

Bölüm 2

MUHAMMET FARUK YAŞAR

MERT KABA

ÖKKEŞCAN KÜÇÜKÖNDER

İBRAHİM YETİŞGEN

2. ÖĞRETİM A ŞUBESİ 2.GRUP

Hazırlık için Gereksinimlerin Ortaya Çıkarılması

Ürün Misyon Beyanı

Yeni bir sistemin geliştirilmesini veya eski bir sistemin yeniden tasarlanmasını üstlenirken yapmamız gereken ilk şey, ne yapması gerektiğine dair kısa bir tanım elde etmek veya geliştirmektir. Böyle bir ifadeye genellikle ürün misyonu beyanı (veya sistem misyon beyanı) denir. Bazı kuruluşlar, daha uzun olma eğiliminde olsa da, misyon beyanına benzeyen bir operasyon kavramı (Conops) beyanına atıfta bulunur. Bazı ortamlarda Conops, çok yüksek seviyeli bir gereksinim belirtimine benzeyen bir belgedir.

Ürün misyon beyanı veya Conops, sisteme dahil olan herkes için bir odak noktası görevi görür ve "bu işlevsellik amacına nasıl hizmet ediyor?" Sorusunu sorarak çeşitli özelliklerin önemini tartmamıza izin verir. Çevik metodolojilerde, daha sonra tartışılacağı üzere, misyon ifadesinin veya Conops'un "sistem metaforu" rolünü oynadığını söyleyebiliriz.

Örneğin, Ek A'daki Bölüm 1.1, Smith Akıllı Ev için bir görev bildirimini içerir ve Ek B'deki Bölüm 1, atık su pompalama kontrol sistemi için bir Conops bildirimini içerir.

Misyon ifadeleri yazmak çekişmeli bir iş olabilir ve birçok insan küçük ayrıntılarda çıkmaza girme eğilimi olabileceğinden bunu yapmaktan çekinir veya korkar. Bazen, ürün misyon beyanları çok uzun sürebilir ve aslında bir Conops belgesine dönüşebilir. Bir ürün misyonu ifadesi çok kısa, açıklayıcı, ilgi çekici ve asla ayrıntılı olmamalıdır.

En çok alıntı yapılan "iyi" sistem görev açıklamalarından biri, orijinal Star Trek televizyon dizisinden Starship Enterprise ile bağlantılı olanıdır.

Kabaca ifade edilen bu misyon ifadesi:

garip yeni dünyalar keşfetmek, yeni yaşam biçimleri ve yeni medeniyetler aramak, daha önce kimsenin gitmediği yerlere cesurca gitmek.

Bu ifade açık, ikna edici ve ilham vericidir. Ve bu "faydalı" - bu klasik dizinin hayranları, yıldız gemisi ekibi tarafından yapılacak belirli eylemlerin sistem görevi beyanına karşı tartıldığı birkaç bölümü hatırlayacak.

Peki, bagaj taşıma için bir ürün veya sistem misyonu ifadesi ne olabilir? sistem benziyor? ne dersin

Yolcu kalkış noktasından varış noktasına kadar bagaj taşımanın tüm yönlerini otomatikleştirmek.

Evcil hayvan mağazası POS sistemi için şunları göz önünde bulundurun:

Müşteri satın alma etkileşimi ve envanter kontrolünün tüm yönlerini otomatikleştirmek.

Bunlar mutlaka zekice ya da hayranlık uyandırıcı değil ama sistemin özünü aktarıyorlar. Ve daha sonra, spesifikasyonuna dahil olanların beklentilerini kalibre etmemiz gerektiğinde faydalı olabilirler. Özellikle küresel olarak dağıtılmış kalkınmada, bir sistem metaforuna duyulan ihtiyaç büyük önem taşımaktadır.

Bir Müşteriyle Karşılaşma

Karınızın (ya da "koca", "arkadaş", "oda arkadaşı" veya herhangi birinin yerine) pasta yapmak istediği için mağazaya gidip aşağıdaki öğeleri almanızı istediğini varsayalım:

- 5 kilo un
- 12 büyük yumurta
- 5 kilo şeker
- 1 kilo tereyağı

En yakın markete (evinize yakın olan) gidin.

Dükkanında, onun beyaz mı yoksa esmer şeker mi istediğinden emin olmadığınızı fark ediyorsunuz. Yani onu cep telefonunuzdan arayıp hangi şeker istediğini soruyorsunuz; esmer şekere ihtiyacı olduğunu öğrenirsin. Alışverişinizi yapıp eve dönüyorsunuz.

Ama karınız seçimlerinizden memnun değil. Yanlış türde un satın aldınız; beyaz istediğini ve senin buğday aldığını bildiriyor. Yanlış türde tereyağı aldınız; tuzsuz istedi. Sen de yanlış türde şeker getirdin, koyu kahverengi; açık kahverengi istedi. Şimdi başın belada.

Yani markete geri dönüyorsun ve un, şeker ve tereyağını geri veriyorsun. Beyaz unu ve esmer şekerini buluyorsunuz, ancak bir küvette yalnızca tuzsuz tereyağı (çubuk değil) bulabilirsiniz, ancak bir küvetin onun için kabul edilebilir olduğunu varsayıyorsunuz. Satın alma işleminizi gerçekleştiriyor ve ürünlerle birlikte iade ediyorsunuz. Ama şimdi yeni hatalar yaptığınızı keşfediyorsunuz. Açık kahverengi şeker alımı iyi, ancak geri getirdiğiniz beyaz un ağırlanmış; ağırlanmamış istedi. Ve küvetteki tereyağı kabul edilemez - tuzsuz tereyağının çubuk şeklinde bulunabileceğine dikkat çekiyor. Cahilliğinizden dolayı şimdi size çok kızgın.

Böylece, mağazaya geri dönersiniz ve eşyaları mahcup bir şekilde iade eder ve uygun ikamelerini alırsınız. Karınızın öfkesini yatıştırmak için, onun en sevdiği çikolatalı şekerlerden de almaya karar veriyorsunuz.

Eve dönüyorsun ve o hala mutsuz. Sonunda tereyağı, şeker ve unu hazırlarken, karınız akşam yemeği için omlet yaptığını ve omlet ve kek için bir düzine yumurtanın yeterli olmayacağını hatırlıyor - 18 yumurtaya ihtiyacı var. Ayrıca çikolatadan da memnun değil - diyetle olduğunu ve etrafta çikolatanın cazibesine ihtiyacı olmadığını söylüyor.

Bir kez daha marketi ziyaret edip çikolatayı ve bir düzine yumurtayı iade edin. 18 yumurta alıp eve dönüyorsun.

"Bu şeyleri nereden aldın?" diye sorduğunda alışverişi doğru yaptığınızı düşünüyorsunuz. Eşyaları marketten aldığınızı fark ettiğinizde, kıpkırmızı kesildi - orada fiyatların çok yüksek olduğunu düşünüyor - yolun birkaç mil ilerisindeki süpermarkete gitmeliydiniz.

Bu örnekle devam edebiliriz - karınız her satın alımlarınızda yeni bir kusur keşfettiğinde, miktar veya marka hakkında fikrini değiştirdiğinde, yeni ürünler eklediğinde, diğerlerini çıkardığında, vb.

Ancak bu durumun gereksinim mühendisliği ve paydaşlarla ne ilgisi var? Durum, gereksinim mühendisliği hakkında birçok noktayı göstermektedir. İlk olarak, uygulama alanını anlamamız gerekir. Bu durumda, fırıncılık bilgisine sahip olmak, tereyağı, un ve şekerin farklı türleri olduğu konusunda sizi önceden bilgilendirirdi ve muhtemelen alışveriş yolculuğunuza başlamadan önce odak soruları sormuş olurdunuz. Bu senaryodan bir başka nokta da—müşteriler her zaman ne istediklerini bilmezler— siz mağazaya üç kez gidene kadar karınız daha fazla yumurtaya ihtiyacı olduğunu fark etmemiştir. Ve hikayede bir ders daha var: Müşterilerin ne istediği konusunda asla varsayımlarda bulunmayın - küvet yağının kabul edilebilir olduğunu düşündünüz; değildi. Sonunda, müşterilere istediklerinden fazlasını (bu durumda en sevdiği çikolatayı) sağlamanın bile bazen yanlış bir şey olabileceğini öğrendiniz.

Ancak daha geniş anlamda, bir müşteriyle bu karşılaşmadan çıkarılacak en önemli ders, onların sorun çıkarabileceğidir. Ne istediklerini her zaman bilemezler ve bildiklerinde bile isteklerini etkisiz bir şekilde iletebilirler. Müşteriler fikirlerini değiştirebilir ve bildikleriniz ve ne sağlayacağımız konusunda yüksek beklentileri olabilir.

Paydaş etkileşimi çok önemli olduğu için, bu bölümün geri kalanını paydaşların ve özellikle de sistemin kendileri için inşa edildiği paydaşların, yani müşterilerin doğasını belirlemeye ve anlamaya adanacağız.

Sistem Sınırlarının Belirlenmesi

Paydaşları belirlemede gerekli ilk adım, üst düzey bir sistem modeli oluşturmaktır. Bu model, sistemle ilgili kişi ve kuruluşlar kümesini tanımlayacak ve sistem sınırını (diğer varlıklarla doğrudan ve dolaylı etkileşimleri) tanımlayacaktır (Laplanche ve diğerleri, 2016). Sistem sınırlarını doğru bir şekilde tanımlamadan, önemli paydaşların gözden kaçırılması mümkündür - potansiyel olarak feci bir durum.

Bağlam Diyagramları

Bağlam diyagramı, sistem sınırlarını ve diğer sistemler, insanlar ve çevre ile etkileşimleri gösteren ilgili bir sistem için görsel bir temsil diyagramıdır. Bağlam diyagramı iki nedenden dolayı önemlidir. İlk olarak, proje yaşam döngüsü boyunca mühendisler ve paydaşlar için sistem hakkında diyalog için bir temel sağlar. İkincisi, bağlam diyagramı, kapsam kaymasıyla, yani işlevsel gereksinimlerin sistemin amacının ötesinde kontrolsüz büyümesiyle mücadeleye yardımcı olur. Bağlam diyagramı, belirsiz, karmaşık ve eksik gereksinimler tartışılırken gereksinim anlaşılmasına ve analizine de yardımcı olabilir.

Bağlam diyagramı birçok biçim alabilir. Bir blok diyagram yeterli olabilir, örneğin, Şekil B.1, ıslak kuyu pompalama sisteminin bağlamını gösteren çok basit bir blok diyagramdır. Kullanım durumu diyagramı gibi UML/SysML diyagram ailesinden biri de kullanılabilir. UML ve SysML, Bölüm 4 ve Ek C (UML) ve E'de (UML modellerinden biri olan kullanım senaryoları) tartışılmaktadır. Kısaca açıklanacak olan yumuşak sistemler metodolojisinden (SSM) zengin resimler de bağlam diyagramları için kullanılabilir.

Paydaşlar

Paydaşlar, söz konusu sistemin başarısında (veya başarısızlığında) bir miktar çıkarları (payları) olan bireyler grubunu temsil eder. Herhangi bir sistem için, hem açık hem de üstün birçok paydaş türü vardır. Bir sistemin en belirgin paydaşı kullanıcıdır.

Kullanıcıyı, sistemi kullanacak olan sınıf (bir veya birden fazla kişiden oluşan) olarak tanımlıyoruz. Müşteri, sistemin yapımını devreye alan (bir veya daha fazla kişiden oluşan) sınıftır. Bazen müşteriye müşteri (genellikle yazılım sistemleri söz konusu olduğunda) veya sponsor (işletmede) denir.

sistemin satış için değil, dahili kullanım için inşa edildiği durum). Ancak birçok durumda "müşteri", "müşteri" ve "sponsor" terimleri bağlama bağlı olarak birbirlerinin yerine kullanılmaktadır. Sponsor ve müşterinin aynı kişi olabileceğini unutmayın. Ve çoğu zaman müşterinin kim olduğu ve sponsorun kim olduğu arasında birçok soruna yol açabilecek bir karışıklık vardır.

Her durumda, müşteriler, müşteriler, kullanıcılar, sponsorlar - bu terimleri yeniden tanımlamak isterseniz - hepsi paydaştır çünkü sistemde kazanılmış bir çıkarları vardır. Ancak bunlardan daha fazla paydaş var. "Müşteri her zaman haklıdır, ancak sistemle ilgilenen daha fazla kişi/kuruluş vardır" diyor. Aslında, herhangi bir yeni sistemde hissesi olan birçok kişi var. Örneğin, herhangi bir sistem için tipik paydaşlar şunları içerebilir:

Müşteriler (müşteriler, kullanıcılar)

Müşterilerin müşterileri (üçüncü kişi tarafından kullanılacak bir sistem olması durumunda partiler)

Sponsorlar (sistemi devreye alan ve/veya ödeyecek olanlar)

Tüm sorumlu mühendislik ve teknik kişiler (örn. sistemler, geliştirme, test, bakım)

Düzenleyiciler (tipik olarak çeşitli düzeylerdeki devlet kurumları)

Sistemde çıkarı olan ancak doğrudan düzenleyicisi olmayan üçüncü şahıslar yetkili (örneğin, standart kuruluşları, kullanıcı grupları)

Toplum (sistem güvenli mi?)

Çevre (fiziksel sistemler için)

Ve elbette, bu eksik bir liste. Örneğin, teslim edilen sistemin üçüncü bir tarafça genişletildiği, tekrar büyütüldüğü, dördüncü bir tarafa teslim edildiği ve bunun gibi müşterilerin müşteri devrim edebiliriz. Bu gözetim zincirinin de önemli yasal sonuçları vardır - bir sistem başarısız olduğunda, başarısızlıktan kim sorumlu veya sorumlu? Her halükarda, "paydaş" terimini kullandığımızda, sadece müşterinin değil, başkalarının da işin içinde olduğunu hatırlamamız gerekir.

Negatif Paydaşlar

Negatif paydaşlar ise sistemden olumsuz etkilenebilecek olanlardır. Bunlara rakipler, yatırımcılar ve işleri sistem tarafından değiştirilecek, olumsuz etkilenecek veya yerinden edilecek kişiler dahildir. Ayrıca dahili olumsuz paydaşlar da vardır—

daha fazla iş yükü alacak diğer departmanlar, kıskanç rakipler, şüpheli yöneticiler ve daha fazlası. Bu iç olumsuz paydaşlar, pasif-agresif direniş sağlayabilir ve ilgili herkes için siyasi kabuslar yaratabilir. Tüm olumsuz paydaşlar mümkün olduğunca tanınmalı ve hesaba katılmalıdır.

Son olarak, her zaman sistemlerden doğrudan etkilenmeyen, yine de bu sistemlerle ilgilenen veya bu sistemlere karşı olan bireyler vardır ve bir miktar güç veya etkiye sahip olabilecekleri için, bunlar dikkate alınmalıdır. Bunlar ilgilenen

partiler arasında çevreciler, hayvan aktivistleri, tek sorunlu fanatikler, her türden savunucular, kendi çıkarlarını düşünenler vb. yer alır. Bazı insanlar bu tür bireyleri "at sinekleri" olarak adlandırır ve bunlar göz ardı edilmemelidir.

Paydaş Tanımlama

Herhangi bir sistem için olası tüm paydaşları (olumlu ve olumsuz) doğru ve eksiksiz olarak belirlemek çok önemlidir. Boehm (2003), paydaş tanımlamasına yardımcı olmak için CRACK kısaltmasını türetti. Yani paydaş, projeye kendini adanmış, bir grup paydaş temsil eden, karar almaya yetkili, ekip üyeleriyle işbirliği yapacak ve bilgili kişidir.

Paydaş belirleme, misyon beyanı yazıldıktan ve sistem sınırları belirlendikten sonra gereksinim mühendisinin atması gereken bir sonraki adımdır. Paydaş belirleme neden bu kadar önemlidir? Önemli bir paydaş dışarıda bırakıp daha sonra keşfetmeyi hayal edin. Ya da daha kötüsü, ilgi duydukları bir sistemin inşa edildiğini ve görmezden geldiğini keşfederler. Bu gecikmiş paydaşlar, sisteme çok maliyetli olabilecek her türlü kısıtlama ve gereksinim değişikliğini dayatmaya çalışabilirler.

Paydaş Soruları

Paydaşları belirlemeye yardımcı olmanın bir yolu aşağıdaki soruları yanıtlamaktır:

Sistemin parasını kim ödüyor?

Sistemi kim kullanacak?

Sistemin kullanıma uygunluğunu kim değerlendirecek?

Sistemin herhangi bir yönünü hangi kurumlar (hükümet) ve kuruluşlar (hükümet dışı) düzenler?

Sistemin yapımını, dağıtımını ve işletimini hangi yasalar yönetiyor?

Sistemin spesifikasyonu, tasarımı, yapımı, testi, bakımı ve kullanımdan kaldırılmasıyla ilgili herhangi bir konuda kimler yer alıyor?

Sistem kurulursa kimler olumsuz etkilenecek?

Bu sistemin var olup olmaması kimin umurunda?

Kimi dışarıda bıraktık?

Havayolu bagaj taşıma sistemiyle ilgili bu soru setini deneyelim. Bu cevaplar tam olmayabilir - zamanla yeni paydaşlar ortaya çıkabilir. Ancak bu soruları şimdi elimizden geldiğince eksiksiz yanıtlayarak, çok önemli bir paydaş sürecin sonunda gözden kaçırma olasılığını azaltmış oluyoruz.

Sistemi kim ödüyor?—havayolu, hibeler, yolcular, vergi dolarlarınız.

Sistemi kim kullanacak?—havayolu personeli, bakım personeli, yolcular (sonda).

Sistemin kullanıma uygunluđuna kim karar verecek?—havayolu, müşteriler, sendikalar, FAA, OSHA, basın, bağımsız derecelendirme kuruluşları.

Hangi kurumlar (hükümet) ve kuruluşlar (hükümet dışı) sistemin herhangi bir yönünü düzenler?—FAA, OSHA, sendika sözleşmeleri, eyalet ve yerel yasalar.

Sistemin yapısını, dağıtımını ve işletimini hangi yasalar yönetir?—çeşitli eyalet ve yerel yapı yönetmelikleri, bagaj taşıma sistemleri için federal yönetmelikler, OSHA yasaları.

Sistemin spesifikasyonu, tasarımı, yapımı, testi, bakımı ve kullanımdan kaldırılmasıyla ilgili herhangi bir konuda kimler yer alıyor?—çeşitli mühendisler, teknisyenler, bagaj taşıma görevlileri sendikası vb.

Sistem kurulursa kimler olumsuz etkilenecek?—yolcular, sendika personel.

Bu sistemin var olup olmaması kimin umurunda?—limuzin sürücüler.

Kimi dışarıda bıraktık?

Ve bu soru setini evcil hayvan mağazası POS sistemi üzerinde deneyelim.

Sistemi kim ödüyor?—evcil hayvan mağazası, tüketiciler.

Sistemi kim kullanacak?—kasiyerler, yöneticiler, müşteriler (belki self servis sağlanırsa).

Başka kim?

Sistemin kullanıma uygunluđuna kim karar verecek?—şirket yöneticileri, yöneticiler, kasiyerler, müşteriler. Başka kim?

Hangi kurumlar (hükümet) ve kuruluşlar (hükümet dışı) sistemin herhangi bir yönünü düzenler?—vergi makamları, ticari kuruluşları yöneten, evcil hayvan mağazası organizasyonları, daha iyi ticaret bürosu. Başka ne?

Sistemin yapısını, dağıtımını ve işletimini hangi yasalar yönetir?—vergi yasaları, iş ve ticaret yasaları. Başka ne?

Sistemin spesifikasyonu, tasarımı, yapımı, testi, bakımı ve kullanımdan kaldırılmasıyla ilgili herhangi bir konuda kimler yer alıyor?—çeşitli mühendisler, CFO, yöneticiler, kasiyerler. Hepsini bilmemiz gerekiyor.

Sistem kurulursa kimler olumsuz etkilenecek?—manuel yazarkasacılar, envanter memurları. Başka kim?

Bu sistemin var olup olmaması kimin umurunda?—rakipler, evcil hayvan ürünleri satıcıları.

Başka kim?

Kimi dışarıda bıraktık?

Zengin Resimler

Bazı sistemlerde paydaşların tamamı, tanımlayıcı paydaş sorularıyla bile kolayca belirlenemez. Bu durum bazen, paydaşların tanımlanmasını bilgilendirmek için kullanım geçmişinin veya etki analizinin olmadığı yeni sistemlerde söz konusudur. Bu durumlarda daha bütüncül sistem tabanlı bir yaklaşım uygun olabilir. Böyle bir yaklaşım SSM'dir (Checkland ve Scholes 1999).

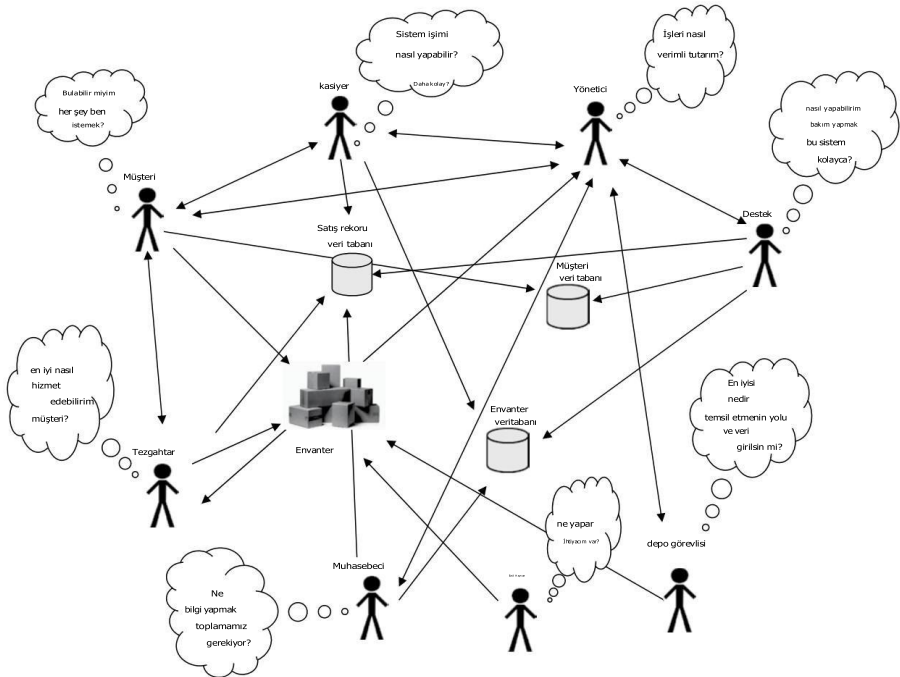
SSM, çeşitli kullanıcıları hedefleri, istekleri ve ihtiyaçları ile birlikte gösteren "zengin resim" adı verilen karikatür benzeri bir çizime sahiptir. Zengin resimler, açıklamalı kullanım durumu diyagramlarına veya kavram haritalarına benzer, ancak bunlar oldukça gayri resmidir. Zengin resimler paydaş belirlemede faydalı olabilir.

Evcil hayvan mağazası POS sistemi için ilk deneme zengin resmi Şekil 2.1'de gösterilmektedir.

Şekil, evcil hayvan mağazası için çeşitli paydaşları ve birincil endişeyi göstermektedir.

Belirgin paydaşlar arasında müşteriler, kasiyerler, yöneticiler, bilgisayar bakım ve destek personeli (sistemi yükseltmek ve düzeltmek için), depo personeli (envanter verilerini girmek için), muhasebeciler (vergi bilgilerini girmek ve kullanmak için) ve satış görevlileri (fiyatları girmek için) yer alır. ve indirim bilgileri). Başka pek çok, daha az belirgin olan paydaşlar da var.

Son olarak, kendi adlarına konuşmasalar da evcil hayvanlar zengin resme dahil edilmiştir. Evcil hayvan mağazasının doğru ürünleri stokladığından ve tüm yasal gerekliliklerin karşılandığından emin olmak için istek ve ihtiyaçları çok önemlidir. Örneğin, evcil yılanlar için canlı yemlerin saklanması ve balık ve kuş gibi hayvanların satılmasına ilişkin belirli kurallar ve yasalar vardır. Bu düzenleyici gereklilikler, konserve ve kuru gıda, tımar malzemeleri ve köpekler ve kediler için oyuncak satışına ilişkin olanlardan çok farklıdır. Bu nedenle, hizmet verilecek evcil hayvan setinin belirlenmesi, evcil hayvan mağazası POS sisteminde önemli bir husus olacaktır. Genel olarak, herhangi bir sistemi tasarlarlarken sessiz paydaşları göz önünde bulundurmanız gerekir.



ve bu paydaşların ihtiyaçlarını belirlemek için bir mekanizmaya sahip olmak isteyeceksiniz. Örneğin, POS sistemindeki evcil hayvanlar söz konusu olduğunda, evcil hayvan sahipleri veya hayvan refahı savunucuları, gereksinimlerin ortaya çıkarılması amacıyla vekil olabilir.

Gereksinimlerin ortaya çıkarılmasından önce bir veya daha fazla zengin resim geliştirmek, yinelemeli bir süreçtir. Her yinelemede, eksik paydaşlar veya hatalar belirlenebilir. Bu gözden geçirme ve gözden geçirme süreci, tüm paydaşların doğru bir şekilde tanımlandığı konusunda anlaşma sağlanana kadar devam etmelidir.

Paydaş/Kullanıcı Sınıfları

Paydaş (kullanıcı dahil) grupları belirlendikten sonra, onların ihtiyaç ve isteklerini yeterince ele almak için bu grupları sınıflara ayırmak gerekli olabilir. Örneğin, POS sisteminde, paydaş "evcil hayvanlarının" sistem gereksinimlerini etkileyebilecek farklı ürün ihtiyaçları olan çeşitli hayvan türlerinden oluşabileceğini zaten görmüştük.

Sınıflar büyük ve/veya heterojen olduğunda bir paydaş/kullanıcı sınıfı alt bölümü genellikle gereklidir. Çoğu durumda, sistem kullanıcılarının toplanması için sınıf alt bölümlerine ihtiyaç duyulacaktır. Örneğin, kullanıcıların aşağıdakileri içerdiği bagaj taşıma sistemini düşünün:

Sistem bakım personeli (yükseltme ve düzeltme yapacak)
Bagaj görevlileri (sistemi açık kapatarak etkileşime giren, hızı artırma/azaltma vb.)
Havayolu planlayıcıları/göndericileri (bagaj talep alanlarına uçuşları atayan)
Havalimanı personeli (uçuşları farklı bagaj teslim alanlarına atayan)
Havalimanı yöneticileri ve politika yapıcılar

Evcil hayvan mağazası POS sistemi için kullanıcı sınıfları şunları içerir:

Kasiyerler
Yöneticiler
Sistem bakım personeli (yükseltme ve düzeltme yapmak için)
Mağaza müşterileri
Envanter/depo personeli (envanter verilerini girmek için)
Muhasebeciler (vergi bilgilerini girmek için)
Satış departmanı (fiyat ve indirim bilgilerini girmek için)

Bu sınıfların ve alt sınıfların her biri için, gereksinimlerin ortaya çıkarılması sırasında etkileşimde bulunmak üzere grubun bir şampiyonu veya temsili bir örneğini seçmemiz gerekir.

Bir yaklaşım, grup için tek bir temsilci seçmek olabilir. Böyle bir yaklaşım, sınıf nispeten küçük ve kompozisyon açısından tek tip olduğunda iyi sonuç verir. Örneğin, evcil hayvan mağazası POS sistemi için bir muhasebeci, evcil hayvan mağazası zinciri için çalışan üç kişilik grubu temsil edebilir.

Başka bir strateji, sınıfın küçük bir alt kümesini seçmek olacaktır. Böyle bir yaklaşım, sınıf büyük olduğunda geçerlidir, ancak sınıfı temsil etmesi için tek bir kişiye güvenmek istemiyoruz. Örneğin, evcil hayvan mağazası POS sisteminde, örneğin 10 müşteriden oluşan bir örneğin 5.000'den fazla müşterinin koleksiyonunu temsil etmesini isteyebiliriz.

Son olarak, paydaş sınıfı küçük ama çeşitli olduğunda pratik olabilecek tüm üyeliği ele alabiliriz. Evcil hayvan mağazası POS örneğiyle devam edersek, evcil hayvan mağazası zinciri beş sahip tarafından özel olarak tutuluyorsa, gereksinimleri ortaya çıkarmak için böyle bir yaklaşım mantıklı olabilir - muhtemelen her bir sahibinden gereksinimleri ortaya çıkarmak isteriz.

Her bir paydaş grubundan nasıl girdi arayacağımızı seçtikten sonra, kullanılacak uygun bilgi toplama tekniğini/tekniklerini seçebiliriz. Gereksinim belirleme teknikleri Bölüm 3'te tartışılacaktır.

Kullanıcı Özellikleri

Bir gereksinim mühendisi, bireylerin, hatta aynı kullanıcı sınıfındakilerin bile, dikkate alınması gereken farklı özelliklere sahip olabileceğinin farkında olmalıdır. Örneğin bagaj taşıma sisteminde yaşlı yolcuların ihtiyaçları, çocuklardan veya yetişkinlerden farklıdır. Belirli engelleri olan kişilerin farklı ihtiyaçları olacaktır. İngilizce bilmeyenlerin başka ihtiyaçları olacaktır. Çoğu durumda, kullanıcı sınıflarını bu özel ihtiyaçlara ve özelliklere göre alt sınıflara bölmek mantıklıdır.

Ayrıca, bu alt sınıflardaki bireyler veya gruplarla iletişim kurmak, muhtemelen farklı tekniklerin kullanılmasını ve büyük olasılıkla güçlendirilmiş empatiyi (örneğin, çocuklar veya yaşlılar söz konusu olduğunda) gerektirir. Newell et al. (2006), bu zorluklardan bazılarını ve bunların çözümlerine ilişkin yararlı bir genel bakış sağlar.

Farklı ülkelerden paydaşlarla etkileşimde bulunurken kültürel özelliklerdeki farklılıklar da göz ardı edilemez. Sosyolog Geert Hofstede (2001) tarafından yapılan ufuk açıcı çalışma, ülkeler arasındaki kültürel farklılıkların algılanabileceği beş boyut olduğunu bulmuştur: otorite ile başa çıkmada rahatlık (güç mesafesi), bireycilik, erkeksi bir dünya görüşüne eğilim (erkeklik endeksi) , belirsizlikten kaçınma ve uzun vadeli yönelim. Bu farklılıklar gereksinim mühendisi için önemlidir. Örneğin, güç mesafesinin yüksek olduğu bir ülkeden bir kişi, bir üstünü utandırabilecek önemli bir konuyu gündeme getirmek konusunda isteksiz olabilir. Veya erkeklik endeksi yüksek bir ülkeden bir kadın, erkeklerin bulunduğu bir toplantıda konuşmaktan korkabilir.

Dolayısıyla bu durumlarda farklı etkileşim tekniklerine ihtiyaç duyulabilir. Bu konulara ilişkin eğlenceli bir tartışma Laplante'de (2010) bulunabilir.

Son olarak, çıkar çatışması veya "kurum sorunlarının" farkında olun, çünkü bunlar paydaşlarla iletişimi önleyebilir ve belirsizleştirebilir. Yani, bir paydaş sınıfının üyesi, başka bir paydaş sınıfının üyesi olabilir. Örneğin evcil hayvan mağazası POS sistemi için bir kasiyer de müşteri olabilir. Bu nedenle, bir işlemin en iyi nasıl yürütüleceği konusunda bu kişiyle iletişim kurarken gereksinim mühendisi

bir müşterinin veya bir kasiyerin çıkarlarının daha iyi temsil edilip edilmediğini bilemez. Gereksinim mühendisi, bu olası önyargıların farkında olmalı ve paydaşlarla her türlü iletişimde bunları kontrol etmelidir.

Müşteri İstek ve İhtiyaçları

Gereksinim mühendisinin birincil amacının müşterilerin ne istediğini anlamak olduğundan bahsetmiştik. Ancak bu istekleri veya arzuları (Latince'den "arzulanan şeyler" için) keşfetmek zordur. Ayrıca, gereksinim mühendisi olarak müşteriye neye ihtiyacı olduğunu önermeniz gerektiğini düşünebilirsiniz, ancak kendi değer sisteminizi bir başkasınının yerine koymakla ilgili nasihati unutmayın - müşterinin ihtiyaç duyduğu şey, onların ihtiyaç duymadığı bir şey olabilir. İstemek (bir müşteriyle karşılaşmayı hatırlayın). Bununla birlikte, ihtiyaç duydukları ancak dikkate almadıkları özellikler olup olmadığını belirleyebilmeleri için müşterilere yeni işlevler göstermek her zaman yararlıdır. Bu durum, özellikle müşterinin gereksinim mühendisinden daha az alan bilgisine sahip olduğu durumlarda geçerlidir. Bununla birlikte, bir noktada, istekler ve ihtiyaçları uzlaştırmanız gerekecek - o zaman gereksinimleri önceliklendirilmesi çok faydalı olacaktır. İstenenleri belirlemek için paydaş katılımına ilişkin çok iyi bir tartışma Hull ve diğerleri tarafından verilmektedir. (2011).

Müşteriler Ne İstiyor?

Gereksinim mühendisi, müşteri istek ve ihtiyaçlarını karşılamaya çalışır, ancak bunların ne olduğunu bilmek her zaman kolay değildir. Niye ya? Müşterilerin istek ve ihtiyaçları pek çok düzeyde var olduğundan - pratik (örneğin, sistemin minimum işlevselliği), rekabetçi (X markasından daha iyi olmalıdır), bencil (göstermek ve sistemin özelliklerini öne çıkarmak istiyorlar) ve daha fazlası . Ve bazen müşteriler "hepsini" isterler ve fazla ödeme yapmak istemezler. Bu nedenle gereksinim mühendisleri, kurulacak sistem için gerçekçi hedefler belirlemelerinde müşterilere yardımcı olmalıdır.

Müşterilerin ihtiyaç düzeylerini anlamamanın bir yolu, Maslow'un klasik kendini gerçekleştirme hiyerarşisini yeniden gözden geçirmektir (Şekil 2.2).

Maslow, insanların en temelden (piramidin alt kısmından) başlayarak ihtiyaçlarını karşılamaya (gerçekleştirmeye) ve en zor, ezoterik ve manevi olana doğru yol almaya çalışacaklarını kuramlaştırdı. Ancak, önce daha düşük ihtiyaçlar/istekler tatmin edilmezse, asla piramidin çok yukarısına çıkamazlar (veya daha yüksek bir seviyede kalamazlar).

Temel ihtiyaçlar yemek, su ve uyuktur. Bununla birlikte, bunlar kişinin fiziksel güvenliğinden önce gelir - eğer açlıktan ölüyorsanız, diğer tarafta yiyecek almak için çok yoğun bir caddeyi geçme riskiyle karşı karşıya kalırsınız. İnsanlar ekme çalarak hapse girme riskini alıyor. Maslow'un piramidinin yukarısında, sevilme ve bir gruba ait olma ihtiyacı vardır, ancak bu ihtiyaçların temelde fiziksel güvenlik ihtiyacına tabi olduğunu varsayar. Bunun hakkında tartışabilirsiniz, ancak düşünce şu ki, bazı insanlar fiziksel sağlıklarını korumak için aşk şansını feda edecekler.



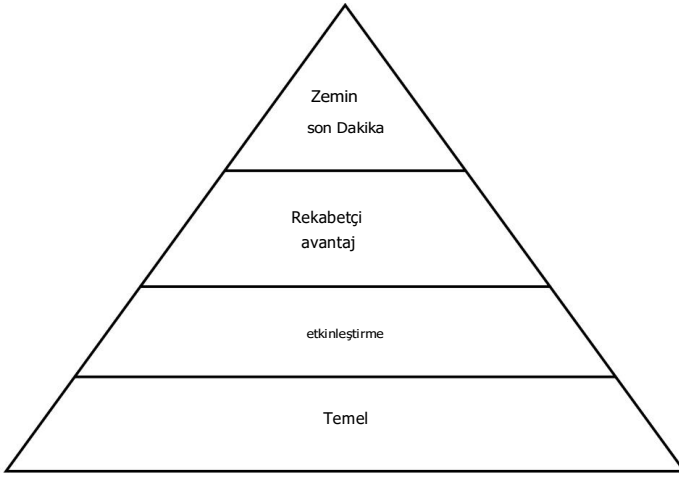
Şekil 2.2 Maslow'un hiyerarşisi.

(Sırf üyelerinden birini sevdiğiniz için Sky Diving Club'a üye olmaya devam eder miydiniz?). Daha sonra, kişinin özgüveni önemlidir, ancak ait olma ve sevilme ihtiyacı kadar önemli değildir (bu yüzden kendinizi küçük düşürecek ve çılgın baldızınızın düşünü için bir Roma kostümü giyeceksiniz). Son olarak, Maslow kendini gerçekleştirmeyi "insanın gerçekleştirme arzusu, yani potansiyel olarak ne ise o olma eğilimi: kişinin olabileceği her şey olma eğilimi..." olarak tanımladı (Maslow 1943).

Şekil 2.3'te gösterilen Maslow hiyerarşisinin bir varyasyonu, müşterilerin ihtiyaç ve isteklerini açıklamaya yardımcı olabilir.

Burada en düşük seviye temel işlevselliştir. Bir satış noktası sistemi olmak, belirli işlevlerin mevcut olması gerektiği anlamına gelir - örneğin bir satış oluşturma, bir öğeyi iade etme, envanteri güncelleme vb. Etkinleştirme düzeyinde, müşteri, kuruluş içindeki diğer sistemlere (yazılım, donanım veya süreç) göre etkinleştirme yetenekleri sağlayan özellikler ister. Böylece POS sistemi, satış verilerinin yöneticiler tarafından tahmin veya envanter kontrolü amacıyla izlenmesine izin veren bazı yönetim yazılımlarına bağlanır. Etkinleştirme düzeyindeki işlevsellik, rakiplerin yeteneklerini karşılamayabilir veya bunları aşmayabilir. Bu işlevsel ihtiyaçlar, rekabet avantajı düzeyinde karşılanır. Burada müşteri, bu yeni sistemin rekabeti aşan veya başka bir şekilde bir iş avantajı yaratan yetenekler sağlamlasını istiyor.

Son olarak, çığır açan arzular, mevcut teori veya pratiği aşan ve söz konusu sistemin ötesinde çıkarımları ve uygulamaları olan teknolojinin gelişimini ima edecektir. Örneğin, mevcut teknolojileri aşan bir tür yeni veri madenciliği teknolojisi istenebilir. Maslow hiyerarşisinde olduğu gibi, fikir, daha yüksek işlevsellik seviyelerini karşılamak için daha düşük seviyeli işlevselliğin feda edilmemesi gerektirir.



Şekil 2.3 Müşteri ihtiyaçlarının/isteklerinin hiyerarşisi.

Bu hiyerarşi, ihtiyacın dört önem düzeyi anlamına gelse de, herhangi bir durumda daha fazla veya daha az düzey olması muhtemeldir. Ancak temel fikir, daha sonra gereksinimlerin önceliklendirilmesine en çok yardımcı olacak anlamlı bir hiyerarşiye göre müşterilerin ihtiyaçlarını keşfetmek ve organize etmektir.

Bu, Maslow'un teorisinin müşterilerin veya kullanıcıların ihtiyaçlarını aydınlatmak için ilk kez kullanılması değil. Örneğin, Valacich ve ark. (2007), Web tabanlı kullanıcı arayüzlerinde kullanıcı tercihlerini tanımlamak için değiştirilmiş dört seviyeli Maslow hiyerarşisini tanıttı. Hunter (2015), nesnelerin internetindeki gereksinim türlerini açıklamak için hiyerarşinin bir versiyonunu da kullanmıştır (bkz. Vignette 2.1).

Her durumda, bazı istekleri değerlendirmek için çalışan örneklerimize dönelim. bagaj taşıma ve evcil hayvan mağazası POS sistemleri için.

Bir havayolu bagaj taşıma sistemi için müşteriler muhtemelen

Güvenlik
Hız
Güvenilirlik
Hata toleransı (kırık bagaj yok!)
Bakım kolaylığı
ve benzeri

Evcil hayvan mağazası POS sistemi için müşteriler,

Hız
Doğruluk
Netlik (basılı makbuzda)

Verimlilik

Kullanım kolaylığı (özellikle self servis sağlanıyorsa)
ve daha fazlası

Bu nedenle, bu ihtiyaçlara katılmak için elimizden gelenin en iyisini yapardık. Ancak sorun şu ki, bu ihtiyaçların memnuniyetini nasıl ölçeceğiz? Çünkü bu istek ve arzular ölçülemiyorsa, onlara ulaşıp ulaşmadığımızı asla bilemeyiz. Ölçülebilir gereksinimler konusunu Bölüm 5'te tartışacağız.

Müşteriler Ne İstemiyor?

Bazen müşteriler, sistemin yapmasını istemedikleri şeyler konusunda çok açıktır. Bu belirli istenmeyen özellikler veya "istemiyorum" veya "yapmayacak" gereksinimleri, gereksinim mühendisi tarafından sıklıkla gözden kaçırılır. İstenmeyen özellikler şunları içerebilir:

İstenmeyen performans özellikleri
Estetik özellikler

Altın kaplama (aşırı ve gereksiz özellikler)
Güvenlik endişeleri (tehlikeler)

"Yapmamalı" gereksinimleri genellikle yakalanması en zor olanlardır. Bazen müşteriler, görene kadar ne istemediklerini bilemezler. Örneğin, teslim edilen sistemi (veya bir prototipi) gördüklerinde şöyle haykırırlar:

"Bunu yapmak istediğimi söylediğimi biliyorum ama sanırım gerçekten öyle demek istemedim."

Açıklayıcı amaçlar için, burada, uygulamanın istenmeyen özelliklerine bazı örnekler verilmiştir. bagaj taşıma sistemi:

Havaalanı ana gücünün kesilmesi durumunda sistem kapanmaz.
Sistem, ana havayolu bilgisayar sisteminde arızaya neden olmayacaktır.
Sistem, bagajın dakikada bir torbadan daha yüksek oranda imha edilmesine neden olmamalıdır.

Sistemin yapmaması gereken şeyi tanımlamanın ne kadar zor olduğunu görebilirsiniz. İstenmeyen davranış konusunu daha sonra inceleyeceğiz. Bu arada, konuyu açıklamak için, evcil hayvan mağazası POS sistemi için bazı "yapılmaması gereken" gereksinimler şunlardır:

Kayıt bandı biterse sistem çökmez.
Ürün kodu bulunamazsa sistem çökmez.
Stok mutabakat kodunda bir sorun bulunursa, cari işlem işlem iptal edilmez.

Bu "yapmamalı" gereklilikler genellikle olumlu olarak "yapmalı" gereksinimler olarak yeniden ifade edilebilir. Bu konu Bölüm 4'te daha ayrıntılı olarak tartışılmaktadır.

Bir sonraki bölümde tartışacağımız ortaya çıkarma tekniklerinin çoğu, istenmeyen özellikleri ortaya çıkarma eğiliminde olacaktır, ancak gereksinim mühendisi her zaman müşterinin ne istemediğini, müşterinin ne istediği kadar güçlü bir şekilde keşfetmeye çalışmalıdır.

Müşteriler Neden Fikirlerini Değiştirir?

Paydaşlarla, özellikle de müşterilerle ilgilenmenin en büyük zorluklarından biri, bazen sistemin ne yapmasını istediklerini tam olarak bilmemeleridir. Bu neden böyle? Birincisi, hiç kimse her şeye kadir değildir. Bir müşteri, başka bir sistemin tam bir kopyasını istemedikçe, olası her isteği göremez. Müşterinin bu yeni özellikleri olabildiğince erken bulmasına yardımcı olmak, gereksinim mühendisinin işidir.

Müşterilerin fikirlerini değiştirmelerinin bir başka nedeni de, değişen düzeylerde gereksinimlerle (örn., özellikler, kısıtlamalar, iş kuralları, kalite özellikleri, vb.), sistem yaşam döngüsü boyunca bu gereksinim olmayanlar değiştikçe önemli olanın değişebileceğidir.

Bazen sistemin çalıştığı ve müşterilerin çalıştığı ortam değişir (fiziksel değişiklikler, ekonomik değişiklikler, rekabet, düzenleyici değişiklikler vb.). Örneğin, evcil hayvan mağazası POS sistemini kurarken, evcil hayvan oyuncaklarıyla ilgili eyalet vergilendirme kuralları değişebilir (belki sadece gıcırıtılı oyuncaklar vergilendirilir) ve bu olay, kodun vergilendirme bölümünün önemli ölçüde yeniden tasarlanmasını gerektirebilir.

Bazı gereksinimler o kadar açıktır ki, müşteri bunları şart koşmayı düşünmez. Örneğin, evcil hayvan mağazası POS sistemindeki müşterilerin listesini hangi sırayla yazdırıyorsunuz? Müşterinin bariz yanıtı soyadına göre alfabetiktir, ancak sistem tasarımcısı müşteri kimliğine göre sayısal olarak düşündü.

Değişen gereksinimlerin bir başka nedeni de yatırım getirisinin keşfedilmesiyle ilgilidir. Örneğin bir projede, müşterinin en önemli gereklilik olarak gördüğü şeyin (kağıt işlerinin azaltılması), gerekliliği uygulama maliyetine göre işletmeye önemli bir ekonomik değer katmadığı keşfedildi. İkincil bir gereksinimin (ortadan kaldırılması gereken kağıt işinin önemsiz olmaması gerekiyordu) temel ekonomik itici güç olduğu ortaya çıktı.

Önceliklendirme ve geç değişikliklerden kaçınma, yatırım getirisi verilerinin gereksinimlerin belirlenmesinde özellikle yararlı olmasının bir nedenidir.

Bazen müşteri tutarsız veya kararsız olabilir. Fikirlerini değiştirirler çünkü bu şekilde çalışırlar. Ya da sadece bir gereksinimin ne olduğunu bilmiyorlar (onları eğitmeniz gerekiyor). Diğer müşteriler basit, şey, aptal. İhtiyaç mühendisleri olarak, bu değişebilirliğin belirli bir miktarını tolere etmeniz gerekir (ne kadarı size kalmış).

Son olarak, müşteriler çeşitli nedenlerle bilgileri kasten saklayacaklardır (örneğin, bilgi özeldir, size güvenmezler, sizden hoşlanmazlar, anlayacağınızı düşünmezler). Saklanan bilgiler projede daha sonra çirkin yüzünü gösterebilir ve sistemde maliyetli değişiklikler gerektirebilir.

Paydaş Önceliklendirme

Şimdiye kadar, çoğunlukla müşteriye birincil paydaş olarak atıfta bulunduk, ancak elbette başkaları da var. Tüm paydaşlar eşit öneme sahip değildir. Örneğin, Bagaj İşleyicileri Birliği'nin endişeleri önemlidir, ancak bagaj taşıma sistemi için ödeme yapan havalimanı yetkilisi (müşteri) kadar önemli olmayabilir. Öte yandan, federal düzenlemeler müşterinin isteklerinin önüne geçer; örneğin, sistem geçerli tüm federal standartlara uygun olmalıdır.

Birçok paydaşımız olduğundan ve bazı ihtiyaç ve istekleri çatışabileceğinden, bu durumları çözmeye yardımcı olmak için paydaş sınıflarını sıralıyor veya önceliklendiriyoruz. Sıralama genellikle paydaşı tatmin edememe riskini ifade eder (örneğin, hapse girmek istemiyorsanız yasal gereklilikler 1 numara olmalıdır). Paydaşları sıralamak, uzlaşma ve risk azaltmanın anahtarı olan gereksinimlerin önceliklendirilmesine yol açacaktır.

Tablo 2.1, basit bir yüksek, orta ve düşük öncelik şemasında sıralanmış bagaj taşıma sistemi için paydaşların kısmi bir listesini içerir. Rütbe ataması için bir gerekçe dahildir.

Tablo 2.2, 1'in en yüksek önemi veya önceliği temsil ettiği 1'den 6'ya kadar olan derecelendirmeler kullanılarak sıralanan evcil hayvan mağazası POS sistemi için kısmi bir paydaş listesi içerir.

Bu sıralama ve önceliklerle kesinlikle tartışabilirsiniz; örneğin mağaza müşterisinin POS sistemindeki en önemli kişi olduğunu düşünebilirsiniz.

Ancak bu anlaşmazlık önemli bir noktayı vurgular: Paydaş çatışmaları ve önceliklendirme hakkında tartışmak istediğinizde gereksinim mühendisliği sürecinin başlarındadır, geri alınması gereken tasarım kararları daha sonra verilmiş olabilir.

Tablo 2.1 Bagaj Taşıma Sistemi Paydaşlarının Kısmi Sıralaması

Paydaş Sınıfı	Rütbe	Gerekçe
Sistem bakım personeli	Orta Sistemle	orta düzeyde etkileşime sahiptirler.
bagaj görevlileri	Orta Sistemle	düzenli etkileşimleri vardır ancak müşteriye ters düşebilecek bir gündemleri vardır.
Havayolu zamanlayıcıları/ sevk görevlileri	Düşük	Sistemle çok az etkileşimleri vardır.
Havaalanı personeli	Düşük	Diğer havaalanı personelinin çoğu sistemle çok az etkileşime sahiptir.
Havaalanı yöneticileri ve politika yapıcılar ("müşteri")	Yüksek	Sisteme para ödüyorlar.

Tablo 2.2 Pet Store POS Sistemi için Paydaşların Kısmi Sıralaması

Paydaş Sınıfı	Rütbe	Gerekeçe
kasiyerler	2	Onlarla en fazla etkileşime sahipler. sistem.
Yöneticiler	1	Onlar birincil müşteri/sponsordur.
Sistem bakım personeli	4	Bir şeyleri düzeltmek zorunda olduklarında kırmak.
Mağaza müşterileri	3	Olumsuz etkilenme olasılığı en yüksek olanlardır.
Envanter/depo personeli	6	Sistemle en az doğrudan etkileşime sahiptirler.
Muhasebeciler/satış personeli	5	Raporları okudular.

Müşteriler ve Diğer Paydaşlarla İletişim

Gereksinim mühendisinin en önemli faaliyetlerinden biri müşterilerle ve bazen de diğer paydaşlarla iletişim kurmaktır. Çoğu durumda, satış personelinin yanı sıra gereksinim mühendisi, işin müşteriye dönük tarafıdır. Bu nedenle, tüm iletişimlerin açık, etik, tutarlı ve zamanında yapılması esastır.

Soru ortaya çıkıyor, müşterilerle iletişim için en iyi format nedir? İletişim kurmanın birçok yolu vardır ve her birinin belirli avantajları ve dezavantajları vardır. Örneğin, yüz yüze toplantılar çok etkilidir. Sözlü bilgiler kullanılan dil aracılığıyla aktarılır, ancak ses kalitesi, ton ve çekimden ve beden dilinden daha ince ipuçları da iletilebilir. Aslında, çevik yazılım metodolojileri (7. Bölümde ele alınmıştır) sürekli yüz yüze iletişimi kolaylaştırmak için her zaman sahada bir müşteri temsilcisinin bulunmasını savunur.

Ancak yüz yüze toplantılar ekonomik değildir ve çok fazla zaman alır. Ayrıca, birden fazla müşteriniz olduğunda ve coğrafi olarak dağılmış müşterileriniz olduğunda, onlarla nasıl tanışıyorsunuz? Bir arada? Ayrı ayrı? Telekonferans yoluyla mı? Tüm bu sorular dikkate alınmalıdır.

İyi planlanmış, yoğun grup toplantıları gereksinim mühendisliği için etkili bir iletişim şekli olabilir ve bu tür teknikleri bir sonraki bölümde tartışacağız. Ancak bu toplantılar pahalı ve zaman alıcıdır ve müşterinin işini bozabilir.

Teşvik sırasında ve sonrasında müşterilere periyodik durum raporları sağlamak, bu sorunlardan bazılarının önlenmesine yardımcı olabilir. En azından yasal bir bakış açısından, gereksinim mühendisi ne bildiğini ve ne bilmediğini tam olarak ifşa etmektedir

Müşteriyle yapılan yazılı iletişimler yasal sözleşmeler ve muhtıralar şeklinde mi olmalı? Resmi sözleşmelerin (veya değişiklik talebi bildirimlerinin) avantajı, bu tür bir iletişimin anlaşmazlıkları önleyebilmesi veya en azından bir anlaşmazlık durumunda kanıt sunabilmesidir. Sonuçta, müşteri ile herhangi bir iletişim mahkemede ilgili olabilir. Ancak resmi iletişimler kişisel değildir, gereksinim mühendisliği sürecini önemli ölçüde yavaşlatabilir ve maliyetli olabilir (özellikle işin içinde bir avukat varsa).

Gereksinim mühendisliği süreci boyunca iletişim kurmak için telefon veya telekonferans aramaları kullanılabilir. Bu modun kayıt dışılığı ve hızı oldukça arzu edilir. Ancak telekonferansla bile, aynı anda yer alan iletişimin bazı nüansları kaybolur ve her zaman yanlış anlama, bırakılan aramalar ve kesinti sorunları vardır. Ve telefon görüşmesinin kayıt dışılığı da bir sorumluluktur—

bir müşteriyle her iletişimin potansiyel yasal sonuçları vardır, ancak her aramayı kaydetmek genellikle elverişsizdir.

E-posta bir iletişim aracı olarak etkili olabilir ve avantajları ve dezavantajları yazılı notlar ile telefon görüşmeleri arasında bir yerdedir.

E-posta hem kendiliğinden hem de gayri resmidir, ancak kalıcıdır; her e-posta işlemi kaydedebilirsiniz. Ancak telefon görüşmelerinde olduğu gibi, kişiler arası nüanslı iletişimin bir kısmı kaybolur ve yasal bir belge olarak e-posta izleri, resmi değişiklik talebi bildirimlerinden daha az ikna edicidir.

Son olarak, wiki teknolojisi, gereksinim bilgilerini müşteriler ve diğer paydaşlarla iletmek için kullanılabilir. Wiki, fikirlerin paylaşılabilmesi ve geliştirilebileceği bir tür beyaz tahta işlevi görebilir. Ayrıca, bazı düzenlemelerle wiki, nihai yazılım gereksinimleri belirtim belgesine dönüştürülebilir. Ayrıca, FitNesse kabul testi çerçevesini kullanarak yürütülebilir test senaryolarını SRS'nin kendisine yerleştirmenin yolları vardır. Bu konular Bölüm 8'de daha ayrıntılı olarak incelenmiştir.

Beklentileri Yönetmek

Müşterilerle başarılı iletişimin anahtarı, beklentileri yönetmektir.

Sadece gereksinim mühendisliğinde değil, tüm çabalarda beklentiler gerçekten önemlidir.

Bu gerçeğe inanmıyorsanız, aşağıdaki durumları göz önünde bulundurun.

Durum A: Danışman olarak bir iş yapmak için sözleşme yaptığınızı ve iş için 5.000 dolarlık bir ücreti kabul ettiğinizi düşünün. İş tamamlarsınız ve müşteri memnun kalır. Ancak müşteriniz size 8.000 dolar ödüyor - başarılı bir yıl geçirdi ve serveti paylaşmak istiyor. Nasıl hissediyorsun?

Durum B: Şimdi saati sıfırlayın—önceki durumun henüz gerçekleşmediğini hayal edin. Şimdi, öncekiyle aynı işi yapmayı kabul ettiğinizi, ancak bu sefer 10.000 dolara aldığınızı hayal edin. A durumunda yaptığınız işin tam olarak aynısını yapıyorsunuz ve müşteri tatmin oluyor. Ama şimdi müşteri çok kötü bir yıl geçirdiğini ve ödeyebileceği tek şeyin 8.000 dolar olduğunu, al ya da bırak. Nasıl hissediyorsun?

Hem A Durumunda hem de B Durumunda tam olarak aynı miktarda iş yaptınız ve size tam olarak aynı miktarda para ödendi. Ama A Durumunda kendinden geçmiş, B Durumunda üzgün olurdunuz. Neden? Farklı olan nedir?

Fark sizin beklentilerinizde. A Durumunda size 5.000 \$ ödenmesini bekliyordunuz, ancak müşteri sizi şaşırttı ve beklentilerinizi aşarak sizi mutlu etti. B Durumunda 10.000\$ alma beklentileriniz karşılanmadı, bu da sizi mutsuz ediyor.

Bazıları, herhangi bir çabada, bu örneğin, müşteriye son derece mutlu edebilmek için müşterilerin beklentilerini kasıtlı olarak düşük tutmanız ve ardından bunları aşmanız gerektiğini gösterdiğini iddia edebilir. Ancak bu her zaman işe yaramayacak ve uzun vadede kesinlikle işe yaramayacak; doldurma programları veya başka türlü düşük beklentilerle ün kazanan insanlar, müşterilerinin ve müşterilerinin güvenini kaybeder.*

Ayrıca gereksinim mühendisinin paydaşlar üzerinde muazzam bir bilinçli ve bilinçsiz etki uyguladığını da bilin. Bir gereksinim mühendisi, müşterilerle iletişim kurarken, "Sistemin bunu yapmasını isterdim..." veya "Sistemin bunu yapmasını sevmiyorum..." gibi sözler söylemekten kaçınmalıdır. daha sonra pişman olacak ve potansiyel olarak sizi suçlayacak.

Bu nedenle gereksinim mühendisleri olarak hedefimiz, müşterilerin beklentilerini dikkatli bir şekilde yönetmektir. Yani her zaman müşteri beklentilerini anlamak, ayarlamak, izlemek, sıfırlamak ve ardından karşılamak.

Paydaş Müzakereleri

Gereksinim mühendisinin bu süreçte müşteriler ve diğer paydaşlarla müzakere etmesi kaçınılmazdır. Genellikle müzakereler, müşteriye istenen bazı işlevlerin imkansız veya çok maliyetli olduğuna ikna etmekle ilgilidir. Ve herhangi bir sistem projesinin yaşam döngüsü boyunca beklenti belirleme ve yönetimi, bir müzakere alıştırmadır. Bu kitaba müzakere teorisinde hızlandırılmış bir kurs yerleştirmek üzere olmasak da, hatırlanması gereken birkaç basit ilkedden bahsetmek istedik.

Temel kuralları önceden belirleyin. Müzakere yakınsa, müzakerelerin kapsamının ve süresinin kararlaştırıldığından emin olun. Üçüncü şahıslar varsa, bunun anlaşıldığından emin olun. Belirli kurallara uyulması gerekiyorsa, her iki tarafı da bilgilendirin. Müzakerede her iki taraf için de istenmeyen sürprizleri ortadan kaldırmaya çalışmak başarıya götürecektir.

İnsanların beklentilerini anlayın. Sizin için önemli olanın diğer taraf için önemli olmayabileceğini anladığınızdan emin olun. Bazı insanlar parayı önemser; diğerleri imajlarına, itibarlarına veya hislerine daha fazla önem verir. Müzakerelerle uğraşırken

* Yazar, seminerler verirken sık sık şu şakayı yapar: "Beklentileri kasten düşük tutmayı severim... ve sonra onları karşılamayı severim."

çevreleyen sistem işlevselliği, müşteri için neyin en önemli olduğunu anlayın. Sıralama gereksinimleri bu konuda en çok yardımcı olacaktır.

Erken başarıları arayın. Küçük bir şeyde bile anlaşmaya varılabılırsa, her zaman olumlu bir ivme oluşturmaya yardımcı olur. En tartışmalı meseleler hakkında erkenden kavga etmek, her türlü kötü duyguyu artıracak ve daha sonra bu küçük meseleler üzerinde anlaşmayı daha da zorlaştıracaktır.

Biraz verdiğinizden ve biraz geri ittiğinizden emin olun. Müzakerede biraz verirsiniz, her zaman iyi niyet gösterir. Ancak bir müzakerede geri adım atmanın değeri biraz mantığa aykındır. Geri itmeyerek, karşı tarafı aldatılmış ve boş hissetmesine neden oluyorsunuz.

Örneğin, birinin kullanılmış bir arabanın 10.000\$'dan satılık reklamını yaptığını varsayalım. Satıcıyı ziyaret ediyorsun, araca bakıyorsun ve 8.000 dolar teklif ediyorsun. Satıcı hemen kabul eder. Nasıl hissediyorsun? Muhtemelen satıcının çok kolay kabul ettiğini ve saklayacak bir şeyi olduğunu hissediyorsunuzdur. Veya 10.000 dolarlık fiyatın aşırı derecede şişirildiğini hissediyorsunuz - neden 10.000 dolar teklif etmiş olsaydınız, satıcı bunu kabul ederdi - ne kadar ağgözlü! Satıcı 8.000\$'lık teklifinizi reddedip yerine 9.000\$ ile karşılık verse, kendinizi daha iyi hissederdiniz. Bu yüzden biraz geri itin.

Müzakereyi yalnızca tüm taraflar tatmin olduğunda sonlandırın. Müzakereyi asla açık sorularla veya kötü duygularla sonlandırmayın. Müzakerenin sonunda herkesin kendini tatmin ve bütün hissetmesi gerekir. Karşılıklı memnuniyeti sağlamazsanız, büyük olasılıkla tekrar birlikte iş yapmayacaksınız ve genel olarak itibarınız zarar görebilir (müşteriler birbirleriyle konuşurlar).

Etkili müzakere üzerine pek çok iyi metin vardır (örneğin, Cohen 2000) ve tüm gereksinim mühendislerinin müzakere becerilerini sürekli olarak uygulamaları ve geliştirmeleri tavsiye edilir.

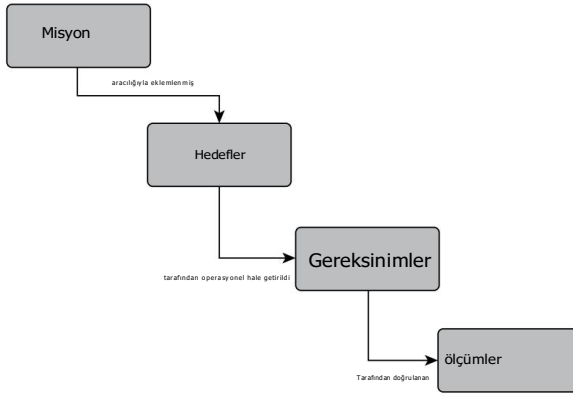
Paydaş Hedeflerini Ortaya Çıkarmak

Bir müşteriyle karşılaşmadan itibaren, müşteri hedeflerini anlamanın, gereksinim mühendisliği sürecinin başarısı için kritik olduğu açık olmalıdır. Hedefler, ürün misyon beyanında özetlenen sistemin amaçlarını daha da detaylandırır.

Örneğin, evcil hayvan mağazası POS'u için bazı hedefler:

- Tüm müşteriler için "sorunsuz" alışveriş sağlayın
- Tüm kupon ve indirim işlemlerini destekleyin
- Tüm müşteri sadakat programlarını destekleyin
- Envanter girişini ve bakımını tamamen otomatikleştirin
- Tüm yerel, eyalet ve federal vergi işlemlerini destekleyin

Ürün misyon beyanı, hedefler ve gereksinimler arasındaki ilişki Şekil 2.4'te gösterilmektedir.



Şekil 2.4 Hedefe dayalı gereksinim mühendisliği.

Hedeflerin, ürün misyon beyanında yer alan amacı nasıl daha fazla dile getirdiğine dikkat edin. Bu hedefler, ölçümlerle doğrulanması gereken gereksinimler aracılığıyla operasyonel olarak tanımlanır ve detaylandırılır.

Hedefe yönelik gereksinim mühendisliği, bu hedeflere ulaşmak için yeni işlevsel gereksinimler elde etmek için paydaş hedeflerinin analizini içerir. Mevcut hedef odaklı gereksinim mühendisliği teknikleri, KAOS, i* ve Tropos resmi yöntemlerini içerir (bkz. Bölüm 6). Bu yaklaşımlar, gereksinimlerin "kim, ne, neden, nerede, ne zaman ve nasıl" şeklinde modellenmesini amaçlar (Asnar ve ark. 2011).

Her paydaş, herhangi bir sistem için farklı bir hedef grubuna sahip olabileceğinden, hedef kümelerindeki farklılıkları uzlaştırabilmemiz için paydaş belirleme ve önceliklendirmeden sonra hedef anlayışını üstlenmeliyiz. Hedef-sorumetrik paradigmasını kullanan amaca dayalı anlayış için basit bir teknik Bölüm 3'te tartışılacaktır.

VIGNETTE 2.1 Sağlık Hizmetleri IoT'sinde Paydaşları Tanımlamak için Zengin Resimleri Kullanma

Nesnelerin İnterneti (IoT), İnternet'i kontrol, veri depolama ve analitik için kullanan sistemleri ifade eder. Tipik bir IoT sistemi, sensörler, aktüatörler ve dağıtılmış işlem gücü içerir. Sağlık hizmetleri, hastane ortamlarında, uzun süreli bakım tesislerinde ve evde IoT için önemli bir uygulama alanıdır. Bu sistemler, bu ortamlardaki insanları, ekipmanı, cihazları ve malzemeleri izlemek için kullanılabilir (Laplante ve diğerleri 2016). IoT sağlık sistemleri, optimum sağlık sonuçları, artan güvenlik, gizlilik ve verimlilik elde etme genel hedeflerine sahiptir.

İnsanların, ekipmanların ve malzemelerin hareketlerini izlemek ve çeşitli hasta kayıt sistemlerine bağlanmak için bir hastanede kullanılacak bir IoT sistemini düşünün. Paydaşları belirlemeye başlıyoruz

Egzersizler

- 2.1 Bir operasyon kavramına veya misyon beyanına sahip olmak neden önemlidir?
gereksinim mühendisliğinin başlangıcı?
- 2.2 Bir sistem veya ürün misyon beyanının bir iletişim ağıyla ilişkisi nedir?
pany'nin misyon beyanı?
- 2.3 Bir alan sözlüğü ne zaman oluşturulmalıdır?
- 2.4 Gereksinim geliştirmenin hangi aşamasında, gereksinimlere yapılan eklemeler
kapsam kayması olarak kabul edilir?
- 2.5 İin zengin bir resim oluřturun
 - 2.5.1 Bagaj taşıma sistemi
 - 2.5.2 Akıllı ev sistemi (Ek A)
 - 2.5.3 Islak kuyu pompalama kontrol sistemi (Ek B)
- 2.6 Bir blok diyagram veya zengin bir resim kullanmak, ařađıdakilerin bađlamını gosterir:
 - 2.6.1 Evcil hayvan mađazası POS sistemi
 - 2.6.2 Bagaj taşıma sistemi
 - 2.6.3 Akıllı ev sistemi
- 2.7 Müşterinin ihtiya ve istekleri hangi kořullar altında ikincil olarak gorlebilir?
- 2.8 Gnlk kullandıđınız bir rn dřnn.
 - 2.8.1 rn reten firma iin misyon ifadesinin ne olduđunu tahmin etmeye alıřın.
 - 2.8.2 Gerek misyon ifadesini internette arařtırın ve tahmininizle karřılařtırın. Varsa,
gerek misyon ifadesi sizin tahmininiz ile nasıl karřılařtırılır?
- 2.9 Havayolu bagaj taşıma sistemi iin beř hedef listeleyin.
- 2.10  paydařı daha (zaten verilmiř olanlar dıřında) ve bunların nceliklerini listeleyiniz.
Mary iin endiře
 - 2.10.1 Bagaj taşıma sistemi
 - 2.10.2 Evcil hayvan mađazası POS sistemi
- 2.11 İřteře bađlı bir sakinler ve ziyaretiler grubu iin (yalnızca iki yařlı yetiřkenden oluřan
mevcut kullanıcı grubu iin deđil) oluřturulacaksa, Ek A'da aıklanan akıllı ev sistemi
iin kapsamlı bir kullanıcı alt sınıfları listesi oluřturun.
- 2.12 Ařađıdaki paydařlar iin bir alt sınıf listesi oluřturun:
 - 2.12.1 Evcil hayvan mađazası POS sistemi mřterileri
 - 2.12.2 Bagaj taşıma sistemi iin yolcular
 - 2.12.3 Bir IoT sađlık sistemindeki hastalar
- 2.13 Ařađıdakiler iin, paydař katılımının bir temsilci kullanması gerekip gerekmediđini aıklayın.
gcendirci, rneklem veya kapsamlı yaklařım:
 - 2.13.1 Evcil hayvan mađazası POS sistemi iin sistem yneticileri
 - 2.13.2 Bagaj taşıma sistemi iin bakım teknisyenleri

- 2.13.3 Akıllı ev sistemi için ev sahipleri
2.13.4 Bir IoT sağlık sistemindeki hemşireler
- 2.14 KAOS gereksinim modelleme metodolojisini araştırın ve bu yaklaşımda hedeflerin nasıl modellendiğini tartışın.

Referanslar

- Asnar, Y., Giorgini, P. ve Mylopoulos, J. (2011). Gereksinim mühendisliğinde hedef odaklı risk değerlendirmesi. *Gereksinim Mühendisliği*, 16(2): 101–116.
- Boehm, B. (2003). *Dengeleme, Çeviklik ve Disiplin, Şaşkınlık İçin Bir Kılavuz*. Addison Wesley. Boston, MA.
- Checkland, P. & Scholes, J. (1999). *Yumuşak Sistemler Metodolojisi İş Başında*. Wiley. Batı Sussex, Birleşik Krallık
- Cohen, H. (2000). *Her Şeyi Pazarlık Edebilirsiniz*. Kale. New York, NY.
- Hofstede, G. (2001). *Kültürün Sonuçları: Değerleri, Davranışları, Kurumları ve Uluslar Arası Örgütler*. Adaçayı. Thousand Oaks, CA.
- Hull, E., Jackson, K. ve Dick, J. (2011). *Problem Alanında Gereksinim Mühendisliği, Gereksinim Mühendisliği*. 2. baskı Springer-Verlag, Londra, s. 93-114.
- Avcı, J. (2015). IoT "Thing" ihtiyaçlarının hiyerarşisi. *Techcrunch*, 15 Eylül 2015. <https://techcrunch.com/2015/09/05/the-hierarchy-of-iot-thing-needs/> (Kasım 2016'ya erişildi).
- Laplante, NL, Laplante, PA ve Voas, JM (2016). Sağlık hizmetlerinde nesnelerin interneti uygulamaları için paydaş tanımlama ve kullanım vakası temsili. *IEEE Sistemleri Dergisi*. DOI: 10.1109/JSYST.2016.2558449.
- Laplante, PA (2010). Carmen Sandiego dünyanın neresinde (ve o bir yazılım mühendisi mi)? *BT Uzmanı*, 12(6): 10–13.
- Maslow, A. (1943). İnsan motivasyonu teorisi. *Psikolojik İnceleme*, 50: 370-396.
- Newell, AF, Dickinson, A., Smith, MJ ve Gregor, P. (2006). Daha yaşlı kullanıcılar için bir portal tasarlama: Bir endüstriyel/akademik işbirliğine ilişkin bir vaka çalışması. *Bilgisayar-İnsan Etkileşimi Üzerine ACM İşlemleri*, 13(3): 347–375.
- Valacich, JH, Parboteeah, DV ve Wells, JD (2007). Çevrimiçi tüketicinin ihtiyaçlar hiyerarşisi. *ACM'nin İletişimi*, 50(9): 84–90.