

SightCom

App for Visual Accessibility

Louis JZ

Flutter app powered by Clarifai API



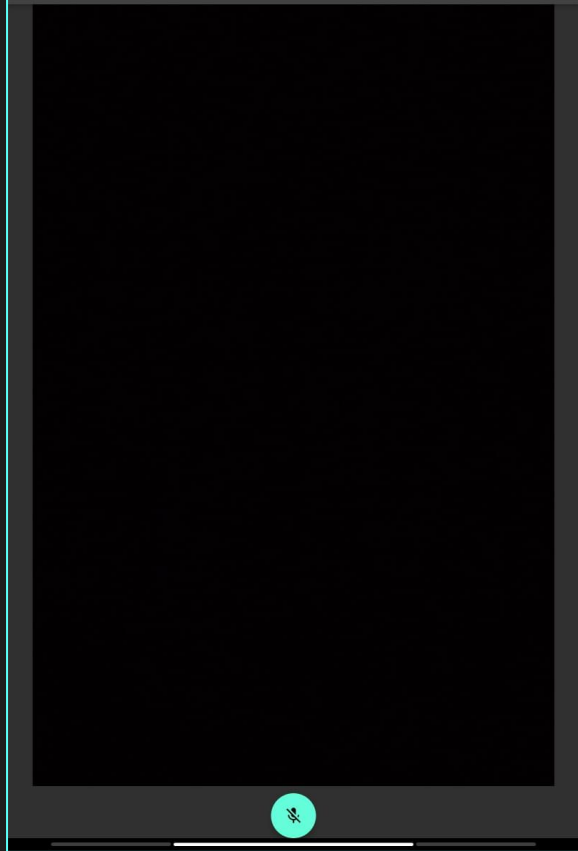


Introduction

A mobile app that aims to empower blind people's accessibility by unlocking vision through AI communication. This app directly aligns with UN SDG: Goal 3 - "Good Health and Well-being," and Goal 10 - "Reduced Inequalities."

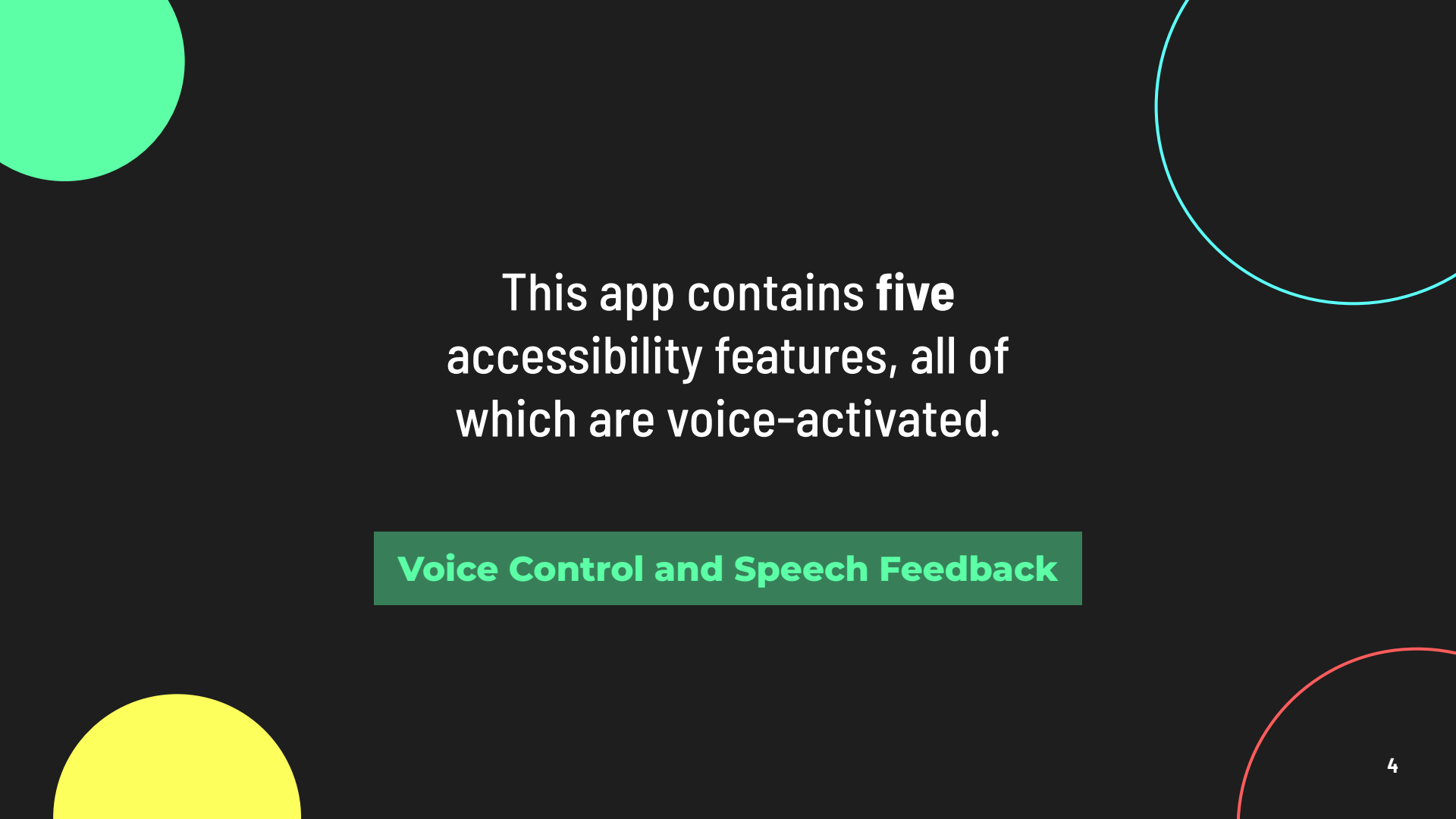


SightCom Blind Accessibility



Technologies

- ❖ Front-End:
Flutter (Dart)
- ❖ Back-End:
Clarifai API
- ❖ Database:
[Barcode Lookup](#)



This app contains **five** accessibility features, all of which are voice-activated.

Voice Control and Speech Feedback



SightCom

Unlocking Vision through
AI Communication

App Demonstration

Image Captioning

```
Future<void> describeScene() async {
  final imageBytes = await takePicture();
  final response = await sendClarifaiRequest(imageBytes, 'image-to-text');
  if (response is Map<String, dynamic>) {
    final description =
      response['text']['raw'];

    speak(description);
  } else {
    speak(response);
  }
}
```

Community > Model > general-english-image-caption-blip-2

general-english-image-caption-blip-2

 salesforce / blip

BLIP-2, a scalable multimodal pre-training method that enables any Large Language Models (LLMs) to ingest and understand images, unlocks the capabilities of zero-shot image-to-text generation. BLIP-2 is quick, efficient, and accurate.

71cb9 71cb98f572694e28a99fa8fa8...  

[See versions table](#)

Optical Character Recognition

```
Future<void> recognizeText() async {
  final imageBytes = await takePicture();
  final response = await sendClarifaiRequest(imageBytes, 'text-recognition');
  if (response is Map<String, dynamic>) {
    final results = response['regions'];
    if (results != null) {
      String recognizedText = results
        .map((region) {
          return region['data']['text']['raw'];
        })
        .join(' ')
        .toLowerCase();

      speak(recognizedText);
    } else {
      speak('no text detected');
    }
  } else {
    speak(response);
  }
}
```

Community > Model > ocr-scene-english-paddleocr

ocr-scene-english-paddleocr

 clarifai / main

An OCR model for detecting and recognizing English text in images that are more complex than scans of a page.

46e99 46e99516c2d94f58baf2bcfaf...  

[See versions table](#)

Barcode Detection

```
Future<void> readBarcode() async {
  final imageBytes = await takePicture();
  final response =
    await sendClarifaiRequest(imageBytes, 'barcode-recognition');
  if (response is Map<String, dynamic>) {
    final results = response['regions'];
    if (results != null) {
      int noOfBarcodes = results.length;
      if (noOfBarcodes > 1) {
        speak('$noOfBarcodes barcodes detected');
        List<String> descriptions = [];
        for (int i = 0; i < noOfBarcodes; i++) {
          final code = results[i]['data']['text']['raw'];
          final productDesc = await barcodeLookup(code);
          int number = i + 1;
          descriptions.add('Barcode $number. $productDesc');
        }
        speak(descriptions.join('; '));
      } else {
        final code = results[0]['data']['text']['raw'];
        final productDesc = await barcodeLookup(code);
        speak(productDesc);
      }
    } else {
      speak('no barcode detected');
    }
  } else {
    speak(response);
  }
}
```

Community > Model > BARCODE-QRCODE-Reader

BARCODE-QRCODE-Reader

 yuchen / workflow-test

Operator that detects and recognizes barcodes and QR codes from the image.

47850 47850e63a4c3436d9527cdb...   [See versions table](#)



<https://www.barcodelookup.com/>

Color Classifier

```
Future<void> detectColor() async {  
  final imageBytes = await takePicture();  
  final response = await sendClarifaiRequest(imageBytes, 'color-recognition');  
  if (response is Map<String, dynamic>) {  
    final color = response['colors'][0]  
      ['w3c']['name'];  
  
    speak(color);  
  } else {  
    speak(response);  
  }  
}
```

Community > Model > color-recognition

color-recognition



clarifai / main

Predict dominant colors and returns density values and colors in hexadecimal format and also mapped to its closest W3C counterparts.

dd945 dd9458324b4b45c2be1a7b...



[See versions table](#)

Llama 2 LLM

```
Future<void> askChatBot(prompt) async {  
  final response = await sendClarifaiRequest(prompt, 'chatbot');  
  if (response is Map<String, dynamic>) {  
    final answer = response['text']['raw'];  
    speak(answer);  
  } else {  
    speak(response);  
  }  
}
```

Community > Model > llama2-7b-chat

llama2-7b-chat



Llama 2-Chat is a fine-tuned large language model(LLM) optimized for dialogue use cases.

e52af e52af5d6bc22445aa7a6761f...



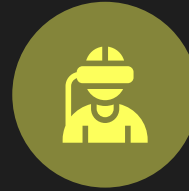
[See versions table](#)

Future Prospects



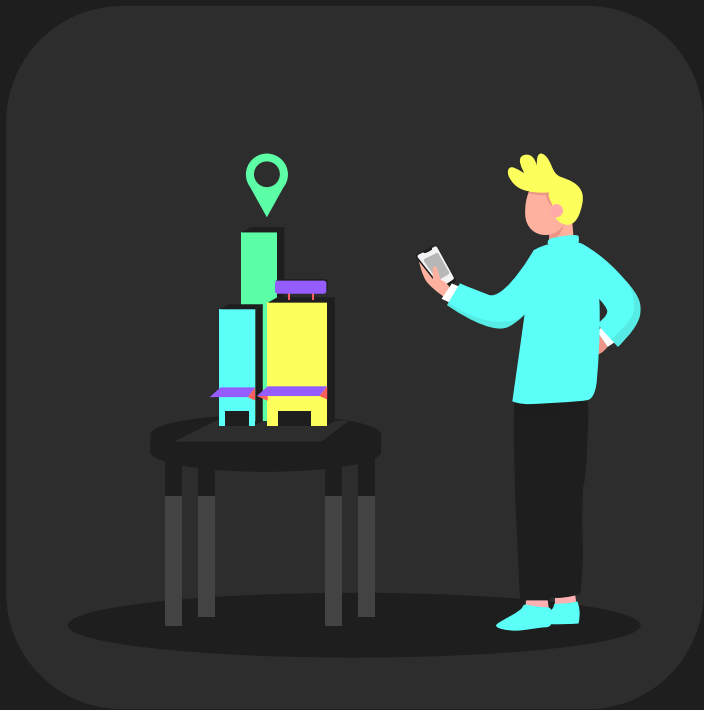
Addition of new Features

- Facial Identification
- Currency Recognition
- Navigation Assistance



Build Hardware Prototype

Using microprocessor like Raspberry Pi, with camera, microphone and speaker devices, to innovate a special glasses for blind people to wear.



Thank You!

Credits: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, infographics & images by **Freepik**