

Tahmin edilebilir bir API'nin 7 katmanı

İstemcinin şaşırmadığı bir API, yedi küçük kararın tutarlılığıdır.

Muhammet Şafak • 3 Ekim 2026



Dört durak

- 01 **Kaynak ve durum kodu** Katman 1 ve 2
- 02 **Liste uçları** Katman 3: sayfalama, filtre, sıralama
- 03 **Yanıt ve hata** Katman 4 ve 5
- 04 **Sürüm ve sözleşme** Katman 6 ve 7



Her katman bir tutarlılık kararı

Kaynak

URL isim tabanlı ve çoğul, eylemi HTTP yöntemi taşır.

Liste

page, per_page, sort, order her uçta aynı adla.

Hata

Hata error.code, error.message, error.details olarak döner.

OpenAPI

Sözleşme koddan önce yazılır.

Durum

Başarı 2xx, istemci hatası 4xx, sunucu hatası 5xx.

Zarf

Her yanıt aynı iskelette: success, data, message, errors.

Sürüm

Breaking değişiklik yeni sürüm ister.

Eylemi URL değil, yöntem taşır

YÖNTEM	ANLAMI	ÖRNEK
GET	Oku	GET /users/42
POST	Oluştur	POST /users
PUT	Tamamen güncelle	PUT /users/42
PATCH	Kısmen güncelle	PATCH /users/42
DELETE	Sil	DELETE /users/42

Durum kodu sonucu söyler

SINIF	KODLAR	ANLAMI
2xx	200 • 201 • 204	Başarılı; oluşturuldu; içerik yok
4xx	400 • 401 • 403 • 404 • 422	Hatalı veri, kimlik, yetki, kaynak yok, doğrulama
5xx	500	Sunucu tarafında beklenmeyen hata

Bir sayfanın künyesi

2

current_page

last_page: 15

20

per_page

Üst sınır: $\min(\$adet, 100)$

21–
40

from – to

Bu sayfadaki kayıtlar

287

total

Toplam kayıt

KAYNAKTAKİ ÖRNEK • OFFSET

Sayfa büyüdükçe OFFSET pahaladır.

On bininci sayfaya gitmek için veritabanının 200.000 kaydı okuyup geçmesi gerekiyor; çözümlerden biri cursor pagination.

Her yanıt aynı iskelette

İstemci her uçta aynı dört alanı okur:
`success`, `data`, `message`, `errors`.

response.json

```
1 {  
2   "success": true,  
3   "data": {},  
4   "message": null,  
5   "errors": null  
6 }
```

KATMAN 4 • YANIT ZARFI

Liste de aynı zarfta

Sayfalı liste `data.items` ve `data.meta` olarak döner; zarf değişmez.

response.json

```
1 {
2   "success": true,
3   "data": {
4     "items": [],
5     "meta": {
6       "current_page": 1,
7       "per_page": 15,
8       "total": 243,
9       "last_page": 17
10    }
11  },
12  "message": null,
13  "errors": null
14 }
```

Hata da bir sözleşmedir

`code` makine için sabittir, `message` insan için değişebilir, `details` ek bağlam taşır.

error.json

```
1 {
2   "error": {
3     "code": "VALIDATION_FAILED",
4     "message": "Gönderilen veriler doğrulanamadı.",
5     "details": [
6       {
7         "field": "email",
8         "message": "Geçerli bir e-posta adresi giriniz."
9       },
10      {
11        "field": "phone",
12        "message": "Telefon numarası zorunludur."
13      }
14    ]
15  }
16 }
```

Zarf ve hata kuralları



200 İLE
HATA

Hata için 200 + "error": true çalışır ama REST'e aykırı; cache ve loglama bozulur.



BİÇİM

Hata kodu büyük harf ve alt çizgiyle: QUOTA_EXCEEDED.



KODLAR

Yeni kod eklenebilir, eski kod kaldırılmamalı.



ÖLÇEK

Tek istemcili küçük serviste durum kodu ve kısa mesaj yeter.

Üç strateji, tercih URI

STRATEJİ	ÖRNEK	TAKAS
URI	GET /api/v1/users	Log ve proxy'de net görünür; URL'leri çoğaltır
Header	X-API-Version: 2	URL temiz kalır; eksik header belirsizlik doğurur
Query	GET /api/users?version=2	Başlangıçta pratik; proxy ve CDN görmeyebilir

Temel düzeyde: hangi değişiklik yeni sürüm ister?

✓ Uyumlu

Yanıtta yeni bir alan eklemek.

İsteğe bağlı bir parametre eklemek.

Hata mesajlarını zenginleştirmek.

✗ Breaking

Var olan bir alanı kaldırmak veya yeniden adlandırmak.

Zorunlu bir parametre eklemek.

Yanıt yapısını ya da durum kodunu değiştirmek.

KULLANIMDAN KALDIRMA

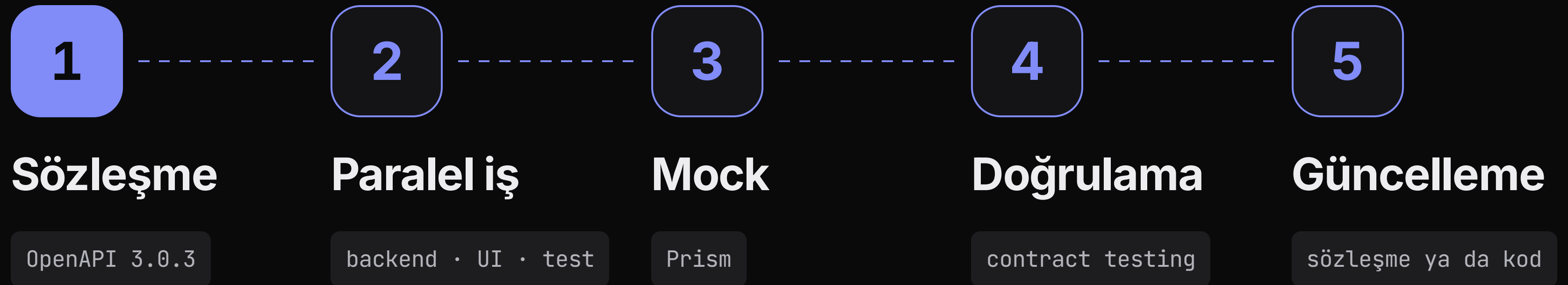
En az 6–12 ay önceden duyurun.

Dış istemcilere açık bir API'de en az 6-12 aylık bir deprecation süreci makul.

Sürümün ömrünü başlıkla bildirin

BAŞLIK	BELGE	NE SÖYLER
<code>Deprecation: @1780272000</code>	RFC 9745	Ne zaman kullanımdan kaldırıldığını; değer bir tarih, true geçersiz
<code>Sunset: Sat, 31 Oct 2026 23:59:59 GMT</code>	RFC 8594	Ne zaman yanıt vermeyeceğini
<code>Link: <.../docs/migrate-v2>; rel="deprecation"</code>	RFC 9745	Geçiş belgesinin adresini

Önce sözleşme, sonra kod



Sözleşme koddan önce yazılır

Örnek sözleşmenin şema kesiti.
`/orders` ucu `status` filtresini
`pending, completed, cancelled` ile
sınırlar.

```
openapi.yaml

1 components:
2 schemas:
3   Order:
4     type: object
5     required: [id, status, total]
6     properties:
7       id:
8         type: integer
9       status:
10        type: string
11       total:
12        type: number
```

Contract-first ne zaman değer?



KÜÇÜK
PROJE

Tek geliştirici, küçük proje: contract-first overkill olabilir.



ÇOK
TÜKETİCİ

Web, mobil, üçüncü taraf: kesinlikle zaman kazandırıyor.



UZUN ÖMÜR

Yıllarca yaşayacak API'de sözleşme bir karar kaydıdır.

Teşekkürler

muhammetsafak.com.tr

Yedi katman, yedi tutarlılık sözü. Ayrıntılar altı kaynak yazıda.

