

U.C. de Projeto Integrado de

Telecomunicações

Ano Letivo: **2021/2022**

**Especificação da Fase C**

**Grupo 2**

* Catarina Neves, a93088
* Eduardo Cardoso, a89627
* José Gomes, a93083
* Luís Oliveira, a89380

06/04/2022

Universidade do Minho

Mestrado Integrado em Engenharia de Telecomunicações e Informática

Índice

[1. Introdução 5](#_Toc102398291)

[2. Especificação da Fase C 6](#_Toc102398292)

[2.1 Arquitetura e Funcionalidades dos sistemas 6](#_Toc102398293)

[2.*2 Hardware* 7](#_Toc102398294)

[2.2.1 ESP32-DevKitC-32D 8](#_Toc102398295)

[2.2.2 Sensor DHT11 8](#_Toc102398296)

[2.2.3 Sensor BME280 9](#_Toc102398297)

[2.3 *Software* 9](#_Toc102398298)

[3. Planificação do projeto 11](#_Toc102398299)

[4. Conclusão 12](#_Toc102398300)

[5. Bibliografia 13](#_Toc102398301)

Índice de figuras

[Figura 1 - Arquitetura da Fase C. 6](https://d.docs.live.net/c614cccb55865bf0/Documents/GitHub/Projeto-Integrado-de-Telecomunicacoes/relatorios/fase_C/REC-G2.docx#_Toc102677467)

[Figura 2 - Diagrama de Gantt do planeamento temporal. 11](https://d.docs.live.net/c614cccb55865bf0/Documents/GitHub/Projeto-Integrado-de-Telecomunicacoes/relatorios/fase_C/REC-G2.docx#_Toc102677468)

Lista de siglas e acrónimos

**BLE** *Bluetooth Low Energy*

**IDE** *Integrated Development Environment*

**MP3** *MPEG Layer 3*

**TCP** *Transmission Control Protocol*

**IP** *Internet Protocol*

**SQL** *Structured Query Language*

**HTML** *HyperText Markup Language*

**ISP** *Internet Service Provider*

**SPI** *Serial Peripheral Interface*

**I2C** *Inter-Integrated Circuit*

# Introdução

Na sequência dos relatórios de especificação anteriores, serve este relatório como introdução e de planificação das etapas a serem desenvolvidas na terceira fase do projeto.

Assim sendo, o documento, passará por descrever os passos a percorrer, tal como as suas implementações: uma parte desta fase foi já realizada na anterior (bases de dados relacional, servidor Web e sua autenticação, etc.), daí que o grupo focar-se-á em desenvolver as restantes funcionalidades, tal como corrigir e implementar outras anteriores.

De igual forma serão detalhadas as técnicas e tecnologias utilizadas, a sua utilização e a arquitetura geral da fase (não desprezando os constrangimentos temporais e a sua devida planificação).

# Especificação da Fase C

## 2.1 Arquitetura e Funcionalidades dos sistemas

A figura seguinte mapeia a arquitetura dos sistemas e as interações entre os mesmos.

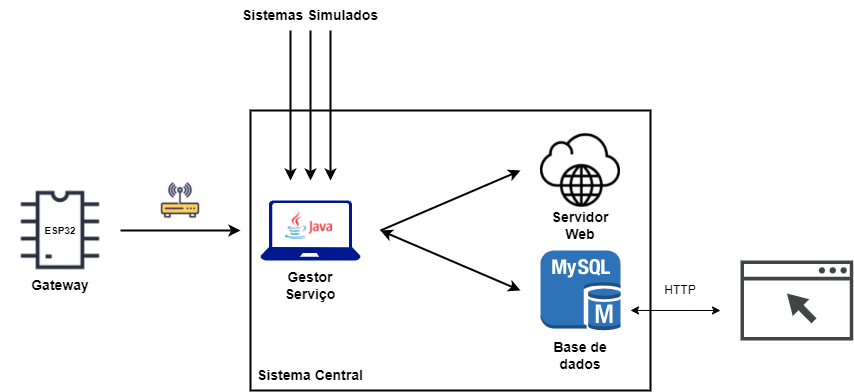


Figura 1 - Arquitetura da Fase C.

A comunicação entre os vários *Gateways* e o Gestor de Serviço será realizada de acordo com um protocolo de comunicação aplicacional, que apesar de ter sido necessário à fase anterior, o grupo falhou a implementação deste, dentro do prazo estipulado. Esta comunicação permitirá o envio das amostras, recebidas por cada *Gateway,* para o Sistema Central, que será responsável pelo armazenamento das mesmas na base de dados.

Para isso, será necessária uma base de dados relacional, contudo, esta já foi implementada pelo grupo na fase anterior, logo, o grupo focar-se-á nos melhoramentos à aplicação já desenvolvida.

Da mesma maneira, o processo de autenticação de utilizadores já foi desenvolvido e apresentado na fase anterior, contudo, este será modificado com o intuito, na etapa final, de obter-se conteúdo, que seja acessível a um determinado tipo de utilizadores.

Por fim, com o objetivo de conseguir-se identificar, quando as amostras são recebidas e se estas têm origem no Gateway ou num dos Sistemas Simulados, será adicionado um identificador extra, que permitirá determinar a origem destas.

## 2.*2 Hardware*

A tabela seguinte apresenta todo o *hardware* necessário ao desenvolvimento da fase C.

O material utilizado será o mesmo que foi utilizado no desenvolvimento das fases anteriores.

Tabela 1 - Hardware necessário à fase C.

|  |  |  |
| --- | --- | --- |
| **Imagem** | **Designação** | **Descrição** |
| Uma imagem com texto, eletrónica, circuito  Descrição gerada automaticamente | ESP32-DevKitC-32D | 2 placas: uma para a implementação do sistema sensor e outra para o gateway BLE/Wi-Fi |
| Uma imagem com texto, eletrónica  Descrição gerada automaticamente | DHT11 | Sensor de temperatura e humidade |
|  | BME280 | Sensor de pressão barométrica |
|  | Fios de ligação | Fios responsáveis pela conexão dos componentes necessários |
| Uma imagem com texto, eletrónica, computador, portátil  Descrição gerada automaticamente | Computadores | Serão necessários dois computadores: um para a ligação ao Sistema Sensor e o restante para a ligação ao G*ateway*. |
|  | *Breadboard* | Interface de conexão entre os circuitos |

### 2.2.1 ESP32-DevKitC-32D

Módulo genérico de Wi-Fi, *Bluetooth* e BLE, [1] que possibilita uma grande variedade de aplicações como por exemplo redes de sensores de baixa potência, decodificação de MP3 e codificação de voz.

A tabela seguinte apresenta as caraterísticas da placa em questão.

Tabela 2 - Caraterísticas da placa ESP32-DevKit-32D.

|  |  |
| --- | --- |
| **Módulos** | **ESP32-DevKit-32D** |
| SPI *flash* | 32 Mbits, 3.3 V |
| *Core* | ESP32-D0WD |
| *Crystal* | 40 MHz (apenas para a funcionalidade do Wi-Fi e do *Bluetooth*) |
| Antena | Conector U.FL (que precisa de estar conectado a uma antena IPEX externa) |
| Dimensões (Unidade: mm) | (18.00±0.10) × (19.20±0.10) × (3.20±0.10) |

O *Bluetooth*, o BLE e o Wi-Fi são diferentes tipos de comunicação, que são intrínsecas a esta placa. A utilização do Wi-Fi permite uma grande cobertura ou alcance físico e conexão direta com a *internet* através de um ISP (*Internet Service Provider*).

A *sleep current* (corrente caraterística, quando a placa opera em modo *standby*) da placa é inferior a 5 microamperes, o que torna este componente ideal para aplicações em sistemas eletrónicos alimentados por baterias. Este componente suporta uma taxa de transmissão até 150 Mbps e 29 dBm de potência de saída, para a antena, garantindo desta forma, o maior alcance físico.

### 2.2.2 Sensor DHT11

O grupo decidiu operar com o sensor *DHT11-WS* essencialmente porque foi sugerido e recomendado pelos docentes, tal como pelo seu custo acessível e pela facilidade de utilização.

O sensor referidoé um sensor de temperatura e humidade que permite realizar leituras de temperaturas compreendidas entre 0 e 50 graus Celcius e valores de humidade compreendidos entre 20 e 90%.

Em termos práticos, este sensor deteta a temperatura e a humidade e envia os valores recolhidos para o seu microcontrolador, que por sua vez deverá estar programado para efetuar alguma ação quando é atingida determinada temperatura ou humidade.

### 2.2.3 Sensor BME280

O sensor BME280 mede a pressão atmosférica desde 30kPa até 110kPa, como também a temperatura e a humidade relativa. Este sensor oferece uma interface de SPI (*Serial Peripheral Interface*) de 3.3 Volts e uma interface I2C (*Inter-Integrated Circuit*), que é tolerante a 5 Volts. No total, a placa BME280 possui 10 pinos, contudo, não podem ser utilizados mais do que 6 pinos numa única vez.

## 2.3 *Software*

A tabela seguinte ilustra o *software* necessário à implementação da fase C. Na fase anterior, o grupo decidiu criar e configurar uma base de dados relacional, para auxiliar o desenvolvimento desta fase final.

Tabela 3 - Software necessário à fase C.

|  |  |  |
| --- | --- | --- |
| **Símbolo** | **Designação** | **Função** |
| Uma imagem com bola de bilhar  Descrição gerada automaticamente | Arduino IDE | Implementação das *sockets* no sistema *gateway.* |
|  | IntelliJ IDEA | Implementação do sistema central, que incorpora: serviços de comunicação com a *web* e o repositório da base de dados. |
| mysql-logo-png-transparent ⋆ Altyra - Desenvolvimento de Software | MySQL | Armazenamento, gestão e configuração da base de dados. |
| Spring Framework SVG Vector Logos - Vector Logo Zone | *Spring Framework* | Responsável pela instanciação das classes da aplicação Java e pela incorporação de várias dependências, que poderão ser aplicadas na respetiva aplicação. |

Dentro do IDE (*Integrated Development Environment*) *Intellij IDEA* serão definidos e estruturados os serviços web e será declarada e inicializada uma *interface* (linguagem *Java*), que possuirá um conjunto de métodos e de *queries* responsáveis pela interação com a base de dados.

A transmissão das amostras do Gateway para a aplicação *java,* será construída em ambos os ambientes de programação (*Arduino IDE e* *Intellij IDEA)* e será necessário definir *sockets* em ambos as aplicações (aplicação java e a aplicação executada pelo *Gateway*).

O programa *MySQL* será uma ferramenta útil para a edição da base de dados, como por exemplo: manipulação das colunas da tabela definida; descrição da tabela e visualização dos elementos presentes.

2.4 Linguagens de programação

A tabela seguinte contém as linguagens de programação, que serão necessárias à implementação desta fase.

Tabela 4 - Linguagens de programação necessárias à implementação da fase C.

|  |  |  |
| --- | --- | --- |
| **Linguagens de programação** | | |
| Download C++ Logo in SVG Vector or PNG File Format - Logo.wine | C++ | A linguagem C++ será empregue na definição das *sockets* e na construção do protocolo de comunicação. |
|  | Java | A linguagem Java será utilizada na construção de várias classes, responsáveis pelo funcionamento do serviço *web*, como também, na definição de um repositório, que irá interagir com a base de dados. |
|  | SQL | A linguagem SQL (*Structured Query Language*) será utilizada na configuração e manipulação da base de dados. |
|  | HTML | A linguagem HTML (*HyperText Markup Language)* será empregue na construção da página web do respetivo sistema. |
| upload.wikimedia.org/wikipedia/commons/thumb/9/... | JavaScript | A linguagem JS será utilizada na manipulação e edição da página web do respetivo sistema. |

# Planificação do projeto

Para garantir consistência, linearidade e para que se cumpram todos os objetivos, é necessário recorrer a um planeamento bem estruturado. O planeamento do projeto encontra-se ilustrado pelo diagrama de *Gantt seguinte*.

**Etapa**

**Sub-etapa**

Figura 2 - Diagrama de Gantt do planeamento temporal.

# Conclusão

Em suma, o grupo considera que esta última fase terá uma execução mais rápida, que permitirá a correção e retificação de erros cometidos anteriormente.

O grupo encontra-se ciente dos passos a serem tomados (serão dados com mais detalhe, visto não serem muito diferenciados, dos que foram dados anteriormente). Espera-se conseguir obter um trabalho ainda melhor e com mais qualidade.

# Bibliografia

|  |  |
| --- | --- |
| [1] | Espressif Systems, “ESP32 Series,” 21 Fevereiro 2022. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32\_datasheet\_en.pdf. |
| [2] | “Bot n Roll,” 20 Fevereiro 2022. [Online]. Available: https://www.botnroll.com/1219-medium\_default/sensor-de-temperatura-e-humidade-dht11.jpg. |
| [3] | dfrobot, 20 Fevereiro 2022. [Online]. Available: https://image.dfrobot.com/image/data/DFR0067/DFR0067\_DS\_10\_en.pdf. |
| [4] | “fnac-static,” 20 Fevereiro 2022. [Online]. Available: https://static.fnac-static.com/multimedia/Images/PT/NR/67/05/62/6423911/1540-1.jpg. |
| [5] | “Iotone,” 22 Fevereiro 2022. [Online]. Available: https://www.iotone.com/files/vendor/logo\_Thingspeak.jpg. |
| [6] | “Sistema de Monitorização de Estações Meteorológicas,” 20 Fevereiro 2022. [Online]. Available: elearning.uminho.pt. |
| [7] | “sparkfun,” 20 Fevereiro 2022. [Online]. Available: https://cdn.sparkfun.com//assets/parts/3/3/4/5/09567-01-Working.jpg. |
| [8] | botnroll, 2 2 2022. [Online]. Available: https://www.botnroll.com/8958-medium\_default/sensor-de-press-o-atmosf-rica-bmp280.jpg. |
| [9] | Bot n Roll, “botnroll,” 22 2 2022. [Online]. Available: https://www.botnroll.com/8958-medium\_default/sensor-de-press-o-atmosf-rica-bmp280.jpg. |