

Endüstride Yazılım Testi ve Kalite Güvencesi Etkinlikleri

(Uygulanan veya yakın gelecekte uygulanacak olan yazılım test ve tasarım doğrulama grubu aktiviteleri)

Doğan Demir

SW KG, YM, Test & Tasarım Doğrulama Mühendisi

Vestel Elektronik AS 45030, Manisa,

dogan.demir@vestel.com.tr

Özet. Yazılım mühendisliği kavramları arasında, yazılım testi ve kalite güvencesi etkinlikleri oldukça önemli bir yer tutmaktadır. Maalesef, Türkiye'deki sektörün alışlagelmiş firmaları bu kavramlar üzerine yeteri kadar önem vermemektedir. Bu yüzden kavramlarla ilgili olan hata takibi, yazılım versiyonlaması, yazılım yapılandırması, yazılım denetlemesi ve kaliteli yazılım hazırlanması tarzı konulara profesyonel yaklaşılmıyor hatta bu konular tamamen uygulanılmıyor. Oysa, Vestel bu konulara yeterli önemi vererek, yazılım test ve kalite güvencesi ile ilgili tüm başlıkları evrensel standartlara uygun olarak takip edip uygulamaktadır. Bu şekilde, yazılımla ilgili olarak oluşturulmuş yapıların(build) kataloglanması ve yazılımların versiyonlaması tarzı işlemlerle, her an yazılımın geçmişteki bir durumuna ulaşabiliyorken, yazılım denetlemesi ve yazılım takibi işlemleriyle de yazılımın geleceği ile ilgili tahminler ve değerlendirmeler yapılabilmektedir. Ayrıca yazılım test yollarının bir çoğunu uygulayarak, hatasız ürünler üretebilme hedefine yaklaşma imkanı oluşmaktadır. Böylelikle, müşteri odaklı firmamızda, aldığımız siparişlerin eksiksiz üretilmesi mümkün olabilmektedir. Yazılım mühendisliğinin kaçınılmazlarından olan bu kavramları uygulamaya başlamak tüm firmalar için olduğu gibi Vestel için de oldukça zor olmaktadır. Ama faydalarını görmeye başladığımız bu uygulamaları kararlılıkla sürdürmeye devam edeceğiz. Başlıca faydası olarak, Vestel'in ana vizyonlarından olan 'üretimde az zaman & yüksek kalite'yi örnek verebiliriz.

1 Giriş

Yazılım testleri, üzerinde çalışılan bir yazılım sisteminin kontrolünün yapılarak ihtiyaçların ve istenenlerin tam olarak karşılanması için var olan bir işlemler dizisidir. Test işlemleri kara kutu testlerinden beyaz kutu testlerine, ünite testlerinden sistem testlerine, birleştirme testlerinden uygulama testlerine kadar bir çok test adımını kapsamaktadır. Dökümanın geri kalan kısmında, Vestelde uygulanan test aşamalarının neler olduklarını ve uygulama yollarından, Vestelde uygulanan versiyon yönetiminden, yazılım yapılandırma yönetiminden ve 'iyi tasarım', 'yazılım yaşam devreleri', 'kalite güvencede dökümantasyon', 'test planı' gibi bazı kalite güvence kavramlarından bahsedeceğim.

2 Yazılım Test Uygulamaları

Profesyonel bir yazılım ortamında uygulanması gereken başlıca 6 çeşit test metodu vardır. Bunlar, kara kutu testleri, beyaz kutu testleri, birleştirme testleri, uygulama testleri, sistem testleri ve ünite testleridir. Şimdi her bir test metodunu inceleyip, Vestel'deki uygulamalarına göz atalım.

2.1 Kara Kutu testleri (Black Box testing)

Ürünlerin test edilmesi sırasında kullanılan en ilkel test metodudur. Bir takım test senaryolarının seçilip, yazılım kodundan bağımsız olarak takip edilmesi temeline dayanmaktadır. Bu yüzden ürünlerin fonksiyonel durumları ve inputlara verdikleri tepkilerin gözlenmesi uygulamada kullanılan kara kutu testlerinin kapsamını oluşturmaktadır. Bu noktada, yazılımın kodunda yapılan herhangi bir değişiklik veya data yapısındaki uyarlamalar kara kutu testleriyle kontrol edilen özellikler değildir.

Vestel'de kara kutu testleri mükemmel düzeyde yapılmaktadır. Bu konuda yıllardır çalışan başarılı bir ekip bulunmakta ve işlerini profesyonelce yapmaktadır. Yazılım yapılandırma müdüründen aldıkları yüklenmiş entegreleri, ilgili televizyona takip, daha önceden hazırlanmış test durumlarını kontrol etmeye başlarlar. Kontrol, kumanda yardımı ile, beklenen özelliklerin var olup olmadığını test etme seviyesindedir. Test sırasında farkedilen hatalar ve eksikler takibi kolaylaştırmak için kullandığımız PVCS Tracker adındaki programa girilir. Program müdürleri veya kaynak müdürleri ilgili hatanın çözülmesi için gerekli görevlendirmeyi yaparlar. Tracker, hataların takibi, ilgili kişilerin görevlendirilmesi gibi işlemlerde

kullanılmakta ve oldukça kullanışlı bir araç olarak gözükmektedir.

2.2 Beyaz Kutu testleri (White box testing)

Temel olarak ürün yazılımının mantığı bilgisine dayanan bir kontrol yapısıdır. Beyaz Kutu testleri yazılım kod ifadeleri, köprüleri, yolları, durumları gibi konuları kapsamaktadır. Test edici kadro genelde mühendis grubu insanlardan oluşmaktadır. Bu kişiler yazılım kodunu takip ederek hataları bulmaya çalışırken, kendileri de test yazılımı ekleyerek hata oluşmasını denetlerler. Beyaz kutu testleri sırasında yazılım üzerinde direk olarak düzenleme yapılabildiği için yazılım geliştirici personele oldukça yardımcı olunabilmektedir. Kara Kutu testlerinden farklı olarak, Beyaz Kutu testleri yazılım üzerinde hatanın var olduğu kod parçasını belirleyebilmektedir. O yüzden geliştiriciler hatanın nereden kaynaklandığını bulmak için tüm kodu incelemek zorunda kalmamaktadırlar.

Vestel’de Beyaz Kutu testi için yeni bir grup oluşturuldu. Şu anda geliştirici arkadaşlara dönülmeden bir takım hataları çözebildikçe ve kod üzerinde ufak değişiklikler yaparak var olan hataları farkedebilmektedirler. Performanslarının ilerleyen zamanlarda artacağı tahmin ediliyor.

2.3 Birleştirme testleri (Integration testing)

Birleştirme testleri, bir uygulamanın bir araya getirilen parçalarının, birleşmeden doğan veya doğabilecek durumlarını incelemek için kullanılır. Kod modüllerinin entegrasyonu, komponentlerin birleşimi ve uyumları ile ilgili testler Birleştirme Testleri kategorisine girmektedir.

Birleştirme testleri, Vestel’de yoğun bir şekilde uygulanmaktadır. Bunda temel neden ise Vestel’in evrensel bir firma oluşu, müşteri ve ortaklarının çeşitliliğidir. Bu şekilde ürünlerimizde bir çok değişiklik ihtiyacı doğmakta ve değişikliklerin modüler olarak üstesinden gelinmekte. Ortaklarımızdan, bazı projelerde temel yazılımı alıyor ve sonrasında bir çok modül ekleyerek yazılımın ürünlerimizde kullanılacak versiyonunu oluşturuyoruz. Bu versiyonların kontrolü için ise Birleştirme testleri dediğimiz testleri uyguluyoruz.

2.4 Uygulama testleri (Regression testing)

Yazılım ve yazılım ortamı üzerindeki herhangi bir değişiklik sonrasında yapının test edilmesi gerekmektedir. Bu tip testler Uygulama Testleri olarak geçer ve herhangi bir etkinin, yan etkilerine kadar tüm kontrolünün yapılmasını kapsar. En ufak bir değişikliğin türlü yan etkileri doğabileceğinden, tüm test durumlarının zaman zaman test edilmesi gerekebilir. Böyle durumlarda otomatik test aletlerinin oldukça faydası görülmektedir.

Vestel yazılım test ve tesa doğrulama bölümü bu testleri oldukça başarılı bir şekilde uygulamakta ve yakın gelecekteki düşleriyle de daha işler hale getirmeyi planlamaktadır. Bölümümüzdeki kadro yeteri kadar tecrübeli olduğu için, yapılan değişikliklerin ne tür yan etkileri olabileceğini tahmin edebilmektedirler. Böylelikle de tüm test durumlarını kontrol etmek yerine, sınırlı sayıda test yapılarak zamansal olarak kazançlı çıkmaktadır. Ayrıca yeni oluşturulan bir yan grupla da otomatik test araçları yapma çalışmaları başlatılmıştır.

2.5 Ünite ve sistem testleri (Unit vs. system testing)

Ünite testleri yazılım komponentlerinden sadece biri için yapılan testleri kapsarken, sistem testleri bir sistem halinde tüm komponentlerin oluşturduğu bütün için yapılır. Bir sistem testinin olumlu dönebilmesi için, her bir ünite için yapılan ünite testlerinin olumlu dönmesi gerekmektedir.

Vestelde özellikle belirtilmemesine rağmen, ünite ve sistem testleri yapılmaktadır.

3 Yazılım kalite güvence kavramları

Takip eden kısımlarda, bazı kalite güvence kavramlarının Vesteldeki anlamları ve uygulamaları hakkında bilgi vereceğim.

3.1 İyi dizayn, tasarım

Vestelde iyi dizayn denilince iki anlam ifade ediliyor. Birincisi, anlaşılır, açık, kolay değiştirilebilir ve uyarlanabilir yazılım kodunun hazırlanması, diğeri ise fonksiyonları müşteri taleplerine göre hazırlanmış bir yazılım oluşturulması. Bizim iyi bir dizayn kavramını uygulamamızda karşımıza çıkan en büyük engelimiz, bizim temel yazılımı destek veren ortak firmalardan alıyor oluşumuz. Eğer temel yazılım hazırlanıyorken, iyi dizayna önem verilmiş ise bizim için sağladığı kolaylıklarıyla birlikte dizaynı koruma imkanımız oluyor. Ama yazılımı aldığımız firma dizayn ile ilgili herhangi bir çalışma yapmamış ise, temeli oluşturulmuş bir yazılım dizaynını değiştirip iyileştirme imkanımız kalmıyor.

3.2 Yazılım hayat evreleri

Vestel için yazılım hayat evreleri, yazılımın elimize ilk ulaşmasından başlayıp, üzerinde yapılan tüm işlemlerin tamamlanmasına kadar devam eder. Değişmekle birlikte evreler şu şekilde sıralanmaktadır, ihtiyaç analizi, fonksiyon ve iç dizayn, doküman planlama, kod hazırlama, doküman hazırlama, modül entegrasyonu, test, destek, değişiklik, son test ve karar aşamalarıdır.

3.3 Kalite güvencede dokümantasyon

Dokümantasyon yazılım kalite güvencede oldukça kritik bir rol oynamaktadır. Bu kapsamda, öncelikle canlı dökümanlar olmak üzere, tarif namerler, dizaynlar, müşteri kural ve istekleri, gözlemci raporları, konfigürasyonlar, yazılım değişiklikleri, test planları, test durumları, hata raporları, kullanıcı kılavuzları gibi konular dokümanlaşmaktadırlar. Bu dokümanlar, Vestel bünyesinde hazırladığımız bir web uygulamasıyla arşivleniyor. BMT(Build Management Tool) adını verdiğimiz başka bir uygulama da versiyonları ve versiyonlarla ilgili dokümantasyonu tutmak için kullanılıyor.

3.4 Test plan

Vestel için test planı, yazılım testinin amacını, kapsamını, ele alınış biçimini ve odağının belirlenmesini içermektedir. Bir yazılım ürününün kabul edilebilirliğinin belirlenmesinde yazılım test planının oldukça önemi bulunmaktadır. Tamamlanan dökümanlar, test grubu dışındaki ilgililer için, ürünün nasıl ve niçin doğrulandığını anlamakta yol göstericidir.

3.5 Test durumları

Bir uygulamanın herhangi bir özelliğinin doğrulanması için, belirli girdilere göre çıktıların incelenmesi ve beklenen çıktılarla karşılaştırılması işlemlerini önceden hazırladığımız test durumlarına göre yapıyoruz. Kabaca, televizyon testleri için uyguladığımız test durumları şöyledir:

- Televizyon açma ve kapama
- Kumanda ve kontrol düğmeleri
- Yükleme, APS ve programlama
- Dış kaynaklar, scart, BAV, S_VHS, VGA
- Menu ekranları
- Resim, brightness, contrast, sharpness, color
- Ses, volume, headphone, equalizer, sound mode
- Özellikler
- Teletext
- PIP
- PAP
- OSD
- Ekran yazıları
- Servis ve test menusu

3.6 Yazılım denetimi

Vestel için, yazılım denetimi, yazılım üzerindeki hataların bulunması, var olabilecek hataların belirlenmesi işlemleridir. Genel olarak bu konuyla ilgilenen kişiler tüm yazılımı inceleyerek, yazılım üzerindeki yanlış geliştirmeleri çıkmaza girmeden durdurmak ve uyarılar yapmakla ilgilenmektedirler. Bu kişiler buldukları hata ve eksikleri düzeltmekle uğraşmazlar ama geliştirici arkadaşlara gerekli bilgilendirmeyi yaparlar. Vestel için en önemli kavram olan 'zaman'a verdiğimiz önem doğrultusunda, hatanın fark edilmesinin yerine hatadan korunma politikasının sürekli uygulamasına dikkat etmekteyiz. Yazılım denetimi de erken teşhis mekanizması olarak çalışmaktadır.

4 Yazılım Yapılandırma Yönetimi(SW Build Management)

Vestelde yazılım yapılarını belli bir düzenle yapmak, teste gidecek yazılımların onayı vermek, test öncesi

yazılımı takip ederek ilk müdahaleleri yapmak, yazılımların derlenmesi ve yapılandırması işlemlerini gerekli ifdef switch'leriyle düzenleyerek yapmak, herhangi bir anda belli bir yazılım versiyonunu çekerek entegrelere yüklenebilir hale getirmek gibi işlemlerin yapıldığı birimdir. Geliştirme grubu ile test grubu arasında köprü vazifesi görmektedir. Yazılım yapılandırma yönetimine destek olması amacıyla hazırlanan BMT adındaki database programıyla da yapıların takibi, test durumları, sahip oldukları özellikleriyle ilgili bilgileri tutuyor, yapı havuzundan da istenen yapıya ulaşılabilmesi sağlanıyor.

5 SW Version Management

Vestelde versiyon yönetimine de oldukça önem verilmektedir. Bunu düzenlemek için bir takım destek araçları kullandığımız gibi kendimize özgü versiyonlama sistemimizle de yazılımlarımızı herhangi bir karmaşaya neden olmadan etiketleyebiliyoruz. Versiyonlamanın bu derece önemli olmasının nedenlerden biri ürünlerimize üretim sonrasında da verdiğimiz destektir. Herhangi bir anda müşteri yazılım ile ilgili bir sorunla karşılaştığında servisine gidip yazılımın versiyon numarasıyla birlikte şikayetini bildirebilmektedir. Ayrıca yakın gelecekte, uydudan ,televizyonlarımızın yazılım versiyonlarını değiştirecek bir sistem geliştirme planımız da vardır.

6 Bitiriş

Yazılım test ve kalite güvence aktiviteleri yazılım dünyasında oldukça önemli yer tutmaktadır. Bu aktiviteler Vestelde aktif olarak çalışmakta, Ar-Ge'mizi en az modern Ar-ge'ler düzeyine çıkarmak için kullanılmaktadır. Yazılım testleri en kaliteli ürünleri oluşturma idealini gerçekleştirmek için kullanılıyorken, kalite güvence aktiviteleri en kaliteli ürünlerin en kaliteli ortamlarda oluşturulmasını sağlamakta etkili olmaktadır. Vestel henüz tam olarak bahsettiğimiz adımları tamamlayabilmiş değildir ama bu yolda uzun yol aldık. Hedef büyük ve koşmamız gerekiyor. Bir gün Vestel sektörünün bir numaralı ismi olacaktır.

Kaynakça

1. Demir, D., Vestel Electronics SW Source Maintenance and Security Policy,2003.
2. Demir, D., User guide for PVCS tracker, 2003.
3. Demir,D., &D_SWQA_HowToVersionAProjectENG110011_2003_1.1.2
4. Demir, D., R&D_SWQA_BMT_UsersGuide_ENG110012_2003_1.1.2