



SAKARYA
ÜNİVERSİTESİ

HAZIRLAYANLAR

Ad: Ahmet Furkan

Soyad: SÖĞÜTCÜ

Numarası: B201210024

Şubesi: 1/A

Dersi Veren: Cüneyt Bayılmış

Ad: Eren

Soyad: BEKMEZCİ

Numarası: B191210021

Şubesi: 1/B

Dersi Veren: Murat İskefiyeli

-Proje Konusu ve Amacı

Günümüz yapay zeka teknolojileri ile artık insanlar günlük yaşantılarındaki işleri daha kolay ve daha rahat yaparak daha az enerji harcamaya ve zamandan kazanmaya başladı. Kısacası insana yüklenen iş yüklerinde ciddi bir azalma gördük. Biz de bu gelişen ve çok fazla talep gören teknolojiye uyarlanabilecek bir proje yapmaya karar verdik. Turizm sektörüne hitap edecek bu yaptığımız projede biz de insan yükünü ve buna bağlı zaman kayıplarını düşürmeyi amaçlıyoruz.

Otellerdeki oda servislerini incelediğimiz zaman bir odada kalmakta olan kişi ihtiyacı olduğu bir şeyin siparişini oda telefonu ile verdiği zaman devreye birden çok çalışan insan devreye giriyor. Örneğin oda servisi telefonlarına bakan kişi, oda servis şefleri, oda servis garsonları ve oda servis komileri gibi çalışanlar bunlara örnek. Bizim amacımız yukarıda da belirtildiği gibi ve işi daha çok otonomlaştırıp insanların iş yüklerini azaltmak ve güvenli bir şekilde odalara teslimatı gerçekleştirerek turizm teknolojilerine katkı sağlamak. Bu da bir otonom araç ile gerçekleştirilecek.

Öncelikle kullanıcı ile oda servis telefonlarını açan kişi arasındaki iş yükünü bir web sitesi ile çözüp odada kalan kişinin odada bulunan bir karekod ile siteye giriş yapıp istediği siparişleri vermesi sağlanacak. Ardından sipariş verildikten sonra kullanıcıya sitede bir kod gözükecek ve bunu odasına siparişini getiren otonom araca girilmesi istenecek. Siparişi hazırlanıp odasına geldiği anda siparişlerine başkası tarafından ulaşılmayıp sadece kendisinin ulaşabilmesi için sitede gözüken kodu girmesi istenecek.

Biz de IOT projesi olarak bu fikirde oda servisini gerçekleştirecek otonom aracın kasa kısmını ve kullanıcı ile kasa şifre haberleşmesini gerçekledik.

-Projenin Çalışma Sistemi

Kasa şifresinin oluşması için öncelikle müşterinin site üzerinden siparişini onaylaması gerekecek. Onayladıktan sonra sitenin ön yüzünde javascript kodlarıyla oluşturulan rastgele bir şifre belirlenecek. Oluşan şifre yine javascript kullanıp Firebase'e aktararak NodeMCU için hazır halde orada bekleyecek. Firebase'deki şifreyi NodeMCU'ya atmak için NodeMCU kodumuzda Firebase bağlantılarını yapıp şifremizi NodeMCU'da tanımladık. Tanımlandıktan sonra NodeMCU'nun ArduinoUno ile haberleşmesini sağlayarak şifremizi oraya gönderdik. ArduinoUno'da gerekli kontroller yapıldıktan sonra ekrandan bize şifreyi girmemiz istenecek. Şifre de doğru girildikten sonra kasa servo motor yardımı ile açılacak ve müşteri verdiği siparişleri alacak. Siparişlerini aldıktan sonra kasayı müşteri kapatacak. Otonom araç da geldiği yere geri dönecek.

-Kullanılan Malzemeler ve Teknolojiler

- ArduinoUno
- NodeMCU ESP8266
- Servo Motor
- LCD I2C 2x16 Ekran
- 4x4 Keypad
- Kablolar
- Firebase
- Html,Css,Javascript
- Fişek Sürgülü Kilit

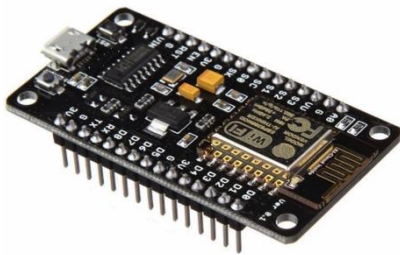
-ArduinoUno

Arduino Uno elektronik donanım ve yazılım temelli bir geliştirme platformu olan Arduino'nun en çok tercih edilen modelidir. Açık kaynaklı olan bu mikrodenetleyici kart elektronik devreleri kontrol etmemizi sağlar.

Biz Uno'yu projemizde NodeMCU ile seri haberleşmesini ve geri kalan motor, lcd ekran, keypad bileşenlerinin ortak bağlantısını kurmak için kullandık.



-NodeMCU8266



NodeMCU üzerinde ESP8266 Wifi modülü bulunduran ve genelde IOT projelerinde kullanılan açık kaynak kodlu bir elektronik geliştirme kartıdır.

Biz NodeMCU kartını internet sitemizde oluşan kasa şifresinin Firebase'den çekilmesinde, çekilen şifrenin Uno ile haberleşerek keypad ile yapılacak şifre kontrolünün yapılması. (?)

-Servo Motor

Servo motor istenilen şekilde hareket kontrolü yapan bir düzeneştir. Servo motorlar robot teknolojilerinde en çok kullanılan motor çeşididir.

Biz servo motoru Uno'ya bağlayarak doğru girilen şifreden sonra kapı kilidinin açılması ve kapanması için kullandık.



-LCD I2C 2x16 Ekran



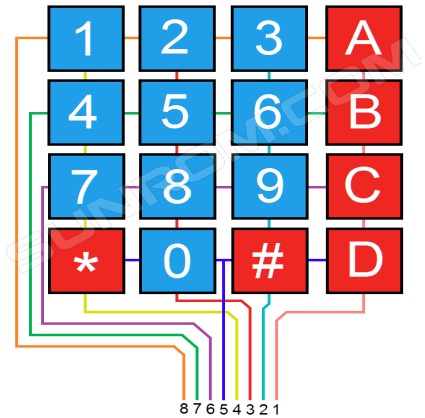
LCD ekran, verilerin ekranda gözükmesini sağlayan bir görüntü teknolojisidir. LCD I2C modülü ise LCD ekranın fazla pin ve kabloya ihtiyaç duyan problemini çözüp bunları sadece 4 pine indirgemıştır.

Biz LCD I2C ekranı projemizde şifre girilmesinin istenmesinde, girilen şifrenin doğruluğunu ekrana çıkartmasında ve kapı kapatma uyarısını vermesi için kullandık.

-4x4 Keypad

Keypad matris mantığı ile çalışan ve projelerimizde menüde gezmek, şifre girmek ve robot kontrolü gibi çeşitli özellikler sağlar.

Biz keypad'i projemizde kasa şifresini girmek ve kasayı kapatmak için kullandık.



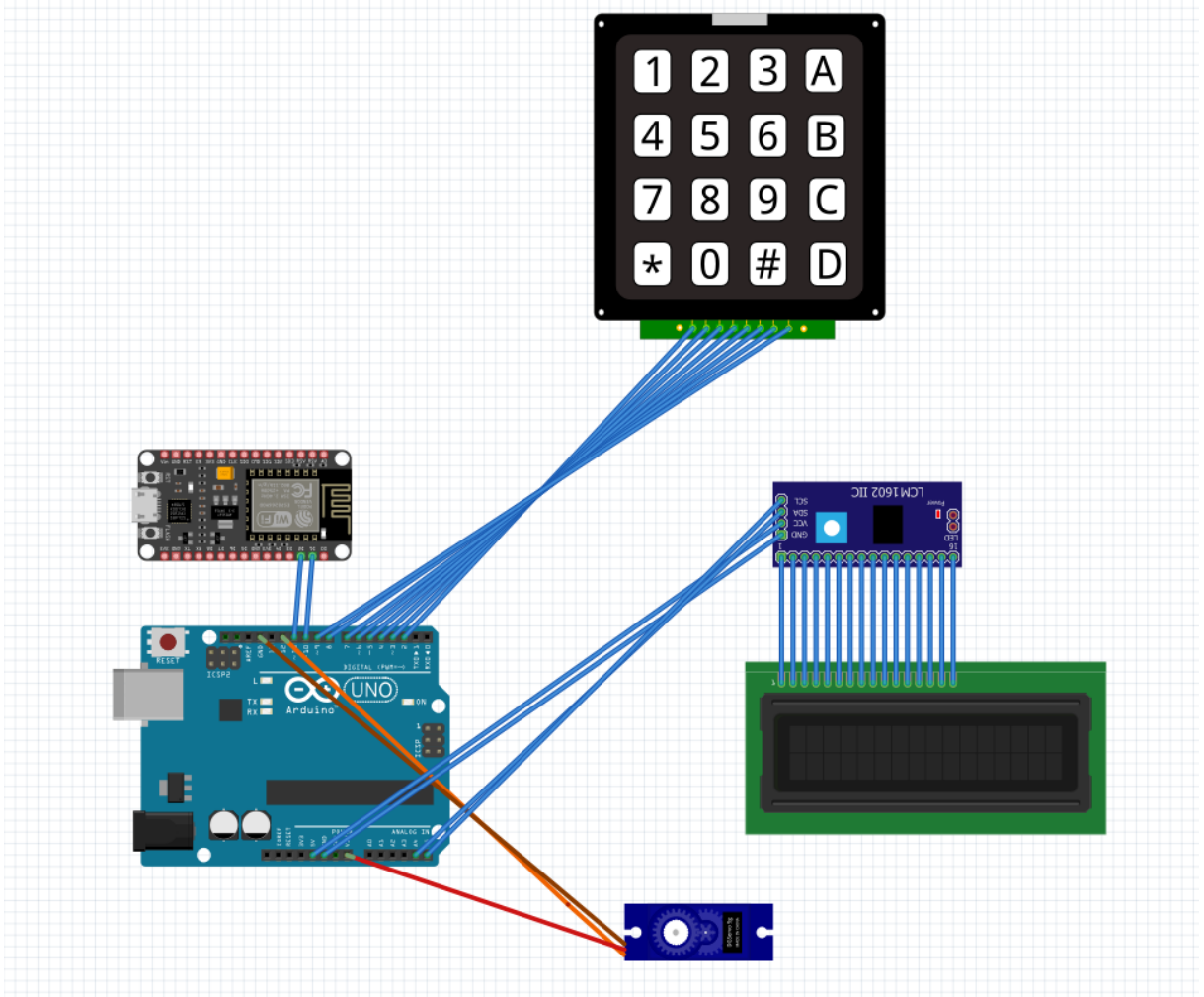
-Firebase

Firebase, herhangi bir proje geliřtirdiđimiz zaman eđer o projede kullanılacak veriler varsa bunları Firebase ile kullanıcı, ürün vs gibi verilerin depolanmasında kullanırız.

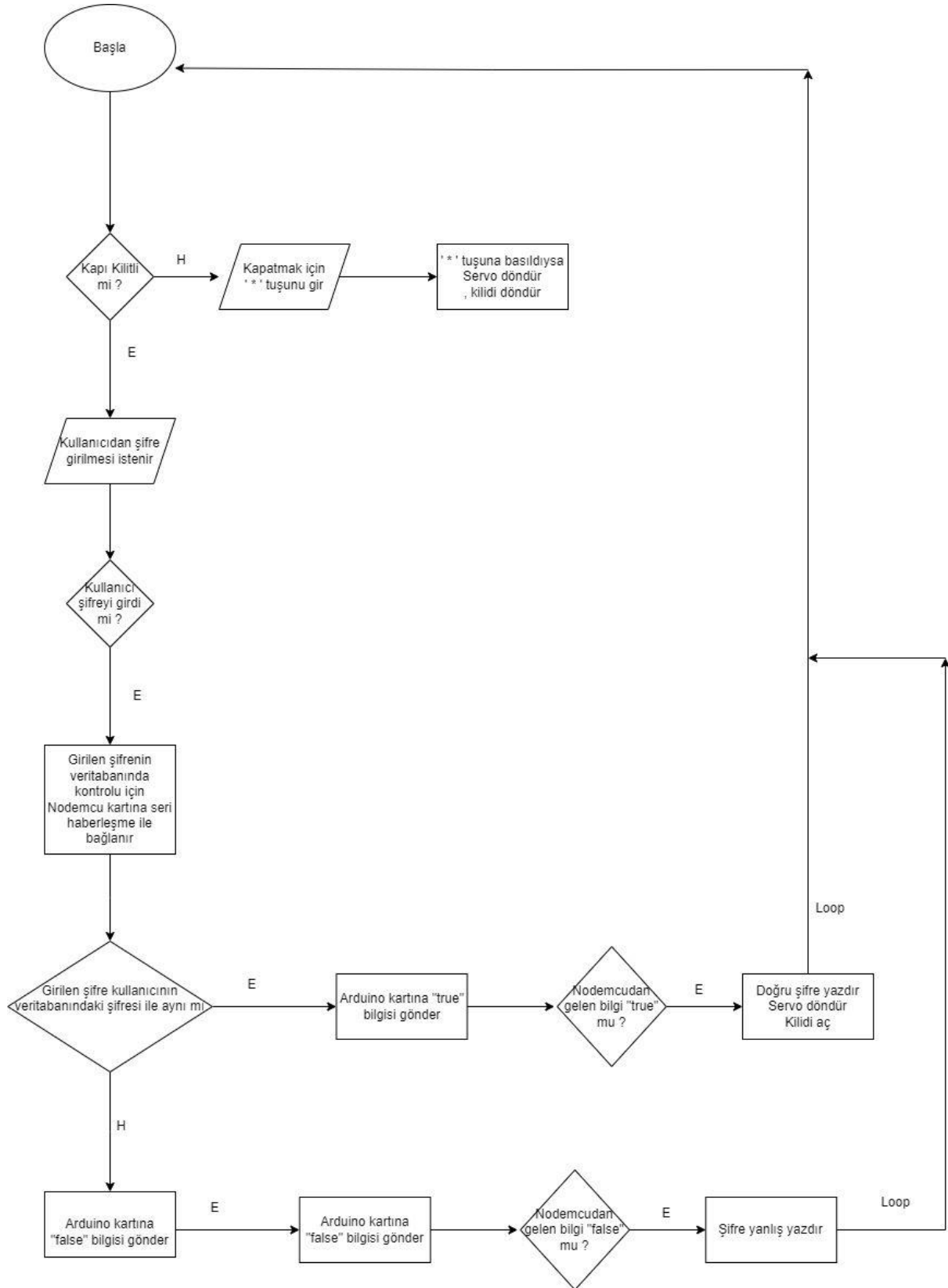
Biz de projemizde Firebase'i siteye giriř yapıp sipariř veren güncel kullanıcının verilerini ile ona atanan kasa řifresi datasını tutmak için kullandık



-Devre řeması



-UML Diyagramı



-Kanvas İş Modeli

<div>Temel Ortaklıklar</div> <div><div>İş Birlikleri</div><p>Çeşitli çevrelerden sponsorluklar bulunabilir, üründe kullanılacak teknolojilerin tedariklerinde satıcı ile bir anlaşma yapılabilir.</p><div>İş Birliğine Düşen Görevler</div><p>Sponsorluk anlamında elini taşın altına koyarak gerekli maddi yardım sağlanmalı.</p></div>	<div>Temel Faaliyetler</div> <div><div>Neler Yapılacak</div><p>Otonom araç ile müşterinin siparişi alınacak. Sipariş yüklemesi otonom araca yapıldıktan sonra otonom araç otel müşterisine gidecek ve müşteri kendisine site üzerinden gösterilen kod ile şifreyi girip siparişlerini alacak.</p><div>Temel Kaynaklar</div><div><div>Nelere Sahip Olunmalı</div><p>Web sayfasını hazırlayacak ekibe, otonom araç tasarımını yapacak bir ekibe, otonom araç yazılımını yapacak bir ekibe ve kullanılacak elektronik veya yazımsal teknolojilere sahip olunmalı.</p></div></div>	<div>Değer Önerisi</div> <div><div>Problemler</div><p>Siparişlerdeki gecikmeler, binden fazla personellerin bir sipariş için uğraşması ve bu arada gerçekleşen zaman kayıplarını gidermek için otonom araç ile çözüm sağlanacak.</p><div>Müşteri Açısından Problemler</div><p>Müşterilerin bekleme sürelerindeki fazlalık ve telefon başlıtı beklemek veya başlıtı için kendisini hazırlamak yerine websitesi üzerinden rahat bir şekilde siparişini verip siparişi hakkında ayrıntılı bilgilere sahip olacak.</p></div>	<div>Müşteri İlişkileri</div> <div><div>Durum Kontrolü</div><p>Otellerle ürünümüz hakkında otonom aracının kullanılmaya başlanmasından 1 ay sonra hizmetimiz hakkında görüşlerini veya sorularını talep ederiz. Bunu da yüz yüze yaparsak müşteri bağlılığını kazanmadaki ihtimalimiz artar.</p><div>Kanallar</div><div><div>Müşterilere Ulaşım</div><p>Otonom araç hizmetimizi otellere ulaştırmak için sosyal medya üzerinde sayfa açarsa onlarla irtibata geçebiliriz veya mail adresleri üzerinden de iletişim kurabiliriz. Turizmin hareketli olduğu yerlerdeki yerel gazetelere ilan vererek hedef kitlemize ulaşabiliriz.</p></div></div>	<div>Müşteri Segmenti</div> <div><div>Hedef Kitlemiz</div><p>Bünyelerinde otonom araç barındırarak iş yükünü azaltmak isteyen ve ekonomik anlamda da rahatlamak isteyen oteller.</p><div>Hizmetten Yararlanacaklar</div><p>Hizmetimizden otonom araç sistemimizi kurmak isteyen oteller ve o otellere kalan müşteriler siparişlerini otonom araçla alarak yararlanacak.</p></div>
<div>Maliyet Yapısı</div> <div><div>Giderler</div><p>Otonom aracının yapılacağı kullanılacak malzemeler ve oluşturulacak teknolojileri gerçekleştiren çalışanların maaşları.</p><div>Hangi Değere Ücret Ödenecek</div><p>Otonom aracımız için ve eş zamanlı sunacağımız web sitesi için müşterilerden ödeme talep edilecek.</p><div>Ücret Nasıl Ödenecek</div><p>Ücret havale yolu ile aktarılacak.</p><div>Fiyatlandırma Politikası</div><p>Müşterilerimize yapacağımız fiyatlandırma politikası abonelik üzerinden olacak. Müşteriler belli bir dönem arasında hizmetimizden yararlanacak (istense kullanılabilir).</p></div>				

-Veri Yönetimi

Projede kullanacağımız verilerin depolanması için Firebase'i kullandık. Müşteri sisteme kayıt olduğu zaman realtime databasede bilgileri eklenecek ve sıra halinde gösterilecek. Site üzerinden sipariş verenler ise sadece bir veri başlığı üzerinden kim sipariş verdiyse onun bilgilerinin gözükeceği şekilde ayarlanacak.

https://iot-deneme3-default-rtbd.firebaseio.com/ users 1 email: "eren@gmail.com" name: "eren" newKey: "755" password: "eren123" -NKUpAmuW0vakE0mFtL1 email: "furkan@gmail.com" name: "furkan" password: "123456" -NKUpDCSMBXT5s93hg7_ email: "bayram@gmail.com" name: "bayram" password: "asdfdsa" -NKUpFzjRAs7V6rIxxq4 email: "burak@gmail.com" name: "burak" password: "asdf1234"

Mavi bölge, kim sipariş verdiyse onun bilgilerini güncelleyerek oraya atacak.

Kırmızı bölge, daha önceden sisteme kayıt olmuş kullanıcılar.

-Kod Açıklamaları

-Web Sitesi

```
// Import the functions you need from the SDKs you need
import { initializeApp } from "https://www.gstatic.com/firebasejs/9.6.1/firebase-app.js";
import { getDatabase, set, ref, push, child, onValue } from "https://www.gstatic.com/firebasejs/9.6.1/firebase-database.js";

const firebaseConfig = {
  apiKey: "AIzaSyCOBL1cIRsB8nVyd95ViAMdt8hx58ikv9A",
  authDomain: "iot-deneme3.firebaseio.com",
  databaseURL: "https://iot-deneme3-default-rtdb.firebaseio.com",
  projectId: "iot-deneme3",
  storageBucket: "iot-deneme3.appspot.com",
  messagingSenderId: "177571169203",
  appId: "1:177571169203:web:e55df40a1fa814672c81c6",
  measurementId: "G-14R7RMY492"
};

const app = initializeApp(firebaseConfig);
const database = getDatabase(app);
```

Javascript kodumuz ile firebase bağlantısını bu şekilde gerçekleştiriyoruz.

```
var kullanıcıId = 0;
// write data
order.addEventListener('click', (e) => {
  let key = generateRandomIntegerInRange(100, 999);
  let newKey = key.toString();

  kullanıcıId++;
  var name = document.getElementById('name').value;
  var email = document.getElementById('email').value;
  var password = document.getElementById('password').value;

  const userId = push(child(ref(database), 'users'), {
    key: |
    name: name,
    email: email,
    password: password,
    newKey: newKey
  });
  alert(newKey);
});
//write data
```

Burada sipariş veren kullanıcının verilerini güncelliyoruz ve daha önce rastgele oluşturduğumuz bir kasa şifresi ataması yapıyoruz.

-Arduino

```
#include <Servo.h>
#include <LiquidCrystal_I2C.h> //Lcd kütüphanesi
#include <Wire.h> //Lcd kütüphanesi
#include <Keypad.h> //Keypad Kütüphanesi
#include <SoftwareSerial.h> //nodemcu ve uno arasında seri haberleşme yapabilmek için gerekli olan kütüphane

SoftwareSerial Uno_serial (10,11); //Rx ve Tx

char c;
String dataIn;
int Position = 0;
bool door = true;
Servo servo;
LiquidCrystal_I2C lcd(0x27 , 16 , 2);

const byte ROWS = 4; //four rows
const byte COLS = 4; //four columns
char keys[ROWS][COLS] = {
  {'1','2','3' , 'A'},
  {'4','5','6' , 'B'},
  {'7','8','9' , 'C'},
  {'*','0','#' , 'D'}
};

byte rowPins[ROWS] = {2,3,4,5}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {6,7,8,9}; //connect to the column pinouts of the keypad
String sifre = "";
String girilen_sifre = "";
int basamak = 0;

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );
```

Gerekli kütüphanelerin tanımlanması, değişkenlerin tanımlanması ve bağlantıların yapılmasını Arduino kodumuzun baş kısmında yaptık

```
void setup() {
  Serial.begin(57600); //uno - pc
  Uno_serial.begin(9600); // uno - nodemcu
  lcd.init();
  lcd.backlight();
  lcd.setCursor(0,0);
  lcd.print("Sifre giriniz");

  servo.attach(12);
  ServoClose();
}
```

Bu kısımda port haberleşme hızları ve lcd ile servonun başlangıç durumları atandı.

```

void loop() {

    if(door == 0) //
    {
        lcd.setCursor(0,1);
        lcd.print("Kapatmak icin *");
        delay(1500);
        for(int i = 0 ; i<16 ; i++)
        {
            lcd.setCursor(i,1);
            lcd.print(" ");
        }

        char tus_door = keypad.getKey();

        if (tus_door == '*') // when the '*' key is pressed
        {
            lcd.setCursor(0,1);
            lcd.print("Kapaniyor..");
            delay(1000);

            for(int i = 0 ; i<16 ; i++)
            {
                lcd.setCursor(i,1);
                lcd.print(" ");
            }
            ServoClose(); // closes the servo motor
            door = 1;
            delay(1500); // waits 3 seconds
        }
    }
}

```

Burada eğer kilit açık ise gerçekleşecek kapatma durumları yer alıyor.

```

char tus = keypad.getKey();

//şifre keypaddan alınır ve ekranda gösterilir
if(tus != NO_KEY)
{
    lcd.setCursor(basamak , 1);
    lcd.print(tus);
    girilen_sifre = girilen_sifre + tus ;
    basamak++;
}

// '*' tuşuna basılırsa şifre girme işlemi tamamlanmış demektir.
if(tus == '*')
{
    Uno_serial.print(girilen_sifre.substring(0 , basamak-1)+'\n'); //girilen şifreyi önce sonundaki '#' karakterini silerek seri haberleşme ile Nodemcu kartına göndeririz
                                                                    //gerekli kontroller nodemcudaki yapılır ve girilen şifre doğruysa "true" bilgisini arduino uno kartına gönderir
    delay(3000);
    if(Uno_serial.available()==0)
    {Serial.println("data gelmedi");}
    while(Uno_serial.available()>0) //nodemcudan gelen veriler sonlana kadar verileri okuyoruz
    {
        c = Uno_serial.read();
        if(c == '\n') {break;} //'n' karakterinin Stringin sonunun gelip gelmediğini kontrol etmek için kullandık
        else {dataIn+= c;}
    }
    if(c == '\n')
    {
        Serial.println(dataIn);
        sifre = dataIn;
        c = 0;
        dataIn="";
    }
}

```

Bu kısımda da NodeMCU kontrolüyle şifremizin keypad üzerinden giriş aşamaları yer alıyor.

```

//nodemcudan gelen bilgi "true" yada "false" ve bu sifre değişkenine atanır.
if(sifre == "true")
{
    //şifre doğruysa ekrana Doğru şifre yazdırılır
    lcd.setCursor(0,1);
    lcd.print("Dogru sifre !");
    ServoOpen(); //Servo döndürme döndürme fonksiyonu
    door = 0; // kilidin açılır
    delay(1000);
    for(int i = 0 ; i<16 ; i++)
    {
        lcd.setCursor(i,1);
        lcd.print(" ");
        //ekranın alt kısmını temizler sadece
    }
    girilen_sifre = "";
    //sifre="";
    basamak = 0;
}
else if(sifre == "false")
{
    lcd.setCursor(0,1);
    lcd.print("Yanlis sifre !");
    delay(1000);
    for(int i = 0 ; i<16 ; i++)
    {
        lcd.setCursor(i,1);
        lcd.print(" ");
    }
    door = 1; // kilit kapanır
    girilen_sifre = "";
    basamak = 0;
}
}
}

```

Şifremiz eğer doğru veya yanlış ise gerçekleşecek durumlar da burada yer alıyor.

```

//servo motor açma ve kapama fonksiyonları

void ServoOpen() // opens the servo
{
    for (Position = 180; Position >= 0; Position -= 5) { // moves from 0 to 180 degrees
        servo.write(Position); // moves to the position
        delay(15); // waits 15 milliseconds
    }
}

void ServoClose() // closes the servo
{
    for (Position = 0; Position <= 180; Position += 5) { // moves from position 0 to 180 degrees
        servo.write(Position); // moves to the position
        delay(15); // waits 15 milliseconds
    }
}

```

Kodumuzun en alt kısmında servo motorun çalışma durumlarının fonksiyonları yer alıyor.

-NodeMCU

```
#include <SoftwareSerial.h>
#include <ESP8266WiFi.h>
#include <FirebaseArduino.h>

#define FIREBASE_HOST "iot-deneme3-default-rtdb.firebaseio.com"
#define FIREBASE_AUTH "HOy1mp9lpam2Oyq8R7OXyp9WlM600mZ7gK6KHT6Y"
#define WIFI_SSID "a.frkn"
#define WIFI_PASSWORD "sezkentel9"
```

```
SoftwareSerial Nodemcu_serial (D1,D2);
```

```
char c;
String dataIn;
```

Gerekli kütüphaneler, port girişleri, internete bağlantı bilgileri ve bazı değişkenler tanımlanıyor.

```
void setup() {

    Serial.begin(57600); //uno - pc

    Nodemcu_serial.begin(9600); // uno - nodemcu

    delay(10);
    Serial.println();
    WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
    Serial.print(WIFI_SSID);
    Serial.println(" Kablosuz Agina Baglaniyor");

    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }

    Serial.println();
    Serial.print(WIFI_SSID);
    Serial.println(" Kablosuz Agina Baglandi");
    Serial.print("IP adresi: ");
    Serial.println(WiFi.localIP());

    Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);

}
```

Haberleşme hızlarının da belirlendiği bu kısımda da wifi ve firebase bağlantısının yapılışı var.

```

String key;

void loop() {
  key = Firebase.getString("users/1/newKey");
  if(Firebase.failed()){
    Serial.print("setting /user failed:");
    Serial.println(Firebase.error());
  } else {
    Serial.println(key);
  }
  delay(1000);
  //Arduinodan gelen şifre verisi okunur
  while(Nodemcu_serial.available()>0)
  {
    c = Nodemcu_serial.read();
    if(c == '\n') {break;}
    else          {dataIn += c;}

  }
  if(c == '\n')
  {
    //gelen şifre veritabanındaki şifreyle eşleşiyorsa arduino kartına true bilgisi geri gönderilir
    if(dataIn == key)
    {
      delay(500);
      Nodemcu_serial.print("true\n");
    }
    else
    {
      delay(500);
      Nodemcu_serial.print("false\n");
    }
    Serial.println(dataIn);
    c = 0;
    dataIn = "";
  }
}

```

NodeMCU kodumuzun son kısmında ise firebase'den aldığımız verinin ArduinoUno'ya gönderimi ve ArduinoUno'da girilen şifrenin buraya gönderilerek yapılan doğrulaması var.

-Kaynakça

http://www.megep.meb.gov.tr/mte_program_modul/moduller_pdf/Oda%20Servisi.pdf

<https://maker.robotistan.com>