

Uma Proposta de Ferramenta de Apoio ao Ensino Prático de Computação Ubíqua

Maria Joelma Pereira
Peixoto
Universidade Federal do
Ceará (UFC)
Fortaleza – CE – Brasil
joelmapereira@great.ufc.br

Windson Viana de
Carvalho
Universidade Federal do
Ceará (UFC)
Fortaleza – CE – Brasil
windson@great.ufc.br

Wellington Wagner
Ferreira Sarmento
Universidade Federal do
Ceará (UFC)
Fortaleza – CE – Brasil
wwagner@virtual.ufc.br

ABSTRACT

Study of Ubiquitous Computing concepts began in various universities, in courses related to Computer Science, which sought to develop the study of ubiquitous systems and technologies. In general, these courses have practical classes or even small ubiquitous application development projects. However, one of the greatest difficulties in teaching Ubiquitous computing is putting into practice the theoretical knowledge acquired since the existing tools already require high-level programming skills or are not designed for educational purposes. As a result, this proposal for master's thesis aims to design, develop and evaluate a tool that helps to teach practical classes involving concepts of Ubiquitous Computing.

Keywords

Ubiquitous Computing; Learning Environment; Mobile Laboratories.

1. CARACTERIZAÇÃO DO PROBLEMA

O progresso tecnológico contribuiu para o barateamento e a disseminação de diversos dispositivos computacionais (e.g. *laptops*, *smartphones*, sensores) e também de uma variedade de tecnologias de comunicação. Esses avanços proporcionaram a concepção e a forte adoção de serviços computacionais em muitos domínios, tais como agricultura, transporte, medicina e educação.

Esse progresso tecnológico aproxima o ser humano, de uma certa forma, da visão de computação preconizada por Mark Weiser. Em suas pesquisas, que tiveram início em 1988, Weiser imaginava o futuro no qual a tecnologia estaria tão inserida no cotidiano das pessoas que ela seria indistinguível no ambiente, misturando-se com objetos do dia a dia [12]. Em “*The Computer for the 21st Century*”, Weiser introduziu o termo *Ubiquitous Computing (UbiComp)*, no português Computação Ubíqua, no qual os sistemas computacionais estariam “embutidos” em elementos da vida diária, sendo capazes de trocar dados entre si com a menor intervenção possível de usuários [12].

In: Workshop de Teses de Dissertações (WTD), 16., 2016, Teresina. Anais do XXII Simpósio Brasileiro de Sistemas Multimídia e Web. Porto Alegre: Sociedade Brasileira de Computação, 2016. v. 2.

ISBN: 978-85-7669-332-1

©SBC – Sociedade Brasileira de Computação

No entanto, a utilização de ferramentas de apoio ao ensino é um dos maiores desafios para a prática de conceitos de UbiComp em sala de aula. Isso porque estas são muito escassas, fugazes e, às vezes, de difícil acesso. Os altos custos do *hardware* utilizado, por exemplo, como o LEGO Mindstorms [10], caracteriza uma das dificuldades encontradas.

Embora existam na literatura diversos relatos de experiências no ensino de conceitos de UbiComp, tais como [9], [3], [10], [2], em geral, as ferramentas ou abordagens utilizadas incluem artefatos ou softwares que não foram concebidos especificamente para o ensino de UbiComp. Como exemplo, podem ser citados o Node-RED [11], voltado para Internet das Coisas, e o App Inventor [8], voltado para o ensino de lógica de programação, ou mesmo ambientes de programação tradicionais de aplicações móveis (e.g. Android Studio).

É nesse contexto que esta pesquisa busca melhorar o ensino-aprendizagem de conceitos de UbiComp por meio da concepção, desenvolvimento e avaliação de uma ferramenta que fará uso de *smartphones* e seus sensores. A escolha destes artefatos ocorre em virtude deles serem mais acessíveis financeiramente e mais utilizados pela maioria dos estudantes. Essa ferramenta consistirá de duas partes: uma aplicação Android, que terá acesso aos sensores do dispositivo e simulará conceitos de Computação Ubíqua; e uma aplicação Web, na qual as configurações da simulação serão definidas tanto pelo professor quanto pelo aluno.

O restante deste documento está organizado em cinco sessões que estão descritas a seguir. A Seção 2 trata da fundamentação teórica e dos trabalhos relacionados, abordando as definições de UbiComp, seus conceitos e características, o ensino de UbiComp e uma revisão bibliográfica sobre os trabalhos correlacionados com o tema. A Seção 3 aborda a proposta deste trabalho, assim como as contribuições esperadas. As Seções 4, 5 e 6 tratam, respectivamente, do estado atual da pesquisa, dos resultados preliminares obtidos e das considerações finais.

2. FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS

2.1 Computação Ubíqua

No final da década de 80, Mark Weiser cunhou o termo Computação Ubíqua, que definia um novo paradigma da computação. Tinha-se, então, o vislumbre de um futuro, no qual as tecnologias de computação se tornariam totalmente integradas aos objetos, podendo ser utilizadas para auxiliar nas atividades cotidianas, tanto no trabalho como

na diversão [6]. Weiser considerava que as tecnologias mais profundas são aquelas que desaparecem. Elas se misturam com os objetos do dia a dia até que se tornam indistinguíveis no ambiente [12].

Com os recursos de *hardware* (e.g. *smartphones*, *tablets*, sensores e outros equipamentos com alto poder de comunicação e processamento) e de *software* disponíveis atualmente, tornou-se possível a realização da UbiComp [4]. Esta se concretiza através de sistemas computacionais que possuem componentes de *hardware* e de *software* integrados para a realização de tarefas, por meio da execução coordenada de atividades, a qual inclui troca de informações, invocações de serviços e compartilhamento de recursos [7]. Pode-se, então, experimentar as idealizações feitas por Weiser e que não eram possíveis em seu tempo, tais como relógios inteligentes que medem os batimentos cardíacos, *smart TVs*, que podem ser conectadas a Internet e controladas por *smartphones*, sistemas de recomendação e espaços multimídia e interativos.

A UbiComp é fruto da evolução de alguns outros paradigmas que tiveram início com a computação de *mainframe*. Nesta primeira fase, os computadores de grande porte ocupavam o cenário da época com grande poder de processamento de informação. Posteriormente, veio a fase da computação pessoal, conhecida pelos computadores de pequeno porte e de baixo custo, destinados exclusivamente ao uso pessoal ou a um pequeno grupo de indivíduos. Em seguida, tem-se a era da computação distribuída. Esta se caracterizou pela dispersão de componentes de *hardware* e de *software* que se comunicam como um único sistema. Então, na fase atual, há a concretização da UbiComp.

A Computação Ubíqua se difere da Computação Pervasiva no que se refere a manutenção da conexão e configuração de serviços necessários mesmo durante a mobilidade do usuário. O conceito de pervasividade diz respeito a embarcação de computadores no ambiente de forma invisível ao usuário e sem garantia de muita mobilidade [1]. Apesar dessas diferenças, a definição dos dois conceitos se aproxima bastante, o que leva este trabalho a considerar Computação Ubíqua e Pervasiva como sinônimas.

2.1.1 Principais Conceitos e Características

Com base em [7] e em [4], após observados alguns projetos de sistemas ubíquos, foram levantadas características consideradas fundamentais para a descrição dos conceitos que determinam a UbiComp. Por exemplo, pode-se citar a heterogeneidade, que é a capacidade do sistema suportar uma grande variedade de dispositivos e serviços sem comprometer seu funcionamento. Outro exemplo é a sensibilidade ao contexto, que é a característica que permite um sistema inferir as necessidades do usuário sem que seja necessário determiná-las explicitamente, conforme o contexto em que o usuário está inserido.

2.2 A Computação Ubíqua em Sala de Aula

Conceitos de UbiComp são vistos em diferentes disciplinas nas universidades. Pode-se citar Redes Móveis Sem Fio, Sistemas Distribuídos, Interação Humano-Computador, Internet das Coisas, Sistemas Multimídias ou mesmo aquelas dedicadas exclusivamente ao ensino de UbiComp. Chalmers *et al.*, por exemplo, descrevem um cenário de organização de disciplinas que possuem conceitos e práticas de UbiComp na Universidade de Sussex, no Reino Unido. Essas aulas estão organizadas em três partes: exposição teórica, seminários e

práticas em laboratório [2]. Na exposição teórica, são apresentados os conceitos fundamentais de UbiComp e trabalhos que abrangem, de forma mais generalizada, o estado da arte relacionado ao tema. No momento da apresentação de seminários, o foco é analisar criticamente as teorias e as soluções existentes para resolução de problemas relacionados aos conceitos ubíquos. Nas práticas em laboratório, o objetivo é programar variados dispositivos, para justificar o entendimento do que foi visto na teoria. O foco é a concepção, a execução e a análise de experimentos, com a posterior apresentação dos resultados obtidos.

Nos cursos de Computação e Sistemas e Mídias Digitais da Universidade Federal do Ceará (UFC), as disciplinas relacionadas à UbiComp possuem estruturas similares a apresentada por Chalmers [2], com as práticas sendo realizadas na maior parte do tempo em Android e, eventualmente, em PhoneGap¹.

2.3 Trabalhos Relacionados

Segundo [2], em algumas universidades, o interesse em determinadas plataformas e em linguagens de programação, assim como o conhecimento prévio dos estudantes, podem levar a diferentes escolhas para a utilização de artefatos que apoiem as aulas práticas. Por exemplo, o professor pode optar pelo uso de *Kinect* e outros dispositivos de jogos para aqueles que trabalham com jogos dentro do currículo acadêmico.

A Tabela 1 apresenta um conjunto de trabalhos relacionados que aplicam ferramentas no ensino de Computação Ubíqua e Pervasiva. Alguns dos projetos apresentados trabalham com *hardware* e *software* inicialmente concebidos para outras finalidades. Por exemplo, os trabalhos utilizam kits SenseBoard, ferramentas Web, LEGO Mindstorms NXT, MATLAB e Phidgets. Esses artefatos não abordam os conceitos de UbiComp diretamente, o que exige mais empenho dos professores para a elaboração de atividades práticas.

O trabalho de mestrado proposto diferencia-se por conceber uma ferramenta para o ensino de conceitos de UbiComp cujo intuito é concentrar uma série de atividades expositivas, interativas e práticas em um mesmo ambiente. Essas atividades ilustram os principais conceitos da disciplina, minimizam o esforço do professor na elaboração das atividades práticas e podem ajudar os estudantes na compreensão dos conceitos teóricos do paradigma proposto por Mark Weiser.

3. CONTRIBUIÇÕES ESPERADAS

O objetivo desta proposta é contribuir para a seguinte questão de pesquisa: Quais características devem existir em uma ferramenta de apoio ao ensino de UbiComp de forma a melhorar o entendimento dos conceitos teóricos por parte dos estudantes?

Este trabalho visa conceber, desenvolver e avaliar um ambiente Web, com acesso a uma aplicação móvel Android, que possa auxiliar professores e alunos na prática de conceitos

¹Para saber mais sobre PhoneGap, acesse: <http://phonegap.com/>

²Inviabilidade de venda do SenseBoard: <http://www.ouw.co.uk/dp/178007395X>

³Para *download* da linguagem Sense, acesse: <http://www.open.edu/openlearnworks/mod/oucontent/view.php?id=22780%3F>

⁴Para saber mais sobre Phidgets, acesse: <http://www.phidgets.com/>

Tabela 1: Trabalhos Relacionados

	SenseBoard [9]	PIP [3]	Pervasive Computing [10]	Pervasive Computing Classroom [2]
Tipo	<i>Desktop e hardware</i>	<i>Desktop</i>	<i>Desktop e hardware</i>	<i>Hardware</i>
Público-Alvo	Graduação - EAD	Sem restrições	Graduação	Pós-graduação
Nível do Público	Sem exigências	Sem exigências	Sem exigências	Conhecimentos de programação
Objetivo	Explorar UbiComp por meio de experimentos práticos	Ensinar conceitos de programação	Ensinar UbiComp através de projetos	Ensinar UbiComp por meio da prática em laboratório
Recursos Exigidos	Kit SenseBoard e Linguagem Sense	Ferramenta instalada	LEGO MINDSTORMS NXT e MATLAB	Phidgets
Custos	Médio/alto	Baixo	Alto	Médio/alto
Disponibilidade	Kit não disponível para compra ² ; Linguagem disponível ³	Material não disponibilizado e não encontrado	Sim	Sim ⁴

teóricos de UbiComp. Conforme a Figura 1, essa ferramenta vai, num primeiro instante, focar em algumas características de UbiComp, tais como heterogeneidade, sensibilidade ao contexto, adaptabilidade e mobilidade. Para cada um destes conceitos, haverá seções com informações teóricas, exemplos de implementações, simulações das camadas de um sistema ubíquo e práticas de programação relacionadas a cada conceito estudado.

A ideia é que, para cada conceito estudado (e.g. heterogeneidade e sensibilidade ao contexto), sejam criados Objetos de Aprendizagem (OA) que exemplifiquem o funcionamento desses conceitos. Os alunos poderão visualizar o fluxo da interação entre as camadas que compõem os sistemas ubíquos, enxergando como funcionam a comunicação e o compartilhamento de informações entre sistemas, ambientes e pessoas. Espera-se facilitar o entendimento dos conceitos ubíquos com a ferramenta proposta, de forma a estimular a criação de aplicações ubíquas, pois ainda não foi encontrada nenhuma ferramenta com características e finalidades semelhantes. O ambiente proposto estará dividido conforme a organização de curso apresentada por [2] e também utilizada na UFC, nos cursos de Computação e Sistemas e Mídias Digitais. Assim, haverá momentos de exemplificações teóricas, trabalhos ou implementações relacionadas ao conceito estudado e atividades práticas. Pretende-se com isso estimular uma maior discussão em torno dos assuntos abordados pela ferramenta.

4. ESTADO ATUAL DO TRABALHO

4.1 Metodologia

A metodologia utilizada neste trabalho é a do *Design Centrado no Usuário* (DCU), na qual o desenvolvedor não só imagina como será a interação do usuário final com o sistema que será criado, mas também testa essa mesma aplicação com usuários reais para dar validade ao trabalho. Segundo

[13], o DCU é a prática de criar produtos de forma que os usuários sejam capazes de utilizá-los com o mínimo de *stress* e o máximo de eficiência.

4.2 Pesquisas com o Público-Alvo

4.2.1 Professores

Foram realizadas quatro entrevistas, três com professores da Universidade Federal do Ceará - UFC e uma com uma professora brasileira lotada na *Université Paris 1 Panthéon-Sorbonne*. O objetivo das entrevistas foi identificar a experiência do professor, a organização do curso, os materiais usados em aulas práticas, o número de práticas, as dificuldades relacionadas ao ensino de conceitos de UbiComp, os conceitos considerados mais importantes pelos professores e como eles imaginam um ambiente para a prática de UbiComp. Dentre as questões apresentadas no questionário-guia⁵, apenas as de maior relevância para a análise dos dados estão discriminadas a seguir.

- Você utiliza algum ambiente/kit para ensinar UbiComp?
 - Professor 1 = Android Studio
 - Professor 2 = Não usa um ambiente específico
 - Professor 3 = Android Studio
 - Professor 4 = Não utiliza
- Quais as dificuldades que você encontra no ensino dos conceitos de UbiComp?
 - Professor 1 = Muito conteúdo para pouco tempo e aplicação da teoria na prática
 - Professor 2 = Aplicar teoria na prática
 - Professor 3 = Complexidade dos conceitos de UbiComp, dificuldade de ver como os conceitos teóricos são aplicados na prática

⁵Questionário guia para entrevista com professores: <https://goo.gl/forms/MHdyKuCtnBk3whx1>

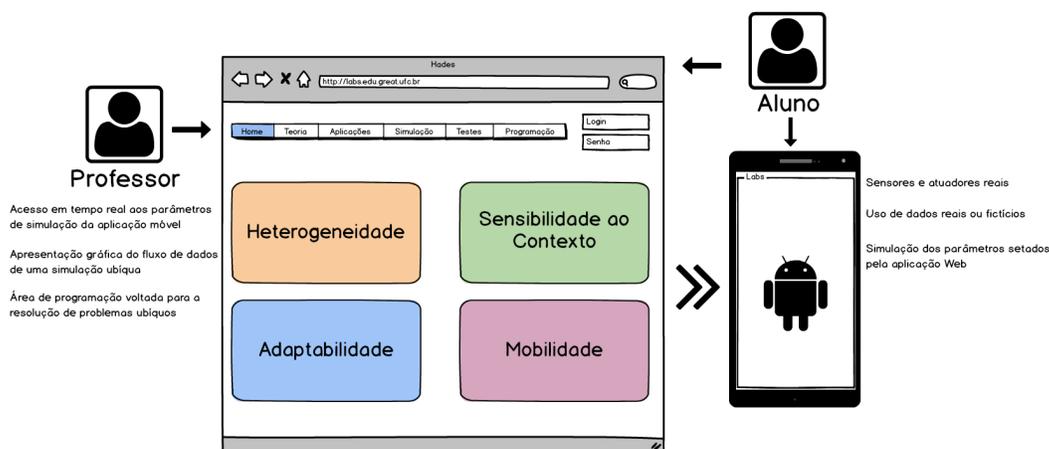


Figura 1: Visão geral do ambiente Web e móvel.

- Professor 4 = Muito conteúdo para pouco tempo
3. O que o ambiente interativo deveria ter?
- Professor 1 = Comportamento de informações contextuais e teoria
 - Professor 2 = Simulação do comportamento de dispositivos e atuação de sensores
 - Professor 3 = Conceitos de sensibilidade ao contexto, como modelar informações contextuais importantes para a aplicação, adaptação da aplicação conforme as informações de contexto
 - Professor 4 = Vídeos, tutoriais que expliquem o funcionamento de ferramentas já existentes (e.g. *arduino* ou *raspberry*) e OA's que incrementem a aprendizagem.
4. Qual o conceito de UbiComp você considera como o mais importante para o ambiente interativo?
- Todos os professores elegeram a Sensibilidade ao Contexto.

Com os resultados iniciais, 50% dos professores dessa primeira coleta utilizam o Android Studio como ambiente para o ensino de UbiComp. As três últimas perguntas são extremamente relevantes para esta pesquisa. Elas apontam as maiores dificuldades que os professores têm em sala de aula, o que eles imaginam que o ambiente proposto por este trabalho tenha e o conceito considerado por eles como o de maior relevância para o ensino de UbiComp.

4.2.2 Estudantes

O questionário⁶ aplicado com os estudantes é composto por onze questões. Ele foi disponibilizado de forma *online*, via grupos de disciplinas no Facebook que tinham relação direta com UbiComp ou que eram exclusivamente sobre o assunto. O objetivo do questionário foi saber dos estudantes se eles possuíam experiência de programação no momento da realização da disciplina, se tinham aulas práticas de UbiComp, se usavam alguma ferramenta para as aulas práticas, quais os conceitos que eles mais tinham dificuldade de entender e o que eles esperavam do ambiente proposto. Depois da disponibilização dos questionários em grupos de disciplinas

⁶Questionário para pesquisas com estudantes:
<https://goo.gl/forms/f7j2IUeCtOFmpBP73>

no Facebook, obteve-se o total de 30 respostas. Os alunos, em geral, eram do curso de Computação ou de Sistemas e Mídias Digitais - UFC. Uma análise mais detalhada de algumas das principais respostas pode ser vista a seguir.

Os resultados dos questionários indicam que a maioria dos estudantes tiveram aulas práticas dedicadas à UbiComp. Quando perguntado o que eles utilizavam nessas práticas, a maioria respondeu que usava o Android Studio como ambiente. Uma outra pergunta foi relacionada aos conceitos mais difíceis de se entender na ausência de aulas práticas. A maioria dos estudantes respondeu que o conceito de “*middleware*” foi o mais difícil de se entender, seguido por “*adaptação*” e “*sensibilidade ao contexto*”, conforme ilustra a Figura 2.

Quando perguntados sobre o que deveria ter no ambiente proposto por este trabalho, a maioria respondeu que, como mostrado na Figura 3, deveria haver “*simulação do comportamento de sensores e serviços a partir da inserção de parâmetros*”. Na sequência, falaram de “*programação de códigos Android, pré-moldados pelo ambiente, para visualização, em dispositivos móveis, dos conceitos ubíquos*” e “*representação gráfica dos conceitos teóricos*”. Na área para sugestões e observações dos alunos sobre as aulas práticas no ensino de UbiComp, a maioria ressaltou a importância delas. Falaram também que elas deveriam abordar as características da ubiquidade dentro do funcionamento dos sistemas, o que é necessário se fazer, por exemplo, para que um sistema seja adaptável e heterogêneo.

4.3 Primeiro Protótipo

A partir destes primeiros requisitos levantados nas entrevistas com professores e alunos, foi decidida a implementação de um primeiro protótipo que seguisse a arquitetura inicial da ferramenta (i.e., dividida em duas partes: Web e Móvel) e permitisse uma atividade prática com os estudantes. O tema escolhido foi o de sensibilidade ao contexto. O objeto de aprendizagem faz uma analogia com as camadas de um sistema sensível ao contexto do Context Toolkit proposto por Dey [5].

Segundo essa proposta de Dey, conforme a Figura 4, as abstrações realizadas para a construção dessa arquitetura facilitam o entendimento de conceitos para a construção de aplicações sensíveis ao contexto. Conforme essa arquitetura, a camada mais básica é a de sensores, que mostra a disponi-

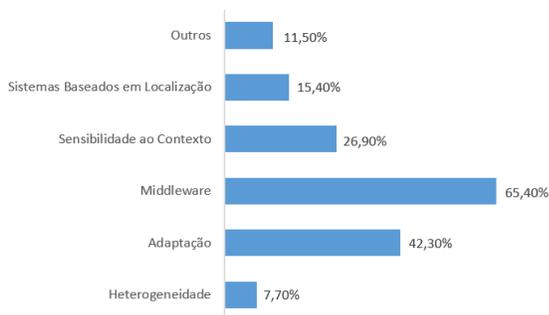


Figura 2: Conceitos difíceis de se entender na ausência de aulas práticas por estudantes

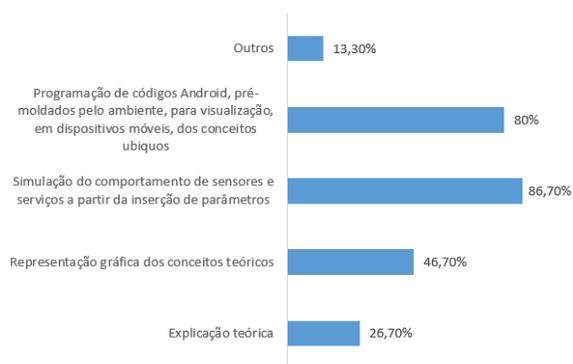


Figura 3: Características do ambiente segundo os estudantes

bilidade de sensores que uma aplicação pode acessar. Mais acima, está a camada de *widgets*, que encapsula uma informação do contexto do usuário, como local ou uma atividade, e gera uma interface para aplicação, escondendo os detalhes de implementação. Depois, tem-se a camada de agregadores, que mescla informações contextuais, como localização e luminosidade, por exemplo, para gerar uma nova informação contextual, como nível de segurança do local. Por último, tem-se os interpretadores, que interpretam informações de contexto de baixo nível, transformando-as em informações de nível superior.

No ambiente proposto, o fluxo dos dados pode ser observado por meio da ativação das camadas, que são habilitadas conforme a camada mais inferior vai sendo ativada pelo aluno ou pelo professor. O ambiente móvel foi criado em versão Android. Nele observa-se o comportamento de um sistema sensível ao contexto conforme as configurações feitas no ambiente Web pelo professor ou pelos alunos. O sistema móvel é inicialmente uma tela em branco, que vai sendo preenchida conforme as configurações dos usuários na ferramenta Web. Assim, se o sensor de luminosidade for iniciado dentro da camada de *widgets* no sistema Web, conforme mostra a Figura 5, valores em lúmens aparecem no sistema móvel, conforme a Figura 6, de acordo com a variação de intensidade da luz do ambiente real captado pelo sensor do celular.

A representação da Figura 5 se assemelha a arquitetura do Context Toolkit proposto por Dey [5]. Isso porque sim-

plifica, para os alunos, a apresentação do fluxo de dados que uma aplicação sensível ao contexto possui no decorrer de cada camada. Na representação proposta por este trabalho, tem-se o acréscimo de mais uma camada, a de tomada de decisões, onde um atuador do celular (e.g. lanterna) pode ser ativado ou desativado conforme a captura de informações configuradas pelo usuário do ambiente Web.

Conforme pode ser visualizado na transição da Figura 6A para Figura 6B, a aplicação móvel só apresenta as informações configuradas pelo usuário no ambiente Web à medida que elas vão sendo adquiridas do seu contexto. A Figura 6A traz, por exemplo, uma espécie de *logcat* no topo para mostrar quais tipos de sensores estão ativos na aplicação e um valor de luminosidade obtido a partir do sensor do próprio dispositivo. A Figura 6B mostra o resultado de outras configurações realizadas.

O primeiro protótipo da ferramenta foi feito a partir da junção de algumas tecnologias. Primeiro foi criado um servidor Web em Ruby que se comunica, via *WebSocket*, com dois tipos diferentes de clientes: um Web e outro Android. No cliente Web foram utilizadas tecnologias como HTML e JavaScript. No cliente Android utilizou-se java. Vale ressaltar que vários clientes Androids podem se conectar e receber as configurações de um mesmo cliente Web. Desta forma, o professor pode controlar o comportamento das aplicações móveis de todos os estudantes e assim, ilustrar a execução de um determinado conceito de UbiComp.

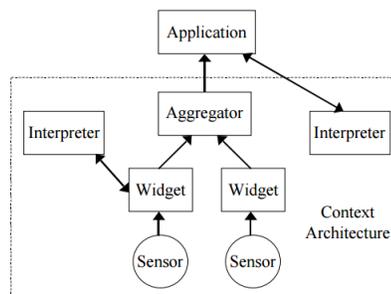


Figura 4: Componentes da arquitetura de contexto proposta por Dey [5].

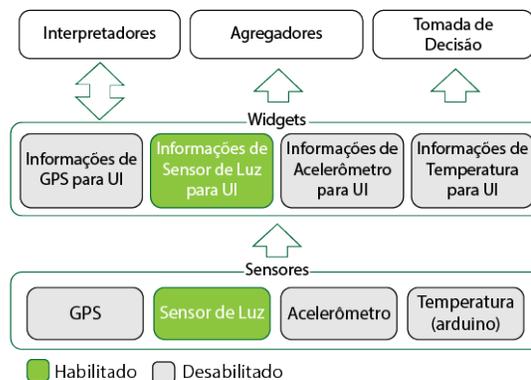


Figura 5: Objeto de Aprendizagem sobre Sensibilidade ao Contexto

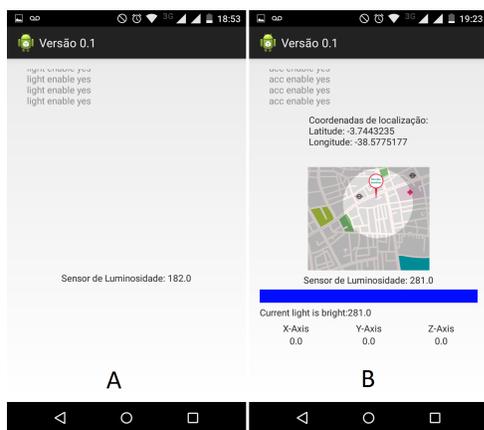


Figura 6: Reflexo da execução no dispositivo móvel das configurações realizadas no ambiente Web: (A) menos sensores ativos e (B) mais sensores ativos.

5. DESCRIÇÃO E AVALIAÇÃO DOS RESULTADOS

5.1 Teste do Primeiro Protótipo

O teste do primeiro protótipo foi realizado na disciplina de Introdução a Computação Móvel e Ubíqua, do Curso de Sistemas e Mídias Digitais - UFC. A prática foi feita guiada por um questionário⁷ que explicava o que devia ser feito e pedia a apresentação e interpretação dos resultados obtidos através do sistema. A sala foi dividida em dois trios e três duplas, que responderam também, ao final da prática, a um questionário⁸ de avaliação dos ambientes.

Como exemplo do passo a passo realizado na atividade prática, os estudantes foram solicitados a habilitar o sensor do acelerômetro e descrever o que seu *widget* e interpretador realizavam. Deviam apresentar também *screenshots* do que era mostrado na tela do dispositivo móvel, conforme suas configurações na aplicação Web. Deviam ainda testar os parâmetros de sensibilidade do interpretador do acelerômetro e descrever a funcionalidade deles.

Todos os alunos conseguiram concluir a atividade e responder a avaliação sobre o ambiente. Então, 62,5% disseram que concordam que a ferramenta avaliada apresenta diferentes caminhos para solucionar um determinado problema. 37,5% disseram que concordam totalmente que a proposta instiga a procura de outras informações em diferentes fontes de pesquisa. Foi gratificante perceber o envolvimento dos estudantes com a ferramenta e suas discussões em torno do assunto abordado por ela. Isso pode ser justificado por 62,5% terem respondido que concordam totalmente que houve a promoção do debate sobre os tópicos trabalhados com outros alunos ou com o próprio professor. 62,5% disseram que concordam que foram apresentados os conteúdos de maneira apropriada, podendo adequar sua utilização ao nível de conhecimento de cada aprendiz.

6. CONSIDERAÇÕES FINAIS

⁷Questionário guia do teste prático realizado com estudantes: bit.ly/praticaCompUbiqua

⁸Questionário de avaliação da ferramenta criada: <https://goo.gl/forms/RYLEvCSacXv9IVF52>

Este documento apresenta como foco a proposta preliminar da concepção, desenvolvimento e avaliação de um ambiente para auxiliar professores e alunos em aulas práticas de Computação Ubíqua. Como já descrito, foi possível verificar, através do levantamento bibliográfico inicial, que existem alguns trabalhos tratando sobre a temática de materiais para o auxílio em aulas práticas de Computação Ubíqua. Porém, eles se utilizam de componentes ou ferramentas com outras finalidades base, o que não supre a maior parte das necessidades das práticas em sala de aula.

O estado atual do trabalho foi discutido. Os primeiros resultados obtidos foram expostos e avaliados a fim de se nortear os próximos passos. A escrita da dissertação deverá ser concluída até março de 2017 e defendida no mesmo mês.

Pretende-se continuar com as entrevistas e questionários aplicados com professores e alunos. O primeiro protótipo também será aperfeiçoado para a realização de novas avaliações em sala de aula. Além disso, mais propostas de conteúdos para a ferramenta serão concebidas e implementadas.

7. REFERÊNCIAS

- [1] R. B. Araujo. Computação ubíqua: princípios, tecnologias e desafios. XXI Simpósio Brasileiro de Redes de Computadores, 2003.
- [2] D. Chalmers. Pervasive computing as a classroom-based course. *Pervasive computing - IEEE*, pages pp. 70–78, 2015.
- [3] J. Chin and V. Callaghan. Educational living labs: A novel internet-of-things based approach to teaching and research. *9th International Conference on Intelligent Environments, IEEE*, pages pp. 92–99, 2013.
- [4] C. A. B. de Queiroz Filho. Um mecanismo de tratamento de exceções sensível ao contexto para sistemas ubíquos orientados a tarefas. Dissertação de mestrado, Universidade Federal do Ceará, Fortaleza, Ceará, Brasil., 2012.
- [5] A. K. Dey and G. D. Abowd. The context toolkit: Aiding the development of context-aware applications. *Workshop on Software Engineering for Wearable and Pervasive Computing*, 2000.
- [6] J. Krumm. *Ubiquitous Computing Fundamentals*. Redmond, Washington. U.S.A.: CRC Press, 2010.
- [7] F. F. Lima. Syssu - um sistema de suporte para computação ubíqua. Dissertação de mestrado, Universidade Federal do Ceará, Fortaleza, Ceará, Brasil, 2011.
- [8] MIT. Mit app inventor.
- [9] M. Richards, M. Petre, and A. Bandara. Starting with ubicomp: Using the senseboard to introduce computing. *SIGCSE'12*, pages pp. 583–588, 2012.
- [10] N. Silvis-Cividjian. Teaching pervasive computing to cs freshmen: A multidisciplinary approach. *ITiCSE '15*, pages pp. 195–200, 2015.
- [11] I. E. Technologies. Node-red.
- [12] M. Weiser. The computer for the 21st century. *Scientific American*, 265(3):66–75, Sept. 1991.
- [13] W. E. Woodson, B. Tilman, and P. Tilman. Human factors design handbook, 1981.