

Gereksinim Analizi

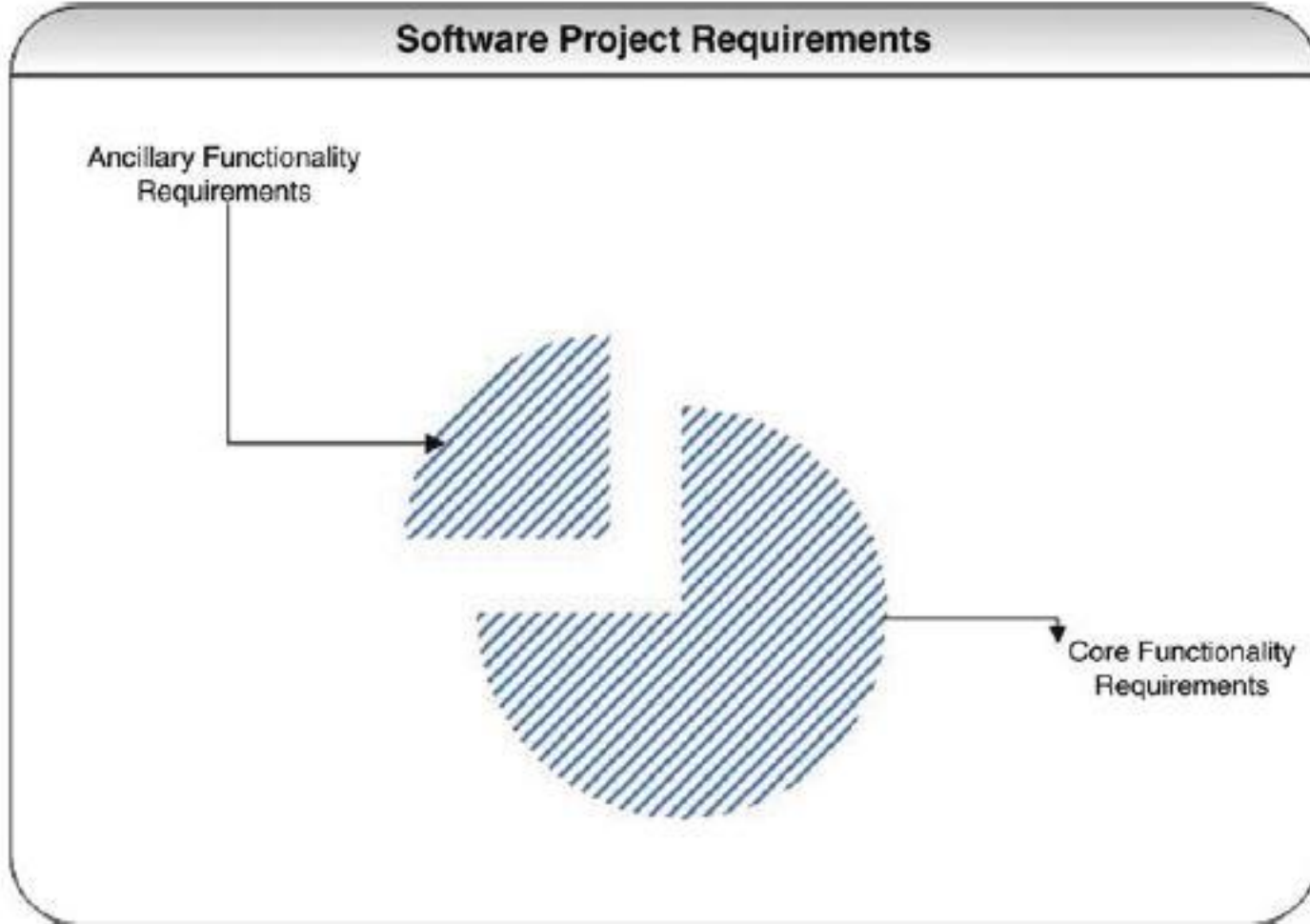
Gereksinimlerin Analizi

Gereksinimleri Anlamak

Gereksinimlerin Sınıflandırılması

- 1. İşlevsellik ile ilgili konular** : Seçilen iş süreçlerini yerine getiren ve sonuçları son kullanıcılara sunan gereksinimlerdir.
- 2. Ürün yapımında dikkat edilmesi gerekenler** : Ürünün daha sonra da sürdürülmesinin yanı sıra, ürünün verimli bir şekilde oluşturulması için gereken şartlardır.
- 3. Kaynak hususları** — yazılım geliştirme gereksinimleri farklı kaynaklardan sağlanmaktadır.

İşlevselliğe göre sınıflandırma



Gereksinimler,

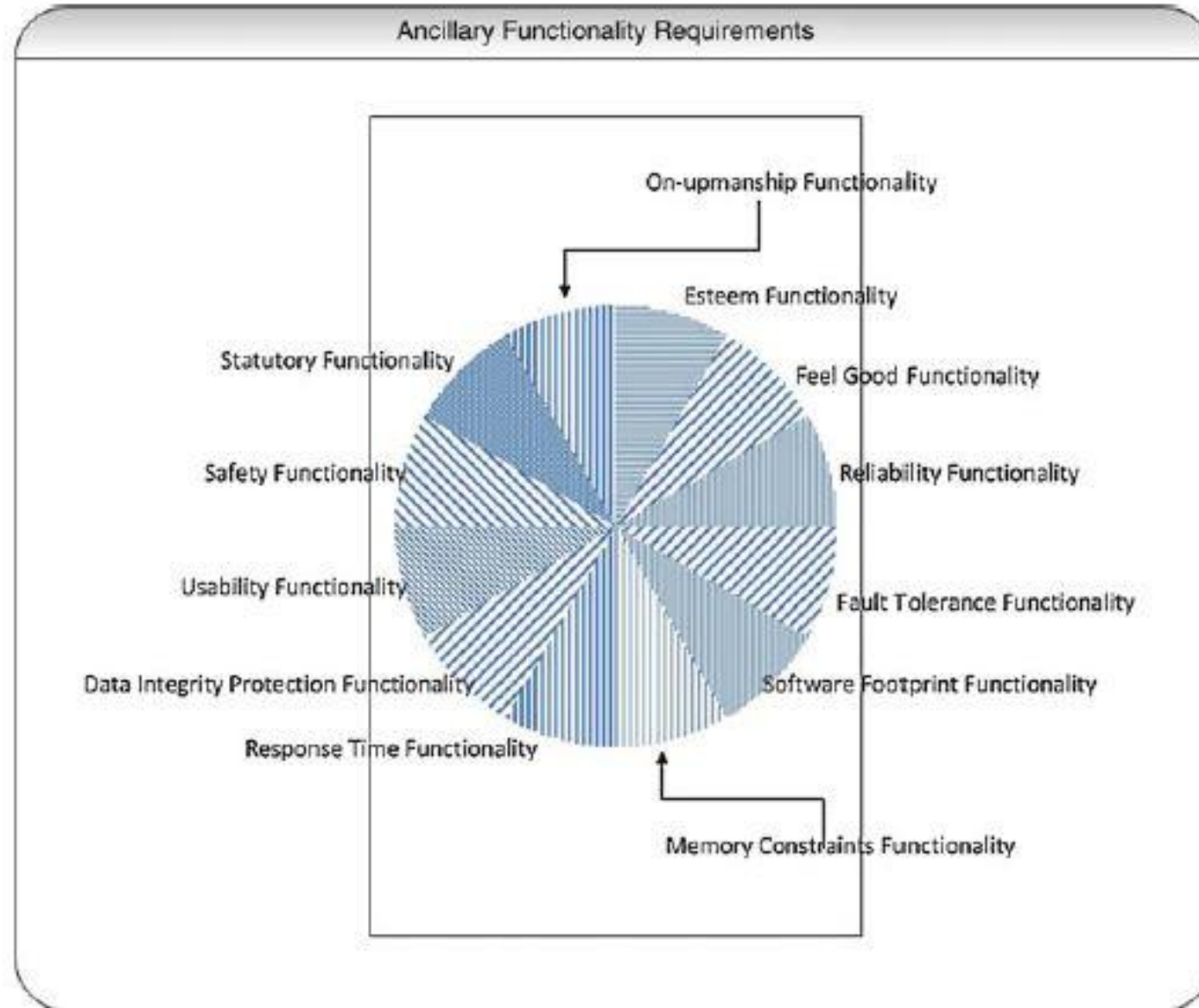
işlevsellik açısından iki ana sınıfa ayrılabilir;

1. Çekirdek işlevsellik gereksinimleri
2. Yardımcı işlevsellik gereksinimleri

İşlevselliğe göre sınıflandırma

- **Temel işlevsellik gereksinimleri,**
- Yerine getirilmesi gereken işlemler
- Ürün geliştirmede olmazsa olmaz özellikler
- Yerine getirilmemesi ürünü kullanışsız kılar
- Temel işlevsellik, bir dizi iş sürecinin performansını ele alır. Yazılım geliştirmenin temel amacı, bu temel işlevselliği yerine getirmektir.
- **Yardımcı işlevsellik gereksinimleri,**
- Temel işlevselliği tamamlar.
- Yardımcı işlevsellik yerine getirilmemiş olsa bile, ürün hala kullanışlıdır, ancak verimlilik veya güvenlik kaybı şeklinde sakıncaya neden olabilir.
- Müşterinin belirtmeyebileceği noktalar.
- Geliştirme ekibinin sorumluluğundadır.

İşlevsellğe göre sınıflandırma



Yardımcı işlevsellik mantığı

- **Yasal işlevsellik: hükümet düzenlemeleri / standartları** veya endüstri birlikleri veya meslek birlikleri nedeniyle gereklidir. ISO, IEEE vb meslek birlikleri.
- **Emniyet işlevselliği:** maddi kayıpları da içeren zararlar. Verilerin kaybı vb.
- **Güvenlik işlevselliği:** Dış saldırılara karşı emniyet. Davetsiz misafirlere karşı koruma. İçerideki kişilerin kötü niyetli kullanımı ve İnternet üzerinden veri hırsızlığı ve benzeri güvenlik işlevlerine örnek olarak verilebilir.
- **Kullanılabilirlik işlevselliği:** Başlangıçta Kullanıcı dostu olarak anıldı. Bu işlevselliğin amacı, yazılımı sezgisel ve kullanım kılavuzlarına en az referansla kullanılabilir hale getirmektir. Grafikselleştirilmiş kullanıcı Arayüzü (GUI) bu işlevselliğin çoğunu başarmıştır, ancak geliştirme her zaman mümkündür.

Yardımcı işlevsellik mantığı

- **Veri Bütünlüğü Koruması işlevi:** Artık bilgisayar EDP (Elektronik Veri İşleme) odalarından son kullanıcılara kaydığı için bu, herhangi bir yazılım ürünü için önemli bir işlevsellik haline geldi. Son kullanıcı istemeden ve masum bir şekilde veri bütünlüğünü etkileyebilir. İsim alanlarına sayı girebilir, sayısal alanlara harf girebilir. İnsan Kaynakları (İK) başvurularında, çalışanın yaşı yasal çalışma yaşının altında veya emeklilik yaşının üzerinde olabileceği bir doğum tarihi girebilirler. Hastane veya otel yönetimi uygulamalarında, check-in tarihinden önce olduğu gibi check-out tarihini girebilirler. Kullanıcılar, yanlış verilerle çeşitli yollardan girebilirler. **Yanlış verilerin girilmesini önlemek için gerekli tüm önlemlerin alınması şarttır.** Böylece veri doğrulama gereklilikleri yardımcı işlevselliğin önemli bir parçası haline gelir.
- **Tepki süresi işlevselliği:** Bazen, özellikle **gerçek zamanlı** uygulamalarda, yanıt süreleri temel işlevselliğin bir parçasını oluşturur. Ancak iş uygulamalarında, yardımcı işlevlerin bir parçasını oluştururlar. Web tabanlı uygulamalarda, kullanıcı ve sunucu tek bir yerde bulunamadığından ve uygulama yeterince hızlı yanıt vermezse, kullanıcı uygulamayı iptal edebilir veya başka bir şey yapabilir. Normalde bunlar kurumsal gelişim standartlarının bir parçasını oluşturur.

Yardımcı işlevsellik mantığı

- **Bellek kısıtlamaları işlevselliği:** Özellikle hafıza kritik sistemlerde, veri yoğunluğu ve veri işleme hacmi yüksek olan yapılarda bellek kısıtları dikkate alınmalıdır.
- **Yazılım ayak izi-kısıtı:** Yazılım küçük el aygıtlarında ve çeşitli makinelerde bir yonga üzerinde bulunduğunda, takılan paketin son boyutu çok önemlidir. Artık tüm Bilgisayar Sayısal Kontrollü (CNC) makinelerde sadece atölyelerde değil evlerde de bu sınırlama var. Günümüzde otomobiller, buzdolapları, çamaşır makineleri, fırınlar, cep telefonları hepsi sınırlı kapasiteye sahip bir çip üzerinde yazılıma sahiptir. Bu, çip üzerine kurulabilecek yazılımın boyutuyla ilgili kısıtlamalar gerektirir. Bu bilgi, yazılımın yükleneceği seçilen yongaları sağlayan donanım üreticilerinden elde edilebilir.

Yardımcı işlevsellik mantığı

- **Hata toleransı işlevselliği:** Kullanıcılar çoğunlukla kasıtsız olarak yazılım kullanırken hata yaparlar. Bu işlem, kullanıcı tarafından bir hata yapıldığında, yazılımın çökmemesini veya iptal edilmemesini sağlar. Bir hata mesajı verir ve alternatif bir yol sunar ve kullanıcının herhangi bir hasara yol açmadan arıza senaryosundan kolayca çıkmasının yanı sıra diğer işlevleri de kullanmasına olanak tanır. MTTF , MTBF, MTTR.
- **Güvenilirlik işlevselliği:** Çevresel değişime bağlı yazılım arızası engellenmeli ve kullanıcılar için uygulamanın sorunsuz bir şekilde değiştirilmesi veya kapatılması için alternatif bir yolun mevcut olması gerekmektedir.

Yardımcı işlevsellik mantığı

- **Hisset-iyi işlevsellik:** Bu işlevsellik, kullanıcı arabiriminin çekici görünmesini sağlar. Fikir, yazılımı kullanırken kullanıcıların kendilerini iyi hissetmelerini sağlamaktır. Bu kullanıcı arayüzü ekranlarına glitz ekliyor ve düğmeler ve simgeler için güzel resimler kullanıyor.
- **Saygı işlevselliği:** Bu işlev, kullanıcılara gurur getirir. Örneğin, bir Rolex saati, diğer saatlerle aynı temel işlevi (zaman göstererek) sunar. Ancak, kasanın daha kalın altın kaplamasının, çizilmeye karşı dayanıklı kristalin ve benzerlerinin kaplanmasını sağlayarak Rolex saatler, sahiplerine daha yüksek bir maliyetle güven ve gurur hissettirir. Yazılımda da bu tür işlevsellik daha iyi mesajlar, hata durumlarının ele alınması, başka hiçbir yazılımın sahip olmadığı işlevler eklenmesi şeklinde sağlanabilir.
- **One-upmanship (rekabet gücü) işlevi:** Bu işlev, yazılım için rekabet avantajı sağlar. Herhangi bir rekabetçi yazılım paketinin sahip olduğundan daha fazla işlevselliğe sahip. Bu işlevsellik çekirdek işlevsellikte veya yardımcı işlevsellikte olabilir. Özellik özelliği ile karşılaştırıldığında, bu paketin rekabetçi yazılım paketleri üzerindeki her özelliğinde bir veya daha fazla işlevsel yönü olacaktır.

Ürün İnşa Etme Koşullarına Dayalı Gereksinimlerin Sınıflandırılması

- İdame/Bakılabilirlik
- Esneklik
- Verim
- Tekrar Kullanılabilirlik
- Taşınabilirlik
- Operasyonların kolaylığı
- Test edilebilirlik
- Arayüz işlevselliği

Ürün İnşa Etme Koşullarına Dayalı Gereksinimlerin Sınıflandırılması

- **İdame/Bakılabilirlik:** Elde edilen ürün korunmalıdır. Diğer bir deyişle, kodu değiştirmek, kod eklemek veya bazı kodları silmek mümkün olmalıdır. Sürdürülebilirliğin kilit yönü sadece eklemek, değiştirmek veya silmek değil, aynı zamanda bu faaliyetleri verimli, etkili ve asgari kaynak harcamasıyla yapmaktır. Sürdürülebilirliğin bir diğer önemli yönü, orijinal geliştiricilerin dışındaki kişilerin ürünü sürdürebilmesidir. Ürün bakımını belirleyen ürün özellikleri bu kategoriye girmektedir. Bunlar normalde ürün için seçilen standartlarda ve yönergelerdedir.
- **Esneklik:** Esneklik, ürünün çok sayıda benzer senaryoda kullanılma yeteneğini ifade eder. Örneğin, bir malzeme yönetim yazılımı ürünü, mühendislik endüstrisi, kimya endüstrisi, seri üretim sistemi ve parti üretim sisteminde faydalı olmalıdır. Bir başka esneklik anlayışı, altta yatan parametrelerden bazıları değiştiğinde, ürünün herhangi bir değişiklik yapmadan yararlı olabilme kabiliyetini ifade eder. Örneğin, bir bordro yazılım ürününde, yeni bir kesinti veya yeni bir ödemenin eklenmesi, ürünü işe yaramaz hale getirmemelidir. Öte yandan, kaynak kodu değiştirilmeden paket hala kullanılabilir olmalıdır. Ürün işleyişinde esneklik sağlamaya odaklanan işlevsellik özelliği bu kategoriye giriyor. Bu özellikler genellikle standartların, rehberlerin ve yardımcı işlevsellik gereksinimlerinin bir parçasıdır.

Ürün İnşa Etme Koşullarına Dayalı Gereksinimlerin Sınıflandırılması

- **Verim:** Elde edilen ürün, kaynakları verimli kullanılmalı. Ürün tarafından kullanılan kaynaklar yalnızca bilgisayar kaynakları (CPU, RAM, disk alanı vb.) değil, aynı zamanda son kullanıcıların zamanı, ağdaki bant genişliği, yedek depolama vb. Ürün, belki de gerekli tuş vuruşlarını ve fare tıklamalarını azaltarak kullanıcılar tarafından harcanması beklenen süreyi en aza indirmelidir. Kaynak kullanımının verimliliğine odaklanan işlevsellik özelliği bu kategoriye giriyor. Standartlar, yönergeler, yardımcı işlevsellik gereksinimleri bu gereksinimleri sağlar.
- **Tekrar Kullanılabilirlik:** İnşa edilmesi gereken ürün, bileşenlerinin diğer ürünlerde de kullanılabileceği şekildedir. Otomobil endüstrisi bu konsepti özenle uygular. Aynı motor birden fazla modelde kullanılıyor. Direksiyon simidi, frenler, lastikler, şaftlar vb. Bileşenler birçok modelde hiçbir değişiklik yapılmadan kullanılır. “Tekerleği yeniden icat etmenin bir anlamı yok” ifadesi oradan kaynaklanmış gibi görünüyor. Yazılım endüstrisinde, yeniden kullanılabilirlik bir kuraldan çok bir istisnadır. Bunlar işlevsel olmayan gereksinimlerdir ve normalde standartlar ve yönergeler veya kuruluş tarafından karşılanmaktadır.

Ürün İnşa Etme Koşullarına Dayalı Gereksinimlerin Sınıflandırılması

- **Taşınabilirlik:** Taşınabilirlik eskiden COBOL gibi aynı dilde geliştirilen yazılımın bir donanım platformundan diğerine kaydırılması olarak adlandırılır. Günümüzde buna ek olarak web uygulamasının bir ana bilgisayardan diğerine kaydırılması yada Cloud computing ile gelecekte uygulamaları bir ana bilgisayardan / veri merkezinden diğerine kaydırmak daha sık gerekenen bir durum olabilecektir.
- **İşlemler kolaylığı:** Modern yazılım ürünleri, belki de uluslararası sınırları kapsayan farklı coğrafi bölgelerden birçok kullanıcıyı içeren büyük ve çok işlevli sistemlerdir. Bu tür sistemleri çalışır durumda tutmak için, işlemleri yürüten uzmanlara ihtiyaçları var. Bu nedenle birçoğunun özel / paylaşılan sistem yöneticilerine, DBA'lara ve ağ yöneticilerine ihtiyacı vardır. Çoğu durumda, sistemleri yıkmadan yazılım veya donanım güncellemesi yapılmalıdır. Bu nedenle, yazılım ürününün tüm bu hususları göz önünde bulundurarak inşa edilmesi gerekir. Bu hususlar normal olarak ürün mimarisi, tasarım ve yapım ile ilgili standartlar ve yönergeler kapsamındadır.

Ürün İnşa Etme Koşullarına Dayalı Gereksinimlerin Sınıflandırılması

- **Testedilebilirlik:** Tabii ki, ürün test edilebilir olmak zorundadır ve öyle olacaktır. O zaman test edilebilirlik konusunda özel olan nedir? Genel olarak, büyük yazılım ürünlerinin% 100 testinin uygulanamayacağı kabul edilir. Bu nedenle, yazılım geliştirme sırasında çeşitli kalite güvence faaliyetleri uygulanmaktadır. Sabitleme kusurlarının maliyeti, kusurun çözüldüğü aşama ile orantılı olarak değişir. Birim testi sırasında açığa çıkan bir kusur, sistem test aşamasında ortaya çıkan bir hasara göre düzeltmek için çok daha düşük maliyetlidir. Nihai ürün her zaman test edilebilirdir ancak bazen test edilmesi zor hale gelen yazılım birimi / bileşenidir. Yazılım ürününün, her yazılım biriminin bağımsız bir şekilde bağımsız bir şekilde test edilebileceği şekilde tasarlanması ve inşa edilmesi gerekir. Test gereksinimleri, normal olarak yazılım mimarisi, tasarım ve yapım kılavuzlarıyla ilgili standartlar ve yönergeler kapsamındadır.
- **Arayüz işlevselliği:** Web tabanlı internet uygulamalarının bu günlerde, arayüz oluşturma önem kazanıyor. İnternetin kendisi birden çok katmanla oluşturulmuştur. Bir İnternet uygulamasının arayüzünü oluşturması gereken birçok tarayıcı ve farklı sunucular, ISS'ler ve ağ protokolleri vardır. Ek olarak, uygulamaların, kuruluşun daha sonra inşa edebileceği uygulamalarla arayüz oluşturma mümkün olacağı şekilde inşa edilmesi gerekir. Bu tür bir işlevsellik normalde yazılım tasarımı ve yapımı ile ilgili standartlar ve yönergeler kapsamında ele alınacaktır.

İhtiyaç Kaynağına Göre İhtiyaçların Sınıflandırılması

- Son Kullanıcılar
- Müşteri organizasyonunun yönetimi
- Etki alanı uzmanları
- Proje takımı
- Tüzükler/Kurallar
- Endüstri Standartları
- Yazılım Tasarımcıları
- Programcılar
- Yazılım Kalite Güvence Ekibi
- Yazılım geliştirme ekibinin yönetimi
- Pazarlama Bölümü

İhtiyaç Kaynağına Göre İhtiyaçların Sınıflandırılması

- **Son kullanıcılar:** Bu insanlar, kendi iş süreçlerini gerçekleştirmek için son ürünü kullananlardır. Yazılım ürünü temel olarak ihtiyaçlarını karşılamayı amaçlamaktadır. Bu insanlar, özellikle girdi düzeyinde, işlem ve çıktıların çalışma seviyesine ilişkin temel işlevselliği sağlarlar. Son kullanıcılar, yazılımdan beklenen yönetim gereksinimlerini sağlayamayabilir. Son kullanıcılar, yazılım geliştirmeyi finanse eden bölümlerde proje senaryosu durumunda (bir kuruluş içinde kullanılması amaçlanan) bulunabilir. COTS (Ticari Rafta) ürün senaryosu durumunda, son kullanıcılar ürün için hedef pazarda dağılır. İhtiyaçlarını almak için pazar araştırmaları yapmamız veya rasgele bazı son kullanıcıları seçmemiz ve onların ihtiyaçlarını ve ürünün temel işlevlerini elde etmek için onlarla görüşmemiz gerekebilir.

İhtiyaç Kaynağına Göre İhtiyaçların Sınıflandırılması

- **Müşteri organizasyonunun yönetimi:** Bu kişiler, temel işlevselliğin MIS (Yönetim Bilgi Sistemi) bölümünü sağlar. Organizasyondan etkin bir şekilde yönetebilmeleri için yazılımdan hangi bilgileri elde etmeleri gerektiğini sağlarlar. Bunlar, yazılım ürününden gerekli olan özel analizleri, özel raporları, denetim izlerini, güvenlik kaygılarını, güvenlik kaygılarını vb. içerebilir. Bir proje senaryosunda, bu insanlar organizasyon şemasına bakarak organizasyonun içinde bulunabilir. Ancak COTS ürün geliştirme durumunda, bu tür uzmanları bulmak için gerçekten çaba sarf etmemiz gerekiyor. Alan uzmanları, akademi ve bilgi sağlamak için üst düzey yönetim personelinin seçen pazar anketleri aracılığıyla bulunabilirler.

İhtiyaç Kaynağına Göre İhtiyaçların Sınıflandırılması

- **Alan uzmanları:** Bu bireyler, önerilen yazılım ürününün geliştirileceği iş alanında uzun yıllar çalışmış olanlardır. Bu bireyler özellikle yararlıdır ve COTS ürün geliştirmesinde yaygın olarak kullanılmaktadır. Bu insanlar IT (Bilgi Teknolojisi) uzmanları olabilir veya olmayabilir, ancak sistemler ve prosedürler veya alan hakkında bilgi sahibi olurlar. Etki alanındaki son kullanıcılar tarafından kullanılan ayrıntılı prosedürleri, biçimleri, şablonları, yönergeleri, standartları ve kontrol listelerini bilen kişilerdir. Ek olarak, girdilerin çıktılara ve alanla ilgili yasal konulara dönüştürülmesinde kullanılan süreçte uzman olacaklardır. Bu insanlar, zaman zaman sektördeki en iyi uygulamalar hakkında veya son kullanıcılar tarafından sağlanan gereksinimlerin eksik veya belirsiz olarak algılandığı durumlarda proje senaryolarında kullanılacaktır. Etki alanı uzmanları uçtan uca çekirdek işlevsellik sağlayabilir veya bunların sorunlarını netleştirebilir.

İhtiyaç Kaynağına Göre İhtiyaçların Sınıflandırılması

- **Proje takımı:**

Proje ekibi proje yöneticileri, proje liderleri, yazılım tasarımcıları, iş analistleri, programcılar, test ediciler, Kullanıcı Arabirimi (UI) geliştiricileri ve Veri Tabanı Yöneticilerinden (DBA'lar) oluşur. Bu kişiler ayrıca, ürünün yazılım geliştirme faaliyetlerinde kullanılması teklif edilmediği sürece, temel işlevsellik sağlayamayacak olmalarına rağmen, gereksinimleri sağlama kaynağıdır. Bununla birlikte, kullanılabilirlik, bakım kolaylığı, güvenlik ve güvenilirlik vb. gibi yardımcı işlevler sağlayabilecektir.

İhtiyaç Kaynağına Göre İhtiyaçların Sınıflandırılması

- **Tüzükler:** Tüzükler, yalnızca yazılımın kullanılabilirliği ile ilgili değil aynı zamanda olası yasa dışı faaliyetlerle ilgili devlet düzenlemelerini de içerir. Amaçlanan yazılım ürünü, hiçbir şekilde cezai işlem yapmaz, bunlara yardım etmez veya bunlara uymaz. Bu nedenle, gereksinim duyulan tüm fonksiyonların önerilen yazılım ürününün dışında tutulduğundan emin olmanın yanı sıra uygulanması gereken tüm statüleri içermesi gerekir. Gereksinim analizi yapan iş analistleri, işlevselliğin yanı sıra yasaklanmış işlevselliğin dahil edilmesini zorunlu kılan kanunların farkında olmalıdır. Yasaklanan faaliyetlere örnek olarak, kişisel verilerin çalınması, hareketsiz banka hesaplarındaki paralardan uzak durulması, spam e-posta gönderilmesi vb.

İhtiyaç Kaynağına Göre İhtiyaçların Sınıflandırılması

- **Endüstri standartları:** Bunlar, endüstri standartlarını (örneğin, ISO, CMMI, NASSCOM (Hindistan Ulusal Yazılım ve Hizmet Şirketleri)) veya meslek kuruluşlarını (IEEE, SPIN'ler, SPMN'ler gibi) veya satıcı kuruluşun veya müşteri organizasyonun organizasyonel standartlarını içerir. Yazılım mühendisliği metodolojilerinin prosesler, kurallar, formatlar, şablonlar ve kontrol listeleri dahil olmak üzere çeşitli yönleri). Bir dizi bilgi ve en iyi uygulamalar bu standartlardan edinilebilir ve yardımcı işlevsellik bu standartlardan elde edilebilir.

İhtiyaç Kaynağına Göre İhtiyaçların Sınıflandırılması

- **Yazılım tasarımcıları:** Yazılım tasarımcıları, nihai üründe verimlilik, hata toleransı, işlem kolaylığı, kurulum kolaylığı, kullanılabilirlik, yapısal stabilite ve benzeri konularda yardımcı gereksinimler sağlayabilir. Yazılım tasarımcıları normalde proje ekibinin bir parçasıdır, ancak yazılım tasarımcıları son üründe önemli bir rol oynadığından ayrı olarak ele alınır. Son olarak, yazılım geliştirmenin son ürünündeki eksik/kusurlu işlevsellik sorumluluğunu yerine getirmek zorunda olan yazılım tasarımcılarıdır.
- **Yazılım programcıları:** Yazılım programcıları, yazılımdaki gereksinimleri uygulamak için zincirin sonundadır. Bununla birlikte, yazılım tasarım belgelerini kullanarak kendilerine iletilen tüm gereklilikleri yerine getirmek isteyen insanlardır. Ayrıca, kodlama kuralları, kullanıcı arabirimi kuralları ve diğer herhangi bir organizasyon kuralları tasarım belgelerinin bir parçasını oluşturmaz ve programcıların bu standartlar hakkında bilgi sahibi olmaları ve kodladıkları her programda etkin bir şekilde uygulamalarını beklemeleri beklenir. Yazılım programcıları, yönetilebilirlik, test edilebilirlik, kodun tekrar kullanılabilirliği vb. ile ilgili yardımcı işlevsellikler belirleyebilirler.

İhtiyaç Kaynağına Göre İhtiyaçların Sınıflandırılması

- **Yazılım kalite güvence ekibi:** Bir Yazılım kalite güvence ekibi, gözden geçirenleri, test edenleri ve süreç uzmanlarını içerir. Bu kişiler, test edilebilirlik ve önerilen yazılım ürününün kalite perspektifleri hakkında yardımcı gereksinimler sağlayabilir.
- **Yazılım geliştirme ekibinin yönetimi:** Bu kişiler, proje yöneticisi, proje lideri ve program yöneticileri dahil olmak üzere diğer üst düzey yönetim personelinin içerir. Bu bireyler, genel projeyi kuşbakışı ile görebilecek bir konumda olacak ve yazılım ürününün organizasyondaki diğer uygulamalarla etkin bir şekilde etkileşime girebilmesini sağlamak için arayüz gereklilikleri sağlayabilecektir. Ayrıca önerilen yazılım ürünü için tüm fonksiyonların dahil edilmesini sağlayabilecektir.

İhtiyaç Kaynağına Göre İhtiyaçların Sınıflandırılması

- **Pazarlama Bölümü:** Özellikle ürün geliştirme organizasyonlarında pazarlama, ürün gereksinimlerinin / şartnamelerinin bir kaynağıdır. Bir Pazarlama departmanı saha personelinden ürün gereksinimleri ya da önerilen ürünün potansiyel kullanıcılarının pazar araştırması yapabilir. Pazar anketleri, kullanıcı gereksinimlerini toplamak ve ürün spesifikasyonlarını belirlemek için çok popüler bir araçtır. Bir Pazarlama departmanı, bir COTS ürününde tek işçilik işlevselliğinin birincil kaynağıdır.

Gereksinimlerin Gelişimi

- Gereksinimler **tek bir fikir** olarak başlar ve belli bir zaman diliminde, daha sonra bir yazılım ürününde daha fazla işlenebilen tam bir **küme**ye dönüşür. Bu bölümde gereksinimlerin gelişimindeki aşamalar tartışılmaktadır.
- Lütfen her kuruluşun bu aşamaları kullanmasının zorunlu olmadığını unutmayın; bazı fazlar **düşürülebilir** veya bazı başka fazlar kullanılabilir; veya tamamen farklı fazlar kullanabilirler!

Gereksinimlerin Gelişimi

- **Yeni bir üründe - pazarda bulunmayan** - Bu, piyasada bulunmayan türün tamamen yeni bir ürünün tasarlandığı bir senaryodur. Bu senaryoda, gereksinimler aşağıdaki gibi geliştirilmiştir:
 - Fikir Çimlenmesi
 - Beyin Fırtınası
 - Pazar/müşteri/danışman anketleri
 - Kişisel görüşmeler
 - Prototip ve demolar
 - Gereksinimlerin sonlandırılması (dondurulması)

Yeni bir üründe - pazarda bulunmayan

- **1. Fikir çimlenmesi** : Burada girişimci veya ürün yöneticisi, hedef pazarın gereksinimlerine yönelik gözlemlerini temel alan bir fikri filizlendirir ve hedef müşterilerin karşılanmayan ihtiyaçlarını karşılayabilecek bir ürüne olan ihtiyacı algılar. Büyük bir organizasyonda, birkaç ürün yöneticisi olabilir ve hepsi yeni ürün fikirleri bulabilir. Bir ürün fikrinin onaylanması ve inşa edilmesi için yaklaşık sekiz ciddi fikir gerektiği belirtildi. Onaylanan fikirlerin tümü başarılı değildir ve atılan fikirlerin tümü kötü fikir değildir. Chester Carlson, Xerox makineleri konusundaki fikri için onlara yaklaştığında, IBM ve GE gibi kuruluşlar tarafından 5 yıl boyunca geri çevrildiğini hatırlıyor musunuz? Bu onaylanmış fikir, türünün pazarda bulunmadığı yeni bir ürün geliştirme gereksinimlerinin gelişimindeki ilk aşamadır.
- **2. Beyin fırtınası**: Beyin fırtınası, eldeki konudaki uzmanların gayri resmi bir toplantıda bir araya geldiği ve hayal güçlerini serbest bıraktığı bir tekniktir. İfade edilen tüm fikirler daha sonra analiz edilmek üzere kaydedilecektir; İhtiyaç yönetiminde, istenen tüm ürün özellikleri beyin fırtınası tarafından numaralandırılır, bunlar daha sonra elenir ve uygulanabilir olanlar ayrılır. Bunlar, ürün için başlangıç gereksinimlerini oluşturur.

Yeni bir üründe - pazarda bulunmayan

- **3. Pazar/müşteri/danışman anketleri** : İlk gereksinimler bir pazar araştırması kullanılarak test edilir. Çeşitli pazar araştırması yöntemleri mevcuttur ve uygun olanı seçilerek kullanılır. Piyasa araştırması başlangıçtaki gereklilikleri doğrulayacak ve genellikle birkaç gereklilik ekleyecektir. Bu gereklilikler, kişisel görüşmeler kullanılarak piyasadan, danışmanlardan veya akademi'den gelen uzmanlar tarafından da doğrulanabilir.
- **4. Kişisel görüşmeler:** Pazarlamacılar, ürün tasarımcıları, destek personeli, tüketiciler, danışmanlar veya akademisyenler olabilecek seçilmiş uzmanlarla kişisel görüşmeler yapılır. Pazar anketleri tarafından sonuçlandırılmış olan gereksinimler onlarla iki amaç için tartışılacaktır. Bir — gereksinimleri doğrulamak için; iki — gereksinimlere ek yapabilmek için. Bu, bir prototip tasarlama girişiminde bulunmadan ve ürünü doğrulamak için tekrar piyasaya çıkmadan önceki son adımdır.

Yeni bir üründe - pazarda bulunmayan

- **5. Prototip ve demolar:** Gereksinimler kişisel görüşmelerle doğrulandıktan sonra normalde ürünün bir prototipi oluşturulacaktır. Şimdi bu prototipler potansiyel müşterilere ve bu alandaki uzmanlara gösterilmektedir. Geri bildirimleri alınır, değerlendirilir ve gereksinimler güncellenir. Bu gereksinimlerin gelişiminde son adımdır.
- **6. Gereklilikleri dondur:** Gereklilikleri dondurmak, kuruluşların standartlarına uyan gereksinimleri belgelemeyi ve yapılandırma kontrolü ve değişiklik yönetimi titizliğinde daha fazla değişiklik yapılmasını içerir. Donmuş gereksinimler daha sonra tam ölçekli ürün tasarımı ve ürünün geliştirilmesi ve pazara sunulması için kullanılır. Normalde donmuş şartlarda önerilen değişiklikler, ürünün bir sonraki yükseltilmesi / bırakılması için göz önünde bulundurulur.

Gereksinimlerin Gelişimi-Pazarda var olan

- **Fikir çimlenmesi:** Girişimci veya ürün yöneticisi veya söz sahibi biri, bir ürünü mevcut bir ürün için **rakip olarak geliştirme** fikrini alır. Mevcut ürün piyasa beklentilerini karşılamayabilir veya yeni ve yenilikçi bir ürünü alabilecek büyüklükte bir pazar olabilir veya böyle bir sebep fikrin arkasında olabilir. Bu ön şarttır.
- **Pazar / müşteri / danışman anketleri:** Ek bir ürüne duyulan ihtiyacı doğrulamak ve pazarda mevcut olan ürünler tarafından karşılanmayan ihtiyaçları ortaya çıkarmak için pazar araştırmaları yapılır.
- **Kişisel görüşmeler:** Alanındaki uzmanlarla kişisel görüşmeler, pazar anketlerinden elde edilen verileri doğrulamak ve listeye daha fazla gereksinim eklemek için yapılır.

Gereksinimlerin Gelişimi-Pazarda var olan

- **Beyin Fırtınası:** Beyin fırtınasında, mevcut ürünle ilgili bir (üstünlük sağlama çabası/laf ebeliği!) fikirleri üretilir. Daha fazla işlevsellik, daha iyi sunum, daha iyi iş akışı, gelişmiş kullanım kolaylığı, daha fazla kullanıcı seçeneği, esneklik vb. ile ilgili fikirler beyin fırtınası sırasında üretilir. Bunlar analiz edilir ve gereksinimler kesinleşir.
- **Prototipler :** Bu senaryoda normal olarak prototipler oluşturulmaz, çünkü çalışan bir ürün pazarda bulunur. Ancak, bazen bir kavram veya özellik ispatlamak ve piyasadan geri bildirim almak için kullanılabilir.
- **Gereklilikleri dondur:** Önceden belirtildiği gibi dondurma gereklilikleri, gereksinimler belgesini onaylamayı ve onu yapılandırma kontrolü ve değişiklik yönetimi titizliğine tabi tutmayı içerir.

Gereksinimlerin Gelişimi- Ürün Yükseltme

- Bir süredir pazarda olan bir ürünümüz var ve müşterilerimizden, saha destek ekibinden, pazarlamacılardan ve Katma Değerli Bayilerden (VAR) mevcut özelliklerdeki istenen ek özellikler veya iyileştirmeler hakkında geri bildirim alıyoruz. Ayrıca, rakiplerin ürünlerini de pazar payımızı azaltan pazara getirdiklerini görüyoruz. Bu nedenle, pazarda rekabetçi ve çekici olmasını sağlamak için ürünümüzü yükseltmek istiyoruz. Gereksinimlerin nasıl geliştiği:
- **VARs'tan Geribildirim / Anketler** : VARs, piyasadaki rakiplere ve son kullanıcılara yakın olmasından dolayı mevcut ürün hakkında değerli bir geri bildirim kaynağıdır. Ne zaman geri bildirim sağarlarsa analiz etmeli ve çözmeliyiz. Ürünü yükseltmeyi düşünürken, ürünümüz için arzu edilen iyileştirmeler hakkındaki görüşlerini ortaya çıkarmak için, halihazırda ilettiklerinin üzerinde ve üstünde bir anket yapmamız gerekir. VARs'ın geri bildiriminden ve anketinden elde edilen bilgilerin, çoğaltmayı ve uygulamanın uygulanabilirliğini gidermek için analiz edilmesi gerekir.
- **Teknik destek personelinde geri bildirim / anketler**: Destek personeli sahadadır ve müşteriler / son kullanıcılar ile ilk temas halindedir. Müşterilerden iyileştirilmesi için sorunlar, endişeler ve öneriler alacaklardır. Çoğu ürün organizasyonu, müşterileriyle etkileşimleri sırasında bu tür bilgileri yakalamak için resmi bir mekanizmaya sahip olacaktır. Bir ürün geliştirmeyi düşündüğümüzde, saha destek yöneticileri tarafından kazanılan tüm uzmanlığı kapsamlı bir şekilde yakalamak için bu tür tüm geri bildirimleri toplamamız ve saha personelinde daha fazla öneride bulunmak için resmi bir anket yapmamız gerekir. Geri bildirim ve anket sonuçları, yükseltme gereksinimlerini tamamlamak için analiz edilebilir.

Gereksinimlerin Gelişimi- Ürün Yükseltme

- **Müşteri / pazar anketleri** : Müşteriler, tedarik zincirinde ilk elden geri bildirim sağlayabilecek tek kişilerdir. Diğerleri yalnızca ikinci el geribildirimde bulunabilir. Bu nedenle müşteri anketleri yapmamız gerekiyor. Bunlar VAR'lardan ve saha destek personelinden elde edilen geri bildirimleri doğrular.
- **Kişisel görüşmeler** : Yapılan çeşitli anketlerin bulgularını doğrulamak ve anket sonuçlarımıza katlanan önyargıları veya önyargıları ortaya çıkarmak için kişisel görüşmeler yapmamız gerekir. Tabakalı örnekleme tekniğini kullanarak örnekleme bazında kişisel görüşmeler yapıyoruz.
- **İhtiyaçların dondurulması** :VARS, saha destek personeli ve müşterilerden geri bildirim / anket sonuçları topladığımızda, gereklilikleri birleştirmek ve kopyaları elemek için sonuçları analiz ediyoruz. Ürün yükseltmesi için gereklilikleri uygulanabilirliklerine göre seçiyoruz. Ardından örgütsel standartlara uygun ve dokümanı yapılandırma ve değişiklik yönetimi titizliğine maruz bırakan gereksinimleri belgeleriz.