

**TÜRKİYE CUMHURİYETİ**  
**YILDIZ TEKNİK ÜNİVERSİTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



**PROJE KİTABI HAZIRLAMA ESASLARI VE LATEX**  
**TASLAĞI KULLANIMI**

19011001 – Ahmet ELBİR  
19011002 – İbrahim Onur SİĞİRCİ

**BİLGİSAYAR PROJESİ**

Danışman  
Dr. Öğr. Üyesi Hamza Osman İLHAN

Haziran, 2020



## TEŞEKKÜR

---

Yıldız Technical University is one of the seven government universities situated in İstanbul besides being the 3rd oldest university of Turkey with its history dating back to 1911. It is regarded as one of the best universities in the country as well. Our university has 10 Faculties, 2 Institutes, the Vocational School of Higher Education, the Vocational School for National Palaces and Historical Buildings, the Vocational School for Foreign Languages and more than 25,000 students.

The Istanbul State Engineering and Architectural Academy and affiliated schools of engineering and the related faculties and departments of the Kocaeli State Engineering and Architecture Academy and the Kocaeli Vocational School were merged to form Yıldız University with decree law no.41 dated 20 June 1982 and Law no. 2809 dated 30 March 1982 which accepted the decree law with changes.

The new university incorporated the departments of Science-Literature and Engineering, the Vocational School in Kocaeli, a Science Institute, a Social Sciences Institute and the Foreign Languages, Atatürk Principles and the History of Revolution, Turkish Language, Physical Education and Fine Arts departments affiliated with the Rectorate.

Yıldız Technical University is one of the seven government universities situated in İstanbul besides being the 3rd oldest university of Turkey with its history dating back to 1911. It is regarded as one of the best universities in the country as well.

Ahmet ELBİR  
İbrahim Onur SİĞİRCİ

# İÇİNDEKİLER

---

<b>SİMGE LİSTESİ</b>	<b>vi</b>
<b>KISALTMA LİSTESİ</b>	<b>vii</b>
<b>ŞEKİL LİSTESİ</b>	<b>ix</b>
<b>TABLO LİSTESİ</b>	<b>x</b>
<b>ÖZET</b>	<b>xi</b>
<b>ABSTRACT</b>	<b>xiii</b>
<b>1 <math>\LaTeX</math></b>	<b>1</b>
1.1 $\LaTeX$ Nedir? . . . . .	1
1.2 $\LaTeX$ Nasıl Okunur? . . . . .	1
1.3 $\LaTeX$ mi, Word mü? . . . . .	1
1.4 $\LaTeX$ Yüklenmesi . . . . .	5
<b>2 Proje <math>\LaTeX</math> Formatı</b>	<b>6</b>
2.1 Dosya Yapısı . . . . .	6
2.2 Proje Bilgilerinin Girilmesi . . . . .	7
2.3 Proje Kitabı Ana Dosyasının Düzenlenmesi . . . . .	9
2.4 Bölümler ve Ekler . . . . .	10
<b>3 Semboller, Kısaltmalar ve Referanslar</b>	<b>12</b>
3.1 Semboller . . . . .	12
3.2 Kısaltmalar . . . . .	13
3.3 Referanslar . . . . .	13
3.3.1 Referans Dosyasının İsmi Değiştirilmesi . . . . .	13
3.3.2 $\LaTeX$ Formatında Referansların Oluşturulması . . . . .	14
3.3.3 Metin İçerisinde Referans Verme . . . . .	14
<b>4 Tablolar, Şekiller ve Matematiksel İfadeler</b>	<b>16</b>
4.1 Tablolar . . . . .	16

4.2	Şekiller . . . . .	17
4.3	Tablo ve Şekillerin Sayfa Yerleşimi . . . . .	18
4.4	Anahtar Kelimeler ve Referans Verme . . . . .	19
4.5	Matematik Formülleri . . . . .	19
<b>5</b>	<b>Listeleme ve Maddeleme</b>	<b>21</b>
5.1	Listeleme . . . . .	21
5.2	Maddeleme . . . . .	21
5.3	İç içe maddelemeler . . . . .	22
<b>6</b>	<b>Proje Kitabı Nasıl Yazılır?</b>	<b>23</b>
6.1	Genel Yazım Kuralları . . . . .	23
6.2	Proje Kitabının Bölümleri . . . . .	23
6.2.1	Kapak Sayfası . . . . .	24
6.2.2	Telif Hakkı Devir Sayfası . . . . .	24
6.2.3	Teşekkür . . . . .	24
6.2.4	İçindekiler . . . . .	24
6.2.5	Simge Listesi . . . . .	24
6.2.6	Kısaltma Listesi . . . . .	24
6.2.7	Şekil Listesi . . . . .	25
6.2.8	Tablo Listesi . . . . .	25
6.2.9	Özet . . . . .	25
6.2.10	Abstract . . . . .	25
6.2.11	Giriş . . . . .	25
6.2.12	Ön İnceleme . . . . .	26
6.2.13	Fizibilite . . . . .	26
6.2.14	Sistem Analizi . . . . .	28
6.2.15	Sistem Tasarımı . . . . .	28
6.2.16	Uygulama . . . . .	30
6.2.17	DeneySEL Sonuçlar . . . . .	30
6.2.18	Performans Analizi . . . . .	30
6.2.19	Sonuç . . . . .	30
6.2.20	Ekler . . . . .	31
6.2.21	Referanslar . . . . .	31
6.2.22	Özgeçmiş . . . . .	31
<b>7</b>	<b>Proje Raporlarının İçerikleri</b>	<b>32</b>
<b>8</b>	<b>Proje Dosyalarının Hazırlanması</b>	<b>34</b>
	<b>Referanslar</b>	<b>36</b>



## SÍMGE LÍSTESI

---

$A_i$	Activities of Daily Life
$c$	Alternate Step Test
$C$	Body Mass Index
$CR$	Cross Step moving on Four Stops
$f_c(.)$	Dynamic Bayesian Networks
$\Delta H$	Demura's Fall Risk Assessment Chart
$\lambda_i$	Electromyography
$\Omega$	Faculdade de Engenharia da Universidade do Porto

## KISALTMA LİSTESİ

---

ADL	Activities of Daily Life
AST	Alternate Step Test
BMI	Body Mass Index
CSFT	Cross Step moving on Four Stops
DBN	Dynamic Bayesian Networks
DFRAC	Demura's Fall Risk Assessment Chart
EMG	Electromyography
FEUP	Faculdade de Engenharia da Universidade do Porto
FPRI	Fall Prediction and Risk Index
FR	Fall Probability
FRI	Fall Risk Index
GDP	Gross Domestic Product
GUGT	Get-Up-and-Go Test
LABIOMEPE	Laboratório de Biomecânica do Porto
MEMs	Micro-Electromechanics
MTC	Minimum Toe Clearance
PCA	Principal Components Analysis
PPA	Physiological Profile Assessment
PPP	Purchasing Power Parities
SMWT	Six Meter Walking Test
STRATIFY	Saint Thomas's Risk Assessment Tool in Falling Elderly Inpatients
STST	Sit-To-Stand Test
STST5	Sit-To-Stand Test with 5 repetitions



SVM	State Vector Machine
SWHSA	Smart Wearable Health Systems and Applications
TUGT	Timed Up-and-Go Test
USB	Universal Serial Bus
USUST	Unstructured and Unsupervised Test
WEFAPS	Wearable Fall Assessment & Prediction System
WHO	World Health Organization

## ŞEKİL LİSTESİ

---

Şekil 1.1	Word ve $\text{\LaTeX}$ belgeleri hazırlanırken belge karmaşıklığına göre harcanan zaman grafiği . . . . .	2
Şekil 1.2	$\text{\LaTeX}$ ile çizilmiş örnek bir şekil . . . . .	4
Şekil 1.3	$\text{\LaTeX}$ ile çizilmiş ağaç . . . . .	4
Şekil 2.1	$\text{\LaTeX}$ taslağı dosya yapısı . . . . .	6
Şekil 2.2	projectChapters klasör örneği . . . . .	10
Şekil 2.3	main.tex içine bölüm ve eklerin eklenmesi . . . . .	10
Şekil 8.1	Hazırlanacak Dosyalama Şeması . . . . .	34

## TABLO LİSTESİ

---

Tablo 1.1 Word ile $\text{\LaTeX}$ 'in çeşitli kriterlere göre 3 puan üzerinden değerlendirilmesi . . . . .	2
---	---

# PROJE KİTABI HAZIRLAMA ESASLARI VE LATEX TASLAĞI KULLANIMI

Ahmet ELBİR

İbrahim Onur SİĞİRCİ

Bilgisayar Mühendisliği Bölümü

Bilgisayar Projesi

Danışman: Dr. Öğr. Üyesi Hamza Osman İLHAN

Ulaştırma alt sistemlerinden biri olan demiryolu, diğer ulaştırma alt sistemleriyle yoğun bir rekabet halinde bulunmaktadır. Yürütüle gelen yanlış politikalar sonucu ülkemizde demiryolu ulaştırmasına olan talep, yolcu ve yük taşımacılığında karayolunun oldukça gerisinde kalmıştır. Demiryolunun pazar payını arttırması ve rekabetini devam ettirebilmesi için hizmet kalitesini arttırması gerekmektedir. Dakiklik ve güvenilirlik bir ulaştırma alt sisteminin kalitesini belirleyen ölçütlerin başında gelmektedir. Bu ölçütlerin istenilen seviyede tutulabilmesi de kısmen etkin trafik kontrolü ile sağlanabilir.

Trenler önceden hazırlanmış bir hareket planına uygun biçimde hareket etmektedir. Ancak beklenmedik bazı olayların gerçekleşmesi sonucu gecikmeler ve trenler arası çatışmalar meydana gelebilmektedir. Trafik kontrolü, trenler arası çatışmaları, gecikmeleri mümkün olduğunca azaltacak şekilde çözüp, yeni bir uygulanabilir çizelge hazırlamak için uygulanır. Problemin zorluk derecesi nedeniyle, problemin en az gecikme içeren çözümüne kabul edilebilir bir süre içerisinde ulaşılması imkânsızdır. Bu çalışmada, 5 dakika gibi kısa bir süre içerisinde uygulanabilir ve gecikme toplamının olabildiğince küçüklendiği bir çizelge hazırlamak için, genetik algoritmalar kullanılmıştır. Geliştirilen algoritmanın çözümleri, belirli boyuttaki problemlerin kesin ve dispeçer çözümleri (yapay sinir ağı) ile karşılaştırıldığında, algoritma kısa sürede yeteri kadar iyi sonuçlar vermektedir. Algoritmanın uygulanması için geliştirilen bilgisayar programı, tren dispeçerleri için bir karar destek sistemi olarak

da kullanılabilir.

Trenler önceden hazırlanmış bir hareket planına uygun biçimde hareket etmektedir. Ancak beklenmedik bazı olayların gerçekleşmesi sonucu gecikmeler ve trenler arası çatışmalar meydana gelebilmektedir. Trafik kontrolü, trenler arası çatışmaları, gecikmeleri mümkün olduğunca azaltacak şekilde çözüp, yeni bir uygulanabilir çizelge hazırlamak için uygulanır. Problemin zorluk derecesi nedeniyle, problemin en az gecikme içeren çözümüne kabul edilebilir bir süre içerisinde ulaşılması imkânsızdır. Bu çalışmada, 5 dakika gibi kısa bir süre içerisinde uygulanabilir ve gecikme toplamının olabildiğince en küçüklendiği bir çizelge hazırlamak için, genetik algoritmalar kullanılmıştır. Geliştirilen algoritmanın çözümleri, belirli boyuttaki problemlerin kesin ve dispeçer çözümleri (yapay sinir ağı) ile karşılaştırıldığında, algoritma kısa sürede yeteri kadar iyi sonuçlar vermektedir. Algoritmanın uygulanması için geliştirilen bilgisayar programı, tren dispeçerleri için bir karar destek sistemi olarak da kullanılabilir.

**Anahtar Kelimeler:** Demiryolu trafik kontrolü, trenlerarası çatışmalar, yeniden çizelgeleme, genetik algoritmalar, yapay sinir ağları

### NAME OF THE PROJECT

Ahmet ELBİR

İbrahim Onur SİĞİRCİ

Department of Computer Engineering

Computer Project

Advisor: Assist. Prof. Dr. Hamza Osman İLHAN

In view of today's economic conditions chemical processes are operated or designed on the basis of optimum energy consumption. Thus primarily heat integration studies are undertaken and the design of the heat exchanger networks has entered into a new phase with the introduction of the pinch-point concept.

In this study, it is aimed at designing heat exchanger networks by the use of pinch-point design method, which is one of the significant heat integration methods. In the presentation of the work various theoretical approaches regarding the pinch-point design method are discussed, and a new "Improved Problem Algorithm Table" developed for the application of the design is introduced. Taking into account the scope of design in actual processes Visual Basic 3.0 programming language is used to develop the computer code called DarboTEK. This computer code can be used both in determining the minimum energy and area targets of a new plant to be constructed, and in making necessary design alterations in an already existing plant.

The crude petroleum unit in the TÜPRAŞ refinery at İzmit has been selected to show the applicability of the computer code developed to a real process, and as a result an original application has been accomplished. The heat integration study carried out on the crude petroleum unit shows that if a capital of 3576627 \$ is invested, the investment payback period is 1.7 years on the basis of the energy conservation achieved. Investment need is high; it is significant that it can be paid back by energy conservation in a reasonable period of time.

The crude petroleum unit in the TÜPRAŞ refinery at İzmit has been selected to show the applicability of the computer code developed to a real process, and as a result an original application has been accomplished. The heat integration study carried out on the crude petroleum unit shows that if a capital of 3576627 \$ is invested, the investment payback period is 1.7 years on the basis of the energy conservation achieved. Investment need is high; it is significant that it can be paid back by energy conservation in a reasonable period of time.

The crude petroleum unit in the TÜPRAŞ refinery at İzmit has been selected to show the applicability of the computer code developed to a real process, and as a result an original application has been accomplished. The heat integration study carried out on the crude petroleum unit shows that if a capital of 3576627 \$ is invested, the investment payback period is 1.7 years on the basis of the energy conservation achieved. Investment need is high; it is significant that it can be paid back by energy conservation in a reasonable period of time.

**Keywords:** Railway traffic control, conflicts between trains, re-scheduling, genetic algorithms, neural networks

Bu bölümde, TeX tanıtılacak ve hakkında genel kültür bilgileri verilecektir.

### 1.1 TeX Nedir?

TeX; makale, kitap, tez, sunum, poster gibi özellikle bilimsel çalışmaların yazılmasında ve raporlamasında kullanılan doküman hazırlama sistemidir. Bilim dünyasında neredeyse standart haline gelmiş TeX'in yüksek kalitede çıktılar üretmesi, üretilen çıktılarının farklı sürüm ve platformlarda kaymalara ve dağılmalara sebep olmaması, ücretsiz olması, derlenebilir bir dil olduğundan şarta bağlı ifadelerin rahatlıkla kullanılabilmesi, sayfa hakimiyetinin tamamen kullanıcıda olması vs. gibi pek çok avantajı vardır.

### 1.2 TeX Nasıl Okunur?

TeX; ilk defa kullanmaya başlayan ya da TeX kullanıcısı olmayan birçok insan tarafından genelde *la-teks* veya *ley-teks* şeklinde okunur. TeX kelimesinin İngilizce bir kelime olduğu düşüncesiyle bu şekilde okunmaktadır. İngilizcede *latex* diye bir kelime vardır ve *ley-teks* diye okunur ancak bu kelime kauçuk ağacından çıkarılmış maddeyi ifade etmektedir.

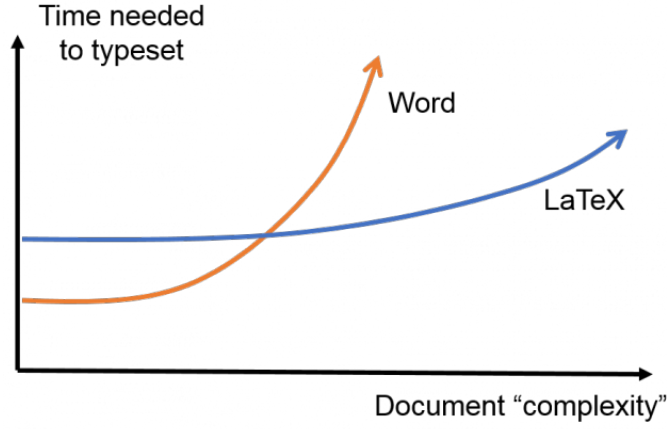
TeX kelimesi Yunan alfabesinde **tau**, **epsilon** ve **chi** harflerinden meydana gelmiştir. Buradaki **chi** harfi *Ki* şeklinde okunmaktadır. Bundan ötürü TeX kelimesi okunurken *lah-tek* veya *ley-tek* şeklinde okunur.

### 1.3 TeX mi, Word mü?

Word varken niçin TeX kullanalım?, sorusu birçok Word kullanıcısının sorduğu ilk sorudur? Word tarzı belge düzenleme programları WYSIWYG (What You See Is What You Get/Ne Görürsen Onu Alırsın) adı altında toplanmaktadırlar. Bu ifade



belgeyi oluştururken belgenin son haline en yakın halinin görülerek hazırlanmasından kaynaklanmaktadır. Ancak  $\text{\LaTeX}$  belgeleri yazılırken dokümanın nihai hali belge oluşturulurken görülmemektedir.



**Şekil 1.1** Word ve  $\text{\LaTeX}$  belgeleri hazırlanırken belge karmaşıklığına göre harcanan zaman grafiği

$\text{\LaTeX}$  mi, Word mü? sorusu ile ilgili birçok çalışma yapılmıştır [1, 2]. Bu çalışmalar sonucunda Şekil-1.1 bu çalışmalar sonucu elde edilmiş bir grafiği göstermektedir. Kullanıcılar üzerinde yapılan çeşitli deneyler sonucunda dokümanın karmaşıklığı arttıkça  $\text{\LaTeX}$  kullanmanın daha verimli olduğu gösterilmiştir.

**Tablo 1.1** Word ile  $\text{\LaTeX}$ 'in çeşitli kriterlere göre 3 puan üzerinden değerlendirilmesi

Özellik	Word Puanı	Latex Puanı
Küçük doküman hazırlama hızı	3	2
Büyük doküman hazırlama ve grafiklerle uğraşma hızı	1	3
Kullanma kolaylığı	3	1
Düzen ve çıktı kalitesi	2	3
Bilimsel özellikler	1	3
Ücret ve kullanılabilirlik (erişilebilirlik)	1	3
Uyumluluk	2	2
<b>Toplam</b>	<b>13</b>	<b>18</b>

Tablo-1.1, çeşitli kriterlere göre Word ve  $\text{\LaTeX}$  kullanmanın puanlamasını göstermektedir.

$\text{\LaTeX}$ 'in Word'e göre kesin olan bazı avantajlarını şu şekilde sıralayabiliriz:

- Yazılan belgeler herhangi bir editör ile okunabilir veya yazılabilir. Genellikle .txt editörleri yazıları görmek için yeterlidir. Ancak Word için Office Word programının yüklü olması gerekmektedir.

- $\text{\LaTeX}$  çıktı olarak (kullanıcının isteğine bağlı olarak) PDF üretir. Haliyle çıktılar farklı ortamlarda kaymalara, bozukluklara sebep olmaz.
- Word belgeleri hazırlanırken oluşan düzen bozuklukları, dokümanı oluştururken aksamalara veya vakit kaybına sebep olabilmektedir.  $\text{\LaTeX}$  kullanıcıları dokümanı hazırlarken tamamen belgenin içeriğine odaklanırlar.
- Word belgelerinde kopyalama işlemi çoğu zaman problemlere yol açmaktadır. Farklı bir belgeye kopyalandığında o belgenin düzenine uydurulması, düzenlenmesi gerekmektedir. Ancak  $\text{\LaTeX}$  için kopyalama çok kolaydır. Sital dosyası farklı olduğundan, sital dosyası değiştiğinde düzen otomatik olarak değişir ve kullanıcı farklı belgeler için aynı dosyayı düzenlemekle uğraşmaz.
- Düzen, yazı tipleri, tablolar, şekiller vs. hepsi hazırlanan doküman boyunca tutarlıdır.
- Dizinler, dipnotlar, alıntılar, kaynaklar, içindekiler vs. kolaylıkla üretilebilir.
- Matematiksel formüllerin metin içerisindeki kalitesi yüksektir.

Elbetteki  $\text{\LaTeX}$  için de bazı olumsuzluklar var. Örneğin yazılan dokümanın son halini yazıyı yazarken görememek (Word kullanmadan gelen alışkanlık) kullanıcılarda başlangıçta tedirginliğe sebep olmaktadır. Ancak zamanla alışılan bir durumdur. Eğer dokümanı siz biçimlendirmek istiyorsanız biçimlendirme komutlarını bilmeniz gerekmektedir.

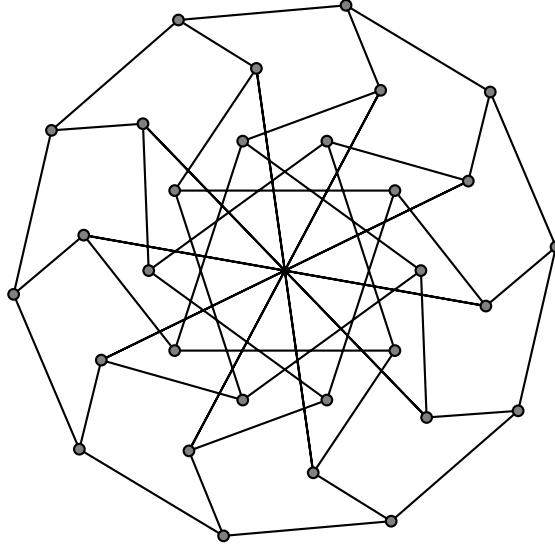
Şekil-1.2  $\text{\LaTeX}$  kullanılarak çizilmiştir. Word kullanarak çizmenin çok zor olduğu bu ve benzeri şekiller basit ve kısa kodlarla çok rahat bir biçimde yapılabilir. Bu şekli çizen kod aşağıda verilmiştir (**Not:**  $\text{\LaTeX}$ 'de karmaşık şekil çizimleri profesyonellik gerektirmektedir. İleri seviyede olmayanlar için bu kodlar karmaşık gelebilir. Tez yazımında bu profesyonellik beklenmemektedir. )

```

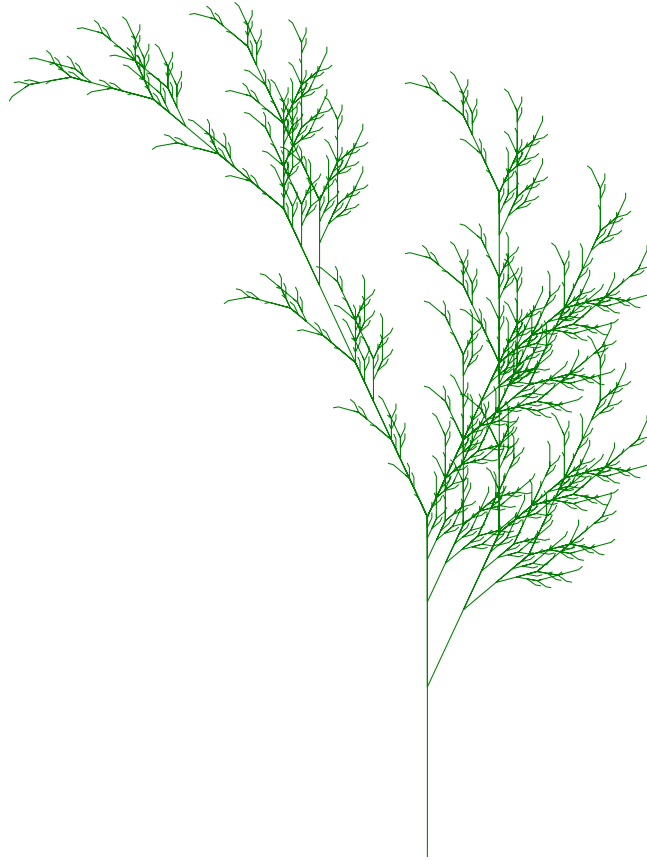
\tikzstyle{every node}=[circle, draw, fill=black!50,
                        inner sep=0pt, minimum width=4pt]
\begin{tikzpicture}[thick,scale=0.9]
  \draw \foreach \x in {0,36,...,324}
  {
    (\x:2) node {} -- (\x+108:2)
    (\x-10:3) node {} -- (\x+5:4)
    (\x-10:3) -- (\x+36:2)
    (\x-10:3) --(\x+170:3)
  }

```

```
(\x+5:4) node {} -- (\x+41:4)
};
\end{tikzpicture}
```



Şekil 1.2  $\text{\LaTeX}$  ile çizilmiş örnek bir şekil



Şekil 1.3  $\text{\LaTeX}$  ile çizilmiş ağaç

## 1.4 $\LaTeX$ Yklenmesi

$\LaTeX$  yerel olarak bilgisayarlar kurulanabileceđi gibi, online platformlar zerinden de kullanılmaktadır. Bilgisayar Mhendisliđi olarak bizim tavsiyemiz [www.sharelatex.com](http://www.sharelatex.com) sitesinin kullanılmasıdır. Yapılması gereken proje koordinatrlđ sayfasında verilen sıkıřtırılmıř  $\LaTeX$  taslak dosyasının bu siteye yklenmesidir. Online bu sistem, paket ykleme derdinden kurtardığı gibi dokman paylařma zelliđi ile birden fazla kiřinin dokmanı inceleme ve dzenlemesine olanak sađlamaktadır.

# 2

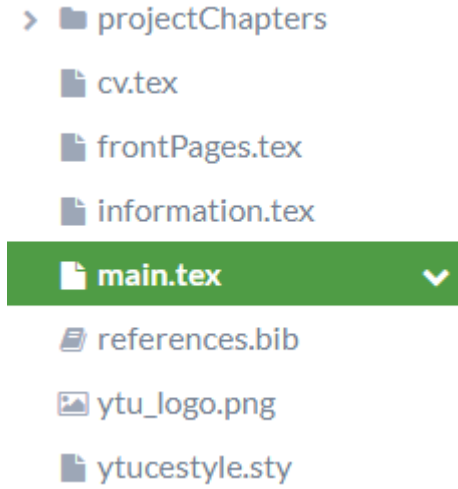
## Proje $\LaTeX$ Formatı

---

Bu bölümde Yıldız Teknik Üniversitesi Bilgisayar Mühendisliği Bölümü'nün bilgisayar ve bitirme projeleri için kullanılacak  $\LaTeX$  formatı tanıtılacaktır.

### 2.1 Dosya Yapısı

$\LaTeX$  formatı hazırlanırken, kullananların en az bilgi ile rahatça kullanabilecekleri bir şekilde olmasına dikkat edilmiştir. Bunun için anlaşılır bir dosya yapısı tercih edilmiştir.



Şekil 2.1  $\LaTeX$  taslağı dosya yapısı

Şekil-2.1, proje yazımı için tasarlanmış  $\LaTeX$  taslağının dosya yapısını göstermektedir. Burada adı geçen dosya ve klasörleri şu şekilde inceleyebiliriz:

**main.tex** Tüm dosya yapısının başında yer almaktadır. Derleyici üzerinde bu kod çalıştırılmalıdır.

**information.tex** Proje ile alakalı bilgilerin doldurulması gereken dosyadır. İlerleyen başlıklarda nasıl doldurulacağı anlatılacaktır.

**references.bib** Proje içerisindeki referansların bulunduğu dosyadır. Referans işlemlerinin nasıl yapılacağı sonraki bölümlerde bahsedilecektir.

**ytuthesis.sty** Proje için hazırlanmış stil dosyasıdır. Bu dosya üzerinde değişiklik yapılmaması gerekmektedir.

**frontPages.tex** dosyası, proje kitabının ön sayfalarında yer alan bölümleri içermektedir. Bu kısımdaki dosyaların değiştirilmemesi gerekmektedir.

**projectChapters** klasörü, proje kitabının içerisindeki bölümleri ve proje kitabında kullanılan resimleri içermektedir.

**cv.tex** dosyası öğrenci özgeçmiş sayfasını oluşturmaktadır. Bu sayfa üzerinde değişiklik yapılmaması gerekmektedir.

## 2.2 Proje Bilgilerinin Girilmesi

Proje ile alakalı bilgiler `information.tex` dosyasında sorulduğu biçimi ile yazılacaktır. Tüm bilgi alanlarının formatı şu şekildedir:

```
% Asagidaki kisma .... yaziniz
\def\tanim{ Bu Kismi Doldurunuz }
```

Yukarıda görüldüğü gibi, her bir bilgi üzerinde açıklama yer almaktadır. Bilgiler ise küme parantezleri ile gösterilen alanın arasına yazılmalıdır. Bu dosyada istenilen bilgileri şu şekilde sıralayabiliriz:

1. **titleTR**, proje başlığı büyük harflerle Türkçe olarak yazılmalıdır.
2. **titleEN**, proje başlığı büyük harflerle İngilizce olarak yazılmalıdır.
3. **studenti**, projeyi yapan öğrencilerden ilkinin ismi ve soyismi yazılmalıdır. İsimlerin sadece ilk harfi büyük olmalıdır.
4. **numberi**, projeyi yapan öğrencilerden ilkinin öğrenci numarası yazılmalıdır.
5. **studentibdate**, projeyi yapan öğrencilerden ilkinin doğum tarihini ve doğum yeri yazılmalıdır.
6. **studentiemail**, projeyi yapan öğrencilerden ilkinin e-posta adresi yazılmalıdır.
7. **studentiphone**, projeyi yapan öğrencilerden ilkinin telefon numarası yazılmalıdır.

8. **studentiintern**, projeyi yapan öğrencilerden ilkinin staj bilgileri yazılmalıdır.
9. **studentii**, projeyi yapan öğrencilerden ikincisinin ismi ve soyismi yazılmalıdır. İsimlerin sadece ilk harfi büyük olmalıdır. Eğer ikinci öğrenci yoksa ~ yazılmalıdır.
10. **numberii**, projeyi yapan öğrencilerden varsa ikincisinin öğrenci numarası yazılmalıdır.
11. **studentiibdate**, projeyi yapan öğrencilerden varsa ikincisinin doğum tarihini ve doğum yeri yazılmalıdır.
12. **studentiemail**, projeyi yapan öğrencilerden varsa ikincisinin e-posta adresi yazılmalıdır.
13. **studentiiphone**, projeyi yapan öğrencilerden varsa ikincisinin telefon numarası yazılmalıdır.
14. **studentiintern**, projeyi yapan öğrencilerden varsa ikincisinin staj bilgileri yazılmalıdır.
15. **date**, proje kitabının teslim edildiği tarih "ay, yıl" olarak yazılmalıdır. Burada ay, proje yazım dilinde kelime olarak yazılmalıdır.
16. **advisorTR**, proje danışmanının ünvanı, ismi ve soyismi yazılmalıdır. Ünvan Türkçe olarak yazılmalıdır. İsimlerin ise sadece ilk harfleri büyük olmalıdır.
17. **advisorEN**, proje danışmanının ünvanı, ismi ve soyismi yazılmalıdır. Ünvan İngilizce olarak yazılmalıdır. İsimlerin ise sadece ilk harfleri büyük olmalıdır.
18. **acknowledgementText**, teşekkür metni proje yazım dilinde yazılmalıdır.
19. **abstractTextEnglish**, projenin özet bilgisi İngilizce olarak yazılmalıdır.
20. **abstractKeywordsEnglish**, proje için geçerli olan anahtar kelimeler İngilizce olarak yazılmalıdır. Aralarına virgül konulmalıdır.
21. **abstractTextTurkish**, projenin vözet bilgisi Türkçe olarak yazılmalıdır.
22. **abstractKeywordsTurkish**, proje için geçerli olan anahtar kelimeler Türkçe olarak yazılmalıdır. Aralarına virgül konulmalıdır.
23. **symbols**, tez içerisinde kullanılan semboller ve anlamları yazılmalıdır. "\item[sembol] Açıklaması" şeklinde yazılmalıdır. Eğer tez içerisinde sembol kullanılmadıysa "\def\symbols{}" olacak şekilde küme parantezleri içerisindeki metin silinmelidir.

24. **abbreviations**, tez içerisinde kullanılan kısaltmalar ve anlamları yazılmalıdır. "`\item[kısaltma]` Açıklaması" şeklinde yazılmalıdır. Eğer tez içerisinde kısaltma kullanılmadıysa "`\def\abbreviations{}`" olacak şekilde küme parantezleri içerisindeki metin silinmelidir.

Dikkat edilirse, isim içerisinde ayırım belirtilmediği sürece tüm bilgiler proje hangi dilde hazırlanıyorsa o dilde yazılmaktadır.

### 2.3 Proje Kitabı Ana Dosyasının Düzenlenmesi

Proje kitabı taslak hiyerarşisinin en üstündeki dosya `main.tex` dosyasıdır. Bu dosya üzerinde bazı düzenlemeler yapılması gerekebilmektedir.

Ana dosyanın içerisinde aşağıdaki gibi bir satır bulunmaktadır:

```
\usepackage[computer, final]{ytucestyle}
```

Bu satırda köşeli parantezler içerisinde yer alan bilgiler yazılan proje ile alakalı bazı parametreleri göstermektedir. Verilebilecek iki farklı parametre vardır:

1. **Projenin türü:** Hangi kademe için proje kitabı yazılıyorsa ona uygun olarak parametre verilmelidir. Bilgisayar projesi ise *computer*, bitirme projesi ise *senior* yazılmalıdır.
2. **Projenin durumu:** Proje kitabı için bulunulan aşama belirtilmelidir. Eğer ara rapor verecekseniz *report*, proje kitabının son halini teslim edecekseniz (final sınavı dahil) *final* yazılmalıdır. Bu parametre üçüncü rapor teslimi için ya da başarılı not alan öğrencilerin teslim edeceği proje kitapları için kullanılmalıdır.

Parametrelerin sırası önemsiz olmakla birlikte aşağıdaki örnekleri inceleyiniz:

- `\usepackage[computer, final]{ytuthesis}`

Bilgisayar projesinin final sınavı ve sonrası için teslim edilecek halini ifade etmektedir.

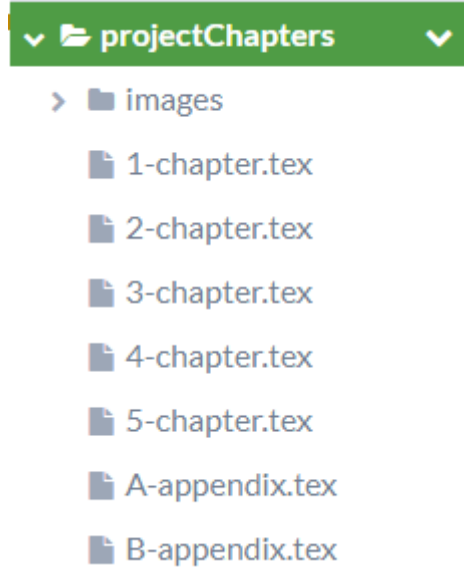
- `\usepackage[senior, report]{ytuthesis}`

Bitirme projesinin ara raporları için teslim edilecek halini ifade etmektedir.



## 2.4 Bölümler ve Ekler

Tercihe bağlı olmakla birlikte, kullanıcıların dosyalarını daha rahat organize edebilmeleri için `projectChapters` isiminde bir klasör oluşturulmuştur. Şekil-2.2 bu klasörün içerisindeki dosyaları göstermektedir.



Şekil 2.2 `projectChapters` klasör örneği

Şekil-2.2'ten görüldüğü üzere, proje için yazılması planlanan her bir bölüm veya ek ayrı bir `.tex` uzantılı dosya ile saklanmıştır. Tamamı tek bir `.tex` dosyasında da saklanabilir ancak bu durum yazma hakimiyeti azaltabilir. Aynı zamanda proje kitabı içinde kullanılacak resimler için `images` isimli bir klasör açılmıştır.

`projectChapters` klasöründe bölümler/ekler için açılan her bir dosyanın `main.tex` içerisine eklenmesi gerekmektedir.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%      CHAPTERS      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
\input {projectChapters/1-chapter.tex}  
\input {projectChapters/2-chapter.tex}  
\input {projectChapters/3-chapter.tex}  
\input {projectChapters/4-chapter.tex}  
  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%      APPENDICES      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
\newpage\appendix  
  
\input {projectChapters/A-appendix.tex}  
\input {projectChapters/B-appendix.tex}
```

Şekil 2.3 `main.tex` içine bölüm ve eklerin eklenmesi

Şekil-2.3’de görüldüğü gibi, eklenmek istenilen bölümler veya ekler `input` komutu ile `main.tex` içerisine eklenmektedir. Dosyalar hangi sıra ile verilirse, proje kitabının bölümleri o sırada oluşur.

Bu bölümde sembollerin kısaltmaların ve referansların projenin  $\text{\LaTeX}$  formatına nasıl ekleneceğinden bahsedilecektir:

### 3.1 Semboller

information.tex dosyasında yer alan symbols bilgisi içerisinde şu formatta eklenmelidir:

```
\item[SEMBOL] Sembolun aciklamasi
```

item komutundan sonra köşeli parantez içerisinde sembol, köşeli parantezden sonra, boşluk sayısı önemli olmaksızın, sembolün açıklaması yazılmalıdır. Verilen formata göre ilgili örnek aşağıda verilmiştir:

```
\begin{abbrv}
  \item[Ai]           Activities of Daily Life
  \item[c]           Alternate Step Test
  \item[C]           Body Mass Index
  \item[CR]          Cross Step moving on Four Stops
  \item[$fc(.)$]     Dynamic Bayesian Networks
  \item[$\Delta H$]  Demura's Fall Risk Assessment Chart
  \item[$\lambda i$] Electromyography
\end{abbrv}
```

Eğer proje raporu içerisinde sembol kullanılmıyorsa, "`\def\symbols{}`" şeklinde küme parantezleri içerisi boş bırakılmalıdır.

## 3.2 Kısaltmalar

information.tex dosyasında yer alan abbreviations bilgisi içerisine şu formatta eklenmelidir:

```
\item[KISALTMA] Kısaltmanın açıklaması
```

item komutundan sonra köşeli parantez içerisinde kısaltma, köşeli parantezden sonra, boşluk sayısı önemli olmaksızın, kısaltmanın açıklaması yazılmalıdır. Verilen formata göre ilgili örnek aşağıda verilmiştir:

```
\begin{abbrv}
  \item[ADL]      Activities of Daily Life
  \item[AST]      Alternate Step Test
  \item[BMI]      Body Mass Index
  \item[CSFT]     Cross Step moving on Four Stops
  \item[DBN]      Dynamic Bayesian Networks
  \item[DFRAC]    Demura's Fall Risk Assessment Chart
  \item[EMG]      Electromyography
\end{abbrv}
```

Eğer proje raporu içerisinde kısaltma kullanılmıyorsa, "`\def\abbreviations{}`" şeklinde küme parantezleri içerisi boş bırakılmalıdır.

## 3.3 Referanslar

Proje için tasarlanan  $\text{\LaTeX}$  formatında referansların rahat eklenebilmesi ve kullanılabilmesi için references.bib dosyası oluşturulmuştur.

### 3.3.1 Referans Dosyasının İsmi Değiştirilmesi

references.bib dosyasının ismi değiştirilmek istenilirse, main.tex dosyasında yer alan şu satırda ilgili değişikliklerin yapılması gerekmektedir:

```
\addbibresource{references.bib}
```

### 3.3.2 L<sup>A</sup>T<sub>E</sub>X Formatında Referansların Oluşturulması

Referansları elde etmenin en iyi yolu, referans gösterilecek kaynağın sitesinden L<sup>A</sup>T<sub>E</sub>X formatında referans bilgilerini almaktır. [scholar.google.com](http://scholar.google.com) adresi üzerinden de bu iş kolayca yapılmaktadır. İlgili kaynak site üzerinde aratıldığında çıkan her bir sonucun sağ alt köşesindeki "Alıntı Yap" seçeneğine tıklanmalıdır. Çıkan pencerenin sol alt köşesinde "BibTex" yazmaktadır. Buraya tıklanıldığında çıkan bilgilerin tamamı `references.bib` dosyasının içerisine kaydedilmelidir. Örnek bir alıntı aşağıda gösterilmiştir:

```
@article{knauff2014efficiency,  
  title={An Efficiency Comparison ...},  
  author={Knauff, Markus and Nejasmic, Jelica},  
  journal={PloS one},  
  volume={9},  
  number={12},  
  pages={e115069},  
  year={2014},  
  publisher={Public Library of Science}  
}
```

Yukarıdaki örnek alıntı önemli olan ilk satırda yer alan `knauff2014efficiency` ifadesidir. Her alıntıda bu kısımda yer alan ifade anahtar kelimedir ve tüm alıntılar için burada yazılı olan ifadenin farklı olması gerekmektedir. (Buradaki ifadeleri siz de belirleyebilirsiniz). Bu anahtar kelimeler, metin içerisinde referans verilirken kullanılacaktır.

Bilgiler `references.bib` içerisine kaydedilirken sıra önemli değildir. Önemli olan projenin bölümleri içerisinde verilme sırasıdır. L<sup>A</sup>T<sub>E</sub>X numaralandırmayı bu düzene göre kendisi yapacaktır.

### 3.3.3 Metin İçerisinde Referans Verme

Metin içerisinde referans verirken, referans verilecek yere `cite` komutu ile alıntının anahtar kelimesi yazılır. Aşağıdaki örnek kullanımı inceleyiniz:

There are two airports in Istanbul. Ataturk Airport is on the  
→ European Side of the city, and Sabiha Gokcen Airport is on  
→ the Asian Side. As both of the airports are located outside  
→ the city centre you may find the taxi\cite{rao2012novel}  
→ fees fairly expensive.

Eğer birden fazla kaynak gösterilecekse; anahtar kelimeler virgül ile ayrılarak tek bir adet cite içerisine yazılmalıdır. Aşağıdaki örneği inceleyiniz:

There are two airports in Istanbul. Ataturk Airport is on the  
→ European Side of the city, and Sabiha Gokcen Airport is on  
→ the Asian Side. As both of the airports are located outside  
→ the city centre you may find the taxi\cite{rao2012novel,  
→ knauff2014efficiency, brischoux2009don} fees fairly  
→ expensive.

Bu bölümde  $\text{\LaTeX}$  için tablo ve resim eklemekten bahsedilecektir.

## 4.1 Tablolar

$\text{\LaTeX}$ 'e tablo eklemek için temel olarak aşağıdaki komut kullanılmaktadır:

```
\begin{table}[yerlesim]
  \centering
  \caption{My caption}
  \label{mylabel}
```

TABLO

```
\end{table}
```

Yukarıdaki kodlardan görüleceği üzere, bir tablo eklemek için dört bilgiye ihtiyaç vardır:

1. Yerleşim: Tablonun sayfanın neresine yerleşeceği bilgisi verilir. Bu bilginin nasıl olacağı daha sonra açıklanacaktır.
2. `caption`: Tablonun açıklaması kısmında yazılacak bilgiler
3. `label`: Tabloya referans verilmesi için gerekli olacak anahtar kelime
4. Tablo: Tablonun kendisi. Burada herhangi bir bilgi tablo olarak eklenebilmektedir. Ancak satır, sütun mantığı ile en temel tablo tabular komutu ile eklenmektedir.

www.tablesgenerator.com web sayfası  $\text{\LaTeX}$ 'e yeni başlayan kullanıcılar için tablo oluşturma konusunda yardımcı olmaktadır. Bu site üzerinden tablonun kodları oluşturulup proje kitabına eklenebilir. Bu site kodları oluştururken yerleşim kısmını boş bırakmaktadır. Bu kısmı sizin doldurmanız gerekmektedir.

## 4.2 Şekiller

$\text{\LaTeX}$ 'e şekil eklemek için temel olarak aşağıdaki komut kullanılmaktadır:

```
\begin{figure}[yerlesim]
  \centering

  SEKIL

  \caption{My caption}
  \label{mylabel}
\end{figure}
```

Yukarıdaki kodlardan görüleceği üzere, bir şekil eklemek için dört bilgiye ihtiyaç vardır:

1. Yerleşim: Şeklin sayfanın neresine yerleşeceği bilgisi verilir. Bu bilginin nasıl olacağı daha sonra açıklanacaktır.
2. Şekil: Şeklin kendisi. Burada herhangi bir bilgi şekil olarak eklenebilmektedir.
3. caption: Şeklin açıklaması kısmında yazılacak bilgiler
4. label: Şeklin referans verilmesi için gerekli olacak anahtar kelime

Burada tablolardan farklı olarak caption bilgisi aşağıda yer almıştır. caption bilgisi nerede yer alırsa, şeklin ya da tablonun açıklaması ona göre yer değiştirmektedir.

Şekil kısmına resim ekleme şu şekilde yapılabilir:

```
\includegraphics[scale=0.6]{projectChapters/images/Picture1.png}
```

Bu komut ile projectChapters klasörü içerisinde oluşturulan images klasöründeki Picture1.png dosyası eklenmiştir. Eklenirken scale komutu ile boyutu %60'a



indirgenmiştir. Eğer resmi metin uzunluğuna göre boyutlandırmak istiyorsanız komut aşağıdaki gibi yazılmalıdır:

```
\includegraphics[width=0.2\textwidth]{projectChapters/images/Picture1.png}
```

Bu komut ile resmin ekleneceği yerdeki metin uzunluğunun %20'si kadar olacak şekilde resmin genişliğini ayarlar. Resmin genişliği ile aynı oranda enide ayarlanmış olur.

### 4.3 Tablo ve Şekillerin Sayfa Yerleşimi

Bu bilgi tablo ve şekillerin dokümanın neresine yerleştirilmesi gerektiğini bildirmektedir:

- **h**: Yaklaşık olarak buraya yerleştir (here)
- **t**: Sayfanın en üstüne yerleştir (top)
- **b**: Sayfanın en altına yerleştir (bottom)
- **p**: Özel bir sayfaya yerleştir
- **!**: Genelleştirilmiş parametreleri burada yoksay.
- **H**: Kesin olarak buraya yerleştir. **h!** komutuna karşılık gelmektedir.

Bu bilgilerin biri veya birkaçı verilerek yerleşim sağlanmaktadır. Birden fazla olduğu durumda sırası ile uygun olan yerleşimi yapmaktadır. Aşağıdaki örneği inceleyiniz:

```
\begin{figure}[!htbp]
  \centering
  \includegraphics{projectChapters/images/Picture1.png}
  \caption{Ornek resim ekleme}
  \label{anahtarkelime}
\end{figure}
```

## 4.4 Anahtar Kelimeler ve Referans Verme

Eklenecek tablo veya şekle metin içerisinde referans verebilmek için ekleme sırasında `label` komutu ile anahtar kelime ataması yapılmıştır.

Anahtar kelimeler boşluksuz ve İngiliz alfabesinin karakterlerinden oluşan kelimeler olmalıdır. Birçok latex kullanıcısı, anahtar kelimenin neye ait olduğunu rahat anlamak için anahtar kelimelerin önüne bazı tanımlayıcılar koyarlar (Not: Bu işlem zorunlu değildir). Örneğin tablonun anahtar kelimesi ise `tab:`, şeklin anahtar kelimesi ise `fig:` tanımlayıcısını kullanırlar. Böylece metin içerisinde referans verilen bilginin tablo mu, şekil mi olduğu anlaşılacaktır.

Metin içerisinde referans verirken `ref` komutu kullanılmaktadır. Bu komutun içerisine anahtar kelime yazılmalıdır. `ref` komutu ile tablo veya şeklin sadece numarası alındığından bu komut öncesinde *tablo* veya *şekil* kelimeleri yazılmalıdır. Aşağıdaki örneği inceleyiniz:

Haydi bir resim ekleyelim:

```
\begin{figure}[!ht]
  \centering
  \includegraphics{projectChapters/images/Picture1.png}
  \caption{Resim ekleme örneği}
  \label{fig:ornekresim}
\end{figure}
```

Resim-`\ref{fig:ornekresim}`, örnek bir haritadır.

## 4.5 Matematik Formülleri

[www.codecogs.com/latex/eqneditor.php](http://www.codecogs.com/latex/eqneditor.php) web sayfası üzerinden matematik formüllerinizi kolayca oluşturup, elde edilen kodları buraya ekleyebilirsiniz. Elde edilen kodlar numaralandırılarak eklenecekse şu şekilde eklenmelidir:

```
\begin{equation}
  Denklem Kodlari
\end{equation}
```

Eğer metin içerisinde matematiksel ifade kullanılacaksa `$` işaretleri arasında yazılmalıdır. Örneğin  $\frac{3}{2}$  yazmak için

`\frac{3}{2}`

yazılmalıdır.

Bu bölümde  $\text{\LaTeX}$  kullanarak listeleme ve maddeleme işlemleri gösterilecektir.

### 5.1 Listeleme

Listeleme yapmak için aşağıdaki komut yazılır:

```
\begin{itemize}
  \item Birinci madde
  \item İkinci madde
  \item Üçüncü madde
\end{itemize}
```

Yukarıdaki komut çalıştırıldığında aşağıdaki sonuç çıkacaktır:

- Birinci madde
- İkinci madde
- Üçüncü madde

### 5.2 Maddeleme

Numaralandırma yaparak maddeleme yapabilmek için aşağıdaki komut yazılır:

```
\begin{enumerate}
  \item Birinci madde
  \item İkinci madde
  \item Üçüncü madde
\end{enumerate}
```

Yukarıdaki komut çalıştırıldığında aşağıdaki sonuç çıkacaktır:

1. Birinci madde
2. İkinci madde
3. Üçüncü madde

### 5.3 İç içe maddelemeler

İç içe listeler veya numaralandırılmış listeler oluşturulmak istenildiğinde belirtilen komutlar iç içe yazılması yeterli olacaktır. Örneğin;

```
\begin{itemize}
  \item Birinci seviye, maddeleme, birinci madde
  \begin{itemize}
    \item İkinci seviye, maddeleme, birinci madde
    \item İkinci seviye, maddeleme, ikinci madde
    \begin{enumerate}
      \item Üçüncü seviye, numaralandırma, birinci madde
      \item Üçüncü seviye, numaralandırma, ikinci madde
    \end{enumerate}
  \end{itemize}
  \item Birinci seviye, maddeleme, ikinci madde
\end{itemize}
```

Yukarıdaki komut çalıştırıldığında sonuç aşağıdaki gibi olacaktır:

- Birinci seviye, maddeleme, birinci madde
  - İkinci seviye, maddeleme, birinci madde
  - İkinci seviye, maddeleme, ikinci madde
    1. Üçüncü seviye, numaralandırma, birinci madde
    2. Üçüncü seviye, numaralandırma, ikinci madde
- Birinci seviye, maddeleme, ikinci madde

Bu bölümde proje kitaplarının nasıl yazılacağı hakkında bilgi verilecektir.

### 6.1 Genel Yazım Kuralları

Proje kitabı ve ara raporlar rahat anlaşılır, dil bilgisi ve yazım kurallarına uygun ve basit bir dille yazılmalıdır. Cümlelerin mümkün olduğunca kısa ve fiil zamanlarının uyumlu olması anlatımı kolaylaştıracaktır. Yazılanlarda anlam ve anlatım bütünlüğüne dikkat edilmelidir. Özellikle İngilizce yazılan raporlarda kaynaklardan kopyala-yapıştır metoduyla bilgi kesinlikle alınmamalıdır. Alınacak bilgi yorumlanarak yeniden yazılmalı ve kaynağa atıfta bulunulmalıdır.

Proje kitabı tamamlanmış bir çalışmayı anlattığı için öğrenilen (miş’li) geçmiş zaman kullanılmalıdır. Proje çalışması edilgen bir yapıda (yapılmıştır, kullanılmıştır gibi) anlatılmalıdır. Genel bilgiler ise geniş zaman kullanılarak (yapılır, eklenir gibi) verilmelidir.

### 6.2 Proje Kitabının Bölümleri

Genellikle proje kitabını okuyanlar, her ana bölümün ilk paragrafını okuyarak o bölüm hakkında fikir sahibi olmaya çalışırlar. Bunun için Giriş bölümünden sonra her ana bölümün ilk paragrafı o bölümü ana hatlarıyla tanıtmalıdır.

Aşağıda verilecek bölümlerin içerikleri projenin kapsamı ve gerçekleştirilen sistemin yapısına bağlı olarak değişiklikler gösterebilir. Ancak her proje için bu başlıklar yer almalıdır. Örneğin bir web uygulaması için uygulama bölümünde çok sayıda ekran çıktısına yer verilirken, performans analizi başlığında verilen bilgi sayfaların açılma süreleri veya uygulamanın tepki süresinin ölçülmesiyle sınırlı kalabilir. Öte yandan akademik sonuçlar üretmeye yönelik bilimsel yöntemler içeren bir çalışmada bunun tam tersi bir yapı söz konusu olacak ve deneysel sonuçlar ile performans analizleri

yoğun içeriğe sahip olacaktır.

Proje Kitabı şablonuna uygun olarak proje kitabı içerisinde bulunması gereken bölümler ve sıralaması aşağıda verilmiştir:

### **6.2.1 Kapak Sayfası**

Bu kısım  $\LaTeX$  tarafından oluşturulmaktadır.

### **6.2.2 Telif Hakkı Devir Sayfası**

Bu kısım  $\LaTeX$  tarafından oluşturulmaktadır.

### **6.2.3 Teşekkür**

Bu kısım, uzun bir çalışmayı tamamlayan ekibin, projenin teknik ve bilimsel içeriğinden bağımsız olarak görüşlerini yazdığı bölümdür. Ayrıca bu bölümde, proje çalışması sırasında bilgi, kaynak v.b. yardımı alınan kişi ve kuruluşlara teşekkür edilmelidir. Bu bölümde, çalışmasını tamamlayan ekip kendisine/kendilerine destek olan, yardım eden ailelerine ve arkadaşlarına da ayrıca teşekkür edebilir.

`information.tex` dosyası içerisinde ilgili kısım doldurulduğunda  $\LaTeX$  tarafından oluşturulmaktadır.

### **6.2.4 İçindekiler**

Bu kısım  $\LaTeX$  tarafından oluşturulmaktadır.

### **6.2.5 Simge Listesi**

`information.tex` dosyası içerisinde ilgili kısım doldurulduğunda  $\LaTeX$  tarafından oluşturulmaktadır. Eğer sembol kullanılmıyorsa `information.tex` dosyasındaki ilgili kısım boş bırakılmalıdır.

### **6.2.6 Kısaltma Listesi**

`information.tex` dosyası içerisinde ilgili kısım doldurulduğunda  $\LaTeX$  tarafından oluşturulmaktadır. Eğer kısaltma kullanılmıyorsa `information.tex` dosyasındaki ilgili kısım boş bırakılmalıdır.

### 6.2.7 Şekil Listesi

Bu kısım  $\LaTeX$  tarafından oluşturulmaktadır.

### 6.2.8 Tablo Listesi

Bu kısım  $\LaTeX$  tarafından oluşturulmaktadır.

### 6.2.9 Özet

Bir proje kitabının en çok okunan bölümleri özet, giriş ve sonuç bölümleridir. Konu hakkında sadece genel bilgi edinmek isteyen kişiler çoğunlukla bu üç bölümü okumakla yetinirler. Bunun için proje konusu ve elde edilen önemli sonuçlar özet, giriş ve sonuç bölümlerinde bölüm özelliğine uygun detayda yinelenmelidir.

Özetin amacı, okuyucunun proje konusu hakkında genel bir fikir sahibi olmasını sağlamaktır. Özetin ilk paragrafında proje konusu tanıtılmalıdır. Diğer paragraflarda çalışmanın motivasyonu (çalışmanın neden yapıldığı), kapsamı, amacı ve katkısı adım adım anlatılmalı, kullanılan yöntemler kısaca tanıtılmalı ve ana sonuçlar verilmelidir.

Özet, tamamlanmış bir çalışmayı anlattığı için öğrenilen (miş’li) geçmiş zaman kullanılmalıdır. Anlatım, “yapılmıştır, tamamlanmıştır” gibi edilgen yapıda olmalıdır.

information.tex dosyası içerisinde ilgili kısım doldurulduğunda  $\LaTeX$  tarafından oluşturulmaktadır.

### 6.2.10 Abstract

Bu bölüm, Özet bölümünün birebir İngilizce çevirisi olmalıdır. Bölümümüze değişim programlarıyla ve yabancı öğrenci kontenjanlarıyla gelen öğrenciler ancak Abstract bölümünü okuyarak Bilgisayar Projeleri hakkında bilgi sahibi olabilmektedirler. Bu sebeple çevirinin özenli yapılması, hazır çeviri araçlarıyla yetinilmemesi projenin hitap edeceği kitleyi uluslararası ölçüğe taşıyacaktır.

information.tex dosyası içerisinde ilgili kısım doldurulduğunda  $\LaTeX$  tarafından oluşturulmaktadır.

### 6.2.11 Giriş

Giriş bölümü, okuyucunun konuyu anlayıp projenin amacını ve konuya nasıl bir katkıda bulunduğunu değerlendirebilmesi için yeterli temel bilgileri içermelidir. Bu



amaçla projenin konusu tanımlanmalı, çalışmanın yapılmasının gereği, amacı ve hedefi kısaca anlatılmalıdır.

Projenin motivasyonu; bir başka deyişle, bu konunun seçiliş sebebi ve konunun neden önemli olduğu, giriş bölümünde iyi bir şekilde vurgulanmalıdır. Ayrıca projenin konuya olan özgün katkısı varsa giriş bölümünde açık bir şekilde anlatılmalıdır. Bunların yanında proje çalışmasının anlaşılabilmesi için bilinmesi gereken ön bilgiler, giriş bölümünde okuyucuya aktarılmalıdır.

Giriş bölümünün sonunda okuyucunun hangi bölümleri okuyacağına karar vermesini kolaylaştırmak için proje kitabının sonraki bölümleri kısaca tanıtılmalıdır.

### 6.2.12 Ön İnceleme

Ön inceleme bölümü, projenin yapılacağı alanda daha önceden yapılmış olan mevcut ve benzeri çalışmaların incelenmesini içerir. Ön incelemenin amacı, projenin önceki çalışmalardaki hangi eksikleri gidereceği ya da hangi yolun izleneceğinin belirlenmesidir. Proje alanının etüt edilmesi ile kabaca gerçekleştirilecek projenin hangi ihtiyaçlara cevap vereceği belirlenir.

Bu bölümde konu ile ilgili daha önce yapılmış olan çalışmalar anlatılmalı ve değerlendirmeleri yapılmalıdır. Bilimsel ve akademik içerikli çalışmalarda literatür özeti olarak bilinen bu bölümde konunun daha önceden hangi yönleriyle ele alınarak hangi sonuçların elde edildiği karşılaştırmalı olarak değerlendirilir. Bu sayede konuda uygulanan yöntemler ve eksik yönler ortaya çıkarılmış olur.

Bilgi sistemi ya da web uygulaması gibi projelerde halihazırda işlerin nasıl yürüdüğü anlatılmalı, aynı konuda yapılmış benzer sistemler ve örnekler karşılaştırmalı olarak değerlendirilmelidir. Bunların çalışma ilkeleri ve özellikleri, varsa eksik yönleri ve projenin bu alanda ne katkı sağlayacağı belirtilmelidir.

### 6.2.13 Fizibilite

Fizibilite bölümünde projenin yapılabilirlik etüdü gerçekleştirilmelidir. Projenin başarıya ulaşması için kaynakların alternatifler arasından hangi şartlara uygun olarak nasıl seçildiği, projenin zaman planlaması, projenin ekonomik öngörülere ve yapılması durumunda oluşacak katma değerler bu bölümde incelenir ve belirlenir. Genel olarak Fizibilite çalışması,

- **Teknik Fizibilite:** Teknik fizibilite, marka bağımsız teknik özelliklerin vurgulandığı bir çalışma olup kendi içinde Yazılım Fizibilitesi, Donanım

Fizibilitesi ve Haberleşme (İletişim) Fizibilitesi şeklinde gruplanabilir.

Yazılım Fizibilitesi başlığında, kurulacak sistemi oluşturmak üzere seçilen İşletim Sistemi, Veri Tabanı Yönetim Sistemi, Web Sunucusu ve/veya diğer sunucu sistemleri, Uygulama Geliştirme Dili, Uygulama Geliştirme Ortamı, Diğer Ek Destek Yazılım Araçları gibi yazılım unsurlarının neler olduğunun ve bu unsurlarda ne tür teknik özellikler beklendiğinin vurgulanması gereklidir. Bu yazılım unsurlarının doğal olarak marka bağımlı alternatifleri de vardır. Beklenen teknik özelliklere göre hangi alternatifin seçildiği karşılaştırmalar yaparak ortaya konmalıdır. Bu durumda seçilen alternatifin neden diğerlerinden üstün olduğu somut nedenlere dayanarak vurgulanmalıdır.

Donanım Fizibilitesinde hedef, yazılım fizibilitesinde seçilen yazılım unsurlarını ve bu unsurlar kullanılarak gerçekleştirilecek olan uygulamayı başarı ile çalıştırmaya müsait, kısa ve orta vadeli gelişmelere açık bir sistemin seçilmesidir. Kullanılacak yazılım araçları için en düşük sistem gereksinimleri (İşlemci hızı, RAM ihtiyacı, Disk ihtiyacı vb.) göz önüne alınarak bir kurulum belirlenmelidir. Ancak gerek teknolojinin geldiği nokta gerek ise işleyişin rahat olması için bu belirlenen özelliklerden daha iyi bir kurulum önerilmelidir. Bu fizibilite tamamen teknik veriler ışığında yapılabilir ve marka belirtilmesi gerekmez. Uygulamanın özel bir donanıma bağlı olarak yapılmasının gerekli olduğu durumlarda Yazılım Fizibilitesi donanımın özellikleri göz önüne alınarak hazırlanabilir.

İletişim Fizibilitesi başlığı altında iletişim teknolojileri ve buna bağlı olarak kullanılması uygun olan cihaz/yazılım vb. teknik özellikler incelenmelidir.

- **İş gücü ve Zaman Planlaması:** İş gücü ve Zaman planlaması yapılırken sonuca ulaşmak için takip edilecek ara adımlar tam olarak belirlenmeli ve bu adımların hangi teknik bilgi birikimine sahip kişilerce ve ne kadar sürede yapılabileceği yönünde planlar oluşturulmalıdır. Bu plan Gantt diyagramı kullanılarak ifade edilmelidir.
- **Yasal Fizibilite:** Yasal Fizibilite, yapılan işin mevcut kanun ve yönetmeliklere uygun olup olmadığı, herhangi bir patent vb. korunmuş hakkı ihlal edip etmediği ve/veya bazı teknolojilerin kullanılması için alınması gereken izinlerin olup olmadığı gibi unsurları değerlendirmek için yapılmalıdır. Eğer bu unsurlar söz konusu değilse Yasal Fizibilite yapılamaz.
- **Ekonomik Fizibilite:** Ekonomik Fizibilitede, teknik (yazılım, donanım, iletişim), iş gücü/zaman fizibilitesi ve yasal fizibiliteden doğacak harcamalar detaylandırılmalı ve projenin toplam giderleri ortaya konmalıdır. Bunun yanında projenin devreye alınması ile sağlanacak tasarruf ya da elde edilecek

gelirler de öngörülerek mali gelir analizi yapılmalıdır. Gelir ve gider kalemleri başabaş noktası grafiği gibi yöntemlerle karşılaştırılarak sistemin harcamaları ne kadar sürede amorti edeceği gösterilebilir.

başlıklarını içermelidir. Bunlar dışında olabilecek sosyal, alternatif vb. fizibilite çalışmaları proje ihtiyaçlarına bağlı olarak proje danışmanı yönlendirmesiyle eklenmelidir.

#### **6.2.14 Sistem Analizi**

Sistem analizinin amacı projede en uygun çözümü bulmak için ana öğeler ve işlevlerin ortaya çıkarılıp tanımlanmasıdır. Bu bölümde projenin hedefleri detaylandırılır. Ayrıca bilgi kaynakları ve gereksinimler belirlenir. Bölüm sonunda proje ile ilgili tüm gereksinimler tanımlanmış ve tasarım aşamasına geçecek en uygun çözüm belirlenmiş olmalıdır.

Gereksinimlerin ortaya çıkarılması için kullanılan araştırma ve bilgi toplama yöntemleri (görüşme, anket, vb.) ve sonuçları bu bölümde verilmelidir. Belirlenen gereksinimler ışığında sistemdeki modüller, kullanıcılar ve roller tanımlanmalıdır.

Gereksinim analizi modeli fonksiyonel ya da nesneye dayalı olarak hazırlanabilir. Eğer fonksiyonel yaklaşım tercih edilirse, sistemin nasıl çalışacağına dair iş akışları ve analiz modeline ait veri akış diyagramları hazırlanmalıdır. Veri akış diyagramları ikinci düzeye kadar hazırlanmalı ve modüllerin analizi yapılmalıdır. Nesneye dayalı analiz tercih edildiğinde ise kullanım senaryoları (use-case) çözümlemesi ile gereksinimler belirlenmeli, kavramsal sınıf diyagramı ile analiz modeli oluşturulmalıdır.

Sistem analizi sonucunda proje sonlandığında ihtiyaçların karşılanıp karşılanmadığını test edecek yapı da ortaya çıkmaktadır. Bu sebeple projede kullanılacak performans metrikleri (hız, tanıma başarısı, özellik sayısı, vb.) de bu bölümde tanımlanmalıdır.

#### **6.2.15 Sistem Tasarımı**

Analiz aşaması bittikten sonra gerçekleştirilen tasarım aşaması bu bölümde raporlanmalıdır. Bu bölüm temel olarak; yazılım tasarımı, veritabanı tasarımı ve girdi-çıkış tasarımı alt bölümlerinden oluşmalıdır.

##### **6.2.15.1 Yazılım Tasarımı**

Yazılım tasarımında, projede gerçekleştirilecek sistemin mimari tasarımı oluşturulmalıdır. Proje, bilgi sistemi gibi bir yazılım projesi ise bu bölüm sistem

tasarımı ilkelerine göre hazırlanmalıdır. Proje, bilimsel bir sonuç veya deney için hazırlanıyorsa o zaman uygulanan bilimsel yöntem ve metodolojiler sırası ile tanıtılarak sistem tasarlanmalıdır.

Yazılım projelerinin tasarımı, bir önceki bölümde olduğu gibi bu alt bölüm de fonksiyonel ya da nesneye dayalı yaklaşımlarla hazırlanabilir. Fonksiyonel yaklaşım tercih edilecek ise veri akış diyagramlarından hareketle modüller hazırlanır ve yapı diyagramları ile tasarlanır. Nesneye dayalı tasarım, etkileşim diyagramları ile (ardışıl diyagramlar veya işbirliği diyagramları) yapılmalıdır. Bu aşamada yazılım sınıfları da ortaya çıkarılarak tasarım sınıf diyagramı da hazırlanmalıdır.

Bilimsel ve akademik çalışmalardan oluşan projelerde ise öncelikle, sistemin genel yapısı ve çalışma şekline ait blok diyagramı verilmelidir. Daha sonra, verilere uygulanan ön işleme adımları, kullanıldıysa özellikler üzerinde yapılan hazırlık işlemleri (özellik çıkarımı gibi), uygulanan yöntemler (sınıflandırma, kümeleme, zaman serisi tahmini, vb.) gibi bileşenler ayrıntılarıyla anlatılmalı ve blok diagramındaki her ana işlem aşamasında yapılan çalışmalara ait teknik bilgi ve gerekli yarı kod ve/veya akış diyagramı verilmeli, parametre seçimi, algoritma özelleştirme adımları gibi konular detaylı olarak anlatılmalıdır.

#### **6.2.15.2 Veritabanı Tasarımı**

Veritabanı tasarımında, projenin analiz ve tasarım aşamalarında belirlenen, hakkında veri tutulması gereken varlıklar ve bunlar arasındaki ilişkiler ortaya koymak için varlık-ilişki diyagramı (Entity-Relationship, E-R) kullanılmalıdır. Varlıklar ve aralarındaki ilişkiler tasarlandıktan sonra, eğer gerekiyorsa veri üzerinde veritabanı düzeyinde işlem yapmak için gerekli olan yordamlar (stored procedures) da tasarlanmalıdır. Veri üzerinde olay ve durumlara göre otomatik tetiklenerek çalışması gereken işlemler (trigger) de oluşturmalıdır.

Bilimsel veya akademik içerikli projelerde ise kullanılan veri kümeleri, özellikleri ve yapılarıyla tanıtılmalıdır. Veri kümeleri oldukları haliyle kullanılmayıp üzerinde seçme ya da azaltma gibi işlemler yapıldıysa bu bölümde açıklanmalıdır. Veri yapıları ve saklama biçimleri bu bölümde tariflenmelidir.

#### **6.2.15.3 Girdi-Çıktı Tasarımı**

Bu bölümde projenin veri girişlerini, işlem kontrollerini ve sonuç çıktılarını sağlayan ekranlar gibi arayüzlerin tasarımları verilmelidir. Uygulama penceresi, web sayfası, mobil cihaz arayüzü gibi çeşitli girdi ortamlarından hangileri kullanıldıysa bu bölümde

yer almalıdır. Çıktı için form, rapor veya grafik gibi değişik ortamlar kullanılıyorsa bunların tasarımları da bu bölümde verilmelidir.

#### **6.2.16 Uygulama**

Bu bölümde gerçekleştirilen projenin çalışan haline ait ekran görüntüleri yer almalıdır. Özellikle gereksinimleri karşılayacak şekilde modül veya işlemlerin gösterilmesi beklenmektedir. Uygulamada yapılan kontroller, gerekirse her adımdan sonra ara çıktılar, girilen verinin sonucunda sistemin işlemesi ve elde edilen çıktılar gibi sistemin çalışan hali bu bölümde verilmelidir.

#### **6.2.17 Deneysel Sonuçlar**

Yazılım uygulamalarında sistemin değişik senaryolarla çalışması sonucunda oluşan çıktılar bu bölümde verilmelidir. Genel senaryoların dışında problemin uç noktaları gözönünde bulundurularak sistemin her koşul için girdi-çıkıtı uyumuna sahip olup olmadığı incelenmelidir. Gerçekleştirilen uygulamada, farklı parametreler kullanılarak yapılan testlere ait sonuçların, farklı yöntemlerle elde edilen sonuçların, ya da farklı veri kümeleri üzerindeki uygulamaların karşılaştırmalı sonuçlarının bu bölümde verilmesi gereklidir.

#### **6.2.18 Performans Analizi**

Sistemin çalışma performansı bu bölümde değerlendirilmelidir. Performans değerlendirme için kullanılan yöntemler tanıtılmalı (örneğin stres testi) ve sonuçları tartışılarak değerlendirilmelidir.

Bilimsel içerikli projelerde ise bir önceki bölümde elde edilen deneysel sonuçlar değerlendirilmeli ve karşılaştırmalı olarak yorumlanmalıdır. Hangi yöntem veya parametre seti ile nasıl başarı sağlandığı, başarının neden yüksek ya da düşük olduğu objektif olarak yorumlanmalıdır.

#### **6.2.19 Sonuç**

Sonuç bölümü, gerçekleştirilen çalışmadan elde edilen sonuçların değerlendirildiği bölümdür. Bu bölümde;

- Proje konusu tanımlanmalı ve kullanılan yöntemler özetlenmelidir.
- Elde edilen sonuçlar açık ve basit cümlelerle ifade edilmelidir.

- Deneysel çalışmalarda, farklı deney sonuçlarının değerlendirilmesinden elde edilen ana sonuçlar anlatılmalı mümkünse bu sonuçlara göre genellemeler yapılmalıdır.
- Bu konuda çalışmak isteyenlere yol göstermek için yapılan proje çalışmasında başlangıçta belirlenen hedefe ne kadar ulaşıldığı, çalışmanın üstün ve eksik yönleri anlatılmalıdır.
- İleriye yönelik çalışmalar için, varsa öneriler belirtilmelidir.

### 6.2.20 Ekler

Proje kitabınızda ihtiyaç duyulursa ekler bölümünde, proje metni içinde yer alması gerekli olmayan ve okunurluğu bozacak kadar büyük ve detaylı tablo, akış diyagramı gibi bilgiler bulunmalıdır. Bu bilgilere proje metni içinde gerektiği yerde referans verilmelidir.

### 6.2.21 Referanslar

Bu kısım  $\text{\LaTeX}$  tarafından oluşturulmaktadır.

### 6.2.22 Özgeçmiş

information.tex dosyası içerisinde ilgili kısım doldurulduğunda  $\text{\LaTeX}$  tarafından oluşturulmaktadır.

Dönem içerisinde iki adet rapor teslim edilmektedir. Bu iki rapordan sonra geçer not alan öğrenciler proje kitabını teslim etmektedirler. Raporlar projenin gelişme durumunu ifade etmektedirler. Raporlarda yer alması gereken başlıklar şu şekildedir:

### Ara Rapor

Ara Raporunda yer alması gereken bölümler aşağıdaki gibidir:

- İçindekiler Listesi *(taslak tarafından otomatik oluşturulmaktadır)*
- Simge Listesi *(information.tex dosyasını doldurmalısınız)*
- Kısaltma Listesi *(information.tex dosyasını doldurmalısınız)*
- Tablo Listesi *(taslak tarafından otomatik oluşturulmaktadır)*
- Şekil Listesi *(taslak tarafından otomatik oluşturulmaktadır)*
- Giriş
- Ön İnceleme
- Fizibilite
- Sistem Analizi
- Sistem Tasarımı
- Uygulama
- Referanslar *(taslak tarafından otomatik oluşturulmaktadır)*

## Final Raporu

Final Raporu, Ara Raporu'da yer alan bölümlere ek olarak şu bölümleri içermelidir:

- Teşekkür *(information.tex dosyasını doldurmalısınız)*
- Özet *(information.tex dosyasını doldurmalısınız)*
- Abstract *(information.tex dosyasını doldurmalısınız)*
- Deneysel Sonuçlar
- Performans Analizi
- Sonuç

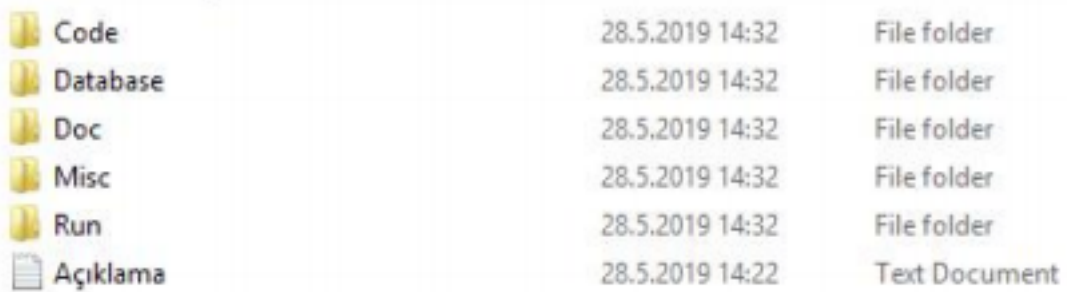
Bölüm-2.3'de belirtilen kurallara göre  $\text{\LaTeX}$  kendi oluşturduğu kısımları duruma göre oluşturmaktadır. Bunun dışında projeyi anlatan hangi bölümün hangi raporda olacağı sizin tarafınızdan eklenecektir.



# 8

## Proje Dosyalarının Hazırlanması

Proje dökümanları koordinatörlük tarafından paylaşılacak portal üzerinden sisteme yüklenmelidir. Teslim aşamasında dosyalama düzeni Şekil 8.1’de gösterildiği gibi olmalıdır.



Code	28.5.2019 14:32	File folder
Database	28.5.2019 14:32	File folder
Doc	28.5.2019 14:32	File folder
Misc	28.5.2019 14:32	File folder
Run	28.5.2019 14:32	File folder
Açıklama	28.5.2019 14:22	Text Document

Şekil 8.1 Hazırlanacak Dosyalama Şeması

Proje dosya isimlendirmelerinde iki kişilik projeler için **OgrNo1\_OgrNo2.rar** olarak grubun öğrenci numaralarının yazılması yeterlidir. Proje dökümanlarına konulması gereken dosyalara ait açıklamalar aşağıda belirtilmiştir. Ayrıca güncellemeler için proje koordinatörlük sayfasını takip etmeniz gerekmektedir. Dosya boyutu, kullanılan veri seti nedeniyle ayrılan alandan fazla olma durumunda, öğrencilerin danışman hocası ile iletişime geçmeleri ve sisteme kullanılan veri seti olmadan proje dökümanları yüklenmelidir.

- **Proje Açıklama Dosyası:** Açıklama.txt isminde projenin yeni bir bilgisayara kurulması aşamasında yapılması gereken işlemlerin adım adım anlatıldığı, gerekli ayarlamaların açıklandığı bir dosya hazırlanacaktır. Bu dosyada yazılanların takip edilerek yapılması durumunda projenin çalıştırılabilir olması esastır.
- **Proje Kitabı:** Doc klasörü altında projenin basılı olarak teslim edilen kitabının elektronik kopyası yer almalıdır. Diğer bir deyişle kitap hazırlanırken kullanılan  $\text{\LaTeX}$  dokümanları ve bu dokümanlardan üretilen PDF dosyası yer almalıdır.

Bilgisayar Projesi öğrencileri için hazırlamaları gereken bildiri formatında 4 sayfalık İngilizce raporun da PDF ve  $\text{\LaTeX}$  dosyaları yüklenen dökümanlar içeriğinde yer almalıdır.

- **Proje Kaynak Kodu:** Code klasörü altında proje dahilinde geliştirilen programların açık kaynak kodları yer almalıdır.
- **Projenin Çalıştırılabilir Hali:** Run klasörü altında projenin derlenmiş ve çalıştırmaya hazır şekli yer almalıdır. Özellikle kurularak çalıştırılması gereken uygulamalarda bu dizin altında programın kurulmasını sağlayacak yazılımların (install scripts vb.) listesi .txt dosyası şeklinde belirtilmelidir.
- **Proje Veri Tabanı:** Database klasörü altında (var ise) proje içinde kullanılan veritabanının yaratılmasını sağlayan dosyalar (creation scripts) ve veri tabanının proje esnasında kullanılan (içinde örnek bilgilerin bulunduğu) halinin yedeklenmiş hali (backup veya export olarak) bulunmalıdır. Veritabanı kullanılmayan projelerde veri saklanan dosyalar (XML vb.) da bu klasörde yer alabilir. Akademik içerikli projelerde kullanılan veri kümeleri bu klasörde verilebilir. Yüksek boyutlu (>50 MB) veri setlerinde ise, veriye ait kaynak linkinin belirtilmesi yeterlidir.
- **Proje içinde kullanılan diğer ek yazılımlar:** Misc klasörü altında proje içinde kullanılan freeware, shareware türü programlar varsa bunlara ait liste .txt olarak belirtilmelidir. (Kullanılan programlama dili, veri tabanı yönetim sistemi, işletim sistemleri gibi yazılımlar buraya kopyalanmamalıdır.)

## Referanslar

---

- [1] M. Knauff and J. Nejasmic, “An efficiency comparison of document preparation systems used in academic research and development,” *PloS one*, vol. 9, no. 12, e115069, 2014.
- [2] F. Brischoux and P Legagneux, “Don’t format manuscripts,” *The Scientist*, vol. 23, no. 7, p. 24, 2009.

### BİRİNCİ ÜYE

**İsim-Soyisim:** Ahmet ELBİR  
**Doğum Tarihi ve Yeri:** 03.06.2020, İstanbul  
**E-mail:** ytu@ytu.com.tr  
**Telefon:** 0555 555 55 55  
**Staj Tecrübeleri:** ABCDE Şirketi Yazılım Departmanı

### İKİNCİ ÜYE

**İsim-Soyisim:** İbrahim Onur SIĞIRCI  
**Doğum Tarihi ve Yeri:** 03.06.2020, İstanbul  
**E-mail:** ytu@ytu.com.tr  
**Telefon:** 0555 555 55 56  
**Staj Tecrübeleri:** ABCDE Şirketi Donanım Departmanı

### Proje Sistem Bilgileri

**Sistem ve Yazılım:** Windows İşletim Sistemi, Python  
**Gerekli RAM:** 2GB  
**Gerekli Disk:** 256MB