

Lightom: Uma solução Open Source para o sistema de iluminação pública

Abstract. *This article describes a free solution for managing and monitoring the public lighting system using hardware and software, this solution is Lightom project (Light On Management). In this project, we used the microcontroller Arduino, the microcomputer Raspberry Pi, Debian GNU/Linux operating system, programming languages Python, C/C++, Lua, and JavaScript, besides the of microframework Bottle, the database Redis and other technologies open source.*

Resumo. *Este artigo descreve uma solução livre para o gerenciamento e monitoramento do sistema de iluminação pública usando hardware e software livre, essa solução é o projeto Lightom (Light On Management). Nesse projeto, foi usado o microcontrolador Arduino, o microcomputador Raspberry Pi, o sistema operacional Debian GNU/Linux, as linguagens de programação Python, C/C++, Lua e JavaScript, além do microframework Bottle, o banco de dados Redis e outras tecnologias open source.*

1. Introdução

A cada dia surgem mais tecnologias, algumas inovadoras, outras simplesmente uma evolução das já existentes. Conceitos e técnicas são criadas a todo momento, porém alguns conceitos como o *free software*, *open source* e *open source hardware* vêm revolucionando o mundo de forma geral, não somente o da tecnologia especificamente.

Na era do conhecimento, compartilhar é fundamental. E este artigo, não poderia deixar de compartilhar, além de códigos, ideias e conceitos sobre o software livre e código aberto. No decorrer do desenvolvimento do projeto citado neste documento foram usados o microcontrolador *Arduino* e o *Single-Board Computer (SBC)*¹ *Raspberry Pi*, além das linguagens de programação *Python*, *C/C++*, *Lua* e *JavaScript*, além de outros softwares *open source* como o sistema operacional *Debian GNU/Linux*, o *microframework Bottle* e o banco de dados não relacional *Redis* para desenvolver uma solução para o gerenciamento e monitoramento da iluminação pública.

O projeto denominado *Lightom (Light On Management)*, surgiu da problemática do gerenciamento e monitoramento da iluminação pública, pois “a iluminação pública é responsabilidade da Administração Pública Municipal e é essencial para o melhor desenvolvimento social e econômico das cidades” (ELETROBRAS, 2004, p. 34). As atuais prefeituras municipais não possuem uma ferramenta eficaz para o gerenciamento e monitoramento dos pontos de iluminação pública em tempo real segundo o Programa Nacional de Conservação de Energia Elétrica - PROCEL (ELETROBRAS, 2004, p. 6), o que existe atualmente são projetos de conscientização para tentar reduzir o consumo de energia elétrica e possíveis melhorias na sua infraestrutura.

A falta de um gerenciamento e monitoramento específico dos pontos de

¹ *Single-Board Computer (SBC)* é um computador montado em uma única placa, com processador, memória, entrada/saída e outros componentes necessários ao seu funcionamento (WIKIPEDIA, 2013, p. 1)

iluminação pública, podem causar gastos desnecessários aos cofres públicos com o desperdício de energia elétrica, pois “é importante resaltar que o consumo de energia da Iluminação Pública representa cerca de 3,3% na matriz energética brasileira, e é da ordem de 10,2 Twh/ano (dados do PROCEL do ano de 2002)” (ELETROBRAS, 2004, p. 34).

O gerenciamento e monitoramento da iluminação pública é de suma importância para a sociedade, pois “quando bem atendidos, os cidadãos podem desfrutar de melhores oportunidades em atividades de turismo, sociais, educacionais, culturais e esportivas, atualmente ela é fundamental para a segurança do tráfego de veículos e para prevenção contra a criminalidade” (ELETROBRAS, 2004, p. 34).

Nesse artigo, há uma seção fundamentação teórica que descreve sobre o projeto e suas tecnologias; uma seção com uma breve descrição do projeto Lightom, sua infraestrutura e o porque das tecnologias utilizadas; concluindo assim o artigo com a seção considerações finais.

2. Fundamentação Teórica

Baseado na problemática de não haver nenhum tipo de gerenciamento e monitoramento eficiente dos pontos de iluminação pública, foram levantadas diversas tecnologias para o desenvolvimento do projeto Lightom, sendo optado pelo *open source* ou código aberto e *open source hardware* ou hardware de código aberto. O projeto Lightom é licenciado pela GPL² e pode ser classificado como FOSS (*Free and Open Source Software*), pois o mesmo segue as diretrizes do *free software* ou software livre e do *open source*.

O termo *free software* surgiu em 1984, quando Richard Matthew Stallman demitiu-se do MIT para trabalhar no projeto de criação de um novo sistema operacional livre. “Software livre é aquele software que respeita a liberdade e senso de comunidade dos usuários. Em grosso modo, os usuários possuem a liberdade de executar, copiar, distribuir, estudar, mudar e melhorar o software” (STALLMAN, 2013, p. 1)

Nessa mesma perspectiva, o *open source*, garante a liberdade de expressão e de compartilhamento do código fonte. Todavia, no *open source*, o desenvolvedor do código determina as suas condições de uso, “em vez de sistemas fechados, projetos de código aberto permitem uma liberdade individual para acessar os arquivos de origem de um projeto, fazer melhorias, e redistribuir essas melhorias para uma comunidade maior” (EVANS, 2011, p. 5).

Em relação ao termo *open source hardware*³, o que se busca é uma “[...] colaboração onde os objetos físicos são os resultados. Ele envolve um modelo distribuído de desenvolvimento de hardware com os contribuintes em geral, residentes em diferentes partes do mundo.” (EVANS, 2011, p. 5).

Assim, *free software* e *open source* determinam a liberdade de compartilhamento para o software, como por exemplo as linguagens de programação. Enquanto *open source hardware* determina a liberdade e o compartilhamento relacionada a dispositivos físicos.

Complementando, de acordo com Stallman (2012), para que um programa seja

2 Licença Geral Pública GNU.

3 *Open source hardware* é equivalente ao *open source* só que para dispositivos eletrônicos, englobando diagramas e *layouts* de placa de circuito.

considerado um software livre, os usuários devem ter as quatro liberdades fundamentais: “A liberdade de executar o programa, [...]; a liberdade de estudar como o programa funciona, [...]; a liberdade de redistribuir cópias [...]; A liberdade de distribuir cópias de suas versões modificadas a outros” (STALLMAN, 2012, p. 1)

Nessa mesma perspectiva, o microcomputador *Raspberry Pi* foi desenvolvido pela Fundação *Raspberry Pi* em 2009 por Eben Upton, Rob Mullins, Jack Lang e Alan Mycroft nos laboratórios de informática da Universidade de Cambridge. Seu propósito é ser um microcomputador de baixo custo com o objetivo principal de ajudar no ensino da computação nas escolas, porém “algumas palavras surgem repetidamente quando as pessoas falam sobre o *Raspberry Pi*: pequeno, barato, hackeável e educacional” (RICHARDSON; WALLACE, 2013, p. 17).

Além dessas características, conforme Richardson e Wallace (2013), a fundação *Raspberry Pi* recomenda o *Raspbian GNU/Linux*, embora existam outras distribuições GNU/Linux disponíveis. O *Raspbian* é um projeto não oficial do *Debian GNU/Linux* na versão *Wheezy* para arquitetura ARM, com configurações de compilação ajustada para produzir códigos otimizados. Schimdt (2012), disponibilizou uma imagem do *Raspberry Pi* no modelo B (Figura 1).

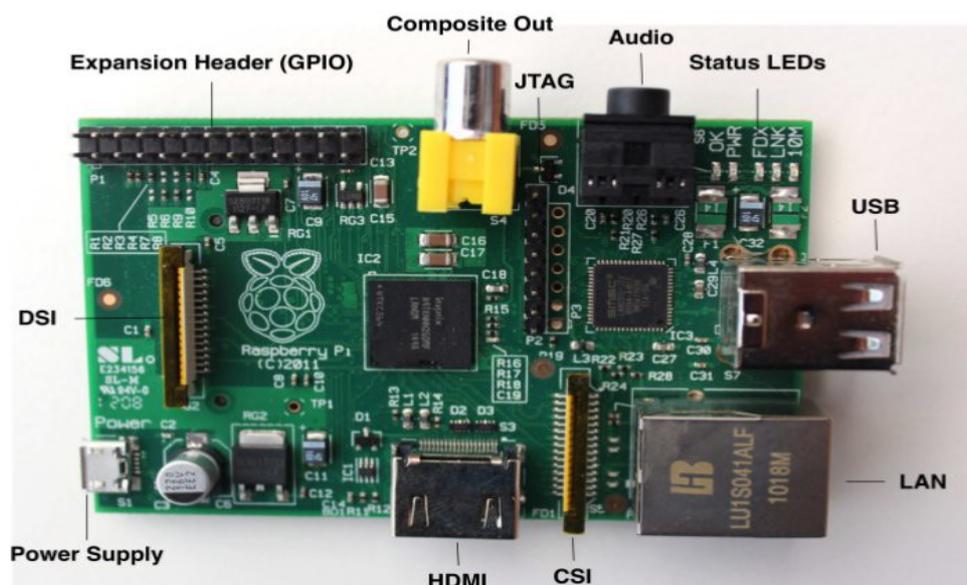


Figura 1. O lado frontal do Raspberry Pi no modelo B

Fonte: (SCHIMDT, 2012, p. 14)

Assim, como o *Raspberry Pi*, a linguagem de programação *Python*⁴ foi concebida para ensinar programação. Segundo Richardson e Wallace (2013), a fundação *Raspberry Pi* escolheu usar o *Python* como linguagem oficial no *Raspberry Pi* por vários motivos, alguns deles são: a simplicidade da linguagem, a baixa complexidade para o desenvolvedor e a velocidade para desenvolver aplicações multi-propósitos. A linguagem *Python* é um “software de código aberto (com licença compatível com a *General Public License (GPL)*” (BORGES, 2010, p. 13), sendo mantida pela *Python Software Foundation*⁵ (PSF).

4 Essa linguagem de programação foi desenvolvida em 1991 por Guido Van Rossum.

5 Fundação que mantém e promove a linguagem de programação *Python*.

De acordo com Rossum (2005, p. 3), “*Python* é ideal para *scripting* e para o desenvolvimento rápido de aplicações em diversas áreas e na maioria das plataformas”, possuindo uma vasta coleção de aplicações e *frameworks*, que “são coleções de componentes de software que foram projetados para serem utilizados por outros sistemas” (BORGES, 2010, p. 20).

Python é usado em diversos nichos, e há muitas aplicações e ferramentas desenvolvidas com essa poderosa linguagem, onde pode-se destacar o *microframework* web *Bottle*. “*Bottle* é um *microframework* web, rápido, simples e leve desenvolvido na linguagem *Python*. Ele é distribuído como um único arquivo módulo e não tem nenhuma outra dependência que a *Python Standard Library*” (HELLKAMP, 2014, p. 1).

Assim como *Python*, “*Lua* é um software *open source*. Pode-se usar em aplicações pessoais, acadêmicas e comerciais, sem nenhum custo ” (JUNG; BROWN, 2007, p. 24). *Lua* é amplamente usado em diversos nichos, como jogos, sistemas embarcados, interfaces gráficas, sistemas operacionais e softwares consagrados como o *Nmap*, *Wireshark* e *Redis* por exemplo. “O desenvolvimento de *Lua* foi em 1993 [...], e atualmente *Lua* é amplamente utilizado em todas as áreas que podem se beneficiar de uma linguagem de *script* simples, extensível, portátil e eficiente” (IERUSALIMSKY, 2010, p. 13).

Apesar de não possuir as linguagens de programação *Lua* e *Python*, o *Arduino* é um famoso microcontrolador *open source*. Segundo Banzi (2011), o *Arduino* surgiu em 2005 em um *pub* chamado *Arduino* na cidade de Ivrea na Itália. O termo microcontrolador é definido como um “pequeno computador com processador e memória, que controla as funções de vários dispositivos diários” (KARVINEN; KARVINEN, 2012, p. 17).

O microcontrolador *Arduino* é uma plataforma de *hardware open source*, baseada em uma placa simples com pinos de entrada e saída (*Input/Output*, ou *I/O*) e um ambiente de desenvolvimento integrado, inspirado na plataforma *Wiring*⁶, usando a linguagem de programação C/C++.

Além de ser usado em diversos projetos profissionais e inovadores, Evans (2012) reforça que o *Arduino* foi projetado especialmente para artistas, *designers*, *makers*⁷ e “hobbyistas” sem muito conhecimento em programação, eletrônica ou hardware. Ele tem como objetivo principal ajudar a criar projetos eletrônicos acessíveis e de baixo custo. O *Arduino* pode ser encontrado em diversas versões, Karvinen e Karvinen (2012), disponibilizaram uma imagem do *Arduino UNO* (Figura 2).

6 Plataforma de prototipagem eletrônica de *hardware open source*.

7 *Makers* são inventores, criadores que constroem seus próprios equipamentos eletrônicos.



Figura 2. Arduino UNO

Fonte: (KARVINEN; KARVINEN, 2012, p. 27)

3. Projeto Lightom

O projeto Lightom surgiu a partir da falta de um gerenciador eficiente para o sistema de iluminação pública como explicado anteriormente nesse artigo. Esse projeto aborda uma solução para o monitoramento das lâmpadas dos postes públicos, possuindo uma interface web que possibilita geolocalizar os pontos de iluminação pública, ambas interligadas em redes sem fio.

A infraestrutura de monitoramento das lâmpadas nos postes, precisou de um microcontrolador *Arduino* em sua versão nano, um sensor de luminosidade ldr, dois resistores de 10 kilohms, um resistor de 100 ohms, um diodo 1n4148, um transistor bc546, um *relay* de 5v dc 250v ac, e um *shield xbee*.

A infraestrutura do sistema web para gerenciamento dos pontos de iluminação pública, necessitou de um microcomputador *Raspberry Pi* modelo B com um cartão de memória micro SD com 8 GB de armazenamento, tendo o sistema operacional *Raspbian* GNU/Linux instalado e conectado a uma rede *mesh* formado por conexões de rede do tipo *zigbee*, conexões essas que são formadas pelos *Arduinos* citados anteriormente.

Para o desenvolvimento do sistema web para o projeto Lightom, foram instalado no *Raspberry Pi*, que está usando o sistema operacional *Raspbian* GNU/Linux, o servidor HTTP *Nginx*⁸ e o servidor HTTP WSGI⁹ *Gunicorn* para servir de infraestrutura para a aplicação de gerenciamento dos pontos de iluminação pública. O sistema web foi desenvolvido com o *microframework* web *Bottle* usando a base de dados *Redis*¹⁰ seguindo o padrão arquitetural do tipo *restful*. Foi usado também as tecnologias HTML5 e uma API para gerar mapas geolocalizados em *JavaScript*. Essa aplicação é monitorada pelo servidor *Supervisor*, que garante que o sistema nunca fique inoperante.

Nesse projeto foi usado um *Arduino* para o monitoramento de um ponto de

8 Servidor proxy HTTP e reverso, rápido, leve e performático.

9 Web Server Gateway Interface é uma interface universal entre servidores web para *Python*.

10 *Redis* é *open source*, leve, em memória e para armazenamento de dados do tipo chave-valor.

iluminação pública, e um *Raspberry Pi* para servir de servidor de baixo custo que serve de infraestrutura para o sistema web que foi desenvolvido para o projeto Lightom, ambos conectados em redes sem fio de baixa impedância. Esse projeto usou apenas um *Arduino* a nível de protótipo, em um projeto real é necessário ter vários.

Para entender melhor o projeto Lightom, foi desenvolvido o *layout* físico usando o software livre *Fritzing*, que pode ser visto na Figura 3.

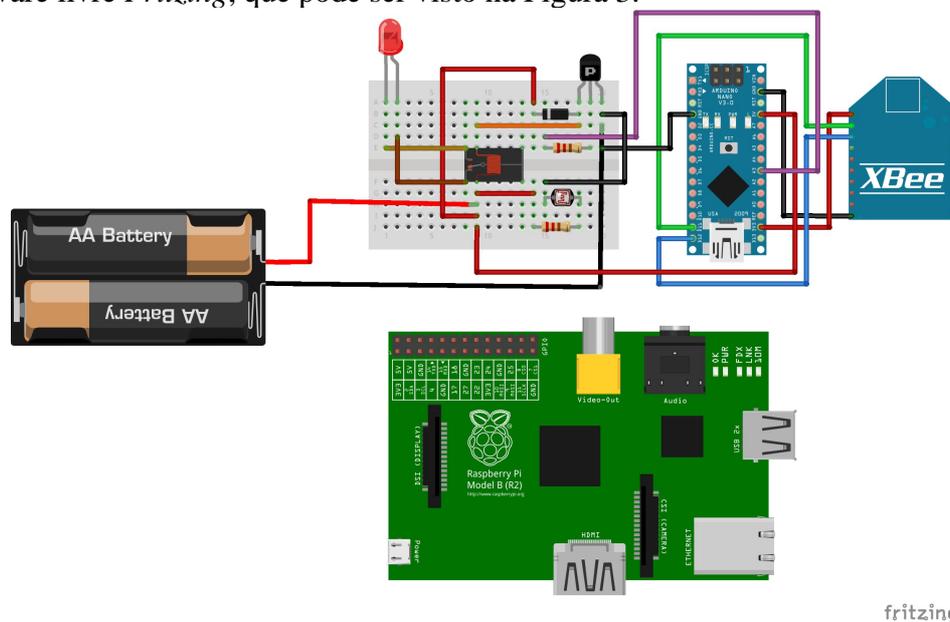


Figura 3. Layout do projeto Lightom

Fonte: (KARVINEN; KARVINEN, 2012, p. 27)

4. Considerações Finais

Apesar do projeto Lightom ainda não ter sido implantado e está em fase de prototipação, o mesmo demonstra que o gerenciamento e monitoramento eficiente dos pontos de iluminação pública podem diminuir os gastos com energia elétrica sem grandes investimentos para os órgãos competentes, como mencionado anteriormente nesse artigo, pois o projeto é *open source* e os equipamentos são de baixo custo.

O projeto Lightom, por meio de uma solução para a iluminação pública, possibilita uma melhor qualidade de vida para os cidadãos, aumentando as atividades turísticas, sociais, educacionais, culturais e esportivas, além de ajudar na prevenção contra a criminalidade.

Referências

- BARRY, Paul. Head First Python. California: O'Reilly, 2011.
- BORGES, Luiz. Python para Desenvolvedores. 2 ed. Rio de Janeiro: s.n., 2010. Disponível em: <<http://ark4n.wordpress.com/python/>>. Acesso em: 10 ago. 2013.
- ELETOBRAS, Guia técnico PROCEL gem – Gestão Energética Municipal. Rio de Janeiro: Eletrobras, 2004.
- EVANS, Brian. Beginning Arduino Programming. New York: Apress, 2011.

- KARVINEN, Kimmo; KARVINEN, Tero. Arduino Bots and Gadgets. California: O'reilly. 2012.
- HELLKAMP, Marcel. Bottle Documentation. Göttingen: s.n., 2014. Disponível em: <<http://bottlepy.org/docs/dev/bottle-docs.pdf>>. Acesso em: 10 mar. 2014.
- IERUSALIMSKY, Roberto. Programming in Lua 2th Edition. Rio de Janeiro: PUC-Rio, 2010.
- JUNG, Kurt; BROWN, Aaron. Beginning Lua Programming. Indiana: Wiley, 2007.
- RICHARDSON, Matt; WALLACE, Shawn. Getting Started With Raspberry Pi. California: O'Reilly, 2013.
- ROSSUM, Guido Van. Tutorial Python, Release 2.4.2. S.l:s.n, 2005. Disponível em: <<http://www.python.org/~guido/Publications.html>>. Acesso em: 13 out. 2013.
- STALLMAN, Richard. What is free software? 2013. Disponível em: <<http://www.gnu.org/gnu/about-gnu.html>>. Acesso em: 09 set. 2013.
- WIKIPEDIA, Single-board computer. Disponível em: <http://en.wikipedia.org/wiki/Single-board_computer>. Acesso em: 06 out. 2013.