



## Problema #3 – IoT: A Internet das Coisas

### 1. Tema

Protótipo de um sistema IoT.

### 2. Objetivos de Aprendizagem

Ao final da realização deste problema, o/a discente deverá ser capaz de:

- Entender como integrar sensores com aplicações através do protocolo MQTT;
- Compreender o uso e funcionalidades do protocolo MQTT;
- Assimilar conceitos básicos sobre integração de sistemas.

### 3. Contexto

Febre. Moda. Hype. Qualquer uma das três palavras descreve bem o cenário do conjunto de tecnologias conhecida como Internet das Coisas (IoT, do inglês, *Internet of Things*). Uma profusão de plataformas em nuvem, sensores, hardware diversos e módulos vêm aparecendo, gerados pelos mais diversos fabricantes. Intel, Microsoft, Samsung, ARM, Google, IBM, Apple, só para citar algumas gigantes, têm hoje setores devotados a criar soluções para IoT. Com este panorama, é ainda difícil dizer quais serão os líderes de mercado dos próximos anos em IoT, se é que teremos algum.

Na área de protocolos não é diferente, inúmeras propostas e implementações já existem. Leves mudanças de escopo, algumas pequenas diferenças de casos de uso já são suficientes para que a companhia proponha o seu novo protocolo para IoT.

Um dos protocolos de troca de mensagens para IoT em uso recente é o MQTT (*Message Queue Telemetry Transport*). Criado pela IBM no final da década de 90, obviamente o protocolo carrega muito do cenário de uso original, mais voltado e adaptado para sistemas de supervisão e coleta de dados do tipo SCADA (*Supervisory Control and Data Acquisition* ou Sistemas de Supervisão e Aquisição de Dados, em português). Mas, mesmo assim, o MQTT encontrou seu espaço nesse amplo mercado de IoT. (Extraído de <https://www.embarcados.com.br/mqtt-protocolos-para-iot/>)

### 4. Problema

Sua empresa foi contratada para desenvolver um protótipo de um sistema para monitoramento ambiental que será posteriormente integrado a um sistema para monitoramento de cidades. O protótipo deve incluir todo o tratamento e controle de sensores analógicos e digitais, bem como uma IHM (interface Homem-Máquina) para apresentação das informações,

incluindo históricos dos dados. O protótipo deve ser desenvolvido num SBC que medirá temperatura, umidade, pressão atmosférica e luminosidade. A IHM deve apresentar, em tempo real, as leituras atuais. Ela também deve permitir a visualização do histórico com as 10 últimas medições de cada sensor. O sistema deve permitir o ajuste local e remoto do intervalo de tempo que serão realizadas as medições.

## 5. Requisitos

Além do descrito na seção anterior, o sistema a ser implementado no SBC deverá atender aos seguintes requisitos:

- 5.1. Códigos e protocolos:
  - 5.1.1. Código deverá ser escrito em linguagem C;
  - 5.1.2. Usar protocolo MQTT.
- 5.2. O sistema deverá possuir uma IHM local com interface baseada em display LCD, botões e chaves;
- 5.3. O sistema deverá implementar uma interface em forma de aplicativo para Desktop ou Smartphone. Esta interface deve ser capaz de apresentar as medições coletadas e as configurações para funcionamento da estação de medição;
- 5.4. O sistema deve apresentar informações sobre a conexão com a estação de medição na IHM local e remotamente.

## 6. Produto

No prazo indicado no cronograma a seguir, cada equipe deverá apresentar:

- 6.1. Códigos desenvolvidos no GitHub.
  - 6.1.1. Todos os códigos deverão estar detalhadamente comentados.
- 6.2. Manual do usuário disponível na página do projeto no GitHub.
  - 6.2.1. Deve conter instruções de uso do sistema desenvolvido;
  - 6.2.2. Deve apresentar as instruções de instalação do sistema;
  - 6.2.3. Deve ser redigido como um LEIAME (README). A página a seguir ilustra boas práticas e exemplos para confecção deste documento.  
<https://blog.rocketseat.com.br/como-fazer-um-bom-readme/>
- 6.3. Apresentação técnica em forma de vídeo contendo, no mínimo:
  - 6.3.1. Descrição do problema;
  - 6.3.2. Descrição em alto nível do sistema proposto, incluindo um ou mais diagramas de blocos apresentando a arquitetura da solução;
  - 6.3.3. Metodologias e técnicas aplicadas para o projeto e desenvolvimento da solução do problema, fundamentadas utilizando fontes confiáveis e diversificadas;
  - 6.3.4. Descrição e análise dos testes e simulações realizadas, de forma a testar todas as funcionalidades requisitadas;
  - 6.3.5. O vídeo deve ter no máximo 10min.**

## 7. Avaliação

Para avaliar o envolvimento do grupo nas discussões e na apresentação, o tutor poderá fazer perguntas variadas a qualquer membro, tanto nas sessões tutoriais quanto na apresentação.

A nota final será a composição de 3 (três) notas parciais:

Critério	Critérios para a nota	Peso
Desempenho Individual	Participação individual nas sessões tutoriais, de acordo com o interesse e entendimento demonstrados pelo aluno, assim como sua assiduidade, pontualidade e contribuição nas discussões.	3
Documentação	Manual do usuário de cada grupo, considerando qualidade da redação (ortografia e gramática) e organização dos tópicos. Apresentação em vídeo com a definição do problema, descrição da solução, explicação dos experimentos, análise dos resultados e conclusões.	4
Códigos	Qualidade do código fonte (organização e comentários), e execução correta dos códigos binários de acordo com testes de validação que explorem as situações de uso.	3

## 8. Cronograma

Semana	Data	Descrição
11	Sex - 20/Mai	Problema 3 – Apresentação do Problema 03
12	Qua - 25/Mai	Problema 3 – Seção Desenvolvimento #1
	Sex - 27/Mai	Problema 3 – Seção Tutorial #2
13	Qua - 01/Jun	Problema 3 – Seção Desenvolvimento #2
	Sex - 03/Jun	Problema 3 – Seção Tutorial #3
14	Qua - 08/Jun	Problema 3 – Seção Desenvolvimento #3
	Sex - 10/Jun	Problema 3 – Seção Tutorial #4
15	Qua - 15/Jun	Problema 3 – Seção Desenvolvimento #4
	Sex - 17/Jun	Problema 3 – Seção Tutorial #5
16	Qua - 22/Jun	<b>Feriado - São João</b>
	Sex - 24/Jun	Problema 3 – Seção Tutorial #6
17	Qua - 29/Jun	Problema 3 – Entrega/Avaliação