



T.C.  
İSTANBUL AYVANSARAY ÜNİVERSİTESİ  
MESLEK YÜKSEKOKULU  
BİLGİSAYAR TEKNOLOJİLERİ BÖLÜMÜ  
BİLGİSAYAR PROGRAMCILIĞI PROGRAMI

**SES İLE KOMUT İŞLEME**

**BİLGİSAYAR PROGRAMCILIĞI PROGRAMI  
MESLEK YÜKSEKOKULU  
BİTİRME PROJESİ**

Enes KILIÇ

Proje Danışmanı  
Öğr. Gör. Şahin AYDIN

## **TEZ ONAYI FORMU**

Kurum : İstanbul Ayvansaray Üniversitesi Meslek Yüksekokulu

Programı : Bilgisayar Programcılığı

Proje Sahibi : Enes KILIÇ

Proje Başlığı : Ses İle Komut İşleme

Sınav Yeri :

Sınav Tarihi :

Bitirme projesi tarafımızdan okunmuş, kapsam ve kalite yönünden kabul edilmiştir.

**Danışman**

**Kurumu**

**İmza**

Öğr. Gör. Şahin AYDIN

**Sınav Jüri Üyeleri**

Yukarıdaki jüri kararı Bölüm Başkanlığının ...../...../..... tarih ve .... sayılı kararı ile onaylanmıştır.

Öğretim Görevlisi .....

..... *Bölümü Başkanı*

## **PROJE TANITIM FORMU**

**YAZAR ADI SOYADI** : Enes KILIÇ

**PROJE DİLİ** : Türkçe

**PROJENİN ADI** : Ses ile Komut İşleme

**BÖLÜM** : Bilgisayar Teknolojileri Bölümü

**ANA BİLİM DALI** : Bilgisayar Programcılığı

**PROJENİN TÜRÜ** : Meslek Yüksekokulu Bitirme Projesi

**PROJENİN TARİHİ** : 05.06.2018

**SAYFA SAYISI** : 95

**PROJE DANIŞMANI** : Öğr. Gör. Şahin AYDIN

**DİZİN TERİMLERİ** :

**TÜRKÇE ÖZET** :

**DAĞITIM LİSTESİ** : İstanbul Ayvansaray Üniversitesi Meslek Yüksekokuluna

Enes KILIÇ

## **BEYAN**

Bu projenin hazırlanmasında bilimsel ahlak kurallarına uyulduğu, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğu kullanılan verilerde herhangi tahrifat yapılmadığını, projenin herhangi bir kısmının bu üniversite veya başka bir üniversitedeki başka bir proje olarak sunulmadığını beyan ederim.

Enes KILIÇ

05/06/2018

## **ÖNSÖZ**

Günümüz teknolojisinde insanlar artık yavaş yavaş elektronik cihazlara dokunmadan onları yönetmek istemesi ile daha kolay bir şekilde makine-insan ilişkisini sağlamaya çalışmışlardır. Teknolojinin gelişme sürecinde insanlar ilk önce basmak için tuşlar yapmış daha sonradan tuşların kullanıssız olduğunu anlayıp dokunmatik yapılar, donanımlar kullanmaya çalışmışlardır. Hatta ileriye doğru bu donanımları da kaldırıp hologram yapılar ile makinelere hükmetmeye çalışmışlardır. Ben de projemin konusunu, yine şuanda insan-makine haberleşmesinde popüler olarak kullanılan sürekli geliştirilen ses ile komut işlemeyi ele almak istedim.

# **İÇİNDEKİLER**

<b>1. BÖLÜM I .....</b>	8
1.1. Giriş ve Amaç .....	8
1.2. Kullanılan Teknoloji ve Programlar .....	8
<b>2. BÖLÜM II ARDUINO.....</b>	9
1.3. Giriş .....	9
1.4. Kullanılan Ekipmanlar .....	9
1.5. Devre Şeması.....	9
1.6. Cihaz Özellikleri .....	10
1.6.1. Müzik Çalma Modu .....	10
1.6.2. Televizyon Kumandası Modu .....	14
1.6.3. Hava Sıcaklığı Özelliği.....	15
1.7. Kodlama.....	16
1.7.1. LCD Ekran ve SD Kart Modülü.....	16
1.7.1.1. Global.....	16
1.7.2. Setup ve Loop .....	16
1.7.2.1. Setup .....	16
1.7.2.2. Loop.....	16
1.7.3. Ses Çalma Mekanizması .....	17
1.7.3.1. Setup .....	19
1.7.3.2. SoundPlay .....	20
1.7.3.3. Bluetooth ile Ses Aktarımı.....	24
1.7.4. Mod Atama ve Kullanım Mekanizması.....	25
1.7.4.1. Müzik Çalma Modu.....	27
1.7.4.2. Sıcaklık Öğrenme Modu.....	33

<b>3. BÖLÜM III ANDROİD UYGULAMASI .....</b>	39
1.8. Giriş .....	39
1.9. Bluetooth Bağlantı Arayüzü.....	39
1.10. Bluetooth Bağlantı Eklemesi.....	43
1.11. MainActivity(Ana Arayüz) .....	45
1.12. Sanatçılara Göre Şarkılar Arayüzü.....	52
1.13. Şarkı Listesi Arayüzü.....	63
1.14. Seçilen Şarkı Arayüzü.....	68
1.15. Hızlı Not Alma ve Hatırlatıcı Arayüzü .....	78
1.16. Televizyon Kumandası Arayüzü.....	86
1.17. Son Sözler .....	93
<b>4. TEŞEKKÜRLER.....</b>	94
<b>5. KAYNAKÇA.....</b>	95

# **BÖLÜM I**

## **1.1. Giriş ve Amaç**

İnsan-Makine haberleşmesinde kullanılan ses ile haberleşme, insanların çok daha kolay bir şekilde makinalara hükmedilmesini sağlamıştır. Ben de ağırlıklı bilgisayar olmak üzere makinaların dilini öğrendiğim Bilgisayar Programcılığı bölümü bitirme projemde makinaların anlayabileceği şekilde insanların söylediği kelimeleri nasıl anlaşılır hale getirebilirim bunun için araştırmalar yaptım. Ardından sesli komutları hangi cihazda nerede kullanacağımı araştırdım. Bunun için hangi programlama dilleri kullanacağıma karar verdim. Amacım sadece ses ile değil bir Android uygulaması ile de kullanışlı hızlı bir şekilde cihaza komut göndererek, gönderilen bu komutun doğru algoritmalar ile cihazın anlayabileceği hale dönüştürmek ve yine bu algoritmalar ile sesi işleyip cihazın özelliklerinden faydalananmak istedim.

## **1.2. Kullanılan Teknoloji ve Programlar**

Android Studio – Java

Arduino IDE 1.8.5 - C ve C++

## BÖLÜM II ARDUINO

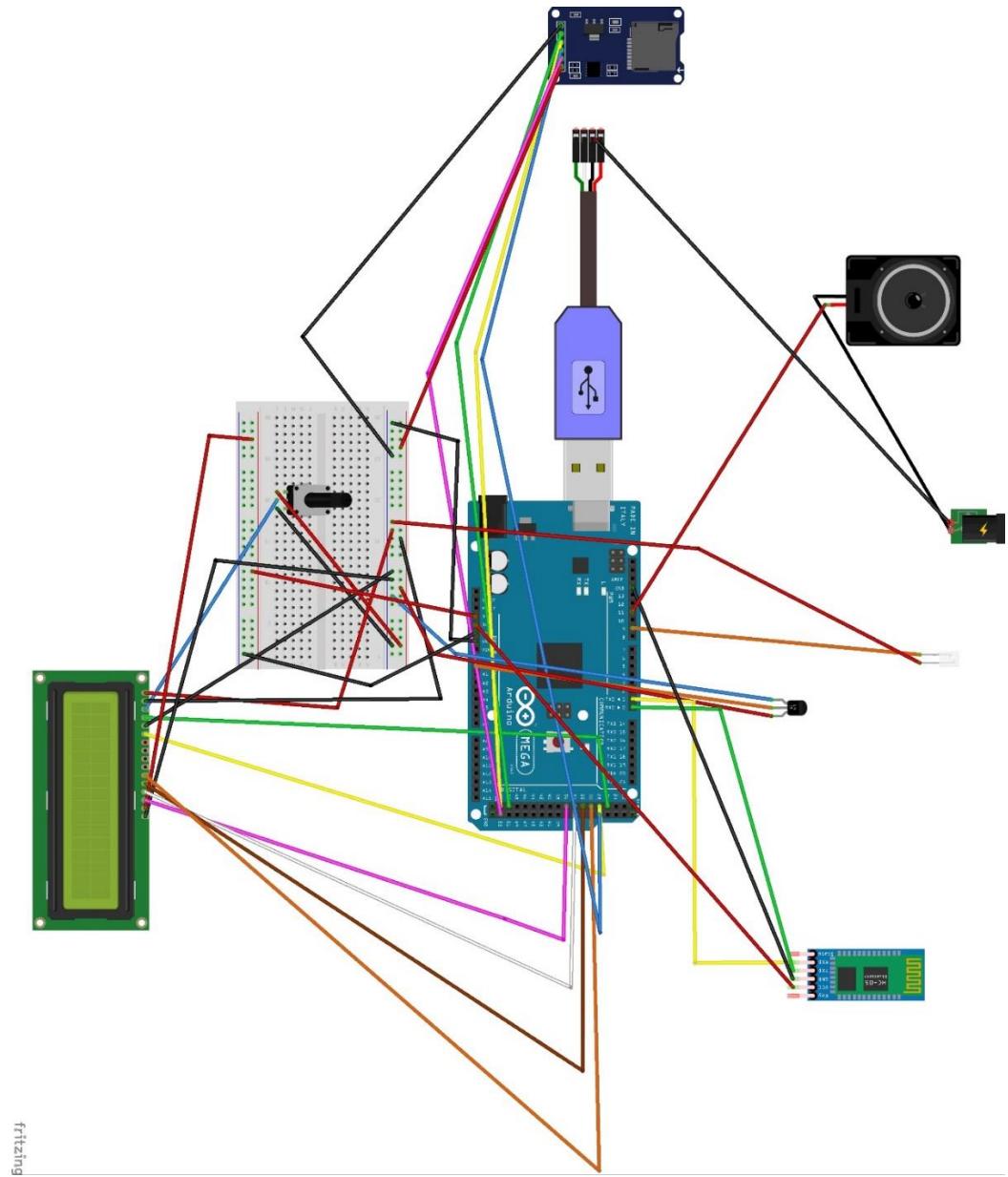
### 1.3. Giriş

Yönetmek istediğim cihazı kendim yapmak istedim. Bu nedenle bunu nasıl yapabilirim diye düşündüğümde Arduino teknolojisinden yararlanmam gerektiğine karar verdim. Cihazında olması gereken özellikleri belirledim. İlgili malzemelerime karar verdim. Ardından işin en zevkli bir o kadar zahmetli olan Arduino IDE 1.8.5 platformu üzerinde C++ kodlamama başladım.

### 1.4. Kullanılan Ekipmanlar

1. Arduino Mega 2560 R3 Klon
2. Mini Breadboard
3. Jumper Kablolar(Dışı-Erkek,Erkek-Dışı)
4. SD Kart Modülü
5. Crocodile(Timsah Ağızı) Kablo
6. 2x16 LCD Ekran
7. 10k Potasiyometre
8. Piranha 7801 Hoparlör
9. Ir(Kızılötesi) Led
10. HC-05 Bluetooth Modülü
11. S-LINK SWAAP 2.1A Taşınabilir Pil Şarj Cihazı 10050MAH-(SİYAH)
12. A'dan B'ye Usb kablosu
13. LM-35 Hassas Isı-Sıcaklık Sensörü
14. Aux Kablosu(Erkek-Erkek)

### 1.5. Devre Şeması



## 1.6. Cihaz Özellikleri

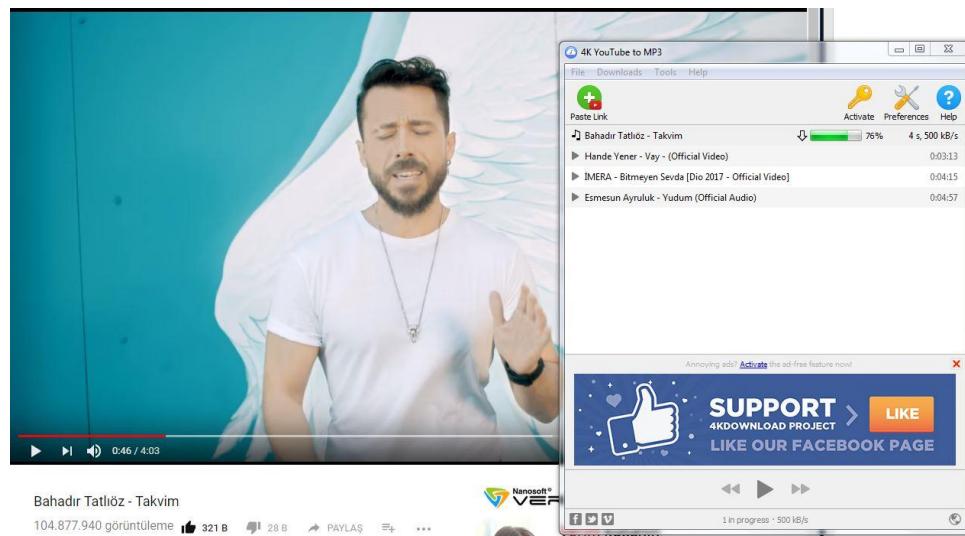
Cihazın bana hizmet edebilmesi için belirli özelliklere sahip olması gerekiyor. Peki hangi özelliklere sahip onlara bir bakalım;

### 1.6.1. Müzik Çalma Modu

Bu özelliği eklememdeki amaç 770 tane şarkı arşivimin içerisindeinden istediğim müziği şarkının adını söyleerek gidip bulup saliseler içinde bana çalmasını sağlamak. Tabi aynı zamanda sesini istediğim gibi ayarlamalıyım.”sesini aç”, ”sesini kıs” yada “sesini arttır” sesini ”azalt”

komutları ile cihazımın sesine hükmedebiliyorum.”Durdur” dedığimde durmalı “oynat” dedığimde oynatılmalı.

Cihazımda 770 tane karışık şarkısı var. Bu şarkıları tek tek indirdim. Dosya uzantılarını .mp3’ten wav'a dönüştürdüm tabi gerekli değerlerde. Aşağıda bir örnek vereyim..



Şarkıyı mp3 olarak indiriyoruz. Sonra aşağıdaki formatta ve değerler ile wav dosyası haline dönüştürüyoruz.

**Upload your audio you want to convert to WAV:**

Dosya Seç | Bahadır Tatlı... - Takvim.mp3

**Or enter URL of the file you want to convert to WAV:**

(e.g. <https://cdn.online-convert.com/example-file/audio/m4p/example.m4p>)

**Or select a file from your cloud storage for a WAV conversion:**

 Choose from Dropbox

 Choose from Google Drive

**Optional settings**

Change bit resolution:

8 Bit ▾

Change sampling rate:

16000 Hz ▾

Change audio channels:

mono ▾

Trim audio:

 to 

00:00:00

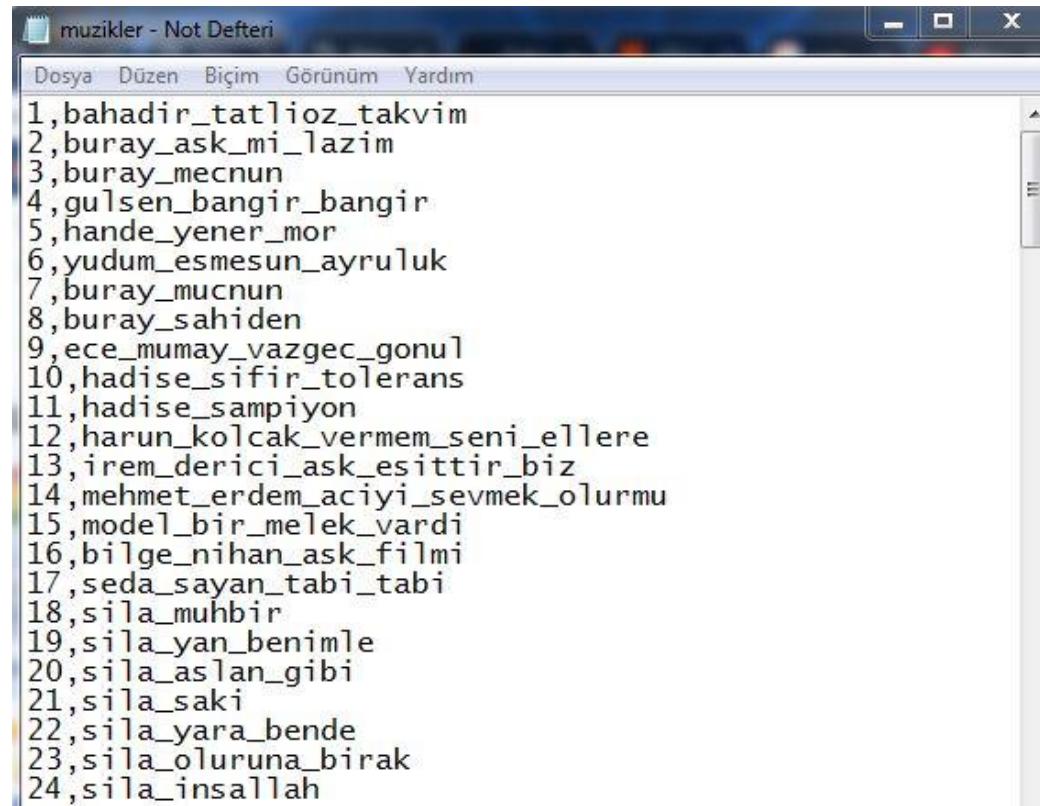
Normalize audio:

**Show advanced options >**

**Convert file**

(by clicking you confirm that you understand and agree to our [terms](#))

Bu muzikler cihazın sd kart girişinde takılı olan sd kartta kayıtlı tutuluyor. Yine aynı şekilde tüm şarkıların isimleri id bilgileri ile beraber txt dosyasına kodlanmış durumda

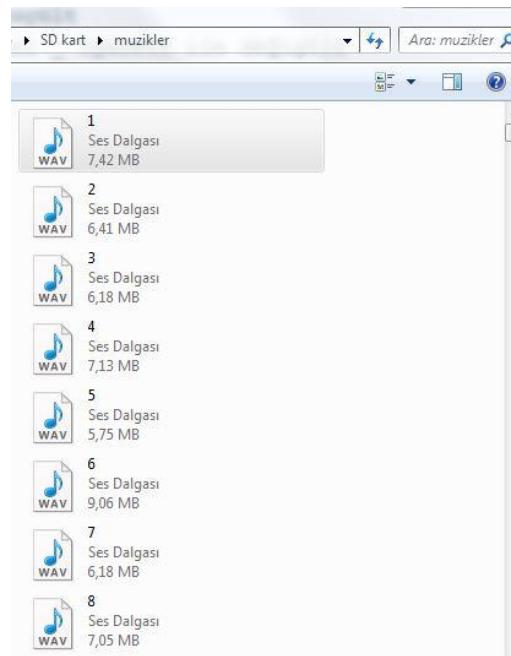


```

muzikler - Not Defteri
Dosya  Düzen  Biçim  Görünüm  Yardım
1 ,bahadir_tatlioz_takvim
2 ,buray_ask_mi_lazim
3 ,buray_mecnun
4 ,gulsen_bangir_bangir
5 ,hande_yener_mor
6 ,yudum_esmesun_ayruluk
7 ,buray_mucnun
8 ,buray_sahiden
9 ,ece_mumay_vazgec_gonul
10 ,hadise_sifir_tolerans
11 ,hadise_sampiyon
12 ,harun_kolcak_vermem_seni_ellere
13 ,irem_derici_ask_esittir_biz
14 ,mehmet_erdem_aciyi_sevmek_olurmu
15 ,model_bir_melek_vardi
16 ,bilge_nihan_ask_filmi
17 ,seda_sayan_tabi_tabi
18 ,sila_muhibir
19 ,sila_yan_benimle
20 ,sila_aslan_gibi
21 ,sila_saki
22 ,sila_yara_bende
23 ,sila_oluruna_birak
24 ,sila_insallah

```

Bu şekilde istediğim şarkının isminden kod buluyor ve bana idsini döndürüyor. Ardından bu idye ait şarkı çalışıyor. Şarkının ismi de id numarası oluyor. Oda yine txt dosyası ismiyle aynı klasör altında.

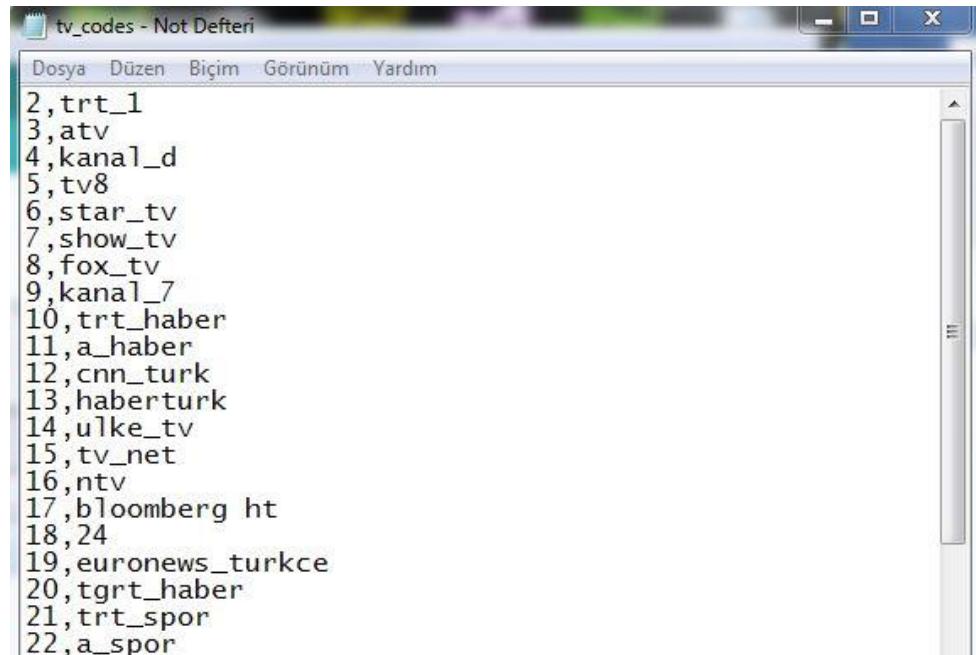


İşin mantığı bu basit gibi görünse de kod kısmı haftalarca uğraştırdı. Bir sonraki özelliğimize geçelim..

### **1.6.2. Televizyon Kumandası Modu**

Aslında benim en sevdiğim bir özellik, çünkü ben keşfettim. Televizyon uydularında kanal kanal gezmekten yoruluyoruz. Çünkü kanallar sıralı bir şekilde olmuyor. Bu nedenle aradığımız kanalı bulamıyoruz. İstediğimiz kanalı direk bulmak için kanal numarasını bilmemiz gerekiyor. Düşündüm acaba ben bir bilgisayar programcılığı okuyan bir öğrenci olarak bu işe nasıl bir çözüm bulurum onu düşündüm.

İlk olarak kanalları cihazımı tanıtmalıydım. Bunu yaparken yine txt dosyasından yararlandım. Bu sefer txt kodlamasını satır satır yaparken id numarasını sıralı bir şekilde değilde. Uydu alıcımın kanal numarasına göre yaptım. Kanal ismini cihaza gönderdiğimde onu tanıyıp kanal numarasını alabilecek ama hangi donanımla nasıl o kanal numarasını uydu alıcıma okutacağım. Bir Ir led yardımıyla bu mümkün. Aslında televizyon gibi kumandaya ihtiyacı olan cihazların kumandalarının önünde hep hiç yanmayan bir led görürüz. Aslında yanar ama biz göremeyiz. Bu ledi projemde kullandım. Bütün işi o led yapıyor. Amaç o ledi kontrol edebilmekte. Ledin çalışma prensibini araştırdım. Led ir kodları ile haberleşiyor ve ben bu ir kodlarının kumandanın üzerindeki sayılar ile eşlettirmem lazım eşlestirdiğim ir kodunu da programda bir yerde tutmalıyorum. Bunu öğrenmek için proje haricinde ayrı bir arduino cihazı ile televizyon uydularındaki alıcı göze ihtiyacım oldu. Göz hangi koda ne tepki veriyor. Kumandanın tuşlarına basarak sinyalin uydu alıcısına değilde benim arduino gözüne okuttum ve değerleri sırasıyla kaydettim. Örneğin; Kumandamın “1” tuşunun ir kod karşılığı daha doğrusu hex kod karşılığı “0x80BF49B6” ifadesine denk düşüyor. Yalnız bunu da direk öğrenemiyoruz belirli bi kaç işlemden geçiyor yeri geldiğinde bahsediceğim. Bu kodları değişkenlerime kaydettikten sonra iş bunları kullanmaya gedi. Bunlarla ilgili algoritma geliştirmeye çalıştım. Gelen kanala göre tuşa basma olayları geliştirdim. Yani bir nevi hızlı kumanda.. Bu kanal listemde yine tv\_codes.txt içerisinde kodlanmış bir şekilde



```
tv_codes - Not Defteri
Dosya Düzen Biçim Görünüm Yardım
2,trt_1
3,atv
4,kanal_d
5,tv8
6,star_tv
7,show_tv
8,fox_tv
9,kanal_7
10,trt_haber
11,a_haber
12,cnn_turk
13,haberturk
14,ulke_tv
15,tv_net
16,ntv
17,bloomberg_ht
18,24
19,euronews_turkce
20,tgrt_haber
21,trt_spor
22,a_spor
```

### 1.6.3. Hava Sıcaklığı Özelliği

Bu özellik ile ortamın sıcaklığını ölçüyoruz ve sesli olarak söylemesini sağlıyoruz. Her özellikte LCD ekrana yazı yazdırma var oraya geldiğimde anlatacağım en zor kısmı orası. Böylelikle bir ses uzağımdaki cihazımdan yada android uygulamamla ortamın sıcaklığını sesli olarak öğrenebiliyorum. Bu sesler, sıcaklık ölçme modülümden okunan sıcaklık değerleri ile eşleşiyor. "cdegers.txt" dosyamda önceden belirlenmiş olabilecek sıcaklık değerleri kayıtlı. Buradan cihazın bana sesli tepki vermesini sağlamış oluyoruz.

## 1.7. Kodlama

Arduino IDE 1.8.5 platformu üzerinde C++ dili ile cihazımı kodladım. Olabildiğince ayrıntılı bir şekilde nasıl bir yapı kurdum nasıl algoritmalar geliştirdim sırasıyla adım adım aşağıda açıklamaya çalıştım;

### 1.7.1. LCD Ekran ve SD Kart Modülü

İlk önce cihazımın düzgün çalışıp çalışmadığını anlamalıydım. Bunun için bir Lcd ekran işimi kolaylaştıracaktı ve verilerimi (bu verileri ileri doğru yazacağım) tutacak bir sd kart modülüne ihtiyacım vardı. Başlangıç için bu iki cihazımın kodlarını global olarak tanımlayıp kütüphanelerini ekledim.

#### 1.7.1.1. Global

```
//LCD
#include <LiquidCrystal.h> // lcd kütüphanesi
LiquidCrystal lcd(26,28,30,32,34,36); //rs,e,d4,d5,d6,d7
//SD Kart
#include <SD.h> //SD kart kütüphanesi
#define SD_ChipSelectPin 53 // sd kart için veri alış-verisi 53.pin
#include <SPI.h> // sd kart kütüphanesi
```

### 1.7.2. Setup ve Loop

Kullandığım Ardinuo IDE'sinin iki temel methodu vardı. Bunlardan biri "setup" diğeri "loop" .. Bu iki metodu cihazımın özelliklerine göre parça parça içindeki kod bloğunu bölerek anlatacağım ki daha anlaşılır bir bütün halinde olsun.

#### 1.7.2.1. Setup

Setup methodunda cihazın ilk açıldığında çalışacak olan kod bloğu yer alır.

```
if (!SD.begin(SD_ChipSelectPin)) { //sd kart için seçilen pine sd kart modülü bağlı mı ?
    Serial.println("SD fail"); // durum negatifse hata mesajını seri porta yazıdır
    return; //geri değer döndür boş olarak
}

lcd.begin(16, 2); // 16. ve 2. pinleri aktif et
lcd.setCursor(2,0); // 2.hane, 0.satırdan başla
lcd.print("Voice Robot"); // ilgili yazıyı yazma
lcd.setCursor(1,1); // 1.hane 1.satırdan başla
lcd.print("Hos Geldiniz :)"); // ilgili yazıyı yaz
```

#### 1.7.2.2. Loop

Bu metot ile cihaz açık kaldığı sürece sürekli çalışacak olan kısım yer alıyor. Asıl iş bundan sonra başlıyor aslında.

### **1.7.3. Ses Çalma Mekanizması**

Ardından cihaz açıldığında “Senin için ne yapabilirim ?” sorusunu kullanıcıya sormak istedim. Bunu için cihazı seslendirecek bir insana ihtiyaç vardı fakat gerçek bir insan sesi çok aksanlı ve gür olmuyordu ayrıca bu ses kadın olmalıydı ve tok bir ses olmalıydı. Televizyon programlarındaki dış sesler gibi standart bir ses olmalıydı. Sesle ile komuta-kontrole ilgili olduğum için 1.sınıf 1.dönemimde ilk C# Windows Application Projemde “Telefon Sesli Servis İşlemleri” yapmıştım. Orada sesleri birebir gerçek bir operatörün sesleri kaydedip kaydedip kesip biçimştirmek fakat bu projemde bu sesleri nereden bulacaktım. Proje konumu karar vermeden önce pek çok şeyi düşündüğüm gibi bunu da düşünmüştüm. Google’nın Google Translate servisindeki bayan sesini hepimiz biliriz. Oradaki bayanın sesini kullanmak istedim fakat sesin temposu çok yavaştı. Heceleyerek söyleyordu ve ayrıca sesi nasıl indirip kullanacaktık. İlk önce amaç sesin kaynağına gitmekti. Google Translate sayfasının kaynağına baktığında bu sesin URL biçimde tam yoluna erişemedim. Internetten araştırma yaptığımda yabancı sitelerden bu URL biçimine ulaştım. Bu URL adresindeki ilgili kısmını değiştirirsek Google Translate sesine istediğimi söyletebileceğimi fark ettim. Ses dosyasına ulaşım fakat .mp3 şeklindeydi. Bu bir sorundu cihazım için çünkü sd kart modülünden ses çalma işi .wav formatı şeklinde oluyordu bunu düzeltmeliydim. Hatta bu wav formatının dosya biçimini de farklıydı aşağıdaki görselde bununla ilgili bir örnek paylaştım.

**Upload your audio you want to convert to WAV:**

Dosya Seç | senin\_icin\_ne...abilirim.mp3

**Or enter URL of the file you want to convert to WAV:**

(e.g. <https://cdn.online-convert.com/example-file/audio/m4p/example.m4p>)

**Or select a file from your cloud storage for a WAV conversion:**

Choose from Dropbox | Choose from Google Drive

Optional settings

Change bit resolution: 8 Bit ▾

Change sampling rate: 16000 Hz ▾

Change audio channels: mono ▾

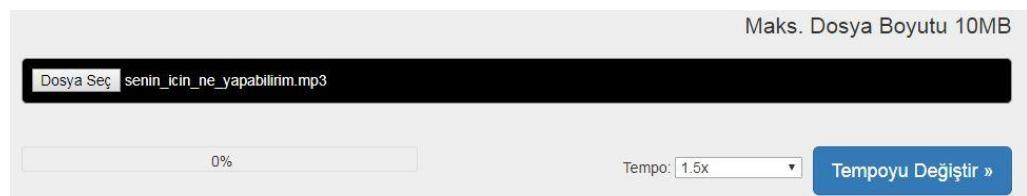
Trim audio: [start] to [end]  
00:00:00

Normalize audio:

Show advanced options >

**Convert file** (by clicking you confirm that you understand and agree to our [terms](#))

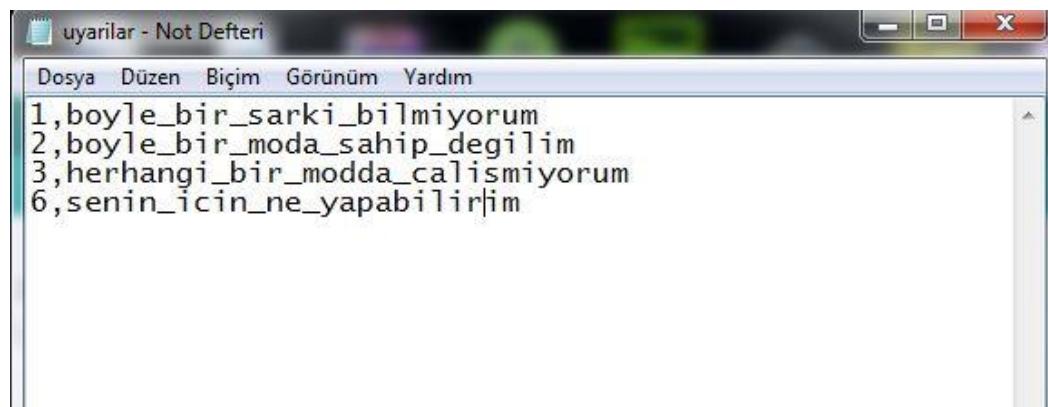
Ses formatı tamamdı fakat ses hala yavaştı. Tempoyu arttırmalıydım. İnternette Online bu işi yapan siteler vardı. Birinden tempoyu 1.5x kadar arttırmam gerektiğini deneme yanılma ile fark ettim.



Bundan sonra ise bu sesi programa entegre etmem gerekiyordu. Bunun gibi birçok sesi ayrı ayrı veritabanı tablolarında tutmalıydim. Ardinuo teknolojisi veritabanına çok sıcak bir teknoloji değilmiş çok araştırdıktan sonra anladım. O nedenle tüm tablolarımı bir txt dosyası içerisine ve satır-sütunu olmadığı için ise belir bir yapı ile yazdım ki daha sonradan ayıklamak istediğimde rahat bir şekilde veri parçamı kesip kullanayım. En temiz ve en hızlı çözümü.( Bu veritabanı işini bitirme projesi danışman hocama sordduğumda üçüncü parti uygulamaların sıkıntı yaratacağını ve kullanmamın çok doğru olmayacağına kanaat getirmiştir.)

Resimde gördüğünüz “senin\_icin\_ne\_yapabilirim.mp3” dosyası. Ses çalma mekanizmasında adı uzun olduğu için çalmadı. Bu nedenle txt dosyasında bir id atayarak ve ses dosyasının adını numara vericek şeklinde yeniden düzenlemeye karar

verdim. Aşağıdaki görselde uyarilar.txt dosyamı görüyorsunuz. Bu şekilde kategoriye göre sesleri ayrı ayrı guruplandırdım ki karışmasınlar.



Ses dosyası isimlerini söylediğleri ses göre değilde sıra numaralarına göre çekmeye karar verdim. Şimdi kod kısmına tekrar geri dönelim.

### 1.7.3.1. Setup

Cihaz ekranımıza en son karşılama yazısı yazdırılmıştı ve yine bir karşılama sesi çalmasını istedim. Bunun için bir metod yazdım.(Projemde bazı metodlar ingilizce bazıları türkçe ben o an öyle yazmak istemişimdir öyle doğru bulmuştum.) Tüm metodlar ve kodlar bana aittir.)

Metodun çalıştırılmasından itibaren, metodumu anlatmaya başlıyayım;

```
SoundPlay("uyarilar","senin_icin_ne_yapabiliirim",0);
```

Bu metod ilgili ses dosyasını bulup bana getiriyor ve çalışıyor. Bunu yaparken üç parametreye ihtiyacı oluyor. Bunlardan ilki ses dosyalarımı hangi tablodan yani txt dosyasından baksın benden bunu istiyor. İkinci olarak çalmak istediğim sesin ismini aralarında boşluk kalmayacak formatta değer veriyorum. Üçüncü olarak sesim oynatılmaya kaçınıcı saniyeden başlasın onu giriyorum. Metod bu şekilde üç parametreye ihtiyaç duyuyor. Şimdi de metodun içeriğine bakalım.

### 1.7.3.2. SoundPlay

```
String SoundPlay(String dosya_ismi, String gelen_komut, int baslangic){  
    // burada gelen verileri adım adım muzik_kodu formatına çevirip caliyoruz.  
    String muzik_kodu=dosya_ismi; // dosya_ismini değişkene atıyoruz  
    muzik_kodu.concat("/"); // dosya ismine slash işaretini ekliyoruz.  
    muzik_kodu.concat(db_dataToID(dosya_ismi,gelen_komut)); // ardından dosya yolunu buluyoruz  
  
    muzik_kodu.concat(".wav"); // dosya uzantısını ekliyoruz  
    char muzik_kodu_format[muzik_kodu.length()+1]; // muzik kodu karakter formatı oluşturuyoruz  
    /muzik kodunu char karakter dizine dönüştürüyoruz  
    muzik_kodu.toCharArray(muzik_kodu_format, sizeof(muzik_kodu_format));  
  
    // Not: Bu formata dönüştürmemizin sebebi müziğin çalışmamasıdır.  
  
    //Müziği Global olarak tanımladığımız sound_player nesnesinin play metodu ile başlatıyoruz.  
    sound_player.play(muzik_kodu_format,baslangic); //Çalınacak muzik,Müziğin başladığı saniye  
  
    /* Müzik kodu yolunu geri döndürmek istedim.  
     * Çünkü geri dönen değerin boş olup olmadığını anlamak istedim.  
     */  
    return muzik_kodu;
```

Yukarıdaki metotda açıklamam gereken bir bir metot daha var. Bu metodu daha çok dikkat çekmesi açısından tekrardan sadece o kısmı aşağıda göstermek istedim.

```
muzik_kodu.concat(db_dataToID(dosya_ismi,gelen_komut));
```

Bu metod gönderilen veriyi ilgili txt dosyasında bulup id numarasını geri döndürüyor. İki parametre alıyor. Bunlardan ilki txt dosyası adı, ikincisi ise aranacak veri. Aldığım id'yi de müzik dosyası ismiyle böylece eşleştirmiş oluyorum. Bu metodun içeriğine geçelim.

```
String db_dataToID(String dosya_ismi, String search_text){  
  
    //Serial.println("Aranacak veri: "+search_text);  
  
    String id="bos"; // geri donecek id değişkeni  
    // gerekli format  
    dosya_ismi.concat(".txt"); // dosya ismine .txt uzantısı ekliyoruz  
  
    // dosya ismimizi char karakter formatına dönüştürüyoruz.  
    char dosya_ismi_format[dosya_ismi.length()+1];  
    dosya_ismi.toCharArray(dosya_ismi_format, sizeof(dosya_ismi_format));  
  
    File dosya; // File tipinde bir değişken oluşturuyoruz  
    String satir; // satırını komple tutacağımız değişken  
    String satir_data; // satırdaki  
    String aranacak_kelime; // satırda aranacak veri
```

```

dosya = SD.open(dosya_ismi_format);
if (dosya) { // dosya tamam ise

    while (dosya.available()) { // veri olmayana kadar okumaya devam et
        satir = String(dosya.readStringUntil('\n')); // boşluğa gelince kes(satır)
        satir_data=String(db_idTodata_parcala(satir)); // satırdaki veriyi parçala
        satir_data.trim(); // satırdatanın sonundaki boşluğu kırıyoruz
        /*aranacak kelime, metoda parametre olarak geliyor bu kelimeyi
         karakterlerini küçültüp ingilizceye çeviriyoruz. Yoksa satır içinde
         arama yapamayız. Eşleşme yapılamıyor
        */
        aranacak_kelime=String(trtoeng(search_text));

        // kelimeyi search metodu yardımıyla arıyoruz ve eğer satırda var ise
        // true dönüyo
        if(search(satir_data,aranacak_kelime)){
            // ilgili veriyi bulduğumuz satırdaki idyi almalıyız
            id=db_dataToid_parcala(satir);
            break;
        }
    }

    //dosyayla işimiz bitti kapatalım..
    dosya.close();
} else {
    // dosya açılmaz ise
    Serial.println(""+dosya_ismi+".txt dosyası açılamadı !");
}

Serial.println(""+dosya_ismi+".txt dosyası açılamadı !");
}
// id değerini geri döndürüyoruz
return id;
}

```

Bu kısımda da iki hatta üç metot görüyoruz. İlk metodu yakından inceleyelim.

```
satir_data=String(db_idTodata_parcala(satir)); // satırdaki veriyi parçala
```

Txt satırındaki veri kümesininin, sadece veri yazan kısmını almam gerekiyor. Bu nedenle bu metodu kullandım. Metodun görevi, bu metoda gönderdiğimiz parametredeki değeri „,“ işaretinden parcalayarak bana veri kısmını geri döndürüyor.

```

String db_idTodata_parcala(String parcalanacak_veri){

    //parcalanacak veri ikiye ayrılacak
    String id, veri; // bunun için değişkenlerimizi oluşturuyoruz

    // verinin harf harf okicak şekilde döngü oluşturuyoruz
    for (int i = 0; i < parcalanacak_veri.length(); i++) { // değişkenin uzunluğuna gelene kadar
        if (parcalanacak_veri.substring(i, i+1) == ",") { // birer birer seç ve virgule eşitse
            id = parcalanacak_veri.substring(0, i); // virgule eşit olana kadarki kısım id
            veri= parcalanacak_veri.substring(i+1); // virgülünden sonraki kısım veri
            break; // döngümüzden çıkıyoruz.
        }
    }
}

```

Bir diğer anlatmam gereken metoda geçiyorum. “trtoeng” metodu içerisinde turkçe kelimeleri barındıran String tipindeki değişkenlerin karakterlerini değiştirek ingilizcye çeviriyor. C++ dili turkçe karakterli değişkenleri eşit kontrolü yaparken göremiyor.

```
aranacak_kelime=String(trtoeng(search_text));
```

Bu metoduda görelim..

```

String trtoeng(String gelenveri){
    //buraya parametre ile gelen değer .replace metodu ile eski,yeni
    //şeklinde arananların hepsini bulup değiştiriyor
    gelenveri.replace("ç", "c");
    gelenveri.replace("İ", "i");
    gelenveri.replace("Ü", "ü");
    gelenveri.replace("Ş", "s");
    gelenveri.replace("Ö", "o");

    gelenveri.replace("ı", "i");
    gelenveri.replace("ğ", "g");
    gelenveri.replace("ü", "u");
    gelenveri.replace("ş", "s");
    gelenveri.replace("ö", "o");
    gelenveri.replace("ç", "c");
    return gelenveri; // geri döndürüyor
}

```

Aranacak kelimeyi satır satır sürekli aramamız gerekiyordu. Satır döngümüzün içinde “search” metodunu kullanarak arama yapmak istedim. Bu metodu da kendim yazdım inceleyelim;

```

bool search(String aranilan, String aranan) {
    // aranilan değerin uzunluğundan aranan değeri çıkarıyoruz.
    int max = aranilan.length() - aranan.length();

    for (int i = 0; i <= max; i++) { // max değere gelene kadar arttır
        if (aranilan.substring(i) == aranan) return true; // aranilan aranana değere eşitse
    }

    return false; // değilse false
}

```

Bu metottan geri dönen eğer true ise aramadığımız kelime var demektir eğer false dönerse yok demektir. Eğer aradığımız kelimeyi bulursak o satırdaki id numarasına bu sefer de erişmemiz gerekiyor. Bunun içinde yeni satırdaki veri kümesini parçalamamız gerek. Aşağıdaki resimde “db\_dataToID” metodumun içerisinde bulunan dördüncü metoduma geçiyorum. Bu metotla bahsettiğim işlemi yapıyorum. Metodu tekrar yakındantáralım ve yapısını inceleyelim.

```
// ilgili veriyi bulduğumuz satırdaki idyi almalıyız
id=db_dataToid_parcala(satir);
```

Bu metoda String tipindeki “satır” değişkenini gönderiyorum bana geri değer olarak satır içindeki id numarasını geri döndürüyor. İlgili metodu inceleyelim.

```

String db_dataToid_parcala(String parcalanacak_veri){

    // verimizi iki parçağa bölücez değiştlenlerimizi tanımlıyoruz
    string id,icerik;

    // parcalanacak verinin karakter sayısı kadar döngü başlatıyoruz
    for (int i = 0; i < parcalanacak_veri.length(); i++) {
        // parcalanacak vernin karakterlerini birer birer bakiyoruz
        if (parcalanacak_veri.substring(i, i+1) == ",") { // virgüle eşite
            id = parcalanacak_veri.substring(0, i); //virgülünden önceki parçayı alıyor
            icerik= parcalanacak_veri.substring(i+1); //virgülünden sonraki parçayı alıyor
            break;
        }
    }

    return id;
}

```

id’mizi aldık ve geri döndürdük. Bu şekilde biraz karmaşık olan “db\_dataToID” metodunun görevini tamamlamış olduk. Buna bağlı olarak da “SoundPlay” motodumu da anlatmış oldum. Bu kısımda iç içe geçmiş çok yapı vardı. Biraz bu kısımda zorlanmıştım. Buraya kadar olanki kısımda tüm “setup” metodunu ayrıntılı anlatmış oldum.

### 1.7.3.3. Bluetooth ile Ses Aktarımı

Cihazımı bir şekilde sesimi göndermem ve gönderirkende bunu yazıya dökmem, daha doğrusu komuta dökmem gerekiyordu. Araştırmalarım sonucunda bir çok sesi yazıya döken program olduğunu gördüm ama maalesef çoğu ücretli hatta çok pahalıydı. Google'nın Text Recognition servisi de en ünlüler arasındaydı. Bu program ücretsizdi bunu kullanmayı tercih ettim. Yalnız bunu nerede kullacanaktım nasıl komutu gönderecektim. Bunun için de bir android uygulaması yazmıştım ve sesi yazıya dökmeyi başarmıştım. Sonra android uygulamamın cihaz ile haberleşmesi gerekiğine karar verdim. Bunun içinde bluetooth modülümü android uygulamama bağlayacak kod yapısını kurdum ve bağlantıyı sağladım. Bu kodların şimdilik Ardinuo tarafını görelim ardından Android uygulamamı detaylı incelerken bu konu ya da değineceğiz;

Global olarak cihaza gelen sesi tutacağımız değişkeni belirliyoruz.Yine cihaz moda sahip mi değil mi ona bakıyoruz.

```
// cihaza gelen sesi tutacağımız değişken
String voice = "";
// cihaz moda sahip mi
bool mod=false
```

Ardından loop metoduna yapılacak kodlar;

```
// serial port haberleşmesi ve sesin karakter karakter okunup
// voice değişkeninde toplanması
while (Serial.available()) { // seri haberleşme bitene kadar
delay(10); // 10 milisaniye bekle
char c = Serial.read(); // karakter
if (c == '#') {break;} ## işaretine eşitse karakter seri haberleşmenin sonuna gelmişiz.
voice += c; // karakteri değişken içine atıyoruz
voice.toLowerCase(); // tüm karakterleri küçült
voice.replace(" ", "_"); // boşlukların hepsini _ işaretini ile değiştir
}
```

Böylece cihazima ses komutumu göndermiş olduk. Bu ses komutu “voice” değişkeninde tutulacak. Projenin asıl amaçlarından bir tanesi bu ses komutunun içeriğinin ne olacağı, hangi biçimde geleceği(büyük-küçük-noktalama), bu komutun kesilip biçimdikten sonra hangi algoritma ile cihazın hangi özelliğine hitap edeceği , cihazın bunu anlayıp yorumlaması ve nasıl tepki göstermesi gerekiği, tüm bunların üzerine kurulmuş bir kod yapısı var aslında projemde.

“voice” değişkenini formatlı kullanılabılır hale yukarıda gördüğünüz şekilde getirdim. Cihazım modlara(özelliklere) ayrılıyor. Bu kısımda bunları anlatmak istedim. Çünkü sesi hangi amaçlara göre işliyeceğim önce genel olarak bilmelisiniz.

Artık bundan sonra cihaza mod atıyalıyız, buna bağlı seslendirmesini yine bağlı Lcd ekran kontrolünü sağlıyacağız. “loop” metodumuzun devamında cihaza tanımlı görev varmı ona bir bakalım; görev boş ise ayrıca aynı anda da cihaza bir mod ataması yapılmıyor; yapılmıyorsa, cihazın özelliğinden biri olan “sıcaklık kaç derece” sorusu cihaza soruldu mu sorulmadımı; sorulmadıysa. Aşağıdaki if bloğunu açıklamış oldum. Eğer bu iç sık biri boş ve ikisi negatifse buraya gelen komut bir mod atama komutu değildir ve cihaz bunu bilmelidir. Bu sayede cihaz bunu anlıyor ve bu sayede kullanıcıya “ herhangi bir modda çalışmıyorum !” uyarısı veriyor. Yani sen bana birşeyler anlatmak istiyorsun fakat her ne kadar anlatsan da benim daha bir modum bile yok seni uyarıyorum diyor. O kısmı görelim..

```
if(gecerli_gorev=="" && !search(voice, "modu") && !search(voice, "sıcaklık_kac_derece")) {  
    // muzik çalmıyorsa  
    if(sound_player.isPlaying()== 0) SoundPlay("uyarilar","herhangi_bir_modda_calismiyorum",0);  
}
```

Ardından mod ataması metoduna geçelim.

```
// görev ataması metodu  
ModAtama(voice);  
  
// sesi aldık kullandık işi bitti  
voice="";
```

#### 1.7.4. Mod Atama ve Kullanım Mekanizması

Aynı if ten burada da var nedense burada mantığa aykırı birşeyler oldu. Okumadığı durumlar oldu bende aynı if’ten iki kere koydum. Çifte güvenlik gibi bişey oldu. Bu metotta yukarıda bahsettiğim durumlardan bir tanesi pozitif ise mod atama kısmına geçebiliriz demektir. Çünkü gelen komut artık bir mod

atama komutudur. Fakat hangi mod onu anlamalıyız. Sırayla hangi modmuş ona bir bakalım;

```
void ModAtama(String gelen_komut){
    Serial.println("Gelen Komut: "+gelen_komut);
    if(search(gelen_komut,"modu") || search(gelen_komut,"moduna") || search(gelen_komut,"sıcaklık_kağı_derece")) {

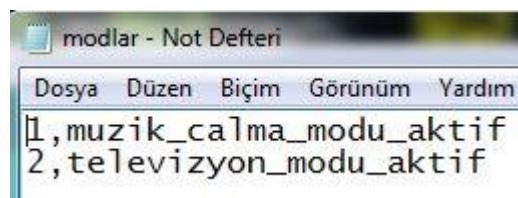
        // gelen komutta sıcaklık kaç derece ise
        if(search(gelen_komut,"sıcaklık_kağı_derece")){
            // geçerli görev ata
            gecerli_gorev="sıcaklık_öğrenme_modu";
            // seri ekranaya yazdır ki daha iyi anlaşılmasını için
            Serial.println("Sıcaklık Öğrenme Modu Aktif");
            // cihaz artık bir moda sahip
            mod=true;
        }else
        // gelen komut müzik çalma modu ise
        if(search(gelen_komut,"muzik_calma_modu")){
            // geçerli görev ata
            gecerli_gorev="muzik_calma_modu";
            // seri ekranaya yazdır ki daha iyi anlaşılmasını için
            Serial.println("Müzik Çalma Modu Aktif :)");
            // müzik çalma modu aktif sesini çal
            SoundPlay("modlar","muzik_calma_modu_aktif",0);
            // Lcd ekranın üst kısmına yazdır
            LcdUstSabit(gecerli_gorev);
            // cihaz artık bir moda sahip
            mod=true;
        }

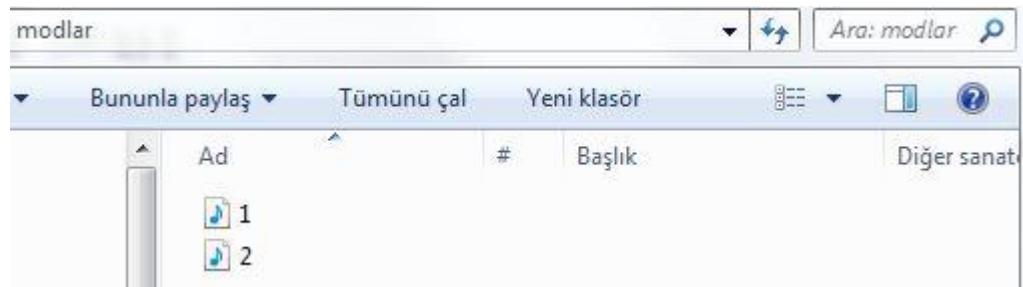
        if(search(gelen_komut,"televizyon_modu")){
            // geçerli görev ata
            gecerli_gorev="televizyon_modu";
            // seri ekranaya yazdır ki daha iyi anlaşılmasını için
            Serial.println("Televizyon Kumandası Modu Aktif :)");
            // televizyon modu aktif sesini çal
            SoundPlay("modlar","televizyon_modu_aktif",0);
            // Lcd üst kısmına Tv Kumandası Mod yazdır
            LcdUstSabit("Tv Kumandası Mod");
            // cihaz artık bir moda sahip
            mod=true;
        }
    }
}
```

Buraya kadar olan kısımda cihazımıza mod ataması yapmış olduk.

Kodların açıklamaları var ama extra bi kaç birşey daha yazmak istiyorum.

Modlar “modlar.txt” dosyasından okunuyor. Bu dosyaya bakalım..





```
//Lcd üst kısma Tv Kumandası Mod yazdır
LcdUstSabit("Tv Kumandası Mod");
```

Metodunu bir inceleyelim. Bu metot LCD ekranına yazı yazmaya yarıyor. İçeriğini inceleleyelim.

```
void LcdUstSabit(string gelen_yazi){

    //gelen_yazi komut olacağı için düzgün temiz bir yazıya
    //çevirmek için tüm alttireleri boşluk yap.
    gelen_yazi.replace("_", " ");

    lcd.clear(); // ekranda yazı varsa temizle
    lcd.setCursor(0,0); //ekranın 0. satır 0.sütünuna
    lcd.print(trtoeng(gelen_yazi)); // açıklıyacağım burayı
}
```

Son kod satırında lcd ekranına yazıyı yazıyor fakat lcd ekranlar türkçe karakter okumuyor. Bu nedenle ile “trtoeng” metodu ile ingilizceye çeviriyoruz.

Mod atamamızı yaptıktan sonra artık cihazımızın belirli bir görevi bir modu var artık. Bundan sonraki gelen komutlar bu çatı altında değerlendirilecek. Bundan sonraki gelen komutlar hangi görevde aitmiş ona göre işlemlerine başlayalım.

#### 1.7.4.1. Müzik Çalma Modu

```
// geçerli görev müzik çalma moduysa.
if(gecerli_gorev=="müzik_çalma_modu") {
    // burası çok uzun olduğu için ayrı bir metoda tutuyoruz.
    YeniMuzik(gelen_komut);
}
```

Metodumuzu incelediğimizde bizi çok uzun bir if karşılıyor. Bu kadar uzun olmasının sebebi her türlü olasılığı düşünmemden dolayı. Bu if buraya gelen komutun “şarkıyı oynat” yada “şarkıyı durdur” anlamını bir çok farklı yoldan anlamamı

sağlıyor. Buraya gelen komut ya bir şarkısı ismi olacak yada şarkıyı durdurup çalıracak bir komut. İlk if ile bunu kontrol ediyoruz.

```
void YeniMuzik(String gelen_komut){  
  
    if(gelen_komut=="şarkayı_durdur" || gelen_komut=="müziği_durdur" || gelen_komut=="dur_dur" || gelen_komut=="şarkayı_oynat" || gelen_komut=="şarkayı_cal"  
    || gelen_komut=="devam_et" || gelen_komut=="çal" || gelen_komut=="oynat"){  
  
        // yeni muzik mi değil mi  
        yeni_muzik=false;  
        // şarkıyı durduruyoruz yada başlatıyoruz hangi durumdaysa  
        sound_player.pause();
```

Muzik yeni bir muzik değil var olan muzik üzerinde değişiklik yapıyoruz ve burada müzik çalışıyorsa durduruyoruz eğer çalışmıyorsa başlatıyoruz. Bir diğer if bloğunda da müziğin istediğimiz saniyeden başlatmamızı sağlayan yani şarkıyı ileri-geri sarmamızı sağlayam bir yapı var. Bu yapı android uygulamamdaki “seekbar” dediğimiz çubuk ile kontrol ediliyor. Bu yüzden bu yapıyı anlamamanız için hem Android tarafını hem Ardinuo tarafına bakmanız gereklidir. Ben şuan sadece Ardinuo tarafını anlatacağım.

Android tarafından şifreli bir şekilde bir veri kümesi geliyor. Bu veri kümesi; “muzığın\_ismi-seekto:saniye” bu veri kümesini kesip biçimemiz gereklidir. İlk önce Serial’dan gelen komut bu şifre mi onun kontrolünü yapalım. Bunun için “:” işaretinin olup olmamasına baktık yetiyor. Hiç bir ses ile gelen komut “:” işaretini kullanamaz. Bu nedenle diğer seslerden ayrılmış oluyor. Bunun için ”.indexOf” metodunu kullanıyoruz. Bunun için “search” metodunu kullanmak mantıksız gereksiz program kasılır. Çünkü tek bir karakter araması yapacağız.

```

}else if(gelen_komut.indexOf(":") !=-1{

    //seekbar değerini(saniyeyi) bulmak için SoundSekkValue metodunu kullanıyoruz.
    int seekValue=SoundSeekValue(gelen_komut).toInt();
    // şarkının ismini SoudSeekName ayıklıyoruz.
    String soundseekname=SoundSeekName(gelen_komut);
    //var olan muzığın özelliğini değiştiriyoruz.
    // o nedenle yeni bir muzik değil
    yeni_muzik=false;
    // sonadı belirsiz boşluğu trim metodu ile
    // temizliyoruz.
    soundseekname.trim();

    // muzikler.txt den ilgili şarkı ismini ilgili saniyeden çal
    //burada mantık yeniden çalışmaya geçmesi fakat istedigim saniyeden
    //direk ilertme-geriletme fonksyonu yoktur. Böyle bir yöntemle
    // ilerletme-geriletme sağladım
    SoundPlay("muzikler",soundseekname,seekValue);
    // lcd 2.satır(1.satır programa göre)'a yazdır şarkı ismini
    LcdAltsabit(soundseekname);

}

else{yeni_muzik=true;}

```

Buraya kadar olan kısımda yeni muzik olmadığını var sayarak işlemleri yürüttük fakat yeni bir muzik ise

```

if(yeni_muzik){// yeni bir muzikse
    // gelen komuttaki çal oynat gibi kısımları kırrıyoruz
    // bunun için bir dizi tanımlıyoruz
    String komut_temizle[4]={"_çal","_oynat","_dinle","_aç"};
    // komut temizleme metodu ile ayıklanmış muziği alıyoruz
    gelen_komut=komutTemizligi(komut_temizle,gelen_komut);
    // Muzik ismimiz artık hazır. Çalmak için SoundPlay metoduma gönderiyorum.
    // müzikler.txt dosyasında muziği 0.saniyeden itibaren başlıyoruz.
    // bosmu adındaki değişkende amacımız böyle bir şarkı varmı onu öğrenmek.
    String bosmu=SoundPlay("muzikler",gelen_komut,0); // dönen değeri alıyoruz
    bosmu.trim(); // bosmu değişkeni gereksiz boşluğu siliyoruz

    // bosmu içersinde "muzikler/bos.wav" ifadesi yer alıysa boş değer dönüştür
    // Serial ekrana boş olup olmadığını döndür
    if(search(bosmu,"muzikler/bos.wav")){
        Serial.println("Boş :)");
        // boş bir muzik döndüğü içinde uyarilar.txt'den..
        // .."böyle bir şarkı bilmiyorum uyarısını döndür" çal.
        SoundPlay("uyarilar","böyle_bir_şarkı_bilmiyorum",0);
    }
    // Lcd ekranın altına gelen komutu yazdırıyoruz
    LcdAltsabit(gelen_komut);
}

```

Satır satır kodların açıklamalarını yazmışım zaten fakat metodları incelememiz gerekiyor.

```

// komut temizleme metodu ile ayıklanmış muziği alıyoruz
gelen_komut=komutTemizligi(komut_temizle,gelen_komut);

```

```

String komutTemizligi(String gelen_komut_dizisi[], String gelen_komut) {

    // gelen_komut_dizisi arınması gereken kelime listesi
    // gelen_komut ise arınacak kelime grubu

    // gelen_komut dizisinin uzunluğu kadar döngümüzü başlatıyoruz
    for (int i = 0; i < (sizeof(gelen_komut_dizisi)/sizeof(int)); i++) {
        // eger gelen komut içerisinde dizinin ilgili indexdeki
        // değeri var ise
        if(search(gelen_komut,gelen_komut_dizisi[i])){
            // bu değeri nerde gördüğsen bulup "" ile temizle
            gelen_komut.replace(gelen_komut_dizisi[i],"");
        }
    }
    // temizlenmiş komutu geri döndür
    return gelen_komut;
}

```

Bu metod ile gelen müzik ismimizdeki pürüzlerden kurtuluyoruz ve böylece tam bir müzik ismine ulaşmış oluyoruz. İçeriğine bakalım..

```

String komutTemizligi(String gelen_komut_dizisi[], String gelen_komut){

    // gelen_komut_dizisi arınması gereken kelime listesi
    // gelen_komut ise arınacak kelime grubu

    // gelen_komut dizisinin uzunluğu kadar döngümüzü başlatıyoruz
    for (int i = 0; i < (sizeof(gelen_komut_dizisi)/sizeof(int)); i++) {
        // eger gelen komut içerisinde dizinin ilgili indexdeki
        // değeri var ise
        if(search(gelen_komut,gelen_komut_dizisi[i])){
            // bu değeri nerde gördüğsen bulup "" ile temizle
            gelen_komut.replace(gelen_komut_dizisi[i],"");
        }
    }
    // temizlenmiş komutu geri döndür
    return gelen_komut;
}

// şarkının ismini SoundSeekName ile ayıklıyoruz.
String soundseekname=SoundSeekName(gelen_komut);

```

Bazı ifadelerin türkçede tam karşılığı yoktur. Bu nedenle bazı değişkenleri ve metod isimlerini ingilizce olarak yazdım. Hazır hiçbir şey kullanmadım bunu belirtmek isterim.

```

String SoundSeekValue(String parcalanacak_veri){

    //saniye değerini tutacağımız değişken
    String value;

    // metoda gelen değerin uzunluğu kadar döngümüzü oluşturuyoruz.
    for (int i = 0; i < parcalanacak_veri.length(); i++) {
        //kelimenin birer birer ":" işaretine kadar
        if (parcalanacak_veri.substring(i, i+1) == ":") { //":" gelene kadar devam et
            value= parcalanacak_veri.substring(i+1); // ":" işaretinden sonrası al
            break;
        }
    }
    // ayıklanan değeri geri döndür
    return value;
}

```

Müziğin saniye değerini bu şekilde alıyoruz. Şimdi de şarkının ismini ayıklayalım.

```

// şarkının ismini SoudSeekName ile ayıkliyoruz.
String soundseekname=SoundSeekName(gelen_komut);

```

Metodun içeriğine bakalım.

```

String SoundSeekName(String parcalanacak_veri){

    //Şarkının ismini tutacağımız değişken
    String Name;

    // metoda gelen değerin uzunluğu kadar döngümüzü oluşturuyoruz.
    for (int i = 0; i < parcalanacak_veri.length(); i++) {
        if (parcalanacak_veri.substring(i, i+1) == "-") { // "-" işaretine eşit oluncaya kadar
            Name = parcalanacak_veri.substring(0, i); // "-" işaretinden öncesini al
            break;
        }
    }
    // şarkının ismini geri döndürüyoruz.
    return Name;
}

```

Geldik bu metod içerisindeki son metoda;

```

// lcd 2.satır(1.satır programa göre)'a yazdır şarkısı ismini
LcdAltsabit(soundseekname);

```

Bu metod beni çok yordu. Çünkü bütün özellikler burda birleşiyor ve güncel kalması gerekiyor buna bağlı bir sürü olasılık meydana çıkıyor. Halbuki çok zorlanacağımı düşünmüyordum. Sürekli tekrar etmesi komut üstüne komut gelmesi biraz işler birbirine girmişi burda, sorunları çözdüm fakat çok vaktimi çaldı burası. Haftalarımı aldı diyebilirim.

```

void LcdAltsabit(String gelen_yazi){

    // gelen yazıyı tüm "_" leri boşluk ile değiştiriyoruz
    gelen_yazi.replace("_", " ");
}

```

Burdan sonra bir while döngüsü var. Bu while döngüsü aslında sonsuz bir while döngüsü sürekli güncel kalıyor. Bütün program buranın üzerinden dönüyor sürekli zaten zor olmasının sebebi bu.

Serial haberleşmeden gelen ses komutunu alıyoruz;

```

// gelen sesi tutacağımız değişken
voice="";
// Serial ekranında veri boşça çıkışa kadar
while (Serial.available()){
delay(10); // 10 milisaniye bekle
char c = Serial.read(); // Serial'dan karakter karakter okuma yapacağız
if (c == '#') {break;} // # işaretini ise karakter serial'den okuma tamamlanmıştır çakılabiliriz
voice += c; // karakterleri bir String tipindeki değişkende tutuyoruz.
voice.toLowerCase(); // tüm karakterleri küçültüyoruz.
voice.replace(" ","_"); // boşlukları "_" işaretini ile değiştiriyoruz.
}

```

Gelen voice burda birebir eşleme yapıcız yani moduna filan bakmayacağız.  
Zaten moduna bakıp buraya gelmiştık.(müzik çalma modu)

```

if(voice=="şarkıyı_durdur" || voice=="müziği_durdur" || voice=="dur_dur" ||
voice=="şarkıyı_oynat" || voice=="şarkıyı_çal" || voice=="devam_et" || voice=="çal" || voice=="oynat"){

    yeni_muzik=false; // yeni bir müzik değil
    sound_player.pause(); // sesi durdur/oynat
}

// komut içerisinde "ses_arttir" var ise
if(search(voice,"ses_arttir")){
    yeni_muzik=false; // yeni bir müzik değil aynı müzik
    sound_player.volume(1); // ses değerini bir arttır
}
// komut içerisinde "ses_azalt" var ise
if(search(voice,"ses_azalt")){// yeni bir müzik değil aynı müzik
    yeni_muzik=false; // yeni bir müzik değil aynı müzik
    sound_player.volume(0); // ses değerini bir azalt
}

```

Döngüde ses de olmayabilir. O an komut gönderilmeydiyse en sonda ne yazılıysa onun kalmasını sağlıyorum. Bir yandan da sesi durdurup kaldırıborulması için fırsat veriyorum.

```

// voice ilgili yazılarından biri eşitse
if(voice=="" || voice=="şarkıyı_durdur" || voice=="müziği_durdur" || voice=="dur_dur" || voice=="şarkıyı_oynat" ||
|| voice=="şarkıyı_çal" || voice=="devam_et" || voice=="çal" || voice=="oynat" || search(voice,"ses_arttir") ||
|| search(voice,"ses_azalt"))

```

```

| search(voice,"ses_azalt") ){

lcd.clear(); // ekranı komple temizle
LcdUstSabit(gecerli_gorev); // 1.satır(programda 0.satır)
lcd.setCursor(0,1); //1.satır(programda 0.satır) 1.hücre
// gelen yazının 17 karaktere kadar olan kısmını al ve yazdır
lcd.print(trtoeng(gelen_yazi.substring(0,17)));
delay(500); // 500 milisaniye bekle
if(gelen_yazi.substring(16,33).length()!=0){
    lcd.clear(); // lcd ekranı temizle
    LcdUstSabit(gecerli_gorev); 1.satır(programda 0.satır)
    lcd.setCursor(0,1); //1.satır(programda 0.satır) 1.hücre
    // gelen yazının 16. dan başlayarak 33 karaktere olan kısmını al
    lcd.print(trtoeng(gelen_yazi.substring(16,33)));
    delay(500); // 500 milisaniye bekle
}
// gelen yazının 33.karakterden 48.karaktere kadar kısmında yeterli karakter varmı ?
if(gelen_yazi.substring(33,48).length()!=0){ // yeterli karakter varsa
    lcd.clear(); // ekranı temizle
    LcdUstSabit(gecerli_gorev); // ekranın üst kısmına görevi yazdır
    lcd.setCursor(0,1); //1.satır(programda 0.satır)
    //gelen yazının 33.karakterden 48.karaktere kadar olan kısmını al
    lcd.print(trtoeng(gelen_yazi.substring(33,48)));
    delay(500); // 500 milisaniye bekle
}
} else {
}

```

Eğer yurdaki şartların herhangi biri sağlanmıyorsa cihaz yeni bir moda geçmiştir. Burda amaç aslında bir yerlerden geri çıkmadan nerdeyse cihaz şuanda ne yapıyorsa anında bırakıp başka bir moda geçsin diye yaptım bu kısmı. Bahsettiğim kısım aşağıda da belirtidiği gibi mod atayarak başa sarıyor. Aynı zamanda da Lcd ekran metodunun sonsuz döngüsündende kurtulmuş oluyor. Bu kısımlar beni çok yoran kısımlardı.

#### 1.7.4.2. Sıcaklık Öğrenme Modu

Bir diğer mod olan sıcaklık öğrenme moduna geçiyoruz. Sıcaklık değeri sensörden alınan değer ile direk ulaşılmıyor. Belir küçük işlemlere tabi tutuluyor. İlk önce global değişkenlerimizi oluşturalım.

```

//sıcaklık sensörü
int sensor_deger = 0; // sensörden direk okunan değer
int sicaklik_deger = 0; // hesaplandıktan sonraki net değer(°C)
float voltaj_deger = 0; // sensördeki anlık voltaj değeri

```

Sonra kaldığımız yerden devam ediyoruz kodumuza..

Öncelikle sıcaklık hesaplamasını yapmamız gereklidir;

```

if(gecerli_gorev=="sicaklik_ogrenme_modu"){
    sensor_deger = analogRead(0); // sensör değeri al 0. analog pinden
    voltaj_deger = (sensor_deger / 1023.0) * 5000 ;
    sicaklik_deger = voltaj_deger / 10.0;
}

```

Sensör değerini aldıktan sonra voltaj değerini hesaplamaya geçiyoruz. Sensör değeri 10 bitlik ADC'den oluşuyor. Bunu 1023'e böülüyoruz ardından milivolt(mV)'a çeviriyoruz. Bunun için ise 5000 ile çarpıyoruz. Bu sayede voltaj değerini hesaplamış oluyoruz. Voltaj değerini 10'a bölersek de sıcaklık değerimize ulaşmış oluyoruz.

```
// geçerli görev sıcaklık öğrenme moduysa
if(gecerli_gorev=="sicaklik_ogrenme_modu"){
    sensor_deger = analogRead(0); // sensör değeri al 0. analog pinden
    voltaj_deger = (sensor_deger / 1023.0) * 5000 ;
    sicaklik_deger = voltaj_deger / 10.0;

    //Sıcaklık değerini Serial ekranına yazdırıyoruz.
    Serial.println("Sıcaklık değeri: "+String(sicaklik_deger));

    // cihazımız artık bir moda sahip
    mod=true;
    // cdegers.txt bilgisıyla sıcaklık değerini 0.saniyeden itibaren çal
    SoundPlay("cdegers",String(sicaklik_deger),0);
    // geçerli görev ortamın sıcaklığı olacağı için bunu ata
    gecerli_gorev="Ortam Sicaklığı";
    // Lcd ekranın alt satırına sıcaklık değerini yazdır
    LcdAltsabit(String(sicaklik_deger)+" derece");

}
```

Bu şekilde ortamın sıcaklığını hızlı bir şekilde ölçmüştük.

#### 1.7.4.3. Tv Kumandası Modu

Sıradaki modumuz televizyon modu. Global değişkenlerimizi tanımlayalım ardından TvKumandası metoduna geçelim.

```

// Televizyon Modu
#include <IRremote.h>/>Televizyon Kızılıotesi(IR) Sinyali Kütüphanesi
IRsend irsend; // ir sinyali göndereceğimiz değişken

const int led = 9; // ir sinyalini gönderecek led port numarasını atıyoruz.
// const sabit bir değişken olduğunu gösterir.

// sayı kumanda hex kodları
#define sifir 0x80BFE11E
#define bir 0x80BF49B6
#define iki 0x80BFC936
#define uc 0x80BF33CC
#define dort 0x80BF718E
#define bes 0x80BFF10E
#define alti 0x80BF13EC
#define yedi 0x80BF51AE
#define sekiz 0x80BFD12E
#define dokuz 0x80BF23DC

//yonetim kumanda hex kodları
#define ses_ileri 0x80BF837C
#define ses_geri 0x80BF9966
#define ileri 0x80BF53AC

// geçerli görev televizyon modu
if(gecerli_gorev=="televizyon_modu"){

    TvKumandası(gelen_komut);

}

```

Modun mod içindeyken değiştmesini mümkün kıldığımı daha önceden yazmıştım. Burada da eğer gelen komutta “modu” ve “moduna” değilse işlemlerimize devam edelim aksi takdirde boş gelip geçsin ve başa dönsün.

```

// Tv Fonksyonları
void TvKumandası(String gelen_komut){
    bool kanal=true;
    //Kumanda Algoritması
    if(!search(gelen_komut,"modu") && !search(gelen_komut,"moduna")) {

        bool tipindeki_kanal; gelen bir kanal mı yoksa ona göre değer ataması yapacağım.
        Yani kanal ise true değilse bir özellikle bu zaten ve false değeri almıştır. Özellikle tuşlarımdan başlayayım. Bu tuşlar ses arttırma ve ses azaltma tuşlarıdır.

        // özellik tuşları
        if(gelen_komut=="ses_ileri") {kanal=false; irsend.sendNEC(ses_ileri,32);} else
        if(gelen_komut=="ses_geri") {kanal=false;irsend.sendNEC(ses_geri,32);}
    }
}

```

Eğer gelen komut ses ileri ise bu bir kanal değildir ve ses ileri hex kodumu uydu alıcısına gönderiyorum. Yine aynı şekil de gelen komut ses geri ise ses geri hex kodumu uydu alıcısına gönderiyorum.

Eğer gelen komut bunlar değilse bir kanaldır. Kanal değilse zaten kumanda basmaz.

```
if(kanal){  
    // tv_kods.txt dosyasında gelen komutun idsini bul getir  
    String tv_id=db_dataToID("tv_kods",gelen_komut);
```

Bu metodu açıklamışım. İçeriğine girmeyeceğim. Amacından bahsetmişim. Gelen bu id benim uydu alıcımındaki kanal sıra numarası. Zaten kaliteli bir marka var. O uydu alıcısını kullanan herkes aynı kanal sıra numarasına sahip olacağı için benim cihazım tüm bu marka uydu alıcılarını destekliyecektir.

Aldığımız bu kanal numarası tek basamaklı, iki basamaklı yada üç basamaklı olabilir ama bende sadece 1-9 kadar hex kodları var. Bu iki yada üç basamaklı sayılar da rakamlardan oluşuyor sonuçta. Rakamları birleştirerek iki yada üç basamaklı sayılar elde edilebilir.

Bulduğum kanal numarası bir haneli ise;

```
if(tv_id.length()==1){  
    //tv_id'nin 0.indexinden başla 1 sayı alarak  
    //metoda gönderilmesini sağlıyoruz.  
    TvTusaBas(tv_id.substring(0,1));  
  
}else
```

tv\_id'min birinci basamağını alıp TvTusaBas metoduna gönderiyorum. Bu metod benim kumandam. Buraya hangi sayıyı gönderirsem bana o sayının ir sinyalini göndermemi sağlayacak , içeriğine bakalım.

```

void TvTusaBas(String gelen_tv_id){
    // gelen sayıya göre ilgili ir kodu gönderiliyor.

    if(gelen_tv_id=="0")
        irsend.sendNEC(sifir,32);
    else
        if(gelen_tv_id=="1")
            irsend.sendNEC(bir,32);
        else
            if(gelen_tv_id=="2"){
                irsend.sendNEC(iki,32);
            }else
                if(gelen_tv_id=="3"){
                    irsend.sendNEC(uc,32);
                }else
                    if(gelen_tv_id=="4"){
                        irsend.sendNEC(dort,32);
                    }else
                        if(gelen_tv_id=="5"){
                            irsend.sendNEC(bes,32);
                        }else

                            if(gelen_tv_id=="6"){
                                irsend.sendNEC(alti,32);
                            }else
                                if(gelen_tv_id=="7"){
                                    irsend.sendNEC(yedi,32);
                                }else
                                    if(gelen_tv_id=="8"){
                                        irsend.sendNEC(sekiz,32);
                                    }else
                                        if(gelen_tv_id=="9"){
                                            irsend.sendNEC(dokuz,32);
                                        }
}

```

Bulduğum kanal numarası iki haneli ise;

```

if(tv_id.length()==2){

    //Seri ekranda iki haneli olduğunu görmek istiyorum.
    Serial.println("iki haneli: "+tv_id.substring(2,1));

    //birinci hanesini al ve gönder
    TvTusaBas(tv_id.substring(0,1));
    delay(100); // 100 milisaniye bekle
    // (deneme yanılma ile keşfettim)
    // arda arda basılırken gerekli hex kodu
    irsend.sendNEC(0xFFFFFFFF,32);
    delay(100); // 100 milisaniye bekle

    // ikincisi hanesini al ve gönder
    TvTusaBas(tv_id.substring(2,1));
    // (deneme yanılma ile buldum)
    // ileri tuşuna basmassak bu kombinasyon çalışmıyor
    irsend.sendNEC(ileri,32);
}else

```

Bulduğum kanal numarası üç haneli ise;

```

if(tv_id.length()==3){
    Serial.println("üç haneli: "+tv_id.substring(5,2));
    //birinci hanesini al ve gönder
    TvTusaBas(tv_id.substring(0,1));
    delay(100); // 100 milisaniye bekle
    // arda arda basılırken gerekli hex kodu
    irsend.sendNEC(0xFFFFFFFF,32);
    delay(100); // 100 milisaniye bekle
    // ikincisi hanesini al ve gönder
    TvTusaBas(tv_id.substring(2,1));
    delay(100); // 100 milisaniye bekle
    // üçüncü hanesini al ve gönder
    TvTusaBas(tv_id.substring(5,2));
    irsend.sendNEC(ileri,32);
}
//Lcd ekranın alt satırına
LcdAltSabit("Kanal: "+gelen_komut);
}

```

Bu kısımda dediğim gibi deneme yanılma yaparak sürekli uydu alıcım ile cihazım arasında iletişimini sağlayama çalıştım. Sonunda başardım.

## BÖLÜM III ANDROID UYGULAMASI

### 1.8. Giriş

Cihazımı gelen ses komutlarını işleyip cihazimin anlayacağı şeke dönüştürdükten sonra istedimki bu cihaza ses komutlarını göndercek bir yapı oluşturayım. Bu yapıyı Android Studio üzerinde Java kullanarak bir uygulama üzerinde kurdum. Android Studio'yu herhangi bir yerde öğrenmemiştüm, bilmiyordum. Bu neden ile internetten öncelikle eğitim seti izleyip bu kısa süre içerisinde bana yetecek kadar öğrenmeliydim.

Programın özelliklerini bu sefer kodlar ile birlikte anlatacağım. Yine java kodlarının bağlı oldukları “Activity” dediğimiz “.xml” uzantılı görsel arayüz dosyalarını da içerecek..

### 1.9. Bluetooth Bağlantı Arayüzü

Android programlamada genel mantık java dosyalarının .xml dosyaları ile birleştirilip dinamik bir arayüz tasarlamaktır. İlk önce görsel arayüzü tanıyalım ardından java dosyam ile ilişkisine bakalım...

Aşağıda gördüğünüz görselde cihazın bluetooth özelliği açık mı ona bakıyoruz. Açıksa pop-up gelmiyor. Değilse kullanıcıdan bluetooth özelliğini açması için izin isteği istiyor. Hayır der ise açana kadar liste gözükmüyor. Evet der ise bluetooth özelliği uygulamam aracılığı ile açılıyor.



Ardından karşımıza telefonun daha önce eşleştigi bluetooth cihazlarının listesi geliyor. Buradan cihazımızı buluyoruz. Tabi benim telefonum ile bluetooth modülü arasında çok deneme yaptığım için modülü farklı farklı adreslerde görüyor ama doğru olan adresi biliyorum. O neden ile ona bağlanıyorum. Yani bu farklı farklı adresler benim telefonum ile alakalı başka bir cihazda tanıdığımda uygulamam, böle bir problem olmacaktır.



Listeden bluetooth modülümü buluyorum ve seçiyorum ardından, beni programın Main sayfasına gönderiyor. Bu işlevleri bu sayfaya bağlı bir java dosyası ile yapıyorum.

Global değişkenlerimizi belirleyerek başlıyoruz..

```

13 import android.widget.ListView;
14 import android.widget.TextView;
15 import android.widget.Toast;
16
17 import java.util.ArrayList;
18 import java.util.Set;
19
20 /**
21 * Created by Enes on 13.03.2018.
22 */
23
24 public class bluetoothBaglantiActivity extends AppCompatActivity {
25
26     // bir bluetootAdapter nesnesi oluşturuyoruz.
27     private BluetoothAdapter myBluetooth = null;
28     // birleşmiş cihazları bir BluetoothDevice değişken tipinde set şeklinde tutuyoruz
29     private Set<BluetoothDevice> eslesmisCihazlar;
30     // bağlanacağımız adres
31     public static String EXTRA_ADDRESS = "device_address";
32     // bluetooth cihazlarını ekranda göstereceğimiz lisview
33     ListView cihazListesi;
34     // bluetooth cihazlarını ekranda göstereceğimiz lisview'i tetikleyecek button
35     Button cihazlarGoster;

```

Teker Teker kodların açıklamalarını yazdım. Tekrardan açıklanacak bir şey yok aslında. O yüzden devam ediyorum..

Arayüz ilk açıldığın çalışacak kodlarımızı görelim...

```

// sayfa açıldığında çalışacak kodlar..
@Override
protected void onCreate(@Nullable Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    //Bu java dosyasına bir View atama işlemi
    setContentView(R.layout.activity_bluetoothbaglanti);

    // cihaz listesini Activity'imiz ile eşleştiriyoruz.
    cihazListesi = (ListView) findViewById(R.id.tv_listView);
    //cihaz göster buttonunu Activity'imiz ile eşleştiriyoruz
    cihazlarGoster = (Button) findViewById(R.id.cihazlarigoster_button);

    //Telefonumuzun bluetoothunu adapter nesnesi ile bağlayarak varsayılan telefon bluetooth..
    //..özellikini alıyoruz
    myBluetooth = BluetoothAdapter.getDefaultAdapter();

    //Bluetoothumuzun olup olmadığını bakıza
    if (myBluetooth == null) {
        //Bluetooth yoksa uyarı mesajı ver ve applicationi kapat
        Toast.makeText(getApplicationContext(), text: "Telefonunuz Bluetooth Desteklemiyor", Toast.LENGTH_LONG).show();
        finish();
    } else if (!myBluetooth.isEnabled()) {
        //Bluetooth var ama açık değilse açmak için istekte bulun
        Intent BTac = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
        // isteği(Acitivity) başlat
        startActivityForResult(BTac, requestCode: 1);
    }

    // cihazlarigoster buttonuna tıkladığımızda
    cihazlarGoster.setOnClickListener((v) → {
        //esleşmiş cihazları bulup listelen metod
        eslesmisCihazlarGoster();
    });
}

```

```

private void eslesmisCihazlariGoster() {
    //eslesmis cihazları al
    eslesmisCihazlar = myBluetooth.getBondedDevices();
    //gecici cihazlistesi dizisi adapter'de kullanmak için geçici olarak olusturdum
    ArrayList list = new ArrayList();

    //eslesmiş cihazlar listesi boş deilse
    //yani eslesmiş cihazlar listesi uzunluğu 0'dan büyükse
    if (eslesmisCihazlar.size() > 0) {
        for (BluetoothDevice bt : eslesmisCihazlar) {
            //Cihazın adını ve adresini listeye ekle
            list.add(bt.getName() + "\n" + bt.getAddress());
        }
    } else {
        Toast.makeText(getApplicationContext(), text: "Eslesmis Cihaz Bulunamadi", Toast.LENGTH_LONG).show();
    }

    // bir dizi adapter nesnesi olustur,bu sayfada,list layoutu ile,ilgili dizide
    final ArrayAdapter adapter = new ArrayAdapter(context: this, android.R.layout.simple_list_item_activated_1, list);
    //adapteri liste ile eslestir.
    cihazListesi.setAdapter(adapter);
    //Listeden bir cihaz secdigimize cagrılacak method tetikleticisi ekliyoruz
    cihazListesi.setOnItemClickListener(cihazSec);
}

//ListView'a tıklayıp seçmemizi saglayan method
private AdapterView.OnItemClickListener cihazSec = (parent, view, position, id) -> {
    //Mac Addresini yani, viewdaki son 17 karakteri alıyoruz
    String info = ((TextView) view).getText().toString();
    String address = info.substring(info.length() - 17);

    //Yeni bir Activity baslatman için bir intent tanimliyoruz
    Intent intent = new Intent( packageContext: bluetoothBaglantiActivity.this, MainActivity.class);
    // Activity ile beraber EXTRA ADDRESS çatısı al
    intent.putExtra(EXTRA_ADDRESS, address);
    //Activity'i calistir
    startActivity(intent);
};

```

Bluetooth haberleşmesini tüm arayüzlerimizde kullanacağız. Bu neden ile bir de “bluetooth ekle” adında bir java dosyası açıyoruz fakat bu java dosyasının bir arayüzü olmayacağı çünkü bu dosya her arayüzde kullanılacak. Görelim..

## 1.10. Bluetooth Bağlantı Eklemesi

```

/**
 * Created by Enes on 14.03.2018.
 */

public class BluetoothEkle extends View {
    private static Context context;

    //Bluetooth Değişkenleri
    //Bluetootj adresi
    static String address = null;
    //Bulutut açılsın mı isteği ?
    private static ProgressDialog progress;
    // cihaz ile uygulma arası adapter
    private static BluetoothAdapter myBluetooth = null;
    // bluetooth soketi boş atıyoruz
    static BluetoothSocket btSocket = null;
    // bluetooth bağlandı mı ?
    private static boolean isBtConnected = false;
    // bluetooth kimlik adresi
    static final UUID myUUID = UUID.fromString("00001101-0000-1000-8000-00805F9B34FB");

    //Constructor metodumuz(yapıcı metod)
    public BluetoothEkle(Context context) {
        super(context);

        this.context=context;
    }

    // bluetooth bağlanacak metod
    public static class BTbaglan extends AsyncTask<Void(Void,Void> {
        //bağlantı başarılı
        private boolean ConnectSuccess = true;
        // bu arayüze geri döndüğünde çalışacak kod..
        @Override
        protected void onPreExecute(){
            //Açılır menü (Resimde var bakabilirsınız.)
            progress = ProgressDialog.show(context, title: "Baglanıyor...", message: "Lütfen Bekleyin");
        }

        // bu olay baslarken arkaplanda çalışacak seyler
        @Override
        protected Void doInBackground(Void...devices){
            // bağlantı durumunda bir sıkıntı var ise program hata vermemesi açısından
            try {

                if(btSocket == null || !isBtConnected){
                    myBluetooth = BluetoothAdapter.getDefaultAdapter();
                    // gelen address bilgisi ile telefonun bluetooth cihaz özelliği tanımlanıyor.
                    BluetoothDevice cihaz = myBluetooth.getRemoteDevice(address);
                    // benzersiz id ile socket kaydediliyor.
                    btSocket = cihaz.createInsecureRfcommSocketToServiceRecord(myUUID);
                    // bluetooth adaptörü geri döndüğünde(sayfada ileri geri) kendini yenilesin
                    BluetoothAdapter.getDefaultAdapter().cancelDiscovery();
                    // btsockete bağlan
                    btSocket.connect();
                }
                } catch (IOException e){
                    ConnectSuccess = false;
                }
            return null;
        }
    }
}

```

```

        }
        // bağlantı gönderildiğinde
        @Override
        protected void onPostExecute(Void result){
            super.onPostExecute(result);
            if(!ConnectSuccess){
                msg( s: "Baglanti Hatası, Lütfen Tekrar Deneyin");
                // arayüzü bitir
                ((Activity) context).finish();
            } else {
                msg( s: "Baglanti Basarılı");
                //bağlantı başarılı
                isBtConnected = true;
            }
            progress.dismiss();
        }
    }
    //Baglantıyı Sonlandırma
    void Disconnect(){
        if(btSocket!=null){
            // bağlantı kapatılamaz ise
            try {
                btSocket.close();
            } catch (IOException e){
                msg( s: "hata");
            }
        }
        // arayüzü bitir
        ((Activity) context).finish();
    }
    //bağlantı arayüzü,mesaj,gösterilme uzunluğu
    // uyarı mesajı
    private static void msg(String s) { Toast.makeText(context,s, Toast.LENGTH_LONG).show(); }
}

```

Buraya kadar herşey tamam fakat Android platformu üzerinde bu tür işler izin ister. Nedeni ise sonuçta kullanıcının cihazının bir özelliğini kullanıyoruz. Bu yüzden ondan izin istemeliyiz. Zaten Android platformu da buna izin vermiyor. O nedenle izinlerimizi alalım. Bu izini “AndroidManifest.xml” dosyasından alıyoruz. Buraya sık sık uğramak zorunda kaldım prosedür gereği.

```

<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH" />

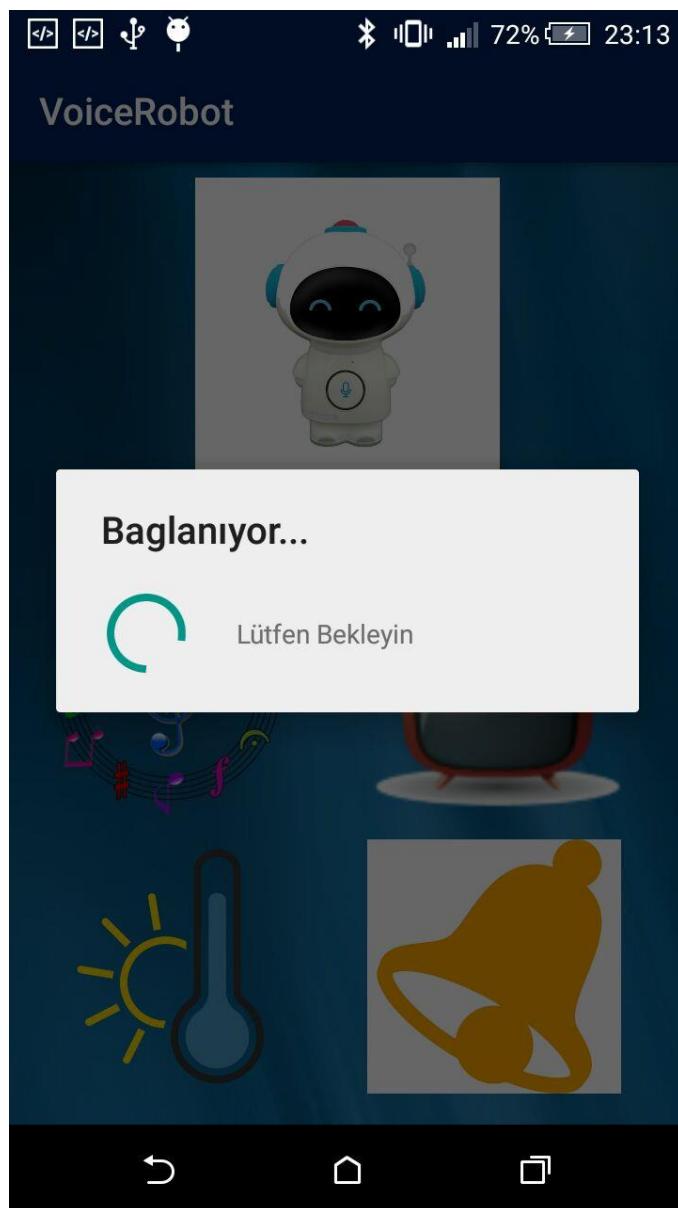
```

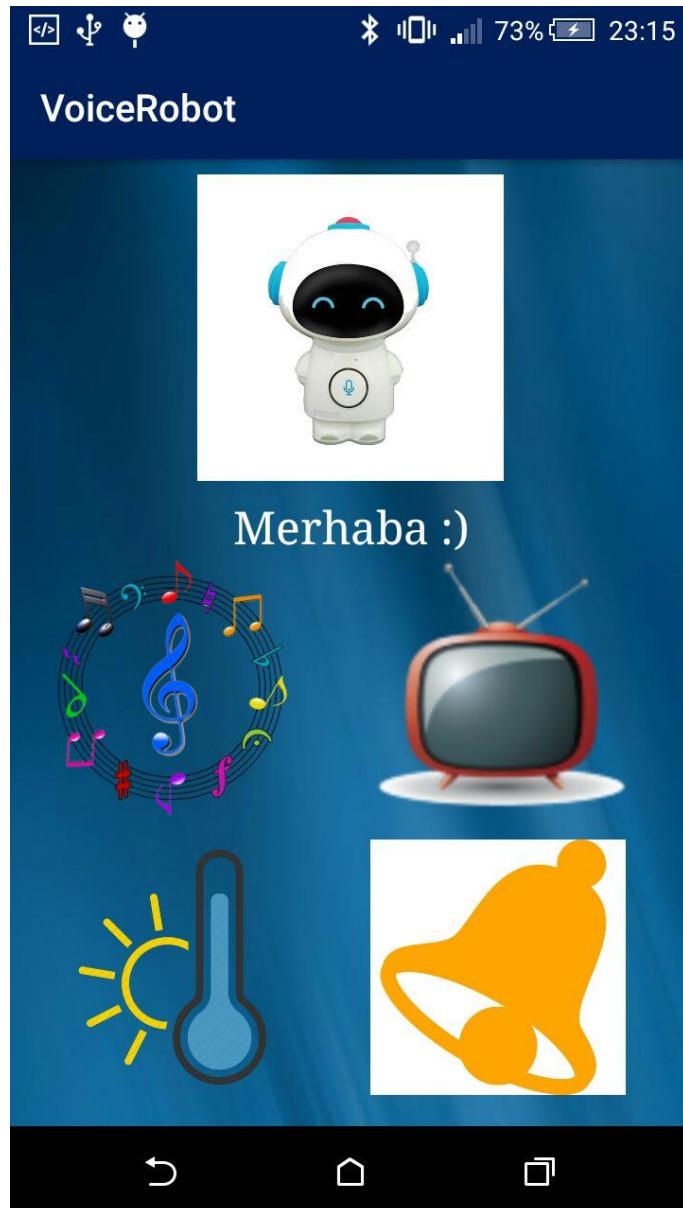
## 1.11. MainActivity(Ana Arayüz)

Bu arayüz adı üstünde ana arayüz, uygulamamın başladığı arayüz. Arayüze geçmeden bu arayüzü sadece izin almak için kullanmadığımız program arayızlarını teker teker tanıttığımız “AndroidManifest.xml” dosyamıza geri dönüyoruz. Burada genel program bilgilerinin de yer aldığı her Activity(Arayüzü)'mizi tanıtıyoruz.

```
<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="VoiceRobot"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/AppTheme">
    <activity android:name=".MainActivity">
        <intent-filter>
            <category android:name="android.intent.category.DEFAULT" />
        </intent-filter>
    </activity>
```

Ardından Arayüzümüzün çalışır halini görelim ki ne yaptığımı daha iyi kavrayalım..





Yazmak istediğim bir durum var. En sonda yine bahsedeceğim ama kodları yazarken değişken, isimleri metod isimleri verirken bazen ingilizce bazen türkçe yazmışım. Genelde ben böyle yapıyorum. O an aklıma bazen ingilizcesi geliyor türkçesi gelmiyor aklıma; bazende ingilizcesi değil direkt türkçesi aklıma geliyor, o nedenle bir yerden alıntı yapılmış gibi bir durum olmasın, ben tüm projemde hazır birşeyler kullanmak yerine araştıruıp yapıları tamamen kendim kurmak istedim.

Global değişkenlerimizden başlayalım..

```

public class MainActivity extends AppCompatActivity {

    // istek tanımlıyoruz
    Intent intent;

    // Ses Değişkenleri
    TextView karsilama_ses;
    // İhtiyaç duyulan ses istediği giriş için istenen karakter uzunluğu
    private final int REQUEST_CODE_SPEECH_INPUT = 100;

    // ImageButton'larımıza tanımlıyoruz
    ImageButton music_imagebutton;
    ImageButton tv_imagebutton;
    ImageButton voicerobot_imageButton;
    ImageButton temparature_imageButton;
    ImageButton reminders_imageButton;

    // arayüzüümüze bluetooth özelliği kazandıracak değişken
    BluetoothEkle bluetoothEkle;

    // Arayüz açılırken

```

Bluetooth özelliğimizi aktif edelim..

```

//Bluetooth Cihazının adresini alır
Intent newint = getIntent();
//bluetooth bağlantı Activity 'inden Ariduno cihazıma
BluetoothEkle.address = newint.getStringExtra(btBaglantiActivity.EXTRA_ADDRESS);

//btBaglantiActivity'sinden ödüng aldığımız BTbaglan() metodunu çalıştırıyoruz
new BluetoothEkle.BTbaglan().execute();

```

Aşağıda gördüğünüz resme tıklandığında açılan pencere; Google'nin sesi yazıya dökmeye yarayan çok kullanışlı servisidir. Benim genel bitirme projemdeki asıl amacım dokunmayla değilde sadece sesi yazıya döküp o sesi işlemekti fakat ayrıca bitirme projesi danışman hocamın isteği ve doğrultusıyla ve tabi benimde isteğimle android uygulamasını yazmaya karar verdik. Asıl amacımdan sapmaması ve daha kolay bir şekilde işlemleri yürütmek için bu buttonu her arayüzüme ekledim.



```
voicerobotImageButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        //Google Servisini kullanıyoruz..
        Intent intent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
        intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL, RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
        // Cihazın varsayılan dilini kullanıyoruz
        intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE, Locale.getDefault());
        // Resimde göreceksiniz. Konuma buttonun altındaki yazı
        intent.putExtra(RecognizerIntent.EXTRA_PROMPT, value: "Konuşabilirsınız..");

        try {
            // bu activity için intent sorğusu oluşturuyoruz
            MainActivity.this.startActivityForResult(intent, REQUEST_CODE_SPEECH_INPUT);
        } catch (ActivityNotFoundException e) {
            Toast.makeText(getApplicationContext(), text: "Desteklenmiyor :(", LENGTH_SHORT);
        }
    }
});
```

Burada sadece isteğimizi başlatıyoruz. Ardından dönen sesi alıp ilgili arayüze yönlendiriyoruz.

```

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    MainActivity.super.onActivityResult(requestCode, resultCode, data);

    switch (requestCode) {
        case REQUEST_CODE_SPEECH_INPUT: {
            if(resultCode == RESULT_OK && data!=null) {
                ArrayList<String> donus = data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
                // intentden dönen veriyi arrayliste atıyoruz.
                karsilama_ses.setText(donus.get(0));

                //gelen sesi cihaza gönder
                // socket
                if ( bluetoothEkle.btSocket != null) {
                    try {

                        bluetoothEkle.btSocket.getOutputStream().write(karsilama_ses.getText().toString().getBytes());
                    } catch (IOException e) {
                        Toast.makeText(getApplicationContext(), text: "Hata !", Toast.LENGTH_SHORT).show();
                    }
                }

                // gelen sese göre ekranlara yönlendir..
                switch (donus.get(0)){
                    case "muzik calma modu":
                        SanatcilarEkranı();
                        break;
                    case "televizyon kumandası modu":
                        TvKumandasıEkranı();
                        break;
                }
            }
            break;
        }
    }
}

```

Bu buttona tıklandığında evente bağlı burasında beraberinde çalışmış oluyor. Burada dönen sesi duruma göre yönlendiriyoruz.

Yönlendirmek istediğimiz arayüzler(ekranlar) aşağıda metodlar halinde yazılıdır.

```

private void SanatcilarEkranı(){
    intent = new Intent( packageContext: MainActivity.this,sanatcilaragoresarkilarActivity.class);
    startActivity(intent);
}

private void TvKumandasıEkranı(){
    intent = new Intent( packageContext: MainActivity.this, tvActivity.class);
    startActivity(intent);
}

private void HatirlaticiEkranı(){
    intent = new Intent( packageContext: MainActivity.this,hatirlaticiActivity.class) ;
    startActivity(intent);
}

```



Düger arayzlere ulaşma işlemini ses ile komut vererek de yapabiliriz.

Eğer telefonun geri tuşuna basarsak da bluetooth bağlantısının kesilmesini istedim çünkü bu arayüzden geri tuşuna basılması uygulamadan çıkışını demek.

```
@Override  
public void onBackPressed() { // telefon geri butonuna basılırsa bağlantı düşsün  
    super.onBackPressed();  
    bluetoothEkle.Disconnect();  
}
```

Burdan sonra artık cihazın özelliklerine göre tasarlanmış arayüzlerime geçiyoruz..



Buttonuna dokunulduğunda ;

## 1.12. Sanatçılara Göre Şarkılar Arayüzü

Şarkılarımın karışık bir şekilde düz liste yapmak istemedim. Madem 770 tane şarkıyı teker teker indirmişim öyleyse bunları güzel , göze hitap edicek bir şekilde sıralamalıydım. Arayüzü görelim..



Her zaman aynı boyutta resim bulmak zor fakat gerekli ayarlar ile resimler bozulmadan listelenebiliyor. İlk liste elemanında tüm şarkıların listelendiği listeyi görebiliyoruz. Kalan elemanlar ise sanatçıların isimleri yer alıyor.

Projemi yaparken satır satır kod açıklaması yapmamışım fakat sırı tezim için extradan tekrardan araştırip yapıyı anlayıp yazdım. Görelim..

Global değişkenlerim..

```
public class sanatcilaragoresarkilarActivity extends AppCompatActivity {

    //Sıralı liste
    GridView gridView;
    //Voice robot karşılaşması
    TextView karsilama_textView;
    //voicerobot image buttonu
    ImageButton voicerobot_imageButton;

    //gerekli dizi listelerini oluşturuyoruz
    ArrayList<String> sanatci_isimleri = new ArrayList<>();
    ArrayList<String> tek1_sanatci_isimleri = new ArrayList<>();
    ArrayList<Integer> tek1_sanatci_idleri = new ArrayList<>();

    ArrayList<String> tum_sarki_listesi = new ArrayList<>();
    ArrayList<Integer> tum_sarki_idleri = new ArrayList<>();

    // bluetooth özelliğini arayüzüümüze ekliyoruz.
    BluetoothEkle bluetoothEkle;

    //İhtiyaç duyulan ses istediği giriş için istenen karakter uzunluğu
    private final int REQUEST_CODE_SPEECH_INPUT = 100;

    //ses calma tipindeki ses calma değişkeni
    private AudioManager audioManager = null;
```

Arayüz açılırken çalışacaklar..

```

💡 @Override
protected void onCreate(@Nullable Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_sanatcilaragoresarkilar);

    //arayüzdeki elemanlar ile eşleştiriyoruz
    gridView = (GridView) findViewById(R.id.sanatcilar_gridview);
    voicerobot_imageButton=(ImageButton) findViewById(R.id.muzik_voicerobot_imageButton);
    karsilama_textView=(TextView) findViewById(R.id.muzik_karsilama_textView) ;

    // bu arayüze ekle
    bluetoothEkle = new BluetoothEkle( context: sanatcilaragoresarkilarActivity.this);

    //Müzik Txt Dosyasındaki verileri okuyalım..
    String data = "";
    StringBuffer stringBuffer = new StringBuffer();
    //muzik.txt dosyasını uygulama kaynağından al
    InputStream is = this.getResources().openRawResource(R.raw.muzikler);

    // yüklemiş veri okuyucu
    BufferedReader reader = new BufferedReader(new InputStreamReader(is));
    // dizi indexi
    int index=0;

    // telefonun ses servisini kullanıyoruz
    audioManager=(AudioManager) getSystemService(getApplicationContext() .AUDIO_SERVICE);

    // telefonu titreşim moduna alalım ki cihazdan sadece ses gelsin
    audioManager.setStreamVolume(AudioManager.STREAM_MUSIC, index: 0, flags: 0);

    // aynı sanatçıdan sadece bir kere bulunduğu resim idsinin..
    // ..ve sanatçı isminin bulunduğu dizilere eklemeye yapıyoruz

```

Veritabanını okuyoruz ve verilerimizi düzenli bir şekilde dizilerimize atıyoruz.

Veritabanından okurken izine ihtiyacımız var.”AndroidManifest.xml” dosyasında

```

<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />

```

```

// aynı sanatçıdan sadece bir kere bulunduğu resim idsini..
// ..ve sanatçı isminin bulunduğu dizilere eklemeye yapıyoruz

tekl_sanatci_idleri.add(index: 0, element: 0);
tekl_sanatci_isimleri.add(index: 0, element: "Tüm Liste");

sanatci_isimleri.add(index: 0, element: "bos");
// değer boş deisse
if(is !=null) {
    try {
        // data boş olana kadar
        while ((data = reader.readLine()) != null) {
            // satır sonu kes
            stringBuffer.append(data + "\n" + "\t");
            // kesilmiş satır yeni itemdir(daha sonradan düzenlenecek)
            String newItem = stringBuffer.toString();
            //kesilmiş satır yeni itemdir.
            String temizNewItem = stringBuffer.toString();
            /*
                770 tane veri olduğu için ve sıralı şekilde sayıldığını düşünürsek
                bir, iki ve üç haneli sayılardan oluşuyor.

            */
            // eğer satırın 1. karakterinde , varsa o bir hanelidir.
            if (newItem.charAt(1) == ',') {
                // o hanenin , den sonradan başlayarak veri sonuna kadar
                newItem = newItem.substring(2, newItem.length());
                // eğer satırın 2. karakterinde , varsa o iki hanelidir.
            } else if (newItem.charAt(2) == ',') {
                newItem = newItem.substring(3, newItem.length());
                // eğer satırının 3.karakterinde , varsa o üç hanelidir
            } else if (newItem.charAt(3) == ',') {
                // o hanenin , den sonradan başlayarak veri sonuna kadar
                newItem = newItem.substring(4, newItem.length());
            }
            // dizinin ilgili indexine
            index++;
            //temiz sanatçı ismini al
            sanatci_isimleri.add(index,ilkHarfBuyuk(temizNewItem));
            // sanatçı isimlerini - işaretinden keserek al
            tekl_sanatci_isimleri.add(index, ilkHarfBuyuk(newItem).split(regex: " - ")[0]);
        }

        // satırla işin bitti sil
        stringBuffer.delete(0, stringBuffer.length());
    }
}

} catch (IOException e) {
    // hata olur ise hata mesajını göster
    e.printStackTrace();
}
}

```

Burada benim doldurup boşaltıp tekrar doldurma dediğim bir yöntemle dizideki aynı elemanları siliyorum. Görelim..

```

/*
    LinkedHashSet tipinde bir liste oluşturuyoruz.
*/
LinkedHashSet<String> lhs = new LinkedHashSet<~>();
// tüm tek_l_sanatci_isimlerini lhs değişkenine atıyoruz

lhs.addAll(tek_l_sanatci_isimleri);

// tek_l_sanatci_isimleri temizle
tek_l_sanatci_isimleri.clear();

//tek_l_sanatci_isimlerine tekrardan lhs ile doldur.
tek_l_sanatci_isimleri.addAll(lhs);

```

Bu şekilde birden fazla şarkısı olan sanatçıların(hepsinin birden fazla şarkısı var) sadece sanatçı ismini almak amacı ile sanatçıları teke indirilmeisini sağlamış oluyorum.

Şimdi de sırada bu sanatçıların resimlerini teke düşen tek\_l\_sanatçılar ile eşleştirmemi yapmamız gereki sanatçılar resimleri ile beraber görünsün.

```

// sadece sanatçıların olduğu resimleri süzüp alalım;
//tek_l_sanatci_isimleri listeliyoruz
int img_id;
for (int i = 0; i < tek_l_sanatci_isimleri.size(); i++) {

    //sanatçı isimlerini listeliyoruz
    for (int k = 0; k < sanatci_isimleri.size(); k++) {
        // sanatçı isimlerini bütün karakterlerini küçültüp içerisinde daha önceden..
        //...tek_sanatci_isimleri eklemişse eklenmemesini sağlan yapı
        if(sanatci_isimleri.get(k).toLowerCase().indexOf(tek_l_sanatci_isimleri.get(i).toLowerCase()) != -1){
            // eğer yeni bir sanatçı ise
            // ilgili sayı indexindeki sanatçayı al gereksiz boşluk kırp id için , kes ve numarayı al
            img_id = Integer.parseInt(sanatci_isimleri.get(k).toString().trim().split( regex: ",") [0]);
            // yeni sanatçayı diziye atıyoruz
            tek_l_sanatci_idleri.add(img_id);

            break; // aynı sanatçidan bir tane alıp döngüden çıksın. Diğerleri aynı olduğu için gerek yok
        }
    }
}

```

Dizimiz artık hazır, bu arayüzdeki gridview'e bağlayıp kullanmamız gereki. Bunun için önceden oluşturduğum Adapter nesneme bakıyorum..

Global değişkenler;

```

package com.example.enes.voicerobot;

import ...

/*
 * Created by Enes on 14.04.2018.
 */

public class muzik_GridAdapter extends BaseAdapter {
    // Gridview'de listelenecek diziler
    private ArrayList<String> sanatci_isimleri;
    private ArrayList<Integer> sanatcilar_imgid;

    // Arayüz bağlantısı
    private Context context;
    // Gridview layoutu(hücre görüntüsü)
    private LayoutInflater layoutInflater;

    public muzik_GridAdapter(Context context,ArrayList<Integer> sanatcilar_imgid,ArrayList<String> sanatci_isimleri){
        // Adaptör metodu için yapıcı metodumuzu oluşturuyoruz
        this.context=context;
        this.sanatcilar_imgid=sanatcilar_imgid;
        this.sanatci_isimleri=sanatci_isimleri;
    }
}


```

Gridview'in özelliklerinin bulunduğu hazır metodlara geçelim. Metodların ismi hazır metot copy-paste yapmadım.

```

// GridView kaç kutulu olsun
@Override
public int getCount() { return sanatci_isimleri.size(); }

//GridView objesi
@Override
public Object getItem(int position) { return sanatci_isimleri.get(position); }

//GridView'den item pozisyonu
@Override
public long getItemId(int position) { return position; }

```

Aşağıdaki metoda ise Gridview'in içeriğini oluşturuyoruz. Tam olarak Gridview hücresi burada yok aslında onu ayrı bir view ile göstereceğiz.

```

//Gridview'in View getirki fakat benim istediğime göre
@Override
public View getView(int position, View convertView, ViewGroup viewGroup) {

    View gridView = convertView;
    if(convertView== null){ // Gridview varmı ?
        //Gridview hücresi oluştur.
        layoutInflater = (LayoutInflater) context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
        //gridviewin Viewi aslında body'si yani vicudunu oluşturuyoruz
        // Bu view benim tasarlardığım ayrı bir dosyada bulunan ayrı bir view ile oluyor.
        gridView = layoutInflater.inflate(R.layout.sanatci_layout, root: null);

    }
    // Grdid Hücresinde olacaklar
    ImageView sanatci_imageview = gridView.findViewById(R.id.sanatci_imageview);
    TextView sanatci_textview = gridView.findViewById(R.id.sanatci_textview);

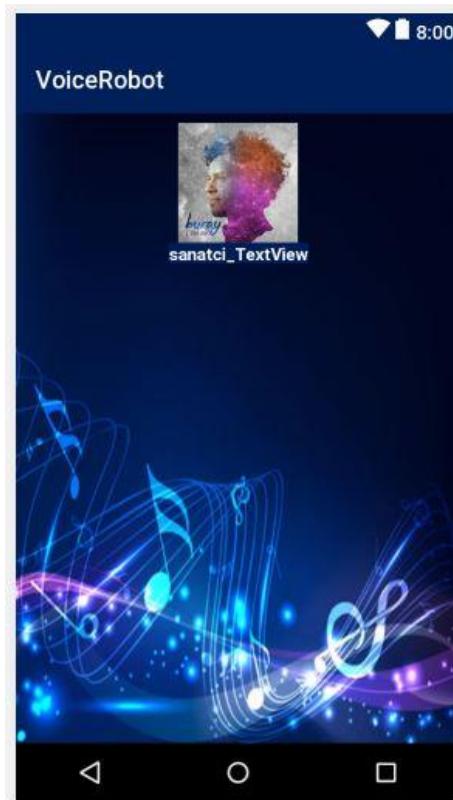
    //telefonun(cihazın) sd kartına erişiyor ve gerekli resimleri alıyor
    String image_path="/storage/ext_sd/VoiceRobot/K_AlbumCover/x"+sanatcilar_imgid.get(position)+".jpg";
    Uri kapak_uri = Uri.parse(image_path); // resim yolunu uri olarak değiştir
    sanatci_imageview.setImageURI(kapak_uri);// sanatci resmine yol ata

    // sanatçı ismini göster
    sanatci_textview.setText(sanatci_isimleri.get(position));

    // hazırlamış olduğum Gridview'i gönder
    return gridView;
}

```

Gridview hücresinin ayrı bir .xml dosyası mevcut inceleyelim.. Tabi bu dosyanın bir java dosyası olmadığı için doğal olarak extra çalışır halini gösteremeyeceğiz fakat, Android Studio üzerinde göstermeye çalışayım..



Bu şekilde döngü ile kutu şeklinde Gridview hücleri oluşmuş oluyor.

Bu şekilde “Sanatçılara Göre Şarkılar” ismini verdığım viewim aslında olmuş oldu genel hatlarıyla fakat ben istedimki ses ile ne çalmak istiyosam o an canım ne istiyorsa söyleyeyim cihazım anlayıp yorumlayıp bana çalsın istedim. Bunu yapmak için;



Buttonuna dokunuyoruz..

```
voicerobot_imageButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        //Google Servisini kullanıyoruz..
        Intent intent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
        intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL, RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
        // Cihazın varsayılan dilini kullanıyoruz
        intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE, Locale.getDefault());
        // Resimde göreceksiniz. Konuma buttonun altındaki yazı
        intent.putExtra(RecognizerIntent.EXTRA_PROMPT, value: "İstediğiniz Şarkı");

        try {
            // bu activity için intent sorusunu oluşturuyoruz
            sanatcilaragoresarkilarActivity.this.startActivityForResult(intent, REQUEST_CODE_SPEECH_INPUT);

        } catch (ActivityNotFoundException e) {
            Toast.makeText(getApplicationContext(), text: "Desteklenmiyor :(", Toast.LENGTH_SHORT);
        }

    }
});
```

Sesi alıp SecilenMuzikEkrani'na gidiyoruz.

```
protected void onActivityResult(int requestCode, int resultCode, Intent data){

    sanatcilaragoresarkilarActivity.super.onActivityResult(requestCode, resultCode, data);

    // istek kodu
    switch (requestCode)
    {
        case REQUEST_CODE_SPEECH_INPUT: { // istek konuşma girişi
            if(resultCode == RESULT_OK && data!=null) { // istek tanimli ve data bos gelmediyse
                //Ses al değişken dizisine ata
                ArrayList<String> donus = data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);

                // karsilamaya etmək zorundayım fünkü direkt donuz.get(0) kullanılmıyor garip bir şəkilde
                karsilama_textView.setText(donus.get(0).toString());

                //gelen ses verisinini müzik listesi arayüzüne göndermemi sağlayan metod
                SecilenMuzikEkrani(karsilama_textView.getText().toString());
            }
            break;
        }
    }
}
```

```
private void SecilenMuzikEkrani(String gelen_ses) {  
  
    //bulunan şarkı;  
    String bulunanansarki = "";  
    //  
    int bulunansarki_id=0;  
  
    // şarkı varsa muzik listesini göndermemizi sağlayak kontrol değişkeni  
    boolean sarki_varmi = false;  
  
    //okucanak veri  
    String data = "";  
    // tüm dosyayı yazıya çevirin  
    StringBuffer stringBuffer = new StringBuffer();  
    // kaynaktan txt dosyasını al  
    InputStream is = this.getResources().openRawResource(R.raw.muzikler);  
    //Toplu yazılı okucu değişkenine aktar  
    BufferedReader reader = new BufferedReader(new InputStreamReader(is));  
    //satır sayı numarası  
    int no = 0;  
    //dizi indexi  
    int index = 0;  
    //sarki idsi  
    int sarki_id;  
    // listenin 0. elemanı işi bozuyor boş dolduralım  
    tum_sarki_listesi.add( index: 0, element: "bos");  
    tum_sarki_idleri.add( index: 0, element: 0);
```

```

if (is != null) {
    try {
        while ((data = reader.readLine()) != null) {
            //txt datasini okuyoruz
            stringBuffer.append(data + "\n" + "\t");
            //satir olarak kesiyoruz
            String newItem = stringBuffer.toString();
            //sarkinin idsini aliyoruz
            sarki_id = Integer.valueOf(stringBuffer.toString().split( regex: "," )[0]);
            // satirin 1.karakteri virgül ise
            if (newItem.charAt(1) == ',') {
                //virgünden sonra data verisinin sonuna kadar al
                newItem = newItem.substring(2, newItem.length());
            // satirin 2.karakteri virgül ise
            } else if (newItem.charAt(2) == ',') {
                // virgünden sonrayi al
                newItem = newItem.substring(3, newItem.length());
            } else if (newItem.charAt(3) == ',') {
                // virgünden sonrayi al
                newItem = newItem.substring(4, newItem.length());
            }
            no++;
            // sıra numarası ile sarkinin ismini ilk harfleri büyük olacak sekilde
            tum_sarki_listesi.add(no, element: no + "." + ilkHarfBuyuk(newItem));
            //resim numarası da bu şeke olacak
            tum_sarki_idleri.add(no,no);

            // yeni item güzel görünüşün bigimi
            // burada ilk harflerini büyültüp formatlıyoruz şarkı ismini
            newItem=ilkHarfBuyuk(newItem).toLowerCase().replace( target: "-", replacement: "");

            // bu kısımda ses ile söylenen şarkıyı direk buluyor
            if (newItem.toLowerCase().trim().contains( gelen_ses.toLowerCase().trim() )) {
                // index numarasını aliyoruz
                index++;
                // bulunan şarkıyı düzenliyoruz nuamrasiyla
                bulunanansarki = no + "." + ilkHarfBuyuk(newItem);
                // artık bir şarkımız var
                sarki_varmi = true;
                // bulunan şarkı id ilse şarkı idmi eşleştiriyorum
                bulunanansarki_id=sarki_id;

            }
            // string buffuer temizleylim tekrar tekrar almal için
            stringBuffer.delete(0, stringBuffer.length());
        }

    } catch (IOException e) {
        e.printStackTrace();
    }
    // dizinin 0.indexini siliyiyoruz
    tum_sarki_listesi.remove( index: 0 );
    tum_sarki_idleri.remove( index: 0 );
}

```

Ses bulundu ise secilen muzik ekranına yönlendiriyoruz..

```

// aktif basılan şarkı var ise bu şarkıyı çalıracak olan başka bir arayüze gönderiyoruz
if(sarki_varmi) {

    //secilen müziği ilgili arayüze gönderiyoruz ve başlatıyoruz
    Intent secilenm_intent= new Intent( packageContext: sanatcilaragoresarkilarActivity.this,secilenMuzikActivity.class);

    secilenm_intent.putExtra( name: "muzikkey",bulunansarki_id);
    secilenm_intent.putExtra( name: "muzikismi",bulunanansarki.trim());
    secilenm_intent.putExtra( name: "muzik_listesi",tum_sarki_listesi);
    secilenm_intent.putExtra( name: "muzik_id_listesi",tum_sarki_idleri);
    startActivity(secilenm_intent);

}

else { // eğer şarkı yoksa
    // Böyle bir şarkı bilmiyorum mesajı kullanıcıya göster
    Toast.makeText(getApplicationContext(), text: "Böyle Bir Şarkı Bilmiyorum :(",Toast.LENGTH_SHORT).show();
    //Cihazma da gönder ve baksın oda bulamacaktır..
    // socket boş deilse yani bluetooth bağlantısı kopmuş mu ?
    if (bluetoothEkle.btSocket != null) {
        try {
            // kopmamış ise
            bluetoothEkle.btSocket.getOutputStream().write("Böyle bir şarkı bilmiyorum".getBytes());
        } catch (IOException e) {
            //kopmuş ise
            Toast.makeText(getApplicationContext(), text: "Veri Gonderilemedi!", Toast.LENGTH_SHORT).show();
        }
    }
}

private String ilkHarfBuyuk(String str)
{
    // str Stringinin içindeki kelimelerin ilk harfleri büyük diğerleri küçük yapılır.
    char c = Character . toUpperCase ( str . charAt ( 0 ));
    //ilk harfini buyuttuk
    str = c + str . substring ( 1 );
    //buyutulen ilk harften sonra kelimenin diger harflerini ekledik.
    String bosluk=" ";
    //gelen yazının karakter uzunluğuna kadar
    for (int i = 1 ; i < str.length();i++)
    {
        //__ kadar olan kısım bir kelime medir
        if (str.charAt(i)=='_')
        {
            //__' kadar teker teker bakıp __'rasında baş harfi büyült
            c=Character.toUpperCase(str.charAt ( i+1));
            //bosluk bırakarak yeniden düzenle
            str = str.substring(0,i)+bosluk +c + str.substring(i+2);

        }
    }

    //düzenlenmiş kelimeleri geri döndürüyoruz
    return str;
}
}

```

Buraya kadar ki kısmında amacımız iki yoldan da müzik listesine ulaşmaktı. Şimdi de müzik listesine geçelim..

### 1.13. Şarkı Listesi Arayüzü

Müzik listesi, gelen şarkıcıya göre şarklarının listelendiği bir arayüz. Bu şekilde daha düzenli bir yapı kurulmuş oluyor. Hangi şarkıcıya ait tüm şarkıları görebiliyordum.



Global değişkenlerim;

```
//değişkenlerim
ListView muzik_listView;
ImageButton voicerobot_imageButton;

//dizilerim
ArrayList<String> etiket = new ArrayList<~>();
ArrayList<Integer> imgid = new ArrayList<~>();
```

Arayüz açılırken çalışacak kodlar;

```

@Override
protected void onCreate(@Nullable Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.acitivity_muzik);

    //arayüzümüz ile java arasındaki bağlantıyı sağlıyoruz
    muzik_listView=(ListView)findViewById(R.id.m_muzik_listview);
    voicerobot_imageButton=(ImageButton)findViewById(R.id.tv_voicerobot_imageButton);

    // SecilenMüzik ve Müzik listesi
    // Buraya gönderdiğimiz gerekli veriler alınıp değişkenlere atılıyor.
    Bundle gelenVeri = getIntent().getExtras();

    CharSequence gelen_sarkicismi=gelenVeri.getCharSequence( key: "sanatcismi");
    // gelen şarkıcı ismi boş bir liste ise hepsini göster..
    if(((String)gelen_sarkicismi).contains("Tüm Liste")) gelen_sarkicismi = "";

    //Txt Dosyasındaki verileri okuyalım..
    String data = "";
    // satır sonu kes
    StringBuffer stringBuffer = new StringBuffer();
    //muzik.txt dosyasını uygulama kaynağından al
    InputStream is = this.getResources().openRawResource(R.raw.muzikler);
    // yüklenmiş veri okuyucu
    BufferedReader reader = new BufferedReader(new InputStreamReader(is));
    // satır sıra numarası
    int no=0;
    // dizi indexi
    int index=0;

```

```

// resim id ve şarkı ismi dizisi 0.indexi boş atıyoruz
imgid.add( index: 0, element: 0);
etiket.add( index: 0, element: "bos");
if(is !=null){
    try {
        while ((data=reader.readLine()) !=null) {
            // txt dosyasından verileri çekiyoruz fakat boşluktan keserek
            stringBuffer.append(data + "\n" + "\t");

            //yeni item olarak atıyoruz
            String newItem = stringBuffer.toString();
            /*
                770 tane veri olduğu için ve sıralı şekilde sayıldığını düşünürsek
                bir, iki ve üç haneli sayılarından oluşuyor.

            */
            // eğer satırın 1. karakterinde , varsa o bir hanelidir.
            if (newItem.charAt(1) == ',') {
                // o hanenin , den sonradan başlayarak veri sonuna kadar
                newItem = newItem.substring(2, newItem.length());
                // eğer satırın 2. karakterinde , varsa o iki hanelidir.
            } else if (newItem.charAt(2) == ',') {
                newItem = newItem.substring(3, newItem.length());
                // eğer satırının 3.karakterinde , varsa o üç hanelidir
            } else if (newItem.charAt(3) == ',') {
                // o hanenin , den sonradan başlayarak veri sonuna kadar
                newItem = newItem.substring(4, newItem.length());
            }
            //satır sayı numarası
            no++;

            //gelen şarkıcı ismini arıyoruz var ise
            if (ilkHarfBuyuk(newItem).indexOf((String) gelen_sarkicismi) != -1) {
                index++;
                // diziye atıyoruz
                imgid.add(index,no);
                etiket.add(index, element: index + "." + ilkHarfBuyuk(newItem));
            }

            // satırla isimiz bitti sil
            stringBuffer.delete(0, stringBuffer.length());
        }
    }
}

```

```

        } catch (IOException e) {
            e.printStackTrace();
        }
        // 0.indexteki elemanları siliyoruz
        etiket.remove(0);
        imgid.remove(0);

        //Dizilerimiz hazır adapter'e bağlayıp kullanalım

        ListViewAdapter mListviewAdapter = new ListViewAdapter(muzikActivity.this,etiket,imgid);
        muzik_listView.setAdapter(mListviewAdapter);

        //muzik listesinden bir şarkı seçtiğimizde
        muzik_listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {
                //seçilen itemin sıra numarasına göre
                //resim ve şarkıcı ismini
                int muzik_key = imgid.get(i);
                CharSequence muzik_isim_charSequence = etiket.get(i);

                // muzik çalma ekranına yönlendir. Extra veriler ile
                Intent secilenm_intent= new Intent(muzikActivity.this,secilenMuzikActivity.class);

                // öbür tarafa gönderilen extra veriler
                secilenm_intent.putExtra("muzikkey",muzik_key);
                secilenm_intent.putExtra("muzikismi",muzik_isim_charSequence);
                secilenm_intent.putExtra("muzik_listesi", etiket);
                secilenm_intent.putExtra("muzik_id_listesi",imgid);
                startActivity(secilenm_intent);

            }
        });
    }
}

```

```

private String ilkHarfBuyuk(String str)
{
    // str Stringinin içindeki kelimeлерin ilk harfleri büyük diğerleri küçük yapılır.
    char c = Character . toUpperCase ( str . charAt ( 0 ) );
    //ilk harfini buyuttuk
    str = c + str . substring ( 1 );
    //buyutulen ilk harften sonra kelimenin diger harflerini ekledik.
    String bosluk=" ";
    //karakter uzunluğuna kadar
    for (int i = 1 ; i<str.length();i++)
    {
        // _ karaktere kadar olan kısmı al
        if (str.charAt(i)=='_')
        {
            //ilk karakterini büüt
            c=Character.toUpperCase(str.charAt (i+1));
            //aralarda boşluk olacak şekilde yeniden düzenle
            str = str.substring(0,i)+bosluk +c + str.substring(i+2);

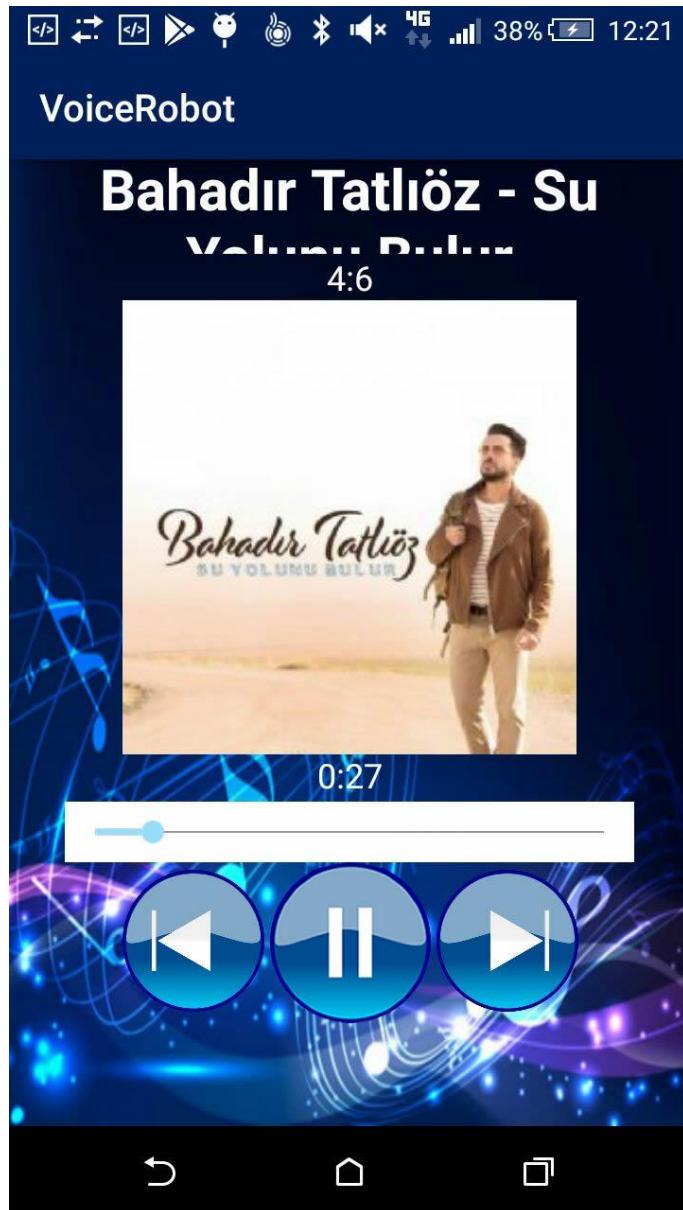
        }
    }

    //düzenlenmiş veri tekrardan geri döndürülüyor
    return str;
}

```

Şarkı listelerini bu şekilde görüyoruz. İstediğimiz şarkıyı seçtiğimizde şarkı çalma ekranına gidiyoruz.

#### **1.14. Seçilen Şarkı Arayüzü**



Bu arayüzde bir çok olasılık mevcut. Yine beni çok uğraştıran arayüzlerden bir tanesiydi...

```
public class secilenMuzikActivity extends AppCompatActivity {

    //Textviy,baslama,bitis
    public TextView muzikIsmi,txtBaslama,txtBitis;
    //medioplayer
    private MediaPlayer mediaPlayer;
    //başlangıç zamanı
    private double startTime = 0;
    //bitiş zamanı
    private double finalTime = 0;
    // handler metodu
    private Handler myHandler = new Handler();

    //muzik seekbar'i
    private SeekBar seekbar;
    //ImageButton'lari
    private ImageButton play_pause_imageButton,previous_imageButton,skipforward_imageButton;
    // tek zaman
    public static int oneTimeOnly = 0;

    // aktif müzikler
    int AktifmuzikID;
    String AktifmuzikIsmi;
    String _AktifmuzikIsmi;
    //basıldı mı buttona
    int basıldı=0;
    // ileri geri mi
    boolean ilerigeri=false;

    // bluettoh eklensin mi
    BluetoothEkle bluetoothEkle;

    //albüm kapığı
    ImageView album_imageView;

    //muzik dizileri
    ArrayList<String> muzikdizisi = new ArrayList<~>();
    ArrayList<Integer> muzikidleri = new ArrayList<~>();
```

Arayüz Açılrken;

```
@Override
protected void onCreate(@Nullable Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_secilenmuzik);

    //bluetooth ekle
    bluetoothEkle = new BluetoothEkle( context: secilenMuzikActivity.this);

    // gelen veriyi aliyoruz
    Bundle gelenVeri = getIntent().getExtras();
    int gelenid=gelenVeri.getInt( key: "muzikkey");
    CharSequence gelen_muzikismi=gelenVeri.getCharSequence( key: "muzikismi");

    // muzik dizileri
    muzikdizisi = gelenVeri.getStringArrayList( key: "muzik_listesi");
    muzikidleri = gelenVeri.getIntegerArrayList( key: "muzik_id_listesi");

    //Controller elementlerimizi tanimliyoruz...
    muzikIsmi = (TextView)findViewById(R.id.SarkiIsmi_textView);
    album_imageView = (ImageView) findViewById(R.id.album_imageView);
    txtBaslama=(TextView)findViewById(R.id.txtBaslama);
    txtBitis =(TextView)findViewById(R.id.txtBitis);
    seekbar = (SeekBar)findViewById(R.id.muzik_seekBar);
    play_pause_imageButton = (ImageButton)findViewById(R.id.play_pause_imageButton);
    previous_imageButton = (ImageButton) findViewById(R.id.previous_imageButton);
    skipforward_imageButton = (ImageButton) findViewById(R.id.skipforward_imageButton);

    // kaydirma metod eventi
    muzikIsmi.setMovementMethod(new ScrollingMovementMethod());
```

```

// play mi pause mi onu anlayağımız etiket
play_pause_imageButton.setTag("play");

// gelen muzik idsi
AktifmuzikID = gelenid;
// aktif muzik ismini alıyoruz buraya gelen veri satırından
AktifmuzikIsmi= ((String) gelen_muzikismi).split( regex "\\\\[1]).toString();

// formatlı ve temiz
_AktifmuzikIsmi=AktifmuzikIsmi;
muzikIsmi.setText(_AktifmuzikIsmi);

// - çift boşlukta cihazım algılamıyor tek boşluğa indirmem gerek
AktifmuzikIsmi=AktifmuzikIsmi.replace( target: "-", replacement: ""); // çizgi yoket
AktifmuzikIsmi=AktifmuzikIsmi.replace( target: " ", replacement: " "); // tek boşluğa düşür

// media player oluşturup yolunu belirliyoruz

mediaPlayer= new MediaPlayer();
String media_path = "/storage/ext_sd/VoiceRobot/Music/x"+AktifmuzikID+".mp3";
mediaPlayer.setAudioStreamType(AudioManager.STREAM_MUSIC);
Uri uri = Uri.parse(media_path);
try {
    mediaPlayer.setDataSource( context: this,uri);
    mediaPlayer.prepare();
} catch (IOException e) {
    e.printStackTrace();
}

//resim yolunu id'sini al
String image_path="/storage/ext_sd/VoiceRobot/b_AlbumCover/x"+AktifmuzikID+".jpg";
Uri kapak_uri = Uri.parse(image_path);
album_imageView.setImageURI(kapak_uri);

// seekbar dokunulmadı olarak işaretle
seekbar.setClickable(false);

skipforward_imageButton.setOnClickListener((view) → {

    // ileri geri metodları
    ileriGeriClick( _ilerigeri: "ileri");

});

previous_imageButton.setOnClickListener((view) → {
    // ileri geri metodları
    ileriGeriClick( _ilerigeri: "geri");
});

```

```

    // başla metodu
    play(play_pause_imageButton);

    // cihazima aktif müzik ismini gönder
    if (bluetoothEkle.btSocket != null) {
        try {

            bluetoothEkle.btSocket.getOutputStream().write(AktifmuzikIsmi.trim().toString().getBytes()); // sondaki iki paraziti alma
        } catch (IOException e) {
            Toast.makeText(getApplicationContext(), text: "Bağlantı Koptu !", Toast.LENGTH_SHORT).show();
        }
    }
}

```

Buraya kadar kurulumunu yaptık. Seekbar değeri değiştiğinde olacak olanlar;

```

// seekbara dokunulduğunda müzik o yerden devam etsin
seekbar.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
    @Override
    public void onProgressChanged(SeekBar seekBar, int p, boolean b) {
        // dokunulur boş değilse
        if(b) {

            // başlangıç geçerli zamanı
            startTime = mediaPlayer.getCurrentPosition();

            //Muzigin basıldıgı andan itibaren geçen sureyi ,startTimeField controller ina yazdırıyoruz..
            txtBaslama.setText(String.format("%d:%d",
                TimeUnit.MILLISECONDS.toMinutes((long) startTime),
                TimeUnit.MILLISECONDS.toSeconds((long) startTime) -
                    TimeUnit.MINUTES.toSeconds(TimeUnit.MILLISECONDS.toMinutes((long) startTime))));

        }

        // dakika ve saniyeyi bölüp dakikayı saniyeye çevirip toplam saniyeye atıyoruz
        String [] dakika_saniye=txtBaslama.getText().toString().split(":");
        double top_saniye = ( Integer.parseInt(dakika_saniye[0]) * 60 ) + Integer.parseInt(dakika_saniye[1]);

        // media playeri gerekli saniyeye keydir
        mediaPlayer.seekTo(p);

        //cihaza şifreli bir şekilde sondaki paraziti silerek temiz kodu gönder
        if (bluetoothEkle.btSocket != null) {
            try {

                bluetoothEkle.btSocket.getOutputStream().write((AktifmuzikIsmi.toString().substring(0,AktifmuzikIsmi.toString().length()-2)).toString()+"-seekto:"+ top_saniye.getBytes());
            } catch (IOException e) {
                Toast.makeText(getApplicationContext(), text: "Bağlantı Koptu !", Toast.LENGTH_SHORT).show();
            }
        }
    }

    @Override
    public void onStartTrackingTouch(SeekBar seekBar) {
    }

    @Override
    public void onStopTrackingTouch(SeekBar seekBar) {
    }
});

```

Adım adım kodların açıklaması var. Tekrardan açıklarsam çok anlaşılır olmaz.

Devam edelim..

İleri Geri durumları...

```

int degistir_id=0;
private void ileriGeriClick(String _ilerigeri){

    for (int i=0;i<muzikidleri.size();i++) { // direk search edicek bişey bulamadım

        if(muzikidleri.get(i) == AktifmuzikID) {
            //AktifmuzikIsmi=rtcoeng(AktifmuzikIsmi);
            // Toast.makeText(getApplicationContext(), "Bir sonraki şarkı id: " + (i+1)+" index", Toast.LENGTH_SHORT).show();
            if(_ilerigeri=="ileri" && muzikidleri.size()-2 >= i) {degistir_id=(i+1); } else if(_ilerigeri=="geri" && 0<i) {degistir_id=(i-1); }
            // Toast.makeText(getApplicationContext(),String.valueOf(muzikidleri.size()),Toast.LENGTH_SHORT).show();
            break;
        }
    }

    AktifmuzikID=muzikidleri.get(degistir_id);
    AktifmuzikIsmi=muzikdizisi.get(degistir_id);

    mediaPlayer= new MediaPlayer();

    _AktifmuzikIsmi=AktifmuzikIsmi; // tam takır düzgün müzik ismini saklamalıyım

    // - çift boşlukta cihazım algılamıyor tek boşluğa indirmem gerek komutu cihaza temiz göndermemeliyim
    AktifmuzikIsmi=AktifmuzikIsmi.replace( target: "-", replacement: ""); // gizgi yoket
    AktifmuzikIsmi=AktifmuzikIsmi.replace( target: " ", replacement: " "); // tek boşluğa düşür
    AktifmuzikIsmi=AktifmuzikIsmi.split( regex: "\\\\"[1].toString();
}

```

```

// aktif müzik ismi
String media_path ="/storage/ext_sd/VoiceRobot/Music/x"+AktifmuzikID+".mp3";
mediaPlayer.setAudioStreamType(AudioManager.STREAM_MUSIC);
Uri uri = Uri.parse(media_path);
try {
    mediaPlayer.setDataSource( context: secilenMuzikActivity.this,uri);
    mediaPlayer.prepare();
} catch (IOException e) {
    e.printStackTrace();
}
String image_path="/storage/ext_sd/VoiceRobot/b_AlbumCover/x"+AktifmuzikID+".jpg";
Uri kapak_uri = Uri.parse(image_path);
album_imageView.setImageURI(kapak_uri);

seekbar.setClickable(false);
// ileri geri aktif
ilerigeri=true;
play(play_pause_imageButton);

// aktif müzik ismini sondaki 2 haneli paraziti göndermeyecek şekilde alıyoruz
if (bluetoothEkle.btSocket != null) {
    try {
        bluetoothEkle.btSocket.getOutputStream().write(AktifmuzikIsmi.toString().substring(0,AktifmuzikIsmi.toString().length()-2).getBytes());
    } catch (IOException e) {
        Toast.makeText(getApplicationContext(), text: "Bağlantı Koptu !", Toast.LENGTH_SHORT).show();
    }
}

```

Oynatma metodu ile devam ediyoruz..

```

public void play(View view){
    muzikIsmi.setText(_AktifmuzikIsmi);
    scrool_Y=0;

    // basıldı ise
    basıldı++;

    // ileri geri aktif ise
    if(ilerigeri) {play_pause_imageButton.setTag("play");}

    if(String.valueOf(play_pause_imageButton.getTag()) == "pause") { // ileri geri true ise basılıp basılmamsına bakma

        if(mediaPlayer.isPlaying()) {
            // socket
            // şarkıyı durdur
            if (bluetoothEkle.btSocket != null) {
                try {

                    bluetoothEkle.btSocket.getOutputStream().write("şarkıyı durdur".toString().getBytes());
                } catch (IOException e) {
                    Toast.makeText(getApplicationContext(), text: "Bağlantı Koptu !", Toast.LENGTH_SHORT).show();
                }
            }
        }
        // arkaplana play resmi
        play_pause_imageButton.setImageResource(R.drawable.play);
        // arkaplana play verme
        play_pause_imageButton.setTag("play");
        //media player durdur
        mediaPlayer.pause();

    }else {
        birkere++;
        // Activity yüklenliğinde şarkıyı oynat komutu cihazımı gitmesin çünkü şarkı sayfa yüklenliğinde zaten çalışıyor..
        // ..Eğer bir daha bu komutu gönderirsek cihazımız..
        // ..olmayan şarkıyı oynatmak istiyeciktir.
        if(birkere >1 && !ilerigeri) {
            // socket
            if (bluetoothEkle.btSocket != null) {
                try {

                    bluetoothEkle.btSocket.getOutputStream().write("şarkıyı oynat".toString().getBytes());
                } catch (IOException e) {
                    Toast.makeText(getApplicationContext(), text: "Bağlantı Koptu !", Toast.LENGTH_SHORT).show();
                }
            }
        }
    }
}

```

```

        // arkaplana pause atayın
        play_pause_imageButton.setImageResource(R.drawable.pause);
        play_pause_imageButton.setTag("pause");
        mediaPlayer.start();
        ilerigeri=false; // ileri geri görevini tamamlamıştır kapanabilir.

    }

    //Sesin millisaniye cinsinden toplam süresini öğreniyoruz...

    //sesin duran yapısı
    finalTime = mediaPlayer.getDuration();
    //sesin geçerli pozisyonu
    startTime = mediaPlayer.getCurrentPosition();

    if(oneTimeOnly == 0){ // tek zamanlı süre varsa
        // maksimum seekbar değerini finaltime ver
        seekbar.setMax((int) finalTime);
        //tek zamanlı süre yoksa
        oneTimeOnly = 1;
    }

    //Muzığın toplamda ne kadar süre olduğunu endTimeField controller'ına yazdırıyoruz...
    txtBitis.setText(String.format("%d:%d",
        TimeUnit.MILLISECONDS.toMinutes((long) finalTime),
        TimeUnit.MILLISECONDS.toSeconds((long) finalTime) -
            TimeUnit.MINUTES.getSeconds(TimeUnit.MILLISECONDS.
                toMinutes((long) finalTime)))
    );
    //Muzığın başladığı andan itibaren geçen süreyi ,startTimeField controller'ına yazdırıyoruz...
    txtBaslama.setText(String.format("%d:%d",
        TimeUnit.MILLISECONDS.toMinutes((long) startTime),
        TimeUnit.MILLISECONDS.toSeconds((long) startTime) -
            TimeUnit.MINUTES.getSeconds(TimeUnit.MILLISECONDS.
                toMinutes((long) startTime)))
    );
    //Muzığın hangi sürede olduğunu göstermek için, seekbar kullanarak gösteriyoruz...
    seekbar.setProgress((int)startTime);
    scroll_Y=0;
    if(birkere <= 1) {
        myHandler.postDelayed(UpdateSongTime, delayMillis: 100);
    }
    //Muzik suan calan durumnda olduğu için , pause true yapıyoruz(durdurulabilir olması için...)
}

//Muzığın calma süresini guncelleyerek,sesin çalmasını devam etmesini saglıyoruz...

```

```

//müziğin sürekli çalışacak metodu
private Runnable UpdateSongTime = new Runnable() {
    public void run() {

        // gecerli şarkı süresini
        startTime = mediaPlayer.getCurrentPosition();

        txtBaslama.setText(String.format("%d:%d",
            TimeUnit.MILLISECONDS.toMinutes((long) startTime),
            TimeUnit.MILLISECONDS.toSeconds((long) startTime) -
            TimeUnit.MINUTES.toSeconds(TimeUnit.MILLISECONDS.
                toMinutes((long) startTime)));
    };

    //Muzığın hangi sürede olduğunu göstermek için, seekbar kullanarak gösteriyoruz...

    seekbar.setProgress((int)startTime);

    // 10'ar 10'ar kaydırırmaya çalış
    scrool_Y+=10;

    // müzik ismi belirlenen araliktaysa eğer çok uzunsa aşağıdaki iften başlar
    if(scrool_Y > 70 && muzikIsmi.getText().toString().length() <= 22) scrool_Y=0;
    else
    if(scrool_Y > 140 && muzikIsmi.getText().toString().length() > 22) scrool_Y=0;
        // kaydır
        muzikIsmi.scrollTo( x: 0, scrool_Y);

    // müzik çalarsa
    if(mediaPlayer.isPlaying()){
        // müzik calıyorsa durdur resmi
        play_pause_imageButton.setImageResource(R.drawable.pause);
    }else{
        // müzik çalışmıyorsa oynat resmi
        play_pause_imageButton.setImageResource(R.drawable.play);
    }
    // saniyede çalışacak metod
    myHandler.postDelayed( @ this, delayMillis: 100);

    //başlama zamanı ile bitiş zamanı eşitse şarkı bitmiştir
    if(txtBaslama.getText().toString().contains(txtBitis.getText().toString())){
        ileriGeriClick( _ilerigeri: "ileri");
    }
}
};


```

Telefon tuşlarının kurucu metodlarına bir bakalım...

```

@Override
public void onBackPressed() { // telefon geri butonuna basılırsa bağlantı düşün
    super.onBackPressed();
    bitir_cik();
    //calismiyor bu kısım
    return;
}

@Override
public boolean onKeyDown(int keyCode, KeyEvent event) {
    if (keyCode == KeyEvent.KEYCODE_BACK) { // geri tuşu
        bitir_cik();
    }
    // ses arttır tuşuna basıldığında
    if (keyCode == KeyEvent.KEYCODE_VOLUME_UP){
        // socket
        if (bluetoothEkle.btSocket != null) {
            try {
                // şifreli bir şekilde cihaza sesi arttırmak
                bluetoothEkle.btSocket.getOutputStream().write((AktifmuzikIsim.substring(0,AktifmuzikIsim.length()-2) + "-ses arttır").toString().getBytes()); // Parasit geliyor
            } catch (IOException e) {
                Toast.makeText(getApplicationContext(), text: "Bağlantı Koptu !", Toast.LENGTH_SHORT).show();
            }
        }
    }
    // sesi azalt tuşuna basıldığında
    if (keyCode == KeyEvent.KEYCODE_VOLUME_DOWN){
        // socket
        if (bluetoothEkle.btSocket != null) {
            try {
                // şifreli ses azalt mesajı gönderiyoruz
                bluetoothEkle.btSocket.getOutputStream().write((AktifmuzikIsim.substring(0,AktifmuzikIsim.length()-2) + "-ses azalt").toString().getBytes()); // Parasit geliyor
            } catch (IOException e) {
                Toast.makeText(getApplicationContext(), text: "Bağlantı Koptu !", Toast.LENGTH_SHORT).show();
            }
        }
    }
    return true;
}

```

Geri tuşuna basıldığında çalışacak metod...

```

private void bitir_cik(){
    // müziği durdur
    mediaPlayer.stop();

    // cihaza gönder
    if (bluetoothEkle.btSocket != null) {
        try {
            bluetoothEkle.btSocket.getOutputStream().write("şarkıyı durdur".toString().getBytes());
        } catch (IOException e) {
            Toast.makeText(getApplicationContext(), text: "Bağlantı Koptu !", Toast.LENGTH_SHORT).show();
        }
    }
    // Arayüzü kapat...
    finish();
}

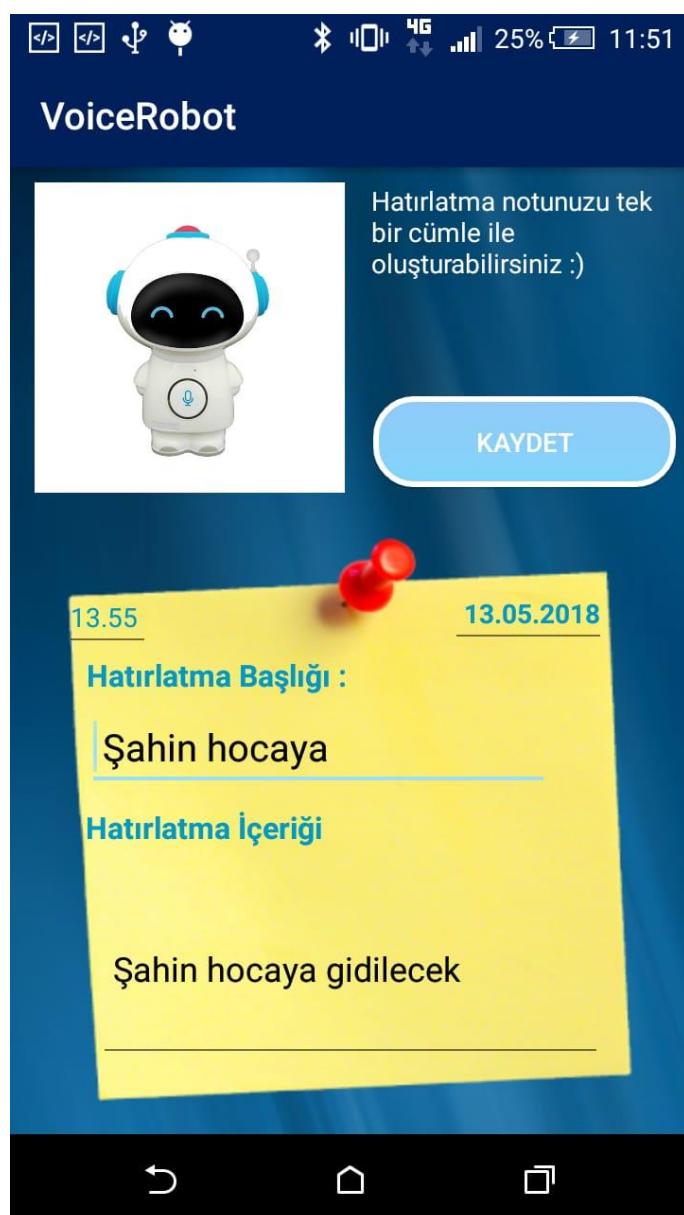
```

## 1.15. Hızlı Not Alma ve Hatırlatıcı Arayüzü

Ben projemde her zaman elimden geldiğince kullandığım platformlar(Android,Arduinuo) üzerinde zamanım yettiğince değişik bir şeyler yapmak istedim.

Pek çokumuz yapacağımız şeyleri unuturuz. Not almaya yada bir yere yazmaya üşeniriz. Zaten Google'da biz yazmaya üşeneler gurubuna sesi yazıya dönüştüren bir servis yapmıştır. Biz yazılımcı adayları da bu özelliğini işleyip kullanmamız için bunu çeşitli platformda eklenti olarak sunmuşlar, zaten bunları biliyoruz önceki arayüzlerimizden. Unuttuğumuz görevlerimizi tek bir cümle ile telefona söylesek ne güzel olur ? Mesela; “Bugün saat 13:55'de Şahin Hocaya gidilecek.” Konuşma buttonuna basıp bunları söylediğimiz zaman; bana zamanı geldiğinde hatırlatmasını

sağlanmasıını isdeim. Bir Apple kullanıcısı olmadığım için bu özelliğin Siri'de olduğunu bilmiyordum yani fikir aklıma geldiğinde bilmiyordum. Sonradan öğrendimki Apple bunu yapmış. Düşündüm ki madem konumuz ses ile komut işleme bunu projemde kullanayım. Zaten Android tarafında böyle bir uygulama yok hemde böyle bir uygulama geliştirmiş olurum diye düşünmüştüm. Herhangi bir hazır algoritma kullanmadım. Kendi alogitmmamı kendim yazmamıştım. Önce arayüzümü görelim ardından algoritmamıza geçelim..



Algoritmanın başında şunu belirlemeliydim. Bana hangi veriler lazım. Yani veritabanı tablolarımı oluşturmamalıydım."Tarih","Zaman","Görev Başlığı","Görev İçeriği" şeklinde düşünüp cümleyi o şekilde parçalara ayırip formatlı bir şekilde

kaydedip bir servis ile sürekli zaman kontrolü yaptııp vakti geldiğinde veriyi formatlanmış halinden uygun algoritmayla alıp ayırip kullanıcının önüne koymalıydım. Koyarkende aynı düzende tekrar göstermeliydim ayrıca telefon internete bağlı isede Google'nin sesli bayan sesi bu mesajı sesli olarak da söylese kullanıcı telefonun ekranını açmadan dair "Ha benim Şahin Hoca'ya gitmem gerekiyordu !" diyecek ve hatırlatma işini tamamlamış olacağız.

Başlayalım...

```
//ImageButton
ImageButton ht_voicerobot_imageButton;

//arayüz değişkenleri
TextView speak_textView;
EditText hatirlatmabaslik_editText;
EditText hatirlatmalcerigi_editText;
EditText time_editText;
EditText date_editText;
Button kaydet_button;
//İhtiyaç duyulan ses istediği girişi için istenen karakter uzunluğu
private final int REQUEST_CODE_SPEECH_INPUT = 100;

//hatırlatacak içeriğin bulunduğu tablo txt
final static String fileName = "hatirlat.txt"; // bu nedense gizli
//cihaz hafızası yolù
final static String path = Environment.getExternalStorageDirectory().getAbsolutePath() + "/" ;
//bu activity ismi ve özellik belirten değişkeni
final static String TAG = hatirlaticiActivity.class.getName();
```

Arayüz başlarken...

```

@Override
protected void onCreate(@Nullable Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_hatirlatici);

    //arayüz elemanlarını java'ya bağlama
    ht_voicerobot_imageButton = findViewById(R.id.ht_voicerobot_imageButton);
    speak_textView = findViewById(R.id.speak_textView);
    hatirlatmabaslik_editText = findViewById(R.id.hatirlatmabaslik_editText);
    hatirlatmaIcerigi_editText = findViewById(R.id.hatirlatmaIcerigi_editText);
    time_editText = findViewById(R.id.time_editText);
    date_editText = findViewById(R.id.date_editText);
    kaydet_button = findViewById(R.id.kaydet_button);

    speak_textView.setText("Hatırlatma notunuzu tek bir cümle ile oluşturabilirsiniz :)");

    //Notification içeriği
    Bundle extras = getIntent().getExtras();
    if(extras!=null) {
        // içeriği alıyoruz
        String baslik = extras.getString( key: "baslik");
        String icerik = extras.getString( key: "icerik");
        String saat = extras.getString( key: "saat");
        String tarih = extras.getString( key: "tarih");

        time_editText.setText(saat);
        date_editText.setText(tarih);
        hatirlatmabaslik_editText.setText(baslik);
        hatirlatmaIcerigi_editText.setText(icerik);
    }
}

ht_voicerobot_imageButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        //ekran isteği..
        Intent intent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
        intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL, RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
        intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE, Locale.getDefault());
        //getdefault metodu seçili olan dili getirir telefondaki
        intent.putExtra(RecognizerIntent.EXTRA_PROMPT, value: "Konuşabilirisiniz..");

        try {
            hatirlaticiActivity.this.startActivityForResult(intent, REQUEST_CODE_SPEECH_INPUT);

        } catch (ActivityNotFoundException e) {
            Toast.makeText(getApplicationContext(), text: "Desteğlenmiyor :(", LENGTH_SHORT);
        }
    }
});

//servis çalışıp çalışmadığını kontrol eder.
if(!servisCalisiyorMu()) {
    Intent NotifyService = new Intent( packageContext: hatirlaticiActivity.this, hatirlaticiService.class);
    startService(NotifyService);
}

```

```

kaydet_button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        // gerekli clasıkları düşündürmek doğru veriye ulaşıyor ve uydurma get methodu dediğim kendi yöntemimle verileri servise taşıyoruz..
        // // // Ardından WriteFile ile şifrelenmiş veri satırını txt satırına kaydediyoruz.
        if(time_editText.getText().toString().length() == 5 && date_editText.getText().toString().length() == 10 && hatirlatmabaslik_editText.getText().toString().length() > 0 &&
           hatirlatmaicerigi_editText.getText().toString().length() > 0 ) {
            if (WriteFile( data + "time=" + time_editText.getText() + "date=" + date_editText.getText() + "&icerik=" + hatirlatmaicerigi_editText.getText())) {
                Toast.makeText( context: hatirlaticiActivity.this, text: "Başarılı bir şekilde hatırlatmanız eklendi !", Toast.LENGTH_SHORT).show();
            } else {
                Toast.makeText( context: hatirlaticiActivity.this, text: "Error save file!!!", Toast.LENGTH_SHORT).show();
            }
        }else{
            Toast.makeText(getApplicationContext(), text: "Lütfen gerekli yerleri düzgün bir şekilde giriniz !",Toast.LENGTH_LONG).show();
        }
    }
});

//Sesten sonra sorgu sonucu metodu
protected void onActivityResult(int requestCode, int resultCode, Intent data) {

    hatirlaticiActivity.super.onActivityResult(requestCode, resultCode, data);

    switch (requestCode) {
        case REQUEST_CODE_SPEECH_INPUT: {
            if (resultCode == RESULT_OK && data != null) {
                ArrayList<String> donus = data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
                // intentden dönen veriyi arrayliste atıyoruz.
                // gerekli arayüz elemanlarını boşaltıyoruz.
                time_editText.setText("");
                date_editText.setText("");
                hatirlatmabaslik_editText.setText("");
                hatirlatmaicerigi_editText.setText("");

                //ses parçaları
                String[] voiceParcalari = donus.get(0).split( regex: " " );
                // socket
                int birkere = 0;
                // özel isimle adlandırdığım günler
                boolean ozelgun=false;
                //ses parçalarını listeley
                for (int i = 0; i < voiceParcalari.length; i++) {
                    //ses parçalarının içinde bugün varsa
                    if (voiceParcalari[i].matches( regex: "bugün" )) {
                        //tarih bilgisini al
                        SimpleDateFormat bugun = new SimpleDateFormat( pattern: "dd.MM.yyyy" );
                        String bugun_tarih = bugun.format(new Date());
                        //ilgili formatta alıp bugün_tarihini
                        //textbox'a bugünün tarihini ata
                        date_editText.setText(bugun_tarih);
                        //evet bu özel gün
                        ozelgun=true;
                    }else
                        if(voiceParcalari[i].matches( regex: "yarın")){// yarın ise
                            //evet
                            ozelgun=true;
                        }
                }
            }
        }
    }
}

```

```

// txt dosya yazma metodu
private boolean WriteFile(String data) { // yazılacak veri

    try {
        new File(path).mkdir();
        File file = new File( pathname: path + fileName); //dosya yolu ve dosya ismi
        if (!file.exists()) { //txt cihazda kayitli deilse
            file.createNewFile(); // dosyayı yarat
        }
        //dosya kesme operatörü
        FileOutputStream fileOutputStream = new FileOutputStream(file, append: true);
        //gizgi hattini al
        fileOutputStream.write((data + System.getProperty("line.separator")).getBytes());
        // sorun yoksa true
        return true;
        //hata mesajları
    } catch (FileNotFoundException ex) {
        Log.d(TAG, ex.getMessage());
    } catch (IOException ex) {
        Log.d(TAG, ex.getMessage());
    }
    //sorun varsa false
    return false;
}

}
public boolean servisCalisiyorMu(){//Servis Çalışıyor mu kontrol eden fonksiyon

    //Activity service manager'i oluştur.
    ActivityManager manager = (ActivityManager) getSystemService(Context.ACTIVITY_SERVICE);
    for (ActivityManager.RunningServiceInfo service : manager.getRunningServices(Integer.MAX_VALUE)) {
        if (TAG.equals(service.service.getClassName())) {
            return true;
        }
    }
    return false;
}

else
if(voiceParcalari[i].matches( regex: "yarın")){// yarın ise
    //eve
    ozelgun=true;
}
if (Character.isDigit(voiceParcalari[i].charAt(0)) && !ozelgun) { // eğer rakam ise parçalardan biri özel gün deilse
if (date_editText.getText().length() == 0) { // tarih bossa
    SimpleDateFormat sdf_yil = new SimpleDateFormat(pattern: "yyyy");
    String currentYear = sdf_yil.format(new Date());
    date_editText.setText(voiceParcalari[i] + "." + currentYear);
}

} else {
    // saatı sıklıyoruz
    if (voiceParcalari[i].matches( regex: "saat") || voiceParcalari[i].matches( regex: "Saat")) {
        Toast.makeText(context this, text: "burada", Toast.LENGTH_SHORT).show();
        time_editText.setText(voiceParcalari[i + 1]);
    }
    // baslik ve iperigini sıklıyoruz
    if (!voiceParcalari[i].matches( regex: "saat") && !voiceParcalari[i].matches( regex: "Saat") && !voiceParcalari[i].matches( regex: "de") && !voiceParcalari[i].matches( regex: "bugün" )
        && !Character.isDigit(voiceParcalari[i].charAt(0))) {
        birkere++;
        if (birkere == 1) {
            hatirlatmabaslik_editText.setText(hatirlatmabaslik_editText.getText() + " " + voiceParcalari[i] + " " + voiceParcalari[i + 1]);
        }
        hatirlatmaicerigi_editText.setText(hatirlatmaicerigi_editText.getText() + " " + voiceParcalari[i]);
    }
}

}
}

```

```

// txt dosya yazma metodu
private boolean WriteFile(String data) { // yazılacak veri

    try {
        new File(path).mkdir();
        File file = new File( pathname: path + fileName); //dosya yolu ve dosya ismi
        if (!file.exists()) { //txt cihazda kayitli deilse
            file.createNewFile(); // dosyayı yarat
        }
        //dosya kesme operatörü
        FileOutputStream fileOutputStream = new FileOutputStream(file, append: true);
        //gizgi hattını al
        fileOutputStream.write((data + System.getProperty("line.separator")).getBytes());
        // sorun yoksa true
        return true;
        //hata mesajları
    } catch (FileNotFoundException ex) {
        Log.d(TAG, ex.getMessage());
    } catch (IOException ex) {
        Log.d(TAG, ex.getMessage());
    }
    //sorun varsa false
    return false;

}

public boolean servisCalisiyorMu(){//Servis Çalışıyor mu kontrol eden fonksiyon

    //Activity service manager'i oluştur.
    ActivityManager manager = (ActivityManager) getSystemService(Context.ACTIVITY_SERVICE);
    for (ActivityManager.RunningServiceInfo service : manager.getRunningServices(Integer.MAX_VALUE)) {
        if (TAG.equals(service.service.getClassName())) {
            return true;
        }
    }
    return false;
}

```

Buraya kadar olan kısımda cümlemizi parçalayıp şifreli bir şekilde veritabanına kaydettim.

Şimdi de bir servis yapıp telefonun arkaplanında çalışacak bir yapı kurmalıyız bu yapı sürekli arkaplanda çalışacak. Bu servis ile sürekli olarak txt dosyasını okuyacağımız ve zamanı gelen hatırlatma varmı bakıcaz..

Global değişkenler;

```

//sayı değişkenleri
int i=0;
// içerik
String ht_icerik;
// mediaPlayer
MediaPlayer mediaPlayer;

```

Program yaratılırken;

```

@Override
public void onCreate() {
    super.onCreate();

    new Thread(new Runnable() { // Yeni bir Thread (is_percisi) oluşturuyorum.
        @Override
        public void run() { // Thread'ım başladığında bitmemesi için while
            // loop'u is_percisi değişye soktum. senaryo gereği
            while (1 == 1) {
                try {
                    Thread.sleep(15000); // Her döngüde Thread'ımı 15000 ms uyutuyorum.
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
            }
            // ve 15 saniyem dolduktan sonra bildirimimi bassyorum.

            Date now = new Date();
            SimpleDateFormat t_Format = new SimpleDateFormat(pattern: "dd.MM.yyyy");
            SimpleDateFormat s_Format = new SimpleDateFormat(pattern: "kk:mm");

            // t'me saat/saat okuyor
            for (int i=0;i<Readfile().size()-1;i++) {
                ///// eslezen saat-dakika ve tarih varmı
                if( Readfile().get(i).indexOf(s_Format.format(now)) != -1 && Readfile().get(i).indexOf(t_Format.format(now)) != -1 ) {
                    //Hاتırlatma içeriği
                    ht_icerik=Readfile().get(i).substring(34);
                    // içeriği söyle
                    SoundPlay(ht_icerik);
                    //perk_parcalarını birleştirilelim.
                    String [] ht_icerik_parcaları=ht_icerik.split(" ");
                    //resim parçalarını uygun bir şekilde okuyoruz
                    BildirimGonder( NContentTitle: "Hatırlatma Notu : ", Readfile().get(i).substring(34), ht_Baslik: ht_icerik_parcaları[1]+ " "+ht_icerik_parcaları[2],s_Format.format(now),t_Format.format(now));
                }
            }
        }
    }).start(); // burada Thread'ımı başlatıyorum.
}

```

Metodlarımızı oluşturuyoruz..

```

// Dosya oku
public ArrayList<String> Readfile() {
    // topu okuyucu
    BufferedReader bufReader = null;
    ArrayList<String> listOfLines = null;
    boolean find=false;
    try {
        // dosya yolu ve dosya adı
        bufReader = new BufferedReader(new FileReader( fileName: path+fileName));
        listOfLines = new ArrayList<>();
        String line = bufReader.readLine();
        while (line != null) {

            //satır oku
            line = bufReader.readLine();
            //satır listesine satır ekle
            listOfLines.add(line);

        }
        bufReader.close();
    } catch (FileNotFoundException e) {
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    }
    return listOfLines;
}

```

```

//NContentTitle, NContentText, hatırlatma başlığı, hatırlatma saat, hatırlatma tarih
public void BildirimGonder(String NContentTitle, String NContentText, String ht_Baslik, String ht_Saat, String ht_Tarih) {
    NotificationManager NM = (NotificationManager) getSystemService(Context.NOTIFICATION_SERVICE);
    Notification bildirim;
    Intent in = new Intent(getApplicationContext(), hatırlaticiActivity.class);

    //Bildirim değişkenlerini giriyoruz
    in.putExtra( name: "baslik",ht_Baslik);
    in.putExtra( name: "icerik",NContentText);
    in.putExtra( name: "saat",ht_Saat);
    in.putExtra( name: "tarih",ht_Tarih);

    bildirim = new Notification.Builder(getApplicationContext()).
        //baslik
        setContentTitle(NContentTitle).
        //icerik
        setContentText(NContentText).
        // var söyleyen ses ile uyarılır
        setSound(Settings.System.DEFAULT_NOTIFICATION_URI).
        //ttresim
        setVibrate(new long[]{10, 500, 500, 500}).
        //renk
        setLights(Color.MAGENTA, 300, 500).
        // içeriğin isteği bu view'de
        setContentIntent(PendingIntent.getActivity(getApplicationContext(), i, in, i)).
        //küçük icon
        setSmallIcon(R.drawable.voicerobot).
        //otomatik tekrar tekrar
        setAutoCancel(true).getNotification();
    NM.notify( tag: "notify", i, bildirim); // bildirimini at
}

// sesli mesaj okuma taktiği
public void VoiceSoundPlay(String soylenecek){

    //sesli mesaj için ses
    mediaPlayer= new MediaPlayer();
    //ses yolu
    String media_path = "https://translate.google.com/translate_tts?ie=UTF-8&total=1&idx=0&textlen=32&client=tw-ob&q=" + soylenecek.replace( target: " ", replacement: "%20")+"&tlt=tr";
    //AudioManager tipi STREAM_MUSIC tipinde
    mediaPlayer.setAudioStreamType(AudioManager.STREAM_MUSIC);
    //uri değişkenin içine url adresini atıyoruz
    Uri uri = Uri.parse(media_path);
    try {
        //kaynağın url adresini ver
        mediaPlayer.setDataSource( context: this,uri);
        //dizئنل
        mediaPlayer.prepare();
    } catch (IOException e) {
        e.printStackTrace();
    }
    //sesi başlat
    mediaPlayer.start();
}

```

## 1.16. Televizyon Kumandası Arayüzü

Bir diğer arayüzümüz televizyon kumandası arayüzü.. Bu arayüzde de yine amaç ses ile belirli alogoritmalardan geçip hızlı bir şekilde sonuca ulaşmak. Herkesin evinde bir televizyon vardır. Televizyon uydularında genelde kanallar karışık oluyor. Yani bugün kimin evine gidersem gideyim bana “Enes bizim kanalları sıralar mısın ?” diyorlar. Aradığım kanalı direk söylesem de bana bir cihaz yorumlayıp o kanalı her nerdeyse bulup açsa güzel olur. Yani bir nevi akıllı kumanda olacak. Arduino tarafından bu iş esesen dönüyor orda ayrıntılı bahsetmiştim, biz sadece burada bu komutu gönderen bir mekanizma yazacağız.



Başlayalım..

```
//tv_listesi
ListView tv_listView;
//ses butonu
ImageButton voicerobot_imageButton;
//İhtiyaç duyulan ses istediği girişi için istenen karakter uzunluğu
private final int REQUEST_CODE_SPEECH_INPUT = 100;
//tv listesi
GridView tvgridview;

//bluetooth ekle
BluetoothEkle bluetoothEkle;

// robot karşılama mesajı
TextView karsilama_textView;

//tv isim ve tv imgid listesi
ArrayList<String> tv_isim = new ArrayList<~>();
ArrayList<String> tv_imgid = new ArrayList<~>();
```

Arayüz açılırken çalışacaklar..

```

@Override
protected void onCreate(@Nullable Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.acitivity_tv);

    //bluetooth ekle bu arayüze
    bluetoothEkle = new BluetoothEkle( context: tvActivity.this);

    // arayüz ile java arasındaki bağlantıyı sağlıyoruz
    tv_listView=(ListView)findViewById(R.id.tv_listView);
    voicerobot_imageButton=(ImageButton)findViewById(R.id.tv_voicerobot_imageButton);
    karsilama_textView = (TextView) findViewById(R.id.tv_karsilama_textView);
    tvgridview = (GridView) findViewById(R.id.tvlist_gridview);

    //Txt Dosyasındaki verileri okuyalım..
    String data = "";
    //toplu string tutucu
    StringBuffer stringBuffer = new StringBuffer();
    //kaynaktan tv_kodlarını alalım
    InputStream is = this.getResources().openRawResource(R.raw.tv_codes);
    //toplu okuyucuya kaynağı aktarıyoruz
    BufferedReader reader = new BufferedReader(new InputStreamReader(is));
    int no=0; // dizi index numarası
    tv_isim.add(no, element: "bos"); // tv ismi
    tv_imgid.add(no, element: "bos");// tv resim idleri
    //item id ve tv
    String newItemID;
    if(is !=null){
        try {
            //data
            while ((data=reader.readLine()) !=null){
                //txt datasını okuyoruz
                stringBuffer.append(data + "\n" + "\t");
                //satır olarak kesiyoruz
                String newItem=stringBuffer.toString();
                //şarkının idsini alıyoruz
                newItemID = stringBuffer.toString();
                // satırın 1.karakteri virgül ise
                if(newItem.charAt(1) == ',') {
                    //virgülden sonra data verisinin sonuna kadar al
                    newItem = newItem.substring(2, newItem.length());
                    // satırın 2.karakteri virgül ise
                }else

```

```

        if(newItem.charAt(2) == ',') {
            // virgülden sonrası al
            newItem = newItem.substring(3, newItem.length());
        }else if(newItem.charAt(3) == ','){
            // virgülden sonrası al
            newItem = newItem.substring(4, newItem.length());
        }
        //dizi index numarasını alıyoruz
        no++;
        //tv dizisine ekliyoruz
        tv_isim.add(no,ilkHarfBuyuk(newItem));
        tv_imgid.add(no, newItemID.split( regex: "," )[0]);

        // toplu okuyucu boşalt
        stringBuffer.delete(0, stringBuffer.length());
    }

} catch (IOException e) {
    e.printStackTrace();
}
}

//tv dizilerinin 0.indexlerini sil
tv_isim.remove( index: 0);
tv_imgid.remove( index: 0);

```

Buraya kadar veri kümemizi düzgün bir şekilde aldık..

Bu aldığımız dizimizi bir adapter ile gridview'a bağılayacağız, bu Adapter'in içeriğine bakalım..

```

public class tv_GridAdapter extends BaseAdapter {
    //veri dizileri
    private ArrayList<String> tv_isimleri;
    private ArrayList<String> tv_imgidleri;
    // arayüz ile bağlantı
    private Context context;

    // bağlanacak arayüzün değişkeni
    private LayoutInflater layoutInflater;

    public tv_GridAdapter(Context context,ArrayList<String> tv_imgidleri,ArrayList<String> tv_isimleri){

        // Yarııcı Metod
        this.context=context;
        this.tv_imgidleri=tv_imgidleri;
        this.tv_isimleri=tv_isimleri;
    }
}

```

```

// gridview uzunluğu
@Override
public int getCount() { return tv_isimleri.size(); }

// itemi getir
@Override
public Object getItem(int position) { return tv_isimleri.get(position); }

// item pozisyonu
@Override
public long getItemId(int position) { return position; }

//arayüzü getir..
@Override
public View getView(int position, View convertView, ViewGroup viewGroup) {

    View gridView = convertView; // gridview
    if(convertView== null){ // boş deilse
        //gridview bağlanacak arayüzü bağla
        LayoutInflater inflater = (LayoutInflater) context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
        //gridView hücresi
        gridView = inflater.inflate(R.layout.tv_layout, root: null);

    }
    //gridview içeriği

    ImageView tv_imageview = gridView.findViewById(R.id.element_tv_imageview);
    //TextView tv_textview = gridView.findViewById(R.id.element_tv_textview);
    // sd kart yolu
    String image_path="/storage/ext_sd/VoiceRobot/b_ChanelLogo/x"+tv_imgidleri.get(position).trim()+" .jpg";
    // resim yolunu Uri'ye dönüştür.
    Uri kapak_uri = Uri.parse(image_path);
    // arkaplana at
    tv_imageview.setImageURI(kapak_uri);

    //hazır hale getirdiğim gridView'i geri döndür.

    return gridView;
}

```

```

// ses dönüşü arayüz sorusunu
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    tvActivity.super.onActivityResult(requestCode, resultCode, data);

    switch (requestCode) { // istek kodu
        case REQUEST_CODE_SPEECH_INPUT: { // ses girişi varmı
            if(resultCode == RESULT_OK && data!=null) { // ses düzungün tanımlı ve boş değilse
                //dönen ses
                ArrayList<String> donus = data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
                // intentden dönen veriyi arraylist'e atıyoruz.

                // socket
                if ( bluetoothEkle.btSocket != null) {
                    try {
                        bluetoothEkle.btSocket.getOutputStream().write(donus.get(0).toString().getBytes());
                    } catch (IOException e) {
                        Toast.makeText(getApplicationContext(), text: "Hata !", Toast.LENGTH_SHORT).show();
                    }
                }
                // karsilamaya atmak zorundayım çünkü direkt donuz.get(0) kullanılmıyor garip bir şekilde
                karsilama_textView.setText(donus.get(0).toString());

                //secilen tv ekranını yansıtıyoruz
                SecilenTvEkranı(karsilama_textView.getText().toString());

            }
            break;
        }
    }

    private void SecilenTvEkranı(String _charSequence) {
        // tv ismini açıklıyoruz
        if(_charSequence.indexOf(",") != -1) {
            String[] TvIsimi = _charSequence.toString().split( regex: "\\,");

            // socket
            if (bluetoothEkle.btSocket != null) {
                try {
                    bluetoothEkle.btSocket.getOutputStream().write(TvIsimi[1].toString().substring(0,TvIsimi[1].toString().length()-2).getBytes()); // Parasız geliyor sonda onu engelliyoruz
                } catch (IOException e) {
                    Toast.makeText(getApplicationContext(), text: "Hata !", Toast.LENGTH_SHORT).show();
                }
            }
        }
    }

    private String ilkHarfBuyuk(String str)
    {
        // str Stringinin içindeki kelimelelerin ilk harfleri büyük diğerleri küçük yapılır.
        char c = Character.toUpperCase(str.charAt(0));
        //ilk harfini büyük
        str = c + str.substring(1);
        //buyutulen ilk harften sonra kelimenin diğer harflerini ekledik.
        String bosluk= " ";
        for (int i = 1 ; i<str.length();i++)
        {
            if (str.charAt(i)=='_')
            {
                c=Character.toUpperCase(str.charAt(i+1));
                str = str.substring(0,i)+bosluk+c + str.substring(i+2);
            }
        }
    }

    return str;
}

```

```

//tuşa basıldığında
@Override
public boolean onKeyDown(int keyCode, KeyEvent event) {
    if (keyCode == KeyEvent.KEYCODE_BACK) { // geri tuşu
        this.finish();
    }
    if ((keyCode == KeyEvent.KEYCODE_VOLUME_UP)){ // ses arttırma tuşu
        // cihaza ses artırma komutu gönder
        if (bluetoothEkle.btSocket != null) {
            try {
                bluetoothEkle.btSocket.getOutputStream().write("ses ileri".toString().getBytes()); // Parazit geliyor sonda onu engelliyoruz
            } catch (IOException e) {
                Toast.makeText(getApplicationContext(), text: "Hata !", Toast.LENGTH_SHORT).show();
            }
        }
    }else
    if ((keyCode == KeyEvent.KEYCODE_VOLUME_DOWN)){ // ses azaltma
        //ses azaltma komutu gönder
        if (bluetoothEkle.btSocket != null) {
            try {
                bluetoothEkle.btSocket.getOutputStream().write("ses geri".toString().getBytes()); // Parazit geliyor sonda onu engelliyoruz
            } catch (IOException e) {
                Toast.makeText(getApplicationContext(), text: "Hata !", Toast.LENGTH_SHORT).show();
            }
        }
    }
    return true;
}

```

Bu şekilde kodu gönderiyor ve Ardinuo cihazımızın yorumlamasına bakıyoruz..

Arayüzlerimiz buraya kadar.. Böylelikle projemin sonuna gelmiş bulunuyoruz..

## 1.17. Son Sözler

Ben okul projelerimde ve yine bitirme projemde de değişik birşeyler hep denemek istedim. Gerek başarılı oldum gerek olamadım yada zamanım yetmedi. Yapılmış projeleri herkes yapar. Yani hiç kod bilmeyen bir öğrenci bile bakıp bakıp ne gerekiyorsa yapar. Kodunu anlamasada yazar, maketini yapar. Hatta bu 1 hafta bile sürmez ama ne kadar, ne öğrenir sizin takdirinize bırakıyorum. Ben istedimki kendim birşeyler yapmaya çalışıyorum, yapıları kendim kurayım istedim. Kendi uğraşınca da insanın baya vaktini ve emeğini istiyor. Çok iyi bir bitirme projesi çıkaramamış olabilirim ama iyi bir proje yaptığımı düşünüyorum ve neredeyse her kod parçasına kadar kendim yazdım. Yazdığım platformlarla neredeyse hiç haşır-neşir degildim. Değişik bir şeyler üretmek istiyorsam çok çalışmalıydım ve çok çalıştım. Sürekli bitirme projesi hocamın koordinatörlüğünde hareket ettim. Her zaman daha iyisi olur tabiki zamanım olsa projemi daha da çok geliştirebilirdim.

## **TEŞEKKÜRLER..**

Bitirme projesi koordinatör hocam Öğr. Gör. Şahin AYDIN'a yine bölümümün Program Başkanı Öğr. Gör. Hilal TÜRK'e benimle ilgilendikleri için teşekkür ederim..

## KAYNAKÇA

- [1] <http://stackoverflow.com/>
- [2] <https://www.journaldev.com>
- [3] <http://abhiandroid.com>
- [4] <https://www.youtube.com/>
- [5] <https://sezerduino.com/>
- [6] <http://lezzetlirobottarifleri.com/>