Générer des clients

Un contrat OpenAPI est lisible par les machines. Elles peuvent donc écrire à votre place le code qui appelle l'API : c'est ce qu'on appelle un **client** ou un **SDK**.

Avantages :

- plus besoin d'écrire les URL, les headers et la sérialisation à la main ;
- l'autocomplétion de l'éditeur connaît toutes les routes et tous les champs ;
- quand le contrat change, on regénère, et les erreurs apparaissent dans l'éditeur plutôt qu'en production.

En Python : openapi-python-client

```
pipx run openapi-python-client generate --path api-ritif.openapi.yaml
```

L'outil génère un package Python complet, avec un module par opération (nommé d'après l'`operationId`) :

```
from api_ritif_client import Client
from api_ritif_client.api.cocktails import get_cocktail

client = Client(base_url="http://api-ritif.portlatch.com/v1")
cocktail = get_cocktail.sync(client=client, cocktail_id=3)
print(cocktail.name)
```

> Les noms exacts des modules dépendent du titre du contrat, de ses tags et de ses `operationId`. Consultez le `README.md` généré.

Pour tous les langages : openapi-generator

openapi-generator sait produire des clients (et des squelettes de serveurs) dans plus de 50 langages : Java, C#, Go, PHP, Kotlin, Swift… Il nécessite Java.

```
npx @openapitools/openapi-generator-cli generate \
  -i api-ritif.openapi.yaml -g php -o ./client-php
```

Les limites

- Le code généré n'est jamais meilleur que le contrat : un contrat approximatif donne un client approximatif.
- On ne modifie **jamais** le code généré à la main : il sera écrasé à la prochaine génération.
- Un contrat qui oublie de déclarer une erreur donne un client qui ne sait pas la gérer.

> Les grandes API publiques (Stripe, GitHub, OpenAI…) publient leur contrat OpenAPI et génèrent ainsi une bonne partie de leurs SDK officiels.

Revision #4
Created 2 October 2026 20:00:52 by Thibaud FRICHET
Updated 13 October 2026 14:52:48 by Thibaud FRICHET