***Akademik Personel Başvuru Sistemi***

***Tasarımı ve Uygulaması***

Bilişim Sistemleri Mühendisliği

Kocaeli Üniversitesi

Barkın Emre Sayar, Erol Malkoç, Hızır Ceylan

[barkinemresayar@gmail.com](mailto:barkinemresayar@gmail.com), [erolmalkoc04@gmail.com](mailto:erolmalkoc04@gmail.com), [hizir5303@gmail.com](mailto:hizir5303@gmail.com)

Özet

*Bu projede, akademik personel başvurularının dijital ortamda etkin, şeffaf ve kullanıcı dostu biçimde yönetilmesi hedeflenmiştir. Geliştirilen sistem, akademik adayların başvuru süreçlerini kolaylaştırırken, yöneticilerin ve jüri üyelerinin değerlendirme süreçlerini hızlı ve hatasız yürütmesine imkân sağlamaktadır. Sistem; React.js, Node.js, Express, PostgreSQL gibi modern web teknolojileri kullanılarak geliştirilmiş olup, kimlik doğrulama işlemleri JWT, dosya depolama AWS S3 ve bildirim işlemleri Firebase kullanılarak gerçekleştirilmiştir.*

Abstract

*In this project, it is aimed to manage academic staff applications in an effective, transparent and user-friendly digital environment. The developed system facilitates the application processes of academic candidates, while enabling administrators and jury members to carry out the evaluation processes quickly and accurately. The system was developed using modern web technologies such as React.js, Node.js, Express, PostgreSQL, authentication processes were carried out using JWT, file storage AWS S3 and notification processes Firebase.*

**Keywords**: panel, veritabanı, bulut depolama sistemi, bildirim işlemleri, web teknolojileri

* **GİRİŞ (INTRODUCTION)**

Günümüzde akademik kadrolara başvuru süreçleri çoğunlukla fiziksel evraklarla yürütülmekte, bu durum adaylar, yönetici ve jüri üyeleri açısından zorluklar oluşturmaktadır. Bu süreçlerin manuel yönetimi, zaman kaybı, belge karmaşası ve değerlendirme süreçlerinde hatalara yol açmaktadır.

Bu problemleri çözmek ve süreci dijitalleştirmek amacıyla, Kocaeli Üniversitesi bünyesinde kullanılmak üzere “Akademik Personel Başvuru Sistemi” geliştirilmiştir. Projenin temel amacı; başvuru süreçlerini kolaylaştırmak, belge karmaşasını ortadan kaldırmak ve jüri değerlendirme süreçlerini daha şeffaf hâle getirmektir.

* **YÖNTEM (METHODOLOGY)**

1. **Kullanılan Teknolojiler:**

**Frontend:** React.js

**Backend:** Node.js (Express.js framework)

**Veritabanı:** PostgreSQL

**Kimlik Doğrulama:** JWT

**Dosya Depolama:** AWS S3

**Bildirim Sistemi:** Firebase Cloud Messaging

**API Entegrasyonları:** e-Devlet ve Nüfus Müdürlüğü

1. **Sistem Mimarisi:**

**Frontend:** Kullanıcı arayüzü React.js ile geliştirilmiş olup, kullanıcı dostu bir arayüz sağlamaktadır.

**Backend:** Node.js ve Express.js kullanılarak API mimarisi ile geliştirilmiş olup, frontend ile güvenli veri alışverişini sağlamaktadır.

**Veritabanı:** PostgreSQL ile kullanıcı bilgileri, ilanlar, belgeler, değerlendirmeler ve başvuru verileri depolanmaktadır.

**Depolama ve Bildirim:** Dosyalar AWS S3 üzerinde güvenli şekilde saklanmakta ve kullanıcı bildirimleri Firebase aracılığıyla gönderilmektedir.

1. **Kullanıcı Rolleri:**

**Adaylar:** Başvuru yapabilir, belgeleri yükleyebilir ve başvuru durumlarını takip edebilir.

**Adminler:** İlanları oluşturabilir ve ilan koşullarını düzenleyebilir.

**Yöneticiler:** Kadro başvuru kriterlerini belirleyebilir, jüri üyelerini atayabilir ve değerlendirmeleri takip edebilir.

**Jüri Üyeleri:** Başvuru belgelerini inceleyip değerlendirme raporlarını oluşturabilir.

* **UYGULAMA DETAYLARI**

**(IMPLEMENTATION)**

* 1. **Frontend Geliştirme**

Frontend geliştirme işlemlerinde React.js kütüphanesi kullanılmıştır. React Router v6 ile sayfa yönlendirmeleri yapılmıştır. Axios kütüphanesi, backend API servisleri ile etkileşim için kullanılmıştır. Kullanıcı arayüzü, üniversitenin kurumsal renklerine uygun (yeşil-beyaz) şekilde tasarlanmıştır.

**Aday Arayüzü:** Kullanıcı giriş ekranı, ilanları listeleme, başvuru formu ve dosya yükleme ekranları bulunmaktadır.

**Admin Arayüzü:** İlan ekleme, ilan düzenleme, ilan tarihlerini belirleme ve başvuruları görüntüleme ekranları bulunmaktadır.

**Yönetici Arayüzü:** Kadro kriterleri belirleme, jüri atama ve değerlendirme raporlarını görüntüleme fonksiyonları yer almaktadır.

**Jüri Arayüzü:** Başvuru belgelerini görüntüleyerek değerlendirme raporu yükleyebilecekleri sade bir tasarım bulunmaktadır.

* 1. **Backend Geliştirme**

Backend servisleri Node.js ve Express.js kullanılarak geliştirilmiştir. RESTful API yapısı benimsenmiştir. JWT ile kimlik doğrulama, rol bazlı yetkilendirme sağlanmış, veri güvenliği açısından middleware (ara katman) kullanılmıştır.

-Kullanıcı kimlik doğrulama ve yetkilendirme işlemleri

-Başvuruların oluşturulması ve yönetimi

-İlan bilgilerinin CRUD işlemleri (oluşturma, görüntüleme, güncelleme ve silme)

-Dosya yükleme ve AWS S3 entegrasyonu

-Bildirim gönderme (Firebase Cloud Messaging)

* 1. **Veritabanı Tasarımı**

PostgreSQL üzerinde tablolar arası ilişkiler kurularak veritabanı oluşturulmuştur.

Kullanıcılar, roller, ilanlar, başvurular, değerlendirmeler ve dosyalar için ayrı tablolar oluşturulmuştur.

Veri bütünlüğü, tablolar arası foreign key ve ilişkiler ile sağlanmıştır.

**Veri Tabanı Tabloları**

## roller: Sistemdeki tüm kullanıcı rollerini tutar. (Örneğin: aday, jüri, yönetici,admin)

## kullanicilar: Sistemdeki tüm kullanıcı bilgilerini tutar. (Örneğin: id, tc\_kimlik\_no,ad\_soyad)

## ilanlar: Yönetici veya admin tarafından açılan akademik kadro ilanları. (Örneğin: baslik, kadro\_turu, fakülte)

## tablolar: Akademik atama süreçlerinde kullanılan ana tablolar. (Örneğin: ad, aciklama, aktif\_mi)

## tablo\_satirlari: Tablo 1-4’teki detaylı kriter satırları. (Örneğin: tablo\_id, kategori, faaliyet\_no)

## kadro\_alanlari: Fakülte/alan bazında ayrımlar. (Örneğin: Mühendislik, Sağlık, Fen bilimleri)

## kadro\_alan\_kriterleri: Her alan için belirlenen minimum ve maksimum puan kriterleri. (Örneğin: kadro\_alan\_id, tablo\_id, toplam\_makale\_sayisi)

## basvurular: Adayların açılan ilanlara yaptığı başvurular. (Örneğin: aday\_id, ilan\_id, durum)

## basvuru\_belgeleri: Adayın yüklediği belge bilgileri ve sistem puanı. (Örneğin: belge\_url, aday\_aciklama, sistem\_puani)

## juri\_atamalar: İlanlara atanan jüri üyeleri. (Örneğin: id, ilan\_id, juri\_id)

## juri\_raporlari: Jüri üyeleri tarafından yüklenen değerlendirme raporları. (Örneğin: id, basvuru\_id, juri\_id)

## tablo5\_ozet: Adayın tüm tabloları bazında toplam puanlarının özeti. (Örneğin: tablo1\_puani, tablo2\_puani, tablo3\_puani)

## bildirimler: Sistem içi kullanıcı bildirimlerini tutar. (Örneğin: icerik, okundu, created\_at)

## giris\_kayitlari: Kullanıcıların sisteme giriş hareketlerini kaydeder. (Örneğin: id, giriş\_zamani, ip\_adresi)

* **TARTIŞMA(DISCUSSION)**

Geliştirilen proje ile akademik başvuru süreçlerinde önemli bir dijital dönüşüm gerçekleştirilmiştir. Sistem, adayların başvuru yapma süreçlerini hızlandırırken, yönetici ve jüri üyelerinin belge karmaşasını azaltarak hata oranını düşürmüştür.

Proje sürecinde karşılaşılan en önemli zorluklar:

Dosya yükleme ve yönetimi sırasında karşılaşılan entegrasyon sorunları

Kullanıcı yetkilendirme süreçlerinde JWT entegrasyonu sırasında yaşanan teknik zorluklar

Bu problemler sistematik hata ayıklama (debugging) ve entegrasyon testleri ile çözülmüştür.

* **SONUÇ (RESULT)**

Proje geliştirme süreci sonunda aşağıdaki fonksiyonel özellikler başarıyla tamamlanmıştır:

Kullanıcı dostu ve modern arayüz geliştirildi.

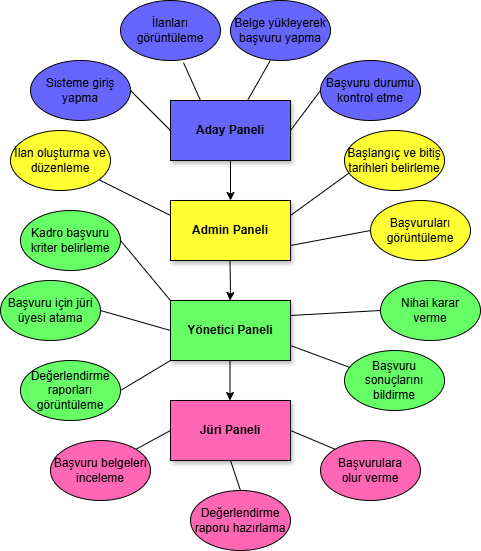
Güvenli kimlik doğrulama ve rol bazlı yetkilendirme başarıyla çalıştırıldı.

Başvuru süreci eksiksiz olarak dijital ortama taşındı.

AWS S3 kullanılarak belgelerin güvenli depolanması sağlandı.

Firebase üzerinden hızlı bildirim sistemi çalıştırıldı.

Sistem; işlevsel testler ve kullanıcı testleri ile doğrulandı, performans ve güvenlik testlerinden başarıyla geçti.



* **KAYNAKÇA(REFERENCES)**

[1] <https://www.w3schools.com/postgresql/>

[2] <https://react.dev/learn>

[3] <https://www.btkakademi.gov.tr/portal/course/react-js-39531>

[4] <https://www.w3schools.com/html/html_css.asp>

[5] <https://expressjs.com/en/guide/routing.html>

[6] <https://docs.aws.amazon.com/s3/>

[7] <https://firebase.google.com/docs/cloud-messaging?hl=tr>

[8] <http://technopat.net/sosyal/konu/e-devlet-api-entegrasyonu.3068959/>

* **DOSYALARIN BULUNDUĞU KONUM**
* https://github.com/hizir-ceylan/Akademik\_Personel\_Basvuru\_Sistemi