

# SAGLIK ALGORITMASI

Veri Odakli Butuncul Yasam ve Besin Yonetimi

## DETAYLI PROJE RAPORU VE KOD ANALIZI

*Gelistirici: Beytullah Daldaban*

[github.com/beytullahdaldaban/Saglik-Algoritmasi](https://github.com/beytullahdaldaban/Saglik-Algoritmasi)

*Ders: Web Tasarimi ve Programlama (Final Projesi)*

# 1. YONETICI OZETI VE PROJE VIZYONU

## 1.1. Problem Tanimi ve Mevcut Durum

Gunumuzde hizli tuketim kulturu ve paketli gidalarin artisti, toplum sagligini tehdit eden 'gizli seker', 'yuksek fruktoz' ve 'bos kalori' sorunlarini beraberinde getirmiştir. Market raflarındaki urunlerin üzerinde yer alan karmasik besin etiketleri, ortalama bir tuketici icin anlasilmasi guc teknik veriler barindirmaktadır. Mevcut mobil uygulamalar ve web siteleri ise genellikle kullaniciyi sadece kalori saymaya yonlendirmekte, alinan kalornin kaynagini (Protein, Yag, Karbonhidrat dengesi) ve kalitesini goz ardı etmektedir. Bu durum, bireylerde 'Bir kereden bir sey olmaz' algisinin yerlesmesine ve uzun vadede obezite, diyabet gibi metabolik rahatsızlıkların artmasına neden olmaktadır.

## 1.2. Cozum: Veri Odakli Yasam Yonetimi

'Saglik Algoritması' projesi, bu bilgi kirliligini onlemek ve kullaniciyi pasif bir tuketiciden, kendi sagligini yoneten bilincli bir 'Veri Analisti'ne donusturmak amaciyla gelistirilmistir. Platform, statik veri girisinin otesine gecerek; OpenFoodFacts API entegrasyonu ile global urun veritabanina anlik erisim saglar. Kullanici, merak ettigi bir urunu arattiginda saniyeler icinde o urunun icerigindeki gizli sekeri, yag oranini ve protein degerini goruntuleyebilir. Proje, kullaniciya yasaklamalar getirmek yerine, 'Yedigim seyin bedelini bil' felsefesiyle yaklasarak surdurulebilir bir beslenme aliskanligi kazandirmayi hedefler.

## 1.3. Bilimsel Altyapi ve Teknolojik Fark

Bu projeyi digerlerinden ayiran en temel ozellik, arka planda calisan guclu matematiksel hesaplama motorudur. Sadece boy ve kilo verisine dayali basit VKI hesaplamalarının otesine gecilerek; US Navy (ABD Donanması) Yag Orani Formulu ve Harris-Benedict Metabolizma Denklemleri sisteme entegre edilmistir. Kullanıcının boyun, bel ve kalca olculerini isleyen sistem, kisiye ozel su tuketim hedefleri ve makro besin planlari olusturur. Teknik olarak Web 2.0 standartlarında, Asenkron (AJAX) mimari ile gelistirilen 'Saglik Algoritması', modern web teknolojilerinin toplumsal fayda icin nasıl kullanilabileceginin somut bir ornegidir.

# 2. YAZILIM MIMARISI VE BACKEND YAPISI

Proje, 'Spagetti Kod' yapisindan kacinilarak, 'Separation of Concerns' (Gorevlerin Ayrılması) prensibine gore tasarlanmistir. Sistem uc ana katmandan olusur: Client (Frontend), API Layer (Islemler) ve Data Layer (Veritabani).

## 2.1. Client-Server Iletisimi

Geleneksel PHP sitelerinde her islemde sayfa yenilenirken, bu projede 'Single Page Application' (SPA) hissi veren bir yapi kurulmustur. Kullanici 'Urun Ara' butonuna bastiginda veya 'Kaydet' dediginde, tarayici arka planda sunucu ile sessizce haberlesir.

# 3. KOD MANTIGI VE DOSYA ANALIZI

Sistemin beyni olan kod yapilari, veri guvenligini ve islem hizini optimize edecek sekilde kurgulanmistir. Asagida temel dosyaların calisma mantigi detaylandırılmıştır:

## 3.1. Frontend Motoru: script.js

Bu dosya, kullanıcı etkileşimlerini yöneten ana JavaScript motorudur. İçinde bulunan 'gidaAra()' fonksiyonu şu adımları izler:

1. Kullanıcının girdiği metni alır ve validasyon (doğrulama) yapar.
2. 'fetch()' metodu ile OpenFoodFacts API'sine asenkron bir istek gönderir.
3. Gelen devasa JSON verisini parse eder ve sadece gerekli olan (Kalori, Seker, Protein) verileri süzer.
4. DOM Manipulasyonu ile sonuçları HTML kartları olarak ekrana basar.

### 3.2. Veri İşleyici: işlemler/kaydet.php

Frontend'den gelen verilerin güvenli bir şekilde veritabanına işlendiği yerdir. İşleyiş şu şekildedir:

1. Veri Alımı: 'php://input' akisi üzerinden ham JSON verisi okunur.
2. Temizlik: 'htmlspecialchars()' fonksiyonu ile XSS saldırılarına karşı veriler temizlenir.
3. Tip Donusumu: Gelen string değerler (Örn: '12.5g') sayısal float değerlere çevrilir.
4. Kayıt: Temizlenen veriler PDO Prepared Statement kullanılarak 'gidalar' tablosuna eklenir.

### 3.3. Algoritma: hesaplamalar/yag\_orani.php

Bu dosya, US Navy (ABD Donanması) formüllerini barındırır. Erkek ve Kadınlar için farklı logaritmik hesaplamalar yapar. Orneğin erkeklerde;  $86.010 * \log_{10}(\text{bel} - \text{boyun}) - 70.041 * \log_{10}(\text{boy}) + 36.76$  formülü kodlanmıştır.

## 4. VERİTABANI SEMASI VE VERİ YONETİMİ

---

Veri tutarlılığı için Relational Database (İlişkisel Veritabanı) yapısı kullanılmıştır. SQLite veritabanı içerisinde 5 ana tablo bulunmaktadır:

### 4.1. Tablo: uyeler

- \* id (PK): Benzersiz üye kimliği.
- \* ad\_soyad: Kullanıcının görünen adı.
- \* email: Giriş için kullanılan benzersiz anahtar.
- \* sifre: 'password\_hash' ile kriptolanmış güvenli veri.
- \* rol: RBAC sistemi için yetki seviyesi (admin/uye).

### 4.2. Tablo: gidalar

- \* id (PK): Kayıt ID'si.
- \* uye\_id (FK): Hangi üyenin eklediğini belirten dış anahtar.
- \* makro\_degerler: Kalori, Protein, Yağ, Seker (Float).
- \* tur: Kaydın 'Analiz' mi yoksa 'Tüketim' mi olduğunu belirtir.

## 5. SİBER GÜVENLİK ONLEMLERİ

---

Kullanıcı verilerini korumak adına, uygulama geliştirme sürecinde 'Security First' (Önce Güvenlik) yaklaşımı benimsenmiştir.

### 5.1. SQL Injection Koruması

Tüm veritabanı sorgularında (SELECT, INSERT, UPDATE) PDO kütüphanesinin 'Prepared Statements' özelliği kullanılmıştır. Bu sayede, kötü niyetli kullanıcıların form alanlarına SQL kodları yazarak veritabanını

manipule etmesi teknik olarak imkansiz hale getirilmistir.

## 5.2. Oturum Guvenligi (Session Hijacking Prevention)

PHP Session mekanizmasi kullanilarak, kullanicinin yetkisi olmayan sayfalara (Orn: Admin Paneli) erismesi engellenmistir. Her kritik islemde (Orn: Silme islemi - sil.php), islem yapan kisinin 'session\_id'si ile verinin sahibi dogrulanmaktadır.

## 6. SONUC VE GELECEK VIZYONU

---

'Saglik Algoritmasi' projesi; sadece bir web sitesi degil, karmasik verileri isleyip kullaniciya anlamlı grafikler ve raporlar sunan bir 'Karar Destek Sistemi'dir. Veritabani mimarisi, API entegrasyonu ve modern frontend tekniklerinin (Bootstrap, JS) basarili bir sentezidir.

Bu proje ile; Web Programlama dersi kapsaminda ogrenilen teorik bilgilerin (CRUD, Session, Database, Security), gercek dunya problemlerini cozen somut bir urune donusturulmesi basariyla saglanmistir.