



Web İçeriği Erişilebilirlik Kılavuzu 1.0



5 Mayıs 1999

Abdulkadir ANAÇ

Gülşen BARDAKCI

Ayten ÖZTÜRK

Arzu KIRAK

Tarafından Türkçeye uyarlanmıştır.

Özet

Web erişilebilirliği özürlü kişiler tarafından Web sayfalarının kullanılabilmesi demektir. Daha detaylı açıklamak gerekirse; Web sayfalarına erişilebilirlik; sayfaların özürlü kişilerin algılayabileceği, anlayabileceği, gezinebileceği, etkili bir şekilde kullanabileceği ve bunun yanında da içerik yaratıp, katkıda bulunabileceği bir şekilde tasarlanmasıdır.

Web erişilebilirliği görme, işitme, fiziksel, konuşma, algılama ve sinirsel özürülülerin de dahil olduğu tüm özürlü kişileri içermektedir.

Dünyada Webi kullanırken olumsuz olarak etkilenen ve özürlü olan milyonlarca insan vardır. Mevcut birçok Web sitesi içerdikleri erişilebilirlik engellerinden dolayı özürlü kişilerin ulaşımını çok zor veya imkansız kılmaktadır. Halbuki Web siteleri veya yazılımları erişilebilir yapılırsa, özürlü kişiler de Web sayfalarını etkili bir şekilde kullanabilecektir.

Web erişilebilirliği kurumlar ve özürlü olmayan kişiler için de yararlı olabilir. Örneğin, Web erişilebilirliğinin ana ilkelerinden biri farklı ihtiyaçların, durumların ve tercihlerin esnek bir şekilde karşılanabilmesidir. Bu esneklik özürlü olmayan kişiler için de yararlı olabilir, örneğin yavaş İnternet bağlantısı olan kişiler, "geçici özürülülüğü olan" kişiler, örneğin kolu kırılmış olanlar ve yaşları ilerlemiş olan kişiler gibi.

Web, özürlü kişilere bilgi ulaşımı ve iletişimi açısından eşi görülmemiş bir olanak sağlamaktadır. Web'in eğitim, işletme, ticaret, devlet işleri ve eğlence alanlarındaki rolü gün geçtikçe artmaktadır. Özürülülerin de toplumda aktif bir rol oynamasına olanak tanıyan erişilebilir bir Web, birçok alanda özürlü kişilere eşit olanaklar sağlamak için şarttır.

Bir Web sitesini erişilebilir kılmak, [içeriğın](#) türü, sitenin büyüklüğü ve karmaşıklığı, kullanılan yazılım gibi birçok faktöre bağlıdır.

Bir kurumun sitesini erişilebilir yapmak için atılması gereken başlıca adımlar bu kılavuzda listelenmektedir.

Bu kılavuz Web içeriğinin özürlü insanlar için nasıl erişilebilir hale getirileceğini açıklamaktadır. Bu kılavuz tüm Web içerik geliştiricilerine (sayfa yapanlar ve site tasarlayanlar) ve yaratım araçları geliştiricilerine yöneliktir. Temel amaç, erişilebilirliğin ilerlemesine yardımcı olmaktır. Bununla birlikte, kılavuzun ilkelerini izlemek Web içeriğini tüm kullanıcılar için daha kullanışlı yapacaktır. Ayrıca kılavuzun ilkelerini izlemek insanlara Web üzerinden daha hızlı bilgi bulmada yardımcı olacaktır. Bu kılavuz içerik geliştiricilerin Görüntü video, vb. kullanmasını engellemez onun yerine multimedya içeriklerinin nasıl daha erişilebilir yapılacağını açıklar.

"Web içeriği için HTML Teknikleri Erişilebilirlik Kılavuzu 1.0" olarak adlandırılan ayrı bir doküman da denetim noktalarının nasıl uygulanması gerektiğini açıklamaktadır.

Teknik Döküman HTML, CSS, SMIL ve MathML'i kullanarak örnekler sağlamaktadır ve her denetim noktasını daha detaylı bir şekilde ele almaktadır. Teknik dokümanı ayrıca teknolojideki değişikliklere göre tasarlanmıştır. Bütün tarayıcılar ve multimedya araçları kılavuzdaki tarif edilen özellikleri desteklemeyebilir. Özellikle HTML 4.0 , CSS 1 yada CSS 2'nin yeni özellikleri desteklenmeyebilir.

İçerik

- [Özet](#)
- [1. Giriş](#)
- [2. Erişilebilir Dizaynın Konuları](#)
 - [2.1 İyi Dönüşüm Sağlama](#)
 - [2.2 İçeriği Anlaşılabilir Yapma](#)
- [3. Organize Kılavuz İlkeleri Nasıl Düzenlenir](#)
 - [3.1 Doküman Kuralları](#)
- [4. Öncelikler](#)
- [5. Uygunluk](#)
- [6. Web İçeriği Erişilebilirlik Rehberi](#)
 - [1. Sesli ve görsel içerik için alternatif metin eşdeğerleri sağlayın.](#)
 - [2. Sadece renklere güvenmeyin](#)
 - [3. Uygun bir şekilde işaretleme ve CSS kullanın.](#)
 - [4. Doğal, anlaşılabilir bir dil kullanın.](#)
 - [5. Mükemmel şekilde dönüşebilen tablolar oluşturun.](#)
 - [6. Sayfaları yeni teknoloji özelliklerine göre uygun hale getirin.](#)
 - [7. Zamanla değişen içerikler \(kayan yazılar gibi\) kullanıcı kontrolünde olmasını sağlayın.](#)
 - [8. Kullanıcı arayüzlerinin erişilebilir bir tasarıma sahip olmasını sağlayın.](#)
 - [9. Cihazlardan \(fare gibi\) bağımsız olarak okunabilecek bir Web sayfası dizaynı yapınız.](#)
 - [10. Gerektiğinde geçici çözümler kullanın.](#)
 - [11. W3C teknolojileri ve kılavuz bilgilerini kullanın.](#)
 - [12. İçerik ve Yönlendirme bilgisi sağlayın.](#)
 - [13. Kolay kullanılabilir bir dolaşım mekanizması sağlayın.](#)
 - [14. Dokümanların anlaşılır ve kolay olmasını sağlayın.](#)

1.Giriş

Web sayfası dizaynıyla ilgili erişilebilirlik konularıyla tanışık olmayanlar birçok kullanıcının sizinkinden çok farklı kapsamlarda(durumlarda) çalışıyor olabileceğini göz önüne alın:

- Onlar göremiyor, duyamıyor, hareket edemiyor yada bazı tip bilgileri kolaylıkla yada hiç işleyemiyor olabilirler.
- Metin okuma yada anlama zorlukları olabilir.
- Klavyeleri ya da fareleri olmayabilir ya da bunları kullanamıyor olabilirler.

- Sadece metin gösteren bir ekranları olabilir ya da küçük bir ekrana ya da yavaş bir İnternet bağlantısına sahip olabilirler.
- Belgenin yazılmış olduğu dili konuşamıyor ya da anlayamıyor olabilirler.
- Gözleri, kulakları ya da ellerinin meşgul olduğu ya da müdahaleli ortamlarda olabilirler (örneğin arabayla işe gidiyorken, gürültülü bir ortamda çalışma, vs.).
- Bir tarayıcının daha erken bir versiyonuna ya da tamamen farklı bir tarayıcıya yahut bir sesli tarayıcıya veya farklı bir işletim sistemine sahip olabilirler.

İçerik geliştiriciler bu farklı durumları sayfa dizaynı boyunca dikkate almalılar. Dikkate alınacak bir çok durum varken, her erişilebilir dizayn seçeneği genel olarak bir seferde çeşitli özür gruplarına ve bir bütün olarak Web topluluğuna fayda sağlar. Örneğin, stil sayfaları kullanarak

Kılavuzlar erişilebilirlik sorunlarını tartışır ve erişilebilirlik dizayn çözümleri yaratırlar. Belirgin özürleri olan kullanıcıların sorunlarını ortaya koyan tipik senaryolara yönlendirme yaparlar. Örneğin birinci kılavuz içerik geliştiricilerin görüntüleri nasıl erişilebilir hale getireceklerini açıklamaktadır. Bazı kullanıcılar görüntüleri göremeyebilirler, bazıları tarayıcın görüntü desteğini yavaş internet bağlantısını engellemek için kapatmış olabilir,diğer bir kısmı da görüntülü gösterimi desteklemeyen text temelli tarayıcı kullanıyor olabilir. Kılavuzlar erişilebilirliği sağlamak için görüntülü gösterimden kaçınmayı önermezler. Onun yerine erişilebilirlik için görüntünün metin eşleniğinin sağlanması gerektiğini açıklarlar.

Metin eşlenikleri imajları nasıl erişilebilir yapmaktadır? İki kelimedede “metin eşleniği” önemlidir.

- Metin içeriği kullanıcılara sentezlenmiş konuşma, kör alfabesi ve görünür metin halinde sunulabilir. Bu üç mekanizmanın her biri çeşitli duyu engelleri ve değişik engelleri bulunan gruplara bilgiyi erişilebilir yapmak için farklı bir algı kullanmaktadır.
- Yararlı olabilmesi için metin imajla aynı amacı veya işlevi taşımalıdır. Örneğin uzaydan dünyanın görünümünün fotoğrafı şeklinde olan bir imajın metin eşleniğini düşünün.eğer fotoğrafın amacı sadece dekorasyon içinse metin eşleniğinin “dünyanın uzaydan görünümünün fotoğrafı” cümlesi ile ifade edilmesi gerekli işlevi görür. Eğer fotoğrafın amacı dünya coğrafyası ile ilgili özerk bir bilgi göstermekse, metin eşleniği bu bilgiyi içermelidir. Eğer fotoğraf dünya hakkında bilgi için imajı seçmeyi (örneğin üstüne tıklayarak) gerektiriyorsa text söyle olmalıdır: “dünya hakkında bilgi”. Dolayısıyla eğer bir metin engeli olan biri için diğer kullanıcılara imajın sağladığı aynı işlevi ve amacı sağlıyorsa buna bir metin eşleniği denilebilir.

Şu söylenebilir ki metin eşlenikleri engellilere yararlı olmakla birlikte arama motorları sayfaları indekslemede metinleri kullandığından sayfanın daha hızlı bulunması açısından tüm kullanıcılara yararlı olabilir.

Web içerik geliştiricileri imajlar ve diğer çoklu ortam içeriklerinin text eşleniklerini sağlamak zorundadır bunun yanı sıra bilgileri kullanıcıya sunmak kullanıcı temsilcilerinin sorumluluğundadır.(örneğin tarayıcılar ve ekran okuyucuları kör alfabesi görüntüleyicileri gibi destek teknolojileri vb.)

Metnin metin olmayan eşlenikleri(örneğin ikonlar,önceden kaydedilmiş konuşmalar yada metni işaret diline çeviren bir kişinin videosu) dokümanları yazılı metne zorlukla erişebilen Anlama özürlü öğrenme özürlü ve sağır bir çok bireyi kapsayan insanlar için erişilebilir yapabilir. Metnin metin olmayan eşlenikleri ayrıca okumayı bilmeyen kişiler içinde yararlıdır. İşitsel bir tanımlama, görsel bilginin metin olmayan eşleniğine bir örnektir. Bir multimedya sunumunun görsel parçasının işitsel tanımlaması görsel bilgiyi göremeyen insanlara fayda sağlar

2. Erişilebilir Dizaynın Konuları

Kılavuzda 2 önemli konu yer alıyor: iyi bir şekilde dönüşüm sağlanmalı, ve içerik anlaşılabilir olmalı.

2.1 İyi Dönüşüm Sağlama

Bu kılavuzu izleyerek, içerik geliştiriciler sayfaları iyi bir şekilde dönüştürüp oluşturabilirler. İyi bir şekilde dönüşen sayfalar giriş de belirtilen fiziksel, duyuşal ve kavramayla ilgili özürlülükler, iş kısıtları, ve teknolojik engelleri kapsayan herhangi bir kısıta rağmen erişilebilirliğini sürdürmelidir. Burada iyi bir şekilde dönüşmüş sayfaların dizaynı için bazı anahtarlar var:

- Sunumdan yapıyı ayırın.
- Metin sağlayın. Metin neredeyse tüm kullanıcılar için erişilebilir ve neredeyse tüm tarayıcılar için geçerli olan yollarla sunulmalı.
- Kullanıcı göremese ve/veya duyamasa bile işe yarayan dokümanlar oluşturun. Duyuşal kanalları değiştirmeye uygun yollarla ses yada video gibi aynı amaca yada fonksiyona hizmet eden bilgiler sağlayın. Bunun
- Uygun yollarla, gerektiğinde duyuşal kanalların yerini alabilen aynı amaca ve fonksiyona hizmet eden video ve ses bilgileri sağlayın. Fakat bunun anlamı görmeyen kullanıcılara siteyi erişilebilir yapmak için önceden kaydedilmiş tüm sitenin sesli versiyonunu oluşturmak değildir. Görme engelli olan kullanıcılar

bir sayfada ki bütün metin bilgisini dönüştürmek için ekran okuyucu teknolojisini kullanabilir.

- Tek bir donanıma dayanmayan dokümanlar oluşturun. Sayfalar fareler olmadan, küçük ekranlarla, düşük çözünürlükte ekranlarla, siyah beyaz ekranlarla yada ekranı olmayan sadece ses veya metin çıktısı olan insanlar tarafından kullanılabilir olmalı.

Dönüşüm konuları bu kılavuzun 1-11. ilkelerinde yer almaktadır.

2.2 İçeriği Anlaşılabilir Yapma

İçerik geliştiriciler içeriği anlaşılabilir yapmalı. Bu sadece dilin basit ve açık olmasını kapsamaz ayrıca sayfalar arasında ve içinde anlaşılabilir mekanizmalar sağlanmalı. Sayfaların içinde dolaşım araçlarını ve yönlendirme bilgilerini sağlayarak erişilebilirlik ve kullanılabilirlik en yüksek dereceye çıkar. Görüntü haritaları, orantılı kaydırma çubukları, yan yana çerçeveler yada grafikler gibi görsel ipuçlarını tüm kullanıcılar kullanamayabilir. Ayrıca kullanıcılar bir sayfanın sadece bir bölümünü görebildikleri zaman içeriksel bilgiyi kaybedebilirler, çünkü onlar sayfaya kelime kelime yada bölüm bölüm erişirler. Yönlendirme bilgileri olmadan, kullanıcılar çok büyük tabloları, listeleri, menüleri.. vb. anlayamayabilirler.

İçeriği anlaşılabilir yapma konuları bu kılavuzun 12-14. ilkelerinde yer almaktadır.

3. Organize Kılavuz İlkeleri Nasıl Düzenlenir

Bu doküman on dört tane kılavuz, veya erişilebilir dizaynın genel prensiplerini içermektedir.

- Kılavuz ilkelerinin numaraları
- Kılavuz cümleleri
- Kılavuz dolaşım linkleri.
- Kılavuzun arkasındaki temel mantık ve kılavuzdan yararlanan bazı gruplar.
- Kontrol noktalarının tanımlarının bir listesi.

Her bir kılavuzdaki kontrol noktası tanımları kılavuzun tipik içerik geliştirme senaryolarını nasıl uygulanacağını açıklar.

- Kontrol noktalarının numaraları
- Kontrol noktası anlatımları.
- Kontrol noktalarının öncelikleri. Öncelik 1 kontrol noktaları stil sayfalarının kullanımı boyunca gölgelendirilmiştir.
- İsteğe bağlı bilgi verici notlar, aydınlatıcı örnekler, kılavuzun kendi içindeki kılavuz ilkelerine ve kontrol noktalarına olan referanslar.
- Anlatılan kontrol noktalarının örneklerini ve yürürlüğe koyuluşunu açıklayan Teknikler dokümanının bir bölümüne bir link.

Her bir kontrol noktasının yeterince açıklayıcı ve özerk olması planlandı dolayısıyla bir sayfayı ya da siteyi gözden geçiren birinin kontrol noktalarının şartlarının sağlandığını doğrulayabilir.

3.1 Doküman Kuralları

Aşağıdaki makale kuralları bu doküman boyunca kullanılmıştır.

- Ögelerin isimleri büyük harfli olacaktır
- Öznitelik isimleri küçük harflerle ifade edilmelidir.
- Tanımlara olan bağlantılar biçim sayfalarının kullanımı boyunca gölgelendirilmelidir.

4. Öncelikler

Her bir kontrol noktası kontrol noktalarının erişilebilirliğe etkisi temeline dayanarak çalışan bir çalışma grubu tarafından atanmış bir öncelik düzeyine sahiptir.

[Öncelik 1]

Bir içerik geliştirici bu kontrol noktalarının şartlarını sağlamak zorundadır. Diğer durumda birçok grubun dokümanda ki bilgilere ulaşması imkansız hale gelecektir. Bazı grupların web dokümanlarını kullanabilmeleri için bu kontrol noktalarını sağlamak temel bir gerekliliktir.

[Öncelik 2]

Bir içerik geliştirici bu kontrol noktalarının şartlarını sağlamalıdır. Aksi takdirde birçok grup dokümanda ki bilgilere ulaşmakta zorlanacaktır. Bu kontrol noktalarının şartlarını sağlamak web sayfalarının erişimindeki bazı belirgin engelleri ortadan kaldıracaktır.

[Öncelik 3]

Bir içerik geliştirici bu kontrol noktalarını referans olarak gösterebilir. Diğer durumda birçok grup dokümandaki bilgilere ulaşmakta biraz zorlanabilir. Bu kontrol noktalarının şartlarını sağlamak web dokümanlarının erişilebilirliğini geliştirecektir.

Bazı kontrol noktaları birtakım belirli durumlarda değişebilir bir öncelik düzeyi belirtebilir.

5. Uygunluk

Bu bölüm bu dokümana uygunluğun üç düzeyini tanımlamaktadır.

- **Uygunluk düzeyi “A”:** Bütün Öncelik 1 kontrol noktaları şartları sağlanmalıdır.
- **Uygunluk düzeyi “Çift-A”:** Bütün Öncelik 1 ve 2 kontrol noktaları şartları sağlanmalıdır.
- **Uygunluk düzeyi “Üçlü-A”:** Bütün Öncelik 1 , 2 ve 3 kontrol noktaları şartları sağlanmalıdır.

Not: Uygunluk düzeyleri metin içerisinde anlatıldı dolayısıyla konuşmada gündeme getirildiğinde anlaşılabilirliklidir.

Bu dokümana uygunluğun ispatı için şu iki biçim kullanılmalıdır.

Biçim 1: Şunları Belirtin :

- Kılavuzların başlıkları : “Web Content Accessibility Guidelines 1.0”
- Kılavuzların URL’leri: <http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505>
- Sağlanmış olan uygunluk düzeyleri: “A”, “Çift-A” veya “Üçlü-A”.
- İspatlarla kapsanmış faaliyet alanı (e.g.,sayfa,site,veya sitenin tanımlı kısmı.)

Biçim 1 Örneği:

Bu sayfa W3C'nin "Web Content Accessibility Guidelines 1.0" a uymaktadır.

Şu linkte bulunabilir: <http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505>, (düzey Çift-A)

Form 2: Her bir sayfadaki uygunluk ispatlarını ve W3C tarafından sağlanan üç ikondan birini içerir ve ikonu ispatın uygun W3C açıklamasına bağlar. İkonlarla ilgili bilgiler ve bunların sayfaya nasıl eklenecekleri [\[WCAG-ICONS\]](#) 'da mevcuttur.

6. Web İçeriği Erişilebilirlik Rehberi

1.İlke Sesli ve görsel içerik için alternatif metin eşdeğerleri sağlayın.

Kullanıcıya gösterildiğinde görsel ve sessel içeriğin amacına uygun olacak şekilde içerik sağlanmalı.

Bazı insanlar resim, video, ses ve uygulamaları direk olarak kullanamasa da bu görsel ve seli içeriğin eşdeğer bilgilerini içeren sayfaları rahatlıkla kullanabilir. Eşdeğer bilgi, görsel ve sessel içeriğin amacına uygun olmalıdır. Böylece indeks sayfasının link ini içeren ok işaretinin metinsel eşdeğeri "indeks sayfasına git" olmalıdır. Bazı durumlarda metinsel eşdeğer, görsel veya sessel biçimin (diyagramlar, grafikler, ses örnekleri) içeriğini de tarif etmelidir.

Bu kılavuz metin tipinde olmayan (imaj, ses kaydı, video) içeriğin metinsel eşleniğinin sağlanmasının önemini vurgular. Metinsel eşleniklerin gücü değişik teknolojiler kullanan farklı özür gruplarından olan insanlara sunum yapabilmesinde yatar. Metin konuşma sentezleyicileri ve körler alfabesi görüntüleyicilerine kolayca çıktı olabilmelidir ayrıca bilgisayar ekranında ve kâğıt üstünde görüntülenebilmelidir. Sentezlenmiş konuşma (yazılımlar tarafından seslendirilmiş) körler, sağırılar okuma güçlüğü, idrak etme engeli, öğrenme güçlüğü olan engelliler için çok önemlidir.

Kontrol noktaları:

1.1 Metin olmayan her bir öge için metinsel bir karşılık sağlanmalıdır.(Örneğin; "alt", longdesc" vb. komutlar yoluyla olabilir.) **[Öncelik 1]**

Mesela, HTML'de:

- IMG, INPUT ve APPLELET elementleri için "alt" kullanın.
- Karmaşık içeriklerde (çizelge gibi) "alt" komutunun tam bir metin eşleniği sağlamadığı durumlarda, IMG veya FRAME ile birlikte "longdesc" kullanarak ek bir açıklama sağlayınız.
- İmaj haritaları içinde "alt" özelliğini AREA ile birlikte kullanın veya MAP elementini A elementleriyle birlikte bir içerik olarak kullanın.

1.2 Sunumcu tarafı bir görüntü haritasının her bir etkin bölgesi için bolca metinsel bağlantı sağlanmalıdır. **[Öncelik 1]**

2.İlke Sadece renklere güvenmeyin.

Renksiz olduğu zamanda grafik ve metinlerin anlaşılabilir olmasını sağlayın. Eğer bilgi vermek için sadece renk kullanılıyorsa, belirli renkleri ayırt edemeyen insanlar ve renksiz veya görüntüsüz ekranları kullananlar bu bilgiyi alamazlar. Ön plan ve arka plan renkleri aynı tona yakın olduğu zaman, tek renkli(siyah-beyaz) ekran kullanılarak bakıldığında ya da farklı tiplerde renk eksikliği olan insanlar tarafından izlendiğinde, bunlar yeterli kontrast sağlamayabilir.

Kontrol noktaları:

2.1 Renkle aktarılan tüm bilgilerin aynı zamanda renksiz de varlığı(anlaşılması) sağlanmalıdır. (Örneğin içerik ve biçimlerden anlaşılması sağlanmalıdır.) **[Öncelik 1]**

2.2 Renk eksikliği olan kişiler tarafından bakıldığında veya siyah beyaz ekrandan bakıldığında ön plan ve arka plan renk kombinasyonları yeterli kontrastı(zıtlığı) sağlamalı. [görüntüler için Öncelik 2, metin için Öncelik 3]

3.İlke Uygun bir şekilde işaretleme ve CSS kullanın.

Kontrol noktaları:

3.1 Uygun bir işaretleme dili mevcutken, bilgiyi vermek üzere görüntülerden çok işaretleme kullanın. [Öncelik 2]

3.2 Yayınlanmış geçerli dil bilgisi kuralları geçerli olan dokümanlar oluşturulması gerekir. [Öncelik 2]

3.3 Düzen ve sunumu kontrol etmek için CSS kullanılması gerekir. [Öncelik 2]

3.4 İşaretleme dili özellik değerlerinde ve tarz sayfası özellik değerlerinde mutlak birimlerden çok göreceli birimler kullanın.(örneğin 'pt' yada 'cm' yerine 'em' yada '%' kullanın.) [Öncelik 2]

3.5 Belge yapısını aktarmak için başlık unsurlarını özelliklerine uygun olarak kullanın. [Öncelik 2]

3.6 Kurallara uygun şekilde listeleri işaretleysin ve maddeleri listeleyin. [Öncelik 2]

3.7 Alıntıları işaretleysin paragraf yapma gibi biçimleme etkileri için alıntı işaretlerini kullanmayın. [Öncelik 2]

4.İlke Doğal, anlaşılabilir bir dil kullanın.

Kısaltılmış ya da yabancı tekstleri okumayı ve yorumlamayı kolaylaştırmak için "markup" kullanın.

Kontrol noktaları:

4.1 Bir belgenin metin ve Örneğin, HTML'de "lang" özelliğini kullanın. XML'de, "xml:lang"ı kullanın metin eşleniklerinin (örneğin başlıklar) doğal dilindeki değişimi açıkça tanımlayın. [Öncelik 1]

Örneğin, HTML'de "lang" özelliğini kullanın. XML'de, "xml:lang"ı kullanın.

4.2 Her bir kısaltma ve kısa adların uzun halini belge içerisinde ilk geçtiği yerde tanımlayın. Örneğin HTML'de, ABBR ve ACRONYM öğelerinin "title" (başlık) özelliğini kullanın. Belgenin ana gövdesinde açık halini sağlamak da ayrıca belgenin kullanılabilirliğine yardımcı olur. [Öncelik 3]

4.3 Bir belgenin birincil doğal dilini tanımlayın. Örneğin HTML'de HTML öğesi üzerinde "lang" özelliğini ayarlayın. XML'de "xml:lang" özelliğini kullanın. [Öncelik 3]

5.İlke Mükemmel şekilde dönüştürülen tablolar oluşturun.

Tablolar dönüştürülebilirlik için gerekli işaretleme dillerine sahip olmalı!

Tablolar, gerçekten tablo halindeki bilgileri (“veri tabloları”) işaretleme üzere kullanılmalıdır. İçerik geliştiriciler bunları yerleşim sayfalarında (“yerleşim sayfaları”) kullanmaktan kaçınmalıdır. Herhangi bir kullanım için olan tablolar ayrıca ekran okuyucuların kullanıcıları açısından da özel problemle neden olur.

Bazı kullanıcı ajanları kullanıcıların tablo hücreleri arasında gezinmelerine ve başlık ya da diğer tablo hücre bilgilerine erişmelerine izin vermektedir. Doğru bir şekilde işaretlenmedikçe, bu tablolar kullanıcı programlarına uygun verileri sağlamayacaktır.

Aşağıdaki uyarılar sesli cihazlar (örneğin bir ekran okuyucu ya da bir taşınabilir kişisel bilgisayar) aracılığıyla erişen ya da bir kerede sayfanın yalnızca bir kısmını gören (örneğin konuşma çıkışı ya da kör alfabesi ekranı kullanan ama ya da düşük görüşe sahip kullanıcılar ya da küçük ekranlı aygıtların kullanıcıları) insanlara doğrudan faydalı olacaktır.

Kontrol noktaları:

5.1 Veri tabloları için, sütun ve satır başlıklarını tanımlayın. **[Öncelik 1]** Örneğin HTML’de veri hücrelerini tanımlamak üzere TD ve başlıkları tanımlamak üzere TH kullanın.

5.2 İki ya da daha fazla satır ya da sütun başlığı mantıksal düzeyine sahip olan veri tabloları için, veri hücrelerini ve başlık hücrelerini ilişkilendirmek üzere işaretleme kullanın. **[Öncelik 1]** Örneğin HTML’de, satırları gruplamak üzere THEAD, TFOOT, ve TBODY ve sütunları gruplandırmak üzere COL ve COLGROUP’u ve veriler arasında daha karmaşık ilişkileri tanımlamak üzere “axis”, “scope”, ve “headers” özelliklerini kullanın.

5.3 Tablo çizgisel hale getirildiğinde anlamlı hale gelmedikçe, yerleşim için tabloları kullanmayın. Aksi takdirde, eğer tablo anlam içermezse, (çizgiselleştirilebilen bir versiyon olan) bir alternatif eşlenik sağlayın. **[Öncelik 2]**

Not: Kullanıcı ajanları tarz sayfası konumlandırmasını desteklediğinde, tablolar yerleşim için kullanılmamalıdır.

5.4 Eğer bir tablo yerleşim için kullanılırsa, görsel formatlama amacıyla yapısal işaretleme kullanmayın. **[Öncelik 2]**

5.5 Tablolar için özetler sağlayın. **[Öncelik 3]** Örneğin HTML ‘de TABLE’ın “summary” özelliğini kullanın.

5.6 Başlık etiketleri için kısaltmalar sağlayın. **[Öncelik 3]** Örneğin HTML’de TH elementi için “abbr” özelliğini kullanın.

6.İlke Sayfaları yeni teknoloji özelliklerine uygun hale getirin.

Yeni teknolojiler desteklemese bile sayfalar her zaman erişilebilir olmalı.

Kontrol noktaları:

6.1 Dokümanlar sayfa stilleri olmadan da okunabilecek şekilde düzenleyin. Örneğin, bir HTML dokümanı bu dokümanla ilintili sayfa stilleri olmadan da okunabilmelidir. **[Öncelik 1]**

6.2 Dinamik içerik değiştiğinde, eşdeğer olan içerikleri de güncellenmeli. **[Öncelik 1]**

6.3 Yazılım nesnelere ("scriptler", "appletler" vb.) devre dışı bırakıldığında ya da desteklenmediğinde bu sayfalar hala kullanılabilir durumda olmalı. Bu mümkün değilse, alternatif olarak erişilebilen bir sayfada eşdeğer bilgi sağlanmalı. **[Öncelik 1]**

6.4 input işleyen betikler ve applet'ler girdi alınan yazılımdan bağımsız olmalıdırlar. **[Öncelik 2]**

6.5 Bir alternatif sayfa ya da sunumun temini ya da dinamik içeriğin ulaşılabilir olması sağlanması gerekir. **[Öncelik 2]**
Örneğin HTML'de her bir frameset için NOFRAMES kullanın.

7.İlke Zamanla değişen içerikler (kayan yazılar gibi) kullanıcı kontrolünde olmasını sağlayın.

Bazı anlama ve görme özürü kişileri bu yazıları yeterince hızlı yada hiç okuyamayabilir. Ayrıca yazılardaki hareket anlama özürü kişileri olan insanlarda dikkat dağıtıcı olabilir. Ekran okuyucular hareketli yazıları okuyamaz.

Kontrol noktaları:

7.1 Kullanıcı temsilcileri, kullanıcıya titreşimi kontrol etmeye izin verinceye kadar ekranın titremesine neden olmaktan kaçınılmalıdır. **[Öncelik 1]**

7.2 Kullanıcı temsilcileri, kullanıcıya yanıp sönmeyi kontrol etmeye izin verinceye kadar yanıp sönen içeriklerden kaçınılmalıdır. **[Öncelik 2]**

7.3 Kullanıcı temsilcileri, kullanıcıya hareketli içerikleri dondurmaya izin verinceye kadar, sayfalarda hareketten kaçınılmalıdır. **[Öncelik 2]**

7.4 Kullanıcı temsilcileri yenilemeyi durdurma yeteneği sağlayana kadar, periyodik olarak otomatik yenilenen sayfalar oluşturulmamalıdır. **[Öncelik 2]**

7.5 Kullanıcı temsilcileri otomatik yeniden göndermeyi durdurma yeteneğini sağlayana kadar, otomatik olarak sayfaları yeniden göndermek için biçimlendirme

kullanılmaması gerekir. Bunun yerine, yeniden göndermeler yapmak için sunucu(server) konfigüre edilir. **[Öncelik 2]**

8.İlke Kullanıcı ara yüzlerinin erişilebilir bir tasarıma sahip olmasını sağlayın.

Kullanıcı ara yüzleri, erişilebilir tasarım prensiplerine uymalı: işlevselliğe aygıttan bağımsız erişim, klavye kullanabilme, kendinden seslendirme, vb

Gömülü bir obje "kendi ara yüzü"ne sahipse, ara yüz – browserın kendi ara yüzü gibi – erişilebilir olmalı. Eğer gömülü nesnenin ara yüzü erişilebilir yapılamıyorsa, alternatif bir erişilebilir çözüm sağlanmalı.

Kontrol noktaları:

8.1 Betik ya da appletler gibi program niteliğindeki unsurlar erişilebilir yada özürlü teknolojileriyle uyumlu yapılmalı. **[Öncelik 1]**

9.İlke Cihazlardan (Mouse gibi) bağımsız olarak okunabilecek bir web sayfası dizaynı yapınız.

Bedensel engelli kullanıcılar için. Bazı bedensel engelli kullanıcılar, bilgisayarın faresini kullanamadıklarından, klavyeye bağımlı olarak web sitelerini gezebiliyor. Bu nedenle ACCESSKEY, TABINDEX, LABEL gibi HTML etiketleri kullanılması mutlaka gereken etiketler arasında yer alır.

Aygıttan bağımsız erişim; kullanıcının kullanıcı aracı(temsilcisi) veya dokümanı ile tercih edilen bir giriş (çıkış) aygıtı (fare, klavye, ses, head wand, veya diğerleri.) ile etkileşim kuracağı anlamına gelir. Eğer, örneğin, bir form kontrolü sadece fare veya başka bir işaretleyici cihaz ile yapılıyorsa, görme duyusu olmadan, ses girişi ile veya klavye ile sayfayı kullanan birinin veya işaretleyici olmayan bir giriş cihazı kullanan birisi formu kullanamayacaktır.

Note: Link olarak kullanılan imaj haritalarının veya imajların metin karşılıklarının sağlanması kullanıcının bunlarla işaretleyici cihaz olmadan etkileşimde bulunmasını sağlar.

Genellikle klavye etkileşimine izin veren sayfalar konuşma girişi (speech input) veya komut satırı arabirimi (command line interface) içinden de erişilebilirdir

Kontrol noktaları:

9.1 Mevcut geometrik bir şekille tanımlanamayan bölgelerdekiler hariç sunucu tarafındaki resim haritaları yerine istemci tarafındaki resim haritaları sağlanmalıdır. **[Öncelik 1]**

9.2 Kendi ara yüzüne sahip herhangi bir elementin aygıttan bağımsız biçimde işletilmesini sağlayın. **[Öncelik 2]**

9.3 Betikler için, cihaza bağlı durum denetimcileri yerine mantıksal durum denetimcileri (işleyicileri) belirleme yoluna gidin. **[Öncelik 2]**

9.4 Linkler, kontrol formları ve nesnelere için mantıklı sekme sıraları oluşturun.

Örneğin HTML’de sekme sıralarını “tabindex” özniteliği ile belirleyin ya da mantıksal bir sayfa dizaynı sağlayın. **[Öncelik 3]**

9.5 Önemli linklere (istemci tarafı görüntü haritalarında bulunanlar dahil), form nesnelere ve form nesnesi gruplarına klavye kısa yolu sağlayın.

Örneğin HTML’de kısa yolları “accesskey” özniteliği ile sağlayın. **[Öncelik 3]**

10. İlke Gerektiğinde geçici çözümler kullanın.

Erişilebilir teknolojiler ve eski tarayıcıların doğru bir şekilde çalışması için geçici erişilebilir çözümler kullanılmalı.

Örneğin, eski tarayıcılar kullanıcıya boş edit kutularında gezinmeye izin vermez. Eski ekran okuyucular arka arkaya listelenen linkleri tek bir link gibi okur. Bu yüzden bu aktif öğelere erişim zor ya da imkansızdır. Ayrıca değişen geçerli(aktif) pencere ya da beliren yeni pencereler bunun olduğunu göremeyen kullanıcılar için çok kafa karıştırıcı olabilir.

Kontrol noktaları:

10.1 Kullanıcı temsilcilerinin açılmış pencereleri kapatmasına izin verene kadar pop-up ve diğer pencerelerin görülmesine neden olmamalı ve aktif pencere kullanıcıyı bilgilendirmeden değiştirilmemelidir. **[Öncelik 2]**

10.2 Kullanıcı temsilcileri form kontrolleri ve etiketler arasındaki belirgin ilişkiyi destekleyene kadar, dolaylı olarak ilişkili etiketlerle bütün form kontrolleri için, etiketin uygun bir biçimde yerleştirilmesi sağlanmalı. **[Öncelik 2]**

10.3 Kullanıcı temsilcileri(özürlü teknolojilerini içeren) metni doğru olarak yan yana getirene kadar, bütün tablolar için doğrusal bir metin alternatifi sağlanmalı. **[Öncelik 3]**

10.4 Kullanıcı temsilcileri “boşluk kontrolünü” doğru bir şekilde kullanana kadar, edit kutularında ve metin alanlarında varsayılan (default) , yer tutucu karakterler içermeli. **[Öncelik 3]**

10.5 Kullanıcı temsilcileri (özürlü teknolojileri dahil) bitişik linkleri ayırt ederek sunana kadar, bitişik linkler arasında link olmayan, basılabilir karakterler(boşluklarla çevrili) dahil edilmeli. **[Öncelik 3]**

11.İlke W3C teknolojileri ve kılavuz bilgilerini kullanın.

W3C teknolojilerini(tanımlamalara göre) kullanın ve erişilebilirlik kılavuzuna uyun. W3C teknolojisi kullanmanın mümkün olmadığı yerde, ya da incelikle çevrilmeyen maddelerde erişilebilir olan içeriğin alternatif bir versiyonu sağlanmalı.

Bugünkü kılavuz W3C teknolojileri(HTML,CSS, gibi...) çeşitli nedenlerden tavsiye edilir:

- W3C teknolojileri hazır erişilebilir özellikler içerir.
- W3C tanımlamaları erken inceleme dener.

Kontrol noktaları:

11.1 Bir görev için uygun ve elverişli olduğunda ve son versiyonla desteklendiğinde W3C teknolojilerini kullan. [**Öncelik 2**]

11.2 W3C teknolojilerinin reddedilen özelliklerinden sakının.Örneğin, HTML'de reddedilmiş FONT elemanını kullanmayın; onun yerine CSS kullanın(CSS'de ki 'font' özelliği gibi) [**Öncelik 2**]

11.3 Kullanıcıların tercihlerine göre doküman alabilmesi için bilgi sağlayın(örneğin; dil, içerik tipi vb.) [**Öncelik 3**]

11.4 Eğer en iyi çabalardan sonra erişilebilir bir sayfa oluşturamıyorsanız, w3c teknolojisini kullanan, erişilebilir, eşdeğer bilgi içeren, mümkün olduğunca sık güncellenen erişilemeyen alternatif sayfalar için bir link sağlanmalıdır. [**Öncelik 1**]

12.İlke İçerik ve yönlendirme bilgisi sağlayın.

Sayfanın içeriğini anlatan ve genel olarak yapısı hakkında açıklayıcı bir yönlendirme yapan bir özet hazırlayınız.

Elemanlar arasındaki ilişkiler hakkında elemanları gruplamak ve içeriksel bilgi sağlamak tüm kullanıcılar için faydalı olabilir. Bir sayfanın bölümleri arasındaki karmaşık ilişkiler anlama ve görme özürülü insanların yorumlaması için zor olabilir.

Kontrol noktaları:

12.1 Her bir çerçevenin tanımını ve dolaşımını kolaylaştırmak için her çerçeve adlandırılmalıdır. [**Öncelik 1**]

12.2 Eğer çerçeve başlıkları açık değilse çerçevelerin amaçlarını ve her bir çerçevenin birbirleriyle nasıl ilişkili olduğunu tanımlayın. [**Öncelik 2**]

12.3 Geniş bilgi bloklarını, uygun ve doğal olan kullanışlı gruplara bölün. [**Öncelik 2**]

12.4 Etiketleri kontrolleriyle açıkça birleştirin. [**Öncelik 2**]

13.İlke Kolay kullanılabilir bir dolaşım mekanizması sağlayın.

Kontrol noktaları:

13.1 Açık bir şekilde her bir ilişkinin amacını açıklayın. İlişim metinleri anlaşılabilir olmalı temsil ettiği içeriği anlamlı bir şekilde ifade etmeli. Ayrıca ilişkiler kısa ve öz olmalı.

Örneğin HTML’ de “burayı tıklayın” yerine “sürüm 4.3 hakkında bilgi” şeklinde açıklama içeren bir ifade yazmak daha yararlı olur. Anlaşılır ilişim metinlerine ek olarak bilgi verici bir ilişim başlığı ile içerik geliştiriciler daha da açıklayıcı olabilir. (örnek HTML ‘ de “title” özniteliği.) **[Öncelik 2]**

13.2 Sayfalara ve sitelere anlamlı bilgi ekleyebilmek için metadata sağlayın. Örneğin dokümanların içerik tipini yazarını göstermek için RFD kullanın. **[Öncelik 2]**

13.3 Sitenin genel tasarımı ile ilgili genel bir bilgi sağlayın.(örneğin site haritası veya içerik tablosu) Site tasarımını anlatırken mevcut erişilebilirlik özelliklerini gölgelendirin ve açıklayın.**[Öncelik 2]**

13.4 Dolaşma mekanizmasını kalıcı (sabit) bir durumda kullanın. **[Öncelik 2]**

13.5 Dolaşma mekanizmasına erişim sağlayan ve vurgulayan dolaşma çubukları sağlayın. **[Öncelik 3]**

13.6 İlgili linklerin gruplanması, belirlenmesi (kullanıcı bileşenleri için) ve kullanıcı bileşeni yapana kadar gruplar arasında geçiş için bir yol tanımlanması. **[Öncelik 3]**

13.7 Eğer araştırma işlevleri sağlanırsa, farklı düzeydeki seviye ve tercihler için farklı araştırma çeşitleri olanaklı kılınabilir.**[Öncelik 3]**

13.8 Konu, paragraf, liste gibi ayırt edici bilgiler başlığa konulabilir. **[Öncelik 3]**

13.9 Doküman koleksiyonu hakkında bilgi sağlanabilir. (Çoklu sayfaları kapsayan bilgiler gibi)**[Öncelik 3]**

13.10 Çok satırlı ASCII resimleri göz ardı ederek atlayacak mekanizma sağlanması **[Öncelik 3]**

14.İlke Dokümanların anlaşılır ve kolay olmasını sağlayın.

Grafikler okunabilir olmalı. Anlama zorluğu olanlar, zihinsel özürler, okuma güçlüğü olanlar için daha basit ve anlaşılabilir bir dil kullanılmalıdır.

Tutarlı sayfa düzeni, ayırt edilebilir grafikler, ve kolay anlaşılabilir dil bütün kullanıcılara fayda sağlar. Özellikle anlama özürlü insanlar veya zor okuyan insanlara yardımcı olur.(Buna rağmen görüntülerin(imajların) metin eşdeğerlikleri kör insanlar, düşük görme yetisine sahip insanlar ve grafikleri seçemeyen herhangi insanlar için sağlanmalı.)

Açık ve basit dil kullanarak etkili iletişim kurulabilir. Yazılı bilgiye erişmek anlama ve öğrenme özürlü insanlar için zor olabilir. Ayrıca açık ve basit dil kullanarak kendi dili sizinkinden farklı olan insanlara da fayda sağlar.

Kontrol noktaları:

14.1 Bir site içeriği için uygun olan en açık ve en basit dil kullanılmalıdır.

[Öncelik 1]

14.2 Sayfanın anlaşılmasını kolaylaştıracak yerde işitsel ya da grafik sunumlu ek metinler sağlanabilir.**[Öncelik 3]**

14.3 Sayfalar üzerinde geçerli genel bir sunum stili oluşturulabilir. **[Öncelik 3]**

Web İçeriği için HTML Teknikleri

Erişilebilirlik Kılavuzu 1.0

6 Kasım 2000

İçerik

İÇERİK.....	18
1.DOKÜMAN YAPISI VE METADATA.....	20
1.1 METADATA (ÜSTVERİ)	20
1.2 YAPISAL GRUPLAMA.....	22
2.DİL BİLGİSİ.....	23
2.1 DİLDEKİ DEĞİŞİKLİKLERİ TANIMLAMA	23
2.2 TEMEL DİLİN TANIMLANMASI	23
3.METİN BİÇİMLENDİRİLMESİ.....	23
3.1 VURGU	23
3.2 KISA ADLAR VE KISALTMALAR	24
3.3 TIRNAKLAR “”	24
3.4 GÖRÜNTÜLER YERİNE BİÇİMLENDİRME VE STİL SAYFALARININ KULLANILMASI.....	24
3.5 DİĞER YAPISAL 8 ÖGE	25
4.LİSTELER	25
4.1 LİSTELERİN MADDE İMLERİNİ DEĞİŞTİRMEK İÇİN STİL SAYFALARININ KULLANIMI	27
5.TABLolar	27
5.1 VERİ TABLOLARI.....	27
5.2 SAYFA GÖRÜNÜMÜ İÇİN TABLOLARIN KULLANILMASI.....	28
5.3 TABLOLARIN LİNEER HALE GETİRİLMESİ	29
5.4 TABLOLAR İÇİN ARKAPLAN İLE UYUŞMAZLIK SORUNU	30
6.LİNKLER.....	30
6.1 LİNK METNİ	30
6.2 GRUPLAMA VE ATLAMA LİNKLERİ.....	31
6.3 KLAVYE ERİŞİMİ	31
6.4 BAĞLANTILAR VE HEDEFLER.....	31
7.GÖRÜNTÜLER VE GÖRÜNTÜ HARİTALARI.....	31
7.1 GÖRÜNTÜLER İÇİN KISA EŞDEĞER METİNLER.....	31
7.2 GÖRÜNTÜLERİN UZUN BETİMLEMELERİ.....	32

7.3	ASCII ART	32
7.4	GÖRÜNTÜ HARİTALARI	33
7.5	GÖRÜNTÜLERDEKİ RENKLER	34
7.6	HAREKETLİ GÖRÜNTÜLER	34
8.	APPLETLER VE PROGRAMA AİT DİĞER NESNELER.....	34
8.1	APPLETLER VE PROGRAMA AİT DİĞER NESNELER İÇİN METİNLER VE METİN OLMAYAN EŞDEĞERLERİ.....	34
8.2	DOĞRUDAN ERİŞİLEBİLİR APPLETLER	35
9.	SES VE GÖRÜNTÜ	35
9.1	SES BİLGİSİ.....	35
9.2	ÇOKLU ORTAM İÇİN METİN EŞDEĞERLERİ	35
9.3	GÖMÜLÜ ÇOKLU ORTAM NESNELERİ.....	36
10.	ÇERÇEVELER	36
10.1	ÇERÇEVE BAŞLIĞININ SAĞLANMASI.....	36
10.2	ÇERÇEVELER ARASI İLİŞKİLERİ TANIMLAMA	36
10.3	ÇERÇEVELERİ DESTEKLEMİYEN TARAYICILAR İÇİN YAZIM ŞEKLİ	37
10.4	ÇERÇEVE KAYNAKLARI	37
10.5	ÇERÇEVE HEDEFLERİNİN KULLANILMASI.....	37
10.6	ÇERÇEVELER İÇİN ALTERNATİFLER.....	37
11.	FORMLAR.....	38
11.1	FORMLAR İÇİN KLAVYE ERİŞİMİ	38
11.2	FORM KONTROLLERİNİ GRUPLAMA.....	38
11.3	FORM KONTROLLERİNİ ETİKETLEME.....	38
11.4	GRAFİKSEL BUTONLAR	38
11.5	ÖZEL KONTROLLER İÇİN TEKNİKLER	39
11.6	FORMLAR İÇİN ARKAPLAN İLE UYUMSUZLUK SORUNU	39
12.	SCRIPTLER	39
12.1	SCRIPTLERİN ZARİF DÖNÜŞÜMÜ	39
12.2	TİTREŞİMLERE NEDEN OLAN SCRIPTLER	40
12.3	HAREKETLİLİĞE VE YANIP SÖNMELERE NEDEN OLAN SCRIPTLER	40
12.4	DOĞRUDAN ERİŞİLEBİLİR SCRIPTLER	40
12.5	SCRIPTLERİN ALTERNATİF SUNUMLARI	40
12.6	SAYFA GÜNCELLEMESİ VE YENİ PENCERELER	41
13.	HTML ÖGELERİNİN VE ÖZNİTELİKLERİNİN LİSTESİ	41
	ÖGELER.....	41
	ÖZNİTELİKLER	43

1.Doküman yapısı ve MetaData

İçerik geliştiriciler yapısal biçimlendirmeyi kullanmalı ve bunu belirlemelere göre yapmalı. Yapısal öğeler ve öznitelikler dokümanlardaki uyumu artırır ve indexleme araçları, arama motorları, veritabanları için tabloları seçen programlar, belge hakkında genel bilgileri kapsayan head(başlık) bölümünün öğelerini kullanan yönlendirme araçları, ve metni bir dilden başka bir dile çeviren otomatik çeviri programları vb.'leri diğer araçlar için bilgi sağlar.

1.1 MetaData (ÜstVeri)

Bu bölümdeki kontrol noktaları:

- [13.2](#) Sayfalar ve siteler için anlamlı bilgi eklemek için metadata (ÜstVeri) sağlayın. **[Öncelik 2]**

Bazı yapısal öğeler, doküman hakkında kendi kendine bilgi sağlar. Bu, doküman hakkında "metadata" olarak adlandırılır- metadata, veri hakkında bir bilgidir. Ustaca hazırlanmış bir metadata(üstveri), kullanıcıya önemli bir oryantasyon sağlayabilir. Bir doküman içeriği hakkında yararlı bilgi sağlayan HTML öğeleri :

1.1.1 TITLE: Dokümanın başlığı

Bir dokümanda sadece bir kez görülen "TITLE" öğesi, her bir HTML4.01 öğesinde görülen "title" özniteliğinden farklıdır. İçerik geliştiriciler, HTML 4.01'in tayinlerine göre "title" özniteliğini kullanmalıdırlar. Örneğin; "title" özniteliği, bağlantının açılacağı sayfa hakkında bilgi veren ("target" parametresi) linklerle birlikte kullanılmalıdır.

1.1.2 ADDRESS öğesi

Bu öğe, sayfanın yaratıcısı hakkında bilgi sağlamak için kullanılır.

1.1.3 META öğesi

Bu bölümdeki kontrol noktaları:

- [7.4](#) Kullanıcı araçları yenilenmeyi durduracak yeteneği sağlayıncaya kadar periyodik olarak otomatik yenilenen sayfalar yaratmayın. ***Ek:İçerik geliştiriciler bazen kendiliğinden yenilenen ya da kullanıcıların isteğinden bağımsız değişen sayfalar yaratırlar. Bu otomatik yenilenme, bazı kullanıcılar için oldukça zihin karıştırıcı olabilir.*** **[Öncelik 2]**
- [7.5](#) Kullanıcı araçları oto-yönlendirmeyi durduracak yeteneği geliştirinceye kadar, sayfaları otomatik yönlendirmek için biçimlendirme kullanmayın. **[Öncelik 2]**

Bu öğe, anahtar kelimeler içeren bir doküman için metadatayı ve yazar hakkındaki bilgiyi tayin edebilir. Lütfen "META"nın neden oto yenileme ya da yönlendirme sayfalarında kullanılmaması gerektiğine dair bilgiyi otomatik sayfa yenileme kısmına bakın.

Kullanıcının sayfası düzenli olarak zaman zaman değişir. İçerik geliştiriciler bu tekniği "push" tekniğini taklit etmek için kullanmamalıdırlar. ***Ek Bilgi:** Günlük yaşantımızdaki uygulamalara baktığımızda kullanıcılara faydalı olan birçok uygulamanın bulunduğunu bilir ancak erişilebilir olduğunu veya değiştiğini bilememekten dolayı göremeyiz. Bu tür bilgilerin kullanıcılara belirli aralıklarla itilmesi (**push**), kullanıcılara bu bilginin erişilebilir olduğunun bildirilmesi gerekir. Örnek vermek gerekirse bir e-mail almak, yatırım borsası bilgilerinin gelmesi vb'leri. İtici (**pull**)*

teknolojiler bir istemci/sunucu modele dayanmaktadır. Ancak sunucu bilgiyi göndermeden istemciden açık bir talepte bulunmamaktadır. Başka bir deyişle çekici işlemler daima istemci tarafından başlatılmasına rağmen itici işlemler daima sunucu tarafından başlatılmaktadır. Geliştiriciler, kullanıcının bir sayfayı okumayı ne kadarlık süre için talep edeceği konusunda kehanette bulunamazlar. Vaktinden önce olan veya gelişen sayfa yenilenmesi kullanıcının zihnini karıştırabilir. İçerik geliştiriciler, periyodik sayfa yenilemelerinden kaçınmalı ve kullanıcılara görmek istedikleri en son bilgiyi seçmeleri için izin vermelidirler.

“META” ögesi kullanılmış olan aşağıdaki HTML örneği kullanıcıyı kısa bir aradan sonra bir sayfadan diğer bir sayfaya geçirir. Bununla beraber kullanıcılar standart olmayan bu biçimlendirmeler ile yönlendirilmemeliler. Bu, kullanıcıların zihnini karıştırır ve ziyaret edilen sayfanın browser'ının geçmişini de karıştırabilir.

1.1.4 !DOCTYPE deyimi

Bu bölümdeki kontrol noktaları:

- [3.2](#) Yayınlanmış biçimsel gramerler için geçerliği denetlenmiş dokümanlar yaratın. **[Öncelik 2]**

Yayınlanmış bir biçimsel gramer için geçerliğin onaylanması ve dokümanın başında geçerliliğin belirtilmesi, kullanıcıya dokümanın yapısını bilmeye izin verir. Bu, üstelik kullanıcı araçlarına, ihtiyaç duyduğunda anlambilimi nerede arayacağı bilgisini de sağlar.

1.1.5 [LINK](#) ögesi ve gezinim araçları

Bu bölümdeki kontrol noktaları:

- [13.9](#) Çoklu sayfalar içeren dokümanlar gibi doküman derlemleri hakkında bilgi sağlayın. **[Öncelik 3]**

İçerik geliştiriciler, yönlendirme mekanizmasını ve organizasyonunu tanımlamak için (HTML4-Bölüm 6.12 de işaret edilen) [LINK](#) ögesini ve link çeşitlerini kullanmalıdırlar. Bazı kullanıcı araçları gezinim araçlarını sentezle birleştirebilir ya da böyle biçimlendirmelere dayalı dokümanların istenilen kısmının çıktısını almaya izin verebilir.

***Not:** [Rel](#) ve [Rev](#); link ile ilişkiyi belirler. [REL](#) (Relationship to link)*

[REV](#) (Relationship from link)

```
<LINK REL=Glossary HREF="foo.html">
```

```
<LINK REV=Subsection HREF="bar.html">
```

1.1.6 [LINK](#) ögesi ve alternatif dokümanlar

Bu bölümdeki kontrol noktaları:

- [6.5](#) Dinamik içeriğin erişilebilir olmasını ya da alternatif bir sunum ya da sayfa sağlamasını garanti edin. **[Öncelik 3]**

[Link](#) ögesi alternatif dokümanları işaret etmek için kullanılmış da olabilir. Browserlar(tarayıcılar) kullanıcı browserının(tarayıcısının) tip ve tercihlerine dayalı alternatif bir sayfayı otomatik olarak yüklemelidir.

1.2 Yapısal gruplama

Bu bölümdeki kontrol noktaları:

- [12.3](#) Uygun olan yerlerde büyük bilgi bloklarını daha idare edilebilir parçalara bölün. **[Öncelik 2]**
- Anlamsal birimler içindeki gruplar halindeki form kontrolleri için FIELDSET kullanın ve grupları LEGEND ögesi ile tanımlayın.
- Uzun bir menü seçenekleri listesini daha küçük gruplar halinde organize etmek için OPTGROUP kullanın. *OPTGROUP; bir formda menü seçeneği grubunu içerir.*
- Cetvel şeklindeki datalar için tablolar kullanın ve tabloyu CAPTION ile tanımlayın.
- Tablonun kolon ve satırlarını THEAD, TBODY, TFOOT and COLGROUP ile gruplayın.
- Listeleri UL, OL ve DL kullanarak iç içe hale getirin. Yapılandırılmış dokümanlar yaratmak için bölüm başlıklarını (H1,..., H6) kullanın.
- Metindeki paragraflar arasına satır boşluklarını P ögesi ile verin.
- İlişkili linkleri gruplayın.

Bütün bu gruplama mekanizmaları doğal ve uygun olduğu zamanlarda kullanılmalı, örneğin; bilgi mantıksal gruplar için uygun olduğunda. İçerik geliştiriciler rastgele gruplar oluşturmamalıdır, bu bütün kullanıcılar için zihin karıştırıcı olacaktır.

1.2.1 Bölüm başlıkları

Bu bölümdeki kontrol noktaları:

- [3.5](#) Başlık-üstbilgi öğelerini dokümanın yapısını ifade etmek için kullanın ve bu öğeleri tayinlere göre yapın. **[Öncelik 2]**

Uzun dokümanlar genellikle farklı kısımlara bölünür, bölümler alt başlıklara sahiptir ve alt başlıklar çeşitli parçalara, parçalar da paragraflara, vb'ne bölünür. Bu bilginin anlamsal yığınları dokümanın yapısını oluşturur.

Bölümler HTML başlık öğeleri ile ortaya koyulmalıdır (H1-H2). Diğer biçimlendirme, sunumu iyileştirmek için bu öğeleri tümleyici olabilir, örneğin; düşey doğrultuda bir çizgi yaratmak için kullanılan HR ögesi. Fakat görsel sunum dokümanın bölümlerini teşhis etmek için yeterli değildir.

Bazı kullanıcılar dokümanı başlıklarını gezinerek gözden geçirdikleri için doküman yapısını ifade etmek başlıkları uygun bir şekilde kullanmak önemlidir. Kullanıcılar başlık öğelerini uygun şekilde düzenlemelidirler. Örneğin HTML'de H2 öğeleri H1 öğelerini, H3 öğeleri de H2 öğelerini takip etmeli. İçerik geliştiriciler seviyeleri atlamamalıdır, örneğin H1'den H3'e. Font etkilerini yaratmak için başlıkları kullanmayın; font stillerini değiştirmek için stil sayfalarını kullanın.

2.Dil Bilgisi

2.1 Dildeki deęişiklikleri tanımlama

Bu bölümdeki kontrol noktaları:

- [4.1](#) Bir doküman metninin doğal dilindeki deęişiklikleri ve herhangi bir metnin eşdeęerlerini açık bir şekilde teşhis edin, örneęin resim altı başlıklar gibi. [**Öncelik 1**]

Eęer bir sayfada farklı dillerden birkaçını kullandıysanız, “lang” öznitelięinin kullanımı ile açık bir şekilde teşhis edilen dildeki herhangi bir deęişiklikten emin olun.

***EK:** <Q> etiketi, doküman görüntülenirken kendi paragrafında görünmesi gereken kısa bir alıntı metni içerir. etiketi, bir dokümanın küçük bir kısmını kendi bloęu içine alır. <DIV> gibi, içerięi içeride organize etmekte kullanılır ve niteliklerle birlikte kullanılarak o kısmın sunulma tarzını deęiştirebilir.*

Dildeki deęişikliklerin tanımlanması birkaç neden için önemlidir:

1. Dokümanı braille alfabesi ile okuyan kullanıcılar dilde deęişiklięin meydana geldięi yerlerde doğru karakterler üretecek braille software dönüşümünü garanti etmek için uygun kontrol kodlar(biçimlendirmeleri)ni deęiştirebileceklerdir. Bu kontrol kodları kullanıcının zihnini karıştıran, kodların üretilmesinden doğan braille kısaltmalarını da önler. Braille kısaltmaları genellikle bir çok hücrede fazlaca görünen karakter gruplarını tek bir hücrede birleştirebilir. Örneęin; “ing” genellikle 3 hücreyi kaplarken (her bir karakter için) tek bir hücreye daraltılabilir.
2. Benzer olarak, çoklu dillerde “speak” olarak adlandırılan konuşma birleştirciler, uygun söyleyiş ile uygun telaffuzda bir heceyi vurgulu olarak okuyan bir metin üretebileceklerdir. Eęer deęişiklikler işaretlenmezse, vurgulayıcı kelimeleri çalıştıęı basit dilde en iyi şekilde söylemeyi deneyecektir. Böylece, Fransızca kelime car için “voiture” temel dil olarak İngilizceyi kullanan bir konuşma birleştirci tarafından “voter” olarak telaffuz edilecektir.
3. Kendileri diller arasında dönüşüm yapamayan kullanıcılar, çeviri makineleri ile bilindik olmayan dilleri dönüştürme yetisine sahip olacaklardır.

2.2 Temel dilin tanımlanması

Bu bölümdeki kontrol noktaları:

- [4.3](#) Dokümanın temel doğal dilinin tanımlanması [**Öncelik 3**]

3.Metin Biçimlendirilmesi

3.1 Vurgu

Bu bölümdeki kontrol noktaları:

- [3.3](#) Sayfa düzenini ve sunumun kontrol için stil sayfaları kullanın [**Öncelik 2**]

Vurgu biçimlendirmeleri için uygun HTML öğeleri kullanılmalıdır, EM ve STRONG gibi. Bold(**B**) ve italik(*I*) öğeleri kullanılmamalıdır. Bunlar, görsel bir sunum efekti yaratmak için

kullanılır. EM ve STRONG öğeleri, ekranda imge oluşturmanın çeşitli yollarından biri olan yapısal vurgu gösterimi için dizayn edilmiştir.

3.2 Kısa adlar ve kısaltmalar

Bu bölümdeki kontrol noktaları:

- [4.2](#) Her bir kısaltmanın ya da kısa adın açıklamasını, dokümanın içinde ilk olarak yer aldığı yerde kesinlikle belirtin. [**Öncelik 3**]

3.3 Tırnaklar “”

Bu bölümdeki kontrol noktaları:

- [3.7](#) Tırnakları işaretleyin. Girintili yazma gibi formatlama efektleri için tırnak imi biçimlendirmesini kullanmayın. [**Öncelik 2**]

Not: <BLOCKQUOTE> etiketi, doküman görüntülenirken kendi paragrafında görünmesi gereken bir alıntı metin bloğunu içerir. Daha kısa alıntılar <Q> etiketinin içinde kullanılabilir. <BLOCKQUOTE> etiketi, bir alıntı söz konusu olmasa da, bir metin bloğuna girinti vermek için yaygın olarak kullanılmaktadır. Bu kullanım tarzı, HTML 4.0 ile eskimiştir ve aynı etkinin sağlanması için Style Sheet kullanımı önerilir.

3.4 Görüntüler yerine biçimlendirme ve stil sayfalarının kullanılması

Bu bölümdeki kontrol noktaları:

- [3.1](#) Uygun bir biçimlendirme dili var olduğu zaman, bilgiyi ifade etmek için imgelerin aksine biçimlendirme kullanın. [**Öncelik 2**]

Mümkün olan yerlerde örneğin bir matematiksel eşitlikte, biçimlendirme ve stil sayfalarının kullanılması aşağıdaki nedenlerden dolayı erişilebilirliği artırır:

- Metin, konuşma ya da braille alfabesi olarak yorumlanmış ya da büyütülmüş olabilir.
- Araştırma teknikleri metin bilgilerini kullanabilir.
- **Not:(Ek: TeX:'tek' (hatta teq) şeklinde okunan, daha çok akademik ortamlarda kullanılan, matematiksel ifadeler ve Türkçe karakterler dışında özel oluşturulmuş sembolleri de yazabilme kabiliyeti olan, içinden komut verilebilen dil.)** Daha karmaşık eşitlikleri MathML ya da TeX ile biçimlendirin. MathML oldukça erişilebilir dokümanlar yaratmak için kullanılabilir fakat TeX kadar kullanışlı ya da geniş çapta desteklenebilir değildir.
- Eşitliğin bir metin tanımlamasını sağlayın ve mümkün olan yerde matematiksel sembolleri yaratmak için ilgili varlıksal karakterleri kullanın. Eşitlik bir ya da daha fazla imge ile temsil ediliyorsa, bir eşitlik metni sağlanmalıdır.

TeX genel olarak Web'de yayınlanmak için HTML formatına dönüştürülmüş teknik kağıtların (akademik sayfaların-makalelerin) yaratılmasında kullanılır. Bununla beraber dönüştürücüler, imgeler üretmeye eğilimlidirler ve şiddetle karşı çıkılan biçimlendirmeleri ve sayfa görünümüleri için tabloları kullanırlar. Sonuç olarak içerik sağlayıcılar aşağıdakileri yapmalıdırlar:

1. Web de kullanışlı orijinal bir TeX dokümanı hazırlayın. ASTER "[Audio System For Technical Readings](#)"olarak adlandırılan TeX ve LaTeX dokümanların işitme duyusu

ile ifadesini yaratan bir sistem vardır. IBM TeX/LaTeX dokümanlarını okuyan Netscape ve Internet Explorer için uyumlu bir ek yazılıma sahiptir. Bu araçlar aslında İngilizce ortamlarda çalışır ve asıl dili İngilizce olmayan konuşma birleştiricileri ile o kadar da iyi çalışmayabilirler.

2. Dönüştürme işlemi ile yaratılan HTML'in erişilebilir olduğunu garanti edin. Eşitliğin tekil bir tarifini sağlayın.

3.5 Diğer yapısal 8 öge

CITE

Bu etiket, bir eserden diğer esere aktarılan bölümü veya benzer türde başka bir kaynağa referansı içerir.

DFN

Bu etiket, bir tanım ya da tanımlanan bir şeyi olan metni işaret eder.

CODE

Bu etiket, bir bilgisayar programlama dilindeki kaynak kodu veya benzer bilgiyi gösteren metni içerir.

SAMP

Bu etiket, bir bilgisayar programı, bir script veya benzer bir prosesin çıktısındaki metni göstermekte kullanılır.

KBD

Bu etiket, bir bilgisayar programında kullanıcı tarafından girilen bilgileri içerir.

VAR

Bu etiket, bir değişken ismini veya bir bilgisayar programında kullanılan bir komut satırı ifadesini temsil eden metni içerir.

INS

Bu etiket, dokümanın önceki düzenlenmesinden sonra eklenen bir doküman kısmını içerir.

DEL

Bu etiket, dokümanın eski düzenlemelerinden sonra silinen bir kısmını içerir. Bu etiketin içindeki metin genellikle üstü çizili karakterlerle gösterilir.

4.Listeler

Bu bölümdeki kontrol noktaları:

- [3.6](#) Liste ve liste öğelerini uygun bir şekilde biçimlendirin. **[Öncelik 2]**

HTML listeleme öğelerinden DL, UL ve OL sadece liste yaratmak için kullanılmalı, girintili yazma gibi formatlama efektleri için kullanılmamalıdır.

1.
 - 1.
 2.
 - 1.
 - 3.
2.
 - 1.

Düzenlenmiş listeler görsel olmayan kullanıcılara yardım eder. Görsel olmayan kullanıcılar listelerin içinde kaybolabilir, özellikle iç içe yuvalanmış ve her bir liste ögesi için özel iç içe geçiş seviyeleri belirtilmeyen listelerde. Kullanıcı araçları liste şartlarını açık bir şekilde tanımlayan bir varlık sağlayıcaya kadar, içerik geliştiriciler kendi listelerinde akışa göre ipuçlarına yer vermelidirler.

Not: Aşağıdaki STYLE kullanımını explorer desteklemiyor. Fakat Netscape destekliyor.

```
<HTML>
<STYLE type="text/css">
  UL, OL { counter-reset: item }
  LI { display: block }
  LI:before { content: counters(item, "."); counter-increment: item
}
</STYLE>
<BODY>
<ol>
<LI>.aaaaaaa
<LI style:LI:before>.bbbbbbb
  <ol>
  <LI>ccccccc
  <LI style:LI:before>.ddddddd
    <ol>
    <LI style:LI:before>.eeeeeee
    <LI style:LI:before>.fffffff
    </ol>
  <LI style:LI:before>.hhhhhhh
  </ol>
<LI style:LI:before>.iiiiiii
<LI style:LI:before>.mmmmmmmm
</ol>
<BODY>
</HTML>
```

Aynı listeleme şekli aşağıdaki html kodları ile de uzun uzadıya yapılabilir.

```
<OL> <!-- (set item[0] to 0 -->
  <LI>item <!-- increment item[0] (= 1) -->
  <LI>item <!-- increment item[0] (= 2) -->
  <OL> <!-- (set item[1] to 0 -->
    <LI>item <!-- increment item[1] (= 1) -->
    <LI>item <!-- increment item[1] (= 2) -->
    <LI>item <!-- increment item[1] (= 3) -->
    <OL> <!-- (set item[2] to 0 -->
      <LI>item <!-- increment item[2] (= 1) -->
    </OL> <!-- ) -->
    <OL> <!-- (set item[3] to 0 -->
      <LI> <!-- increment item[3] (= 1) -->
    </OL> <!-- ) -->
    <LI>item <!-- increment item[1] (= 4) -->
  </OL> <!-- ) -->
  <LI>item <!-- increment item[0] (= 3) -->
  <LI>item <!-- increment item[0] (= 4) -->
</OL> <!-- ) -->
<OL> <!-- (reset item[4] to 0 -->
  <LI>item <!-- increment item[4] (= 1) -->
  <LI>item <!-- increment item[4] (= 2) -->
</OL> <!-- ) -->
```

Görmeyen kullanıcılar listenin nerede başlayıp nerede bittiği ve her bir maddenin nerede başladığını bilmede zorluklara sahip olabilir.

4.1 Listelerin madde imlerini değiştirmek için stil sayfalarının kullanımı

LI ögesi ile yaratılmış düzenlenmemiş liste öğelerinin madde imlerini değiştirmek için stil sayfalarını kullanın. Eğer bir madde imi görüntüsü yüklenemiyorsa, CSS’de yedek bir madde imi stili tayin etmek mümkündür, “disc” gibi.

4.1.1 Madde imi gibi kullanılan görüntüler

Bu bölümdeki kontrol noktaları:

- [1.1](#) Metin olmayan her bir öğe için eşdeğer bir metin sağlayın. Bu; görüntüleri, semboller içeren bir metnin grafiksel sunumlarını, görüntü haritası bölgelerini, animasyonları, appletler ve diğer programa ait nesnelere, çerçeveleri, scriptleri, liste madde imleri olarak kullanılan görüntüleri, grafiksel butonları, videoları, bağımsız ses dosyalarını, görüntünün ses izleri vb.ni içerir. **[Öncelik 1]**
- DL, DT ve DD ile yaratılmış tanımlama listelerinde imgeleri madde imi olarak kullanmaktan kaçınin. Bununla beraber, eğer bu metod kullanılırsa, imgeler için bir eşdeğer metin sağladığından emin olun.

5.Tablolar

5.1 Veri tabloları

- Özet bilgiler verme
- Satır ve sütun bilgisi tanımlama

5.1.1 Özet bilgiler verme

Bu bölümdeki kontrol noktaları:

- [5.5](#) Tablolar için özetler verme. **[Öncelik 3]**
- [5.6](#) Başlık etiketleri için kısaltmalar verme. **[Öncelik 3]**
- CAPTION ögesi ile bir (resim altı) başlık verin. Bir tablo başlığı, 1-3 tümce ile tablonun doğasını-içeriğini tanımlar.

Bir başlık her zaman gerekli olmayabilir.

- Eğer bir başlık verilmemişse, bir kaç kelime ile tablonun doğasını tanımlamak için TABLE ögesinde “title” özneliğini kullanın.
 1. “summary” özneliği ile bir özet sağlama. Özetler, özellikle görsel olmayan okuyucular için -görme engelliler- faydalıdır. Hücreler arasındaki ilişkilerin özeti, özellikle yuvalanmış başlıklara sahip tablolar için önemlidir. Çoklu sütunlar ya da satırlar arasında açıklık olan hücreler, ya da tablonun yapı analizinden elde edilen diğer ilişkiler açık olmayabilir fakat tablonun görselleştirilmesi kolay anlaşılır olabilir. Özet tablonun geçerli dosyanın

bağlamına nasıl tutturulmuş olduğunu tanımlayabilir. Eğer verilmiş bir başlık yoksa, bu bir özet verme için daha kritiktir.

- Başlık hücresinde-TH-“abbr”-kısaltma özniteliği ile tablo başlıkları için özlü değişiklikler sağlayın. *Not: Uzun uzun başlıklar yazmak yerine “abbr-kısaltma” özniteliği ile uzun bir metin kısa şekliyle de verilebilir.* Bunlar gelecekteki konuşma teknolojileri için kısmen faydalı olabilecektir.

5.1.2 Tanımlayıcı satırlar ve sütun bilgisi

Bu bölümdeki kontrol noktaları:

- [5.1](#) Veri tabloları için tanımlayıcı satır ve sütun başlıkları. **[Öncelik 1]**
- [5.2](#) Satır ya da sütun başlıkları için bir veya daha fazla mantıksal düzeye sahip veri tabloları için, veri hücreleri ve başlık hücrelerini ilişkilendirmek için biçimlendirme kullanın. **[Öncelik 1]**
- Satır ve kolon gruplarının yapısını tanımlayın. THEAD tablo başlığının her sayfada tekrarlanması için, TFOOT tablo alt başlığının her sayfada otomatik olarak tekrarlanması için ve TBODY diğer satır grupları için, COLGROUP & COL diğer kolon grupları için.
- Gelecekteki browserlar ve destek teknolojileri bir tablodaki verileri kategorilerden filtre ederek seçebilsin diye tablo öğelerini “scope”, “headers” ve “axis” öznitelikleri ile etiketleyin.
- Destek teknolojileri tablo olarak tanıyabilsin diye bir metnin çizelgesel şablonunu yaratmak için PRE tagını kullanmayın, TABLE öğesini kullanın.
- Tablo başlıkları hakkında bilgi için tablo başlık algoritmasını işaret edin.

“headers” özniteliği geçerli veri hücresi ile birlikte başlık hücrelerinin bir listesini belirtir. Bu, her bir başlık hücresinin bir “id” özniteliğine sahip olmasını gerektirir.

“headers” özniteliğinden “scope” özniteliği de kullanılabilir. “Scope” değerlerden birine sahip olmalıdır: “row”, “col”, “rowgroup”, ya da “colgroup”. “Scope” başlık hücresi ile ilişkilendirilmiş bir veri hücresi setini belirtir. Bu metod özellikle basit tablolar için faydalıdır. “headers” ve “scope” öznitelikleri arasındaki seçim, tablonun karmaşıklığına bağlıdır. Bu seçim biçimlendirmede başlık ve veri hücreleri arasındaki ilişkiler kesinleştiği sürece çıkışı-tablonun görünüşünü etkilemez.

5.2 Sayfa görünümü için tabloların kullanılması

Bu bölümdeki kontrol noktaları:

- [5.3](#) Lineer hale getirildiğinde tablo anlamlı hale gelmiyorsa, tabloları yerleşim için kullanmayın. **[Öncelik 2]**
- [5.4](#) Eğer yerleşim için bir tablo kullanılmışsa, görüntüsel formatlama amaçlı herhangi bir yapısal biçimlendirme kullanmayın. **[Öncelik 2]**

Sayfa görünümü ve konumlandırma için sayfayı hazırlayanlar stil sayfaları kullanılmalıdır. Bununla beraber, yerleşim için tablo kullanımı gerekli olduğunda, tablo okunabilir bir sırada doğrusal olmalıdır. Tablo doğrusallaştırıldığında hücrelerin içerikleri biri diğerinden sonra paragraflar serisi haline gelir (sayfada aşağıya doğru). Sayfanın lineerleştirilmesinden sonra anlamlı hale gelmesi için hücreler, belirli bir sırada okunduğunda anlamlı olmalı ve yapısal öğeler içermeli.

Bir şablon yapı yaratmak için tablolar kullanıldığında görsel formatlama yaratmak için yapısal biçimlendirme kullanmayın. Örneğin; TH (tablo başlığı) ögesi genellikle merkezde ve kalın olarak görüntülenir. Eğer bir hücre verinin bir satır ya da kolonu için gerçekten bir başlık değilse stil sayfalarını ya da formatlayıcı elementlerin özniteliklerini kullanın.

5.3 Tabloların lineer hale getirilmesi

Bu bölümdeki kontrol noktaları:

- [10.3](#) Destek teknolojilerini içeren kullanıcı araçları doğru bir şekilde yan yana metin oluşturuncaya kadar, metinle aynı paralellikte tasarlanan bütün tablolar için aktif olan sayfada ya da açılan diğer bir sayfada doğrusal bir metin alternatifi sağlayın. **[Öncelik 3]**

Yerleşim için kullanılan tablolar, hücre metni bitmiş gibi görünen yerlerde HTML kaynağını okuyamayan ya da tablo hücreleri arasındaki gezinime izin vermeyen browserlara sahip eski ekran okuyucuları için problemler yaratır. Bu ekran okuyucular sayfayı karşı tarafa geçerek yani aynı satırda farklı sütunlarda olan cümleleri tek bir satırmış gibi okuyacaktır.

Örnek: Reddedilen bir tablo ise ekranda aşağıdaki gibi görünecektir:

Az gören bir kişi fare kullanmakta sorun yaşar. Çünkü fare kullanımı elin ve gözlerin koordineli çalışmasını gerektirir. Duymayan ve	işitme güçlüğü yaşayan kişiler web sayfasında yer alan sesli bilgilerin görsel olarak da ifade edilmesine ihtiyaç duyarlar.
--	---

Ekran okuyucu tarafından bu şekilde okunmalıydı:

Az gören bir kişi fare kullanmakta sorun yaşar. Çünkü fare kullanımı elin ve gözlerin koordineli çalışmasını gerektirir. Duymayan ve işitme güçlüğü yaşayan kişiler web sayfasında yer alan sesli bilgilerin görsel olarak da ifade edilmesine ihtiyaç duyarlar.

Sayfada şablon yapı kullanılarak bir tablonun lineer hale getirilmesi genellikle oldukça kolaydır. Bunu yapmak için birkaç yöntem vardır, ve bazı browserlar ve ekran okuyucuları için tabloların lineer hale getirilmesi daha yaygın bir hale gelmeye başlamıştır.

Örneğin; sütunun yerleşim sırasını belirtmek. Yazılan yöndeki doğal dil sütununun yerleşimini ve böylece lineer hale getirilmiş tablonun sıralamasını etkileyebilir. “dir” özniteliği kolonun yerleşimini belirtir. Örneğin; dir=”rtl” specifies right-to-left layout.

5.4 Tablolar için arkaplan ile uyumsuzluk sorunu

HTML 3.2 tarayıcısında bir **TFOOT** öğesinin sütunları tablo gövdesinden önce görünecektir. **Not:** Bu etiket, tablodaki tüm diğer satırların altında alt satır olarak görülen bir tablo satırını içerir. <THEAD> etiketi gibi, bu alt satır tablodaki belirli sütunlar hakkında bilgi sağlamakta kullanılabilir. İlişkili olduğu <TBODY> ve <TFOOT> tablo satırlarıyla aynı sayıda sütuna sahip olmalıdır. Bir tablonun alt satırının, o tablodaki <TBODY> etiketlerinden önce yer alması gerekir.

6.Linkler

6.1 Link metni

Bu bölümdeki kontrol noktaları:

- [13.1](#) Her bir hedef linki açık bir şekilde tanımlayın. **[Öncelik 2]**

İyi bir link metni aşırı derecede genel olmamalı, “buraya tıklayın” şeklini kesinlikle kullanmayın. Bu, link takip edildiğinde ne bulunacağı konusunda hiçbir şey söylemez. “Buraya tıklayın” yerine link metni, “deniz ayıları hakkında daha fazla bilgi” şeklinde ya da “sayfanın yalnızca metin versiyonu” gibi link hedefinin doğasını işaret etmeli.

Açık bir link metnine ilaveten içerik geliştiriciler, açık ve doğru bir şekilde linkin hedefini tanımlayan “title” özniteliğinin değerini de belirtebilir.

Eğer bir sayfadaki birden fazla link aynı link metnini paylaşırsa; bütün linkler aynı kaynağı işaret etmelidir. Böyle bir tutarlılık iyi bir erişilebilirlikle sayfa tasarımına yardım edecektir.

Eğer bir ya da daha fazla link farklı hedefleri işaret ediyor fakat aynı link metnini paylaşıyorsa, her bir A öğesinin “başlık” özniteliği için farklı bir değer ile tayin edilen linkleri ayırın.

6.1.1 Metin için link olarak kullanılan görüntüler

Bu bölümdeki kontrol noktaları:

- Metin olmayan her bir öğe için eşdeğer bir metin sağlayın. Bu; görüntüleri, semboller içeren bir metnin grafiksel sunumlarını, görüntü haritası bölgelerini, animasyonları, appletler ve diğer programa ait nesnelere, çerçeveleri, scriptleri, liste madde imleri olarak kullanılan görüntüleri, grafiksel butonları, videoları, bağımsız ses dosyalarını, görüntünün ses izleri vb.ni içerir. **[Öncelik 1]**

6.2 Grublama ve atlama linkleri

Bu bölümdeki kontrol noktaları:

- [13.6](#) İlişkili linkleri gruplayın, kullanıcı araçları için grupları tanıtır ve kullanıcı araçları böyle yapınca kadar grubu atlamak için bir yol sağlayın. **[Öncelik 3]**
- [10.5](#) Destek teknolojileri içeren kullanıcı araçları kesin olarak bitişik linkler oluşturuncaya kadar bitişik linkler arasında link olmayan, basılabilir karakterlere (boşluklarla çevrilmiş) yer verin. **[Öncelik 3]**

Linkler mantıksal kümeler halinde gruplandırıldığında, bir birim gibi işaretlenmelidirler. Gezinme çubukları genellikle birisinin sayfada ilk karşılaştığı şeydir. Konuşma birleştiricileri kullanıcıları için, bunun anlamı sayfanın ilgilenilen içeriğine ulaşmadan önce her sayfadaki birtakım linki duyabilme imkanına sahip olmaktır.

- Kullanıcılara gezinme linkleri kümesini atlamaya izin veren bir linki dahil edin.
- Kullanıcılara gezinme linklerini saklamaya izin verecek bir stil sayfası sağlayın.
- Linkleri gruplamak için HTML 4.01 MAP ögesini kullanın, sonra grupları "title" özneliği ile tanımlayın.

Gelecekte, kullanıcı araçları kullanıcılara gezinme çubukları gibi öğeleri atlama izni verebilir.

6.3 Klavye erişimi

Bu bölümdeki kontrol noktaları:

- [9.4](#) Linkler, form kontrolleri ve nesnelere boyunca mantıklı bir sıra yaratın. **[Öncelik 3]**
- [9.5](#) Önemli linkler, form kontrolleri ve kontrol grupları için klavye kısa yolları sağlayın. **[Öncelik 3]**

Bir sayfanın aktif öğelerine klavyeden erişim işaret edici aletler kullanamayan bazı kullanıcılar için önemlidir. Kullanıcı araçları kesin işlemler için kör klavye vuruşlarına izin veren özellikler içerebilir. HTML 4.01 içerik geliştiricilerine dokümandaki klavye kısa yollarını "accesskey" özneliği yoluyla belirtmeye izin verir.

6.4 Bağlantılar ve hedefler

Bu bölümdeki kontrol noktaları:

- [10.1](#) Kullanıcı araçları kullanıcılara sonuç pencerelerini kapatma izni verinceye kadar pop-up yada diğer pencerelerin görünmesine neden olmayın ve kullanımda olan pencereyi kullanıcının bilgisi olmadan değiştirmeyin. **[Öncelik 2]**

7. Görüntüler ve görüntü haritaları

Takip eden bölümler GIF animasyonları gibi basit animasyonlar içeren görüntünün ve görüntü haritalarının erişilebilirliğini tartışıyor.

7.1 Görüntüler için kısa eşdeğer metinler

Bu bölümdeki kontrol noktaları:

- [1.1](#) Metin olmayan her bir öge için eşdeğer bir metin sağlayın. Bu; görüntüleri, semboller içeren bir metnin grafiksel sunumlarını, görüntü haritası bölgelerini, animasyonları, appletler ve diğer programa ait nesnelere, çerçeveleri, scriptleri, liste madde imleri olarak kullanılan görüntüleri, grafiksel butonları, videoları, bağımsız ses dosyalarını, görüntünün ses izleri vb.ni içerir. **[Öncelik 1]**
- IMG kullanıldığında “alt” özniteliği ile kısa bir metin eşdeğeri belirtin. Bu özniteliğin değeri “alt-text” olarak temsil edilir.

7.2 Görüntülerin uzun betimlemeleri

Bu bölümdeki kontrol noktaları:

- [1.1](#) Metin olmayan her bir öge için bir eşdeğer metin sağlayın. Bu; görüntüler, semboller içeren bir metnin grafiksel sunumları, görüntü haritası bölgeleri, animasyonlar, appletler ve diğer programa ait nesnelere, çerçeveler, scriptler, liste madde imleri olarak kullanılan görüntüler, grafiksel butonlar, videolar vb.ni içerir. **[Öncelik 1]**

Kısa bir eşdeğer metin bir görüntünün rolü ya da fonksiyonunu ifade etmek için layıkıyla yeterli olmadığı zaman, ilave olarak “[longdesc](#)” özniteliği ile dizayn edilmiş bir bilgi dosyası sağlayın.

“longdesc”i desteklemeyen kullanıcı araçları için grafiğin yakınında bir yere betimleyici bir link sağlayın.

OBJECT kullanıldığında daha uzun eşdeğer metni ögenin içeriğinde belirtin.

7.2.1 Görünmeyen d-linkleri

Bir görünmeyen d-link 1 pixel kadar küçüktür ya da değeri “D-link” veya “D” olan “alt” ögesinin saydam görüntüsüdür ya da A ögesinin içeriğinin bir kısmıdır.

7.3 Ascii art

NOT: [ascii art](#): ansi kullanılmadan harfler ve ascii sembollerle yapılan grafik çalışmalarına verilen isim. ansi renk kodları, ascii karakterler ve özel işaretlerle yapılan grafik çalışmalarına verilen isim.. bunların renksiz standard ascii kullananlarına [ascii_art](#) denir.

Bu bölümdeki kontrol noktaları:

- [13.10](#) Çoklu–satır ASCII art’ları atlayarak geçen bir araç sağlayın. **[Öncelik 3]**

ASCII art’dan kaçınınız ve görüntüler için eşdeğer bir metin sağlamak daha kolay olduğu için yerinde gerçek görüntüler kullanın.

Daha küçük ascii artlar için diğer bir seçenek “title” ile ABBR ögesini kullanmaktır.

Eğer ascii art karmaşık ise eşdeğer metnin ascii artı layıkıyla tanımladığını garanti edin.

ASCII art'ın yerine başka bir şey koymanın diğere bir yolu dil yedekleri kullanmaktır.

7.4 Görüntü haritaları

Bir görüntü haritası "aktif bölgeler"e sahip bir görüntüdür. Kullanıcı, bölgelerden birisini seçtiğinde bazı eylemler gerçekleşir- bir link takip edilebilir, sunucuya bilgi gönderilebilir vb. Erişilebilir bir görüntü haritası yapmak için içerik geliştiriciler, işaret edici bir cihazdan bağımsız olarak aktif hale getirilebilir görüntüsel bir bölge ile her bir eylemin bağlantılı olmasını garanti etmelidir.

Görüntü haritaları MAP ögesi ile yaratılır. HTML iki tür görüntü haritasına izin verir: istemci taraflı ve sunucu taraflı. Bütün görüntü haritaları için içerik geliştiriciler, eşdeğer bir metin sağlamalıdır.

İçerik geliştiriciler, sunucu taraflı görüntü haritalarındansa ("ismap" ile oluşturulmuş) istemci taraflı ("usemap" kullanılarak oluşturulmuş) görüntü haritaları yaratmalıdır. Çünkü sunucu taraflı görüntü haritaları özel bir giriş cihazı gerektirir. Eğer sunucu taraflı görüntüler kullanılsaydı (bölgenin geometrisi "shape" özneliği değeri ile sunulmadığı için) tasarımcılar aynı fonksiyonelliği ya da erişilebilir alternatif formatta bir bilgi sağlamak zorunda kalırlardı. Bunu yapabilmeyen bir yolu her bir link klavye ile gezinilebilinsin diye her bir aktif bölge için metne ait bir link sağlamaktır.

7.4.1 Metin eşdeğerleri için istemci taraflı görüntü haritaları

Bu bölümdeki kontrol noktaları:

- [1.1](#) Metin olmayan her bir öge için bir eşdeğer metin sağlayın. Bu; görüntüleri, semboller içeren bir metnin grafiksel sunumları, görüntü haritası bölgeleri, animasyonlar, appletler ve diğere programa ait nesnelere, çerçevelere, scriptlere, liste madde imleri olarak kullanılan görüntüler, grafiksel butonlar, videolar vb.ni içerir. **[Öncelik 1]**

7.4.2 İstemci taraflı görüntü haritaları için artık metin linkleri

Bu bölümdeki kontrol noktaları:

- [1.5](#) Kullanıcı araçları istemci taraflı resim haritaları linkleri için metin eşdeğerlerini oluşturana kadar, istemci taraflı resim haritasının her bir aktif bölgesi için gereğinden fazla metin linki sağlayın. **[Öncelik 3]**

Eşdeğer bir metin sağlamaya ilave olarak, metne ait artık linkler sağlayın. AREA'nın yerine A ögesi kullanılırsa içerik geliştiriciler aktif bölgeler tanımlayabilir ve aynı anda artık linkler sağlayabilir.

7.4.3 Sunucu ya da istemci taraflı resim haritaları

Bu bölümdeki kontrol noktaları:

- [9.1](#) Uygun bir geometrik şekil ile tanımlanmamış bölgeler hariç sunucu taraflı resim haritaları yerine istemci taraflı resim haritaları sağlayın. **[Öncelik 1]**

7.4.4 Sunucu taraflı resim haritaları

Bu bölümdeki kontrol noktaları:

- [1.2](#) Server taraflı resim haritasının her bir aktif bölgesi için gereğinden fazla link sağlayın. **[Öncelik 1]**

Sunucu taraflı bir resim haritasının kullanılması zorunlu olduğunda, içerik geliştiriciler resim haritası seçeneklerini veren alternatif bir liste sağlamalıdır.

- Eğer diğer yaklaşımlar erişilebilir resim haritaları yapmıyorsa erişilebilir olan alternatif bir sayfa yaratın.

Sunucu taraflı ya da istemci taraflı resim haritaları formlarda ifade butonları olarak kullanılabilir.

7.5 Görüntülerdeki renkler

Bu bölümdeki kontrol noktaları:

- [2.2](#) Renk kaybına sahip birisi tarafından ya da siyah-beyaz ekrandan izlendiğinde önalan ve arka plan renk kombinasyonunun uygun karşıtlığı sağladığından emin olun. **[Görüntüler için Öncelik 2, Metin için Öncelik 3].**

7.6 Hareketli görüntüler

Bu bölümdeki kontrol noktaları:

- [7.3](#) Kullanıcı araçları kullanıcılara hareket eden içeriği dondurma izni verinceye kadar sayfalardaki hareketlilikten kaçınin. **[Öncelik 2]**

8.Appletler ve programa ait diğer nesnelere

Bir dokümandaki uygulamada APPLET ya da OBJECT etiketlerinden biri kullanılacaksa OBJECT'in kullanılması tercih edilen metottür.

8.1 Appletler ve programa ait diğer nesnelere için metinler ve metin olmayan eşdeğerleri

Bu bölümdeki kontrol noktaları:

- [1.1](#) Metin olmayan her bir öge için eşdeğer bir metin sağlayın. Bu; görüntüleri, semboller içeren bir metnin grafiksel sunumlarını, görüntü haritası bölgelerini, animasyonları, appletler ve diğer programa ait nesnelere, çerçeveleri, scriptleri, liste madde imleri olarak kullanılan görüntüleri, grafiksel butonları, videoları, bağımsız ses dosyalarını, görüntünün ses izleri vb.ni içerir. **[Öncelik 1]**
- [6.2](#) Dinamik içerik için hazırlanmış olan eşdeğerlerin, dinamik içerik değiştiğinde güncellendiğinden emin olun. **[Öncelik 1]**

- [6.3](#) Scriptler, appletler ya da diğer programa ait nesnelere kapatıldığında ya da desteklenmediği zaman sayfaların kullanışlı olduğundan emin olun. Eğer bu mümkün değilse, alternatif erişilebilir bir sayfada eşdeğer bilgi sağlayın. **[Öncelik 1]**

8.2 Doğrudan erişilebilir appletler

Bu bölümdeki kontrol noktaları:

- [8.1](#) Scriptler ve appletler gibi doğrudan erişilebilir ya da destek teknolojileri ile uyumlu programa ait öğeler yaratın.

Eğer OBJECT ya da APPLET özniteliklerinden birisi ile yaratılmış bir applet alternatif formattaki bir kopyası yapılamayan kullanıcı etkileşimi sağlıyorsa, appleti doğrudan erişilebilir yapın.

İlgili kontrol noktaları:

- [3.4](#) İşaretleme dilindeki öznitelik ve stil sayfası özellik değerlerinde kesin birimler yerine göreliler kullanın. **[Öncelik 2]**
- [6.4](#) Scriptler ve appletler için olay işleyicilerin giriş aygıtından bağımsız olduğundan emin olun. **[Öncelik 2]**
- [6.5](#) Dinamik içeriğin erişilebilir ya da alternatif bir sunum ya da sayfa sağladığından emin olun. **[Öncelik 2]**
- [7.1](#) Kullanıcı araçları kullanıcılara titreşim uyarıları kontrol etmeleri için izin verinceye kadar titreşim uyarılara neden olan ekranlardan kaçının. **[Öncelik 1]**
- [7.2](#) Kullanıcı araçları kullanıcılara yanıp sönmeleri kontrol etmeleri için izin verinceye kadar yanıp sönmelere neden olan içeriklerden sakının. **[Öncelik 2]**
- [7.3](#) Kullanıcıları temsilcileri kullanıcılara hareketli içeriği dondurmaya izin verinceye kadar sayfalardaki hareketlilikten kaçının. **[Öncelik 2]**
- [7.4](#) Kullanıcı araçları yenilemeyi durdurma yeteneği sağlayıncaya kadar periyodik olarak otomatik yenilenen sayfalar yaratmayın. **[Öncelik 2]**
- [9.2](#) Kendi ara yüzüne sahip herhangi bir öğenin aygıttan bağımsız bir biçimde işletilmesini sağlayın. **[Öncelik 2]**
- ve [10.1](#) Kullanıcı araçları kullanıcılara sonuç pencerelerini kapatma izni verinceye kadar pop-up ya da diğer pencerelerin görünmesine neden olmayın ve kullanımda olan pencereyi kullanıcının bilgisi olmadan değiştirmeyin. **[Öncelik 2]**
- [1.4](#) Zamana dayalı herhangi bir çoklu ortam sunumu için (bir film ya da animasyon), sunumlar ile eşdeğer alternatifleri eşzamanlı kılın. **[Öncelik 1]**

9.Ses ve görüntü

9.1 Ses bilgisi

9.2 Çoklu ortam için metin eşdeğerleri

Bu bölümdeki kontrol noktaları:

- [1.1](#) Metin olmayan her bir öge için eşdeğer bir metin sağlayın. Bu; görüntüleri, semboller içeren bir metnin grafiksel sunumlarını, görüntü haritası bölgelerini, animasyonları, appletler ve diğer programa ait nesnelere, çerçeveleri, scriptleri, liste madde imleri olarak kullanılan görüntüleri, grafiksel butonları, videoları, bağımsız ses dosyalarını, görüntünün ses izleri vb.ni içerir. **[Öncelik 1]**
- Gerekli olduğunda, eşdeğer bir metin sayfasının anlaşılabilirliği için görsel bilginin seçilir kılınması sağlanmalı.

9.3 Gömülü çoklu ortam nesnelere

Netscape tarayıcısı ile geriye uyumluluk için OBJECT ögesinin içindeki özel EMBED ögesini kullanın.

10.Çerçeveler

Görme engeli olmayan kullanıcılar için çerçeveler bir sayfayı farklı alanlarda düzenleyebilir. Görme engelli kullanıcılar için ise çerçevelerdeki içerikler arasındaki ilişkiler diğer ifadeler boyunca da aktarılmalı.

Çerçevelerin FRAMESET, FRAME, IFRAME ögeleri ile gerçekleştirilmesi bugün birkaç nedenden ötürü şüphelidir:

- Script olmaksızın, çerçeveler fonksiyonel olarak tarayıcılar tarafından önerilen “önceki sayfayı” kesmeye eğilimlidirler.

URI ile bir framesetin “geçerli durumunu” işaret etmek imkansızdır. Bir frameset içeriği bir kez değiştirirse, orjinal URI etkili olmaz.

- Frame’in yeni bir tarayıcı penceresinde açılması kullanıcının aklını karıştırabilir ya da sadece kullanıcıyı sıkmaya yarar.

10.1 Çerçeve başlığının sağlanması

Bu bölümdeki kontrol noktaları:

- [12.1](#) Çerçeveyi tanıma ve gezinimi kolaylaştırmak için her bir çerçeveye başlık verin. Çerçevelere isim vermek için “title” özneliğini kullanın. **[Öncelik 1]**

10.2 Çerçeveler arası ilişkileri tanımlama

Bu bölümdeki kontrol noktaları:

- [1.1](#) Metin olmayan her bir öge için eşdeğer bir metin sağlayın. Bu; görüntüleri, semboller içeren bir metnin grafiksel sunumlarını, görüntü haritası bölgelerini, animasyonları, appletler ve diğer programa ait nesnelere, çerçeveleri, scriptleri, liste

madde imleri olarak kullanılan görüntüleri, grafiksel butonları, videoları, bağımsız ses dosyalarını, görüntünün ses izleri vb.ni içerir. **[Öncelik 1]**

- [12.2](#) Çerçevelerin amacını ve eğer çerçeve başlıkları yalnız başlarına aşıkara değil ise çerçevelerin birbirleri ile nasıl ilişkilendirileceğini tanımlayın. **[Öncelik 2]**

10.3 Çerçeveleri desteklemeyen tarayıcılar için yazım şekli

Bu bölümdeki kontrol noktaları:

- [1.1](#) Metin olmayan her bir öge için eşdeğer bir metin sağlayın. Bu; görüntüleri, semboller içeren bir metnin grafiksel sunumlarını, görüntü haritası bölgelerini, animasyonları, appletler ve diğer programa ait nesnelere, çerçevelere, scriptleri, liste madde imleri olarak kullanılan görüntüleri, grafiksel butonları, videoları, bağımsız ses dosyalarını, görüntünün ses izleri vb.ni içerir. **[öncelik 1]**
- [6.5](#) Dinamik içeriğin erişilebilir olduğundan ya da alternatif bir sunum ya da sayfa sağladığından emin olun. **[öncelik 2]**

10.4 Çerçeve kaynakları

Bu bölümdeki kontrol noktaları:

- [6.2](#) Dinamik sayfa değiştiğinde dinamik sayfa eşdeğerinin de güncellendiğinden emin olun. **[öncelik 1]**

İçerik geliştiriciler, sayfa içeriklerini ve çerçeveler arası ilişkileri daha hassas yapmak için çerçevelerin metin eşdeğerlerini sağlamalı. Bir frame'in içeriği değiştiğinde tanımlamanın da değiştirilmesi gerektiğine dikkat edin Bu, eğer bir img doğrudan çerçevenin içerisine yerleştirilmişse mümkün değildir. Bu yüzden içerik geliştiriciler her zaman bir HTML sayfasındaki çerçevenin kaynağını "src" etiketi ile vermelidir. Görüntüler HTML dosyasının içine yerleştirilebilir ve onların metin alternatifleri doğru bir şekilde geliştirilebilecektir.

10.5 Çerçeve hedeflerinin kullanılması

Bu bölümdeki kontrol noktaları:

- [10.1](#) Kullanıcı araçları kullanıcılara sonuç pencerelerini kapatma izni verinceye kadar pop-up yada diğer pencerelerin görünmesine neden olmayın ve kullanımda olan pencereyi kullanıcının bilgisi olmadan değiştirmeyin. **[Öncelik 2]**

İçerik geliştiriciler target="_blank" kullanarak çerçevenin yeni bir sayfada açılmasını temin etmekten kaçınmalılar.

10.6 Çerçeveler için alternatifler

Framelerin en yaygın kullanımlarından biri kullanıcının tarayıcı penceresini gezinim penceresi ve içerik penceresi olmak üzere iki kısma ayırmaktır.

Not. Gezinim mekanizması ya da diğer içerik sunucu taraflı içerikler yoluyla dokümana yerleştirilebilir.

10.6.1 İlişkili birimler ile çerçevelerin boyutlandırılması

Bu bölümdeki kontrol noktaları:

- [3.4](#) biçimlendirme dilindeki öznitelik değerlerindeki kesin birimlense bağlı değerleri ve stil sayfalarının özellik değerlerini kullanın. **[Öncelik 2]**

11.Formlar

11.1 Formlar için klavye erişimi

Bu bölümdeki kontrol noktaları:

- [9.4](#) Linkler, form kontrolleri ve nesnelere arasında mantıksal bir tab sırası yaratın. **[Öncelik 3]**
- [9.5](#) Önemli linkler (istemci tarafı görüntü haritaları içeren), form kontrolleri ve form kontrol grupları için klavye kısa yolları sağlayın. **[Öncelik 3]**

11.2 Form kontrollerini gruplama

Bu bölümdeki kontrol noktaları:

- [12.3](#) Büyük bilgi bloklarını doğal ve uygun olan yerlerde daha idare edilebilir gruplara bölün. **[Öncelik 2]**

Form kontrolleri FIELDSET ögesi ile mantıksal birimler halinde gruplanabilir ve bu birimler LEGEND ögesi etiketlenebilir.

11.2.1 Menü seçeneklerini gruplama

İçerik geliştiriciler bilgiyi doğal ve uygun olan yerlerde gruplamalıdır. Menü seçeneklerindeki uzun listeler için içerik geliştiriciler (OPTION ögesi ile tanımlanan) SELECT öğelerini OPTGROUP etiketi ile hiyerarşik bir sırada gruplamalıdır. Bir seçenekler grubunu etiketlemeyi OPTGROUP daki label özneliği ile tayin edin.

11.3 Form kontrollerini etiketleme

Bu bölümdeki kontrol noktaları:

- [12.4](#) Etiketleri açık bir şekilde kendi kontrolleri ile birleştirin. **[Öncelik 2]**
- [10.2](#) Kullanıcı araçları etiketler arasındaki birleştirmeleri ve form kontrollerini açıkça destekleyinceye kadar etiketlerin doğru bir şekilde yerleştirildiğinden emin olun. **[Öncelik 2]**

11.4 Grafikselle butonlar

Bu bölümdeki kontrol noktaları:

- [1.1](#) Metin olmayan her bir öğe için eşdeğer bir metin sağlayın. Bu; görüntüleri, semboller içeren bir metnin grafikselle sunumlarını, görüntü haritası bölgelerini, animasyonları, appletler ve diğer programa ait nesnelere, çerçevelere, scriptlere, liste

madde imleri olarak kullanılan görüntüleri, grafiksel butonları, videoları, bağımsız ses dosyalarını, görüntünün ses izleri vb.ni içerir. **[Öncelik 1]**

Dekoratif butonlar için görüntülerin kullanılması içerik geliştiricilere kendilerine ait benzersiz ve daha kolay anlaşılır formlar yapma izni verir. Bir buton için bir görüntünün kullanılması (örneğin INPUT ya da BUTTON öğeleri ile) doğal olarak erişilemez değildir-görüntü için eşdeğer bir metnin sağlandığı varsayılır.

Bununla beraber grafiksel bir form INPUT, ile yaratılan butonu teklif edilir, type="image" sunucu taraflı bir görüntü haritası çeşidini yaratır. Butona mouse ile tıklanıldığı zaman mouse kliklerinin x ve y koordinatları formun parçaları şeklinde sunucuya gönderilir.

Eğer sunucu yerleşime bağlı olarak farklı doğrultular alırsa bu grafiksel olmayan tarayıcı kullanıcıları için dezavantaj olacaktır. Bu sebepten dolayı tasarımcılar alternatif yaklaşımlar düşünmelidir:

- Tek bir grafiksel onay butonu yerine çoklu onay butonları kullanın. Tasarımcılar bu butonların yerleşimini kontrol etmek için stil sayfaları kullanabilir.
- Scriptler ile istemci taraflı bir görüntü haritası kullanın.

11.5 Özel kontroller için teknikler

Bu bölümdeki kontrol noktaları:

- [10.4](#) Kullanıcı araçları boş kontrolleri doğru bir şekilde tanıtıncaya kadar varsayılan düzenleyici kutular "edit boxes" ve metin alanları "text areas" dahil edin. **[Öncelik 3]**

"Onay" butonu olarak kullanılan görüntüler için eşdeğer bir metin sağlayın.

11.6 Formlar için arka plan ile uyumsuzluk sorunu

Bazı HTML 3.2 tarayıcılarında,

- BUTTON öğesi görülmez
- type="buton" ile INPUT bir metin giriş alanı görülecektir.

12.Scriptler

12.1 Scriptlerin zarif dönüşümü

Bu bölümdeki kontrol noktaları:

- [6.5](#) Dinamik içeriğin erişilebilir ya da sayfada alternatif bir sunum sağladığından emin olun. **[Öncelik 2]**

İçerik geliştiriciler, scriptlerin devre dışı kalması halinde ya da tarayıcıların scriptleri desteklemediği durumda sayfanın erişilebilir olduğundan emin olmalıdırlar.

- İstemci tarafındaki çabuk geçişli içerikler yaratmaktan kaçının. Eğer bir kullanıcı tarayıcısı scriptleri işletmiyorsa, içerik görüntülenemeyecektir. Bununla beraber bu stil sayfalarının kombinasyonu ve scriptlerin kullanılmasıyla var olan içeriğin görüntülenmesi ya da gizlenmesinden farklıdır, eğer script yok ise içerik her zaman görüntülenecektir.
- URI olarak “javascript” kullanarak linkler yaratmaktan kaçının. Eğer bir kullanıcı scriptler kullanmıyorsa tarayıcı içerik linki yaratamayacağı için onlarda link yaratamayacaklardır.

12.2 Titreşimlere neden olan scriptler

Bu bölümdeki kontrol noktaları:

- Kullanıcı araçları kullanıcılara titreşimleri kontrol etme izni verinceye kadar ekranda titreşimlere neden olmaktan kaçının. **[Öncelik 1]**

12.3 Hareketliliğe ve yanıp sönmelere neden olan scriptler

Bu bölümdeki kontrol noktaları:

- [7.2](#) Kullanıcı araçları kullanıcılara yanıp sönmeleri kontrol etmeleri için izin verinceye kadar yanıp sönen içeriklere neden olmaktan kaçının. **[Öncelik 2]**
- [7.3](#) Kullanıcı araçları hareketli içerikleri dondurabilme izni verinceye kadar sayfalardaki hareketlilikten kaçının. **[Öncelik 2]**

12.4 Doğrudan erişilebilir scriptler

Bu bölümdeki kontrol noktaları:

- [6.4](#) Scriptler ve appletler için olay işleyicilerin giriş aygıtından bağımsız olduğundan emin olun. **[Öncelik 2]**
- [6.3](#) Scriptler, appletler ya da programa ait diğer nesnelerin kapalı olması ya da desteklenmediği durumlarda sayfanın kullanışlı olduğundan emin olun. Eğer bu mümkün değilse alternatif erişilebilir sayfada eşdeğer bilgi sağlayın. **[Öncelik 1]**
- [8.1](#) Scriptler ve appletler gibi doğrudan erişilebilir ya da destek teknolojileri ile uyumlu programa ait nesnelere oluşturun.
- [9.3](#) Scriptler için aygıtla bağlı olay işleyicilerin yerine mantıklı olay işleyicileri tayin edin. **[Öncelik 2]**

12.5 Scriptlerin alternatif sunumları

Bu bölümdeki kontrol noktaları:

- [1.1](#) Metin olmayan her bir öge için eşdeğer bir metin sağlayın. Bu; görüntüleri, semboller içeren bir metnin grafiksel sunumlarını, görüntü haritası bölgelerini, animasyonları, appletler ve diğer programa ait nesnelere, çerçeveleri, scriptleri, liste madde imleri olarak kullanılan görüntüleri, grafiksel butonları, videoları, bağımsız ses dosyalarını, görüntünün ses izleri vb.ni içerir. **[Öncelik 1]**

- [6.2](#) Dinamik içerikteki değişiklikler güncellendiğinde dinamik içerikler için eşdeğerlerin de güncellendiğinden emin olun. **[Öncelik 1]**

Bunu başarmanın bir yolu NOSCRIPT ögesinin kullanılmasıdır.

12.6 Sayfa güncellemesi ve yeni pencereler

Bu bölümdeki kontrol noktaları:

- [7.4](#) Kullanıcı araçları sayfa yenilenmesini durdurma yeteneğini sağlayıncaya kadar periyodik olarak otomatik yenilenen sayfalar yaratmaktan kaçının. **[Öncelik 2]**
- [7.5](#) Kullanıcı araçları otomatik yönlendirmeyi durduracak yeteneği sağlayıncaya kadar sayfaların otomatik olarak yönlendirilmesi için biçimlendirmeler kullanmayın. **[Öncelik 2]**
- [10.1](#) Kullanıcı araçları kullanıcılara açılmış olan pencereleri kapatma izni verinceye kadar pop-up ve diğer pencerelerin görünmesine neden olmayın ve aktif pencereyi kullanıcıyı bilgilendirmeden değiştirmeyin. **[Öncelik 2]**

13.HTML öğelerinin ve özniteliklerinin listesi

Bu bölümdeki kontrol noktaları:

- [11.2](#) W3C teknolojilerinin şiddetle itiraz ettiği özelliklerden kaçının. **[Öncelik 2]**

Ögeler

Öge adı	Tanımlı olduğu yer	Rolü	Teknik
A	2.0, 3.2	Yapı	N/A
ABBR		Yapı	N/A
ACRONYM		Yapı	N/A
ADDRESS	2.0, 3.2	Metadata	N/A
APPLET*	3.2	Yerleştirme	N/A
AREA	3.2	Yapı	N/A
B	2.0, 3.2	Sunum	N/A
BASE	2.0, 3.2	İşleme	N/A
BASEFONT*	3.2	Sunum	N/A
BDO		İşleme	N/A
BIG	3.2	Sunum	N/A
BLOCKQUOTE	2.0, 3.2	Yapı	N/A
BODY	2.0, 3.2	Yapı	N/A
BR	2.0, 3.2	Sunum	N/A
BUTTON		Yapı	N/A
CAPTION	3.2	Yapı	N/A
CENTER*	3.2	Sunum	N/A
CITE	2.0, 3.2	Yapı	N/A
CODE	2.0, 3.2	Yapı	N/A
COL		Yapı	N/A

COLGROUP		Yapı	N/A
DD	2.0, 3.2	Yapı	N/A
DEL		Metadata	N/A
DFN	3.2	Yapı	N/A
DIR*	2.0, 3.2	Yapı	N/A
DIV	3.2	Yapı	N/A
DL	2.0, 3.2	Yapı	N/A
DT	2.0, 3.2	Yapı	N/A
EM	2.0, 3.2	Yapı	N/A
FIELDSET		Yapı	N/A
FONT*	3.2	Sunum	N/A
FORM	2.0, 3.2	Yapı	N/A
FRAME		Yerleştirme	N/A
FRAMESET		Sunum	N/A
H1	2.0, 3.2	Yapı	N/A
HEAD	2.0, 3.2	Yapı	N/A
HR	2.0, 3.2	Sunum	N/A
HTML	2.0, 3.2	Yapı	N/A
I	2.0, 3.2	Sunum	N/A
IFRAME		Yerleştirme	N/A
IMG	2.0, 3.2	Yerleştirme	N/A
INPUT	2.0, 3.2	Yapı	N/A
INS		Metadata	N/A
ISINDEX*	2.0, 3.2	Yapı	N/A
KBD	2.0, 3.2	Yapı	N/A
LABEL		Yapı	N/A
LEGEND		Yapı	N/A
LI	2.0, 3.2	Yapı	N/A
LINK	2.0, 3.2	Metadata	N/A
MAP	3.2	Yapı	N/A
MENU*	2.0, 3.2	Yapı	N/A
META	2.0, 3.2	Metadata	N/A
NOFRAMES		Alternatif	N/A
NOSCRIPT		Alternatif	N/A
OBJECT		Yerleştirme	N/A
OL	2.0, 3.2	Yapı	N/A
OPTGROUP		Yapı	N/A
OPTION	2.0, 3.2	Yapı	N/A
P	2.0, 3.2	Yapı	N/A
PARAM	3.2	İşleme	N/A
PRE	2.0, 3.2	Sunum	N/A
Q		Yapı	N/A
S*		Sunum	N/A

SAMP	2.0, 3.2	Yapı	N/A
SCRIPT	3.2 (DTD)	İşleme	N/A
SELECT	2.0, 3.2	Yapı	N/A
SMALL	3.2	Sunum	N/A
SPAN		Yapı	N/A
STRIKE*	3.2	Sunum	N/A
STRONG	2.0, 3.2	Yapı	N/A
STYLE	3.2 (DTD)	İşleme	N/A
SUB	3.2	Sunum	N/A
SUP	3.2	Sunum	N/A
TABLE	3.2	Yapı	N/A
TBODY		Yapı	N/A
TD	3.2	Yapı	N/A
TEXTAREA	2.0, 3.2	Yapı	N/A
TFOOT		Yapı	N/A
TH	3.2	Yapı	N/A
THEAD		Yapı	N/A
TITLE	2.0, 3.2	Metadata	N/A
TR	3.2	Yapı	N/A
TI	2.0, 3.2	Sunum	N/A
U*	3.2	Sunum	N/A
UL	2.0, 3.2	Yapı	N/A
VAR	2.0, 3.2	Yapı	N/A

Öznitelikler

Özniteliğin adı	Uygulandığı öge	Rolü	teknik
abbr	TD , TH	Alternatif	N/A
accesskey	A , AREA , BUTTON , INPUT , LABEL , LEGEND , TEXTAREA	Kullanıcı arayüzü	N/A
alt	APPLET , AREA , IMG , INPUT	Alternatif	N/A
axis	TD , TH	Yapı	N/A
class	Pek çok öge	Yapı	N/A
dir	Pek çok öge	İşleme	N/A
for	LABEL	Yapı	N/A
headers	TD , TH	Yapı	N/A
hreflang	A , LINK	Metadata	N/A
id	Pek çok öge	Yapı	N/A
label	OPTION	Alternatif	N/A
lang	Pek çok öge	Metadata	N/A
longdesc	IMG , FRAME , IFRAME	Alternatif	N/A

scope	TD , TH	Yapı	N/A
style	Pek çok öge	İşleme	N/A
summary	TABLE	Alternatif	N/A
tabindex	A , AREA , BUTTON , INPUT , OBJECT , SELECT , TEXTAREA	Kullanıcı arayüzü	N/A
title	Pek çok öge	Metadata	N/A
usemap	IMG , INPUT , OBJECT	İşleme	N/A

Diğer yapısal öznitelikler:

start*, value*, rowspan, colspan, span

Diğer sunum öznitelikleri:

align*, valign*, clear*, nowrap*, char, charoff, hspace*, vspace*, cellpadding, cellspacing, compact*, face*, size*, background*, bgcolor*, color*, text*, link*, alink*, vlink*, border, noshade*, rules, size (deprecated according to element), marginheight, marginwidth, frame, frameborder, rows, cols

Diğer işlem komut öznitelikleri:

ismap, coords, shape

Diğer kullanıcı arayüzleri

target, scrolling, noresize

Diğer metadeta öznitelikleri

type, cite, datetime

Olay işleyici öznitelikler:

onblur, onchange, onclick, ondblclick, onfocus, onkeydown, onkeypress, onkeyup, onload, onunload, onmousedown, onmousemove, onmouseout, onmouseover, onmouseup, onreset, onselect, onsubmit, onunload