**Proyecto de Curso – Secure Coding**

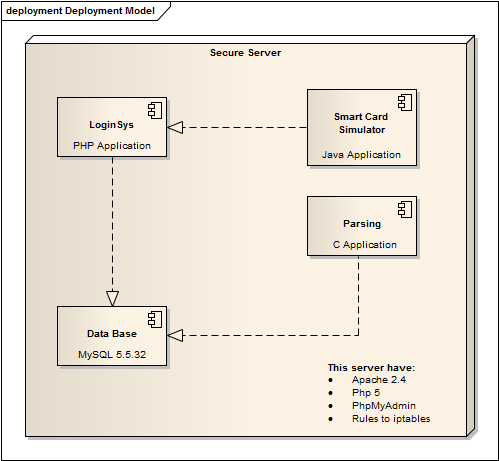
**Dr. Martín Ochoa**

**PROPÓSITO.**

Este archivo tiene como finalidad plasmar algunos detalles para el correcto funcionamiento de la maquina virtual que aloja la integración de las aplicaciones creadas.

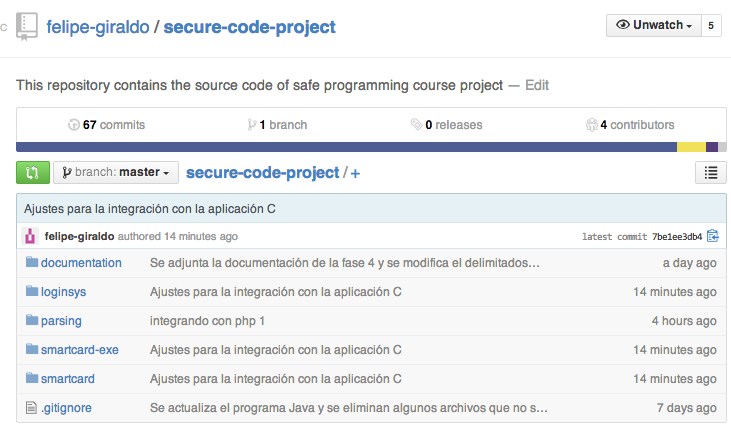
**INTRODUCCIÓN.**

La máquina virtual denominada Secure Code, contiene las herramientas y servicios para soportar la arquitectura planteada en el curso. A continuación se muestra una breve descripción de sus componentes principales:



* **LoginSys:** este componente corresponde a la aplicación creada en PHP, la cual contiene el frontend para realizar las operaciones transaccionales que hacen parte de los requerimientos.
* **SmartCard:** este componente corresponde a la aplicación creada en Java, la cual permite generar tokens fuera de línea, con los cuales es posible validar la realización de una transferencia.
* **Parsing:** este componente corresponde a la aplicación creada en C, la cual permite realizar transferencias en batch, realizando inserciones directas a al motor de base de datos, una vez corrobore las validaciones respectivas.
* **DataBase:** este componente es un servicio de almacenamiento de datos, montado sobre el motor de base de datos relacional MySQL 5.5.32.

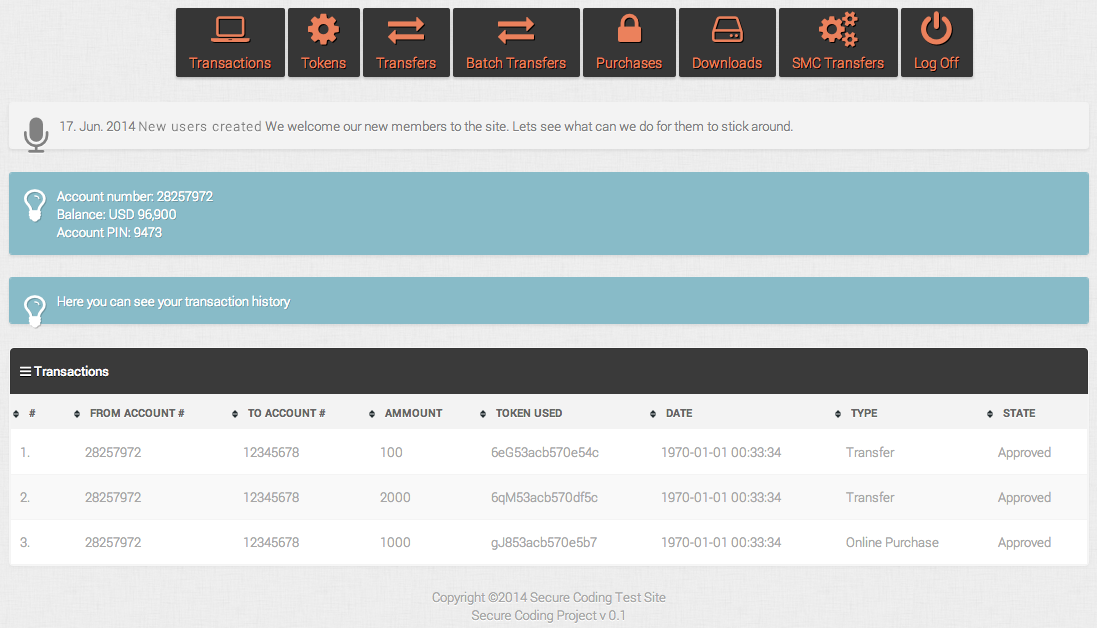
Adicionalmente es posible consultar el código fuente de todas las aplicaciones en el repositorio de GitHub en la url: <https://github.com/felipe-giraldo/secure-code-project>.



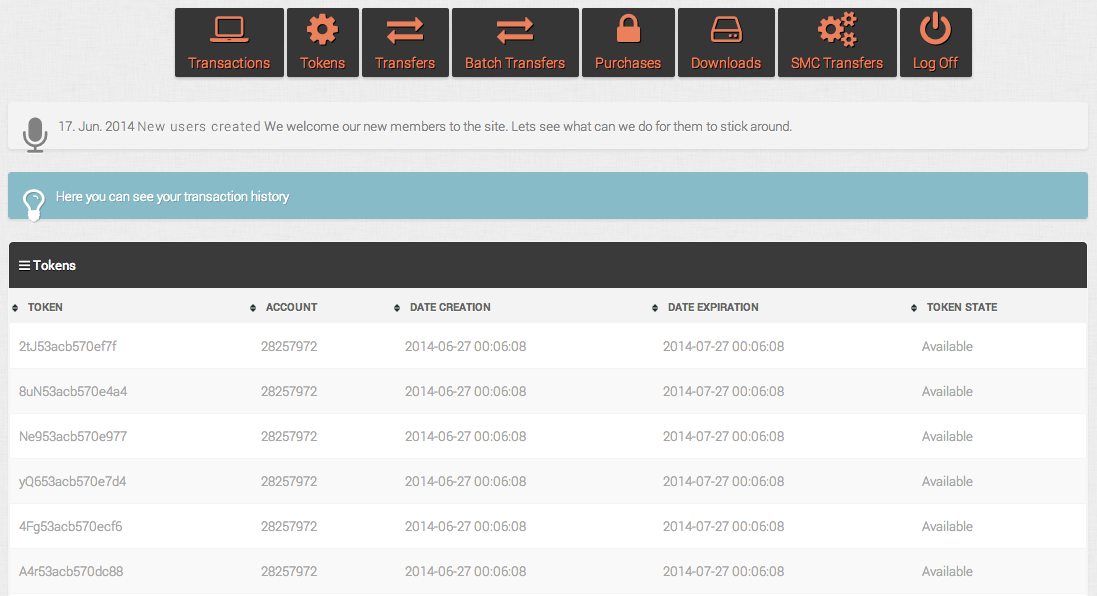
**FUNCIONALIDADES DE LA APLICACIÓN PHP.**

Dentro de las funcionalidades desarrolladas en la aplicación PHP, encontramos las siguientes:

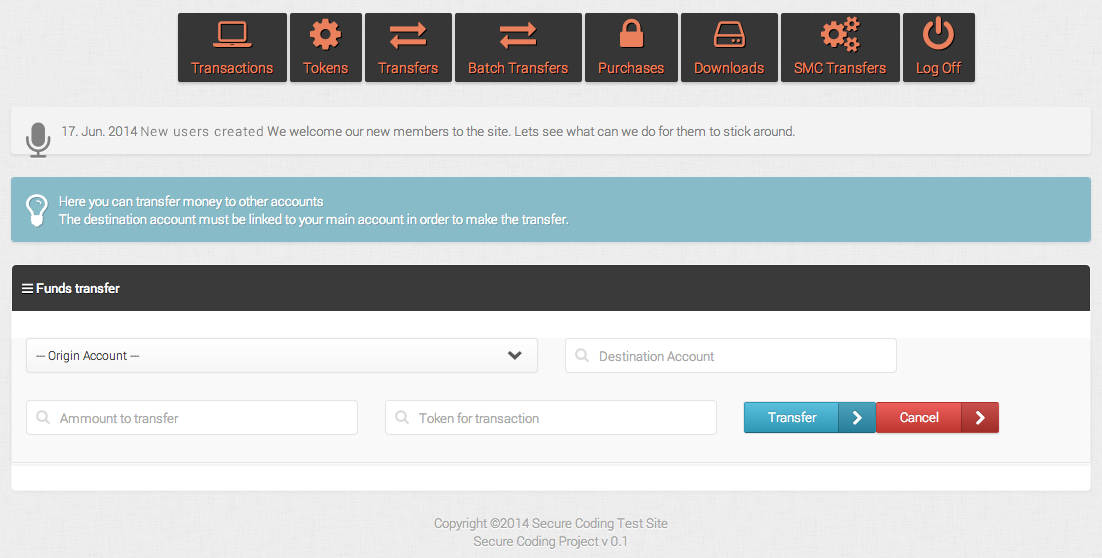
* **Registro de usuario:** esta funcionalidad permite registrar un usuario nuevo, el cual queda en estado pendiente por activación. Posteriormente un usuario con perfil de administrador dará de alta el nuevo usuario creado con el fin de que el sistema automáticamente genere los 100 tokens necesarios para realizar transacciones. Estos tokens serán incluidos en un archivo con extensión PDF cifrado, para enviarlos al correo electrónico suministrado por el usuario al momento de diligenciar los campos de registro.
* **Ingreso a la aplicación:** una vez se ha ingresado a la aplicación, el sistema muestra el perfil del usuario con opción de edición en caso de ser requerido.
* **Transactions:** en esta funcionalidad se muestra el histórico de transacciones que ha realizado el cliente que actualmente esta logueado en el sistema. Aquí también es posible ver el numero de la cuenta, el saldo de la cuenta y el PIN asociado a la misma.

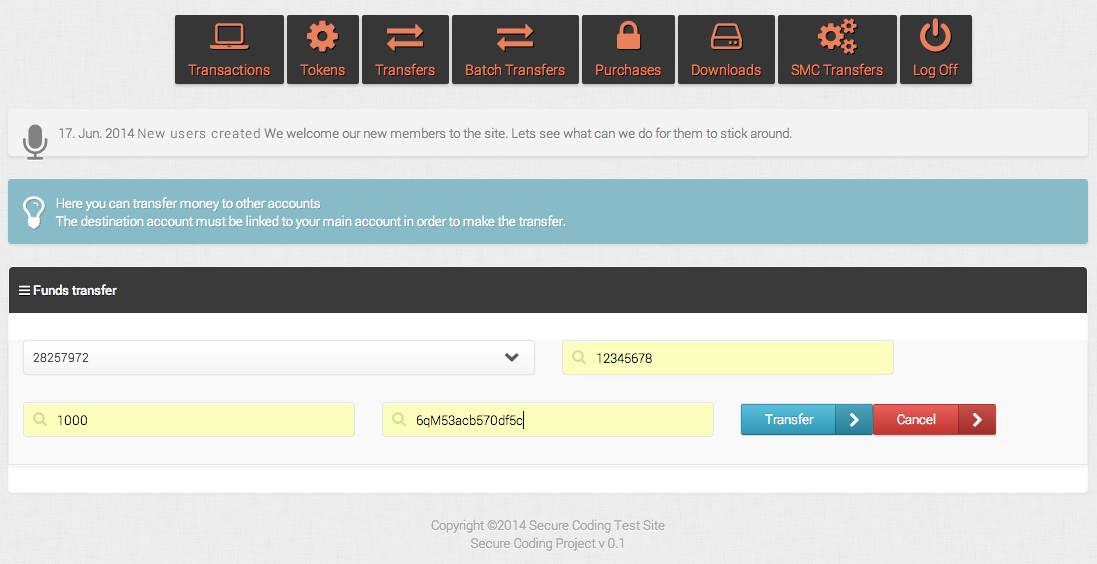


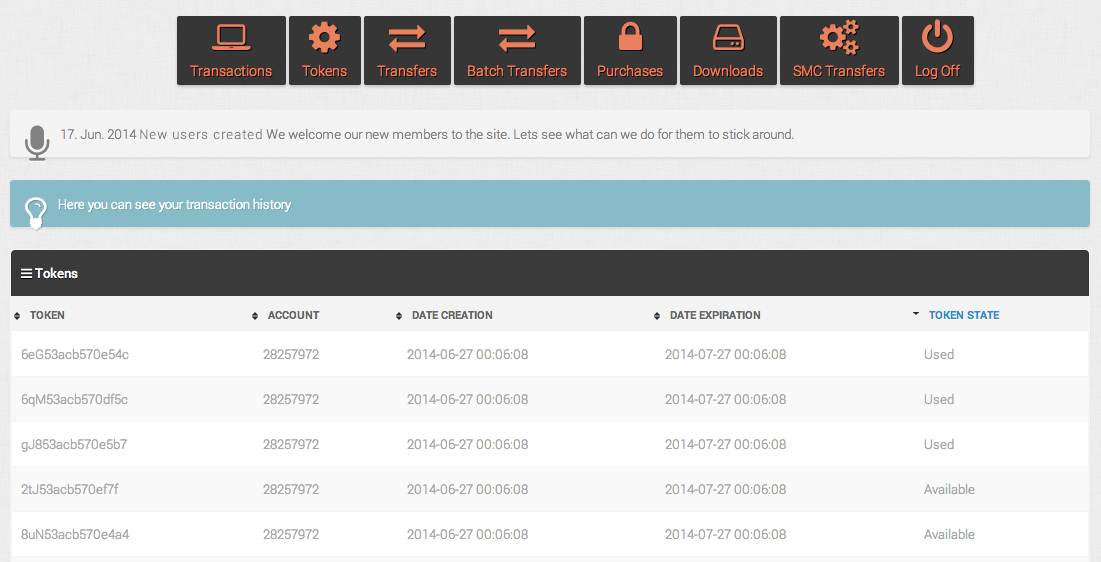
* **Tokens:** en esta funcionalidad se muestran los tokens que el sistema generó para el usuario, los cuales fueron enviados a través de correo electrónico en un archivo PDF cifrado.



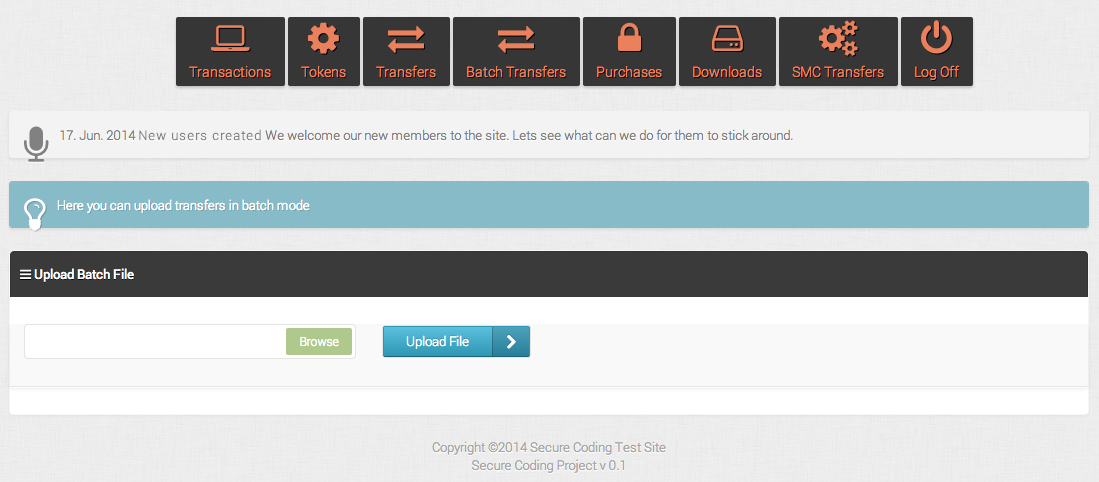
* **Transfers:** en esta funcionalidad se pueden realizar las transferencias entre cuentas. Como se muestra en el segunda imagen con datos diligenciados, es necesario seleccionar el numero de la cuenta origen (en este campo solo se muestran las cuentas asociadas al usuario logueado), ingresar el numero de la cuenta destino, ingresar el monto de la transferencia y por último ingresar uno de los tokens generados. Una vez es realizada la transferencia, el sistema marca el token como usado, tal como se muestra en tercera imagen.

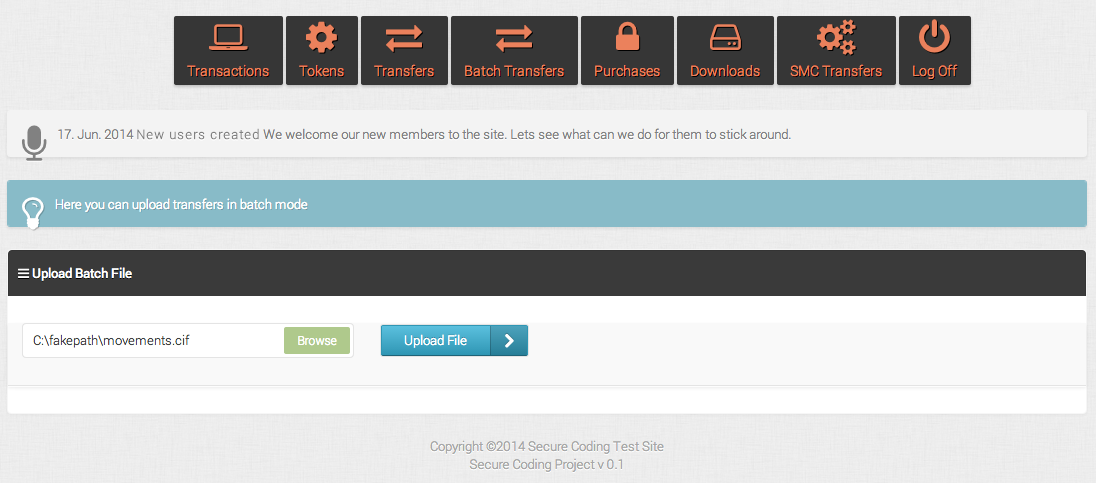




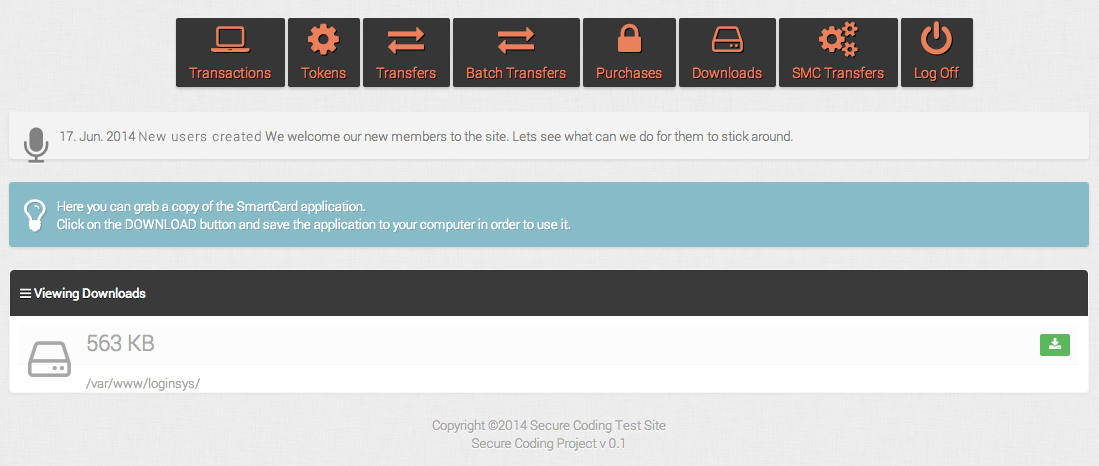


* **Batch Transfers:** esta funcionalidad permite realizar transacciones en batch, cargando al sistema un archivo con extensión .CIF, el cual debe estar cifrado con AES-128. Mas adelante será mostrada esta funcionalidad con mas detalle, en la aplicación Java. Posteriormente se invoca una rutina del sistema operativo desde PHP, la cual se encarga de pasar el archivo cifrado a la aplicación C, donde será descifrado y validado. Una vez cumpla los criterios de validación establecidos, se hará la inserción de las transacciones en el motor de base de datos.

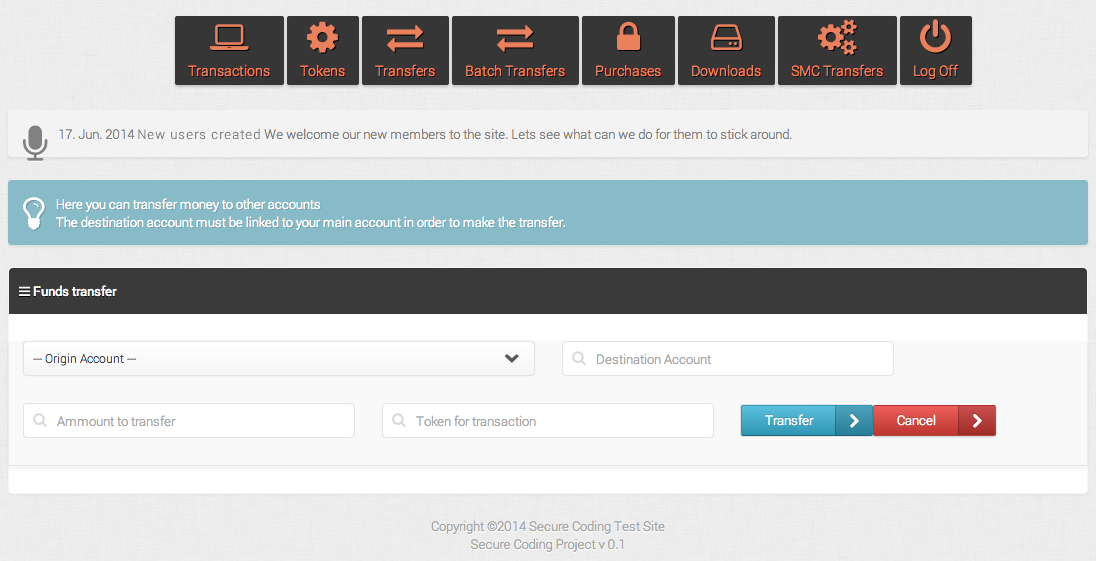


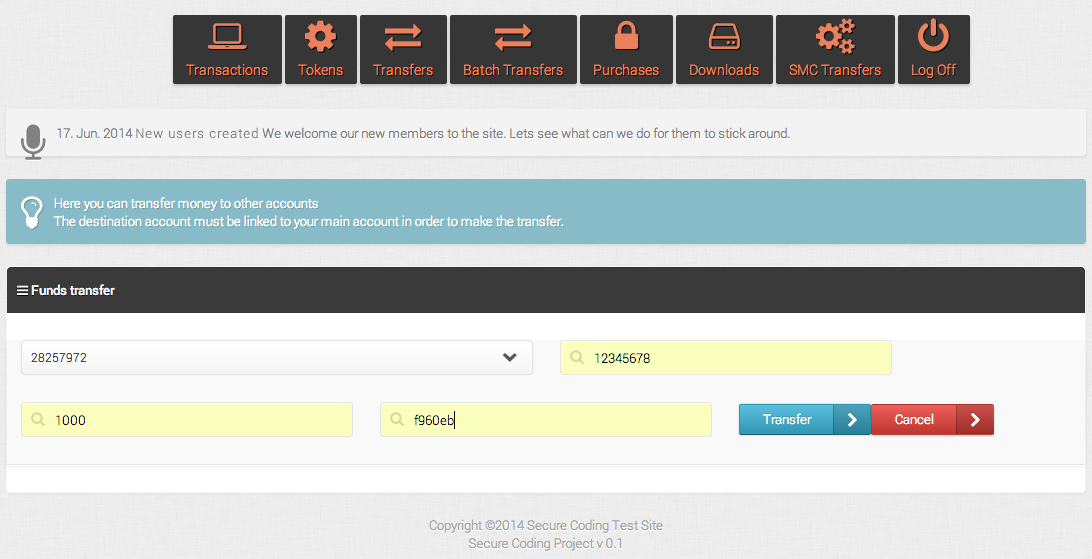


* **Purchases (compras):** esta funcionalidad permite realizar copras debitando de las cuenta del usuario. Funciona de forma similar a la funcionalidad de transacciones.
* **Downloads:** esta funcionalidad permite descargar la aplicación Java encargada de generar los tokens fuera de línea, además permite construir el archivo batch para realizar transacciones por lotes, según se explicó en la funcionalidad respectiva.



* **SMC Transfers:** esta funcionalidad permite realizar transferencias haciendo uso de los tokens generados con la aplicación Java, los cuales difieren totalmente de los tokens enviados a través de correo electrónico al momento de la activación de la cuenta del usuario. Su funcionamiento es similar a las transferencias, con la única salvedad de que el token es generado de forma diferente.

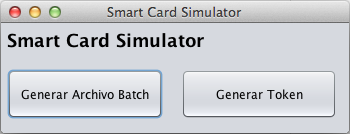


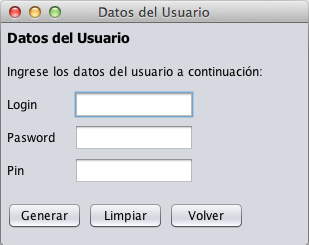


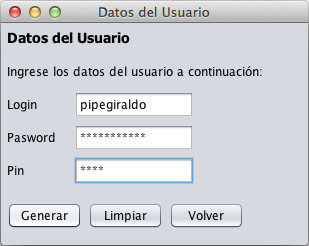
**FUNCIONALIDADES DE LA APLICACIÓN JAVA.**

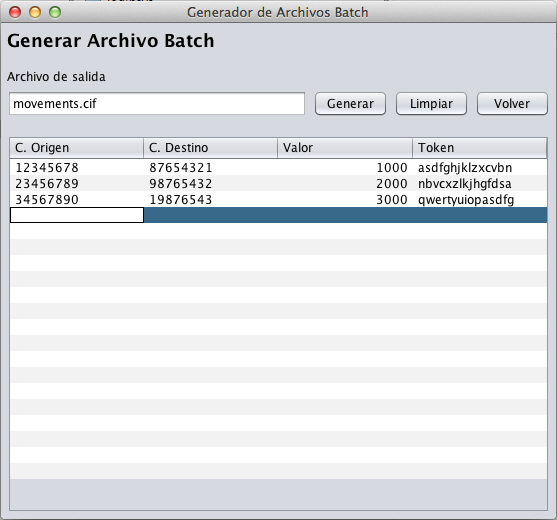
Dentro de las funcionalidades desarrolladas en la aplicación Java, encontramos las siguientes:

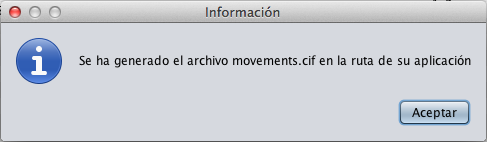
* **Generación de archivo para transacciones en batch:** esta funcionalidad permite generar un archivo cifrado usando AES-128 con los datos de las transacciones que se realizarán por lotes. Inicialmente se solicitan los datos de autenticación del usuario en la plataforma PHP, además del PIN enviado por correo electrónico al momento de crear la cuenta. Con estos datos será generada la clave para cifrar el archivo con AES. Posteriormente se ingresa a una interfaz con una tabla, la cual permite diligenciar los datos de las transferencias a realizar, según como se muestra en la segunda imagen a continuación. El archivo generado queda almacenado en la misma ruta de la aplicación Java.

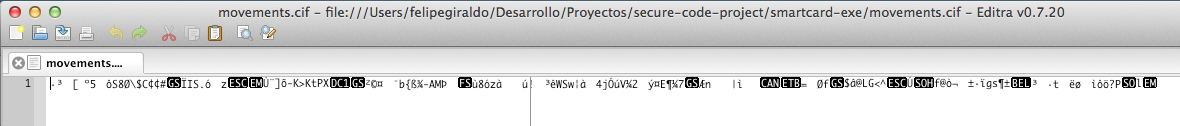












* **Generación de tokens:** esta funcionalidad permite generar un token para realizar transacciones en la funcionalidad SMC Transfers de la aplicación PHP. Es necesario tener a la mano el PIN enviado por correo electrónico al momento de registro, el valor a transferir, la cuenta origen y la cuenta destino. EL token generado se muestra en color rojo, tal como se puede apreciar en la segunda imagen.



