BANORTE

Anexo I – Información General Interredes

Diciembre 2019
Versión 2.3

Introducción

En este capítulo daremos una idea general con la finalidad de familiarizar al usuario con la API para el cobro mediante presencia de plástico en sus puntos de venta.

¿Qué es Interredes?

Este esquema surge de la necesidad de integrar más de un punto de venta en el sistema del comercio con el motor de pagos Banorte a fin de procesar transacciones de tarjetas de crédito y débito.

El concepto de Interredes permite la centralización de todas las transacciones de las sucursales en un sólo punto, o bien, la intercomunicación con cada uno de los puntos de venta en las diferentes sucursales.

Es posible integrar a la Interred cualquier dispositivo de lectura de banda magnética, PIN Pads, kioscos, terminales, lectores de tarjeta, etc.

Para realizar la lectura del plástico en el punto de venta, se requiere un dispositivo certificado en los estándares de la industria, como se describe a continuación.

¿Qué es un PIN Pad?

Un PIN Pad es un dispositivo electrónico que provee diversas funciones:

- Lectura de tarjetas con banda magnética.
- Lectura de tarjetas con chip que cumplan con estándares EMV.
- Lectura de tarjetas contactless que cumplan con estándares EMV.
- Implementación interna de estándares EMV.
- Captura y codificación de números de identificación personal (NIP).

Es una alternativa al pago de efectivo. Se utiliza con alguna aplicación para recibir pagos con tarjetas de débito, crédito o monedero electrónico.

La aplicación normalmente se ejecuta en alguna computadora e interactúa con el PIN Pad a través de comandos de software. Físicamente, éste se encuentra conectado a alguno de los puertos del equipo, donde se ejecuta la aplicación, generalmente puerto serial o USB.

El PIN Pad cuenta con los lectores necesarios, tanto de banda magnética, chip como contactless, para poder leer las tarjetas de los clientes en el momento en que la aplicación lo indique. También cuenta con una pequeña pantalla para poder desplegar textos enviados desde la aplicación. Finalmente, para aquellas tarjetas emitidas por instituciones que requieran validación del número de identificación personal del tarjetahabiente, el PIN Pad provee un pequeño teclado numérico por medio del cual el cliente puede autentificarse.

Además, el PIN Pad posee internamente una aplicación responsable de administrar los componentes de hardware presentes en el dispositivo, así como de implementar el protocolo de comunicación con la aplicación controladora, a fin de poder recibir e interpretar correctamente los comandos enviados por ésta, cuando se desea que el PIN Pad ejecute tal o cual función. Actualmente Banorte maneja los PIN Pads del fabricante VeriFone.

Seguridad

El PIN Pad encripta los datos sensibles, al momento de enviar una transacción hacia Banorte. Esto con la finalidad de que la información viaje de forma segura hacia el emisor para su autorización. Los datos que se encriptan al ser enviados hacia Banorte son los siguientes:

- Track 1
- Track 2
- Código de seguridad (CVC2, CVV2, 4DBC)

Cada dispositivo recibirá una llave única de cifrado, por lo tanto es necesario realizar la carga del dispositivo de forma individual. Cada dispositivo estará enlazado a una afiliación. Si se requiere mover el dispositivo para ser utilizado en una afiliación diferente, será necesario realizar nuevamente una carga de la llave de encriptación. Los PIN Pad de Banorte que cuentan con la aplicación segura, no podrán transaccionar hasta que realicen la inyección de la llave de cifrado.

La inyección de la llave de encriptación es requerida para el funcionamiento del dispositivo. Cuando al comercio se le haga entrega de un dispositivo, únicamente recibirá los comandos para la inyección de llave. Mientras no se realice la inyección de llave el PIN Pad no realizará ningún comando o transacción. Si el comercio cree que la seguridad de la información que procesa el dispositivo es comprometida, podrá solicitar nuevamente la carga de llave. Para esto es necesario solicitar el cambio de la llave de encriptación a su ejecutivo.

Interfaz de programación (API)

Pensando en simplificar el trabajo a nuestros clientes e integradores, Banorte ha desarrollado una interfaz de programación (también conocida como API, siglas del inglés Application Program Interface) que internamente se hace cargo de la complejidad técnica del protocolo de comunicación con el PIN Pad, y provee a cambio, una serie de funciones de muy fácil uso que la aplicación puede ejecutar desde su propio código, a fin de instruir al PIN Pad a realizar determina función, en un punto preciso de la operación.

Además, la función de lectura de tarjetas, la interfaz de programación desarrollada por Banorte, permite al cliente que así lo desee, efectuar la autorización en línea de la transacción, entregándole como salida la información necesaria para su aplicación de punto de venta, tal como el resultado (si la operación fue aceptada o declinada), el código de autorización (en caso de haber sido aceptada), el nombre del tarjetahabiente, y toda la información adicional que normalmente se requiere para este tipo de aplicaciones.

Obsérvese que integrado de esta forma, el PIN Pad se convierte en una especie de terminal de punto de venta tradicional, pero con la ventaja de que es la aplicación del propio cliente quien lo controla.

Esta serie de funciones está disponible para diversos lenguajes y plataformas de programación, y se describen en forma detallada dentro del presente manual.

Modos de operación

La interfaz de programación Banorte PIN Pads posee dos modalidades de operación:

1. PROCESAR TRANSACCIÓN - Lectura de tarjeta con procesamiento de transacción incluido (Envió a Banorte de datos de operación y respuesta de la transacción)
2. LEER, ENVIAR Y NOTIFICAR - Sólo lectura de tarjeta para procesamiento interno de la información.

Bajo la primera modalidad, al momento de efectuar una transacción, el PIN Pad no sólo leerá la tarjeta del cliente, sino que enviará la información de entrada suministrada por la aplicación, así como la información recopilada de la tarjeta hacia el procesador central de pagos de Banorte con el fin de efectuar la autorización de la misma. Una vez recibida la respuesta del procesador central, la información será entregada a la aplicación del cliente en los parámetros de salida de la misma llamada.

Esta modalidad se recomienda para aquellos clientes que por la arquitectura de su aplicación, pueden enviar directamente la transacción a Banorte sin requerir de la intervención de un tercer componente de software.

Sin embargo, para aquellos clientes que por necesidades de arquitectura requieren hacer llegar la transacción a Banorte por un medio alterno (por ejemplo, clientes con un servidor de transacciones ISO 8583, los cuales requieren sólo la información leída de la tarjeta para formar el mensaje de la transacción), la interfaz de programación provee la segunda modalidad, en la que únicamente se regresan a la aplicación del cliente los datos de la tarjeta leída sin enviar ningún requerimiento de autorización al procesador central de pagos de Banorte. Se asume, por supuesto, que el cliente se hará cargo de hacer llegar la transacción al banco a través de su medio alterno.

Esta segunda modalidad puede ser también de utilidad para aquellos clientes que deseen implantar algún esquema de tarjeta propia, en la que únicamente requieren de la función de lectura de la misma. Por supuesto, las tarjetas deberán cumplir con los estándares mínimos necesarios para poder ser leídas por los dispositivos PIN Pads.

El usar la interfaz de programación en una modalidad u otra, es simplemente cuestión de hacer llamadas a diferentes funciones incluidas dentro del propio API; no hay necesidad de hacer uso de configuraciones o instalaciones especiales. La aplicación puede decidir en cualquier momento, de acuerdo a la operación, cuándo usar el API, de un modo o de otro.

Instrucciones Básicas para operar con la API

En esta sección se detallarán las operaciones básicas de la API ejemplificando el proceso completo desde la lectura de la tarjeta hasta el envío de la operación a Banorte.

Inicialización y liberación del PIN Pad

Los PIN Pads proporcionados por Banorte utilizan una interfaz virtual de bajo nivel basada en puerto serial. Esto significa que aún cuando el puerto físico utilizado sea un USB, el controlador instalado para el dispositivo crea un puerto serial virtual, mediante el cual la API puede interactuar con el PIN Pad.

Banorte provee a sus clientes el controlador necesario para los PIN Pads; los cuales deberán solicitarse vía correo electrónico a las direcciones de contacto descritas en la última sección de este manual, especificando el modelo de dispositivo que le fue entregado.

Como en el caso de cualquier otro puerto serial, ya sea virtual o físico, es necesario garantizar que el puerto designado para usarse con el PIN Pad se encuentre disponible para uso de la API.

La interfaz de programación provee dos llamadas, denominadas ***prepareDevice*** y ***releaseDevice***, las cuales deberán ser ejecutadas, la primera al inicio de operaciones para realizar la inicialización del puerto serial asignado, y la segunda al término de la ejecución de la aplicación para garantizar que el puerto quede libre para un eventual uso de otras aplicaciones.

No se recomienda llamar estas dos operaciones por transacción, ya que se estaría haciendo la inicialización y la liberación del puerto innecesariamente múltiples veces. Sin embargo es de mucha importancia que dentro del manejo de excepciones se establezca el procedimiento para liberar el dispositivo.

Inicio y fin de transacciones

La API provee dos llamadas, denominadas ***startTransaction*** y ***endTransaction***, las cuales deberán ser ejecutadas por la aplicación al inicio y al término de una transacción, respectivamente. Internamente, estas llamadas se utilizan para realizar los pasos necesarios en el PIN Pad de modo que éste sepa cuándo está por iniciar una transacción y cuándo se ha completado.

Estas llamadas deberán hacerse por cada transacción efectuada con el PIN Pad, ya que es necesario informar a éste sobre los eventos de inicio y fin, para que la aplicación residente en el dispositivo se sincronice adecuadamente.

Es importante comentar que ambas llamadas son necesarias aún cuando la API se utilice en modalidad de sólo lectura de tarjetas. En ese caso, la operación de lectura de tarjeta (y eventual notificación del resultado de la transacción) se considera como una transacción, aún cuando ésta no se haga llegar a Banorte por medio de la API.

Carga de llaves de Encriptación

Para la seguridad de los comercios y de los tarjetahabientes, la información viaja de forma encriptada desde el dispositivo hacia Banorte. Para esto, es necesario realizar la carga de la llave de encriptación en el dispositivo. Dependiendo la forma en que se realice la conexión entre el punto de venta y Banorte, será la llamada a efectuarse a la API para la carga de la llave. Se recomienda al desarrollador generar un módulo o una aplicación por separado para la carga de llaves del dispositivo.

Modo de Operación: Procesar Transacción

Dando como hecho que el punto de venta tiene conexión directa a Internet y a Banorte, la aplicación del cliente deberá solicitar la información del PIN Pad mediante la llamada ***getInformation***. Esta llamada como parámetros de salida entregará la información del dispositivo que incluirá el número de serie y la versión de la aplicación instalada en el dispositivo.

Con la información obtenida de la llamada ***getInformation***, se realizará la llamada ***updateMasterKey***, la cual irá a Banorte y solicitará la llave de encriptación y la inyectará en el dispositivo. Después de que la llave se inyecte, el dispositivo podrá realizar transacciones financieras.

Modo de Operación: Leer, Enviar y Notificar Transacción

Cuando el punto de venta no tiene acceso directo a la nube y por ende no tiene acceso directo a Banorte, sino que se conecta a un servidor antes de enviar la transacción a Banorte, la aplicación tiene llamadas específicas para estos casos. La aplicación de punto de venta del cliente deberá solicitar la información del dispositivo mediante la llamada ***getInformation***. Este comando entregará la información del dispositivo que incluye el número de serie del dispositivo y la versión de la aplicación instalada en el dispositivo.

Para obtener la llave de encriptación Banorte solicita un selector del dispositivo. Este selector se obtiene realizando la llamada ***getSelector***. Esta llamada retornará el selector necesario para solicitar la llave de encriptación a Banorte. Ya con el selector se enviará el comando **GET_KEY** utilizando la llamada ***sendTransaction***. Banorte retornará la llave de encriptación que se inyectará en el dispositivo mediante la llamada ***loadMasterKey***. Después de la inyección de la llave el dispositivo podrá realizar transacciones.

Procesamiento de transacciones

La API provee la funcionalidad para enviar a procesar las transacciones al emisor mediante Payworks. Existe dos maneras de procesar las transacciones: **Proceso integrado de transacciones** y **Proceso independiente de transacciones** o por módulos.

Modo de operación Procesar Transacción

La API de Banorte está diseñado para trabajar en una modalidad que permita en una sola llamada hacer llegar la transacción al procesador central de pagos de Banorte para su

autorización. Esta es la modalidad generalmente recomendada a los clientes, a menos que por motivos de su arquitectura requieran hacer llegar la transacción a Banorte por otro medio.

Para utilizar la API en este modo, la aplicación del cliente deberá ejecutar la llamada ***processTransaction***. En ese momento, el PIN Pad se preparará para recibir la tarjeta del cliente (ya sea de banda magnética o de chip), la leerá, generará la información necesaria para enviar la información a Banorte, recibirá la respuesta y entregará el resultado a la aplicación del cliente en los parámetros de salida de esta misma llamada.

La llamada espera ciertos parámetros de entrada que indican, entre otras cosas, el tipo de transacción, el importe de la misma, etc.

Cuando la API intente enviar la transacción hacia Banorte, se requerirá que las aplicaciones que utilice tengan disponible la conexión necesaria al momento de ejecutar la llamada. La conexión a Banorte puede estar disponible en diferentes modos: internet, línea dedicada, VPN, etc., y normalmente es gestionada por el cliente al momento de tramitar su afiliación.

Al momento de ejecutar esta llamada, el cliente observará que en la pantalla del PIN Pad se despliega la leyenda "Inserte tarjeta". En este momento el cliente podrá deslizar su tarjeta (si es de banda magnética) o insertarla (si es de chip). En el caso de una tarjeta de chip, ésta **NO DEBERÁ SER RETIRADA** hasta que la transacción se complete, ya que de lo contrario ésta no podrá concluirse satisfactoriamente.

La llamada a ***processTransaction*** entregará como parámetros de salida a la aplicación del cliente la información necesaria para que ésta pueda proceder a hacer sus registros en la base de datos, efectuar impresión de ticket, etc.

Modo de Operación Leer, Enviar y Notificar Transacción

En el caso de aplicaciones que requieran hacerse cargo de la transacción directamente, la API provee dos llamadas: ***readCard*** y ***notifyResult***.

Como su nombre lo indica, la primera llamada se utiliza exclusivamente para realizar el proceso de lectura de una tarjeta del cliente, sea de banda, chip o contactless, y entregará como parámetros de salida la información necesaria para que la aplicación del cliente, o un tercer componente de software, pueda construir el mensaje de la transacción y enviarlo a Banorte.

En el caso de tarjetas con banda magnética, la API internamente detecta cuando una tarjeta leída es bancaria, y en ese caso entrega como parámetros de salida los datos desglosados del nombre del tarjetahabiente, el número de tarjeta y la fecha de expiración. Si la tarjeta no es reconocida como bancaria (Visa, MasterCard o American Express), la llamada devolverá únicamente el contenido de los tracks 1 y 2 que se hubieren reconocido en la lectura.

Es importante mencionar en este punto, que el mecanismo de lector del PIN Pad está diseñado para leer tarjetas que cumplen con los estándares **ISO 7810**, **7811** y **7813** que reglamentan el diseño de las tarjetas, las características del material magnético de la banda, el formato de grabación de los tracks, el contenido de los mismos, etc. **No se garantiza la lectura de tarjetas de Banda que no se adhieran a dichos estándares.**

Para el caso de tarjetas con chip y contactless, el equipo podrá leer sin problema todas aquellas tarjetas que cumplan con los estándares de EMVCo publicados sobre la materia. **No se garantiza la lectura de tarjetas de chip que no cumplan con dichos estándares**. Cuando la lectura se ha realizado sin problema, la llamada devolverá igualmente a la aplicación del cliente la información desglosada del nombre del tarjetahabiente, el número de tarjeta y la fecha de expiración para facilitar el proceso de la transacción en el punto de venta.

En el caso de tarjetas que posean tanto chip como banda magnética, la aplicación del PIN Pad considerará al chip como el medio de lectura preferido, ya que se considera más seguro. Si la tarjeta tiene Chip, Contactlees y Banda Magnética, la aplicación del PIN Pad considera al Chip y Contactless como lecturas preferidas. Por tanto, si el cliente intenta deslizar por el lector de banda una tarjeta de chip, la lectura será rechazada y se invitará al cliente a que utilice el lector de chip.

Como parte de la información de salida suministrada por esta llamada, la aplicación del cliente podrá determinar cuál fue el medio por el que se leyó la tarjeta, a saber: banda magnética, chip, o contactless. Esta información requiere ser enviada al banco emisor, por lo que si el cliente se está haciendo cargo de la transacción, deberá informar a Banorte sobre el tipo de lectura realizada, lo cual se hará dependiendo del medio que se utilice para hacer llegar la transacción a Banorte. Por ejemplo, aquellos clientes que cuentan con un servidor de transacciones ISO 8583 deberán generar el campo 22 (POS Entry Mode) con los valores adecuados dependiendo de la salida que entregue esta llamada.

Para el caso de las tarjetas de chip, la llamada a ***readCard*** devolverá igualmente la información de Tags EMV que la especificación define como requisito para ser enviados al banco emisor para su validación. Esta es una cadena ASCII de dígitos hexadecimales que la aplicación del cliente deberá hacer llegar a Banorte dependiendo del medio que utilice para tal fin. Por ejemplo, si se tiene un servidor de transacciones ISO 8583, la cadena entregada por esta llamada deberá ser convertida a la secuencia binaria respectiva y colocada en el campo 55 del mensaje ISO 8583 como lo señala la especificación Terminal To Host.

Es importante comentar que el procesador central de pagos de Banorte puede rechazar aquellas transacciones que no cumplan con la información necesaria requerida por los estándares de transacciones financieras vigentes en el momento.

La llamada a ***notifyResult*** se requiere **ÚNICAMENTE PARA TARJETAS CON CHIP O CONTACTLESS CHIP**, ya que la especificación EMV obliga a la aplicación del cliente a informar al dispositivo sobre el resultado de la transacción.

Esta llamada deberá hacerse una vez que ya se tiene el resultado de la transacción, informando de ésta a la API a través de los parámetros de entrada que dicha llamada espera. Obsérvese que esta llamada deberá hacerse independientemente del resultado de la transacción; es decir, deberá ejecutarse para transacciones aprobadas, declinadas, e inclusive para aquellas en las que no se recibió respuesta por parte del emisor (siempre asumiendo que se trata de transacciones con lectura de chip, o contactless chip). Al momento de ejecutarse la llamada a ***notifyResult***, el PIN Pad completa los registros necesarios en el chip de la tarjeta, y despliega el mensaje respectivo invitando al tarjetahabiente o al cliente a retirar su tarjeta, una vez que el

proceso de la transacción ha concluido. Esta es la razón por lo que la tarjeta no debe retirarse anticipadamente.

El usuario podrá apreciar en este punto la diferencia entre las dos modalidades de operación que provee la API: Mientras que en la modalidad con procesar transacción se encarga de suministrar todos los datos requeridos a Banorte y completar la transacción con una sola llamada, en la segunda modalidad, con Leer, Enviar y Notificar transacción es la aplicación del cliente la que con base a la información de salida proporcionada por la llamada a ***readCard***, deberá ser responsable de formar el mensaje correspondiente, hacer llegar la transacción a Banorte, recibir la respuesta, procesar el resultado de la respuesta y notificar la API del resultado final por medio de una llamada a ***notifyResult***.

Despliegue de mensajes en pantalla

Para aquellos clientes que deseen personalizar los mensajes desplegados en la pantalla del PIN Pad, la API incluye una llamada ***desplegarTexto***, que permite pasar como parámetro de entrada el texto deseado e internamente se encarga de que éste sea desplegado en el dispositivo.

Esta llamada puede ejecutarse antes de procesar una transacción, o bien una vez que ésta ha sido completada.

Selección de idioma

La API permite especificar el idioma que se usará para los nombres de variables, valores de éstas y textos de los posibles mensajes de error. Los idiomas actualmente soportados son español e inglés. La forma específica en que el idioma se selecciona depende del lenguaje de programación.

Igualmente, cada una de las llamadas de la API tiene un nombre en inglés y en español. Por ejemplo, existe ***displayText*** equivalente a ***desplegarTexto***.

Arquitectura

En este capítulo, se proporciona un panorama general de la construcción de la interfaz de programación para PIN Pad de Banorte, así como diversos aspectos técnicos que ayudarán a los consultores especializados a facilitarles el trabajo de integrar la API a las aplicaciones.

Estructura

La API de Banorte está estructurado por niveles, para facilitar así la futura integración de nuevos modelos y/o marcas de dispositivos, evitando así trasladar al cliente las complejidades técnicas internas requeridas por el esquema de comunicación de cada dispositivo.

Es importante enfatizar que la aplicación del cliente no tiene que involucrarse en los detalles de bajo nivel, y que la arquitectura anterior tiene por objetivo hacer lo más transparente posible a los clientes un posible cambio de equipo a futuro.

Físicamente, la API se entrega como una **biblioteca de carga dinámica** (DLL para usuarios Windows, SO para usuarios UNIX). Las llamadas disponibles en la API se encuentran convenientemente exportadas para facilitar su localización desde el código de la aplicación. Dependiendo del lenguaje de programación seleccionado, es posible que se entregue algún componente adicional de software (por ejemplo, para usuarios Java se entrega un archivo .jar que expone la interfaz de programación como un conjunto de clases que son fácilmente utilizables por usuarios de dicho lenguaje).

Parámetros de entrada y salida

Tratando de seleccionar un esquema que resultara fácil de utilizar para los clientes, considerando la diversidad de lenguajes de programación y plataformas existentes, se adoptó un esquema de comunicación para las llamadas la API basado en parámetros, tanto de entrada como de salida. Por medio de ciertos parámetros de entrada la aplicación puede proporcionar a la llamada específica la API la información que requiere para hacer su trabajo, y es también por medio de parámetros de salida que la aplicación del cliente puede recibir retroalimentación de la API, sobre el resultado de una llamada en particular.

Cada parámetro, tanto de entrada como de salida, está identificado por un nombre que es representativo de su uso. Este nombre puede estar en inglés o en español, dependiendo del lenguaje seleccionado al momento de iniciar la operación con la API.

Los valores de cada parámetro invariablemente serán de tipo texto, para evitar las complejidades que implica la implementación de otros tipos de lenguajes de programación y/o plataformas. Será responsabilidad de la aplicación del cliente realizar las conversiones necesarias entre su propio código y el API cuando maneje algún parámetro con un tipo de dato diferente a texto.

Cada llamada específica de la API señala sus parámetros de entrada y/o de salida. La estructura de datos específica para pasarlos depende del lenguaje de programación; en el presente documento puede hallarse la implementación para cada lenguaje soportado.

Validación de PIN Offline

PIN Offline

Actualmente existen en el mercado tarjetas de crédito y/o débito que solicitan al tarjetahabiente insertar el PIN Offline en el dispositivo al momento de realizar una transacción. El dispositivo proporcionado por Banorte está listo para solicitar el PIN Offline en caso de que la tarjeta así lo requiera.

La DLL de Banorte, al momento de realizar la lectura de la tarjeta, retorna un indicador que señala si el PIN Offline de la tarjeta fue correctamente validado. Al ejecutar la llamada a la DLL ***readCard*** (***leerTarjeta***) y la llamada ***processTransaction*** (***procesarTransaccion***), la DLL retorna como

parámetro de salida el parámetro **PIN_ENTRY** (**CAPTURA_NIP**). Este parámetro indica si se realizó la validación del PIN Offline de la tarjeta. Si se retorna un '1', indica que la verificación del PIN Offline fue satisfactoria. Si retorna un '0', indica que la verificación del PIN Offline no fue satisfactoria o que no se solicitó un PIN Offline al momento de realizar la transacción.

Si el PIN Offline fue verificado correctamente y la tarjeta no solicita la firma autógrafa del tarjetahabiente se deberá desplegar en el comprobante de compra el mensaje: **AUTORIZADO CON FIRMA ELECTRÓNICA**. Si el PIN Offline fue verificado correctamente y la tarjeta solicita la firma autógrafa del tarjetahabiente, se debe mostrar en el comprobante de compra el mensaje "**PIN VERIFICADO**" y el espacio respectivo para colocar la firma autógrafa de tarjetahabiente.

Firma del tarjetahabiente

El solicitar el PIN Offline en una transacción no indica que no se deberá solicitar la firma autógrafa de tarjetahabiente. La firma autógrafa del tarjetahabiente puede ser solicitada aún y cuando el tarjetahabiente insertó el PIN Offline de forma correcta.

Para saber si la tarjeta requiere que se solicite o no se solicite la firma autógrafa del tarjetahabiente, la aplicación de Punto de Venta del comercio deberá analizar el Cardholder Verification Method (CVM) Results (Tag 9F34). El byte 1 de este tag proporciona la información necesaria para conocer si solicitar o no solicitar la firma autógrafa del tarjetahabiente. Si el byte 1 del tag 9F 34 viene con valores de **03**, **05**, **1E**, **43**, **45** o **5E**, la aplicación de punto de venta del comercio deberá imprimir en el comprobante de venta la línea respectiva para que el tarjetahabiente firme. Para conocer cómo interpretar los Tags EMV ver el APÉNDICE E en la sección de ¡Error! No se encuentra el origen de la referencia.

Si el voucher muestra la línea para la firma del tarjetahabiente y el voucher no está firmado, no será válido para una controversia, por eso se recomienda imprimir la línea para la firma del tarjetahabiente únicamente cuando sea solicitado por la tarjeta. Si la tarjeta solicita el PIN Offline y además requiere que el tarjetahabiente firme el voucher, tanto el mensaje de **PIN VERIFICADO** como la firma autógrafa del tarjetahabiente deben de estar presentes en el voucher.

Glosario de términos

La Tabla 1 presenta una lista de términos que se consideran relevantes para la comprensión del presente documento.

Tabla 1. Glosario de términos.

Término	Descripción
API	Acrónimo de "Application Program Interface". Designa un componente de software que permite a una aplicación hacer uso de ciertos servicios implementados por un tercero, siguiendo las especificaciones de programación proporcionadas por el propio componente.
Archivo .JAR	Es un archivo comprimido utilizado comúnmente por aplicaciones Java. Internamente contiene la versión compilada de un conjunto de clases que son requeridas por la aplicación que hace uso del archivo.
Biblioteca de ejecución dinámica	Es un componente de software que provee ciertos servicios a una aplicación, y tiene la característica de que no está físicamente ligado al código de la aplicación, sino que se mantiene como una entidad independiente, y es activado por el sistema operativo cuando la aplicación hace uso por primera vez de él. De esta forma, el mismo código puede ser compartido por varias aplicaciones a la vez, en lugar de aparecer físicamente ligado a cada una.
Cashback	Operación realizada con tarjeta de crédito o débito en la que se presenta disposición de efectivo por parte del cliente en el punto de venta.
Chip	Es un componente electrónico integrado a ciertas tarjetas bancarias con la capacidad de almacenar datos y software. Se utiliza como un medio para reducir el nivel de fraude en el uso de tarjetas tanto de crédito como de débito.
Contactless Chip	Tecnologías de identificación por radiofrecuencia incorporadas en tarjetas de crédito o débito, tarjetas inteligentes, teléfonos móviles u otros dispositivos que permiten a los consumidores pagar una transacción acercando el dispositivo a un lector del terminal punto de venta, de tal forma que no es necesario leer la tarjeta de forma física a través de una ranura de lectura.
Clase	Es una estructura utilizada en lenguajes de programación orientada a objetos que define una plantilla con características a partir de la cual se hará la creación de objetos. Define atributos propios y servicios (métodos) que pueden ser utilizados por otros objetos.
Código de autorización	Es un número (típicamente 6 dígitos) que es retornado cuando una transacción de tarjeta bancaria es aprobada por el banco emisor.
Controlador	Es un componente de software de bajo nivel responsable de atender un dispositivo. Producido normalmente por los fabricantes de hardware; su estructura es dependiente por completo del sistema operativo.
Criptograma	Es un mensaje cifrado cuyo significado resulta ininteligible hasta que no es descifrado. Es utilizado por algunos bancos emisores para incrementar el nivel de seguridad en las transacciones con tarjetas de chip.
Declinada	Declinada indica que la transacción viajo hasta el Banco emisor y este fue quien rechazo la transacción.

Declinada EMV	La transacción fue enviada al Banco emisor, esta pudiera estar aprobada pero al momento de notificar al Chip y este realizó validaciones de prevención de fraudes y determinó que debe ser rechazada. En caso de utilizar las funciones Leer/Enviar/Notificar se deberá generar una Reversa (Reversal) de la transacción. Si utiliza la función procesarTransacción, ésta la generará de forma automática.
Declinada Offline	La transacción fue declinada por reglas de validación del CHIP sin ser enviada al Banco emisor.
DLL	Acrónimo del inglés "Dynamic Link Library". Véase Biblioteca de ejecución dinámica.
Driver	Véase Controlador.
EMV	Acrónimo de "Europay Mastercard Visa". Estándar de interoperabilidad internacional entre tarjetas con chip definido en conjunto por esas empresas para incrementar el nivel de seguridad en las transacciones.
EMVCo	Empresa creada en 1999 por Europay, Mastercard y Visa para regular y administrar la especificación EMV.
Excepción	Evento que tiene la característica de interrumpir el flujo normal de un programa, debido típicamente a un problema encontrado en un punto de ejecución. Disponible en varios lenguajes de programación; en algunos de ellos orientados a objetos estos eventos se modelan como clases que mantienen información específica sobre el error.
ISO	Acrónimo de "International Organization for Standardization". Es un organismo creado en 1947 para promover el desarrollo de normas internacionales de fabricación, comercio y comunicación para todas las ramas industriales a excepción de la eléctrica y la electrónica.
ISO 8583	Estándar definido por ISO que define especificaciones para el intercambio de transacciones de tipo financiero.
JDK	Acrónimo de "Java Developer Kit". Es un conjunto de herramientas normalmente instalado en equipos en donde se hace desarrollo Java.
JRE	Acrónimo de "Java Runtime Environment". Es la implementación de la máquina virtual Java que debe estar presente para poder ejecutar una aplicación desarrollada en dicha plataforma. Es responsable de interpretar los códigos de operación de la máquina virtual Java, convertirlos a código nativo de la plataforma donde se encuentra y ejecutarlos.
Método	En lenguajes de programación orientados a objetos, cada uno de los servicios (funciones) ofrecidos por una clase.
.NET Framework	Símil del JRE de Java pero exclusivo para plataforma Microsoft
PIN/NIP/No. de Identificación Personal	Secuencia de dígitos que se utiliza para autentificar la identidad de un cliente al hacer uso de un dispositivo electrónico bancario.
PIN Pad	Dispositivo con pantalla y teclado numérico integrados, que posee lectores de banda magnética y de chip. Utilizado para aceptar tarjetas de los clientes y enviar información al equipo que lo controla. Para aquellas tarjetas que así lo requieran, permite ingresar y/o cambiar el NIP del cliente.
Punto de venta	Con este término se designa comúnmente tanto al hardware como al software utilizado para registrar la venta de algún bien o servicio.
QPS	Quick Payment Service. Tipo de transacción en la que no es necesaria la firma autógrafa del tarjetahabiente en el comprobante de la transacción.

Tag EMV	Dentro de la especificación EMV, cada uno de los componentes que aportan un elemento de información sobre la transacción. Cada tag tiene un número que lo identifica de manera única y su contenido normalmente es una secuencia de bytes que es interpretado por el receptor del tag.
Track 1	Dentro del estándar de grabación de tarjetas bancarias de banda magnética, se refiere al primero de los tracks que deben estar presentes en la banda magnética. Normalmente tiene un formato predefinido que contiene información sobre el tarjetahabiente, la cuenta, la fecha de expiración, el código de servicio, etc.
Track 2	Dentro del estándar de grabación de tarjetas bancarias de banda magnética, se refiere al segundo de los tracks que deben estar presentes en la banda magnética. Normalmente tiene un formato predefinido que contiene información sobre la cuenta, la fecha de expiración y el código de servicio.
URL	Acrónimo de "Universal Resource Locator". Es una secuencia de caracteres que sigue un formato estándar, y que usa con la finalidad de encontrar recursos en una red (normalmente Internet), tales como documentos, imágenes, aplicaciones, etc.

Verificación de transacciones

Las transacciones pueden ser verificadas mediante la herramienta administrativa web que Banorte provee. La dirección URL de la herramienta es **https://via.banorte.com/PaywPlusComercios/**. Las credenciales para ingresar a esta herramienta son proporcionadas mediante correo electrónico al momento de dar de alta la afiliación. La información de la transacción estará disponible hasta 90 días naturales después de realizada la transacción.