

Uygulama Geliştirme ve Yaygınlaştırma Süreçlerindeki Performans Değerlendirmesinde AHP Yönteminin Uygulanması

Hayriye Göksu

Yıldız Teknik Üniversitesi Davutpaşa Kampüsü Teknoloji Geliştirme Bölgesi Ziraat Teknoloji
A.Ş. No: 163 34220 Esenler, İstanbul
htekbir@ziraatteknoloji.com

Özet. Günümüzde bilgi teknolojileri üreten şirketlerin önemli problemlerinden biri yazılım kalitesini objektif ve nicel olarak tanımlamak ve ölçmektir. Yazılım kalitesinin değerlendirilmesinde birçok kriter olduğu gibi çeşitli modeller de yer almaktadır. Bu kriter ve modellerle yapılan değerlendirmeler sonucunda, yazılımın kalite düzeyinin belirlenmesi, yazılımlar arasında kalite karşılaştırmaları yapılması, hataların analiz edilmesi ve çalışan performansının ölçülmesi konularında çıkarımlar elde edilmektedir. Bu çalışmada Ziraat Teknoloji A.Ş. firmasında bulunan farklı uygulama geliştirme bölümlerinin ürettiği yazılımların karşılaştırılması ve bölümlerin elde edilen sonuçlara göre sıralanıp, performans değerlendirmelerinin yapılması için AHS(Analitik Hiyerarşi Süreci)[1] yönteminin ve tasarlanan yapıya özgü hesaplamaların kullanılması amaçlanmıştır. AHS yöntemi kullanılarak, uygulama yaygınlaştırmaları öncesinde yapılan kod ve ekran gözden geçirme çalışmalarında tespit edilen bulgu tiplerinin oranlarına ve veri yapısı hata oranlarına göre bölümlerin aldıkları puanlar, kriterleri bulgu çözüm durumu, alt kriterleri bulgu karakteri olan bir hiyerarşi modeli ile değerlendirilmiş, tasarlanan yapıya özgü hesaplamaların bu sonuçlara uygulanması ile yazılım kalitesi ölçülüp uygulama geliştirme bölümlerinin genel performansları karşılaştırılmıştır.

Anahtar Kelimeler. Yazılım Kalite Yönetimi, Yazılım Kalite Sınama, Yazılım Ölçütleri

1 Giriş

Yazılım mühendisliğinde kalite yönetimi son yıllarda proje maliyetlerinin büyümesi ile birlikte en çok konuşulan konulardan biri haline gelmiştir. Yazılım dünyasında kalite soyut ve öznel bir ölçü olmakla birlikte, hissedilebilen ancak tanımlanamayan bir kavram olarak karşımıza çıkmaktadır. Bir disiplin içerisinde kalitenin tam olarak tanımlanabilmesi için gelişmiş ölçme araçlarına ve tutarlı karşılaştırmalar için tanımlı

referans noktalarına ihtiyaç duyulmaktadır. Ölçme yapabilmek için de varlıkların özelliklerinin sayısallaştırılması gerekmektedir[2].

Bu bakımdan yazılım kalite yönetimi ve ölçme alanında yapılan bu çalışmada Ziraat Teknoloji A.Ş.'de uygulama geliştirme bölümlerince gerçekleştirilen yazılımlarda karşımıza çıkan kod gözden geçirme, kullanıcı ara yüzü denetimi, veri yapısı ve mimarisi ile ilgili yapılan çalışmalar üzerinden bölümlerin başarı oranlarının ölçülebilmesi hedeflenmiştir. Kod gözden geçirme ve kullanıcı ara yüzü denetimi ile ilgili performans ölçümü, kontrol listelerine göre oluşturulan bulgu(standart dışı kullanımlar) tipleri üzerinden gerçekleştirilmiştir. Bulgu tipleri ekiplerin geçmiş deneyimlerinden yararlanılarak belirlenmiştir. Bölümlerin farklı yaygınlaştırma türlerinde değişen kategorilerde performans ölçümlerinin yapılabilmesi amacıyla AHS yöntemi ve tasarlanan yapıya özgü bazı ek hesaplamalar yapılmıştır.

Bu çalışma üç bölümden oluşmaktadır. Veri Yapısı bölümünde, çalışmada kullanılan verinin yapısı ifade edilmektedir. Metodoloji bölümünde, AHS yönteminin ve tasarlanan sisteme özgü hesaplamaların, veri üzerine uygulanması bulunmaktadır. Üçüncü bölümde ise elde edilen sonuçlar yer almaktadır.

2 Veri Yapısı

Yazılım kalitesinin ölçülmesi amacıyla üç temel kontrol noktası göz önünde bulundurulmaktadır. Bu kontrol noktaları kod gözden geçirme, kullanıcı ara yüzü denetimi, veri yapısı ve mimarisinin kontrol edilmesidir. Kontrol noktalarının belirlenmesinde yaygınlaştırılan nesne türleri baz alınmıştır. Kod gözden geçirmede yazılım kalitesi, etkileşim, hata yönetimi, isimlendirme ve yazım standartlarına uygunluk değerlendirilmektedir. Kullanıcı ara yüzü denetiminde, ekran düzeni ve yerleşimi, kontrol bileşenlerinin kullanımı, kullanılabilirlik ve renklendirme göz önünde bulundurulmaktadır. Bu amaçla kod kontrol listesi ve ekran kontrol listesine göre bulgu(standartlara uyumsuzluk) ve uygunsuzluklar tespit edilmektedir. Veri yapısı ve mimarisinin kontrol edilmesinde tablo ve elemanlarının isimlendirme standartlarına uygunluğu, tablo tasarım hatalarının yapılmamış olması[2], tablolara eklenen indekslerin hatalı olmaması, girilen açıklamaların hatasız ve eksiksiz olması ve ER diyagram kontrolü yapılmaktadır. Ziraat Teknoloji A.Ş. Mimari ve Kontrol Bölümü tarafından yapılan bu çalışmada, her bir yaygınlaştırılan yazılım parçası(App Kernel, Common, UI Kernel, Contract, Entity, Interop, WCF Service, Operation, Stored Function, Facade, Client Application, Stored Procedure, Batch Application, Screen, User Library) nesne olarak değerlendirmekte ve bu nesnelerdeki bulgu sayıları performans göstergesi olarak alınmaktadır. Bu sorgulamaların tümü bulgu olarak adlandırdığımız bölümü ifade etmektedir.

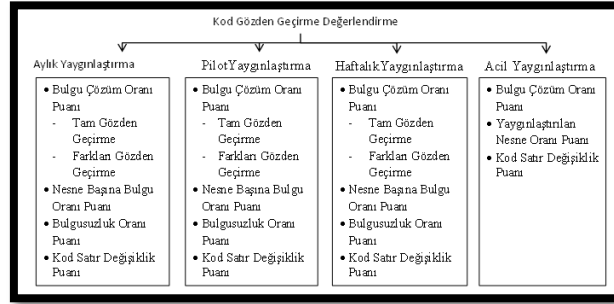
Ziraat Teknoloji A.Ş.' de yaygınlaştırmalar Aylık Yaygınlaştırma, Haftalık Ara Yaygınlaştırma, Acil Yaygınlaştırma ve Pilot Yaygınlaştırma isimleri ile sınıflandırılmaktadır.

Kod gözden geçirme, kullanıcı ara yüzü denetimi, veri yapısı ve mimarisinin incelenmesi sonucunda elde edilen nesne başına düşen bulgu sayıları bölümlerin yazılım performanslarını ölçmede kullanılır.

3 Metodoloji

Bu çalışmada bölümlerin performans değerlendirmelerinde esas alınan kategoriler, Ağırlıklı Bulgu Çözüm Oranı Puanı(ABÇOP), Nesne Başına Bulgu Oranı Puanı(NBBO), Bulgusuzluk Oranı(BZO), Kod Satır Değişiklik Puanı(KSDP) olarak belirlenmiştir. Bu kategoriler yaygınlaştırma türü bazında değişiklik gösterebilmektedir. Şekil 1’ de yaygınlaştırma türlerine göre belirlenen kategoriler yer almaktadır.

Kod gözden geçirme, kullanıcı ara yüzü denetimi, veri yapısı ve mimarisinin incelenmesi sonucunda elde edilen veri, bölümlerin yazılım performanslarını ölçmede kullanılmıştır.



Şekil 1. Kod Gözden Geçirme Değerlendirme

3.1 Ağırlıklı Bulgu Çözüm Oranı Puanı(ABÇOP) Hesaplaması

Değerlendirmede esas alınan kategorilerden biri olan ABCOP hesaplanmasında AHS yöntemi kullanılmıştır.

Tablo 1. Kategoriler

Kalite Kısaltması	Kalite Kategorileri	Kalite Kategorisi Açıklaması
TGG	Tam Gözden Geçirme	Nesnelerin ilk defa gözden geçirilmesi
FGG	Farklı Gözden Geçirme	Daha önce gözden geçirilmiş nesnenin farkına bakılması
-	Ana Kriterler	Gözden geçirilip gönderilen nesneler için girilen bulguların çözülme durumlarıdır
-	Alt Kriterler	Gözden geçirilip gönderilen nesneler için girişi yapılan bulguların seviyeleridir

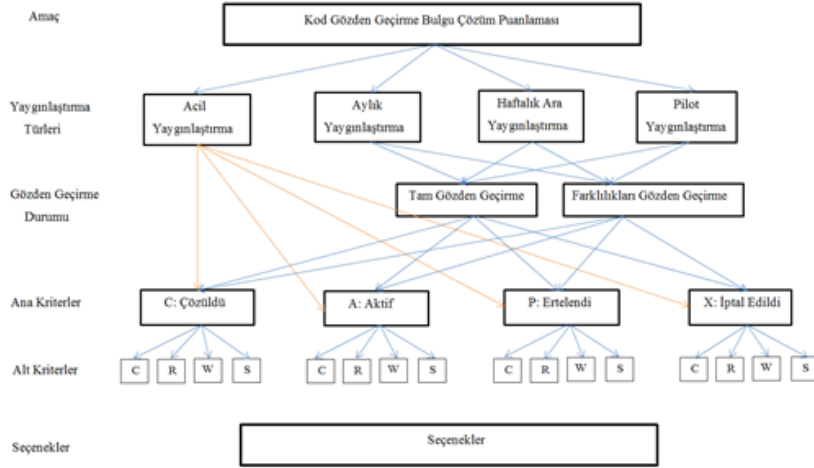
AHS Yöntemi.

AHS yöntemi, gruplara ve bireylere, karar verme sürecindeki nitel ve nicel faktörleri birleştirme olanağı veren, güçlü ve kolay anlaşılır bir yöntem bilimidir. Bu yöntem, karışık, anlaşılması güç veya yapılaşmış sorunlar için genel bir çok kriterli karar verme olanağı sağlar. Yöntemde her sorun için amaç, kriter, olası alt kriter seviyeleri ve seçeneklerden oluşan bir hiyerarşi modeli kullanılır[3].

Tablo 2. Kriterler

Kriter Kısaltması	Kriter Kodu	Kriter Açıklaması
C	Düzeltildi	Bulgu çözülmüştür
A	Aktif Bırakıldı	Bulgu aktif durumdadır
P	Ertelendi	Bulgu ileride yapılmak üzere ötelenmiştir
X	İptal Edildi	Teknik olarak çözülemeyecek bir bulgudur ya da bulgu değildir

Bu çalışmada amaç, kod gözden geçirme, kullanıcı ara yüzü denetimi sonucunda ortaya çıkan bulguların çözümlenmesine dair çözüm puanlaması olarak belirlenmiştir. Kriterler belirlenirken bulguların bölümler tarafından yanıtlanma durumu baz alınmıştır. Mimari ve Kontrol Bölümü'nce gözden geçirilip gönderilen nesnelere için girilen bulgulara yönelik davranış durumları için 4 farklı kod(Tablo 2) belirlenmiştir. Alt Kriterlerin belirlenmesinde ise bulguların yazılımdaki önem seviyesine göre gruplandırılması esas alınmıştır. Alt kriterler, bulgular için belirtilen türler olup 4 farklı durum koduna(Tablo 3) sahiptir. Seçenekler verisi olarak 26 farklı Uygulama Geliştirme bölümü değerlendirilmiştir.(Şekil 2)



Şekil. 2. Bulgu çözüm oranı puanı hesaplaması - AHS hiyerarşisi

Tablo 3. Alt kriterler

Alt Kriter Kısaltması	Alt Kriter Kodu	Alt Kriter Açıklaması
C	Kritik	Düzeltilip sonucunun üretim ortamına kadar taşınması gerekmektedir
R	Önemli	Standartlara uymayan bir kullanım vardır, düzeltilmelidir
W	Uyarı	Uyarı niteliğindedir, düzeltilmesinde fayda vardır
S	Öneri	Programlama pratikleri açısından önerilen yöntemleri belirtir

TGG ve FGG Ağırlıklı Bulgu Çözüm Oranı Puanı(ABCOP) Hesaplaması.

Genel olarak puan hesaplamalarında, yaygınlaştırma türlerine bağlı olarak iki farklı değerlendirme yapılmıştır. Aylık, Haftalık Ara ve Pilot Yaygınlaştırma türlerinde TGG ve FGG durumlarına göre C, A, P ve X ana kriterleri değerlendirmeye tabi tutulurken, acil yaygınlaştırma türünde TGG ve FGG durumları için bir ayrıma gidilmemiştir. Bulguların önemleri ve kontrol noktalarındaki deneyimler göz önüne alınarak 1 ile 9 arasında öncelik puanları verilerek, her bir puan genel puan toplamına bölünüp ağırlık puanı hesaplanmıştır(Tablo 4 ve Tablo 5).

Tablo 4. Aylık, Haftalık ve Pilot Yaygınlaştırma türlerinde TGG ve FGG durumlarına göre C, A, P ve X ana kriterleri için verilen öncelik puanları ve bu puanlara göre hesaplanan ağırlık puanları

Ana Kriterler	TGG		FGG	
	Öncelik Puanları	Ağırlık Puanları	Öncelik Puanları	Ağırlık Puanları
C	8	0,421	9	0,474
A	2	0,105	1	0,053
P	3	0,158	4	0,211
X	6	0,316	5	0,263

Tablo 5. Acil Yaygınlaştırma türü için kabul edilen C, A, P ve X ana kriterleri için verilen öncelik puanları ve öncelik puanlarına göre hesaplanan ağırlık puanları

Ana Kriterler	Öncelik Puanları	Ağırlık Puanları
C	2	0,105
A	8	0,421
P	6	0,316
X	3	0,158

Tüm yaygınlaştırma türlerinde ve bu yaygınlaştırma türlerinin TGG ve FGG durumlarında alt kriterlerin öncelik puanları eşit alınmaktadır(Tablo 6).

Tablo 6. Alt kriterlerin ağırlık puanları

Alt Kriterler	Ana Kriterler							
	C		A		P		X	
	Öncelik Puanı	Ağırlık Puanı	Öncelik Puanı	Ağırlık Puanı	Öncelik Puanı	Ağırlık Puanı	Öncelik Puanı	Ağırlık Puanı
C	8	0,400	1	0,063	1	0,059	1	0,250
R	6	0,300	3	0,188	2	0,118	1	0,250
W	5	0,250	5	0,313	6	0,353	1	0,250
S	1	0,050	7	0,438	8	0,471	1	0,250

AHS yöntemi ile yapılan hesaplamalar sonucunda her bir bölümün her bir yaygınlaştırma türünde TGG ve FGG ABÇOP'larını hesaplamak için Eşitlik 1, Eşitlik 2, Eşitlik 3 ve Eşitlik 4' den yararlanılmaktadır.

$$TGG \text{ Oranı}(TGGO) = \frac{TGG \text{ Nesne Sayısı}}{Gözden Geçirilen Toplam Nesne Sayısı} \quad (1)$$

$$TGG \text{ ABÇOP} = TGG \text{ BÇOP} * TGGO \quad (2)$$

$$FGG \text{ Oranı}(FGGO) = \frac{FGG \text{ Nesne Sayısı}}{Gözden Geçirilen Toplam Nesne Sayısı} \quad (3)$$

$$FGG \text{ ABÇOP} = FGG \text{ BÇOP} * FGGO \quad (4)$$

3.2 Nesne Başına Bulgu Oranı(NBBO)

NBBO hesaplamasında, belirlenen yaygınlaştırma döneminde ilgili bölümlerce yaygınlaştırılan nesnelere gözden geçirilir ve tespit edilen bulgu sayısına ile gözden geçirilen nesne sayısına göre Eşitlik 5' den yararlanır. Bu oran değerlendirmeye katılırken en düşük NBBO'ya sahip servis en başarılı olarak düşünülmüştür.

$$NBBO = \frac{Toplam \text{ Bulgu Sayısı}}{Gözden Geçirilen Toplam Nesne Sayısı} \quad (5)$$

3.3 Bulgusuzluk Oranı(BZO)

BZO hesaplamasında, belirlenen yaygınlaştırma döneminde bölümlerce yaygınlaştırılan bulgusuz nesnelere göz önünde bulundurulur ve Eşitlik 6 kullanılır. Bu oran değerlendirmeye katılırken en yüksek BZO'ya sahip servis en başarılı olarak düşünülmüştür.

$$BZO = \frac{Bulgusuz \text{ Nesne Sayısı}}{Gözden Geçirilen Toplam Nesne Sayısı} \quad (6)$$

3.4 Kod Satır Değişiklik Puanı(KSDP) Hesaplaması

KSDP, yaygınlaştırma dönemi içinde bölümlerce geliştirilen her kod satır sayısının, nesne türüne göre belirlenen katsayıyla çarpılması ile elde edilir. Katsayılar, nesnelere içerisinde yer alan metotların toplam çağrılma sayıları ve nesnelere katman türüne göre belirlenmiş kod satır sayıları kullanılarak hesaplanır.

3.5 Diğer Hesaplamalar

Ekran kontrolleri için yaygınlaştırma türü bazında hesaplanan TGG ABÇOP, FGG ABÇOP, NBBO ve BZO toplanarak sonuç puanı(Eşitlik 7) elde edilir.

$$\text{Sonuç Puanı} = \text{TGG ABÇOP} + \text{FGG ABÇOP} + \text{NBBO} + \text{BZO} \quad (7)$$

Kod gözden geçirme için yaygınlaştırma türü bazında hesaplanan TGG ABÇOP, FGG ABÇOP, NBBO ve BZO toplanarak sonuç puanı(Eşitlik 8) elde edilir.

$$\text{Sonuç Puanı} = \text{TGG ABÇOP} + \text{FGG ABÇOP} + \text{NBBO} + \text{BZO} + \text{KSDP} \quad (8)$$

Her bir servisin 4 farklı yaygınlaştırma türünden aldığı sonuç puanları, yaygınlaştırma türü bazında 3 ile -1 arasındaki katsayılarla(Tablo 7) Eşitlik 9' daki gibi işleme tabi tutulup, ekran gözden geçirme ve kod gözden geçirme Genel Sonuç Puanı hesaplanır(Eşitlik 9). Katsayıların belirlenmesi noktasında, yapılması beklenen yaygınlaştırma türüne teşvik ön planda tutulup en yüksek katsayı ona verilmiş olup, istenmeyen yaygınlaştırma türüne doğru katsayı azaltılmıştır.

$$\text{Genel Sonuç Puanı} = (\text{AYY Sonuç Puanı} * 3) + (\text{PIY Sonuç Puanı} * 2) + (\text{HAY Sonuç Puanı} * 1) + (\text{ACY Sonuç Puanı} * -1) \quad (9)$$

Tablo 7. Yaygınlaştırma Türü Katsayıları

Yaygınlaştırma Türü Kısaltması	Yaygınlaştırma Türü	Katsayı
AYY	Aylık Yaygınlaştırma	3
PIY	Pilot Yaygınlaştırma	2
HAY	Haftalık Yaygınlaştırma	1
ACY	Acil Yaygınlaştırma	-1

Veri yapısı ve mimari ayrı olarak değerlendirilmekte olup, yapılan kontrollere göre bölümlerin, reddedilen tablo sayısının onaya gelen tablo sayısına bölünmesi ile her bir bölümün veri yapısı ve mimari tasarım uygunluk oranı bulunur. Elde edilen oran ile yaygınlaştırma sürecinde tüm bölümlerden onaya gelen toplam tablo sayısına göre normalizasyon yapılarak, ağırlıklı veri yapısı ve mimarisi puanı hesaplanır.

Uygulama geliştirme bölümleri için hesaplanan puanlar kod gözden geçirme(Tablo 8), ekran gözden geçirme(Tablo 9), veri yapısı ve mimarisi niteliklerine(Tablo 10) göre sıralanarak, bölümlerin birbirlerine göre yazılım kalitesi ölçümü gerçekleştirilir.

Tablo 8. Örnek Kod Gözden Geçirme puanlamalarına göre bölümlerin sıralanması

Bölüm	AYY Puanı	PİY Puanı	HAY Puanı	ACY Puanı	Toplam Puan	Ağırlıklı Toplam Puan
B5	6,214	7,799	5,280	0,793	44,009	8,868
B3	4,203	8,802	5,206	0,325	40,301	8,120
B26	9,663	8,557	3,650	14,750	38,653	7,788
B1	6,646	0	5,831	2,743	37,857	7,628
...
B22	4,622	0	5,102	23,227	0,844	0,170

Tablo 9. Örnek Ekran Gözden Geçirme puanlamalarına göre bölümlerin sıralanması

Bölüm	AYY Puanı	PİY Puanı	HAY Puanı	ACY Puanı	Toplam Puan	Ağırlıklı Toplam Puan
B8	8,226	9,427	6,182	1,640	54,256	11,626
B1	5,527	0	14,649	0	45,880	9,831
B26	9,357	0	5,883	0	39,838	8,537
B11	7,384	5	2,749	1,588	36,096	7,735
...
B4	2,387	4,247	2,803	9,571	11,688	2,505

Tablo 10. Örnek Veri Yapısı ve Mimarisi puanlamalarına göre bölümlerin sıralanması

Bölüm	Gelen Taşıma İsteği Sayısı	Onaylanan Taşıma İsteği Sayısı	Veri Yapısı ve Mimarisi Puanı	Ağırlıklı Veri Yapısı ve Mimarisi Puanı
B3	129	120	0,930	0,047
B20	138	118	0,855	0,043
B16	133	105	0,789	0,040
B9	188	173	0,920	0,046
...
B19	45	22	0,490	0,025

4 Sonuç

Yazılım dünyasında soyut ve öznel bir olgu olan kalite kavramının, ölçülebilir bir disiplin içerisinde tanımlanabilmesi için gelişmiş ölçme araçlarına ve tutarlı karşılaştırmalar yapılabilmesi için tanımlı referans noktalarına ihtiyaç duyulmaktadır. Ayrıca yazılım kalite değerlendirmesinin yapılabilmesi için karşılaştırma yapma gereksinimi doğmaktadır.

Bu nedenle, uygulama geliştirme bölümlerince gerçekleştirilen yazılımlarda karşımıza çıkan kod gözden geçirme, kullanıcı ara yüzü denetimi, veri yapısı ve mimarisi ile ilgili yaygınlaştırılan nesnelereki bulgu sayıları üzerinden ekiplerin başarı oranlarının ölçülerek değerlendirilmesi, nesnel bir yaklaşım olmakla birlikte, bölümler arasında kıyaslama yapmamızı da sağlamıştır. Hesaplama esnasında AHS yöntemi ve tasarlanan sisteme özgü yapılan hesaplamalar bütünsel şekilde kullanılmıştır.

AHS yönteminde kullanılan katsayılar, deneyimli çalışanların istekleri doğrultusunda belirlenmiştir. Katsayı tespitinde uygun yöntemlerin kullanımına ilişkin araştırmaların yapılması gerekmektedir.

Gelecek çalışma olarak aynı nesnelere, aynı yaygınlaştırma döneminde tekrar gözden geçirilme durumu oluştuğunda ABÇOP hesaplaması için yeni bir yöntem geliştirilmesi hedeflenmektedir. Ayrıca bulgu çeşitliliğinin artırılması ve bulgu tespitinde mümkün olduğu kadar manuel gözden geçirmeden kaçınılarak otomatik tespitlerin yapılması amaçlanmaktadır.

Kaynaklar

1. Saaty, T. L., How to make a decision: The Analytic Hierarchy Process. Joseph M. Katz Graduate School of Business, University of Pittsburgh, Pittsburgh, PA 15260, USA
2. Jørgensen, M.: "Software Quality Measurement", Advances in Engineering Software, Vol. 30, pp. 907-912 (1999)
3. Saaty, T. L. 1980. The Analytic Hierarchy Process, McGraw-Hill Book Co., N.Y
4. Lee, H.: Information Processing & Management, January-February 1995. Vol. 31, pp. 59-67, Seoul, Korea, ISSN: 0306-4573 (2006)
5. Belton, V. and Gear, T. 1982. On a Shortcoming of Saaty's Method of Analytic Hierarchies. Omega Vol. 11 No. 3, 226-230.
6. Forman, E. H., Saaty, T. L., Selly, M. A., Waldron, R., 1983. Expert Choice, Decision Support Software, McLean, VA.
7. Saaty, T. L. 1986. Axiomatic Foundation of the Analytic Hierarchy Process, Management Science, 32, 841- 855.
8. Saaty, T. L. 1990. An Exposition of the AHP in Reply to the Paper "Remarks on the Analytic Hierarchy Process", Management Science, Vol. 36, No. 3, 259-268.
9. Saaty, T. L. 1991a. How to Make a Decision: The Analytic Hierarchy Process, European Journal of Operations Research, Vol. 48, 9-26.
10. Saaty, T. L. 1991b. Rank and the Controversy About the Axioms of Utility Theory -- A Comparison of AHP and MAUT, Proceedings of the 2nd International Symposium of The Analytic Hierarchy Process, Pittsburgh, PA, 87-111.

11. Saaty, T. L. 1994a. How to Make a Decision: The Analytic Hierarchy Process, *Interfaces*, 24, 19-43.
12. Saaty, T. L. 1994b. *Fundamentals of Decision Making*, RWS Publications, Pittsburgh, PA
13. Saaty, Thomas L., 1996. *Decision Making with Dependence and Feedback*, RWS Publications, Pittsburgh, PA, 1996.
14. Saaty, T. L. 1980. *The Analytic Hierarchy Process*, McGraw-Hill Book Co., N.Y.