

**DOCUMENT MADE AVAILABLE UNDER THE
PATENT COOPERATION TREATY (PCT)**

International application number:	PCT/CA2013/050294
International filing date:	16 April 2013 (16.04.2013)
Document type:	Certified copy of priority document
Document details:	Country/Office: US
	Number: 61/673,096
	Filing date: 18 July 2012 (18.07.2012)
Date of receipt at the International Bureau:	15 August 2013 (15.08.2013)

Remark: Priority document submitted or transmitted to the International Bureau in compliance with Rule 17.1(a),(b) or (b-bis)

24 July 2013 (24-07-2013)

PA 7416156

**THE UNITED STATES OF AMERICA****TO ALL TO WHOM THESE PRESENTS SHALL COME:****UNITED STATES DEPARTMENT OF COMMERCE****United States Patent and Trademark Office****May 03, 2013**

**THIS IS TO CERTIFY THAT ANNEXED HERETO IS A TRUE COPY FROM
THE RECORDS OF THE UNITED STATES PATENT AND TRADEMARK
OFFICE OF THOSE PAPERS OF THE BELOW IDENTIFIED PATENT
APPLICATION THAT MET THE REQUIREMENTS TO BE GRANTED A
FILING DATE UNDER 35 USC 111.**

APPLICATION NUMBER: 61/673,096**FILING DATE: July 18, 2012**

**THE COUNTRY CODE AND NUMBER OF YOUR PRIORITY
APPLICATION, TO BE USED FOR FILING ABROAD UNDER THE PARIS
CONVENTION, IS US61/673,096**

**By Authority of the
Under Secretary of Commerce for Intellectual Property
and Director of the United States Patent and Trademark Office**



W. Montgomery
W. MONTGOMERY
Certifying Officer

Electronic Acknowledgement Receipt

EFS ID:	13285666
Application Number:	61673096
International Application Number:	
Confirmation Number:	4284
Title of Invention:	SYSTEMS AND METHODS FOR FACILITATING A TRANSACTION USING A VIRTUAL CARD ON A MOBILE DEVICE
First Named Inventor/Applicant Name:	Simon LAW
Customer Number:	293
Filer:	Ralph A. Dowell/Alyssa Finamore
Filer Authorized By:	Ralph A. Dowell
Attorney Docket Number:	18126PRO
Receipt Date:	18-JUL-2012
Filing Date:	
Time Stamp:	17:49:23
Application Type:	Provisional

Payment information:

Submitted with Payment	yes
Payment Type	Credit Card
Payment was successfully received in RAM	\$125
RAM confirmation Number	4919
Deposit Account	
Authorized User	

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)/ Message Digest	Multi Part /.zip	Pages (if appl.)
-----------------	----------------------	-----------	-------------------------------------	------------------	------------------

1	Provisional Cover Sheet (SB16)	18126pva.pdf	394110 3f5031dfd0963a290e82743232b29f01c01251c	no	2
Warnings:					
This is not a USPTO supplied Provisional Cover Sheet SB16 form.					
Information:					
2	Drawings-only black and white line drawings	18126drw.pdf	5703671 88144e93cca1c73a4d763f75b7b0a9c0859d65f5	no	13
Warnings:					
Information:					
3		18126spec.pdf	255248 053b7d6425b0f3e601b4f06c52c930192d8a7d15	yes	33
	Multipart Description/PDF files in .zip description				
	Document Description		Start	End	
	Specification		1	31	
	Claims		32	32	
	Abstract		33	33	
Warnings:					
Information:					
4	Fee Worksheet (SB06)	fee-info.pdf	29901 8a1c0e053e1181ea759c4cce2a7b151436bd635b	no	2
Warnings:					
Information:					
Total Files Size (in bytes):			6382930		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office

If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PTO/SB/16 (09-11)

Approved for use through 01/31/2014. OMB 0661-0032

U.S. Patent and Trademark Office, U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PROVISIONAL APPLICATION FOR PATENT COVER SHEET - Page 1 of 2

This is a request for filing a PROVISIONAL APPLICATION FOR PATENT under 37 CFR 1.53(c).

Express Mail Label No. _____

INVENTOR(S)		
Given Name (first and middle (if any))	Family Name or Surname	Residence (City and either State or Foreign Country)
Simon	LAW	Mississauga, Canada
Pierre Antoine	ROBERGE	Toronto, Canada
Peter Thien	DUONG	Mississauga, Canada
Michael	SHVARTSMAN	Miami, Florida
Additional inventors are being named on the _____ separately numbered sheets attached hereto.		
TITLE OF THE INVENTION (800 characters max):		
SYSTEMS AND METHODS FOR FACILITATING A TRANSACTION USING A VIRTUAL CARD ON A MOBILE DEVICE		
Direct all correspondence to: CORRESPONDENCE ADDRESS		
☒ The address corresponding to Customer Number: 000293		
OR		
☐ Firm or Individual Name		
Address		
City	State	Zip
Country	Telephone	Email
ENCLOSED APPLICATION PARTS (check all that apply)		
☐ Application Data Sheet. See 37 CFR 1.75		
☐ CD(s), Number of CDs _____		
☒ Drawing(s) Number of Sheets <u>13</u>		
☐ Other (specify) _____		
☒ Specification (e.g. description of the invention) Number of Pages <u>33</u>		
Fees Due: Filing Fee of \$250 (\$125 for small entity). If the specification and drawings exceed 100 sheets of paper, an application size fee is also due, which is \$310 (\$155 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).		
METHOD OF PAYMENT OF THE FILING FEE AND APPLICATION SIZE FEE FOR THIS PROVISIONAL APPLICATION FOR PATENT		
☒ Applicant claims small entity status. See 37 CFR 1.27.		
☐ A check or money order made payable to the Director of the United States Patent and Trademark Office is enclosed to cover the filing fee and application size fee (if applicable).		
☒ Payment by credit card. ~~Form PTO 2028 is attached.~~		
☐ The Director is hereby authorized to charge the filing fee and application size fee (if applicable) or credit any overpayment to Deposit Account Number: _____		
125.00 TOTAL FEE AMOUNT (\$)		

USE ONLY FOR FILING A PROVISIONAL APPLICATION FOR PATENT

This collection of information is required by 37 CFR 1.51. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to take 16 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

If you need assistance in completing the form, call 1-800-PTO-2199 and select option 2.

PROVISIONAL APPLICATION COVER SHEET
Page 2 of 2

PTO/38/16 (09-11)

Approved for use through 01/31/2014, OMB 0651-0032

U.S. Patent and Trademark Office, U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

The invention was made by an agency of the United States Government or under a contract with an agency of the United States Government.	
☒	No.
☐	Yes, the name of the U.S. Government agency and the Government contract number are:

WARNING:

Petitioner/applicant is cautioned to avoid submitting personal information in documents filed in a patent application that may contribute to identity theft. Personal information such as social security numbers, bank account numbers, or credit card numbers (other than a check or credit card authorization form PTO-2038 submitted for payment purposes) is never required by the USPTO to support a petition or an application. If this type of personal information is included in documents submitted to the USPTO, petitioners/applicants should consider redacting such personal information from the documents before submitting them to the USPTO. Petitioner/applicant is advised that the record of a patent application is available to the public after publication of the application (unless a non-publication request in compliance with 37 CFR 1.213(a) is made in the application) or issuance of a patent. Furthermore, the record from an abandoned application may also be available to the public if the application is referenced in a published application or an issued patent (see 37 CFR 1.14). Checks and credit card authorization forms PTO-2038 submitted for payment purposes are not retained in the application file and therefore are not publicly available.

SIGNATURE /Alyssa Ann Finamore/ Date July 18, 2012

TYPED or PRINTED NAME Alyssa Ann Finamore REGISTRATION NO. 55,177
(if appropriate)

TELEPHONE 703-739-9888 Docket Number: 18126PRO

1 **SYSTEMS AND METHODS FOR FACILITATING A TRANSACTION USING A VIRTUAL**
2 **CARD ON A MOBILE DEVICE**

3
4 **TECHNICAL FIELD**

5 **[0001]** The following relates generally to facilitating a payment transaction at a merchant
6 location using a virtual card on a mobile device.

7 **DESCRIPTION OF THE RELATED ART**

8 **[0002]** Mobile devices can be used to facilitate a payment transaction, for example, in
9 exchange for a good or service at a merchant store. A mobile device can be equipped with
10 a near field communication (NFC) system which can be used to transfer the buyer's payment
11 credential, such as credit card information, to a point of sale terminal that is also equipped
12 with a NFC-compatible system to complete the payment transaction.

13 **BRIEF DESCRIPTION OF THE DRAWINGS**

14 **[0003]** Embodiments will now be described by way of example only with reference to the
15 appended drawings wherein:

16 **[0004]** Fig. 1 is a schematic diagram of an example embodiment of a payment system.

17 **[0005]** Fig. 2 is a schematic diagram of an example embodiment of one side of a funding
18 credit card.

19 **[0006]** Fig. 3 is a schematic diagram of the other side of the funding credit card in Fig. 2.

20 **[0007]** Fig. 4 is a schematic diagram of an example embodiment of a payment system
21 showing the flow of data when using a virtual card to facilitate a payment.

22 **[0008]** Fig. 5 is another schematic diagram view of the entities involved in an example
23 embodiment of a payment transaction using a virtual card to facilitate a payment.

24 **[0009]** Fig. 6 is a block diagram of an example embodiment of a mobile device.

25 **[0010]** Fig. 7 is a flow diagram of an example embodiment of computer executable or
26 processor implemented instructions for generating and using a virtual card in a payment
27 transaction according to a user's perspective, the instructions performed by at least a mobile
28 device, a point of sale terminal and a payment gateway server.

29 **[0011]** Fig. 8 is a flow diagram of an example embodiment of computer executable or
30 processor implemented instructions for facilitating payment transaction using a virtual card,

22259402.1

1 the instructions showing the interaction between at least a merchant and a payment gateway
2 server.

3 **[0012]** Fig. 9a is a flow diagram of an example embodiment of computer executable or
4 processor implemented instructions for computing virtual card details by the payment
5 gateway server, the instructions implemented as part of the transaction process of Fig. 7.

6 **[0013]** Fig. 9b is a flow diagram of an example embodiment of computer executable or
7 processor implemented instructions for authorising virtual card details by the payment
8 gateway server, the instructions implemented as part of the transaction process of Fig. 8.

9 **[0014]** Fig. 10 is a flow diagram of an example embodiment of computer executable or
10 processor implemented instructions for a registration process between a mobile device and
11 a payment gateway server.

12 **[0015]** Fig. 11 is a flow diagram of an example embodiment of computer executable or
13 processor implemented instructions for computing virtual card details including a primary
14 account number and discretionary data, using the data exchanged during the registration
15 process of Fig. 10.

16 **[0016]** Fig. 12 is a flow diagram of an example embodiment of computer executable or
17 processor implemented instructions for verifying the discretionary data of Fig. 11.

18 DETAILED DESCRIPTION

19 **[0017]** It will be appreciated that for simplicity and clarity of illustration, where considered
20 appropriate, reference numerals may be repeated among the figures to indicate
21 corresponding or analogous elements. In addition, numerous specific details are set forth in
22 order to provide a thorough understanding of the example embodiments described herein.
23 However, it will be understood by those of ordinary skill in the art that the example
24 embodiments described herein may be practiced without these specific details. In other
25 instances, well-known methods, procedures and components have not been described in
26 detail so as not to obscure the example embodiments described herein. Also, the
27 description is not to be considered as limiting the scope of the example embodiments
28 described herein.

29 **[0018]** Fig. 1 shows an example embodiment of a typical in-person payment flow
30 between a cardholder 101, merchant 102, merchant acquirer 103, payment network 109 and
31 card issuer 104. By way of background, this is also known as a standard 4-party model, or a
22259402.1

standard 4-party protocol. The card issuer 104 is typically a bank which is associated with a payment network. It facilitates the cardholder's use of the payment card to pay for goods or services based on the cardholder's promise to repay the card issuer for the purchase. Non-limiting examples of payment networks include Visa, MasterCard, American Express and Diners Club/Discover. The card issuer 104 issues the payment card, which is associated with a payment network 109, to the cardholder 101 (block 105). It is understood that all references to credit card would also apply to debit card, prepaid card, or other payment credentials used to facilitate payment at a merchant location. The term "funding card" is used herein to refer to debit cards, credit cards, prepaid cards, and the like.

[0019] To make a purchase or payment, the cardholder 101 (e.g. a person) provides funding card information to the merchant 102 (block 106). For example, the funding card can be "swiped" using a magnetic stripe card reader device, or the funding card number can be read to the merchant 102. The merchant 102 (e.g. a computing device) sends the payment authorisation, including the funding card details, to a merchant acquirer 103 (block 107). An acquirer 103 is an organization that collects payment-authorisation requests from merchants and facilitates the payment transaction with the payment network on behalf of the merchants.

[0020] When the acquirer 103 gets the funding card payment authorisation request, the acquirer submits the payment transaction (block 108) to the payment network 109.

[0021] When the payment network 109 receives the funding card payment authorisation request, the payment network forwards the payment transaction (block 110) to the card issuer 104 for payment authorisation.

[0022] When the card issuer 104 receives the payment transaction, it checks the transaction details for validity. This includes checking the card number, the expiration date, the funding card limit, etc. The card issuer 104 responds to the merchant acquirer 103 with a payment authorisation code (approved or declined response, transaction identification number, etc) via the payment network 109. The merchant acquirer 103 forwards the response to the merchant 102. Then the merchant 102 share the payment authorisation results with the cardholder.

[0023] By way of background, an example embodiment of a credit card 201 is shown in Fig. 2 and Fig. 3. One side of the credit card, as per Fig. 2, shows a logo 202 or mark identifying the card issuer. Typically, the card issuer is a bank. A logo 203 or mark is also

shown to identify the payment network. The card 201 also shows the credit card number 204. The credit card number 204 is sometimes referred to as a primary account number (PAN). The length and format of the credit card number 204 varies depending on the card issuer and the payment network. Generally, the first digit in the credit card number 204 identified the payment network. The last digit of the credit card number 204 is a check digit. A check digit is a form of redundancy check used for error detection. The intermediary numbers can signify a bank number (e.g. associated with the card issuer) and an account number (e.g. of the bank).

[0024] In the example shown in Fig. 2, as per block 207, the first digit "4" identifies that Visa is the payment network. Digits two through six are the issuing bank identification number. Digits seven through fifteen are the account number. Digit sixteen is the check digit.

[0025] The credit card 201 also includes a range of dates 205 that the credit card is valid. The range of dates includes an expiry date of the credit card. The name 206 of the cardholder is also shown.

[0026] In Fig. 3, on the other side of the credit card 201, there is a magnetic strip 301, the signature 302 of the cardholder's name, and a static security code 303. The static security code is printed on the credit card 201.

[0027] The magnetic stripe 301, also sometimes referred to as a magstripe, is typically made up of tiny iron-based magnetic particles in a plastic-like film. The stripe 301 has information that is written on it.

[0028] Often there are two or three tracks on data encoded on the stripe 301. In an example embodiment, each track is about one-tenth of an inch wide. The ISO/IEC standard 7811, which is used by banks, specifies that track one is 210 bits per inch (bpi), and holds 79 6-bit plus parity bit read-only characters; track two is 75 bpi, and holds 40 4-bit plus parity bit characters; and track three is 210 bpi, and holds 107 4-bit plus parity bit characters. Other data formats may apply.

[0029] In an example embodiment, the format for Track Two, developed by the banking industry, is as follows (maximum of 40 characters):

- Start sentinel - one character (generally ';')
- Primary account number - up to 19 digits

22259402.1

- 1 • Separator - one character (generally '=')
- 2 • Expiration date - four digits in the form of the last two digits of the year, and the two digits
- 3 representing the month (e.g. YYMM)
- 4 • Service Code – three digits. The first digit specified the interchange rules, the second
- 5 specifies authorisation processing and the third specifies the range of service
- 6 • Discretionary data – balance of available space to fill the length of the Track Two (ex 10
- 7 digits if the PAN is 19 digits long)
- 8 • End sentinel – one character (generally '?')
- 9 • Longitudinal Redundancy Check (LRC) - one character

10 **[0030]** The above data is herein generally referred to as Track Two data.

11 **[0031]** It is recognized, that in addition to the typical plastic card form factor, as shown in
 12 Fig.2 and Fig. 3, there are a growing numbers of other form factors that can be used for
 13 payment. Non-limiting examples include a mini card, fob, mobile device, etc. It is also
 14 appreciated that other types of funding cards, in addition to credit cards, include similar
 15 information (e.g. PAN, expiry date, Track Two data, etc.).

16 **[0032]** The way the card data itself is stored on a funding card and shared with a
 17 payment terminal has many variants, such as storing the card data on a magnetic stripe and
 18 reading the data using a magnetic card reader, or storing the card data using chip
 19 technology and using a chip-reader compatible reader to interact with the card. For
 20 example, although not shown in Fig. 2 and Fig. 3, the credit card can include an integrated
 21 circuit or logic chip capable of performing computations. Various examples of chip-based
 22 card technology includes contactless magnetic stripe emulation, EMV contact, EMV
 23 contactless, etc. EMV stands for Europay, MasterCard and VISA, a global standard for
 24 inter-operation of integrated circuit cards (IC cards or "chip cards") and IC card reader. In
 25 addition to facilitate the exchange of information via the contact interface of the card,
 26 information can also be exchanged using the contactless capabilities of capable card. Such
 27 contactless cards are sometimes referred to as PayPass, payWave, and ExpressPay.

28 **[0033]** It is recognized that contactless payment is well suited for a mobile device. For
 29 example, some mobile devices have a near-field communication (NFC) system that allows
 30 data to be transferred wirelessly, using the ISO/IEC 18092 standard or other compatible
 31 standards, over a relatively short distance. A NFC-enabled mobile device can establish
 32 radio communication with another contactless-enabled device by touching them together or
 33 keeping them into close proximity. The term "mobile device" and "mobile phone" are
 34 interchangeably used herein.

22259402.1

1 **[0034]** An NFC-enabled mobile device can be equipped with a “software payment card”
2 (e.g. a software implementation of a contactless payment card) application running inside
3 the mobile device. The software payment card uses the compatible contactless
4 communication capabilities of the device to interact with contactless-enabled POS terminal
5 to facilitate payment transactions.

6 **[0035]** In many cases, the software payment card is stored on the secure element to
7 protect the sensitive payment data such as the one used to generate the dynamic data
8 necessary to complete a typical contactless payment transaction. It is recognized that
9 secure elements are limited in their storage capabilities. They are typically available in the
10 form of an embedded chip, such as in a universal integrated circuit card (UICC), or in a
11 subscriber identity module (SIM) card.

12 **[0036]** In part to maintain the security of the various applications running inside the
13 secure element, the secure element is typically managed by a mobile phone carrier
14 distributing the secure element with the mobile device. Part of the managed service includes
15 delivering applications into the secure element directly, or giving permission to a third party
16 organization to deploy their application on a particular secure element. The managed
17 service is typically delivered using what the industry referred to as a trusted service manager
18 (TSM).

19 **[0037]** All applications stored and running inside the secure element, such as the
20 individual “software payment card”, need their own space. Payment cards are issued to
21 consumers by the card issuer. Deploying software payment cards on mobile phones requires
22 a high level of coordination between the mobile phone carrier and the card issuer where the
23 mobile phone carrier provides access to individual secure elements, one at a time, to the
24 issuer. Only cards from funding card issuers that have the infrastructure and the agreement
25 with the mobile phone carrier can be used on the mobile phone for contactless payment.
26 This is limiting for both the card issuers and for cardholders.

27 **[0038]** In the United States, for example, there are thousands of issuing banks and tens
28 of mobile phone carriers. The mobile phone carriers use the TSM, in part, to issue and
29 manage payment credentials into mobile phone. The TSM enables mobile phone carriers
30 (also called service providers) or other entities controlling the secure element on mobile
31 phones to distribute and manage remotely the secure applications running inside the secure
32 element by securing access to the secure element in mobile-enabled devices. An example
22259402.1

1 of secure application is a software payment card. The issuing banks also interact with the
2 mobile phone carriers through a service (in an example embodiment, called ISIS) to manage
3 payment credentials issued to NFC-enabled devices. In Canada, Rogers Communication is
4 a mobile carrier that is in partnership with CIBC, an issuing bank, to issue software payment
5 card to CIBC cardholder's phone on the Rogers network. Establishing systems and
6 methods to manage the secure issuance and management of software payment cards
7 between the issuing banks, the mobile phone carriers, and the mobile device is costly and
8 complex. It involves pre-arranged agreement between the parties as well as some
9 customizations of the managed service to meet the needs of all the entities participating.
10 This can make it difficult for smaller-sized card issuers to adopt NFC technology for facilitate
11 payments. It also means that the issuing bank will likely require numerous point to point
12 connections with the various mobile operators it wants to supports. The systems and
13 methods described herein attempt to address this issue.

14 **[0039]** It is also recognized that from the user's perspective, the process of associating
15 their mobile device with their funding card to be used for contactless payment is very much
16 dependent on pre-arranged relationships between the mobile phone carrier and the card
17 issuers. Therefore, a user has limited options or no options when determining if their current
18 funding card can be associated with their mobile phone for contactless payments. For
19 example, a user has a funding card from Funding Card Issuer A. The user also has a NFC-
20 enabled mobile phone from Mobile Phone Carrier B. However, Mobile Phone Carrier B only
21 has a pre-arranged agreement and infrastructure to facilitate contactless payments with
22 Funding Card Issuer B. Therefore, even if the user wanted to use their mobile phone to
23 make a contactless payment, the user would not be able to because there is no pre-
24 arranged agreement and infrastructure between Mobile Phone Carrier B and Funding Card
25 Issuer A to issue a software payment card into the user's phone. This limits the user's ability
26 to make NFC-type payments with their mobile device.

27 **[0040]** In a typical example embodiment of making a payment using an NFC-enabled
28 mobile device, a user first request to its issuer to load a funding card on their mobile device.
29 When this process is completed, a software funding card has been delivered and installed
30 securely on the secure element on the user's mobile device. The user can now use the
31 funding card on the mobile device's to facilitate contactless payment transactions at
32 merchant locations that are equipped to accept such type of payment transaction. Once the
33 user attempt to pay with their mobile device at a merchant, the payment details are sent and
22259402.1

1 are verified by the funding card issuing server in a similar manner to other payment
2 transaction.

3 **[0041]** When a contactless funding card credential is used to facilitate a payment
4 transaction, the transaction most often includes dynamic data from the card to securely
5 authenticate the payment credential. The dynamic data changes values every time the
6 credential is used. If the dynamic data received on the funding card issuing server matches
7 the expected value computed by the server for the card, the authentication of the payment
8 credentials is deemed successful and the payment authorisation can continue. It can be
9 appreciated that there are various ways in which the mobile device and the funding card
10 issuing server can compute the dynamic data used to authenticate the payment credential.

11 **[0042]** In an example embodiment for credit cards, the dynamic data is a rotating card
12 verification value (CVV, also referred sometime to dynamic CVV or dCVV). This rotating
13 CVV is computable based on changing information provided by the integrated circuit inside
14 the card. In another example embodiment, the dynamic data is dynamic EMV data which is
15 computed using random data from the funding card, or random data from a merchant's point
16 of sale terminal, or both. A common implementation of dynamic data uses an Application
17 Transaction Counter (ATC) on the card so that every transaction produces a different data
18 stream. This is achieved as the ATC is incremented by 1 for every transaction performed.
19 When a user taps, touches or positions their mobile device near a contactless-enabled point
20 of sale (POS) terminal, the funding card data (card number, dynamic data, expiry date, etc.),
21 hereafter referred to as the card's Track Two data, is sent from the mobile device to the POS
22 terminal. This information then makes it way to the funding card issuing server for
23 verification. The funding card issuer will perform numerous checks to validate the
24 transaction, including comparing the dynamic data of the mobile device with the value
25 generated by the server. If the dynamic data is matching, and all the other checks and
26 controls performed by the funding card issuer are successful, the funding card issuer will
27 respond with a positive payment authorisation response.

28 **[0043]** It is also recognized that a card application specific to a given funding card can
29 be installed on the mobile device and used to interact with POS terminal as described
30 above. It is also recognized that the card application is typically installed on the mobile
31 device's secure element. Typically, each funding card has its own corresponding card
32 application that resides on the mobile device's secure element. It can be appreciated that as

22259402.1

1 each card application takes up storage space on the secure element, and that the secure
2 element typically has very limited storage space, having multiple card applications on the
3 secure element in some cases is not possible due to insufficient storage space. By way of
4 background, the secure element can have a native operating system that be programmed
5 to perform various tasks and activities, including for example, a card application that
6 emulates the magnetic strip data of a funding card or a card application that emulates the
7 data used in an EMV contactless payment. Also by way of background, and by way of
8 example, a typical secure element has memory of 256 kB, and each card application can
9 consume memory of 40-80 kB. It can therefore be appreciated that associating multiple
10 funding cards (and each of their card applications) with a mobile device for NFC payments
11 can be limiting.

12 **[0044]** Therefore, it is desirable to reduce the amount of required storage space card
13 applications need on the secure element so as to not limit the number of software payment
14 cards a user can load into a secure element. Along the same lines, it is desirable for mobile
15 phone carriers to reduce the amount of data used by "card application" on the secure
16 element so that other types of applications can be loaded thereon. It is also desirable to
17 reduce costs incurred by the funding card issuer to issue and operate software payment
18 cards into secure elements. By way of background, a mobile phone carrier typically charges
19 application providers, such as funding card issuers, for the amount of storage space used on
20 the secure element. It is also desirable to reduce the amount of infrastructure required by
21 the funding card issuer to issue a software payment card for the mobile phone. It is also
22 desirable to reduce the amount of coordination required between the funding card issuer and
23 the mobile phone carrier to issue a software payment card on particular mobile phone. It is
24 also desirable to enable the user (e.g. the cardholder) to load any, and as many, funding
25 cards they want into their NFC-enabled mobile phone, independently of the funding card
26 issuer having the infrastructure or a commercial relationship or agreement with a particular
27 mobile phone carrier.

28 **[0045]** The systems and methods described herein attempt to address the above issues.

29 **[0046]** In general, the systems and methods described herein allow a cloud-based wallet
30 payment gateway server to synchronize with a NFC-enabled mobile device and application
31 to facilitate contactless payment transactions at a merchant store accepting contactless
32 payment. A pre-arranged agreement or additional infrastructure between the funding card

22259402.1

1 issuer and the mobile phone carrier is not required. A user selects a funding card for making
2 the contactless payment, through their mobile device. A second card, herein referred to as a
3 virtual card, the card is generated along with all required card data (e.g. PAN, expiry date,
4 dynamic data, discretionary data, etc.) to complete the payment transaction. The virtual card
5 is temporary associated with the funding card on the payment gateway server. The data
6 required to compute virtual card data set is sent on the mobile device, typically every time
7 that the user wants to make a payment. For example, a new virtual card is created for each
8 and every payment transaction and has a much shorter usage period compared to a
9 standard funding card which can be used typically over several years.

10 **[0047]** When a payment is initiated, the virtual card data is sent through the NFC system
11 on the mobile device to a contactless-enabled POS terminal. This information is sent from
12 the POS terminal to the merchant system; from the merchant system to the merchant
13 acquiring bank; and from the acquiring bank to the cloud-based wallet payment gateway
14 server (also referred to the "payment gateway server") via the payment network. The cloud-
15 based wallet payment gateway server acts as the virtual card issuer server and verifies the
16 virtual card. If successfully verified, the funding card details associated with the virtual card
17 are retrieved and sent to the funding card issuer server via the payment network to complete
18 the payment authorisation transaction. The funding card issuer server verifies the funding
19 card and sends back an authorization code to the payment gateway server. The payment
20 gateway server sends back a corresponding authorization code to the merchant system.

21 **[0048]** The payment can be settled when the merchant system initiates a settlement
22 request, typically once at the end of every business day. To complete the settlement, the
23 merchant system sends all the virtual card numbers and the corresponding authorization
24 codes received during the period to the virtual card issuer, via the merchant acquiring bank.
25 The virtual card issuer verifies the virtual card numbers and authorisation codes. For all the
26 matching records, the virtual card issuer retrieves the associated funding card numbers and
27 authorisation codes and sends the data to the funding card issuer for settlement via the
28 virtual card issuer acquiring bank. The funding card issuer verifies the funding card numbers
29 and authorization codes, and if successfully verified, sends the money via a standard
30 method to the payment transaction originator, in this case the virtual card issuer. At that
31 point, the virtual card issuer sends the money to the merchant acquiring bank also using a
32 standard method. In an example embodiment, the virtual card issuer is the payment
33 gateway server, or a module within the payment gateway server.
22259402.1

[0049] In an example embodiment, the systems and methods described herein allow a cloud-based wallet (e.g. associated with a funding card) to be synchronized with a NFC-enabled mobile device (e.g. associated with a virtual card). In another example embodiment, the systems and methods described herein provide real time rotation of a primary account number (PAN) and dynamic data used to facilitate contactless payment transactions. In another example embodiment, the systems and methods described herein use virtual payment credentials to complete a brick-and-mortar contactless purchase transaction without disclosing the user's funding card details.

[0050] Turning to Fig. 4, an example embodiment showing the flow of payment data using a virtual card is provided. A card issuer, also called the funding card issuer 104, issues a standard card account, most often in the form of a plastic card, to the user 101 (block 404). The standard card is the funding card, and the user becomes the cardholder.

[0051] Although not shown, the cardholder 101 registers one or many funding cards and their mobile device with a virtual card issuer 401. It can be appreciated that any funding card can be registered, and is not limited or dependent on the mobile phone carrier having agreement with the funding card issuer. It can also be appreciated that any number of funding cards can be registered in association with the mobile device. The cardholder's mobile device includes a payment application that can interact with the virtual card issuer 401.

[0052] In an example embodiment of the registration, for each funding card the user wishes to register, the user types in card details into the mobile device (e.g. the name printed on the funding card, the PAN printed on the funding card, the expiry date printed on the funding card, and the static security code printed on the funding card). The mobile device sends this data, plus a user provided PIN and the mobile device ID to the payment gateway server. For each funding card, the payment gateway server computes a funding card identifier which identifies the given funding card. The payment gateway server stores the funding card identifier in association with the funding card details, mobile device ID and PIN, and it sends the funding card identifier to the mobile device for storage. In an example embodiment, the funding card identifier is a value that is different from the PAN, expiry date or static security code of the funding card. For example, the funding card identifier is a random value so that, if intercepted by an adversary, would not be able to recognize any funding card details. In an example embodiment, the mobile device does not store any

22259402.1

1 funding card details but only stores limited funding card details (e.g. the name funding card
 2 issuer and the last 4 digits of the PAN). The mobile device stores the funding card identifier,
 3 which it sends to the payment gateway server to indicate a specific funding card. It can be
 4 appreciated that there are other methods to capture the funding card details (e.g. besides
 5 the user typing in the data), which can be used with the principles described herein.

6 **[0053]** It can be appreciated that a single payment application is required on the mobile
 7 device, which can manage multiple funding cards. If multiple funding cards are registered,
 8 each of the associated funding card identifiers are stored on the mobile device, within the
 9 single payment application. The details of each individual funding card are stored on the
 10 payment gateway server. In this way, the payment gateway server acts as a cloud-based
 11 server that stores the details of multiple funding cards. Additional details of a funding card
 12 registration process are described below with respect to Fig. 10.

13 **[0054]** Continuing with Fig. 4, when the registered cardholder 101 wishes to make a
 14 contactless payment using their mobile device, the user selects the funding card. The
 15 selected funding card is conveyed to the virtual card issuer 401, and the virtual card issuer
 16 401 issues a virtual card to the cardholder 101, and more specifically to the payment
 17 application running on the mobile device (block 405). The cardholder touches their mobile
 18 device on the merchant system and the virtual card details are delivered to the merchant
 19 106, via a NFC-enabled POS terminal (block 406). The merchant sends the virtual card
 20 details to the merchant acquirer 103 (block 407). Based on some of the virtual card details,
 21 the merchant acquirer 103 sends the virtual card details to the virtual card issuer 401 (block
 22 408) via the payment network. The virtual card issuer 401 uses the virtual card details to
 23 determine the corresponding funding card (block 409). The virtual card operator 402 (e.g.
 24 the virtual card issuer now acting as a "merchant") sends the corresponding funding card
 25 details, including the original transaction amount, to its acquirer 403 (block 410). This
 26 parallels the typical process of a merchant 106 sending the funding card details to the
 27 merchant acquirer 103. The virtual card acquirer 403 sends the funding card details to the
 28 funding card issuer 104 (block 411) via the payment network for a standard payment
 29 authorisation.

30 **[0055]** The funding card issuer 104 then can authorise, or not, the payment request and
 31 responds back with an authorisation code to the virtual card issuer's acquirer, via the
 32 payment network, which in turn notify the virtual card issuer. The virtual card issuer notifies

22259402.1

1 the merchant acquirer using a corresponding authorization code for the merchant system via
2 the payment network.

3 **[0056]** To settle the funds (not shown), the merchant will initiate, or the merchant
4 acquirer will initiate on behalf of the merchant, a settlement request to the payment network
5 via the merchant acquirer. The payment network will forward the corresponding virtual card
6 transactions to settle to the virtual card issuer. When the virtual card issuer receives the
7 transactions, the virtual card issuer will send a settlement request for the corresponding
8 funding card transactions to its own acquirer. The settlement request will be sent to the
9 appropriate funding card issuer by the payment network. The funding card issuer receives
10 the transactions settlement request, and will send the funds associated with all matching
11 transactions to the payment transaction originator, in this case the virtual card issuer. When
12 the virtual card issuer receives the funds, the virtual card issuer will send the corresponding
13 funds to the merchant.

14 **[0057]** In an example embodiment, the virtual card issuer 401, the virtual card operator
15 402, and the virtual card acquirer 403 are represented by the same entity, referred herein as
16 the payment gateway server. The payment gateway server itself can include one or more
17 servers. For example, each of the entities 401, 402, 403 can be individual servers that,
18 combined, form the payment gateway server.

19 **[0058]** Turning to Fig. 5, example embodiment components of a system for facilitating
20 payment using the virtual card is shown. The user 101 has one or more funding cards 505
21 pre-registered into its cloud-wallet (process not shown). For example, the user has multiple
22 funding cards. The user 101 also own an NFC-enabled mobile device 501, which includes a
23 payment application. The mobile device 501 is configured to interact, via NFC, with a
24 merchant POS terminal device 502. The merchant POS terminal device 502 is in data
25 communication via the merchant payment system with the merchant acquirer 103 (e.g. a
26 server). The merchant acquirer and the payment gateway server 506 are in communication
27 with a payment network (e.g. Visa, MasterCard, Discover, etc.), which is configured to send
28 data related to payments and transactions to relevant parties, including the payment
29 gateway server 506 and the funding card issuer 104 (e.g. a server). The communication
30 between the merchant POS terminal device 502, the merchant acquirer 103, the payment
31 network 504, the payment gateway server 505 and the funding card issuer 104 can occur
32 over wired or wireless communication networks, or both.

22259402.1

1 **[0059]** The payment gateway server 506 is also in communication with the mobile device
 2 501 through a wireless network. For example, the wireless network is provided by a mobile
 3 phone carrier.

4 **[0060]** Example components of the payment gateway server 506 are shown in block
 5 507. The server 506 can include the virtual card issuer 401, the virtual card operator 402
 6 and the virtual card acquirer 403. During a registration process conducted by the user 101,
 7 the payment gateway server 506 stores the user selected personal identification number
 8 (PIN), funding card data in association with the user's mobile device ID (e.g. in database
 9 508). For example, the user PIN, funding card1 and funding card2 (and other funding cards)
 10 are stored in association with the mobile device ID of mobile device 501. Also, in database
 11 509, the temporary data associations between a given funding card, a virtual card and
 12 authorisation code when applicable are stored. For example, virtual card1, funding card1
 13 and authorisation code 1 are all stored in association with each other's. Another example is
 14 virtual card2 and funding card2 are also associated but have no authorisation code issued
 15 yet for the card pair. In an example embodiment, there is no authorisation code issued yet
 16 because the user has not yet tapped the mobile device using funding card2 (e.g. really
 17 virtual card2), or because the merchant payment authorisation was interrupted such that the
 18 payment gateway server never received the authorisation code.

19 **[0061]** Turning to Fig. 6, example components of the mobile device 501 are shown.
 20 The mobile device 501 includes a main processor 601 which interacts with a number of
 21 components including, among other things, auxiliary inputs/outputs 302, a data port 603, a
 22 keyboard 604, a speaker 605 (e.g. an audio speaker), a microphone 606, a GPS receiver
 23 607 and a camera 608. The mobile device 501 also includes an NFC subsystem 609, a
 24 secure element 622 which may or may not also have direct connectivity to the NFC
 25 subsystems 609, and other device subsystems 611.

26 **[0062]** The mobile device 501 uses a communication system 613 to interact with a
 27 wireless network 612. Memory types include flash memory 614 and random access memory
 28 (RAM) 615. The mobile device's display 616 can be a touch-screen type display or another
 29 type of display.

30 **[0063]** An operating system 617 may be used to manage and run software components
 31 618. Software components or applications include a web browser or internet browser 619
 32 and payment application 620. Other software components 621 are included.

22259402.1

1 **[0064]** The secure element 622 can be used for storing information such as application
 2 and data elements, for example a payment application and a mobile identifier. In an example
 3 embodiment, the secure element is inside a subscriber identity module (SIM) card. Non-
 4 limiting examples of mobile devices include cell phones, smart phones, PDAs, tablets,
 5 netbooks, and laptops.

6 **[0065]** It will be appreciated that any module or component exemplified herein that
 7 executes instructions or operations may include or otherwise have access to computer
 8 readable media such as storage media, computer storage media, or data storage devices
 9 (removable and/or non-removable) such as, for example, magnetic disks, optical disks, or
 10 tape. Computer storage media may include volatile and non-volatile, removable and non-
 11 removable media implemented in any method or technology for storage of information, such
 12 as computer or processor readable instructions, data structures, program modules, or other
 13 data, except transitory propagating signals per se. Examples of computer storage media
 14 include RAM, ROM, EEPROM, flash memory or other memory technology, CD-ROM, digital
 15 versatile disks (DVD) or other optical storage, magnetic cassettes, magnetic tape, magnetic
 16 disk storage or other magnetic storage devices, or any other medium which can be used to
 17 store the desired information and which can be accessed by an application, module, or both.
 18 Any such computer storage media may be part of the any one of the servers 103, 504, 104,
 19 506, 401, 402, 403, the mobile device 501, the POS terminal 502, etc. or accessible or
 20 connectable thereto. Any application or module herein described may be implemented using
 21 computer or processor readable/executable instructions or operations that may be stored or
 22 otherwise held by such computer readable media.

23 **[0066]** Turning to Fig. 7, example computer executable or processor implemented
 24 instructions are provided for facilitating a transaction between a mobile device 501 and a
 25 POS terminal device 502.

26 **[0067]** It is assumed that the end-user has a compatible NFC-mobile device 501. It is
 27 assumed that payment application 620 has been installed on the mobile device 501. It is
 28 assumed that the end user has registered for the service.

29 **[0068]** Following the merchant capturing the purchase transaction details into its point of
 30 sale system, the NFC-enabled POS terminal device 502 displays a message to the end user
 31 to "tap" to pay (block 701). For example, the user can tap their mobile device 501 or a

22259402.1

1 contactless funding card. The examples described herein relate to tapping the mobile
2 device 501.

3 **[0069]** The end user 101 sees the message from the POS terminal device 502 and
4 starts the payment application 620 on the mobile device 501 (block 702). For example, the
5 user selects an icon for the payment application 620 on a home screen of the mobile device
6 501, thereby launching the payment application 620. The payment application 620
7 determines if the user has successfully registered to the service (block 703), and if so,
8 shows a menu (block 704) of supported actions by the application. If the user has not
9 registered, the menu offers the user to register with the service (e.g. register one or more
10 funding cards, provide a PIN, link a mobile device identifier to the registration record).

11 **[0070]** A graphical user interface (GUI) on the menu is able to receive an input from the
12 user to initiate a payment with a virtual card with the payment application (block 705).
13 Example of other menu item includes "add a funding card", "delete a funding card", etc.

14 **[0071]** The mobile device 501 then displays the funding cards that have been pre-
15 registered by the user (block 706). A user input is provided to select one of the funding
16 cards (block 707). In an example embodiment, the displayed funding card information is
17 loaded after the end user successfully registers to the service and has registered at least
18 one funding card. The list is updated when the user adds an additional funding card into the
19 payment application or when a funding card is deleted. For each registered funding card,
20 there is a corresponding record stored on the payment application 620 and on the payment
21 gateway server database 508 which includes an identifier for the payment network
22 associated with the funding card, a funding card identifier, etc.

23 **[0072]** The payment application 620 sends the mobile device identifier to the payment
24 gateway server 506 (block 708). The payment application 620 also sends the transaction
25 type (selected action in the application menu, in this case, "Pay with a virtual card") and the
26 identifier for the selected funding card to the payment gateway server 506 (block 709).
27 Based on the information received by mobile device 501, the payment gateway server 506
28 creates and computes the details regarding the virtual card (block 710). In particular, the
29 payment gateway server 506 computes a virtual PAN, an expiry date of the virtual card and
30 all or some of the data elements that form the Track Two data. It is noted that Track Two
31 data includes, among other things: the PAN, a service code, an expiry date, discretionary
32 data and a LRC. In an example embodiment, the payment gateway server 506 at this time
22259402.1

1 does not compute the discretionary data, which is dynamic in nature (e.g. the discretionary
 2 data is dynamic data). In an example embodiment, the expiry date of the virtual card is
 3 identical to the expiry date of the funding card, so that from the perspective of the merchant
 4 and the user the virtual card is identical to the funding card. In fact, based on certain
 5 similarities between the funding card and the virtual card, the merchant and the user will be
 6 unaware that a virtual card is being generated and used and will instead believe that the
 7 funding card is being used in the payment. The virtual card details may further include an
 8 internal expiry date that is known only to the payment gateway server, and has a short
 9 timeline of about a few days from the date that the virtual card is created. The internal expiry
 10 date is different from the virtual card's expiry date, and the function of the internal expiry date
 11 is to provide an additional indicator to the payment gateway to determine whether or not a
 12 virtual card has expired. I The payment gateway server 506 encrypts the virtual card data,
 13 which does not include the internal expiry date, and sends the encrypted virtual card payload
 14 to the mobile device's payment application 620 (block 711).

15 **[0073]** As an alternate example embodiment, instead of the payment gateway server
 16 506 sending the virtual card PAN as part of the encrypted virtual card payload to the mobile
 17 device, the payment gateway server 506 instead sends a key value (called Kpan) that the
 18 mobile device can use to generate an identical virtual card PAN as computed by the
 19 payment gateway server 506.

20 **[0074]** In an example embodiment, the card last four digits of the virtual card PAN are
 21 the same as the last four digits of the funding card PAN. The last four digits of the virtual
 22 card PAN are identical to the funding card PAN so that, after PAN truncation, it appears as if
 23 the PAN of the virtual card is identical to the PAN of the funding card. In other words, the
 24 user cannot detect that a virtual card is being used in place of a funding card. By way of
 25 background, PAN truncation is enforced by the payment industry as part of the merchant
 26 certification process for accepting card payment. A truncated PAN means that the card
 27 number, when printed on a customer receipt, is replaced with a printout of only the last four
 28 digits, and the remainder of the other PAN digits are replaced usually by asterisks. An
 29 example embodiment of a truncated PAN is * * * * * 7777. This hides the card
 30 number from anyone who obtains the receipt when discarded, or by other means, while still
 31 allowing a cardholder with multiple cards to identify which was used, and thus record the
 32 transaction.

22259402.1

1 **[0075]** In an example embodiment, the first portion of digits of the PAN of the virtual card
2 is static and refers to the payment gateway server 506. For example, the first six digits point
3 to the payment gateway server 506; the merchant acquirer 103 and associated payment
4 network uses this information to send the transaction and payment details to the payment
5 gateway server 506.

6 **[0076]** In an example embodiment, the PAN of the virtual card is nineteen digits long
7 and compliant to the algorithm LUHN-10. The algorithm, also known as the "modulus 10" or
8 "mod 10" algorithm, is a checksum formula used to validate a variety of identification
9 numbers, such as card numbers. As described above, the first six digits are used to identify
10 the payment gateway server and the last four digits are identical to the last four digits of the
11 funding card PAN. The remaining digits can be computed in a number of ways. In an
12 example embodiment, the remaining digits of the virtual card PAN are randomly generated.
13 In another example embodiment, the remaining digits are computed using the Kpan value;
14 further details in this regard are described in Fig. 11. Other methods can be used to
15 compute the virtual card PAN.

16 **[0077]** The mobile device 501 receives the encrypted virtual card payload, decrypts the
17 encrypted communication and extracts the virtual card details (e.g. the virtual card PAN and
18 other card details).

19 **[0078]** In another example embodiment, if the virtual card payload includes a Kpan (e.g.
20 a key value) instead of a virtual card PAN, the mobile device 501 uses the Kpan to compute
21 the virtual card PAN.

22 **[0079]** The payment application 620 on the mobile device 501 displays a GUI
23 requesting the user 101 to input their PIN (e.g. which should be the same PIN provided
24 when the user registered for the service) (block 712). The application 620 receives the PIN
25 from the user 101 (e.g. the user enters in the PIN) (block 713). The application 620 uses the
26 PIN to compute the discretionary data of the virtual card (block 714). With the discretionary
27 data computed, the Track Two data set is complete. The virtual card data set (e.g. the Track
28 Two data) is delivered to the mobile device's secure element 622 (block 715). In an
29 example embodiment, data to be sent over the NFC subsystem 609 needs to be provided by
30 the secure element 622 (block 716). In another example embodiment, though not shown,
31 the operating system of the mobile device 501 can directly transfer information to the NFC
32 subsystem 609 without the use of the secure element. The virtual card data set includes the
22259402.1

1 virtual card PAN, the expiry date, the discretionary data and all other elements in a Track
2 Two data set to complete a standard contactless payment transaction. The mobile's
3 payment application 620 displays a message to the user to "Tap to pay" (block 717).

4 **[0080]** At this stage, the user 101 touches or positions the mobile device 501 in close
5 proximity to the POS terminal device 502 (block 718). When the mobile device 501
6 becomes in communication range to the POS terminal device 502, the mobile device 501
7 sends the virtual card data to the POS terminal device 502, using the mobile device's NFC
8 subsystem 609 (block 719) following the payment network standard for such contactless
9 payment transaction. The transfer of data occur using radio communication means. In an
10 example embodiment, the POS terminal device 502 displays a message to the user 101 to
11 "Remove card" (block 720) once the terminal as received all the card data it needed. The
12 user positions the mobile device 501 away from the POS terminal device (block 721). Other
13 computations and processes can occur within the mobile device 501 which completes the
14 mobile device's participation in the transaction process (block 722).

15 **[0081]** Although not shown in Fig. 7, the POS terminal device 502 will send the virtual
16 card details to the merchant acquirer, along with other transaction data (e.g. cost for
17 transaction, merchant ID, etc.). This information is eventually routed to the payment
18 gateway server 506, as identified by the first portion of digits of the virtual card's PAN. See
19 Fig. 8 for more details.

20 **[0082]** In an example embodiment, one PIN can be used for all the funding cards
21 associated with the mobile device. In another example embodiment, the registration process
22 can request a specific PIN for each funding card. In other words, if a user selects a different
23 funding card, the user will need to enter in a different PIN.

24 **[0083]** In a preferred example embodiment, the mobile device's payment application 620
25 does not verify the PIN. Instead, the PIN is indirectly or implicitly verified by the payment
26 gateway server 506 when verifying the virtual card data set. In other words, the payment
27 gateway server 506 uses the PIN that was stored at registration time to compute the virtual
28 card data set. If an incorrect PIN was provided by the user or an adversary during the
29 transaction, it will cause the computed virtual data to result in a different, incorrect, value
30 compared to the virtual card data computed by the payment gateway server. Once the
31 received virtual card data set is compared with the expected value by the server, an invalid
32 PIN entry can be detected. When the payment gateway server receives and verifies the
22259402.1

1 virtual card data set and detect an un-expected virtual card data set (e.g. due to the incorrect
2 PIN), then the payment authorization can be declined.

3 **[0084]** Turning to Fig. 8, example computer executable or processor implemented
4 instructions are provided for facilitating a transaction between a mobile device 501 and a
5 POS terminal device 502, including further details regarding the processing of data by the
6 merchant acquirer 103, payment gateway server 506 and funding card issuer server 104.

7 **[0085]** Blocks 701, 718, 719, 720, and 721 shown again to provide context for the
8 process. After the POS terminal device 502 receives the virtual card data from the mobile
9 device 501 (block 719), the POS terminal device 502 sends the virtual card data, the
10 transaction information (e.g. amount of payment), and other information (e.g. merchant ID) to
11 the merchant acquirer 103 (block 801). In an example embodiment, the data sent by the
12 POS terminal device 502 is in a standard payment authorization request format.

13 **[0086]** The merchant acquirer 103 sends the payment authorization request, which
14 includes at least the virtual card data (e.g. virtual card's Track Two data) and the payment
15 amount, to the payment gateway server 506 (block 802) via the payment network. The
16 acquirer 103 and the payment network are able to identify the payment gateway server 506
17 by the first portion of digits in the PAN of the virtual card. It can be appreciated the PAN is
18 part of the Track Two data.

19 **[0087]** The payment gateway server 506 validates the virtual card data (block 803). To
20 validate the virtual card data received from the mobile device (via the merchant acquirer
21 103), the payment gateway server computes the Track Two data set on its own. The Track
22 Two data computation includes the payment gateway server computing the discretionary
23 data using the PIN originally received and stored during registration by the user. In an
24 example embodiment, some of the Track Two data portions (like the PAN and expiry date)
25 were pre-computed and stored; these pre-computed data portions can be compared against
26 the received Track Two data. If the virtual card data is successfully validated (received card
27 Track Two data set matches computed card Track Two data set by the payment gateway
28 server), then the payment gateway server retrieves the funding card data that is associated
29 with the virtual card data card(block 804).

30 **[0088]** The validation of the virtual card data in block 803 can also include verifying if the
31 internal expiry date associated with the virtual card has passed or not. If the present date of
32 the validation is occurring before or on the internal expiry date, then the virtual card can be
22259402.1

1 considered validated and the transaction processing can continue. Otherwise, the virtual
2 card is deemed invalid and the payment authorisation request is denied.

3 **[0089]** If the virtual card is validated, the payment gateway server 506, playing a role
4 that is similar to a merchant at that point, sends a payment authorisation request to its own
5 acquirer, where the payment authorisation includes at least the funding card data and the
6 same payment amount received in block 802, to the funding card issuer server 104 (block
7 805) via the payment network. The funding card issuer 104 then receives the payment
8 authorisation request and process the transaction as standard. Once the transaction is
9 processed, the funding card issuer 104 sends a payment authorisation code to the payment
10 gateway server (block 806) via the payment network. The authorisation code response
11 includes whether the payment authorisation has been accepted or denied, a transaction
12 identification number, etc.

13 **[0090]** The payment gateway server 506 then sends a corresponding payment
14 authorisation response to the merchant acquirer 103 (block 807) via the payment network,
15 and the acquirer 103 sends the payment authorisation response to the POS terminal device
16 502 (block 808) also via the payment network. Although not shown, the payment gateway
17 server stores the payment authorisation code into the database 509. Although not shown as
18 well, the POS terminal device 502 may display a message to the end user that the payment
19 has been accepted or denied, according to the response.

20 **[0091]** In an example embodiment, the payment gateway server 506 may also send a
21 confirmation to the mobile device 501 indicating whether or not the payment was accepted
22 or denied (block 809). The indication can be sent as data specific to the payment application
23 620 on the mobile device, or as an email. After receiving such indication, the mobile device
24 501 can display a message to the user according to the indication.

25 **[0092]** In an example embodiment, after the payment gateway server 506 receives a
26 response that the payment has been processed (block 806), the payment gateway server
27 506 marks the virtual card as being used. In this way, if the same virtual card is used again
28 in a future transaction, the payment gateway server will decline the transaction. This is
29 because the virtual card is meant to be used only once.

30 **[0093]** Turning to Fig. 9a, an example embodiment of computer executable or processor
31 implemented instructions is provided for facilitating a transaction using a virtual card. In an
32 example embodiment, the instructions of Fig. 9a are an example detailed implementation of
22259402.1

1 blocks 710 and 711. In the example embodiment, the virtual card number (also called the
2 PAN of the virtual card), the expiry date of the virtual card and the other data are all
3 computed by the payment gateway server 506. Details explaining the flow of data between
4 the virtual card issuer server 401, the virtual card operator 402 and the virtual card acquirer
5 403 are also provided. The entities 401, 402, 403 can be part of the payment gateway
6 server 506. In another example embodiment, the entities 401, 402, 403 are separate
7 entities.

8 **[0094]** The operations in blocks 901 through 908 are part of the overall process for the
9 mobile device getting a virtual card (block 909).

10 **[0095]** The process in Fig. 9a assumes that the mobile device 501 has the payment
11 application 620 installed, and that the mobile device and the funding cards have been
12 registered with the payment gateway server 506. It also assumes that a transaction with the
13 merchant has already been initiated. For example, the POS terminal device 502 has
14 displayed the message "Tap to pay", and the user has already selected a funding card
15 through the payment application 620. These conditions are achieved using, for example,
16 blocks 701, 702, 703, 704, 705, 706 and 707.

17 **[0096]** After receiving a user input to select a funding card, the mobile device 501 sends
18 the transaction type (selected action in the application menu, in this case, "Pay with a virtual
19 card") and a funding card identifier. In a preferred example embodiment, the funding card
20 identifier is determined at the time of registration.

21 **[0097]** The virtual card operator 402 confirms that the selected funding card is valid
22 (block 902) by ensuring for example that the funding card identifier is linked with the
23 registered user and that the selected funding card has not expired since its registration.

24 **[0098]** The virtual card operator 402 sends a request to create a virtual card PAN to the
25 virtual card issuing server 401 (block 903). The request for the virtual card PAN includes the
26 last four digits of the funding card PAN and the expiry date of the funding card. The virtual
27 card issuing server creates a virtual card PAN which has the last four digits identical to the
28 last four digits of the funding card PAN. The virtual card also has the same expiry date and
29 the funding card. The virtual PAN is created by using the first 6 digits assigned to the virtual
30 card issuer, a unique, never used before, 8 digits long random number (to prevent collision),
31 and the check mod10 digit.

22259402.1

1 **[0099]** After the virtual card PAN is computed, the virtual card issuing server 401 sends
 2 this information, in a response, to the virtual card operator 402 (block 904). The virtual card
 3 operator will generate portions of the Track Two data such as the PAN, the card expiry,
 4 service code, etc (block 905). In an example embodiment, the discretionary data (which is
 5 part of the Track Two data) is not computed by the payment gateway server 506 at this time.
 6 The virtual card operator 402 associates the virtual card PAN with the funding card PAN
 7 (block 906) and stores it in the database 509 (not shown). It also adds an internal expiry
 8 date to the virtual card records in the database. In an example embodiment, the internal
 9 expiry date is computed to be some predetermined time period after the time that the virtual
 10 card is created. For example, the internal expiry date is 48 hours from the date and time that
 11 the virtual card is created. The virtual card operator 402 then encrypts and sends the virtual
 12 card data (e.g. virtual card PAN, external expiry date, etc.) required by the mobile application
 13 to generate the final virtual card data set card(block 907). The result forms an encrypted
 14 virtual card payload that is sent to the mobile device 501 (block 908).

15 **[00100]** The mobile device 501 receives the PIN from the user (block 925). The mobile
 16 device 501 then uses the PIN to compute the discretionary data (which is a portion of the
 17 virtual card's Track Two data). The computation of the discretionary data can involve a
 18 dynamic value which changes over time, or with each transaction, thereby making the
 19 discretionary data the dynamic data. The data received from the virtual card payload (e.g.
 20 PAN, expiry date, and other data) and the discretionary data form the complete Track Two
 21 data set of the virtual card (block 926).

22 **[00101]** Turning to Fig. 9b, an example embodiment of computer executable or processor
 23 implemented instructions is provided for validating and authorizing a transaction using a
 24 virtual card. The process in Fig. 9b assumes that a virtual card was already created (e.g. as
 25 per Fig. 9a) and focuses on the processing for verifying the virtual card. In an example
 26 embodiment, the instructions of Fig. 9b are an example detailed implementation of blocks
 27 802 to 807. The entities 401, 402, 403 can be part of the payment gateway server 506. In
 28 another example embodiment, the entities 401, 402, 403 are separate entities.

29 **[00102]** The processes described with respect to blocks 910 through 923 are part of the
 30 overall authorization process of the virtual card (block 924).

31 **[00103]** The flow starts after the mobile device 501 has received the encrypted virtual
 32 card payload, decrypted the virtual card payload, captured via the GUI the user's PIN, and
 22259402.1

1 computed the complete virtual card data set (e.g. the complete Track Two data set). At that
 2 point, the mobile application can send the virtual card data set to the POS terminal device
 3 502 via the NFC subsystem 609 (block 910). The POS terminal device 502 sends the virtual
 4 card data, along with transaction data, to the merchant acquirer 103 (block 911). The
 5 merchant acquirer 103 sends a payment authorisation request, which includes the virtual
 6 card data and the amount of payment, to the virtual card issuing server 401 (block 913) via
 7 the payment network.

8 **[00104]** The virtual card issuing server 401 sends the virtual card data to the virtual card
 9 operator 402 (block 914). The virtual card operator 402 uses the virtual card data, such as
 10 the virtual card PAN, to retrieve the associated user profile and funding card data (e.g. PAN
 11 and expiry date) (block 915). The virtual card operator 402 then verifies the received virtual
 12 card data, for example, by comparing data with the server computed version of the expected
 13 card data. The validation process is similar to operation of block 803.

14 **[00105]** If the virtual card data is successfully verified, then the virtual card operator 402
 15 uses the funding card details associated with the virtual card transaction and send a
 16 payment authorisation request, which includes the funding card data and the corresponding
 17 payment amount, to the virtual card acquirer 403. The virtual card acquirer 403 forwards the
 18 details to the funding card issuing server 104 (block 917) via the payment network.

19 **[00106]** The funding card issuing server 104 receives and verifies the funding card data. If
 20 the funding card data is successfully verified, the funding card issuing server sends a
 21 payment authorisation code response to the virtual card acquirer 403, which forwards the
 22 same to the virtual card operator 402 (block 918), all via the payment network. The payment
 23 authorisation code response indicates whether the funding card data is accepted or denied,
 24 and a transaction identification number, etc.

25 **[00107]** The virtual card operator 402 sends a corresponding validation authorisation
 26 code response to the virtual card issuing server 401 (block 919). The virtual card operator
 27 402 and the virtual card issuing server 401 marks the virtual card as being used (block 920
 28 and block 921). In this way, if the same virtual card is used again at a later time, the virtual
 29 card operator 402 or the virtual card issuing server 401 will be able to detect a possible
 30 fraud.

31 **[00108]** The virtual card issuing server 401 sends the payment authorisation code
 32 response for the virtual card to the virtual card operator 402 (block 922). This payment
 22259402.1

1 authorisation code response for the virtual card is then forwarded to the merchant acquirer
2 103 (block 923) via the payment network.

3 **[00109]** Turning to Fig. 10, an example embodiment of computer executable or processor
4 implemented instructions are provided for registering a funding card and mobile device with
5 payment gateway server. Such an embodiment can be used in combination with the
6 principles described above. The mobile device 501 sends a registration request to the
7 payment gateway server 506 (block 1001). The request includes the mobile device's device
8 ID, a user provided PIN, and funding card details. After receiving the request, the payment
9 gateway server sends a registration response to the mobile device (block 1002). The
10 response include a certificate, an Application Transaction Counter (ATC), a key value (called
11 Kpan) for generating a PAN, a secure element key value (called Ksec), and a funding card
12 identifier that identifies each registered funding card(s). The certificate is a client certificate
13 for the mobile device 501 and, in an example embodiment, it is configured to according to
14 the RSA algorithm and has a 2048 bit length. The ATC is a counter that is initially set to a
15 random value between "0" or "1000" and increments with each transaction. The initialized
16 ATC value is a random value to prevent adversaries from predicting the ATC values. Copies
17 of the ATC are stored and synchronise on both the payment gateway server and the mobile
18 device. In an example embodiment, the ATC is a 10 digit value. In an example
19 embodiment, the Kpan is 128 bits long, and the Ksec is 128 bits long.

20 **[00110]** The payment gateway server 506 stores the data from the registration request
21 and the registration response in association with each other (block 1003). The mobile
22 device 501 also stores the data from the registration response (block 1004).

23 **[00111]** Turning to Fig. 11, an example embodiment of computer executable or processor
24 implemented instructions are provided for computing virtual card data. Such an embodiment
25 can be used in combination with the principles described above for computing virtual card
26 data. The mobile device 501 sends a virtual card request to the payment gateway server
27 506 (block 1101). This is similar to block 901 in Fig. 9a. The request includes the device ID,
28 the identifier of the selected funding card, the Certificate and the stored Kpan. The stored
29 Kpan can be from registration or from the previous transaction.

30 **[00112]** The payment gateway server 506 determines if the received Kpan and Certificate
31 match the Kpan and Certificate stored on the payment gateway server (block 1102)
32 associated with the DeviceID stored in the payment gateway server database. If so, then
22259402.1

1 the payment gateway server 506 sends a virtual card response to the mobile device 501 that
 2 includes a new Kpan (block 1103). A new Kpan is used to generate a different PAN from the
 3 previous transaction, and to also prevent against replay attacks. The payment gateway
 4 server 506 computes the PAN for the virtual card using the new Kpan (block 1104). This
 5 new Kpan and the pre-computed PAN are stored by the payment gateway server 506 for
 6 later use (block 1105).

7 **[00113]** After the mobile device receives the new Kpan, it replaces the stored Kpan with
 8 the new Kpan (block 1106). It uses the new Kpan to compute a PAN for the virtual card, in
 9 the same way the payment gateway server computed the PAN (block 1107). If the
 10 conditions and data are correct, although the mobile device computes the PAN
 11 independently of the payment gateway server, the PAN computed by the mobile device
 12 should be identical to the PAN computed by the payment gateway server.

13 **[00114]** The mobile device then computes the discretionary data (which is part of the
 14 Track Two data) using the PIN, the Ksec, the PAN, the Certificate, and the ATC (block
 15 1108). The ATC and the PAN keep changing with each transaction, which causes the
 16 discretionary data to be dynamic data.

17 **[00115]** The mobile device increments the ATC by 1 (block 1109).

18 **[00116]** In particular, the PAN is computed by the mobile device and the payment
 19 gateway server according to the following:

20
$$\text{PAN} = \text{BIN}(6) + \text{SHA256}[\text{Kpan}](8) + \text{Luhn}(1) + \text{Reserved}(4)$$

21

22 wherein

23 BIN(6) is a 6-digit binary number to identify the payment gateway server;

24 SHA256[Kpan](8) is an 8-digit number generated by taking the sha256 of the Kpan value,
 25 further including converting the sha256 value to decimal and truncating to eight digits from
 26 the hash value;

27 Luhn(1) is a single digit used to ensure the virtual card always passes LUHN algorithm;

28 and

29 Reserved(4) is a 4-digit number that is the same as the last 4 digits of the funding card PAN.

22259402.1

1

2 **[00117]** The above values are concatenated together to form the PAN. The symbol "+" in
3 the above computation refers to the concatenation operation.

4 **[00118]** In an example embodiment of computing the PAN, the other values may be
5 known or computed first, and the LUHN value is computed last. For example, an
6 intermediate form to the PAN is 66666612345678X1111, whereby X represents Luhn(1). In
7 other words, the BIN(6) = 666666; SHA256[Kpan](8) = 12345678; and Reserved(4) = 1111.

8 **[00119]** Different LUHN algorithms can be used to solve for X, so as to ensure the PAN
9 satisfies the LUHN algorithm. For example, an example embodiment algorithm includes
10 (step 1) doubling the value of every other digit, starting from the right-most digit; (step 2)
11 summing the all the individual digits including the digit X from step 1; and (step 3) solving the
12 expression $\text{mod}10(\text{sum total from step 2})=0$ for the digit X. In this example embodiment, the
13 solution to the equation is $X = 9$. Other LUHN algorithms can be used.

14 **[00120]** In an example embodiment, any digit before the last 4 digits of the PAN that
15 satisfies the LUHN criteria can be used.

16 **[00121]** The discretionary data is computed using the following:

17 Discretionary data = $\text{HMAC\_SHA256}[\text{Ksec}+\text{PIN}, \text{M}](10)$

18

19 where M=concatenation of (PAN, Certificate ID, ATC)

20 **[00122]** Truncation is performed by encoding the SHA results in decimal, then taking the
21 left most digits. It can be appreciated that SHA256 is a known cryptographic hash function,
22 and HMAC is a hash based message authentication code involving a cryptographic hash
23 function in combination with a secret cryptographic key. The secret cryptographic key of the
24 HMAC function is the concatenated values of Ksec and PIN. The message M is the
25 concatenated values of PAN, Certificate ID and ATC. The Certificate ID is from the
26 Certificate.

27 **[00123]** Turning to Fig. 12, an example embodiment of computer executable or processor
28 implemented instructions are provided for verifying virtual card data, and in particular the
29 verification by the payment gateway server of the discretionary data (e.g. the dynamic data).
30 Such an embodiment can be used in combination with the principles described above for
22259402.1

1 verifying the virtual card data. This can be a continuation of the process described in Fig.
2 11.

3 **[00124]** The mobile device 501 sends a virtual card to the merchant POS terminal 502,
4 which then creates and sends a payment authorisation request to the merchant acquirer 103
5 (block 1201). The payment authorisation request is eventually received by the payment
6 gateway server 506 (block 1202). The payment authorisation request includes, among other
7 things, the PAN and the discretionary data of the virtual card, as computed by the mobile
8 device 501.

9 **[00125]** After receiving the request, the payment gateway server 506 uses the PAN to find
10 the relevant stored associated data (e.g. Kpan, Certificate, Ksec, funding card identifier,
11 device ID, and other user data). The relevant stored associated data is identified by the
12 precomputed PAN, which acts as an index to search into the server database. In other
13 words, the received PAN is compared with a number of precomputed PANs, and if a match
14 is found with a given precomputed PAN, then the stored data associated with that given
15 precomputed PAN is considered the relevant stored associated data.

16 **[00126]** The payment gateway server 506 uses the relevant stored associated data to
17 computes its own discretionary data. This can include using the PIN (received during the
18 user registration process), the stored Ksec, the Certificate ID from the stored Certificate, and
19 an incremented value of the ATC (block 1203).

20 **[00127]** The computation of the payment gateway's own discretionary data uses the
21 following:

22 Discretionary data = $\text{HMAC\_SHA256}[\text{Ksec} + \text{PIN}, \text{M}](10)$

23
24 where M=concatenation of (PAN, Certificate ID, (ATC+1))

25 **[00128]** The ATC+1 represents the incremented ATC value.

26 **[00129]** The payment gateway server 506 determines if its own discretionary data is equal
27 to the received discretionary data (block 1204). If they are not equal, the payment gateway
28 server further increments ATC by "1" and re-computes its own discretionary data (block
29 1205). The process returns to block 1204 to check if the discretionary data sets are equal.
30 The process involving block 1205 can be repeated, such so that each time the ATC value is

22259402.1

1 further incremented by 1. This can be done up to a certain number of times (e.g. 10 times),
 2 after which the transaction process will be stopped.

3 **[00130]** If the range of ATC values (e.g. between ATC+1 and ATC+10) does not generate
 4 an identical discretionary data set, then the verification is unsuccessful. The range is to
 5 account for the possibility that the mobile device's ATC counter may have incremented
 6 without the payment gateway server's knowledge. Therefore, a buffer or range of ATC
 7 values is used.

8 **[00131]** If the payment gateway server's own discretionary data is the same as the
 9 received discretionary data, then the payment gateway server replaces the previous ATC
 10 value with the currently incremented ATC value that is used in the computation of its own
 11 discretionary data. In this way, the ATC value stored on the payment gateway server should
 12 now be equal to the ATC value stored on the mobile device.

13 **[00132]** It is noted that if the PIN used by the mobile device to compute the discretionary
 14 data is incorrectly entered by the user (e.g. is not the same as the PIN provided at
 15 registration), then the discretionary data from the mobile device will not equal the payment
 16 gateway server's own discretionary data. This is because its own discretionary data is
 17 computed using the PIN provided at registration. In this way, the PIN provided by the user
 18 into the mobile device during each transaction is implicitly verified by the payment gateway
 19 server.

20 **[00133]** Continuing with Fig. 12, after the data is successfully verified or not, the payment
 21 gateway server eventually respond to the merchant by sending an authorisation response to
 22 the merchant (103, 502) (block 1207). To further clarify, if the discretionary data verification
 23 was not successful (as per block 1205 after a certain of iterations has been reached), then a
 24 negative payment authorisation response is sent to the merchant (block 1207).

25 **[00134]** The methods and systems described herein do not rely on having a software
 26 payment card stored on the secure element on the mobile device to facilitate contactless
 27 payment transaction. This is because the PAN, and the other card data elements of the
 28 funding card are not stored on in the mobile device. Instead the PAN and expiry date of the
 29 funding card is stored on a secure server (e.g. the payment gateway server). Furthermore,
 30 the virtual card data is created on a secure server, and is encrypted for a particular mobile
 31 device. Furthermore, the discretionary data in the Track Two data is a function of the PIN
 32 which re-entered with each payment. The virtual card data is only sent at the time of making
 22259402.1

1 the payment, and can only be used once. In addition, only the server can validate the virtual
2 card data. In other words, there is nothing for an adversary to access via brute force on the
3 mobile device without being detected.

4 **[00135]** Furthermore, the virtual card encrypted payload is not reusable. This is because
5 as soon as the card is used once, it cannot be used again by any other party.

6 **[00136]** It is also noted that the one-time virtual card is created just-in-time, ahead of the
7 payment authorization from merchant.

8 **[00137]** Furthermore, the funding card details are never provided to the merchant, since
9 the virtual card details are provided instead.

10 **[00138]** The methods and systems described herein are fully compatible with standard 4-
11 party card payment model and systems. There is no system change required by the issuer
12 of the funding card, merchant, acquirer or payment network.

13 **[00139]** The methods and systems described herein also allow for the use of a "single
14 application" to be used to support one or many virtual cards as well as one or many funding
15 cards. This reduces the required storage on the secure element (and generally storage on
16 the mobile device) so as not to limit the number of cards a customer can load into a secure
17 element.

18 **[00140]** It can be appreciated that the methods and systems described herein enable the
19 cardholder (e.g. end user) to register any funding card they want into their NFC-enabled
20 mobile device, independently of the card issuer having the infrastructure or a commercial
21 relationship with a particular mobile operator.

22 **[00141]** The methods and systems described herein can be used with any type of funding
23 card. The funding cards and the mobile device are pre-registered with the payment gateway
24 server 506. This registration is independent of any particular mobile phone carrier and any
25 particular funding card issuer. As a result, no commercial agreement and additional
26 infrastructure are required between a mobile phone carrier and a funding card issuer to
27 facilitate contactless payment using a mobile device. This in turn reduces the cost incurred
28 by the funding card issuer to issue software payment cards to mobile devices.

29 **[00142]** The steps or operations in the flow charts described herein are just for example.
30 There may be many variations to these steps or operations without departing from the spirit

22259402.1

1 of the invention. For instance, the steps may be performed in a differing order, or steps may
2 be added, deleted, or modified.

3 **[00143]** While the basic principles of this invention has been herein illustrated along with
4 the example embodiments shown, it will be appreciated by those skilled in the art that
5 variations in the disclosed arrangement, both as to its details and the organization of such
6 details, may be made without departing from the spirit and scope thereof. Accordingly, it is
7 intended that the foregoing disclosure and the showings made in the drawings will be
8 considered only as illustrative of the principles of the invention, and not construed in a
9 limiting sense.

22259402.1

- 31 -

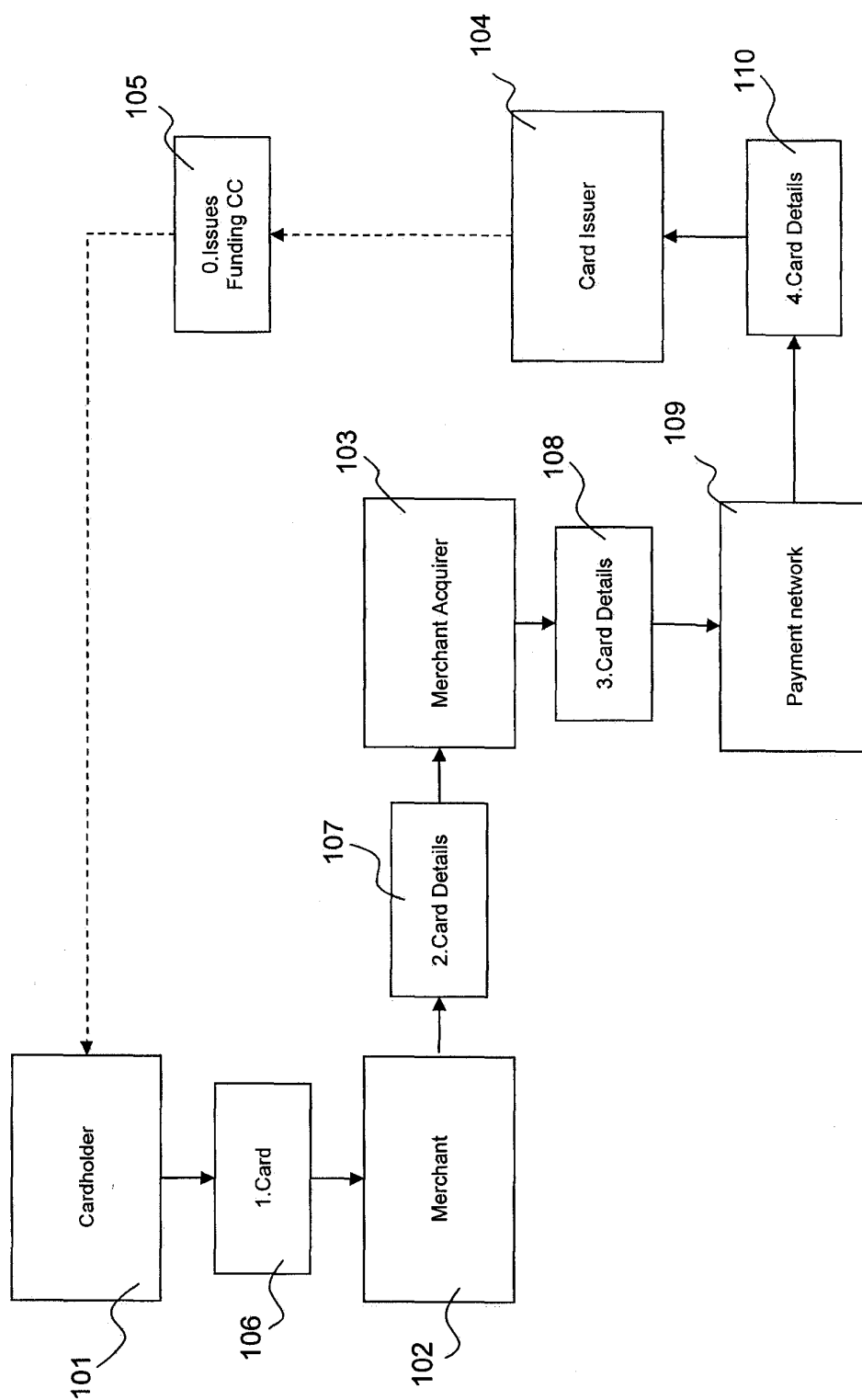
Claims:

1. A method performed by a server for facilitating payment, comprising:
 - receiving a message from a mobile device identifying a funding card;
 - searching a database of multiple cards associated with the mobile device for a funding card number associated with the identified funding card;
 - computing data for a virtual card, the data comprising a card number and an expiry date;
 - storing the data for the virtual card number in association with the funding card number;
 - sending the data for the virtual card to the mobile device;
 - computing the card details using the user PIN as input;
 - receiving a first payment authorisation request from a merchant acquirer, the request comprising the data for the virtual card and a requested payment amount;
 - retrieving the funding card number based on the data for the virtual card;
 - sending a second payment authorisation request to a funding card issuer, the request comprising the funding card number and the requested payment amount;
 - receiving a payment authorisation response from the funding card issuer; and
 - sending the payment authorisation response to the merchant acquirer.

ABSTRACT

Systems and methods are provided for facilitating contactless payment using cloud-based wallet containing payment credentials (e.g. Visa, Mastercard, American Express) using a near field communication (NFC)-capable device and payment gateway servers. A user can use their existing payment card, herein referred to as a funding card, for contactless payments. A second payment card, herein referred to as a virtual card, is generated. The virtual card is associated with the funding card on a payment gateway server. The virtual card is used on a NFC-enabled mobile device. When a payment is initiated, the virtual card data is sent through the NFC system from a point of sale terminal. This information is sent to the payment gateway server, which retrieves the funding card to make the payment. The funding card, not the virtual card, is used to transfer the money to make payment.

22259402.1

**FIG. 1 (Prior Art)**

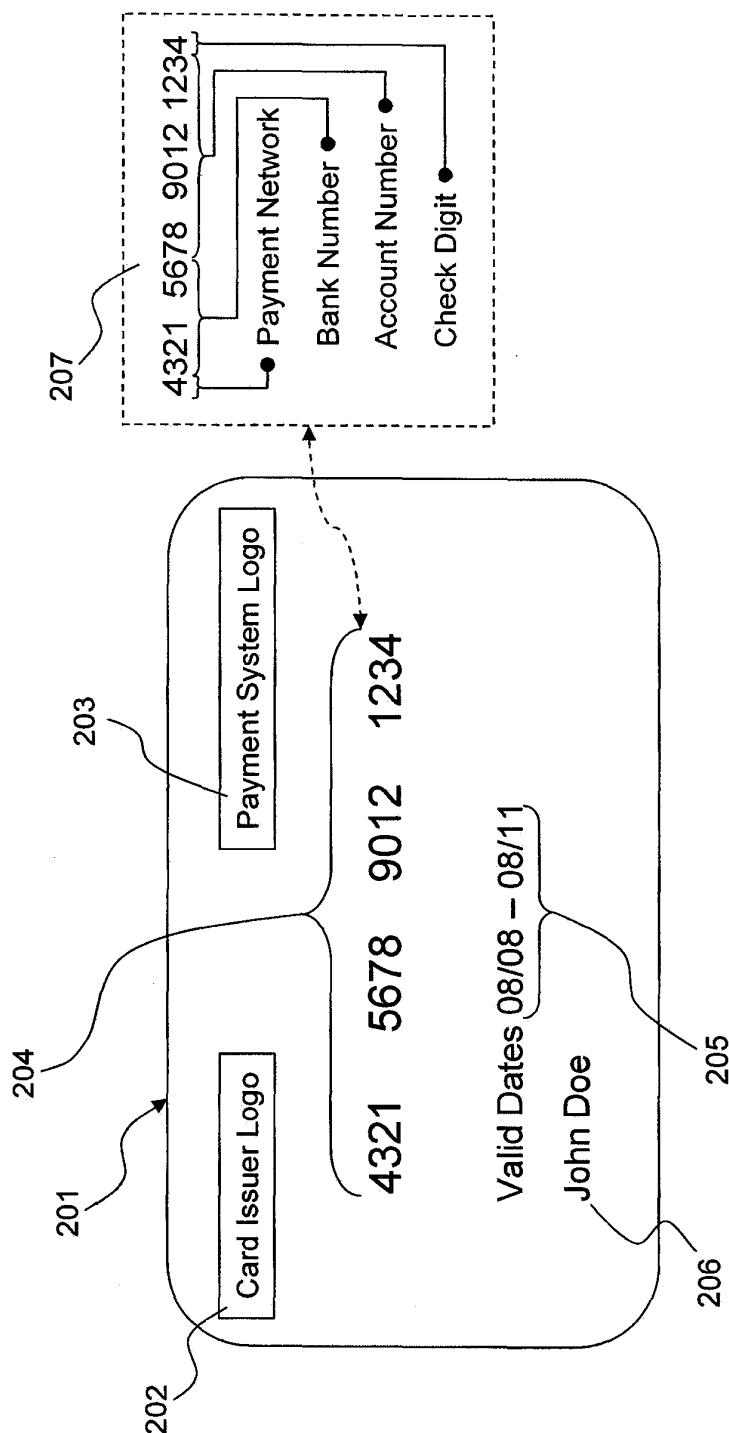
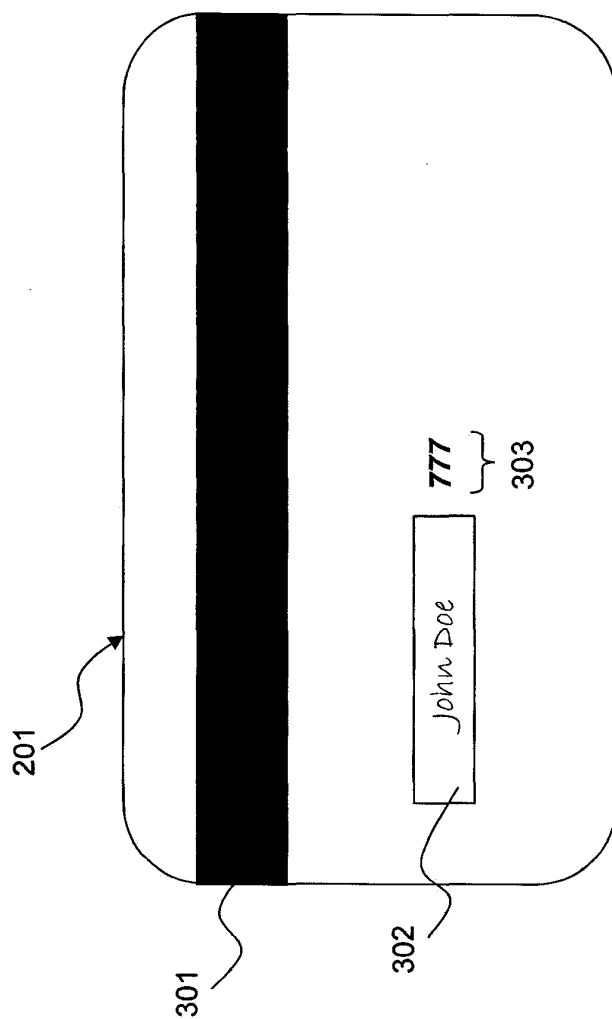


FIG. 2 (Prior Art)

**FIG. 3 (Prior Art)**

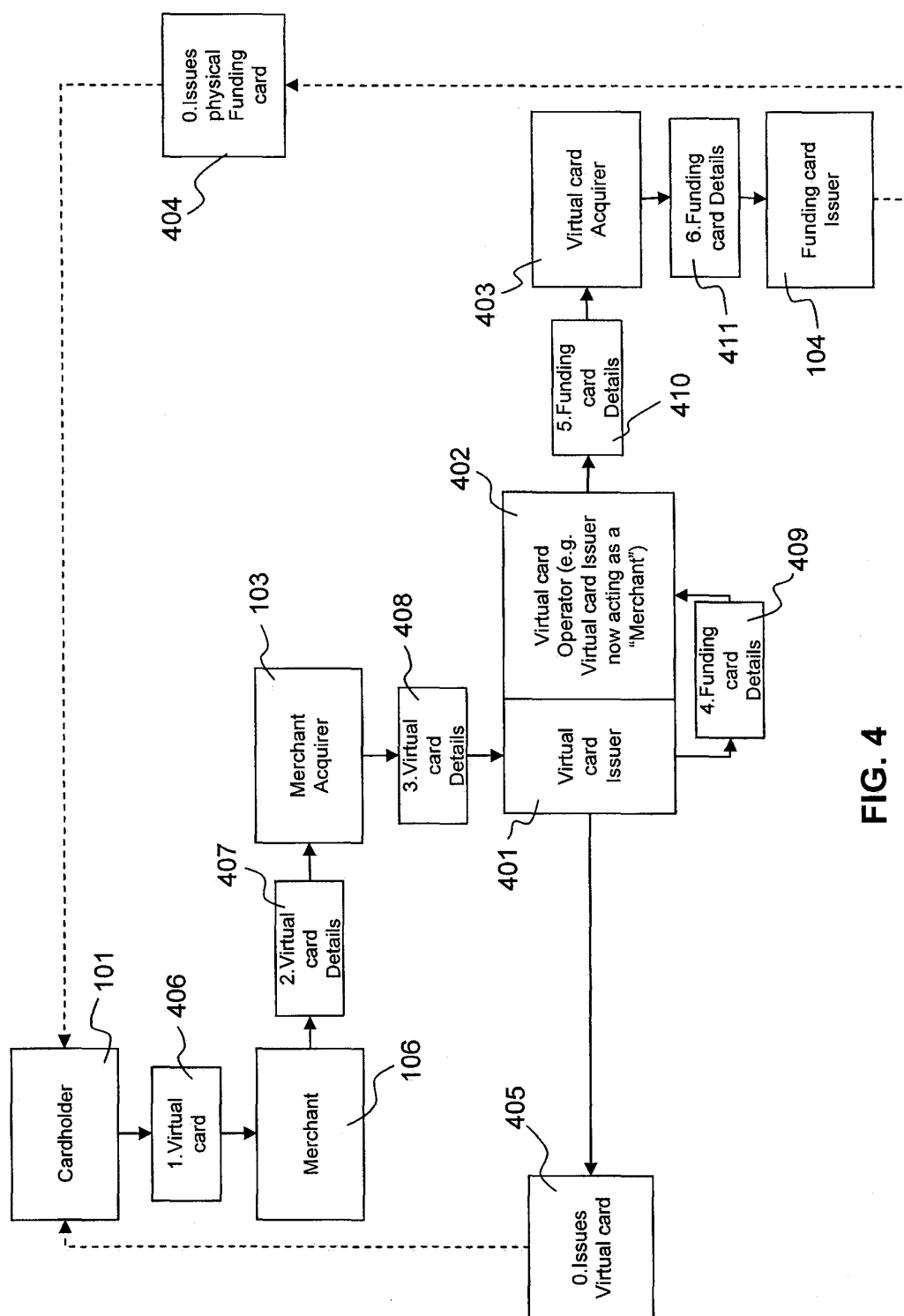


FIG. 4

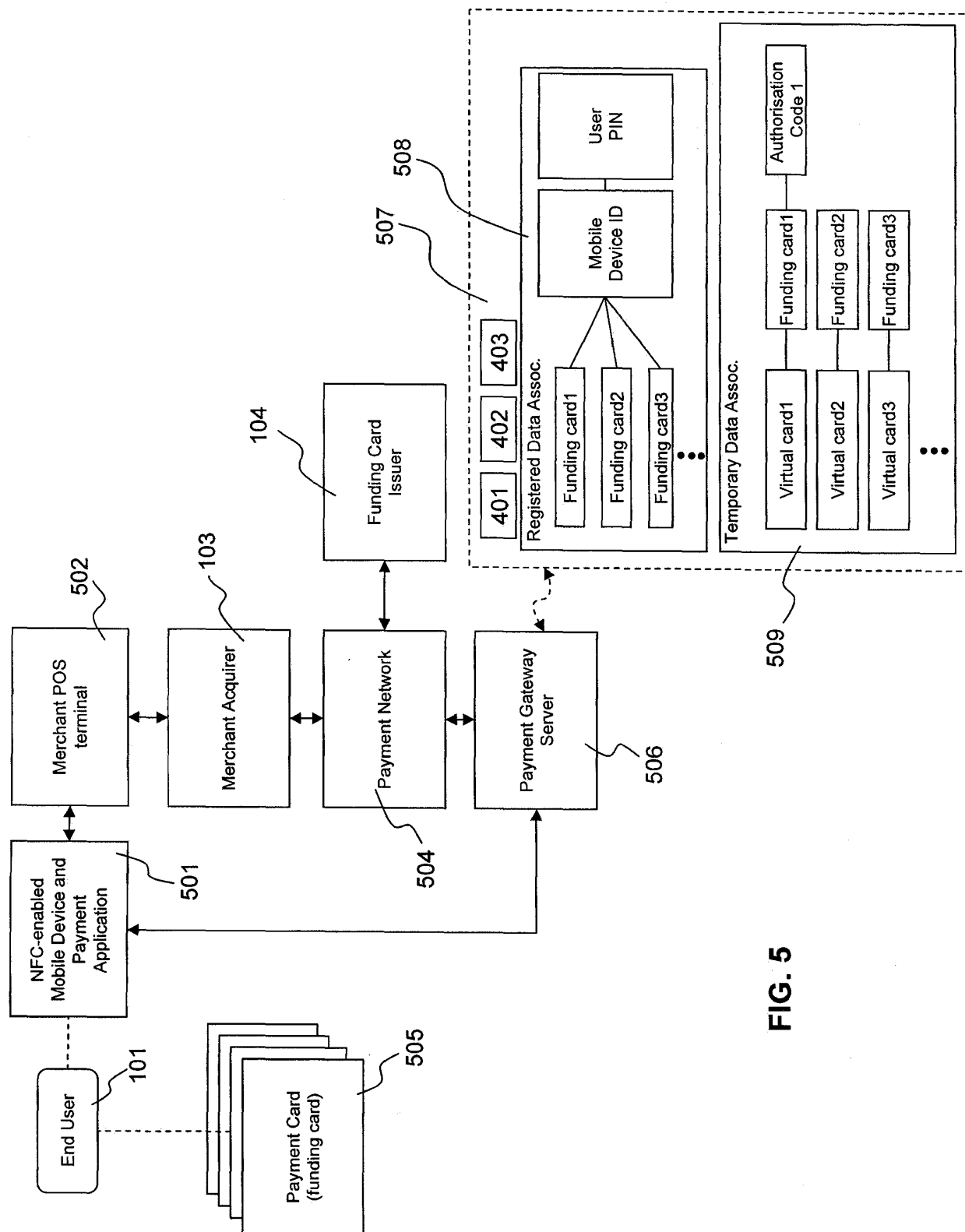


FIG. 5

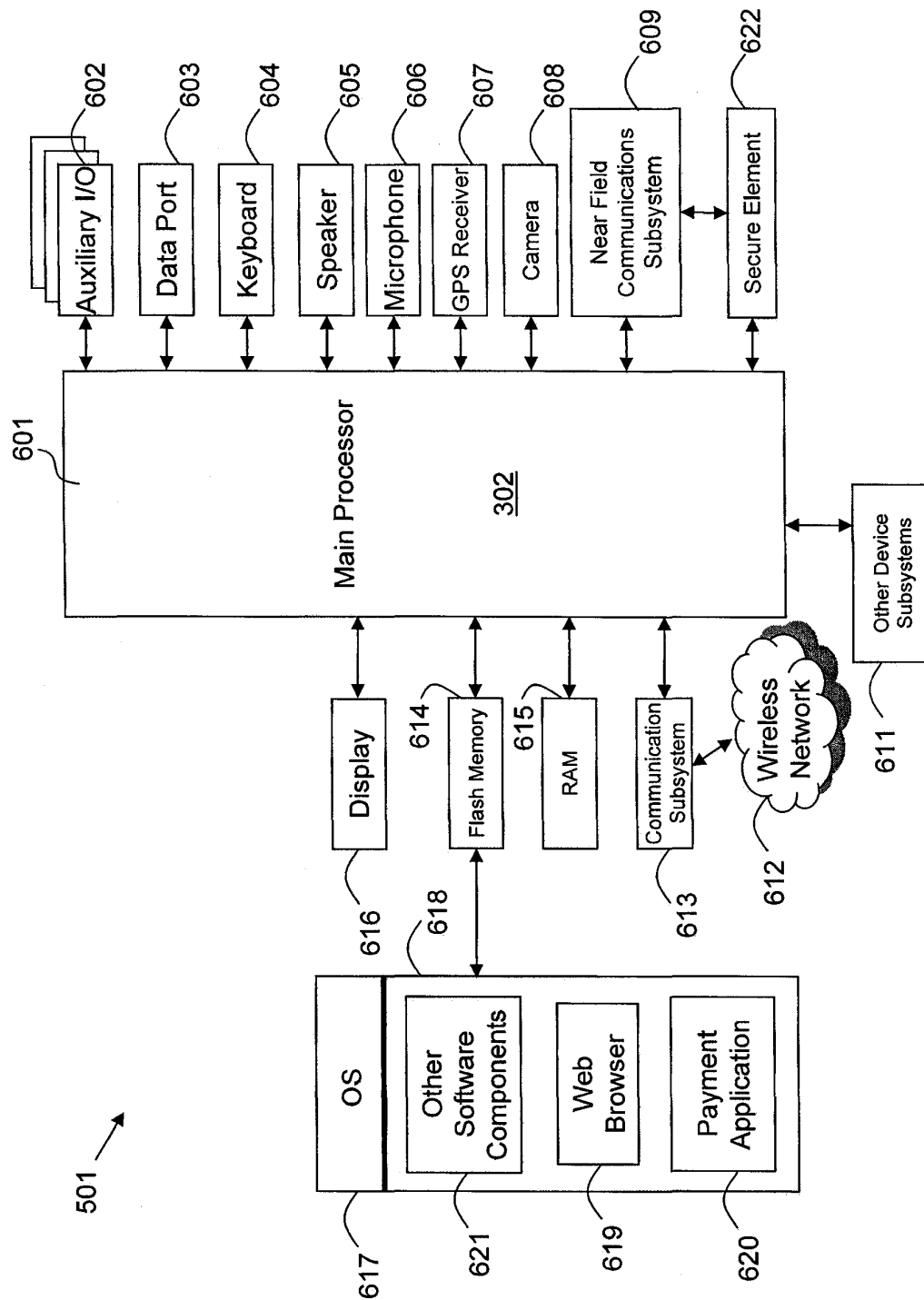


FIG. 6

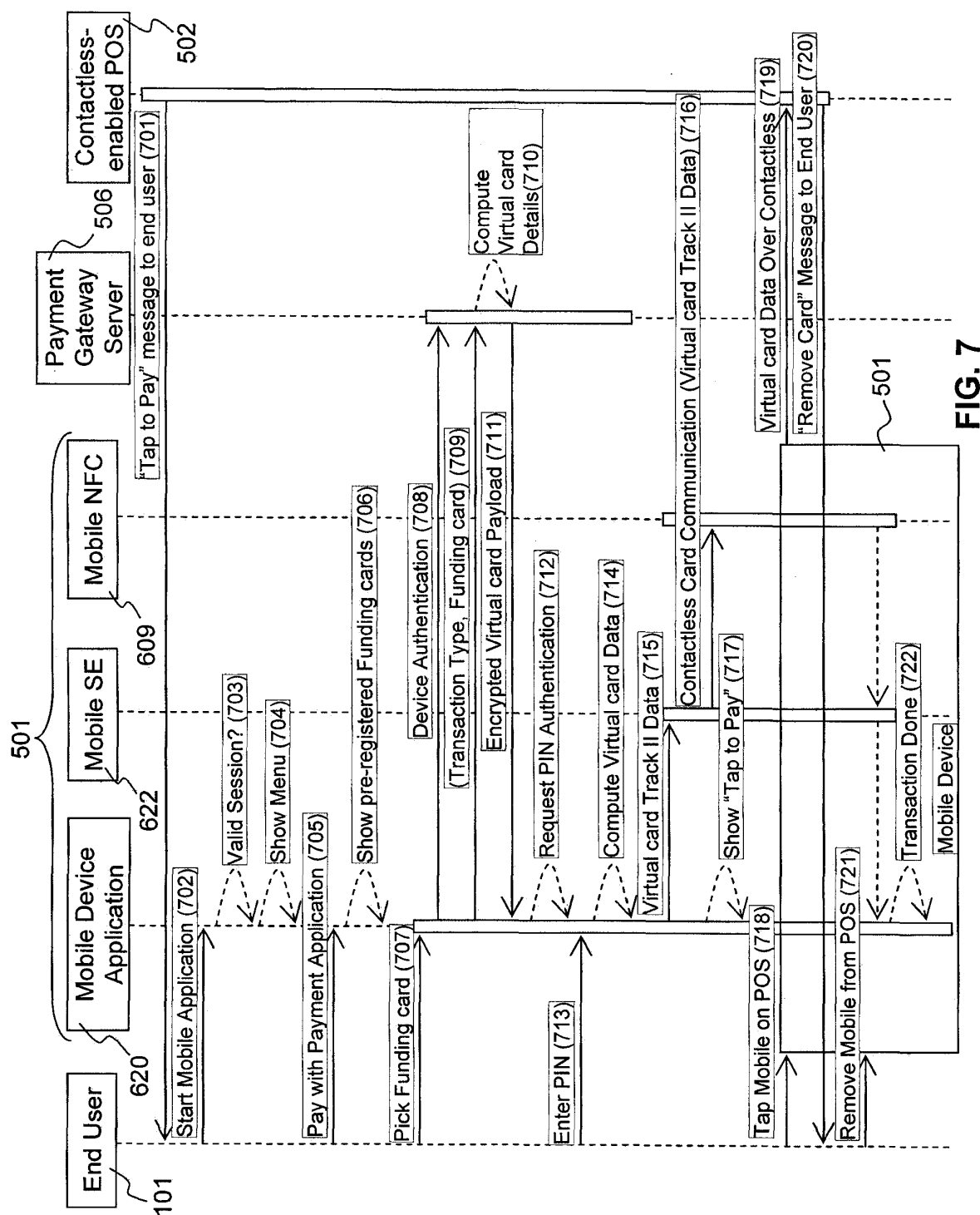


FIG. 7

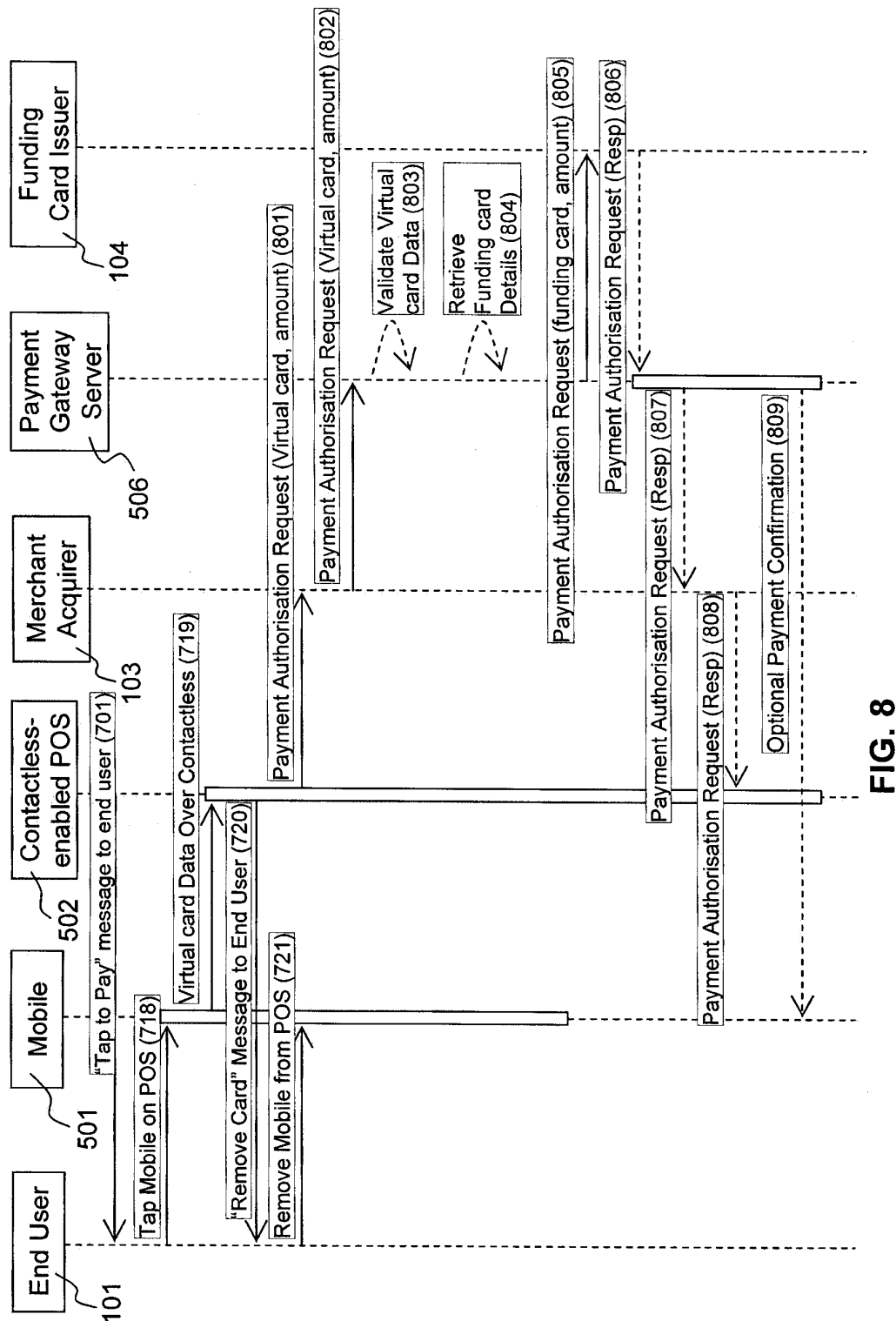


FIG. 8

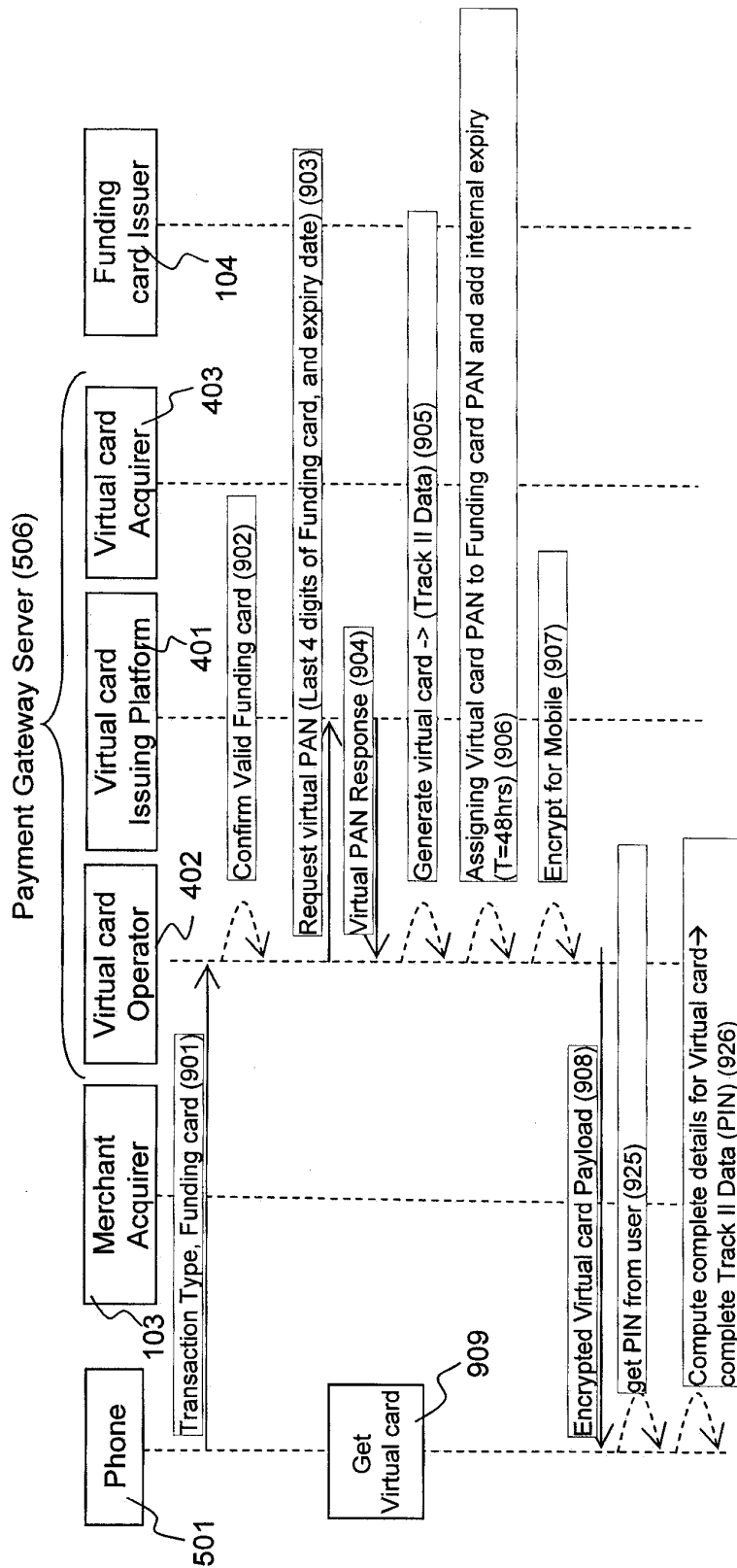


FIG. 9a

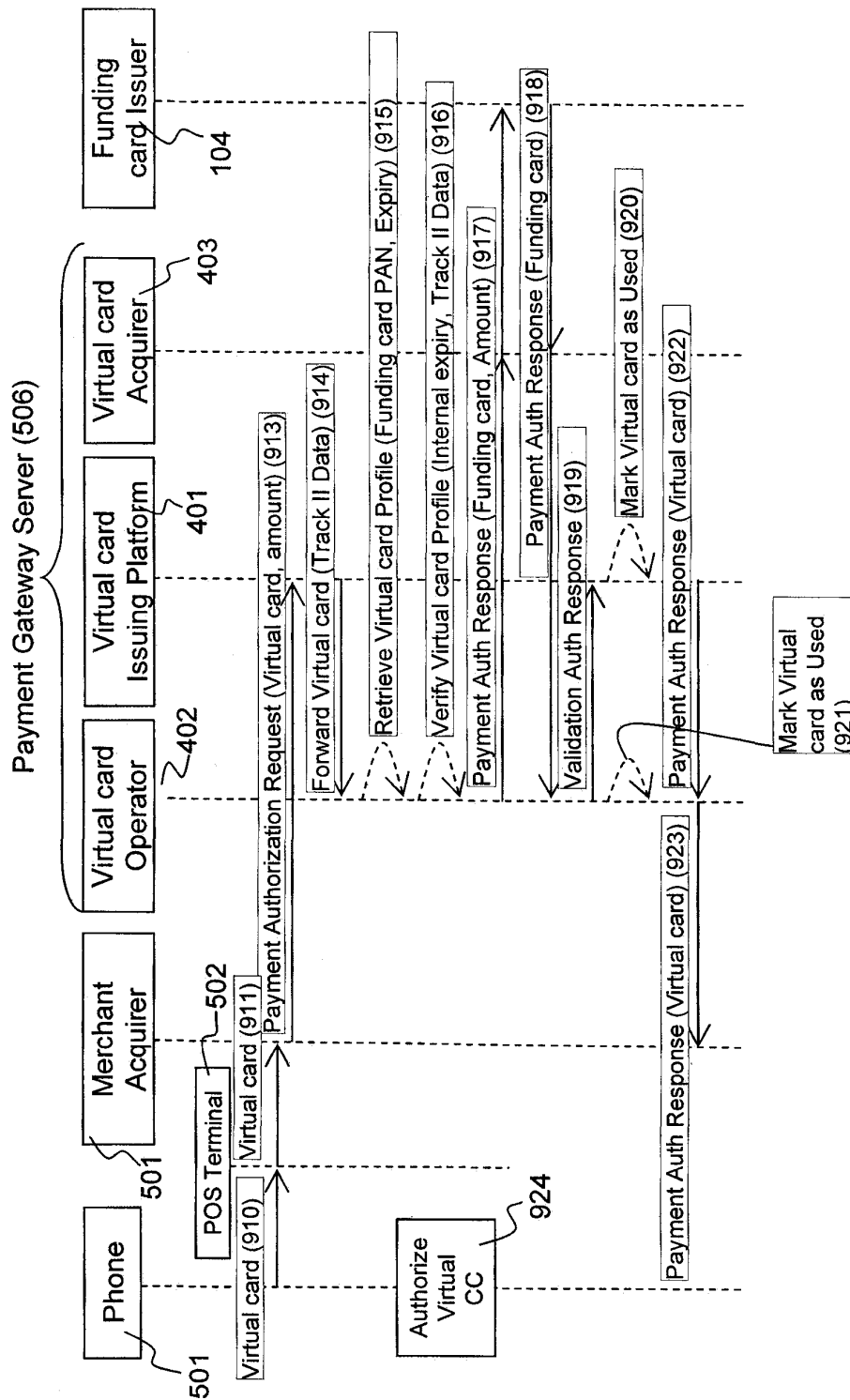
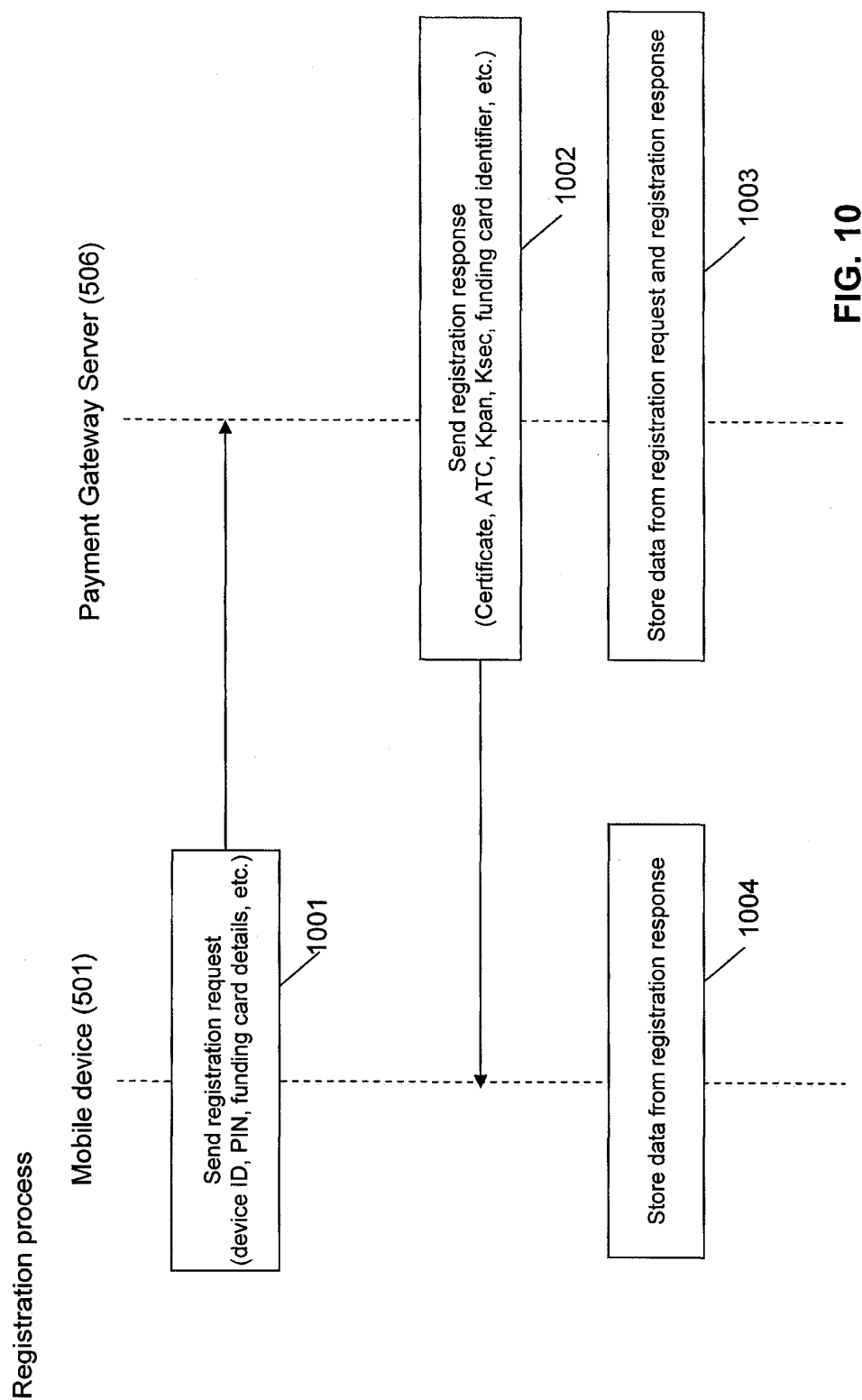


FIG. 9b



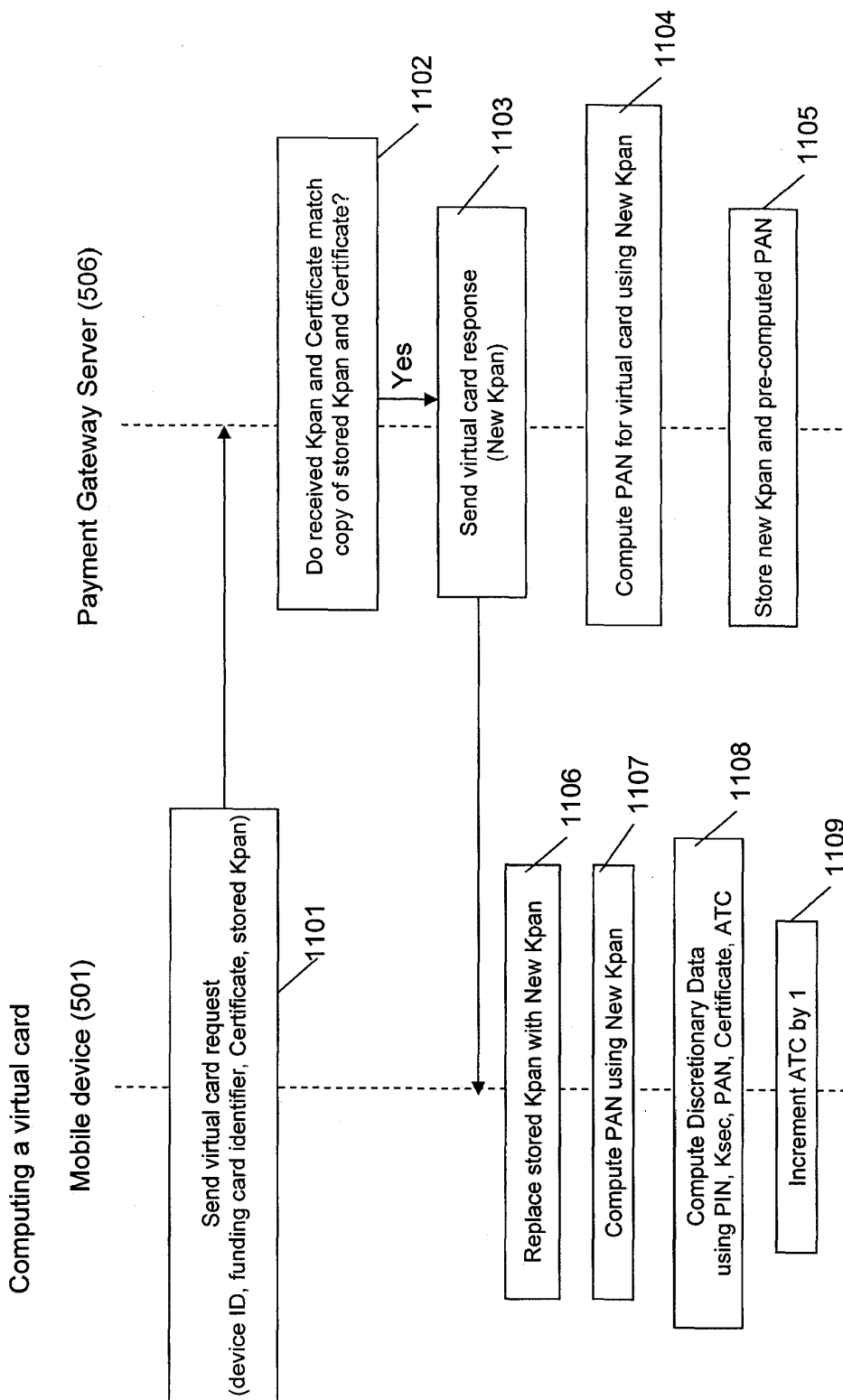


FIG. 11

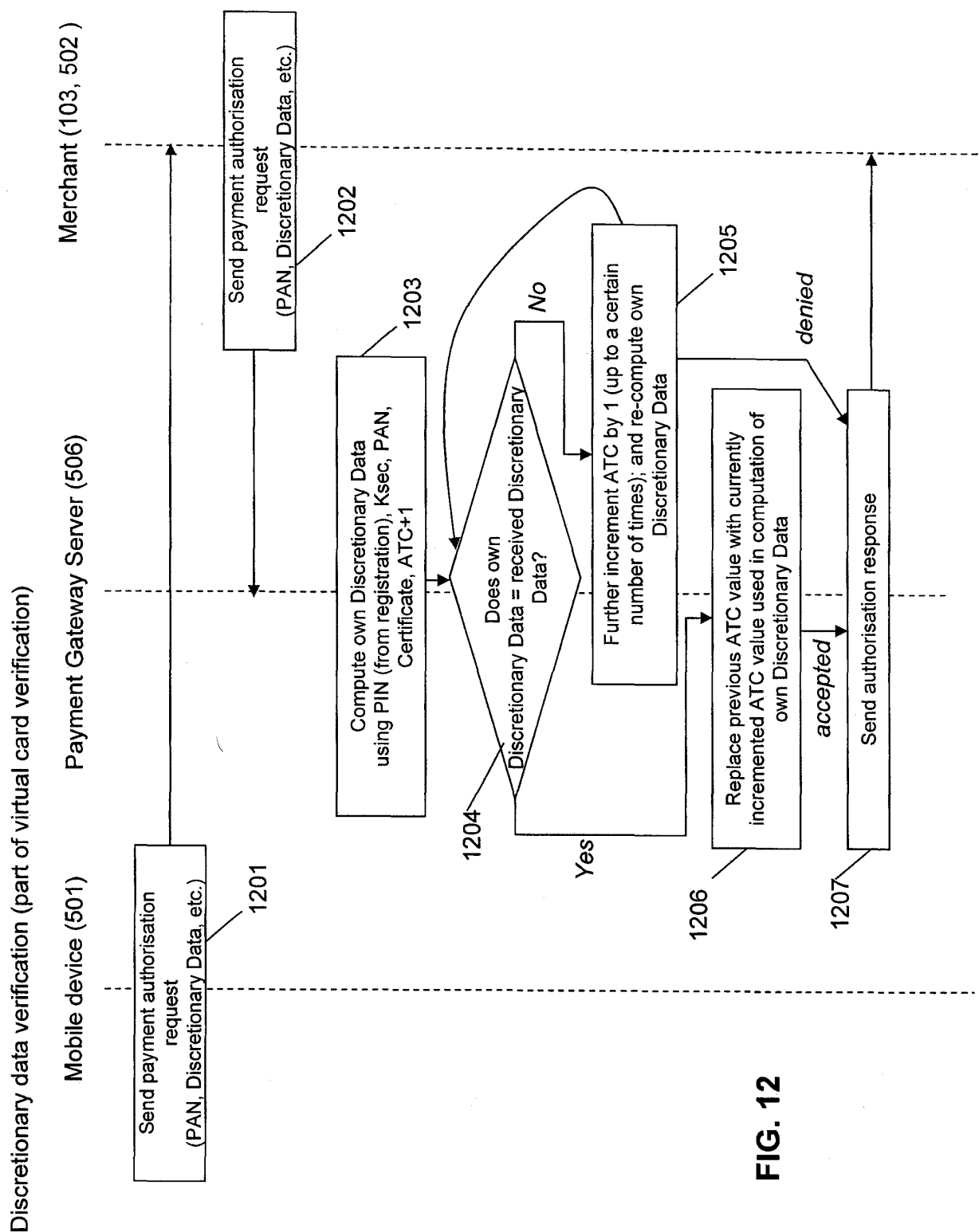


FIG. 12