**RELATÓRIO DO PROJETO DE**

**PROJECT FACTORY**

**LICENCIATURA EM ENGENHARIA INFORMÁTICA**

Professor Orientador: André Sabino

André Custódio, 20220112

André Mendes, 20220355

Steve Vilas, 20200856

15/02/2024

O relatório encontra-se em condições para ser apresentado

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ciclo de Formação 2022/2025  
Ano Letivo 2024/2025

# Agradecimentos

…

# Índice

[Agradecimentos 2](#_Toc190614713)

[Índice 3](#_Toc190614714)

[Índice de Imagens 5](#_Toc190614715)

[Introdução 6](#_Toc190614716)

[Capítulo I – Cronograma Inicial 7](#_Toc190614717)

[Descrição do Cronograma 8](#_Toc190614718)

[Capítulo II – Conceção do Projeto 9](#_Toc190614719)

[Objetivos 9](#_Toc190614720)

[Tecnologias 10](#_Toc190614721)

[Framework 10](#_Toc190614722)

[O que é um framework? 10](#_Toc190614723)

[Vantagens da Framework 10](#_Toc190614724)

[API 11](#_Toc190614725)

[Peças para o Arduíno 11](#_Toc190614726)

[Linguagem de Programação 12](#_Toc190614727)

[C# 12](#_Toc190614728)

[JavaScript 12](#_Toc190614729)

[C 12](#_Toc190614730)

[Recursos Necessários para o Programa 13](#_Toc190614731)

[Visual Studio Code 13](#_Toc190614732)

[Visual Studio 2022 13](#_Toc190614733)

[Arduino IDE 14](#_Toc190614734)

[Codificação 15](#_Toc190614735)

[Bibliotecas instaladas (Dlls) 15](#_Toc190614736)

[Capítulo III – O Projeto 16](#_Toc190614737)

[Arquitetura do Programa 17](#_Toc190614738)

[Planeamento do Projeto 18](#_Toc190614739)

[Recursos Utilizados 19](#_Toc190614740)

[Ferramentas de Desenvolvimento: 21](#_Toc190614741)

[Ferramentas para desenvolvimento de apresentação e relatório: 21](#_Toc190614742)

[Aplicação/Site de Comunicação: 21](#_Toc190614743)

[Capítulo IV – Cronograma Final e Justificação de desvios 22](#_Toc190614744)

[Justificação 23](#_Toc190614745)

[Capítulo V – Análise do percurso pessoal 24](#_Toc190614746)

[Conclusão 25](#_Toc190614747)

[Bibliografia e Web Grafia 26](#_Toc190614748)

# Índice de Imagens

[Imagem 1 - Cronograma inicial 7](file:///E:\GitHub\S.T.A.R\Resource\Relatorio%20-%20STAR%20-%201.docx#_Toc190614702)

[Imagem 2 - Exemplo de veículo 16](#_Toc190614703)

[Imagem 3 - Visual Studio Code 19](#_Toc190614704)

[Imagem 4 *- Visual Studio 2022* 19](#_Toc190614705)

[Imagem 5 - Arduíno IDE 19](#_Toc190614706)

[Imagem 6 - Microsoft Word 19](#_Toc190614707)

[Imagem 7 - Microsoft Excel 20](#_Toc190614708)

[Imagem 8 - Microsoft Power Point 20](#_Toc190614709)

[Imagem 9 - Brave 20](#_Toc190614710)

[Imagem 10 - Discord 20](#_Toc190614711)

[Imagem 11 - GitHub 20](#_Toc190614712)

# Introdução

# Capítulo I – Cronograma Inicial

Imagem 1 - Cronograma inicial

## Descrição do Cronograma

O desenvolvimento deste trabalho irá decorrer entre os dias 7 de fevereiro a 29 de maio de 2025.

Elaboração de Relatório final e apresentações começou no dia 15 de fevereiro a 20 de maio. A testagem de Mono com C# começou dia 16 de fevereiro a 21 de maio de 2025.

Começo da codificação em Javascript foi 17 de fevereiro a 20 de maio. A codificação C com o Arduíno IDE foi de 22 de fevereiro a 21 de maio de 2025.

Durante o desenvolvimento deste projeto irá haver três momentos de avaliação, o primeiro está agendado para 7 de março de 2025, a segunda avaliação está marcada para 11 de abril de 2025 e para a terminar a terceira e última avaliação está marcada para 29 de maio de 2025.

# Capítulo II – Conceção do Projeto

**Projecto**: *S.T.A.R*



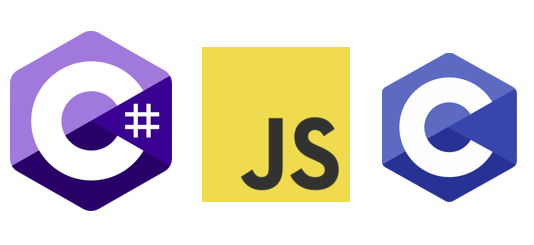
**Descrição do Projeto**: *Surveillance & Tactical Autonomous Rover* (Sistema Terrestre de Análise e Reconhecimento) ou o seu acrónimo *S.T.A.R* é um veículo com a capacidade de ajudar o homem em algumas tarefas difíceis ou impossíveis. Este veículo irá possibilitar ajudar a polícia e meteorologistas em algumas atividades como, resgate de reféns, transporte de material para locais perigosos, visualizar o espaço, medir temperatura do espaço, entre outras tarefas mais pequenas.

Estes valores irão estar todos recebidos pelo *ESP32* e visualizados numa *dashboard*.

## Objetivos

* Ambiente de utilização
  + Ouvir o que se passa no espaço;
  + Transporte de material para locais perigosos;
  + Medir a temperatura do espaço;
  + Sensor de Fumo;
  + Reproduzir som;
  + Medir Humidade no espaço;
* Rapidez e fluidez
  + Ser rápido e responsivo
  + Ter uma interface apelativa e intuitiva para os utilizadores, facilitando o uso de quem está a usar.

## Tecnologias

O nosso projeto utiliza algumas *Framework* (*.Net Framework* e *Express*) e três linguagens de programação (*C#*, *JavaScript* e *C*).

## Framework

### O que é um framework?

Um framework consiste numa abstração que une códigos entre vários projetos de software,

fornecendo uma funcionalidade sem pormenores. Trata-se de um conjunto de classes que

colaboram para realizar uma responsabilidade para um domínio de um subsistema da aplicação.

Em vez de ser as bibliotecas a controlar é o framework quem dita o controlo da aplicação.

### Vantagens da Framework

A framework é rápida no seu debug, na solução do projeto. A linguagem de programação ou o

Software Development Kit evolui com ajuda da comunidade ou das empresas, tem uma grande

auxílio em fóruns caso os programadores tenham dúvidas.

A comunidade quando deteta um problema de segurança é rapidamente atualizada e modificado

esse problema.

### API

Application Programing Interface ou API é um conjunto de protocolos, rotinas e ferramentas que permitem que múltiplos códigos Backend consigam aceder a mesma, independentemente da sua linguagem de programação. Podemos ter um codigo Javascript e C# que irão conseguir fazer o request sem qualquer problema, se tiver com os parâmetros certos!

É fácil de fazer manutenção e é segura, pois não existe uma exposição dos dados não desejados da Base de Dados. A API faz uma comunicação com a base de dados permitindo assim uma segurança maior e fiabilidade dos dados, não havendo inserção, modificação ou eliminação de dados. Os pedidos da API referem-se principalmente a usar o protocolo HTTP da maneira que foi pretendido usado os requests GET, POST, PUT, DELETE, OPTIONS, PATCH e HEAD.

(Mostrar um codigo de NODEJS)

# Peças para o Arduíno

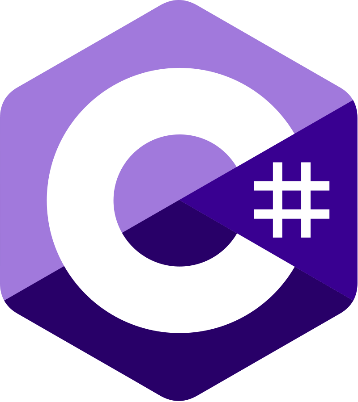
Para o nosso veículo estar bem estruturado e preparado para os desafios precisamos que ele não colida contra paredes, consiga subir obstáculos, virar, entre outros desafios. Para tal, iremos precisar de:

* Microphone Sensor;
* Temperature Sensor;
* Clock sensor;
* Sound Sensor;
* Flamer Sensor;
* Lazer sensor;
* Common Calthode 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 Arduíno necessárias.

## Linguagem de Programação

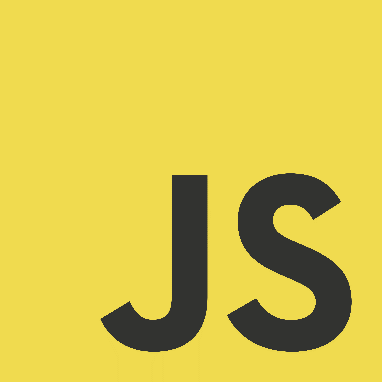
### C#

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.

### JavaScript

No meio de 1990, as páginas web eram estáticas e muitas limitações interativas. Devido a esse problema, a comunidade Netscape decidiram criar uma linguagem que mudasse a experiência dos clientes que naveguem na web. Brendan Eich aceitou o desafio e em 1995 criou o seu protótipo com DOM (Document Object Model) e chamou “Mocha”, mas depois decidiu mudar para “LiveScript” e no fim ficou “JavaScript”.

A linguagem foi inspirada em Java, AWK, HyperTalk, Scheme e Self.

A comunidade tem estado a evoluir a linguagem com correções, bibliotecas e implementações únicas, para ajudar outros programadores.

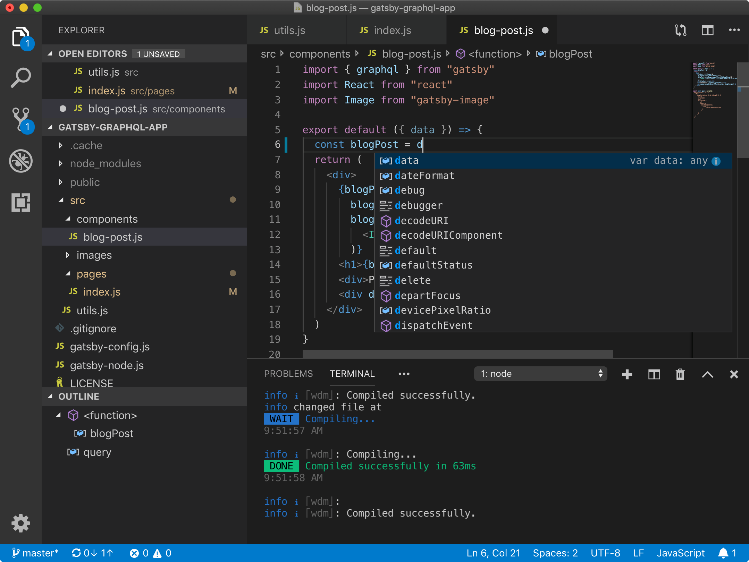
### GitHub - rathod-tirth/C_Language: It contains C programming files C

O Sr. Dennis MacAlistair Ritchie foi o criador da linguagem c, uma linguagem que tem suporte a structured Programing, 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.

## Recursos Necessários para o Programa

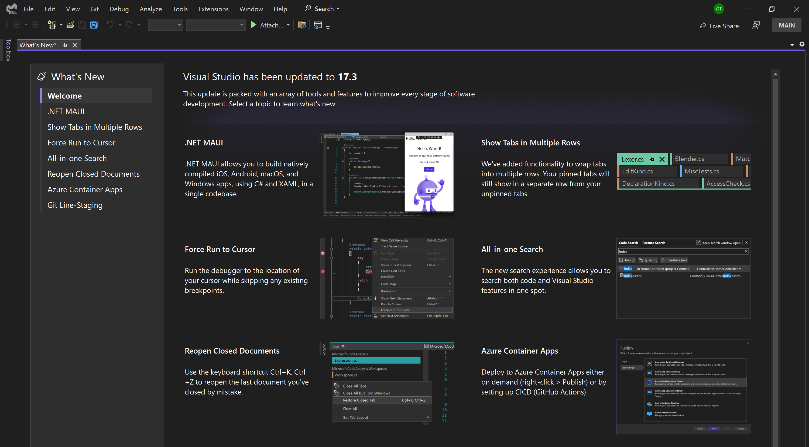
### Visual Studio Code

Visual Studio Code é um editor de código fonte, possível usar em todos os sistemas operativos. O Visual Studio, contem extensões para ajudar o utilizador a programar ou a publicar o seu código, o exemplo é o GitLens que envia os projetos para o GitHub.

Foi desenvolvida pela Microsoft e foi programado com o TypeScript, JavaScript e Css.

Algumas linguagens dependem de alguns recursos, um deles é a depuração, processo que tenta encontrar erros, tanto no hardware ou software.

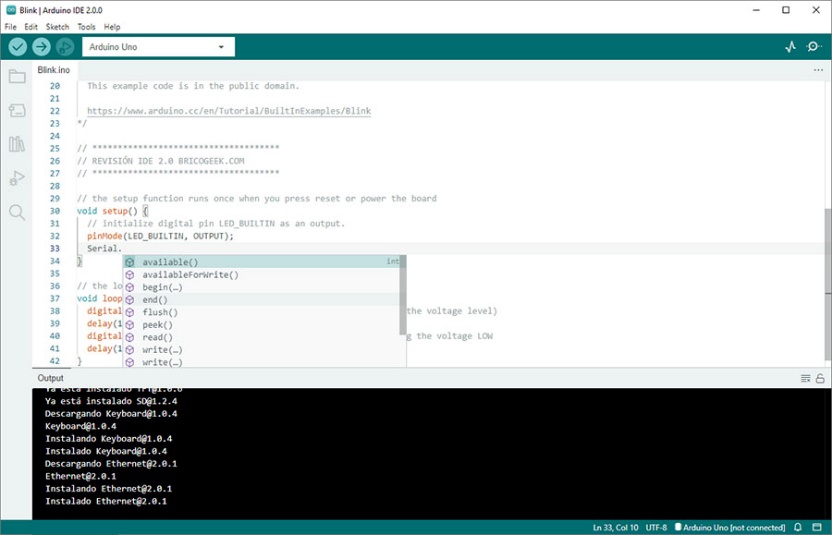
### Visual Studio 2022

O Visual Studio 2022 é uma aplicação desenvolvida pela Microsoft, a sua primeira versão foi em 1997, Visual Studio 97. É uma IDE, ambiente de desenvolvimento integrado que auxilia na criação dos objetos e na sua localização.

A mesma contem Visual Basic, C, C++, C#, F#, JavaScript, Python, Type Script e outras mais. Este contem templates .Net Framework, .Net Core, Asp .Net e muitos mais.

O Visual Studio tem muitas boas vantagens, desde auxílio na codificação, rápido debug e possível colaboração de trabalho entre colegas.

### Arduino IDE

O Arduino IDE (Integrated development environment) é um editor de codigo fonte, dedicada ao desenvolvimento. Este editor ajuda a enviar o codigo em flash para os arduíno que tiverem ligados ao computador do programador. Este editor foi feito pela Arduíno Software, disponibilizado aos clientes no ano 2021.

A mesma originalmente foi escrita em Java, C e C++, mas a versão mais atualizada (20 de fevereiro de 2024) está escrita em TypeScript, JavaScript e Go. A versão mais atual contem nova gestão de boards, nova gestão de bibliotecas, novo explorador de projetos, dark mode e suporte a 64 bits.

Atualmente este IDE encontra-se disponibilizado para os vários sistemas operativos, como Windows, Mac e Linux.

## Codificação

…

## Bibliotecas instaladas (Dlls)

…

# Capítulo III – O Projeto

## Projetos que nos inspiraram

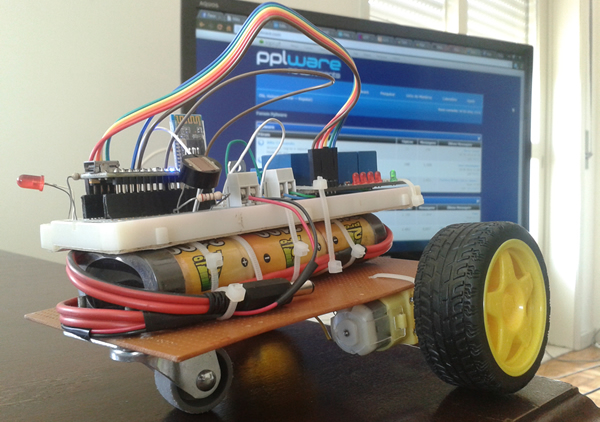
Para sentir-nos inspirados e pensarmos na estruturar do nosso projeto, pesquisamos alguns trabalhos já existentes na internet e encontramos 3 projetos similares a nossa ideia. Componentes de arduíno que detetam valores e enviam para o Software do computador, com algumas ideias diferentes e alguns ajustes que o nosso projeto irá ter.

### Ideia 1

Pequeno projeto realizado por um estudante do 12º ano do curso técnico de GPSI (Gestão e Programação de Sistemas Informáticos), que o estudante fez um veículo telecomandado com a adição de um maquinismo que faz parar, quando o veículo se encontra em risco de impacto frontal. Este projeto tem Arduíno, Bluetooth, breadboard, chassis robótico, Driver Motores, Suporte de pilhas 9V, cabos jumper e muito mais.

Este projeto irá ter uma arquitetura parecida a nossa, usar peças parecidas a nossa e também tem usa a mesma linguagem que a nossa (C).

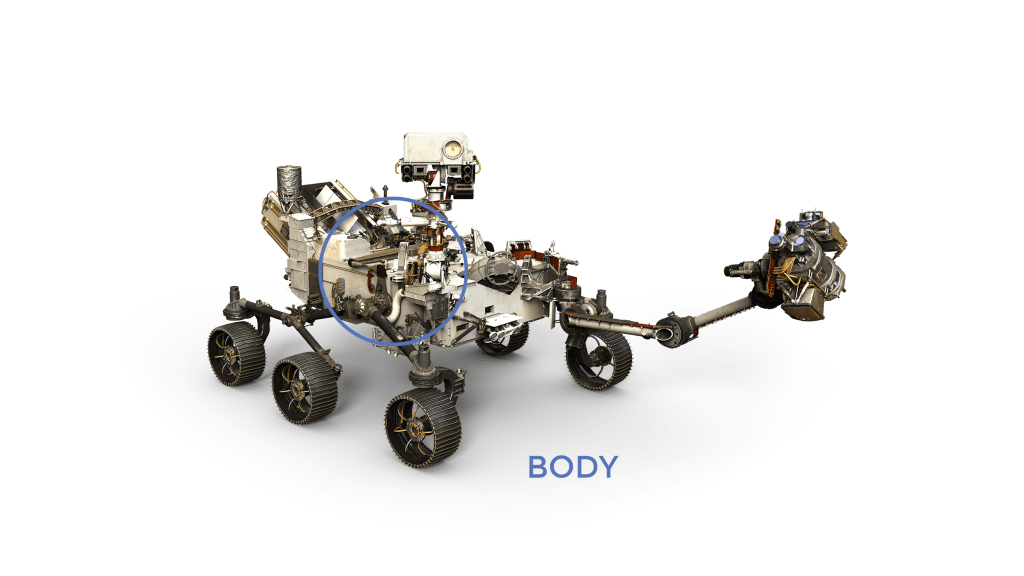
### Ideia 2



O outro projeto que nos inspirou foi este do site pplware, que faz analise da temperatura e essa leitura é enviada para uma aplicação do telemóvel. Essa aplicação também tem uma funcionalidade adicional de controlar o carro.

As características deste veículo são Arduino, Base de 4 relés, Breadboard, Bateria 7.2V-2100mA, Roda livre, LDR, entre outros.

Ideia 3

O outro e último projeto que nos inspirou foi o robô da NASA. Apesar de o nosso projeto ser um “pouco mais amador” comparado ao da NASA, achamos interessante apresentar aqui algumas ideias desde linguagens, peças, programas, desafios, etc. Ambos os veículos, deslocam-se autonomamente ou manualmente e ambos recolhem dados meteorológico para ser analisado.

Apesar de a NASA, ser uma empresa grande e ter muito dinheiro, achamos sensato analisar detalhadamente os seus documentos e desafios, pois os “problemas” passados deles, podem ou poderão ser os nossos problemas de hoje, para ter sucesso na entrega deste projeto.

A NASA usa no seu robô Python para cálculo e simulação e juntamento tem Assembly incorporado para fácil controlo no robô evitando glitches e/ou erros.

## Ideia do nosso projeto

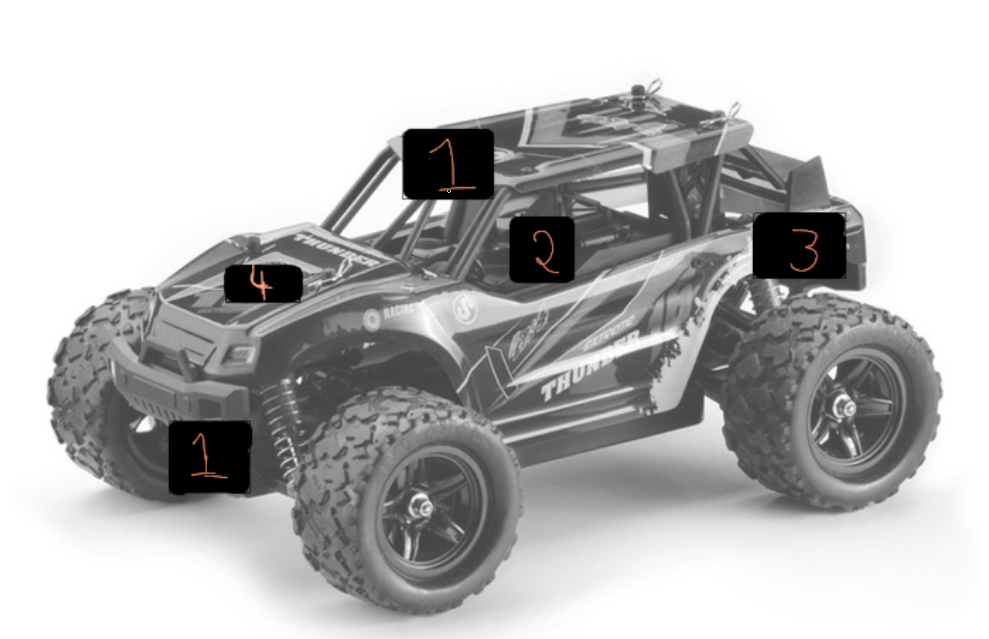
A ideia do grupo é fazer um carro com uma boa estrutura para andar em todo o terreno e conseguir passar vários obstáculos e vários terrenos como por exemplo este veículo: 

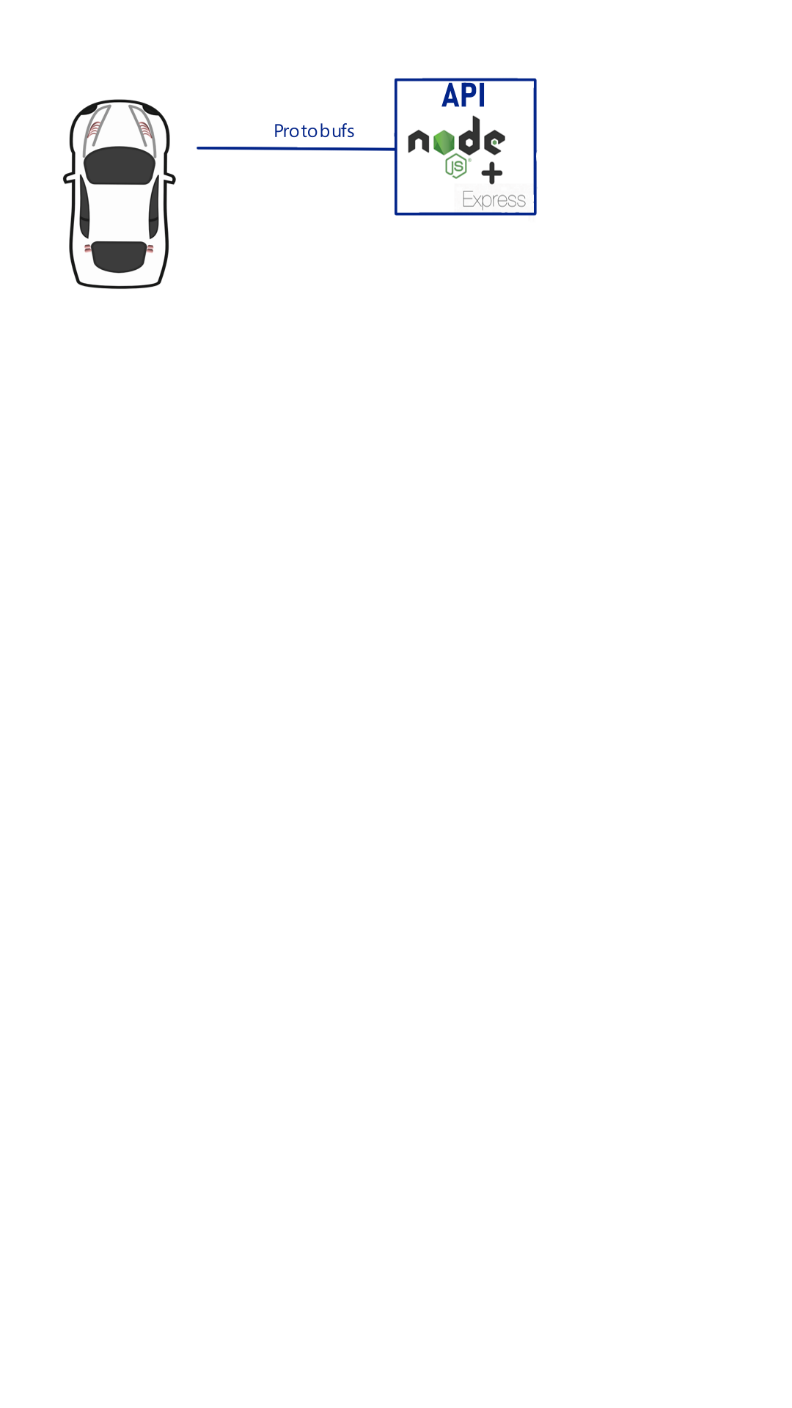
Imagem 2 - Exemplo de veículo

O veículo iria ter 2 sensores de distância para calcular a distância (número 1), uma câmera para visualizar o ambiente (número 2), uma zona onde ligar o arduíno ao computador (número 3), sensor de calcular a temperatura do espaço (número 4) e entre o número 1 e 2 iria haver uma caixa para transportar os objetos.

Como é obvio, está imagem não é o resultado final do carro pretendido, mas é uma pequena ideia do que poderá aparecer no nosso projeto final. Irá depender do orçamento, dificuldades que encontremos ao desenvolver o projeto PBL e ideia do grupo de designers.

## Arquitetura do Programa

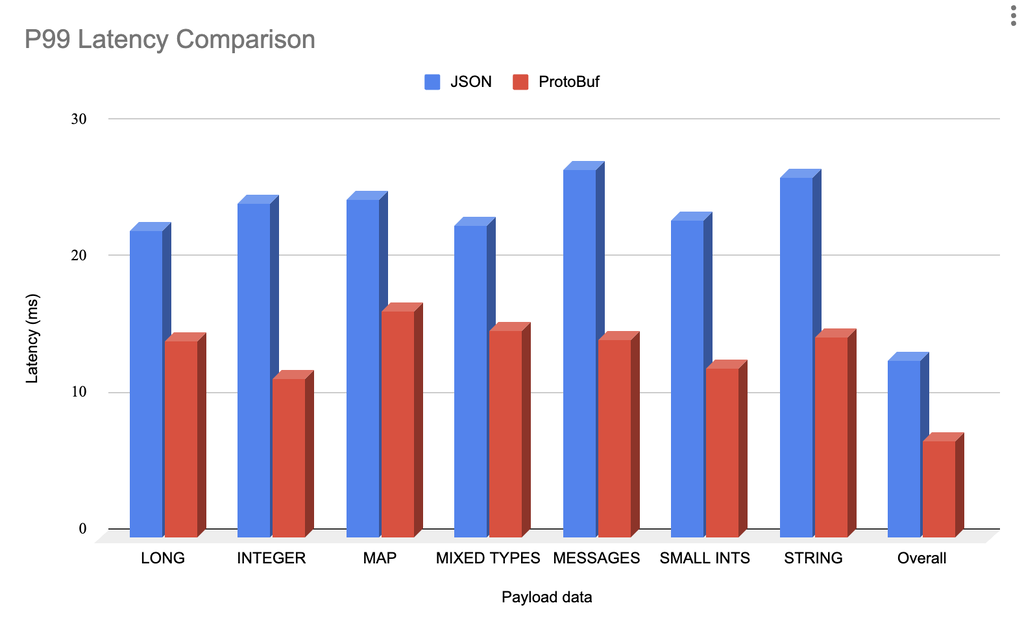




Neste projeto existe dois tipos de arquitetura possíveis. Haver uma comunicação com o projeto em C# com a framework .Net Framework e usando a biblioteca TPCCliente para comunicar com o carro ou então em node.js com a framework express para simular uma API.

Em ambas as soluções iriam ser enviados os dados em formato Protobufs, novo data-format muito mais eficiente e melhor que o json. Devido a estarmos a falar de um hardware mais fraco, relativamente ao veículo, teríamos de então optar com esse formato.

Poderão ver abaixo a latência de um JSON em comparação com um ProtoBuf:



## Planeamento do Projeto

…

# Recursos Utilizados

|  |  |
| --- | --- |
| Using VS Code Tasks to Create Template Files | gitconnected  Imagem 3 - Visual Studio Code | **Visual Code**– Ambiente de desenvolvimento da Microsoft para o desenvolvimento de software. Foi utilizado para desenvolver uma parte do projeto usando a linguagem Javascript no ambiente nodejs. |
| kolcasino - Blog  Imagem 4 *- Visual Studio 2022* | **Visual Studio 2022** – Ambiente de desenvolvimento da Microsoft para o desenvolvimento de software. Foi utilizado para desenvolver uma parte do projeto usando a linguagem c# com a .Net Framework |
| Arduino IDE | ElCoM  Imagem 5 - Arduíno IDE | **Arduíno IDE –**Ambiente de desenvolvimento da Arduíno Software, para o desenvolvimento e configuração do nosso veículo. Foi utilizado para desenvolver uma parte do nosso projeto usando a linguagem c. |
| Microsoft Word Logo: valor, história, PNG  Imagem 6 - Microsoft Word | **Word *–*** Foi utilizado para realizar este relatório e relatório da cadeira de Sistemas Operativos. |

|  |  |
| --- | --- |
| Microsoft Excel - Apen Informática  Imagem 7 - Microsoft Excel | **Excel** – Foi utilizado para fazer o TODO list e usar o registo das tarefas semanais. |
| Microsoft PowerPoint Logo PNG vector in SVG, PDF, AI, CDR format  Imagem 8 - Microsoft Power Point | **Power Point** – Utilizado para concecção das apresentações Referentes ao PBL. |
| The Brave browser basics - what it does, how it differs from rivals ...  Imagem 9 - Brave | **Brave –** browser que permitiu navegar na internet que utilizamos para pesquisar informação e esclarecer dúvidas. |
| Discord Logo, symbol, meaning, history, PNG, brand  Imagem 10 - Discord | **Discord** - Foi utilizado para comunicar com os colegas de projeto e comunicar com os docentes das cadeiras envolvidos no PBL. |
| GitHub Logo, symbol, meaning, history, PNG, brand  Imagem 11 - GitHub | **GitHub** – Utilizado para que o projeto possa ser acedido por qualquer programador que tenha acesso ao repositório para que possa consultar ou contribuir no mesmo. |

## Ferramentas de Desenvolvimento:

* Visual Studio Code;
* Visual Studio 2022;
* Arduino IDE;

Browser:

* Brave;
* Opera;
* Google Chrome

## Ferramentas para desenvolvimento de apresentação e relatório:

* Microsoft Office Power Point 365
* Microsoft Office Word 365;
* Microsoft Office Excel 365;
* Office Timeline;
* Draw.io;

## Aplicação/Site de Comunicação:

* Discord;
* Whatsapp;
* Git;
* GitHub;
* Gmail;

# Capítulo IV – Cronograma Final e Justificação de desvios

# Justificação

# Capítulo V – Análise do percurso pessoal

# Conclusão

# Bibliografia e Web Grafia

Projeto 1:

- <https://www.instructables.com/Carro-Telecomandado-Arduino/>

Projeto 2:

- <https://pplware.sapo.pt/tutoriais/arduino-robot-controlado-por-movimentos-do-telemovel/>

Projeto 3:

- <https://www.linkedin.com/pulse/what-programming-languages-does-nasa-use-analytics-insight-p5nrc>

- <https://science.nasa.gov/mission/mars-2020-perseverance/rover-components/>