DIGITZS

Making Payments Painless - Now that Rocks.

API USER MANUAL

Prepared by CorSource

April 24, 2019

Contents

Introduction

Welcome to the API User Manual. Our API provides Digitzs customers with platforms for processing ticket payments for sporting events, donations for non-profit organizations, and rent for property managers. It can be used for merchant payments, split payments, automatic monthly payments, and refunds. The API also provides direct access to authentication, authorization, customer, merchant, token, and payment information.

This manual is organized into four main parts:

- Part One – API Basics. Overviews the structure of the API.
- Part Two – Integration. Explains how to configure a computer to run the API.
- Part Three – Endpoints. Describes the endpoints available with the API.
- Part Four – Codes. Provides a set of tables with details about API-related status codes.

Note: Contact Digitzs at techsupport@digitzs.com or 855.563.4489 if you have any questions about our API.

Part One – API Basics

This section provides some high-level information about the Digitzs API.

- Currency. Describes the processing of foreign currencies.
- SSL/TLS. Describes the use of Secure Socket Layer (SSL) and Transport Layer Security (TLS).
- X.509 Certificates. Describes the use of X.509 certificates.
- REST. Describes the use of the Representational State Transfer (REST) interface.
- JSON. Describes the use of the JSON protocol.
- HTTP. Describes the use of the HTTP protocol.

Currency

Digitzs uses the ISO 4217 standard three-character currency codes to represent various supported currencies. See Currency Codes for more information.

Secure Socket Layer (SSL) and Transport Layer Security (TLS)

Digitzs recognizes the importance of handling financial transactions in a secure manner and ensures that the best transmission security is available. Digitzs ensures that API request information is transmitted using the latest SSL/TLS encryption practices. SSL/TLS creates a secure connection between client and server over which encrypted information is sent. Digitzs hosts the SSL/TLS certificate for this connection type.

X.509 Certificates

The Digitzs API methods are performed over an SSL channel hosted the X.509 certificate. Specific API methods require the usage of a X.509 certificate as an added measure of security where Digitzs has deemed the method to carry additional risk factors. This certificate is an added layer of security and protection from loss in the event of credential theft.

Representational State Transfer (REST) Interface

REST streamlines access to the Digitzs API. It combines a Base Uniform Resource Identifier (URI) with mapped Resource URI elements and standard Hypertext Transfer Protocol (HTTP) methods for most API methods. When using the GET or DELETE HTTP methods, the API request does not require additional information to be passed. When using POST and PUT HTTP methods, information is sent in the form of JSON objects. The API returns a JSON object as a response to each API request regardless of the HTTP method used.

JSON Protocol

JavaScript Object Notation (JSON) is a common, language-independent data format that provides a simple text representation for arbitrary data structures. See json.org for more information.

HTTP Protocol

The API service conforms to the Hypertext Transfer Protocol (HTTP)/1.1 specification.

HTTP Headers

Header Field	Value
Authorization	Basic TXlUZXJtSWQ6TXlDZXJ0U3Ry
x-api-key	Bearer EmBfRrvTez2CWkEvP4kxn5merinmzc1y3g3VGNmk
Content-Type	application/json

Table 1 – HTTP Headers

HTTP Request Methods

Method	Description
CONNECT	Used to convert the request connection to a transparent Transmission Control Protocol/Internet Protocol (TCP/IP) tunnel.
DELETE	Used to delete the specified resource.
GET	Used to request a representation of the specified resource.
HEAD	Used to ask for a response identical to that of a GET request without the response body.

Method	Description
OPTIONS	Used to return the HTTP methods that the server supports for the specified Uniform Resource Locator (URL).
PATCH	Used to apply partial modifications to a resource.
POST	Used to request that the server accept the entity enclosed in the request as a new subordinate of the web resource identified by the URI.
PUT	Used to request that the enclosed entity be stored under the supplied URI.
TRACE	Used to echo the received request so that a client can see any changes or additions have been made by intermediate servers.

Table 2 – HTTP Request Methods

Objects

An unordered collection of name/value pairs where the keys are strings.

Name	Description
attributes	A key for grouping identification and payment information.
bank	A key for grouping banking information.
bankInfo	A key for grouping banking information.
billingAddress	A key for grouping billing address information.
businessAddress	A key for grouping business address information.
businessInfo	A key for grouping business information.
card	A key for grouping payment card information.
data	A key for grouping different types of API information.
internal	A key for grouping internal information.
links	A key for grouping link information.
meta	A key for grouping result, limit, and error information.
originalSplit	A key for grouping information about the original split payment.
originalTransaction	A key for grouping information about the original transaction.
personalAddress	A key for grouping personal address information.
personalInfo	A key for grouping personal information.
split	A key for grouping split payment information.
token	A key for grouping token information.
transaction	A key for grouping payment transaction information.
verificationData	A key for grouping server and timestamp information.

Table 3 – Objects

Part Two – Integration

This section provides some useful system configuration information for the Digitzs API.

- Authentication. Describes the authentication process and whitelisting.
- Postman. Explains how to install and configure Postman.
- Merchants. Describes the account creation and data validation process.
- Payments. Provides some payment-related data tables.

Authentication

Authentication Setup

The three-step authentication setup process is described in this section.

Download Digitzs Credentials

Once your Partner Application is approved, the named technical contact for your account receives a secure link (via email) to download Digitzs credentials. The link is good for a single download and expires in 7 days. Your credentials are private and should be stored securely. You receive two credentials:

- **apiKey.** The API Key is required in the header of each call your application makes to Digitzs API.
- **appId.** The App ID is used to create an App Key for the API environment that you will access. Do not use the App ID in your application, only use it to create an App Key.

The partner creates App Tokens and the application uses the apiKey and the /auth/token POST endpoint to create App Tokens (appToken). An App Token is a reusable JSON Web Token (JWT) that is good for one hour. When the token expires, the application creates a new one.

Note: See Authentication for more information.

Create App Key

Use your App ID (appId) and the /auth/key POST endpoint to manually create an App Key (appKey) for each environment used by your application. The App Key you create is private and should be stored securely in a way that is accessible to your application, typically as an environment variable. The App Key does not expire and you may create a new key at any time. Creating a new key renders the old key unusable.

An example command line request is shown below:

```
# how to create an App Key in the sandbox environment using cURL
# be sure to replace the placeholders with your own apiKey and appId
curl -X POST --header "Content-Type: application/json"\
--header "Accept: application/json"\
--header "x-api-key: *<YOUR_API_KEY>*"\
-d "{
  \"data\": {
    \"type\": \"auth\",
    \"attributes\": {
      \"appId\": \"*<YOUR_APP_ID>*\"
    }
  }
}" "https://beta.digitzsapi.com/sandbox/auth/key"
```

Figure 1 – Command Line Request

The returned appKey can be used by your application. An example response is shown below:

```
{
  "links": {
    "self": "https://beta.digitzsapi.com/sandbox/auth/key"
  },
  "data": {
    "type": "auth",
    "id": "sOQnjKCDRk8pEhMo3Au0qyd4WpfqLxm9ZmmZnDy7",
    "attributes": {
      "appKey": "HxYWZHA7BImW24NZKyXsvIPy9xYZioxxSqmPeR8nd1rFdvcx0GFHp8x7DHeoMek7"
```

```
    }
  }
}
```

Figure 2 – Example Response

Create App Tokens

Your application uses your self-created App Key and the /auth/token POST endpoint to create App Tokens (appToken). An App Token is a reusable JSON Web Token (JWT) that is good for one hour. When this token expires, your application should create a new one. It is common for this step to be managed by cron. A PHP example is shown below.

```
// how to create an App Token using PHP and cURL
// assumes API Key, App Key, and URL are stored as environment variables

$digitzsAppKey = $_ENV['DIGITZS_APP_KEY'];
$digitzsApiKey = $_ENV['DIGITZS_API_KEY'];
$digitzsUrl = $_ENV['DIGITZS_URL'];

$data = [
    "data" => [
        "type" => "auth",
        "attributes" => [
            "appKey" => $digitzsAppKey
        ]
    ]
];

$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $digitzsUrl."auth/token");
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($data));
curl_setopt($ch, CURLOPT_SSLVERSION, CURL_SSLVERSION_TLSv1_2);
curl_setopt($ch, CURLOPT_HTTPHEADER, [
    "Content-Type: application/json",
    "Accept: application/json",
    "x-api-key: ".$digitzsApiKey
]);

$result = curl_exec($ch);
curl_close($ch);

$result = json_decode($result);

if ($result->data->attributes->appToken)
{
    $theToken = $result->data->attributes->appToken;
}
else
{
    die('Failed to get a token.');
}
```

Figure 3 – PHP Example

You may test an App Token by using the /auth/token GET endpoint.

Authentication Headers

When making calls to Digitzs API, you authenticate by adding the Authorization and x-api-key headers to your calls. These headers should be added when calling all endpoints other than /auth. Implementation varies by language.

```
"Content-Type": "application/json",
"Accept": "application/json",
"Authorization": "Bearer *<YOUR_APP_TOKEN>*",
"x-api-key": "*<YOUR_API_KEY>*"
```

Figure 4 – Authentication Headers

Note: See HTTP Protocol for more information.

Whitelisting

A security function that restricts system access to designated IP addresses. A whitelisting example is shown below:

CIDR	Description
127.0.0.1/32	A development server.
192.168.1.1/32	A development server.
192.168.0.1/32	A recurring payment server.
192.168.0.2/32	The main server.

Table 4 – Whitelisting Example

Postman

This section describes how to download, configure, and use Postman to run the Digitzs API.

1. Go to https://www.getpostman.com/.
2. Download and install **Postman**.
3. Launch **Postman**.
4. Click the **Import** button to display the Import dialog.

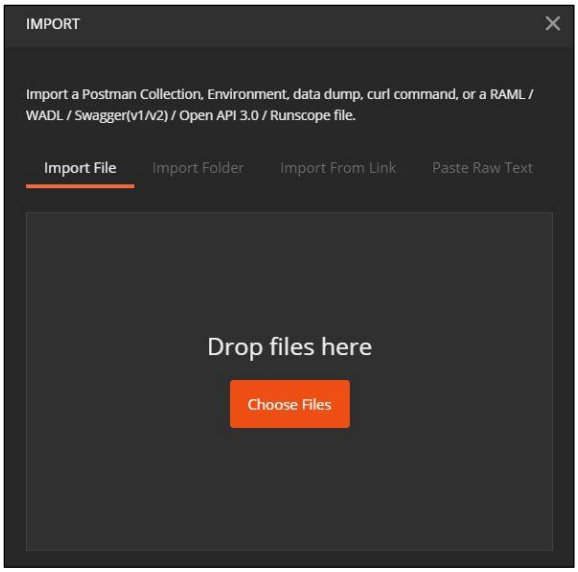

Figure 5 – Import Dialog

5 Click the **Choose Files** button to display the Open dialog.

6 Select the **Digitzs_postman_collection.json** file.

7 Click the **Open** button to import the file into Postman.

8 Click the **Manage Environments** button to display the Manage Environments dialog.

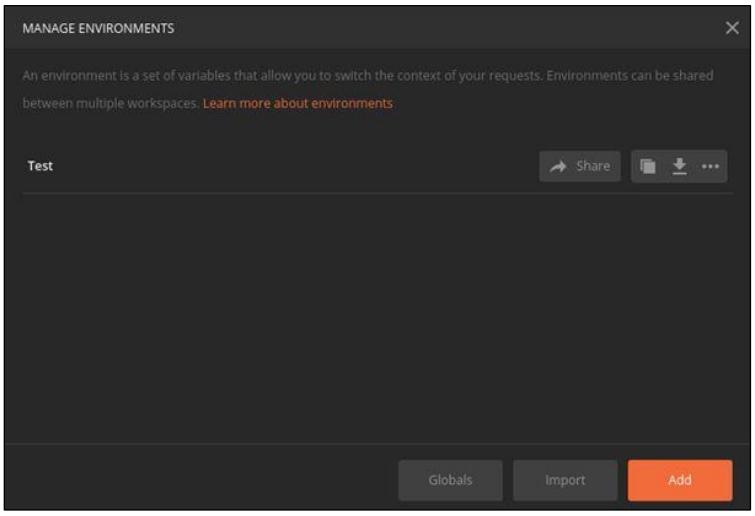

Figure 6 – Manage Environments Dialog

9 Click the **Add** button to display the Add Environment area.

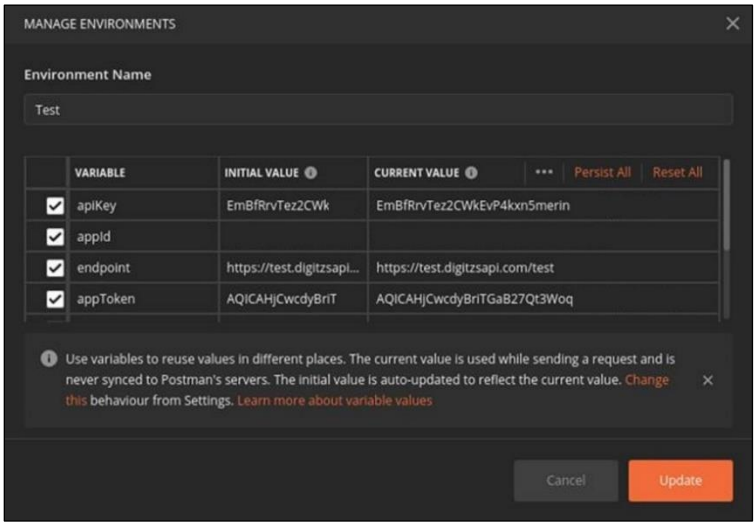

Figure 7 – Environment Variable Values

10 Enter the **apiKey**, **appId**, and **endpoint** environment variables provided by Digitzs during onboarding.

11 Click the **Update** button.

12 Run **POST Auth/Key** to get an appKey. Add it to the Environment Variables.

13 Run **Auth/Token** to get an appToken that is valid for one hour. Add it to the Environment Variables.

14 Run **POST Test API-Card Payments** to get an "id". Add it to the Environment Variables as cardpaymentId.

15 Run **POST Test API-Card Payment Refund**. You should receive a Response as confirmation that the Environment Variables are configured accurately.

Merchants

Digitzs merchant accounts are created in real-time and merchants can accept payments immediately.

Signup Flow

Digitzs has a few fraud prevention and verification requirements to consider when designing your signup flow.

Fraud Prevention

Since merchant accounts are activated instantly, you must implement sensible controls to prevent fraudulent merchant signups. The signup form should not be available on the open internet allowing anyone to sign up freely. Some methods of securing are to place the form behind your application's log in, require a PIN, or send invitations. Before being certified to use our production API, you will be asked to provide screen shots and a brief description of your signup flow to ensure these requirements are met.

Email Verification

You are required to verify the email address of each merchant and capture an IP address and timestamp of when the verifications took place.

Merchant Agreement

The Digitzs Merchant Agreement must be displayed in an iframe as part of your signup flow. The merchant must click to agree and type their name to sign the Agreement. You must capture a timestamp and IP address to confirm these actions.

```
<iframe src="https://ma.digitzs.com"></iframe>
```

Figure 8 – Merchant Agreement

Account Creation

Example Request Body

All fields shown below are required for business accounts. Personal accounts do not need to provide businessInfo.

```
{
  "data": {
    "type": "merchants",
    "attributes": {
      "accountType": "business",
      "accountName": "Susan's Widgets",
      "personalInfo": {
        "firstName": "Susan",
        "lastName": "Smith",
        "email": "susan@example.com",
        "dayPhone": "3105551212",
        "eveningPhone": "3105551212",
        "birthDate": "11-23-1980",
        "socialSecurity": "123456789"
      },
      "personalAddress": {
        "line1": "21 Jump Street",
        "line2": "Suite 007",
        "city": "Los Angeles",
        "state": "CA",
        "zip": "90210",
        "country": "USA"
      },
```

```
      "businessInfo": {
        "businessName": "Susan's Widgets",
        "ein": "123456789"
      },
      "businessAddress": {
        "line1": "21 Jump Street",
        "line2": "Suite 007",
        "city": "Los Angeles",
        "state": "CA",
        "zip": "90210",
        "country": "USA"
      },
      "bankInfo": {
        "bankName": "Bank of Abundance",
        "accountOwnership": "business",
        "accountType": "checking",
        "accountName": "Susan's Widgets",
        "accountNumber": "1234567",
        "routingNumber": "123456789"
      },
      "verificationData": {
        "ipAddress": "192.168.0.1",
        "emailVerified": true,
        "emailVerifiedTimestamp": "2016-01-31T11:16:07.234Z",
        "signature": "Susan Smith",
        "signatureTimestamp": "2016-01-31T11:16:07.234Z",
        "termsAccepted": true
      }
    }
  }
}
```


Figure 9 – Example Request

Data Validation Details

Variable Name	Example	Type	Description
data	–	Object	A key for grouping different types of API data.
type	"merchants"	Enum	The type of transaction.
attributes	–	Object	A key for grouping identification and payment information.
accountType	"business"	Enum	The type of bank account (Business or Personal).
accountName	"MegaCorp"	String	The name of the business.
personalInfo	–	Object	A key for grouping personal information.
firstName	“Susan”	String	The merchant’s first name.
lastName	“Smith”	String	The merchant’s last name.
email	"susan@example.com"	String	The merchant's email address.
dayPhone	"3105551212"	String	The merchant’s daytime contact number. Do not include any dashes or spaces in the phone number.
eveningPhone	"2135551212"	String	The merchant’s evening contact number. Do not include any dashes or spaces in the phone number.

Variable Name	Example	Type	Description
birthDate	"01-25-1980"	String	The merchant's date of birth.
socialSecurity	"123456789"	String	The merchant's social security number.
personalAddress	–	Object	A key for grouping personal address information.
line1	"21 Jump St."	String	The personal street address.
line2	"3rd floor"	String	A second line for street address information.
city	"Los Angeles"	String	The city where the individual is located.
state	"CA"	String	The state where the individual is located.
zip	"90061"	String	The postal code where the individual is located.
country	"USA"	Enum	The three-character country code.
businessInfo	–	Object	A key for grouping business information.
businessName	"MegaCorp"	String	The legal name of the business as registered.
ein	"123456789"	String	The Employer Identification Number (EIN).
businessAddress	–	Object	A key for grouping business address information.
line1	"21 Jump St."	String	The business street address.
line2	"3rd floor"	String	A second street address line.
city	"Los Angeles"	String	The city where the business is located.
state	"CA"	String	The state where the business is located.
zip	"90061"	String	The postal code where the business is located.
country	"USA"	Enum	The three-character country code.
bankInfo	–	Object	A key for grouping banking information.
bankName	"Wells Fargo"	String	The name of the financial institution.
accountOwnership	"business"	Enum	The valid values are Personal or Business.
accountType	"checking"	Enum	The type of bank account (Checking or Savings).
accountName	"MegaCorp"	String	The name of the individual or business associated with the account.
accountNumber	"1234567"	String	The financial institution account number. It is required if boarding with the primary bank on file.
routingNumber	"026009593"	String	The bank routing number.
verificationData	–	Object	A key for grouping server and timestamp information.
ipAddress	"192.168.0.1"	String	The Internet Protocol (IP) address used for the transaction.
emailVerified	"true"	Boolean	A true or false value.
emailVerifiedTimestamp	"2016-01-31T11:16:07.234Z"	String	The date and time the email was sent.
signature	"Susan Smith"	String	The first and last name of the customer.
signatureTimestamp	"2016-01-31T12:16:07.234Z"	String	The date and time the transaction was authorized.
termsAccepted	"true"	Boolean	A true or false value.

Table 5 – Data Validation Details

Signup Exceptions

When a merchant signs up for a Digitzs account, we verify their SSN and home address in real-time through Experian. If you have proper form validation in place and make sure customers know they need to submit their SSN and home address, 95-98% of merchants will be instantly approved. In 2-5% of cases, manual verification is required.

The most common triggers for manual verification are:

- Merchant submitted an EIN in the SSN field.
- Merchant provided a business address or PO Box instead of home address.
- Merchant has a new last name due to marriage.
- Merchant ordered a fraud lock at Experian (usually due to identity theft).

In cases of failed identity verification, Digitzs API returns the default error message below. If you are handling merchant support internally or using a white label support address, you may customize the message before displaying it to users.

"Unable to verify SSN and address. Contact merchantsupport@digitzs.com to request manual verification. Signed exception form, driver's license, and voided check are required. Business entities may submit SS-4 and business license if SSN is not available."

Digitzs will help facilitate manual identity verification for your merchants who request it. In most cases, the merchant is required to submit a signed exception form, a copy of their driver's license, and a voided check showing their current address. Companies that do not want to provide an SSN have the option of submitting business documents for verification.

Payments

Simulated Processing

Reserved Card Numbers

Number	Card Brand	Simulated Response
4747474747474747	Visa	Success
4111111111111111	Visa	Success
5454545454545454	MasterCard	Success
371449635398431	American Express	Success
6011000000000012	Discover	Success
355355335533553	JCB	Success
4404040404040404	Visa	Invalid credit card number as reported by the issuing bank.
4909090909090909	Visa	Credit card issuer's bank timed out. Try the transaction again.
4828282828282828	Visa	Card limit exceeded.
4616161616161616	Visa	Insufficient funds.
4535353535353535	Visa	Invalid credit card number. Credit card networks cannot locate the card's issuing bank.

Table 6 – Reserved Card Numbers

Reserved Amounts

Amount	Result Status	Response Code	Simulated Response
110	58	14	Invalid credit card number as reported by issuing bank.
111	58	19	Credit card issuer's bank timed out, please attempt this transaction again.
112	58	17	Card limit exceeded.
113	58	51	Insufficient funds. Will also produce a decline for Multi-Currency transactions.
114	58	58	Card issuing bank will not allow this type of transaction. Some cards such as gasoline cards or HSA can only be used for specific types of transactions.
115	58	15	Invalid credit card number. Credit card networks cannot locate this card's issuing bank.

Table 7 – Reserved Amounts

Reserved CVV Codes

Card Brand	CVV	Result Status	Response Code	Simulated Response
Visa	999	0	0	Success
MasterCard	999	0	0	Success
Discover	999	0	0	Success
American Express	3714	0	0	Success
All	All Others	82	82	CVV data is not correct.

Table 8 – Reserved CVV Codes

Part Three – APIs / Endpoints

This section describes the following endpoints available with the Digitzs API.

- Authentication. Verifies user information for access to system resources.
- Authorization. Sets permissions for access to system resources.
- Card Payment Request. Processes internet credit and debit card payments.
- Card Refund Payment. Processes internet credit and debit card refunds.
- Card Split Payment. Processes the initial credit card transaction and the shared transaction fees.
- Card Split Refund. Processes the credit card transaction and shared transaction fee refunds.
- Create a New Merchant. Sets up a new merchant for a specific partner.
- Create a New Customer. Sets up an account for a brand-new customer.
- Create Token for Credit Card. Used to designate a credit card for automatic monthly payments.
- Create Token for ACH. Used to designate a checking or savings account for automatic monthly payments.
- GET Customer by ID. Used to view specific customer information.
- GET Merchant by ID. Used to view specific merchant information.
- ACH Payment Request. Provides authorization to debit directly from the customer's checking or savings account.
- Token Payment. Used to perform payment transactions without sensitive data going through the Digitzs system.
- Token Split Payment. Used to process split payment transactions without sensitive data going through Digitzs system.
- GET Merchant List. Used to view a list of merchant information.
- GET Authentication. Used to view authentication information.
- GET Authorization. Used to view authorization information.
- GET Customer List. Used to view a list of customer information.
- GET Token by ID. Used to view information about a specific token.
- GET Token List. Used to view a list of token information.
- GET Payment by ID. Used to view specific payment information.
- GET Payment List. Used to obtain a list of payment information.

Authentication

Use this endpoint to authenticate user information for access to system resources.

Path

```
https://auth/key/
```

Figure 10 – Endpoint Path

Note: This is the path for the Production environment. The Test environment path is /test/auth/key.

Request

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "type": "auth", "attributes": { "appId": "fade5009-d8ad-4ea3-bfe7-ebdaa3f6bf9e" } } }
type	This value must be "auth".	
attributes	A key for grouping identification and payment information.	
appId	Supplied by Digitzs. It is used to create the appKey.	

Figure 11 – Authentication Request

Response

Parameter	Description	Example
data	A key for grouping different types of API data.	{ "data": { "type": "auth", "id": "MYaYlZvW3T85fDKkJTOBh2jL4rUfTRuqal9MZnBU", "attributes": { "appKey": "l9lRManqYoEaxVxw327DxMUxauug5FGxPamWDyZSGlrQksjlyaexREtXTeRg4vOB" } } }
type	This value must be "auth".	
id	Supplied by Digitzs. It is used to access the API.	
attributes	A key for grouping identification and payment information.	
appKey	Supplied by Digitzs. It is used to create the auth/token.	

Figure 12 – Authentication Response

Authorization

Use this endpoint to set permissions for access to system resources.

Path

https://auth/token/

Figure 13 – Endpoint Path

Request

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "type": "auth", "attributes": { "appKey": " l9IRManqYoEaxVxw327DxMUxauug5FGxPamWDyZ SGIrQksjIyaexREtXTeRg4vOB" } } }
type	This value must be "auth".	
attributes	A key for grouping identification and payment information.	
appKey	Supplied by Digitzs. It is used to create the auth/token.	

Figure 14 – Authorization Request

Response

Parameter	Description	Example
data	A key for grouping different types of API data.	{ "data": { "type": "auth", "id": "MYaYlZvW3T85fDKkJTOBh2jL4rUfTRuqal9MZnBU", "attributes": { "appToken": " AQICAHgX5ymjRYCIjktlP/NzH5wGSPM5Im6Mn7Y lEYOEIRjstQH1sO8cJylmFK1TIemRXZFlAAABszCCAa8GCSqGS Ib3DQEHBqCCAaAwggGcAgEAMIIBlQYJKoZIhvcNAQcBMB4GC WCGSAFlAwQBLjARBAzRwIAaY5XG6/UBZDcCARCAggFm5Eqx QMo9VaM0lNVoW0omCWQaKTcAPoVE3GmISFIJKV33Ix/Wq8lP LLFAYvRGBzMmZxQdviPV69JxH/diCKr3l7uuuRPRgXv2p2oc+k0 xKsFyxGZvqP0sT7WHljMRr4dDk33lcVqreqAJYIplrTSbWQ0WYsS 6ngFeKLQTgRrG5jo7xIjGEloRg/x05S/GiwFo9vzdWzrIJoeyoqyDgi oqZLge+suzquxu0aYMwTcOPSQSsZWDQjJKH/y68ZaI+YmxJyma EaLdcPjgVxiRhsC/FSW1I7uPBkfvQIKQ6WsNmqIeNkKMVVwxaN ScuM8beSzC5I0J4tTPY0e9ATmtZRIFJ+12deeRomjs3LodSJTOm D5gJe22y7LRfZxTu/4qLje+ih644WED4RUm1dh7qU5DCa6JHmxP OOSAD6yJmhKRLdzyCIyM+OTITsLtB5HNJogqMLdPEaHxvXUH pJcdMU1qHxi30ftpjQ==” } } }

Figure 15 – Authorization Response

Card Payment

Used to process a payment by a card (debit or credit) when customer wants to pay for a purchase on a merchant's site.

Path

https://payments/

Figure 16 – Endpoint Path

Request

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "type": "payments", "attributes": { "paymentType": "card", "merchantId": " corsourc-susanswidg-32429758-2302978-153897104, "card": { "holder": "Jane Doe", "number": "5215245673758689986", "expiry": "0919", "code": "999" }, "transaction": { "amount": "999", "currency": "USD", "invoice": "123456" }, "billingAddress": { "line1": "21 Jump Street ", "line2": "Suite 007", "city": "Los Angeles", "state": "CA", "zip": "90210", "country": "USA" } } } }
type	The type of transaction. This value must be “payments”.	
attributes	A key for grouping identification and payment information.	
paymentType	The type of payment transaction. This value must be "card". See Payment Types for more information.	
merchantId	The merchant account number used for the transaction.	
card	A key for grouping payment card information.	
holder	The first and last name on the account for a payment method.	
number	The unique number assigned to an account. For a credit card, this is the card number.	
expiry	The expiration date of the payment method. This is a four-digit MMYY numeric value.	
code	The three-character CVV2 code on back of the card. See CVV2 Response Codes for more information.	
transaction	A key for grouping payment transaction information.	
amount	The transaction amount in dollars and cents.	
currency	The three-character code. See Currency Codes for more information.	
invoice	The invoice number for the transaction.	
billingAddress	A key for grouping billing address information.	
line1	The card holder’s street address.	
line2	A second address line.	
city	The card holder’s home city.	
state	The card holder’s home state.	
zip	The card holder’s zip or postal code. See State and Province Codes for more information.	
country	The card holder’s home country. See Country Codes for more information.	

Figure 17 – Card Payment Request

Response

Parameter	Description	Example

data	A key for grouping different types of API data.	{
		"data": {
		"type": "payments",
		"id": "corsourc-susanswidg- 32429758-2302978-153897104-67-1540174725",
		"attributes": {
		"paymentType": "card",
		"transaction": {
		"code": "0",
		"message": "Success",
		"amount": "999",
		"invoice": "123456",
		"currency": "USD",
		"authCode": "A11111",
		"avsResult": "T",
		"gross": "999",
		"net": "940",
		"grossMinusNet": "59",
		"fee": "30",
		"rate": "2.90"
		}
		}
		}
		}

type	The type of transaction. This value must be “payments”.	
id	The payment transaction identification code.	
attributes	A key for grouping identification and payment information.	
paymentType	The type of payment transaction. This value must be "card".	
transaction	A key for grouping payment transaction information.	
code	The banking entity response code. This should be “0”. See Transaction Response Codes for more information.	
message	The status of the transaction. This should be “Success”.	
amount	The transaction amount in dollars and cents.	
invoice	The invoice number associated with the transaction.	
currency	The currency used for the transaction.	
authCode	The transaction authorization code.	
avsResult	The transaction result code. See AVS Response Codes for more information.	
gross	The gross transaction amount.	
net	The net transaction amount.	
grossMinusNet	The difference between the gross and net amounts.	
fee	The transaction fee.	
rate	The percentage or fixed amount of the transaction fee.	

Figure 18 – Card Payment Response

Card Refund Payment

Used to process card refund payment transaction from merchant's account to customer's account.

Path

https://payments/

Figure 19 – Endpoint Path

Request

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "type": "payments", "attributes": { "paymentType": "cardRefund", "merchantId": "corsourc-hongle- 32413168-2280562-153363721", "transaction": { "amount": "113", "currency": "USD", "invoice": "123456" }, "originalTransaction": { "id": "corsourc-hongle- 32413168-2280562-153363721" } } } }
type	The type of transaction. This value must be “payments”.	
attributes	A key for grouping identification and payment information.	
paymentType	The type of payment. Must equal “cardRefund”. See Payment Types for more information.	
merchantId	Thant account number used for the transaction.	
transaction	A key for grouping payment transaction information.	
amount	The transaction amount in dollars and cents.	
currency	The currency used for the transaction. See Currency Codes for more information.	
invoice	The invoice number associated with the transaction.	
originalTransaction	A key for grouping information about the original transaction.	
id	The card refund payment.	

Figure 20 – Card Refund Payment Request

Response

Parameter	Description	Example
data	A key for grouping different types of API data.	{ "data": { "type": "payments", "id": "corsourc-hongle- 32413168-2280562-153363722-556-154259474", "attributes": { "paymentType": "cardRefund", "transaction": { "code": "0", "message": "Success", "amount": "113", "invoice": "123456",
type	The type of transaction. This value must be “payments”.	
id	The payment transaction identification code.	
attributes	A key for grouping identification and payment information.	
paymentType	The type of payment. Must equal “cardRefund”.	
transaction	A key for grouping payment transaction information.	

Parameter	Description	Example

code	The banking entity response code. This should be “0”. See Transaction Response Codes for more information.	<pre>"currency": "USD"
		}
		}
		}
		}</pre>

message	The status of the transaction. This should be “Success”.	
amount	The transaction amount in dollars and cents.	
invoice	The invoice number associated with the transaction.	
currency	The currency used for the transaction.	

Figure 21 – Card Refund Payment Response

Card Split Payment

Used to process the initial credit card transaction and any shared transaction fees.

Path

https://payments/

Figure 22 – Endpoint Path

Request

Parameter	Description
data	A key for grouping different types of API data. See Objects for more information.
type	The type of transaction. This value must be “payments”.
attributes	A key for grouping identification and payment information.
paymentType	The type of payment transaction. This value must be “cardSplit”. See Payment Types for more information.
merchantId	The merchant account number used for the transaction.
card	A key for grouping payment card information.
holder	The first and last name on the account for a payment method.
number	The unique number assigned to an account. For a credit card, this is the card number.
expiry	The expiration date of the payment method. This is a four-digit (MMYY) numeric value.
code	The three-character CVV2 code on back of the card. See CVV2 Response Codes for more information.
split	A key for grouping split payment information.
merchantId	A merchant identifier. This is the account where the split goes after the transaction settles.
amount	The split payment amount.
transaction	A key for grouping payment transaction information.
amount	The transaction amount in dollars and cents.
currency	The currency used for the transaction. See Currency Codes for more information.
invoice	The invoice number associated with the transaction.

```
{
  "data": {
    "type": "payments",
    "attributes": {
      "paymentType": "cardSplit",
      "merchantId": "corsourc-hongle- 32413168-2280562-
      153363721",
      "card": {
        "holder": "Jane Doe",
        "number": "4747474747474747",
        "expiry": "0219",
        "code": "999"
      },
      "split": {
        "merchantId": "corsourc-hongle- 32413168-2280562-
        153363721",
        "amount": "100"
      },
      "transaction": {
        "amount": "500",
        "currency": "USD",
        "invoice": "123456"
      }
    }
  }
}
```

Figure 23 – Card Split Payment Request

Response

Parameter	Description	Example
data	A key for grouping different types of API data.	{ "data": { "type": "payments", "id": "corsourc-hongle- 32413168-2280562-153363721-563-154260022", "attributes": { "paymentType": "cardSplit", "transaction": { "code": "0", "message": "Success", "amount": "500", "invoice": "123456", "currency": "USD", "authCode": "A11111", "avsResult": "T", }, "split": { "merchantId": "corsourc-hongle- 32413168-2280562-153363721", "amount": "100", "splitId": "corsourc-hongle- 32413168-2280562-153363721-565-154260022" } } } }
type	The type of transaction. This value must be “payments”.	
id	The payment transaction identification code.	
attributes	A key for grouping identification and payment information.	
paymentType	The type of payment transaction. This value must be “cardSplit”.	
transaction	A key for grouping payment transaction information.	
code	The banking entity response code. This should be “0”. See Transaction Response Codes for more information.	
message	The status of the transaction. This should be “Success”.	
amount	The transaction amount in dollars and cents.	
invoice	The invoice number associated with the transaction.	
currency	The currency used for the transaction.	
authCode	The transaction authorization code.	
avsResult	The transaction result code. See AVS Response Codes for more information.	
split	A key for grouping split payment information.	
merchantId	A merchant identifier. This is the account where the split goes after the transaction settles.	
amount	The split payment amount.	
splitId	A customer identifier generated by Digitzs.	

Figure 24 – Card Split Payment Response

Card Split Refund

Used to process the credit card transaction and shared transaction fee refunds.

Path

https://payments/

Figure 25 – Endpoint Path

Request

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "type": "payments", "attributes": { "paymentType": "cardSplitRefund", "merchantId": "corsourc-merchant2-32457485-2355941-1548838840", "split": { "merchantId": "corsourc-merchant3-32457486-2355943-1548838859", "amount": "001" }, "transaction": { "amount": "100", "currency": "USD", "invoice": "123455" }, "originalSplit": { "id": "corsourc-merchant2-32457485-2355941-1548838840-9-1548839516" } } } }
type	The type of transaction. This value must be “payments”.	
attributes	A key for grouping identification and payment information.	
paymentType	The type of payment transaction. This value must be “cardSplit”. See Payment Types for more information.	
merchantId	The merchant account number used for the transaction.	
split	A key for grouping split payment information.	
merchantId	A merchant identifier. This is the account where the split goes after the transaction settles.	
amount	The split amount.	
transaction	A key for grouping payment transaction information.	
amount	The transaction amount in dollars and cents.	
currency	The currency used for the transaction. See Currency Codes for more information.	
invoice	The invoice number associated with the transaction.	
originalSplit	A key for grouping information about the original split payment.	
id	The payment transaction identification code.	

Figure 26 – Card Split Refund Request

Response

Parameter	Description	Example

data	A key for grouping different types of API data.	{
		"data": {
		"type": "payments",
		"attributes": {
		"paymentType": "cardSplitRefund",
		"merchantId": "corsourc-merchant2-32457485-2355941-1548838840",
		"split": {
		"merchantId": "corsourc-merchant3-32457486-2355943-1548838859",
		"amount": "001"
		},
		"transaction": {
		"amount": "100",
		"currency": "USD",
		"invoice": "123455"
		},
		"originalSplit": {
		"id": "corsourc-merchant2-32457485-2355941-1548838840-9-1548839516"
		}
		}
		}
		}

type	The type of transaction. This value must be “payments”.	
attributes	A key for grouping identification and payment information.	
paymentType	The type of payment transaction. This value must be “cardSplit”.	
merchantId	A merchant identifier. This is the account where the split goes after the transaction settles.	
split	A key for grouping split payment information.	
merchantId	A merchant identifier. This is the account where the split goes after the transaction settles.	
amount	The split amount.	
transaction	A key for grouping payment transaction information.	
amount	The transaction amount in dollars and cents.	
currency	The currency used for the transaction. See Currency Codes for more information.	
invoice	The invoice number associated with the transaction.	
originalSplit	A key for grouping information about the original split payment.	
id	The payment transaction identification code.	

Figure 27 – Card Split Refund Response

Create a New Merchant

Used to create a new merchant for a specific partner.

Path

https://merchants/

Figure 28 – Endpoint Path

Request

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "type": "merchants", "attributes": { "accountType": "business", "accountName": "Susan's Widgets", "personalInfo": { "firstName": "Susan", "lastName": "Smith", "email": "susan1@example.com", "dayPhone": "3105551222", "eveningPhone": "3105551222", "birthDate": "11-23-1981", "socialSecurity": "123456789" }, "personalAddress": { "line1": "21 Jump Street", "line2": "Suite 007", "city": "Los Angeles", "state": "CA", "zip": "90210", "country": "USA" }, "businessInfo": { "businessName": "Susan's Widgets", "ein": "123456789" }, "businessAddress": { "line1": "21 Jump Street", "line2": "Suite 007", "city": "Los Angeles", "state": "CA", "zip": "90210" "country": "USA" }, "bankInfo": { "bankName": "Bank of Abundance", "accountOwnership": "business", "accountType": "checking", "accountName": "Susan's Widgets", "accountNumber": "1234567",
type	The type of transaction. This value must be "merchants".	
attributes	A key for grouping identification and payment information.	
accountType	The type of bank account (Checking or Savings).	
accountName	The name of the individual or business associated with the account.	
personalInfo	A key for grouping personal information.	
firstName	The merchant's first name.	
lastName	The merchant's last name.	
email	The merchant's email address.	
dayPhone	The merchant's daytime contact number. Do not include any dashes or spaces in the phone number.	
eveningPhone	The merchant's evening contact number. Do not include any dashes or spaces in the phone number.	
birthDate	The merchant's date of birth.	
socialSecurity	The merchant's social security number.	
personalAddress	A key for grouping personal address information.	
line1	The personal street address.	
line2	A second line for street address information.	
city	The city where the individual is located.	
state	The state where the individual is located. See State and Province Codes for more information.	
zip	The postal code where the individual is located.	
country	The three-character country code. See Country Codes for more information.	
businessInfo	A key for grouping business information.	

Parameter	Description	Example

businessName	The legal name of the business as registered.	"routingNumber": "026009593" },
		"verificationData": {
		"ipAddress": "192.168.0.1",
		"emailVerified": true,
		"emailVerifiedTimestamp": " 2019-01-31T11:16:07.234Z",
		"signature": "Susan Smith",
		"signatureTimestamp": " 2016-01-31T11:16:07.234Z",
		"termsAccepted": true
		}
		}
		}
		}

ein	The Employer Identification Number (EIN).	
businessAddress	A key for grouping business address information.	
line1	The business street address.	
line2	A second street address line.	
city	The city where the business is located.	
state	The state where the business is located.	
zip	The postal code where the business is located.	
country	The three-character country code.	
bankInfo	A key for grouping banking information.	
bankName	The name of the financial institution.	
accountOwnership	The valid values are Personal or Business.	
accountType	The type of bank account (Checking or Savings).	
accountName	The name of the individual or business associated with the account.	
accountNumber	The financial institution account number. It is required if boarding with the primary bank on file.	
routingNumber	The bank routing number. This is required if the payment method is checking or savings.	
verificationData	A key for grouping server and timestamp information.	
ipAddress	The Internet Protocol (IP) address used for the transaction.	
emailVerified	A true or false value.	
emailVerifiedTimestamp	The date and time the email was sent.	
signature	The first and last name of the customer.	
signatureTimestamp	The date and time the transaction was authorized.	
termsAccepted	A true or false value.	

Figure 29 – Create a New Merchant Request

Response

Parameter	Description	Example

data	A key for grouping different types of API data.	{
		"data": {
		"type": "merchants",
		"id": "corsourc-susanswidg-32437120-2311544-154112710"
		}
		}

type	The type of transaction. This value must be “merchants”.	
id	A merchant identifier. This is the account where the split goes after the transaction settles.	

Figure 30 – Create a New Merchant Response

Create a New Customer

Used to set up an account for a brand-new customer.

Path

```
https://customers/
```

Figure 31 – Endpoint Path

Request

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "type": "customers", "attributes": { "merchantId": "ccorsourc-susanswidg-32414161-2281597-153431435", "name": "John Smith", "externalId": "CustomerNumberVip" } } }
type	The type of transaction. This value must be “customers”.	
attributes	A key for grouping identification and payment information.	
merchantId	The merchant account number used for the transaction.	
name	The customer's name.	
externalId	Used to identify a customer.	

Figure 32 – Create a New Customer Request

Response

Parameter	Description	Example
data	A key for grouping different types of API data.	{ "data": { "type": "customers", "id": "ccorsourc-susanswidg-32414161-2281597-153431435-1045175642679504-154356815", "attributes": { "name": "John Smith", "externalId": "CustomerNumberVip" } } }
type	The type of transaction. This value must be “customers”.	
Id	A customer identifier generated by Digitzs.	
attributes	A key for grouping identification and payment information.	
name	The customer's name.	
externalId	Used to identify a customer.	

Figure 33 – Create a New Customer Response

Create Token for Credit Card

Used to designate a credit card for automatic monthly payments.

Path

https://tokens/

Figure 34 – Endpoint Path

Request

Parameter	Description
data	A key for grouping different types of API data. See Objects for more information.
type	The type of transaction. This value must be “customers”.
attributes	A key for grouping identification and payment information.
tokenType	The banking method. This value must equal “card”.
customerId	The account number used to process the transaction.
label	A brief description of the payment method. See Payment Methods for more information.
card	A key for grouping payment card information.
type	The method of payment. See Payment Methods for more information.
holder	The first and last name on the account for a payment method.
number	The unique number assigned to an account. For a credit card, this is the card number.
expiry	The expiration date of the payment method. This is a four-digit MMYY numeric value.
billingAddress	A key for grouping billing address information.
line1	The billing address for the customer payment method.
line2	A second row for billing address information.
city	The city associated with the billing address.
state	The state associated with the billing address. See State and Province Codes for more information.
zip	The postal code associated with the billing address.
country	The three-character country code. See Country Codes for more information.

```
{
  "data": {
    "type": "tokens",
    "attributes": {
      "tokenType": "card",
      "customerId": "corsourc-hongle- 32413168-2280562-
      153363721-7892132626826839-153369332",
      "label": "Visa",
      "card": {
        "type": "visa",
        "holder": "Jane Doe",
        "number": "5454545454545454",
        "expiry": "0219"
        ”protected": true
      },
      "billingAddress": {
        "line1": "21 Jump Street",
        "line2": "Suite 007",
        "city": "Los Angeles",
        "state": "CA",
        "zip": "90210",
        "country": "USA"
      }
    }
  }
}
```

Figure 35 – Create Token for Credit Card Request

Response

Parameter	Description	Example

data	A key for grouping different types of API data.	{
		"data": {
		"type": "tokens",
		"attributes": {
		"tokenType": "card",
		"customerId": "corsourc-hongle-32413168-2280562-153363721-7892132626826839-153369332",
		"label": "Visa",
		"bank": {
		"accountType": "checking",
		"accountName": "MegaCorp",
		"accountNumber": "1234567",
		"routingNumber": "026009593",
		"country": "USA"
		},
		"billingAddress": {
		"line1": "21 Jump Street",
		"line2": "Suite 007",
		"city": "Los Angeles",
		"state": "CA",
		"zip": "90210",
		"country": "USA"
		}
		}
		}
		}

type	The type of transaction. This value must be “customers”.	
attributes	A key for grouping identification and payment information.	
label	A brief description of the payment method.	
customerId	The Customer ID.	
label	A brief description of the payment method.	
bank	A key for grouping banking information.	
accountType	The type of bank account (Checking or Savings).	
accountName	The name of the individual or business associated with the account.	
accountNumber	The unique number assigned to the payment account.	
routingNumber	The bank routing number. This is required if the payment method is checking or savings.	
country	The three-character country code.	
billingAddress	A key for grouping billing address information.	
line1	The billing street address.	
line2	A second street address line.	
city	The city where the billing is located.	
state	The two-character billing address state code.	
zip	The postal code for the billing address.	
country	The three-character billing address country code.	

Figure 36 – Create Token for Credit Card Response

Create Token for ACH

Used to designate a bank account for automatic monthly payments.

Path

https://tokens/

Figure 37 – Endpoint Path

Request

Parameter	Description	Example

data	A key for grouping different types of API data. See Objects for more information.	{
		"data": {
		"type": "tokens",
		"attributes": {
		"tokenType": "ach",
		"customerId": "corsourc-hongle- 32413168-2280562-153363721-7892132626826839-153369332",
		"label": "Checking",
		"card": {
		"type": "visa",
		"holder": "Jane Doe",
		"number": "5454545454545454",
		"expiry": "0219"
		”protected": true
		},
		"billingAddress": {
		"line1": "21 Jump Street",
		"line2": "Suite 007",
		"city": "Los Angeles",
		"state": "CA",
		"zip": "90210",
		"country": "USA"
		}
		}
		}
		}

type	The type of transaction. This value must be “customers”.	
attributes	A key for grouping identification and payment information.	
tokenType	The banking method. This value must equal “ach”.	
customerId	The account number used to process the transaction.	
label	A brief description of the payment method. See Payment Methods for more information.	
protected	The valid values are: “true” or “false”. Payment methods set to “true” cannot be deleted by a customer.	
card	A key for grouping payment card information.	
type	The method of payment. See Payment Methods for more information.	
holder	The first and last name on the account for a payment method.	
number	The unique number assigned to an account. For a credit card, this is the card number.	
expiry	The expiration date of the payment method. This is a four-digit MMYY numeric value.	
billingAddress	A key for grouping billing address information.	
line1	The billing address for the customer payment method.	
line2	A second row for billing address information.	
city	The city associated with the billing address.	
state	The state associated with the billing address. See State and Province Codes for more information.	
zip	The postal code associated with the billing address.	
country	The three-character country code. See Country Codes for more information.	

Figure 38 – Create Token for ACH Request

Response

Parameter	Description	Example

data	A key for grouping different types of API data.	{
		"data": {
		"type": "tokens",
		"attributes": {
		"tokenType": "bank",
		"customerId": "corsourc-hongle-32413168-2280562-153363721-7892132626826839-153369332",
		"label": "Citi Visa",
		"bank": {
		"accountType": "checking",
		"accountName": "MegaCorp",
		"accountNumber": "1234567",
		"routingNumber": "026009593",
		"country": "USA"
		},
		"billingAddress": {
		"line1": "21 Jump Street",
		"line2": "Suite 007",
		"city": "Los Angeles",
		"state": "CA",
		"zip": "90210",
		"country": "USA"
		}
		}
		}
		}

type	The type of transaction. This value must be “customers”.	
attributes	A key for grouping identification and payment information.	
label	A brief description of the payment method.	
customerId	The Customer ID.	
label	A brief description of the payment method.	
bank	A key for grouping banking information.	
accountType	The type of bank account (Checking or Savings).	
accountName	The name of the individual or business associated with the account.	
accountNumber	The unique number assigned to the payment account.	
routingNumber	The bank routing number. This is required if the payment method is checking or savings.	
country	The three-character country code.	
billingAddress	A key for grouping billing address information.	
line1	The billing street address.	
line2	A second street address line.	
city	The city where the billing is located.	
state	The two-character billing address state code.	
zip	The postal code for the billing address.	
country	The three-character billing address country code.	

Figure 39 – Create Token for ACH Response

GET Customer by ID

Used to view specific customer information.

Path

```
https://customers/{id}
```

Figure 40 – Endpoint Path

Request

```
https://beta.digitzsapi.com/sandbox/customers/corsourc-dinhhuyen-32452812-2344440-1546498220-7613503575427935-1548318668
```

Figure 41 – GET Customer by ID Request

Note: There are no request elements for this endpoint.

Response

Parameter	Description
data	A key for grouping different types of API data. See Objects for more information.
attributes	A key for grouping identification and payment information.
name	The name of the customer.
externalId	A value used to identify customer.
internal	A key for grouping results.
protectpayId	A ProtectPay value used to identify customers.
merchantId	The merchant account number used for the transaction.
id	A customer identifier generated by Digitzs.
type	The type of transaction. The value must be “customers”.


```
{
  "data": {
    "attributes": {
      "name": "dinh huyen",
      "externalId": "CustomerNumberVip",
      "internal": {
        "protectpayId": "7613503575427935"
    },
      "merchantId": "corsourc-dinhhuyen-32452812-2344440",
      "created": 1548318668
    },
      "id": "corsourc-dinhhuyen-32452812-2344440-1546498220",
      "type": "customers"
  }
}
```


Figure 42 – GET Customer by ID Response

GET Merchant by ID

Used to view specific merchant information.

Path

https://merchants/{id}

Figure 43 – Endpoint Path

Request

https://beta.digitzsapi.com/sandbox/merchants/corsourc-susanswidg-32436665-2310890-154103821

Figure 44 – GET Merchant by ID Request

Note: There are no request elements for this endpoint.

Response

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "partnerId": "corsourc-susanswidg- 32436665-2310890-154103821", "attributes": { "personalInfo": { "eveningPhone": "3105551222", "firstName": "Susan1", "lastName": "Smith1", "dayPhone": "3105551222", "socialSecurity": "xxxx6789", "birthDate": "xxxx1981", "email": "susan1@example.com" }, "bankInfo": { "routingNumber": "xxxx9593", "bankName": "Bank of Abundance", "accountNumber": "xxxx4567", "accountName": "Susan's Widgets", "accountOwnership": "business", "accountType": "checking" }, "verificationData": { "ipAddress": "192.168.0.1", "emailVerified": true, "signature": "Susan Smith", "emailVerifiedTimestamp": "2016-01-31T11:16:07.234Z", "termsAccepted": true, "signatureTimestamp": "2016-01-31T11:16:07.234Z" }, "accountName": "Susan's Widgets111", "created": "2018-11-01T02:10:14.415Z", "accountType": "business", "partnerId": "corsourc-152910534",
partnerId	A merchant identifier generated by Digitzs.	
attributes	A key for grouping identification and payment information.	
personalInfo	A key for grouping personal information.	
eveningPhone	The merchant's evening phone number. Do not include any dashes or spaces in the phone number.	
firstName	The merchant's first name.	
lastName	The merchant's last name.	
dayPhone	The merchant's daytime phone number. Do not include any dashes or spaces in the phone number.	
socialSecurity	The merchant's social security number.	
birthDate	The merchant's date of birth.	
email	The customer's email address.	
bankInfo	A key for grouping banking information.	
routingNumber	The bank routing number. This is required if the payments from checking or savings accounts. See Payment Methods for more information.	
bankName	The name of the financial institution.	
accountNumber	The unique number assigned to the payment account.	
accountName	The name of the individual or business associated with the account.	
accountOwnership	The valid values are Personal or Business.	
accountType	The type of bank account (Checking or Savings).	

Parameter	Description	Example
verificationData	A key for grouping server and timestamp information.	
ipAddress	The Internet Protocol (IP) address used for the transaction.	
emailVerified	A true or false value.	
signature	The first and last name of the customer.	
emailVerifiedTimestamp	The date and time the email was sent.	
termsAccepted	A true or false value.	
signatureTimestamp	The date and time that the transaction was authorized.	
accountName	The name of the individual or business associated with the account.	
created	The date that the payment was created.	
accountType	The account type (Personal or Business).	
partnerId	A unique identifier assigned to the partner.	
businessAddress	A key for grouping business address information.	
line1	The business street address.	
line2	A second business street address line.	
city	The city where the business is located.	
state	The two-character business address state code. See State and Province Codes for more information.	
zip	The postal code for the business address.	
country	The three-character business address country code. See Country Codes for more information.	
personalAddress	A key for grouping personal address information.	
line1	The personal street address.	
line2	A second personal street address line.	
city	The city where the individual is located.	
state	The two-character personal address state code.	
zip	The postal code for the personal address.	
country	The three-character personal address country code.	
businessInfo	A key for grouping business information.	
businessName	The legal name of the business as registered.	
ein	The Employer Identification Number (EIN).	
type	The type of transaction. The value must be “merchants”.	


```
    "businessAddress": {
      "line1": "21 Jump Street",
      "line2": "Suite 007",
      "city": "Los Angeles",
      "state": "CA",
      "zip": "90210",
      "country": "USA"
    },
    "personalAddress": {
      "line1": "21 Jump Street",
      "line2": "Suite 007",
      "city": "Los Angeles",
      "state": "CA",
      "zip": "90210",
      "country": "USA"
    },
    "businessInfo": {
      "businessName": "Susan's Widgets",
      "ein": "xxxx6789"
    }
  },
  "type": "merchants"
 }
}
```


Figure 45 – GET Merchant by ID Response

ACH Payment

Used to give authorization to debit directly from the customer's checking or savings account.

Path

```
https://payments/
```

Figure 46 – Endpoint Path

Request

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "type": "payments", "attributes": { "paymentType": "ACH", "merchantId": "corsourc-hongle-32413168-2280562-1533637", "StandardEntryClassCode": "WEB", "bank": { "accountType": "checking", "accountName": "MegaCorp", "accountNumber": "1234567", "routingNumber": "026009593" }, "transaction": { "amount": "4000", "currency": "USD", "invoice": "123456" } } } }
type	The type of business. This value must be “Payments”.	
attributes	A key for grouping identification and payment information.	
paymentType	The type of payment. This value must be “ACH”. See Payment Types for more information.	
merchantId	The merchant account number used for the transaction.	
standardEntryClassCode	The valid values are WEB, TEL, CCD, and PPD.	
bank	A key for grouping results.	
accountType	The type of bank account (Checking or Savings). See Payment Methods for more information.	
accountName	The name of the individual or business associated with the account.	
accountNumber	The unique number assigned to the payment account.	
routingNumber	The bank routing number. This is required if the payment method is checking or savings.	
transaction	A key for grouping payment transaction information.	
amount	The transaction amount in dollars and cents.	
currency	The three-character currency code. See Currency Codes for more information.	
invoice	The invoice number associated with the transaction.	

Figure 47 – ACH Payment Request

Response

Parameter	Description	Example

data	A key for grouping different types of API data.	{
		"data": {
		"type": "payments",
		"id": "corsourc-hongle-32413168-2280562-153363721-495-154209955",
		"attributes": {
		"paymentType": "ACH",
		"transaction": {
		"code": "0",
		"message": "Success",
		"amount": "4000",
		"currency": "USD",
		"invoice": "123456",
		}
		}
		}

type	The type of transaction.	
id	A value created by Digitzs.	
attributes	A key for grouping identification and payment information.	
paymentType	The type of payment transaction. This value must be “ACH”.	
transaction	A key for grouping payment transaction information.	
code	The banking entity response code. This should be “0”. See Transaction Response Codes for more information.	
message	The status of the transaction. This should be “Success”.	
amount	The transaction amount in dollars and cents.	
currency	The currency used for the transaction.	
invoice	The invoice number associated with the transaction.	

Figure 48 – ACH Payment Response

Token Payment

Used to perform payment transactions without sensitive data going through the Digitzs system.

Path

https://payments

Figure 49 – Endpoint Path

Request

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "type": "payments", "attributes": { "paymentType": "token", "merchantId": "corsourc-susanswidg-32442731-2321569-154225521", "token": { "customerId": "corsourc-susanswidg-32442731-2321569-154225521-1589455281052624-154225566", "tokenId": "c6cf7b25-d46e-4acc-8024-db38f0d6a6e5-154225579" }, "transaction": { "amount": "363541", "currency": "USD", "invoice": "PAYMENT-ACH-HONG-7" } } } }
type	The type of transaction. The value must be "payments".	
attributes	A key for grouping identification and payment information.	
paymentType	The type of payment transaction. The value must be “token”. See Payment Types for more information.	
merchantId	The Merchant ID.	
token	A key for grouping token information.	
customerId	The Customer ID.	
tokenId	The Token ID.	
transaction	A key for grouping payment transaction information.	
amount	The transaction amount in dollars and cents.	
currency	The currency used for the transaction. See Currency Codes for more information.	
invoice	The invoice number associated with the transaction.	

Figure 50 – Token Payment Request

Response

Parameter	Description	Example

data	A key for grouping different types of API data.	{
		"data": {
		"type": "payments",
		"id": "corsourc-susanswidg-32442731-2321569-154225521-21-154235859",
		"attributes": {
		"paymentType": "token",
		"transaction": {
		"code": "0",
		"message": "Success",
		"amount": "300",
		"invoice": "PAYMENT-ACH-HONG-7",
		"currency": "USD",
		"authCode": null,
		"codeResult": "NotPresent",
		"gross": null,
		"net": null,
		"grossMinusNet": null,
		"fee": null,
		"rate": null
		}
		}
		}
		}

type	The type of transaction. The value must be "payments".	
id	A customer identifier generated by Digitzs.	
attributes	A key for grouping identification and payment information.	
paymentType	The type of payment transaction. The value must be “token”.	
transaction	A key for grouping payment transaction information.	
code	The banking entity response code. This should be “0”. See Transaction Response Codes for more information.	
message	The status of the transaction. This should be “Success”.	
amount	The transaction amount in dollars and cents.	
invoice	The invoice number associated with the transaction.	
currency	The currency used for the transaction.	
authCode	The transaction authorization code.	
codeResult	The transaction result code.	
gross	The gross transaction amount.	
net	The net transaction amount.	
grossMinusNet	The difference between the gross and net amounts.	
fee	The transaction fee.	
rate	The percentage or fixed amount of the transaction fee.	

Figure 51 – Token Payment Response

Token Split Payment

Used to process split payment transactions without sensitive data going through Digitzs system.

Path

https://payments/

Figure 52 – Endpoint Path

Request

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "type": "payments", "attributes": { "paymentType": "tokenSplit", "merchantId": "corsourc-hongle-32413168-2280562-153363721", "token": { "customerId": "corsourc-hongle-32413168-2280562-153363721-7892132626826839-153369332", "tokenId": "acfff292-3f1d-46b3-bd7d-851a14feef6b-154285692" }, "split": { "merchantId": "corsourc-hongle-32413168-2280562-153363721", "amount": "10000" }, "transaction": { "amount": "15000", "currency": "USD", "invoice": "123456" } } } }
type	The type of transaction. This value must equal “payments”.	
attributes	A key for grouping identification and payment information.	
paymentType	The type of payment. Must equal “tokenSplit”. See Payment Types for more information.	
merchantId	The merchant account number used for the transaction.	
token	A key for grouping token information.	
customerId	The customer account number used for the transaction.	
tokenId	The token ID.	
split	A key for grouping split payment information.	
merchantId	The account where the split goes after the transaction settles.	
amount	The split amount.	
transaction	A key for grouping payment transaction information.	
amount	The transaction amount in dollars and cents.	
currency	The currency used for the transaction. See Currency Codes for more information.	
invoice	The invoice number associated with the transaction.	

Figure 53 – Token Split Payment Request

Response

Parameter	Description	Example
data	A key for grouping different types of API data.	{ "data": { "type": "payments", "id": "corsourc-hongle-32413168-2280562-153363721-621-154285847", "attributes": { "paymentType": "tokenSplit", "transaction": { "amount": "15000", "invoice": "123456", "currency": "USD", "authCode": "A11111", "codeResult": "NotPresent" }, "split": { "merchantId": "corsourc-hongle-32413168-2280562-153363721", "amount": "10000", "splitId": "corsourc-hongle-32413168-2280562-153363721-623-154285847" } } } }
type	The transaction type. This value must equal “payments”.	
id	A customer identifier generated by Digitzs.	
attributes	A key for grouping identification and payment information.	
paymentType	The type of payment. This value must equal “tokenSplit”.	
transaction	A key for grouping payment transaction information.	
amount	The transaction amount in dollars and cents.	
invoice	The invoice number associated with the transaction.	
currency	The currency used for the transaction.	
authCode	The transaction authorization code.	
codeResult	The transaction result code.	
split	A key for grouping split payment information.	
merchantId	The account where the split goes after the transaction settles.	
amount	The split payment amount.	
splitId	A customer identifier generated by Digitzs.	

Figure 54 – Token Split Payment Response

GET Merchant List

Used to view a list of merchant information.

Path

```
https://merchants?limit=25
```

Figure 55 – Endpoint Path

Request

```
https://beta.digitzsapi.com/sandbox/merchants?limit=25
```

Figure 56 – GET Merchant List Request

Note: There are no request elements for this endpoint.

Response

Parameter	Description
meta	A key for grouping different types of API data. See Objects for more information.
items	The number of items in the database. It is equal to or smaller than limit's value.
limit	The maximum number of results to return.
more	A true/false value that indicates whether “items” or “limit” allows more results.
data	A key for grouping results.
id	A value created by Digitzs.
attributes	A key for grouping identification and payment information.
accountName	The name of the individual or business associated with the account.
accountType	The account type (Personal or Business).
created	The date and time that the payment was created.
type	The transaction type. Must equal "merchants".


```
{
  "meta": {
    "items": 25,
    "limit": 25,
    "more": true
  },
  "data": [
    {
      "id": "corsourc-susanswidg-32436665-2310890-154103821",
      "attributes": {
        "accountName": "Susan's Widgets",
        "accountType": "business",
        "created": "2018-11-01T02:10:14.415Z"
      },
      "type": "merchants"
.....
    {
      "id": "corsourc-susanswidg-32414181-2281631-153431784",
      "attributes": {
        "accountName": "Susan's Widgets1",
        "accountType": "business",
        "created": "2018-08-15T07:24:01.832Z"
      },
      "type": "merchants"
    }
  }
}
```


Figure 57 – GET Merchant List Response

GET Authentication

Used to view authentication information.

Path

```
https://auth/key/
```

Figure 58 – Endpoint Path

Request

```
https://beta.digitzsapi.com/sandbox/auth?appid=fade5009-d8ad-4ea3-bfe7-ebdaa3f6bf9f
```

Figure 59 – GET Authentication Request

Note: There are no request elements for this endpoint.

Response

Parameter	Description	Example

data	A key for grouping different types of API data. See Objects for more information.	{
		"data": {
		"type": "auth",
		"id": "fade5009-d8ad-4ea3-bfe7-ebdaa3f6bf9f",
		"attributes": {
		"appid": "fade5009-d8ad-4ea3-bfe7-ebdaa3f6bf9f"
		}
		}
		}

type	This value must be "auth".	
id	A value created by Digitzs.	
attributes	A key for grouping identification and payment information.	
appId	Supplied by Digitzs. It is used to create the appKey.	

Figure 60 – GET Authentication Response

GET Authorization

Used to view authorization information.

Path

```
https://auth/token/
```

Figure 61 – Endpoint Path

Request

```
https://beta.digitzsapi.com/sandbox/auth?appToken=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXJ0bmVySWQiOiJjb3Jzb3VyYy0xNTI5MTA1MzQiLCJwYXJ0bmVyUHJlZml4IjoiY29yc291cmMiLCJwcm9wYXlUaWVyIjoiRGlnaXR6cyIsInByb3BheU1jYyI6IjczNzIiLCJpYXQiOjE1NDQ3NzE4NzMsImV4cCI6MTU0NDc3NTQ3M30.sOfIG0tC813aOvVeESUxXXtlS6JDpV_SVS9CxAJCQo8
```

Figure 62 – GET Authorization Request

Note: There are no request elements for this endpoint.

Response

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "type": "auth", "id": "cRe4WcuXKszn3bJabjnjeVUXBKPxyYiNVQFNIQuo", "attributes": { "appToken": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJwYXJ0bmVySWQiOiJjb3Jzb3VyYy0xNTI5MTA1MzQiLCJwYXJ0bmVyUHJlZml4IjoiY29yc291cmMiLCJwcm9wYXlUaWVyIjoiRGlnaXR6cyIsInByb3BheU1jYyI6IjczNzIiLCJpYXQiOjE1NDQ3NzE4NzMsImV4cCI6MTU0NDc3NTQ3M30.sOfIG0tC813aOvVeESUxXXtlS6JDpV_SVS9CxAJCQo8" } } }
type	This value must be "auth".	
id	This is the apiKey.	
attributes	A key for grouping identification and payment information.	
appToken	The appToken is generated by appKey and apiKey.	

Table 9 – GET Authorization Response

GET Customer List

Used to view a list of customer information.

Path

https://customers?merchantId={id}&limit=25

Figure 63 – Endpoint Path

Request

.https://beta.digitzsapi.com/sandbox/customers?id=corsourc-hongle-32413168-2280562-153363721&limit=25

Figure 64 – GET Customer List Request

Note: There are no request elements for this endpoint.

Response

Parameter	Description
meta	A key for grouping results.
items	The number of items in the database. It is equal to or smaller than limit's value.
limit	The maximum number of results to return.
more	A true/false value that indicates whether “items” or “limit” allows more results.
data	A key for grouping different types of API data. See Objects for more information.
id	A value created by Digitzs.
attributes	A key for grouping identification and payment information.
created	The date and time that the payment was created.
externalId	The unique identifier assigned to customer.
merchantId	The merchant account number used for the transaction.
name	The name of the customer.
type	This value must be “customer”.


```
{
  "meta": {
    "items": 5,
    "limit": 25,
    "more": false
  },
  "data": [
    {
    "id": " 2413168-2280562-153363721-8574933098692461-
    153430651 ",
    "attributes": {
      "created": "2018-08-15T04:15:17.368Z",
      "externalId": "1234577",
      "merchantId": "corsourc-hongle- 32413168-2280562-
      153363721 ",
      "name": "ABC Customer"
    },
    "type": "customers"
  },
  {
    "id": "corsourc-hongle- 32413168-2280562-153363721-
    5128807619772100-153424326",
    "attributes": {
      "created": "2018-08-07T10:20:35.778Z",
      "externalId": "123456",
      "merchantId": "corsourc-hongle- 32413168-2280562-
      153363721 ",
      "name": "Pizza Customer"
    },
    "type": "customers"
    }
  }
}
```


Figure 65 – GET Customer List Response

GET Token by ID

Used to view information about a specific token.

Path

```
https://payments/{id}
```

Figure 66 – Endpoint Path

Request

```
https://beta.digitzsapi.com/dev/tokens/5331087e-0936-4b4a-b97b-a0712f43be88
```

Figure 67 – GET Token by Id Request

Note: There are no request elements for this endpoint.

Response – Token Payment

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "id": "ff209f7e-a233-4926-a4ae-68141c467bf9-154380837", "attributes": { "customerId": "corsourc-hongle-32413168-2280562-153363721-7892132626826839-153369332", "label": "MasterCard", "billingAddress": { "line1": "21 Jump Street", "line2": "Suite 007", "city": "Los Angeles", "state": "CA", "zip": "90210", "country": "USA" }, "tokenType": "card", "card": { "number": "xxxx5454", "holder": "Jane Doe", "expiry": "0219", "type": "visa" }, "created": "2018-12-03T03:39:31.159Z" }, "type": "tokens" } }
id	A value created by Digitzs.	
attributes	A key for grouping identification and payment information.	
customerId	The account number used to process the transaction.	
label	A brief description of the payment method.	
billingAddress	A key for grouping billing address information.	
line1	The billing address for the payment method.	
line2	A second row for billing address information.	
city	The city associated with the billing address.	
state	The state associated with the billing address. See State and Province Codes for more information.	
zip	The postal code associated with the billing address.	
country	The three-character country code. See Country Codes for more information.	
tokenType	The banking method. This value must equal “card”.	
card	A key for grouping payment card information.	
number	The unique number assigned to an account. For a credit card, this is the card number.	
holder	The first and last name on the account for a payment method.	
expiry	The expiration date of the payment method. This is four-digit MMYY numeric value.	
type (card)	The type of payment card used. See Payment Met0hods for more information.	

Parameter	Description	Example
created	The date and time that the payment was created.	
type	The type of token being used.	

Figure 68 – Token Payment Response

Response – Token ACH

Parameter	Description	Example

data	A key for grouping different types of API data.	{
		"data": {
		"id": "0b203a66-8bfe-4723-81a2-849e14e22369-154388865",
		"attributes": {
		"customerId": "corsourc-hongle- 32413168-2280562-153363721-7892132626826839-153369332 ",
		"bank": {
		"routingNumber": "xxxx6789",
		"country": "USA",
		"accountNumber": "xxxx4567",
		"accountName": "MegaCorp",
		"accountType": "checking"
		},
		"label": "MasterCard",
		"tokenType": "bank",
		"created": "2018-12-04T01:57:38.357Z"
		},
		"type": "tokens"
		}
		}

id	A value created by Digitzs.	
attributes	A key for grouping identification and payment information.	
customerId	The account number used to process the transaction.	
bank	A key for grouping banking information.	
routingNumber	The bank routing number. This is required if the payment method is checking.	
country	The three-character country code.	
accountNumber	The unique number assigned to the payment account.	
accountName	The name of the individual or business associated with the account.	
accountType	The type of bank account (Checking or Savings).	
label	A brief description of the payment method.	
tokenType	The banking method. This value must equal “card”.	
created	The date and time that the payment was created.	
type	The type of token being used.	

Figure 69 – Token ACH Response

GET Token List

Used to view a list of token information.

Path

https://tokens?limit=25&customerId={id}

Figure 70 – Endpoint Path

Request

https://beta.digitzsapi.com/sandbox/tokens?id=corsourc-hongle-32413168-2280562-153363721-7892132626826839-153369332&limit=25

Figure 71 – GET Token List Request

Note: There are no request elements for this endpoint.

Response

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": [{ "id": " 55ea1d2c-d9eb-4465-8623-c51046b642e5-154451427", "attributes": { "customerId": "corsourc-hongle- 32413168-2280562-153363721-7892132626826839-153369332 ", "label": "MasterCard", "billingAddress": { "line1": "21 Jump Street", "line2": "Suite 007", "city": "Los Angeles", "state": "CA", "zip": "90210", "country": "USA" }, "tokenType": "card", "card": { "number": "xxxx5454", "holder": "Jane Doe", "expiry": "0219", "type": "visa" }, "created": "2018-12-11T07:44:34.899Z" }, "type": "tokens" }] }
id	A value created by Digitzs.	
attributes	A key for grouping identification and payment information.	
customerId	The account number used to process the transaction.	
label	A brief description of the payment method.	
billingAddress	A key for grouping billing address information.	
line1	The billing address for the customer payment method.	
line2	A second row for billing address information.	
city	The city associated with the billing address.	
state	The state associated with the billing address. See State and Province Codes for more information.	
zip	The postal code for the billing address.	
country	The three-character country code. See Country Codes for more information.	
tokenType	The banking method. This value must equal “card” or “bank”.	
card	A key for grouping payment card information.	
number	The unique number assigned to an account. For a credit card, this is the card number.	
holder	The first and last name on the account for a payment method.	
expiry	The expiration date of the payment method. This is four-digit MMYY numeric value.	

Parameter	Description	Example
type (card)	The type of payment card used. See Payment Methods for more information.	
created	The date and time that the payment was created.	
type	The type of token being used.	

Figure 72 – GET Token List Response

GET Payment by ID

Used to view specific payment information.

Path

https://payments/{id}

Figure 73 – Endpoint Path

Request

https://beta.digitzsapi.com/sandbox/payments/corsourc-susanswidg-32448949-2335714-154466644-1-154466655

Figure 74 – GET Payment by ID Request

Note: There are no request elements for this endpoint.

Response

Parameter	Description	Example
data	A key for grouping different types of API data. See Objects for more information.	{ "data": { "id": "corsourc-susanswidg- 32448949-2335714-154466644-1-154466655", "attributes": { "merchantId": "corsourc-susanswidg- 32448949-2335714-154466644", "created": "2018-12-13T02:02:34.417Z", "billingAddress": { "line1": "21 Jump Street", "line2": "Suite 007", "city": "Los Angeles", "state": "CA", "zip": "90210", "country": "USA" }, "card": { "number": "xxxx5454", "holder": "Jane Doe", "expiry": "0219", "code": "xxx" }, "transaction": { "amount": "100", "code": "0", "authCode": "A11111", "avsResult": "T", "gross": "100", "fee": "30", "message": "Success", "grossMinusNet": "33", "rate": "2.90", "currency": "USD", "codeResult": "M", "invoice": "123456",
id	A value created by Digitzs.	
attributes	A key for grouping identification and payment information.	
merchantId	The merchant account number used for the transaction.	
created	The date and time that the payment was created.	
billingAddress	A key for grouping billing address information.	
line1	The billing address for the customer payment method.	
line2	A second row for billing address information.	
city	The city associated with the billing address.	
state	The state associated with the billing address. See State and Province Codes for more information.	
zip	The postal code associated with the billing address.	
country	The three-character country code. See Country Codes for more information.	
card	A key for grouping payment card information.	
number	The unique number assigned to an account. For a credit card, this is the card number. See Payment Methods for more information.	
holder	The first and last name on the account for a payment method.	
expiry	The expiration date of the payment method. This is four-digit MMYY numeric value.	
code	The three-character CVV2 code on back of the card. See CVV2 Response Codes for more information.	

Parameter	Description	Example
transaction	A key for grouping payment transaction information.	"net": "67" }, "paymentType": "card" }, "type": "payments" } }
amount	The transaction amount in dollars and cents.	
code	The banking entity response code. This should be “0”. See Transaction Response Codes for more information.	
authCode	The transaction authorization code.	
avsResult	The transaction result code. See AVS Response Codes for more information.	
gross	The gross transaction amount.	
fee	The transaction fee.	
message	The status of the transaction. This should be “Success”.	
grossMinusNet	The difference between the gross and net amounts.	
rate	The percentage or fixed amount of the transaction fee.	
currency	The currency used for the transaction. See Currency Codes for more information.	
codeResult	The transaction result code.	
invoice	The invoice number associated with the transaction.	
net	The net transaction amount.	
paymentType	The type of payment. See Payment Types for more information.	
type	The transaction type. This value must equal “payments”.	

Figure 75 – GET Payment by ID Response

GET Payment List

Used to obtain a list of payment information.

Path

```
/dev/payments?limit=25&start=YYYYMMDD&end=YYYYMMDD&id={id}
```

Figure 76 – Endpoint Path

Request

```
https://beta.digitzsapi.com/sandbox/payments?id=corsourc-hongle-32413168-2280562-153363721&limit=25&start=20181212
&end=20181212
```

Figure 77 – GET Payment List Request

Note: There are no request elements for this endpoint.

Response

Parameter	Description
meta	A key for grouping different types of API data. See Objects for more information.
items	The number of items in the database. It is equal to or smaller than limit's value.
limit	The maximum number of results that should be displayed.
more	A true/false value that indicates whether “items” or “limit” allows more results.
data	A key for grouping different types of API information.
id	A customer identifier generated by Digitzs.
attributes	A key for grouping identification and payment information.
created	The date and time that the payment was created.
merchantId	The account number used for the transaction.
paymentType	The type of payment transaction. See Payment Types for more information.
transaction	A key for grouping payment transaction information.
amount	The transaction amount in dollars and cents.
authCode	The transaction authorization code.
gross	The gross transaction amount.
rate	The percentage or fixed amount of the transaction fee.
fee	The transaction fee.
codeResult	The transaction code result.
invoice	The invoice number associated with the transaction.

Example


```
{
  "meta": {
    "items": 25,
    "limit": 25,
    "more": true
  },
  "data": [
    {
    "id": "corsourc-susanswidg-32448949-2335714-154466644-7-
    154466818",
    "attributes": {
      "created": "2018-12-13T02:29:41.173Z",
      "merchantId": "corsourc-susanswidg-32448949-2335714-
      154466644",
      "paymentType": "token",
      "transaction": {
        "amount": "100",
        "authCode": null,
        "gross": null,
        "rate": null,
        "fee": null,
        "currency": "USD",
        "codeResult": "NotPresent",
        "invoice": "123456",
        "net": null,
        "grossMinusNet": null
      }
    },
    "type": "payments"
    }
  ]
}
```

Parameter	Description	Example
net	The net transaction amount.	
grossMinusNet	The difference between the gross and net amounts.	
type	The transaction type. Must equal "payments".	

Figure 78 – GET Payment List Response

Part Four – Codes

This section includes the following Digitzs API-related data tables.

- AVS Response Codes. A list of the Address Verification System (AVS) codes used with credit card payment transactions.
- Country Codes. A list of country codes for various countries around the world.
- Currency Codes. A list of the currency codes for various countries around the world.
- CVV2 Response Codes. A list of the Card Verification Value (CVV) codes used with credit card payment transactions.
- Error Codes. A list of the error codes included in an API response for credit card payment transactions.
- Payment Methods. A list of the payment methods that can be used to complete transactions.
- Payment Types. A list of payment types available with the Digitzs API.
- State and Province Codes. A list of codes for U.S. states and Canadian provinces.
- Status Response Codes. A list of the status response codes used with credit card payment transactions.
- Transaction Response Codes. A list of the transaction response codes for credit card payments.

AVS Response Codes

Code	Type	Description
A	Domestic	The street address matches. The 5-digit and 9-digit postal code do not match.
B	International	The address matches but the postal code could not be verified.
C	International	The street address and postal code do not match.
D	Domestic	There is an exact address match.
E	Domestic	The AVS data is invalid. AVS is not allowed for this card type.
F	American Express	The name and/or postal code do not match.
G	International	The non-U.S. issuing bank does not support AVS.
H	American Express	The name and/or the full AVS do not match.
I	International	The address cannot be verified.
J	American Express	The name and/or full AVS do not match.
K	American Express	The name matches. The full AVS does not match.
L	American Express	The name and postal code match.
M	International	There is an exact address match.
N	Domestic	The zip code and street address do not match.
O	American Express	The name and address match. The postal code does not match.
P	International	The zip code matches.
Q	American Express	There is an exact address match.
R	Domestic	The issuer system is unavailable.
S	Domestic	The service is not supported.
T	American Express	The name does not match. The street address does match.
U	Domestic	The verification process is unavailable.
V	American Express	There is an exact address match.
W	Domestic	The street address does not match. The nine-digit postal code does match.
X	Domestic	The street address and nine-digit postal code match.
Y	Domestic	The street address and five-digit postal code match.
Z	Domestic	The street address does not match. The five-digit postal code does match.
0	Domestic	There was no data provided to perform the AVS check.

Table 10 – AVS Response Codes

Country Codes

Code	Country
ARG	Argentina
AUS	Australia
BRA	Brazil
CAN	Canada
CHE	Switzerland
CHL	Chile
CHN	China
COL	Colombia
DEU	Germany
EGY	Egypt
FRA	France
GBR	Great Britain
IDN	Indonesia
ISR	Israel
IND	India
JPN	Japan
KOR	South Korea
MEX	Mexico
MYS	Malaysia
NOR	Norway
NZL	New Zealand
PAK	Pakistan
PAN	Panama
PER	Peru
PHL	Philippines
RUS	Russia
SAU	Saudi Arabia
SWE	Sweden
SGG	Singapore
THA	Thailand
TUR	Turkey
UKR	Ukraine
USA	United States of America
VEN	Venezuela
VNM	Vietnam
ZAF	South Africa

Table 11 – Country Codes

Currency Codes

Code	Currency
ARS	Argentine Peso
AUD	Australian Dollar
BRL	Brazilian Real
CAD	Canadian Dollar
CHF	Swiss Franc
CLP	Chilean Peso
CNY	Chinese Yuan Renminbi
COP	Colombian Peso
EGP	Egypt Pound
EUR	Euro
GBP	British Pound Sterling
IDR	Indonesian Rupiah
ILS	Israeli New Shekel
INR	Indian Rupee
JPY	Japanese Yen
KRW	South Korean Won
MXN	Mexican Peso
MYR	Malaysian Ringgit
NOK	Norwegian Krone
NZD	New Zealand Dollar
PAB	Panamanian Balboa
PEN	Peruvian Sol
PHP	Philippines Peso
PKR	Pakistani Rupee
RUB	Russian Ruble
SAR	Saudi Arabia Riyal
SEK	Swedish krona
SGD	Singapore Dollar
THB	Thai Baht
TRY	Turkish Lira
UAH	Ukrainian Hryvnia
USD	United States Dollar
VES	Venezuelan Bolívar Soberano
VND	Vietnamese Dong
ZAR	South African Rand

Table 12 – Currency Codes

CVV2 Response Codes

Code	Description
M	The CVV2 codes match.
N	The CVV2 codes do not match.
P	The payment was not processed.
S	There is no CVV2 code on the credit card.
U	The card issuer is not certified and/or has not provided the required encryption keys.

Table 13 – CVV2 Response Codes

Error Codes

Code	Title	Description
104	Transaction Not Refunded	The transaction could not be refunded due to current transaction state.
200	OK	The standard response for successful requests.
400	Bad Request	The requested item does not exist.
401	Unauthorized	An authorization header is required.
402	Payment Required	Reserved for future use in connection with digital payments.
403	Forbidden	The account is not authorized to access the requested resource.
404	Not Found	The requested resource cannot be found.
408	Request Timeout	The server timed out waiting for the request.
409	Conflict	Three is an editing conflict between two or more updates.
429	Too Many Requests	The user has sent too many requests in a given amount of time.
500	Internal Server Error	The server encountered an unexpected internal error.
501	Not Implemented	The server does not recognize the request method or cannot fulfill the request.
502	Bad Gateway	The server was acting as a gateway or proxy and received an invalid response.
503	Service Unavailable	The server is currently unavailable.
504	Gateway Timeout	A timely response was not received from server.

Table 14 – Error Codes

Payment Methods

Method	Description
American Express	A bank and financial services company that issues consumer, small business, and corporate cards.
Apple Pay	A mobile payment and digital wallet service that processes payments in person, in iOS apps, and on the web.
Cartes Bancaires	A French national interbank network with over 46,000 ATMs and over one million Electronic Funds Transfer at Point of Sale (EFTPOS) acceptance points. **Note:** This payment method is only used when connecting to the Payvision gateway.
Checking	A deposit account held at a bank or other financial institution. It is available to the account owner on demand and is used for frequent and immediate access by the account owner or to others as the account owner may direct.
Diners Club	A charge card company owned by Discover Financial Services.
Discover	A financial services company and credit card issuer that processes transactions through the Discover Network payment network.
JCB	A Japanese financial services company and credit card issuer. JCB is in a strategic alliance with Discover Network merchants in the United States, UnionPay merchants in China, American Express merchants in Canada, and RuPay merchants in India.
Maestro	A multi-national debit card service owned by Mastercard. **Note:** This payment method is only used when connecting to the Payvision gateway.
Mastercard	A financial services company that processes payments between merchants and the card issuing banks or credit unions of the purchasers who use Mastercard-brand debit, credit, and prepaid cards to make purchases.
ProPay	A merchant account provider that shares attributes of both a traditional credit card processor and services like PayPal and Stripe.
Savings	A deposit account held at a retail bank that pays interest. These accounts let customers set aside a portion of their liquid assets while earning a monetary return.
SEPA	The Single Euro Payments Area (SEPA) is a European Union payment-integration initiative applicable to bank transfers denominated in Euros. **Note:** This payment method is only used when connecting to the Payvision gateway.
Visa	A financial services company that processing electronic funds transfers throughout the world, most commonly through Visa-branded debit, credit, and prepaid cards to make purchases.

Payment Types

Type	Description
ACH	Authorizes debits directly from the customer's checking or savings account.
card	Processes a payment by a card when customer wants to pay for a purchase on a merchant's site.
cardRefund	Processes the initial credit card transaction and any shared transaction fees.
cardSplit	Processes the initial credit card transaction and any shared transaction fees.
cardSplitRefund	Processes the initial credit card transaction and any shared transaction fees.
tokenPayment	Processes payment transactions without sensitive data going through the Digitzs system.
tokenSplit	Processes split payment transactions without sensitive data going through Digitzs system.

Table 15 – Payment Types

State and Province Codes

Code	State	Code	State
AK	Alaska	OK	Oklahoma
AL	Alabama	OR	Oregon
AR	Arkansas	PA	Pennsylvania
AZ	Arizona	RI	Rhode Island
CA	California	SC	South Carolina
CO	Colorado	SD	South Dakota
CT	Connecticut	TN	Tennessee
DC	District of Columbia	TX	Texas
DE	Delaware	UT	Utah
FL	Florida	VA	Virginia
GA	Georgia	VT	Vermont
HI	Hawaii	WA	Washington
ID	Idaho	WV	West Virginia
IA	Iowa	WI	Wisconsin
IL	Illinois	WY	Wyoming
IN	Indiana	**Code**	**Territory**
KS	Kansas	AS	American Samoa
KY	Kentucky	FM	Federated States of Micronesia
LA	Louisiana	MH	Marshall Islands
MA	Massachusetts	MP	Northern Mariana Islands
MD	Maryland	PR	Puerto Rico
ME	Maine	PW	Palau
MI	Michigan	**Code**	**Province**
MN	Minnesota	AB	Alberta
MO	Missouri	BC	British Columbia
MT	Montana	MB	Manitoba
MS	Mississippi	NB	New Brunswick
NC	North Carolina	NL	Newfoundland and Labrador
ND	North Dakota	NS	Nova Scotia
NE	Nebraska	NT	Northwest Territories
NJ	New Jersey	NU	Nunavut
NH	New Hampshire	ON	Ontario
NM	New Mexico	PE	Prince Edward Island
NV	Nevada	QC	Quebec
NY	New York	SK	Saskatchewan
OH	Ohio	YT	Yukon

Table 16 – State and Province Codes

Status Response Codes

Code	Title	Description
0	Success	The payment was successful.
20	Invalid Username	The value username is already in use within the system.
21	Invalid Transaction Type	The requested transaction method is not permitted for the account.
22	Invalid Currency Code	The currency code value is not valid.
23	Invalid Account Type	The account type is not valid.
24	Invalid Source Email	The email address is not valid.
25	Invalid First Name	The first name value is not valid.
26	Invalid Middle Initial	The middle initial value is not valid.
27	Invalid Last Name	The last name value is not valid.
28	Invalid Billing Address	The billing address value is not valid.
29	Invalid Apartment Number	The apartment number value is not valid.
30	Invalid City	The city value is not valid.
31	Invalid State	The state value is not valid.
32	Invalid Zip Code	The zip code value is not valid.
38	Invalid Day Phone	The daytime contact phone number is not valid.
39	Invalid Evening Phone	The evening contact phone number is not valid.
40	Invalid Social Security Number	The social security number is not valid.
41	Invalid Date of Birth	The date of birth is not valid.
44	Invalid Amount	The transaction/payment amount is not valid.
45	Invalid Invoice Number	The invoice number is not valid.
46	Invalid Bank Routing Number	The routing number is not valid.
47	Invalid Account Number	The account number is not valid.
48	Invalid Credit Card Number	The credit card number is not valid.
49	Invalid Expiration Date	The credit card expiration date is not valid.
50	Invalid CVV2	The three-digit verification code is not valid.
51	Invalid Transaction Number	The transaction number is not unavailable.
52	Invalid Split Number	The split account number is invalid.
58	Declined	The payment was declined.
63	Insufficient Funds in Account	There are not enough funds available to process the payment.
64	Over Credit Card Limit	There is not enough credit available to process the payment.
65	Miscellaneous Error	There was a general system error.
67	Unauthorized Service	The account is not configured to perform the requested action.
68	Account Not Affiliated	The action requires that the account share the same affiliation as the requesting credential.

Table 17 – Status Response Codes

Transaction Response Codes

Code	Title	Description
0	Success	Approved. The transaction was completed successfully.
01	Call Issuer	Declined. The transaction was blocked by the credit card company. The card holder should call the number on the back of the card to find out why.
04	Pick Up Card	Declined. There is a problem with the card. The merchant should keep the card and call the number on the back of the card to determine the issue.
05	Do Not Honor	Declined. There is a problem with the customer's account or the card company doesn't recognize the billing address on file. Try adding the cardholder's correct billing address information and run the card again.
06	Error	Declined. The customer requested that a recurring payment be stopped.
07	Fraud	Declined. The card has been marked for fraud. The merchant should keep the card and call the number on the back of the card to determine the issue.
10	Partial Approval	Declined. Digitzs does not support partial approvals.
12	Invalid Transaction	Declined. Verify that the card information is input correctly and run it again.
13	Card Amount Invalid	Declined. The transaction amount is incorrect.
14	Card Number Invalid	Declined. The card number is incorrect. Verify the card number and re-enter.
15	No Such Issuer	Declined. The transaction could not be routed.
17	Over the Limit	Declined. The credit card is over the allowed limit.
19	Reenter Transaction	Declined. The transaction should be resubmitted.
41	Lost Card	Declined. The card has been reported as lost. The merchant should keep the card and call the number on the back of the card to determine the issue.
43	Stolen Card	Declined. Card has been reported as stolen. The merchant should call the number on the back of the card to determine the issue.
51	Insufficient Funds	Declined. There are not enough available funds on the card to complete the transaction.
52	No Checking Account	Declined. The checking account number does not exist.
53	No Savings Account	Declined. The savings account number does not exist.
54	Expired Card	Declined. The card has expired.
55	Incorrect PIN	Declined. The PIN is incorrect. Verify the PIN and run the card again.
57	Service Not Allowed	Declined. The card-issuing bank does not allow this type of purchase.
59	Suspected Fraud	Declined. There has been fraudulent activity reported for the card. The merchant should call the number on the back of the card to determine the issue.
61	Exceeds Limit	Declined. The card cannot be used until the withdrawal limit has been increased.
75	PIN Tries Exceeded	Declined. The card is locked due to too many incorrect PIN attempts.
93	Cannot Complete Transaction	Declined. The transaction cannot be completed due to a legal issue.

Table 18 – Transaction Response Codes

FAQs

What is Digitzs?

Digitzs is a painless, quick, and profitable way to bake payments into your merchant software platform.

Is Digitzs a Processor?

No. Think of us more as a gateway connected to Visa, Mastercard, and American Express. We facilitate merchant accounts in white label from within your site, process transactions, split payments, pay commissions and provide reporting to you. Your merchants won't know that we exist. We do not touch the money nor do either of us ever see card numbers. Our processor is one of the top three in the world. They move all the money and take the risk on your behalf.

Do I need to be a registered Payment Facilitator to use Digitzs?

Not at all! With a single integration to our API, you'll get all the benefits of registering as a Payment Facilitator without any of the cost, delays, risk or overhead.

What types of software platforms are perfectly suited for Digitzs?

We created our solution for large white label platforms that process ticketing payments for events, fine payments for city governments, tuition payments for schools, etc. The more new merchants added monthly, the higher the pain point and the better the fit.

What payment types can our merchants accept and what are your fees?

We process Visa, Mastercard, Discover, American Express, Automated Clearing House (ACH), and Apple Pay for U.S. merchants. We'll charge your merchants 2.9% and $.30 on US cards and 3.9% and $.30 on non-US cards. ACH transactions cost .20% (20 basis points) so $100 ACH will cost $.20.

Can I use Digitzs for international payments?

US-based merchants can accept payments in US dollars from anywhere in the world using the Digitzs API.

What about Chargebacks? Who takes the loss?

Your merchants are responsible for their chargebacks. If for some reason they don't make us whole on the loss, you are in the clear and we'll take the hit. We help merchants resolve chargeback issues quickly and effortlessly.

Do you offer ACH in house?

Yes and a major differentiator is that we deposit ACH with cards in a single batch.

Can we earn residuals on Digitzs' merchant fees?

Indeed. Unlike other modern payment providers, we pay you a substantial share of revenue on our 2.9% and $.30 merchant fees based on your current and near-term processing potential.

Can we add our platform fees to Digitzs' merchant fees?

You bet. Our ability to split a sale and deposit the proceeds to you, your merchant and even to a sales rep simultaneously sets us apart.

Do you offer cash discounting?

No as we do not perceive cash discounting to be a compliant program. However, we do have a zero-merchant fee solution that is fully compliant.

What programming languages do you support?

We are platform agnostic. Whatever floats your boat will work.

What kind of reporting do you offer?

You can build your own reporting portal inside of your UX, or, your merchants can log into our dashboard to see detailed information on their settled card and ACH transactions, track chargebacks and more. Schedule a demo today of Digitzs IQ.

How can I start Integrating?

Get in touch today. Once we understand your level of Payment Card Industry (PCI) security, processing volume and merchant risk profile, we can send you everything you'll need to get started.

Will we need to be PCI compliant?

Yes, anyone who passes cardholder data must be PCI compliant. That said, with us, you won't be storing card numbers which can help you get through an audit quicker.

Does Digitzs have a Platform Referral Program for Agents?

Absolutely. Get in touch and let's get to know each other. Software is eating the world. We've got the biggest fork!

Who Do I Contact with Questions About the API?

Contact Digitzs at techsupport@digitzs.com or 855.563.4489.

Glossary

A

Access Control List (ACL)
An optional layer of security that acts as a firewall for controlling traffic in and out of one or more subnets.

Amazon Web Services (AWS)
A subsidiary of Amazon that provides on-demand cloud computing platforms on a paid subscription basis.

API Key
A code passed in by computer programs calling an API to identify programs, developers, and user.

Application Programming Interface (API)
A set of subroutine definitions, communication protocols, and tools for building software.

Array
An ordered list of values.

Automated Clearing House (ACH) Payments
An electronic network for low payment financial transactions.

ACH Payment Request
The authorization to debit directly from the customer's checking or savings account.

Authentication
The verification of user information for access to system resources.

Authorization
The setting of permissions for access to system resources.

Address Verification System (AVS)
A process for verifying the address of a person claiming to own a credit card.

AWS Certificate Manager (ACM)
A tool for managing public SSL/TLS certificates for AWS-based websites and applications.

B

Boolean
A data type with two possible values (true or false).

Business-to-Business-to-Consumer (B2B2C)
A software platform that allows merchants to accept payments from their customers.

C

Card Payment Request
The processing of internet credit and debit card payments.

Card Refund Payment
The processing of internet credit and debit card refunds.

Card Split Payment
Processes the initial credit card transaction and any shared transaction fees.

Card Verification Value (CVV)
An authentication system used to reduce fraud for credit card internet transactions.

CloudFront
A Content Delivery Network (CDN) offered by Amazon as part of the AWS portfolio.

Create a New Customer
The setup of an account for a brand-new customer.

Create a New Merchant
The setup of a new merchant for a specific partner.

Create, Read, Update, and Delete (CRUD)
The four basic functions of persistent data storage.

Create Token for Credit Card and ACH
The designation of a single payment method (credit card or bank account) for automatic monthly payments.

D

DynamoDB
A database service that supports key-value and document data structures and is offered by Amazon as part of the AWS portfolio.

E

eCheck
The electronic transfer of money from one bank account to another, either within a single financial institution or across multiple institutions, via computer-based systems, without the direct intervention of bank staff.

Elastic Compute Cloud (EC2)
A tool offered by Amazon as part of the AWS portfolio that allows users to rent virtual computers on which to run their own computer applications.

Endpoint
The address or connection point to a web service. It is typically represented by a simple HTTP URL string.

F

Field
A data point included in a request.

Filter
A parameter included in a request.

Flow Log
A tool offered by Amazon as part of the AWS portfolio that captures information about the Internet Protocol (IP) traffic going to and from network interfaces in your Virtual Private Cloud (VPC).

G

GEO Restriction
Technology that blocks access to Internet content based upon the user's geographical location.

GET Authentication
A request to view authentication information.

GET Authorization
A request to view authorization information.

GET Customer by ID
A request to view specific customer information.

GET Customer List
A request to view a list of customer information.

GET Merchant by ID
A request to view specific merchant information.

GET Merchant List
A request to view a list of merchant information.

GET Payment by ID
A request to view specific payment information.

GET Payment List
A request to obtain a list of payment information.

GET Token by ID
A request to view information about a specific token.

GET Token List
A request to view a list of token information.

H

Hostname
A label assigned to a device connected to a computer network that identifies it for electronic communication.

Hypertext Transfer Protocol (HTTP)
An application protocol for distributed, collaborative, hypermedia information systems.

Hypertext Transfer Protocol Secure (HTTPS)
The use of HTTP for communication with encryption for authentication of the visited website and protection of the privacy and integrity of the exchanged data.

I

Internet Gateway
A piece of networking hardware used in telecommunications for telecommunications networks that allows data to flow from one discrete network to another.

J

JavaScript Object Notation (JSON)
An open-standard file format that uses human-readable text to transmit data objects consisting of attribute–value pairs and array data types (or any other serializable value). It is a very common data format used for asynchronous browser–server communication, including as a replacement for XML in some AJAX-style systems.

JSON Web Token (JWT)
A JSON-based open standard (RFC 7519) for creating access tokens that assert some number of claims.

K

Key
A parameter that determines the functional output of a cryptographic algorithm.

L

Lambda
An event-driven, serverless computing platform provided by AWS. It is a computing service that runs code in response to events and automatically manages the computing resources required by that code.

M

Multi-factor Authentication (MFA)
An authentication method that only grants user access after successfully presenting two or more identification codes to an authentication mechanism.

MySQL
An open source Relational Database Management System (RDBMS).

N

Network Address Translation (NAT) Gateway
A method of remapping one Internet Protocol (IP) address space into another by modifying network address information in the IP header of packets while they are in transit across a traffic routing device.

Null
An empty value within a response.

O

Object
An unordered collection of name/value pairs where the keys are strings.

P

Payment Card Industry (PCI)
The debit, credit, prepaid, e-purse, ATM, and POS cards and associated businesses.

Payment Card Industry Data Security Standard (PCI DSS)
An information security standard for organizations that handle branded credit cards from the major card schemes.

Payment Card Industry Security Standards Council
Formed by American Express, Discover, JCB, Mastercard, and Visa with the goal of managing the ongoing evolution of payment card security standards.

Postman
A software tool used to develop and run APIs.

Protocol
A set of communication rules that allow two or more entities to transmit information.

R

Relational Database Service (RDS)
A distributed relational database service by AWS. It streamlines the setup, operation, and scaling of a relational database for use in applications.

Request
A message sent to a server resource looking for data that matches specified parameters.

Representational State Transfer (REST)
A software architectural style that defines a set of constraints to be used for creating web services.

Response
A reply that contains data that matches the parameters specified in a request.

S

Script
A programming language that supports scripting programs written for special runtime environments that automate the execution of tasks.

Secure Sockets Layer (SSL)
A cryptographic protocol designed to provide communications security over a computer network. SSL is the predecessor of Transport Layer Security (TLS).

Simple Object Access Protocol (SOAP)
A messaging protocol specification for exchanging structured information in the implementation of web services in computer networks.

SplitPay
The distribution of payments to multiple accounts by splitting transaction proceeds between a Digitzs merchant and a Digitzs partner at the time of transaction.

Status
A code that indicates whether the API request succeeded or encountered and error.

T

Timeout
The amount of time before the API switches to asynchronous mode.

Timestamp
A sequence of characters or encoded information identifying when a certain event occurred, usually giving date and time of day, sometimes accurate to a small fraction of a second.

Transport Layer Security (TLS)
A cryptographic protocol designed to provide communications security over a computer network.

Token Payment
Used to perform payment transactions without sensitive data going through the Digitzs system.

Token Split Payment
Used to process split payment transactions without sensitive data going through Digitzs system.

Tokenization (Tokens)
A process of substituting a sensitive data element with a non-sensitive equivalent for data security.

U

Unicode

A standard for the consistent encoding, representation, and handling of text in programming languages.

V

Virtual Private Cloud (VPC)

An on-demand configurable pool of shared computing resources allocated within a public cloud environment, providing a certain level of isolation between the different organizations using the resources.

Virtual Private Network (VPN)

A private network across a public network that enables users to send and receive data across shared or public networks as if their computing devices were directly connected to the private network.

W

Web Application Firewall (WAF)

A security safeguard that filters, monitors, and blocks HTTP traffic to and from a web application.

Web Service

A service offered by an electronic device to another electronic device, communicating with each other via the web.

Web Services Description Language (WSDL)

An XML-based interface description language that is used for describing the functionality offered by a web service.

Whitelisting

A security function that restricts system access to designated IP addresses.

X

X.509

A standard defining the format of public key certificates.

Z

Zapier

A web-based service that allows end users to integrate the web applications they use.

Index