

UM ESTUDO SOBRE DESENVOLVIMENTO DE APLICAÇÕES PARA ENGENHARIA EM TABLETS COM ANDROID

Julie Anne Braun¹
Eluane Cristine Costa Pires¹
Thiago Moraes Santos²
Vinícius Rodrigo May³
Humberto Azevedo Nigri⁴

Resumo: O presente artigo tem por objetivo apresentar os estudos realizados sobre a plataforma Android[®] na criação de aplicações em áreas da Engenharia para uso em *Tablets*. Esses dispositivos abrem novas possibilidades de uso da tecnologia na Engenharia e facilitam o exercício da profissão, fornecendo novas ferramentas para cálculos, consultas ou uso de aplicativos especializados. O estudo buscou aprender sobre o processo de criação de aplicações para *tablets* com Android[®] e concluiu-se que, apesar do surgimento de novas ferramentas, que podem facilitar o seu uso, o processo de criação para certos tipos aplicativos mais voltados para Engenharia ainda exige um grande conhecimento em programação orientada a objetos, Java[®] e outras ferramentas específicas

Palavras Chave: Android. Desenvolvimento. Aplicações. Tablets

Abstract: This paper has the intent to present studies made on Android[®] platform in creation of Engineering applications for use in tablets. This appliances open new possibilities of technology use in Engineering field and make easier professional's work, adding new tools for calculus, queries or use of specialized applications. The study focused on learning process of applications for *tablets* with Android[®] plattaform and it would be concluded that besides the new building tools available that make this work easy, application creation process still demands a great knowledge in object-oriented programming, Java[®] and other specifics tools.

Key Words: Android, Development, Applications, Tablets

INTRODUÇÃO

O *Android* foi criado no início de 2005 como um sistema operacional móvel para telefones e, posteriormente, tornou-se o principal competidor do *iPhone*, da *Apple*. Seu grande trunfo foi usar uma plataforma aberta e flexível, que permitiu a sua customização e atraiu diversos parceiros e fabricantes de aparelhos *smartphones* (os celulares com funcionalidades avançadas), reforçando e alavancando o projeto deste sistema operacional tão popular atualmente (LECHETA, 2010, 22-). O crescimento do uso de aparelhos celulares em todo o mundo já supera a marca de 3 bilhões de pessoas (LECHETA, 2010, 19) e o *Android* é hoje um dos sistemas mais utilizados em *smartphones* e já supera os 50% só no mercado norte-americano (SHACKLES, 2012, 16).

O surgimento dos *tablets* deu-se, inicialmente, com o *iPad* da *Apple*. Os concorrentes, visualizando o sucesso obtido e aproveitando as possibilidades oferecidas pelo *Android* como plataforma, passaram a utilizá-lo também nesses dispositivos. A Google, então, evoluiu o *Android* para a versão 3.0, também chamada de *Honeycomb* (LECHETA, 2012, 17) e, posteriormente, lançando novas versões como a 4.0, 4.1 e 4.2. Com isso, vem mudando também o perfil dos consumidores que encontram novos usos para aplicações móveis de forma prática. Para o profissional de engenharia, seja no trabalho em campo ou na indústria, essa ferramenta pode representar um diferencial de produtividade, além da praticidade de acesso a informações e aplicações que o auxiliem em sua prática profissional. Os usuários destes aparelhos, apesar de contar com uma grande oferta de aplicativos, ainda dependem que empresas e programadores independentes criem e lancem aplicativos de seu interesse. No entanto, para o engenheiro, um uso interessante desta tecnologia poderia ser a possibilidade de transformar o seu conhecimento e *know-how* técnico em aplicativos que o auxiliem na sua atividade prática. Embora já venham surgindo aplicativos criados para engenheiros como em asme, 2013, a gama de soluções que poderiam ser criadas pelo próprio profissional poderia ajudá-lo muito mais, desde que este consiga dominar o processo de criação de seus próprios aplicativos.

Este estudo foi motivado pela questão: quais as facilidades e dificuldades que um engenheiro, ou estudante de engenharia, encontra no desenvolvimento de pequenas aplicações voltadas para a prática diária da engenharia? Este trabalho foi realizado com quatro estudantes de cursos de engenharia, sendo dois de engenharia civil e dois dos cursos de engenharia elétrica e controle e automação do Centro Universitário Newton Paiva, em Belo Horizonte. Focamos também na busca de informações e publicações

no idioma português, uma vez que o inglês ainda é uma barreira para muitos alunos e profissionais da área. Felizmente, constatamos que já existem diversos títulos publicados no Brasil, principalmente, nos três últimos anos, o que, de certa forma, ajuda no acesso ao aprendizado deste desenvolvimento.

ESTRUTURA DE UMA APLICAÇÃO

ANDROID

O *Android* é uma a plataforma de código aberto para dispositivos móveis da Google. Para o desenvolvimento de aplicações para esta plataforma, foi criado o *Android SDK (Software Development Kit)* que é o kit de desenvolvimento que disponibiliza as ferramentas e APIs (*Application Program Interfaces*) necessárias para desenvolver aplicações para a plataforma *Android*, utilizando a linguagem Java. Dentre os seus principais recursos, destacamos: a *Application framework* que proporciona a reutilização e substituição de componentes de código, dentro da filosofia da programação orientada a objetos; a *Dalvik Virtual Machine* que é uma Máquina Virtual criada e otimizada para dispositivos móveis; um *Browser* (navegador internet) Integrado, baseado no *webkit engine*; Gráficos Otimizados, possuindo uma biblioteca para gráficos 2D e 3D baseada na especificação *OpenGL ES 1.0* (um padrão de linguagem para criação de softwares gráficos); *SQLite* que permite o armazenamento de dados estruturados, funcionando como um pequeno bancos de dados; Suporte multimídia para áudio, vídeo e formatos de imagem (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF); Telefonia GSM (dependente do hardware); Suporte a comunicação sem fio: *Bluetooth*, EDGE, 3G, e WiFi; Câmera, GPS, compasso, e acelerômetro, todos dispositivos disponíveis no *hardware* dos *smartphones* atuais; possui ainda um rico ambiente de desenvolvimento, incluindo um emulador de dispositivo (permite simular diversos aparelhos para testar as aplicações), ferramentas de depuração, memória e performance. Utiliza por padrão o ambiente integrado (IDE) de desenvolvimento Eclipse, que integra as funções de edição de código, compilação e execução das aplicações desenvolvidas. É necessária a instalação de um *plugin* para o Eclipse chamado ADT (*Android Development Tools*). Esse pacote adiciona funcionalidades importantes ao ambiente, permitindo a criação de telas de forma mais visual reduzindo a necessidade de se codificar os arquivos em XML (*Extended Markup Language*) para a definição dos *layouts* de tela (MEDNIELKS, 2012). Na figura 1, mostramos um exemplo de um dos projetos utilizando o IDE Eclipse com o ADT em uso na criação de uma das telas da aplicação.

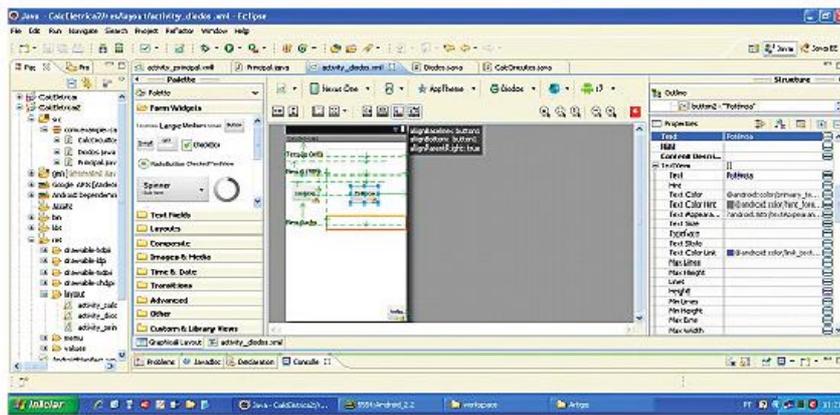


Figura 1 - Ambiente de Desenvolvimento para Android com Eclipse.

Fonte: Tela capturada pelos autores, maio/2013

Uma aplicação *Android*, criada a partir de um código em Java, é toda organizada em classes. Apesar de utilizarem a linguagem Java como linguagem nativa, as aplicações *Android* não permitem que um programa Java já desenvolvido para outra plataforma consiga ser compilado e executado em seu ambiente. A seguir, vamos descrever alguns dos elementos essenciais de uma aplicação *Android*. O arquivo principal de uma aplicação *Android* é o chamado *Android Manifest*. Esse arquivo existe no formato XML e é o responsável pelas definições gerais da aplicação, tais como o seu nome, versão compatível, qual a sua tela principal, dentre outras.

Todas as telas de uma aplicação no *Android* precisam ser criadas a partir de uma classe chamada *Activity* (LECHETA, 2010, 93). Esta classe é a responsável por gerenciar a execução da aplicação e suas subtelas. Dentro do ADT no Eclipse, cada *activity* é criada a partir de um menu de opções disponível para o programador. Uma *activity*, no *Android*, é toda e qualquer aplicação possível de se executar. Isso se aplica tanto às aplicações já disponíveis no sistema quanto às novas aplicações desenvolvidas pelo programador. As *activities* são também especializadas em diversos outros tipos de classes para usos mais específicos tais como: *ListActivity* (para uso com listas de itens), *MapActivity* (para se trabalhar com mapas), dentre outras.

Dentro de uma *activity*, são definidas outro tipo de classes chamadas Views. As Views são as classes responsáveis pela visualização dos seus elementos tais como: textos, imagens, botões, campos de edição, etc. As views são, por sua vez, divididas em dois grandes grupos: um deles conhecidos como *Layouts* e o outro como *views* de objetos visuais. Os *Layouts* têm como tarefa organizar as demais *views*, controlando a sua localização e desenho na tela. Os vários *Layouts* disponíveis tais como: *LinearLayout*, *RelativeLayout*, *TableLayout*, dentre outros, serão utilizados de acordo com a necessidade de visualização dos elementos de uma aplicação.

Nas diversas *views* de uma aplicação *Android*, são necessários diversos elementos, que podem ser textos, imagens, sons. Cada elemento desse, precisa definir diversas propriedades tais como: tamanho (dimensões), localização, conteúdo, nome (um identificador para ser reconhecido no programa), etc. Essas definições podem ser separadas e mantidas em arquivos de configuração separados, o que facilita o seu gerenciamento e eventual substituição – por exemplo, na tradução de um aplicativo para outro idioma. Esses elementos são chamados de Recursos (ou *Resources*). São mantidos em pastas separadas dentro do projeto da aplicação e, posteriormente, integradas a esta durante a compilação e geração do aplicativo final.

Para se executar uma *activity*, é necessário se criar uma *Intent* (LECHETA, 2010, 135). Essa classe é a responsável informar ao Android que uma *activity* irá ser executada. Ela representa a “intensão” de se executar uma determinada tarefa e é quem carrega as informações necessárias para se saber sob que condições essa tarefa irá ser executada. Por exemplo, podemos chamar uma *activity* de mapas a partir de um contato da lista telefones para localizar o endereço de um determinado contato, ou solicitar ao sistema que faça uma ligação para um número de celular. Todas essas ações precisam de uma *intent* que as descrevam para que o sistema as execute.

Outro recurso importante das aplicações *Android* são os *layouts* XML disponíveis para a criação de telas. Os elementos visuais de uma aplicação *Android* tais como: imagens, textos, botões, etc., precisam ser criados por comandos específicos durante a execução de uma aplicação. Da forma tradicional que se faz em Java, são chamados os métodos de criação de cada elemento visual e cada campo deste, que descreve seu tamanho, cor, comportamento, etc., deve ser preenchido via códigos em Java. O ADT oferece uma outra alternativa que facilita esse processo, tanto em termos de esforço de codificação, quanto clareza e organização do código. São os *layouts* em XML. Esses ar-

quívos contêm dados relativos aos elementos visuais e são processados de forma automática durante a criação das classes de *layouts* já citadas. A grande vantagem de utilizar essa forma de trabalho é que grande parte dos códigos Java referentes à criação da parte visual da aplicação são retirados das fontes da aplicação, ficando apenas códigos relativos às funcionalidades da aplicação, tais como cálculos, regras e ações realizadas pelo programa.

Além destes, estão disponíveis uma enorme variedade de recursos nas bibliotecas do *Android* que permitem o controle de elementos do *hardware* do aparelho, tais como o acelerômetro, funções do GPS, Câmera, Telefone GSM, acesso e navegação *Web*, Mapas, além das mídias de áudio e vídeo. Durante o trabalho de pesquisa realizado alguns desses elementos foram estudados objetivando a construção de aplicações simples, mas o potencial existente permite a criação de aplicativos bastante sofisticados que podem compor soluções muito úteis tanto para os usuários comuns como para os profissionais da engenharia.

METODOLOGIA

Utilizou-se para esse trabalho tanto a pesquisa bibliográfica e teórica, quanto o trabalho de experimentação prática. Através da busca de informações em livros, artigos, publicações, sites e comunidades na internet, procurou-se conhecer mais sobre a plataforma *Android* e as ferramentas necessárias para o desenvolvimento de aplicações para a mesma. Algumas alternativas de desenvolvimento foram investigadas e seus limites de atuação estudados. Foram estudados o *AppInventor* (APPINVENTOR, 2012), e o *Mono Framework* (XAMARIN, 2012), sendo estas ferramentas alternativas para criação de aplicações para o *Android*.

O *AppInventor*, atualmente, é um projeto sediado no MIT (*Massachusetts Institute of Technology*). Foi criado originalmente na Google, com o objetivo de facilitar a criação de aplicativos por pessoas leigas sem um conhecimento prévio de programação ou algoritmos. Ele se baseia em interfaces totalmente visuais e figuras como blocos de construção que se encaixam e vão montando as sequências lógicas de uma aplicação ou ativam determinados recursos como sons, vídeos, sites, mapas, etc. Nas fases iniciais da pesquisa, foram estudados alguns dos recursos disponíveis no site, porém, à época, o site se encontrava em fase de transição, passando do domínio da Google para o MIT e, por isso, algumas funcionalidades não puderam ser totalmente avaliadas. Independentemente dessas dificuldades, decidiu-se também investigar outras possibilidades, pois as aplicações típicas que são desenvolvidas no *AppInventor* são mais voltadas para o uso pessoal e em aplicativos de divertimento dos usuários finais. Em aplicações de en-

genharia, espera-se poder desenvolver cálculos, equações, estruturas de dados como vetores e matrizes que possam apresentar soluções para um engenheiro de forma prática e objetiva, dentro de um uso mais profissional de aplicações. Neste ramo de aplicações, percebeu-se que a ferramenta não é totalmente aplicável.

Já o *Mono Framework* (XAMARIN, 2012) pareceu ser uma proposta bastante interessante. Além de contar com uma comunidade de desenvolvedores, e com publicações em português (SHACKLES, 2012), possui um atrativo a mais: a possibilidade de se criar aplicativos que executem não somente no *Android*, mas também no *iPad/iPhone*, e *smartphones* com o *Windows Mobile* (SHACKLES, 2012, 17-23). Neste caso, a linguagem base para criação deixa de ser o Java e passa a ser a linguagem C# (pronuncia-se C "sharp").

C# é a linguagem criada e lançada em 2000, pela *Microsoft Corporation*, para o desenvolvimento de aplicações portáteis dentro da plataforma chamada .NET (pronuncia-se *dot net*). Essa plataforma veio para concorrer com a plataforma Java de desenvolvimento de aplicações locais (em computadores fora da internet), aplicações *web*, e móveis (*smartphones* e outros dispositivos). Porém, com a existência de diversas plataformas distintas de aplicações móveis o *Windows Mobile* tornou-se apenas mais uma plataforma na disputa acirrada da preferência do consumidor final.

O grande mérito do *Mono Framework* é o de proporcionar ao programador a facilidade de poder criar as suas aplicações em apenas um ambiente – o utilizado para isso o C#, e poder executar a mesma aplicação em todas as demais plataformas. A avaliação dessa ferramenta, porém, ficou prejudicada, pois as versões que permitem o desenvolvimento pleno de soluções é paga, o que limitou o seu uso pelo por esse projeto e as versões gratuitas testadas apresentaram erros seguindo-se os tutoriais indicados. Mas de qualquer forma, acredita-se que, quando essa ferramenta tiver um uso mais difundido pela comunidade, poderá evoluir bem e trazer uma grande facilidade aos seus usuários programadores.

Montou-se também um ambiente de desenvolvimento utilizando-se o ferramental proposto pelo próprio fabricante que é a linguagem Java, o IDE Eclipse, e o SDK (*software development kit*) do *Android* (ANDROID, 2013). Foram estudados também alguns conceitos de programação orientada a objetos e eventos.

RESULTADOS E DISCUSSÕES

Com o objetivo de vivenciar o processo de criação de aplicativos para a plataforma *Android* foram definidas duas aplicações simples a serem criadas. O objetivo foi

o de servirem como um protótipo do que pode vir a ser construído utilizando as ferramentas disponíveis para a plataforma *Android*, e em especial para uso nos *tablets*. No contexto desse projeto, os alunos bolsistas e voluntários não tinham uma grande vivência em programação, apenas uma disciplina de Algoritmos cursada como parte da grade básica dos cursos de engenharia da Newton Paiva. Tais limitações implicaram maior dificuldade no aprendizado de conceitos mais avançados em programação, como o paradigma da orientação a objetos, essencial no desenvolvimento em Java. Outras dificuldades percebidas ao longo do trabalho com essa plataforma foram:

Dificuldade na instalação e configuração do ambiente de desenvolvimento – embora os roteiros e tutoriais tenham sido obtidos com facilidade por diversas fontes, nem sempre os mesmos funcionavam adequadamente e como as versões das ferramentas mudam constantemente, nem todos os tutoriais estão atualizados, o que gerou alguns erros e retrabalhos durante essa fase.

Configurações das estações de desenvolvimento – o ambiente de desenvolvimento, composto pelo IDE Eclipse, o SDK *Android*, Java e demais ferramentas exigem uma configuração melhor nas estações que visam o desenvolvimento destas aplicações. Alguns dos computadores utilizados se mostraram bastantes lentos para sua utilização nesse trabalho. Outra percepção também diz respeito ao monitor de vídeo. O Eclipse, por abrir uma grande quantidade de janelas, cada uma delas contendo informações importantes do projeto tais como: código fonte, telas, *layouts*, mensagens, propriedades, estruturas de pastas, dentre outras; necessita de estações com monitores maiores sendo recomendados tamanhos acima de 17 ou 20 polegadas.

Dificuldades com o desenhador ADT. O ADT (*Android Development Tools*) possui um desenhador visual que permite se clicar e arrastar para a tela elementos que se deseja para compor o visual da aplicação, tais como:

Textos, Botões, Campos de Edição, etc. Essa montagem, porém, na tela, exige um certo esforço, pois nem sempre os elementos ficam da forma como se deseja, e no Layout Relativo, que foi o mais utilizado nos testes, a movimentação posterior dos elementos visuais alterava a localização de outros na tela. Tais dificuldades ocorrem porque uma aplicação *Android*, por ser preparada para executar em diferentes dispositivos (*smartphones* e *tablets*), com diferentes tamanhos de tela, precisa se basear não em coordenadas fixas de *pixels* (pontos gráficos na tela), mas em relação ao tamanho disponível na tela ou em relação à posição de outros elementos. Nestes casos, a melhor estratégia utilizada foi a de planejar primeiro qual o *layout* desejado para cada tela e, posteriormente, posicionar os elementos visuais em uma certa ordem, evitando-se a sua posterior movimentação.

Ao final deste trabalho, foram criadas duas calculadoras simples: uma Calculadora de Custo de Construção e outra de Cálculo de Circuitos Elétricos. Nas figuras 2 e 3 a seguir, tem-se algumas das telas das duas aplicações criadas. Estas aplicações apresentam menus de opções para os tipos de cálculos e parâmetros desejados e telas de entrada de dados para a realização dos cálculos. O objetivo na sua concepção foi o de facilitar a operação por parte do engenheiro, realizando cálculos simples, evitando a necessidade de se fazer contas em uma calculadora convencional. Apesar de simples, as aplicações construídas exigem o domínio de conceitos da plataforma *Android* já explicados tais como: *Activity*, *Intent*, *Layout*, *View* e outros. O tempo dedicado à programação propriamente foi de cerca de dois meses, entre final de janeiro e início de março, após os demais estudos estarem concluídos. Por não haver disponibilidade maior de tempo e devido à complexidade do processo de programação, optou-se pela criação de aplicações mais simples.

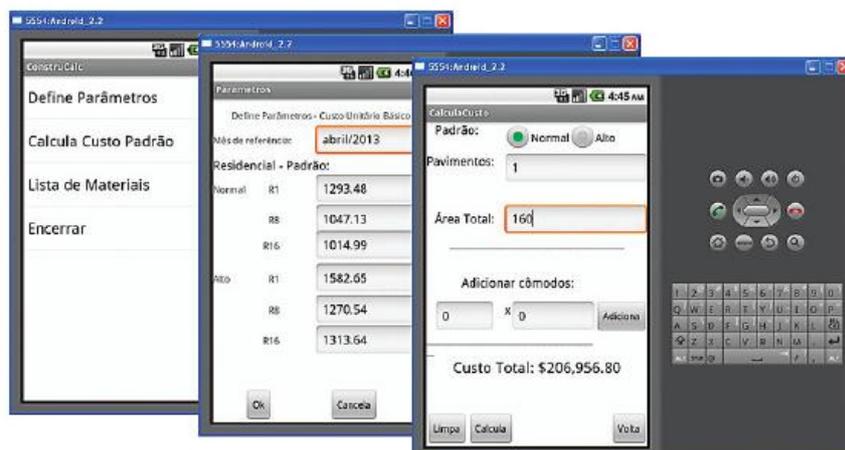


Figura 2 – Telas da Calculadora de Custos de Construção.

Fonte: Tela capturada pelos autores, maio/2013



Figura 3 – Telas da Calculadora de Circuitos Elétricos.

Fonte: Tela capturada pelos autores, maio/2013

CONSIDERAÇÕES FINAIS

Este trabalho de pesquisa teve como objetivo uma avaliação prática e qualitativa do esforço necessário para a criação na plataforma *Android* de aplicações para engenharia, por engenheiros ou estudantes da área. Foi objeto de estudo entender o processo necessário para se criar tais aplicações, as dificuldades e soluções existentes para tanto. Com esse trabalho, concluímos que o desenvolvimento de aplicações para a plataforma *Android* é possível e viável. No entanto, esse aprendizado ainda exige um considerável esforço e dedicação por parte do engenheiro para dominar conceitos mais avançados de programação e elementos essenciais à plataforma *Android*. Em um mundo competitivo como o atual, a diferenciação e qualificação em diversas áreas torna-se um diferencial para o profissional moderno, e o engenheiro, pelo seu contato permanente com a tecnologia, precisa saber usá-la como ferramenta de apoio às suas necessidades. Assim como uma calculadora científica ou qualquer outro instrumento é utilizado pelo engenheiro para obter os resultados desejados, o computador e, atualmente, o *tablet*, também devem ser encarados sob essa ótica. Diferentemente de um programador ou analista de sistemas, cuja atividade fim é o programa ou o sistema em si, o engenheiro necessita da programação para elaborar e aplicar os programas, construídos com base em sua experiência, para facilitar a execução de tarefas repetitivas e frequentes do seu cotidiano profissional. Dominar conceitos mais avançados como a programação orientada a objetos, a linguagem Java e outros assuntos correlatos pode representar um esforço considerável para o engenheiro em um primeiro momento, porém, esse conhecimento está cada vez mais acessível. Encontrase disponível uma gama enorme de fontes bibliográficas, boas publicações, mesmo no idioma português, sites e tutoriais (gratuitos ou pagos) e diversos cursos que podem auxiliar nessa tarefa.

Foi observado também que já existem alternativas acessíveis para a criação de pequenas aplicações que não exijam grandes estruturas de dados ou cálculos complexos. Programas que fazem uma interação com outras mídias,

como vídeos, mapas, sites e outros recursos também já vêm sendo bastante facilitada pelas novas ferramentas de desenvolvimento. Tais ferramentas têm trazido cada vez mais aos profissionais que não são programadores, a habilidade de poder construir suas próprias aplicações com base em conceitos simples e intuitivos, e a tendência é que num futuro próximo tal capacidade esteja ao alcance de muitos que hoje se beneficiam da tecnologia móvel e das comunicações em geral.

Acreditamos, finalmente, que o domínio por parte da engenharia desse ferramental poderá ser de grande valia para os atuais e futuros engenheiros em sua prática profissional.

REFERÊNCIAS

- ANDROID, Google. **Site oficial dos desenvolvedores do Android**. Disponível em: <http://developer.android.com/index.html>. Acesso em maio/2012 a abr/2013.
- APPINVENTOR. **Site do projeto App Inventor**. Disponível em: <http://appinventor.mit.edu/explore/>. Acesso em ago/2012.
- ASME. **5 Mobile Apps for Engineers**. Disponível em: <http://www.asme.org/kb/news-articles/articles/technology-and-society/mobile-apps-for-engineers-what-s-in-store>. Acesso em fev/2013.
- LECHETA, Ricardo Rodrigues. **Google Android**. 2ª Edição. São Paulo: Editora Novatec, 2010.
- LECHETA, Ricardo Rodrigues. **Google Android para Tablets**. 2ª Edição. São Paulo: Editora Novatec, 2012.
- MEDNIELKS, Zigurd, et al. **Programando o Android**. 1ª Edição. São Paulo: Editora O'Reilly/Novatec, 2012.
- SHACKLES, Greg. **Construindo Aplicativos Móveis com C#**. 1ª Edição. São Paulo: Editora O'Reilly/Novatec, 2012.
- XAMARIN. **Developer Center**. Disponível em: <http://docs.xamarin.com/guides/android>. Acesso em set/2012 e abr/2013.

NOTAS

- 1-Discentes do curso de Engenharia Civil, alunos de Iniciação Científica do Centro Universitário Newton Paiva.
- 2-Discente do curso de Engenharia Elétrica, aluno de Iniciação Científica do Centro Universitário Newton Paiva.
- 3-Discente do curso de Engenharia de Controle e Automação, aluno de Iniciação Científica do Centro Universitário Newton Paiva.
- 4-Coordenador da pesquisa, Docente do Centro Universitário Newton Paiva.