

## Enunciado do projeto 2024/2025

Controlo e Monitorização Inteligente de uma Estufa Agrícola

---

### 1. ENQUADRAMENTO

O desenvolvimento de aplicações multiplataforma permite interligar diversos sistemas, nomeadamente para o seu controlo e monitorização. Como tal, este projeto pretende desenvolver uma *Single Page Application* (SPA), utilizando os conteúdos abordados na UC (HTML, Bootstrap, jQuery, Node.js, ...) que permita monitorizar e controlar os parâmetros ambientais de uma estufa agrícola. A solução incluirá sensores e atuadores para otimizar condições de crescimento das plantas, tais como temperatura, humidade e iluminação, com dados em tempo real e automação baseada em regras.

### 2. CENÁRIO

A aplicação deverá apresentar aos seus utilizadores, no **ecrã principal**, informação geral sobre o ambiente (ex.: temperatura atual através de uma API de meteorologia, temperatura real medida por um sensor). Além da informação geral do ambiente, a página principal deve ainda apresentar informação sobre a produção agrícola, dividida por zonas (ex.: vegetais, frutas, ...). Cada zona é composta por um conjunto de sensores (ex.: humidade do solo/ar, iluminação, ...) e atuadores (ex.: sistema de rega).

A aplicação deve ter um **ecrã secundário** onde serão definidas regras de automação dos sistemas (ex.: apresentar uma notificação no ecrã principal quando o valor de um determinado sensor descer abaixo de um valor definido, ativar o sistema de rega quando o valor da humidade do solo estiver abaixo de um determinado valor).

Como **terceiro ecrã**, a aplicação deverá permitir, para cada zona, adicionar e remover sensores e atuadores.

zonas, e.g. categoria frutas/vegetais/etc

-> Época normal, deve ser possível adicionar/remover dinamicamente zonas na terceira página.

### 3. REQUISITOS DA APLICAÇÃO

De acordo com o cenário descrito, a solução deve cumprir com os seguintes requisitos:

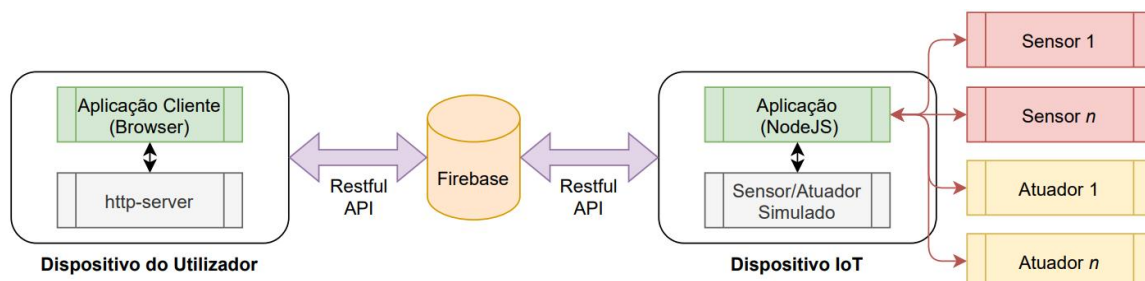
- a) apresentar **2 ou mais informações** gerais do ambiente; e.g. **Temperatura/humidade**
- b) apresentar **2 ou mais zonas distintas**, cada uma com **2 ou mais** elementos; **elementos:**
  - atuadores
  - sensores
- c) utilizar **2 ou mais** tipos de **sensores**; **Analogico, Digital**
- d) utilizar **2 ou mais** tipos de **atuadores**; **Switch, valor absoluto**
- e) permitir definir **1 ou mais** tipos de **regras**, utilizando pelo menos 1 sensor e/ou atuador; **E.g. Ações automatizadas com base nas leituras dos sensores**
- f) utilizar **2 ou mais** APIs externas; **Ver API dos slides**
- g) apresentar os dados em **tempo real**;
- h) utilizar a base de dados Firebase. **Ver fig. 1 abaixo**

Deverá ser utilizado **pelo menos 1 dispositivo IoT** (ex.: Raspberry Pi ou ESP8266), que irá proceder à recolha dos dados e controlo dos atuadores e comunicar periodicamente os dados dos sensores.

Por questões de simplicidade, não será necessário implementar mecanismos de autorização nem autenticação dos utilizadores.

### 4. ARQUITETURA DA SOLUÇÃO

Os sistemas deverão comunicar através de uma base de dados em tempo real, nomeadamente a **Firestore Realtime Database**. Salienta-se que a SPA nunca deverá comunicar diretamente com o dispositivo IoT. Na **Figura 1** é representada a arquitetura geral da solução, composta pelo dispositivo do utilizador, a base de dados (Firestore), o dispositivo IoT e respetivos sensores e atuadores. Caso pretenda, pode recorrer à simulação de um ou mais sensores e atuadores, desde que exista pelo menos um sensor e um atuador reais.



**Figura 1.** Representação da arquitetura geral da solução.

## 5. RELATÓRIO

O projeto deverá ser documentado em relatório. O documento deverá apresentar a arquitetura da solução implementada, descrevendo os principais módulos implementados e *hardware* utilizado. Deverão também ser descritas as principais características, funcionalidades e potencialidades da solução, bem como a justificação de eventuais decisões tomadas (ex.: escolha de sensores/atuadores, regras).

O relatório deverá também discutir e justificar as funcionalidades que não estejam implementadas ou apenas parcialmente funcionais. Por fim, o relatório deve contar com uma secção de autoavaliação, onde os estudantes devem fazer uma apreciação global do trabalho desenvolvido e do seu desempenho. O relatório **não deve** conter código-fonte.

## 6. SUBMISSÃO

Os estudantes deverão submeter no Moodle, até à data limite de submissão, um ficheiro único comprimido (ex.: *.zip*, *.rar*, *.7z*, *.tar.gz*) contendo os seguintes elementos:

- a) Código fonte da **SPA**: todos os ficheiros da aplicação cliente e servidor, e ficheiros relacionados que sejam considerados relevantes;
- b) Código fonte do dispositivo **IoT**: todos os ficheiros da aplicação do dispositivo IoT, bem como eventuais ficheiros de configuração que sejam relevantes para a sua execução;
- c) Configuração da **Firebase**: ficheiro JSON de configuração da base de dados Firebase;
- d) **Relatório**: ficheiro em formato PDF, com tamanho de letra 12, com o máximo de 10 páginas.

## 7. CRITÉRIOS DE AVALIAÇÃO

O projeto deve ser realizado em grupos de 2 ou 3 estudantes.

A avaliação do projeto será de acordo com os seguintes critérios:

- a) Aplicação SPA: 50%
- b) Aplicação IoT: 40%
- c) Relatório: 10%

A apresentação oral será avaliada individualmente (entre 0 e 100%), a multiplicar pela nota obtida nos critérios anteriores.

## 8. DATAS RELEVANTES

Lançamento do enunciado: 11 de novembro de 2024

### **Avaliação Periódica:**

- Submissão do projeto: 4 de janeiro de 2025
- Apresentação oral do projeto: 10 de janeiro de 2025

### **Época Normal:**

- Submissão do projeto: 20 de janeiro de 2025
- Apresentação oral do projeto: 22 de janeiro de 2025

## 9. ADENDA PARA ÉPOCA NORMAL

Os estudantes que pretenderem entregar o projeto na Época Normal, devem continuar o desenvolvimento do mesmo. Assim, além das funcionalidades acima descritas, deve também ser implementada, no **terceiro ecrã**, a funcionalidade de adicionar novas zonas ao ecrã principal dinamicamente.