

بنام خدا  
علی فتحی  
گزارش پروژه اول  
موضوع پروژه تایید گوینده با استفاده از رمز از پیش ذخیره شده  
شماره دانشجویی: 9832372

درس گیتهاب پروژه:  
[github.com/alifthi/speakerRecognitionWithLabView](https://github.com/alifthi/speakerRecognitionWithLabView)

مدل مورد استفاده:

<https://ssd.mathworks.com/supportfiles/audio/speakerRecognition/ive-c-english-16kHz.zip>

## هدف پروژه:

در این پروژه قصد داریم با مقایسه صدای فرد گوینده با یک رمز از پیش تعیین شده تشخیص دهیم آیا صدای ضبط شده همان مرتبط با یک فرد از پیش تعیین شده است یا خیر.

## شرح روش انجام پروژه:

جهت مقایسه صدای 2 فرد به روش های متعددی میتوان عمل کرد یکی از این روش ها استفاده از روش های هوش مصنوعی است که در این پروژه از این روش استفاده شده است. در هوش مصنوعی برای رسیدن به این هدف نیز روشهای متعددی وجود دارد.

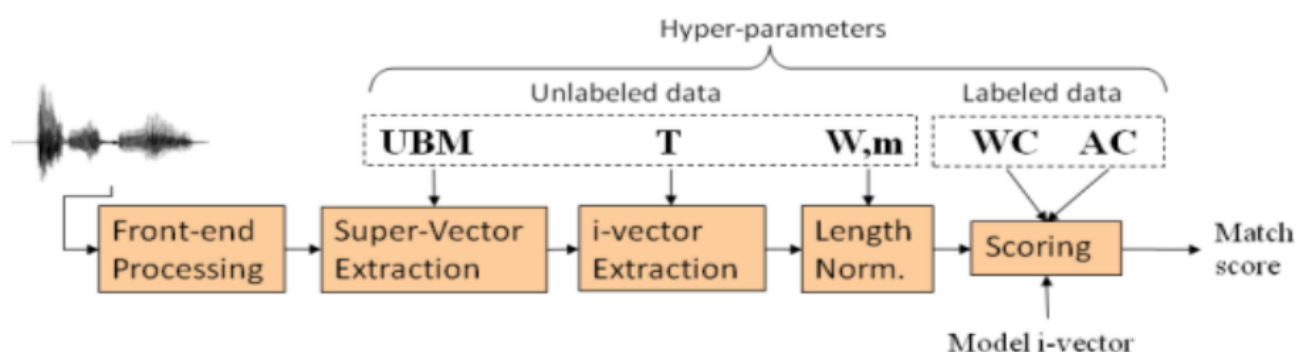
به عنوان مثال میتوان یک شبکه عصبی را با استفاده از صدای فردی که قرار است رمز قرار دهد بارها و با حالت های مختلف آموزش دهیم و در واقع یک تسک classification را انجام دهیم که این روش غیر منطقی بنظر میرسد زیرا ممکن است به تعداد مناسب داده جهت آموزش به یک شبکه دسترسی نداشته باشیم و از طرفی جمع اوری این حجم از داده بسیار مشکل است.

روشی دیگر که منجر به حل این مسئله میشود آن است که ما بتوانیم دو شبکه موازی را آموزش دهیم به گونه ای که بتواند ویژگی های یک صوت را به گونه این استخراج کند که دو فاصله بین دو صوت مشابه را کم و فاصله بین دو صوت غیر مشابه را زیاد کند که توضیح این روش از حوصله موضوع مورد بحث خارج است.

روش دیگر آن است که بتوانیم ویژگی های یک صدا را استخراج کنیم و در نهایت این دو ویژگی را بایک دیگر مقایسه کند. در این پروژه از این روش استفاده کرده ام و با استفاده از متلب برادر های ویژگی ivec را استخراج کرده و با پسورد ورودی توسط کاربر مقایسه شده است. در صورتی که دو تشابه داشته باشند پسورد ورودی توسط کاربر مورد قبول قرار خواهد گرفت.

جهت بدست آوردن score متد های متعددی وجود دارد مانند انواع norm ها (اقلیدوسی، منهتن، بینهایت و...) و یا میتوان از این تشابه را بر اساس کسینوس زاویه بین دو بردار قرارداد کرد. دیاگرام ساده شده این مدل در تصویر 1 رسم شده است.

تصویر 1

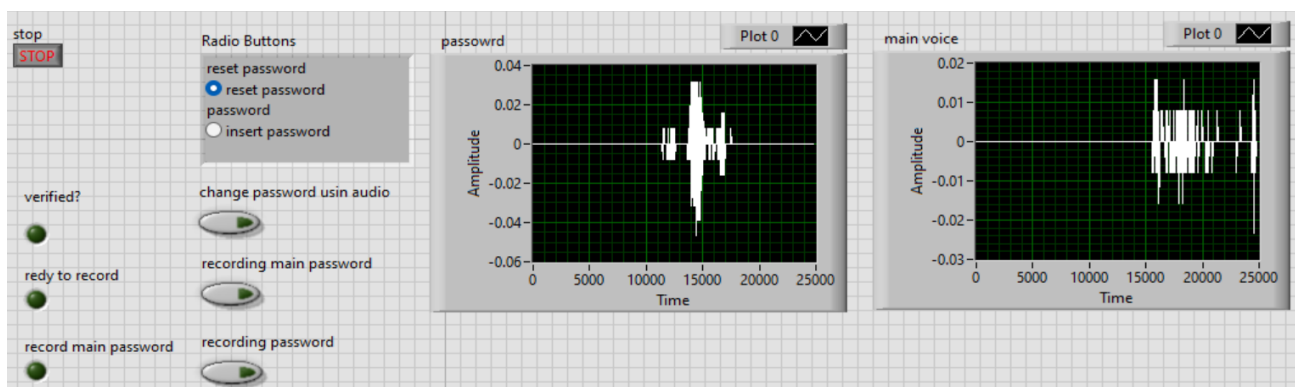


تمامی عملیات های فوق با استفاده از یک دستور متلب انجام خواهد شد که در ادامه ان را بررسی خواهیم کرد.

## شرح Front Panel:

در این پروژه کدهای اصلی پروژه با استفاده از متلب زده شده است و محیط گرافیکی و UI توسط لب و یو توسعه داده شده است. در تصویر 2، front panel برنامه نشان داده شده است که هر کدام از المانهای این برنامه را بررسی خواهیم کرد.

تصویر 2



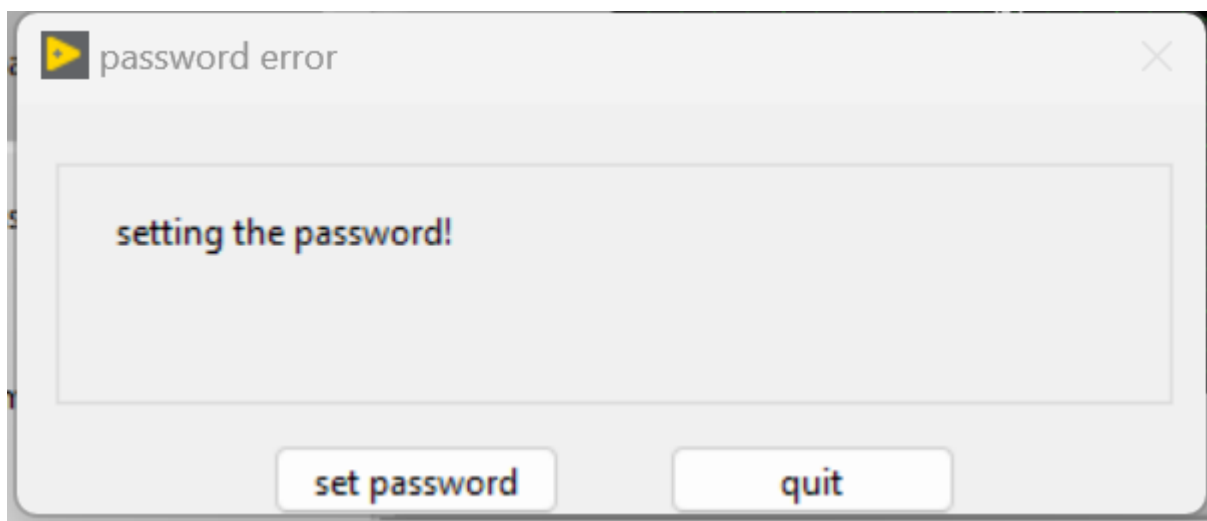
در front panel تعداد 6 المان وجود دارد:

- 3 دکمه جهت ضبط صدا
  - Change password using audio
  - Record main password
  - Record password
- 3 نمایشگر باینری جهت نشان دادن وضعیت
  - Verified
  - Redy to record
  - Record main password
- یک دکمه stop جهت اتمام پروسه اجرای برنامه
- یک radio buttons جهت انتخاب مود شروع برنامه
- 2 نمایشگر جهت نمایش صدای اصلی و صدای ضبط شده وجود دارد

## شرح فرایند اجرا برنامه:

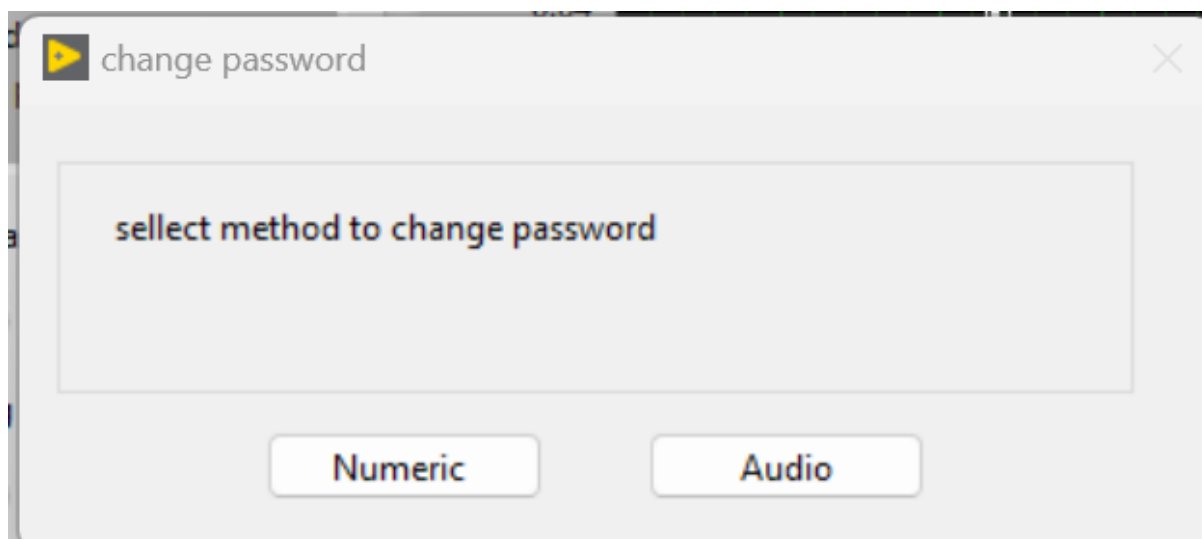
برنامه در دو مود reset password و insert password اجرا میشود. در مود اول برنامه ابتدا رمز از پیش تعیین شده را تغییر خواهد داد و در مود دوم کاربر مستقیماً رمز را وارد میکند. کاربر در مود اول میتواند با دو روش وارد کردن رمز عددی یا وارد کردن رمز صوتی، رمز صوتی اصلی را تغییر دهد. در صورت اجرای برنامه در این مود یک دیالوگ بصورت زیر باز خواهد شد که کاربر میتواند برنامه را ببند یا رمز را ست کند.

تصویر 3



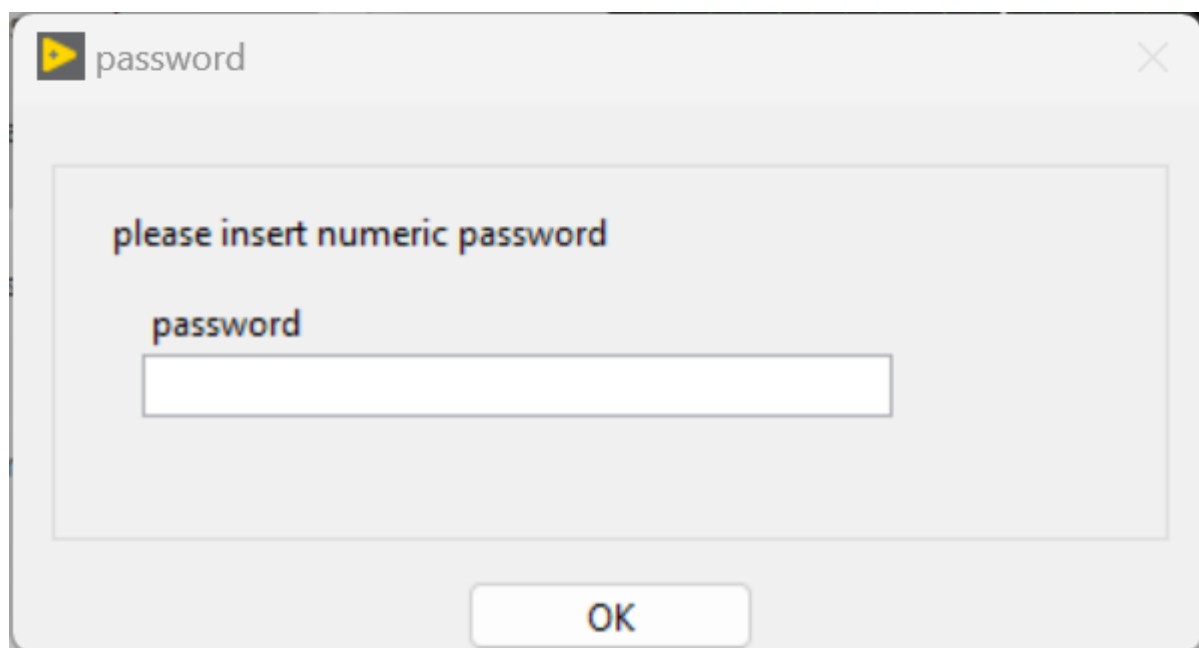
در صورت انتخاب گزینه set password یک دیالوگ دیگر نمایش داده خواهد شد. در غیر اینصورت کاربر میتواند اجرا را تمام کند

تصویر 4



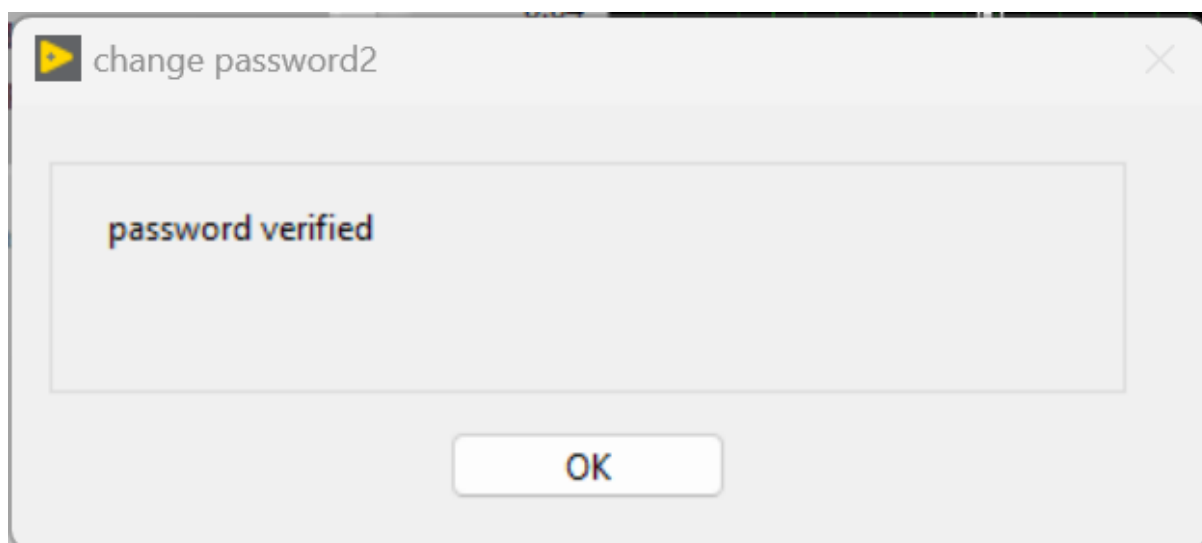
کاربر میتواند رمز خود را به 2 صورت صوتی یا عددی وارد کند. در صورتی که کاربر گزینه numeric را انتخاب کند دیالوگی دیگر نمایش داده میشود که کاربر میتواند رمز عددی را وارد کند.

تصویر 5



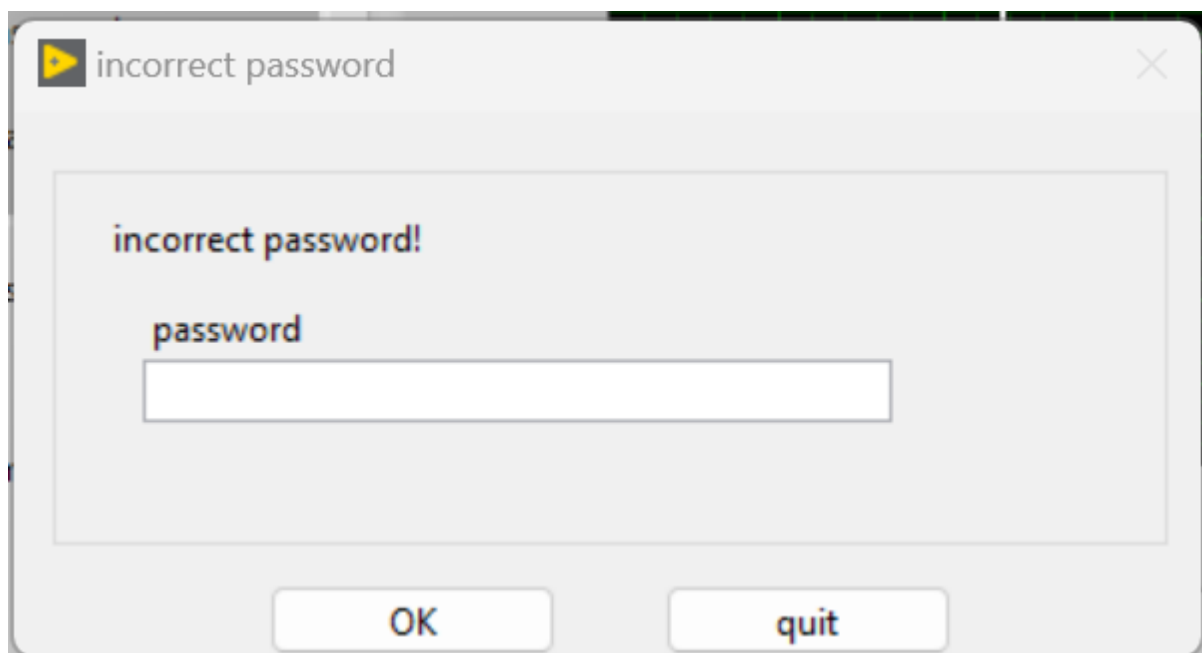
در صورتی که کاربر رمز را صحیح وارد کرده باشد، دیالوگ زیر نمایش داده میشود، نمایشگر record main audio روشن میشود و کاربر میتواند با استفاده از دکمه record main password رمز صوتی جدید را ضبط کند.

تصویر 6



در صورتی که رمز عددی وارد شده توسط کاربر صحیح نباشد دیالوگ زیر نمایش داده میشود که کاربر میتواند برنامه را ببندد یا دوباره رمز را وارد کند.

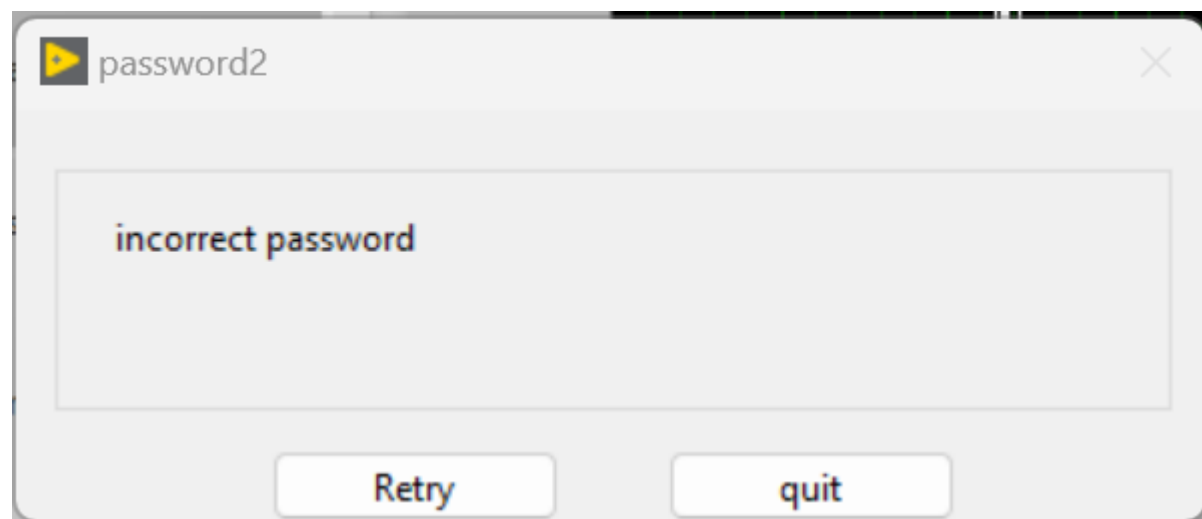
تصویر 7



کاربر در صورت انتخاب گزینه تغییر رمز با استفاده از رمز صوتی، میتواند با استفاده از دکمه Change password using audio رمز اصلی مطابقت داشت میتواند رمز اصلی را با استفاده از دکمه Record main password ضبط و تغییر دهد.

در صورتی که کاربر رمز صوتی را اشتباه وارد کند یک دیالوگ نمایش داده میشود که کاربر میتواند دوباره امتحان کند یا برنامه را ترک کند.

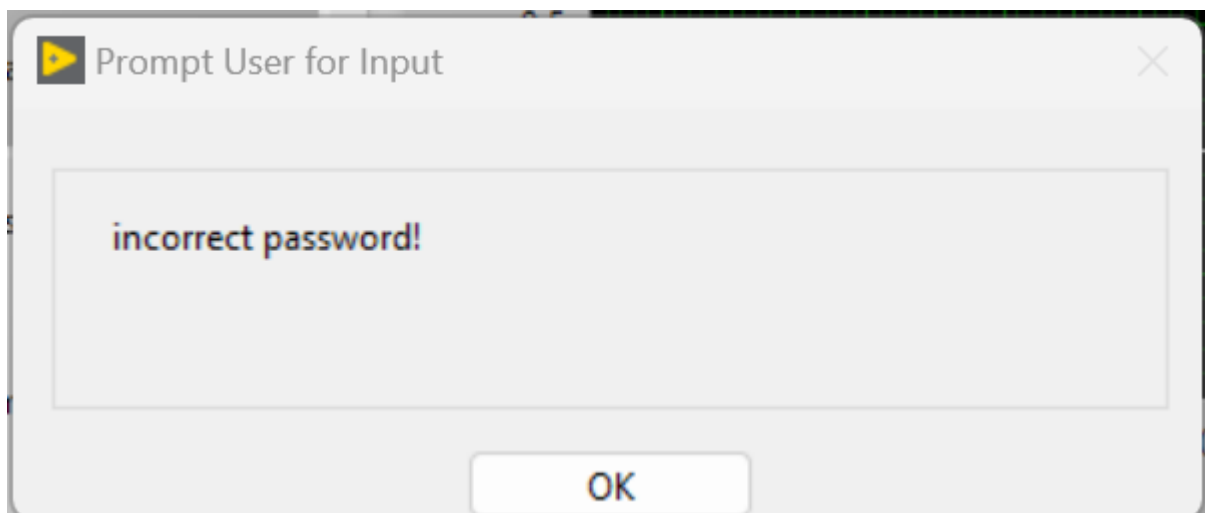
تصویر 8



در پوشه password با نام password.wav ذخیره شده است کاربر در صورتی که رمز صوتی یا عددی را صحیح وارد کرده باشد برنامه این فایل را با فایل ضبط شده توسط کاربر جایگزاری میکند. رمز عددی نیز در پوشه پس ورد و در فایل password.txt ذخیره شده است. برنامه ابتدا رمز درون این فایل را میخواند و با رمز ورودی کاربر مقایسه میکند. حال در صورتی که کاربر برنامه را در مود دوم یعنی insert password اجرا کند برنامه فایل مودل ivec را لود میکند سپس پس ورد اصلی را نیز لود میکند سپس پس ورد اصلی را در مودل enroll میکند.

پس از انجام این پروسه نمایشگر redy to record روشن میشود و از آن پس کاربر میتواند با ضبط صدای خود توسط دکمه record password تازمانی که دکمه فشرده شده است صدای خود را ضبط کند سپس این صدای ضبط شده نیز به مدل ivec داده میشود و پس از آن این دو خروجی باهم مقایسه میشوند در صورت verify شدن صوت نمایشگر verify روشن میشود و در صورتی که صوت تایید نشود دیالوگ زیر برای کاربر نشان داده خواهد شد. در هر دو صورت صوت های ضبط شده نمایش داده میشود و کاربر به ادامه برنامه برمیگردد و میتواند رمز را دوباره ضبط کند.

#### تصویر 9



این فرایند تا زمانی ادامه دارد که کاربر دکمه stop را فشار دهد و برنامه بسته شود.



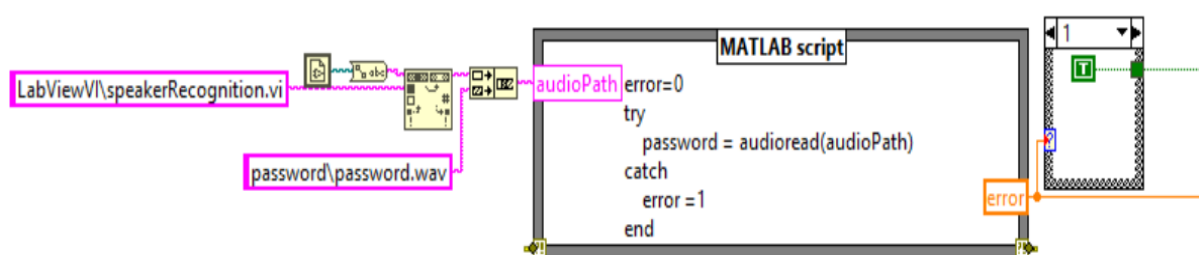
## شرح کدها و labview block diagram

در ابتدا پس از شروع برنامه همه المانها به حالت پیش فرض تنظیم میشوند و سپس برنامه فریم ها را اجرا میکند.

### فریم اول:

در این فریم یک بوک تصمیم گیری وجود دارد که بسته به موقعیت radio button سه تصمیم گیری انجام میدهد در صورتی که کاربر قصد داشتن رمز خود را وارد کند، در فریم اول با استفاده از کد زیر رمز صوتی اصلی لود میشود و مستقیم به فریم چهارم میرود.

تصویر 10



در صورتی که کاربر در مود ضبط مجدد رمز برنامه را اجرا کند یا فایل رمز اصلی وجود نداشته باشد برنامه به فریم دوم میرود.

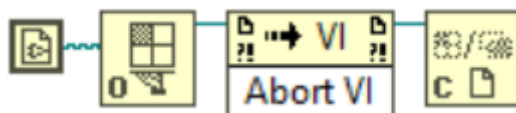
در حال سوم که حالت پیش فرض است تا زمانی که radio button در حال off باشد برنامه هیچ کاری انجام نمیدهد.

تصمیم گیری برای آنکه برنامه به کدام فریم برود توسط یک پرچم صورت میگیرد. این متغیر در بلوک های case structure ست میشود.

### فریم دوم:

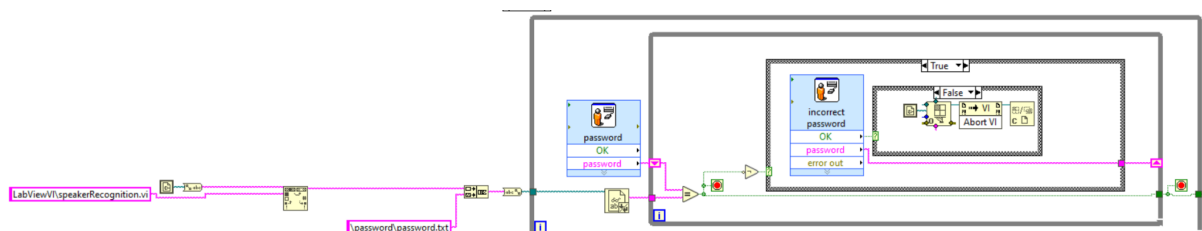
در صورتی که کاربر در مود ورود رمز وارد شده باشد این بلاک هیچ کاری انجام نمیدهد. در صورتی که کاربر در مود دوم وارد شده باشد یا آنکه فایل اصلی رمز وجود نداشته باشد، بلاک مربوط به دیالوگ تصویر 3 اجرا میشود و در صورتی که کاربر خروج را انتخاب کرده باشد توسط بلوک های تصویر 11 برنامه بسته میشود.

تصویر 11



در صورتی که کاربر گزینه set password را انتخاب کرده باشد وارد شرط داخلی می‌شویم.  
در این شرط ابتدا بلوک مربوط به دیالوگ شماره 5 اجرا می‌شود. در صورتی که کاربر گزینه رمز عددی را وارد کند یا آن که از قبل فایل صوتی به عنوان رمز ذخیره نشده باشد بلوک های زیر اجرا می‌شود.

## تصویر 12

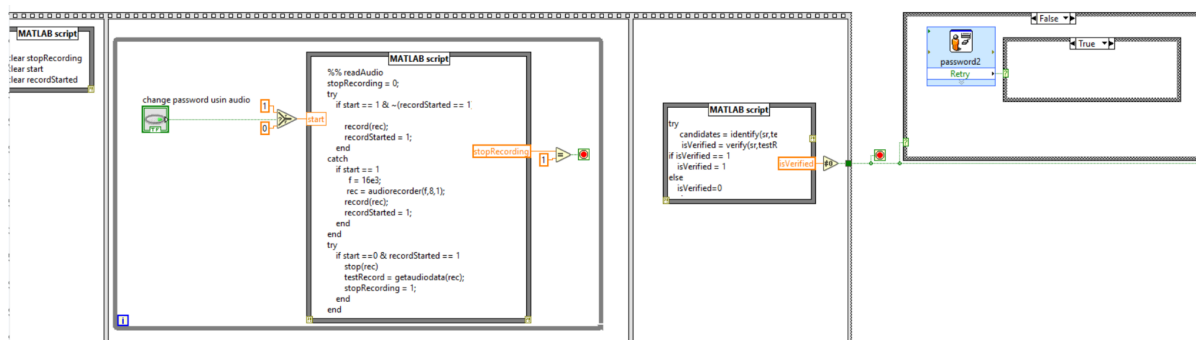


که در این بلوک برنامه ابتدا از فایل password.txt رمز عددی را میخواند(این رمز 123456 است)، سپس بلوک مربوط به دیالوگ تصویر 5 اجرا میشود در صورتی که کاربر رمز را درست وارد کند از فریم خارج میشویم و وارد فریم بعد میشویم.

اما در صورتی که کاربر برای بار اول رمز را درست وارد نکند برنامه وارد یک حلقه میشود که در آن بلوک دیالوگ مربوط به تصویر 7 وجود دارد این حلقه دارای یک shift register میباشد که در هر دور از حلقه مقدار رمز عددی وارد شده توسط با رمز اصلی مقایسه میشود و اگر کاربر گزینه quite را بزند بلوک تصویر شماره 11 اجرا میشود.

حال اگر کاربر قصد داشته باشد با استفاده از رمز صوتی رمز اصلی را تغییر دهد برنامه ابتدا فایل اصلی را لود میکند و در مدل ivec وارد میکند، نمایشگر redy to record روشن میشود و سپس وارد یک لوپ میشود که در آن کاربر میتواند با استفاده از دکمه change password using audio رمز صوتی خود را وارد کند برنامه رمز صوتی و رمز از پیش تعیین شده را هم اندازه میکند و سپس با رمز وارد شده را با رمز از پیش ذخیره شده مقایسه میکند. در صورتی که رمز صوتی درست وارد شده باشد برنامه وار فریم بعد میشود در غیر این صورت بلوک مربوط به تصویر 8 اجرا شده و کاربر میتواند رمز را دوباره وارد کند یا از برنامه خارج شود.

### تصویر 13

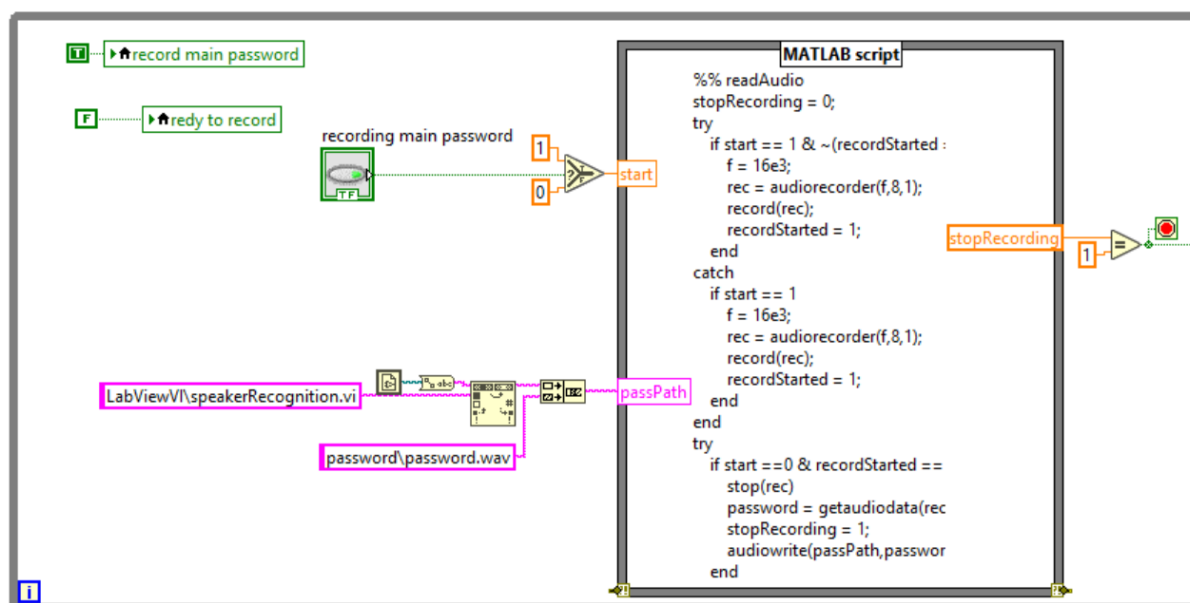


اگر از قبل هیچ فایل صوتی در پوشه password وجود نداشت تنها گزینه کاربر جهت تغییر رمز صوتی استفاده از رمز عددی می باشد.

### فریم سوم:

حال در صورتی که کاربر رمز عددی یا صوتی را درست وارد کرده باشد، کاربر میتواند رمز صوتی را با استفاده از دکمه record main password تغییر دهد. در این مرحله برنامه ابتدا بلاک مربوط به دیالوگ تصویر 6 را اجرا میکند، نمایشگر ready to record را خاموش میکند، نمایشگر record main password را روشن میکند و پس از آن بلوک زیر اجرا میشود که در آن متلب یک صوت را تازمانی که دکمه فشرده شده است ضبط میکند و زمانی که دکمه رها شود آن فایل را ذخیره میکند در پوشه password.

## تصویر 14



اما در صوتی که کاربر در مود ورود رمز برنامه را اجرا کرده باشد این فریم کاری انجام نخواهد داد.

نکته قابل توجه آن است که در کد لب ویو سه متغیر stop, stopRecording, و recordStarted مربوط به ضبط صدا میباشد و در هر دور ریکورد صدا پاک میشود.

### فریم چهارم:

در این فریم مدل ivec لود میشود و password صورتی لود شده را در آن به عنوان پس ورد اصلی enroll میکند.

### فریم پنجم:

در این فریم نمایش گر record main password خاموش میشود و متغیرهای ضبط صدا پاک میشود جهت آماده کردن برای ضبط صدای جدید.

### فریم ششم:

در این فریم ابتدا برنامه وارد یک حلقه بینهایت میشود تا زمانی که کاربر دکمه stop را نزند این حلقه ادامه دار میشود سپس نمایشگر redy to recoord روشن میشود و برنامه تا زمانی که دکمه record password فشرده شود آن را صدا را ضبط میکند. اما اگر کاربر دکمه stop را فشار دهد برنامه کر تصویر 11 را اجرا میکند و برنامه را میندند.

پس از آن نمایشگر redy to record خاموش شده، برنامه طول صدای ضبط شده و طول سیگنال اصلی را هم اندازه میکند و سپس رمز اصلی را در مدل enroll میکند و سیگنالها را نمایش میدهد.

پس از آن اگر برنامه صدای ضبط شده را به مدل می‌دهد و در صورتی که صدای ضبط شده با رمز مطابقت داشت نمایشگر verified را روشن می‌کند و در غیر این صورت بلاک مربوط به تصویر 9 را اجرا می‌کند و پس از آن نیز متغیرهای مربوط به ضبط صدا را جهت آماده سازی برای ضبط بعد آماده می‌کند.

### شرح کدها:

#### کد فریم اول:

```
error=0
try
    password = audioread(audioPath)
catch
    error =1
end
```

کد فوق رمز صوتی را لود می‌کند در صورتی که فایل وجود نداشت متغیر error را 1 می‌کند.

#### کد فریم دوم:

```
fname = 'ivec-english-16kHz.zip';
zipFile = fullfile(modelPath,fname);
unzip(zipFile,modelPath);
%% enroll new record
addpath(tempdir)
sr = speakerRecognition;
enrollRecord = recData;
label = "Ali";
```

این کد در ابتدا مدل را لود می‌کند و متغیرهای مورد نیاز را مقدار دهی می‌کند.

مابقی کدهای استفاده شده مشابه کدهای فریم های بعد است که در ادامه توضیح خواهم داد

کد فریم سوم:

```
%% readAudio
stopRecording = 0;
try
    if start == 1 & ~(recordStarted == 1)

        record(rec);
        recordStarted = 1;
    end
catch
    if start == 1
        f = 16e3;
        rec = audiorecorder(f,8,1);
        record(rec);
        recordStarted = 1;
    end
end
try
    if start == 0 & recordStarted == 1
        stop(rec)
        testRecord = getaudiodata(rec);
        stopRecording = 1;
    end
end
```

کد های فوق تازمانی که متغیر start مقدار 1 دارد رکورد صدا را ادامه میدهد.

#### کد فریم چهارم:

```
recDataLen = length(password)
testRecordLen = length(testRecord)
minim = min(testRecordLen,recDataLen)
tmp = password(1:minim - 100)
testRecord = testRecord(1:minim-100)
enroll(sr,tmp,label)
```

کدهای فوق مقدار دوسیگنال را هم اندازه و برابر با کوتاه تری طول میکند و سپس در مدل enroll میکند.

#### کد فریم ششم:

در این فریم ابتدا از کد فریم سوم وچهرم نیز استفاده شده است که پیش از این توضیح داده شد.

```
try
    candidates = identify(sr,testRecord);
    isVerified = verify(sr,testRecord,label);
if isVerified == 1
    isVerified = 1
else
    isVerified=0
end
catch
    isVerified = 0
end
```

در صورتی که رمز وارد شده با رمز enroll شده یکسان بود کد فوق متغیر isVerified را برابر 1 میکند. و در صورتی که دو صورت ورودی با یکدیگر مطابقت نداشتند یا آن که مدل تشخیص دهد که ورودی صحبت نبوده(مثلا نویز بوده است) این متغیر برابر 0 میشود. از این کد در فریم سوم جهت عوض کردن رمز صوتی نیز استفاده شده است.

## **شرح فایل های پیوست پروژه:**

در این پروژه یکسری پیوست قرار داده شده است که در ادامه توضیح خواهم داد.

### **:LabViewVI**

در این فایل VI اصلی پروژه قرار داده شده است که برنامه را از این محل باز میکنیم.

### **:MatlabCode**

در این پوشه کدهای متلب استفاده شده برنامه قرار داده شده است. برنامه اصلی این کدها را لود نمیکند و صرفا کدها به صورت مجزا باعنوان پیوست پروژه قرار داده شده است.

### **:Password**

در این پوشه رمزهای عددی و صوتی قرار داده شده است. لب ویو مستقیما این فایلها را لود میکند و استفاده میکند.

### **:Model**

در این پوشه مدل مورد استفاده در پروژه قرار داده شده است و لبویو مستقیما از این پوشه مدل را لود میکند.

### **نکته:**

در ادرس ها در پروژه به صورت نسبی قرار داده شده است در نتیجه ادرسها در پروژه نیاز به تغییر ندارند و فقط باید موقعیت نسبتي فایلها نسبت به فایل VI پروژه حفظ شود. در صورت دائلود مدل به صورت دستی لطفا فایل زیپ شده مدل را در پوشه مدل قرار دهید.