

**YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

UZAKTAN EĞİTİM YÖNTEM ve UYGULAMALARI

Matematik Mühendisi Mert BAL

**F.B.E Matematik Mühendisliği Anabilim Dalında
Hazırlanan**

YÜKSEK LİSANS TEZİ

106189

Tez Danışmanı

: Yrd. Doç. Dr. İbrahim EMİROĞLU

İ.E.

Prof.Dr. Mustafa Sivri

[Signature]

Prof.Dr. Mehmet AHLATGİOĞLU

İSTANBUL, 2001

**T.C. YÜKSEKÖĞRETİM KURULU
DOKÜMANTASYON MERKEZİ**

İÇİNDEKİLER

	Sayfa
KISALTMA LİSTESİ.....	vi
ÇİZELGE LİSTESİ.....	vii
ÖNSÖZ.....	viii
ÖZET.....	ix
ABSTRACT	x
1. GİRİŞ.....	1
2. UZAKTAN EĞİTİM YÖNTEM ve UYGULAMALARI	2
2.1 Uzaktan Eğitim ve Uzaktan Öğrenim.....	2
2.2 Uzaktan Eğitim Teknolojileri	3
2.2.1 Bilgisayar Tabanlı Eğitim.....	4
2.2.1.1 Teknik Gereksinimler ve Tasarım	6
2.2.1.2 Multimedya İmkanları	6
2.2.1.3 Genel ve Tutarlı Arayüz	7
2.2.1.4 Belgelerin Düzenlenmesi.....	7
2.2.1.5 BTE'lerin İçerikleri	7
2.2.1.6 Etkileşimlik.....	8
2.2.1.7 BTE'nin Avantajları	9
2.2.1.8 BTE'nin Dezavantajları	9
2.2.2 Bilgisayar Destekli Eğitim.....	10
2.2.2.1 Gereksinim ve Tasarım.....	11
2.2.2.2 Multimedya Olanakları.....	11
2.2.2.3 Arayüz.....	12
2.2.2.4 Etkileşim.....	12
2.2.2.5 Konu Başlığına Mahsus Olma	12
2.2.2.6 BDE'nin Avantajları.....	12
2.2.2.7 BDE'nin Dezavantajları.....	14
2.2.3 Web Tabanlı Eğitim.....	15
2.2.3.1 Genel Olarak WTE'de Gereksinimler ve Tasarım	16
2.2.3.2 İletişim İmkanları Olarak WTE'de Gereksinimler ve Tasarım	18
2.2.3.3 Gereksinimler	20
2.2.3.4 WTE'nin Avantajları	21
2.2.3.5 WTE'nin Dezavantajları.....	23
2.2.4 Telekonferans Sistemi	27
2.2.4.1 Gereksinimler ve Tasarım.....	28
2.2.4.2 Telekonferans Sisteminin Avantajları	29
2.2.4.3 Telekonferans Sisteminin Dezavantajları	30
2.2.5 Videoteyp Sistemleri	32
2.2.5.1 Gereksinimler ve Tasarım.....	32
2.2.5.2 Videoteyp Sistemlerinin Avantajları	33

2.2.5.3	Videoteyp Sistemlerinin Dezavantajları	33
2.2.6	Video Tele Eğitim.....	34
2.2.6.1	Gereksinimler ve Tasarım.....	35
2.2.6.2	Video Tele Eğitimin Avantajları	36
2.2.6.3	Video Tele Eğitiminin Dezavantajları	37
3.	LINUX İŞLETİM SİSTEMİ.....	40
3.1	Linux Komut Yapısı	40
3.2	Dosya ve Dizin Yapısı	41
3.2.1	Dosyaların Listesi ve İçerikleri.....	43
3.2.2	Dosya İçeriklerinin less Komutu ile Görüntülenmesi	43
3.2.3	Dosyaların Kopyalanması.....	43
3.2.4	Dosyaların Silinmesi.....	44
3.3	Linux Kabukları.....	45
3.4	Dosya ve Dizin İşlemleri	46
3.4.1	Erişim Hakları.....	46
3.4.2	Dosya İzinlerinin Değiştirilmesi.....	46
3.4.3	Dosyanın Sahibi ve Grubu.....	48
3.5	Dizin Tarama	50
3.6	Metin Düzenleyiciler	50
3.6.1	Pico Editörü	50
3.6.2	Vi Editörü	51
3.7	Linux Sistem Yönetimi.....	53
3.7.1	Kullanıcı Hesabı Açılması	53
3.7.2	Kullanıcı Grubu	53
3.7.3	Yeni Kullanıcı Grubu Eklenmesi.....	54
3.7.4	Kullanıcı Grubunun Silinmesi	55
3.7.5	Kullanıcı Hesabının Silinmesi	55
4.	POSTGRESQL VERİTABANI YÖNETİM SİSTEMİ.....	56
4.1	Veritabanının Yönetimi	57
4.1.1	Veritabanının Yaratılması.....	57
4.1.2	Veritabanına Erişim	58
4.1.3	Veritabanının Silinmesi	59
4.2	Veri Türleri	59
4.3	Operatörler	60
4.3.1	Sayısal Operatörler	60
4.3.2	Sözcüksel Operatörler.....	61
4.3.3	Genel Operatörler	61
4.4	Sorgulama Dili.....	62
4.4.1	Sınıf Oluşturma.....	62
4.4.2	Bir Sınıfa Veri Yükleme.....	63
4.4.3	Bir Sınıfın Sorgulanması	63
4.4.4	Select Sorgularının Yönlendirilmesi.....	64
4.4.5	Bir Sınıfın Güncellenmesi	64
4.4.6	Bir Sınıftaki Verilerin Silinmesi.....	65
4.4.7	Toparlama Fonksiyonları.....	65
4.5	SQL Komutları	66
4.5.1	Abort.....	66
4.5.2	Alter Table	66
4.5.3	Alter User.....	66

4.5.4	Begin Work.....	67
4.5.5	Commit.....	67
4.5.6	Copy.....	67
4.5.7	Create Database	68
4.5.8	Create Index.....	68
4.5.9	Create Sequence.....	68
4.5.10	Create Table.....	68
4.5.11	Create User	69
4.5.12	Delete.....	69
4.5.13	Drop Database	70
4.5.14	Drop Index.....	70
4.5.15	Drop Table.....	70
4.5.16	Insert	70
4.5.17	Select.....	71
4.5.18	Update.....	71
4.6	Postgres ile Gelen Bazı Uygulamalar.....	71
4.6.1	Createdb.....	71
4.6.2	Createuser	72
4.6.3	Destroydb.....	73
4.6.4	Destroyuser	73
4.6.5	Psqll	74
5.	PERL PROGRAMLAMA DİLİ.....	77
5.1	Perl Programlama Dilinin Temelleri	77
5.1.1	Derleme ve Yorumlama İşlemleri	77
5.1.2	Standart Bilgi Giriş ve Çıkışları	78
5.1.3	Fonksiyonlar	78
5.1.4	Değişkenler	80
5.1.4.1	Skaler Değişkenler.....	80
5.1.4.2	Diziler	81
5.1.4.3	Özel Tablolar	82
5.1.5	If Deyimi ve Anlatımlar.....	83
5.1.6	Döngüler	87
5.1.7	Mantıksal Operatörler.....	90
5.2	Dosya İşlemleri.....	90
5.2.1	Dosyadan Okuma İşlemleri	92
5.2.2	Dosyaya Yazma İşlemleri.....	92
5.3	Regüler Terimler.....	93
5.3.1	Regüler Terimlerin Kullanılması	93
5.3.2	Regüler Terimlerde Kullanılan Özel İşaretler	94
5.3.3	Split() Fonksiyonu	96
5.4	Altprogramlar	98
6.	HTML PROGRAMLAMA	102
6.1	HTML Programlamaya Giriş.....	102
6.2	Bağlantılar.....	103
6.3	Metin Hizalama	104
6.4	Metin Biçimlendirme ve Karakter Türü Denetimi	105
6.5	HTML Formları.....	106
6.6	Resim ve Renk Özellikleri.....	108
6.7	Tablolar.....	109

7.	CGI PROGRAMLAMA.....	112
7.1	CGI Programlarının Çalışması	113
7.2	Cgi-lib.pl Kitaplığı.....	114
7.3	Perl Destekli CGI Uygulamaları.....	114
7.4	Bir Perl ve Postgresql Destekli CGI Uygulaması.....	116
8.	SONUÇLAR ve ÖNERİLER.....	118
KAYNAKLAR.....		119
ÖZGEÇMİŞ.....		121



KISALTMA LİSTESİ

BDE	Bilgisayar Destekli Eğitim
BDÖ	Bilgisayar Destekli Öğrenim
BTE	Bilgisayar Tabanlı Eğitim
BTÖ	Bilgisayar Tabanlı Öğretim
CD-ROM	Compact Disc-Read Only Memory
CGI	Common Gateway Interface
CODEC	Compressor Decompressor
FAQ	Frequently Asked Questions
FTP	File Transfer Protocol
HTML	Hyper Text Markup Language
IBM	International Business Machine
ISP	Internet Service Provider
ISS	Internet Service Server
PC	Personal Computer
PERL	Practical Extraction and Report Language
PPP	Point to Point Protocol
SQL	Structured Query Language
TCP/IP	Transmission Control Protocol/ Internet Protocol
UUCP	Unix to Unix Copy Program
VTE	Video Tele Eğitim
WTE	Web Tabanlı Eğitim

ÇİZELGE LİSTESİ

Çizelge 2.1 Yönlendirme teknolojilerinin avantaj ve dezavantajları	5
Çizelge 2.2 WTE dersleri için gereklilikler ve amaçlar	27
Çizelge 2.3 Telekonferans sistemi için gereklilikler ve amaçlar	32
Çizelge 2.4 Videoteyp için gereklilikler ve amaçlar	34
Çizelge 3.1 Linux dosya ve dizin yapısı.....	42
Çizelge 3.2 Komut listesi	51
Çizelge 4.1 Postgresql’de veri türleri	60
Çizelge 4.2 Postgresql’de kullanılan sayısal operatörler.....	61
Çizelge 4.3 Postgresql’de kullanılan sözcüksel operatörler	61
Çizelge 4.4 Postgresql’de kullanılan genel operatörler	62



ÖNSÖZ

Bu tezin amacı uzaktan eğitimde kullanılan ve yönlendirme teknolojileri adı verilen yöntemlerin açıklanması, birbirleri ile karşılaştırılması ve bu yöntemlerin birbirlerine göre avantaj ve dezavantajlarının ortaya konmasıdır.

Ayrıca bu uzaktan eğitim ile ilgili proje geliştirme aşamasında kullanılabilecek Perl, Cgi ve Html programlama dilleri, Postgresql veritabanı yönetim sistemi ve Linux işletim sistemi hakkında bilgiler verilmektedir.

Bu tez çalışmam sırasında bana yardımlarını esirgemeyen, beni her zaman destekleyen Sayın Hocam Yrd. Doç. Dr. İbrahim Emiroğlu'na teşekkürü bir borç bilirim. Ayrıca çalışmalarım sırasında bana destek olan arkadaşlarım Araştırmacı Uğur Kaşif Boyacı, Elektrik Y. Müh. Tuncer Beyazoğlu, Yazılım Müh. Semih Yıldırım ve Mimar Bahadır Özcan ve hayatım boyunca bana her türlü desteği veren aileme sonsuz teşekkürlerimi sunarım.



ÖZET

Uzaktan Eğitim’de kullanılan ve yönlendirme teknolojileri adı verilen bilgisayar tabanlı eğitim, bilgisayar destekli eğitim, web tabanlı eğitim, telekonferans sistemi, videoteyp sistemleri ve video tele eğitim sistemleri incelenmiştir.

Bu yönlendirme teknolojilerinin teknik gereksinimleri ve tasarım aşamaları ile eğitmen, öğrenci ve kurum açısından her bir teknolojinin avantaj ve dezavantajları açıklanmıştır.

Ayrıca bir uzaktan eğitim projesi geliştirme aşamasında kullanılabilecek Perl, Cgi ve Html programlama dilleri ile Postgresql veritabanı yönetim sistemi ve Linux işletim sistemi ile ilgili bilgiler ve bazı örnekler verilmiştir.

Anahtar Kelimeler: Uzaktan eğitim, bilgisayar tabanlı eğitim, bilgisayar destekli eğitim, web tabanlı eğitim, telekonferans sistemi, videoteyp sistemi, video tele eğitim.



ABSTRACT

The tools called navigation technologies used in distance learning systems which are computer based training, computer aided instruction, web based training, teleconferencing systems, videotape systems and video tele training systems are focused.

All these navigation technologies are reviewed with their design phase, technical requirements advantages and disadvantages from the point of instructors, learners and organizations.

Moreover information and examples are given about programming languages to be used by developing and implementing a distance learning project such as Perl, Cgi, Html and database management system Postgresql and Linux operating system.

Keywords: Distance learning, computer based training, computer aided instruction, teleconferencing system, videotape system, video tele training system.



1. GİRİŞ

Uzaktan eğitim, öğrencilerin dersleri İnternet üzerinden hem görüntü hem de ses destekli olarak izleyebildiği, öğrenci kayıt işlemlerinin yapılabildiği, dersleri takip edebildikleri, ders notlarına, kılavuzlara ve alıştırmalara erişebildiği, sınav olabildikleri ve bu uygulama dahilinde başarı durumlarının ölçülebildiği ve buna göre not değerlendirmesinin yapılabildiği bir ortamı sağlamak amacıyla oluşturulan bir teknolojidir.

Uzaktan eğitim için bir diğer tanım ise, aynı ortamda eğitmenin fiziksel olarak bulunmadığı bir eğitim şeklidir. Yapılan bu tanımlama çok genel olmakla beraber bu sistemin kurulduğu merkezlerdeki gereksinimler çerçevesinde bu tanım oldukça fazla değişkenlikler gösterebilmektedir. Söz edilen gereksinimler olarak, kapasiteleri hızla artan üniversitelerde yer, zaman ve eğitmen sıkıntılarının artması, silahlı kuvvetlerde simülasyon uygulamalarına dönük ses ve görüntü destekli program eksikliklerinin ortaya çıkması sıralanabilir.

Uzaktan eğitim düşüncesine dayanan yapılar özellikle eğitim sektöründe bulunan ticari işletmeler ile üniversiteler tarafından kurulmakta ve kullanıma açılmaktadır. Ticari işletmeler arasında Macromedia ve NetG firmaları bu konuda program paketi üretmektedirler. Ekonomik yönden ağır koşullar sistemin günümüzde yaygınlaşmaya başlaması ve sürekli gelişim göstermesi ve rekabet ortamının ortaya çıkması ile iyileşmeye başlamıştır. Ticari işletmeler tarafından hazırlanan paketler Microsoft firmasının Windows işletim sistemlerini taban alarak hazırlanmaktadır. Eğitimi verilecek konular hakkında görsel yönden çekiciliği olan yapılar kurmak ve eğitime katılacak kişilere bunu en iyi şekilde aktarmak temel amaçtır. Üniversiteler tarafından oluşturulan yapılar ise, İnternet üzerinden belirli konulara destek veren anlatımları ve sertifika programlarını içermektedir. Bu konuda özellikle Amerika Birleşik Devletleri'nde bulunan bir çok üniversitede benzer uygulamalar ile karşılaşılabilir. Bu türde programlar genelde kayıt işlemleri, kursların takip edilmesi, belirtilen zamanlarda projelerin yapılması gibi uygulamaları içermektedir. Bu uygulamaların yanı sıra sınav uygulamaları ise genelde çoktan seçmeli yöntemine dayanmaktadır. Uzaktan eğitimde kullanılan yöntemler bilgisayar tabanlı eğitim, bilgisayar destekli eğitim, web tabanlı eğitim, telekonferans sistemi, videoteyp sistemleri ve video tele eğitim şeklinde sıralanabilir.

2. UZAKTAN EĞİTİM YÖNTEM ve UYGULAMALARI

2000 yılı itibariyle çeşitli çevrelerden uzaktan eğitim ile ilgilenen kişi sayısı 3 milyonu aşmış bulunmaktadır (Gubernick ve Ebeling, 1997). Uzaktan eğitimi başarıyla uygulanan en çok bilinen örnek, ulusal destekli ve 133000'ün üzerinde öğrenciye on-line kurslar sunan İngiltere'deki açık üniversitesidir (Encarta, 1998). Bu üniversite eğitimine 1969'da başladı ve dağıtım mekanizması olarak televizyonu kullandı. Daha sonraları bilgisayar tabanlı eğitimi ve özellikle son zamanlarda da digital kütüphaneleri ve multimedya imkanları kullanmak üzere kendini geliştirdi. Şu anda dünya çapında 20'nin üzerinde ülkede ulusal destekli açık üniversiteler bulunmaktadır. Bunun yanında 2500'den fazla kurs imkanı sunan 100'ün üzerinde üniversite de Internet kanalıyla eğitim verilmektedir (LaRose vd., 1998). Uzaktan eğitimin bu şekilde hızla yayılmasındaki nedenler şöyle sıralanabilir (Race ve Brown, 1995):

- 1) Düşük maliyetli donanım, yazılım ve telekomünikasyon imkanı.
- 2) Daha genç nesillerin bilgisayar ve Internet dünyasına adım atması.
- 3) Yeni nesillerde en aza indirgenmiş teknoloji korkusu.
- 4) Genel olarak toplum arasında bilgisayar ve imkanlarına daha yüksek erişim.

21. yüzyıla girdiğimiz şu günlerde, özel ve kamusal alanlardaki her türlü girişimler yoğunlaşmış bulunmaktadır. Bu girişimler uzaktan eğitimi pratik bir çözüm olarak zorunlu kılmaktadır ki bu sayede:

- 1) Kişi başına düşen eğitim faaliyetlerinin maliyeti azaltılsın.
- 2) Fiziksel işgücünden ziyade bilgiye değer katan tüm bilgi işçilerinin eğitim olanakları arttırılsın.
- 3) Her yaş ve imkandaki insan grupları için hayat boyu eğitim imkanı sağlansın.

2.1 Uzaktan Eğitim ve Uzaktan Öğrenim

Tabiatı gereği eğitim, spesifik yeteneklerdeki performans ve gelişim üzerinde odaklanmıştır. İş çevrelerinde, edinilmiş olan bilgi birikimini iş ile ilişkili davranış ve hareketlerde uygulamak amaçlı eğitim faaliyetleri sağlanır.

Eğitim daha çok meslek ve firma eğilimlidir ve çalışanlara iş ile ilgili kabiliyetlerinde gelişme ve kazanç sağlar. Hatta bilgisayar kullanımı gibi genel eğitimler, bu ihtiyacı gerektiren pozisyonlardaki çalışanlar veya isteyenler için sık sık tavsiye edilmektedir. Öte yandan,

öğrenim ise kişileri spesifik ve hareketlere direkt etkisi olmayan daha genel ve soyut bilgi birikimine yönlendirir. Örneğin, “Bilgi Sistemlerinin Yönetimi” başlıklı bir kurs, herhangi bir kimse için spesifik bir mesleğe yönelik olmamakla birlikte bilgi teknolojilerinin yönetiminde karşılaşılabilecek önemli konular veya problemleri içerebilir. Bu durumda alınan bu kurs kişinin kendi yetenekleri çerçevesinde spesifik konu veya mesleğine uygulayabilmesiyle alakalıdır, tabii buda yüksek seviyede bir bilgi birikimi ve analitik kabiliyetler gerektirir.

Öğretmek ve öğrenmek terimleri kendi aralarında değişken olarak kullanılabilirken her ikisine de uzaktan kelimesinin eklenmesiyle tamamen farklı iki görüş açısı ortaya çıkar. Uzaktan öğretime öğreten kişinin görüş açısından bakılmalıdır ki bu kişi eğitim öğretim malzemelerini öğrencisi ile aynı fiziksel ortamda bulunmadan aktarma durumundadır. Öte yandan uzaktan öğrenimde öğrenen kişinin görüş açısından bakılarak incelenmelidir. Aslında, uzaktan öğretim ile sıkı sıkıya bağlıdır. Fakat eğer ortamdan veya kişilerden kaynaklanan teknolojiyi kullanma geriliği veya soru cevap sırasındaki bilgi aktarımı eksikliğinden kaynaklanan problemler oluştuğunda öğrenimde olmayabilir.

2.2 Uzaktan Eğitim Teknolojileri

Uzaktan eğitim ve alıştırmalarını desteklemek için bir çok teknoloji kullanılabilir. Bunlardan bazıları geleneksel sınıf ortamına destek olurken diğerleri geleneksel okutman tabanlı derslerin tamamen yerine geçecek şekilde kullanılır. Her teknolojinin belli bir ortama diğerlerinden daha uygun düşebileceği avantaj ve dezavantajları bulunmaktadır. Mesela bazıları öğrenen ve öğretici veya öğrenenler arasında neredeyse hiç etkileşim veya iletişim sağlamaz. Bazı teknolojiler ders malzemesinin yanında sınırlı konu bilgisi sağlarken bazıları ilgili öğrenciye zengin ek bilgi sunar. Bazı teknolojiler öğreti için eş zamansız destek sağlayan daha esnekliğe imkan verirken öğrencileri de kendi hızına göre çalışmasına imkan verirken, diğerleri iletişim imkanlarının daha fazla olduğu eş zamanlı eğitim ortamı ile sınırlıdır. Burada kullanılan teknoloji kelimesiyle kitap, bilgisayar ve haberleşme ağ yapıları gibi eğitim ve öğretime ulaşma imkanı sağlayan araçları kapsamaktadır. Bunlara dağıtım teknolojileri denilmektedir (Clark, 1994).

Ayrıca yönlendirme teknolojileri olarak adlandırılan öğrencilerin öğrenim deneyimlerini etkileyen testler, alıştırmalar, yardım kılavuzları ve örnekler gibi araçlara da yönlendirme teknolojileri adı verilir (Clark, 1994). Bunlara klasik anlamda ders verme grup veya tek kişilik projeler, konu alıştırmaları ve deneysel çalışmalara da dahil edilerek öğretim stratejileri de vardır. Yönlendirme teknolojisi olarak adı geçen terim öğretimin dağıtılması ve

iyileştirilmesine yarayan araçların tüm kombinasyonlarını içerir. Bir kuruluş için ideal teknoloji dersin özelliğine, dinleyicilere, kuruluşun mali imkanlarına ve var olan teknoloji altyapısına bağlıdır. Çizelge 2.1’de uzaktan eğitim teknolojilerinin avantaj ve dezavantajları özet halinde verilmektedir.

2.2.1 Bilgisayar Tabanlı Eğitim

Son yıllarda Bilgisayar Tabanlı Eğitim (BTE) yaygınlaşmıştır. Bilgisayar tabanlı eğitim genelde CD-ROM veya disketlerde dağıtılan bilgisayar üzerinde sunulan eğitim derslerini kapsar. İdeal ortamlardan birisi uygulama sırasında kullanıcıların kendi elleri ile hard drive sabit sürücüye ulaşabilme imkanı ve veriyi saklama imkanı sağlayan CD-ROM’dan verilen derslerdir. BTE kılavuzlar veya dersin tamamı için vasıta olarak kullanılabilir. Öğrenciler kendi hızlarıyla verilen dersin malzemelerini çalışma imkanı bulabilirler. Bir çok işletme öğrenme laboratuvarı ile donanmıştır. Tipik bir öğrenim laboratuvarı derslerin bulunduğu bir kütüphane ve birbirinden bölmeler ile ayrılmış sıralar üzerindeki bilgisayarlar dan oluşmaktadır. Çalışanların öğrenim ihtiyacı, gerektiğinde odada kendilerine bir yer ayırırlar ve kendilerinin ilgisini çeken derse kayıt olurlar. Genelde bu BTE kursları daha geleneksel sınıf içi eğitim için ön hazırlık veya destek şeklinde kullanılır. Geçmişte BTE dersleri ders kitabı ile beraber yürütürken bugün bu kursların çoğu on-line malzeme ile desteklenmektedir. Son yıllarda BTE kursları gittikçe daha çok multimedya dağıtım için geliştirilmektedir. Etkileşimli multimedya yönlendirmeleri öğrencilere bilgisayarı kullanarak etkileşimli alıştırmalarla bilgilerini geliştirme imkanı sağlarken yazılı metin okuma görüntü veya video izleme ve işitsel bilgi alımına da imkan sağlar. Etkileşimli multimedya dersleri ticari birçok araç ve yazılım paketleri ile birlikte geliştirilebilir.

Aşağıdaki Çizelge 2.1’de yönlendirme teknolojilerinin Eğitimci (E), İşletme (İ) ve Öğrenci (Ö) açısından avantaj ve dezavantajları gösterilmektedir. Piyasada BTE derslerinin geniş bir yelpazesi mevcuttur. Bazı ders türleri, bilgisayar programlama gibi özellikle CD-ROM ortamına uygundur. Örneğin FORTRAN dilini ve programlama prensiplerini beraber öğretmek için okullarda learnfortran isimli bir program kullanılır (Bateman ve Simpson, 1995).

Çizelge 2.1 Yönlendirme teknolojilerinin avantaj ve dezavantajları

Yönlendirme teknolojisi	Avantajlar	Dezavantajlar
Bilgisayar Tabanlı Eğitim: Öğreticinin müdahalesi olmadan bilgisayar üzerinden ulaştırılan yönlendirmeler	Zamandan bağımsızlık. (Ö) Öğrenci merkezliliğinin artması (Ö) Modülerlik (Ö) Anında geri besleme (Ö) Cevap zamanının çabukluğu (Ö) Ders programının esnekliği (İ) İşlem maliyetinin ucuzluğu (İ)	Etkileşim eksikliği. (Ö) Ders verenin eksikliği (Ö) Öğrenim ortamının kontrol edilememesi (Ö) Bilgisayara ulaşımın mecburiyeti (Ö) Yer bağımlılığı (İ) Kuruluş bağımlılığı (İ) Yenileme masrafları (İ) Sonuçların kontrol edilememesi (İ)
Video kayıtları: Eş zamanlı olmadan video bantlar ile görsel olarak yapılacak eğitim	Zamandan bağımsızlık (Ö&İ) Mekandan bağımsızlık (Ö) Sınırlı gereksinim olması (Ö) Ders programı esnekliği (İ) Genel masrafların düşük olması (İ)	Geri beslemesiz olması (Ö) Sadece tek bir öğrenim modunun olması (Ö) Etkileşim esnekliği (Ö) Eğitmen eksikliği (Ö) Öğrenim ortamının kontrol edilememesi (Ö) Tekrar gözden geçirmenin güçlüğü (Ö) Sonuçların denetlenememesi (İ) Güvenlik (İ)
Bilgisayar Destekli Eğitim: Düzenli eğitime yardımcı olarak web tabanlı veya bilgisayar tabanlı araştırmalar	Öğrenimin gelişmesi (Ö&İ) Öğrenci merkezliliğinin artması (Ö) Ders programı esnekliği (Ö) Modülerlik (Ö) Anında geri besleme (Ö)	Öğrenciler arasında sınırlı etkileşim (Ö) Koordinasyonun artması(E) Yer bağımlılığı (İ) Gelişim maliyeti (İ)
Web Tabanlı Eğitim: Bir ağ (internet, intranet) ile bağlı bilgisayar vasıtası ile verilen eğitim	Bir çok öğrenim modunun kullanılması (Ö) Öğreti esnekliği (E) Öğrencilerle tek tek ilgilenebilme imkanı (E) İşletme maliyeti (İ) Dersin standartlaştırılması (İ) Gelişim maliyeti (İ) Bedava yazılım (Ö) Mekan bağımsızlığı (Ö&İ) Zaman bağımsızlığı (Ö&İ) Öğrenci merkezliliğinin artması (Ö) Kullanım kolaylığı (Ö&E) Eğitici ile iletişimin artması (Ö) Diğer öğrencilerle iletişimin artması (Ö&E) Pahalı olmayan iletişim	ISP'ye bağımlılık (Ö&E&İ) Düşük hızlı bağlantılar (Ö) Virüsler (Ö&E) Bilgisayar erişiminin zorunluluğu (Ö) Ağ erişim maliyeti (Ö) Malzemenin kalitesi (Ö) Güvenlik ve gizlilik (Ö&E& İ) Artan genel maliyet (Ö) (web'de bilginin kaybolma ihtimali) Elektronik iletişime bağımlılık (Ö&E) Artan koordinasyon (E) Dijital malzeme gerekliliği HTML bilgisi gerekliliği(E)

	araçları (Ö&E) Güncellenebilir dersler (Ö) Her şeyin dijital olması (E) Yer bağımsızlığı (E&İ) Fazladan kaynaklara bağlantı (E) Tekrar kullanılabilirlik (E) İleriye yönelik geliştirme (E&İ) Var olan altyapının kullanımı (İ) Merkezleştirilmiş sonuç takibi (İ)	Telif hakları (E) Yüksek hızda bağlantı gerekliliği (İ) Eğitimci gerekliliği (İ) Standart eksikliği (İ) Altyapı desteği (İ) Kuruluş ve genel masrafları (İ)
Telekonferans uygulamaları Veri sesli veya video konferans araçları vasıtasıyla yapılan, dağıtılan eğitim. Eşzamanlı	Yüz yüze (Ö&E) Öğrenci-öğrenci etkileşimi (Ö) Öğrenci-eğitmen etkileşimi (Ö&E) Yer bağımsızlığı (Ö&İ) Karmaşık problemlerin açıklanması (Ö&İ) İşletme maliyetleri (E) Gözden geçirme kolaylığı (İ)	Bant genişliği zorunluluğu (Ö&E) Ses sınırlaması (Ö&E) Video sınırlaması (Ö&E) Off-line çalışmanın olmaması (Ö) Kuruluş masrafları (İ) Güvenlik ve gizlilik (İ)
Video Tele-Eğitim: Canlı video (bir veya iki yönlü) ve iki yönlü ses kullanılarak yapılan eğitim	Gerçek zaman imkanları (Ö) Güncellenebilir dersler (Ö) Öğrenci-öğrenci etkileşimi (Ö) Eğitimci-öğrenci etkileşimi (Ö) Dinamik güncelleme (E) Öğrencilerle etkileşim (E) Mekan olarak ulaşılabilen yer fazlalığı (İ) İşletme masrafları (İ) Altyapının tekrar kullanılabilirliği (İ)	Ders programı ayarlaması (Ö&E) Mekan bağımlılığı (Ö) Görüntü sınırlaması (Ö) Artan koordinasyon (E) Öğrencilerin katılımının güçlüğü (E) Destek personeline bağımlılık (E) Tecrübe gerekliliği (İ) İşletim maliyetleri (İ) Üretim personeli gerekliliği (İ)

2.2.1.1 Teknik Gereksinimler ve Tasarım

BTE dersleri ile ilgili konulardan biriside kurumun belli bir takım ihtiyaçlarını karşılamaya yönelik ayarlamalardır. CD-ROM'dan bir dersi almanın maliyeti makul olabilirken CD-ROM üzerinde yeni bir ders yaratmanın maliyeti aşırı yüksek olabilir. Genelde ders malzemesi geliştiriminin en pahalı bileşeni yazılım veya donanım değil personel iş gücüdür.

Bir dersin iyi tasarım derecesi dersin yayılımının verimliliği ile doğru orantılıdır (Stanton ve Stammers, 1990). Bu yüzden kurumlar ders malzemesi tasarımına çok dikkat etmelidir. Aşağıda tasarım esnasında en çok dikkate alınması gereken noktalar özetlenmiştir.

2.2.1.2 Multimedya İmkanları

Tasarımcılar statik medya (metin, grafik ve resim) ile dinamik medyayı (ses, video,

Brown, 1995). Hata mesajları bireylerin belli bir yazılım kullanırken veya belli bir konudaki bilgilerini test ederken yaptıkları hatalardan ders almasını sağlayan öğrenim sürecinin bütünleşmiş ve önemli bir parçasıdır. Düğmeler (butonlar) açıklık sağlayacak şekilde tasarlanmalı ve tutarlı bir şekilde kullanılmalıdır. Örneğin, ana menüye dönüş düğmesi bir ekranın sağ üst köşesinde yer alıyorsa her ekranda aynı yere yerleştirilmelidir. Öğrencinin dersin ana işleyişine geri dönüş yolunu kolaylaştırmak için anlaşılması ve takip etmesi kolay surf araçları bulunmalıdır.

2.2.1.4 Belgelerin Düzenlenmesi

BTE yazılımlarının ilk versiyonları hiyerarşik yapı içerisinde bulunan menü tabanlı bir yaklaşım içerisindeydi. Günümüzde BTE ara yüzleri olarak hiperlinkler daha sık kullanılmaktadır. Hiperlinkli web sayfaları ve hiperlinkli ders yazılımları göz önüne alındığında kullanıcılar belli bir konuda daha fazla bilgi edinebilmek için bir kelime veya bir cümleye tıklama fikrine daha yakındır. BTE derslerinde hiperlinkler genelde tanım vermek için kullanılır. Örneğin, Microsoft Corporation'ın tescilli markası olan Encorta türü CD-ROM ansiklopedileri. On-line ders malzemesi geliştirmek için kullanılan bazı araçlar özel olarak hiperlinkli belgeler yaratmak için özel olarak tasarlanmıştır. Hipermetin BTE'ler kullanıcıların çoğu tarafından faydalı bulunmaktadır (Scotney ve Mc Clean, 1995).

2.2.1.5 BTE'lerin İçerikleri

BTE öğrenimi kişinin kendi hız ve gücüne bağlı olduğundan BTE'nin öğrenciyi mümkün olduğunca çeşitli öğrenme aracı ile yüz yüze getirebilmesi önemlidir. Öğrenme sürecinde ne kadar çok duyu kullanılırsa o kadar çok bilgi tutma oranı ve verilen malzemeyi o kadar hızlı alma olduğu gösterilmiştir (Gardner, 1993). BTE'ler klasik ders ortamında verebileceğin

ötesinde bu imkanı sağlar. Öğrenme için bütünleşmiş ortam, video, ses, resim, metin, hipermetin ve animasyonun hepsi beraber kullanılabileceğinden öğrenme için daha ideal bir ortam oluşturabilir. Kullanılan sözcük haznesinin seviyesi hitap edilen kitleyle tutarlı olmalıdır. Son olarak BTE kurslarının bir ağı bağlantı şartı olmaması ve güçlü PC'lerin fiyatlarının makul olması nedeniyle BTE'ler öğrencinin dikkatini çekecek yüksek kaliteli grafikler ile tasarlanabilir.

2.2.1.6 Etkileşimlik

BTE derslerinin önemli noktalarından biriside öğrenci ve ders malzemesi arasında sağlayacağı etkileşimliliklerdir. Artık öğrencinin dizi olarak okuduğu malzeme sadece “sayfa çevrili” dersler hazırlamaya yetmemektedir. Öğrenme verimliliğini yakalayabilmek için ders tasarımcıları aynı zamanda öğrenci bir takım hareketleri yapmaya zorlamalıdır (Race ve Brown, 1995). Etkileşimliliğin öğrenimi daha kişisel yapan anahtar öğelerden birisi öğrenciyi öğrenme sisteminin bir parçası durumuna getirebilmesidir. Ders içeriği ile etkileşim öğrenmeyi daha kişisel yapar. Örnek vermek gerekirse öğrenciler ders veya klavuza başlarken adlarını girmeleri istenir ve bu daha sonrada sistem tarafından öğrenciye geri besleme sağlamak için kullanılır (Race ve Brown, 1995). Race ve Brown geri beslemenin hem “iyi cevaplar” hem de “kötü cevaplar” içermesini tavsiye eder. Öğrenciler aynı zamanda öğrenim sırasında gördükleri malzemeyi pratikte kullanmalarına imkan verecek bir takım işleri disk üzerinde gerçekleştirmelerine imkan vermelidir. BTE derslerinin planlaması, tasarlanması ve kurulması için aşağıdakiler gerekmektedir:

- 1) Yüksek hafıza kapasiteli ve orta-yüksek işlem gücü olan multimedya bilgisayarları.
- 2) Mikrofon, kulaklık, video kartı, ses kartı, yüksek çözünürlüklü ekranlar gibi multimedya çevre birimleri ve CD-ROM sürücüler.
- 3) Adobe Premiere, Adobe Photoshop, çeşitli Quicktime araçları gibi multimedya geliştirme yazılımları.
- 4) İşyeri veya evde öğrencilerin dersleri sakin, sessiz izleyebilme ortamı, CD yazıcıları (eğer dersler kurum tarafından geliştiriliyorsa).
- 5) Hareket kaydedici yazılım. Bu yazılım programları kurs tasarımcısının hareket dizilerini bilgisayar ekranına kaydetmeye yarar. Örneğin, Lotus Screem Cam veya Microsoft'un Direct Show'u.

2.2.1.7 BTE'nin Avantajları

Öğrenci açısından BTE'nin avantajları şunlardır:

- 1) Geçici bağımsızlık: Öğrenciler kendi hız ve uygun koşullarında çalışabilirler.
- 2) Öğrenci merkezliliğinin artması: Öğrenciler çalışma ortamı üzerinde daha fazla kontrole sahiptir ve malzemenin bazıları kendi gücü ve hızına göre çalışırlar. Ses, video, animasyon ve diğer etkileşimli faaliyetler öğrenci ilgisini artırır.
- 3) Modülerlik: Modüler dersleri tasarlama BTE ortamında kolaydır. Öğrenciler dersleri bütün olmak yerine modüler halde alabilir.
- 4) Anında geri besleme: Öğrenciler dersleri alırken yapılmış olan kılavuz ve küçük sınavlar ile çalıştıkları ders başlarındaki performanslarını anında geri besleme ile öğrenebilirler.
- 5) Çoklu öğrenme modu: Öğrenciler kurs ortamındaki tasarlanmış farklı duylara hitap eden ses, görüntü, yazı sayesinde çoklu öğrenme modu kullanılabilir.
- 6) Hızlı cevap zamanı: İletişim olanakları veya interneti kullanarak uzaktaki bilgisayar sistemlerine ulaşmaya kıyaslandığında bilgiye CD-ROM'dan ulaşmak daha verimlidir. Bu kurs tasarımcıların daha yüksek kalitede grafikler ve daha çok sayıda grafik kurs malzemesini katmalarına imkan sağlar.

Eğitmen açısından ise BTE'nin avantajları şunlardır:

- 1) Geçici bağımsızlık: BTE kursları için eğitmen gerekmediğinden kurumun işine gelen zamanda bağımsız eğitmenler grubu tarafından farklı zamanlarda ders geliştirebilir.
- 2) Zaman çizelgesi esnekliği: Büyük sayıda çalışan tarafından arada bir alınan dersler için BTE kursları zaman çizelgesi problemlerini önlemeye yardım eder ve çalışanların verimli zaman kaybını önlemeye katkıda bulunur.
- 3) Maliyet: Çok sayıda çalışan veya öğrenci tarafından tekrarlanan dersler sayesinde öğrenci başına maliyet düşmüş olur. Bunun yanında eğitmen tabanlı eğitim ile kıyaslandığında öğreticinin eğitimi ve seyahat masrafları da göz önüne alınırsa uzun vadede maliyet kazancı çok daha büyük olur.

2.2.1.8 BTE'nin Dezavantajları

Öğrenci açısından BTE'nin dezavantajları şunlardır:

- 1) Öğrenciler arasında etkileşim: Öğrenciler dersi kendilerine uygun zamanda aldıklarından öğrenciler arasında etkileşim kalmaz. Etkileşimin öğrenme sürecinde çok büyük faydası olduğu gösterilmiştir.
- 2) Eğitici eksikliği: Zor kavramların anlaşılmasına veya öğrenim sürecinde öğrenciye rehberlik edecek bir eğiticinin bulunmaması.
- 3) Öğrenim ortamının kontrolü: Kurumlar BTE dersi olan bir öğrenci veya stajyerin çalışma ortamı üzerindeki kontrolü kalmaz veya çok azalır. Örneğin, bazı ev ortamlarında öğrenci, çocuk bakma veya diğer ev işleri ile uğraşma esnasında öğrenmeye çalışıyorsa bu öğrenim için uygun olmayabilir veya öğrencinin sessiz çalışabileceği bir ortam olmayabilir.
- 4) Bilgisayar erişimi: Bir çok kurum BTE kursu almak isteyen çalışanlarına bir büro veya oda ayırır. Tersine fertler şahsi olarak kurs almak istediklerinde kişisel nedenler veya devam eden eğitim programının gereği olarak multimedya destekli PC'nin içinde bulunduğu CD-ROM tabanlı kurslar için gerekli minimum bilgisayar konfigürasyonuna erişme zorunluluğu duyar.

Kurum açısından ise BTE'nin dezavantajları şunlardır:

- 1) Yer bağımlılığı: Kurum eğer çalışma ortamında çok yer kullanıyorsa aynı malzemenin birkaç değişik setini hazırlamalıdır. Örneğin; işletim sistemi farkı (Unix, Windows gibi) veya PC uyumluluğu (Macintosh, IBM gibi)
- 2) Geliştirme maliyeti: Eğer kurumun amaçlarına yönelik geliştirilmiş dersler yoksa, ders malzemesinin kurumun şartlarına göre ayarlanmış tasarım ve geliştirimine ayrılan maliyet önemli boyutlarda olabilir. Buda ancak finansal olarak derse katılacak çok sayıda öğrenci varsa makul olur.
- 3) Yenileme kolaylığı: Dersin içeriğindeki her hangi bir güncellemesi büyük miktarda çabayı gerektirir. Birçok CD'yi tekrar yazmak ve kurs içeriğini değiştirmenin maliyeti yüksek olur. Ayrıca genellikle “modası geçmiş” disklerin atılması gerekir.
- 4) Sonuçların kontrolü: Kurum alınan belli bir BTE kursunun öğrenciler kurs aldıktan sonra test etme araçları kullanma yoluna gitmedikçe genelde öğrencilerin öğrenme seviyelerini ölçmek mümkün olmaz.

2.2.2 Bilgisayar Destekli Eğitim

Aynı temel eğitim teknolojisi yaklaşımını ifade etmek için birkaç değişik terim kullanılmıştır; Bilgisayar Destekli Öğrenim (BDÖ), Bilgisayar Tabanlı Öğretim (BTÖ), Bilgisayar Destekli

Eğitim (BDE). BDE, BTE'ye benzer fakat tipik olarak geleneksel öğrenim metotlarına bir ek veya tamamlayıcı olarak kullanılır. Bu eğitim teknolojileri genelde okuma tabanlı eğitime ek olarak bilgisayar laboratuvarlarında dağıtılan kılavuzlar veya öğrenciler için anında elle girip egzersiz sağlamak için bu teknolojiler genelde sınıfta çalışanları tamamlayıcı olacak şekilde kurulur ve kullanılır. Geçen son on yıllarda BDE araçları şu iki şekilde gözükmiştir; özel konulara odaklanmış, özel olarak tasarlanmış yazılım programları. Örneğin, yabancı dilleri öğrenmek için CD-ROM'lar veya öğretmenler tarafından verilen ve hazırlanan egzersizleri uygulamaya yönelik geliştirilmiş yazılımlar. Örneğin, hesap programları ile uğraşırken MS Excel programı yazmak gibi. Bunlar gerek ev ve gerekse bilgisayar laboratuvarlarındaki bilgisayarda çalıştırılıyordu. Son zamanlarda bilgisayar eğitiminde web üzerinde on-line araçlar sağlayan internet, BDE'de artan bir rol oynamaktadır. Web üzerinde BDE şu anda eğitim amaçlı olarak bir çok akademisyen tarafından kullanılmaktadır. İlk başta web kullananların çoğu sadece bilgiyi dağıtma, postalama mekanizması olarak kullanılıyordu. Fakat günümüzde çok daha fazla insan bunu öğrencileri ile iletişim imkanı veya ek alıştırma sağlamak için kullanılmaktadır.

2.2.2.1 Gereksinim ve Tasarım

BTE kurslarının tasarımına çok benzer. Öğrenci, öğreticiden açıklamaları elde edemediği için BTE ile arasındaki fark BTE'de öğreticiden direk olarak açıklamalar elde edilemediği için bütün bilgilerin içinde olması zorunluluğudur.

BDE'de ise öğretmenler ya normal sınıf ortamında yada telekomünikasyon olanakları ile öğrencilere bilgi aktarabilmektedir. Öyleki modüllerin veya kılavuzların tasarımı daha az bilgi içerebilir ve iletişim olanaklarının ihtiva edilmesi gerekmeyebilir. BDE için kullanılan araçlarından bazıları arasında kılavuzlar, simülasyonlar, yardım kılavuzu veya referans bilgileri bulunur. Tasarım konusunda dikkate alınması gereken bir çok şey BTE'dekine benzer.

2.2.2.2 Multimedya Olanakları

Tasarımcılar statik medya (metin, grafik ve görüntü) ile dinamik medyayı (ses, video ve animasyon) birleştirebilmek için, BDE yazılım geliştirme araçlarını multimedya olanaklarını kullanmalıdır. Bu öğrencilerin değişik duyularına hitap ederek çoklu öğrenim moduna imkan sağlar.

2.2.2.3 Arayüz

BDE yazılımları, kullanılması kolay ve kendi kendini açıklayıcı olmalıdır. Örneğin, anlaşılır hata mesajları ve bir seferde sınırlı sayıda bilgi gösterimi vb.

2.2.2.4 Etkileşim

BTE derslerinde olduğu gibi etkileşim önemlidir ve malzeme ile öğrenci arasında ki etkileşim BDE dersleri veya kılavuzlarına dahil edilmelidir. Bunun içinde sık ve faydalı geri besleme sağlanmalıdır. Bu da öğrenmede önemli roller oynayan “iyi cevaplar” ve “kötü cevaplar” içerilmesi ile sağlanır. Etkileşim öğrencilerin bazı ders notlarını diskete kaydedebilme imkanı ile CD-ROM’da malzemenin bütününe koruyarak, fakat öğrencilere ders ortamını terk ettikten sonra dahi çalışma ve alıştırmaya olanağı sağlayarak daha da arttırabilir.

2.2.2.5 Konu Başlığına Mahsus Olma

BDE dersleri belli bir konu başlığını öğretmeye yönelik hazırlanmalıdır. Sınıf, ders ortamına bütünleyici olarak kullanıldıklarından bu belirli malzemeye çok yakından bağlı olmalıdır. İdeal BDE araçları öğrencinin önceden öğrenilmiş kavramları uygulayabilme olanağına katkıda bulunur. Bu durumda BTE kurslarında olması gerektiği gibi ders ve kılavuzlar ev içi geliştirilmelidir. Fakat bu gerektiği takdirde türetilmiş yazılım paketlerinin kullanımını göz ardı etmez. BDE kurslarının kurulumu ve yaratılması için yazılım, donanım ve diğer gereksinimler arasında şunlar vardır:

- 1) Yüksek hafıza kapasiteli ve yüksek işlem gücü olan multimedya bilgisayarları.
- 2) Mikrofon, kulaklık, video kartı, ses kartı, yüksek çözünürlüklü ekranlar gibi multimedya çevre birimleri ve CD-ROM sürücüler.
- 3) Adobe Premiere, Adobe Photoshop, çeşitli Quicktime araçları gibi multimedya çevre birimleri ve CD-ROM sürücüler.
- 4) İşyeri veya evde öğrencilerin dersleri sakın, sessiz izleyebilme ortamı, CD yazıcıları (eğer dersler kurum tarafından geliştiriliyorsa).
- 5) Ekran hareketini yakalayan yazılım.

2.2.2.6 BDE’nin Avantajları

Öğrenci açısından BDE’nin avantajları şunlardır:

1) Öğrenimin artması: BDE'nin, öğrencilerin öğrenim yeteneğini arttırdığı söylenmektedir (Darby, 1992).

2) Artan öğrenci merkezliliği: Öğrenciler kılavuz veya ek malzemeyi kendi hızları ve kendilerine uygun zamanda çalışabilirler. Bu özellikle dersin hızına yetişemeyen fakat kendi başlarına çalıştıklarında eksik bilgiyi bulan ve derse yetişebilme yeteneği olan öğrenciler için özellikle önemlidir.

3) Ders programı ayarlama esnekliği: Öğrenciler sınıf çalışmasındaki dersleri tamamlamak için kendilerine hangi bilgisayar modülünün uygun olduğuna karar verme esnekliğine sahiptir. Bu da onların ders programını ayarlayabilme esnekliğini önemli ölçüde artırır.

4) Dersin modülerliği: Modüler ders tasarımı BDE ortamında kolaydır. Modüler ders malzemesi için yararlı ilavelerdir ve gerekli ön bilgi malzemesi görüldüğünde uygun bir zamanda verilebilirler. Öğrenciler için bu sınıfta görülen her konunun gerektirdiği deneysel çalışmaları yapabilme imkanı sağlarken onları aşırı bilgi ve ev ödevi ile yüklemmez.

5) Anında geri besleme: BTE'lere benzer olarak öğrenciler daha önce yapılmış kılavuzları ve sınavları kullanarak çalıştıkları konu başlığındaki performansları hakkında anında bilgi edinilebilirler.

6) Çoklu öğrenme modu kullanmak: BTE'dekine benzer bir şekilde dersler öyle bir şekilde tasarlanmalıdır ki öğrenciler ders malzemesi ve ortamında oluşturulabilecek çeşitli duyulara hitap eden metotlar ile (ses, görüntü, metin, yazılı metin) çeşitli çalışma modları ile öğrenim yapabilsin.

Eğitmen açısından BDE'nin avantajları şunlardır:

1) Öğretim esnekliği: Eğitmenler BDE araçlarını kullanarak ders malzemesi ile birlikte özel modüllerin birleştirilebilme imkanına kavuşurlar. Bu eğitmenleri gereksinim duyduklarında sınıf içerisinde kavramlara daha fazla vakit ayırma fırsatı verirken, öğrencilere de dersin pratik kısmını kendi başlarına çalışmalarına imkan verir.

2) Öğrencilerle şahsi olarak ilgilenme: Öğrenciler BDE araçları ile kendi hızlarına göre çalışırken, eğiticiye de daha çok ilgi gerekenlere daha fazla vakit ayırma imkanı sağlar.

Kurumlar açısından ise BDE'nin avantajları şunlardır:

1) Düşen işletme maliyetleri: Bir çok sayıda öğrenci tarafından bilgisayar üzerinde önemli miktarda alıştırma gereken dersler de, bu büyük grupların öğretimi veya alıştırılması için daha

az eğitime gereksinim duyulabilir. Gruplar, dersin klasik ders anlatımı gerektiren konularında büyük gruplar halinde öğrenim görüp, sonradan kılavuzları kendi hızlarına göre tek tek veya küçük gruplara ayrılarak çalışabilirler.

2) Dersin standartlaştırılması: Potansiyel olarak büyük sayıda öğrencinin alacağı derslerde BDE araçları bütün öğrenciler tarafından kullanılacak malzemeyi nicelik ve nitelik olarak belli bir dereceye kadar standartlaştırılmasına yardım eder. Örneğin, çok sayıda öğrencinin benzer ders alması gerektiğinde malzemeyi öğretmek için birkaç eğitici tayin edilebilir. Öğretmenler arasında tarz farkları olabilirken eğer bütün öğrenciler aynı BDE aracını kullandıkları takdirde alıştırmayı yaptıkları, ders bileşenleri standartlaşmış olur.

3) Daha düşük ders geliştirme masrafı: Öğrenim araçlarından bazılarının gelişimini merkezleştirerek, kurumlar öğretmenlerin odaklaşması gereken diğer önemli öğretim ve alıştırmayı konularında daha fazla vakit ayırabilmelerine olanak sağlar.

4) Daha iyi öğrenim: BDE'nin ders ve programlarda kullanımı üzerine 100'den fazla değerlendirmenin yapıldığı kapsamlı bir alıştırmayı da aşağıdaki sonuçlar bulunmuştur (Kulik, 1994).

- Öğrenciler bu sınıflarda daha çok öğreniyorlar.
- Öğrenciler daha az zamanda derslerini öğreniyorlar.
- Öğrenciler sınıflarını daha çok seviyorlar.
- Öğrenciler bilgisayarlara karşı daha olumlu davranış geliştiriyorlar.

2.2.2.7 BDE'nin Dezavantajları

Öğrenci açısından BDE'nin dezavantajları şunlardır:

1) Öğrenciler arasında sınırlı etkileşim: Eğer öğrenciler klasik ders ortamında derslerinin BDE olmayan kısmını tamamladıklarında bir dereceye kadar etkileşim vardır. Eğer BDE diğer iletişim destekli uzaktan eğitim veya alıştırmayı dersleri ile birleştirilmişse öğrenciler arasında etkileşim olmamaktadır. Etkileşim genelde hemen bütün öğrenciler için faydalı bir eğitim unsurudur.

Eğitmen açısından BDE'nin dezavantajları şunlardır:

1) Artan koordinasyon: Derslerinde BDE araçlarını benimseyen öğretmenler genelde özel bilgisayar laboratuvarları veya internet bağlantılı bilgisayarlarda sunulan BDE kısmını

koordine etmek için çok büyük çaba harcarlar. Bu koordinasyonun yanı sıra ders için gerekli klasik malzemeleri hazırlamaları ve dağıtmaları da gerekir.

Kurum açısından ise BDE'nin dezavantajları şunlardır:

- 1) Platform bağımlılığı: BTE'dekine benzer olarak kurum eğer çalışma ortamı içinde birden fazla platform kullanılıyorsa (örneğin, Unix ve Windows veya Macintosh ve IBM uyumluluk gibi) aynı malzeme için farklı versiyonlar hazırlanmalıdır.
- 2) Gelişim masrafları: BDE malzemesi genellikle belli bir takım konu başlıklarına uygun olarak hazırlanır. Bu nedenle genelde kurumun kendi çalışanları tarafından geliştirilir. Gerçi matematik öğretimi veya programlama dilleri için hazır genel amaçlı kılavuz türü yazılım programları bulunmaktadır. Fakat üretilmiş ders malzemesi bulunmuyorsa bu dersin malzemesini geliştirme masrafları yüksek olur. Bu durum ancak dersi alacak çok sayıda öğrenci olduğunda ve ders malzemesi zaman içerisinde göreceli olarak sabit kalacaksa uygun olur.

2.2.3 Web Tabanlı Eğitim

Web Tabanlı Eğitim (WTE), eğitimin dağıtılması için temel yayın vasıtası olarak internet veya duruma göre intranet kullanılır. Her ne kadar web tabanlı olarak tanımlansa da bu tür eğitim teknolojisinde kullanılan bazı araçlar, örneğin dosya transfer protokolü (FTP), usenet, internet mail, telnet, listservs veya chat odaları web tabanlı değildir. WTE şu iki formdan birinde olabilir:

- 1) BDE'nin bir şekli olarak klasik türden dersleri destek amacıyla kullanılır.
- 2) Klasik sınıf ortamında verilebilecek bir dersin yerine kullanılır.

Bu ikinci şekildeki dersler web tabanlı dersleri, siber dersler veya sanal dersler ile de anılır. Bu derslerde WTE öğrencilerinin kendi hızlarında çalışmalarına imkan veren eğitimde ve alıştırmalarda bu teknoloji de öğrenci merkezliliğin önemi artmış olur. Bir de BDE eğitim aracı olarak WTE genelde sadece bir bilgi dağıtım aracı gibi kullanılır. Bu fonksiyonda özellikle iyi tasarlanmış hipermetinler sayesinde sağlar. BTE dünyanın herhangi bir yerinden, herhangi bir zamanda ders malzemesinin dağıtılmasına fırsat verir ve akademik çevrelerde daha önce görülmemiş bir popülerlik kazanmıştır. WTE derslerinin bir karakteristiği hipermedyayı sonuna kadar kullanmalarıdır.

WTE, öğrencilerin kendi arasında ve eğitmen ile öğrenciler arasında elektronik posta (e-mail),

bilgisayar konferansı yada listservs ve chat imkanlarıyla iletişime imkan verir. Bu etkileşim sunumu, WTE'yi iletişimin birinci derecede önemli olduğu uzaktan eğitim ve alıştırmalar için kullanılan bilgisayar tabanlı diğer eğitim teknolojilerinden daha uygun kılar. Örneğin, iletişim veya gazetecilik derslerinde bilgisayar gerçek anlamda öğrencilerin yazılı veya sözlü sunularını yorumlayamaz ve ölçme-değerlendirme yapamaz. Bu yüzden WTE programları ve derslerinin değerlendirilmesi yapılırken tek başına web tabanlı dersler ve iletişim destekli web tabanlı dersler arasındaki ayırımı dikkat edilmelidir.

WTE türleri aşağıdaki gibi sıralanabilir:

- 1) Eş zamansız haberleşme,
- 2) Eş zamanlı işbirliği,
- 3) Web ile iyileştirilmiş dersler,
- 4) Web yönetimli kurslar,
- 5) Web ile dağıtımı yapılan dersler,
- 6) Karma dağıtım (CD-ROM , internet , vs..).

2.2.3.1 Genel Olarak WTE'de Gereksinimler ve Tasarım

WTE malzemesi tasarımı günümüzde çoğu Internet'te bulunabilecek kolayca erişimi mümkün olan çok çeşitli araçlar sayesinde kolaylaşmıştır. Tasarımda göz önüne alınması gereken üç önemli konu başlığı şunlardır:

- 1) Teknik olarak web tabanlı dersleri nasıl geliştirilmeli?
- 2) Web tabanlı ortamda ne sağlanmalı?
- 3) Dersin gerçek içeriği ne olmalı?

İçerik bazı konuların Internet ortamında daha kolay biçimde belirlenebilmesine imkan sağlarken, örneğin veri iletişimi dersleri gibi genelde her derste kendine özgü olur. Herhangi bir ders konu başlığı şöyle yada böyle web tabanlı içeriğe dönüştürülebilir. Web tabanlı ortamda sağlanması gereken bazı gereksinimler şunlardır:

- 1) Yardım sistemi: Yardım sistemi kolay ulaşılır ve anlaşılır olmalıdır. İçeriğe bağımlı olarak tasarlanmalıdır.

2) Ara yüz: Ara yüz, basit ve standart olmalıdır. Kullanılan düğmeler (butonlar) anlaşılır olmalı, amaçlarının ne olduğu belli olmalı ve bütün ders sayfalarına aynı yerlere yerleştirilmelidir. Bütün web sayfaları okuması ve anlaşılması kolay olmalıdır.

3) Görüntü: Video ve diğer animasyon görüntülerinin içeriği boğmamasına dikkat edilmelidir. Görüntüler genellikle göze hoş gelir, ancak bilgiye bir değer kattığında kullanılmalı ve küçük tutulmalıdır. Ayrıca görüntü alma imkanı olmayan veya yükleme çok uzun süreci için bunları kullanmak istemeyen öğrenciler için alternatif olarak mutlaka metinde bulunmalıdır.

4) Yapı: Web sayfalarının yapısı ders içerisinde kolayca gezinim ve dersin geliştirilebilmesine imkan sağlayacak şekilde tasarlanmalıdır. Anlaşılır bir yapı sitenin bakımı içinde yardımcı olur. Öğrencilerin kaybolmamaları için dersin haritasına bir bağlantıda standart düğme olarak yerleştirilmelidir.

5) Sesler: Web sayfalarındaki sesli bölümler öğrencilere önemli derecede ek bilgi sağlayabilir. Ayrıca farklı bir duyuya da hitap ettiği için öğrenim potansiyelini arttırabilir. Fakat sesleri ders sayfalarına dahil ederken dikkatli olunması gerekir. Çünkü bir çok öğrencinin bilgisayarında ses dosyaları çalışmayabilir. Bu yüzden kliklemeden sonra eğer ses dosyası açılmadıysa öğrencinin ne yapması gerektiği konusundaki talimatlar da tasarımda göz önüne alınmalıdır.

WTE malzemesi tasarımda dört yol önerilmektedir (McCormack ve Jones, 1997):

a) Dağıtım: Hemen her standart formatta ki belgeler HTML uzantısı sayesinde hiç çevrim yapılmadan direkt olarak bağlanabilir. Belgeler sunucu da orijinal formatlarıyla saklanır ve dağıtılır. Bir öğrenci bir belgeyi görmek istediğinde sadece bağlantı (link) üzerine klikler. Tarayıcı dosya uzantısını tanır ve öğrencinin bilgisayarında uygun yazılım varsa belgeyi açar. Bu yaklaşım en büyük avantajı web sayfaları yaratırken sadece bağlantı uzantıları kullanılması gerektiğinden (HTML gibi) büyük zaman tasarrufu sağlar. En önemli dezavantajı ise uygun yazılımın öğrencinin bilgisayarında bulunmadığı zamandır. Bunu çözmek için öğrenci dosyaları uygun bir ortama (disk, yazılabilir CD, vs.) taşır.

b) Çeviri: Buradaki fikir ise MS Word dosyaları gibi var olan malzemeyi alıp HTML çeviricilerinden geçirerek çalıştırmaktır. Bunlardan bazılarını saymak gerekirse Microsoft ürünleri için Internet Assistant, Latex çevirisi için Latex2 HTML veya Rich Text Format çevirisi için Rtfoweb. Çevirici kullanarak HTML oluşturulurken öğrenci bilgiye ulaşmak için sadece bir tarayıcıya gereksinim duyar. Fakat ders tasarımını geliştirmek için hipermetin ve multimedya imkanlarını kullanmaya fırsat vermemesi gibi bir dezavantaj vardır. Ayrıca bazı

çeviriciler açık olmayan veya hatalı HTML kodları üretebilir.

c) Katma değerli çevirim: Bu biraz önceki dosyaların çevrimini gerektirir. Fakat aynı zamanda eğitmen ve geliştirici hipermetin ve multimedya olanaklarını kullanabilmek için çaba sarfeder. Bu da bazı görüntü ve diğer dosyalara bağlantı için kod yazılmasını gerektirir. Avantajı daha öğrenci merkezli bir yaklaşım olmasıdır (öğrenciler malzemeyi sadece okumayıp, aynı zamanda keşfederler). Dezavantajı ise çok daha fazla zaman harcanmasına rağmen web ortamının imkanları tam anlamıyla kullanmış olmaz.

d) Yaratma: En çok sarf edilmesi gereken ders geliştirme seviyesi budur. Dersler web tabanlı ortamların bütün imkanlarının kullanılabilmesi için özel olarak yapılır. Kurs geliştirenleri iletişim kolaylıklarını ders tasarımı içerisinde değerlendirebilir. Ayrıca web'in ses, video ve grafik olanaklarından mümkün olduğunca yararlanan sayfalar tasarlayabilirler. Tabiki bu seviye en zor, en çok zaman gerektiren ve maliyet en yüksek yaklaşımdır. İnternet üzerinde bedava veya ücretli bir çok web yazım aracı bulunmaktadır. Ayrıca Toolbook, Webfuse veya Course Info gibi çeşitli yazılım paketleri de kolaylıkla sağlar.

2.2.3.2 İletişim İmkanları Olarak WTE'de Gereksinimler ve Tasarım

Öğrenciler WTE derslerini uzak bir ortamdan alırken diğer öğrencilere, eğitmenlere veya kılavuzlara erişebilirler. WTE'nin BTE'ye göre en büyük avantajı budur. İlaveten öğrenciler arasındaki iletişimin öğrenimi iyileştirdiği gözlenmiştir. Bu yüzden uzaktan eğitim ortamlarının bu şekilde iletişimi destekleyecek şekilde tasarlanması önemlidir. Bu iletişim araçlarından;

1) Elektronik posta (e-mail) ve listservs: Elektronik posta paketleri son yıllarda çok büyük gelişmeler göstermiş ve bütünleşmiş büro araçları, grafik kullanımlı ara yüzler, güvenlik özellikleri gibi özellikleri de sunmaktadır. Ayrıca elektronik posta uzun süredir kullanıldığından öğrencilerin bir çoğu için tanıdık bir araçtır. Eğitmenler bütün öğrencilere kolaylıkla ders malzemelerini elektronik ortamda yollayabilir. Fakat eğitmenlerin verilen cevapları, cevaplama süreleri standart olmadığından sorun olabilir. Ayrıca mesaj sistemlerindeki büyük gelişmelere rağmen hala elektronik posta ortamları büyük dosya ve ilişkilerin bütün kullanıcılar tarafından ulaşılmasına imkan verir. Elektronik postanın avantajlarından birisi çeşitli yazılım paketlerinin bedava veya ortak yazılım olarak daha hafif versiyonlarının bulunmasıdır. Ayrıca günümüzdeki tarayıcılarda elektronik fonksiyonları da dahil edilmiş olup kurs tarayıcıları kılavuzların sonunda veya çok karmaşık malzemelerde e-posta hiperlink'i koyarak eğitmene veya kılavuza otomatik olarak e-posta atılmasına imkan

verir. Elektronik posta kullanılması pahalı değildir. Ayrıca eğitmene önemli bilgileri zamanında yollayabilmelerini sağlar. Böylece öğretmen daha önceden belirlenmiş zamanlar veya web sitesinin yenilenmesi dışında yeni bilgi gönderme imkanı bulur. Dezavantajlarından birisi yüz yüze etkileşimle karşılaştırıldığında daha verimsiz bir iletişim aracı olduğu gösterilmiştir (Datt ve Lengel, 1986). Ayrıca ders için önemli mesaj öğretmenin posta kutusunda ve hatta bazen öğrencilerin posta kutularında kaybolabilir. Sonuç olarak, öğrenciler arasında bilgi alışverişi gerekiyorsa bütün öğrencilerin öğretmenin e-posta'larını bilmesi gerekir. Bu sorun genellikle ortak bir liste oluşturularak giderilebilir.

2) Bilgisayar konferansı: Aynı zamanda Usenet, kullanıcı tartışma forumları adlarıyla da anılan bilgisayar konferansları bir diğer eş zamansız iletişim aracıdır. Bilgisayar konferansı iletişim altyapısı olarak e-posta formunu kullanır. Fakat iletinin bilginin kullanılması farklıdır. Listeye yollanan bilgi o andaki tartışmaya eklenir. Böylece grup üyeleri arasındaki bütün bilgi alışverişinin kaydını tutmak mümkün olur. Öğrenciler daha sonra kendi yorumlarını eklemeden daha önce ne tartışıldığını okuyabilir. Zamanla öğretmen soru ve cevapların listesini biriktirip web tabanlı ders malzemesini sık sorulan sorular (FAQ) kısmına ekler. Konferans bölümleri sunucu üzerinde olan ulaşım yetkileri ve genelde öğretmen olan bir ara birim gerektirir. Kurulması oldukça kolay olduğundan çok fazla çaba sarf etmeden WTE derslerinin tasarımına eklenebilirler. Ayrıca bu olanakları kullanabilmek için çok fazla yazılım veya donanım gereksinimi yoktur. En büyük avantajlarından birisi gerektiğinde başvurulabilecek grup üyeleri arasındaki iletişim ve tartışmalarının kaydının tutulabilmesidir. Dezavantajlarından birisi verimsiz bir iletişim ortamı olmasıdır. Elektronik posta ile kıyaslandığında bilgisayar konferansının en büyük dezavantajı ders boyunca kurmak ve aracı olmak için çok daha fazla sistem yönetimi gerektirmesidir. Bu sistem yönetimi genelde öğretmen veya bir yardımcısı tarafından yapılır.

3) İlan tahtası: İlan tahtaları da tek yönlü, eş zamansız iletişim araçlarıdır. Genelde öğretmen uzaktan ulaşımı kolay olan ilan tahtasına bilgiyi yollar. Web tabanlı ders tasarımında bilgi iletişimi daha verimli şekillerde yapılabildiğinden ilan tahtaları pek kullanılmaz. Çünkü ilan tahtaları da tek yönlü ve eş zamansız aynı zamanda durağandır.

4) Web tabanlı sohbet (chat) imkanları: Genelde metin komutları ile oyun için kullanılan bu fonksiyonlar uzaktan eğitim ve alıştırma dersleri içinde faydalı olabilir. Çeşitli kullanıcılar ortak bir ekran alanını kullanarak birbirleriyle iletişim kurabilirler. Oturumdaki herkes ya gruba yönelik genel yorumlar bütün kullanıcıların ekranında görüntülenebilir. Öğretmenler bu oturumları problemleri tartışmak veya grup iletişimini arttırmak için kullanılabilir. Öğrenciler

aynı zamanda chat oturumlarında zor konular hakkında birbirlerine yardımı olabilirler. İnternet üzerindeki bedava yüklenebilecek Chat oturumları kullanmanın avantajı, çoktan-çoka iletişim türüne imkan vermesidir. Böylece grup hem kendi içinde hem de eğitmenle iletişimde bulunabilir. Aynı zamanda eş zamanlı olduğundan sorular gerçel zamanlı olarak cevaplanabilir. Son olarak WTE ders altyapısı hazır ise web tabanlı chat olanklarını eklemek masraflı değildir. Dezavantajları arasında ise toplantının yapılacağı zamanlarda bütün katılanların hazır olmasının gerekliliğidir. Ayrıca bu iletişim vasıtası metin elle yazılmakta olduğundan verimsiz olmaktadır. Ayrıca klavyeyi iyi kullanamayan öğrenci dezavantajlı olmaktadır. Son olarak eğitmen bilgisi haricinde chat oturumları düzenlenirse değişilen bilginin doğruluğu ve niteliği üzerinde herhangi bir kontrol bulunmayacaktır.

2.2.3.3 Gereksinimler

1) Donanım: Sunucular, istemciler ve ağ yapısı

a) Sunucular: BTE ve BDE dersleri ile kıyaslandığında WTE sadece kişisel bilgisayarlara değil aynı zamanda yüksek işletim gücüne erişim gerektirir. Web sayfalarının saklanması ve işlenmesi için yüksek kapasiteli sunuculara ihtiyaç duyulur. Çünkü hem bütün ders sayfaları için gerekli yeterli saklama kapasitesi hem de kabul edilebilir cevap süresi sağlanması için gerekli işlem gücüne ihtiyaç vardır.

b) İstemci: İstemci iş istasyonları sunucularda olduğu kadar yüksek işlem gücü gerektirmez. WTE sadece bir web tarayıcısı kullanımı gerekli kılar. Daha gelişmiş özellikler daha yüksek işlem ve derleme gücü olan kişisel bilgisayarları zorunlu yapar. Kullanılan kişisel bilgisayarlar gücü ne olursa olsun, ekranlar mümkün olduğunca yüksek çözünürlüklü olmalıdır. Ne yazık ki web tabanlı sayfalar tasarlanırken birçok tasarımcı üç boyutlu karmaşık şekiller veya simule edilmiş resimler koyarlar fakat öğrenciler ekranlarından bu görüntüleri açık bir şekilde göremeyebilirler.

c) Ağ yapısı: Gerçek anlamda İnternet tabanlı olması için kurumun TCP/IP ağ yapısı bağlantısı sağlaması gerekir. Ayrıca bir çok WTE dersi uzun bağlantı süreleri gerektirdiğinden ağ yapısının bant genişliği yüksek seviyeli veri transferini halletmeye yeterli olmalıdır. Bu, iletişim araçları yazılım ve donanım için önemli masraflar yapılması demektir. Ayrıca yedekleme ağ yapıları ve gelişmiş ağ yapı yönetim araçları da gerekli olabilir.

2) Yazılım: Web tarayıcıları, elektronik posta paketleri, ağ yapısı yazılım.

İnternet üzerinde bir çok yazılım bulunduğundan dolayı yazılım için çok masraf yapılması

gerekmeyebilir. Daha gelişmiş ağ yapısı paketleri ve geliştirme araçları pahalı olabilir. Fakat genelde esas masraflar insan gücü masraflarıdır.

2.2.3.4 WTE'nin Avantajları

Öğrenciler açısından WTE'nin avantajları şunlardır:

- 1) Bedava yazılım: Çok kullanılan web tarayıcıları, Internet üzerinde bedava bulunmaktadır. Öğrenciler için masraf çıkarabilecek tek yazılım bazı malzemelere ulaşmak için gerekli kullanıcı yazılım programlarıdır. Örneğin, Microsoft Word vs.
- 2) Coğrafyadan bağımsızlık: Daha önce yer sorunu ile ders ortamında bulunamayacak kişiler derse katılabilir. Internet iletişim ve ders yayılma için temel olarak kullanıldığından WTE ile gerçek dağıtımli öğrenim sağlanmış olur.
- 3) Zaman bağımsızlığı: Öğrenciler web üzerinde kendi hızlarında ve kendilerine uygun vakitlerde çalışma imkanı bulurlar.
- 4) Artan öğrenci merkezliliği: Öğrencilerin çalışma ortamı üzerindeki kontrolü artar. Bazı malzemeler için kendi hızlarıyla çalışabilirler. Hipermetinler sadece okumaya oranla daha fazla keşfetme imkanı sağladığından öğrencinin katılımında artar. Öğrenci merkezliliğinin, öğrenimi kolaylaştırıldığına inanılmaktadır.
- 5) Kullanma kolaylığı: Günümüzde bir çok öğrenci web tasarımına aşina olduğundan WTE derslerinin kullanımı kolaydır.
- 6) Güncel dersler: Web ortamının dinamik yapısı itibariyle, eğitmenler gerektiğinde ders malzemesini güncelleyebilir. Öğrencilere toplum veya kurumdaki değişiklikler zamanında aktarılabilir. İlave olarak, iletişim destekli WTE araçları aşağıdaki avantajları sağlar.
- 7) Eğitmenle iletişimin artması: İletişim olanaklarından faydalanan web tabanlı ortamlarda ders alan bireyler eğitmenle diğer uzaktan eğitim teknolojilerinde olduğuna kıyasla daha fazla iletişimde bulunabilirler.
- 8) Diğer üyeler ile artan iletişim: Öğrenciler arasındaki iletişim öğrenim kalitesini arttırdığı gözlemlenmiştir. İletişim tabanlı WTE dersleri eşzamanlı veya eşzamansız olarak dersi alan diğer kişilerle bilgiyi paylaşma veya bir diğerinin probleminin çözümüne yardım edilmesine imkan sağlar.
- 9) Pahalı olmayan iletişim araçları: WTE ortamlarında bulunan iletişim araçlarının çoğu ucuz

kullanımı kolay ve/veya geliştirilmesi nispeten kolay araçlardır.

Eğitmen açısından WTE'nin avantajları şunlardır:

- 1) Her şeyin dijital olması: Öğrenciler tarafından Internet vasıtasıyla gönderilen bilgi dijitalleştirilmiş olduğundan performans ölçme ve değerlendirilmesinde eğitmenin işi nispeten daha kolay olur. Eğitmenler öğrencilerin el yazılarını çözmeye uğraşmazlar.
- 2) Platform bağımsızlığı: WTE dersleri web üzerinde yayınlandıktan sonra erişim, yazılım veya donanım platformundan bağımsız olur.
- 3) Kullanım kolaylığı: Grafik web ara yüzlerinin kullanım kolaylığı eğitmenin kullanım talimatları için daha az süre harcanmasını ve dolayısı ile dersin içeriğine daha fazla vakit ayırabilmesini sağlar.
- 4) Dinamik güncelleme: Ders bakımı yapılmadığı takdirde ders ileri bir zamanda “modası geçmiş” olabilir. Web üzerindeki derslerin güncellenmesi daha kolaydır.
- 5) Ek bilgi: Ders malzemesinin dışındaki ek bilgiler web sitesinde ekranı kalabalıklaştırmadan dahil edilebilir. Hiperlinkler vasıtası ile bir çok referans malzemesi sunulabilir.
- 6) Tekrar kullanılabilirlik: Web üzerindeki dersler tasarlanır ve kurulurken malzeme modüller içinde organize edilebilir. Böylece başka eğitmenler, başka dersler için bu modüllerin bir kısmını tekrar kullanabilir.
- 7) İleriye dönük geliştirme: Web üzerinde daha önce kurulmuş bir derse eğitmenlerin ekleme yapması kolaydır. İhtiyaç duyulduğunda yeni modüller eklenebilir ve ek bağlantılar kolayca oluşturulabilir. Bunların yanında iletişim destekli WTE'nin eğitmen açısından faydaları aşağıdaki gibi sıralanabilir.
- 8) Öğrencilerle artan iletişim: Öğrencilerin eğitmene ulaşım kadar, eğitmende öğrencilere ulaşımı sayesinde bazı kazançlar olur. Örneğin, ders içeriğinin konusunda geri besleme sağlanabilir. Öğrencilerin yaptığı sunulara erişebilir ve öğretimde verimlilik sağlanır.
- 9) Pahalı olmayan ve kullanımı kolay kurulum araçları: İnternette bir çok iletişim aracı kullanıma hazır olarak yüklenebilir ve var olan web sayfasıyla kolayca bütünleştirilebilir.

Kurum açısından ise WTE'nin avantajları şunlardır:

- 1) Mevcut altyapının kullanımı: Bir çok firma kendi içinde kullanıldığı internet ve intranet alt yapısının WTE ve bilgi transferi içinde kullanılabilir.

2) Coğrafyadan bağımsızlık: Kurumlar derslere direkt katılma sorunu yaşayan öğrencilerin sorunlarını WTE ile aşarak Pazar paylarını büyütebilirler. Ayrıca kurumlar ulaşım sorunu yaşayan öğrencilere video tele eğitimi sistemi kurmadan, bu sorunları dikkate alarak daha uygun dersler önerebilir.

3) Zamandan bağımsızlık: Takvim ve program sorunu nedeniyle derslere gelemeyecek bireyler için daha avantajlı bir ortam oluşturur.

4) Yerden bağımsızlık: Özellikle öğretmenler ve kurumlar ders tasarımı yaparken farklı sistemler içeren ortamlar kurmaları gerekmez, çünkü yapılacak iş web tabanlıdır. Örneğin, farklı farklı işletim sistemleri gerektiren laboratuarlara ihtiyaç yoktur. Sadece bir web tarayıcısıyla tüm sistem çalışabilir.

5) Merkezi sonuç takibi: BTE ile kıyaslandığında bir çok yönden WTE sonuçları takip etmekte daha avantajlı olmuştur.

6) Gelecekte büyüme: Kurumlar için sistemi bir kere kurduklarında geliştirmek oldukça kolaydır istemci/sunucu modüleritesi oluşuktan sonra ağ yapılanması daha hızlı ve sağlıklı büyüyecektir.

2.2.3.5 WTE'nin Dezavantajları

Öğrenci açısından WTE'nin dezavantajları şunlardır:

1) Servis sağlayıcısının güvenilirliği: Derslere internetten katılan öğrencilerin servis sağlayıcısındaki herhangi bir sorun halinden derslere katılamamasını söz konusudur.

2) Düşük hızda bağlantı: Evden internete bağlanan bir çok kullanıcı düşük hızda ve kapasitede modemler kullanılmaktadırlar (14,4 Kbps den 33,6 Kbps). Ayrıca normal olarak kullanıcı uzun süre bağlantıda kalmak zorunda olacaktır. Eğer ki pek çok sayfanın indirilmesi (download) ya da az ama büyük boyutta dosyaların indirilmesi gerektiğinde bu ciddi bir sorun haline gelmektedir. Eğer ki herhangi bir grafiğin indirilmesi söz konusu ise bu süre bilgisayarda yapılması gereken diğer işlemlere de engel olur. Diğer taraftan eğitimcilerin kullanılmasını istedikleri bazı yazılımlar ise bilgisayarlarda ek sürücü ve alan zorunluluğu kılmaktadır.

3) Virüsler: Veri transferi esnasında virüs dağılımı daha kolay ve kontrolsüz olmaktadır.

4) Bilgisayar gerekliliği: Tüm kullanıcılarda en az bir web tarayıcısı ve minimum zorunlulukta yazılım gerekmektedir.

5) Ağ'a bağlanma ücreti: Evlerinden internete bağlanması gereken kullanıcıların evlerinde bağlantıyı sağlayacak olanaklara ya da işçilerin bu konuda maddi olarak fazla bir sıkıntısı olmazken diğer kullanıcıların bir ISS'den Internet bağlantısı satın almaları gerekmektedir.

6) Malzemenin kalitesi: Web bazlı malzemelerin en büyük sorunlarından biri sağlanan bilginin kalitesinin kontrolünün zorluğudur. Bu bağlamda Internet üzerinden sağlanan malzemenin ortak bir kalite standardına ulaşması oldukça zordur. Resmi diye bilinen pek çok sitede pek çok bilgi yanlışlıkla karşılaşılabılır. Başka bir siteden alınan bilgilerin doğruluğu çoğu zaman tam olarak doğru ve sağlıklı olmayabilir, bunun sebebi de isteyen herkesin Internet ortamına güvenilirliği onaylanmamış bilgiyi koyma rahatlığıdır.

7) Güvenlik ve gizlilik: WTE'de sürekli ve büyük oranlarda bilgi transferi yapıldığı ve bu bilginin kişisel boyutunun olmamasından dolayı (güvenlik sistemi kullanılsa bile) öğrencilerin bu konuda bazı sorunlar yaşaması olasıdır. Örneğin kişiler sayfaya kullandıkları kullanıcı adı ve şifre gibi bazı bilgiler başkalarının eline geçerse bu sorun yaratır.

8) Toplam maliyet yüksekliği: Öğrenciler WTE imkanlarından faydalanmak için multimedya özellikleri fazla olan kaliteli bilgisayarlara ihtiyaç duyarlar. Ek olarak uzun süre Internette bağlı olarak da kalmak zorunda oldukları için ikinci bir telefon hattı ve ek bütçeye ihtiyaçları vardır. Bir yazıcı da her zaman ihtiyaç duyulan bir cihazdır.

9) Öğrenim çevresi: Eğitimcilerin öğrencilerin dersi görüyor olduğu çevreyi kontrol etme şansları yoktur. Örneğin öğrencilerin dersi gördüğü ortamlarda gürültü, eğitimci tarafından kontrol edilemez (BTE içinde aynı durum geçerlidir).

10) Web'de kaybolma: Kullanıcılar Internet ortamında dolaşırken kötü dizayn edilmiş sayfalarda aradıklarını bulma zorluğu çekebilirler. Eğer ki iyi dizayn edilmiş ve kullanıcıya yol gösteren bir site sistemi oluşturulursa, bu sorun engellenebilir. Ancak kullanıcı nerede kaldığını veya nerede durması gerektiğini bilemeyebilir yada unutabilir. Sadece birkaç web tabanlı sistemde bu tarz bir sistem vardır. Diğer taraftan BTE sistemlerinde, kullanıcıya sıralı ve düzenli gidilmesi gereken yerlerde son kaldığı yeri hatırlatma normal bir özelliktir.

11) Elektronik iletişimin güvenilirliği: Elektronik iletişim gerek günlük hayatta gerek iş alanlarında pek çok doyurucu özelliğe sahiptir. Elektronik iletişim sonucunda ortaya çıkan medya faktörünün etkinliği ve sosyal mevcudiyet gibi pek çok konu, yüz yüze iletişim eksikliği tabanında artı ve eksileriyle incelenmiştir. Eğitimciler ve öğrenciler elektronik ekipmanların zayıflığından etkilenmektedirler. Elektronik posta da kullanılan bir diğer elektronik iletişim aracıdır. Fakat kişisel postalarla ders dokümanı niteliğindeki postaların

karışması, posta hesabının aşırı doluluğu gibi sebeplerle bazen acil cevap bekleyen postalar için kullanışlı bir araç değildir.

Eğitmenler açısından WTE'nin dezavantajları şunlardır:

- 1) Bilişim sistemi sağlayıcısının güvenilirliği: Öğrencilerin karşılaştığı sorunlarla aynıdır.
- 2) Yüksek koordinasyon: WTE'de pek çok ek parça ve iletişim sorunu olduğundan dolayı, eğitmenler BTE ve klasik sınıf sistemine oranla daha koordineli bir sistemde çalışmak durumundadırlar.
- 3) Dijital malzeme gereksinimi: WTE'de kullanılacak malzeme dijital formatta olmalıdır. Diagram, grafik, resim ve benzeri malzemelerde önceden taranmış ve dijital formata geçirilmiş olmalıdırlar.
- 4) Virüsler: Eğitmenlerinde veri transferi esnasında virüslerle sorun yaşamaları olasıdır.
- 5) HTML bilgisi gereksinimi: Kendi malzemesini oluşturacak olan eğitmenlerin bazı sorunları giderme ve kendi malzemelerini rahatça oluşturmaları için en azından belli bir miktar HTML bilgisine sahip olmaları gerekmektedir.
- 6) Teknik desteğe bağımlılık: Eğitmenler yaşadıkları sorunların giderilmesinde ve derslerin formatını geliştirme safhasında teknik elemanlara ve programcılara bağımlı olma riskini taşırlar.
- 7) Lisans ve kullanım hakları: Klasik sınıf sisteminde eğitmenler konuyla alakalı herhangi bir kitaptan veya dergiden alacakları yazı, diagram, grafik gibi malzemelerle dersi işleyebilirler. Ancak bu malzemeler dijital formata geçirildiğinde eğitmenler kopya haklarını (copyright) çiğnememek için fazladan gayret sarf etmek zorundadırlar.
- 8) Elektronik iletişimin güvenilirliği: Eğitmenler de öğrenciler gibi elektronik iletişimde sorunlarla yüzleşebilirler. Bunlar medyanın etkinliği, posta hesaplarının doluluğu ve hemen cevaplanması gereken postaların cevaplanmasında karşılaşılan sorunlardır.
- 9) Güvenilirlik: Sanal bir ders ortamında öğrenciler ödevlerini elektronik olarak verirler. Ancak verilen ödevin gerçekten öğrenci tarafından mı yapılıp yapılmadığı bir soru işaretidir. Tabii olarak bu sorunla sadece değerlendirme safhasında karşılaşılacaktır. Bununla beraber pek çok WTE derslerinde de değerlendirme söz konusu değildir. WTE'de eğitmen öğrencinin sınavı yada ödevi alıp almadığını kontrol edebilir ama verilen görevin gerçekten öğrenci tarafından yapılıp yapılmadığını kontrol edemez. Güvenilirlik için retina ve parmak izi testi

gibi pek çok sistem kullanılmaktadır ancak uzaktan eğitim ortamında fiyat bazında oldukça pahalıdır.

Kurum açısından ise WTE'nin dezavantajları şunlardır:

- 1) Internet servisi sağlayıcısının (ISP) güvenilirliği: Kullanıcılar ve eğitmenler ISP üzerinden on-line bir çalışma yaptıklarında ISP'nin sağlıklı çalışması gerekmektedir.
- 2) Yüksek hızda ağ bağlantısı gerekliliği: Kuruluşlarla ISP'ler arasında hızlı bir bağlantı gerekmektedir. Ayrıca kullanıcıların ulaşımı için yüksek hızda modem ağları gerekmektedir. Bu tarz işlemler hem kurulumda hem kullanımda oldukça pahalı olabilirler.
- 3) Eğitmenin eğitimi: Dersleri verecek olan eğitmenlerin kullandıkları sistemler ve derslerin oluşturulması noktasında yeterince iyi bir eğitimden geçmeleri gerekmektedir, ayrıca gerekli durumlar için teknik personel bulunması zorunluluktur.
- 4) Standart eksikliği: WTE malzemelerinin oluşturulması ve geliştirilmesi için pek çok araç gerekmektedir. Fakat bu araç için herhangi bir standart koyulmamıştır. Eğer kurumlar Web tabanlı sistemlerini sadece tek bir yolla oluştururlarsa piyasada yetersiz kalabilirler.
- 5) Destek altyapısı: WTE sistemi oluşturulacak kurumların oldukça sağlam destek ve yönetim organizasyonuna ihtiyaçları vardır. Bunun içinde gerekli yedeklemelerin yapılması, herhangi bir sorunda kullanıcılara ve eğitmenlere destek olabilecek bir yardım bölümünün olması gerekmektedir. Örneğin, gece ertesi gündüz teslim edeceği projeyle uğraşan bir öğrencinin bağlantıda ya da ağda bir sorunla karşılaştığında arayabileceği bir destek biriminin olması gerekmektedir.
- 6) Kurulumunda ve uygulamada maliyet yüksekliği: Kurulum, uygulama ve destek için oldukça yüksek insan ve teknolojiye ihtiyaç vardır.

WTE için özetle aşağıdakileri söyleyebiliriz:

Akademik ve ticari kuruluşlarda Web tabanlı eğitim giderek önem kazanmaktadır. Öğrenci merkezli eğitimin oldukça önemli bir uygulama alanıdır. Bu sistem hem eğitmen-öğrenci arasında etkileşimli bir ortam yaratırken diğer taraftan öğrencinin tek başına öğrenme ve yeteneklerini keşfetme olanağı sağlamaktadır. Bu yüzden bu sistemi motive olmuş bireyler ve klasik sınıf sisteminde uygulanması gerekmeyen dersler için kullanmak daha mantıklıdır. Bu sistem ayrıca öğrenci ve eğitmen arasında iletişim gereken derslerde daha avantajlıdır. Dezavantaj olarak bu sistemde, bilginin aktarılması ve hazırlanmasında harcanan zaman

aslında ders işleyerek de telafi edilebilir olmasıdır.

Aşağıdaki Çizelge 2.2’de WTE dersleri için gereklilikler ve amaç özet halde gösterilmektedir.

Çizelge 2.2 WTE dersleri için gereklilikler ve amaçlar

Gereklilikler			
Donanım	Yazılım	Diğer	Amaç
1) Yüksek kapasiteli sunucular 2) Mutimedya özellikli bilgisayarlar 3) Multimedya gereçleri 4) Ağ bağlantısı	1) Sunucu için yeterli ve güvenilir yazılımlar 2) Web geliştirme araçları 3) Tarayıcılar 4) Ağ paketleri	1) Ders hazırlayıcısı ya da geliştirme personeli	1) Tamamıyla değişim, birleşik iletişim veya teknoloji girişi 2) Motive olmuş öğrenciler 3) Doğru dağıtılmış eğitim alanı 4) İletişim kullanan eğitim sistemi

2.2.4 Telekonferans Sistemi

Bu sistem, ders esnasında tüm kullanıcıların ve eğitmenin birbirleriyle bağlantılı olmalarıdır. Bu sistem üç değişik konferans sistemini kapsar; ses, görüntü ve veri. Eski sesli konferans sistemlerinde sadece telefon üzerinden ses transferi mümkündü. Bu sistem uzaktan eğitim için çok iyi bir araç olmamakla birlikte sadece uzaktaki öğrencinin eğitmene gerekli soruları sorabilmesi için kullanılabilir. Görüntülü konferans için bilgisayarda gerekli olan görüntü taşıyıcı yazılım, mikrofon gibi cihazlar olması gereklidir. Veri konferans sistemi tüm kullanıcılar arasında eş zamanlı veri transferi imkanı taşıyan bir sistemdir. Bu sistemin içinde dokümanların paylaşımı, otomatik olarak doküman ve çalışmaların yollanması, “chat” ve ilan panosu kullanımı vardır. Eğer görsel grafiklerle birlikte yazı transferi söz konusuysa, bu sisteme ses grafik konferans sistemi denir. Son olarak video konferans sistemi için kamera, grafik uyumlu monitör ve kullanıcının ulaşımını sağlayacak gerekli yazılım gerekmektedir ki sanal da olsa yüz yüze bir ders ortamı yaratılabilsin.

Video konferans sistemi iki biçimde oluşturulabilir:

- 1) Geniş olanakların kullanımı ile
- 2) Masaüstü video konferansı

Geniş olanaklar kullanılarak hazırlanmış video konferans sistemi genellikle büyük kuruluşlarda, daha önceden hazırlanmış bir web tabanı ve bağlantı hattının kullanılmasıyla uygulanmaktadır. Masaüstü video konferans sistemi genellikle daha ucuz yazılımlarla internet

ve intranet sistemleri kullanılarak kişisel bilgisayarlarla yapılmaktadır. Örneğin, IBM'in daha önceden masaüstü video konferansları için hazırlanmış olduğu yazılımı kullanılmaktadır (Whalen ve Wright, 1998).

Video konferans sistemi kullanılan uzaktan eğitim sistemlerinde etkileşim üç değişik biçimde incelenmelidir:

- 1) Tek yönlü etkileşim: Sadece eğitmen tarafından öğrencilere video, ses ve veri transferinin yapılması (öğrencinin bir etkisi söz konusu değildir).
- 2) Çift yönlü etkileşim: Eğitmenle öğrenci arasında etkileşimin olduğu platformdur. Karşılıklı görüntü ses ve veri transferi söz konusudur.
- 3) Çok yönlü etkileşim: Eğitmenle öğrencilerin birbirleriyle etkileşim halinde bulundukları platformdur.

Çift yönlü etkileşim sistem tek yönlüsüne oranla daha popülerdir çünkü eğitmen öğrenciden direkt tepki alabilmektedir. Burada genellikle ses transfer olanakları kullanılmaktadır.

2.2.4.1 Gereksinimler ve Tasarım

Tüm telekonferans sistemlerinde sadece bir adet önemli gereklilik vardır; yeterli bant genişliği. Diğer gereklilikler kullanılacak telekonferans sistemine bağlıdır.

Geniş Kapsamlı Telekonferans Sistemleri:

Bu sistemi kullanacak kurumların her yerden ulaşımı kolay olan, ders için özel tasarlanmış salonları olmalıdır. Bu odaların kamera ile donatılmış olmaları gerekmektedir. Ayrıca büyük ekran ve bazı ses transferi için araçlar da odada bulunmalıdır. Gerekli destek personeli de ortam içinde bulunmalıdır. Bu bağlamda oldukça pahalıdır. Ayrıca çok ders verilmediği takdirde bu sistemi kullanmak pahalıya gelmektedir.

Ayrıca bant genişliği ve hatlarını oluşturulması için de oldukça yüksek oranda yatırım yapmak gerekmektedir. Bağlantıyı yükseltmek ve kalitesini arttırmak için ya uydu bağlantısı ya da T1* ve T3** gibi özel bağlantı sistemleri kullanılmaktadır.

* AT&T tarafından metin, bilgi ve görüntü transferlerinin yüksek bant genişliğinde, bir saniyede 1.544 Mbps ileten 24 ses kanallı iletişim hattı. Genellikle Internet gibi geniş iletişimlerde kullanılır.

** Bir saniyede 44.736 Mbps iletişim hızlı ve 672 ses kanallı iletişim hattı.

Geniş kapsamlı telekonferans sistemlerinde dersi desteklemek için eğitmen sesi ve görüntüsü öğrenciye ulaştırılmalıdır.

Masaüstü Telekonferans Sistemi:

Masaüstü telekonferans sistemi geniş kapsamlı telekonferans sistemine oranla hem de maliyet açısından oldukça avantajlıdır. Buna rağmen ağı pek çok kullanıcı bağlansa bile toplam yatırım oldukça düşüktür. Kurumlar çalışanlarına işlerini halletmeleri için kişisel bilgisayar sağlamaktadır. Kurumlar işte kurdukları bu sistemin üzerine ekleyecekleri bazı araçlarla uzaktan eğitim olanağı sağlayabilirler. Kullanıcıların bu durumda yeni bilgisayar sistemleri ve multimedya olanaklarıyla (ses kartı, görüntü kartı, monitor, vs.) ve ağı bağlantıyla bu sistemi kullanılabilir.

Diğer kullanıcıların kameraya ve mikrofona ihtiyaçları vardır. Bu gereçler sadece devamlı gönderilen ve alınan ses ve görüntü sinyalleri için gereklidir. Bunlar olmadan da bir öğrenci eğer düzgün çalışan bir ses kartı ve hoparlöre sahipse derslere yine katılabilir. Son olarak gerekli telekonferans yazılımlarına gerek vardır, örneğin Microsoft NetMeeting ya da CU SeeMe (Cornel üniversitesinde üretilen ve ticari amaç altında kullanılan yazılım) sadece ses transferinde kullanılan bazı yazılımlar (örneğin Netscape CoolTalk kullanılmalıdır).

Eğer kurumlar eşit olarak paylaştırılmış bir uzaktan eğitim sistemi kullanılacaksa ağı seçenekleri genelde internet üzerinden olmaktadır. Eğer kullanıcılar nispeten orta hızlı modemler kullanılıyorsa grup çalışmaları ve doküman paylaşımlarında sabır ve zaman gerekmektedir.

Telekonferans ile yapılan eğitim sistemi kullanılarak yapılan eğitime oranla henüz çok yetersiz bir teknolojidir fakat coğrafyadan bağımsızlık ve maliyet noktasında pek çok avantajlara da sahiptir.

2.2.4.2 Telekonferans Sisteminin Avantajları

Öğrenci açısından Telekonferans Sisteminin avantajları şunlardır:

1) Yüz yüze etkileşim: Görsel video konferans sistemlerinde bu özellik bulunmaktadır. Bu her iki tür konferans sisteminde (geniş kapsamlı ya da masaüstü konferans sistemlerinde) öğrenciye hem eğitmeni hem de öğrencileri eş zamanlı görme olanağı sağlar.

2) Öğrenci-eğitmen etkileşimi: Öğrenciler eğitmenlerle eş zamanlı olarak bağlantıda oldukları

için herhangi bir soruyu hemen sorup anında cevap alabilirler.

3) Coğrafyadan bağımsızlık: Masaüstü telekonferans sistemlerinde öğrenciler derslerine Internet'e ulaşma olanakları olan dünyanın herhangi bir noktasından bağlanabilir. Hatta derslerine evlerinden de katılabilirler. Geniş kapsamlı telekonferans sistemlerinde öğrencinin belli yerlere ulaşması gerekebilir. Her şeye rağmen telekonferans sistemleri derslere katılma olanağı olmayan öğrencilere geniş olanaklar tanımıştır.

4) Karmaşık problemlerin açıklanması: Telekonferans sistemlerinde ses, görüntü, veri ve görsel grafiklerin öğrenciye ulaştırılması şansı olduğu için karmaşık problemler daha rahat çözülmektedir.

Eğitmen açısından Telekonferans Sisteminin avantajları şunlardır:

1) Yüz yüze etkileşim: Sadece öğrenciler değil, eğitmen de yüz yüze etkileşimle konuların daha iyi anlaşılıp anlaşılmadığını test edebilir, uzaktan bağlansa bile öğrenciye sosyal bir ortamda bulunduğu izlenimini verebilir.

2) Karmaşık problemlerin açıklanması: Görsel ve işitsel araçlarla eğitmen konuları öğrencilere daha rahat anlatılabilir ve karmaşık problemleri rahatça açıklayabilir.

3) Yenileme kolaylığı: Eğitmen aynı anda dersi anlatabileceği için gerekirse malzemeleri değiştirebilir ve yeniden düzenleyebilir.

Kurum açısından ise Telekonferans Sisteminin avantajları şunlardır:

1) Operasyon maliyeti: Eğer kurumun çalışanları kişisel bilgisayarlarında masaüstü telekonferans sistemlerine sahiplerse maliyet oldukça düşük olmaktadır. Sadece telekonferans araçları ve ağ bağlantısı yeterli olmaktadır.

2) Coğrafyadan bağımsızlık: Kurumlar bu sayede daha geniş alanlara ulaşabilir, pazar paylarını geliştirebilirler. Teorik olarak herhangi bir kullanıcı dünyanın herhangi bir noktasından derse katılabilir.

2.2.4.3 Telekonferans Sisteminin Dezavantajları

Öğrenci açısından Telekonferans Sisteminin dezavantajları şunlardır:

1) Bant genişliği zorunluluğu: Masaüstü telekonferans sistemi kullanıcıları için bant genişliği oldukça önemlidir. Çoğu zaman 56 Kbps modemler bile yetersiz kalabilmektedir.

2) İşitsel engeller: Pek çok masaüstü telekonferans sistemlerinde sadece iki kişinin aynı anda ses transferini kullanmasına olanak sağlamaktadır. Bu her kullanıcı ses iletişimi olarak aynı anda sadece bir başka kullanıcıya bağlanabilmektedir.

3) Görsel engeller: İşitsel engellerde olduğu gibi telekonferans paketlerinde sadece iki kullanıcı kendi aralarında görüntü transferi yapabilirler.

4) Bağlantı olmadan çalışamama: Telekonferans sistemleri eş zamanlı olarak çalıştığı için kullanıcı tüm seansa katılmak zorundadır. Yani, bağlantı olmadan dersi tekrar izleyemez.

Eğitmen açısından Telekonferans Sisteminin dezavantajları şunlardır:

1) Bant genişliği gerekliliği: Öğrencilerde olduğu gibi eğitmenler de derslere yüksek hızda bir sistemde bağlanmaları gerekmektedir.

2) İşitsel ve görüntüsel engeller: Masaüstü telekonferans sistemlerinde öğrencilerde olduğu gibi, eğitmenlerde de görsel ve işitsel veri transferinde çeşitli engeller ortaya çıkmaktadır. Geniş kapsamlı telekonferans sistemlerinde böyle bir sorunla karşılaşılmamaktadır.

Kurum açısından ise Telekonferans Sisteminin dezavantajları şunlardır:

1) Uygulama: Geniş kapsamlı telekonferans uygulamalarında gerekli olanaklar sağlandığı takdirde maliyet yüksek çıkmaktadır. Bu maliyeti düşürmek için olanaklara sahip odada pek çok ders verilmesi gerekmektedir.

2) Güvenlik ve gizlilik: Masaüstü telekonferans uygulamalarında pek çok kurum kendi ağ omurgasında kullanmaktadır. Bu durumda gizlilik ve güvenlik sorunlarıyla karşılaşmak olasıdır. Bu durumda firmanın kendi intraneti üzerinden uygulama yapması kurumun yararınadır.

Telekonferans sistemi için özetle aşağıdakileri söyleyebiliriz:

Geniş kapsamlı telekonferans uygulamalarında gerekli araçların sayısı artmakta ve sistem karmaşıklaşmaktadır. Küçük gruplar için bu sistem daha uygundur. Bu sistemin maliyeti de kurumların verdiği ders sayısını da kısıtlamaktadır. Eğer gerçekten görsel veri transferi bir gereklilikse bunun için farklı yollar mevcuttur. Diğer taraftan masaüstü telekonferans sistemleri pek çok yönden daha avantajlı ve ulaştığı kitle daha geniştir. ayrıca veri transferinde (görsel, işitsel ve metin) pek çok artıları vardır.

Aşağıdaki Çizelge 2.3'te telekonferans sistemi için gereklilikler ve amaç özet halde gösterilmektedir.

Çizelge 2.3 Telekonferans sistemi için gereklilikler ve amaçlar

Gereklilikler			
Donanım	Yazılım	Diğer	Amaç
Geniş Kapsamlı Sistemler			
*Yeterli bant genişliği * Kamera *Büyük ekranlar * Ses donanımı	* Ses ve diğer sistemlerin transferini sağlayacak yazılım	* Eğitimli destek personeli * Özel odalar	* Tamamıyla değişmiş, birleşik iletişim veya teknoloji girişi * Küçük gruplar * Az sayıda site * Sık kullanım
Masaüstü sistemleri			
* Multimedyalı bilgisayarlar * Ağ bağlantısı * Kamera ve mikrofon	* Multimedya yazılımı * Konferans için yazılım		* Tamamıyla değişim, birleşik iletişim veya teknoloji girişi * Eşit olarak dağıtılmış eğitim hakkı

2.2.5 Videoteyp Sistemleri

Uzaktan eğitim sistemlerinde her ne kadar eski bir teknoloji olsa bile video teypler hali hazırda sıkça kullanılan iyi bir alternatiftir.

Taşıması ve ulaştırılması hayli kolay olan bu yöntemde gerekli olan bir video, ekran ve uygun bir odadır. Hatta öğrencinin evinde gerekli araçlar var ise, öğrenci dersi evde kendi kendine yapabilir. Ancak malzeme durağandır ve yeni geliştirilmiş içerikli bir kaset üretilmedikçe konular güncellenemez. Bu sistemde öğrenci pasif konumdadır. Bu sistem kitap ve benzeri malzemelerle desteklenmelidir. Örneğin öğrencinin kendi kendine çözebileceği alıştırma kitapları gibi. Bu sistem bilgisayar ortamında da kullanılabilir. Eğer öğrencinin kişisel bilgisayarına varsa kurumlar kaset içeriğindeki çeşitli görsel malzemeleri CD ve disket gibi araçlarla öğrenciye ulaştırılabilir. Tabiki bunun için, öğrencinin gerekli multimedya araçlarına sahip olması gerekmektedir.

2.2.5.1 Gereksinimler ve Tasarım

Durağan yapıya sahip olan bu sistemde konuların güncellenmesi ve değiştirilmesi maliyet açısından oldukça zahmetlidir. Bu sebeple hazırlanan derslerin öncelikle konuyu kısaca genel

olarak tanıtılması ve kullanılacak malzemeyi tanıtması gerekir ki öğrencinin zihninde konu hakkında bir fikir oluşabilsin. Gereken yerlerde özellikle bireysel alıştırmalarda, öğrenciye uygun yeterli teneffüs ve aralıkların da koyulması gerekir. Bu sistemin daha iyi ve geniş uygulama alanlarından biride derslerin CD-ROM veya disketlerle kullanıcıya ulaştırılmasıdır.

2.2.5.2 Videoteyp Sistemlerinin Avantajları

Öğrenci açısından Videoteyp Sistemlerinin avantajları şunlardır:

- 1) Zamandan bağımsızlık: Öğrenci kendisi için en uygun zamanı ayarlayıp dersi o zaman izleyebilir.
- 2) Coğrafyadan bağımsızlık: Kasetler dünyanın her yerine ulaştırılabildiği için öğrenci dersi istediği yerde alabilir.
- 3) Kısıtlı zorunluluklar: Uygun standartlardaki her kaset ve video için öğrencinin ihtiyacı olan gereçler oldukça az ve bilgisayar gerekliliği ile kıyaslandığında oldukça ucuzdur. Uzaktan eğitimde kullanılan diğer sistemlerle kıyaslandığında öğrenci açısından en ucuz sistem budur.

Kurum açısından ise Videoteyp Sistemlerinin avantajları şunlardır:

- 1) Zamandan bağımsızlık: Bu sistemde her zaman hazır bulunması gereken sabit bir öğretmene ihtiyaç yoktur. Dersler bir grup öğretmen tarafından hazırlanıp dünyanın her yerine gönderilebilir.
- 2) Zamanlama esnekliği: Pek çok öğrenciye hitap eden derslerden BTE'nin aksine ders zamanlarını belirlenmesi öğrenciye bağlı olduğu için rahattır.
- 3) Maliyet: Geniş kitlelere hitap eden eğitim kurumlarında hem eğitim ücretinin çabuk ödenmesi hem de öğretmen ve ulaşım masrafının azlığından dolayı maliyet kurumlar açısından oldukça düşüktür.

2.2.5.3 Videoteyp Sistemlerinin Dezavantajları

Öğrenci açısından Videoteyp Sistemlerinin dezavantajları şunlardır:

- 1) Tepki: Eğer ki öğrencinin anlamasının ve bilgisini pekiştirecek ve değerlendirecek bir ek kaynak yok ise öğrenci anlamadığı yerleri hemen tamamlayamaz.
- 2) Tek yönlü öğrenim: Öğrenciler pasif konumdadırlar, sadece görsel ve işitsel yönden tatmin olurlar.

3) Öğrenciler arasında iletişim eksikliği: Öğrenciler arasında herhangi bir iletişim ve etkileşim yoktur.

4) Eğitimci eksikliği: Zorluklarla ve karmaşık problemlerle karşılaşan öğrencilerin birinci elden soru sorabilecekleri bir eğitimci yoktur.

Kurum açısından ise Videoteyp Sistemlerinin dezavantajları şunlardır:

1) Yenileme zorluğu: Kasetlerin güncellenmesi oldukça zahmetlidir, hatta maliyet açısından neredeyse yeni bir kaset üretmeye yakındır.

2) Ölçme ve değerlendirmede eksiklik: Kurum düzenli bir şekilde öğrencilerin bilgilerini kontrol etmezse bilgiler kalıcı olmayabilir.

3) Öğrenme ortamı: Öğrencinin dersi işlediği ortam yeterli ve öğrenim açısından uygun olmayabilir. Kasetler bir şekilde zarar görebilir.

Kasetler halen uzaktan eğitim için oldukça kullanışlı bir araçtır. Özellikle durağan ve geniş kitleye ulaştırılması gereken konularda daha kullanışlı ve maliyeti düşüktür. Yapılan bir araştırmada çalışanlar özellikle mesleki eğitimlerinde bu yöntemle daha sıcak baktıklarını belirtmişlerdir (Dobson, 1995).

Aşağıdaki Çizelge 2.4'te videoteyp için gereklilikler ve amaçlar gösterilmektedir.

Çizelge 2.4 Videoteyp için gereklilikler ve amaçlar

Gereklilikler			
Donanım	Yazılım	Diğer	Amaç
* Video ya da bilgisayar		* Gürültüsüz ders ortamı	* Tamamıyla değişim, birleşik iletişim * Kendi kendine motive olmuş bireyler * Eşzamansız öğrenme * Geniş öğrenci potansiyeli * Durağan malzeme

2.2.6 Video Tele Eğitim

Video Tele Eğitim (VTE), uzaktan toplantı yapmanın ve videoteype çekmenin bazı kolaylıklarını sunmaktadır. Bu teknoloji, farklı yerlerde (konumlarda) uzaktan öğretim veya eğitim sunmak için canlı video ve ses yeteneklerini kullanmayı içermektedir. Eğitimci, uzaktaki bir veya daha fazla yere yayınlanan gerçek zamanlı şekilde ders vermektedir. Öğrenciler, telefon hattını veya diğer iki yönlü işitsel haberleşme olanaklarını kullanarak sık

sık derslere katılabilmektedir. Uzaktaki konumlarda özel öğretmenler veya asistanlar, sorulan sorulara cevap vermek için beklemektedir. Uzaktan eğitim veren akademik kurumlar da dahil birkaç büyük organizasyon, VTE'yi kapsamlı olarak kullanmaktadır.

2.2.6.1 Gereksinimler ve Tasarım

VTE uygulanması, ana eğitim merkezinde özel odaların olmasını ve uzaktaki bütün konumların bu özel dersten haberdar olmasını gerektirir. İşitsel olarak ses, video ve diğer verilerin gönderilmesi için, bu siteleri birbirine bağlamak için büyük bant genişliğine ihtiyaç vardır. Genellikle kurumlar, ya kendi uydu bağlantılarını veya yüksek hızlı fiber hatlarını kullanmakta ya da her bir dönem için bu hatları kiralamaktadır. Yerel olarak eğitmen, bir veya daha fazla kameraya ve en az iki monitöre gereksinim duymaktadır. Monitörlerden bir tanesi giden videoyu görmek için, diğer monitör ise gelen videoyu görmek için kullanılmaktadır. Bu, eğitmenin uzaktaki konumlardan bir anda sadece bir tanesini gördüğü anlamına gelmektedir. Buna ilaveten, bütün konumlardaki öğrenciler tarafından görülmekte ve duyulmaktayken metinleri ve grafikleri göstermek için bir takım bütünleşik aletleri kullanmaya gereksinim duymaktadır. Kamera kaydediciler, bilgisayar çıkışları, slaytlar veya diğer gösteri aletleri de eğitmen için mevcut olan bütünleşik aletler takımına ilave edilebilir.

Bu teknolojinin kullanımı, teknik personelin VTE aracılığıyla sunulmakta olan her bir kurs için eğitmenlere yardım etmek üzere mevcut olması gerektiği anlamına gelmektedir. Uzaktaki konumlarda, (ve eğer bazı öğrenciler varsa yerel konumlarda da) ana sinyali ve öğrenciler mikrofonlarını kullanarak kontrolü ele aldıkları zaman diğer uzaktaki konumlardan gelen sinyalleri göstermek için monitörlerin kurulmuş olması gerekmektedir. Eğer on kişiden fazla öğrenci varsa, kurumlar daha fazla monitör kurmak gerçeğini dikkate almalıdır. Sonuç olarak, kursların kendileri bazı kontrollere sahip olunurken öğrencilerle eğitmenin arasında bazı etkileşimlerin olmasına zorlamak amacıyla tasarlanmaktadır. Bu, iletişim karışıklığını engellerken çekingen öğrencileri derse katılmak için cesaretlendirecektir.

Bir VTE uzaktan oturumu kurmak için belirli minimum gereksinimler şunları kapsamaktadır:

- Her bir mevkide iki veya daha fazla televizyon monitörü (veya bilgisayar monitörü).
- Her bir mevkide bir veya daha fazla kamera.
- Tartışmaya iştirak edecek ve soru soracaklar ise bütün katılımcılar için mikrofon.
- Bütün mevkilere sinyallerin iletimi için ağ bağlantıları.

- Bir CODEC (kompresör/dekompresör). Bu teçhizat, iş/zaman oranını iyileştirmek için sinyallerin sıkıştırılmasını ve açılmasını sağlar.
- Bütün uzaktaki mevkilerde bir kolaylaştırıcı.
- Bütün uzaktaki mevkilerde ve yerel mevkilerde teknik destek personeli.

2.2.6.2 Video Tele Eğitimin Avantajları

Öğrenciler açısından Video Tele Eğitiminin avantajları şunlardır:

1) Gerçek zamanlı: VTE’de iki yönlü haberleşme bağlantıları mevcut olduğu için, ders sırasında ele alınan malzemeler gerçek zamanlıdır. Anlama sorunu çeken öğrenciler, sağlıklı bir biçimde yardım alabilmektedir. Eğitimci, açıklamaları desteklemek için görsel girdiler kullanılabilmektedir.

2) Güncel kalan dersler: VTE dersleri gerçek zamanlı verildiği için eğitimci, öğrencilerine en güncel bilgileri sunmak için malzemeleri gerektiğinde değiştirilebilmektedir. Diğer bir seçenek ise eğitimcilerin anlama sorununun mevcut olduğunu gösteren bir geribildirim olduğu zaman bilgileri tekrarlayabilmesidir.

3) Öğrenciler arasında etkileşim: Öğrenciler çeşitli konular arasında sözlü olarak birbirlerini etkileyebilmektedir. Bu etkileşim, öğrencilerin her birinin sorduğu sorulardan ve cevaplardan öğrenebilmelerine yardımcı olmaktadır.

4) Eğitimci ile etkileşim: Öğrenciler eğitimcileriyle sözlü olarak ve bazen uzaktaki konularda da kameralar kuruluysa gerçeği gibi yüz yüze etkileşim içine girebilirler. Bu, öğrenciler için dersi daha kişisel yapmaktadır.

Eğitimci açısından Video Tele Eğitiminin avantajları şunlardır:

1) Dinamik güncellemeler: VTE’deki dersler canlı olarak sunulduğu için eğitimci içeriği öğrencilerden gelen geribildirimle göre veya ortamda meydana gelen ilişkili önemli olaylara göre değiştirebilirler.

2) Öğrenciler arasında etkileşim: Eğitimci, ders içeriği ve sunumu üzerine geribildirim elde etmek için öğrenciler ile etkileşim içine girebilirler. Ayrıca, öğrencilerle daha kişisel görüşmeler yapabilirler.

Kurumlar açısından ise Video Tele Eğitiminin avantajları şunlardır:

1) Coğrafi ulaşım: Potansiyel öğrenci pazarını uzak konumlara genişletmek isteyen kurumlar VTE'nin bu olanağı sunduğunu görebilirler. Bir ders bir konumda bir defa verilebilmektedir ve VTE yolu ile dağıtıldığı için farklı uzak konumlardaki farklı öğrenciler aynı ders için kayıt olabilmekte ve bu dersi aynı anda alabilmektedir.

2) İşletme masrafları: Farklı büyük öğrenci grupları farklı uzak konumlara dağılmış olduğu zaman, VTE maliyeti düşüren bir dağıtım mekanizması olabilmektedir. Her bir uzak konumda moderatörler, yardımcılar veya asistanlar çalıştırılmalıdır. Fakat bu kişilere, kalifiye bir eğitmenen daha az maaş verilmektedir. Buna ilaveten, eğitmenin birçok uzak konuma gitmesi gerekmemekte, bir organizasyon için önemli giderler olan seyahat masrafları azaltılabilmektedir.

2.2.6.3 Video Tele Eğitiminin Dezavantajları

Öğrenciler açısından Video Tele Eğitiminin dezavantajları şunlardır:

1) Plan yapmak: Eğitim teknolojilerinin büyük bir kısmı, öğrenciler için kendi kendine çalışma seçeneklerinin bazı formlarını sunmaktadır. Bununla birlikte VTE'de, bir ders verildiği zaman öğrenciler, geleneksel sınıf ortamındaki gibi tam anlamıyla mevcut olmalıdır.

2) Coğrafi bağımlılık: VTE bazı uzaktan eğitim imkanları sunduğu zaman, bu dersleri sunmak için gerekli olan altyapı sistemi sadece sınırlı sayıda uzak konumların mevcut olabileceği anlamına gelmektedir. Buna rağmen öğrenciler, bu konumlara gitmelidirler. Bu gezi, eğer uzaktan öğretim olanağı yoksa gerekli olandan çok daha az olabilir. Fakat bu, yine de öğrenciler için kısıtlayıcı olmaktadır.

3) Görsel ekran sınırlamaları: VTE'de kullanılmakta olan monitörler genellikle sıradan büyük ekran televizyon monitörleridir. Bu monitörler, bir sınıfta birkaç öğrenci olduğu zaman çok küçük olabilmektedir. Bu durum, monitörlerden belirli bir uzaklıkta oturması gereken bu öğrenciler tarafından görülenlerin ve duyulanların kalitesini etkiler.

Eğitmenler açısından Video Tele Eğitiminin dezavantajları şunlardır:

1) Plan yapmak: VTE eş zamanlı bir dağıtım mekanizması olduğu için dersler VTE'de planlanmalıdır. Eğitmenler bu dersleri planlarken, potansiyel zaman kuşağı farklılıklarını hesaba katmalıdır. (eğer dersler dünya çapında sunulmaktaysa). Ayrıca belirli sınıflarda kurulum için yeterli ön ve ileri ders zamanını sağlamalıdır.

2) Artırılmış koordinasyon: VTE'nin gerçekleştirilmesi, yerel ve uzaktaki öğrencilerin derse

katılımlarını kontrol altına alacak olan eğitimde büyük koordinasyon gücü olmasını gerektirir.

3) Güç katılım: Diğer öğrenciler daha az çekingenken bazı öğrenciler kameradan çekinir. Eğitimci bu iki grubu çok dikkatli bir biçimde dengelemelidir. Böylece sınıf tartışmalarında tamamen ölü sessizlik olmaz ve kontrolden çıkmış ve dağılmış tartışmalar engellenmiş olur.

4) Destek personeline bağımlı olmak: VTE ses ve video yapacak bazı personel gerektirdiği için eğitimci, dersinin düzgün dağıtımını garantiye almak için bu kişilerin varlığına bağımlı hale gelmektedir.

Kurumlar açısından ise Video Tele Eğitiminin dezavantajları şunlardır:

1) Eğitim gereklidir: Olanakların hızlı ve verimli kullanımı için, uzak konumların her biri moderatörlere veya asistanlara sahip olmalıdır. Bu kişiler, bu ortamda öğrenmeyi kolaylaştırmada hızlı ve verimli olmak için eğitilmiş olmalıdır.

2) Uygulama masrafları: VTE olanaklarını kullanarak uzaktan eğitimi gerçekleştirmek için asıl kurulum masrafları önemli olabilir. Özel odalar monitörlerle, kameralarla, mikrofonlarla ve haberleşme olanaklarıyla donatılmış olmalıdır. Eğer bu olanakları kullanacak birkaç ders varsa veya sadece küçük öğrenci grupları varsa, bu kurulum masraflarının geri alınması çok uzun zaman alabilir. Bu gibi küçük gruplar için masa üstü video konferans daha iyi bir seçim olabilir.

3) Üretim personeli: VTE yayın teknolojilerini kapsar. Bu yüzden, derslerin üretimi ve desteği için çok iyi eğitilmiş teknik personel gereklidir. Bu teknik personel, her dönem için destek vermelidir. Eğer organizasyon daha önceden bu tip personele ulaşamamışsa, masraflar bu beceriler için personel kiralamayla ve eğitimini veya taşeronları kapsamaktadır.

Video tele eğitim için özetle aşağıdakileri söyleyebiliriz:

Uzaktan öğretim için video tele eğitiminin tahsis edilmesi, büyük ölçekli telekonferanstaki ile benzerdir. Video kameralar ve ses becerileri kullanılarak karmaşık bir kurulum gerekmektedir ve altyapı sisteminin maliyeti büyük olabilmektedir. Bununla birlikte, neredeyse temel sınıf donanımlarıyla (monitörler ve mikrofonlar dahil olmak üzere) uzak konumlarda daha büyük gruplara ders verilebilmektedir. Bu, farklı fakat sınırlı sayıdaki uzak konumlara video tele eğitim kullanarak çok sayıda ders vermek isteyen kurumlar için iyi bir seçimdir. Kurumlar, bu uzun yarışta kazanmak için direnmekte ve uygun bir zaman aralığı içinde geri kazanıma ulaşabilmektedir.

Sonuç olarak eğitim teknolojileri için özetle aşağıdakileri söyleyebiliriz:

Bir uzaktan eğitim programı için ideal eğitim teknolojisi, organizasyonun belirli ihtiyaçlarına bağlıdır. Bununla birlikte, gerçekte ideal teknoloji sadece bir teknoloji olmayabilir. Örneğin, uzaktan eğitim alan öğrenciler bir internet sitesindeki bir malzemeye web tabanlı bağlantıları kapsayan, ders malzemesi içerikli bir CD-ROM alabilmektedir. Bu CD-ROM, ders için planların büyük bir kısmını ve referans malzemeleri içerebilmektedir. Web siteleri sık sık değişen herhangi bir dinamik bilgiyi içerebilir. Ayrıca aynı dersi alan diğer öğrenciler ile chat yapmak veya gerekli malzemelerle yardım etmek için müsait olan eğitmenlere soru sormak gibi iletişim olanakları da mevcuttur. Öğretim teknolojilerini birleştirmenin diğer bir örneği ise öğrencilerin web sayfasındaki ilgili bilgilere (derslerin ana hatları ve diyagramları) bakarken önceden kaydedilmiş malzemeleri duyabildiği audiografik tele-derslerdir. Bu karma dağıtım mekanizması uzaktan eğitim alan öğrencilere en iyiyi vermektedir.

Teknolojinin uygulanmasında dikkate alınması gereken diğer önemli yönler şunlardır:

- 1) Teçhizat: Teçhizat güvenilir olmalıdır. Böylece öğrenciler ve eğitmenler bir oturumun tam ortasında derslerini tamamlamayı engelleyecek bir teknoloji arızasıyla karşılaşmazlar. Bu nedenle desteğin verilmesi şarttır.
- 2) Destek altyapı sistemi: Organizasyonlar, uzaktan eğitim alan öğrenciler için teknoloji desteğini ve ders içeriği desteğini kapsayan destek altyapı sistemini dikkatli bir şekilde planlamalıdır.
- 3) Kullanım kolaylığı: Kurumlar, hangi teknolojinin kullanılacağına karar verilirken planladıkları belirli eğitim teknolojilerinin kullanımının kolay olmasına dikkat etmelidir.
- 4) Maliyet: Dikkate alınması gereken diğer bir faktör teknoloji uygulama ve anlatmanın kullanıcıya yansıyacak olan maliyetidir.
- 5) Etkileşimli ve eş zamanlı olma: Etkileşimli ve eşzamanlı olmanın düzeylerinin önemli noktaları, teknolojiye erişim ve malzemelerin eğitmen tarafından gözden geçirip düzeltilmesinin kolaylığı, belirli bir teknoloji seçilmeden önce düşünülmelidir.

3. LINUX İŞLETİM SİSTEMİ

Linux, uzun yıllardır çeşitli gelişim aşamalarından geçmiş Unix işletim sisteminin PC'lerde de kullanılabilen bir gerçekleştirmedir ve Unix kökenli olduğu içinde kararlı ve güvenilir bir işletim sistemidir.

1991 yılında ortaya çıkan Linux işletim sistemi diğer Unix türevlerine benzer şekilde çok kullanıcı (multiuser) ve çok görevli (multitasking) bir işletim sistemidir. Linux'un çekirdeği aradan geçen yıllar boyunca bir milyon satıra ulaştı. Binlerce çeşit ve tipte kartlar ve cihazlar desteklemeye başlandı. Linux işletim sistemi, eksiksiz bir ağ (network) desteği sağlamaktadır. TCP/IP, UUCP, PPP ve diğer modern ağ teknolojisinin pek çoğunu desteklemektedir. Linux'un genel kullanım amaçları:

- 1) İnternet haberleşmesi
- 2) Yazılım geliştirme
- 3) Kişisel kullanım
- 4) İnternet sunucusu şeklinde sıralanabilir.

Linux'un atası olan Unix'in tarihi ise 1960'lara kadar uzanmaktadır. Unix dünya üzerindeki ilk ciddi işletim sistemlerinden birisidir ve DOS işletim sistemi başta olmak üzere pek çok işletim sistemi Unix'ten esinlenerek geliştirilmiştir. Halen Unix'in iki ayrı sürümü standart olarak kabul edilmektedir. Bunlardan bir tanesi BSD adı verilen Berkeley Unix ve diğeri de System V Unix'tir. Unix, Bell laboratuvarlarında geliştirilmiştir. Aynı zamanda Unix, AT&T firmasının da tescilli markasıdır.

3.1 Linux Komut Yapısı

Unix ve benzeri işletim sistemlerinde kullanıcının komut yazmasını sağlayan , bu komutları yorumlayarak gerekli işlemleri yapan programlara kabuk (shell) adı verilir. Linux'ta bütün komutlar ve dosya isimlerinde büyük / küçük harf ayrımı önemlidir. Sistem komutlarının ve dosyaların çoğu küçük harf ile yazılır. Komut ve dosya adlarında kullanılan bazı karakterlerin gerek dosya ve izin yapısı, gerekse kabuk ve diğer komutlar nedeniyle bazı özel anlamları vardır. Örneğin, "/" karakteri hiçbir dosya adında bulunmaz (Dosya ve izinler için ayrıç olarak kullanıldığından). "-" ile başlayan bir dosya oluşturulabilir olsa da silmeye kalkıldığında yeni bir Linux kullanıcısına problem olabilir. Linux 'ta komutlara seçenek verirken seçenektan önce "-" karakteri kullanılır.

Linux komutları tersi istenmedikçe girişlerini standart girdiden (klavye) alır, çıktılarını standart çıktıya (ekran) yazar.

Linux kabukları komut satırından verilen komutu çalıştırmadan önce bir dizi karakteri yorumlayarak dosya adlarına çevirir. Bu karakterler;

* (Yıldız) karakteri: Sıfır dahil herhangi bir sayıda karakter yerine geçer.

Örneğin; `rm*` komutu bütün dosyaları siler, `ls-a*` komutu 'a' ile başlayan dosyaların listesini verir.

? (Soru işareti) karakteri: Tek bir karakter yerine geçer.

Örneğin; `??` adı iki karakterden oluşan bütün dosyalar anlamına gelir.

[] (Köşeli parantez) karakteri: Bu karakter arasında yazılan liste içindeki herhangi bir harfe dönüştürür.

Örneğin; `cp*[abc]/tmp` komutu 'a', 'b' ya da 'c' ile biten bütün dosyaları /tmp dizinine kopyalayacaktır.

Liste içinde aralarında "-" işareti koyarak aralıklar verilebiliriz.

Örneğin; `[A-Z]*` büyük harf ile başlayan bütün dosyalar anlamına gelir.

3.2 Dosya ve Dizin Yapısı

Linux altında bazı karakterlerin özel anlamları vardır. Dizin ve dosya isimlerinin başında nokta olması durumunda bu dosyalar gizli dosya haline gelir ve parametresiz yazılan `ls` komutuyla görüntülenemez. Dosya ve dizin isimleri 255 karakteri aşamazlar. Sisteme girilince önceden tanımlanmış bir dizin altında bulunulur.

Bu dizin normal kullanıcılar için genellikle /home ve ardından gelen kullanıcı dizini ismidir. Bulduğumuz dizinin ismini görmek için `pwd` (print working directory) komutu kullanılır. Bu komutun kullanımı;

```
$pwd
```

şeklindedir.

Linux işletim sisteminin komut yapısı DOS işletim sistemine benzemektedir. Linux'ta yeni bir dizin oluşturmak için kullanılabilecek olan komut `mkdir` komutudur. `mkdir` komutunun yazılış

yazılış biçimi;

cd <Dizin Belirleyicisi> şeklindedir. cd komutu ile dizin belirleyicisi arasında en az bir boşluk bulunmalıdır. Bu komutun bazı kullanımları aşağıdaki gibidir:

\$cd.. (Bu komut bir üst dizine çıkmak için kullanılır)

\$cd/ (Kök dizine çıkmak için kullanılır)

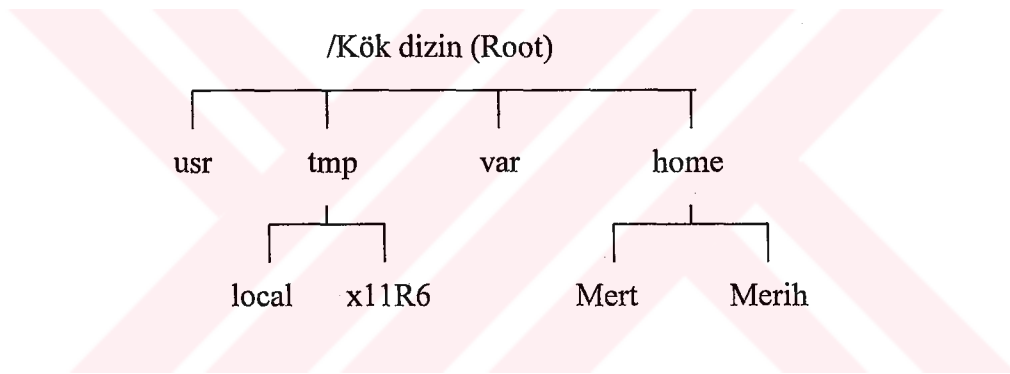
\$cd (Kullanıcının giriş dizinine girer)

\$cd public.html (Etkin dizideki public.html alt dizinine geçmek için kullanılır)

\$cd /tmp (Kök dizindeki tmp dizinine geçeriz)

Hiyerarşik bir sıraya sahip olan Linux'ta en üstte / dizini (Kök dizin) yer alır. Sistemdeki tüm diğer dosya ve dizinler bunun altında yer alır. Aşağıdaki Çizelge 3.1'de Linux dosya ve dizin yapısı gösterilmektedir.

Çizelge 3.1 Linux dosya ve dizin yapısı



3.2.1 Dosyaların Listesi ve İçerikleri

Dosya ve dizinleri görebilmek için ls komutu kullanılır. Seçeneksiz yazarsak bulunduğumuz dizin hakkında bilgi alabiliriz. ls komutunun yazılış biçimi aşağıdaki gibidir:

ls [-seçenekler] [Dosya veya dizin ...]

ls komutunu “-a” seçeneği ile kullanırsak normalde görünmeyen ve nokta karakteri ile başlayan dosya ve dizinlerde görülür. ls komutu “-l” seçeneği ile dosyalar hakkındaki tüm bilgiyi verir. Bunlar dosyanın sahibi, ne zaman yaratıldığı ve grubu gibi bilgilerdir. Bir dizin içindeki tüm alt dizinlerin listesini görmek için ls komutu “-R” seçeneği ile birlikte kullanılır. “a”, “l” ve “R” seçenekleri birleştirilerek de kullanılabilir.

Tüm dosyayı ekrana yazdırmak için cat komutu kullanılır. Bir dosyanın başından veya sonundan itibaren belli miktarda satırı ekrana getirmek de mümkündür. Bunun için head ve tail komutları kullanılır.

3.2.2 Dosya İçeriklerinin less Komutu ile Görüntülenmesi

Dosyaların içeriklerini görmek için kullanılan birkaç komuttan en pratiği less komutudur. Ok tuşları ile dosya içinde hareket edebilir ve q karakteri ile dosyadan çıkabiliriz. Dosyanın içeriği birden fazla sayfadan ibaret ise dosya sayfalar halinde ekrana gelir. less komutu yardımıyla sadece dosyanın içine bakmakla kalmaz dosya üzerinde belli bir isimle arayabiliriz.

less komutu ile kullanılabilecek komutların listesi aşağıdaki gibidir:

Bir satır ileri git: Enter yada aşağı ok

Bir satır geri git: Yukarı ok

Bir ekran boyu ileri git: Boşluk

Bir ekran boyu geri git: b

Dosya üzerinde arama yap: /

Çıkış: q

3.2.3 Dosyaların Kopyalanması

Bir dosya veya dosyalar grubunun bir dizinden başka bir dizine veya aynı dizin içine farklı bir

isimle kopyalamak için kullanılan komut cp komutudur. cp komutunu yazılış biçimi;

cp [-seçenekler] <kaynak> <hedef>

şeklindedir. cp komutu “-b” seçeneği ile kullanılırsa dosyaların üzerine yazılma ya da silinmelere karşı yedekler. Eğer cp komutu “-i ” seçeneği ile birlikte kullanılırsa kopyalarken üzerine yazılacak dosyalar olursa üzerini yazıp yazmamak için onay ister. Eğer “-p” seçeneği ile birlikte kullanılırsa dosyaların orijinal izinlerini korur. Normalde kopyalandığın yerin izinleri atanır. “-r ” seçeneği dizinleri kopyalarken kullanılır. cp komutunun kullanımına bir örnek aşağıdaki gibidir:

```
$ cp yazi.txt/tmp
```

Bu komut ile yazi.txt adlı dosya /tmp dizinine kopyalanır.

Bir dosya veya dosyalar grubunu bir dizinden başka bir dizine veya aynı dizin içine farklı bir isimle aktarmak için kullanılan komut mv komutudur. Yazılış biçimi;

mv [-seçenekler] <kaynak> <hedef>

şeklindedir. mv komutu “-b” seçeneği ile kullanılırsa dosyaları üzerlerine yazılma veya silinmelere karşı yedekler. “-i ” seçeneği ile kullanılırsa isim değiştirirken aynı isimli dosyalar olursa üzerine yazıp yazmamak için onay ister. mv komutunun kullanımına bir örnek aşağıdaki gibidir.

```
$ mv yazi.txt. /tmp /yazi
```

Bu komut ile yazi.txt isimli dosya /tmp dizini altına, ismi yazi olarak değiştirilerek aktarılır.

3.2.4 Dosyaların Silinmesi

Bir dosya veya dosyalar grubunu silmek için Linux'ta rm komutu kullanılır. Linux altında silinen bir dosyanın geri dönüşü ancak mc komutu yardımıyla olur. Bir dizin boş iken rmdir komutu ile silinebilir. Eğer boş değilse bu komut işe yaramaz. rm komutunun yazılış biçimi;

rm [-seçenekler] dosya ...

şeklindedir. Eğer boş olmayan bir dizin silinmek istenirse rm komutu “-r” seçeneği ile kullanılır.

rm komutunun “-f” seçeneği ile kullanıcının onayını almadan tüm dosyaları tek adımda silmek mümkündür. Bulunduğumuz dizindeki tüm dosyaları silmek için “*” karakteri

kullanılır. Örneğin;

\$rm*

komutu bulunduğumuz dizindeki tüm dosyaları siler.

3.3 Linux Kabukları

Sisteme girdiğimiz anda kabuk programları çalıştırılır ve bu andan sonra yapacağımız tüm işlemler bu kabuk programı tarafından yönetilir ve denetlenir. Kabuk, klavyeden girilen komutları çalıştırarak bir ara birim görevi yapar. Unix ve Linux altında geliştirilen sayısız kabuk çeşidi ve bunların her birinin kendine ait özelliği vardır. Her programcı istediği kabuğu seçebilir, sistemde yer alan kullanıcılar farklı kabuklar üzerinde çalışabilirler.

Kabuğun içinde mevcut olan programlama komutları kullanılarak program yazılabilir. Bu programı kabuk scripti (shell script) adı verilen bir dosyaya yerleştirerek bu dosyayı icra edebilir veya çalıştırabiliriz.

Aşağıda sık kullanılan kabuklar hakkında kısa bilgi verilmektedir:

Sh (Shell yada Bourne Shell): İlk Unix kabuğu

Ksh (Korn Shell): Sh uyumlu, birçok ek programlama özelliği içeren bir kabuktur. Bu kabuk da yaygın bir kitle tarafından kullanılmaktadır.

Bash (Bourne Again Shell): Kullanım kolaylığı bakımından en çok rağbet gören kabuklardan biri bash kabuğudur. Bash kabuğu sh ve ksh uyumluluğunu korurken, özellikle etkileşimli kullanıma yönelik (komut tamamlama vb.) birçok yenilik de içerir.

Csh (C Shell): Berkeley Üniversitesi'nde geliştirilen csh'in C programlama diline benzeyen bir programlama yapısı vardır.

Tcsh : Csh'in biraz geliştirilmiş halidir.

Yukarıdaki kabuk programlarında sh, ksh, bash birbirleriyle uyumludur ve yukarıda en ilkinden en gelişmişe göre sıralanmışlardır. Aynı durum csh ve tcsh içinde geçerlidir. Kabuklar / bin dizini altında yer alır. Kullanıcının sisteme girerken hangi kabuğu kullanacağı / etc / passwd dosyasında yer alır. Bunu değiştirmek için chsh (change shell) komutunu kullanabiliriz. Kullanabileceğimiz tüm kabuklar /etc/shells dizininde yer almaktadır.

3.4 Dosya ve Dizin İşlemleri

3.4.1 Erişim Hakları

Erişim hakları, Linux dosya sistem güvenliğinin belkemiğini oluşturur. Her dosyaya ayrı verilebilen erişim izinleri sayesinde çok daha rahat bir sistem yönetimi gerçekleştirilebilir. Linux altında üç çeşit erişim hakkı vardır:

Okuma izni: Dosyanın okuma izni varsa içeriği görülebilir, dizinin okuma izni varsa içerdiği dosyaların listesi alınabilir.

Yazma izni: Dosyanın yazma izni varsa dosyayı değiştirebilir veya silebiliriz. Dizine yazma izni verildiğinde dizin altındaki dosyalar yazılabilir veya silinebilir.

Çalıştırma izni: Dosyayı çalıştırma hakkını verir.

Bir dosya veya dizin ilk yaratıldığında Linux tarafından ön tanımlı bazı izinler verilir. Genellikle bu izin çalıştırma ve okumadır. Dosyanın oluşma anında verilen izni değiştirmek için erişim yetki kalıbı olarak da bilinen umask komutu kullanılır. Erişim hakları dışında bir dosyanın üç izin düzeyi daha vardır. Bunlar dosyanın sahibi, dosyanın grubu ve diğer kullanıcılarıdır.

Dosyanın sahibi, o dosyayı oluşturan kişidir. Her kullanıcının bir grubu da olduğu için dosya oluşturulurken kullanıcı hangi grupta ise dosya da o gruba dahil olacaktır. Dosyanın sahibi olmayan ve grubu da dosyanın grubuyla uyuşmayan sistemdeki kullanıcılar diğer sınıfına girerler.

3.4.2 Dosya İzinlerinin Değiştirilmesi

Dosya izin bilgilerini görebilmek için ls komutu “-l ” seçeneği ile kullanılır. Örneğin, aşağıdaki dosya üzerindeki erişim haklarını inceleyelim.

```
-rwxr-xr-x 2 mert mert 150 Jan 15 12:15 deneme
```

Dosya bilgisinin en sol kısmında izin hakları yer alır. En baştaki “-” işareti bunun düz bir dosya olduğunu gösterir. mert dosyanın sahibi, diğer mert ise grubudur. Red Hat Linux’ta her kullanıcının adına karşılık gelen bir grubu vardır.

Ardından sırayla uzunluk, son değiştirilme tarihi, dosya ismi gibi bilgiler gelir. Solda yer alan r,w ve x karakterleri sırayla okuma, yazma ve çalıştırma haklarını belirtir. En soldaki “-” işaretini ayırdığımız zaman geri kalan harfleri üç adet üçlü grup haline getirelim. Sırayla

birinci harf kümesi dosya sahibinin izinlerini, ikinci harf kümesi grup izinlerini ve son küme de diğer kullanıcıların izinlerini belirtir. Buna göre yukarıdaki dosyada;

rwX: Kullanıcı okuyabilir, yazabilir, çalıştırılabilir.

r-x: Grup okuyabilir, çalıştırabilir fakat yazamaz.

r-x: Diğerleri okuyabilir, çalıştırabilir fakat yazamaz.

Dosyanın izinlerini değiştirmek için chmod komutu kullanılır. bu komutun yazılış biçimi aşağıdaki gibidir.

chmod [-seçenekler] <değişken mod > <dosya|dizin> [dosya|dizin...]

Bu komutu kullanırken hangi izin düzeyini (kullanıcı, grup veya diğerleri) hangi izinlerin verileceği veya kaldırılacağı yazılır.

Örnek olarak yukarıdaki dosyanın okuma iznini kaldırmak için;

\$chmod -r deneme

komutu kullanılabilir.

“-“ işareti, iznin kaldırılacağını belirtir.

“+” işareti, izin vermek için kullanılır.

Bu komutun kullanımından sonra dosyanın yeni görüntüsü şöyle olacaktır;

--wx--x--x 2 mert mert 150 Jan 15 12:15 deneme

Sadece grubun, sadece kullanıcının veya sadece diğerlerinin erişim hakkını değiştirebilmek için her izin düzeyi için bir harf tanımlanmıştır.

Kullanıcı için “u” (user), grup için “g” (group) ve diğerleri için “o” (others) yazarak belirli bir izin düzeyi için erişim hakkını değiştirmek mümkün olur. Bu üç harf izinden hemen önce yazılır.

\$chmod u+x deneme (Dosyanın sahibi çalıştırabilir)

\$chmod o+r deneme1 (Diğerleri okuyabilir)

\$chmod g-w deneme2 (Dosyanın grubu yazamaz)

Her izin için aynı zamanda sayılarda tanımlanmıştır. Dosyanın erişim izinlerini değiştirmek

için her izine ait sayıları kullanabiliriz .

Aşağıda, her izin düzeyi (Dosyanın sahibi, grubu ve diğerleri) için üçer tane olmak üzere toplam dokuz sayıdan oluşan izin numaraları görülmektedir.

- 400 Sahibi okur
- 200 Sahibi yazar
- 100 Sahibi çalıştırır
- 040 Grubu okur
- 020 Grubu yazar
- 010 Grubu çalıştırır
- 004 Diğerleri okur
- 002 Diğerleri yazar
- 001 Diğerleri çalıştırır

Bir izin vermek için sıra ile numaraları toplarız.

Örneğin; Sahibinin okuması (400), yazması (200), çalıştırması (100)

Grubunun okuması (40), çalıştırması (10)

Diğerlerinin okuması (4), çalıştırması (1) için kullanılan rakamların toplamı 755'tir.

\$chmod 755 deneme

komutu ile yukarıdaki örnekteki izinler verilmiş olunur. Aynen dosyalarda olduğu gibi bir dizinin de sahibi ve grubu, izin hakları vardır.

3.4.3 Dosyanın Sahibi ve Grubu

Bir dosyanın sahibini sadece sistemdeki root kullanıcısı değiştirebilme yetisine sahiptir. Dosya sahibinin değiştirilebilmesi için chown komutu kullanılır.

chown komutunun yazılış biçimi aşağıdaki gibidir:

chown [-seçenekler] <kullanıcı[.grup]> <dosya|dizin>[dosya|dizin...]

Kullanımı isteğe bağlı seçenekler bölümünde “-R” rekürsif (özyineli) anlamındadır. Verilen

bir dizindeki tüm dosya ve alt dizinlerin sahibini değiştirmek için kullanılır.

“-V” chown komutu tarafından gerçekleştirilen tüm işlemleri ve komutun icrası sonucu sahibi değişmiş olan ve değişmeyen dosyaları listeler.

“-C” seçeneği “-V”ye benzer fakat sadece sahibi değişen dosyaları listeler.

```
-rwxr-xr-x 2 mert mert 150 Jan 15 12:15 deneme
```

Yukarıdaki dosyanın sahibini merih isimli kullanıcı yapalım.

```
#chown merih deneme
```

Komutu ile dosyanın yeni hali;

```
-rwxr-xr-x 2 merih mert 150 Jan 15 12:15 deneme
```

şeklinde olur.

Dosyanın grubunu değiştirmek için chgrp komutu kullanılır

chgrp komutunun yazılış biçimi aşağıdaki gibidir:

```
chgrp [-seçenek] <grup> <doya|dizin> [dosya|dizin...]
```

seçenekler kısmında yer alacak parametreler chown komutundakilerinin aynısıdır ve benzer şekilde kullanılır.

Yukarıdaki deneme dosyasının grubunu www yapmak için;

```
#chgrp www deneme
```

komutu kullanılabilir.

Tabi burada yeniden değiştirdiğimiz kullanıcılar ve grupları da sistemde tanımlı olmalıdır. chown komutu, dosyanın hem kullanıcıasını hem de grubunu değiştirme özelliğine sahiptir. chown komutundan sonra kullanıcı ve grup isimlerinin aralarında bir nokta olacak şekilde yazılmalıdır.

```
$chown merih.www deneme
```

komutu ile dosyanın yeni hali aşağıdaki gibi olacaktır:

```
-rwxr-xr-x 2 merih www 150 Jan 15 12:15 deneme
```

3.5 Dizin Tarama

Linux işletim sistemi ilk kurulduğunda yüzlerce dizin altında, binlerce dosya olacaktır. find komutu, bu dosyaların arasında tarama yapabilmek için geliştirilmiştir. Bu komutla sadece dosya isimlerini değil, dosyanın sahibi, erişim hakları, son erişim tarihi gibi verilere de ulaşmak mümkündür. find komutunun en sık kullanım şekli aşağıdaki gibidir:

```
find <dizin ismi> -name <aratılan karakter kümesi>
```

Dizin ismi yerine taranacak olan dosyanın bulunduğu dizin ismi yazılır.

“-name” parametresinden sonra ise aranan dosyanın ismi girilir.

Burada “*” ve “?” gibi joker karakterleri de kullanılabilir.

Aşağıda find komutunun kullanımına ilişkin örnekler verilmektedir:

```
$find . -name "*.html"
```

komutu bulunan dizinden itibaren sonu “.html” ile biten dosyaları arar.

```
$find /home/mert -name "deneme"
```

komutu ile /home/mert dizininden itibaren deneme isimli dosyaları arayacaktır.

```
$find /usr/ -name "linux" -type d
```

komutu “-type d” parametresi yardımıyla sadece dizin isimleri arasında arama yapar.

```
#find /usr/src -size +500k -print
```

komutu /usr/src altında yer alan ve büyüklüğü 500 kb’tan fazla olan tüm dosyaları ekrana yazdırır.

3.6 Metin Düzenleyiciler

3.6.1 Pico Editörü

Pico, çok geniş bir kesim tarafından kullanılan, kolay kullanımlı bir metin düzenleyicidir. Komut istemcisi üzerindeyken;

```
$pico
```

yazılarak program çalıştırılır. Gerek duyulan komutlar ekranın alt kısmında iki satır halinde

yer alır. Bir dosyayı yüklemek için ctrl-R tuşuna basılır. Dosya üzerinde istenilen değişiklik yapılabilir. Dosyayı kaydetmek için ctrl-O tuşuna basılır. Pico programında tüm komutlar ctrl tuşuyla birlikte verilir. Komut listesi, Çizelge 3.2’de gösterilmiştir.

Çizelge 3.2 Komut listesi

İşlem	Tuş Karşılığı
Bir satır yukarı git	Yukarı ok tuşu
Bir satır aşağı git	Aşağı ok tuşu
Sola git	Sola ok tuşu
Sağa git	Sağa ok tuşu
Satırın sonuna git	Control-E (ctrl-E)
Satırın başına git	Control-A (ctrl-A)
Satırı sil	Control-K (ctrl-K)
Dosyayı oku	Control-R (ctrl-R)
Arama yap	Control-W (ctrl-W)
Pico’dan çık	Control-X (ctrl-X)
Dosyayı kaydet	Control-O (ctrl-O)
Bir sayfa aşağı in	Control-V (ctrl-V)
Bir sayfa yukarı çık	Control-Y (ctrl-Y)

3.6.2 Vi Editörü

Unix işletim sistemi ile kullanılabilen birçok editör programı bulunmaktadır. Bu editörler farklı kuruluşlar tarafından geliştirilerek kullanıcıların hizmetine sunulmuştur. Bunlar arasında Ed editörü etkileşimli bir program olup özel komutlar yardımıyla metinler oluşturulur.

Vi editörü Berkeley Üniversitesi’nde geliştirilmiştir. Daha sonra da diğer Unix kullanıcıları tarafından tercih edilen bir editör olmuştur. Vi ile genel anlamda iki tür işlem yapmak olasıdır. Birincisi yeni bir metin dosyası yaratmak, ikincisi ise mevcut bir dosyayı çağırarak üzerinde işlemler yapmaktır.

Vi editörü;

Vi dosya adı

şeklinde çalıştırılabilir.

Vi editöründe iken eğer dosya üzerinde bazı düzeltmeler yapmak üzere çağırılıyorsa;

vedit dosya adı

biçimi kullanılabilir.

Herhangi bir düzeltme yapılmaması ve sadece görüntülenmesi amaçlanıyorsa;

View dosya adı

tanımını tercih edilebilir.

Vi programı, çeşitli işlemleri yapmaya yönelik komut kullanma olanağı sağlar. Bu komutların önemlileri ve işlevleri aşağıda gösterilmektedir.

Dosyalarla ilgili işlemler:

Zz Dosyanın yazılması suretiyle Vi'dan çıkış

:w Dosyanın yazılmasını sağlar

:w! Mevcut dosyanın üzerine yazılmasını sağlar

:r Başka dosyaların okunması amacıyla kullanılır

:q Vi'dan çıkış

:q! Değişiklikleri gözardı ederek Vi'dan çıkış

:n Bir sonraki dosya

Unix komutlarına erişim :

:sh Vi'dan Unix komut yardımcısına geçiş

:! Unix komutlarını çalıştırma olanağı sağlar

Arama işlemleri :

/kalıp Belirli bir kalıbın ileri doğru aranması

?kalıp Kalıbın geriye doğru aranması

n İleriye doğru arama işlemlerinin tekrarlanması

N Geriye doğru arama işlemlerinin tekrarlanması

Karakter ve satır ekleme, değiştirme ve silme işlemleri :

a İmlecin sağ tarafından itibaren bilgi girişi

A Satırın sonuna bilgi girişi

- i İki karakter arasına bilgi girişi
- dv Bir kelimenin silinmesi
- dd Bir satır siler
- x Bir karakter siler
- o İmlecin bulunduğu satır ile bir sonraki satır arasına boş satır ekler
- O Bir önceki satır ile imlecin bulunduğu satır arasına boş satır ekler
- r İmlecin üzerinde bulunduğu karakteri değiştirir
- cw Bir kelimenin tümüyle değiştirilmesini sağlar
- C Bir satırın tümüyle değiştirilmesini sağlar

3.7 Linux Sistem Yönetimi

3.7.1 Kullanıcı Hesabı Açılması

Kullanıcı hesabı adduser komutu ile açılabilir. Kullanıcı isminin üç karakterden az olmamasına özen göstermek gerekir. adduser komutu, eğer özel olarak değiştirmek istemezsek, her kullanıcının grup kimliğini (GID) kendisi tanımlayacaktır. Bu gruba karşılık gelen grup ismi (group name) aynı zamanda söz konusu kullanıcının kendi adıdır. Örnek olarak mert adında bir kullanıcı yaratmışsak, bu kullanıcının grubu da otomatik olarak mert olacaktır. Kullanıcı kimliği (UID=User ID) ise genellikle 500'den sonra ve kullanılmayan rakamlar olarak atanır.

3.7.2 Kullanıcı Grubu

Eğer Linux makinesi birkaç yoğun görev üstlenmişse (örneğin www veya ftp sunucusu olarak kullanılıyorsa) her bir işlem için birden fazla grup tanımlanabilir. Grubuna bağlı kalmadan wwwadm veya ftpadm gibi grup isimleri belirtilebilir, bu grupları da bir veya birkaç kullanıcıya ait olacak şekilde değiştirebiliriz.

Bir kullanıcının kullanıcı kimliğini öğrenmek için id komutu kullanılabilir.

Birden fazla ek grup tanımlanması, sistemin denetlenmesini bir miktar güçleştirse de sistem güvenliğini artırır. Bazı kullanıcıların birden fazla grubunun olmasıyla bu kullanıcılar rahatlıkla sistemde root kullanıcısının üstlenmesi gereken işleri yapabilirler. Grup kimliğinin

değiştirilmesi veya yeni grubun eklenmesi için /etc/group dosyası kullanılır. Bu dosyaya grubun ismi, numarası, grup parolası ve gruba ait kullanıcılar yazılır. Her kullanıcının en az bir grubu vardır.

/etc/group dosyasının iki satırında yer alan parametreler;

root :: 0 : root

bin :: 1 : root, bin, daemon gibidir. Sırayla “:” karakteri ile ayrılmış her kolon;

- 1) Grup ismi
- 2) Grup parolası
- 3) Grup numarası
- 4) Gruba ait kullanıcılar

şeklinde özetlenebilir.

İlk sütun grup ismini belirtir. Grup ismi ls-al komutunu işletince kullanıcı isminin hemen sağında görülen kelimedir. Grup parolası, gruba ait olmayan kullanıcıların dosyalarına erişim sağlamak için kullanılır. Grup parolasına nadiren başvurulur. Grup numarası, kullanıcının ait olduğu grubun karşılığıdır. Son sütunda ise gruba ait kullanıcılar, her biri birer virgülle ayrılmış olarak yazılır. Kullanıcı sisteme girdiği zaman otomatik olarak /etc/passwd dosyasında yer alan grup ismi aktif olur, ancak kullanıcı dilerse kendisine tanımlanan gruplardan birine istediği zaman geçiş yapabilir. Sistem ilk kurulduğu anda birkaç grup hali hazırda zaten vardı. Örneğin, root grubunun grup numarası 0'dır ve pek çok sistem dosyasının sahibidir. bin grubunun grup numarası ise 1 olup /usr/bin ve /bin dizinlerindeki bazı dosyaların grubudur.

Hangi grubun (veya tanımlanmışsa grupların) içinde olduğumuzu öğrenmek için groups komutunu kullanabiliriz.

3.7.3 Yeni Kullanıcı Grubu Eklenmesi

Kullanıcı grupları, bir projeyi aynı anda birden fazla kişinin yürütülebilmesi için kullanılabilir. Gruplar, belirli kullanıcıların ait oldukları kümelere verilen isimdir. Grup işlemleri için usercfg programı kullanılabilir veya komut satırından groupadd komutundan faydalanılabilir. Örneğin, sisteme staj adında bir grup eklemek için;

#groupadd staj

komutu kullanılabilir.

Bir kullanıcıyı açılan bir gruba yerleştirmek için ise;

#gpasswd - a mert staj

komutu kullanılır.

Bu komutla mert kullanıcısı staj grubuna dahil edilmiştir. Bu aşamadan sonra staj grubunun tüm hakları mert içinde geçerlidir. Örneğin, bir dosya staj grubuna dahil olsun ve gruba yazma hakkı verilmiş olsun. Bu durumda mert dosyayı diğer staj grubu üyeleri gibi silebilecektir.

3.7.4 Kullanıcı Grubunun Silinmesi

Bir kullanıcı grubunu sistemden kaldırmak için groupdel komutunu kullanabiliriz. Örneğin, staj adlı grubu silmek için;

#groupdel staj

komutu kullanılır.

3.7.5 Kullanıcı Hesabının Silinmesi

Bir kullanıcıya ait hesabı silmek için /etc/passwd dosyasında bu kullanıcıya ait olan satırdaki parola kolonunun başına “*” işareti koyulur. Şifrelenmiş bir parolanın içinde “*” karakteri olamayacağından bu kullanıcı sisteme bir daha giremez. Hesabı tamamen silmek için, kullanıcının ev dizinini ve e-postalarını yedekledikten sonra userdel komutu kullanılır. Örnek olarak daha önce yaratılan mert kullanıcısının ev dizinini (/home/mert), /etc/passwd ve /etc/shadow dosyasında yer alan kimlik bilgilerini silmek için aşağıdaki komut yazılır:

#userdel - r mert

Burada “-r “ parametresi, kullanıcının ev dizinini silmek için kullanılır.

4. POSTGRESQL VERİTABANI YÖNETİM SİSTEMİ

Geleneksel anlamdaki İlişkisel Veritabanı Yönetim Sistemleri, özel türlerin özelliklerinden oluşan adlandırılmış ilişkilerin bileşiminden oluşur. Şu anki ticari sistemlerde kullanılan olası türler şu şekilde sıralanabilir; kayan noktalı sayılar, tamsayılar, karakterler ve karakterlerden oluşan katarlar, para ve tarih.

Gelecekteki veri işleme uygulamaları için bu tür modellerin yetersiz kalacağı görüşü hakimdir. İlişkisel model, bazı alanlarda eski modellerin yerini basitliği sayesinde olmaya başlamıştır. Tahmin edildiği gibi bu basitlik, karmaşık uygulamaların uygulanmasında sorunlarla karşılaşılmasına neden olmaktadır.

Postgres, bazı basit kavramları kullanıcıların sistemlerini kolayca genişletilebilecekleri şekilde birleştirerek, çok güçlü bir yapı sunmaktadır.

Bu kavramlar dört tanedir:

- 1) Sınıflar (Classes)
- 2) Kalıtım (Inheritance)
- 3) Türler (Types)
- 4) Fonksiyonlar (Functions)

Postgres'e ek güç ve esneklik sağlayan diğer yapılarda şu şekilde sıralanabilir:

- 1) Kısıtlar (Constraints)
- 2) Tetikler (Triggers)
- 3) Kurallar (Rules)
- 4) İş bütünlüğü (Transaction integrity)

Bu özellikleriyle Postgres, nesne ilişkisel veritabanları sınıfına girer.

Nesne ilişkisel veritabanlarının, geleneksel ilişkisel veritabanlarını genelde desteklemeyen nesne yönelimli veritabanlarından farklı olduğu unutulmamalıdır.

Yani Postgres, bazı nesne yönelimli özelliklere sahip olmasına rağmen, ilişkisel veritabanları arasındadır.

İlişkisel Veritabanı Yönetim Sistemi olarak Postgres'in 1986 yılında Berkeley

Üniversitesi'nde başlayan gelişimi günümüze kadar ulaşmış ve halen üzerinde yapılan çalışmalar ile gereksinimlere yanıt vermektedir.

Postgres 1991 itibariyle pek çok araştırma ve uygulama için kullanılabilir oldu. Bunlardan bazıları; finansal veri analiz sistemi, bir jet motoru için başarımlı gözetim sistemi ve birkaç coğrafi bilgi sistemidir. Birkaç üniversitede de öğretim aracı olarak da kullanılmıştır. Sonunda Illustra Information Technologies, kodları toplayarak Postgres'i ticari bir marka haline getirdi.

Postgres, 1992'nin sonunda Sequoia 2000 bilimsel hesaplama projesinde ana veri yöneticisi olarak seçildi.

1994 yılında Andrew Yu ve Jolly Chen tarafından Postgres'e bir SQL dili yorumlayıcısı eklendi. Postgres 95 olarak adlandırılan bu sürüm Berkeley'de geliştirilen orijinal Postgres'in bir devamıdır ve orijinalinden %25 daha kısa şekilde tamamen ANSI C'de yazılmıştır.

1996 yılında Postgres 95 ismi PostgreSQL olarak değiştirdi. Bu isim, Postgres'in yeni sürümlerindeki SQL uyumluluğunu vurgulamak için seçildi.

4.1 Veritabanının Yönetimi

Veritabanının yönetimi için, veritabanının yaratılması, güncellenmesi, silinmesi, veritabanına erişim gibi temel özelliklerin gerçekleştirilmesi gerekmektedir.

4.1.1 Veritabanının Yaratılması

Veritabanının yaratılması iki farklı yoldan yapılabilir. Birinci yol, Postgres'e girerek create database komutu kullanılabilir veya aynı işleve sahip createdb komutu, komut satırından çalıştırılabilir. Bu işlevlerin gerçekleştirilebilmesi için komutu çalıştıran kullanıcının veritabanı yaratma hakkına sahip bir kullanıcı olması gerekmektedir. Bir kullanıcının Postgres'de veritabanı yaratılabilmesi için, Postgres'in kendi veritabanında bu kullanıcının bulunması gerekmektedir.

Postgres'e bir kullanıcı eklemek için;

```
Createuser root
```

komutu kullanılır. Bu işlem ile birlikte root kullanıcısı PostgreSQL'de veritabanı oluşturma hakkına sahip olur.

Postgres'de veritabanı isimleri bir karakter ile başlamalıdır. İsimlerin uzunluğu 32 karakterle

sınırlandırılmıştır. Bir kullanıcı yarattığı veri tabanının sahibidir.

4.1.2 Veritabanına Erişim

Veritabanının yapılandırılmasının ardından veritabanına erişim için psql türü metin tabanlı gözetim uygulamalarından biri çalıştırılabilir ya da C, Perl veya TCL gibi programlama dilleri kullanılır. Vt1 isimli veritabanıyla etkileşmek için;

```
psql vt1
```

komutu girilirse, aşağıdaki mesajla karşılaşılır:

```
welcome to the POSTGRESQL interactive sql monitor.
```

```
Please read the file COPYRIGHT for copyright terms of POSTGRESQL
```

```
type\? for help on slash commands
```

```
type\q to quit
```

```
type\g or terminate with semicolon to execute query
```

```
You are currently connected to the database: vt1
```

```
Vt1→
```

Bu istemci vt1 isimli veritabanına bağlanıldığını ve gözetim programının kullanıcıdan komut beklediğini belirtir. Burada her türlü SQL komutu çalıştıracağı gibi, PostgreSQL tarafından komut olarak algılanan “/” karakteri ile başlayan kaçış kodları da girilebilir. Örneğin, “/h” kaçış kodu ile PostgreSQL komutları hakkında yardım alınabilir.

```
Vt1→\h
```

Bu kod verildiğinde kullanıcı sistem içerisinde bulunan komutların bir listesi ile karşılaşır.

Sorgulamalar yapıldıktan sonra bunları Postgres’e aktarmak için;

```
Vt1→\g
```

kaçış kodu girilebilir. Bu komut Postgres’e önceden girilen sorguyu çalıştırması için gerekli sinyali gönderir. Eğer sorgu “;” ile bitirilirse, Postgres sorguyu kendiliğinden çalıştırır, “/g” koduna gerek kalmaz.

Sorguları etkileşimli olarak girmek yerine önceden hazırlanan bir dosyadan okumak için “\i ”

kaçış kodu girilebilir.

Vt1 → \i dosya ismi

komutu ile “dosya ismi” isimli dosyadan sorgular okunur, çalıştırılır ve ardından etkileşimli gözetim programına dönülür.

Postgres’den çıkıp Unix ortamına dönmek için ise “q” kaçış kodu kullanılır. psql programı kullanılırken boşluk, TAB ve NL karakterleri kullanılabilir.

Tek satırlık yorumlar “...” ile başlayan satırlarla, birden çok satırlı yada bir satırın içindeki yorumlar ise “/*....*/” arasında kalacak şekilde yazılabilirler.

4.1.3 Veritabanının Silinmesi

Bir veritabanının sahibi kendi veritabanının destroydb Unix komutunu kullanarak silebilir.

Örneğin vt2 isimli bir veritabanını silmek için;

destroydb vt2

komutu kullanılır. Bu komut ile vt2 veritabanı ile ilgili her türlü dosya geri dönüşsüz biçimde silinecektir.

4.2 Veri Türleri

Postgres oldukça zengin bir veri türü listesine sahiptir. Ayrıca kullanıcının bu listeye ekler yapabilmesini sağlayan DEFINE TYPE komutunu da içermektedir. Postgres’de kullanılan veri türleri Çizelge 4.1’de gösterilmiştir.

Çizelge 4.1 Postgresql’de veri türleri

Postgres tipi	SQL 92 veya SQL tipi	Tanım
Bool	Boolean	Mantıksal sonuçlar için kullanılan veri tipidir. Değeri “true” veya “false” olabilir.
Box		İki boyutlu uzayda bir dikdörtgenin köşe noktalarının değerlerini tutmak için kullanılan veri tipidir.
Char(n)	Character(n)	Sabit uzunluklu karakter katarları için kullanılan bir veri türüdür.
Circle		İki boyutlu uzayda bir dairenin bilgilerini tutmak için kullanılan bir veri türüdür.
Date	Date	Takvim tarihini tutmak için kullanılır. fakat gün ve saat bilgilerini içermez.
Float 4/8	Float(p)	P duyarlıklı kayan noktalı sayıları tutmak için kullanılan bir veri türüdür.
Int 2	Smallint	İşareti iki baytlık tamsayıları tutmak için kullanılan bir veri türüdür.
Int 4	Int,Integer	İşareti dört baytlık tamsayıları tutmak için kullanılan bir veri türüdür.
Int 8		İşareti sekiz baytlık tamsayıları tutmak için kullanılan bir veri türüdür.
Line		İki boyutlu bir uzayda bir doğrunun bilgilerini tutmak için kullanılan bir veri türüdür.
Lseg		İki boyutlu bir uzayda bir doğru parçasının bilgilerini tutmak için kullanılan bir veri türüdür.
Money	Decimal (9,2)	Amerikan para birimi (\$) türündeki bilgileri tutmak için kullanılan bir veri türüdür.
Path		İki boyutlu bir uzayda açık ve kapalı geometrik yolların bilgilerini tutmak için kullanılan bir veri türüdür.
Point		İki boyutlu bir uzayda bir geometrik noktanın bilgilerini tutmak için kullanılan bir veri türüdür.
Polygon		İki boyutlu uzayda kapalı geometrik yol bilgilerini tutmak için kullanılan bir veri türüdür.
Time	Time	Zaman bilgisini tutmak için kullanılan bir veri türüdür.
Timespan	Interval	Genel olarak zaman bilgilerini tutmak için kullanılan bir veri türüdür.
Varchar(n)	Character varying(n)	Değişken uzunluklu karakter katarlarını tutmak için kullanılan bir veri türüdür.

4.3 Operatörler

4.3.1 Sayısal Operatörler

Aşağıdaki Çizelge 4.2’de Sayısal Operatörler, bunların tanımları ve kullanımları gösterilmektedir.

Çizelge 4.2 Postgresql’de kullanılan sayısal operatörler

Operatör	Tanım	Kullanımı
!	Faktoriyel işlemi için kullanılır.	3!
!!	Faktoriyel işlemi için kullanılır.	!!3
%	Modül alma işlemi için kullanılır.	5%4
*	Çarpma işlemi için kullanılır.	2*3
+	Toplama işlemi için kullanılır.	2+3
-	Çıkarma işlemi için kullanılır.	2-3
/	Bölme işlemi için kullanılır.	4/2
:	Doğal üs alma (e üzeri) işlemi için kullanılır.	: 3.0
;	Doğal logaritma işlemi için kullanılır.	(; 5.0)
@	Mutlak değer işlemi için kullanılır.	@-5.0
^	Üs alma işlemi için kullanılır.	2.0^3.0
/	Kare kök işlemi için kullanılır.	/25.0
/	Küp kök işlemi için kullanılır.	/27.0

4.3.2 Sözcüksel Operatörler

Aşağıdaki Çizelge 4.3’te Sözcüksel Operatörler ve bunların tanımları gösterilmektedir.

Çizelge 4.3 Postgresql’de kullanılan sözcüksel operatörler

Operatörler	Tanım
[]	Dizi ayırıcı olarak kullanılan operatördür.
.	Tablo/kolon ayırıcı olarak kullanılan operatördür.
;	Deyim sonlandırıcısı olarak kullanılan operatördür.
	Aralık başlangıcı olarak kullanılan operatördür.
IS	TRUE, FALSE ve NULL değerlerini denemek için kullanılan operatördür.
IS NULL	NULL değerlerini denemek için kullanılan operatördür.
NOT NULL	NOT NULL değerlerini denemek için kullanılan operatördür.
IN	Üyelik denemek için kullanılan operatördür.
BETWEEN	Aralıkta içerilme durumu için kullanılan operatördür.
LIKE	Katar türündeki örüntülerin taraması için kullanılan operatördür.
<>	Mantıksal eşitsizliklerde kullanılan operatördür.
=	Eşitlik için kullanılan operatördür.
NOT	Mantıksal olumsuzluklar için kullanılan operatördür.
AND	Mantıksal kesişim için kullanılan operatördür.
OR	Mantıksal birleşim için kullanılan operatördür.

4.3.3 Genel Operatörler

Aşağıdaki Çizelge 4.4’te Genel Operatörler ve bunların tanımları gösterilmektedir.

Çizelge 4.4 Postgresql’de kullanılan genel operatörler

Operatör	Tanım	Kullanımı
<	Küçüklük karşılaştırması için kullanılan operatördür.	1<2
<=	Küçüklük veya eşitlik karşılaştırması için kullanılan operatördür.	1<=2
<>	Eşit olmama karşılaştırması için kullanılan operatördür.	1<>2
=	Eşitlik karşılaştırması için kullanılan operatördür.	1=1
>	Büyüklik karşılaştırması için kullanılan operatördür.	2>1
>=	Büyüklik veya eşitlik karşılaştırması için kullanılan operatördür.	2>=1
	Katarları birleştirmede kullanılan operatördür.	‘Postgre’ ‘SQL’
!=	İçermeme karşılaştırması için kullanılan operatördür.	3!=i
~~	Benzeme karşılaştırması için kullanılan operatördür.	‘scrappymarch,hermit’~ ~‘%scrappy%’
!~	Benzememe karşılaştırması için kullanılan operatördür.	‘bruce’!~‘%al%’
~	Büyük küçük harf ayrımı dikkate alınmaksızın uyma karşılaştırması için kullanılan operatördür.	‘thomas’~‘.*thomas.*’
~*	Büyük küçük harf ayrımı dikkate alınarak uyma karşılaştırması için kullanılan operatördür.	‘thomas’~*‘.*Thomas.*’
!~	Büyük küçük harf ayrımı dikkate alınmaksızın uymama karşılaştırması için kullanılan operatördür.	‘thomas’!~‘.*Thomas.*’
!~*	Büyük küçük harf ayrımı dikkate alınarak uymama karşılaştırması için kullanılan operatördür.	‘thomas’!~*‘.*vadim.*’

4.4 Sorgulama Dili

Postgres’in sorgulama dili SQL92’nin bir benzeridir. Postgres’in sorgulama dilinde SQL3’e ek olarak, bazı veri türü sistemleri, kalıtım ile ilgili özellikler, fonksiyonlar ve kurallar vardır. Postgres’deki ana kavram sınıflardır.

Bir veritabanı sınıflardan oluşurken, pek çok veritabanı bir postmaster uygulaması tarafından denetlenebilir.

4.4.1 Sınıf Oluşturma

Yeni bir sınıf oluşturmak için oluşturulacak sınıfın ismi ile birlikte içereceği alanların isimleri ve türleri de verilmelidir. Yeni bir sınıf oluşturmak için;

CREATE DATABASE isim;

komutu kullanılır. Bu komut belirtilen isimdeki sınıfı oluşturur. Burada DATABASE ile belirtilen veritabanı, içinde çok sayıda veri tabanı kütüğü yada SQL terimleri ile tablo bulunan bir çevre bellek bölgesi veya bir altdizin adıdır.

Örneğin, öğrenci adında bir sınıf oluşturalım:

```
CREATE TABLE ogrenci (
```

```
isim text,
```

```
soyad text,
```

```
numara int,
```

```
tarih date,
```

```
);
```

CREATE ve TABLE anahtar sözcükleri büyük küçük harf ayırımına duyarlı değilken öğrenci belirteci büyük küçük harf ayırımına duyarlıdır.

Yukarıdaki örnekte isim ve soyad text türünde, numara int türünde ve tarih de date türünde olmak üzere toplam dört alan vardır.

4.4.2 Bir Sınıfa Veri Yükleme

Bir sınıfa veri girişi veya veri yüklenmesi işlemi için INSERT komutu kullanılır. Yazılış biçimi;

```
INSERT INTO Tablo adı (sütun adı 1,...,sütun adı n) VALUES (değer 1,...,değer n);
```

şeklindedir. Aşağıda bu komutun bir kullanımı gösterilmektedir:

```
INSERT INTO ogrenci VALUES ('Merih','BAL',75,'12/15/2001');
```

Ayrıca, çok fazla miktarda verinin bir adımda yüklenebilmesi için COPY komutu kullanılır. Bu işlem, bir metin dosyası okunarak yapılır ve sınıfa doğrudan yazım olduğundan dolayı çok hızlıdır. COPY komutunda bir örnek aşağıdaki gibidir:

```
COPY INTO ogrenci FROM '/home/user/ogrenci.txt' USING DELIMITERS'|';
```

Burada dikkat edilmesi gereken nokta, belirtilen dosyanın istemcinin değil sunucunun kalıcı belleğinde olması gerektiğidir.

4.4.3 Bir Sınıfın Sorgulanması

Bir sınıftan gerekli bilgileri elde etmek için sorgulama yapabilecek komut SELECT komutudur. SELECT komutu, istenilen özelliklerin belirtildiği bir hedef listeden ve kısıtların

belirtildiği nitelik bölümlerinden oluşur. Bu komutun en basit şekli aşağıdaki gibidir:

```
SELECT * FROM ogrenci
```

Bu komut ile ogrenci tablosundaki tüm veriler herhangi bir kısıtlama olmaksızın ekranda görüntülenecektir.

SELECT sözcüğünü izleyen kısımda sütun (alan) adları, FROM sözcüğünü izleyen kısımda ise tablo ismi belirtilmektedir. Birbirinin aynı olan satırların, listeleme esnasında bir kez yazılması için SELECT komutuna DISTINCT sözcüğü eklenir.

SELECT komutunun farklı kullanımları aşağıdaki gibidir:

```
SELECT isim,soyad FROM ogrenci;
```

```
SELECT * FROM ogrenci WHERE soyad='BAL' and numara > 60;
```

```
SELECT DISTINCT isim FROM ogrenci ORDER BY isim;
```

4.4.4 Select Sorgularının Yönlendirilmesi

Her sorgu başka bir sınıfı oluşturmak veya değiştirmek için yönlendirilebilir.

```
SELECT * INTO TABLE liste FROM ogrenci;
```

Bu komut ile ogrenci sınıfının içerdiği özelliklere ve türlere bağlı kalarak, verileri de alarak liste isminde yeni bir sınıf oluşturur ve bu verileri liste sınıfına yerleştirir.

4.4.5 Bir Sınıfın Güncellenmesi

Bir sınıfta verilerin güncellenmesi için UPDATE komutu kullanılır. UPDATE komutunu koşullu yada koşulsuz olarak kullanmak mümkündür. Koşul belirtilmemişse, belirtilen değişiklik tüm tablo satırları üzerinden gerçekleştirilir. Koşul belirtiliği takdirde sadece koşulu sağlayan satırlar üzerinde değişiklik gerçekleşir.

UPDATE komutunun yazılış biçimi aşağıdaki gibidir:

Koşulsuz ise,

```
UPDATE tablo adı
```

```
SET kolon adı 1=değer 1, kolon adı 2=değer 2,.....,kolon adı n=değer n;
```

Koşul olduğu takdirde,

UPDATE tablo adı

SET kolon adı 1=değer 1,kolon adı 2=değer 2,.....,kolon adı n=değer n

WHERE koşul;

Örneğin, numarası 50 olan öğrencinin ismini mustafa olarak düzeltmek istersek;

UPDATE ogrenci SET isim='mustafa' WHERE numara=50;

komutu kullanılır.

4.4.6 Bir Sınıftaki Verilerin Silinmesi

Verilerin silinmesi işlemi için DELETE komutu kullanılır. Verilerin silinmesi koşullu ya da koşulsuz olarak gerçekleştirilebilir. Bu komutun kullanımına ilişkin örnekler aşağıdaki gibidir:

DELETE FROM ogrenci WHERE numara=45;

Komutu ile numarası 45 olan veriler silinir.

DELETE FROM ogrenci;

Komutu ile ogrenci sınıfındaki tüm veriler silinir.

4.4.7 Toparlama Fonksiyonları

Diğer pek çok sorgulama dili gibi PostgreSQL de toplama fonksiyonlarını destekler. Postgres'in şu anki uygulamaları toplama fonksiyonları için bazı kısıtlar içerir. Bu tür fonksiyonlar yalnızca hedef listesi içinde kullanılabilirler, nitelik bölümünde kullanılamazlar.

SELECT max(numara) FROM ogrenci;

Komutu geçerli bir kullanımken;

SELECT isim FROM ogrenci WHERE numara=max(numara);

komutu Postgres için geçerli bir komut değildir.

Ancak, iç içe sorgular kullanılarak yukarıda istenen seçme işlemi gerçekleştirilebilir.

SELECT isim FROM ogrenci WHERE

Numara=(SELECT max(numara) FROM ogrenci);

4.5 SQL Komutları

4.5.1 Abort

O an yapılan işlemi kesmek için kullanılır. Parametre almaz.. Komut kullanıldığında, o an çalışan işlemin işleyişi kesilir ve o işlem tarafından yapılan tüm değişiklikler iptal edilir. Bütün değişiklikleri iptal etmek için;

```
ABORT WORK;
```

SQL komutu kullanılabilir.

4.5.2 Alter Table

Bir tablonun özelliklerini değiştirmek için kullanılan komuttur. Komutun yazılışı;

```
ALTER TABLE tablo_ismi
```

```
ADD[COLUMN] kolon_ismi kolon_türü
```

```
ALTER TABLE tablo_ismi
```

```
RENAME[COLUMN] kolon_ismi To yeni_kolon_ismi
```

```
ALTER TABLE tablo_ismi
```

```
RENAME To yeni_tablo_ismi
```

şeklindedir. Örneğin, numara isimli kolonun adını puan olarak değiştirmek istersek;

```
ALTER TABLE ogrenci RENAME COLUMN numara To puan;
```

komutu kullanılır.

4.5.3 Alter User

Kullanıcı hesap bilgilerinin güncellenmesi için kullanılan komuttur. Komutun yazılışı;

```
ALTER USER kullanıcı_ismi
```

```
[WITH PASSWORD şifre]
```

```
[CREATEDB | NOCREATEDB]
```

```
[CREATEUSER | NOCREATEUSER]
```

[IN GROUP grup_ismi[,...]][VALID UNTIL 'abstime']

şeklindedir. Aşağıda bu komutun kullanımına bir örnek gösterilmiştir:

Bir kullanıcı isminin geçerli olduğu süreyi değiştirmek için;

ALTER USER mert VALID UNTIL 'NOV 30 2005';

komutu kullanılır.

4.5.4 Begin Work

Bir işlemin başlamasını sağlar. Bir kullanıcının işlemlerini başlatmak için;

BEGIN WORK;

SQL komutu kullanılır.

4.5.5 Commit

Durdurulan işlemi devam ettirir. Durdurulan bir işlemi devam ettirmek ve iptal edilen değişiklikleri yinelemek için;

COMMIT WORK;

SQL komutu kullanılır.

4.5.6 Copy

Dosyalar ve tabolar arasında veri taşımak için kullanılır. Komutun yazılışı;

COPY [BINARY] tablo_ismi [WITH OIDS]

FROM 'dosya_ismi'stdin

[USING DELIMITERS 'ayırac']

COPY [BINARY] tablo_ismi [WITH OIDS]

To 'dosya_ismi'|stdout

[USING DELIMITERS 'ayırac']

şeklindedir. Komutun kullanımına ilişkin bir örnek aşağıda gösterilmektedir:

COPY ulke FROM '/users/proje/sql/ulke.txt';

Bu komut ile bir Unix dosyasında veriler bir tabloya koplanır.

4.5.7 Create Database

Yeni bir veritabanı oluşturmak için kullanılan komuttur. Yazılışı;

CREATE DATABASE isim [WITH LOCATION='erişim_yolu']

şeklindedir. Örneğin, '/users/db/tez' dizinde tez 1 isminde bir veritabanı oluşturmak için;

CREATE DATABASE tez1 WITH LOCATION = '/users/db/tez';

SQL komutu kullanılır.

4.5.8 Create Index

Bir ikincil indeks yapılandırmak için kullanılır. Komutun yazılışı;

CREATE [UNIQUE] INDEX indeks_ismi ON tablo_ismi [kolon_ismi]

şeklindedir.

4.5.9 Create Sequence

Bir sıralı sayı üreticisini yapılandırmak için kullanılır. Komutun yazılışı,

CREATE SEQUENCE isim

[INCREMENT artım]

[MINVALUE mindeğer]

[MAXVALUE maksdeğer]

[START başlangıç]

[CACHE geçici bellek][CYCLE]

şeklindedir.

4.5.10 Create Table

Yeni bir tablo yaratmak için kullanılır. Komutun yazılışı;

CREATE [TEMP] TABLE tablo_ismi(

kolon_ismi kolon_türü

[NULL|NOT NULL][UNIQUE][DEFAULT değer]

[PRIMARY KEY] [...]]

[,...]

[,PRIMARY KEY (kolon_ismi [...])]

[,CHECK (koşul)]

[,tablo_kısıt_deyimi]

[INHERITS (kalıtsal_tablo[,...])]

şeklindedir. Yaratılan tablonun sahibi tabloyu yaratan kullanıcı olur.

4.5.11 Create User

Yeni bir kullanıcı için hesap bilgilerinin oluşturulması için kullanılan komuttur.

Komutun yazılışı,

CREATE USER kullanıcı_ismi

[WITH PASSWORD şifre]

[CREATEDB-NOCREATEDB]

[CREATEUSER-NOCREATEUSER]

[IN GROUP grup_ismi [...]]

[VALID UNTIL 'abstime']

şeklindedir.

4.5.12 Delete

Bir tablodan veri silmek için kullanılan komuttur. Komutun yazılışı;

DELETE FROM tablo_ismi [WHERE koşul]

şeklindedir. Örneğin, ogrenci tablosundan ismi Ahmet olan öğrencileri silmek istersek;

DELETE FROM ogrenci WHERE isim='ahmet';

SQL komutu kullanılır.

4.5.13 Drop Database

Varolan bir veritabanını silmek için kullanılır. Komutun yazılışı;

DROP DATABASE isim

şeklindedir. Örneğin, dl1 isimli veritabanını silmek için;

DROP DATABASE dl1;

komutu kullanılır.

4.5.14 Drop Index

Varolan bir indeksi silmek için kullanılan komuttur. Komutun yazılışı;

DROP INDEX indeks_ismi

şeklindedir. Örneğin, başlık isimli indeksi silmek için;

DROP INDEX başlık;

SQL komutu kullanılır.

4.5.15 Drop Table

Varolan bir tabloyu silmek için kullanılan komuttur. Komutun yazılışı;

DROP TABLE tablo_ismi [...]

şeklindedir. Örneğin ders_kod ve ogrn_sin adlı tabloyu silmek için;

DROP TABLE ders_kod,ogrn_sin;

SQL komutu kullanılabilir.

4.5.16 Insert

Tabloya yeni satırlar eklemek için kullanılır. Komutun yazılışı;

INSERT INTO tablo_ismi [(kolon[,...])]

VALUES (anlatım [...])-SELECT sorgu

şeklindedir.

4.5.17 Select

Bir tablodan veri almak için kullanılan komuttur. Komutun yazılışı;

```
SELECT [ALL-DISTINCT [ON kolon]]
    anlatım [AS isim][,...]
    [INTO [TEMP][TABLE] yeni_tablo]
    [FROM tablo_ismi [alias][,...]]
    [WHERE koşul]
    [GROUP BY kolon [...]]
    [HAVING koşul [...]]
    [UNION [ALL]-INTERSECT-EXCEPT seçim]
    [ORDER BY kolon [ASC-DESC][,...]]
```

şeklindedir.

4.5.18 Update

Tablodaki verileri güncellemek için kullanılan komuttur. Komutun yazılışı;

```
UPDATE tablo_ismi SET kolon=anlatım[,...]
    [FROM liste]
    [WHERE koşul]
```

şeklindedir.

4.6 Postgres ile Gelen Bazı Uygulamalar

4.6.1 Createdb

Yeni bir veritabanı oluşturmak için kullanılan komuttur. Komutun yazılışı;

```
createdb [-h host][-p port][-D veridizini][-U][veritabanı_ismi]
```

şeklindedir. Parametreleri kısaca şu şekilde açıklanabilir:

“-h host” ile postmaster uygulamasının çalıştığı sistemin (sunucusunun) ismi verilebilir.

“-p port” ile postmaster uygulamasını kullandığı kapı (port)’nın numarası verilebilir.

“-U” ile işlem yapılmadan önce kullanıcı ismi ve şifresi sorulması istenebilir.

“-D veri dizini ” ile sistem dosyalarının yeri verilebilir.

“veritabanı_ismi” ile oluşturacak veritabanının ismi verilir.

Bazı örnek kullanımları aşağıda gösterilmektedir:

“uzeg” isimli bir veri tabanı oluşturmak için;

createdb uzeg;

“uzeg” ismindeki veritabanını “uzegsvr” isimli sistemin “5555” numaralı kapısını kullanarak oluşturmak için;

createdb-h uzegsvr -p 5555 uzeg

komutları kullanılabilir.

4.6.2 Createuser

Yeni bir Postgres kullanıcısı oluşturmak için kullanılan komuttur. Komutun yazılışı;

createuser [-h host][-p port][-i kullanıcı-id][-d|-D][-u|-U][kullanıcı_ismi]

şeklindedir. “-h host” ve “-p port” parametreleri createdb komutundaki işlevlerine sahiptirler.

Diğer parametreler kısaca şu şekilde açıklanabilir:

“-d” parametresi kullanılırsa oluşturulacak kullanıcıya veritabanı oluşturma hakkı verilir.

“-D” parametresi kullanılırsa oluşturulacak kullanıcıya veritabanı oluşturma hakkı verilmez.

“-i kullanıcı -id” parametresi kullanılırsa oluşturulacak kullanıcıya “kullanıcı-id” ile belirtilen kullanıcı belirteci (UID) verilir.

“-u” parametresi kullanılırsa oluşturulacak kullanıcıya başka kullanıcılar oluşturma hakkı verilir.

“-U” parametresi kullanılırsa oluşturulacak kullanıcıya başka kullanıcılar oluşturma hakkı

verilmez.

“kullanıcı-ismi” parametresi kullanılırsa oluşturulacak kullanıcıya “kullanıcı-ismi” ile belirtilen kullanıcı ismi verilir.

Örnek olarak, “proje” isminde ve her şeye yetkili bir kullanıcı oluşturmak için;

```
createuser -h uzegsvr -p5555 -d -u proje
```

komutu kullanılabilir.

4.6.3 Destroydb

Varolan bir veritabanını silmek için kullanılan komuttur. Komutun yazılışı;

```
destroydb [-h host] [-p port] [-i][veritabanı-ismi]
```

şeklindedir. “-h host” ve “-p port” parametreleri createdb komutundaki işlevlerine sahiptir. Diğer parametreler kısaca şu şekilde açıklanabilir:

“-i” parametresi kullanılırsa , kullanıcıdan onay alınmadan işlem yapılmaz.

“veritabanı-ismi” ile silinecek veritabanının ismi verilir.

Örneğin, “uzeg” isimli veritabanını silmek için;

```
destroydb -h uzegsvr -p 5555 uzeg
```

komutu kullanılabilir.

4.6.4 Destroyuser

Varolan bir postgres kullanıcıasını silmek için kullanılan komuttur. Komutun yazılışı;

```
destroyuser [-h host] [-p port][kullanıcı_ismi]
```

şeklindedir. “-h host” ve “-p port” parametreleri createuser komutundaki işlevlerine sahiptirler. Diğer parametre kısaca şu şekilde açıklanabilir:

“kullanıcı-ismi” ile silinecek kullanıcının ismi verilir. Örneğin, “proje” kullanıcıasını silmek için;

```
destroyuser -h uzegsvr -p 5555 proje
```

komutu kullanılabilir.

4.6.5 Psq1

Postgres'in etkileşimli istemcisidir. Komutun yazılışı;

```
psql -A[ -c sorgu][ -d veritabanı_ismi]
```

```
-e-E[ -f dosya_ismi][ -F ayıraç][ -h hostname]-Hln
```

```
[-o dosya_ismi][ -p port] - [qsSt][ -T tablo_seçenekleri]
```

```
-ux [veritabanı_ismi]
```

şeklindedir. “-h hostname” ve “-p port” parametreleri diğer komutlardaki işlevlerine sahiptirler. Diğer parametreler kısaca şu şekilde açıklanabilir:

“-A” parametresi kullanılırsa tablo elemanları ekrana yazılırken görüntünün daha düzgün çıkması için kullanılan boşlukların doldurulması yöntemi kullanılmaz.

“-c sorgu” parametresi kullanılırsa , psql sorgu ile verilen sorguyu çalıştırır ve çıkar. Genelde , kabuk programları ile kullanılır.

“-d veritabanı_ismi” ile bağlanılacak veritabanının ismi verilir.

“-e” parametresi kullanılırsa arkauca gönderilen sorgular ekrana yazılır.

“-E” parametresi kullanılırsa psql'in “d” gibi komutları için oluşturulan sorgular ekrana yazılır.

“-f dosya_ismi” parametresi ile psql'in kullanıcıdan girdi beklemeyeceği belirtilen dosyadan aldığı sorguları doğrudan çalıştıracak belirtilmiş olur.

“-F ayıraç” parametresi ile ayıraç olarak kullanılacak karakter belirtilmiş olur.

“-H” parametresi ile HTML 3.0 çıktısı aktifleştirilir.

“-l” parametresi ile kullanılabilecek veritabanı isimleri listelenir ve çıkarılır.

“-n” parametresi kullanılarak , önceden verilen komutlara ok tuşlarıyla ulaşımı sağlayan geçmiş komutlar (command history) yapısı için readline kitaplığının kullanılmayacağı belirtilmiş olur.

“-o dosya_ismi”parametresi ile bütün çıktıların yönlendirileceği dosyanın ismi verilir.

“-s” parametresi ile sorguların teker teker yapılacağı belirtilmiş olur.

Genelde “-c” parametresi ile birlikte kullanılır.

“-T tablo_seenekleri” parametresi ile HTML 3.0 uyumlu tablo komutları iin kullanılacak seenekler verilir.

“-u” parametresi kullanılırsa veritabanına baėlanmadan nce kullanıcı ismi ve parolası sorulur.

“-x” parametresi kullanılırsa kolon isimlerinin solda, verilen saėda yazılacaėı moda geilir.

psql baėladıktan sonra “\” ile baėlayan komutlar kullanılabilir. Bu komutlar aėaėıda listelenmiřtir:

“\a” ile, tablo elemanlarının listelenmesi sırasında olan hizalama seeneėi aktifleřtirilir.

“\c bařlık” ile, oluřturulacak HTML 3.0 tablolarının bařlıėı verilir.

“\connect veritabanı_ismi [username]” ile yeni bir veritabanına baėlanır.

“\copy veritabanı_ismi FROM To dosya_ismi” ile, veritabanı bir dosyaya aktarılır ya da bir dosyadan okunur.

“\d[tablo_ismi]” ile, tablo_ismi bilgisi verilmediėinde veritabanındaki tabloları listeler, aksi takdirde, tablo_ismi ile verilen tablonun iindeki alanları listeler.

“\df” ile, fonksiyonlar listelenir.

“\di” ile, indeksler listelenir.

“\do” ile, operatrler listelenir.

“\ds” ile, sistem tabloları ve indeksleri listelenir.

“\dt” ile, sistem tabloları haricindeki diėer tablolar listelenir.

“\dT” ile, veri trleri listelenir.

“\e[dosya_ismi]” ile, o anki sorgu emici belleėi (query buffer) biimlenir yada

“dosya_ismi” ile verilen dosya biimlemek zere aılır.

“\E[dosya_ismi]” ile, o anki sorgu emici belleėi biimlenir yada “dosya_ismi” ile, verilen dosya biimlemek zere aılır ve editr kapandıėında alıřtırılır.

“\f [ayıra]” ile, ayıra grntlenir yada “ ayıra “ olarak deėiřtirilir.

“\l” ile, sunucudaki veri tabanları listeler.

“\m” ile, tablo görünümü değiştirilir.

“\o [dosya_ismi || komut] ile, o andan sonraki çıktılar bir dosyaya yazılır yada bir Unix komutuna girdi olarak gönderilir.

“\lp” ile, o anki sorgu emici belleği ekrana yazılır.

“\lq” ile, psql’den çıkılır.

“\r” ile, sorgu emici belleği temizlenir.

“\w dosya_ismi” ile, o anki sorgu emici belleğinin çıktısı bir dosyaya yönlendirilir.

“\![komut]” ile, Unix kabuğuna dönülür yada bir Unix komutu çalıştırılır.

“\?” ile “\” ile başlayan komutlarla ilgili bilgi alınır.

5. PERL PROGRAMLAMA DİLİ

Perl (Practical Extraction and Report Language) 1986 yılında Larry Wall tarafından Unix işletim sistemi için geliştirilmiş bir programlama dilidir. Perl metin dosyalarını okuyabilen, bu dosyalardan bilgi çekişi yapabilen ve bu bilgiler üzerinden raporlar hazırlayabilen bir programlama dilidir. C++ ve Java programlama dillerinde olduğu gibi nesneye yönelik program yazmak da mümkündür.

Perl, Unix işletim sisteminde çok sık kullanılan sed, awk, shell C gibi bir çok programın özelliklerini bünyesinde barındırmaktadır. Perl'ün en büyük avantajlarından birisi çok çeşitli işletim sistemleri üzerinde kullanılıyor olmasıdır. Unix işletim sistemi üzerinde çalıştırılabilir.

5.1 Perl Programlama Dilinin Temelleri

5.1.1 Derleme ve Yorumlama İşlemleri

Perl, C++ ya da C'nin aksine yorumlanabilir bir dildir. Yani yazılan bir kod, BASIC dilinde olduğu gibi yorumlayıcı (interpreter) olarak isimlendirilen Perl programı tarafından okunur ve program çalıştırılır. Yorumlayıcı, Windows'dan bilinen exe uzantılı dosyalar oluşturmaz. Makine koduna çevrilen kodların aksine, yorumlayıcıların tek dezavantajı biraz daha yavaş olmalarıdır. Yazılan kod analiz edilmeden önce Perl programının çalışması ve program kodunu incelenmesi programın hızının düşürür. Makine koduna çevrilmiş kodlar bilgisayar işletim biriminin anlayacağı bir dilde olduğu ve tekrar derlenmek zorunda olmadığı için doğal olarak daha hızlıdır. Perl programlama dilinde yazılmış bir programın derleme ve yorumlama aşamalarından aynı anda geçilir. Program kodu hazırlandıktan sonra iki farklı yöntem ile sonuca ulaşılabilir. Birinci yol, Perl komutunun "perl dosya_adi" şeklinde komut satırından verilmesidir. İkinci yol ise, program dosyasına çalıştırma hakkı verilerek komut satırından "dosya_adi" kullanılarak çıktıya ulaşılmasıdır. Bu olanağı sağlayan program dosyasının ilk satırına yazılan `#!/usr/bin/perl` bilgisidir. Bu satır perl komutunun hem derleyicisi, hem de yorumlayıcı olarak çalışmasını sağlar. Bu satırın yapısı özeldir ve sözdiziminin hep bu şekilde olması gerekir. Ancak, her sistemde Perl derleyicisinin bulunduğu yer `/usr/bin/dizini` olmayabilir. Which Perl ya da where is Perl komutlarıyla Perl derleyicisinin sistemde bulunduğu yeri (yolu) öğrenmek mümkündür. `#!/usr/bin/perl` satırından sonra Perl kodu yazılmaya başlanır.

5.1.2 Standart Bilgi Giriş ve Çıkışları

Standart bilgi giriş ve çıkışları klavye ve ekran ile düzenlenmektedir. Programlama bilgi girişi STDIN ile de kısaltılan standart giriş dosyası ile yapılmaktadır.

```
$Value 1= <STDIN>;
```

<STDIN> ile klavye üzerinden girilen verilere ulaşılır. <STDIN> kullanılan bir program, klavye üzerinde herhangi bir string yada rakam girilmesini beklenir. Girilen veri \$value1 değişkenine yerleştirilir. Programın ilerleyen satırlarında \$value1 değişkenini kullanarak klavye üzerinden girilen veriyi işleyebiliriz.

```
$value 1= <>;
```

şeklindeki kullanım da geçerlidir.

5.1.3 Fonksiyonlar

Programda bir fonksiyon kullanıldığı zaman, Perl yorumlayıcısı bu fonksiyon ile ilgili kitaplık fonksiyonunu çağırır. Bir çok kitaplık fonksiyonu kullanıcıdan üzerinde işlem yapacağı bilgileri de ister. Bu bilgiye argüman denir. “()” parantez işaretleri içerisinde verilir.

Print (): Bu fonksiyonun görevi, verilen bilgiyi standart çıktı dosyasına göndermektir. Bunun sonucunda standart çıktı olarak ekran algılandığından istenilen bilginin ekranda görüntülenmesi sağlanabilmektedir.

```
#!/usr/bin/perl
```

```
$selam="merhaba";
```

```
print($selam);
```

Fonksiyonun argümanlarını yazarken “()” parantez işaretlerini kullanma zorunluluğu yoktur. Argümanlar istenilen sonuca göre tek tırnaklar veya çift tırnaklar arasında bu iki yöntemin özellikleri göz önüne alınarak da yazılabilir. Tek tırnaklı yapıda yorumlayıcı değişkenleri değerlendiremez. Değişkenin değerini yazdırabilmek için ya değişkenin ismi doğrudan yazılır yada çift tırnaklar arasında belirtilir. Ayrıca tek tırnaklı yazımda kaçış karakterleri de dikkate alınmaz. Bu karakterleri kullanabilmek için çift tırnaklar arasında yazımı yapılmalıdır.

```
#!/usr/bin/perl
```

```
$selam="merhaba";
```

```
print $selam,"\\n";
```

```
print "$selam",'2001','\\n';
```

Yukarıda verilen örnekte değişkenler, ekranda görüntülenecek olan anlatımlar ve kaçış karakterlerinin kurallara uygun olarak print() fonksiyonu ile kullanım şekilleri verilmektedir. Bu örneğin ekranda ki çıktısı ise aşağıdaki gibidir.

```
Merhaba
```

```
Merhaba 2001-04-28
```

Chomp (): Programa dışarıdan bir değer girilirken basılan “enter” tuşuna karşılık gelen yeni satır karakteri bilginin atandığı değişkene bilgi ile birlikte gelir. Bu durum bilginin doğru şekilde saklanmasını önler. Karakter topluluğun sonunda bulunan bu kaçış karakterini yok etmenin yolu chomp () fonksiyonunu kullanmaktır.

```
#!/usr/bin/perl
```

```
$bilgi=<STDIN>;
```

```
chomp($bilgi);
```

```
print $bilgi, "\\n";
```

Program yorumlandığı zaman bilgi isimli değişkene kullanıcı tarafından gönderilen değer sonunda bulunan kaçış karakteri verilen örnekte olduğu gibi silinir ve istenilen işlemler herhangi bir hataya yol açmadan yapılabilir. Eğer bu işlem yapılmazsa karakterler veya katarlar için ekrana görüntülemeye ve katar işlemleri sırasında, sayısal atamalarda ise sayısal işlemlerde hatalı sonuçlar elde edilir.

Chop(): Bu fonksiyon yukarıda tanıtilan Chomp() fonksiyonu ile benzer özellikler taşır. Fakat chop() fonksiyonu katarın son karakterini siler. Katarın son elemanının kaçış karakteri olup olmasının herhangi bir ayrıcalığı yoktur. Böylece katarların sonunda bulunan istenmeyen karakterler atılabilir.

```
#!/usr/bin/perl
```

```
$bilgi="Perl, C tabanlı bir programlama dilidir.";
```

```
chomp "$bilgi" ;
```

```
$bilgi1="Perl, C tabanlı bir programlama dilidir";
```

```
chomp "$bilgi" ;
$bilgi1="Perl, C tabanlı bir programlama dilidir";
chop ($bilgi1);
print $bilgi, "\n",$bilgi1, "\n";
```

Bu örneğin çıktısına bakılarak birbiriyle benzer özellikler taşıyan bu iki fonksiyon arasındaki tek farklılık görülebilir:

Perl, C tabanlı bir programlama dilidir.

Perl, C tabanlı bir programlama dilidir

5.1.4 Değişkenler

Bir değişken, daha sonra kullanılmak üzere bir değeri geçici olarak saklayan bellek bölgesidir. Perl’de üç tür değişken vardır. Bunlar;

- 1) Skaler değişkenler (scaler variables)
- 2) Diziler (arrays)
- 3) Özel tablolar (hashes)

Değişkenlere program içerisinde değer atama “=” işareti ile yapılır.

5.1.4.1 Skaler Değişkenler

Bir skaler değişken tek bir skaler değer içerir. Perl’de skaler değişkenler ön ek olarak bir “\$” işareti ile başlar. Örneğin; \$x,\$y,\$z,\$kullanıcı ve \$url hepsi birer geçerli skaler değişken isimleridirler. Değişken isimleri rakamla başlayamaz, boşluk ve standart dışı karakter içeremezler. Değişkenler bir harf karakter ile başlamak zorundadır. Daha sonraki karakterlerde rakam kullanılabilir. Skaler değişkenler en fazla 32 karakter uzunluğunda olabilir. Perl’de skaler değişkenler için sayısal, string ya da mantıksal diye bir ayırım söz konusu değildir. Diğer dillerin aksine Perl dilinde değişkenleri kullanmadan önce tanımlamak zorunluluğu yoktur. Değişken kullanılmak istenilen yerde tanımlanır ve program içerisinde kullanılabilir. Aynı değişkenin çeşitli satırlarda sayı ve string ihtiva etmesi mümkündür. Eğer yapılan işlem gerektiriyorsa, sayı içeren bir değişken stringe yada bir string sayı veri tipine çevrilebilir. Skaler değişken örnekleri aşağıda gösterilmektedir.

```
$isim= "Mert Bal";
```

```
$yas=27;
```

```
$mesaj1= "$isim şu anda $yas yaşındadır.";
```

```
$mesaj2= '$isim şu anda $yas yaşındadır.';
```

```
$pi=3.141592;
```

5.1.4.2 Diziler

Bir dizi değişkeni, birden çok değer içeren bir listedir. Bir skaler değişkende tek bir değer saklanırken bir dizi değişkende birden çok değer saklanabilir. Perl’de dizi değişkenler ön ek olarak bir “@” işareti ile başlar. Dizideki elemanlarla çalışırken tek tek erişebileceği gibi bir döngü değişkeni ile de ele alınabilir. Bir dizinin elemanlarını klavyeden kullanıcının girilmesini sağlamak amacıyla döngüler kullanılır. En çok kullanılan döngü for döngüsüdür. Aşağıda dizi örnekleri gösterilmektedir.

```
@EmptyArray=();
```

```
@NumberArray(12,014,oxoc,34.34,23.3E-3);
```

```
@StringArray=("This", "is", 'an', "array", 'of', "strings");
```

```
@MixedArray=("This", 30, "is", 'a', "mixed array", 'of', oxo8, "items");
```

Liste içinde bulunan elemanlar 0’dan başlayan indeksle adreslenir. Yani dizi indeksleri 0’dan başlar. Örneğin; @NumberArray dizisinin ilk elemanını belirtmek için @NumberArray [0] şeklinde bir rotasyon kullanılır. Eğer dizinin tek bir elemanı ile ilgilenilirse “@” işareti ile “\$” işareti kullanılır.

Perl’de dizilere eleman ekleme, eleman silme, iki diziyi birleştirmek, dizideki elemanları sıralarına kaydırmak vb. işlemleri gerçekleştirmek için bir takım hazır fonksiyonlar vardır. Bunlara dizi fonksiyonları denilmektedir. Aşağıda dizi fonksiyonlarına ait bir örnek gösterilmektedir:

```
@Renkler=("kırmızı", "beyaz", "bordo", "mavi");
```

```
$renk=pop(@Renkler); # dizideki son elemanın, "mavi"yi verir.
```

```
$renk=shift(@Renkler); # dizideki ilk elemanı "kırmızı"yı verir.
```

Bu örnekteki, dönen değer \$renk adlı değişkende tutulmuştur. Ancak doğrudan da kullanılması mümkündür. Her iki deyim pop ve shift, diziden ilgili elemanları silmek, yok etmek için kullanılır.

Diziye eleman eklemek için push fonksiyonu kullanılır:

```
push(@Renkler, "yeşil");
```

şeklindeki push fonksiyonunun kullanımı ile @Renkler dizisinin sonuna "yeşil" eklenir.

Diziler ile ilgili çok kullanılan bazı fonksiyonlar şunlardır:

```
sort (@Renkler );           #dizideki elemanları alfabetik olarak sıralar.
```

```
reverse (@Renkler);        #dizideki elemanları tersinden yer değiştirir.
```

```
$#Renkler;                 #Renkler dizisinin uzunluğunu verir. ( eleman _ sayısı -1 )
```

```
join ("," , @Renkler);    #dizideki elemanların aralarına virgül koyarak bir string olarak birleştirir.
```

5.1.4.3 Özel Tablolar

Özel tablolar, dizilerin özel bir hali olup her bir eleman, "anahtar_değer" şeklindeki ikililerden oluşur. Perl'de özel tablo isimleri % işareti ile başlar. Örneğin;

```
% webs=( "merih", "http://mertbal.yildiz.edu.tr/~merih",
```

```
        "tuncer", "http://mertbal.yildiz.edu.tr/~tuncer",
```

```
        "semih", "http://mertbal.yildiz.edu.tr/~semih");
```

Aynı tablo aşağıdaki gibi de tanımlanabilir:

```
% webs=( "merih"→"http://mertbal.yildiz.edu.tr/~merih",
```

```
        "tuncer"→"http://mertbal.yildiz.edu.tr/~tuncer",
```

```
        "semih"→"http://mertbal.yildiz.edu.tr/~semih");
```

Bu örnekte tablodaki her bir eleman, bir kullanıcı ismi ise ona ait web sayfasından oluşmaktadır. Tıpkı dizilerde olduğu gibi tablolarda da her bir elemanı bir skaler gibi düşünmek mümkündür. Örneğin, aşağıdaki deyim "merih" e ait URL 'yi ekrana yazacaktır.

```
print "$webs{'merih'}\n";
```

Özel tablolarla çalışırken sık kullanılan fonksiyonlar aşağıda verilmiştir:

`delete $tablo{$key}` # Belirtilen anahtar/değer ikilisinin tablodan siler ve silinen değeri geri döndürür.

`exist $tablo{$key}` # Belirtilen anahtar tabloda mevcut ise true döndürür.

`keys %tablo` # Tablonun anahtarlarını bir liste olarak verir.

`values %tablo` # Tablonun değerlerini bir liste olarak verir.

`scaler %tablo` #Tabloda bir eleman varsa, boş değilse true değerini verir.

5.1.5 If Deyimi ve Anlatımlar

If deyimi, Perl programlama dilinde koşullara bağlı deyim oluşturur. Böyle durumlarda koşul sağlandığı sürece yani, koşul doğru olduğu sürece istenilen işlemlerin program tarafından yapılması sağlanır. If deyimi iki bölümden oluşur. Birinci bölüm “()” parantez işaretleri arasında yazılan koşul anlatımı, ikinci bölüm ise; “{ }” küme işareti arasında yazılan deyim bloğudur. Aşağıda if deyimi ile ilgili bir örnek gösterilmektedir:

```
#!/usr/bin/perl

print ("Enter a number: \n");

$number=<STDIN>;

chop ($number) ;

if ($number) {

    print ("The number is not zero.\n");

}

print ("This is the last line of the program.\n");
```

Programın çıktısı şu şekilde olur:

Enter a number :

5

The number is not zero

This is the last line of the program.

Koşul anlatımı doğru olduğu sürece deyim bloğunu oluşturan deyimlerin geçerli olduğu durumudur.

If deyiminde bulunan koşulları oluşturan anlatımlar belirli kurallara göre bu blok kapsamında yer alabilir. Sayısal koşullar ile katar koşulları arasında yazım kuralları yönünden farklılıklar bulunmaktadır.

Sayısal koşul anlatımları aşağıdaki gibi yazılabilir:

$x > y \rightarrow x$ büyüktür y 'den

$x < y \rightarrow x$ küçüktür y 'den

$x \geq y \rightarrow x$ büyüktür veya eşittir y 'ye

$x \leq y \rightarrow x$ küçüktür veya eşittir y 'ye

$x = y \rightarrow x$ eşittir y 'ye

$x \neq y \rightarrow x$ eşit değildir y 'ye

Katarlar için olan koşul anlatımları ise aşağıdaki gibi kullanılır:

$x \text{ gt } y \rightarrow x, y$ 'den sonra gelir.

$x \text{ lt } y \rightarrow x, y$ 'den önce gelir.

$x \text{ ge } y \rightarrow x, y$ 'den sonra gelir veya y ile aynıdır.

$x \text{ le } y \rightarrow x, y$ 'den önce gelir veya y ile aynıdır.

$x \text{ eq } y \rightarrow x, y$ ile aynıdır.

$x \text{ ne } y \rightarrow x, y$ ile aynı değildir.

Katarlar arasındaki karşılaştırmalarda ASCII çizelgesinde hücre numaraları kullanılır. Katarlar bu sayılara göre sıralanır.

Koşul anlatımları kullanılarak if deyimi sayesinde program içerisinde denetimler rahatlıkla yapılabilir. Bunun yanında aynı if deyimi içerisinde birden fazla denetim noktası konulabilir. Bunu sağlayan, dallanmış if -else veya if-elsif- else deyim yapılarıdır. Aşağıda if-else deyimine ait bir örnek gösterilmektedir:

```
#!/usr/bin/perl

print ("Enter a number:\n");

$number1=<STDIN>;

chop ($number1);

print ("Enter another number:\n");

$number2=<STDIN>;

chop ($number2);

if ($number1==$number2){

print("The two numbers are equal.\n");

} else {

print("The two numbers are not equal.\n");

}

print ("This is the last line of the program.\n");
```

Programın çıktısı aşağıdaki gibidir:

Enter a number:

17

Enter another number:

18

The two numbers are not equal.

This is the last line of the program.

Bu programda dışarıdan girilen iki sayının birbirine eşit olup olmadığı kontrol etmektedir.

Program içerisinde ikiden fazla denetim yapmak istenildiğinde dallanmış bir deyim yapısı kullanmak gerekir. Bunu if –elsif-else yapısı sağlamaktadır. Bu yapıya ait bir örnek aşağıda gösterilmektedir:

```
#!/usr/bin/perl
```

```

print ("Enter a number:\n");

$number1=<STDIN>;

chop ($number1);

print ("Enter another number:\n");

$number2=<STDIN>;

chop ($number2);

if ($number1==$number2){

print("The two numbers are equal.\n");

} elseif ($number1==$number2+1){

print ("The first number is greater by one.\n");

} elseif ($number1+1==$number2){

print ("The second number is greater by one.\n");

} else {

print("The two numbers are not equal.\n");

}

print ("This is the last line of the program.\n");

```

Programın çıktısı aşağıdaki gibidir:

Enter a number:

17

Enter another number:

18

The second number is greater by one.

This is the last line of the program.

Yukarıdaki örnekte kullanıcı tarafından girilen iki sayı arasında yapılan denetimlerde

sağlanan koşullara göre program tarafından yapılması istenilen işlemler if-elsif-else yapısı içerisinde verilmektedir. Bu şekilde fazla denetim yapılan if deyimlerinde deyim yapılarının iç içe girmesini önlemek amacıyla küme parantezleri kullanılırken dikkat edilmesi gerekir.

5.1.6 Döngüler

Döngüler, bir değişken yada liste üzerinde operasyonları bazı şartlara bağlı olarak birden fazla yapmak için kullanılır. Döngüler kullanılırken çeşitli şartlar göz önünde bulundurulur. Kullanıcı şartların geçersiz ya da mantıksız olması durumunda sonsuz bir döngü içine girilir, bu da programın yapması gerekeni yerine getiremeyeceği anlamına gelir.

While döngüsü: While döngüsü verilen şart geçerli olduğu sürece blok içindeki komutları çalıştırır. Bu döngü iki kısımdan oluşur. Birinci kısım “()” parantez işaretleri arasında verilen koşul anlatımı, ikinci kısım “{ }” küme işaretleri arasında verilen deyimler bloğudur. Aşağıda while döngüsüne ait iki örnek verilmektedir:

```
#!/usr/bin/perl

print "Bir sayı giriniz:";

$sayi=<STDIN>;

chop ($sayi);

while ($sayi !=20){

print $sayi;}
```

Kullanıcının girdiği sayı 20'ye eşit olmadığında sayıyı yeniden ekrana yazan ve bunu while döngüsü ile denetleyen program yukarıda gösterilmektedir.

```
#!/usr/bin/perl

$done=0;

$count=1;

print("This line is printed before the loop starts.\n");

while ($done==0) {

print ("The value of count is $count \n");

if ($count==3) {
```

```

    $done=1;

}

$count=$count+1;

}

print ("End of loop.\n");

```

Bu program çıktısı aşağıdaki gibi olacaktır:

This line is printed before the loop starts.

The value of count is 1

The value of count is 2

The value of count is 3

End of loop.

For döngüsü: Bir döngüyü aynı anda üç bileşen ile denetleyebilen yapıya sahiptir. “()” parantez işaretleri arasında bu koşulları barındıran ve bu koşullara göre “{ }” küme işaretleri arasında bulunan deyimleri çalıştıran döngüdür. Koşulları belirlenen bir sayaç üzerinden işleme sokulur. For döngüsünün yapısı aşağıdaki gibidir:

```

For (başlangıç,test;artırım/azalım) {

    deyim 1

    deyim 2

    .....

}

```

İlk olarak başlangıç deyimi işletilir. Bundan sonra da test işletilir. Sonucu true ise döngü deyimleri işletilir. Ardından artırım/azalım kısmı işletilir. Test sonucu true olduğu müddetçe döngü tekrarlanır. Aşağıda for döngüsüne bir örnek gösterilmiştir:

```

#!/usr/bin/perl

print "Bir sayı giriniz.";

$sayi =<STDIN>;

```

```
chop($sayi);

for($i=0 ; $i<=$sayi; $i++) {

print$i; }
```

Buradaki örnekte, sıfırdan başlayarak kullanıcının klavyeden verdiği sayıya kadar döngü sayacının değerini ekrana yazan Perl program görülmektedir.

Foreach döngüsü: Bir dizideki her bir eleman yada liste benzeri yapılarla çalışırken (örneğin, metin dosyaları) her bir satırı tek tek ele almak için Perl’de foreach döngüsü kullanılır. Dizide kaç tane eleman varsa, her seferinde elemanın değeri döngü değişkenine aktarılır. Her seferinde döngü deyimleri icra edilir.

Aşağıda foreach döngüsüne örnekler gösterilmektedir.

Ekrana 1’den 10’a kadar rakamları yazar.

```
foreach $i(1..10)

{print $i;

}
```

Ekrana Array tipinde listenin elemanlarını yazar.

```
foreach $eleman(@Array)

{print $eleman;

}
```

Ekrana Hash tipinde listenin anahtarlarını yazar.

```
foreach $eleman(%HASH)

{print $eleman;

}
```

Aşağıda bir başka örnek gösterilmektedir:

```
#!/usr/bin/perl

@array=(1..5,5..10);
```

```
print (“@array\n”);

foreach(@array) {

    $_=”* *” if ($_==5)

}

print (“@array\n”);
```

Bu programın çıktısı aşağıdaki gibidir:

```
1 2 3 4 5 5 6 7 8 9 10
```

```
1 2 3 4 * * * * 6 7 8 9 10
```

5.1.7 Mantıksal Operatörler

If deyimi ve while döngüsünde yapılan denetimler tek koşul üzerinden yapılır. Aynı anda birden fazla denetim yaptırmak istenildiğinde mantıksal operatörlere gereksinim duyulur. Bu operatörler; “veya” ya karşılık gelen ||, ”ve” ye karşılık gelen &&, “değil” e karşılık gelen ! şeklindedir. Ve operatörünün doğru verisini geri verebilmesi için iki değişkeninde doğru olması gerekmektedir. Veya operatörünün doğru verisini geri verebilmek için en az bir değişkenin doğru olması gerekmektedir. Değil operatörü bir değişkenin değerini tersine çevirir. Doğru cinsinden bir veriyi yanlış, yanlış cinsinden bir veriyi doğru yapar. Aşağıda mantıksal operatörlere ilişkin bir örnek gösterilmektedir:

```
#!/usr/bin/perl

print “number”;

$number=<STDIN>;

chop ($number);

if ($number>=5 &&number<=10) {

print $number, “\n”;}

}
```

5.2 Dosya İşlemleri

Bilgilerin programa alınmasını ve program aracılığıyla saklanması bir diğer yolu da dosyalama işlemleridir. Bu işlemleri gerçekleştirebilmek için öncelikle standart veri giriş ve

çıkışlarına karşılık gelen “STDIN” ve “STDOUT” veri yollarına paralel veri yolları oluşturmak gerekir. Üzerinde işlem yapılacak her dosya için özel veri yolları tanımlanması gerekir. Bu tanımlama ele alınan dosyanın Perl programı içerisinde açılması sırasında yapılır. Bir dosya üzerinde çeşitli işlemler gerçekleştirebilmek için, bu dosyanın öncelikle açılması gerekir.

open (ANAHTAR, “dosya.ismi”);

İşlem yapılacak dosyanın Perl programında kullanıma açılması ve açılırken dosya ile program arasındaki bilgi alışverişinin hangi yolda olduğunun belirtilmesi yukarıdaki satır ile sağlanmaktadır.

Çift tırnaklar arasında verilen bilgi hangi dosyanın ne tür bir işlem yapılmak üzere açıldığını söylemektedir. Çift tırnaklar arasında yalnızca dosya isminin yazıldığı durumlarda dosyanın yalnızca okuma amaçlı açıldığı belirtilmektedir.

open (ANAHTAR, “<dosya.ismi”);

Yeni bir dosya açmak istenirse yukarıdaki ifade kullanılır. Eğer sistem üzerinde bu isimde bir dosya varsa, dosya silinir ve kullanılmak üzere açılır. Sistem üzerindeki bir dosya “>” operatörü ile silinebilir. Bu yüzden bu operatör kullanılırken dikkat etmek gerekir.

open (ANAHTAR, “>>dosya.ismi”);

Sistem üzerinde bir dosyanın sonunda ekleme yapılacaksa yukarıdaki ifade kullanılır. “>>” operatörü istenilen dosyayı açar ve verileri dosyanın sonuna ekler. Dosyalar open fonksiyonu ile açıldıktan sonra, program sonunda close komutuyla kapatılmalıdır. İşletim sistemi aynı anda sınırlı sayıda dosyayı açık tutabilir. Belirli sayıda dosya open komutuyla açıldıktan sonra, sistem diğer dosyaları açamaz hale gelebilir. Bunu engellemek ve sistemin düzgün bir şekilde çalışmasını sağlamak için close komutu kullanılmalıdır. Aşağıda close komutunun kullanımı gösterilmektedir:

close (ANAHTAR);

open ve close fonksiyonlarında kullanılan ANAHTAR değişkeni bir tür anahtar olarak düşünülebilir. Bir dosya açıldıktan sonra ANAHTAR değişkeni içine bu dosya üzerinde operasyon yapabilmek için gerekli anahtar yerleştirilir. ANAHTAR yerine istenilen bir isim kullanılabilir.

5.2.1 Dosyadan Okuma İşlemleri

Bir dosyadaki bilgileri bir Perl programına okutturabilmek için önce dosyanın okuma amaçlı açılması ve temel olarak dosyanın sonuna kadar bilgilerin satır satır okunabilmesi gerekir. Bu koşulları sağlayan örnek program aşağıda gösterilmektedir:

```
#!/usr/bin/perl

if (open (MYFILE,"file1")) {

$line =<MYFILE>;

while ($line ne "" ) {

print ($line);

$line =<MYFILE>;

}

} close (MYFILE);
```

Burada verilen örnekte hem dosyanın açılıp açılmadığı denetleniyor hem de dosya satır satır sonuna kadar okunarak elde edilen bilgiler ekrana yazdırılıyor.

5.2.2 Dosyaya Yazma İşlemleri

İşlem yapılacak olan dosya yazma veya bilgi ekleme amaçlı açıldıktan sonra print fonksiyonu kullanılarak yazma gerçekleştirilebilir. Aşağıdaki örnekte dosyaya yazma işlemi gerçekleştirilmektedir.

```
#!/usr/bin/perl

$line =20

open (MYFILE,">file1");

print MYFILE $line;

close (MYFILE);
```

Dosya yazma amaçlı açıldıktan sonra dosya değişkeni kullanılarak print fonksiyonu aracılığıyla "file1" isimli dosyaya istenilen yazdırılmaktadır. Bu örnek program çalıştırılırsa "file1" isimli dosyanın içeriği tek satırdan ve bu satırda yazılı olan 20 bilgisinden oluşur.

5.3 Regüler Terimler

Perl programlama dilinde regüler terim olarak isimlendiren harf ve kelime gruplarıyla bir metni taramak mümkündür.

5.3.1 Regüler Terimlerin Kullanılması

Regüler terimler bir dosyanın içinde geçen kelime ve kelime gruplarını bulmak için kullanılır. Bununla ilgili bir örnek aşağıda gösterilmektedir:

```
$var="bugün hava çok güzel.";

if ($var =~ /çok/) {

    print "çok kelimesi kullanılıyor\n";

}
```

\$var değişkeni bir string ifade ihtiva etmektedir. Burada if döngüsü kullanılarak çok kelimesinin string içinde kullanılıp kullanılmadığı kontrol edilmektedir. “/çok/” burada kullanılan regüler terimdir. Regüler terimler “/ /” işaretleri arasında yer alır. Eğer aranan regüler terim \$var stringi içinde yer alıyorsa, ekranda “çok kelimesi kullanılıyor” mesajı görüntülenir.

İçinde aranan regüler terimin bulunmadığı bir dosya taramak istendiğinde NOT(!) operatörü kullanılarak, regüler terimin string içinde bulunmaması gerektiği belirtilir. Aşağıdaki örnekte if döngüsü, string içinde aranan regüler terimin olmaması durumunda print komutunu çalıştıracaktır.

```
$var="Bugün hava çok güzel.";

if ($var !~ /kötü/) {

    print "kötü kelimesi kullanılıyor\n";

}
```

Tarama yapılırken regüler terime opsiyonlar verilebilir. Kullanılan opsiyona göre, regüler terimin tarama sürecinde davranışı değişik olacaktır.

Tarama yaparken büyük/küçük harf ayrımı yapılmak istenmiyorsa “i” opsiyonu kullanılır, Aşağıda bununla ilgili bir örnek gösterilmektedir:

```
$var="Bugün hava çok güzel.";

if ($var =~ /çok/) {

    print "Aradığınız kelime mevcut.\n";

}
```

Genelde regüler terim taramaları ilk satır üzerinde yapılır. Tüm satırlarda tarama yapmak için “m” opsiyonu kullanılır.

“S” opsiyonu kullanılarak, satır içinde bulunan regüler terimi başka bir kelimeyle değiştirmek mümkündür. “G” (global) opsiyonu ile cümle içinde birden fazla bulunan regüler terimi aynı anda değiştirmek mümkündür. “G” opsiyonu kullanılmadığı takdirde bulunan ilk regüler terim değiştirilir. Diğer uyan regüler terimler değiştirilmez.

Eğer string içinde bulunan bir kelimeyi silmek istersek “s” opsiyonu kullanılır.

5.3.2 Regüler Terimlerde Kullanılan Özel İşaretler

Kapsamlı arama yapmak için regüler terimlerde özel anlam taşıyan işaretler kullanılabilir. Bu işaretler aşağıda gösterilmiştir:

[abc]: a.b yada c

[a-z]: a’dan z’ye kadar küçük bir harf

[^a-f]: a’dan f’ye kadar harf grubu dışındaki herhangi bir küçük harf

\w: harfle başlayan herhangi bir kelime

\W: harfle başlamayan herhangi bir kelime

\s: iki kelime arasındaki boşluk

\d: 0 ile 9 arasında herhangi bir sayı

^: Satır başlangıcı

\$: Satır sonu

(abc|de): abc yada de içeren bir kelime (ikisinden birisi)

reg*: reg olarak aranan regüler terimin bir, birden fazla yada sıfır kere kullanılması

reg+: reg olarak aranan regüler terimin bir veya birden fazla kullanılması

reg?: reg olarak aranan regüler terimin sıfır yada bir kere kullanılması

Regüler terimler kullanılarak belli bir harfi ya da kelime içinde geçen herhangi bir rakamı lokalize etmek mümkün değildir. Regüler bir terim yerine harf sınıfı oluşturmak gerekir. Harf sınıfları oluşturarak belirli kelime, harf ve rakamları lokalize etmek mümkündür.

If(\$var=~/[xyz]/){print "OK";} #\$var içerisinde x,y yada z harfleri varsa OK yazar.

If(\$var=~/[Pp]erl/){print "OK";} #\$var içinde Perl yada perl kelimelerini lokalize eder.

If(\$var=~/[0.9]x/){print "OK";} #\$var içinde 1x2x.....9x gibi terimleri bulur.

“/[xyz]/” kalıbı kullanılarak, harf sınıfı oluşturulabilir. Burada dikkat edilmesi gereken nokta, köşeli parantezler içinde yer alan harf veya rakamların sadece bir harf yada rakam arama yapmak için kullanılabilir olmasıdır. Buna göre “/[abc]/” sadece içinde a ya da b ya da c olan kelimeleri bulacaktır. Eğer abc’yi ararsak /abc/ şeklinde yazılması gerekir:

If (\$var=~/[a-z- A-Z]/) {print "OK";}

Yukarıdaki örnekte, a’dan z’ye yada A’dan Z’ye kadar herhangi bir harften oluşan tüm kelimeleri bulur.

If (\$var=~/[^0-9]/) {print "OK";} #\$var içinde rakam yok.

Köşeli parantez içinde ^ operatörü kullanılarak, bir harf sınıfının cümle içinde aranmadığı belirtilebilir.

If (\$var=~/[^abc]/) {print "OK";} #\$var içinde a,b yada c yok.

^ operatörü kullanılarak, aramayı satır başında yapabilmek mümkündür. Burada ^ operatörü köşeli parantez dışındadır.

If (\$var=~ /^[0-9]/) {print "OK";} #\$var içindeki string bir rakamla başlıyor.

[0-9] yerine \d kullanılabilir.

If (\$var=~ /^[\\d]/) {print "OK";} #\$var içindeki string bir rakamla başlıyor.

Belirli bir harf yada rakam aranıyorsa “/-/” kalıbı kullanılır. Burada .(nokta) sadece bir harf ya da rakam yerini tutar.

If (\$var=~ /. /) # En az bir harf / rakamdan oluşan kelimeleri bulur.

If (\$var=~/.{5,}/) # En az beş harf / rakamdan oluşan kelimeleri bulur.

If (\$var=~/[A-Z]{2,}/) # A'dan Z'ye büyük bir harfi iki harf / rakam takip eder.

If (\$var=~/[A-Z][0-9]/) Büyük bir harf ve bir rakam arasında herhangi bir harf/rakam olan kelimeleri bulur. Ax1, B23, Cy9 kelimelerini bulur.

Alternatifleri kullanılarak tarama sırasında birden fazla regüler terim kullanılabilir. Listede bulunan terimlerden biri bulunana kadar tarama işlemi sürdürülür.

If (\$var=~/mert|merih| tuncer/) {print"OK"; # \$var içinde adı geçen herhangi bir ismi arar.

Kivantör (quantor) olarak isimlendirilen *,?,+ işaretleri kullanılarak bir kelime içinde arka arkaya kullanılan harf veya rakam grupları lokalize edilebilir.

* operatörü ile bir harf/rakamı sıfır, bir yada birden fazla lokalize edebilmek mümkündür.

If(\$var=~/[a-z]{0,}/) # bilumum strigleri bulur. Boş olan stringleri de bulur.

If(\$var=~/[a-z]{0,}[0-9]{1,}/) # sıfır, bir yada birden fazla küçük harften sonra bir rakam ihtiva eden tüm kelimeleri bulur. Örneğin; a1,aaaaa3,bbbb5,....

+ operatörü ile bir harf/rakamı en az bir yada daha fazla büyük harften sonra bir rakam içeren kelimeleri bulur. Örneğin; A3,B4,....

? operatörü ile bir harf/rakamı sıfır ya da en fazla bir defa lokalize etmek mümkündür.

If(\$var=~/[a?]/) # \$var içinde A harfi hiç yok yada sadece bir tane A var.

If(\$var=~/[A-Z]?/) # \$var içinde ya büyük harf yok ya da iki tane büyük harf yan yana değil.

5.3.3 Split () Fonksiyonu

Split fonksiyonu, bir string'i belirtilen karaktere göre ayrıştırarak parçaları bir diziye yerleştirir. Fonksiyonun birinci parametresi bir regüler terimdir. İkinci parametresi de ayrıştırılacak string'tir. Eğer ikinci parametre belirtilmezse "\$_" değişkeni üzerinde işlem yapar.

Split fonksiyonunun kullanımı şöyledir.

```
$bilgi ="Mert.Bal:1974:İstanbul:Matematik Mühendisi";
```

```
@personel = split (/,,$bilgi);
```

deyimleri sonucunda @personel dizisinin aldığı değerler aşağıdaki gibi olur. Burada alan ayırıcı olarak : (iki nokta üst üste) karakteri kullanılmıştır.

Eğer \$_ değişkenindeki bilgileri ayrıştırmak istersek split fonksiyonu aşağıdaki gibi kullanılır:

```
@karakterler=split(/,$kelime);
```

```
@kelimeler=split(/ /,$cümle);
```

```
@cumleler=split(/\./,$paragraf);
```

Aşağıda toplam kelime sayısını bulan bir program gösterilmektedir:

```
#!/usr/bin/perl

$wordcount=0;

$line=<STDIN>;

while ($line ne "" ) {

    chop($line);

    @array=split ( / /,$line);

    $wordcount +=@array ;

    $line =<STDIN>;

}

print ("Total number of words : $wordcount \n");
```

Program çalıştırılırsa çıktısı aşağıdaki gibi olur:

Here is some input.

Here are some more words.

Here is my last line.

^D

Total number of words : 14

5.4 Altprogramlar

Altprogramlar, programı daha küçük, yönetimi kolay modüllere bölmek amacıyla kullanılır. Perl programlama dilinde altprogram yazarak belirli kod parçaları programın istenilen bir yerinde kullanılabilir. Altprogram kullanma, kod karmaşıklığını ortadan kaldırma olanağı sağlar. Yazılan kod düzenli bir yapıya kavuşur ve daha da önemlisi yazılan kod programın çeşitli yerlerinde altprogramın ismi yazılarak kullanılır.

Perl’de bir altprogramı çağırmak için alternatif bulunmaktadır. Bunlar aşağıdaki gibidir:

`&subname`

`&subname(args)`

`subname`

`subname(args)`

`subname( )`

Burada args, alt programa değer göndermek için kullanılan değişkenlerdir. Genel olarak bir alt programın yapısı şöyledir:

```
sub altprogramadı {
    kodlar.....
    return($değer); veya exit (değer);
}
```

my özel fonksiyonu, değişkenin geçerlilik alanını altprogram çerçevesinde sınırlandırmak için kullanılabilirler.

Perl’de altprogram olarak yazılmış matematik ile ilgili fonksiyonlar da bulunmaktadır.

Bunlar; `sqrt ( )`, `abs( )`, `sin( )`, `cos( )` ve `log( )` şeklinde sıralanabilir.

Aşağıda altprogramlar ile ilgili örnekler gösterilmektedir:

```
#!/usr/bin/perl
```

```
&topla;
```

```
sub topla {
```

```
print"iki sayıyı giriniz";

$sayı1=<STDIN>;

chop($sayı1);

$sayı2=<STDIN>;

chop($sayı2);

$toplam=$sayı1+$sayı2;

print $toplam;

}
```

Yukarıda, tüm işlemleri topla isimli bir altprogramda yapan ve programın ana kısmında bu alt programın çağrıldığı bir Perl program örneği verilmiştir. Bu örneğin ikinci satırında altprogram "&" işareti kullanılarak çağrılmaktadır.

```
#!/usr/bin/perl

print topla( );

sub topla {

my ($sayı,$toplam);

print"bir sayı giriniz";

$sayı1=<STDIN>;

chop($sayı1);

$toplam=$sayı+10;

return $toplam;

}
```

Yukarıda, yalnızca altprograma ait olan ve programın başka bir yerinde gerekli olmayan değişkenlerin tanımlanmasına ve altprogram içerisinde bulunan bir sonucun ana programa return deyimi ile gönderilmesine ait örnek verilmiştir. Burada altprogram içerisinde bulunan sonuç return deyimi ile ana programa gönderilir ve ana programda fonksiyon ikinci satırdaki gibi doğrudan ismi verilerek kullanılır.

```
#!/usr/bin/perl

print " iki sayı giriniz";

$sayı1=<STDIN>;

chop ($sayı1);

$sayı2=<STDIN>;

chop ($sayı2);

&topla ($sayı1+ $sayı2);

sub topla {

my($sayı1,$sayı2)=@_;

my ($toplama);

$toplama=$sayı1+ $sayı2);

print $toplama; }
```

Yukarıda verilen örnekte bir altprograma ana programdan değer gönderme bulunmaktadır. Altprogram bu değerleri programın dokuzuncu satırında bulunan my deyimi ve bu deyimin eşitlendiği “@_” işaretleri sayesinde alabilmektedir. Bu satırda altprograma hangi değerlerin argüman olarak gönderileceği belirtilmektedir.

```
#!/usr/bin/perl

$total = 0;

&getnumbers;

foreach $number (@numbers) {

    $total+=$number;

}

print ("The total is $total \n");

sub get numbers {

    $line=<STDIN>;
```

```
$line=~s/^\s+|\s*\n$//g;
```

```
@numbers=split (/s+/, $line);
```

```
}
```

Yukarıda klavyeden girilen sayıların toplamı bulunan bir Perl programı gösterilmektedir. Bu programın çıktısı aşağıdaki gibi olabilir:

```
11 10 25 4 30 20
```

```
The total is 100
```



6. HTML PROGRAMLAMA

HTML (HyperText Markup Language), Internet uygulamaları ile kullanıcılar arasında bir arayüz olan, web sayfalarının hazırlanmasında araç görevi gören ve yaygın olarak kullanılan bir programlama dilidir. Hareketli görüntülerin, ses özelliklerinin ve etkileşimli uygulamaların kullanılabildiği ortamlar HTML sayesinde kullanılır. Web sayfalarının tasarımında kullanılabilen bu özelliklerin etkin hale geldiği yapıyı oluşturur. HTML ile başlıkların ve metinlerin stillerini (koyu, alt çizgili, eğik) belirlemek mümkün olabilmektedir. HTML'in font tipinin belirlenebileceği eklentileri de mevcuttur.

6.1 HTML Programlamaya Giriş

Her HTML dokümanı, <HTML> anlamlandırma etiketi ile başlamalı ve </HTML> anlamlandırma etiketi ile sona ermelidir. Başlangıç etiketi <HTML>, inceleyiciye dokümanın HTML formatlı bir doküman olduğunu belirtir ve dokümanın başlangıcını işaret eder. Bitiş etiketi </HTML>, dokümanın sonuna işaret eder ve herhangi bir HTML dokümanında son öğedir. Her HTML dokümanının bir başlık ve bir gövdesi bulunmalıdır. Başlık, ilk <HTML> etiketini izler. HTML başlığı, dokümanın başlığı gibi anahtar görüntüleri belirler. Başlığın başlangıcı, <HEAD> etiketi (başlık başlangıç etiketi) ile belirtilir. Başlığın sonu, </HEAD> (başlık bitiş etiketi) ile belirtilir. Dokümanın başlığı takip eden ana bölümüne gövde denir. Gövde okuyucunun inceleyicisinin görüntülenmesi istenilen metin ve nesneleri içerir. Başlık gibi, gövde de başlangıç etiketi <BODY> ve bitiş etiketi </BODY> ile sınırlanır. Başlık bölümü öncelikle hizmet birimlerine doküman hakkındaki bilgileri sağlamak için kullanılır. Başlık bölümündeki her şey başlangıç ve bitiş başlık etiketi arasında yer alır. Aşağıdaki HTML etiketleri başlık için ayrılmıştır.

<TITLE> Doküman başlığı.

<BASE> Doküman için temel URL'yi tanımlar.

<ISINDEX> İnceleyicinin metinsel arama mekanizmasına dönmek için bir yol sağlar.

<LINK> Dokümanın diğer dokümanlarla olan ilişkilerini tanımlar.

<META> Doküman hakkında genel bilgi veya meta bilgi tanımlama imkanı verir.

<NEXTID> Oluşturulacak bir sonraki tanımlayıcı tanımlanır.

En çok kullanılan başlık etiketi <TITLE> etiketidir. Doküman başlığı etiketi <TITLE>,

doküman başlığının başlangıcını gösterir. Bitiş doküman başlığı etiketi </TITLE>, doküman başlığının sonunu gösterilebildiği için başlıklar kısa fakat açıklayıcı olabilir. Kısıtlı sayıda karakter gösterilebildiği için başlıklar kısa fakat açıklayıcı olmalıdır. Genel bir kural olarak başlık uzunluğu 65 veya 65 karakterden az olmalıdır. Başlık, biçimlendirme veya anlamlandırma içeremez. Başlık sadece ASCII karakterleri içermelidir. HTML, <H1>'den <H6>'ya kadar altı düzeyde başlık yaratma imkanı verir. Diğer bir çok etiket gibi başlık etiketleri de çiftler halinde kullanılır. Örneğin, birinci düzey başlık başlangıç etiketi <H1>, birinci düzey başlık bitiş etiketi </H1> ile birlikte kullanılır. Genellikle, birinci düzey başlıklar en büyük yazı karakteriyle, altıncı düzey başlıklar en küçük yazı karakteriyle yazılırlar. İnceleyiciler başlıktan önce ve sonra birer boşluk koyar. Başlıklar aşağıdaki şekilde yaratılabilir:

<H1>Birinci Düzey Başlık</H1>

<H2>İkinci Düzey Başlık</H2>

<H3>Üçüncü Düzey Başlık</H3>

<H4>Dördüncü Düzey Başlık</H4>

<H5>Beşinci Düzey Başlık</H5>

<H6>Altıncı Düzey Başlık</H6>

HTML'de dokümanı paragraflara ayırmak için paragraf <P> kullanılır. İnceleyici paragraf etiketini gördüğünde, o andaki satırı sonlandırır ve paragraf etiketini takip eden metni veya resmi görüntülemeyen önce boş bir satır bırakır. HTML standardında bir paragraf sonu etiketi tanımlandığı halde, </P> etiketini kullanmak seçime bağlıdır. Satır sonu etiketi
, satırlar arasına boşluk koymadan bir satırı sonlandırmayı sağlar. Dökümana yatay cetvel eklemek için <HR> etiketi kullanılır. <HR> etiketi dokümanı görsel olarak bölümlere ayırmaktadır.

6.2 Bağlantılar

Bulunulan HTML sayfasından bir başka sayfaya bağlantı kurabilmek amacıyla <A> etiketi kullanılır.

“HREF” parametresiyle birlikte başka bir dokümana veya sabitlemeye bağlantı yaratır.

“NAME” parametresiyle bağlanabilecek bir sabitleme yaratır.

HREF =”...” Bu dokümana bağlanabilecek doküman URL’si.

NAME =”....” Sabitlemenin adı.

Buraya tıklayınız!

Yukarıda verilen örnekte o anda ekranda görüntülenen HTML sayfasından bir başka HTML veya eşdeğer sayfaya geçişi sağlayan etiketin kullanımı bulunmaktadır.

6.3 Metin Hizalama

Bir metin içerisindeki satırları, paragrafları ve blokları listelemek veya belli bir düzene sokmak amacıyla <DIV> etiketi ile birlikte “align” parametresi kullanılır. <DIV> etiketi üzerinde işlem yapılacak metin parçasını diğer bölümlerden ayırır. Hizalamayı dokümandaki bütün paragraflara uygulamak için BODY elemanının sonlanmasından önce ayırma bitiş etiketi </DIV>’i yerleştirmek gerekir. “align” parametresi ile birlikte kullanılan “right, left, center, justify” seçenekleri ise ayrılan metnin hangi biçimde görüntüleneceğini belirler. Aşağıda bu özellikleri içeren bir HTML program parçası görülmektedir.

<DIV ALIGN= “center”>Yıldız Teknik Üniversitesi</DIV>

HTML’de listeleme üç şekilde yapılmaktadır. Bunlardan birincisi numaralandırma yaparak oluşturulan listelerdir. Numaralı listeler aynı zamanda sıralanmış listeler olarak da adlandırılırlar. Bu liste için, ayrılmış HTML etiketi, Ordered List (sıralanmış liste)’nin kısaltılması olan ’dir. Tanım listesi etiketi çift olarak kullanılırlar, başlangıç etiketi , bitiş etiketi ’dir. Sıralanmış listedeki her elemandan önce etiketi kullanılır. Sıralanmış listedeki her eleman, numaralandırılmış veya harflendirilmiştir. Harfler sadece iç içe listelerde kullanılır. Bir inceleyici, liste başlangıç etiketi ’yi gördüğünde aşağıdakilerini yapar:

- 1) Yeni bir satıra atlar.
- 2) Liste elemanının metnini içerden başlatır.
- 3) Liste elemanının önüne uygun harfi veya numarayı koyar.

İç içe liste, bir listenin içinde yer alan başka bir listedir. HTML ‘de, iç içe listeleri bir liste yaratmak için gerekli bütün yapıyı, yürürlükteki listenin içine yerleştirerek elde etmek mümkündür. Örneğin, numaralı liste yapısının içine noktalı listeler yerleştirilebilir. İkinci yöntem noktalı listelerdir. Noktalı listeler, belli bir sırası olmayan amaçları, konuları ve

fikirleri listelemek için kullanılırlar. Bu tip listeye ait HTML etiketi etiketidir, UL Unordered List (düzenlenmemiş liste) kelimelerinin baş harflerini temsil eder. Noktalı liste etiketleri çiftler halinde kullanılırlar. Noktalı liste başlangıç etiketi ; bitiş etiketi 'dir. Listedeki maddeler için etiketi kullanılır. etiketi başlangıç ve bitiş etiketi ile kullanılabileceği halde, bitiş etiketi genelde gerekli olmamaktadır.

Son yöntem ise tanım listeleridir. Küçük sözcükler, tanım listeleri olarak da adlandırılırlar. Bu liste için ayrılmış HTML etiketi, Definition List (tanım listesi)'nin kısaltılması olan <DL>'dir. Tanım listesi etiketleri çift olarak kullanılırlar. Başlangıç etiketi <DL>, bitiş etiketi </DL>'dir. Tanım listesindeki her elemanın iki ögesi vardır:

1) Tanım başlığı adı verilen bir anahtar kelime

2) Tanım verisi adı verilen bir anahtar kelime

Tanım başlığı etiketi <DT>, küçük sözcük terimini veya kullanıcının tanımladığı anahtar kelimeyi belirler. Tanım verisi etiketi <DD>, küçük sözcük terimiyle veya anahtar kelimeyle ilgili tanımı belirler. Eğer terim birden fazla tanıma sahipse birden fazla tanım verisi etiketi kullanılabilir.

6.4 Metin Biçimlendirme ve Karakter Türü Denetimi

Görüntülenecek metnin biçimi, kullanılacak karakterlerin büyüklüğü, çeşidi ve rengi HTML etiketlerinin yardımıyla denetlenebilir. Bir dokümanı yaratırken, bazı metin bölümlerinin onu çevreleyen metinden daha büyük olması istenirse <BIG> etiketi kullanılır. Metin normalden küçük yazılmak istenirse <SMALL> etiketi kullanılır. <SMALL> etiketi ile vurgulanan metin, normal paragraf metninden daha küçük görüntülenecektir. <BIG> ve <SMALL> etiketleri çiftler halinde kullanılır.

<BIG>Büyük yazılacak metin.</BIG>

<SMALL>Küçük yazılacak metin.</SMALL>

Üstü çizili metinler oluşturulmak isteniyorsa <S> etiketi kullanılır. <S> etiketi de çiftler halinde kullanılır. Üstü çizili metinler, inceleyiciler tarafından yazı boyunca üzerine bir çizgi çekilmiş olarak gösterilirler. Üstü çizili metinler aşağıdaki şekilde tanımlanabilir:

<S>Üstü çizili metin.</S>

Fiziksel stiller, inceleyiciye göstereceği formatı anlatır. HTML'de koyu, yatay, altı çizili ve eş

genişlikli olmak üzere toplam dört fiziksel stil vardır. Her fiziksel stilin bir başlangıç, bir bitiş etiketi vardır. Kullanılacak karakterler etiketi ile koyu renkte, <I> etiketi ile italik biçiminde <U> etiketi ile altı çizilmiş olarak basılabilir.

<TT> etiketi ile eş genişlikli olarak basılır.

Koyu font

<I>İtalik font</I>

<U>Altı çizili font</U>

<TT>Daktilo fontu veya eş genişlikli font</TT>

HTML dosyasındaki yazım şeklini bırakılan boşluklarda göz önüne alınarak metnin aynen görüntülenmesi amacıyla <PRE> etiketi kullanılır. Belirtilen bu etiketlerin başlangıç ve bitiş noktaları arasında belirtilen karakter veya karakter toplulukları etiketin ne olduğuna bağlı olarak şekil alırlar.

Ayrıca kullanılan karakterlerin büyüklükleri ve renkleri de etiketi ile belirlenir. Font büyüklüğünü 1 ve 7 arasındaki değerleri kullanarak doğrudan belirlemek mümkündür.

Yıldız Teknik Üniversitesi

Yukarıda verilen örnekte etiketi ile belirlenen bölüm bu etiketin içinde verilen karakter büyüklüğü ve rengine bağlı olarak web sayfasında görüntülenecektir. Karakterin rengi saptanırken renk isimlerinin yerine bu renklere karşılık gelen iki basamaklı ondalık sayı düzenindeki değerleri “rgb” (kırmızı, yeşil, mavi) renk standardında kullanılabilir.

6.5 HTML Formları

Bu bölüme kadar HTML hakkında sözü edilen temel konular web sayfalarında kullanıcılara bilgi vermeye yönelik uygulamalardır. HTML aynı zamanda bunun tersi olan kullanıcıdan bilgi almaya yönelik özellikler taşıyan uygulamalara da sahiptir. HTML formları, kullanıcılardan bilgi almaya yönelik özellikler taşıyan uygulamalara da sahiptir. HTML formları, kullanıcılardan bilgi almaya yönelik alanları içeren Web sayfalarının bir parçasıdır. Bir HTML dokümanı içerisinde birden fazla form kullanılabilir. HTML’de iç içe formlar oluşturulamaz. Her form, <FORM> etiketi ile başlayan “method” ve “action” isminde iki parametre ile kullanılabilen ve </FORM> etiketi ile sonlandırılan yapıdan oluşur. “Method” parametresi formu geri postalama yöntemini belirler. “Method” parametresi iki seçeneğin

şekillendirilmesine olanak tanır. Bu seçenekler post ve get seçenekleridir. Seçeneklerden post yöntemi, form aracılığıyla gönderilen bilgiyi doküman olarak döndürür. İkinci seçenek olan get yöntemi ise, formlardan gelen bilgileri URL'nin bir uzantısı olarak gönderildiğinden, web sunucusunun log dosyalarına bakılarak bu tür bilgilerin tamamının herhangi bir kullanıcı tarafından öğrenilmesi mümkündür. Get metodu, en çok bir veri tabanından basit sorgulamalar yaparak çeşitli bilgileri listelemek amacıyla kullanılır. Gizli ve özel bilgiler gönderilirken post metodu kullanılmalıdır. Post metodu ayrıca verileri kriptolama özelliğine sahiptir. Ayrıca uzun ve karmaşık veriler göndermek için post metodu daha uygundur.

“Action” parametresi bir form gönderilirken gerçekleştirilecek hareketi belirler. Bir form action parametresi olmaksızın hiçbir şekilde işlenemez bu nedenle formların bu parametreyi taşıması zoruludur.

```
<HTML>
```

```
<BODY>
```

```
<FORM METHOD = "POST" ACTION = "form1.cgi">
```

```
<PRE> isim: <INPUT TYPE = "text" NAME = "isim"> </PRE>
```

```
<INPUT TYPE = "checkbox" NAME = "secim" VALUE = "d">Dogru <BR>
```

```
<INPUT TYPE = "checkbox" NAME = "secim" VALUE = "y">Yanlış <BR>
```

```
<SELECT NAME = "konu">
```

```
<OPTION>Perl<OPTION>Postgresql</SELECT>
```

```
<P>Yorum:<TEXTAREA NAME = "yorum" ROWS = 4 COLS = 20></TEXTAREA>
```

```
<INPUT TYPE = "submit" VALUE = "tamam">
```

```
<INPUT TYPE = "reset" VALUE = "temizle">
```

```
</FORM>
```

```
</BODY>
```

```
</HTML>
```

Bir HTML formunda kullanıcıdan bilgi alınan alanlar için <INPUT> etiketi kullanılır. Bu etiket içerisinde kullanılan “type”, “name” ve “value” önemli parametrelerdir. “Type”

parametresi kullanıcıdan alınan bilgilerin türünü belirler. Metin, tamsayı, şifre türündeki bilgiler bu parametre yardımıyla tanımlandığı gibi, çoktan seçmeli durumlar ile karşılaşıldığında da bu parametrenin seçeneklerinden (checkbox) yararlanır. Ayrıca formdaki bilgilerin kullanıcı tarafından onaylanması veya gözden çıkarılması da yine bu parametre aracılığıyla sağlanır.

“Name” parametresi Perl, CGI, HTML etkileşimli programlar açısından oldukça önemli bir yere sahiptir. Bu parametrenin karşılığında verilen isim aslında kullanıcıdan form aracılığıyla alınan bilginin atandığı değişkendir ve bu değişkeni HTML’in yanı sıra Perl’de kullanılabilir. Çoktan seçmeli durumlar için verilen bilgilerin aynı değişkene atanabilmesi amacıyla “value” parametresi kullanılır. Bu yapılara yukarıda verilen örnek program içerisinde rastlanabilir. Bu program içerisinde en son anlatımı yapılan “value” parametresi için örnek vermek gerekirse, programın dördüncü ve beşinci satırlarını oluşturan çoktan seçmeli yapılarda eğer seçim “Doğru” şeklinde yapılmış ise seçim isimli değişkene “d”, “Yanlış” şeklinde yapılmış ise “y” değeri atanır. Çoktan seçmeli durumlarda izlenebilecek bir ikinci yol ise, <SELECT> etiketinin kullanılmasıdır. Bu etiket ile seçenekler menülü bir yapıda sıralanır ve kullanıcı bu listeden istediği seçimi yapabilir. Bu seçenekler <OPTION> etiketi aracılığı ile verilir. Form içerisinde kullanıcılar için uzun metinler yazabilecekleri alanlar da oluşturulabilir. Yine yukarıdaki örnekte verildiği gibi bu durumlarda <TEXTAREA> etiketinden yararlanılabilir. Bu etikette, oluşturulacak alanın görüntülenecek maksimum satır ve sütun sayıları da verilebilir. Bu özellikler, “rows” ve “cols” parametreleri kullanılarak katılabilir.

6.6 Resim ve Renk Özellikleri

Web sayfalarının hazırlanması sırasında göze hitap edecek özelliklerin varlığı HTML programları içerisinde “jpg” ve “gif” gibi biçimlere sahip dosyaların kullanılmasına bağlıdır. Bu biçimler görsel özellikler taşıdıklarından tasarım konusunda çok fazla kullanılırlar. Bu dosyaları HTML sayfalarında görüntülemenin kullanım amacına göre iki yolu vardır. Web sayfalarında yalnızca görüntü amaçlı basmak için etiketi ile diğer HTML sayfaları ile bağlantı kurmak için <A HREF> etiketi bulunmaktadır. etiketi, dokümanlarda metinle birlikte görüntü de kullanılabilmeye olarak sağlar. etiketi üç parametreye sahiptir ve sonlanma elemanı yoktur. etiketinin kullanılması gereken tek parametresi “SRC” dir. Bu şekilde sözü edilen resime ait, kaynak, yol ve isim belirlenir.

Ekranda bir “gif” biçimli resim dosyası görüntülemek için yukarıda en basit özellikleri ile verilmiş olan satırı HTML programı içerisine yerleştirmek yeterlidir. Ayrıca bağlantı aracı olarak bu tür biçimler kullanılmak istendiği zaman aşağıdaki gibi bir satırın program içerisine yazılması gerekmektedir.

```
<A HREF =”sinav.html”><IMG SRC=”yildiz.gif”></A>
```

Sayfaları hazırlanan resimler konulacağı gibi metnin veya arka planın renkleri üzerinde ayarlamalar da yapılarak tasarıma ek özellikler katabilir. Bunlara ek olarak sayfaların arka planları olarak resim dosyaları da yüklenebilir.

```
<BODY BGCOLOR =”aqua” TEXT =”blue”>
```

```
<BODY BGGROUND =”yildiz.gif”>
```

HTML sayfalarının arka planlarının, metnin renginin veya bağlantıların renklerinin ne şekilde olacağına <BODY> etiketi içerisinde yukarıda belirtilen yazım şekli ile karar verilebilir.

6.7 Tablolar

Web sayfalarının tasarımında kullanılan en etkin araçlardan birisi tablolardır. Tablolar, metin ve/veya resimleri belirli bir satır ve sütun düzeni içerisine koyar. Tabloların etkin hale getirilmesini <TABLE> etiketi sağlar. <TABLE> etiketinin bir çok parametresi kullanılarak, sayfada tabloların nerede ve nasıl görüntüleneceği tanımlanabilir. <TABLE> etiketi başlangıç ve bitiş etiketi ile birlikte kullanılır. Başlangıç tablo etiketi <TABLE>’ın geçerli bazı parametreleri aşağıdaki gibidir:

“CELLPADDING” ve “CELLSPACING” parametreleri: Tablonun okunabilirliğini arttırmak için “CELLPADDING” ve “CELLSPACING” parametreleri kullanılır. Veri hücreleri içine boşluk yerleştirmek için “CELLPADDING” parametresi kullanılır. Hücre içindeki ve arasındaki boşluklar yürürlükteki birim cinsinden genellikle piksel cinsinden belirlenir.

“WIDTH” parametresi: “WIDTH” parametresi, bir tablo için bağlı veya sabit genişlik belirlemek amacıyla kullanılır. Önceden belirlenmiş “WIDTH” değeri , yürürlükteki pencere genişliğidir. Genişlik, birimler cinsinden veya yürürlükteki pencere genişliğinin bir yüzdesi cinsinden tanımlanmış olur. Bu sayısal değerden sonra bir yüzde işareti yoksa bu sayı yürürlükteki birim, genellikle piksel cinsinden yorumlanır.

Aşağıdaki tablonun büyüklüğü, yürürlükteki pencere genişliğinin yüzde 25’idir:

<TABLE WIDTH =25%>

“Cols” parametresi: “COLS” parametresi, bir tablodaki sütun sayısını belirlemek için kullanılır. Dört sütunlu bir tabloyu aşağıdaki gibi belirleyebiliriz:

<TABLE COLS =4>

“BORDER” ve “FRAME” parametreleri: HTML’de tablo yaratmak, genellikle iki adımdan oluşan bir işlemdir. Tablonun çevresindeki, çerçevenin genişliğini, “BORDER” parametresi kullanılarak belirlemek mümkündür. Tablo çevresine bir çerçeve yerleştirmek için “FRAME” parametresi kullanılabilir. Tablolar önceden belirlenmiş olarak çerçeveye sahip değildirler. Tablonun çerçeveleme tipi aşağıdaki değerler kullanılarak belirlenebilir.

FRAME = NONE Tablonu çevresinde çerçeve yoktur (default)

FRAME = TOP Tablonun sadece üst kısmına çerçeve konur.

FRAME = BOTTOM Tablonun sadece alt kısmına çerçeve konur.

FRAME = TOPBOT Tablonun alt ve üst kısmına çerçeve konur.

FRAME = SIDES Tablonun sağ ve sol kenarlarına çerçeve konur.

FRAME = ALL Tablonun dört kenarına da çerçeve konur.

FRAME = BORDER Tablonun dört kenarına da çerçeve konur.

(ALL ile aynı görevi görmektedir.)

Bütün tablolar bir veya daha fazla veri satırı içermek zorundadır. Satırlar bir tablonun başlık, gövde veya alt bilgi bölümünde tanımlanır. <TR> etiketi kullanılarak, tablonun içerdiği her bölüm için bir satır tanımlanabilir. Bu yüzden bir başlık, iki gövde ve bir alt bilgi bölümünden oluşan bir tablo en az dört satıra sahip olacaktır. Tablolarda sadece başlangıç <TR> etiketi kullanılır.

Hücre düzeyinde yapılan atamalar, bağımsız veri kümeleri veya başlıklar içindir. Veri hücreleri, tabloda görüntülenecek sayıları ve ifadeleri içerirler. <TD> etiketi kullanılarak veri hücrelerini tanımlamak mümkündür. Başlık hücreleri, bölüm, sütun ve satır başlıkları içerir. <TH> etiketi kullanılarak başlık hücreleri tanımlanabilir.

<TABLE BORDER=0 WIDTH ="100%" CELLPADDING=0>

<TR> <TD WIDTH ="50%">ÜLKE</TD>

```
<TD WIDTH =50%>ŞEHİR</TD></TR>
```

```
<TR><TD WIDTH ="50%">Türkiye</TD>
```

```
<TD WIDTH ="50%">İstanbul</TD></TR><TABLE>
```

Bu örnekte toplam iki satır ve bu satırların ikiye bölünmesinden ortaya çıkan iki sütundan oluşan bir tablo bulunmaktadır. Bu biçimlendirme sayesinde sayfa üzerinde yan yana veya alt alta gelmesi gereken satır ve sütunlar düzenlenebilmektedir. Ayrıca her bir sütun için tablonun kaçta kaçının ayrılacağı , tablonun sınırları gibi özellikler “BORDER” ve “WIDTH” parametreleri ile belirlenebilmektedir.



7. CGI PROGRAMLAMA

CGI (Common Gateway Interface) programları, Web sunucusu üzerinde çalışan dış kaynaklı programlardır. Web sunucusu üzerinde kullanılan yabancı kaynaklı programların standartları ve ortak ağ geçit ara yüzünün diğer uygulamalarla etkileşiminin nasıl olması gerektiği belirlenmiştir. Ortak, ağ geçit ve ara yüz sözcükleri bu standartların anahtar sözcükleridir ve nasıl çalıştığını tanımlar.

CGI; programın erişilebilir olmasını sağlamak için ortak bir yol tanımlayarak, her hangi bir kimsenin kullandığı ortam ne olursa olsun bir CGI programına bilgi girişi sağlar.

CGI; program, sunucusu ve diğer uygulamalar arasında bir bağlantı ya da ağ geçit tanımlayarak dış kaynaklı programları genelleştirmiş girişler olarak kabul edilmesini ve diğer uygulamalara bilgi aktarabilmesini mümkün kılar.

CGI; kullanıcıların dış kaynaklı programlara erişebilmesi için bir ara yüz yada bir yol tanımlayarak dış kaynaklı programlar için gerçekleştirilen karmaşık ara yüz işlemlerini bir kaç basit prosedüre indirger.

Programlar yürütülürken arka planda gerçekleşen olaylar ise şöyledir:

- 1) Okuyucunun inceleyicisi girişleri Web sunucusuna aktarır.
- 2) Sunucu sistem sırası geldiğinde girişleri CGI programına aktarır.
- 3) CGI programı girişi işler ve eğer başka bir uygulamaya gerekiyorsa uygulamayı programa geçirir ve çıkışı web sunucusuna gönderir.
- 4) Web sunucusu okuyucunun cevabını onun inceleyicisine geri yollar. Bir CGI programının çıkışı, bir veritabanı sorgulamasının sonucundan, okuyucunun cevabına göre yaratılmış yeni bir dokümana kadar her şey olabilir.

Unix Sistemlerinde CGI programları usr dosya sistemi içinde cgi_bin olarak adlandırılan kataloqlara yerleştirilmiştir ve CGI uygulamaları da yine usr dosya sistemi içinde cgi_scr olarak adlandırılan kataloqlara yerleştirilmiştir. CGI programları, ağ geçit programcıları (gateway scripts) olarak adlandırılır. Programcıklar terimi shell programcığının arkasındaki Unix ortamından gelmektedir, ancak ağ geçit programcıları bir programcık formatında olmak zorunda değildir. Ağ geçit programcılarını çalıştırılabilir bir dosya yaratabilen herhangi bir programlama dilinde yazabilmek mümkündür. En çok kullanılan diller;

Perl, C/C++, C Shell, Bourne Shell, Tcl, Python, Visual Basic

şeklinde sıralanabilir.

7.1 CGI Programlarının Çalışması

Ağ geçit programcıları, web yayınlarının okuyucuları tarafından geri yollanan girişlerin kullanılması için kullanılır. Giriş genellikle web sunucusu tarafından ağ geçit programcısına aktarılan ortam değişkenleri biçimindedir. Ortam değişkenleri sunucunun kullandığı CGI programlarının sürüm bilgisi, veri tipi, veri boyutu gibi aktarılmakta olan bilgi hakkında bazı değişkenleri tanımlar. Ağ geçit programcığı girişleri bir komut satırı ifadesinden ve standart bir girişten de alabilir. Bir CGI programını çalıştırmak için sunucunun, bir ağ geçit programcığını çalıştıracak ve kullanıcının onu kullanmayı planladığı amaca uygun olarak konfigüre etmesini sağlayacak yetenekte olması gerekmektedir. Okuyucu, programcığa bir referans içeren bağlantıyı etkinleştirerek CGI programına bilgi aktarır. Ağ geçit programcığı girişi işler ve sonucu web sunucusunun kullanabileceği şekilde biçimlendirir. Web sunucusu sonucu alır ve okuyucunun inceleyicisine geri gönderir. İnceleyici çıkışı ekranda görüntüler.

Bir ağ geçit programcığı çıkışı sunucuya bir yönlendirme taşıyan bir başlıkla başlar. Şu anda geçerli olan üç sunucu yönlendirmesi; Content-Types (içerik tipi), Location (konum) ve Status (durum)'dur. Başlık, HTML başlığı gibi boş bir satırla izlenecek şekilde bir yönlendirme taşıyabilir. Boş satırlar başlığı okuyucuya aktarılan veriden ayırır. Location ve Status yönlendirmesi içeren çıkışlar genellikle tek satırdır. Bu nedenle sunucunun ihtiyacı olan Location ve Status bilgilerinin hepsi bu satırdadır ve veri içeren satırlar bulunmadığından boş satıra da ihtiyaç yoktur. Sunucu çıkışı yorumlar, ortam değişkenlerini belirler ve çıkışı istemciye iletir. Bu nedenle sunucu ve istemci arasında gerçekleşen olaylar, bir çok aşama içerir. Bunlar aşağıdaki gibi sıralanır:

- 1) İstemci, girişi sunucuya geçirir.
- 2) Sunucu, girişe uygun olacak şekilde ortam değişkenlerini ayarlar.
- 3) Sunucu, girişi CGI programının adı ile bir değişken gibi geçirir.
- 4) Sunucu, eğer varsa komut satırı girişini ya da standart giriş katarını CGI programına geçirir.
- 5) Program girişi işler.

6) Program, çıkışı sunucuya döndürür. Bu çıkış daima belirli bir başlığı taşır ve eğer ek veri varsa, bir de gövdesi bulunur.

7) Sunucu, ortam değişkenlerini çıkışa uygun olarak ayarlar.

8) Sunucu, çıkışı istemciye geçirir.

7.2 Cgi-lib.pl Kitaplığı

CGI uygulamalarının Perl programlama dili ile oluşturulmasında kullanılan ve kolaylıklar sağlayan bazı alt programlar içeren kitaplıktır. Bu kitaplık içerisinde bulunan önemli bir yere sahip olan iki tane alt program Read Parse ve PrintHeader alt programları mevcuttur.

ReadParse isimli alt program, sunucu tarafından yayınlanan formlar aracılığıyla Internet kullanıcıları tarafından gönderilen bilgilerin Perl ile geliştirilmiş CGI uygulamaları sayesinde program içerisindeki değişkenlere alınmasını sağlayan yapıya destek verir.

PrintHeader isimli alt program ise CGI uygulamasının HTML özellikleri taşıdığını programa anlatır. Internet üzerinde etkin olan CGI uygulamaları bu ortamı kullanmak için HTML formlarından yararlanır. Bu yüzden bu alt program Perl için önemlidir.

7.3 Perl Destekli CGI Uygulamaları

CGI uygulamalarının devreye girebilmesi için HTML ile biçimlendirilmiş formlar aracılığıyla bilginin alınması gerekmektedir. Dolayısıyla, bir uygulama için temel olarak iki dosya gerekmektedir. Birincisi Internet üzerinden bilgilerin alınmasını sağlayan formların bulunduğu HTML veya HTML biçiminde hazırlanmış CGI dosyasıdır. İkincisi ise, bu dosyadan gelen bilgilerin değişkenlere atanarak istenilen şekilde işlenebilmesini sağlayan CGI dosyasıdır. Formlardan gelen bilgiler arka planda Perl tarafından yeğlenmeye bağlı olarak dosyalama işlemleri ile, bir veritabanı yönetim sistemi kullanılarak saklanır veya hemen işleme sokularak sonuçlar yine Internet üzerinden verilebilir. Aşağıda önce basit bir HTML formuna örnek olan bir dosya ve daha sonra da bu bilgilere karşılık gelen CGI uygulamasına örnek olan bir dosyanın içeriği verilmektedir.

```
<HTML><HEAD><TITLE>UZEG</TITLE></HEAD>
```

```
<BODY><CENTER><B>KAYIT ISLEMLERİ</B></CENTER>
```

```
<FORM METHOD ="POST" ACTION ="kayit.cgi">
```

```

<TABLE BORDER =0 WIDTH ="100%" CELLPADDING ="0">

<TR><TD WIDTH ="18%">ISIM:</TD>

<TD WIDTH ="82%"><INPUT TYPE ="text" NAME ="isim"></TD></TR>

<TR><TD WIDTH ="18%">SOYAD:</TD>

<TD WIDTH ="82%"><INPUT TYPE ="text" NAME ="soyad"></TD></TR>

<TR><TD WIDTH ="18%">NUMARA:</TD>

<TD WIDTH ="82%"><INPUT TYPE ="text" NAME ="no"></TD></TR>

</TABLE>

<INPUT TYPE ="Submit" VALUE ="tamam">

<INPUT TYPE ="reset" VALUE ="iptal">

</FORM></BODY></HTML>

```

Bu HTML örneğinde bir form aracılığıyla kullanıcıdan bilginin nasıl alındığı görülmektedir. Input etiketi içinde bulunan name seçeneğinin karşılığında yazılmış olan değişken ismine, formlarda o bölüme kullanıcı tarafından gönderilen bilgiler depolanır. Bu değişken aşağıda örnek olarak verilen CGI uygulamasında Perl tarafından kullanılmaktadır. Bu HTML dosyası aracılığıyla alınan bilgilerin hangi CGI dosyası ile işleneceğini bildiren satır ise form etiketinin kullanıldığı satırdır.

Kullanılan bu form ile alınan bilgilerin işlendiği örnek CGI dosyası aşağıdadır:

```

#!/usr/bin/perl

require "/var/lib/httpd/cgi_bin/cgi_lib.pl";

print &printHeader;

&ReadParse;

$isim=$in{'isim'};

$soyad=$in{'soyad'};

$no=$in{'no'};

print '<BR>', $isim, " ", $soyad, " ", $no , '</BR>';

```

Verilen CGI uygulamasında ikinci satırda CGI ve Perl ile ilgili olan `cgi_lib.pl` kitaplığı işler hale getirilmektedir. Üçüncü ve dördüncü satırlarda bu kitaplığın birer özelliği olan ve daha önce söz edilen alt programlar devreye girmektedir. Bu satırları izleyen üç satırda ise, HTML dosyasındaki form aracılığıyla alınan ve yine bu form içerisinde belirtilen değişkenlerde saklanan kullanıcı bilgileri arka plandaki Perl destekli CGI uygulaması ile saklanabilmekte, istenilen işlemler yapılabilmektedir. Bu örnek CGI uygulaması, tepki olarak aldığı bilgileri yeniden HTML özelliklerini kullanarak sunucu aracılığıyla gezginde görüntülenmektedir. Bu örneğin son satırında verilmiş olan `print` fonksiyonu HTML etiketlerinin Perl programlama dili içerisinde nasıl kullanıldıklarını vurgulamak açısından önemlidir. Perl uygulamaları içerisinde ekranda HTML biçiminde görüntüler oluşturma gereksiniminin olduğu açıktır. Formlar aracılığıyla alınan bilgilerin işlenip ortaya çıkan sonuçların yeniden Internet ortamına geri gönderilmesi Perl-HTML ilişkisinin gerekliliğini ortaya çıkarmaktadır. HTML etiketleri, Perl destekli CGI uygulamaları içerisinde `print` fonksiyonu aracılığıyla kullanılır. Bu fonksiyona tek tırnaklar arasında argüman olarak gönderilen HTML etiketleri sayesinde Perl programları HTML dilinin özelliklerini de kendi içine alır.

7.4 Bir Perl ve Postgresql Destekli CGI Uygulaması

Aşağıda bir HTML formundan alınan öğrenci bilgilerini (ad, soyad, doğum tarihi, baba adı, fakülte, bölüm, numara) veritabanına aktaran ve veritabanında oluşan bilgi kümesinde belirli bir bölüme göre öğrencilerin ad listesini hazırlayacak bir sorgulama yapan, sonuçları görsel yöre sunucusunda yayınlayan küçük bir uygulama verilmektedir.

```
#!/usr/bin/perl

require "/var/lib/httpd/cgi_bin/cgi_lib.pl";

print&PrintHeader;

&ReadParse;

use Pg;

$ad=$in {'AD'};

$soyad=$in {'SOYAD'};

$dogum_yeri=$in {'DOGUM YERI'};

$dogum_tarihi=$in {'DOGUM TARIHI'};
```

```

$baba adi=$in {'BABA ADI'};

$fakulte=$in {'FAKULTE'};

$bolum=$in {'BOLUM'};

$conn=Pg::setdb($pghost, ' ', ' ', ' ', $dbname);

$pghost=$conn->host;

$dbname=$conn->db;

$conn=Pg::connectdb("dbname =kayit");

$veri="INSERT INTO ogrnbil VALUES (\'$ad\',\'$soyad\',\'$dogum yeri\',\'$dogum
tarihi\',\'$baba adi\',\'$fakulte\',\'$bolum\',\'$numara\')";

$result=$conn->exec($veri);

$sara=" SELECT ad FROM ogrnbil WHERE bolum ='Matematik Muhendisligi'";

$son=$conn->exec ($sara);

$kayitsayisi=$son->n tuples;

for($i=0;$i<$kayitsayisi;$i++) {

$veri [$i]=$son->getvalue ($i,0);

print $veri [$i],"\n";}

```

HTML formundan alınan öğrenci bilgileri CGI özellikleri sayesinde Perl değişkenlerine aktarılmaktadır. İstenilen veritabanı ile bağlantı kurulduktan sonra veri tabanında daha önceden oluşturulmuş olan bir tabloya bilgiler aktarılmaktadır. Veri tabanında bulunan bilgileri öğrenmek üzere select komutunun kullanılması ile birlikte istenilen sorgulama yaptırılabilir. Son olarak ise, elde edilen bilgiler print () fonksiyonu yardımıyla görsel yöre sunucusunda görüntülenebilmektedir.

8. SONUÇLAR ve ÖNERİLER

Uzaktan eğitim, son yıllarda giderek artan bir kitleye hitap etmektedir. Uzaktan eğitimde kullanılan çeşitli yöntemler bulunmaktadır. Bu yöntemlerin herbirinin eğitmen, öğrenci ve kurum açısından avantaj ve dezavantajları bulunmaktadır. Yönlendirme teknolojileri olarak adlandırılan bu yöntemlerin herbirinin kurulumu, gereksinimleri ve masrafları birbirinden farklı olmaktadır. Ortaya çıkan gereksinimler doğrultusunda oluşturulacak yapılarda gözönünde bulundurulması gereken insana, organizasyon türüne ve ekonomiye bağlı etkenler bulunmaktadır. Kurum içerisinde kullanılan uzaktan eğitim programı için ideal eğitim teknolojisi organizasyonun ihtiyacına bağlıdır. Uzaktan eğitimde eğitmen ile öğrenci arasındaki etkileşim çok önemlidir. Dersin anlatım tarzının eğitmenin yeteneklerini kısıtlamaması, öğrencileri uzaklaştırmaması, süreklilik sağlaması gibi konularda dikkatli davranılması uzaktan eğitim sistemini başarıya götürebilir. Bu koşulları sağlamak amacıyla dersin anlatım hızına dikkat edilmesi, monotonluğun engellenmesi, ses ve görüntü desteği ile eğitimin ilgi çekici olmasının sağlanması sistemin özelliklerine katılabilir. Ayrıca eğitimin verildiği süreç içerisinde öğrencinin durumunu denetim altında tutabilecek çalışma soruları, ödevler, projeler, kısa sınavlar gibi başarı ölçen uygulamalarında sisteme katılması öğrencinin verilen eğitime daha fazla ve etkin katılmasını sağlayabilir.

Eğitmenin fiziksel olarak öğrenciler ile aynı ortamda bulunmaması etkileşimin hiç olmadığı anlamına gelmemelidir. Böyle bir sistemde eğitmen ile öğrenciler arasında birebir etkileşim olması özelliğinin bulunması gerekmektedir. Bu etkileşim kullanılan teknolojiye göre video tele eğitim türündeki kurslarda birebir soru sorma, İnternet tabanlı sistemlerde ise elektronik posta (e-posta) yolu ile soru sorma şeklini almaktadır. Ayrıca sıkça sorulan sorular bölümünde bu konuda etkin olarak hizmet verebilmektedir.

Uzaktan eğitim çerçevesinde destek verilecek olan derslerin veya programların seçilmesi sırasında dikkat edilmesi gereken bir başka öğe ise eğitime katılacak olan kişilerin eğitmen ile çok fazla etkileşime ihtiyaç duymayacakları ve fiziksel olarak bir laboratuvar uygulanmasına gereksinim olmayacağı bir ortamın yaratılmasıdır. Uzaktan eğitim kapsamında verilen ders desteği ve sertifika programlarının ortak özellikleri, kursların İnternet üzerinden sahip olunan bilgisayarda gerekli donanım ve yazılım sağlandığı sürece, takip edilebilmesi, görüntü ve ses ile desteklenmesi, öğrencinin İnternet olanaklarını kullanarak sorularını yöneltebilmesi, istenilen çalışmaları ve projeleri gerçekleştirmesi şeklinde sıralanabilir.

KAYNAKLAR

- Acar,Ö.(2001),Perl-Cgi,Pusula Yayıncılık ve İletişim Ltd.,İstanbul.
- Bateman,D. ve Simpson, V. (1995),Hypertext in the Classroom: Changing the Roles of Teacher and Students,In F. Percival,R.Land ve D.Edgar-Nevill (Eds.), Computer Assisted and Open Access Education. London: Kogan Page Ltd.
- Belanger,F. ve Jordan, D.H.(2000), Evaluation and Implementation of Distance Learning: Technologies,Tools and Techniques,Idea Group Publishing,London.
- Clark,R.E. (1994), Assessment of Distance Learning Technology.In E.L.Baker veF.O.Harold (Eds.), Technology Assessment in Education and Training, Hillsdale, NJ:Erlbaum.
- Çetin,G.(2000),Linux İşletim Sistemi Red Hat 6.2,Seçkin Yayıncılık,Ankara.
- Daft,R.L. ve Lengel R.H.(1986), A Proposed Integration Among Organizational Requirements, Media Richness and Structural Design. Management Science,32,554-571.
- Darby,J.(1992). Computers in Teaching and Learning in U.K.Higher Education. Computers and Education,19(1/2),1-8.
- Dobson,R. (1995), The Case for Flexible Learning.In F.Percival,R. Land ve D.Edgar-Nevill(Eds.),Computer Assisted and Open Access Education, London:Kogan Page Ltd.
- Encarta 98 Encyclopedia, Microsoft Corporation.1993-1997.
- Gardner, H.(1993), Frames of Mind: The Theory of Multiple Intelligences. New York: Basic Books.
- Gubernick, L. ve Ebeling, A.(1997), I Got My Degree Through E-Mail.Forbes,159,84-92.
- Güneş,A.(2000),Perl ile Cgi Programlama Klavuzu,Alfa Yayınevi,İstanbul.
- Kulik,J.M.(1994), Meta-Analytic Findings on Computer-Based Instruction. In E.L. Baker ve F.O. Harold. (Eds.). Technology Assessment in Education and Training.Hillsdale: Erlbaum.
- La Rose,R.,Gregg,J. ve Eastin,M.(1998), Audiographic Telecourses for the Web: An Experiment Journal of Computer Mediated Communication ,4(2).
- Lockhart,T.(1998),Postgresql User's Guide,Postgres Global Development Group,Berkeley.
- Mc Cormack,C. ve Jones,D.(1997), Building A Web-Based Education System. New York: John Wiley & Sons.
- Özkan,Y.(1996),Unix İşletim Sistemi,Alfa Yayınevi,İstanbul.
- Race, P. ve Brown, S. (1995), Getting the Word Right .In F. Percival, R.Land ve D. Edgar-Nevill.(Eds.). Computer Assisted and Open Access Education. London :Kogan Page Ltd.
- Scotney,B. ve Mc, Clean, S.(1995), The Design and Use of Computer Based Tutorials for Teaching. Learning and Assessment of Quantitative Research Methods,80-86.
- Stanton,N. ve Stammers,R.(1990), A Comparison of Structured and Unstructured Navigation Through a CBT Package. Computers and Education,15(1-3),159-163.

- Till,D.(1997),Teach Yourself Perl 5 for Windows NT 21 Days,Sams Publishing,Indianapolis.
- Tunga,M.A.(1999),Sanal Ortamda Eğitim Görselyöre Sunucusunda Kayıt İzleme ve Sınav Program Paketi,Yüksek Lisans Tezi, İTÜ Fen Bilimleri Enstitüsü (yayımlanmamış).
- Uysal,M.(2000),Linux İşletim Sistemi,Beta Yayınevi,İstanbul.
- Uysal,M.(2000),Sql Veritabanı Sorgulama Dili,Beta Yayınevi,İstanbul.
- Whalen,T. ve Wright,D. (1998), Distance Training in the Virtual Workplace.In M. Igbaria, M.Tan.(Eds.). The Virtual Workplace.Hershey,PA:Idea Group Publishing.
- Yurt,B.,Ötkünç,B.ve Karaçayır, A.(2000),Html-Java-Cgi-Vrml-Sgml Unleashed,Sistem Yayıncılık,İstanbul.

ÖZGEÇMİŞ

Doğum tarihi	04.04.1974	
Doğum yeri	İstanbul	
Lise	1988-1991	Şehremini Lisesi
Lisans	1994-1999	Yıldız Teknik Üniversitesi Kimya Metalurji Fak. Matematik Mühendisliği Bölümü
Yüksek Lisans	1999-2001	Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Matematik Mühendisliği Anabilim Dalı

Çalıştığı kurum

1999-Devam ediyor YTÜ Fen Bilimleri Enstitüsü Araştırma Görevlisi

