

Yazılım Geliştirme Süreci

Software Development Process

ÖZET

Yazılım geliştiren işletmeler için, yazılım geliştirme süreci, piyasa araştırması ve sistem analizi konularında, başarıyı artırıcı öneri ve çözümler.

ABSTRACT

Recommendations and solutions for software development companies on software development process market research to improve their success.

Özhan SEZGİN

e-mail: ozhan.sezgin@ayosis.com.tr

1. GİRİŞ

Bugün artık birçok kuruluş, rekabet güçlerini arttırmak, hatta denebilir ki ayakta kalabilmek için, bilgi teknolojilerinden yararlanmaktadır. Ancak, bilgi teknolojileri kapsamında, donanımsal yatırımlara ağırlık verilmesi, bilgi teknolojilerinden elde edilecek yararların sınırlı kalmasına neden olmaktadır. Bu eksikliğin farkına varan kuruluşlar için ise çıkış yolu oldukça labirentli olmaktadır.

Bilgi teknolojileri kapsamında, donanımsal teknolojilerin yanı sıra yazılım teknolojilerinden de yararlanmak başlangıçta çok kolay olmadı. Az sayıdaki yazılım üreticileri, piyasa ihtiyaçlarına uygun çözümler sunamadıkları için, piyasanın bu durumdan büyük kayıpları olmuştur. Buna rağmen, başlangıçtaki bu sürecin kazanımları, yazılım üreticileri için önemli deneyimler olarak ortaya çıkmıştır.

Yazılım üreticileri, hızlı gelişen yazılım araçlarını kullanarak profesyonel uygulamalar geliştirip piyasa ihtiyaçlarını karşılarken, kimi üreticiler çok amatörce yapılmış uygulamalarla piyasanın olumsuz etkilenmesine neden olmuşlardır.

Bilgi teknolojilerinde ülkemizdeki süreç akademik camiada daha hızlı ve önce olmuştur. Bu süreci özel sektör vakit kaybetmeden yakalayıp akademik camiayı geçmiştir. Bu gelişmeler, yazılım sektörü için tüketicilerin bilinçlenmesi yönünde olumlu katkıları beraberinde getirirken, yazılım sektöründeki üreticileri kurumsallaşmaya zorlamıştır. Önceleri, yazılım ihtiyaçlarını bireyler ile çözen birçok kuruluş, artık karşısında bir kurum talep eder hale gelmiştir.

Ülkemizde sayıları oldukça fazla olan yazılım üreticileri, ülkenin dört bir yanında faaliyet göstermektedir. Ancak ne yazılım üreticileri açısından ne de hedef sektörler açısından bilgi teknolojilerinden istenen düzeyde yararlanmak mümkün olmamıştır.

Günümüzde yazılım teknolojileri üreten firmalar için, yazılım geliştirme ve uygulama süreci kadar, pazarlama süreci de büyük önem taşımaktadır. Yazılım pazarı ile ilgili olarak, **PAM** (INTERPRO Pazar Araştırma Merkezi) yaptığı bir araştırmada, *"...Yazılım pazarı son üç yılda ortalama yüzde 26 oranında büyüdü. 1999 yılında sistem yazılımlarının yazılım pazarındaki payının yüzde 26, uygulama yazılımları payının ise yüzde 74 olarak gerçekleştiği tahmin ediliyor. ..."* açıklaması yapılmıştır.



Böylesine hızlı büyüyen bir pazara sahip yazılım sektöründe, hem yazılım üreticileri hem de hedef sektörler açısından uyulması gereken, adı konulmuş ürün geliştirme süreçlerinin oluşturulması gerekmektedir.

Yapmış olduğum bu çalışma, yazılım üretici firmanın, yeni bir ürünü geliştirme süreci ve satış/destek hizmetleri arasındaki ilişkileri ortaya koymaktadır.

2. YAZILIMIN ÜRETİM NEDENİ

Yazılım firmasının, bir yazılımın üretimine başlaması için belli başlı iki yol vardır. Bunlar; sektör kuruluşlarından gelen bir talep veya sektörün böyle bir ürüne ihtiyaç duyduğunun firma tarafından belirlenmesi ve ihtiyaç oluşturulması şeklindedir.

Yazılım firmasının yazılım üretme süreci, daha çok, sektörden gelen talep üzerine başlar. Bunun nedeni ise yazılım üreticisinin en azından bir satışı garanti altına alarak zarar etmek istememesidir. Her iki şekilde de izlenecek yol aynıdır. Ve bir çok yazılım firmasının yükselişini yada inişini etkileyen aşamalar bundan sonra ortaya çıkacaktır.

3. YAZILIMIN KULLANILACAĞI SEKTÖRÜN İNCELENMESİ

Geliştirilmesine karar verilen yazılım için bu süreçte ilk adım sektördeki satış potansiyeli olmalıdır. Yazılım sektörüne yapılan yatırımı öncelikli yatırım olarak görmeyen, bu yöndeki yatırımlarını düşük bütçelerde tutan bir sektör için, oldukça fazla zaman ve emek sarf edilerek geliştirilen uygulamanın öncelikli gereği yoktur. Yazılım sektörünün bu süreçte harcadığı emeğin karşılığını ne kadar alabileceği en başta ortaya konulmalıdır.

Üretilen yazılım için hedef sektörün incelenmesi sürecinde, sektöre ilişkin dikkat edilecek noktaların belli başlıları şunlardır:

- Yoğunluk.
- Ekonomik imkanlar.
- Yönetici ve işletme sahiplerinin, bilgi sistemlerine ilgisi
- Çalışanların bilgisayar kültürü.
- Yönetim organizasyon yapısı.

Sektör araştırılırken, varolan kuruluşların sayısal çoğunluğu göz önüne alınmalıdır. Çok az sayıdaki işletmenin faaliyet gösterdiği bir sektöre yönelik yazılım geliştirilmesi, sektör için maliyeti çok yüksek bir hal alabilir. Böyle bir durumda da, yazılımın geliştirilmesi sürecine hiç girilmeyebilir.

Diğer yandan önemli bir konu da yazılımı kullanacak kişilerin bilgisayar kültürüdür. Yazılımın, kullanıcı açısından oldukça kolay ara yüzlere sahip olma zorunluluğunun yanında kullanıcılara geliştirilecek yazılım için ne derecede eğitim verilmesi gerekliliği de bu aşamada planlanmalıdır.

Çok sayıda işletmenin olduğu bir sektörde, kuruluşlar bilgi teknolojilerine daha az önem veriyor olabilirler. Yada yeterli yatırımı yapabilecek bütçeyi ayıramıyor olabilirler. Bu gibi durumlarda da uygulamanın, gereksinim duyduğu donanımsal ihtiyaçlar ve yetişmiş personel ihtiyacı, analiz aşamasında mutlaka değerlendirilmelidir.

4. SEKTÖRDE KULLANILAN DİĞER YAZILIMLAR

Maalesef hala birçok yazılım firması, hiç araştırma yapmadan yazılım geliştirmeye başlayabiliyorlar. Ancak, uzun zamandır sektörde kullanılan ve kendi ürünlerinden çok daha iyi olan bir yazılım ile karşılaştıklarında, diğer ürünleri inceleme ve değerlendirme gerekliliği hissediyorlar. Tabii bu durumda yazılım üreticisi, her şeye yeniden başlamak yada ürünü rafa kaldırmak gibi iki alternatiften birini seçmek zorunda kalıyor.

Bunun yerine, yazılım üretimine başlamadan önce sektörde kullanılan diğer yazılım ürünlerini araştırıp, ürünlerin iyi ve kötü yönlerini tespit ederek daha üstün bir yazılım ortaya çıkartmak mümkündür.

Sektörde birden fazla ürün kullanılıyor olabilir. Bu durumda, rekabetin boyutları, ürünler arası farklılıklar, pazar payları, üretici firmaların bölgesel yoğunlukları ve müşterilerin düşünceleri mutlaka incelenmeli, yazılıma başlamadan önce değerlendirilmeye alınmalıdır.

Değerlendirmede öncelikler:

- Diğer üreticilerin faaliyetleri.
- Üreticilerin bölgesel konumu ve teknolojik gücü.
- Üreticilerin diğer bölgelerdeki faaliyetleri.
- Ürünlerin pazar payları.
- Ürünlerin üstün özellikleri.

Yazılım firmaları belli bölgelerde pazara tamamen hakim olabilirler. Bu gibi durumlarda boşluk olan bölgeler seçilmeli ve bu bölge üzerinde araştırmalar yapılmalıdır. Bayii alt yapısı, ulaşım ve ilgili diğer konular, yoğunlaşılan bölge için değerlendirilmelidir.

5. YAZILIM ANALİZİNİN YAPILMASI

Daha önce bahsettiğim, ön çalışmaların tamamlanmasının ardından firma üretime başlama kararı alabilir. Alınan bu karar, ön çalışmalar ışığında olmalı, olası riskler önceden belirlenmelidir.

Analiz aşamasında, firma belirleyeceği bir yada daha çok müşteri üzerinde çalışmalıdır. Ancak, analiz araştırmalarında birden çok müşteri ile çalışmak, sektörün tanınması ve daha kapsamlı analiz (yada çözümlenme) yapma olanağı sağlar.

Çoğu sektörde homojen bir yapının olmadığı, bilinmesi gereken önemli bir husustur.

Analize kaynak olan unsurlar:

- Yazılıma ihtiyaç duyulmasının nedenleri.
- Faaliyetlerin, başlangıçtan bitişe doğru aşamaları.
- Kullanılan, tüm belgeler, formlar, resmi ve diğer evraklar.
- Varsa kullanılan uygulama yazılımları ve kullanım amaçları.
- İş gören personelin görüş ve düşünceleri.
- Müşteri tarafından ihtiyaç duyulan, saklanacak veriler.
- Elde edilmesi gereken, yararlı bilgiler, rapor ve formlar.

Ayrıca:

- Sektörde lider kuruluşlar araştırılarak, daha sonra karşılaşılabilecek sorunlar ve çözümlerin tespiti ile elde edilen bilgiler.

Sektör bünyesinde yer alan küçük, orta ve büyük ölçekteki tüm işletmeler göz önüne alınarak, yazılımın işletmelerin gelişim süreçlerine uyum sağlayabilmesi de dikkate alınmalıdır. Gelişim süreçlerinde işletmelerin karşılaştıkları sorunların çözümleri geliştirilen yazılım kapsamında değerlendirilmelidir.

Yanı sıra, analizde dikkat edilmesi gereken bir diğer husus, elde edilen tüm bilgilerin düzenli olarak kağıda dökülmesi ve saklanmasıdır.

Yukarıda belirttiğim unsurlara bağlı kalınarak yapılan, analiz çalışmasının sonuçları, şüphesiz sağlam temellere oturtulmuş demektir.

6. ANALİZE UYGUN TASARIMIN YAPILMASI

Geliştirilen yazılımın kullanıcı ara yüzünün kolay ve kullanışlı olması tamamen profesyonel bir ekip işidir. Yazılımı kullanacak kişilerin yazılımı zorlanmadan kullanabilmeleri ve kullanırken de sıkılmamaları önemli unsurlardır. Tasarım aşaması, yazılımın hem kullanıcı ara yüzünü hem de programın omurgasını ortaya koymaktadır.

Tasarım aşamasının diğer teknik unsurları şöyle sıralanabilir:

A. Veri tabanı seçimi ve tasarımı.

- Veri tablolarının belirlenmesi.
- Sahaların belirlenmesi (Kayıt deseni).
- İndeks ve anahtar alanların belirlenmesi.
- Tablolar arası ilişkilerin kurulması (İlişkisel veri tabanı tasarımı).
- Tetikleyici ve prosedür kodlarının tasarımı.
- Veri tabanı seçimi.

A. Ara yüzlerin tasarımı ve geliştirme araçları.

- Yazılımın adı.
- İkon tasarımı.
- Kapak resmi (Yada açılış resmi).

- Kullanılacak derleyici yada yorumlayıcı tespiti.
- Kodlama ve iş akış diyagramlarının hazırlanması.
- Veri giriş ekranlarının tasarlanması.
- Sorgulama (Yada arama) ekranlarının tasarlanması.
- Yazıcı ve ekran raporlarının tasarlanması.

Yukarıdaki işlerden de anlaşılacağı gibi, tasarım iki ana başlık altında toplanmıştır. Buna göre, yazılımın alt yapısı olarak adlandırılan, veri tabanı tasarımı, ve görsel kanadı oluşturan ara yüz tasarımı ve araçların seçimi, yazılımın tasarım sürecindeki ana hatlardır.

Veri tabanı seçimi ve tasarımı;

Bir çok yazılım ürünü, çok iyi görsel araçlara sahip olmasına rağmen, ne yazık ki zamanla yok olup gitmiştir. Bunun nedeni ise, çok kullanıcı ortamlara aktarılamaması yada aşırı veri yoğunluğunu destekleyememesi gibi nedenlerdir.

Veri tabanı tasarımı ve seçiminde, tasarım ekibi, analizden gelen bilgiler doğrultusunda maksimum değerleri göz önüne almalıdır. Örneğin; yazılımın kullanılması ile, müşterinin bir yıl sonra milyonlarca kayda ulaşması bekleniyorsa, veri tabanı bu desteği verebilecek şekilde seçilmeli ve tasarlanmalıdır.

Yeri gelmişken bir konuya da değinmek gerekir, günümüzde en basit uygulama dahi açık sistem veri tabanı kullanılarak üretilmektedir. Desen gizlemek, artık müşteri için bile olumsuz bir opsiyon olarak değerlendirilmektedir. Müşteri doğal olarak, ileride karşılaşabileceği olumsuz durumlar için açık sistem veri tabanlarını tercih etmektedir.

Bunun yanı sıra, elektronik ticaretin gelecekte alacağı yeri düşünecek olursak, veri tabanı tasarımı ve seçiminin önemli bir unsur olduğu daha iyi anlaşılır.

Ara yüz tasarımı ve geliştirme aracı seçimi;
Bazen öyle uygulamalar tasarlanıyor ki, uygulamayı kapatmak için dakikalarca uğraşmak gerekebiliyor. İyi tasarlanmış ara yüze sahip uygulamalarda, hem kullanıcı kolaylık yaşar hem de firmanın destek ekibi rahat olur. Ayrıca, programcıların hakim olduğu yada destekleyebilecekleri kodlama araçları ile çalışılmalıdır.

Unutulmaması gereken nokta, tasarımının sadece müşteri için değil, firmanın destek birimi de düşünülerek yapılması gerektiğidir. Hiçbir müşteri, yeterli destek alamayacağını bildiği bir uygulama yazılımını satın almak istemez.

7. TASARIMIN BİLGİSAYAR ORTAMINA AKTARILMASI (KODLAMA)

Artık program kodları, eskiden olduğu gibi uzun uzun yazılmıyor. Gelişen teknoloji, kodlamayı daha az kullanıyor, araçlar (Tools) kullanmayı öneriyor. Ancak bu yöntem, iyi kod yazma gerekliliğini ortadan kaldırmıyor.

Tasarımı tamamlanmış bir uygulama, programcıya ulaştığında, programcının yapacağı ilk iş modülü iyi anlamak olmalıdır. Yazılacak modülü zihninde canlandıramayan programcı kodlamayı yapamaz.

Tasarımda hazırlanan, kodlama detayları, iş akış diyagramları, tablolar ve tetikleyici kodların, programcıya yazılı olarak verilmesi ve yapılan işin anlatılması gereklidir.

Kodlamanın unsurları şunlardır:

- Sağlam ve anlaşılır olmalıdır.
- Şekil (Yazım) kurallarına uymalıdır.
- Yardımcı fonksiyonlar ve özel kitaplıklarla sadeleştirilmelidir.
- Desteklenebilir olmalıdır.

Üzerinde durulması gereken diğer bir nokta da, yazılımın kodlama işlemi ile bitmediğidir. Gerçekte, bir yazılım ürününün kodlaması hiçbir zaman bitmez. Yapılan eklemelerle kod sürekli yenilenir. Bu durumun istisnası yoktur. Ancak zamanla gelen taleplere destek verebilen yazılımlar ayakta kalabilir. Tüm bu anlatılanlar, kodun sağlam ve anlaşılır yazılması gerekliliğini ortaya koyar.

Bazı programcılar, bir işi yerine getiren özel fonksiyonlara ihtiyaç duyarlar. Ama bunu yazmak onların günler, bazen haftalarını alabilir. Bu konuda, yazar *Charles Calvert Amerika' yı yeniden keşfetmeyin* başlıklı yazısında, "... Genelde böyle sistemleri, makul fiyata satıcılarından almak mümkündür ve eğer mümkünse, bunu istiyorsanız her seferinde yapmalısınız. Çok çok iyi programcılar genelde programlama araçlarını diğer programcılara satma yoluna giderler. TurboPower gibi bir dükkanı kendi oyunlarında yenemezsiniz. Onların kodunu satın alın ve onları kendi tarafınıza çekin. Kısacası Amerika' yı yeniden keşfetmeye gerek yok. ... " diyerek programcılara akılcı bir çözüm sunmaktadır.

8. YAZILIMIN TEST EDİLMESİ.

Test, üretimin son aşaması olmasına rağmen, aynı zamanda süreklilik arz eden bir diğer süreçtir. Yazılım üretiminde ilk testler geliştirme sürecinde programcı tarafından yapılır. Bununla birlikte, asıl hata ayıklama ve geri bildirim (FeedBack) hizmeti test ekipleri tarafından yapılır. Testler ve geribildirim müşteri yazılımı kullandığı sürece devam eder. Programcıların yaptığı testler ağırlıklı olarak, iş akışı değil, teknik testlerdir. Bu nedenle iş akışı yönünden yazılım testi, özel bir ekip tarafından yapılır.

Yazılım Test süreci:

- Programcı testleri.
- Test ekibinin testleri.
- Kullanıcı grubu testleri.

Test sürecinde, en faydalı geri bildirimler, son kullanıcı test gruplarından gelir. Yazılımın beta testlerinde mutlaka müşteriden test grupları oluşturması istenmelidir. Bu sayede müşterinin yeni yazılıma adaptasyonu da sağlanmış olur. Testlerin sonunda ilgili birimlerle birlikte değerlendirme toplantıları yapılmalıdır. Bazen hataların kaynağı analizde yada tasarımda olabilir. Yada kullanıcıdan gelen yeni bir istek var ise, bu talep doğrudan analiz ekibine iletilmelidir. Çünkü yazılımın mimarisinin temelleri analizciler tarafından hazırlanmıştır.

9. EĞİTİM VE ORYANTASYON MATERYALLERİNİN HAZIRLANMASI

Müşteri için kullanıma hazır hale getirilen yazılım, tamamlayıcı öğelerinde sağlanmasıyla bitmiş demektir. Ortaya çıkarılan uygulamanın eğitiminin ve dokümantasyonunun da sağlanması tamamlayıcı öğeler olarak düşünülmelidir.

Satış ve destek aşamalarında kullanılan en önemli malzeme, dokümanlardır. Altyapı gereksinimleri, kurulum, ayarlar ve bunlara ilişkin çeşitli eğitsel dokümanlar, uzman kişilerce hazırlanmalıdır.

Satış öncesi yapılacak hazırlıklar;

- Kurulum ve ayarlama dokümanları.
- Uygulama eğitim programlarının oluşturulması.
- Uygulama kullanım kılavuzu hazırlanması.
- Sıkça sorulan soruların cevaplanması.
- Bayiiler için eğitim programı oluşturulması.
- Destek birimi eğitim programı oluşturulması.

Yazılım sürecinde yer alan tüm birimler, her zaman son kullanıcıdan birkaç adım önde olmalıdır. Bir sonraki adımda; uygulamanın çoklu ortamlara, internet ortamına veya başka etkileşimli ortamlara taşınması istenebilir. Böylece, belki başlangıçta düşük maliyetlerde hesaplanan uygulama, değerini de arttıracak teknik özelliklere sahip olabilir.

10. SONUÇ

Anlatılan tüm bu süreçler, detaya inilmeden sadece ana hatları ile belirtilmiştir. Sonuç olarak, müşterinin verimliliğini arttıran en önemli teknolojik araç uygulama yazılımlarıdır. İyi bir uygulama yazılımını üretebilmek için, üretim süreci unsurlarına bağlı kalınarak çalışılmalıdır. Aşağıda görülen grafikte, uygulama yazılımının, müşterinin performansına ve verimliliğine olan etkisi daha açık olarak görülebilmektedir.



Bu bilgiler ışığında ortaya çıkan ve ayrıca üzerinde durulması gerekenler şunlardır:

1. Başarı için sadece bilgi yeterli değildir, bilgiyi kullanacak tekniklerinde firma çalışanları tarafından bilinmesi gerekir.
2. Kaliteli yazılım firması, geliştirdiği ürünü destekleyebilen ve onu zamana uydurandır.
3. Uygulama yazılımının kalitesi, satış, destek ve performans bileşiminin büyüklüğü ile ölçülür.
4. Yazılacak uygulamanın kullanılacağı sektörü en iyi şekilde bilmek analizcinin ve destek biriminin görevidir.
5. Geliştirme sürecinde analiz için harcanan süre ne kadar uzun olursa olsun boşa harcanmış sayılmaz.
6. Tasarımın her aşamasında ekip içinde bir psikolog doktorun yada EİP(Eğitim İletişim Planlama) uzmanının olması, ürünün performansını arttıran faktörlerdendir. Kullanıcının iş yapma isteğini ortadan kaldıracak renk ve dizaynlar şüphesiz ürün satışını olumsuz etkiler.
7. Yazılımda güvenlik iki türdür. bunlar, sistem güvenliği ve veri güvenliği olarak adlandırılır. Müşteri için her ikisi de vazgeçilmez iki unsurdur.
8. Yazılım firması en yoğun zamanlarında bile araştırma-geliştirme faaliyetlerini sürdürmeli ve yenilikleri tüm ekiplere yaymalıdır. Firma içinde bilgi paylaşımı, kaynakların herkese açılması ile yeterli ilerlemeyi sağlamıyorsa, bilginin dolaşımı, hizmet içi eğitimler ve sosyal etkinlikler ile sağlanmalıdır. Yazılımın, ekip işi olduğu herkesçe kabul gören bir gerçektir ve en önemli kaynağı da insandır.
9. Yazılım geliştirme sürecinde ekipler arası koordinasyon sağlanamamışsa, ortaya çıkan ürün toplama bilgisayara benzer ve entegrasyon için ayrıca vakit ayırmanız gerekir.
10. Yazılım üretiminde, düşünce üretime kaynak olan temel hammadedir, bu nedenle çalışanların da düşünecek uygun şartları olmalıdır. Düşüncenin olmadığı bir yerde üretim olmaz, ancak kopyalama olur.
11. Birçok kişi farkında olmasa da, dünyanın en büyük yazılım üreticileri dahi, ürünlerine yamalar yapmaktadır. İşin esprisi bunu müşteriye en az etkiyle yapabilmektir. Oysa ki ülkemizde hala ulusal yazılımlarda yama yapmak yerine, yeniden yükleme yapılmaktadır. Üretici yazılımlarında bu sorunu ortadan kaldıran yöntemleri mutlaka geliştirmelidir.

KAYNAKÇA

[1] BTHABER dergisi, 259. Sayı, Yıl 2000.

[2] DELPHI UNLEASHED, SAMS PUBLISHING, Charles Calvert, 1997.