

## Proje-1 Final Sınavı Raporu

TESTİFY

Öğrenci Ad Soyad: NECATİ ARMAN

Öğrenci Numarası: 2312903063

# İçindekiler

İçindekiler.....	1
Özet .....	4
Giriş.....	5
Amaç ve Kapsam.....	6
Amaç: .....	6
Kapsam:.....	6
• Kullanıcı Girişi ve Kayıt İşlemleri .....	6
• Şifre Yenileme.....	6
• Soru Ekleme .....	6
• Sınav Sonuçlarını Anlık Görüntüleme .....	6
• Sınav Yönetimi .....	6
• Sosyal Medya Paylaşımı .....	6
Kapsam Dışı: .....	7
Kullanılan Teknolojiler ve Araçlar .....	7
Kullanılan Teknolojiler:.....	7
HTML, CSS, JavaScript.....	7
PHP .....	7
Laravel.....	7
React.....	7
MatX Teması .....	7
JWT (Json Web Token).....	7
Geliştirme Araçları: .....	8
Visual Studio.....	8
Git .....	8
Github .....	8
Notion .....	8
Postman .....	8
Veri Tabanı Sistemleri .....	8
MySQL .....	8
Oracle.....	8
Tasarım ve Geliştirilme Süreci .....	8
Tasarım Süreci.....	8

Geliştirilme Süreci.....	9
Geliştirme Sürecinde Karşılaşılan Zorluklar .....	9
Test Süreçleri.....	9
Sürüm Kontrol Sistemi.....	9
Fonksiyon ve Özellikler .....	9
Kullanıcı Giriş ve Kayıt Olma İşlemleri .....	10
Kayıt Olma .....	10
Giriş Yapma .....	10
Şifre Sıfırlama.....	10
Öğretmen Özellikleri.....	10
Soru Ekleme:.....	10
Sınav Sonuçlarını Görüntüleme.....	10
Sınav Kağıtlarını Görüntüleme.....	10
Öğrenci Özellikleri:.....	10
Sınava Katılma .....	10
Sınav Sonuçlarını Öğrenme.....	10
Sınav Kağıtlarını Görüntüleme.....	10
Soru Tipleri ve Yönetimi:.....	11
Desteklenen Soru Tipleri.....	11
Güvenlik Özellikleri:.....	11
Kimlik Doğrulama ve Yetkilendirme .....	11
Ekstra Özellikler:.....	11
Resimli Soru Ekleme.....	11
Sınav Kağıtların ve Sınava Giren Kişilerin Listesini Yazdırma .....	11
Sınav Paylaşımı .....	11
Karşılaşılan Sorunlar ve Çözümler.....	11
Fotoğrafların Sunucuya Ulaşmama Sorunu: .....	11
Sorun.....	11
Çözüm.....	11
Sunucu Tarafı Fotoğraf Gönderme Sorunu .....	12
Sorun.....	12
Çözüm.....	12
Sonuç ve Gelecek Çalışmalar .....	12
Sonuç.....	12
Gelecek Çalışmalar.....	12
Yeni Teknolojik Araçlar .....	12

Soru Havuzu ve Soru Seçme .....	12
Sınav Süresi Yönetimi .....	13
Gelişmiş Yazılım Entegrasyonu .....	13
Server Side Rendering (SSR).....	13
Kullanıcı Dostu Özellikleri .....	13
Kaynakça .....	13

## Özet

Testify, öğretmenlerin soru yüklediği öğrencilerin bu soruları çözebildiği online bir sınav sistemidir. Proje dört farklı soru tipini desteklemektedir bunlar 1. Açık uçlu 2. Çoktan seçmeli 3. Birden Fazla Cevap 4. Boşluk doldurmadır. Sistemimizde bulunan Açık uçlu soru tipinin kontrolleri OpenAI firmasının ChatGPT4o dil modelini kullanarak sağlanmaktadır. Testify projesinde HTML, CSS, JavaScript, PHP, Laravel, React ve Git gibi modern web teknolojilerini kullanarak geliştirilmiştir. Proje öğretmenler ve öğrencilerin kullanımı için tasarlanmış olup, kullanıcı dostu ve kolay kullanılabilir bir arayüze sahiptir ve eğitim sürecini daha etkili hale getirmeyi ve sınav süreçlerini hızlandırmayı amaçlamaktadır.

Proje tamamlanmış olup, gelecekte sınav süresi yönetimi, soru havuzundan soru seçme, daha gelişmiş yazılım çözümleri ve server-side-rendering gibi ek özellikler ile geliştirilmeye açık bir projedir. Testify Öğretmenler ve Öğrenciler arasında daha etkileşimli ve verimli bir eğitim süreci sunmayı amaçlamaktadır.

## Giriş

Proje “Proje-1” dersi kapsamında geliştirilmiş olup, mevcut sistemlerden farklı olarak kolay kullanımı, yenilikçi UI/UX tasarımı, yapay zeka destekli olması ve yeni teknolojiler kullanılarak yazılmış olması mevcut sistemlerden öne çıkaran yönleridir. Bu proje, öğretmenlerin ve öğrencilerin sınav süreçlerini daha verimli ve etkili bir şekilde yönetmelerine olanak sağlamayı amaçlamaktadır.

Testify projesinin ana hedefleri arasında öğretmenlerin sınav okuma süreçlerini kısaltmak, sınav sorusu oluşturma sürecini hızlandırmak ve öğrencilerin sınav sonuçlarını daha kolay ve hızlı bir şekilde öğrenmelerini sağlamaktır. Öğrenciler, sınav kağıtlarını sistem üzerinden tek tık ile görüntüleyebilir ve sonuçlarını hızlıca öğrenebilirler. Bu özellikler sayesinde, eğitim sürecinin her iki taraf içinde akıcı, verimli ve hızlı bir hale getirmektedir.

Proje, bireysel olarak Necati ARMAN tarafından geliştirilmiştir. Testify dört farklı soru tipini desteklemesi ve yapay zeka çözümleriyle diğer projelerden öne çıkmaktadır. OpenAI’nin ChatGPT4o dil modeli kullanılarak açık uçlu soruların değerlendirilmesi sağlanmakta ve bu sayede öğretmenlerin açık uçlu soruların değerlendirilme yükünü önemli bir ölçüde azaltmaktadır.

Proje, dört farklı soru tipini desteklemesini yanında görsel tabanlı soru sorma sistemini de desteklemektedir bu sayede her soruya ve her şıkka fotoğraf ekleyerek resim/görsel tabanlı sormanız gereken soruları da kolay ve rahat bir şekilde sistemimize ekleyip sorabilirsiniz.

Testify, yenilikçi sistemler ve yapay zeka destekli çözümleri ile benzer projelerden ayrılmaktadır. Proje hem öğrenciler hem de öğretmenler için eğitim deneyimini daha interaktif ve verimli hale getirmeyi hedeflemektedir. Mevcut haliyle tamamlanmış olan bu proje, gelecekte sınav süresi yönetimi, soru havuzundan soru seçme, daha gelişmiş yazılım çözümleri ve server-side rendering gibi ek özelliklerle geliştirilmeye açıktır.

# Amaç ve Kapsam

## Amaç:

Testify projesi, eğitim sürecini hem öğretmenler hem de öğrenciler için daha verimli ve etkili hale getirmeyi amaçlamaktadır.

Projenin ana hedefleri şunlardır:

- Öğretmenlerin sınav okuma ve değerlendirme süreçlerini kısaltmak,
- Sınav sorusu oluşturma sürecini hızlandırmak,
- Öğrencilerin sınav notlarını kolayca öğrenmelerini sağlamak,
- Öğrencilerin sınav kağıtlarını anında görüntülemelerine olanak tanımak bulunmaktadır

Bu amaçlarda, modern teknolojilerini ve yenilikçi UI/UX tasarımı ile kullanıcı dostu bir sistem sunmaktadır.

## Kapsam:

Testify, kapsamlı bir web tabanlı sınav yönetim sistemidir ve aşağıdaki özellikleri içermektedir

- **Kullanıcı Girişi ve Kayıt İşlemleri:** Öğretmen, öğrenci, admin girişi aynı sayfadan yapılmaktadır, yetkiler sayesinde farklı panellere yönlendirmeleri sağlanmaktadır. Kullanıcı kaydolma kısımları aynıdır. Öğretmen ve öğrenci seçenekleri seçilerek seçilen rol de hesap oluşturulur. Admin kaydı hiçbir şekilde bulunmamaktadır. Sunucu kurulurken atanır ve değiştirilmez yeniden eklenemez.
- **Şifre Yenileme:** Sistemdeki kullanıcılar “şifremi unuttum” özelliği ile şifrelerini istedikleri zaman e-posta adreslerin gelen link ile şifrelerini yenileyebilirler. Gönderilen linkleri kullanıcılar sadece 10 dakika içinde kullanmalı ve linkler tek kullanımlık olarak gönderilmektedir.
- **Soru Ekleme:** Öğretmenler, sistem üzerinde hem resimli hem de normal resimsiz soru ekleyebilir ve dört farklı soru tipini (açık uçlu, çoktan seçmeli, birden fazla cevap, boşluk doldurma) desteklemektedir.
- **Sınav Sonuçlarını Anlık Görüntüleme:** Öğrenciler sınav sonunda, sınav sonuçlarını hemen öğrenebilir ve sınav kağıtlarını anlık olarak görüntüleyebilir. Öğretmenler de kendi testlerini çözen öğrencilerin sınavdan kaç aldıklarını ve sınav kağıtlarını inceleyebilirler.
- **Sınav Yönetimi:** Öğretmenler, sınav kağıtlarını ve sınava giren kişilerin listesini çıktı olarak alabilirler.
- **Sosyal Medya Paylaşımı:** Öğretmenler sorulardaki “Paylaş” butonu ile oluşturdukları testlerini “WhatsApp, LinkedIn, E-Posta, Twitter” sosyal medya platformlarında rahatlık ve kolaylıkla paylaşabilirler.

Projem, öğretmen ve öğrencilerin bulunduğu tüm okullarda kullanılabilir. Bu sayede, geniş bir kullanıcı kitlesine hitap etmektedir ve eğitim süreçlerini dijitalleştirerek daha verimli hale getirmeyi hedeflemektedir.

## Kapsam Dışı:

Projem, şu aşamada sınav süresini yönetimi, soru havuzundan soru seçme ve server-side-rendering gibi özellikleri içermemektedir ancak gelecekte bu özellikleri eklenmesi planlanmaktadır.

## Kullanılan Teknolojiler ve Araçlar

Testify projesini geliştirilmesinde modern teknolojiler ve araçlar kullanılmıştır. Bu bölümde, projenin geliştirilmesinde kullanılan teknolojilerden detaylandırılacaktır.

### Kullanılan Teknolojiler:

**HTML, CSS, JavaScript:** Web sitesinin temel yapısını ve stilini oluşturmak için kullanılmıştır. JavaScript, dinamik içeriklerin yönetilmesi ve kullanıcı etkileşimlerinin sağlanması için kullanılmıştır.

**PHP:** Sunucu tarafında programlama dili olarak kullanılmıştır. PHP, veri tabanı işlemlerinin gerçekleştirilmesinde etkin bir çözüm sunmaktadır.

**Laravel:** PHP tabanlı bir web framework'ü olan Laravel, API isteklerinin yönetilmesi ve veri tabanı işlemlerinin gerçekleştirilmesi için kullanılmıştır. Laravel MVC yapısı ile zengin özellik seti sayesinde projeyi daha modüler ve yönetilebilir hale getirilmiştir. Laravel Query yapısı, veri tabanı sorgularının daha etkin ve okunabilir bir şekilde yazılmasını sağlamaktadır.

**React:** Kullanıcı arayüzünde kullanılmıştır. React bileşen tabanlı yapısı sayesinde tekrar kullanılabilir ve yönetilebilir bileşenler oluşturmamıza olanak sağlamaktadır. Proje JSX yapısı ile yazılmıştır, bu da React bileşenlerinin oluşturulması ve yönetilmesini kolaylaştırmaktadır.

**MatX Teması:** Projenin kullanıcı arayüzü tasarımında kullanılmıştır. MatX kullanmamızdaki neden, modern ve yenilikçi bir UI/UX deneyimi sunarak kullanıcı etkileşimlerini olumlu bir şekilde arttırmaktır.

**JWT (Json Web Token):** Ön ile arka yüz arasında haberleşmede güvenliği sağlamak amacıyla kullanılmıştır. JWT, kullanıcı kimlik doğrulaması ve yetkilendirme işlemlerinde güvenli bir yöntem sunmaktadır.



## Geliştirme Araçları:

**Visual Studio:** Projenin kodlanmasında için kullanılan birincil entegre geliştirme ortamıdır.

**Git:** Projenin sürüm kontrolünü yönetmek için kullanılmıştır. Git, kodun farklı sürümlerinin kaydedilmesi ve gerektiğinde geri dönülmesini sağlayan sürüm kontrol sistemidir.

**Github:** Projenin online olarak barındırılması ve sürüm kontrol işlerinin çevrimiçi olarak yönetilmesi için kullanılmıştır.

**Notion:** Proje yönetimi için kullanılmıştır. Notion, görevlerin takibi, not alma ve iş birliği için çok yönlü kullanım sağlayan bir platformdur.

**Postman:** API testleri ve entegrasyon süreçlerinde kullanılmıştır. Postman, API'lerin doğru çalışıp çalışmadığını kontrol etmek için test aracıdır.

## Veri Tabanı Sistemleri:

**MySQL:** Projenin veri tabanı yönetim sistemi olarak kullanılmıştır. MySQL, hızlı ve güvenli veri tabanı işlemleri için tercih edilmiştir.

**Oracle:** Web uygulamasının barındırılması için Oracle kullanılmıştır. Oracle, yüksek güvenlik ve büyük veri yönetimi gereksinimleri için ideal bir çözüm sunmaktadır.

Bu teknolojiler ve araçlar, projenin başarılı bir şekilde geliştirilmesi ve yönetilmesi için kritik olarak öneme sahiptir. Testify, modern web teknolojilerinin etkin bir şekilde kullanılmasıyla, kullanıcı dostu ve verimli bir eğitim platformu sunmayı amaçlamaktadır.

## Tasarım ve Geliştirilme Süreci

Testify projesinin tasarım ve geliştirme süreci, kullanıcı ihtiyaçlarına uygun, yenilikçi ve kullanıcı dostu bir platform oluşturmak amacıyla dikkatle planlanmıştır.

Bu bölümde, projenin başlangıcından tamamlanmasına kadar geçen süreç detaylandırılmaktadır.

### Tasarım Süreci:

Proje, taslak çizimleri ile başladı. İlk aşamada, projenin genel yapısı ve kullanıcı arayüzü için çeşitli taslaklar oluşturuldu.

Tasarım araçları olarak Figma kullanıldı. Figma, tasarımların dijital ortama aktarılması, prototiplerin oluşturulması ve tasarım sürecinin yönetilmesi için etkili bir platform sundu. Taslak çizimleri, Figma üzerinde detaylandırılarak, projenin nihai kullanıcı arayüzü tasarımları oluşturuldu.

## Geliştirilme Süreci:

Geliştirme süreci, tasarım aşamasının ardından başladı. Proje, bireysel olarak tasarlandı ve geliştirildi. Geliştirme sürecinde çeşitli modern web teknolojileri ve araçlar kullanıldı:

- **HTML, CSS, JavaScript:** Web sitesinin temel yapısını ve stilini oluşturmak için kullanıldı.
- **PHP ve Laravel:** Sunucu tarafı programlama ve API isteklerinin yönetimi için kullanıldı. Laravel, veri tabanı işlemlerinin gerçekleştirilmesinde ve API isteklerinin yönetilmesinde kullanıldı.
- **React ve JSX:** Kullanıcı arayüzünün oluşturulması için tercih edildi. React, bileşen tabanlı yapısı sayesinde tekrar kullanılabilir ve yönetilebilir arayüz bileşenleri sunarken, JSX yapısı ile bileşenlerin oluşturulması ve yönetilmesi kolaylaştırıldı.

## Geliştirme Sürecinde Karşılaşılan Zorluklar:

Geliştirme sürecinde en zorlayıcı aşamalardan biri, fotoğraflı soru sisteminin entegrasyonu oldu. Bu sorunu çözmek için önce soruların kaydedilmesi, ardından fotoğrafların eklenmesi işlemi uygulanarak sistemin sorunsuz çalışması sağlandı.

## Test Süreçleri:

Testify projesinin test süreçleri bireysel olarak gerçekleştirildi. Proje, geliştirme aşamalarında düzenli olarak test edilerek hatalar giderildi ve sistemin stabil çalışması sağlandı.

## Sürüm Kontrol Sistemi:

Projenin sürüm kontrolü için Git kullanıldı. Git, kodun farklı sürümlerinin etkin bir şekilde takip edilmesini ve gerektiğinde geri dönülmesini sağladı. Ayrıca, kodun güvenli bir şekilde yedeklenmesi ve yönetilmesi için de etkili bir araç oldu.

Tasarım ve geliştirme süreci boyunca, kullanıcı dostu ve işlevsel bir sistem oluşturmak için dikkatli ve planlı bir yaklaşım benimsedim. Bu süreçte kullanılan teknolojiler ve yöntemler, projenin başarılı bir şekilde tamamlanmasını sağladı.

## Fonksiyon ve Özellikler

Testify platformu, kullanıcılarına çeşitli fonksiyonlar ve özellikler sunarak hem öğretmenlerin hem de öğrencilerin ihtiyaçlarını karşılamayı hedeflemektedir. Bu bölümde, platformun ana fonksiyonları ve özellikleri detaylandırılmaktadır.

## Kullanıcı Giriş ve Kayıt Olma İşlemleri:

**Kayıt Olma:** Kullanıcılar, Testify platformunun ana sayfasında "Kayıt Ol" butonunu kullanarak kayıt işlemini gerçekleştirebilirler. Kayıt işlemi sırasında kullanıcıdan e-posta, şifre, isim, soy isim ve rol (öğretmen veya öğrenci) bilgileri istenir.

**Giriş Yapma:** Kullanıcılar, e-posta ve şifre bilgileri ile "Giriş Yap" butonunu kullanarak sisteme giriş yapabilirler.

**Şifre Sıfırlama:** Kullanıcılar, "Giriş Yap" butonuna tıkladıktan sonra "Şifremi Unuttum" butonuna tıklayarak e-posta adreslerini girerler. E-posta adresine gönderilen link ile şifre sıfırlama işlemi tamamlanır.

## Öğretmen Özellikleri:

**Soru Ekleme:** Öğretmenler, sisteme giriş yaptıktan sonra sol tarafta bulunan navbar'dan "Öğretmen → Oluşturduğum Testler" kısmına girerek test oluşturabilirler. Test oluşturulduktan sonra testin "Bak" butonuna basarak soru ekleme ekranına geçilir. Sol altta bulunan artı (+) ikonunun üzerine gelerek istenilen soru çeşidi eklenebilir. Sorulara şık eklemek için sorunun yanındaki artı (+) iconu kullanılır.

**Sınav Sonuçlarını Görüntüleme:** Öğretmen olarak sisteme giriş yaptıktan sonra "Öğretmen → Oluşturduğum Testler" ekranına geldikten sonra testin üstünde bulunan "Seçenekler" sekmesine tıklanarak açılan sekmede "Çözenleri Gör" butonuna tıklanır. Ekrana gelen sayfada sınav sonuçları görüntülenebilir.

**Sınav Kağıtlarını Görüntüleme:** Öğretmen olarak sisteme giriş yaptıktan sonra "Öğretmen → Oluşturduğum Testler" ekranında "Seçenekler" sekmesinde "Çözenleri Gör" butonuna tıklanır. Ekrana gelen sayfada öğrenci sınav kağıtlarını görmek için listeden "Bak" butonuna basılması yeterlidir.

## Öğrenci Özellikleri:

**Sınava Katılma:** Öğrenci olarak sistem giriş yaptıktan sonra panelde sol tarafta bulunan navbar'dan "Öğrenci → Bütün Testler" sekmesine giderek çözmek istedikleri testi seçip "Çöz" butonuna basarak sınavlara katılabilirler.

**Sınav Sonuçlarını Öğrenme:** Öğrenci olarak sistem giriş yaptıktan sonra panelde sol tarafta bulunan navbar'dan "Öğrenci → Çözdüğüm Testler" sekmesine giderek sınav sonuçlarını görebilirler.

**Sınav Kağıtlarını Görüntüleme:** Öğrenci olarak sistem giriş yaptıktan sonra panelde sol tarafta bulunan navbar'dan "Öğrenci → Çözdüğüm Testler" sekmesine gidip "Gözden Geçir" butonuna basarak sınav kağıtlarını görüntüleyebilirler.

## Soru Tipleri ve Yönetimi:

**Desteklenen Soru Tipleri:** Açık uçlu, çoktan seçmeli, birden fazla cevap, boşluk doldurma soruları desteklenmektedir.

## Güvenlik Özellikleri:

**Kimlik Doğrulama ve Yetkilendirme:** Kullanıcı kimlik doğrulama ve yetkilendirme işlemleri JWT (JSON Web Token) kullanılarak yapılmaktadır.

## Ekstra Özellikler:

**Resimli Soru Ekleme:** Testin kaydedilmesinin ardından, soruların yanında bulunan resim yükleme ikonu aktif hale gelir ve resimli soru ekleme işlemi gerçekleştirilebilir.

**Sınav Kağıtların ve Sınava Giren Kişilerin Listesini Yazdırma:** Öğretmenler, "Öğretmen → Oluşturduğum Testler" ekranında sınavın "Seçenekler" sekmesinde "Çözenleri Gör" butonuna tıkladıktan sonra ekrana gelen sayfada, başlıkların solunda bulunan yazdırma ikonuna basarak sınava giren kişilerin listesi ve sınav kağıtlarını yazdırabilirler.

**Sınav Paylaşımı:** WhatsApp, LinkedIn, E-posta ve Twitter gibi platformlarda sınavları paylaşmak mümkündür. Kullanıcılar, sistem üzerinden testlerini bu platformlar aracılığıyla kolayca paylaşabilirler.

Bu fonksiyonlar ve özellikler, Testify platformunun kullanıcı dostu ve işlevsel bir eğitim aracı olmasını sağlamaktadır. Öğretmenler ve öğrenciler, platformun sunduğu çeşitli imkanlarla eğitim süreçlerini daha verimli hale getirebilirler.

## Karşılaşılan Sorunlar ve Çözümler

Projenin geliştirilmesi sırasında genel olarak ciddi bir teknik, tasarım veya entegrasyon sorunlarıyla karşılaşmadı. Ancak, iki sorun yaşandı ve bu sorun çözülerek projenin sorunsuz bir şekilde ilerlemesi sağlandı.

### Fotoğrafların Sunucuya Ulaşmama Sorunu:

**Sorun:** Projenin başlangıç aşamasında, fotoğrafların sunucuya ulaşmadığı bir problemle karşılaşıldı. Bu sorunun nedeni, form işlemlerinde 'Content-Type' başlığında form verisi olarak belirtmemekten kaynaklanıyordu. Form verileri doğru şekilde tanımlanmadığı için sunucu, fotoğraf dosyalarını doğru bir şekilde alamıyordu.

**Çözüm:** Bu sorunu çözmek için, form verilerinin 'Content-Type' başlığı doğru bir şekilde 'multipart/form-data' olarak belirtildi. Bu değişiklik yapıldıktan sonra, fotoğraf dosyaları sunucuya başarılı bir şekilde ulaştı.

## Sunucu Taraflı Fotoğraf Gönderme Sorunu:

**Sorun:** Proje geliştirme sürecinde, sunucu tarafında fotoğrafların gelmediği bir problemle karşılaşıldı. Bu sorunun nedeni, SSL için kullanılan CloudFlare'in cache sistemi idi. CloudFlare, ön bellekleme yaparak belirli içeriklerin sunucuya ulaşmasını engelliyordu.

**Çözüm:** Bu sorunu çözmek için CloudFlare'in "Developer Mode" (Geliştirici Modu) açıldı. Developer Mode, CloudFlare'in ön bellekleme işlevlerini geçici olarak devre dışı bırakarak, içeriklerin doğrudan sunucudan alınmasını sağladı. Bu sayede, fotoğraf gönderim sorunu çözüldü ve sistem sorunsuz çalışmaya devam etti.

Genel olarak, projenin geliştirilmesi sürecinde karşılaşılan bu iki teknik sorun, hızlı ve etkili bir şekilde çözülerek projenin başarılı bir şekilde tamamlanması sağlandı.

## Sonuç ve Gelecek Çalışmalar

**Sonuç:** Testify projesi, "Proje-1" dersi kapsamında tek başıma geliştirdiğim bir web sitesi projesi olarak başarılı bir şekilde tamamladım. Projenin genel başarısı değerlendirildiğinde, bazı eksiklikler olmasına rağmen, mevcut haliyle bile oldukça iyi bir performans sergilediğini ve kullanışlı bir iş çıkardığını düşünüyorum. Proje, belirlenen hedeflere ulaşmanın ötesinde, ek özellikler de içerecek şekilde planlanan çalışmanın bir adım ötesine geçmektedir. Kullanıcılar tarafından iyi karşılanan ve sevilen bu proje, eğitim alanında hem öğretmenlerin hem de öğrencilerin işlerini kolaylaştırmayı başardı.

**Gelecek Çalışmalar:** Proje, bitmiş bir proje olmasına rağmen, teknolojinin sürekli gelişmesi ve yeni yapay zeka araçlarının ortaya çıkmasıyla güncellemeye ve geliştirmeye açık bir sistemdir. Gelecekte, aşağıdaki alanlarda projeyi daha da geliştirmeyi planlıyorum:

### Yeni Teknolojik Araçlar:

- Çıkan yeni yapay zeka araçları ve teknolojilerle entegrasyon sağlanarak, sistemin performansı ve kullanıcı deneyimi artırılabilir.
- Özellikle sınavların değerlendirilmesi ve analizinde daha gelişmiş yapay zeka çözümleri kullanılabilir.
- Yapay zeka ile soru oluşturma özelliği eklenebilir. Bu, öğretmenlerin daha hızlı ve daha kolay bir şekilde sınav soruları oluşturmalarını sağlar.

### Soru Havuzu ve Soru Seçme:

- Soru havuzundan soru seçme özelliği eklenebilir. Bu, öğretmenlerin mevcut soru havuzundan soruları hızlıca seçip sınav oluşturma süreçlerini hızlandırır.

### Sınav Süresi Yönetimi:

- Gelecekte, sınav süresi yönetimi eklenerek, öğrencilerin sınavlarını belirli bir zaman diliminde tamamlamaları sağlanabilir. Bu, sınavların adil ve düzenli olmasını destekler.

### Gelişmiş Yazılım Entegrasyonu:

- Daha iyi yazılım entegrasyonları ile sistemin verimliliği artırılabilir. Örneğin, farklı eğitim platformları ve öğrenme yönetim sistemleri ile entegrasyon sağlanarak, kullanıcıların daha geniş bir ekosistemde çalışması mümkün olabilir.

### Server Side Rendering (SSR):

- Performansı ve SEO'yu artırmak için server side rendering (SSR) teknikleri uygulanabilir.
- Bu, kullanıcıların daha hızlı ve etkili bir şekilde siteyi kullanmalarını sağlar.

### Kullanıcı Dostu Özellikleri:

- Kullanıcı deneyimini geliştirmek için arayüzde yapılacak iyileştirmeler ve kullanıcı geri bildirimlerine göre eklemeler yapılabilir.

Sonuç olarak, Testify projesi, başlangıç aşamasında belirlenen hedeflerin ötesine geçerek başarılı bir performans sergilemiş ve kullanıcılar tarafından beğenilmiştir. Gelecekte yapılacak geliştirmelerle, projenin daha da ileriye taşınması ve kullanıcı ihtiyaçlarını daha iyi karşılaması hedeflenmektedir.

## Kaynakça

- [1] <https://react.dev/>
- [2] <https://demos.ui-lib.com/matx-react-doc/>
- [3] <https://jwt-auth.readthedocs.io/en/develop/laravel-installation/>
- [4] <https://laravel.com/>
- [5] <https://axios-http.com/docs/intro>
- [6] <https://learning.postman.com/docs/introduction/overview/>
- [7] <https://reactrouter.com/en/main>