# Code Smell Değerlendirme Raporu

🔹 Proje Adı: KelimeEzberApp

🔹 İnceleme Tarihi: 2025-06-01

## 🚩 Tespit Edilen Code Smell Türleri ve Açıklamaları

### 1. God Class (Tanrı Sınıf)

Tespit: FormTest.cs dosyası çok fazla sorumluluk üstlenmektedir.  
Neden Sorunlu: Kodun okunabilirliğini ve bakımını zorlaştırır.  
Öneri: Bu sınıfı TestManager, UIHandler gibi daha küçük sınıflara bölün.

### 2. Long Method (Uzun Metot)

Tespit: btnTest\_Click metodu çok uzundur.  
Neden Sorunlu: Uzun metotlar anlaşılmaz ve test edilemez hale gelir.  
Öneri: HazirlaTestSorulari(), DegerlendirCevaplar() gibi parçalara bölün.

### 3. Duplicate Code (Yinelenen Kod)

Tespit: FormKelimeEkle.cs ve FormTest.cs içinde benzer kodlar tekrar ediyor.  
Neden Sorunlu: Kodun bakımı zorlaşır.  
Öneri: Ortak işlemleri yardımcı sınıfa taşıyın (örneğin: KelimeServisi).

### 4. Poor Naming (Kötü İsimlendirme)

Tespit: btn1\_Click gibi anlamsız isimler kullanılmış.  
Neden Sorunlu: Kodun ne yaptığını anlamak güçleşir.  
Öneri: Buton ve metodlara işlevsel isimler verin (btnKaydet\_Click gibi).

### 5. Data Clumps (Veri Yığınları)

Tespit: FormKelimeEkle içinde sürekli aynı veri setleri (örnek: kelime, anlam, cümle) birlikte taşınıyor.  
Neden Sorunlu: Kod esnekliğini azaltır.  
Öneri: Bu verileri içeren bir 'Kelime' sınıfı oluşturun ve sınıf üzerinden taşıyın.

## 🎯 Genel Öneriler

- Sınıfları tek sorumluluğa göre bölün (Single Responsibility Principle)  
- Uzun metotları anlamlı alt metotlara ayırın  
- Yinelenen kodları yardımcı sınıflarda toplayın  
- Anlamlı isimlendirme standartlarına uyun  
- Veri yığınlarını sınıflar içinde gruplayın