# Relatório

Trabalho realizado por:

Felipe Campelo Sabbado - 20191012

Leonardo Rodrigues - 20200859

João Araújo - 20200989

José Ribeiro - 20200104

Data: 21/05/2023

# Lista de equipamentos:

Esp32 | Quantidade: 3

Bateria | Quantidade: 2

MPU6050 | Quantidade: 1

Leds | Quantidade: 3

Sensor Hall | Quantidade: 1

Botões | Quantidade: 1

Buzzer | Quantidade: 1

Buck converter | Quantidade: 1

Servo motor | Quantidade: 1

Termistor | Quantidade: 1

Câmera | Quantidade: 1

Speaker | Quantidade: 1

PCM5102A | Quantidade: 1

Sensor Ultra Som | Quantidade: 1

# Funcionalidades:

O MPU é usado para:

1. Medir a temperatura do lado de fora do capacete.
2. Ajuda a perceber se houve uma queda caso tenha havido uma queda, aciona o buzzer e manda um sms para um contacto.
3. Aciona os piscas para a direita e esquerda
4. Aciona o pisca de travagem

O sensor Hall é usado para:

1. Saber se o capacete tem o trinco fechado ou não.

O botão e a câmera são usados para:

1. Quando clicamos num botão é tirado uma foto e armazenado num servidor e depois pode ser acessada pelo um link ( <https://ulideparty.ddns.net/gallery.php> ).

O Buck converter é usado para:

1. Converte voltagem de 2.4 para 12 volts.

O Servo motor é usado para:

1. Abrir um *flap* no capacete. Isso ocorre quando o sensor de temperatura do MPU atinge uma temperatura superior a 32 graus. E fechar quando ela diminui abaixo dessa temperatura.

O Termistor é usado para:

1. Medir temperatura no interior do capacete.

O Speaker é usado para:

1. Reproduzir sons vindos do telemóvel.

O PCM5102A é usado para:

1. Converter o sinal digital em analógico para o *speaker* conseguir reproduzir sons. Este sinal é recebido através de bluetooth.

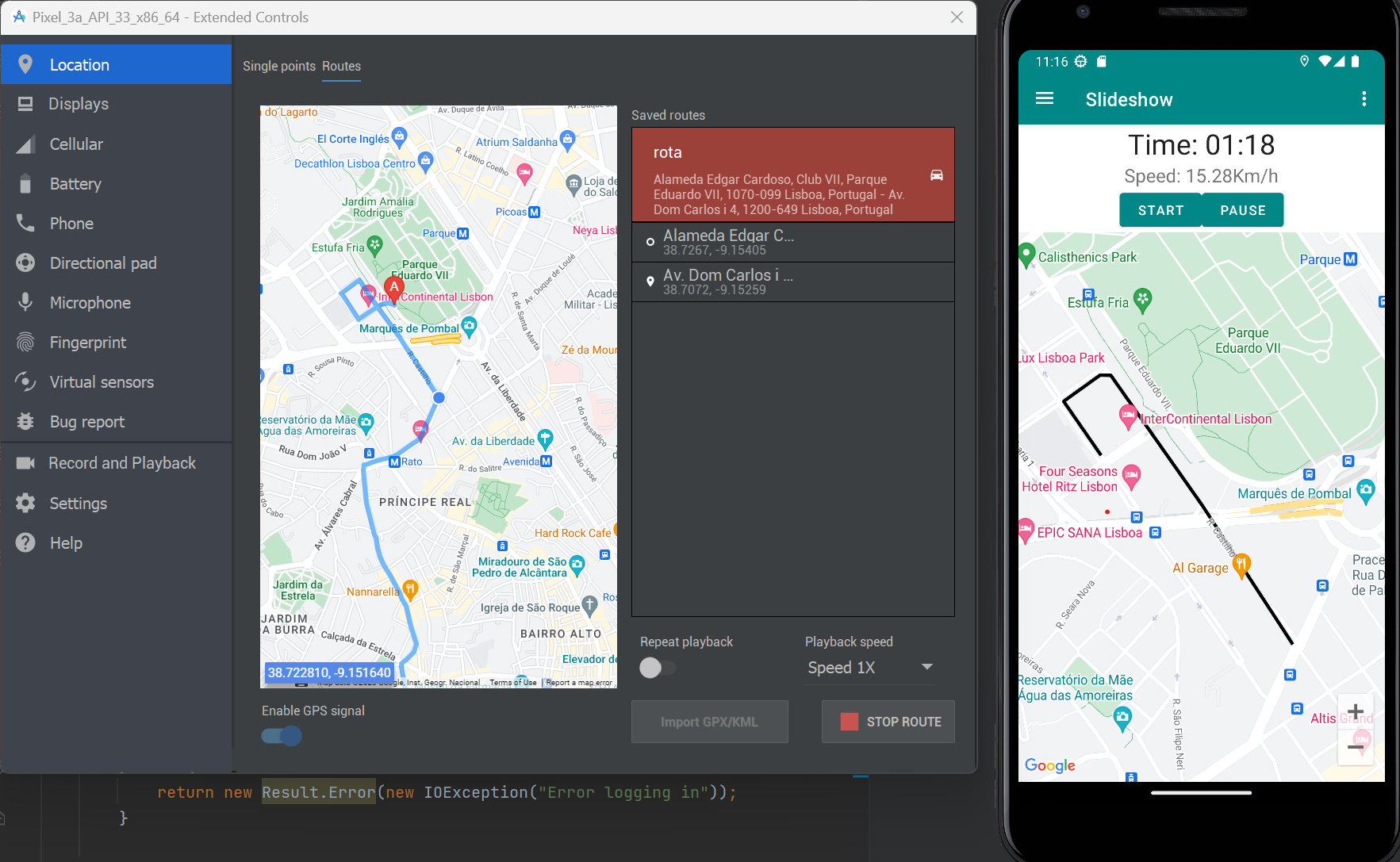
O Sensor Ultra Som é usado para:

1. Ver se há carros atrás da pessoa que está a usar o capacete. Se um carro estiver muito próximo vai acender um led na viseira do capacete.

O Bluetooth é usado para:

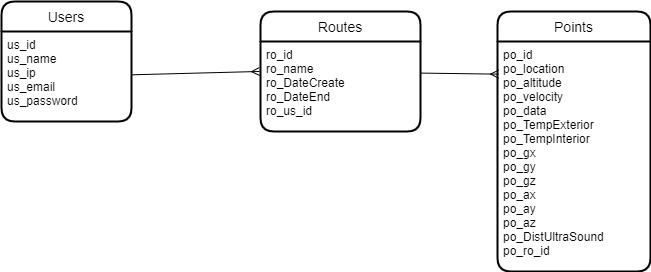
1. Receber e transmitir dados. Enviamos o id da pessoa do android para o esp32 e recebemos do lado do esp32.

Desenho da rota em tempo real:



(Para obter a velocidade estamos a usar o telemóvel e não o ESP32)

# Base de dados:



# Comunicação entre Esp 32, APP Android, Servidor NodeJs e Base de dados:

# 