

...arazerinde de çalışabilirler.

10.8.5. Arayüz Tasarım İlkeleri

Kullanıcı arayüz tasarımı genellikle tasarımcının deneyimine ve uygulama alanını ne kadar tanıdığına bağlıdır. Yazılım geliştirmeye başlamadan önce uygulama alanının özellikleri dikkate alınarak arayüz standartlarının belirlenmesi büyük yarar sağlar.

Papatya Yayıncılık Eğitim

6. Hafta
1. Grup
Eskiye

Bundan sonra da herbir arayüz yazılım birimini tasarlayan ve geliştirenler bu standartlara uyarak en etkin, en dostça arayüzü tasarlamaya ve gerçekleştirmeye çalışırlar. Bu kısımda, iyi bir arayüz tasarlamak için dikkate edilmesi gereken çeşitli noktalara değinip önerilerde bulunacağız.

10.8.5.1. Yüksek Nitelik

Arayüzün nitelikli olmasını sağlayabilecek bazı temel kurallar şunlardır:

- Arayüzde kullanılan dil, yani lisan, kullanıcının rahatça anlayabileceği ve işini güvenli bir şekilde yapabileceği bir dil olmalıdır. Örneğin, yabancı dilin geçerli olduğu bir kullanım ortamında Türkçe bir arayüz kullanmaya çalışmak anlık çeviri ve algılama güçlüğü yaratabileceğinden arayüzün o dilde gerçekleştirilmesi daha uygun olur. Benzer şekilde, yabancı dil bilmeyen kullanıcılar için Türkçe dışında bir dilde sözcükler bulunması, uyarılar yapılması iyi olmayacaktır.
- Arayüzde uygulama alanında yaygın olarak kullanılan teknik terimler veya kısaltmalar kullanılmalıdır. Bu şekilde hem görsel karmaşıklık engellenir, hem de kullanıcının alışkın olduğu terimlerle çalışması sağlanmış olur.
- Arayüz üzerinde tüm bilgiler, uygulama alanına ve gerçeğine uygun olarak mantıksal bir sıralama içinde bulunmalıdır.
- Bilgi işleme için gerekli girdilerin tümü en çabuk ve hatasız şekilde girilebilmeli ve çözüm üretici birime aktarılabilirdir. Tek tek veri girişi isteyen ve defalarca onay gerektiren arayüzlerin kullanımı oldukça güçtür.
- Arayüz işlevleri uygulama alanı için gereksiz bilgi içermemelidir. Diğer bir deyişle, girdi ve çıktılar belirli düzeylere bölünmeli, gereksinim duyulduğunda ayrıntılı bilgiye erişilmelidir.
- Arayüz, olabildiğince tekdüze olmalı, algılamayı azaltacak karışıklıklara yer verilmemelidir. Çok bilgi sergilemek, çok sayıda denetimi aynı pencerede toplamak her zaman istenen bir özellik değildir; ayrıntıya ancak istenirse ulaşılmalıdır.
- Arayüz, sistemin o an içinde bulunduğu durum, çalışma kipi, yürütmekte olduğu işlem, hata durumları gibi bilgileri her an kullanıcıya gösterebilmelidir. Bu bilgileri saklamak ya da sürekli gösterebilmek seçeneklerin sunulması da yararlı olur.
- Arayüzün değişik bölümlerinde bir etkinliğin başlatılmasında herhangi bir çelişik durum olmamalıdır. Örneğin, dosya açmak için kullanıcı karşısına çıkan bir pencerenin özellikleri, bir dosyayı arayıp bulup onu belgeye ekleyen bir işlev için de aynı görüntüye sahip olmalıdır. Başka bir deyişle, arayüzün değişik yerlerinde aynı işlev, benzer şekilde başlatılmalı veya aynı tür sonuçlar benzer şekilde verilmelidir.
- Kullanıcı, sistem kullanım kılavuzlarını sürekli hatırlamak zorunda bırakılmamalıdır. Eğer bir işi yapmak için birbirini izleyen etkinlikler varsa bu durum, kullanıcı arayüzü tarafından açıklanmalıdır.

- Kullanıcı, arayüzün bir bölümünden diğer bölümüne geçtiğinde bir önceki bölümdeki bilgiyi hatırlamak zorunda kalmamalıdır.
- Etkinlikler doğrudan başlatılabilmesi, grafik ekranda dolaşma en az hareketle, gereksiz basamakları aşmak zorunda kalmadan yapılabilmesi. Sık tekrarlanan komutlar için kısayol tuşlarından oluşan menüler kullanılmalıdır.
- Sistemin bütün durumlarında, kullanıcı o anda yapmakta olduğu etkinliği sonlandırıp sistemden kolaylıkla ve sorunsuzca çıkabilmelidir (İptal veya Geri tuşlarının kullanımı gibi).
- Çok gerekli olmadıkça kullanıcı onayı isteyen bir uygulama diğer uygulamaların giriş-çıkışını durdurmamalıdır; yani tüm giriş odağını üzerine almamalıdır.
- Arayüz üzerinde verilen her komutun sistem tarafından doğru algılanıp işleme sokulduğunun bir göstergesi sergilenmelidir. Buna sistemin yanıtı da denebilir. Bu yanıt süresi kullanıcıyı sıkıntıya sokacak ya da paniğe düşürecek kadar uzun olmamalıdır (1-2 saniye normal bir süre sayılabilir).
- Sisteme uzun sürecek bir işlem yapması komutu verildiğinde, arayüz üzerinde işlemin neresinde olduğunu sürekli olarak kullanıcıya bildirecek bir zaman çubuğu bulunmalıdır. Bu çubuk üzerinde ne kadar iş yapıldığı ve daha ne kadar yapılacak iş kaldığını belirtmek yerine, ne kadar zaman harcandığı ve daha ne kadar süre beklemek gerektiği belirtilmelidir. Aksi takdirde, kullanıcı yanılgıya düşer (Örneğin, bir uygulamayı başlatırken hangi modüllerin belleğe yüklenmekte olduğu değil, yüklemenin bu hızla toplam 30 saniyede tamamlanması gerektiğine ilişkin bilgi ve süre geri sayımı yer almalıdır. Bazı uygulamalar, %80 yükleme için 10 saniye harcamış iken, kalan %20 yükleme için 40 saniye beklemektedir. Bu durumda kullanıcı daha ne kadar süre beklemesi gerektiği hakkında bilgi sahibi olamamakta, hatta işlemin sonlarına doğru sistemin kilitlendiğini bile düşünebilmektedir).
- Arayüz, az bir öğretim gerektirmelidir. Gerekli olan bilgiler arayüz kullanılırken verilmeli, bir tür kılavuzlama, küçük simgelerle anlatım ve standart terimler kullanılmalıdır. Kullanım sırası gelmemiş eylem tuşlarını aktif hale getirmek, veri giriş bölgelerini sırası geldikçe açmak gibi tedbirler kullanılmalıdır.
- Arayüz, herhangi bir uygulama tarafından gerek duyulan sesli uyarılar, görüntülü yardımlar gibi çeşitli çıkış türlerini destekleyebilmelidir.
- Biçim, büyüklük, yer, renk, nesne hareketleri, sesli işaretler, iletiler anlaşılır biçimde, estetik ve kültürel koşullara uygun olmalıdır.
- Sistemin görüntülü, yazılı, sesli ileti ve uyarıları tarafsız bir şekilde hazırlanmış olmalı, kesinlikle kişiselleştirilmemelidir. Uyarı için kullanılan metinler kısa ve öz olmalı, sorunu açıklamalı, suçlayıcı veya incitici ifadeler içermemelidir.
- Arayüzde kullanılacak menü şekilleri, seçenek isimleri, komutlar, terimler, şekiller, eylem tuşları ve bunların diziliş sıraları uygulama alanının özelliğine göre değişiklik gösterse bile, yaygın standartlara yakın olmalıdır. Bu şekilde kullanıcıların her zaman alışık oldukları görünüm ve kullanım yöntemleri nedeniyle öğrenme hızları yükselir, verimleri artar.

10.8.5.2. Kullanıcı Dostluğu

İnsan-bilgisayar arayüzü, etkileşimli yazılım sistemlerinde kullanıcı ile bilgi alış-verişini sağlayan kısım olduğundan, olabildiğince kullanıcı dostu olması, verimliliğini ve iletişim güvenilirliğini artırır. Kullanıcı dostu olan bir arayüzün sağladığı yarar-lardan bazıları şunlardır:

- Kullanıcı, kullanım kılavuzlarını (elkitabı) yanında bulundurmak veya kullan-mak zorunda kalmadan bilgisayardan yararlanabilir.
- Kullanıcı, yazılımı kullanırken birçok bilgiyi aynı anda karşısında görebildiği için çok miktarda bilgiyi belleğinde tutmak ve gerek duyduğunda anımsamak zorunda kalmaz (birçok bilgi aşırı bilgi demek değildir).
- Kullanıcı, uygulama alanının terimlerini kullanabilir.
- Değişik giriş ve çıkış birimleri yardımıyla (fare, klavye, ekran ve diğer düğme ya da göstergeler gibi) bilgi giriş/çıkışı çok kolaylaşabilir. Özel amaçlı girdi araçlarıyla bilgisayara seri bir şekilde komut verilebilir.
- Kullanıcı, yazılımı ve sistem araçlarını kullanırken veya yardım pencerelerinde dolaşırken pencere ve menü seçimlerini kolaylıkla yapabilir.
- Kullanıcı, en seri ve en doğru biçimde bilgi girebilir ve alabilir.
- Arayüz denetimi ile bilgi girişi ve çıkışının mutlaka doğru biçimde ve izin veri-len sınırlar içinde olması sağlanır.

10.8.5.3. Güvenilirlik

Arayüzü yeterince güvenli olmayan bir sistemin işlevleri ne denli iyi ve yüksek başarılı olursa olsun kullanıcıya zor anlar yaşattığı ve çöktüğü sürece hiçbir değer taşımaz. Yazılımın asıl bilgi işleme kısmı kadar kullanıcı arayüzünün de güvenilir olması için şunlara dikkat edilmelidir:

- Veri veya komut girişi sırasında yapılabilecek hataların bir çoğu arayüzden gelecek uyarılar ile önlenir. Bu nedenle, giriş arayüzü, hataları önlemek için tüm girdileri görüntülemeli, gerekli tip ve sınır kontrollerini yapabilmelidir.
- Giriş arayüzü, hatalara dayanıklılık göstermeli, yanlış veri ya da yanlış tuşa basılması gibi nedenlerden dolayı kilitlenmemeli ve çökmemelidir.
- Çıkış arayüzü tanımlanmış sınırlar içindeki değerleri göstermelidir.
- Çıkış arayüzü, kullanıcının beklediği konumda, beklediği şekilde bilgi sergile-meli, sistem yanıtı olarak yanlış gelen bir bilgiyi, yanlış olarak göstermemelidir.
- Uygulama alanının özelliğine göre, yapılmış bir işlevi geri almak mümkün olabilmelidir (İptal veya Geri Dön gibi).
- Bir arayüz, geri dönüşü mümkün olmayan işleri yapmadan önce kullanıcıya doğrulatmalıdır. Ancak, uygulama alanının özelliklerine bağlı olarak, çok kritik durumlarda zamandan kazanmak için onay aşaması kullanılmayabilir. Bu durumda, yetkilendirme işleminin başında bir kez uygulanır. Gerekirse çok önemli onaylar için özel korumalı tuşlar kullanılabilir.

10.8.5.4. Yardımlar

Grafiksel bir kullanıcı arayüzünün vazgeçilmez özelliklerinden biri her etkileşim aşamasında kullanıcıya yardım verebilmesidir. Yardım olanakları ve menüleri tasarlanırken dikkate alınması gerekli noktalar arasında şunları sayabiliriz:

- Yardım tüm işlevleri kapsayabileceği gibi, seçenekli olarak, yalnızca o anda yapılan işleve ait olarak da tasarlanabilir.
- Yardıma erişmek için bir menüden seçmek, bir tuşu tıklamak ya da klavyeden bir tuşa basmak yeterli olmalıdır.
- Yardım şekli olarak ayrı bir pencerede bir metin göstermek, yanda başka bir pencere açmak, ekranın belirli bir yerinde (durum çubuğu) ya da işaretçinin altında kısa bir açıklama göstermek, bunların biri ya da tamamını içermek şeklinde bir tasarım kararı verilerek uygulamanın tamamında tutarlı olacak şekilde aynı yöntem izlenmelidir.
- Yardımdan ana uygulamaya dönmek için iptal veya geri tuşuna basmak, pencereyi kapatmak, kaçış karakteri kullanmak gibi yöntemler benimsenmeli, hiç çıkmayıp bir kenarda tutma özelliği de dikkate alınmalıdır.
- Yardım metni basit bir düzyazı şeklinde olabileceği gibi konudan konuya geçme olanağı sağlayan yapıli metin (HTML gibi) şeklinde de olabilir. Pencere yönetim sistemi ve uygulama alanına bağıli olarak bir yöntem seçilmelidir.
- Yardım penceresi içinde zenginleştirici eklemeler de yapılabilir. Örneğin:
 - Arama motoru kullanılması günümüzde pek çok yazılım paketinde yaygın olarak kullanılır hale gelmiştir.
 - Sıkça sorulan sorular şeklinde verilen yardım da kullanıcının kendi sorununu ifade edebilecek hazır yapılar bulabilmesini sağlar.
 - Bağlama bağıli yardım da önemli özelliklerden biri olarak sayılabilir. Bu şekilde, özellikle programlama ortamlarında, bir anahtar sözcüğü işaret ederek onunla ilgili yardıma ulaşmak mümkün olabilmektedir.
- Yardımın içeriği erişim şekli kadar önemlidir. Unutulmaması gereken nokta, yardım metninin yardıma gereksinim duyan kullanıcının kafasını karıştırmayacak şekilde açık olması ve arzu edilen bilgiyi hemen vermesidir.
- Bazı kullanıcılar uygulama yazılımını yardımlardan giderek öğrenmeye çalışırlar. Bu gerçeği göz önünde bulundurarak, yardımların aşamalı olarak, yönlendirici ve uygulatici bir yapıya sahip olması yararlı olur.

10.8.5.5. Hatalar ve Uyarılar

Kullanıcı arayüzünün bir özelliği de bilgi işleyici kısmı kullanıcı hatalarından korumak ve hata durumlarında kullanıcıyı uyarmaktır. Bu uyarıların bir kısmı kullanıcının hatalı bilgi girmesinden dolayı yapılırken, bir kısmı geri dönüşü olmayan bir işlem gerçekleştirilmeden önce onay almak için yapılır. Bir kısmı da yazılımın bütünü içinde meydana gelen bir hatayı kullanıcıya bildirmek ve ne gibi bir işlem yapması gerektiğini açık lamakta kullanılır.

Bu bildirimler yapılırken aşağıda belirtilen noktalara dikkat edilmesi yararlı olacaktır.

- Kullanıcının bilgi girmesi gereken yerde yanlış biçimde bilgi girilmesi ya da eksik alan bırakılması durumunda, kullanıcı arayüzü hatalı girişi belirterek gerekli uyarıyı vermelidir.
- Görsel uyarılar gerçekten dikkat çekecek şekilde, ekranın ortasına ve gerekli görüldüğü takdirde başka bir iş yapılmasına izin vermeyecek şekilde (giriş odağını üzerine alarak) yapılmalıdır. Görsel uyarılar sesli uyarılarla da desteklenebilir; fakat sesli uyarı tek başına kullanılmamalıdır.
- Sistem hatası ile ilgili açıklamaların açık ve sade bir dille, panik yaratmayacak şekilde yapılması gereklidir (Örneğin; "Sistemde büyük bir hata oluştu, bilgisayarınızı yeniden başlatınız" ifadesi hiç de hoş değildir).
- Hata iletilisi sorunu açıklıkla tanımlamalı, bir takım kod numaraları sergileyerek yalnızca hata oluştuğunu belirtmekle yetinmemelidir.
- Hata iletilisi içinde kesinlikle kullanıcıyı suçlayıcı, incitici ifadeler yer almamalıdır. Bu tür yaklaşımlar profesyonellikten uzaklaşma anlamına gelir.
- Sistem bir hata oluştuğu takdirde mutlaka geri kazanım olanağı sağlanmalıdır. Örneğin, bir metin düzenleyicide önemli bir belge yazarken önemsiz bir giriş hatası nedeniyle programı kapatan hata iletilisi, kullanıcının emeklerinin boşa gitmesine neden olur. Onun yerine, ya hata giderilmeli ya da programdan çıkmadan önce belgenin son halini saklamaya fırsat verilmelidir.

10.8.5.6. Yapısal Özellikler

Bir arayüzde bulunması gereken yapısal özellikleri de şu şekilde özetleyebiliriz:

- Bir arayüzde, kullanım sırasında ortaya çıkabilecek programlama hatalarının düzeltilmesi ya da daha sonra olabilecek kullanıcı istekleri ile ilgili değişiklikler kolay ve hızlı olarak yapılabilenlidir.
- Mümkünse belirli bir kütüphane kullanılarak standart geliştirme sağlanmalıdır.
- Grafiksel kullanıcı arayüzleri için estetik görünüm oldukça önemlidir. Pencere-lerde renklendirme aşırıya kaçmadan, belirli bir yöntem izlenerek yapılmalı, tüm nesnelere hizalanmalı ve etiketler çok düzgün yerleştirilmelidir. Bu maksatla önceden hazırlanmış standart öğelerin kullanımı sağlanmalıdır.
- Arayüz, uzmanlar, yeni başlayanlar gibi farklı kullanıcılar için gerekli işlevleri (kısayol gibi) sağlamalıdır.
- Arayüzün kullanıcı ile daha uyumlu hale gelebilmesi için çeşitli seçeneklerin değiştirilip saklanabilmesi gereklidir. Örneğin, kullanılan sergileme lisansının, menü içeriklerinin, simgelerin, komutları başlatmak için kısa yol tanımlamalarının değiştirilebilir olması yarar sağlayıcı özelliklerdir.

6. Hafta
1. Erp
San