDO\_CAJERO

after insert

on MOVIMIENTOS

for each row

BEGIN

IF (:NEW.tipo = 1) THEN

UPDATE CAJEROS

SET SALDO = SALDO + :NEW.importe

WHERE CODIGO = :new.codigo\_cajero;

END IF;

IF (:NEW.tipo = 2) THEN

UPDATE CAJEROS

SET SALDO = SALDO - :NEW.importe

WHERE CODIGO = :new.codigo\_cajero;

end if;

END;

/

-- Inserts

INSERT INTO USER12.CAJEROS (CODIGO, SALDO) VALUES (100000, 10000);

INSERT INTO USER12.TIPOS\_TRANSACCION (CODIGO, DESCRIPCION) VALUES (1, 'Depósito');

INSERT INTO USER12.TIPOS\_TRANSACCION (CODIGO, DESCRIPCION) VALUES (2, 'Retiro');

INSERT INTO USER12.CLIENTES (DNI, NOMBRES, AP\_PATERNO, AP\_MATERNO, EMAIL, TELEFONO) VALUES ('46499778', 'José', 'Chero', 'Sojo', 'jchero.sojo@gmail.com', '929230498');

INSERT INTO USER12.CLIENTES (DNI, NOMBRES, AP\_PATERNO, AP\_MATERNO, EMAIL, TELEFONO) VALUES ('40106279', 'Edward', 'Mercado', 'Guerrero', 'edward@gmail.com', '989098989');

INSERT INTO USER12.CLIENTES (DNI, NOMBRES, AP\_PATERNO, AP\_MATERNO, EMAIL, TELEFONO) VALUES ('47417543', 'Yuri', 'Escobar', 'Arcaya', 'yuri@gmail.com', '656765643');

INSERT INTO USER12.CUENTAS (NUMERO\_CTA, NIP, MONEDA, SALDO, DNI\_CLIENTE) VALUES ('11111', '11111', 'Soles', 125, '46499778');

INSERT INTO USER12.CUENTAS (NUMERO\_CTA, NIP, MONEDA, SALDO, DNI\_CLIENTE) VALUES ('22222', '22222', 'Soles', 1740, '40106279');

INSERT INTO USER12.CUENTAS (NUMERO\_CTA, NIP, MONEDA, SALDO, DNI\_CLIENTE) VALUES ('33333', '33333', 'Soles', 830, '47417543');

COMMIT;

****

****

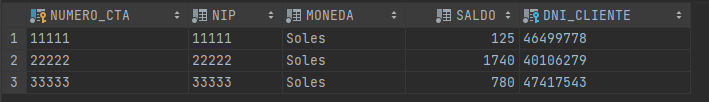
**PRUEBAS**

Se inicializan los datos de algunas tablas

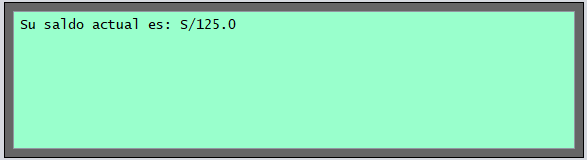
CAJEROS

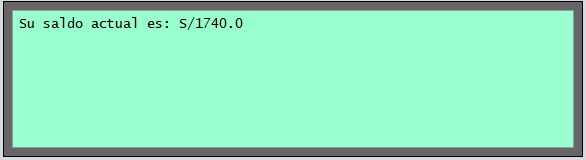
****

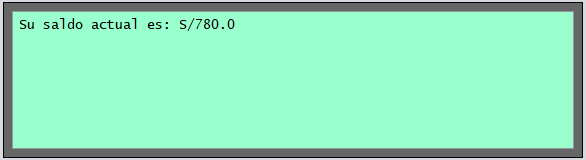
CUENTAS

****

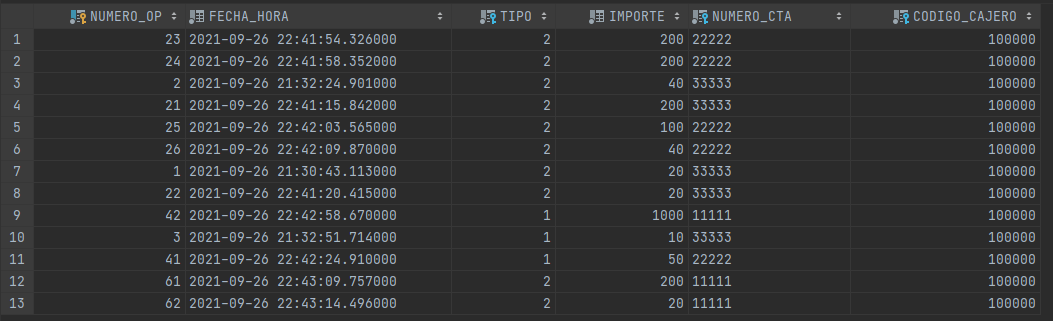
Al usar las distintas opciones de los menús de la aplicación se puede observar que se carga la información de la base de datos y que las operaciones hechas se reflejan en las tablas correspondientes (registro de movimientos por tipo de operación, actualización de saldo del cliente y actualización del saldo del cajero automático)

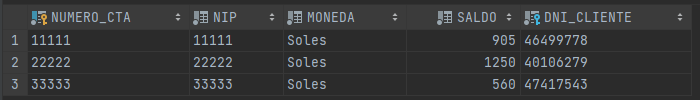






Los retiros y depósitos se pueden validar en la tabla movimientos (El tipo 1 corresponde a depósitos y el 2 a retiros) y cuentas





**CONCLUSIONES**

* Los tipos de bases de datos relacionales como Oracle, DB2 de IBM y SQL Server, forman la columna vertebral para el almacenamiento de datos y la gestión en la mayoría de organizaciones de hoy en día.
* Una de las ventajas de Oracle Database es que el software del servidor puede ejecutarse en multitud de sistemas operativos.
* La programación en capas nos permitió hacer el cambio de lógica sólo en la capa de servicios sin afectar demasiado a la interfaz de usuario.

**REFERENCIAS**

* (21 de abril de 2018). Programación ATS. 89. Programación en Java || Gráficas || Paneles (JPanel) - Creación de un panel. [Youtube] <https://youtu.be/_FhKqDj95g0>
* (03 de octubre de 2017). La Geekipedia De Ernesto. Curso Java desde cero #42 | Proyecto Final - Interfaz de Bienvenida. [Youtube] <https://www.youtube.com/watch?v=fMZWGDQZknQ>
* Torres, M. (09 de abril de 2014). Yo Soy Dev. Herramientas básicas para desarrollar GUI en Java. [Website] <https://yosoy.dev/herramientas-basicas-para-desarrollar-gui-en-java/>