VeriTrans4G Development Guide Separate Volume

LINE Pay User Guide

Ver. 1.0.0 beta(April 2017~)

Table of Contents

Chapter 1 Contents of this Guide

1-1 Contents of this Guide

This guide is intended for developers integrating their website with VeriTrans4G's LINE Pay service offered by VeriTrans Inc.

1-2 Copyright and Contact Details

[Contact Details]

Technical Support, VeriTrans Inc.

E-mail: tech-support@veritrans.jp

1-3 Revision History

2017/04: Released Ver. 1.0.0

* The following are the updates from "VeriTrans3G Development Guide Separate Volume LINE Pay User Guide" Ver 1.0.3.
Deleted "6-3 Mail Sent to Consumer from LINE”.

Chapter 2 LINE Pay Service Overview

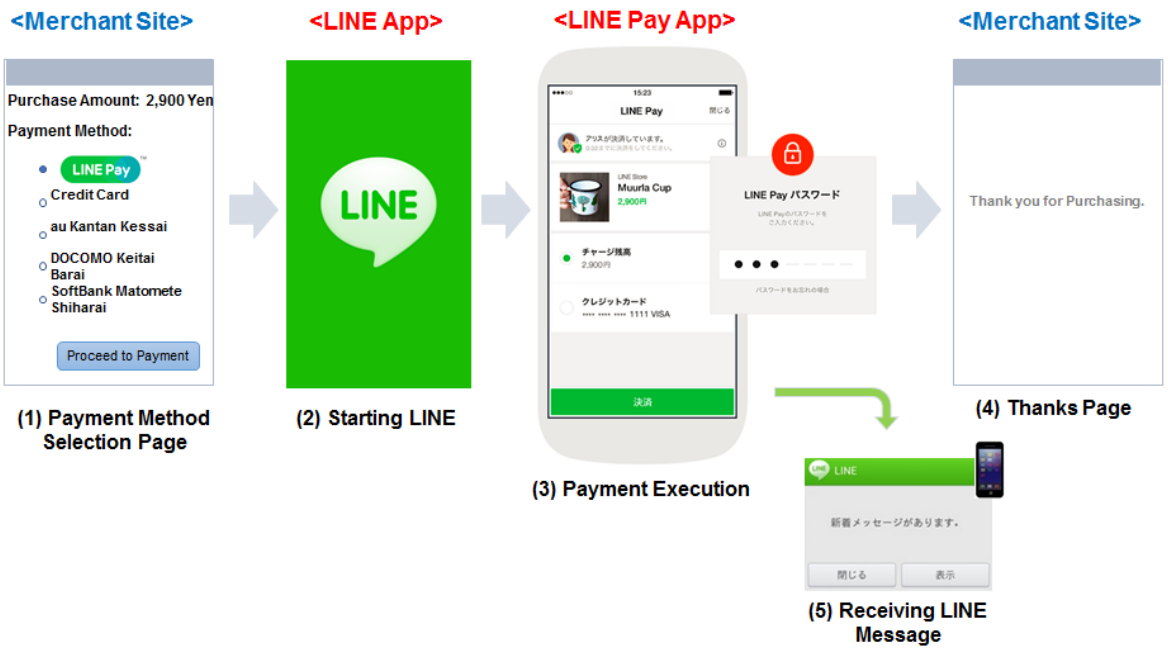
2-1 LINE Pay Service Overview

The LINE Pay service enables making payments of items purchased on merchant sites with a LINE user ID and LINE Pay payment password. It facilitates making payments with credit cards registered with LINE Pay or by using the balance from a LINE Pay account.

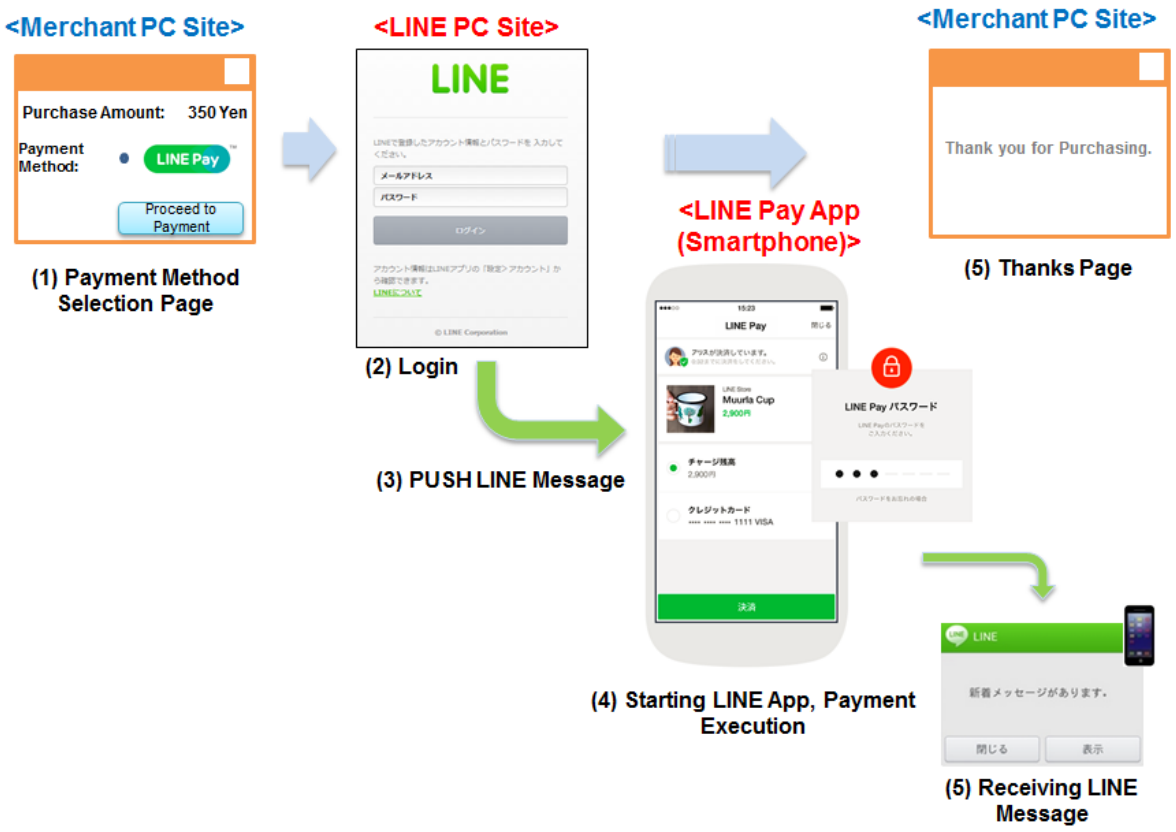
- This is a payment method where a registered LINE Pay user can easily make a payment by only using a password.
- It enables consumers to make payments with their credit cards that are registered with LINE Pay.
  - Payment is possible depending on the credit limit status of the consumer's credit card.
  - If the card company rejects the payment, this will result in an error. However, the merchant will be unable to understand the detailed reason for the rejection.
- This service enables making payments with the balance from a LINE Pay account.
- It also enables making payments with smartphones on which the LINE app is installed.
  - It also enables the functionality to make a payment request from a PC site and use a smartphone for payment.
  - It also enables consumers to make payments with a LINE app barcode/QR code for in-store purchase at the merchant's cash register. (This is supported from LINE ver5.1 and above.)
- The products are compliant to digital contents/sale of goods.

The following image shows the page transition of payment.

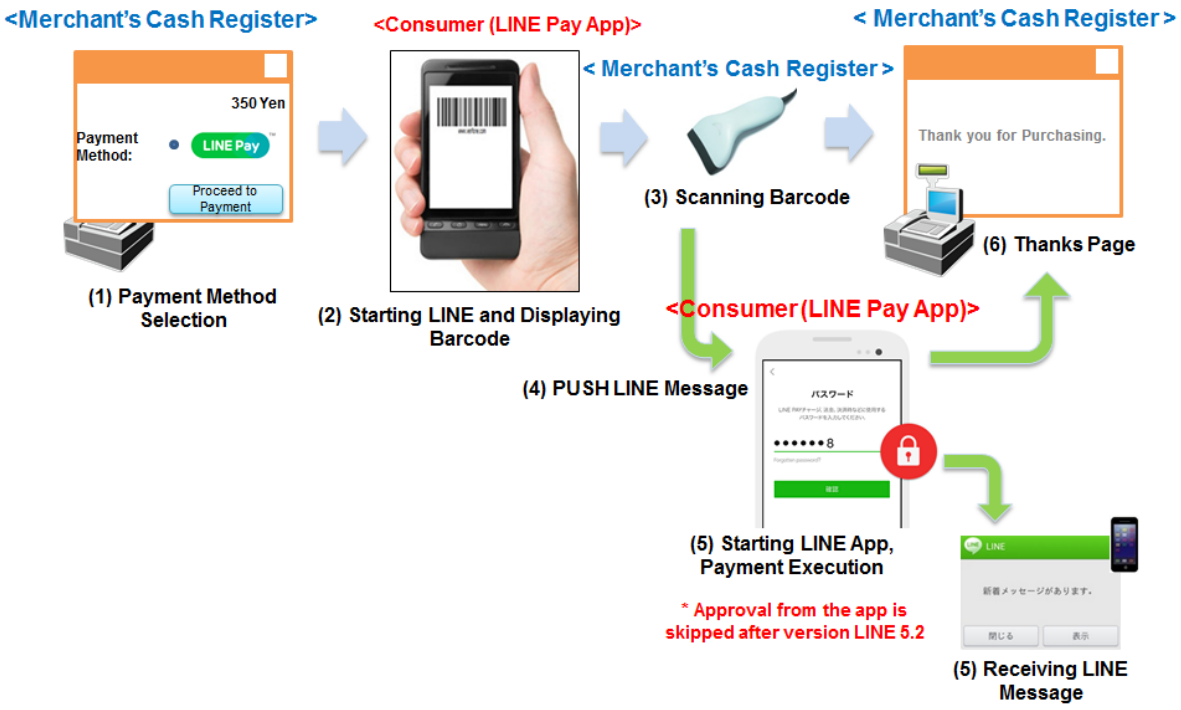
(Payment with smartphone)



(Payment with PC + smartphone)

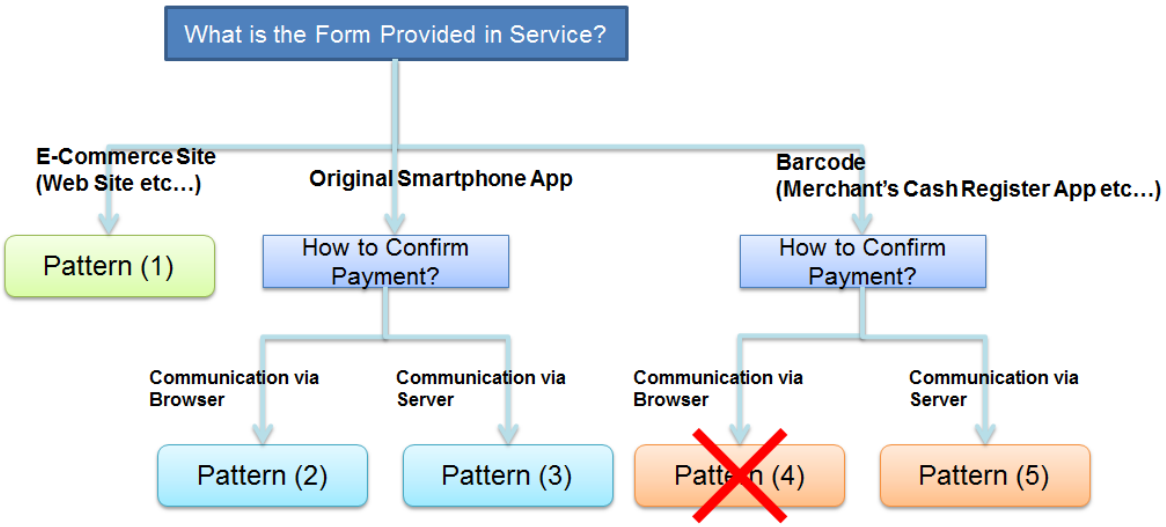


(In-store payment at merchant cash register with smartphone)



2-2 **LINE Pay Service Pattern**

The multiple service patterns offered in LINE Pay are shown as follows.



Supplementary information for patterns

Pattern (1): A pattern where the consumer mainly makes payment request using Web browser.
It will be the same for PC Web browser as well as for smartphone Web browser.

Pattern (2): A pattern where the payment confirmation is done on the app browser (WebView) of merchant's app.

Pattern (3): A pattern where the payment confirmation is done by the result notification (or search command) using merchant's app.

Pattern (4): A pattern where the consumer scans the barcode from his app and confirms the payment on the browser (WebView) of the merchant's cash register app. **(* This function is currently unavailable.)**

Pattern (5): A pattern where the consumer scans the barcode from his app and confirms the payment by result notification (or search command) on merchant's cash register.

The relationship of this document with each pattern mentioned above is shown in following table.

○: Mandatory　　Δ: Required based on usage method　　X: Not required

Chapter Number	Pattern (1)	Pattern (2)	Pattern (3)	Pattern (4)	Pattern (5)
3-1	○				
3-2					
Payment processing request	PC + smartphone, smartphone	Smartphone app to LINE Pay * Via browser	Smartphone app to LINE Pay * Inter-server communication	Merchant's cash register to LINE Pay * Via browser	Merchant's cash register to LINE Pay * Inter-server communication
Capture processing	○				
Cancellation processing	○				
4-1	Δ (*1)				

Chapter Number	Pattern (1)	Pattern (2)	Pattern (3)	Pattern (4)	Pattern (5)
5-1					
5.1.1	○	○	○ * "Payment Confirmation Method" is specified.	○ * “Onetime Key” is specified.	○ * “Onetime Key” and “Payment Confirmation Method” are specified.
5.1.2	○	○	X	○	X
5.1.3	○				
5.1.4	○				
5-2	Δ (*2)				
5-3	Δ (*1)				
6-1	○				
6-2	○				
6-3	Δ (*2)				
6.3.1	X	X	Δ	X	Δ
6-4					
6.4.1	Δ (*3)	○	X	○	X
6.4.2	X	○	X	○	X
6.4.3	Δ (*4)	Δ (*4)	X	Δ (*4)	X
7-1					
7.1.1	○	X	X	X	X
7.1.2	X	○	○	○	X
7-2	X	X	X	X	○
7-3	○				

*1: When using result notification. *2: When using search command.

*3: When specifying used browser.

*4: When a problem occurs if the browser at the time of payment request (including the browser in app) and the browser at the time of payment completion are different (sample: when using session.)

Chapter 3 Process Overview of MDK

3-1 List of Functions (Payment Request Command)

The available payment request commands are shown in following table.

Payment Request Type / Command	Digital Contents	Sales of Goods	Overview
Request (Authorization) /Authorize	○	○	It makes the payment request (authorization [*1]).
Request (Authorization + Capture) /Authorize	○	○	It makes the payment request (authorization) and capture at the same time.
Capture /Capture	○	○	It requests the capture of the requested payment. The capture amount cannot be specified beyond authorization amount (partial capture).
Cancel /Cancel	○	○	It cancels the payment. For cancellation before capture, authorization will get canceled. For cancellation after capture, cancellation amount cannot be specified beyond capture amount (partial cancellation) [*2].

*1 In LINE Pay "Authorization", authorization is performed on a credit card registered with LINE Pay excluding the payment with a balance in a LINE Pay account.

*2 Partial cancellation can be repeated until the balance in payment amount becomes 0.

3-2 **Payment Processing Sequence**

In LINE Pay services, the flow of the LINE Pay connection differs depending on the access environment (PC, smartphone) and the payment method (EC site, in-store) of the consumer as well as the payment confirmation method (via browser and inter-server communication).

The details of each flow are described below.

■ Payment flow when EC site is used

- The payment flow when an EC site is used differs slightly depending on the access environment (PC, smartphone).

  (There is almost no difference in the interface between merchant site and payment server.)

  - ✧ In the case of access from a PC, payment is made by LINE app (LINE Pay app) on a smartphone after the consumer authenticates LINE on a PC browser. After that, it will automatically transit to the payment completion page on the merchant's PC browser.
  - ✧ In the case of access from a smartphone, payment is made by automatically started LINE app after moving to the payment URL of LINE Pay from the merchant site. After payment execution, the LINE app will start the browser and it will transit to the payment completion page of the merchant.

■ Payment flow when merchant's app is used

- The merchant can connect to LINE Pay from his smartphone app as well. This chapter only describes the overview flow. For details, please refer to the "6-4 Regarding Smartphone Site and Connection between Original App and LINE Pay App".

■ In-store payment flow when barcode and QR code are used

- Payment request can be made by scanning a barcode (QR code) displayed in the LINE Pay app on the consumer's smartphone at the merchant's cash register and by setting that code value to MDK request parameter. (Barcode/QR code are supported from version LINE5.1 and above.)

◆ The merchant can select a method to confirm whether payment at his side was successful in each payment flow.

- ✓ Payment confirmation via browser ... this is mainly done when completing payment on an EC site or when the browser within the merchant's app (including merchant's cash register app) is used.
- ✓ Payment confirmation by inter-server communication ... this is mainly done when the browser on the merchant's app (including merchant's cash register app) is not used

◆ To complete the payment, it is necessary for the consumer to complete the operation on the LINE app. However, after payment Reserve request from the payment server to the LINE Pay system is received, if the consumer takes 20 minutes or more to complete the operation on the LINE app, the concerned order on the LINE Pay system side becomes invalid. Please request that the consumer makes the payment within 20 minutes.

[LINE Pay: Processing at the time of Payment Request (PC + Smartphone)]

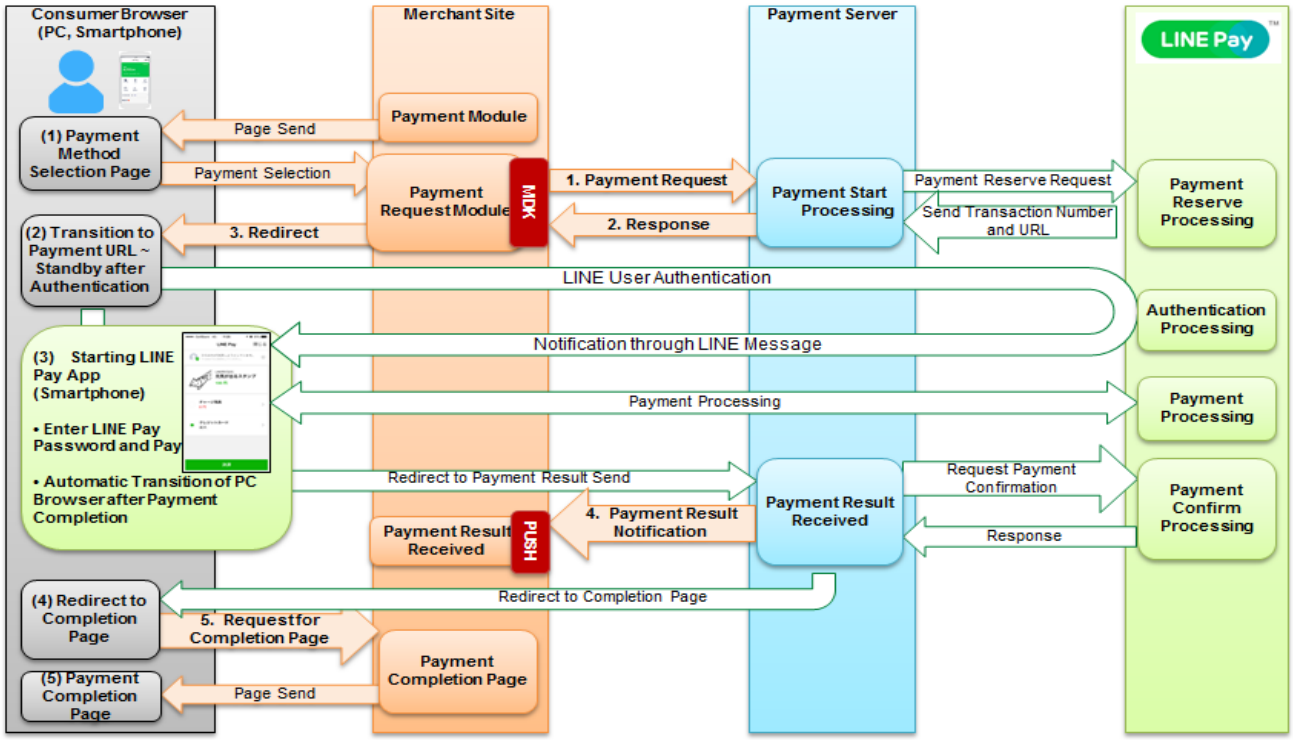


Figure 3-2-1 System Process Overview Diagram when Using MDK (LINE Pay Request on PC)

[LINE Pay: Processing at the time of Payment Request (Smartphone)]

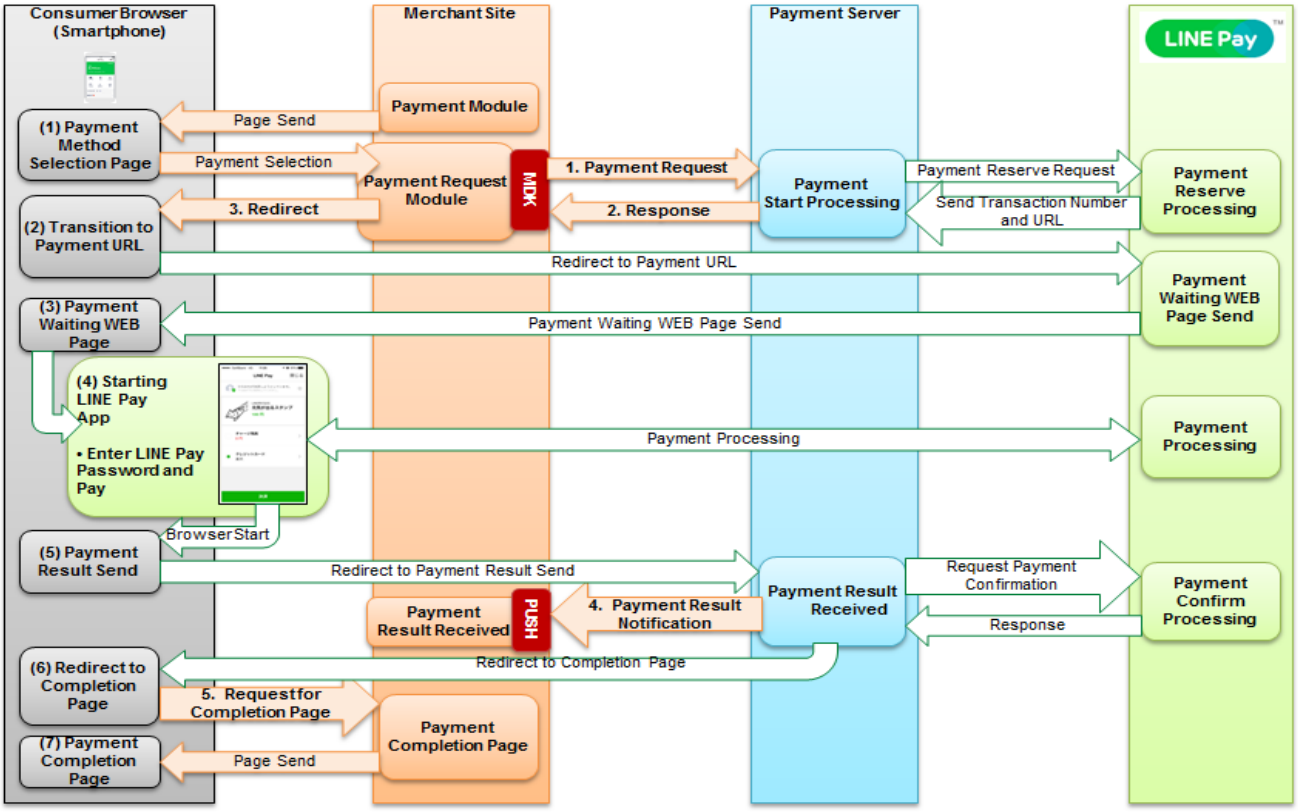


Figure 3-2-2 System Process Overview Diagram when Using MDK (LINE Pay Request on Smartphone)

	Basic Functions	Description
1	Payment Request	Sends payment request received from EC site to the payment server.
2	Response	Receives response returned from the payment server. If the response is successful (payment is possible), execute "3. Redirect". The response returned from the payment server includes a redirect page URL which is returned to the consumer browser.
3	Redirect	Sends redirect request of LINE Pay payment URL to the consumer browser by using a redirect page URL (payment URL) returned from the payment server. (*1)
4	Payment Result Notification (PUSH)	Notifies payment result only if authorization request is successful. The payment server notifies the payment result to the result notification URL that is registered from contract information check/change page on MAP (Merchant Administration Portal) as POST request. After receiving the payment results, reflect the same in order data.
5	Request for Completion Page	Receives response from the consumer browser and sends completion page. This sequence does not occur if the communication is disconnected or if the browser is accidentally closed. However, the payment result will not be lost if the 4. Payment Result Notification (PUSH) is received.

*1 Either set redirect page URL to HTTP response header (Location header) and redirect by HTTP status code = 302 or allow it to automatically transit to redirect page URL using JavaScript.

[LINE Pay: Processing at the time of Payment Request (from Smartphone App to LINE Pay) * via Browser]

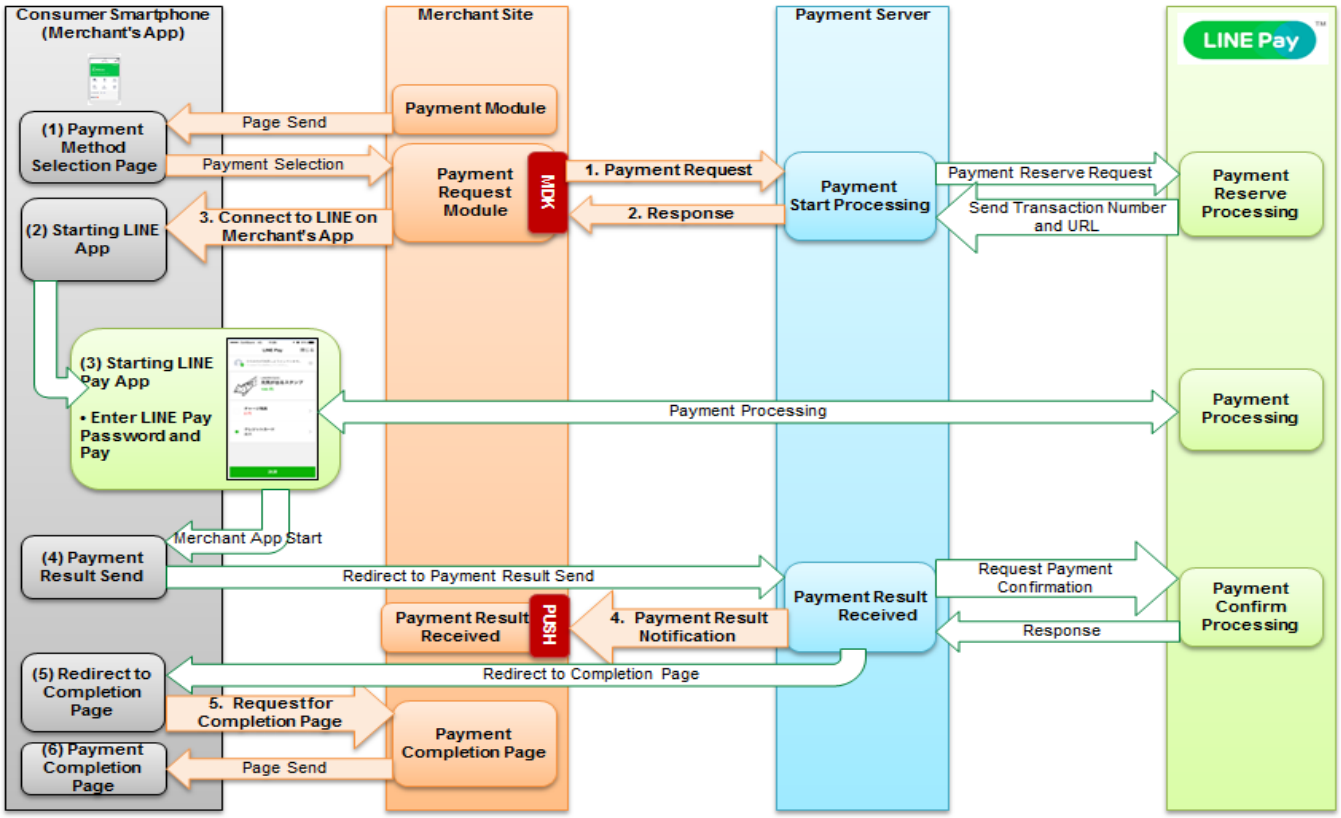


Figure 3-2-3 System Process Overview Diagram when Using MDK (Line Pay Request by Smartphone App * via Browser)

No.	Basic Functions	Description
1	Payment Request	Sends payment request received by EC site from merchant's smartphone app to the payment server.
2	Response	Receives response returned from the payment server. If the response is successful (payment is possible), execute "3. Connect to LINE on Merchant's App". The response returned from the payment server includes the URL scheme to start LINE app.
3	Connect to LINE on Merchant's App	Sends the URL scheme to start LINE app returned from the payment server to merchant's app started on smartphone. Starts the LINE app using this URL scheme in merchant's app. If payment is made with the LINE app (LINE Pay app), since control returned back to merchant's app again, sends response to the payment server URL which was given when app starts.
4	Payment Result Notification (PUSH)	Notifies payment result only if authorization request is successful. The payment server notifies the payment result as POST request to the result notification URL that is registered from contract information check/change page on MAP (Merchant Administration Portal). After receiving the payment results, reflect the same in order data.
5	Request for Completion Page	Receives response from the merchant's app and sends completion page. This sequence does not occur if the communication is disconnected or if the app is accidentally closed. However, the payment result will not be lost if the 4. Payment Result Notification (PUSH) is received.

[LINE Pay: Processing at the time of Payment Request (from Smartphone App to LINE Pay) * Inter-Server Communication]

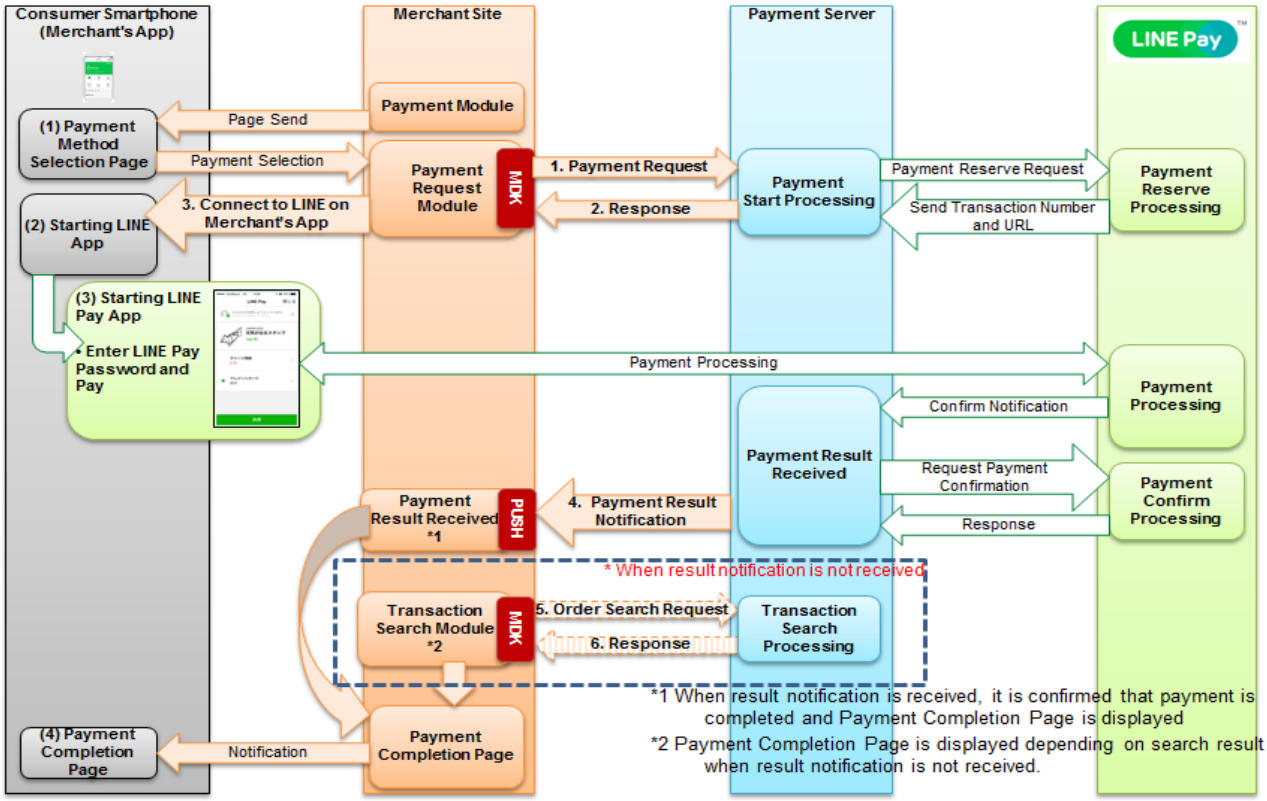


Figure 3-2-4 System Process Overview Diagram when Using MDK (LINE Pay Request on Smartphone App * Inter-Server Communication)

No.	Basic Functions	Description
1	Payment Request	Sends payment request received by EC site from the merchant's smartphone app to the payment server.
2	Response	Receives response returned from the payment server. If the response is successful (payment is possible), execute "3. Connect to LINE on Merchant's App". The response returned from the payment server includes the URL scheme to start LINE app.
3	Connect to LINE on Merchant's App	Sends the URL scheme to start the LINE app returned from the payment server to the merchant's app that has been started on a smartphone. Starts LINE app using this URL scheme in merchant's app.
4	Payment Result Notification (PUSH)	Notifies payment result. (Notifies both success and failure.) The payment server notifies the payment result as POST request to the result notification URL that is registered from the contract information check/change page on MAP (Merchant Administration Portal). After receiving the payment results, reflect the same in order data and carry out completion process.

No.	Basic Functions	Description
5,6	Order Search Request and Response	Searches requested order and checks the payment status when payment result notification is not received. Checks again when payment is not yet completed (success or failure). If the payment is completed (success or failure), reflect the same in order data and carry out completion process. * For the confirmation method of payment completion, please refer to the “6.3.1 Regarding Search Result when Payment Confirmation Method is Inter-Server Communication”.

[LINE Pay: Processing at the time of Payment Request (from Merchant’s Cash Register to LINE Pay) * via Browser]

(* This function is currently unavailable.)

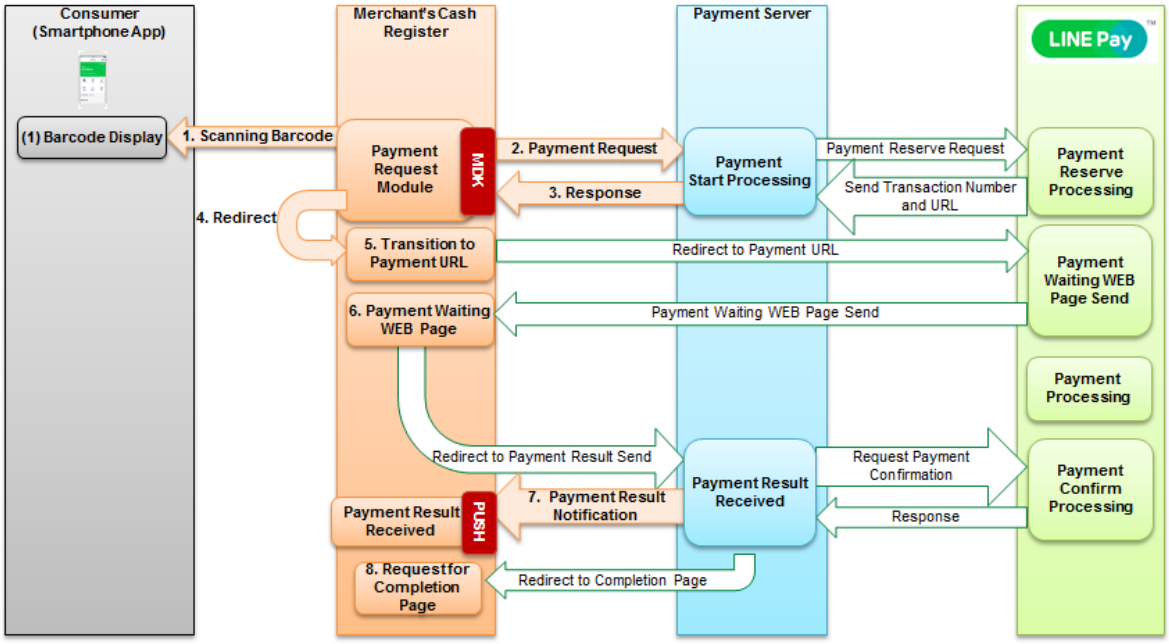


Figure 3-2-5 System Process Overview Diagram when Using MDK (Line Pay Request at Merchant’s Cash Register * via Browser)

No.	Basic Functions	Description
1	Scanning Barcode	Scans barcode displayed on consumer's LINE app using merchant’s cash register app.
2	Payment Request	Sends barcode received from the merchant and payment request to the payment server.
3	Response	Receives response returned from the payment server. If the response is successful (payment is possible), execute "4. Redirect". The response returned from the payment server includes page URL which is redirected by an internal browser of merchant’s cash register app.
4	Redirect	Sends redirect request of LINE Pay payment URL to the internal browser of merchant’s cash register app by using redirect page URL (payment URL) returned from the payment server. (* 1)
5	Transition to Payment URL	Transits to LINE Pay page and shows standby page until payment is completed.

No.	Basic Functions	Description
6	Payment Waiting WEB Page	Redirects to the payment server from standby page once payment is completed.
7	Payment Result Notification (PUSH)	Notifies payment result only if authorization request is successful. The payment server notifies the payment result as POST request to the result notification URL that is registered from the contract information check/change page on MAP (Merchant Administration Portal). After receiving the payment results, reflect the same in order data.
8	Request for Completion Page	Receives response from merchant's app and sends completion page. This sequence does not occur if the communication is disconnected or if the app is accidentally closed. However, the payment result will not be lost if the 7. Payment Result Notification (PUSH) is received.

*1 Either set redirect page URL to HTTP response header (Location header) and redirect by HTTP status code = 302 or allow it to automatically transit to redirect page URL using JavaScript.

[LINE Pay: Processing at the time of Payment Request (from Merchant's Cash Register to LINE Pay) * Inter-Server Communication]

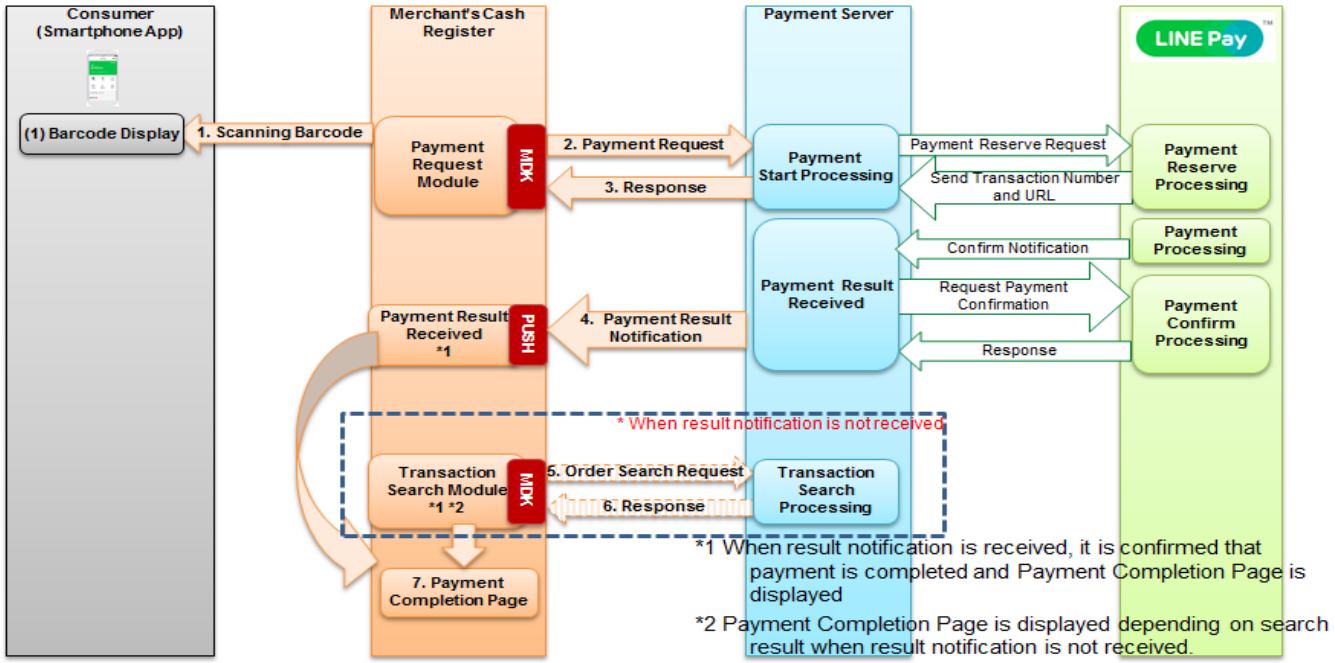


Figure 3-2-6 System Process Overview Diagram when Using MDK (Line Pay Request at Merchant's Cash Register* Inter-Server communication)

No.	Basic Functions	Description
1	Scanning Barcode	Scans barcode displayed on consumer's LINE app using merchant's cash register app.
2	Payment Request	Sends barcode received from merchant and payment request to the payment server.
3	Response	Receives response returned from the payment server. If the response is successful (payment is possible), the authentication & payment happen between LINE Pay and consumer. Displays payment standby page on merchant's cash register app.

No.	Basic Functions	Description
4	Payment Result Notification (PUSH)	Notifies payment result. (Notifies success and failure both.) The payment server notifies the payment result as POST request to the result notification URL that is registered from the contract information check/change page on MAP (Merchant Administration Portal). After receiving the payment results, reflect the same in order data and carry out completion process.
5,6	Order Search Request and Response	Searches requested order and checks the payment status when payment result notification is not received. Checks again when the payment is not yet completed (success or failure). If the payment is completed (success or failure), reflect the same in order data and carry out completion process. * For the confirmation method of payment completion, please refer to the “6.3.1 Regarding Search Result when Payment Confirmation Method is Inter-Server Communication”.
7	Payment Completion Page	Displays completion page when payment completion is confirmed either by search result or result notification.

[LINE Pay: Capture Processing]

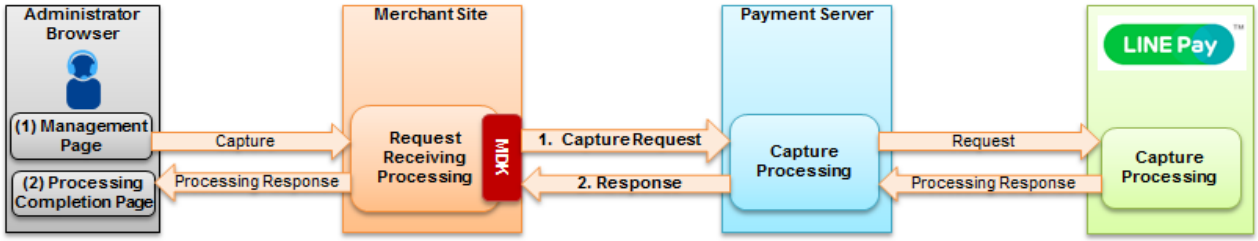


Figure 3-2-7 System Process Overview Diagram when Using MDK (LINE Pay Capture Request)

No.	Basic Functions	Description
1	Capture Request	Sends information of the order to be captured to the payment server.
2	Response	Receives response from the payment server.

[LINE Pay: Cancel Processing]

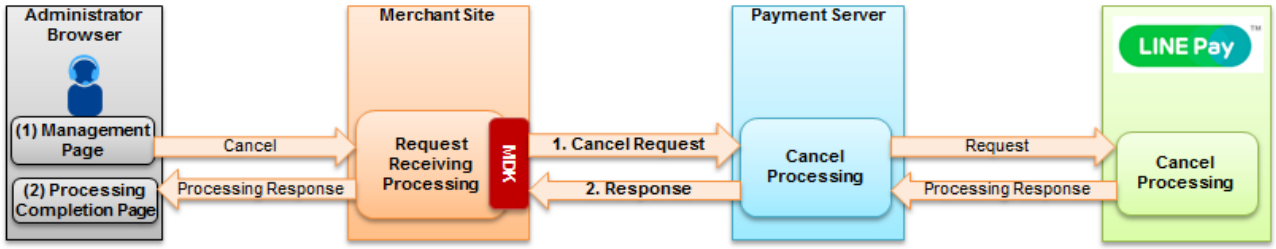


Figure 3-2-8 System Process Overview Diagram when Using MDK (LINE Pay Cancel Request)

No.	Basic Functions	Description
1	Cancel Request	Sends information of the order to be canceled to the payment server.
2	Response	Receives response from the payment server.

Chapter 4 Handling Result Notification

4-1 Handling Result Notification

The functions for handling result notifications of LINE Pay are shown below.

Payment Service Name	Payment Request (Authorization / Authorization and Capture)	Capture	Cancel
LINE Pay	○	-	-

Table 1 List of Notification Targets

For the specifications about how a merchant should handle and respond to notifications, please refer to the "Development Guide".

For details about LINE Pay specific notification message interface, please refer to the “5.3.2 Result Notification Message of LINE Pay.

Chapter 5 Interface Details

The MDK interface items used in the LINE Pay Service are explained here.

■ The contents of the "Settings" column are as follows.

Request Message … Mandatory field: ○ Optional field: Δ Settings disabled: X Other conditions: *, *n

(Conditions are given in the description column or outside the column)

Response Message … Always returned: ○ Returned only when processing is successful: Δ Not returned: X Other conditions: *

■ About orderId (order ID)

Merchant should assign a (unique) number randomly. The number needs to be assigned for each order. Assign a (unique) ID that is different from other order IDs. The order ID must be unique across other payment services as well. The order ID used in the test order cannot be re-used in the live order.

In the order ID, “-” (hyphen), and “_” (underscore) can also be used besides the single byte alphanumeric characters.

■ About response message

The response message also contains the fields (parameters) that are not given in this document. Normally, merchants need not be aware of such fields.

■ About “Settings” column

Title (B/S) of the Settings column indicates the payment confirmation method. Please note that the items that can be set differ depending on the payment confirmation method.

B: Communication via browser S: Inter-server communication

5-1 LINE Pay

5.1.1 Request

Request Message: LinepayAuthorizeRequestDto					
Field Name	Item Name	Format and Limitations	Description	Settings	
				B	S
orderId	Order ID	Single byte alphanumeric characters; 100 characters or less	Specifies (unique) order ID assigned at merchant side.	○	○
amount	Payment amount	Single byte numbers; 7 digits or less	Specifies payment amount (total amount).	○	○
withCapture	Capture flag	Please refer to the description in the right column.	"true": Authorize with Capture. "false": Authorization only (default value). * If not specified, default is Authorization.	Δ	Δ

Request Message: LinepayAuthorizeRequestDto					
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**	
				B	**S**
itemId	Item ID	Single byte alphanumeric characters; 64 characters or less	Specifies the number for managing the item or service that is assigned at merchant side. • It is not send to the LINE Pay system.	Δ	Δ
itemName	Item name	Double byte and single byte; 4000byte or less	Specifies the item name. * It is displayed on the consumer page offered by LINE or in the e-mail body.	○	○
itemImageUrl	Item image URL	Single byte characters that can be used in URL; 256 characters or less	Specifies the URL of the item image displayed on the payment page. Size: 84 x 84	Δ	Δ
checkUseBrowser	Check used browser	Please refer to the description in the right column.	"true": Provides the guidance page to return to the browser that requested payment at LINE Pay when the browser that requested payment and the browser transited from LINE app differ. "false": Does not confirm the browser that requested payment at LINE Pay and the browser that transits from the LINE app. * If not specified, default is "false". * For details, please refer to the “6.4.3 How to Use checkUseBrowser.	Δ	Δ
appUrlScheme	App start URL scheme	Single byte characters that can be used in URL; 256 characters or less	Browser started by the LINE Pay app or URL scheme to start original app. Specify this while using LINE Pay from a smartphone site or from the original app running on a smartphone. * For more details, please refer to the “6.4.1 How to Specify appUrlScheme”.	*	X
useOriginalApp	Optional specification at the time of starting original app	Please refer to the description in the right column.	"0": No optional specification (default value). “1": Encode the <url> with URL encoding. When this option is specified, set the URL string where the URL on the payment server is encoded in a placeholder <url> specified in appUrlScheme.	Δ	X
mid	LINE member ID	Single byte alphanumeric characters; 50 characters or less	Unique ID to identify the LINE user. * This is currently unavailable.	X	X
packageName	packageName	Single byte alphanumeric characters; 4000 characters or less	packageName to be specified at the time of starting the app on Android. * For details, please refer to the “6.4.1 How to Specify appUrlScheme”.	Δ	X

Request Message: LinepayAuthorizeRequestDto					
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**	
				B	**S**
successUrl	URL at the time of payment completion	Single byte characters that can be used in URL; 256 characters or less	Specifies the URL to return to the page on the merchant site when payment is successful (query parameter can be specified). * If not specified, use the value registered from MAP (Merchant Administrator Portal).	Δ	X
cancelUrl	URL at the time of cancellation	Single byte characters that can be used in URL; 256 characters or less	Specifies the URL to return to the page on the merchant site when payment is canceled (query parameter can be specified). * If not specified, use the value registered from MAP (Merchant Administrator Portal).	Δ	X
errorUrl	URL at the time of payment error	Single byte characters that can be used in URL; 256 characters or less	Specifies the URL to return to the page on the merchant site when error occurs during payment (query parameter can be specified). * If not specified, use the value registered from MAP (Merchant Administrator Portal).	Δ	X
pushUrl	Result notification URL	Single byte characters that can be used in URL; 256 characters or less	Specifies the URL that receives payment completion notification at the time of a “dummy order” (query parameter can be specified). * If not specified, use the value registered from MAP (Merchant Administrator Portal). * It is assumed that this parameter is used only at the time of developing the merchant system and it can be specified in a dummy order.	*	*
oneTimeKey	Onetime Key	Single byte alphanumeric characters; 12 characters or less	It is specified when payment is to be executed by scanning the QR code or barcode at the merchant’s cash register etc. Obtain the QR/BarCode offered by the LINE Pay app and specify that value. The validity period for this value is within 5 minutes and the request must be done within the validity period. Returns IG08 when same value is set after payment is completed or when incorrect value is set. QR/Barcode are supported from version LINE 5.1 and above. * When specifying this item, specify paymentConfirmType as “1”: Inter-server communication.	Δ	Δ

Request Message: LinepayAuthorizeRequestDto					
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**	
				B	**S**
paymentConfirmType	Payment confirmation method	Please refer to the description in the right column.	Specifies the method of payment confirmation. "0": Communication via browser (default value). "1": Inter-server communication.	Δ	○

Response Message: LinepayAuthorizeResponseDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
serviceType	Payment service type	Single byte alphanumeric characters; 10 characters or less	Payment service type sent by request message.	○
mstatus	Process result code	Single byte alphanumeric characters; 32 characters or less	“success”: Normal termination. "failure": Abnormal termination.	○
vResultCode	Detailed result code	Single byte alphanumeric characters; 16 characters	Code that represents the process result in detail. It consists of 4 blocks of 4 characters each and each block represents the process result of each service. For details, please refer to the "Result code list" separately.	○
merrMsg	Error message	String; 1024 bytes or less	Process result in Japanese or English.	○
marchTxn	Message ID	String; 100 characters or less	ID assigned by payment server per payment process message (including internal process). Multiple IDs are assigned to a single Order ID.	○
orderId	Order ID	Single byte alphanumeric characters; 100 characters or less	(Unique) order ID randomly assigned and sent by the merchant at the time of payment request.	○
linepayOrderId	LINEPay Order number	Single byte alphanumeric characters; 19 characters or less	Order number assigned at the LINEPay system. * It is not set in case of mstatus=failure.	Δ
custTxn	ID assigned per transaction	String; 100 characters or less	ID (uniquely) assigned by payment server to link the order (Order ID).	○
txnVersion	MDK version	Single byte alphanumeric characters; 5 characters	Message version. It is not generally used except in the case of a problem.	○

Response Message: LinepayAuthorizeResponseDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
redirectWebUrl	WebURL for Redirect	String	Payment URL offered by LINE Pay. When payment request is made from a PC, or from website for a smartphone (without starting merchant's app), it will directly transit to this URL. POST and GET both are available. * It is not set when oneTimeKey and paymentConfirmType="1" are set in request message or when process fails.	*
redirectAppUrl	URL to move between app in LINE (URL Scheme)	String	URL Scheme to move to payment pages in LINE Pay app. Use these URLs to move to LINE Pay payment pages from the merchant's app. Check if the LINE is installed on the device or not. Also check the version of the device to ensure if LINE Pay can be used on it. Move to LINE app by this URL. For details, please refer to the "6-4 Regarding Smartphone Site and Connection between Original App and LINE Pay App". * It is not set when oneTimeKey and paymentConfirmType="1" are set in request message or when process fails.	*

5.1.2 Query Parameters when Redirecting to Merchant's Completion Page from Consumer Browser

* It is not redirected when paymentConfirmType="1" (inter-server communication) in payment request.

Redirect contents: Contents that are redirected (GET) to the merchant from payment server via consumer browser				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
mstatus	Process result code	Single byte alphanumeric characters; 32 characters or less	"success": Normal termination. "failure": Abnormal termination. "pending": Pending.	○
vResultCode	Detailed result code	4 Single byte alphanumeric characters	Code that represents the process result in detail. For details, please refer to the "Result code list" separately.	○
orderId	Order ID	Single byte alphanumeric characters; 100 characters or less	(Unique) order ID randomly assigned and sent by the merchant at the time of payment request.	○
txnType	Transaction type	String Please refer to the description in the right column.	Type of transaction to be notified is set. "Authorize"	○
linepayOrderId	LINEPay order number	Single byte alphanumeric characters; 19 characters or less	Order number assigned at the LINE Pay system. * It is not set in the case of mstatus=failure, pending.	Δ

vAuthInfo	Hash value for tampering check	String Please refer to the description in the right column.	The following strings are concatenated and hash value is calculated by SHA-256. • Merchant CCID. • Concatenated string of parameter value (concatenated as per the sequence indicated by authParams). • Password. When concatenating parameter values, only the parameter values are concatenated without including the parameter name and delimiter. UTF-8 is used as the character encoding while encoding the concatenated character string into the binary.	○
authParams	Hash value calculation parameter sequence	String Please refer to the description in the right column.	The value indicating the concatenating sequence of the parameter is used to calculate the hash value of vAuthInfo. The comma delimiter string used between the parameter names is Base64 encoded. On decoding, the Comma delimiter string can recover. Sample) "orderId,vResultCode,mstatus" "mstatus,orderId,vResultCode" (As the sequence is not fixed, it is necessary to perform the process dynamically on receiving the request.)	○

✧ About tampering check using vAuthInfo and authParams

vAuthInfo and authParams are the parameters to verify that the query parameters received by the merchant system are not tampered with when the consumer browser is redirected from the payment server to the online shop.

When the hash value calculated by the merchant system matches with the “vAuthInfo” received through the query parameter, it is presumed that the parameters are not tampered.

Although this tampering check is not mandatory, invalid redirect messages may be received from the third party with the bad intent. For this reason, we strongly recommend that you perform a tampering check.

For details of the implementation method, please refer to the sample program offered by VeriTrans.

5.1.3 Capture

Request Message: LinepayCaptureRequestDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
orderId	Order ID	Single byte alphanumeric characters; 100 characters or less	Specifies order ID to be captured.	○
amount	Capture amount	Single byte numbers; 7 digits or less	Specifies capture amount. * If not specified, full amount is captured.	Δ

Response Message: LinepayCaptureResponseDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
serviceType	Payment service type	Single byte alphanumeric characters; 10 characters or less	Payment service type sent by request message.	○
mstatus	Process result code	Single byte alphanumeric characters; 32 characters or less	“success”: Normal termination. "failure": Abnormal termination. “pending": Pending.	○
vResultCode	Detailed result code	Single byte alphanumeric characters; 16 characters	Code that represents the process result in detail. It consists of 4 blocks of 4 characters each and each block represents the process result of each service. For details, please refer to the "Result code list" separately.	○
merrMsg	Error message	String; 1024 bytes or less	Process result in Japanese or English.	○
marchTxn	Message ID	String; 100 characters or less	ID assigned by payment server per payment process message (including internal process). Multiple IDs are assigned to a single Order ID.	○
orderId	Order ID	Single byte alphanumeric characters; 100 characters or less	(Unique) order ID randomly assigned and sent by the merchant at the time of payment request.	○
custTxn	ID assigned per transaction	String; 100 characters or less	ID (uniquely) assigned by payment server to link the order (Order ID).	○
txnVersion	MDK version	Single byte alphanumeric characters; 5 characters	Message version. It is not generally used except in the case of a problem.	○
captureDatetime	Capture date and time	String; 14 characters	YYYYMMDDhhmmss format.	Δ
balance	Balance	Single byte numbers; 7 digits or less	Returns current payment amount.	Δ

5.1.4 Cancel

Request Message: LinepayCancelRequestDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
orderId	Order ID	Single byte alphanumeric characters; 100 characters or less	Please refer to the "About orderID (order ID)" described above.	○

Request Message: LinepayCancelRequestDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
amount	Reduced amount	Single byte numbers; 7 digits or less	Specifies the amount to be reduced. * If not specified, full amount is canceled. * In case of the order for which capture is not yet completed, only full amount can be specified.	Δ

Response Message: LinepayCancelResponseDto				
Field Name	**Item Name**	**Format and Limitations**	**Description**	**Settings**
serviceType	Payment service type	Single byte alphanumeric characters; 10 characters or less	Payment service type sent by request message.	○
mstatus	Process result code	Single byte alphanumeric characters; 32 characters or less	“success”: Normal termination. "failure": Abnormal termination. “pending": Pending.	○
vResultCode	Detailed result code	Single byte alphanumeric characters; 16 characters	Code that represents the process result in detail. It consists of 4 blocks of 4 characters each and each block represents the process result of each service. For details, please refer to the "Result code list" separately.	○
merrMsg	Error message	String; 1024 bytes or less	Process result in Japanese or English.	○
marchTxn	Message ID	String; 100 characters or less	ID assigned by payment server per payment process message (including internal process). Multiple IDs are assigned to a single Order ID.	○
orderId	Order ID	Single byte alphanumeric characters; 100 characters or less	(Unique) order ID randomly assigned and sent by the merchant at the time of payment request.	○
custTxn	ID assigned per transaction	String; 100 characters or less	ID (uniquely) assigned by payment server to link the order (Order ID).	○
txnVersion	MDK version	Single byte alphanumeric characters; 5 characters	Message version. It is not generally used except in the case of a problem.	○
cancelDatetime	Cancel date and time	String; 14 characters	YYYYMMDDhhmmss format.	Δ
balance	Balance	Single byte numbers; 7 digits or less	Returns current payment amount.	Δ

5-2 **Common**

5.2.1 Search

■ The contents of the "Settings" column are as follows.

Request Message … Mandatory field: ○ Optional field: Δ Settings disabled: X Other conditions: *

(Conditions are given in the description column)

Response Message … Always returned: ○ Returned when corresponding order exists: Δ Not returned: X

Other conditions: *

■ Multiple specifications are specified with subscript from 0~.

Sample) exparam.serviceTypeCd[0]=card&exparam.serviceTypeCd[1]=linepay

■ Wild card is searched by combining part of value and "*". Only "*" cannot be specified.

Sample) exparam.searchParameters.common.orderId=123*

■ Other than a normal search, master information can be obtained. The following are the interfaces at the time of obtaining master information.

Request Message: SearchRequestDto						
*** The following are the common search request fields.**						
Search Field Name	**Search Item Name**	**Format and Limitations**	**Multiple Specification**	**Wild Card**	**Description**	**Settings**
requestId	Request ID	Single byte alphanumeric characters and symbols; 128 characters or less			Specifies the request ID. When it is specified, other parameters cannot be specified.	Δ
serviceTypeCd	Payment service type	Please refer to the description in the right column.	○		Specifies payment to be searched. If not specified, all payments are searched. "linepay": LINE Pay	Δ

newerFlag	Latest transaction flag	Please refer to the description in the right column.			Used when only latest transaction is to be searched regardless of success, failure. “true": Searches only the latest transaction in 1 order. "false": Searches all transactions. * If not specified, default is “false”.	∆
containDummyFlag	Dummy payment target flag	Please refer to the description in the right column.			It is used when dummy order is also to be searched. “true": Searches dummy order as well. "false": Does not search dummy order. * If not specified, default is “false”.	∆
maxCount	Maximum search cases	1~1000			It is the maximum cases of search results to be acquired. If not specified, default is maximum value. * Please avoid repitative search with in short spans which returns large result set as is may loads up the server.	∆
common	**Common**					
orderId	Order ID	Single byte alphanumeric characters; 100 characters or less		○	Specifies order ID to be searched.	∆
orderStatus	Order payment status	Please refer to the description in the right column.	○		Specifies any of the following. “ initial ": Initial status “ end ": Termination " end_presentation ": Page redirect normal termination " pending ": Pending " validation_error ": Verification error “ expired ": Expired " error ": Error	∆
command	Command	Please refer to the description in the right column.	○		Specifies the command that is used in payment. “ Authorize ": Authorize, Request " Capture ": Capture " Cancel ": Cancel	∆
mstatus	Status code	Please refer to the description in the right column.	○		Specifies the status code that is returned as payment result. “ success ": Successful " failure ": Failed “ pending ": Pending	∆
txnDatetime.from	Transaction date (From)	String; 12 characters			Specifies the range of transaction date and time (From). YYYYMMDDhhmm format.	∆
txnDatetime.to	Transaction date (To)	String; 12 characters			Specifies the range of transaction date and time (To). YYYYMMDDhhmm format.	∆

	amount.from	Amount (From)	Numbers; 12 digits or less			Specifies the range of payment amount (From).	Δ
	amount.to	Amount (To)	Numbers; 12 digits or less			Specifies the range of payment amount (To).	Δ
*** The following are the LINE Pay-specific search request fields.**							
linepay	**LINE Pay**						
	detailOrderType	Detailed order payment status	Please refer to the description in the right column.	○		* For details of “detailOrderType”, please refer to the “6-3 Supplementary Information for Search (Detailed Command Type/Detailed Order Payment Status)”.	Δ
	detailCommandType	Detailed command type	Please refer to the description in the right column.	○		* For details of “detailCommandType”, please refer to the “6-3 Supplementary Information for Search (Detailed Command Type/Detailed Order Payment Status)”.	Δ
	itemId	Item number	Single byte alphanumeric characters; 64 characters or less		○	Item number specified at the time of payment request.	Δ

Response Message: SearchResponseDto

*** The following are the common search result fields.**

Search Field Name	Search Item Name	Format and Limitations	Description	Settings
result	Process result	-		○
mstatus	Process result code	Single byte alphanumeric characters; 32 characters or less	Process result status is saved. “success”: Normal termination. "failure": Abnormal termination.	○
vResultCode	Detailed result code	Single byte alphanumeric characters; 16 characters	Code that represents the process result in detail. For details, please refer to the "Result code list" separately.	○
merrMsg	Error message	String; 1024 bytes or less	Process result in Japanese or English.	○
exresult	Extended result	-		Δ
overMaxCountFlag	Flag of maximum cases exceeded	Please refer to the description in the right column.	Displays whether data to be searched is more than the maximum cases to be searched specified in request message. "true": More than maximum cases. "false": Less than maximum cases.	Δ
searchCount	Search result cases	0~1000	Search result cases (order cases) are saved.	Δ
orderInfos	Order information list		Multiple order information (orderInfo) is saved.	Δ
orderInfo	Order information	-	Information of order applicable to search conditions is repeated for the times equivalent to the applicable cases. 0~1000 cases (till the maximum search cases specified in request message).	Δ
index	Index	0~999	Index of searched information is saved.	Δ
serviceTypeCd	Payment service type	Please refer to the description in the right column.	Payment type is saved. "linepay": LINE Pay	Δ
orderId	Order ID	String	Order ID of order is saved.	Δ

				orderStatus		Order payment status	Please refer to the description in the right column.	Payment status is saved. “ initial ": Initial status “ end ": Termination " end_presentation ": Page redirect normal termination “pending": Pending. " validation_error ": Verification error " expired ": Expired " error ": Error * This field does not express the status of order data related to order ID completely. Since a detailed status transition is maintained in a separate field depending on the service type, please refer to the successDetailTxnType and payment specific status field according to the usage of the system at the merchant side.	∆
				lastSuccessTxnType		Latest successful transaction type	String	Latest successful command name is saved.	∆
				successDetailTxnType		Detailed transaction type	String	Detailed status of transaction is saved. Please refer to the detailOrderType of each payment of search request message.	∆
				properOrderInfo		Specific order information	-	Specific order information of each payment service is saved. Please refer to the list described later.	∆
				transactionInfos		Payment transaction list	-	Multiple payment transaction information (transactionInfo) is saved.	∆
					transactionInfo	Payment transaction information	-	Information of order applicable to search conditions is repeated for the times equivalent to the applicable cases.	∆
					txnId	Transaction management ID	String	Management ID (uniquely) assigned by payment server.	∆
					command	Command	String	Executed command name is saved. * Not only the command requested by MDK, but the internal process command of the payment server is also included.	∆
					mstatus	Process result code	Single byte alphanumeric characters; 32 characters or less	Process result status is saved. “success”: Normal termination. "failure": Abnormal termination. "pending": Pending.	∆
					vResultCode	Detailed result code	String; 16 characters	Code that represents the process result in detail. For details, please refer to the "Result code list" separately.	∆
					txnDatetime	Order date and time	String; 23 characters	Order date and time is saved. YYYY-MM-DD hh:mi:ss.mmm format.	∆

						amount	Amount	Single byte numbers; 12 digits or less	Paid amount is saved.	Δ
						properTransactionInfo	Specific transaction information	-	Specific transaction information of each payment service is saved. Please refer to the list described later.	Δ

☆ The following information is set in hierarchy of said 'Specific order information' and 'Specific transaction information'.

*** The following are the LINE Pay-specific search request fields.**

properOrderInfo		LINE Pay Specific order information			
	Authorize with Capture flag	withCapture	String	Authorize with Capture flag specified at the time of payment request.	Δ
	Item name	itemName	String	Item name specified at the time of payment request.	Δ
	Item number	itemId	String	Item number specified at the time of payment request.	Δ
	Request amount	authorizeAmount	Single byte numbers; 7 digits or less	Amount specified at the time of payment request.	Δ
	Balance	balance	Single byte numbers; 7 digits or less	Balance (current payment amount).	Δ
	Payment request date and time	authorizeDatetime	String; 14 characters	Payment request date and time. YYYYMMDDhhmmss format.	Δ
	LINE Pay order number	linepayOrderId	String; 19 characters or less	Order number assigned at the LINE Pay system.	Δ
properTransactionInfo		LINE Pay Specific transaction information			
	LINE Pay error code	linepayErrorCode	String	Error code returned from LINE Pay system is saved.	Δ
	Detailed command type	detailCommandType	String	* For details of "detailCommandType", please refer to the "6-3 Supplementary Information for Search (Detailed Command Type/Detailed Order Payment Status)".	Δ
	Date and time of request to LINE Pay	linepayRequestDatetime	String; 14 characters	Date and time of request to LINE Pay is saved. YYYYMMDDhhmmss format.	Δ
	Date and time of returning from LINE Pay	linepayResponseDatetime	String; 14 characters	Date and time of returning from LINE Pay is saved. YYYYMMDDhhmmss format.	Δ

5-3 Result Notification Message

5.3.1 Overview of Result Notification Message

For details of the implementation method, please refer to the sample program offered by VeriTrans.

■ **Functional overview**

Payment server sends result notification to merchant system.

It is notified in HTTP POST.

*** Supported Protocol is HTTP (Port:80), HTTPS (Port:443).**

For process overview, please refer to the "Chapter 4 Handling Result Notification**".**

■ **About HMAC**

At the time of data notification in POST, HMAC is set in the request header.

Also, the value of HMAC is calculated from the request body.

The algorithm is set to "HmacSHA256".

Field Name	Setting Value
content-hmac	h={algorithm name};s={CCID};v={HMAC value}

Table 2 HMAC Fields Set in Request Header

■ **About response to result notification**

When the merchant system returns an HTTP status code of "200" for notification from the payment server, it treats that notification as successfully received. For a code other than "200", it treats that notification as not successfully received by the merchant and it continues to notify the merchant at regular intervals.

*** Please note that the notification stops if the number of times of failure reaches the stipulated value.**

5.3.2 Result Notification Message of LINE Pay

The notification timing of result notification message in LINE Pay differs depending on the payment confirmation method.

[When payment confirmation method is Communication via browser]

The notification is sent when payment request is completed at the LINE Pay side (when request from payment server is successful). At this time, the transition to the completion page URL of merchant does not happen.

If the consumer withdraws the request during the page transition or if time out occurs during the page transition due to a problem in the communication environment, you may not land on the last completion page URL. In such cases, the merchant system receives notification from the payment server and payment completion can be searched.

[When payment confirmation method is Inter-server communication]

The notification is sent when result of payment request at LINE Pay is confirmed. This is different from "Communication via Browser" and the notification is sent when request is successful as well as when request fails.

No.	Notification Function	Notification Timing	Important Point
1	Payment Request Completion Notification (Authorize) * Communication via browser	It is sent when payment request is completed (successful).	It is not sent when payment request fails.

No.	Notification Function	Notification Timing	Important Point
2	Payment Request Completion Notification (Authorize) * Inter-server communication	It is sent when payment request is confirmed (successful or failed).	

Table 3 Result Notification Function of LINE Pay

The following are the result notification items of LINE Pay.

Item Number	Field Name	Item Name	Format and Limitations	Description
1	numberOfNotify	Number of records in notification	Single byte numbers; 4 digits or less	Single notification can contain a maximum of 1,000 records. 1001 onwards records are notified next time.
2	pushTime	Transmission time	Single byte numbers; 14 digits	Date and time when the notification is sent from payment server. yyyyMMddHHmmss format.
3	pushId	Identification ID	Single byte numbers; 8 digits	(Unique) ID assigned while performing push process. Note) ID used in other payment services may be duplicated.
Below mentioned item numbers (4~10) are repeated for the notification records. Further, 4 digits serial number (0000~0999) is assigned after the field name.				
4	orderId	Order ID	Single byte alphanumeric characters; 100 characters or less	Order ID.
5	txnType	Transaction type	* Please refer to the description in the right column.	"Authorize": Payment request completion notification
6	txnTime	Process date and time	Single byte numbers; 14 digits	yyyyMMddHHmmss format.
7	vResultCode	Detailed result code	Single byte alphanumeric characters; 4 characters	For details of process result code, please refer to the “Result code list”.
8	mstatus	Process status	Single byte alphanumeric characters; 8 characters or less	“success”: Normal termination. "failure": Abnormal termination (only in case of inter-server communication). "pending": Pending (only in case of inter-server communication).
9	linepayOrderId	LINE Pay order number	Single byte alphanumeric characters; 19 characters or less	Order number assigned at the LINE Pay system.
10	dummy	Dummy request flag	Single byte numbers; 1 digits	“1” is set in case of dummy order.

(Important points)

- ✧ As pushId (identification ID) may be duplicated with the Id used in other payment services, please do not process as a unique key.
- ✧ The order of items does not necessarily match the order on the table.

Chapter 6 Other - Supplementary Items

6-1 Precautions related to Capture and Cancel

6.1.1 Period for which Capture and Cancel can be Requested

The period for which Capture and Cancel are allowed to request in LINE Pay is shown below.

Request Type	Period for which Request is Allowed
Capture Confirm	Within 30 days from payment completion (authorization).
Cancel	Before capture: Within 30 days from payment completion (authorization). After capture: Within 30 days from day of capture (or day when partial cancel is executed).

Table 4 Period for which Capture and Cancel can be Requested

- Capture confirmation may fail even when authorization is completed. Always send the item or provide the service after capture is successful in LINE Pay.
- Cancel fails if request to card company is rejected (reason of failure is not returned).

6-2 About Charge Back

When a credit card which doesn't belong to that LINE Pay user is registered or used, chargeback is initiated.

In the case of chargeback, the merchant needs to bear the credit card payment risk.

Note) Currently chargeback details are being confirmed. The contents may change later.

6-3 Supplementary Information for Search (Detailed Command Type/Detailed Order Payment Status)

Detailed command (type of processing request) and detailed payment status (status of the given order) are managed in the payment server as internal data in the system.

The merchant does not require this internal information in normal operations. However, merchant may require this internal data in case of getting the status of an order or as reference information at the time of inquiry. This data is returned in search command response.

The list of internal data available by the Search command is shown in below table.

Detailed Command Type		Detailed Order Payment Status when Command is Successful	
Logical Name	Value	Logical Name	Value
Payment approval	PreAuth	Payment Request	Init
Payment quit	QuitAuth	Payment quit	QuitAuth
Authorization	Auth	Authorization	Auth
Cancel (Authorization)	VoidAuth	Cancel (Authorization)	VoidAuth
Authorization and Capture	AuthCapture	Authorization and Capture	AuthCapture
Cancel (Authorization and Capture)	VoidAuthCapture	Cancel (Authorization and Capture)	VoidAuthCapture
Capture	PostAuth	Capture Confirm	PostAuth
Cancel (Capture)	VoidPostAuth	Cancel (Capture)	VoidPostAuth
Invalid Authorization	ExpiredAuth	Invalid Authorization	ExpiredAuth

Table 5 Detailed Command Type and Detailed Order Payment Status

Note) The command types and status not mentioned in the above list also exist.

6.3.1 Regarding Search Result when Payment Confirmation Method is Inter-Server Communication

When payment confirmation method is inter-server communication, decision of "success/failure" of payment should be based on result notification or search result. In case of decision based on search result, use following criteria.

The payment is considered as **“success”** when below transaction exists in “Transaction information” by searching with order ID.

Capture Flag (withCapture) of Authorize Command	Status Code	Detailed Command Type
In case of false or undecided	success	Auth
In case of true	success	AuthCapture

The payment is considered as **“Failed”** when below transaction exists.

Capture Flag (withCapture) of Authorize Command	Status Code	Detailed Command Type
(Not applicable)	-	QuitAuth
In case of false or undecided	failure	Auth
In case of true	failure	AuthCapture

When above transaction does not take place even after 20 minutes after payment request, the recommended period to confirm the status by Search is 20 minutes in order to judge if consumer has stopped the payment.

6-4 Regarding Smartphone Site and Connection between Original App and LINE Pay App

* In this section, the merchants who pay using their own app and specify payment ConfirmType=”0” (inter-server communication) in payment request are considered.

6.4.1 How to Specify appUrlScheme

While using LINE Pay from a smartphone site or from the original app running on a smartphone, after executing payment in the LINE Pay app, it is required to specify the URL scheme to return to the browser or the app. Specify it in appUrlScheme of request message (LinepayAuthorizeRequestDto).

Specify the strings in which the placeholder “<url>” is embedded in appUrlScheme. On the payment server side, it will set the transition URL (URL of payment server) after payment at LINE Pay app and will connect to LINE Pay. Please note that URL will not contain “https://”.

The samples of URL the scheme of main browsers and specification method of the original app are described below.

Start App (Browser)	Android		iOS	
Safari	appUrlScheme	-	appUrlScheme	No need to specify as the default browser is Safari.
	packageName	-		
Chrome	appUrlScheme	-	appUrlScheme	googlechromes:**<url>**
	packageName	com.android.chrome		
Firefox	appUrlScheme	-	appUrlScheme	-
	packageName	org.mozilla.firefox		
Opera	appUrlScheme	-	appUrlScheme	-
	packageName	com.opera.browser		
Opera mini	appUrlScheme	-	appUrlScheme	opera-https:**<url>**
	packageName	com.opera.mini.android		
Original app	appUrlScheme	**Specification parameter of original app that contains original scheme name://”<url>”.** Sample) myApp://**<url>**	appUrlScheme	**Specification parameter of original app that contains original scheme name:”<url>”.** Sample) myApp:**<url>**
	pacakgeName	**Original app package name**		

Table 6 Specification Method of appUrlScheme and packageName

At the time of starting the browser, determine the used browser according to user-agent at the time of access from the user and set the appropriate URL scheme. Set packageName in case of Android.

At the time of starting the original app, obtain placeholder <url>, use the browser in the app and transit to the <url>. After transition, it will judge the success/failure of payment at the payment server and transit to the URL as specified by the merchant. At the time of starting the original app, when **"1"** is set in **useOrignalApp** of request message (LinepayAuthorizeRequestDto), use URL by decoding since the URL encoded string is set in parameter placeholder <url> when LINE Pay app is started.

(Regarding linking LINE Pay transaction number at the time of starting original app)

A LINE Pay order number is assigned to the appUrlScheme when LINE Pay app uses URL scheme and calls the app.

It is assigned in the format of **&transactionId=order number.**

Since the LINE Pay order number is linked to a response message (LinepayAuthorizeResponseDto.linepayOrderId) of the request, it can be used to perform a check on the merchant's app. Below is a sample of actual URL scheme.

(Sample of Android)

myApp://api.veritrans.co.jp/tercerog/webinterface/…..**&transactionId=2015029910000274310**

(Sample of iOS)

myApp:api.veritrans.co.jp/tercerog/webinterface/…..**&transactionId=2015029910000274910**

Note) When URL encoding is set with useOrignalApp = 1, only the part which is specified in <url> before &transactionId=… is encoded.

6.4.2 How to Use redirectAppURL

The method to carry out payment by moving to the LINE app after executing payment request from the merchant's smartphone app is explained here. Please refer to below implementation samples as the implementation in each OS may differ depending on the version.

Sample of Android app

Installation necessity of LINE app and available LINE Pay versions can be confirmed by the following sample codes.

Once LINE app is installed and available LINE Pay versions are confirmed, it will redirect to the LINE Pay payment page.

```
int linePaySupportedVersion = 230;
String paymentUrl = "..."; // Please set redirectAppURL here.
Context context = getActivity();
try {
	PackageManager pm = context.getPackageManager();
	PackageInfo packageInfo = pm.getPackageInfo("jp.naver.line.android", 0);
	int versionCode = packageInfo.versionCode;
	if (linePaySupportedVersion <= versionCode) {
		launchUri(paymentUrl);
	} else {
		confirmLineInstall(context);
	}
} catch (NameNotFoundException e) {
	confirmLineInstall(context);
}

private void confirmLineInstall(Context context) {
	new AlertDialog.Builder(context)
	.setTitle("LINE Pay")
	.setMessage(getString(R.String.linepay_confirm))
	.setCancelable(false)
	.setPositiveButton(getString(R.String.linepay_install), new
  DialogInterface.OnClickListener() {
		@Override
		public void onClick(DialogInterface dialog, int which) {
			launchUri("market://details?id=jp.naver.line.android");
		}
	})
	.setNegativeButton(getString(R.String.linepay_cancel), new
  DialogInterface.OnClickListener() {
		@Override
		public void onClick(DialogInterface dialog, int which) {
		}
	})
	.show();
}

private void launchUri(String uriString) {
	Uri uri = Uri.parse(uriString);
	Intent intent = new Intent(Intent.ACTION_VIEW, uri);
	startActivity(intent);
}
```

File : res/values/Strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    ...
    <String name="linepay_confirm">Supported by Android/iPhone LINE versions 4.8.0 or
higher.</String>
    <String name="linepay_install">Get it now</String>
    <String name="linepay_cancel">cancel</String>
    ...
</resources>
```

Sample of iPhone app

Installation necessity of LINE app can be confirmed by the following sample codes. Once LINE app is installed, it will redirect to LINE Pay payment page.

```
NSString* lineScheme = @"line://";
BOOL installed = [[UIApplication sharedApplication]
                    canOpenURL:[NSURL URLWithString:lineScheme]];
if (installed) {
    UIAlertView *alert =
    [[UIAlertView alloc] initWithTitle:@"LINE Pay"
message:NSLocalizedString(@"linepay.confirm", nil)
                                delegate:self cancelButtonTitle:NSLocalizedString(@"linepay.ok",
nil) otherButtonTitles:nil];
    alert.tag = 1;
    [alert show];
} else {
    UIAlertView *alert =
    [[UIAlertView alloc] initWithTitle:@"LINE Pay"
message:NSLocalizedString(@"linepay.confirm", nil)
                                delegate:self
cancelButtonTitle:NSLocalizedString(@"linepay.cancel", nil)
                                otherButtonTitles:NSLocalizedString(@"linepay.install", nil),
nil];
    alert.tag = 2;
    [alert show];
}

- (void)alertView:(UIAlertView*)alertView clickedButtonAtIndex:(NSInteger)buttonIndex {
    if (alertView.tag == 1 && buttonIndex == 0) {
        NSString *paymentUrl = ...; // Please set redirectAppURL here.
        [self launchUrl:paymentUrl];
    } else if (alertView.tag == 2 && buttonIndex == 1) {
        [self launchUrl:@"itms-
apps://itunes.apple.com/WebObjects/MZStore.woa/wa/viewSoftware?id=443904275&mt=8"];
    }
}

- (void)launchUrl:(NSString*)urlString {
    NSURL *url = [NSURL URLWithString:urlString];
    [[UIApplication sharedApplication] openURL:url];
}
```

File : en.lproj/Localized.Strings

```
"linepay.confirm" = "Supported by Android/iPhone LINE versions 4.8.0 or higher.";
"linepay.ok" = "OK";
"linepay.cancel" = "Cancel";
"linepay.install" = "Get it now";
```

*** Important points at the time of linking the app with iOS10 and above**

At the time of moving from the merchant's smartphone app to the LINE app and requesting payment, CustomURLScheme is called and it will then redirect to LINE app. However, the calling of the app in which CustomURLScheme is used is restricted to iOS10 and above.

Therefore, when the merchant's app is provided, please use below guidelines for implementation.

For specific responding method, please consider any of the following methods.

Method 1. Adding LINE to management list of LSApplicationQueriesSchemes

File : info.plist

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
...
<dict>
  <key>LSApplicationQueriesSchemes</key>
  <array>
    <string>line</string>
  </array>
</dict>
...
</plist>
```

Method 2. UIWebViewDelegate implementation (* When Method 1 cannot be used)

```
import UIKit

class ViewController: UIViewController {

      let webView = UIWebView.init()

      override func viewDidLoad() {
            super.viewDidLoad()
            webView.frame = self.view.frame

            // setup delegate
            webView.delegate = self

            self.view.addSubview(webView)

   // merchant develop code
      }

}

extension ViewController: UIWebViewDelegate {
      // Set in order to execute without any condition in case of the external schemes other than http
or https.
      func webView(_ webView: UIWebView, shouldStartLoadWith request: URLRequest,
navigationType: UIWebViewNavigationType) -> Bool {
            if let url = request.url, url.scheme != "http" && url.scheme != "https" {
                  UIApplication.shared.open(url, options: [:], completionHandler: nil)
                  return false
            }
            return true
      }
}
```

6.4.3 How to Use checkUseBrowser

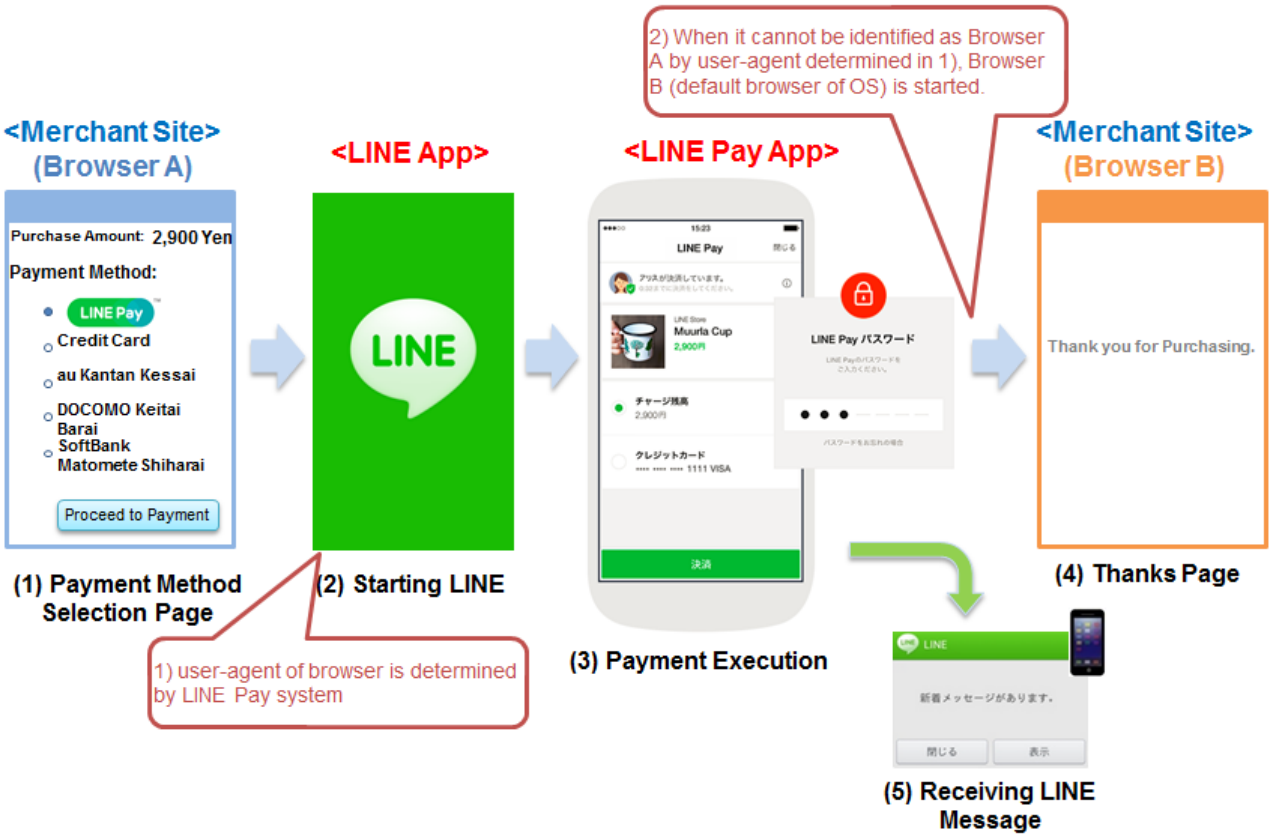
At the time of payment request, when the browser cannot be identified by the LINE Pay system, you may be redirected to the payment result page by the browser (default browser of OS) which is different from the payment request browser.

In case of such a page transition, use checkUseBrowser when any problem occurs such as payment is not completed in merchant site. However, please use it after confirming the actual operation as the consumer's smartphone operation (page transition) differs slightly from the normal operation.

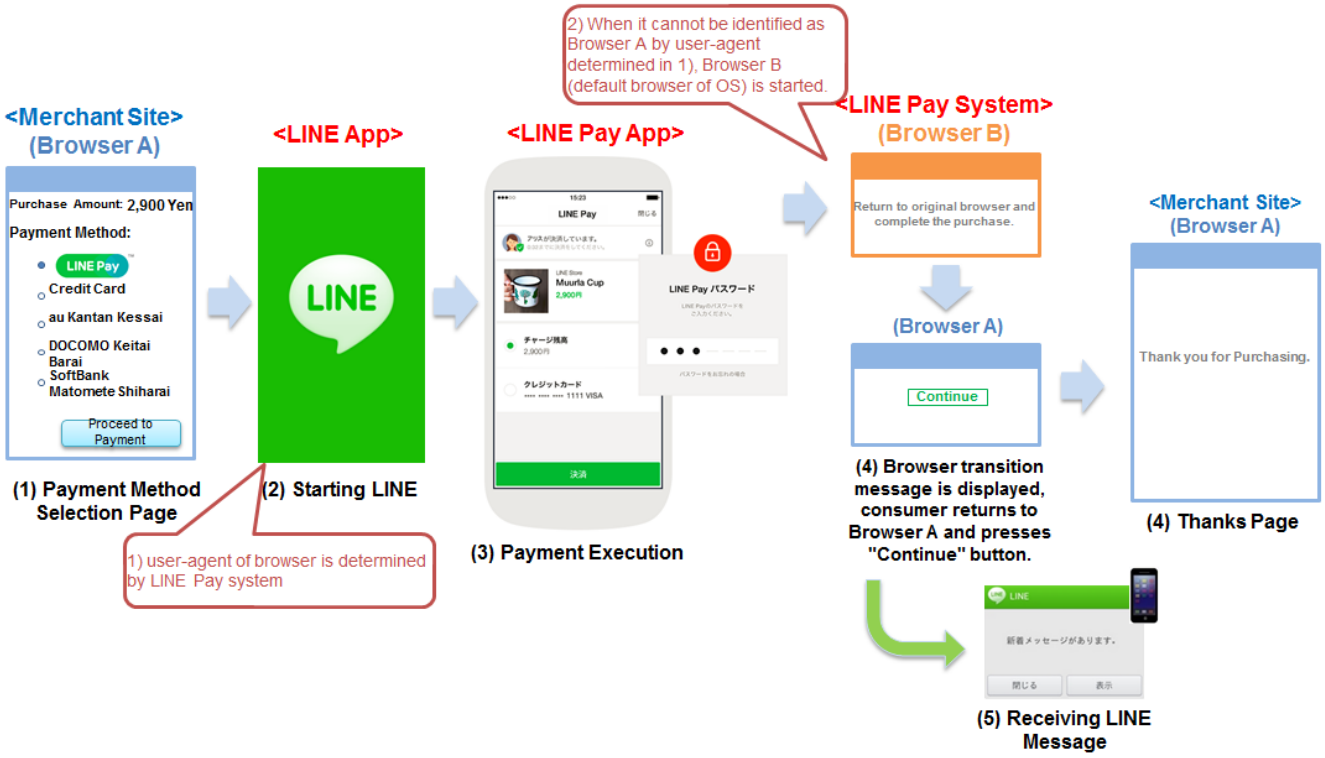
* This function is applicable only when using "redirectWebUrl" of LinepayAuthorizeResponseDto.
  Please note this is not applicable while using "redirectAppUrl".

The following are samples of the page transition.

Transition when "false" is set in checkUseBrowser (same even when checkUseBrowser is not set)



Transition when "true" is set in checkUseBrowser



Chapter 7 Supplementary Information for Integration and Testing

Connect to payment server and refer to the "Integration Test Guide" for details of various procedures to perform tests.

7-1 Simulate Page Transition at the time of Payment Request

We offer a pseudo environment for the testing of LINE Pay. The operation where LINE app is used cannot be checked in this pseudo environment. However, it may be used in development and testing at the merchant site as the interface between payment server and merchant site is the same as that of the live environment.

Discrepancy in access environment (PC/smartphone browsers, or merchant's apps) cannot be fully reproduced. However, browser base transition and merchant's app transition can be simulated by calling URL returned as response of payment request (Authorize).

Further, when oneTimeKey is specified by Authorize command and "1" (inter-server communication) is specified in paymentConfirmType, simulation is carried out by the method in 7-2 and not by below page transition, since redirectWebUrl and redirectAppUrl are not returned.

7.1.1 Page Transition of PC or Smartphone (Browser)

Below pseudo environment is displayed when consumer browser is transited to redirect URL (LinepayAuthorizeResponseDto.redirectWebUrl) which is returned as a result of Authorize command.

Figure 9 Pseudo Environment (Dummy - LINE Pay - PC Login Page)

Image of LINE authentication page is displayed. Click on "Login" button and go to next page.

This page is not displayed in the case of smartphones or when oneTimeKey is specified.

Figure 10 Pseudo Environment (Dummy - LINE Pay - Payment Page)

The standby page displayed on a PC (or smartphone) browser is displayed on the upper half part of the image above.

In case of the Live payment, payment is made by using LINE Pay app (smartphone) after this standby page is displayed.

In the pseudo environment, an image of the smartphone page is displayed on the lower half part of the page. Confirm subsequent transitions by selecting "Purchase" or "Cancel".

When paymentConfirmType is "0" (communication via browser), transition to merchant's payment completion page (URL at the time of success) or URL at the time of cancellation is possible.

When paymentConfirmType is "1" (inter-server communication), order status can be confirmed either by payment request completion notification or by Search.

For transition to merchant error page, please refer to the “7-3 Simulate Error of Payment Request, Capture, Cancel Command”.

7.1.2 Page Transition when Smartphone (Merchant App) is Used

Below pseudo environment is displayed when merchant app is redirected to URL (LinepayAuthorizeResponseDto.redirectAppUrl) returned as a result of Authorize command.

In actual operation, LINE app will start. But in pseudo environment, LINE app image is displayed on browser.

Figure 11 Pseudo Environment (Dummy - LINE Pay - LINE App Password Input)

Figure 12 Pseudo Environment (Dummy - LINE Pay - LINE App - Payment Page)

7-2 **Simulate Payment Request at the time of using oneTimeKey when Payment Confirmation Method is Inter-Server Communication**

When oneTimeKey is specified by Authorize command and "1" (inter-server communication) is specified in paymentConfirmType, payment request can be simulated by the following flow as page transition is not performed.

1. Request Authorize command (specify arbitrary value in oneTimeKey and “1” (inter-server communication) in paymentConfirmType).
2. Status will change after around 30 seconds.
   - 2.1 When result notification URL is specified, confirm the notification contents, since payment request completion notification will be sent.
   - 2.2 When result notification URL is not specified, confirm by Search.
3. Confirm the order status and inform the consumer that payment is completed as per the merchant's specification.

7-3 **Simulate Error of Payment Request, Capture, Cancel Command**

Error can be intentionally generated by adjusting the last single digit of request amount (“amount”) of Authorize, Capture and Cancel command.

Last single digit of request amount and correspondence table of returned vResultCode are shown below.

<table>
<tr><th>Last single digit of request amount</th><th>Authorize (Request)</th><th>Transit to error URL</th><th>Result notification at the time of inter-server communication</th><th>Capture (Capture)</th><th>Capture (Partial capture)</th><th>Cancel (Cancel)</th><th>Cancel (Partial cancel)</th></tr>
<tr><td>0</td><td rowspan="10">I001
(successful)</td><td rowspan="3">I001
(successful)</td><td rowspan="3">I001
(successful)</td><td rowspan="5">I001
(successful)</td><td rowspan="5">I001
(successful)</td><td rowspan="6">I001
(successful)</td><td rowspan="6">I001
(successful)</td></tr>
<tr><td>1</td></tr>
<tr><td>2</td></tr>
<tr><td>3</td><td>IG02 (error)</td><td>IG02 (error)</td></tr>
<tr><td>4</td><td rowspan="6">I001
(successful)</td><td rowspan="6">I001
(successful)</td></tr>
<tr><td>5</td></tr>
<tr><td>6</td><td>IG03 (error)</td><td>IG03 (Error)</td></tr>
<tr><td>7</td><td rowspan="3">I001
(successful)</td><td rowspan="3">I001
(successful)</td></tr>
<tr><td>8</td><td>IG03 (error)</td><td>IG03 (error)</td></tr>
<tr><td>9</td><td>I001
(successful)</td><td>I001
(successful)</td></tr>
</table>

- When amount is not specified for Capture, Cancel, full amount is requested.
- In the case of an order having the last single digit as 3, request of Authorize command will be successful. However, it will redirect to merchant’s error URL with transition of consumer page after that.
- “oneTimeKey is invalid” error can be generated by using oneTimeKey other than the one at the time of simulation when above amount is used. Its condition is as follows.

When “999999999999” is specified in oneTimeKey of Authorize command, IG08 error code is returned.

- Returned error code (vResultCode) may be changed without prior notice in future.