**DOCUMENTACION DEL CODIGO DEL TALLER**

**Clase: User**

Clase con Getters y Setters los cuales nos servirán para recuperar información. Dicha clase es un “pojo” pues solo almacena la información del usuario como lo es el nombre de usuario, la contraseña del usuario y el dinero que este tiene en la cuenta. Cuenta también con un pojo como atributo, el cual se llama “person” el cual almacena información también.

**Clase: Person**

Clase con Getters y Setters los cuales nos servirán para recuperar información. Dicha clase es un pojo y solo contiene dos atributos los cuales almacenan información personal del usuario, como lo es el nombre de este y su número de documento.

**Clase: Register**

Clase con Getters y Setters los cuales nos servirán para recuperar información. Dicha clase es un “pojo” pues solo almacena la información de transacciones, pues esta guarda registros independientes de el dinero que se movió, el dinero actual de una persona y un mensaje detallado del tipo de movimiento que se hizo. Tambien cuenta con un método “toString” que nos sirve para

**Clase: ATMBussinessManager**

Clase en la cual inicialmente se declaran arreyList de tipo User(nombrado Users) y Register(nombrado temporalUserRegisters) junto con un objeto de tipo User (nombrado currentUser). Los cuales posteriormente son creados en el constructor llamado ATMBussinesManager.

Posteriormente se emplea el desarrollo del objetivo de esta clase que es permitir administrar los datos del usuario, desde la creación de este, el inicio de sesión, y los diferentes movimientos de su dinero, todo esto desarrollado en los siguientes métodos:

***signIn :*** método de tipo booleano el cual permite hacer una búsqueda por medio del log in hecho por el usuario el cual valida el usuario y la contraseña.

***createUser :*** método el cual permite hacer la creación de un nuevo usuario.

***Withdraw :*** método de tipo booleano el cual valida los retiros que quiera hacer el usuario teniendo en cuenta el dinero disponible que estos tengan en el momento de la transacción.

***consignMoney :*** método de tipo booleano el cual valida el usuario por medio de la cedula de ciudadanía, y el sí es posible hacer la transacción del dinero con su saldo disponible, en caso de ser posible realiza la transacción y devuelve el nuevo valor del saldo, y en caso de no ser posible realizar la transacción, devuelve al usuario un rechazo en su solicitud.

***moneyLoans :*** método que permite al usuario hacer una crédito/préstamo/avance de dinero, el cual se posteriormente se ve reflejado en su saldo total.

***consultMoney :*** método el cual le permite al usuario realizar la consulta del dinero que tiene disponible en ese momento.

***changePassword :*** método el cual le permite al usuario realizar un cambio en la contraseña de acceso a la cuenta.

***GetTransactionSummary :*** método el cual le retorna al usuario el resumen de los registros generados al hacer diferentes transacciones.

***addRegister :*** método el cual añade un nuevo registro luego de una transacción.

***SearchUser :*** método el cual permite hacer la búsqueda de un usuario mediante el número de su cedula de ciudadanía.

***searchUserPerLogin :*** método el cual permite el acceso a la cuenta mediante la validación del nombre de usuario y la contraseña.