TEKNOFEST

HAVACILIK, UZAY VE TEKNOLOJİ FESTİVALİ

ENGELSİZ YAŞAM TEKNOLOJİLERİ YARIŞMASI

PROJE DETAY RAPORU

PROJE ADI: SES KONTROLLÜ HESAP MAKİNASI ANDROİD UYGULAMASI

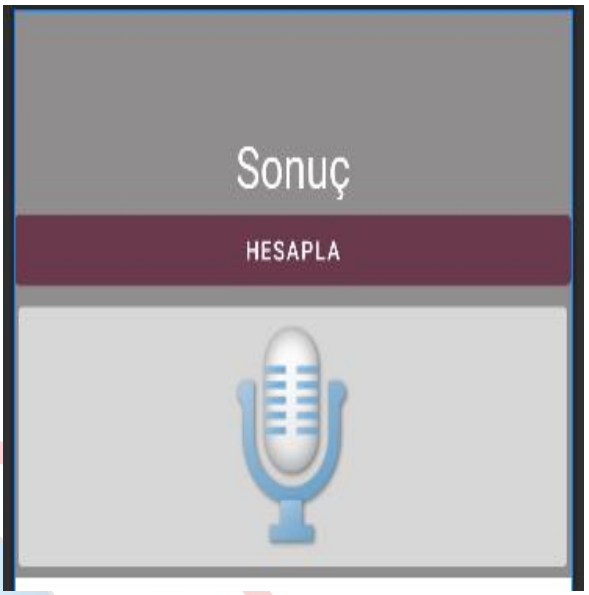
TAKIM ADI: MNV TEKNO

Başvuru ID: 320098

TAKIM SEVİYESİ: İlkokul-Ortaokul / Lise / Üniversite-Mezun

İÇİNDEKİLER

1. Proje Özeti (Proje Tanımı)



Günlük hayatımızın içinde sayısal hesaplama işlemlerinde halkın büyük bir kısmının zorluk çektiği bir gerçektir. Böylesi durumda normal olarak hesap makinası kullanarak işlemin doğruluğunu kontrol edebilirsiniz. Peki, görme engeliniz olduğunu düşünürseniz ne yapabilirsiniz? Biz de bu sorunun önüne geçebilmek için insanların ihtiyaç duyduğunda hızlı ve kolay bir şekilde sayısal işlemlerde sonuca ulaşması için Ses Kontrollü Hesap Makinası android uygulaması geliştirmeye karar verdik.

Projemizde tasarıma projede kullanım kolaylığı sağlamak için sade bir yapı tercih edilmiştir.

Yazılım için android programlamada en çok tercih edilen Android Studio programı kullanılmıştır. Geliştirmiş olduğumuz uygulamayı benzerlerinden ayıran özelliği birçok sayı ve matematiksel işlemi tek seferde hesaplayabilmesidir. Benzer uygulamalar da iki sayı ve tek bir işlem yapılabilmektedir.

Geliştirdiğimiz android uygulama sayesinde hesaplanacak sayısal ifade sesle girilip sonuç tekrar sesli olarak geri bildirilmektedir.

Bu android uygulama ile hesaplama işlemlerinde sorun yaşayan görme engeliler ve sayısal işlemlerde zorluk yaşayan kişilere destek olmak istenmektedir.

Bu projemizle alışveriş sırasında görme engellilere hesap yapmada kolaylık sağladığı görülmüştür. Para üstü alırken veya hesap öderken kolay ve hızlı bir şekilde sonuca ulaşmak mümkün olmaktadır. Projemiz yeniliklere ve geliştirmelere acıktır.

2. Problem/Sorun:

Günlük hayatımızın içinde sayısal hesaplama işlemlerinde halkın büyük bir kısmının zorluk çektiği bir gerçektir.

Bunun sebebi;

Eğitim düzeyinin yetersiz olması,
Kişiden kişiye sayısal işlemlerdeki becerinin değişken olması,
Zihinsel hesaplama işleminde yapılabilecek basit hatalar,
Fiziksel bir engelimizin olması olabilir.

Geliştirmiş olduğumuz uygulamayı benzerlerinden ayıran özelliği birçok sayı ve matematiksel işlemi tek seferde hesaplayabilmesidir. Benzer uygulamalar da iki sayı ve tek bir işlem yapılabilmektedir.

Bu sorunun önüne geçebilmek için insanların ihtiyaç duyduğunda hızlı ve kolay bir şekilde sayısal işlemlerde sonuca ulaşması için Sesli Hesap Makinası android uygulaması geliştirilmiştir.

Bu amaçla tasarladığımız Ses Kontrollü Hesap Makinası android uygulamasının denemeleri yapılarak olumlu sonuçlar alınmıştır.

Google Voice özelliği kullandığı için internet olmadan çalışmamaktadır. Sesli ifade girilirken tane tane ve arada bekleme yapmadan tek seferde söylenmeye dikkat edilmesi gerekir.

3. Çözüm

Projemizde görme engellilerin matematiksel hesaplama işlemlerini daha hızlı bir şekilde yapabilmeleri için sesle bilgi girişi yapan ve bunu metin formatında matematiksel ifadeye çevirdikten sonra hesaplama işlemi yapılarak sonuç sesli olarak bildirilir. Genelde görme engelliler telefon kullanırken ekrana dokunarak hızlı geçişler yaparak kullanabiliyorlar. Her bir karakter girişi için seçim yapmaları gerekmektedir. Bizde buna çözüm olarak sesli giriş yapılan bir hesap makinası geliştirdik.

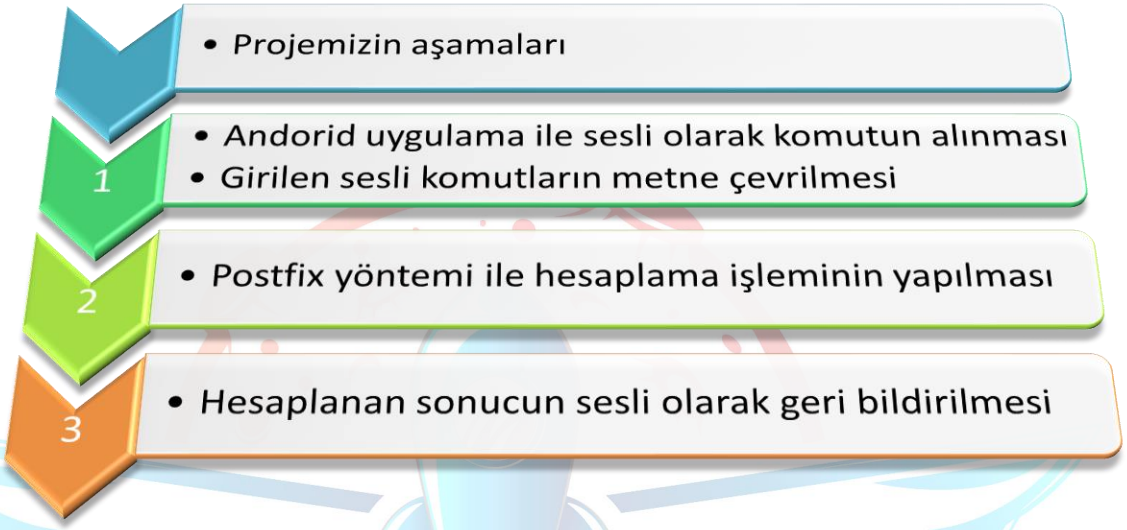
Projemizde android uygulama içerisinde sesli komut girişi yapabilmek için uygulamada ortadaki mikrofon simgesine basılması gerekir. Ses girişi bittiğinde veya tekrar mikrofona basıldığında ses girişi sonlandırılır. Sonuç hesaplanarak sesli olarak bildirilir.

4. Yöntem

Öncelikle problemler doğrultusunda çözüm önerileri oluşturuldu. Çözüm önerilerinin özgünlüğü için literatür taraması yapıldı. Etkinliği konusunda uzman görüşleri ve kaynak taraması yaparak çözüm elemesi yapıldı. Projenin başlangıç aşamasında beyin fırtınası yöntemi kullanılarak projenin içeriği hakkında fikirler üretildi. Daha sonra prototip geliştirildi. Literatür taramasında benzer uygulamalar incelendi eksik ve iyi yönleri not edildi. Yapılan araştırmada mevcut uygulamaların sesli komut vererek tek bir işlem yapabildiği görüldü. Daha sonraki aşamada proje ekibine uygulama yaptırılarak android program yazmak için kullanılan Android Studio programını kullanma becerileri geliştirildi. Yapmış olduğumuz andorid uygulamada işlem kapasitesi geliştirilerek sınırsız hale getirildi.

Uygulamada işlem yapılabilmesi için sesli komutlar tanımlandı. Örneğin; parantez, kapa parantez, üssü , çarpı, bölü, eksi, artı ve sayılar eklendi. Hesaplama işleminde stack(yığın)

yapıları kullanılarak işlem öncelik sırasına göre hesaplama yapması sağlandı. Son olarak uygulamanın tasarımı ve programlama aşamalarında çıkan sorunlar, problem çözme yöntemi ile çözüldü. Projenin test aşamasında yine problem çözme metodu ile oluşan problemler giderildi.



1. AŞAMA-MATEMATİKSEL İFADENİN SESLE ALINMASI

Eclipse programı içindeki "android.speech.RecognizerIntent" özelliği kullanılarak ses alma ve sesli geri bildirim sağlanabilmektedir. Fakat bu özelliği kullanılması için internet olması gerekir. Bu özelliğin kullanılmasına örnek program kodu aşağıdaki gibidir.

```
private void promptSpeechInput() {
    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());
    intent.putExtra(RecognizerIntent.EXTRA_PROMPT,
            getString(R.string.speech_prompt));
    try {
        startActivityForResult(intent, REQ_CODE_SPEECH_INPUT);
    } catch (ActivityNotFoundException a) {
        Toast.makeText(getApplicationContext(),
                getString(R.string.speech_not_supported),
                Toast.LENGTH_SHORT).show();
    }
}
```

```
/**
 * Girilen ses bilgisinin çözümlenmesi
 * */
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
   super.onActivityResult(requestCode, resultCode, data);

   switch (requestCode) {
   case REQ_CODE_SPEECH_INPUT: {
      if (resultCode == RESULT_OK && null != data) {

         ArrayList<String> result = data
               .getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
         txtSpeechInput.setText(result.get(0));
      }
      break;
   }}}
```

2. AŞAMA: POSTFIX YÖNTEMİ İLE SONUCUN HESAPLANMASI:

Bu kısımda ilk olara yığın(stack) yapısı açıklanacak daha sonra ise infix, postfix ve prefix kavramları açıklanacaktır.

2.1. YIĞIN (STACK)

Eleman ekleme çıkarmaların en üstten (top) yapıldığı veri yapısına yığın (stack) adı verilir. Bir eleman ekleneceğinde yığının en üstüne konulur. Bir eleman çıkarılacağı zaman yığının en üstündeki eleman çıkarılır. Bu eleman da yığıntaki elemanlar içindeki en son eklenen elemandır. Bu nedenle yığınlara LIFO (Last In First Out : Son giren ilk çıkar) listesi de denilir.(1)

2.2 INFIX, POSTFIX, PREFIX

Bu kısımda infix, postfix ve prefix kavramları gösterilecektir.
A+B
operator (işlemci) : +
operands (işlenenler) : A, B
infix gösterim : A+B
prefix gösterim : +AB (benzeri bir gösterim add(A,B) fonksiyonu)
postfix gösterim : AB+

in, pre ve post, operator'ün operand'lara göre yerine karşılık gelir. Infix gösterimde işlemci (+), işlenenlerin (A,B) arasında yer alır. Prefix gösterimde işaretçi işlenenlerden önce gelir, postfix gösterimde de sonra gelir.

A+B*C infix ifadesini postfix'e çevirelim.

Bunun için işlem önceliğine bakmak gerekir. Çarpmanın toplamaya önceliği olduğu için, A+(B*C) şeklinde düşünülebilir. Önce çarpma kısmı postfix'e çevrilecek sonra da sonucu.

A+(B*C) anlaşılırlığı artırmak için parantez kullandık.
A+(BC*) çarpım çevrildi.
A(BC*)+ toplam çevrildi.
ABC*+ postfix form.

İşlem önceliği (büyükten küçüğe)
Üs alma
Çarpma/Bölme
Toplama/Çıkarma
Parantezsiz ve aynı önceliğe sahip işlemcilerde işlemler soldan sağa doğru yapılır (üs alma hariç). Üs almada sağdan sola doğrudur. A-B+C 'de öncelik (A-B)+C şeklindedir. A^B^C'de ise A^(B^C) şeklindedir. Parantezler default öncelikleri belirtmek için konulmuştur.

Infix	**Postfix**	**Prefix**
A+B-C	AB+C-	-+ABC
(A+B)*(C-D)	AB+CD-*	*+AB-CD
A^B*C-D+E/F/(G+H)	AB^C*D-EF/GH+/+	+-*ÂBCD//EF+GH
((A+B)*C-(D-E))^(F+G)	AB+C*DE—FG+^	^-*+ABC-DE+FG
A-B/(C*D^E)	ABCDE^*/-	-A/B*C^DE

Postfix formda parantez kullanımına gerek yoktur. İki infix ifadeyi düşünün : A+(B*C) ve (A+B)*C. İlk ifadede parantez gereksizdir. İkincide ilk ifade ile karıştırılmaması için gereklidir. Postfix forma çevirmek bu karışıklığı önler. (A+B)*(C+D)'yi infix formda parantezsiz ifade etmek için çarpım işlemi yapılırsa işlem sayısı çoğalır.

Infix formdan postfix forma çevrilen bir ifadede operandların (sayı veya sembol) bağlı olduğu operator'leri (+,-,*,/) görmek zorlaşır (3 4 * + ifadesinin sonucunun 23'e, 3 4 + 5 ifadesinin sonucunun 35'e karşılık geldiğini bulmak zor gibi görünür). Fakat parantez kullanmadan tek anlama gelen bir hale dönüşür. İşlemleri, hesaplamaları yapmak kolaylaşır. (1)

postfix ifadenin sonucunun hesaplanması (ve bunu gerçekleştiren algoritma) :

Bir postfix string'inde her operator kendinden önce gelen iki operand üzerinde işlem yapacaktır. Bu operandlar daha önceki operator'lerin sonuçları da olabilir. Bir anda ifadeden

bir operand okunarak yığına yerleştirilir. Bir operator'e ulaşıldığında, yığının en üstündeki iki eleman bu operator'ün operand'ları olacaktır. İki eleman yığıttan çıkarılır işlem yapılır ve sonuç tekrar yığına konulur. Artık bir sonraki operator'ün operand'ı olmaya hazırdır.

Birçok derleyici 3*2+5*6 gibi bir infix ifadenin değerini hesaplayacağı zaman postfix forma dönüştürdükten (belirsizliği ortadan kaldırdıktan sonra) sonucu hesaplar : "3 2 * 5 6 * +"

5. Yenilikçi (İnovatif) Yönü

Yapılan araştırmada mevcut uygulamaların sesli komut vererek tek bir işlem yapabildiği görüldü. Yapmış olduğumuz android uygulamada işlem kapasitesi geliştirilerek sınırsız hale getirildi. Tek seferde aralıksız olarak söylenen matematiksel ifade içindeki dört işlem, üslü işlemler ve parantez içindeki işlemler dahil hepsini hesaplayabilir.

Projede kullanılan android uygulamanın düzgün çalışabilmesi için android cihazın internet bağlantısının olması yeterlidir.

Kodlamada sesli giriş sonrası metne çevrilen matematiksel ifade içindeki sayılar ve özel işaretler ayrı ayrı yığınlara(stack) atılarak ayrıştırılmış olur. Daha sonra özel karakter olan yığın içinde işlem öncelik sırasına göre sayıların olduğu yığındaki ilgili sıradaki sayılar arasında işlem yapılır. Sonuç tekrar alınan sayıların olduğu yığındaki yere girilir.

6. Uygulanabilirlik

Projede yazılan uygulama android marketlere yüklenerek herkesin kullanımına olanak sağlanacaktır. Projenin tam olarak çalışabilmesi için internet bağlantısının olması gerekir.

7. Tahmini Maliyet ve Proje Zaman Planlaması

Maliyet 25$ Amerikan Dolarıdır. Bu ücret sadece Google play'de development hesap açılışı için bir defa mahsus ödenmiştir. Kullanıcılar için uygulama tamamen ücretsizdir.

AYRINTILI ÇALIŞMA TAKVİMİ

Başlıca	**Ayrıntılı Bilgi**	**Zamanlama**
Literatür Araştırması	Proje kapsamında gerçekleştirilecek literatür taramasının 3 Haftalık bir dönemi kapsamıştır.	**Ocak-Şubat (6 Hafta)**
Bilgisayar Yazılımı	Proje kapsamında kullanılacak programlar tespit edilerek internet ortamından üretici firmaların sitesinden temin edildi. (Android Studio, Java SE Development Kit,)	**Mart (3 Hafta)**

Android Uygulamanın Kod yazımı	Anadorid Studio programında android uygulamanın java dili kullanılarak kod yazımı yapıldı.	**Mart-Nisan (5 Hafta)**
Sistemin Çalışması	Sistem çalışır hale geldikten sonra gerekli testler yapıldı.	**Mayıs (4 Hafta)**
Rapor Hazırlama	Gerekli veriler kullanılarak rapor hazırlandı.	**Haziran (2 Hafta)**

8. Proje Fikrinin Hedef Kitlesi (Kullanıcılar):

Projenin hedef kitlesi öncelikli olarak görme engelliler olarak belirlenmiştir. Dolaylı olarak matematiksel işlemlerde zorluk çeken herkese de hitap etmektedir.

9. Riskler

Projenin çalışmasını olumsuz yönde etkileyecek unsurlar internet bağlantısın olmaması ve sesli giriş yapılırken ortam sesinin çok gürültülü olması olarak tespit edilmiştir. Sesli giriş yapılırken anlaşılır ve sade şekilde komut verilmelidir. Yoksa komutlar yanlış çözümlenmektedir. Bu duruma ise sesli komutun tekrar girilmesi istemektedir.

Projenin android markete yükleme sırasında uygulamanın onay alıp yayınlanması biraz zaman alabilir.

Bütçe 25$ Amerikan Dolarıdır. Bu ücret sadece Google play'de development hesap açılışı için ödenmiştir. Uygulama tamamen ücretsizdir.

10. Kaynaklar

1. https://www.javatpoint.com/conversion-of-postfix-to-prefix-expression, 17/05/2022
2. http://www.androidhive.info/2014/07/android-speech-to-text-tutorial,17/05/2022
3. UZ, Hacer Ebru; DEMIRCI, Birim Balcı. Engelli Kullanıcılar İçin Multimedya Öğrenme Araçları.
4. Matos, Victor. "Android Environment Emulator." Cleveland State University, Jul 20 (2009): 11.
5. Meier, Reto. Professional Android 4 application development. John Wiley & Sons, 2012.
6. Gargenta, Marko. Learning Android. " O'Reilly Media, Inc.", 2011.