



**Universidade do Minho**  
Escola de Engenharia

## **Laboratórios de Telecomunicações e Informática II**

ENGENHARIA DE TELECOMUNICAÇÕES E INFORMÁTICA  
**2020/2021**

(Docentes: Bruno Daniel Mestre Viana Ribeiro, Vadym Serhiyovych Hapanchak,  
José Augusto Afonso, Sérgio Adriano Fernandes Lopes)

21 de abril de 2021

# **Relatório de especificação**

## **FASE B**

Rui Filipe Ribeiro Freitas – [a84121@alunos.uminho.pt](mailto:a84121@alunos.uminho.pt)

Sandro Teixeira Ribeiro – [a85316@alunos.uminho.pt](mailto:a85316@alunos.uminho.pt)

Tiago João Pereira Ferreira – [a85392@alunos.uminho.pt](mailto:a85392@alunos.uminho.pt)

# Índice

Índice de figuras .....	3
Índice de tabelas .....	4
Lista de abreviaturas .....	5
Introdução .....	6
1. Planeamento do projeto .....	7
1.1. Planeamento temporal .....	7
1.2. Tecnologias/Ferramentas necessárias .....	8
1.2.1. Ao nível do <i>hardware</i> .....	8
1.2.2. Ao nível do <i>software</i> .....	9
2. Fundamentos .....	10
2.1. Sistema central .....	10
2.2. Sistema sensor simulado .....	10
3. Desenvolvimento .....	11
3.1. Protocolo de comunicação entre concentradores e sistema central .....	11
3.2. Base de dados .....	13
3.3. Interface Web .....	13
Referências .....	14

# Índice de figuras

Figura 1 - Planeamento temporal tabela.....	7
Figura 2 - Planeamento temporal gráfico.....	7
Figura 3 - Esquema da Fase B. ....	10
Figura 4 - Mensagem START. ....	11
Figura 5 - Mensagem STOP.....	11
Figura 6 - Mensagem DATA. ....	11
Figura 7 - Mensagem ERROR. ....	12
Figura 8 - Mensagem BEGIN. ....	12
Figura 9 - Mensagem END. ....	12
Figura 10 - Diagrama Relacional.....	13

## Índice de tabelas

Tabela 1 - Tecnologias ao nível do hardware. ....	8
Tabela 2 - Tecnologias ao nível do software. ....	9

# Lista de abreviaturas

**LTI II** – Laboratórios de Telecomunicações e Informática II

**TCP** – Transmission Control Protocol

**UDP** – User Datagram Protocol

# Introdução

No âmbito da Unidade Curricular de LTI II (Laboratórios de Telecomunicações e Informática II) foi-nos proposto o desenvolvimento de um sistema de iluminação inteligente que permita monitorizar a ocupação e a luminosidade em diferentes áreas de uma residência, bem como controlar a luminosidade desses ambientes através do acionamento de lâmpadas.

Este relatório diz respeito à segunda fase de 3 do desenvolvimento deste projeto em que nesta fase os objetivos passam pela implementação de sistemas sensores simulados e de um sistema central que será responsável pela comunicação com os vários concentradores através de um protocolo de comunicação por nós realizado. Para além disso este sistema central terá como objetivo apresentar ao utilizador um menu com várias funções como observar gráficos sobre a luminosidade e movimento numa dada área.

De modo a sermos capazes de cumprir com os objetivos deste projeto semestral devemos pôr em prática conhecimentos adquiridos noutras unidades curriculares, nomeadamente Redes de Computadores, Sistemas Operativos, Sistemas Distribuídos, Eletrónica e Laboratórios de Telecomunicações e Informática I.

# 1. Planeamento do projeto

## 1.1. Planeamento temporal

De modo que o grupo se mantenha focado no trabalho e com um compromisso para cumprir horários, resolvemos planear as tarefas a fazer nesta fase. Na figura 1 observamos em forma de tabela os vários assuntos a ser tratados nesta fase com um período dado por nós para cumprir. Em relação à figura 2 demonstramos em forma de gráfico o tempo despendido nas várias tarefas. Ambas as figuras foram retiradas do programa *Gantt*.

GANTT project		
Nome	Data de início	Data de fim
☐ Fase B	14-04-2021	14-06-2021
• Planeamento Temporal	14-04-2021	16-04-2021
• Definição da Arquitetura do Sistema	20-04-2021	20-04-2021
• Início da elaboração do relatório	21-04-2021	21-04-2021
• Desenvolvimento código sensores simulados	22-04-2021	28-04-2021
• Desenvolvimento código Concentrador	29-04-2021	05-05-2021
• Desenvolvimento código Sistema Central	06-05-2021	11-05-2021
• Realização de Testes	12-05-2021	12-05-2021
• Discussão dos resultados obtidos	13-05-2021	13-05-2021
• Conclusão e revisão do relatório	14-06-2021	14-06-2021

Figura 1 - Planeamento temporal tabela.

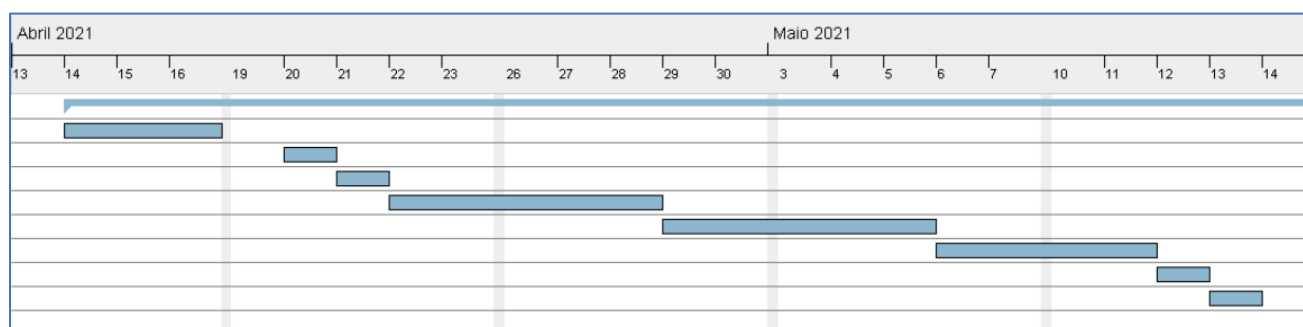


Figura 2 - Planeamento temporal gráfico.

## 1.2. Tecnologias/Ferramentas necessárias

Em qualquer projeto são necessárias certas tecnologias/ferramentas que nos facilitem o trabalho e ao mesmo tempo que aumentem a nossa produtividade. Para este trabalho em específico teremos de recorrer a tecnologias específicas tanto ao nível do *hardware* como ao nível do *software*, de modo a cumprir com os objetivos propostos.

### 1.2.1. Ao nível do hardware

No que toca à parte do *hardware* deste nosso trabalho as tecnologias por nós utilizadas estão presentes na tabela 1, sendo que estas podem vir a ser alteradas no futuro.

Tabela 1 - Tecnologias ao nível do hardware.









FERRAMENTA	QUANTIDADE	UTILIZAÇÃO
Computador Portátil	1	Desenvolvimento de código, elaboração do relatório, pesquisa e testes
Placa ESP32	1	Comunicação com os sensores e os LEDs
<i>Sensor de movimento</i> (PIR HC-SR501)	1	Deteção de movimento
Sensor de luminosidade (LDR 5...10kΩ)	1	Medição da luminosidade
LED	3	Simulação de lâmpadas
Cabo Micro USB	1	Ligações físicas entre o PC e a placa ESP32
Fios de ligação	Vários	Efetuar as ligações entre as placas ESP32 e os sensores e LEDs
Resistência	Várias	Limitar a corrente drenada pelos LEDs



### 1.2.2. Ao nível do *software*

Em relação ao *software* que irá ser utilizado na execução do nosso projeto este passará maioritariamente pelas aplicações apresentadas na tabela 2. Nesta realçamos as mais importantes e fundamentais, sabendo que, no entanto, poderão sofrer alterações no futuro dado ainda nos encontrarmos numa fase mais introdutória do trabalho.

Tabela 2 - Tecnologias ao nível do *software*.

LOGO	DESCRIÇÃO	UTILIZAÇÃO
	<b>Microsoft Word</b>	Realização e sincronização do relatório do projeto
	<b>GanttProject</b>	Planeamento do projeto através de diagramas de Gantt
	<b>Arduino IDE</b>	Software para a conexão com a placa ESP32 e comunicação com o mesmo
	<b>Visual Studio Code</b>	Execução e compilação de código
	<b>GitHub</b>	Sincronização do código da aplicação
	<b>Visual Paradigm</b>	Implementação de fluxogramas para a elaboração da aplicação e do projeto
	<b>Google Drive</b>	Sincronização de ficheiros essenciais à realização do trabalho
	<b>Discord</b>	Comunicação entre o grupo

## 2. Fundamentos

### 2.1. Sistema central

O sistema central deve recolher a informação enviada pelos concentradores da rede de forma a apresentar um conjunto de dados de monitorização dos valores recolhidos nos sensores. Esta comunicação será baseada no protocolo TCP/IP. Os dados recebidos dos concentradores podem ser filtrados, antes de serem registados (em base de dados simples), para que não tenham o mesmo nível de detalhe. Nesta fase, o sistema central deve possuir uma interface de administração através de serviço web, para que o sistema possa ser consultado local ou remotamente. O papel do sistema central é o de aplicação ou de serviço TCP/IP, contendo uma lista dos concentradores com os quais pode estabelecer comunicação.

### 2.2. Sistema sensor simulado

A implementação de um sistema sensor simulado permite o teste das funcionalidades do sistema central e concentradores num ambiente com múltiplos sistemas sensores ativos em simultâneo.

Estes sistemas sensores devem ser implementados numa aplicação software e devem incluir um ou mais ficheiros com valores previamente armazenados com o resultado da monitorização do funcionamento de um sistema sensor real. Deve também utilizar o protocolo de comunicação definido anteriormente na Fase A, mas para estes sistemas simulados, as mensagens deverão ser encapsuladas em pacotes UDP.

Na figura seguinte apresentamos o esquema da Fase B com o sistema central a comunicar com o concentrador através do protocolo TCP/IP e o concentrador com o sistema sensor simulado através de uma ligação UDP/IP.

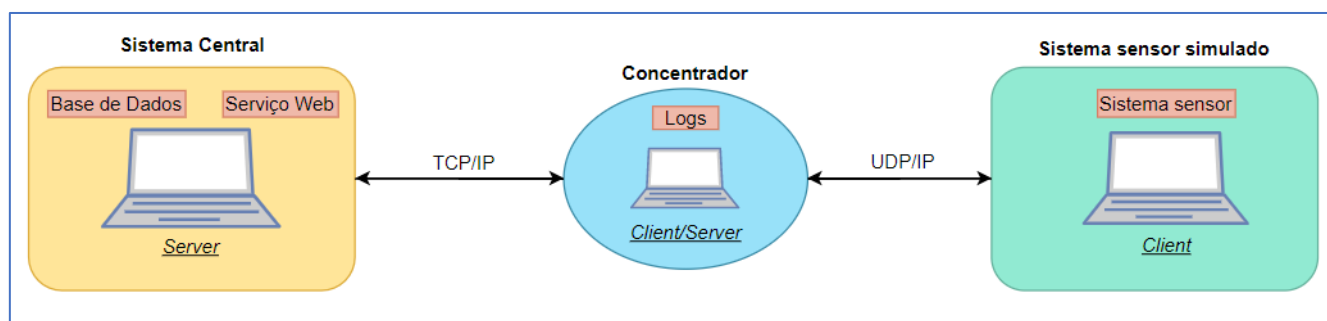


Figura 3 - Esquema da Fase B.

### 3. Desenvolvimento

#### 3.1. Protocolo de comunicação entre concentradores e sistema central

O protocolo de comunicação serve para os concentradores enviarem para o sistema central os dados recolhidos nos sistemas sensores através de um conjunto de mensagens, as quais devem ser encapsuladas através do protocolo TCP. Como o TCP já é um protocolo confirmado e orientado à conexão não é necessário implementar mecanismos de controlo de fluxo. No entanto é necessário incluir uma forma de os gestores saberem quando a comunicação com um determinado concentrador é terminada. Para isso decidimos implementar vários tipos de mensagens, que são apresentados em seguida.

- **START** – A mensagem *START* é enviada do concentrador para o sistema central a pedir autorização para enviar dados. Esta mensagem tem os campos apresentados na figura 4.

Tipo (1 byte)	Timestamp (4 bytes)	Area ID (1 byte)	Endereço IP (4 bytes)	Password (4 bytes)
------------------	------------------------	---------------------	--------------------------	-----------------------

Figura 4 - Mensagem *START*.

- **STOP** – A mensagem *STOP* é enviada do concentrador para o sistema central a indicar que já não tem dados para enviar e que deseja terminar a sessão TCP. Esta mensagem tem os campos apresentados na figura 5.

Tipo (1 byte)	Timestamp (4 bytes)	Area ID (1 byte)	Endereço IP (4 bytes)	Password (4 bytes)
------------------	------------------------	---------------------	--------------------------	-----------------------

Figura 5 - Mensagem *STOP*.

- **DATA** – A mensagem *DATA* é enviada do concentrador para um gestor de serviços com os valores das amostras recolhidas de todos os sistemas sensores que estão conectados. Esta mensagem tem os campos apresentados na figura 6.

Tipo (1 byte)	Timestamp (4 bytes)	Amostra 1 (4 bytes)	...	Amostra N (4 bytes)	GPS (2 bytes)
------------------	------------------------	------------------------	-----	------------------------	------------------

Figura 6 - Mensagem *DATA*.

- **ERROR** – A mensagem *ERROR* é enviada do concentrador para o sistema central a indicar que ocorreu um erro. Esta mensagem tem os campos apresentados na figura 7.

Tipo (1 byte)	Timestamp (4 bytes)	Area ID (1 byte)	GPS (2 bytes)	Error ID (2 bytes)
------------------	------------------------	---------------------	------------------	-----------------------

Figura 7 - Mensagem *ERROR*.

- **BEGIN** – A mensagem *BEGIN* é enviada do sistema central para o concentrador a indicar que está disponível para receber dados. Esta mensagem tem os campos apresentados na figura 8.

Tipo (1 byte)	Timestamp (4 bytes)	Begin ID (1 byte)
------------------	------------------------	----------------------

Figura 8 - Mensagem *BEGIN*.

- **END** – A mensagem *END* é enviada do sistema central para um concentrador a indicar que não está mais disponível para receber dados. Esta mensagem tem os campos apresentados na figura 9.

Tipo (1 byte)	Timestamp (4 bytes)	End ID (1 byte)
------------------	------------------------	--------------------

Figura 9 - Mensagem *END*.

### 3.2. Base de dados

De modo a armazenar todas as informações enviadas pelos vários concentradores, iremos elaborar uma base de dados para guardar todos esses dados numa forma mais eficiente e mais prática aos olhos do utilizador. O grupo decidiu implementar a base de dados em linguagem MySQL e o planeamento inicial da mesma num diagrama de entidades e relacionamentos é apresentado na figura seguinte.

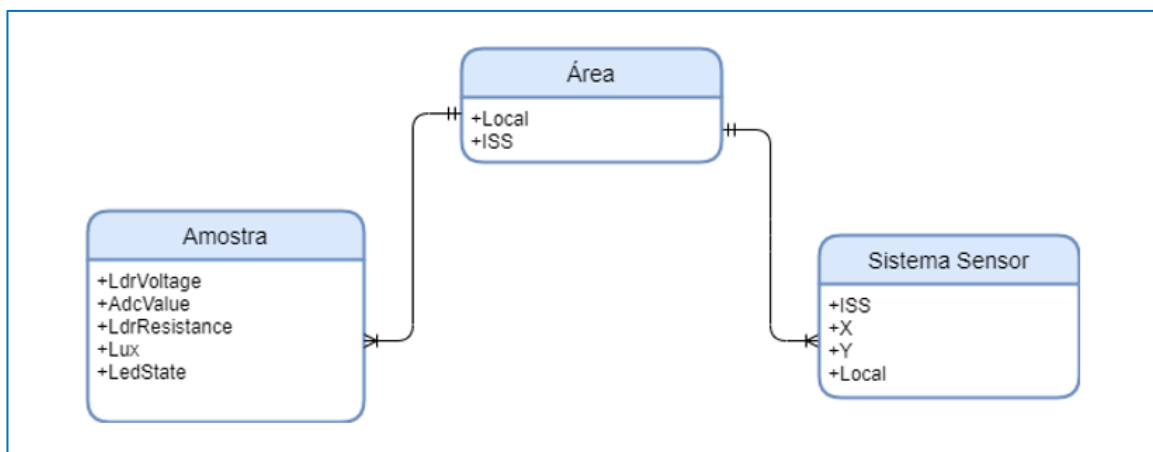


Figura 10 - Diagrama de Entidades e Relacionamentos.

### 3.3. Interface Web

O sistema central que vai ser implementado numa interface Web terá como principais funções a gestão do armazenamento de dados recebidos pelos vários concentradores assim como controlar o começo e término destes começarem a fornecer dados. Este também terá como função apresentar uma interface simples para que o utilizador consiga visualizar vários aspetos das várias áreas como gráficos dos luxs que permitem media a luminosidade e gráficos de movimentos que permitem saber quando houve movimento num dado local.

## Referências