

SENIOR PROJECT II BİTİRME TEZİ

AKILLI AFET SİSTEMİ

ÖĞRENCİ

AYŞE YILDIRIMEL – 2015010213027

FURKAN AKDOĞAN – 2015010206015

ÖĞRETİM GÖREVLİSİ

Dr. Öğr. Üyesi Yasin ORTAKCI

İÇİNDEKİLER

KAPAK SAYFASI	1
İçindekiler	2
Şekiller Tablosu	3
ÖZET	4
1.GİRİŞ	5
1.1 Projenin Amacı ve Önemi	5
1.2. Projenin İçerdiği Yenilik (Özgünlük) Unsuru	5
1.3. Projenin İlgili Olduğu Teknoloji Alanları	5
2. YÖNTEM VE TEKNİKLER	6
2.1 Elektronik Algılayıcı Sistem	6
2.2 Verici IoT Sistem	6
2.3 Alıcı IoT Sistem	6
2.4 Mekanik Sistem	7
2.5 Güç Kaynağı	7
Proje İş-Zaman Çizelgesi	7
3. BULGULAR	8
4. SONUÇ	13
6. KAYNAKLAR	13

ŞEKİLLER TABLOSU

Örnek 1: Arduino ile Titreşim Sensörünün Örnek Bağlantısı	6
Örnek 2: Arduino ile ESP-8266 Bağlantı Örneği	6
Örnek 3: Arduino ile Röle Bağlantısı Örneği	7
Tablo 1: Proje'nin Gantt Şeması	7
Şekil 1: Geliştirme Kartı ile Sensörlerin Bağlanması	8
Şekil 2: Örnek Sensör Kodlaması	8
Şekil 3: Geliştirme Kartı ile ESP-8266 Wifi Modülünün Bağlantısı	8
Şekil 4: ESP8266 Wifi Modülünün İnternet Bağlantısı	9
Şekil 5: ESP8266 Wifi Modülü Veri Gönderimi	9
Şekil 6: Ana Sayfa ve Örnek Afet Bilgilendirme Sayfaları	10
Şekil 7: Afet Anında Mobil Uygulamada Açılacak Sayfalar	10
Şekil 8: Deprem Olduğu Zaman Mobil Uygulamada Kullanıcının Konumunu ve Kullanıcıya En Yakın Toplanma Bölgesinin Konumunun Gösterilmesi	11
Şekil 9: Harita Kodlamaları	11
Şekil 10: Thingspeak Mobil Bağlantısı Kodları	12
Şekil 11: Tüm Cihaz Bağlantısı	12
Şekil 12: Cihazın Kutulanmış Hali ve İç Görünümü	12

ÖZET

Afetler geçmişten günümüze insanoğlunun en büyük problemlerinden biridir. Doğal ya da insan kaynaklı oluşabilen afetler birçok insanın hayatına mal olmuş, psikolojik sorunlara yol açmış, mal kaybına sebebiyet vermiştir. Örnek vermek gerekirse; Ülkemizde geçtiğimiz yıl içerisinde oluşan depremler onlarca insanı sevdiklerinden koparmıştır. Tam da bu sebeplerle yola çıkarak hazırladığımız cihazımız basitçe anlatmak gerekirse deprem, yangın, su taşkını ve doğalgaz kaçağını sensörler aracılığıyla algılayarak insanları uyarmakta, doğalgaz, su ve elektriği kesmekte ve geliştirdiğimiz mobil uygulamaya sinyal göndererek evlerinde olmasalar dahi kullanıcıların durumdan haberdar olmalarını sağlamaktadır.

Kapı girişlerinde konumlandırılan cihazımız afet durumunu sensörler aracılığıyla tespit ederek doğalgaz, su ve elektriği kesmekte ardından kendi ışığını yakarak insanların dışarı çıkmasını kolaylaştırmaktadır. Aynı zamanda cihazımız mobil uygulamamıza da sinyal göndererek uygulamanın alarm konumuna geçmesini sağlamak ve insanları bilgilendirmektedir. Mobil uygulamamız ise insanları bilinçlendirmek üzerine tasarlanmıştır. Afet anlarında nasıl davranmak gerekir? Öncesinde ne gibi önlemler alınabilir? gibi soruların cevabını vermektedir. Ayrıca deprem anında insanlara en yakın toplanma alanının konumunu göstermektedir.

Projemiz Arduino ve sensör sistemleri, nesnelerin interneti teknolojisi ve mobil uygulama içermektedir. Arduino geliştirme kartı ile sensörler üzerinden okunan veriler değerlendirilerek tehlike durumunda ESP-8266 Wifi modülü aracılığıyla mobil uygulamaya Thingspeak kullanılarak bilgilendirme yapılmaktadır. Aynı zamanda Arduino ile kontrol edilen röleler ile elektrik kontrollü vanaların kapatılması ve elektriğin kesilmesi sağlanmaktadır. Daha sonra insanların karanlıkta rahat hareket edebilmeleri için cihaz kendi lambasını yakmaktadır. Bunun için şarjlı lamba kullanılmıştır.

1.GİRİŞ

1.1. Projenin Amacı ve Önemi

Dünya, varlığı boyunca birçok yıkıcı doğal afete tanıklık etmiştir. Bu doğal afetlerin yanı sıra insanların sebep olduğu beşerî afetlerde yaşanmaktadır. Bu doğal ve beşerî afetler dünya üzerindeki yaşama ve yapılara ciddi hasarlar vermiştir. Geçmişte yaşanan yıkıcı afetler gibi, gelecekte de birçok afet yaşanacağı aşıkardır ve bu afetler sonucunda mal ve can kaybı kaçınılmazdır. Birleşmiş Milletler Afet Riski Azaltma Ofisi'nin (UNISDR) Afetlerin Epidemiyolojisi Araştırma Merkezi tarafından (CRED) hazırlanan 2018 afet raporuna göre, 2017 yılında dünyada 11 bin 804 kişinin hayatını kaybettiği afetlerden 68 milyondan fazla kişi etkilenmiştir. [1]

Afetler ülkemizin büyük bir problemidir. İnsanların afet konusunda bilinçsiz olması da bu problemin zararlı etkilerini arttırmaktadır. Özellikle afet sonrasında insanların atlamaması gereken önemli aşamalardan biri elektrik, su ve doğalgazı kapatmaları olmasına rağmen bu çoğunlukla atlanmaktadır. Bunun sonucunda ise gaz sızıntıları elektrik kaçakları vs. sebebiyle hasarlar oluşabilmektedir. Bunun yanı sıra insanlar afete uykularında da yakalanabilmektedirler. Örneğin birçok insan uykularında yangını fark edemedikleri için can vermiştir.

Mobil uygulamamız ile insanlar afet konusunda bilinçlendirilecektir. Afetle ilgili bilmeleri gerekenler ve uygulamaları gereken adımlar uygulamada yer almaktadır. Örn; deprem çantası hazırlama vb. Afet sırasında ise yapmış olduğumuz cihaz Arduino titreşim sensörü aracılığıyla depremleri, alev sensörü ile yangınları gaz sensörü ile gaz kaçaklarını ve su seviyesi sensörü ile de su taşkınlarını algılayabilmekte ve elektrikle kontrol edilebilen vanaları kapatabilmektedir. Daha sonra sigortalarda bulunan kaçak akım rölesinden faydalanarak sigortaların atmasını sağlamaktadır. Böylece insanların atladığı bu önemli adım sistem tarafından yapılmaktadır. Afet algılandığında insanların akıllı telefonlarına kablosuz iletişim araçları aracılığıyla bilgi göndermekte ve insanların bir alarm ile uyarılmasını sağlamaktadır. Aynı zamanda sisteme bağlı sirenlerle de alarm vermektedir. Elektrik kesintisi sonrasında devreye giren ışık kaynağımız sayesinde insanlar güvenli bir şekilde çıkışları kolayca bulabilir ve evlerinden ayrılabilirler. Afet boyunca uygulamamız sayesinde nasıl davranmaları konusunda bilinçlenmesini amaçladığımız insanlar kendilerini korumaya alacak ve böylece panikten kaynaklı ölüm ve yaralanmaların önüne geçilmiş olacaktır. Afet sırasında evlerinde olmayan insanlar yine mobil uygulamamız sayesinde anında bilgilendirilmekte ve böylece insanlar gerekli önlemleri alabilmektedirler. Deprem sonrasında ise insanların birçoğu toplanma alanlarını bilmemektedirler. Mobil uygulamamız bu konuda da en yakın toplanma alanlarını göstererek insanlara destek sağlamaktadır.

Özetle projemiz afet sonrasında bilinçsizlik ve panik kaynaklı oluşabilecek hasar ve kayıpların önüne geçmek amacıyla tasarlanmıştır.

1.2.Projenin İçerdiği Yenilik (Özgünlük) unsuru

Literatür taraması yaptığımızda projemize çeşitli yönlerden benzerlik gösteren birçok ürün karşımıza çıkmıştır (Yangın alarmları, duman detektörleri vb.). Yalnız bu ürünlerin hiçbirisi tam bir paket halinde deprem, su taşkını, gaz kaçağını ve yangını algılamamaktadır. Bunlardan sadece bir ya da ikisine odaklanmaktadır. Bazı ürünlerde dijital ekran bulunuyor ancak hiçbirisi mobil platform sunmuyor. Bizim yapmak istediğimiz projemiz bütün bu işlemleri tek bir paket halinde yerine getirmesi ve mobil destek platformu sağlaması açısından piyasadaki bu açığı kapatmaktadır. Benzer şekilde bazı akıllı ev sistemleri de bu uygulamaları ayrı ayrı sunmaktadır ve mobil uygulama ile takibinin yapılmasını sağlamaktadır. Ancak bizden farklı olarak afet sırasında vanalar ve elektriği kendiliğinden kesmemektedir.

Literatürden örnek vermek gerekirse “Secty Life Patron” [2] adındaki sistem depremi önceden algılayıp uyarı verdiğini iddia etmektedir ve bizim sistemimizde yapmayı planladığımız gibi vanaları ve elektriği kapatmaktadır. Ancak bu sistem ülkemizde yaygın olarak kullanılmamaktadır ve bizim yapacağımız projeden farklı olarak bir mobil platform sunmamaktadır ve sadece depreme yöneliktir. Aynı zamanda geniş çaplı bir merkez üsse ihtiyaç duymaktadır. Diğer uygulama ise “Siscut” [3] adı verilen elektronik deprem cihazıdır. Bu cihaz ise deprem sırasında doğalgaz vanasını kapatmakta ve alarm vermektedir. Aynı şekilde bu cihazda da bir mobil platform bulunmamaktadır ve sadece depreme yöneliktir.

1.3.Projenin İlgili Olduğu Teknoloji Alanları

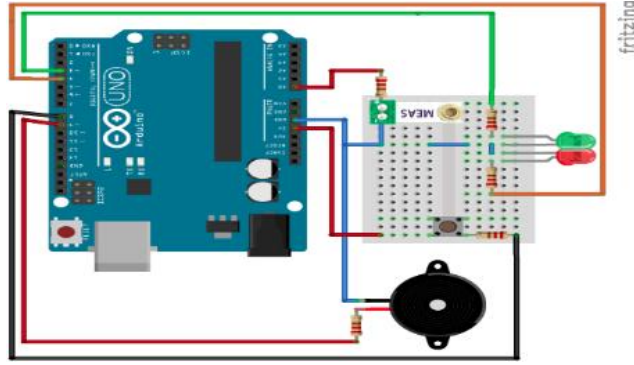
Projemiz IoT, Arduino, mobil uygulama, sensörler ve elektriksel devreler içermektedir.

2.YÖNTEM VE TEKNİKLER

Arduino Mega 2560 geliştirme kartı ile sensörler (titreşim, alev, duman, gaz) üzerinden okunan veriler değerlendirilerek, tehlike varsa Esp8266 Wifi modülü aracılığıyla Thingspeak (IoT Veri Analiz Sitesi) platformuna verileri kaydetmektedir. Mobil uygulamamız ile Thingspeak platformundaki veriler çekilerek sürekli olarak kontrol edilmektedir. Bu sayede afet anında mobil uygulamamız bilgilendirilmektedir. Aynı zamanda Arduino ile kontrol edilen röleler ile elektrik kontrollü vanaların kapatılması ve elektriğin kesilmesi sağlanmaktadır. Daha sonra insanların karanlıkta rahat hareket edebilmeleri için cihaz kendi lambasını yakmaktadır. Bunun için şarjlı lamba kullanılmıştır.

2.1 Elektronik Algılayıcı Sistem

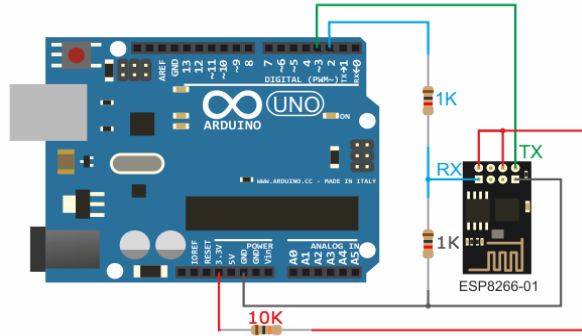
Projemizde Arduino Mega 2560 geliştirme kartı kullanılmıştır. Deprem algılanması için SW-420 titreşim sensörü, gaz kaçaqlarının algılanması için MQ-4 gaz sensörü, su taşkınlarının algılanması için su seviyesi sensörü ve yangınları algılamak için kızıl ötesi sensörü ve MQ-2 duman sensörü kullanılmıştır. Bu sensörlerin datasheetlerinden faydalanılarak bir devre oluşturulmuştur. Sensör verilerini okumaya yönelik program Arduino dili ile yazılmış ve geliştirme kartına yüklenerek sistem çalışır hale getirilmiştir. Sistem sürekli dinleme halindedir.



2.2 Verici IoT Sistem

Örnek 4: Arduino ile Titreşim Sensörünün Örnek Bağlantısı

Sensörlerden alınan veriler bir afete işaret ederse verici sistem (ESP-8266) aracılığıyla Thingspeak adlı siteye veri yazmaktadır.



Örnek 5: Arduino ile ESP-8266 Bağlantı Örneği

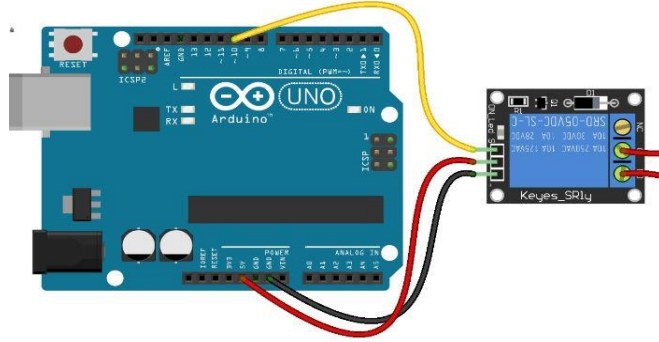
2.3 Alıcı IoT Sistem

Alıcı sistem akıllı telefonlar için yazdığımız bir programdır. Bu program içerisinde afetle ilgili bilgilendirmeler bulunmaktadır ve yazmış olduğumuz arka planda çalışan bir servis sayesinde sürekli olarak Thingspeak sitesinden veri okumaktadır eğer Arduino bu siteye afet verisi girerse alarm vererek insanların uyarılmasını sağlamaktadır. Programda afetlerle ilgili verilen bilgiler; Deprem çantası nasıl hazırlanır? Afet sırasında neler yapılmalıdır? Afetten sonra ne yapılmalıdır? En yakın toplanma alanları nelerdir? gibi sorulara cevap verecek şekilde tasarlanmıştır. Bu algılayıcı sistem insanlar evlerinde yokken bile onları bilgilendirmekte ve böylece gerekli önlemleri almalarını sağlamaktadır. Ayrıca uygulamamız kullanıcı konumuna en yakın toplanma alanı konumunu da göstermektedir. Bunun için Google Map API kullanılmıştır. Kullanıcı

konumu çekilerek daha önceden hazırladığımız toplanma alanları veri tabanına erişip kullanıcı konumu ile veri tabanındaki konumları karşılaştırmakta ve en yakın konumu tespit edip kullanıcıya göstermektedir. Bu veri tabanı localde tutulmaktadır.

2.4 Mekanik Sistem

Projemizin mekanik kısmında elektrikle kontrol edilebilen vanalar bulunmaktadır ve bu vanaların kontrolü geliştirme kartına bağlanan röleler aracılığıyla yapılmaktadır. Röle projemizde Arduino ile kontrol edilen bir anahtar görevi görmektedir. Vana olarak manuel açmalı normalde açık selenoid vana kullanılmıştır. Arduino geliştirme kartına siren bağlanarak afet anında alarm verecek şekilde programlanmıştır ve elektrik kesintisinden sonra devreye giren şarjlı lamba ile çıkışlarda aydınlatma sağlanarak insanların ortamdaki uzaklaşmasını kolaylaştırmaktadır. Aynı zamanda Arduino devrelerinin, sensör ve rölelerin muhafazası için bir proje kutusu kullanılmıştır.



Örnek 6: Arduino ile Röle Bağlantısı Örneği

2.5 Güç Kaynağı

Projemizde güç kaynağı olarak 10.000mAh powerbank kullanılmıştır. Bu sayede elektrik kesilmesi halinde dahi cihazımız çalışabilmektedir.

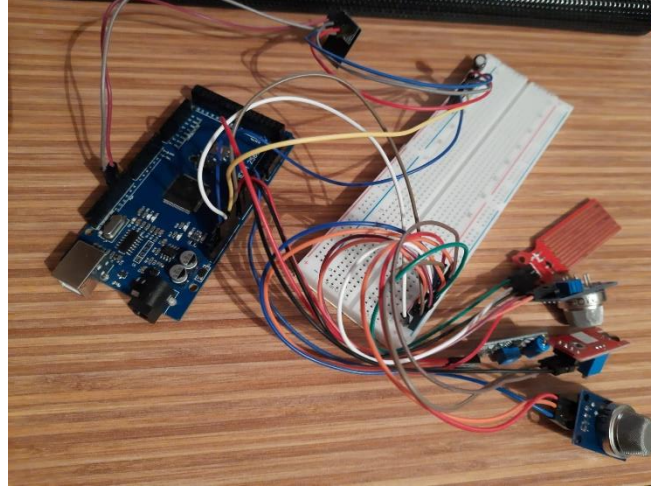
Proje İş-Zaman Çizelgesi

AYLAR/HAFTALAR																												
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
GÖREVLER																												
Arduino Öğrenme																												
Android Mobil Programlama Öğrenme																												
Mobil platformun Java ile kodlanması																												
Arduino geliştirme kartı ile titreşim sensörünün bağlantısı ve kodlanması																												
Arduino geliştirme kartı ile rölelerin bağlantısı ve kodlanması																												
Röleler ile vanaların kontrolü için gerekli devrenin kurulması																												
Arduino ürün ile mobil platform arasında iletişimin sağlanması																												
Arduino geliştirme kartı ile wifi modül bağlantısının yapılması ve kodlanması																												
Arduino geliştirme kartı ile gsm modül bağlantısının yapılması ve kodlanması																												
Arduino geliştirme kartı ile xbee modül bağlantısının yapılması ve kodlanması																												
Xbee, gsm, wifi modüllerinden hangisinin daha verimli olmasının belirlenmesi																												
Arduino geliştirme kartı ile siren bağlantısı ve kodlanması																												
Sistemin muhafazasının uygun malzemeler ile sağlanması																												
Sistemin senkronizasyonunun denetlenmesi																												
Son denemeler ve iyileştirmelerin yapılması																												

Tablo 2: Proje'nin Gantt Şeması

3. BULGULAR

Öncelikle geliştirme kartı ile sensörlerin bağlantıları yapıldı (Şekil 1) ve alarm kodlamaları gerçekleştirildi (Şekil 2). Sensörlerin eşik değerleri tespit edilerek gerekli düzenlemeler yapıldı.



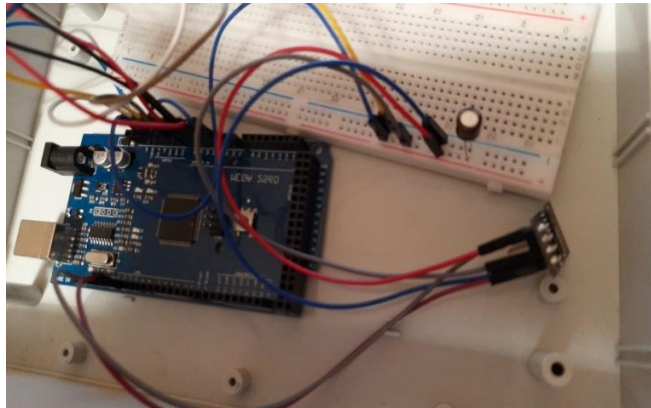
Şekil 10: Geliştirme Kartı ile Sensörlerin Bağlanması

```
void loop()
{
    int sensorValue_gaz = analogRead(sensor_gaz);
    int sensorValue_alev = analogRead(sensor_alev);
    int sensorValue_su = analogRead(sensor_su);
    int sensorValue_titresim = digitalRead(sensor_titresim);
    int sensorValue_yangin = analogRead(sensor_yangin);
    delay(100);

    if(sensorValue_gaz >= threshold_gaz)
    {
        alarm_gaz(500);
        Serial.println("gaz");
        while(a<5){
            wifi("field1",1);
            alarm_gaz(500);
            a++;
        }
        digitalWrite(role_gaz,LOW);
        alarm_gaz(500);
        digitalWrite(role_elektrik,LOW);
    }
}
```

Şekil 11: Örnek Sensör Kodlaması

Geliştirme kartı ile ESP-8266 Wifi Modülünün bağlantısı ve gerekli kodlamalar yapıldı (Şekil 3).



Şekil 12: Geliştirme Kartı ile ESP-8266 Wifi Modülünün Bağlantısı


```

Serial.begin(115200);
Serial.println("AT");
Serial.println("AT Yollandı");
while(!Serial.find("OK")){
    Serial.println("AT");
    Serial.println("ESP8266 Bulunamadı.");
}
Serial.println("OK Komutu Alındı");
Serial.println("AT+CWMODE=1");
while(!Serial.find("OK")){
    Serial.println("AT+CWMODE=1");
    Serial.println("Ayar Yapılıyor...");
}
Serial.println("Client olarak ayarlandı");
Serial.println("Aga Bağlanılıyor...");
Serial.println("AT+CWJAP=\"" + agAdi + "\", \"" + agSifresi + "\"");
while(!Serial.find("OK")){
    Serial.println("Aga Bağlandı.");
}

```

Şekil 13: ESP8266 Wifi Modülünün İnternet Bağlantısı

ESP8266 Wifi modülünün Thingspeak sitesine veri göndermesi sağlandı.

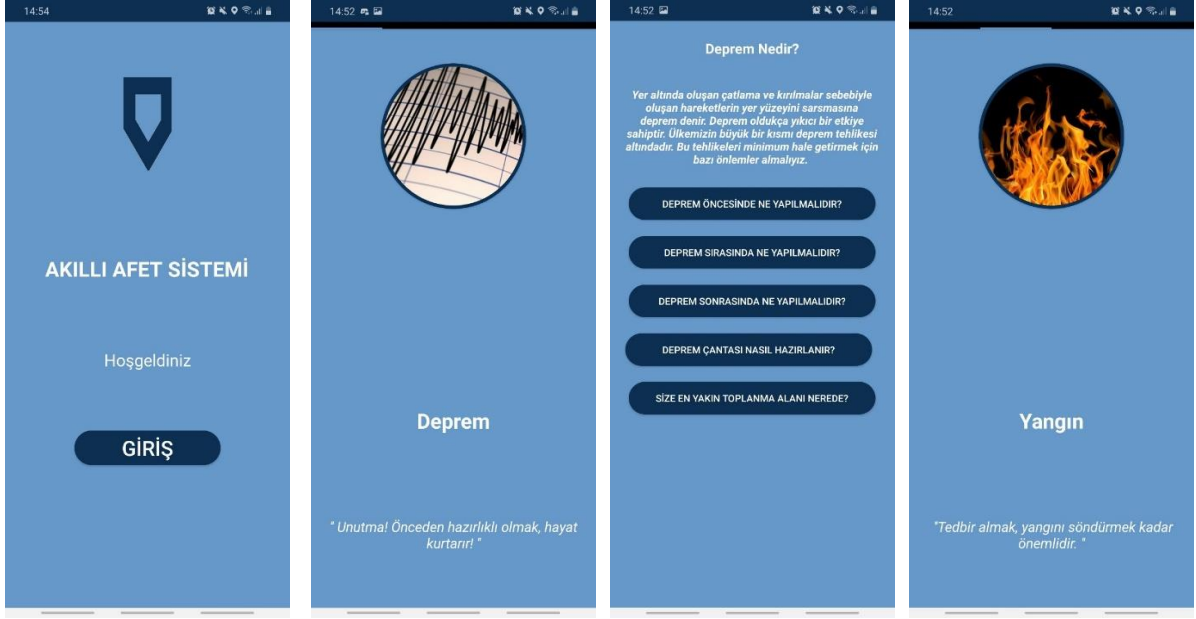
```

void wifi(String field, int durum){
    int i = 0;
    while(i == 0){
        Serial.println("wifi fonksiyonuna girdi");
        Serial.println("AT+CIPSTART=TCP,\""+ip+"\",80");
        if(Serial.find("Error")){
            Serial.println("AT+CIPSTART Error");
        }
        String veri = "GET https://api.thingspeak.com/update?api_key=8LMMX806VAMDC6ZC";
        veri += "&";
        veri += field;
        veri += "=";
        veri += durum;
        veri += "\r\n\r\n";
        Serial.println("esp hazırlanıyor");
        Serial.print("AT+CIPSEND=");
        Serial.println(veri.length()+2);
        delay(100);
        if(Serial.find(">")){
            Serial.print(veri);
            Serial.println("Veri gönderildi.");
            i=1;
        }
        else{
            Serial.println("Bağlantı Kapatıldı.");
            Serial.println("AT+CIPCLOSE");
            delay(100);
        }
    }
}

```

Şekil 14: ESP8266 Wifi Modülü Veri Gönderimi

Mobil uygulama çalışmalarına başlandı ve ara yüz oluşturuldu. (Şekil 4) Afet durumunda ekranda alarm olarak açılması gereken sayfalar yapıldı (Şekil 6).

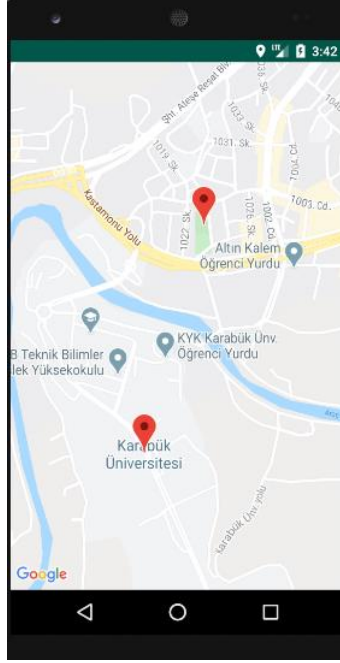


Şekil 15: Ana Sayfa ve Örnek Afet Bilgilendirme Sayfaları



Şekil 16: Afet Anında Mobil Uygulamada Açılacak Sayfalar

Deprem olduğu zaman mobil uygulamada açılacak en yakın toplanma bölgesini göstermesi için gerekli kodlar yazıldı (Şekil 6). İnsanları afetler konusunda bilinçlendirmek için gerekli sayfalar oluşturuldu (Şekil 8).



Şekil 17: Deprem Olduğu Zaman Mobil Uygulamada Kullanıcının Konumunu ve Kullanıcıya En Yakın Toplanma Bölgesinin Konumunun Gösterilmesi

```
public void onMapReady(final GoogleMap googleMap) {
    mMap = googleMap;

    locationManager = (LocationManager) this.getSystemService(Context.LOCATION_SERVICE);

    locationListener = new LocationListener() {
        @Override
        public void onLocationChanged(Location location) {

            mMap.clear();
            LatLng userLocation = new LatLng(location.getLatitude(), location.getLongitude());
            MarkerOptions Location = new MarkerOptions().position(userLocation).title("Your Location");
            mMap.addMarker(Location);

            if (i==true) {
                mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(userLocation, 17));

                String[] getColumnNane = {"_id","il","ilce","mahalle","enlem","boyam","ad"};

                imlec = db.query("toplama_alanlari", getColumnNane, null, null, null, null, null);

                Double min = 50.0;

                LatLng alanLocation;

                imlec.moveToFirst();

                while (imlec.moveToNext()) {

                    Double enlem = imlec.getDouble(imlec.getColumnIndex("enlem"));
                    Double boyam = imlec.getDouble(imlec.getColumnIndex("boyam"));
                    String adres = imlec.getString(imlec.getColumnIndex("ad"));

                    alanLocation = new LatLng(enlem, boyam);
                    Double distance = Math.sqrt(Math.pow((userLocation.latitude - enlem), 2) +
                    Math.pow((userLocation.longitude - boyam), 2));

                    if (distance <= min) {
                        nearest = alanLocation;
                        min = distance;
                        nearest_ad = adres;
                    }
                    i=false;
                }

                MarkerOptions destination = new MarkerOptions().position(nearest).title(nearest_ad);

                mMap.addMarker(destination);
            }

            @Override
            public void onStatusChanged(String s, int i, Bundle bundle) {
            }

            @Override
            public void onProviderEnabled(String s) {
            }

            @Override
            public void onProviderDisabled(String s) {
                db.close();
                imlec.close();
            }
        }
    };

    if (ContextCompat.checkSelfPermission(this, Manifest.permission.ACCESS_FINE_LOCATION) !=
    PackageManager.PERMISSION_GRANTED) {
        {
            ActivityCompat.requestPermissions(this, new String[] {Manifest.permission.ACCESS_FINE_LOCATION}, 1);
        }
        else
        {
            locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, locationListener);
        }
    }
}
```

Şekil 18: Harita Kodlamaları

Thingspeak platformunda gerekli ayarlar yapıldı ve geliştirme kartı ile mobil uygulama arasında veri akışı sağlandı (Şekil 10).

```
@Override
public int onStartCommand(final Intent intent, int flags, int startId) {

    String url = "https://api.thingspeak.com/channels/1011062/feeds.json?api_key=V3AWFFP0RNU2IQ77&results=1";

    RequestQueue requestQueue = Volley.newRequestQueue( context: this);

    JSONObjectRequest objectRequest = new JSONObjectRequest(Request.Method.GET, url, jsonRequest: null, (response) -> {
        Log.e( tag: "Rest Response", response.toString());

        try {
            JSONObject = new JSONObject(response.toString());
            JSONArray = JSONObject.getJSONArray( name: "feeds");
            JSONObject2 = JSONArray.getJSONObject( index: 0);

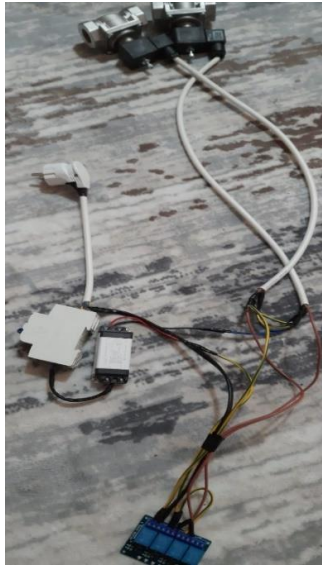
            int gaz = JSONObject2.getInt( name: "field1");
            int su = JSONObject2.getInt( name: "field2");
            int deprem = JSONObject2.getInt( name: "field3");
            int yangin = JSONObject2.getInt( name: "field4");

            if(gaz==0){
                Intent intent = new Intent(getApplicationContext(), DogalgazAlarm.class);
                intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
                startActivity(intent);
            }
            if(su==0){
                Intent intent = new Intent(getApplicationContext(), SuTaskini.class);
                intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
                startActivity(intent);
            }
            if(deprem==0){
                Intent intent = new Intent(getApplicationContext(), DepremAlarm.class);
                intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
                startActivity(intent);
            }
            if(yangin==0){
                Intent intent = new Intent(getApplicationContext(), YanginAlarm.class);
                intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
                startActivity(intent);
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }, (error) -> { Log.e( tag: "Rest Response", error.toString()); }
    );
    requestQueue.add(objectRequest);

    return super.onStartCommand(intent, flags, startId);
}
```

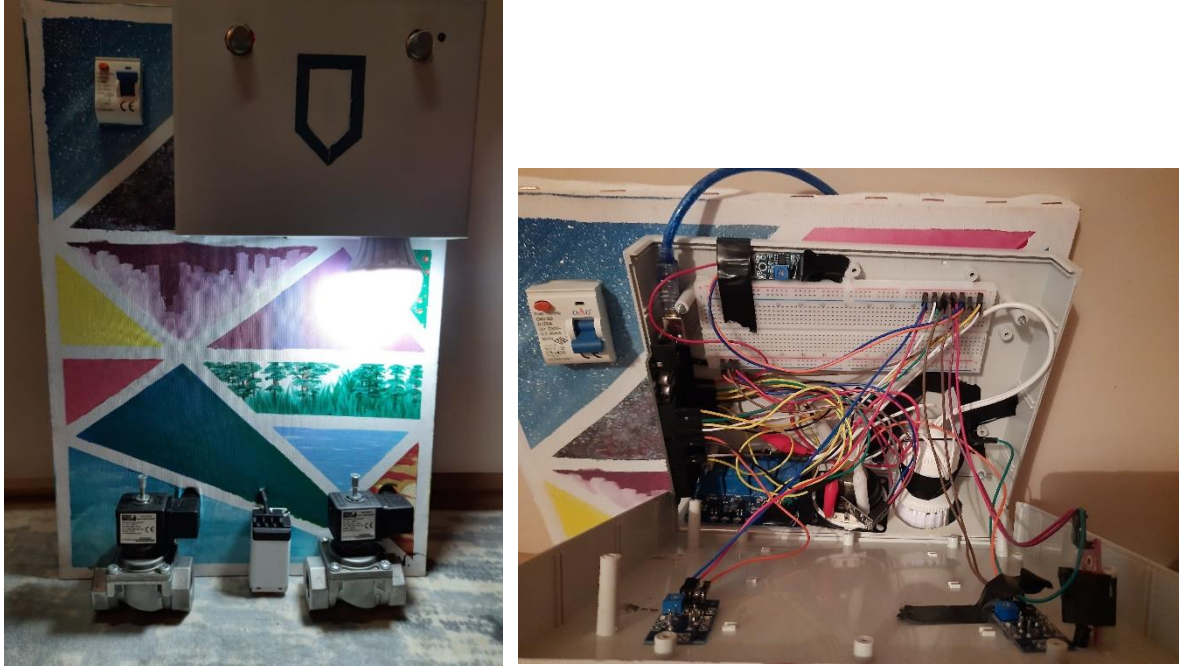
Şekil 10: Thingspeak Mobil Bağlantısı Kodları

Röle ve vanaların devreleri tamamlandı (Şekil 11).



Şekil 11: Tüm Cihaz Bağlantısı

Cihazın kutulanması ve platformu oluşturuldu.



Şekil 12: Cihazın Kutulanmış Hali ve İç Görünümü

4. SONUÇ

Projemiz planlandığı şekilde tamamlanmıştır. Ancak projemiz geliştirilebilir bir projedir. Arduino içerisinde kullandığımız birçok cihaz daha iyisiyle değiştirilebilir ya da farklı bir cihazla aynı işi görebilir. Örnek vermek gerekirse biz projemizde Wifi modül kullandık ancak Wifi yerine GSM modül kullanılarak daha iyi bir sonuç elde edilebilir. Bizim Wifi modül tercih etmemizin nedeni maliyetinin düşük olmasıdır. Bizim projedeki temel amacımız böyle bir cihazın yapılabileceğini ve çalışabileceğini göstermektir.³

5. KAYNAKLAR

1-UN Office for Disaster Risk Reduction (UNDRR). (2019, Ekim). *Disasters 2018: Year in review*. Centre for Research on the Epidemiology of Disasters (CRED) : <https://www.preventionweb.net/publications/view/65061> adresinden alındı.

2-Secty Life Patron (2019, Ekim).

<https://www.secty-electronics.de/en/projects/overview-about-projects-and-references.html>

3-Sismik (2019, Ekim).

<https://www.sismik.com.tr>