# KISS Prensibi Kod Kalitesi Değerlendirme Raporu

🔹 Proje Adı: KelimeEzberApp

🔹 İnceleme Tarihi: 2025-06-01

## 🔎 KISS Prensibi Nedir?

KISS (Keep It Simple, Stupid) prensibi, yazılım geliştirmede basitliği teşvik eden bir yaklaşımdır. Kodun gereksiz karmaşıklıklardan arındırılarak daha anlaşılır, sürdürülebilir ve hataya kapalı şekilde yazılması gerektiğini savunur.

## 🧪 Proje Üzerinde Tespit Edilen Noktalar

### 1. Uzun ve Karmaşık Metotlar

Tespit: `btnTest\_Click` gibi metotlar, birden fazla işlemi aynı anda gerçekleştirmektedir.  
Sorun: Bu durum kodun okunabilirliğini ve test edilebilirliğini düşürür.  
Öneri: Metotlar tek sorumluluk alacak şekilde bölünmelidir. Örneğin, `HazirlaTestSorulari()`, `DegerlendirCevaplar()` gibi alt metotlara ayrılabilir.

### 2. Anlamsız İsimlendirmeler

Tespit: `btn1\_Click`, `btn2\_Click` gibi anlam taşımayan isimlendirmeler kullanılmıştır.  
Sorun: Bu tür isimler kodun anlaşılmasını zorlaştırır.  
Öneri: İsimlendirmeler işlevi yansıtacak şekilde yeniden düzenlenmelidir. Örneğin `btnKaydet\_Click` gibi.

### 3. Yinelenen Kodlar

Tespit: `FormKelimeEkle.cs` ve `FormTest.cs` içerisinde benzer kod parçaları tekrar edilmektedir.  
Sorun: Kodun bakım maliyeti artar ve hata riski yükselir.  
Öneri: Ortak kodlar yardımcı sınıflara taşınarak tekrar kullanılabilir hale getirilmelidir.

### 4. Gereksiz Karmaşık Yapılar

Tespit: Bazı sınıflar birden fazla sorumluluk taşımaktadır.  
Sorun: Bu durum kodun okunabilirliğini ve esnekliğini azaltır.  
Öneri: Her sınıf yalnızca bir sorumluluk almalı ve işlemler farklı sınıflara dağıtılmalıdır (SRP).

## 🎯 Genel Öneriler

- Her sınıf ve metot yalnızca tek bir sorumluluk taşımalıdır.  
- Kodda gereksiz soyutlamalardan kaçınılmalıdır.  
- Yinelenen kod parçaları ortak sınıflarda toplanmalıdır.  
- Anlamlı, işlevsel isimlendirmeler tercih edilmelidir.  
- Gerektiğinde küçük yardımcı metotlarla kod sadeleştirilmelidir.