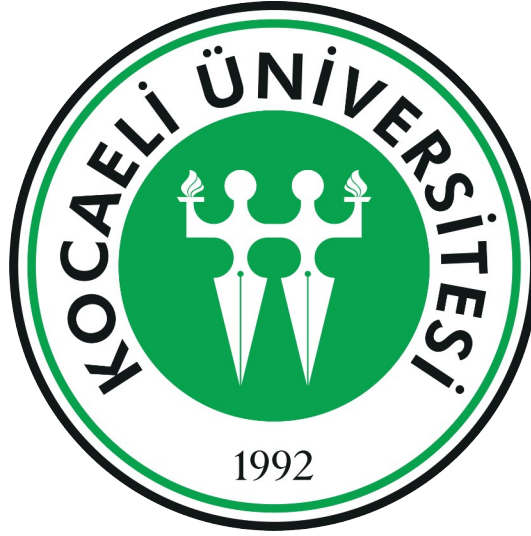


KOCAELİ ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ
BİLİŞİM SİSTEMLERİ MÜHENDİSLİĞİ BÖLÜMÜ



PROJE A RAPORU
IoT Uygulaması
ÖĞRENCİ YOKLAMA SİSTEMİ

Onur AKYILDIZ
PROF. DR. MEHMET YILDIRIM
KOCAELİ 2023

ÖNSÖZ VE TEŞEKKÜR

Bu çalışma Kocaeli Üniversitesi, Teknoloji Fakültesi, Bilişim Sistemleri Mühendisliği Programı, Proje A dersi kapsamında gerçekleştirilmiştir.

Çalışmamda desteğini esirgemeyen, çalışmalarına yön veren, bana güvenen ve yüreklendiren, çalışmamın tüm aşamalarında bilgi ve destekleriyle katkıda bulunan hocam, danışmanım, Mehmet Yıldırım'a sonsuz teşekkürlerimi sunarım.

Hayatım boyunca bana güç veren en büyük destekçilerim, her aşamada sıkıntılarımı ve mutluluklarımı paylaşan sevgili aileme teşekkürlerimi sunarım.

Ocak – 2023

Onur AKYILDIZ

İÇİNDEKİLER

ÖZET	1
ABSTRACT.....	1
1. Kullanılan Teknolojiler Hakkında Bilgiler.....	2
1.1 Arduino.....	2
1.1.1 Arduino Nedir.....	2
1.1.2 Arduino Avantajları.....	2
1.1.3 Arduino Dezavantajları.....	3
1.1.4 Arduino Uno.....	3
1.2 MFRC522 RFID Okuyucu.....	4
1.2.1 Özellikleri.....	4
1.2.2 Çalışma Prensipleri.....	5
1.3 Python.....	6
1.3.1 Python Nedir.....	6
1.3.2 Python Avantajları.....	6
1.3.3 Python Dezavantajları.....	7
1.4 JavaScript.....	8
1.4.1 JavaScript Nedir.....	8
1.4.2 JavaScript Avantajları.....	8
1.4.3 JavaScript Dezavantajları.....	8
2. Projenin Adımları.....	9
2.1 RFID Karta Veri Yazma.....	9
2.1.1 MFRC522 RFID Okuyucu Bağlantıları.....	9
2.1.2 Dump Info ile Bloklardaki Verileri Gösterme.....	10
2.1.3 Bloklara Verileri Yazma.....	11
2.2 RFID Karttan Verileri Okuma.....	12
2.2.1 Bloklardan Verileri Okuma.....	12
2.3 Python'a Verileri Gönderme.....	12
2.3.1 Python'a Veriyi Gönderme.....	12
2.3.2 Python'a Gelen Veriyi İşleme.....	13
2.3.3 Düzenlenen Verinin Aratılması.....	13
2.4 JavaScript Kodları.....	14

2.4.1 Gerekli Değişkenlerin Tanımı.....	14
2.4.2 Gelen Verilerin Ayrılması.....	14
2.4.2.1 Veriler Nereden Geliyor?.....	14
2.4.3 Verilerin Yerleştirilmesi.....	14
3. Karşılaşılan Zorluklar ve Çözümleri.....	15
4. Kaynakça.....	15

Bu dokümandaki tüm bilgiler, akademik kurallar çerçevesinde elde edilip sunulmuştur. Ayrıca yine bu kurallar çerçevesinde kendime ait olmayan başka kaynaklardan elde edilen bilgiler ve materyaller (resim, şekil, tablo vb.) gerekli şekilde referans edilmiş ve dokümanda belirtilmiştir.

Öğrenci No: 191307026

Adı Soyadı: Onur AKYILDIZ

İmza:

Öğrenci Yoklama Sistemi IoT Uygulaması

ÖZET

Bu proje C++, Python, JavaScript dilleri kullanılarak Arduino IDE ve PyCharm geliştirme ortamları kullanılarak gerçekleştirilmiştir. Projenin amacı derslerde yoklama alınırken yaşanan karmaşıklığı ve zorluğu nesnelerin internetini kullanarak gidermektir. Proje 4 adımdan oluşmaktadır, RFID karta verileri yükleme, RFID karttaki bilgileri okuma, portlardan PyCharm geliştirme ortamına iletilmesi ve verilerin Google Sheets'te hangi hücrelere geleceğinin ayarlanmasıdır.

Anahtar Kelimeler: Arduino IDE, Python, PyCharm, Google Sheets, RFID, Arduino UNO, Yoklama Sistemi, JavaScript, IoT.

Student Attendance System IoT Application

ABSTRACT

This project has been done using Arduino IDE and PyCharm IDE using C++, Python, JavaScript languages. The aim of the project is to eliminate the complexity and difficulty experienced while taking attendance in classes by using the Internet of Things. The project consists of 4 steps, uploading the data to the RFID card, reading the information on the RFID card, transmitting the data from the ports to the PyCharm IDE and setting which cells the data will come to in Google Sheets.

Key Words: Arduino IDE, Python, PyCharm, Google Sheets, RFID, Arduino UNO, Student Attendance System, JavaScript, IoT.

1. Kullanılan Teknolojiler Hakkında Bilgiler

1.1 Arduino

1.1.1 Arduino Nedir?

Arduino bir G/Ç kartı ve Processing/Wiring dilinin bir uygulamasını içeren geliştirme ortamından oluşan bir fiziksel programlama platformudur.

Arduino kartlarının donanımında bir adet Atmel AVR mikrodeneleyici (ATmega328, ATmega2560, ATmega32u4 gibi) ve programlama ve diğer devrelere bağlantı için gerekli yan elemanlar bulunur. Her Arduino kartında en azından bir 5 voltluk regüle entegresi ve bir 16MHz kristal osilatör (bazılarında seramik rezonatör) vardır. Arduino kartlarında programlama için harici bir programlayıcıya ihtiyaç duyulmaz, çünkü karttaki mikrodeneleyiciye önceden bir bootloader programı yazılıdır.

1.1.2 Arduino Avantajları

Arduino aynı zamanda açık kaynak kodludur. Yani devre şemasından program ara yüzüne kadar istediğiniz projede (ticari olarakta) kullanıp yeni ürünler geliştirebilirsiniz. Hatta kendinize ait bir Arduino üretebilirsiniz. Arduino, elektronik ve programlama konularında iyi derecede bilgi birikimi olmadan ilgili ve meraklı olan birçok kişinin kolaylıkla kullanabileceği bir yapıya sahiptir.

Arduino'nun başlıca avantajları;

- Üzerinde Atmel marka mikrodeneleyici vardır ve gerekli bağlantıları(Osilatör, +5V besleme, Reset butonu gibi) yapılmış halde kullanıma hazırdır.
- USB ile bilgisayara doğrudan bağlantısı yapılabilir.
- Kitlere kolay şekilde adapte edilebilen, eklenti(shield) adı verilen devreler piyasada yaygın olarak bulunmaktadır.
- Çevresel modüllerle(Shield) birlikte devre kurmayı daha basit hale getirmektedir.
- Mikrodeneleyiciler için gerekli olan programlayıcıya ihtiyaç duymamaktadır.
- Programlaması diğer geliştirme kitlerine nazaran daha anlaşılır ve kolaydır.
- Geniş kütüphane desteği ile çok karmaşık projeleri bile mümkün olduğunca sadeleştirmiştir.
- Maliyeti benzer sistemlere göre daha uygundur.
- Piyasada kolay bulunabilir bir üründür.

1.1.3 Arduino Dezavatajları

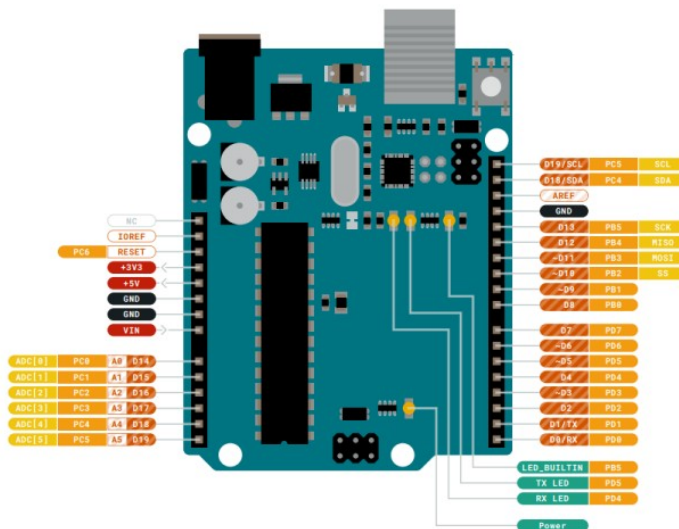
Saniyede birkaç milyon işlem kapasitesine sahip bir mikrodenetleyici ile çevre üniteler kontrol edilmekte ve dış dünya ile etkileşim sağlanmaktadır. Dolayısıyla devrenin bağlantıları yapılırken dikkatli olunmadığı takdirde bazı zararlar kaçınılmaz olabilir. Aynı şekilde yazdığımız programdan kaynaklanan hatalarda muhtemeldir. 32Kb'lık hafızası ile yapabileceklerimiz sınırlıdır

1.1.4 Arduino UNO

Arduino Uno ile çeşitli sensörlerden fiziksel bilgi alabilir, bu bilgiler ile çeşitli deneyler yapabilirsiniz. Ayrıca motor, LED, buzzer gibi uyarıcılardan bir çıktı elde edebilirsiniz. Bu gibi elektronik komponentleri Arduino Uno kartına bağlayarak kontrol etmek için temel bir programlama bilgisi yeterlidir. Projelerin seviyesine göre gerekli olan elektronik ve programlama bilgisi seviyesi de artacaktır.

Arduino Uno'nun genel anlamda özellikleri aşağıdaki gibidir:

- Mikrodenetleyici: ATmega328
- Çalışma Gerilimi: 5V
- Dijital G/Ç Pinleri: 14 (6 tanesi PWM çıkışı)
- Analog Giriş Pinleri: 6 (10 bitlik 0 – 1024 arası değerler)
- Flash Hafıza: 32 KB (ATmega328)
- SRAM: 2 KB (ATmega328)
- EEPROM: 1 KB (ATmega328)



Şekil 1: Arduino UNO Pin Çıkışları

1.2 MFRC522 RFID Okuyucu

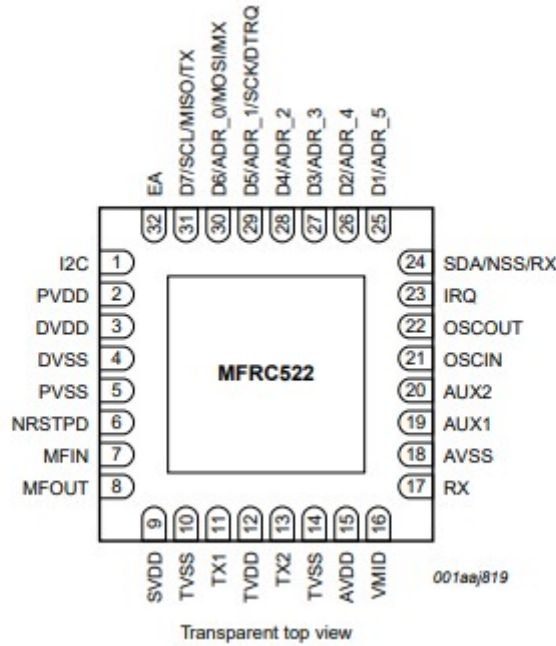
1.2.1 Özellikleri

NFC frekansı olan 13,56 MHz frekansında çalışan tagler üzerinde okuma ve yazma işlemi yapabilen, düşük güç tüketimli, ufak boyutlu bir karttır.

Arduino başta olmak üzere bir çok mikrodenetleyeci platformu ile beraber rahatlıkla kullanılabilir. 424 kbit/s haberleşme hızına sahiptir. RFID üzerinde farklı şifreleme türlerini desteklemektedir. Desteklediği kart türleri mifare1 S50, mifare1 S70 mifare ultralight, mifare pro ve mifare desfire'dir.

MFRC522 RFID Okuyucu Kartı Özellikleri:

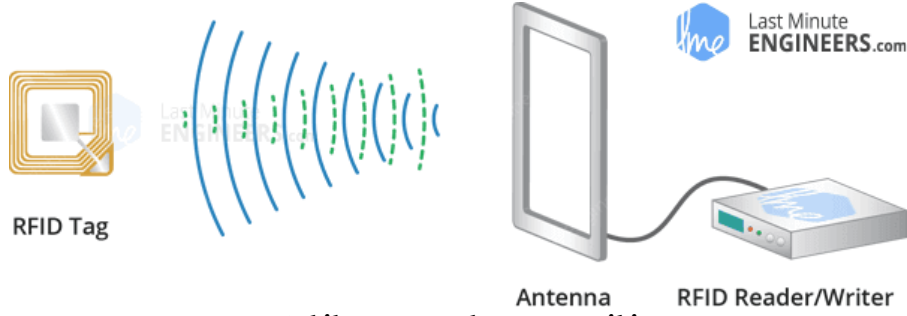
- Çalışma Gerilimi: 3,3V
- Çalışma Frekansı: 13,56 MHz
- Çalışma Akımı: 13-26mA
- Uyku Akımı: <80 uA
- Haberleşme Protokolü: SPI
- Desteklenen Kartlar: mifare1 S50, mifare1 S70 mifare ultralight, mifare pro ve mifare desfire
- Kart Boyutları: 40x60mm



Şekil 2: MFRC522 Pin çıkışları

1.2.2 Çalışma Prensibi

RFID veya Radyo Frekansı Tanımlama sistemi iki ana bileşenden oluşur: tanımlanacak bir nesneye bağlı bir aktarıcı / etiket ve sorgulayıcı / Okuyucu olarak da bilinen bir Alıcı-Verici.



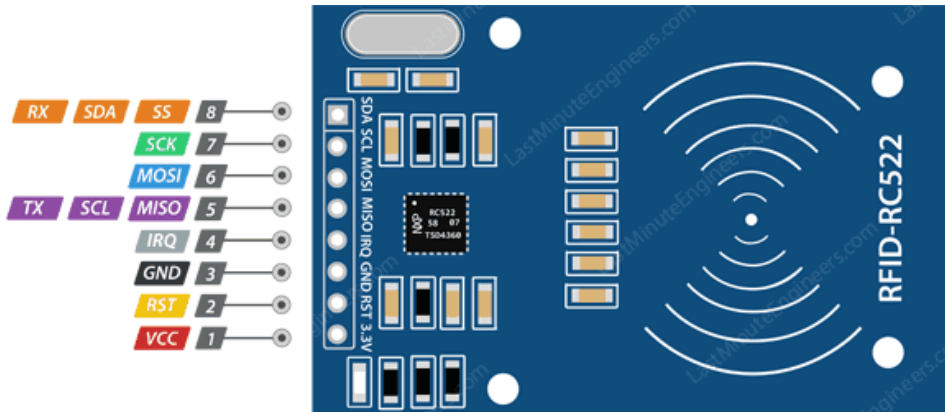
Şekil 3: RFID çalışma prensibi

RFID okuyucu bir Radyo Frekansı modülü ve yüksek frekanslı elektromanyetik alan üreten bir antenden oluşur. Rfid etiket aslında pasif bir rfid cihazdır. Enerjisiz çalışmasına en olanak tanıyan özelliği bilgiyi depolayan ve işleyen bir mikroçip ve bir sinyal almak ve iletmek için bir anten içermesidir.

Bir etikette kodlanan bilgileri okumak için, okuyucunun menziline etiket yerleştirilir (okuyucunun doğrudan görüş alanı içinde olması gerekmez). Okuyucu, elektronların etiketin anteninden geçmesine ve daha sonra çipe güç vermesine neden olan bir elektromanyetik alan oluşturur.

Etiketin içindeki güç yongası daha sonra saklanan bilgilerini başka bir radyo sinyali biçiminde okuyucuya geri göndererek yanıt verir. Buna backscatter denir. RF dalgasındaki değişiklik, daha sonra verileri bir bilgisayara veya mikrodenetleyiciye gönderen okuyucu tarafından algılanır ve yorumlanır.

RFID etiketleriyle (ISO 14443A standart etiketleri) iletişim kurmak için 13.56MHz elektromanyetik alan oluşturur. Okuyucu, maksimum 10Mbps veri hızına sahip 4 pimli Seri Çevresel Arabirim (SPI) üzerinden bir mikro denetleyici ile iletişim kurabilir. Ayrıca I2C ve UART protokolleri üzerinden de iletişimi destekler. Modülün çalışma voltajı 2.5 ila 3.3 V arasındadır, ancak iyi haber logic pimlerinin 5 volt toleranslı olmasıdır.



Şekil 4: MRFC522 pin çıkışları

1.3 Python

1.3.1 Python Nedir

1991'den beri Python programlama dili sadece gereksiz programlar için tamamlayıcı bir dil olarak değerlendiriliyordu. Hatta “Automate the Boring Stuff” (Türkçe'ye "Sıkıcı Şeyleri Otomatikleştiren" olarak çevirebileceğimiz popüler bir kitap) adında bir kitap dahi yayınlanmıştır. Bununla birlikte son birkaç yılda Python modern yazılım geliştirme, altyapı yönetimi ve veri analizinde birinci sınıf bir programlama dili olarak ön plana çıkmıştır. Artık hackerlar için bir arka kapı oluşturucusu değil, web uygulaması oluşturma ve sistem yönetiminde önemli rol alma, veri analizleri ve makine öğreniminde parlayan bir dil olarak ün kazanmıştır.

Diğer karmaşık program dillerini öğrenmek çok zaman alır ve kullanım alanları büyük olmasından dolayı kullanımını öğrenmek çok zordur. Ancak Python sözdizimi hem okunabilir hem de ileriye dönüktür. Öğrenim, kararlı programlama dili sayesinde basittir. Yeni başlayanlar için de ideal bir seçim olarak ön plana gelir. Sonuç olarak, Python kullanarak program geliştirmeye yeni bir adım atmış olan herkes hızlı ve basit şekilde ilerleme kaydedebilir. Diğer karmaşık dillere göre basitlik söz konusu olduğunda, en önde yer almaktadır.

İsminin az duyulmasına rağmen, yazılımcılar arasında hem popülerdir hem de yaygındır. Github projelerinin birçoğu Python tabanlıdır. Hatta Tiobe Index ve Github gibi programlama örneklerinin sıralandığı sistemlerde, Python tabanlı programları en üst sıralarda göstermektedir. Kullanım alanı en küçük işletim sistemlerinden en büyük işletim sistemlerine kadar birçok OS tarafından desteklenmektedir. İnternette yer alan büyük yazılım kütüphanelerin ve API-destekli servislerin mutlaka Python tabanlı bindings ya da wrappers'ları vardır. Bundan dolayı Python, bu servisler ile serbestçe arayüz oluşturur veya bu kütüphaneleri doğrudan kullanır. Yazılım dilleri arasında en hızlısı olmasa da çoklu platformlarda kullanılması bir adım önde olmasını sağlamıştır.

1.3.2 Python Avantajları

- Okuması, Öğrenmesi ve Yazması Kolay

Python, İngilizce benzeri sözdizimine sahip üst düzey bir programlama dilidir . Bu, kodun okunmasını ve anlaşılmasını kolaylaştırır.

- Geliştirilmiş Verimlilik

Python'un basitliği nedeniyle geliştiriciler sorunu çözmeye odaklanabilir. Programlama dilinin sözdizimini veya davranışını anlamak için çok fazla zaman harcamaları gerekmez

- Yorumlanan Dil

Python'un kodu satır satır doğrudan yürüttüğü anlamına gelir . Herhangi bir hata durumunda, daha fazla çalışmayı durdurur ve oluşan hatayı geri bildirir.

Python, programın birden fazla hatası olsa bile yalnızca bir hata gösterir. Bu, hata ayıklamayı kolaylaştırır .

- Dinamik Olarak Yazılan Bir Dil

Yürütme sırasında veri türünü otomatik olarak atar . Programcının değişkenleri ve veri türlerini bildirmek konusunda endişelenmesine gerek yoktur.

- Ücretsiz ve Açık Kaynak

Python, OSI onaylı açık kaynak lisansı altında gelir . Bu yapar özgür için kullanmak ve dağıtmak . Kaynak kodunu indirebilir, değiştirebilir ve hatta Python sürümünüzü dağıtabilirsiniz. Bu, bazı belirli davranışları değiştirmek ve sürümlerini geliştirme için kullanmak isteyen kuruluşlar için kullanışlıdır.

- Geniş Kitaplıklar Desteği

Python'un standart kütüphanesi çok büyüktür, göreviniz için gereken hemen hemen tüm fonksiyonları bulabilirsiniz. Bu nedenle, harici kütüphanelere bağımlı olmanız gerekmez.

- Taşınabilirlik

C/C++ gibi birçok dilde , programı farklı platformlarda çalıştırmak için kodunuzu değiştirmeniz gerekir . Python'da durum aynı değil. Yalnızca bir kez yazıp her yerde çalıştırırız.

1.3.3 Python Dezavantajları

- Yavaş Hız

Kodun satır satır yürütülmesi genellikle yavaş yürütmeye neden olur .Python'un dinamik doğası, Python'un yavaş hızından da sorumludur, çünkü kodu yürütürken fazladan iş yapması gerekir.

- Bellek Verimli Değil

Python programlama dili büyük miktarda bellek kullanır . Bellek optimizasyonunu tercih ettiğimizde uygulamalar oluştururken bu bir dezavantaj olabilir.

- Mobil Bilişimde Zayıf

Python genellikle sunucu tarafı programlamada kullanılır. Python bellek açısından verimli değildir ve diğer dillere kıyasla yavaş işlem gücüne sahiptir.

- Veritabanı Erişimi

Python'da programlama yapmak kolay ve stressizdir . Ancak veritabanı ile etkileşime girdiğimizde, geride kalıyor.Python'un veritabanı erişim katmanı, JDBC ve ODBC gibi popüler teknolojilere kıyasla ilkel ve az gelişmiştir .Büyük kuruluşlar , karmaşık eski verilerin sorunsuz etkileşimine ihtiyaç duyar ve bu nedenle Python, kuruluşlarda nadiren kullanılır.

- Çalışma Zamanı Hataları

Python'un dinamik olarak yazılan bir dil olduğunu bildiğimiz gibi, bir değişkenin veri tipi her an değişebilir. Tamsayı içeren bir değişken gelecekte bir dize tutabilir ve bu da Çalışma Zamanı Hatalarına yol açabilir.

1.4 JavaScript

1.4.1 JavaScript Nedir

JavaScript, geliştiricilerin etkileşimli web sayfaları oluşturmak için kullandığı bir programlama dilidir. JavaScript işlevleri, sosyal medya akışlarını yenilemekten animasyonlar ve etkileşimli haritalar göstermeye kadar, bir web sitesi kullanıcısının deneyimini iyileştirebilir. İstemci tarafındaki bir betik dili olarak, World Wide Web'in temel teknolojilerinden biridir. Örneğin, internette gezinirken bir görsel döngüsü, görmek için tıkla açılır menüsü ya da bir web sayfasında dinamik olarak değişen öğe renkleri gördüğünüzde JavaScript efektlerini görmüş olursunuz.

1.4.2 JavaScript avantajları

JavaScript diğer programlama dilleri ile kıyaslandığında birçok avantajı olduğunu söyleyebiliriz. JavaScript avantajlarından bazıları ise şunlardır:

- Derleyici kullanmanıza gerek yoktur çünkü web tarayıcıları HTML ile yorumlamaktadır.
- Öğrenmesi kolaydır, diğer yazılım dillerinden daha hızlı öğrenilir.
- Hataları tespit etmek ve çözmek daha kolaydır.
- Belirli web sayfası öğelerine göre ayarlanabilmektedir.
- Birden fazla platformda ya da tarayıcıda çalışabilmektedir.
- JavaScript kullanarak input değerlendirmesi yapılabilir.
- Manuel very kontrollerinin azaltılmasını sağlamaktadır.
- Web siteleri interaktif hale getirir, kullanıcı dostudur.
- Diğer yazılım dilleri ile kıyaslandığında daha hızlı ve hafif olduğu görülmektedir.

1.4.3 JavaScript dezavantajları

Yazımızın giriş bölümünde de belirttiğimiz gibi günümüz web sitelerinin neredeyse yüzde 90'ı JavaScript kullanmaktadır. Bu kadar yaygın olması kötü niyetli kişi ve uygulamaların da hedefi haline gelmesine sebep olmaktadır. Güvenlik açıklarını tespit eden kötü niyetli kişiler nedeniyle de karşımıza bazı JavaScript dezavantajlarını çıkarmaktadır. JavaScript'in dezavantajlarını şu şekilde sıralayabiliriz:

- Güvenlik açıklara karşı zayıftır.
- Zararlı kod çalıştırmak için bilgisayarlarda kullanılabilir.
- Her tarayıcı ve cihaz tarafından desteklenmemektedir.
- JavaScript kodları diğer yazılım dillerine göre büyüktür.
- Farklı cihazlarda çalıştırılması tutarsızlık yaratmaktadır.

2. Projenin Adımları

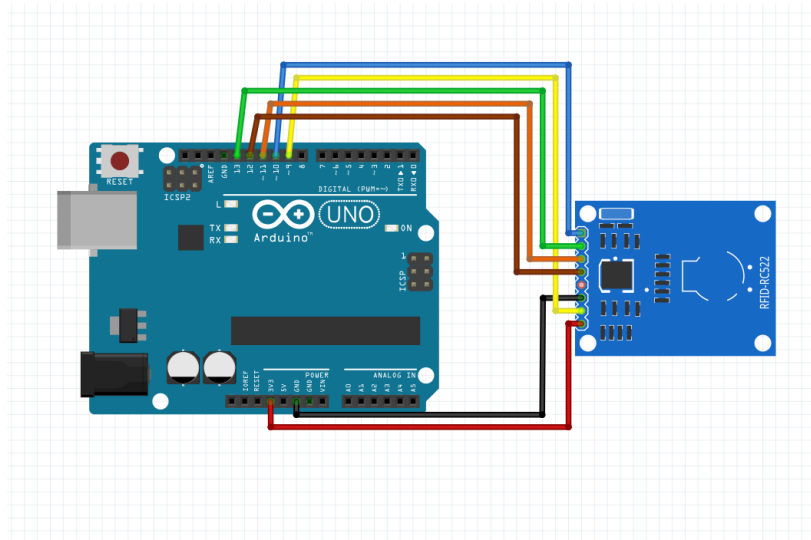
2.1 RFID Karta Veri Yazma

2.1.2 MFRC522 RFID Okuyucu Bağlantıları

MFRC522 RFID okuyucu modülü 3.3V ile beslenir Arduino Uno üzerindeki 3.3V hattına bağlamamız gerekir aksi takdirde modülü yakabiliriz. Arduino Unoda 14 adet dijital pin mevcuttur RFID modülü için 5 tane dijital pin bağlantısı mevcuttur 9, 10, 11, 12 ve 13 dijital pinlere bağlantı yapılır 11 12 ve 13 sabittir tabloda gözüktüğü gibi RST(reset) hattı ve SS hattını istediğimiz dijital pine bağlayabiliriz kodda belirtmek şartı ile.

Typical pin layout used:

	MFRC522 Reader / PCD	Arduino Uno / 101	Arduino Mega	Arduino Nano v3	Arduino Leonardo/Micro	Arduino Pro Micro
Signal	Pin	Pin	Pin	Pin	Pin	Pin
RST/Reset	RST	9	5	D9	RESET/ICSP-5	RST
SPI SS	SDA(SS)	10	53	D10	10	10
SPI MOSI	MOSI	11 / ICSP-4	51	D11	ICSP-4	16
SPI MISO	MISO	12 / ICSP-1	50	D12	ICSP-1	14
SPI SCK	SCK	13 / ICSP-3	52	D13	ICSP-3	15



Şekil 5: Bağlantı Şeması

2.1.2 Dump Info ile Bloklardaki Verileri Gösterme

Dump Info kısaca kart içindeki verileri gösterir, kart içindeki veriler bit şeklinde tutar, toplamda 64 adet blok mevcuttur, bu blokların 16 tanesi kartın UID'sini barındırır geri kalan 48 tane Bloka 16 bitlik veri yazılabilir. Resim 6 ve 7 de gözükten mavi hatlar UID'yi barındırır kırmızı hatlar ise yazılabilir hatlardır. 4, 5, 6, 8 ve 9 bloklara veriler yazıldı

Sector	Block	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	AccessBits
15	63	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	62	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	61	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
14	59	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	58	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	57	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	56	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
13	55	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	54	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	53	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	52	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
12	51	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	49	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	48	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
11	47	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	46	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	45	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	44	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
10	43	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	42	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	41	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
9	39	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	38	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	37	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	36	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
8	35	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	34	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	33	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	32	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
7	31	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	29	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	28	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
6	27	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	26	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	25	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	24	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
5	23	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	22	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	21	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
4	19	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	18	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	17	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	16	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
3	15	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	14	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	13	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	12	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
2	11	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	9	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	8	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
1	7	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	6	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	5	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
0	3	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	2	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	1	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	0	50	00	40	1D	0D	08	04	00	62	63	64	65	66	67	68	69	[000]

Şekil 6: kartın içine veriler yazılmadan önce

Sector	Block	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	AccessBits
15	63	00	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	[001]
	62	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	61	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
14	59	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	58	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	57	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	56	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
13	55	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	54	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	53	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	52	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
12	51	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	49	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	48	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
11	47	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	46	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	45	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	44	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
10	43	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	42	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	41	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
9	39	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	38	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	37	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	36	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
8	35	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	34	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	33	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	32	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
7	31	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	29	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	28	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
6	27	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	26	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	25	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	24	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
5	23	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	22	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	21	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
4	19	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	18	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	17	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	16	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
3	15	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	14	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	13	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	12	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
2	11	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	9	42	69	6C	20	53	69	73	20	4D	3C	BC	68	20	20	20	20	[000]
	8	54	65	6B	6E	6F	6C	6F	6A	69	20	66	61	6B	20	20	20	[000]
1	7	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	FF	[001]
	6	4B	6F	63	61	65	6C	69	20	55	6E	69	20	55	6E	69	20	[000]
	5	4F	6E	75	72	20	41	6B	79	C4	B1	6C	64	C4	B1	7A	20	[000]
	4	31	39	31	33	30	37	30	32	36	20	30	32	36	20	20	20	[000]
0	3	00	00	00	00	00	00	00	FF	07	80	69	FF	FF	FF	FF	FF	[001]
	2	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	1	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	[000]
	0	50	40	4D	0D	08	04	00	62	63	64	65	66	67	68	69		[000]

2.1.3 Bloklara Verileri Yazma

MFRC522.h kütüphanesi yardımı ile DumpInfo örneğinden kartın veri kaydetme blokları görüntülendi şekil 7 da yer aldığı gibi veri blokları olan 4. 5. 6. 8. 9. bloklara veriler yazıldı 7. bloka yazılmama sebebi görselde gözüktüğü gibi mavi alanlar kartın UID'sini barındırır, bu kimliği değiştirmek kartı bozabilir.

```
#include <SPI.h>
#include <MFRC522.h>

//GPIO 0 --> D3
//GPIO 2 --> D4
const uint8_t RST_PIN = 9;
const uint8_t SS_PIN = 10;

MFRC522 mfrc522(SS_PIN, RST_PIN);
MFRC522::MIFARE_Key key;

/* blok numara başlangıcı */
int blockNum = 4;
byte block_data[16];
/* karttaki verileri dizi şeklinde okuyup yazma için */
byte bufferLen = 18;
byte readBlockData[18];

MFRC522::StatusCode status;
```

Şekil 8: kütüphaneler ve pinlerin tanımlanması

```
void setup()
{
  /* seri haberleşme */
  Serial.begin(9600);
  SPI.begin();

  mfrc522.PCD_Init();
  Serial.println("kartı okutup işlem bitene kadar kaldırmayınız ...");
}

void loop()
{
  /* doğrulama */
  for (byte i = 0; i < 6; i++){
    key.keyByte[i] = 0xFF;
  }

  /* yeni kart tarama */
  /* yeni kart okutulunca reset işlemi */
  if ( ! mfrc522.PICC_IsNewCardPresent() ){return;}

  /* okunan kartı seçme */
  if ( ! mfrc522.PICC_ReadCardSerial() ){return;}
  Serial.print("\n");
  Serial.println("***kart algılandı***");

  /* kart UID */
  Serial.print(F("Kart UID:"));
  for (byte i = 0; i < mfrc522.uid.size; i++){
    Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");
    Serial.print(mfrc522.uid.uidByte[i], HEX);
  }
  Serial.print("\n");

  /* kart tipi */
  Serial.print(F("PICC tip: "));
  MFRC522::PICC_Type piccType = mfrc522.PICC_GetType(mfrc522.uid.sak);
  Serial.println(mfrc522.PICC_GetTypeName(piccType));
}
```

Şekil 9: yeni kart tarama ve kart UID'sini yazdırma

```
void WriteDataToBlock(int blockNum, byte blockData[])
{
  Serial.println("bloklara veri yazılıyor ... ");
  Serial.println("blok numarası : "+blockNum);
  /* yazma erişimi için istenen veri bloğunun kimliğinin doğrulanması */
  status = mfrc522.PCD_Authenticate(MFRC522::PICC_CMD_MF_AUTH_KEY_A, blockNum, &key, &(mfrc522.uid));
  if (status != MFRC522::STATUS_OK)
  {
    Serial.println("başarısız: ");
    Serial.println(mfrc522.GetStatusCodeName(status));
    return;
  }
  else
  {
    Serial.println("başarılı");
  }

  /* bloka verileri yazma */
  status = mfrc522.MIFARE_Write(blockNum, blockData, 16);
  if (status != MFRC522::STATUS_OK)
  {
    Serial.println("başarısız: ");
    Serial.println(mfrc522.GetStatusCodeName(status));
    return;
  }
  else
  {
    Serial.println("yazma başarılı");
  }
}
```

Şekil 10: yazma fonksiyonu

2.2 RFID Karttan Verileri Okuma

2.2.1 Bloklardan Verileri Okuma

Okuma işleminin gerçekleşeceği bloklar tanımlandı, gerekli atamalar yapıldı. Kart tarama işlemi gerçekleştirildi, veriler belirtilen bloklardan okundu ve python'a gönderilecek veriler birleştirildi.

```
String values = "", data;

//url oluşturma
for (byte i = 0; i < total_blocks; i++)
{
    ReadDataFromBlock(blocks[i], readBlockData);
    if (i == 0)
    {
        data = String((char*)readBlockData);
        data.trim();
        student_id = data;
        values = "\"" + data + ",";
    }
    else if (i == total_blocks-1)
    {
        data = String((char*)readBlockData);
        data.trim();
        values += data + "\"";
    }
    else
    {
        data = String((char*)readBlockData);
        data.trim();
        values += data + ",";
    }
}
Serial.println(values);
```

Şekil 12: Bloklardan gelen verilerin birleştirilmesi

```
void ReadDataFromBlock(int blockNum, byte readBlockData[])
{
    for (byte i = 0; i < 6; i++)
    {
        key.keyByte[i] = 0xFF;
    }
    /* Okuma erişimi için istenen veri bloğunun kimliğinin doğrulanması */
    status = mfrc522.PCD_Authenticate(MFRC522::PICC_CMD_MF_AUTH_KEY_A, blockNum, skey, s(mfrc522.uid));

    if (status != MFRC522::STATUS_OK)
    {
        //Serial.print("okuma hatası: ");
        //Serial.println(mfrc522.GetStatusCodeName(status));
        return;
    }
    else
    {
        //Serial.println("okuma başarılı");
    }
    /* bloktan veri okuma */
    status = mfrc522.MIFARE_Read(blockNum, readBlockData, sbufferLen);
    if (status != MFRC522::STATUS_OK)
    {
        //Serial.print("okuma hatası: ");
        //Serial.println(mfrc522.GetStatusCodeName(status));
        return;
    }
    else
    {
        readBlockData[16] = ' ';
        readBlockData[17] = ' ';
        //Serial.println("başarılı");
    }
}
```

Şekil 11: belirtilen bloklardan verileri okuma

2.3 Python'a Verileri Gönderme

2.3.1 Python'a Veriyi Gönderme

Python'a veriyi gönderme işleminde bi gönderim değilde alım kullanılmaktadır. Arduinolara bilgisayara USB portundan bağlanır enerjiyi ve bilgileri burdan alır ve iletir iletirken python Serial kütüphanesi ile porttan gelen veriyi aldım.

```
import serial
import webbrowser

#veriyi alma
uno = serial.Serial("com3", 9600)
x = 0

while x != 1:
    while uno.inWaiting() == 0:
        pass
    data = uno.readline()
    data = str(data, 'utf-8')
    print("gelen data : ", data)
    x += 1
```

Şekil 13: porttaki veriyi alma

2.3.2 Python'a Gelen Veriyi İşleme

Gelen veri kullanılamaz halde olduğundan istenmeyen karakterleri çıkartıp diziye aktardım,

Gelen veri: "191307026 ,Onur Akyildiz ,Kocaeli Uni ,Teknoloji fak ,Bil Sis Muh }

```
#veriyi isleme
data = data.split(",")

print("ayrılmış data : ", data)

for i in range(len(data)):
    clean = data[i].replace(" ", "").replace("\",", "").replace("NAKSOH", "").replace("}", "")
    print("data tek tek : ", clean)
    urlList.append(clean)

for i in range(len(urlList)):
    FullURL = urlPart + str(i+1) + "=" + urlList[i]
    print(FullURL)
    finalURL.append(FullURL)
print(finalURL)
```

Şekil 14: gelen verileri işlenmesi

Aralardaki virgüllerden “split();” komutu ile ayırdım ve istenmeyen karakterleri “replace();” komutu ile yer değiştirdim sonrasında listeye aktardım. İkinci for döngüsünde aratılacak bağlantının bi kısmını oluşturdum.

2.3.3 Düzenlenen Verinin Aratılması

Google bize JAVASCRIPT ile yazdığımız .gs dosyasını dağıttığımız zaman bi key ve bağlantı veriyor.

```
URL = "https://script.google.com/macros/s/AKfycbw16NxQF41e3Rs5fYDBSrBsXPtYWQQ3MFgD90YifYNYbcyyX0CasxAabQuiRgYe_LZaPA/exec?"
Key = "AKfycbw16NxQF41e3Rs5fYDBSrBsXPtYWQQ3MFgD90YifYNYbcyyX0CasxAabQuiRgYe_LZaPA"
#?data1=191307026&data2=OnurAkyildiz&data3=KocaeliUni&data4=Teknolojifak&data5=BilSisMüh
```

Şekil 15: verilen URL ve KEY

Verilen key ve url şekil 15 te gösterilmektedir bu verilen url ile birleştirmiş olduğumuz url(öğrenci bilgileri olan) birleştirilip aratıldığı zaman tabloya öğrenci bilgileri gidiyor, aslında şekil 14 te yapmış olduğumuz şekil 15 teki yorum satırını elde etmek oldu

```
for i in range(len(finalURL)):
    print(finalURL[i], end="")
    strURL = strURL + finalURL[i] + "&"

strURL = strURL.replace("_", "")
print("str : ", strURL)
URL = URL + strURL
print("final URL : ", URL)
# web browser ilw arama
webbrowser.open(URL)
```

Şekil 16: URL'nin Webbrowser kütüphanesi ile aratılması

2.4 JavaScript Kodları

2.4.1 Gerekli Değişkenlerin Tanımı

Belgenin ID'si belgedeki hangi sayfada işlem yapılacağı ve tarih bilgilerini doğru almak için bölge tanımlaması yapıldı

```
var ss = SpreadsheetApp.openById('1wmysnUqHb3umEI33FCf6IpF9pqWWBv-jqy-DTqKr_eQ');
var sheet = ss.getSheetByName('Sayfa1');
var timezone = "europe/istanbul"
var hours = 3
```

Şekil 17: tanımlamalar

2.4.2 Gelen Verilerin Ayrılması

Gelen veriler python'da birleştirilen verilere göre ayrıldı ve değişkenlere atandı. + hours ifadesindeki amaç google sheets sunucu tabanlı (Bulutta) çalıştığı için biz ne kadar konum, saat dilimi belirtsek 2 3 saat fark çıkabiliyor o farkı engellemek için böyle bir çözüm buldum.

```
var Curr_Date = new Date(new Date().setHours(new Date().getHours() + hours));
var Curr_Time = Utilities.formatDate(Curr_Date, timezone, 'HH:mm:ss');
var student_id = stripQuotes(e.parameters.data1);
var fullname = stripQuotes(e.parameters.data2);
var uni = stripQuotes(e.parameters.data3);
var faculty = stripQuotes(e.parameters.data4);
var department = stripQuotes(e.parameters.data5);
```

Şekil 18: gelen verilerin ayrılması ve değişkenlere atanması

2.4.2.1 Veriler Nerden Geliyor?

Python'da birleştirdiğimiz URL yani şekil 15'teki yorum satırı kısmı her verinin önünde "data1=" ifadesi var. Bu ifadeti ayırıma işleminde "stripQuotes();" fonksiyonu ile ayrıldı ve değişkenlere atandı.

2.4.3 Verilerin Yerleştirilmesi

Gelen verilerin atandığı değişkenleri hangi hüçreye geleceği ayarlandı. Sütun değerleri veride satır değerleri her gelen veri için arttırıldı.

```
var nextRow = sheet.getLastRow() + 1;
sheet.getRange("A" + nextRow).setValue(Curr_Date);
sheet.getRange("B" + nextRow).setValue(Curr_Time);
sheet.getRange("C" + nextRow).setValue(student_id);
sheet.getRange("D" + nextRow).setValue(fullname);
sheet.getRange("E" + nextRow).setValue(uni);
sheet.getRange("F" + nextRow).setValue(faculty);
sheet.getRange("G" + nextRow).setValue(department);
```

Şekil 19: verilerin yerleştirilmesi

3. Karşılaşılan Zorluklar ve Çözümleri

problem 1 : Karta veri yazarken ilk verinin boş yazılması

çözüm 1 : kodda fazladan +1 ifadesi vardı kaldırıldı

problem 2 : Python'a veriler birden çok gitmesi

çözüm 2 : porttan veriyi okuma kısmı 1 kere çalıştırıldı

problem 3 : verileri okurken bozuk gelmesi

çözüm 3 : ön işleme ile temizlendi

problem 4 : verileri python'a gönderirken eksik gitmesi

çözüm 4 : kart yeterli süre okutulmadığı için olduğu fark edildi

problem 5 : kablolama hatasından arduinunun yanması

çözüm 5 : ikinci bi arduino kullanılması

4. Kaynakça

- [https://medium.com/@mehyalas/arduinonun-avantajlari-839002962229#:~:text=Arduino%27nun%20başlıca%20avantajları%3B&text=Çevresel%20modüllerle\(Shield\)%20birlikte%20devre,nazaran%20daha%20anlaşılır%20ve%20kolaydır.](https://medium.com/@mehyalas/arduinonun-avantajlari-839002962229#:~:text=Arduino%27nun%20başlıca%20avantajları%3B&text=Çevresel%20modüllerle(Shield)%20birlikte%20devre,nazaran%20daha%20anlaşılır%20ve%20kolaydır.)
- <https://www.circuito.io/blog/arduino-uno-pinout/>
- <https://akademi.robolinkmarket.com/arduino-ile-raspberry-pi-karsilastirilmasi/>
- <https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf>
- <https://www.direnc.net/rc522-rfid-okuyucu-1356mhz>
- <https://docs.arduino.cc/hardware/uno-rev3>
- <https://bilginc.com/tr/blog/python-nedir-python-hakkinda-hersey-158/>
- <https://www.uzaktanegitim.com/blog/python-nedir-python-avantajlari-ve-dezavantajlari-nelerdir>
- https://docs.google.com/spreadsheets/d/1wmysnUqHb3umEI33FCf6IpF9pqWWBv-jqy-DTqKr_eQ/edit?usp=sharing
- <https://www.dopinger.com/tr/blog/javascript-nedir>
- <https://aws.amazon.com/tr/what-is/javascript/#seo-faq-pairs#what-are-the-benefits-of-javascript>
- <https://aws.amazon.com/tr/what-is/javascript/>

- <https://html.alldatasheet.com/html-pdf/227839/NXP/MFRC522/3834/64/MFRC522.html>
- <https://www.circuito.io/blog/arduino-uno-pinout/>
- <https://www.youtube.com/watch?v=pdBrvLGH0PE>
- <https://www.youtube.com/watch?v=M9Sci4nsTEk&t=1s>
- [https://www.w3schools.com/jsref/jsref_split.asp#:~:text=The%20split\(\)%20method%20splits,string%20is%20split%20between%20words.](https://www.w3schools.com/jsref/jsref_split.asp#:~:text=The%20split()%20method%20splits,string%20is%20split%20between%20words.)
- <https://forum.arduino.cc/t/append-string-to-array-of-characters/403322/2>
- <https://www.delftstack.com/howto/arduino/arduino-split-string/>
- <https://arduino.stackexchange.com/questions/1013/how-do-i-split-an-incoming-string>
- <https://www.arduino.cc/reference/en/language/variables/data-types/string/functions/substring/>
- <https://docs.arduino.cc/built-in-examples/strings/StringSubstring>
- https://www.youtube.com/watch?v=dbZZlq1_M4o
- <https://www.youtube.com/watch?v=mfiRJlqgToc&t=2589s>
- <https://www.computervision.zone/courses/computer-vision-arduino-chapter-1/?ld-registered=true#login>
- <https://www.diyengineers.com/2021/04/15/learn-how-to-read-an-rfid-tag-with-rc522-and-arduino/>
- <https://fritzing.org/projects/mfrc522>
- <https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf>
- <https://www.alldatasheet.com/datasheet-pdf/pdf/227839/NXP/MFRC522.html>