

COMPARANDO A EFICIÊNCIA ENERGÉTICA DE APLICATIVOS NATIVOS À DE APLICATIVOS WEB

JEAN ROBERTO ANTUNES¹; JONATHAN GONCZOROSKI¹; MILENA R. S. MARQUES¹, PAULO R. FERREIRA JÚNIOR¹, LISANE BRISOLARA¹

¹*Universidade Federal de Pelotas – {jrantunes, jmgonczorosk, mrsmarques, paulo, lisane}@inf.ufpel.edu.br*

1. INTRODUÇÃO

Pesquisas recentes mostram um aumento crescente no volume de vendas de dispositivos móveis, como smartphones e tablets. Este mercado atraiu empresas como Google e Apple, que proveem respectivamente os sistemas operacionais (SOs) Android e iOS, dominantes neste mercado (IDC, 2015).

Cada um destes SOs utiliza uma plataforma própria de desenvolvimento, provendo APIs específicas a linguagem empregada (WASSERMAN, 2010). Os aplicativos criados para uma plataforma específica são denominados aplicativos nativos. Estes aplicativos aproveitam melhor os recursos do dispositivo, porém são difíceis de portar para outras plataformas devido às diferenças nas APIs e linguagens empregadas, exigindo assim a total reescrita do aplicativo.

Com a grande diversidade de dispositivos e plataformas existentes no mercado, se fez necessária a criação de aplicativos multiplataforma, isto é, aplicativos que possam ser executados em diferentes dispositivos com diferentes sistemas operacionais, sem a necessidade de que se reescreva totalmente o código. Esta portabilidade diminui significativamente o esforço no desenvolvimento.

Neste contexto, uma estratégia para evitar o desenvolvimento nativo e facilitar a portabilidade de aplicativos móveis vem se destacando, na qual emprega-se tecnologias da área de desenvolvimento Web (SERRANO et al., 2013). O processamento destas aplicações Web podem ocorrer localmente ou remotamente, sendo o último executado em um servidor que não depende de recursos do dispositivo.

Aplicações Web, quando implementadas remotamente, necessitam comunicação com o servidor, o que pode ocasionar impacto negativo em seu desempenho. Em contrapartida, no que se refere ao consumo energético, os aplicativos nativos, por serem executados localmente, tendem a consumir mais energia da bateria quando comparados às implementações baseadas em tecnologias Web.

Estudos comparativos entre aplicativos nativos e baseados em Web podem ser encontrados em (CORRAL et al., 2012) e (GRANLUND et al., 2013), porém não se encontram trabalhos que considerem a eficiência energética em seus comparativos. Este artigo tem por objetivo comparar aplicativos desenvolvidos usando tecnologias Web com aplicativos nativos, avaliando o impacto desta decisão no consumo energético destes quando executados em dispositivos Android.

O presente artigo está organizado da seguinte maneira: a seção 2 discute a metodologia adotada; a seção 3 apresenta o estudo comparativo entre aplicativos nativos e Web, apresentando e discutindo seus resultados; e por fim, a seção 4 apresenta as conclusões e trabalhos futuros.

2. METODOLOGIA

Visando comparar eficiência energética de aplicativos nativos em relação a aplicativos Web, foram desenvolvidas implementações do algoritmo de ordenamento Bubblesort, o qual possui uma complexidade $O(n^2)$. Estas implementações geram um vetor para ordenação no pior caso, com elementos na ordem decrescente, para posterior ordenamento. O aplicativo nativo foi implementado em Java e o aplicativo Web foi implementado em PHP. Um aplicativo Java foi implementado para fazer a invocação do código PHP que é executado em um servidor web, fazendo com que a diferença entre os dois aplicativos seja transparente para o usuário. Cabe ressaltar que no aplicativo Web, o vetor é criado no servidor, não requerendo o envio do mesmo do dispositivo para o servidor.

Para avaliar consumo energético dos aplicativos foi utilizado o aplicativo PowerTutor (ZHANG et al., 2012), que estima, em Joules, o consumo da aplicação no dispositivo Android. Para todas as execuções dos experimentos realizados o aplicativo PowerTutor é ativado e, após, a aplicação em análise é executada. Após o ordenamento concluído, o consumo é determinado. Todos os experimentos foram realizados utilizando um smartphone Samsung Galaxy s4 mini (modelo GT-I9192) com o Android versão 4.4.2.

Nos experimentos o tamanho do vetor a ser ordenado foi variado entre 100, 1000, 10000 e 100000 elementos. Todos os testes foram repetidos trinta vezes para validade estatística dos resultados. Após as execuções, é calculada a média e desvio padrão dos valores obtidos e as médias são então comparadas. O teste estatístico t é aplicado para verificar se existe diferença significativa entre as médias comparadas. Para vetores de 100000 elementos não foram realizados experimentos com o aplicativo Web, já que esta execução excede os 30s de *timeout* padrão dos servidores Web.

3. RESULTADOS E DISCUSSÃO

Os resultados obtidos para o consumo médio dos aplicativos, bem como o desvio-padrão entre as diferentes execuções, são resumidos na Tabela 1. O emprego do aplicativo Web pode reduzir o consumo energético do dispositivo, transferindo a carga de processamento para um servidor. No entanto, para uma carga pequena de processamento (ordenamento de 100 e 1000), os experimentos indicam que é mais eficiente energeticamente realizar o processamento localmente. Já no caso do ordenamento do vetor com 10000 elementos, a implementação Web passa a ser mais eficiente que a versão nativa, representando uma redução do consumo energético de 68% nos experimentos realizados.

Os resultados de consumo médio para os vetores de ambas as implementações também são ilustrados na Figura 1. Nestes experimentos, o aplicativo nativo apresenta um ganho significativo de eficiência energética em relação ao aplicativo Web quando se trata de vetores menores (100 e 1000 elementos). No entanto, quando vetores maiores são empregados, o aplicativo Web apresenta o menor consumo.

Tabela 1. Resultados de Consumo Energético

Tamanho	Implementação	Consumo médio (J)	Desvio Padrão
100	Nativo	0,0089	0,0039
	Web	0,1402	0,0954
1000	Nativo	0,0358	0,0451
	Web	0,2160	0,1074
10000	Nativo	4,4700	0,4991
	Web	1,4321	1,5706
100000	Nativo	144,71	10,4437
	Web	-	-

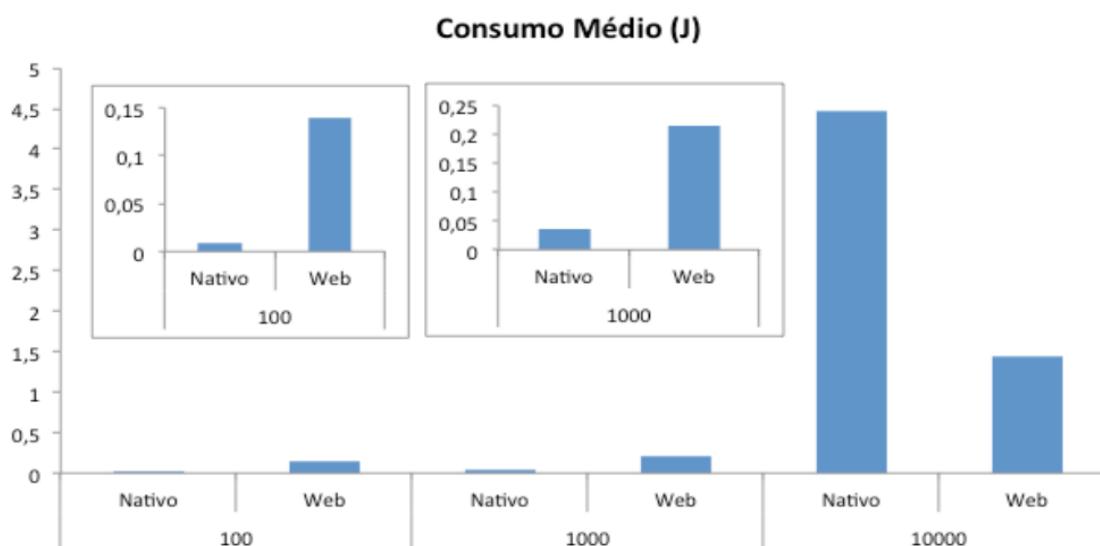


Figura 1. Comparativo do Consumo Médio

4. CONCLUSÕES

Aplicativos móveis podem ser implementados utilizando a abordagem nativa ou utilizando tecnologias Web. Este trabalho apresenta um comparativo entre essas duas abordagens considerando o impacto no consumo da bateria do aparelho. Nos experimentos, algoritmos de ordenação baseados no algoritmo Bubblesort foram avaliados, variando a sua carga entre 100, 1000, 10000 e 100000 elementos.

Os resultados apontaram que a abordagem nativa é eficiente no consumo para cargas pequenas, como por exemplo 100 e 1000. Porém, aumentando a carga para 10000 elementos, a abordagem utilizando tecnologias Web se mostrou mais eficiente. Cabe destacar que os aplicativos empregados nos experimentos não fazem o envio do vetor para o servidor web, o que poderia causar um aumento no consumo do aplicativo web para que essa comunicação se realize.

Como trabalhos futuros, pretende-se modificar os aplicativos e refazer os experimentos considerando que o vetor deverá ser passado para o servidor para então ser ordenado. Além disso, como neste trabalho somente aplicativos desenvolvidos para a plataforma Android foram avaliados, pretende-se avaliar também o impacto do emprego de aplicativos Web para a plataforma iOS.

5. REFERÊNCIAS BIBLIOGRÁFICAS

IDC. Smarthphone OS Market Share. Acessado em 25 jun. de 2015. Disponível em: <http://idc.com/prodserv/smarthos-market-share.jsp>

Wasserman, A. I. Software engineering issues for mobile application development, **ACM**, p.397-400, 2010

SERRANO, N.; HERNANTES, J.; GALLARDO, G. Mobile Web Apps, **IEEE Software**, v.30, n. 5, p. 22-27, Sept.-Oct. 2013.

CORRAL, L.; SILLITI, A.; SUCCI, G. Mobile multiplatform development: An experiment for performance analysis. **Procedia Computer Science**, v. 10, p. 736-743, 2012.

GRANLUND, D. ; JOHANSSON, D.; ANDERSSON, K. ; BRANNSTROM, R. A Case Study of Application Development for Mobile and Location-Based Service. In **INTERNATIONAL CONFERENCE ON INFORMATION INTEGRATION AND WEB-BASED APPLICATIONS & SERVICES**. Viena, 2013, *Anais...* New York, ACM, p. 658, 2013

ZHANG, L.; TIWANA, B.; DICK, R. P.; ZHIYUN, Q.; MAO, Z, M.; ZHAOGUANG, W.; LEI, Y. Accurate online power estimation and automatic battery behavior based power model generation for smartphone, In **Proceedings of the Eighth IEEE/ACM/IFIP International Conference on Hardware/software codesign and system synthesis (CODES/ISSS)**, p.105-114, 2010