

TEKNOFEST

HAVACILIK, UZAY VE TEKNOLOJİ FESTİVALİ

ENGELSİZ YAŞAM TEKNOLOJİLERİ YARIŞMASI

PROJE DETAY RAPORU

PROJE ADI: ENGELSİZ GÖRÜ GÖZLÜĞÜ (EGG)

TAKIM ADI: ŞAH

Başvuru ID: 446246

TAKIM SEVİYESİ: Lise

İçindekiler

1. Proje Özeti (Proje Tanımı)

EGG görme eyleminde sorun yaşayan bireylerin akıllı gözlük teknolojileriyle hayatlarını kolaylaştırmayı çalışıyor, görme eyleminde sorun yaşayan bireylerin hayatlarında yapay zeka destekli bir yardımcı oluyor.



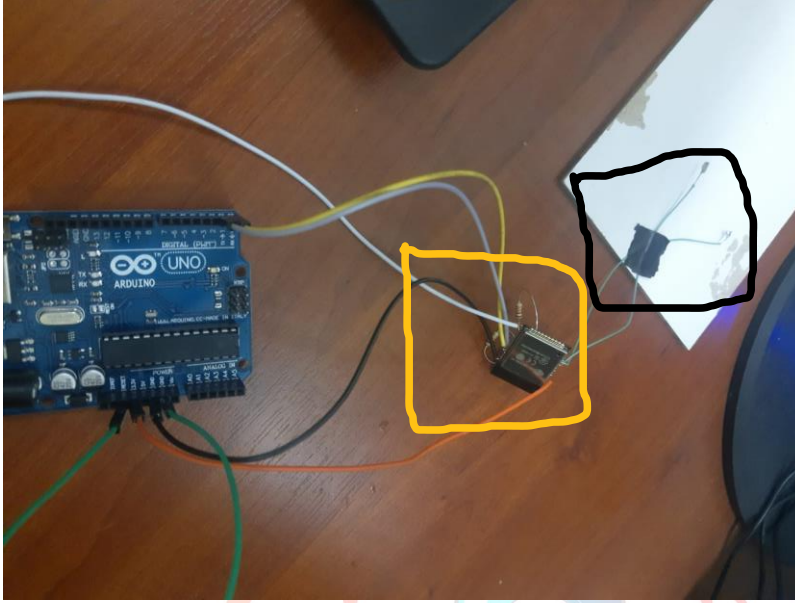
1.1. Belge İçeriği

BU belge şuanları içeriyor

1.2. Tasarım

EGG'nin toplam ağırlığı 93g olmak üzere dış tasarımsal açıdan çok ilkel, gelişimini tamamlamamış halde EGG şuan bir 3d sinema gözlüğünün çerçevesinin alt kısımlarının kesilmesiyle oluşturulmuş kasayı kullanıyor. Üzerinde sağ ve sol olmak üzere iki mantar LED bulunmakta üzerinde laptop batarya kutusundan çıkmış, ömrü azalmış, üzerinde 2Ah yazan bir adet 18650 batarya bulunmakta ve değiştirilebilir olması için 18650 batarya yatağı bulunmakta ve bataryanın şarj edilebilir olması için ve herhangi bir kısa devre vb. durumda korumaya da alması için bms devreli olan şarj devresi TP4056 bulunmakta ama micro usb soketi pil yatağının soluna konumlandırılmış vaziyette -daha kolay şarj edilebilir olması açısından- . Üzerinde bir adet ESP32-WROOM bulunmakta ve bir adet ESP32-CAM olarak adlandırılan geliştirme kartı bulunuyor ESP32'yi tercih etme sebeplerim uygun fiyatlı ve kablosuz iletişim kurma olanaklarını bünyesinde hazır olarak bulundurması (WiFi, Bluetooth 4.2, BLE), daha hızlı olması ve kapasitif ölçüm yapabilmesi. Normalde sadece ESP32-CAM yetecekken bir adet daha ESP32 kullanmamın sebebi ESP32-CAM üzerinde bulunan hafıza kartı soketini kullanınca I/O olarak kullanabileceğim pinin kalmaması ve ESP32-CAM ile WiFi üzerinden yayın yaparken Bluetooth'u kullanmak istersem hata vermesiydi bunun üzerine iki adet ESP32 kullanarak birbirlerini de UART üzerinden haberleştirerek bu sorunu çözdüm. Üzerinde iki adet dokunmatik yüzey bulunuyor bu yüzeyleri ESP32'nin kapasitif ölçüm yapabilen pinleri ile kontrol ediyorum ve hem elimizle daha kolay erişilebilmesi hem de daha stabil çalışması adına yüzeyleri nikel şeritten parça keserek oluşturdum. Ses işlevleri içinse mikrofon kalitesi çok iyi olan, sesi çok iyi veren ve alan ve bu özelliklerine rağmen telefoncularda en ucuz alabileceğiniz seviyede olan bluetooth kulaklığı keşfettim ve kullandım hazır kullanmamın bana katkısı şu oldu hali hazırda ses kalitesi iyi olduğundan sesi yazıya çevirme işlemlerinde daha iyi tanımlama sağladı ama video çekimi yapacağım zaman mikrofon bluetooth kulaklıkta olduğundan ses alamayı eş zamanlı yapamadım. EGG özünde bir akıllı gözlük olduğundan akıllı gözlük gibi de kullanılabilir. Dokunmatik yüzey ilk başta bir adet vardı sadece akıllı gözlük işlevlerini yapıyordu yapay zeka fonksiyonlarını uygulama üzerinden buton ile ya da sesli asistan ile veriyorduk ama bu hem süreyi uzatıyordu hem de yolda giderken sesli bir şekilde söylenmesi

iyi karşımayacağından ikinci yüzeyi ekledim ve test ettim Arduino Uno'yu programlayıcı olarak kullanarak ESP32'ye test programını atıp yüzeyi test ettim.



Sarı işaretli yerde olan şey ESP32, siyah işaretli olan yere dokunmatik yüzeylerimiz.

Yüzeyler şöyle çalışıyordu her birinin 4 işlevi vardı tek dokunma, çift dokunma, üç dokunma ve basılı tutma ama gelin görün ki tek yüzeyde çalışan bu mantık diğer yüzeye olunca çalışmadı aslında sebebi interrupt kullanmamamdı kullanmayı denedim ama kapasitif ölçümde çalıştıramadım bu beni daha iyi bir sisteme itti buradaki sistem sabitti benimde aklıma değişebilir bir sistem geldi yine iki yüzey olacaktı ilk yüzey 4 işlevli olan sistemi kullanacaktı ama kullanırken çağıracağı fonksiyonlar index diye bir değişkene bağlı olacaktı bu sayede indexin değişimiyle dinamik fonksiyonları elde edecektim ikinci yüzey ise indexi arttıracaktı ve basılı tuttuğunda sıfırlayacaktı bunu bu şekilde yaparak dinamik fonksiyonları çalışır bir sisteme döktüm.

Dokunmatik yüzey ve işlevleri index değişimine göre tablodaki gibidir

	İndex = 0	İndex = 1	İndex = 2	İndex = 3
1. Dokunuş	Sesli Asistan	Sesli Asistan	Sesli Asistan	Sesli Asistan
2. Dokunuş	Fotoğraf Çek	Ses başlat/durdur çağırıcı kabul et	Ortam Analizi	Para Tespiti
3. Dokunuş	Yayına Geç	Çağırıcı reddet	Metin Analizi	Yüz Tanıma
Basılı tutma	LED aç/kapa	Kulaklığı aç/kapa	Metin Çevirisi	Şu anda nerdeyim

Sesli asistan fonksiyonu Bluetooth ile bağladığımız telefona veri göndererek kodladığımız sesli asistanı çağırır.

Fotoğraf Çek fonksiyonu sisteme fotoğrafı çekip ESP32-CAM'e taktığımız sd karta fotoğrafı kaydeder ve ismini çekildiği tarih ve saati olarak yazar.

LED aç/kapa fonksiyonu üzerinde bulunan mantar ledlerin durumu her neyse zıttına çevirir.

Ses başlat/durdur fonksiyonu çağrıyı kabul et fonksiyonu kulaklıklardan bildiğimiz play/pause işlevini görür.

Çağrıyı reddet fonksiyonu daha uzun süre kulaklığa tetik vererek çağrıyı reddeder.

Kulaklığı aç/kapa 3 saniye boy kulaklığa tetik vererek kulaklığı açar veya kapatır.

Ortam analizi fonksiyonu telefonun hotspotunu açar, fotoğraf çeker, sunucuya atar ve sunucuda işlenir sonra tespit edilen nesneleri sayılarıyla birlikte döndürür android uygulamaysa döndürülen bilgiyi alır ve okutur.

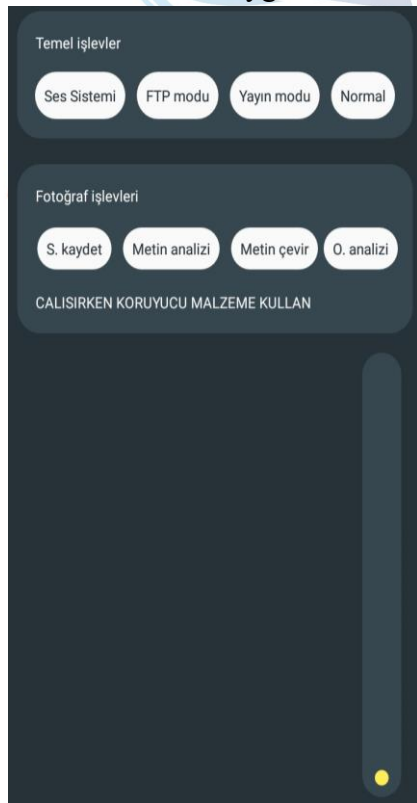
Metin analizi fonksiyonu telefonun hotspotunu açar, fotoğraf çeker, sunucuya atar ve sunucuda işlenir sonra tespit edilen metinleri okutur.

Metin çevirisi fonksiyonu aynı metin analizi ile aynı sistemi kullanır ama sesli okutmadan önce Google çeviriye metni yollar ve türkçeye çevirip okutur.

Para tespiti fonksiyonu telefonun hotspotunu açar, fotoğraf çeker, sunucuya atar ve sunucuda banknotu tespit eder ve sonucu telefona okutur örneğin; “200 türk lirası” şeklinde okutur.

Yüz tanıma fonksiyonu telefonun hotspotunu açar, fotoğraf çeker, sunucuya atar ve sunucuda yüz tespit edilir, yüz eğer evvelden tanıtılmışsa o kişi nasıl kaydedilmişse onun ismini okutur, eğer kaydedilmemişse o kişinin ismini söyleyip kayıt edebiliriz.

Komutları mobil uygulama ile de verebiliriz



S. kaydet butonu fotoğrafı çeker ve linkini verir ve size fotoğrafı gösterir.

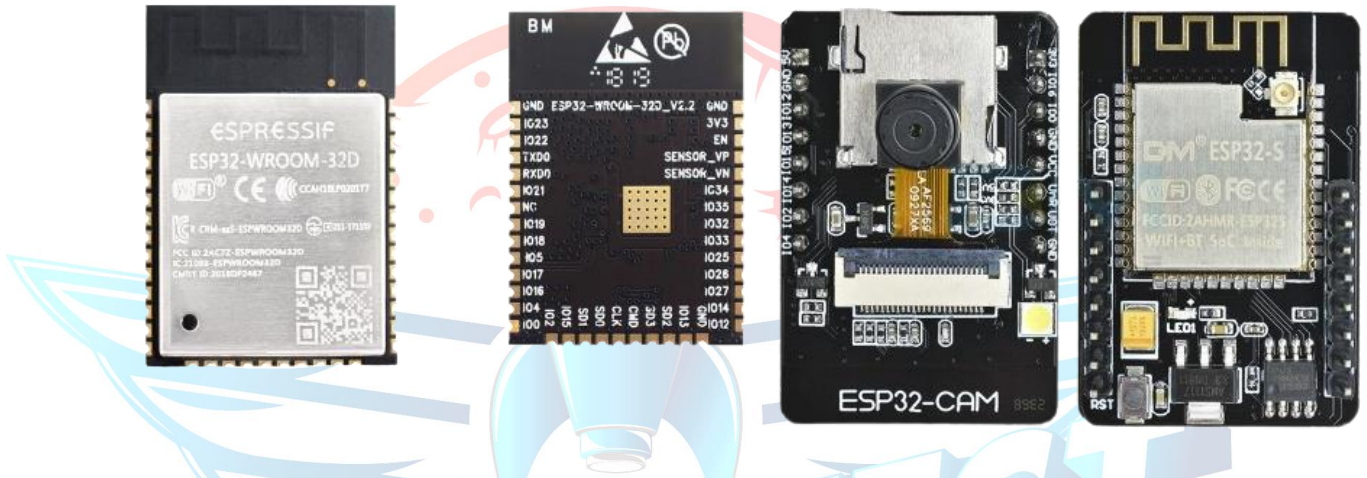
FTP modu butonu telefonun hotspotunu açar, ve gözlüğü ona bağlatır sonra sd kart dizinine erişmemizi sağlar.

Komutları sesli komut ile de verebiliriz ama hem gecikme hem de yanlış anlamalar sebebiyle yavaş ve yavan kalıyor.

Yazılım

Yazılım, tasarımdaki kısımları işletiyor işletme sistemin yukarıdan anlayabileceğiniz gibi yazılım kısmı temel olarak üç kısımdan oluşuyor; gözlükteki yazılım, mobil uygulamadaki ve sunucudaki yazılım.

Gözlüğün Yazılımı



Gözlüğün yazılımını bu iki kartın içerisinde bulunuyor. Bu yazılımları mantıken ikiye ayırabiliriz; ilki kamera, WiFi ve SD kartı ile işlevleri kontrol eden kart yani ESP32-CAM diğeryse ESP32-WROOM bu kart bileşenleri, Bluetooth'tan gelen verileri, Dokunmatik yüzeyleri ve ESP32-CAM'i kontrol eden kart. ESP32WROOM'a gelen veri eğer ESP32-CAM ile alakalı ise UART üzerinden iletir ve eğer oradan veri gelirse o veriyi alır telefona iletilecek ise iletir eğer kendisi kullanacaksa kullanır.

ESP32-WROOM' dan temel kodlar (Kodları okuyabilmek için büyütebilirsiniz)

```
void setup() {
  Serial.begin(115200); // esp32ler in baudrate i 115200 olarak ayarlanıyor
  Serial.setTimeout(10); //serial'den veri gelmesini 10ms bekleyecek
  BLUETOOTH.begin("AkıllıGözlük"); // bt ismi
  BLUETOOTH.setTimeout(10); //bt'den veri gelmesini 10ms bekleyecek
  pinMode(solLed, OUTPUT);
  pinMode(sagLed, OUTPUT);
  pinMode(titresimMotoru, OUTPUT);
  pinMode(blueetoothKulaklik, OUTPUT);
  BLUETOOTH.register_callback(bt_callback_function); //bt olaylarında çağrılacak fonksiyon ayarı
}

void bt_callback_function(esp_spp_cb_event_t event, esp_spp_cb_param_t *param) {
  if (event == ESP_SPP_START_EVT) {
  }
  else if (event == ESP_SPP_SRV_OPEN_EVT) {
    //bağlanıldığında
  }
  else if (event == ESP_SPP_CLOSE_EVT) {
    //çıkıldığında
  }
  else if (event == ESP_SPP_DATA_IND_EVT) {
    BtDenEnSonenKomut = BLUETOOTH.readString(); //gelen komutu kaydediyor
    btKomutuUygula(BtDenEnSonenKomut); //gelen komutu işleyecek fonksiyona parametre geçiyor
  }
}
```

```

int ikiNoktadanSonrakiSayiNedir(String _s){
    return _s.substring(_s.indexOf(":")+1).toInt(); // iki noktadan sonraki sayıyı döndürür
}

void btKomutUygula(String _komut){
    if(_komut.indexOf("solled:") >= 0){ //eğer gelen metin solled: içeriyorsa kontrolu
        solledParlaklikAyarla(ikiNoktadanSonrakiSayiNedir(_komut) );
    }else if (_komut.indexOf("sagLed:") >= 0){
        sagLedParlaklikAyarla(ikiNoktadanSonrakiSayiNedir(_komut) );
    }else if(_komut.indexOf("titresinMotoru:") >= 0){
        titresinMotorunuSur(ikiNoktadanSonrakiSayiNedir(_komut),1);
    }else if(_komut.indexOf("bluetoothKulaklik") >= 0){
        bluetoothKulaklikSur(); // bluetoothKulakligi açıyor
    }else if(_komut.indexOf("zaman:") >= 0){ //zaman ayarı yapıyor
        rtc.setTime(ikiNoktadanSonrakiSayiNedir(_komut));
        String temp = "zaman:"+ikiNoktadanSonrakiSayiNedir(_komut);
        Serial.println(temp);
    }

    }else if(_komut.indexOf("mod") >= 0){ // kamera kartının modunu sorguluyor
        Serial.println("mod");
    }else if(_komut.indexOf("ftpModu") >= 0){ //moda geçiriyor
        Serial.println("ftpModu");
    }else if(_komut.indexOf("normalMod") >= 0){ //moda geçiriyor
        Serial.println("normalMod");
    }else if(_komut.indexOf("yayinModu") >= 0){ //moda geçiriyor
        Serial.println("yayinModu");
    }else if(_komut.indexOf("ipAdressi") >= 0){ // kamera kartının ip adresini sorguluyo
        Serial.println("ipAdressi");
    }else if(_komut.indexOf("zaman?") >= 0){
        BLUETOOTH.println(rtc.getTime("%F %H:%M:%S"));
    }else{
        Serial.println(_komut); //kendisiyle alakalı değilse veriyi kamera kartına yolluyor
    }
}

```

```

void bluetoothKulaklikTask(void *parameter){
    digitalWrite(bluetoothKulaklik,HIGH); //tetik ver
    vTaskDelay(3000 / portTICK_PERIOD_MS); //3sn bekle
    digitalWrite(bluetoothKulaklik,LOW); //tetigi kes
    vTaskDelete( NULL ); // işlemi hafızadan sil
}

void bluetoothKulaklikSur(){
    xTaskCreate(
        bluetoothKulaklikTask, /* işlem fonksiyonu */
        "BluetoothKulaklikTask", /*işlem ismi */
        1000, /* stakteki baytı. */
        NULL, /* parametre */
        1, /* önceliği. */
        NULL);
}

```

```

void komutIndexiKontrolcusu(){ // index dokümanatik yüzeyinin kontrolcusu
    unsigned long baslangictakiSure = millis(); //Kod ilk çalıştığındaki zaman
    unsigned long basilmaSuresi;
    int touchDegeri = touchRead(sw2); //kapasitif ölçüm yapıyo
    if(touchDegeri<touchKalibreDegeri){ //eğer eşik değerinden küçükse tekrar ölçüm yapıyo hatalı ölçmemek amacı ile
        delay(5);
        touchDegeri = touchRead(sw2);
    }
    if(touchDegeri<touchKalibreDegeri){
        touchDegeri = touchRead(sw2);
    }
    while(touchDegeri<touchKalibreDegeri){ //basılı kaldığı sürece kodu kitliyo
        touchDegeri = touchRead(sw2);
    }
    basilmaSuresi = millis() - baslangictakiSure; // sonra basılma süresini ms cinsinden buluyor
    if(basilmaSuresi>1000){ //eğer uzun basıldıysa

        komutIndexi = 0;
        titresinMotorunuSur(1000,1);
    }else if(basilmaSuresi>30){ //eğer kısa basıldıysa
        if(komutIndexi<3){
            komutIndexi++; //indexi bir arttırıyor
        }else{
            komutIndexi = 0;
            titresinMotorunuSur(1000,1);
        }
    }
}
}
}
}

```

ESP32-CAM'den bazı temel kodlar

```

bool fotografCekVeKaydet()
{
    fb = esp_camera_fb_get(); //fotograf ceker fb ye atar
    if (!fb) // çekildi mi kontrolu
    {
        return false;
    }
    String dirPath = "/resimler/" + rtc.getTime("%F"); //klasor ismi
    if (!createDir(SD_MMC, dirPath.c_str())) //oluşturamazsa
    {
        return false;
    }
    String path = dirPath + "/RESIM" + rtc.getTime("_%FZ_%H.%M.%S") + ".jpg"; //resim ismi oluşturur
    fs::FS &fs = SD_MMC;
    File file = fs.open(path.c_str(), FILE_WRITE);
    if (!file)
    {
        return false;
    }
    else
    {
        file.write(fb->buf, fb->len); // sd karta yazdır
    }
    file.close();
    esp_camera_fb_return(fb);
    return true;
}

```

```

void komutUygula(String _komut)
{
    //esp32wroom dan mesaj geldiğinde kod buraya gelir ve işlemleri yapar
    if (_komut.indexOf("ftpModu") >= 0)
    {
        ftpModunaGec();
    }
    else if (_komut.indexOf("normalMod") >= 0)
    {
        normalModaGec();
    }
    else if (_komut.indexOf("yayinModu") >= 0)
    {
        websocketYayinModunaGec();
    }
    else if (_komut.indexOf("mod") >= 0)
    {
        //mod sorgusu atılır
        Serial.println(aktifOlanMod); //modu esp32wroom a yollar
    }
    else if (_komut.indexOf("zaman:") >= 0)
    {
        //zaman ayar yap
        rtc.setTime(ikiNoktadanSonrakiSayiNedir(_komut));
    }
    else if (_komut.indexOf("fotografCek") >= 0)
    {
        if (!fotografCekVeKaydet()) {
            Serial.println("fotografCekilemedi"); //cekilmesse bunu esp32wroom a atar
        } else {
            Serial.println("fotografCekildi");
        }
    }
    else if (_komut.indexOf("ipAdressi") >= 0)
    {
        Serial.println(WiFi.localIP());
    }
    else if (_komut.indexOf("fotografiYukle:") >= 0)
    {
        //esp32wroom dan bu veri gelirse fotografi api ye atar ve hangi amaçla geldiyse onuda sunucuya bildirir
        //fotograf analizi yaptıracağız böyle komut vermeliyiz "fotografiYukle:fotograf-analizi"
        cekVeSunucuyaYukle(_komut.substring(_komut.indexOf(":") + 1, _komut.indexOf('/')));
    }
}

```

```

void ftpModunaGec()
{
    preferencesNesnesi.putString("aktifOlanMod", "ftpModu"); //kalıcı hafızaya ftpModu diye deger atar
    disableWifi();
    Serial.println("kartKapandi");
    ESP.restart(); // yeniden baslatir
    //yeniden baslatmadaki amac isletim sistemi olmadigindan degiskenler kullanmayinca silinmiyor
    //bu da hatalara yol aciyor yeniden baslatınca hatalar gidiyor temiz bir bellekle basliyoruz
}

void kaydedilenAyarlariAyarla() {
    preferencesNesnesi.begin("mod", false);
    aktifOlanMod = preferencesNesnesi.getString("aktifOlanMod"); //kayıt edilmiş modu hafızaya alır
}

void setup()
{
    WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0); //guc hatalarini onlemek amaclı
    kaydedilenAyarlariAyarla(); //
    Serial.begin(115200);
    Serial.setTimeout(10);
    pinMode(19, INPUT_PULLUP); //sd kart hatalarini cozmek icin
    SD_MMC.begin(); //sd karti baslatir
    //eger kayit edilmiş mod ftp modu ise kod bu if de kitlenir
    if (aktifOlanMod == "ftpModu")
    {
        ...
    }
    //degilse ve yayin modu ise burda kitlenir
    else if (aktifOlanMod == "websocketYayinModu")
    {
        ...
    }
    //buraya gelince kod normal mod demektir
    kameraInit();
    Serial.println("kartBasladi");
}

```

Mobil uygulama kodları

Mobil uygulama kodları temel olarak gözlükten gelen verileri alır işleve çevirir ya da emir verir ve sunucudan gelecek olan yanıtı bekler işlevler yaptırır

Arayüz kontrollerinin gozluğe aktardığımız kodlar

```

btn1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View _view) { //butona tıklanınca fotoyu sunucuya kaydediyö
        bt.sendData(bt_bluetooth_connection_listener, "fotografiYukle:sunucuya-yukle/", "A");
        sgkbutonla = "kaydet";
    }
});

btn2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View _view) { //butona tıklanınca metin analizi yapıyor
        bt.sendData(bt_bluetooth_connection_listener, "fotografiYukle:ocr/", "A");
        sgkbutonla = "ocr";
    }
});

btn3.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View _view) { //butona tıklanınca metin çevirisi yapıyor
        bt.sendData(bt_bluetooth_connection_listener, "fotografiYukle:ocr/", "A");
        sgkbutonla = "metin çevirisi";
    }
});

btn4.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View _view) { //butona tıklanınca ortal analizisi yapıyor
        bt.sendData(bt_bluetooth_connection_listener, "fotografiYukle:fotograf-analizi/", "A");
        sgkbutonla = "analizi";
    }
});

_bt_bluetooth_connection_listener = new BluetoothConnect.BluetoothConnectionListener() {
    @Override
    public void onConnected(String _param1, HashMap<String, Object> _param2) {
        final String _tag = _param1;
        final HashMap<String, Object> _deviceData = _param2;
        SketchwareUtil.showMessage(getApplicationContext(), "baglandi");
    }

    @Override
    public void onDataReceived(String _param1, byte[] _param2, int _param3) {
        final String _tag = _param1;
        final String _data = new String(_param2, Offset: 0, _param3);
    }
};

seekbar1.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
    @Override
    public void onProgressChanged(SeekBar _param1, int _param2, boolean _param3) {
        final int _progressValue = _param2;
    }

    @Override
    public void onStartTrackingTouch(SeekBar _param1) {
    }

    @Override
    public void onStopTrackingTouch(SeekBar _param2) { //cubukten ayar yapınca led parlakligini ayarlıyor
        bt.sendData(bt_bluetooth_connection_listener, "ledleriDegereGetir:".concat(
            String.valueOf((long)(seekbar1.getProgress()))), "");
    }
});

textView1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View _view) { //bt kulakligi açma emri veriyor
        bt.sendData(bt_bluetooth_connection_listener, "bluetoothkulaklik", "");
    }
});

textView2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View _view) { //ftp moduna geçme emri veriyor
        bt.sendData(bt_bluetooth_connection_listener, "ftpModu", "");
    }
});

textView3.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View _view) { //yayinModuna geçme emri veriyor
        bt.sendData(bt_bluetooth_connection_listener, "yayinModu", "");
    }
});

textView4.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View _view) { //normalModa geçme emri veriyor
        bt.sendData(bt_bluetooth_connection_listener, "normalMod", "");
    }
});

```

Sunucu yanıtına göre işlem yapan kodlar

```

_request_request_listener = new RequestNetwork.RequestListener() {
    @Override
    public void onResponse(String _param1, String _param2, HashMap<String, Object> _param3) { //yanıt geldiğinde
        final String _tag = _param1;
        final String _response = _param2;
        final HashMap<String, Object> _responseHeaders = _param3;
        sunucuYaniti = _response;
        sunucuYaniti = sunucuYaniti.substring((int)(1), (int)(sunucuYaniti.length() - 1)); //başlarındaki tırnakları kaldırıyorum
        if (!sunucuYaniti.equals(sonGelenYaniti)) { //eger yeni bir yanıtla işlem yaptırıyorum
            if (sgkbutonla.toLowerCase().contains("metin çevirisi")) { //eger verilen emir metin çevirisi ise
                if (cevirdiMi) {
                    TranslateAPI translate = new TranslateAPI();//cevirme işlemine başla
                    language.ENGLISH, //ing den
                    language.TURKISH, //turkçeye
                    sunucuYaniti);
                    translate.setTranslateListener(new TranslateAPI.TranslateListener() {
                        @Override
                        public void onSuccess(String translateText) { //becerilebilirse
                            cevirdiMi = true;
                            textView11.setText(translateText); //ekrana yazdırıyorum
                            tts.speak("ceviri yaniti", ".concat(translateText.toLowerCase()), TextToSpeech.QUEUE_ADD, null); //okutuyor
                        }
                        @Override
                        public void onFailure(String translateError) {
                            textView11.setText(translateError); //hatayı yazdırıyor
                            tts.speak("metin sorunu", TextToSpeech.QUEUE_ADD, null); //metin sorunu diye hata mesajı okutuyor
                        }
                    });
                }
            }
        }
        else {
            textView11.setText(sunucuYaniti); //ekrana sunucu yanitini yazdırıyorum
            intent.setAction(Intent.ACTION_VIEW);
            intent.putExtra("img", sunucuYaniti);
            intent.setClass(getApplicationContext(), ImageViewerActivity.class);
            if (sunucuYaniti.contains("jpg")) { //eger .jpg iceriyorsa dosya resim mi degilmi kontrolunu yapıyorum
                tts.speak("fotoğraf hazır", TextToSpeech.QUEUE_ADD, null); //fotoğraf hazır dediriyor
                startActivity(intent); //fotoyu gösteriyor
            }
            else {
                if (!sunucuYaniti.equals(sonGelenYaniti)) { //eger resim degilse ve daha evvel okunmamisla
                    tts.speak(sunucuYaniti.toLowerCase(), TextToSpeech.QUEUE_ADD, null); //oku diyor
                }
            }
        }
    }
};

sonGelenYaniti = sunucuYaniti;

```

Sunucu kodları

Sunucu kodları Amazon Web Services de çalışmaktadır. AWS'nin birçok nimetinden yararlanmaktadır. Amazonu tercih etmemim sebebi 12 ay boyunca ücretsiz deneme hakkı

sunmasıdır.

Fotoğraf işleme işlevleri

Gözlükten bir API isteğinde bulununca istek API Gateway'a ulaşır

```
1 {
2   "amac" : "$input.params('amac')",
3   "base64Image": "$input.body"
4 }
```

Ulaştıktan sonra veri yukarıdaki gibi Json'a çevrilir. İstek atılırken '/' karakterinden sonraki metin "amac" değişkenine atılır, yüklenen fotoğrafsa "base64Image" değişkenine atanır. Oluşturulan Json verisi AWS Lambda ya gelir yani JavaScript kodumuzun çalışacağı yere

```
exports.handler = (event, context, callback) => {
  let amac = event.amac;
  let encodedImage = event.base64Image;
  let decodedImage = Buffer.from(encodedImage, 'base64');
  var d = new Date();
  d.setHours(d.getHours() + 3);
  if (amac.includes("sunucuya-yukle")) {
    else if (amac.includes("ocr")) { //eğer amac metin tespiti ise çalışır
      const ocr = new TEXTTRACT(); //texttract sınıfından nesne oluşturur
      const params = {}; //gondereceğimiz parametrelerin değişkeni

      ocr.detectDocumentText(params, function(err, data) { //tanıma işlemini başlatıyoruz
        if (err) {
          else {
            let textOfImage = "";
            var yalnızArray = data.Blocks;
            console.log(yalnızArray);
            for (var i = 0; i < yalnızArray.length; i++) { //algılanan kelimelerin hepsini boşluk koyup birbirine ekliyoruz
              if (yalnızArray[i].BlockType == "WORD" && yalnızArray[i].Confidence > 39) {
                textOfImage += (yalnızArray[i].Text + " ");
              }
            }
            firebaseAdmin.database().ref().set({ //firebase ye degeri yazdırıyoruz mobil uygulama için
              amaci: amac,
              sunucu_yaniti: textOfImage
            });
            let response = {};
            callback(null, response); //http yanıtını döndürüyoruz
          }
        });
      });
      let Filename = d.toString(); // dosya ismini bugünün tarihi yapıyoruz
      let Folder = "ocr-isleminden-fotolar/" + d.getFullYear().toString() + "-" + (d.getMonth() + 1).toString() + "-" + d.getDate().toString(); //klasor ismi ayarlanıyor
      var filePath = Folder + "/" + Filename + ".jpg";
      var pparams = {
        "Body": decodedImage,
        "Bucket": "depolama-alani",
        "Key": filePath
      };
      s3.upload(pparams, function(err, data) {}); //resimi kaydediyoruz
    }
  }
  else if (amac.includes("para-tespiti")) {
  }
  else if (amac.includes("fotoğraf-analiz")) {
  }
  else if (amac.includes("yuz-analiz")) {
  }
};
```

Burada amaca göre işlemler yaptırılır ve her işlemin sonucunda yüklenen fotoğraf S3 Bucket'ta saklanır.

Objects (4)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Find objects by prefix

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	gozluk-resimleri/	Folder	-	-	-
<input type="checkbox"/>	ocr-isleminden-fotolar/	Folder	-	-	-
<input type="checkbox"/>	ortam-analizinden-fotolar/	Folder	-	-	-
<input type="checkbox"/>	para-tespiti-isleminden/	Folder	-	-	-

Fotoğraf yüklenince güncellenmiş yanıt

```
amac: "sunucuya-yukle"
sunucu_yaniti: https://depolama-alani.s3.eu-west-2.amazonaws.com,
```

Yayın modu

Yayın yapabilmesi için bir Web Scket oluşturmamız lazım bunun için AWS'nin EC2, yani sanal bilgisayar hizmetini kullanıp aşağıdaki kodları çalıştırıyorum.

```

const path = require('path');
const express = require('express');
const WebSocket = require('ws');
const app = express();

const WS_PORT = 65080;
const HTTP_PORT = 88;

const wsServer = new WebSocket.Server({port: WS_PORT}, () => console.log('websocket portu'));

let connectedClients = [];
wsServer.on('connection', (ws, req) => {
  console.log('baglandi');
  connectedClients.push(ws);

  ws.on('message', data => {
    connectedClients.forEach((ws,i) => {
      if(ws.readyState === ws.OPEN){
        ws.send(data);
      } else {
        connectedClients.splice(i, 1);
      }
    });
  });
});

app.get('/client', (req, res) => res.sendFile(path.resolve(__dirname, './client.html')));
app.listen(HTTP_PORT, () => console.log('HTTP server listening at ${HTTP_PORT}'));

```

```

<html>
  <head>
    <style>_</style>
  </head>
  <body>
    <img src(unknown) height="320" width="480" style="transform:rotate(270deg);">
    <script>
      const img = document.querySelector('img');
      const WS_URL = 'ws://x.x.x.x:65080';
      const ws = new WebSocket(WS_URL);
      let urlObject;
      ws.onopen = () => console.log('Connected to ${WS_URL}');
      ws.onmessage = message => {
        const arrayBuffer = message.data;
        if(urlObject){
          URL.revokeObjectURL(urlObject);
          urlObject = URL.createObjectURL(new Blob([arrayBuffer]));
          img.src = urlObject;
        }
      }
    </script>
  </body>
</html>

```

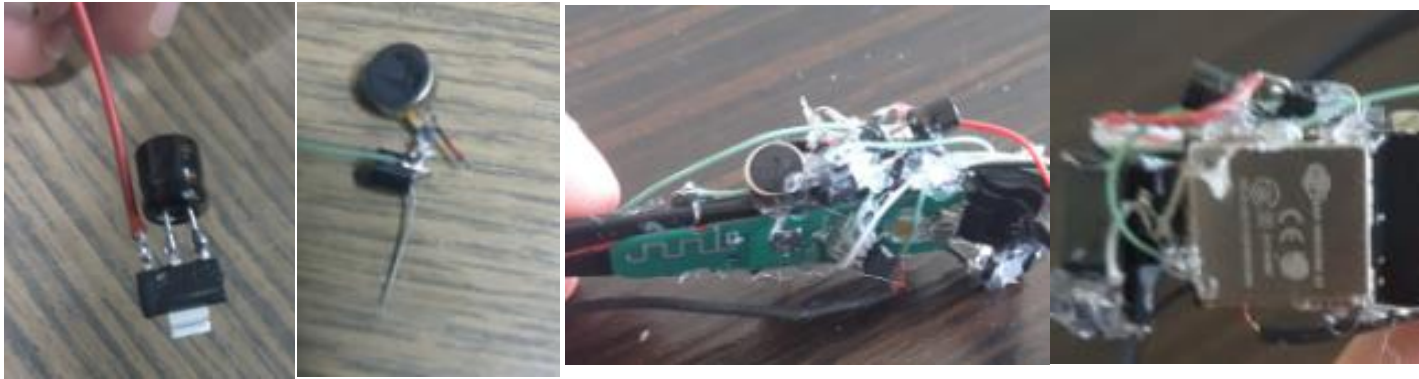
Eğer ssh ortamından çıkarsam kod çalışmıyordu araştırdım ama bulamamıştım teknik bir bilgiye bakarken nasıl çalışacağını buldum “Nohup command [arguments]: Sistemden çıktıktan sonra çalışmasına devam edecek şekilde verilen komutu çalıştırır.”[1]

Montaj

İlk olarak Bluetooth kulaklık kartına BC237 NPN transistörümü lehimledim base ucundan ESP32 ile tetik vereceğiz sonra kulaklık güç hattını bağladım ve kulaklığını bağladım ve sabitledim.



ESP32-WROOM'a gerekli dirençleri ekledim ve voltaj regülatörü devresini yaptım sonra ledleri bağladım, sonra titreşim motorunun önüne bir BC237 koyarak titreşim motoru devresini oluşturdum ve dokunmatik yüzeyin nikel şeritini bağladım ve sistemi sabitledim.



Sonra ESP32-CAM'i monte ettim ve iki kartı tx-rx rx-tx olacak şekilde birbirine bağladım. Pil yatağı ile sürgülü anahtar bağlantılarını yaptım oradan gelen hattı step-up a bağladım şarj için devremi bağladım micro-usb soketimi pil yatağının yanına yerleştirdim ve çıkan kabloları gereken yerlere bağladım.



Son olarak gözlük şu şekilde gözüktü



2. Problem Durumunun Tanımlanması:

Görme eyleminde sorun yaşayan bireylerin problemlerini çözmek amacıyla gözlüğü geliştirdim ve geliştiriyorum yaşadıkları problemler; Tek başlarına özgür birey olarak hareket edememeleri, ayrımcılığa tabi tutulmaları, dolandırılmaları , ticaret yapamamaları, normal bir bireyin yaparken onların yapamadığı her şeyi çözmeye çalışıyorum bu gözlük ile Yapılmış projelerin çoğu bir bireyin çoğu ihtiyacını karşılamak konusunda yetersiz akıyor örneğin yapılmış olan nesnenin ya da elbisenin rengini söyleyen aletler çoğunlukla tek fonksiyonellikten yetersiz kalmakta ya da mesafe ye göre öten gözlükler de çünkü tek fonksiyonları oluyor

3. Çözüm

Görme eyleminde sorun yaşayan bireylerin problemlerini çözmek amacıyla ben gözlük sistemine kamera takıp yapay zeka ile analiz ettirip bu durumları minimize etmeyi çözüm olarak buldum

Örneğin bir görme eyleminde sorun yaşayan birey ticaret yapmaya çalışırsa muhtemelen dolandırılıcağı ya da okumak istediği her yazı onların okuyabileceği yazılmıyor o sebepten okuyamayacaktır bu sebepten kamera ile tespit yapıp ona anlatmak tarif etmek çözüm olacaktır



Şu anda gözlükle verilen komutlar ile sorunlara çare olmakta tasarımsal olarak daha ilkel aşamalarda ama geliştirilebilirliği çok yüksek

4. Yöntem

Kullandığım yöntem gözlüğe kamera takıp onu sunucuda işleyerek çözmeye çalışıyor. Yöntemim bilgisayar sistemleri ile yapay zekanın birleşimi sonucu akıllı gözlük sistemleri üzerine kurgulu. Yaptığım prototipte yaptıklarımın işe yaradığını ama gecikmenin çok fazla olduğunu gördüm 30 saniyelere çıkmakta ve düşük ışıktaki yetersiz çalıştığını gördüm.

5. Yenilikçi (İnovatif) Yönü

Piyasa da gördüğüm ürünlerde mesafesel olarak ya da tek fonksiyonlu olarak yapılan sistemler gördüm

Projenin özgün yönleri belirtilmelidir. Ek olarak, tasarlanan özgün ürünün donanımsal ve yazılımsal parçaları hakkında(varsa) bilgi verilmelidir.

5.1. Çalışmada, kodlarda ve tasarımda yenilikçi bir yön varsa mutlaka belirtilmelidir. Takımlar bu alanda kendi çalışmalarını nasıl bir yöntemle hazırladığını, benzer çalışmalardan hangi yönleri ile ayrıldığını açık ve net bir şekilde detaylı olarak belirtmelidir.

6. Uygulanabilirlik

6.1. Proje fikrinin hayata nasıl geçirileceği ve mevcut şartlar altında projenin ticari bir ürüne dönüştürülebilir olup olmadığı hakkında bilgi verilmelidir. Uygulanabilir olduğunda risk analizi yapılarak, mevcut risklerin neler olduğu da belirtilmelidir. (Risk analizi 9.bölümde detaylandırılabilir.)

6.2.

7. Tahmini Maliyet ve Proje Zaman Planlaması

Projenin en az maliyetle uygulanabilir olma durumu ve tahmini bütçesi hakkında bilgi verilmelidir.

7.1. Bu kısımda ürünün tasarım, üretim ve test süreçlerini içeren bir zaman planlaması ve kullanılacak malzeme listesi hazırlanmalıdır.

7.2. Proje planına bağılı olarak mali analiz gerçekleştirilmeli, hangi harcamaların hangi dönemde yapılacağı yazılmalıdır.

Piyasada benzer projeler varsa piyasadaki diğerk projelerle birlikte maliyet karşılaştırması yapılmalıdır.

7.3. Proje takvimi bir zaman çubuğı üzerinde gösterilebilir. Ancak çok karmaşık olmamasına özen gösterilmelidir.

8. Proje Fikrinin Hedef Kitlesi (Kullanıcılar):

8.1. Projenin kimler tarafından kullanılacağı ve kimlere hitap edeceği burada belirtilmelidir.

8.2. Problemi yaşayanların kimler olduğu hakkında kısaca bilgi verilmelidir.

9. Riskler

Projeyi olumsuz yönde etkileyecek unsurların (risklerin) tespit edilmesi gerekmektedir.

Proje hayata geçirilirken ortaya çıkabilecek problemler tanımlanmalıdır.

Proje hayata geçirilirken ortaya çıkabilecek problemlere yönelik tedbirlerin, çözüm önerilerinin

(B Planı) tanımlaması yapılmalıdır.

Zaman planlamasında iş paketleri, iş tanımları ve süreçleri ayrıntılı bir şekilde açıklanmalıdır.

Zamanlama ve bütçe planlaması ile ilgili olarak risk analizleri yapılmalıdır.

Risk planlamasında olasılık ve etki matrisi eklenmelidir. Olasılık ve Etki matrisi, her riskin gerçekleşme olasılığını ve proje hedeflerine etkisini değerlendirmeye yarayacaktır. Olasılık ve etki belirli gruplara (Az-Normal-Çok vb) ayrılmalıdır.

10. Kaynaklar

[1] 2016, Teknik Bilgiler, EMO AJANDASI,108.syf

RAPOR TASLAKLARI İLE İLGİLİ NOT:

- Yukarıda yer alan **ilk 9 madde en fazla 10 (On)** sayfada anlatılacaktır.
- **Kapak, açıklama ve görsel olmak üzere en fazla 15 sayfa olacaktır. 15 sayfayı geçen raporlar değerlendirmeye alınmayacaktır. (Kaynakça ve içindekiler sayfa sayısına dahil değildir.)**
- **Tüm raporlar akademik rapor standartlarına uygun olarak yazılmalıdır.**
- **Her rapor “kapak” ve “içindekiler” sayfası içermelidir.**
- **Yazı tipi: Times New Roman, Punto: 12, Satır Aralıkları: 1,15 , İki tarafa yaslı, Sayfa kenar boşlukları üst-alt-sağ-sol 2,5 cm olmalıdır.**
- **Rapor içindeki cümleler birbirinin aynı ve tekrarı niteliğinde olmamalıdır.**
- **Raporunda, web sitemizde yer alan geçmiş yıl raporlarından yararlanmış olan takımlar alıntı yaptığını ilgili sayfada belirtmesi gerekmektedir. Açıklamayı alıntı yapılan cümlelerin ardından belirtmeniz gerekmektedir.**
ALINTI FORMATI: "Alıntı yapılan Cümle/ler" (Yıl, Yarışma Adı, Kategori, Takım Adı)
ÖRNEK ALINTI: "Enkazda depremzedenin nerede olduğunu tespit edilememesi, enkaz kaldırma ve deprem-zede arama çalışmalarını yavaşlatan en önemli sorundur." (2020, İnsanlık Yararına Teknoloji Yarışması, Afet Yönetimi, X Takımı)
- **Kaynakçada referanslar aşağıdaki şekilde belirtilebilir.**
Dijital Kaynak: Yazarların Soyadı, Adlarının Baş Harfi., Yazının Başlığı, Yazının Tarihi, Erişim Tarihi, Erişim Adresi.
Basılı Kaynak: Yazarların Soyadı, Adlarının Baş Harfi., (Basım Tarihi) Yazının Başlığı, (Varsa) Yayınlandığı Derginin Adı, (Varsa) Derginin Sayısı, Sayfa numarası.