YAZILIM KALİTE GÜVENCESİ VE STANDARTLARI ÖDEVİ

KAMUSIS EYP SERVİSLERİ

Müzeyyen BÜTÜN

2021

**YAZILIM NEDİR?**

Bir bilgisayar nelerden oluşur sorusuna karşılık nasıl cevap verirdiniz? Bir bilgisayar en genel anlamıyla donanım ve yazılımdan oluşur. Donanım; bilgisayarın fiziksel parçalarını ifade ederken, yazılım ise fiziksel olmayan yani donanıma göre daha soyut olan ve bilgisayara yön veren, neler yapması gerektiğini söyleyen kodlar bütünüdür. Donanıma örnek olarak klavye,mouse vb. örnekler verirken, yazılıma örnek olarak yazılım dilleri ile yazılmış programlar örnek verilebilir. Yazılım dillerine ise C# PHP,Java , C++ gibi örnekler verbiliriz.Yazılım mühendisleri, bilgisayar mühendisleri ve başka bir bölüm okuyup, yazılıma yönelen kişiler bu programlama dillerini kullanarak yeni yazılımlar çıkartmaktadır. Şuanda kullanılan birçok web siteleri,mobil uygulamalar,masaüstü uygulamalar,web servisler,tasarım yapmak amacıyla kulanılan uygulamalar yazılımla yapılmış programlara örnek verilebilir. Bu uygulamaların amaca yönelik iyi yönde kullanılması insanlara katkı sağlayacaktır.

**YAZILIMIN HAYATIMIZDAKİ ÖNEMİ**

Günümüzde teknoloji her geçen gün ilerliyor. Bundan 10 yıl öncesini düşünecek olursak bu ilerlemenin ne kadar olduğunu anlayabiliriz. Mesela; eskiden akıllı telefon kavramı yoktu. Şu ana bakacak olursak akıllı telefon kavramı var ve her geçen gün akla gelmeyecek teknolojilerle geliştiriliyor.Televizyonlar önceden siyah,beyaz yayın yapıyordu ve de lcd,hd kavramları yoktu.Çamaşır ve bulaşık makinaları yoktu.Böyle sayabileceklerimizin listesi hayli fazla.Teknoloji her geçen gün ne kadar geliştiği söylemeden bile anlaşılabilir vaziyettedir. Dolayısıyla yazılımcıların yaptığı her uygulama kat be kat değerli hale geldi.Artık sadece çalışanlar değil okullarda bile küçük yazılımcılar yetiştirilmeye başlandı.

**Çocuklar için kodlama,** günümüzün en popüler eğitimleri arasına girmektedir. Kodlama,çocuklar için vazgeçilmez bir alan haline gelmiştir. Çünkü normal hayatta hayal ettikleri şeyleri kodlama vasıtasıyla gerçekleştirme imkanları bulunuyor. Çocuklar eğlenerek öğrenmeyi gerçekleştiriyorlar.Kodlama yani yazılım hem çocuklar için hem de yetişkinler için analitik düşünme becerisi kazandırır. Farklı düşünmeyi, problem çözme yeteneğini geliştirme imkanı sağlar. Bir çocuk kodlama öğrendiği vakit hayata bakış açısı bile değişiyor.

Aslında hayatımızın neredeyse her alanında yazılımdan faydalanmaktayız.Yapay zeka ile birçok alanda teknolojik gelişmeler sağlandı.**Yapay zekâ;** insan beynini yani zekasını taklit eden bir bakıma insanlar gibi düşünmeye, hareket etmeye çalışan bir sistemdir. Tabiki tam olarak insanlar gibi düşünmesi, hareket etmesi beklenemez ama benzer durumlarla karşılaşılabilir. Yazılımla yapılan bir sistem olduğu için yazılan kodlar neyse ona göre daha çok ilerleyecektir.  Bu teknoloji birçok kişi tarafından biliniyor ve bu teknolojinin neler yapabildiğini öğrendikçe **yapay zekâ** teknolojisine yöneliyorlar.Görüntü işleme: Yapay zekâ teknolojilerinde sık kullanılan alanlardan biridir. Yüz tanıma, internetteki çeviri siteleri görüntü işleme ile yapılabilmektedir.Ses tanıma: Aslında bize pek yabancı gelmeyecek Apple'daki siri uygulaması, android telefonlardaki google asistan uygulaması yapay zekâ ile ses tanıma işlemleri yaparak öncelikle söylenen sözleri algılıyor, sonra ise ona göre cevap veriyor. İşte tamda bu kısımda yapay zekâ devreye giriyor.Robotik Uygulamalar: Yapay zekâ sayesinde yapılan robotlar hem ses tanıma hem de görüntü işleme ile birçok işlem yapabiliyor.Tıp bilimi: Bilim adamları, doktorlar yapay zekanın birçok hastalığa çözüm olduğunu söylüyorlar. Mesela; kanser hastalığına uzun yıllardır bulunamayan çözüm yapay zekâ sayesinde bulunmuştur.Konum bulma servisleri: Bu servisler önce bulunduğunuz yeri algılıyor sonrasında ise gidilecek yeri seçim yaptırarak yapay zekâ ile gidebilmenizi sağlıyor.

Son yıllarda en çok kullanılan diğer uygulamalar ise sosyal medya uygulamalarıdır. Instagram,linkedin,facebook,twitter ve tiktok sosyal medya uygulamaları olarak sıkça kullanılanlardır.Bu uygulamalara sürekli yeni özellikler eklenmekte ve her geçen gün kullanıcı sayısı artmaktadır.Sosyal medya kullanıcılar tarafındano kadar çok kullanılıyorki bazen bağımlılık haline dönüşüyor.Tabi teknolojinin faydası olduğu gibi az da olsa zararı da vardır.Dolayısıyla teknolojiyi doğru şekilde kullanmak gerekir.

Günümüzde yazılım hayatımızdaki en önemli neredeyse 3 şey arasına girecek kadar değerli hale gelmiştir.

**YAZILIM KALİTE GÜVENCESİ VE STANDARTLARI NEDİR VE NEDEN KULLANILIR?**

Konuya başlamadan önce yazılımda kullanılan terimlerin birkaçından bahsedelim.

* **Hata:**  İnsan kaynaklı olup,genellikle hataların zor tespit edildiği söylenir.
* **Kusur:** Hataların görmezden gelindiği veya hatalar tespit edilememesi.
* **Arıza:** Hataların tespit edilemeyip kusur haline geleip bu kusurun çalıştırılmasıyla meydana gelir.

Bu terimler çok önemlidir ve bazen çok büyük problemlere yol açabilir. Bu yüzden çok dikkatli olmak gerekir.

Yazılımdan ve yazılımın hayatımızdaki öneminden önceki sayfada kısaca bahsettik.Peki yazılım yaparken nelere dikkat edilmeli, neler önemli ve kod yazarken herhangi bir standart var mı biraz da bunlara değinelim.

Bir yazılım yapıyorsak eğer belli bir kurallarımız,standartlarımız olması gerekiyor.Bu sadece yazılım için geçerli bir durum olmayıp hayatta yapılan birçok işte olması gereken durumdur.Dolayısıyla bunlar yazılım içinde geçerli bir durumdur.

Yazılım yaparak ortaya çıkarılacak olan ürünün veya programın istenilen özelliklerde,istenilen kalitede olması için verilen tüm emek ve zaman için yazılımın başından sonuna kadar geçen sürece yazılım kalite güvencesi denmektedir.Yazılım kalite güvencesinin asıl amacı ortaya çıkacak yazılım ürününün en kaliteli haliyle yani müşterinin isteğine uygun ve işlevsel olarak en üst düzeyde olmasını sağlayarak müşteriye ürünü teslim etmektir.

Yazılım kalitesinin özellikleri şu şekilde sıralanabilir.

* İşlevsellik
* Güvenilirlik
* Kullanım Kolaylığı
* Verimlilik
* Bakım Kolaylığı
* Taşınabilirlik

Bu özellikleri taşıyan yazılımlar müşteri tarafından daha çok tercih edilmektedir.Müşteriye ürün teslim edilmeden önce en iyisi nasıl yapılabilir,en iyisini nasıl teslim edilebilir şeklinde sorularla yazılım geiştirilmelidir.

Bu süreçte yazılımdaki hatalar tespit etme,hata çıkmasını engelleme,geçerleme ve doğrulama işlemleri yapılmalıdır.McCabe karmaşıklık ölçütüyle yazılımın ne kadar karmaşık olduğunu,riskli olduğunu ölçerek eğer fazla karmaşıksa ona göre önlemler alınmalıdır.

Aslında yazılım ile bir ürün ortaya koyabilmek ve bu ürünü çıkarma aşamalarında yazılım kalitesine uygun olarak bütün süreçlerini gerçekleştirmek ortaya çıkacak yazılımın değerini kat be kat üste katlayacaktır.Bu yüzden yazılımın kalitesine çok dikkat edilmeli hatta kod yazma standartları belirleyerek ürünün daha kalıcı ,daha kullanışlı olmasını sağlayabilmek gerekir.

Yazılıtım kalite güvencesinden bu kadar bahsetmişken yazılım kalite güvencesine nasıl başlanabileceğine bakalım.

* Yazılımın hangi kriterlere göre yapılacağını içeren şablonlar hazırlayarak
* Hazırlanan şablonun içeriğindekiler hangi sıraya göre yapılacak
* Yazılımda standartlarının uygulandığından emin olmak için teyit edin
* Önceden yapılan benzer projelerde neler yapılmış analizi yapın
* Önceden tecrübe ettiklerinizi,bildiklerinizi kullanın

Yazılım ve yazılım kalitesinden bahsettikten sonra ödev konusu olan **KAMUSIS\_EYP\_SERVIS** projesinden söz edeceğiz.

**KAMUSIS EYP SERVIS (EYP-EYAZIŞMA) PROJESİ**

**EYP NEDİR?**

**EYP’nin açılımı;** Elektronik Yazışma Paketi’dir. Yani e-yazışma paketidir.

Özellikle kamu kurum ve kuruluşlarının kendi aralarındaki yazışmalarda güvenli iletişim sağlayabilmesi için [**T.C. Kalkınma Bakanlığı**](http://www.e-yazisma.gov.tr/) tarafından geliştirilen ortak kurallar setine göre oluşturulmuş bir sıkıştırılmış dosya formatıdır.**T.C. Cumhurbaşkanlığı** tarafından yayınlanan kütüphaneyi kullanarak e-yazışma paketi üretiliyor.Eyp’ler **Elektronik Belge Yönetim Sistemine(EBYS)** entegre edilerek EBYS’deki belge oluşturma süreçlerini tamamlayacak nitelikte olup, EBYS’ye katkı sağlamıştır. Belgenin ekler kısmında eyp uzantılı dosya halinde bulunur.

Eyp içerisinde birden fazla bileşen bulunmaktadır.Bu bileşenler vasıtasıyla e-yazışma paketleri oluşturulmaktadır.Eyp bileşenlerini şu şekilde sıralayabiliriz;

* Üst yazı
* Üst veri
* Dağıtım
* İmza
* Mühür
* Şifreleme
* Paket Özeti
* Nihai özet
* Nihai üst veri
* Paraf özeti
* Core
* Ek

Eyp ile neler yapılabilir?

* Belgeyi ve belgenin eklerini zaman damgalı olarak imzalayabilmeyi,
* Bu belgeyi,eklerini,kurum bilgilerini,zaman damgasını toplu olarak tutmayı sağlar.

Eyp’nin birden fazla versiyonu bulunmaktadır.Bu versiyonlar eyp 1.3 ve eyp 2.0 dır.Eyp 1.3 te olmayan özellikler eklenere,hataya sebep olan durumlar kaldırılmış ve eyp nin bu hali ise eyp 2.0 olarak adlandırılmıştır.Eyp yani e-yazışma paketi oluşturulduğu gibi bu oluşan paketlerin kontrolünüde gerçekleştirme imkanı sağlanmıştır. Kontrolünü ise imzager ve imzager kurumsal uygulamalarıyla yapılabilir olanağı sağlanmıştır.Bu yüzden eyp kontrolü yapabilmek için bu prıgramları bilgisayarınıza kurmanız gerekecektir.

Eyp’ler şifreli ve şifresiz olarak 2 çeşittir. Şifreli ve şifresizdeki bileşenler farklıdır.

* Şifresiz EYP paketi bileşenleri

Üst Yazı

Üst Veri

Belge Hedef

Belge İmza

Paket Özeti🡪e-imza

Core

Ek-1

Ek-2

Niha-i Özet🡪e-mühür

* Şifreli EYP paketi bileşenleri

Şifresi EYP paketi bileşenleri

Şifreli İçerik Bilgisi

Eyp’den söz etmişken imza profillerinede göz atmadan geçmeyelim.

|  |  |  |  |
| --- | --- | --- | --- |
| **Profil** | **Zaman Damgası** | **İmza Formatı** | **İmza Ömrü** |
| P1 | Yok | BES | Düşük |
| P2 | Var | ES-T | Orta |
| P3 | Var | ES-XL | Çok Yüksek |
| P4 | Var | ES-XL | Yüksek |

**ELEKTRONİK İMZA PROFİLLERİ(P1,P2,P3,P4)**

**P1:** Anlık kullanım, güvenliği düşük seviyedeki uygulamalarda kullanılabilirlik,imzayı tek seferde doğrulamak için kullanılabilir,doğrulamada ÇiSDuP kullanılır.İmza zamanı belli değildir.

**P2:** Uzun ömürlü olmayan imzalar kullanılır,orta düzeyde imza boyutu kulanılır.İmza zamanı ispatı yapılır.

**P3:** Uzun imza ömürlüdür,güvenilirlik,imza zamanı ispatı yapılır.

**P4:** Uzun imza ömürlü,maksimum güvenilirlik,imza zamanı ispatı yapılır.

**ELEKTRONİK İMZA TÜRLERİ (BES,ES-T,ES-X LONG,ES-A)**

**BES:** Içerisinde zaman bilgisi bulunmamaktadır.İmzanın ömrü kısadır.Son kullanıcının sertifikasının ömrü kadardır.İmzanın ömrü bittikten sonra doğrulanamayacağı için zaman damgası alınması gerekir.Zaman bilgisi içerisine eklenebilir.

**ES-T:** BES imzanın özelliklerini içermekle birlikte zaman damgası bilgisinide içerir.Ayrıyeten zaman damgası almak gerekmez.İmza sertifikası süresi bittikten sonra ve sertifika iptal ettikten sonra da sertifika geçerlidir.

**ES-X LONG:** Bu imza türüde ES-T imza türünün özelliklerini içermekte olup kök ve alt kök sertifikalarıyla birlikte SİL ve OCSP cevaplarını içermektedir.

**ES-A:** ES-X LONG gibi kök ve alt sertifikalarını,SİL ve OCSP cevaplarını içermektedir.Bunlara ek olarak zaman damgasının uzun bir süre saklanması gerektiği durumlardada bu imza türü tercih edilmelidir.

**BES+ES-T+ES-XL**

**ES-A**

**BES+ES-T**

**BES+**

**ES X-LONG**

**ES-T**

**BES**

Eyp’nin ne olduğuna,eyp türlerine ve bileşenlerine,eyp kontrolünün nasıl yapıldığına,imza profillerine ve imza türlerine kısmen değindikten sonra KAMUSIS EYP SERVİSLERİ projesinde neler yapıldığına bakalım.

**KAMUSIS EYP SERVİS** projesi şu teknolojiler kullanılarak geliştirilmiştir;

* Visual Studio
* C# programlama dili
* Asp.net

İç yazı,dış yazı,yeni iç yazı,yeni dış yazı gibi belge tiplerinde eyp’ler kullanılıyor.Belge oluşturulduktan sonra belge açılarak en üst kısımda bulunan ekler sekmesinden eyp uzantılı eyp dosyasına ulaşılıp,imzager ile açıldığında paketin içinde geçerli olup olmayan bilgilerin listesini görebiliyoruz.

Eyp servis ilk olarak **Elektronik Belge Yönetim Sitemi(EBYS)** projesinin içerisinde dll olarak bulunmaktaydı.Yani EBYS projesi ile iç içeydi. Sadece eyp servis kısmında değişiklik yapılsa bile tüm **Elektronik Belge Yönetim Sitemi(EBYS)’ne** güncelleme atılması gerekiyordu.Bu yüzden bu durum hem yazılımcıyı hemde güncelleme atılacak kurumu yormaktaydı.

Şimdi ise eyp servis **Elektronik Belge Yönetim Sistemi’nden(EBYS)** çıkartılarak,ayrı bir proje haline getirildi.Böylelikle önceden toplu güncelleme atılması problemine çözüm getiriliş oldu. Yani artık eyp servislerde herhangi bir değişiklik yapıldığı zaman **Elektronik Belge Yönetim Sistemi’ne(EBYS)** güncelleme atılmasına gerek kalmayacak,sadece eyp servislere güncelleme atılması yeterli olacaktır.Eyp servislerin **EBYS’den** ayrılmasıyla kod yazımı daha kolay hale gelmiştir.Bu şekilde eyp servisler reasuble hale getirilmiştir.

Kurumlardaki eyp servislerinde oluşacak hataların tespiti için postman programı kullanılarak koleksiyonlar hazırlanıp kurum testi yapılabilmektedir.Dolayısıyla postman programının kullanımınıda iyice öğrenmek gerekiyor.Çünkü servis işlemlerinde test etmede çok işe yarayan bir programdır.

Eyp servisler ilk olarak 1.3 versiyonuyla çıktı ve buna yeni özellikler eklenerek 2.0 versiyonu çıkarıldı.Kurumlar şuanda her iki versiyonuda kullanabilir vaziyettedir.

Ayarlarda bulunan EYP20,SONONAYDAEYPOLUSTUR vb. Parametreler açılarak 2.0 versiyonu kullanılabilmektedir.Eyp servislerin 2 versiyonuda kurumların IIS sunucusuna web sitesi olarak kurulup,servislerde değişiklik olduğunda her iki servisede ayrı ayrı güncelleme atılıp,değişiklikler kontrol edilebiliyor.Güncellemeleride visual studioda projeleri açarak publish alıp ,kurumdaki IIS sunucusundaki eyp servislerin dosyalarının olduğu kısma atılabilyor.

Önceden eyp servisler **EBYS’den** ayrılmadan önce yazılan metotlar ve referans aldığı yerler etkilenirken,**EBYS’den** ayrılmasıyla eyp servislerde yapılan kısımdaki değişiklikler haricindeki yerler etkilenmez hale geldi.

Eyp servisler **EBYS’den** ayrılmadan önce kodlar çok okunaklı değil,birbiriyle iç içeydi.Şuanda ise kodlar daha okunaklı ve yorum satırlarıyla kod parçacığında ne yapılamk istendiği açıklanmış vaziyettedir.

Bir metot önceden birden fazla iş yaparken şuanda bir metot bir iş yapmakla görevlidir.Böylelikle kodun sürdürülebilrliği sağlanmıştır.Yani projeyi sadece yazan yazılımcı değil ,projeyle hiçbir alakası olmayan yazılımcıda projenin kaldığı yerden kod yazmaya devam edebilecektir.

Projedeki değişken isimleride daha net,daha anlaşılabilecek şekilde verilmeye başlanmıştır.Projelerin ayrılmasıyla hata çıkması durumunda daha az risk alınmış oldu.Çünkü ayrılmadan önce herhangi bir hata çıkması durumunda tüm **EBYS** patlarken,şu anda sadece eyp serviste patlama oluyor.Bu durumda güncelleme maliyetini arttıyordu.Ancak artık böyle bir durum yaşanmıyor.

Projenin ayrılmasıyla kurumlar başta olmak üzere yazılımcılarda daha rahat kod yazar hale geldi.Her geçen günde bu projelerde güzel gelişmeler yaşanarak,daha büyük proje haline gelecektir.

|  |  |  |
| --- | --- | --- |
|  | **KAMUSIS EYP SERVİS EBYS’DEN AYRILMADAN ÖNCE** | **KAMUSIS EYP SERVİS EBYS’DEN AYRILDIKTAN SONRA** |
| **Yazılım İşlevsel Kalitesi** | Tüm gereksinimleri tam olarak sağlamamaktadır,bununla birlikte kod yazmaka ve güncelleme atmak hayli zordur. | Tüm gereksinimleri tam olarak sağlamamaktadır,bununla birlikte kod yazmaka ve güncelleme atmak hayli zordur. |
| **Yazılım Yapısal Kalitesi** | Sağlamlık ve sürdürülebilirlik düşük seviyededir. | Sağlamlık ve sürdürülebilirlik yüksek seviyededir. |
| **Yazılım Kalite Güvencesi** | Yazılımda kalite güvencesi süreçlerinde istenilen tam olarak elde edilememiştir. | Yazılımda kalite güvencesi süreçlerinde istenilenler öncekine daha çok elde edilmekle beraber halen eksikleri bulunmaktadır. |
| **Yazılım Kalite Kontrolü** | Yazılımdaki kalite seviyesi düşüktür. | Yazılımdaki kalite seviyesi yüksektr. |
| **İşlevsellik** | İşlevsel düzeltilebilirlik orta seviyede,standartlara uygunluk düşük seviyede,işlevsel uyumluluk var ancak zor,güvenlik düşük,doğruluk düşük. | İşlevsel düzeltilebilirlik yüksek seviyede,standartlara uygunluk yüksek seviyede,işlevsel uyumluluk var,güvenlik yüksek,doğruluk yüksek. |
| **Güvenilirlik** | Bütünlük düşük, iyileştirilebilirlik orta,geri getirilebilirlik orta ,hata toleransı düşük seviyedir. | Bütünlük yüksek, iyileştirilebilirlik yüksek,geri getirilebilirlik yüksek,hata toleransı yüksek seviyededir. |
| **Kullanım Kolaylığı** | Birden fazla proje iç içe olduğu için kullanımı zordur. | Kullanımı öncekine göre daha kolaydır. |
| **Verimlilik** | Daha fazla uğraşı gerektirdiği için çok verimli değildir. | Verimlidir. |
| **Bakım Kolaylığı** | Analiz,test edilmesi daha karmaşık bir yapı olduğu için daha zordur. | Analiz,test edilmesi çok daha kolaydır. |
| **Taşınabilirlik** | Kurulumda ve diğer işlemlerde daha büyük bir proje olarak kurulduğu için taşınabilirlik zordur. | Taşınabilirlik sadece eyp servisler olduğu için daha kolaydır. |
| **Uygulama Geliştirilirken Yapılan Hatalar** | Genel anlamda hatalar yapılmakta olup,bu hataların çözümü daha zordur.Çünkü iç içe yapı olmasından kaynaklı hatalı olan bir yer projedeki başka yeride etkileyeceği için hatanın giderilmesi daha fazla zaman alacaktır. | Tek proje halinde olduğu için ve kod yazarken daha okunaklı yazılıdığından dolayı hata çıkma olasılığı daha düşük olmakla beraber bu hataların çözümü daha hızlı olacaktır. |
| **Yetersiz inceleme ve test** | Kod incelemesi yapılmamaktaydı ve testler çok önemsenmemekteydi.Bu yüzden hatalar anında fark edilmiyordu. | Yapılan her geliştirme incelenerek hataya anında müdahale ediliyor ve testler düzenli bir şekilde yapılıyor. |
| **Belgelendirme hataları** | Kod parçacıklarının yanlarında herhangi bir açıklama vs. bulunmamakla birlikte geliştirme yapıldıktan sonra üzerinden belli bir süre geçtikten sonra bu kodların ne için yazıldığı anlaşılmıyordu.Belgelendirmede yapılmıyordu. | Kod parçacıklarının yanlarında ne işlev gördüğü açıklamalarla belirtilmiş olup,belgelendirme işlemleri yapılmaktadır.Bundan dolayı kodu yeni gören bir kişi yazılıma hemen adapte olup devamını getirebilmektedir. |
| **Yazılım Geliştirme Hayat Döngüsü Aşamaları** | Kavrama,Geliştirme,Üretim,Kullanım,Destek,Kullanımdan Kaldırmak aşamaları üstün körü geçiliyor.Tam anlamıyla uygulanmıyordu. | Kavrama,Geliştirme,Üretim,Kullanım,Destek,Kullanımdan Kaldırmak aşamaları tam anlamıyla uygulanıyordu. |
| **Yazılım Geliştirme Modelleri** | Waterfall modeli kullanılmıştır. | Agile modeli kullanılmıştır. |
| **ModüllereAyırma** | Elektronik Belge Yönetim Sistemi içerisinde bulunduğu için modüllere ayırma burada olmamıştır. | Elektronik Belge Yönetim Sisteminden ayrılarak tek başına bir modül ,bir proje haline getirilmiştir.Böylelikle bağımsız görevler yapılabilir hale getirilmiştir. |
| **Eşzamanlılık** | Birden fazla kodu aynı anda çalıştırmak hatalara sebep olabilir dolaysıyla burada eşzamanlılıktan bahsedilemez. | Burada modüllere ayırma işlemi olduğu için farklı kod parçacıkları çalıştırılabilir yani eşzamanlılıktan söz edilebilir. |
| **Sözde Kod** | Herhangi bir kod parçacığını açıklayan yorum satırı vs. bulunmadığı için sözde koddan bahsedilemez. | Açıklamalar ve belgelendirme işlemleri yapıldığı için sözde kod kavramından bahsedilebilir. |
| **Programlama Stili** | Belirli standartlara uygun olarak görevleri yerine getirme burada olmadığı için herhangi bir programlama stilinden söz edilemez. | Daha düzenli,açıklama satırlarının olduğu,kod yazım standartları olduğu için burada programlama stilinden söz edilebilir. |
| **Versiyonlama** | Herhangi bir versiyon yoktur. | Eyp 1.3 ve eyp 2.0 olarak versiyonlara ayrılmıştır. |
| **Risk Analizi** | Risk analizi yapılmamaktadır. | Risk analaizi yapılmaktadır. |
| **Proaktif ve Reaktif testleri** | Burada tam olarak denemesede reaktif teste yakın testler yapılmaktadır. | Proaktif test yapılmaktadır.Yazılım geliştirme aşamasında henüz süreç bitmeden hatalar tespit edilmeye çalışılır. |
| **Belgelendirme Testleri** | Bu kısımda herhangi kod parçacığı açıklaması veya belgelendirme yoktur.Bu yüzden belgelendirme testi yapılmamaktadır. | Burada kod parçacıkları açıklamalı olup,bütün proje ayrıntısıyla belgelendirilmiştir.Belgelendirme testleri yapılmaktadır. |
| **Erişilebilirlik Testi** | Erişilebilirlik testi yapılmaktadır ancak zordur. | Erişilebilirlik testi yapılmaktadır,diğerine göre daha kolaydır. |
| **Güvenilirlik testi** | İki proje iç içe bulunduğu için bir hata veya arıza çıktığında düzeltme süresi biraz uzun olacağı için u testin sonucu düşük çıkabilir. | Tek proje halinde olduğu için bir hata veya arıza çıkması durumunda çözüm için geçecek süreç daha kısadır.Bunun içinde projenin daha güveniliriği yüksekolacaktır. |
| **Alfa Testi** | Kurumlara yapılan değişiklikler gitmeden önce alfa testi yapılır. | Kurumlara yapılan değişiklikler gitmeden önce daha detaylı alfa testi yapılır. |
| **Beta Testi** | Kurumlara yapılan değişiklikler gitmeden önce testler yapıldı,ikinci aşama olarak kurumlara bu değişiklikler atılarak beta testiyle test yapılmaktadır. | Kurumlara yapılan değişiklikler gitmeden önce testler yapıldı,ikinci aşama olarak kurumlara bu değişiklikler atılarak beta testiyle test yapılmaktadır. |
| **Sızma(Penetrasyon) Testi** | Güvenlik amaçlı sızma testleri yapılmaktadır. | Sızma testi yapılmamaktadır. |
| **McCabe Karmaşıklık Ölçütü** | Ayrı modüller halinde olmadığı için riskli karmaşık modüldür.Karmaşıklığını sayısal olarak ölçme imkanı bulunmaktadır. | Ayrı modüller halinde olduğu için fazla riskli olmayan basit bir modüldür diyebiliriz. |
| **Metriklerin Analizi** | Test Durumları Metrikleri,Hata / Kusur Metrikleri,Görev Metrikleri tek tek analiz edilebilir. | Test Durumları Metrikleri,Hata / Kusur Metrikleri,Görev Metrikleri tek tek analiz edilebilir. |
| **CASE Araçları** | Case araçları kullanılmamaktadır. | Case araçları kullanılmaktadır. |
| **Proje İlerleme Kontrolü Bileşenleri** | Risk yönetimi faaliyetlerinin kontrolü  Proje çizelgesi kontrolü  Proje kaynak kontrolü  Proje bütçe kontrolü bileşenleriyle daha çok proje yöneticileri ilgilendiği için yazılımcı bu bileşenlerle ilgil bir çalışma yapmaz. | Risk yönetimi faaliyetlerinin kontrolü  Proje çizelgesi kontrolü  Proje kaynak kontrolü  Proje bütçe kontrolü bileşenleriyle daha çok proje yöneticileri ilgilendiği için yazılımcı bu bileşenlerle ilgil bir çalışma yapmaz. |
| **McCall Faktör Modeli** | Doğruluk;hatanın tespit edilmesi zor olduğundan ve de belgelendirme işlemlerinin olmamasından dolayı doğruluğundan tam emin olunamayabilir.  Güvenilirlik;Hata oranı yüksek olacağı için güvenilirliği düşüktür.  Verimlilk;Büyük bir proje olduğu için,yapılan değişiklikler farketmeden hatalara sebep olacağı için ve bu hataları farketmek bazen hayli zor olacağı için verimlilik düşük olabilir.  Bütünlük;yüksek seviyede vardır.  Kullanılabilirlik;yeterince kullanışlıdır.  Bakım;yüksek seviyededir.Ancak daha maliyetlidir.  Esneklik;orta seviyede esnektir denilebilir.  Test Edebilirlik;test edilebilirdir ancak bazı durumlarda sıkıntı çıkarabilir.  Taşınabilirlik; yazılım taşındığı yere hemen uyum sağlayabilir.  Yeniden Kullanılabilirlik;orta seviyedir.  Birlikte Çalışabilirlik; Başka bir yazılımla birlikte çalışma seviyesi düşüktür. | Doğruluk;hata tespiti ve çözümü hızlı bir şekilde yapılacağı için,belgelendirme ve açıklamaların bulunmasından dolayı doğruluk yüksek seviyededir.  Güvenilirlik;Hata oranı yüksek olsada olmasada çözüm hızlı olacağı çin güvenilirliği yüksektir.  Verimlilik;yüksek verimlilik sağlar.  Bütünlük;yüksek seviyede vardır.  Kullanılabilirlik;yeterince kullanışlıdır.  Bakım;yüksek seviyededir.  Esneklik; yüksek seviyede esnektir.  Test Edilebilirlik;yüksek seviyede test edilebilirdir.  Taşınabilirlik; yazılım taşındığı yere hemen uyum sağlayabilir.Diğerine göre daha hızlı adapte olur.  Yeniden Kullanılabilirlik;yüksek seviyedir.  Birlikte Çalışabilirlik;Başka bir yazılımla birlikte çalışma seviyesi yüksektir. |