



# BIMU4098 – BİLİŞİM PROJE GELİŞTİRME

Dr. Öğr. Ü. Emel ARSLAN

# Bu bölümde öğrenilecek kavramlar

- Planın genel özellikleri
- Planın Önemi
- Plan türleri
- Plan geliştirme süreci
- Planın güncellenmesi
- Ortak katılım ve onay
- İş bölümlene yapısı



# Planlama



*Planlama*, proje hedeflerine ulaşmak ve müşteri ihtiyaçlarını karşılamak için izlenecek yöntemi, yapılacak işleri, kullanılacak kaynakları ve iş takvimlerini proje kısıtlarını da dikkate alarak belirlemektir.

*Planda* kimin, neyi, ne zaman, nasıl ve ne maliyetle yapacağı yer alır [Sandoejan-2008 ]. Plan belli standartlara göre belgelenmelidir.

*Sadece kişilerin akıllarında olan plan, plan değildir. Richard H. Thayer*

Planlama ve yapılan planın izlenmesi, projenin başarılı şekilde tamamlanabilmesi için olmazsa olmaz şarttır. Plan olmaması durumunda süreyi, kaynak kullanımı ve iş hedeflerini takip etmek mümkün olmaz. Ekip ruhu kaybolur ve yazılımcılar kendi başlarına çalışmaya başlar. Proje yöneticisinin otorite sağlamak için kişisel yakınlık, zorlama veya korkutmadan başka bir aracı kalmaz.

# Planlama

*Planlama*, diğer tüm yazılım geliştirme aktiviteleri gibi bir nevi analizdir. Ancak planlama projeye üst seviye bir bakışla gerçekleştirilir ve görevin özellikleri belirlenir. Görevin ayrıntılı olarak nasıl yapılacağının belirlenmesi sorumlu ekip üyesine bırakılır. Öncelikle genel bir plan hazırlanır. Sonra bu plan her aşama başında veya sonunda yapılanlar, yapılacaklar, kaynakların kullanımları ve gerçekleşen sürelerin tekrar incelenmesiyle ayrıntılı hale getirilir. Böylece her aşama genel plana bağlı olarak, kendi planına sahip olur. Plan ayrıntılı hale getirilirken, proje geliştirme süreci en alt düzeyde tek bir kişi tarafından gerçekleştirilecek birim görevlere kadar bölünmelidir.

Planlamayla ilgili genel standartlar belirlenmeli ve daha sonra bunlar her projede ihtiyaca göre özelleştirilmelidir. Planlanması gereken birçok konu mevcuttur. Bunlar gruplanarak çeşitli standart plan türleri ortaya çıkmıştır. Geliştirilecek yazılımın özelliklerine göre farklı türlerde daha ayrıntılı planlar yapılması da gerekebilir.

# 1- Planın Temel Özellikleri

Sağlıklı bir planlama için öncelikle planlamanın önemine ve yararına inanmak gerekir. Planlama konusunda temel kavramlar da tam olarak anlaşılmalıdır.



# 1- Planın Temel Özellikleri

## 1- Planlamanın Önemi ve Yararları

Karmaşık bir geliştirme süreci olan ve farklı roller içeren ekiplerle yürütülen yazılım geliştirme sürecinin başarısı, kaliteli bir plana bağlıdır. Planlamanın hedeflediği alanlar ve buna bağlı yararları aşağıda sıralanmıştır[west-2002]:

- Ürünün ortaya çıkacağı zaman ve aşamaları özet olarak tarif eder.
- Proje hedeflerini net şekilde ortaya koyar.
- Ekip, yönetici ve diğer paydaşlar arasında bir iletişim köprüsü oluşturur.
- Görevleri, görevler arası ilişkileri ve kaynak ihtiyacını gösterir.
- Proje yöneticisinin yapılanları kontrol edebilmesini sağlar.
- Kaynaklara ne zaman ihtiyaç duyulacağını gösterir.
- Kontrol noktaları ve yapısını tanımlar.

# 1- Planın Temel Özellikleri

## 1- Planlamanın Önemi ve Yararları

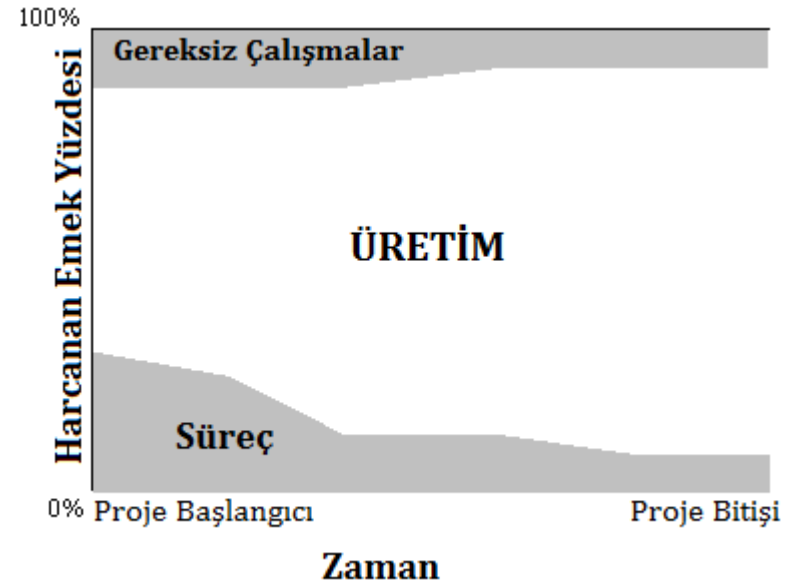
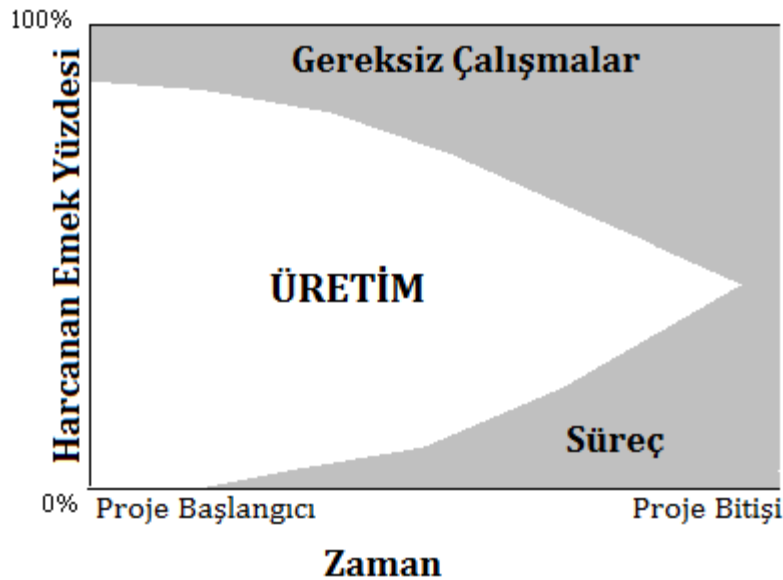
Proje başlangıcında planlama ve analiz için ayrılan süre, sonraki aşamalarda daha verimli çalışmaya imkan verir, değişiklik ve hataların azalmasını sağlayarak üretkenliği arttırır. Böylece **projenin zamanında bitme ihtimali artar**. Planlı yürütülen bir proje de çalışan kişi ne yapacağını bildiği için işlerini daha kolay anlar ve yapar.

Plan için başlangıçta süre ayrılmaması durumunda üretim hemen başlamış görüntüsü oluşur. Ancak zaman ilerledikçe geri dönüşlerle üretkenlik hızla düşer. **Projenin başlangıcında plan ve analize ayrılmayan zaman sonraki aşamalarda çok daha fazla kayba yol açacaktır**. İşleri yürütmek ve hatta işleri belirlemek zorlaşır. Projede kontrol kaybolur. Ayrıca sonradan yapılan düzeltmeler çok daha maliyetlidir.

# 1- Planın Temel Özellikleri

## 1- Planlamanın Önemi ve Yararları

**Eğer analiz ve planlama gereken kalitede yapılmazsa, her sefer başa döneceği için üretkenlik sürekli azalacaktır.** Eğer analiz hiç yapılmazsa veya yapılan plan için müşteri onayı alınmazsa geri dönecek bir nokta da kalmaz. Bu durumda projedeki anlaşmazlıkların çözümü için kullanılabilecek en önemli temellerden birisi zarar görür.



**Yazılım süreci ve üretkenlik [McConnell-1998]**



# 1- Planın Temel Özellikleri

## 1- Planlamanın Önemi ve Yararları

**Planlı geliştirilen bir projede projenin başlangıç aşamalarında son kullanıcı ve üst yönetim tarafından görünen üretim az olur.** Zamanın önemli bir kısmı planlama ve analiz için ayrılır. Sonraki aşamalarda bu çalışmalara ayrılan zaman azalır, görünen üretim artar. Planlı ve plansız çalışma yöntemleri dikkatle incelendiğinde, planlı yöntemde gereksiz çalışmanın toplamda daha az olduğu görülebilir. Süreç yönetimi için ayrılan çaba da benzer şekilde planlı çalışırken daha az olacaktır.

**Tüm önemlerine rağmen planlama ve analiz süresi aşırı uzun tutulmamalıdır.** Projenin başlangıcında müşteri ve üst yönetim katılımını sağlamak için prototiplerle bazı geliştirmeler gösterilmelidir.

# 1- Planın Temel Özellikleri

## 1- Planlamanın Önemi ve Yararları

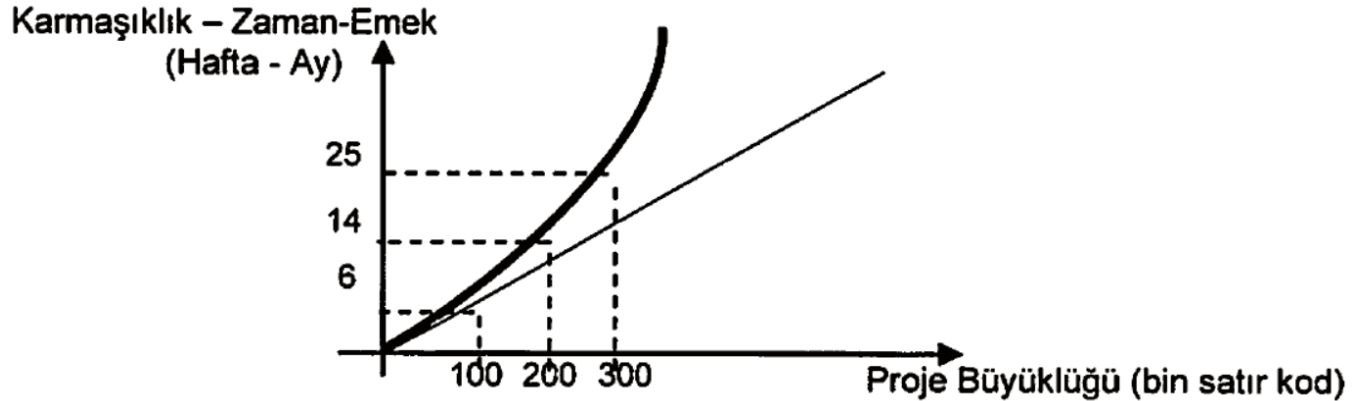
Planlama ve analiz süresini kısa tutma sebepleri:

- **İlk anda belirlenemeyecek konular:** Analiz uzun tutulsa da ilk anda bazı ihtiyaçlar belirsizdir. Bunlar ancak proje ilerlediğinde netleşir.
- **İhtiyaçların değişmesi:** Sürecin ilerlemesi ve ürünün görünmeye başlaması ihtiyacın daha net anlaşılması dolayısıyla değişmesine sebep olabilir.
- **İlgi azalması:** Uzun süre üretilen bir şey görülemeyeceğinden projeye dış ilgi ve destek azalacaktır. Hatta birçok projede üst yönetimin "*Niçin bir şey ortaya çıkmıyor?*" sorusu gerginliğe yol açabilmektedir.
- **Ekibin bir kısmının beklemesi:** Planlama ve analiz ekibi dışındaki üyelerin uzun süre beklemesi sonucu verim düşebilir.
- **Hedefin değişmesi:** Analizin çok uzaması halinde analiz edilen sürecin çalışma şekli değişebilir. Kurumların yaklaşık beş yılda bir önemli teknolojik değişikliklere gittiği düşünülürse bir yıllık bir analiz süreci özel bilimsel projeler hariç çok fazla anlam ifade etmez. Süre bir yılı aşarsa firma teknoloji, mevzuat veya firmanın çalışanları değişebilir.

# 1- Planın Temel Özellikleri

## 2- Proje Büyüklüğü ve Planlama İhtiyacı

Yazılımın büyüklüğü arttıkça, gerçekleştirmek için gerekli süre gibi faktörler üstel olarak artar. 100 bin satırlık bir proje 1 ayda bitiyorsa 200 bin satırlık bir proje 2 aydan daha uzun bir sürer. Yazılım bu yönüyle diğer ürünlerden ayrılır. Ekonomideki diğer ürünlerde genellikle büyüklük arttıkça birim maliyet düşmektedir. Bu durum negatif ölçek ekonomisi (diseconomy of scale) olarak isimlendirilir.



Proje büyüklüğü zaman ve karmaşıklık ilişkisi

# 1- Planın Temel Özellikleri

## 2- Proje Büyüklüğü ve Planlama İhtiyacı

Küçük ölçekli yazılım projelerinde ayrıntılı tasarım, kodlama ve birim test; büyük projelerde ise planlama, ihtiyaç analizi, tasarım ve bütünleşme daha çok zaman alır [*Boehm-1981*]. Bunun temel sebebi yazılımın birbiriyle bütünleşik çalışan birçok parçadan oluşmasıdır. Parça sayısındaki artış, proje karmaşıklığını da üstel olarak arttırır. Karmaşıklık planlama ihtiyacının artmasına sebep olur. Projenin büyümesiyle artan ekip sayısı da kişiler arasındaki aktif iletişim ihtiyacı ve karmaşayı arttırır.

# 1- Planın Temel Özellikleri

## 3- Planın Planı

Planlanması gereken konuların dolayısıyla plan türlerinin ortak özellikleri mevcuttur [IEEE-12207]. Bu ortak özelliklerinden yola çıkarak herhangi bir alanda kullanılacak planın temel yapısı belirlenir. Temel plan şablonuna, geliştirilecek projeye özel ekler yapılır ve projeye özel plan oluşturulur. Bu yaklaşım planlama yaparken önemli konuların unutulmasını önler.

# 1- Planın Temel Özellikleri

## 3- Planın Planı

Her planda ortak olan özellikler:

- Yapılacak işin tanımı, girdi ve çıktıları
- Başarı kriterleri
- Yapılacak işin önce ve sonrasındaki işler
- Yapılacak işin tahmini başlangıç ve bitiş tarihi
- İş yapmak için gerekli uzmanlık türleri, işi yapacak kişi ve roller
- Kişiler arasındaki iletişim yöntemi
- İşle ilgili standart yöntemler, tercih edilen yöntem ve tercih sebepleri
- İşin kontrol sıklığı ve yöntemi
- Kalite şartları
- Maliyet, risk vb.

# 1- Planın Temel Özellikleri

## 3- Planın Planı

Bu liste birçok farklı plana uygulanabilir. Örneğin belgelemeye uygulanırsa:

- Belge tür, içerik ve tamamlanma kriterlerin tanımlanması
- Belgeyi yazacak ve belgeyi kullanacak kişilerin tanımlanması
- Belgelemeyle ilgili tarih planlaması yapılması
- Belgeleme yöntem ve standartlarının tanımlanması
- Belge güncelleme ve kontrol sıklığının tanımlanması
- Belge kalite faktörlerinin tanımlanması
- Belgelemeyle ilgili maliyet ve risk gibi diğer faktörlerin tanımlanması

# 1- Planın Temel Özellikleri

## 4- Proje Plan Türleri

Bir bilgi işlem projesi kapsam, zamanlama, görevler arası ilişkiler, proje kısıtları ve risk gibi birçok farklı bakış açısından incelenebilir. Bunun sonucunda risk, iletişim, kapsam ve maliyet gibi farklı plan türleri ortaya çıkmıştır.

Proje büyüklük ve karmaşıklığına göre oluşturulacak plan türleri değişebilir. Farklı türdeki planlar ayrı ayrı hazırlanabileceği gibi genel bir plan içerisinde alt başlık olarak da yer alabilir. Ancak her plan türü için farklı bakış açısı kullanılmalı ve alt planlar arasında tutarlılık olmalıdır. Böylece yapılacaklar unutulmaz, kaynak ve zaman doğru planlanabilir.



# 1- Planın Temel Özellikleri

## 4- Proje Plan Türleri

- **Kapsam Planı:** Projede yapılacakların belirlendiği ve proje sınırlarının çizildiği plandır.
- **Proje bütünleşme planı:** Projedeki insan, zaman, risk ve kalite gibi farklı konulardaki faaliyetlerin tutarlı ve bütünleşik yürütülmesi için hazırlanan plandır.
- **Risk Planı:** Projedeki risk yönetimine yönelik risk tanım, cevap ve sorumluları gibi bilgiler içeren plandır.
- **Zaman Planı:** Yapılacak işlerin tarih sırası, işler arasındaki ilişkiler, hangi kaynağa hangi tarihler arası ve ne düzeyde ihtiyaç duyulacağı, süre tahminleri, kaynak takvimleri ve kontrol noktaları bu planla belirlenir.
- **Kaynak Planı:** Projedeki işleri yürütmek için gerekli kaynaklar ve kaynak takvimlerinin planıdır. Yazılımda temel kaynak ve hedef insandır.
- **Maliyet Planı:** İşlerin yürütülebilmesi için gerekli mali kaynaklar, bütçe, harcama takvimi ve kontrol yönteminin belirlenmesidir.

# 1- Planın Temel Özellikleri

## 4- Proje Plan Türleri

- **Kalite Güvence Planı:** Projenin kurumsal standartlara uygun ve beklentileri en iyi karşılayacak şekilde geliştirilmesini güvenceye almak için yapılacak faaliyet, kalite kriteri, yöntem, kontrol aralığı ve sorumluları içerir.
- **Test Planı:** Projenin kullanıcı beklenti ve teknik gereksinimleri karşılaması için yürütülmesi gerekli kontrol işlemlerinin zaman ve kaynak gibi açılardan planlanmasıdır.
- **Eğitim Planı:** Müşteri ve son kullanıcılara yazılım ürününü kullanımı hakkında verilecek eğitimi kapsar. Eğitime katılacak kişiler, eğitmenler, içerik, eğitim yeri, eğitim tarihi ve süresi gibi başlıklar içerir.
- **Devreye Alma Planı:** Projenin devreye alınma aşamasında yapılacak çalışmalar ve gerekli donanım-yazılım altyapısının planlanmasıdır.
- **Teknik Altyapı Planı:** Bu plan devreye alma planıyla benzer içeriğe sahiptir. Ancak devreye alma planı projenin ilk kurulumunu hedeflerken teknik altyapı bakım gibi sürekli işlemleri hedefler.

# 1- Planın Temel Özellikleri

## 5- Proje Planının Diğer Belgelerle İlişkisi

Yazılım projelerinde proje planının yanında analiz, tasarım ve çalışma tablosu gibi birçok belge bulunur. Plan bu belgelerle karşılıklı etkileşimlidir. Etkileşimin yönü genellikle proje planına doğrudur. Analiz ve tasarım, planı etkiler ve belirler. Ancak proje planı da diğer belgeleri etkileyebilir. Örneğin zaman kısıtları ürünün çeşitli özelliklerinden vazgeçilmesine sebep olabilir.

Bilgi tekrarıdan kaçınmak için diğer belgeleri plan içerisine doğrudan eklemek yerine referans olarak göstermek tercih edilir. Belgelerin kısa bir özeti de plana eklenebilir. Ancak bilginin tekrar etmesi güncelleme için önemli bir çaba gösterilmesine sebep olur. Güncelleme hataları - ki bunlardan kaçınmak oldukça zordur - bilgi tutarsızlıklarına yol açabilir.

# 1- Planın Temel Özellikleri

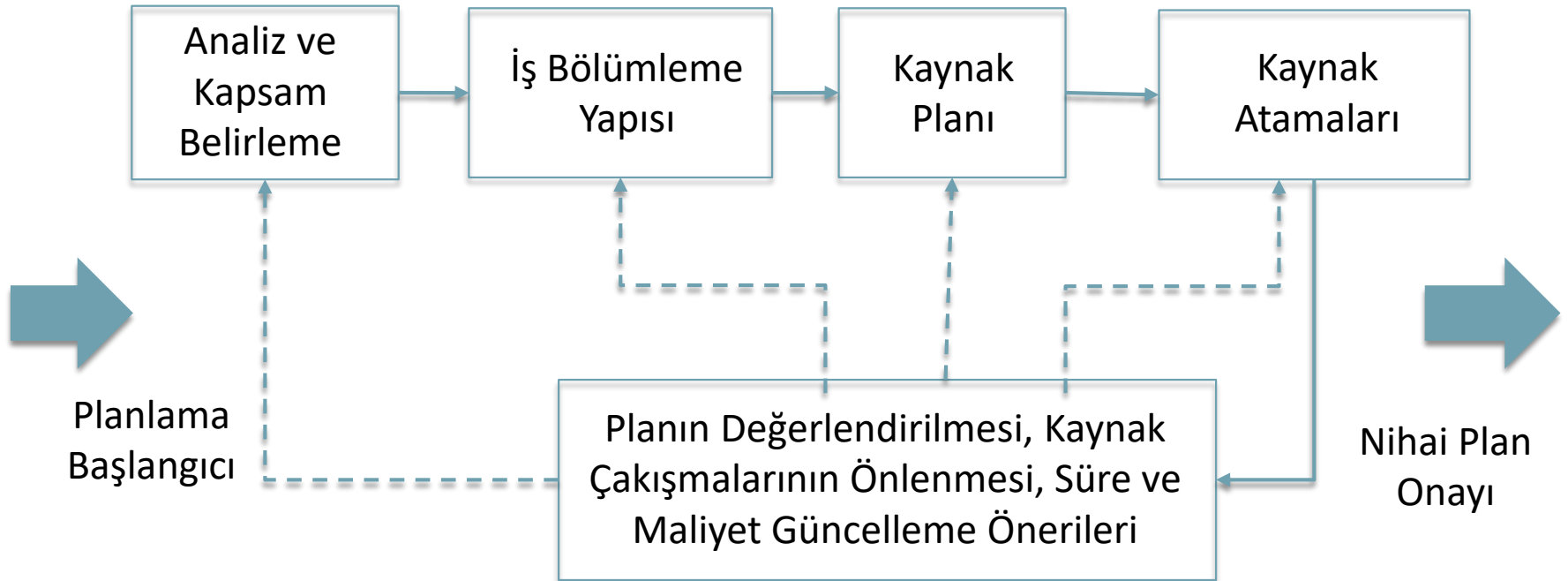
## 5- Proje Planının Diğer Belgelerle İlişkisi

Hangi istek veya özelliğin nasıl planlandığını göstermek için belgeler arasında eşleşme matrisi oluşturulabilir. Ancak uygulamada bunu güncel tutmak da oldukça güçtür. Eğer plandaki görev hiyerarşisi, görev isimleri ve görevler arası ilişkiler projedeki ihtiyaçları açıkça yansıtacak şekilde belirlenirse, belgeler arası eşleşmeler kendiliğinden açığa çıkacaktır. Planın kendi kendini açıklaması en iyi yöntemdir.

Tüm bilgileri tek bir belgede toplamak uygun değildir [Kliem-2008]. Belgelerin hedef ve kullanıcıları farklıdır. Eğer tüm belgeler tek bir proje planında toplanırsa paylaşım, güncellenme ve güvenlik sorunları oluşur. Örneğin proje maliyetlerinin planda doğrudan yer alması kişi ücretlerinin herkes tarafından görülmesine yol açabilir. Şirketlerde kişilerin birbirinin ücretini öğrenmesi istenmeyen bir durumdur.

# 2- Planlama Süreci

Planlama adım adım olgunlaşan bir süreçtir.



**Planın ilk oluşturma süreci**

## 2- Planlama Süreci

Yazılımı isteyen müşteriyle iletişim kurmak, ihtiyaç ve proje kısıtlarını anlamak bu sürecin ilk noktasıdır. Bunu kapsam belirleme çalışması takip eder. Sonrasında ihtiyaçlar ayrıntılı hale getirilir, yapılacaklar belirlenir ve genel bir iş bölümlene yapısı oluşturulur. Planın sağlıklı yapılabilmesi için ihtiyaçlar mümkün oldukça ayrıntılı belirlenmelidir.

Görevler tespit edilirken akış ve hiyerarşi şemalarıyla görevlerin birbiriyle olan bağlantıları da belirginleşir. Sonrasında yapılacak projenin büyüklüğü, harcanacak emek, süre ve gerekli kaynak tahmin edilerek plana yansıtılır. Varsa kaynak çakışmaları giderilerek plan iyileştirilir. Kaynağa görev atanarak işlemler başlatılır.

## 2- Planlama Süreci

Planlama sürecinde yapılanlar aşağıda adım adım sıralanmıştır:

- Görevler ve görevler arası ilişkileri belirlemek.
- Görevi yapmak için hangi uzmanlığa ihtiyaç duyulduğunu (rol) belirlemek.
- İhtiyaç duyulan diğer kaynakları belirlemek.
- İhtiyaç duyulan kaynakların ne kadarının temin edilebileceğini belirlemek.
- Projedeki görevlere kaynakları atamak. Kaynakların hangi yoğunlukta ve sürede görevle ilgileneceğini belirlemek.
- Kritik yolu bulmak.
- Görev çakışmaları ve aşırı yüklenen kaynakları, tespit etme ve çözümlenmek. Kritik yol bu aşamada tekrar düzenlenebilir.
- Geri besleme almak. Planlama aşaması analize de geri besleme sağlar. Çıktıları net tanımlanmamış bir görev için tekrar analiz gerekebilir.

## 2- Planlama Süreci

Planlama, müşteri katılımıyla yapılması gerekli bir çalışmadır. Sistemde her isteğin bir maliyeti olduğu müşteri ve üst yönetime anlatılmalıdır. Maliyeti düşürmek, kalite ve bazı özelliklerden feragat etmeyi gerektirebilir. Tüm paydaşlar, buradaki riskleri çok iyi anlamalıdır.

### **Planlama ve Tahmin**

Planlama ve tahmin iç içe geçmiş kavramlardır. Geliştirme sürecinde proje kapsamının belirlenmesini takiben proje büyüklüğü kabaca tespit edilir. Daha sonra sırayla harcanacak emek ve süre tahmini gerçekleştirilir. İşler ayrıntılı hale getirilerek birim görevlere parçalanır. Her görev için gereken süre ve kaynak tahmin edilir. Dikkat edilirse önce genel bilgiler sonra ayrıntılar üzerinden tahmin yapılmaktadır.

Bu çalışmada önce planlamadan bahsedilmiştir. Bunun iki sebebi vardır. Birincisi yapılacaklar analiz ve planlamayla ortaya çıkar. Dolayısıyla tahminlerin yapılacak işlere göre belirlenmesi daha sağlıklıdır. İkinci sebep de tahmin konusunda kullanılacak iş bölümlleme yapısı gibi bazı temel kavramların öncelikle planlamanın konusu olmasıdır. Eğer istenirse önce tahmin sonra planlama bölümleri de incelenebilir.



# 2- Planlama Süreci

## Aşamalı Planlama

*Plan*, ihtiyaçlar ve kapsama göre belirlenir. Ancak proje ilk başladığında ihtiyaçlar ve kapsam kesin belli değildir. Bunun sonucunda ilk anda genel bir planla yola çıkılması gerekir. Sonraki aşamalarda ihtiyaçlar ve kapsam netleştikçe planda ayrıntılı hale getirilir. Proje geliştirme aşamalarında bilgiler netleştikçe planlama süreci tekrar tekrar yürütülebilir.

Planı ayrıntılı hale getirme veya tekrar düzenleme işlemleri zaman periyodu veya süreç aşamasına bağlı yapılabilir. Periyodik yaklaşımda, plan ilk yapılmasından sonra 15 gün, bir ay gibi belli süre aralıklarında tekrar incelenir. Sürece bağlı yaklaşımda sürecin önemli noktalarında plan kontrol edilir ve gerekirse tekrar düzenlenir.

Hangi yaklaşım kullanılırsa kullanılsın yazılım sürecinin her temel aşaması tamamlandığında, bir sonraki aşamanın planı tekrar incelenmelidir. Analiz büyük ölçüde tamamlandığında tasarım için gerekli ayrıntıları plana eklenme vakti gelmiştir.

İlk anda tüm ayrıntıları plana eklemeye çalışmak, gereksiz bir vakit kaybıdır. Başlangıçta ulaşılamayacak bilgilerin yol açacağı belirsizlikler içerisinde, ayrıntıları değil planlamak, fark etmek dahi mümkün olmayabilir.

# 3- Plan Oluşturmadan Önce Yapılacak Faaliyetler

Planlama süreci öncesinde proje ismi, hedefi, kapsamı ve kullanılacak kaynaklarla ilgili temel kararlar verilmiş olmalıdır. Plan bu temeller üzerine inşa edilir. Gerekli altyapı doğru kurulursa gerçekleştirilebilir bir plana ulaşma ihtimali artar.

## **1. Hedefin Belirlenmesi**

Yazılım, temel bir ihtiyaca çözüm olarak geliştirilir. ilk olarak "*Kurumsal belgeleri düzenlemek için bir yazılım çözümü gerekli.*" veya "*Piyasada belge yönetimi alanındaki ihtiyaca cevap verecek bir teknolojik ürün geliştirmeliyiz*" şeklinde temel bir hedef belirlenmelidir. İhtiyaçlar, kapsam ve proje sınırları bu hedefe bağlı şekillenir. Elbette veritabanı, işletim sistemi gibi çok amaçlı altyapılar mevcuttur. Ancak bu ürünlerde dahi özellikler belli hedeflerle sınırlanmıştır. Bir işletim sisteminden veritabanı fonksiyonu beklenemez [Nizam-2011].

# 3- Plan Oluşturmadan Önce Yapılacak Faaliyetler

## *1. Hedefin Belirlenmesi*

Projede temel hedeflerden bazıları:

- Elle veya kağıtla takip edilen işlemlerin bilgisayar ortamına taşınması
- Mevcut projeye ilave modüller eklenmesi
- Var olan bir yazılımın kullanıcı ihtiyaçlarındaki veya altyapı teknolojilerindeki değişimlerden dolayı tekrar geliştirilmesi
- Yazılım sistemleri arasında entegrasyon
- Yazılım geliştirme sürecinin bir kısmının gerçekleştirilmesi; bazı projelerde hedef sadece analiz veya tasarım belgesi hazırlamak şeklinde olabilir.
- Mevcut yazılımı veya kodları iyileştirmek.



# 3- Plan Oluşturmadan Önce Yapılacak Faaliyetler

## *1. Hedefin Belirlenmesi*

Proje hedefi, kurumsal hedefleri destekleyici yönde olmalıdır. Diğer durumda orta ve uzun vadede projeye destek kaybolur ve projenin başarı şansı azalır. Hedef projeye katılan herkes için manalı olmalıdır. Ticari bir firmanın (hayır işleri hariç) sadece bilimsel neticeleri olan veya parasal kazancı az projelere yatırım yapmayacağı hep akılda olmalıdır.

Hedefle birlikte projenin ismi de belirlenmelidir. Proje ismi tüm paydaşların anlayabileceği ve proje hedeflerini açıklayan kısa bir ifade olmalıdır. Proje ismi çok önemlidir ve projenin yapısını anlamaya yardım eder. Eğer isim çok uzunsa projede birden fazla hedef gizlenmiş olabilir. İsim net değilse proje amacı belirsiz olabilir. Bu tür durumlarda proje hedef ve kapsamı tekrar gözden geçirilmelidir.

# 3- Plan Oluşturmadan Önce Yapılacak Faaliyetler

## ***2. Genel Kapsamın Belirlenmesi***

Planlamaya başlamadan önce kapsam genel olarak belirlenmelidir. İlk anda birçok ihtiyaç vardır. Ancak süre ve imkanlar sınırlıdır. Bu adımda şirketin ihtiyaç duyduğu yazılımın temel olarak hangi özelliklere sahip olacağı, hangi özelliklerin de projeye dahil edilmeyeceği sorularına cevap aranır. Bir proje ancak makul süre ve maliyetle geliştirilirse istenen faydayı sağlar. Bu yüzden faydalı da olsa bazı özelliklerin proje dışında bırakılması gerekecektir.

# 3- Plan Oluşturmadan Önce Yapılacak Faaliyetler

## 3. Proje Kabulü

Projeye başlangıcında hedef, genel süre, maliyet ve ekiple ilgili temel bilgiler büyük ölçüde netleşmiş olmalıdır. Hazırlanacak bir form üzerinde tarafların mutabakatı sağlandıktan sonra projeye başlanabilir [Zambruski-2009]. Bu belge proje başlangıç-bildiri (project charter) belgesi olarak isimlendirilir ve sürüm kontrol sistemine projeye ilgili eklenecek ilk belgelerden birisidir[PMBok-2008].

	Tarih:
Proje kodu - İsmi	
Proje amacı ve kapsamı	
Teslimat ve Çıktılar	
Başarı Kriterleri	
Planlanan Başlangıç Tarihi	
Planlanan Bitiş Tarihi	
Proje Personeli	
Müşteri	
Sponsor	
Proje Ekibi	
Diğer Firmalar ile ilişkiler	
Ekler: Proje ön analizi / Riskler	
Onaylar	

Bu örnek belge, en az olması gereken özellikleri içermektedir. Proje aşamalı geliştirilecekse aşamaların hedef ve tarihleri de belgeye eklenmelidir. Proje geliştirme yöntemi, önemi ve bütçe gibi bilgiler de isteğe bağlı eklenebilir.

Proje bildiri (başlangıç) belgesi

# 3- Plan Oluşturmadan Önce Yapılacak Faaliyetler

## 4. Proje Yöneticisi Atama

Projede çalışacak kişilerin belirlenmesi ilk aşamada alınan en önemli kararlardan birisidir. Kurum üst yönetimi, öncelikle projeye en uygun proje yöneticisini bulmalıdır. Proje yöneticisi proje alanını iyi bilen, lider karaktere sahip ve kurumsal yapıyı tanıyan kişiler arasından seçilmelidir. Proje yöneticisi, ilerleyen aşamalarda ekibini projenin gerektirdiği şartlara göre belirleme yetkisine sahip olmalıdır.



# 3- Plan Oluşturmadan Önce Yapılacak Faaliyetler

## ***5. Ekip Organizasyonunun Belirlenmesi***

Proje ekibi plandan sonra netleşir. Ancak ilk aşamada kurumun imkanları nispetinde kimlerin projede çalışabileceği az çok belli olur. Özellikle sabit fiyat esasına göre yapılan kurumsal projelerde, ilk aşamalarda henüz ayrıntılı plan yapılmadan ekipte görevlendirilecek kişilerle ilgili bilginin alıcı kuruma verilmesi gerekir.

Ekibin nasıl bir organizasyon yapısına sahip olacağı proje, kurum ve müşteri özelliklerine göre belirlenir. Örneğin ekibin farklı birimlerdeki kişilerin katılımıyla oluşması gerekiyorsa matris veya proje odaklı örgütlenme şekilleri arasında tercih yapılmalıdır. *Ekip yapısının* katılımcı veya otoriter, derin hiyerarşilere sahip veya yatay olması arasındaki tercihler de bu aşamada yapılır.

Çalışan temin ederken tam veya yarı zamanlı istihdam alternatifleri değerlendirilir. Ekipteki kişilerin tam zamanlı olmasının yönetimi kolaylaştırdığı unutulmamalıdır.

Proje ekibi sadece yazılım geliştiricilerden oluşmaz. Projenin hedef kitlesi olan kullanıcılar ve bunların rolleri de belirlenmelidir. Yazılım müşteri ve son kullanıcıları içinden, proje alanında bilgi sahibi ve bu bilgiyi paylaşmakta istekli kişiler özellikle tespit edilmelidir. Bu kişilerin proje sürecine aktif katılımı sağlanmalıdır. Proje için bilgi alınabilecek doğru kişiyi tespit etmek, önemli bir proje yöneticisi becerisidir.



# 3- Plan Oluşturmadan Önce Yapılacak Faaliyetler

## ***6. Temel Altyapı Kararlarının Verilmesi***

Yazılım geliştirme teknoloji altyapısı birçok parçadan oluşur. Bunlar yazılım geliştirme aracı, veritabanı, uygulama sunucusu, yazılım konfigürasyon yönetim aracı, proje yönetiminde kullanılacak araçlar gibi sıralanabilir. Altyapıda hangi araçların kullanılacağı ve kullanırken izlenecek yöntemin belirlenmesi çok önemlidir. Çünkü bunlar doğrudan proje süre ve insan kaynak ihtiyacını etkiler. Yeni bir yazılım geliştirme aracı kullanımı düşünülüyorsa öğrenme süresi de projeye dahil edilmelidir.

Proje kısıtları ile altyapı arasındaki denge proje risklerini azaltacak şekilde kurulmalıdır. Eğer bir projede zaman kısıtları çok fazlaysa, yeniliklerin denenme ihtimali azalır. Böylesi durumlarda ekip tecrübeli ve konunun uzmanı kişilerden oluşturulmalıdır.

# 3- Plan Oluşturmadan Önce Yapılacak Faaliyetler

## ***7. Kaynaklar ve Kısıtların Uyumunun Kontrolü***

Projenin temel kaynakları ve kısıtları ilk anda genellikle belirlidir. Proje süresi 12 ay ve görevlendirilebilecek personel sayısı 10 gibi. Proje yöneticisi bu kaynaklarla projenin geliştirilip geliştirilemeyeceği incelemeli ve ilgili kişilere bilgi vermelidir.

Proje yöneticisinin en büyük mücadelesi, öncelikle gerçekçi olmayan beklenti ve planlara karşı durmaktır. Ekibini anlamsız baskılardan korumak ve baskıyı, yapanlara (ayrıntılı gerekçelerle bilgi vererek) geri yansıtmaktır[Page-1985]. Mali öncelikli düşünen bazı üst yöneticiler, işlerin yetersiz kaynakla ve çok kısa sürede bitmesini talep edebilmektedir. Bu yüzden proje yöneticisi ve üst yönetim arasında ciddi tartışmalar olur. Proje yöneticisi bu tartışmalardan kaçmak için aşırı beklentilerin altına imza atmamalıdır. Eğer imza atarsa, sadece sorunu birkaç gün daha ertelemiş olur. Böylesi durumlarda proje yöneticisi niçin proje süresinin daha kısaltılamayacağını veya maliyetin azaltılamayacağını teknik gerekçelerle basitçe izah etmelidir. Süreyi verimli kullanmak için ihtiyaçları sürümlere bölmeyi de talep edebilir.

# 4- Kapsam Planı

Kapsamın planlanması, projede yapılacak (ve yapılmayacak) temel işleri belirleyerek, proje sınırlarını çizmektir. İnsan ihtiyaçlarının bir sınırı yoktur ancak imkanlar sınırlıdır. Sınırları çizmek bir projenin en hassas aşamalarından birisidir. Yapılacakları belirlemek genellikle kolaydır. Asıl sorun neyin yapılmayacağını belirlenmesinde çıkar. Kapsamın belirlenmesinde ayrıntılı olarak yapılacaklardan ziyade üst düzey proje ve ürün yaklaşımı önem kazanır. Bu yüzden projede temel vizyon ifadesi aynı zamanda kapsamı da tarif etmeli ve sınırlara da işaret etmelidir [McConneii-1997].

Kapsam planı, sadece projenin ilk anında kapsamı belirlemekten ibaret değildir. Bu plan, proje sürecinde kapsamı değiştirebilecek isteklerin değerlendirilmesi ve kapsamdaki değişimlerin kontrolünü de kapsar.



# 4- Kapsam Planı

## Kapsamı sürümlere dağıtarak planlamak

Müşterinin tüm isteklerini en mühimlerinden başlayarak gruplanmak ve her grubun bir sürüm olarak gerçekleştirmek, projeyi planlama ve yürütmeyi kolaylaştırır. Günümüzdeki hemen tüm yazılım ürünleri birkaç sürüm öncesine göre oldukça gelişmiştir. Eğer bu ürünlerin son sürümlerindeki özellikler beklenseydi birkaç yıl önceki hiçbir özellikten faydalanma imkanı olmayabilirdi. Ayrıca yeni sürümler firmalar için önemli gelir kaynaklarıdır.

Mağaza önlerinde yazılımların yeni sürümünü almak için sabaha kadar bekleyen müşteriler oldukça yeni sürümler de olacaktır!



# 4- Kapsam Planı

## **Diğer firmalardaki benzer projelerin kapsamını incelemek**

Firmaların yazılım ihtiyaçları birbirine benzer. Yazılım ekibinin benzer bir kurumda aynı alanda çalışan yazılım projesini incelemesi ve sonuçlarını kullanıcılarla paylaşması, kapsamın doğru belirlenmesi için yol göstericidir. Aynı sektörde çalışan benzer büyüklükteki A ve B gibi iki kurum düşünölsün. A kurumu yeni bir personel sistemi yazılım projesine başlamış olsun. B kurumunda halihazırda çalışan personel sisteminin 100 adet fonksiyonel özelliğı bulunsun. Bu durumda A kurumunun ihtiyacını 20 adet özellik içeren bir ürünle çözmek zordur. Belki A kurumundaki kullanıcılar ilk anda 20 adet özelliğın kendilerine yeteceğini düşünebilir. Ancak proje ilerledikçe A kurumunda geliştirilen ürününün özellikleri adım adım B kurumunda mevcut olan yazılımın özelliklerine yaklaşacaktır.

## 4- Kapsam Planı

### **Firma büyüklüğü, kullanıcı sayısı ve hitap edilen pazar**

Yazılımı kullanan sayısının artması proje büyüklüğü, kapsam ve planı etkiler. Yüzlerce kullanıcının kullandığı basit bir ekranı geliştirmek bile güvenlik, performans ve gerekli donanım altyapısı için ayrıntılı planlama gerektirir.





# 5- Proje Bütünleşme (Entegrasyon) Planı

Geliştirilen projelerde bilgilerin, farklı planların, kaynakların, çelişen hedef ve alternatiflerin bütünleşik olarak yönetilmesi gereklidir [PMBok-2008]. Bütünleşme planında; kullanılacak yöntem, ara yüzler, sistemler arası veri alışverişi ve konunun sorumluları bulunmalıdır. Bütünleşme işlemine plan yaparken, yazılım sistemleri veya bir yazılımın alt modülleri arasında entegrasyon sağlarken ihtiyaç duyulabilir:

- **Plan türlerini bütünleşik yürütme:** Kapsam, maliyet ve risk gibi farklı planların projede birlikte uygulanmasında nasıl hareket edileceği bir tür bütünleşme planıdır [ PMBok-2008 ].
- **Sistemler arası bütünleşik çalışma:** Günümüzde kurumlar temel yazılım ve donanım altyapılarını oluşturmuş durumdadır. Yeni yazılacak bir projenin var olan projelerle birlikte çalışması gerekir.
- **Modüller arası bütünleşme:** Eğer çok büyük bir sistem tasarlanıyorsa, bu sistemin alt sistemlere bölünerek tasarlanması gerekir. Bunlar arasındaki bütünleşme ve iletişim de planlanmalıdır.

# 6-Görevlerin Belirlenmesi:İş Bölümleme Yapısı

Kapsam belirlendikten sonra yapılacak görevlerin ayrıntılı belirlenmesi aşamasına geçilir. Yazılım geliştirme karmaşık bir işlemdir. Orta ölçekli bir yazılımda dahi bir çok aşamaya dağılmış yüzlerce ilişkili görev vardır. Bu görevlerin tutarlılık bozulmadan, müşteri talep ve teknik gereksinimleri karşılayacak şekilde gerçekleştirilmeleri gerekir. Yapılan araştırmalar yazılımın karmaşıklığını tekrar tekrar ortaya koymuştur.

Hiç kimse modern bir bilgisayar programını tamamıyla bir defada anlayabilecek bir zekaya sahip değildir. [Edsger Dijkstra 1972]

Yapılması gereken zekamızı bir defada tüm karmaşayı yönetecek şekilde geliştirmeye çalışmak değildir. Zira bu zaten mümkün değil. Kişilerin bir defada anlaması gereken karmaşanın miktarını en aza indirmektir. [McConnell-2004.1]



## 6-Görevlerin Belirlenmesi:İş Bölümleme Yapısı

Karmaşıklığı azaltmanın en önemli yöntemi, sistemi kendi içerisinde anlamlı parçalara bölmektir. Bir yazılımı oluşturan yapılar özetten ayrıntıya doğru modül, paket, nesne, metot ve metot içerisindeki kodlar şeklinde ifade edilebilir. Bu yapıya yazılım mimarisi veya yazılımın mimari bölümlenmesi ismi verilir. Yazılımın bölümlere ayrılması diğer yararlarının yanında görevlerin belirlenme ve atanmasını da kolaylaştırır. Örneğin bir nesneyle ilgili yapılacak işler, bir görevin altında toplanabilir.

Yazılım proje planı, yazılım hiyerarşik yapısına paralel olarak geliştirilmelidir. Yapılacaklar seviyelendirilerek birim işlere veya alt görevlere bölünür. Böylece yapılacakların tanımlanması ve birbiriyle ilişkilerinin belirlenmesi gerçekleştirilir. Bu bölümlemeye *İş Bölümleme Yapısı* veya *İş Bölümleme Ağacı* (*Work Breakdown Structure*) ismi verilir. İş bölümleme yapısı planın temelidir.

## 6-Görevlerin Belirlenmesi:İş Bölümleme Yapısı

*İş bölümleme* yapısının amacı, yazılım projesindeki işlemleri yönetilebilir küçük parçalara bölmektir. Böylece her parçayı gerçekleştirecek kişi, gerçekleştirme zamanı ve tamamlanma kriteri tanımlanabilir. Görevler gruplanarak proje geliştirme sürecinin önemli noktaları için kilometre taşı veya kontrol noktası oluşturulur. Görevler arası ilişkiler Gantt ve Ağ Şeması gibi farklı gösterim şekilleriyle analiz edilebilir.

Yazılım mimarisi ve iş bölümlemesi birbirine paralel ancak farklı kavramlardır [Fairley-2009]. Yazılım mimarisi genellikle nesneleri açıklar ve isim cümleleriyle ifade edilir. İş bölümleme yapısı ise yapılacakları fiil cümleleriyle ifade eder. Bir görevde tek bir nesne olması tercih edilir. Ancak bir nesneyle ilgili birden fazla görev tanımlanabileceği gibi bir görev tanımı da birden fazla nesneyi etkileyebilir. Aradaki eşleşme planda yer almalıdır. Yazılım bileşenleri arasında hiyerarşi haricinde de ilişkiler olabileceği unutulmamalıdır. Bazı bileşenlerin tasarım veya kodlanması diğer nesneler bağlı olabilir. Örneğin kod kütüphanesindeki ortak bir fonksiyonu birçok bileşen kullanacaksa, bu fonksiyon diğerlerinden önce tasarlanmalıdır.

# 6-Görevlerin Belirlenmesi:İş Bölümleme Yapısı

## ***1. Görev Tanımı***

Görev proje planındaki insan veya diğer kaynağın yapacağı, manalı, bütünleşik ve birim iştir. Görev genellikle iş bölümleme yapısının en alt seviyesinde gösterilir.

Görevlerin gruplandığı üst işler de *özet görev* olarak adlandırılır. Her görevin bir kişi tarafından yürütülmesi tercih edilir. Görevin özellikleri:

- Görev tanımı
- Girdiler ve çıktılar, üretilecek ürün
- Başlangıç ve bitiş tarihi, tahmini süre
- Başarı/Tamamlanma kriterleri
- Gerekli kaynaklar: Kişi, araç gibi
- Riskler (istenirse tanımlanabilir)

# 6-Görevlerin Belirlenmesi:İş Bölümleme Yapısı

## ***2. Görevlerin Tespiti İçin Yöntemler***

Proje planının yapısı; özetten ayrıntıya, ayrıntıdan özete, akış mantığına veya birden fazla yöntem birlikte düşünülerek inşa edilebilir. Özetten ayrıntıya incelemede öncelikle en temel görev tespit edilir. Temel görev alt görevlere bölünür. Bölümleme sistemin en alt seviyesine ulaşıncaya kadar devam eder. Bu plan ayrıntıdan özete incelemesi yapılacak bir işin eksik kalıp kalmadığını tespit etmeyi sağlar. Bir projede farklı yöntemlerden faydalanılabilir ve hatta faydalanılması da gereklidir. En önemli şey nereden bakıldığının unutulmamasıdır [Nizam 2011 ].

*Akış-Süreç temelli analiz*, işlerin birbirine bağlantısını incelemek ve sırasını belirlemek için kullanılabilecek bir yöntemdir. İşlerin birbiriyle olan bağlantısı hiyerarşik yapıdan daha karmaşık olabilir. Bu yüzden son yıllarda proje plan oluşturma araçlarında hiyerarşik Gantt Şemasının yerini *Ağ Şeması* almaktadır.

Model tekniğini belirlerken proje yapısı önemlidir. Temel bir amaç etrafında geliştirilen bir proje, özetten ayrıntıya daha kolay planlanır. Temel amaç belirsizse ayrıntıdan özete bir modelleme tekniği kullanmak daha uygundur. [McConneii-2004.1 ]

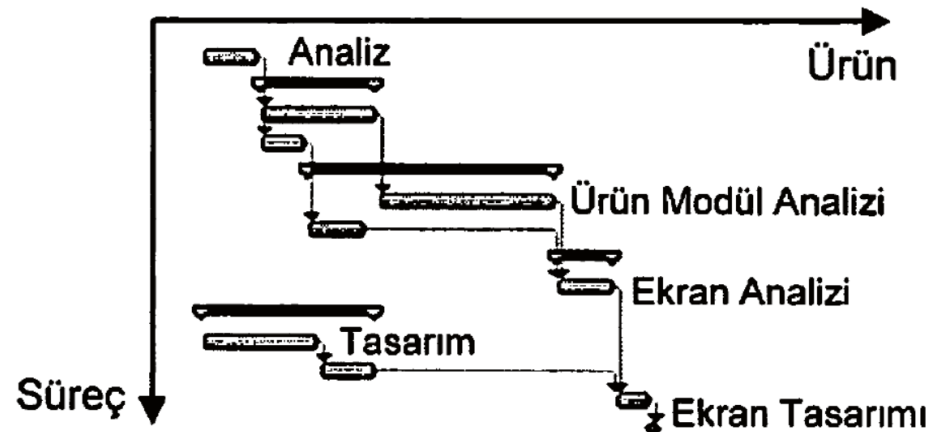
# 6-Görevlerin Belirlenmesi:İş Bölümleme Yapısı

## 2. Görevlerin Tespiti İçin Yöntemler

### İş bölümleri, ürün ve yazılım süreci uyumu

İşlerin bölünmesi ürün ve süreç şeklinde iki eksende düşünülebilir. Yazılım mimarisi işlerin hiyerarşik ayrıntılarını, yazılım süreci de zamana bağlı sırasını belirlemeye yardımcı olur. Örneğin bir muhasebe sisteminde geliştirilecek her bir ekran için geliştirme aşamasının en alt seviyesinde bir görev tanımlanabilir. Bu ekranların analizi planın analiz aşamasında, tasarımı planın tasarım aşamasında yer alır. Farklı tür yazılım süreçlerinde farklı eksen yapıları kullanılabilir.

Görevler, tercih edilen yazılım süreç yönetim yöntemine göre proje planına yerleştirilir. Örneğin şelale modeli için sıra analiz, tasarım, kodlama şeklindedir. Artımlı geliştirmede ise her çevrim plana dahil edilmelidir.



Görev Yapısında ürün ve süreç eksen

# 6-Görevlerin Belirlenmesi:İş Bölümleme Yapısı

## *2. Görevlerin Tespiti İçin Yöntemler*

### **Analizde istenenlerin ana faaliyet kabul edilmesi**

Genellikle analizdeki bir ihtiyaç için tasarım birden fazla bileşen içerir. Bu yüzden hiyerarşik yapıda analizdeki ihtiyaçların üst kademelere yazılması önerilir [Fairley-2009]. Böylece yapılan işlerle istekler arasında net bir bağlantı kurularak isteklerin ne kadarının karşılandığı görülebilir. Müşterinin de planı izlemesi kolaylaşır.

# 6-Görevlerin Belirlenmesi:İş Bölümleme Yapısı

## 3. Görevler Arası İlişki Türleri

Proje planı yapılan işler arasında gerçek hayatta var olan ilişkileri yansıtmalıdır. Proje süresi ve yapılacakların sırasının belirlenmesinde bu ilişkilerin çok büyük önemi vardır. İlişkiler en çok kullanılanıdan başlayarak aşağıdaki tabloda sıralanmıştır.

İlişki Türü	Açıklama	Örnek
Bitiş-Başlangıç	Bir görevin bitiş tarihi diğerinin başlangıç tarihini belirler	Tasarım bitmeden kodlama başlayamaz veya tasarım bitince kodlama başlar
Başlangıç-Başlangıç	İki görevin aynı anda başlaması	Yazılım geliştirme ve donanım temini birlikte başlayabilir.
Bitiş-Bitiş	Bir görevin bitişi diğerinin de tamamlanmasına sebep olabilir.	Test tamamlandığında donanımlarda temin edilmiş olmalıdır ki proje devreye alınabilsin
Başlangıç-Bitiş	Bir görevin başlangıç tarihinin diğerinin bitiş tarihini belirlemesi	Tasarım konusunda dışarıdan alınacak ve tarihi belli bir uzman desteği başlamadan önce ihtiyaç analizinin bitirilmesi

Gerçek hayatta her zaman, bir görevin diğerinin tam bittiği tarihte başlaması gerekmez. Burada gecikme süresi (*lag time*) kavramı devreye girer.

# 6-Görevlerin Belirlenmesi:İş Bölümleme Yapısı

## 4. Gantt Şeması

Gantt Şeması, Henry Gantt tarafından önerilen, proje planını görsel olarak yansıtan ve planlama araçlarında çok kullanılan bir çizelgedir. Görev takvimi, görevler arası bağlantılar ve kilometre taşları bu çizelgede özel sembollerle ifade edilir

Görev	
Görevin tamamlanan kısmı	
Kilometre taşı	
Görev bağlayıcı	
Gruplayıcı	

**Gantt Şeması şekilleri**



# 6-Görevlerin Belirlenmesi:İş Bölümleme Yapısı

## 4. Gantt Şeması

### Örnek proje planı

No	Görev İsmi	Süre	Önceki görevler	Gantt Şeması
1	1.Analiz	5 hafta		
2	1.1 İhtiyaç analizi yapılması	5 hafta		
3	2.Tasarım	10 hafta		
4	2.1. İş kuralları tasarımı	10 hafta	2	
5	2.2. Güvenlik Modül tasarımı	4 hafta	2	
6	3.Kodlama	15 hafta		
7	3.1. İş kurallarının kodlanması	15 hafta	4	
8	3.2. Güvenlik Modülü kodlanması	5 hafta	5	
9	4.Test	5 hafta		
10	4.1. Test yapılması	5 hafta	7; 8	
11	5.Alt yapı Çalışmaları	15 hafta		
12	5.1. Donanım temini	10 hafta		
13	5.2. Kurulumların yapılması	5 hafta	12	
14	6.Yazılımın Kurulması	3 hafta	10; 13	
15	7.Devreye alma	Proje Sonu	14	

## 7-Kaynakların Planlanması

Kaynak planlama aşamasında öncelikle projede hangi uzmanlıklara hangi tarih aralığında ihtiyaç duyulduğu belirlenir. Sonrasında kurumun bu uzmanlıkları nasıl sağlayacağı, kendi imkanlarıyla mı yoksa dış kaynaklardan mı temin edeceği ayrıntılı şekilde belirlenerek plana eklenir.

Kaynak planı hazırlanması, zaman planıyla birlikte yürütülmesi gereken bir çalışmadır. Görevin süresini, istihdam edilen kişinin üretkenliği ve projeye katılım oranı belirler. Bunu yaparken kaynak takvimleri ayrıntılı olarak incelenir. İzinler, hastalık ihtimali ve resmi tatiller de dikkate alınmalıdır.

# 7-Kaynakların Planlanması

Kaynak planlamasında yapılanlar:

- **Kaynakların belirlenmesi:** Hangi tür kaynaklara ihtiyaç duyulduğunun tespiti (kişiler için rol), kaynak temin önerilerinin değerlendirilmesi ve seçimi, kaynak takvimleri oluşturma
- **Kaynak maliyetlerinin belirlenmesi:** Normal çalışma, fazla mesai ve diğerleri için gider ve maliyet değişim tablosu tanımlama
- **Bütçeleme:** Ayrıntı maliyetleri toplayarak proje bütçesini oluşturma
- **Görevlere kaynak atama:** Kaynak görev eşleştirilmesi, kaynağın görevde ne kadar yer alacağını belirlenmesi, kaynak atamalarının kaynak ve faaliyete göre kontrol edilmesi
- **Çakışma düzenleme:** Kaynak çakışmalarını gözden geçirme ve düzeltme

Görev tanımlarının netleşmesi, projedeki kaynak ihtiyacının da tam olarak ortaya çıkmasını sağlar. Görevlerdeki çakışmalar ve aşırı yüklenen kaynaklar tespit edilir. Görevler, çakışmalar en aza indirilecek veya çözümlenecek şekilde tekrar zamanlanır. Görevlerin kaydırılmasında özellikle *bolluk zamanları* dikkate alınır.

# 7-Kaynakların Planlanması

## 1.Kaynak Türleri

Projedeki hedefleri gerçekleştirmek ve görevleri yerine getirmek için kullanılan herşey projenin kaynağıdır. Bunlar görev yapan kişi, işlerin yapılmasında kullanılan malzeme, makine ve diğer mali değerler olarak sayılabilir. Kaynaklar yapılarına göre farklı şekillerde sınıflanabilir. Her kaynak türü kendi içerisinde farklı özellikler gösterir. *Kaynak türü* kaynağın nasıl sağlanacağını, kullanım şeklini ve maliyetini belirlerken yardımcı olur. Kaynak türleri:

- **İnsan:** İşlerin yerine getirilmesi için gerekli uzmanlık ve kişileri ifade eder. İnsan her projenin kaynağı ve hedefidir. Genellikle diğer kaynaklardan ayrı değerlendirilir. Öncelikle ihtiyaç duyulan uzmanlık, sonrasında buna sahip kişilerin tespiti önerilmektedir.
- **Malzeme:** Projenin tamamlanması için gerekli malzemelerdir. Kırtasiye, bilgisayar, yazılım ve donanım şeklinde sayılabilir. Malzemelerin satın alma ve bakım maliyetleri dikkate alınmalıdır.
- **Mali:** Doğrudan mali değer olarak ifade edilen kaynaklardır. Yapılan seyahat ve otel masrafı gibi maliyetler örnek olarak verilebilir.
- **Diğer:** Projenin yapılması için gerekli altyapı, kurumsal standartlar, önceki projelerden elde edilen tecrübeler ve kod kütüphaneleri gibi kurumsal kaynaklardır.

# 7-Kaynakların Planlanması

## ***2.Kaynak Takvimleri Belirlenmesi***

Kaynağın projede aktif görev alabileceği süre, katılım yüzdesi ve takvimi belirlenmelidir. Görevlerin gerçek çalışma zamanlarına göre planlanması için kaynak takvimlerinin proje başında oluşturulması uygun olur. Kişi ve diğer kaynakların proje çalışmalarını etkileyecek tüm bilgileri sisteme girilmelidir. Örneğin izin ve seyahatler takvime işlenmelidir. Girilmesi unutulmuş her süre, projeye olumsuz yansır ve riski arttırır. Hastalık gibi beklenmeyen durumlar düşünülerek belli bir esneklikte zaman bırakılmalıdır.

# 7-Kaynakların Planlanması

## 3.Kaynakların Göreve Atanması

Kaynak atama, görev ile ilgili kişi ve diğer kaynakların eşleştirilmesidir. Kişi göreve atandığında kişinin uzmanlığı, proje takvimi ve kişinin takvimi karşılaştırılarak görevin alacağı toplam süre bulunur. Bu eşleştirmede kişinin göreve ne kadar zaman ayıracağı belirlenir. Görev için harcanacak emek ve süre aşağıdaki değişkenlerle modellenir[Quiring-20i2]:

- **Süre:** Görev için planlanan toplam süre. Gün cinsinden ölçülür.
- **Birim:** Bir kaynağın toplam zamanının ne kadarını görev için ayırabileceğidir. Yüzde cinsinden ölçülür.
- **Çalışma:** Görevi yapmak için fiilen ne kadar zaman gerektiğidir. Saat cinsinden ölçülür.

Bir kişi 2 gün süren bir görevi tamamlamak için günlük 8 saatlik mesainin 4 saatini harcamışsa; *süre* 2 gün, *birim* %50 ve *çalışma* 8 saat olarak bulunur. Görevin süresi, harcanacak emek ve kaynak ilişkisinde aşağıdaki denklem kullanılır.

$$\text{Süre} \times \text{Birim} = \text{Çalışma}$$

# 7-Kaynakların Planlanması

## 3.Kaynakların Göreve Atanması

Buradaki denklemde değişkenlerden birisi sabit kabul edilirse kalan iki değişken birbirini etkiler. Burada bir görev için temelde üç durum söz konusudur. Bu durumlar görev türleri olarak standartlaşmıştır:

- **Sabit Süre:** Görevin alacağı toplam (gün) süresi sabittir. Süre ilave kaynaklarla değişmez. Örneğin "*donanım temini* " görevi ne kadar kaynak ilave edilirse edilsin, sağlayıcı firmanın teslimat süresine bağlıdır.
- **Sabit Birim:** Çalışan kişinin göreve ayıracağı birim zaman sabittir. Bu birim zaman görev süresi veya çalışma değişiminden etkilenmez. İlave kaynak atanırsa süre kısalır. Bir programcı tek görevdeki birbirine bağlı iki ekranı 4 günde yazıyorsa ikinci bir programcı ilavesi süreyi iki güne düşürür. Dikkat edilirse programcı çalışma yüzdesi değil, sayısı arttırılmıştır.
- **Sabit Çalışma:** Toplam fiili çalışma süresi (saati) sabittir. Fiili çalışma süresi görevin süresindeki veya birim çalışmadaki değişimden etkilenmez. Kişilerin çalışma yüzdesi artarsa görev süresi kısalır. Kişinin zamanının %50'ini harcayarak 40 saatlik çalışmayla ve 10 günde geliştirdiği ekran eğer zamanın %100'ünü harcarsa yine 40 saatte ancak bu defa 5 günde tamamlanır. Çalışma birimi %25'e düşerse görev 20 gün sürer.

# 7-Kaynakların Planlanması

## 3.Kaynakların Göreve Atanması

Sıralanan bu durumların her zaman geçerli olduğu söylenemez. Örneğin sabit birim türünden 10 gün süren bir göreve, 9 kişi ilave etmek süreyi 1 güne düşürmez. Proje yönetim yazılımlarının bu konuda yaptıkları otomatik hesaplar ancak kontrol edildikten sonra plana uygulanmalıdır.

Kaynağın göreve atanmasıyla görevin gerçek süre ve maliyet tahmini ortaya çıkar. Plan gerçek durumu daha ayrıntılı yansıtmaya başlar. Atanan kaynağın becerisi (üretkenliği, *productivity*) ve işe ayırdığı verimli zaman (*utilisation*) da görev süresi üzerinde önemli ölçüde etkilidir. %70 üretken çalışan bir kişi zamanının %60'lık kısmında verimli çalışıyorsa 10 günlük bir iş  $(10 / 0.7) / 0.6 = 23$  gün sürecektir. Daha kalifiye bir personelin atanması, görevin daha kısa sürede bitmesini sağlayabilir.



# 7-Kaynakların Planlanması

## 4. Kaynak Dengeleme

Kaynakların işlere atanması sonrasında bazı kaynaklar üzerinde birden fazla iş aynı zamanda çıkışabilir. Bu durumda kişi veya diğer kaynak aşırı yüklenmiş olur. Kaynak dengeleme, aşırı yüklü bir kaynağın iş yükünün yapılacak düzenlemeyle azaltılmasıdır. Aşırı yüklemeleri düzenlemek için öneriler[Happy-20i0]:

- Fazla yükleme yapılmış kaynak tanımı ve bu kaynakların belirlenmesi
- Görev önceliklerinin belirlenmesi; Kişinin üzerindeki yüksek öncelikli görevler ilk olarak yerleştirilir. Düşük öncelikli görevlerse kaydırılabilir.
- Kısıtlara (*constraints*) sahip görevler önce yerleştirilir.
- Kaynak üzerindeki görevleri kısaltacak düzenlemeler yapılabilir.
- Bolluk süresi fazla görev önce kaydırılır.
- Erken başlangıç tarihine sahip görev, geç başlayana göre önce kaydırılır.
- Kaynağın başka projelere ayırdığı zaman azaltılabilir.

# 7-Kaynakların Planlanması

## 4. Kaynak Dengeleme

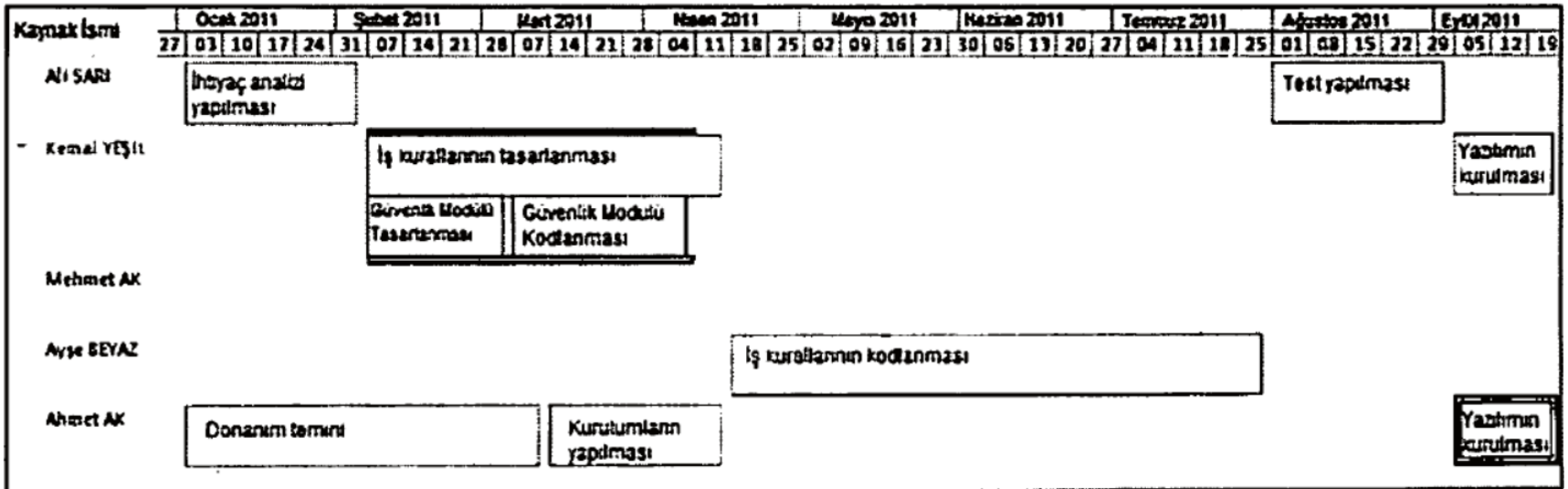
Proje planı üzerinden kaynakların yoğunluğunu gösteren raporlar da alınabilir. Bunu gösterebilmek için ilk örnekteki plana kaynaklar eklenmiştir.

No	Görev İsmi	Süre	Önceki görevler	Kaynak İsimleri
1	1.Analiz	5 hafta		
2	1.1 İhtiyaç analizi yapılması	5 hafta		Ali SARI
3	2.Tasarım	10 hafta		
4	2.1. İş kuralları tasarımı	10 hafta	2	Kemal YEŞİL
5	2.2. Güvenlik Modül tasarımı	4 hafta	2	Kemal YEŞİL
6	3.Kodlama	15 hafta		
7	3.1. İş kurallarının kodlanması	15 hafta	4	Ayşe BEYAZ
8	3.2. Güvenlik Modülü Kodlanması	5 hafta	5	Kemal YEŞİL
9	4.Test	5 hafta		
10	4.1. Test yapılması	5 hafta	7; 8	Ali SARI
11	5.Altıyapı Çalışmaları	15 hafta		
12	5.1. Donanım temini	10 hafta		Ahmet AK
13	5.2. Kurulumların yapılması	5 hafta	12	Ahmet AK
14	6.Yazılımın Kurulması	3 hafta	10; 13	Kemal YEŞİL; Ahmet AK
15	7.Devreye alma	Proje Sonu	14	Kemal YEŞİL; Ahmet AK

# 7-Kaynakların Planlanması

## 4. Kaynak Dengeleme

Kişinin iş yoğunluğu ve çakışmalar, proje planlama araçları yardımıyla bir takvim üzerinde gösterilir. "Kemal Yeşil" üzerinde aynı anda birden fazla iş olduğu semadaki farklı şekil ve renkteki gösterimden kolayca anlaşılır. Birden fazla kişiye atanan görevler de "Yazılımın kurulması" görevinde olduğu gibi farklı şekilde gösterilir.



Kişiler üzerindeki iş yükleri

# 7-Kaynakların Planlanması

## *4. Kaynak Dengeleme*

Bazı küçük üst üste binmeler ihmal edilebilir. Proje plan yazılımları aşırı yüklemeleri çözmek için proje, kaynak veya görev seviyesinde çeşitli otomatik çözüm araçlarına sahiptir. Bu işlem kaynak seviyelendirme (leveling) olarak adlandırılır. Ancak bu tür çözümler bir benzetim kabul edilmeli ve kontrol edilmeden uygulanmalıdır.

## 8- Proje Zaman Planı

Proje zaman planının hedefi, görev takvimlerini projeyi zamanında tamamlayacak şekilde oluşturmaktır. Projenin ilk aşamalarında başlangıç ve bitiş tarihlerinden en azından birisi kabaca belli olmaktadır. Görev süreleri de ilk aşamada aşağı yukarı tahmin edilir. Görevlere kaynakların atanmasından sonra görev süreleri daha net hale gelir. Görev süreleri kritik yol hesabıyla toplanarak, toplam proje süresi ortaya çıkar. Plan müşterinin veya üst yönetimin belirleyeceği kısıtlara göre tekrar incelenip düzenlenir ve gerekirse süre kısaltılmaya çalışılır.

# 8- Proje Zaman Planı

## 1. Zaman Çizelgesi Oluşturma Yöntemleri

Zaman çizelgesi aşağıdaki yöntemleri kullanarak görevler ve görevler arasındaki bağımlılıkların farklı açılardan analiziyle hazırlanır[PMBok-2008].

- 1. Ağ analizi:** İşlemler arasındaki ilişkilerin, iş akış mantığı ve ağ şeması kullanarak analizidir.
- 2. Kritik yol:** Proje süresinin uzunluğunu belirleyen işlemleri tespit yöntemidir. Kritik yol ileri ve geri doğru incelenerek bir görevin önce veya sonra başlatılabileceği süreler tespit edilir.
- 3. Kritik zincir:** Kaynak sınırlarının zaman çizelgesine etkisini incelemekte kullanılır. Kaynak kullanılabilirlikleri girildikten sonra kritik yol genellikle değişir.
- 4. Kaynak dengeleme:** Kaynak kullanımının bir süre aralığında sıkışmasının önüne geçilmesi ve düzenli kullanım sağlamak için planın tekrar düzenlenmesidir. Kritik yol bu durumda da değişebilir.
- 5. Senaryo analizleri:** Varsayılan bir özel durumunun gerçekleşmesi halinde planı nasıl etkileyeceği incelenir. Olumsuz durumları önlemek veya etkilerini azaltmak hedefiyle yapılan bir çalışmadır.
- 6. Önden gitme ve bekleme:** Planın önce başlama veya gecikme durumlarına göre tekrar incelenmesi ve gerekirse düzenlenmesidir.
- 7. Sıkıştırma:** Kaynak ilavesi ve işlerin paralel hale getirilmesiyle kapsam değişmeden plan süresinin kısaltılmasına yönelik çalışmalardır.

# 8- Proje Zaman Planı

## 2. Proje Takvimi Oluřturulması

Proje takvimi eřitli řekillerde belirlenir. Genellikle projenin bitiř tarihi veya sũresi belli olur. Bazı projelerin de bařlangı tarihleri belli olabilir. Oluřabilecek durumlar:

- **Sũresi belirli projeler:** Bu tũr projelerde projenin sũresi belirli ancak ilk anda bařlangı tarihi belirsizdir. Ŗzellikle ihaleyle alınan projelerde bu durum ortaya ıkar. Alıcı firma řartnamede sũreyi belirtir. İhale sũrecinin ne kadar sũreceęi net belli olmadığından ilk anda bařlangı ve bitiř tarihleri net deęildir. İhale sonucunda iřin bařlangı tarihi belirlendięinde son tarih de belli olmuř olur. Bu durumda projenin en uzun yolu mũřteri tarafından belirlenir. Yapılacak tũm iřler bu sũreye sıędırılmalıdır.

- **Bitiř tarihi belirli projeler:** Projenin bitiř tarihi kesin belirlidir. řirketlerde mali iřler, stok gibi dũnem esaslı alıřan birimler iin geliřtirilen projeler bu tũrdendir. Proje bařlangıcı gecikirse sũre kısalır. Ŗrneęin 2013 bařında devreye girecek bir proje 2012 Nisanda bařlarsa 9, 2012 Ekimde bařlarsa 4 ay sũrer. Yapılacak son iřin bitiř tarihi, proje bitiř tarihine eřitlenerek toplam sũre belirlenir. İřler bu sũreye sıędırılır. Proje yũneticisi, bu tũr projeleri mũmkũn olan en kısa sũrede bařlatmaya alıřmalıdır.

# 8- Proje Zaman Planı

## *2. Proje Takvimi Oluşturulması*

- **Başlangıç tarihi belirli projeler:** Bazı projelerin başlangıç tarihi belirlidir. Bitiş tarihini belirlemek de proje ekibine bırakılmıştır. Proje yöneticisi ve ekip, plandaki görev akışına göre toplam süreyi ve bitiş tarihini önerir. Bu durum ideal olmakla birlikte, bu tür projeler çok nadirdir.

Proje takvimi normalde günlük 8 saatlik bir çalışmaya göre belirlenir. Proje takviminde ayrıca resmi tatil günleri gibi projede çalışan tüm kişileri etkileyecek bilgiler bulunmalıdır. Projenin kritikliğine göre çift vardiyalı çalışma gibi farklı çalışmalar da takvime eklenir. Ancak planın ilk anda fazla mesai yapılacaktı gibi düzenlemesi, sonraki aşamalar için yedek kaynak bırakmayacağından, *önerilmez*



# 8- Proje Zaman Planı

## ***3. Kritik Yol Yöntemi***

Projenin bitmesi için izlenebilecek alternatif yolların, toplam süre olarak en uzun, kritik yol olarak isimlendirilir [Kelley-1959]. Toplam süre, projenin başlangıcından bitişine kadar birbirini takip eden görevlerin süreleri toplamıdır.

Kritik yol en uzun süreli yol olduğu için projenin toplam süresini gösterir. Kritik yoldaki görevlerin herhangi birinin gecikmesi proje süresini uzatır. Farklı bir bakışla kritik yol proje süresini uzatabilecek kritik görevleri bulmayı sağlar denebilir. Kritik yola paralel diğer yollar kaynaklarda düşük kritiklikte veya kritik olmayan şeklinde isimlendirilir. Ancak kritik olmayan ifadesi, bazı görevlerin öneminin, olduğundan daha düşük değerlendirilmesine yol açmamalıdır. Çünkü bu görevlerin de aşırı uzaması durumunda kritik yol yine değişecektir.

# 8- Proje Zaman Planı

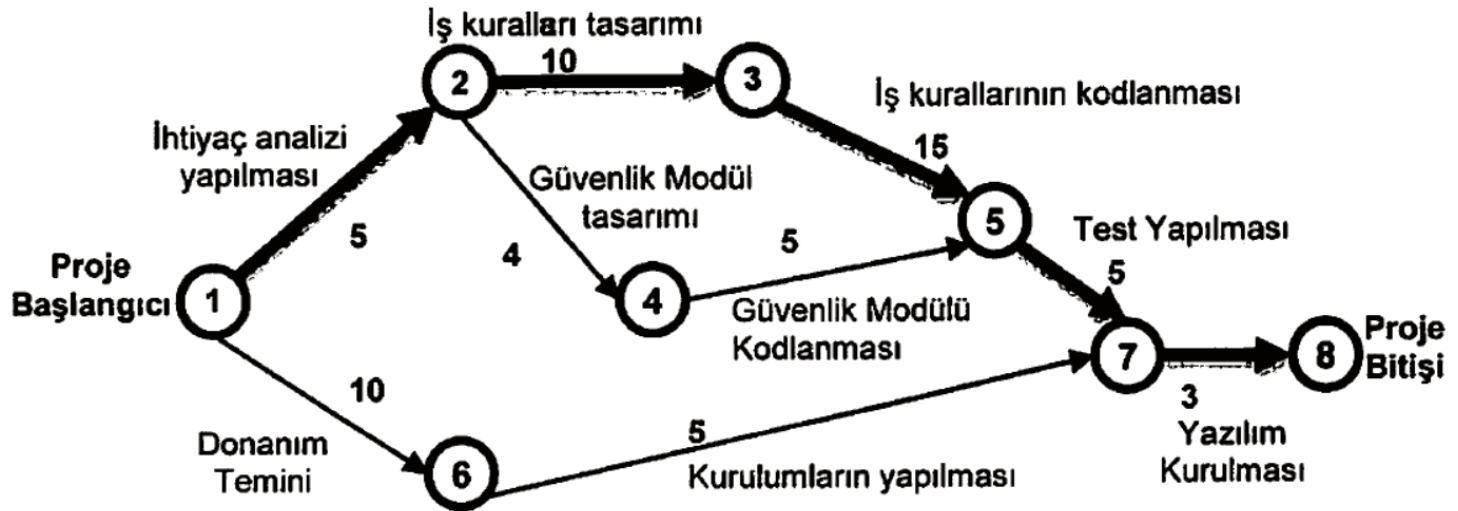
## 3. Kritik Yol Yöntemi

Görev No	Görev İsmi	Tahmini Ortalama Süre	En Kısa Süre	En Uzun Süre	Önceki görevler
1	1.Analiz	5 hafta	4	7	
2	1.1 İhtiyaç analizi yapılması	5 hafta	4	7	
3	2.Tasarım	10 hafta	8	11	
4	2.1. İş kuralları tasarımı	10 hafta	8	11	2
5	2.2. Güvenlik Modül tasarımı	4 hafta	4	5	2
6	3.Kodlama	15 hafta	14	20	
7	3.1. İş kurallarının kodlanması	15 hafta	14	20	4
8	3.2. Güvenlik Modülü kodlanması	5 hafta	5	7	5
9	4.Test	5 hafta	4	8	
10	4.1. Test yapılması	5 hafta	4	8	7; 8
11	5.Alt yapı Çalışmaları	15 hafta	9	20	
12	5.1. Donanım temini	10 hafta	5	12	
13	5.2. Kurulumların yapılması	5 hafta	4	8	12
14	6.Yazılımın Kurulması	3 hafta	2	4	10; 13
15	7.Devreye alma	Proje Sonu	-	-	14

## 8- Proje Zaman Planı

### 3. Kritik Yol Yöntemi

Yukarıdaki proje planındaki görevler arasındaki bağlantılar ağ şeması üzerinde de gösterilebilir. Görevlerin akışını gösteren birçok farklı gösterim şekli olmakla birlikte bu çalışmada hepsi ağ şeması başlığı altında incelenecektir.



Proje ağ şeması

# 8- Proje Zaman Planı

## 3. Kritik Yol Yöntemi

Örnekte kritik yol hesabının nasıl yapılacağı grafik üzerinde gösterilmiştir. Koyu renkli okların toplam süresi, projenin toplam süresidir. Bu süre 1 -> 2 -> 3 -> 5 -> 7 -> 8 yolundaki görevlerin toplamından 38 hafta olarak hesaplanmıştır.

Ağ şemasında görevlerin birleştiği ve ayrıldığı kesişim noktalarına *kilometre taşı* ismi verilir. *Kilometre taşları*, şekilde daire içerisinde numaralarla gösterilmiştir. Uygulamada ise sadece önemli noktalar, kilometre taşı veya kontrol noktası olarak belirlenir ve projenin bu noktalardaki performansına göre genel başarı değerlendirilir.

Görevlerin bazılarının uzaması kritik yolu değiştirebilir. Bu durumda kritik yol haricindeki görevlerin başlangıç ve bitiş tarihlerinin ne kadar değişebileceği önem kazanır. Bu amaçla öncelikle bir görevin en erken-en geç başlayabileceği ve bitebileceği noktaların hesaplanması gerekir. Böylece görevin zaman içerisinde ne kadar kaydırılabileceği tespit edilmiş olur.

# 8- Proje Zaman Planı

## 3. Kritik Yol Yöntemi

Aşağıdaki hesaplamalar görevler arasında bitiş başlangıç ilişkisi olduğu düşünülerek yapılmıştır.

- **En erken başlama zamanı:** Bir görevin en erken başlama zamanı, öncesindeki görevin en erken başlama zamanı ile yine öncesindeki görevin süresi toplamıdır. Öncesinde birden fazla görev olan bir görevin (bir görevin başlangıç noktasındaki kilometre taşına birden fazla paralel yolla ulaşılıyorsa) en erken başlama zamanı önceki görevler içinden erken başlama zamanıyla süresi toplamı en büyük olan seçilerek belirlenir.

Şekildeki 2 noktalı kilometre taşından sonraki görevlerin en erken başlama zamanı "ihtiyaç analizi yapılması" görevi için başlangıçtan itibaren 5 haftadır. "Test yapılması" görevinin öncesinde 5'nolu kilometre taşı mevcuttur. Bu noktaya 1 -> 2 -> 3 -> 5 ve 1 -> 2 -> 4 -> 5 yollarından ulaşılabilir. 1 -> 2 -> 3 -> 5 yolu için en erken başlama zamanı  $5+10+15=30$ . 1 -> 2 -> 4 -> 5 yolu için en erken başlama zamanı  $5+4+5=14$  hafta bulunur. Bu durumda 5 noktasındaki görevler için erken başlama zamanı 30 olarak seçilir. Çünkü her iki yoldan bu noktaya ulaşılmadan 5 noktasına bağlı bir görev başlayamaz.

# 8- Proje Zaman Planı

## 3. Kritik Yol Yöntemi

- **En erken bitiş zamanı:** Görevin en erken başlama zamanı ile süresi toplamıdır.
- **En geç başlama zamanı:** Bir görevin proje süresini uzatmayacak en geç başlama zamanıdır. Bu sürenin hesabı, proje bitiş noktasından başlanarak geriye dönük yapılır. Proje bitişinin en geç başlama zamanı ile en erken başlama zamanı eşit kabul edilir. Önceki bir görevin en geç başlama zamanı bu görevin süresi sonraki görevin en geç başlama zamanından çıkartılarak bulunur. Kendisini takip eden kilometre taşından birden fazla görev başlayan bir görevin en geç başlama zamanı, en geç başlama zamanı en küçük olan sonraki görev üzerinden belirlenir.

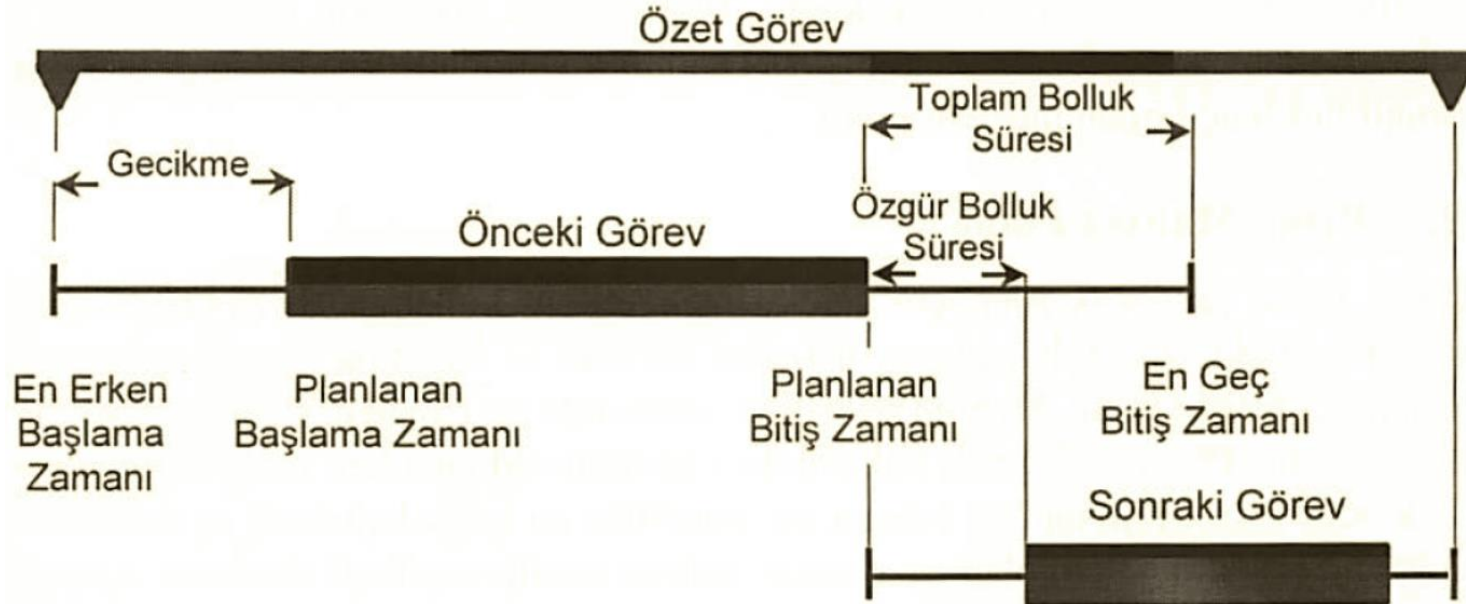
Şekildeki 8 numaralı kilometre taşına bağlı "Yazılım Kurulması" görevinin en geç başlama zamanı  $38-3=35$  haftadır. 5 noktasının en geç başlama zamanı  $35-5=30$  haftadır. 3 noktasının  $30-15=15$  ve 4 noktasının  $30-5=25$  hafta şeklinde hesaplanır. 2 noktasında plan iki yola ayrılmaktadır. Bu yollardan  $7 \rightarrow 5 \rightarrow 3 \rightarrow 2$  sırası izlendiğinde en geç başlama zamanı 5,  $7 \rightarrow 5 \rightarrow 3 \rightarrow 2$  izlendiğinde en geç başlama zamanı 21 olarak bulunur. Bu durumda 2 noktasında sonlanan görev için 5 değeri en geç başlama zamanı olarak seçilir.

- **En geç bitiş zamanı:** Bir görevin proje süresini geciktirmeden en son bitebileceği zamandır. En geç başlama zamanı ile görev süresi toplamıdır.

# 8- Proje Zaman Planı

## 3. Kritik Yol Yöntemi

Kritik yol haricindeki bir görevin bitişinde, kritik yolun süresini aşmayacak kadar, gecikme olması projenin gecikmesine yol açmaz. Bir görevin proje süresini uzatmayacak kadar geç başlayabileceği süreye, *toplam bolluk-gevşeklik süresi (total float-slack time)* denir. Bir görevin diğer hiçbir görevi geciktirmeyecek kadar gecikebileceği süreye; *özgür bolluk-gevşeklik süresi (free float-slack time)* olarak isimlendirilir. Kritik yolda bolluk süresi yoktur.



Görev süresi, bağımlılıklar ve bolluk süresi

# 8- Proje Zaman Planı

## 3. Kritik Yol Yöntemi

Herhangi bir görev için toplam bolluk zamanı; en geç başlama zamanından en erken başlama zamanı çıkartılarak bulunur. Örneğin 4 noktası için en erken başlama zamanı 1->2->4 yolu izlenerek 9, en geç başlama zamanı 8->7->5->4 yolu izlenerek 25 olarak bulunur. Bu durumda  $25-9=16$  haftalık bir bolluk süresi oluşur.

Herhangi bir görev için *özgür bolluk süresi*; görevlerin birbirine olan etkileri *toplam bolluk süresi* de dikkate alınarak incelenerek bulunur. Örnekteki donanım temini ve kurulumlar üzerinden giden yol 18 hafta tutmaktadır. Kritik yol 38 hafta olduğuna göre donanım temini ile çıkılan yoldaki görevler için 20 haftalık bir *toplam bolluk süresi* mevcuttur. Yollar üzerindeki bolluk süresi görevler üzerine dağıtıldığında asıl anlamını kazanır. "*Kurulumların yapılması*" görevi için özgür bolluk süresi 20 haftadır. "*Donanım Temini*" için ise özgür bolluk süresi sıfırdır. Çünkü "*Donanım Temini*" gecikirse "*Kurulumların yapılması*" da gecikir.

Proje kısmen incelenirken sadece projenin başlangıç ve bitişine göre değil aradaki paralel yolların da birbirine göre bolluk süreleri bulunabilir. Örneğin 2->3->5 yolu ile 2->4->5 yolu birbiriyle paraleldir. 2->3~>5 yolu 25 gün 2->4->5 yolu ise 9 gün süreye sahiptir. Bu durumda ikinci yol için  $25-9 = 16$  gün bir bolluk süresi vardır.



# 8- Proje Zaman Planı

## *3. Kritik Yol Yöntemi*

Projenin birden fazla bitişı olması kritik yolu belirlemeyi zorlaştırabilir. Bir projede birden fazla bitiş varsa, birden fazla kritik yol varmış gibi görünür. Uygulamada ise projenin tek bir bitişı mevcuttur. Bu durumda birleştirici yapay görevler tanımlanarak proje planında birden fazla bitiş noktası olması önlenabilir. [Fairley-2009]

Kritik yolun bulunması sonrasında projeyle ilgili analiz edilecek birçok yeni konu açığa çıkar. Öncelikle kritik yol kritik görevleri bulmayı sağlar. Böylece proje ilerlerken proje yöneticisinin süreyi uzatabilecek işlemleri takip etmesi mümkün olur. Kritik yolun kısaltılması için yapılabilecek çalışmalar ve kaynak ilaveleri belirlenir.

Kritik yolun bulunması ve süre hesaplanmasında çeşitli matematiksel algoritmalar kullanılmaktadır. Oldukça karışık bu işlemler için proje plan oluşturma yazılımları içerisinde hazır çözümler sunulmaktadır. Bu bölümde kritik yol hesaplanırken görevlerin ortalama süreleri kullanılmıştır. Tahmin kısmında kritik yolun gerçekleşme ihtimali farklı açılardan incelenecektir.

## 9- Proje Maliyet Planı

*Maliyet planı* yapılacak işler için gerekli maliyetlerin kullanılacak kaynaklara bağlı olarak plana eklenmesi, bütçeleme, harcama takvimi ve harcama kontrol işlemlerinin belirlenmesinden oluşur. Yazılım geliştirme sürecinde hedeflenen üründe birçok değişiklik olabilir. İlk anda kaynak kullanımları tahmini olduğundan, maliyet de tahmini olarak belirlenir. Proje toplam bütçesi ise genellikle en başta belirlenir ve sabittir. Bu yüzden en başta mümkün olan en ayrıntılı maliyet analiz yapılarak ilerleyen aşamalarda mali sıkıntı oluşması önlenmelidir. Bu amaçla kurum içi geliştirme ve dışarıdan temin etme gibi alternatifler arasında seçim yapılır.

# 9- Proje Maliyet Planı

## *1. Maliyet Kalemleri*

Bir yazılım projesindeki maliyet kalemleri aşağıda sıralanmıştır [Sandogan-2008]:

- **Yönetim:** Proje yönetiminde görevlendirilecek kişilerin istihdamı
- **Geliştirme Ekibi:** Geliştirmede görevlendirilecek kişilerin istihdamı
- **Yazılım Altyapısı:** Geliştirme, yönetim, test, veritabanı gibi yazılımlar
- **Donanım Altyapısı:** Yazılım geliştirme ve çalıştırma ortamında gerekli sunucu, istemci ve ağ altyapı maliyetleridir. Müşteriye sadece yazılım satılacaksa çalıştırma ortamı maliyeti projeye dahil değildir.
- **Teslimat:** Yazılımı müşteri ortamına kurmakla ilgili paketleme, yardım belgesi ve kılavuzlar üretme gibi maliyetlerdir.
- **Test:** Müşteri özel test ve sertifika isteyebilir. Ayrıca müşteri kendi kurumuna test ortamı kurulumu da isteyebilir. Bu maliyetler projeye eklenir.

# 9- Proje Maliyet Planı

## 1. Maliyet Kalemleri

- **Bakım Destek Hizmetleri:** Müşteriye garanti süresince sağlanacak destek, danışmanlık ve eğitim gibi hizmetler
- **Diğer giderler:** Çalışma ortamı kurma, kırtasiye harcaması, seyahat vb.
- **Değişim Maliyeti:** Yazılım projesinde sürekli değişiklik istekleri görülür. Hangi tür talebin *yeni istek* dolayısıyla ilave maliyet kabul edileceği, hangisinin mevcut bütçeden karşılanacağı proje başında kararlaştırılmalıdır.

Fiyat ve maliyet farklı kavramlardır[saridogan-2008]. Fiyat; sağlayıcı-satıcı tarafından kar, sürdürülebilirlik, müşteri beklentileri gibi birçok faktör göz önüne alınarak ürün veya işgücü için takdir edilen parasal değerdir. Maliyet ise ürünü geliştirmek için harcanan emek ve çabanın parasal karşılığıdır. Yazılımdaki değişim ve riskler yüzünden genellikle projenin fiyatı, maliyetin ve kar toplamına göre biraz fazla belirlenir.

# 9- Proje Maliyet Planı

## ***2. Kaynak Maliyetlerinin Plana Eklenmesi***

Projede kullanılacak her kaynak için maliyet oluşturan tüm işlemler ve her işlem türü için birim maliyet belirlenir. Bu sayede her görevin toplam maliyeti hesaplanır. Süre belirlerken olduğu gibi, burada da maliyet oluşturan tüm işlemler dikkate alınmalıdır. Bazı hizmetler maliyetsiz kabul edilirse, toplam proje maliyeti yanlış hesaplanır.

Maliyetin bir defalık mı yoksa sürekli mi olduğu incelenmesi gereken önemli bir konudur. Örneğin kişiye ücreti süreye bağlı veya bir defaya mahsus şekilde ödenebilir.

# 9- Proje Maliyet Planı

## 3. Bütçeleme

İdeal durumda projenin maliyet tahminleri proje planı analizi edilerek ortaya çıkar. Görevlerin tahmini maliyetleri toplanarak, *toplam tahmini maliyet* belirlenir. Buna göre gerekli bütçe ayrılır. Uygulamada ise genellikle projelere önceden bir bütçe ayrılır. Plan da bu bütçe sınırları içerisinde kalacak şekilde düzenlenir. Bu düzenlemede öncelikle kritik yol haricindeki işlemlere harcanan maliyeti azaltma çalışması yapılır.

Bütçeyle birlikte harcama takvimleri ve eğer varsa yükleniciye ödeme yapmak için bitirilmesi gereken iş listeleri (*hakediş*) de oluşturulur. Ödeme noktaları genellikle projelerdeki kilometre taşları olarak seçilir.

# 9- Proje Maliyet Planı

## ***3. Maliyet Kontrol ve İzleme***

Maliyet bütçesi oluşumundan sonra onaylanır. Belirlenen kilometre taşı geldiğinde yapılanlar kontrol edilir ve değerlendirilir. Yüklenici kullanılıyorsa ve tamamlanma kriterleri karşılanmışsa gerekli ödeme yapılır. Yazılım projelerinde ödeme şartları, ortaya çıkan ürüne göre çok net belirlenmelidir. Burada netlik olmaması en çok yazılım sağlayan yüklenici firmayı mağdur eder.

# 10- Devreye Alma Planı

Yazılımın devreye alma aşamasında izlenecek yöntem, görev alacak kişiler ve tarihler planlanmalıdır. Devreye alma planı eğitim, yazılım ve teknik altyapının kurulumu işlemlerini içerir. Yazılım kısmı bitmesine rağmen, donanım teminindeki aksaklıklardan dolayı devreye girmesi geciken projeler görülmektedir. Örneğin devreye alma aşamasında sunucu odasında yeni makine koymak için yer olmadığı fark edilebilir. Bu durumda yeni bir oda, ağ ve elektrik altyapısı hazırlanması aylar alabilir.

Yazılım projeleri diğer yazılım ve donanımlara bağlı olarak çalışırlar. Devreye alma planında bulunacak bilgiler:

- Altyapı yazılımları ve kurulacak yazılım bileşenleri arası ilişki
- Projenin çalışma ortamındaki donanımlar ve aralarındaki ilişkiler
- Donanımlar üzerinde yük paylaşım planı
- Yedekli çalışma planı
- Kurulumu yapacak kişilerin görev ve sorumlulukları
- Kurulumda oluşabilecek sorunlar ve bunların çözüm yöntemleri
- Yedekleme ve test sisteminin kurulumu
- Ağ altyapısı konfigürasyonları
- Sunucu odası elektrik vb. diğer altyapılarının planlanması



# 10- Devreye Alma Planı

Projenin teknik altyapısı yazılımın istenilen performans ve kararlılık düzeyinde çalışmasına uygun olarak planlanmalıdır. Projenin teknik hedeflerini sayısal olarak ölçmek için kullanılacak kriterler de yapılacak plana eklenebilir. Örneğin ekran açılma süresi veya rapor çalışma süresi gibi performans hedefleri belirlenebilir.

Proje kurulumu için yapılması gerekenler genellikle yazılım proje ekibinin sınırlarını aşar. Gerekli sunucu makinelerin temini, kurulumu ve işletim sistemi ayarları farklı birimler tarafından yapılır. Bu konuda kısa süreli dış uzmanlardan da yardım temin edilebilir. Diğer birim ve kişilerden alınacak destek en başta planlanmalı, ilgili kişi ve kurumlarla mutabakat imzalanmalıdır.

# 11- Eğitim Planı

Yazılım, birçok parçadan oluşan karmaşık bir üründür. Bu yüzden yazılımı devreye almadan önce son kullanıcıya eğitim verilmelidir. Eğitim, farklı kullanım amaçlarına göre çeşitlenir. Projenin üst düzey ayarları hakkında sistem yöneticisi ve kullanımla ilgili son kullanıcı eğitim alabilir. Büyük yazılımlarda, son kullanıcıya sadece doğrudan kullanacağı alt sistem ve bunun ana sistemle ilişkisi konusunda eğitim verilebilir.

*Eğitim planı* eğitime katılacak kişiler, eğitimi verecek kişi, eğitim içerik, yer, tarih ve süre bilgilerini içerir. Dış kaynak kullanılan projelerde, destek sürekliliğini sağlanmak için kurum içi yazılım ekibini hedefleyen, kod ve geliştirme ortamıyla ilgili eğitimler de planlanır. Bu plan, sadece proje sonunda eğitim verecek şekilde değil her aşamada ekiplerin birlikte ilerleyeceği dikkate alınarak hazırlanmalıdır.

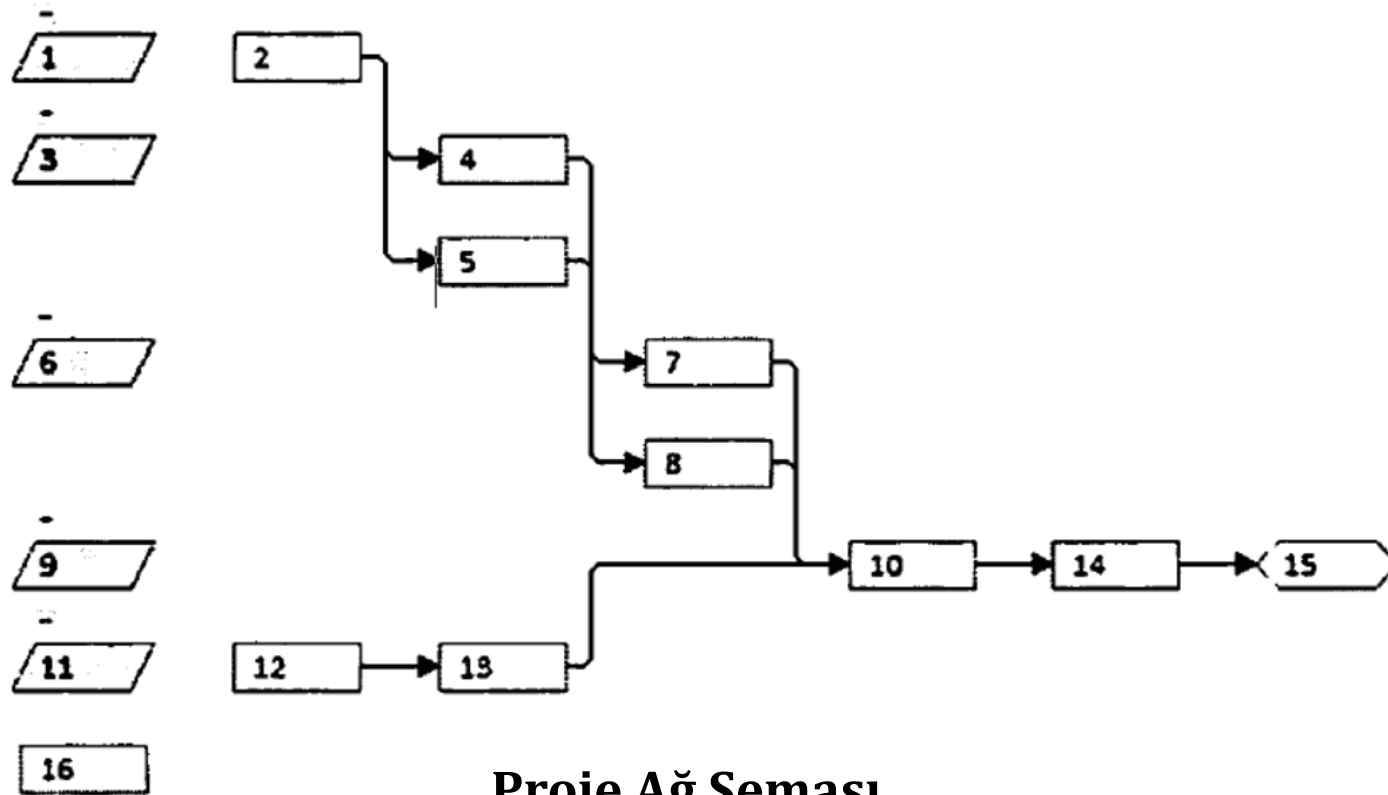
# 12- Planın Farklı Açılardan Analizi ve Tekrar Düzenlenmesi

Proje planında ilk anda sadece görev tanımı ve tahmini süreler mevcuttur. Kaynak tanımı, kaynak takvimi ve kaynak atanması ile plan zenginleşir. Gerçek durumu daha iyi yansıtmaya başlar. Bu aşamada plan farklı açılardan incelenerek süre, maliyet ve kaynak ihtiyacı tekrar düzenlenir. Kaynak dengeleme de bir tekrar düzenlemedir.

Plan, projeyle ilgili birçok bilgi içeren bir veritabanı gibi düşünülebilir. [Fairley-2009]. Plan kullanılarak işlerinin birbiri ile bağlantılarının ağ şeklinde gösterilimi, toplam maliyet analizi, kişi yüklerinin gösterilmesi, aşırı yük analizi, kişi ve diğer kaynakların verimi gibi konularda standart raporlar üretilmiştir.

Proje planındaki görevler arası ilişkileri yansıtan *ağ diyagramı* planı farklı açıdan incelemeye bir örnektir. Bu diyagramda görevlerin zamanda dağılımı ve birbiriyle bağlantılarını akış mantığında bütünleşik görmek mümkündür. Günümüzde görevlerin birbiriyle ilişkileri çok karmaşık bir yapı gösterir. Bu sebeple görevler arası ilişkileri hiyerarşi yerine akış şeklinde modellemek daha faydalıdır.

# 12- Planın Farklı Açılardan Analizi ve Tekrar Düzenlenmesi



**Proje Ağ Şeması**

Plan farklı açılardan analiz edildikten sonra ihtiyaca göre tekrar düzenlenebilir. Örneğin eğer proje maliyeti ayrılan toplam bütçeyi geçerse, tekrar bütçeleme yapılır. Piyasada birçok proje planlama ürünü mevcuttur. Bu ürünlerin özellikleri incelerken planı değerlendirirken faydalanılabilecek farklı araçlar tespit edilebilir.

# 13- Plan Kabulü

Proje ekibinin planı kabul etmesi ve benimsemesi yalnız başına yeterli değildir. Proje planı, tamamlanma aşamasında, projenin tüm paydaşlarının onayına sunulmalıdır. Burada yapılacak mutabakatın ilgili kişilerin imzalarıyla teminat altına alınması, sonrasındaki değişikliklerin makul düzeyde olmasına yardım eder.

## **Ortak Katılım ve Onay**

Planın hazırlanması sürecinde proje ile ilgili tüm kişilerin görüş ve önerileri dikkate alınmalıdır. Sürecin sonunda mutabakat kişilerin imzalarıyla güvenceye alınır. Proje sürecinde tüm ekibin planı kabul etmesi hesap verilebilirlik açısından çok önemlidir.

Plan hazırlanma aşamasında ekibe planla ilgili fikirleri sorulmalıdır. En önemli soru "*Kişinin kendisine atanan bir görevi hangi sürede bitirebileceği ?*" sorusudur. Bu konuda proje yöneticisi de kendi fikrini söyledikten sonra ortak bir mutabakata varılmalıdır. Görevi yapacak kişiyle ilk anda mutabakat yapmak planı yürüme, izleme ve kontrolde kolaylık sağlar. Kişi, ilk başta görevinin süresini kendisi belirlediği veya süre belirlenirken etkin rol oynadığı durumlarda işi yetiştirmek için daha fazla azim gösterir. Ayrıca sürenin kısa olduğundan şikayet etme ihtimali de azalır.

# 14- Proje Planına Yazılıma Özel İlaveler

Yazılım projeleri ve planlama alanının ortak özelliklerinden yola çıkarak, örnek yazılım proje planları önerilmektedir. Plan şablonları, kurumsal proje standartlarına dahil edilerek, özel bir proje için plan hazırlamak kolaylaşır. Yazılıma özel plan adımları:

- Yazılımla gerçekleştirilecek işe yönelik yapılacakların ayrıntıları
- Yazılımın bileşenleri
- Her yazılımda ortak olan ihtiyaç ve modüllere referanslar
- Yazılım geliştirmede kullanılacak modelin plana yansımaları
- Yazılım geliştirme ortamının oluşturulması
- Yazılım kurulumu için yapılacaklar
- Yeni yazılımın mevcut yazılımlarla entegrasyon yöntemleri

# 14- Proje Planına Yazılıma Özel İlaveler

## Görev İsmi

### Analiz aşaması

İşin kapsamına özel ihtiyaç analiz

Veri model analizi

Nesne modeli analizi

Güvenlik model analizi

Nesneler arası etkileşim veya akış modeli

Kullanıcı ara yüzü ihtiyaçların analizi

Raporlama analizi

Sistemler arası entegrasyon analizi

### Tasarım aşaması

İşin kapsamına özel tasarım

...

### Kodlama

...

### Test

...

### Devreye Alma

Kullanıcı eğitimleri

Donanım temini

Altyapı yazılımlarının temini ve kurulması

Geliştirilen yazılımın kurulması

Yazılım projeleri ve planlama alanının ortak özelliklerinden yola çıkarak hazırlanmış bir örnek yanda görülmektedir.

# 14- Proje Planına Yazılıma Özel İlaveler

Projedeki ihtiyaçlara göre yukarıdaki maddelere ilave veya çıkarmalar olabilir. Örnek proje planından faydalanmak, yapılması gereken işlemlerin unutulmamasını da sağlar.

**Tüm şablonların hedefi en az olması gerekenleri belirtmektir. Hiçbir şablon değiştirilmeksizin doğrudan kullanılmamalıdır!**

Örnek planda tasarım, kodlama ve test aşamasında yapılacaklar analize paralel olduğundan ayrıntılı yazılmamıştır. *İşin kapsamına özel ihtiyaç analiz* adımı iş kuralları ve özel algoritmalar gibi ayrıntıları kapsar. Bu adım onlarca maddeye genişleyebilir.

Yukarıdaki plan şelale modeline uygun olarak hazırlanmıştır. Yinelemeli veya çevik geliştirmenin bir çevrimine de uyarlanabilir. Risk, iletişim ve değişim gibi plan türlerinde yapılacaklar eklenmemiştir.



# 15- Özet

*Planlama* proje kısıtlarını dikkate alarak, eldeki kaynaklarla proje hedefine ulaşmak için izlenecek yöntem, kullanılacak kaynaklar, yapılacak görevler ve görev takvimlerinin belirlenmesidir. Proje planının sağlıklı olması proje başarısı için kesin şarttır.

Proje planı, projenin kapsamı belirlendikten sonra genel olarak hazırlanır. Proje ilerlerken ihtiyaç ve yapılacaklar netleştikçe plan daha ayrıntılı hale gelir. Planın tekrar gözden geçirilmesi periyodik olarak, proje aşaması veya sürüm geçişinde yapılabilir. Projenin başlarında analiz ve planlama için harcanan çaba ilerleyen aşamalarda projenin sağlıklı ilerlemesini sağlar, geri dönüş ve hataları azaltır. Küçük projelerde planlama için harcanan emek az olabilir. Ancak proje büyüdükçe bu ihtiyaç artar.

Planlanması gereken konular standartlaşarak çeşitli plan türleri ortaya çıkmıştır. Temel plan türleri kapsam, bütünleşme, risk, zaman, insan, kaynak, maliyet, teknik altyapı, gözden geçirme, test, eğitim ve kurulum şeklinde sayılabilir. Geliştirilecek yazılımının yapısına ve alanına göre farklı türde planların yapılması da gerekebilir.

# 15- Özet

Karmaşık yazılım ürünlerinin geliştirmek için yürütülen süreçler, yapılacak çalışmalar alt bölümlere ayrılarak daha anlaşılır ve yönetilebilir hale gelir. Planın da bu şekilde ayrıntılı hale getirilmesinde *iş bölümlene yapısı* kullanılır. Görevlerin birbiriyle ilişkilerini göstermek için *Gantt Şeması* ve *Ağ Şeması* gibi yöntemler önerilir. İşlerin bölümlenmesi üzerinden projenin bitiş süresini belirleyen kritik yol, bolluk süresi, en erken ve en geç görev bitiş tarihleri hesaplanabilir.

Kaynak, işin yapılmasını sağlayan kişi veya araçtır. Planın yürütülmesi için gerekli roller belirlenmeli ve görevlerle eşleştirilmelidir. Daha sonra role uygun kişiler istihdam edilerek göreve atanır. Bunun sonucunda bazı kaynaklar aşırı yüklenmiş olabilir. Kaynakların aşırı yükleri, kaynak dengeleme yöntemleriyle tekrar dağıtılır.

*Maliyet planı* projenin yürütülmesi için gerekli maliyetler, bütçeleme, harcama takvimi ve harcama kontrol işlemlerinin belirlenmesinden oluşur.

Yazılımların ve yapılacak çalışmaların ortak özelliklerinden yola çıkılarak her planda bulunacak konular önerilebilir. Bir kurum proje, birim veya yatırım alanına göre kendi standart plan şablonlarını belirlemelidir.