# **Erişilebilirlik Platformu MVP Proje Planı**

## **1. Proje Vizyonu ve MVP Kapsamı**

**Proje Vizyonu:** Eğitimde fırsat eşitliğini teknolojiyle desteklemek amacıyla, görme ve işitme engelli öğrencilerin dijital ders materyallerine (PDF, ses, video) engelsiz bir şekilde erişmelerini sağlayan, yapay zeka destekli, web tabanlı bir "Erişilebilirlik Dönüşüm Platformu" oluşturmak. Bu platform, statik içeriği dinamik, etkileşimli ve erişilebilir formatlara dönüştürerek öğrenme sürecindeki bariyerleri ortadan kaldırmayı hedefler. Temel amaç, her öğrencinin, engel durumundan bağımsız olarak, eğitim materyallerinden tam ve eşit bir şekilde faydalanabilmesini sağlamaktır.

**MVP Kapsamı:** 6-8 haftalık bootcamp süresi ve kısıtlı kaynaklar göz önünde bulundurularak, MVP (Minimum Uygulanabilir Ürün) kapsamı, en temel ve en yüksek değerli iki ana kullanıcı yolculuğuna odaklanacaktır. Bu odaklanma, projenin belirtilen süre içinde tamamlanabilir, test edilebilir ve sunulabilir olmasını garanti altına almak için stratejik bir karardır.

1. **Görme Engelli Öğrenci Yolculuğu:** Kullanıcının bir PDF dokümanı yükleyip, metin içeriğini sese dönüştürebilmesi ve bu ses dosyasını dinleyebilmesi. Bu yolculuğun kritik bir parçası olarak, PDF içindeki görseller için yapay zeka tarafından otomatik olarak oluşturulmuş betimleyici açıklamaların (alternative text) da seslendirilecek metne dahil edilmesi hedeflenmektedir.
2. **İşitme Engelli Öğrenci Yolculuğu:** Kullanıcının bir ses veya video dosyası yükleyip, konuşma içeriğinin metne dönüştürülerek (transkripsiyon) ekranda görüntülenebilmesi. Bu, özellikle altyazısı olmayan eğitim videoları veya podcast'ler için hayati bir işlevdir.

**Kapsam Dışı Bırakılanlar (MVP Sonrası İçin):** Proje fikrinde belirtilen "işaret dili desteği" gibi son derece karmaşık ve özel uzmanlık gerektiren özellikler, MVP kapsamının bilinçli olarak dışında tutulmuştur. Bu kararın altında yatan neden, projenin başarısını riske atmamaktır. İşaret dili çevirisi, basit bir API çağrısının çok ötesinde, kendi içinde büyük bir araştırma ve geliştirme projesidir. Teknik olarak, metinden veya sesten işaret diline çeviri süreci; (1) kaynak metni anlamak için ileri düzey Doğal Dil İşleme (NLP), (2) metni, kelime kelime bir çeviri olmayan işaret dilinin dilbilgisel yapısına dönüştürme, (3) bu çeviriyi canlandıracak bir 3D avatarın insan pozu tahmini (pose estimation) ve render edilmesi gibi çok sayıda karmaşık ve birbiriyle ilişkili yapay zeka alanı gerektirir. Bu seviyede bir geliştirme, bir bootcamp ekibinin 6-8 haftalık zaman çizelgesini ve kaynaklarını fersah fersah aşar. Bu özelliğe odaklanmak, projenin temel ve ulaşılabilir hedeflerinin tamamen gözden kaçırılmasına neden olacaktır. Bu nedenle, bu özellik "Gelecek Faz" olarak etiketlenerek, ekibin ulaşılabilir ve yine de son derece etkileyici bir MVP sunmasına olanak tanınmıştır. Bu, bir ürün yönetimi ve çözüm mimarisi perspektifinden, projenin başarısını garantileyen en önemli stratejik kararlardan biridir.

## **2. Hedef Kitle ve Kullanıcı Hikayeleri**

Bu bölümde, platformun hizmet edeceği kullanıcıları ve onların ihtiyaçlarını somutlaştıracağız. Bu hikayeler, geliştirilecek özelliklerin temelini oluşturacak ve ekibin empati kurarak, kullanıcı odaklı bir ürün geliştirmesine yardımcı olacaktır.

### **2.1. Persona: Görme Engelli Öğrenci (Arda, 21, Üniversite Öğrencisi)**

* **Profil:** Arda, bir devlet üniversitesinde tarih bölümü öğrencisidir. Ders materyallerinin büyük bir kısmı hocaları tarafından PDF formatında paylaşılmaktadır. Mevcut ekran okuyucu yazılımları, özellikle çok sütunlu veya resim içeren akademik makalelerde yetersiz kalmakta, okuma sırasını karıştırmakta ve en önemlisi, tarihi harita, gravür gibi görselleri tamamen atlamaktadır. Arda, derslerine verimli bir şekilde hazırlanmak ve görsel materyalleri de anlayabilmek için materyalleri kolayca dinleyebileceği entegre bir çözüme ihtiyaç duymaktadır.
* **Kullanıcı Hikayesi 1 (Temel Dönüşüm):** Bir **tarih öğrencisi olarak**, ders notlarımın olduğu 100 sayfalık bir PDF dosyasını platforma yükleyebilmek istiyorum, **çünkü** bu sayede tüm metni tek bir ses dosyası olarak indirip metroda veya yürürken dinleyerek ders tekrarı yapabilirim.
* **Kullanıcı Hikayesi 2 (Görsel Erişilebilirlik):** Bir **görme engelli öğrenci olarak**, yüklediğim PDF'deki haritaların ve tarihi resimlerin ne içerdiğini anlatan kısa açıklamaları metinle birlikte duyabilmek istiyorum, **çünkü** bu görseller sınavda karşıma çıkabilir ve içeriğini bilmem, dersin bütünlüğünü anlamam için kritik öneme sahip.
* **Kullanıcı Hikayesi 3 (Kullanım Kolaylığı):** Bir **ekran okuyucu kullanıcısı olarak**, platformdaki dosya yükleme, dönüştürme ve ses oynatma düğmelerini klavyemle kolayca kullanabilmek ve her bir elementin ne işe yaradığını net bir şekilde duyabilmek istiyorum, **çünkü** erişilemez bir arayüz, platformun kendisini benim için kullanışsız hale getirir.

### **2.2. Persona: İşitme Engelli Öğrenci (Elif, 19, Bootcamp Öğrencisi)**

* **Profil:** Elif, bir yazılım bootcamp'ine katılan ve işitme güçlüğü çeken bir öğrencidir. Derslerin bir kısmı, özellikle konuk konuşmacıların yer aldığı oturumlar veya teknik mülakat simülasyonları, video kaydı olarak paylaşılmaktadır. Bu videolarda genellikle doğru veya senkronize altyazılar bulunmamaktadır. Elif, bu konuşmaları eksiksiz anlayabilmek ve önemli teknik terimleri doğru bir şekilde not alabilmek için konuşmaların metne dökülmüş haline ihtiyaç duymaktadır.
* **Kullanıcı Hikayesi 1 (Temel Dönüşüm):** Bir **bootcamp öğrencisi olarak**, 45 dakikalık bir ders videosu (mp4) veya ses kaydını (mp3) platforma yükleyebilmek istiyorum, **çünkü** konuşmaların tam metnini okuyarak derste anlatılan kodlama konseptlerini ve karmaşık algoritmaları kaçırmadığımdan emin olabilirim.
* **Kullanıcı Hikayesi 2 (İçerik Arama):** Bir **işitme engelli öğrenci olarak**, oluşturulan metin transkripti içinde anahtar kelime (örneğin, "dependency injection") araması yapabilmek istiyorum, **çünkü** bu sayede videonun tamamını izlemeden sadece ilgili bölümün metnini hızlıca bulup tekrar edebilirim.
* **Kullanıcı Hikayesi 3 (Kullanıcı Deneyimi):** Bir **kullanıcı olarak**, dosya yükleme ve işleme sürecinin ilerlemesini (örneğin, "%50 tamamlandı") görsel bir ilerleme çubuğu ile takip edebilmek istiyorum, **çünkü** büyük dosyaların işlenmesinin ne kadar süreceğini bilmek ve işlemin tarayıcıda takılıp kalmadığından emin olmak istiyorum.

## **3. Özellik Listesi ve Önceliklendirme**

Kullanıcı hikayelerinden türetilen özellikler, projenin yol haritasını oluşturmak için önceliklendirilecektir. Önceliklendirme, MoSCoW (Must-have, Should-have, Could-have, Won't-have) metodunun basitleştirilmiş bir versiyonu olan "Mutlaka Olmalı" ve "Olsa İyi Olur" kategorileri kullanılarak yapılacaktır. Bu yaklaşım, ekibin sınırlı zamanını en değerli işlevlere odaklamasını sağlar.

### **3.1. Mutlaka Olmalı (Must-Have for MVP)**

Bu özellikler, projenin temel değer önerisini sunmak ve bootcamp sonunda çalışır bir prototip sergilemek için kesinlikle tamamlanması gerekenlerdir. Her bir özellik, doğrudan bir veya daha fazla kullanıcı ihtiyacına cevap vermektedir.

| # | Özellik | İlgili Kullanıcı Hikayesi | Açıklama |
| --- | --- | --- | --- |
| 1 | **Kullanıcı Kayıt ve Giriş Sistemi** | - | Kullanıcıların kendi dosyalarını yönetebilmesi ve verilerinin gizliliğini sağlamak için temel e-posta/şifre tabanlı kimlik doğrulama. |
| 2 | **PDF Dosyası Yükleme Arayüzü** | Arda-1 | Kullanıcının bilgisayarından PDF dosyası seçip sunucuya yüklemesini sağlayan basit ve erişilebilir bir web arayüzü. |
| 3 | **PDF'ten Metin Çıkarma (OCR/Text Extraction)** | Arda-1, Arda-2 | Yüklenen PDF'in metin katmanını ve yapısal bilgilerini (sütunlar, paragraflar) doğru bir sırada çıkaran backend servisi. |
| 4 | **Metinden Sese Dönüştürme (TTS)** | Arda-1 | Çıkarılan metni, akıcı ve doğal bir insan sesine dönüştüren yapay zeka servisi entegrasyonu. |
| 5 | **Ses Dosyası Oynatma ve İndirme** | Arda-1 | Oluşturulan ses dosyasını web arayüzünde oynatmak için erişilebilir bir ses oynatıcı ve kullanıcıların dosyayı cihazlarına kaydetmesi için bir indirme butonu. |
| 6 | **PDF'ten Görüntü Çıkarma ve Açıklama (Image-to-Text)** | Arda-2 | PDF içindeki görüntüleri tespit edip, bu görüntüleri bir Vision AI servisine göndererek açıklayıcı metin (alt text) üreten ve bunu ana metin akışına ekleyen backend süreci. |
| 7 | **Ses/Video Dosyası Yükleme Arayüzü** | Elif-1 | Kullanıcının bilgisayarından ses (mp3, wav) veya video (mp4) dosyası seçip yüklemesini sağlayan arayüz. |
| 8 | **Sesten Metne Dönüştürme (STT)** | Elif-1 | Yüklenen ses/video dosyasının konuşma içeriğini metne dönüştüren (transkripsiyon) yapay zeka servisi entegrasyonu. |
| 9 | **Transkript Görüntüleme ve Arama** | Elif-1, Elif-2 | Oluşturulan metin transkriptini okunabilir bir formatta gösteren ve içinde basit metin araması yapmaya olanak tanıyan arayüz bileşeni. |
| 10 | **Dosya İşleme Durum Bildirimi** | Elif-3 | Büyük dosyaların sunucuda işlenmesi sırasında kullanıcıya geri bildirim sağlayan basit bir durum göstergesi (örn: "İşleniyor...", "Tamamlandı", "Hata oluştu"). |
| 11 | **Erişilebilir Frontend Tasarımı (WCAG AA Temelleri)** | Arda-3 | Tüm interaktif elementlerin (düğmeler, formlar, linkler) klavye ile erişilebilir olması, role ve aria-\* etiketlerinin WAI-ARIA standartlarına uygun şekilde kullanılması. |

### **3.2. Olsa İyi Olur (Nice-to-Have / Gelecek Faz)**

Bu özellikler, MVP'nin temel işlevselliğini geliştirecek ancak zaman kısıtlamaları nedeniyle ilk sürüme dahil edilmeyebilecek öğelerdir. Bu liste, projenin potansiyel büyüme alanlarını ve gelecek vizyonunu göstermesi açısından da önemlidir.

| # | Özellik | Açıklama |
| --- | --- | --- |
| 1 | **Gelişmiş Ses Kontrolleri** | Kullanıcının oluşturulan sesi kendi dinleme tercihine göre ayarlayabilmesi için ses hızını (0.5x, 1.5x), perdesini veya farklı seslendirme karakterlerini (erkek/kadın) seçme imkanı. |
| 2 | **Altyazı Dosyası (SRT/VTT) Oluşturma** | Oluşturulan transkripti, standart video oynatıcılarla (VLC, YouTube vb.) uyumlu olan .srt veya .vtt formatlarında indirme seçeneği sunarak içeriğin taşınabilirliğini artırma. |
| 3 | **Çoklu Konuşmacı Tespiti (Speaker Diarization)** | Birden fazla kişinin konuştuğu ses kayıtlarında, transkriptte farklı konuşmacıları ("Konuşmacı A:", "Konuşmacı B:") otomatik olarak etiketleyerek diyaloğun takibini kolaylaştırma. |
| 4 | **Gelişmiş Durum Takibi (WebSocket)** | Dosya işleme durumunu periyodik sorgulama (polling) yerine WebSocket kullanarak sunucudan istemciye anlık olarak bildirme. Bu, daha modern ve verimli bir kullanıcı deneyimi sunar. |
| 5 | **Özetleme (Summarization)** | Yapay zeka servislerini kullanarak uzun PDF'ler veya transkriptler için anahtar noktaları içeren otomatik bir özet çıkarma özelliği. |
| 6 | **İşaret Dili Desteği** | **Kesinlikle MVP Dışı.** Metin veya sesten işaret dili çevirisi yapan bir avatar oluşturma. Bu, son derece yüksek teknik karmaşıklığı nedeniyle ayrı ve uzun soluklu bir proje olarak ele alınmalıdır. |

## **4. Önerilen Teknik Yığın (Tech Stack)**

Teknoloji seçimleri, bootcamp ekibinin hızlı öğrenme ve uygulama kabiliyeti, modern teknolojileri sergileme hedefi, projenin teknik gereksinimleri (asenkron işlemler, AI entegrasyonu) ve kaynak kısıtlamaları (ücretsiz veya düşük maliyetli hizmetler) doğrultusunda dikkatle yapılmıştır.

* **Frontend:** **Vue.js**
  + **Gerekçe:** Bootcamp gibi zamanın en değerli kaynak olduğu bir ortamda, geliştirme hızı kritik bir faktördür. Yapılan karşılaştırmalar, Vue.js'in React'e kıyasla daha yumuşak bir öğrenme eğrisine, daha temiz ve sezgisel bir sözdizimine sahip olduğunu göstermektedir. Bu durum, özellikle küçük ekiplerin ve yeni başlayanların daha hızlı bir şekilde üretken olmalarını sağlar. React, devasa ekosistemiyle büyük kurumsal projeler için güçlü bir seçenek olsa da, bu projenin MVP'si için Vue'nun sunduğu yapılandırılmış yaklaşım ve mükemmel dokümantasyon daha büyük bir avantajdır.
  + **State Management:** Uygulamanın durum yönetimi (state management) ihtiyacı, global olarak kullanıcı bilgisi ve işlenen dosyaların listesi gibi basit senaryolarla sınırlı olacaktır. Bu ölçekte, Redux gibi karmaşık ve çok fazla standart kod (boilerplate) gerektiren bir kütüphane kullanmak gereksiz bir karmaşıklık yaratacaktır. Bunun yerine, Vue'nun resmi state management kütüphanesi olan **Pinia** veya daha basit durumlar için Vue'nun kendi reaktiflik sistemi yeterli ve çok daha verimli olacaktır. Bu yaklaşım, React dünyasındaki "Context API mi, Redux mu?" tartışmasında küçük projeler için Context API'nin tercih edilmesine benzer bir mantığa dayanır.
* **Backend:** **Python** ile **FastAPI**
  + **Gerekçe:** Projenin omurgasını yapay zeka servisleriyle olan etkileşim oluşturmaktadır ve Python, bu alanda tartışmasız en zengin kütüphane ve araç ekosistemine sahiptir. Backend framework seçimi olarak FastAPI, bu proje için Django gibi daha geleneksel bir framework'e göre belirgin avantajlar sunar. FastAPI, asenkron programlamayı (async/await) temelden destekler. Bu, dış AI servislerine yapılan ve yanıt süresi belirsiz olan ağ istekleri sırasında sunucunun diğer istekleri işlemeye devam etmesini sağlayarak yüksek performans ve verimlilik sunar. Django'nun sonradan eklenen asenkron desteğine kıyasla FastAPI'nin bu konudaki mimarisi çok daha doğal ve performanslıdır. Ayrıca, FastAPI'nin Pydantic entegrasyonu sayesinde otomatik veri doğrulama ve Swagger/ReDoc arayüzleri üzerinden interaktif API dokümantasyonu oluşturması, geliştirme ve test süreçlerini önemli ölçüde hızlandırır. Bu proje API-merkezli olduğu için, Django'nun getireceği admin paneli gibi "dahili piller" birincil ihtiyaç değildir.
  + **Proje Yapısı:** Kodun sürdürülebilir ve ölçeklenebilir olması için modüler bir proje yapısı benimsenmelidir. Bu yapı, endüstri standartlarına uygun olarak "endişelerin ayrılığı" (separation of concerns) ilkesine dayanmalıdır. Önerilen yapı şu şekildedir: app/routers/ (API endpoint'leri), app/services/ (tüm iş mantığının bulunduğu katman), app/models/ (veritabanı modelleri), app/schemas/ (Pydantic veri doğrulama modelleri) ve app/core/ (konfigürasyon ve ayarlar).
* **Veritabanı:** **MongoDB**
  + **Gerekçe:** Bir MVP geliştirme sürecinde, veri modellerinin esnek olması ve sık sık değişebilmesi beklenir. MongoDB'nin şemasız (schema-less) yapısı, bu esnekliği mükemmel bir şekilde sağlar ve geliştiricileri katı şema migrasyonlarının getirdiği ek iş yükünden kurtarır. PostgreSQL gibi ilişkisel bir veritabanı, veri bütünlüğü açısından güçlü olsa da, her model değişikliğinde migrasyon gerektirmesi bootcamp projesinin hızını yavaşlatabilir. Projemizdeki veri ilişkileri (bir kullanıcının birden çok işlenmiş dosyası olabilir) karmaşık değildir. MongoDB'nin doküman tabanlı modeli, bu tür "bir'e çok" ilişkileri, bir kullanıcı dokümanı içinde işlenmiş dosya ID'lerini bir dizi olarak tutarak verimli bir şekilde yönetebilir. Ayrıca, JSON-benzeri BSON formatı, hem Python (sözlükler) hem de JavaScript/Vue.js (nesneler) ile doğal bir uyum içinde çalışarak geliştirici deneyimini iyileştirir.
  + **Örnek Koleksiyon Şemaları:**
    - users: \_id (ObjectId), email (String, unique), hashed\_password (String), created\_at (Date).
    - processed\_files: \_id (ObjectId), user\_id (ObjectId, users koleksiyonuna referans), original\_filename (String), file\_type (String, 'pdf' veya 'audio'), status (String, 'processing', 'completed', 'failed'), storage\_path (String), output\_text (String, transkript için), output\_audio\_path (String, TTS için), created\_at (Date).
* **Yapay Zeka Servisleri:** **Google Cloud Platform (GCP)**
  + **Gerekçe:** Projenin başarısı için en kritik teknoloji seçimi, kullanılacak yapay zeka servisleridir. Bootcamp'in bütçesiz doğası gereği, cömert ücretsiz kullanım katmanları (free tier) sunan platformlar önceliklidir. Bu bağlamda, Google Cloud Platform, projenin üç temel AI ihtiyacını (TTS, STT, Image-to-Text) tek bir platformda, entegre bir şekilde ve cazip ücretsiz limitlerle sunmasıyla öne çıkmaktadır. Farklı sağlayıcılardan (örn: STT için AssemblyAI, TTS için ElevenLabs) tek tek en iyi servisleri seçmek yerine, tek bir ekosistemde kalmak, kimlik doğrulama (tek bir servis hesabı anahtarı), SDK yönetimi ve potansiyel faturalandırma takibini önemli ölçüde basitleştirir. Bu, zaman kısıtlı bir projede geliştirme karmaşıklığını azaltan önemli bir stratejik avantajdır.
    - **Text-to-Speech (TTS):** Google Cloud Text-to-Speech API, WaveNet ve Neural2 gibi son derece doğal ve akıcı sesler sunar. Aylık 1 milyon karaktere kadar standart sesler ve 1 milyon karaktere kadar WaveNet sesleri için sağlanan ücretsiz kullanım hakkı, bu projenin tüm ihtiyaçlarını karşılayacaktır.
    - **Speech-to-Text (STT):** Google Cloud Speech-to-Text API, aylık 60 dakikaya kadar ücretsiz transkripsiyon imkanı sunar ve 125'ten fazla dili destekleyerek gelecekteki potansiyel genişlemelere de olanak tanır. Bu limit, MVP aşamasındaki testler için yeterlidir.
    - **Image-to-Text (Vision AI):** Google Cloud Vision API, PDF'lerdeki görselleri anlamlandırmak için gereken "Label Detection" veya "Web Detection" gibi özellikleri sunar. Aylık ilk 1000 birimlik ücretsiz kullanım, Arda'nın kullanıcı hikayesini gerçekleştirmek için fazlasıyla yeterlidir. Ayrıca, yeni GCP kullanıcılarına sunulan 300 dolarlık ücretsiz kredi, limitlerin aşılması durumunda bir güvence sağlar.
* **PDF İşleme:** **PyMuPDF (fitz)**
  + **Gerekçe:** Yüklenen PDF dosyalarından hem metin hem de görüntüleri verimli bir şekilde çıkarmak için güçlü bir kütüphane gereklidir. Bu alanda PyMuPDF, diğer popüler kütüphanelere (PyPDF2, pdfplumber) göre hızı ve kapsamlı yetenekleriyle öne çıkmaktadır. Sadece metin çıkarmakla kalmaz, aynı zamanda PDF içindeki görüntü nesnelerini de doğrudan çıkarabilir, bu da Arda'nın ikinci kullanıcı hikayesini (görsel betimleme) gerçekleştirmek için zorunludur. Ayrıca, metin bloklarını koordinat bilgileriyle birlikte yapısal olarak çıkarma yeteneği, çok sütunlu metinlerin doğru okuma sırasına göre işlenmesine olanak tanıyarak daha kaliteli bir sonuç üretir.
* **CI/CD & Dağıtım (Deployment):** **GitHub Actions & Render**
  + **Gerekçe:** Modern yazılım geliştirme pratiği olarak, projenin en başından itibaren bir CI/CD (Sürekli Entegrasyon/Sürekli Dağıtım) boru hattı kurmak, kod kalitesini artırır ve dağıtım süreçlerini otomatikleştirir. GitHub Actions, GitHub depolarıyla doğal entegrasyonu sayesinde bu iş için en basit ve en erişilebilir seçenektir. Her push işleminde otomatik olarak kod formatlama (linting) ve birim testlerini çalıştıran bir iş akışı (workflow) YAML dosyası ile kolayca tanımlanabilir. Uygulamanın canlıya alınması için **Render** gibi bir PaaS (Platform as a Service) hizmeti, bootcamp ekibini karmaşık sunucu yönetimi ve Docker yapılandırmalarından kurtarır. Render, ücretsiz bir katman sunar ve GitHub deposuna bağlanarak her main branch güncellemesinde otomatik dağıtımı tetikleyebilir. Bu, ekibin sadece koda odaklanmasını sağlayan son derece verimli bir yaklaşımdır.

## **5. Proje Fazlandırması (Sprint Planı Taslağı)**

Proje, her biri 2 hafta sürecek 3 ana sprint'e ayrılmıştır. Bu plan, Agile/Scrum metodolojisine uygun olarak tasarlanmış olup, ekibe her sprint için net hedefler, odaklanılacak görevler ve somut çıktılar sunar.

### **Sprint 1 (Hafta 1-2): Temel Altyapı ve Görme Engelliler için Çekirdek Özellik**

* **Sprint Hedefi:** Projenin temel iskeletini kurmak ve ilk somut, uçtan uca çalışan özelliği (PDF'ten sese) tamamlamak. Bu sprintin sonunda, bir kullanıcı sisteme giriş yapıp bir PDF yükleyebilmeli ve karşılığında bir ses dosyası alabilmelidir.
* **Tamamlanacak Görevler:**
  + **Altyapı ve Kurulum:**
    - GitHub deposu oluşturulacak, main ve develop branch'leri ile temel bir branch stratejisi belirlenecek.
    - Frontend (Vue.js) ve Backend (FastAPI) projeleri için modüler klasör yapıları oluşturulacak.
    - MongoDB veritabanı, yerel bir Docker konteynerinde veya bulut tabanlı MongoDB Atlas'ın ücretsiz katmanında kurulacak.
    - Temel bir CI (Sürekli Entegrasyon) boru hattı için GitHub Actions workflow dosyası (.github/workflows/ci.yml) oluşturulacak. Bu workflow, push anında kodun linting (biçim kontrolü) işlemini yapacak ve basit bir "hello world" testini çalıştıracak.
  + **Backend Geliştirme:**
    - Kullanıcı kaydı (/register) ve JWT (JSON Web Token) tabanlı giriş (/login veya /token) için API endpoint'leri oluşturulacak. Güvenlik için şifreler bcrypt ile hash'lenecek.
    - Kimlik doğrulaması gerektiren, PDF dosyası yüklemek için bir API endpoint'i (/upload/pdf) oluşturulacak.
    - Yüklenen PDF'i PyMuPDF kütüphanesi ile işleyip metin içeriğini çıkaran bir servis fonksiyonu yazılacak.
    - Google Cloud Text-to-Speech API'sine bağlanıp, çıkarılan metni sese dönüştüren ve bir MP3 dosyası olarak kaydeden bir servis entegrasyonu yapılacak.
    - Bu dosya işleme süreci, kullanıcı isteğini bekletmemek için FastAPI'nin yerleşik BackgroundTasks özelliği kullanılarak arka planda çalıştırılacak.
  + **Frontend Geliştirme:**
    - Kullanıcıların sisteme kaydolabileceği ve giriş yapabileceği formlar ve sayfalar oluşturulacak.
    - Giriş yapmış kullanıcılar için, PDF dosyası seçip yükleyebilecekleri bir arayüz bileşeni tasarlanacak.
    - Arka plan işlemi tamamlandığında, sunucudan dönen ses dosyasını oynatmak için temel bir HTML5 <audio> elementi ve indirme linki gösterilecek.

### **Sprint 2 (Hafta 3-4): İşitme Engelliler için Çekirdek Özellik ve Arayüz Geliştirmeleri**

* **Sprint Hedefi:** İkinci ana kullanıcı yolculuğunu (sesten metne) tamamlamak ve kullanıcı arayüzünü daha işlevsel ve kullanıcı dostu hale getirmek.
* **Tamamlanacak Görevler:**
  + **Backend Geliştirme:**
    - Ses (mp3, wav) veya video (mp4) dosyalarını kabul eden yeni bir API endpoint'i (/upload/audio) oluşturulacak.
    - Google Cloud Speech-to-Text API'sine bağlanarak yüklenen medya dosyasının transkriptini çıkaran bir servis entegrasyonu yapılacak. Uzun ses dosyaları için asenkron tanıma özelliği kullanılacak.
    - Oluşturulan transkript metni ve ilgili dosya bilgileri MongoDB'deki processed\_files koleksiyonuna kaydedilecek.
  + **Frontend Geliştirme:**
    - Ses ve video dosyalarını yüklemek için ayrı bir arayüz bileşeni oluşturulacak.
    - Elif'in 3. kullanıcı hikayesine yanıt olarak, dosya yükleme ve sunucuda işleme sırasında kullanıcıya görsel geri bildirim sağlamak amacıyla bir ilerleme göstergesi (progress bar) eklenecek. Bu bileşen, WAI-ARIA standartlarına uygun olarak role="progressbar" ve ilgili aria-valuenow, aria-valuemin, aria-valuemax etiketleriyle erişilebilir bir şekilde kodlanacak.
    - İşlem tamamlandığında, sunucudan gelen transkript metni ekranda okunabilir bir biçimde gösterilecek.
    - Elif'in 2. kullanıcı hikayesini karşılamak üzere, gösterilen transkript metni içinde basit bir Ctrl+F benzeri metin arama işlevselliği eklenecek.
    - Kullanıcının geçmişte yüklediği ve işlediği tüm dosyaları (hem PDF'ler hem de ses dosyaları) ve durumlarını listeleyen basit bir "Dosyalarım" sayfası oluşturulacak.

### **Sprint 3 (Hafta 5-6): Geliştirme, Test, Erişilebilirlik ve Sunuma Hazırlık**

* **Sprint Hedefi:** MVP kapsamındaki tüm özellikleri entegre edip tamamlamak, uygulamanın kararlılığını ve en önemlisi erişilebilirliğini en üst düzeye çıkarmak ve projeyi bootcamp sunumuna hazır hale getirmek.
* **Tamamlanacak Görevler:**
  + **Backend Geliştirme:**
    - Arda'nın 2. kullanıcı hikayesini tamamlamak için PDF içindeki görselleri açıklama özelliği entegre edilecek: PyMuPDF ile PDF'ten görseller çıkarılacak ve Google Vision API'ye gönderilerek açıklayıcı metinler üretilecek. Bu metinler, TTS servisine gönderilmeden önce ana metne eklenecek.
    - Tüm API endpoint'leri için kapsamlı hata yönetimi (error handling) ve Pydantic ile gelen veri doğrulama (validation) kuralları iyileştirilecek.
    - Temel güvenlik önlemleri gözden geçirilecek: Dosya yükleme boyut limitleri, izin verilen içerik tiplerinin (Content-Type başlığı) sunucu tarafında kontrolü gibi önlemler eklenecek.
  + **Frontend Geliştirme ve Erişilebilirlik:**
    - Uygulamanın genel UI/UX (Kullanıcı Arayüzü/Kullanıcı Deneyimi) tasarımı, tutarlı bir renk paleti ve yazı tipi kullanımıyla iyileştirilecek.
    - **Erişilebilirlik Odaklı İyileştirme (Accessibility Polish):**
      * Özelleştirilmiş ses oynatıcı gibi bileşenler, tamamen klavye ile kontrol edilebilir (play/pause, ses ayarı, ilerleme çubuğu) hale getirilecek ve durum değişiklikleri (örn: "oynatılıyor", "duraklatıldı") role="button" ve aria-pressed gibi ARIA etiketleriyle ekran okuyuculara bildirilecek.
      * Tüm uygulama genelinde kapsamlı bir klavye navigasyon testi yapılacak; odaklanılabilir her elemente (düğme, link, form alanı) Tab tuşu ile erişilebildiğinden ve odak sırasının mantıksal olduğundan emin olunacak.
      * Renk kontrastları, WCAG AA seviyesi için gerekli olan 4.5:1 oranını karşılayacak şekilde kontrol edilecek.
      * Ekran okuyucularla (Windows'ta NVDA, macOS'ta VoiceOver) temel bir test gerçekleştirilerek uygulamanın ana akışları deneyimlenecek.
  + **Genel Görevler:**
    - Uçtan uca (end-to-end) test senaryoları (örn: "Bir kullanıcı kaydolur, giriş yapar, bir PDF yükler, ses dosyasını indirir ve dinler") oluşturulacak ve ekip tarafından manuel olarak test edilecek.
    - Render üzerinde production ortamı kurulacak ve son sürüm dağıtılacak.
    - Projenin README.md dosyası, projenin amacını, kurulum adımlarını, kullanılan teknolojileri ve API endpoint'lerini açıklayacak şekilde detaylı olarak hazırlanacak.
    - Bootcamp final sunumu hazırlanacak ve prova edilecek.

## **6. Potansiyel Riskler ve Çözüm Önerileri**

Her projede olduğu gibi, bu projede de öngörülebilecek riskler ve bu risklere karşı alınabilecek proaktif önlemler bulunmaktadır. Bu riskleri projenin başında tanımlamak, ekibin beklenmedik sorunlar karşısında hazırlıklı olmasını sağlar.

* **Risk 1: Üçüncü Parti API Bağımlılığı ve Maliyet Yönetimi**
  + **Açıklama:** Projenin temel işlevselliği, Google Cloud tarafından sağlanan harici API'lerin (TTS, STT, Vision) çalışmasına ve bu servislerin ücretsiz kullanım limitlerine tamamen bağımlıdır. API'lerde yaşanabilecek bir kesinti, beklenmedik bir versiyon değişikliği veya kullanımdan kaldırılma (deprecation) durumu projenin kritik özelliklerini işlevsiz bırakabilir. Ayrıca, geliştirme ve test aşamasında ücretsiz limitlerin farkında olmadan aşılması, bootcamp projesi için öngörülmeyen maliyetlere yol açabilir.
  + **Çözüm:**
    1. **Soyutlama Katmanı (Abstraction Layer):** Google Cloud API'lerine yapılan çağrıları doğrudan iş mantığının içine yazmaktan kaçının. Bunun yerine, backend kodunda bir "servis" veya "adaptör" katmanı oluşturun. Örneğin, TextToSpeechService adında bir sınıf oluşturun ve bu sınıfın içinde Google'ın API'si çağrılsın. Bu yaklaşım, gelecekte farklı bir TTS sağlayıcısına (örneğin, Microsoft Azure veya AWS Polly) geçmek gerekirse, sadece bu servis sınıfının içindeki kodun değiştirilmesini gerektirir. Bu, API sağlayıcısına olan sıkı bağımlılığı (vendor lock-in) azaltır.
    2. **API Kullanım Takibi:** Google Cloud Console üzerinden, projenin API kullanımını gösteren bir pano (dashboard) oluşturun. Her sprintin başında bu panoyu düzenli olarak kontrol ederek limitlere ne kadar yaklaşıldığını takip edin. Bu proaktif izleme, sürprizleri önler.
    3. **Bütçe Uyarıları (Billing Alerts):** Google Cloud'un faturalandırma bölümünde, belirli bir harcama eşiği (örneğin, 10 dolar) aşıldığında proje sahiplerine e-posta gönderen bütçe uyarıları kurun. Ücretsiz limitler aşılsa bile bu uyarılar sayesinde maliyetler kontrol altında tutulur.
    4. **API Versiyon Sabitleme:** Kullandığınız istemci kütüphanelerinin (client libraries) versiyonlarını, backend projesinin requirements.txt dosyası içinde sabitleyin (örneğin, google-cloud-speech==2.20.0). Bu, pip install komutu çalıştırıldığında, sizin bilginiz dışında kütüphanenin yeni ve potansiyel olarak uyumsuz (breaking change içeren) bir versiyonuna geçilmesini engeller.
* **Risk 2: Kullanıcı Verilerinin Gizliliği ve Güvenliği**
  + **Açıklama:** Platform, öğrencilerin ders notları, akademik makaleler ve kişisel ses kayıtları gibi potansiyel olarak hassas ve özel verileri yüklemesini gerektirir. Bu verilerin yetkisiz erişime karşı korunmaması, hem ciddi bir güvenlik açığı hem de kullanıcı güvenini sarsacak bir gizlilik ihlalidir.
  + **Çözüm:**
    1. **Güvenli Dosya Yükleme Protokolü:** Yüklenen dosyalara asla orijinal isimleriyle sunucuda saklamayın. Bunun yerine, her dosya için rastgele ve tahmin edilemez bir dosya adı (örneğin, bir UUID) oluşturun. Dosyaları, web sunucusunun doğrudan erişebileceği bir genel klasör (/public veya /static) yerine, web kök dizininin dışında, yalnızca backend uygulamasının erişim iznine sahip olduğu güvenli bir klasörde depolayın. Bu, dosyaların URL üzerinden doğrudan tahmin edilerek indirilmesini engeller.
    2. **Kapsamlı Giriş Validasyonu:** Yalnızca izin verilen dosya türlerini (.pdf, .mp3, .mp4, .wav) ve makul bir dosya boyutu limitini (örneğin, 100 MB) hem frontend'de hem de **mutlaka backend'de** zorunlu kılın. Sadece frontend tarafında yapılan doğrulamalar, kötü niyetli bir kullanıcının API'ye doğrudan istek göndererek bu kontrolleri aşmasını engelleyemez. "Asla istemciye güvenme" (never trust the client) prensibi burada esastır.
    3. **Katı Erişim Kontrolü:** Bir kullanıcının, başka bir kullanıcının dosyasına erişemediğinden emin olun. İşlenmiş bir dosyayı getiren her API endpoint'i, isteği yapan kullanıcının JWT token'ından çıkarılan user\_id ile istenen dosyanın veritabanındaki user\_id'sini karşılaştırmalıdır. Eşleşme yoksa, 403 Forbidden veya 404 Not Found hatası döndürülmelidir.
    4. **Her Aşamada Şifreleme:** Uygulamayı canlıya alırken mutlaka HTTPS (SSL/TLS sertifikası ile) üzerinden sunun. Veritabanında kullanıcı şifrelerini asla düz metin olarak saklamayın; bcrypt gibi güçlü ve yavaş bir hash algoritması kullanarak geri döndürülemez şekilde hash'leyin.
* **Risk 3: Uzun Süren Arka Plan İşlemlerinin Yönetimi**
  + **Açıklama:** Büyük bir PDF'in veya bir saatlik bir ses kaydının işlenmesi dakikalar sürebilir. FastAPI'nin temel BackgroundTasks özelliği, görevi bir arka plan iş parçacığına (thread) atar ve hemen kullanıcıya yanıt döner. Bu "ateşle ve unut" (fire and forget) yaklaşımıdır ve görevin başarılı olup olmadığı, ne kadar sürdüğü veya bir hata ile karşılaşıp karşılaşmadığı hakkında bir geri bildirim mekanizması sunmaz. Kullanıcı, işlemin durumunu bilmeden sayfayı kapatabilir veya işlemin takıldığını düşünebilir.
  + **Çözüm (Pragmatik MVP Yaklaşımı):** Bu sorunu çözmek için endüstri standardı yaklaşım, Celery ve Redis/RabbitMQ gibi tam teşekküllü bir görev kuyruğu (task queue) sistemi kurmaktır. Ancak bu, bootcamp projesi için ek bir karmaşıklık ve kurulum yükü getirecektir. Bunun yerine, mevcut teknoloji yığınını kullanarak daha basit ama etkili bir çözüm uygulanabilir:
    1. **Veritabanı Tabanlı Durum Takibi:** processed\_files MongoDB koleksiyonuna bir status alanı ekleyin. Bu alan 'pending', 'processing', 'completed', 'failed' gibi durumları tutabilir.
    2. İşlem Akışı:  
       a. Kullanıcı dosyayı yüklediğinde, backend API'si veritabanına status: 'pending' olarak yeni bir doküman oluşturur ve bu dokümanın \_id'sini (task\_id olarak) kullanıcıya hemen döndürür.  
       b. Arka plan görevi (yine BackgroundTasks ile tetiklenebilir) başladığında, ilk iş olarak veritabanındaki ilgili kaydın durumunu status: 'processing' olarak günceller.  
       c. Görev başarıyla bittiğinde, durumu status: 'completed' olarak günceller ve çıktı dosyasının yolunu kaydeder. Bir hata oluşursa, durumu status: 'failed' olarak günceller ve hata mesajını loglar.
    3. **Frontend'den Periyodik Sorgulama (Polling):** Frontend, dosyayı yükledikten sonra aldığı task\_id ile belirli aralıklarla (örneğin, her 5 saniyede bir) backend'e /tasks/{task\_id}/status gibi bir endpoint'e istek atarak işlemin güncel durumunu sorgular. Gelen yanıta göre arayüzdeki ilerleme göstergesini ("İşleniyor...", "Tamamlandı" vb.) günceller. Bu yaklaşım, WebSocket gibi daha karmaşık bir teknolojiye ihtiyaç duymadan, MVP için yeterli ve sağlam bir kullanıcı deneyimi sunar. Bu, mevcut kaynaklarla en iyi çözümü üretme konusunda bilinçli bir mimari karar örneğidir.

## **7. Başarı Metrikleri**

Projenin bootcamp sonunda başarılı kabul edilmesi için aşağıdaki somut, ölçülebilir ve gerçekçi kriterler belirlenmiştir. Bu metrikler, ekibin hedeflerine ulaşıp ulaşmadığını objektif bir şekilde değerlendirmeyi sağlar.

* **Fonksiyonel Başarı:**
  + **Kriter 1:** "Mutlaka Olmalı" (Must-Have) listesinde tanımlanan özelliklerin en az %90'ının tamamlanmış ve final sunumu sırasında canlı (live demo) olarak hatasız bir şekilde çalışıyor olması.
  + **Kriter 2:** Hem görme engelli kullanıcı yolculuğunun (PDF yükleme -> Sesli içerik dinleme) hem de işitme engelli kullanıcı yolculuğunun (Ses dosyası yükleme -> Transkript okuma) baştan sona, herhangi bir kritik hata olmadan tamamlanabilmesi.
* **Performans Başarı:**
  + **Kriter 3:** 20 MB'tan küçük, 10 sayfalık, yalnızca metin içeren standart bir PDF dosyasının yüklenmesi, işlenmesi ve seslendirilmeye hazır hale gelmesi sürecinin, dosya yüklendikten sonra 60 saniyenin altında tamamlanması.
  + **Kriter 4:** 5 dakikalık, 16000 Hz örnekleme oranına sahip, net bir konuşma içeren bir MP3 dosyasının transkriptinin, dosya yüklendikten sonra 90 saniyenin altında oluşturulması.
* **Kalite ve Erişilebilirlik Başarısı:**
  + **Kriter 5:** Uygulamanın ana sayfaları (giriş, dosya yükleme, sonuç görüntüleme) üzerinde çalıştırılan Lighthouse veya Axe gibi standart bir accessibility audit (erişilebilirlik denetimi) aracında, "Accessibility" kategorisinde 90 veya üzeri bir puan alınması. Bu, teknik erişilebilirlik standartlarına uyumu gösterir.
  + **Kriter 6:** Proje ekibi dışında, potansiyel bir son kullanıcı (veya bu rolü üstlenecek bir testçi) ile yapılacak basit bir Kullanıcı Kabul Testi (UAT) senaryosunun başarıyla tamamlanması. Örneğin: "Sadece klavyeyi kullanarak sisteme giriş yap, bir PDF yükle ve üretilen sesin ilk 30 saniyesini dinle." Bu test, projenin sadece engelli kullanıcılara hizmet etmeyi amaçlamadığını, aynı zamanda kendi arayüzünün de erişilebilirlik prensiplerine gerçekten uyduğunu kanıtlar.

## 8. Infographic

‘’’HTML

<!DOCTYPE html>

<html lang="tr">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Erişilebilirlik Platformu - Proje Planı İnfografiği</title>

<script src="https://cdn.tailwindcss.com"></script>

<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Inter:wght@400;600;700;900&display=swap" rel="stylesheet">

<!-- Palette: Energetic & Playful -->

<!-- NO SVG or Mermaid JS used. Diagrams are built with HTML/CSS. -->

<style>

body {

font-family: 'Inter', sans-serif;

background-color: #2A1B3D; /\* Deep Purple from Palette \*/

color: #F1F1F1;

}

.gradient-text {

background: linear-gradient(90deg, #FF8360, #E847AE);

-webkit-background-clip: text;

-webkit-text-fill-color: transparent;

background-clip: text;

text-fill-color: transparent;

}

.card {

background-color: #3C1874; /\* Purple from Palette \*/

border: 1px solid #4A238A;

transition: transform 0.3s ease, box-shadow 0.3s ease;

}

.card:hover {

transform: translateY(-5px);

box-shadow: 0 10px 20px rgba(232, 71, 174, 0.2);

}

.chart-container {

position: relative;

margin: auto;

height: 300px;

width: 100%;

max-width: 500px;

}

@media (min-width: 768px) {

.chart-container {

height: 350px;

}

}

.timeline-item::before {

content: '';

position: absolute;

left: 20px;

top: 0;

bottom: 0;

width: 4px;

background-color: #E847AE;

}

.timeline-dot {

position: absolute;

left: 9px;

top: 0;

height: 2.5rem;

width: 2.5rem;

display: flex;

align-items: center;

justify-content: center;

background-color: #FF8360;

border-radius: 50%;

border: 4px solid #3C1874;

font-weight: 900;

}

</style>

</head>

<body class="antialiased">

<div class="container mx-auto p-4 md:p-8">

<header class="text-center my-12">

<h1 class="text-4xl md:text-6xl font-black tracking-tight leading-tight">

Erişilebilirlik Platformu <br class="hidden md:block">

<span class="gradient-text">MVP Proje Planı</span>

</h1>

<p class="mt-4 text-lg md:text-xl max-w-3xl mx-auto text-gray-300">

Eğitimde dijital bariyerleri yıkan, yapay zeka destekli bir dönüşüm projesinin stratejik yol haritası.

</p>

</header>

<main class="space-y-16">

<section id="vision">

<div class="text-center mb-12">

<h2 class="text-3xl font-bold tracking-tight">Proje Vizyonu ve Hedef Kitle</h2>

<p class="mt-2 text-gray-400 max-w-2xl mx-auto">Her öğrencinin, engel durumundan bağımsız olarak, eğitim materyallerine tam ve eşit bir şekilde erişebilmesi.</p>

</div>

<div class="grid grid-cols-1 md:grid-cols-2 gap-8">

<div class="card rounded-xl p-6">

<h3 class="font-bold text-2xl">Persona: Arda 🧑‍🎓</h3>

<p class="text-lg font-semibold gradient-text">Görme Engelli Öğrenci</p>

<p class="mt-4 text-gray-300">Ders notu PDF'lerindeki metinleri dinlemek ve görsellerin ne anlattığını anlamak istiyor. Mevcut ekran okuyucular, özellikle karmaşık akademik dokümanlarda yetersiz kalıyor.</p>

<blockquote class="mt-4 border-l-4 border-pink-500 pl-4 text-gray-200 italic">

"Bir tarih öğrencisi olarak, yüklediğim PDF'deki haritaların açıklamalarını metinle birlikte duyabilmek istiyorum."

</blockquote>

</div>

<div class="card rounded-xl p-6">

<h3 class="font-bold text-2xl">Persona: Elif 💻</h3>

<p class="text-lg font-semibold gradient-text">İşitme Engelli Öğrenci</p>

<p class="mt-4 text-gray-300">Altyazısız ders videolarının ve ses kayıtlarının içeriğini eksiksiz anlamak için konuşmaların metne dökülmüş haline ihtiyaç duyuyor.</p>

<blockquote class="mt-4 border-l-4 border-orange-500 pl-4 text-gray-200 italic">

"Bir bootcamp öğrencisi olarak, ders videosundaki konuşmaların tam metnini okuyarak karmaşık algoritmaları kaçırmadığımdan emin olmak istiyorum."

</blockquote>

</div>

</div>

</section>

<section id="features">

<div class="text-center mb-12">

<h2 class="text-3xl font-bold tracking-tight">Çözüm: MVP Özellik Akışları</h2>

<p class="mt-2 text-gray-400 max-w-2xl mx-auto">Kullanıcılarımızın temel ihtiyaçlarına odaklanan iki ana işlevi hayata geçiriyoruz.</p>

</div>

<div class="grid grid-cols-1 md:grid-cols-2 gap-8">

<div class="card rounded-xl p-6">

<h3 class="text-xl font-bold mb-4">Görme Engelliler İçin Akış</h3>

<div class="space-y-4">

<div class="flex items-center space-x-4"><span class="flex-shrink-0 bg-orange-500 text-white font-bold rounded-full h-8 w-8 flex items-center justify-center">1</span><p>Kullanıcı PDF Yükler</p></div>

<div class="ml-4 text-2xl text-pink-500">↓</div>

<div class="flex items-center space-x-4"><span class="flex-shrink-0 bg-orange-500 text-white font-bold rounded-full h-8 w-8 flex items-center justify-center">2</span><p>Sistem Metin ve Görselleri Çıkarır</p></div>

<div class="ml-4 text-2xl text-pink-500">↓</div>

<div class="flex items-center space-x-4"><span class="flex-shrink-0 bg-orange-500 text-white font-bold rounded-full h-8 w-8 flex items-center justify-center">3</span><p>AI Görseli Açıklar, Metni Sese Çevirir</p></div>

<div class="ml-4 text-2xl text-pink-500">↓</div>

<div class="flex items-center space-x-4"><span class="flex-shrink-0 bg-green-500 text-white font-bold rounded-full h-8 w-8 flex items-center justify-center">✓</span><p>Erişilebilir Ses Dosyası Hazır!</p></div>

</div>

</div>

<div class="card rounded-xl p-6">

<h3 class="text-xl font-bold mb-4">İşitme Engelliler İçin Akış</h3>

<div class="space-y-4">

<div class="flex items-center space-x-4"><span class="flex-shrink-0 bg-pink-500 text-white font-bold rounded-full h-8 w-8 flex items-center justify-center">1</span><p>Kullanıcı Ses/Video Yükler</p></div>

<div class="ml-4 text-2xl text-orange-500">↓</div>

<div class="flex items-center space-x-4"><span class="flex-shrink-0 bg-pink-500 text-white font-bold rounded-full h-8 w-8 flex items-center justify-center">2</span><p>Sistem Sesi Metne Dönüştürür</p></div>

<div class="ml-4 text-2xl text-orange-500">↓</div>

<div class="flex items-center space-x-4"><span class="flex-shrink-0 bg-pink-500 text-white font-bold rounded-full h-8 w-8 flex items-center justify-center">3</span><p>Transkript Oluşturulur</p></div>

<div class="ml-4 text-2xl text-orange-500">↓</div>

<div class="flex items-center space-x-4"><span class="flex-shrink-0 bg-green-500 text-white font-bold rounded-full h-8 w-8 flex items-center justify-center">✓</span><p>Okunabilir ve Aranabilir Metin Hazır!</p></div>

</div>

</div>

</div>

</section>

<section id="tech-stack">

<div class="text-center mb-12">

<h2 class="text-3xl font-bold tracking-tight">Teknoloji Mimarisi</h2>

<p class="mt-2 text-gray-400 max-w-2xl mx-auto">Projemizi hayata geçirmek için seçtiğimiz modern, hızlı ve ölçeklenebilir araçlar.</p>

</div>

<div class="grid grid-cols-2 md:grid-cols-3 lg:grid-cols-4 gap-6">

<div class="card text-center p-6 rounded-lg"><h4 class="font-bold text-lg">Frontend</h4><p class="text-3xl mt-2">VUE.JS</p><p class="text-sm text-gray-400 mt-1">Hızlı ve reaktif arayüzler</p></div>

<div class="card text-center p-6 rounded-lg"><h4 class="font-bold text-lg">Backend</h4><p class="text-3xl mt-2">FastAPI</p><p class="text-sm text-gray-400 mt-1">Yüksek performanslı Python API</p></div>

<div class="card text-center p-6 rounded-lg"><h4 class="font-bold text-lg">Veritabanı</h4><p class="text-3xl mt-2">MongoDB</p><p class="text-sm text-gray-400 mt-1">Esnek doküman tabanlı</p></div>

<div class="card text-center p-6 rounded-lg"><h4 class="font-bold text-lg">PDF İşleme</h4><p class="text-3xl mt-2">PyMuPDF</p><p class="text-sm text-gray-400 mt-1">Hızlı metin & resim çıkarma</p></div>

<div class="card text-center p-6 rounded-lg col-span-2 lg:col-span-2"><h4 class="font-bold text-lg">Yapay Zeka Servisleri</h4><p class="text-2xl mt-2">Google Cloud AI</p><p class="text-sm text-gray-400 mt-1">TTS, STT ve Vision API</p></div>

<div class="card text-center p-6 rounded-lg col-span-2 lg:col-span-2"><h4 class="font-bold text-lg">Dağıtım ve CI/CD</h4><p class="text-2xl mt-2">Render & GitHub Actions</p><p class="text-sm text-gray-400 mt-1">Otomatik test ve dağıtım</p></div>

</div>

</section>

<section id="roadmap">

<div class="text-center mb-12">

<h2 class="text-3xl font-bold tracking-tight">Proje Yol Haritası (6 Hafta)</h2>

<p class="mt-2 text-gray-400 max-w-2xl mx-auto">Projemizi 2 haftalık 3 sprint'e bölerek hedeflerimize adım adım ilerliyoruz.</p>

</div>

<div class="relative pl-8">

<div class="space-y-12">

<div class="relative timeline-item pl-8">

<div class="timeline-dot">S1</div>

<h3 class="text-xl font-bold gradient-text">Sprint 1: Altyapı ve İlk Çekirdek Özellik</h3>

<p class="text-gray-300">Temel altyapıyı kuruyor ve ilk somut çıktıyı hedefliyoruz: PDF'ten sese dönüşüm.</p>

<ul class="list-disc list-inside mt-2 text-gray-400">

<li>Backend/Frontend proje iskeletleri</li>

<li>Kullanıcı kayıt ve giriş sistemi</li>

<li>PDF yükleme ve metin çıkarma</li>

<li>Metinden Sese (TTS) API entegrasyonu</li>

</ul>

</div>

<div class="relative timeline-item pl-8">

<div class="timeline-dot">S2</div>

<h3 class="text-xl font-bold gradient-text">Sprint 2: İkinci Çekirdek Özellik ve Arayüz</h3>

<p class="text-gray-300">İşitme engelli kullanıcılar için değeri sunuyor ve kullanıcı deneyimini iyileştiriyoruz.</p>

<ul class="list-disc list-inside mt-2 text-gray-400">

<li>Ses/Video yükleme özelliği</li>

<li>Sesten Metne (STT) API entegrasyonu</li>

<li>Transkript görüntüleme ve arama</li>

<li>Dosya işleme durum göstergesi</li>

</ul>

</div>

<div class="relative timeline-item pl-8">

<div class="timeline-dot">S3</div>

<h3 class="text-xl font-bold gradient-text">Sprint 3: Geliştirme, Test ve Sunum</h3>

<p class="text-gray-300">Tüm özellikleri birleştiriyor, hataları gideriyor, erişilebilirliği en üst düzeye çıkarıyoruz.</p>

<ul class="list-disc list-inside mt-2 text-gray-400">

<li>Görselden metin betimleme (Vision AI)</li>

<li>Kapsamlı hata yönetimi ve güvenlik</li>

<li>Erişilebilirlik testleri (WCAG)</li>

<li>Final sunumuna hazırlık</li>

</ul>

</div>

</div>

</div>

</section>

<section id="metrics">

<div class="text-center mb-12">

<h2 class="text-3xl font-bold tracking-tight">Başarı Metriklerimiz</h2>

<p class="mt-2 text-gray-400 max-w-2xl mx-auto">Projenin başarısını somut ve ölçülebilir hedeflerle tanımlıyoruz.</p>

</div>

<div class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-8">

<div class="card rounded-xl p-6 flex flex-col items-center">

<h3 class="text-xl font-bold text-center">MVP Fonksiyonellik</h3>

<div class="chart-container flex-grow">

<canvas id="featureCompletionChart"></canvas>

</div>

<p class="mt-2 text-center text-gray-300">Temel özelliklerin en az %90'ının demo gününde hatasız çalışması hedefleniyor.</p>

</div>

<div class="card rounded-xl p-6 flex flex-col items-center">

<h3 class="text-xl font-bold text-center">Performans Hedefleri</h3>

<div class="chart-container flex-grow">

<canvas id="performanceChart"></canvas>

</div>

<p class="mt-2 text-center text-gray-300">PDF ve ses dosyalarının belirlenen süreler altında işlenmesi kritik.</p>

</div>

<div class="card rounded-xl p-6 flex flex-col items-center justify-center">

<h3 class="text-xl font-bold text-center">Erişilebilirlik Skoru</h3>

<div class="my-6">

<p class="text-8xl font-black gradient-text">90+</p>

</div>

<p class="mt-2 text-center text-gray-300">Lighthouse denetimlerinde 90 üzeri bir erişilebilirlik puanı almayı taahhüt ediyoruz.</p>

</div>

</div>

</section>

</main>

<footer class="text-center mt-16 py-8 border-t border-gray-700">

<p class="text-gray-400">Bu infografik, bir bootcamp bitirme projesi için oluşturulmuş MVP planını görselleştirmektedir.</p>

</footer>

</div>

<script>

document.addEventListener('DOMContentLoaded', function () {

const palette = {

orange: '#FF8360',

pink: '#E847AE',

purple: '#3C1874',

deepPurple: '#2A1B3D',

lightText: '#F1F1F1',

grayText: '#A0A0A0'

};

const chartDefaultOptions = {

responsive: true,

maintainAspectRatio: false,

plugins: {

legend: {

labels: {

color: palette.lightText,

font: {

size: 14

}

}

},

tooltip: {

callbacks: {

title: function(tooltipItems) {

const item = tooltipItems[0];

let label = item.chart.data.labels[item.dataIndex];

if (Array.isArray(label)) {

return label.join(' ');

} else {

return label;

}

}

}

}

}

};

function wrapLabel(label, maxLength = 16) {

if (label.length <= maxLength) {

return label;

}

const words = label.split(' ');

const lines = [];

let currentLine = '';

for (const word of words) {

if ((currentLine + ' ' + word).trim().length > maxLength) {

lines.push(currentLine.trim());

currentLine = word;

} else {

currentLine = (currentLine + ' ' + word).trim();

}

}

if (currentLine) {

lines.push(currentLine.trim());

}

return lines;

}

const featureCtx = document.getElementById('featureCompletionChart');

if(featureCtx) {

new Chart(featureCtx, {

type: 'doughnut',

data: {

labels: ['Tamamlanan Özellikler', 'Kalan'],

datasets: [{

label: 'MVP Fonksiyonellik',

data: [90, 10],

backgroundColor: [palette.pink, palette.purple],

borderColor: [palette.deepPurple],

borderWidth: 4,

hoverOffset: 4

}]

},

options: {

...chartDefaultOptions,

cutout: '70%',

}

});

}

const performanceCtx = document.getElementById('performanceChart');

if(performanceCtx) {

new Chart(performanceCtx, {

type: 'bar',

data: {

labels: [wrapLabel('PDF İşleme Süresi (sn)'), wrapLabel('Ses İşleme Süresi (sn)')],

datasets: [{

label: 'Hedef Süre (sn)',

data: [60, 90],

backgroundColor: palette.orange,

borderColor: palette.orange,

borderWidth: 1

}]

},

options: {

...chartDefaultOptions,

scales: {

y: {

beginAtZero: true,

grid: {

color: 'rgba(255, 255, 255, 0.1)'

},

ticks: {

color: palette.lightText

}

},

x: {

grid: {

display: false

},

ticks: {

color: palette.lightText

}

}

}

}

});

}

});

</script>

</body>

</html>

‘’’

## 9. Web Page

‘’’ HTML

<!DOCTYPE html>

<html lang="tr" class="scroll-smooth">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Erişilebilirlik Platformu - İnteraktif Proje Planı</title>

<script src="https://cdn.tailwindcss.com"></script>

<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Inter:wght@400;500;600;700;800&display=swap" rel="stylesheet">

<!-- Chosen Palette: Soothing Neutrals & Warm Accents (Customized) -->

<!-- Application Structure Plan: The SPA is structured as a narrative journey that guides the user from the high-level vision to the concrete details of execution. It starts with the 'Why' (Vision & Personas), moves to the 'What' (Features), then the 'How' (Tech Stack), the 'When' (Roadmap), anticipates 'What If' (Risks), and finally measures 'How Well' (Success Metrics). This top-down, story-driven approach is more engaging than a simple report and makes complex information digestible by revealing details interactively (e.g., clicking on roadmap sprints or risk items). The navigation bar reinforces this structure, allowing non-linear exploration. -->

<!-- Visualization & Content Choices:

- Vision: Goal: Inform. Method: Large typography and impactful text. Justification: Sets a strong, clear tone for the project's purpose.

- Personas: Goal: Inform/Organize. Method: Two-column card layout with Unicode icons. Justification: Clearly separates and defines the target users, fostering empathy.

- Features: Goal: Organize/Compare. Method: Interactive tabbed interface for 'Must-Have' vs 'Nice-to-Have'. Justification: Organizes a large list of features into manageable, prioritized chunks, which is more user-friendly than a long table.

- Tech Stack: Goal: Inform/Organize. Method: Grid of cards with HTML/CSS based icons. Justification: Visually represents the technology choices without using external images/SVG, making it clean and fast-loading.

- Roadmap: Goal: Change/Organize. Method: Interactive vertical timeline built with HTML/Tailwind. Justification: Clearly visualizes the project's phased approach over time. Click interaction reveals details for each sprint, preventing information overload.

- Risks: Goal: Organize/Inform. Method: Accordion component built with HTML/JS. Justification: Presents potential problems and solutions compactly, allowing users to focus on one risk at a time.

- Metrics: Goal: Compare/Inform. Method: Chart.js Donut and Bar charts, and a large stat display. Justification: Transforms abstract numbers into compelling, easy-to-grasp visuals that clearly define what success looks like.

-->

<!-- CONFIRMATION: NO SVG graphics used. NO Mermaid JS used. -->

<style>

body {

font-family: 'Inter', sans-serif;

background-color: #f8f7f4; /\* Warm Neutral Background \*/

}

.text-brand-primary { color: #2c3e50; }

.text-brand-secondary { color: #34495e; }

.bg-brand-accent { background-color: #e67e22; }

.border-brand-accent { border-color: #e67e22; }

.card {

background-color: #ffffff;

border: 1px solid #e0e0e0;

transition: transform 0.2s ease-in-out, box-shadow 0.2s ease-in-out;

}

.card:hover {

transform: translateY(-4px);

box-shadow: 0 10px 15px -3px rgb(0 0 0 / 0.05), 0 4px 6px -4px rgb(0 0 0 / 0.05);

}

.chart-container {

position: relative;

width: 100%;

max-width: 400px;

margin-left: auto;

margin-right: auto;

height: 250px;

max-height: 300px;

}

@media (min-width: 768px) {

.chart-container { height: 300px; }

}

.accordion-content {

max-height: 0;

overflow: hidden;

transition: max-height 0.3s ease-in-out;

}

</style>

</head>

<body class="text-brand-secondary">

<nav class="bg-white/80 backdrop-blur-md sticky top-0 z-50 border-b border-gray-200">

<div class="max-w-7xl mx-auto px-4 sm:px-6 lg:px-8">

<div class="flex items-center justify-between h-16">

<div class="flex-shrink-0">

<span class="font-bold text-lg text-brand-primary">Proje Planı</span>

</div>

<div class="hidden md:block">

<div class="ml-10 flex items-baseline space-x-4">

<a href="#vision" class="px-3 py-2 rounded-md text-sm font-medium hover:bg-gray-200">Vizyon</a>

<a href="#features" class="px-3 py-2 rounded-md text-sm font-medium hover:bg-gray-200">Özellikler</a>

<a href="#tech-stack" class="px-3 py-2 rounded-md text-sm font-medium hover:bg-gray-200">Teknoloji</a>

<a href="#roadmap" class="px-3 py-2 rounded-md text-sm font-medium hover:bg-gray-200">Yol Haritası</a>

<a href="#risks" class="px-3 py-2 rounded-md text-sm font-medium hover:bg-gray-200">Riskler</a>

<a href="#metrics" class="px-3 py-2 rounded-md text-sm font-medium hover:bg-gray-200">Metrikler</a>

</div>

</div>

</div>

</div>

</nav>

<div class="max-w-7xl mx-auto px-4 sm:px-6 lg:px-8 py-12">

<!-- Section 1: Vision & Scope -->

<section id="vision" class="text-center mb-24">

<h1 class="text-4xl md:text-5xl font-extrabold text-brand-primary tracking-tight">Erişilebilirlik Platformu</h1>

<p class="mt-4 max-w-3xl mx-auto text-lg md:text-xl text-brand-secondary">

Eğitimde dijital bariyerleri yıkan, yapay zeka destekli bir dönüşüm projesinin stratejik ve interaktif yol haritası.

</p>

<div class="mt-12 grid grid-cols-1 md:grid-cols-2 gap-8 text-left">

<div class="card p-6 rounded-lg">

<h3 class="text-2xl font-bold text-brand-primary flex items-center">

<span class="text-3xl mr-3">👁️</span> Arda (Görme Engelli)

</h3>

<p class="mt-3">"PDF'teki ders notlarımı dinlemek ve içindeki görsellerin ne anlattığını anlamak istiyorum. Mevcut çözümler, özellikle akademik içeriklerde yetersiz kalıyor."</p>

</div>

<div class="card p-6 rounded-lg">

<h3 class="text-2xl font-bold text-brand-primary flex items-center">

<span class="text-3xl mr-3">👂</span> Elif (İşitme Engelli)

</h3>

<p class="mt-3">"Altyazısı olmayan ders videolarının içeriğini kaçırmamak için konuşmaların tam metnine ihtiyacım var. Bu, teknik terimleri doğru öğrenmem için kritik."</p>

</div>

</div>

</section>

<!-- Section 2: Features -->

<section id="features" class="mb-24">

<div class="text-center mb-12">

<h2 class="text-3xl font-bold text-brand-primary">MVP Özellikleri</h2>

<p class="mt-2 max-w-2xl mx-auto">Projenin temelini oluşturan ve kullanıcılarımıza en yüksek değeri sunacak özellikler.</p>

</div>

<div>

<div class="flex justify-center border-b border-gray-300 mb-6">

<button id="must-have-btn" class="feature-toggle-btn px-6 py-3 border-b-2 border-brand-accent text-brand-primary font-semibold">Mutlaka Olmalı</button>

<button id="nice-to-have-btn" class="feature-toggle-btn px-6 py-3 border-b-2 border-transparent text-gray-500 font-medium">Olsa İyi Olur</button>

</div>

<div id="features-container" class="grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-3 gap-6">

<!-- Feature cards will be dynamically inserted here -->

</div>

</div>

</section>

<!-- Section 3: Tech Stack -->

<section id="tech-stack" class="mb-24">

<div class="text-center mb-12">

<h2 class="text-3xl font-bold text-brand-primary">Teknoloji Mimarisi</h2>

<p class="mt-2 max-w-2xl mx-auto">Projeyi hayata geçirmek için seçilen modern, hızlı ve ölçeklenebilir araçlar.</p>

</div>

<div class="grid grid-cols-2 md:grid-cols-3 lg:grid-cols-4 gap-6">

<div class="card p-6 text-center"><h3 class="font-bold text-lg">Frontend</h3><p class="text-2xl font-semibold mt-2">Vue.js</p><p class="text-sm text-gray-500 mt-1">Hızlı ve reaktif arayüzler</p></div>

<div class="card p-6 text-center"><h3 class="font-bold text-lg">Backend</h3><p class="text-2xl font-semibold mt-2">FastAPI</p><p class="text-sm text-gray-500 mt-1">Yüksek performanslı Python API</p></div>

<div class="card p-6 text-center"><h3 class="font-bold text-lg">Veritabanı</h3><p class="text-2xl font-semibold mt-2">MongoDB</p><p class="text-sm text-gray-500 mt-1">Esnek doküman tabanlı</p></div>

<div class="card p-6 text-center"><h3 class="font-bold text-lg">AI Servisleri</h3><p class="text-2xl font-semibold mt-2">Google AI</p><p class="text-sm text-gray-500 mt-1">TTS, STT, Vision</p></div>

</div>

</section>

<!-- Section 4: Roadmap -->

<section id="roadmap" class="mb-24">

<div class="text-center mb-12">

<h2 class="text-3xl font-bold text-brand-primary">6 Haftalık Proje Yol Haritası</h2>

<p class="mt-2 max-w-2xl mx-auto">Hedeflerimize 2 haftalık sprintlerle adım adım ilerliyoruz. Detaylar için başlıklara tıklayın.</p>

</div>

<div class="relative max-w-2xl mx-auto">

<!-- Vertical Line -->

<div class="absolute left-4 top-0 h-full w-0.5 bg-gray-300"></div>

<!-- Sprint Items -->

<div id="sprint-container" class="space-y-12">

<!-- Sprints will be dynamically inserted here -->

</div>

</div>

</section>

<!-- Section 5: Risks -->

<section id="risks" class="mb-24">

<div class="text-center mb-12">

<h2 class="text-3xl font-bold text-brand-primary">Potansiyel Riskler ve Çözümler</h2>

<p class="mt-2 max-w-2xl mx-auto">Projenin başarısını garantilemek için öngördüğümüz riskler ve proaktif önlemlerimiz.</p>

</div>

<div id="risk-accordion" class="max-w-3xl mx-auto space-y-4">

<!-- Risks will be dynamically inserted here -->

</div>

</section>

<!-- Section 6: Success Metrics -->

<section id="metrics" class="mb-24">

<div class="text-center mb-12">

<h2 class="text-3xl font-bold text-brand-primary">Başarı Metrikleri</h2>

<p class="mt-2 max-w-2xl mx-auto">Projenin başarısını bu somut ve ölçülebilir hedeflerle tanımlıyoruz.</p>

</div>

<div class="grid grid-cols-1 md:grid-cols-3 gap-8">

<div class="card rounded-lg p-6 flex flex-col items-center text-center">

<h3 class="text-xl font-bold text-brand-primary mb-4">Fonksiyonel Başarı</h3>

<div class="chart-container">

<canvas id="featureChart"></canvas>

</div>

<p class="mt-4 text-sm">"Mutlaka Olmalı" özelliklerinin en az %90'ı tamamlanacak.</p>

</div>

<div class="card rounded-lg p-6 flex flex-col items-center text-center">

<h3 class="text-xl font-bold text-brand-primary mb-4">Performans Hedefleri</h3>

<div class="chart-container">

<canvas id="performanceChart"></canvas>

</div>

<p class="mt-4 text-sm">Temel işlemler belirlenen sürelerin altında tamamlanacak.</p>

</div>

<div class="card rounded-lg p-6 flex flex-col items-center justify-center text-center">

<h3 class="text-xl font-bold text-brand-primary">Erişilebilirlik Skoru</h3>

<div class="my-6 text-7xl font-extrabold bg-brand-accent text-white rounded-full h-40 w-40 flex items-center justify-center">

90+

</div>

<p class="mt-4 text-sm">Lighthouse denetimlerinde 90 üzeri puan hedefleniyor.</p>

</div>

</div>

</section>

</div>

<footer class="bg-white border-t border-gray-200">

<div class="max-w-7xl mx-auto py-6 px-4 sm:px-6 lg:px-8 text-center text-gray-500">

<p>&copy; 2025 Erişilebilirlik Platformu Proje Ekibi. Bu interaktif plan, bir bootcamp projesi için oluşturulmuştur.</p>

</div>

</footer>

<script>

document.addEventListener('DOMContentLoaded', () => {

const data = {

features: {

mustHave: [

{ title: "Kullanıcı Yönetimi", description: "Güvenli kayıt ve giriş sistemi." },

{ title: "PDF Yükleme", description: "Erişilebilir PDF yükleme arayüzü." },

{ title: "Metin Çıkarma (OCR)", description: "PDF'ten yapısal metin çıkarma." },

{ title: "Metinden Sese (TTS)", description: "Akıcı ve doğal seslendirme." },

{ title: "Ses Oynatma/İndirme", description: "Erişilebilir oynatıcı ve indirme." },

{ title: "Görsel Açıklama", description: "Görseller için otomatik alt metin." },

{ title: "Ses/Video Yükleme", description: "MP3, MP4, WAV desteği." },

{ title: "Sesten Metne (STT)", description: "Konuşmaların transkripsiyonu." },

{ title: "Transkript Arama", description: "Metin içinde anahtar kelime arama." },

{ title: "Durum Bildirimi", description: "Dosya işleme için ilerleme takibi." },

{ title: "Erişilebilir Tasarım", description: "WCAG AA standartlarına uyum." }

],

niceToHave: [

{ title: "Gelişmiş Ses Kontrolü", description: "Ses hızı ve perdesi ayarlama." },

{ title: "Altyazı Oluşturma", description: ".srt/.vtt formatında indirme." },

{ title: "Konuşmacı Tespiti", description: "Diyaloglarda konuşmacıları etiketleme." },

{ title: "Anlık Durum Takibi", description: "WebSocket ile anlık bildirimler." },

{ title: "Otomatik Özetleme", description: "Uzun metinler için özet çıkarma." },

{ title: "İşaret Dili Desteği", description: "MVP sonrası için uzun vadeli hedef." }

]

},

roadmap: [

{

sprint: 1,

title: "Temel Altyapı ve İlk Çekirdek Özellik",

details: "Proje iskeletini kurmak ve ilk somut, uçtan uca çalışan özelliği (PDF'ten sese) tamamlamak. Kullanıcı giriş yapıp bir PDF yükleyebilmeli ve karşılığında bir ses dosyası alabilmelidir."

},

{

sprint: 2,

title: "İkinci Çekirdek Özellik ve Arayüz Geliştirmeleri",

details: "İkinci ana kullanıcı yolculuğunu (sesten metne) tamamlamak ve kullanıcı arayüzünü daha işlevsel ve kullanıcı dostu hale getirmek. Transkript arama ve ilerleme göstergesi eklenecek."

},

{

sprint: 3,

title: "Geliştirme, Test, Erişilebilirlik ve Sunum",

details: "MVP kapsamındaki tüm özellikleri entegre edip tamamlamak, uygulamanın kararlılığını ve en önemlisi erişilebilirliğini en üst düzeye çıkarmak. Vision AI entegrasyonu ve kapsamlı testler bu sprint'in odağıdır."

}

],

risks: [

{ title: "Üçüncü Parti API Bağımlılığı", solution: "Soyutlama katmanı oluşturmak, API kullanımını düzenli takip etmek ve bütçe uyarıları kurmak. Bu, sağlayıcıya olan bağımlılığı azaltır ve maliyetleri kontrol altında tutar." },

{ title: "Kullanıcı Verilerinin Güvenliği", solution: "Güvenli dosya yükleme protokolleri uygulamak, kapsamlı giriş doğrulaması yapmak ve katı erişim kontrolü kuralları belirlemek. Veriler her aşamada şifrelenmelidir." },

{ title: "Uzun Süren Arka Plan İşlemleri", solution: "Veritabanı tabanlı durum takibi ve frontend'den periyodik sorgulama (polling) mekanizması kurmak. Bu, karmaşık görev kuyruğu sistemlerine MVP aşamasında basit bir alternatiftir." }

]

};

// Features Logic

const featuresContainer = document.getElementById('features-container');

const mustHaveBtn = document.getElementById('must-have-btn');

const niceToHaveBtn = document.getElementById('nice-to-have-btn');

function renderFeatures(type) {

featuresContainer.innerHTML = '';

const features = data.features[type];

features.forEach(feature => {

const card = document.createElement('div');

card.className = 'card rounded-lg p-5';

card.innerHTML = `<h4 class="font-bold text-brand-primary">${feature.title}</h4><p class="text-sm mt-1">${feature.description}</p>`;

featuresContainer.appendChild(card);

});

}

mustHaveBtn.addEventListener('click', () => {

mustHaveBtn.classList.add('border-brand-accent', 'text-brand-primary', 'font-semibold');

mustHaveBtn.classList.remove('border-transparent', 'text-gray-500', 'font-medium');

niceToHaveBtn.classList.add('border-transparent', 'text-gray-500', 'font-medium');

niceToHaveBtn.classList.remove('border-brand-accent', 'text-brand-primary', 'font-semibold');

renderFeatures('mustHave');

});

niceToHaveBtn.addEventListener('click', () => {

niceToHaveBtn.classList.add('border-brand-accent', 'text-brand-primary', 'font-semibold');

niceToHaveBtn.classList.remove('border-transparent', 'text-gray-500', 'font-medium');

mustHaveBtn.classList.add('border-transparent', 'text-gray-500', 'font-medium');

mustHaveBtn.classList.remove('border-brand-accent', 'text-brand-primary', 'font-semibold');

renderFeatures('niceToHave');

});

renderFeatures('mustHave'); // Initial render

// Roadmap Logic

const sprintContainer = document.getElementById('sprint-container');

data.roadmap.forEach(sprint => {

const item = document.createElement('div');

item.className = 'relative pl-10';

item.innerHTML = `

<div class="absolute left-0 top-0 transform -translate-x-1/2 -translate-y-1/2">

<div class="w-8 h-8 bg-brand-accent rounded-full text-white flex items-center justify-center font-bold text-sm">${sprint.sprint}</div>

</div>

<div class="card rounded-lg p-5 cursor-pointer roadmap-item">

<h4 class="font-bold text-lg text-brand-primary">${sprint.title}</h4>

<p class="text-sm mt-2 hidden">${sprint.details}</p>

</div>`;

sprintContainer.appendChild(item);

});

document.querySelectorAll('.roadmap-item').forEach(item => {

item.addEventListener('click', () => {

item.querySelector('p').classList.toggle('hidden');

});

});

// Risks Accordion Logic

const riskAccordion = document.getElementById('risk-accordion');

data.risks.forEach((risk, index) => {

const item = document.createElement('div');

item.className = 'card rounded-lg overflow-hidden';

item.innerHTML = `

<button class="w-full text-left p-5 flex justify-between items-center accordion-toggle">

<span class="font-bold text-brand-primary">${risk.title}</span>

<svg class="w-5 h-5 transition-transform" fill="none" stroke="currentColor" viewBox="0 0 24 24"><path stroke-linecap="round" stroke-linejoin="round" stroke-width="2" d="M19 9l-7 7-7-7"></path></svg>

</button>

<div class="accordion-content">

<div class="p-5 pt-0 text-gray-600">

${risk.solution}

</div>

</div>`;

riskAccordion.appendChild(item);

});

riskAccordion.addEventListener('click', (e) => {

const toggle = e.target.closest('.accordion-toggle');

if (toggle) {

const content = toggle.nextElementSibling;

const icon = toggle.querySelector('svg');

icon.classList.toggle('rotate-180');

if (content.style.maxHeight) {

content.style.maxHeight = null;

} else {

content.style.maxHeight = content.scrollHeight + "px";

}

}

});

// Chart.js Implementations

const chartDefaultOptions = {

responsive: true,

maintainAspectRatio: false,

plugins: {

legend: { display: false },

tooltip: {

callbacks: {

title: (tooltipItems) => {

let label = tooltipItems[0].label;

if (Array.isArray(label)) return label.join(' ');

return label;

}

}

}

}

};

const featureChartCtx = document.getElementById('featureChart').getContext('2d');

new Chart(featureChartCtx, {

type: 'doughnut',

data: {

labels: ['Tamamlandı', 'Kalan'],

datasets: [{

data: [90, 10],

backgroundColor: ['#e67e22', '#ecf0f1'],

borderColor: '#f8f7f4',

borderWidth: 4,

hoverOffset: 4

}]

},

options: { ...chartDefaultOptions, cutout: '75%' }

});

const performanceChartCtx = document.getElementById('performanceChart').getContext('2d');

new Chart(performanceChartCtx, {

type: 'bar',

data: {

labels: ['PDF İşleme', 'Ses İşleme'],

datasets: [{

label: 'Hedef Süre (sn)',

data: [60, 90],

backgroundColor: ['#e67e22', '#f39c12'],

borderRadius: 4

}]

},

options: {

...chartDefaultOptions,

scales: {

y: {

beginAtZero: true,

grid: { color: '#ecf0f1' },

ticks: { color: '#34495e' }

},

x: {

grid: { display: false },

ticks: { color: '#34495e' }

}

}

}

});

});

</script>

</body>

</html>

‘’’