



REPUBLIK INDONESIA
KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA

SURAT PENCATATAN CIPTAAN

Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan : EC00202309560, 31 Januari 2023

Pencipta

Nama : **Arbi Haza Nasution, Winda Monika dkk**
Alamat : Jln Banda Aceh Gg Kencana II No 17/21 RT002/RW015 Kel
Tangerang Utara Kec Bukit Raya, Pekanbaru, RIAU, 28289
Kewarganegaraan : Indonesia

Pemegang Hak Cipta

Nama : **Arbi Haza Nasution, Winda Monika dkk**
Alamat : Jln Banda Aceh Gg Kencana II No 17/21 RT002/RW015 Kel
Tangerang Utara Kec Bukit Raya, Pekanbaru, RIAU, 28289
Kewarganegaraan : Indonesia

Jenis Ciptaan : **Program Komputer**
Judul Ciptaan : **Mesin Penerjemah Hibrida Bahasa Daerah Indonesia**

Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia : 31 Januari 2023, di Pekanbaru

Jangka waktu perlindungan : Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.

Nomor pencatatan : 000442483

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.

Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.



a.n Menteri Hukum dan Hak Asasi Manusia
Direktur Jenderal Kekayaan Intelektual
u.b.
Direktur Hak Cipta dan Desain Industri

Anggoro Dasananto
NIP.196412081991031002

Disclaimer:

Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan.

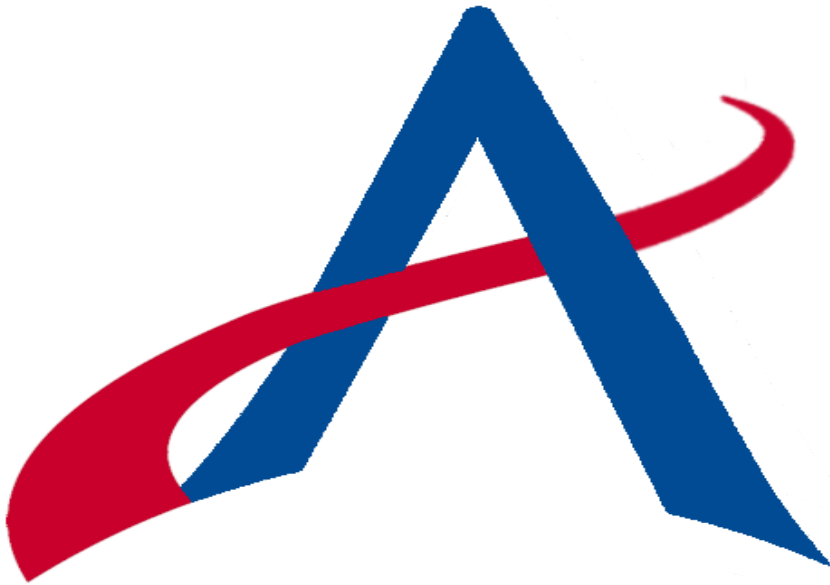
LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Arbi Haza Nasution	Jln Banda Aceh Gg Kencana II No 17/21 RT002/RW015 Kel Tangkerang Utara Kec Bukit Raya
2	Winda Monika	Jln Banda Aceh Gg Kencana II No 17/21 RT002/RW015 Kel Tangkerang Utara Kec Bukit Raya
3	Adam Yahya	Jln Pemuda No. 67 Kecamatan Payung Sekaki Kelurahan Tampan RT03 RW01
4	Wiku Karno Prasetyagama	Jln Pemuda No. 67 Kecamatan Payung Sekaki Kelurahan Tampan RT03 RW01

LAMPIRAN PEMEGANG

No	Nama	Alamat
1	Arbi Haza Nasution	Jln Banda Aceh Gg Kencana II No 17/21 RT002/RW015 Kel Tangkerang Utara Kec Bukit Raya
2	Winda Monika	Jln Banda Aceh Gg Kencana II No 17/21 RT002/RW015 Kel Tangkerang Utara Kec Bukit Raya
3	Adam Yahya	Jln Pemuda No. 67 Kecamatan Payung Sekaki Kelurahan Tampan RT03 RW01
4	Wiku Karno Prasetyagama	Jln Pemuda No. 67 Kecamatan Payung Sekaki Kelurahan Tampan RT03 RW01





Mesin Penerjemah Hibrida Bahasa Daerah Indonesia

Pemegang Hak Cipta :

- 1. Arbi Haza Nasution, Universitas Islam Riau**
- 2. Winda Monika, Universitas Lancang Kuning**
- 3. Adam Yahya, CV. Arti Cerdas Nusantara**
- 4. Wiku Karno Prasetyagama, CV. Arti Cerdas Nusantara**

❖ SOURCE CODE PROGRAM APLIKASI

Source Code Halaman Utama

```
1 import 'dart:async';
2
3 import 'package:flutter/material.dart';
4 import 'package:speech/theme.dart';
5
6 class SplashPage extends StatefulWidget {
7   @override
8   State<SplashPage> createState() => _SplashPageState();
9 }
10
11 class _SplashPageState extends State<SplashPage> {
12   @override
13   void initState() {
14     // TODO: implement initState
15     Timer(
16       Duration(seconds: 3),
17       () => Navigator.pushNamed(context, '/home'),
18     ); // Timer
19     super.initState();
20   }
21
22   @override
23   Widget build(BuildContext context) {
```

```
20 }
21
22 @override
23 Widget build(BuildContext context) {
24   return Scaffold(
25     backgroundColor: backgroundColor1,
26     body: Center(
27       child: Container(
28         width: 120,
29         height: 158,
30         decoration: BoxDecoration(
31           image: DecorationImage(
32             image: AssetImage('assets/img_splash_page.png'),
33           ), // DecorationImage
34         ), // BoxDecoration
35       ), // Container
36     ), // Center
37   ); // Scaffold
38 }
39 }
40
```

Source Code Room Pembicara

```
1 import 'package:avatar_glow/avatar_glow.dart';
2 import 'package:clipboard/clipboard.dart';
3 import 'package:flutter/material.dart';
4 import 'package:speech/api/speech_api.dart';
5 import 'package:speech/main.dart';
6 import 'package:speech/models/speech_model.dart';
7 import 'package:speech/provider/speech_provider.dart';
8 import 'package:speech/theme.dart';
9 import 'package:speech/widget/substring_highlighted.dart';
10
11 import '../utils.dart';
12
13 class SpeechPage extends StatefulWidget {
14   @override
15   _SpeechState createState() => _SpeechState();
16 }
17
18 class _SpeechState extends State<SpeechPage> {
19   TextEditingController textSpeechController = TextEditingController();
20   SpeechProvider speechProvider = SpeechProvider();
21   String text = 'Press the button and start speaking';
22   bool isListening = false;
23   String id = '';
```

```
25   @override
26   Widget build(BuildContext context) {
27     Widget english() {
28       return SingleChildScrollView(
29         reverse: true,
30         padding: const EdgeInsets.all(30).copyWith(bottom: 150),
31         child: Container(
32           height: 150,
33           width: double.infinity,
34           decoration: BoxDecoration(
35             color: backgroundColor3,
36             borderRadius: BorderRadius.circular(20),
37           ), // BoxDecoration
38         child: Column(
39           crossAxisAlignment: CrossAxisAlignment.start,
40           children: [
41             Row(
42               children: [
43                 Padding(padding: EdgeInsets.only(left: 20, top: 30)),
44                 Text('English', style: blackTextStyle.copyWith(fontSize: 16)),
45               ],
46             ), // Row
47             Container(
```



```

95 |     ), // Row
96 |     Container(
97 |       padding: EdgeInsets.only(left: 20, top: 10, right: 20),
98 |       child: Column(
99 |         children: [
100 |           Padding(padding: EdgeInsets.only(left: 100)),
101 |           SubstringHighlight(
102 |             text: '',
103 |             terms: Command.all,
104 |             textStyle: TextStyle(
105 |               fontSize: 20.0,
106 |               color: Colors.black,
107 |               fontWeight: FontWeight.w400,
108 |             ), // TextStyle
109 |             textStyleHighlight: TextStyle(
110 |               fontSize: 20.0,
111 |               color: Colors.red,
112 |               fontWeight: FontWeight.w400,
113 |             ), // TextStyle
114 |           ), // SubstringHighlight
115 |         ],
116 |       ), // Column
117 |     ) // Container

```

```

124 |   Widget cardButton() {
125 |     return AvatarGlow(
126 |       animate: isListening,
127 |       endRadius: 75,
128 |       glowColor: Theme.of(context).primaryColor,
129 |       child: FloatingActionButton(
130 |         child: Icon(isListening ? Icons.mic : Icons.mic_none, size: 36),
131 |         // onPressed: toggleRecording,
132 |         onPressed: () {
133 |           SpeechApi.toggleRecording(
134 |             onResult: (text) => setState() => this.text = text),
135 |             onListening: (isListening) {
136 |               setState() => this.isListening = isListening);
137 |
138 |             if (!isListening) {
139 |               Future.delayed(Duration(seconds: 1), () {
140 |                 Utils.scanText(text);
141 |                 SpeechModel().createSpeech(id, text);
142 |               }); // Future.delayed
143 |             }
144 |           },
145 |         );
146 |     },

```


Source Code inialisasi data ke database

```
1 import 'package:cloud_firestore/cloud_firestore.dart';
2 import 'package:firebase_core/firebase_core.dart';
3
4 class SpeechModel {
5   static final speechCollection = Firebase.initializeApp().then((value) {
6     return FirebaseFirestore.instance.collection('speech');
7   });
8
9   Future<void> createSpeech(String id, String text) async {
10    await speechCollection.then((value) {
11      value.doc(id).set({
12        'text': text,
13      });
14    });
15  }
16 }
17
```

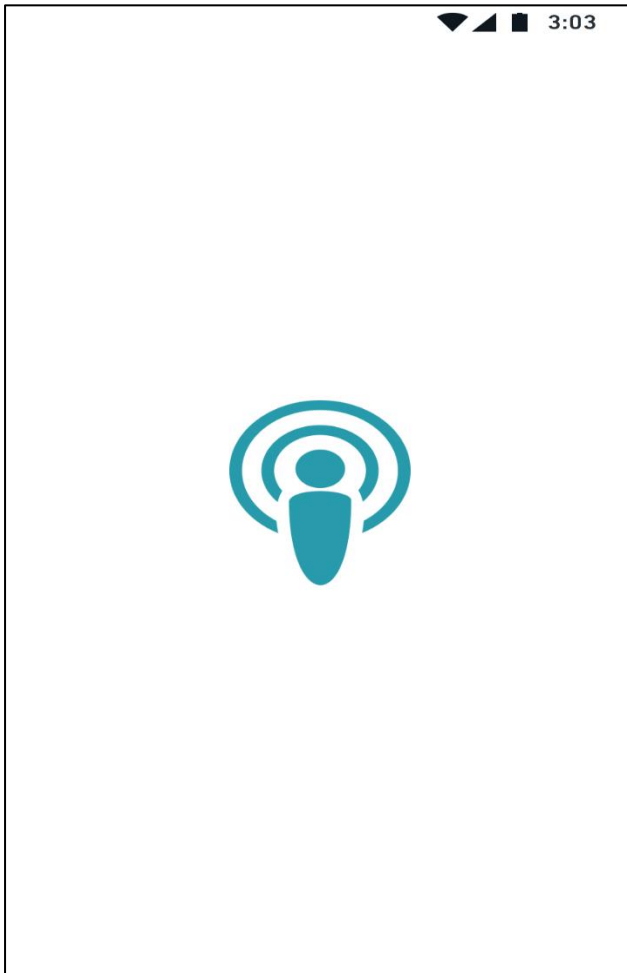
Source Code controller speaker

```
1 import 'package:flutter/cupertino.dart';
2 import 'package:speech_to_text/speech_to_text.dart';
3
4 class SpeechApi {
5   static final _speech = SpeechToText();
6
7   static Future<bool> toggleRecording({
8     required Function(String text) onResult,
9     required ValueChanged<bool> onListening,
10  }) async {
11    if (_speech.isListening) {
12      _speech.stop();
13      return true;
14    }
15
16    final isAvailable = await _speech.initialize(
17      onStatus: (status) => onListening(_speech.isListening),
18      onError: (e) => print('Error: $e'),
19    );
20
21    if (isAvailable) {
22      _speech.listen(onResult: (value) => onResult(value.recognizedWords));
23    }
24  }
25 }
```

❖ USER MANUAL MESIN PENERJEMAH

Pengujian Halaman Mesin Penerjemah

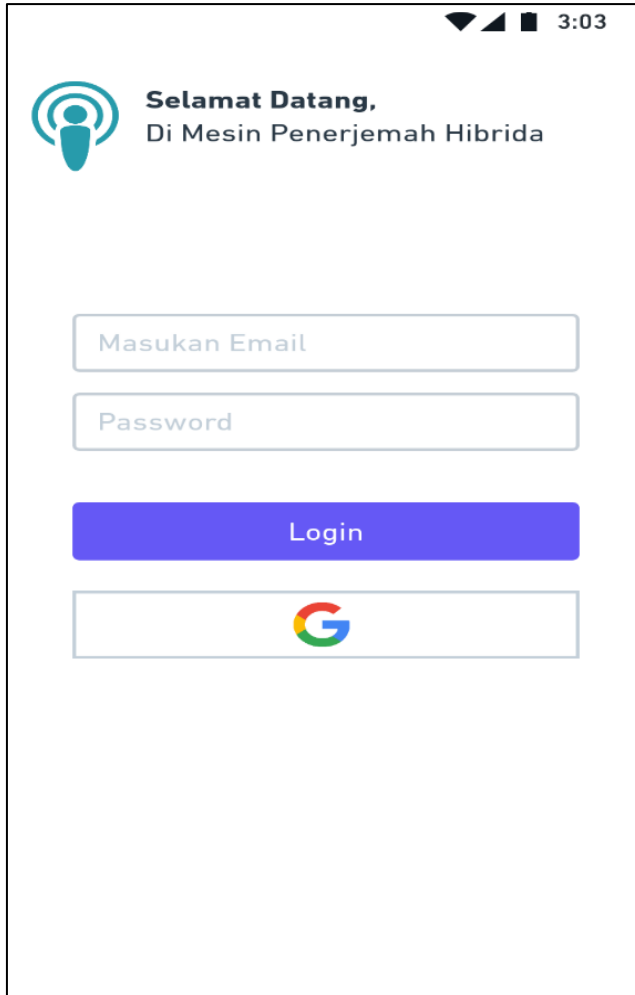
Dibawah ini adalah tampilan awal saat pertama kali membuka aplikasi.



Komponen Yang Diuji	Skenario Pengujian	Hasil Yang Diharapkan	Hasil
Halaman Mesin Penerjemah	Menampilkan halaman awal dari aplikasi	Sistem berhasil menampilkan tampilan awal aplikasi	Berhasil

Pengujian Halaman Login

Dibawah ini adalah tampilan aplikasi untuk login ke aplikasi mesin penerjemah hibrida dengan memasukkan email dan password atau bisa login menggunakan akun google.



The screenshot shows a mobile application interface for a hybrid translation machine. At the top right, the time is 3:03. The main heading reads "Selamat Datang, Di Mesin Penerjemah Hibrida" (Welcome to the Hybrid Translation Machine). Below the heading are four input fields: "Masukan Email" (Enter Email), "Password", a blue "Login" button, and a field with the Google logo for sign-in.

Komponen Yang Diuji	Skenario Pengujian	Hasil Yang Diharapkan	Hasil
Halaman Login	Menampilkan halaman untuk melakukan login	Sistem berhasil menampilkan halaman untuk melakukan login	Berhasil

Pengujian Halaman Gabung Room

Dibawah ini adalah tampilan untuk memasukkan room ID dan password untuk bergabung ke dalam sebuah room



Masukan Room Id dan Password untuk bergabung

Masukan Room ID

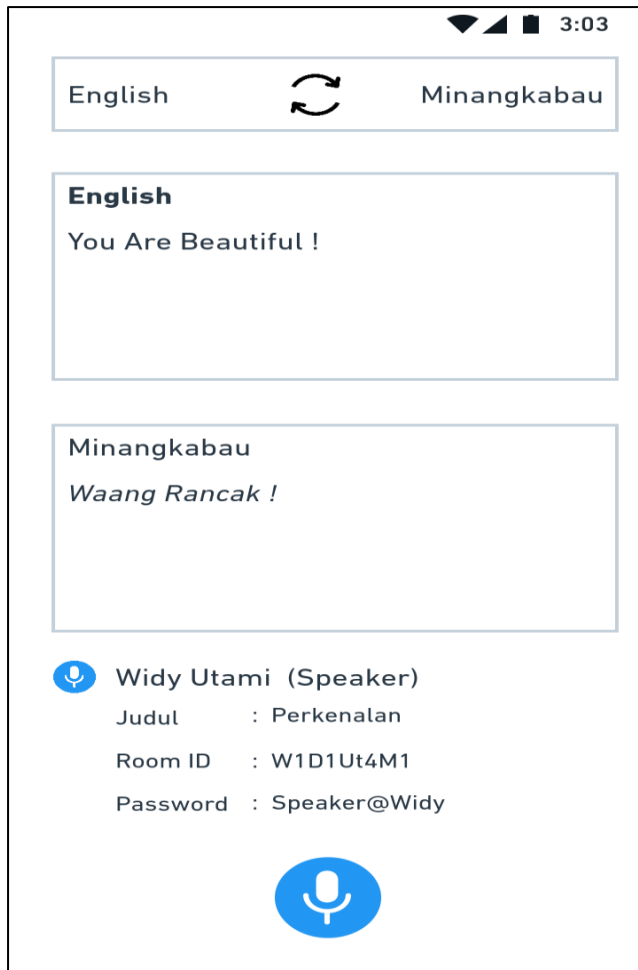
Password

Gabung Sekarang

Komponen Yang Diuji	Skenario Pengujian	Hasil Yang Diharapkan	Hasil
Halaman Gabung Room	Menampilkan halaman untuk masuk ke room	Sistem berhasil menampilkan halaman untuk masuk ke room	Berhasil

Pengujian Halaman Room Untuk Pembicara

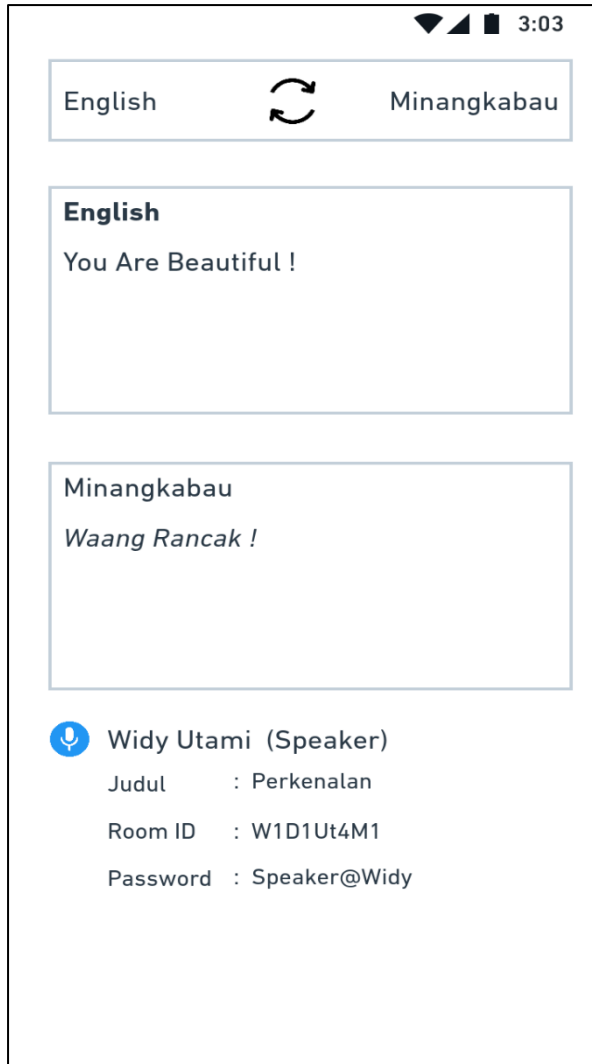
Dibawah ini adalah tampilan room speaker atau room pembicara dimana pembicara bisa berbicara sesuai Bahasa yang dipilih dan dibawah terdapat deskripsi nama pembicara, judul, room ID dan password.



Komponen Yang Diuji	Skenario Pengujian	Hasil Yang Diharapkan	Hasil
Halaman Room Untuk Pembicara	Menampilkan halaman room untuk pembicara sesuai Bahasa yang dipilih	Sistem berhasil menampilkan halaman room untuk pembicara	Berhasil

Pengujian Halaman Room Untuk Peserta

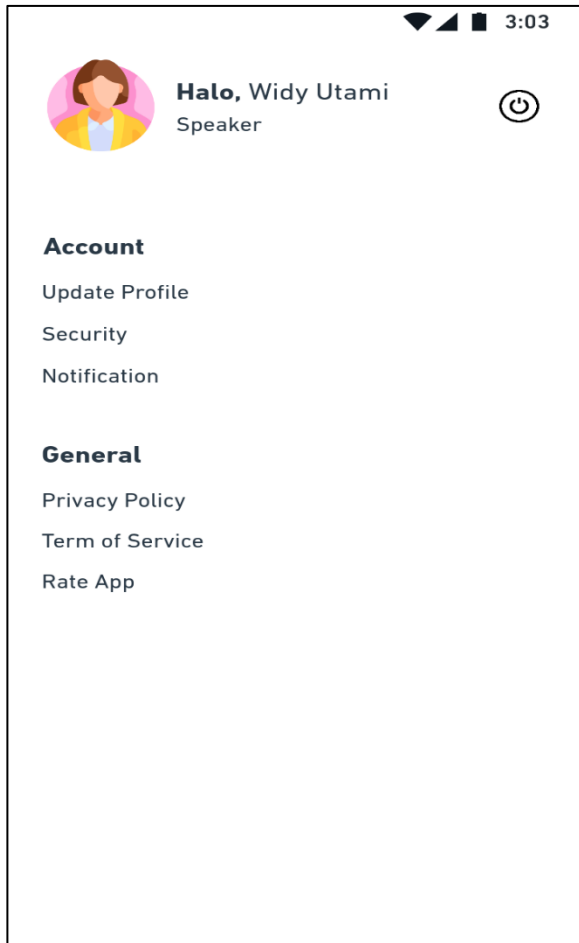
Dibawah ini adalah tampilan room peserta dimana peserta hanya bisa mendengarkan pembicara dan dibawah terdapat deskripsi nama pembicara, judul, room ID dan password.



Komponen Yang Diuji	Skenario Pengujian	Hasil Yang Diharapkan	Hasil
Halaman Room Untuk Peserta	Menampilkan halaman room peserta hanya untuk mendengarkan pembicara	Sistem berhasil menampilkan halaman room untuk peserta	Berhasil

Pengujian Halaman Akun

Dibawah ini adalah tampilan dari halaman akun pembicara ataupun peserta dimana mereka bisa melakukan update profile, security, dan notification dan mereka juga bisa melihat privacy policy, term of service dan memberikan rating pada aplikasi melalui rate app



Komponen Yang Diuji	Skenario Pengujian	Hasil Yang Diharapkan	Hasil
Halaman Akun	Menampilkan halaman Akun	Sistem berhasil menampilkan halaman akun	Berhasil

