

Bilgisayar Mühendisliği Yüksek Lisans Programı 2026 Bahar

Gereksinim Analizi, Modelleme ve OOP (Nesneye Yönelim Tasarım)
Temiz Kod (Clean Code), Refactoring ve SOLID Prensipleri

Dr.Öğr.Üyesi Zeynep ALTAN

Ürün- Pazar Uyumu/ Product-Market Fit

- ❑ Bir ürünün güçlü ve tanımlanabilir bir pazar talebini karşılama derecesidir.
 - Tek seferlik bir faaliyetten ziyade sürekli bir süreçtir.
 - **Ürün, müşteriler ve pazar** koşulları zaman içinde değişebilir
 - ✓ Ürün-pazar uyumunu korumak için ürünün mevcut koşullara uyarlaması gerekir.
- ❑ Ürün- pazar uyumu bir ürün veya hizmetin hedef pazarlarının ihtiyaçlarını karşılması ve aynı zamanda büyüme ve karlılığı sürdürmesi durumunda ortaya çıkar.

Ürün/Pazar Uyumlu

- ❑ **Ürün başarısı ve pazarın geçerliliği** arasındaki ilişki, bir ürünün, hizmetin veya iş modelinin pazarda ne kadar başarılı ve karlı olabileceğini değerlendirmede kritik öneme sahiptir.
- ❑ Pazarın geçerliliği, bir ürün veya hizmetin belirli bir pazar içinde başarılı olma olasılığıdır.
 - ❖ Mali açıdan anlamlı bir iş kurmanın ilk adımlarından biridir.

Product:

UX

Feature Set

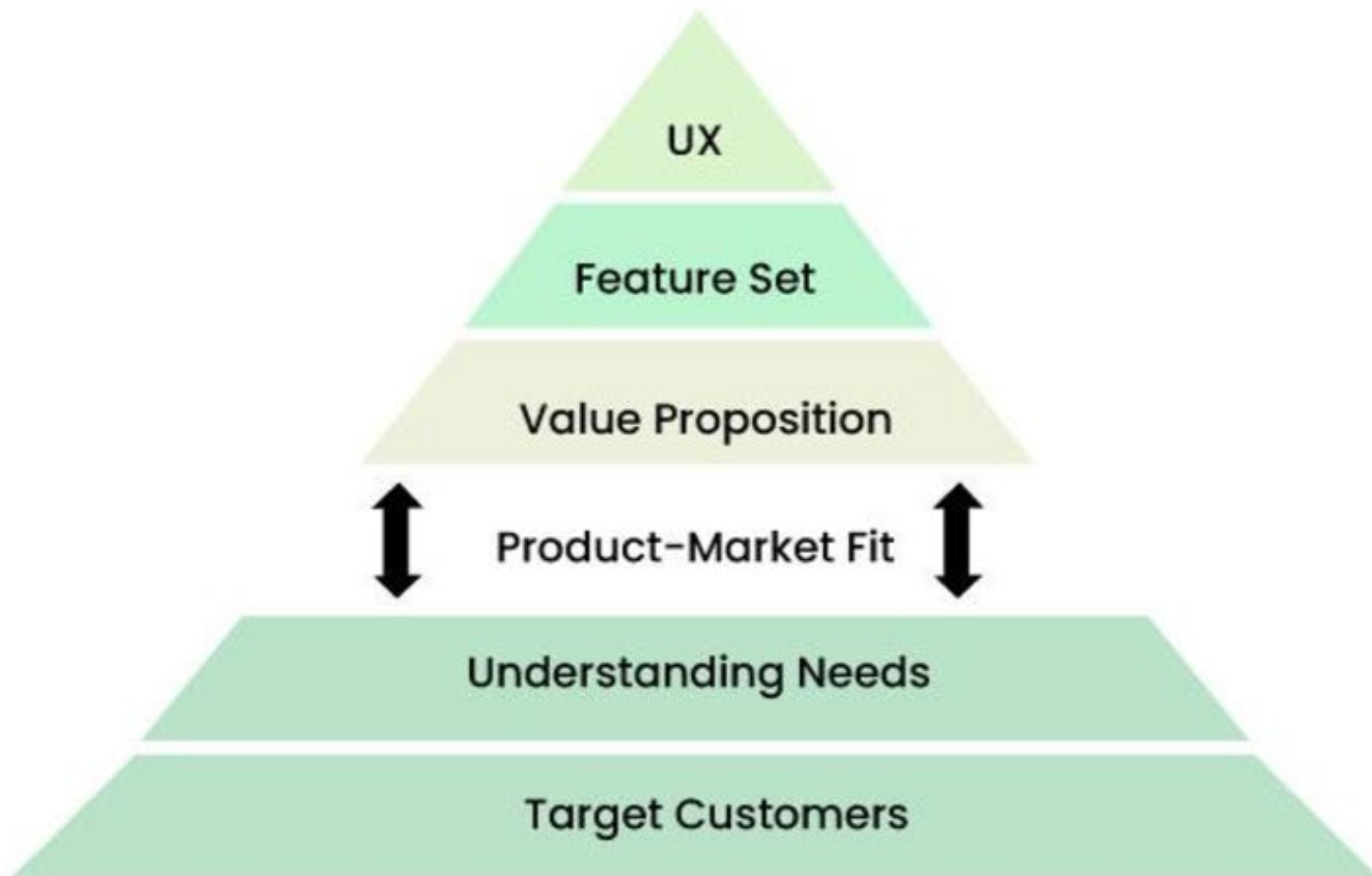
Value Proposition

Product-Market Fit

Market:

Understanding Needs

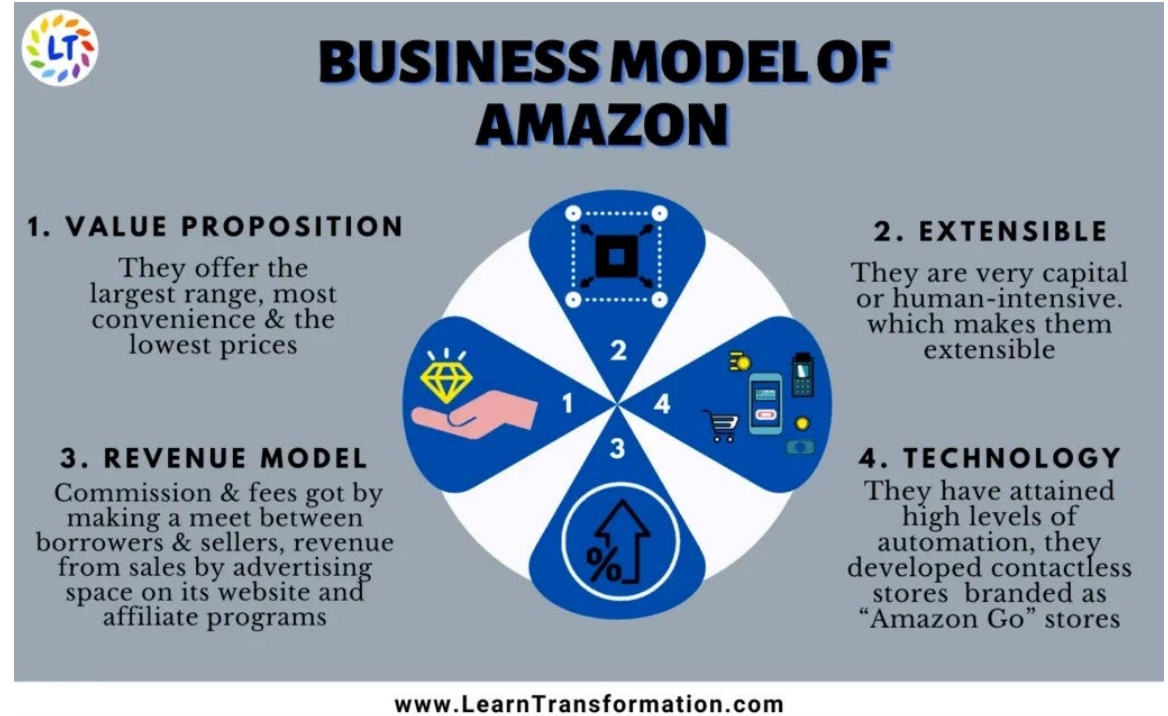
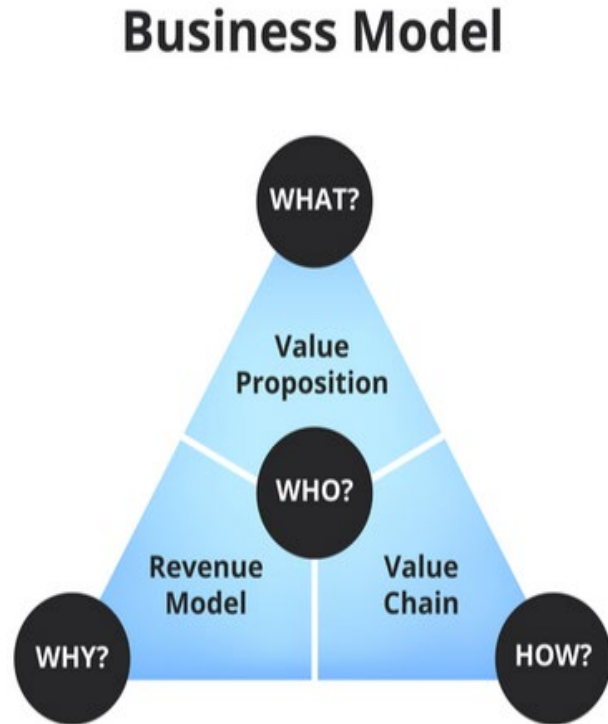
Target Customers



İş Modeli

□ İş modeli, şirketin nasıl değer yarattığını, sunduğunu ve yakaladığını açıklar.

❖ Şirketin nasıl para kazandığını açıklar.



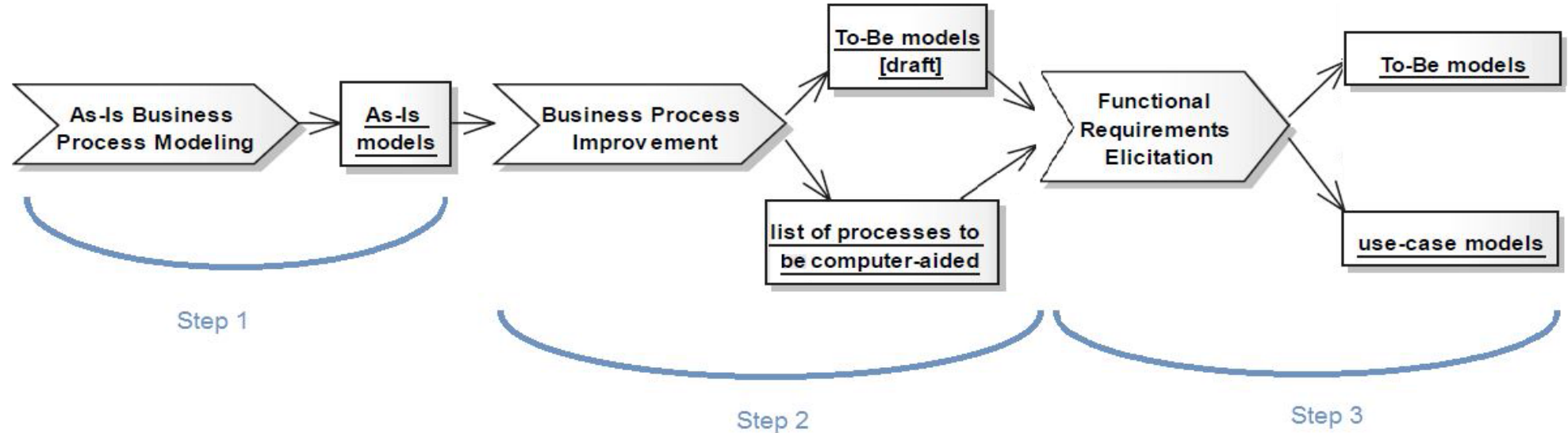
İş Süreci Modeli için Gereksinimlerin Çıkartılması

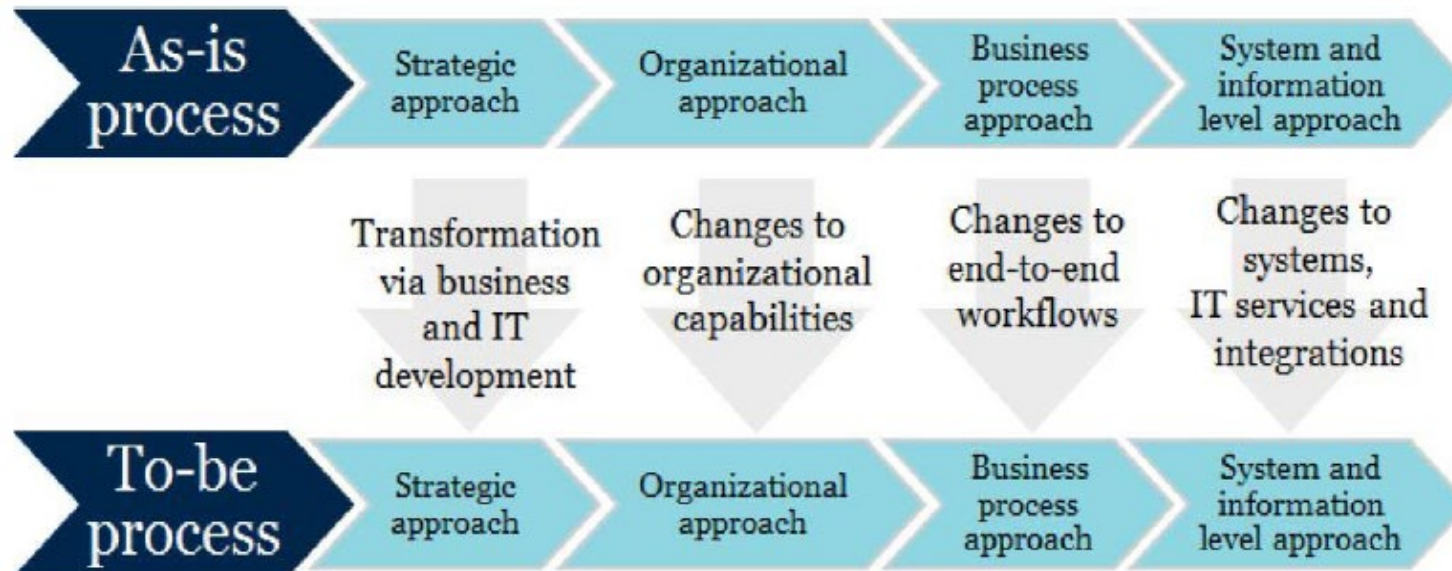
- ❑ Başarılı süreç modellemesi büyük ölçüde doğru gereksinim mühendisliğine bağlıdır.
- ❑ BORE (İş Odaklı Gereksinim Mühendisliği) yöntemi ile BDD (İş Odaklı Geliştirme) yönteminin özellikleri birleştirilerek karmaşık süreçler modellenebilir.

BORE / Business Oriented Requirements Engineering

Müşteri ihtiyaçları konusunda belirsizlik koşullarında gereksinim belirleme sürecini desteklenir.

Sistem gereksinimlerinin temel iş süreçleriyle uyumlu olması ve bu süreçleri desteklemesini sağlar.





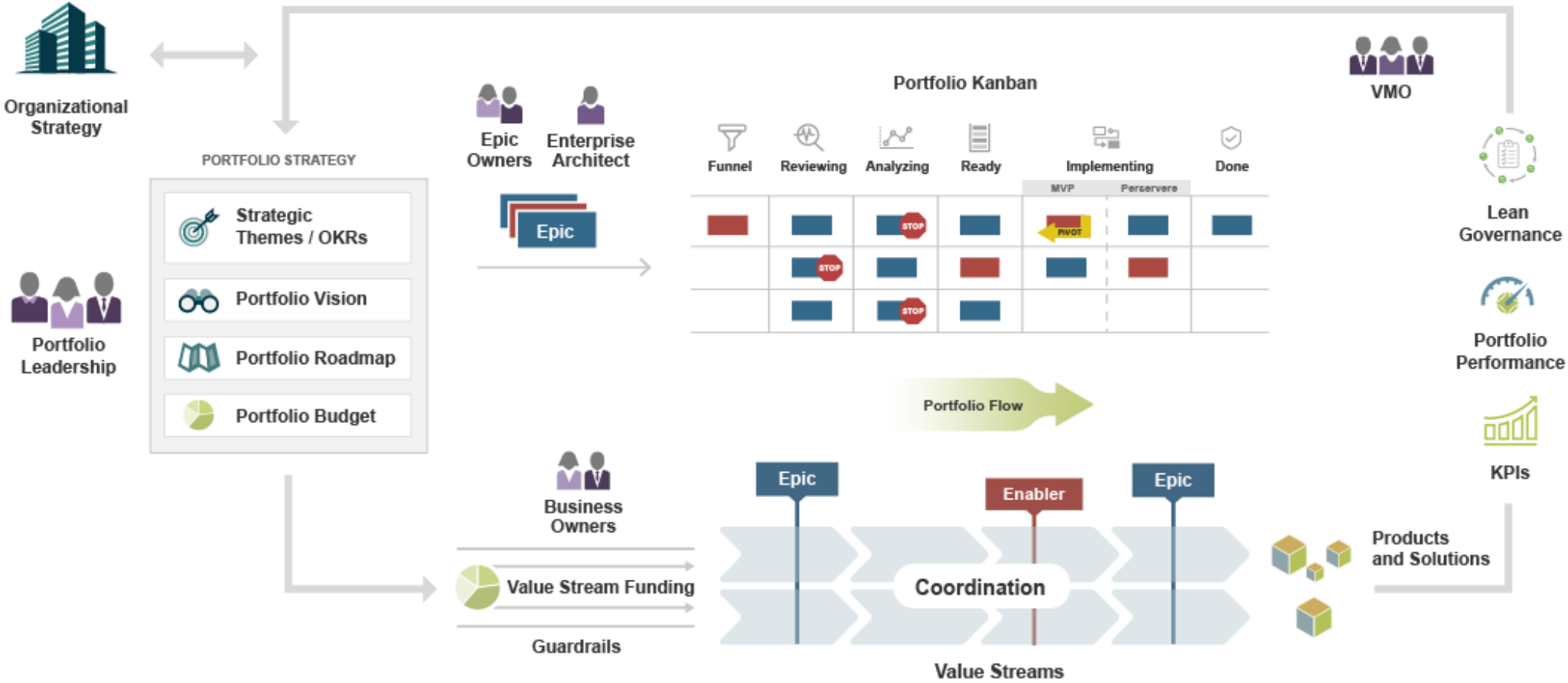
Business Driven Development (BDD)

- ❑ Yazılımda iş odaklı geliştirme (BDD), teknolojiden ziyade işletmenin ihtiyaçlarına odaklanır.
- ❑ Tüketici uygulamaları için değil, kurumsal yazılımlar için tasarlanmıştır.
- ❑ Kurumların ürün odaklı bir stratejiye geçerken birçok zorluğu aşmaları gerekir.
 - ❖ Örneğin, ürünleri müşteri ihtiyaçlarıyla uyumlu hale getirmenin satış performansını nasıl artırabileceği ve müşteri odaklı değerlere dayalı güçlü bir marka kimliği nasıl oluşturabileceği araştırılmalıdır.
- ❑ Mevcut portföyler sürekli olarak değerlendirilmeli, pazar talepleriyle uyum fırsatları belirlenmeli ve ürünleri ve müşterileri ön plana çıkaran bir strateji geliştirilmelidir.

Ürün odaklı Portföyler İş Çevikliği Dönüşümü

- ❑Yapay zeka, sürekli iyileştirmeyi ve yeniliğe müşteri odaklı bir yaklaşımı teşvik eden ürün odaklı bir ortamda gelişir.
- ❑stratejik uyumu sağlayan, hızlı inovasyon, çeviklik ve ürün yatırımları ile iş hedefleri arasında daha güçlü bir bağlantı Yalın Portföy Yönetimi'nin (Lean Portfolio Management) ile sağlanabilir.
- ❑Projelerden ürünlere nasıl geçileceği ve yapay zeka etkili bir şekilde nasıl kullanılacağı önemlidir.

Business Agility



OKR (Objectives and Key Results)

- Hedefler ve Anahtar Sonuçlar, şirketlerin ve ekiplerin iddialı hedeflere odaklanmasını, şeffaflığı ve ölçülebilir sonuçları sağlayan bir performans yönetim metodolojisidir.
 - ❖ **Objectives** (Hedefler - Nereye gitmek istiyoruz?): İlham verici, net ve iddialı hedeflerdir.
 - ❖ **Key Results** (Anahtar Sonuçlar - Oraya nasıl ulaşacağız?): Hedefe ulaşıldığını kanıtlayan, ölçülebilir ve somut sonuçlardır

OKR Örneği -1

The city of Syracuse, New York recently set this **Objective**: to “achieve fiscal sustainability.”

Fiscal sustainability is a great goal, but it has to be measurable.

That’s why Syracuse uses Objectives and Key Results.

When written out, Syracuse’s OKR looks like this:

O: Achieve fiscal sustainability.

Kr1: Reduce the general fund budget variance from 11% to 5%.

Kr2: Spend 95% of authorized capital project dollars by the end of the fiscal year.

Kr3: Spend 95% of grant dollars for grants from prior fiscal years.

OKR-Örneği -2

Allbirds is a popular clothing company known for their sustainable footwear.

They have a mission to “make better things in a better way.”

As a part of their commitment to sustainability, they decided they wanted their shoes to be carbon neutral.

So they wrote this OKR:

- O: Create the lowest carbon footprint in our industry.
- Kr1: Supply chain and shipping infrastructure 100% zero waste.
- Kr2: Pay 100% carbon offset for calculated carbon dioxide emissions.
- Kr3: 25% of material is compostable.
- Kr4: 75% of material is biodegradable.

OKR ile KPI farkı

OKR : Hedefleri ölçülebilir sonuçlara bağlayan bir hedef belirleme çerçevesidir.

Stratejik hedeflere yönelik anlamlı ilerleme sağlar.

Zaman sınırı (genellikle üç aylık veya stratejik döngülere bağlı) vardır.

Dinamik; öncelikler değiştikçe uyum sağlayacak şekilde tasarlanmıştır.

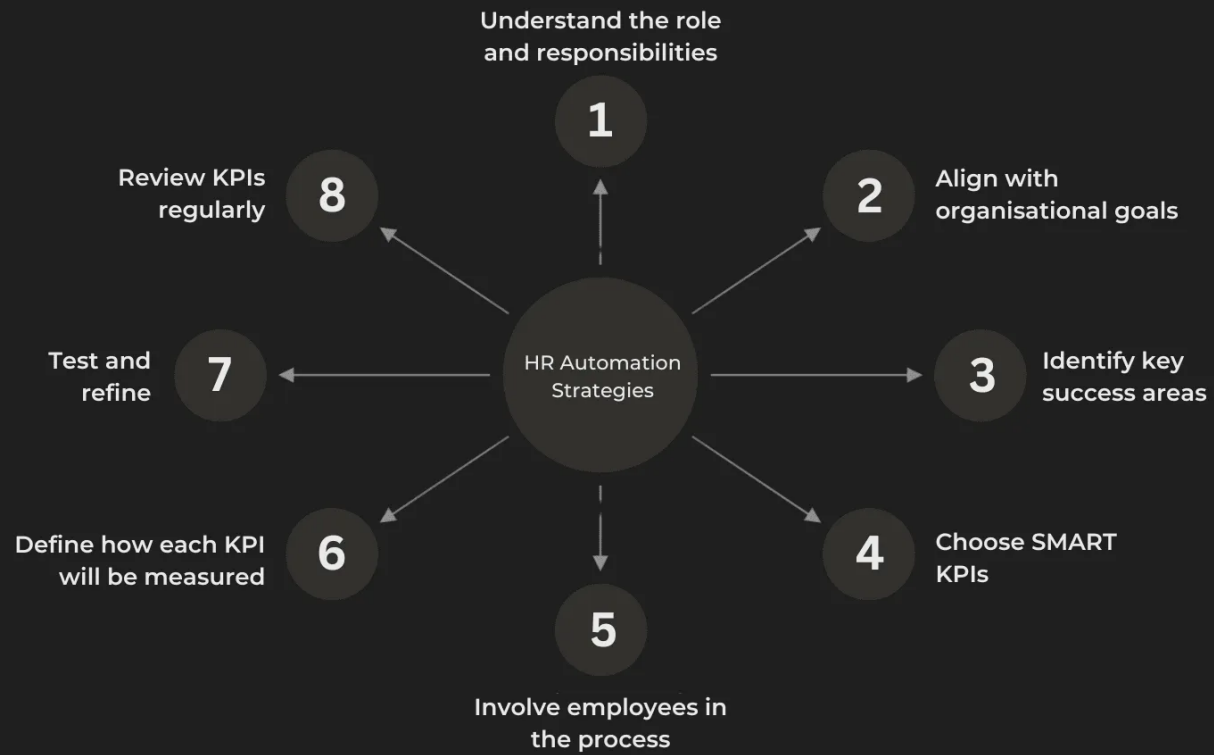
KPI : Mevcut performansı yansıtan, belirli bir hedefi olan bir ölçüttür.

Performansı ve trendleri ölçer.

Belirli bir hedefi olan nicel ölçüttür.

Devamlılığı vardır.

How to Develop KPIs for Effective Employee Performance Management



Üretkenlikle ilgili KPI

Çalışan her ay kendisine verilen görevlerin %95'ini tamamlıyor.

Müşteri hizmetleri temsilcisi günde ortalama 30 destek talebiyle ilgileniyor.

İçerik yazarı haftada dört adet yüksek kaliteli makale üretiyor.

İşin Niteliği ile ilgili KPI

Tasarımcı, gönderilen dosyalarda %2'den daha düşük bir hata oranıyla çalışır.

Proje yöneticisi, sürekli olarak yeniden çalışma gerektirmeyen işler teslim eder.

Çalışanlar, 8 veya daha yüksek bir puanla olumlu akran değerlendirmeleri alırlar.

Hedef Başarısı – Hedef Başarısının Değerlendirilmesi ile ilgili KPI

Satış elemanı, üç aylık satış hedefinin %90'ına ulaşıyor.

Pazarlama ekibi, bir ürün lansmanı için tüm kampanya son tarihlerine uyuyor.

İnsan kaynakları uzmanı, planlandığı gibi bir ay içinde beş pozisyon için işe alım sürecini tamamlıyor.

Zaman Yönetimi ile ilgili KPI

Çalışan üç ay boyunca tüm raporları son teslim tarihine kadar gönderiyor.

Ekip üyesi, dahili taleplere ortalama 24 saat içinde yanıt veriyor.

Mühendis, sprint görevlerini planlanan zaman dilimi içinde sürekli olarak tamamlıyor.

Nesneye Yönelik Tasarım

- ❑ Nesne yönelimli tasarım (OOD), sınıfların örnekleri olan "nesneler" kavramına odaklanan bir programlama paradigmasıdır.
- ❑ Bu nesneler, veri ve davranışları kapsayarak yazılım geliştirme için modüler ve yeniden kullanılabilir bir yapı sağlar.

Nesneye Yönelik Tasarımın Anahtar Bileşenleri

Objects

Classes

Encapsulation

Inheritance

Polymorphism

Abstraction

Kapsülleme

- ❑ Nesnenin içsel durumunu gizler ve tüm etkileşimlerin onun metotları üzerinden gerçekleşmesini gerektirir.
- ❑ Kontrollü bir arayüz sağlar ve istenmeyen müdahale ve nesnenin içsel durumunun kötüye kullanılma olasılığını azaltır.
- ❑ Örneğin, bir nesnenin doğrudan erişilemeyen ancak public metotlar aracılığıyla değiştirilebilen veya okunabilen özel bir öz niteliği olabilir.

Kapsülleme ve polimorfizm daha çok nesnelere bağlantılıyken, kalıtım ve soyutlama sınıflarla bağlantılıdır.

Polimorfizm

Farklı sınıflara ait nesnelere ortak bir üst sınıfın parçası olarak ele alınmasını sağlar.

Nesneler aynı adı paylaşırsa bile, yöntemlerin etki ettikleri nesnelere bağlı olarak farklı işlevler gerçekleştirebilir.

Bir alt sınıf, bir metodunun implementasyonunu üst sınıfından «overriding» yoluyla elde eder.

Kalıtım ve Soyutlama

Kalıtım: yeni bir sınıfın mevcut bir sınıfın niteliklerini ve metotlarını miras almasına olanak tanır.

Kodun tekrar kullanımını teşvik eder ve sınıflar arasında doğal bir hiyerarşik ilişki kurar.

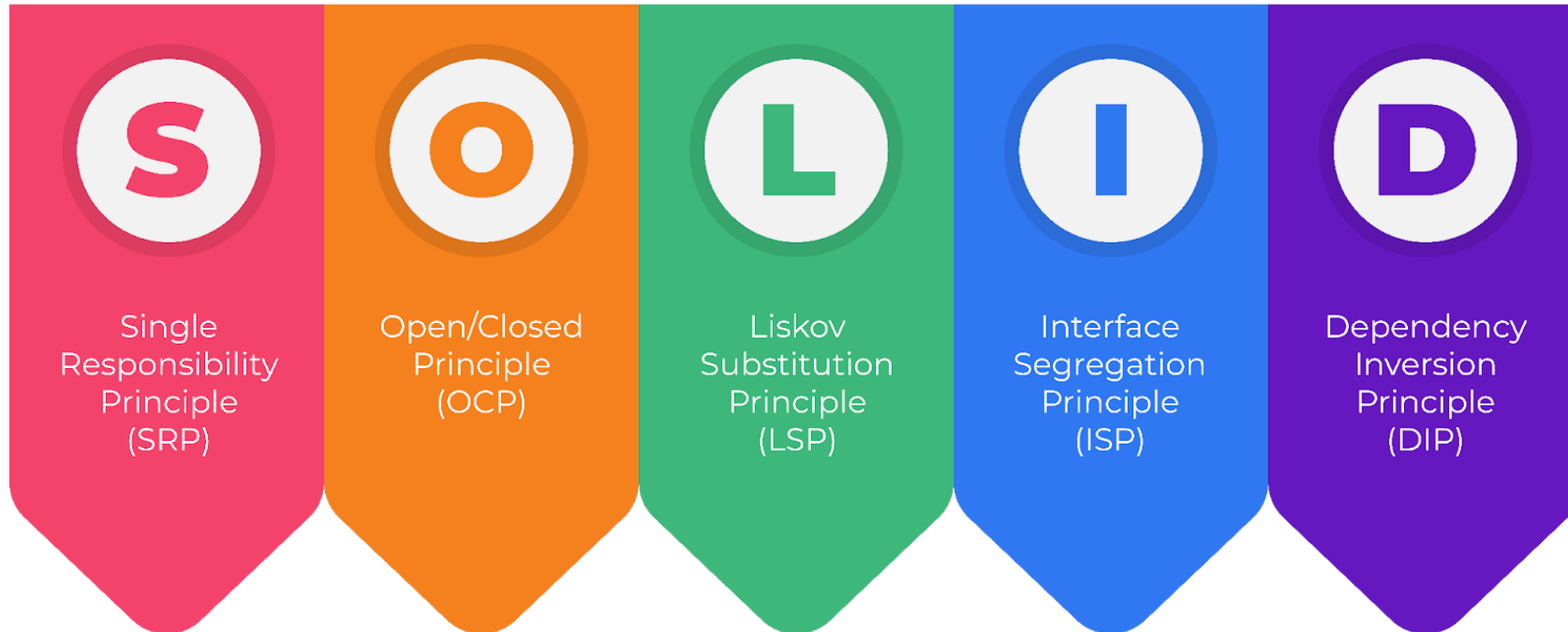
Örneğin, bir "Araç" sınıfı varsa, bir "Araba" sınıfı "Araç"tan miras alarak niteliklerini ve metotlarını alabilir ve arabalara özgü belirli özellikler ekleyebilir.

Soyutlama: Probleme uygun sınıfları modelleyerek ve uygulama ayrıntılarıyla uğraşmak yerine daha yüksek bir seviyede çalışarak karmaşık sistemleri basitleştirir.

Tasarımcılar, mevcut probleme uygun bir nesnenin temel niteliklerine odaklanabilirler.

Nesneye Yönelik Tasarım

Principles That Guide OOD



Single Responsibility Principle (SRP)	Open/Closed Principle (OCP)	Liskov Substitution Principle (LSP)	Interface Segregation Principle (ISP)	Dependency Inversion Principle (DIP)
A class should have only one reason to change, meaning it should only have one job or responsibility.	Software entities should be open for extension but closed for modification. When new functionality is added, existing code should not be changed.	Subtypes must be substitutable for their base types without altering the correctness of the program.	No client should be forced to depend on methods it does not use. It promotes the creation of smaller, more specific interfaces.	High-level modules should not depend on low-level ones. Both should depend on abstractions, and abstractions should not depend on details.