

ANKARA ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



IOS İle Mobil Uygulama Geliştirme II

Çeviri ve Dil Öğrenme Uygulaması

Özge Çelik

21290514

19.06.2025

<https://github.com/ozgecelk01/CeviriVeDilOgrenmeUygulaması>

https://drive.google.com/file/d/1jrQLPoDL0RtapPEN_p_pDRvFPJd0ctlu/view?usp=drivesdk

Proje Amacı: Kullanıcıların dil öğrenme süreçlerine yardımcı olacak bir mobil uygulama geliştirmek. Uygulama, kullanıcıların kelime veya cümle çevirilerini hızlıca yapabilmesini sağlayacak. Çeviri yapabilme, favori kelimeleri kaydetme ve günlük kelime önerileri gibi özelliklerle kullanıcıların öğrenme süreçlerini daha etkili hale getirmeyi amaçlamaktadır.

Proje Aşamaları

1.Login ve Register Sayfası

Projenin ilk aşamasında, kullanıcıların uygulamaya giriş yapmasını sağlayan bir Login Sayfası oluşturuldu. Kullanıcılar, giriş yaptıktan sonra favori kelimelerini ve geçmiş çevirilerini görüntüleyebilir. Yeni üye olacaklar için bir register sayfası oluşturuldu.

← Login

← Register

Hello

Sign in to your account

Email

Password

Forgot your password?

Sign in →

Create account

Email

Password

Create →

Already have an account? Login

Don't have an account? Register

```
const mongoose = require("mongoose");

const userSchema = new mongoose.Schema({
  name: String,
  email: { type: String, required: true, unique: true },
  password: { type: String, required: true }
});

module.exports = mongoose.model("User", userSchema);
```

```
const handleRegister = async () => {  
  try {  
    const response = await axios.post('http://localhost:5000/api/auth/register', {  
      email,  
      password  
    });  
  }  
};
```

2. Tasarım ve Kullanıcı Arayüzü

Uygulamanın temel ekranlarının tasarımı yapıldı. Kullanıcı dostu ve kolay anlaşılır bir arayüz oluşturmak için React Native Paper ve Styled Components gibi UI kütüphaneleri kullanıldı.

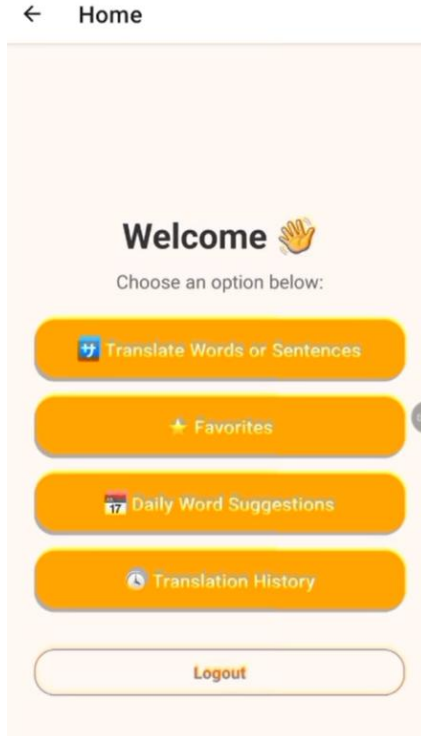
Ana Ekran: Kullanıcıların sayfalar arası seçim yapabileceği ana sayfa.

Çeviri Ekranı: Kullanıcıların kelime veya cümle girerek çeviri yapabileceği ana sayfa.

Favoriler Ekranı: Kullanıcının sık kullandığı veya kaydettiği kelimeleri ve cümleleri görebileceği ekran.

Günlük Kelime Önerileri Ekranı: Her gün belirlenen yeni kelime veya cümle önerilerinin gösterildiği ekran.

Geçmiş Ekranı: Kullanıcının daha önce yaptığı çevirilere erişebileceği ekran.



```

<ScrollView contentContainerStyle={styles.container}>
  <Text style={styles.header}>Welcome 🌟</Text>
  <Text style={styles.subText}>Choose an option below:</Text>
  <Button
    mode="contained"
    style={styles.button}
    labelStyle={styles.buttonText}
    onPress={() => navigation.navigate('Translate')}
  />
  Translate Words or Sentences
  </Button>
  <Button
    mode="contained"
    style={styles.button}
    labelStyle={styles.buttonText}
    onPress={() => navigation.navigate('Favorites')}
  />
  Favorites
  </Button>
  <Button
    mode="contained"
    style={styles.button}
    labelStyle={styles.buttonText}
    onPress={() => navigation.navigate('DailyWords')}
  />
  Daily Word Suggestions
  </Button>

```

3. Navigasyon Yapısının Oluşturulması

Uygulama içindeki sayfalar arasında kolay geçiş sağlamak için React Navigation kütüphanesi kullanıldı. Kullanıcılar, menü veya sekmeler aracılığıyla farklı ekranlara yönlendirilebildi.

```

import React from 'react';
import { NavigationContainer } from '@react-navigation/native';
import { createNativeStackNavigator } from '@react-navigation/native-stack';
import LoginScreen from '../screens/LoginScreen';
import RegisterScreen from '../screens/RegisterScreen';
import HomeScreen from '../screens/HomeScreen';
import TranslateScreen from '../screens/TranslateScreen';
import HistoryScreen from '../screens/HistoryScreen';
import DailyWordsScreen from '../screens/DailyWordsScreen';
import FavoritesScreen from '../screens/FavoritesScreen';

const Stack = createNativeStackNavigator();

export default function AppNavigator() {
  return (
    <NavigationContainer>
      <Stack.Navigator initialRouteName="Login">
        <Stack.Screen name="Login" component={LoginScreen} />
        <Stack.Screen name="Register" component={RegisterScreen} />
        <Stack.Screen name="Home" component={HomeScreen} />
        <Stack.Screen name="Translate" component={TranslateScreen} />
        <Stack.Screen name="History" component={HistoryScreen} />
        <Stack.Screen name="DailyWords" component={DailyWordsScreen} />
        <Stack.Screen name="Favorites" component={FavoritesScreen} />
      </Stack.Navigator>
    </NavigationContainer>
  );
}

```

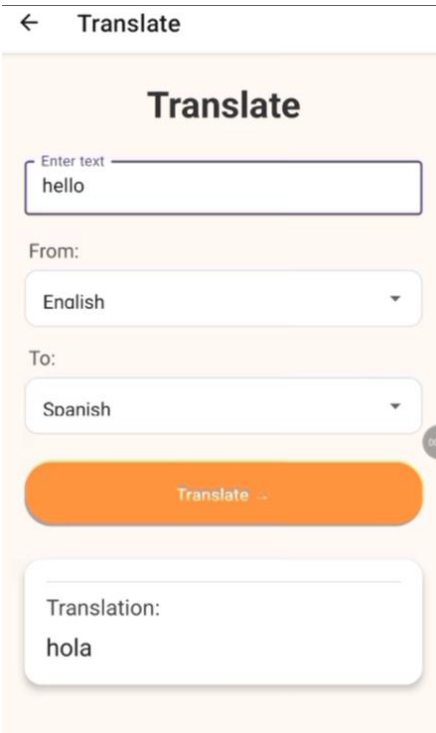
4. Çeviri API'lerinin Entegre Edilmesi

Kullanıcıların kelime ve cümlelerini anında çevirebilmeleri için LibreTranslate API kullanıldı. Kullanıcının girdiği metin API'ye gönderildi. API'den gelen çeviri sonucu kullanıcıya gösterildi.

```
try {
  const response = await axios.post(
    'https://libretranslate.com/translate',
    {
      q: text,
      source: sourceLang,
      target: targetLang,
      format: 'text',
    },
    { headers: { 'Content-Type': 'application/json' } }
  );
  setTranslated(response.data.translatedText);
} catch (error) {
  Alert.alert("Translation Failed", error.message);
} finally {
  setLoading(false);
}
```

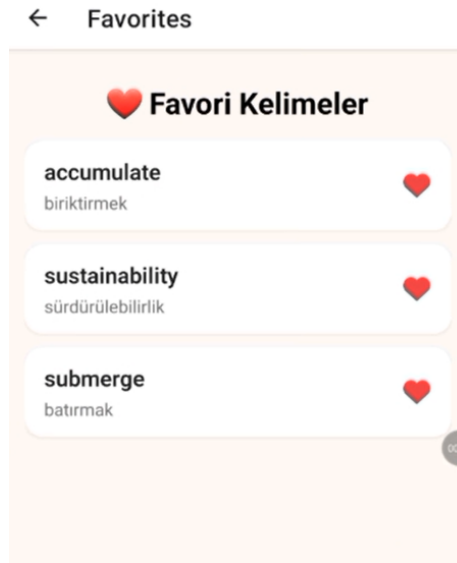
5. Kullanıcı Girdileri ve Çeviri İşlemi

Kullanıcılar, çevirmek istedikleri kelime veya cümleleri giriş alanına yazabilecekler. API'den gelen çeviri anında ekranda gösterilecektir.



6. Favori Kelimeler ve Cümleler

Kullanıcılar, sık kullandıkları kelimeleri veya cümleleri favorilerine ekleyebilecektir. Favorilere eklenen kelimeler ve cümleler veritabanında saklanacaktır. Kullanıcı, daha sonra bu listeyi görüntüleyebilecektir.



```
<View style={styles.container}>
  <Text style={styles.title}>❤️ Favori Kelimeler</Text>
  {favoriteWords.length > 0 ? (
    <FlatList
      data={favoriteWords}
      renderItem={renderItem}
      keyExtractor={(item) => item.id}
      contentContainerStyle={{ paddingBottom: 20 }}
    />
  ) : (
    <Text style={styles.emptyText}>Henüz favori kelimen yok.</Text>
  )}
</View>
```

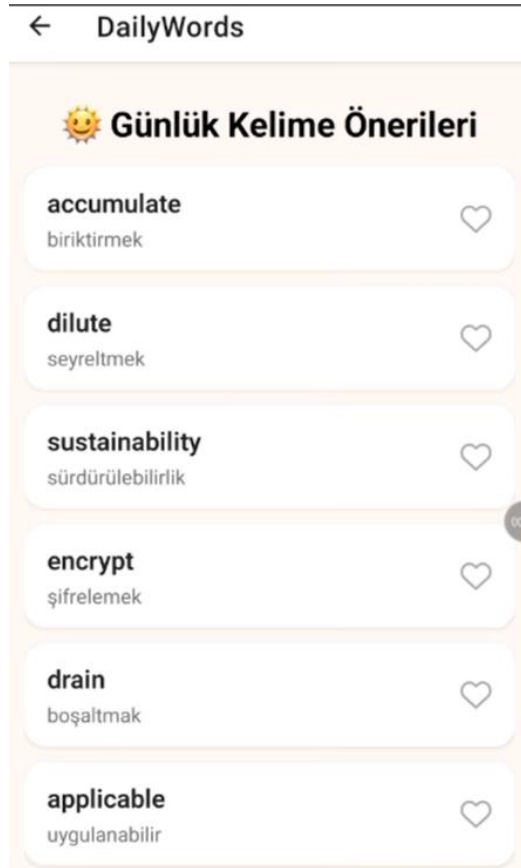
7. Günlük Kelime Önerileri

Uygulama, kullanıcıların dil öğrenme sürecini desteklemek için her gün yeni kelimeler veya cümleler önerecektir. Bu kelimeler ve cümleler, sistem tarafından otomatik olarak belirlenecek ve kullanıcıya gösterilecektir. Kelimeler words.json dosyasından tutulmaktadır.

```
import wordData from '../assets/words.json';

const DailyWordsScreen = () => {
  const [dailyWords, setDailyWords] = useState([]);
  const [favorites, setFavorites] = useState([]);

  const getRandomWords = () => {
    const shuffled = wordData.sort(() => 0.5 - Math.random());
    return shuffled.slice(0, 10);
  };
};
```



8. Geçmiş Sayfası

Kullanıcıların daha önce yaptığı çeviriler veritabanında saklanacaktır.

```
useEffect(() => {  
  const fetchHistory = async () => {  
    const data = await AsyncStorage.getItem('translation_history');  
    if (data) setHistory(JSON.parse(data));  
  };  
  fetchHistory();  
}, []);
```

