

1) İşveren proje ekibinde proje eşgüdüm birimi, kalite yönetim birimi, proje ofisi, teknik altyapı izleme birimi, yazılım üretim izleme birimi, eğitim izleme birimi ve kullanıcı eşgüdüm biriminden oluşmaktadır.

Proje eşgüdüm birimi => Üst düzey alan uzmanlarından oluşmalı.

Kalite yönetim birimi => Projenin gerekliliklere veya standartlara göre uygun olarak geliştirilip geliştirilmediğinin kontrolü.

Proje ofisi => Proje koordinasyonu ve yazışma gibi işlerden sorumlu birim.

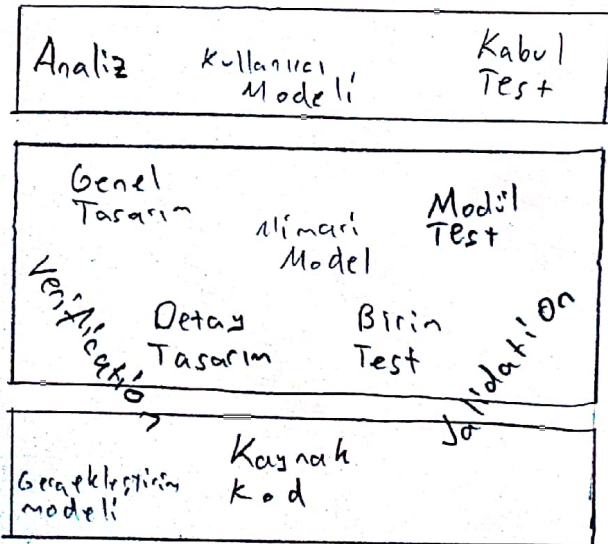
Teknik altyapı izleme birimi => Teknik ve donanım konularıyla ilgilenen birim.

Yazılım üretim izleme birimi => Yüklenici tarafından geliştirilen süreçleri inceleyen birim.

Kullanıcı eşgüdüm birimi => Geliştirilen uygulamayı kullananların yardım alabileceği birim.

Eğitim izleme birimi => Geliştirilen sistemin kullanıcılarının eğitim süreçlerinden sorumlu birim.

2)



Belirsizliklerin az ve iş tanımlarının kesin olduğu projeleri için uygundur.

Basit ve kullanımı kolaydır. Planlama ve test tasarımı gibi faaliyetleri kodlamadan önce gerçekleştirdiği için proje içerisinde çok zaman kazandırır. Bu nedenle süre modeline göre daha yüksek başarı şansı vardır. Hataların tespiti daha erken aşamalarda gerçekleşir ve bir sonraki aşamaya geçmesi önlenir.

Yazılım + Zaman

Uygulama geli' oldukca katı kurallara baēlıdır.

Yazılım, selale modelinde olduēu gibi geliftine aasamasında geliřtirilir. B. nedenle yazılımın erken prototipleri üretilemez.

Herhangi bir aasanda gereksinimler üzerinde deēiřiklik olursa test belgelerinin de diēer belgeler ile birlikte gūncellenmesi gerekir.

3) Ara model, temel modelin eksikliēini gidermek amacıyla oluřturulmuřtur. Őç basamaktan oluřur:

tiř gūcū hesaplama

- Maliyet carpanı hesaplama

- ilk iř gūcū deēerini dūzeltme

Maliyet carpanları 4 bařlıkta incelenebilir ve toplam 15 tane dir. B. gruplar ve carpanlar ; Őrūn ōzellikleri (RELY, DATA, CPLX), bilgisayar ōzellikleri (TIME, STOR, VIRT, TURN), personel ōzellikleri (ACAP, AEXP, PCAP, VEXP, LEXP), ve proje ōzellikleri (MODP, TOOL, SCED). B. maliyet etmenlerinin carpımı maliyet carpanını verir.



#### 4) İnsan kaynakları

- Yazılım kaynakları
- Donanım kaynakları

Planlama, bu kaynakların tanımını yapar ve zaman kullanımı, görev süreleri, edinilme zamanlarını planlar. Planlama, hangi tür elementlerin ne kadar süreyle ve projenin hangi kısımlarında görev alacağını belirler. Yazılımın geliştirileceği ortam, gerçek kullanım yapılan ortam dışında olmalıdır. Ayrıca uygulama ve geliştirme ortamlarının aynı konfigürasyonlara sahip olmaları ileride yapılacak kurulumu kolaylaştırır. Yazılım kaynakları, geniş ölçekte otomatize edilmiş ve bilgi sağlar destekli olarak kullanılmaktadır. Genel olarak CAD (Computer Aided Design) ve CASE (Computer-aided Software Engineering) olarak bilinir.



5)

## Tasarım kalite ölçütleri:

### Bağlaşım:

Tasarımı oluşturan modüller arası bağıntılarla ilgilidir.

Yüksek kaliteli tasarımda bağlaşım ölçümü az olmalıdır.

Bağlaşımın düşük olması; hatanın dalgasal özelliğinin azaltılması, modüllerin yakın kolaylığı, modüller arası ilişkilerde karmaşıklığın azaltılması, nedenleriyle istenmektedir.

Yalın veri bağlaşımı: Herhangi iki modül arası iletişim yalın veriler ile gerçekleşiyorsa bu iki modül böyle tanımlanır.

Karmaşık veri bağlaşımı: Herhangi iki modül arasındaki iletişimde karmaşık veri kullanılması durumunda modüller karmaşık veri paylaşımı olarak adlandırılır.

Denetim bağlaşımı: İki modül arasında iletişim parametresi olarak denetim verisi kullanılıyorsa bu iki modül böyle tanımlanır.

Ortak veri bağlaşımı: Eğer iki modül ortak bir alanda tanımlanmış verilere ulaşabiliyorsa bu iki modül ortak veri bağlaşımı olarak tanımlanır.

İçerik bağlaşımı: Modüllerin iç içe tanımlanması sonucu, bir modülün başka bir modül içinde tanımlanmış veri alanına erişebilmesi olanakları ve bu durum içerik bağlaşımına yol açar.

### Yapışıklık:

Bir modülün kendi içindeki işlemler arasındaki ilişkilere ilişkin bir ölçüttür. Modül gücü olarak da tanımlanır.

Tasarımda yapışıklık özelliğinin yüksek olması tercih edilir. Yapışıklık ile bağlaşım ters orantılıdır.

İşlevsel yapışıklık: Tek bir iş problemine ilişkin sorunu çözer.

Sirasal yapışıklık: Modüller arasında birinin çıktısı diğerinin girdisi olması durumunda bu isimle adlandırılır.

İletişimsel yapışıklık: Bir modülün içindeki farklı işlemler aynı girdi ya da çıktıyı kullanıyorsa bu modül iletişimsel yapışık olarak adlandırılır.



**Yordamsal yapışıklık:** Yordamsal yapışık modüldeki işlemler arasında denetim ilişkisi bulunmaktadır. İşlemlerin birbirleri ile veri ilişkisi yoktur, ancak işlenme sırası önemlidir.

**Mantıksal yapışıklık:** Mantıksal olarak aynı türdeki işlemlerin bir araya toplandığı modüller mantıksal yapışık olarak isimlendirilir.

**Gelişigüzel yapışıklık:** İşlemler arasında herhangi bir ilişki bulunmaz.