

Explorando a Consciência de Situação em Smart Homes utilizando a Plataforma LabVIEW

Maicon Ança dos Santos, UFPel; Patrícia Teixeira Davet, UFPel

Resumo—Ambientes Inteligentes direcionados às Smart Homes (Casas Inteligentes) são capazes de prestar e integrar serviços otimizados para atender as tarefas cotidianas, de forma transparente, tornando a vida de seus habitantes mais simples. As Smart Homes possuem sistemas inteligentes capacitados a responder de forma autônoma a necessidade dos usuários, possibilitando acesso a informações diversas, comunicação interpessoal e até o controle dos elementos físicos do ambiente. Estes dados captados e gerados pelo sistema podem ser tratados como informações contextuais e estas quando abstraídas em uma visão de mais alto nível definem uma situação. O artigo discute o desenvolvimento de sistemas inteligentes, conscientes da situação e aplicados em Smart Homes que utilizam o software LabVIEW como um processador de regras contextuais.

Palavras-Chave—Consciência de situação, computação ubíqua, smart home, LabVIEW.

1 Introdução

A perspectiva introduzida pela Computação Ubíqua vem se tornando uma tendência mundial em função do rápido avanço tecnológico dos dispositivos móveis, redes sem fio, redes de sensores, sistemas e dispositivos inteligentes, sistemas distribuídos, grades computacionais, sistemas autônomos e outras tecnologias relacionadas a integração de sistemas. São tecnologias convergentes, que se adaptam sinergicamente no cotidiano dos usuários. Tendo em vista a dinamicidade, decorrente desta integração, a capacidade de reação ao contexto torna-se um componente importante e fundamental na Computação Ubíqua [1].

Considerando esta perspectiva, o emprego da consciência de situação é essencial para os Sistemas Ubíquos e tem ganhado espaço na literatura especializada [2].

Diversos autores tentam desenvolver formas inovadoras de como tornar o uso da computação

ubíqua e pervasiva mais próxima da visão de Weiser segundo a qual “as tecnologias mais profundas são aquelas que desaparecem. Elas se integram à vida cotidiana até se tornarem indistinguíveis da mesma”.

Com isso, surgem cenários interessantes, que empregam o poder computacional em qualquer lugar, a qualquer momento (always-on), acessível com qualquer dispositivo. A completa integração de diversos (e diferentes) dispositivos, aplicações e plataformas com o ambiente físico possibilitam a criação de ambientes dinâmicos e heterogêneos chamados de espaços pervasivos (pervasive spaces), ambientes inteligentes (smart spaces), casas inteligentes (smart homes) ou também de ambientes reativos (responsive environments) [3]. Em todos esses casos, sempre é visada a automatização e o aperfeiçoamento das soluções empregadas pelos usuários em problemas diários de forma a antecipar-se e adaptar-se às suas necessidades.

Cada vez mais, o homem tenta adaptar a casa onde vive às suas necessidades, tendo em vista fatores como economia, conforto, segurança, ecologia, gestão do espaço, capacidade de adaptação e integração. É neste âmbito que surgiu o conceito de Smart Homes, casas que se utilizam de diversas ferramentas tecnológicas, muitas vezes já utilizadas com sucesso em outras áreas, como por exemplo a indústria automobilística, para gerenciar todos os recursos habitacionais e

• **Maicon Ança dos Santos:** Programa de Pós-Graduação em Computação, Universidade Federal de Pelotas - UFPel, Centro de Desenvolvimento Tecnológico - CDTec.
E-mail: maiconsantos@gmail.com

• **Patrícia Teixeira Davet:** Programa de Pós-Graduação em Computação, Universidade Federal de Pelotas - UFPel, Centro de Desenvolvimento Tecnológico - CDTec.
E-mail: ptdavet@gmail.com

tornar o cotidiano de seus usuários mais simples, satisfazendo as suas necessidades de comunicação, conforto e segurança.

Dentre as ferramentas utilizadas para este fim está o LabVIEW, software criado pela *National Instruments* que possui um ambiente de desenvolvimento de fácil utilização e programação, integrado a diversos sistemas operacionais e com características que facilitam conexão a uma variada gama de hardwares e outros softwares. Este trabalho tem como objetivo realizar uma pesquisa de projetos na área de Smart Homes, que exploram a consciência de situação como forma de tratar e inferir informações contextuais coletadas por diferentes sensores, utilizando para tal a plataforma LabVIEW.

O artigo está organizado da seguinte forma: a seção 2 apresenta uma revisão sobre os principais conceitos da Computação Sensível ao Contexto. A seção 3 aborda definições dos conceitos, princípios e requisitos inerentes em uma Smart Home tendo como premissa ela ser um Ambiente Inteligente controlado por Inteligência Ambiental. A seção 4 apresenta o LabVIEW, sua linguagem de programação e trata de conceitos introduzidos pela plataforma. A seção 5 discute os trabalhos relacionados. Por fim, a seção 6 apresenta as considerações finais.

2 Computação Sensível ao contexto

Ambientes ubíquos oferecem grande interconectividade entre os serviços computacionais e também interações com o mundo físico onde estão inseridos, gerando complexidade e dinamismo nas aplicações.

Com isso, adaptações no comportamento das aplicações são necessárias de acordo com mudanças que ocorrem a sua volta. A computação sensível ao contexto é responsável por trabalhar como e quando efetuar as adaptações necessárias ao ambiente.

Definições

Em computação, a primeira noção de contexto foi dada por Schilit [4], o qual propôs a divisão das informações de contexto em três categorias:

- **Contexto computacional:** que diz respeito a recursos computacionais utilizados pelo usuário em determinada atividade.

- **Contexto do usuário:** informações como perfil do usuário, localização, pessoas ao redor e até mesmo a situação social (sozinho, em grupo, etc.).
- **Contexto físico:** são informações do ambiente físico como luz, ruído, condições do trânsito, temperatura, etc.

Na sequência, demais pesquisadores propuseram a adição de mais uma categoria: o contexto temporal que diz respeito às informações temporais hora, dia da semana, mês, etc. [5]

A figura 1 ilustra a classificação de informações de contextos com base na origem a partir de fatores humanos e do ambiente [6].

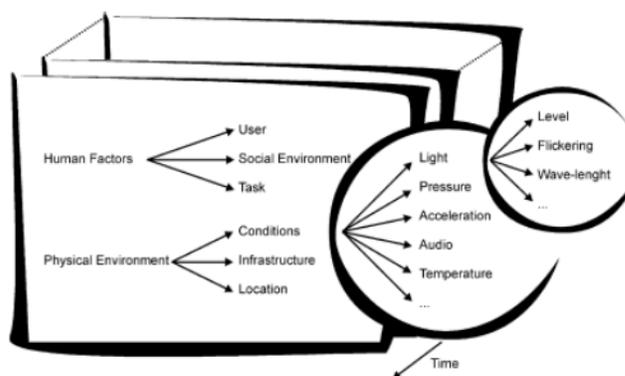


Figura 1. Classificação de contextos. Extraído de [7]

Dey [8] nos traz uma das definições de contexto mais aceitas no âmbito da área computacional que diz que “contexto é qualquer informação que pode ser usada para caracterizar a situação de uma entidade. Uma entidade é uma pessoa, um lugar ou um objeto que é considerado relevante para a interação entre o usuário e uma aplicação, incluindo o próprio usuário e a própria aplicação”.

Outros pesquisadores nos apresentam mais definições para contexto, conforme segue: Lieberman [9] definiu como “tudo aquilo que afeta a computação, exceto as entradas e saídas explícitas”; para Kotz [5], contexto é o conjunto de estados do ambiente e configurações que determinam tanto o comportamento ou eventos de um aplicativo e são de interesse do usuário; Gray [10], conceituaram contexto como “propriedades que caracterizam um fenômeno, são sentidas (sensed) e potencialmente relevantes para as tarefas executadas por uma aplicação e/ou os meios pelos quais essas tarefas são desempenhadas”; para Yamin

[11], “contexto é toda informação relevante para a aplicação que pode ser obtida da infraestrutura computacional, cuja alteração em seu estado dispara um processo de adaptação na aplicação”; em trabalho mais atual, Gray [12] remete à idéia que as informações de contexto referem-se a informações que caracterizam o ambiente físico onde se está inserido.

As informações sobre contexto podem ser descritas em cinco dimensões:

- **Quem:** pelo fato dos humanos viverem em sociedade e serem influenciados pelo aspecto social as informações a respeito de todas as pessoas envolvidas em uma dada atividade devem ser obtidas;
- **O quê:** percepção daquilo que o usuário está fazendo. Em sistemas onde atividades variadas podem ser desenvolvidas, identificar o quê um usuário está fazendo em determinado momento pode ser uma tarefa complexa
- **Onde:** as informações da localização onde uma tarefa está sendo desenvolvidas são importantes. Geralmente ela é associada com outras dimensões como a temporal (quando) e a identificação (quem) no sentido de prover funcionalidades às aplicações. Em ambientes móveis a localização, as pessoas e os objetos que estão em volta do usuário são dinâmicos, possibilitando numerosas situações diferentes.
- **Quando:** o aspecto temporal do contexto tem sido utilizado para indexar um registro capturado ou para informar por quanto tempo um usuário esteve em um determinado local
- **Por quê:** entender o por quê da ação que o usuário está fazendo é mais complexo do que entender o quê ele está fazendo. Esse aspecto é um dos principais desafios da computação consciente de contexto pois acaba impactando em aspectos de predição de contexto.

Observa-se que toda a definição de contexto engloba todas as informações instantâneas das características de entidades que compõe um ambiente.

Modelagem de contexto

De acordo com Bettini [13], uma boa modelagem das informações de contexto pode reduzir a com-

plexidade, melhorando sua manutenção e escalabilidade. Também são definidos critérios que os modelos de contexto devem obedecer:

- **Heterogeneidade e mobilidade:** os modelos precisam ser capazes de manipular uma grande quantidade de informações obtidas de dispositivos que diferem entre si na taxa de atualização e na semântica dos dados. Como as informações de contexto podem ser obtidas de fontes estáticas, dinâmicas, ou ainda serem geradas a partir de outras informações de contexto, o modelo deve ser capaz de expressar diferentes tipos de informações e proporcionar que seja efetuado o gerenciamento destas de acordo com a necessidade das aplicações.
- **Relacionamentos e dependências:** é importante que o modelo consiga lidar com os vários relacionamentos entre as informações de contexto que são importantes para o comportamento das aplicações.
- **Temporalidade:** os modelos devem ser capazes de representar o histórico dos acontecimentos de contextos os quais serão gerenciados pelo sistema de gerenciamento de contexto.
- **Imperfeição:** um bom modelo deve possuir suporte a verificação da consistência e qualidade da informação buscando uma correta inferência sobre os dados.
- **Raciocínio:** é importante que modelos possuam verificação de consistência dos dados e técnicas de raciocínio que podem, inclusive, ser utilizadas para derivar novas informações de contexto a partir de dados armazenados ou situações que representem o mundo real com alto grau de abstração.
- **Usabilidade e formalismo:** os modelos devem possibilitar que os projetistas representem de forma fácil e formal os fatos do mundo real em informações de contexto que podem ser usadas pelas aplicações.
- **Provisionamento eficiente de contexto:** o modelo deve disponibilizar formas que agilizem o acesso aos objetos relevantes. Isso geralmente é obtido através do uso de índices os quais são referenciados como contextos primários, enquanto informações acessadas a partir destes são chamadas de contextos

secundários.

Outros autores [14] [6] [15] apresentaram propostas de classificações para os modelos de contexto baseados nas técnicas utilizadas, destacando o uso de tuplas "chave/valor", representando o atributo e o valor correspondente. Tal técnica é utilizada por linguagens de marcação como XML e pelo padrão W3C para definição de dispositivos móveis. Críticas a esse modelo se referem a: (i) tipos de contexto que podem ser capturados, (ii) captura de relações, dependências, temporariedade e qualidade das informações do contexto, (iii) verificação da qualidade do contexto, e (iv) suporte a inferência e incertezas de contexto e também a abstrações de alto nível.

Contextos também podem ser representados por modelagens com foco em informações específicas do domínio de um problema. Como exemplo, o modelo W4 representa as informações contextuais através de tuplas *Who*, *What*, *Where*, *When* [16]. Este modelo foi criado para ser simples com capacidade para gerenciar as informações de contexto vindas de diversas fontes e relacionadas aos mais variados tipos de situações que podem ser desde propriedades físicas ou geográficas até fatos sociais que aconteçam no mundo.

Consciência de situação

Entende-se por consciência de situação a percepção de elementos ambientais com relação ao tempo ou espaço, a compreensão do seu significado e a projeção do seu estado depois da mudança de alguma variável, como, por exemplo, um evento pré-determinado.

A noção de situação é usada como um conceito de alto-nível para representação de estado [7]. Aplicações que se utilizam de técnicas de sensibilidade à situação são chamadas de aplicações conscientes da situação. Nelas, interpretações semânticas externas de contexto de baixo nível permitem uma especificação de alto-nível do comportamento humano e da sua interação como sistema.

Uma vantagem das aplicações sensíveis à situação está ligada a facilidade de proporcionar uma representação compreensível pelos humanos dos dados gerados para as aplicações.

A figura 2 ilustra a relação de contexto ao nível de abstração, demonstrando que as informações

de contexto de baixo nível são semanticamente interpretadas por camadas de contexto de mais alto-nível

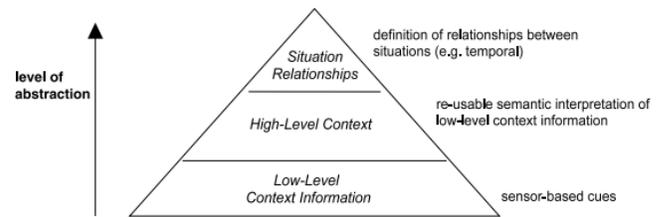


Figura 2. Níveis de abstração. Extraída de [7]

As situações de contexto pode apresentar as seguintes situações:

- **Generalização:** uma situação pode ser considerada como mais generalista que outra se a sua ocorrência tem implicação em outra mais específica.
- **Composição:** uma situação pode ser decomposta em um conjunto de pequenas situações.
- **Dependência:** uma situação depende de outra situação se a ocorrência da primeira é determinada pela ocorrência da segunda.
- **Contradição:** duas situações podem ser consideradas como mutuamente exclusivas uma da outra se elas não podem ocorrer ao mesmo tempo, no mesmo lugar e com o mesmo objeto.
- **Sequência temporal:** uma situação pode ocorrer antes ou após outra situação, ou ainda no intervalo de outra situação.

Para o desenvolvimento de arquiteturas de software com suporte a execução de aplicações sensíveis à situação, algumas características devem ser atendidas:

- I) Suporte para especificar os requerimentos da sensibilidade à situação para as aplicações/agentes móveis, incluindo contexto e situações de interesse, e comportamento das aplicações/agentes em diferentes situações.
- II) Suporte para descoberta de contexto do ambiente baseando-se nas necessidades das diferentes aplicações/agentes existentes, adquirir contextos de várias fontes e entregar os dados dos contextos adquiridos para as aplicações móveis.
- III) Suporte para análise dos dados de contexto adquiridos para determinar a situação, a entrega dos resultados obtidos da análise

da situação para as aplicações móveis em um intervalo de tempo aceitável para disparar as ações ou adaptações necessárias.

IV) Suporte para compartilhamento de informações de contexto e de situações entre as aplicações móveis.

V) Suporte para reconfiguração das aplicações móveis em tempo de execução devido a requisitos da consciência da situação.

VI) Suporte para incorporação de vários mecanismos de reconhecimento/aprendizado das análises de situações.

VII) Interoperabilidade sintática e semântica entre as diferentes aplicações distribuídas.

Aplicações ubíquas requerem vários contextos para se comunicar e prestar seus serviços aos usuários. Entretanto, as técnicas de consciência de contexto não são suficientemente adequadas para trabalharem com múltiplos contextos e as interações dos usuários.

A consciência de situação tem por objetivo suprir tais dificuldades com uma modelagem de mais alto nível, com tópicos de pesquisa envolvendo as seguintes questões:

- **Representação:** definir as primitivas que serão utilizadas para construir a especificação lógica da situação.
- **Especificação:** formar uma especificação lógica da situação a qual pode ser adquirida por especialistas ou aprendida por dados de treinamento.
- **Raciocínio:** inferir situações de uma grande quantidade de dados imperfeitos de sensores e como manter a consistência e integridade de conhecimento das situações.

Sendo assim, a consciência de situação é uma forma promissora de resolução de limitações com relação à simplicidade dos modelos de representação de contexto, auxiliando no desenvolvimento de aplicação mais robustas e que possam, realmente, atender as necessidades do usuários.

3 Smart Homes

De acordo com Chin [17] “as casas do futuro serão ambientes ricos em uso da tecnologia. Serão utilizadas dezenas ou centenas de serviços pervasivos baseados em redes, alguns sendo executados no próprio ambiente e outros a partir de provedores

de serviços externos de forma que todos os dispositivos do ambiente sejam capazes de usar tais serviços”. Estas aplicações serão centralizadas e utilizarão um *middleware* para realizar a descoberta dos recursos acessíveis aos usuários dentro do ambiente.

Definições

Embora pareçam sinônimos, os termos Inteligência Ambiental e Ambientes Inteligentes são conceitos diferentes.

Trata-se Inteligência Ambiental como um novo paradigma computacional no qual pessoas estão imersas em um ambiente que é capaz de reconhecer a sua presença e o seu contexto e então adaptar-se de forma a responder as suas necessidades, hábitos, gestões e emoções [18]. Por outro lado, Ambientes Inteligente tem foco nos dispositivos. Eles precisam ser capazes de perceber as interações e os fatos do ambiente, respondendo ao que está acontecendo através de adaptações dos componentes (remetendo à ideia de computação sensível ao contexto).

Observa-se na figura 3 as interações que a Inteligência Ambiental pode ter com outras áreas da computação.

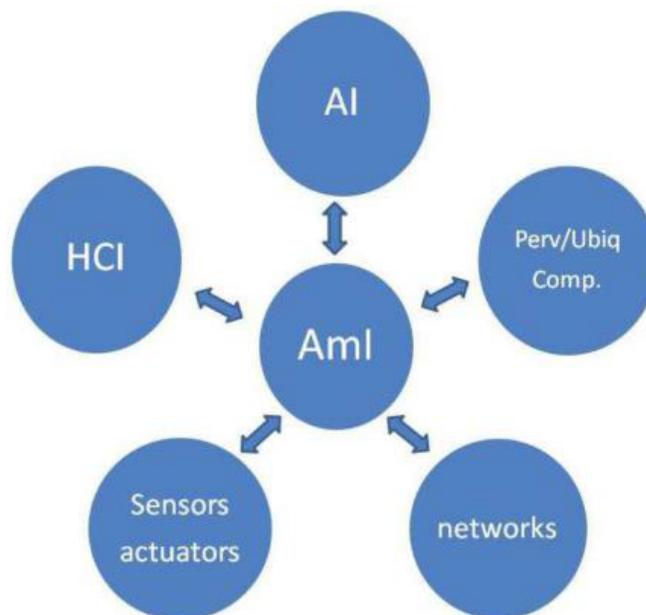


Figura 3. Interação da Inteligência Ambiental com outras áreas. Extraído de [7]

A partir do estudo da definição dos conceitos de Ambiente Inteligente e Inteligência Ambien-

tal, fica claro que o primeiro faz referência à questões físicas e de aplicação da Inteligência Ambiental nas interações entre os dispositivos. Por isso, quando se fala em Inteligência Ambiental inerentemente se fala em Ambientes Inteligentes e vice-versa. Dificilmente um existe isoladamente sem o outro.

Arquiteturas de Inteligência Ambiental

Várias arquiteturas buscam aplicar a Inteligência Ambiental à diversas áreas como, por exemplo, em segurança de ambientes críticos, ambientes educacionais, salas de conferência, ambientes residenciais e algumas que fazem uso de ontologias.

Porém, não existe uma definição clara de como deve ser uma arquitetura de Inteligência Ambiental, com cada uma focando em algum problema específico, sem existir um consenso do que realmente é necessário que tenham em comum.

A proposta que parece ser a mais adequada nos é apresentada na figura 4 onde são demonstrados os componentes básicos que devem existir com a intenção de resolver qualquer problema.

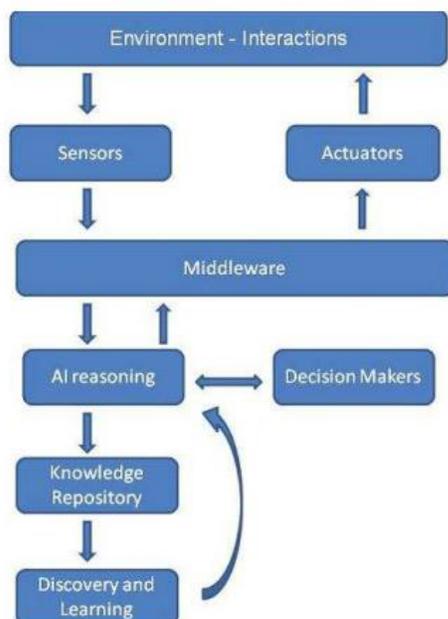


Figura 4. Arquitetura genérica da Inteligência Ambiental. Extraído de [7]

A proposta apresentada é composta pelas seguintes camadas:

- **Ambiente (*Environment*)**: é a camada mais baixa. Ela representa o mundo físico e todas as interações que ali acontecem;
- **Sensores (*Sensors*)**: representa todos os sensores que são utilizados na coleta de dados a partir do ambiente;
- **Middleware**: é a camada intermediária da arquitetura responsável por disponibilizar as condições necessárias para a execução de aplicações no ambiente;
- **Raciocínio (*AI Reasoning*)**: é responsável por toda a parte de inteligência que existe na arquitetura. Ela se comunica com as demais camadas e é considerada fundamental;
- **Tomada de decisões (*Decision Makers*)**: é a camada que contém os mecanismos necessários para auxiliar na tomada de decisão. Tais mecanismos vão fornecer as possíveis decisões que podem ser tomadas pelo sistema;
- **Repositório de conhecimento (*Knowledge Repository*)**: representa um repositório que contém todo o conjunto de conhecimento previamente aprendido;
- **Descoberta e aprendizado (*Discovery and Learning*)**: proporciona a descoberta de conhecimentos necessários para o raciocínio sobre o ambiente e também é responsável pelo aprendizado de novos conhecimentos resultantes de uma tomada de decisão;

Princípios fundamentais da Inteligência Ambiental

Nesta seção serão apresentadas alguns princípios que devem nortear o desenvolvimento de sistemas para que a Inteligência Ambiental possa estar mais próxima da realidade.

- **Efetuar sensoriamento**: devido a Ambientes Inteligentes precisarem capturar informações do ambiente, o uso de sensores é vital. Sensores são o elo de ligação entre o poder computacional e aplicações físicas;
- **Raciocínio**: sensores são o elo de ligação entre os algoritmos inteligentes e o mundo real no qual eles atuam. Uma grande quantidade de atividades de raciocínio devem ser efetuadas, incluindo a modelagem, o reconhecimento e predição de atividades, tomada de decisões e também o raciocínio espaço-temporal;
- **Predição de atividades e reconhecimento**: algoritmos de raciocínio devem ser

capazes de reconhecer e prever atividades que ocorrem no ambiente;

- **Tomada de decisões:** sistemas de Inteligência Ambiental devem ser desenvolvidos não apenas para lembrar o usuário do desenvolvimento de determinada tarefa, mas também de completar a tarefa para o usuário caso seja possível;
- **Raciocínio espacial e temporal:** muito pouco pode ser feito pelos sistemas de Inteligência Ambiental se não existir referências implícitas ou explícitas de quando um evento ocorreu. As percepções do ambiente em conjunto com outras, irão proporcionar importantes pistas a respeito das atividades que foram desenvolvidas pelos usuários e qual a resposta mais adequada que deve ser dada;
- **Atuação:** sistemas de Inteligência Ambiental são aplicados no mundo real pois são capazes de perceber o mundo e então agir de acordo com as decisões tomadas. Devem ser utilizadas tecnologias que sejam capazes de atuar no ambiente e também de interagirem adequadamente com os usuários;
- **Interfaces amigáveis:** uma das principais características que vai proporcionar que sistemas de Inteligência Ambiental sejam amplamente aceitos e utilizados pela sociedade é a facilidade de uso e convivência com estes. Existe o apelo pelo uso de *front-ends* amigáveis, preferencialmente utilizando interfaces naturais (como gestos ou comandos de voz), para as interações que ainda sejam necessárias com os usuários;
- **Dispositivos e sistemas seguros:** um benefício de sistemas com Inteligência Ambiental é a customização do ambiente pelos próprios usuários de acordo com suas necessidades. Isso faz com que o ambiente tenha maiores condições de executar as tarefas que forem definidas. Devem ser tomadas precauções quanto aos dispositivos existentes no ambiente, especialmente no que se refere à confiabilidade e disponibilidade dos dados de sensores, atuadores e demais dispositivos que interajam com os usuários. Tudo isso deve ser previsto pelos projetistas, através do uso de mecanismos de segurança e tolerância a falhas.

Requisitos tecnológicos

Para que os princípios fundamentais citados anteriormente possam ser empregados em Inteligência Ambiental, foram listados os seguintes requisitos tecnológicos:

- **Hardware não intrusivo:** os componentes de hardware utilizados devem ser pouco intrusivos. Sensores, atuadores e dispositivos devem ser preferencialmente embarcados ou então com interfaces que os tornem mais invisíveis possíveis na integração com o ambiente dos usuários;
- **Infraestrutura de comunicação móvel e fixa transparente:** a interoperabilidade de comunicação das redes heterogêneas deve acontecer de forma transparente. O núcleo das redes, responsáveis por alto tráfego de dados, deve usar fibra óptica, tecnologias que convirjam e permitam a reconfiguração dinâmica;
- **Redes dinâmicas massivamente distribuídas:** em ambientes inteligentes são utilizadas grandes diversidades de redes de dispositivos. Dessa forma, as redes precisam ser capazes de serem configuradas ad hoc, de acordo com a tarefa que está sendo desenvolvida a qual pode demandar um curto período de tempo e envolver diferentes atores e dispositivos. Os bancos de dados, centralizados ou distribuídos, devem ser acessíveis de qualquer local do sistema e o gerenciamento sobre a disponibilização e o armazenamento de dados deve ser efetuado de forma inteligente;
- **Interfaces semelhantes ao comportamento humano:** um desafio constante é a criação de sistemas que tenham formas intuitivas de uso. Técnicas de Inteligência Artificial podem ser empregadas de forma a auxiliar os sistemas a interagirem com os usuários. As interações entre os usuários e os sistemas podem se beneficiar pelo uso de reconhecimento de padrões como gestos ou voz;
- **Dependabilidade e segurança:** os ambientes devem ser seguros e tolerantes a falhas. É importante que o sistema seja capaz de lidar com situações de uso de forma incorreta por parte dos usuários;

4 LabVIEW

O LabVIEW (*Laboratory Virtual Instrument Engineering Workbench*) é um software criado pela *National Instruments* (NI) totalmente integrado aos sistemas operacionais Windows, Linux, Mac OS X e Solaris, de fácil utilização e com características que facilitam conexão a uma grande variedade de hardwares e de outros softwares. É um ambiente de desenvolvimento com um conjunto abrangente de ferramentas que lhe permite desenvolver aplicações de medição ou controle em menor tempo, criar inovações, fazer descobertas e obter resultados com maior rapidez [19].

Instrumentação Virtual

Uma das motivações para criação e uso do LabVIEW são os Instrumentos Virtuais. Instrumentos tradicionais de medição, como osciloscópios e geradores de forma de onda são muito poderosos, caros e projetados para desempenhar uma ou mais tarefas definidas pelo fornecedor. No entanto, o usuário geralmente não pode expandi-los ou personalizá-los, as funções disponíveis para o usuário são inerentes ao instrumento. A plataforma LabVIEW trabalha com o conceito de instrumentação virtual que consiste de um computador ou estação de trabalho (Workstation) equipado com um poderoso software aplicativo, hardware com custo otimizado, como placas plugin e drivers, que juntos desempenham as funções dos instrumentos tradicionais. Instrumentos virtuais representam uma revolução nos sistemas de instrumentação tradicionais focados em hardware para sistemas centrados em software que exploram o poder computacional, a produtividade, a visualização gráfica e as funcionalidades de conectividade dos populares computadores de mesa (PC desktops) e estações de trabalho.

O emprego do LabVIEW na criação de Instrumentos Virtuais é apenas parte do seu escopo de uso. Atualmente o LabVIEW vem sendo empregado como um poderoso processador de regras contextuais em sistemas que precisam de consciência de situação para controle e tomada de decisões.

Linguagem de Programação G

O LabVIEW se utiliza de uma linguagem de programação gráfica desenvolvida para aplicações

que são interativas, executadas em paralelo, multicore e que usa ícones (blocos de funções) no lugar de linhas de texto para criar aplicações. Estes blocos são interligados de forma a orientar o fluxo de dados. Em contraste com as linguagens baseadas em texto, onde as instruções determinam a execução do programa, o LabVIEW usa uma programação tipo fluxo de dados, onde este fluxo determina a execução.

Os programas em LabVIEW são chamados de instrumentos virtuais ou, simplesmente, IVs e estes não são processados por um interpretador, mas sim compilados, tornando a sua performance comparável à exibida pelas linguagens de programação de alto nível. Nos instrumentos virtuais há duas janelas para programação ilustradas nas figuras 5 e 6 e descritas abaixo:

- **Painel Frontal:** faz a interface com o usuário. Pode-se personalizar o painel com botões, mostradores e gráficos, a fim de simular instrumentos tradicionais, criar painéis com ensaios personalizados e representar visualmente o controle de operações e processos (vide Figura 5).
- **Diagrama de Blocos:** contém o código gráfico fonte que define as funcionalidades dos instrumentos virtuais. É possível determinar o comportamento dos instrumentos virtuais conectando ícones entre si para criar diagramas de blocos, que são notações de desenho usuais na engenharia (vide Figura 6).

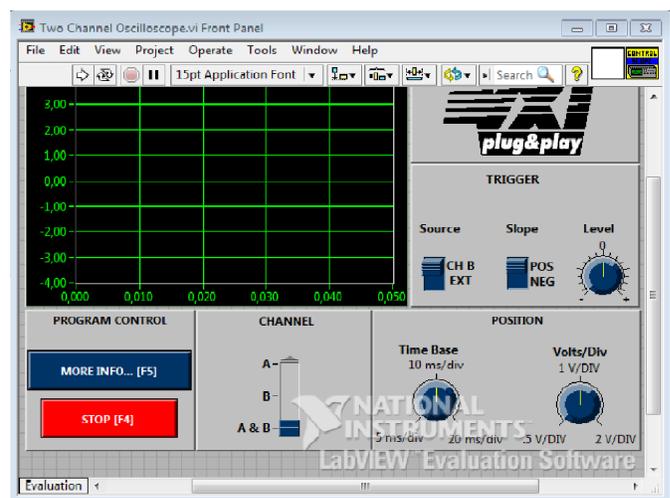


Figura 5. Painel Frontal

Devido a forma de programação a criação de processos paralelos no LabVIEW é facilitada em

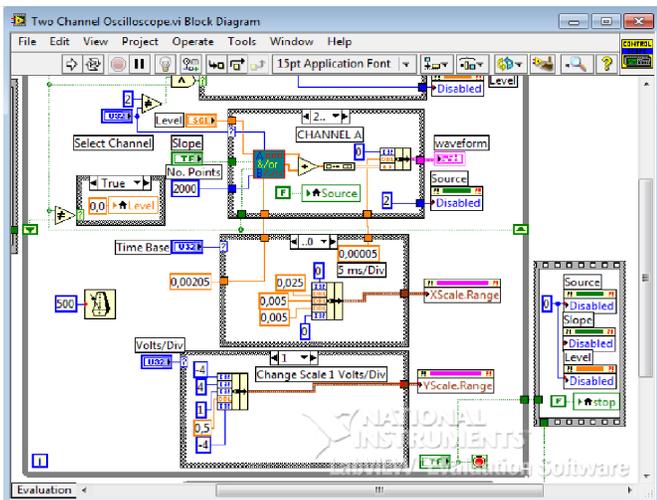


Figura 6. Diagrama de Blocos

consequência dos blocos não possuírem interdependência de dados.

Os dados gerados na saída dos blocos de função podem alimentar um outro bloco de função (gerando um novo processo) ou ainda serem armazenados em disco rígido, disponibilizados pela rede de dados ou enviados para impressão, já os blocos que não possuem interligação podem receber dados de equipamentos periféricos ou também gerarem os seus próprios dados (um exemplo é um gerador de números aleatórios).

Muitos blocos de função no LabVIEW são polimorfos, ou seja, a sua funcionalidade adapta-se aos tipos de dado que recebem. Por exemplo, a função Build-Array pode ser usada para a criação de quaisquer variáveis, ou seja, de strings, de inteiros e também de arrays e de clusters.

Os dados podem ser ligados ao Painel frontal (interface com o usuário) através de manipuladores. Por exemplo, a inserção de números pode ser dependente de um manípulo (caixa de entrada, seletor via mouse) e o estado de uma variável de saída booleana pode ser mostrado por um LED colocado no painel.

A plataforma LabVIEW possui aplicações e projetos em diversas áreas, dentre elas destacam-se: aquisição de dados e processamento de sinais, controle de instrumentos, sistemas embarcados de monitoramento e controle, automatização de sistemas de teste de validação, ensino acadêmico, servidor de consciência de situação aplicado em smart homes, dentre outras. Particularmente, este artigo explora o emprego do LabVIEW en-

quanto tecnologia para consciência de situação em ambientes com múltiplos sensores, heterogêneos quanto a natureza e perfil operacional.

5 Trabalhos relacionados

Diversas pesquisas têm sido feitas com o objetivo de desenvolver sistemas inteligentes em smart homes que se integram de forma transparente na vida cotidiana com o intuito de facilitar e melhorar o acesso a recursos pelos usuários de uma forma simples e flexível, utilizando a plataforma LabVIEW. Foram selecionados quatro artigos que abordam o tema e serão descritos nesta seção.

5.1 Real Time System Development for Home Automation using LabVIEW

Discute a automação de casas como uma aplicação da computação ubíqua em que o ambiente da casa é monitorado por inteligência ambiental para prestar serviços sensíveis ao contexto e facilitar o seu controle remoto, tendo como abordagem o desenvolvimento do sistema em tempo real, utilizando ferramenta de aquisição de dados e software gráfico avançado pertencentes a plataforma de desenvolvimento LabVIEW. Apresenta o conceito de automação residencial inteligente, em um esforço para reduzir o desperdício e consumo de energia [20].

O diagrama em blocos do sistema de monitoramento em tempo real é mostrado na figura 7. O sistema é constituído por um microcontrolador AVR Atmega16 usado para adquirir dados de diferentes tipos de sensores, modem GSM para obter informações em tempo real independentes de localização, através de aplicativos em dispositivos móveis como smartphones e para comunicação sem fio, além de quatro módulos (Módulo Principal, Sistema de Iluminação, Sistema de Alarme para Incêndio e Sistema para Detecção de Intruso).

O monitoramento dos parâmetros do sistema é realizado através de um display LCD 16X2 e pela interface de usuário gráfica LabVIEW GUI (*Graphical User Interface*). Parâmetros contextuais como luminosidade, temperatura, umidade e outros são coletados através dos sensores e entregues ao microcontrolador que realiza o interfaceamento através de comunicação serial, utilizando protocolo RS232, com o software LabVIEW para

fins de processamento, análise, armazenamento ou outra manipulação de dados.

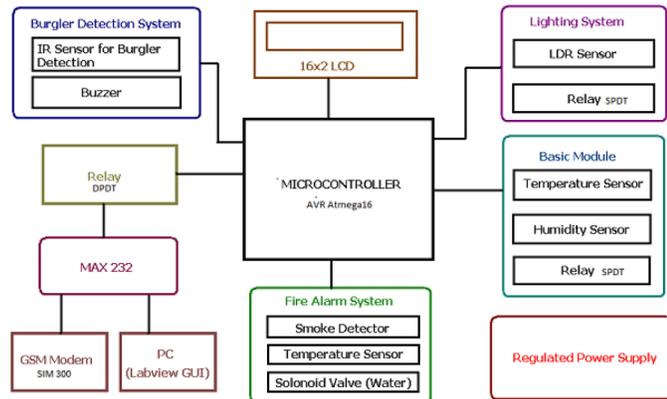


Figura 7. Diagrama de blocos mostrando diferentes parâmetros para o monitoramento em tempo real. Extraída de [20]

5.2 Smart Home Control using LabVIEW

Possui como objetivo mapear os processos de forma a otimizar as tecnologias empregadas em uma smart home, permitindo que o maior número de usuários possa ter acesso à tecnologia mais relevante para as suas necessidades com maior economia de energia e segurança utilizando para tal os recursos do software LabVIEW [21].

O controle de uma Smart Home utilizando LabVIEW é mostrado na figura 8. O diagrama de blocos da Smart Home mostra uma Smart House que consiste principalmente de equipamentos eletrônicos e elétricos, além de sensores e indicadores. Todos os subsistemas estão ligados ao software LabVIEW como a principal unidade de controle do sistema.

O LabVIEW possui várias entradas com sensores conectados e fornece uma saída lógica para o sistema de energia de toda a casa de acordo com o processamento definido pelo programa e variáveis de entrada (parâmetros contextuais). Realiza o controle da iluminação interna, iluminação externa, temperatura e alarmes de incêndio e roubo. Cada um desses componentes executa uma única função e não há sincronização com outros componentes. O sistema de controle possui características adicionais para proteção, que interrompe imediatamente a fonte de energia quando há ocorrência de alguma falha.

O monitor é usado para fazer alguma operação no sistema de conexão com o software LabVIEW. Ele é usado para exibir e alterar a carga em todos os cômodos da casa usando o receptor de cada unidade. O sistema também pode ser controlado manualmente devido o LabVIEW não ser um instrumento real e sim um instrumento virtual.

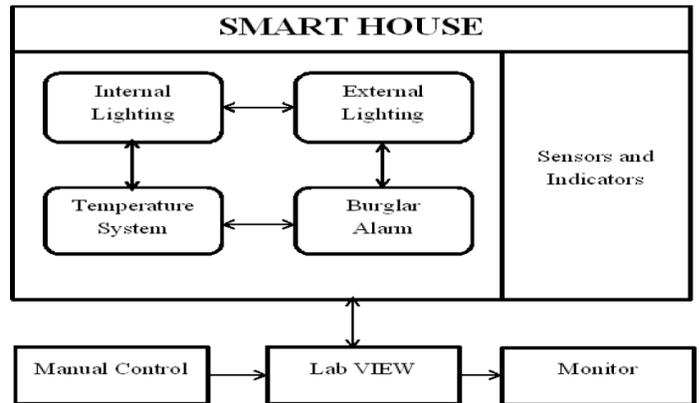


Figura 8. Diagrama de Blocos da Smart Home. Extraída de [21]

5.3 Design & Implementation of Smart House Control Using LabVIEW

Aborda a monitoração e controle de ambientes em uma casa como alguns dos recursos presentes em uma Smart House. Implementa um sistema de controle multiplataforma para automação baseado em hardware e software LabVIEW, além de um sistema individual de controle remoto. Pode atuar como um item de segurança da casa, pois possui sensor infravermelho e conexão com a internet para monitoração e controle remoto independente de localização [22].

A Smart House proposta para este artigo é formada por um sistema principal composto por cinco partes, sistemas de segurança, que incluem o sistema de alarme de incêndio e de sinalização da ocorrência de assalto, sistema de iluminação que inclui a iluminação interna e externa da casa, sistema de controle remoto da casa, sistema de controle de temperatura para condicionadores de ar e sistema de comutação de energia principal da casa. Estas cinco partes são conectadas ao software LabVIEW que controla todo o sistema. Possui duas interfaces, interface de computador e interface de unidade de controle remoto.

O software LabVIEW é a interface de computador que recebe dados dos sensores da casa, informações de processo e dados de atualizações para os sistemas de diferença e transmite o sinal de controle para os sistemas da casa. Além disso, o LabVIEW possui a capacidade de monitorar operações importantes no sistema para os usuários, com o intuito de informar as mudanças ocorridas neste. Os usuários também podem controlar o sistema e escolher melhores parâmetros que se adaptem as suas exigências.

A interface de controle remoto está disponível para controlar algumas aplicações na casa, e está também interligado ao software LabVIEW para outras aplicações. A Figura 9 mostra o diagrama de blocos da Smart House projetada para este trabalho.

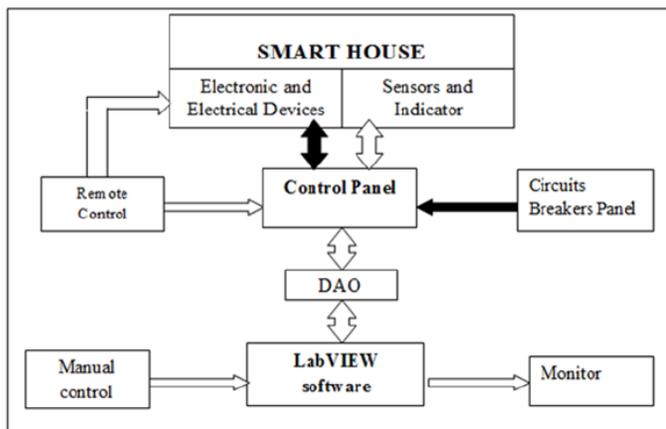


Figura 9. Diagrama de Blocos Smart House. Extraída de [22]

5.4 Implementation of Smart Home Control using LabVIEW and PDA

Implementa um programa de controle em PDA utilizando Wireless LAN como protocolo sem fio, realiza a modelagem de uma smart home e propõe funções úteis no controle das casas [23].

O artigo trata que os programas (instrumentos virtuais) construídos utilizando LabVIEW são mais fáceis de serem controlados e compartilhados através da Internet por possuírem componentes prontos como VI server, data Socket server and visual basic and active X, Java and Java Script que podem ser usados para projetar e desenvolver instrumentos virtuais habilitados para se

comunicarem pela internet em qualquer navegador web. Este recurso expande enormemente a aplicação, porque um operador pode executar um aplicativo de qualquer lugar. Além disso, possui a capacidade de vários utilizadores em diferentes locais acessarem o programa simultaneamente. O Módulo NI LabVIEW PDA possibilita o desenvolvimento de aplicações para dispositivos portáteis PDA. Com este módulo, pode-se desenvolver programas no LabVIEW e facilmente fazer download de aplicativos específicos para PDA.

Apresenta como vantagens em utilizar o software LabVIEW: o fato do desenvolvimento de programas ser de uma forma fácil e rápida por utilizar a linguagem de programação gráfica que é de fácil aprendizagem e compreensão; e não necessitar de ajustes de configuração e interface devido a incorporação de outros dispositivos. No entanto LabVIEW tem menos usuários do que em linguagem C, não é um software livre, além de possuir um custo elevado para aquisição de suas licenças.

6 Considerações Finais

O trabalho de pesquisa relatado neste artigo, teve como principal contribuição a sistematização de conceitos relevantes para o desenvolvimento de aplicações sensíveis ao contexto em Smart Homes, além de realizar análise sobre sistemas inteligentes que utilizam a plataforma LabVIEW para processamento de contexto.

O estudo dos trabalhos relacionados caracterizou que o software LabVIEW pode ser utilizado como processador de regras contextuais em ambientes inteligentes, conscientes de situação, porém apesar do potencial de aplicação na área de sensibilidade de situação, seu caráter comercial faz com que não estejam disponíveis informações sobre a concepção do mesmo, e sim sobre seu uso. Nesta perspectiva, o LabVIEW é tratado como uma ferramenta para desenvolver softwares (instrumentos virtuais) utilizados na automação de Smart Homes, de fácil programação, manipulação e integração com outros sistemas, tendo a desvantagem de não ser um software livre.

Referências

- [1] John Krumm. Ubiquitous computing fundamentals, 2010.

- [2] M. Knappmeyer, S.L. Kiani, E.S. Reetz, N. Baker, and R. Tonjes. Survey of context provisioning middleware. *IEEE Communications Surveys & Tutorials*, 15(3):1492–1519, Third Quarter 2013.
- [3] J. Alves Lino, B. Salem, and M. Rauterberg. Responsive environments: User experiences for ambient intelligence. *Journal of Ambient Intelligence and Smart Environments*, 2(4):347–367, 2010.
- [4] Bill N. Schilit, Norman I. Adams, and Roy Want. Context-aware computing applications. In *Proceedings of IEEE Workshop on Mobile Computing Systems and Applications*, pages 85–90. IEEE Computer Society, December 1994.
- [5] Guanling Chen and David Kotz. A survey of context-aware mobile computing research, 2000.
- [6] Mikko Perttunen, Jukka Riekkki, and Ora Lassila. Context representation and reasoning in pervasive computing: a review. *International Journal of Multimedia and Ubiquitous Engineering*, 4(4):1–28, Oct 2009.
- [7] Carlos Oberdan Rolim. Estudo sobre raciocínio e predição de contexto voltados à inteligência ambiental, December 2011.
- [8] Gregory D. Abowd, Anind K. Dey, Peter J. Brown, Nigel Davies, Mark Smith, and Pete Steggles. Towards a better understanding of context and context-awareness. In *Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing*, pages 304–307, London, UK, UK, 1999. Springer-Verlag.
- [9] Henry Lieberman and Ted Selker. Out of context: Computer systems that adapt to, and learn from, context. *IBM Systems Journal*, 39, 2000.
- [10] Philip D. Gray and Daniel Salber. Modelling and using sensed context information in the design of interactive applications. In *Proceedings of the 8th IFIP International Conference on Engineering for Human-Computer Interaction*, pages 317–336. Springer-Verlag, 2001.
- [11] Adenauer Corrêa Yamin, Jorge L. V. Barbosa, Iara Augustin, Luciano Cavalheiro da Silva, Rodrigo Araújo Real, Cláudio Fernando Resin Geyer, and Gerson G. H. Cavalheiro. Towards merging context-aware, mobile and grid computing. *IJHPCA*, 17(2):191–203, 2003.
- [12] Philip D. Gray and Daniel Salber. Modelling and using sensed context information in the design of interactive applications. In *Proceedings of the 8th IFIP International Conference on Engineering for Human-Computer Interaction*, pages 317–336. Springer-Verlag, 2010.
- [13] Claudio Bettini, Oliver Brdiczka, Karen Henriksen, Jadwiga Indulska, Daniela Nicklas, Anand Ranganathan, and Daniele Riboni. A survey of context modelling and reasoning techniques. *Pervasive Mob. Comput.*, 6(2):161–180, April 2010.
- [14] Christian Hoareau and Ichiro Satoh. Modeling and processing information for context-aware computing: A survey. *New Generation Comput.*, 27(3):177–196, 2009.
- [15] Thomas Strang and Claudia Linnhoff-Popien. A context modeling survey. In *In: Workshop on Advanced Context Modelling, Reasoning and Management, UbiComp 2004 - The Sixth International Conference on Ubiquitous Computing, Nottingham/England*, 2004.
- [16] Gabriella Castelli, Alberto Rosi, Marco Mamei, and Franco Zambonelli. A simple model and infrastructure for context-aware browsing of the world. In *PerCom*, pages 229–238. IEEE Computer Society, 2007.
- [17] Jeannette Chin, Victor Callaghan, and Graham Clarke. End-user customisation of intelligent environments. In Hideyuki Nakashima, Hamid Aghajan, and JuanCarlos Augusto, editors, *Handbook of Ambient Intelligence and Smart Environments*, pages 371–407. Springer US, 2010.
- [18] G. Riva, F. Davide, and W.A. IJsselsteijn. *Being There: Concepts, Effects and Measurements of User Presence in Synthetic Environments*. Emerging communication : studies in new technologies and practices in communication. IOS Press, 2002.
- [19] National Instruments. Labview: Uma plataforma de desenvolvimento de sistemas para medição e controle, 2014. Disponível em: <http://www.ni.com/labview/pt/>. Acesso em Janeiro 2014.
- [20] A.J. Patil, Rajesh R. Karhe, and Mahesh S.Patil. Real time system development for home automation using labview. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 2:1213–1219, April 2013.
- [21] Akshatha N Gowda, Girijamba D L, Rishika G N, Shruthi S D, and Niveditha S. Smart home control using labview. *International Journal of Advanced Research in Computer Science and Electronics Engineering*, 2(5):485–490, May 2013.
- [22] Basil Hamed. Design & implementation of smart house control using labview. *International Journal of Soft Computing and Engineering*, 1:98–106, January 2012.
- [23] Sang-Min Baek, Kyung-Bae Chang, Il-Joo Shim, and Gwi-Tae Park. Implementation of smart home control using labview and pda. *IEEE*, pages 558–562, 2004.



Maicon Ança dos Santos Possui especialização em Administração de Banco de Dados e graduação em Tecnologia em Análise e Desenvolvimento de Sistemas pela Faculdade de Tecnologia SENAC/Passo Fundo (2008). Atualmente é Analista de Tecnologia da Informação da Diretoria de Tecnologia da Informação e Comunicação, no Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense, Reitoria. Tem experiência na área de Ciência da Computação, com ênfase em Redes de Computadores e Máquinas Agregadas.



Patrícia Teixeira Davet Possui graduação em Engenharia Eletrônica pela Universidade Católica de Pelotas (2012). Atualmente trabalha na empresa Enterprise Telecomunicações LTDA atuando nos segmentos de centrais telefônicas, tecnologia Voip e softwares de tarifação. Tem interesse na área de Engenharia Eletrônica, com ênfase em Computação Autônoma e Instrumentação

Virtual Ubíqua.