



MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
VERİ BİLİMİNE GİRİŞ DERSİ FİNAL PROJESİ

Randevu Sistemi

24120205056 Abdulsamet KARATAŞ

24120205061 Yusuf SAYILGAN

Ders Sorumlusu

Dr. Öğr. Üyesi Muhammet Sinan BAŞARSLAN

Ocak, 2026

İstanbul Medeniyet Üniversitesi, İstanbul

1. GİRİŞ

Bu çalışma, verimli bir randevu yönetim sisteminin veri yapıları ve algoritmalar kullanılarak nasıl optimize edilebileceğini göstermektedir. Projenin temel amacı, sadece veri depolamak değil; zaman çakışmalarını önlemek, salonlar arası en kısa yolu hesaplamak ve sistem performansını $O(\log n)$ seviyesinde tutmaktır. Yazılım dünyasında hataların erken tespiti ve kalite güvencesi maliyeti düşüren en önemli unsurdur. Bu sistemde, hatalı girişlerin önlenmesi için algoritma seviyesinde denetimler (validation) önceliklendirilmiştir.

2.MATERYAL METOD

2.1.Kullanılan IDELER

- VS Code: Projenin tüm kodlama süreci bu editör üzerinde gerçekleştirilmiştir.
- Node.js: Web sunucusu ve C programı ile iletişim (IPC) katmanı olarak kullanılmıştır.

2.2 Kullan Veri Yapıları

- AVL Interval Tree: Randevuların başlangıç saatlerine göre dengeli saklanması ve çakışma (overlap) kontrolünün hızlı yapılmasını sağlar.
- Dijkstra Algoritması: Salonlar arası mesafelerin analizi ve en kısa rota hesaplaması için kullanılmıştır.
- Stack (Yığın): "Geri Al" (Undo) özelliği için son yapılan işlemi tutan LIFO yapısı.
- Queue (Kuyruk): Çakışan veya bekleme listesine alınan randevuların yönetimi (FIFO).
- Quicksort & Binary Search: Verilerin ID tabanlı hızlı sıralanması ve aranması için entegre edilmiştir.

3.UYGULAMA

3.1 Tasarım

Sistem, "Hibrit" bir yapıdadır. Node.js üzerinden gelen kullanıcı istekleri, child_process

kütüphanesi ile C dilinde yazılmış olan randevusistemi.exe dosyasına iletilir. C motoru tüm

algoritmik hesaplamaları (AVL dengeleme, Dijkstra hesabı) yaptıktan sonra sonuçları data.json

ve randevular.csv dosyalarına işler.

3.2 Test Süreci ve Fonksiyonellik

- Hata Denetimi: Başlangıç saati bitişten sonra olan veya aynı ID ile girilen kayıtlar sistem tarafından otomatik reddedilir.
- BFS Görüntüleme: seviyeGosterBFS fonksiyonu ile ağaç yapısı katman katman taranarak sistemin doğruluğu test edilmiştir.
- Kalıcı Veri: Program her açıldığında CSV dosyasından verileri otomatik yükleyerek süreklilik sağlar.

4.SONUÇ VE TARTIŞMA

Uygulama aşamasında, C dilinin düşük seviyeli bellek yönetim gücü ile JavaScript'in hızlı arayüz geliştirme esnekliği birleştirilmiştir. AVL ağaçlarının kullanımı, binlerce kayıt olsa dahi sistemin yavaşlamamasını sağlamıştır.

Gelecek Çalışmalar ve AI Kullanımı: Proje geliştirme sürecinde ChatGPT ve AI araçları, özellikle AVL ağacı döndürme fonksiyonlarının (left/right rotate) hata ayıklamasında ve karmaşık C kütüphanelerinin (time.h gibi) kullanımında etkin rol oynamıştır. AI araçları, modern yazılım geliştirme süreçlerinde birer "yardımcı pilot" olarak dokümantasyon ve kod kalitesini artırmak için kullanılmıştır.

5. EKLER

Github: <https://github.com/cyberperfo/Randevu-Sistemi2.0>

Medium: <https://medium.com/@abdulsametkaratas2841/c-programlama-dili-ile-randevu-sistemi-projesi-b278dd45f806>