

---

# **SDLC**

## **Software Development Life Cycle**

**Gun 2**  
**19 eylul 2020**

**SDLC PHASSES - SDLC ASAMALARI**  
**WATERFALL METHODOLOGY**

---

---

# SDLC NEDİR

**SDLC** : **S**oftware **D**evelopment **L**ife **C**ycle  
Yazılım Gelistirme Yasam Dongusu

Yazılım endüstrisi tarafından **yüksek kaliteli** yazılım tasarlamak, geliştirmek ve test etmek için kullanılan bir süreçtir.

**Müşteri beklentilerini** karşılayan veya beklentilerin de ötesinde bir ürün ortaya koyan, öngörülen zaman içerisinde tamamlanan ve maliyeti (bütçe) doğru bir şekilde hesaplanan yazılım üretmeyi amaçlar.

TECH PRO ED

---

# SOFTWARE DEVELOPMENT PHASES

## YAZILIM GELİSTİRME ASAMALARI





---

# SOFTWARE DEVELOPMENT PHASES

## YAZILIM GELISTIRME ASAMALARI



---

# STANDART YAZILIM GELİŞTİRME YAŞAM DÖNGÜSÜ AŞAĞIDAKILERDEN OLUŞUR:

Stage 1: Planning and Requirement Analysis  
(Planlama ve İhtiyaç Analizi)

Stage 2: Defining Requirements  
(Gereksinimleri Tanımlama)

Stage 3: Designing the product architecture  
(Ürün dizaynını tasarlama)

Stage 4: Building or Developing the Product  
(Ürünü oluşturma veya geliştirme)

Stage 5: Testing the Product  
(Ürünü test etme)

Stage 6: Deployment in the Market and Maintenance  
(Ürünü pazarlama ve Bakım)

---



---

# STAGE 1:

## PLANNING AND REQUIREMENT ANALYSIS

ilk toplantıda belirlenen **HIGH LEVEL** hedefler  
APPLICATION KONUSU : eğitim

- 1) Her insane ulaşabilecek bir eğitim
- 2) Öğrenme garantisi
- 3) Üniversite kazanma garantisi
- 4) Sadece okul derslerinde değil sosyal alanda da başarı, uluslararası dolasına uygun bir eğitim



---

# STAGE 1:

## PLANNING AND REQUIREMENT ANALYSIS

- İhtiyaç analizi SDLC'nin en önemli ve temel aşamasıdır.
  - Müşteriden gelen fikirler de göz önünde bulundurularak ekibin kıdemli üyeleri (expert) tarafından gerçekleştirilir.
  - Bu bilgiler daha sonra temel proje yaklaşımını planlamak için kullanılır.
  - Kalite güvence gerekliliklerinin planlanması ve projeye ilişkili risklerin belirlenmesi de planlama aşamasında yapılır.
  - Minimum risklerle projeyi başarıyla uygulamak için izlenebilecek teknik yaklaşımlar planlanır.
-



---

## STAGE 2:

# DEFINING REQUIREMENTS(FEASIBILITY STUDY)

ilk toplantıda belirlenen **high level** hedeflerin detayları oluşturulur

- 1) Her insana ulaşabilecek bir eğitim  
ucreti makul  
kullanım kolaylığı  
bolgesel öğretmenler  
yas aralıklarına göre eğitim planları çıkarma  
son teknolojiyi kullanma  
zengin ders seçeneği  
basit ve verimli eğitim  
farklı zaman dilimlerine göre eğitim planları





## STAGE 2:

# DEFINING REQUIREMENTS(FEASIBILITY STUDY)

- İhtiyaç analizi yapıldıktan sonraki adım, ürün gereksinimlerini açıkça tanımlamak ve belgelendirmektir (dokumante etmek)
- Stakeholder / işletmeciden onay alınır.
- Bu proje yaşam döngüsü boyunca tasarlanacak ve geliştirilecek tüm ürün gereksinimlerini içeren '**BRD**' (Business Requirement Document) İş Gereksinimi Belgesi ile yapılır.
- **FRD** Functional Requirement Document  
Teknik İşlev İhtiyaçları Dokumani hazırlanır
- FRD ve BRD en küçük User Case lere kadar hazırlanır

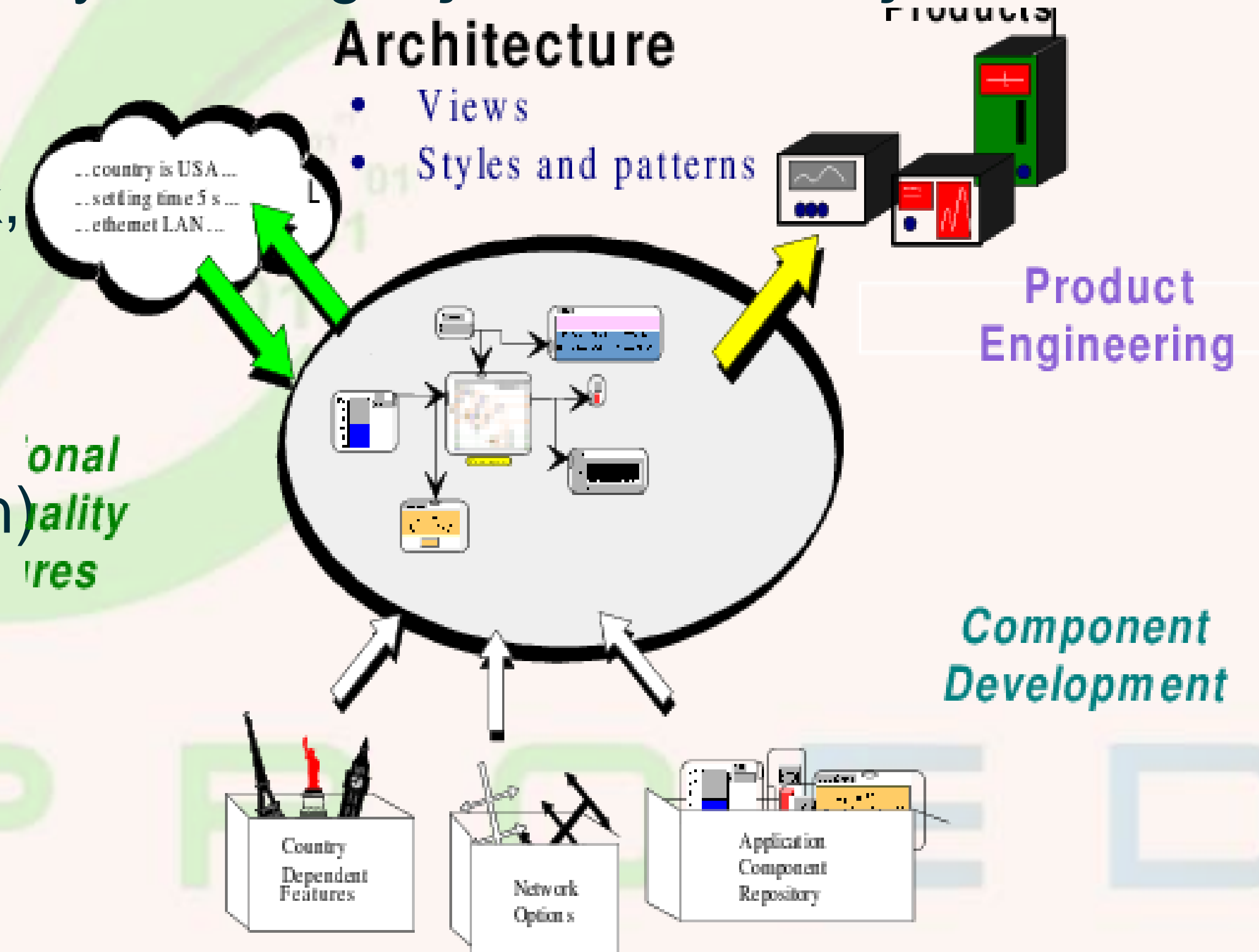


# STAGE 3: DESIGNING THE PRODUCT ARCHITECTURE:

- BRD (Business Requirement Document) Dizaynırların geliştirilecek ürün için en iyi dizaynla ortaya çıkacakları referanstır.

- BRD'de belirtilen gereksinimlere dayanarak, ürün mimarisi için genellikle birden fazla tasarım yaklaşımı taslagi oluşturulur

ve bir DDS (Design Document Specification) Tasarım Belgesi Spesifikasyonu'nda belgelenmektedir.





# STAGE 4: BUILDING OR DEVELOPING THE PRODUCT:

- SDLC'nin bu aşamasında gerçek gelişme başlar ve ürün inşa edilir.
- Yazılımcılar (Developers), kuruluşları tarafından tanımlanan kodlama yönergelerine uymak zorundadır.
- Kodlama için FRD baz alınarak Developerlar gereken Functionality'leri olustururlar
- Kodlama için C ++, Java, dot Net vs. gibi farklı üst düzey programlama dilleri kullanılır.



```
public static void main(String[] args) {  
    System.out.println("Hello world");  
}
```

---

# STAGE 5: TESTING THE PRODUCT:

- Bu aşama, ürün BRD'de tanımlanan kalite standartlarına ulaşıncaya kadar, ürün kusurlarının rapor edildiği, izlendiği, düzeltildi(fixlendiği) ve tekrar test edildiği asamadır.
- Ürün iş beklentilerini de karşılamalıdır (requirement specifications)
- STLC => Software Testing Life Cycle
- Test → Takip → Bulunan Hatanın Dev.Gonderilmesi → raporlama  
Düzeltilme → Yeniden Test etme → Onaylama → raporlama





# STAGE 5: TESTING THE PRODUCT:

1. *“https://www.saucedemo.com” Adresine gidin*
2. *Username kutusuna “standard\_user” yazdirin*
3. *Password kutusuna “secret\_sauce” yazdirin*
4. *Login tusuna basin*
5. *Ilk urunun ismini kaydedin ve bu urunun sayfasina gidin*
6. *Add to Cart butonuna basin*
7. *Alisveris sepetine tiklayin*
8. *Sectiginiz urunun basarili olarak sepete eklendigini control edin*
9. *Sayfayi kapatın*

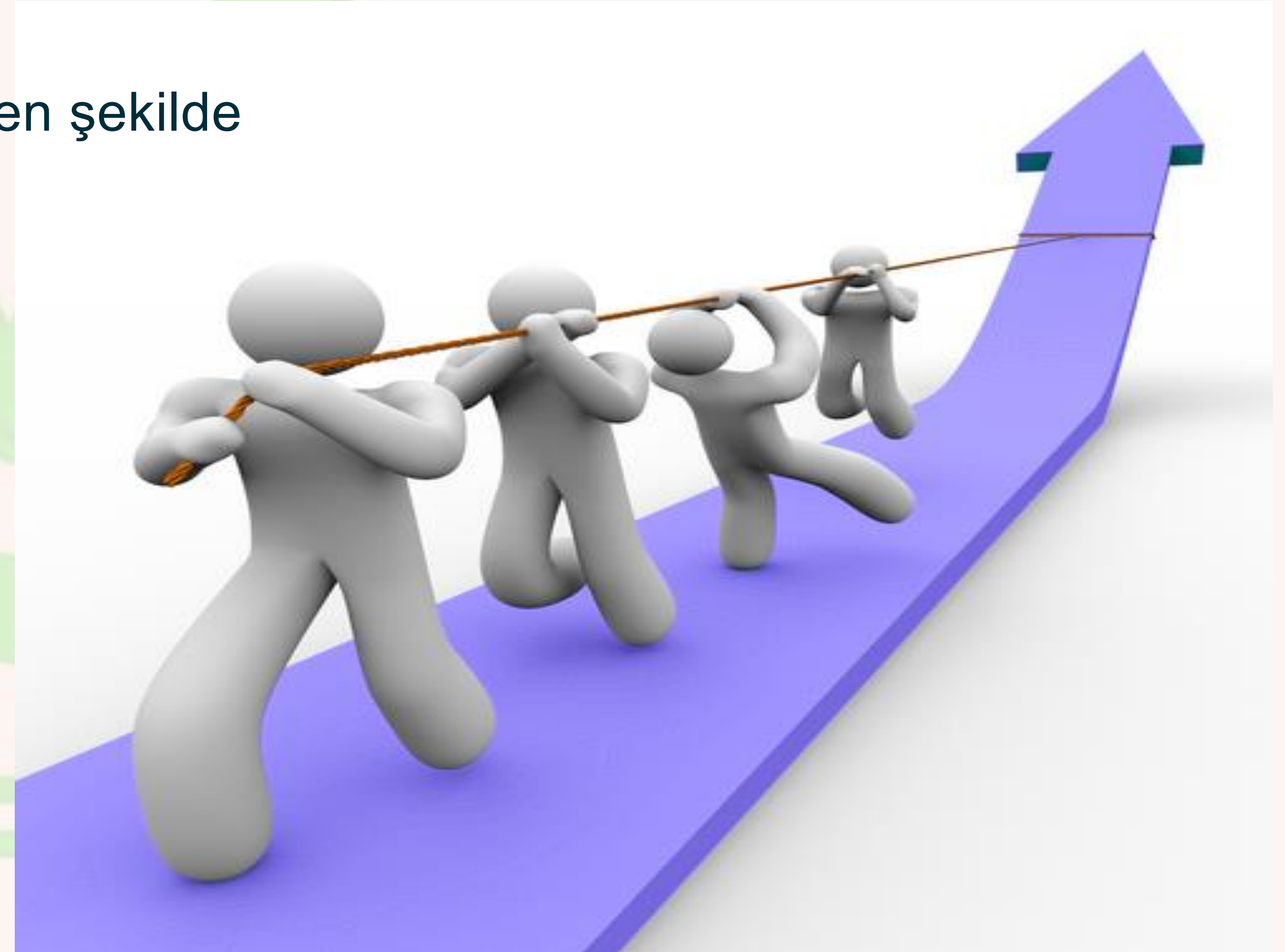
A screenshot of the login page on the Saucedemo.com website. The page has a white background. At the top right, there is a red circular icon with a white lightning bolt and the text "SWAG LABS" in red. Below this, there are two input fields: "Username" and "Password". Below the "Password" field is a red button with the text "LOGIN" in white. To the right of the login form is a cartoon character of a red robot with a blue visor, wearing a grey hoodie with a red "S" on it, and holding a red flag with a white lightning bolt. The robot is standing on a grey shadow.

---

## STAGE 6: DEPLOYMENT IN THE MARKET AND MAINTENANCE:

---

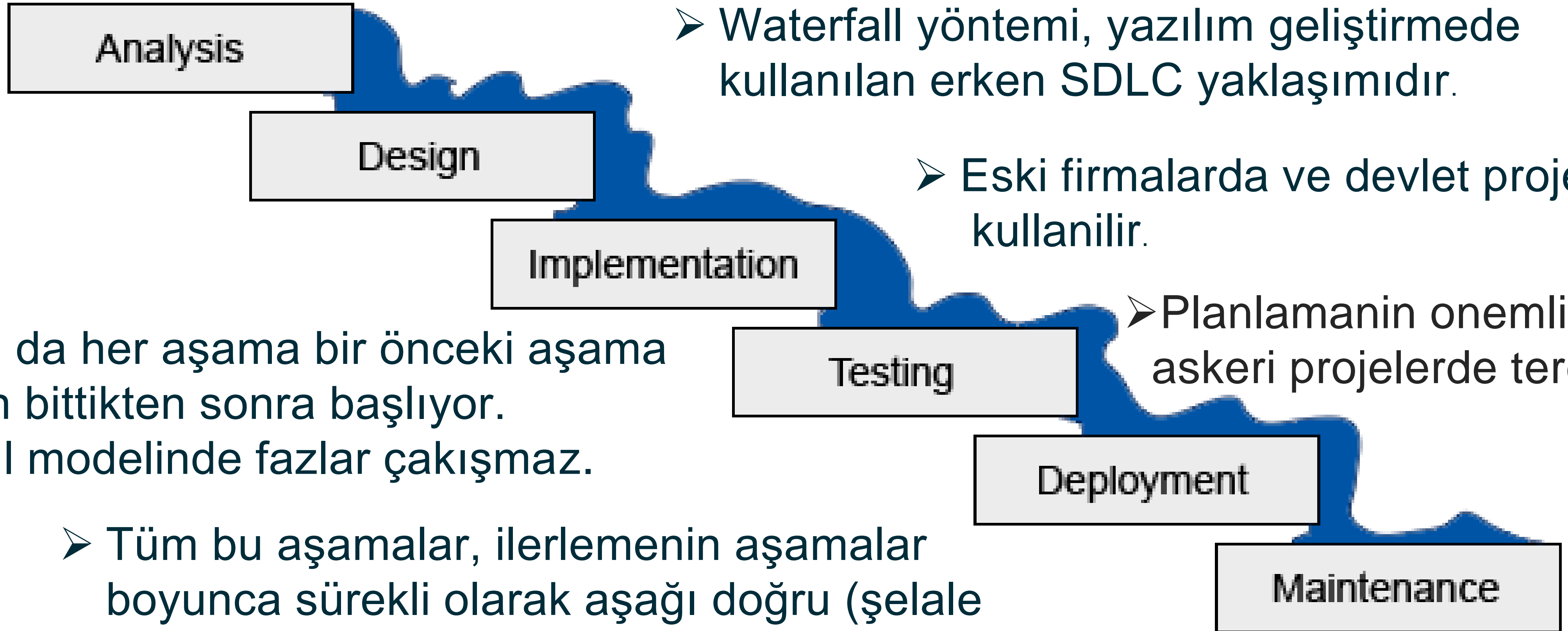
- Ürün test edildikten ve onaylandıktan sonra (hazır olduğunda), resmi olarak uygun görülen şekilde release edilir(piyasaya sürülür).
- Ürün piyasaya sunulduktan sonra mevcut müşteri tabanı için bakımı yapılır.
- Musteriden (End-User) gelen feedbackler ve Teknolojik Gelişmeler ile ihtiyaçlar yeniden belirlenir ve dongu yeniden baslatılır.



TECH



# WATERFALL METHOD (ŞELALE METODU)



➤ Waterfall yöntemi, yazılım geliştirmede kullanılan erken SDLC yaklaşımıdır.

➤ Eski firmalarda ve devlet projelerinde kullanılır.

➤ Planlamanın önemli olduğu askeri projelerde tercih edilir

➤ Waterfall da her aşama bir önceki aşama tamamen bittikten sonra başlıyor.  
Waterfall modelinde fazlar çakışmaz.

➤ Tüm bu aşamalar, ilerlemenin aşamalar boyunca sürekli olarak aşağı doğru (şelale gibi) aktığı şekilde kademelendirilir.

---

# WATERFALL METHOD

## Analiz Süreci

- Sistemden ne istenildiği ve kısıtların belirlendiği aşamadır. Bu asamaya “Gereksinim Mühendisliği (Requirements Engineering)” de denir.
  - Analiz süreci yazılım geliştirmenin sürecinin başlangıcı olduğu için tüm ihtiyaçların doğru şekilde anlaşılmış olması gerekmektedir.
  - Analiz dokümanları yalın bir şekilde hazırlanmalı ve herkesin anlayabileceği şekilde yazılmalıdır. Çünkü bu dokümanlara istinaden;
    - Yazılım geliştirme ekibinin müşterinin tam ihtiyacını anlaması için yardımcı olur,
    - Tasarım ekibine; bu analize istinaden sistemin nasıl tasarlanacağına yardımcı olur,
    - Test ekibine; bu analizlere istinaden hangi test case lerinin koşulacağına yardımcı olur.
  - Gereksinimler tamamlandıktan sonra dokümantasyonu müşteri ile mutabık kalınmalı ve bu doküman üzerinden müşteriden onay alınmalıdır.
-



---

# WATERFALL METHOD

## Tasarım Süreci

- Yazılımın temelini yani mimarisini oluşturmak için; ihtiyaçları çalışır sisteme çevirmek gerekmektedir.
  - Müşterinin ihtiyaçlarını karşılamak için nasıl yazılım geliştirme yapılacağını teknik olarak tasarlandığı süreçtir
  - Bu süreç iki farklı adımdan oluşmaktadır.
    - a) Sistemin net belirlenmesi: Müşteriye sistemin ne yapacağını anlatır. Buna “Kavramsal Sistem Tasarım” denir.
    - b) Teknik Tasarım: Müşteri kavramsal sistem tasarımı uygun görürse sistem kurucularına, programcılara, yazılım ve donanım kurmalarına izin veren “Teknik Tasarım” hazırlanır.
  - Tasarım sürecinde; eğer analiz sürecinde anlaşılmayan bir kısım varsa kesinlikle bu soru işareti analistlerle görüşülerek giderilmelidir.
-

# WATERFALL METHOD

## AVANTAJ VE DEZAVANTAJLARI

### Avantajlar:

- Proje bilgisini aktarmak daha kolaydır
- Projeyi yönetmek daha kolaydır
- Küçük projeler için daha iyidir
- Görevler mümkün olduğunca sabit kalır



### Dezavantajlar:

- Değişim ve yenilikler zordur
- Müşteri öngörü ve önerileri önemsizdir
- uzun süre devam eden projeler için zayıf model
- Testi tamamlanıncaya kadar projeyi geciktirir