Safon Payment Gateway API

Integration Documentation

A Comprehensive Guide for Vendors

Table of Contents

---

1. Introduction

The **Safon Payment Gateway API** enables vendors to process payments securely and efficiently. This documentation provides a comprehensive guide for full-stack developers, covering authentication, session creation, callback handling, error management, and security considerations.

Key Information

Component	Details
Admin Panel URL	`admin.safon.am`
API Base URL	`api.safon.am`
API Version	Current (unversioned)
Protocol	HTTPS only

Component	Details
Data Format	JSON

---

2. Getting Started

Prerequisites

- Active Safon vendor account
- HTTPS-enabled website for callbacks
- Secure server environment for credential storage

API Credentials

Access your unique **Client ID** and **Client Secret** through the Vendor Admin Panel:

1. Navigate to `admin.safon.am`
2. Log in to your vendor account
3. Go to **Settings** page
4. Retrieve your static credentials

> ⚠️ **Important**: Credentials are static and don't rotate automatically. If regeneration is needed due to security concerns, contact Safon support immediately.

---

3. Authentication

Bearer Token Generation

All API requests require a valid Bearer Token for authentication.

Endpoint

```
POST /api/tokens
```

Request Headers

```
http
```

```
Content-Type: application/json
```

Request Body

```
json

{
  "client_id": "YOUR_CLIENT_ID",
  "client_secret": "YOUR_CLIENT_SECRET"
}
```

Response

```
json

{
  "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
  "token_type": "Bearer",
  "expires_in": 3600
}
```

Response Codes

Status Code	Description
`200 OK`	Token generated successfully
`401 Unauthorized`	Invalid Client ID or Client Secret

Token Management

- **Expiration**: Tokens expire after **3600 seconds (1 hour)**
- **Renewal**: Generate a new token before expiration
- **No Refresh**: No refresh token mechanism available

---

4. Categories

Getting Available Categories

Retrieve the list of available categories for payment sessions. This endpoint helps you get the correct `category_id` when creating payment sessions without using an existing product.

Endpoint

```
GET /api/categories
```

Request Headers

```http
Content-Type: application/json
```

Authentication

No authentication required - This is a public endpoint.

Response

```json
[
  {
    "id": 2,
    "localeName": {
      "en": "Construction work",
      "hy": "Construction work",
      "ru": "Construction work"
    },
    "createdAt": "2025-05-19T09:10:32.447Z",
    "updatedAt": "2025-05-19T09:15:44.877Z"
  },
  {
    "id": 1,
    "localeName": {
      "en": "Schools, Trade and Vocational",
      "hy": "Schools, Trade and Vocational",
      "ru": "Schools, Trade and Vocational"
    },
    "createdAt": "2025-05-18T12:52:26.693Z",
    "updatedAt": "2025-05-19T09:15:40.492Z"
  }
]
```

Response Fields

Field	Type	Description
`id`	integer	Category ID to use in payment sessions
`localeName`	object	Category names in different languages
`localeName.en`	string	Category name in English
`localeName.hy`	string	Category name in Armenian
`localeName.ru`	string	Category name in Russian
`createdAt`	string	Category creation timestamp (ISO 8601)
`updatedAt`	string	Category last update timestamp (ISO 8601)

Response Codes

Status Code	Description
`200 OK`	Categories retrieved successfully
`500 Internal Server Error`	Server error

Usage

Use the `id` field from this response as the `category_id` when creating payment sessions without an existing product.

---

5. Product Session Management

Creating a Payment Session

Create a payment session to initiate a transaction and obtain a session URL for user redirection.

Endpoint

```
POST /api/product/session
```

Request Headers

```
http
```

```
Authorization: Bearer YOUR_BEARER_TOKEN
Content-Type: application/json
```

Request Body

Option 1: Using existing marketplace product

```json
{
  "product_id": 1,
  "callback_url": "https://example.com/api/payment-callback",
  "return_url": "https://example.com/thank-you",
  "cancel_url": "https://example.com/payment-cancelled",
  "email": "user@example.com",
  "username": "john_doe",
  "full_name": "John Doe",
  "order_id": "ORDER_123456",
  "defaultPaymentMethod": "arca",
  "availablePaymentMethods": ["arca"],
  "language": "en"
}
```

Option 2: Processing payment with product details

```json
{
  "product_name": "Custom Product",
  "price": 1000,
  "currency": "amd",
  "category_id": 1,
  "callback_url": "https://example.com/api/payment-callback",
  "return_url": "https://example.com/thank-you",
  "cancel_url": "https://example.com/payment-cancelled",
  "email": "user@example.com",
  "username": "john_doe",
  "full_name": "John Doe",
  "order_id": "ORDER_123456",
  "defaultPaymentMethod": "arca",
  "availablePaymentMethods": ["arca"],
  "language": "en"
}
```

Field Validation

Field	Type	Required	Description
`product_name`	string	Conditional	Product name/identifier (required when not using `product_id`)
`product_id`	integer	Conditional	ID from marketplace (if exists in admin panel)
`price`	number	Conditional	Exact price amount (required when not using `product_id`)
`currency`	string	Conditional	"AMD", "RUB", "USD", "EUR" (required when not using `product_id`)
`category_id`	integer	Conditional	Required when not using `product_id`
`callback_url`	string	Yes	HTTPS URL for payment notifications
`return_url`	string	Yes	HTTPS URL for user redirect after payment completion (regardless of status)
`cancel_url`	string	No	HTTPS URL for user redirect when clicking the back button on payment page
`email`	string	No	Customer email address
`username`	string	No	Customer username
`full_name`	string	No	Customer full name
`order_id`	string	No	Optional order identifier for tracking purposes
`defaultPaymentMethod`	string	No	"arca", "mir" (Optional name of the default payment method)
`availablePaymentMethods`	array	No	["arca" , "mir"] (Optional list of the available payment methods)
`language`	string	No	"en", "hy", "ru" (Optional language for the payment session page)

Note: When using `product_id`, only send the product ID along with callback and return URLs. Optionally include cancel_url (for back button navigation), email, username, full_name, order_id, and language. When not using `product_id`, you must provide `product_name`, `price`, `currency`, and `category_id` for payment processing.

Response

```
json
```

```
{
  "session_url": "https://spayon.io/session/5716f328-5adb-46fe-938f-d72f1d65b98e",
  "session_id": "5716f328-5adb-46fe-938f-d72f1d65b98e",
  "status": "pending",
  "order_id": "ORDER_123456"
}
```

Response Codes

Status Code	Description
200 OK	Session created successfully
400 Bad Request	Invalid input parameters
401 Unauthorized	Invalid or expired Bearer Token

Important Notes

- **Immediate Redirect**: Redirect users to the session URL immediately
- **Session Timeout**: Sessions expire after **15 minutes**
- **Single Use**: Each session URL can only be used once

---

6. Session Status Check

Checking Session Status

Query the current status of an existing checkout session.

Endpoint

```
GET /api/product/session/{session_id}
```

Request Headers

```http
Authorization: Bearer YOUR_BEARER_TOKEN
```

Sample Request

```http
GET /api/product/session/4ae3108a-3a1c-42df-bce9-503bbd70ab24
Authorization: Bearer YOUR_BEARER_TOKEN
```

Response

```json
{
  "sessionId": "4ae3108a-3a1c-42df-bce9-503bbd70ab24",
  "status": "failed",
  "productName": "Iphone 16S",
  "price": "10",
  "currency": "AMD",
  "categoryId": 1,
  "createdAt": "2025-06-11T17:02:51.118Z",
  "updatedAt": "2025-06-11T17:03:15.202Z",
  "orderId": "ORDER_123456"
}
```

Status Values

Status	Description
`paid`	Payment completed successfully
`pending`	Payment is being processed
`failed`	Payment failed or was declined
`expired`	Session has expired

Response Codes

Status Code	Description
`200 OK`	Status retrieved successfully
`401 Unauthorized`	Invalid or expired Bearer Token
`404 Not Found`	Session ID not found

---

7. Callback & Return Flow

Callback Mechanism

After a payment session completes, Safon sends a POST callback to your configured Callback URL and redirects the user to your Return URL.

Callback Request

Method: `POST`
Headers:

- `Content-Type: application/json`
- `X-Signature: HMAC_SHA256_SIGNATURE`

Callback Payload

```json
{
  "sessionId": "4ae3108a-3a1c-42df-bce9-503bbd70ab24",
  "status": "failed",
  "productName": "Iphone 16S",
  "price": "10",
  "currency": "AMD",
  "categoryId": 1,
  "createdAt": "2025-06-11T17:02:51.118Z",
  "updatedAt": "2025-06-11T17:03:15.202Z",
  "orderId": "ORDER_123456"
}
```

Signature Verification

The `X-Signature` header contains an HMAC SHA256 signature of the request body:

```
X-Signature = HMAC_SHA256(secret, request_body)
```

Where:

- `secret` is your callback secret (obtained from admin panel)
- `request_body` is the raw JSON payload

Callback Status Values

Status	Description
paid	Payment completed successfully
failed	Payment failed or was cancelled
pending	Payment is still being processed
expired	Session has expired

Callback Reliability

- **Retry Logic**: Up to **3 retries** at **5-minute intervals**
- **Timeout**: No callback for incomplete sessions after timeout
- **Timestamp Format**: UTC, ISO 8601 format

Return URL Flow

- User is redirected to your configured Return URL after payment completion (regardless of success or failure)
- User is redirected to your configured Cancel URL when clicking the back button on the payment page (if cancel_url is provided)
- **Session ID included**: Both return and cancel URLs include `session_id` as a query parameter
- Example: `https://example.com/thank-you?session_id=4ae3108a-3a1c-42df-bce9-503bbd70ab24`
- **Important**: Cancel URL redirect does not change the session status - the session remains active until it expires (15 minutes) or payment is processed
- Handle the redirect gracefully in your application

---

8. Vendor Admin Panel

Configuration Setup

Access the Vendor Admin Panel at `admin.safon.am` to configure your integration settings.

Required Configurations

1. **Callback URL**: HTTPS endpoint to receive payment notifications
2. **Return URL**: HTTPS page where users are redirected after payment completion

3. **Cancel URL**: HTTPS page where users are redirected when clicking back button (optional)
4. **API Credentials**: Retrieve your Client ID and Client Secret

URL Requirements

- Must use **HTTPS protocol**
- Must be publicly accessible
- Should handle both GET and POST requests appropriately

Multiple Environments

- Currently not supported through the panel
- Contact Safon support for multi-environment setup assistance

Testing Configuration

- No built-in callback testing mechanism
- Request testing tools from Safon support
- Use session status endpoint for validation

---

9. Integration Flow

Step-by-Step Implementation

Step	Action	Endpoint	Details
1	**Obtain Credentials**	Admin Panel	Retrieve Client ID and Client Secret
2	**Generate Token**	`POST /api/tokens`	Get Bearer Token (1-hour expiry)
3	**Get Categories**	`GET /api/categories`	Retrieve available categories (optional)
4	**Create Session**	`POST /api/product/session`	Submit product details, receive session URL
5	**Redirect User**	Session URL	Direct user to Safon payment page
6	**Handle Callback**	Your Callback URL	Process POST callback with payment result
7	**User Return**	Your Return URL	Handle user redirect after payment
8	**Status Check**	`GET /api/product/session/{session_id}`	Optional status verification

10. Support & Resources

Getting Help

- **Email Support**: Contact via official Safon support channels

Additional Resources

- **Swagger Documentation**: `https://api.safon.am/api/vendor/docs`
- **Admin Panel**: `admin.safon.am`
- **Status Page**: Check for service status updates

Appendix

Supported Currencies

Currency Code	Currency Name	Symbol
`USD`	US Dollar	$
`RUB`	Russian Ruble	₽
`AMD`	Armenian Dram	Ö
`EUR`	Euro	€

Contact Safon support to request additional currency support.

HTTP Status Code Reference

Code	Status	Description
200	OK	Request successful
400	Bad Request	Invalid request parameters
401	Unauthorized	Authentication required or failed
404	Not Found	Resource not found

Code	Status	Description
429	Too Many Requests	Rate limit exceeded
500	Internal Server Error	Server error

Need Help? Contact Safon support for technical assistance and integration guidance.