

# **SADOS ÖĞRENCİ YÖNETİM SİSTEMİ VERSİYON 2 GELİŞTİRİLMESİ HAKKINDA TEKNİK ŞARTNAME**

## **1. Proje Tanımı**

Sakarya Uygulamalı bilimler üniversitesi Dil Eğitimi uygulama ve araştırma merkezini kayıt olan veya olacak olan yabancı öğrenciler için ; Başvuru Takibi , Ödeme Takibi Öğrenci Bilgileri alınması , Öğrenci Belgesi Verilmesi , Eğitim Dönemlerine Atanması , Kur Seçimi Yapılması , Derslik Ataması yapılması , Şube Seçimi Yapılması , Öğretmen Atamaları yapılması , Sınıfların oluşturulması , Yoklama Takibi , Devamsızlık Takibi , Sınav sonuçlarının ilan edilmesi , Detaylı İstatistiklerin saklanması , Öğrenci Kesin Kayıt formu dijital ortamda oluşturulması, Arşiv için sicil etiketinin online ortamda oluşturulması, Amaçlanmaktadır.

## **Projenin Kapsamı**

### **a. Kayıt İşlemlerinin Yönetimi**

Yazılım, öğrencilerin çevrimiçi kayıt olabilmelerine imkan tanır. Kullanıcı dostu arayüzü ile kayıt formu doldurma sürecini basitleştirir ve gerekli bilgileri (isim, soyisim, doğum tarihi, milliyet, iletişim bilgileri vb.) toplar. Ayrıca, kullanıcı doğrulama ve e-posta onayı gibi güvenlik önlemleri ile süreci destekler.

### **b. Ödeme İşlemlerinin İzlenmesi ve Yönetilmesi**

Öğrenciler, ders ücretlerini çevrimiçi ödeme sistemi üzerinden kolayca gerçekleştirebilir. Yazılım, ödeme geçmişinin ve faturaların görüntülenmesine olanak tanır. Ayrıca, ödeme hatırlatıcıları ve bildirimleri ile öğrencilerin ödeme süreçlerini düzenli tutar.

### **c. Devamsızlık Takibi**

Öğrenci yoklamalarının dijital olarak alınmasını sağlar. Öğrencilerin derslere katılım durumları kolayca izlenebilir ve yoklama raporları oluşturulabilir. Bu, öğretmenlerin ve yöneticilerin öğrenci devamsızlığını etkin bir şekilde takip etmelerine yardımcı olur.

### **d. Ders Programlarının Planlanması ve Düzenlenmesi**

Yazılım, öğrenci ve öğretmenler için ders programlarının oluşturulmasını ve yönetilmesini sağlar. Ders saatleri, sınıf bilgileri ve öğretmen atamaları gibi detaylar kolayca düzenlenebilir. Ders programları hem öğretmenler hem de öğrenciler tarafından görüntülenebilir ve güncellenebilir.

### **e. Öğrenci Bilgileri ve Profillerinin Yönetilmesi**

Öğrenciler, kişisel profillerini oluşturabilir ve güncelleyebilir. Profildeki bilgiler (iletişim bilgileri, eğitim durumu, vb.) sürekli güncellenebilir ve yönetilebilir. Ayrıca, profil fotoğrafı yükleme ve değiştirme özelliği de bulunmaktadır.

## f. Öğrencilerin Menşelerine Göre Sınıflandırılması

Yazılım, öğrencilerin milliyetlerine göre kategorilendirilmesini sağlar. Bu sınıflandırma, kategori bazlı raporların oluşturulmasına ve milliyete göre öğrenci gruplarının yönetilmesine imkan tanır.

## 2. Teknolojiler ve Araçlar

- **Backend Framework:** PHP Laravel
- **Veritabanı:** MySQL
- **Ön Yüz:** HTML, CSS, Tailwind CSS, Livewire, Alpine.js
- **Diğer:** Composer, Artisan CLI, Laravel Eloquent ORM

## 3. Özellikler ve Fonksiyonlar

### 3.1 Kayıt Sistemi

- Öğrencilerin çevrimiçi kayıt olabilmeleri için kullanıcı dostu bir arayüz.
- Kayıt formunda gerekli bilgilerin (isim, soyisim, doğum tarihi, milliyet, iletişim bilgileri vb.) toplanması.
- Kullanıcı doğrulama ve e-posta onayı.

### 3.2 Ödeme Takibi

- Öğrencilerin ders ücretlerini çevrimiçi ödeme sistemi üzerinden yapabilmesi.
- Ödeme geçmişi ve faturaların görüntülenmesi.
- Ödeme hatırlatıcıları ve bildirimleri.
- Kullanılan banka için paket geliştirme işlemleri
  - Kütüphane Oluşturma: Ödeme işlemlerini destekleyen, güvenli ve ölçeklenebilir bir yazılım kütüphanesi geliştirilmelidir. Bu kütüphane, ödeme işlemleri için gerekli tüm fonksiyonları ve API'ları içermelidir.
  - Ödeme Entegrasyonu: Farklı ödeme yöntemlerini (kredi kartı, banka havalesi, dijital cüzdanlar vb.) destekleyen entegrasyonlar yapılmalıdır. Entegrasyon, kullanıcı dostu bir arayüz ve sorunsuz bir işlem deneyimi sunmalıdır.
  - Güvenlik: Tüm ödeme işlemlerinin güvenliğini sağlamak için gerekli şifreleme ve güvenlik protokolleri uygulanmalıdır.
  - Test ve Doğrulama: Kütüphane ve entegrasyonun tüm bileşenleri titizlikle test edilmeli ve doğrulanmalıdır.

### 3.3 Yoklama Sistemi

- Öğrenci yoklamalarının dijital olarak alınabilmesi.
- Öğrencilerin derslere katılım durumlarının izlenmesi.
- Yoklama raporlarının oluşturulması ve yönetimi.

### 3.4 Ders Programı

- Öğrenciler ve öğretmenler için ders programlarının oluşturulması ve yönetimi.

- Ders saatleri, sınıf bilgileri ve öğretmen atamaları.
- Ders programlarının görüntülenmesi ve güncellenmesi.

### 3.5 Profil Yönetimi

- Öğrencilerin kişisel profillerini oluşturabilmesi ve güncelleyebilmesi.
- Profildeki bilgilerin (iletişim bilgileri, eğitim durumu, vb.) güncellenebilmesi.
- Profil fotoğrafı yükleme ve değiştirme.

### 3.6 Kategorilendirme

- Öğrencilerin milliyetlerine göre kategorilendirilmesi.
- Kategori bazlı raporların oluşturulması.
- Milliyete göre öğrenci gruplarının yönetimi.

## 4. Güvenlik

- Veri şifreleme ve güvenli veri saklama.
- Kullanıcı oturum yönetimi ve yetkilendirme.
- Güvenlik duvarları ve giriş denemesi sınırlamaları.
- Tüm ödeme işlemlerinin güvenliğini sağlamak için gerekli şifreleme ve güvenlik protokolleri uygulanması.

## 5. Performans

- Yüksek trafikte performansın korunması için önbellekleme mekanizmaları.
- Veritabanı optimizasyonu ve sorgu performansı iyileştirmeleri.
- Ölçülenebilir mimari.

## 6. Geliştirme ve Test

- **Geliştirme:** Laravel framework'ü kullanarak modüler ve bakımı kolay bir yapı oluşturulması.
- **Test:** Birim testleri, entegrasyon testleri ve kullanıcı kabul testleri.
- **Versiyon Kontrolü:** Git kullanılarak kaynak kod yönetimi.
- **Ödeme Kontrolü:** Ödemesi gerçekleştirildikten sonra sistem ve banka arasında bilgilerin doğruluk testi.

## 7. Kullanıcı Desteği

- Kullanıcı eğitimleri ve rehber dokümanlar.
- Teknik destek ve yardım masası hizmetleri.
- Sıkça sorulan sorular (SSS) bölümü.

## 8. Dokümantasyon

- Kullanıcı kılavuzu ve teknik dokümantasyon.
- API dokümantasyonu (eğer harici sistemlerle entegrasyon mevcutsa).
- Sistem mimarisi ve veri modeli dokümantasyonu.

## 9. Dağıtım ve Bakım

- Sunucu yapılandırması ve dağıtım süreçleri.
- Periyodik bakım ve güncellemeler.
- Yedekleme ve geri yükleme prosedürleri.

## 10. Öğrenci Arayüzü

- Öğrenci Platformu Arayüz Şeması
  - Devamsızlık Durumu: Öğrencinin derslere katılım durumu ve devamsızlık detayları.
  - Başvurular: Öğrencinin yaptığı tüm başvuruların durumu ve detayları.
  - Sınav Sonuçları: Öğrencinin sınavlardan aldığı notlar ve sınav detayları.
  - Bakiye ve Ödemeler: Öğrencinin hesap bakiyesi ve yapılan ödemelerin detayları.

## 11. Öğretmen Arayüzü

- Öğretmenlerin sınıflarını görebilmesi
- Yeni devamsızlık listelerini hazırlayabilmeleri
- Yapılan devamsızlık listelerinin öğrenci işlerine aktarımı

## 12-Teknik Bilgiler

Sistem, PHP (8.0) programlama dilinin Laravel çerçevesi ile yazılacaktır. Laravel'in tercih edilmesinin en büyük sebebi, yazılımın güvenliği ve sürekli güncellenmesinin hedeflenmesidir. Laravel'in ortak çalışma prensibi sayesinde ekip ne kadar büyürse büyüsün yazılımın ilerlemesi sürekli bir hız kazanacaktır.

İşlemlerin hızlandırılması ve kullanıcı etkileşimlerinin optimize edilmesi için, Livewire kütüphanesi kullanılacaktır. Livewire, Laravel ile mükemmel bir uyum içinde çalışarak, JavaScript kullanmadan dinamik kullanıcı arayüzleri oluşturmayı sağlar. Ayrıca, Livewire ile eş zamanlı çalışan Alpine.js, basit ve kompakt bir JavaScript frameworkü olarak, etkileşimlerin yönetilmesine katkı sağlayacaktır. Bu kombinasyon, sistemin hızlı çalışmasını ve modüler olmasını büyük ölçüde desteklemektedir.

Sistem, internet bağlantısı olan bilgisayar, tablet ve telefon gibi tüm cihazlardan erişilebilecektir. Veri tabanı olarak MySQL kullanılacaktır. Sistem içinde REST API kullanılmaktadır. Bu API, sistem içinde sürekli haberleşmeyi sağlayarak, sistemin sadece PHP'ye bağlı kalmamasını ve gereken tüm platformlardan destek almasını sağlamaktadır. Örneğin, bir destek oluşturulduğunda, diğer sisteme REST API ile iletişime geçilerek gerekli veriler iletilmekte ve diğer sistem yine aynı şekilde REST API ile bu verileri almaktadır.

Sistem içerisinde arayüz, Tailwind CSS kütüphanesi kullanılarak tasarlanacaktır. Tailwind, bileşen tabanlı yapısı ile arayüzlerin hızlı ve tutarlı bir şekilde geliştirilmesini sağlar. Sistem içerisinde işlemler sonucunda bilgilendirme için SweetAlert kütüphanesi kullanılacak, arayüz görüntülerinin güzelleştirilmesi ve kişiselleştirilmesi için ise Tailwind CSS kütüphanesi kullanılacaktır.