



**Miguel Neves de Sousa Lopes**

Licenciado em Ciências da Engenharia Eletrotécnica e de Computadores

## **Iluminação Inteligente Um conceito de entretenimento**

Dissertação para Obtenção do Grau de Mestre em  
Engenharia Eletrotécnica e de Computadores

Orientador: Pedro Miguel Ribeiro Pereira,  
Professor Auxiliar, FCT-UNL

Co-orientador: João Francisco Alves Martins,  
Professor Associado, FCT-UNL

Júri:

Presidente: Prof. Doutor Paulo da Fonseca Pinto

Arguentes: Prof. Doutor Filipe Moutinho

Vogais: Prof. Doutor Pedro Miguel Ribeiro Pereira



FACULDADE DE  
CIÊNCIAS E TECNOLOGIA  
UNIVERSIDADE NOVA DE LISBOA

**Setembro, 2018**

[Verso da Capa]

## **Iluminação Inteligente – Um conceito de entretenimento**

*Copyright* © Miguel Neves de Sousa Lopes, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa.

A Faculdade de Ciências e Tecnologia e a Universidade Nova de Lisboa têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objetivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.



*Para a minha família e amigos,*

# Agradecimentos

Gostaria de iniciar os meus agradecimentos ao meu orientador, o Professor Pedro Miguel Pereira, por me ter dado a oportunidade de realizar este projeto e por todo o apoio e disponibilidade, como também todos os conselhos, sugestões, comentários e críticas extremamente oportunas e úteis para a realização desta dissertação.

Agradeço à instituição onde realizei todo o meu percurso académico, Faculdade de Ciências e Tecnologia e a todo o corpo docente do Departamento de Eng. Eletrotécnica por todo o conhecimento e ideias partilhadas que tanto contribuíram para a minha formação académica e pessoal.

Agradeço especialmente aos meus colegas de trabalho da sala 3.5, Andreia Ribeiro, João Simões, Pedro Lopes, Ricardo Ferreira e Rodrigo Francisco por toda a ajuda dada e por todos os momentos divertidos vivenciados. Também quero deixar um especial obrigado aos meus colegas e amigos de faculdade que sempre estiveram ao meu lado durante estes cinco intensos anos, os quais menciono seguidamente: Diogo Logrado, Manuel Faustino, Nuno Carvalhão, Pedro Estêvão e Pedro Martins.

Também quero deixar um especial agradecimento e destaque, a todos os membros da minha família, que sempre me apoiaram em todos os momentos, bons e menos bons, e por terem acreditado no meu sucesso.

A todos, um grande obrigado.

## Resumo

Assiste-se, nos dias de hoje, a um crescente desenvolvimento de aplicações tecnológicas que permitem a interação de um qualquer utilizador com um determinado sistema. Um dos sistemas que pode crescer significativamente, são os sistemas de iluminação. Os atuais sistemas de iluminação para as épocas festivas apresentam bastantes limitações no que diz respeito à interação com o público; muitas das vezes nem sequer é possível essa opção, estando todo o controlo do lado do operador do sistema.

Com a presente dissertação propõe-se a idealização um sistema de iluminação inteligente, baseado no conceito da *Internet of Things (IoT)*, que transforme por completo as épocas festivas de uma cidade, passando grande parte do controlo dos padrões luminosos para os habitantes.

O sistema é constituído por uma componente de hardware, empregando componentes de baixo custo e de elevada modularidade, facilitando a inclusão de novos dispositivos na rede e novas funcionalidades. Utiliza-se a plataforma *Arduino* como processador dos pontos luminosos e a tecnologia *iBeacon (Bluetooth 4.0 BLE)* para a comunicação. Para a iluminação, foi usada um tipo de iluminação *LED* controlada digitalmente (*WS2812B*), que permite a gestão individual de cada dígito *LED*.

A vertente de *software* consiste numa aplicação desenvolvida para telemóvel, com a qual o utilizador consegue consultar a localização dos pontos interativos num mapa, com os quais é possível estabelecer uma conexão direta e alterar os parâmetros associados ao sistema de iluminação com o qual se está a interagir, como, por exemplo, padrões luminosos personalizados.

Pretende-se, deste modo, dotar os atuais sistemas de iluminação, tradicionalmente implementados com tecnologia obsoleta, de tecnologia de baixo custo, proporcionando um elevado grau de interação com a população, para que se transforme toda a experiência das épocas festivas.

**Palavras-chave:** Iluminação Inteligente, Entretenimento, *Bluetooth Low Energy*, *IoT*, Aplicação Móvel, *Outsystems*.



## *Abstract*

Nowadays, there is an ever-increasing development of technological applications, which allows users some level of interaction with a system. One of the systems that has the most potential of growth are illumination systems. Current illumination systems designed for use in festive environments are quite limited when it comes to interaction with the public. This option is often not even possible, because all the of the system's control is on the administrator's side.

This dissertation proposes the idea of an intelligent illumination system based on the Internet of Things concept. The main objective of this project is to give a city's inhabitants the ability to connect to these interactive illumination devices and be able to control them and not the system administrator.

The system comprises of a hardware and software side. The components used are well established in the IoT market and have proven their worth. This system was designed to be modular and have low running costs. An Arduino Uno was chosen as the microcontroller for the illumination devices. For the communication, iBeacons were used (Bluetooth 4.0 BLE) for their functionality. Regarding the lighting, a digital LED device (WS2812B) was preferred, as it allows for more individual LED's to be addressed and controlled.

The software side consists of a mobile application, designed for smartphones, with which the user can check the location of the decorations on a digital map and quickly establish a direct connection, being able to change some of the system's parameters, such as custom lighting patterns, for example.

This master thesis aims to provide current illumination systems, traditionally built using obsolete technology, with some form of low-priced technology, while providing a high level of interaction with the population, so that the holiday experience is entirely transformed.

**Keywords:** Intelligent Illumination, Entertainment, Bluetooth Low Energy, IoT, Mobile Application, Outsystems.



# Conteúdo

<b>Agradecimentos .....</b>	<b>iv</b>
<b>Resumo .....</b>	<b>v</b>
<b>Abstract .....</b>	<b>vii</b>
<b>Conteúdo .....</b>	<b>ix</b>
<b>Índice de Figuras .....</b>	<b>xi</b>
<b>Lista de Siglas .....</b>	<b>xiii</b>
<b>1 Introdução.....</b>	<b>1</b>
1.1 Motivação.....	2
1.2 Contribuições .....	2
1.3 Estrutura da tese .....	3
<b>2 Estado de Arte .....</b>	<b>5</b>
2.1 <i>Internet of Things</i> e as Cidades Inteligentes .....	6
2.2 Interação homem-objeto.....	8
2.2.1 <i>Gamification</i> .....	8
2.2.2 Aplicações interativas existentes .....	10
2.3 Tecnologias de comunicação.....	16
2.3.1 <i>Wi-Fi HaLow</i> .....	16
2.3.2 <i>Bluetooth Low Energy (BLE)</i> .....	17
2.3.3 <i>ZigBee</i> .....	18
2.3.4 Redes móveis 5G.....	18
<b>3 Modelação do sistema .....</b>	<b>19</b>
3.1 Conceito .....	20
3.1.1 Aplicação/Utilizador.....	22
3.1.2 Microcontrolador .....	22
3.1.3 Módulo de transmissão de dados .....	23

3.1.4	Dispositivos de iluminação.....	23
3.1.5	Comunicação .....	23
3.2	Arquitetura .....	24
3.3	Funcionamento previsto .....	25
3.3.1	Disposição de pontos luminosos.....	25
3.3.2	Deteção de pessoas próximas .....	26
3.3.3	Encontrar pontos luminosos .....	26
3.3.4	Vertente <i>gaming</i> .....	27
3.3.5	Adicionar pontos luminosos .....	27
3.3.6	Ligação ao enfeite e interação .....	28
3.3.7	Deteção de falhas.....	28
<b>4</b>	<b>Implementação .....</b>	<b>31</b>
4.1	<i>Hardware</i> .....	32
4.1.1	Iluminação .....	32
4.1.2	Processamento .....	33
4.1.3	Módulo de comunicação.....	34
4.1.4	Circuito.....	36
4.1.5	Protótipo .....	37
4.2	Aplicação.....	38
4.2.1	Menu principal.....	39
4.2.2	Ecrã de procura aos pontos luminosos.....	40
4.2.3	Ecrã de ligação de pontos luminosos .....	41
4.2.4	Ecrã de interação com os pontos luminosos .....	42
4.3	<i>Outsystems</i> – Sequências da aplicação .....	44
4.3.1	Função <i>ScanDevices</i> .....	44
4.3.2	Função <i>Scan_iBeacons</i> .....	45
4.3.3	Funções <i>TrackPosition</i> e <i>GetPosition</i> .....	46
4.3.4	Funções dos padrões luminosos .....	47
<b>5</b>	<b>Testes e Resultados.....</b>	<b>49</b>
5.1	Testes e utilização da aplicação.....	50
5.2	Validação da aplicação e funcionamento do protótipo .....	54
<b>6</b>	<b>Conclusões e trabalho futuro .....</b>	<b>57</b>
6.1	Síntese e considerações finais .....	58
6.2	Trabalho futuro.....	59
	<b>Bibliografia .....</b>	<b>61</b>
	<b>Anexo 1 – Código Arduino .....</b>	<b>64</b>

## Índice de Figuras

Figura 2.1 - Ilustra a cadeia de comando de um medidor de energia [9].	8
Figura 2.2 - Dados da <i>Google</i> para pesquisas utilizando o termo 'gamification' [10].	8
Figura 2.3 - Exemplificação da interação entre utilizador e sistema. [12]	9
Figura 2.4 - Primeira versão do painel interativo [13].	10
Figura 2.5 - Algumas capturas de ecrã ilustrando os diferentes jogos propostos [13].	11
Figura 2.6 - Ecrã da escolha da refeição [13].	11
Figura 2.7 – Ilustração do parque com a iluminação <i>LED</i> [16].	12
Figura 2.8 - Mapa da atividade pedonal registada numa hora de funcionamento do sistema [16].	13
Figura 2.9 - Detalhe da luminária (esquerda) e da matriz <i>LED</i> (direita).	13
Figura 2.10 – Relação entre consumo e cenário [16].	14
Figura 2.11 - Protótipo do <i>Urban Pixel</i> [17].	15
Figura 2.12 - Montagem dos <i>Pixels</i> na fachada de um edifício. À direita, o sistema de fixação do módulo [17].	15
Figura 2.13 - À esquerda, um utilizador comuta os módulos com uma lanterna. Na direita pode-se observar uma das janelas da aplicação utilizada para definir a disposição luminosa dos módulos [17].	16
Figura 3.1 - Diagrama representativo de um Sistema Inteligente genérico.	21
Figura 3.2 - Ilustração da arquitetura do sistema.	24
Figura 3.3 - Diagrama de sequência da deteção de dispositivos próximos.	26
Figura 3.4 - Diagrama de sequência da pesquisa de pontos luminosos.	27
Figura 3.5 - Diagrama de sequência do processo de adicionar pontos à base de dados da cloud.	28
Figura 3.6 - Diagrama de sequência que mostra o processo de conexão e interação com o ponto luminoso.	28
Figura 3.7 - Diagrama de sequência do processo de deteção de falhas.	29
Figura 4.1 - Imagem amplificada dos díodos <i>LED</i> da fita <i>WS2812B</i> , à esquerda, e da fita <i>RGB</i> analógica, à direita.	32
Figura 4.2 - Ilustração da placa de desenvolvimento utilizada, <i>Arduino Uno</i> [25].	33

Figura 4.3 - Ilustração dos dois módulos estudados para o sistema. À esquerda encontra-se o módulo BLE, da Jinan Huanmao Co., e à direita, o módulo <i>Wi-Fi</i> , da Nordic Semiconductor, [27] e [28].	35
Figura 4.4 - Esquema elétrico das ligações entre componentes.	36
Figura 4.5 - Protótipo do <i>hardware</i> montado numa <i>breadboard</i> .	37
Figura 4.6 - Protótipo de uma árvore de Natal, um exemplo de um possível ponto luminoso interativo.	38
Figura 4.7 - Diagrama de casos de uso da aplicação.	39
Figura 4.8 – Ilustração do menu principal e do menu lateral.	39
Figura 4.9 - Ecrã de procura de enfeites.	40
Figura 4.10 - Ilustração do pop-up de dispositivo encontrado.	41
Figura 4.11 - Ilustrações do ecrã "Ligar a Dispositivos".	42
Figura 4.12 - Ilustração do ecrã de interação do enfeite luminoso.	43
Figura 4.13 - Diagrama de sequência da função <i>ScanDevices</i> do ambiente <i>Outsystems</i> .	44
Figura 4.14 - Diagrama de sequência da função <i>Scan_iBeacons</i> do ambiente <i>Outsystems</i> .	45
Figura 4.15 - Diagrama de sequência das funções de <i>TrackPosition</i> (esquerda) e <i>GetPosition</i> (direita) do ambiente <i>Outsystems</i> .	46
Figura 4.16 - Ilustração da função de envio de informação para o microcontrolador (esquerda) e lista de padrões luminosos (direita). Ambiente <i>Outsystems</i> .	47
Figura 5.1 - Diagrama <i>UML</i> de casos de uso da aplicação.	50
Figura 5.2 - Capturas de ecrã do menu principal à esquerda, ecrã de procura ao centro e de conexão aos pontos luminosos à direita.	51
Figura 5.3 - Capturas de ecrã da utilização da procura automática de pontos luminosos, recorrendo aos <i>iBeacons</i> .	51
Figura 5.4 - Capturas do ecrã de conexão com os pontos luminosos.	52
Figura 5.5 - Captura do ecrã de interação com o ponto luminoso.	53
Figura 5.6 - Ilustração do padrão luminoso "Neve cintilante" do dispositivo testado.	53
Figura 5.7 - Respostas à pergunta sobre a interface de utilizador da aplicação.	54
Figura 5.8 - Respostas à pergunta sobre o funcionamento da procura automática de pontos luminosos.	55
Figura 5.9 - Respostas à pergunta acerca da conexão ao ponto luminoso.	55
Figura 5.10 - Respostas à pergunta sobre a interação com o protótipo.	56
Figura 5.11 - Respostas à pergunta acerca da experiência de utilização no geral.	56

## Lista de Siglas

<i>AC</i>	<i>Alternated Current</i>
<i>AP</i>	<i>Access Point</i>
<i>BLE</i>	<i>Bluetooth Low Energy</i>
<i>D2D</i>	<i>Device to Device</i>
<i>DC</i>	<i>Direct Current</i>
<i>LED</i>	<i>Light Emitting Diode</i>
<i>LTE</i>	<i>Long Term Evolution</i>
<i>GSM</i>	<i>Global System for Mobile Communications</i>
<i>GPS</i>	<i>Global Positioning System</i>
<i>I2C</i>	<i>Inter-Integrated Circuit</i>
<i>IDC</i>	<i>International Data Corporation</i>
<i>IDE</i>	<i>Integrated Development Environment</i>
<i>IEEE</i>	<i>Institute of Electrical and Electronics Engineers</i>
<i>IoIT</i>	<i>Internet of Important Things</i>
<i>IoT</i>	<i>Internet of Things</i>
<i>IR</i>	<i>Infrared</i>
<i>IRC</i>	<i>Índice de Reprodução de Cor</i>
<i>MIT</i>	<i>Massachusetts Institute of Technology</i>
<i>RF</i>	<i>Rádio Frequência</i>
<i>RGB</i>	<i>Red, Green, Blue</i>
<i>SMS</i>	<i>Short Message Service</i>
<i>SPI</i>	<i>Serial Peripheral Interface</i>
<i>UML</i>	<i>Universal Modelling Language</i>



# 1 Introdução

Os atuais sistemas de iluminação para as épocas festivas são bastante limitativos no que diz respeito à interação com a população. Estes sistemas dividem-se em duas partes essenciais: o controlador do sistema (*hardware*), que pode ser um pequeno microcontrolador ou mesmo um computador, e a aplicação de comando (*software*), um painel de controlo que permite configurar o funcionamento do sistema de iluminação.

Com o surgimento da *Internet of Things* foi possível dar a capacidade de conexão à internet de todo um conjunto de dispositivos. A proliferação destes dispositivos inteligentes ajudou a criar uma infraestrutura que permite a sua interligação em rede, associados a um conjunto de serviços de análise de dados. Num estudo realizado pela Gartner, estima-se que aproximadamente 20,5 mil milhões de dispositivos irão estar conectados com uma fatia de mercado de quase 3 biliões de dólares [1].

As autarquias pretendem melhorar os sistemas de iluminação para as épocas festivas (festas populares, feriados municipais, entre outras), surgindo, assim, a procura por um sistema inteligente que permita aos habitantes interagirem diretamente com a iluminação. Toda a infraestrutura associada ao sistema de iluminação apresenta custos elevados de aquisição e funcionamento para os municípios, pelo que apenas alguns possuem capacidade financeira para os adquirir.

## 1.1 Motivação

Atualmente, a proliferação de dispositivos com capacidade de computação é uma realidade incontornável. Assiste-se a uma transformação da vida cidadina, pretendendo-se mudar as cidades de modo a servir os seus cidadãos, não o contrário. Os municípios esforçam-se para tornar o ambiente citadino o mais cativante possível para os seus habitantes, especialmente em épocas festivas e festas populares.

Numa altura em que a esmagadora maioria das pessoas possui um telemóvel com elevada capacidade de processamento de informação e de comunicação, vulgarmente conhecido por *smartphone*, não é assim tão descabido pensar que é por aqui que se pode pensar num modo de fazer a interligação cidadão-cidade.

O grande motor desta mudança é sem dúvida a *Internet of Things*. De entre as mais diversas áreas de aplicação, nesta tese dar-se-á dar um especial enfoque aos setores de iluminação e de turismo. A necessidade de melhorar estes sistemas de iluminação festivos manifesta-se pela elevada ineficiência dos mesmos e a falta de capacidade de comunicação e interação com a população.

## 1.2 Contribuições

Com o desenvolvimento desta tese, pretende-se modelar a arquitetura para a implementação de uma solução de iluminação interativa, adequada às épocas festivas e desenvolver um protótipo de iluminação para demonstrar o funcionamento do conceito, a parte de *hardware* e a parte de *software*.

Relativamente ao *hardware* do protótipo, como desenvolvimento pretende-se escolher os diversos componentes e integrá-los, nomeadamente o microcontrolador, os módulos de comunicação rádio e os dispositivos de iluminação. Pretende-se manter o custo do protótipo o mais baixo possível, de modo a ser uma alternativa interessante para as autarquias, do ponto de vista financeiro.

Para a parte de *software*, pretende-se desenvolver uma aplicação que irá ser disponibilizada gratuitamente aos utilizadores, e lhes permitirá procurar por pontos luminosos distribuídos pela cidade e conectarem-se aos mesmos, usando os seus telemóveis. Uma vez ligados poderão alterar os padrões luminosos exibidos.

Para além destas funções fundamentais de pesquisa e ligação, a aplicação conta com algumas funcionalidades adicionais de modo a tornar a experiência de interação intuitiva. Existe um mapa de navegação integrado na aplicação onde os principais pontos luminosos se encontram assinalados. É possível ao utilizador adicionar à base de dados algum ponto que descubra e não

se encontre assinalado. Pretende-se também dar uma vertente *gaming* à procura de pontos luminosos, na medida em que são atribuídos pontos e um *ranking* aos utilizadores mais ativos na procura.

### 1.3 Estrutura da tese

Para além deste primeiro capítulo de introdução, a presente dissertação está estruturada de seguinte forma:

No segundo capítulo, Estado de Arte, abordam-se os temas essenciais ao desenvolvimento desta tese, nomeadamente, a *Internet of Things (IoT)* aplicada às cidades inteligentes, algumas aplicações já existentes que demonstram a interação entre pessoa e tecnologia. Por último, referem-se alguns protocolos de comunicação mais comuns a aplicações *IoT*.

No terceiro capítulo descreve-se o modelo do sistema de iluminação inteligente, esclarece-se o conceito do sistema, seguido de uma breve explicação da função das partes que o constituem. Apresenta-se a arquitetura prevista e as relações entre os blocos constituintes do sistema. Por último, abordam-se as principais funções previstas para o sistema e para a aplicação para o telemóvel.

No capítulo 4 encontra-se a implementação do sistema, dividida em duas partes fundamentais, *hardware* e *software*. Numa primeira instância, abordam-se os vários componentes escolhidos para a construção do protótipo e os motivos que levaram à sua escolha. Em segundo, analisa-se a aplicação desenvolvida para o telemóvel, sendo feita uma breve explicação de cada um dos ecrãs e das suas funcionalidades.

No capítulo 5 apresenta-se o método usado para validar os resultados obtidos. Testou-se a utilização da aplicação no telemóvel em conjunto com o protótipo de um dispositivo de iluminação inteligente, recorrendo a um pequeno questionário respondido pelos utilizadores após um breve contacto com o conjunto do sistema de iluminação.

Por último, no sexto capítulo, retiram-se algumas conclusões acerca do projeto desenvolvido na presente dissertação e perspetivados alguns trabalhos futuros que se baseiem neste trabalho.



## 2 Estado de Arte

Neste capítulo, apresenta-se uma breve introdução do tema proposto na presente dissertação sobre ‘Iluminação Inteligente’, um conceito de entretenimento.

Introduz-se o conceito da ‘*Internet of Things*’ e da sua importância para o progresso e evolução das cidades inteligentes. Realiza-se uma breve análise à expressão inglesa ‘*gamification*’, frequentemente associada ao desenvolvimento de aplicações móveis e *software*, como são aplicados os elementos na interface do utilizador e o motivo de ser um termo cada vez mais cunhado pelos programadores.

Na secção seguinte, analisam-se três soluções que foram estudadas e, posteriormente, implementadas e testadas em situação de uso real, de modo a exemplificar a relação entre a tecnologia e a interação humana. Dois *case-studies* estão intimamente relacionados com a iluminação em ambiente urbano, enquanto que um dos casos foi analisado pelo impacto que a interação com a tecnologia provoca nos indivíduos em estudo.

Por último, estudam-se algumas das tecnologias e protocolos de comunicação que são mais relevantes para a aplicação em cidades inteligentes e que melhor se adequam à utilização integrada com dispositivos móveis, nomeadamente, o *Wi-Fi*, *Bluetooth* e redes móveis.

## 2.1 *Internet of Things* e as Cidades Inteligentes

O termo '*Internet of Things*' (abreviado para *IoT*) foi proposto por Kevin Ashton, um pioneiro tecnológico, em 1999 e desde então, tem vindo a ganhar cada vez mais impacto no mundo da tecnologia. O nome '*Internet of Things*' apresenta duas vertentes: uma orientada para a ligação em rede (*network*) e a outra para os objetos a serem integrados na infraestrutura. Ao juntar estas duas palavras, criou-se um termo que designa a rede de dispositivos interligados, baseada em determinados protocolos de comunicação [2].

Atualmente, a maioria dos produtos que saem das linhas de produção trazem tecnologia que lhes permite estabelecer uma ligação sem fios, de maneira a ser possível interagir com esse mesmo aparelho à distância. Alguns autores fazem distinção entre *IoT* e *IoIT*, sigla inglesa para Internet das Coisas Importantes, assumindo que à partida, a maioria dos dispositivos não se encontrarão conectados, apenas os que influenciam positivamente os seres humanos [3].

A *IoT* assume-se como um paradigma recente de comunicação digital. Prevê que num futuro próximo, ocorra um crescimento explosivo de objetos banais equipados com processadores e rádios, dando-lhes capacidade de poder processar informação e transmiti-la. Esta intercomunicação pode ser realizada entre aparelhos ou diretamente com o utilizador [2].

Existem uma infinidade de aplicações da *IoT*, desde eletrodomésticos inteligentes conectados a toda uma cidade, as denominadas '*Smart Cities*'. É possível abranger quase todos os serviços públicos que temos ao nosso dispor, como por exemplo, a saúde, energia, gestão de resíduos, entre outros [3].

Ultimamente, os países e as suas câmaras municipais têm apostado em tornar mais eficientes e ecológicos os seus concelhos e freguesias, como uma forma de reduzir o impacto ambiental e diminuir os custos operacionais e simultaneamente, aumentar o índice de qualidade de vida da população.

Através do conceito '*urban IoT*', é possível encontrar um método de dotar uma cidade de inteligência, através da implementação de uma rede de sensores que informe as entidades responsáveis por um serviço, acerca do estado de um certo recurso. Este conceito abre um vasto leque de opções para a administração de uma cidade. Através da partilha de informação com os habitantes é possível sensibilizar a população para um determinado aspeto da sua cidade e estimular os cidadãos a intervir mais na administração [4]. Este conceito também abrange a parte lúdica da *Smart City*, relacionada com a implementação de aplicações de realidade aumentada, parques infantis interativos, entre outros [3].

Atualmente, esta noção de *Smart City* ainda está numa fase de desenvolvimento precoce, ou seja, é necessário um grande investimento por parte dos organismos administrativos para a integração destes sistemas na cidade ou mesmo adaptar as infraestruturas existentes a esta nova realidade. Trata-se de um mercado com enorme potencial de investimento [5]. Segundo um estudo

da empresa *IDC (International Data Corporation)*, o mercado emergente associado à *IoT* está avaliado acima de cinco mil milhões de euros em 2020.

Estamos perante uma excelente oportunidade para serem criadas empresas que se dediquem ao desenvolvimento de novas soluções de *IoT* em conjunto com os governantes de modo a tornar possível esta visão de *Smart City*. Cabe agora às câmaras disponibilizarem capital para investir neste mercado e impulsionar a difusão da *IoT*.

Estamos, portanto, a dirigirmo-nos para um futuro onde tudo está interligado. Deste modo, é necessário proceder à sua regulamentação, para garantir o seu correto funcionamento e impedir que haja uma fragmentação do mercado ao nível destes dispositivos.

Chegamos então ao reverso da medalha da *IoT*. Ao existirem tantos dispositivos interligados e alguns deles associados a serviços críticos, como garantir a segurança e privacidade dos utilizadores? A *IoT* permite que várias coisas estejam conectadas e em monitorização constante, logo informação e dados considerados privados são recolhidos automaticamente. Devido à elevada incidência de ataques nestes dispositivos, é fácil perceber que é um problema a ser resolvido tão depressa quanto possível [6].

De acordo com um estudo sobre tendências da *IoT*, a maioria dos dispositivos ligados à Internet tem mais de 30% de probabilidade de ser alvo de intrusão, sendo que os dispositivos *wearable (smartwatches e bandas fitness)*, fechaduras inteligentes e equipamentos de vigilância são os mais prováveis de serem utilizados para obter acesso ao sistema [7]. No caso dos *smartwatches*, compreende-se o receio das pessoas em adotar estas novas tecnologias, uma vez que sem o nível adequado de segurança é possível obter dados sensíveis do utilizador, sem o próprio se aperceber do ataque.

Esta falta de segurança nestes dispositivos deve-se exclusivamente ao facto de terem fraco poder de processamento e a maioria é alimentada por bateria, pelo que assegurar uma ligação encriptada *end-to-end* (desde dispositivo até ao utilizador) consome imensos recursos do sistema [8]. Visto que uma grande percentagem de dispositivos é de baixo custo e computação, estamos perante um grave problema.

Na Figura 2.1, situada na página seguinte, observa-se que através de um ataque ao *Smart Meter* que não esteja devidamente protegido, pode permitir o acesso à base de dados e por sua vez, ao centro de comando. Daqui é possível controlar a produção de energia.

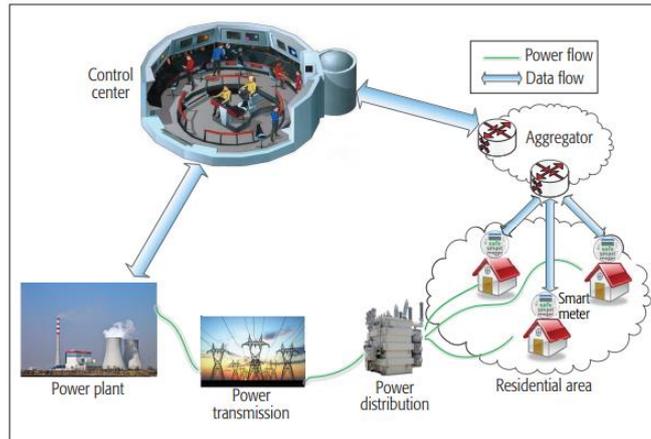


Figura 2.1 - Ilustra a cadeia de comando de um medidor de energia [9].

Ao passar-se para o domínio da cidade inteligente, é necessário garantir a segurança de toda a rede, caso contrário, um simples sensor de trânsito ou de luminosidade, pode permitir a um utilizador mal-intencionado ganhar acesso à central de comando e a partir de aí inutilizar o sistema e até mesmo pôr em causa a integridade física das pessoas.

## 2.2 Interação homem-objeto

### 2.2.1 Gamification

Na última década tem-se assistido a uma proliferação do uso do termo *gamification* em múltiplos sectores económicos. Segundo uma análise de tendências da *Google*, pesquisas com recurso à palavra '*gamification*' registaram um crescimento pronunciado a partir de 2010. Ilustrado na Figura 2.2.

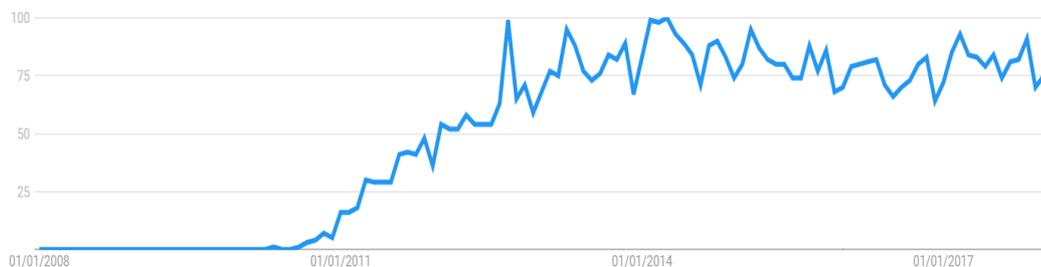


Figura 2.2 - Dados da *Google* para pesquisas utilizando o termo 'gamification' [10].

Existe pouca clareza no que toca à definição do termo ‘*gamification*’. Os limites entre *gamification* e conceitos relacionados não estão bem delineados. Contudo, é possível explicar este termo recorrendo a dois componentes chave: é utilizado para fins não lúdicos e retira inspiração de jogos, isto é, alguns elementos básicos que compõem um jogo [11].

Na teoria, o objetivo de uma aplicação recorrendo à *gamification* é encorajar a alteração comportamental dos utilizadores, seja na participação, no desempenho ou complacência [11].

Na prática, cabe a quem desenvolve a aplicação escolher que elementos quer incluir na aplicação e deve sempre ter em vista os três pilares essenciais: mecânica, dinâmica e estética do sistema. Fazendo uma breve descrição de cada termo, a vertente mecânica diz respeito à representação dos dados (pontos, troféus), a componente dinâmica afeta o comportamento temporal da mecânica, relativo aos inputs dos jogadores (nível de conclusão e escolhas) e, por último, a estética corresponde às respostas emocionais quando interagem com o ambiente “gamificado” (ser desafiado, noção de pertencer à comunidade, por exemplo) [12].

Na Figura 2.3, pode-se observar como se processa o funcionamento da aplicação. Através de um conjunto de determinadas ações, como por exemplo, ao 25º início de sessão ou à criação de um produto novo, o utilizador recebe pontos ou desbloqueia medalhas, permitindo o utilizador subir no *ranking* da comunidade.

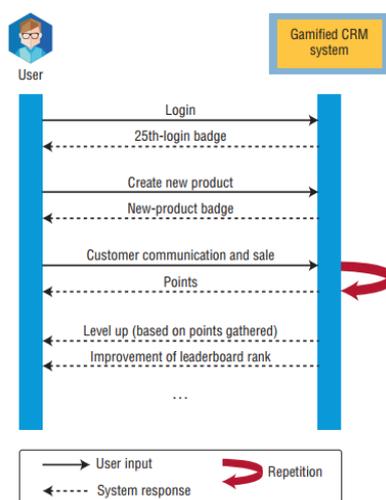


Figura 2.3 - Exemplificação da interação entre utilizador e sistema. [12]

Assim pode-se observar como este conceito de ação-recompensa pode ter um impacto enorme na utilização de uma aplicação. Ao oferecer ao utilizador um conjunto de recompensas por realizar certas ações, podemos influenciar o seu comportamento, podendo levá-lo a usar a aplicação durante mais tempo. Isto pode ser uma boa medida, caso o autor da aplicação esteja a rentabilizá-la com publicidade. Deste modo, através de perguntas educativas ou através de desbloqueio de funções adicionais, o utilizador é levado inconscientemente a querer permanecer mais tempo na aplicação.

## 2.2.2 Aplicações interativas existentes

A presente dissertação propõe-se a desenvolver uma aplicação para dispositivos móveis, onde na qual se pretende a interação entre pessoa e objeto, esta secção tem como objetivo dar a conhecer algumas soluções já existentes e mostrar o efeito provocado nos utilizadores, derivado do seu uso.

### 2.2.2.1 Painel interativo para crianças em ambiente hospitalar

A hospitalização pode ser uma experiência traumática, especialmente em crianças. De modo a tornar mais tolerável o internamento, o Instituto Científico de San Raffaele (HSR) desenvolveu uma solução que explora as potencialidades da *IoT* com o objetivo de ajudar as crianças a suportar alguns problemas decorrentes da permanência prolongada em hospitais [13].

Numa primeira aproximação é necessário contextualizar o motivo que levou ao desenvolvimento deste produto. Segundo um estudo realizado por Dynnete Hart e Elizabeth Bossert que contou com a participação de 82 crianças entre os 8 e 11 anos de idade, chegou-se à conclusão que os três medos específicos que mais as afligiam eram: estar afastados da família, serem vacinadas e estarem internadas por um longo período [14].

De modo a combater a angústia da hospitalização, desenvolveu-se este painel interativo para distrair as crianças durante o internamento e através de jogos didáticos estimular o seu desenvolvimento. Aqui entra o conceito de *IoT*. Assim é possível melhorar a vertente funcional de objetos e ambientes através da integração de tecnologia de comunicação e sensores que permita recolher informação em redor [15].

Na Figura 2.4 está ilustrado um painel de 26 polegadas sensível ao toque, que está equipado com tecnologia *Wi-Fi*. Integra a dois altifalantes, permitindo aos utilizadores ouvir sons e áudio e ainda contém uma pequena câmara de vídeo, utilizada principalmente para reconhecimento facial e obtenção de alguns dados, como por exemplo, o tempo de uso do dispositivo e o número de utilizadores diário [13].



Figura 2.4 - Primeira versão do painel interativo [13].

Todas as crianças internadas na ala pediátrica do hospital podem utilizar o painel. O conteúdo exibido está agrupado de acordo com a idade do utente, desde a pré-primária até ao secundário. O sistema foi desenhado para detetar a aproximação da pessoa e mostrar automaticamente o conteúdo personalizado para a idade, apresentando alguns jogos didáticos para estimular o desenvolvimento cognitivo do jovem (consultar Figura 2.5) [13].



Figura 2.5 - Algumas capturas de ecrã ilustrando os diferentes jogos propostos [13].

Além de toda a oferta educativa elaborada pelos investigadores do projeto, também é possível à criança escolher a refeição do dia seguinte recorrendo a uma aplicação especial. O sistema reconhece a criança e mostra-lhe os diferentes menus que pode escolher (ver Figura 2.6) O conceito de *gamification* foi implementado, na medida em que é avaliada a capacidade da criança escolher uma refeição equilibrada e seguir os conselhos nutritivos apresentados. Caso se confirme a escolha saudável, a mascote premeia a criança e é passado um episódio curto de desenhos animados adaptados à idade e género [13].



Figura 2.6 - Ecrã da escolha da refeição [13].

Embora este projeto não esteja diretamente relacionado com o tema desenvolvido nesta tese, evidencia a interação entre pessoa e objeto e como esta experiência pode ter um impacto positivo ou menos positivo nela. Dependendo da forma como a aplicação é construída e se apresenta, é possível contribuir positivamente para o bem-estar e desenvolvimento do utilizador.

### 2.2.2.2 Iluminação urbana interativa

Nesta subsecção, aborda-se uma experiência realizada no âmbito da iluminação pública num parque da cidade de Aalborg, Dinamarca. Na Figura 2.7 está ilustrado o *layout* do espaço do parque, com a entrada para o metro situada no centro da figura, iluminada com tons amarelados.



Figura 2.7 – Ilustração do parque com a iluminação *LED* [16].

Nos últimos anos, tem-se assistido a uma crescente pressão para tornar mais eficaz e eficiente o uso sustentável de energia. As autarquias têm feito um esforço para substituir as luminárias existentes por *LED*. Para além do gasto energético mais reduzido e maior tempo de vida útil, estas lâmpadas permitem o controlo da intensidade luminosa de forma mais simples, apresentam valores de IRC (Índice de Reprodução de Cor) mais altos, construção mais robusta e de volume reduzido. O facto desta tecnologia ser pouco volumosa, permite uma melhor integração em espaços públicos causando o mínimo impacto visual para as pessoas.

O estudo consistiu em analisar o impacto causado nas pessoas pela iluminação interativa, bem como os possíveis ganhos energéticos da instalação.

Através de um sistema de visão que utilizou três câmaras térmicas (Axis Q-1921-E) montadas a 15m, desenvolveu-se um algoritmo para calcular a posição e velocidade dos indivíduos no espaço. Estes dados foram utilizados como parâmetros para controlar o comportamento das lâmpadas.

Na Figura 2.8, situada na página seguinte, está ilustrado um mapa de atividade obtido após a análise dos dados obtidos com as câmaras térmicas. Na figura observam-se os vetores de deslocamento das pessoas adquiridos durante uma hora de funcionamento.

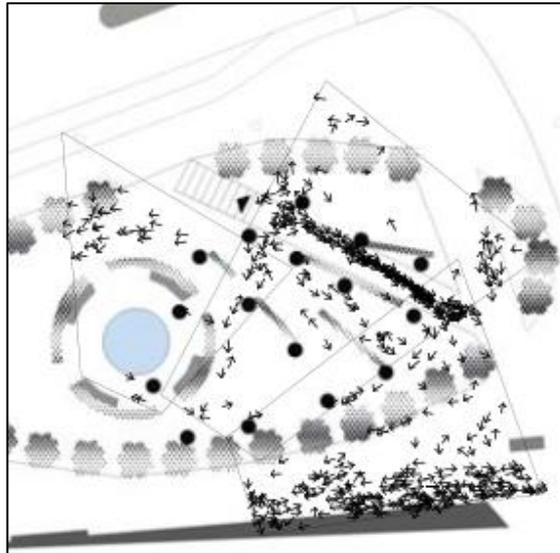


Figura 2.8 - Mapa da atividade pedonal registada numa hora de funcionamento do sistema [16].

O módulo *LED* presente à direita na Figura 2.9 é composto por 18 lâmpadas *LED* de 1 W, com seis para cada cor (*RGB*). O círculo interior contém os seis *LED* vermelhos, enquanto que no exterior os *LED* verdes e azuis estão dispostos alternadamente [16].



Figura 2.9 - Detalhe da luminária (esquerda) e da matriz *LED* (direita)

No âmbito do projeto de iluminação pública do parque em Aalborg, foram realizados quatro ensaios com níveis distintos de iluminação [16]:

1. **Ambient Illumination** – Distribuição homogénea da luz pelo parque com o nível de intensidade estabelecido em 80%. Corresponde à instalação típica.
2. **Glowing Light** – Este cenário tinha como objetivo dar vida ao parque. Mantendo o nível de luz para constante, algumas luminárias desligavam e ligavam gradualmente de forma aleatória.
3. **White Aura** – Nesta configuração, as luminárias acendiam conforme a pessoa se deslocava pelo parque. Era mantido o nível de iluminação mínimo pelo espaço e um círculo com cerca de 10m em redor das pessoas, como se lhe apontassem um foco de holofote.

4. **Red Treasure Hunt** – Este é um cenário de entretenimento. Pretende-se tornar o parque num jogo de caça ao tesouro. Existe uma luminária mais forte que as restantes que desencadeia uma reação de onda luminosa radial quando a pessoa se aproxima.

Uma conclusão interessante a que se chegou foi a de que a maioria das pessoas que atravessava o parque não se apercebia dos efeitos luminosos implementados. Também é referido que a caça ao tesouro não suscitou a participação do público tanto quanto era esperado [16].

Na Figura 2.10 está registado o consumo durante três horas do parque de acordo com o cenário em questão. Como seria expectável, o cenário 1 foi o mais gastador (230 W), pois tinha todas as luzes ligadas a 80%. Os ensaios 3 e 4 foram os mais comedidos com o terceiro a registar apenas um consumo de 20 W.

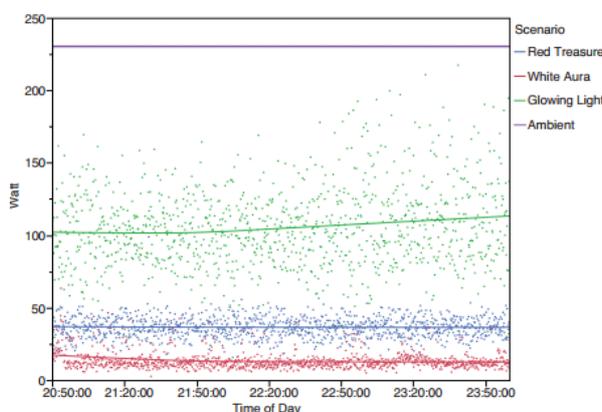


Figura 2.10 – Relação entre consumo e cenário [16].

Esta alternativa à iluminação tradicional aparenta ser uma excelente opção. Foi possível reduzir o consumo energético do parque sem alterar o comportamento das pessoas que o utilizavam no dia a dia. Esta interação entre luzes e pessoas tem um vasto leque de aplicações e modos de implementar. Com este projeto, mostrou-se que é possível reduzir o gasto energético e, simultaneamente, montar um espaço lúdico e entusiasmante.

### 2.2.2.3 Urban Pixels

Este projeto foi desenvolvido no âmbito das cidades inteligentes por um grupo no MIT, numa tentativa de combater a proliferação dos grandes painéis digitais recorrendo a pequenos módulos luminosos integráveis em rede com os quais é possível interagir diretamente. As condições impostas pela equipa à construção do produto final foram a fácil montagem, independência da rede elétrica, ser responsivo e não ter restrições de implementação. A premissa deste projeto é modelar uma rede sem-fios de pixéis em espaço urbano [17]. Na Figura 2.11 apresenta-se a ilustração do primeiro protótipo desenvolvido.



Figura 2.11 - Protótipo do *Urban Pixel* [17].

Cada unidade apresenta uma construção simples e com poucos componentes eletrônicos. A tratar do processamento existe um microcontrolador CC1010 que integra um transceptor RF (433 MHz). A parte luminosa fica encarregue de um módulo *LED* com dez díodos brancos. É integrada uma bateria de íões de lítio de 3.7 V com 66 Ah e um pequeno painel fotovoltaico. Para evitar o funcionamento em períodos elevada luminosidade, uma fotocélula *IR* (infravermelho) controla a operação do módulo. Toda esta parte elétrica é envolta com duas meias esferas (10 cm de diâmetro) em acrílico, uma delas com tratamento fosco branco para dispersar melhor a luz [17].

Em termos de protocolos de comunicação, é utilizada a frequência 433 MHz para os *Pixels* comunicarem com um *hub* central que lhes envia a informação necessária. Através de um portátil e de uma porta RS-232, é possível programar toda a rede. É necessário referir que os *Pixels* não sabem da existência uns dos outros e cada um tem um endereço único e exclusivo. Foi a forma que a equipa encontrou para evitar a reprogramação sempre que se alterava a disposição dos *Pixels* (exemplificada na Figura 2.12) [17].

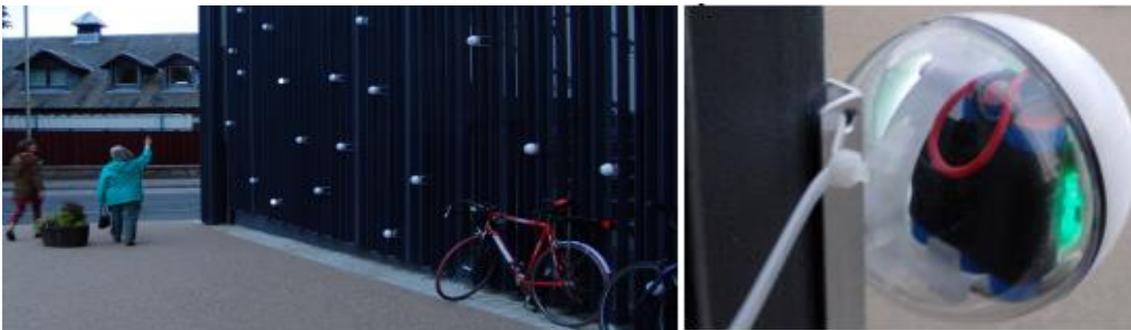


Figura 2.12 - Montagem dos *Pixels* na fachada de um edifício. À direita, o sistema de fixação do módulo [17].

Na configuração inicial do projeto, era possível interagir com a rede de *Pixels* de três maneiras distintas. Uma delas é com o portátil, já referido, reservado para os instaladores. Para o público em geral, recorrendo a um *SMS* enviado pelo *smartphone* para um *modem GSM* ou a uma lanterna, seria possível alterar os padrões exibidos ou comutar módulos [17].

Estas duas formas de interação produziu um efeito bastante positivo na população, especialmente a da lanterna. A relação causa-efeito facilitava uma conexão mais pessoal entre a pessoa e o *Pixel*, isto é, apontava-se o foco para o módulo e este, ora acendia, ora apagava. Com o telemóvel tinha-se mais controlo, mas a pessoa nunca tinha a perceção de quando a sua mensagem já tinha sido processada pelo sistema. Havendo muitos pedidos do mesmo padrão, os tempos de permanência eram longos, ao invés de uma transição fluida entre padrões [17]. Na Figura 2.13 encontra-se a ilustração das duas formas de interagir com os módulos.

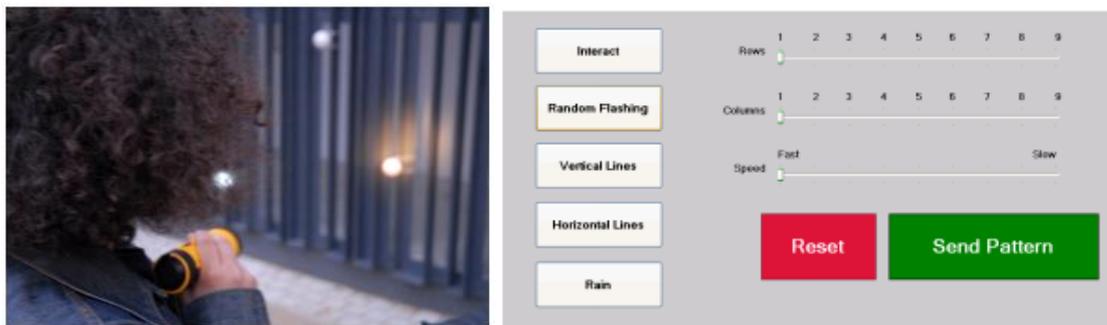


Figura 2.13 - À esquerda, um utilizador comuta os módulos com uma lanterna. Na direita pode-se observar uma das janelas da aplicação utilizada para definir a disposição luminosa dos módulos [17].

Este projeto mostra o enorme potencial de uma aplicação deste tipo em ambiente festivo, por exemplo o Natal. Recorrendo a outros meios de comunicação mais rápidos, aplicações específicas para telemóvel e *hardware* mais avançado é possível levar este sistema a um patamar mais elevado, desbloqueando novas formas de a população interagir com os *Pixels* ou outro produto semelhante.

## 2.3 Tecnologias de comunicação

A implementação destes produtos está imensamente dependente de tecnologia de comunicação sem fios para troca de dados/comandos entre objeto e o utilizador. A *IoT* prevê o crescimento exponencial de dispositivos nos próximos anos e é necessário que os protocolos de comunicação atuais se adaptem a esta nova realidade.

Nesta secção irão ser abordados alguns protocolos que poderão ser utilizados no desenvolvimento da solução proposta nesta tese.

### 2.3.1 *Wi-Fi HaLow*

O standard 802.11 no qual se baseia a tecnologia *Wi-Fi* já conta com quase 20 anos de existência. De forma a continuar a ser uma tecnologia competitiva devido ao surgimento da *IoT*,

o consórcio *IEEE* propôs a criação de um apêndice ao standard existente. Esta norma, designada por *Low-Power Wi-Fi* (802.11ah) [18], pode ser vista como um híbrido entre o *Bluetooth* e redes *LTE*, isto é, pretende-se combinar a eficiência energética do *Bluetooth* com o longo alcance e taxas de transmissão de dados elevadas típicos das redes móveis [19].

Esta tecnologia opera a frequências próximas de 900 MHz de modo a conseguir uma melhor propagação do sinal através de obstáculos. Conta com uma taxa de transmissão e alcance máximos de 78 Mbps e 1000 m, respetivamente. Bastante inferior às taxas de 433 Mbps a 1 Gbps (úteis) da norma 802.11ac. Contudo esta opera a 2,4 GHz e 5 GHz e tem um consumo energético bastante mais alto, comparativamente a outras tecnologias de comunicação [19].

Do ponto de vista de organização dos elementos da rede, a grande mudança surge com a utilização de *relay access points* (*relay AP*) que ajudam a comunicação de dispositivos que estejam fora do alcance do *AP* principal. Também conta com alguns serviços que permitem um menor gasto energético para aplicações a bateria [19].

### 2.3.2 *Bluetooth Low Energy (BLE)*

O *Bluetooth* é uma tecnologia baseada no standard *IEEE* 802.15.1 para comunicações sem fios de curta distância em configuração *Master-Slave*, onde um *Master* pode ter até sete *Slaves* ligados numa rede denominada por *piconet* [20]. Opera na frequência do *Wi-Fi*, ou seja, a 2,4 GHz. Na altura em que esta norma foi proposta, seria uma alternativa à ligação com fios entre dispositivos. A energia consumida não foi uma prioridade no desenvolvimento da primeira iteração da tecnologia [18].

De modo a tornar concretizável a implementação das redes *IoT*, o grupo responsável pela evolução do standard introduziu o *BLE*, que corresponde à versão 4.0 da norma, sacrificando a velocidade e distância de transmissão de dados em prol do consumo de energia bastante inferior ao *Bluetooth* clássico [21].

Uma das fraquezas tecnológicas do *Bluetooth* clássico e um dos motivos que levou ao aparecimento do *BLE* é de não permitir que se estabeleça uma ligação ativa entre mais de dois aparelhos [18]. Os dois protocolos estabelecem ligações de forma diferente. Uma vez que o *Bluetooth* clássico está orientado para aplicações multimédia com elevado volume de dados, o emparelhamento entre dispositivos central e periférico é obrigatório. Em contraste, os sensores *BLE* enviam pacotes de dimensões reduzidas, embora com mais frequência. Uma das grandes vantagens é não ser necessário o emparelhamento com o dispositivo central [21].

Para além das claras aplicações possíveis para redes de sensores, uma das mais interessantes é a capacidade que o *BLE* tem em detetar a proximidade de um *beacon BLE*. Assim, pode-se colocar um destes *beacons* num objeto parado ou em movimento e utilizar a função de localização espacial para desencadear um evento para tornar a interação entre utilizador-objeto o mais fluída possível [21].

Existe uma nova versão do *Bluetooth* que já se encontra incluída nos aparelhos móveis mais recentes. O *Bluetooth 5.0* é uma evolução em termos de alcance máximo, cerca de quatro vezes superior e taxas de transmissão duas vezes superiores. Este novo standard irá permitir uma maior difusão de dispositivos compatíveis com *IoT* [22].

### 2.3.3 *ZigBee*

O protocolo *ZigBee* foi lançado em 2003. É mantido e distribuído por um grupo de empresas denominado de *ZigBee Alliance*. Tem como base o standard *IEEE 802.15.4* e é adequado para redes de curto alcance (<100m) com taxas de transmissão baixas [23].

Uma das características chave do *ZigBee* é a formação de redes em malha (*mesh networks*), tendo suporte para redes em estrela (um nó central e vários periféricos) ou em árvore. A rede em malha apresenta redundância de funcionamento intrínseco ao *design* da rede. Caso um nó falhe, a rede reajusta-se para assegurar a transmissão correta dos dados ponto a ponto. Outra particularidade é a facilidade de expansão da rede, visto que um nó coordenador suporta 65536 dispositivos ligados [20].

Tal como a tecnologia *Bluetooth*, o *ZigBee* faz uma gestão avançada de energia, permitindo um tempo de operação longo a dispositivos alimentados por bateria. Os dispositivos possuem dois modos de funcionamento, *beacon* e *non-beacon*. No primeiro, os dispositivos emitem sinais periódicos para confirmar a sua existência na rede. Quando não estão a transmitir podem comutar para um estado de *sleep* para conservar o máximo de energia. Os *non-beacon* estão continuamente em funcionamento [23].

### 2.3.4 **Redes móveis 5G**

Para aplicações *IoT* nas quais não seja possível a ligação à *Internet* pelos meios tradicionais (*Wi-Fi*, cabo), uma rede móvel tem de ser utilizada. Espera-se que a nova geração de redes móveis, 5G, tenha suporte para o cada vez maior número de dispositivos ligados à *Internet* e melhor gestão de recursos [18].

Esta rede tem como objetivo principal a aplicação em áreas de funcionamento críticas e de resposta em tempo real, nomeadamente, controlo de processos, veículos e deslocação a alta velocidade. O standard atual, 3GPP, já prevê a utilização para dispositivos *IoT*, mas pretende-se que a rede 5G se desenvolva de raiz para suportar a *IoT* de baixo consumo [18].

Onde esta tecnologia perde para as outras é no consumo energético bastante mais elevado para volumes de dados baixos, mas frequentes. Tendo em vista que o objetivo principal desta tecnologia é o longo alcance de transmissão (dezenas de quilómetros) e elevada largura de banda [18], explica-se este elevado consumo de energia.

### 3 Modelação do sistema

Neste terceiro capítulo introduz-se o modelo do sistema inteligente de iluminação e o comportamento e funcionalidades esperadas, idealizado na presente dissertação.

Na secção 3.1 realiza-se a explicação do conceito pensado para o sistema inteligente de uma forma genérica, que posteriormente será aplicada no *case-study* proposto nesta tese, aplicado à iluminação festiva. Para simplificar a compreensão do sistema, dividiu-se o mesmo nos seus blocos primários e analisou-se o seu funcionamento.

Na secção seguinte, 3.2, apresenta-se a arquitetura do sistema de iluminação com base na descrição dada na secção anterior. Mostra-se um diagrama *UML* que ilustra as relações entre os blocos do sistema, bem como algumas das principais ações, resultantes do funcionamento normal do sistema.

Por último, na secção 3.3 descreve-se o funcionamento previsto para o sistema, ou seja, apresentam-se algumas das funções que foram pensadas para implementação, por exemplo, como se processa a disposição de pontos luminosos, como é feita a ligação e como se encontram dispositivos compatíveis, entre outras.

### 3.1 Conceito

Os atuais sistemas de iluminação utilizados nas épocas festivas são muito básicos no que diz respeito ao número de funcionalidades. A esmagadora maioria opera de uma forma muito simples, apenas sendo possível comutar entre os estados ligado ou desligado. Outros sistemas apresentam alguns modos de operação que lhes permitem melhor atrair a atenção das pessoas, por exemplo, através da mudança de cor ou dos padrões luminosos. De um modo geral, apenas é possível observar os ditos sistemas clássicos à distância, sendo que grande parte das vezes, os padrões alteram-se de forma automática e são previamente escolhidos pelo projetista, uma vez que qualquer forma de interação com esses pontos luminosos é inexistente.

Com o projeto desenvolvido, pretende-se dar o passo seguinte em termos de evolução dos sistemas de iluminação. Propõe-se a criação de uma arquitetura que possa ser transversal e adaptável a inúmeras aplicações que envolvam interação entre utilizador e objeto. No caso específico estudado na presente dissertação, considerou-se que o objeto é um sistema de iluminação adequado para utilização em épocas festivas, embora se possa considerar a hipótese de ser adaptado a outros espaços recreativos.

Pretende-se desenvolver uma solução que seja atrativa do ponto de vista financeiro, apresentando um custo o mais baixo possível e que seja modular, de modo a permitir a adição de tantos pontos luminosos quantos desejados pelo projetista encarregue. Portanto, deverá ser realizada a integração e encapsulamento de todos os componentes necessários para o seu funcionamento.

Oferecendo uma solução completamente integrada e de baixa manutenção, permite-se que quem utilize este sistema mobilize rapidamente os dispositivos de iluminação e os possa adaptar a qualquer contexto e local, devido à enorme versatilidade do sistema de iluminação em estudo nesta tese.

Um sistema inteligente típico ramifica-se em três camadas distintas: *software*, *hardware* e a camada de *cloud computing*. A camada de *software* permite realizar a gestão e confere algum grau de inteligência ao sistema, sendo utilizada para controlar diretamente a parte correspondente de *hardware*, composta pela unidade de processamento e todos os sensores e atuadores necessários ao correto funcionamento do sistema. A terceira camada, de *cloud computing*, confere ao sistema a possibilidade de tornar o conjunto de ‘*software + hardware*’ mais leve, do ponto de vista tecnológico. Ao invés de se tratar a informação localmente, esta é enviada para a *cloud* e posteriormente é analisada e tratada consoante a finalidade desejada. Estes dados vão desde a valores lidos por sensores a algumas estatísticas de utilização do sistema.

Pretende-se utilizar este conceito de sistema inteligente e adaptá-lo às necessidades do projeto proposto por esta dissertação, onde se pretende adaptar um sistema inteligente ao controlo de dispositivos de iluminação.

Na Figura 3.1 encontra-se o diagrama genérico representativo de um sistema inteligente, que engloba as três camadas discutidas no parágrafo anterior.

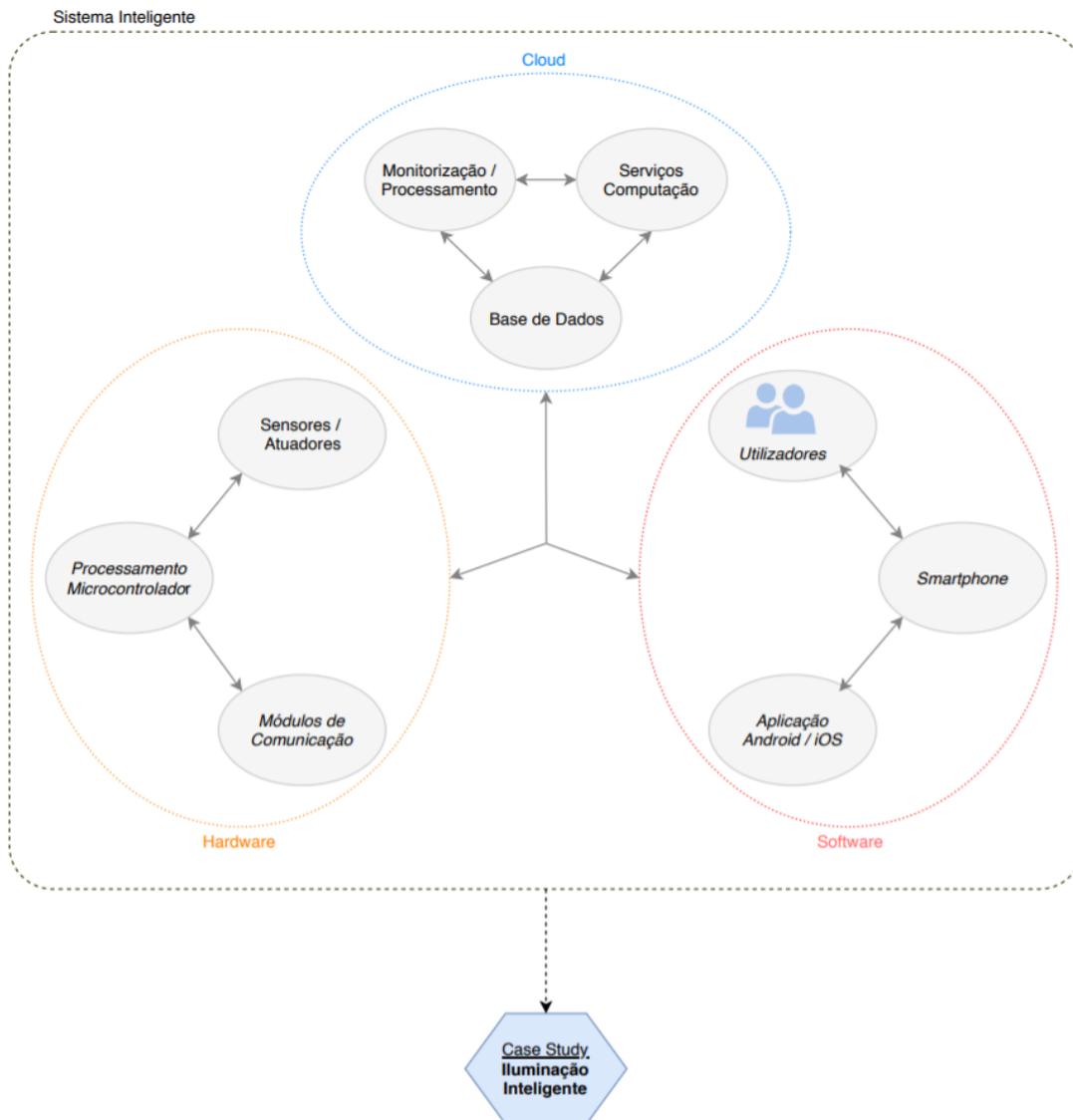


Figura 3.1 - Diagrama representativo de um Sistema Inteligente genérico.

Na figura acima, observam-se alguns dos serviços e módulos que integram as três camadas discutidas anteriormente. A camada de *hardware*, composta pelo microcontrolador, sensores e módulos de comunicação, comandada pelo *software*, neste caso, uma aplicação desenvolvida para telemóvel. A componente da *cloud* reúne os dados recolhidos durante a operação do sistema e posteriormente realizada a sua análise.

Com este conceito de sistema inteligente proposto, pretende-se realizar a sua adaptação ao projeto a realizar na presente tese. Um sistema inteligente de iluminação, no qual o utilizador possa interagir livremente com pontos luminosos distribuídos em ambiente citadino.

Disponibilizando uma ferramenta que transforme os telemóveis num comando remoto, cada utilizador irá ter a capacidade de decidir que cores e padrões desejam ver exibidos e ligar-se em rede. Antes de aprofundar mais o conceito e para simplificar a sua compreensão, dividiu-se o sistema global nos seus constituintes fundamentais, esclarecendo a sua função nas sub-secções seguintes.

### 3.1.1 **Aplicação/Utilizador**

A interação direta com os pontos luminosos apenas é possível mediante a utilização de uma aplicação desenvolvida especialmente para o efeito. Com a aplicação instalada no telemóvel, o utilizador pode conectar-se aos pontos luminosos e interagir com eles diretamente, percorrendo todas as funções disponíveis.

Esta apresenta diversos sub-menus nos quais o utilizador pode usar as funcionalidades disponibilizadas, desde ativar a procura automática de pontos luminosos enquanto se observa num mapa onde se encontram localizados, a realizar a conexão ativa ao enfeite. Caso seja bem-sucedida, a aplicação abre um novo ecrã exclusivo de cada enfeite, no qual é possível ao utilizador escolher entre alguns padrões pré-definidos ou, caso o pretenda, desenhar o seu próprio padrão luminoso e armazená-lo em memória local no telemóvel ou partilhar com outros utilizadores.

A aplicação estará disponível para dispositivos com sistemas operativos baseados em *Android* e *iOS*, que apresentam a maior cota de mercado dos *smartphones*.

### 3.1.2 **Microcontrolador**

O microcontrolador tem como propósito fazer a ponte entre o módulo de comunicação e o dispositivo de iluminação e desempenhar todas as funções de gestão relativas ao *hardware*.

Está programado para descodificar a informação enviada para o módulo de comunicação e consoante a escolha do utilizador, correr uma determinada função que desencadeará um padrão luminoso escolhido pelo utilizador.

O microcontrolador verifica se existem dispositivos nas proximidades do enfeite e, caso não encontre nenhum, ativa-se um temporizador que, passado um intervalo de tempo pré-definido, coloca todo o sistema num estado de baixo consumo. Assim que seja detetado um utilizador, é desencadeada uma animação aleatória para despertar a atenção do utilizador. Pretende-se, assim, automatizar este processo, de modo a que não seja precisa qualquer intervenção do gestor do sistema.

Para o administrador, é possível averiguar a ocorrência de falhas ou maus-funcionamentos, bem como a consulta de estatísticas de utilização e controlar individualmente cada enfeite.

### 3.1.3 Módulo de transmissão de dados

O módulo de comunicação escolhido terá de desempenhar duas funções distintas:

1. Será responsável por aceitar a ligação com o smartphone por meio da aplicação e receber todos os comandos enviados pelo utilizador.
2. Tendo em vista a redução do consumo energético do sistema, o microcontrolador utilizará as funções de pesquisa do módulo de comunicação para detetar a existência e proximidade de dispositivos móveis, indicativo da presença de uma pessoa, e desencadear uma determinada função do sistema.

É necessário realizar um estudo posterior para averiguar qual das tecnologias melhor se adapta às necessidades do projeto.

### 3.1.4 Dispositivos de iluminação

Um sistema de iluminação, tal como o nome indica, é um projeto de natureza essencialmente visual, carecendo de uma reflexão cuidada e uma especial atenção na escolha do tipo de tecnologia de luminárias, seguindo alguns critérios essenciais. Atualmente, os produtores mundiais de lâmpadas estão a avançar no sentido de difundir e democratizar a utilização da tecnologia *LED*, sendo raras as aplicações que não a utilizem, nomeadamente as de muito alta potência, como as utilizadas em estádios, entre outras.

Como o tema da eficiência energética é um dos capítulos mais importantes num cálculo luminotécnico, faz todo o sentido a escolha da tecnologia *LED* para ser integrada no projeto proposto nesta dissertação. Outro dos critérios essenciais para a iluminação é ser totalmente programável, isto é, terá de ser possível escolher, individualmente para cada *LED*, a sua cor e intensidade luminosa.

### 3.1.5 Comunicação

A comunicação entre componentes do sistema inteligente foi idealizada para ser dispositivo para dispositivo (*D2D – Device to Device*), ou seja, a informação é enviada diretamente da aplicação para o módulo de transmissão do microcontrolador. Escolheu-se este tipo de abordagem, ao invés do envio de informação para um sistema de gestão central e depois deste para o microcontrolador, por permitir uma rapidez de comunicação mais elevada, consumo de menos recursos tecnológicos e implementação menos dispendiosa e mais célere, devido à elevada modularidade do sistema.

Periodicamente, haverá o envio de informação de cada ponto luminoso para a *cloud*, na qual existirá um conjunto de serviços que analisarão os dados recebidos e apresentá-los-ão de forma a comunicar o estado de cada dispositivo acerca de alguns parâmetros como taxa de utilização ou existência de falhas.

### 3.2 Arquitetura

De modo a melhor ilustrar algumas das funcionalidades do sistema de iluminação, apresenta-se, na página seguinte, a Figura 3.2. Recorreu-se à linguagem de modelação *UML (Universal Modelling Language)*, mais concretamente a um *Activity Diagram*.

Neste diagrama de ações ilustram-se as diferentes interações entre os vários constituintes do sistema de iluminação inteligente, bem como algumas das suas ações principais, apresentados na secção 3.1. Algumas funcionalidades previstas podem não estar presentes na Figura 3.2, no entanto, serão apresentadas e discutidas de forma mais completa na secção 3.3.

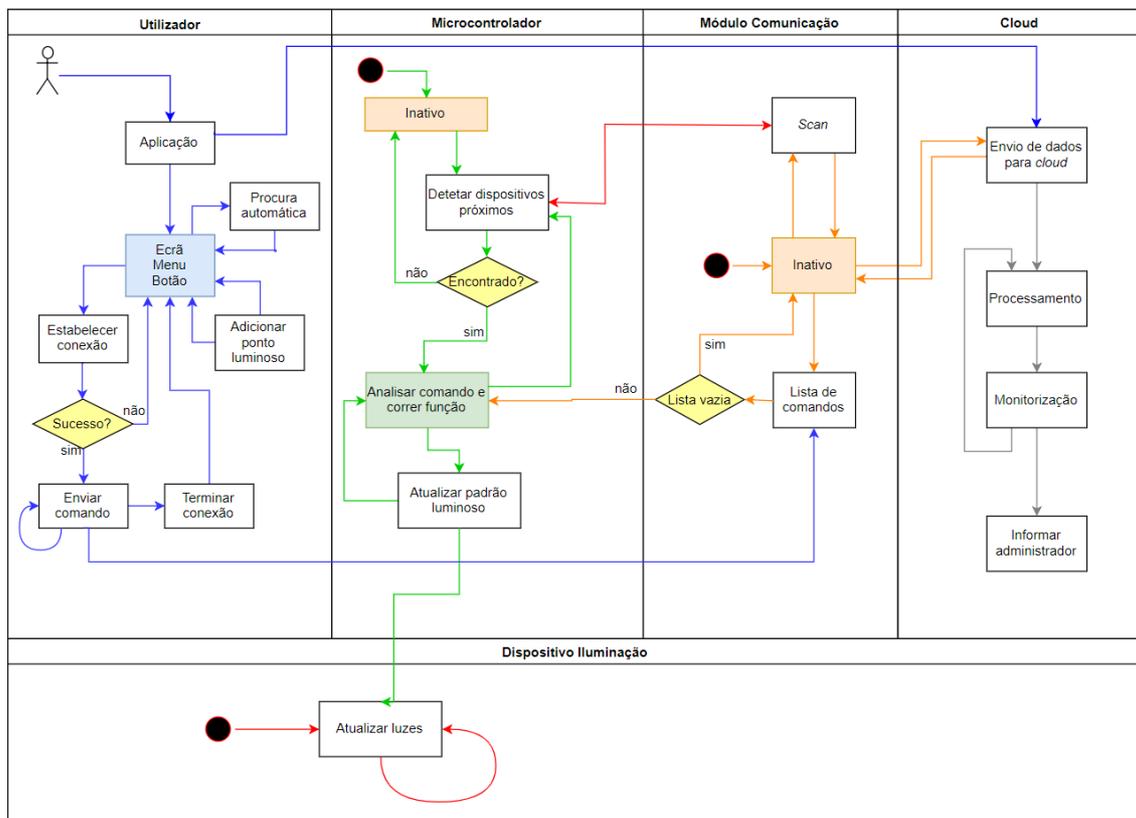


Figura 3.2 - Ilustração da arquitetura do sistema.

Na Figura 3.2, em cima, encontram-se os cinco principais constituintes do sistema, o conjunto utilizador e aplicação, o microcontrolador, o módulo de comunicação, o dispositivo de iluminação e a *cloud*.

Do lado do utilizador, através da aplicação, é possível através do ecrã apropriado, ativar a procura automática de dispositivos, conectar-se ao ponto luminoso e enviar os comandos que alteram os padrões luminosos exibidos, estes são recebidos pelo módulo de comunicação através

da seta correspondente. Realçado a azul está o bloco central que indica a posição de espera por ações do utilizador.

Relativamente ao microcontrolador, este apresenta dois estados essenciais, um de inatividade, a laranja, onde se assume um estado de baixo consumo energético e o estado de funcionamento normal, assinalado a verde. Neste modo, o microcontrolador, verifica se existem comandos pendente no *buffer* de memória do módulo de comunicação e executa as funções necessárias para os padrões luminosos. Periodicamente, certifica-se que existem dispositivos móveis (representando utilizadores) nas imediações e caso nenhum seja detetado, é assumido o estado de baixo consumo até serem detetados novos utilizadores.

O dispositivo de iluminação apresenta um único estado, pois recebe a informação necessária do microcontrolador para iluminar de uma forma específica os díodos *LED*.

Os módulos de comunicação apresentam um conjunto de funções/serviços, dependendo da tecnologia utilizada (*Bluetooth*, *Wi-Fi* ou redes móveis), disponíveis para o microcontrolador. Entre elas, encontra-se a função de pesquisa, que permite averiguar a presença de dispositivos nas proximidades, utilizado para desencadear o estado de inatividade do microcontrolador para a diminuição do consumo energético. O módulo apresenta um registo que guarda em memória os comandos enviados pela aplicação, que posteriormente o microcontrolador os verifica e processa.

Por último, a cloud recebe os dados do ponto luminoso por meio do módulo de comunicação e também diretamente da aplicação. O tratamento e processamento desta informação serve para se obter as estatísticas de utilização e de operação. O administrador do sistema é informado, na eventualidade de existir algum mal funcionamento dos aparelhos no terreno.

### **3.3 Funcionamento previsto**

Nesta secção, ir-se-á pormenorizar o funcionamento idealizado para o sistema, integrando os elementos abordados na secção 3.1, consoante a arquitetura apresentada na Figura 3.2. Escolheu-se abordar as funções mais relevantes do sistema, descrevendo-se como se desenvolvem e apresentando-se os diagramas de sequência das mesmas para melhor ilustrar o processo.

#### **3.3.1 Disposição de pontos luminosos**

Cada ponto luminoso contém todos os componentes necessários ao seu funcionamento autónomo. Contém um microcontrolador, módulo de comunicação e de iluminação, sendo que este último varia consoante o tipo de enfeite e o local onde está colocado.

Será possível formar uma rede que interligará todos os pontos luminosos, permitindo ao administrador do sistema o controlo total à distância ou local para os desligar, tendo em vista a realização de operações de manutenção.

Uma vez que está prevista a ligação em malha dos dispositivos, o administrador do sistema poderá desencadear uma animação especial recorrendo a alguns dos pontos luminosos que se encontram num determinado local, por exemplo, num evento de abertura do espaço. Todos os enfeites estarão, à partida, disponíveis para os utilizadores e a sua localização colocada numa base de dados para ser disponibilizada na aplicação, fora os que não se encontram assinalados, com o objetivo de serem os utilizadores a descobri-los num jogo de “caça ao tesouro”.

### 3.3.2 Detecção de pessoas próximas

De modo a tornar o consumo tão baixo quanto possível, o sistema irá adotar um estado de funcionamento ocioso no qual cessa todas as funções de conexão do módulo de comunicação e vai alternando os padrões luminosos aleatoriamente num brilho mais reduzido.

Periodicamente, será chamada uma função que reativa o módulo de comunicação e este executa a função de pesquisa de dispositivos compatíveis. Se detetar um smartphone, é mostrado um padrão luminoso específico para chamar a atenção da pessoa que se encontra perto. Caso contrário, é retomado o estado de consumo reduzido. O módulo de comunicação passa a funcionar em pleno, aceitando, novamente, ligações externas.

Na Figura 3.3 está representado o diagrama de sequência desta ação.

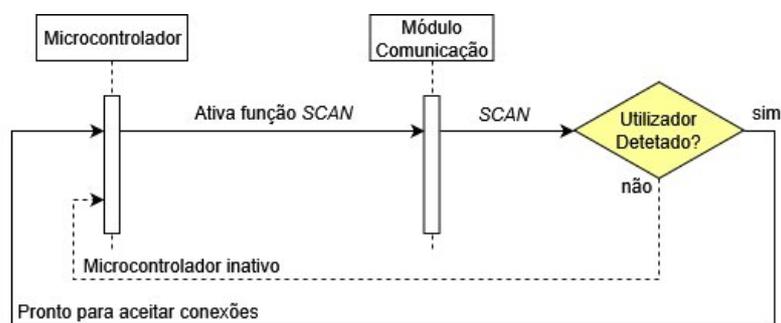


Figura 3.3 - Diagrama de sequência da detecção de dispositivos próximos.

### 3.3.3 Encontrar pontos luminosos

Relativamente ao processo de procura de pontos luminosos, existirá um sub-menu na aplicação que mostra um mapa que permite ao utilizador saber a sua posição atual e consultar a localização dos pontos luminosos na sua proximidade.

Nesse ecrã, é possível ativar o modo de pesquisa através de um botão. Com este modo, a aplicação procura automaticamente dispositivos compatíveis em *loop*, até o utilizador desativar o modo. Assim que deteta um, o telefone vibra e é mostrada a notificação. O utilizador é reencaminhado para o ecrã de ligação ao ponto luminoso. Na Figura 3.4 está representado a sequência desta ação.

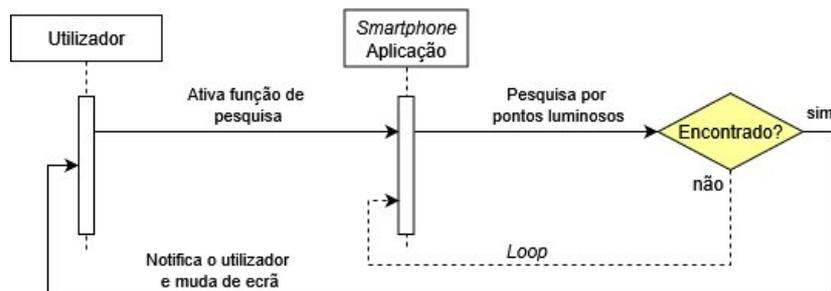


Figura 3.4 - Diagrama de sequência da pesquisa de pontos luminosos.

### 3.3.4 Vertente gaming

Uma vez que este sistema apresenta uma vertente de entretenimento, pensou-se em aplicar um conceito de jogo de “caça ao tesouro”, com o objetivo de tornar o sistema mais apelativo para os utilizadores mais jovens, através de um sistema de pontuação. Os participantes que decidirem participar nesta caça ao tesouro, deverão realizar um registo na aplicação para que os pontos sejam contabilizados e o seu resultado ser disposto numa tabela de classificação.

Deste modo, por cada dispositivo de iluminação encontrado que não esteja assinalado no mapa presente na aplicação, serão atribuídos alguns pontos ao utilizador. Ocasionalmente também será apresentado uma pequena pergunta de cultura geral, que atribuirá ainda mais pontos.

Pretende-se assim que, no fim do desafio, seja atribuído um prémio simbólico aos utilizadores que acumularem um maior número de pontos.

### 3.3.5 Adicionar pontos luminosos

Prevê-se que o sistema inteligente de iluminação proposto seja composto por vários pontos luminosos. Contudo, apenas alguns estarão assinalados, à partida, no mapa disponível na aplicação, sendo que fica à responsabilidade do administrador do sistema definir esses mesmos dispositivos. Cada utilizador, durante a utilização da função de pesquisa, terá a possibilidade de adicionar à base de dados da aplicação os dispositivos ainda não encontrados.

Esta funcionalidade de adicionar pontos luminosos está integrada na função de procura automática descrita na sub-secção 3.3.3, na medida em que o utilizador a ativa e caso se encontre perto de um ponto luminoso é notificado. A aplicação trata de verificar se esse dispositivo já foi previamente identificado e se este se encontra na base de dados. Verificando-se essa condição, a aplicação retoma o funcionamento normal e apresenta o ecrã de ligação. Se não, avisa o utilizador que encontrou um ponto luminoso novo, devendo por isso pressionar o botão apresentado no ecrã.

A sincronização do mapa da aplicação é automática, aparecendo o dispositivo no mapa, ficando disponível para os restantes utilizadores. Na Figura 3.5 está representado o diagrama de sequência da ação acima descrita.

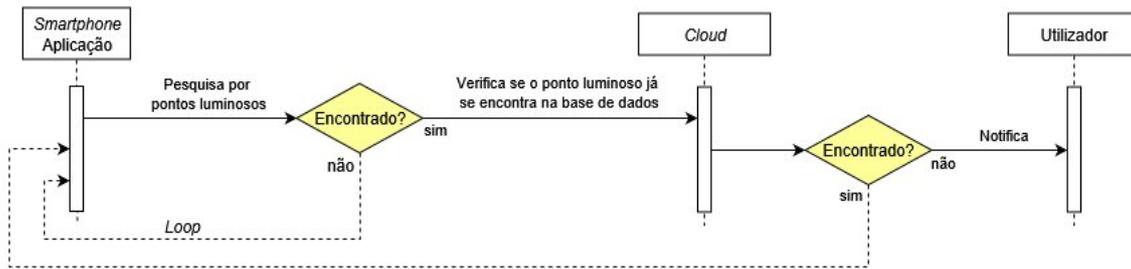


Figura 3.5 - Diagrama de sequência do processo de adicionar pontos à base de dados da cloud.

### 3.3.6 Ligação ao enfeite e interação

No seguimento da ação descrita na sub-seção 3.3.3, apresenta-se ao utilizador um novo ecrã, onde se mostra uma lista dos pontos luminosos nas proximidades que estejam a aceitar novas conexões. Para se ligar o utilizador deverá clicar no dispositivo com o qual pretende conectar.

Se a ligação ao dispositivo for bem-sucedida, a aplicação acede ao ecrã de interação com o ponto luminoso. Neste ecrã serão mostrados todos os efeitos e padrões que é possível enviar para o dispositivo. Dada a quantidade e variedade de dispositivos que poderão vir a existir, cada um contará com padrões exclusivos.

Na Figura 3.6 encontra-se o diagrama de sequência que ilustra todo o processo de ligação ao enfeite e interação.

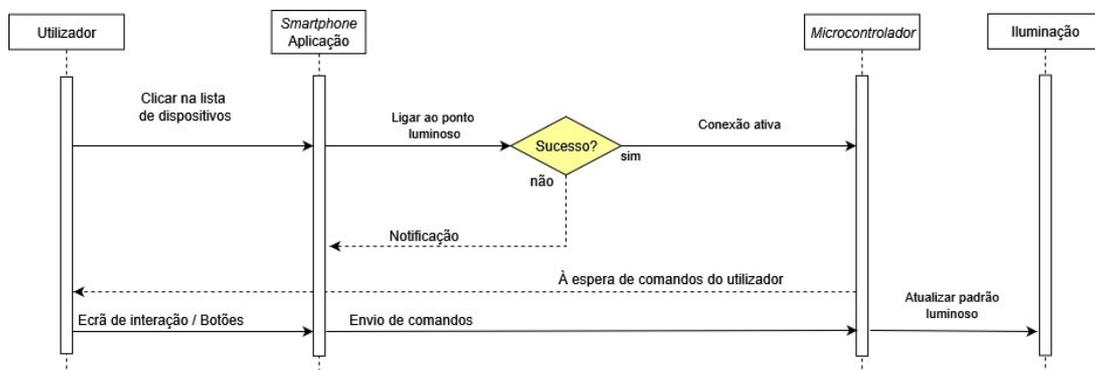


Figura 3.6 - Diagrama de sequência que mostra o processo de conexão e interação com o ponto luminoso.

Haverá um menu onde será possível desenhar um padrão personalizado, segundo diversos parâmetros escolhidos pelo utilizador. Este pode, posteriormente, guardar esse padrão localmente ou partilhar com outros utilizadores numa plataforma da cloud desenvolvida para o efeito.

### 3.3.7 Detecção de falhas

Uma vez que os pontos luminosos enviam, periodicamente, informação acerca do seu estado de funcionamento e das taxas de uso diárias, pretende-se utilizar os serviços da cloud para

processar estes dados e fornecer aos administradores do sistema informação útil para manter o sistema a funcionar em boas condições e efetuar a manutenção de algum ponto luminoso em situação de operação anómala.

Recorrendo a tecnologias de análise de dados como, por exemplo, algoritmos genéticos ou redes neuronais, é possível dotar o sistema de alguma inteligência, permitindo a deteção atempada de falhas do sistema que possam ocorrer e notificar os responsáveis pela manutenção. Na Figura 3.7 está ilustrado o diagrama de sequência da deteção de falhas no sistema.

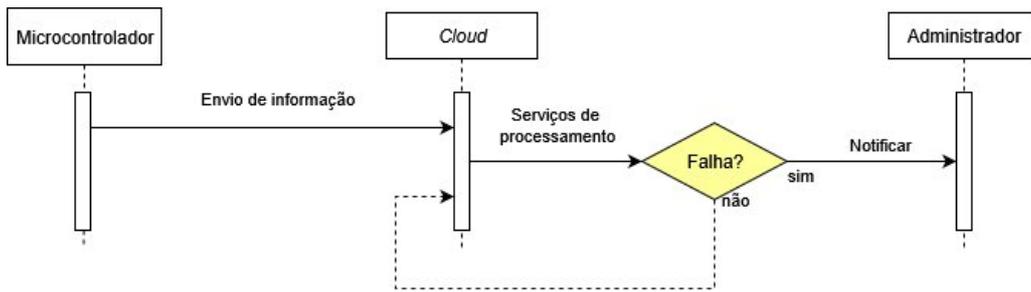


Figura 3.7 - Diagrama de sequência do processo de deteção de falhas.



## 4 Implementação

Neste quarto capítulo descreve-se todo o método de trabalho realizado na implementação das funcionalidades apresentadas no capítulo 3. O trabalho de implementação foi dividido em duas partes fundamentais, *hardware* e *software*.

Numa primeira instância, analisam-se todos os componentes físicos utilizados para a construção do protótipo de um enfeite luminoso. A secção 4.1 está subdividida em subsecções onde se estudam os diferentes componentes a serem integrados no protótipo final: iluminação, processamento e módulo de comunicação. Em cada subsecção discutem-se os motivos que levaram à escolha de determinado componente, tendo em conta algumas alternativas existentes no mercado.

Seguidamente, na secção 4.2 realiza-se uma descrição da aplicação desenvolvida para o telemóvel que será utilizada para encontrar e conectar-se aos pontos luminosos. Analisam-se os principais ecrãs disponíveis para o utilizador e como se utilizam as funções descritas na secção 3.3.

## 4.1 Hardware

Para a construção do sistema de iluminação e de processamento foram analisadas algumas tecnologias-chave para a escolha dos componentes integrantes do protótipo, bem como alguns critérios, como por exemplo, eficiência e custo.

### 4.1.1 Iluminação

Dado que este projeto é de natureza visual, a escolha do tipo de iluminação a ser utilizada é bastante importante. Sendo o tema “eficiência energética” o principal motor no desenvolvimento de soluções inteligentes de iluminação, pode-se, à partida, pôr de parte todas as tecnologias de iluminação que não sejam baseadas em *LED*, uma vez que a sua relação eficiência/custo e durabilidade é das mais altas do mercado.

Um dos formatos de iluminação para enfeites mais utilizados são as fitas *LED*. Estas apresentam uma facilidade de utilização e versatilidade enormes. As duas principais tecnologias existentes são a analógica e a digital. A primeira apresenta o custo mais reduzido, sendo apenas possível escolher uma única cor do espectro *RGB* para toda a fita, ou seja, é possível visualizar todas as cores do espectro visível ao olho humano.

Por outro lado, as fitas *LED* digitais contém um pequeno *chip* controlador que gere a iluminação de cada lâmpada *LED*, permitindo assim endereçá-la individualmente, controlando o brilho e a cor. Dependendo do modelo do *chip*, a forma de se ligar à fita e o envio de informação apresenta algumas diferenças importantes para o projetista do sistema.

Observando a Figura 4.1, encontra-se um exemplo de cada uma das fitas digital e analógica. À esquerda, vê-se o *LED* com o controlador integrado (parte negra ao centro do díodo). Este faz com que o preço da fita aumente, mas abre um leque de possibilidades enorme.

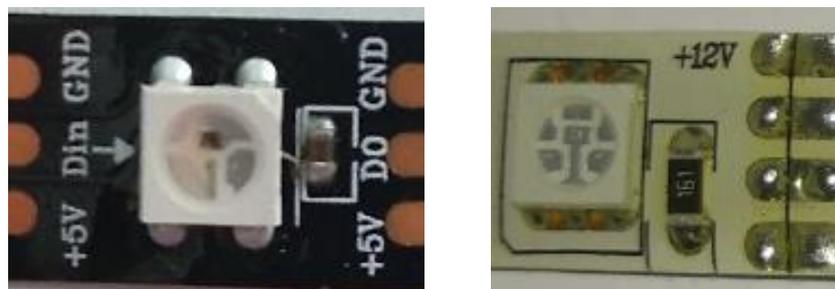


Figura 4.1 - Imagem ampliada dos díodos *LED* da fita WS2812B, à esquerda, e da fita *RGB* analógica, à direita.

No caso do modelo *WS2812B*, combina-se um *LED RGB* e um controlador programável de corrente constante. Consultando a *datasheet* do controlador [24], constata-se que cada *LED* apresenta um total de 4 pinos, dois de alimentação e dois de transmissão de dados, um de entrada ( $D_{in}$ ) e outro de saída ( $D_{out}$ ), em que o pino  $D_{out}$  do *LED n* se liga ao pino  $D_{in}$  do *LED n+1*.

A ligação do *bus* de comunicação ( $D_{xx}$ ) é unidirecional em cascata e apresenta intervalos extremamente precisos de temporização, sendo necessário um microcontrolador externo para enviar a informação corretamente para o *chip*.

Uma vez que se pretende construir um protótipo de iluminação, no qual se pretende implementar algumas animações luminosas, ir-se-á utilizar a fita *LED* que incorpora os controladores *WS2812B* que, dada as suas características, melhor se integra no desenvolvimento de um sistema de iluminação inteligente.

#### 4.1.2 Processamento

Como foi descrito no subcapítulo 4.1.1, para aceder ao potencial da fita *LED* digital com o chip *WS2812B* é necessário enviar pulsos de informação em intervalos de tempo extremamente precisos. Desta forma, é fundamental utilizar um processador suficientemente rápido para enviar os dados para a fita.

Atualmente, existe no mercado das placas de desenvolvimento uma enorme oferta e é possível, muitas vezes, encontrar mais do que um adequado ao projeto a desenvolver. A decisão para o microcontrolador recaiu para o *Arduino Uno* (ver Figura 4.2), devido à sua enorme difusão no universo dos microcontroladores e à imensa documentação e aplicações existentes.



Figura 4.2 - Ilustração da placa de desenvolvimento utilizada, *Arduino Uno* [25].

Uma grande percentagem destas placas de desenvolvimento é utilizada com a filosofia *open-source* em mente. Isto é, existe toda uma comunidade que dá suporte ao desenvolvimento contínuo de bibliotecas de funções que permitem aos programadores aceder às potencialidades dos microcontroladores e utilizarem os mais variados sensores e módulos eletrónicos (*Bluetooth*, *Wi-Fi*, redes móveis, etc.) existentes.

O *Arduino Uno* é uma placa de desenvolvimento que integra o microprocessador Atmega328P [26], providenciando uma base para se poder aceder às funções do microprocessador de forma fácil e sem correr o risco de o danificar. A placa contém 14 entradas e saídas digitais e 6 analógicas e pode ser utilizada em conjunto com outras placas de expansão designadas por ‘*shields*’, que encaixam diretamente por cima do *Arduino Uno*, acrescentando funcionalidades não presentes na placa, como por exemplo, *Ethernet* e suporte para redes móveis, entre outros.

A equipa, que desenvolveu a placa, criou um ambiente de programação designado por *Arduino IDE*, que é utilizado para escrever o código e carregá-lo para o microprocessador. Assim o utilizador programa-o utilizando uma linguagem de alto-nível que, neste caso, é a linguagem C++, ao invés de programar diretamente na linguagem do microprocessador (*Assembly*) que é de baixo nível e, portanto, mais complicada e pouco *user-friendly*.

A escolha para esta placa deveu-se ao facto de apresentar uma curva de aprendizagem baixa e apresentar uma versatilidade enorme, permitindo a integração com todo o tipo de módulos eletrónicos e sensores, pois apresenta vários protocolos de comunicação, como o *I2C*, *SPI* e comunicação série, entre outros. O seu baixo custo, comparativamente a algumas placas de desenvolvimento existentes no mercado como o *Raspberry Pi 3*, também é um dos pontos-chave, uma vez que um dos objetivos deste projeto de iluminação inteligente é manter o custo da solução o mais reduzido possível.

#### 4.1.3 Módulo de comunicação

Sendo que neste projeto se pretende dar ao utilizador a capacidade de interagir à distância com os enfeites luminosos, o sistema deverá utilizar algum tipo de comunicação sem-fios. Foram estudados e analisados os dois *standards* mais difundidos no mercado tecnológico, nomeadamente, os *standards Wi-Fi* e o *Bluetooth 4.0*, mais conhecido por *Bluetooth Low Energy (BLE)*.

Chegou-se a considerar a hipótese de utilizar pontos de acesso com recurso a redes móveis 4G, embora tenha sido rapidamente rejeitada a sua utilização pelo seu custo e consumo de recursos elevados.

Sendo que o *Wi-Fi* e o *BLE* vêm incluídos nos smartphones mais recentes, faz todo o sentido considerar estas duas tecnologias para serem aplicadas no desenvolvimento da solução de iluminação proposta.

Os módulos escolhidos para cada uma das tecnologias foram, o **nRF24L01** para o *Wi-Fi* e o **HM-10** para o *BLE*. Na Figura 4.3, situada na página seguinte, encontra-se uma ilustração dos dois módulos.

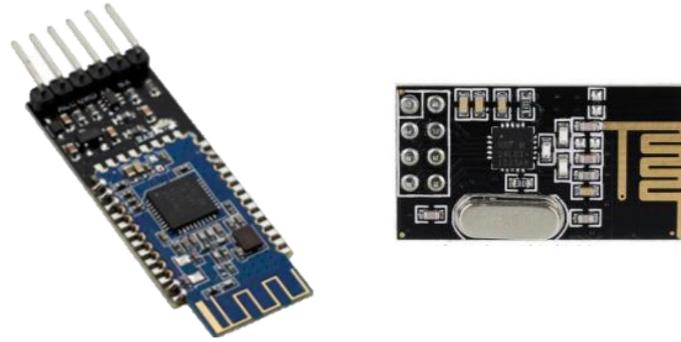


Figura 4.3 - Ilustração dos dois módulos estudados para o sistema. À esquerda encontra-se o módulo BLE, da Jinan Huanmao Co., e à direita, o módulo *Wi-Fi*, da Nordic Semiconductor, [27] e [28].

Foi realizado um estudo sobre qual a tecnologia seria mais adequada para aplicar no sistema de iluminação inteligente tendo em vista as características do mesmo. Primeiramente, foi estudado o módulo *Wi-Fi*, visto ser a tecnologia mais desenvolvida nos *smartphones* atuais, contudo, rapidamente se chegou à conclusão de que consumia mais recursos do *Arduino* que o módulo *Bluetooth* e que não era possível implementar a função de localização com recurso a *iBeacon*, usando a tecnologia *Wi-Fi*. Ir-se-á deixar a aplicação desta tecnologia para integrar em rede todos os dispositivos de iluminação que possam ser instalados.

Deste modo, para demonstrar a prova de conceito do sistema, ir-se-á utilizar a tecnologia *Bluetooth* 4.0, para deteção de pontos luminosos e ponte de comunicação entre utilizador e ponto luminoso. Este *standard* tem apenas o ponto negativo de apenas estar presente nos *smartphones* mais recentes. Sendo que os mais antigos podem não ter a função *BLE* instalada.

O módulo *Bluetooth BLE HM-10* foi escolhido por ser baseado em tecnologia *Bluetooth* 4.0, podendo ser programado diretamente como *iBeacon* ou utilizado como placa de comunicação clássica, características que são úteis para o sistema de iluminação. Também apresenta um custo ligeiramente inferior ao módulo *Wi-Fi* e consome menos energia na transmissão de dados.

A tecnologia *iBeacon*, patenteada pela *Apple Inc.*, atua primariamente como serviço de localização. O módulo envia periodicamente pulsos de informação, que contêm a sua identificação única e a força do sinal, permitindo, deste modo, calcular a distância relativa ao dispositivo. Através de uma notificação, o utilizador é avisado que entrou nas proximidades do *iBeacon*. Dependendo da forma como é configurado, o mesmo pode assumir um estado de consumo tão reduzido, que é possível alimentá-lo com uma pilha de botão e ter autonomia para mais de um ano a emitir informação, podendo ter vida útil mais longa utilizando pilhas com maior capacidade. Assim, a sua manutenção, na prática, é quase nula. [29]

No caso do sistema de iluminação, pretende-se associar um *iBeacon* a cada dispositivo luminoso, permitindo ao utilizador procurá-los utilizando a aplicação criada para o efeito, pois será notificado quando se aproximar do enfeite luminoso.

Irá ser ainda utilizado outro módulo *BLE* HM-10 para efetuar a comunicação entre o smartphone e o *Arduino*, uma vez que os comandos enviados pelo utilizador necessitam de ser processados.

#### 4.1.4 Circuito

Nos capítulos anteriores foram analisados os principais componentes que irão ser utilizados para construir o sistema de iluminação. Nesta secção irá ser apresentada a forma como irão ser interligados todos os componentes, de modo a criar um sistema funcional.

Utilizando um programa de design de circuitos, *Fritzing* [30], desenhou-se o esquema de ligações elétricas entre os componentes que integram o sistema, representado na Figura 4.4.

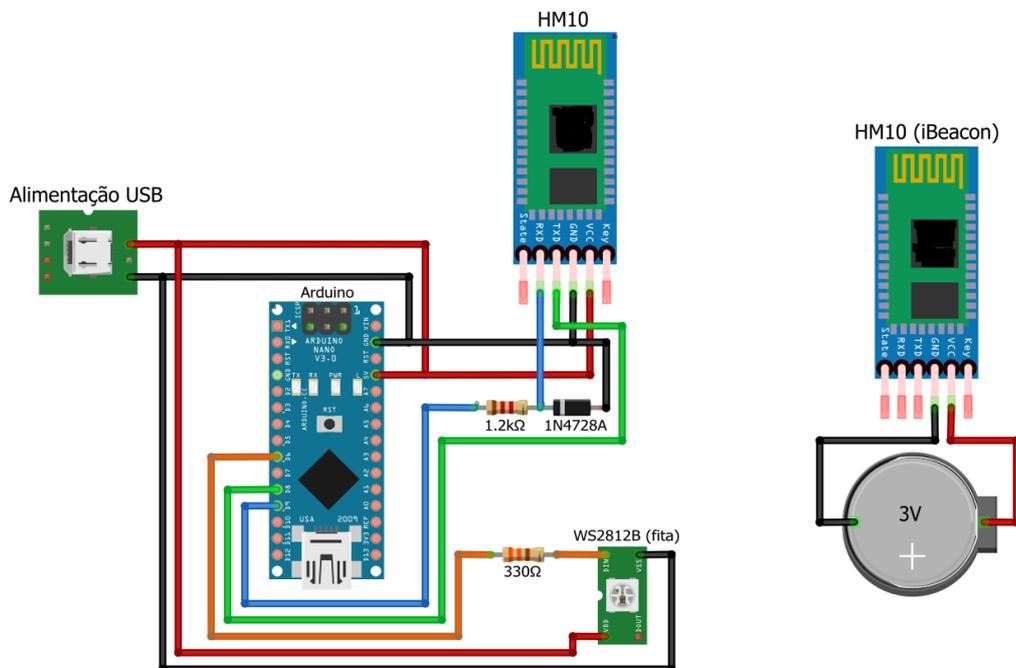


Figura 4.4 - Esquema elétrico das ligações entre componentes.

Pretende-se alimentar todo o circuito principal, composto pelo microcontrolador, módulo *Bluetooth* e dispositivo de iluminação, com uma tensão de 5 V, fornecida por um transformador de corrente AC-DC e um regulador de tensão.

A resistência de 1.2 kΩ em conjunto com o diodo *Zener* de 3.3 V (1N4728A) atua como um *level-shifter*, reduzindo a tensão da linha de transmissão de 5 V do *Arduino* para 3.3 V do pino *Rx* do módulo HM-10. É necessária esta redução de tensão, pois a tensão máxima admitida pelo pino *Rx* do módulo é 3.6 V, valor disponível na *datasheet* [31], caso contrário, danificar-se-ia a porta de comunicação do módulo.

Na Figura 4.4 consta apenas um diodo WS2812B, contudo apenas serve para demonstrar a ligação entre a fita e o microcontrolador, uma vez que será utilizada uma fita de tamanho personalizado no protótipo final. A resistência de  $330\ \Omega$  está em série com a porta digital do *Arduino* e o pino de entrada da fita e serve para mitigar algum pico de corrente que possa surgir durante o envio de informação entre o microcontrolador e a fita.

À direita da Figura 4.4 encontra-se o módulo *iBeacon*, totalmente configurável de acordo com as preferências do utilizador. Este é alimentado por uma pequena bateria e é independente do circuito do microcontrolador e acessórios. Pode ser colocado no local mais conveniente ao instalador. Não é necessário estar à vista e é utilizado apenas para referenciação geográfica do enfeite luminoso e desencadear a notificação na aplicação para o *smartphone*.

#### 4.1.5 Protótipo

O protótipo engloba o microcontrolador, o recetor *Bluetooth*, o dispositivo *iBeacon* e a componente de iluminação, que neste caso é uma fita *LED* baseada no diodo WS2812B. Na Figura 4.5 está ilustrado a montagem do protótipo.

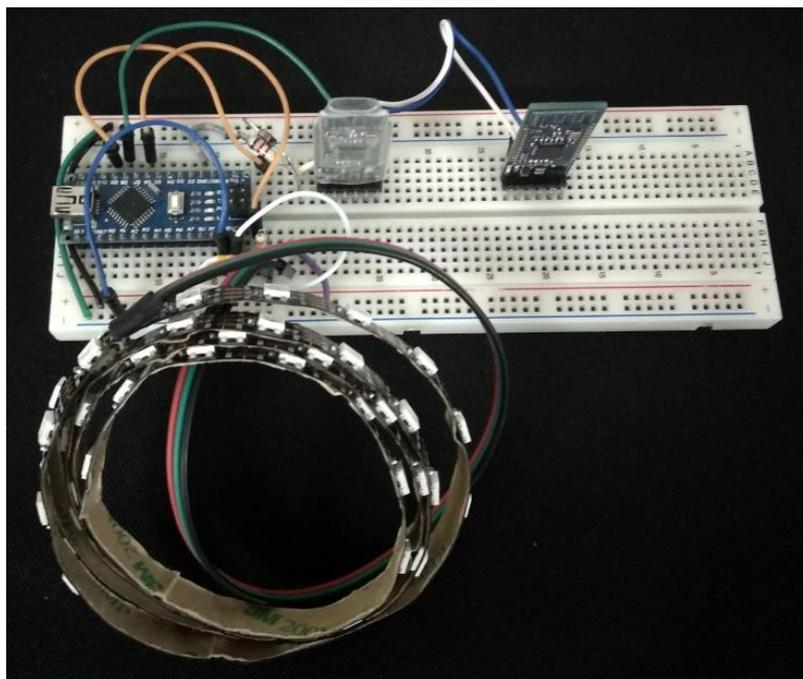


Figura 4.5 - Protótipo do *hardware* montado numa *breadboard*.

Na figura acima, encontra-se o microcontrolador, à esquerda, responsável pelo processamento e gestão de iluminação. Ao centro está o módulo *Bluetooth* e à esquerda o dispositivo *iBeacon*.

Na Figura 4.6 encontram-se ilustrados dois dos vários padrões luminosos que se podem enviar através da aplicação móvel. À esquerda está o padrão branco desvanecente, enquanto que à direita se encontra um outro padrão mais vistoso de arco-íris.



Figura 4.6 - Protótipo de uma árvore de Natal, um exemplo de um possível ponto luminoso interativo.

## 4.2 Aplicação

Para desenvolver a aplicação para o telemóvel recorreu-se à ferramenta de programação *Outsystems* [32]. Este programa difere dos demais por permitir a programação de aplicações de forma extremamente rápida e por permitir um elevado nível de integração com diversos serviços da *cloud*. Tem uma grande orientação para o aspeto visual apelativo e uma experiência de utilização fluida e intuitiva.

A aplicação apresenta quatro ecrãs principais. O menu principal, o de pesquisa de dispositivos, o de conexão ao ponto luminoso e o ecrã de interação. Nas próximas sub-seções ir-se-á analisar em detalhe e realizada uma breve explicação de cada um destes ecrãs.

Na Figura 4.7 está ilustrada a sequência de ações de funcionamento do sistema, isto é, como o utilizador deve proceder para utilizar a aplicação e interagir com os pontos luminosos.

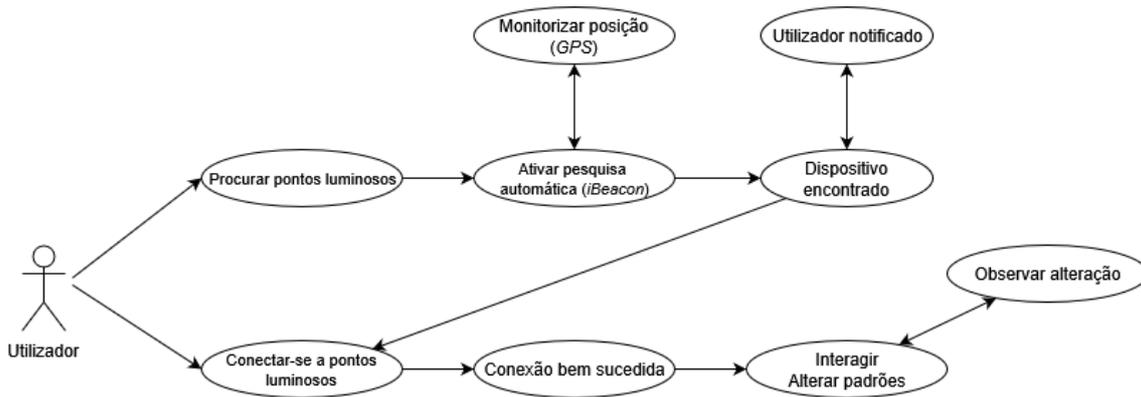


Figura 4.7 - Diagrama de casos de uso da aplicação.

#### 4.2.1 Menu principal

Quando se inicia a aplicação e após uma curta sequência de carregamento, é levado para o menu principal. Esta aplicação é livre, pelo que não é necessário fazer qualquer tipo de registo para a usar. O utilizador fica apenas condicionado a utilizá-la consoante o sistema de iluminação estar ou não em funcionamento. Neste ecrã existe uma barra inferior onde o utilizador pode navegar diretamente para os diferentes ecrãs da aplicação, nomeadamente o ecrã de pesquisa de pontos luminosos e o ecrã de ligação. Na barra lateral pode-se aceder às restantes funções não presentes na barra inferior. A Figura 4.8 mostra o ecrã de menu principal e a barra lateral.

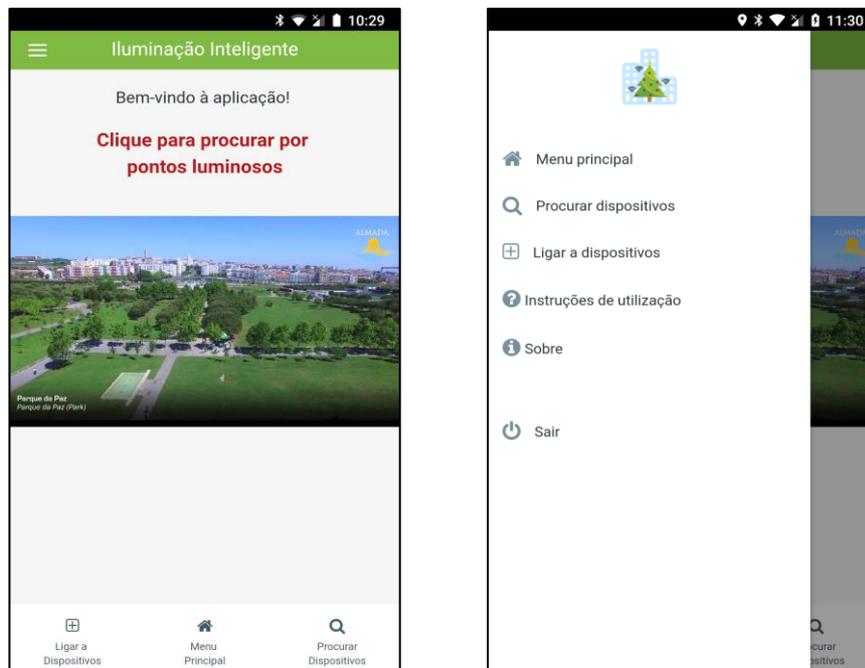


Figura 4.8 – Ilustração do menu principal e do menu lateral.

#### 4.2.2 Ecrã de procura aos pontos luminosos

Ao idealizar este sistema de iluminação inteligente, pensou-se em torná-lo num jogo de caça ao tesouro para o utilizador, isto é, pretende-se que este utilize a aplicação e os serviços de localização do smartphone para navegar pelo espaço onde se encontrem instalados os enfeites luminosos. A aplicação encarregar-se-á de avisar o utilizador caso esteja próximo ou mesmo o enfeite produzirá um efeito luminoso para despertar a atenção.

Neste ecrã encontra-se mapa onde se pode ver a localização dos pontos luminosos que já foram descobertos e assinalados por outros utilizadores da aplicação. Quando se entra neste ecrã, apenas se mostra o mapa centrado na localização do utilizador e a área circundante. Pode-se utilizar os botões de navegação disponíveis para ver onde estão os pontos luminosos mais próximos.

Consultando a Figura 4.9, observa-se o ecrã de procura de dispositivos, com especial atenção para o mapa e os marcadores assinalados a vermelho, que simbolizam os pontos luminosos já descobertos. O botão inferior comuta o estado de procura ativa dos *iBeacons* que marcam a posição dos dispositivos de iluminação. Ativando este botão, a aplicação utiliza os serviços *Bluetooth* do telemóvel para detetar os *iBeacons* e notificar o utilizador através de um menu *pop-up* e uma vibração.

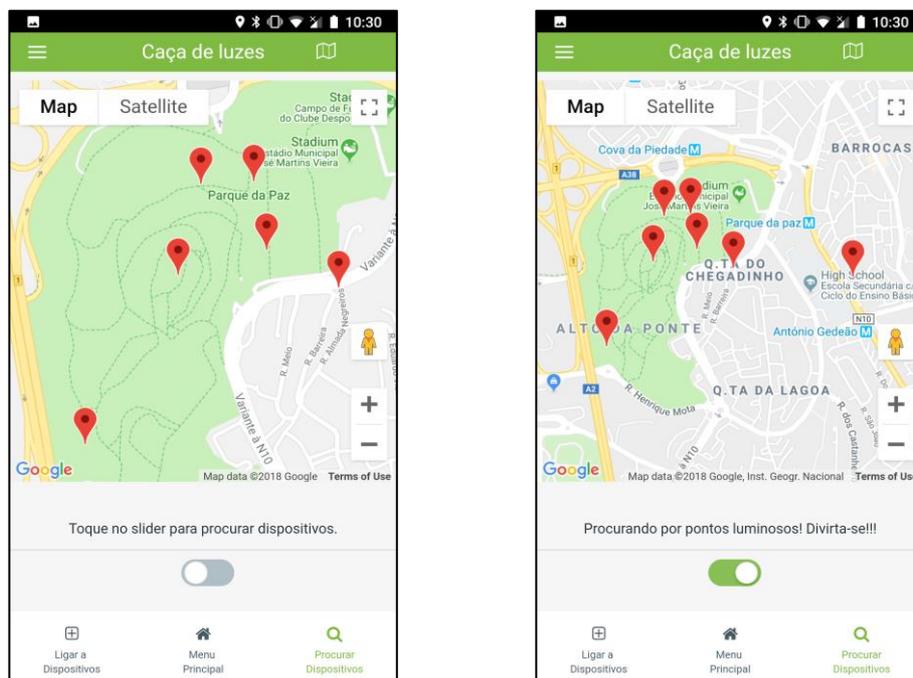


Figura 4.9 - Ecrã de procura de enfeites.

Através do botão no canto superior direito, é possível desligar a vista do mapa (ver Figura 4.10, à direita), evitando o consumo excessivo de recursos do telemóvel, mas mantendo a procura de *iBeacons* ativa.

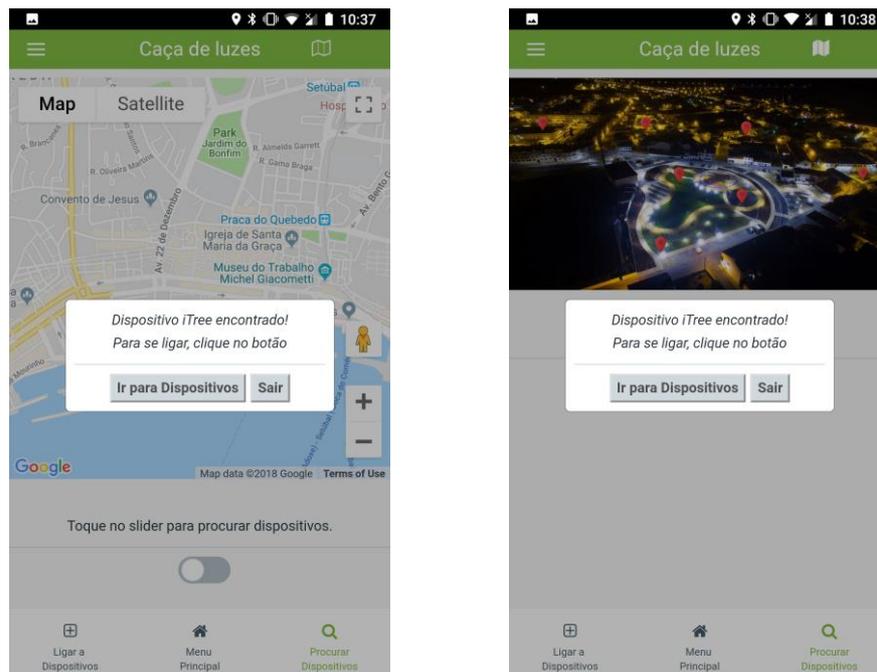


Figura 4.10 - Ilustração do pop-up de dispositivo encontrado.

Na figura em cima, observa-se a notificação mostrada ao utilizador e que foi encontrado com sucesso um ponto luminoso compatível com a aplicação. No *pop-up* existem duas opções disponíveis, o utilizador pode querer ligar-se ao dispositivo, bastando para isso clicar no botão esquerdo “Ir para Dispositivos” ou caso não queira, simplesmente, deverá clicar no botão “Sair”.

### 4.2.3 Ecrã de ligação de pontos luminosos

Este ecrã permite ao utilizador procurar todos os dispositivos que estejam nas proximidades e que se encontrem livres para aceitar uma ligação de um utilizador. Dado que cada dispositivo aceita apenas uma ligação em exclusivo, pode suceder que o utilizador, mesmo estando próximo do ponto luminoso, este não apareça na lista de pesquisa.

Existem duas formas distintas de aceder a este ecrã: uma manual, através da barra de navegação inferior ou pelo menu lateral, seleccionando a opção “Ligar a Dispositivos” e realizando a pesquisa manualmente; ou então pelo modo de procura automática que leva o utilizador automaticamente para este ecrã, caso seja detetado um dispositivo.

Na página seguinte encontra-se a Figura 4.11, que ilustra os casos de utilização deste ecrã de ligação aos dispositivos.

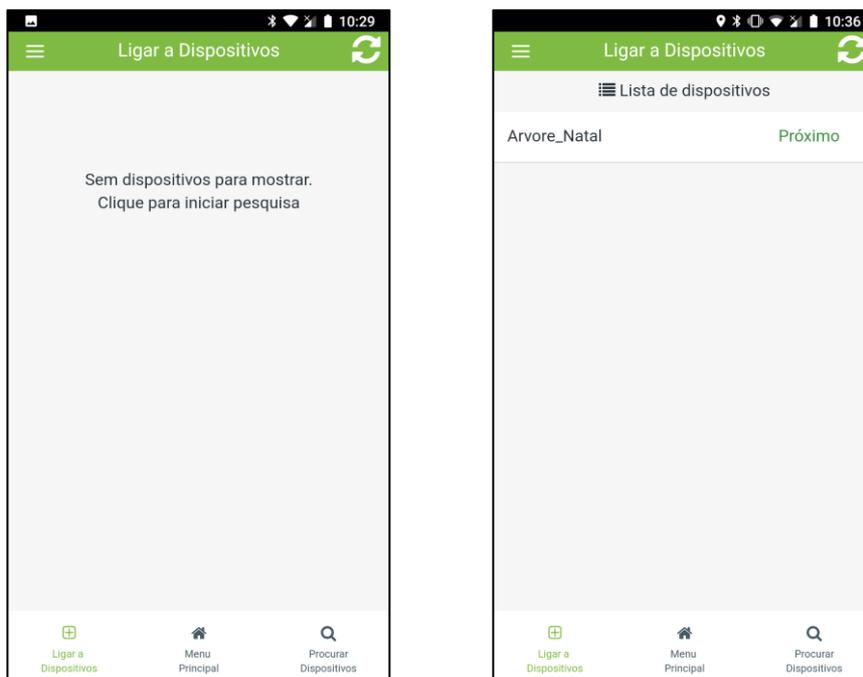


Figura 4.11 - Ilustrações do ecrã "Ligar a Dispositivos".

Consultando a figura acima, à esquerda, observa-se a mensagem inicial do ecrã. O utilizador pode clicar na mensagem ou no botão localizado no canto superior direito para iniciar uma nova pesquisa por dispositivos compatíveis com a aplicação.

Na ilustração da direita da Figura 4.11 encontra-se listado um dispositivo livre designado por 'Arvore\_Natal'. Uma vez que o alcance da pesquisa varia de telemóvel para telemóvel, podem aparecer mais dispositivos, sendo que se estiver muito afastado do utilizador, aparecerá a mensagem 'Distante' a vermelho, ao invés de 'Próximo'.

Para se ligar ao dispositivo, apenas é necessário clicar no elemento ao qual se pretende ligar e aguardar que a ligação seja bem-sucedida. Se tal acontecer, aparece uma mensagem a avisar o utilizador que está conectado ao dispositivo e é levado automaticamente para o ecrã de interação que será abordado na próxima sub-secção.

#### 4.2.4 Ecrã de interação com os pontos luminosos

Quando o utilizador se liga com sucesso ao ponto luminoso com o qual pretende interagir, a aplicação muda para o painel de controlo do enfeite automaticamente. O sistema está pensado para apresentar diversas categorias de pontos luminosos, cada um com as suas características e limitações. Deste modo, é necessário idealizar um ecrã de interação dedicado a cada tipo de enfeite.

Este ecrã está pensado para ser bastante intuitivo e de fácil utilização, apresentado diferentes botões para alternar entre os diversos padrões luminosos implementados no microcontrolador que gere a iluminação do enfeite.

Na Figura 4.12 exibe-se um possível ecrã de interação com um ponto luminoso. Neste caso específico é uma árvore de natal. Para percorrer os diversos padrões luminosos pré-definidos, o utilizador deve realizar um gesto de deslizar para um dos lados do ecrã.

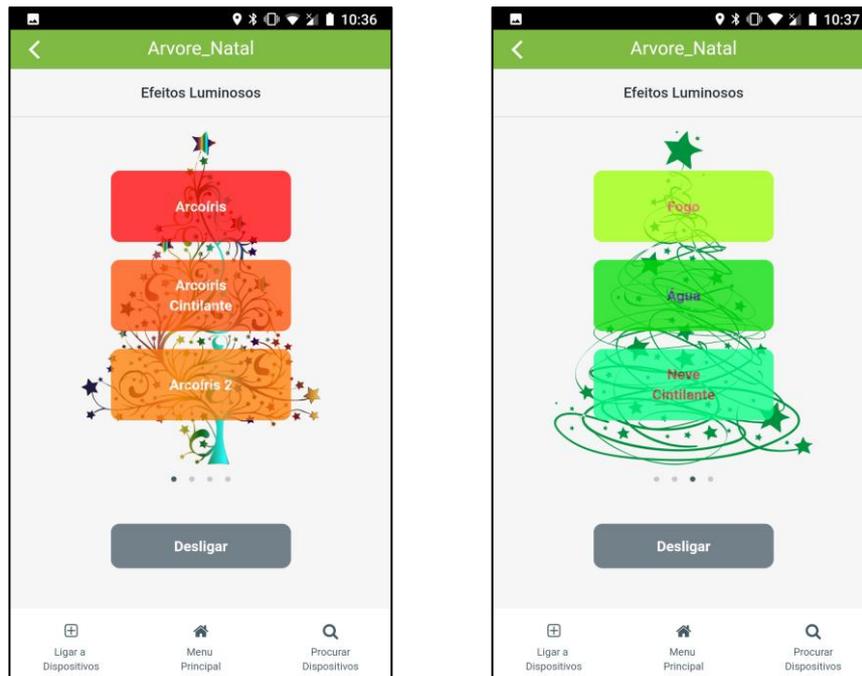


Figura 4.12 - Ilustração do ecrã de interação do enfeite luminoso.

Quando quiser alterar o padrão atual do enfeite, o utilizador deverá clicar no botão do padrão desejado. A aplicação enviará a informação necessária para o microcontrolador que se encarregará de atualizar o dispositivo de iluminação.

A comunicação entre o telemóvel e o microcontrolador é realizada em tempo real. Escolheu-se esta abordagem de dispositivo para dispositivo, ao invés de enviar os comandos para um servidor e posteriormente para o microcontrolador, por permitir uma comunicação mais rápida, permitindo que a experiência de utilização do utilizador seja a mais fluida, imersiva e pessoal possível, sem tempos mortos de espera pela mudança dos padrões luminosos.

Uma das desvantagens desta abordagem é o facto de não permitir a ligação simultânea de mais do que um telemóvel a cada enfeite, embora se tenha idealizado a conceção de um ponto luminoso principal que permita a ligação de vários utilizadores e que estes interajam uns com os outros, como por exemplo, decorar uma árvore de natal gigante no centro da cidade com recurso à aplicação.

Uma das formas de resolver esta falha do sistema inteligente é distribuir mais pontos luminosos pelo espaço. Uma vez que cada conjunto de microcontrolador, módulo de comunicação e dispositivo de iluminação foi pensado para apresentar um custo reduzido. Assim é uma solução mais viável para colmatar esta limitação de ligações em simultâneo.

### 4.3 *Outsystems* – Sequências da aplicação

Nesta secção ir-se-ão apresentar as principais funções utilizadas pela aplicação em ambiente *Outsystems*, realizando uma breve descrição de como se processa cada uma das funções e os blocos usados.

#### 4.3.1 Função *ScanDevices*

Esta função está associada ao ecrã de ligação aos pontos luminosos, descrito na subsecção 4.2.3. Quando o utilizador carrega no botão de pesquisa manual ou quando a aplicação o faz de forma automática, é ativada a função *ScanDevices*, representada na Figura 4.13.

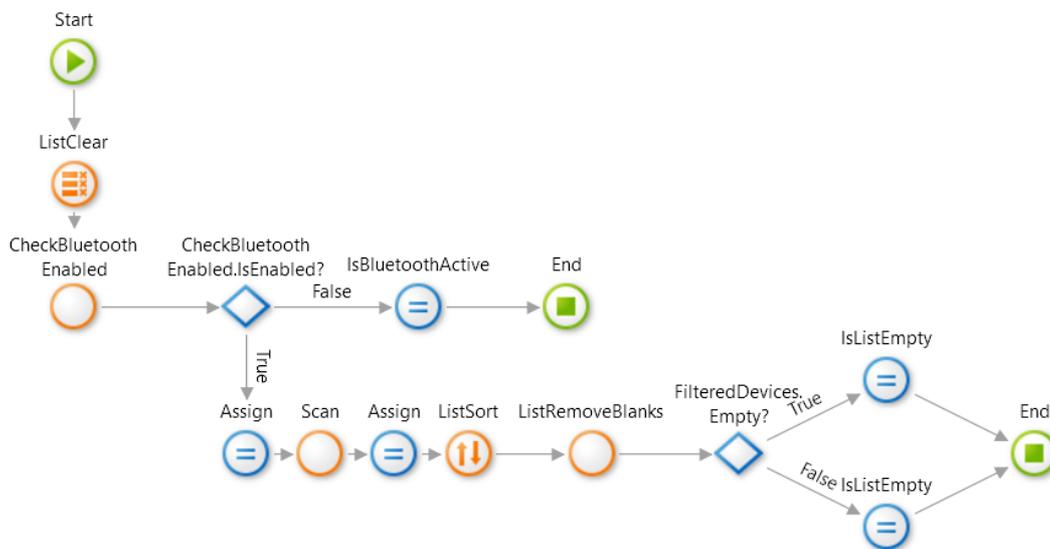


Figura 4.13 - Diagrama de sequência da função *ScanDevices* do ambiente *Outsystems*.

Ao ecrã de procura de pontos luminosos associa-se uma lista de dispositivos denominada de ‘*Devices*’, onde aparece a mensagem a avisar que não existem dispositivos (ver Figura 4.11) caso a lista esteja vazia após a pesquisa.

Antes de iniciar a procura, a função verifica se o *Bluetooth* se encontra ativado, terminando se tal não se verificar. Através do bloco ‘Scan’ e do ‘Assign’ à direita, a lista ‘Devices’ é preenchida e posteriormente é ordenada por ordem alfabética e retirados todos os dispositivos que não apresentem um nome válido (bloco ‘ListRemoveBlanks’).

#### 4.3.2 Função *Scan\_iBeacons*

Cada ponto luminoso encontra-se associado a um dispositivo *iBeacon*, que envia, periodicamente, sinais rádio que podem ser captados pelo telemóvel e, conseqüentemente, pela aplicação. Este dispositivo possui um identificador associado, o qual é reconhecido pela função que pesquisa pelos *iBeacons*. Na Figura 4.14 está ilustrada a função *Scan\_iBeacons*.

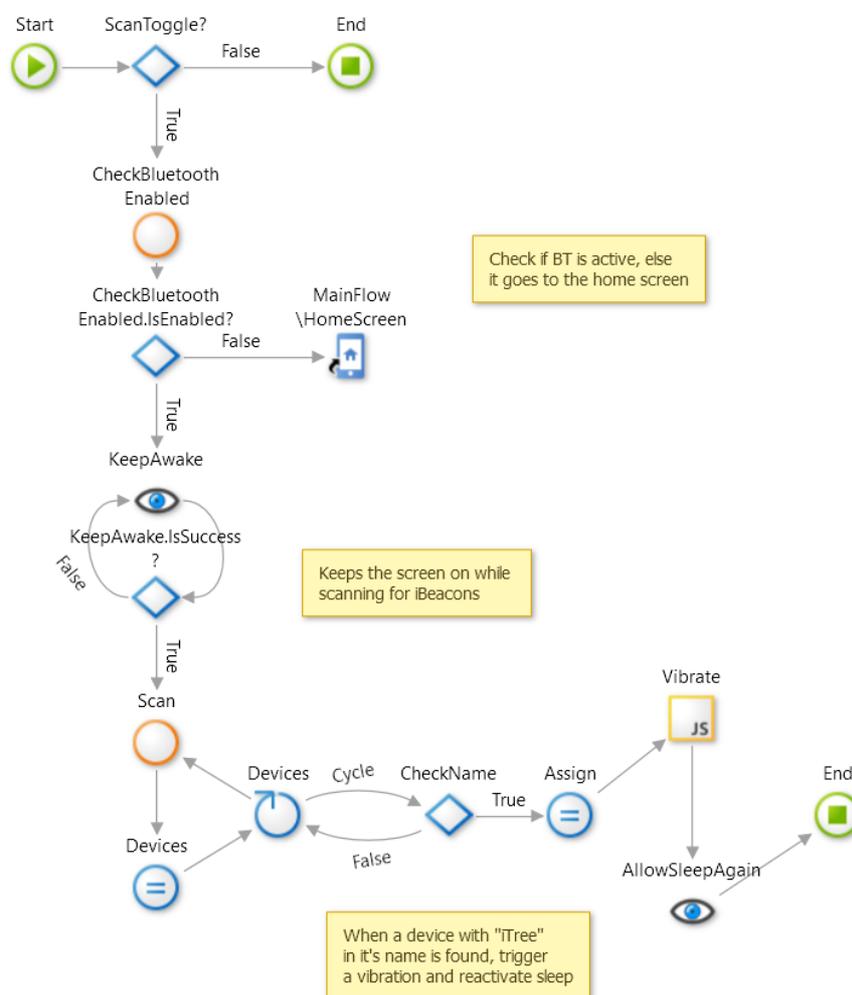


Figura 4.14 - Diagrama de sequência da função *Scan\_iBeacons* do ambiente *Outsystems*.

Esta função corre automaticamente assim que o utilizador pressiona o botão para ativar a pesquisa automática de pontos luminosos, na qual a aplicação está na realidade à procura dos identificadores associados ao *iBeacon* (ver Figura 4.9).

As primeiras duas condições verificam se o botão de *Scan* foi pressionado e se o *Bluetooth* está ativo, se não a função é terminada e no caso de o *Bluetooth* se encontrar desativado, retorna-se ao ecrã principal.

O bloco '*KeepAwake*' é utilizado para desativar a suspensão do ecrã, ou seja, permanece ligado, enquanto a função estiver ativada e não forem encontrados *iBeacons*.

O conjunto de blocos inferior é o essencial desta função. Com o bloco *Scan*, a função pesquisa por todos os dispositivos que estejam nas proximidades do telemóvel e utiliza-se o bloco *CheckName* para verificar se algum elemento dessa lista possui o identificador correto associado a um ponto luminoso compatível.

Assim que seja encontrado um *iBeacon*, a aplicação notifica o utilizador através de uma vibração dupla e aparece uma janela *pop-up* (ver Figura 4.10). O utilizador depois decide se pretende ir para o ecrã de ligação ao ponto luminoso, carregando no botão apropriado.

### 4.3.3 Funções *TrackPosition* e *GetPosition*

Uma vez que a aplicação apresenta um mapa onde estão incluídos os pontos luminosos assinalados por um marcador (ver Figura 4.9), é necessária a inclusão de duas funções que façam a geo-localização do utilizador recorrendo ao serviço de *GPS* do telemóvel. Na Figura 4.15 estão representadas as duas funções descritas.

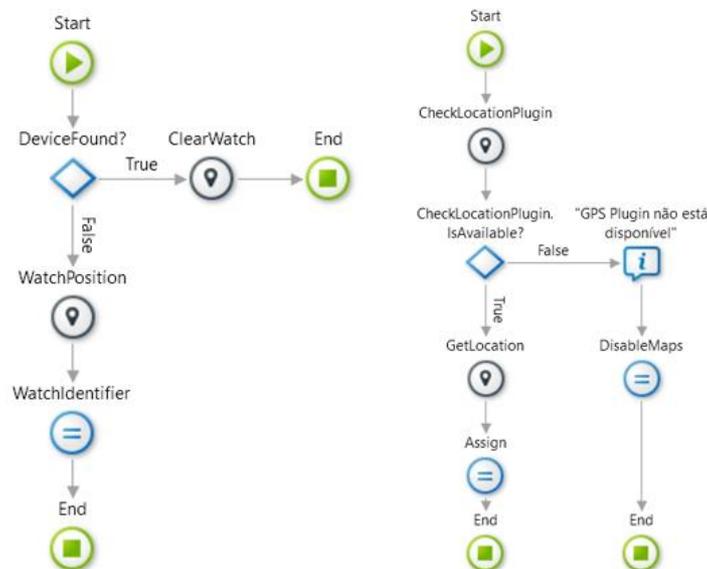


Figura 4.15 - Diagrama de sequência das funções de *TrackPosition* (esquerda) e *GetPosition* (direita) do ambiente *Outsystems*.

Estas duas funções têm como principal objetivo encontrar a posição do utilizador. A função *GetPosition* é utilizada para obter as coordenadas de latitude e longitude do telemóvel para centrar o mapa do ecrã na posição do utilizador ao carregar o ecrã. Deste modo, não necessita de andar à procura do local onde se encontra, pois é feito de forma automática.

A segunda função *TrackPosition*, é utilizada para seguir a posição do utilizador enquanto a pesquisa de *iBeacons* está ativada. Deste modo, o mapa centra-se sempre na posição do utilizador enquanto ele caminha.

#### 4.3.4 Funções dos padrões luminosos

No ecrã de interação com os pontos luminosos, existem diversas funções que são utilizadas para enviar a informação correta para ser decodificada pelo microcontrolador. Cada botão do ecrã de interação do dispositivo (ver Figura 4.12) tem uma função específica associada. Na Figura 4.16 encontra-se a representação da função de envio do padrão luminoso e lista de padrões disponíveis.

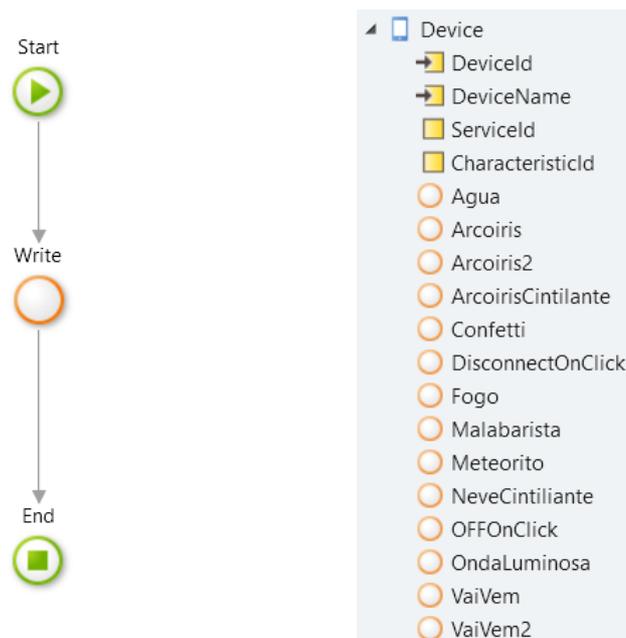


Figura 4.16 - Ilustração da função de envio de informação para o microcontrolador (esquerda) e lista de padrões luminosos (direita). Ambiente *Outsystems*.

A função responsável pelo envio da informação para o microcontrolador apenas apresenta um único bloco 'Write'. Apenas é necessário fornecer o identificador do dispositivo, obtido durante a fase de pesquisa e o conjunto de variáveis de serviço e característica, necessárias para que o módulo HM-10 receba a informação no canal de comunicação correto. Este bloco ainda apresenta mais um parâmetro que é um carácter (código ASCII) a ser enviado para que o microcontrolador saiba que padrão apresentar.



## **5 Testes e Resultados**

Neste quinto capítulo pretende-se dar a conhecer o método utilizado para validar os resultados obtidos com a implementação do sistema de iluminação inteligente apresentado no capítulo 4. Para além dos ensaios realizados ao protótipo, também se realizou um estudo sobre a utilização da aplicação para interagir com o ponto luminoso construído para testes.

## 5.1 Testes e utilização da aplicação

Nesta subsecção pretende-se apresentar um caso de uso para exemplificar a utilização da aplicação e mostrar o que se sucede no ponto luminoso. Numa primeira instância, ir-se-á apresentar um diagrama *UML* com a sequência de passos para interagir com um enfeite. Posteriormente, será apresentada a mesma sequência apresentada anteriormente, porém recorrendo a capturas de ecrã durante a utilização da aplicação móvel.

Na Figura 5.1 apresenta-se o diagrama de casos de uso da aplicação móvel utilizada pelo utilizador para procurar e interagir com os pontos luminosos e como se processa a sequência de ações para um funcionamento correto do sistema.

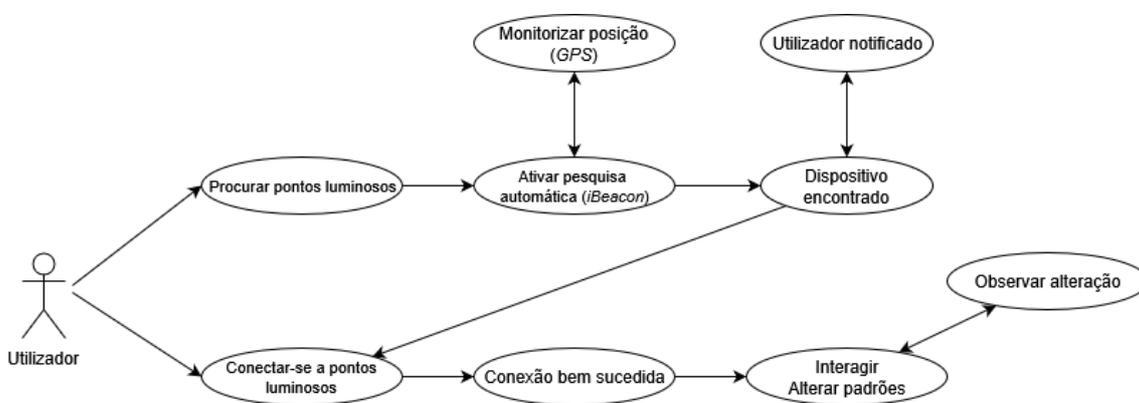


Figura 5.1 - Diagrama *UML* de casos de uso da aplicação.

Existem duas ações principais disponíveis para o utilizador, uma de procura automática e outra de conexão. Se o utilizador desejar encontrar pontos luminosos, deve escolher a primeira opção, devendo aguardar pela notificação apresentada pela aplicação, de que foi encontrado um dispositivo compatível. Caso já se encontre na vizinhança de um ponto luminoso, deverá utilizar o ecrã de conexão diretamente para estabelecer uma ligação ativa e poder interagir com o enfeite e observar as alterações aos padrões luminosos exibidos.

Apresenta-se, de seguida, um exemplo de como o utilizador pode usar a aplicação para procurar os diferentes pontos luminosos existentes, conectando-se posteriormente para poder interagir com os mesmos. Utilizaram-se capturas de ecrã retiradas da aplicação durante um teste ao sistema, assinalando-se os passos a serem seguidos pelo utilizador para uma utilização correta do sistema.

Na Figura 5.2, situada na página seguinte, encontram-se os três principais ecrãs para o utilizador, devendo para tal o utilizador carregar no botão correspondente na barra de navegação para aceder aos mesmos, assinalado a verde.

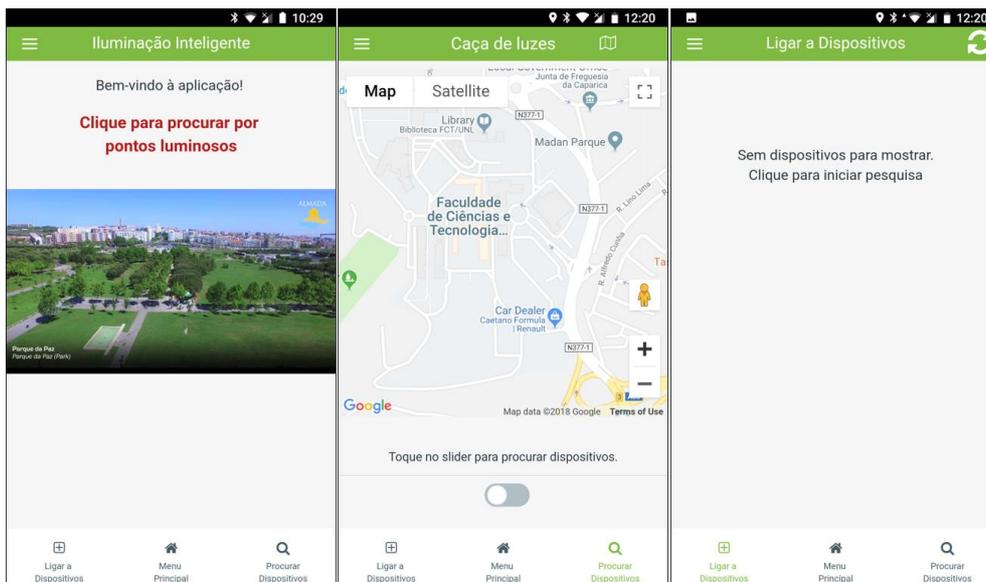


Figura 5.2 - Capturas de ecrã do menu principal à esquerda, ecrã de procura ao centro e de conexão aos pontos luminosos à direita.

Na Figura 5.3 ilustra-se o processo de encontrar os pontos luminosos. À esquerda encontra-se o ecrã inicial, devendo o utilizador clicar no botão assinalado a azul para iniciar a pesquisa. Após ativar a funcionalidade apenas necessita de caminhar e esperar pela notificação, ilustrada à direita. Se o utilizador desejar conectar-se ao ponto luminoso, deverá clicar no botão assinalado.

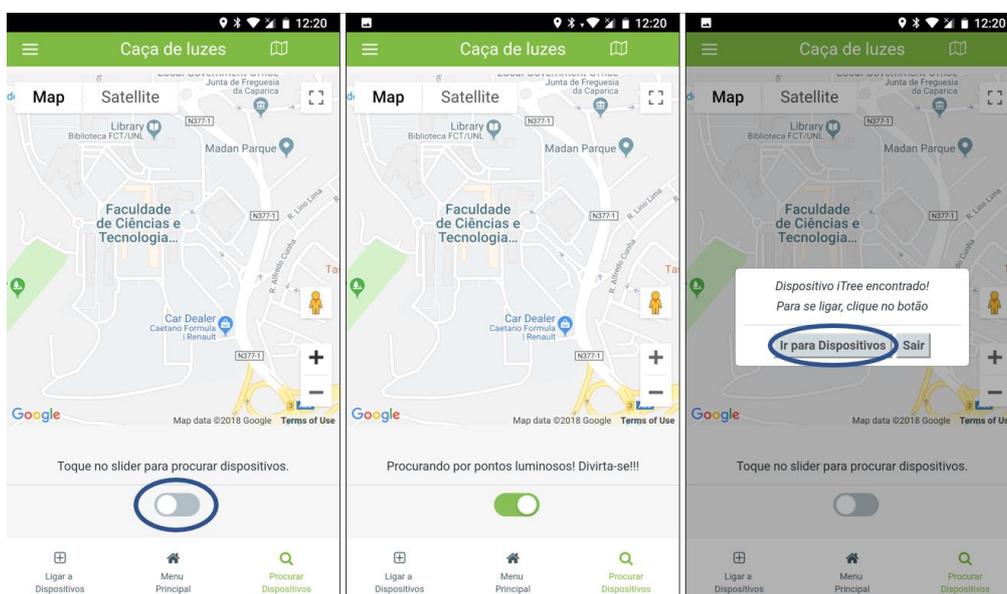


Figura 5.3 - Capturas de ecrã da utilização da procura automática de pontos luminosos, recorrendo aos *iBeacons*.

Na Figura 5.4, à esquerda, encontra-se o ecrã inicial de conexão. Para aceder a este ecrã utiliza-se barra inferior ou o botão presente na notificação do ecrã de procura (Figura 5.3 à direita). Para procurar por dispositivos que estejam livres para estabelecer uma ligação, deve-se carregar num dos locais assinalados a azul no ecrã à esquerda. Enquanto a aplicação procura, apresenta a mensagem “A procurar”. Caso encontre um ponto luminoso livre, o mesmo aparece listado como exemplificado no ecrã à direita da Figura 5.4.

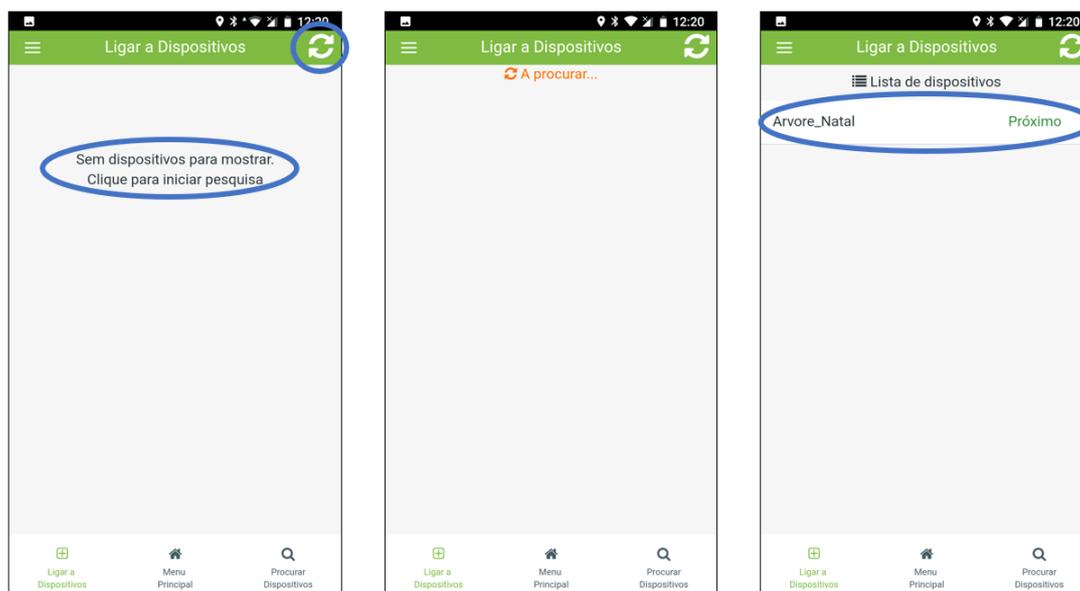


Figura 5.4 - Capturas do ecrã de conexão com os pontos luminosos.

Carregando no dispositivo listado e apenas se a ligação se realizar com sucesso, a aplicação redireciona o utilizador para o ecrã de interação que é exclusivo para o tipo de decoração em questão. Neste exemplo, o ponto luminoso testado é uma árvore de Natal.

O ecrã de interação apresenta 12 padrões luminosos que o utilizador pode escolher livremente havendo um botão situado em baixo para desligar por completo a iluminação. Esta opção de desligar o ponto luminoso apenas se encontra disponível para testes, dado que não estará ao dispor dos utilizadores durante o funcionamento normal do sistema.

Na Figura 5.5, situada na página seguinte, encontram-se ilustrados dois ecrãs do menu de interação com o ponto luminoso. Para navegar pelos padrões disponíveis, o utilizador deve realizar um movimento de deslizar com o dedo para qualquer um dos lados do ecrã. Após escolher o padrão que deseje observar, o utilizador deve pressionar o botão correspondente e aguardar que o sistema processe o comando e apresente o padrão luminoso.

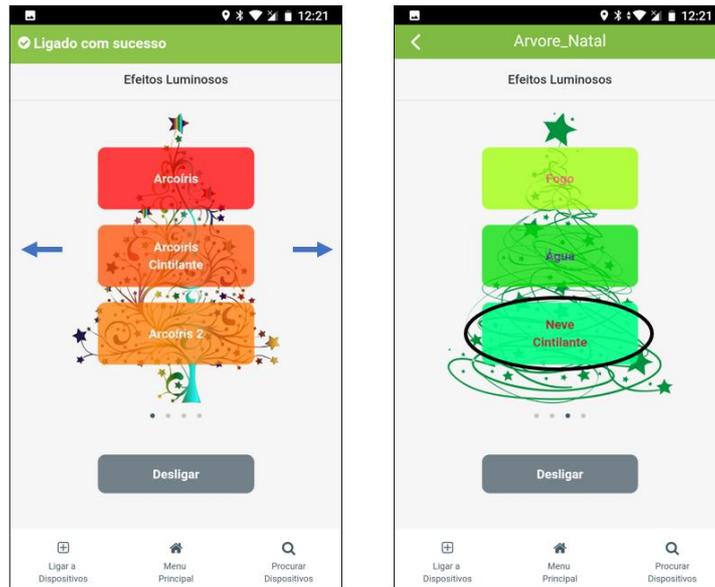


Figura 5.5 - Captura do ecrã de interação com o ponto luminoso.

Após carregar no padrão assinalado a preto à direita da Figura 5.5, a árvore de Natal muda para o padrão assinalado, podendo observar-se o mesmo na Figura 5.6.



Figura 5.6 - Ilustração do padrão luminoso "Neve cintilante" do dispositivo testado.

## 5.2 Validação da aplicação e funcionamento do protótipo

Para se testar a aplicação e para fazer os resultados obtidos serem coerentes com um ambiente o mais próximo possível daquele em que será utilizado o sistema, disponibilizou-se o protótipo do enfeite luminoso e o ficheiro de instalação da aplicação para os utilizadores poderem interagir com o protótipo.

Utilizou-se a aplicação *online Google Forms*<sup>1</sup> para se elaborar um breve questionário com cerca de cinco perguntas: a interface de utilizador; o funcionamento da pesquisa dos pontos luminosos; a conexão ao enfeite; a interação com o ponto luminoso (alteração dos padrões exibidos); e a experiência geral de utilização da aplicação. Também existia um campo de escrita para um pequeno comentário adicional acerca da aplicação. O principal objetivo deste formulário seria para facilitar o feedback dos utilizadores em relação à utilização da aplicação e testar a conectividade entre telemóvel e dispositivo.

Esta ferramenta da Google permite ao administrador obter informação em forma de gráficos estatísticos que ilustram de uma forma bastante simples e esclarecedora as respostas obtidas. No total, obteve-se resposta de uma amostra de 20 utilizadores, sendo o suficiente para, numa primeira fase de testes, verificar a recetividade deste *case-study* de um sistema inteligente de iluminação. Nas próximas figuras apresentam-se os resultados obtidos na resposta ao questionário.

Na Figura 5.7 estão apresentados os resultados da pergunta acerca da estruturação da interface do utilizador, a escala utilizada é do 1, para confusa, ao 4, para organizada. Observando as respostas, é possível concluir que a aplicação apresenta uma interface de utilizador bem construída e estruturada, com cerca de 95% de votos favoráveis.

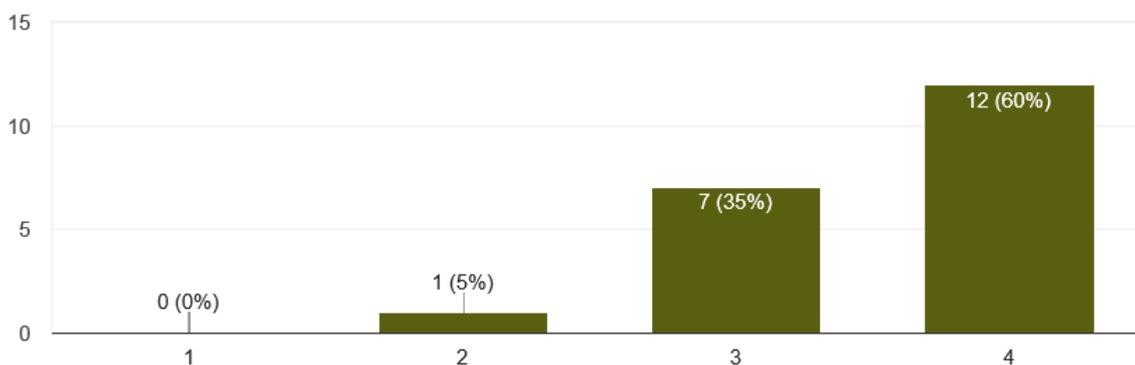


Figura 5.7 - Respostas à pergunta sobre a interface de utilizador da aplicação.

Na Figura 5.8 constam as respostas ao funcionamento da procura automática dos pontos luminosos, sendo o objetivo deste teste verificar se o telemóvel notifica o utilizador quando se

<sup>1</sup> <https://www.google.com/forms/about/>

aproxima do dispositivo. Verificando as respostas, chega-se à conclusão de que quatro utilizadores não obtiveram um funcionamento satisfatório da aplicação. Uma possível explicação para o sucedido possa ser alguma limitação do telemóvel à tecnologia *BLE* ou então uma falha do módulo *Bluetooth* HM-10. A escala utilizada segue o raciocínio da anterior, o 1 representa um mau funcionamento, e o 4, um funcionamento muito bom, sem falhas.

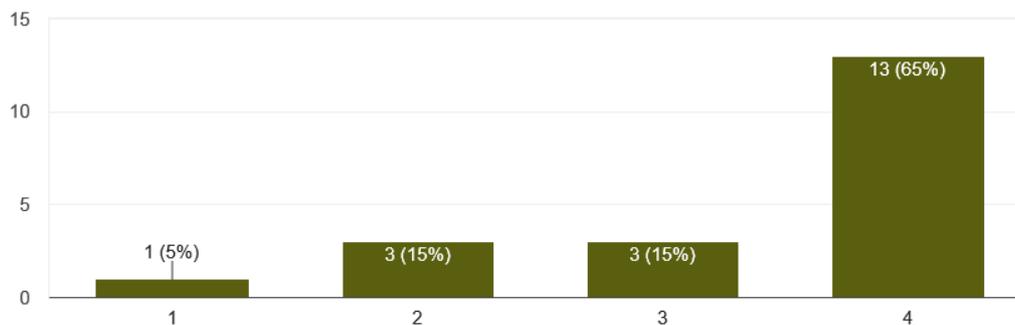


Figura 5.8 - Respostas à pergunta sobre o funcionamento da procura automática de pontos luminosos.

Na Figura 5.9 encontram-se as respostas ao funcionamento da conexão do utilizador ao ponto luminoso, passo necessário para poder interagir com o mesmo. Apenas 12 utilizadores conseguiram uma experiência sem falhas, enquanto que 12 utilizadores experienciaram problemas na ligação, sendo que alguns comentaram que a falha foi momentânea e que, após aguardarem uns instantes, a ligação foi bem-sucedida.

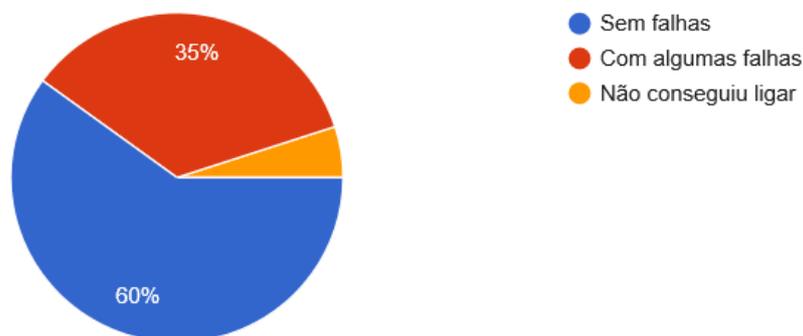


Figura 5.9 - Respostas à pergunta acerca da conexão ao ponto luminoso.

Na Figura 5.10 encontram-se as respostas dos utilizadores relativas à interação com o ponto luminoso, isto é, o envio de comandos diretamente para o dispositivo e o padrão luminoso alterar-se em conformidade. Observando o gráfico circular, conclui-se que o funcionamento do envio de comandos para o protótipo fica aquém das expectativas. Apenas 9 utilizadores relataram um funcionamento sem falhas. Das que responderam ‘Com algumas falhas’, algumas comentaram que ocasionalmente, um comando não era registado e que insistindo, era recebido com sucesso. Tal

deverá ser provocado por uma falta de otimização do código presente no microcontrolador, fazendo com que se perca alguns comandos.

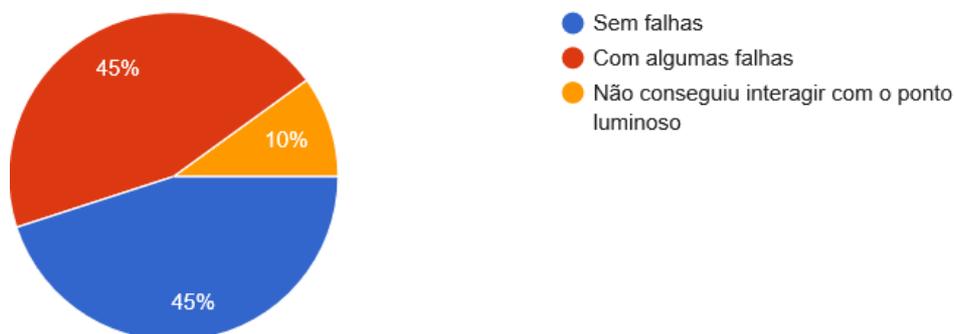


Figura 5.10 - Respostas à pergunta sobre a interação com o protótipo.

Por último, na Figura 5.11 encontram-se as respostas que os utilizadores deram relativamente à experiência de utilização geral da aplicação. A escala tem a pontuação máxima de 5 para uma experiência fluida e de 1 para uma experiência inconstante. Treze utilizadores (6+7) reportaram uma utilização fluida da aplicação.

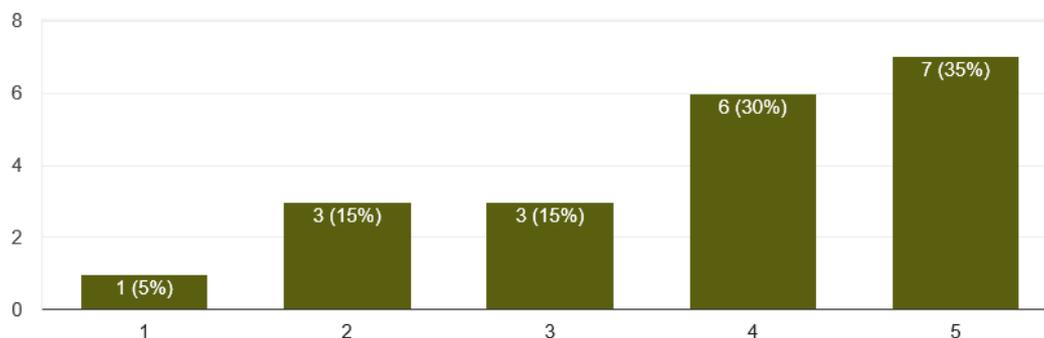


Figura 5.11 - Respostas à pergunta acerca da experiência de utilização no geral.

No geral, a aplicação apresenta um funcionamento satisfatório para uma versão de testes inicial. Apresentam-se alguns problemas derivados do funcionamento do *hardware* do protótipo originários de uma má otimização do código ou mesmo de um processador com uma velocidade de relógio inferior ao necessário para a quantidade de dados a serem processados que faz com que ocorra perda de informação entre a aplicação e o enfeite luminoso.

Outra variável que é complicada de se controlar é a enorme variação da tecnologia dos componentes presente nos telemóveis *Android*. Aliando este facto à vasta quantidade de diferentes versões no *software* da *Google* que existem no mercado, a probabilidade de ocorrerem algumas falhas decorrentes da utilização da aplicação sobe dramaticamente.

## **6 Conclusões e trabalho futuro**

Neste último capítulo apresentam-se as considerações finais referentes ao projeto que foi desenvolvido na presente dissertação. Também se constata algumas sugestões para o desenvolvimento de trabalhos futuros que poderão ter como ponto de partida a arquitetura sugerida por este sistema inteligente de iluminação.

## 6.1 Síntese e considerações finais

Atualmente, o tema das cidades inteligentes é um assunto cada vez mais discutido e, rapidamente, têm surgido novas ideias e conceitos para tornar as cidades em locais que servem os seus cidadãos e não o contrário. O aparecimento de soluções baseadas na *Internet of Things* permitiu a disseminação de tecnologia

Esta tendência estende-se a múltiplos departamentos de gestão das cidades como, por exemplo, o departamento de finanças, de turismo ou de energia. Estes dois últimos merecem um especial enfoque por parte dos autarcas, uma vez que são os que geram mais receita e permitem uma maior redução de custos operacionais, respetivamente. Tem sido feito um grande esforço por parte dos municípios em utilizar a tecnologia *LED* associada a sistemas de controlo inteligente e adaptativos para obter a máxima poupança energética.

Nos últimos anos, assistiu-se a uma clara aposta dos autarcas nas alturas de comemorações, na tentativa de atrair cada vez mais pessoas, sejam locais ou deslocadas. Esta promoção de eventos em épocas de festa tem provocado um aumento do número de decorações e enfeites luminosos. Aproveitando este movimento, surgiu a ideia de desenvolver um sistema inteligente de iluminação.

A solução proposta apresenta duas vertentes, uma parte de *software* e outra de *hardware*, para realizar a validação do trabalho. A primeira diz respeito à aplicação desenvolvida para o telemóvel, utilizando a ferramenta de programação, *Outsystems*. Seguidamente, construiu-se um pequeno protótipo de iluminação para demonstrar o funcionamento da aplicação.

A aplicação disponibilizada para o telemóvel apresenta uma interface simples e intuitiva, permitindo ao utilizador aceder a um conjunto de funções que lhe permite encontrar os pontos luminosos com extrema facilidade, pois existe uma função de procura automática que envia uma notificação quando se está nas proximidades de um dispositivo compatível. Todo o processo de ligação e interação com o enfeite também é bastante simples e objetivo. Cada enfeite conta com um conjunto de padrões exclusivos consoante as suas características.

Apesar dos problemas reportados pela amostra de utilizadores teste relativamente à comunicação entre a aplicação e o protótipo, o feedback proporcionado foi bastante positivo, no geral, com algumas críticas construtivas.

Este sistema foi pensado e desenhado para ser facilmente ampliável, uma vez que todos os pontos luminosos disponíveis estão registados e encontram-se numa base de dados acessível ao programador. Os utilizadores da aplicação têm ao seu dispor um mapa interativo onde se encontram assinalados os pontos disponíveis e é atualizado automaticamente cada vez que são acrescentados novos dispositivos à base de dados ou quando um utilizador os descobre e adiciona.

Foi possível manter o custo do hardware baixo, através da inclusão de componentes eletrônicos baseados em tecnologia provada e com um excelente suporte técnico e pela imensa documentação disponível. Dependendo das funcionalidades desejadas para o ponto luminoso, podem ser alterados os componentes constituintes de modo a reduzir ainda mais os custos do produto e o consumo energético.

A solução apresentada tem capacidade para vir a ser integrada nos sistemas de iluminação festiva das autarquias, tornando, assim, a experiência mais pessoal e interativa.

## 6.2 Trabalho futuro

Como proposta para trabalho futuro, sugere-se a utilização de um microprocessador com uma velocidade de relógio mais rápida e tornar o código mais otimizado, de modo a mitigar algumas das falhas que originam perda de informação no envio de informação da aplicação para o ponto luminoso.

Fica por implementar na presente dissertação a deteção de pessoas com base nos sinais rádio emitidos pelos telemóveis de modo a dotar o sistema de alguma capacidade de interação autónoma com os utilizadores em situação de aproximação; de se desativar, caso seja detetado um período de inatividade prolongado; a função de criação de padrões luminosos personalizados pelos utilizadores e posterior partilha para um repositório acessível a partir da aplicação.

Em alternativa à utilização do módulo *Bluetooth* HM-10 que apenas permite uma ligação ativa em cada instante, sugere-se o uso de outros módulos que apresentem a mesma ou outras tecnologias de comunicação que permitam a ligação de mais dispositivos de modo a que os utilizadores possam interagir não só com os pontos luminosos, mas também uns com os outros em simultâneo, numa espécie de jogo.

Sugere-se a finalização da arquitetura proposta, fazendo a inclusão dos serviços baseados em *cloud computing*. Incluem-se alguns serviços de tratamento de dados e análise estatística da utilização de cada enfeite para determinar a popularidade de cada um e a deteção de falhas com base em redes neuronais e algoritmos genéticos.

Relativamente ao consumo energético, é sugerido realizar um estudo sobre a possibilidade de utilizar um conjunto de painel solar fotovoltaico com um banco de baterias para efetuar a alimentação de um módulo de iluminação ou alimentar uma ilha de pontos luminosos. Deste modo, seria possível isolar completamente o sistema de iluminação da rede elétrica e reduzir os seus custos operacionais.



## Bibliografia

- [1] “Gartner Says 8.4 Billion Connected ‘Things’ Will Be in Use in 2017, Up 31 Percent From 2016.” [Online]. Available: <https://www.gartner.com/en/newsroom/press-releases/2017-02-07-gartner-says-8-billion-connected-things-will-be-in-use-in-2017-up-31-percent-from-2016>. [Accessed: 06-Sep-2018].
- [2] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey,” *Comput. Networks*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.
- [3] M. Dohler, I. Vilajosana, X. Vilajosana, and J. Llosa, “Smart Cities: An Action Plan,” *Proc. Barcelona Smart Cities Congr. 2011*, 2011.
- [4] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, “Internet of Things for Smart Cities,” *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.
- [5] J. M. Hernández-Muñoz *et al.*, “Smart Cities at the Forefront of the Future Internet,” Springer, Berlin, Heidelberg, 2011, pp. 447–462.
- [6] M. Abdur Razzaq, S. Habib Gill, E. Rahim Yar Khan Campus Rahim Yar Khan, P. Muhammad Ali Qureshi, S. Ullah, and R. Yar Khan, “Security Issues in the Internet of Things (IoT): A Comprehensive Study,” *IJACSA Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 6, 2017.
- [7] Spiceworks, “2016 IoT Trends.” [Online]. Available: <https://www.spiceworks.com/marketing/reports/iot-trends/>. [Accessed: 09-Feb-2018].
- [8] R. Kitchin and M. Dodge, “The (In)Security of Smart Cities: Vulnerabilities, Risks, Mitigation, and Prevention,” *J. Urban Technol.*, pp. 1–19, Dec. 2017.
- [9] K. Zhang, J. Ni, K. Yang, X. Liang, J. Ren, and X. S. Shen, “Security and Privacy in Smart City Applications: Challenges and Solutions,” *IEEE Commun. Mag.*, vol. 55, no. 1, pp. 122–129, Jan. 2017.
- [10] Google, “Google Trends.” [Online]. Available: [https://trends.google.com/trends/explore?date=2008-01-01 2018-01-01&q=gamification](https://trends.google.com/trends/explore?date=2008-01-01%2018-01-01&q=gamification).

[Accessed: 09-Feb-2018].

- [11] K. Seaborn and D. I. Fels, "Gamification in theory and action: A survey," *Int. J. Hum. Comput. Stud.*, vol. 74, pp. 14–31, Feb. 2015.
- [12] D. Basten, "Gamification," *IEEE Softw.*, vol. 34, no. 5, pp. 76–81, 2017.
- [13] S. Vicini, S. Bellini, A. Rosi, and A. Sanna, "An internet of things enabled interactive totem for children in a living lab setting," in *2012 18th International ICE Conference on Engineering, Technology and Innovation*, 2012, pp. 1–10.
- [14] D. Hart and E. Bossert, "Self-reported fears of hospitalized school-age children," *J. Pediatr. Nurs. Nurs. Care Child. Fam.*, vol. 9, no. 2, pp. 83–90, Apr. 1994.
- [15] E. Fleisch, "What is the Internet of Things? An Economic Perspective What is the Internet of Things - An Economic Perspective," 2010.
- [16] E. S. Poulsen, H. J. Andersen, O. B. Jensen, R. Gade, T. Thyrrerstrup, and T. B. Moeslund, "Controlling Urban Lighting by Human Motion Patterns Results from a Full Scale Experiment."
- [17] S. Seitinger, D. S. Perry, and W. J. Mitchell, "Urban Pixels: Painting the City with Light," in *Human Factors in Computing Systems*, 2009, no. Boston, MA, 4-9 April, pp. 839–848.
- [18] G. A. Akpakwu, B. J. Silva, G. P. Hancke, and A. M. Abu-Mahfouz, "A Survey on 5G Networks for the Internet of Things: Communication Technologies and Challenges," *IEEE Access*, pp. 1–1, 2017.
- [19] N. Ahmed, H. Rahman, and M. I. Hussain, "A comparison of 802.11ah and 802.15.4 for IoT," *ICT Express*, vol. 2, no. 3, pp. 100–102, Sep. 2016.
- [20] K. Shahzad and B. Oelmann, "A comparative study of in-sensor processing vs. raw data transmission using ZigBee, BLE and Wi-Fi for data intensive monitoring applications," in *2014 11th International Symposium on Wireless Communications Systems (ISWCS)*, 2014, pp. 519–524.
- [21] K. E. Jeon, J. She, P. Soonsawad, and P. C. Ng, "BLE Beacons for Internet of Things Applications: Survey, Challenges and Opportunities," *IEEE Internet Things J.*, pp. 1–1, 2018.
- [22] Bluetooth SIG, "Bluetooth 5 | Bluetooth Technology Website." [Online]. Available: <https://www.bluetooth.com/specifications/bluetooth-core-specification/bluetooth5>. [Accessed: 20-Feb-2018].
- [23] S. Bhattacharjee, D. J. Sanghvi, B. Gokalgandhi, and A. Singh, "Industrial Automation and Sensing using ZigBee," *Int. J. Comput. Appl.*, vol. 105, no. 10, pp. 975–8887, 2014.
- [24] World-Semi Inc, "<http://www.world-semi.com/products/details-178.html>."
- [25] D. R. P. Patnaikuni, "A Comparative Study of Arduino, Raspberry Pi and ESP8266 as IoT Development Board," *Int. J. Adv. Res. Comput. Sci.*, vol. 8, no. 5, pp. 2350–2352, Jun. 2017.
- [26] Microchip Technology Inc., "ATmega328P Documentation." [Online]. Available:

- <https://www.microchip.com/wwwproducts/en/ATmega328P>. [Accessed: 09-Aug-2018].
- [27] A. Rahim, Z. Ali, R. Bharti, S. Sabeel NSP, and A. Professor, “Design and Implementation of a Low Cost Wireless Sensor Network using Arduino and nRF24L01(+),” 2016.
- [28] J.-H. Huh and K. Seo, “An Indoor Location-Based Control System Using Bluetooth Beacons for IoT Systems.,” *Sensors (Basel)*, vol. 17, no. 12, Dec. 2017.
- [29] M. Kouhne and J. Sieck, “Location-Based Services with iBeacon Technology,” in *2014 2nd International Conference on Artificial Intelligence, Modelling and Simulation*, 2014, pp. 315–321.
- [30] “Fritzing.” [Online]. Available: <http://fritzing.org/home/>. [Accessed: 20-Aug-2018].
- [31] “HM-10 Datasheet.” [Online]. Available: [ftp://imall.iteadstudio.com/Modules/IM130614001\\_Serial\\_Port\\_BLE\\_Module\\_Master\\_Slave\\_HM-10/DS\\_IM130614001\\_Serial\\_Port\\_BLE\\_Module\\_Master\\_Slave\\_HM-10.pdf](ftp://imall.iteadstudio.com/Modules/IM130614001_Serial_Port_BLE_Module_Master_Slave_HM-10/DS_IM130614001_Serial_Port_BLE_Module_Master_Slave_HM-10.pdf). [Accessed: 20-Aug-2018].
- [32] “Outsystems Documentation.” [Online]. Available: <https://success.outsystems.com/Documentation>. [Accessed: 23-Aug-2018].

## Anexo 1 – Código Arduino

```
#include <Arduino.h>
#include <FastLED.h>
#include <AltSoftSerial.h>

#define NUM_LEDS 55
#define DATA_PIN 6
// Pins
// BT VCC to Arduino 5V out.
// BT GND to GND
// Arduino D8 (ASS RX) - BT TX no need voltage divider
// Arduino D9 (ASS TX) - BT RX through a voltage divider

AltSoftSerial BTserial;

char c=' ';

#define BRIGHTNESS 90
#define FRAMES_PER_SECOND 250

CRGB leds[NUM_LEDS];

uint8_t gHue = 0; // rotating "base color" used by many of the patterns
uint8_t glow = 5; //used for snow() function
bool up = true; //used for snow() function
bool ok = false; //used for snow() function
bool ok_pt = false; //used for portugal() function

CRGBPalette16 WaterColors_p = CRGBPalette16(CRGB::Black, CRGB::MediumSeaGreen,
CRGB::Aqua, CRGB::GhostWhite);
CRGBPalette16 FireColors_p = CRGBPalette16(CRGB::Black, CRGB::OrangeRed,
CRGB::Red, CRGB::SandyBrown);
uint8_t cooling = 65;
uint8_t sparking = 120;
////////////////////////////////////////////////////////////////

void fadeall() {
  for(int i = 0; i < NUM_LEDS; i++) {
    leds[i].nscale8(255);
  }
}

////////////////////////////////////////////////////////////////
// Rainbow
////////////////////////////////////////////////////////////////
void rainbow()
{
  // FastLED's built-in rainbow generator
```

```

fill_rainbow(leds, NUM_LEDS, gHue, 7);
}

////////////////////////////////////
// Rainbow with glitter
////////////////////////////////////
void addGlitter( fract8 chanceOfGlitter)
{
  if( random8() < chanceOfGlitter) {
    leds[ random16(NUM_LEDS) ] += CRGB::White;
  }
}

void rainbowWithGlitter()
{
  // built-in FastLED rainbow, plus some random sparkly glitter
  rainbow();
  addGlitter(80);
}

////////////////////////////////////
// Confetti
////////////////////////////////////
void confetti()
{
  // random colored speckles that blink in and fade smoothly
  fadeToBlackBy( leds, NUM_LEDS, 10);
  int pos = random16(NUM_LEDS);
  leds[pos] += CHSV( gHue + random8(64), 200, 255);
}

////////////////////////////////////
// Sinelon - Vai-Vem
////////////////////////////////////
void sinelon()
{
  // a colored dot sweeping back and forth, with fading trails
  fadeToBlackBy( leds, NUM_LEDS, 5);
  // int pos = beatsin16( 13, 0, NUM_LEDS-1 );
  int pos1 = beatsin16(10,0,NUM_LEDS);
  int pos2 = beatsin16(40,0,NUM_LEDS);
  leds[(pos1+pos2)/2] += CHSV( gHue, 255, 192);
}

void sinelon2()
{
  fadeToBlackBy( leds, NUM_LEDS, 10);
  int pos = beatsin16( 25, 0, NUM_LEDS-1 );
  leds[pos] += CHSV( gHue, 255, 192);
}

////////////////////////////////////

```

```

// BPM
////////////////////////////////////
void bpm()
{
  // colored stripes pulsing at a defined Beats-Per-Minute (BPM)
  uint8_t BeatsPerMinute = 80;
  CRGBPalette16 palette = PartyColors_p;
  uint8_t beat = beatsin8( BeatsPerMinute, 64, 255);
  for( int i = 0; i < NUM_LEDS; i++) { //9948
    leds[i] = ColorFromPalette(palette, gHue+(i*2), beat-gHue+(i*10));
  }
}

////////////////////////////////////
// Juggle
////////////////////////////////////
void juggle() {
  static uint8_t  numdots = 6; // Number of dots in use.
  static uint8_t  faderate = 27; // How long should the trails be. Very low value = longer trails.

  static uint8_t  thissat = 200; // Saturation of the colour.
  static uint8_t  thisbright = 255; // How bright should the LED/display be.

  static uint8_t  basebeat = 8; // Higher = faster movement.

  fadeToBlackBy( leds, NUM_LEDS, faderate);
  byte dothue = 0;
  for( int i = 0; i < numdots; i++) {
    leds[beatsin16( i+basebeat, 0, NUM_LEDS-1 )] |= CHSV(dothue, thissat, thisbright);
    dothue += 32;
  }
}

////////////////////////////////////
// Fire and Water
////////////////////////////////////
//      based      on      FastLED      example      Fire2012WithPalette:
https://github.com/FastLED/FastLED/blob/master/examples/Fire2012WithPal-
ette/Fire2012WithPalette.ino
void heatMap(CRGBPalette16 palette, bool up)
{
  fill_solid(leds, NUM_LEDS, CRGB::Black);

  // Add entropy to random number generator; we use a lot of it.
  random16_add_entropy(random(256));

  // Array of temperature readings at each simulation cell
  static byte heat[NUM_LEDS];

  byte colorindex;

  // Step 1. Cool down every cell a little
  for ( uint16_t i = 0; i < NUM_LEDS; i++) {

```

```

    heat[i] = qsub8( heat[i], random8(0, ((cooling * 10) / NUM_LEDS) + 2));
}

// Step 2. Heat from each cell drifts 'up' and diffuses a little
for ( uint16_t k = NUM_LEDS - 1; k >= 2; k-- ) {
    heat[k] = (heat[k - 1] + heat[k - 2] + heat[k - 2]) / 3;
}

// Step 3. Randomly ignite new 'sparks' of heat near the bottom
if ( random8() < sparking ) {
    int y = random8(7);
    heat[y] = qadd8( heat[y], random8(160, 255) );
}

// Step 4. Map from heat cells to LED colors
for ( uint16_t j = 0; j < NUM_LEDS; j++ ) {
    // Scale the heat value from 0-255 down to 0-240
    // for best results with color palettes.
    colorindex = scale8(heat[j], 190);

    CRGB color = ColorFromPalette(palette, colorindex);

    if (up) {
        leds[j] = color;
    }
    else {
        leds[(NUM_LEDS - 1) - j] = color;
    }
}
}

void fire()
{
    heatMap(FireColors_p, true);
}

void water()
{
    heatMap(WaterColors_p, false);
}

////////////////////////////////////
//  Alternative Rainbow
////////////////////////////////////

void rainbow2()
{
    static uint16_t sPseudotime = 0;
    static uint16_t sLastMillis = 0;
    static uint16_t sHue16 = 0;

    uint8_t sat8 = beatsin88( 87, 220, 250);

```

```

uint8_t brightdepth = beatsin88( 341, 96, 224);
uint16_t brightnesstheta16 = beatsin88( 203, (25 * 256), (40 * 256));
uint8_t msmultiplier = beatsin88(147, 23, 60);

uint16_t hue16 = sHue16;//gHue * 256;
uint16_t hueinc16 = beatsin88(113, 1, 3000);

uint16_t ms = millis();
uint16_t deltams = ms - sLastMillis ;
sLastMillis = ms;
sPseudotime += deltams * msmultiplier;
sHue16 += deltams * beatsin88( 400, 5, 9);
uint16_t brightnesstheta16 = sPseudotime;

for ( uint16_t i = 0 ; i < NUM_LEDS; i++) {
  hue16 += hueinc16;
  uint8_t hue8 = hue16 / 256;

  brightnesstheta16 += brightnessthetainc16;
  uint16_t b16 = sin16( brightnesstheta16 ) + 32768;

  uint16_t bri16 = (uint32_t)((uint32_t)b16 * (uint32_t)b16) / 65536;
  uint8_t bri8 = (uint32_t)(((uint32_t)bri16) * brightdepth) / 65536;
  bri8 += (255 - brightdepth);

  CRGB newcolor = CHSV( hue8, sat8, bri8);

  uint16_t pixelnumber = i;
  pixelnumber = (NUM_LEDS - 1) - pixelnumber;

  nblend( leds[pixelnumber], newcolor, 64);
}
}

////////////////////////////////////
//  snow with glitter
////////////////////////////////////

void snow(bool up)
{
  if(!up and ok){ glow--; }
  if(up and ok){ glow++; }
  ok=false;
  fill_solid(leds, NUM_LEDS, CRGB(glow,glow,glow));
  addGlitter(50);
}

////////////////////////////////////
//  portugal flag
////////////////////////////////////

void portugal()
{

```

```

if(!up and ok_pt){ glow--; }
if(up and ok_pt){ glow++; }
ok=false;
int green = NUM_LEDS*0.3;
int yellow = NUM_LEDS*0.1;
for (int i=0; i<green; i++){
  leds[i].setRGB(0,glow,0);          //GREEN
}
for (int i=green; i<green+yellow; i++){
  leds[i].setRGB(glow,glow,0);      //YELLOW
}
for (int i=green+yellow; i<NUM_LEDS; i++){
  leds[i].setRGB(glow,0,0);         //RED
}
}

////////////////////////////////////
void setup() {
  // Serial.begin(9600);
  BTserial.begin(9600);
  FastLED.addLeds<NEOPIXEL, DATA_PIN>(leds, NUM_LEDS);    // for WS2812 (Ne-
  opixel)
  LEDES.setBrightness(BRIGHTNESS);
  FastLED.clear();
  FastLED.show();
}

void loop() {

  FastLED.show();

  // insert a delay to keep the framerate modest
  FastLED.delay(1000/FRAMES_PER_SECOND);
  EVERY_N_MILLISECONDS(20) {
    gHue++;          // slowly cycle the "base color" through the rainbow
  }
  EVERY_N_MILLISECONDS(10) {
    ok_pt=true;     // slowly cycle the "base color" through the rainbow
  }
  EVERY_N_MILLISECONDS(75) {
    ok=true;        // slowly cycle the "base color" through the rainbow
  }

  BTserial.flush();
  if(BTserial.available()){
    char aux = BTserial.read();
    if((aux>47 and aux<58) || (aux>96 and aux<103) || aux==65){
      c=aux;
      Serial.println(c);
    }
  }
}

```

```

switch (c) {
case 48:    //'0'
  rainbow();
  break;
case 49:    //'1'
  rainbowWithGlitter();
  break;
case 50:    //'2'
  rainbow2();
  break;
case 51:    //'3'
  juggle();
  break;
case 52:    //'4'
  bpm();
  break;
case 53:    //'5'
  confetti();
  break;
case 54:    //'6'
  fire();
  break;
case 55:    //'7'
  water();
  break;
case 56:    //'8'
  if(glow==75){ up=false;}
  if(glow==5){ up=true;}
  snow(up);
  break;
case 57:    //'9'
  sinelon2();
  break;
case 97:    //'a'
  FastLED.clear();
  break;
case 98:    //'b'
  if(glow==250){ up=false;}
  if(glow==50){ up=true;}
  portugal();
  break;
case 99:    //'c'
  sinelon();
  break;
default:
  break;
// Serial.println(c);
// delay(100);
}
}

```