

Composable & Headless Commerce 2022

Vendors & platforms compared - by developers for developers

Author:
Johannes Thorn

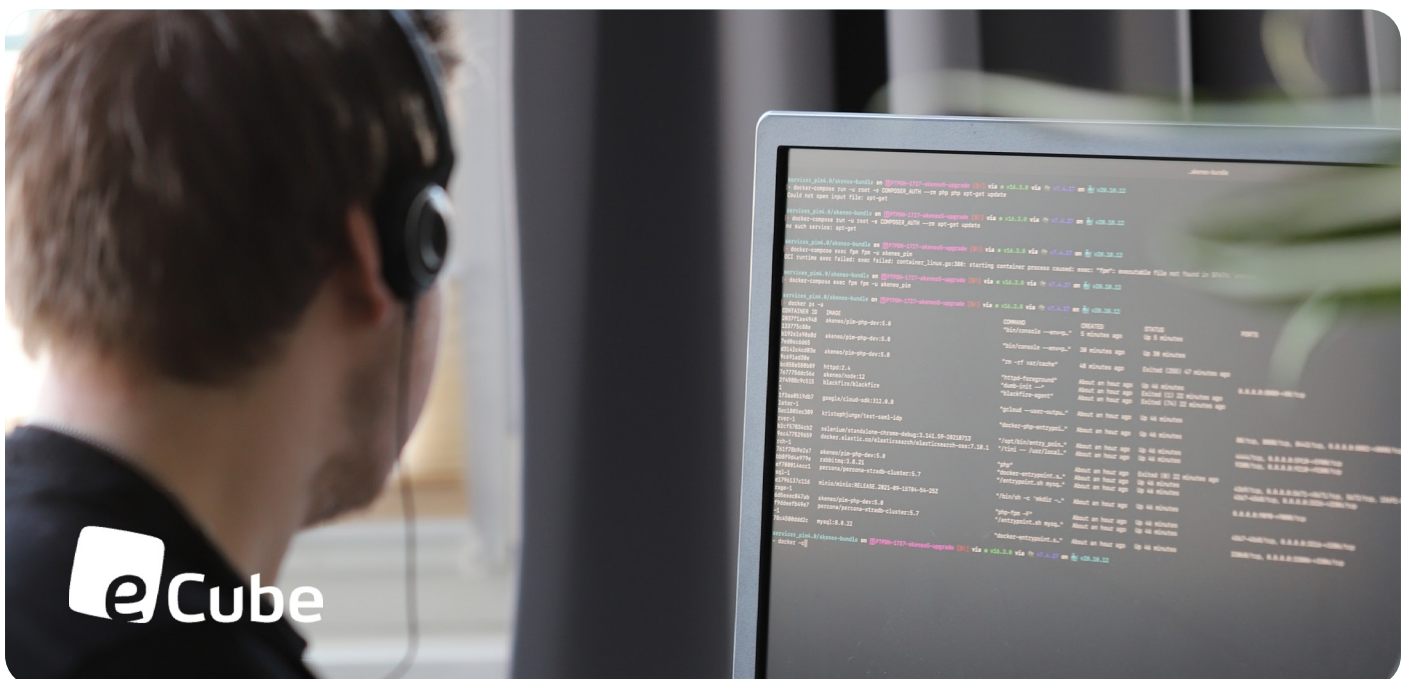
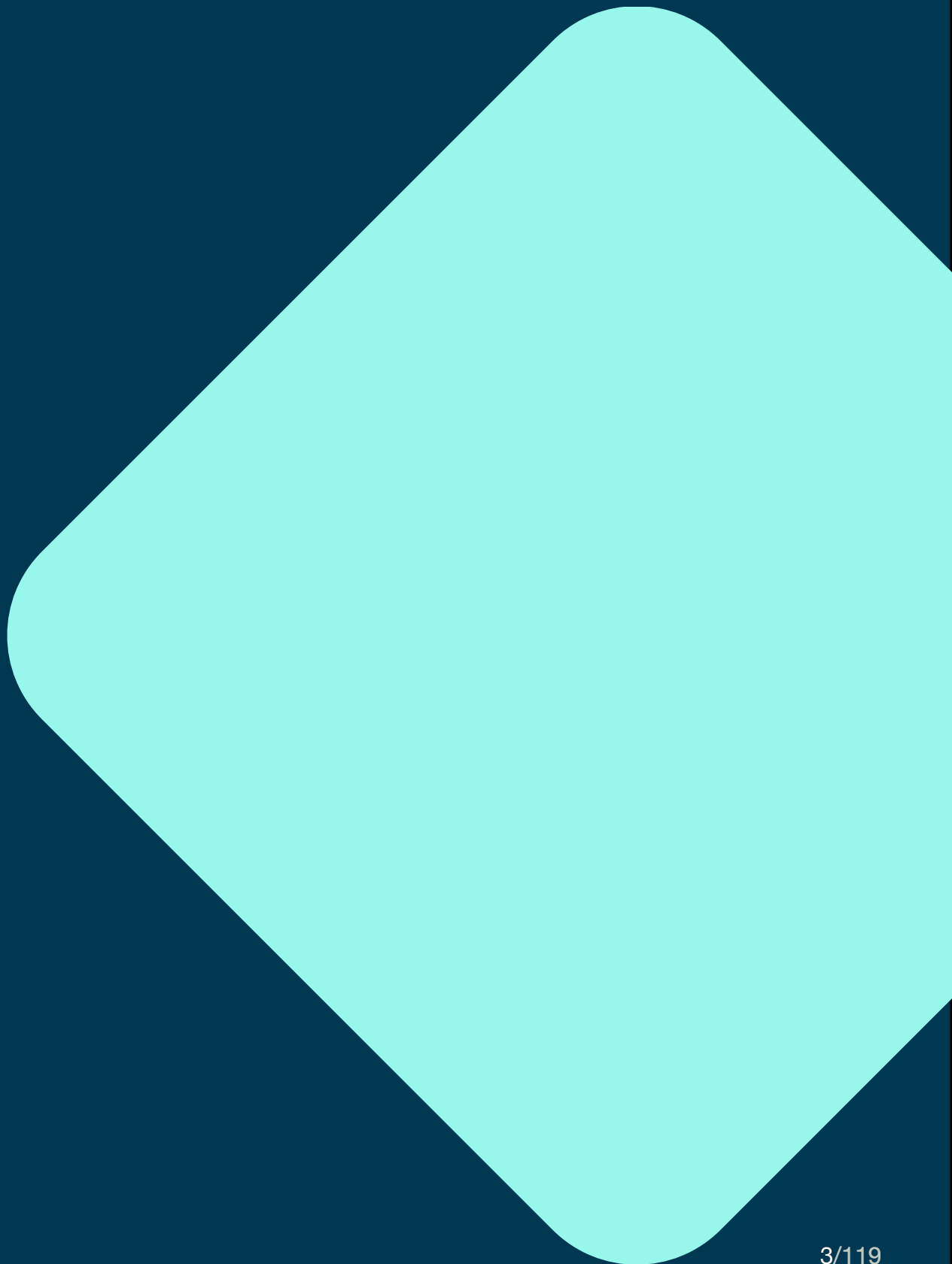


Table of Content

Introduction	03
BigCommerce	09
Commerce Layer	18
commercetools	28
Elastic Path	40
Emporix	49
Fabric	60
Salesforce Commerce Cloud	70
Shopify	79
Spryker	90
Swell	99
VTEX	110
Methodology	118

01 Introduction



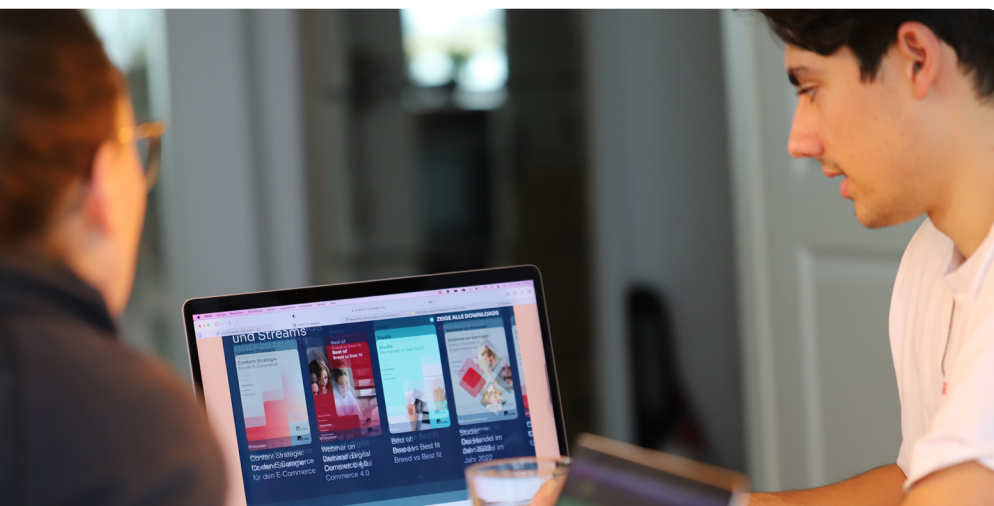
Introduction & Key Findings

Finding the right e-commerce platform can be painful, especially since many solutions that seem simple to introduce at first glance turn out to be more complex during implementation than project management and IT architects had expected.

To help developers evaluate solutions, in this study we examined and compared common composable commerce platforms for strengths and weaknesses in test environment, APIs and documentation that could make access and first steps difficult and time consuming. We also checked the platforms for useful extras that can make it easier to get started.

In summary, each platform has strengths and weaknesses and there is not one platform that meets all individual requirements. Well-known brands do not per se offer the easiest solution for developers just because they are promoted with a lot of marketing effort. "Best of breed" doesn't always mean "Best fit" for developers. This is one key finding from this study.

Note: The results of this analysis do not claim to be complete and can only represent the insights that we were able to obtain per platform in half a day of dealing with it. In some cases, providers may offer solutions or workarounds for flaws that we have identified, but we were not able to consider them during our initial review. In these cases, we appreciate a brief feedback from the providers.



Key Findings

- **Test environment:** Only some of the platforms allow developers to retrieve credentials for testing themselves, which would simplify and speed up the process of getting started. The same is true for creating initial content, product, price, product category, and testing features, for example, adding product to cart.
- **APIs:** Since all vendors take an API-first approach, they all offer common API types that are more or less intuitive to use and help developers customize. Unfortunately, we were only able to test API performance on a few platforms for various individual reasons, for example because they were not clearly specified.
- **Documentation:** An easy and comfortable start depends on good documentation. This is especially true for developers who want to get started without having to ask for support and watching product demos. Here, almost all providers leave much room for improvement, for example, when it comes to understanding API extensions or triggering API calls directly from the documentation.
- **Useful extras:** In addition to well-structured documentation, providers can help make the developers' work easier with useful extras, such as SDKs for various languages, templates for store frontends, demo data, or a developer community. Not all providers make use of this opportunity; as with the documentation, we also see a need to catch up here.

Results: Overview

The following tables provide an overview of the results of our analysis, which are explained in detail below for each platform.

Table legend

✓ **yes**
available, easy
satisfactory

? **no indication**
e.g. because it could
not be checked

✗ **no**
not available,
hard to understand,
room for improvement

BigCommerce
Commerce Layer
commercetools
Elastic Path
Emporix
Fabric.inc
Salesforce Commerce Cloud
Shopify
Spryker Cloud Commerce OS
Swell
VTEX

Environment											
Can I retrieve test credentials by myself?	✓	✓	✓	✗	✓	✗	✗	✓	✗	✓	✗
Is the test environment equal to the productive environment?	✓	✗	✓	? ²	✓	? ²	? ²	✓	✓	? ¹	? ²
Am I able to do the first content creation* steps without additional setup of the environment?	✓	✗	✗	? ²	✓	? ²	? ²	✗	✗	✗	? ²
APIs											
What kind of API is provided?	RESTful API + GraphQL	REST API	REST API + GraphQL API	REST API	REST API	REST API	RESTful API	REST API GraphQL API	REST API	REST API	RESTful API
Is there a concept of how I can extend the data model?	✓	✓	✓	✓	✓	✗	? ²	✓	✓	✓	✗
Are the error responses detailed enough?	✓	✓	✓	? ⁶	✓	? ⁴	✓	✓	? ⁵	✗	✗

	BigCommerce	Commerce Layer	commercetools	Elastic Path	Emporix	Fabric.inc	Salesforce Commerce Cloud	Shopify	Spryker Cloud Commerce OS	Swell	VTEX
APIs											
Is it possible to run meaningful performance tests?	? ³	✓	✓	✗	? ³	? ⁴	? ⁸	? ³	? ³	? ³	? ⁸
Is the authorization concept conclusive?	✓	✗	✓	✓	✓	? ⁴	✓	? ⁹	✓	✓	✗
Is the API intuitive?	✓	✗	✗	? ⁸	✓	✗	✓	✓	✓	✓	✓
Documentation											
Is the documentation easy to access/ find?	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓	✓
Is the documentation well structured?	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✓
Seems the documentation to be complete?	✓	✗	✓ ✗ ¹⁰	✓	✓	✗	✗	✓	✗	✗	✓
Are the API extensions explained properly?	✓ ✗	✗	✓	✗	✓	✗	✗	✓	✗	✗	✗
Can I generate API calls directly from the documentation?	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
Useful Extras											
Are SDKs provided for my most important language?	✓	✗	✓	✗	✗	? ¹¹	✗	✓	✗	✓	✗
Does a template for a simplistic shop frontend exist?	✓	✓	✓	✗	✗	? ¹¹	✓	✓	✓	✓	✓
Is there useful additional tooling?	✓	✓	✓	? ¹²	✗	? ¹¹	✗	✓	✓	✓	? ¹³

BigCommerce
 Commerce Layer
 commercetools
 Elastic Path
 Emporix
 Fabric.inc
 Salesforce Commerce Cloud
 Shopify
 Spryker Cloud Commerce OS
 Swell
 VTEX

Useful Extras

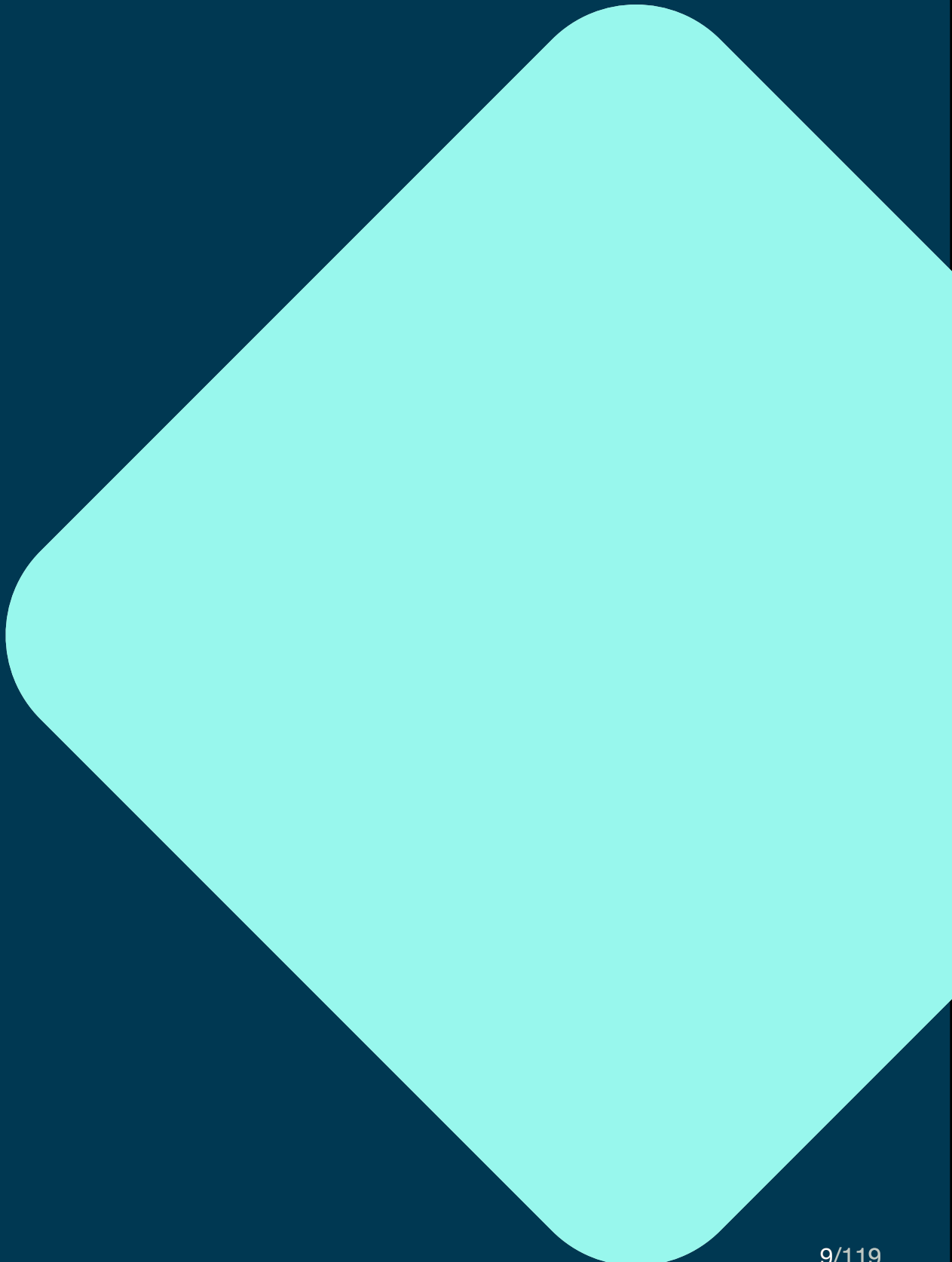
Is demo data provided?	✓	✓	✓	? ¹²	✗	✗	? ¹³	✗	✗	✗	✗
Is there a community??	✓	✓	✓	✗	✗	✗	✓	✓	✗	✗	✓

* Create a product, add a price, add the product to a category, create a cart, add the product to the cart

1. Not specified
2. Not able to sign up and check
3. Not clearly specified, no restrictions found
4. Not explained and not possible to check on our own
5. API reference provides no examples
6. Unclear as no access to API is provided
7. Remains unclear after studying the documentation
8. Can't be checked because it is not available
9. The usage appears unclear
10. The documentation seems to contain almost every relevant information but did not fit the particular intent.
11. Not possible to check due to incomplete documentation
12. Not specified in the documentation
13. Functionality cannot be checked with medium effort

02

BigCommerce



Introduction

BigCommerce, founded in 2009, provides an “e-commerce software platform that provides startups and established companies ... with an easy-to-use interface and advanced built-in features.”¹

¹ <https://digital.com/best-ecommerce-platforms/bigcommerce>

Test Environment

How can I retrieve test credentials?

- How difficult is it to retrieve them?
- What additional steps are needed before I can start my tests?
- Do I have to wait for approval or do I need to get a demo first?

↳ **Self-registration is possible without the need for approval. The test duration is quite short with only 15 days.**

How close is the test environment to the production environment?

- Am I restricted in what I can do?
- Is the performance close to the final product?
- Could I transform my testing environment into a production one?

↳ **The trial environment seems to be ready for production. During trial the storefront is only visible in preview mode and requires a preview code.**

Am I able to do the following steps without any external help and without additional setup of the environment?

1. Create a product,
2. add a price,
3. add the product to a category,
4. create a cart,
5. add the product to a cart

↳ **A new product with price could be added. Then the product was added to a category. Also shopping carts containing the product could be created via the management API.**

APIs

What kind of API is provided?

- REST, GraphQL or other?

↳ **Management REST API, Storefront REST API, GraphQL API, Customer Login API, CurrentCustomer API.**

Is there a concept of how I can extend the data model of a product or other entities?

- How do I add additional fields to products, customers, carts, orders, etc.?

↳ **Custom data can be stored on variants, products, categories, and brands via meta-fields which are key-value pairs. External services can be integrated via webhooks.**

How detailed are the error responses?

- Does the API tell me which fields are invalid, e.g. if I have an invalid payload?
- Do I get in general helpful error messages?

↳ **The API reference specifies error responses per API endpoint including payload examples. The responses contain error descriptions and details including missing or invalid fields.**

Is it possible to run meaningful performance tests?

- Do I have the same or a similar performance than in production?
- If there are restrictions am I still able to run a performance test?

↳ **The environment seems to be production ready and no performance limitations are mentioned.**

How good is the authorization concept?

- Is the authorization concept
 - easy to understand
 - flexible to different needs

↳ **By splitting into multiple APIs, authorization is handled differently per API and allows for flexible configuration.**

Is the API well structured/intuitive?

↳ The structure seems to be well formed.

Documentation

Is the documentation easy to access/ find?

- Do I have direct (or with low effort) access to the documentation?
- Or is there some kind of approval needed to get access?

↳ The development center documentation is accessible from the homepage without prior registration or approval.

Is it well structured?

- Are there guides to get me started?
- How do I add a product?
- How to add a product to the cart?
- Documented best/accepted practices?
- Is there a getting started guide?

↳ Guides exist for the storefront and management APIs as well as the GraphQL API. Some of these are detailed explanations of the various steps required to achieve a goal, while others are more of a collection of examples of use, with no additional explanation of why something is done a certain way.

Completeness

One guide explicitly states some best practices, while other guides contain smaller sections of best practices related to the guide's specific topic.

It is not entirely clear how the selection of topics was made, since, e.g. the creation or import of products is not described, while the management of email templates is covered. This also means that the getting started guide only covers the necessary steps up to the first API request.

- Are there meaningful payload examples in the API documentation?
- Are all possible error responses documented in the API documentation?

↳ Both the textual documentation and the API reference contain realistic examples of payloads. The reference even contains multiple examples of requests per use case for some of the endpoints. The reference also specifies all error responses per endpoint with semantic interpretations of the status codes and further explanations.

API extensions

- How well are the extension concepts explained?
- Can I extend the data model on my own?
- Can I include external services?

↳ **The mechanism for adding custom attributes is not explicitly explained and could only be discovered by searching the API reference for typical naming of this feature. It is not clear which resources can be extended and which cannot.**

In contrast, the management of webhooks is explained in detail and even received a dedicated tutorial for local testing the created webhook.



Can I generate API calls directly from the documentation?

- Is there a try-it-out function in documentation or is there some kind of code (e.g. curl or Postman collections) I can copy from the documentation?

↳ **The API reference allows to send test requests directly from the browser while maintaining full control over the request. The result also contains the request as it was sent over the wire. Furthermore, it is possible to generate and copy the matching code for different languages and libraries.**

Useful Extras

Are SDKs provided for my most important languages?

Language	SDK available?
Java	no
PHP	yes
JavaScript	yes
DotNET Core	no

↳ In addition, SDKs are available for Python and Ruby.

Does a template for a simplistic shop frontend exist?

- Minimum desktop (bonus points for non-desktop)?
- Access to templates GitHub repository OSS license?

↳ Quick launch templates for multiple platforms and frameworks are available, e.g. Wordpress, Next.js, Nuxt.js, Drupal, Gatsby and others.

What additional tooling exists?

- Importer?

↳ Widget builder for local development of custom widgets, no Importer, Stencil CLI is a server emulator for local theme development.

Is demo data provided?

↳ The shop environment launches with some already existing demo products and categories.

Is there a community?

↳ BigCommerce provides its own community platform and is also discussed on Stack Overflow. BigCommerce also provides a dedicated learning platform.

Noteworthy

↳ The split into multiple APIs simplifies considerations about the best approach to reach a goal. There are plentiful guides, but they could be better organized to allow for better progress in learning the platform.

Conclusion

↳ Our first impression: very good. BigCommerce makes it very easy to start.

Test environment: fully meets our expectations

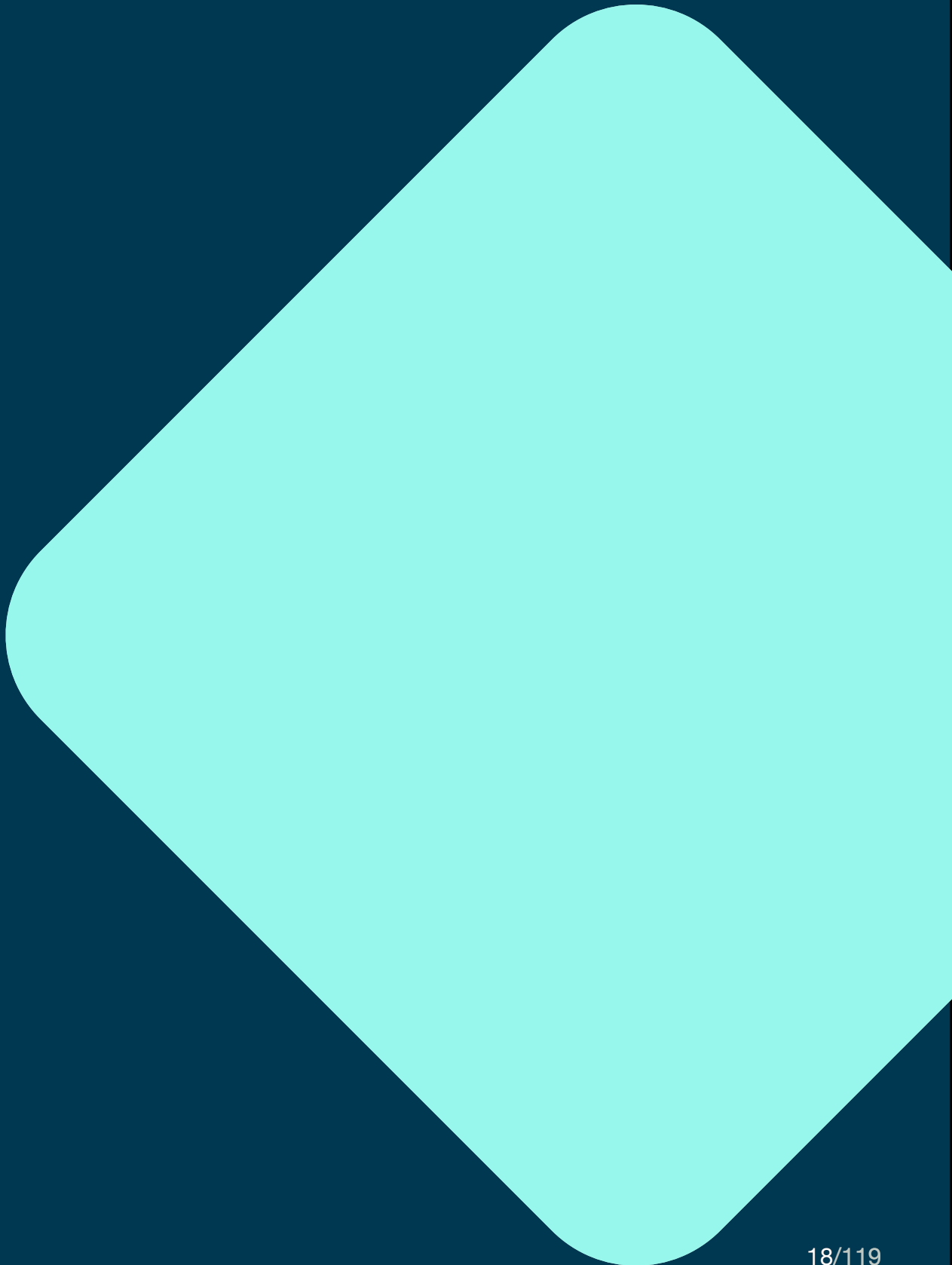
APIs: fully meets our expectations, performance could not be tested

Documentation: could be improved for basic tasks

Useful extras: leaves nothing to be desired

03

Commerce Layer



Introduction

Commerce Layer, founded in 2017, provides “a transactional commerce API that lets you add multi-market ecommerce to any digital experience, with ease. Make it shoppable with Commerce Layer APIs and pre-built integrations.”²

² <https://commercelayer.io/APIs>

Test Environment

How can I retrieve test credentials?

- How difficult is it to retrieve them?
- What additional steps are needed before I can start my tests?
- Do I have to wait for approval or do I need to get a demo first?

↳ **Self-registration by e-mail and e-mail verification is possible. Demo data can be created during initial registration.**

How close is the test environment to the production environment?

- Am I restricted in what I can do?
- Is the performance close to the final product?
- Could I transform my testing environment into a production one?

↳ **Commerce Layer distinguishes between test and live mode and it is possible to switch from test to live mode after selecting a paid plan.**

Only in live mode real orders can be processed. Other restrictions are not explicitly mentioned.

Am I able to do the following steps without any external help and without additional setup of the environment?

1. Create a product,
2. add a price,
3. add the product to a category,
4. create a cart,
5. add the product to a cart

↳ **Commerce Layer does not provide category management via a category tree. It wasn't clear how to gain an access token that would allow the creation of a new product using the API alone, as the necessary roles and permissions are only briefly outlined in the description of the permissions.**

APIs

What kind of API is provided?

- REST, GraphQL or other?

↳ **Only a REST API is provided.**

Is there a concept of how I can extend the data model of a product or other entities?

- How do I add additional fields to products, customers, carts, orders, etc.?

↳ There is a concept called metadata for some of the resources. The metadata is not explained in detail how to use it. Most likely it is just a set of key-value pairs that do not allow for nested structures.

How detailed are the error responses?

- Can the API tell me which fields are invalid, e.g. if I have an invalid payload?
- Do I get in general helpful error messages?

↳ The error responses contain detailed validation errors describing the path to the validation error and the type of error. Both are helpful to identify the error. However, these validation errors can also be misleading. Trying to put a non-existent SKU into the shopping bag returned multiple validation errors, but no indication of the unknown SKU.

Is it possible to run meaningful performance tests?

- Do I have the same or a similar performance than in production?
- If there are restrictions am I still able to run a performance test?

↳ **No specific restrictions mentioned.**

How good is the authorization concept?

- Is the authorization concept
 - easy to understand?
 - flexible to different needs?

↳ **The authorization concept is based on OAuth and distinguishes between sales channel, integration and web-app authorization flows, each providing a specific subset of permissions, which in turn depend on the grant type used. The permissions are pre-configured and cannot be changed.**

It is only mentioned as a side note that using the password grant type in the sales channel grant flow, all resources are pre-filtered for the logged-in customer. This concept is not fully comprehensible at first glance.

Is the API well structured/intuitive?

↳ **The API uses special attributes starting with underscores to allow status changes of some resources that do not meet the expectations of the RESTful API.**

Documentation

Is the documentation easy to access/ find?

- Do I have direct (or with low effort) access to the documentation?
- Or is there some kind of approval needed to get access?

↳ **The developer's documentation is accessible from the front page without registration and is open to everyone. Further Information can be accessed via links from the core documentation.**

Is it well structured?

- Are there guides to get me started?
- How do I add a product?
- How to add a product to the cart?
- Documented best/accepted practices?
- Is there a getting started guide?

↳ **A getting started guide covers the process from registration to the first API call. Other guides describe specific scenarios and how those issues are resolved with the API. However, adding new products is not covered by a guide.**

Completeness

- Are there meaningful payload examples in the API documentation?
- Are all possible error responses documented in the API documentation?

↳ The document-like part of the API reference contains examples of both requests and the corresponding responses. However, only the successful cases per resource are described. Error responses are summarized only once in an overview over all possible errors and are not specified per resource.

API extensions

- How well are the extension concepts explained?
- Can I extend the data model on my own?
- Can I include external services?

↳ Commerce Layer provides multiple pre-defined integration points with external services, such as external payment gateways, external prices, external promotions, and external tax calculators. In each case, the external service must match to the response pattern defined by the API. Furthermore, webhooks can be defined to be notified on a variety of topics, from new order creations to parcel tracking status changes. Currently it doesn't seem to be possible to extend the data model with custom attributes.

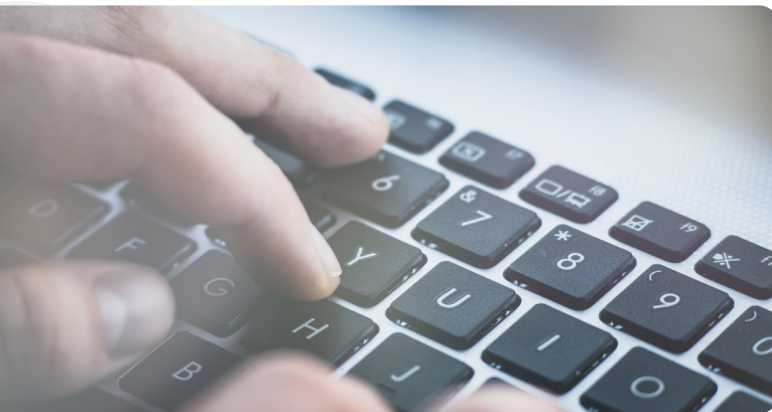


Can I generate API calls directly from the documentation?

- Is there a try it out function in documentation or is there some kind of code (e.g. curl or Postman collections) I can copy from the documentation?

↳ The API Reference contains curl calls with sample data and the corresponding response payloads. The documentation doesn't provide a test function. A Postman collection and an openAPI definition are also provided for download. The Postman collection could not be imported successfully with a non-descriptive error.

On the other hand, the openAPI definition does not describe the response bodies for all requests and does not specify the full set of possible parameters for all requests. It is also not consistent with the document-like part of the API reference.



Useful Extras

Are SDKs provided for my most important languages?

Language	SDK available?
Java	no
PHP	no
JavaScript	yes
DotNET Core	no

Does a template for a simplistic shop frontend exist?

- Minimum desktop? (Bonus points for non-desktop)
- Access to templates GitHub repository OSS license?

↳ Commerce Layer provides a collection of reusable React components on GitHub. There are several shop templates and demos that demonstrate integration with various headless CMS systems.

What additional tooling exists?

- Importer?

↳ Commerce Layer provides a CLI tool which supports managing resources from the command line. This includes importing new resources and CRUD operations for existing resources. The latest addition is a hosted checkout app that can be hosted by Commerce Layer or self-hosted with the necessary modifications.

Is demo data provided?

↳ Demo data can be created during registration.

Is there a community?

↳ **There is a Commerce Layer community in Slack that anyone can join. We have not verified how active the community is.**

Noteworthy

↳ **The guides are organized by scenario following a possible customer journey, which simplifies the learning.**

Not all decisions on the presentation of prices are comprehensible:

Price values (amount_cents and compare_at_amount_cents) must be expressed in cents with the only exception of HUF and JPY currencies which do not have decimals (e.g. 100\$ - "amount_cents": 10000, 100Ft - "amount_cents": 100, 100¥ - "amount_cents": 100).

Conclusion

↳ **Our first impression: ok. The API seems appropriate for implementing a new shop.**

Test environment: not equal to production environment, content creation could be easier.

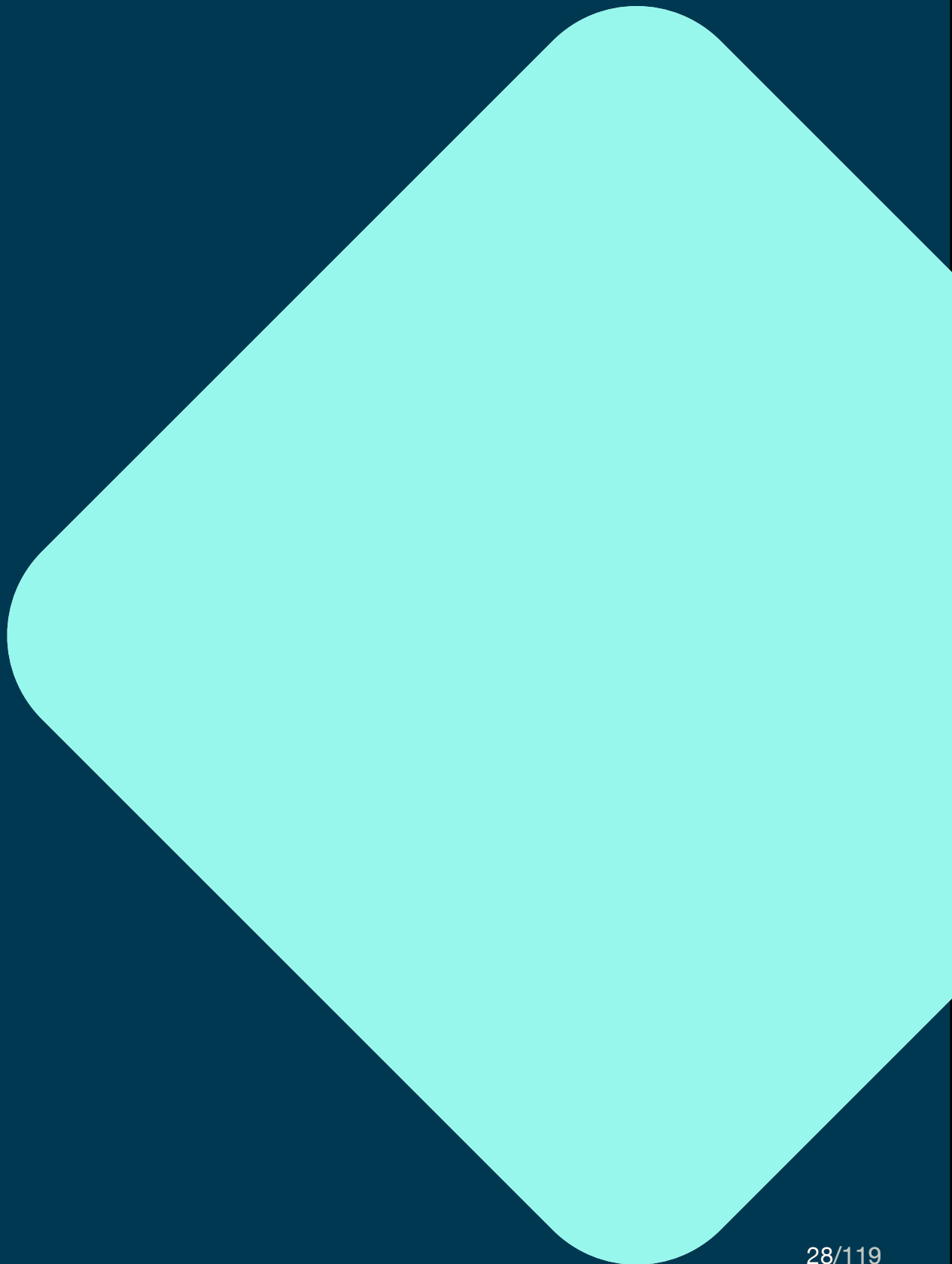
APIs: Authorization concept is not conclusive, APIs could be more intuitive.

Documentation: Overall very good, textual API reference could be better explained.

Useful extras: Overall very good, SDKs for relevant languages would be helpful.

03

commercetools



Introduction

commercetools, founded in 2013, provides “the world’s leading digital commerce platform that allows you to create powerful, highly customized commerce experiences while building a profitable, sustainable brand.”³

³ <https://commercetools.com/company/about>

Test Environment

How can I retrieve test credentials?

- How difficult is it to retrieve them?
- What additional steps are needed before I can start my tests?
- Do I have to wait for approval or do I need to get a demo first?

↳ **Requires self-signup via email, free trial duration is 60 days.**

Before running any tests from the getting started guide, a new API client must be created. It also recommends importing some demo data which is provided via an npm project. The credentials can be downloaded as cURL, SUNRISE, YAML, Java, JavaScript, PHP, iOS, Environment Variables, and Postman configuration.

No approval or demo is needed before using the API.

How close is the test environment to the production environment?

- Am I restricted in what I can do?
- Is the performance close to the final product?
- Could I transform my testing environment into a production one?

↳ **No reference to restrictions, the environment is production-ready, as staging and production are handled via a “current” and a “staged” version of the product data. Since there are no two different environments, there are also no differences in the possible performance.**

Am I able to do the following steps without any external help and without additional setup of the environment?

1. Create a product,
2. add a price,
3. add the product to a category,
4. create a cart,
5. add the product to a cart.

↳ **Unfortunately the creation of a new product via Postman collection fails because a value for the field “baseId” is required. However, this field is not documented or included in the search results. The only reference is found in a GitHub issue, which was closed without resolution.**



APIs

What kind of API is provided?

- REST, GraphQL or other?

↳ REST API and GraphQL API.

Is there a concept of how I can extend the data model of a product or other entities?

- How do I add additional fields to products, customers, carts, orders, etc.?

↳ The functionality provided can be extended via API extensions/webhooks. The mechanism is explained in a dedicated tutorial focused on serverless functions. API extensions must be registered on specific events to trigger a call to the external function.

How detailed are the error responses?

The external function can return a JSON representation of the internal action to be executed on return.

Additional fields can be added to some of the resources via custom types and fields. Custom types extend the internal types by subtyping. Custom fields are registered with these types.

- Can the API tell me which fields are invalid, e.g. if I have an invalid payload?
- Do I get in general helpful error messages?

↳ All error messages describe the problems in full detail and even describe a compound parameter problem, where the problem was with the combination of two parameters. The messages are very helpful for troubleshooting.

Is it possible to run meaningful performance tests?

- Do I have the same or a similar performance than in production?
- If there are restrictions am I still able to run a performance test?

↳ There is no explicit mention of performance limitations. As the provided test environment contains production

How good is the authorization concept?

and staging data at the same time, no explicit performance limitations are expected. Instead, there are api limits that are intended to lead to appropriate use in production and test environments.

- Is the authorization concept
 - easy to understand?
 - flexible to different needs?

↳ The authorization is based on OAuth and provides various authorization flows for different usage scenarios. There is no apparent problem with the setup. The API provides some “me” endpoints to distinguish between customer-specific and public endpoints.

Is the API well structured/intuitive?

↳ Resources creation and querying is supported as expected from a RESTful API. However, updating existing resources is done by sending update commands to the resource endpoint. This was unexpected and it feels like the internal data storage solution leaks into the API layer, as it is no longer the resource’s representation that is updated.

Documentation

Is the documentation easy to access/ find?

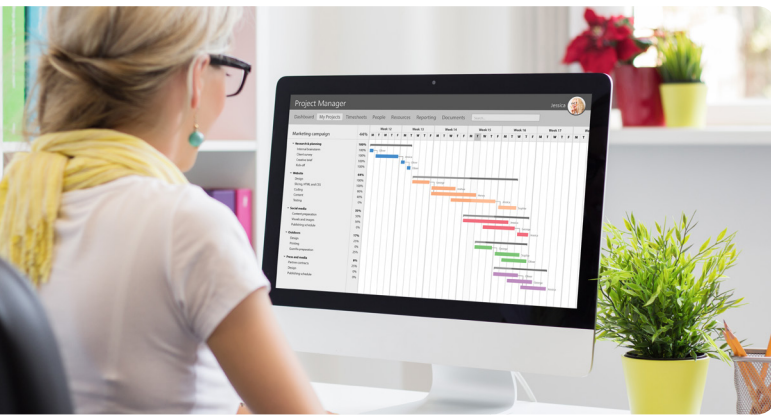
- Do I have direct (or with low effort) access to the documentation?

↳ **The technical documentation is freely accessible from the home page without requiring prior registration. Since the documentation is very extensive, it sometimes seems overwhelming and lacks a clearly marked learning path. The guides could be categorized as basic, advanced and expert to simplify the learning curve. In addition to the written technical documentation, commercetools offers paid training courses.**

Is it well structured?

- Are there guides to get me started?
- How do I add a product?
- How to add a product to the cart?
- Documented best/accepted practices?
- Is there a getting started guide?

↳ **In addition to a getting started guide which explains how authorization works and how products can be queried, there are a number of advanced guides that cover common scenarios. The scenarios include**



Completeness

managing guest carts, product bundles, product data modeling and nested product data.

There seems to be a lack of guidance that covers intermediate topics between the basic getting started guide and the more advanced guides, e.g. there is no guide for simply creating a product and adding that product to a cart. It would be nice to have more support on this level of complexity.

Some of the guides contain a section on best practices related to the current topic, but there is no general collection of best practices, with the exception of performance tips, which is a separate guide.

- Are there meaningful payload examples in the API documentation?
- Are all possible error responses documented in the API documentation?

↳ The documented online API reference does not contain any payloads or recognizable schema descriptions. Instead the documentation contains a serial collection of textual data type descriptions with very little explanation. This makes understanding the expected structure unnecessarily complicated. Payloads are only

API extensions

Can I generate API calls directly from the documentation?

included in the Postman collection.

Possible error responses are only summarized once and not specified per endpoint. The assignment between technical error and the underlying functional problem remains unclear. cases per resource are described. Error responses are summarized only once in an overview over all possible errors and are not specified per resource.

- How well are the extension concepts explained?
- Can I extend the data model on my own?
- Can I include external services?

↳ API extensions and custom fields are well explained in separate guides. Both data model extension and inclusion of external services are possible.

- Is there a try it out function in documentation or is there some kind of code (e.g. curl or Postman collections) I can copy from the documentation?

↳ There is no try out function in the documentation itself. Instead, a

Postman collection is provided, and the documentation contains curl calls and code for the Java and PHP clients to copy.

Useful Extras

Are SDKs provided for my most important languages?

Language	SDK available?
Java	yes
PHP	yes
JavaScript	yes
DotNET Core	yes

Does a template for a simplistic shop frontend exist?

- Minimum desktop? (Bonus points for non-desktop)
- Access to templates GitHub repository OSS license?

↳ **commercetools provides multiple templates and examples of how the API can be used to implement a shop or other applications. The templates and other tools are accessible via GitHub. The licenses appear to be Apache License 2.0.**

What additional tooling exists?

- Importer?

↳ A Java-based synchronization library enables the import and update of different entities from external sources or another commercetools project. In addition, there is a collection of React UI components.

Is demo data provided?

↳ Yes, there is an npm project that can be used to import prepared demo data and clean it up afterwards.

Is there a community?

↳ There is a commercetools page at GitHub which contains 143 projects with active contributions. There is also some activity at StackOverflow.

Noteworthy

↳ Searching the documentation for “baseld” is not possible as the search engine stubbornly replaces it with “based”. It is unclear how a multi-region setup can be implemented:

“The commercetools platform is provided in multiple cloud infrastructures and locations, called Regions. Self-service sign-up is only possible in the Google Cloud Regions.”⁴

“User accounts are separate per Region, authentication will fail when logging into a different Region than the one the account was created in. The same email address can have independent user accounts in multiple regions.”

Conclusion

↳ **Our first impression: very good. Easy to get started with rich, well-documented APIs.**

Test environment: overall good, no content creation without setting up the test environment.

APIs: broad, flexible, makes many things possible.

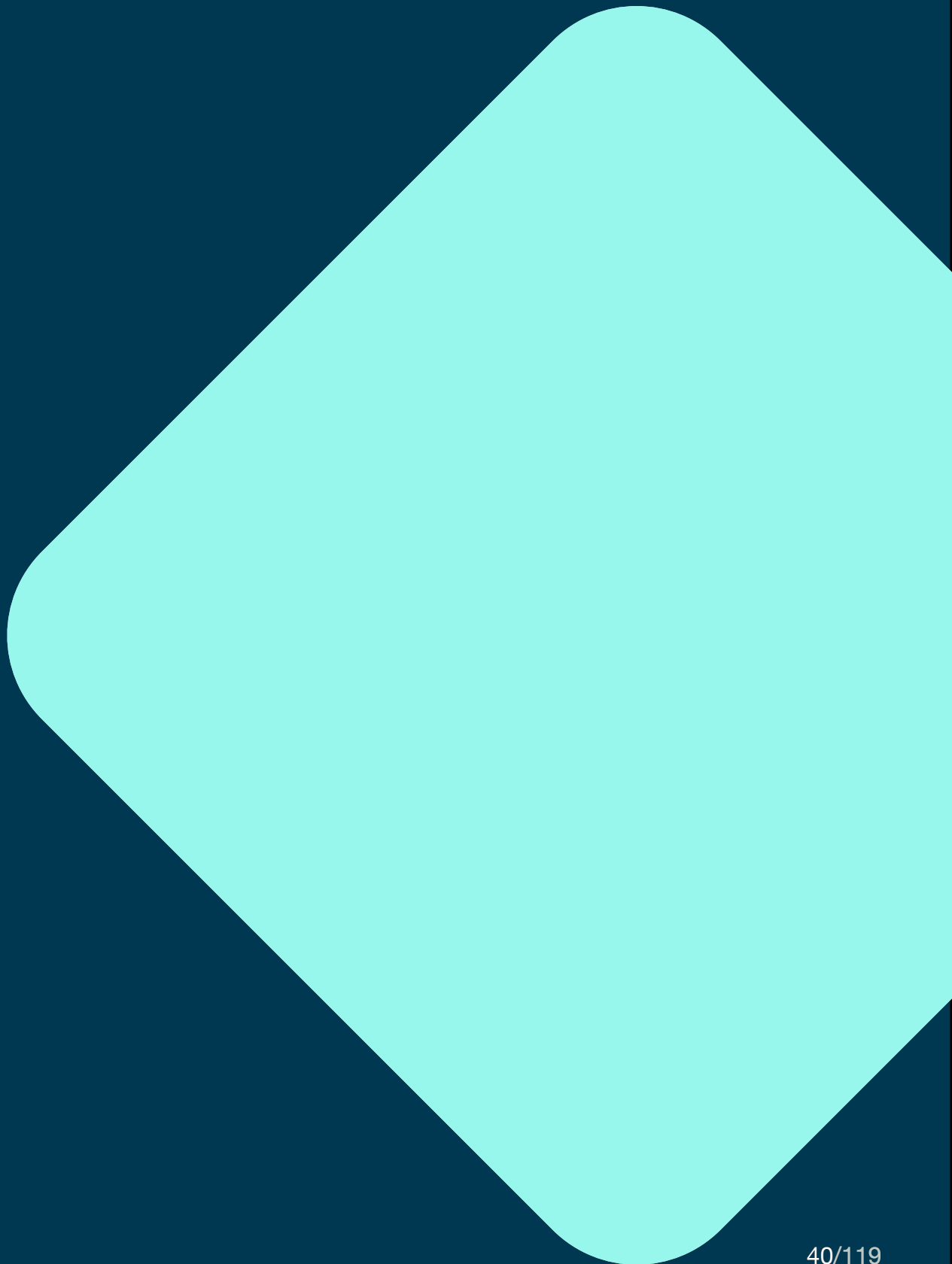
Documentation: comprehensive, almost overwhelming. No learning path.

Useful extras: leaves nothing to be desired.

⁴<https://docs.commercetools.com/docs/login>

04

Elastic Path



Introduction

Elastic Path, founded in 2000, provides “composable, API-first, headless commerce solutions that make it simple for brands with complex requirements to deliver unique digital commerce experiences, fast.”⁵

⁵ <https://www.elasticpath.com>

Test Environment

How can I retrieve test credentials?

- How difficult is it to retrieve them?
- What additional steps are needed before I can start my tests?
- Do I have to wait for approval or do I need to get a demo first?

↳ **No self-signup possible. Access to a free trial version is granted only after a personal demo. The demo library is also only accessible after registration.**

How close is the test environment to the production environment?

- Am I restricted in what I can do?
- Is the performance close to the final product?
- Could I transform my testing environment into a production one?

↳ **Could not be tested without signup.**

Am I able to do the following steps without any external help and without additional setup of the environment?

1. Create a product,
2. add a price,
3. add the product to a category,
4. create a cart,
5. add the product to a cart.

↳ **Could not be tested without signup**

APIs

What kind of API is provided?

- REST, GraphQL or other?

↳ **Only a REST API is provided.**

Is there a concept of how I can extend the data model of a product or other entities?

- How do I add additional fields to products, customers, carts, orders, etc.?

↳ **Custom attributes can be added to products via the API. New resources can be added via a concept called “custom flows” which allows the creation of custom entities.**

How detailed are the error responses?

- e.g. if I have an invalid payload. Can the API tell me which fields are invalid?
- Do I get in general helpful error messages?

↳ **Could not be tested without access to API.**

Is it possible to run meaningful performance tests?

- Do I have the same or a similar performance than in production?
- If there are restrictions am I still able to run a performance test?

↳ **Since no test environment was available, no performance tests could be performed.**

How good is the authorization concept?

- Is the authorization concept
 - easy to understand?
 - flexible to different needs?

↳ **The authentication concept distinguishes between read/write and read-only tokens based on OAuth flows and combines them with internal customer tokens to enable fine-grained access control. These also allow us to filter the resources based on the current customer.**

Is the API well structured/intuitive?

↳ **Could not be tested without access to API.**

Documentation

Is the documentation easy to access/ find?

- Do I have direct (or with low effort) access to the documentation?
- Or is there some kind of approval needed to get access?

↳ **The technical documentation is accessible without prior registration or approval.**

Is it well structured?

- Are there guides to get me started?
- How do I add a product?
- How to add a product to the cart?
- Documented best/accepted practices?
- Is there a getting started guide?

↳ **In addition to a getting started guide multiple other guides describe advanced usage scenarios.**

Completeness

- Are there meaningful payload examples in the API documentation?
- Are all possible error responses documented in the API documentation?

↳ **The API reference contains curl calls with payload examples. The error responses are only summarized once and not described specifically per endpoint.**

API extensions

- How well are the extension concepts explained?
- Can I extend the data model on my own?
- Can I include external services?

↳ **A guide explains how to extend core resources with additional custom attributes and how to create custom resources. Integration with external services is possible via classic webhooks and also via AWS SQS queues. In both cases, the integration is registered on specific internal events.**



Can I generate API calls directly from the documentation?

- Is there a try it out function in documentation or is there some kind of code (e.g. curl or Postman collections) I can copy from the documentation?

↳ The reference does not provide a try out function instead curl calls are provided.

Useful Extras

Are SDKs provided for my most important languages?

Language	SDK available?
Java	no
PHP	community
JavaScript	yes
DotNET Core	no

↳ In addition, SDKs are available for Go, Swift, and Ruby



Does a template for a simplistic shop frontend exist?

- Minimum desktop? (Bonus points for non-desktop)
- Access to templates GitHub repository OSS license?

↳ **There is a reference storefront implementation that only works with a prior version 2 of the API.**

Additionally there is an embeddable cart and checkout provided as a shop kit to be embedded into pre-existing sites.

What additional tooling exists?

- Importer?

↳ **No additional tooling could be discovered.**

Is demo data provided?

↳ **No demo data provided.**

Is there a community?

↳ **Multiple repositories exist on GitHub**

Noteworthy

↳ Nothing to mention beyond the above.

Conclusion

↳ Our first impression: mediocre. We can't say much because the platform doesn't make it easy to check everything out.

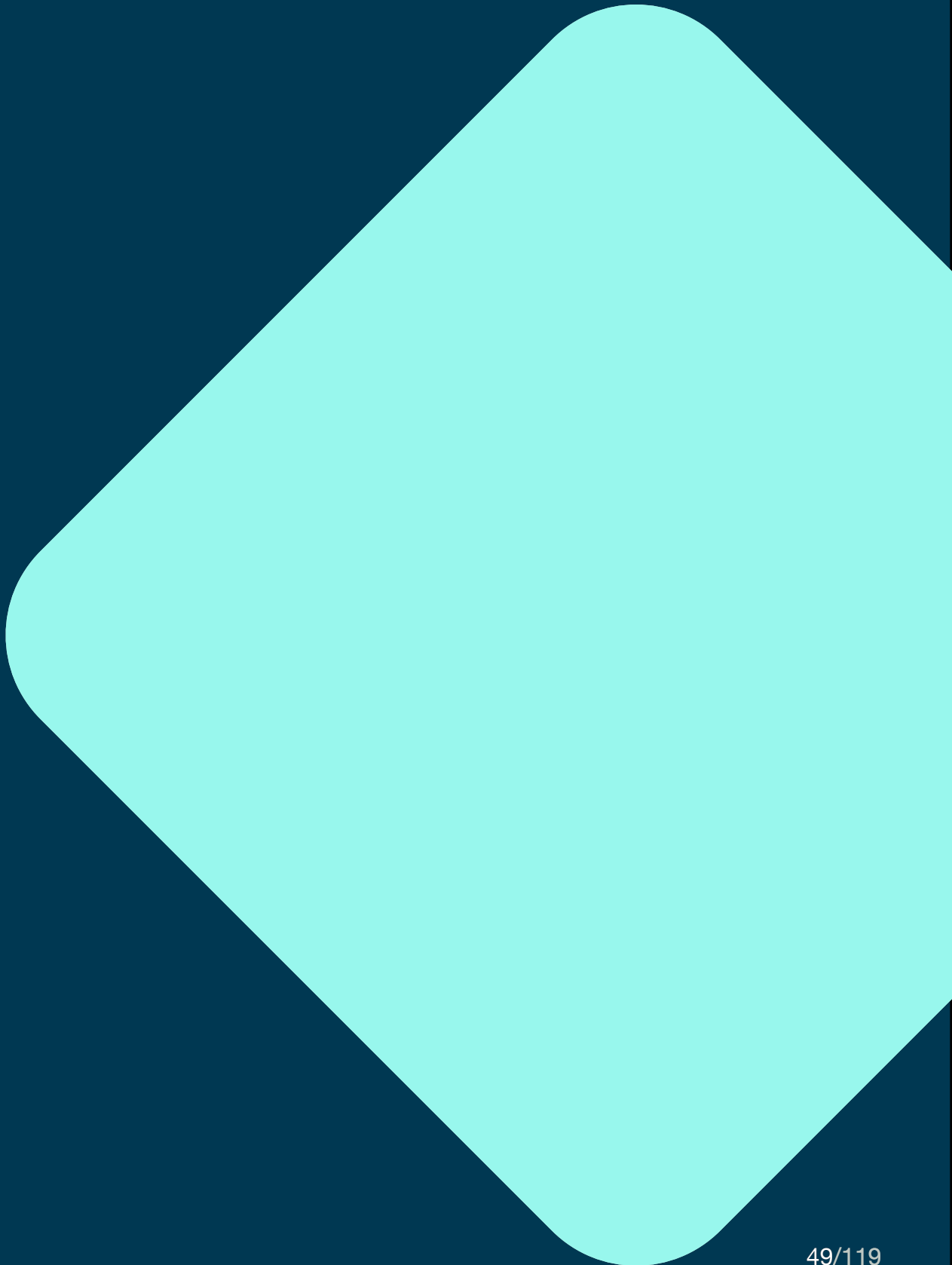
Test environment: either not available or we couldn't check

APIs: overall ok, we could only partially check

Documentation: very good, no API calls directly from the documentation.

Useful extras: mediocre, either not available or no access

05
Emporix



Introduction

Emporix, founded in 2015, provides “an enterprise-grade digital commerce platform. The Emporix Digital Commerce Platform (DCP) manages advanced commerce in B2B and B2C with a state-of-the-art composable commerce architecture.”⁶

⁶ <https://www.emporix.com>

Test Environment

How can I retrieve test credentials?

- How difficult is it to retrieve them?
- What additional steps are needed before I can start my tests?
- Do I have to wait for approval or do I need to get a demo first?

↳ **Self-signup via Microsoft, Google, LinkedIn or email is easy like any other online service. Neither an approval nor a demo was necessary to access the developer portal. Running API calls against the tenant was possible after retrieving client credentials from the developer portal.**

Some important questions left us unanswered at signup: How long will I have access? Why do I need to choose my language and currency? Can I change it later? What consequences will this have? Why NOK besides EUR, USD and GBP?

How close is the test environment to the production environment?

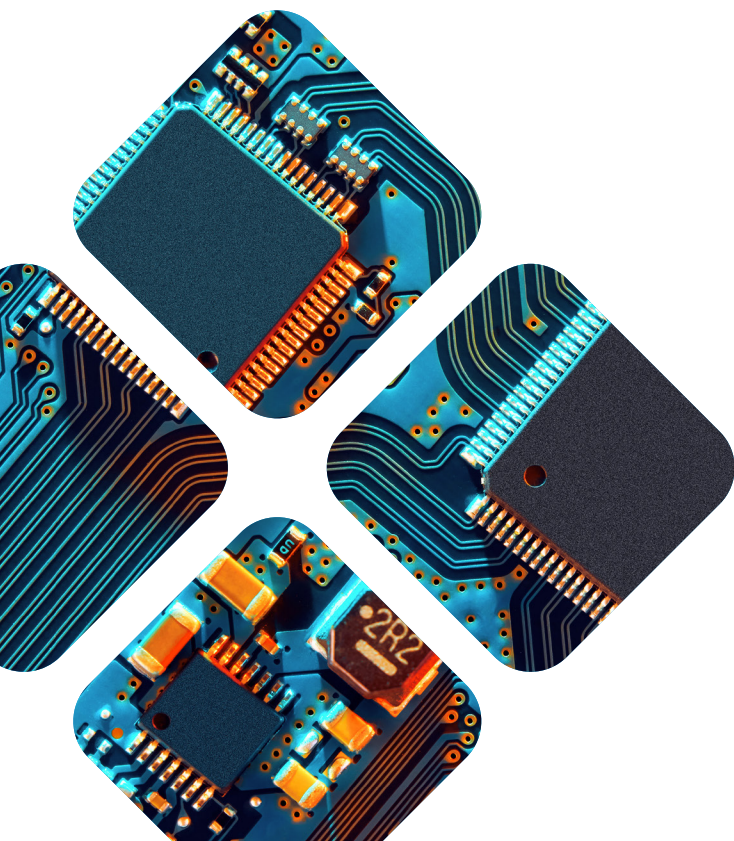
- Am I restricted in what I can do?
- Is the performance close to the final product?
- Could I transform my testing environment into a production one?

↳ Differences between production and test environments are not explicitly stated. Most likely, there are none either, since the initial environment seems to be already production ready. A second staging tenant can be created from the developer portal.

Am I able to do the following steps without any external help and without additional setup of the environment?

1. Create a product,
2. add a price,
3. add the product to a category,
4. create a cart,
5. add the product to a cart.

↳ Before creating the first product, some basic settings must be configured (see getting started guide for details). After that, products, categories and carts can be created. The following irregularities occurred: After creating the product via API call, the product is not found in the management dashboard. Repeating the category creation call creates the same category again. This was unexpected and did not happen during the duplicate call for product creation.



APIs

What kind of API is provided?

Other than that the steps could be executed smoothly.

- REST, GraphQL or other?

↳ Only REST API available.

Is there a concept of how I can extend the data model of a product or other entities?

- How do I add additional fields to products, customers, carts, orders, etc.?

↳ There is a concept called “mixins”, which allows you to extend some entities with custom attributes described via a json schema.

How detailed are the error responses?

- Can the API tell me which fields are invalid, e.g. if I have an invalid payload?
- Do I get in general helpful error messages?

↳ The error messages contain the path to the problematic field and a description of the validation failure. The error messages contain a detailed description of the problem.

Is it possible to run meaningful performance tests?

- Do I have the same or a similar performance than in production?
- If there are restrictions am I still able to run a performance test?

↳ **No performance differences are specified and no performance restrictions are mentioned.**

How good is the authorization concept?

- Is the authorization concept
 - easy to understand?
 - flexible to different needs?

↳ **Initially, two API credentials are registered: Emporix API and Storefront API. It is unclear how new credentials can be created. The authorization concept differentiates between three types of tokens, anonymous, customer, and service, which control resource access via associated scopes. These required scopes are referenced in the API reference.**

Is the API well structured/intuitive?

↳ **The API reference provides a clear structure, and the expected endpoints are specified.**

Documentation

Is the documentation easy to access/ find?

- Do I have direct (or with low effort) access to the documentation?

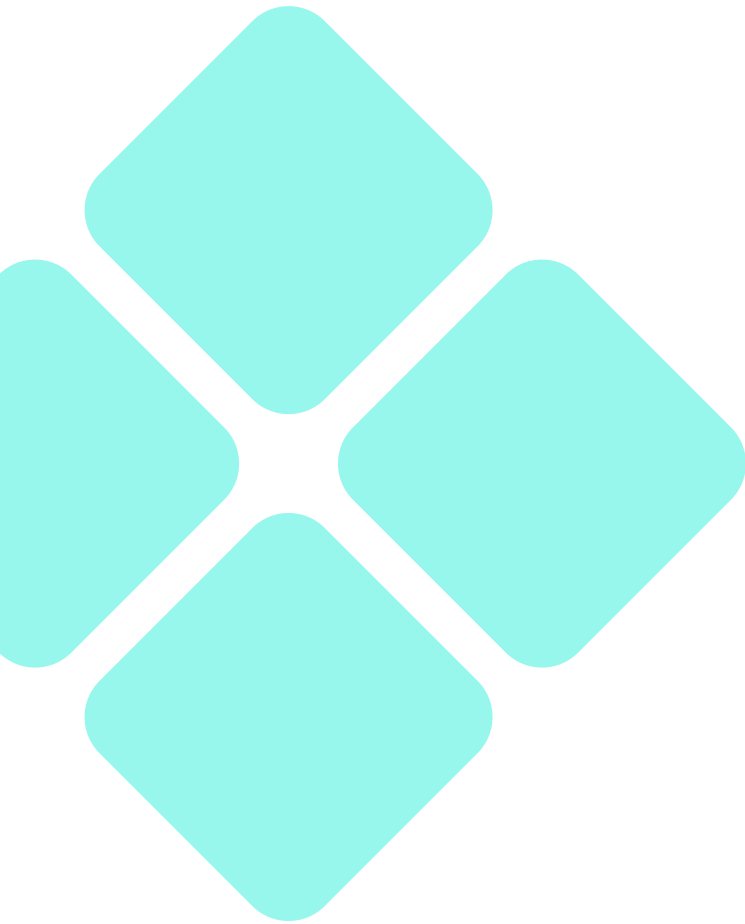
↳ **The documentation and guides are accessible without registration and are linked from the homepage. Some more cross links between related API endpoints and the remaining documentation/guides would make it easier to find API functions.**

Is it well structured?

- Are there guides to get me started?
- How do I add a product?
- How to add a product to the cart?
- Documented best/accepted practices?
- Is there a getting started guide?

↳ **Only some elements of the api model are documented, and only on a very technical level with overview tables. Accepted or best practices are not mentioned. In addition, a getting started guide is available. It is at the bare minimum, more a collection of curl calls with no additional explanations of any kind, no explanation of the API calls to be executed and**





Completeness

no links to further information. Some resources were already created by default, so that the guide recreates duplicate content

Some questions remain unanswered: Why do I need to configure delivery settings now? What consequences will this have? Why are delivery areas and times not tenant specific? Why do I need to set this up? Here the guide seems to be limited.

In addition, inconsistencies emerge, e.g. after creating and linking a product to a category the product is nowhere to be found.

- Are there meaningful payload examples in the API documentation?
- Are all possible error responses documented in the API documentation?

↳ The API reference contains examples of request and response payloads with realistic values. The complete range of possible HTTP status responses seems to be documented for each endpoint with an explanation of what the status code means.

Most of the schemas seem to be documented while some only contain type definitions without proper descriptions or example/default values.

API extensions

- How well are the extension concepts explained?
- Can I extend the data model on my own?
- Can I include external services?

↳ **The extension concept via mixins is not documented separately and only rudimentary when first used. The data model can be extended via mixins. This requires a three step process**

- 1. Define custom attributes schema via JSON schema and upload it to a hosting service,**
- 2. reference custom attributes schema via the metadata attribute,**
- 3. use custom attributes inside mixins attribute.**

A concept for using custom external services is not described. There are guides for integrating with Auth0 and Contentful.

Can I generate API calls directly from the documentation?

- Is there a try it out function in documentation or is there some kind of code (e.g. curl or Postman collections) I can copy from the documentation?

↳ In the getting started guide, there are curl calls to be copied. They are not pre-filled with already known information, eg. the tenant. The API reference does not provide a try out function but the payloads can be freely copied from the reference. There seems to be no Postman collection or openAPI document for download.

Useful Extras

Are SDKs provided for my most important languages?

Language	SDK available?
Java	no
PHP	no
JavaScript	no
DotNET Core	no

Does a template for a simplistic shop frontend exist?

- Minimum desktop? (Bonus points for non-desktop)
- Access to templates GitHub repository OSS license?

↳ **There is no template available.**

What additional tooling exists?

- Importer?

↳ **There is no additional tooling available.**

Is demo data provided?

↳ **No demo data provided.**

Is there a community?

↳ **No community available.**

Conclusion

↳ Our first impression: good. API-first is realized, documentation and useful extras could be improved.

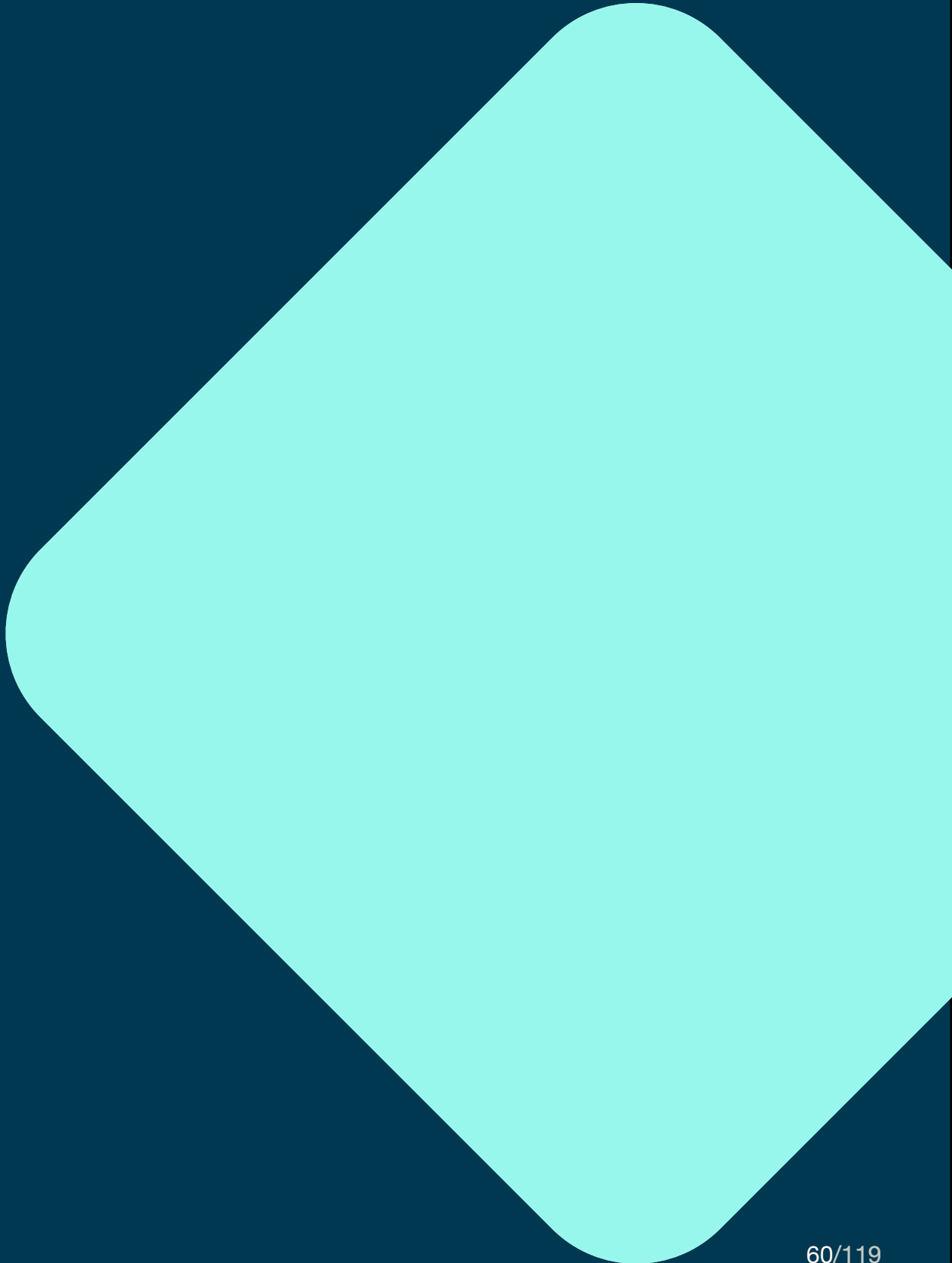
Test environment: fully meets our expectations.

APIs: overall very good, we could not test performance.

Documentation: Advanced instructions for typical problem scenarios are missing.

Useful extras: missing; Additional tools and templates for basic shops would be helpful.

06 Fabric



Introduction

Fabric was founded in 2020 “by commerce veterans to help brands migrate from legacy commerce systems. Our headless platform offers a modular suite of intuitive business applications for retail buyers, planners, and marketers.”⁷

⁷ <https://fabric.inc/about>

Test Environment

How can I retrieve test credentials?

- How difficult is it to retrieve them?
- What additional steps are needed before I can start my tests?
- Do I have to wait for approval or do I need to get a demo first?

↳ **Fabric.inc does not provide a self-signup. You are forced to request a demo first.**

How close is the test environment to the production environment?

- Am I restricted in what I can do?
- Is the performance close to the final product?
- Could I transform my testing environment into a production one?

↳ **Could not be tested without signup.**

Am I able to do the following steps without any external help and without additional setup of the environment?

1. Create a product,
2. add a price,
3. add the product to a category,
4. create a cart,
5. add the product to a cart.

↳ **Could not be tested without signup.**

APIs

What kind of API is provided?

- REST, GraphQL or other?

↳ **Only REST API available.**

Is there a concept of how I can extend the data model of a product or other entities?

- How do I add additional fields to products, customers, carts, orders, etc.?

↳ **There is no apparent concept how entities might be extended with custom attributes, also no mention of webhooks. There is only so much in the FAQs:**

“What is the process for integrating a new third-party application with Fabric? Your Fabric project team will work with you to ensure that any third party integrations and specific use cases you need are identified in the discovery stage and

are appropriately incorporated into your onboarding plan.”⁸

Integration with third parties seems to be a manual process.

How detailed are the error responses?

- Can the API tell me which fields are invalid, e.g. if I have an invalid payload?
- Do I get in general helpful error messages?

↳ **Could not be tested.**

Is it possible to run meaningful performance tests?

- Do I have the same or a similar performance than in production?
- If there are restrictions am I still able to run a performance test?

↳ **Could not be tested.**

⁸ <https://fabric.inc/faq/integrations/new-connection-process>

How good is the authorization concept?

- Is the authorization concept
 - easy to understand?
 - flexible to different needs?

↳ **The authorization concept is not explained in detail, except that an API key is provided. It remains unclear how to implement a Javascript-based application securely running inside the customers browser, or how to secure user-specific resources as opposed to more management related endpoints.**

Is the API well structured/intuitive?

↳ **The API is subdivided into multiple smaller APIs, which makes it difficult to learn how to use them.**



Documentation

Is the documentation easy to access/ find?

- Do I have direct (or with low effort) access to the documentation?
- Or is there some kind of approval needed to get access?

↳ **The documentation is accessible without prior registration. Unfortunately there are two API reference documentations which are not equal in descriptiveness.**

Is it well structured?

- Are there guides to get me started?
- How do I add a product?
- How to add a product to the cart?
- Documented best/accepted practices?
- Is there a getting started guide?

↳ **There are no dedicated guides for using the API. Also, the internal entity model is not explained. In addition, the internal entity model is not explained, nor is how a new product is created. Last but not least, the API reference categories seem to be read-only, as there is no description how to create them.**

Completeness

- Are there meaningful payload examples in the API documentation?
- Are all possible error responses documented in the API documentation?

↳ **The API documentation contains payload examples for all endpoints, but the values don't look like working examples that can be simply copied. Some of the endpoints only describe the successful response without describing error responses.**

API extensions

- How well are the extension concepts explained?
- Can I extend the data model on my own?
- Can I include external services?

↳ **There are no explanations of extension mechanisms to be found. This does not mean that they don't exist but they are not discoverable.**

Can I generate API calls directly from the documentation?

- Is there a try it out function in documentation or is there some kind of code (e.g. curl or Postman collections) I can copy from the documentation?

↳ **The API reference does not provide a try out function or curl calls to copy. Instead, an openAPI document is offered for download for each API.**

Useful Extras

Are SDKs provided for my most important languages?

Language	SDK available?
Java	N/A
PHP	N/A
JavaScript	N/A
DotNET Core	N/A

Does a template for a simplistic shop frontend exist?

- Minimum desktop? (Bonus points for non-desktop)
- Access to templates GitHub repository OSS license?

↳ **Could not be found.**

What additional tooling exists?

- Importer?

↳ Could not be found.

Is demo data provided?

↳ No demo data provided.

Is there a community?

↳ No community available.

Noteworthy

↳ **The platform does not look like it's geared to developers, as many operations need direct interaction with account managers. This is true even for simple tasks like retrieving new API tokens or configuring channel IDs.**

“Some APIs require an “access token” via the “authorization” header. The token is generated when a user is authenticated using the login local user API. In some cases you will need an “API token” that will be provided to you by your account manager. You want to make sure to keep this token safe and secure.”⁹

⁹ <https://fabric.inc/docs/developer-portal/#authentication--authorization>

There are two different API References one is based on openAPI, although it is not clear how the two are related. The primary one displays a deprecated notice and links to the openAPI one. This leads to some confusion about which of the two is to be used.

Conclusion

↳ Our first impression: immature.
Focus on enabling developers seems to be missing.

Test environment: not available or could not be tested.

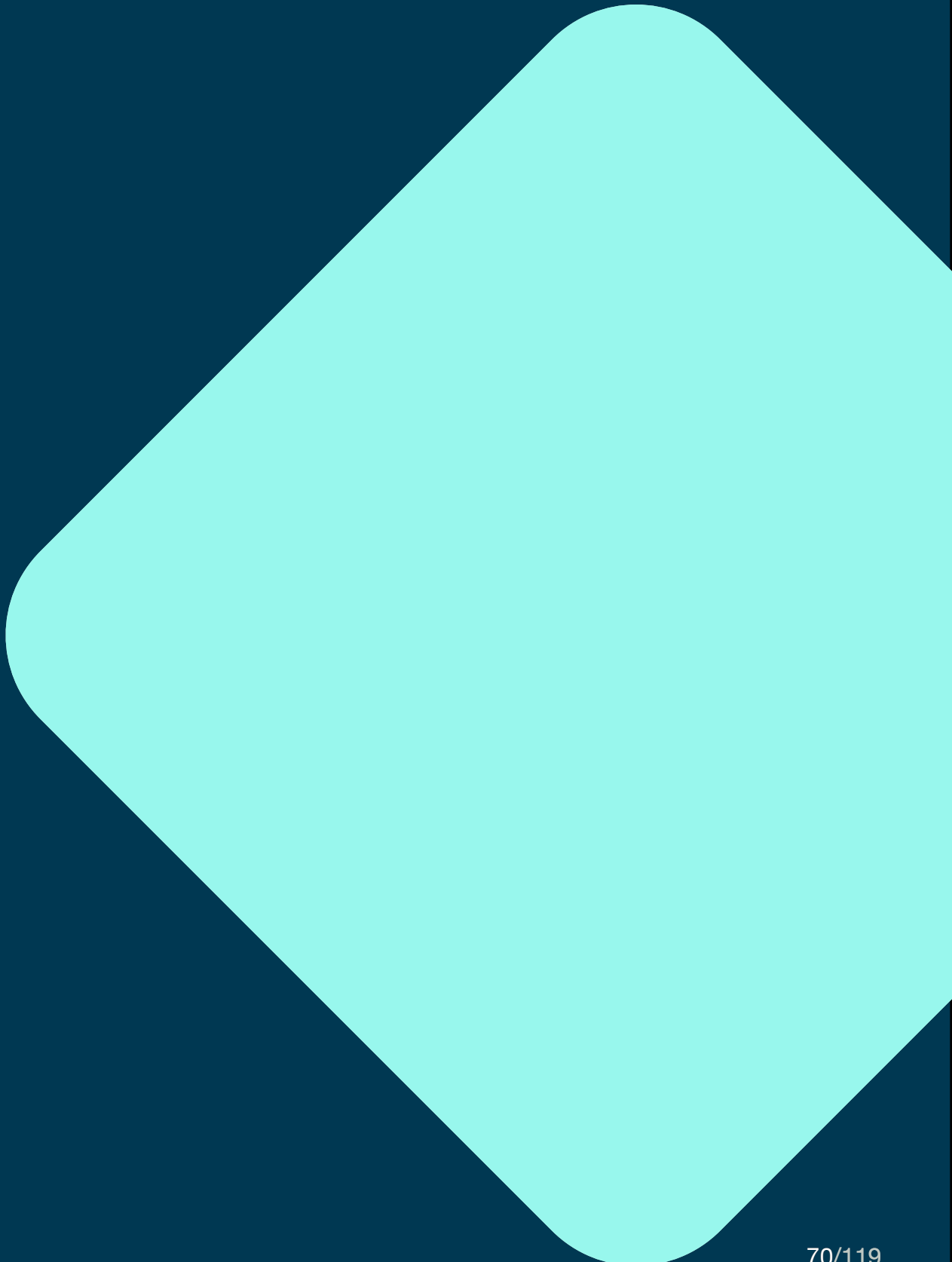
APIs: poorly explained or not testable without support.

Documentation: does not meet our expectations.

Useful extras: either not available or not documented.

07

Salesforce Commerce Cloud



Introduction

The Commerce Cloud, launched in 2016, is “an e-commerce platform by Salesforce. It includes Salesforce B2C Commerce and Salesforce B2B Commerce, which are often used in conjunction to provide a comprehensive, seamless ecommerce experience.”¹⁰

¹⁰ <https://www.salesforce.com/de/products/commerce-cloud/overview>

Test Environment

How can I retrieve test credentials?

- How difficult is it to retrieve them?
- What additional steps are needed before I can start my tests?
- Do I have to wait for approval or do I need to get a demo first?

↳ **Self-signup is not possible. Access to a test environment for the commerce cloud is not provided without first requesting a quote and possibly taking further necessary steps.**

How close is the test environment to the production environment?

- Am I restricted in what I can do?
- Is the performance close to the final product?
- Could I transform my testing environment into a production one?

↳ **Could not be tested without signup.**

Am I able to do the following steps without any external help and without additional setup of the environment?

1. Create a product,
2. add a price,
3. add the product to a category,
4. create a cart,
5. add the product to a cart.

↳ **Could not be tested without signup.**

APIs

What kind of API is provided?

- REST, GraphQL or other?

↳ **Only RESTful API available.**

Is there a concept of how I can extend the data model of a product or other entities?

- How do I add additional fields to products, customers, carts, orders, etc.?

↳ **It is not clear from the documentation alone how custom attributes or external services can be incorporated.**



How detailed are the error responses?

- Can the API tell me which fields are invalid, e.g. if I have an invalid payload?
- Do I get in general helpful error messages?

↳ **The error responses contain human-readable error descriptions as well as descriptions of the invalid fields, if applicable. In general, additional information is provided to help fix the problems.**

Is it possible to run meaningful performance tests?

- Do I have the same or a similar performance than in production?
- If there are restrictions am I still able to run a performance test?

↳ **Could not be tested.**

How good is the authorization concept?

- Is the authorization concept
 - easy to understand?
 - flexible to different needs?

↳ **The authorization concept distinguishes between shopper APIs and other APIs. While shopper APIs are secured via OAuth, other APIs are secured via clientId and secrets. Therefore, the other APIs cannot be used directly from the user's browser.**

Is the API well structured/intuitive?

↳ Creation of new resources is handled via PUT operations for the object's resource to be created, which is quite surprising and differs from other API practices. But other than that, the API seems to be appropriate.

Documentation

Is the documentation easy to access/ find?

- Do I have direct (or with low effort) access to the documentation?
- Or is there some kind of approval needed to get access?

↳ The documentation is spread over multiple different locations that appear to contain duplicates. It is also not always trivial to identify if the documentation refers to the B2C Commerce API or to one of the other Salesforce products. Most of the documentation points to Salesforce's training platform "Trailhead", which itself requires registration to get more information.



Is it well structured?

- Are there guides to get me started?
- How do I add a product?
- How to add a product to the cart?
- Documented best/accepted practices?
- Is there a getting started guide?

↳ **There is a collection of guides but none of them addresses API usage directly. The guides explain the context of the API rather than the API itself. Therefore, there is no guide explaining how to execute the first API calls or how to add the first products. There is also no documentation of best practices.**

Completeness

- Are there meaningful payload examples in the API documentation?
- Are all possible error responses documented in the API documentation?

↳ **The API reference does contain payloads with real world examples and also specifies the possible error responses per endpoint including matching response bodies.**

API extensions

- How well are the extension concepts explained?
- Can I extend the data model on my own?
- Can I include external services?

↳ **No explanation of API extensions could be found.**

Can I generate API calls directly from the documentation?

- Is there a try it out function in documentation or is there some kind of code (e.g. curl or Postman collections) I can copy from the documentation?

↳ **The API reference does not provide a try out function but provides code examples for CURL, JavaScript, Python, C, Java and even different frameworks for each language.**

Useful Extras

Are SDKs provided for my most important languages?

Language	SDK available?
Java	no
PHP	no
JavaScript	yes
DotNET Core	no

Does a template for a simplistic shop frontend exist?

- Minimum desktop? (Bonus points for non-desktop)
- Access to templates GitHub repository OSS license?

↳ **A starter template for a React based PWA is provided under BSD 3-Clause license.**

What additional tooling exists?

- Importer?

↳ **There are no additional tools available.**

Is demo data provided?

↳ **No demo data is provided.**

Is there a community?

↳ **Salesforce offers a dedicated Commerce Cloud discussion platform within the developer center, and the training platform Trailhead contains community features to discuss with other participants.**

Noteworthy

↳ **Nothing to add to the above.**

Conclusion

↳ Our first impression: ok. We expected more from a well-known brand.

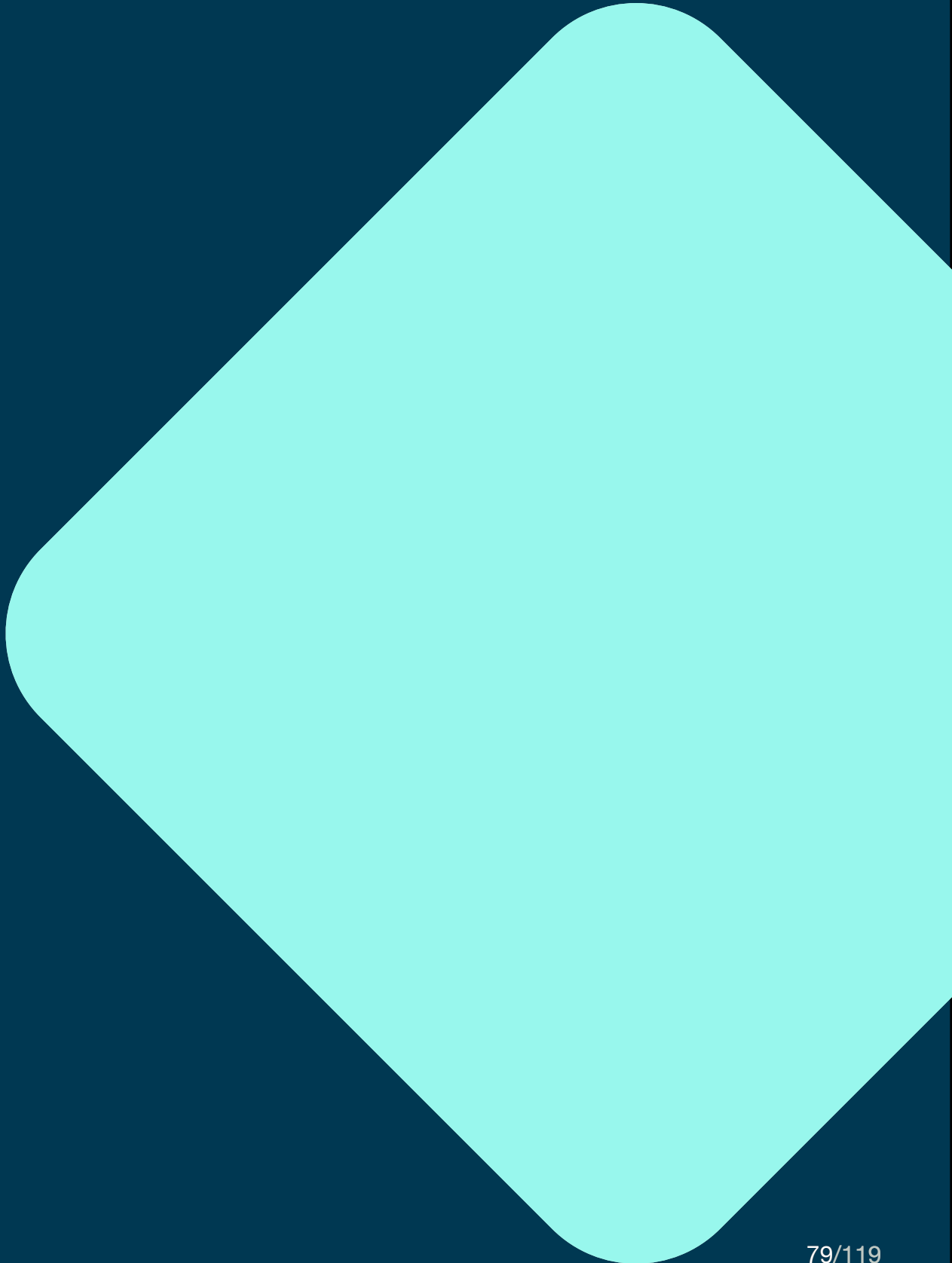
Test environment: not available, or could not be tested.

APIs: overall ok, performance and data model extension could not be tested.

Documentation: comprehensive, but poorly structured, no API usage instructions.

Useful extras: mediocre, no extras worth mentioning.

08 Shopify



Introduction

Shopify, founded in 2006, is “a subscription-based software that allows anyone to set up an online store and sell their products. Shopify store owners can also sell in physical locations using Shopify POS, our point-of-sale app and accompanying hardware.”¹¹

¹¹ <https://www.shopify.com/blog/what-is-shopify>

Test Environment

How can I retrieve test credentials?

- How difficult is it to retrieve them?
- What additional steps are needed before I can start my tests?
- Do I have to wait for approval or do I need to get a demo first?

↳ **Self-signup via email is possible, no approval needed. The trial duration is 14 days, which feels too short for a proper evaluation.**

How close is the test environment to the production environment?

- Am I restricted in what I can do?
- Is the performance close to the final product?
- Could I transform my testing environment into a production one?

↳ **The environment seems to be production-ready, as no performance restrictions are explicitly mentioned.**

Am I able to do the following steps without any external help and without additional setup of the environment?

1. Create a product,
2. add a price,
3. add the product to a category,
4. create a cart,
5. add the product to a cart.

↳ The existence of an API is not announced during signup or in the admin interface. To use the API, the “private apps” function must be explicitly activated, only then is it possible to access the APIs. However, this does not seem to be the intended use case for apps, as apps should extend the shop's functionality and always refer to a partner account.

It was not possible to create a product and add it to a category via API, as the admin API is only meant for implementing apps which were out of scope.



APIs

What kind of API is provided?

- REST, GraphQL or other?

↳ The storefront API is provided only as a GraphQL API, while the admin API is provided as REST and GraphQL API.

Is there a concept of how I can extend the data model of a product or other entities?

- How do I add additional fields to products, customers, carts, orders, etc.?

↳ **Additional fields are supported via a concept called “metafields” which can be added to some of the resources including products, orders and customers. Not all resources can be extended via metafields. There are public and private metafields, where the private metafields can be accessed only by the app that created the field. Apps can also register webhooks to subscribe to specific events.**

How detailed are the error responses?

- Can the API tell me which fields are invalid, e.g. if I have an invalid payload?
- Do I get in general helpful error messages?

↳ **The error responses contain human-readable error messages that explain the problem. It is not clear from the API reference alone if the error responses would include invalid fields.**

Is it possible to run meaningful performance tests?

- Do I have the same or a similar performance than in production?
- If there are restrictions am I still able to run a performance test?

↳ **No performance restrictions are mentioned.**

How good is the authorization concept?

- Is the authorization concept
 - easy to understand?
 - flexible to different needs?

↳ **The authorization concept for the admin API is geared towards public apps installed in the Shopify shop. For integrating a custom external service, the authorization seems to be possible only via basic HTTP authentication, which is called "private app" in shopify speak.**

The storefront API is described as unauthenticated, while it is necessary to create Storefront API access tokens for private and public apps. It is not clear why apps should use the Storefront API and not the admin API.

The usage patterns for admin API and Storefront API in combination with apps are unclear.

Is the API well structured/intuitive?

↳ The structure of the API meets our expectations.

Documentation

Is the documentation easy to access/ find?

- Do I have direct (or with low effort) access to the documentation?
- Or is there some kind of approval needed to get access?

↳ The documentation is accessible without prior registration or approval. However, the existence of API documentation is not announced on the homepage, and you have to google for the URL of the developer portal.



Is it well structured?

- Are there guides to get me started?
- How do I add a product?
- How to add a product to the cart?
- Documented best/accepted practices?
- Is there a getting started guide?

↳ **The internal structure of the documentation is confusing and not intuitive at all. The menu structure is not truly hierarchical so that menu entries lead to entirely different documentation trees instead of the expected subsections of the current documentation topic.**

There are getting started guides for all APIs but they only cover the bare minimum up to the first request. The guides also mix REST and GraphQL examples without the possibility to choose the preferred API variant.

There are other guides, but they differ greatly in quality. Some provide detailed descriptions of how to achieve certain goals and others contain mere collections of API call examples without any additional explanation. Furthermore, the examples only target the GraphQL admin API while there don't seem to be any examples for the REST API.



Completeness

- Are there meaningful payload examples in the API documentation?
- Are all possible error responses documented in the API documentation?

↳ **The REST API reference contains realistic payload examples. For some endpoints there are multiple examples for different use cases. The error responses are described only once in an overview and are not specified per endpoint. The GraphQL admin API also contains payload examples for most of the possible operations.**

It seems that the REST and the GraphQL admin API are not functionally identical, with the GraphQL API being the more complete one.

API extensions

- How well are the extension concepts explained?
- Can I extend the data model on my own?
- Can I include external services?

↳ **Both the creation of custom attributes and the registration of new webhooks are described in great detail in specific guides.**

Can I generate API calls directly from the documentation?

- Is there a try it out function in documentation or is there some kind of code (e.g. curl or Postman collections) I can copy from the documentation?

↳ It is not possible to execute API calls from the documentation, but the reference contains example code for cURL, Node.js and PHP.

Useful Extras

Are SDKs provided for my most important languages?

Language	SDK available?
Java	no
PHP	yes
JavaScript	yes
DotNET Core	no

↳ There are additional client libraries available for Python and Ruby, also Storefront client libraries for Android, iOS and Unity.

Does a template for a simplistic shop frontend exist?

- Minimum desktop? (Bonus points for non-desktop)
- Access to templates GitHub repository OSS license?

↳ **Shopify offers "Hydrogen" for this purpose, its own frontend web development framework.**

What additional tooling exists?

- Importer?

↳ **Shopify offers a lot of additional tools.**

Is demo data provided?

↳ **No demo data is provided.**

Is there a community?

↳ **Shopify maintains active communities on Slack, YouTube, Twitch, and its own developer forums.**

Noteworthy

↳ **You are expected to create a Shopify partner account before using the APIs. It seems the API is not geared towards inhouse development. The distinction between "installing an app" and "using the API" is not clear.**

Conclusion

↳ Our first impression: ok. We expected more from a well-known brand.

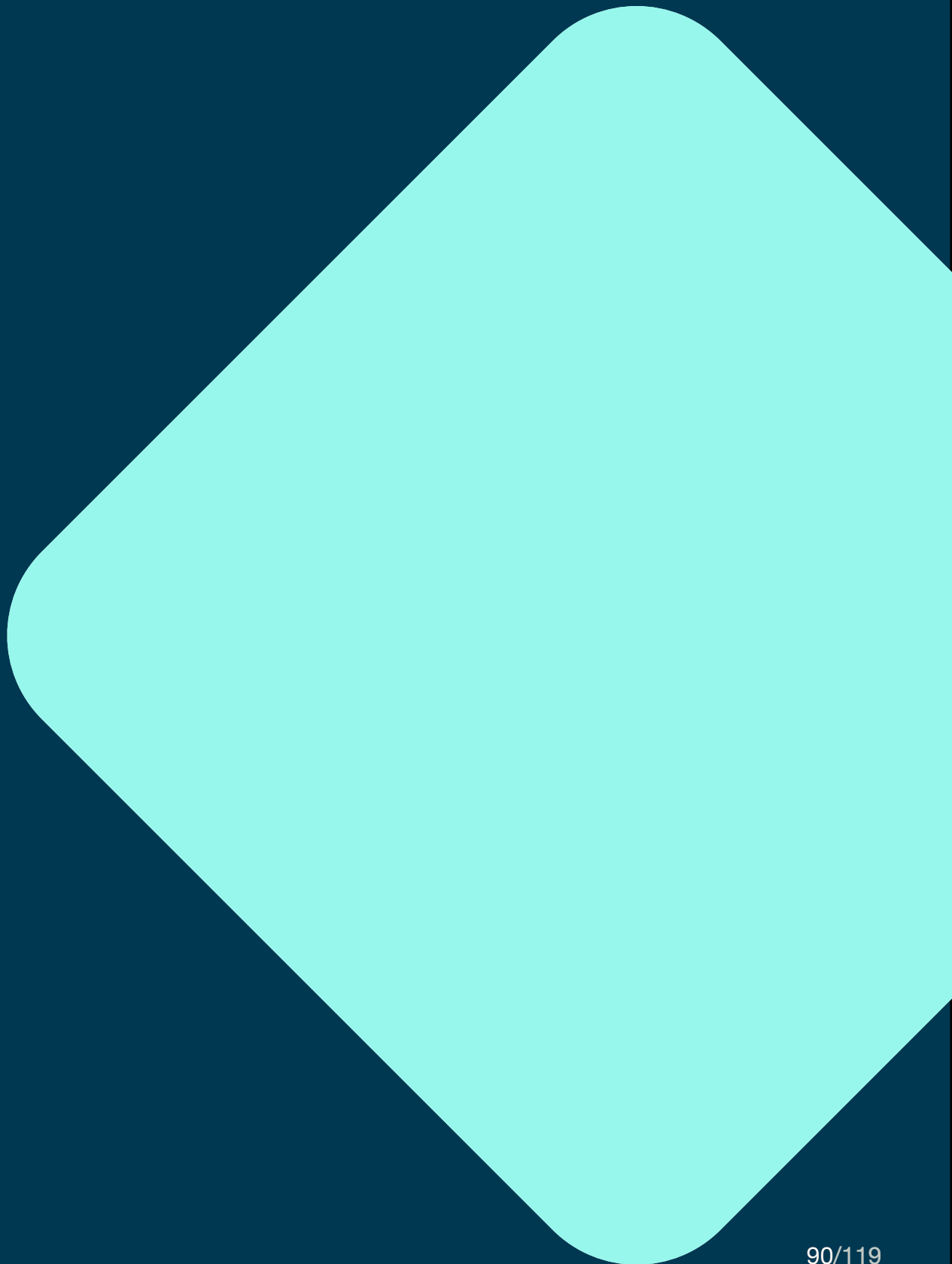
Test environment: ok, no content creation without setting up the test environment.

APIs: API usage requires Shopify partner account, distinction between API and apps often unclear.

Documentation: not easy to access, poorly structured.

Useful extras: many useful extras, except demo data.

09 Spryker



Introduction

Spryker, founded in 2014, claims to provide "the most modern platform-as-a-service solution with over 900 API-based modules, cloud-enabled, enterprise-ready, and loved by developers and business users worldwide."¹²

¹² <https://www.linkedin.com/company/spryker-systems-gmbh>

Test Environment

How can I retrieve test credentials?

- How difficult is it to retrieve them?
- What additional steps are needed before I can start my tests?
- Do I have to wait for approval or do I need to get a demo first?

↳ **Self-signup is not possible. A demo must be attended to start the evaluation.**

How close is the test environment to the production environment?

- Am I restricted in what I can do?
- Is the performance close to the final product?
- Could I transform my testing environment into a production one?

↳ **This is not specified. Two environments are created during setup: staging and production.**

Am I able to do the following steps without any external help and without additional setup of the environment?

1. Create a product,
2. add a price,
3. add the product to a category,
4. create a cart,
5. add the product to a cart.

↳ **Could not be tested without self-signup.**

APIs

What kind of API is provided?

- REST, GraphQL or other?

↳ **Only REST API available.**

Is there a concept of how I can extend the data model of a product or other entities?

- How do I add additional fields to products, customers, carts, orders, etc.?

↳ **There are two different concepts. For products, custom attributes can be created via the back office. The second concept for extending a REST API resource also includes extending the database schema via migrations.**

How detailed are the error responses?

- Can the API tell me which fields are invalid, e.g. if I have an invalid payload?
- Do I get in general helpful error messages?

↳ **This remains unclear, as the API reference does not include real life example values.**

Is it possible to run meaningful performance tests?

- Do I have the same or a similar performance than in production?
- If there are restrictions am I still able to run a performance test?

↳ **Could not be tested.**

How good is the authorization concept?

- Is the authorization concept
 - easy to understand?
 - flexible to different needs?

↳ **The concept looks reasonable and allows for different access roles.**

Is the API well structured/intuitive?

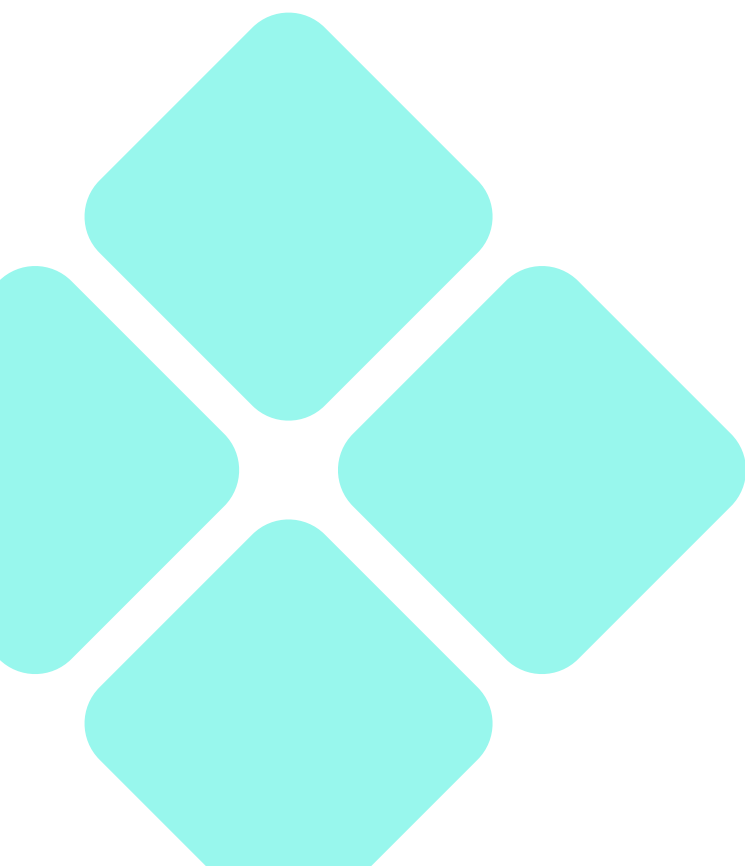
↳ The initial API provided looks good, but since it appears to be fully modifiable, maintaining the structure is up to the developers.

Documentation

Is the documentation easy to access/ find?

- Do I have direct (or with low effort) access to the documentation?
- Or is there some kind of approval needed to get access?

↳ There is extensive documentation, but most of it is geared towards the back office usage and not API development. The documentation is accessible without any registration.



Is it well structured?

- Are there guides to get me started?
- How do I add a product?
- How to add a product to the cart?
- Documented best/accepted practices?
- Is there a getting started guide?

↳ **Instructions for getting started are not easy to find. However, there are guides for more advanced topics. Unfortunately, these guides are not easy to navigate, as all information is located in one large documentation hub. There is no learning path apparent which would lead to a deeper understanding of the platform.**

Completeness

- Are there meaningful payload examples in the API documentation?
- Are all possible error responses documented in the API documentation?

↳ **The API reference does not contain real world example payloads. The payloads are included only in one representation of the reference, while the other one does not contain payloads and is structured differently. The question why there are two different references remains unanswered.**

API extensions

- How well are the extension concepts explained?
- Can I extend the data model on my own?
- Can I include external services?

↳ There is a guide explaining how to extend the REST API with new resources and how to add new columns to the underlying data store. Unfortunately, there are no guides that explain how external services can be integrated.

Can I generate API calls directly from the documentation?

- Is there a try it out function in documentation or is there some kind of code (e.g. curl or Postman collections) I can copy from the documentation?

↳ There is no try out function, curl requests nor Postman collection. However, an openAPI document is provided.



Useful Extras

Are SDKs provided for my most important languages?

Language	SDK available?
Java	no
PHP	no
JavaScript	no
DotNET Core	no

Does a template for a simplistic shop frontend exist?

- Minimum desktop? (Bonus points for non-desktop)
- Access to templates GitHub repository OSS license?

↳ There are demo shops for B2C and B2B and a B2B marketplace at GitHub.

What additional tooling exists?

- Importer?

↳ Spryker provides a CSV importer and an architecture validation tool, furthermore a setup for VM and Docker based development.

Is demo data provided?

↳ There is no demo data provided.

Is there a community?

↳ A comprehensive list of repositories for Spryker can be found on Github, Stackoverflow hosts only six spryker discussions.

Noteworthy

↳ The Glue API, the headless API on top of the Spryker platform, is not the central access point for Spryker, but seems bolted on.

The deployment/provisioning model of Spryker seems to be different from the other platforms, as it is not a headless commerce API in the same sense as the others.

Spryker provides multiple certifications and training through its Spryker Academy.

Conclusion

↳ Our first impression: poor. Leaves us confused about what the platform offers (to developers).

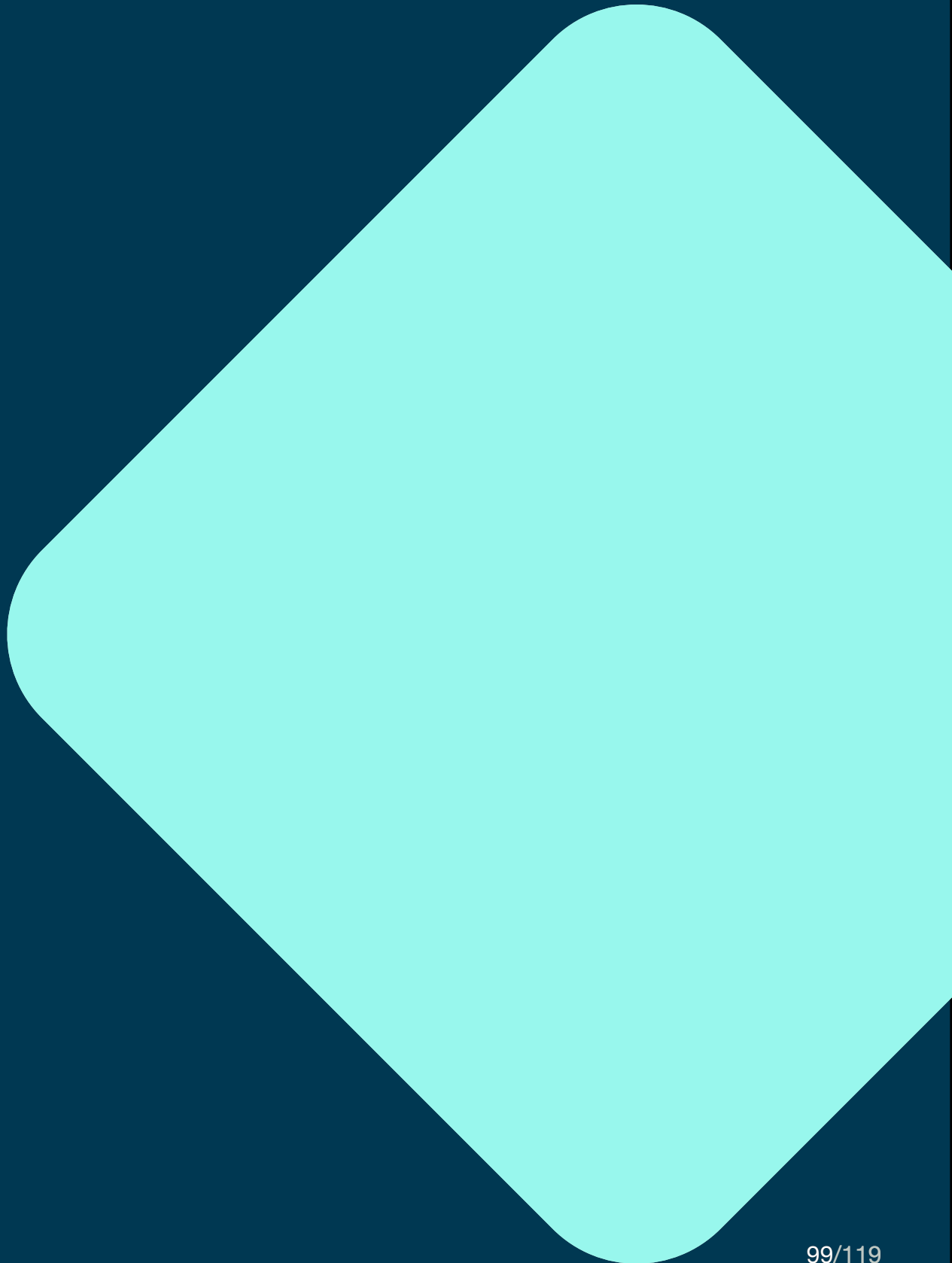
Test environment: much room for improvement, not designed for DIY.

APIs: overall meets our expectations.

Documentation: overloaded, poorly structured, use of APIs remains unclear.

Useful extras: we would have expected more.

10 Swell



Introduction

Swell, founded in 2021, provides “a customizable headless ecommerce platform that supports unique business models and customer experiences.”¹³

¹³ <https://www.linkedin.com/company/swellcommerce>

Test Environment

How can I retrieve test credentials?

- How difficult is it to retrieve them?
- What additional steps are needed before I can start my tests?
- Do I have to wait for approval or do I need to get a demo first?

↳ **Self-signup is possible via email. There is no approval nor demo required to access the api other than standard email verification. Duration of the test period is only 14 days, which feels too short for a proper evaluation.**

During the signup, in addition to the option to create a new store, various platforms such as Shopify, WooCommerce, Magento, BigCommerce, or custom ones, can be migrated. Selecting a theme for the shop storefront is also part of the signup, but can be skipped.

How close is the test environment to the production environment?

- Am I restricted in what I can do?
- Is the performance close to the final product?
- Could I transform my testing environment into a production one?

↳ **Could not be tested.**

Am I able to do the following steps without any external help and without additional setup of the environment?

1. Create a product,
2. add a price,
3. add the product to a category,
4. create a cart,
5. add the product to a cart.

↳ **The guides don't explain how to subsequently add a product to a category via API. Experiments based on information from the api reference to solve this problem were not successful.**



APIs

What kind of API is provided?

- REST, GraphQL or other?

↳ **Two different APIs are provided, one fully functional REST Backend API and a more restricted Frontend API. The Frontend API is session-based and allows access to some entities restricted only by the customer account. The Frontend API is documented only through the `swell.js` framework. However, it is unclear if it is supported by a REST API.**

Is there a concept of how I can extend the data model of a product or other entities?

- How do I add additional fields to products, customers, carts, orders, etc.?

↳ **Entities can be extended via content and data models. Furthermore, fully custom entities are also possible. Unfortunately, entities can also be unintentionally extended with erroneous attributes, as there is no validation on newly added attributes, as the following category representation shows.**

In addition to custom fields, external services can be integrated via webhooks configured from the Swell Dashboard.



How detailed are the error responses?

- Can the API tell me which fields are invalid, e.g. if I have an invalid payload?
- Do I get in general helpful error messages?

↳ **The error messages contain a list of missing attributes or attributes with validation errors. When trying to update a non-existent entity, the response returned validation errors instead of indicating that the entity in question does not exist.**

Is it possible to run meaningful performance tests?

- Do I have the same or a similar performance than in production?
- If there are restrictions am I still able to run a performance test?

↳ **Could not be tested.**

How good is the authorization concept?

- Is the authorization concept
 - easy to understand?
 - flexible to different needs?

↳ **The authorization concept meets our expectations.**

Is the API well structured/intuitive?

↳ The API's structure meets our expectations.

Documentation

Is the documentation easy to access/ find?

- Do I have direct (or with low effort) access to the documentation?
- Or is there some kind of approval needed to get access?

↳ The documentation is available without prior registration.

Is it well structured?

- Are there guides to get me started?
- How do I add a product?
- How to add a product to the cart?
- Documented best/accepted practices?
- Is there a getting started guide?

↳ There are guides that describe the platform as a whole, but they don't address how to use the API itself. The guides explain what entities exist and how they should be used, but don't help with running the initial requests to the APIs. There is even a guide series "How to build a marketplace", although it isn't even clear how to build a shop.

The backend API reference provides descriptions of the models, but no description of a meaningful workflow. In addition, there are guides for migration in general and migration from Shopify describing the batch import backend API functionality.

The API reference is organized as a long list of entities with relevant endpoints, making it difficult to find the relevant information.

Completeness

- Are there meaningful payload examples in the API documentation?
- Are all possible error responses documented in the API documentation?

↳ Swell provides a localization concept for product entities and others. There are also payload examples for curl, Node and PHP clients, but not every documented interaction provides usage examples for every language.

Error responses are documented only on a general level, but not in detail for every resource. Therefore, there is no detailed explanation which error codes could be caused by misuse of the API.



API extensions

- How well are the extension concepts explained?
- Can I extend the data model on my own?
- Can I include external services?

↳ **There are content and data models to extend the entities. For data models, it is a low-level extension concept that does not provide automatic support. Content models form a layer above data models.**

It is even possible to define new, fully custom entities to be used inside the commerce platform. Swell also supports Webhooks to be executed when certain events occur. plain how external services can be integrated.

Can I generate API calls directly from the documentation?

- Is there a try it out function in documentation or is there some kind of code (e.g. curl or Postman collections) I can copy from the documentation?

↳ **There is no try out function, but code examples for curl, Node and PHP for the Backend API. Swell. js calls are provided for the Frontend API where applicable. The documentation does not contain payload examples for multiple use cases. Mostly only one use case is presented.**

Useful Extras

Are SDKs provided for my most important languages?

Language	SDK available?
Java	no
PHP	yes
JavaScript	yes
DotNET Core	no

Does a template for a simplistic shop frontend exist?

- Minimum desktop? (bonus points for non-desktop)
- Access to templates GitHub repository OSS license?

↳ Swell offers multiple storefront themes to be adjusted to specific requirements.

What additional tooling exists?

- Importer?

↳ For Shopify there is a migration import project that copies the most important entities.

Is demo data provided?

↳ No demo data is provided.

Is there a community?

↳ There is a discussion board on the company's website. Furthermore, some projects exist on GitHub, mostly theme repositories and SDKs.

Noteworthy

↳ Some parts of the documentation seem incomplete or unpolished, eg. there are missing sentences, broken links, and duplicate examples.

Many descriptions remain unclear, including this example:

Use the where parameter to filter results by their values. Swell offers native support for most MongoDB Query Operators including comparison operators like \$gt and \$lt—logical operators like \$and and \$or—and many others. You can combine many fields in one query.

Does this mean that the operators are passed through directly to the MongoDB?

Very positive to note that Swell provides a fully functional Checkout process hosted by Swell itself, in addition, there are useful hosting storefronts within the Swell infrastructure.

Conclusion

↳ **Our first impression: ok. Doesn't go overboard with supporting developers.**

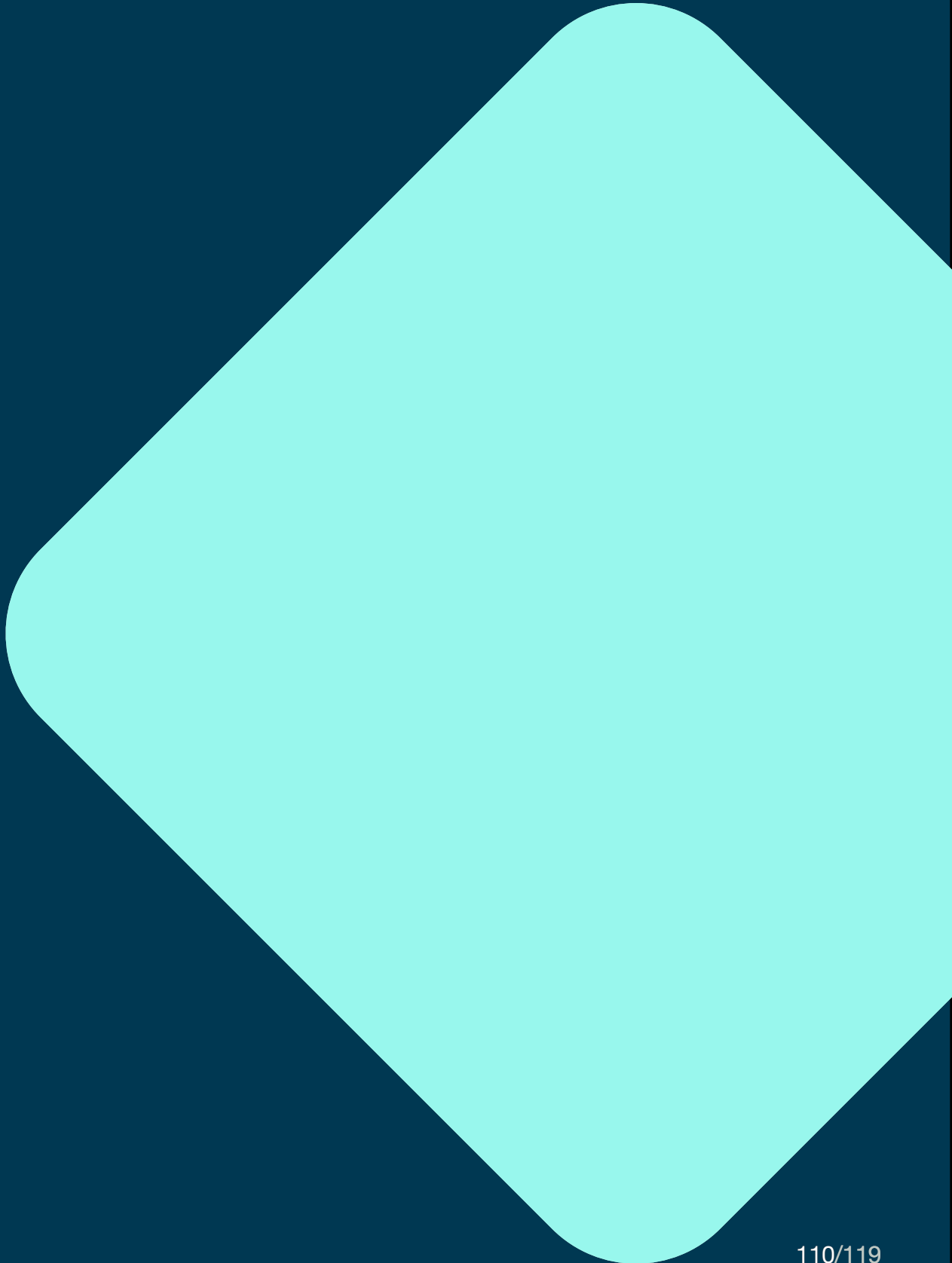
Test environment: Set up and use could be easier.

APIs: Overall meet our requirements, clear distinction between backend and frontend APIs.

Documentation: limited, structure of API reference should be better explained.

Useful extras: offers some useful extras, except demo data.

11 VTEX



Introduction

VTEX, founded in 2020, provides a “Enterprise Digital Commerce Platform that enables brands and retailers to achieve faster time to market, reach their customers across any channel, and uncover new growth areas.”¹⁴

¹⁴ <https://vtex.com/us-en/about-us>

Test Environment

How can I retrieve test credentials?

- How difficult is it to retrieve them?
- What additional steps are needed before I can start my tests?
- Do I have to wait for approval or do I need to get a demo first?

↳ **Self-signup is not possible. Vtex does not provide a test environment.**

How close is the test environment to the production environment?

- Am I restricted in what I can do?
- Is the performance close to the final product?
- Could I transform my testing environment into a production one?

↳ **No test environment available.**

Am I able to do the following steps without any external help and without additional setup of the environment?

1. Create a product,
2. add a price,
3. add the product to a category,
4. create a cart,
5. add the product to a cart.

↳ **Could not be tested.**

APIs

What kind of API is provided?

- REST, GraphQL or other?

↳ **There is only the REST API.**

Is there a concept of how I can extend the data model of a product or other entities?

- How do I add additional fields to products, customers, carts, orders, etc.?

↳ **It seems like only the cart can be extended with custom attributes. No way to extend the data model for other resources could be found. Moreover, only the status changes of the order can be used as triggers for webhooks.**

How detailed are the error responses?

- Can the API tell me which fields are invalid, e.g. if I have an invalid payload?
- Do I get in general helpful error messages?

↳ **The API reference does not describe any error responses at all.**

Is it possible to run meaningful performance tests?

- Do I have the same or a similar performance than in production?
- If there are restrictions am I still able to run a performance test?

↳ **Could not be tested.**

How good is the authorization concept?

- Is the authorization concept
 - easy to understand?
 - flexible to different needs?

↳ **The authorization concept uses API keys to secure the API, which prohibits the direct usage from the customer's browser.**

Is the API well structured/intuitive?

↳ **The API fully meets our expectations.**

Documentation

Is the documentation easy to access/ find?

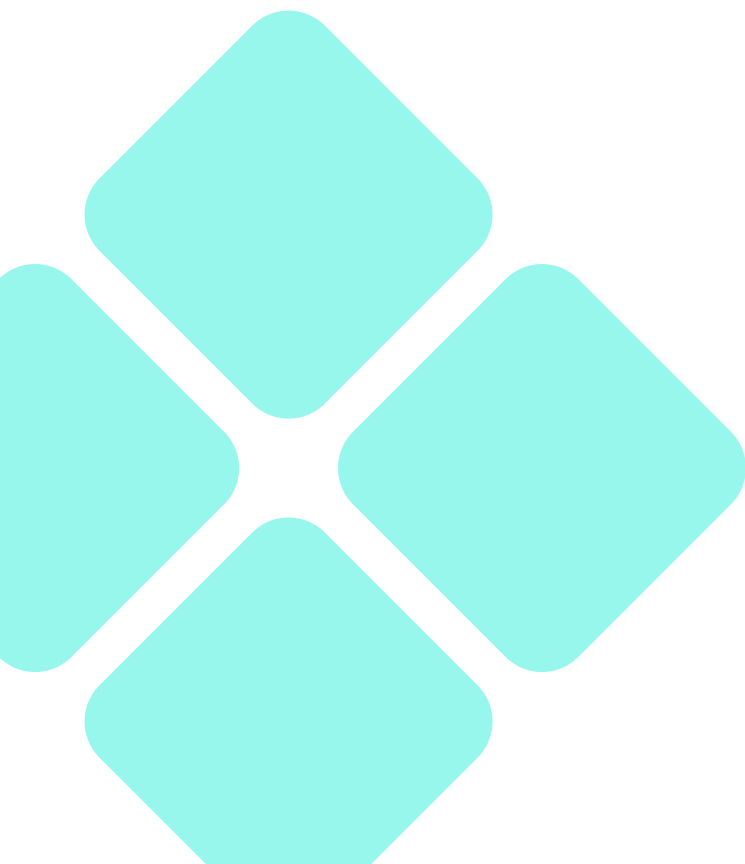
- Do I have direct (or with low effort) access to the documentation?
- Or is there some kind of approval needed to get access?

↳ **The documentation is accessible from the homepage without prior registration or approval.**

Is it well structured?

- Are there guides to get me started?
- How do I add a product?
- How to add a product to the cart?
- Documented best/accepted practices?
- Is there a getting started guide?

↳ **There are guides covering the first steps with the API. This includes adding a new product and how to place an order. Also more advanced topics such as how to manage large amounts of coupons are covered. Additionally there are guides for the integration with external systems like PIM, ERP, or WMS.**



Completeness

- Are there meaningful payload examples in the API documentation?
- Are all possible error responses documented in the API documentation?

↳ **The API reference contains examples for requests and responses with realistic values, but the reference does not specify any error responses.**

API extensions

- How well are the extension concepts explained?
- Can I extend the data model on my own?
- Can I include external services?

↳ **Only very specific extensions are possible, and these are not well explained or explanations were hard to find. The cart can be extended with custom attributes while only the order status changes can trigger webhooks.**

Can I generate API calls directly from the documentation?

- Is there a try it out function in documentation or is there some kind of code (e.g. curl or Postman collections) I can copy from the documentation?

↳ **The API reference provides a try out function and code examples for cURL, Node, Ruby, PHP and Python.**

Useful Extras

Are SDKs provided for my most important languages?

Language	SDK available?
Java	no
PHP	no
JavaScript	no
DotNET Core	no

Does a template for a simplistic shop frontend exist?

- Minimum desktop? (Bonus points for non-desktop)
- Access to templates GitHub repository OSS license?

↳ **A basic theme for a storefront is provided via GitHub repositories. There are also multiple basic components or vtex-apps that can be reused.**

What additional tooling exists?

- Importer?

↳ **There is a CLI that supports local development of VTEX IO apps.**

Is demo data provided?

↳ **No demo data provided.**

Is there a community?

↳ VTEX has its own community platform.

Noteworthy

↳ Integration guides describe a process to integrate with various neighboring systems (ERP, PIM, WMS).

Conclusion

↳ Our first impression: mediocre.

Test environment: not available or could not be tested.

APIs: meets our requirements only to a limited extent.

Documentation: good, API extensions could be better explained.

Useful extras: ok, could offer more for developers.

Methodology

Our analysis primarily focuses on:

- An easy access to test environments via self-service for developers.
- Simple testing of the very first steps, when starting a new shop for an established merchant.
- An easy to understand API reference as well as accompanied usage instructions.
- An overall helpful documentation describing more than just “how to do it”, but also “why to do it”.
- Additional tools to help you get started on your first day.

What we haven't considered is:

- The range of functions
- Prices
- Long-term sustainability
- Performance

The boundary conditions of the study were:

- Evaluation was done by an experienced developer.
- There was no previous knowledge of the platforms.
- Each platform was analyzed for half a day up to one day.

eCube designs and implements enterprise e-commerce systems for large and medium-sized businesses. It advises retailers, agencies and providers on technology issues and supports them in complex software projects by providing experience, expertise and specialists for a wide range of project roles.

Together with its technology partners, eCube has been setting standards for IT and e-commerce solutions for over 20 years, ensuring the highest performance, sustainability and added value.