

EduLink+ Proje Sunumu 1

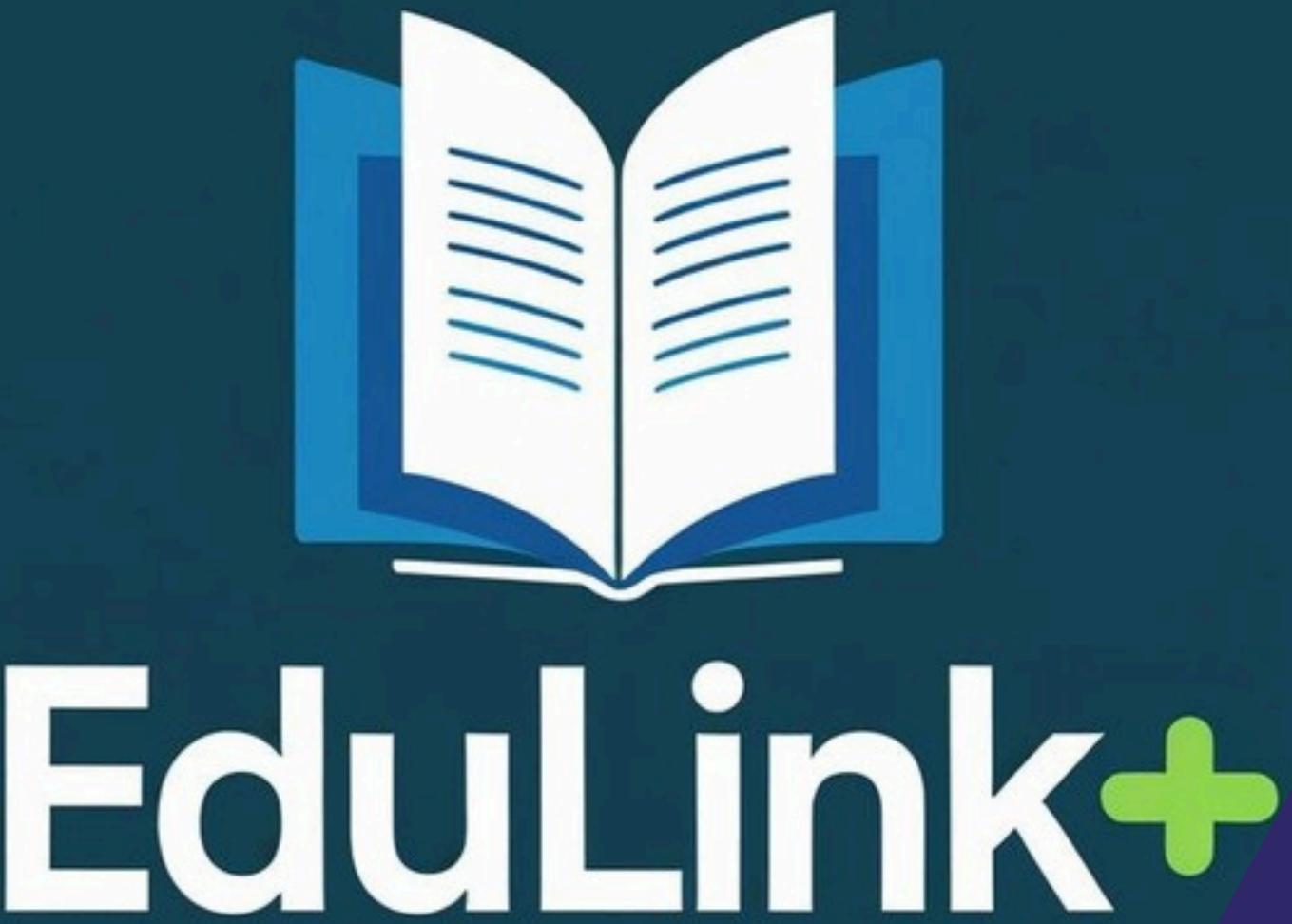
21.10.2025

Selanur AYAZ
Şevval Öcal
Yunus Emeç
Emin Töre



EduLink+

Kullanıcının ilgi alanına göre en doğru içerikleri (video, makale, podcast, test, akademik kaynak vb.) akıllı şekilde öneren; isterse odak moduna geçerek dikkat takibi yapan ve kişiselleştirilmiş öğrenme/çalışma planları sunan bulut tabanlı bir öğrenme ve bilgi yönetim platformudur.



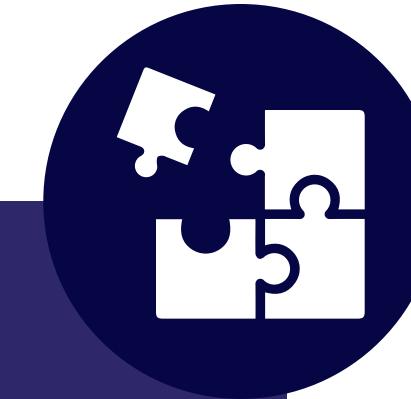
Uygulama Fikrimiz



Problem/Ihtiyaç

İnsanlar öğrenmek / çalışmak istiyor ama:

- Doğru içeriği bulamıyor
- Dikkati dağılıyor
- Kişiselleştirilmiş yönlendirme yok
- Verimliliğini ölçemiyor



Hedef Kitlemiz

- Öğrenciler
- Akademisyenler
- Kendi kendine öğrenen herkes
- "Sadece öğrenci değil"

Öne Çıkan Özelliklerimiz

01 **İçerik Öneri Motoru**

02 **Konu/İçerik Seçimi
filtreleme sistemi**

03 **Verilerin bulutta
saklanması**

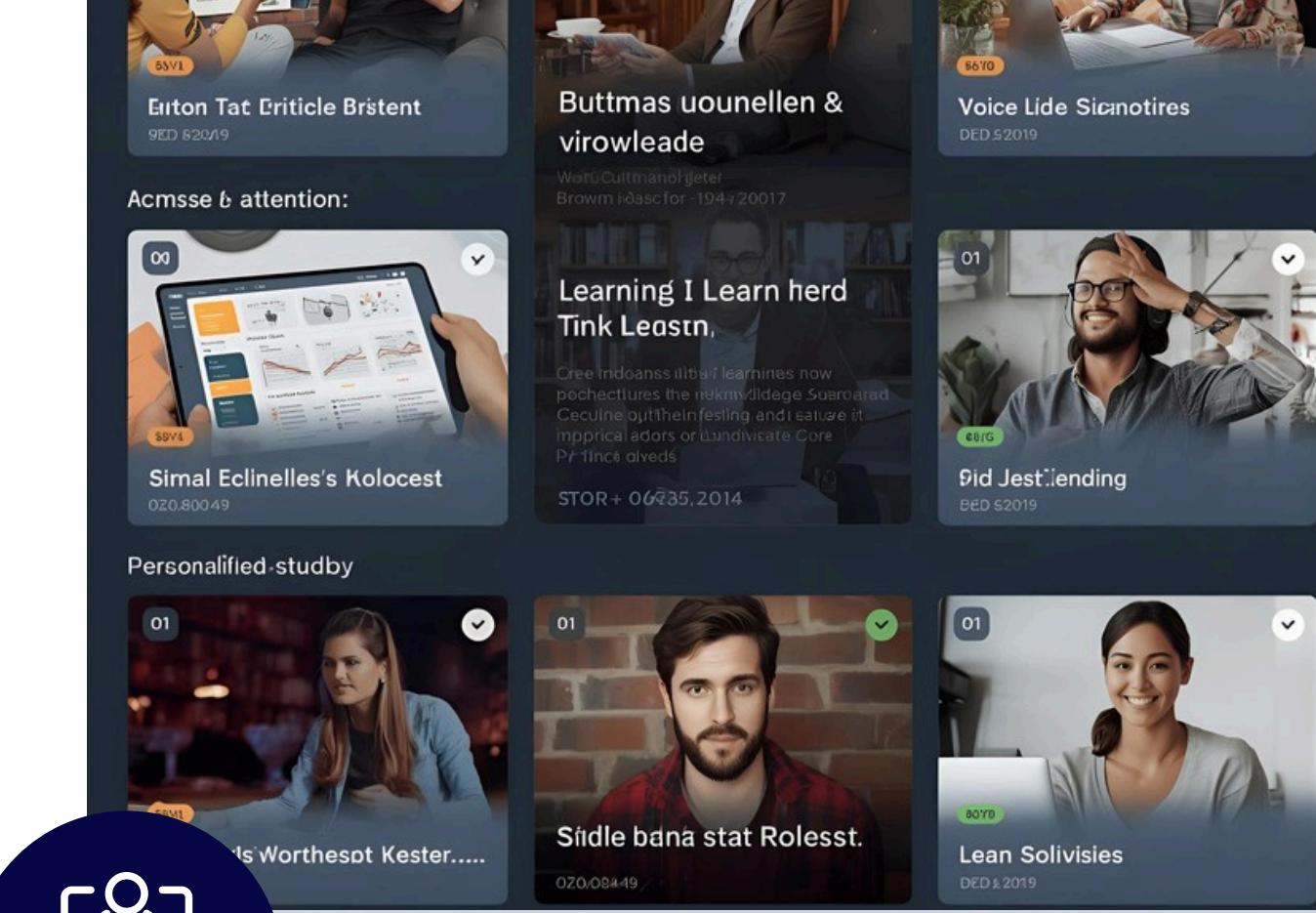
04 **Basit raporlama**

05 **Odak modu aç/kapa**

06 **Odak takibi**

07 **Odak skoru belirleme**

08 **Odak skoruna göre
çalışma yöntemi
önerisi**



Persuralle Edcialation,
Bnowlerts the Cuiltemiy
Managemet pary

Join representateamearning lorliou donesuents for
personalized nequmegetry.complaint.

Reeuemtore



1

İçerik Öneri Motoru

Kullanıcının daha önce incelediği kaynaklara benzer kaynak önerisi.

2

Konu/İçerik Seçimi filtreleme sistemi

Arama motorunda olacak içerik konusu filtreleme.
Örneğin Biyoloji-İskelet Sistemi-Video , Tıp-Onkoloji-Makale gibi.

Basit raporlama

Kullanıcı belirtilen içeriğe ne kadar vakit geçirdi? Ne kadarını tamamladı? (Yüzde ile gösterme)

3

Odak modu aç/kapa

Kullanıcı isteğine bağlı olarak kullanılması önemli , öğrenciyi geren bir sistem istemeyiz . "Sanırım odaklanmadan zorlanıyorsun, bir molaya ne dersin?" , " Odağını kaybediyorsun 25-5 teknigini deneyebilirsin." gibi uyarılar alabilir.

4

5

TEKNOLOJİ KISMI



SQL

1. SENARYO

- PostgreSQL (Supabase) → Ana veri
- TimescaleDB (PostgreSQL tabanlı zaman serisi DB) → Odak/analitik/log

2. SENARYO

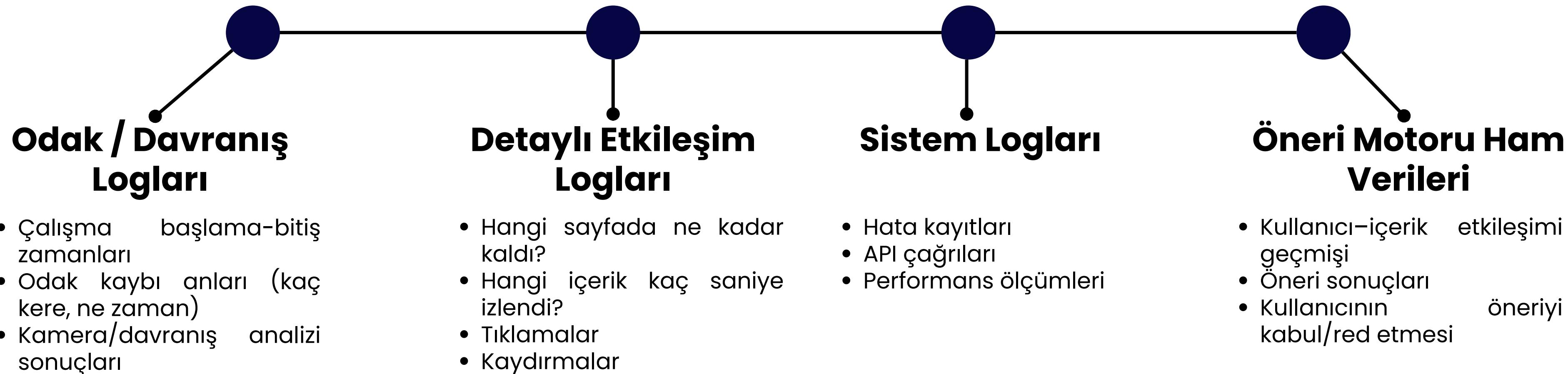
- PostgreSQL
- MySQL / MariaDB (ayrı instance olarak analitik/log için)



1.ANA VERİTABANI



2. VERİTABANI



Supabase



Supabase bizim ana backend altyapımız olacak.

Kullanıcı kayıt/giriş

Kullanıcı profilleri

İçerik listeleri (video, makale vs.)

Çalışma planları

Puanlar / rozetler / seviyeler

API üzerinden frontend ile haberleşme

Gerçek zamanlı senkronizasyon

- PostgreSQL tabanlı güçlü bir SQL veritabanı sağlar.
- Otomatik API oluşturur (tablolar → REST/GraphQL).
- Auth sistemi (kullanıcı giriş/çıkış) hazır gelir.
- Realtime özelliklere sahiptir (canlı veri güncellemesi).
- Storage (dosya/resim/video) yönetimi içerir.
- Serverless fonksiyonlar yazılabilir.

İçeriklerimiz Nereden Sağlanacak?

Video İçerikleri

YouTube Data API v3
Khan Academy API

Sesli İçerikler

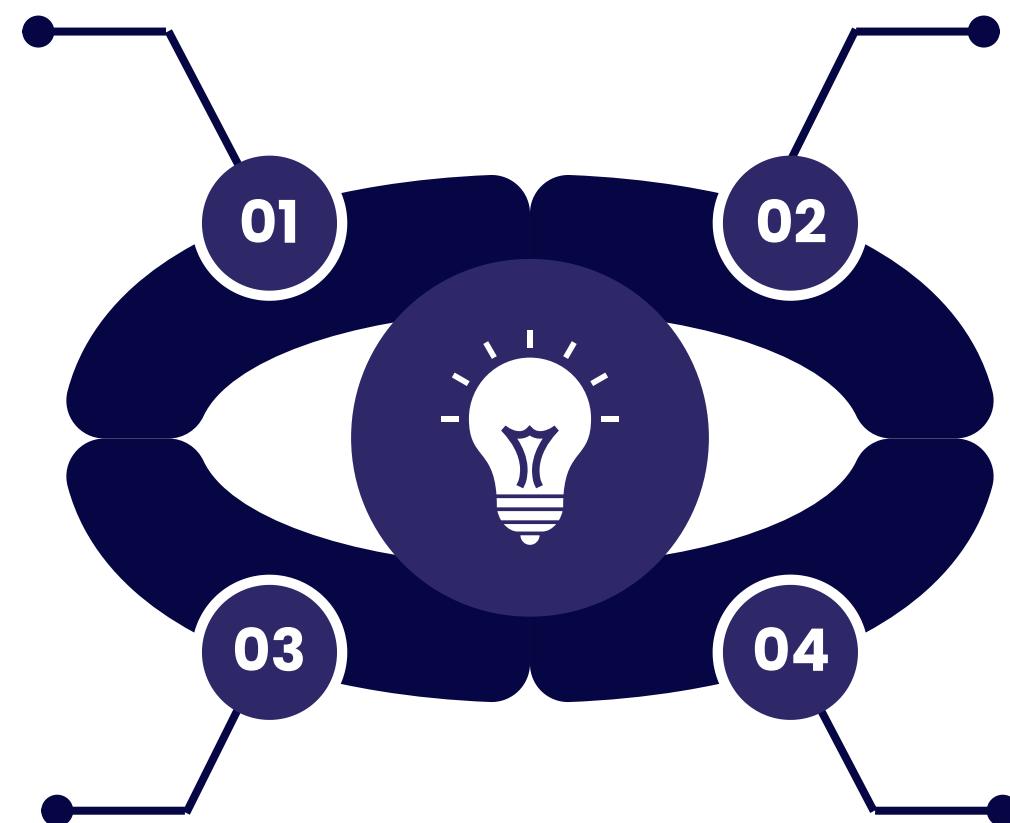
ListenNotes API

Yazılı İçerikler

Open Library API
Wikipedia API
Medium API
Semantic Scholar API

Test İçerikleri

Open Trivia Database
(OpenTDB)
Khan Academy API



EduLink+, içeriklerini manuel olarak girmek yerine çeşitli açık API'lardan otomatik olarak çeker.

YouTube ve Khan Academy API'leri üzerinden video dersler, Open Library ve Semantic Scholar API'leri üzerinden yazılı ve akademik içerikler, ListenNotes API'siyle de podcast'ler toplanır.

Tüm içerikler Supabase veritabanında meta olarak saklanır ve kullanıcının ilgi alanına göre öneri sistemine aktarılır.

İLERLEME PLANIMIZ

1.AY

Sistemin iskeleti kurulsun, içerik dış kaynaklardan çekilebilsin.

Hafta 1: Proje kurulumu + Supabase + Auth

Hafta 2: Kullanıcı ve içerik veritabanı modelleri

Hafta 3: YouTube ve Khan Academy'den içerik çekme entegrasyonu

Hafta 4: İçerik gösterme + çalışma seansı başlatma ve süre kaydetme

- Kullanıcı giriş yapabiliyor
- Konu seçip içerik önerisi alıyor
- İçeriğe tıklayıp çalışmaya başlayabiliyor
- Süre kaydediliyor



İLERLEME PLANIMIZ

2.AY

Öğrenciye akıllı öneri sunalım, çalışmasını takip edelim, arayüzü daha kullanışlı yapalım.

Hafta 5: Basit öneri motoru (kural tabanlı)

Hafta 6: Raporlama / Dashboard (ne çalıştı, ne kadar süre)

Hafta 7: Kullanıcı testleri + iyileştirmeler

Hafta 8: Favori içerik, ilerleme yüzdesi, küçük ödül/başarı sistemi

- Sistem kişiselleşmeye başlıyor
- Kullanıcı “ilerlediğini” hissediyor
- UI daha kullanışlı hale geliyor
- İlk defa gerçek test kullanıcılarından feedback alıyoruz



İLERLEME PLANIMIZ

3.AY

Hafta 9: Odak modu

Hafta 10: Performans, güvenlik, hata düzeltme

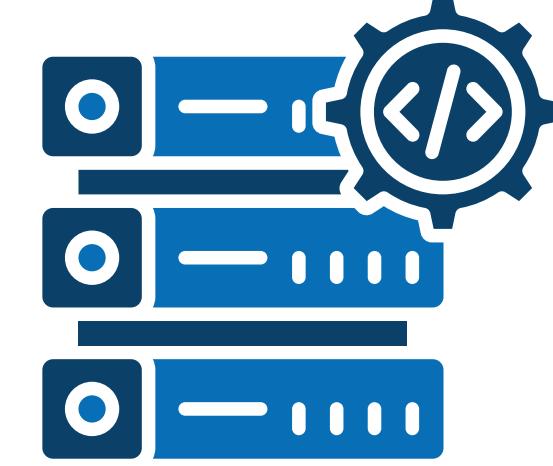
Hafta 11: Final demo + Sunum + Yedek video

- MVP çalışır halde
- Opsiyonel odak modu var





EKİP DAĞILIMI



Frontend – Şevval

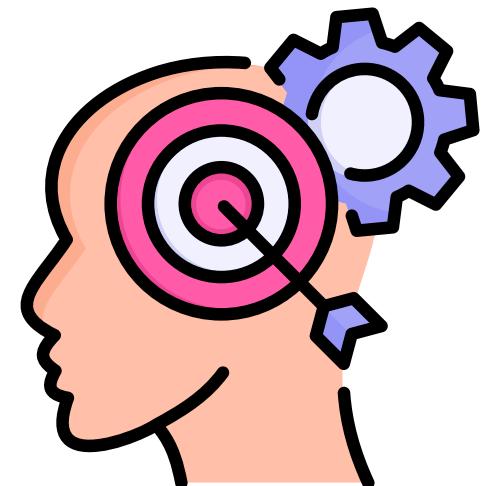
Uygulamanın kullanıcıya görünen kısmı:

- Login / Register sayfaları
- Dashboard (öğrenme ilerlemesi, önerilen içerikler)
- İçerik listeleme ve görüntüleme ekranları
- Çalışma seansı başlatma arayüzü
- Favoriler / profil sayfası
- UI/UX tasarım ve responsive yapı
- Tasarım + kullanıcı deneyimi tamamen Şevval'de

Odak Takibi – Yunus

Odak modu aktif olduğunda çalışan sistem:

- Kamera erişimi / izin yönetimi
- MediaPipe veya göz/baş pozisyon takibi
- Odaklanma/dikkat kaybı logic
- Odak skor hesaplama (ör. yüz görünmüyör = düşüş)
- Verileri backend'e gönderme



Selanur & Emin

Veritabanı (Supabase üzerinde)

- Kullanıcılar
- İçerikler (YouTube, Khan vb. kaynaklardan çekilen)
- Kullanıcı içerik etkileşimleri (hangi video açıldı, ne kadar izledi?)
- Çalışma seansları logları
- Odak verileri (Yunus'tan gelecek)
- Gamification (puan, seviye)

API Entegrasyonları

- YouTube API (video çek)
- Khan Academy API (eğitim içerikleri)
- Opsiyonel: Open Library / Podcast API
- İçeriği Supabase'e kaydet / önbellekle

Öneri Sistemi (basit → gelişmiş)

- Basit kural tabanlı (kategori seç + uygun içerik ver)
- Günlük motivasyon / çalışma önerileri
- "Bugün şu kadar çalışın, şöyle devam et!" mesajları

Authentication + Authorization

- Supabase Auth
- Kullanıcı rolleri (gerekirse: öğrenci/pro)

Backend API (REST veya GraphQL)

- Frontend ile bağlantı kurulan endpointler

TEŞEKKÜRLER

