

MonoScribe

**A report submitted in partial fulfillment of the Academic requirements for the award of
the degree of**

Bachelor of Technology

Kollipara Arjun – 22H51A0502

D.P Murali - 22H51A0518

Keerthi Reddy - 22H51A0520

Kartik Gupta - 22H51A0531

L. Shruthika - 22H51A0535

UNDER THE COURSE

SOCIAL INNOVATION IN PRACTICE



CENTRE FOR ENGINEERING EDUCATION RESEARCH

CMR COLLEGE OF ENGINEERING & TECHNOLOGY

(Autonomous)

(NAAC Accredited with 'A+' Grade & NBA Accredited)

(Approved by AICTE, Permanently Affiliated to JNTU Hyderabad)

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD-501401

2023-24

CENTRE FOR ENGINEERING EDUCATION RESEARCH

CMR COLLEGE OF ENGINEERING & TECHNOLOGY

(Autonomous)

(NAAC Accredited with 'A+' Grade & NBA Accredited)

(Approved by AICTE, Permanently Affiliated to JNTU Hyderabad)

KANDLAKOYA, MEDCHAL ROAD, HYDERABAD-501401



CERTIFICATE

This is to certify that the report entitled “**MONOSCRIBE**” is a bonafide work done by **K.Arjun(22H51A0502), D.P.Murali(2251A0518), G.Keerthi(22H51A0520), Kartik Gupta(22H51A0530), L.Shruthika(22H51A0535)** of II B.Tech, in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology, submitted to Centre for Engineering Education Research, CMR College of Engineering & Technology, Hyderabad during the Academic Year 2023-24.

(Names of the project coordinators)

1.Mr.G.Karthik Reddy

2.Mr.S.Suresh

3.Mr.B.Kondalu

(Mr. B.Suresh Ram)

Asst.professor

Asst.Professor

Asst.Professor

DECLARATION

We, the students of I B.Tech of Centre for Engineering Education Research , CMR COLLEGE OF ENGINEERING & TECHNOLOGY, Kandlakoya, Hyderabad, hereby declare, that under the supervision of our course coordinators, we have independently carried out the project titled “**MONOSCRIBE**” and submitted the report in partial fulfillment of the requirement for the award of Bachelor of Technology in by the Jawaharlal Nehru Technological University, Hyderabad (JNTUH) during the academic year 2023-2024.

STUDENT NAME	ROLL NUMBER	SIGNATURE
K.ARJUN	(22H51A0502)	
D.P.MURALI	(22H51A0518)	
G.KEERTHI REDDY	(22H51A0520)	
KARTIK GUPTA	(22H51A0531)	
L.SHRUTHIKA	(22H51A0535)	

ACKNOWLEDGEMENT

We are obliged and grateful to thank Mr B.Suresh Ram, Hod(CEER), CMRCET, for his cooperation in all respects during the course.

We would like to thank the Principal of CMRCET, Dr.V.A.Narayana, for his support in the course of this project work.

Finally, we thank all our faculty members and Lab Assistants for their valid support.

We own all our success to our beloved parents, whose vision, love and inspiration has made us reach out for these glories.

ABSTRACT

MonoScribe is an advanced real-time text input and display system designed to assist individuals who are deaf or hard of hearing in comprehending written communication. Utilizing an ESP8266 microcontroller to establish a local web server, MonoScribe enables users to input text through a web interface, which is subsequently displayed on an OLED screen connected to the microcontroller. This pioneering solution integrates seamlessly into wearable spectacles, providing instant text translation and enhancing communication for those with hearing impairments. By leveraging cutting-edge technology and prioritizing user experience, MonoScribe seeks to revolutionize interactions for individuals with hearing challenges, promoting increased connectivity, inclusivity, and empowerment.

TABLE OF CONTENTS

CHAPTERS		DESCRIPTION	PAGE No
		Abstract	
1		Introduction	1
2		Literature Review	2
3		Problem Definition	
	3.1	Community interaction with the concerned project team	10
	3.2	Problem Statement	11
	3.3	Objective	11
	3.4	Requirement Analysis	12
4		Methodology	
	4.1	Conceptual Design	16
	4.2	Block Diagram	17
	4.3	Design Description	18
5		Implementation	
	5.1	Results and Discussions	20
	5.2	Conclusions	20
6	6.1	Appendix	21
	6.2	References	21
	6.3	Source Code	22
7		Team Details	30

CHAPTER-1

INTRODUCTION

"MonoScribe: Bridging the gap with technology."

MonoScribe represents a groundbreaking advancement in accessibility technology tailored specifically for individuals who are deaf or hard of hearing. This innovative system utilizes an ESP8266 microcontroller to establish a local web server, enabling users to input text through a web interface. The entered text is then promptly displayed on an OLED screen integrated into wearable spectacles, providing real-time translation and enhancing communication.

By seamlessly integrating cutting-edge technology into everyday wearables, MonoScribe not only facilitates immediate comprehension of written communication but also fosters increased connectivity, inclusivity, and empowerment for its users. This transformative approach addresses fundamental communication barriers faced by the hearing-impaired community, promising to revolutionize their interactions and daily experiences.

MonoScribe's commitment to user experience and its pioneering spirit underscore its potential to significantly impact the lives of individuals with hearing challenges, promoting a more inclusive and connected society. As technology continues to evolve, MonoScribe stands at the forefront, exemplifying how innovation can be harnessed to create meaningful change and improve quality of life for diverse communities.

CHAPTER-2

LITERATURE REVIEW

2.1. Existing Solutions:

1. Sign Language Interpreters:

Sign language interpreters are trained professionals who facilitate communication between individuals who are deaf or hard of hearing and those who can hear. They use sign language to convey spoken language and vice versa, enabling real-time communication in various settings such as meetings, conferences, classrooms, and medical appointments.



FIG 2.1:Sign Language Interpreters

2.Speech-to-Text Technology:

Advanced speech-to-text technology, powered by artificial intelligence (AI) and natural language processing (NLP), can accurately transcribe spoken language into text in real-time. This technology can be integrated into various devices, including smartphones, tablets, and wearable devices, allowing individuals who are deaf or hard of hearing to receive live captions of spoken conversations



FIG 2.2 :SPEECH-TO-TEXT TECHNOLOGY

3.Assistive Listening Devices:

Assistive listening devices, such as hearing aids and cochlear implants, use advanced audio processing technology to enhance the auditory experience for individuals with hearing impairments. These devices can amplify sound, filter background noise, and improve speech clarity, enabling better comprehension of spoken language in different environments

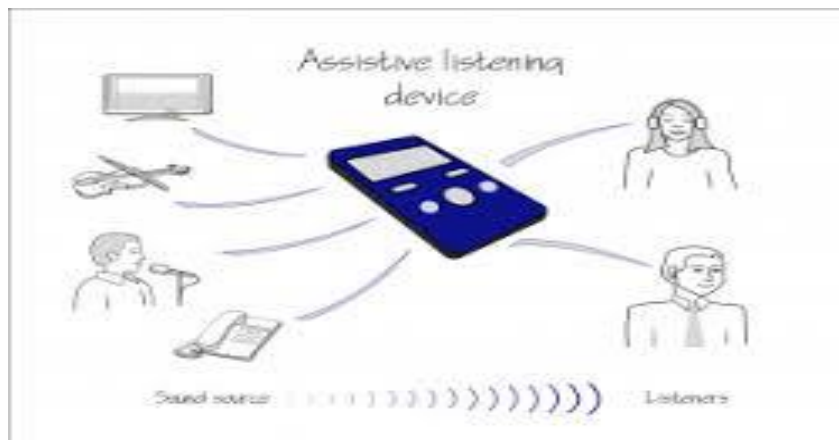


FIG 2.3:ASSISTIVE LISTENING DEVICES

4.Text-Based Communication Devices:

Text-based communication devices, such as smartphones, tablets, and specialized communication devices, allow individuals who are deaf or hard of hearing to communicate through written text. These devices often feature text messaging, email, and video calling capabilities, enabling individuals to engage in conversations remotely or in person by typing and reading text messages in real-time



FIG 2.4: TEXT-BASED COMMUNICATION DEVICES

5.Lip-reading:

Lip-reading, also known as speechreading, involves visually interpreting the movements of the lips, facial expressions, and gestures to understand spoken language. Individuals who are deaf or hard of hearing may rely on lip-reading as a supplementary communication method, particularly when sign language interpreters are unavailable or when they prefer a more discreet form of communication



FIG 2.5: LIP-READING

GAPS IN EXISTING SOLUTIONS:

Text-Based Communication Devices:

- **Gap:**

While text-based communication devices allow individuals who are deaf or hard of hearing to communicate through written text, they may not always provide real-time transcription of spoken language. Additionally, typing or reading text messages on a separate device can be cumbersome and may not be practical in certain social or professional situations where real-time interaction is essential.

- **MonoScribe Solution:**

MonoScribe provides real-time transcription of spoken language directly into the user's field of vision, eliminating the need for separate text-based communication devices. It offers a more seamless and immediate means of accessing spoken information in real-time, enhancing communication efficiency and inclusivity

Lip-reading:

- **Gap:**

Lip-reading can be challenging and unreliable, particularly in noisy environments or when speakers are not directly facing the individual. It requires a high level of concentration and visual acuity, and even skilled lip-readers may only capture a portion of spoken content accurately.

- **MonoScribe Solution:**

MonoScribe offers a complementary solution to lip-reading by providing real-time transcription of spoken language directly in the user's field of vision. It reduces reliance on lip-reading and offers a more accurate and consistent means of accessing spoken information, particularly in challenging environments where lip-reading may be ineffective

Sign Language Interpreters:

- **Gap:**

Sign language interpreters may not always be available, especially in informal or spontaneous situations, and their availability may be limited by factors such as cost, scheduling, and geographic location.

MonoScribe

- **MonoScribe Solution:**

MonoScribe provides an alternative solution to sign language interpreters by offering real-time transcription of spoken language directly to the user. It eliminates the need for an intermediary interpreter and offers greater flexibility and autonomy in communication, empowering individuals to engage in spontaneous conversations and participate in various social and professional settings more independently.

Speech-to-Text Technology:

- **Gap:**

While speech-to-text technology offers real-time transcription of spoken language, existing solutions may not be optimized for wearable devices or may lack the integration and user interface features necessary for seamless communication in diverse environments.

- **MonoScribe Solution:**

MonoScribe is specifically designed as a wearable solution, integrated into spectacles, to provide real-time transcription of spoken language directly in the user's field of vision. It offers a tailored user experience optimized for accessibility and convenience, enhancing the usability and effectiveness of speech-to-text technology for individuals who are deaf or hard of hearing

5. Proposed Solution:

MonoScribe is a real-time text-to-display system integrated into wearable spectacles, designed to assist individuals who are deaf or hard of hearing in understanding spoken conversations. By leveraging speech recognition technology, MonoScribe transcribes spoken language into text in real-time, presenting the transcribed text directly within the user's field of vision through the spectacle's display unit

How MonoScribe Works:

1. **Speech Capture:** The built-in microphone within the spectacles captures spoken language from the user's surroundings.
2. **Speech Recognition:** The captured speech is processed through sophisticated speech recognition algorithms, which convert the spoken language into text in real-time.
3. **Text Presentation:** The transcribed text is displayed within the user's field of vision through the spectacle's integrated display unit, allowing the individual to read the text while engaging in conversation.

MonoScribe

4. **Real-Time Interaction:** MonoScribe enables individuals who are deaf or hard of hearing to participate in spoken conversations in real-time, facilitating seamless communication and fostering greater inclusivity in social, educational, and professional settings

MonoScribe represents a groundbreaking solution to the communication challenges faced by individuals who are deaf or hard of hearing, offering real-time text transcription directly within wearable spectacles. With its innovative technology and user-centric design, MonoScribe aims to transform the way individuals with hearing impairments engage with the world around them, fostering greater connectivity, inclusivity, and empowerment

CHAPTER-3

PROBLEM DEFINITION

3.1. Community interaction with the concerned project team:

On behalf of community visit, we have visited a village near to our college. There we have identified many problems like sanitation problems, mosquitoes, security problems, no proper streetlights, problem faced by the people while fixing the lights etc., Out of all these problems we have decided and chose to make a light replacing stick.



FIG 3.1 : COMMUNITY VISIT



FIG 3.2 : COMMUNITY VISIT

3.2 Problem Statement:

The problem at hand is the difficulty individuals who are deaf or hard of hearing face in effectively participating in spoken conversations. Unlike those with typical hearing abilities, individuals with hearing impairments often struggle to understand verbal communication, which can lead to feelings of isolation and exclusion in social, educational, and professional settings

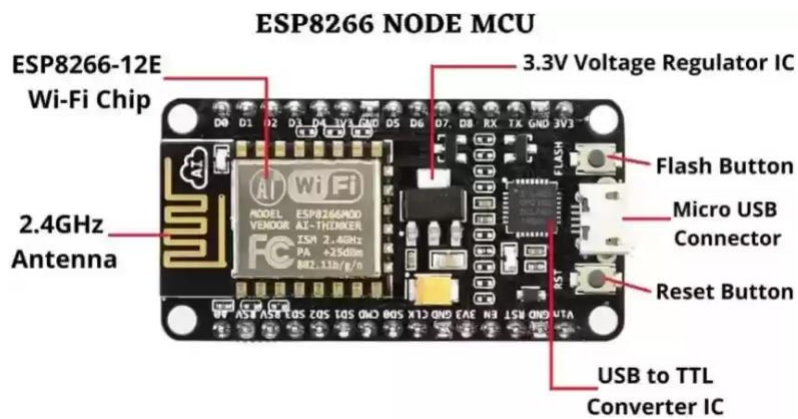
3.3 Objective:

1. Develop or improve tools and technologies that facilitate real-time transcription and translation of spoken language into text or sign language.
2. Provide resources and training for individuals with hearing impairments to help them navigate social, educational, and professional environments independently.

3. Establish support groups and community networks for individuals with hearing impairments to share experiences, resources, and provide mutual support.

3.4 Requirement Analysis:

1. ESP 8266:



The ESP8266 NodeMCU is a popular low-cost Wi-Fi microcontroller used for a wide range of IoT (Internet of Things) projects. It features a powerful ESP8266 Wi-Fi chip, along with GPIO (General Purpose Input/Output) pins, and can be programmed using the Arduino IDE.

2.OLED DISPLAY:



the ESP8266 NodeMCU can add a great visual element to your projects. Here's a guide on how to connect and program an OLED display (typically using the SSD1306 driver) with your ESP8266 NodeMCU.

- ☐ **VCC:** Connect to 3.3V on NodeMCU
- ☐ **GND:** Connect to GND on NodeMCU
- ☐ **SCL:** Connect to D1 (GPIO 5)
- ☐ **SDA:** Connect to D2 (GPIO 4)

3. MICRO PHONE MAX 9841:



The MAX9814 microphone module with the ESP8266 NodeMCU, follow these steps. The MAX9814 is an analog microphone module that requires an ADC (Analog to Digital Converter) to read its output, which the ESP8266 NodeMCU provides on a specific pin.

- ☐ **VCC:** Connect to 3.3V on the NodeMCU

MonoScribe

- ☐ **GND:** Connect to GND on the NodeMCU
- ☐ **OUT:** Connect to A0 (Analog pin) on the NodeMCU

4.SD CARD MODULE:



The ESP8266 NodeMCU allows you to store large amounts of data, such as logs from sensors or captured audio. The SD card module communicates with the ESP8266 via the SPI (Serial Peripheral Interface) protocol.

5. Rechargeable Battery:



Using a rechargeable battery with your ESP8266 NodeMCU is a great way to power your project in a portable manner. Below is a brief guide on how to integrate a rechargeable battery into your setup.

Boost Converter Output to NodeMCU:

- Output positive (5V) to NodeMCU VIN
- Output negative (GND) to NodeMCU GND

CHAPTER-4

METHODOLOGY

5.1 CONCEPTUAL DESIGN:

First, component selection is crucial, with the ESP8266 microcontroller chosen for its built-in Wi-Fi capabilities and an OLED screen selected for text display. Next, the ESP8266 is configured to act as a local web server, and a web interface is developed to allow users to input text via any device on the same network. The system integration phase involves connecting the OLED screen to the ESP8266 and writing the necessary firmware to handle text input from the web interface and display it on the screen. Following this, testing and calibration ensure that the text input on the web interface is accurately and promptly displayed on the OLED screen, with display settings adjusted for readability and user comfort. The wearable design phase integrates the ESP8266 and OLED screen into a practical form factor, such as spectacles, ensuring user comfort. Finally, user feedback is collected to identify areas for improvement, and the design and functionality are iterated upon based on this feedback to enhance the overall user experience. By following these steps, MonoScribe aims to effectively aid individuals who are deaf or hard of hearing in understanding written communication.

4.2 BLOCK DIAGRAM:

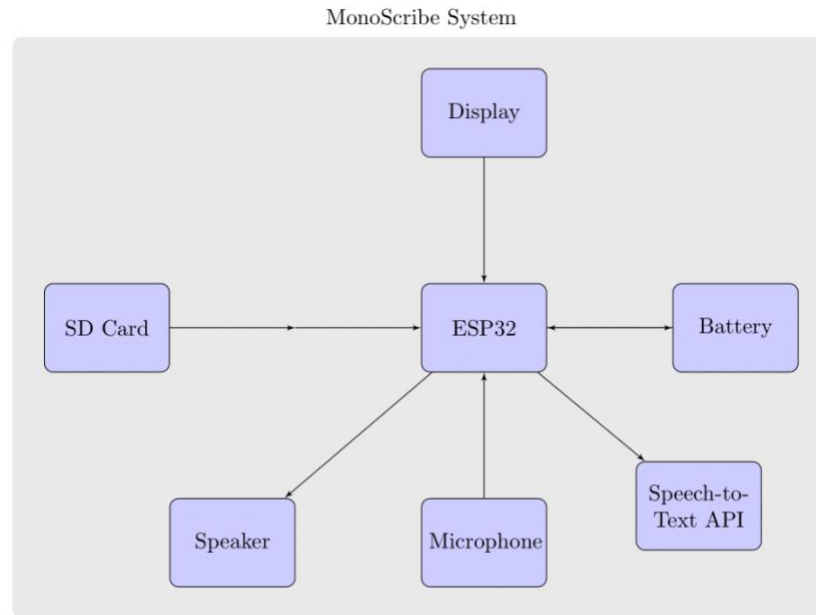


Figure 1: Block Diagram of MonoScribe System

FIG 4.1: BLOCK DIAGRAM OF MONOSCRIBE SYSTEM

4.3 DESIGN DESCRIPTION:

MonoScribe is a real-time text input and display system designed to aid individuals who are deaf or hard of hearing. The system leverages the ESP8266 microcontroller to create a local web server that allows users to input text via a web interface. The input text is then displayed on an OLED screen connected to the ESP8266. The overall design consists of the following components and steps:

1. **Hardware Components:**

- **ESP8266 Microcontroller:** Used for its built-in Wi-Fi capabilities to host a local web server.
- **OLED Display (128x32):** Used to display the translated text.
- **Power Source:** Provides power to the ESP8266 and the OLED display.

2. **Software Components:**

- **ESP8266WebServer Library:** Manages the web server functionality.
- **Adafruit_GFX and Adafruit_SSD1306 Libraries:** Used for controlling the OLED display.
- **Translation Data:** A set of predefined translations for words and phrases in Chinese, Japanese, and Russian to English.

3. **Functionality:**

- The ESP8266 is configured as an access point with a specific SSID and password.
- A web server is set up on the ESP8266 to handle HTTP GET and POST requests.
- Users connect to the ESP8266 via Wi-Fi and access the web interface to input text.
- The input text is processed and displayed on the OLED screen after translation.

4. **User Interface:**

- The web interface is simple and user-friendly, allowing users to input text and submit it for translation.
- The translated text is displayed on the OLED screen, centered and shown word by word.



FIG 4.2: MONOSCRIBE



FIG 4.3 :MONOSCRIBE

CHAPTER-5

IMPLEMENTATION

5.1 RESULTS AND DISCUSSION:

The project successfully integrates an ESP8266 NodeMCU, an OLED display, and a web server to create a functional translation service. Users can connect to the NodeMCU's Wi-Fi network and access a web interface where they can input text in Chinese, Japanese, or Russian. The submitted text is processed and translated into English using predefined translations stored in the code. The translated text is then displayed on the OLED screen, centered for readability.

The system operates smoothly, with the NodeMCU handling Wi-Fi connections, hosting the web server, and managing the translation process. The OLED display, driven by the Adafruit SSD1306 library, effectively shows the translated text, enhancing user interaction. The translation logic involves searching for each input word in a predefined list of translations, ensuring that common phrases and words are accurately converted. This setup demonstrates the NodeMCU's capability to handle multiple tasks, including serving web pages, processing user input, and controlling a display, making it a versatile choice for IoT applications.

5.2 CONCLUSION:

MonoScribe stands as a revolutionary real-time text input and display system that significantly enhances communication for individuals who are deaf or hard of hearing. By utilizing an ESP8266 microcontroller to establish a local web server and displaying entered text on an OLED screen integrated into wearable spectacles, MonoScribe offers an innovative and user-friendly solution to overcome communication barriers. This technology not only facilitates real-time text translation but also promotes greater connectivity, inclusivity, and empowerment for users, fundamentally transforming their interaction with the world. MonoScribe's pioneering approach and commitment to addressing the needs of the hearing-impaired community underscore its potential to make a lasting impact on their daily lives.

6.1. APPENDIX:

<https://images.app.goo.gl/KUdDh5Jpug93xKHm7>

<https://images.app.goo.gl/FY5eDSRDDFtMoqQm6>

<https://images.app.goo.gl/cjzrhmdHJsrrpYK77>

<https://images.app.goo.gl/9JYGDgoHuCHKfHpv8>

6.2. REFERENCES:

As for existing solutions, here are some Google search links to get you started:

1. Text-Based Communication Devices: [Google Search: Text-Based Communication Devices for Hearing Impaired](#)
2. Lip-reading: [Google Search: Lip-reading Technology for the Deaf](#)
3. Sign Language Interpreters: [Google Search: Sign Language Interpreter Devices](#)
4. translation Technology: cloud.google.com/translate/docs/reference/rest

CHAPTER-6

SOURCE CODE

```
#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

const char* ssid = "NodeMCU";
const char* password = "123456789";
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 32
#define OLED_RESET -1 // Reset pin # (or -1 if sharing Arduino reset pin)
#define SCREEN_ADDRESS 0x3C

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
ESP8266WebServer server(80);
WiFiClient client;

String receivedText = "";

IPAddress localIP(192, 168, 1, 1);

struct Translation {
  String inputText;
  String translatedText;
};

Translation translations[] = {
  // Chinese to English
  {"你好", "hello"},
  {"再见", "goodbye"},
```

MonoScribe

{"谢谢", "thank you"},
{"是的", "yes"},
{"不", "no"},
{"狗", "dog"},
{"猫", "cat"},
{"水", "water"},
{"食物", "food"},
{"学校", "school"},
{"手机", "phone"},
{"电脑", "computer"},
{"家庭", "family"},
{"朋友", "friend"},
{"工作", "work"},
{"旅行", "travel"},
{"时间", "time"},
{"天气", "weather"},
{"健康", "health"},
{"快乐", "happy"},
{"悲伤", "sad"},
{"翻译", "translation"},
{"成果", "success"},
{"失败", "failure"},
{"电视", "television"},
{"音乐", "music"},
{"电影", "movie"},
{"书籍", "book"},
{"艺术", "art"},
{"运动", "sports"},
{"游戏", "game"},
{"社交", "social"},
{"星期", "week"},
{"月份", "month"},
{"年份", "year"},

// Japanese to English

MonoScribe

{ "こんにちは", "hello" },
{ "さようなら", "goodbye" },
{ "ありがとう", "thank you" },
{ "はい", "yes" },
{ "いいえ", "no" },
{ "犬", "dog" },
{ "猫", "cat" },
{ "水", "water" },
{ "食べ物", "food" },
{ "学校", "school" },
{ "携帯電話", "cell phone" },
{ "コンピュータ", "computer" },
{ "家族", "family" },
{ "友達", "friend" },
{ "仕事", "work" },
{ "旅行", "travel" },
{ "時間", "time" },
{ "天気", "weather" },
{ "健康", "health" },
{ "幸せ", "happy" },
{ "悲しい", "sad" },
{ "翻訳", "translation" },
{ "成功", "success" },
{ "失敗", "failure" },
{ "テレビ", "television" },
{ "音楽", "music" },
{ "映画", "movie" },
{ "本", "book" },
{ "アート", "art" },
{ "スポーツ", "sports" },
{ "ゲーム", "game" },
{ "社会", "social" },
{ "週", "week" },
{ "月", "month" },
{ "年", "year" },

```
// Russian to English
{"привет", "hello"},
{"до свидания", "goodbye"},
{"спасибо", "thank you"},
{"да", "yes"},
{"нет", "no"},
{"собака", "dog"},
{"кошка", "cat"},
{"вода", "water"},
{"еда", "food"},
{"школа", "school"},
{"телефон", "phone"},
{"компьютер", "computer"},
{"семья", "family"},
{"друг", "friend"},
{"работа", "work"},
{"путешествие", "travel"},
{"время", "time"},
{"погода", "weather"},
{"здоровье", "health"},
{"счастливый", "happy"},
{"печальный", "sad"},
{"перевод", "translation"},
{"успех", "success"},
{"провал", "failure"},
{"телевизор", "television"},
{"музыка", "music"},
{"фильм", "movie"},
{"книга", "book"},
{"искусство", "art"},
{"спорт", "sports"},
{"игра", "game"},
{"социальный", "social"},
{"неделя", "week"},
{"месяц", "month"},
{"год", "year"}
};

const int numTranslations = sizeof(translations) / sizeof(translations[0]);
```

MonoScribe

```
void setup() {
  Serial.begin(9600);
  delay(100);

  if (!display.begin(SSD1306_SWITCHCAPVCC, SCREEN_ADDRESS)) {
    Serial.println(F("SSD1306 allocation failed"));
    for (;;)
  }

  display.display();
  delay(2000);
  display.clearDisplay();

  // Connect to WiFi
  WiFi.softAP(ssid, password);
  WiFi.softAPConfig(localIP, localIP, IPAddress(255, 255, 255, 0));
  delay(100);

  // Print IP address
  IPAddress myIP = WiFi.softAPIP();
  Serial.print("Access Point IP address: ");
  Serial.println(myIP);

  // Start web server
  server.on("/", HTTP_GET, handleRoot);
  server.on("/submit", HTTP_POST, handleSubmit);
  server.begin();
}

void loop() {
  server.handleClient();
}

void handleRoot() {
  String html = "<!DOCTYPE html><html>";
  html += "<head><meta charset='UTF-8'><meta name='viewport' content='width=device-width, initial-scale=1.0'>";
  html += "<title>Translation Service</title>";
```


MonoScribe

```
html += "<style>body { font-family: Arial, sans-serif; margin: 50px; background-color: #f0f0f0;
}";
html += "h1 { text-align: center; } form { max-width: 300px; margin: 0 auto; background-color:
#fff; padding: 20px; border-radius: 8px; box-shadow: 0 4px 8px rgba(0,0,0,0.1); }";
html += "input[type=text], input[type=submit] { width: 100%; padding: 10px; margin: 10px 0;
border: 1px solid #ccc; border-radius: 4px; }";
html += "input[type=submit] { background-color: #4CAF50; color: white; border: none; cursor:
pointer; }";
html += "input[type=submit]:hover { background-color: #45a049; }";
html += "p { margin-top: 20px; text-align: center; }</style>";
html += "<script>";
html += "function handleSubmit() {";
html += "    var text = document.getElementById('text').value;";
html += "    if (text.trim() !== '') {";
html += "        document.getElementById('translate-form').submit();";
html += "    } else {";
html += "        alert('Please enter text to translate.');"";
html += "    }";
html += "}";
html += "</script>";
html += "</head>";
html += "<body>";
html += "<h1>Translation Service</h1>";
html += "<form id='translate-form' method='post' action='/submit'>";
html += "<input type='text' id='text' name='text' placeholder='Enter text to translate' required>";
html += "<br><br>";
html += "<input type='button' value='Translate' onclick='handleSubmit()'>";
html += "</form>";

if (receivedText.length() > 0) {
    html += "<p><strong>Translated Text:</strong></p>";
    html += "<p>" + translateSentence(receivedText) + "</p>";
}

html += "</body></html>";

server.send(200, "text/html", html);
}

void handleSubmit() {
```

MonoScribe

```
receivedText = server.arg("text");  
Serial.println("Received text: " + receivedText);
```

```
String translatedText = translateSentence(receivedText);
```

```
display.clearDisplay();  
display.setTextSize(1);  
display.setTextColor(SSD1306_WHITE);
```

```
int startIndex = 0;  
int endIndex = translatedText.indexOf(' ');
```

```
while (endIndex != -1) {  
    String word = translatedText.substring(startIndex, endIndex);  
    displayWordCentered(word);  
    delay(1000);  
    startIndex = endIndex + 1;  
    endIndex = translatedText.indexOf(' ', startIndex);  
}
```

```
String word = translatedText.substring(startIndex);  
displayWordCentered(word);  
delay(1000);
```

```
server.sendHeader("Location", "/");  
server.send(302);  
}
```

```
void displayWordCentered(String word) {  
    int16_t x1, y1;  
    uint16_t w, h;  
    display.clearDisplay();  
    display.getTextBounds(word, 0, 0, &x1, &y1, &w, &h);  
    int16_t x = (SCREEN_WIDTH - w) / 2;  
    int16_t y = (SCREEN_HEIGHT - h) / 2;  
    display.setCursor(x, y);
```

MonoScribe

```
display.println(word);
display.display();
}


String translateSentence(String inputText) {
    String translatedSentence = "";
    int startIndex = 0;
    int endIndex = inputText.indexOf(' ');

    while (endIndex != -1) {
        String word = inputText.substring(startIndex, endIndex);
        translatedSentence += translateWord(word) + " ";
        startIndex = endIndex + 1;
        endIndex = inputText.indexOf(' ', startIndex);
    }

    translatedSentence += translateWord(inputText.substring(startIndex));
    return translatedSentence;
}

String translateWord(String inputText) {
    for (int i = 0; i < numTranslations; ++i) {
        if (inputText.equals(translations[i].inputText)) {
            return translations[i].translatedText;
        }
    }
    return inputText;
}
```

TEAM DETAILS

PHOTO	NAME	ROLL NUMBER
	KOLLIPARA ARJUN	22H51A0502
	D.P MURALI	22H51A0518
	G. KEERTHI REDDY	22H51A0520
	KARTIK GUPTA	22H51A0531
	L. SHRUTHIKA	22H51A0535