



İ.T.Ü.  
Bilgisayar Mühendisliği Bölümü  
Doç.Dr. Feza BUZLUCA  
<http://akademi.itu.edu.tr/buzluca>

10.11.2025

## BLG 625 YAZILIM TASARIMI KALİTESİ DÖNEM PROJESİ Yazılımların Tasarım Kalitelerinin Değerlendirilmesi

### Amaç:

- Projenin amacı, bir yazılım projesinin tasarım kalitesi hakkında bilgi sağlayarak karar vericilere yol gösterebilecek bir yöntemin geliştirilmesi ve elde edilen sonuçların yorumlanmasıdır.
- Projenin önemli bir unsuru, elde edilen sonuçların yorumlanması ve gerçek yazılımlar üzerinde sağlamasının yapılarak geliştirilen yöntemde temsil özelliğinin (*representational theory of measurement*) varlığının araştırılmasıdır.
- Yazılım kalitesini değerlendirmek üzere oluşturulan yöntemlerde çıktı olarak sadece belli sayıların üretilmesi pratik açıdan yazılım sektörüne bir yarar sağlamamaktadır. Bu nedenle, sonuçların temsil özelliğinin olduğunu (gerçek dünya ile uyumluluğunu) ve proje çalışmaları açısından nasıl yarar sağlayabileceklerini (çeşitli kararların verilmesi, yazılımlarının iyileştirilmesi veya maliyetleri düşürülmesi gibi) göstermek önemlidir.

### Kullanılacak olan yöntem:

- Ders notlarının 5, 6, 7, 8 ve 10. bölümlerinde anlatılan yöntemlere benzer yaklaşımlar kullanılabilir. Ancak burada verilen yöntemlerin hiçbirini aynı veya çok benzer şekilde kullanmayınız.
- Eğer benzer bir yöntem kullanacaksanız kullanılan kalite niteliklerinde, ölçme elemanlarında (QME), katsayılarda, referans değerlerde veya yöntemlerde anlamlı değişiklikler yapınız ve bunları açıklayınız.
- Büyük dil modellerinin (LLM) ve üretken yapay zekâ yöntemlerinin (*generative AI*) yazılım kalitesinin ölçülmesinde ve değerlendirmesinde kullanımına yönelik çalışma yapabilirsiniz.

### İncelenecek olan yazılım projesi:

- Yazılımlar nesneye dayalı yöntemle göre geliştirilmiş olmalı. Bu yöntemi destekleyen herhangi bir dil kullanılmış olabilir (C++, Java, C# gibi).
- Yazılımda en az 50 yazılım sınıfı olmalı (kütüphane sınıfları dışında).
- İncelenen yazılım sistemleri servis odaklı veya mikroservis mimarisine göre tasarlanmış olabilir.
- Yazılım projelerini aşağıdaki iki kaynaktan (A, B) alabilirsiniz.

#### A. Üzerinde çalıştığınız (çalıştığınız firmada geliştirilen) bir yazılımı inceleyebilirsiniz.

- Bu durumda elde ettiğiniz sonuçları yazılımı geliştiren ekip ile tartışıp sınavabilirsiniz.
- Mutlaka firmanın yetkili kişisinden yazılımı incelemek için izin alınız.**
- Yazılımın ve yazılımdaki bileşenlerin (sınıf, metod) adını ve gizleyiniz, ancak kullanım amacını belirtiniz.

#### B. Açık kaynaklı yazılımlar:

- İnternet'te birçok yazılımın kaynak koduna ve hata raporlarına erişebilirsiniz. Günümüzde GitHub (<https://github.com>) bu tür verileri elde etmek için uygun bir ortamdır.
- Metrikler yardımıyla elde ettiğiniz sonuçları sınamak için projelerin kaynak kodlarını inceleyebilirsiniz veya hata raporlarından yararlanabilirsiniz.

**Kullanılacak olan metrikler:**

- Çalışmaya uygun şekilde farklı türlerde (kod/tasarım, evrimsel, programcı tabanlı) metrikler kullanabilirsiniz.
- Yazılımlardan metrik toplamak için dersin web sayfasında (Ninova) bulunan veya kendi bulduğunuz ücretli/ücretsiz herhangi bir aracı kullanabilirsiniz.
- Daha önceki bilimsel çalışmalarda oluşturulmuş olan ve kullanıma açılan hazır metrik kümelerini kullanabilirsiniz. Yazılım dünyasında yaygın olarak kullanılan kümeler dersin web sayfasında (Ninova) yer almaktadır.

**Örnek çalışmalar:**

Proje kapsamında yapılabilecek çalışmalara ışık tutması için aşağıdaki örnekler verilmiştir.

- Ç. Biray, F. Buzluca, "A learning-based method for detecting defective classes in object-oriented systems", 2015 IEEE Eighth International Conference on Software Testing, Verification and Validation Workshops (ICSTW), Graz, Austria, April 17, 2015.  
<https://doi.org/10.1109/ICSTW.2015.7107477>
- S. Eski and F. Buzluca, "An Empirical Study on Object-Oriented Metrics and Software Evolution in Order to Reduce Testing Costs by Predicting Change-Prone Classes," in *2011 IEEE Fourth International Conference on Software Testing, Verification and Validation Workshops*, Berlin, Germany, 2011, pp. 566–571.  
<https://doi.org/10.1109/ICSTW.2011.43>
- F. Arcelli Fontana, M. V. Mäntylä, M. Zanoni, and A. Marino, "Comparing and experimenting machine learning techniques for code smell detection," *Empirical Software Engineering*, vol. 21, no. 3, pp. 1143–1191, Jun. 2016.  
<https://doi.org/10.1007/s10664-015-9378-4>
- G. Catolino, F. Palomba, A. De Lucia, F. Ferrucci, and A. Zaidman, "Developer-Related Factors in Change Prediction: An Empirical Assessment," 2017, pp. 186–195.  
<https://doi.org/10.1109/ICPC.2017.19>
- F. Aktas and F. Buzluca, "A Learning-Based Bug Prediction Method for Object-Oriented Systems," in *2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS)*, Singapore, Singapore, 2018, pp. 217–223.  
<https://doi.org/10.1109/ICIS.2018.8466535>
- R. Yılmaz, F. Buzluca, "A Fuzzy Logic-Based Quality Model for Identifying Microservices with Low Maintainability", *Journal of Systems and Software*, Elsevier, Vol. 216, 2024.  
<https://doi.org/10.1016/j.jss.2024.112143>
- R. Yılmaz, A.H. Köse, F. Buzluca, A Deep Learning-based Approach for Predictive Evaluation of Microservice Maintainability, Preprint, Research Square, 14 April 2024.  
<https://doi.org/10.21203/rs.3.rs-4243688/v1>

**Rapor:**

- Yaptığınız çalışmayı açıklayan bir raporu iki sütunlu IEEE bildirisi (konferans) formatında ve A4 boyutunda hazırlayınız.  
<https://www.ieee.org/conferences/publishing/templates.html>
- Raporda aşağıdaki bölümler yer alacaktır:  
Giriş, ilgili çalışmalar (literatür taraması), kullanılan/geliştirilen yaklaşım, incelenen yazılımlarla ilgili temel bilgiler, sonuçlar, yorumlar, referanslar.
- Rapor uzunluğunun alt veya üst sınırı yoktur.

**Sunum:**

- Çalışmanızda önerdiğiniz yöntemi ve sonuçları açıklayan **10 dakika** sürecek bir sunum hazırlayınız.

**Değerlendirme ölçütleri:**

- Öğrencinin yöneme katkısı,
- Yöntemin geçerliliği ve yazılım projesine uygulanma şeklinin uygunluğu,
- Sonuçların yorumlanması ve gerçek yazılımla ilişkisinin kurulması,
- Raporun içerik ve biçim yönünden uygunluğu.
- Uygulanan yöntem, sonuçlar ve yorumların sunumda dinleyicilere aktarılması,
- Sorulan sorulara verilen yanıtların uygunluğu

**Önemli Tarihler:**

- Raporların (pdf) ve sunum dosyalarının (pdf veya pptx) Ninova'ya yüklenmesi:  
En geç **29 Aralık 2025 Pazartesi saat 23.59'**a kadar.
- Geç gönderilen dosyalar kabul edilmeyecektir. Lütfen gönderimi son dakikaya bırakmayınız.
- **Ödevlerinizi e-posta ile göndermeyiniz.** Sadece dersin resmî sitesi olan Ninova'ya zamanında yüklenen dosyalar kayıt altına alınmakta ve kabul edilmektedir.
- Sunumlar, **31 Aralık 2025** günü yapılacaktır.  
Sunum programı 30 Aralık günü duyurulacaktır.