



Universidade Católica Portuguesa  
Faculdade de Ciências Sociais - Centro Regional de Braga  
Mestrado de Ensino em Informática

# O Potencial Educativo do *Raspberry Pi*

Alexandra Cardoso: ig2949@gmail.com

Edgar Costa: info@edgarcosta.net

Graça Sousa: gracialorga@gmail.com

Mestrandos em Ensino de Informática da Faculdade de Ciências Sociais – Centro Regional de Braga

**Resumo** – O presente artigo apresenta uma abrangente ferramenta educacional, de baixo custo, direcionada para o ensino da programação através de uma forma séria e ao mesmo tempo de uma forma divertida, a que se alia o ensino de Linux. Neste contexto, e dadas as funcionalidades que o *Raspberry Pi* disponibiliza para a criação de uma aprendizagem efetiva, colaborativa e interativa, tal como para a (re)construção do pensamento lógico e crítico, damos destaque às suas características. Urge-nos também apresentar os resultados de um estudo exploratório realizado na disciplina Sistemas de Informação do Curso Técnico de Informática de Gestão, acerca das perceções dos sujeitos envolvidos onde destacamos as potencialidades e eventuais limitações deste microcomputador. Num primeiro momento realçamos as suas características e aplicações e, num segundo momento, focaremos a sua utilidade educativa, bem como o reconhecimento desta ferramenta como um importante recurso no ensino da programação.

**Palavras-chave:** Aprendizagem, Ciências da Computação, Programação, *Raspberry Pi* e Tecnologia Educativa.

## Introdução

A “escola” está em mudança! Por isso mesmo, já não pode ficar alheia à ciência da computação. Com os avanços tecnológicos, as escolas necessitam fomentar ainda mais o interesse e gosto dos alunos para o estudo desta ciência. Em diversas investigações (Gomes & Mendes, 2007; Gomes, Henriques & Mendes, 2008) verificamos que os “nossos” alunos ainda sentem grandes dificuldades nesta área, não se sentindo motivados, em parte, devido à conotação negativa que está associada à disciplina de programação. Embora a aprendizagem da programação não seja tarefa fácil, já que exige do aluno bastante esforço, empenho e um estudo muito prático e intensivo, a verdade é que os métodos tradicionais de ensino também já não se encontram adequados às necessidades e estratégias de motivação da maioria dos alunos.

Em 2006, surge um projeto intitulado *Raspberry Pi* criado com o intuito de promover o estudo e o gosto pela ciência da computação nas escolas do ensino básico e ensino secundário, potenciando um instrumento capaz de responder às necessidades atuais dos discentes.

Neste contexto, o presente artigo tem como objetivo apresentar, num primeiro momento, as características do *Raspberry Pi* e suas aplicações e, posteriormente, de forma a avaliar a sua utilidade em contexto educativo, através de um estudo de caso exploratório e, tendo como método de recolha de dados a observação participante, focamos a sua utilidade educativa, bem como o reconhecimento desta ferramenta como um importante recurso no ensino da programação, junto de um grupo de alunos que frequentam o Curso Técnico de Informática de Gestão na disciplina de Sistemas de Informação.

## Estado da Arte

Após lidarmos de perto com o *Raspberry Pi*, aquando do nosso estudo exploratório, parece-nos pertinente partilhar que este “pequeno” computador é um equipamento bastante interessante e com elevado potencial, principalmente, no ensino e no desenvolvimento da arquitetura dos computadores, das linguagens de programação e até mesmo a nível do entretenimento. Implementado e desenvolvido sem fins lucrativos, o *Raspberry Pi* surgiu para que seja utilizado por crianças e jovens de todo o mundo, sobretudo por aqueles que não possuem computador de forma a poderem aprender programação e a programar.

Indubitavelmente, uma das vantagens do *Raspberry Pi* para a educação é a aprendizagem de programação e do *Linux* ao nível da gestão do sistema operativo, em ambiente gráfico e, principalmente, através da linha de comandos (Halfacree & Upton, 2012, Membrey & Hows, 2012).

O *Raspberry Pi* destaca-se por se poder usar como um outro qualquer computador, bem como pela sua capacidade de integração com a eletrónica, daí os componentes estarem visíveis para que os utilizadores possam observar a arquitetura deste pequeno sistema.

Segundo Green<sup>1</sup> (2012), projetos como o *Raspberry Pi*, com as plataformas e ecossistemas certos, potenciam uma revolução na educação tecnológica. Proveniente do sucesso emergente deste dispositivo, a *Google* decidiu apoiar esta iniciativa para auxiliar a criação da próxima geração de engenheiros informáticos e cientistas da computação, de acordo com o site *The Institution of Engineering and Technology*<sup>2</sup>.

Semelhante ao projeto *One Laptop Per Child (OLPC)*<sup>3</sup>, onde a prossecução da missão é fornecer, a cada criança de países subdesenvolvidos, um *laptop* como ferramenta pedagógica de apoio ao processo de ensino aprendizagem em sala de aula (e familiar), com alta durabilidade, baixo consumo de energia e, principalmente, com acesso à Internet, o *Raspberry Pi* destaca-se não só pelo seu baixo custo (cerca de 45€, mesmo depois de se adquirir todos os acessórios para o colocar funcional), mas também pelo seu tamanho reduzido, bem como pelo incentivo ao nível da programação nas escolas tanto nos países desenvolvidos como nos países em desenvolvimento.

Apesar de vivermos numa era digital, Bianchetti (2001) alerta-nos que esta realidade “não é uma realidade mundial no sentido de abrangência total de países e blocos”, pois em países do terceiro mundo, onde predominam limitações económicas e tecnológicas (e mesmo no nosso país, embora em menor número, apesar do lançamento do projeto Magalhães e outros), “ainda persistem elevados índices de analfabetos, semi-analfabetos, analfabetos funcionais e cyberanalfabetos” e onde “as ‘trocas’ de informações se dão num único sentido” (citado por Moreira, 2004: 129). De acordo com o site *Internet World Stats*<sup>4</sup> verificamos que, em 2012, como nos ilustra a Fig. 1, os países menos desenvolvidos são os que apresentam um menor número de utilizadores da Internet.

“A promoção do processo de constituição de uma sociedade mais justa e aberta às diferenças demanda que todo aluno deva ter, de maneira irrestrita e inclusiva, acesso ao

---

<sup>1</sup> CEO da *Premier Farnell*

<sup>2</sup> <http://eandt.theiet.org/news/2013/jan/google-raspberry-pi.cfm>

<sup>3</sup> <http://one.laptop.org/>

<sup>4</sup> <http://www.internetworldstats.com/stats.htm>

equipamento e à possibilidade de interagir para ter acesso ao conhecimento” (Miranda *et al.*, 2007: 288).

A informação, sendo um facto “intencionalmente seleccionado, codificado e submetido a um processo de refinamento, informatizado ou não, para a veiculação das ideias, imagens, sons, cores, mensagens” (Catapan, 2003: 4) é essencial, na medida em “que está na base do conhecimento e da comunicação entre as pessoas” (Coutinho & Lisboa, 2011: 5). Por isso, é necessário torná-la facilmente acessível de uma forma divertida e de forma envolvente (Jain, Singhal & Gupta, 2010).

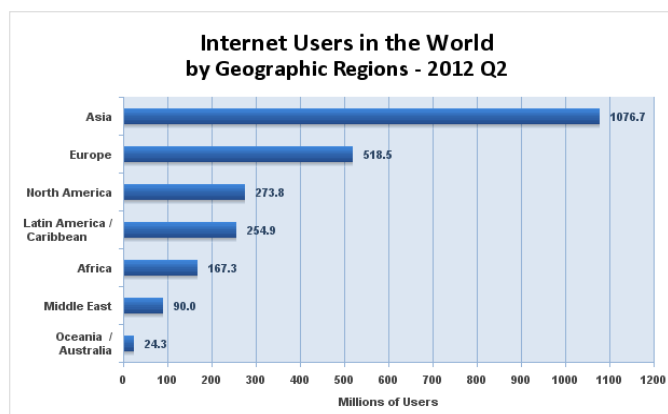


Fig. 1 – Utilizadores da Internet no mundo  
(Fonte: <http://www.internetworldstats.com/stats.htm>)

Ter acesso a um computador a baixo custo e com ligação à Internet significa ter acesso a benefícios educacionais, tais como “uma melhor qualidade do ensino e ambientes de aprendizagem mais ricos e motivadores para os discentes” (Miranda *et al.*, 2007: 281). “As tecnologias têm influência nas estratégias de aprendizagem que activam os processos mentais” (Pires, 2009: 48). Considerados como “veículos de ideias poderosas e de mudanças culturais enormes”, os computadores “podem fazer com que as pessoas estabeleçam uma nova forma de relação com o conhecimento” (Pinto, 2010: 34), que mais não é do que “um processo dinâmico de interpretação, de reelaboração das informações a que são conferidos sentido e significados operados pelos sujeitos no processo da comunicação” (Catapan, 2003: 4). É desta forma, que “o aluno deixa de ser um agente passivo e passa a ser um agente activo na construção do seu próprio saber” (Pires, 2009: 50), não só pelos processos formais mas também pelos informais.

Nesta era da Sociedade da Informação, onde predomina “um mundo onde o fluxo de informações é intenso, em permanente mudança, e “onde o conhecimento é um recurso flexível, fluido, sempre em expansão e em mudança”” (Hargreaves, 2003, citado por Coutinho & Lisboa, 2011: 5), “é imprescindível que se estabeleçam critérios para organizar e seleccionar as informações, e não simplesmente ser influenciado e “moldado” pelos constantes fluxos informativos disponíveis” (Coutinho & Lisboa, 2011: 10). Por isso, em contexto educativo é fulcral “potencializar a construção conceitual a partir da interpretação da informação e da reelaboração da mensagem, isto é, pelo compromisso com a construção de novos conhecimentos” (Catapan, 2003: 4).

É nesta linha de pensamento que estamos em crer que, tal como não existe uma realidade mundial, também ainda existem crianças que não são consideradas verdadeiras nativas digitais (Halfacree & Upton, 2012). É necessário que as crianças e jovens sejam capazes de manipular dados, ultrapassar problemas e até construir produtos digitais que jamais imaginariam existir. É fundamental colocá-las a pensar, para que sejam alunos ativos “com poder para criar e gerir o seu próprio conhecimento, evoluindo como resultado da sua própria experiência adquirida” (Pinto, 2010: 2). Face a isto, é indispensável que as ciências da computação, esta habilidade de interação homem-

máquina, literacia processual ou simplesmente programação (Prensky, 2008), faça parte do currículo das escolas, onde também é importante esclarecer como esta se desenvolve (Schmidt, 2011). Se esta geração de alunos não tiver a ideia de como as coisas funcionam, caminharemos para um “lugar” muito pobre no que se trata de inovação (Mitchel, 2012: 26).

O pensamento lógico e algorítmico é importante, uma vez que particulariza a evolução cognitiva e o desenvolvimento de capacidades para ultrapassar os problemas educacionais, em ciências como a matemática ou a física, entre outras. Geralmente, os alunos não entendem a importância do desenvolvimento de algoritmos e exercícios de construção lógica. Contudo, é importante ter em mente que, se as técnicas de construção lógica forem desenvolvidas, a programação tornar-se-á muito mais fácil (Jain, Singhal & Gupta, 2010). Também Gomes, Henriques e Mendes (2008) sustentam esta opinião ao afirmarem que “a preocupação principal deveria ser antes de mais o desenvolvimento da capacidade de resolução de problemas, aparecendo a linguagem de programação apenas como um veículo para concretizar essa resolução, ou seja, para expressar o algoritmo ou estratégia de resolução” (p. 95). Por isso, a capacidade para pensar logicamente deve ser enriquecida.

Prensky (2008) destaca a importância desta nova literacia, justificando que, atualmente e no futuro, a literacia será a capacidade não só de ler e de escrever ou de poliglótismo, mas também a capacidade de dominar as ciências da computação. Refere o autor que esta ciência é um dos componentes mais importantes do nosso quotidiano e a capacidade do ser humano em transformar as máquinas em componentes, que fazem o que desejam, torna-as ainda mais valiosas.

Perante esta realidade, afigura-se-nos que os atuais métodos de ensino, os quais se encontram bem desfasados da realidade a nosso ver, já que descuram formas de interação, de colaboração e de desenvolvimento do raciocínio lógico, fazem “com que os alunos tenham dificuldade em compreender o comportamento dos programas informáticos e reconhecer a relevância das tarefas que lhes são propostas” (Price *et al.*, 2002, citado por Santos *et al.*, 2007: s. p).

Existem autores (Jain, Singhal & Gupta, 2010) que referem que a atual metodologia para o ensino de ciências da computação ainda requer grandes volumes de leitura de texto, o que se repercute numa aprendizagem monótona.

Neste cenário, parece-nos que os professores estão mais concentrados em ensinar a linguagem de programação e seus detalhes sintáticos, em vez de promover a resolução de problemas utilizando uma linguagem de programação (Gomes & Mendes, 2007). É primordial entendermos, como professores que somos, que os nossos alunos são todos diferentes, assim como cada um deles apresenta ritmos e estilos distintos de aprendizagem. Importante é por isso, que abandonemos “as práticas pedagógicas que valorizam a transmissão do conhecimento” e optemos “por práticas pedagógicas mais activas, criativas, dinâmicas, motivadoras e apoiadas na aprendizagem pela descoberta” (Merrelho, 2010: 10), tornando-nos em mediadores da aprendizagem.

Com a proliferação da tecnologia é fundamental que os discentes não sejam meros consumidores de conteúdos, mas que também possam (e devam) ser criadores de conteúdo.

Elucida-nos Pinto (2010) que “a aprendizagem deve ser uma actividade interactiva e não individualista. Todas as acções educativas devem ocorrer em interacção social, promovendo uma educação mais efectiva em intercâmbio constante entre colegas e professores” (p. 8). Por isso, é necessário motivar os alunos e estimulá-los para que a sua criatividade sobressaia, já que no processo de aprendizagem “as pessoas e particularmente as crianças e os jovens têm uma necessidade inata de compreender o mundo e tudo o que nele existe para terem sucesso nas acções que empreendem” (Speck-Hamdan, 2005, citado por Teixeira, 2011: 77).

A programação é uma área científica complexa, que requer esforço e uma abordagem especial na forma como é aprendida e ensinada. Parece-nos pois pertinente, que os alunos comecem desde cedo (a partir dos 9/10 anos de acordo com o *síte* do governo do Reino Unido<sup>5</sup>) a lidar com a programação, já que esta acaba por “desenvolver nos mais novos a capacidade de pensar” (Pólya, 2003, citado por Pinto, 2010: 8), pois requer habilidades como abstracção, generalização, transferência e pensamento crítico. Desta forma, também se colmatava a dificuldade que a maioria dos alunos sente, ao ingressar no ensino superior, quando se depara pela primeira vez com a aprendizagem da programação (Gomes & Mendes, 2007).

Recentemente, verificámos mudanças no sistema de ensino, no que concerne à disciplina de Tecnologias de Informação e Comunicação (TIC), no 7.º e no 8.º anos de escolaridade em Portugal, o que nos parece que estamos a “caminhar” no bom sentido, apesar de muito haver a percorrer. As atuais metas curriculares, vistas como objetivos finais de aprendizagem, visam o desenvolvimento de atividades que estimulem “uma literacia digital generalizada” nos alunos, de forma que sejam capazes de fazer “análise crítica da função e do poder das tecnologias de informação e comunicação”, bem como urgem desenvolver “as capacidades analíticas dos alunos, através da exploração de ambientes computacionais apropriados à sua idade” (Horta *et al.*, 2012: 1). É desta forma, que se prevê que os alunos criem “um produto original de forma colaborativa e com uma temática definida, com recurso a ferramentas e ambientes computacionais apropriados à idade e ao estágio de desenvolvimento cognitivo dos alunos, instalados localmente ou disponíveis na Internet, que desenvolvam um modo de pensamento computacional, centrado na descrição e resolução de problemas e na organização lógica das ideias” (Horta *et al.*, 2012: 13).

Uma entrevista do Jornal de Ciência, Tecnologia e Empreendedorismo<sup>6</sup> a Jaime Rei, coordenador do projeto “Clube da Robótica”<sup>7</sup> da Escola Básica de São Gonçalo em Torres Vedras, revela o quanto o ensino da programação é fundamental na vida dos estudantes. Elucida-nos este professor que “o aluno precisa estar preparado para enfrentar desafios, ampliar horizontes, actuar no presente e no futuro com sucesso, e mais, precisa estar qualificado para o mercado de trabalho. O profissional do novo milénio deverá contar com uma sólida base de conhecimento e, ao mesmo tempo, ser criativo para encontrar soluções para os desafios que surgem a cada dia”.

Nesta nova era digital, é importante aprender, pelo menos, uma linguagem de programação, porque todas as inovações e tecnologias são baseadas em conhecimentos profundos de

---

<sup>5</sup> <https://www.gov.uk/>

<sup>6</sup> <http://www.cienciahoje.pt/>

<sup>7</sup> <http://roboticasaogoncalo.blogspot.pt>

computadores, sistemas operativos, API<sup>8</sup> e periféricos de *hardware* (Bogdanchikov, Zhaparov & Suliyeu, 2013).

Nestes novos cenários interativos da educação o *Raspberry Pi* revela-se um dispositivo com características inovadoras que, tendo um contributo valioso no ensino aprendizagem, poderá resolver grande parte dos problemas ancorados à programação, bem como motivar cada vez mais as crianças e jovens para o gosto da informática.

## O Raspberry Pi

O *Raspberry Pi*, como ilustra a Fig. 2, é um microcomputador de reduzidas dimensões, do tamanho de um cartão de crédito, desenvolvido pela *Raspberry Pi Foundation*<sup>9</sup> e pela Universidade de Cambridge, que foi criado com o intuito de promover o estudo da ciência da computação nas escolas do ensino básico ao ensino secundário.

Este microcomputador, onde todo o *hardware* é integrado numa única placa de circuito impresso, está disponível em dois modelos: o A e o B. Ambos são baseados num SoC<sup>10</sup> BCM2835, que integra um processador do tipo ARM a 700Mhz e uma unidade de processamento gráfico. São constituídos ainda por uma entrada de cartão SD para armazenamento de dados, pois não incluem uma memória não volátil, como um disco rígido. É, por isso, necessário adquirir um cartão de memória SD, com 4GB no mínimo, para instalar o sistema operativo bem como outras aplicações, de forma a ficar totalmente funcional. Salientamos, no entanto, que embora estejam à venda cartões com o sistema operativo pré-instalado, o processo de instalação do mesmo é bastante simples como referido no nosso *site*<sup>11</sup>, construído aquando desta investigação.

Como este microcomputador tem dimensões muito reduzidas e de forma a protegê-lo e/ou de forma a facilitar o seu transporte ou manuseamento, aconselhamos os utilizadores a adquirir uma *case*, por exemplo igual à da Fig. 3.

A grande diferença entre os dois modelos, embora ambos forneçam um nível básico de funcionalidade, é que o modelo B, alvo do nosso

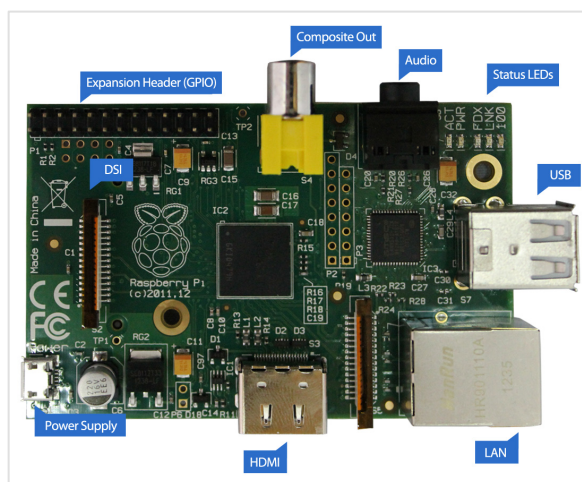


Fig. 2 - *Raspberry Pi*: Modelo B  
(Fonte: <http://www.raspedu.com/raspberry-pi/anatomia/>)



Fig. 3 - *Raspberry Pi* Case  
(Fonte: Elaborada pelos autores)

<sup>8</sup> Acrónimo de *Application Programming Interface*

<sup>9</sup> <http://www.raspberrypi.org/>

<sup>10</sup> Acrónimo de *system on chip* (um computador num único chip)

<sup>11</sup> <http://www.raspedu.com/>

estudo, possui um controlador Ethernet<sup>12</sup>, duas portas *USB*<sup>13</sup>, que podem servir para ligar um teclado e um rato, caso não esteja ligada a servidor de serviços, enquanto o modelo A, que sendo inferior nas suas características e por isso mais barato, tem apenas uma porta *USB* e não possui uma ligação de rede.

Como referimos, o *Raspberry Pi* “corre” num processador *ARM*, por isso não é possível instalar o sistema operativo *Windows* ou o sistema operativo *Ubuntu*. No entanto, estão disponíveis gratuitamente sistemas operativos compatíveis com este microcomputador, como é o caso do *Raspbian*, uma distribuição *Linux* baseada em *Debian*, o qual optamos por instalar no nosso projeto. Alertámos, contudo, que a escolha do sistema operativo recai sobre o propósito de utilização deste microcomputador. No nosso caso, optámos pelo *Raspbian*, que sendo um sistema estável, que é atualizado periodicamente e com bastante suporte pela comunidade *on-line*, parece-nos o mais indicado para utilizarmos com os nossos alunos. Este sistema operativo traz, por padrão, o ambiente *desktop LXDE*<sup>14</sup> que é um ambiente gráfico utilizado em sistemas com baixas capacidades técnicas, e corre com o gestor de janelas *Openbox*<sup>15</sup>, permitindo que o aspeto gráfico do *Raspbian* (vide Fig. 4) seja idêntico ao *desktop* de qualquer outro sistema operativo (ex: *Windows* ou *OS X*).

Tal como noutros ambientes gráficos, no menu iniciar do *Raspbian*, os itens estão organizados por tipo de aplicação ou ferramenta (acessórios, educação, Internet, programação, ferramentas do sistema, preferências, executar e *logout*). Ao nível das aplicações base engloba o *browser Midori*<sup>16</sup>, além de diversas ferramentas de programação como o *Python*<sup>17</sup>, o *Scratch*<sup>18</sup> e o *Squeak*<sup>19</sup> que permitem o ensino de algoritmia através de métodos entusiasmantes e divertidos. Para além destes, podemos instalar inúmeras ferramentas, também elas dedicadas à programação, como por exemplo o *KidsRuby*<sup>20</sup>.

Disponível neste projeto está a *Pi Store*<sup>21</sup>, que é facilmente instalada no *Raspberry Pi* onde se pode fazer a transferência de diversos jogos e aplicações disponíveis para o mesmo, como por exemplo, o *LibreOffice*.

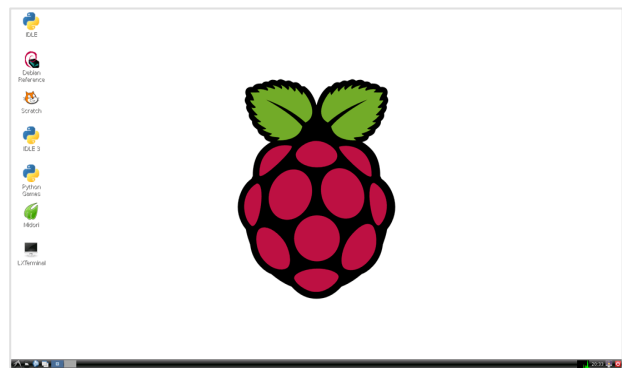


Fig. 4 - Sistema Operativo *Raspbian*  
(Fonte: <http://www.raspedu.com/raspberry-pi/sistemas-operativos/>)

<sup>12</sup> Protocolo de ligação para redes locais

<sup>13</sup> Acrónimo de *Universal Serial Bus*

<sup>14</sup> Acrónimo de *Lightweight X11 Desktop Environment*

<sup>15</sup> Gestor de janelas para sistemas *Unix-like*

<sup>16</sup> *Browser web* com *WebKit GTK+*

<sup>17</sup> <http://www.python.org>

<sup>18</sup> <http://scratch.mit.edu>

<sup>19</sup> <http://www.squeak.org>

<sup>20</sup> <http://www.kidsruby.org>

<sup>21</sup> Loja de aplicações para o *Raspberry Pi*



## Aplicações de Programação Para o Raspberry Pi

Como já referido, existem inúmeras aplicações destinadas ao ensino da programação. Contudo, neste nosso projeto apenas destacaremos o *Python*, o *Scratch*, o *Squeak* e o *KidsRuby*, pelo facto de auxiliarem crianças e jovens no ensino dos conceitos básicos de programação.

### O Python

O *Python* (vide Fig. 5) é considerado uma excelente linguagem de programação como ponto de partida para o desenvolvimento de código. Além de ser *open source*, é também uma linguagem que aglutina vários paradigmas de programação (programação estruturada, orientada-a-objetos e funcional). Sendo uma linguagem interpretada, permite escrever programas ou *scripts* e executá-los em vez de serem compilados.



Fig. 5 - Logótipo *Python*  
(Fonte: <http://www.python.org>)

Esta linguagem, de acordo com a *Wikipédia*<sup>22</sup>, foi projetada com a filosofia de enfatizar a importância do esforço do programador sobre o esforço computacional.

Diversos autores (Bogdanchikov, Zhaparov & Suliyeu, 2013) sustentam que o *Python* é a linguagem de programação mais adequada para programadores iniciantes aprenderem a programar, porque tem ferramentas poderosas que refletem a forma de pensar e de implementar o código.

O ambiente de desenvolvimento integrado do *Python* no *Raspbian* é o *IDLE*<sup>23</sup>. A razão pela qual estão disponíveis duas versões é, pelo facto de existirem atualmente duas versões do *Python*, em simultâneo, que não são compatíveis entre si. Por isso, é necessário alguma atenção na consulta de documentação durante o processo de desenvolvimento de código.

Alertámos, contudo, que o desenvolvimento de programação sem observar o efeito real do código fonte (vide Fig. 6) não é estimulante, daí que durante este projeto tenhamos apoiado os nossos alunos no desenvolvimento de pequenos programas nesta linguagem.

```
1 # Calculo da Formula Resolvente
2
3 import math
4
5 def formResolv(a, b, c):
6     result = (b*b)-(4*a*c)
7     if result < 0:
8         return "Impossivel calcular raiz quadrada de numeros negativos!"
9     total = []
10    total.append((-b + math.sqrt(result)) / (2*a))
11    total.append((-b - math.sqrt(result)) / (2*a))
12    total.sort()
13    return total
14
15 print formResolv(2, -10, 5)
```

Fig. 6 – Código fonte *Python*  
(Fonte: Elaborado pelos autores)

### O Scratch

O *Scratch* (vide Fig. 6), desenvolvido pelo MIT<sup>24</sup> Media Lab, é uma linguagem de programação educativa, e que inspirada na linguagem *Logo*<sup>25</sup>, e *Etoys*<sup>26</sup> destina-se a ser usada por crianças e jovens a partir dos 8 anos.



Fig. 6 - Logótipo *Scratch*  
(Fonte: <http://scratch.mit.edu>)

<sup>22</sup> <http://pt.wikipedia.org/wiki/Python>

<sup>23</sup> Ambiente de desenvolvimento integrado para *Python*

<sup>24</sup> *Massachusetts Institute of Technology*

<sup>25</sup> Linguagem de programação interpretada, voltada para crianças, jovens e até adultos

<sup>26</sup> Linguagem de programação orientada a objetos para uso na educação



Não exige conhecimentos prévios de programação, uma vez que o desenvolvimento é feito através de uma *interface* gráfica (vide Fig. 7), que permite que os programas sejam desenvolvidos através de blocos coloridos, com uma manipulação semelhante aos brinquedos *Lego*, que executam determinada operação.

Sendo gratuita “o uso da linguagem *Scratch* na introdução de conceitos algorítmicos tem sido bem vista” (Scaico *et al.*, 2012: s. p). Por isso, programar nesta aplicação é muito mais simples do que nas tradicionais linguagens de programação. “É uma linguagem adequada para entender as principais estruturas, que são a base para a criação de algoritmos” (Trajano, 2010, citado por Scaico *et al.*, 2012: s. p). Por ser uma ferramenta amplamente utilizada na educação, o *Scratch* faz parte das aplicações do *Raspbian*.

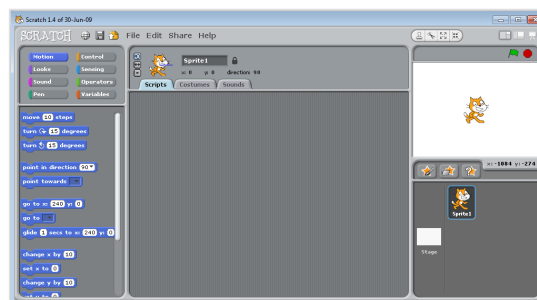


Fig. 7 - Interface do Scratch  
(Fonte: <http://www.raspedu.com/explorar/scratch/>)

## O Squeak

Tal como o *Python* e o *Scratch*, também o *Squeak* (vide Fig. 8) faz parte integrante do *Raspbian*.

Considerado como a primeira linguagem de programação orientada a objetos e inspirada em *smalltalk*, o *Squeak* “abre um mundo de possibilidades de exploração e experimentação em todas as áreas de conhecimento, excepcionalmente visível na simulação e representação de modelos” (Allen-Conn & Rose, 2003, citado por Valente & Osório, 2006: s. p).



Fig. 8 - Logótipo Squeak  
(Fonte: <http://www.squeak.org>)

Desenvolvido na *Apple* e enquanto ferramenta *open source*, o seu objetivo é “colocar o aluno no centro do processo de ensino-aprendizagem, uma vez que se considera que é através dessa construção que o aluno produz a sua aprendizagem” (Varela, 2007, citado por Silva, 2009: 26).

O *Squeak* é “a implementação de uma proposta sobre como podemos utilizar os computadores para interagir com a informação, para criarmos o nosso próprio conhecimento” (Deck *et al.*, 2005, citado por Teixeira, 2011: 84), daí ter sido projetado para ajudar as crianças a aprenderem colaborativamente.

O modo de trabalhar no *Squeak* consiste em selecionar, criar e importar objetos, para depois arrastá-los e soltá-los, “funciona como um simulador de mundos virtuais, onde é possível experimentar, reproduzindo fenómenos e processos reais ou inventados” (vide Fig. 9), como refere o *site* do Centro de Competências em Tecnologias Educativas<sup>27</sup>.

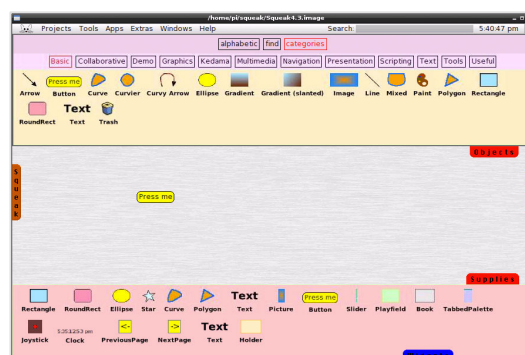


Fig. 9 - Interface Squeak  
(Fonte: <http://www.raspedu.com/explorar/squeak/>)

<sup>27</sup> [http://ccc.cercifaf.org.pt/index.php?option=com\\_content&task=view&id=56&Itemid=125](http://ccc.cercifaf.org.pt/index.php?option=com_content&task=view&id=56&Itemid=125)

## O KidsRuby



Fig. 10 - Logótipo KidsRuby  
(Fonte: <http://www.kidsruby.org>)

O KidsRuby (vide Fig. 10) é um utilitário *open source* orientado a auxiliar as crianças e jovens a aprender os conceitos básicos da linguagem de programação Ruby, que é uma linguagem de programação interpretada e orientada a objetos.

O programa apresenta uma *interface* bastante simplificada e intuitiva, procurando uma interação maior com a criança. Uma das limitações desta aplicação é o facto de ainda não existir uma versão em português.

A interface desta aplicação, como ilustra a Fig. 11, está dividida em duas seções. O lado esquerdo é o local onde o código é escrito, enquanto a do lado direito atua como um guia orientador, para além de mostrar a execução do código programado.

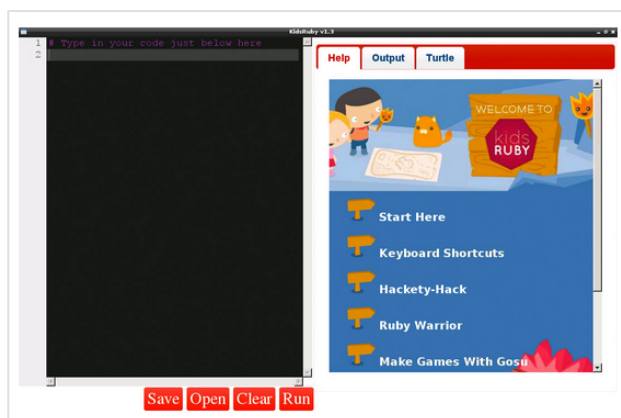


Fig. 11 - Interface KidsRuby  
(Fonte: <http://www.raspedu.com/explorar/kidsruby/>)

## O Estudo Exploratório

O presente estudo, enquadra-se numa tipologia de natureza exploratória de carácter descritivo e que, integrando-se num projeto de investigação tendo como eixo a identificação e exploração das potencialidades do *Raspberry Pi* na educação, tal como a ampliação do interesse, a participação e a interação dos alunos, contextualiza-se numa metodologia qualitativa, já que “centra-se na compreensão dos problemas, analisando os comportamentos, atitudes ou os valores” (Sousa & Baptista, 2011: 56).

O caso em estudo reporta-se ao período desta investigação e refere-se a uma turma de 26 alunos com idades entre os 15 e os 18 anos, do 10.º ano do curso Técnico de Informática de Gestão da delegação de Pedome da Escola Profissional Bento de Jesus Caraça.

Refere Reis (2008) que “a pesquisa exploratória é o primeiro passo de qualquer pesquisa” (p. 55), e tem como objetivo “proceder ao reconhecimento de uma dada realidade pouco ou deficientemente estudada” (Sousa & Baptista, 2011: 57). Neste sentido, este tipo de pesquisa que

“busca ampliar o número de informações sobre determinado ponto que se quer investigar” (Bastos, 2009: 75), permite descobrir novas ideias com o intuito a um aprofundamento posterior.

Partindo dos objetivos que delineamos, utilizámos como método de recolha de dados a observação participante, onde um dos investigadores deste estudo se identifica como professor da turma, inteirando-se do “meio a “investigar”” e acedendo, “às perspetivas das pessoas com quem interage” (Sousa & Baptista, 2011: 88).

### **RaspEdu.com**

Refere Costa (2010) que “a Internet tornou-se num meio de comunicação que possibilita comunicar de uma forma rápida, à distância, com pessoas de todo o planeta e em tempo real”, onde se utiliza “uma linguagem escrita sincopada, combinando *emoticons* (...), imagem, som e voz”, e possibilita “uma comunicação de muitos para muitos” (p. 49), permitindo desta forma, uma comunicação interativa, bem como a troca de informações. É neste sentido, que partimos para a concretização de um repositório de informação sobre a utilização deste microcomputador em contexto educativo, designado por *RaspEdu* (vide Fig. 12), e que disponibilizamos na *web* através de um endereço eletrónico próprio.

Outra das razões para a construção deste repositório, relaciona-se com o facto de, aquando a revisão bibliográfica, constatarmos um baixo nível de conteúdos disponíveis em língua portuguesa, no que diz respeito ao suporte *on-line* sobre o *Raspberry Pi*.

Neste sentido, todas as atividades levadas a cabo, desde a aquisição do dispositivo e dos acessórios fundamentais para o seu funcionamento até à configuração de serviços de acesso remoto constam neste *website*.

Refere Carvalho (2006) que “um *site* educativo tem que ter subjacente os princípios básicos estruturais, de navegação, de orientação, de *design* e de comunicação de qualquer *site* mas, para além disso, um *site* educativo tem que motivar os utilizadores a quererem aprender, a quererem consultar e a quererem explorar a informação disponível” (p. 7). Nesta linha de pensamento, Taylor, Kent e White (2001) “ressaltam a utilidade das informações que os web sites oferecem; as informações não têm que estar ali sem nenhum motivo, mas serem úteis” (citado por Junior *et al.*, 2013: 6), para que sejam assimiladas pelos utilizadores.

Daí, o nosso cuidado em que o mesmo estivesse bem organizado de forma a não fazer perder tempo aos utilizadores e a fornecer-lhes informação pertinente e interessante da forma mais simples e intuitiva possível. Não obstante, as informações disponibilizadas no nosso *site* estão bem legíveis e apresentam-se numa linguagem clara e acessível, bem como existe uma alta qualidade das imagens e uma harmonia de cores. No que concerne à *interface*, a mesma é simples, intuitiva e muito fácil de utilizar, refletindo o propósito da aprendizagem. Relativamente ao nome do *site* e,



Fig. 12 - *Raspedu.com*  
(Fonte: <http://www.raspedu.com/>)

como o próprio indicia, centra-se em contexto educativo, e tendo em conta o objetivo da sua conceção, o mesmo encontra-se adequado para professores e alunos.

### ***Raspberry Pi* em Sala de Aula**

Desde a aquisição do *Raspberry Pi* que o grupo de alunos participou ativamente na instalação e configuração do sistema. Procurou-se enquadrar as características técnicas deste microcomputador com o plano modular do curso, composto, nesta fase do ano letivo, por módulos ligados à programação, à arquitetura de computadores e ao planeamento e gestão de bases de dados. Neste sentido, apuramos a possibilidade de interligar as capacidades do *Raspberry Pi* às necessidades educativas em contexto de sala de aula, nomeadamente, à utilização deste sistema como servidor de base de dados SQL<sup>28</sup>. Em contexto prático, procedeu-se à instalação e configuração do *SQLite*<sup>29</sup> e, posteriormente, à realização dos exercícios da disciplina de Sistemas de Informação junto do servidor de base de dados.

No seguimento deste processo, convenientemente preparado, convidamos os alunos à participação ativa no desenvolvimento de *scripts Python* dos algoritmos previamente desenvolvidos na disciplina de Linguagens de Programação do mesmo curso. Na prossecução do ensino diferenciado, designado pelo projeto educativo de escola, propôs-se aos alunos menos preparados a criação de pequenos programas em *KidsRuby*, um utilitário orientado a auxiliar as crianças e jovens a aprender os conceitos básicos da linguagem de programação *Ruby*.

Relativamente à arquitetura de computadores, foi possível com este equipamento demonstrar a operacionalidade entre o *hardware* e o *software*, nomeadamente, com a gestão de pacote de aplicações através da linha de comandos do *Linux*.

### **Apresentação de Resultados**

No lançamento deste repto, sabíamos, à partida, que nos íamos deparar com algumas desvantagens na aplicação deste microcomputador em sala de aula. A referir o facto do ensino, neste caso particular, ser modular e alguns dos módulos importantes à integração deste projeto não contextualizarem no elenco modular do presente ano letivo.

Contudo, decidimos arriscar! Por isso, procuramos saber junto do nosso público-alvo se conheciam outros sistemas operativos que não o *Windows* ou o *OS X*. Prontamente os alunos acederam à nossa questão, onde, de uma forma geral, nos informaram que conhecem o *Linux* de ouvir falar, mas nunca trabalharam com ele, o que tornou este desafio ainda mais interessante.

Dado que para o nosso público-alvo o sistema operativo *Linux* era novidade, foi necessário recuarmos um pouco e explicar alguns conceitos da arquitetura deste sistema operativo, como por exemplo, que o termo *Linux* é utilizado para designar sistemas operativos que usam o núcleo *Linux* baseado em *Unix*. Outra dificuldade com que nos deparamos foi trabalhar junto dos alunos o modo de consola ou linha de comandos tão célebre nestes sistemas operativos. Apesar de, tanto a

---

<sup>28</sup> *Structured Query Language*

<sup>29</sup> Biblioteca em linguagem C que implementa um banco de dados SQL embutido

instalação e a configuração terem sido realizadas nos dois modos: gráfico e consola, a verdade é que com empenho e afincos por parte dos alunos, o nosso objetivo acabou por se concretizar.

Depreende-se assim, que o *Raspberry Pi* acabou por apoiá-los na sua própria aprendizagem, o que permitiu que ficassem com uma maior motivação para o estudo da programação, bem como permitiu a interação entre pares, a troca e a partilha de ideias, levando-os a expressarem-se e a construírem relacionamentos uns com os outros e mesmo com o próprio professor.

Mesmo no ensino diferenciado, obtivemos um *feedback* muito positivo, onde os alunos, através da aplicação *KidsRuby* e apesar de não existir uma versão em português, mostraram-se muito motivados e, principalmente, porque conseguiram aprender alguns conceitos básicos da linguagem de programação.

A questão emergente que envolve toda a comunidade que surgiu em torno do *Raspberry Pi* concretiza todas as semanas novos pacotes de *software*, novas distribuições e até periféricos adaptados e otimizados para este microcomputador. Neste cenário a reação dos alunos, apesar de inicialmente sentirem algumas dificuldades, não podia ser superior. Trabalharam com afincos na exploração da base aplicacional deste equipamento, consolidando os conhecimentos adquiridos nas disciplinas, transportando e adaptando ao nosso projeto.

### **Conclusão e Trabalho Futuro**

Programar requer pensamento crítico, habilidades matemáticas, habilidades de comunicação, resolução de problemas, criatividade e perseverança por parte dos alunos. Por isso, ao se iniciar o ensino da programação desde cedo, os alunos “passam a ter uma visão mais ampla do futuro”, passando a desenvolverem “a capacidade de investigar, explorar, aprendem a programar, elaboram *software* e têm uma noção do que é uma máquina programada”, como refere Jaime Rei, coordenador do Clube de Robótica.

Refere Gomes e Mendes (2007) que a programação “é uma arte”, na medida em que “existem muitas maneiras diferentes de codificar instruções, com alguma criatividade” mas também “uma ciência (...) porque é constituída por um conjunto de regras orientadoras, porque é necessário o uso de lógica e porque existem alguns métodos rigorosos de programação que asseguram a eficiência, economia e utilidade dos programas gerados” (p. 93).

Perante esta era digital em que nos encontramos, “a escola precisa de ser um espaço de desafio, de ousadia e de aventura” (Moreira, 2001, citado por Silva, 2009: 26) de forma aliciar esta nova geração de alunos.

Segundo Papert (1986), “a criança programa o computador. E, ao ensinar o computador a “pensar”, a criança embarca em uma exploração sobre a maneira como ela própria pensa. Pensar sobre modos de pensar faz a criança tornar-se um epistemólogo, uma experiência que poucos adultos tiveram” (citado por Brasão, 2007: s. p.). Daí que, de acordo com este autor, “o contacto com a linguagem computacional pode contribuir para atingir níveis de conhecimento complexos de uma forma natural” (Pinto, 2010: 3).

Indubitavelmente, consideramos que a introdução do *Raspberry Pi* na sala de aula contribuiu em muito para o processo mental do aluno, tal como proporciona um ambiente interativo e de

criatividade, onde o aluno desenvolve capacidades de interação com a realidade, formula e equaciona problemas. Como sustenta Correia (2005), “aprendemos melhor fazendo, mas aprendemos ainda melhor se além e fizermos, falarmos e pensarmos sobre o que fizemos” (citado por Brazão, 2007: s. p.).

Desta forma, como trabalho futuro, sugerimos a aplicação (e talvez uma melhoria na elaboração) do questionário que se encontra em anexo a este artigo, junto de alunos que frequentam cursos de informática, já que por limitação de tempo, o mesmo não foi alvo de estudo nesta investigação.

Este instrumento de recolha de dados, de acordo com Sousa e Baptista (2011) “permite recolher uma amostra dos conhecimentos, atitudes, valores e comportamentos” (p. 91). Por isso, parece-nos fundamental a aplicação do mesmo junto do nosso público-alvo de forma a obtermos informações sobre aspetos importantes relacionados com a exploração do *Raspberry Pi* em contexto educativo, nomeadamente no sucesso do ensino e aprendizagem de programação, de uma forma muito mais abrangente.

## Referências

- Bastos, R. L. (2009). *Ciências humanas e complexidades: projetos, métodos e técnicas de pesquisa: o caos, a nova ciência*. Rio de Janeiro: E-papers. <http://books.google.pt/books?id=cRMjRVUW4G0C&hl> (Acessível em 5 de maio de 2013).
- Bogdanchikov, A., Zhaparov, M., & Suliyev, R. (2013). Python to learn programming. *Journal of Physics: Conference Series* 423. [http://iopscience.iop.org/1742-6596/423/1/012027/pdf/1742-6596\\_423\\_1\\_012027.pdf](http://iopscience.iop.org/1742-6596/423/1/012027/pdf/1742-6596_423_1_012027.pdf) (Acessível em 7 de maio de 2013).
- Brasão, M. dos R. (2007). *Logo – uma linguagem de programação voltada para a educação*. <http://www.fucamp.edu.br/wp-content/uploads/2010/10/5%C2%BA-MAUR%C3%8DCIO-DOS-REIS-BRAS%C3%83O1.-LOGO.pdf> (Acessível em 7 de maio de 2013).
- Carvalho, A. A. A. (2006). Indicadores de qualidade de “sites” educativos. *Cadernos SACAUSEF – Sistema de Avaliação, Certificação e Apoio à Utilização de Software para a Educação e a Formação, Número 2, Ministério da Educação*, 55-78. <http://repositorium.sdum.uminho.pt/bitstream/1822/5922/1/Indicadores%20de%20Qualidade%20de%20Sites%20-SACAUSEF%20-AAC.pdf> (Acessível em 13 de maio de 2013).
- Catapan, A. H. (2003). *Pedagogia e tecnologia: a comunicação digital no processo pedagógico*. [http://www.portalnepsul.com.br/admin/uploads/2000/Educacao\\_e\\_formacao\\_de\\_professores/Mesa\\_Redonda\\_-\\_Trabalho/07\\_10\\_51\\_1M1003.pdf](http://www.portalnepsul.com.br/admin/uploads/2000/Educacao_e_formacao_de_professores/Mesa_Redonda_-_Trabalho/07_10_51_1M1003.pdf) (Acessível em 9 de maio de 2013).
- Costa, M. do C. D. da. (2010). *Comunicação na Internet: a rede social Myspace – um estudo de caso*. Dissertação de Mestrado. Mestrado de Relações Interculturais. Lisboa: Universidade Aberta. <https://repositorioaberto.uab.pt/handle/10400.2/1696> (Acessível em 24 de maio de 2013).
- Coutinho, C., & Lisboa, E. (2011). Sociedade da informação, do conhecimento e da aprendizagem: desafios para educação no século XXI. *Revista de Educação, Vol. XVIII, nº 1, (p. 5 – 22)*. [http://repositorium.sdum.uminho.pt/bitstream/1822/14854/1/Revista\\_Educa%C3%A7%C3%A3o%2cVolXVIII%2cn%C2%BA1\\_5-22.pdf](http://repositorium.sdum.uminho.pt/bitstream/1822/14854/1/Revista_Educa%C3%A7%C3%A3o%2cVolXVIII%2cn%C2%BA1_5-22.pdf) (Acessível em 11 de maio de 2013).
- Gomes, A. & Mendes, A. J. (2007). Learning to program - difficulties and solutions. *International Conference on Engineering Education ICEE*. <http://www.ineer.org/Events/ICEE2007/papers/411.pdf> (Acessível em 7 de maio de 2013).
- Gomes, A., Henriques, J., Mendes, A. J. (2008). Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. *Educação, Formação & Tecnologias, vol. 1 (1), Maio* <http://eft.educom.pt/index.php/eft/article/viewFile/23/16> (Acessível em 7 de maio de 2013).

- Halfacree, G. & Upton, E. (2012). *Raspberry Pi User Guide*. <http://www.google.pt/books?id=h4odLCArZAE&dq> (Acessível em 7 de maio de 2013).
- Horta, M. J., Mendonça, F., & Nascimento, R. (2012). *Metas curriculares: Tecnologias de Informação e Comunicação - 7.º e 8.º anos*. <http://www.google.pt/url?sa=t&rct=j&q=metas+curriculares+de+TIC> (Acessível em 7 de maio de 2013).
- Jain, A. K., Singhal, M. & Gupta, M. S., (2010). Educational Tool for Understanding Algorithm Building and Learning Programming Languages. *Proceedings of the World Congress on Engineering and Computer Science* Vol I WCECS. San Francisco, USA. [http://www.iaeng.org/publication/WCECS2010/WCECS2010\\_pp292-295.pdf](http://www.iaeng.org/publication/WCECS2010/WCECS2010_pp292-295.pdf) (Acessível em 8 de maio de 2013).
- Junior, E. R. D. F., Lignos, A., & Huertas, M. K. Z. (2013). Capítulo 1. Comunicações de marketing na internet: uma análise do uso de ferramentas dialógicas em web sites brasileiros. *Revista Inovação Tecnológica, São Paulo*, v.3, n. 1, p. 3-15. <http://www.faculadeflamingo.com.br/ojs/index.php/rit/article/view/63/81> (Acessível 12 de maio de 2013).
- Lage, S. (2012). *A escola “não pode ficar à margem” da ciência da computação*. <http://www.cienciahoje.pt/index.php?oid=56279&op=all> (Acessível em 6 de maio de 2013).
- Membrey, P., & Hows, D. (2012). *Learn Raspberry Pi with Linux*. Technology in Action. <http://books.google.pt/books?id=kbtCMBHYMP4C&printsec=frontcover&hl=pt-PT#v=onepage&q&f=false> (Acessível em 16 de maio).
- Merrelho, A. de A. (2010). *As atitudes dos alunos do 1º Ciclo do Ensino Básico face à utilização do computador Magalhães: estudo de caso*. Dissertação de Mestrado em Ciências da Educação. Área de Especialização em Tecnologia Educativa. Braga: Universidade do Minho. <http://repositorium.sdum.uminho.pt/bitstream/1822/14593/1/Anabela%20de%20Abreu%20Merrelho.pdf> (Acessível em 8 de maio de 2013).
- Miranda, L. C. de., Hornung, H. H., Solarte, D. S. M., Romani, R., Weinfurter, M. R., Neris, V. P. de A., & Baranauskas, M. C. C. (2007). Laptops Educacionais de Baixo Custo: Prospectos e Desafios. *XVIII Simpósio Brasileiro de Informática na Educação - SBIE*. <http://www.lbd.dcc.ufmg.br/colecoes/sbie/2007/0031.pdf> (Acessível em 7 de maio de 2013).
- Mitchell, G. (2012). The Raspberry Pi single-board computer will revolutionise computer science teaching. *Engineering & Technology, April*. <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=06210296> (Acessível em 7 de maio de 2013).
- Moreira, F. H. S. (2004). Yes, nós temos computador: Ideologia e formação de professores na era da informação. *Trab. Ling. Aplic., Campinas*, (43): 127-137. <http://www.iel.unicamp.br/revista/index.php/tla/article/viewFile/2221/1727> (Acessível em 6 de maio de 2013).
- Pinto, A. S. (2010). *Scratch na aprendizagem da Matemática no 1.º Ciclo do Ensino Básico: estudo de caso na resolução de problemas*. Dissertação de Mestrado em Área de Especialização em Estudos da Criança Tecnologias de Informação e Comunicação. <http://repositorium.sdum.uminho.pt/bitstream/1822/14538/1/tese.pdf> (Acessível em 7 de maio de 2013).
- Pires, S. M. B. (2009). As TIC no currículo escolar. *EDUSER: Revista de Educação, Vol 1(1)*. <https://www.eduser.ipb.pt/index.php/eduser/article/viewFile/3/1> (Acessível em 6 de maio de 2013).
- Prensky, M. (2008). *Programming Is the New Literacy*. <http://www.edutopia.org/programming-the-new-literacy> (Acessível em 6 de maio de 2013).
- Reis, L. G. (2008). *Produção de monografia: da teoria à prática*. Brasília: Senac. <http://books.google.pt/books?id=syG59k2nRogC&hl> (Acessível em 5 de maio de 2013)
- Robinson, J. (2011). *Eric Schmidt, chairman of Google, condemns British education system*. <http://www.guardian.co.uk/technology/2011/aug/26/eric-schmidt-chairman-google-education> (Acessível em 7 de maio de 2013).
- Santos, E. A., & Fermé, E. & Fernandes, E. (2007). Utilização de robots no ensino de programação: o projecto droide. *Actas do IX Congresso da SPCE “Educação para o sucesso: políticas e actores”*. Madeira. <http://nonio.fc.ul.pt/atms/learn/produtos/publicacoes/pdf/EF17.pdf> (Acessível em 6 de maio de 2013).



- Scaico, P. D., Lima, A. A. de., Silva, J. B. B. da., Azevedo, S., Paiva, L. F., Raposo, E. H. S., Alencar, Y., & Mendes, J. P. (2012). Programação no Ensino Médio: Uma Abordagem de Ensino Orientado ao *Design* com Scratch. *Anais do XVIII WIE Rio de Janeiro*. <http://br-ie.org/pub/index.php/wie/article/view/2112/1878> (Acessível em 5 de maio de 2013).
- Silva, L. P. P. da., (2009). *Squeak e aprofundamento de competências numéricas em crianças do 1.º ano de escolaridade*. Dissertação de Mestrado em Estudos da Criança - Área de Especialização em Tecnologias de Informação e Comunicação. <http://repositorium.sdum.uminho.pt/bitstream/1822/10973/1/Tese.pdf> (Acessível em 6 de maio de 2013).
- Solon, O. (2012). *Ready for the revolution? It's time for us all to rally behind the Raspberry*. <http://www.wired.co.uk/news/archive/2012-03/05/rally-behind-the-raspberry> (Acessível em 8 de maio de 2013).
- Sousa, M. J. & Baptista, C. S. (2011). *Como fazer investigação, dissertações, teses e relatórios segundo Bolonha*. Lisboa: Pactor.
- Teixeira, A. L. V. de S., (2011). *Integração das TIC na educação: o caso do Squeak Etoys*. Tese de Doutoramento em Estudos da Criança Especialidade de Tecnologias de Informação e Comunicação. [http://repositorium.sdum.uminho.pt/bitstream/1822/14206/1/TeseDout\\_LuisValente\\_2011.pdf](http://repositorium.sdum.uminho.pt/bitstream/1822/14206/1/TeseDout_LuisValente_2011.pdf) (Acessível em 7 de maio de 2013).
- Valente, L., & Osório, A. J. (2006). *Programar uma Corrida com o Squeak*. Instituto de Estudos da Criança. Universidade do Minho. [http://www.valente.org.pt/downloads/artigos/workshop\\_squeak\\_ji\\_ciane.pdf](http://www.valente.org.pt/downloads/artigos/workshop_squeak_ji_ciane.pdf) (Acessível em 7 de maio de 2013).