

YAZILIMIN YAŞAM DÖNGÜSÜ

- Planlama
- Analiz
- Tasarım
- Uygulama
- Bakım

PLANLAMA:

En zor adımlardan ama en önemlidir.

☐ Veri Toplama:

- En güzel çıktıyı prototip sağlar. (prototip kodları kullanılmaması gerekir.)
- Hızlı yanılma (fast fail)

☐ Tasarım Dökümanı:

- Ekibe ortak bir bakış açısı sağlar.
- Pusula görevi görür.(Detay-Kapsam-Sınır belirleyici)
- Proje yöneticisinin dostudur.Py kendi milestonelerini belirleyebilir.
- Ekibin yapılacaklar listesinin ana kaynağı.
- Proje ilerledikçe tasarım dokümanı da büyür.
- Tasarım dokümanı takımlar arası işbirliği ile oluşturulur.

ANALİZ:

☐ İsterleri kazma:

- İsterleri irdelemek önemli.(The pragmatic programmer - tracer bullet)
- İsterleri olabildiğince inceleyip, isterlerin gerçekleşmesi sonunda gerçekten hedeflenen bumu ?
- Düşündürücü ve belirleyici sorular sorulmalı.
- Kodu herkes yazabilir ama iletişim önemli.

☐ Risk Analizi:

- Benim işimde domain şampiyonları olmamalı.
- İşi analiz ederken aslında dile gelmeyen sorunlar çözülebilir.
- Bizden istenilen şeyin kapsam içinde mi bakılmalı.
- İyi kaliteli ve ilerleyen iş olmalı.
- Bir developer 62 saatte iş yapacak ama o iş 3 saat canlıda kalacaksa değer mi ?
- Milestonelerin ne kadar ayağı yere basıyor buda konuşulmalı.
- Beklenen ortak bir iş birliği. Her şeye evet deyip hiçbir şeyi yetiştirememek kayıp.

TASARIM:

☐ Mimari:

- Teknoloji seçimi.Eğer doğru teknoloji seçmezsek insan kaynağı da sıkıntı.
- Kendi sistemleriniz yeni isteklerinize cevap vermiyorsa planlamaya geri dönülebilir.Çünkü bu aynı zamanda isterlere cevap verme de etkili olacaktır.

UYGULAMA:

☐ Kod:

-Kodun isterlere ve planlanana göre yazılması.

☐ Test:

- Testlerimizi yazdığımız sürece uygulama güvenli hali ile canlıya çıkıcaz.
- Bunlar yazılımcını yazacağı testlerden bahsediyoruz.
- Uygulamadaki kalite burda ortaya çıkıyor.

☐ Takip:

- İşi nasıl takip edicez ? Yazılımın hangi durumda olduğunu görmek için ne kullanıcaz Trello?
- Bizim yazılımcı olarak ekip arkadaşlarımız tarafından takip edilebiliyor olmamız gerekiyor.
- Süreç akıcılığı için şeffaflık şart !

☐ Dağıtım:

- Son kullanıcı testinin yapılabileceği çıktı nasıl elde edilecek ?

BAKIM:

☐ İş Akışının Tekrarı Ve İyileştirilmesi:

- Özellikle yaşlı projeleri için önemli
- Geçmiş teknik borçlar, Yeni eklenen istekler burda yapılır.
- Geri bildirimler ve planlamaya aktarılması burda yapılır.
- İş akışını tekrar edilebilecek hale gelmesini sağlamak.
- Süreç bakımında burada yapılır.

Önemli Notlar.

- Ekip arası iletişim çok önemli.
- Kalite anlayışı gerekiyor.
- Her yeni döngüde gelişme beklenmeli(Sistemsal,yazılımsal)
- Yaşam döngüsü şirket kültürü ile doğru orantılıdır.