POLi™ Implementation Guide

POLi LINK V1.5

Disclaimer and limitation of Liability

This Merchant Implementation Guide sets out the proper procedures for the operation and use of the POLi™ service offered by Centricom Pty Limited. This service, and your access to them, is governed by the terms and conditions of the Centricom Merchant Agreement. Only persons who have signed and agreed to be bound by the Centricom Merchant Agreement may use the transaction processing services offered by Centricom Pty Limited (including POLi™).

Trademarks

Centricom, the Centricom Logo, POLi™, and other trademarks, service marks, and logos are registered or unregistered trademarks of Centricom Pty Limited in Australia and in foreign countries. Other trademarks and service marks in this document are the property of their respective owners.

User Guide for Merchants – Generating POLi LINKs via the POLi™ Console

Logging in to the POLi™ Console

The following URL provides access to the POLi™ Console:

Country	URL
Australia / NZ	https://consoles.apac.paywithpoli.com

1   POLi LINKS

POLi LINKS are a new method for merchants to collect payments.

This method enables merchants to request payment from an individual outside of the 'checkout' scenario. Merchants are able to generate a POLi Link which takes the form of a short URL, send this to the customer via email or place it for example within a pdf invoice. When the POLi Link is clicked the user is taken to a POLI payment page pre-populated with relevant reference and amount data, as specified by the merchant when creating the POLi Link. This method is ideal for billing scenarios and also for instances where payment is required outside of their traditional website or eCommerce platform.

1.1   Single POLi Link

The Single POLi Link function can be used to generate individual POLi Links without the need to create a batch file.

See below for the details of each item on the Single POLi Link page.

NOTE: The information icon next to each field label provides information about the field on mouseover.

1. **Request Type** – this can be Manual or Email. Selecting 'Email' requires additional information (customer name and customer email address) and triggers an email to the specified email address when the Payment URL is generated.

2. **Amount** – this is the amount of the POLi Link. This must be a valid currency value with up to two decimal points. The amount must be between 1.00 and 10000.00.

3. **Your Reference** – this is the Merchant reference for the payment. This field is displayed on your reports under the *Merchant Reference* column. This is a mandatory field and can contain up to 50 characters.

4. **Merchant Data** – this field can be used to specify additional transaction information or reference data. This field is displayed on your reports under the *Merchant Data* column. This is an optional field and can contain up to 100 characters.

5. **Confirmation Email** – check this box if you wish to receive a confirmation email when successful payment is made. The email is sent to your support email address as specified in your merchant configuration.

6. **Customer Reference** – check this box if you require the customer to enter reference details for the payment. This field is displayed on your reports under the *Merchant Data* column.

7. **Multiple Payment** – check this box if you wish to allow this POLi Link to be used multiple times. This link will not expire when used by a customer (however it can be set to expire on a specific date or manually via Manage POLi Links).

8. **Payment URL Expiry** – if you want to set an expiry date for a POLi Link request, it can be specified here. The generated POLi Link cannot be used on or after this expiry date.

If this field is left blank, no expiry date is set against the POLi Link.

9. **Payment URL result field** – when the Generate URL button is clicked and all data is successfully validated, the resulting Payment URL is displayed here. This URL can be copied using the copy icon below the field.

10. **Clear button** – clicking this button clears all data and URL information from the page

11 **Generate URL button** –clicking this button will validate the data entered in the fields on the page, and if successfully validated, will generate the Payment URL.

The Email Request Type contains two further fields (see below):

12. **Recipient Name** – The name of the customer receiving the POLi Link. This information is included in the email sent to the customer. This can only contain letters, spaces, dashes and apostrophes, and is mandatory if Request Type is email.

13. **Recipient Email** - The email address of the customer receiving the POLi Link. This must be in a valid email format (e.g. abc@abcdef.com), and is mandatory if Request Type is email.

Once all fields have been completed and contain valid data, clicking the Generate Payment URL button will populate the Payment URL result field with the URL for the POLi Link. This can be copied to the clipboard for pasting into a document or email by clicking the copy icon below the field. Additionally, if the Request Type is email, an email will be sent to the Recipient Email address specified.

1.2   Multiple POLi Link

The Multiple POLi Link function can be used to generate a one or more POLi Links via uploading a batch file, in either Excel or CSV format.

Validation rules for each item in the batch file is as per the Single POLi Link.

See below for the details of each item on the Multiple POLi Link page:

1. **Download Templates** – these two icons link to downloadable templates in Excel and CSV format.

2. **Select a File** – to upload a Multiple POLi Link batch file, click "Browse...", locate the file and click the Upload File button. Once upload is complete, the Last File Uploaded field will display the filename.

3. **Clear button** – clicking this button clears all data from the page, including URL information

4. **Upload File button** – Once a filename is displayed in the Select A File field, clicking this button will initiate file upload of the selected filename.

When a file has been successfully uploaded and validated, a successful message is displayed and the Generate URLs button is shown:

5. **Successful validation message** – when upload is complete and all data in the uploaded file has been successfully validated, this message is displayed

6. **Valid POLi Link data notification** – a green tick is displayed next to each successfully validated line item in the batch file

7. **Generate URLs button** – this button is displayed when all data has been successfully validated. Clicking this button will generate the Payment URL for each line item in the batch file.

Data has been validated successfully...

⑤ Select a File [_____] Browse...
Last File Uploaded: demo.csv

Clear   Upload File   Generate URLs ⑦

Status	Payment Request Type	Recipient Name	Recipient Email	Payment Amount	Payment Reference	Confirmation Email	Customer Reference	Is Multiple	Expiry Date	Merchant Data	Payment URL
✓	Manual	John Smith	support@pricebuster.com	25.5	RefNo12345	yes	yes	no			
✓	Email	Jane Smith	support@pricebuster.com	122.22	RefNo67890	yes	no	yes	1/02/2012	Sub 1	

⑥

On clicking the Generate URLs button, the Payment URL for each line item in the batch file is displayed, and an email is sent to the Recipient Email Address for each POLi Link Type of Email.

8. **Payment URL** – the generated URL for the POLi Link is displayed here

9. **Copy URL button** – to copy the Payment URL for an individual line item to the clipboard, for pasting into a document or email, select the relevant line item and click this button

10. **Export** – click the icons to export the complete list of POLi Links displayed, including the generated Payment URLs, in either Excel or CSV format.

Payment URL's generated successfully...

Select a File [_____] Browse...
Last File Uploaded: demo.csv

Clear   Upload File   Export [Excel] [CSV] ⑩

ℹ Select a Row and Click this **Copy URL** button to copy Payment URL from the selected row ⑨

⑧

Status	Payment Request Type	Recipient Name	Recipient Email	Payment Amount	Payment Reference	Confirmation Email	Customer Reference	Is Multiple	Expiry Date	Merchant Data	Payment URL
✓	Manual	John Smith	support@pricebuster.com	25.5	RefNo12345	yes	yes	no			http://poli.to/eLAu
✓	Email	Jane Smith	support@pricebuster.com	122.22	RefNo67890	yes	no	yes	1/02/2012	Sub 1	http://poli.to/NIUp

⑥

If any line item within the batch file contains invalid data, an error message is displayed and the details of the invalid data is displayed:

11. **Validation Error message** – this message will display if any of the data contained within the file does not pass validation on upload

12. **Invalid data notification** – this icon will display in the Status column for each line item that contains invalid data

13. **Invalid data detail** – the specific data that has failed validation will be displayed as crossed-out red text. This data will require correction and the file will need to be re-uploaded with the corrected data.

⊗ Data has some validation errors, please review and upload again...

(11)

| Select a File | | Browse... |

Last File Uploaded: demoerror.csv

Clear   **Upload File**

ℹ Mouse over on ⊗ to view all the related errors at once. Mouse over on text to view individual error.

Status	Payment Request Type	Recipient Name	Recipient Email	Payment Amount	Payment Reference	Confirmation Email	Customer Reference	Is Multiple	Expiry Date	Merchant Data	Payment URL
⊗	Manual	John Smith	support@pricebuster.com	25.5	~~invalid~~	yes	yes	no			
⊗ (12)	Email	Jane Smith	~~supportpricebuster.com~~	~~0.12~~ (13)	RefNo67890	yes	no	yes	1/02/2012	Sub 1	

If you mouseover the invalid data notification icon to the left of the line item, a tooltip will display containing the reason(s) for the data failing validation:

Status	Payment Request Type	Recipient Name	Recipient Email
⊗			
⊗	1) Invalid **Recipient Email** address. 2) Invalid **Amount**. **Amount** must be a decimal value between 1.00 and 10000.00.		

This information can also be obtained if you mouseover the crossed-out data in each line item:

Invalid **Recipient Email** address.

~~supportpricebuster.com~~   0.

1.3   Manage POLi Links

The Manage POLi Links page can be used to track all existing POLi Link requests for your merchant. This includes both active POLi Links and those that have expired, either due to being used, passing their expiry date, or by being manually expired.

See below for the details of each item on the Manage POLi Links page:

1. **Request Type** – The type of POLi Link request you are searching for. Can be selected from "All", "Manual" or "Email".

2. **Payment Amount** – The amount of the POLi Link(s) you are searching for. Can be any currency value. Only the specified amount will be returned in the search results

3. **Payment Reference** – The Payment Reference of the POLi Link(s) you are searching for. Some or all of the Reference can be specified and all POLi Links containing the search term will be returned.

4. **Link Status** – The status of the POLi Link request you are searching for. Can be selected from "All", "Active" or "Expired".

5. **Expiry Date** – The expiry date of the POLi Link(s) you are searching for. Can be any valid date value. Only the specified date will be returned in the search results. For expired POLi Links, this will be the date the Link expired (as opposed to any originally specified expiry date)

6. **Recipient Name** – The Recipient Name of the POLi Link(s) you are searching for. Some or all of the Name can be specified and all POLi Links containing the search term will be returned. Note this field will only be applicable to Email requests.

7. **Recipient Email** – The Recipient Email address of the POLi Link(s) you are searching for. Some or all of the Email address can be specified and all POLi Links containing the search term will be returned. Note this field will only be applicable to Email requests.

8. **Date Created** – The creation date of the POLi Link(s) you are searching for. Can be any valid date value. Only the specified date will be returned in the search results.

9. **Expire Selected Links button** – clicking this button will manually expire any currently selected POLi Links. Note that once a POLi Link is expired it cannot be made active again.

10. **Search button** – clicking this button submits the specified search criteria and returns all matching transactions.

11. **Clear button** – clicking this button clears all search filters and displays the complete list of POLi Links.

12. **Select Payment checkbox** – Check this box to select one or more POLi Links that you want to manually expire

13. **Expired POLi Link icon** – This icon indicates which POLi Links in the list have expired.

There is also a **POLi Link Report** available in Reporting. The search fields are as per the above but the report details are read-only, and the data can be exported.

2   Generating POLi Links via web services / API

2.1   PaymentAPI GenerateURL Web Service

2.1.1   Description

The PaymentAPI GenerateURL web service is used to generate a POLi Link URL for initiating a custom payment transaction. When supplied with a valid Merchant Authentication data and Payment data (reference, amount etc.), PaymentAPI web service generates a short Payment URL. When the Payment URL is clicked the user is taken to a POLi payment page pre-populated with relevant reference and amount data, as specified by the merchant when generating the Payment URL. The API exposes a service method GenerateURL to generate Payment URL which takes an input of type PaymentDataRequest and return output of type PaymentDataResponse.

2.1.2   PaymentDataRequest

Field	Description	Data Type	Field Length	Required	Allowable Values
MerchantCode	A code that uniquely identifies the merchant generating the Payment URL.	String	Max: 20	Y	The merchant code provided by your POLi™ Implementation Manager
Authentication Code	Merchants must authenticate themselves to POLi™ via use of an authentication code (complex password).	String	Max: 500	Y	The Authentication Code must match what is set up in the POLi™ system via the POLi™ Console. The following special characters are allowed: !@#$%^+
RequestType	A Value determines whether to send an email or not after generating url. If set to Email, an email is sent to RecipientEmail address with payment details.	String	Max: 20	Y	Manual or Email.
PaymentAmount	Merchant requesting customer total amount for the payment	Decimal	Max:(7,2)	Y	Minimum is 1.00 and Maximum is 10000.00
PaymentReferen	Merchant	String	Max: 50	Y	The Merchant

Field	Description	Data Type	Field Length	Required	Allowable Values
ce	Reference/reconciliation data				reference
MerchantData	Additional data a merchant can supply to further identify this payment.	String	Max:100	Y	Any string/text or leave it empty.
ConfirmationEmail	A value determines whether to send a payment confirmation email or not to Merchant registered support email address.	String	Max: 3	Y	Yes or No.
CustomerReference	A value determines whether to provide an option or not to the customer, the ability to add additional reference data.	String	Max: 3	Y	Yes or No.
IsMultiple	Indicates whether the generated Payment URL can be used multiple times.	String	Max:3	Y	Yes or No.
ExpiryDate	The date and time the generated Payment URL will expire	DateTime	Max:19	N	Default value is: 31/12/9999 If a value is supplied, the timestamp must conform to this format: dd/mm/yyyy The value must be future-dated
RecipientName	Customer name. Only provided when RequestType is Email.	String	Max: 50	Y (conditional)	Default value is Empty. This cannot be empty when RequestType is Email.
RecipientEmail	Customer Email address. Only provided when RequestType is Email.	String	Max: 100	Y (conditional)	Default value is Empty. This cannot be empty when RequestType is Email.

2.1.3   PaymentDataResponse

Field	Description	Data Type	Possible Values
PaymentURL	The generated Payment URL	String	A short URL with token
Errors	List of Errors occurred while generating Payment URL if any.	Array	List of Error objects, each Error object has the following members Code. Message. Field.

POLi™ Payment API

POLi™ provides a Payment API consisting of the following web services:

GenerateURL – used to generate a Payment URL

In order to access the Payment API, you must use the secure URLs provided by the POLi™ Implementation Manager at the time of registration

XML Web Services SOAP
Australia / New Zealand:

https://paymentapi.apac.paywithpoli.com/PaymentRouterService.svc

Generating a Payment URL Sample Code (SOAP)

```
…
//PaymentAPI is the sample Namespace generated by Service Reference(or svcutil).
//If you wish to send an email with the link, set RequestType="Email", //RecipientName and
RecipientEmail with valid values.
PaymentAPI.PaymentDataRequest pdr = new PaymentAPI.PaymentDataRequest();
pdr.MerchantCode = "PriceBusterDVD_AU";
pdr.AuthenticationCode = "MyPassword";
pdr.RequestType = "Manual";
pdr.PaymentAmount = 123;
pdr.PaymentReference = "Payment Reference";
pdr.ConfirmationEmail = "No";
pdr.CustomerReference = "No";
pdr.IsMultiple = "No";
pdr.ExpiryDate = "12/12/2012";                   // This is optional
pdr.MerchantData = "Merchant additional data";
PaymentAPI.PaymentDataResponse res = null;
String url = string.empty;

//PaymentAPIServiceClient is generated by Service Reference and using SoapService endpoint
using (PaymentAPI.PaymentAPIServiceClient svc = new
       PaymentAPI.PaymentAPIServiceClient("SoapService"))
{
    res = svc.GenerateURL(pdr);
}

if(res != null)
```

```
{
    if(res.Errors.length > 0)
    {
        String message = String.Format(
            "Error while generating Payment URL.\\nError code: {0}\\nError Message:
            {1}\\n\\nPlease try again.", res.Errors[0].Code, res.Errors[0].Message);

        showAlert(message);
        return;
    }else
        url = res.PaymentURL;
}else
{
    showAlert("Error: Could not generate a Payment URL.\\n\\nPlease try again.");
    return;
}
...
```

Australia / New Zealand:

https://paymentapi.apac.paywithpoli.com/PaymentRouterService.svc/Xml/generate

Generating a Payment URL Sample Code (REST)

GenerateURL Request

To generate a Payment URL, the merchant performs a HTTP(S) post to the POLi™ PaymentAPI REST URL with the request content-type set to 'text/xml' and the following XML data in the request body.

Manual request (RequestType = 'Manual')

```xml
<?xml version="1.0" encoding="utf-8"?>

<PaymentDataRequest xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

        <MerchantCode>PriceBusterDVD</MerchantCode>

        <AuthenticationCode>MerchantPassword</AuthenticationCode>

        <RequestType>Manual</RequestType>

        <PaymentAmount>123.11</PaymentAmount>

        <PaymentReference>LandingPageReferenceText</PaymentReference>

        <ConfirmationEmail>No</ConfirmationEmail>

        <CustomerReference>No</CustomerReference>

        <IsMultiple>No</IsMultiple>

        <ExpiryDate></ExpiryDate>

        <RecipientName></RecipientName>

        <RecipientEmail></RecipientEmail>

        <MerchantData>Merchant additional data</MerchantData>

</PaymentDataRequest>
```

Email request (RequestType = 'Email')

```xml
<?xml version="1.0" encoding="utf-8"?>

<PaymentDataRequest xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

        <MerchantCode>PriceBusterDVD</MerchantCode>

        <AuthenticationCode>MerchantPassword</AuthenticationCode>

        <RequestType>Email</RequestType>

        <PaymentAmount>123.11</PaymentAmount>

        <PaymentReference>LandingPageReferenceText</PaymentReference>

        <ConfirmationEmail>No</ConfirmationEmail>

        <CustomerReference>No</CustomerReference>

        <IsMultiple>No</IsMultiple>

        <ExpiryDate>31/12/2012</ExpiryDate>

        <RecipientName>PayerName</RecipientName>

        <RecipientEmail>PayerEmailAddress</RecipientEmail>

        <MerchantData>Merchant additional data</MerchantData>
```

```
</PaymentDataRequest>
```

GenerateURL Response

Providing the request is authenticated and configured successfully, the merchant will receive the generated short payment URL in the XML data returned in the HTTP(S) response body as follows:

```xml
<?xml version="1.0" encoding="utf-8"?>

<PaymentDataResponse>

    <Errors />

    <PaymentURL>poli.to/GMEY</PaymentURL>

    <MerchantData>Merchant additional data</MerchantData>

</PaymentDataResponse>
```