

Yazılım
Mühendisliği
Bölüm - 3
Planlama



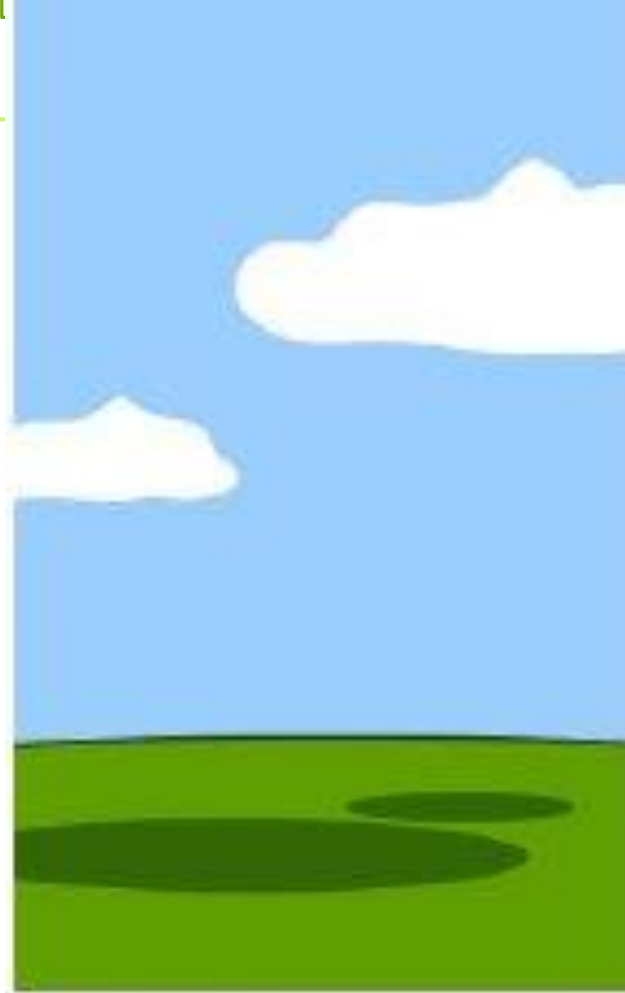
**Müsterinin
İstedığı**



**Analistin
Anladığı**



**Yazılımcının
Geliştirdiği**



**Projenin
Dokümantasyonu**



**Müşterinin Asıl
İhtiyacı**

Planlama

- Yazılım geliştirme sürecinin ilk aşaması
- Başarılı bir proje geliştirebilmek için projenin tüm resminin çıkarılması işlemi
- Proje planlama aşamasında yapılan işlemler
 - Proje Kaynaklarının Belirlenmesi
 - Proje Maliyetlerinin Kestirilmesi
 - Proje Ekip Yapısının Oluşturulması
 - Ayrıntılı Proje Planının Yapılması
 - Projenin İzlenmesi
- Proje planı tüm proje süresince sürekli olarak kullanılacak, güncellenecek ve gözden geçirilecek bir belgedir

Proje Kaynakları

- İnsan Kaynakları
- Donanım Kaynakları
- Yazılım Kaynakları

Planlama; bu kaynakların tanımını yapar ve

zaman kullanımı,

görev süreleri,

edinilme zamanlarını

planlar

İnsan Kaynakları

- Planlama; hangi tür elemanların, hangi süre ile ve projenin hangi aşamalarında yer alacağını belirler

Proje Yöneticisi	Donanım Ekip Lideri
Yazılım Ekip Lideri	Donanım Mühendisi
Web Tasarımcısı	Ağ Uzmanı
Sistem Tasarımcısı	Yazılım Destek Elemanı
Programcı	Donanım Destek Elemanı
Sistem Yöneticisi	Eğitmen
Veri Tabanı Yöneticisi	Denetleyici
Kalite Sağlama Yöneticisi	Çağrı Merkezi Elemanı

Donanım Kaynakları

- Günümüzde giderek açık sistem mimarisine dönüşmektedir.
- Donanım Kaynakları:
 - Ana Bilgisayarlar
 - Sunucular (Web, E-posta, Veri Tabanı)
 - Kullanıcı Bilgisayarları (PC)
 - Yerel Alan Ağı (LAN) Alt Yapısı
 - Geniş Alan Ağı (WAN) Alt Yapısı
- Yazılımın geliştirileceği ortam, gerçek kullanım ortamı dışında olmalıdır.
- Öte yandan, geliştirme ve uygulama ortamlarının aynı konfigürasyonda olmaları, ileride kurulum sırasında ortaya çıkabilecek taşıma sorunlarını büyük ölçüde giderecektir.

Yazılım Kaynakları

- Büyük ölçekte otomatik hale getirilmiş ve bilgisayar destekli olarak kullanılmaktadır.
- Bilgisayar Destekli Tasarım (CAD) ve Bilgisayar Destekli Mühendislik (CASE) araçları olarak bilinmektedirler.

Yazılım Kaynakları

- İş sistemleri planlama araçları
 - İş akış yapısının üst modelinin üretilmesinde kullanılır.
 - Bilgi akışı, bilgi yapısı iş birimlerindeki tıkanıklıklar bu araçlar kanalıyla ortaya çıkarılır.
- Proje yönetim araçları
 - Yönetici tarafından, projede yapılan işlerin izlenmesi, kaynak ataması, proje iş yapısının üretilmesi, gözlenen değerlerin işlenmesini sağlayan araçlar.
- Analiz ve tasarım araçları
 - Kullanılan modelleme tekniklerini ayrı ayrı ya da bütünlük olarak uygulayan araçlar. Üretilen modelin kalitesinin ölçülmesi
- Programlama araçları
 - Derleyiciler, nesne-tabanlı programlama araçları, görsel programlama platformları.

Yazılım Kaynakları (devam)

- Test araçları
 - Yazılımı doğrulama ve geçерleme işlemlerinde kullanılır. Test verisi üreticiler, otomatik test yordamları, ...
- Prototipleme ve simülasyon araçları
 - Geliştirmenin erken aşamalarında kullanıcıya, sonuç ürünün çalışması ile ilgili fikir veren ve yönlendiren araçlar.
- Bakım araçları
 - Programın bakımını kolaylaştıran, bir kaynak koddan program şemalarının üretilmesini, veri yapısının ortaya çıkarılmasını sağlayan araçlar.
- Destek araçları
 - İşletim sistemleri, ağ yazılımları, e-posta ve ortam yönetim araçları.

Proje Maliyetleri

- Maliyet kestirimi; bir bilgi sistemi ya da yazılım için gerekebilecek iş gücü ve zaman maliyetlerinin üretimden önce belirlenebilmesi için yapılan işlemlerdir.
- Kullanılan Unsurlar
 - Geçmiş projelere ilişkin bilgiler
 - Proje ekibinin deneyimleri
 - İzlenen geliştirme modelibirden çok kez uygulanabilir

Proje Maliyetleri

Maliyet yönetimi sayesinde;

- Gecikmeler önlenir
- Bilgi sistemi geliştirme süreci kolaylaştırılır
- Daha etkin kaynak kullanımı sağlanır
- İş zaman planı etkin olarak gerçekleştirilir
- Ürün sağlıklı olarak fiyatlandırılır
- Ürün zamanında ve hedeflenen bütçe sınırları içerisinde bitirilir

Gözlemlenebilecek değerler

- Projenin toplam süresi
- Projenin toplam maliyeti
- Projede çalışan eleman sayısı, niteliği, çalışma süresi
- Toplam satır sayısı
- Bir satırın maliyeti (ortalama)
- Bir kişi/ay'da gerçekleştirilen satır sayısı
- Toplam işlev sayısı
- Bir işlevin maliyeti
- Bir kişi/ay'da gerçekleştirilen işlev sayısı
- Bir kişi/ay'da maliyeti

Proje Sınıfları

- Ayrık Projeler:
 - Boyutları küçük,
 - Deneyimli personel tarafından gerçekleştirilmiş
 - LAN üzerinde çalışan insan kaynakları yönetim sistemi gibi
- Yarı Gömülü:
Hem bilgi boyutu hem donanım sürme boyutu olan projeler
- Gömülü Projeler:
Donanım sürmeyi hedefleyen projeler (pilotsuz uçağı süren yazılım - donanım kısıtları yüksek)

İşlev Noktaları Yönetimi

İşlev Noktaları Yöntemi

İşlev noktaları, geliştirmenin erken aşamalarında (çözümleme aşamasında) saptanan bir değerdir. Bu değer, sistemin oluşturulduğu uygulama geliştirme ortamından bağımsız olarak elde edilmektedir. İşlev noktalarının hesaplamasında problem tanımı girdi olarak alınarak üç temel adım izlenir:

1. Problemin bilgi ortamının incelenmesi
2. Problemin teknik karmaşıklığının incelenmesi
3. İşlev noktası hesaplama

Özellikle, çözümleme çalışması, yapısal yöntemlerle yapıldığında, işlev noktaları yarı otomatik bir biçimde kolayca elde edilebilir. İşlev noktaları yönteminin proje geliştirme sürecinin erken aşamalarında oldukça etkili olduğu belirtilmektedir.

Ölçüm Parametresi	Sayı	Ağırlık Faktörü		
		Yalın	Ortalama	Karmaşık
Kullanıcı Girdi Sayısı		3	4	6
Kullanıcı Çıktı Sayısı		4	5	7
Kullanıcı Sorgu Sayısı		3	4	6
Kütük Sayısı		7	10	15
Dışsal Arayüz Sayısı		5	7	10
Toplam Sayı				

Kullanıcı Girdileri

Yazılıma girdi olarak verilen her farklı uygulama bileşeni bir kullanıcı girdisi olarak sayılır. Kullanıcı girdilerinin sayısı hesaplanırken, alan bazında değil, daha genel olarak mantıksal kayıt bazında uygulama yapılmalıdır. Örneğin, bir personel sisteminin geliştirilmesinden söz edildiğinde, personel sicil bilgileri, personel izin bilgileri gibi bilgiler ayrı kullanıcı girdileri olarak sayılabilir. Personel sicil no, personel adı soyadı türündeki alan bilgileri kullanıcı girdisi olarak sayılmamalıdır.

Ölçüm Parametresi	Sayı	Ağırlık Faktörü		
		Yalın	Ortalama	Karmaşık
Kullanıcı Girdi Sayısı		3	4	6
Kullanıcı Çıktı Sayısı		4	5	7
Kullanıcı Sorgu Sayısı		3	4	6
Kütük Sayısı		7	10	15
Dışsal Arayüz Sayısı		5	7	10
Toplam Sayı				

Kullanıcı Çıktıları

Kullanıcıyı ilgilendiren her tür mantıksal çıktı, kullanıcı çıktısı olarak sayılmalıdır. Bu kapsamda kullanıcı çıktılarına örnek olarak, raporlar, ekran çıktıları, hata iletileri vb. verilebilir. Girdiler gibi çıktılar da genel olarak sayılmalıdır. Bir raporun içerisindeki bir alan kullanıcı çıktısı olarak sayılmamalıdır. Öte yandan, aynı bilgileri içeren ancak bilgi düzeni ya da sıralaması farklı olan raporlar da ayrı kullanıcı çıktısı olarak ele alınmamalıdır. Örneğin, soyadına göre sıralanmış personel listesi ve sicil no'suna göre sıralanmış personel listesi raporları farklı kullanıcı çıktıları olarak sayılmamalıdır.

Ölçüm Parametresi	Sayı	Ağırlık Faktörü		
		Yalın	Ortalama	Karmaşık
Kullanıcı Girdi Sayısı		3	4	6
Kullanıcı Çıktı Sayısı		4	5	7
Kullanıcı Sorgu Sayısı		3	4	6
Kütük Sayısı		7	10	15
Dışsal Arayüz Sayısı		5	7	10
Toplam Sayı				

Kullanıcı Sorguları

Kullanıcı sorgusu, çevrim içi olarak bilgisayara verilen bir girdi sonucu yine çevrim içi olarak bir kullanıcı çıktısı alınması biçiminde tanımlanmaktadır. Her farklı sorgu ayrı olarak sayılmalıdır. Örneğin; personel sicil bilgilerinin sorgulanması, personel maaş bilgilerinin sorgulanması vb.

Ölçüm Parametresi	Sayı	Ağırlık Faktörü		
		Yalın	Ortalama	Karmaşık
Kullanıcı Girdi Sayısı		3	4	6
Kullanıcı Çıktı Sayısı		4	5	7
Kullanıcı Sorgu Sayısı		3	4	6
Kütük Sayısı		7	10	15
Dışsal Arayüz Sayısı		5	7	10
Toplam Sayı				

Kütükler

Her mantıksal bilgi yığını ya da kütük ayrı olarak sayılmalıdır. Personel sicil kütüğü, personel maaş kütüğü, personel hareket kütüğü vb.

Ölçüm Parametresi	Sayı	Ağırlık Faktörü		
		Yalın	Ortalama	Karmaşık
Kullanıcı Girdi Sayısı		3	4	6
Kullanıcı Çıktı Sayısı		4	5	7
Kullanıcı Sorgu Sayısı		3	4	6
Kütük Sayısı		7	10	15
Dışsal Arayüz Sayısı		5	7	10
Toplam Sayı				

Dışsal Arayüzler

Geliştirilmesi öngörülen bilgi sisteminin, gerek kurum içinde gerekse kurum dışında bir başka bilgi sistemi ile bilgisayar ortamında bir iletişimi (çevrim dışı disket vb. ya da çevrim-içi doğrudan bağlantı) söz konusu ise bu durum bir dışsal arayüz olarak sayılmalıdır. Kağıt ya da rapor türündeki arayüzler bir dışsal arayüz sayılmazlar. Örneğin, personel bilgi sistemi, yine aynı kurumda olduğu varsayılan muhasebe bilgi sistemine, personel maaşlarına ilişkin bilgiyi bilgisayar ortamında aktarması bir dışsal arayüz olarak sayılabilir.

- Bunların ağırlık faktörleriyle çarpımları toplanarak, **Ayarlanmamış İşlev Nokta (AİN) sayısı** hesaplanır.

Problem Teknik Karmaşıklığının İncelenmesi - TKF

- Bu adımda, probleme ilişkin dışsal etkenlerin değerlendirilmesi yapılır. Problem **Teknik Karmaşıklık Faktörü (TKF)**, tablo 3.3'de belirtilen ve problemin karmaşıklığına ilişkin sorulara verilecek yanıtların toplamıyla elde edilir. Uygulama ortamının ve uygulamanın özellikleri dikkate alınarak her soruya 0 ile 5 arasında bir yanıt verilir. Örneğin, 8. soruyu dikkate alalım. Eğer uygulama, ana veritabanının, tümüyle çevrim içi olarak güncellenmesini gerektiriyorsa bu durumda yanıtımız 5 olacaktır. Ancak, bazı veri tabanı işlemleri (örneğin %10 gibi) çalışma saatleri dışında, çevrim dışı olarak yapılıyorsa bu durumda 4 gibi bir yanıt verilebilir.

Yanıt Kılavuzu:**0: Hiçbir Etkisi Yok****1: Çok Az etkisi var****2: Etkisi Var****3: Ortalama Etkisi Var****4: Önemli Etkisi Var****5: Mutlaka Olmalı, Kaçınılamaz**

1. Uygulama, güvenilir yedekleme ve kurtarma gerektiriyor mu?
2. Veri iletişimi gerekiyor mu?
3. Dağıtık işlem işlevleri var mı?
4. Performans kritik mi?
5. Sistem mevcut ve ağır yükü olan bir işletim ortamında mı çalışacak?
6. Sistem, çevrim içi veri girişi gerektiriyor mu?
7. Çevrim içi veri girişi, bir ara işlem için birden çok ekran gerektiriyor mu?
8. Ana kütükler çevrim-içi olarak mı günleniyor?
9. Girdiler, çıktılar, kütükler ya da sorgular karmaşık mı?
10. İçsel işlemler karmaşık mı?
11. Tasarlanacak kod, yeniden kullanılabilir mi olacak?
12. Dönüştürme ve kurulum, tasarımda dikkate alınacak mı?
13. Sistem birden çok yerde yerleşik farklı kurumlar için mi geliştiriliyor?
14. Tasarlanan uygulama, kolay kullanılabilir ve kullanıcı tarafından kolayca değiştirilebilir mi olacak?

- Cevaplar 0 ile 5 arasında puanlandırılır
- Bunlar hesaplanıp toplanarak Teknik Karmaşıklık Faktörü (TKF) elde edilir.

İşlev Nokta Sayısı Hesaplama

Bu son adımda, bilinen ve sık kullanılan deneysel formül kanalıyla ilk iki adım sonucundaki bulgular kullanılarak işlev nokta sayısı belirlenir.

Gerek AİN, gerekse TKF önceki bölümlerde açıklandığı gibi hesaplandıktan sonra, geliştirilecek bilgi sistemine ilişkin İşlev Nokta Sayısı (İN) aşağıdaki formül kullanılarak hesaplanır.

İşlev Nokta Sayısı çeşitli amaçlarla kullanılabilir: İşlev Noktaları, başka yöntemlerle birlikte kullanılmak istenildiğinde, kullanılacak yazılım geliştirme platformuna göre aşağıdaki değerler kullanılarak kolayca satır-sayısı kestirimine dönüştürülebilir.

$$\text{İN} = \text{AİN} \times (0.65 \times 0.01 \times \text{TKF})$$

$$\text{Üretkenlik} = \text{İN} / \text{Kişi-Ay}$$

$$\text{Kalite} = \text{Hatalar} / \text{İN}$$

$$\text{Maliyet} = \$ / \text{FP}$$

İşlev Noktaları, başka yöntemlerle birlikte kullanılmak istenildiğinde, kullanılacak yazılım geliştirme platformuna göre aşağıdaki değerler kullanılarak kolayca satır-sayısı kestirimine dönüştürülebilir.

Örneğin İN kestirimi olarak 300 İN bulduğumuzu varsayalım. Eğer nesne-kökenli bir dil kullanarak (smalltalk gibi) yazılımı geliştirecek isek yaklaşık satır sayısı kestirimi **300x30=9000** satır olacaktır.

Programlama Platformu	Satır Sayısı/İN (Ortalama)
Assembly Dili	300
COBOL	100
FORTRAN	100
Pascal	90
C	90
Ada	70
Nesne-Kökenli Diller	30
4. Kuşak Dileri	20
Kod Üreticiler	15

Etkin Maliyet Modeli - COCOMO



Tüm COCOMO modelleri, temel girdi olarak satır sayısı kestirimini alır ve çıktı olarak iş-gücü (Kişi-ay, Kişi-yıl vb.) ve zaman (ay, yıl, hafta vb.) çıktıları verir .

- İş Gücü (K) $K=a*S^b$
- Zaman (T) $T=c*K^d$

a,b,c,d : her bir model için farklı katsayılar

S : bin türünden satır sayısı

Proje Sınıfları

■ Ayrık Projeler:

- Boyutları küçük,
- Deneyimli personel tarafından gerçekleştirilmiş
- LAN üzerinde çalışan insan kaynakları yönetim sistemi gibi

■ Yarı Gömülü:

Hem bilgi boyutu hem donanım sürme boyutu olan projeler

■ Gömülü Projeler:

Donanım sürmeyi hedefleyen projeler (pilotsuz uçağı süren yazılım - donanım kısıtları yüksek)

Temel Model

- Küçük-orta boy projeler için hızlı kestirim yapmak amacıyla kullanılır
- Dezavantajı: Yazılım projesinin geliştirileceği ortam ve yazılımı geliştirecek ekibin özelliklerini dikkate almaz
- Avantajı: Hesap makinesi ile kolaylıkla uygulanabilir

Temel Model

- Ayrık Projeler
 - İş Gücü $K=2.4*S^{1.05}$
 - Zaman $T=2.5*K^{0.38}$

- Yarı Gömülü Projeler
 - İş Gücü $K=3,0*S^{1.12}$
 - Zaman $T=2.5*K^{0.35}$

- Gömülü Projeler
 - İş Gücü $K=3,6*S^{1.20}$
 - Zaman $T=2.5*K^{0.32}$

Arařtırma ödevleri

- Projeniz için gereksinim analizini yazınız
- Süreç modelini belirleyiniz
- Maliyet kestirimini öğrendiđiniz yöntemlere göre hesaplayınız
- Gömülü ve yarı gömülü sistemlere örnekler veriniz