

**UNIVERSIDADE DO EXTREMO SUL CATARINENSE - UNESC
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

DIOVANA DA SILVA SANTANA

**MODELAGEM DE CARACTERÍSTICAS PARA REPRESENTAR INFORMAÇÕES
DE CONTEXTO RELEVANTES EM UM DOMÍNIO DE APLICAÇÃO SENSÍVEL A
MUDANÇAS EM TEMPO DE EXECUÇÃO**

CRICIÚMA

2018

DIOVANA DA SILVA SANTANA

**MODELAGEM DE CARACTERÍSTICAS PARA REPRESENTAR INFORMAÇÕES
DE CONTEXTO RELEVANTES EM UM DOMÍNIO DE APLICAÇÃO SENSÍVEL A
MUDANÇAS EM TEMPO DE EXECUÇÃO**

Trabalho de Conclusão de Curso, apresentado para obtenção do grau de Bacharel no curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC.

Orientadora: Prof.^a Ma. Leila Laís Gonçalves

CRICIÚMA

2018

DIOVANA DA SILVA SANTANA

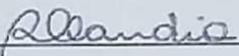
**MODELAGEM DE CARACTERÍSTICAS PARA REPRESENTAR
INFORMAÇÕES DE CONTEXTO RELEVANTES EM UM DOMÍNIO DE
APLICAÇÃO SENSÍVEL A MUDANÇAS EM TEMPO DE EXECUÇÃO**

Trabalho de Conclusão de Curso aprovado pela Banca Examinadora para obtenção do grau de Bacharel no Curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC, com linha de pesquisa em Engenharia de Sistemas e Sistemas Multimídia.

Criciúma, 29 de novembro de 2018

BANCA EXAMINADORA


Prof.^a Leila Laís Gonçalves – Ma. - UNESC - Orientador


Prof.^a Ana Cláudia Garcia Barbosa - Ma - UNESC


Prof. Gustavo Bisognin – Me - UNESC

Aos meus pais e irmãos

AGRADECIMENTO

Primeiramente gostaria de agradecer a Deus por toda força que me deu para continuar, vencer os obstáculos e desafios que surgiram, mesmo nos momentos que pensamos em desistir ou não acreditamos que é possível.

Agradeço aos meus pais e irmãos, em especial à minha mãe, Severina, por acreditar em meu potencial, me apoiar incondicionalmente, e sempre estar ao meu lado; sua companhia foi fundamental a cada fase; obrigada por tudo Mãe.

Agradeço à Professora MSc. Leila pela orientação e atenção, ajudando a conduzir este trabalho da melhor forma possível. Agradeço também aos professores membros da banca e aos demais professores, que, através de suas aulas e trabalhos, me fizeram crescer acadêmica e profissionalmente.

E por fim, a todos os meus amigos e a todos que em mim confiaram e apoiaram desde o primeiro momento.

“O insucesso é apenas uma oportunidade para recomeçar de novo com mais inteligência.” Henry Ford

RESUMO

Este estudo trata da modelagem de características para representar informações de contexto em um domínio de aplicação sensível. O assunto é relevante, pelo fato de discutir um tema em evidência na era da computação móvel. Com o aumento da utilização de dispositivos móveis, surge também diferentes aplicações computacionais com a necessidade de realizar tarefas complexas e processar grande volume de informação em pouco tempo. Devido a esta necessidade, surgem desafios de interpretação e uso das informações para desenvolvimento de aplicações sensíveis ao contexto. Portanto, o trabalho tem o objetivo de desenvolver um modelo de características para representar informações de contexto relevantes em um domínio de aplicação sensível às mudanças em tempo de execução. A metodologia utilizada é a revisão bibliográfica de obras publicadas entre os anos de 1997 à 2017 a fim de se obter um embasamento teórico necessário sobre computação sensível ao contexto e alguns exemplos de aplicações que utilizam este conceito. Também se enfatiza o estudo de modelagem de contexto, o levantamento de informações para modelagem de características e a variabilidade, apresenta os trabalhos correlatos, enfatizando a experiência do usuário em sistemas computacionais e se versa sobre a elaboração do modelo que foi realizado, a estruturação das informações contextuais, apontando a metodologia utilizada junto ao cronograma. O trabalho conclui que no desenvolvimento de uma nova aplicação, existe uma alta expectativa de reusabilidade, implicando em redução de tempo e custo de projeto, sem queda na qualidade dos trabalhos.

Palavras-chave: Sistema Sensível ao Contexto. Modelagem de características. Domínio de aplicação sensível ao contexto

ABSTRACT

This study considers characteristics modeling to represent context information in an application domain that is sensitive. The subject is relevant, for the fact of discussing an evidence topic in the Mobile Computing Era. With the increase in the use of mobile devices, the need of different computational applications arise to perform complex tasks and process great amounts of information in a short time. Due to this need, interpretation and use of information challenges increase from the interpretation to develop context sensitive applications. Therefore, the work aims to develop a model of characteristics to represent relevant context information in an application domain which is sensitive to changes at run time. The methodology used is the bibliographic review of papers published between 1997 and 2017, in order to obtain a necessary theoretical basis on context sensitive computation and some examples of the use the concept of Context Sensitive System applications. It also emphasizes the study of context modeling, that shows in related works finding of information for modeling of characteristics and variability which emphasizes the userexperience in computational systems and it is focused on the elaboration of the model that was performed , the structuring of contextual information, pointing out the methodology used along the schedule. The work concludes that in the development of a new application, there is a high expectation of reusability, implying a reduction of time and cost of design, without a decrease in the quality of the works.

Keywords: Context Sensitive System. Characteristics Modeling. Field of context sensitive application

LISTA DE ILUSTRAÇÕES

Figura 1 - O relacionamento entre a Computação Sensível ao Contexto e outras áreas da Computação Geral	22
Figura 2 - Alguns fatos importantes sobre a evolução da Computação Sensível ao Contexto.....	23
Figura 3 - Aspectos importantes contexto.	27
Figura 4 - Aplicação Tradicional x Aplicação Sensível ao contexto.....	29
Figura 5 – Arquitetura conceitual sensível ao contexto.	33
Figura 6 - Técnica par de chave-valor.	43
Figura 7 - Técnica de modelos gráficos.	43
Figura 8 – Exemplo de ontologia para modelagem de contexto.....	44
Figura 9 - Aplicações de Ontologia.	45
Figura 10 - Exemplo de grafo contextual para a tarefa Reservar Transporte.....	46
Figura 11 - Visão geral do gerenciamento de contexto.	48
Figura 12 - Comunicação através de middleware.	49
Figura 13 - Arquitetura Socam.	52
Figura 14 - Arquitetura Gaia.....	54
Figura 15 – Arquitetura geral da plataforma Infraware.....	55
Figura16 – Grafo Contextual.	58
Figura 17 - Diagrama de blocos do projeto de arquitetura do sistema.	60
Figura 18 - Modelo do método de seleção de sensores.....	63
Figura 19 - Arquitetura da solução.	67
Figura 20 - MCSS: MODELO DE CARACTERÍSTICAS.....	73

LISTA DE QUADROS

Quadro 1 - Comparativo entre técnicas de representação de contexto.	46
Quadro 2 - Classificação das informações contextuais.	48
Quadro 3 – Regra Contextual #1.....	58
Quadro 4 – Regra Contextual #2.....	59
Quadro 5 – Regra Contextual #3.....	59
Quadro 6 - Comparativo entre os correlatos.	69

LISTA DE ABREVIATURAS E SIGLAS

API	<i>Application Programming Interface</i>
CSS	Sistemas Sensíveis ao contexto (do inglês Context-Sensitive System)
ER	Entidade-relacionamento
GPS	<i>Global Positioning System</i>
IA	Inteligência Artificial
IHC	Interação humano-computador
IoT	Internet das Coisas (do inglês <i>Internet of Things</i>)
IU	Interface de usuário
LPS	Linha de Produto de Software
MC	Modelagem de Características
NMEA	<i>National Marine Electronics Association</i>
OVM	Modelos de Variabilidade Ortogonais
SGBD	Sistema de Gerenciador de Banco de Dados
SOCAM	<i>Service Oriented Context Aware Middleware</i>
UML	Linguagem de Modelagem Unificada
XML	<i>Extensible Markup Language</i>

SUMÁRIO

1 INTRODUÇÃO	14
1.1 OBJETIVO GERAL	15
1.1.1 Objetivos específicos	16
1.2 JUSTIFICATIVA	16
1.4 ESTRUTURA DO TRABALHO.....	17
2 REVISÃO DE LITERATURA	18
2.1 COMPUTAÇÃO SENSÍVEL AO CONTEXTO	18
2.1.1 Panorama	21
2.1.2 Sistemas Sensíveis ao Contexto	25
2.1.3 Exemplos de Aplicações	28
2.1.3.3 Amazon	31
2.1.3.4 Gmail	32
2.2 MODELAGEM DE SISTEMAS SENSÍVEIS AO CONTEXTO	32
2.2.1 Elementos para modelagem de contexto	34
2.2.2 Características da informação de contexto	35
2.2.3 Abordagens de representação de contexto	42
2.2.4 Monitoramento de contexto	47
2.2.5 Middlewares	48
3 TRABALHOS CORRELATOS	54
3.1 DESENVOLVIMENTO DE UMA SOLUÇÃO SENSÍVEL AO CONTEXTO COMO SUPORTE A UM SISTEMA DE TRANSPORTE PÚBLICO INTELIGENTE	57
3.2 SELEÇÃO DE SENSORES SENSÍVEL A CONTEXTO PARA REDUÇÃO DO CONSUMO DE ENERGIA EM SMARTPHONES.....	61
3.3 UBIFEX: UMA ABORDAGEM PARA MODELAGEM DE CARACTERÍSTICAS DE LINHA DE PRODUTOS DE SOFTWARES SENSÍVEIS AO CONTEXTO	63
3.4 RECOMENDAÇÃO ADAPTATIVA E SENSÍVEL AO CONTEXTO DE RECURSOS PARA USUÁRIOS EM UM CAMPUS UNIVERSITÁRIO.....	65
3.5 COMPARATIVO DOS TRABALHOS CORRELATOS.....	68
4 MCSS: MODELO DE CARACTERÍSTICAS PARA REPRESENTAR INFORMAÇÕES DE CONTEXTO EM UM DOMÍNIO DE APLICAÇÃO	70
4.1. METODOLOGIA.....	70

4.2 LEVANTAMENTO BIBLIOGRÁFICO	71
4.2.1 DESCRIÇÃO DO MODELO	71
4.2.2 ESPECIFICAÇÃO DE USO	74
5 RESULTADOS E DISCUSSÃO	75
6 CONCLUSÃO	81
REFERÊNCIAS.....	84

1 INTRODUÇÃO

Nossas atividades rotineiras envolvem alguma forma de comunicação. Quando dois indivíduos se comunicam, de maneira natural, observam o contexto que essa comunicação acontece. Para isso, leva-se em consideração a linguagem que se compartilha, o entendimento comum de como o mundo funciona e o entendimento implícito das situações rotineiras dos seres humanos (VIEIRA; TEDESCO; SALGADO, 2009).

Atualmente, os usuários necessitam esforçar para informar preferências de uso, indicar suas necessidades, respondendo perguntas, configurando parâmetros sobre como um serviço deve se apresentar, transmitindo ideias sobre a atividade realizada, entre outros (VIEIRA; TEDESCO; SALGADO, 2009). O ideal seria que os sistemas computacionais percebessem a intenção do usuário minimizando sua complexidade e esforço na execução das tarefas. Para este cenário é necessário que as aplicações se adaptem no tempo real onde são executadas, para que assim possam fornecer resultados adequados para cada situação, tendo em consideração o contexto onde estão sendo executados. Sem a informação de contexto não é possível às aplicações se adaptarem ao seu comportamento e responderem de forma personalizada (RIBEIRO et al., 2013).

Mesmo o contexto sendo um conceito presente e conhecido nas interações pessoais, ainda é pouco explorado nos sistemas computacionais. Com o aumento do uso de dispositivos móveis, surge diferentes aplicações computacionais com a necessidade de realizar tarefas complexas e processar grande volume de informação em pouco tempo, isso apresenta desafios de interpretação e uso das informações para desenvolvimento de aplicações sensíveis ao contexto (SILVA, 2014).

A necessidade surgiu da percepção de projetistas de sistemas computacionais sobre uma demanda na necessidade dos usuários. É desejo de quem opera um sistema que seja ele adaptável, flexível, fácil de usar, enfim, interativo. Sistemas que exigem demasiadamente intervenção do usuário tendem a perder espaço para aqueles que oferecem serviços mais adaptáveis às nossas necessidades e intenções. A partir desta motivação, os desenvolvedores têm-se

confrontado com o desafio de diminuir a necessidade do usuário interagir com o sistema e tornar os sistemas adaptáveis aos usuários com informações e serviços relevantes às suas necessidades no tempo desejado (PINHEIRO, 2014).

Sistemas sensíveis ao contexto são sistemas que gerenciam dados referentes a uma aplicação em um domínio e utilizam esses elementos com a integração da tecnologia em ambientes físicos, de forma clara para apoiar as pessoas na execução das suas tarefas diárias. As áreas da Computação Ubíqua e Inteligência Artificial foram as pioneiras nos estudos e uso do conceito de contexto (LINDER et al., 2007), evidenciando a capacidade da aplicação desse conceito nos sistemas computacionais.

Os benefícios do sistema sensível ao contexto é que tem capacidade de fornecer serviços adaptáveis às necessidades e intenções do usuário, que permite adaptar as respostas de acordo com as preferências do usuário, a sua interação com o sistema, o momento em que a consulta está sendo realizado, o dispositivo utilizado, entre outras características (VIEIRA; TEDESCO; SALGADO, 2009). A compreensão de como o contexto pode ser usado ajudará os desenvolvedores a determinar quais comportamentos sensíveis ao contexto serão utilizadas em aplicações.

Aplicações precisam ser inteligentes e utilizar os recursos disponíveis dinamicamente, tomando decisões de acordo com a situação. É importante que seja apresentado ao usuário geralmente o mesmo ambiente, as mesmas aplicações e os mesmos dados, não importando de onde ou de qual dispositivo esteja utilizando o sistema.

Nos dias atuais, cresce o interesse no desenvolvimento de aplicações sensíveis ao contexto. No entanto, entender e identificar o contexto, e executar ações de maneira automática segundo não é uma tarefa comum em sistemas computacionais. Neste sentido, este trabalho propõe um modelo de características para auxiliar o desenvolvimento de aplicações sensíveis ao contexto.

1.1 OBJETIVO GERAL

Desenvolver um modelo de características para representar informações

de contexto relevantes em um domínio de aplicação sensível às mudanças em tempo de execução.

1.1.1 Objetivos específicos

Os objetivos da pesquisa consistem em:

- a) analisar sistemas sensíveis, observando elementos, comportamento nas variações de contexto e o suporte no desenvolvimento de aplicações;
- b) identificar elementos que representam informações de contexto e situações de mudança para compor o modelo de características em um domínio;
- c) descrever técnicas para modelagem de características que representam informações de contexto, e
- d) definir mecanismos para verificação das mudanças de contexto com base no modelo de características.

1.2. JUSTIFICATIVA

Por se tratar da relevância pertinente a evolução da computação móvel. Segundo Anjos (2006), cada vez mais, as aplicações móveis devem se adaptar ao contexto do usuário, fornecendo informações e serviços atualizados.

As áreas da Computação Ubíqua e Inteligência Artificial foram às pioneiras nos estudos e utilização do conceito de contexto e, com isso, foram as que demonstraram o potencial da aplicação desse conceito nos sistemas computacionais (VIEIRA, 2009).

Pesquisas vêm utilizando o conceito de contexto para beneficiar sistemas ligados a outras áreas como, por exemplo: sistemas colaborativos, visando melhorar os serviços de percepção e prover assistência aos grupos; hipermídia adaptativa, possibilitando a personalização e adaptação do conteúdo de sites web; integração de dados, facilitando a resolução de conflitos semânticos; e Interação Humano-Computador (IHC), com o uso do contexto para adaptar as interfaces dos sistemas

tornando mais intuitiva a sua interação com os usuários (VIEIRA, 2009).

Computação ciente de contexto é um paradigma computacional no qual aplicações podem descobrir e utilizar informações contextuais, tal como a localização de um usuário, pessoas ou dispositivos próximos e a atual atividade para prover serviços apropriados para uma determinada pessoa, num determinado local, em um dado momento (ANJOS, 2006).

Dentre várias ferramentas disponibilizadas na internet, encontramos inúmeros softwares que buscam atender as necessidades das pessoas, desde localizar amigos e conhecer novas pessoas, como compartilhar ideias e gostos em comum. Com base nas informações contextuais, a aplicação pode enriquecer semanticamente a solicitação explícita do usuário e, com isso, executar serviços mais próximos às suas necessidades.

A modelagem de sistemas auxilia a entender a funcionalidade do sistema e os modelos que são usados de comunicação com os clientes. Um problema referente aos modelos de contexto usados é que, para a descrição de cenários dinâmicos, é preciso que as estruturas desses modelos precisam ser atualizadas. Assim, atualizações em modelos de contexto implicam no (re)desenvolvimento das implementações correspondentes a esses modelos (MARINHO, 2012).

Um modelo contextual identifica e representa as entidades e os contextos que são relevantes para a aplicação sensível ao contexto, como por exemplo, a localização e a temperatura de uma pessoa. Um modelo de situações é construído com base no modelo de contexto e permite ao desenvolvedor explicitar situações que são de interesse da aplicação (MIELKE, 2012).

1.4. ESTRUTURA DO TRABALHO

O presente trabalho é composto pelos seguintes. No capítulo 2 são discutidos pontos da pesquisa que são relevantes para o trabalho. Os trabalhos correlatos são apresentados no capítulo 3, como um comparativo entre eles e as principais características. No capítulo 4 apresentamos o modelo, e suas principais características. O capítulo 5 discute resultados e discussão. Por fim, o capítulo 6 contém a conclusão do trabalho e aponta os trabalhos futuros.

2 REVISÃO DE LITERATURA

O presente trabalho apresenta a metodologia de pesquisa que direcionou o desenvolvimento deste capítulo e os conceitos das áreas de pesquisa envolvidas, incluindo: Computação sensível ao contexto onde relata a origem e evolução, sistemas sensíveis ao contexto e exemplos de aplicações, Modelagem de sistema sensíveis ao contexto, destacando os elementos, características, abordagens de representação, monitoramento de contexto, middleware e trabalhos correlatos.

2.1 COMPUTAÇÃO SENSÍVEL AO CONTEXTO

A maioria dos sistemas computacionais não conta com informação contextual a ponto de conceder serviços voltados às necessidades do usuário. Tais aplicações recebem somente informações fornecidas diretamente pelo usuário para executar transações com fluxos de movimentos já pré-determinados e redução da dinamicidade em relação aos comportamentos tomados pelos usuários (SOARES, 2010).

Sistemas que utilizam contexto são sensíveis ao meio que os cerca e capazes de realizar ações específicas de acordo com as informações necessárias para cada tarefa, são ditos sistemas inteligentes, pois conseguem adaptar-se ao ambiente. As informações são disponibilizadas aos sistemas pelo usuário bem como aqueles disponíveis em uma base de conhecimento contextual, as inferidas por um processo de raciocínio e aquelas que são percebidas a partir do ambiente (VIEIRA et al., 2006).

Antes de conceituar Computação Sensível ao Contexto, é necessário compreendermos a ideia de contexto sob olhar da computação. Sá (2012) afirma que a ideia de contexto tem sido aplicada em diversas áreas, inserindo linguística, filosofia, representação do conhecimento, teoria da comunicação e resolução de problemas no campo da inteligência artificial. O autor utiliza-se do conceito de Theimer e Schilit, o qual referem-se ao contexto como sendo "a informação de localização, a identidade das pessoas e objetos localizados e ainda as mudanças nesses objetos" (SÁ, 2012, p. 393).

Contexto pode ter diversas definições, pois se trata de um termo amplo e que abrange diversas áreas. Segundo Dey (2001), contexto é qualquer informação que pode ser aplicada para identificar a situação de uma entidade que seria uma pessoa, lugar ou objeto, tornando essa informação importante para a comunicação de um usuário e aplicativo.

Para Vieira e Souza (2006) contexto pode ser determinado como uma forma de comunicação entre um usuário e aplicação, tornando-se capaz de executar alguma tarefa, e o grupo de elementos contextuais necessários para a tarefa ser executada

Contexto abrange mais do que apenas a localização do usuário, porque outras coisas de interesse também são móveis e estão sempre em mudança, como iluminação, nível de ruído, conectividade de rede, os custos de comunicação, largura de banda, e até mesmo a situação social; por exemplo, se você está com o seu gerente ou com um colega de trabalho (ADAMS; SCHILIT; WANT, 1994, tradução nossa).

O contexto é importante, pois contribui na comunicação dos sistemas computacionais e usuários, diminuindo os conflitos entre as aplicações e aperfeiçoamento dos serviços, deixando as aplicações amigáveis, flexíveis e fáceis de usar (VIEIRA; SANTOS, 2006).

Dourish (2004) delimita seu conceito de acordo com uma classificação partindo de sua origem na computação pervasiva. Segundo ele, classifica-se em duas vertentes, a primeira, o contexto é encarado como um problema representacional, da mesma forma que sistemas de software são representacionais, por esse motivo, uma das preocupações é como o contexto pode ser codificado e representado. Para isso, Dourish (2004) delimitou quatro suposições:

- a) contexto é uma forma de informação, ou seja, é algo que pode ser conhecido e por isso codificado e representado;
- b) contexto é delineável. Para um dado conjunto de aplicações ou requisitos, pode-se definir antecipadamente o que irá contar como contexto de atividades que a aplicação executa;
- c) contexto é estável. Apesar dos elementos da representação de contexto variarem de aplicação para aplicação, eles não variam de

instância para instância de uma atividade ou um evento pode ser realizada uma única vez.

- d) contexto e atividade são separáveis. As atividades ocorrem dentro de um contexto. O contexto, por sua vez, descreve características do ambiente onde a atividade ocorre, mas são separadas da atividade. Uma pessoa pode estar conversando com outra em uma localização, então a conversa é a atividade da pessoa e a localização é um dos aspectos do contexto.

Segundo o autor, essas hipóteses são com base no contexto na área de computação pervasiva. Apresenta a ideia de que o contexto se baseia num grupo de características de um ambiente que envolve uma atividade genérica, e que essas características podem ser codificadas e torna acessíveis a um sistema de software juntamente com uma codificação da própria atividade, é uma suposição comum em vários sistemas.

No paradigma de computação sensível ao contexto (*context-aware computing*), é apresentado um ambiente em que diferentes sensores, computadores e recursos de gravação de som e imagem estão conectados e, ativamente, monitorando mudanças no ambiente (ROCHA, 2010).

Tais dispositivos de sensoriamento e computadores colaboram, de forma interativa e proativa, a fim de facilitar a interação homem-computador ou prover informações detalhadas do que ocorre no ambiente. Em tal paradigma, o ambiente é considerado "inteligente", visto que este é equipado com sensores com a finalidade de estar ciente das mudanças nesses ambientes. Desta forma, os dados fornecidos por sensores ou outros dispositivos que são capazes de capturar informações do meio em que operam, são chamados de dados contextuais, visto que eles contêm informações sobre o contexto em que cada entidade se encontra (TONILLI et al., 2009).

Os processadores disponíveis para a tarefa, os dispositivos acessíveis para visualização do usuário de entrada e, a capacidade de rede, conectividade e custos podem mudar ao longo do tempo e lugar. Em suma, a configuração de hardware está mudando continuamente. Da mesma forma, o usuário de computador pode se mover de um ponto para outro, juntar grupos de pessoas, e frequentemente

agir com computadores (SILVA et al., 2015).

O avanço da computação móvel destaca-se com o aumento no poder computacional, os recursos como sensores, *Global Positioning System* (GPS), Bluetooth, o processamento e a capacidade de armazenamento, onde é possível a execução de softwares complexos e aplicativos inteligentes (BEZERRA, 2015). Com o desenvolvimento dessas tecnologias e a presença delas nas nossas vidas, foi-se o tempo em que somente computadores pessoais ou de grande porte eram capazes de processar informações (BARBOSA; SILVA, 2010).

Um desafio de computação móvel é explorar a mudança de ambiente com uma nova classe de aplicativos que são cientes do contexto em que eles são executados. É possível encontrar diversos eletrônicos ditos inteligentes, que se configuram automaticamente de acordo com o ambiente que estão, tais como máquinas fotográficas que se adaptam conforme a luminosidade; o grau de proximidade de outros usuários; mudanças de temperatura. Esta é a forma como estes conjuntos de dispositivos móveis estão se comunicando e cooperando em nome do usuário e de um CSS (BARBOSA; SILVA, 2010).

2.1.1 Panorama

A trajetória da Computação Ubíqua tem origem na década de noventa quando Mark Weiser, cientista-chefe da Xerox PARC nos Estados Unidos, publicou um artigo na revista *Scientific American* em setembro de 1991, chamado “*The Computer for the Twenty-First Century*”, que indicava inúmeras afirmações sobre o futuro da computação (LIMA; GUERRA; FIORIN, 2015).

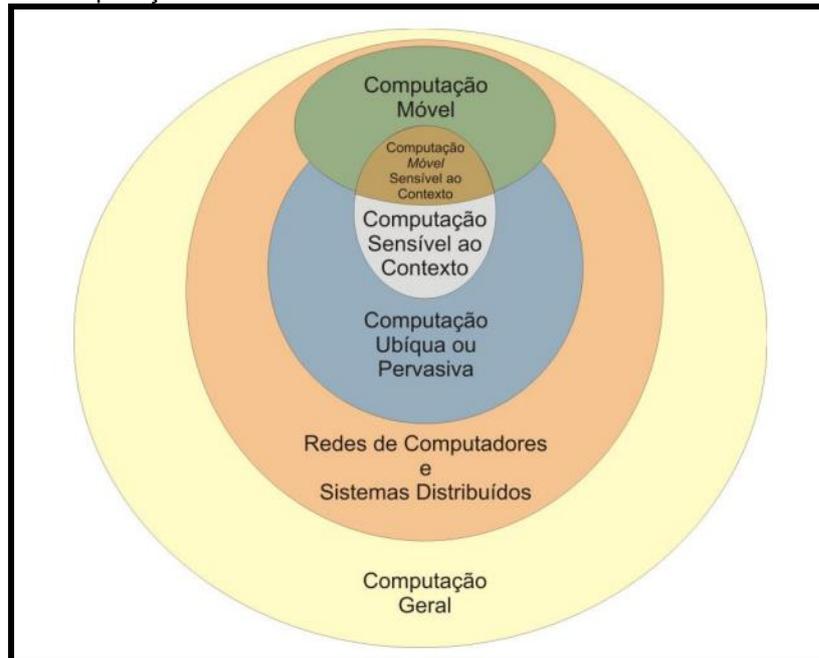
As áreas da Computação Ubíqua e Inteligência Artificial (IA) foram as pioneiras nos estudos e utilização do conceito de contexto e, com isso, foram as que demonstraram o potencial da aplicação desse conceito nos sistemas computacionais (VIEIRA; TEDESCO; SALGADO, 2009).

A Computação Sensível ao Contexto é uma subárea dentro de uma área maior denominada de Computação Ubíqua ou Pervasiva¹ que, encontra-se situada

¹ Ou computação pervasiva é um termo usado para descrever a onipresença da informática no cotidiano das pessoas. (Em inglês: Ubiquitous Computing ou ubicomp).

na subárea da Computação Geral designada Redes de Computadores e Sistemas Distribuídos. Ainda, a união da computação sensível ao contexto com a computação móvel origina uma promissora e nova área, conhecida por computação móvel Sensível ao Contexto (figura 1).

Figura 1 - O relacionamento entre a Computação Sensível ao Contexto e outras áreas da Computação Geral



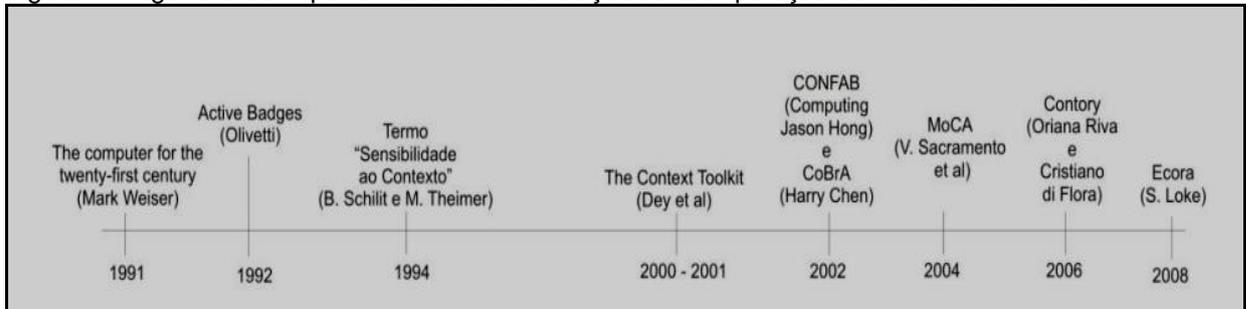
Fonte: Sá (2012).

Weiser (1991) em seu artigo, previa um aumento significativo em termos de disponibilidade e funcionalidade de serviços computacionais para os usuários finais, o qual afirmava que a computação tornar-se-ia pervasiva e onipresente na vida das pessoas.

[...] estamos tentando conceber uma nova maneira de pensar sobre computadores no mundo, um que leva em conta o ambiente humano natural e permite que os próprios computadores desapareçam em segundo plano. Tal desaparecimento é uma consequência fundamental não da tecnologia, mas da psicologia. Sempre que as pessoas aprendem algo suficientemente bem, elas deixam de ter consciência disso. Quando você olha para uma placa de rua, por exemplo, você absorve suas informações sem conscientemente realizando o ato de ler. (WEISER, 1991, p. 94).

Para ilustrar essa pequena história da Computação Sensível ao Contexto e, por conseguinte, da Computação Ubíqua, uma linha do tempo sobre alguns fatos importantes dessas duas áreas é exibida na figura 2.

Figura 2 - Alguns fatos importantes sobre a evolução da Computação Sensível ao Contexto.



Fonte: Sá (2012).

O termo computação sensível ao contexto surgiu no início da década de 1990 com Schilit e Theimer, a ideia era ser um software que se adapta segundo seu local e uso, apontar pessoas próximas e objetos, como também as alterações a esses objetos no decorrer do tempo. Computação sensível ao contexto é um paradigma computacional que propõe que as aplicações tenham acesso e tirem proveito de informações, buscando aperfeiçoar seu processamento. Trata-se da ideia de que os dispositivos podem tanto perceber quanto reagir ao ambiente em que estão inseridos para facilitar as atividades humanas (DEY; ABOWD, 2000).

Em 1994 sensibilidade ao contexto, pela primeira vez foi aplicado, no projeto dos pesquisadores B. Schilit e M. Theimer. No projeto, os responsáveis atribuíam sensibilidade ao contexto como a capacidade que um software tem de se adaptar ao local em que está sendo executado, ao conjunto de pessoas e objetos próximos, e às variações dessas pessoas e objetos ao longo do tempo. No entanto, antes mesmo dessa publicação, ainda em 1992, um projeto pioneiro já estava sendo desenvolvido: o *Active Badge Location System*, da Olivetti (um sistema baseado em localização). Este e demais trabalhos contemporâneos são, de algum modo, derivados da visão visionária de Mark Weiser, que em 1991, desenvolveu sua pesquisa baseado numa diferente forma de desenvolver e utilizar software, com capacidades avançadas de interação tanto com os usuários e com o ambiente em que é executado.

No entanto, acredita-se que ao longo dos anos seguintes, vários

pesquisadores têm procurado definir o significado do termo “contexto” sob o ponto de vista da Computação, é possível definir contexto como o termo do contexto que trata da localização do usuário, o ambiente, a identidade e o tempo, ou qualquer informação que possa ser usada para caracterizar a situação de entidades (por exemplo, pessoas, locais ou objetos) que são consideradas relevantes para a interação entre um usuário e uma aplicação, incluindo também o usuário e a aplicação.

Pesquisas vêm utilizando o conceito de contexto para beneficiar sistemas ligados a outras áreas como, por exemplo: Sistemas Colaborativos, visando melhorar os serviços de percepção e prover assistência aos grupos; Hipermídia Adaptativa, possibilitando a personalização e adaptação do conteúdo de sites web; Integração de Dados, facilitando a resolução de conflitos semânticos; e Interface Humano-Computador, com o uso do contexto para adaptar as interfaces dos sistemas tornando mais intuitiva a sua interação com os usuários (VIEIRA; TEDESCO; SALGADO, 2009).

Atualmente, vem crescendo o interesse no desenvolvimento de aplicações sensíveis ao contexto. No entanto, compreender e identificar o contexto, e executar ações automaticamente de acordo com esse contexto é uma tarefa não trivial em sistemas computacionais. É preciso definir o que exatamente considerar como contexto, em que cenários este se aplica e que informações são necessárias para descrevê-lo. É preciso viabilizar formas de adquirir o contexto o mais automaticamente possível, sem que o usuário tenha que ser questionado insistentemente. Mecanismos de raciocínio e inferência devem ser implementados, pois nem todo tipo de informação contextual pode ser obtido, automaticamente, via monitoramento. Por exemplo, uma informação contextual relacionada a um sistema de guia turístico indica se o clima está quente ou frio no local onde o turista pretende ir. No entanto, os conceitos frio e quente não podem ser inferidos apenas mediante a temperatura. Outros fatores, como a localidade ou a estação do ano, devem ser levados em consideração para derivar essa informação (VIEIRA; TEDESCO; SALGADO, 2009).

O modelo de Computação pervasiva ajudou a moldar, o nascimento de outro conceito, o de Internet das Coisas (do inglês Internet of Things IoT), que é uma

rede de dispositivos interconectados que possuem sensores, softwares e conectividade de rede que lhes permitem capturar e trocar informações com outras “coisas” como: computadores, relógio, *smarts*, entre outros (MANCINI, 2018). Do inglês a IoT é um paradigma que preconiza um mundo de objetos físicos embarcados com sensores e atuadores, conectados por redes sem fio e que se comunicam usando a Internet, moldando uma rede de objetos inteligentes capazes de realizar variados processamentos, capturar variáveis ambientais e reagir a estímulos externos

Mais do que um conceito a internet das coisas é, essencialmente, uma estrutura de arquitetura que permite a troca e integração de dados entre os sistemas mundiais e computador físico sobre a infraestrutura de rede existente. Isso mesmo. Roupas, aparelhos domésticos, acessórios, veículos, máquinas, empresas e tudo o que a tecnologia ainda irá permitir. Diversos setores já estão utilizando a internet das coisas para facilitar as rotinas, sejam elas pessoais ou profissionais, isso porque ela permite um nível de informação em tempo real que nunca tivemos antes.

O desenvolvimento de sensores com capacidade de detectar aspectos do mundo real, como temperatura, umidade, presença, e enviá-los para centrais de informação para trata-las, e dar a elas o status de informação, é a chave para entender e compreender como será a vida na era da Internet das Coisas.

Podemos, por exemplo, monitorar nossas casas e famílias remotamente para mantê-los seguros. Além disso, no âmbito profissional, as empresas podem melhorar os processos para aumentar a produtividade e reduzir o desperdício de material e tempo de inatividade. Nas cidades, sensores na infraestrutura podem ajudar a reduzir o congestionamento rodoviário e nos avisar caminhos alternativos. Esta tecnologia pode até informar sobre a mudança das condições ambientais e nos alertar sobre possíveis desastres iminentes.

2.1.2 Sistemas Sensíveis ao Contexto

Devido à importância do estudo referente a contexto, diferentes áreas da computação iniciaram a pesquisa sobre sistemas sensíveis ao contexto, como IA, que procura investigar o uso das técnicas de modelagem, o conceito nos sistemas

inteligentes (VIEIRA, 2009). Com uma ampla gama de possíveis situações, precisa-se ter uma maneira para o serviço se adaptar de forma adequada, a fim de melhor apoiar as interações por computador humano e humano-ambiente (ABOWD; DEY; SALBER, 1999).

Sistemas sensíveis ao contexto, do inglês *Context-Sensitive System* (CSS), são aqueles que utilizam contexto para prover informações ou serviços relevantes para uma execução de uma tarefa (PATRÍCIO, 2010). Em um CSS, os dispositivos detectam e aproveitam recursos próximos: um computador portátil localizado ao lado de um quadro eletrônico pode usar a superfície de exibição maior ou permitir que o usuário interaja com outros usuários portáteis próximos. Finalmente, os aplicativos de recuperação baseados em contexto coletam e armazenam informações de contexto e permitem a posterior recuperação de informações com base em informações de contexto (ABOWD; DEY; SALBER, 1999).

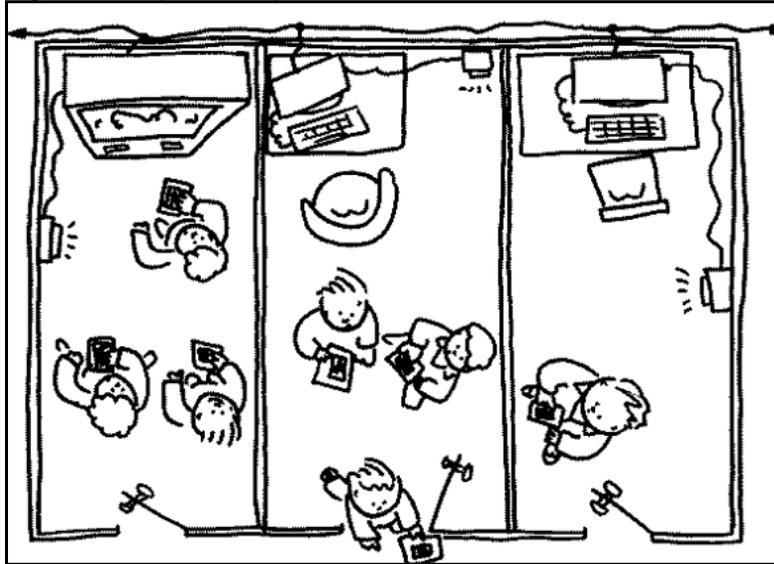
Identifica-se um CSS se ele usa contexto para fornecer informações relevantes e/ou serviços para o usuário (CHOI, 2007). Outra característica é que se adaptam de acordo com a localização do usuário, o conjunto de pessoas próximas, dispositivos móveis, bem como a alteração ao longo do tempo (ADAMS; SCHILIT; WANT, 1994, tradução nossa). Estes sistemas têm como objetivo proporcionar acesso ubíquo à informação, comunicação e computação.

A computação ubíqua é um paradigma caracterizado pela presença de dispositivos portáteis, que estão cada vez mais fazendo parte das atividades diárias das pessoas. O termo computação ubíqua foi definido pelo cientista Mark Weiser, para se referir a dispositivos conectados em todos os lugares de forma transparente para o ser humano (NAZÁRIO, DANTAS TODESCO, 2015).

Esse tipo de software sensível ao contexto se adapta de acordo com a localização do usuário, o conjunto de pessoas próximas, dispositivos acessíveis, bem como a alteração de tais coisas ao longo do tempo. Um sistema com esses recursos pode examinar o ambiente de computação e reagir a mudanças no ambiente (FIGUEIREDO; NAKAMURA, 2003).

Segundo Adams, Schilit e Want (1994, tradução nossa), são três aspectos importantes do contexto: onde você está, com quem está, e que recursos estão nas proximidades, conforme apresentado na figura 3.

Figura 3 - Aspectos importantes contexto.



Fonte: Admans, Schilit e Want (1994).

No entanto, projetar um CSS não é uma tarefa trivial, uma vez que é necessário lidar com questões relacionadas à: que tipo de informação deve ser considerado contexto, em como representar esta informação, como podem ser adquiridos e processados, como projetar o uso do contexto pelo sistema (SANTOS, 2008). Assim, além de lidar com entradas explícitas, também considera informação de contexto aquelas capturadas por meio de sensores, ou seja, entradas implícitas, tais como: localização, recursos e infraestrutura disponível, preferências do usuário, atividade do usuário (LOPES, 2006).

A utilização de contexto em aplicações permite que desenvolvedores possam melhorar a usabilidade da aplicação, onde o sistema reagiria de acordo com o ambiente sem intervenção do usuário, um exemplo simples seria o nível de bateria do celular quando está abaixo do normal, ele identifica e executa as ações para economizar a bateria, como reduzir a luminosidade, desativar alguns sensores (PINHEIRO, 2014). Aplicações sensíveis ao contexto identificam o ambiente no qual estão sendo utilizados e tomam decisões, estas mesmas aplicações funcionam de acordo com as ações executadas por usuários, objetos ou até mesmo por outros sistemas (PINHEIRO, 2014).

Um CSS necessita que informações contextuais sejam trocadas e utilizadas por diferentes entidades, como objetos, pessoas, dispositivo com uma mesma compreensão semântica, sendo assim um modelo de contexto precisa ser

apropriado para dar suporte à ligação e o funcionamento em conjunto com outros computadores (SANTOS, 2013).

Fonteles (2013) descreve as tarefas que um CSS segue para seu funcionamento:

- a) adquirir informações contextuais de baixo nível;
- b) elevar o nível dessas informações por meio de técnicas, como inferências ou transformações, para que elas se tornem utilizáveis pela aplicação;
- c) armazenar as informações para que elas possam ser acessadas posteriormente e/ou gerar eventos quando determinados estados de contexto são detectados.

Na intenção de facilitar o entendimento sobre contexto, Augustin *et al*, (2013), categoriza contexto em:

- a) contexto computacional, onde se trata de informações lógicas e físicas de dispositivos computacionais, sejam eles móveis ou fixos;
- b) contexto de usuário, que são as informações baseadas no usuário, como o perfil, as atividades realizadas, atual localização;
- c) contexto físico, que são ambientes, que a informação é iluminação, temperatura, pessoas próximas;
- d) contexto de tempo, informações que podem ser utilizadas para a consulta de histórico para a tomada de decisões.

Contexto, portanto, nas palavras de Santos (2006), é uma ferramenta importante para apoiar a comunicação entre usuários e sistemas, pois ajuda a diminuir os conflitos, aumenta a expressividade dos diálogos, possibilita a melhoria dos serviços e informações oferecidos pela aplicação, com isso, a tendência é que as aplicações se tornem mais amigáveis, flexíveis e fáceis de usar.

2.1.3 Exemplos de Aplicações

Aplicações sensíveis a contexto são aquelas que utilizam informações de contexto para fornecer serviços e informações relevantes a usuários e a outras

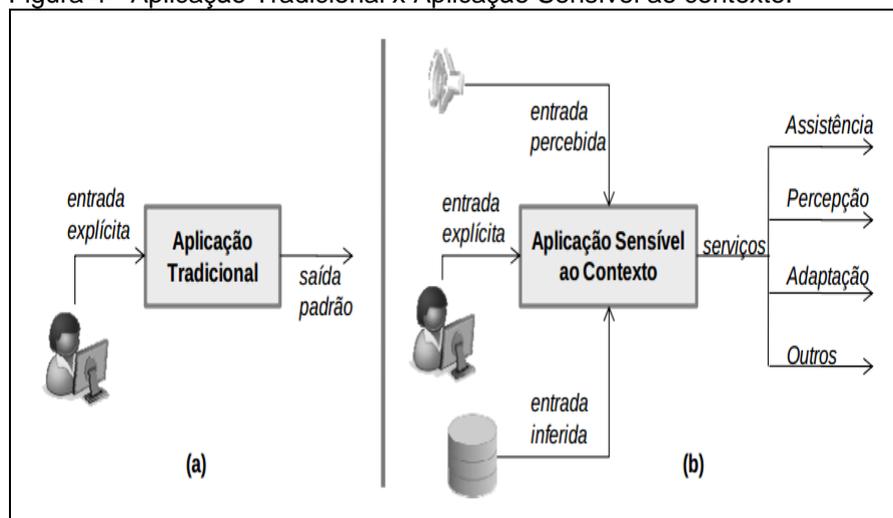
aplicações na realização de alguma tarefa (DEY, 2001). Uma característica importante das aplicações, é que através do armazenamento de informações capturadas, é possível obter um histórico do contexto do usuário, para certas aplicações este pode ser um dado importantíssimo (ANJOS, 2006).

Aplicações sensíveis ao contexto estão se tornando mais prevalentes e podem ser encontradas nas áreas de computação móvel, robótica, adaptável e interfaces inteligentes; usuário, realidade aumentada, computação adaptativa, ambientes inteligentes e ao contexto interfaces importantes. Não é de surpreender que, na maioria dessas áreas, o usuário é móvel e seu contexto é mudado rapidamente (ALMEIDA, 2011, p. 231).

Os aplicativos podem exibir informações de contexto, capturá-las para acesso posterior e fornecer recuperação baseada em contexto de informações armazenadas. De grande interesse são as aplicações sensíveis ao contexto, que detectam as informações de contexto e modificam o seu comportamento em conformidade, sem a intervenção explícita do usuário (ABOWD; DEY; SALBER, 1999).

Para as aplicações sensíveis ao contexto, um dos desafios está em como definir um modelo de contexto que seja o mais completo e adequado para o domínio da aplicação a ser desenvolvida (ROCHA, 2010).

Figura 4 - Aplicação Tradicional x Aplicação Sensível ao contexto.



Fonte: Vieira (2009).

Na figura 4 tem-se um comparativo entre aplicações tradicionais e aplicações sensíveis ao contexto, (a) são aplicações tradicionais, as que são executadas com informações fornecidas pelos usuários, sendo que o usuário faz

entrada dos dados e a saída é padrão, diferente do (b) que são aplicações sensíveis ao contexto, além de obter informações fornecidas pelos usuários, fazem monitoramento do ambiente, coletando dados armazenados em bases de conhecimento.

A computação móvel permite aos usuários se moverem ao longo de um ambiente enquanto carregam seu dispositivo móvel. Combinando isso com Wireless, os usuários têm acesso a informações e serviços não diretamente disponíveis em seu dispositivo de computação portátil (ABOWD et al., 1997).

A mobilidade do usuário produz situações tais, onde o contexto é mais dinâmico, isto é, as informações do contexto modificam-se de contínuo, tornando complicado para que o usuário passe todas as informações do contexto para o computador a cada situação. Deste modo, o computador precisa coletar informações do contexto de modo automático, e fornecê-las de forma transparente para o usuário (DEY, 2001).

2.1.3.1 Cyberguide

O projeto *Cyberguide* foca em como os computadores portáteis podem auxiliar na exploração de espaços físicos e ciberespaços. O sistema *Cyberguide* é um protótipo de guias turísticos inteligentes e portáteis, que fornecem informações a um turista com base na informação de contexto do usuário, especificamente a localização e orientação.

Os visitantes do GUV recebem um mapa dos vários laboratórios e um pacote de informações descrevendo todos os projetos. Com o *Cyberguide*, a ideia é transformar todas as informações do papel num guia que saiba onde se está, o que se está olhando e mostrar informações adicionais. Para o desenvolvimento da aplicação com reconhecimento de localização, a base foram os trabalhos *PARCTab*, projeto *InfoPad* e no sistema *Olivetti Active Badge* (ABOWD et al., 1997).

2.1.3.2 CO2DE/POLITeam

Este é um editor síncrono/assíncrono de diagramas Linguagem de

Modelagem Unificada (UML), projetado para permitir que usuários compartilhem o mesmo espaço de edição e controlem versões dos documentos sendo construídos (VIEIRA, 2009).

O conceito de sistemas sensíveis de contexto entra na questão de resolver os problemas de sobrecarga inerentes aos mecanismos de percepção em sistemas colaborativos; ele utiliza o contexto de trabalho do usuário (abre uma pasta, executa passo a passo em algum processo, entre outros), para permitir que usuários especifiquem perfis de percepção.

Como ferramenta colaborativa, ele provê funcionalidades para ajudar membros individuais a estarem “cientes” das informações contextuais relacionadas à tarefa corrente. Por exemplo: informação sobre composição do grupo, o papel de cada membro do grupo, cada novo documento sendo simultaneamente discutido, dentre outras. O usuário pode configurar preferências de percepção em seu perfil que será anexado a um ou mais artefatos (VIEIRA, 2009).

2.1.3.3 Amazon

Esse portal exemplifica a utilização do contexto para a promoção da adaptação de sistemas web, mantendo o histórico de uso do site pelos utilizadores, como por exemplo - materiais visualizados, comprados e pesquisas feitas, como também avaliações dos materiais feitas por outros utilizadores e o tipo de material que o utilizador está a consultar (MATIAS, 2014).

Com o cruzamento dessas informações de contexto, o sistema vai descobrindo padrões como, por exemplo: “Clientes que visualizaram este livro também visualizaram”, e faz sugestões de materiais que o utilizador atual poderia se interessar (MATIAS, 2014, p. 23).

O sistema recomenda ainda, guias de navegação pelo website, orientando-se por elaborações de outros utilizadores, vinculadas ao assunto do material que o utilizador visualiza naquele momento (MATIAS, 2014).

2.1.3.4 Gmail

O *Gmail* é um serviço de gerenciamento de e-mail on-line que foi criado pela empresa americana Google, e trabalha com recomendação de links e propaganda conforme conteúdo lido pelo usuário (VIEIRA, 2009).

O GMail mantém uma área de propaganda periférica nos seus sistemas, embutida no ecrã de visualização dos e-mails. Essas propagandas são sensíveis ao conteúdo da mensagem que o utilizador está a ler. Desse modo, as propagandas exibidas (área *sponsored links*) exibem sugestões de páginas comerciais relacionadas com o conteúdo do texto do e-mail. (MATIAS, 2014, p. 03).

A Google é um exemplo de CSS, com o mecanismo de propaganda sensível ao contexto *AdSense*. O *AdSense* tem como objetivo mostrar anúncios mais relevantes e úteis ao usuário - a página que está visitando ou e-mail que está lendo. As informações de contexto são provenientes do conteúdo exibido na página atual.

2.2 MODELAGEM DE SISTEMAS SENSÍVEIS AO CONTEXTO

O desenvolvimento de um CSS deve ser visto de acordo com duas perspectivas: uma parte que é dependente de um domínio e aplicação específicos (especificação e uso do contexto) e outra parte que é independente de domínio ou aplicação (gerenciamento do contexto). Diferentes domínios ou aplicações demandam distintos conjuntos de CEs² e de tratamentos do contexto para seu uso. Por outro lado, o gerenciamento de contexto pode ser modularizado e tratado de maneira independente, uma vez que envolve as funcionalidades de adquirir, processar, armazenar e disseminar CEs (VIEIRA; TEDESCO; SALGADO, 2009).

Modelagem de contexto representa uma tarefa que estrutura uma informação contextual objetivando permitir a interpretação sem erros da informação de contexto pelos usuários, permite o processamento de informações de contexto e sua reutilização (LAMAS, 2009).

As modelagens de sistemas sensíveis ao contexto apresentam algumas

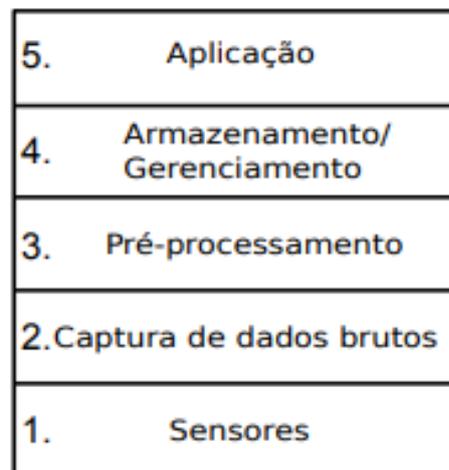
² Um elemento contextual do inglês *Context-Sensitive System* (CSS), é qualquer dado, informação ou conhecimento que permite caracterizar uma entidade em um domínio (VIEIRA; TEDESCO; SALGADO, 2009).

dificuldades, tais como: a caracterização dos elementos para uso na aplicação, a coleta da informação através dos dispositivos, a representação do modelo formal de contexto, o processamento e interpretação das informações de contexto, o tratamento de desempenho, segurança (VIEIRA, 2009). Estes modelos são necessários, pois representam quais informações contextuais serão levadas em consideração e como será a relação junto com a aplicação (PINHEIRO, 2014).

Os modelos de representação do contexto podem ser formais ou informais. Onde modelos informais são esquemas próprios de representação e específicos para algum sistema, sendo assim, tornam difícil a troca e entendimento do contexto pelos sistemas. Já os modelos formais utilizam abordagens de modelagem para manipulação do contexto, de modo que o mesmo possa ser utilizado, compreendido e compartilhado por diferentes sistemas (SANTOS, 2006).

Na figura 5 é apresentado a arquitetura conceitual que representa a estruturação básica de um sistema sensível ao contexto.

Figura 5 – Arquitetura conceitual sensível ao contexto.



Fonte: Azevedo (2015, p. 18).

A primeira camada trata-se de sensores, responsáveis por capturar informações, os mesmos podem ser físicos, que obtém dados físicos, como a temperatura, luminosidade, localização entre outros, sensores virtuais que buscam informações via software como, buscar a localização de algum empregado a partir de uma calendário virtual, e sensores lógicos que utiliza informação dos sensores virtuais e físicos para capturar alguma informação de contexto, por exemplo, a identificação da localização de um usuário a partir do seu login em um computador e

um mapeamento de onde este computador se encontra (FONTELES, 2013).

Na segunda camada (captura de dados) é responsável por recuperar as informações de contexto onde os dados, de forma a permitir que um dado seja obtido independentemente de como ele é coletado, ou seja, o sensor utilizado para sua captura pode ser alterado sem que as camadas superiores precisem se preocupar com isso (AZEVEDO, 2015).

Na camada de pré-processamento ocorre a transformação dos dados de sensores em dados de contexto; interpretação de dados: combinação dos dados sensoriais, leitura de vários dados de sensores ao longo do tempo e análise de padrões; nesta camada ocorre a definição dos contextos complexos (SANTOS, 2013).

A próxima camada é responsável por armazenar e gerenciar as informações obtidas para a camada superior, responsável pela organização e distribuição das informações de contexto para seus clientes. E por último, se tem a camada da aplicação, que deve reagir aos diferentes eventos e as mudanças de contexto. Essas mudanças podem ocorrer, por exemplo, em função da localização do usuário ou do tipo do usuário (SANTOS, 2013).

2.2.1 Elementos para modelagem de contexto

Sistemas sensíveis ao contexto são aqueles que gerenciam elementos contextuais relacionados a um domínio de aplicação e que usam esses elementos para apoiar na execução de alguma tarefa. Essa ajuda pode ser alcançada, melhorando a percepção do agente sobre a tarefa, ou provendo adaptações que facilitem a execução da tarefa (QUEIROZ FILHO, 2012).

Existem alguns critérios para identificar o que escolher como elemento contextual, a capacidade de relatar as propriedades úteis; possibilidade de inferência de contextos complexos; e a facilidade em ser identificado de forma automática, mais precisa possível (VIEIRA, 2009).

Algumas classificações, segundo Santos (2006), foram criadas com objetivo de facilitar a identificação dos elementos de contexto. Uma delas trata-se de dividir em: contexto primário ou de baixo nível; e contexto complexo ou de alto nível.

Contexto primário, para Patrício (2010), são elementos que podem ser identificados automaticamente por sensores (temperatura, horário, localização, atividade atual), e contexto complexo, trata-se de informações fornecidas pelo próprio usuário ou identificadas por motores de inferência (com quem está, onde está, fazendo o quê). Para identificar se um usuário está numa sala de aula, a regra obtém os contextos básicos como a identificação da sala, se existem outras pessoas na mesma sala, a posição do usuário em relação a outras pessoas, e se existe alguma apresentação rodando (SANTOS, 2006).

Um dos desafios ao desenvolver um CSS é limitar as ações junto ao contexto nesses sistemas e definir os elementos contextuais que identificam a situação na qual as ações são executadas. Estudos mostram que a identificação dos elementos de contexto depende fortemente do tipo da tarefa e domínio em questão (VIEIRA; TEDESCO; SALGADO, 2009). Para facilitar a identificação, as informações foram classificadas de acordo com alguns critérios (PATRICIO, 2010).

Temos como elementos para modelagem de contexto, objeto do contexto que são as entidades, atividades, tempo; estado, onde tem estado estático e estado dinâmico, local e distribuído, temporalidade; existe relacionamento entre contextos que é abstração, composição, distribuição, inferência; a qualidade de contexto, ambiguidade e confiabilidade, negociação; representação e eventos, declaração de meta-eventos, abstração e composição.

2.2.2 Características da informação de contexto

Características da informação mostra que o contexto dinâmico tem características constantes referentes à forma como as informações de contexto são obtidas (SOUZA, 2012), e as mesmas informações mudam instantaneamente. Na informação contextual existem algumas características, como temporais, que são estáticos e dinâmicos. (RODRIGUES, 2017). Como exemplo, se tem a localização do usuário e a atividade atual; contexto estático é onde as informações se mantêm fixas, como CPF e RG de uma pessoa (VIEIRA, 2009).

Dentro da classificação dinâmica, segundo Kudo (2004), há três tipos:

a) informação dinâmica sentida: são informações capturadas através de

- sensores, como a localização do usuário e a temperatura do ambiente.
- b) informação de contexto dinâmica explícita: são fornecidas pelo usuário, como senha de acesso, seu perfil.
- c) informação interpretada: trata-se do contexto obtido através de regras de interpretação que varia de uma simples condição até algoritmos de IA mais complexos.

Visando a proposta de facilitar a identificação do que é contexto foram criadas classificações. As informações contextuais podem ser divididas em: dimensões dos elementos contextuais, granularidade da informação, periodicidade de atualização e sua relevância em relação ao foco (VIEIRA, 2009).

Vieira (2009) descreve as características da informação de contexto:

- a) informação contextual pode possuir várias características temporais: estático X dinâmico; atraso na recuperação de contexto; histórico de contexto;
- b) informação contextual é imperfeita: imperfeição dos sensores, desatualização, algoritmos de inferência;
- c) contexto pode ter várias representações alternativas: abstrações;
- d) informação contextual é altamente inter-relacionada.

Existem algumas diretrizes que facilitam o processo para definir qual informação é importante ou relevante que são: *who* indica informações contextuais sobre as pessoas, quem é o usuário que grupo pertence, pessoas próximas; *where* determina a localização, orientação, temperatura; *what* identifica qual atividade está sendo realizado pelo usuário, exemplo são falando, caminhando, em reunião; *when* indica o momento da atividade como, o da, hora, estação; *why* indica a motivação do usuário ao executar uma atividade, como por exemplo o estilo musical do usuário de acordo com as buscas; *how* define a forma que os dados são coletados, como GPS que são utilizados para identificar a localização do usuário (VIEIRA, 2009).

Na granularidade, o elemento pode ser considerado básico ou complexo. Um elemento básico é quando a granularidade é mais baixa e é identificada de forma automática (coordenadas geográficas, informações temporais). Já o elemento complexo é composto/inferência de outros elementos complexos. Como exemplo se pode citar que a partir de um elemento básico como a sala onde o usuário encontra-

se, é a indicação de que outras pessoas estão presentes na sala, isso pode inferir em um elemento complexo, indicando uma atividade atual como uma reunião (VIEIRA, 2009).

Um sistema sensível ao contexto possui funcionalidades comuns e as que podem ser selecionadas no desenvolvimento do software; essas funcionalidades são chamadas de características e referem-se aos requisitos do sistema (SILVA, 2016).

A MC segundo Pereira e Maia (2008), é o exercício de especificar a variabilidade de propriedades dos conceitos e suas interdependências por meio de um modelo. A MC é uma atividade que trata da complexidade em expressar vários requisitos em forma de características (MASSEN; LICHTER, 2004).

Para Lopes (2013) a MC trabalha com a identificação das necessidades, viabiliza os processos e tecnologias necessárias mostrando as vantagens e desvantagens de uma implementação. Recentemente grandes números de aplicações sensíveis ao contexto têm sido desenvolvidos baseando-se em diversos modelos contextuais para uma variedade de domínios (ROCHA, 2010).

A MC é uma forma de representação de um domínio ou sistema, onde se relaciona às características da aplicação (SILVA, 2016).

A Modelagem de Características provê uma representação para expressar a variabilidade e a dependência entre as características. Características são entidades de primeira classe do projeto, que encapsulam representações de programas estruturadas hierarquicamente (como código, arquivos de configuração, etc) e que descrevem em alto nível de abstração os requisitos de um sistema. São criadas/adaptadas em função de um contexto. É considerada a presença de uma característica base (um ponto de partida para os produtos de uma linha) e seu comportamento é refinado por sua combinação com outras características. Esta mudança de comportamento altera a semântica do produto final. Os diagramas de características da Modelagem de Características definem as diferentes possibilidades de produtos a serem gerados a partir de um conjunto de características organizadas hierarquicamente. Em essência, eles definem uma linha de produtos. (PEREIRA; MAIA, 2008, p. 03).

Essas características segundo Costa (2012), podem ser categorizadas de acordo com a origem: sentido (informação gerada através de sensores), derivado (a qual pode ser aprendida ou inferida), informações definidas pelo usuário (onde fornece valores à aplicação durante alguma configuração), e fornecido por um serviço (a informação é dada por algum fornecedor de serviços, como agenda ou cronograma onde há uma lista de atividades).

Um sistema pode ser considerado sensível ao contexto mesmo não apresentando todas as características. Algumas características que devem ser consideradas por sistemas sensíveis ao contexto são:

- a) heterogeneidade: há várias fontes de informações com atualização de dados, enquanto alguns sensores podem observar determinadas situações do ambiente físico e prover acesso rápido e em tempo real, outros sensores não oferecem os mesmos recursos (como imagens de uma câmera, ou coordenadas de um GPS que precisam ser interpretados antes de serem utilizados pelas aplicações) (ROCHA, 2010);
- b) captura de experiência: capacidade de identificar as necessidades dos usuários e interesses que podem ou precisam ser realizados tomando como base a percepção de informações previamente capturadas e atuais (QUEIROZ FILHO, 2012);
- c) mobilidade: CSS dependem de fontes móveis; isso traz problemas à heterogeneidade, já que o fornecimento de informações contextuais deve ser adaptável à mudança do ambiente (ROCHA, 2010);
- d) comportamento adaptável: refere-se a um sistema de se auto adaptar às limitações do ambiente, dos dispositivos e dos serviços, para aumentar, para permanecer em funcionamento de forma otimizada e personalizada (QUEIROZ FILHO, 2012);
- e) histórico: aplicações sensíveis ao contexto podem necessitar do acesso a estados passados e estados futuros, então históricos de contexto é outra informação (ROCHA, 2010);
- f) raciocínio: aplicações sensíveis ao contexto requerem do raciocínio quando usa informação para identificar alguma mudança na situação do contexto do usuário e/ou ambiente. Existem técnicas para verificação de consistência do modelo e técnicas de raciocínio sobre contexto; essas técnicas são utilizadas para definir novos fatos/situações através dos fatos já existentes (ROCHA, 2010).

O objetivo da MC é identificar as características externamente visíveis dos produtos que farão parte da Linha de Produto de Software (LPS) (LEE; MUTHIG,

2006), capturando os pontos comuns e variáveis que existem entre eles. Tais características são organizadas num modelo chamado modelo de características, usado como ponto de partida para o recorte necessário à instanciação de novos produtos. Ainda, esse modelo uniformiza o entendimento entre todos os envolvidos no processo de desenvolvimento, como usuários, especialistas de domínio e desenvolvedores (MASSEN; LICHTER, 2002).

As informações de contexto são um conceito chave na definição das aplicações sensíveis ao contexto, assim sendo, para a elaboração de uma Linha de produtos que contribui para o desenvolvimento de produtos sensíveis ao contexto, importa que as informações de contexto significativas para o domínio sejam representadas no modelo de características, além do impacto dessas informações na reconfiguração dos produtos em tempo de execução (FERNANDES, 2009).

Tais características para Fernandes (2009) podem representar, por exemplo, conceitos ou funcionalidades do domínio, como para o domínio de guias turísticos móveis, onde poderiam existir as características Mapa e Visualizador de Imagens.

O modelo de características representa as características de uma família de sistemas em um domínio, suas semelhanças e diferenças e as relações entre elas. Características (*features*) representam as capacidades ou abstrações do domínio obtidas por especialistas, usuário ou a partir de sistemas já existentes (BLOIS, 2006).

As características são independentes de outras, e não modificam os seus estados quando agrupadas. Estas podem ser combinadas sem mudar as demais características. Entretanto, uma característica pode refinar o comportamento de outras ou necessitar a exclusão/substituição de uma outra por questões de incompatibilidade. (PEREIRA; MAIA, 2008, P. 03).

Além de características comuns, os sistemas têm características próprias que diferenciam uns dos outros, estas características que diferenciam são chamadas de variabilidade e responsáveis por permitir o desenvolvimento de aplicações variadas (COSTA, 2012).

Segundo Blois (2006), o modelo de características tem alto nível de abstração podendo ser entendido como um elemento chave para relacionar os conceitos de mais alto nível aos demais artefatos de análise e projeto, sendo usado

como ponto de partida para o recorte necessário à instanciação de novos produtos.

As características em uma LPS podem ser classificadas quanto à variabilidade em três tipos, segundo Fernandes (2009, p. 23):

- a) pontos de variação, que representam pontos onde decisões são tomadas, refletindo a parametrização no domínio de uma maneira abstrata, e são configuráveis por meio de variantes;
- b) variantes, que representam características que são alternativas disponíveis para configurar um ponto de variação; e
- c) invariantes, que compreendem as características “fixas”, que não são configuráveis no domínio.

Com relação as opções, as características podem se classificar segundo Oliveira (2006) e Van Der Linden, Schmid e Rommes (2007) em:

- a) comuns ou mandatórios: que precisam obrigatoriamente estar presentes em todos os produtos derivados a partir de uma LPS;
- b) opcionais: que estão presentes somente em parte dos produtos e fazem parte da arquitetura da LPS; e,
- c) específicas de produto: que se apresentam somente num produto específico, não fazendo parte da arquitetura da LPS.

No decorrer do ciclo de vida de uma LPS, as características podem variar sua classificação. Por exemplo, uma característica específica de produto poderá fazer parte da arquitetura da LPS e ser introduzida noutros produtos, passando a ser uma característica variável, devido à própria evolução da LPS. Um dos conceitos ligados à MC é a cardinalidade, que é usada para definir o número mínimo e máximo de características que podem ser escolhidas a partir de um conjunto de alternativas de um ponto de variação (FERNANDES, 2009).

Outro conceito importante para Fernandes (2009) é a especificação de restrições entre as características, incluindo os relacionamentos de dependência e mútua exclusividade.

No desenvolvimento de um novo produto em uma LPS, precisa haver uma maneira de indicar a necessidade ou incompatibilidade da seleção conjunta de características. Importa destacar que tais indicações precisam ser verificadas no processo de instanciação dos produtos de uma LPS, como modo de garantir a

derivação de produtos mais consistentes com relação às especificações do domínio (FERNANDES, 2009).

Outra característica de funcionalidade do sistema é a variabilidade, que segundo Burgareli (2009), é a forma pela qual as variáveis do software podem se diferenciar entre si, e a capacidade de um sistema realizar algumas mudanças de uma forma fácil.

Dias (2005), diz que variabilidade pode ser entendida como a possibilidade de configuração, ou, a habilidade que um sistema ou objeto de software possui de ser alterado, customizado, ou configurado para um contexto em particular.

Os modelos de variabilidade definem as características que representam os pontos de variação, que são as características que podem ou não ser selecionadas durante a configuração de um produto (MARINHO, 2012). Algumas abordagens foram propostas para modelagem de variabilidade, tais como Modelos de Variabilidade Ortogonais (OVM) e Modelos de Características (MC).

Os elementos básicos que compõem o OVM são pontos de variação e suas variantes, para o conjunto de variantes relacionadas a um mesmo ponto de variação, é possível definir um intervalo para o número de variantes selecionáveis desse grupo. O OVM enfatiza uma distinção clara entre pontos de variação e variantes. Um MC denota o conjunto de produtos válidos para o sistema, nesses modelos as características são os blocos de construção e representam os aspectos variáveis ou pontos comuns a todos os sistemas. Combinando as características de um MC de maneiras diferentes, respeitando as restrições de composição impostas pelo MC, obtém-se um conjunto de produtos (MARINHO, 2012).

Introduzir variabilidade em sistemas sensíveis ao contexto é uma característica que pode ser selecionada após definição do método que utiliza ações, onde é capaz de adaptar-se a contextos diferentes cooperando para reuso (WINCK, 2005).

Caldas (2010) expõe que esse é um conceito chave para a abordagem de LPS, já que permite explicitar os pontos onde os produtos se assemelham, podendo ser reutilizados, e os pontos onde os produtos diferem entre si, necessitando de tratamento específico.

Dessa maneira, a variabilidade necessita de gerenciamento durante todo o processo de desenvolvimento. A notação para modelagem de variabilidade pode ser gráfica, textual ou uma combinação das duas formas (MASSEN; LICHTER, 2002).

2.2.3 Abordagens de representação de contexto

Para o desenvolvimento de aplicações sensíveis ao contexto é necessário entender as técnicas usadas para modelagem de contexto, essas técnicas são modelos que servem como estrutura para apoiar a modelagem de um sistema. Serão apresentadas algumas técnicas utilizadas para modelagem de contexto, detalhando cada técnica e suas peculiaridades, com o objetivo de prover mecanismos que sirvam de base para o desenvolvimento de novas aplicações relacionadas ao paradigma de Computação Sensível ao Contexto (LONDERO, 2012).

Uma representação formal do contexto permite que mecanismos de raciocínio lógico possam ser usados para checar a consistência das informações de contexto, possam realizar comparações com outros contextos e possam inferir novas informações complexas a partir dos contextos existentes. (NUNES, 2010). Sabem que um CSS usa informações contextuais de várias fontes heterogêneas, sendo assim é necessária uma representação que permita a troca de informações entre aplicação e usuário. (PATRICIO, 2010).

Alguns pesquisadores vêm estudando diferentes técnicas e modelos de representação de informações e conhecimento para representação de contexto. A técnica Par de chave-valor (figura 6) é considerada uma das mais simples, onde a informação de contexto é representada através de pares compostos por uma chave como atributo de identificação e um valor associado; a princípio é uma estrutura fácil de gerenciar, mas impossibilita o uso de alguns algoritmos de recuperação de contexto. (PINHEIRO, 2014).

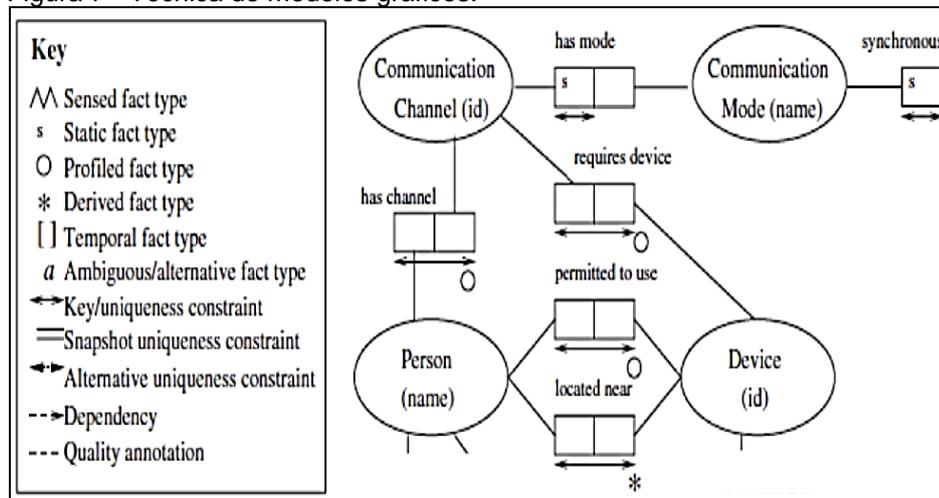
Figura 6 - Técnica par de chave-valor.

CHAVE	VALOR
DATA	01/01/2014 - 31/12/2014
HORA	00:00 - 23:59
LOCALIZAÇÃO	-2° 30.692', -44° 18.268'
AÇÃO	
Exibir mensagem "Bem vindo à sua casa!"	

Fonte: Pinheiro (2014).

Em outra técnica como Linguagem de marcação se utiliza o padrão *Extensible Markup Language* (XML) para modelar a informação contextual. A característica principal é a presença de hierarquias que contém *tags* de marcação com atributos e conteúdo. Como é um padrão, facilita a troca de informações entre fontes heterogêneas. (BEZERRA, 2015).

Figura 7 - Técnica de modelos gráficos.



Fonte: Avancini (2016).

Modelos gráficos (figura 7) equivalem em modelos como os da *Unified Modeling Language* (UML), sendo apropriados para modelagem de contextos, pois trata-se de uma linguagem de modelagem de propósito geral (MARINHO, 2012), além de possuir uma estrutura genérica e são particularmente úteis para a estruturação contextual, de informações, usando elementos visuais, como diagramas.

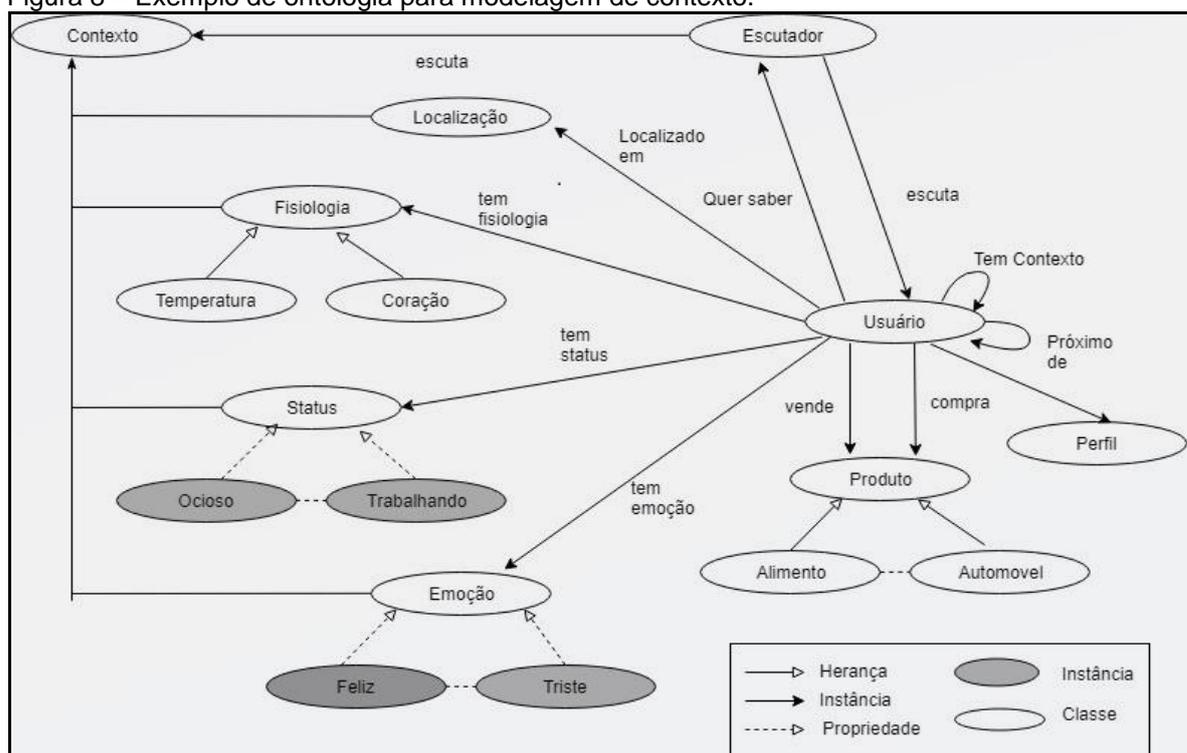
Em diversas áreas, ontologia é um modelo de dados que representa um

conjunto de conceitos dentro de um domínio e os relacionamentos entre estes. No âmbito da elaboração de Ontologias uma problemática que tem se colocado é a ausência de um padrão teórico-metodológico para sua elaboração.

As ontologias são utilizadas em inteligência artificial, *web* semântica, arquitetura da informação e engenharia de *software*, como uma forma de representação de conhecimento sobre o mundo. Geralmente descrevem indivíduos, classes, atributos ou relacionamentos. Trata-se de um modelo que representa objetos, conceitos, Entidade e Relacionamento – ER, dentro de um domínio de conhecimento. O domínio estabelece um vocabulário comum para compartilhamento de informações, o que facilita a troca de conhecimento entre usuário e sistema. (PATRICIO, 2010).

A figura 8 apresenta um exemplo de ontologia para modelagem de contexto.

Figura 8 – Exemplo de ontologia para modelagem de contexto.

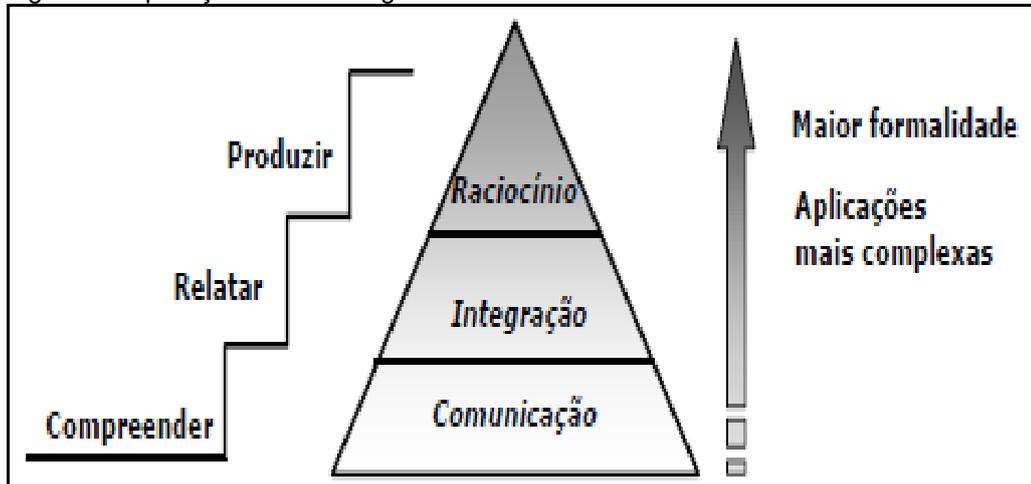


Fonte: Fernandes (2009).

A base da pirâmide apresentada na figura 9, trata-se da comunicação e compartilhamento de informações entre usuários e *software*. A camada de integração indica que diferentes ontologias são armazenadas de forma e local diferentes, podendo ser reutilizadas para compor uma ontologia sem a necessidade

de começar do zero. O raciocínio exige um maior formalismo na representação da informação de contexto, mas permite construir sistemas mais complexos (VIEIRA et al., 2006).

Figura 9 - Aplicações de Ontologia.



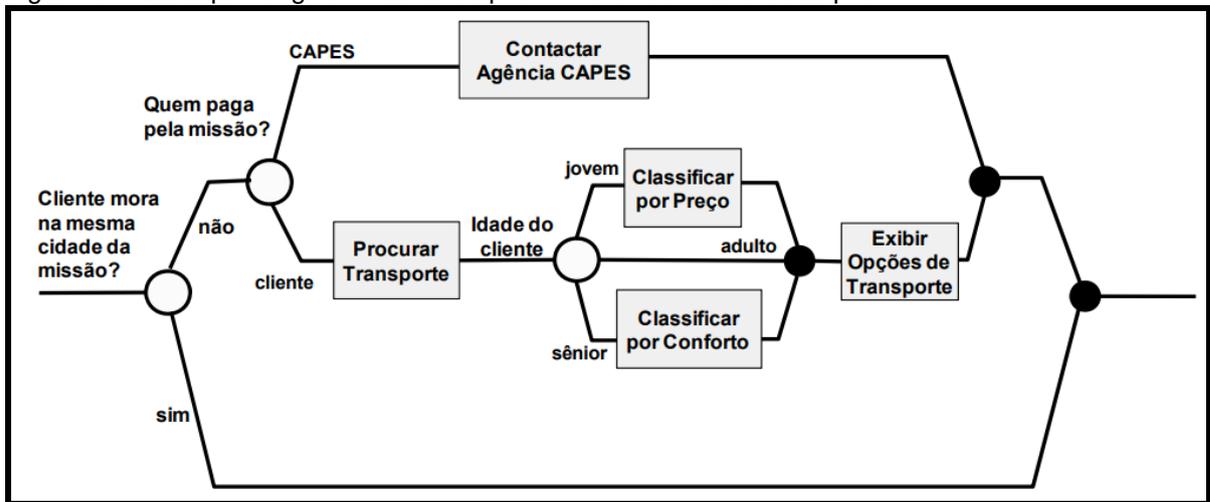
Fonte: Vieira et al. (2006).

O mapa de tópicos é uma técnica padrão para representação de conhecimento composto por tópicos baseada na associação dos indivíduos, facilitando a navegação e visualização de conteúdo (SILVA, 2016).

O grafo contextual é uma técnica de representação que objetiva modelar as atividades e raciocínio envolvidos na resolução de um problema, considerando condições específicas (o contexto) em que as atividades são executadas. (VIEIRA; TEDESCO; SALGADO, 2009).

Um CxG (figura 10) é um grafo acíclico direcionado com uma única entrada, uma única saída e uma organização serial-paralela de nós, conectados por arcos orientados, indicando um caminho de um problema até uma solução. Existem três tipos de nós: nós contextuais (círculos claros na figura), indicam pontos de decisão onde o valor do elemento contextual é avaliado; nós de recombinação (círculos escuros), que indicam que o valor testado para o elemento contextual não é mais válido; e nós ação, que representam uma atividade a ser executada. Os arcos indicam caminhos dependentes do contexto que avaliam se um conjunto de atividades deve ou não ser executado na resolução daquele problema (VIEIRA; TEDESCO; SALGADO, 2009).

Figura 10 - Exemplo de grafo contextual para a tarefa Reservar Transporte



Fonte: Vieira, Tedesco e Salgado (2009, p. 19).

No quadro 1 tem-se um comparativo entre as técnicas, como vantagens e desvantagens e a forma de processamento e recuperação de cada técnica.

Quadro 1 - Comparativo entre técnicas de representação de contexto.

Técnica	Vantagens	Desvantagens	Processamento e recuperação
Par chave-valor	Estrutura simples, de fácil implementação e uso.	Não considera hierarquia. Inadequado para aplicações com estruturas complexas.	Busca linear com casamento exato de nomes.
Linguagem de marcação	Prevê hierarquia. Esquema de marcação implementa o próprio modelo.	Modelo não resolve incompletude e ambiguidade. Inadequado para estruturas complexas.	Linguagem de consulta baseada em marcação.
Mapas de tópicos	Facilita a navegação entre os elementos contextuais e a leitura por humanos.	Técnica imatura com baixo apoio de ferramentas.	Navegação por semântica.
Ontologias	Agrega regras, conceitos e fatos em um só modelo. Padrões facilitam reuso e compartilhamento. Viabiliza compreensão semântica entre humanos e máquinas.	Não permite modelar o comportamento do SCC. Tecnologia ainda imatura e com poucas ferramentas. Impacto no desempenho	Motor de inferência, linguagens de consulta baseadas em OWL ou frames.
Modelos gráficos	Facilita a especificação dos conceitos e definição do comportamento do SCC.	Não permite processar os conceitos: mapeamento para estruturas de dados	Pode ser traduzido para XML e usa de processamento em XML.
Grafo contextual	Modela comportamento contextual. Descrevem como as ações devem ser executadas por uma aplicação	Ausência de práticas de engenharia de software que deem apoio as atividades de	Ontologia CODI-User e nós.

	e especificam como o contexto as afeta. Permitem uma representação clara de múltiplas ações que dependem do contexto e são facilmente entendidos pelos usuários e engenheiros de software.	representação gerenciamento e uso de contexto.	
--	--	--	--

Fonte: Patrício (2010).

2.2.4 Monitoramento de contexto

Aplicações sensíveis ao contexto exigem uma infraestrutura básica para monitorar o contexto, que é através de uma série de recursos e técnicas. A informação de contexto pode ser retirada através, de sensores, informações já existentes (previsão do tempo, lista de tarefas), estado do dispositivo e a interação do usuário com este equipamento. Estas mesmas informações também podem ser capturadas a partir da entrada explícita de dados pelo usuário (ANJOS, 2006).

A etapa de aquisição de contexto é importante, pois possui recursos onde permite que a aplicação defina quais contextos têm interesse e quais ações devem ser tomadas quando o contexto for identificado (KUDO, 2004). A fase de aquisição trata-se de monitorar, capturar e obter informações contextuais (SANTOS, 2006).

Informações percebidas são obtidas através de sensores físicos ou lógicos. Informações estáticas são informações fornecidas pelo próprio usuário. Informações “vinda de perfil” também são fornecidas pelo usuário, mas podem ser obtidas automaticamente por outras aplicações. Informações derivadas são determinadas pelas propriedades dos dados de entrada e dos mecanismos de derivação (SANTOS, 2006).

O quadro 2, traz a classificação para as informações de contexto que podem ser: percebida, estática, proveniente de perfil ou derivada. O quadro mostra também nível de persistência, questões de qualidade e as fontes de imperfeição (SANTOS, 2006).

Quadro 2 - Classificação das informações contextuais.

Classificação	Persistência	Questões de Qualidade	Fontes de imperfeição
Percebida	Baixa	Pode ser imprecisa, desconhecida ou caduca.	Erro na percepção; falhas do sensor ou desconexões da rede; demorar introduzidas pela distribuição e processo de interpretação.
Estática	Alta	Geralmente nenhuma.	Erro humano.
Vinda de Perfil	Moderada	Tende a caducar.	Omissão do usuário em atualizar mudanças ocorridas.
Derivada	Variável	Sujeita a erros e imperfeições.	Entradas imprecisas; uso de um mecanismo de derivação imaturo ou muito simplificado.

Fonte: Santos (2006).

A figura 11 ilustra o gerenciamento de contexto utilizando alguns elementos, as fontes de contexto podem estar relacionadas ao ambiente físico, base de dados existentes, perfis de usuário ou mesmo o próprio usuário informar o seu contexto atual.

Figura 11 - Visão geral do gerenciamento de contexto.



Fonte: Vieira (2009).

2.2.5 Middlewares

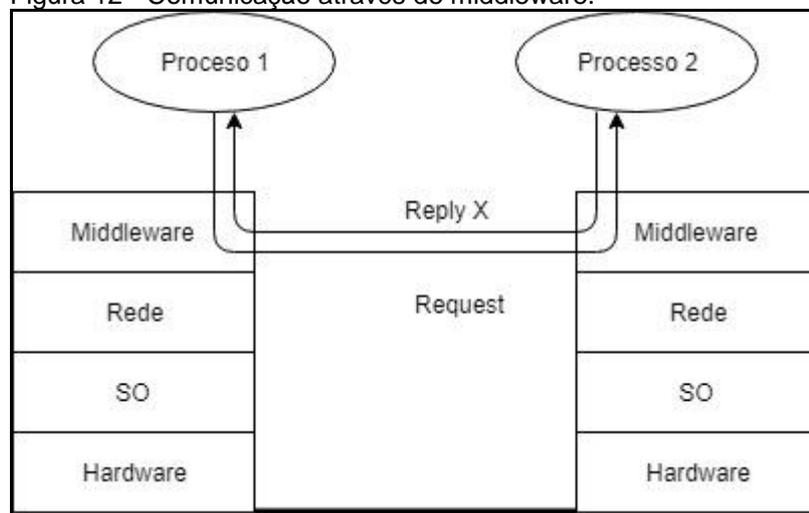
O termo *middleware* caracteriza uma camada de *software* que possibilita

a comunicação dos processos clientes e processos servidores, tendo como objetivo disponibilizar um conjunto de serviços que interligam a categoria de aplicações de forma transparente. Divide-se em componentes do ambiente de programação e do ambiente de execução (ALVES JUNIOR; VALE; SILVA, 2015). Pode-se definir ainda que o *middleware* é um conjunto de serviços, que permite a interação entre aplicações e usuário através de uma rede de dispositivos. (FERREIRA, 2001).

O *middleware* é responsável pelo gerenciamento de apresentação, comunicação, gerenciamento de informação, controle, sistema de entrega entre outros, onde o principal objetivo é de restabelecer uma espécie de “homogeneidade virtual” para os sistemas, onde o mesmo foi desenvolvido para ser portátil a uma grande variedade de plataformas, assim independente de qual camada os *middlewares* são usados, um bom *middleware* não somente torna as aplicações mais seguras, como as protegem de uma complexidade desnecessária, tornando-as mais escaláveis (ARAÚJO, 2010).

A figura 12 ilustra o método de comunicação entre as aplicações por meio de um *middleware*, todo processo é claro e a heterogeneidade existente é tratada pelo *middleware*.

Figura 12 - Comunicação através de *middleware*.



Fonte: Araújo (2010).

As primeiras soluções *middlewares* foram desenvolvidas com ideia de facilitar o desenvolvimento de sistemas de *software* que suportavam diversas atividades, tais como a computação, a descoberta e a disseminação de informação, sem o devido foco e preocupação em sistemas distribuídos (KON et al, 2002).

A ideia do *Middleware* é comparada a um Sistema Gerenciador de Banco de Dados (SGBD), pois permite separar detalhes de implementação de baixo nível, gerenciamento de transações e comunicação na rede, facilitando o processo de desenvolvimento da aplicação (OLIVEIRA JUNIOR, 2014).

O uso de sistemas *middlewares* torna livre ao desenvolvedor de aplicações de ter que se comprometer com os detalhes de cada fonte de dados e das complexidades dos ambientes computacionais (BARBOSA, 2001).

Os sistemas *middlewares* fornecem uma camada de abstração onde os engenheiros de software podem inserir comportamento adaptativo de maneira direta ou simplesmente fazendo uso de outras abordagens (ROCHA, 2007).

Alguns requisitos devem ser levados em consideração para elaboração de um *middleware* segundo Oliveira (2014), tais quais como comunicação; coordenação; confiabilidade; escalabilidade e segurança.

Os sistemas *middlewares* fornecem uma camada de abstração onde os engenheiros de software podem inserir comportamento adaptativo de maneira direta ou simplesmente fazendo uso de outras abordagens (ROCHA, 2007).

Alguns requisitos devem ser levados em consideração para elaboração de um *middleware* segundo Oliveira (2014), tais quais como comunicação; coordenação; confiabilidade; escalabilidade e segurança.

A comunicação do *Middleware* vai garantir a troca de informações entre os sistemas integrantes de forma transparente dando a impressão ao usuário que o sistema é único e integrado. Durante esse processo o *Middleware* terá o papel de interpretar e traduzir as informações passadas pelo componente solicitante e fornecê-las num formato que seja enviado pelo protocolo de transporte até o usuário.

Como segurança das informações obtidas, propõe-se que sejam criptografadas, dentre os quais, sugere-se como tema para futuros trabalhos.

O processo de coordenação se dará a partir da sincronização dos dados. Para aumentar a confiabilidade da entrega da informação será utilizado técnicas de replicação de serviços. E para a escalabilidade que segundo Oliveira Júnior (2014), é a capacidade de suportar o aumento de carga e requisições feitas pelos consumidores de serviços aos *Middleware*. Sugere-se para fazer o balanceamento de carga, o mecanismo de *load-balancing*, que segundo o autor mencionado,

objetiva reduzir ou aumentar a carga, dependendo se o servidor tiver parado ou iniciado.

Os usuários não acessam diretamente os servidores de mídia, em vez disso, enviam instruções SQL ao middleware, o qual executa chamadas de API aos servidores de mídia e envia instruções SQL aos servidores de banco de dados. As instruções SQL podem combinar operações tradicionais em dados simples com operações especializadas em dados complexos. O middleware de banco de dados evita que o usuário tenha de conhecer uma API distinta para cada servidor de mídia. Além disso, com esse middleware, a localização não é importante, pois os usuários não precisam saber onde os dados complexos residem.

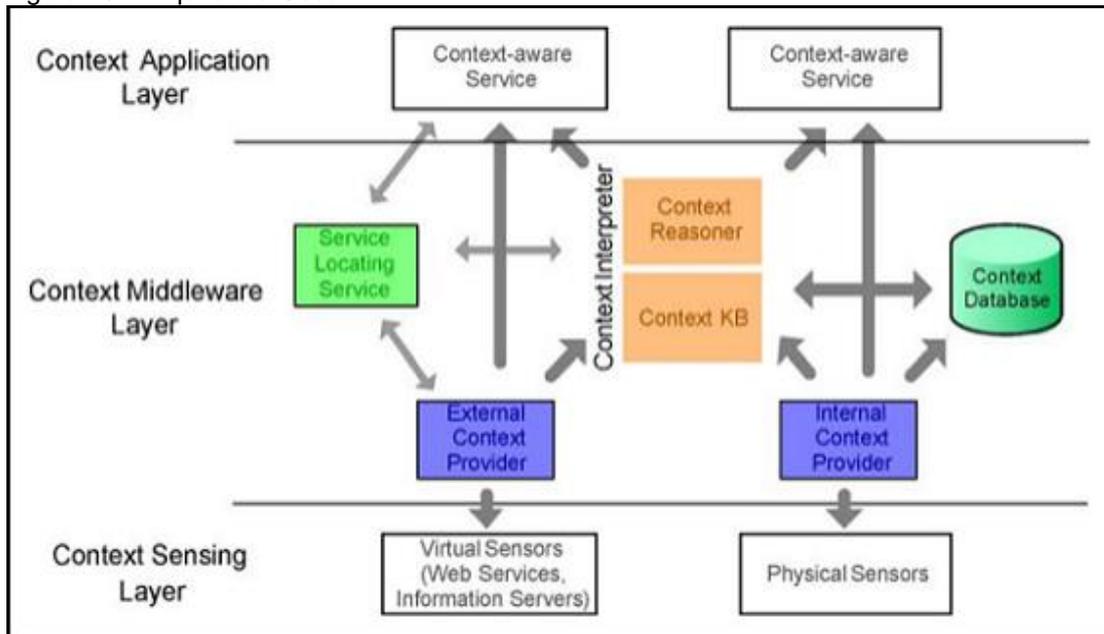
2.2.5.1 Socam

Service Oriented Context Aware Middleware (SOCAM) é um *middleware* que usa RMI como tecnologia de comunicação, adota um modelo de contexto que se baseia em ontologias, o que evita ambiguidades e habilita a utilização de técnicas poderosas de inferências de contexto, interoperabilidade sintática e semântica e compartilhamento de conhecimento entre diferentes domínios. SOCAM determina um modelo de contexto elaborado por duas camadas de ontologias: a camada mais acima é formada por uma ontologia genérica que representa conhecimentos gerais sobre ambientes da computação ubíqua (GU; PUNG; ZHANG, 2005).

Segundo Gu, Pung e Zhang (2005), a segunda camada é composta por ontologias de domínio específico que estabelecem detalhes sobre conceitos e suas propriedades em cada sub-domínio de ambientes ubíquos (por exemplo, ontologias específicas para ambientes industriais, escritórios, hospitais, residências, trânsito, viagens, entre outros sub-domínios). Tais ontologias podem ser carregadas e descarregadas em caso de dispositivos saírem de um ambiente para outro.

A figura 13 apresenta a arquitetura socam.

Figura 13 - Arquitetura Socam.



Fonte: Gu, Pung e Zhang (2005).

Lopes (2011) aponta que a arquitetura do SOCAM é composta pelos seguintes componentes:

- context providers*: são provedores de contexto disponibilizados como serviços que adquirem as informações de contexto e se classificam em provedores de contexto internos e provedores de contexto externos. Os primeiros são provedores que obtêm informações diretamente dos sensores e os provedores externos obtêm informações de contexto de fontes externas, como por exemplo, servidores remotos;
- service locating service*: é o repositório de serviços que permite que aplicações encontrem diferentes provedores de serviços, e para tal, cada provedor de serviço precisa publicar seus serviços neste componente;
- context interpreter*: sua responsabilidade é modificar informações de contexto de baixo nível em informações de mais alto nível. Esse componente se compõe por uma máquina de inferências e por uma base de conhecimento. A máquina de inferência pode utilizar diferentes tecnologias de raciocínio e a base de conhecimento fornece *Application Programming Interface* (API) para que outros componentes tenham acesso às informações de contexto que foram sensoriadas ou inferidas.

É nesse componente que ficam armazenadas as ontologias e suas instâncias;

d) *context aware services*: são as aplicações que usam a infraestrutura SOCAM e seus componentes para acessar as informações de contexto.

SOCAM, portanto, divide um domínio de Computação Pervasiva em vários subdomínios, definindo ontologias individuais em cada subdomínio. Cada uma destas ontologias implementada em OWL provê um vocabulário especial, usado para representar e compartilhar conhecimento de contexto (LOPES et al., 2017.)

Ainda conforme Lopes (2017) a arquitetura SOCAM, permite a rápida prototipação de aplicações conscientes de contexto em ambientes pervasivos. Essa arquitetura utiliza uma ontologia genérica chamada de *Con-24 text Ontology* (CONON) e é completado por ontologias de domínio. Todas implementadas utilizando OWL.

2.2.5.2 Gaia

Gaia é um *middleware* distribuído que coordena entidades de software e dispositivos heterogêneos presentes em um espaço físico, gerencia recursos e serviços de um espaço, abstraindo a complexidade e heterogeneidade associada ao ambiente. Foi desenvolvido pela universidade de *Urbana-Champaign*, onde o desenvolvimento do *middleware* foi o resultado de 6 anos de pesquisa (HAEZEL, 2012).

De acordo com Haezel (2012), o Gaia provê serviços de localização, contexto, eventos e repositórios de informações sobre espaços ativos e gerenciados. Os principais blocos de construção do Gaia são ilustrados na figura 14.

Figura 14 - Arquitetura Gaia



Fonte: Haezel (2012).

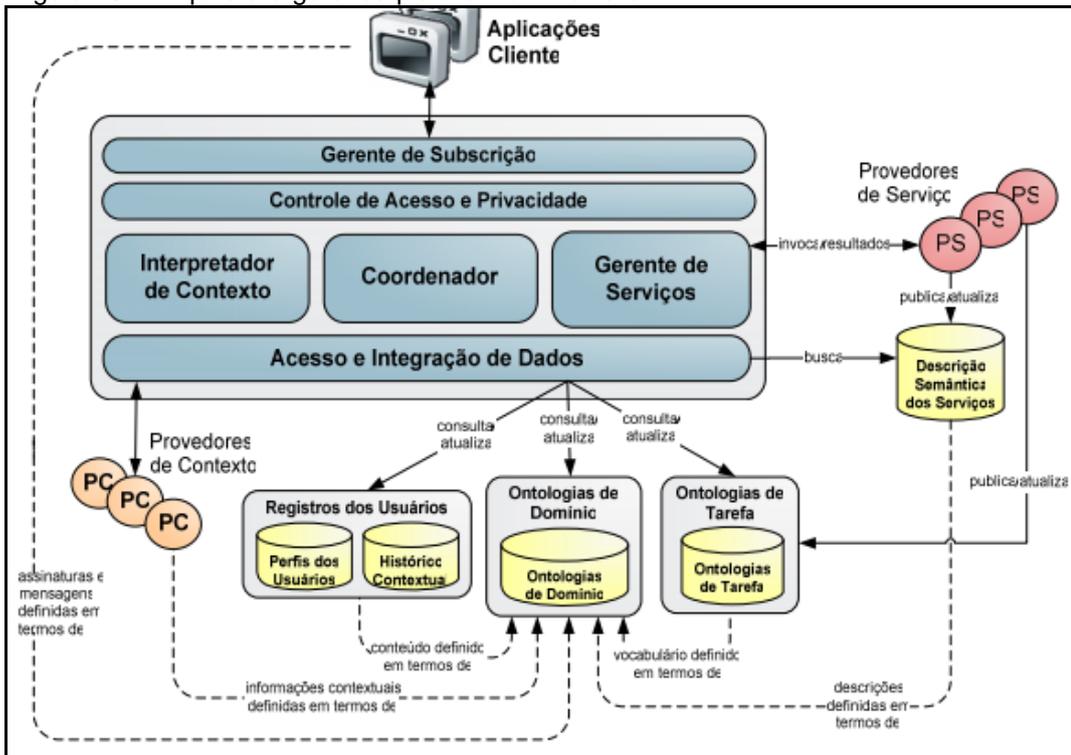
Gaia é uma infraestrutura para ambientes inteligentes através da computação pervasiva. A principal ideia do Gaia é integrar dispositivos com o ambiente físico, permitindo que as entidades virtuais e físicas interajam de maneira satisfatória (MILLER; MCBURNEY, 2007).

Ontologias são utilizadas no Gaia para controlar a diversidade e complexidade dos recursos disponíveis no ambiente, provendo uma taxonomia padrão para os diferentes tipos de entidades e permitindo a descoberta semântica e interoperabilidade entre as entidades. As ontologias são modeladas utilizando DAML+OIL (RANGANATHAN et al, 2004).

2.2.5.3 *Infraware*

Infraware é um middleware baseado em web Services com para o desenvolvimento, construção e execução de aplicações móveis sensíveis ao contexto. Foi definido visando o atendimento a vários requisitos funcionais presentes em ambientes sensíveis ao contexto e a integração desses em uma infraestrutura única, formando uma arquitetura flexível e adequada ao desenvolvimento de aplicações ubíquas reais.

Figura 15 – Arquitetura geral da plataforma Infraware



Fonte: Pessoa.(2006).

Uma das principais características da Infraware é o uso de conceitos da Web Semânticas: ontologias especificam modelos formais extensíveis que descrevem não somente o domínio das aplicações, mas também os serviços. A plataforma tem como componentes:

- a) gerente de subscrição: permite remover, adicionar ou atualizar pedidos de subscrição. Trata-se de sentenças e expressões trocadas entre as aplicações e a plataforma através de uma linguagem específica.
- b) controle de acesso e privacidade: atua como filtro sobre o fluxo de dados entre a plataforma e as aplicações, com base em um conjunto de restrições envolvendo preferencias e políticas de privacidade estabelecer limites de visibilidade para os dados coletados.
- c) Interpretador de contexto: responsável pela manipulação, derivação, refinamento e inferência de contexto elaborados a partir de informações contextuais, com o objetivo de tornar as informações disponíveis para as aplicações de forma transparente, reduzindo a complexidade no tratamento dos dados.
- d) acesso e integração de dados: capaz de tratar e manipular informações

oriundas de diversas fontes de contexto, oferecendo também uma interface homogênea e transparente de acesso aos dados;

- e) gerente de serviços: atividade de publicação, descoberta, seleção e composição de serviços.
- f) coordenador: Suas principais funções estão relacionadas com a monitoração e o controle do estado da plataforma. (PEREIRA FILHO, 2006).

3 TRABALHOS CORRELATOS

Este capítulo tem como finalidade abordar trabalhos relacionados aos assuntos abordados, como CSS e modelagem de sistemas. Desse modo, apresenta-se uma monografia de graduação e três dissertações de mestrado com intuito de apresentar as aplicações delimitadas pelos pesquisadores neste modelo.

3.1 DESENVOLVIMENTO DE UMA SOLUÇÃO SENSÍVEL AO CONTEXTO COMO SUPORTE A UM SISTEMA DE TRANSPORTE PÚBLICO INTELIGENTE

Esta pesquisa tratou-se de uma monografia de graduação apresentado ao curso de Ciências da Computação da Universidade Federal da Bahia, no ano de 2010 por Luiz Rodrigo Rosário Caldas, com o objetivo de desenvolver uma solução computacional sensível ao contexto que auxilie um sistema inteligente de transporte capaz de disponibilizar aos usuários de transporte público o horário de chegada e a localização atual do ônibus.

Caldas (2010), menciona que, apesar deste Sistema fazer-se presente nos países desenvolvidos, encontra-se maiores barreiras nos países em desenvolvimento, porque muitos deles ainda não possuem informações exclusivas e atualizadas sobre os ônibus, o que resulta em problemas, por exemplo, para se estimar tempo de chegada às paradas.

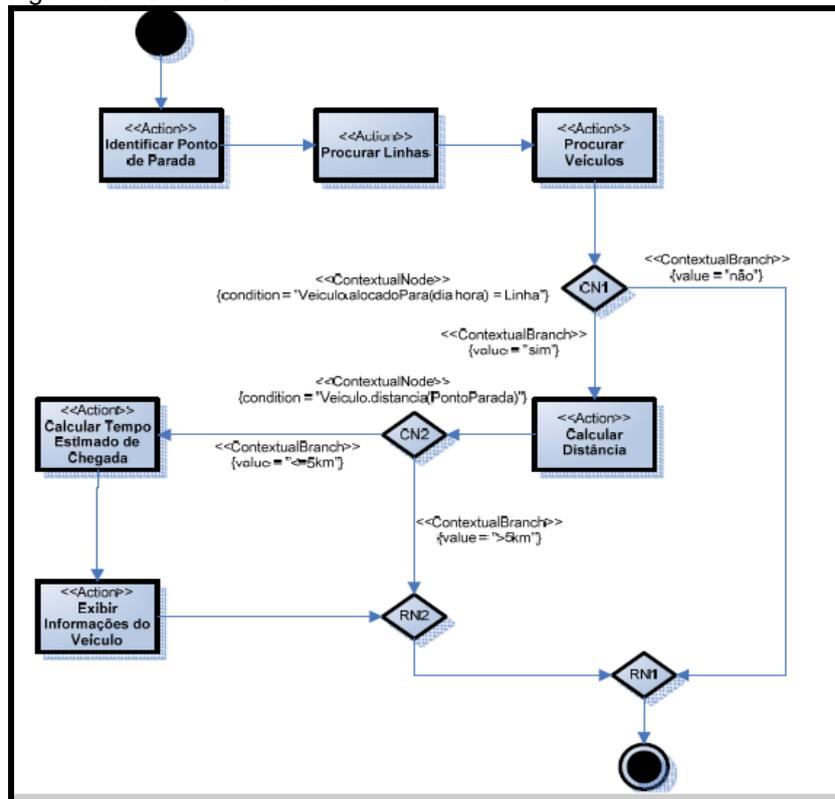
O projeto apresentado pelo autor dessa monografia, contempla as fases de aquisição, processamento e uso de elementos de contexto referentes aos ônibus, pontos e rotas, como por exemplo, a localização, velocidade e o tempo estimado da chegada a um dado ponto de parada. Para oferecer essa funcionalidade, o sistema considera a identificação e a localização dos ônibus assim como, os pontos de parada e a intensidade de congestionamento como elementos de contexto. A intensidade de congestionamento é inferida a partir do processamento das velocidades desempenhadas pelos próprios veículos.

A localização e a velocidade dos ônibus, é obtida através do GPS instalados dentro de cada veículo, o qual funcionam como sensores. O GPS é composto por vários elementos de hardware e software, sendo que um destes

elementos é o protocolo para enviar e receber mensagens (*National Marine Electronics Association - NMEA*), no formato arquivo texto com os dados de latitude, longitude, e velocidade do veículo (CALDAS, 2010).

Outra aplicação para a construção das rotas foi o Google Maps, a partir dessas aplicações, o autor utilizou como técnica para analisar o uso do contexto, o grafo contextual, conforme apresentado na figura 15.

Figura16 – Grafo Contextual.



Fonte: Caldas (2010).

Segundo Caldas (2010), com o grafo contextual é possível observar qual o tipo de lógica deve ser implementada de forma que o uso do contexto contemple as variações de comportamento identificada. O grafo contextual utilizado pelo autor da monografia foi traduzido em regras contextuais conforme apresentado nos quadros 3, 4 e 5.

Quadro 3 – Regra Contextual #1

<p>Condições</p> <p>Veiculo.alocadoPara(dia, hora) = Linha</p> <p>Veiculo.distancia(PontoParada) <=5</p> <p>Ações:</p> <p>CallBehavior (“Identificar Ponto de Parada”)</p> <p>CallBehavior (“Procurar Linhas”)</p> <p>CallBehavior (“Procurar Veiculos”)</p> <p>CallBehavior (“Calcular Distancia”)</p> <p>CallBehavior (“Calcular Tempo Estimado de Chegada”)</p> <p>CallBehavior (“Exibir Informações do Veiculo”)</p>

Fonte: Caldas (2010).

Quadro 4 – Regra Contextual #2

<p>Condições</p> <p>Not (Veiculo.alocadoPara(dia, hora) = Linha)</p> <p>Ações:</p> <p>CallBehavior (“Identificar Ponto de Parada”)</p> <p>CallBehavior (“Procurar Linhas”)</p> <p>CallBehavior (“Procurar Veiculos”)</p>
--

Fonte: Caldas (2010).

Quadro 5 – Regra Contextual #3

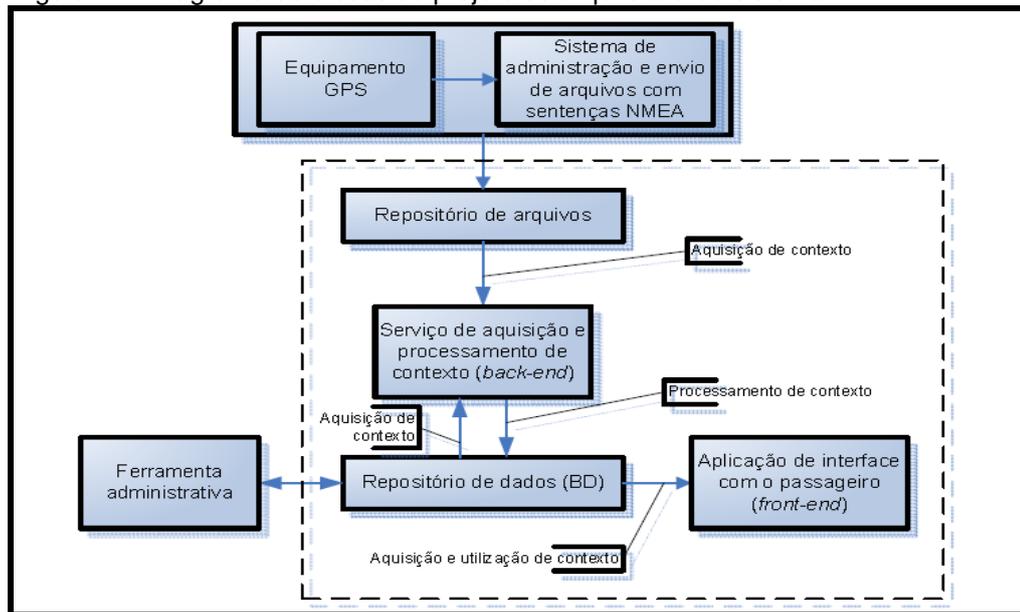
<p>Condições</p> <p>Veiculo.alocadoPara(dia, hora) = Linha</p> <p>Veiculo.distancia(PontoParada) >5</p> <p>Ações:</p> <p>CallBehavior (“Identificar Ponto de Parada”)</p> <p>CallBehavior (“Procurar Linhas”)</p> <p>CallBehavior (“Procurar Veiculos”)</p> <p>CallBehavior (“Procurar Distancia”)</p>

Fonte: Caldas (2010).

A intensidade do congestionamento, é inferida através dos próprios dados sobre localização e velocidade emitidos pelos ônibus, mas para saber o quanto um trecho congestionado influenciará no tempo de viagem, é preciso conhecer a exata rota que cada um destes efetua, informação que poderá ser armazenada em um banco de dados (CALDAS, 2010).

A figura 16 mostra o projeto de arquitetura que engloba a aquisição, processamento e utilização de contexto.

Figura 17 - Diagrama de blocos do projeto de arquitetura do sistema.



Fonte: Caldas (2010).

O CSS modelado e analisado foi codificado na linguagem de programação C#. A aplicação *back-end* consistiu na implementação de um serviço, enquanto a aplicação *front-end* caracterizou-se como uma aplicação web, principalmente devido ao requisito de disponibilização de mapa através da API do Google Maps.

No que se refere a banco de dados, o MySQL foi a opção escolhida. A administração do mesmo foi feita com auxílio de um software chamado *Toad*, versão 4.5. Em termos de codificação, a geração da camada de dados, baseada no modelo relacional criado em banco, foi feita utilizando um framework intitulado 4A BOF.

A utilização do protocolo NMEA - 0183 serviu para extrair informações emitidas por um equipamento GPS, constituindo uma das principais fontes de contexto utilizado pelo autor.

Com o requisito não funcional NF1.1 definiu que o tipo de sentença GPRMC foi sucintamente útil para adquirir as informações necessárias como: data, hora, latitude, longitude e velocidade.

Para o módulo de aquisição, o autor desenvolveu uma API (+MEARReaderAPI) o qual transformou as sentenças e *strings* de um arquivo numa matriz, fazendo desta API algo genérico e passível de utilização por qualquer outro sistema que necessitasse de dados de um equipamento GPS que utilize tal

protocolo. O autor, desenvolveu em contrapartida, um adaptador (+*MEARReaderAdapter*) que atuava como um *parser*.

Para realizar a interação com o Google Maps, o autor utilizou dois momentos: (i) ao adquirir o elemento de contexto distância/Próximo/Ponto; (ii) ao renderizar mapa georreferenciado na aplicação front-end.

No primeiro momento utilizou o *Google Directions* API com as informações de origem (*origin*), destino (*destination*) e sensor para implementar na sua aplicação.

No segundo momento, o autor recorreu ao *JavaScript* API, o que permitir a renderização de mapas em aplicações *web*.

O protocolo NMEA e a família de APIs do *Google Maps* nos ofereceram a base necessária para os aspectos relacionados a georreferenciamento, enquanto o algoritmo de atualização ponderada incremental nos permitiu, de forma simples, definir um elemento contextual (velocidade ponderada) que refletisse o aspecto da intensidade de congestionamento presente nas vias de um centro urbano.

O sistema foi avaliado com dados reais coletados de uma rota criada para a cidade de Salvador, com o estudo que mostra a viabilidade e potencial da solução proposta. Como resultado, observou-se que o sistema se comporta conforme o contexto que lhe é apresentado e dispõe de resultados mais confiáveis que aplicações tradicionalmente estáticas (CALDAS, 2010).

3.2 SELEÇÃO DE SENSORES SENSÍVEL A CONTEXTO PARA REDUÇÃO DO CONSUMO DE ENERGIA EM SMARTPHONES

Este correlato trata de uma Dissertação de mestrado apresentada no curso de Pós-graduação em Ciência da Computação na Universidade Federal de Pernambuco no ano de 2015 por Pamela Thays Lins Bezerra.

O problema que norteou o desenvolvimento dessa dissertação foi com base no argumento: com a evolução das redes moveis e *wifi*, com o aumento do processamento e da capacidade de armazenamento, é possível a execução de softwares complexos e aplicativos inteligentes. Por outro lado, com o aumento no número de sensores, como GPS e acelerômetro, e de periféricos, como microfone e

bluetooth, possibilita a coleta de dados do usuário, como localização e mobilidade. Contudo, a utilização de muitos sensores associados a um alto processamento e compartilhamento de dados ocasiona o aumento do consumo de energia destes dispositivos. As baterias usadas nos smartphones não evoluíram no mesmo ritmo que os demais componentes. Desta forma, surge um novo desafio: como prover mais serviços e de melhor qualidade sem desperdiçar energia? Por este motivo, a autora menciona que há muitos artigos sobre *hardware* e *software* utilizados em dispositivos móveis voltados para desenvolver soluções em eficiência energética, ou seja, com intuito de reduzir o consumo de energia sem que os dispositivos percam a performance e a qualidade dos serviços oferecidos.

Desse modo, Bezerra (2015), apresentou como proposta, um método que visa reduzir o consumo de energia baseado em contexto para *smartphones* através da seleção de sensores, com o objetivo de entender quais atividades o usuário executa assim como, o ambiente que ele se encontra para selecionar apenas os sensores adequados com a realidade atual do usuário. Segundo o autor, com isso é possível evitar o desperdício de energia causado por diversos sensores ativos.

No estudo foram utilizadas regras de associação para relacionar diferentes características do usuário, como comportamento e a localização, de modo a inferir todo o contexto do usuário de maneira simples e rápida. Para validar o estudo, foi desenvolvido um serviço *Android* e realizados experimentos de maneira que o serviço contenha regras estabelecidas assim como a monitoria do estado atual do *smartphone*. Desta forma, sempre que o usuário se movimenta, o sistema verifica se o nível de bateria está superior a 15%, para que os serviços de seleção de sensores sejam ativados. Este serviço verifica o status atual do aparelho e compara com as regras de associação estabelecidas para verificar caso o usuário esteja *indoor* ou *outdoor*. Com esta verificação ativam-se apenas os sensores necessários para aquele momento e cenário.

Para a realização do estudo foram necessários os seguintes sensores: GPS, *Wifi*, redes (GSM, 3G), *bluetooth*. Além desses elementos, foi desenvolvido um serviço *Android* capaz de coletar dados do usuário, verificar as regras e selecionar ao menos um sensor. Foi preciso seguir 5 etapas para a concretização do método:

- a) especificação de contexto, listado as características o qual desejou-se

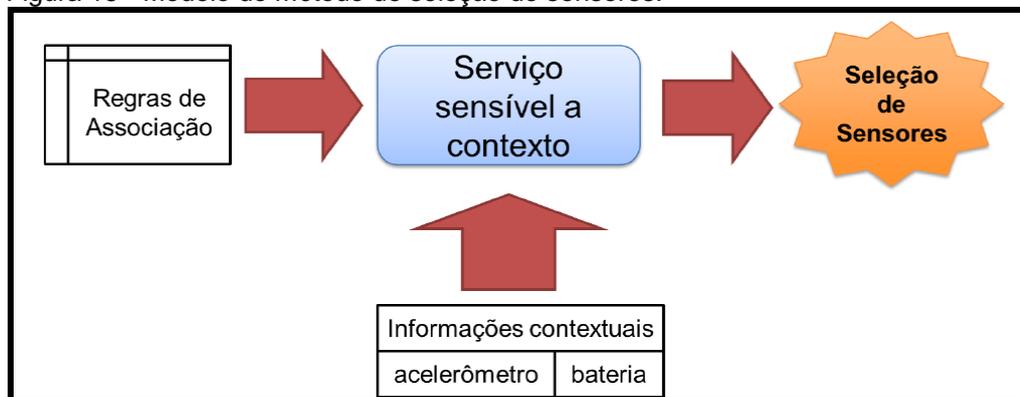
estudar;

- b) coleta de dados, análise de vários usuários;
- c) análise dos dados, processamento;
- d) extração das regras;
- e) desenvolvimento do serviço.

Nos testes a autora constatou através de seu experimento que ocorreu a economia de energia nos aparelhos. E com base nos dados obtidos, foi possível constatar que o serviço aumentou em média duas horas o tempo de vida da bateria.

A figura 17 apresenta o modelo proposto por Bezerra (2015).

Figura 18 - Modelo do método de seleção de sensores.



Fonte: Bezerra (2015).

3.3 UBIFEX: UMA ABORDAGEM PARA MODELAGEM DE CARACTERÍSTICAS DE LINHA DE PRODUTOS DE SOFTWARES SENSÍVEIS AO CONTEXTO

O correlato a seguir trata de uma dissertação de mestrado do curso de Pós-graduação em Engenharia de Sistemas e Computação no ano de 2009 na Universidade Federal do Rio de Janeiro de Paula Cibele Cavalcante Fernandes.

Como problema norteador de sua pesquisa, Fernandes (2009) fez as seguintes considerações:

[...] as informações de contexto são um conceito chave na definição das aplicações sensíveis ao contexto. Assim, para a construção de uma Linha de Produtos de Software Sensíveis ao Contexto (LPSSC), ou seja, uma linha de produtos que apoia o desenvolvimento de produtos sensíveis ao contexto, é importante que as informações de contexto relevantes para o domínio sejam representadas no modelo de características, além do impacto dessas informações na reconfiguração dos produtos em tempo de execução. (FERNANDES, 2009, p. 78).

Porém, a autora menciona que grande parte das abordagens existentes para LPS, as variabilidades de cada produto manifestam-se de forma estática, focando apenas em uma adaptação em tempo de desenvolvimento, onde diferentes configurações de um produto são desenvolvidas com intuito de atender às necessidades dos clientes e às características do ambiente de execução. No entanto, essas abordagens não são adequadas para lidar com o dinamismo exigido pelas aplicações sensíveis ao contexto, de maneira que possam ser estendidas para que sejam capazes de acomodar características de adaptabilidade e sensibilidade ao contexto. Outras abordagens apresentam essa preocupação de tratar variabilidade de forma dinâmica, porém elas não possuem uma forma de representação explícita das informações de contexto e da forma como elas influenciam na adaptação dinâmica dos produtos com base nessas informações. Além disso, essas abordagens não apresentam mecanismos para verificar se essas adaptações realizadas em tempo de execução estão consistentes com as restrições definidas no modelo em tempo de desenvolvimento.

Diante do problema, o objetivo deste trabalho apresentou uma abordagem para a modelagem de características no desenvolvimento de LPSSC, o qual foi possível representar as informações de contexto relevantes para o domínio desde as fases iniciais de desenvolvimento, facilitando o entendimento dos envolvidos no processo.

Este trabalho de pesquisa propôs uma abordagem para MC de linha de produtos sensíveis ao contexto, denominada UBIFEX, cujo objetivo era permitir a representação das informações de contexto relevantes para o domínio desde as fases iniciais de desenvolvimento, facilitando o entendimento dos envolvidos no processo.

A pesquisadora propôs um mecanismo para verificação da consistência das configurações dos produtos em diferentes cenários de execução com base nas restrições definidas em modelos de características, realizado também um estudo preliminar para avaliar o mecanismo de verificação. Além disso, um protótipo foi desenvolvido para viabilizar a aplicação da abordagem proposta.

Para atingir os objetivos, foram detectadas algumas etapas: primeiro analisar as técnicas existentes para modelagem de contexto no desenvolvimento

tradicional de aplicações sensíveis ao contexto, e depois identificar os elementos necessários no modelo de características para representar esse tipo de informação. Na terceira etapa seria o momento de definir os mecanismos de extensão no modelo, e por último, definir um mecanismo para verificação da configuração dos produtos em relação às variações de contexto com base em modelos de características.

Como forma de obter os requisitos foi proposto a abordagem *UbiFEX* que possui como principal objetivo de fornecer mecanismos para apoiar a MC de LPS sensíveis ao contexto. São dois componentes responsáveis, *UbiFEX-Notation* uma notação para a representação dos conceitos relacionados a sensibilidade ao contexto; e *UbiFEX-Simulation* um mecanismo para verificação da configuração dos produtos em relação as variações de contexto, com base em modelos de características construídos utilizando *UbiFEX-Notation*.

Um estudo foi realizado para avaliar o processo de verificação, e contribuiu de forma significativa para a melhoria da abordagem e motivou o desenvolvimento de um protótipo para apoiar sua aplicação. O protótipo foi desenvolvido no contexto de ambiente *Odyssey*, envolvendo etapas de engenharia de domínio e engenharia de aplicação.

3.4 RECOMENDAÇÃO ADAPTATIVA E SENSÍVEL AO CONTEXTO DE RECURSOS PARA USUÁRIOS EM UM CAMPUS UNIVERSITÁRIO

Este tema tratou-se de uma Dissertação de mestrado apresentada por Guilherme Medeiros Machado em 2014 na Universidade Federal do Rio Grande do Sul em Porto Alegre, para obtenção do Grau de Mestre em Ciência da Computação.

Como problema de pesquisa, Machado (2014) alegou que, o ambiente de um campus universitário, é formado por vários recursos do mundo físico (restaurantes, bibliotecas, salas de aula, livros), do mundo virtual (artigos científicos, softwares, laminas de uma aula) e de pessoas que utilizam esses recursos. Devido essa gama de recursos espalhados neste ambiente, escolher um conjunto que melhor se adeque as necessidades do usuário não é uma tarefa fácil. Uma das razões que levam as pessoas escolherem um conjunto de recursos que não

atenderá suas necessidades da melhor maneira possível.

Desta forma, a escolha de um recurso a ser utilizado torna-se uma tarefa difícil, já que a pessoa fica cognitivamente sobrecarregada por ter que analisar e filtrar os objetos que possam ser de seu interesse. Esta sobrecarga cognitiva acaba levando a pessoa a escolher um conjunto de recursos que não atenderá suas necessidades da melhor maneira possível. Para isso, o autor propôs a utilização de sistemas de recomendação, de maneira que este recurso seja capaz de lidar com informações diferentes das manipuladas pelos sistemas tradicionais, levando-se em conta a ciência dos recursos e do relacionamento deles com o usuário no ambiente.

A partir do problema de pesquisa, o autor apresentou como objetivo geral, propôs uma abordagem sensível ao contexto e de interface adaptativa, a ser utilizada em um dispositivo móvel, para recomendar objetos de aprendizagem próximos e interessantes aos usuários, sendo este um membro da comunidade acadêmica em um campus universitário.

Machado (2014), mencionou a situação de um estudante na biblioteca a procura de um livro de seu interesse, porém devido à grande quantidade de livros na biblioteca o estudante acaba demandando uma parte de seu tempo para encontrar o livro que pode não suprir suas necessidades. Com intuito de resolver esse problema, criou-se uma abordagem sensível ao contexto e adaptativa para recomendação de objetos de aprendizagem a comunidade de uma universidade. Desenvolveu-se um sistema na plataforma Android para dispositivos móveis, o qual, pode-se trabalhar de maneira pervasiva estando integrado a vida do usuário sem que ele perceba, apesar disso a interface permite que o usuário selecione quais recomendações lhe interessam. Outra característica que o torna sensível ao contexto, é de que o sistema é capaz de examinar e reagir a mudanças do ambiente.

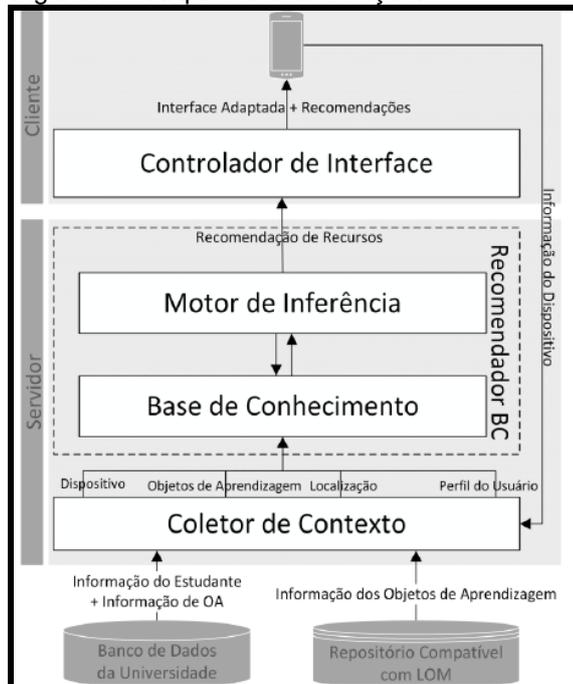
A variável de contexto utilizada durante o processo de recomendação é a localização do usuário e dos objetos de aprendizagem. A base de conhecimento utilizada pela abordagem é composta do modelo ontológico de contexto, um banco de dados de informações contextuais que é alimentado com informações coletadas do ambiente e do usuário, um motor de inferências lógicas e regras descritas em forma de lógica de primeira ordem. Esta base de conhecimento tem como saída um conjunto de objetos de aprendizagem recomendados ao usuário, bem como a

localização de tais objetos. Tais informações são apresentadas ao usuário através da aplicação desenvolvida.

O modelo ontológico de contexto foi construído em OWL, através do *Software Protege*, e tem o usuário como elemento central. O modelo é apresentado em quatro dimensões de informação contextual, a) informações sobre o perfil do usuário, b) informações sobre localização, c) informações sobre os elementos tecnológicos e d) informações sobre objetos de aprendizagem.

A arquitetura da solução é dividida em três componentes, o Coletor de Contexto e o Recomendador Baseado em Conhecimento (Recomendador BC) no lado de Servidor e o Controlador de Interface no lado do cliente (figura 18).

Figura 19 - Arquitetura da solução.



Fonte: Machado (2014).

O trabalho apresentou um modelo ontológico de contexto, utilizando quatro dimensões de informações e que tem o usuário como seu elemento central, as dimensões de informação deveriam atender as informações relativas ao perfil do usuário, a localização, a tecnologia e os objetos de aprendizagem a serem recomendados. O modelo ontológico foi instanciado com as informações coletadas do ambiente, do banco de dados da universidade e de repositórios de objetos de aprendizagem. Junto a esse modelo foram construídas regra responsáveis pela

filtragem dos objetos de aprendizagem, tais regras foram escritas em forma de antecedente e conseqüente e utilizou-se os conceitos e relacionamentos do modelo para isso.

Segundo Machado (2014), a vantagem de construir as regras responsáveis pela recomendação dos objetos de aprendizagem era de tornar a semântica de comportamento da aplicação explícita e inteligível tanto por humanos quanto por agentes computacionais. O autor desenvolveu um estudo de caso, com intuito de avaliar a eficácia da proposta, com um cenário de aplicação e a atual execução das regras de recomendação. O motor inferência Pellet foi utilizado para a execução das inferências lógicas sobre as regras construídas. Segundo o autor, os resultados obtidos através do estudo de caso mostraram que a abordagem comprovou eficaz em realizar as recomendações propostas.

3.5 COMPARATIVO DOS TRABALHOS CORRELATOS

O quadro 6 apresenta o comparativo dos trabalhos correlatos. Os pontos levantados para análise foram: a necessidade do uso dos sistemas sensível ao contexto, os elementos utilizados pelos pesquisadores, as técnicas utilizadas, o gerenciamento (aquisição, processamento, disseminação), uso do contexto que trata dos resultados obtidos nos correlatos.

Buscou-se identificar os correlatos no quadro 6 como C1, C2, C3 e C4. A análise dos correlatos será apresentado na análise e discussão dos resultados desse trabalho.

Quadro 6 - Comparativo entre os correlatos.

SISTEMAS	C1	C2	C3	C4
Necessidade do SSC	Sistema inteligente de transporte que disponibiliza o horário de chegada e a localização atual do ônibus.	Método que visa reduzir o consumo de energia baseado em contexto para smartphones através da seleção de sensores.	Abordagem para a modelagem de características no desenvolvimento de LPSSC.	Abordagem sensível ao contexto e de interface adaptativa, para recomendar objetos de aprendizagem.
Elementos	Localização, pontos de parada e intensidade de congestionamento.	Usuário, dispositivo, mobilidade, sociabilidade, interatividade e localização.	Rede, dispositivo móvel e usuário	Usuário, localização, dispositivos, objetos de aprendizagem.
Fontes de contexto	GPS, Google Maps,	GPS, <i>wifi</i> , Redes (GSM, 3G, LTE), acelerômetro e <i>bluetooth</i> .	-	API Google maps, motor de inferência.
Representação	Grafo contextual	Regras de associação	-	Ontologia e metadados
Gerenciamento	Aquisição: Repositório de dados, GPS, NMEA – 0183. Processamento: Informações obtidas pelo sensor GPS, <i>Google Maps</i> e NMEA e desenvolveu um serviço e dois tipos de APIs. Disseminação: as informações processadas e armazenadas no (MySQL).	Aquisição: microfone, <i>bluetooth</i> . GPS, redes de telefonia ou <i>wifi</i> , acelerômetro, giroscópio Processamento: Cálculo do nível de ruídos em baixa, média e alta. Ligações feitas ou recebidas, ler e-mail ou assistir vídeos (nível alto de interação), perda e falta de ativação (nível baixo de interação). analisou-se os sensores de luz, proximidade e magnetômetro. A mobilidade do usuário é avaliada pelo acelerômetro.	UbiFEX-Notation: inclui elementos para representar a influência dessas informações na configuração dos produtos. UbiFEX-Simulation: mecanismo para a verificação de consistência da configuração dos produtos em relação às variações de contexto, com base em modelos de características construídos utilizando <i>UbiFEX-Notation</i> .	Aquisição: coletor de contexto; recomendador BC; Controlador de interface; Processamento: interface adaptativa (<i>web services</i>). O sistema envia ao Coletor de Contexto. A informação de localização é obtida através do GPS ou do roteador <i>wireless</i> . Disseminação: Motor de Inferência é um conjunto de pares que é enviado por <i>web services</i> ao Controlador de Interface.
Uso do contexto pelo usuário final	O sistema busca a informações como a localização corrente de um ônibus e o tempo estimado de sua chegada a um dado ponto de parada.	Verificou-se que o serviço aumentou o tempo de vida das baterias em média de 22,82%, sendo 18 horas e 25 minutos o aumento mais significativo.	Avaliação preliminar do mecanismo de verificação <i>UBIFEX - Simulation</i> , com intuito de utilizar os resultados para melhoria da abordagem servindo para o planejamento de uma avaliação mais completa;	A utilização de ontologias e de informações de contexto mostram-se eficazes na recomendação de recursos em ambientes pervasivos.

Fonte: Desenvolvido pela autora.

4 MCSS: MODELO DE CARACTERÍSTICAS PARA REPRESENTAR INFORMAÇÕES DE CONTEXTO EM UM DOMÍNIO DE APLICAÇÃO

Esta pesquisa consistiu em desenvolver um modelo de características para representar informações de contexto relevantes em um domínio de aplicação sensível as mudanças em tempo de execução. Para atingir o objetivo desta pesquisa visou analisar os sistemas sensíveis ao contexto observando os elementos, comportamento nas variações de contexto; a identificação dos elementos que representam informações de contexto e situações de mudança para compor o modelo de características em um domínio; estudo das técnicas para modelagem de características que representam informações de contexto, e estudos dos mecanismos para verificação das mudanças de contexto com base no modelo de características

4.1. METODOLOGIA

Dentro das linhas de pesquisa do curso de Ciência de Computação da UNESC, este projeto se insere nas linhas Engenharia de Sistemas e Sistemas Multimídia.

Este Trabalho de Conclusão de Curso foi desenvolvido com base num estudo bibliográfico. Segundo Amaral (2007, p. 5):

A pesquisa bibliográfica é uma etapa fundamental em todo trabalho científico que influenciará todas as etapas de uma pesquisa, na medida em que der o embasamento teórico em que se baseará o trabalho. Consistem no levantamento, seleção, fichamento e arquivamento de informações relacionadas à pesquisa.

As etapas de desenvolvimento da pesquisa foram: inicialmente realizar um levantamento de materiais que contribuiriam com a fundamentação teórica necessária para o desenvolvimento da pesquisa. Os principais recursos foram: dissertações, TCCs, monografias, teses, revistas eletrônicas, artigos, e livros em geral.

Após o levantamento bibliográfico, procurou-se investigar trabalhos já realizados, sendo estes, uma monografia e três dissertações de mestrado, que

apresentavam diferentes situações de desenvolvimento de projetos voltados a Sistema Sensível ao Contexto. O critério de escolha foi com base nos trabalhos que apresentavam mais proximidade com o tema pesquisado. Procurou-se descrever a síntese de cada correlato, o qual procurou-se relatar brevemente o problema, objetivo, desenvolvimento e conclusão. Em seguida, organizou-se um quadro contendo a necessidade que levou cada autor desenvolver o projeto, os elementos envolvidos, fontes de contexto, técnicas, modo de gerenciamento e resultados obtidos.

Com base nos dados colhidos e analisados nestes correlatos, desenvolveu-se um modelo de características para representar informações de contexto relevantes em um domínio de aplicação sensível a mudanças em tempo de execução.

4.2 LEVANTAMENTO BIBLIOGRÁFICO

A primeira etapa desta pesquisa foi o levantamento bibliográfico, buscando organizar as referências. O desenvolvimento foi baseado nos materiais já produzidos sobre o assunto, como livros artigos, dissertações, site da internet, projetos. Fontes variadas, como sites específicos sobre Computação sensível ao contexto. Mas tanto no que diz respeito sobre a modelagem de características para sistemas sensíveis ao contexto, pretendeu-se abordar uma bibliográfica que fundamentasse o tema de forma acadêmica.

Através dos estudos analisamos sistemas sensíveis a contexto observando elementos, comportamento nas variações de contexto e o suporte no desenvolvimento de aplicações, identificamos quais elementos que representam informações de contexto e situações de mudança, técnicas para modelagem de características que representam informações de contexto, e os mecanismos para verificação das mudanças de contexto com base no modelo de características.

4.2.1 Descrição do modelo

De acordo com os estudos realizados com a pesquisa bibliográfica, desenvolveu-se um modelo de características para representar informações de

contexto relevantes em um domínio de aplicação sensível a mudanças em tempo de execução. A utilização de informações de contexto permite que desenvolvedores de sistemas possam enriquecer a usabilidade de sua aplicação e, assim, o sistema sensível ao contexto pode reagir a determinadas situações sem a necessidade da interação com o usuário.

Para que um contexto possa ser representado, é fundamental que seja aplicado uma técnica. Um modelo de contexto define tipos, nomes, propriedades e atributos dos elementos nas aplicações sensíveis ao contexto, tais como usuários, dispositivos móveis.

O modelo tenta prever a representação, a busca, a capacidade de troca de informação contextual entre as aplicações. Utilizado a técnica de modelos gráficos, pois facilita a especificação dos conceitos e definição do comportamento do sistema sensível ao contexto, podendo ser traduzido para XML e usa processamento em XML. É uma abordagem com maior foco nos humanos. A partir da modelagem gráfica é possível derivar algum código.

Segundo Fonteles (2013), existem algumas tarefas que um sistema sensível ao contexto segue para seu funcionamento: a) aquisição de informações contextuais, que podem ser através de sensores físicos, virtuais e lógicos – temos como informações obtidas através destes sensores a luminosidade, localização, clima, características do perfil do usuário, contatos, atividade atual, entre outros; b) elevar o nível dessas informações, essa etapa ocorre a interpretação de dados, com a utilização de técnicas, inferências ou transformações de baixo nível para o nível superior; c) armazenar e gerenciar as informações obtidas.

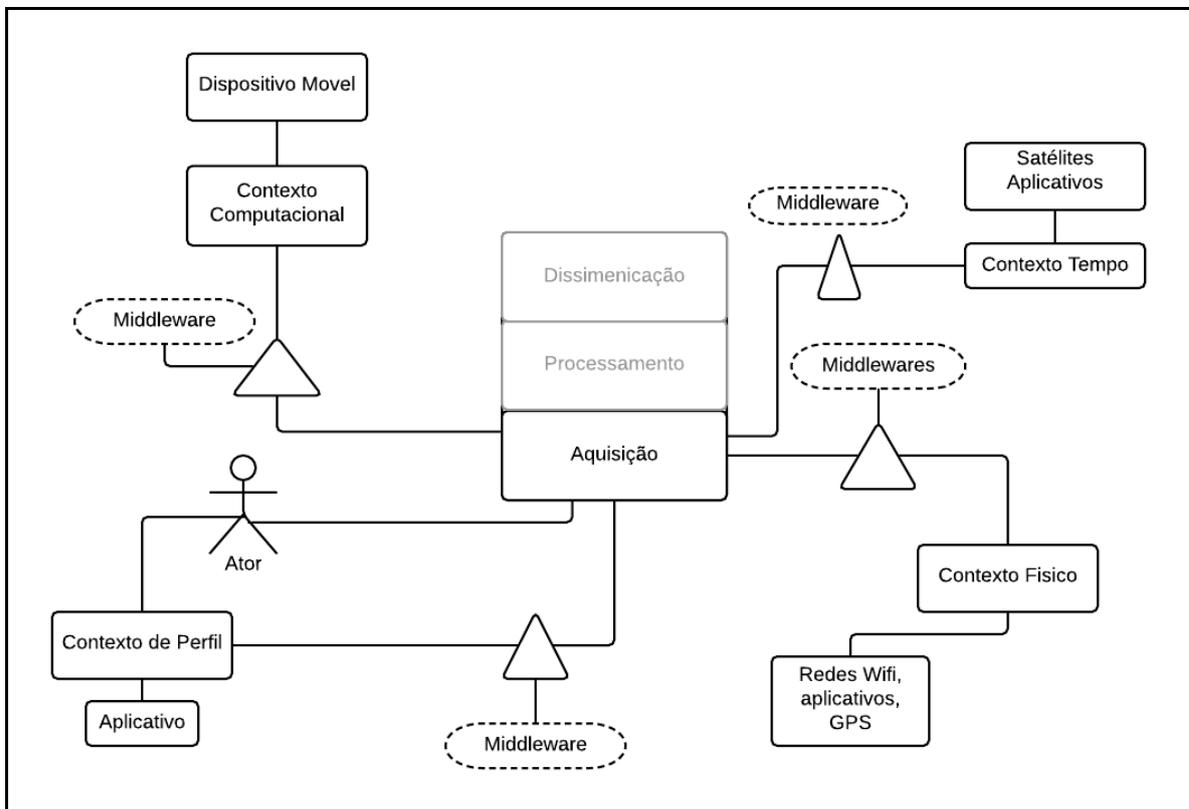
O contexto foi classificado em: computacional, que se refere a informações sobre um sistema computacional e seus recursos, por exemplo, o nível de bateria, consumo de memória, conexões de rede disponíveis; usuário, informações relacionadas ao perfil do usuário, como atividade atual, localização, status; contexto físico, informações sobre o mundo real, que são obtidas por meio de sensores, como a luminosidade, temperatura, localização, sensor de ruído e tempo que são informações relacionadas ao tempo de uma atividade, tem relação com a dimensão do tempo, como a hora do dia, data, estação climática.

As informações podem ser percebidas e obtidas por sensores lógicos e físicos. Quando se trata das informações de perfil do usuário, o próprio usuário

fornece alguns valores a aplicação, ou podem ser obtidas automaticamente por outras aplicações. Encontramos informações contextuais em aplicações que funcionam como serviço, como por exemplo, uma agenda que é composta com compromissos e atividade do usuário. E de forma derivada, o qual pode ser apreendida ou inferida.

No modelo proposto, no processo de aquisição, apresentamos, as fontes de contexto físico, computacional, tempo e perfil do usuário. A figura 19 apresenta o modelo genérico do SSC desenvolvido.

Figura 20 - MCSS: MODELO DE CARACTERÍSTICAS



Fonte: Autor (2018)

As fontes de contexto físico proposto para o modelo é o GPS e o API Google maps, com intuito de obter informações como: localização, coordenadas, ambiente físico, e outras informações, como, pontos de referência próximos do usuário, dentre outras.

Para o processamento e disseminação das informações obtidas, utilizamos o *Middleware*, o qual serve como mediador que se encontra entre as

aplicações distribuídas e suas plataformas que podem ser, processadores e suas arquiteturas, além de funções de baixo nível e APIs dos sistemas operacionais.

O contexto tempo no modelo apresentado implicará nas informações que são utilizadas em conjunto com outras partes de um contexto, informando data e hora ou o intervalo em que uma determinada informação contextual foi percebida ou é relevante, como por exemplo, clima e estação do ano. (GOMINHO; TEDESCO; BELIAN, 2014).

O contexto computacional é responsável pela aquisição de dados de redes, hardware e software que estão sendo utilizados pelo usuário. Este contexto é necessário, por exemplo, para selecionar recursos de aprendizagem adequados ao dispositivo que está sendo usado (GOMINHO; TEDESCO; BELIAN, 2014).

Tais características permitem que sistemas sensíveis sejam capazes de identificar, coletar dados e informações por meio de interfaces de comunicação, essas informações podem ser variações do ambiente ou ações dos usuários, possibilitando a geração de modelos computacionais automáticos, configurados conforme a necessidade do usuário, da aplicação e dos dispositivos da rede. Com essa interação é possível que cada membro do sistema seja capaz de identificar a presença de outros elementos no ambiente, tanto, usuários, como outros dispositivos eletrônicos, sendo assim possível a interação automática entre os elementos, onde pode ser construído um contexto inteligente, que potencializa o uso de sistemas pervasivos. (SILVA, 2013).

4.2.2 Exemplo de uso

O modelo desenvolvido, apresenta etapa de aquisição das informações para uma aplicação sensível ao contexto: o usuário utiliza um contexto computacional, nesse caso, o smartphone. Ao solicitar ajuda ao sistema, o usuário terá diversas informações já armazenados num banco de dados. Essas informações como, tempo e localização, chegam ao sistema, através de sensores por via satélite, GPS, redes *Wifi* e aplicativos diversos.

As captar essas informações através destes sensores, elas passarão por mediadores que tratarão de gerenciar essas informações. Esse processo de gerenciamento, é o middleware que pode ser um API, Saas, Web Services, dentre

outros.

Na situação contextual sugerida no exemplo em que se trata de um usuário com restrições alimentares, o sistema oferecerá informação baseadas nas dimensões: *who* (perfil), o modelo oferecerá informações como: nome, idade, preferências e restrições. No *where* (localização), as informações serão com base no ponto de localização, proximidade, coordenadas e rotas. No *what* (atividade e status atual), estudando, viajando, trabalhando, lazer. No *when* (momento atual), clima, trânsito, estação do ano, fuso horário, data e hora. No *Why* (motivação), alimentar-se, saúde, qualidade, diferencial, variedade e estrutura. *How* (define a coleta dos dados), já informado anteriormente.

Para aquisição de informação do contexto do perfil o desenvolvedor, precisa utilizar *API login Facebook*, onde terá informações pessoais do usuário como nome, e-mail, status atual. Para ter outras informações como preferencias, profissão, data de nascimento deve disponibilizar um formulário que o usuário informe os dados, e ficar armazenado.

Para aquisição de informações de localização é feito com auxílio da *API Location* desenvolvida pela Google, que indica a localização do aparelho e captura as alterações de local.

O contexto tempo no modelo apresentado implicará nas informações que são utilizadas em conjunto com outras partes de um contexto, informando data e hora ou o intervalo em que uma determinada informação contextual foi percebida ou é relevante, como por exemplo, clima e estação do ano. (GOMINHO; TEDESCO; BELIAN, 2014).

4.2.4 Resultados e discussão

A discussão desse trabalho se deu com base em dois contextos: o primeiro uma análise dos quatro correlatos escolhidos como fonte de inspiração para o desenvolvimento de um modelo de um SSC. Para isso, buscou-se relacionar os seguintes aspectos: necessidade do desenvolvimento (para quê?), os elementos, representação, gerenciamento e uso do contexto pelo usuário final.

Em relação a necessidade o primeiro correlato buscou desenvolver um SSC que se informa ao usuário final o horário de chegada e a localização atual do ônibus. Como elementos utilizou a localização, pontos de paradas e intensidade de congestionamento. Para aquisição das informações utilizou o GPS e o API Google maps. O tipo de representação foram os grafos contextual. Para gerenciar as informações utilizou um sistema de *Back end* e *Front end*.

O trabalho apresentou vantagens e desvantagens. Segundo o autor, levando-se em conta uma avaliação comparativa com outras soluções, foi possível observar que a utilização de contexto desenvolvida, contribuiu para o alcance de resultados mais precisos. Concluiu-se que o CSS desenvolvido mostrou-se eficiente tanto quando foi exposto a diferentes contextos quanto quando comparado a possíveis soluções estáticas que poderiam ser dadas como alternativa para o mesmo desafio. Ao mesmo tempo, foi possível identificar algumas restrições como a dependência das atualizações do API *Google maps*, o qual pode fornecer erros decorrentes dessa desatualização. A ausência de simuladores de GPS, o que dificultou a informação de navegação em trânsito real na cidade. Essa ação poderia colaborar com a aquisição de informações dos cenários reais sem a necessidade de execução de testes de rua. (CALDAS, 2010).

O modelo apresentado pelo autor apresentou-se eficaz para o contexto específico do estudo, porém, apresentou fragilidades na aquisição de informações, por necessitar de contínuas atualizações de rota e arquitetura do tráfego. O desenvolvedor de certa forma fica refém do GPS ou do próprio desenvolvedor.

O segundo correlato apresentou como necessidade, desenvolver um método que visava reduzir o consumo de energia baseado em contexto para smartphones através da seleção de sensores. Utilizou como elementos: Usuário, dispositivo, mobilidade, sociabilidade, posição, interatividade e localização. Como fontes de contexto, usou: GPS, *wifi*, Redes, acelerômetro e *bluetooth*. Como sistema de representação utilizou regras de associação. Desenvolveu um experimento envolvendo equipamentos e usuários, com intuito de testar a durabilidade das baterias com método sensível a contexto para reduzir o consumo de energia em smartphones.

Bezerra (2015), concluiu com seu experimento que a qualidade do sistema está diretamente relacionada com a qualidade dos dados coletados. Alegou

que reduzir o consumo energético da aquisição e processamento de contexto está diretamente ligado ao gerenciamento dos sensores. Desse modo, deliberou 3 métodos para gerenciar o consumo de energia dos sensores e a qualidade das informações coletadas: (i) seleção de sensores, (ii) adaptação da taxa de amostragem e (iii) predição do comportamento do usuário.

Para reconhecer o contexto do usuário, Bezerra (2015) definiu-se um conjunto de regras de associação dentre as quais pudessem estabelecer relações entre os diferentes elementos contextuais. Para isso foi preciso analisar 5 características de diversos usuários, baseadas na definição de contexto 5W+1H: mobilidade, sociabilidade, interatividade, localização e posição. Com esse método, foi possível ter um aumento de tempo de bateria num valor de 22,82% a mais em termo de durabilidade.

Como desvantagem verificou-se algumas fragilidades no experimento como: amostra pequena de usuários na participação do experimento, num experimento de maior amostra, a escolha por regras de associação torna-se mais difíceis. Utilização de mais sensores, principalmente os mais modernos, dentre outras fragilidades.

De modo geral, verificou-se que ao desenvolver um SSC voltado para o gerenciamento que visa a redução do consumo de energia de bateria de um smartphone, é relevante e segundo o experimento, apresentou um resultado positivo. Verificou-se que mesmo diante das fragilidades apresentadas no estudo, é possível alcançar um resultado ainda mais otimista se considerar equipamentos e sensores mais modernos.

O terceiro correlato apresentou uma abordagem para a modelagem de características no desenvolvimento de LPSSC, o qual foi possível representar as informações de contexto relevantes para o domínio desde as fases iniciais de desenvolvimento, facilitando o entendimento dos envolvidos no processo.

Utilizou como elementos: rede, dispositivo móvel e usuário. Como fonte de contexto utilizou o *AdaptiveRME* para aquisição das informações. O sistema de representação utilizada por Fernandes (2009) foi a elaboração de uma modelagem de característica. Apresentou a abordagem UBIFEX com dois componentes: o *UbiFEX-Notation* e o *UbiFEX-Simulation*.

O presente estudo teve como objetivo propor uma abordagem para

modelagem de características de linha de produtos de software sensíveis ao contexto. Apresentou como principais contribuições: definição de uma abordagem denominada *UbiFEX* para a modelagem de características de LPSSC; avaliação preliminar do mecanismo de verificação *UBIFEX - Simulation*, por meio de estudo realizado com alunos de pós-graduação, com intuito de utilizar os resultados para melhoria da abordagem servindo como base para o planejamento de uma avaliação mais completa; desenvolvimento de um protótipo que implementa a abordagem proposta no contexto do ambiente *Odyssey*. (FERNANDES, 2009).

Como fragilidade, apresenta-se os seguintes pontos: *UbiFEX-Simulation* tem como objetivo antecipar a identificação de possíveis inconsistências em tempo de execução, porém, essa aplicação não garante que o produto não irá falhar, pois cenários não previstos podem ocorrer durante a execução dos produtos. A aplicação de *UbiFEX-Simulation* é restrita a modelo de características desenvolvidos utilizando *UbiFEX-Notation*.

Desse modo, o estudo de Fernandes (2009), apresentou as informações de contexto relevantes para o domínio e para a representação do impacto dessas informações na variabilidade dos produtos em tempo de execução. Além disso, foi proposto um mecanismo para verificação da consistência das configurações dos produtos em diferentes cenários de execução com base nas restrições definidas em modelos de características. A abordagem *UBIFEX* para o estudo realizado apresentou resultados relevantes em função do objetivo do estudo em pauta. Porém, apresentou fragilidades pois a abordagem apresenta limitações quanto a aplicação e execução dessas informações, no caso de mudança de execução e contexto.

O quarto relato, de Machado (2014), propôs uma abordagem sensível ao contexto e de interface adaptativa, a ser utilizada em um dispositivo móvel, para recomendar objetos de aprendizagem próximos e interessantes aos usuários, sendo este um membro da comunidade acadêmica em um campus universitário. Como elementos utilizou-se: usuário, localização, dispositivo e objetos de aprendizagem. Como fonte de contexto: API *Google maps* e motor de inferência. Como sistema de representação utilizou ontologia e metadados.

A arquitetura do gerenciamento dividiu-se em três componentes: coletor de contexto, o recomendador baseado em conhecimento no lado do servidor e o controlador de interface.

Como resultado o autor delimitou que o motor de inferência Pellet foi utilizado para a execução das inferências lógicas sobre as regras construídas. Os resultados obtidos com o estudo de caso mostraram que a abordagem é eficaz em realizar as recomendações propostas. A dificuldade apresentada neste trabalho, foi a atualização de dados do usuário, o que pode influenciar no desempenho da abordagem.

O estudo e análise dos correlatos apresentados, permitiram ao pesquisador um olhar reflexivo perante a definição de quais elementos, técnicas, middleware, modo de aquisição, processamento e disseminação das informações, poderiam contribuir da melhor forma para seu desenvolvimento.

Verificou-se nos quatro correlatos, que nos diferentes casos apresentados, todos apresentaram resultados positivos, porém com fragilidades.

Desse modo, elaborou-se o modelo de características que auxiliasse o desenvolvimento de aplicações sensíveis ao contexto que busca atender as necessidades das pessoas, facilitando a identificação das características relevantes dos elementos em uma aplicação. Para isso foi necessário definir quais informações de contexto computacional, físico, do usuário e temporal, seriam necessários para a elaboração do modelo.

O presente trabalho apresentou um modelo de características para representar informações relevantes em um domínio de aplicação sensível a mudanças em tempo de execução, desenvolvido a partir do embasamento teórico obtido por meio da pesquisa bibliográfica e através de quatro correlatos. O modelo elaborado, teve como foco, o SSC onde teve como usuário um sujeito com restrições alimentares em busca de um local para suprir sua necessidade alimentar emergencial no referido momento. Para isso, elaborou-se uma arquitetura específica que contextualizasse o SSC para descrever o trajeto das informações e de que maneira elas seriam processadas e disseminadas.

Com este trabalho foi possível compreender que cada contexto tem sua especificidade, para isso, é necessário compreender as necessidades dos usuários e as ferramentas de aquisição das informações para o desenvolvimento do sistema sensível ao contexto.

A usabilidade de contexto em uma aplicação mostra-se fundamental para disponibilizar ao usuário sistemas adaptáveis e serviços relevantes as necessidades

do usuário no tempo desejado, a compreensão de como o contexto pode ser usado ajuda a determinar quais comportamentos uma aplicação deve tomar de formar sensível ao contexto.

6 CONCLUSÃO

Diante da grande evolução computacional, o aumento de informações disponíveis e a integração de tecnologias têm impulsionado a evolução dos sistemas de software, por este motivo sistemas adaptativos tornam-se a cada dia mais essenciais. Um dos tipos de sistemas adaptativos mais comuns são os sistemas sensíveis ao contexto, devido ao acesso aos dispositivos móveis, a facilidade em conectar-se à internet e o aumento dos mais variados tipos de sensores que possibilitam e fornecem a informação de diversas formas. Quem procura uma informação, seja ela qual for, busca um sistema que seja adaptável, flexível e fácil de usar.

A modelagem de características para representar um domínio de aplicação sensível a mudanças em tempo de execução representa um novo paradigma para unificar esforços no desenvolvimento de sistemas.

Um sistema sensível ao contexto tem a capacidade de fornecer serviços adaptáveis as necessidades e intenções do usuário. Permite adaptar resposta de acordo com as preferências do usuário e interação com o sistema no momento em que a consulta é realizada no dispositivo entre outras características.

A abordagem do texto procurou descrever metodologias sem a preocupação de classificar ou avaliar o método utilizado, ou se existe uma metodologia melhor que outra.

O referido trabalho apresentou como objetivo geral, desenvolver um modelo de características para representar informações de contexto relevantes em um domínio de aplicação sensível às mudanças em tempo de execução. Para isso foi necessário realizar uma pesquisa bibliográfica que serviu como fonte referencial dos conceitos essenciais para definir os componentes que seriam utilizados no modelo que se pretendia desenvolver.

Procurou-se apresentar quatro correlatos de trabalhos acadêmicos desenvolvidos, sendo um de graduação e três de pós-graduação stricto sensu (mestrado), o qual os pesquisadores apresentaram a aplicação de um SSC em situações diferentes. Verificou-se que nos quatro trabalhos, todos apresentaram algum tipo de elemento, os tipos de sensores de aquisição de informação, a forma de representação e gerenciamento dos dados.

A disseminação do projeto de execução dos quatro correlatos apresentou resultados positivos, dentro do esperado pelos pesquisadores, no entanto, ambos apontaram algumas dificuldades enfrentadas durante o processo de experimentação. No entanto, considerou-se natural pois todo experimento solicita novos desafios, o que acaba gerando novas pesquisas.

Partindo para a definição dos objetivos específicos dessa pesquisa, com a contribuição de vasta pesquisa em diversos periódicos, TCCS, Dissertações, Teses, Livros e nos próprios correlatos, foi possível atingir os objetivos específicos, dentre os quais delimitavam a construção de um modelo a partir da identificação, análise, descrição e definição, o que permitiu a construção de um modelo de características de um SSC.

O problema de pesquisa partiu do seguinte questionamento: de que maneira é possível construir um modelo que atenda uma necessidade e represente um Sistema Sensível ao Contexto?

Com o desenvolvimento da pesquisa foi possível compreender que não há como desenvolver qualquer sistema sensível ao contexto, sem que haja sensores de captação de informações sendo que estas, devem ter alguma utilidade específica para um determinado contexto com seus referidos elementos. Estas informações precisam estar organizadas, ou precisam de um caminho para que possam chegar ao destino, para isso é preciso de um mediador, no caso do modelo usado nesse trabalho, utilizou-se um middleware. O *Middleware* contribui significativamente com o gerenciamento dos dados. As informações são levadas por ele e ficam armazenadas num banco de dados, o qual junto com um tipo de representação, no caso, um metadados, as informações são levadas ao destino de forma organizada.

Desse modo, este trabalho de pesquisa contribuiu com um modelo de características para representar as informações para aperfeiçoar o desenvolvimento de sistemas sensíveis ao contexto, sendo suficiente para funcionar adequadamente em diversos tipos de dispositivos formas de representação. Hoje, para desenvolver uma aplicação sensível de contexto deve programar todas as etapas do processo, desde o reconhecimento do contexto a parte de monitoramento e adaptação. A utilização de middleware que provêm os serviços de infraestrutura pode facilitar esse processo e permitir que o programador se preocupe apenas com o desenvolvimento da aplicação.

Concluiu-se que, recomenda-se ao utilizar modelagem de características para representar um domínio de aplicação sensível a mudanças em tempo de execução. No desenvolvimento de uma nova aplicação, existe uma alta expectativa de reusabilidade (adaptar uma classe existente em um sistema a outro sistema), implicando em redução de tempo e custo de projeto, sem queda na qualidade dos trabalhos.

Para se obter um bom proveito da modelagem, considera-se alguns aspectos que devem ser observados antes do desenvolvimento completo das aplicações, como a definição das metodologias de modelagem, análise e projeto; os padrões para a criação de desenvolvedores de sistemas possam enriquecer a usabilidade de sua aplicação.

Para aplicações comerciais onde uma base de dados é indispensável, um mapeamento forçado dos modelos é importante. Isso significa uma perda da semântica para a implementação das operações e métodos. A evolução desse paradigma busca a padronização de alguns conceitos e uma maior produtividade em uma área crítica como o desenvolvimento de software na busca de acelerar o surgimento de novas ferramentas. O que precisa ser melhor entendido, é que a utilização de modelagem de características para representar um domínio de aplicação sensível a mudanças em tempo de execução representa requer que se abandone práticas tradicionais e se experimente novos modelos como o desenvolvimento paralelo, onde múltiplas partes de um mesmo problema são resolvidas simultaneamente.

Durante o desenvolvimento do modelo, sentimos a necessidade de explorarmos com mais profundidade estudos, voltados ao gerenciamento e processamento de Middleware, banco de dados e metadados, visto que no modelo proposto, sugerimos alguns tipos, no entanto, não foi objetivo do trabalho desenvolver uma arquitetura específica para cada um deles.

REFERÊNCIAS

ABOWD, D. G.; ATKESON, C.; AUST, D.; LONG, S. Cyberguide: um guia turístico móvel compatível com o contexto. **Redes sem fio**, v. 3, n. 5, 1997.

ABOWD, G. D.; DEY, A. K.; SALBER, D. The Context Toolkit: aiding the development of context-enabled applications. In: **Proceedings of CHI'99, College of Computing Pittsburg**. ACM Press. PA, 1999.

ADAMS, N.; SCHILIT, B. N.; WANT, R. Context-Aware Computing Application. To Appear: **IEEE Workshop on Mobile Computing Systems and Applications**, December 8-9 1994. Disponível em: <<https://dl.acm.org/citation.cfm?id=1440041>> Acesso em: 12 set. 2018.

ALMEIDA, R. A. P. **Blueyou**: uma plataforma de comunicação ciente de contexto baseada em serviços para computação móvel. Dissertação (Mestre em Ciência da Computação) Universidade Federal de São Carlos, São Carlos, 2011.

ALVES JUNIOR, M. H.; VALE, S.; SILVA, F. J. S. VALE, S. B.; SILVA, F. J. S. E.COUTINHO, L. R.. **Confiança em Sistemas Sensíveis ao Contexto aplicados ao Domínio da Saúde**. Exame de qualificação (Mestrando em Ciência da Computação) - Universidade Federal do Maranhão, 2014. Disponível em: <http://lsd.ufma.br/~marcelo/docs/Qualificacao_Marcelo.pdf> Acesso em: 12 out. 2018.

AMARAL, J. J. F. **Como fazer uma pesquisa bibliográfica**. 25f. 2007. Universidade Federal do Ceará. Disponível em: <<http://200.17.137.109:8081/xiscanoe/courses-1/mentoring/tutoring/Como%20fazer%20pesquisa%20bibliografica.pdf>> Acesso em: 12 set. 2018.

ANJOS, A. G. R. **Aplicações móveis cientes de contexto**: um gerenciador de grupos de usuários para o *mobile collaboration architecture*. 65f. Monografia (Graduação em Ciência da Computação) Universidade Federal de Pernambuco, Recife, 2006.

ARAÚJO, B. G. **Modelo arquitetural de comunicação para monitoramento de pacientes baseado em *middleware*, computação móvel e ubíqua**. 72f. Dissertação (Mestrado Engenharia Elétrica). Universidade Federal do Rio Grande do Norte, Natal, 2010.

AUGUSTIN I.; LIBRELOTTO, G. R.; MARAN, V.; SACCOL, D. B. Uma definição ontológica de elementos de contexto relevantes na adaptação de documentos em ambientes hospitalares persavisos. **Revista Brasileira de Computação Aplicada**, Passo Fundo, v.5, n.1, abril, 2013.

AVANCINI, V. P. **Um sistema de recomendação sensível ao contexto para atividade de lazer**. 78f. Trabalho de Conclusão de curso (Bacharel em Sistemas de Informação) Universidade Federal de Santa Catarina, Florianópolis, 2016.

AZEVEDO, B. R. **Inferências do esforço cognitivo despendido em atividades humanas a partir de um sistema sensível ao contexto.** 68f. Dissertação (Mestrado em Informática) Universidade Federal de Santa Maria, Centro de Tecnologia, Santa Maria, 2015.

BARBOSA, A. C. P. **Middleware para integração de dados heterogêneos baseado em composição de frameworks.** 167f. Tese (Doutorado em Informática: Ciência da Computação) Universidade Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2001.

BARBOSA, S. D.J; SILVA, B. S. **Interação humano-computador.** Rio de Janeiro: Elsevier, 2010.

BEZERRA, P. T. L. Seleção de Sensores Sensível ao Contexto para redução do consumo de energia em *Smartphones*. 183f. Dissertação (Mestrado em Ciência da Computação) Universidade Federal de Pernambuco. CIN, Recife. 2015.

BLOIS, A. **Uma abordagem de projeto arquitetural baseado em componentes no Contexto de Engenharia de Domínio.** 219f. Tese (Ciências em Engenharia de Sistema de Computação). Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2006.

BURGARELI, L. A. **Gerenciamento de variabilidade de linhas de produtos de software com utilização de objetos adaptáveis e reflexão.** 247f. Tese (Doutorado Engenharia Elétrica) Escola Politécnica da universidade de São Paulo. 2009.

CALDAS, L. R. **Desenvolvimento de uma solução sensível ao contexto como suporte a um sistema de transporte público inteligente.** 89 f. Monografia (Graduação em Ciências da Computação) Universidade Federal da Bahia, Bahia. 2010.

CHOI, J. Context-Driven Requirments Analysis. **O. Gervasi and M. Gavrilova (Eds.): ICCSA 2007, LNCS 4707, Part III, pp. 739–748, 2007.**

COSTA, P. A. **Uma ferramenta para análises automáticas de modelos de características de linhas de produtos de software sensível ao contexto.** 121f. Dissertação (Mestrado em Ciência da Computação) Universidade Federal do Ceará. Fortaleza, 2012.

DIAS, K. L. **Negociação e acordo de acesso à informação de contexto.** 14f. TCC (Conclusão disciplina de Mestrado em Informática) Pontifícia Universidade Católica Do Rio De Janeiro, Rio de Janeiro. 2005.

DEY, A. K. Understanding and using context. **Rev. Personal and ubiquitous computing.** V. 5 Ed. 1, Fev. 2001.

_____. **Providing architectural support for building context-aware applications.** 34f. Doctoral Dissertation. Georgia Institute of Technology, 2000.

DIAS, M. M. K.; PIRES, D. **Usos e usuários da informação**. São Carlos: EdUFSCar, 2004.

FERREIRA, C. **Maestro**: um *middleware* para suporte a aplicações distribuídas baseadas em componentes de *software*. 112f. Dissertação (Mestre Engenharia) Escola Politécnica da Universidade de São Paulo, São Paulo, 2001.

FERNANDES, P. C. C. **UbiFEX**: Uma Abordagem para Modelagem de Características de Linha de Produtos de *Software* Sensíveis ao Contexto. 139f. Dissertação (mestrado Engenharia de Sistemas e Computação). Rio de Janeiro: UFRJ/COPPE, XIII, 2009.

FIGUEIREDO, C. M. S.; NAKAMURA, E. Computação Móvel: Novas Oportunidades e Novos Desafios. **Rev. T&C Amazônia**, Ano 1, nº 2, Jun de 2003. Disponível em: <https://www.researchgate.net/publication/268435975_Computacao_Movel_Novas_Oportunidades_e_Novos_Desafios_COMPUTACAO_MOVEL_NOVAS_OPORTUNIDADES_E_NOVOS_DESAFIOS> Acesso em: 10 jul. 2018.

FONTELES, A. S. **Um framework para aquisição adaptativa e fracamente acoplada de informação contextual para dispositivos móveis**. 97f. Dissertação (Mestrado Ciências da Computação). Universidade Federal do Ceará, Fortaleza, 2013.

FRANÇA, C. T. P. L. **Um middleware para o gerenciamento de Clusters de SGBDs Heterogêneos**. 115 f. 2010. Dissertação [Mestrado Profissional em Computação Aplicada]. Universidade Estadual do Ceará e Instituto Federal de Educação, Ciência e Tecnologia do Ceará. Fortaleza - Ceará - CE, 2010.

GALEGALE, G. P.; SIQUEIRA, E.; SILVA, C. B. H.; SOUZA, C. A. Internet das Coisas Aplicada a negócios: um estudo Bibliométrico. **Revista de Gestão da Tecnologia e Sistemas de Informação**. V. 13, N. 3, Set/Dez., 2016 pp. 423-438.

GU, T.; PUNG, H.; ZHANG, D. A Service-Oriented Middleware for Building Context-Aware Services. **Journal of Network and Computer Applications**, v. 28, 2005.

HAEZEL, F. S. **Simulador para desenvolvimento de software voltado a computação sensível ao contexto**. 56f. Trabalho de Conclusão (Bacharel em Ciência da Computação) UNIJUÍ, Santa Rosa. 2012.

HOENTSCH, S.; MENEZES, L.; RIBEIRO A. Uma proposta de rede social móvel ciente de contexto de localização. **Revista de Sistemas e Computação**, Salvador, v. 2, n. 2, p. 158-167, jul./dez. 2012.

JANDI JUNIOR, P. Computação, ubiquidade e transparência. In: **UNIANCHIETA**. Ubiquidade. São Paulo, 2011.

KON, F.; COSTA, F.; BLAIR, G.; CAMPBELL, R. H. The case for reflective middleware. In: **Communications of the ACM**, V.45, n. 6, jun. 2002.

KUDO, T. N. **Computação ciente de contexto aplicada ao monitoramento de condições críticas em ambientes físicos**. 118f. Dissertação (Mestrado em Ciência da Computação) Universidade Federal de São Carlos, São Carlos, 2004.

LAMAS, A. R. **Uma arquitetura para o desenvolvimento de sistemas de informação geográfica móveis sensíveis ao contexto**. 83f. Dissertação (Ciência da Computação) Universidade Federal de Viçosa, Minas Gerais. 2009.

LEE, J.; MUTHIG, D. Feature-Oriented Variability Management in Product Line Engineering. **Communications of the ACM**, v. 49, n. 12, Dec. 2006.

LIMA, R. M.; GUERRA, L. T. B.; FIORIN, A. Educação Ubíqua: um modelo de adaptação para o Moodle. **Rev. Eletrônica Argentina-Brasil Tecnol. da Informação e da Comun.** 2015. Disponível em: <<https://revistas.setrem.com.br/index.php/reabtic/article/download/80/39>> Acesso em: 18 set. 2018.

LINTHICUN, D. **Database-oriented middleware**. Junho de 2001. Disponível em: <<http://searchsoa.techtarget.com/answer/Database-oriented-middleware>>. Acesso em 18 out. 2018.

LOPES, J. L. **Sensibilidade ao contexto na computação pervasiva: avaliando o uso de ontologias**. 70f. Trabalho Individual (Pós-graduação em Informática) Universidade Católica de Pelotas – Escola de Informática, Pelotas, 2006.

LOPES, F. **Uma Plataforma de Integração de *Middleware* para Computação Ubíqua**. 171f. Tese (Doutorado Ciências da Computação), Universidade Federal do Rio Grande do Norte, Natal/RN, 2011.

LOPES, J. L.; PERNAS, A. M.; AFONSO, F.; LIBRELOTTO, G.; AUGUSTIN, I.; PALAZZO, L. A.; YAMIN, A. **Uma abordagem baseada em ontologias para sensibilidade ao contexto na computação Pervasiva**. Publicado em março de 2014. Disponível: <https://www.researchgate.net/publication/242401730_Uma_Abordagem_Baseada_em_Ontologias_para_Sensibilidade_ao_Contexto_na_Comp_utacao_Pervasiva> Acesso em: 24 set. 2018.

LONDEIRO, I. Z. Técnicas de Computação Sensível ao contexto. 46 f. Trabalho de Conclusão de Curso (Bacharel em Ciência da Computação). Universidade Federal do Pampa, Alegrete. 2012.

LOUREIRO, A. A.; RICARDO, A. R. O.; THAIS, R. B. S. WALDIR, R. P. J. **Computação Ubíqua ciente de contexto: desafios e tendências**. Cap. 3. 27º Simpósio Brasileiro de Redes de Computadores e sistemas distribuídos. Livro texto dos Minicursos, 2009.

MACHADO, G. M. **Recomendação adaptativa e Sensível ao Contexto de Recursos para Usuários Móveis em um Campus Universitário**. 69f. Dissertação (Mestrado) Universidade Federal do Rio Grande do Sul, Porto Alegre. 2014.

MACIEL, R. S. P.; ASSIS, S. R. *Middleware*: uma solução para o desenvolvimento de aplicações distribuídas. **Científico**. Ano IV, v. I, Salvador, jan/jun 2004.

MANCINI, M. **Internet das Coisas**: história, conceitos, aplicações e desafios. Junho de 2018. Disponível em: <https://www.researchgate.net/publication/326065859_Internet_das_Coisas_Historia_Conceitos_Aplicacoes_e_Desafios> Acesso em: 12 set. 2018.

MANNINO, M. V. **Projeto, desenvolvimento de aplicações e administração de banco de dados**. 3 ed. Porto Alegre: AMGH Editora Ltda, 2008.

MARINHO, F. G. **Precise**: um processo de verificação Formal para modelos de características de aplicações móveis sensíveis ao contexto. 182f. Tese (Doutorado em Ciências da Computação), Ciências da Computação da UFC, Fortaleza, 2012.

MASSEN, T.V.D.; LICHTER, H. Deficiencies in Feature Models. In: Workshop on Software Variability Management for Product Derivation - **Towards Tool Support, Boston, MA, USA, August, 2004**.

_____. Modeling Variability by UML Use Case Diagrams. In: **Proceedings of the International Workshop on Requirements Engineering for Product Lines (REPL'02), Essen, Germany, September, 2002**.

MATIAS, B. **Computação Sensível ao Contexto**. Paper. Escola Superior de Tecnologia de Castelo Branco. Publicado em 2013/2014. Disponível em: <https://www.artigos.com/index.php?option=com_mtree&task=att_download&link_id=14431&cf_id=24> Acesso em: 12 set. 2018.

MIELKE, I. T. **Uma abordagem baseada em modelos para especificação e detecção de situações em sistemas sensíveis ao contexto**. 154 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal do Espírito Santo, Vitória, 2013.

MILLER, T.; MCBURNEY, P. A formal semantics for Gaia liveness rules and expressions. **Int. J. Agent-Oriented Softw. Eng.**, Inderscience Publishers, Geneva, Switzerland, v.1, 2007.

NAZÁRIO, D. C.; DANTAS, M. A. R.; TODESCO, J. L. Modelo de conhecimento de qualidade de contexto para ambientes Ubíquos baseado em ontologia. **Int. J. Knowl. Eng. Manage.**, Florianópolis, v.4, n.9, p. 51-86, jul/out. 2015.

NUNES, V.; MAGDALENO, A.; WERNER, C. M. Modelagem de contexto sobre o domínio de processos de desenvolvimento de *software*. **Relatório Técnico – Programa de Engenharia de Sistemas e Computação**, UFRJ, Rio de Janeiro. 2010.

OLIVEIRA JUNIOR, G. M. O. **UbiMid**: um *middleware* de integração e sensível ao contexto voltado para aplicações e sistemas inteligentes de transporte. 80 f. Dissertação (mestrado) - UFPE, Centro de Informática, Programa de Pós-graduação

em Ciência da Computação, Recife, 2014.

OLIVEIRA, R. **Formalização e verificação de consistência na representação de variabilidades**. 146f. Dissertação (Ciências em Engenharia de Sistemas de Computação), Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2006.

PATRICIO, R. F. CEManTIKA CASE: uma ferramenta de apoio ao desenvolvimento de sistemas sensíveis ao contexto. Dissertação (Mestrado) – Universidade Federal de Pernambuco. Ciência da Computação, Recife, 2010.

PEREIRA, H. M. O.; ROCHA, R. C. A. **Localização, mapas e contexto: um Framework** para desenvolvimento de aplicações baseadas em Mapas Contextuais. Disponível em: < <http://www.inf.ufg.br/~ricardo/pub/heitor-sbcup.2010.pdf>> Acesso em: 10 out. 2018.

PEREIRA FILHO, J. G.; PESSOA, R. M.; CALVI, C. Z.; OLIVEIRA, N. Q.; BARBOSA, A. C. P.; FARIAS, C. R. G. **Infraware: um Middleware** de Suporte a Aplicações Móveis Sensíveis ao Contexto. Publicado em 2006. Disponível em: < http://www.lbd.dcc.ufmg.br/colecoes/sbrc/2006/sc2_4.pdf> Acesso em: 12 set. 2018.

PEREIRA, R. R.; MAIA, M. **Desenvolvimento de software orientado a características e dirigido por modelos**. Publicado em: 2008. Disponível: <www.facom.ufu.br/posgrad/2008/artigos/Rodrigreis.pdf>. Acesso em: 22 set.,2018.

PESSOA, R. M. **Infraware: um middleware** de suporte a aplicações sensíveis ao contexto. 140 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal do Espírito Santo, Vitória, 2006.

PINHEIRO, D. N. **MHNCS: um middleware** para o desenvolvimento de aplicações móveis cientes de contexto com requisitos de QoC. 117 f. Dissertação (mestrado Engenharia de Eletricidade). Universidade Federal do Maranhão. São Luiz, 2014.

QUEIROZ FILHO, C. A. B. **Um mecanismo de tratamento de exceções sensível ao contexto para sistemas ubíquos orientados a tarefas**. 96 f. Dissertação (Mestrado em ciência da computação) - Universidade Federal do Ceará, Fortaleza-CE, 2012.

RANGANATHAN, A.; AL-MUHTADI, J. MiddleWhere: a middleware for location awareness in ubiquitous computing applications. In: **ACM/IFIP/USENIX International Conference on Middleware**, New York, NY, USA. Proceedings. Springer-Verlag New York: Inc., 2004.

RIBEIRO, F. R.; VICENTE, F.; RISCADO, L.; METROLHO, J. Framework para caracterização e disponibilização do contexto em espaços públicos. In: **Conferência Ibérica de Sistemas e Tecnologias de Informação**, 8, Lisboa. 2013. Disponível em: <www.eeexplore.ieee.org/stamp/stamp.jsp?tp>. Acesso: 24 out. 2018.

ROCHA, C. C. **Uma arquitetura para autenticação sensível ao contexto baseada em definições comportamentais**. 109f. Dissertação (Mestre em Ciência da

Computação) Universidade Federal de Santa Catarina, Florianópolis, 2010.

ROCHA, L. **Adaptive RME e Aspect Compose: Middleware Adaptativo e um processo de composição orientado a aspectos para o desenvolvimento de software móvel e ubíquo.** 103f. Dissertação (Ciência da Computação) Universidade Federal do Ceará, Fortaleza, 2007.

RODRIGUES, V. **Percepção de Contexto.** Disponível: <www.inf.ufg.br/vagner/mobilecomputing/docs.pdf>. Acesso em: 28 Ago. 2017.

SÁ, M. P. Conhecendo a Computação Móvel Sensível ao Contexto. In: **Anais da XII Escola Regional de Computação dos Estados da Bahia, Alagoas e Sergipe**, em abril de 2012. Disponível em: <http://www.erbase2012.univasf.edu.br/wpcontent/uploads/2012/AnaisERBASE2012_Minicursos_Labs.pdf> Acesso em: 12 set. 2018.

SANTOS, I. S. **Um ambiente para geração de cenários de testes para linhas de produtos de software sensíveis ao contexto.** 135 f. Dissertação (Mestrado em ciência da computação) - Universidade Federal do Ceará, Fortaleza - CE, 2013.

SANTOS, L. F. C. **Modelo de contexto para recomendação de rotas baseada em interesse.** 67f. Trabalho de conclusão de curso (Graduação Ciência da Computação), Universidade Tecnológica Federal do Paraná, Campo Mourão, 2016.

SANTOS, V. V. **Gerenciamento de contexto em sistemas colaborativos.** 117f. Monografia de Qualificação e Proposta de Tese (Doutor em Ciência da Computação) Centro de Informática da Universidade Federal de Pernambuco, Recife, 2006.

SILVA, F. P. **Abordagem baseada em metamodelos para a representação e modelagem de características em linhas de produto de software dinâmicas.** 114f. Dissertação (Mestrado) Universidade Federal de Goiás, Instituto de Informática. Goiânia, 2016.

SILVA, V. G. **Descoberta dinâmica, sensível ao contexto, de serviços web.** 107f. Dissertação (Mestrado em Ciências da Computação), Universidade Federal de São Carlos, São Paulo, 2014.

SILVA, D. **Computação ubíqua: a informática no cotidiano das pessoas.** Disponível em: <<https://www.estudopratico.com.br/computacao-ubiqua-a-informatica-no-cotidiano-das-pessoas/>> Acesso em: 18 set. 2018.

SILVA, E. G. **Desenvolvimento de um sistema para adaptação de contexto em automação residencial.** 88f, TCC (Graduação em Tecnologias de Informação e Comunicação). Universidade Federal de Santa Catarina, Araranguá, 2013.

SILVA, E.; BOTELHO, L.; SANTOS, I.; SANCHEZ, G. Computação Ubíqua: definição e exemplos. **Rev. de Empreendedorismo, Inovação e Tecnologia**, 2(1): 23-32, 2015. Disponível em: <<file:///home/chronos/u-57df9975a5ddb0438a445b1bea726dc7e9b31754/Downloads/926-4624-1-PB.pdf>> Acesso em: 10 jul. 2018.

SOARES, P.R. S. **Desenvolvimento de Propaganda Interativa e Sensível ao contexto para TV Digital**. 64f. Trabalho de graduação (Bacharel em Ciência da Computação) Centro de Informática da Universidade Federal de Pernambuco, Recife, 2010.

SOARES, R. H. **Gerenciamento Dinâmico de Modelos de Contexto: Estudo de Caso Baseado em CEP**. 72f. Dissertação (Mestrado Ciências da Computação) Instituto de Informática da Universidade Federal de Goiás, Goiânia, 2012.

SOUZA, J. P. **Uma arquitetura para computação sensível ao contexto aplicada a processo de software**. 104f. Dissertação (Mestre em Computação Aplicada), Universidade tecnológica Federal do Paraná, Curitiba, 2012.

STEFANELLO, D. **Um framework baseado em MDE e Weaving para suporte ao desenvolvimento de sistemas de software sensíveis ao contexto**. 107f. Dissertação (Mestre em engenharia de Eletricidade), Universidade Federal do Maranhão, São Luiz, 2017.

VAN DER LINDEN, F.; SCHMID, K.; ROMMES, E. **Software Product Lines in Action: The Best Industrial Practice in Product Line Engineering**. New York: Springer Verlag, 2007.

VIEIRA, V.; TEDESCO, P.; SALGADO, A. C. Modelos e processos para desenvolvimento de Sistema Sensíveis ao Contexto. Publicado em 2009. Disponível em: < http://homes.dcc.ufba.br/~vaninha/context/2009_TextoJAI_Final.pdf> Acesso em: 12 set. 2018.

VIEIRA, V.; SOUZA, D.; SALGADO, A. C.; TEDESCO, P. Uso e Representação de Contexto em Sistemas Computacionais. In: TEIXEIRA, C. A. C.; FARIAS, C. R. G.; LEITE, J. C.; PRATES, R. O. (Org.). **Tópicos em Sistemas Interativos e Colaborativos**. São Carlos: UFSCAR. 2006.

KOWALTOWSKI, T. (Org.). **Jornadas de atualização em Informática**. Porto Alegre: SBC, 2009.

WEISER, M. The computer for the 21 st Century. **Rev. Scientific American**, set. 1991. Disponível em: <<https://www.lri.fr/~mbl/Stanford/CS477/papers/Weiser-SciAm.pdf>> Acesso em: 18 set. 2018.

WINCK, D. V. **Extensão para UML destinada à modelagem de variabilidade transversa em componentes através da orientação a aspectos**. 89f. Dissertação (Mestre em Ciência da Computação) Universidade Federal de Santa Catarina. Florianópolis, 2005.

Modelagem de Características para representar Informações de Contexto relevantes em um Domínio de Aplicação Sensível a Mudanças em Tempo de Execução

Diovana da Silva Santana¹, Leila Laís Gonçalves²

Universidade do Extremo Sul Catarinense (UNESC)

Caixa Postal 15.064 – 91.501-970 – Criciúma –SC – Brazil

diovana_ssantana@hotmail., llg@unesc.net

Abstract. *The methodology used is the bibliographic review of papers published between 1997 and 2017, in order to obtain a necessary theoretical basis on context sensitive computation and some examples of the use the concept of Context Sensitive System applications. It also emphasizes the study of context modeling, that shows in related works finding of information for modeling of characteristics and variability which emphasizes the userexperience in computational systems and it is focused on the elaboration of the model that was performed , the structuring of contextual information, pointing out the methodology used along the schedule. The work concludes that in the development of a new application, there is a high expectation of reusability, implying a reduction of time and cost of design, without a decrease in the quality of the works*

Resumo. *A metodologia utilizada é a revisão bibliográfica de obras publicadas entre os anos de 1997 à 2017 a fim de se obter um embasamento teórico necessário sobre computação sensível ao contexto e alguns exemplos de aplicações que utilizam este conceito. Também se enfatiza o estudo de modelagem de contexto, o levantamento de informações para modelagem de características e a variabilidade, apresenta os trabalhos correlatos, enfatizando a experiência do usuário em sistemas computacionais e se versa sobre a elaboração do modelo que foi realizado, a estruturação das informações contextuais, apontando a metodologia utilizada junto ao cronograma. O trabalho conclui que no desenvolvimento de uma nova aplicação, existe uma alta expectativa de reusabilidade, implicando em redução de tempo e custo de projeto, sem queda na qualidade dos trabalhos.*

1. INTRODUÇÃO

Nossas atividades diárias envolvem alguma forma de comunicação. Quando dois indivíduos se comunicam, de maneira natural, observam o contexto que essa comunicação acontece. Para isso, leva-se em consideração a linguagem que se compartilha, o entendimento comum de como o mundo funciona e o

entendimento implícito das situações rotineiras dos seres humanos. Atualmente, os usuários necessitam esforçar para informar preferências de uso, indicar suas necessidades, respondendo perguntas, configurando parâmetros sobre como um serviço deve se apresentar, transmitindo ideias sobre a atividade realizada, entre outros (VIEIRA; TEDESCO; SALGADO, 2009). O ideal seria que os sistemas computacionais percebessem a intenção do usuário minimizando sua complexidade e esforço na execução das tarefas. Para este cenário é necessário que as aplicações se adaptem no tempo real onde são executadas, para que assim possam fornecer resultados adequados para cada situação, tendo em consideração o contexto onde estão sendo executados. Sem a informação de contexto não é possível às aplicações se adaptarem o seu comportamento e responderem de forma personalizada (RIBEIRO et al., 2013).

A necessidade surgiu da percepção de projetistas de sistemas computacionais sobre uma demanda na necessidade dos usuários. É desejo de quem opera um sistema que seja ele adaptável, flexível, fácil de usar, enfim, interativo. A partir desta motivação, os desenvolvedores têm-se confrontado com o desafio de diminuir a necessidade do usuário interagir com o sistema e tornar os sistemas adaptáveis aos usuários com informações e serviços relevantes às suas necessidades no tempo desejado (PINHEIRO, 2014).

Sistemas sensíveis ao contexto são sistemas que gerenciam dados referentes a uma aplicação em um domínio e utilizam esses elementos com a integração da tecnologia em ambientes físicos, de forma clara para apoiar as pessoas na execução das suas tarefas diárias. As áreas da Computação Ubíqua e Inteligência Artificial foram as pioneiras nos estudos e uso do conceito de contexto, evidenciando a capacidade da aplicação desse conceito nos sistemas computacionais.

Os benefícios do sistema sensível ao contexto é que tem capacidade de fornecer serviços adaptáveis às necessidades e intenções do usuário, que permite adaptar as respostas de acordo com as preferências do usuário, a sua interação com o sistema, o momento em que a consulta está sendo realizado, o dispositivo utilizado, entre outras características (VIEIRA; TEDESCO; SALGADO, 2009). A compreensão de como o contexto pode ser usado ajudará os desenvolvedores a determinar quais comportamentos sensíveis ao contexto para suas aplicações.

Aplicações precisam ser inteligentes e utilizar os recursos disponíveis dinamicamente, tomando decisões de acordo com a situação. É importante que seja apresentado ao usuário geralmente o mesmo ambiente, as mesmas aplicações e os mesmos dados, não importando de onde ou de qual dispositivo esteja utilizando o sistema.

Nos dias atuais, cresce o interesse no desenvolvimento de aplicações sensíveis ao contexto. No entanto, entender e identificar o contexto, e executar ações de maneira automática segundo esse contexto não é uma tarefa comum em sistemas computacionais. Neste sentido, este trabalho propõe um modelo de características para auxiliar o desenvolvimento de aplicações sensíveis ao contexto.

1.1 OBJETIVOS

Desenvolver um modelo de características para representar informações de contexto em relevantes em um domínio de aplicação sensível as mudanças em tempo de execução.

Para o desenvolvimento deste projeto de pesquisa foram selecionados os seguintes objetivos:

- a) analisar sistemas sensíveis, observando elementos, comportamento nas variações de contexto e o suporte no desenvolvimento de aplicações;
- b) identificar elementos que representam informações de contexto e situações de mudança para compor o modelo de características em um domínio;
- c) descrever técnicas para modelagem de características que representam informações de contexto, e
- d) definir mecanismos para verificação das mudanças de contexto com base no modelo de características.

1.2 JUSTIFICATIVA

As áreas da Computação Ubíqua e Inteligência Artificial foram às pioneiras nos estudos e utilização do conceito de contexto e, com isso, foram as que demonstraram o potencial da aplicação desse conceito nos sistemas computacionais (VIEIRA, 2009). Pesquisas vêm utilizando o conceito de contexto para beneficiar sistemas ligados a outras áreas como, por exemplo: sistemas colaborativos, visando melhorar os serviços de percepção e prover assistência aos grupos; hipermídia adaptativa, possibilitando a personalização e adaptação do conteúdo de sites web;

integração de dados, facilitando a resolução de conflitos semânticos; e Interação Humano-Computador (IHC), com o uso do contexto para adaptar as interfaces dos sistemas tornando mais intuitiva a sua interação com os usuários (VIEIRA, 2009).

Computação ciente de contexto é um paradigma computacional no qual aplicações podem descobrir e utilizar informações contextuais, tal como a localização de um usuário, pessoas ou dispositivos próximos e a atual atividade para prover serviços apropriados para uma determinada pessoa, num determinado local, em um dado momento (ANJOS, 2006).

Dentre várias ferramentas disponibilizadas na internet, encontramos inúmeros softwares que buscam atender as necessidades das pessoas, desde localizar amigos e conhecer novas pessoas, como compartilhar ideias e gostos em comum. Com base nas informações contextuais, a aplicação pode enriquecer semanticamente a solicitação explícita do usuário e, com isso, executar serviços mais próximos às suas necessidades.

A modelagem de sistemas auxilia a entender a funcionalidade do sistema e os modelos que são usados de comunicação com os clientes. Um problema referente aos modelos de contexto usados é que, para a descrição de cenários dinâmicos, é preciso que as estruturas desses modelos precisam ser atualizadas. Assim, atualizações em modelos de contexto implicam no (re)desenvolvimento das implementações correspondentes a esses modelos (MARINHO, 2012). Um modelo contextual identifica e representa as entidades e os contextos que são relevantes para a aplicação sensível ao contexto, como por exemplo, a localização e a temperatura de uma pessoa. Um modelo de situações é construído com base no modelo de contexto e permite ao desenvolvedor explicitar situações que são de interesse da aplicação (MIELKE, 2012)

2 Computação sensível ao contexto

Sistemas que utilizam contexto são sensíveis ao meio que os cerca e capazes de realizar ações específicas de acordo com as informações necessárias para cada tarefa, são ditos sistemas inteligentes, pois conseguem adaptar-se ao ambiente. As informações são disponibilizadas aos sistemas pelo usuário bem como aqueles disponíveis em uma base de conhecimento contextual, as inferidas por um processo de raciocínio e aquelas que são percebidas a partir do ambiente (VIEIRA et

al., 2006).

Contexto pode ter diversas definições, pois se trata de um termo amplo e que abrange diversas áreas. Segundo Dey (2001), contexto é qualquer informação que pode ser aplicada para identificar a situação de uma entidade que seria uma pessoa, lugar ou objeto, tornando essa informação importante para a comunicação de um usuário e aplicativo

No entanto, projetar um CSS não é uma tarefa trivial, uma vez que é necessário lidar com questões relacionadas à: que tipo de informação deve ser considerado contexto, em como representar esta informação, como podem ser adquiridos e processados, como projetar o uso do contexto pelo sistema (SANTOS, 2008). Assim, além de lidar com entradas explícitas, também considera informação de contexto aquelas capturadas por meio de sensores, ou seja, entradas implícitas, tais como: localização, recursos e infraestrutura disponível, preferências do usuário, atividade do usuário (LOPES, 2006).

A utilização de contexto em aplicações permite que desenvolvedores possam melhorar a usabilidade da aplicação, onde o sistema reagiria de acordo com o ambiente sem intervenção do usuário, um exemplo simples seria o nível de bateria do celular quando está abaixo do normal, ele identifica e executa as ações para economizar a bateria, como reduzir a luminosidade, desativar alguns sensores (PINHEIRO, 2014). Aplicações sensíveis ao contexto identificam o ambiente no qual estão sendo utilizados e tomam decisões, estas mesmas aplicações funcionam de acordo com as ações executadas por usuários, objetos ou até mesmo por outros sistemas (PINHEIRO, 2014).

Um CSS necessita que informações contextuais sejam trocadas e utilizadas por diferentes entidades, como objetos, pessoas, dispositivo com uma mesma compreensão semântica, sendo assim um modelo de contexto precisa ser apropriado para dar suporte à ligação e o funcionamento em conjunto com outros computadores (SANTOS, 2013).

2.1 Modelagem de sistemas sensíveis ao contexto

Modelagem de contexto representa uma tarefa que estrutura uma informação contextual objetivando permitir a interpretação sem erros da informação de contexto pelos usuários, permite o processamento de informações de contexto e

sua reutilização (LAMAS, 2009).

As modelagens de sistemas sensíveis ao contexto apresentam algumas dificuldades, tais como: a caracterização dos elementos para uso na aplicação, a coleta da informação através dos dispositivos, a representação do modelo formal de contexto, o processamento e interpretação das informações de contexto, o tratamento de desempenho, segurança (VIEIRA, 2009). Estes modelos são necessários, pois representam quais informações contextuais serão levadas em consideração e como será a relação junto com a aplicação (PINHEIRO, 2014).

Sistemas sensíveis ao contexto são aqueles que gerenciam elementos contextuais relacionados a um domínio de aplicação e que usam esses elementos para apoiar na execução de alguma tarefa. Essa ajuda pode ser alcançada, melhorando a percepção do agente sobre a tarefa, ou provendo adaptações que facilitem a execução da tarefa (QUEIROZ FILHO, 2012).

Existem alguns critérios para identificar o que escolher como elemento contextual, a capacidade de relatar as propriedades úteis; possibilidade de inferência de contextos complexos; e a facilidade em ser identificado de forma automática, mais precisa possível (VIEIRA, 2009).

Características da informação mostra que o contexto dinâmico tem características constantes referentes à forma como as informações de contexto são obtidas (SOUZA, 2012), e as mesmas informações mudam instantaneamente. Na informação contextual existem algumas características, como temporais, que são estáticos e dinâmicos. (RODRIGUES, 2017). Como exemplo, se tem a localização do usuário e a atividade atual; contexto estático é onde as informações se mantêm fixas, como CPF e RG de uma pessoa (VIEIRA, 2009).

Essas características segundo Costa (2012), podem ser categorizadas de acordo com a origem: sentido (informação gerada através de sensores), derivado (a qual pode ser aprendida ou inferida), informações definidas pelo usuário (onde fornece valores à aplicação durante alguma configuração), e fornecido por um serviço (a informação é dada por algum fornecedor de serviços, como agenda ou cronograma onde há uma lista de atividades).

Para o desenvolvimento de aplicações sensíveis ao contexto é necessário entender as técnicas usadas para modelagem de contexto, essas técnicas são modelos que servem como estrutura para apoiar a modelagem de um sistema.

Serão apresentadas algumas técnicas utilizadas para modelagem de contexto, detalhando cada técnica e suas peculiaridades, com o objetivo de prover mecanismos que sirvam de base para o desenvolvimento de novas aplicações relacionadas ao paradigma de Computação Sensível ao Contexto (LONDERO, 2012).

Uma representação formal do contexto permite que mecanismos de raciocínio lógico possam ser usados para checar a consistência das informações de contexto, possam realizar comparações com outros contextos e possam inferir novas informações complexas a partir dos contextos existentes. (NUNES, 2010). Sabem que um CSS usa informações contextuais de várias fontes heterogêneas, sendo assim é necessária uma representação que permita a troca de informações entre aplicação e usuário. (PATRICIO, 2010).

3 MCSS: MODELO DE CARACTERÍSTICAS PARA REPRESENTAR INFORMAÇÕES DE CONTEXTO EM UM DOMÍNIO DE APLICAÇÃO

Esta pesquisa consistiu em desenvolver um modelo de características para representar informações de contexto relevantes em um domínio de aplicação sensível a mudanças em tempo de execução.

Através dos estudos analisamos sistemas sensíveis a contexto observando elementos, comportamento nas variações de contexto e o suporte no desenvolvimento de aplicações, identificamos quais elementos que representam informações de contexto e situações de mudança, técnicas para modelagem de características que representam informações de contexto, e os mecanismos para verificação das mudanças de contexto com base no modelo de características.

A utilização de informações de contexto permite que desenvolvedores de sistemas possam enriquecer a usabilidade de sua aplicação e, assim, o sistema sensível ao contexto pode reagir a determinadas situações sem a necessidade da interação com o usuário.

Para que um contexto possa ser representado, é fundamental que seja aplicado uma técnica. Um modelo de contexto define tipos, nomes, propriedades e atributos dos elementos nas aplicações sensíveis ao contexto, tais como usuários, dispositivos móveis.

O modelo tenta prever a representação, a busca, a capacidade de troca

de informação contextual entre as aplicações. Utilizado a técnica de modelos gráficos, pois facilita a especificação dos conceitos e definição do comportamento do sistema sensível ao contexto, podendo ser traduzido para XML e usa processamento em XML. É uma abordagem com maior foco nos humanos. A partir da modelagem gráfica é possível derivar algum código.

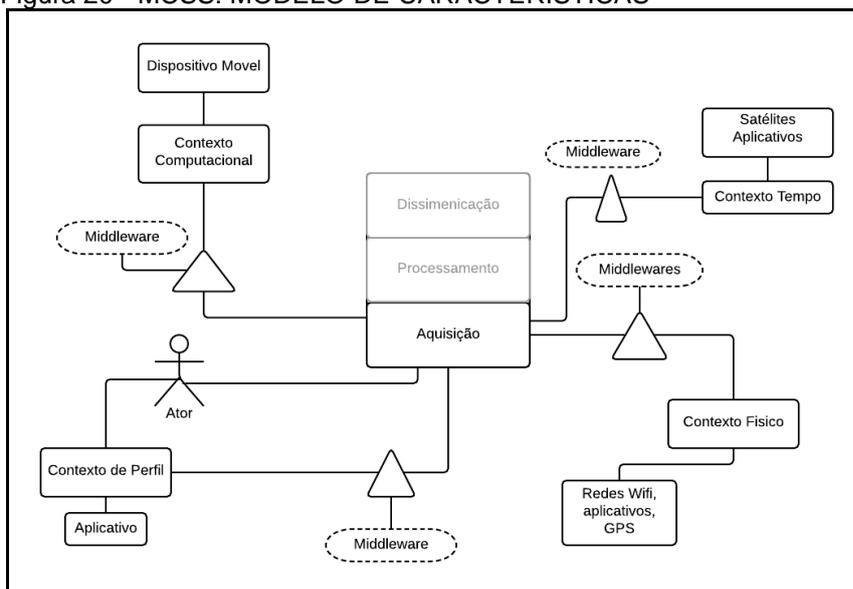
Segundo Fonteles (2013), existem algumas tarefas que um sistema sensível ao contexto segue para seu funcionamento: a) aquisição de informações contextuais, que podem ser através de sensores físicos, virtuais e lógicos – temos como informações obtidas através destes sensores a luminosidade, localização, clima, características do perfil do usuário, contatos, atividade atual, entre outros; b) elevar o nível dessas informações, essa etapa ocorre a interpretação de dados, com a utilização de técnicas, inferências ou transformações de baixo nível para o nível superior; c) armazenar e gerenciar as informações obtidas.

O contexto foi classificado em: computacional, que se refere a informações sobre um sistema computacional e seus recursos, por exemplo, o nível de bateria, consumo de memória, conexões de rede disponíveis; usuário, informações relacionadas ao perfil do usuário, como atividade atual, localização, status; contexto físico, informações sobre o mundo real, que são obtidas por meio de sensores, como a luminosidade, temperatura, localização, sensor de ruído e tempo que são informações relacionadas ao tempo de uma atividade, tem relação com a dimensão do tempo, como a hora do dia, data, estação climática.

As informações podem ser percebidas e obtidas por sensores lógicos e físicos. Quando se trata das informações de perfil do usuário, o próprio usuário fornece alguns valores a aplicação, ou podem ser obtidas automaticamente por outras aplicações. Encontramos informações contextuais em aplicações que funcionam como serviço, como por exemplo, uma agenda que é composta com compromissos e atividade do usuário. E de forma derivada, o qual pode ser apreendida ou inferida.

No modelo proposto, no processo de aquisição, apresentamos, as fontes de contexto físico, computacional, tempo e perfil do usuário. A figura 19 apresenta o modelo genérico do SSC desenvolvido.

Figura 20 - MCSS: MODELO DE CARACTERÍSTICAS



Fonte: Autor (2018)

As fontes de contexto físico proposto para o modelo é o GPS e o API Google maps, com intuito de obter informações como: localização, coordenadas, ambiente físico, e outras informações, como, pontos de referência próximos do usuário, dentre outras.

Para o processamento e disseminação das informações obtidas, utilizamos o *Middleware*, o qual serve como mediador que se encontra entre as aplicações distribuídas e suas plataformas que podem ser, processadores e suas arquiteturas, além de funções de baixo nível e APIs dos sistemas operacionais.

O contexto tempo no modelo apresentará nas informações que são utilizadas em conjunto com outras partes de um contexto, informando data e hora ou o intervalo em que uma determinada informação contextual foi percebida ou é relevante, como por exemplo, clima e estação do ano. (GOMINHO; TEDESCO; BELIAN, 2014).

O contexto computacional é responsável pela aquisição de dados de redes, hardware e software que estão sendo utilizados pelo usuário. Este contexto é necessário, por exemplo, para selecionar recursos de aprendizagem adequados ao dispositivo que está sendo usado (GOMINHO; TEDESCO; BELIAN, 2014).

Tais características permitem que sistemas sensíveis sejam capazes de identificar, coletar dados e informações por meio de interfaces de comunicação, essas informações podem ser variações do ambiente ou ações dos usuários,

possibilitando a geração de modelos computacionais automáticos, configurados conforme a necessidade do usuário, da aplicação e dos dispositivos da rede. Com essa interação é possível que cada membro do sistema seja capaz de identificar a presença de outros elementos no ambiente, tanto, usuários, como outros dispositivos eletrônicos, sendo assim possível a interação automática entre os elementos, onde pode ser construído um contexto inteligente, que potencializa o uso de sistemas pervasivos. (SILVA, 2013).

Procurou-se descrever os conceitos e as utilidades de cada elemento, abordagem, técnicas e aplicações no modelo proposto. A seguir apresentaremos um exemplo de uso do modelo

3.1 EXEMPLO DE USO

O modelo desenvolvido, apresenta etapa de aquisição das informações para uma aplicação sensível ao contexto: o usuário utiliza um contexto computacional, nesse caso, o smartphone. Ao solicitar ajuda ao sistema, o usuário terá diversas informações já armazenados num banco de dados. Essas informações como, tempo e localização, chegam ao sistema, através de sensores por via satélite, GPS, redes *Wifi* e aplicativos diversos.

As captar essas informações através destes sensores, elas passarão por mediadores que tratarão de gerenciar essas informações. Esse processo de gerenciamento, é o middleware que pode ser um API, Saas, Web Services, dentre outros.

Na situação contextual sugerida no exemplo em que se trata de um usuário com restrições alimentares, o sistema oferecerá informação baseadas nas dimensões: *who* (perfil), o modelo oferecerá informações como: nome, idade, preferências e restrições. No *where* (localização), as informações serão com base no ponto de localização, proximidade, coordenadas e rotas. No *what* (atividade e status atual), estudando, viajando, trabalhando, lazer. No *when* (momento atual), clima, trânsito, estação do ano, fuso horário, data e hora. No *Why* (motivação), alimentar-se, saúde, qualidade, diferencial, variedade e estrutura. *How* (define a coleta dos dados), já informado anteriormente.

Para aquisição de informação do contexto do perfil o desenvolvedor, precisa utilizar *API login Facebook*, onde terá informações pessoais do usuário como

nome, e-mail, status atual. Para ter outras informações como preferências, profissão, data de nascimento deve disponibilizar um formulário que o usuário informe os dados, e ficar armazenado.

Para aquisição de informações de localização é feito com auxílio da API *Location* desenvolvida pela Google, que indica a localização do aparelho e captura as alterações de local.

O contexto tempo no modelo apresentado implicará nas informações que são utilizadas em conjunto com outras partes de um contexto, informando data e hora ou o intervalo em que uma determinada informação contextual foi percebida ou é relevante, como por exemplo, clima e estação do ano. (GOMINHO; TEDESCO; BELIAN, 2014).

4 CONCLUSÃO

Diante da grande evolução computacional, o aumento de informações disponíveis e a integração de tecnologias têm impulsionado a evolução dos sistemas de software, por este motivo sistemas adaptativos tornam-se a cada dia mais essenciais. Um dos tipos mais comuns são os sistemas sensíveis ao contexto, devido ao acesso aos dispositivos móveis, a facilidade em conectar-se à internet e o aumento dos mais variados tipos de sensores que possibilitam e fornecem a informação de diversas formas. Quem procura uma informação, seja ela qual for, busca um sistema que seja adaptável, flexível e fácil de usar.

Um sistema sensível ao contexto tem a capacidade de fornecer serviços adaptáveis as necessidades e intenções do usuário. Permite adaptar resposta de acordo com as preferências do usuário e interação com o sistema no momento em que a consulta é realizada no dispositivo entre outras características.

Concluiu-se que, recomenda-se ao utilizar modelagem de características para representar um domínio de aplicação sensível a mudanças em tempo de execução. No desenvolvimento de uma nova aplicação, existe uma alta expectativa de reusabilidade (adaptar uma classe existente em um sistema a outro sistema), implicando em redução de tempo e custo de projeto, sem queda na qualidade dos trabalhos. Compreender que cada contexto tem sua especificidade, para isso, é necessário compreender as necessidades dos usuários e as ferramentas de aquisição das informações para o desenvolvimento do sistema sensível ao contexto.

Para se obter um bom proveito da modelagem, considera-se alguns aspectos que devem ser observados antes do desenvolvimento completo das aplicações, como a definição das metodologias de modelagem, análise e projeto; os padrões para a criação de desenvolvedores de sistemas possam enriquecer a usabilidade de sua aplicação.

Para aplicações comerciais onde uma base de dados é indispensável, um mapeamento forçado dos modelos é importante. Isso significa uma perda da semântica para a implementação das operações e métodos. A evolução desse paradigma busca a padronização de alguns conceitos e uma maior produtividade em uma área crítica como o desenvolvimento de software na busca de acelerar o surgimento de novas ferramentas. O que precisa ser melhor entendido, é que a utilização de modelagem de características para representar um domínio de aplicação sensível a mudanças em tempo de execução representa que se abandone práticas tradicionais e se experimente novos modelos como o desenvolvimento paralelo, onde múltiplas partes de um mesmo problema são resolvidas simultaneamente.

REFERÊNCIAS

- ANJOS, A. G. R. Aplicações móveis cientes de contexto: um gerenciador de grupos de usuários para o mobile collaboration architecture. 65f. Monografia (Graduação em Ciência da Computação) Universidade Federal de Pernambuco, Recife, 2006.
- DEY, A. K. Understanding and using context. Rev. Personal and ubiquitous computing. V. 5 Ed. 1, Fev. 2001.
- FONTELES, A. S. Um framework para aquisição adaptativa e fracamente acoplada de informação contextual para dispositivos móveis. 97f. Dissertação (Mestrado Ciências da Computação). Universidade Federal do Ceará, Fortaleza, 2013
- MARINHO, F. G. Precise: um processo de verificação Formal para modelos de características de aplicações móveis sensíveis ao contexto. 182f. Tese (Doutorado em Ciências da Computação), Ciências da Computação da UFC, Fortaleza, 2012

- PATRICIO, R. F. CEManTIKA CASE: uma ferramenta de apoio ao desenvolvimento de sistemas sensíveis ao contexto. Dissertação (Mestrado) – Universidade Federal de Pernambuco. Ciência da Computação, Recife, 2010.
- PINHEIRO, D. N. MHNCS: um middleware para o desenvolvimento de aplicações móveis cientes de contexto com requisitos de QoC. 117 f. Dissertação (mestrado Engenharia de Eletricidade). Universidade Federal do Maranhão. São Luiz, 2014.
- QUEIROZ FILHO, C. A. B. Um mecanismo de tratamento de exceções sensível ao contexto para sistemas ubíquos orientados a tarefas. 96 f. Dissertação (Mestrado em ciência da computação) - Universidade Federal do Ceará, Fortaleza-CE, 2012.
- RIBEIRO, F. R.; VICENTE, F.; RISCADO, L.; METROLHO, J. Framework para caracterização e disponibilização do contexto em espaços públicos. In: Conferência Ibérica de Sistemas e Tecnologias de Informação, 8, Lisboa. 2013. Disponível em: <www.eeexplore.ieee.org/stamp/stamp.jsp?tp>. Acesso: 24 out. 2018.
- VIEIRA, V.; TEDESCO, P.; SALGADO, A. C. Modelos e processos para desenvolvimento de Sistema Sensíveis ao Contexto. Publicado em 2009. Disponível em: <http://homes.dcc.ufba.br/~vaninha/context/2009_TextoJAI_Final.pdf> Acesso em: 12 set. 2018.