

Developer Guide for checkout using BlackStone Payment Integration

Payment Links

A Payment Link is the base behind Blackstone Payment Integration, its fields are:

Field	Type	Description
Id	guid	The identifier of the payment link
MerchantId	int	Id of the Merchant that created the Payment Link
Amount	decimal	The amount charged
Description	string	A description for additional information
Type	int	Type of the Payment Link
CreatedOn	DateTime	Date of Creation
PaidOn	DateTime	Date in which it was paid
DisabledOn	DateTime	Date in which it was disabled
Status	int	Status of the Payment Link
Active	bool	Specify whether Payment Link is active or not
ServiceReferenceNumber	string	Service reference number
InvoiceNumber	string	Invoice number
Link	string	Url to proceed with the payment of the Payment Link

Payment Link Type

Payment Links can be created for a fixed or an open amount to be defined later by the client. This idea is represented with an enum of Payment Link possible types. Said types are:

Value	Type
0	Open
1	Fixed

Payment Link Status

Payment Links can be either Paid or Unpaid, this is what we call the Payment Link Status. This idea is represented with an enum of Payment Link possible status:

Value	Status
-------	--------

Value	Status
0	Unpaid
1	Paid

API Endpoints

Common Request and Response Values

Request

Your requests to Blackstone api share some basic fields in their body, those fields refer to your company specific credentials through which the system will manage, among other things, authentication. Said fields are:

Field	Type	Description
mid	int	Id assigned to you by Blackstone as a Merchant
UserName	string	User name
Password	string	Password
AppType	int	App Type
AppKey	int	App Key
cid	int	Cashier Id

They must be placed as part of the body of your request, just like:

```
{
  "mid": xxx,
  "UserName": xxx,
  "Password": xxx,
  "AppType": xxx,
  "AppKey": xxx,
  "cid": xxx
}
```

Response

In the same way, responses share three basic fields:

Field	Type	Description
ResponseCode	int	The response code that the server is returning
Msg	string	A short message explaining the response
verbiage	string	Extended explanation of the response

A successful response will look like this:

```
{
  "ResponseCode": 200,
  "Msg": [
    "SUCCESS."
  ],
  "verbiage": null
}
```

Error codes

Requests can fail and the server will return a specific response code to provide insights of what failed:

Code	Meaning
1	Invalid Credentials
11	Unauthorized operation for Merchant
28	Application has no permissions to execute that task
300	System Error
404	Resource Not Found
506	Invalid parameter

GetPaymentLink

Endpoint URL: <https://app.blackstoneonline.com/api/PaymentLinks/GetPaymentLinkById>

HTTP Method: GET

Description: This endpoint will look for a Payment Link with the specified Id or Invoice Number

Request Params: The `id` of the Payment Link to look for, you could also use the Invoice Number of the payment as value for this param.

Field	Description
<code>id</code>	Id or Invoice Number of the Payment Link to look for

Request Body: No additional data, only the [shared request body](#)

Response:

Added to the [shared response data](#) will be included a [Payment Link](#) object.

Field	Description
<code>PaymentLink</code>	The requested Payment Link object

Error Codes:

The [404 error code](#) in this endpoint is triggered if there isn't a Payment with the specified Payment Link Id.

Example

You should create the request like:

```
method: GET
url: https://app.blackstoneonline.com/api/PaymentLinks/GetPaymentLink?id=aa11-
bb22-cc33-dd44-ee55
body: {
  "mid": 12345,
  "UserName": "user_name",
  "Password": "password",
  "AppType": 1,
  "AppKey": "12345",
  "cid": 1
}
```

Then a successful response would look like:

```
{
  "ResponseCode": 200,
  "Msg": [ "SUCCESS." ],
  "verbiage": null
  "PaymentLink": {
    "Id": "aa11-bb22-cc33-dd44-ee55",
    "MerchantId": 12345,
    "Amount": "99760.78",
    "Description": "Payment of some apples",
    "Type": 1,
    "CreatedOn": "2024-08-22T01:42:06.38",
    "PaidOn": null,
    "DisabledOn": null,
    "Status": 0,
    "Active": true,
    "ServiceReferenceNumber": null,
    "InvoiceNumber": null,
    "Link": "https://app.blackstoneonline.com/payments/link/aa11-bb22-cc33-
dd44-ee55"
  }
}
```

If none of the Payment Links have the specified Id or Invoice Number then the response will look like:

```
{
  "ResponseCode": 404,
```

```
"Msg": [ "PAYMENT LINK NOT FOUND." ],
"verbiage": null,
"PaymentLink": null
}
```

AddPaymentLink

Endpoint URL: <https://app.blackstoneonline.com/api/PaymentLinks/AddPaymentLink>

HTTP Method: Post

Description: This endpoint creates a new Payment Link

Request Params: None

Request Body: Added to the [shared request body](#) is needed a short version of the [Payment Link object](#) that includes only [Amount](#), [Description](#) and [InvoiceNumber](#). If [Amount](#) field is set to null an [open type](#) Payment Link will be created, otherwise it will be a [fixed type](#).

Field	Description
PaymentLink	The Payment Link short version

Response:

Added to the [shared response data](#) will be included the recently created [Payment Link](#) object.

Field	Description
PaymentLink	The requested Payment Link object

Error Codes:

The [506 error code](#) in this endpoint is triggered by an invalid amount format.

Example

You should create the request like:

```
method: POST
url: https://app.blackstoneonline.com/api/PaymentLinks/AddPaymentLink
body: {
  "mid": 12345,
  "UserName": "user_name",
  "Password": "password",
  "AppType": 1,
  "AppKey": "12345",
  "cid": 1,
  "PaymentLink": {
    "Amount": "10.50",
    "Description": "Created Test Payment Link 1",
```

```
    "InvoiceNumber": "8888"  
  }  
}
```

Then a successful response would look like:

```
{  
  "ResponseCode": 200,  
  "Msg": [ "PAYMENTLINK SUCCESSFULLY CREATED." ],  
  "verbiage": null,  
  "PaymentLink": {  
    "Id": "aa11-bb22-cc33-dd44-ee55",  
    "MerchantId": 12345,  
    "Amount": "10.50",  
    "Description": "Created Test Payment Link 1",  
    "Type": 1,  
    "CreatedOn": "2024-11-13T13:43:19.3431802-05:00",  
    "PaidOn": null,  
    "DisabledOn": null,  
    "Status": 0,  
    "Active": true,  
    "ServiceReferenceNumber": null,  
    "InvoiceNumber": "8888",  
    "Link": "https://app.blackstoneonline.com/payments/link/aa11-bb22-cc33-  
dd44-ee55"  
  }  
}
```

Step-by-Step guide of use

Applications can easily use BlackStone Payment Integration to checkout their products. You should first contact us to get [all the necessary credentials](#) to start consuming our Payment Link API. Once you have them you are ready to start issuing payment requests to your clients.

Given that it's wanted to checkout a product the communication workflow would be as follows:

1. Create the Payment Link

To create a Payment Link you should consume the [create Payment Link endpoint](#) specifying the **Amount** to charge, a **Description** of the product/service and an **Invoice Number** that makes sense to the application logic. It will return the details of the just created Payment Link and a **BlackStone URL** to execute the payment that looks like <https://app.blackstoneonline.com/payments/link/aa11-bb22-cc33-dd44-ee55>.

2. Use the returned URL

With the returned **BlackStone URL** you can build the button in your application that proceeds with the payment. You can add to that url a new parameter called **returnURL** and specify there the url to which the logic of your application wants to be returned once the payment is submitted by the user. The new

url should look like `https://app.blackstoneonline.com/payments/link/aa11-bb22-cc33-dd44-ee55?returnUrl="www.MyStore.com/User/Products/aa11-bb22-cc33-dd44-ee55"`

3. Check for Payment Status

Status of Payments can be checked consuming the [get Payment Link](#) endpoint. The Payment Link will be paid once its status changes to **Paid**. The application can use the status of the Payment Link to make changes in its logic given the **Paid** or **Unpaid** status

And that's it. Congratulations!! You have successfully integrated Blackstone Payment System!.

Your application can also make use of other available endpoints to edit your payment link in the case some data should be updated, disable it so it's no longer available for payment, and get a list of all PaymentLinks issued by your application using BlackStone. See more details about them [here](#).