



EGE ÜNİVERSİTESİ  
MÜHENDİSLİK FAKÜLTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

# Microcontroller Based System Design

2020-2021 BAHAR YARIYILI

## RFID-based Attendance Systems

### Dersi Laboratuvar Projesi

(Report)

Grup No:

05130001350

Atiqurhaman Mayar

05170000809

Morteza Yosefy

05160000856

Mohammad Reza Yaqobi

---

Tarih: 08-07-2021

## Table of Contents

1. İhtiyaç ve Proje Analizi.....	2
a. Projenin amacı ve hedefi .....	2
b. Kullanıcıya yönelik faydaları .....	2
c. Kullanılan yazılım dili, derleyici veya diğer araç gereksinimleri .....	2
2. Tasarım .....	3
a. Proje tasarımında kullandığınız platformlar .....	3
Bileşenler .....	4
b. Projenin devre şeması .....	5
c. Projenin akış diyagramı.....	5
d. Tasarım ekran görüntüleri .....	6
3. Kodlama.....	7
a. Kullandığınız giriş ve çıkış değişkenleri (portlar, vs.).....	7
b. Kullandığınız Timer ve Interrupt verileri, ne amaçla kullanıldığı .....	8
c. Kullandığınız fonksiyonlar veya alt program çağrıları, ne amaçla yazıldığı.....	8
4. Doğrulama ve Test.....	11
a. Projenizin kısıtları.....	11
i. Projede neler yapıldı? .....	11
ii. Projenizin özgünlüğü ve pratik yaşama katkısı nedir? .....	12
iii. Projenin eksiklikleri neler? .....	12
iv. Projede olması gereken diğer durumlar .....	12
b. Projeden beklenen sonuçlar .....	13
c. Ortak çalışma adına gerçekleştirilen görev paylaşımları ve toplantılardan görüntüler .....	13

# 1. İhtiyaç ve Proje Analizi

## a. Projenin amacı ve hedefi

Günümüzde okullara ve kolejlere katılım kağıda dayanmaktadır. Bazen bu işlem hatalara neden olur ve daha fazla zaman alır. Bu proje, bir sınıfa giren her öğrencinin not alması için RFID teknolojisini kullanır ve ayrıca bir sınıfta geçen süreyi hesaplar.


Bu sistemde her öğrenciye bir RFID etiketi verilmektedir. Katılım, kart RFID okuyucunun yanına yerleştirilerek yapılabilir.

## b. Kullanıcıya yönelik faydaları

RFID Tabanlı Katılım Sistemi Uygulamaları

- RFID tabanlı katılım sistemleri, eğitim kurumlarında, endüstrilerde, her yerde kullanılabilir.
- RFID gelişmekte olan bir teknolojidir ve kimlik doğrulamanın gerekli olduğu uygulamalarda kullanılır.

## c. Kullanılan yazılım dili, derleyici veya diğer araç gereksinimleri

 yazılım dili:

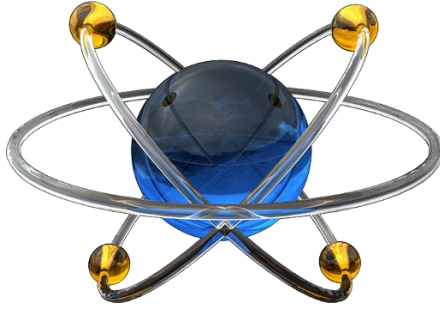
- C programming for 8051

 derleyici:

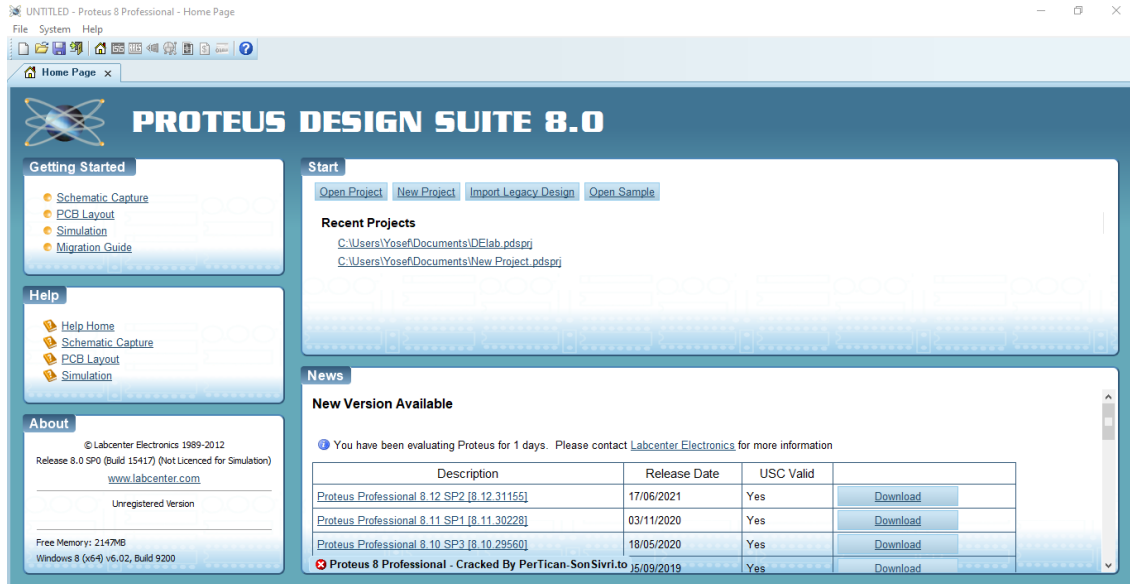
- Keil (µVision® IDE)
- Proteus 8 Professional

## 2. Tasarım

### a. Proje tasarımında kullandığınız platformlar



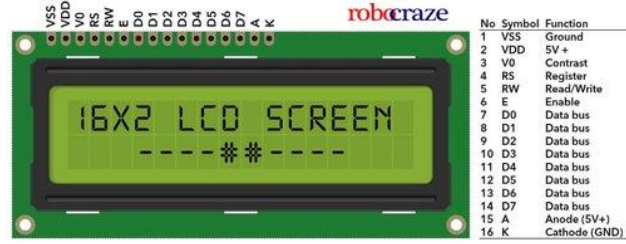
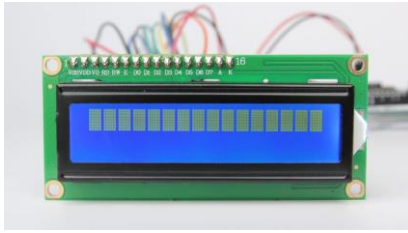
**ARMKEIL**  
Microcontroller Tools



## Bileşenler

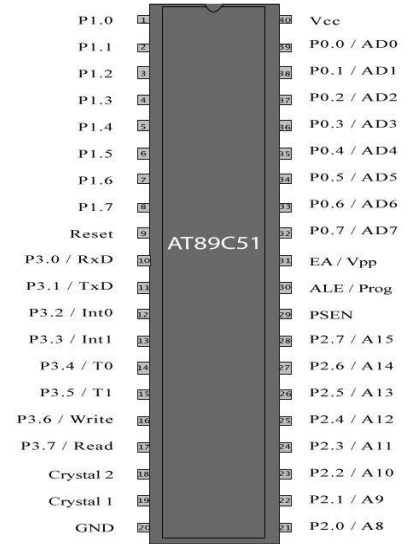
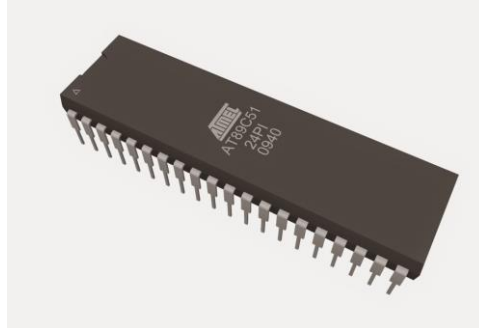
Tasarımınızda kullandığınız malzemeler (devre elemanları, lojik kapılar, denetleyici, led, göstergeler, vs.) ve ne amaçla kullanıldığı ile ilgili kısa bir açıklama:

### ✚ Alphanumeric LCD 16x2



### ✚ AT89C51 Microcontroller

AT89C51, 4K bayt Flash programlanabilir ve silinebilir salt okunur belleğe (PEROM) sahip, düşük güçlü, yüksek performanslı bir CMOS 8 bit mikro bilgisayardır. Atmel ailesinden 8 bitlik eski bir mikro denetleyicidir. Popüler 8051 mimarisiyle çalışır.

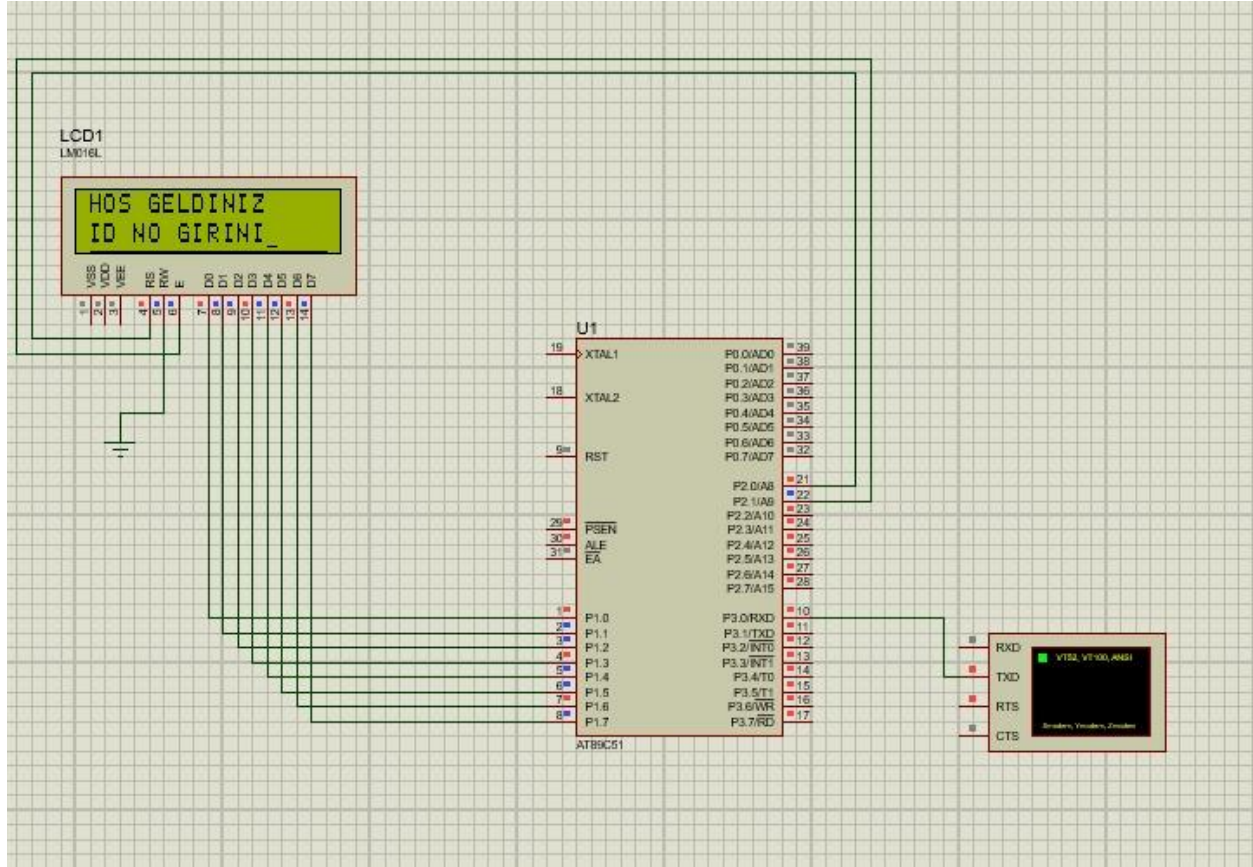


### ✚ EM-18 rfid reader module

- Biz virtual terminal kullandık

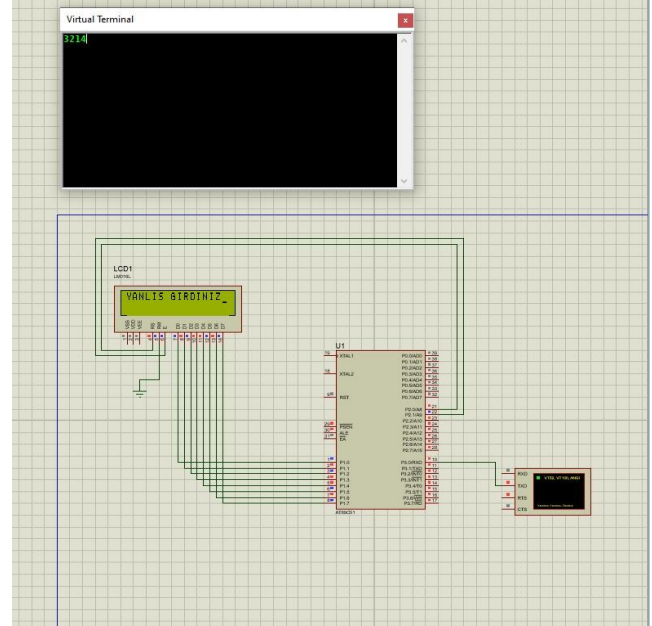
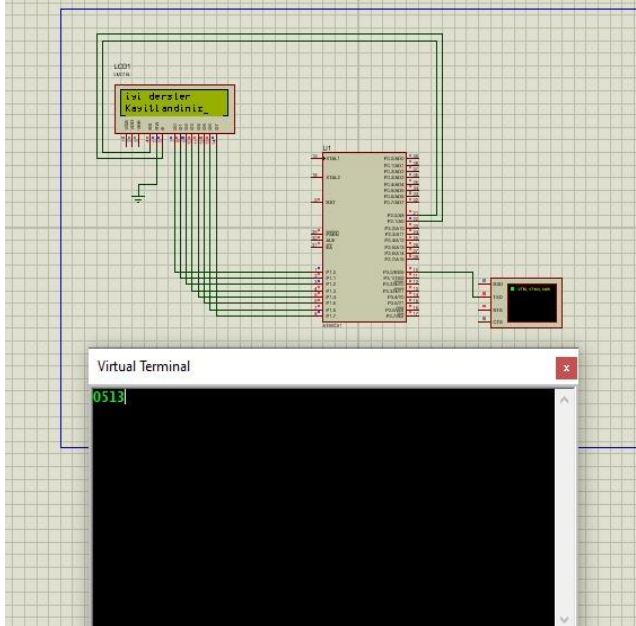


b. Projenin devre şeması



c. Projenin akış diyagramı

#### d. Tasarım ekran görüntüleri



Kiel ortamında öncelikle kodu çalıştırdık - sorunsuz çalışıyor.

##### Build Output

```
Rebuild started: Project: RFID katilim sistemi son
Rebuild target 'Target 1'
assembling STARTUP.A51...
compiling RFID katilim son.c...
linking...
Program Size: data=20.0 xdata=0 code=929
creating hex file from ".\Objects\RFID katilim sistemi son"...
".\Objects\RFID katilim sistemi son" - 0 Error(s), 0 Warning(s).
Build Time Elapsed: 00:00:00
```

### 3. Kodlama

#### a. Kullandığınız giriş ve çıkış değişkenleri (portlar, vs.)

```
#include <reg51.h>    // microcontrollerlerin fileleri hep burada saklanır
#include <string.h>    // string fonksiyonu LCD ye giris icin kullanilir.

sbit rs = P2^0; // LCD nin rs komutu verileri / komutlari gonderme islemi yapar
sbit e = P2^1;  // high levelden low levele sinyal gonderen e komutu , yine LCD y
e ait baska bir komut

void lcd_cmd(int a)
{
    P1=a;        //  parametre olarak a degerini alan P1 portu, cikti islemini go
ruyor. lcd'den ciktilari aliyor
    rs=0;        //   LCD ye komut gondermek icin rs degeri =0 olmalı
    e=1;         //   enable high olacak
    delay(10);   // enable high ve low arasinda bir delay gerceklesecek.
    e=0;         //   enable low olck
}

void lcd_data(int a)
{
    P1=a;
    rs=1;        // LCD ye veri / komutlar gondermek istedigimizde RS =0 olmalı burada

    e=1;
    delay(10);
    e=0;
}

void init()
{
    EA=1; // interuptin global olarak enable olmasini gosteren komut
    ES=1; // interuptin serial olarak enable olmasini gosteren komut
    TMOD =0X20; // timer Modlari secmemizde yardimci olan TMOD komutu. burada 8-
bit auto reload modu kullandik.
    SCON = 0X50; // 8bit data , 1 stop bit , REN enabled. serail port kullanara
veriler / sinyaller gonderim ve alimi yapar
    TH1 = -3;    // baud rate 9600 ulasmasini saglar.
    TR1 = 1;     //timer 1 olarak alip baslatiyor.
}
```



**b. Kullandığınız Timer ve Interrupt verileri, ne amaçla kullanıldığı**

```
for( i=0;i<4;i++);    //4 rakamli ( bit) bir ID girilmesi için for dongusu üzeri  
nden string sayisini kontrol edilir.  
    {  
        while (RI ==0); // interrupt komutunu enable ediyoruz burada. EA=1  
        sayesinde bu islem aktif hale geliyor  
                                // bufferden bir komut saklan  
masi gerektiğinde Reciever interrupt ( RI) = 1 olacak  
        a2[i]=SBUF; //stringlerin a2 dizisinde sakladigini kontrol ed  
iyor.  
        RI=0;    //saklam islemi gerceklestikten sonra RI = 1 yerine te  
krar 0 esit oluyor.  
    }  
}  
  
void init()  
{  
    EA=1; // interruptin global olarak enable olmasini gosteren komut  
    ES=1; // interruptin serial olarak enable olmasini gosteren komut  
    TMOD =0X20; // timer Modlari secmemizde yardimci olan TMOD komutu. burada 8 b  
it auto reload modu kullandik.  
    SCON = 0X50; // 8bit data , 1 stop bit , REN enabled. serail port kullanarak  
veriler / sinyaller gonderim ve alimi yapar  
    TH1 = -3; // baud rate 9600 ulasmasini saglar.  
    TR1 = 1; //timer 1 olarak alip baslatiyor.  
}
```

**c. Kullandığınız fonksiyonlar veya alt program çağrıları, ne amaçla yazıldığı**

```
void init ();    // baslama fonksiyonu  
  
void delay(int a)  
{  
    int i,j;  
    for(i=0;i<a;i++)  
        for(j=0;j<a;j++);  
}  
  
void lcd_cmd(int a)  
{
```

```

    P1=a;          //  parametre olarak a degerini alan P1 portu, cikti islemini go
ruyor. lcd den ciktilari aliyor
    rs=0;          //  LCD ye komut gondermek icin rs degeri =0 olmalı
    e=1;           //  enable high olacak
    delay(10);     //  enable high ve low arasında bir delay gerceklesecek.
    e=0;           //  enable low olck
}

void lcd_data(int a)
{
    P1=a;
    rs=1;  // LCD ye veri / komutlar gondermek istedigimizde RS =0 olmalı burada

    e=1;
    delay(10);
    e=0;
}

void lcd_string( char *str)
{
    while(*str)
    {
        lcd_data(*str);
        delay(100);
        str++;
    }
}

void main()
{
    int i;
    unsigned char a2[8]; // 8 boyut buyuklugune sahip bir dizi tanimliyoruz
    // baslatmak amaciyle LCD ye gonderilen komutlar
    lcd_cmd(0x38);      // 2 satir ve 5x7 matris boyunta verileri LCD ye aktariyo
r.
    lcd_cmd(0x0E);      //
    lcd_cmd(0x01);      // ekran temizleme icin kullanilir.
    lcd_cmd(0x80);      // ilk satir gosteren komutumuzdur.
    init();
    lcd_string("HOS GELDINIZ");
    lcd_cmd(0xC0); // ikinci satirdaki ilk cumleyi gosteren komut
    lcd_string("ID NO GIRINIZ");
    lcd_cmd(0x01); //
    while(1) // while 1 de iken, kosulun her daim dogru oldugu anlami verir.
        // bu da govdesinde kodlari kosulu sagalandigi

```

```

        // takdirde calistirmasini anlami veriyor
    {
        for( i=0;i<4;i++);    //4 rakamli ( bit) bir ID girilmesi icin for dongu
su üzerinden string sayisini kontrol edilir.
        {
            while (RI ==0); // interupt komutunu enable ediyoruz burada. EA=1
sayesinde bu islem aktif hale geliyor

                                // bufferden bir komut saklan
masi gerektiginde Reciever interupt ( RI) = 1 olacak
            a2[i]=SBUF; //stringlerin a2 dizisinde sakladigini kontrol ed
iyor.

            RI=0;    //saklam islemi gerceklestikten sonra RI = 1 yerine te
krar 0 esit oluyor.

        }

        if (strcmp (a2,"0513")!=0) //a2 dizisinde atanan bu deger , strcmp metho
du sayesinde karsilastirilir. dogru ise govdesindeki komutu donduruyor.
        {
            lcd_string ("Atiq Mayar");
            lcd_cmd(0x01);
            lcd_string ("iyi dersler");
            lcd_cmd(0xc0);
            lcd_string ("Kayitlandiniz");
            delay(200);
        }

        else if (strcmp (a2,"0517")==0)
        {
            lcd_string ("Morteza Yosefy");
            lcd_cmd(0x01);
            lcd_string ("iyi dersler");
            lcd_cmd(0xc0);
            lcd_string ("Kayitlandiniz");
            delay(200);
        }

        else if (strcmp (a2,"0516")==0)
        {
            lcd_string ("Reza");
            lcd_cmd(0x01);
            lcd_string ("iyi dersler");
            lcd_cmd(0xc0);
            lcd_string ("Kayitlandiniz");
            delay(200);
        }
    }

```

```

    }

    else {
        lcd_string("YANLIS GIRDINIZ");
        lcd_cmd(0x01);
    }

}

}

void init()
{
    EA=1; // interruptin global olarak enable olmasini gosteren komut
    ES=1; // interruptin serial olarak enable olmasini gosteren komut
    TMOD =0X20; // timer Modlari secmemizde yardimci olan TMOD komutu. burada 8 b
it auto reload modu kullandik.
    SCON = 0X50; // 8bit data , 1 stop bit , REN enabled. serail port kullanarak
veriler / sinyaller gonderim ve alimi yapar
    TH1 = -3; // baud rate 9600 ulasmasini saglar.
    TR1 = 1; //timer 1 olarak alip baslatiyor.
}

```

## 4. Doğrulama ve Test

### a. Projenizin kısıtları

#### RFID Tabanlı Katılım Sisteminin Kısıtlamaları

- ✚ RFID katılım sistemi güvenlidir, ancak kartların yanlış kullanım olasılığı vardır. RFID kartı varsa, bir kişi başka bir kişinin katılımını sağlayabilir.
- ✚ Kart birden fazla kez kaydırıldıysa, kodun doğru yazılmaması durumunda da sonraki günler için katılım hakkı kazanma şansı vardır.

### i. Projede neler yapıldı?

Oncelikle, proje kiel arm ortamini kullanarak C dilinde projeye ait kodu gelistirdik ve denedik. Ardindan c dosyasmizinn hex uzantisini kaydederek proteues uzzerinden daha once semasinizi cizdigimiz projenin calistirmasini sagladik.

Proje ilk, hosgeldin mesaji baslamasiyla beraber, daha once kaydolan ID numarali ogrencilerin girisiyle kayitli ve ya kayitsiz olarak adalandirliyor. Devrenin dogru ciktsi saglamak acısından timer ve interuptlar kullanılmıştır.

## **ii. Projenizin özgünlüğü ve pratik yaşama katkısı nedir?**

Ekstra masraf ve zaman kaybını ve daha çok şeffaflık getirmek açısından çok önemli olduğunu düşündüğümüz projenin, üniversite başta olmak üzere bir sürü kurumlara denetleme ve işlerin daha hızlı olmasında yardımcı oluyor.

## **iii. Projenin eksiklikleri neler?**

Gerçek bir ortamda gerçekleştiremediğimiz projemizin ana eksiklerinden biri RFID çiplerin olmaması idi. Bununla birlikte bazı devrelerin eksikliğinden de bahsedebiliriz.

- AT89C51 Mikrodenetleyici
- AT89C51 Programlama Kartı
- 11.0592 MHz Kuvars Kristali
- 2 x 33pF Seramik Kapasitör
- 2 x 10K $\Omega$  Direnç
- 10 $\mu$ F Elektrolitik Kondansatör
- 2 x Basma Düğmesi
- 16 x 2 LCD Ekran
- 3 x 1K $\Omega$  Direnç
- 10K $\Omega$  TENCERE
- EM-18 RFID Okuyucu Modülü
- RFID Etiketleri veya Kartları
- Bağlantı Telleri

## **iv. Projede olması gereken diğer durumlar**

