# **TEMEL VERİTABANI TASARIMI**

(MONGODB örnekli, SQL karşılıkları da notlu)

## ****1.1 Kullanıcı ve Kimlik Yönetimi****

### ****users (kullanıcılar)****

json

{

"\_id": ObjectId,

"email": "test@domain.com",

"password\_hash": "...",

"role": "user" | "admin" | "instructor",

"fullname": "Ad Soyad",

"verified": true/false,

"created\_at": date,

"birth\_year": 1990,

"city": "İstanbul",

"devices": [

{ "device\_id": "...", "refresh\_token": "...", "last\_login": date }

],

"wallet\_balance": 120.50,

"subscription": {

"active": true/false,

"plan": "premium",

"expire\_at": date,

"subscription\_id": "..."

},

"instructor": { // eğitmense

"bio": "...",

"social\_links": [ ... ],

"application\_status": "pending" | "approved" | "rejected"

}}

* ****Her kullanıcı:**** role, cihazlar, abonelik ve cüzdan alanlarını kendi dokümanında tutar.

### ****devices (isteğe bağlı ayrı koleksiyon)****

(eğer çoklu cihaz yönetimini ayrı optimize etmek istersen)

json

{

"user\_id": ObjectId,

"device\_id": "...",

"refresh\_token": "...",

"last\_used": date}

Normalde users.devices alt dizisinde tutulabilir, ama log analizi veya çoklu cihaz için ayrı dizi de olur.

### ****sessions****

(isteğe bağlı, Redis veya expiring collection ile)

json

{

"session\_id": "...",

"user\_id": ObjectId,

"device\_id": "...",

"created\_at": date,

"last\_activity": date,

"jwt\_token": "..."}

## ****1.2 Abonelik ve Faturalandırma****

### ****subscriptions****

(her abone, geçmiş/aktif abonelikler için)

json

{

"\_id": ObjectId,

"user\_id": ObjectId,

"plan": "premium",

"status": "active" | "inactive" | "trial",

"start\_date": date,

"end\_date": date,

"payment\_id": "...",

"auto\_renew": true/false}

### ****invoices****

json

{

"\_id": ObjectId,

"user\_id": ObjectId,

"subscription\_id": ObjectId,

"product\_id": ObjectId,

"amount": 89.90,

"discount\_applied": 0,

"wallet\_used": true/false,

"created\_at": date,

"pdf\_url": "..."}

### ****payments****

json

{

"\_id": ObjectId,

"user\_id": ObjectId,

"method": "card" | "wallet" | "coupon",

"amount": 119.90,

"payment\_provider": "Stripe" | "Iyzico" | ...,

"status": "success" | "failed" | "pending",

"transaction\_id": "...",

"date": date}

## ****1.3 Video ve İzleme/İlerleme****

### ****videos****

json

{

"\_id": ObjectId,

"title": "Başlık",

"description": "...",

"slug": "baslik-video",

"stream\_id": "...", // Cloudflare Stream/videos ids

"public\_meta": {...},

"category": "Yoga",

"duration": 1200, // saniye

"created\_at": date,

"author": ObjectId,

"comments": [ { ... } ] // opsiyonel, ayrı collection tercih edilir}

### ****video\_progress****

(her user-video ilişkisi)

json

{

"\_id": ObjectId,

"user\_id": ObjectId,

"video\_id": ObjectId,

"watched": true/false,

"progress\_percent": 67,

"last\_time": 345, // saniye

"rated": 4, // (puan)

"comment\_id": ObjectId,

"updated\_at": date}

### ****video\_comments****

json

{

"\_id": ObjectId,

"video\_id": ObjectId,

"user\_id": ObjectId,

"comment": "Çok güzel!",

"rate": 5,

"created\_at": date}

## ****1.4 Ürün, Satış ve Sipariş Yönetimi****

### ****products****

json

{

"\_id": ObjectId,

"name": "Protein Tozu",

"slug": "protein-tozu",

"category": "Supplement",

"price": 125.00,

"stock": 25,

"description": "...",

"variants": [],

"image\_url": "...",

"active": true}

### ****orders****

json

{

"\_id": ObjectId,

"user\_id": ObjectId,

"products": [

{ "product\_id": ObjectId, "quantity": 2, "unit\_price": 120 }

],

"discount\_code": "KAZAN5",

"total\_amount": 220.00,

"status": "paid" | "cancelled" | "pending",

"payment\_id": ObjectId,

"invoice\_id": ObjectId,

"created\_at": date,

"delivery\_address": "..."}

### ****coupons****

json

{

"\_id": ObjectId,

"code": "INDIRIM5",

"discount\_percent": 5,

"max\_use": 100,

"expire\_at": date,

"active": true}

## ****1.5 Cüzdan / Wallet Servisi****

### ****wallets****

(isteğe bağlı, yoksa users.wallet\_balance da olabilir)

json

{

"\_id": ObjectId,

"user\_id": ObjectId,

"balance": 250.00,

"updated\_at": date}

### ****wallet\_transactions****

(Her para yükleme-ödeme-kullanımı)

json

{

"\_id": ObjectId,

"user\_id": ObjectId,

"amount": -50.00, // negatif: harcama, pozitif: yükleme

"type": "upload" | "spend" | "discount",

"reference": "orderId",

"date": date}

## ****1.6 Canlı Yayın & Topluluk****

### ****live\_streams****

json

{

"\_id": ObjectId,

"title": "Yoga Canlı",

"host\_id": ObjectId,

"start\_time": date,

"end\_time": date,

"attendee\_count": 13,

"chat\_id": ObjectId,

"public\_link": "https://link",

"stats": { ... }}

### ****community\_links****

json

{

"\_id": ObjectId,

"type": "telegram" | "whatsapp" | "discord",

"link": "https://t.me/kanal...",

"visible": true}

### ****live\_chat\_messages****

json

{

"\_id": ObjectId,

"live\_id": ObjectId,

"user\_id": ObjectId,

"message": "...",

"sent\_at": date}

## ****1.7 Eğitmen Yönetimi****

### ****instructor\_applications****

json

{

"\_id": ObjectId,

"user\_id": ObjectId,

"bio": "...",

"status": "pending" | "approved" | "rejected",

"social\_links": [ { "name": "instagram", "url": "..." } ],

"submitted\_at": date}

## ****1.8 Bildirimler & E-posta****

### ****notifications****

json

{

"\_id": ObjectId,

"user\_id": ObjectId,

"type": "email" | "push" | "sms",

"event": "purchase" | "subscription" | "live" | "progress",

"content": "...",

"sent\_at": date,

"read": false}

## ****1.9 Admin ve Loglar****

### ****admin\_logs****

json

{

"\_id": ObjectId,

"type": "login" | "product\_update" | "video\_upload" | "error" | ...,

"trigger\_user": ObjectId,

"data": { ... },

"created\_at": date}

# **2. İLİŞKİLER VE NOTLAR**

* ****Çoğu ilişki ObjectId veya user\_id ile referanslanmalı.****
* ****orders/products**** örneği: Toplu alışveriş ve kupon işlemleri için esnek olmalı.
* ****Ayrı koleksiyonlar**** (ör: notifications, logs) ölçeklenebilirlik için iyidir.
* ****Progress ve video ilişkisi**** Number+boolean tutmak için ayrı koleksiyonda saklanmalı (bir kullanıcı birçok video izleyebilir!)

# **3. PERFORMANS & GÜVENLİK**

* JWT refresh ve cihaz yönetimi için users.devices kullan ya da ayrı devices koleksiyonu.
* Stripe/Ödeme eventleri webhook tablosunda tutulabilir.
* Ödemede wallet ile ödeme için order/payment/wallet\_transactions zincirine dikkat et.
* Her event sonrası notification ve admin log eventlerini ayrı enqueue et.

# **4. ENTEGRASYON & GELİŞTİRİCİ DOKÜMANTASYONU**

* Tüm API uç noktalarını (endpointleri) OpenAPI/Swagger ile dökümante et.
* Servisler arası ortak alanlar uuid/güvenli token ile ilişkilendirilmeli.

# **5. TABLO/KOLEKSİYON LİSTESİ (Özet)**

* users
* devices (opsiyonel)
* sessions (opsiyonel)
* subscriptions
* payments
* invoices
* products
* orders
* coupons
* wallets / wallet\_transactions
* videos
* video\_progress
* video\_comments
* live\_streams / community\_links / live\_chat\_messages
* instructor\_applications
* notifications
* admin\_logs