

# Projeto IoT com ESP32

---

- **Disciplina:** Sistemas Embarcados
  - **Professor:** Bella Nunes | Jymmy Barreto
  - **Data de Entrega:** 04/12
  - **Duração da Apresentação:** 15 minutos
- 

## 1. Objetivo do Projeto

Desenvolver um sistema IoT que integre:

- **ESP32** como dispositivos sensores/atuadores.
- **Comunicação sem fio** (Wi-Fi) e protocolo **MQTT** para troca de dados.
- **Dashboard** para visualização em tempo real dos dados coletados.

Os estudantes terão que propor a aplicação se baseando nos seguinte temas: monitoramento ambiental, automação residencial, Smart Build, Cidades Inteligentes ou temas relacionados com outras disciplinas do período corrente), desde que cumpram os requisitos técnicos.

---

## 2. Requisitos Técnicos

### Hardware Obrigatório

- **1x ou 2x módulos ESP32 (NodeMCU)**- Pode usar ESP8266 como módulo secundário.
- **Sensores básicos** (ex.: DHT11/DHT22, LM35, LDR, PIR, etc.).
- **Componentes eletrônicos** (resistor, LED, jumper, etc.).

Os grupos devem atentar a disponibilidade de placas, sensores, atuadores e outros recursos de hardware ofertados pela School (Kit de projetos + recursos no Garagem). Caso ocorra, os estudantes deverão providenciar com recurso próprio.

### Software/Ferramentas

- **Broker MQTT**
    - **Raspberry Pi**: Sistema Operacional - **Raspberry Pi OS (Linux)** ou **Raspbian (Debian ARM)**.
    - **ESP32**: Um módulo ESP32
  - Aplicação Web (Python + Flask/Django, JavaScript, TypeScript, NodeJS, React, Angular, Banco de Dados etc).
  - **ESP32/ESP8266:**
    - Firmware (FreeRTOS).
    - Biblioteca MQTT.
  - **Versionamento:** GitHub (repositório público).
-

### 3. Critérios de Avaliação

Item	Pontuação	Detalhes
<b>Check Point (18/11)</b>	5 pts	Demonstração e pequenas entregas do projeto.
<b>Funcionamento do Sistema</b>	25 pts	Projeto operando conforme requisitos (comunicação MQTT, Protocolo IPv6 coleta de dados, dashboard).
<b>Dashboard</b>	20 pts	Interface visual funcional, exibindo dados em tempo real.
<b>Relatório Técnico (ABNT2)</b>	20 pts	Estrutura completa: introdução, metodologia, resultados, códigos em apêndice.
<b>Apresentação (04/12)</b>	15 pts	Clareza, demonstração ao vivo, divisão de tarefas no grupo.
<b>GitHub (Organização)</b>	15 pts	Repositório bem documentado, commits colaborativos, README detalhado.

**Nota:** Projetos com funcionalidades extras (ex.: **Alertas por e-mail ou Apps de mensagens, Integração com Zabbix, Infraestrutura de comunicação em IPv6**) ganharão bônus de até **10 pts**.

### 4. Cronograma e Acompanhamento

- **04/11 a 02/12:** Checkpoints semanais (presenciais/remotos).
  - **04/11:** Definição da ideia do projeto.
  - **18/11:** Testes preliminares (Prototipação de hardware, esboço de código e outros artefatos de encaminhamento do projeto).
  - **02/12:** Validação do Aplicação Web.
- **04/12:** Apresentação final (**15 min por grupo**).
- **09/12:** Entrega final de todos os artefatos do Projeto.
- **13/12:** **Apresentação na Mostra Tech**

### 5. Estrutura do Relatório (ABNT2)

1. **Capa** (Nome da Faculdade, título, autores, data).
2. **Introdução** (Contexto, motivação, objetivos).
3. **Metodologia**
  - Diagrama do sistema.
  - Lista de hardware/software.
  - Fluxo de comunicação (MQTT + Wi-Fi).
4. **Resultados**
  - Prints do dashboard.
  - Dados coletados (gráficos/tabelas).
  - Evidências de funcionamento (fotos/vídeos).
5. **Conclusão** (Desafios, aprendizados, melhorias futuras).
6. **Apêndice**

- Códigos completos (Raspberry Pi + ESPs).
- Configurações do broker MQTT.

---

## 6. Template do Repositório GitHub

```
|— README.md          # Descrição do projeto, requisitos, instruções.
|— /docs              # Relatório em PDF (ABNT2) + imagens.
|— /raspberrypi       # Códigos do broker MQTT + dashboard.
|— /esp32-esp8266     # Firmware dos módulos (FreeRTOS).
|— /schematics        # Diagramas eletrônicos (Fritzing, KiCad ou outra ferramenta
de prototipação).
```

**Dica:** Use o [GitHub Projects](#) para gerenciar tarefas em equipe.

---

## 7. Ideias Sugeridas e Links de Apoio para o Projeto

---

Aqui estão algumas ideias para inspirar seu projeto IoT, além de links úteis com tutoriais e exemplos práticos:

### Ideias Sugeridas

- **Estação Meteorológica:** ESPs enviam temperatura/umidade para o Pi, que exibe em um gráfico.
- **Alarme com Sensor PIR:** ESP detecta movimento e envia alerta MQTT para o Pi acionar um aviso no dashboard.
- **Irrigação Automática:** Sensor de umidade do solo (ESP) controla uma válvula solenoide via Pi.
- **Monitoramento de Energia:** Medição de consumo elétrico com sensor de corrente (ACS712) + exibição no dashboard.

### Links de Apoio

#### 1. Projetos com Raspberry Pi:

- [Random Nerd Tutorials – Projetos Raspberry Pi](#)
- Exemplos: servidor MQTT, controle de GPIO, integração com sensores.

#### 2. Projetos IoT Prontos (ESP e Pi):

- [IoT Design Pro – Ideias de Projetos](#)
- Tutoriais passo a passo com código e esquemas.

#### 3. Projetos com ESP8266/ESP32:

- [IoT Circuit Hub – ESP8266 Projects](#)
- Projetos focados em MQTT, Wi-Fi e sensores.

#### 4. Extras:

- [Mosquitto MQTT Broker Setup](#) (para configurar o broker no Pi).

- [Node-RED Dashboard Tutorial](#) (para criar interfaces visuais).

---

**Dica:** Combine ideias dos links com os requisitos do projeto (ex.: usar um tutorial de **ESP8266 + MQTT** e adaptar ao dashboard no Pi).

Se precisar de ajuda para integrar componentes, consulte os tutoriais ou peça orientação durante os checkpoints! 🚀

---

## 8. Formação dos Grupos

Total de participantes por grupos:

- **Mínimo 4**

Todo grupo deverá eleger uma pessoa para representá-la. O representante deverá indicar um dos participantes para a apresentação do projeto no dia **05/06**.

## 9. Entrega Final

- **Formato:**
  - Repositório GitHub (link do projeto).
  - Relatório em PDF (nome no padrão: **Projeto\_IoT\_GrupoX.pdf**).
- **Prazo:** 09/12 (até 23h59).

As entregas serão acompanhadas via Google Classroom.

**Observação:** **Projetos não funcionais na apresentação perderão 50% da pontuação de "Funcionamento".**

---