

Localizador online de Veículos Estapar Estacionamentos

Controle do Documento

Histórico de revisões

Data	Autor	Versão	Resumo da atividade
17/11	Luiz Granville	3.1	Atualização da seção 1.1 e 1.2
21/11	Luísa Leite	3.2	Atualização da seção 2

Índice

1. Introdução	3
1.1. Solução	3
1.2. Arquitetura da Solução	3
2. Componentes e Recursos	4
2.1. Componentes de hardware	4
2.2. Componentes externos	4
2.3. Requisitos de conectividade	4
3. Guia de Montagem	5
4. Guia de Instalação	6
5. Guia de Configuração	7
6. Guia de Operação	8
7. Troubleshooting	9
8. Créditos	10

1. Introdução

1.1. Solução (sprint 3)

O dispositivo apresentado se trata de uma solução IOT integrada ao Prisma Estapar, tem como objetivo realizar o monitoramento e controle dos veículos em estacionamentos, para acompanhamento do processo de valet, verificando o estágio do valet bem como seu tempo estimado para conclusão. Outro importante objetivo é a associação do prisma ao manobrista, para que seja possível controlar e entender melhor o trabalho de cada um dos manobristas, criando novos KPIs de desempenho.

A solução em questão depende de 2 tecnologias essenciais, o Wi-Fi e o RFID, o Wi-Fi é necessário para que haja intercomunicação entre os prismas e o sistema do estacionamento, bem como para gerar a localização do veículo. Já a tecnologia RFID é necessária para que seja possível a autenticação e associação do manobrista ao prisma, servindo como chave de acesso para executar os processos do valet. Dessa maneira, o Prisma inteligente permite à Estapar agregar valor ao seu cliente, diminuindo o tempo de espera e dando mais autonomia ao usuário.

1.2. Arquitetura da Solução (sprint 3)

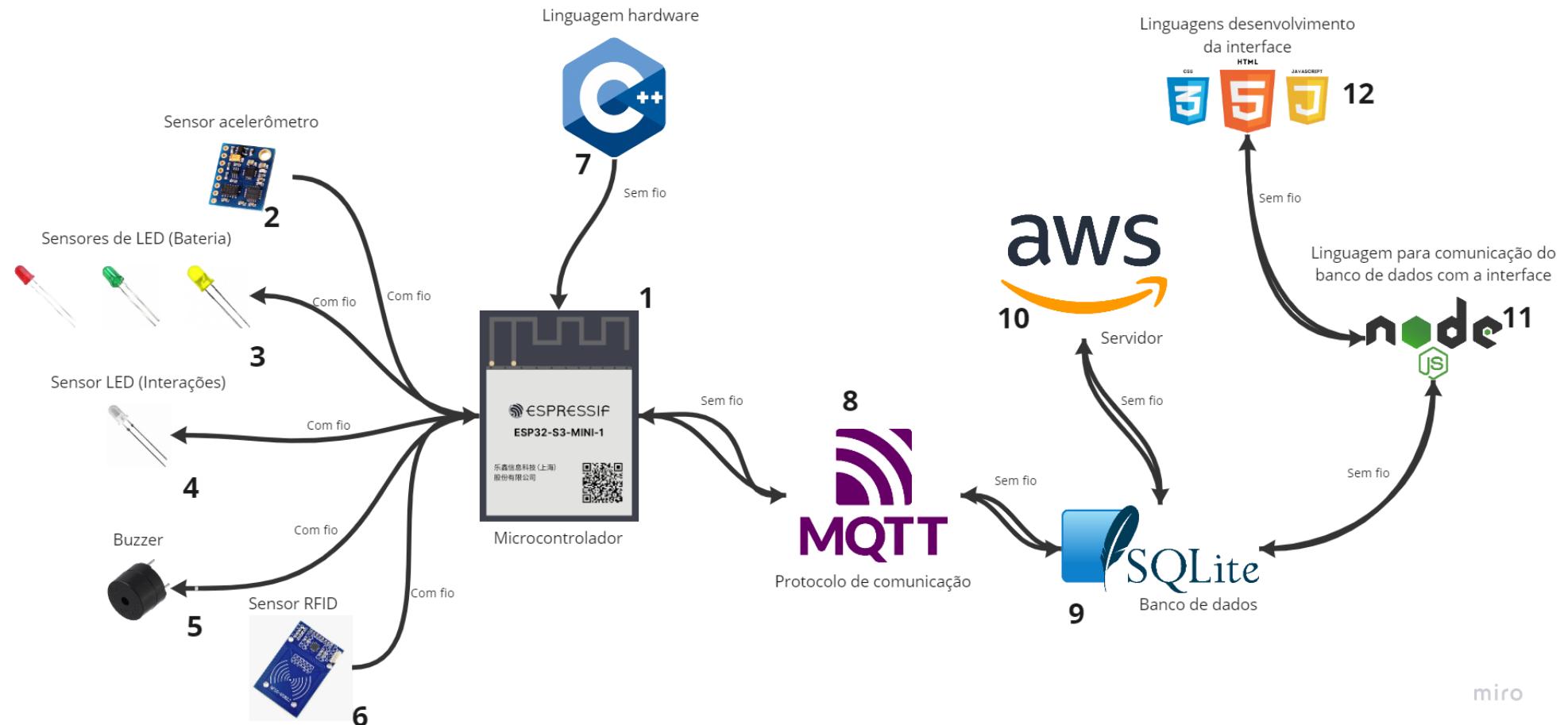
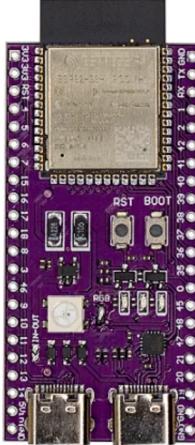


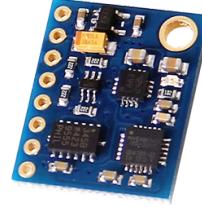
Imagen 1: Arquitetura da solução

2. Componentes e Recursos

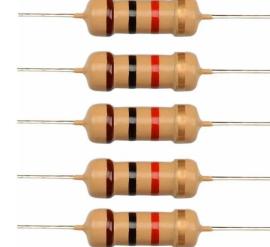
(sprint 3)

2.1. Componentes de hardware

1	Microcontrolador (ESP32 S3)	 A photograph of an ESP32-S3 Dev Board, which is a purple printed circuit board with a central microcontroller chip and various peripheral components.	Será necessário dois ou mais microcontroladores ESP32-S3, para realizar a geolocalização deles no projeto. Ele é um pequeno computador num único circuito integrado o qual contém um núcleo de processador, memória e periféricos programáveis de entrada e saída. Neste projeto ele é responsável por localizar o carro, processar as informações e se comunicar com o banco de dados.
---	-----------------------------	--	---

2	Módulo sensor acelerômetro	 A photograph of a blue printed circuit board module containing a small orange chip and several surface-mount components, labeled as an accelerometer and gyroscope module.	O módulo sensor acelerômetro é um módulo que contém em uma única placa um acelerômetro e um giroscópio tipo MEMS. O sensor envia dados de variação de velocidade para serem processados pelo controlador.
3	Sensores de LED (bateria)	 Three photographs of individual LEDs: one red, one yellow, and one green, arranged vertically.	Os três leds neste projeto irão representar o nível da bateria do dispositivo, a cor verde representará nível de carregamento alto, amarelo o nível médio e o vermelho nível baixo.

4	Sensores de LED (interações)	 <p>Esse led em específico será usado para indicar quando o cartão do manobrista encostou o cartão no RFID</p>	
5	Buzzer	 <p>O buzzer indica também quando o cartão é encostado no RFID, ele emite um sinal sonoro junto com o led branco</p>	
6	Sensor RFID	 <p>O sensor RFID acompanha junto o cartão e um chip para fazer a leitura. Esse sensor será o principal autenticador e leitor do dispositivo, e servirá essencialmente para avançar entre os estados de manobra do veículo.</p>	7 Jumper Macho/Macho e Jumper Macho/Fêmea  <p>Os Cabos Jumper são fios de ligação elétrica, ideais e indispensáveis para quem faz montagens com Protoboard.</p>

8	Protopboard 830 Pontos	 <p>Protopboard, também conhecida como Breadboard, Placa de Ensaio ou Matriz de Contato, é uma placa com furos e conexões pré-definidas, que visa auxiliar a montagem de teste de circuitos eletrônicos experimentais de forma simples e ágil.</p>	10	Resistor	 <p>Resistores para controlar a passagem de corrente elétrica dos leds, para que não queiem.</p>
9	Carregador Portátil Bateria Power Bank	 <p>Um carregador portátil como auxiliar para carregar o outro ESP32-S3 que ficará longe para calcular a distância de um esp para outro.</p>			<p>1- Toten: Para visualizar o tempo de espera para chegar o carro até o dono;</p> <p>2- Computador: Para visualizar a produtividade do manobrista;</p> <p>3- MQTT (MQ Telemetry Transport): Protocolo de comunicação;</p> <p>4- IDE Arduino: Se conecta às placas Arduino para carregar programas e se comunicar com eles;</p> <p>5- AWS: Plataforma de serviços de computação em nuvem;</p> <p>6- SQLite: Biblioteca em linguagem C;</p> <p>7- VSCode: Editor de código;</p> <p>8- Node JS: Um conjunto de bibliotecas que funciona como um interpretador de JavaScript.</p>

2.2. Componentes externos

- 1- Toten: Para visualizar o tempo de espera para chegar o carro até o dono;
- 2- Computador: Para visualizar a produtividade do manobrista;
- 3- MQTT (MQ Telemetry Transport): Protocolo de comunicação;
- 4- IDE Arduino: Se conecta às placas Arduino para carregar programas e se comunicar com eles;
- 5- AWS: Plataforma de serviços de computação em nuvem;
- 6- SQLite: Biblioteca em linguagem C;
- 7- VSCode: Editor de código;
- 8- Node JS: Um conjunto de bibliotecas que funciona como um interpretador de JavaScript.

2.3. Requisitos de conectividade

Rede Wi-fi: O principal requisito para nossa IoT funcionar é uma rede de Wi-fi que funcione bem, pois é por ela que conseguiremos saber a localização de cada carro.

MQTT: O protocolo para as mensagens e os dados serem passados, portanto, eficiente em termos de energia e fácil de implantar para milhões de dispositivos. Conectando dispositivos durante redes não confiáveis. O MQTT em IoT usa níveis de QoS para garantir a entrega garantida de mensagens aos receptores, mesmo quando as conexões entre os dispositivos não são confiáveis.

SQLite: SQLite é uma biblioteca em linguagem C que implementa um banco de dados SQL embutido. Será utilizado para armazenar e gerenciar o Banco de Dados.

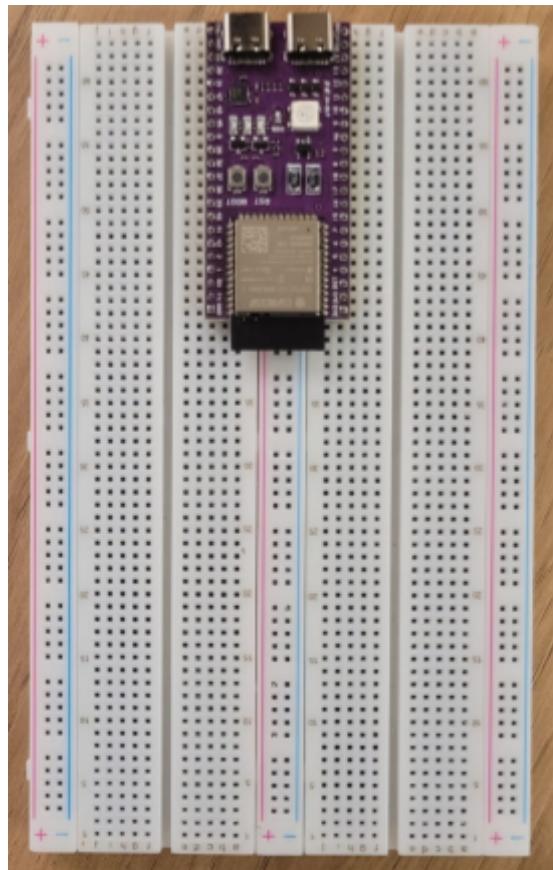
AWS: O Servidor AWS é uma plataforma de serviços de computação em nuvem, que formam uma plataforma de computação na nuvem oferecida pela Amazon. Será utilizado para hospedar o serviço na nuvem.

FTM: O protocolo de medição de tempo fino Wi-Fi (FTM) especificado no padrão IEEE 802.11-2016, fornece uma abordagem de alcance bidirecional para aprimorar a capacidade de posicionamento. Por meio dele conseguimos medir a distância de ESP32 de outro.

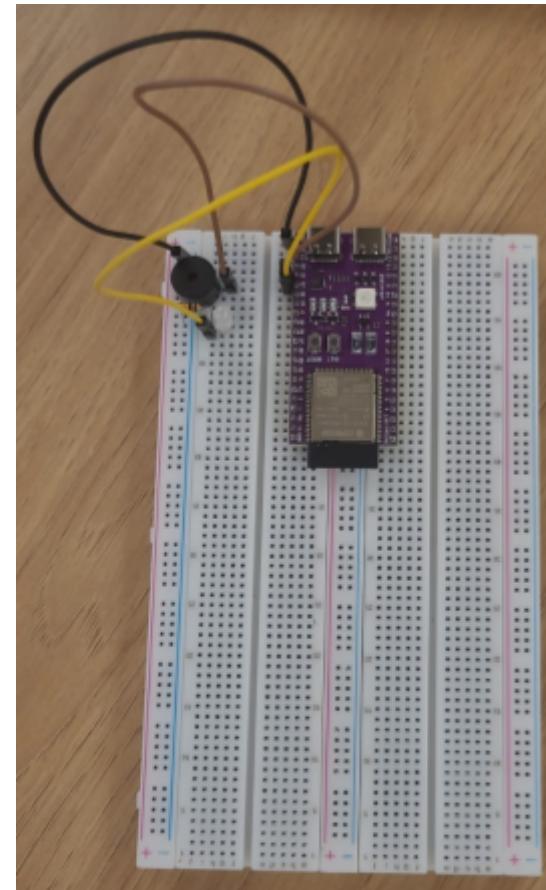
I2C: Significa o controlador integrado. Este é um protocolo de comunicação serial que pode conectar dispositivos de baixa

velocidade . É uma comunicação mestre-escravo na qual podemos conectar e controlar vários escravos a partir de um único mestre. Neste, cada dispositivo escravo possui um endereço específico.

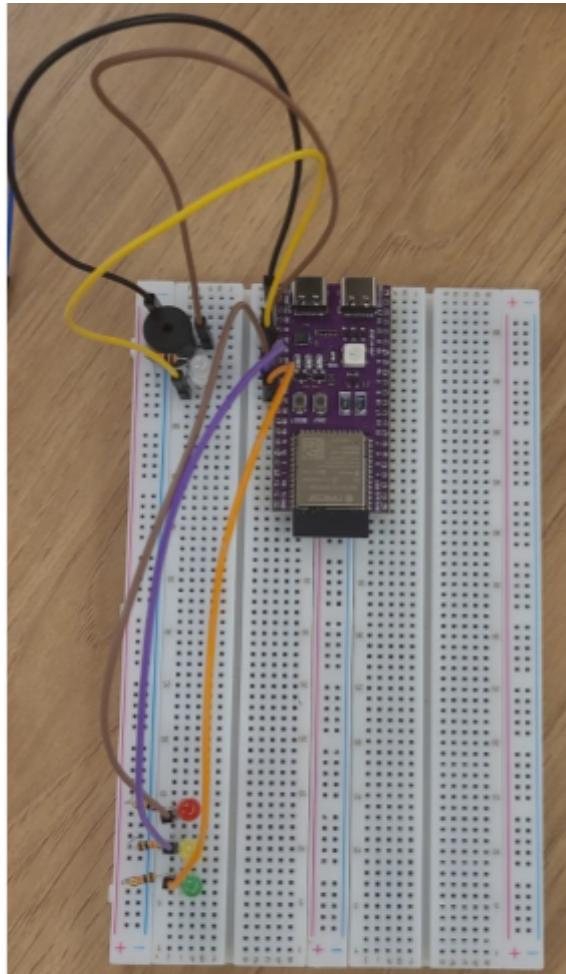
3. Guia de Montagem



Passo 1: Conecte o ESP32-S3 de preferência centralizado na protoboard.



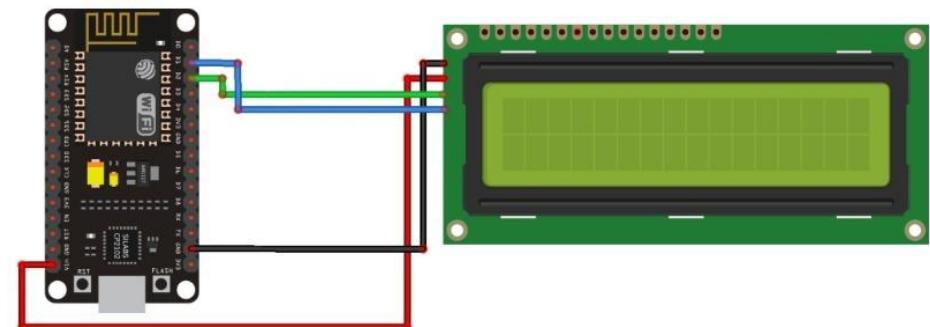
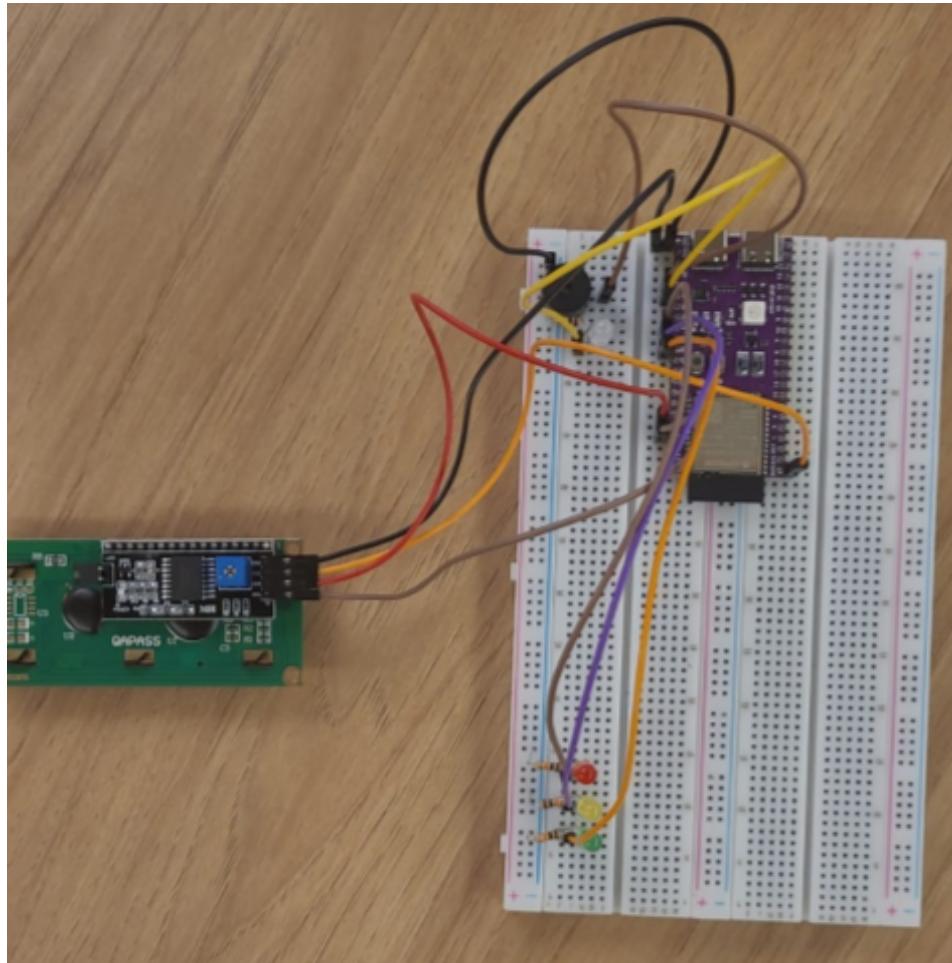
Passo 2: Com o buzzer ligue uma das extremidades do jumper macho na perna negativa do buzzer e com a outra extremidade ligue na porta GND do ESP. Na outra perna do buzzer use outro jumper e ligue em uma das portas do microcontrolador.
Depois ligue o led junto com um resistor.



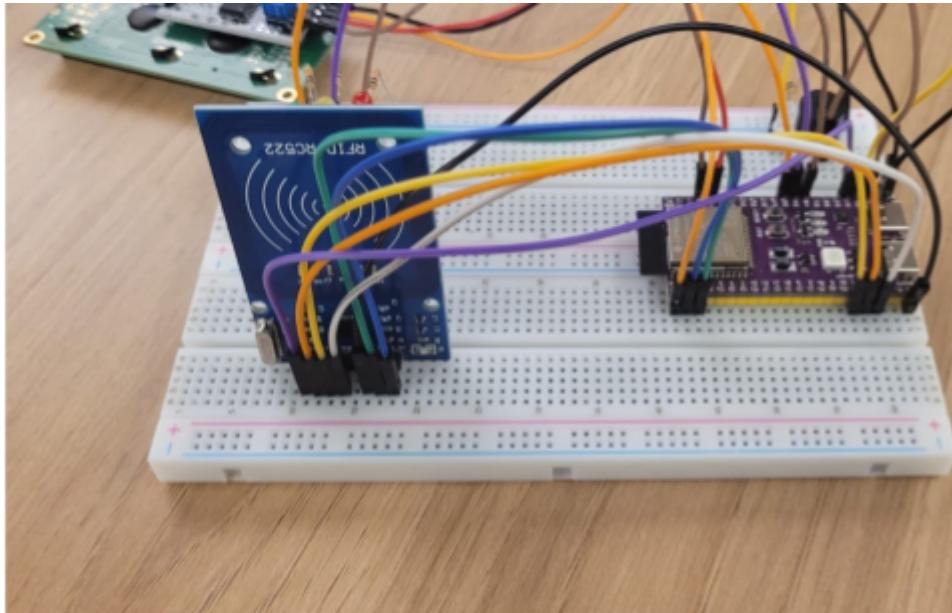
Passo 3: Nessa parte conectamos os três leds que indica o nível de bateria do dispositivo, e os resistores. No lado da perna chanfrada do led coloque o resistor, do outro lado o jumper conectado em uma porta.

Tabela de referência para conexão de cada pino do LCD na porta do ESP:

LCD I2C	ESP32-S3
GND	GND
VCC	VIN
SDA	Pino 2
SCL	Pino 1



Passo 4: Conectando o LCD com os jumpers macho/fêmea nas portas do ESP para mostrar o número do prisma e a distância de um ESP para outro.



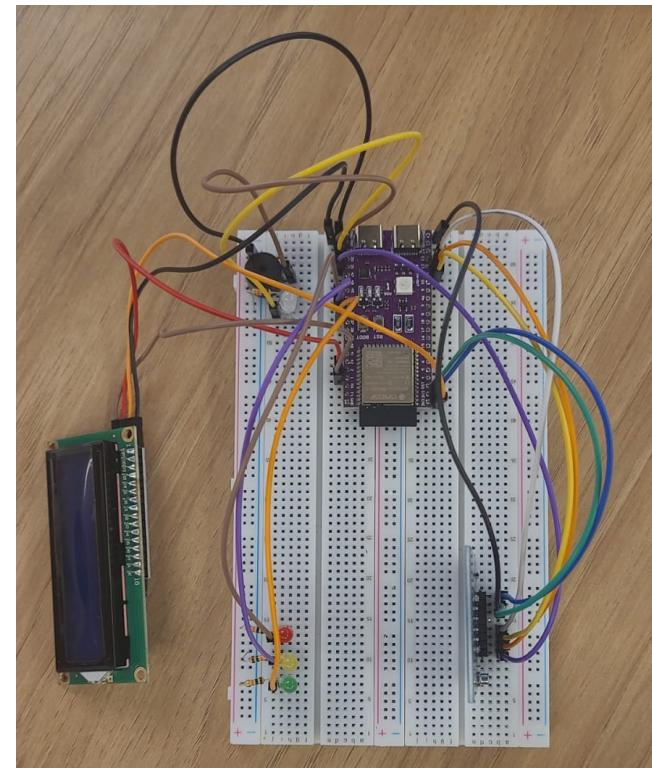
Passo 5: Por último conectamos o RFID ao microcontrolador.

Tabela de referência para conexão de cada pino do RFID na porta do ESP:

RFID	ESP32-S3
GND	GND
VCC	3V3

RST	Pino 17
SDA	Pino 18
SCK	Pino 3
MISO	Pino 10
MOSI	Pino 8

Por fim o seu Hardware se parecerá com isso:



4. Guia de Instalação

(sprint 4)

Descreva passo-a-passo como instalar os dispositivos IoT no espaço físico adequado, conectando-os à rede, de acordo com o que foi levantado com seu parceiro de negócios.

Não deixe de especificar propriedades, limites e alcances dos dispositivos em relação ao espaço destinado.

Especifique também como instalar softwares nos dispositivos.

Utilize fotografias, prints de tela e/ou desenhos técnicos para ilustrar o processo de instalação.

5. Guia de Configuração

(sprint 4)

Descreva passo-a-passo como configurar os dispositivos IoT utilizando os equipamentos devidos (ex. smartphone/computador acessando o servidor embarcado ou a página na nuvem).

Utilize fotografias, prints de tela e/ou desenhos técnicos para ilustrar o processo de configuração.

6. Guia de Operação

(sprint 5)

Descreva os fluxos de operação entre interface e dispositivos IoT. Indique o funcionamento das telas, como fazer leituras dos dados dos sensores, como disparar ações através dos atuadores, como reconhecer estados do sistema.

Indique também informações relacionadas à imprecisão das eventuais localizações, e como o usuário deve contornar tais situações.

Utilize fotografias, prints de tela e/ou desenhos técnicos para ilustrar os processos de operação.

7. Troubleshooting

(sprint 5)

Liste as situações de falha mais comuns da sua solução (tais como falta de conectividade, falta de bateria, componente inoperante etc.) e indique ações para solução desses problemas.

#	Problema	Possível solução
1		

2		
3		
4		
5		

8. Créditos

(sprint 5)

Seção livre para você atribuir créditos à sua equipe e respectivas responsabilidades