Solid Prensipleri

1. (S)ingle Responsibility

2. (O)pen/Closed

3. (L)iskov ‘s Substitution

4. (I)nterface Segregation

5. (D)ependency Inversion

# Single Responsibility

Projenin içerisindeki her bir yapı, diğer yapılara o kadar bağımlıdır ve yapılan işler o kadar iç

içe geçmiştir ki; küçük bir değişikliğin neleri etkileyeceğini kestirmeniz çok zordur ve genelde

böyle projeler çöp proje olarak görülür. İçerisindeki class’ları methodları alıp başka bir

projede kullanamazsınız.

Eğer tek sorumluluk prensibine uyarsanız bu şekilde binlerce satırlık class’larınız

methodlarınız olmaz. Her class’ın, her mothodun sadece tek bir yaptığı iş vardır, böylece bir

değişiklik yapmak için sadece bir nedeniniz olmuş olur. Genişleyebilir, tekrar kullanılabilir ve

test edilebilir yapılar kurmak için tek sorumluluk ilkesini dikkate almamız gerekir.

# Open/Closed Principle

Açık kapalı prensibi, yazılım geliştirirken kullandığımız varlıkların (class, method vs.) gelişime

açık, kodların ise değişime kapalı olması ilkesidir. Örneğin; bir loglama altyapısı

oluşturduğunuzu düşünün, Veritabanına ve XML’e kayıt tutuyorsunuz. Daha sonradan

Eventloglara da log tutma ihtiyacı hissettiğinizde, sadece Eventloglara kayıt tutan kodları

yazmanız yetecek, kodunuzda hiçbir değişiklik yapmadan bu yapı sisteme entegre olacak.

# Liskov ‘s Substitution

Türeyen sınıf yani alt sınıflar ana(üst) sınıfın tüm özelliklerini ve metotlarını aynı

işlevi gösterecek şekilde kullanabilme ve kendine ait yeni özellikler

barındırabilmelidir.

# Interface Segregation

Arayüz ayırım prensibi, bir arayüze gerektiğinden fazla yetenek eklemememiz gerektiği

söyler.

# Dependency Inversion

Bağımlılığın ters çevirilmesi ilkesine göre üst seviye sınıflar, modüller, methodlar vs. alt

seviyeli sınıflara bağımlı olmamalıdır. Alt sınıflarda yapılan değişiklikler üst sınıfları

etkilememelidir.

# Takip ettiğim yazılımcılar

1-DARON YONDEM

2-BURAK SELİM ŞENYURT

3-GENÇAY YILDIZ

4-ŞADİ EVREB ŞEKER

# MİCROSOFT Build 2020 Yenilikleri

Build 2020’de Azure Synapse Link tanıtıldı. Maliyetleri düşüren ve zaman kazandıran Azure

Synapse Link, veri hareketlerini yönetmeye gerek kalmadan müşterilerin değerli bilgiler elde

etmesini sağlıyor.

Geliştiricilerin Visual Studio ve Visual Studio Code’dan Teams uygulamaları oluşturmaları ve

yayınlamaları için BT yöneticilerinin Microsoft Teams’teki kullanıcıların iş kollarını ve ISV

uygulamalarını değerlendirmesine ve iş dağılımı yapmasına olanak tanıyan yenilikler yapıldı.

Azure Machine Learning ve OSS araçlarına getirilen yeniliklerle müşterilerin yapay zekâ

modellerini daha sorumlu bir şekilde kullanmasına yardımcı olmak için yeni Responsible ML.

araçları sunuldu. Bu araç, model yorumlama yeteneğini geliştirerek veri güvenliğini ve

kullanıcı gizliliği garanti altına alacak, yapay zeka sistemlerinin geliştirilmesinde sorumluluk

anlayışını güçlendirecek.

# Yazılım ile ilgili yarışmalar

Bu siteden hackhatonları takip edebilirsiniz. Her dalda yarışmalar mevcut ve büyük potansiyelleri olan yarışmalar, oyun ve inovatif hackhatonlar benim favorimdir.

<https://hackathonturkiye.com/>

Bunuda unutmayalım Global Game Jam

<https://globalgamejam.org/>