

Um Sistema para Automação Residencial Sensível ao Contexto

Eduardo Germano da Silva¹, Tatiana Nilson dos Santos², Cristian Cleder Machado¹,
Luis Augusto Dias Knob¹, Anderson Luiz Fernandes Perez²

¹Universidade Federal do Rio Grande do Sul - UFRGS
Porto Alegre - Rio Grande do Sul, Brasil

²Universidade Federal de Santa Catarina - UFSC
Araranguá - Santa Catarina, Brasil

{eduardo.germano, ccmachado, luis.knob}@inf.ufrgs.br,
tatiana.nilson@gmail.com, anderson.perez@ufsc.br

Abstract. *This paper describes the development of a context-sensitive home automation system. Our system monitors environments of a residence, configuring devices of a particular room. Further, our application moves the computational context of an environment to other, considering the positioning and mobility of user inside the residence. Thus, our solution adapts computational contexts according with the users' preference. Our solution allows that users fully control the residence, through local and remote communication interfaces. Finally, we elaborate a prototype based in hardware of reduced dimensions, allowing that the system runs pervasively in environment.*

Resumo. *Este artigo descreve o desenvolvimento de um sistema de automação residencial sensível ao contexto. Nosso sistema monitora os ambientes de uma residência, configurando os dispositivos de um determinado cômodo. Além disso, nossa aplicação move o contexto computacional de um ambiente para outro, considerando o posicionamento e a mobilidade do usuário dentro da residência. Deste modo, nossa solução adapta contextos computacionais de acordo com as preferências dos usuários. Nossa solução permite ao usuário o controle total da residência, através de interfaces de comunicação local e remota. Por fim, elaboramos um protótipo baseado em hardware com dimensões reduzidas, permitindo que o sistema execute de modo pervasivo no ambiente.*

1. Introdução

O rápido avanço tecnológico dos últimos anos permitiu que a computação evoluísse, de modo que dispositivos computacionais fossem reduzidos de tamanho e ao mesmo tempo otimizassem suas capacidades de processamento. Este avanço tecnológico permite o desenvolvimento de soluções mais sofisticadas e precisas para auxiliar nos mais diversos propósitos, desde a comunicação entre equipamentos, tomada de decisões organizacionais, na automação de processos, ou até mesmo na automação de residências. Mais especificamente no contexto de automação residencial, o benefício deste avanço tecnológico se torna interessante, pois esta área da automação faz uso de mecanismos computadorizados para realizar de forma automática as tarefas que originalmente são executadas por usuários [Muratori and Bó 2011], tais como acionamento e configuração de eletrodomésticos, abrir e fechar de portas e janelas, entre outras.

A automação residencial traz benefícios para o cotidiano dos usuários em aspectos como: segurança, com uso de sistemas de segurança residencial; acessibilidade, permitindo que usuários portadores de necessidades especiais levem uma vida mais independente; conveniência, fornecendo mais conforto e serviços para os usuários; melhor eficiência energética, já que a automação residencial permite gerenciar e monitorar o consumo energético de equipamentos eletrônicos. Tais benefícios fazem com que a automação residencial seja uma tendência e um diferencial em novos empreendimentos imobiliários. Além disso, atualmente a automação residencial é alvo de estudos em diversos centros de pesquisa do mundo inteiro, que visam melhorar esse ramo da automação [Muratori and Bó 2011].

A automação residencial tradicional executa os processos de forma reativa, onde a automação só é realizada quando o usuário aciona um determinado comando. Entretanto, com o emprego de periféricos como sensores e câmeras, torna-se possível monitorar e mapear os hábitos, características e necessidades dos usuários de uma residência. Tais informações permitem o desenvolvimento de aplicações que utilizam as características do ambiente proativamente a favor de um usuário. Este tipo de sistema é denominado aplicação sensível ao contexto. Esta categoria de aplicações aumentam o grau de acessibilidade de uma residência, pois permitem que o ambiente seja uma agente proativo, que reconhece quem reside em seu interior e atua de modo inteligente, se antecipando as diversas situações que podem ocorrer, aplicando soluções que beneficiam seus usuários [Jiang et al. 2004].

Residências que empregam aplicações sensíveis ao contexto são denominadas *smart-homes*. *Smart-homes* não visam somente ampliar o conforto e a segurança de seus usuários, mas sim a qualidade de vida de modo geral. Essa característica, aliada com o aumento da expectativa de vida das pessoas, faz com que as *smart-homes* sejam ícones importantes em cenários contemporâneos e futurísticos [Raj 2012]. Considerando que a residência conhece quem reside em seu interior, a mesma pode inferir soluções inteligentes para eventuais situações emergenciais, como: cortar o fornecimento de gás à um fogão ao perceber que um usuário se ausentou do imóvel, e que tenha acidentalmente esquecido um acendedor aceso; ou, contactar os parentes ou corpo de bombeiros que uma pessoa de idade mais avançada encontra-se em estado emergencial de saúde. Permitir que a residência perceba o que ocorre em seu interior para prevenir situações catastróficas é uma subárea da automação residencial, denominada *Home Care* [Giroux and Pigot 2005]. *Home Care* é recomendada para pessoas que necessitam de cuidados contínuos e possuem dificuldade para realizar as tarefas do cotidiano.

Uma aplicação sensível ao contexto têm implicações positivas em uma residência, pois ela assume a responsabilidade de distribuir corretamente as funções e configurações, além de monitorar seus respectivos dispositivos finais. Deste modo, permitir que uma residência reconheça o que acontece em seu interior para adaptar seus ambientes internos de forma proativa, de acordo com as necessidades de seus usuários é o principal objetivo e contribuição deste trabalho. Nossa aplicação sensível ao contexto utiliza uma arquitetura distribuída e emprega dispositivos computacionais de dimensões reduzidas, que podem ser acoplados de forma pervasiva ao ambiente. Além disso, nossa solução consegue mover o contexto computacional, todo ou em parte, de um ambiente para outro, dentro de uma residência.

Este trabalho está organizado em mais cinco seções, onde a Seção 2 apresenta uma contextualização sobre automação residencial e computação ubíqua. A Seção 3 descreve nossa aplicação sensível ao contexto para automação residencial. A Seção 4 apresenta a avaliação experimental da nossa solução proposta. Os trabalhos relacionados são apresentados na Seção 5. Por fim, a Seção 6 apresenta a conclusão deste trabalho.

2. Contextualização

Nesta seção é apresentada uma breve contextualização sobre os temas que são abordados neste trabalho. Na Seção 2.1 são apresentados conceitos sobre automação de residências. Por fim, a Seção 2.2 discorre sobre conceitos principais de computação ubíqua.

2.1. Automação Residencial

A evolução da automação aplicada à indústria trouxe novos padrões, novas soluções inteligentes sofisticadas e especializadas, além de aplicações flexíveis para os meios de produção. Consequentemente, com o intuito de melhorar a qualidade de vida das pessoas, essas tecnologias de automação foram migradas e adaptadas do ambiente industrial para o ambiente residencial. Neste contexto, a automação residencial busca trazer mais segurança, acessibilidade, conveniência para os usuários do sistema de automação, e melhor eficiência energética para a residência. Por tais características, esse tipo de automação vem sendo cada vez mais requisitada em novos empreendimentos imobiliários [Muratori and Bó 2011].

Sistemas de automação residencial utilizam dispositivos de hardware, como sensores e atuadores, que são fundamentais para o acionamento dos eletroeletrônicos e coleta de informações sobre o estado de funcionamento dos mesmos. Deste modo, a automação residencial é caracterizada por empregar um ou mais equipamentos que realizam processos de modo automatizado dentro de uma residência, como o abrir ou fechar de janelas, ou o ligar e desligar de eletroeletrônicos [Baraka et al. 2013]. Para obter maior precisão na coleta de informações da residência, os sensores precisam estar estrategicamente acoplados ao ambiente. Além disso, os atuadores, que podem ser de acionamento magnético, pneumático, elétrico, ou misto, são responsáveis por efetivar o acionamento dos dispositivos que são aplicadas a automação.

Os sistemas de automação residencial são basicamente divididos em dois tipos de arquiteturas, uma mais rudimentar, baseada somente em automação, e outra mais inteligente, que leva em consideração o comportamento do usuário que utiliza o sistema de automação. Na primeira arquitetura, que é baseada somente em automação, o usuário é quem deve se adaptar ao sistema de automação implantado. Este tipo de automação residencial utiliza dispositivos que são configurados pelos próprios usuários. Deste modo, este tipo de automação favorece a implantação de operações voltadas a deixar o sistema com interfaces minimalistas e objetivas, empregando simples controles remotos ou apenas sensores de movimentos [Ferreira 2010].

Por outro lado, a automação residencial baseada em comportamento incorpora mecanismos de tomada de decisão que utilizam técnicas de aprendizagem de máquina. Esta automação inteligente modela regras a partir do comportamento de seus usuários, e das interações dos mesmos com o sistema de automação implantado na residência. Assim, este tipo de automação faz com que a residência busque se adaptar aos estilos e

costumes dos usuários, monitorando o ambiente e se antecipando as necessidades das pessoas [Ferreira 2010]. Além disso, esta tecnologia sensível ao contexto integra inteligência e pervasividade computacional, o que a torna uma solução vantajosa e benéfica para seus usuários.

2.2. Computação Ubíqua

O termo computação ubíqua, também conhecido como UbiComp ou computação pervasiva, começou a ser utilizado no início da década de 1990, mais precisamente em 1991. Este termo foi sugerido por Mark Weiser, em seu artigo intitulado *The Computer for the 21st Century* [Weiser 1991]. Na época, Weiser era um renomado cientista de um centro de pesquisas da multinacional Xerox¹. A computação ubíqua se diferencia da computação tradicional, que é caracterizada pelo baixo grau de integração com o ambiente em que se encontra, possuindo um nível muito baixo de mobilidade. Além disso, na computação tradicional, *mainframes*, estações de trabalho e *desktops* realizam tarefas com o auxílio de interfaces de interação tradicionais, tais como mouse e teclado.

Dentro dos princípios da computação ubíqua, dispositivos com poder de processamento computacional são embarcados em todos os cantos de um ambiente. Estes dispositivos são projetados para atuar de forma inteligente, percebendo a presença e interação dos usuários para reagir instantaneamente perante situações ocasionadas [Barbosa et al. 2008]. Sistemas ubíquos visam tornar a computação onipresente, tornando natural a interação entre usuários e sistemas computacionais, da mesma forma que ocorreu com outras tecnologias difundidas no cotidiano da sociedade moderna, *e.g.*, a escrita ou o uso da energia elétrica. A computação ubíqua utiliza uma abordagem que permite a movimentação de seus usuários, permitindo que os serviços ou aplicações os localizem e ofereçam soluções inteligentes e sofisticadas [nam Lee et al. 2009].

3. Arquitetura do Sistema de Automação Residencial Sensível ao Contexto

Para desenvolver o nosso sistema de automação residencial sensível ao contexto, elencamos alguns requisitos que a aplicação deve satisfazer. Dentre tais requisitos, é possível listar que a aplicação deve ter funcionalidades, tais como: (i) detectar a presença de um usuário em determinado cômodo da residência em que o sistema atua; (ii) capturar informações sobre o estado computacional dos dispositivos eletroeletrônicos de um cômodo; (iii) configurar corretamente os equipamentos da residência de acordo com as necessidades de um usuário; e (iv) armazenar as informações de contexto de um ambiente para uso posterior.

Para atender tais requisitos e fornecer estas funcionalidades ao nosso sistema de automação residencial sensível ao contexto, desenvolvemos uma arquitetura distribuída. Esta arquitetura possui dois componentes bem definidos, onde controladores locais atuam diretamente em um cômodo da residência, e um controlador centralizado distribui corretamente as informações de contexto para os cômodos, considerando as configurações previamente armazenadas em uma base de dados, mantida pelo controlador central. A Figura 1 apresenta os componentes de nossa arquitetura proposta. Além disso, o controlador central funciona como um servidor concorrente, que pode atender mais de um controlador local em simultâneo.

¹<http://www.xerox.com/>

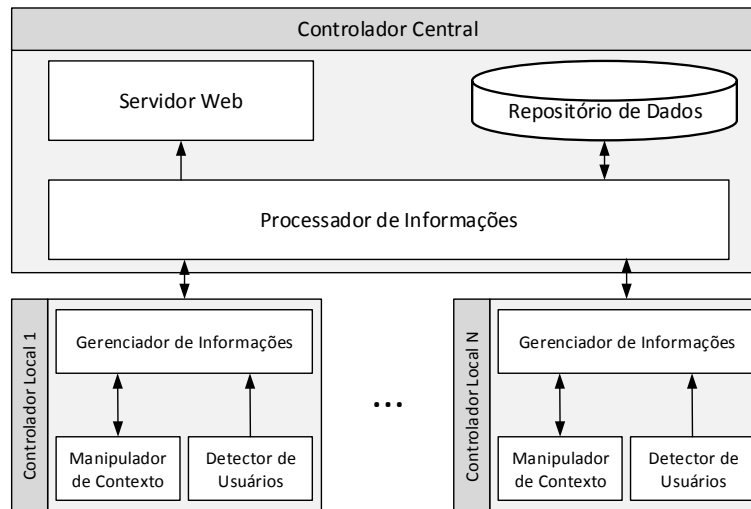


Figura 1. Arquitetura do sistema de automação residencial sensível ao contexto.

Levando em consideração um controlador local, que é a aplicação cliente de nosso sistema distribuído, este dispositivo possui três componentes bem definidos. Estes componentes utilizam informações de contexto fornecidas pelo controlador central para aplicar a automação diretamente no cômodo em que um usuário se encontra, e possuem funcionalidades específicas. Cada componente do controlador local é descrito a seguir:

- **Detector de Usuários:** responsável por reconhecer qual é o usuário que interage com o sistema de automação, e o associa à um código identificador. Além disso, este componente também é responsável por detectar se o usuário entrou ou saiu de um cômodo da residência;
- **Gerenciador de Informações:** este componente se comunica diretamente com o controlador central, recebendo e enviando mensagens contendo informações contextuais sobre a residência. Quando este componente recebe uma mensagem de contexto do controlador central repassa para o Manipulador de Contexto, que efetiva a configuração dos dispositivos do cômodo. Entretanto, este componente também pode receber informações do Manipulador de Contexto, deste modo, o Gerenciador de Informações fica responsável por formatar e enviar estas informações para o controlador central do sistema;
- **Manipulador de Contexto:** este componente monitora constantemente o estado do ambiente, e configura diretamente os dispositivos eletroeletrônicos de um determinado cômodo. O Manipulador de Contexto se comunica diretamente com o Gerenciador de Informações do controlador local, para enviar informações do contexto computacional do cômodo, ou receber os dados de contexto para configurar os dispositivos de um ambiente.

Por outro lado, o controlador central executa em um dispositivo computacional distinto dos controladores locais, e é tratado como o servidor de nosso sistema de automação residencial sensível ao contexto. O controlador central mantém um repositório de dados, que armazena informações contextuais de todos os cômodos para cada usuário que mora na residência. Além disso, este servidor contém um Processador de

Informações, responsável por requisitar e atualizar as informações do repositório de dados, e repassar corretamente os dados contextuais para os devidos controladores locais. Por fim, este componente também fornece um servidor web, que provê uma interface de comunicação entre nosso sistema de automação e o usuário. Esta interface web foi desenvolvida para ser acessada de modo local, porém, dependendo da necessidade do usuário, ela também pode ser disponibilizada para acesso remoto, fornecendo informações em tempo real sobre os ambientes da residência.

Para melhor ilustrar o funcionamento de nossa proposta, a Figura 2 apresenta um diagrama de sequência. Este diagrama apresenta uma sequência de treze passos, onde demonstra as mensagens trocadas entre os componentes do sistema quando um usuário entra ou sai de um ambiente da residência.

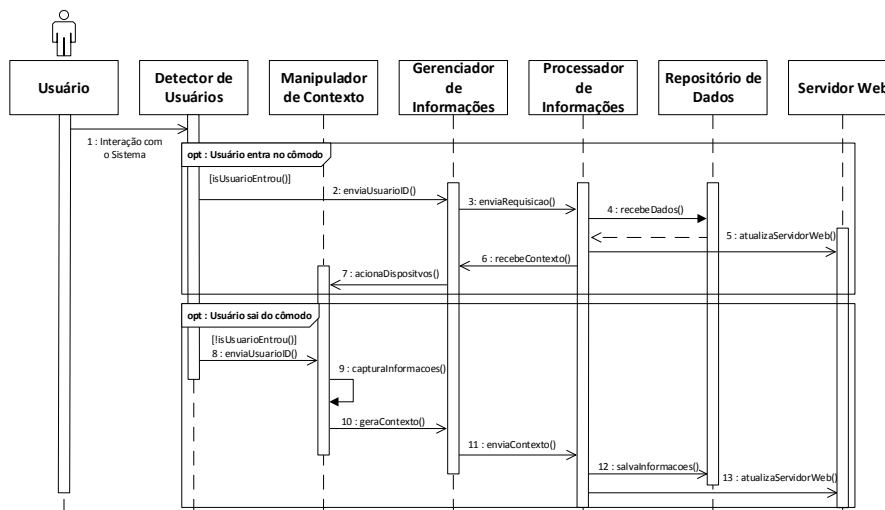


Figura 2. Diagrama de sequência do sistema de automação residencial.

4. Avaliação Experimental

Esta seção descreve o protótipo que utilizamos para avaliar nossa proposta. Além disso, esta seção também descreve os resultados obtidos na avaliação experimental do nosso sistema de automação residencial sensível ao contexto. Para avaliar o sistema foram elaborados dois experimentos distintos.

4.1. Protótipo

Para poder realizar uma análise experimental de nossa solução, desenvolvemos um protótipo. Nosso protótipo utilizou dispositivos computacionais de dimensões reduzidas, que podem facilmente ser acoplados ao ambiente de modo pervasivo. Entretanto, nada impede que nosso sistema de automação sensível ao contexto seja reproduzido utilizando equipamentos de dimensões normais. Deste modo, para prototipar nosso controlador central, utilizamos um microcomputador Raspberry PI². Este microcomputador possui as

²<http://www.raspberrypi.org/>

dimensões de um cartão de crédito convencional, e possui poder de processamento suficiente para executar tarefas que originalmente são realizadas por *desktops*, tais como processamento de jogos, edição de texto e planilhas, navegação na internet, entre outras tarefas. Escolhemos a linguagem de programação Python³ para a prototipação de nosso controlador central.

No contexto dos controladores locais, escolhemos como dispositivo computacional o Arduino⁴ UNO. Deste modo, este equipamento foi responsável por configurar e monitorar os ambientes da residência. Assim como o Raspberry PI, o Arduino UNO é um hardware de baixo custo, versátil e de dimensões reduzidas. Em nosso protótipo a interconexão dos dispositivos foi feita utilizando uma rede LAN cabeada. Além disso, por questões de experimentação, simulamos o funcionamento de eletroeletrônicos com o auxílio de periféricos, tais como potenciômetros, pilhas de alimentação, botões, *protoboards*, pequenos motores, e LEDs. A programação interna dos controladores locais foram realizadas utilizando a linguagem de programação nativa do Arduino.

Com este ferramental, elaboramos dois experimentos para realizar uma análise experimental de nosso sistema de automação residencial sensível ao contexto. Simulamos um cenário de uma residência com um único usuário, que se movimenta entre os cômodos e configura os dispositivos destes ambientes. Tais experimentos foram repetidos trinta vezes cada, para podermos atingir um nível de confiança de 95% em nossos resultados, que são descritos a seguir.

4.2. Cenário 1 - Controle de um ventilador

Neste primeiro cenário, procuramos simular o comportamento de um usuário que visita periodicamente um cômodo. Com isso, a cada vez que o usuário entra neste ambiente, ele altera a configuração de um dispositivo e se retira do local. Por tanto, para simular este cenário criamos um ambiente com um único dispositivo, no caso um ventilador, que inicialmente estava desligado. A Figura 3 apresenta o esquemático do circuito utilizado para simular este cenário, onde utilizamos uma *protoboard*, dois botões para capturar a presença do usuário no cômodo, pilhas de alimentação, um pequeno motor, responsável por simular o funcionamento do ventilador, e um potenciômetro. Tal potenciômetro permite alterar a velocidade de rotação do ventilador em tempo de execução.

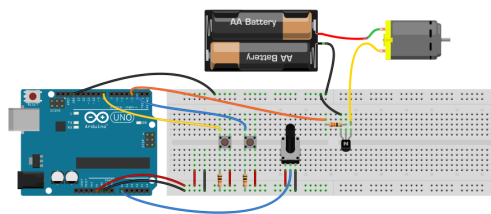


Figura 3. Esquemático do circuito de controle do primeiro cenário de teste.

4.3. Cenário 2 - Controle de luminosidade entre dois ambientes

No último experimento, procuramos simular o comportamento de um usuário que se movimenta constantemente entre dois cômodos. Neste caso, os dois cômodos continham

³<https://www.python.org/>

⁴<http://www.arduino.cc/>

sistemas de iluminação semelhantes, *i.e.*, ambos os cômodos possuíam dispositivos similares, onde o primeiro cômodo era composto por um sistema de iluminação com três luminárias, e o segundo com cinco luminárias. Essa característica de similaridade é designada por nosso sistema para dispositivos idênticos, com o mesmo código identificador, porém em ambientes distintos. Com este experimento, procuramos avaliar se nossa solução consegue de fato migrar o contexto computacional de um ambiente para outro, considerando a mobilidade do usuário. Para tanto, simulamos os dois ambientes em *proto-board*, onde ambos possuíam dois botões que permitiam detectar a presença do usuário em um cômodo. Entretanto, como ilustrado na Figura 4, além destes botões, o primeiro ambiente foi composto por três LEDs e três potenciômetros, enquanto o segundo ambiente foi composto por cinco LEDs.

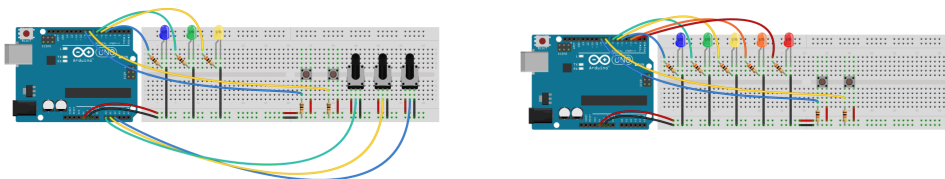


Figura 4. Esquemático dos circuitos de controle do primeiro e segundo ambiente do segundo cenário de teste.

4.4. Resultados dos Experimentos

Analisamos as ações de nossa solução no primeiro cenário, e durante as trinta repetições do experimento, percebemos que em todos os casos, nosso sistema se comportou como o esperado. Ou seja, em todas as trinta repetições, nossa solução conseguiu manter a integridade das informações contextuais transferidos entre o controlador local e o controlador central. Deste modo, armazenando com êxito os contextos no repositório de dados e reconfigurando corretamente os dispositivos do ambiente. A Figura 5 apresenta as informações apresentadas no console da aplicação servidor durante uma repetição deste experimento. Esta figura informa que no início do experimento, quando o usuário (ID de usuário 1001) entrou no cômodo, o dispositivo 104 (ventilador) encontrava-se desligado (valor 0). Entretanto, quando o usuário saiu do cômodo, o contexto salvo pelo sistema indicava que o ventilador estava ligado e na velocidade de rotação máxima (valor 255).

```
Python 3.3.1 (v3.3.1:d9893d13c628, Apr 6 2013, 20:30:21) [MSC v.1600 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
Mensagem de Contexto inicial: '1001;104;0'
Mensagem de Contexto Final: '1001;104;255'
>>>
```

Figura 5. Imagem da interface de saída do controlador central no primeiro cenário.

Assim como no primeiro experimento, todas as repetições do segundo experimento obtiveram o resultado esperado, não apresentando nenhum comportamento anormal. Em todas as repetições do experimento, nosso sistema de automação residencial conseguiu migrar o contexto computacional dos sistemas de luminosidade entre os dois ambientes. Deste modo, constatamos que nossa abordagem consegue migrar o contexto

computacional de um cômodo para outro, levando em consideração o posicionamento e a mobilidade do usuário. A Figura 6 apresenta os dados manipulados pelo Processador de Informações durante uma repetição deste experimento. Nesta figura podemos analisar que no início do experimento, quando o usuário (ID de usuário 1001) entrou no primeiro ambiente, as três luminárias (106, 107 e 108) estavam desligadas (valor 0). Porém, quando o usuário se retirou do primeiro cômodo, ele deixou as luminárias 106, 107 e 108 ligadas, com as configurações 125, 217 e 20 respectivamente. Por fim, quando entrou no segundo cômodo, o contexto computacional foi migrado com êxito, *i.e.*, o sistema de luminosidade deste cômodo foi configurado tomando por base as informações que estavam previamente armazenadas no repositório de dados. Assim, nosso sistema consegue com êxito, migrar o contexto computacional entre dispositivos similares.

```
Python 3.3.1 (v3.3.1:d9893d13c628, Apr 6 2013, 20:30:21) [MSC v.1600 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
Mensagem de Contexto Inicial do Primeiro Ambiente: '1001;106;0;107;0;108;0'
Mensagem de Contexto Final do Primeiro Ambiente: '1001;106;125;107;217;108;20'
Mensagem de Contexto Inicial do Segundo Ambiente: '1001;106;125;107;217;108;20;109;0;110;0'
Mensagem de Contexto Final do Segundo Ambiente: '1001;106;125;107;217;108;20;109;0;110;0'
```

Figura 6. Imagem da interface de saída do controlador central no primeiro cenário.

5. Trabalhos Relacionados

Nesta seção apresentamos estudos que são relacionados com a nossa proposta. Tais estudos apresentam soluções que empregam dispositivos computacionais de dimensões reduzidas nas tarefas de automação. [Baraka et al. 2013] apresenta um sistema de automação de baixo custo, que utiliza como controlador central um Arduino. Além disso, este trabalho integra comunicação entre os dispositivos de forma cabeada, utilizando o protocolo X10, e sem fio, utilizando redes Zigbee. Este estudo também permite configurar os dispositivos da residência utilizando um aplicativo de Android. [Del Carmen Curras-Francos et al. 2014] apresenta uma plataforma baseada em Arduino. Esta plataforma é disponibilizada abertamente para estudos e desenvolvimento de novas aplicações de automação e módulos para Arduino. Por fim, [Jain et al. 2014] apresenta uma aplicação para automatizar residências por e-mail. Esse estudo emprega Raspberry PI como servidor do sistema. Este Raspberry executa um algoritmo que consegue acionar dispositivos a partir da leitura do campo *subject* de novos e-mails recebidos pela aplicação.

6. Conclusão

Este trabalho apresentou um sistema de automação residencial sensível ao contexto. Nossa aplicação foi desenvolvida com o intuito de permitir que a residência reconheça seus usuários e adapte seus ambientes internos de acordo com as preferências dos mesmos. Assim, apresentamos um sistema que consegue capturar informações sobre um ambiente, e transferir o contexto computacional de um cômodo para outro, considerando a mobilidade do usuário. Com o intuito de realizar uma análise experimental de nossa solução, desenvolvemos um protótipo, que foi avaliado em dois cenários distintos. Perante estes experimentos, os resultados indicaram que nosso sistema se comportou de maneira satisfatória.

Concluimos que nosso sistema de automação residencial sensível ao contexto permite que a residência automatize tarefas de maneira proativa. Deste modo, nosso sistema traz benefícios aos seus usuários, permitindo que estes se concentrem na realização de outras tarefas. Por fim, como trabalhos futuros, pretendemos aprimorar nosso protótipo, aplicando um esquema de comunicação sem fio entre os controladores locais e o controlador central. Além disso, também pretendemos implantar um mecanismo de gerência de conflitos baseado em aprendizagem de máquina, permitindo que nosso sistema de automação sensível ao contexto decida, de modo inteligente, o que fazer em situações onde exista mais de um usuário dentro de um ambiente específico da residência.

Referências

- Baraka, K., Ghobril, M., Malek, S., Kanj, R., and Kayssi, A. (2013). Low cost arduino/android-based energy-efficient home automation system with smart task scheduling. In *Computational Intelligence, Communication Systems and Networks (CICSyN), 2013 Fifth International Conference on*, pages 296–301.
- Barbosa, J., Hahn, R., Barbosa, D., and Geyer, C. (2008). Learning in small and large ubiquitous computing environments. In *Embedded and Ubiquitous Computing, 2008. EUC '08. IEEE/IFIP International Conference on*, volume 1, pages 401–407.
- Del Carmen Curras-Francos, M., Diz-Bugarin, J., Garcia-Vila, J., and Orte-Caballero, A. (2014). Cooperative development of an arduino-compatible building automation system for the practical teaching of electronics. *Tecnologias del Aprendizaje, IEEE Revista Iberoamericana de*, 9(3):91–97.
- Ferreira, V. Z. G. (2010). A domótica como instrumento para a melhoria da qualidade de vida dos portadores de deficiência. Monografia, Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, João Pessoa.
- Giroux, S. and Pigot, H. (2005). *From Smart Home to Smart Care*, volume 15. IOS Press.
- Jain, S., Vaibhav, A., and Goyal, L. (2014). Raspberry pi based interactive home automation system through e-mail. In *Optimization, Reliability, and Information Technology (ICROIT), 2014 International Conference on*, pages 277–280.
- Jiang, L., you Liu, D., and Yang, B. (2004). Smart home research. In *Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on*, volume 2, pages 659–663 vol.2.
- Muratori, J. R. and Bó, P. H. D. (2011). Automação residencial: Histórico, definições e conceitos. In *O Setor Elétrico*, number 62, pages 70 – 76, São Paulo. Mensal.
- nam Lee, H., Lim, S.-H., and Kim, J.-H. (2009). Umons: Ubiquitous monitoring system in smart space. *Consumer Electronics, IEEE Transactions on*, 55(3):1056–1064.
- Raj, S. (2012). Implementation of pervasive computing based high-secure smart home system. In *Computational Intelligence Computing Research (ICCIC), 2012 IEEE International Conference on*, pages 1–8.
- Weiser, M. (1991). The computer for the 21st century. *Scientific american*, 265(3):94–104.