



**Desenvolvimento de um dispositivo *ESP32*
com vários sensores e objetivos**

LICENCIATURA EM ENGENHARIA INFORMÁTICA

Professor de UC: Pedro Rosa
Professor Orientador: André Sabino

André Custódio, 20220112

André Mendes, 20220355

Steve Vilas, 20200856

22/05/2025

O relatório encontra-se em condições para ser apresentado

Ciclo de Formação 2022/2025
Ano Letivo 2024/2025

Descrição do problema a resolver

Devido ao aquecimento global, muitas companhias necessitam de calcular e medir a temperatura em vários espaços ao mesmo tempo. Devido a uma grande afluência de medição de temperatura e falta de recursos humanos, apresentamos uma solução mais eficiente, mais barata e totalmente remota.

Projecto: S.T.A.R

Descrição do Projeto: Surveillance & Tactical Autonomous Rover (Sistema Terrestre de Análise e Reconhecimento) ou o seu acrónimo S.T.A.R é um veículo com a capacidade de ajudar o homem em algumas tarefas difíceis ou impossíveis. Este veículo irá possibilitar ajudar meteorologistas em algumas atividades como, transporte de material de locais perigosos, medir temperatura do espaço, gravar o som do espaço, entre outras tarefas mais pequenas. Estes valores irão estar todos recebidos pelo ESP32 e visualizados numa dashboard.



O Grupo alvo é companhias meteorológicas de Portugal, que está quantificado em 90 estações no total. O nosso grupo alvo necessita de um veículo que possa ser controlado remotamente, que tenha câmaras para visualizar o espaço, que tenha sensores para ler dados concretos e que tenha a capacidade de transportar objetos.

Descrição genérica da solução a implementar

Para o nosso veículo estar bem estruturado e preparado para os desafios precisamos que ele não colida contra paredes, consiga subir obstáculos, virar, entre outros desafios. Para tal, iremos precisar de:

- Microphone Sensor;
- Temperature Sensor;
- Clock sensor;
- Sound Sensor;
- Flame Sensor;
- Lazer sensor;
- Common Cathode Led;
- Seven-Color Automatic Flashing LED;
- Passive Buzzer;
- Suporte de Bateria;
- Converter modulo Blinghe;
- Sensor de Humidade;
- 4 peças de Geared Motoro DC3V-12V;
- DC Motor Driver Board Drive;
- ESP32;
- BreadBoard;
- Active *Buzzer*;

Ao longo do tempo está lista poderá ser modificada. Mas para já irá ser esta a nossa lista de peças ligadas ao Arduino necessárias.

Enquadramento nas áreas da Unidade Curricular

- Compiladores – com a cadeira de compiladores, trouxe-nos uma grande abertura e aprendizagem como estruturar o código e pensar em *LowLevel*.
- Computação Física e IOT – em IOT trouxe uma abertura para aprendermos como os vários componentes IOT funcionam e métodos de comunicação como MQTT.
- Empreendedorismo – Base Tecnológica – graças a cadeira de empreendedorismo, ganhamos aptidões para saber vender bem o produto ao cliente e fazer um sell-in, sell-out e benchmarking.
- Ética e Deontologia Profissional – Ética deu aptidão de conhecer o que está certo e errado do nosso projeto e como podemos adaptar sem quebrar as questões morais e éticas.
- Project Factory – Conseguimos adiantar o projeto
- Sistemas Operativos – aprendemos como as threads funcionam e como podem ser importantes para o projeto e para o SO.

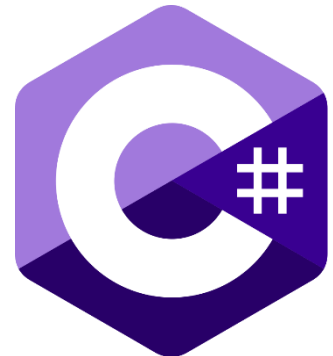
Linguagem de Programação

C#

Microsoft em 2000 criou o C#, uma linguagem simples, moderna, orientada por objetos, flexível e versátil. É semelhante ao C++ e Java, só em 2002 foi lançada para a comunidade.

As suas implementações mais utilizadas são .Net Framework que utiliza o form (Framework) e .Net Core que utiliza a linha de comandos. É uma linguagem utilizada em jogos, aplicações de clientes, aplicações webs, inteligência artificial e muitos mais.

A comunidade tem dado uma grande ajuda, na evolução da linguagem e na criação de bibliotecas, que auxiliam na codificação da aplicação.



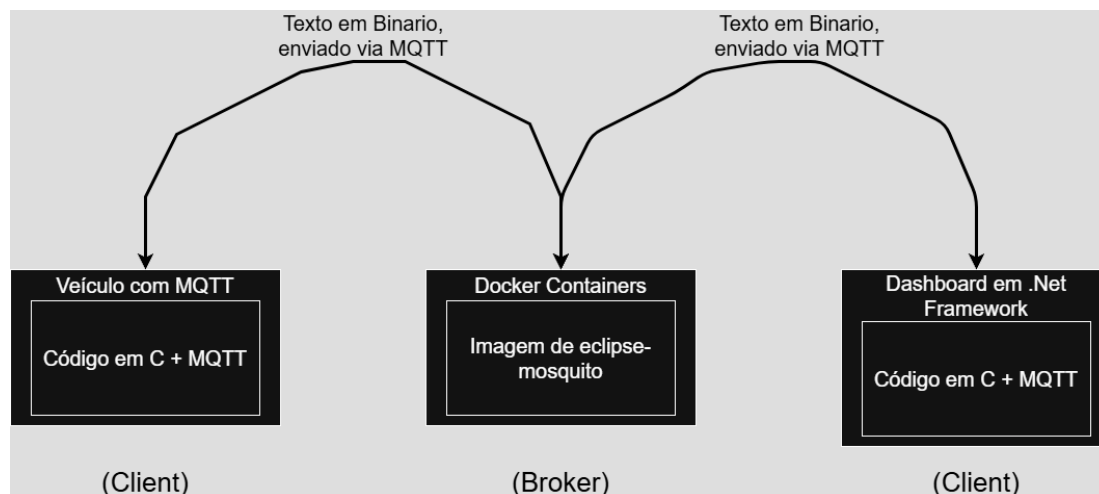
C

O Sr. Dennis MacAlistair Ritchie foi o criador da linguagem C, uma linguagem que tem suporte a *structured Programming*, *lexical variables* e recursão.

Originalmente a linguagem foi pensada para o desenvolvimento de sistemas operativos, incluindo o *Unix*. Atualmente a linguagem continua a ser usada, devido a ser uma linguagem de baixo nível, fazendo uma codificação próxima do hardware.



Arquitetura da Solução



A arquitetura utilizada no nosso projeto, esta dividida em dois componentes:

- ❖ *Broker*
- ❖ *Cliente*

Na componente de *broker* é usado para comunicar com o cliente do C# e do cliente do ESP32. O broker está a usar uma imagem do "eclipse-mosquitto", em *Docker-Container*.

Tanto o veículo como a *dashboard* comunicação via *MQTT* via wireless na internet local e os dados são enviados em binário. Devido a estarmos a falar de um hardware mais fraco, relativamente ao veículo, teríamos de então optar com esse formato.

O código do *ESP32* é enviado via flash programado em *C* no *Arduino IDE*, enquanto o cliente *c#* é feito no *IDE Visual Studio 2022* com a Framework *.Net Framework*!

Tecnologias a Utilizar

MQTT

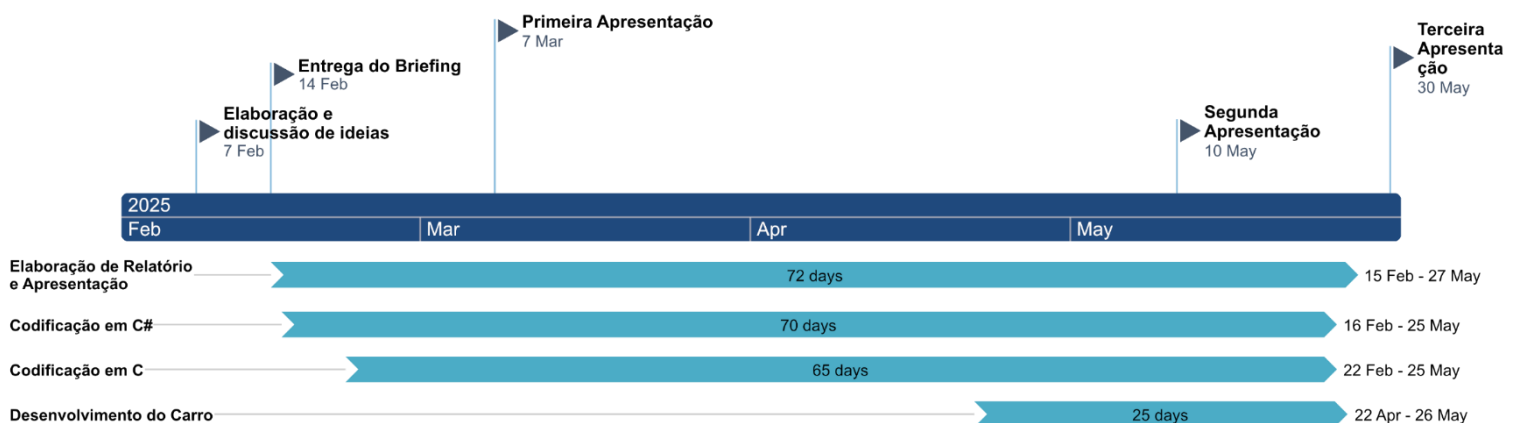
O MQTT foi criado pela IBM em 1999 como um protocolo de comunicação leve, eficiente e ideal para comunicações em redes instáveis ou com pouca largura de banda. Baseado no modelo *publish/subscribe*, permite que dispositivos troquem mensagens através de um *broker*, sem necessidade de conexão direta entre eles.

É muito utilizado em sistemas de *IoT* (Internet das Coisas), automação residencial, sensores remotos e aplicações que exigem comunicação em tempo real. O protocolo é simples, rápido e consome poucos recursos, o que o torna ideal para dispositivos com capacidade limitada.



Cronograma

Distribuição de Tarefas



A tarefa de trabalhar no *C* é o Steve Vilas, o André Mendes fica responsável pela parte do *arduino* perceber como os componentes e o circuito funciona e o André Custódio fica responsável pela execução do código em *C#*. Ambos os elementos ficam responsáveis pelo relatório, Planeamento semanal e ajuste no desenvolvimento da estrutura do veículo!

Protobuf e *MQTT* foi executado pelo André Custódio

Relatório de Ética, soldagem de componentes e apoio moral ficou responsável pelo Steve Vilas.

Codificação *MQTT* e construção do veículo foi efectuada pelo André Mendes.

Bibliografia e Web Grafia

Inspirações de Projeto:

- <https://www.instructables.com/Carro-Telecomandado-Arduino/>
- <https://pplware.sapo.pt/tutoriais/arduino-robot-controlado-por-movimentos-do-telemovel/>
- <https://www.linkedin.com/pulse/what-programming-languages-does-nasa-use-analytics-insight-p5nrc>
- <https://science.nasa.gov/mission/mars-2020-perseverance/rover-components/>

Curso Prático de C# do Autor Paulo Capela Marques