

Mobil Cihazlar için Bağlam Duyarlı Arayüz Tasarımı

Şehra Şen, Ata Önal

Ege Üniversitesi, Bilgisayar Mühendisliği Bölümü, 35100, İzmir
sehra_sen@yahoo.com, ata.onal@ege.edu.tr

Özet: Mobil bilgi işlem, bilgisayarları masaüstünden yeni bağlamlara taşıyarak kullanım amaçlarını çarpıcı bir şekilde arttırma olanağı sunmaktadır. Bununla birlikte, bilgisayarların artan çok yönlülüğü kullanıcı arayüzü tasarımcıları için çözülmesi zor problemler yaratmaktadır [1].

Bir kullanıcının fiziksel ve sanal ortamları büyük oranda birleşmiş durumdadır. Günümüzde mobil cihazlar günlük yaşamın ayrılmaz bir parçası haline gelmişlerdir. Birçok kullanıcı, geleneksel bilgisayarlar konusunda deneyimsizdir, ancak bilgisayarlı sistemler ile günlük iletişim sağlamada oldukça deneyimlidir. Mobil cihazların hızla farklılaşması cihazların potansiyel kullanım alanlarını genişletmektedir. Uygulamalar için yazılım geliştirmeye verilen desteğe rağmen kullanıcı arayüzlerinin tutarlılığının, kullanılabilirliğinin sağlanmasında hala birtakım boşluklar bulunmaktadır.

Günümüzde abone olunan servislere erişim için çok sayıda farklı mobil cihaz kullanılmaktadır. Tek bir uygulama için çok sayıda kullanıcı arayüzü versiyonu sağlamak zorunda olan uygulama geliştiricilerinin işi giderek zorlaşmaktadır. Ayrıca uygulama hata yapmaya açık hale gelmekte ve giderek daha çok zaman almaktadır. Uygulamayı, istemcinin uygulama bağlamına kolayca adapte edebilen yeni yazılım geliştirme modelleri ihtiyacı doğmaktadır.

Anahtar Kelimeler: Mobil bilgi işlem, Kullanıcı Arayüzü Geliştirme, Kullanıcı Arayüzü Adaptasyonu.

User Interface Design for Mobile Devices

Abstract: Mobile computing offers a dramatic increase in uses of computers by carrying them from desktops to new contexts. However, the increasing versatility of computers causes difficult problems for user interface designers [1].

The physical and virtual environments of a user are almost integrated. Today, mobile devices became indispensable in everyday life. Several users are inexperienced about traditional computers, but they are almost experienced about communicating through computing systems. The rapid change of mobile devices, also extend the usage area of the devices. Despite the given support to software development, there are still gaps at providing consistency and usefulness of user interfaces.

Today lots of different mobile devices are being used to access subscribed services. The work of application developers, who have to provide several user interfaces for an application, gets increasingly difficult. Additionally, the application becomes error-prone and time consuming. New software development models, that can easily adapt the application to clients application context, are needed.

Keywords: Mobile Computing, User Interface Development, User Interface Adaptation.

1. Giriş

Bilgi işlem ortamlarının artan heterojenliği, bu ortamlar için etkileşimli sistemlerin geliştirilmesinde gereken çabayı giderek arttırmaktadır. Çözümü amaçlanan bir problemin iyi bir ön analizinin yapılabilmesi için öncelikle açık ve özlü bir problem tanımı yapılmalıdır. Kullanıcı arayüzü tasarımında günümüzdeki problem temel olarak şu şekilde tanımlanabilir: çok sayıda cihazı destekleyen kullanıcı arayüzlerinin tasarlanmasında, geliştirilmesinde ve yayılmasındaki desteğin eksikliği.

Bağlamsal bilgi işlem Weiser'in 1991'deki ilk çalışmalarında, bilgi işlem cihazları ile dolu bir ortamda cihazların birbirleriyle iletişim kurabilmeleri ve aynı zamanda insan kullanıcıları ile etkileşimde bulunabilmeleri hakkında kendi görüşünü belirtmesi ile başlar. Weiser'in görüşü en derin teknolojilerin, ayırt edilmeyecek kadar günlük hayatın içinde yerini almış teknolojiler olduğu fikrine dayanmaktaydı [5].

Bilgi işlem cihazları ile dolu ortamlarda, cihazların heterojenliği ve devingenliği nedeniyle uygulamalar güvenilir ve sabit bir işletim bağlamı üzerine geliştirilmemektedir. Ortamdaki değişikliklere tepki verebilen adaptif uygulamalar geliştirmek için bağlam duyarlı mekanizmalara gereksinim duyulmaktadır. Günümüzde var olan yaklaşımlar tam anlamıyla bu ortamların gereksinimlerini karşılayamamaktadır [5][9].

Bağlamsal bilgi işlem senaryoları oluşturmanın temel nedenlerinden bir tanesi kullanıcının heterojen terminallerde, farklı ağ teknolojileri üzerinden farklı erişim teknikleri kullanarak aynı uygulamaya erişebilme gereksinimleridir. Bunu gerçekleştirmek için, farklı bir cihaz veya farklı bir ağ teknolojisi için aynı uygulamayı yeniden geliştirmek yükünden kurtarmak için uygulama geliştiricisine yeterli düzeyde açıklık sağlayan özel yazılımlar ile birlikte yeni programlama paradigmalarına ihtiyaç duyulmaktadır. Geliştiricinin perspektifinden bakıldığında, mobil bilgi işlem pazarına her yeni

cihaz girildiğinde yeni bir kullanıcı arayüzü ve yeni içerik tipleri geliştirmek makul bir çözüm değildir. Uygulamanın sunum katmanının adaptif ve işletim bağlamından olabildiğince bağımsız olması sağlanmalıdır [5].

Etkileşimli yazılım sistemleri artık sadece belirli bir donanım veya tek bir bağlamda kullanılmak üzere tasarlanmamaktadır. Bu makalede kullanıcı arayüzünün geliştirilmesinde karşılaşılan problemlere değinilmekte, model-tabanlı kullanıcı arayüzü geliştirilmesi konusu incelenmekte ve örnek model-tabanlı kullanıcı arayüzü uygulama mimarilerinden söz edilmektedir.

2. Kullanıcı Arayüzü Geliştirilmesinde Karşılaşılan Problemler

Bu bölümde kullanıcı arayüzü tasarımında mevcut olan problemler ele alınmaktadır.

Etkileşimli uygulamalar ve uygulamaların kullanıcı arayüzleri birçok farklı bilgi işlem ortamında işlem görebilirler. Bu bilgi işlem cihazları, çok güçlü iş istasyonlarından küçük cep telefonlarına kadar geniş bir skalada yerlerini almaktadır. Kullanıcı arayüzü geliştirilmesinde karşılaşılan problemlerden bir tanesi *bilgi işlem gücündeki farklılıkların, geliştirilecek olan kullanıcı arayüzü için kısıtlar oluşturmasıdır*. Bir diğer problem *mobil bilgi işlem platformlarının kendilerine özgü girdi ve çıktı kısıtlarının olmasıdır*. Bazı cihazlar gelişmiş grafiksel yetenekleri desteklerken bazıları sınırlı ekran çözünürlüklerini desteklemektedir; bazıları gelişmiş girdi/çıkı cihazları ile donatılmışken bazıları ise sınırlı girdi cihazları ile limitlendirilmiştir [8].

Kullanıcı arayüzü mobil cihazların uygun olduğu çeşitli bağlamlardan yararlanabilmek için tasarlanmalıdır. Örneğin; PDA mobil bir cihazdır ve farklı amaçlar için kullanılabilir. Mobil bilgi işlem, kullanıcı bir iş yaparken bağlamsal değişimlerin olasılığını artırır. Örneğin; karanlık bir ortama girildiğinde PDA'nın ekranı güç tasarrufu sağlamak için karartılabilir; çevredeki

gürültü seviyesi arttığında sesin duyulabilmesini sağlamak için ses çıkış gücü artırılabilir veya aynı gürültülü ortamda telefon çaldığında sistemin ses etkileşimi ile birlikte titreşim etkileşimi devreye sokulabilir.

Farklı bağlamlardan yararlanma problemi, kullanıcı hareket halinde bir veya birkaç işi birlikte yaparken karmaşık bir hal alabilir. Örneğin, cep telefonu almak isteyen bir tüketici öncelikle masaüstü bilgisayarından cep telefonu özellikleri ile ilgili arama yapar ve otobüste yolculuk yaparken okumak üzere cep telefonu hakkında bulmuş olduğu bilgileri bir PDA'ya yükler, sonrasında cep telefonu yardımı ile sipariş verir. Ayrıca kullanıcılar heterojen platformları kullanırken bir iş hakkında bu platformların işbirliği yapmasını da isteyebilirler. Tüm bu platformların ve bağlam kısıtlarının ele alınmasında tutarlılığın ve yararlılığın sağlanması önemlidir.

Mobil bilgi işlem kullanıcı arayüzleri, *platforma* (ekran yüzeyine, renk derinliğine, ekran çözünürlüğüne adapte olan gösterimler ve ağ bant genişliğine adapte olan diyaloglar ile), *etkileşime* (önceden kullanılmış olan etkileşim tekniklerini, ekran boyutlarını ve konumlarını hatırlayan mekanizmalara sahip olarak ve bu gösterimlerde kullanıcı tercihlerine bağlı kalarak), *kullanıcılara* (kullanıcının deneyim düzeyine, sistem ve iş deneyimlerine, yeteneklerine, koşullara ve tercihlerine adapte olarak) karşı duyarlı olmalıdır [5].

Kullanıcı arayüzü geliştirilirken karşılaşılan bu sorunları çözmek için en çok benimsenen yöntem her durum için ayrı ayrı kullanıcı arayüzlerinin geliştirilmesidir. Ancak bu çözüm beraberinde başka problemleri getirmektedir. Tasarım ve geliştirme zamanlarında çok sayıda platform için benzer ve tutarlı bir kullanıcı arayüzü geliştirilmelidir. Bu ise platform bağımsız yazılımların geliştirilmesi ihtiyacını artırmaktadır ve süreci tekrarlı bir hale getirmektedir. Bu tekrarlı süreç kaynak tüketimini arttırmakta ve süreci karmaşıklamaktadır,

çünkü her platform kendi kısıtlarına sahiptir. Kullanıcı arayüzü gereksinimlerindeki küçük bir değişiklik bile son koda birçok önemli değişikliğe neden olabilmektedir. Ayrıca sözü geçen platform bağımsız tasarımın, farklı deneyimleri olan birçok farklı birey tarafından gerçekleştirilmesi de kaçınılmazdır. Bu da farklı kullanıcı arayüzleri arasında tutarlılığın sağlanmasını ve stillerin koordinasyonunun sağlanmasını zorlaştırmaktadır.

Tasarım hiçbir zaman sonuçlanmaz. Yeni mobil cihazlar hızla pazardaki yerlerini alırken, yazılım geliştiricileri onlara en hızlı ve en ucuz şekilde uyum sağlamak zorunda kalacaklardır. Kullanıcı arayüzü, her cihaz için tamamen yeniden tasarlanmak ve yeniden gerçekleştirilmek zorunda kalırsa süreç çok yavaşlayacaktır. Kullanıcı arayüzüne ait gerçekleştirim bir uygulamanın en zor taşınabilir kod kısmı içinde yer aldığı ise durum kötüleşecektir.

Sonuç olarak bağlamların çeşitliliği ve farklı cihazlar tasarımcıların farkında olmayabilecekleri birçok kullanılabilirlik sorunlarının merkezinde yer almaktadır. Kullanıcı arayüzü geliştirmek için mevcut ticari yazılımlar geliştiricilere bu sorunları çözmelerinde çok az yol göstermekte veya hiç yol göstermemektedirler. Bu nedenlerle mobil cihazların kullanıcı arayüzlerinin tasarımı için yapılan çalışmalar geliştirilmelidir. Kullanıcı arayüzü modelleme, mobil bilgi işlem için kullanıcı arayüzü geliştirmeye uzun dönemde herhangi bir etkin yaklaşımın gerekli bir bileşeni olacaktır [3].

3. Model Tabanlı Kullanıcı Arayüzü Geliştirme

Modeller etkileşimli bir sistemin farklı yanları hakkında bilgileri tanımlarlar. Model-tabanlı kullanıcı arayüzü geliştirme hakkında yazılan makalelerin sayısının büyüklüğüne rağmen, bu terimin aslında neleri içerdiği hakkında açık bir tanım yoktur. Bir model, elemanları ve bu elemanlar arasında belirlenmiş ilişkileri içeren, boş olmayan bir küme olarak tanımlanabilir.

Bir model sonuçta oluşan arayüzün yansıtacağı belirli bir fikir hakkındaki bilgiyi biraraya toplamakta ve ilişkilendirmektedir. Bir model bu fikrin bir soyutlamasını sağlar, alt seviyedeki detayları saklarken önemli detayları içerir. Arayüz fikrini oluşturan önemli karakteristikler üzerinde odaklanır, alt seviye detayların spesifikasyonu tasarım sürecinin daha sonraki bir aşamasına ertelenir.

Bir modelin elemanları belirli bir zamanda belirli değerlere sahiptir. Bu, model için çok genel bir tanımdır, ancak model-tabanlı kullanıcı arayüzü geliştirmeyi her bir modelin kendi özel değerlerine ve ilişkilerine sahip olduğu, ve farklı modellerin birbirleriyle ilişkilendirilebileceği bir küme model olarak görmemizi sağlar. Bu ilişkiler farklı şekillerde ifade edilebilir. Örneğin bazen çeşitli modeller aynı bilgiyi ifade eder veya bir model üzerine ek bilgiler eklenerek ayrı bir modele dönüştürülebilir.

Model-tabanlı kullanıcı arayüzü geliştirmede, kullanıcı arayüzünün farklı yanlarının tanımlanması için kullanılacak çok sayıda farklı model mevcuttur: veri modelleri, alan modelleri, uygulama modelleri, görev modelleri, diyalog modelleri, (soyut ve somut) sunum modelleri ve kullanıcı modelleri bilinen ve birçok model-tabanlı kullanıcı arayüzü geliştirme ortamlarında kullanılmış olan modellerdir. Veri, alan ve uygulama modeli sistemin uygulama mantığının sonuna yerleştirilebilir. Bu modeller, etkileşimli sistem tarafından kullanılacak veya sistem tarafından desteklenmesi gereken nesnelerin türlerini ve nesneler üzerinde kullanılacak olan işlemleri tanımlar. Görev ve kullanıcı modelleri kullanıcıya en yakın olan modellerdir ve kullanıcının gerçekleştirdiği görevleri veya kullanıcı profillerini sırasıyla belirler. Diyalog modeli ve sunum modeli en çok son kullanıcı arayüzüne yakındır. Yeni oluşan bir model ise bağlam modelidir: etkileşimli bir sistem için kullanım bağlamını tanımlayan bir modeldir. Kullanıcı arayüzü modelleme kullanıcı arayüzünün çeşitli bileşenlerini (gösterim, diyalog, platform, görev yapısı ve bağlam gibi)

tanımlayan bilgi tabanlarının oluşturulmasını içerir. Bu bilgi tabanları, tasarımcılara her bağlamın gereksinimleri ile eşleşen kullanıcı arayüzleri üretmelerinde, bir kullanıcı arayüzü geliştirme aracı yardımı ile kullanılabilir. Birçok tasarım kararı, akıllı kullanıcı arayüzü tasarım yazılımları ile otomatikleştirilebilir [3].

4. Kullanıcı Arayüzü Adaptasyonu

Kullanıcı arayüzünün yeni bir bağlam ile dinamik olarak değişebildiğinde adaptif olarak nitelendirilir. Adaptasyon tasarım zamanında otomatikleştirilmiş tasarım desteği olarak veya dinamik bir arayüz yaratarak çalışma zamanında gerçekleşebilir. Mobil bilgi işlemin değişen çeşitli bağlamlarına adaptasyonu sağlamak için bazı model tabanlı teknikler mevcuttur. Herbir teknik çeşitli model bileşenleri arasında eşlemelerin yapılmasını içerir. Bu eşleşmeler çevrildiğinde, ilgili cihaz ve kullanım bağlamına özel statik veya dinamik kullanıcı arayüzleri yaratılır [2].

Yüksek derecede adaptif, bağlam duyarlı bir çoklu platform kullanıcı arayüzünün geleneksel kullanıcı arayüzü tasarım teknikleri kullanılarak uygulanması oldukça zordur. Bunun gibi tüm teknikler bir kullanıcı arayüzü modelinin geliştirilmesine dayanmaktadır.

Bir kullanıcı arayüzü modeli kullanıcı arayüzünün formal, deklaratif ve uygulamadan bağımsız bir tanımıdır. Bir kullanıcı arayüzü modeli bir modelleme dili ile ifade edilir. Bir modelleme dili deklaratif olmalıdır, dili destekleyen yazılım araçları hakkında bilgi sahibi olmayan insanlar için bile anlaşılır olmalıdır. Bununla birlikte bir yazılım sistemi tarafından anlaşılabilmesi ve analizinin yapılabilmesi için biçimsel olmalıdır. Bir kullanıcı arayüzü modelleme dili platform bağımsız olmalıdır. Desteklenecek olan herbir platform için bir araç bir modelleme dilinden çalıştırılabilir bir kod parçasına çevrim yapabilmelidir. Bu çevrim statik kullanıcı arayüzleri oluşturulmak istendiğinde derleme zamanında, dinamik arayüzler oluşturmak için çalışma zamanında gerçekleştirilebilir [1].

Bir platform modeli bir kullanıcı arayüzünü çalıştırabilecek bilgisayarlı sistemleri tanımlar. Bu model platformun kullanıcı arayüzü üzerinde yarattığı kısıtlar hakkındaki bilgileri içerir. Tasarımcılar istenilen her bir platform için ayrı ayrı kullanıcı arayüzü tasarlayabilirler. Ancak platformun kullanım amaçlarına uygun olarak değişen koşullara duyarlı olmasını sağlamak için platform modelinden çalışma zamanında yararlanılmalıdır. Örneğin; platform modeli bant genişliğindeki ani bir azalmayı fark edebilmeli ve uygun şekilde davranabilmelidir. Sunum modeli kullanıcı arayüzünün görsel özelliklerini tanımlar. Sunum modeli pencere hiyerarşilerini ve bileşenlerini (kaydırma çubuğu, listeler gibi), stil seçeneklerini ve pencere bileşenlerinin seçimini ve yerleştirilmesini içerir. Bir görev modeli uygulama kullanıcılarının gerçekleştirmek isteyebileceği görevlerin yapısal bir gösterimidir. Görev modeli hiyerarşik alt görevlerden oluşur. Bir görevin seçimli olup olmadığı veya başka bir alt göreve izin verip vermediği gibi özellikler görev modeli ile modellenilebilir.

Daha önce de bahsettiğimiz gibi bu üç model dışında da mobil bilgi işlem ile ilgili modeller vardır. Örneğin; birçok uygulama için farklı tercihlere ve yeteneklere sahip kullanıcıların modellenmesi gerekmektedir. Ayrıca kullanıcı arayüzü tarafından desteklenen alan karakteristiklerinin modellenmesi de gerekmektedir. Bunun gibi bilgiler kullanıcı arayüzü bileşenlerinin seçilmesine yardım etmektedir.

Bugünlerde çok yaygın olan bir senaryoda, bir servis sağlayıcı bir küme bilgi servisini geniş bir potansiyel kullanıcı çeşitliliğine sunmak istemektedir ve her bir kullanıcı sisteme o anda elinde olan herhangi bir cihaz aracılığı ile nerede olursa olsun erişebilmektedir. Bu yazılım geliştiricilerin servisin kullanıcı arayüzü için kullanıcının cihazının özelliklerine bağlı olarak farklı servis versiyonları oluşturmak zorunda oldukları anlamına gelmektedir. Bisignano ve ark. tarafından önerilen çerçeve uygulama geliştiricilerini, her bir cihazın işletim ortamı için kullanıcı arayüzü ile ilişkili olarak yeniden kod

yazmaktan kurtaracaktır. Çerçevenin çekirdeği, kullanıcı ile cihaz arasındaki etkileşimi tanımlamaya yarayan bir XML tabanlı soyut bir spesifikasyon dili içermektedir. Bu tanım sonradan, istemcinin kullandığı cihaz için gerçek bir kullanıcı arayüzüne dönüştürülebilmektedir. Çerçeve istemcinin özelliklerine bağlı olarak medya içeriği için adaptasyonu sağlamaktadır. Hem lokal olarak işlenebilen olayları ve uygulamanın iş mantığı ile uzaktan işletilmesi gereken olayları ele alan bir olay işleme (*event handling*) mekanizması mevcuttur [5] [7].

Diğer uygulama ise Jacob Eisenstein ve ark. tarafından geliştirilmiş olan, mobil cihazlar için kullanıcı arayüzü geliştirmenin zorluklarının birçoğunu gösteren MANNA isimli kuramsal yazılımdır. MANNA çok sayıda farklı platform üzerinde çalışması gereken ve internet üzerinden iyileştirilebilen bir çoklu ortam uygulamasıdır. Yerbilimciler, mühendisler ve askeri personel tarafından kullanılması amaçlanmaktadır. [3]

5. Sonuç

Günümüzde mobil bilgi işlem sınırlarını genişletirken giderek karmaşıklaşmakta ve birçok bağlam için çeşitli erişim cihazları sunmaktadır. Bu ise kullanıcı arayüzü tasarımında bir küme soruna neden olmaktadır. Mobil bilgi işlem, tasarımcıları, sayısı giderek büyüyen cihazları desteklemeye zorlaması yönüyle, kullanıcı arayüzü tasarımının zorluğunu ve karmaşıklığını çarpıcı bir şekilde arttırmıştır. Geleneksel modeller ile dinamik kullanıcı arayüzleri geliştirmek yazılımı hataya açık hale getirmekte ve kaynak tüketimini arttırmaktadır. Bu nedenle sunum katmanının uygulamanın iş mantığından ayrılması gerekmektedir. Üstelik sunum katmanı, kullanıcı etkileşiminin soyut bir spesifikasyonundan başlanarak gerçek kullanıcı arayüzünü oluşturmaya yardımcı olacak bir şekilde yapılandırılmalıdır. Bu yapılandırmanın sağlanmasında ise model-tabanlı kullanıcı arayüzleri geliştirme teknikleri kullanılmalıdır. Model-tabanlı kullanıcı arayüzü geliştirme teknikleri mobil bilgi işlem bağlamlarına adaptasyon sağ-

lanmasını kolaylaştırmaktadır. Böylece yazılım geliştiricilerin üzerindeki yük azalırken, daha tutarlı uygulamalar geliştirilebilmektedir.

6. Kaynaklar

[1] Alamn, X., Cabello, R., Gmez-Arrıba, F., Haya P., Martnez, A., Martnez, J., Montoro, G., “Using context information to generate dynamic user interfaces”, 10th International Conference on Human-Computer Interaction, HCI International, 2003

[2] Jakobsson, M., “User interface adaptation in dynamic web applications for mobile devices”, Helsinki University of Technology Department of Computer Science and Engineering, December 2004

[3] Luyten, K., “Dynamic User Interface Generation for Mobile and Embedded Systems with Model-Based User Interface Development”, PhD thesis, transnational University Limburg: School of Information Technology, 2004

[4] [Bisignano, M., Di Modica, G., Tomarchio, O. , “Dynamic user interface adaptation for mobile computing devices”, Applications and the Internet Workshops, 2005. Saint Workshops 2005, 31-04 Jan. 2005

[5] Eisenstein, J., Vanderdonckt, J., Puerta, A., “Adapting to Mobile Contexts with User-Interface Modeling”, Mobile Computing Systems and Applications, 2000 Third IEEE Workshop, 2000, p. 83-92, Los Alamitos, CA, USA

[6] Eicken, T., Vogels, W., “Evolution of the Virtual Interface Architecture”, *IEEE Computer*, vol. 31, pp. 61–68, Nov. 1998.

[7] Bruemmer, D.J., Few, D.A., Kapoor, C., Goza, M., “Dynamic Autonomy for Mobile Manipulation”, In Proc. of the ANS / IEEE 11th Annual Conference on Robotics and Remote Systems for Hazardous Environments, Salt Lake City, UT, Feb. 12-15, 2006

[8] Satyanarayanan, M., “Fundamental Challenges in Mobile Computing”, Proc of the fifteenth annual ACM symposium on Principles of distributed computing, ACM Press, New York, 1996, pp. 1-7.

[9] Banerjee, S., Youssef, M., Larsen, R., Shankar A., A. A. et al. Rover, “Scalable Location-Aware Computing”, *IEEE Computer*, 35(10):46–53, Oct. 2002.