

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA**

**UMA ABORDAGEM PARA UM
AMBIENTE PERVASIVO VOLTADO AO
CUIDADO DE PACIENTES COM
DEMÊNCIA EM AMBIENTES DE
*HOME CARE***

DISSERTAÇÃO DE MESTRADO

Ederson Bastiani

Santa Maria, RS, Brasil

2013

**UMA ABORDAGEM PARA UM AMBIENTE PERVASIVO
VOLTADO AO CUIDADO DE PACIENTES COM
DEMÊNCIA EM AMBIENTES DE *HOME CARE***

por

Ederson Bastiani

Dissertação apresentada ao Programa de Pós-Graduação em Informática da
Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para
a obtenção do grau de
Mestre em Computação

Orientador: Prof. Dr. Giovani Rubert Librelotto (UFSM)

Santa Maria, RS, Brasil

2013

Bastiani, Ederson

Uma abordagem para um ambiente pervasivo voltado ao cuidado de pacientes com demência em ambientes de *homecare* / por Ederson Bastiani. – 2013.

98 f.: il.; 30 cm.

Orientador: Giovani Rubert Librelotto (UFSM)

Dissertação (Mestrado) - Universidade Federal de Santa Maria, Centro de Tecnologia, Programa de Pós-Graduação em Informática, RS, 2013.

1. Computação Pervasiva, Ontologias, *Homecare*, Demência. I. Rubert Librelotto (UFSM), Giovani. II. Título.

© 2013

Todos os direitos autorais reservados a Ederson Bastiani. A reprodução de partes ou do todo deste trabalho só poderá ser feita mediante a citação da fonte.

E-mail: edersonbastiani@gmail.com

**Universidade Federal de Santa Maria
Centro de Tecnologia
Programa de Pós-Graduação em Informática**

A Comissão Examinadora, abaixo assinada,
aprova a Dissertação de Mestrado

**UMA ABORDAGEM PARA UM AMBIENTE PERVASIVO VOLTADO
AO CUIDADO DE PACIENTES COM DEMÊNCIA EM AMBIENTES
DE *HOMECARE***

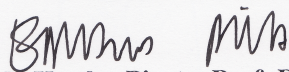
elaborada por
Ederson Bastiani

como requisito parcial para obtenção do grau de
Mestre em Computação

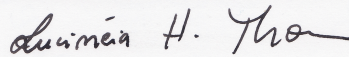
COMISSÃO EXAMINADORA:



Giovani Rubert Librelotto (UFSM), Dr.
(Presidente/Orientador)



Eduardo Kessler Piveta, Prof. Dr. (UFSM)



Lucinéia Heloisa Thom, Prof.^a Dr.^a (UFRGS)

Santa Maria, 12 de Abril de 2013.

DEDICATÓRIA

Dedico este trabalho ao meu pai **Nole Bastiani** e à minha mãe **Dirce B. Bastiani**, meus maiores exemplos, e que sempre me apoiaram. Eu começo, pai e mãe, a seguir os passos de vocês e Deus queira que eu seja tão bom quanto vocês na tarefa de ensinar.

AGRADECIMENTOS

Ao final de mais uma etapa concluída, gostaria de agradecer a todos que participaram, de alguma forma, da construção deste trabalho. Primeiramente, agradeço a **Deus**, que me deu a capacidade de aprender e coragem para seguir atrás dos meus sonhos.

À minha família, que sempre me apoiou e me incentivou a alcançar meus objetivos, sabendo contornar as situações difíceis que se apresentaram até o presente momento. Obrigado pai, **Nole**, mãe, **Dirce**, e irmãos, **Fabiane**, **Fabiola** e **Everton**.

Não poderia deixar de agradecer à minha namorada, **Angélica Bisognin Moro**, por estar ao meu lado durante este tempo, sendo compreensiva e companheira.

Ao meu orientador, Prof. Dr. **Giovani R. Librelotto**, agradeço pelas horas de orientação disponibilizadas e pelo auxílio durante esse período.

Aos professores **Deise de Brum Saccol**, **Eduardo K. Piveta** e **Lucinéia H. Thom**, deixo meu obrigado por fazerem parte da comissão examinadora deste trabalho.

Por fim, agradeço aos meus **colegas** e **amigos** pela companhia e horas de descontração, essenciais, proporcionadas ao longo deste trajeto.

Muito obrigado a todos!

*“Não faça do amanhã o sinônimo de nunca,
nem que o ontem seja o mesmo que nunca mais.
Teus passos ficaram. Olhes para trás, mas vá em frente,
pois há muitos que precisam que chegues para poderem seguir-te.”*

— CHARLIE CHAPLIN

RESUMO

Dissertação de Mestrado
Programa de Pós-Graduação em Informática
Universidade Federal de Santa Maria

UMA ABORDAGEM PARA UM AMBIENTE PERVASIVO VOLTADO AO CUIDADO DE PACIENTES COM DEMÊNCIA EM AMBIENTES DE *HOMECARE*

AUTOR: EDERSON BASTIANI

ORIENTADOR: GIOVANI RUBERT LIBRELOTTO (UFSM)

Local da Defesa e Data: Santa Maria, 12 de Abril de 2013.

A crescente quantidade de pessoas idosas vem exigindo cada vez mais infraestrutura dos sistemas de saúde atuais devido, principalmente, à alta incidência de doenças associadas ao perfil idoso. Dentre elas, uma das mais preocupantes é a demência, uma síndrome que afeta progressivamente as funções cognitivas e motoras dos pacientes, os quais tornam-se dependentes de cuidados e monitoramento constantes. O cuidado em *homecare* pode ser uma alternativa interessante para evitar um possível colapso hospitalar, proporcionando cuidados aos pacientes em sua própria casa. No entanto, o ambiente domiciliar pode se tornar complexo, visto que muitas vezes o cuidador é alguém da própria família e que não possui a técnica correta de monitoramento ou está desatento em função do desgaste provocado pelo intenso cuidado ao paciente. Assim, torna-se necessário o desenvolvimento de soluções que diminuam a sobrecarga do cuidador, ao passo em que o paciente continue sendo monitorado. Uma maneira de viabilizar tais soluções é através da aplicação de conceitos de computação pervasiva em ambientes domiciliares. Através de um ambiente povoado de sensores e de outros dispositivos computacionais, o paciente pode ser monitorado constantemente e tal ambiente pode se adaptar às necessidades deste usuário, antecipando-se a situações de acordo com o contexto atual. Para isso, é necessário que o contexto seja representado de maneira clara e com a semântica de suas entidades bem definida. Considerando isso, o presente trabalho apresenta uma abordagem para ambientes pervasivos voltados ao cuidado de pacientes com demência em ambientes de *homecare*. Tal abordagem é composta por uma arquitetura para o desenvolvimento de sistemas pervasivos, bem como uma ontologia para representação do conhecimento de ambientes pervasivos direcionados a pacientes com este perfil. Acredita-se, com a modelagem proposta, preservar a autonomia dos pacientes, bem como estimular suas funções cognitivas remanescentes, além de diminuir a sobrecarga sofrida por seus cuidadores.

Palavras-chave: Computação Pervasiva, Ontologias, *Homecare*, Demência.

ABSTRACT

Master's Dissertation
Program in Computer Science
Universidade Federal de Santa Maria

AN APPROACH FOR PERVASIVE HOMECARE ENVIRONMENTS FOCUSED ON CARE OF PATIENTS WITH DEMENTIA

AUTHOR: EDERSON BASTIANI

ADVISOR: GIOVANI RUBERT LIBRELOTTO (UFSM)

Defense Place and Date: Santa Maria, April 12th, 2013.

The increasing number of elderly people is demanding increasingly infrastructure of health systems mainly due to the incidence of diseases associated with older profile. Among these, one of the most worrying is dementia, a syndrome that progressively affects cognitive and motor functions of patients, who become dependent care and constant monitoring. The home care could mean an interesting alternative to avoid a possible hospitalar collapse , providing care to patients in their own home. However, the home environment can become complex, since often the caregiver is someone of the family and does not have the correct technique of monitoring or being inattentive due to wear caused by intense patient care. Thus it becomes necessary to develop solutions which reduce the overhead of the caregiver while the patient continues to be monitored. One way to achieve such solutions is by applying concepts of pervasive computing in home environments. Through an environment populated by sensors and other computing devices the patient can be monitored constantly and the environment can adapt to the needs of the user, anticipating the situations according to the current context. For this, the context must be represented clearly and with the semantics of their entities well-defined. Considering this, this work presents a approach for pervasive environments aimed to the care of the patients with dementia in homecare environments. This approach is composed of an architecture for development of pervasive systems, and an ontology for knowledge representation environments pervasive oriented to patients with this profile. It is believed, with the approach proposed, preserve the autonomy of patients as well as stimulate their cognitive functions remaining, besides to reduce the burden suffered by their caregivers.

Keywords: Pervasive Computing, Ontologies, Homecare, Dementia.

LISTA DE FIGURAS

6.1	Hierarquia de classes da ontologia proposta por Freitas <i>et al.</i> (2012)	45
6.2	Hierarquia da ontologia de <i>homecare</i> voltado a portadores de demências	45
6.3	Exemplo de relacionamento na ontologia	48
7.1	Arquitetura para sistemas de <i>homecare</i> pervasivos voltados ao cuidado de pacientes com demência.	55
7.2	Fluxo das informações que entram no módulo de monitoramento e entrada de dados.	58
7.3	Esquema do arquivo XML com informações detectadas pelos sensores (sensor.xml).	59
7.4	Fluxo de informações no módulo <i>OntoHC</i>	61
7.5	Esquema do arquivo XML que informa as classes encontradas no contexto atual (nuvem.xml).	62
7.6	Esquema do arquivo XML utilizado para cadastrar prescrições médicas (prescricao.xml).	63
7.7	Esquema do arquivo XML que representa uma atividade cognitiva (atividades.xml).	64
7.8	Esquema do arquivo XML que descreve os campos da base de dados de saúde do sistema PEP (pep.xml).	65
7.9	Fluxo do processo de criação de uma nova ontologia de contexto atual	66
7.10	Fluxo do processo de notificação.	67
7.11	Esquema do arquivo XML para interação com os dispositivos computacionais (dispComp.xml).	67
8.1	Fluxo de um notificação de prescrição médica	72
8.2	Ilustração do aplicativo móvel apresentando um lembrete ao paciente	75
8.3	Ilustração do aplicativo móvel apresentando um exercício cognitivo ao paciente	78

LISTA DE TABELAS

4.1	Prevalência de demência nas diferentes faixas etárias.....	27
9.1	Comparativo entre trabalhos relacionados	81
A.1	Tabela de propriedade de dados da ontologia	92
B.1	Tabela de propriedades de objetos da ontologia	94
C.1	Tabela de restrições da ontologia.....	96

LISTA DE ABREVIATURAS E SIGLAS

DA	<i>Doença de Alzheimer</i>
DAML	<i>DARPA Agent Markup Language</i>
DTD	<i>Document Type Definition</i>
DFT	<i>Demência Fronto-Temporal</i>
DV	<i>Demência Vascular</i>
ER	<i>Entidade-Relacionamento</i>
HTML	<i>HyperText Markup Language</i>
OIL	<i>Ontology Inference Layer</i>
PEP	<i>Prontuário Eletrônico do Paciente</i>
RDF	<i>Resource Description Framework</i>
RDFS	<i>RDF Schema</i>
SGML	<i>Standard Generalized Markup Language</i>
SQL	<i>Structured Query Language</i>
SQWRL	<i>Semantic Query-enhanced Web Rule Language</i>
SWRL	<i>Semantic Web Rule Language</i>
UML	<i>Unified Modeling Language</i>
URI	<i>Uniform Resource Identifier</i>
W3C	<i>World Wide Web Consortium</i>
XML	<i>Extensible Markup Language</i>

SUMÁRIO

1	INTRODUÇÃO	15
2	COMPUTAÇÃO PERVASIVA	18
2.1	Computação sensível ao contexto	19
2.2	Computação pervasiva aplicada à saúde	20
2.3	Considerações do capítulo	21
3	HOMECARE	22
3.1	Rede de <i>homecare</i>	23
3.2	Sistemas de <i>homecare</i>	24
3.3	Desafios de sistemas <i>homecare</i>	25
3.4	Considerações do capítulo	26
4	DEMÊNCIAS	27
4.1	Doença de <i>Alzheimer</i>	28
4.2	Demência Vascular	29
4.3	Demência Fronto-Temporal	30
4.4	Demência com <i>Corpos de Lewy</i>	31
4.5	Outras causas de demências	31
4.6	Considerações do capítulo	32
5	ONTOLOGIAS	33
5.1	Motivação para o uso de ontologias	34
5.2	Classificação das ontologias	36
5.3	Representação de ontologias	37
5.3.1	XML	37
5.3.2	RDF	38
5.3.3	RDF (S)	39
5.3.4	OWL	39
5.3.5	OWL 2	40
5.4	Inferências sobre ontologias	41
5.4.1	SWRL	41
5.4.2	SQWRL	42
5.5	Considerações do capítulo	42
6	MODELAGEM DE UMA ONTOLOGIA PARA REPRESENTAÇÃO DO CONHE- CIMENTO DE UM AMBIENTE HOMECARE PERVASIVO VOLTADO A POR- TADORES DE DEMÊNCIA	43
6.1	Domínio e escopo da ontologia	43
6.2	Classes	44
6.3	Propriedades de dados	46
6.4	Propriedades de objetos	47
6.5	Restrições	48
6.6	Instâncias	49
6.7	Consultas	50

6.8	Inferências	51
6.9	Considerações do capítulo	53
7	ARQUITETURA DE UM SISTEMA HOMECARE PERVASIVO VOLTADO A PACIENTES PORTADORES DE DEMÊNCIA.....	54
7.1	Prontuário eletrônico do paciente	57
7.2	Sensores	57
7.3	Módulo de monitoramento e entrada de dados	58
7.4	Módulo OntoHC	60
7.5	Módulo do plano de cuidados	62
7.6	Módulo de processamento da nuvem computacional	64
7.7	Módulo de notificação	66
7.8	Dispositivos computacionais	68
7.9	Considerações do capítulo	69
8	ESTUDOS DE CASOS E RESULTADOS	70
8.1	Estudo de caso 1: Notificação de prescrição médica	70
8.2	Estudo de caso 2: Exercício de estimulação cognitiva	76
8.3	Considerações do capítulo	79
9	TRABALHOS RELACIONADOS	80
10	CONCLUSÃO	83
	REFERÊNCIAS	85
	APÊNDICE A PROPRIEDADES DE DADOS DA ONTOLOGIA	92
	APÊNDICE B PROPRIEDADES DE OBJETOS DA ONTOLOGIA	94
	APÊNDICE C RESTRIÇÕES DE PROPRIEDADES NA ONTOLOGIA.....	96
	APÊNDICE D CONSULTAS E INFERÊNCIAS	97

1 INTRODUÇÃO

A maioria dos países está passando por um processo de envelhecimento populacional. No Brasil, uma pesquisa do Instituto Brasileiro de Geografia e Estatística, revelou que em 2008 havia 21 milhões de idosos dentre os 190 milhões de brasileiros, correspondendo a 11,1% da população (IBGE, 2010). Fatores como a melhora na qualidade de vida da população, controle de natalidade e avanço tecnológico são importantes contribuintes para esse irreversível fenômeno.

Em paralelo ao envelhecimento populacional, a maioria dos países enfrentará desafios econômico-sociais no cuidado da saúde em função, principalmente, do aumento significativo de incidência das doenças crônico-degenerativas, sendo a demência uma das mais preocupantes. Estima-se que 35,6 milhões de pessoas são portadoras de demência e até 2050 este número deve ser três vezes maior (WHO, 2012).

O termo demência pode ser definido como o comprometimento da memória associado a um prejuízo em pelo menos uma das outras funções cognitivas da linguagem, gnosis, praxias ou funções executivas, que interferem na capacidade funcional ou social do paciente (RAMOS et al., 2009). Em sua maioria, demências apresentam um quadro neurodegenerativo, progressivo e irreversível. O tratamento foca-se em minimizar os efeitos da doença, controlando os sintomas e diminuindo o progresso com a utilização de medicamentos e a realização de atividades físicas e mentais. Logo, em muitos casos, é necessário que os pacientes recebam cuidados e monitoramento constante por parte de um cuidador, extrapolando a capacidade e infraestrutura dos atuais sistemas de saúde.

Uma das maneiras de minimizar esses problemas é realizar o tratamento de pacientes em ambientes domiciliares através de serviços de *homecare*. Esse tipo de cuidado se propõe a prestar serviços de saúde a um paciente em sua própria casa ao invés de deslocar o paciente até o hospital, seja por necessidade, quando o paciente é impossibilitado de se locomover, ou por preferência do paciente, uma vez que estará em um ambiente familiar. No entanto, por se tratar de um ambiente domiciliar, o monitoramento do paciente se torna complexo pela grande dinamicidade, uma vez que o paciente pode circular pela casa, receber visitas ou realizar as atividades de rotina.

Outro problema diz respeito ao cuidador, que chama para si a incumbência de realizar as atividades diárias que o paciente não tem mais condições. Com a evolução da doença, os idosos passam a ser progressivamente dependentes dos cuidadores. Estes podem estar desatentos em

função do desgaste físico ou emocional propiciado pelo intenso cuidado. Logo, a saúde do paciente fica vulnerável a riscos relacionados ao monitoramento. Ainda, o cuidador pode não possuir domínio da técnica de monitoramento, visto que em muitos casos o cuidador é algum familiar do próprio paciente.

A computação pervasiva aliada à ideia de *homecare* apresenta-se como uma solução promissora para solucionar problemas deste tipo. Pode-se descrever a computação pervasiva como um novo paradigma computacional com tecnologia de comunicação e informação em qualquer lugar, acessível por qualquer pessoa, disponível a todo tempo e de forma invisível ao usuário, ou seja, o usuário foca-se na tarefa que está realizando, sem perceber a infra-estrutura computacional ao seu redor (SAHA; MUKHERJEE, 2003).

Tratando-se de ambientes de *homecare*, dados fisiológicos do paciente e condições do ambiente podem ser coletados em momentos variados do dia de forma proativa, fazendo com que o ambiente seja capaz de se adaptar as necessidades do usuário e capaz de reagir a situações críticas de saúde, conforme o contexto atual. Porém, para que seja possível tal adaptação, é necessário que a representação do conhecimento do domínio em questão seja detalhada.

Uma das maneiras mais utilizadas para representação do conhecimento é através do uso de ontologias. Em uma ontologia, os relacionamentos são definidos formalmente e a semântica de um dado relacionamento é detalhada. Se esses relacionamentos possuem nomes apropriados que identifiquem seu significado, um humano pode entendê-la diretamente, assim como um programa pode assumir a semântica de um dado relacionamento e atuar semanticamente através da mesma (GASSEN, 2010).

Em ambientes de *homecare*, diversas entidades podem ser encontradas, tais como enfermeiro, médico, paciente, visitantes, além de diversos dispositivos computacionais dispersos no ambiente, tornando difícil de representar computacionalmente a forma com que cada uma dessas entidades expressa seu contexto atual. Logo, tendo em vista o seu poder de expressividade, as ontologias podem ser utilizadas para a descrição desses ambientes.

Nesse contexto, baseando-se nos conceitos de computação pervasiva e ontologias, essa dissertação apresenta uma abordagem para o desenvolvimento de sistemas pervasivos voltados a auxiliar no tratamento de pacientes portadores de demência em suas casas. Mais precisamente, a abordagem proposta busca resolver questões como a integração de sensores e dispositivos computacionais, monitoramento constante e preservação da autonomia do paciente, e diminuição da sobrecarga sofrida pelo cuidador ao passo em que o paciente fique em segurança e continue

recebendo os cuidados necessários.

De forma a apresentar tal abordagem, o documento está organizado da seguinte forma: No Capítulo 2 são descritos os conceitos e características da computação pervasiva, bem como a importância de sua aplicação na área da saúde. A seguir, o Capítulo 3 apresenta conceitos de ambientes de *homecare*, bem como a descrição das características destes sistemas. A seguir, o Capítulo 4 apresenta as principais causas de demência e seus sintomas característicos. No Capítulo 5 são detalhados conceitos de ontologias, áreas de utilização e classificação, assim como as formas e linguagens de representação e consultas.

Como parte da abordagem proposta, o Capítulo 6 apresenta a modelagem de uma ontologia para a representação do conhecimento no âmbito de ambientes de *homecare* pervasivos voltados ao cuidado de pacientes com demência, seguida da modelagem de uma arquitetura para o desenvolvimento de sistemas pervasivos para este tipo de ambiente, a qual é apresentada no Capítulo 7, bem como a descrição de cada um dos módulos que compõem tal arquitetura. Com o objetivo de validar e exemplificar a abordagem proposta, no Capítulo 8 é utilizado um caso de estudo já publicado na literatura e outro criado para demonstrar o fluxo de funcionamento de um sistema desenvolvido com base na nesta abordagem, além da descrição dos resultados obtidos.

A comparação entre o trabalho apresentado e outros existentes na literatura é descrita no Capítulo 8. Por último, no Capítulo 9, apresentam-se as conclusões do trabalho, através da descrição das contribuições e trabalhos futuros.

2 COMPUTAÇÃO PERVASIVA

A computação pervasiva é considerada o novo paradigma computacional do século XXI com tecnologia de comunicação e informação em qualquer lugar, acessível por qualquer pessoa, disponível a todo o tempo, onde recursos computacionais devem estar integrados ao ambiente físico de forma mais transparente possível (SAHA; MUKHERJEE, 2003). Tal paradigma remete a um ambiente no qual os serviços e os dispositivos dispostos em um ambiente descobrem-se uns aos outros e integram-se provendo um serviço ao usuário sem sua interferência direta.

A proposta da computação pervasiva é oriunda da visão de Mark Weiser (WEISER, 1991), denominada computação ubíqua, sobre o futuro da tecnologia onde computadores estarão totalmente integrados ao ambiente e a vida dos usuários, de forma que sua utilização se dará de forma invisível. Segundo Weiser, "as tecnologias mais duradouras e intensas são aquelas que desaparecem". Ou seja, elas dissipam-se nas coisas do dia-a-dia ao ponto de tornarem-se indistinguíveis e assim, o foco do usuário volta-se totalmente para a tarefa a ser executada e não mais para a ferramenta que será empregada na realização dessa tarefa.

Este novo paradigma computacional é considerado a terceira onda da computação, onde a primeira foi a era dos *Mainframes*, onde um computador era compartilhado por diversas pessoas através de *workstations*. Em seguida, a segunda era foi marcada pelos computadores pessoais, de modo que cada computador era utilizado por uma pessoa. Por fim, iniciou-se a era da computação pervasiva, terceira onda, caracterizada por uma pessoa e muitos computadores, com milhões de computadores embutidos no ambiente de forma a se tornar praticamente invisível (SOYLU; CAUSMAECKER; DESMET, 2009).

Diferentemente da computação tradicional, a computação pervasiva prevê que os dispositivos e as aplicações devem ser capazes de se adaptar ao meio no qual estão inseridos. O ambiente deve ser capaz de detectar e reagir a outros dispositivos que venham a fazer parte dele ou então os dispositivos devem ser capazes de detectar os diferentes ambientes. Dessa interação surge a capacidade de computadores agirem de forma inteligente em um ambiente heterogêneo povoado por sensores e serviços computacionais (ARAUJO, 2003).

Neste cenário, a infraestrutura deve suportar a mobilidade do usuário e das aplicações e serviços, além de ser amplamente distribuída. Logo, aplicações pervasivas devem ser desenvolvidas seguindo a *semântica siga-me*. A semântica siga-me prevê que as aplicações "sigam" seus usuários fisicamente, ou seja, a aplicação acompanha seu usuário esteja ele portando seu dispo-

sitivo ou não. Para que as aplicações tenham esse comportamento, elas devem ser distribuídas, móveis e adaptativas ao contexto (YAMIN, 2004).

2.1 Computação sensível ao contexto

Uma das principais ideias da computação pervasiva é que o ambiente pervasivo tenha ciência de contexto, onde dispositivos e aplicações tenham capacidade de detectar mudanças que possam ocorrer no ambiente em que se encontram. Essa característica define a computação sensível ao contexto (LOUREIRO et al., 2009).

A computação sensível ao contexto define uma área de pesquisa que possui aplicações em diferentes cenários computacionais e que apresenta desafios de implementação importantes, os quais têm sido o alvo da atenção de pesquisadores provenientes de diferentes partes do mundo. Resumidamente, a proposta desta área é elaborar uma maneira de coletar as condições atuais do usuário, do ambiente em que esse usuário se encontra e dos dispositivos computacionais dispostos em tal ambiente. Assim, além de lidar com informações explícitas, a computação sensível ao contexto também considera entradas implícitas, por meio de sensores, como localização, recursos de infraestrutura disponíveis, preferências e atividade do usuário, números de dispositivos, carga computacional, etc (CHEN; KORTZ, 2000).

Conceitualmente, existem diferentes definições para contexto. De acordo com Dey e Abond (2006), contexto pode ser definido como qualquer informação que pode ser usada para caracterizar a situação atual de uma entidade. Uma entidade é uma pessoa, lugar ou objeto que é considerado relevante para interação entre o usuário e a aplicação, incluindo o usuário e a aplicação em si. Já em (YAMIN et al., 2003), contexto é dito como qualquer informação gerada pela infraestrutura computacional do ambiente que pode ser relevante para uma aplicação, na qual uma alteração pode disparar um processo para a adaptação desta aplicação.

Com o surgimento da nova geração de dispositivos móveis (PDAs, Smartphones), a computação sensível ao contexto ganhou uma atenção especial. Porém, o desenvolvimento de ferramentas que dão suporte a ela traz à tona diversos desafios como a obtenção, modelagem, armazenamento, distribuição e monitoramento do contexto. Na computação móvel, o contexto é formado por dois aspectos : o primeiro se refere às características do ambiente que determinam o comportamento das aplicações; o outro aspecto diz que contexto é um conjunto de estados e configurações do ambiente que determina um comportamento da aplicação (CHEN; KORTZ, 2000).

2.2 Computação pervasiva aplicada à saúde

Uma das áreas de aplicação mais relevantes para tecnologias pervasivas é a saúde (*Pervasive Healthcare*). Existem diferentes pesquisas neste âmbito abordando desde um ambiente por inteiro, como um problema específico.

Em (BARDRAM, 2004) é discutida a relevância da sensibilidade do contexto integrada à sistemas de Prontuário Eletrônico de Pacientes (PEP), fazendo com que estes consigam se adaptar às mudanças do ambiente. Os autores apresentam um protótipo de aplicação clínica, que inclui um PEP, uma cama e um *dispenser* de medicamentos sensíveis ao contexto. Essa cama possui uma tela que, na maior parte do tempo, é utilizada como um aparelho de TV normal para entreter o paciente. Porém, quando um médico entra no quarto, a cama, através de um conjunto de sensores RFID, detecta a mudança no ambiente e passa a mostrar na tela informações relevantes ao médico no tratamento do paciente como, por exemplo, resultados de exames. Por sua vez, o *dispenser* também utiliza-se de sensores. Quando o paciente se aproxima de um *dispenser*, o sistema procura a tela mais próxima do paciente para indicar onde está o equipamento que contém o medicamento prescrito que deve ser utilizado naquele momento. Nesse contexto, propostas como estas podem ser utilizadas na construção de um ambiente *homecare* pervasivo para portadores de demência.

Algumas propostas interessantes para alcançar a pervasividade no cuidado da saúde são propostas por (HODGINS et al., 2008), onde diversos produtos, nomeados implantes médicos inteligentes, estão sendo desenvolvidos. Entre eles estão sensores, que podem ser implantados dentro do corpo humano para coleta de dados, além de ter a capacidade de comunicação com dispositivos externos através de tecnologias de comunicação sem fio para enviar os dados coletados. Estes sensores podem ser utilizados para prover dados relevantes a aplicações sensíveis ao contexto. O projeto também tem desenvolvido fontes de energia autossustentáveis, que derivam energia diretamente de fluídos do corpo humano, afim de alimentar esses implantes.

O trabalho de (WOOD et al., 2008) propõe um sistema de monitoramento e assistência ao dia-a-dia que integra sensores fisiológicos, ambientais e de atividades. O sistema possui um *middleware* de rede extensível e heterogêneo, o qual integra dispositivos embarcados, sistemas de *back-end*, análises remotas e interfaces de usuário. As informações são processadas pelo sistema e apresentadas para análise remota pelos profissionais. Porém, torna-se limitado, visto que não prevê a utilização de conhecimento de profissionais da saúde.

Aproximando-se da ideia desta dissertação, Copetti *et al.* (2008) propõem um *framework*

inteligente, consciente do contexto, para monitoramento de pacientes em ambientes *homecare*. Este *framework* inclui recursos para gerenciamento de contexto e serve para raciocínio e funções de aprendizagem. Foi desenvolvido um protótipo que permite identificar situações anormais de saúde através dos dados coletados. Assumindo a infraestrutura de computação pervasiva, os autores baseiam-se em um conjunto de variáveis que devem ser relevantes para o monitoramento de um paciente neste tipo de ambiente. Além dos dados fisiológicos, são monitoradas as condições do ambiente (luz, umidade, som, etc), e atividades do paciente, como dormir e caminhar. Informações deste tipo são importantes para avaliar o real estado de saúde do paciente.

2.3 Considerações do capítulo

Este capítulo apresentou uma panorama geral sobre computação pervasiva, citando alguns de seus principais conceitos. Foram descritos também alguns trabalhos que buscam aplicar esse paradigma à área da saúde, melhorando o tratamento e o cuidado de pacientes. Trabalhos que se aproximam da idéia desta dissertação serão apresentados em uma seção à parte (Seção 9). O próximo capítulo tem o objetivo de detalhar a ideia de ambientes *homecare*.

3 *HOMECARE*

O crescente aumento no número pessoas idosas com uma variedade de doenças, incapacidades ou deficiências, está cada vez mais pressionando os recursos dos sistemas de saúde existentes. Com isso, a prestação de cuidados em ambientes *homecare* é visto como uma alternativa promissora para soluções de saúde tradicionais (WANG; TURNER, 2008).

O termo *homecare* pode ser definido como um conjunto de serviços ligados de assistência social, cuidados da saúde, ou ambos, que permitem a prestação de cuidados em casa (MCGEE-LENNON, 2008). Ou seja, pacientes que antes eram tratados basicamente em um hospital, agora são levados para continuarem seus tratamentos em sua própria casa.

Essa mudança na maneira como os pacientes recebem cuidados vem sendo impulsionada por uma série de benefícios. Primeiramente, o paciente sente-se mais confortável em um ambiente familiar do que num hospital, onde permanece mais tempo com seus familiares e amigos, além de reduzir o risco de infecção hospitalar. Outro benefício é o econômico, visto que está cada vez mais complexo para as entidades hospitalares fornecerem serviços de qualidade a baixo custo devido ao envelhecimento populacional. Ainda, tratando-se de idosos, o *homecare* insere-se dentro de um modelo gerontológico que visa preservar sua autonomia e manter o paciente um cidadão ativo, participativo e produtivo, aumentando assim sua qualidade de vida (MCGEE-LENNON, 2008) (FLORIAN; SCHRAMM, 2004).

O cuidado em *homecare* é auxiliado pelas tecnologias atuais. Estas tecnologias podem ser utilizadas para monitorar situações onde o paciente esteja imóvel ou incapaz, e necessitam de alguma intervenção externa, assim como a comunicação entre paciente, amigos, familiares ou equipe médica e social envolvida, mesmo quando eles não se encontram no mesmo ambiente de tratamento.

Os serviços oferecidos por sistemas de *homecare* variam desde aplicações simples, como alarmes eletromecânicos independentes instalados na casa do paciente e que o avisam quando uma banheira está transbordando ou uma porta ficou entreaberta, até sistemas integrados à infraestrutura da casa que monitoram constantemente o estado de saúde do paciente e que tem a capacidade realizar análises sofisticadas, trocar informações customizadas entre médico e paciente e suportar uma comunicação remota entre eles (MCGEE-LENNON, 2008).

3.1 Rede de *homecare*

Em um sistema de *homecare* existe um conjunto de entidades (pessoas e organizações) envolvidas direta ou indiretamente no cuidado do(s) paciente(s). Estas entidades podem ser divididas em diferentes categorias (MCGEE-LENNON; GRAY, 2007):

- O paciente: a pessoa cuja saúde está sendo cuidada é a entidade central de um sistema de atendimento domiciliar. Um paciente pode estar recebendo tratamento para mais de um problema e, ainda, é possível que em um mesmo ambiente *homecare* exista mais de uma pessoa sendo cuidada. Por exemplo, o marido pode ter um problema cardíaco e sua mulher pode estar sob cuidados de diabetes;
- Cuidadores: definidos como as pessoas que prestam os primeiros atendimentos médicos, de enfermagem ou de assistência social ao paciente. Muitas vezes o cuidador é um membro da família e essa relação de cuidados pode tornar-se complexa quando o cuidador também necessita de cuidados. Por exemplo, um cuidador idoso (esposa, marido, etc). Este tipo de situação exige que sistemas de *homecare* sejam configurados para lidar com diferentes usuários, em momentos e situações diferentes, ou seja, de acordo com o contexto. Ainda, é necessário considerar que os cuidadores podem querer acessar as informações sobre o cuidado e a saúde do paciente remotamente;
- Visitantes: os visitantes incluem profissionais como assistentes sociais, enfermeiros, e paramédicos, que são chamados para dar uma assistência. A diferença entre visitantes e cuidadores está, basicamente, na frequência da visita. Se o profissional realiza visitas regularmente e de forma planejada então ele é classificado como cuidador. Porém, se a visita é ocasional, decorrente de alguma situação inesperada (e.g., parada cardíaca), então o profissional é classificado como visitante. Pessoas que entram no ambiente de *homecare* também são incluídas neste grupo, como amigos que visitam o paciente;
- Usuários Remotos: neste grupo estão incluídos os membros da equipe médica e social que farão uso das informações do sistema *homecare* sem ir até a casa do paciente. Os usuários remotos podem ser também membros da família que vivem em outros lugares e querem manter-se informados sobre a condição de saúde ou em contato com o paciente;
- Fornecedores de tecnologia: pessoas que projetam, produzem e distribuem a tecnologia que podem influenciar diretamente o funcionamento do sistema;

- Investidores institucionais: são pessoas que não são usuárias diretas do sistema, mas que, de alguma forma, possam influenciar sua forma ou conteúdo. Por exemplo, a administração de uma associação de habitação pode impor regras sobre o tipo de tecnologia que pode ser utilizada nas suas casas;
- Outros envolvidos: além dos grupos já especificados, podem existir outras pessoas que tenham interesses no sistema, mesmo que não o utilizem. Essas pessoas podem ser membros da família que vivem em outros lugares, mas possuem alguma obrigação financeira e/ou moral com o cuidado. Também estão inclusos vizinhos que estão preocupados com o efeito potencial da tecnologia em suas casas;

3.2 Sistemas de *homecare*

Existe uma ampla gama de tecnologias que podem apoiar os cuidados em ambiente domiciliar atualmente, desde sensores simples até um conjunto de tecnologias pervasivas interligadas. Porém, de acordo com (MCGEE-LENNON; GRAY, 2007), os sistemas voltados para aplicações de *homecare* devem possuir algumas características básicas, descritas a seguir.

Sensores devem ser capazes de fornecer dados sobre o estado do paciente. Além de dados fisiológicos, como temperatura, pressão arterial ou frequência cardíaca, estes sensores devem levar em conta o contexto do ambiente, como localização e a atividade que o usuário está realizando. Os sensores são distribuídos de forma variada, desde equipamentos ligados ao paciente, até sensores incorporados no ambiente, como uma câmera fixada na parede.

A maioria dos sistemas existentes ou objetos de pesquisa são multiusuários. De modo geral, qualquer pessoa envolvida em torno da pessoa que está sendo cuidada pode ser considerada um usuário em potencial. A pessoa que recebe cuidado é, muitas vezes, fonte de informações, mas pode também ser um usuário de serviços como lembretes e informações de gestão. Já os profissionais de saúde podem, ou não, utilizar as tecnologias em si, mas provavelmente utilizarão o sistema como forma de auxílio em seu trabalho.

As informações, de diversos sensores, podem ser relacionadas e utilizadas para construir o cenário atual do ambiente, como detecção de situação anormal de saúde ou necessidade de uma consulta remota. Uma questão importante é relacionar o que está acontecendo em casa com o mundo externo, de forma distribuída. Assim, através da utilização de tecnologias de redes de computadores sofisticadas e gerenciamento de software, uma variedade de situações adversas podem ser detectadas e alertadas para os usuários do sistema, de modo que eles possam

atender adequadamente e com rapidez a cada cenário em particular. Essas informações podem ser entregues através de várias formas de interação com os usuários, como dispositivos móveis, TV digital e sistemas de áudio.

Outra característica importante é a de que sistemas de *homecare* devem suportar aplicações dinâmicas, devido ao alto grau de heterogeneidade. Cada usuário envolvido na rede de cuidados pode possuir diferentes necessidades, perspectivas e responsabilidades que variam com o decorrer do tempo. Esses usuários desejam obter e analisar as informações de saúde por diferentes dispositivos de entrada e saída e de diferentes formas, por exemplo, uns através de texto, outros de imagens.

Um sistema dinâmico, que atende as necessidades de cada usuário de forma individual, é capaz de melhorar consideravelmente a qualidade do serviço quando comparado a um sistema com um tipo único de resposta aos usuários. Essa abordagem centrada na pessoa exige que as informações importantes sobre o paciente sejam coletadas, atualizadas e comunicadas sistematicamente. Geralmente os responsáveis pelos cuidados do paciente, sejam familiares ou médicos, acreditam ter um conhecimento único das necessidades e preferências do paciente. Dessa forma, deveriam ter o poder de influenciar a forma como os serviços são executados, além de implementar mudanças de acordo com as novas necessidades do paciente.

3.3 Desafios de sistemas *homecare*

Tratando-se de um ambiente bastante heterogêneo, como é o caso de *homecare*, diversos conflitos podem surgir durante a implementação de um sistema devido as divergências de necessidades e expectativas dos usuários envolvidos, por exemplo, quando um usuário não consegue entender de outros usuários ou interage de forma incorreta com o sistema. Para minimizar os danos causados por tais conflitos, é necessário que eles sejam identificados e descritos de forma que suas estruturas originem alguma solução (MCGEE-LENNON, 2008).

No tratamento de pacientes em suas casas é comum, principalmente tratando-se de idosos, que tenha-se um conjunto de condições de saúde para gerenciar e estas situações podem estar interligadas. Logo, um sistema *homecare* deve ser capaz de lidar com as situações onde hajam informações conflitantes. Assim, o sistema precisa analisar corretamente as informações recebidas para assim executar o serviço correto, visto que uma informação interpretada erroneamente pode comprometer seriamente a saúde do paciente. Há ainda situações onde as informações múltiplas não são referentes apenas a uma pessoa, mas são propagadas entre quem

está no ambiente.

O ambiente domiciliar geralmente inclui outras pessoas além do paciente, o que pode gerar problemas relacionados ao modo pelo qual a informação é apresentada aos usuários. Por exemplo, um paciente pode receber as informações do seu estado de saúde apresentadas através de autofalantes, seja por necessidade ou por opção. Porém, a maneira como essas informações são apresentadas pode ser incômoda para algumas pessoas. Do mesmo modo, uma TV digital, antes utilizada para entreter outros usuários, pode se tornar perturbadora a eles ou ainda causar algum desconforto ao paciente, uma vez que suas informações estarão sendo compartilhadas por outras pessoas, que a princípio não deveriam ter acesso àquela informação, como é o caso de visitantes.

A questão da privacidade é um dos principais pontos de sistemas *homecare*. Como o paciente tem total direito de visualizar suas informações, é necessário encontrar maneiras de manter isso em sigilo, a fim de que somente pessoas autorizadas consigam vê-las. De acordo com o número de usuários que desejam acessar ou controlar informações é necessário a criação de algumas regras sobre a utilização de tais informações.

Outro ponto importante é considerar a mudança no contexto em função das necessidades do paciente. Essas mudanças incluem alterações de condições médicas do paciente, novos dispositivos, circunstâncias familiares ou, simplesmente, mudanças de comportamento das pessoas.

Por fim, o cuidador também pode representar um problema para um sistema *homecare*. Este pode não possuir domínio da técnica de monitoramento ou ainda estar desatento em função do desgaste emocional e físico propiciado pelo intenso cuidado ao paciente (BASTIANI; SOARES; LIBRELOTTO, 2012).

Analisando os problemas apresentados, é possível identificar uma série de consequências geradas a sistemas *homecare*. A principal delas é que o sistema pode deixar de cumprir os benefícios desejados ou fazer isso a um preço inaceitável, comprometendo a aceitação do sistema e a prestação dos cuidados ao paciente.

3.4 Considerações do capítulo

Este capítulo trouxe conceitos relacionados a ambientes *homecare*. Foram descritas as entidades envolvidas em uma rede deste tipo de ambiente, além de características dos sistemas e os desafios encontrados para que tais sistemas sejam aceitos pelos usuários. Na seção seguinte, serão discutidas as demências, subgrupo de doenças abordado neste trabalho.

4 DEMÊNCIAS

A demência é uma síndrome caracterizada pelo declínio da memória associado a déficit de, pelo menos, uma outra função cognitiva com intensidade suficiente para interferir no desempenho social ou profissional do indivíduo. Os sintomas implicam em uma perda progressiva da capacidade de funcionamento cerebral e motor do paciente, diminuindo a capacidade de memória, linguagem, reconhecimento, orientação e, conseqüentemente, de realização de suas atividades de vida diárias (AVDs). No portador de demência, é possível também perceber perturbações comportamentais como agitação, inquietação, gritos, insônia, sintomas depressivos, etc (SCHLINDWEIN-ZANINI, 2005).

O número de pessoas com demência vem crescendo exponencialmente com o envelhecimento populacional, e a prevalência de demência dobra a cada cinco anos após os 65 anos, como pode ser observado na Tabela 4.1 (SMID, 2008). Ainda, em um estudo realizado com 1660 idosos, pessoas do sexo feminino representaram 8,7% da prevalência de demência, enquanto pacientes homens 4,8%. O analfabetismo e o nível socioeconômico também mostraram-se fatores importantes no aparecimento das síndromes demenciais. Dentre a população diagnosticada com demência, 12% eram pessoas analfabetas, e 10,9% pertenciam a classe social mais baixa (HERRERA; CARAMELLI; NITRINI, 1998).

Tabela 4.1: Prevalência de demência nas diferentes faixas etárias

Idade	Demência (%)
60–64	0,7
65–69	1,4
70–74	2,8
75–79	5,6
80–84	10,5
85–89	20,8
90–95	38,6

O tratamento de demências não possui um padrão específico. Em geral, os distúrbios cognitivos são tratados com medidas farmacológicas, através do uso de inibidores da acetilcolinesterase colinesterase, ou através de programas de reabilitação e cognitiva. Por sua vez, os sintomas neuropsiquiátricos são tratados através de antidepressivos e antipsicóticos ou de forma não farmacológica, com a inclusão de suporte social do paciente. Ainda, é importante estimular o paciente a fim de proteger as funções cognitivas remanescentes e aspectos funcionais. Dentre esses estímulos estão músicas, solicitações, tarefas, atividades físicas, etc . Assim, o tratamento

do paciente unindo estratégias medicamentosas, ou não, deve focar-se em três eixos básicos: diminuição da dependência funcional do paciente; retardo da deterioração cognitiva; e redução da sobrecarga do cuidador (ENGELHARDT et al., 2005).

Existem várias causas de demência, cujo diagnóstico específico depende de conhecimento das diferentes manifestações clínicas e de uma sequência específica de exames complementares. O diagnóstico sindrômico depende da avaliação do funcionamento cognitivo e do desempenho na realização em atividades de vida diária, como o mini exame do estado mental (FOLSTEIN; FOLSTEIN; MCHUGH, 1975). Ainda, é necessário o diagnóstico etiológico que baseia-se em exames laboratoriais e de neuroimagem, além da constatação de perfil neuropsicológico característico. Esse aspecto é particularmente importante para o diagnóstico diferencial das demências degenerativas.

As principais causas de demência são as doenças neurodegenerativas, responsáveis por 60 a 70% dos casos (TEIXEIRA-JR; CARDOSO, 2005). Por essas doenças respondem a doença de *Alzheimer*, a demência fronto-temporal e demência com corpos de *Lewy*. Entre as demências secundárias, a demência vascular é a mais relevante, sendo responsável por 10 a 20% dos casos. Outras demências secundárias incluem demências por doenças infecciosas, como demência por HIV e Neurosífilis, processos metabólicos e condições neurológicas adversas.

Essas causas são classificadas de acordo com a *International Classification of Diseases* (ICD), a ferramenta padrão da *World Health Organization* (WHO) para a epidemiologia, gestão da saúde e fins clínicos. A ICD fornece códigos para padronizar a classificação de doenças e de uma gama de sinais, sintomas, aspectos anormais, queixas, circunstâncias sociais e causas externas para ferimentos ou doenças. Nesta classificação, as demências, estão inseridas dentre os distúrbios mentais e comportamentais, com códigos que variam de *F00* até *F09* (WHO, 2012). A seguir são detalhadas as principais causas de demências, segundo (NETO; TAMELINI; FORLENZA, 2005).

4.1 Doença de *Alzheimer*

A doença de *Alzheimer* (DA) é considerada a principal causa de demência, responsável por mais de 50% dos casos de demência na faixa etária igual ou superior aos 65 anos de idade, sendo esse o principal fator de risco, e atualmente representa a sétima causa de morte nos Estados Unidos. Além da idade, os outros fatores de risco para a DA incluem histórico familiar, sexo feminino, traumatismo cranoencefálico e diminuição de acetilcolina (SMID, 2008).

Trata-se de uma doença crônico-degenerativa e irreversível que ocorre de forma insidiosa e com progressão lenta dos sintomas, podendo chegar a vinte anos de sobrevida, e pode ser classificada em três estágios de acordo com sua evolução. No estágio inicial podem aparecer problemas leves de memória, como memorizar novas informações e eventos recentes, e o paciente realiza suas atividades do dia-a-dia como dirigir e cuidar da casa de forma mais ineficiente e com menos interesse, porém com relativa independência. No estágio intermediário, os problemas de memória acentuam-se, com dificuldades de lembrar eventos recentes ou de longo tempo. Nesse estágio, associam-se déficits das outras funções cognitivas aos sintomas iniciais, como linguagem, praxia, atenção, funções executivas e habilidades visuoespaciais, tornando-se difícil interpretar os sentidos e as atividades diárias, como se vestir e alimentar-se, tornam-se complexas de serem realizadas sem ajuda. Ainda, as alterações do ciclo sono-vigília são comuns, podendo haver piora dos sintomas cognitivos e comportamentais ao final do dia. Sintomas psiquiátricos também são comuns, com frequente presença de delírios e paciente também pode apresentar alucinações, agitação, apatia e sintomas depressivos.

Por fim, o estágio mais avançado da doença é caracterizado pela demência grave, onde o paciente torna-se totalmente dependente de um cuidador. Há incontinência vesical e fecal, incapacidade de reconhecer os familiares, dificuldade em alimentar-se e locomover-se, perda das funções cognitivas e comportamentais, rigidez, e instabilidade à marcha (SMID, 2008).

4.2 Demência Vascular

A demência vascular ocorre secundariamente ao comprometimento vascular do sistema nervoso central e constitui a segunda maior causa de demência na população idosa, ocidental. Ao contrário da doença de *Alzheimer*, a demência vascular tem um início mais precoce e os homens são mais afetados que pessoas do sexo feminino (NETO; TAMELINI; FORLENZA, 2005).

Apesar da definição de demência priorizar comumente o de memória como comprometimento primário, em muitos casos de demência vascular os primeiros sintomas são o comprometimento das funções executivas ou focais múltiplos. Em uma comparação com a doença de *Alzheimer*, os pacientes portadores de demência vascular apresentaram taxas mais elevadas de depressão e comprometimento funcional.

Dentre os fatores de risco para demência vascular estão a hipertensão arterial, diabetes, tabagismo, alcoolismo, doenças cardíacas, obesidade, sexo masculino, raça negra e baixa escolaridade (CARAMELLI; BARBOSA, 2002).

A demência vascular pode ser dividida em demência por múltiplos infartos, demência isquêmica subcortical e demência por infarto estratégico. A primeira está ligada à acidentes vasculares cerebrais recorrentes. Mudanças de marcha, incontinência urinária, disfagia, disartria e labilidade emocional, flutuações cognitivas e piora noturna são sintomas comuns. Já a demência isquêmica subcortical é caracterizada pelo início lento e insidioso dos sintomas. O principal domínio cognitivo afetado é a disfunção executiva, que interfere no desempenho das atividades diárias do paciente. Alterações de humor com depressão e mudança de personalidade também ocorrem frequentemente.

Por fim, a demência por infarto estratégico é reconhecida por pequenos infartos nas áreas corticais ou estruturas subcorticais. Podem ocorrer confusão mental ou coma seguido de amnésia, apatia, abulia, depressão, falta de atenção e desinibição (ENGELHARDT et al., 2005).

4.3 Demência Fronto-Temporal

Estima-se que as demências frontotemporais (DFT) correspondam por 10% a 15% dos casos de demência degenerativa, ocorrendo após os quarenta anos de idade, com incidência igual em homens e mulheres. A memória e habilidades visioespaciais apresentam-se pouco comprometidas, enquanto as alterações comportamentais são mais intensas. Dentre estas alterações podem aparecer, de forma geral, isolamento social, apatia, irritabilidade, rigidez e inflexibilidade mental, hiperoralidade e descuido da higiene pessoal, sintomas depressivos, preocupações somáticas, bizarras e estereotipias motoras (CARAMELLI; BARBOSA, 2002).

Clinicamente, existem três diferentes síndromes que constituem a demência fronto-temporal. A forma mais comum de apresentação é demência frontotemporal do lobo frontal, caracterizada por progressivas mudanças no comportamento e de personalidade, onde os sintomas surgem precocemente. Estes sintomas podem ser desinibição, apatia, falta de higiene pessoal, alterações no comportamento alimentar e alterações da linguagem, desde discurso estereotipado até mudismo.

A segunda variante diz respeito as alterações da linguagem que caracteriza-se por um discurso espontâneo não-fluente associado à agratismo, parafasias fonêmicas ou anomia. Também há dificuldade de encontrar a palavra correta.

Por sua vez, o doente acometido de demência semântica, a terceira forma de manifestação da demência fronto-temporal, apresenta parafasias semânticas com dificuldade de compreensão e perda do significado das palavras. Habitualmente o paciente tem um discurso vazio mas

fluente. Além dessas características, eles podem apresentar dislexia, disgrafia e incapacidade de reconhecimento de objetos e faces familiares.

Ainda, é importante ressaltar que alguns pacientes desenvolvem síndromes de declínio motor que pode preceder, coincidir ou ocorrer secundariamente às alterações de comportamento. Tais déficits motores incluem a doença do neurônio motor, paralisia supranuclear progressiva e degeneração córtico-basal (GUIMARÃES; FONSECA; GARRETT, 2006).

4.4 Demência com Corpos de Lewy

A demência com corpos de Lewy (DCL), descrita em 1961 era considerada uma síndrome rara até 1980. Porém, o desenvolvimento de novas técnicas de diagnóstico permitiu o reconhecimento da demência com corpos de Lewy como uma importante causa de demência, acometendo 20% dos pacientes dementes (TEIXEIRA-JR; CARDOSO, 2005).

Embora muitos autores afirmem que a DCL seja uma variante da doença de *Alzheimer*, a conferência para consenso sobre DCL unificou seus critérios e definiu que a característica principal para diagnóstico da DCL é a associação do declínio cognitivo suficiente para interferir no funcionamento social e ocupacional do paciente com parkinsonismo espontâneo, com alucinações visuais recorrentes e/ou flutuação das funções cognitivas (TATSCH; NITRINI; NETO, 2002).

Outras características são quedas repetidas, síncope, perdas transitórias da consciência, sensibilidade aos neurolépticos, ilusões sistematizadas, transtornos do comportamento associado ao sono REM (*Rapid Eye Movement*) e depressão. Esses sintomas se manifestam de forma variável e podem ocorrer simultaneamente (TEIXEIRA-JR; CARDOSO, 2005).

4.5 Outras causas de demências

Além das causas de demência supracitadas, várias outras condições médicas estão associadas com a presença de sintomas demenciais, como distúrbios metabólicos, o alcoolismo e a utilização de certos medicamentos.

O grupo das doenças degenerativas primárias, por exemplo, completa-se com a doença de *Wilson*, a doença de *Huntington*, mal de *Parkinson*, atrofia de múltiplos sistemas, insônia familiar fatal e doença de *Gertsman-Sträussler-Scheinker*.

Existe ainda uma gama de demências classificadas como reversíveis. Tal grupo é composto por causas raras de demência, entretanto são necessárias do ponto de vista do diagnóstico, visto

que o tratamento pode conter o declínio cognitivo. É importante ressaltar que apenas uma pequena parte dos pacientes melhoram após o tratamento. Dentre as causas de demências reversíveis pode-se destacar o hipotireoidismo, deficiência de vitamina B12 e a depressão (FORNARI et al., 2010).

Além disso, algumas doenças infecciosas também são capazes de causar dano neurológico e acarretar sintomas demenciais. Atualmente, as causas mais importantes de demências por agentes infecciosos são a demência induzida por HIV e a doença de *Creutzfeldt-Jakob*. Além dessas doenças, pertencem a esse grupo a Neurosífilis, Neurocisticercose, Meningoencefalites e Encefalites víricas (NETO; TAMELINI; FORLENZA, 2005).

4.6 Considerações do capítulo

Neste capítulo foram detalhadas as principais causas de demências, bem como seus sintomas característicos, epidemiologia e o grau de dependência do paciente. Tais informações serão indispensáveis para o desenvolvimento da proposta do presente trabalho. A seção seguinte aborda conceitos e questões relacionadas a ontologias, como classificação, utilização e representação.

5 ONTOLOGIAS

Ontologia é um conceito utilizado há muito tempo pela Filosofia, definido como o estudo das coisas que existem. O termo original é a palavra aristotélica *categoria*, utilizada com o propósito de classificar e caracterizar qualquer entidade. Aristóteles ainda introduz o termo “*differentia*” para propriedades que distinguem diferentes espécies do mesmo gênero (SOWA, 1999).

Com uma interpretação um pouco diferente, a comunidade da computação também passou a adotá-lo, mais precisamente na área de Inteligência Artificial, onde ontologia refere-se a artefatos de engenharia, constituídos por um vocabulário específico usado para descrever certa realidade e um conjunto de decisões explícitas, de forma a fixar rigorosamente o significado pretendido para o vocabulário. Ela captura os conceitos e relações em determinado domínio e um conjunto de axiomas, que restringem a sua interpretação (GUARINO, 1998).

Atualmente, uma das definições mais citadas na área de computação, e que será adotada neste trabalho, é descrita por (BORST, 1997). Segundo o autor: “uma das ontologia pode ser vista como uma especificação formal e explícita de uma conceitualização compartilhada”. Nessa definição: *conceitualização* refere-se ao modelo abstrato de algum fenômeno do mundo o qual identifica conceitos relevantes ao próprio fenômeno; *formal* significa o fato de a ontologia ser legível por computadores; *especificação explícita* diz respeito a conceitos, propriedades, relações, funções, restrições, axiomas definidos explicitamente; *compartilhada* quer dizer conhecimento consensual, ou seja, uma ontologia obtém o conhecimento apresentado não somente por um indivíduo único, mas sim por um grupo.

De forma geral, ontologia é um modelo de dados que representa um conjunto de conceitos de um determinado domínio de informação, bem como os relacionamentos entre estes conceitos. A diferença entre ontologias e outros modelos de domínios é que o seu principal objetivo é focado nos conceitos e seus relacionamentos, no qual a semântica destes relacionamentos é aplicada uniformemente (LIBRELOTTO et al., 2008).

Uma ontologia, além de definir um vocabulário sobre determinado domínio, também possui relacionamentos e restrições entre os conceitos definidos no vocabulário. Esses relacionamentos são definidos uniformemente e a semântica é feita de forma detalhada. Logo, se os relacionamentos possuírem nomes apropriados, tanto um humano ao visualizar a ontologia entenderá seu significado de maneira clara, quanto um programa pode assumir a semântica de um

relacionamento e atuar sistematicamente através da mesma. Um exemplo de tipo de relacionamento básico é o hierárquico “é-um”. Existem especificações definidas como relacionamentos hierárquicos denominadas taxonomias. Ontologias também incluem relacionamentos não hierárquicos, como “tem interesse em” entre os conceitos *Pessoa* e *Interesse*, sem que se trate de um relacionamento hierárquico (LOPES, 2006).

As restrições de uma ontologias, servem para criar uma restrição para um conceito baseando-se em um relacionamento. Em uma ontologia sobre pessoas, por exemplo, pode-se criar uma restrição sobre conceito de pessoa baseada no relacionamento “tem nome”, onde “uma pessoa terá exatamente um nome”.

As ontologias não apresentam sempre a mesma estrutura, mas existem características e componentes básicos comuns presentes em grande parte delas (GRUBER, 1993). Dentre esses componentes, é possível citar três elementos, tidos como base, para a construção de uma ontologia: classes, indivíduos e propriedades. As *classes* são conceitos organizados em uma taxonomia e são as unidades básicas da ontologia. Definem um determinado objeto, como pessoa ou medicamento. Essas classes possuem instâncias ou objetos, os *indivíduos*, como o indivíduo João que pertence à classe Paciente. As propriedades de uma ontologia, por sua vez, dividem-se entre propriedades de dados e propriedades de objetos. O primeiro tipo refere-se as características que diferenciam os indivíduos de uma determinada classe, como “idade” ou “nome”. Já propriedades de objeto definem a maneira como os indivíduos e as classes se relacionam (e.g. paciente *possui* sintomas).

É importante ressaltar que o processo de construção de ontologias não é uma tarefa trivial, sendo necessário o conhecimento detalhado sobre o domínio em questão para que não haja nenhum tipo de ambiguidade ou contestação quanto a sua validade.

5.1 Motivação para o uso de ontologias

Pode-se afirmar que ontologias são aplicadas para proporcionar a comunicação entre diferentes pessoas e aplicações que fazem parte de um domínio do conhecimento em comum, mas que por vezes não compartilham de uma mesma conceituação a respeito dos componentes deste domínio. Essa falta de entendimento compartilhado pode desencadear problemas na interoperabilidade e possibilidade de reuso e compartilhamento de conhecimento, o que é muito importante em função da grande variedade de métodos, paradigmas, linguagens e ferramentas existentes na área de computação. Logo, devido a natureza formal de uma ontologia, os

conflitos conceituais e terminológicos são reduzidos (LOPES, 2006).

De acordo com o trabalho de (GUIMARÃES, 2002), as principais vantagens na utilização de ontologias são:

- Ontologias fornecem um vocabulário comum para representação do conhecimento, o qual tem uma conceitualização bem definida, evitando assim interpretações ambíguas desse vocabulário.
- Ontologias permitem o compartilhamento de conhecimento, ou seja, caso exista uma ontologia que modele adequadamente certo domínio de conhecimento, essa pode ser compartilhada e usada por pessoas que desenvolvam aplicações dentro desse domínio. Por exemplo, considerando que existe uma ontologia para o domínio de livrarias, várias livrarias podem construir seus catálogos utilizando o vocabulário fornecido por essa ontologia sem a necessidade de refazer uma análise deste domínio.
- Fornece uma descrição exata do conhecimento. Diferentemente da linguagem natural em que as palavras podem ter semântica totalmente diferente conforme o seu contexto, a ontologia por ser escrita em linguagem formal, não deixa espaço para o *gap* semântico existente na linguagem natural. Por exemplo, quando uma pessoa fala para outra a palavra Globo, ela pode estar querendo falar a respeito de um corpo esférico, como também de um canal de televisão brasileiro. A interpretação da palavra pode ser atribuída a um conceito ou outro conforme o estado mental do indivíduo. Porém, se há uma conceitualização comum entre essas duas pessoas, a possibilidade de mal entendido diminui consideravelmente. Por exemplo, se essas pessoas concordam em uma ontologia sobre o domínio de formas geométricas, possivelmente não haverá mal entendido.
- É possível fazer o mapeamento da linguagem da ontologia sem que com isso seja alterada a sua conceitualização, ou seja, uma mesma conceitualização pode ser expressa em várias línguas.
- Pode-se estender o uso de uma ontologia genérica de forma que ela se adeque a um domínio específico. Por exemplo, se alguém precisa de uma ontologia sobre bicicletas para construir uma aplicação e só encontra uma ontologia sobre o domínio genérico de veículos, pode utilizar essa ontologia estendendo-a para o domínio específico da aplicação, que no caso são de bicicletas.

A partir das vantagens citadas percebe-se que o uso de ontologias é uma maneira adequada para representação de conhecimento, uma vez que há necessidade de confiabilidade com relação aos conceitos do vocabulário ou linguagem utilizada em certo ambiente. A representação formal que se adquire com seu uso pode tornar possível a automação da checagem de consistência, o que implicará em ambientes mais confiáveis (GRUNINGER, 1996).

5.2 Classificação das ontologias

Ontologias podem ser classificadas em diferentes tipos, os quais variam de acordo com o autor que os propõem. Entre essas classificações, (STUDER R.; BENJAMINS, 1998) descrevem quatro tipos:

- ontologias de domínio: utilizadas para representar o conhecimento válido para um tipo particular de domínio, como mecânica, medicina, biologia, etc. Expressam o vocabulário relativo de um domínio específico, descrevendo situações reais deste domínio;
- ontologias genéricas: são similares às ontologias de domínio, porém os conceitos definidos neste tipo de ontologia são considerados em diversas áreas. Estes conceitos são tipicamente gerais como espaço, tempo, etc, os quais são independentes de um problema particular ou domínio. Conceitos em ontologias de domínio são frequentemente definidos como especializações de conceitos de ontologias genéricas;
- ontologias de aplicação: compreende os conceitos necessários para modelar o conhecimento requerido por uma aplicação específica. Esses conceitos correspondem, frequentemente, aos papéis desempenhados por entidades do domínio enquanto executam determinada atividade;
- ontologias de representação: este tipo de ontologia não se compromete com nenhum domínio em particular e determinam entidades representacionais sem especificar o que deve ser representado. Ontologias de domínio e genéricas descrevem seus domínios a partir de primitivas fornecidas pelas ontologias de representação.

Guarino (1998) descreve outro tipo de ontologias chamadas “ontologias de tarefas”, as quais descrevem um vocabulário relacionado a uma tarefa ou atividade genérica, através da especialização dos termos introduzidos pelas ontologias genéricas.

5.3 Representação de ontologias

Ontologias podem ser escritas de diferentes maneiras, inclusive em formato de texto. Entretanto, para representar conceitos compartilhados sobre determinado domínio, é necessária uma linguagem de representação conhecida. Atualmente existem várias maneiras de representação de ontologias baseadas em, basicamente, dois grupos: Lógica de Primeira Ordem e linguagens baseadas em XML. O primeiro grupo refere-se a linguagens utilizadas para representação do conhecimento, enquanto o segundo é usado em ambientes *Web*, tendo impacto principal sobre aplicações da *web* semântica, onde as informações são apresentadas a partir de significados bem definidos, possibilitando, assim, que pessoas e computadores cooperem mais facilmente (LOPES, 2006).

A linguagem XML (BRAY et al., 2008) é considerada a linguagem padrão para troca de informações na web, logo é desejável também a utilização de sua sintaxe para modelar ontologias. Partindo desse pressuposto, diversas linguagens foram propostas, as quais diferem nas suas expressividades e propriedades computacionais.

5.3.1 XML

XML é uma metalinguagem que define uma sintaxe para ser utilizada na criação de outras linguagens de marcação para um domínio específico, com estrutura e semântica próprias. Pode também ser definida como uma linguagem de meta-marcação que, como a HTML (*Hipertext Markup Language*), inclui dados entre marcadores. Entretanto, elas possuem propósitos diferentes: HTML é uma linguagem para apresentação, enquanto XML é uma linguagem para descrição de dados estruturados, ou seja, os marcadores XML estão relacionados ao significado do texto delimitado por eles ao passo que os marcadores HTML especificam como os dados serão apresentados. Devido aos marcadores XML indicarem o conteúdo e a estrutura dos dados, é possível construir documentos legíveis por seres humanos e que podem ser facilmente processados por máquinas (LIMA; CARVALHO, 2005a).

Duas características relevantes de XML são a independência de dados e separação entre conteúdo e apresentação, podendo ser perfeitamente processado por uma aplicação. Além disso, XML permite a criação de *tags* personalizadas, possibilitando ao autor construir o modo como as informações serão estruturalmente representadas no documento. Outra vantagem é que XML é baseada em texto, assim qualquer pessoa pode criar um documento a partir de uma ferramenta simples.

Outras vantagens proporcionadas pela linguagem XML incluem informação semântica, independência de plataforma, facilidade de geração de visões diferentes de dados, simplicidade de leitura por pessoas e máquinas e facilidade de compartilhamento de dados entre aplicações.

5.3.2 RDF

A linguagem RDF é a base para processamento de metadados promovem a interoperabilidade entre aplicações que trocam informações interpretáveis por máquinas na *web*. Essa linguagem busca facilitar o processamento autônomo de recursos da *web*, tornando possível a especificação de semânticas de dados baseadas em XML, de forma padronizada e interoperável (LASSILA; SWICK, 1999).

O objetivo geral do RDF é definir um mecanismo para descrever recursos um mecanismo para descrição de recursos que não façam suposições a respeito de um domínio de aplicação específico e que não defina as semânticas de nenhum domínio de aplicação. A definição deste mecanismo deve ser feita de forma neutra, ou seja, não pode ser específica para um determinado domínio, mas sim, deve ser aplicável para descrever informações sobre qualquer domínio de aplicação.

O modelo básico de dados para criação de um documento RDF consiste em três tipos de objetos: um recurso, uma propriedade e uma afirmação. O recurso é tudo o que está sendo descrito através de expressões RDF. A propriedade é um aspecto específico, característica, atributo ou relação usado para descrever um recurso. Cada propriedade tem um significado específico, define seus valores permitidos, os tipos de recursos que podem descrever, e sua relação com outras propriedades.

Um recurso específico, junto com uma propriedade e o valor dessa propriedade caracteriza uma afirmação RDF. Essas três partes individuais de uma afirmação que são chamados, respectivamente, o sujeito, o predicado e objeto. O objeto de uma declaração pode ser um outro recurso ou pode ser um literal, ou seja, um recurso (especificado por um URI) ou uma simples string ou outro tipo de dados primitivo definido por XML.

A linguagem RDF é orientada a objetos, onde uma coleção de classes é chamada de *schema*. Estas classes são organizadas em uma hierarquia, provendo extensibilidade através das subclasses.

5.3.3 RDF (S)

A linguagem RDF Schema – RDF(S) – é uma extensão da linguagem RDF que permite a definição de vocabulários formados por propriedades e classes específicas de um domínio, onde as classes são organizadas em uma hierarquia, provendo extensibilidade através das subclasses definidas. Ainda, a linguagem RDFS fornece mecanismos para descrever grupos de recursos relacionados e os relacionamentos entre tais recursos.

Pode-se afirmar que, na semântica RDFS, uma classe corresponde a um tipo ou categoria, onde recursos pertencentes à uma classe pode ser instância de uma ou mais classes. As classes no RDFS também podem ser divididas em subclasses, herdando as propriedades de sua superclasse. As propriedades, por sua vez, são utilizadas para caracterizar uma instância ou relacioná-la com outra instância. Além disso, o RDFS permite criar restrições para as propriedades como, por exemplo, os elementos `rdfs:domain` e `rdfs:range` que são usados para dizer a quem esta propriedade se aplica (FREITAS, 2011).

Ainda, a RDF(S) provê informações sobre a interpretação dos enunciados apresentados em um modelo de dados RDF, o que difere do propósito das DTDs (BRAY et al., 2008) pertencentes a linguagem XML, as quais fornecem restrições específicas a estrutura de um documento XML (LOPES, 2006).

5.3.4 OWL

A linguagem OWL, uma recomendação da W3C para web semântica, tem sua origem no DAML + OIL (HORROCKS et al., 2001). Apesar de ser baseada em RDF e RDFS e utilizar a sintaxe de XML, a OWL é uma linguagem mais rica que as demais, e tem a vantagem de ter mais facilidade para expressar significados e semântica. Ela supre algumas limitações das outras linguagens, como restrições, disjunções de classes, restrições de cardinalidade, entre outras (ANTONIOU; HARMELEN, 2012).

A OWL 1.0 é dividida em três sub-linguagens, distintas pelo nível de formalidade exigido e oferecido e a liberdade dada ao usuário para a definição de ontologias: OWL-Lite, OWL-DL e OWL-Full (LIMA; CARVALHO, 2005b).

A OWL-Lite é voltada para usuários que necessitam de uma classificação hierárquica com restrições simples. Tem como vantagem ser mais facilmente entendida e ter uma implementação razoavelmente fácil. Porém, suporta apenas classificações hierárquicas e restrições simples, como restrições de cardinalidade, onde são permitidos apenas valores de 0 ou 1, limitando sua

expressividade.

Já a OWL-DL provê um maior grau de expressividade onde todas as conclusões computáveis e todas as computações terminam em tempo finito. É relacionada com a lógica descritiva. Ela inclui todos os construtores da OWL, mas eles só podem ser usados sob certas restrições.

Por fim, a OWL-Full permite que os usuários utilizem a máxima expressividade e total liberdade de sintaxe do RDF. Apesar de ser mais expressiva que as demais sublinguagens, a OWL-Full não possui garantia computacional e tem a desvantagem de ter um custo de decisão mais alto, pois dá mais liberdade para o programador desenvolver o código, o que pode torná-lo bastante complexo ao ser processado pelo computador. Esta linguagem também permite unir OWL com RDFS e não requer disjunção de classes, propriedades, indivíduos e valores de dados, ou seja, uma ontologia criada em OWL Full pode possuir uma classe que é ao mesmo tempo classe e indivíduo.

Cada uma das sub-linguagens é uma extensão de sua predecessora. Isto significa que, por exemplo, uma ontologia válida em OWL Lite também será válida em OWL DL, mas nem toda ontologia válida em OWL DL será válida em OWL Lite.

5.3.5 OWL 2

Apesar da OWL ter sido bem sucedida, com o crescimento de usuários que passaram a utilizá-la, alguns problemas foram sendo identificados. Isoladamente nenhum desses problemas são considerados graves, porém, em conjunto indicaram a necessidade de uma nova versão da OWL, a OWL 2 (GRAU et al., 2008).

A OWL 2 traz também um suporte estendido a novos tipos de valores de dados, proporcionando maior expressividade. Além disso, foram introduzidos três perfis, ou sub-linguagens que oferecem importantes vantagens em cenários de aplicações específicas. Cada perfil é definido como uma restrição sintática da especificação estrutural da OWL 2, isto é, como um subconjunto de elementos estruturais que podem ser usados em uma ontologia.

O perfil OWL 2 EL captura o poder expressivo usado em ontologias de grande escala, e é voltado para ontologias grandes, mas simples, e com bom desempenho no tempo de inferência sobre ela. Por sua vez, o OWL 2 QL captura a expressividade usada normalmente em ontologias simples e fornece as características necessárias para capturar os modelos conceituais, como diagramas de classes UML, diagramas ER e *database schemas*. Este perfil é mais adequado para aqueles que necessitam uma fácil comunicação com bancos de dados relacionais e onde o ra-

ciocínio sobre grandes conjuntos de dados é o mais importante. Por último, o OWL 2 RL foi projetado para acomodar aplicações OWL 2 que necessitam usar toda a expressividade da linguagem com mais eficiência e em aplicações RDF(S) que precisam adicionar a expressividade da OWL 2.

Os três perfis possuem restrições sintáticas sobre determinados construtores e axiomas. As versões QL e RL são adequados para aplicações onde ontologias relativamente leves são usadas com grandes conjuntos de dados. A escolha entre eles vai depender dos tipos de dados a serem processados. Quando for necessário acessar dados diretamente através de consultas SQL, o perfil OWL 2 QL é o mais indicado. Por sua vez, o OWL 2 RL é voltado para quem precisa operar sobre dados na forma de triplas RDF (?).

5.4 Inferências sobre ontologias

A concepção atual de ontologias é voltada para representação do conhecimento, onde por meio de classes e propriedades podemos descrever um domínio. Porém, a expressividade semântica não está na capacidade de classificar, mas sim na possibilidade de inferência sobre seu próprio conteúdo. Desta forma, para aumentar o poder de expressividade das ontologias podem ser utilizadas linguagens que permitam execução de regras de inferência sobre a ontologia.

5.4.1 SWRL

A linguagem OWL, utilizada neste trabalho, permite a escrita de predicados unitários (pertencer a uma classe) e binários (relações), no entanto, apenas pode-se inferir predicados unitários a partir de seu axioma. Ou seja, OWL só permite a inferência para classificação.

Esta limitação é bastante crítica, ao ponto de não ser possível representar, por exemplo a relação *possuiPrimo* a partir das relações *possuiPai*, *possuiIrmao* e *possuiFilho* de uma forma genérica. Nessa descrição apenas é possível relacionar dois indivíduos específicos da classe pessoa. Pode-se dizer que João é filho de Pedro e Maria é filha de Alberto onde Alberto é irmão Pedro e Maria e João são primos, mas não dizer que todos os filhos de Alberto e Pedro são primos. Para resolver essas limitações, pode-se utilizar a linguagem SWRL (HORROCKS et al., 2004).

A linguagem SWRL possui alguns tipos regras que podem ser utilizados em conjunto com o conhecimento expresso nas ontologias, aumentando seu poder de expressividade. A partir destas regras, é possível inferir novos conhecimentos sobre a ontologia.

Ao executar uma regra, seu resultado irá afetar diretamente a estrutura da ontologia, ou seja, toda vez que as restrições presentes em uma regra forem aceitas, um novo conhecimento sobre o domínio será aplicado à ontologia. Assim, para uma representação mais poderosa, uma provável solução seria a utilização da combinação ontologias OWL + regras SWRL.

5.4.2 SQWRL

A linguagem SWRL também possui uma linguagem para consulta chamada SQWRL (*Semantic Query-Enhanced Web Rule Language*) (O'CONNOR; DAS, 2008) que permite a criação de consultas a ontologia de forma análoga ao SQL. Diferentemente das regras SWRL, que aplicam o resultado diretamente na ontologia, as consultas SQWRL apenas retornam as informações que satisfazem a restrição descrita e, então, esses resultados podem ser tratados utilizando uma linguagem de programação qualquer.

O SQWRL utiliza uma biblioteca com os métodos do SWRL, podendo utilizar-se destas regras para fazer buscas em OWL. O operador central de SQWRL é *sqwrl:select* e é necessário um ou mais argumentos, tipicamente associados a variáveis definidas na query. O código a seguir apresenta a sintaxe utilizada por consultas SQWRL:

```
Classe(?c) ^ propriedade(?c, ?v) -> sqwrl:select
```

Pelo fato de poder trabalhar usando as regras SWRL, o SQWRL torna-se uma poderosa linguagem para buscas em OWL, podendo aproveitar ao máximo a expressividade semântica representada em OWL.

5.5 Considerações do capítulo

Este capítulo apresenta a última etapa da revisão bibliográfica, cujo objetivo foi introduzir os assuntos utilizados na construção deste trabalho. Foram apresentados, no presente capítulo, conceitos sobre ontologias, formas de utilização e classificação. Além disso, as linguagens mais relevantes para construção de ontologias, as regras de inferência, que aumentam o poder de expressividade das mesmas, e as consultas sobre ontologias também foram descritas.

No capítulo seguinte, é apresentada uma ontologia que representará um ambiente *homecare* pervasivo voltado ao cuidado de pacientes portadores de demência, a primeira diz respeito à primeira parte da metodologia adotada.

6 MODELAGEM DE UMA ONTOLOGIA PARA REPRESENTAÇÃO DO CONHECIMENTO DE UM AMBIENTE HOMECARE PERVASIVO VOLTADO A PORTADORES DE DEMÊNCIA

Este capítulo apresenta a ontologia modelada para representar um ambiente *homecare* voltado a pessoas idosas acometidas de síndromes demenciais. Para o desenvolvimento da ontologia optou-se pela linguagem OWL-DL (Seção 5.3.4), utilizando a ferramenta Protégé (JUPP et al., 2007), uma das mais utilizadas para criação e edição de ontologias. O Protégé possui uma série de vantagens em relação a outras ferramentas, como importação e exportação de ontologias em diferentes formatos, facilitando o reuso, além de possuir diversos *plugins* para visualização e manipulação das informações.

A construção da ontologia se deu através dos seguintes passos, baseados na metodologia proposta por Noy e McGuinness (2001): (a) definição do domínio e escopo que a ontologia irá cobrir; (b) criação e organização hierárquica das classes presentes na ontologia; (c) determinação das propriedades de dados, as quais têm como objetivo caracterizar suas entidades, bem como as propriedades de objetos, cujo papel é estabelecer relações entre as classes; (d) estabelecimento das restrições dessas propriedades; e (e) criação das instâncias das classes. A seguir, tais etapas serão detalhadas.

6.1 Domínio e escopo da ontologia

No contexto deste trabalho, a ontologia tem como proposta mapear o conhecimento existente no âmbito do tratamento de pacientes acometidos de síndrome demencial em ambientes domiciliares de forma que se represente as relações existentes entre médicos, cuidadores, pacientes e as atividades que os pacientes possam realizar durante seus tratamentos, sejam elas recomendações médicas ou atividades de vida diárias. Além de cuidados envolvendo o paciente, a ontologia representa informações do ambiente em si, como cômodos e equipamentos que sejam utilizados no tratamento.

Como dito na Seção 5.1, no processo de construção de ontologias, uma das vantagens diz respeito ao reuso de informações, ou seja, reutilizar conceituações previamente estabelecidas por outras ontologias. Considerando isso, conceitos abordados no trabalho de Freitas *et al.* (2012), o qual apresenta uma metodologia para assistir pacientes em ambientes *homecare* pervasivos. O trabalho traz uma arquitetura que utiliza uma ontologia para representar de forma

genérica um ambiente *homecare* pervasivo. Logo, é possível que a ontologia seja estendida para domínios mais específicos, como é o caso desta dissertação.

Além da revisão bibliográfica já descrita, utilizaram-se informações obtidas através de reuniões, por meio de entrevistas, com profissionais da saúde envolvidos no tratamento de pacientes com esse perfil que trabalham em lares geriátricos, dentre eles médicos e enfermeiros. A partir das reuniões, pôde-se esclarecer tanto como é realizado o fluxo de tratamento de pacientes com demência, quanto o levantar o vocabulário específico da área da saúde, diretrizes médicas para o controle da saúde do paciente (e.g. limites aceitáveis para um determinado sinal vital) e as funções desempenhadas por cada profissional envolvido.

Segundo os profissionais, a diferença entre o tratamento de pacientes residentes em instituições e em suas casas se restringe basicamente no que diz respeito a autonomia desses pacientes, visto que terá de se adaptar aos padrões da instituição. Assim, o trabalho realizado pelos profissionais se assemelha quanto ao local de tratamento.

Com essas informações, oriundas das reuniões e da revisão bibliográfica, tornou-se possível a representação do ambiente domiciliar em questão, a qual contém as entidades, atributos e relacionamentos. Ainda, as informações relativas à saúde, como doenças e sintomas, foram mapeados de acordo com a *International Classification of Diseases* (ICD). Desta forma, utilizando informações padronizadas, diversas aplicações são capazes de entender e utilizar o conhecimento sobre saúde representado na ontologia.

6.2 Classes

A proposta de Freitas et al. (2012) descreve diversas classes que representam as entidades envolvidas em ambientes de *homecare* genéricos, as quais podem ser visualizadas na Figura 6.1. Esse conjunto de classes foi definido pois pode, de alguma forma, influenciar o fluxo de tratamento genérico de pacientes.

No entanto, considerando o objetivo principal desta dissertação uma abordagem para auxiliar no tratamento de pacientes idosos portadores de demências, os conceitos abordados nessa ontologia são insuficientes para a representação do conhecimento. Essa questão pode ser solucionada estendendo a ontologia, ou seja, inserindo novos termos e reutilizando o que já existe.

Assim, a partir do conhecimento adquirido junto aos profissionais e à pesquisa na literatura, chegou-se a um conjunto específico de classes para o domínio em questão, as quais podem ser agregadas na ontologia de Freitas. Além das entidades relacionadas em torno do monitoramento



Figura 6.1: Hierarquia de classes da ontologia proposta por Freitas *et al.* (2012)

do paciente, como sinais vitais, a ontologia estendida representa classes do ambiente em geral, como os equipamentos, além das atividades realizadas pelo paciente.

A Figura 6.2 apresenta a ontologia que representa um ambiente *homecare* pervasivo voltado a portadores de demências, com as classes acrescentadas na ontologia.

Sobre as classes criadas, é importante frisar algumas informações. Nem todas possuem instâncias, visto que foram definidas como classes abstratas e serão utilizadas para organização da hierarquia ontológica, e assim nenhum indivíduo poderá pertencer somente a elas. Um exemplo é a classe *Pessoa*. Qualquer pessoa que venha a fazer parte da rede de *homecare* deverá ser classificada como sendo *Paciente*, *Profissional* ou *Acompanhante*, subclasses de *Pessoa*. Por sua vez, *Profissional*, também é uma classe abstrata, visto que um profissional da saúde, dentro do ambiente, pode ser *Médico*, *Enfermeiro* ou *Técnico de Enfermagem*. E, da mesma forma, *Acompanhante* possui as subclasses *Cuidador* e *Visitante*.

Outra classe abstrata relevante é a classe *Demência*. Optou-se por definir *Demência* como subclasse de *Doença*, permitindo assim que a ontologia seja estendida futuramente para outras doenças. Suas subclasses englobam os tipos de demência, como *TipoAlzheimer*, *TipoVascular*, *TipoMista*, *TipoLewy* e *OutrasDemencias*. As demais superclasses da ontologia são *Ambiente*, *AtividadePaciente*, *Deficiencia*, *Equipamento*, *Exame*, *Notificacao*, *Sensor*, *Sindrome*, *Sintoma* e *Software*.

Ao contrário de classes abstratas, as classes são ditas concretas se possuírem uma ou mais instâncias, por exemplo, a classe *Móvel*, a qual engloba todos os móveis que possam estar presentes no ambiente de *homecare*. A partir da definição de superclasses e subclasses é criado

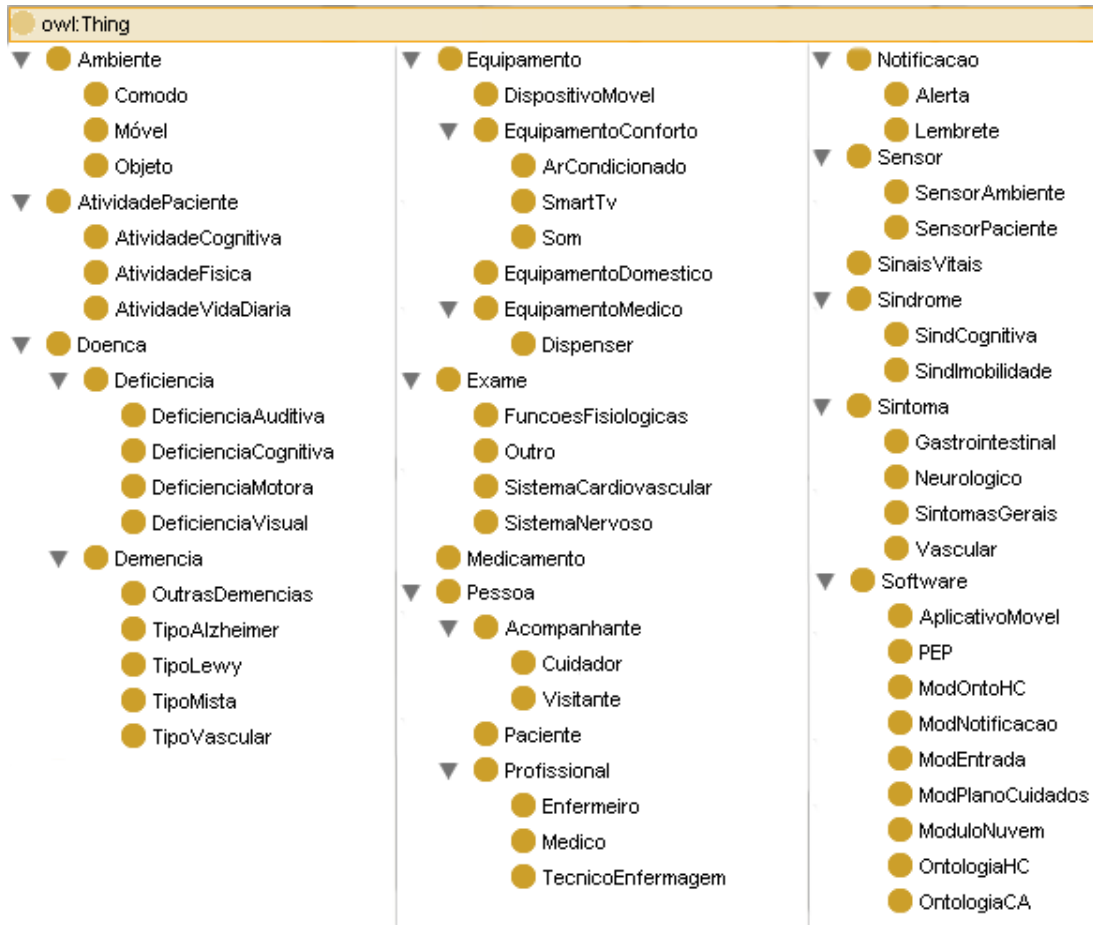


Figura 6.2: Hierarquia da ontologia de *homecare* voltado a portadores de demências

um sistema hierárquico na ontologia.

No entanto, as classes por si só não conseguem fornecer informações suficientes para o entendimento da ontologia. Logo, algumas propriedades são utilizadas para resolver este problema. As propriedades em uma ontologia podem ser de dois tipos: propriedades de dados, as quais são responsáveis por caracterizar as instâncias de cada classe; e propriedades de objeto que definem os relacionamentos existentes entre os indivíduos de cada classe. Na hierarquia da ontologia as subclasses herdam automaticamente as propriedades definidas na sua classe mãe. As propriedades de dados e de objetos serão descritas a seguir.

6.3 Propriedades de dados

As propriedades de dados de uma classe são as características utilizadas para descrevê-la, diferenciando seus indivíduos. A classe *Pessoa*, por exemplo, possui as propriedades de dados *nome*, *identificador*, *CPF*, *data_nascimento*, *sexo* e *nivel_escolaridade*. Logo, suas subclasses também possuem essas propriedades. Por exemplo, a propriedade de dado *nome* representa o

nome de cada indivíduo que pertence à alguma subclasses de *Pessoa*, como *Médico* ou *Enfermeiro*.

Sobre o ambiente, algumas propriedades de dados foram definidas para representar o contexto. Nesse sentido, a classe *Cômodo*, por exemplo, tem as propriedades *luminosidade*, a qual recebe valores entre 0 e 100 (ordem crescente de luminosidade); *temperatura*, representando a temperatura atual em graus *Celsius* do cômodo em questão; e *umidade*, a qual indica a umidade no ambiente com escala de 0 a 100.

As atividades que o paciente realiza também merecem destaque. Para a classe *AtividadeCognitiva* foi criada a propriedade *tipoAtividadeCognitiva*, a qual pode receber os valores *memória*, *raciocínio*, *atenção* ou *linguagem*. Já a classe *AtividadeFisica* tem as propriedades *tipoAtividadeFisica*, *intensidade* e *necessitaAuxilio*. A primeira indica o tipo de exercício físico que será realizado, podendo receber os valores *caminhar*, *alongar*, *correr* ou *outro*; o segundo atributo indica a intensidade do exercício, o qual pode ser de intensidade *baixa*, *moderada* ou *alta*, o que pode influenciar os sinais vitais do paciente, como frequência cardíaca; enquanto a propriedade de dado *necessitaAuxilio* informa se o paciente precisa, ou não, de ajuda para realizar um exercício. Por fim, a classe *AVD* representa as atividades de vida diárias, e possui a propriedade de dado *tipoAVD*, a qual caracteriza o tipo de atividade diária que o paciente realiza, por exemplo, *comer*, *dormir* e *sentar*.

Ainda, essas classes herdam os atributos definidos na sua classe mãe, *AtividadePaciente*, são eles: *inicioAtividade* que representa o horário em que atividade começou; *status* (*true* ou *false* que indica se o paciente está realizando a atividade ou não; e *terminoAtividade*, a qual marca o horário em que a atividade foi encerrada.

As demais propriedades de dados da ontologia são apresentadas na Tabela A.1 disponível no Apêndice A, com exceção à propriedade *identificador* e *descrição*, presentes em todas as classes. Juntamente com as propriedades, o apêndice apresenta a classe a que eles pertencem, o tipo de dado e os valores que podem assumir.

6.4 Propriedades de objetos

As propriedades de objetos de uma ontologia tem o papel de relacionar indivíduos entre as classes. Para exemplificar o uso de relações na ontologia, supõe-se a situação onde o paciente apresenta um sintoma (por exemplo, agitação) e necessita tomar algum medicamento. Na ontologia existe a relação entre a classe *Medicamento* e a classe *Sintoma*, através da propriedade

combate, indicando que um medicamento é indicado para um sintoma em específico. Ainda, é considerado que no ambiente *homecare* pervasivo cada *Medicamento* pode estar em um *Dispenser* (*Medicamento estaEm Dispenser*). Nesse cenário, um sistema de *homecare* pervasivo pode apresentar ao paciente, ou cuidador, através de uma tela onde está o *dispenser* mais próximo que contém o medicamento indicado para aquela situação. Para melhor entendimento, a Figura 6.3 apresenta um grafo gerado pela ferramenta Protégé, o qual contém os relacionamentos entre as classes presentes neste exemplo.

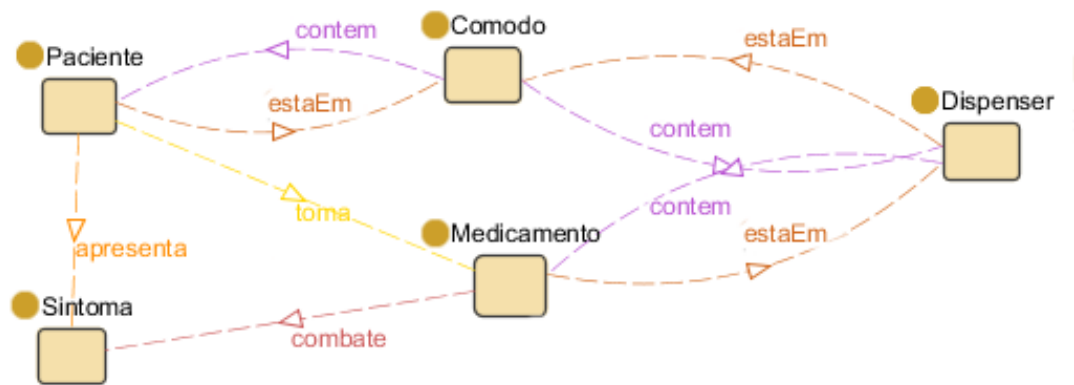


Figura 6.3: Exemplo de relacionamento na ontologia

Analisando o grafo acima é possível perceber que alguns relacionamentos na ontologia possuem uma relação inversa, por exemplo *Dispenser contem Medicamento* ao passo que *Medicamento estaEm Dispenser*. Esse tipo de relacionamento aumenta ainda mais a eficiência da ontologia, visto que a sua abrangência torna-se maior. A Tabela B.1, encontrada no Apêndice B, apresenta a relação completa de propriedades de objetos.

Pela expressividade da linguagem, algumas propriedades podem ser utilizadas por mais de uma classe, como é o caso do relacionamento *estaEm*. Ao mesmo tempo em que essa propriedade indica que uma pessoa está em um cômodo do ambiente (*Pessoa estaEm Comodo*), ela também indica, dentro do conhecimento representado, que um medicamento está em um determinado *dispenser* de medicamentos (*Medicamento estaEm Dispenser*), o que pode gerar ambiguidades na ontologia (*Paciente estaEm Dispenser*, *Medicamento estaEm Comodo*, etc). Para evitar incoerências desse tipo, é necessário impor restrições nas para o uso das propriedades da ontologia.

6.5 Restrições

As restrições têm a função de limitar os indivíduos que pertencerão a determinadas classes. Logo, é possível criar classes abstratas que irão englobar diversos indivíduos que satisfazem determinada restrição. Em OWL existem três tipos de restrições de propriedades: restrições de quantificador, restrições de cardinalidade e restrições de valor (HORRIDGE et al., 2004). O primeiro tipo é usado quando os indivíduos de uma classe devem possuir pelo menos um valor (*someValuesFrom*), ou somente valores de determinado conjunto (*onlyValuesFrom*). Um exemplo pode ser visto no relacionamento entre as classes *Paciente* e *Sintoma* através da propriedade *apresentaSintoma*, onde é dito que um paciente deve apresentar pelo menos um sintoma.

Por sua vez, as restrições de cardinalidade são utilizadas para restringir a quantidade de valores que uma propriedade pode assumir. Por exemplo, uma pessoa deve estar em um cômodo dentro da casa, ou seja a cardinalidade definida é exatamente 1 *cardinality*. Ainda podem existir restrições de cardinalidade mínima *minCardinality* e máxima *maxCardinality*.

Por fim, as restrições de valor descrevem classes anônimas de indivíduos que estão relacionados a outros indivíduos específicos por uma propriedade. Trata-se de uma situação diferente da restrição de quantificador, onde os indivíduos descritos pela restrição estão relacionados a qualquer indivíduo de uma classe específica, através da propriedade específica. Por exemplo, quando tem-se na ontologia uma instância de pessoa e várias instâncias de cômodo, e diz-se que uma pessoa em específico está em um cômodo em específico *Paulo estaEm Quarto*.

A Tabela C.1, encontrada no Apêndice C, apresenta as demais restrições presentes na ontologia.

6.6 Instâncias

O último passo da metodologia adotada consiste na criação das instâncias. As instâncias em uma ontologia representam os objetos do domínio em questão. Cada instância dentro da ontologia deve pertencer a uma determinada classe, e os indivíduos do mesmo tipo devem ser colocados em uma classe em comum. Por exemplo, se existem dois medicamentos, mesmo que indicados para sintomas diferentes e cada um possua um nome, eles serão agrupados na mesma classe, neste caso a classe *Medicamento*.

Na sua maioria, as classes da ontologia são instanciadas por informações oriundas de sensores, prontuário eletrônico do paciente e dispositivos computacionais. À exceção disso, as classes *Doença*, *Sintoma* e *Deficiência* foram preenchidas com instâncias de acordo com a *In-*

ternational Classification of Diseases. Assim, quando o paciente apresentar um determinado sintoma, o sistema pervasivo busca essas informações na ontologia e associa ao paciente, através da propriedade *apresentaSintoma*.

6.7 Consultas

Em ontologias, as consultas são utilizadas para obter dados relevantes ao domínio que está sendo representado. As consultas foram criadas através da linguagem SQWRL (Seção 5.4.2) e serão executadas pelo motor de inferência sempre que as condições definidas para elas forem satisfeitas. Para exemplificar, supõe-se a situação em que um sistema pervasivo, desenvolvido a partir da modelagem proposta, deva monitorar os sinais vitais do paciente ao invés de um cuidador fazer isso. Nesse caso, o sistema poderia executar uma consulta semelhante à consulta a seguir a fim de buscar informações sobre os sinais vitais, por exemplo, temperatura de um paciente:

```
Paciente(?p) ^ possui(?p, ?sinais) ^ SinaisVitais(?sinais) ^
temperatura(?sinais, ?temp) -> sqwrl:select(?temp)
```

Com o resultado desta consulta, o sistema pode tomar uma decisão de sugerir um medicamento ou notificar o cuidador caso a temperatura atingir limites pré-estabelecidos, por exemplo, acima de 37°.

No caso do paciente necessitar tomar um medicamento (por exemplo, *Paracetamol*) em determinado momento, a ação adequada do sistema pervasivo seria localizar em quais *dispensers* estão os medicamentos e informar ao paciente. Para localização do medicamento, a seguinte consulta seria executada:

```
Medicamento(?med) ^ nome(?med, ?nm) ^ estaEm(?med, ?disp) ^
swrlb:contains(?nm, "Paracetamol") ^ estaEm(?disp, ?comodo) ->
sqwrl:select(?disp, ?comodo)
```

Através da consulta acima, é retornado para o sistema os *dispensers* que contém os medicamentos, cujo nome contém a palavra "Paracetamol", e os cômodos onde se encontram. Logo, esses resultados poderiam ser apresentados ao paciente.

Ao enviar uma notificação ao paciente, deve-se primeiramente considerar a hipótese do paciente possuir alguma deficiência visual ou auditiva, para depois escolher a forma de entrega da informação. Logo, as seguintes consultas poderiam ser executadas:

- Para deficiência visual:

```
Paciente(?p) ^ possui(?p, ?def) ^ DeficienciaVisual(?def) ^
  grauDefVisual(?def, ?graudefv) -> sqwrl:select(?def, ?graudefv)
```

- E para deficiência auditiva:

```
Paciente(?p) ^ possui(?p, ?def) ^ DeficienciaAuditiva(?def) ^
  grauDefAuditiva(?def, ?graudefa) -> sqwrl:select(?def, ?graudefa)
```

Após essas consultas, o grau de cada deficiência (se existir) será retornado para o sistema e então escolhido como a mensagem vai ser entregue para o paciente. Por exemplo, se for constatado que o paciente possui deficiência visual, então a mensagem pode ser informada por texto, utilizando uma fonte maior.

6.8 Inferências

Uma grande vantagem na utilização de ontologias é a possibilidade de inferir novo conhecimento a partir do conhecimento existente. Este trabalho utiliza regras de inferência baseadas na linguagem SWRL. Tal como nas consultas, essas regras serão executadas sempre que as premissas forem verdadeiras.

Algumas regras foram criadas e são executadas conforme o sintoma apresentado pelo paciente. Por exemplo, agitação é um sintoma comum de demência e uma das maneiras adequadas para tranquilizar o paciente com tratamento não-farmacológico é através de música. Logo, a partir da constatação do sintoma e de que o cômodo em que o paciente está possui um equipamento de som, é possível que um sistema ligue esse aparelho. Isso se dá pela regra abaixo:

```
Paciente(?p) ^ apresentaSintoma(?p, ?sintoma) ^
  tipoSintoma(?sintoma, ?tipo) ^ swrlb:equal(?tipo, "agitação") ^
  estaEm(?p, ?comodo) ^ Som(?som) ^ contem(?comodo, ?som)
-> status(?som, true)
```

É possível por esta regra ver se o cômodo em que o paciente está possui um aparelho de som (propriedade de objeto *contem*). Caso tenha, e seja necessário ligar, a propriedade de dado *status* é marcada como *true*. Através disso, o sistema entende que deve ligar o aparelho proporcionando bem-estar ao paciente.

Outro caso interessante diz respeito às situações onde o paciente deve receber uma notificação em um dispositivo computacional próximo dele, como por exemplo o horário de um medicamento. Para isso, a seguinte regra seria executada:

```
Paciente (?p) estaEm (?p, ?comodo) ^ Equipamento(?equi) ^
status(?equi, ?s) ^ swrlb:equal(?s, true) ^
estaEm(?equi, ?comodo) ^ Notificacao (?note) ->
dispDestino(?note, ?equi)
```

A partir da regra acima, o motor infere na ontologia para qual dispositivo o sistema deve encaminhar uma notificação, nesse caso, um equipamento que esteja no mesmo cômodo da casa do paciente e que esteja ligado (*status = true*). Assim, o paciente seria notificado independente de sua localização no ambiente de *homecare* pervasivo.

O sistema pervasivo também é capaz de enviar alertas ao cuidador ou médico responsável pelo paciente, caso ocorra uma situação crítica de saúde. Isso é verificado a partir de diretrizes médicas, como verificação da frequência cardíaca. Por exemplo, quando a taxa de batimentos cardíacos for superior a 100 bpm (batimentos por minuto), o paciente está com taquicardia, então é necessário alertar o cuidador. Quando o sistema detectar situações desse tipo, também é necessário definir a prioridade do alerta, que pode assumir valores *alta*, *normal* ou *baixa*.

Nesse caso, o sistema primeiramente executa uma regra que associa uma mensagem à um alerta (id *AL001*), de acordo com a situação apresentada pelo paciente, como realiza a regra abaixo:

```
Paciente(?p) ^ possui(?p, ?sv) ^ SinaisVitalis(?sv) ^
frequenciaCardiaca(?sv, ?fq) ^ swrlb:greaterThan(?fc, 100) ^
Alerta(?al) ^ identificador(?al, ?id) ^
swrlb:equal(?id, "AL001") ->
mensagem(?al, "Pac. c/ Taquicardia") ^ prioridade (?al, "alta")
```

A partir da execução dessa regra, o alerta AL001 passa a conter uma mensagem informativa sobre o estado de saúde do paciente. No entanto, apenas foram definidas a mensagem e a prioridade, e não qual dispositivo irá receber a notificação. Isso pode ser alcançado através da seguinte regra de inferência:

```
Paciente(?p) ^ acompanha(?cuidador, ?p) ^
```

```

DispositivosMoveis(?disp) ^ pertenceA(?disp, ?cuidador) ^
Alerta(?al) ^ identificador(?al, ?id) ^
swrlb:equal(?id, "AL001") -> dispDestino(?al, ?disp)

```

Com a regra acima, o motor consegue inferir na ontologia o dispositivo computacional que deve receber o alerta. Isso é indicado através da relação *dispDestino*, a qual relaciona a instância da classe *Alerta* ao dispositivo do cuidador.

6.9 Considerações do capítulo

Este capítulo apresentou detalhadamente a ontologia criada para representar um ambiente *homecare* pervasivo voltado ao cuidado de pacientes portadores de demência. Além das classes, propriedades e restrições, foram discutidas algumas regras de inferência criadas em SWRL e consultas desenvolvidas em SQWRL.

A ontologia faz parte da metodologia proposta pelo trabalho e é de fundamental importância para o sistema de *homecare* pervasivo, visto que os dados descritos na ontologia descrevem todos os conceitos envolvidos neste tipo de ambiente. Essas informações serão utilizadas por sistemas pervasivos que implementem a arquitetura proposta, a qual será vista no capítulo seguinte.

7 ARQUITETURA DE UM SISTEMA HOMECARE PERVASIVO VOLTADO A PACIENTES PORTADORES DE DEMÊNCIA

Este capítulo apresenta a arquitetura proposta para um sistema de *homecare* pervasivo, cujo objetivo principal é auxiliar no tratamento de pacientes com demência, diminuindo a sobrecarga do cuidador e melhorando a qualidade de vida dele e do paciente, ao passo em o paciente continue a receber a atenção e cuidados necessários. Para isso, é necessário o sistema que se adapte ao contexto atual do ambiente, o qual pode ser representado pela ontologia de *homecare* apresentada no Capítulo 6.

A arquitetura proposta nesta dissertação toma como referência o trabalho de Freitas (2012), onde é descrita uma arquitetura genérica para monitoramento de pacientes em ambientes *homecare*. Logo, a partir da extensão dessa arquitetura, é possível desenvolver sistemas para auxiliar no cuidado doméstico de pacientes acometidos de demência. Para isso, notou-se a necessidade de adicionar alguns módulos e componentes à essa arquitetura. A Figura 7.1 apresenta uma visão geral da arquitetura, onde as partes destacadas em cor verde são resultantes do desenvolvimento desta dissertação.

A arquitetura é baseada em dois domínios, que interagem trocando informações sobre as entidades envolvidas em cada um. O primeiro refere-se ao ambiente *homecare*, ou seja, a casa do paciente, onde estão localizados os sensores, dispositivos computacionais pervasivos (TV digital, *smartphones*, etc.), a ontologia que representa o contexto atual do ambiente e plano de cuidados do paciente. Na nuvem computacional, o segundo domínio da arquitetura, ficam armazenadas informações referentes ao ambiente pervasivo. Essa nuvem contém o sistema de prontuário eletrônico do paciente (PEP), com sua base de dados, a ontologia que representa um ambiente *homecare* pervasivo voltado à pacientes portadores de demência, apresentada no Capítulo 6, e o módulo de servirá para realizar o processamento das informações que circulam na nuvem. É importante ressaltar que nesse ponto a ontologia servirá apenas como uma estrutura ontológica contendo as entidades e relacionamentos presentes que existem em um ambiente *homecare*. A partir dela será criada a ontologia que representará um contexto atual do ambiente.

Com a perspectiva de manter a interoperabilidade, a comunicação entre as partes da arquitetura será realizada através da troca de arquivos XML, os quais possuem *tags* que contém informações importantes sobre determinada situação de contexto, como o identificador de alguma entidade detectada por sensores. Para isso, cada módulo deve conter um *parser* para

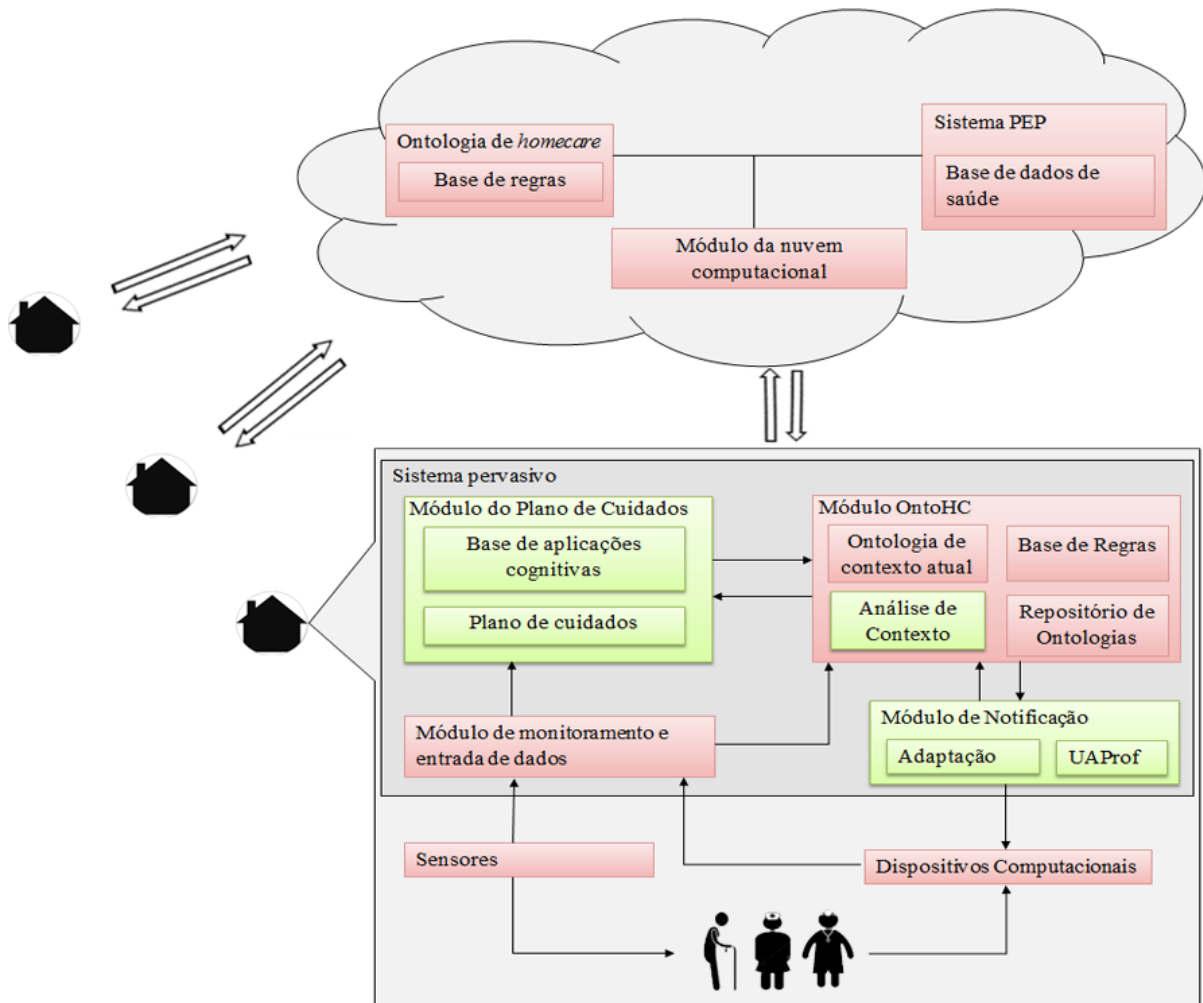


Figura 7.1: Arquitetura para sistemas de *homecare* pervasivos voltados ao cuidado de pacientes com demência.

manipular essas informações de forma adequada.

O funcionamento de um sistema que utilize essa arquitetura ocorre da seguinte maneira: sensores realizam frequentes varreduras no ambiente *homecare* em busca de informações relevantes ao cuidado do paciente, como chegada de um médico, alterações nos sinais vitais, atividade do paciente, etc. Quando um sensor captura uma informação de contexto, o *módulo de monitoramento e entrada de dados* é notificado. Esse módulo é responsável por processar as informações oriundas dos sensores e padronizá-las, utilizando arquivos XML, para que circulem dentro do sistema. Em seguida, essas informações estruturadas são enviadas ao módulo *OntoHC*.

O *módulo OntoHC* é considerado o módulo central do sistema. Após o recebimento das informações oriundas dos sensores, através do *módulo de monitoramento e entrada de dados*, o *módulo OntoHC* verifica se a ontologia de contexto atual possui informações das en-

tidades detectadas. Quando as entidades detectadas pelos sensores não estiverem representadas na ontologia, essa será enviada para o *repositório de ontologias* e uma nova estrutura ontológica será criada. Para isso, o *módulo OntoHC* envia as informações capturadas para o *módulo de processamento da nuvem*, o qual cria uma nova ontologia a partir da ontologia de *homecare* com as classes, relações, atributos e restrições referentes às entidades capturadas, que irá representar o contexto atual do ambiente, e preenche essa estrutura com informações atualizadas, provenientes do *sistema PEP*. Nesse momento, o *módulo de processamento da nuvem* envia a ontologia criada que representa o contexto atual ao *módulo OntoHC*. Porém, caso a ontologia de contexto atual possua as entidades, o módulo irá apenas atualizar essa ontologia, sem a necessidade de criar uma nova estrutura.

O *módulo OntoHC* realiza uma análise de contexto a fim de verificar se as informações capturadas pelos sensores são referentes à saúde do paciente (como frequência cardíaca), ou dizem respeito a alguma atividade realizada pelo paciente. Se as informações forem sobre a saúde do paciente, o módulo busca regras em sua base de regras que atendam as entidades detectadas pelos sensores, para que possam ser disparadas aplicações para serem utilizadas pelos usuários. Entre essas regras estão diretrizes médicas, as quais representam o conhecimento médico que será aplicado diretamente no sistema (por exemplo, quando a taxa de um sinal vital ultrapassa um limite pré-definido). Quando encontrada alguma aplicação, o *módulo de notificação* é comunicado e este envia a aplicação para um dispositivo computacional do usuário, para que este interaja com o sistema.

Caso as informações sejam sobre alguma atividade realizada pelo paciente, o *módulo OntoHC* verifica junto ao *módulo do plano de cuidados* se essa atividade é uma recomendação médica ou não. O *módulo do plano de cuidados* contém uma série de prescrições médicas para o tratamento do paciente, além de uma base de aplicações cognitivas que podem ser executadas a partir de um dispositivo computacional utilizado pelo paciente. Cada prescrição médica tem um horário específico para ser atendida. Assim, da mesma maneira que o *módulo OntoHC* se comunica com o *módulo do plano de cuidados*, este verifica junto ao *módulo OntoHC*, na ontologia de contexto atual, se o paciente está cumprindo a prescrição. Logo, pode-se enviar uma notificação para o cuidador ou médico, avisando sobre a aderência, ou não, do paciente ao tratamento, o que pode indicar uma mudança no estágio da doença.

Por fim, o *módulo de notificação* tem o objetivo de entregar as informações, sejam elas notificações ou aplicações cognitivas, aos dispositivos computacionais. Para isso, ele deve con-

siderar o dispositivo mais próximo e respeitar as necessidades e/ou deficiência dos usuários. Por exemplo, se um paciente com deficiência visual interage com uma TV digital, as informações apresentadas no *display* devem ser apresentadas de tal forma que esse paciente consiga ler. Cada uma das partes que compõem a arquitetura para sistemas de *homecare* pervasivos será descrita a seguir.

7.1 Prontuário eletrônico do paciente

O sistema de prontuário eletrônico do paciente (PEP) contém uma base com as informações de saúde do paciente, como medicamentos ministrados à ele e o histórico clínico, além das aplicações para manipulação desses dados. As informações disponíveis no PEP são utilizadas para preencher a ontologia de contexto atual, sempre que o módulo da nuvem computacional criar uma nova estrutura ontológica.

A escolha por hospedar o PEP na nuvem computacional se dá por diversos motivos. Através do armazenamento na nuvem, vários sistemas de *homecare* podem utilizá-lo, e não apenas um sistema em específico. Ainda, caso o paciente troque de ambiente, por exemplo quando o paciente for internado em um hospital por algum motivo, as informações podem ser obtidas do PEP da mesma forma, pois não sofrem alterações (VAQUERO et al., 2008).

Com relação à arquitetura proposta, a utilização do sistema em uma nuvem traz a possibilidade de alcançar o nível de abstração proposto pela computação pervasiva, onde os usuários acessam as informações desejadas em qualquer lugar, a qualquer hora, sem a necessidade de se preocupar com a infraestrutura computacional que mantém o sistema em funcionamento, a qual fica "invisível" aos usuários. Assim, qualquer dispositivo que permita acesso *web* e contenha um *browser* poderá acessar as aplicações do PEP. Além disso, outros ganhos da computação em nuvem, como a manutenção dos sistemas ou elasticidade de processamento de acordo com a necessidade seriam obtidos (GASSEN, 2010).

7.2 Sensores

Os sensores representam dispositivos básicos dentro da computação pervasiva que são utilizados para monitoramento do ambiente a fim de detectar mudanças no contexto. Esses sensores podem ser de diversos formatos e capturar diferentes tipos de dados. Neste trabalho não são discutidos os tipos de sensores que possam ser utilizados, visto que a arquitetura não impõe restrição quanto à escolha dos dispositivos e forma de implementação. Porém, na modelagem

da arquitetura são consideradas três classes de variáveis a serem monitoradas.

O primeiro tipo refere-se as variáveis ambientais, onde sensores capturam informações do ambiente como incidência de luz, som, umidade, temperatura, piso molhado, entre outras, que possam influenciar a saúde do paciente.

A segunda classe trata de informações sobre os dados fisiológicos do paciente. Entre eles estão pressão arterial, nível de saturação de oxigênio e frequência cardíaca.

Por fim, a última classe atende aos aspectos comportamentais. Tratando-se de pacientes portadores de demência, o monitoramento do comportamento é de grande importância, visto que esses pacientes necessitam maior auxílio em suas atividades diárias. Ainda, a mudança no comportamento do paciente pode indicar alteração no estágio da doença ou o não cumprimento de um tratamento prescrito pelo médico.

7.3 Módulo de monitoramento e entrada de dados

No trabalho de Freitas (2011), foi proposto um módulo chamado *módulo de monitoramento e distribuição das informações*, responsável tanto pelas informações que entram no sistema, quanto pelas informações recebidas pelas pessoas envolvidas no cuidado do paciente. Porém, pensando pelo lado da implementação da arquitetura, optou-se aqui por desmembrar o módulo original em dois módulos novos: o *módulo de monitoramento e entrada de dados* e o *módulo de notificação*. Dessa maneira, cada módulo tem sua função claramente definida e torna-se responsável por um comportamento específico do sistema pervasivo.

O *módulo de monitoramento e entrada de dados* é responsável por gerenciar os sensores presentes no ambiente de *homecare* pervasivo, bem como atuar como um *gateway* direcionando as informações que entram no sistema para que os módulos adequados as manipulem, como visto no diagrama de processos apresentado na Figura 7.2.

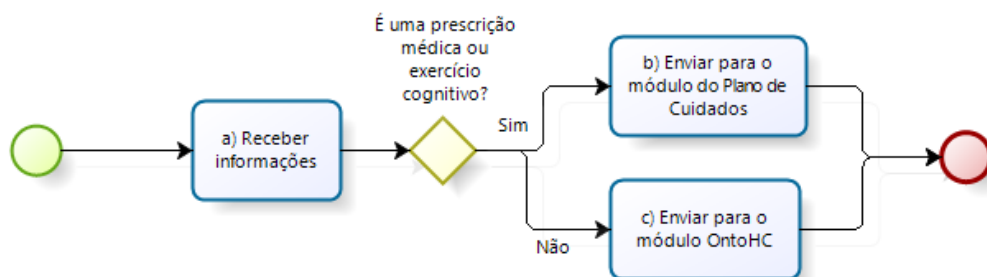


Figura 7.2: Fluxo das informações que entram no módulo de monitoramento e entrada de dados.

Por exemplo, quando o médico deseja inserir uma nova prescrição médica, ele pode cadastrar a mesma através do aplicativo móvel que está executando em seu *smartphone*, e então, o *módulo de monitoramento e entrada de dados* recebe essa informação (a) e envia o novo registro para o *módulo do plano de cuidados* (b), onde será manipulado e/ou armazenado.

Outra função do *módulo de monitoramento e entrada de dados* é estruturar as informações sobre alguma mudança de contexto, capturadas pelos sensores, em arquivos XML. A utilização de arquivos XML visa solucionar o problema da grande heterogeneidade de sensores encontrados em um ambiente de homecare e garantir a interoperabilidade entre os módulos do sistema. Assim, ao fazer uma chamada ao *módulo OntoHC*, o *módulo de monitoramento e entrada de dados* envia junto um arquivo XML *sensor.xml* contendo as informações de forma que o *módulo OntoHC* consiga manipulá-las. A Figura 7.3 apresenta o *XML Schema* (SPERBERG-MCQUEEN; THOMPSON, 2004) gerado a partir da ferramenta XML Spy (ALTOVA, 2012) dos documentos que contém informações dos sensores.

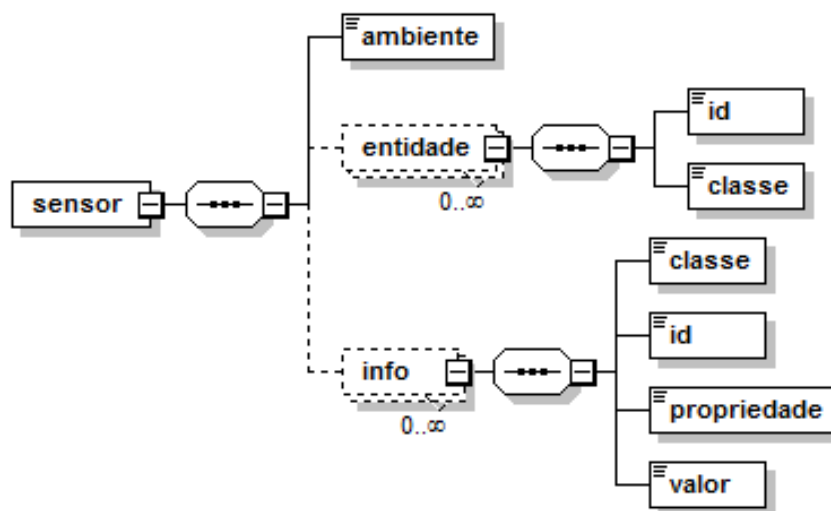


Figura 7.3: Esquema do arquivo XML com informações detectadas pelos sensores (*sensor.xml*).

A partir da Figura 7.3 pode-se entender como as informações são organizadas para serem enviadas ao *módulo OntoHC*. O documento XML é composto por um identificador do ambiente que o sensor está monitorando (elemento *ambiente*), além das entidades detectadas no ambiente (elemento *entidade*). Essas entidades são compostas por sua classe da ontologia (elemento *classe*) e por seu identificador (elemento *id*). Ainda, podem ser enviados valores mais específicos de determinadas classes (por exemplo, frequência cardíaca). Tais valores são representados pelo elemento *info*, o qual é composto pela classe (elemento *classe*) a qual pertence a propriedade, o nome da propriedade de dado da classe (elemento *propriedade*), o identificador

da entidade (elemento *id*) e o valor propriamente dito (elemento *valor*). O documento pode conter vários elementos *entidade*, bem como elementos *info*, tantos quantos forem detectados pelos sensores.

7.4 Módulo OntoHC

O *módulo OntoHC* é responsável pela manipulação da ontologia que representa o contexto atual do ambiente *homecare*, ou seja, as entidades encontradas no ambiente em determinado momento. Entretanto, algumas entidades devem estar mapeadas nesta ontologia, mesmo que não estejam, de fato, presentes no contexto atual do ambiente. Um exemplo é a entidade *Médico*. Um médico responsável por um paciente não necessita estar todo o tempo no ambiente, porém é importante mantê-lo mapeado na ontologia para caso seja necessário enviar uma notificação, por exemplo.

Esse tipo de configuração é definido quando os serviços de *homecare* são instalados na casa do paciente. Caso seja necessário ter uma determinada entidade sempre mapeada, essa classe passa a ter a propriedade de dado *necessario* com valor *true* na ontologia de contexto atual. Dessa forma, é possível indicar as classes necessárias em qualquer representação de contexto.

Além da ontologia, este módulo possui uma base de dados que contém um conjunto de regras de inferência e consultas, um repositório onde ficam armazenadas as ontologias que já foram utilizadas por algum contexto em específico, e o componente que realiza a análise das informações que chegam no módulo. Tais informações podem ser sobre alguma situação de contexto, oriunda dos sensores e recebida do *módulo do monitoramento e entrada de dados*, informações sobre prescrições médicas vindas do *módulo do plano de cuidados* ou uma ontologia que representa o contexto atual do ambiente enviada pelo *módulo de processamento da nuvem*. A Figura 7.4 apresenta, através de um diagrama de processos, o fluxo de informações capturadas pelos sensores, quando chegam ao *módulo OntoHC*.

No momento em que o *módulo OntoHC* recebe as informações sobre entidades detectadas pelos sensores, ele utiliza um conjunto de consultas SQWRL para verificar se tais entidades estão mapeadas na ontologia de contexto atual. Caso estejam, o *módulo OntoHC* deve verificar se as informações recebidas pelos sensores são referentes a alguma atividade realizada pelo paciente (e.g. caminhar) ou outras informações de contexto, como sinais vitais ou informações do ambiente (c). Essa tarefa é realizada pelo *componente de análise de contexto*.

Caso as informações sejam sobre alguma atividade realizada pelo paciente, o *módulo On-*

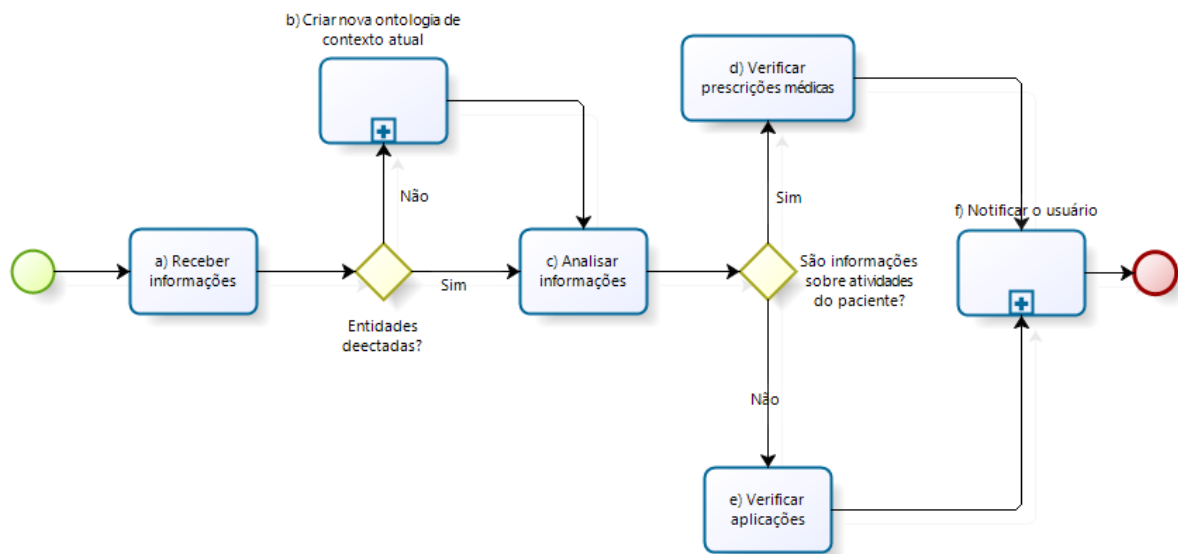


Figura 7.4: Fluxo de informações no módulo *OntoHC*.

toHC verifica junto ao *módulo do plano de cuidados* se essa atividade é uma prescrição médica ou se o paciente está realizando uma atividade por contra própria (d). Da mesma forma, quando há uma prescrição para determinado horário, o *módulo do plano de cuidados* verifica através deste componente, na ontologia de contexto atual, se o paciente está cumprindo, ou não, a prescrição. Com isso, o sistema de *homecare* pervasivo pode enviar uma notificação ao cuidador ou ao médico, informando sobre a aderência ao tratamento ou situações de risco, bem como enviar um elogio ao paciente por realizar as atividades.

Entretanto, se as informações não forem sobre a saúde do paciente, o *módulo OntoHC* irá verificar na sua base de regras, quais ações podem ser disparadas (e.g. ligar um equipamento) (e) e comunica o *módulo de notificação* para que tais ações sejam executadas (f).

Agora, retornando ao início do fluxo onde o *módulo OntoHC* recebe as informações (a) e considerando que a ontologia de contexto atual não possua as classes, uma nova ontologia deve ser criada (b). Para isso, o *módulo OntoHC* se comunica com o *módulo de processamento da nuvem*, responsável pela criação de uma nova ontologia, informando quais entidades devem ser mapeadas na nova ontologia. Nesta comunicação é enviado também um arquivo XML criado no *módulo OntoHC*, com informações recebidas do *módulo de monitoramento e entrada de dados*. A estrutura desse arquivo pode ser visualizada na Figura 7.5.

O documento é composto pelo elemento *dado*, o qual contém todas as classes (elemento *classe*) e o identificador dos indivíduos (elemento *id*) que foram detectados pelos sensores. O *módulo de processamento da nuvem* recebe apenas um arquivo XML contendo o contexto

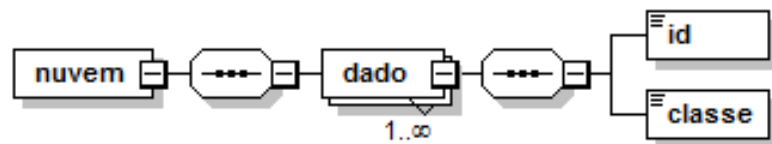


Figura 7.5: Esquema do arquivo XML que informa as classes encontradas no contexto atual (nuvem.xml).

modificado e com isso, esse módulo é capaz de criar uma ontologia contendo apenas informações referentes aos indivíduos detectados (atributos, relações, restrições, regras de inferência e consultas).

É importante considerar que antes da criação da nova ontologia, as informações que estão representadas na ontologia de contexto atual devem ser inseridas no PEP. Assim, junto com o arquivo *nuvem.xml*, o módulo de *OntoHC* também envia para o *módulo de processamento da nuvem* um outro arquivo, o qual contém as informações de cada entidade representada da ontologia de contexto atual. Este processo será detalhado na Seção 7.6.

A partir do momento que o *módulo OntoHC* envia os arquivos XML para serem utilizados na nuvem, ele também armazena a ontologia de contexto atual em um repositório de ontologias, localizado no próprio módulo. Então, quando o *módulo OntoHC* receber da nuvem computacional uma nova ontologia de contexto atual já instanciada, será possível manipular da maneira mais adequada, além de manter um histórico das mudanças que ocorreram no ambiente.

Após a atualização da ontologia de contexto atual, o fluxo das informações continua normalmente a partir do passo *c*.

É importante salientar que através do componente de *análise de contexto* é possível também que sejam adicionados novos módulos na arquitetura que necessitam utilizar informações da ontologia de contexto atual. Desta forma, seria necessário alterar apenas o componente, e não os módulos da arquitetura.

7.5 Módulo do plano de cuidados

O *módulo do plano de cuidados* tem a finalidade de gerenciar o plano de cuidados e a base de atividades cognitivas. O *plano de cuidados* contém prescrições médicas orientando o paciente a cumprir uma rotina específica do tratamento como horários de medição de pressão, medicação e exercícios que o paciente deva realizar. Dessa forma, o sistema permite uma economia de tempo e diminuição da sobrecarga do cuidador, além da individualização do paciente.

Para isso, a elaboração do plano de cuidados deve ser feita utilizando terminologia simples e de fácil compreensão, e mudando de acordo com as necessidades do paciente (GUIMARÃES et al., 2002). Com isso, estima-se que paciente consiga se auto gerenciar melhor, retardando os efeitos causados pela demência e mantendo-se um cidadão ativo e participativo (FLORIAN; SCHRAMM, 2004).

As prescrições médicas podem ser inseridas no plano de cuidados a partir de qualquer dispositivo computacional que o médico utilize e tenha acesso ao sistema, como um *smartphone*. A fim de padronização, cada nova prescrição é escrita em um documento XML, cuja estrutura pode ser vista na Figura 7.6. Assim, quando o médico cadastra uma prescrição, a aplicação do dispositivo móvel envia um arquivo XML para o *módulo de monitoramento e entrada de dados*, o qual encaminha a prescrição para *módulo do plano de cuidados*.

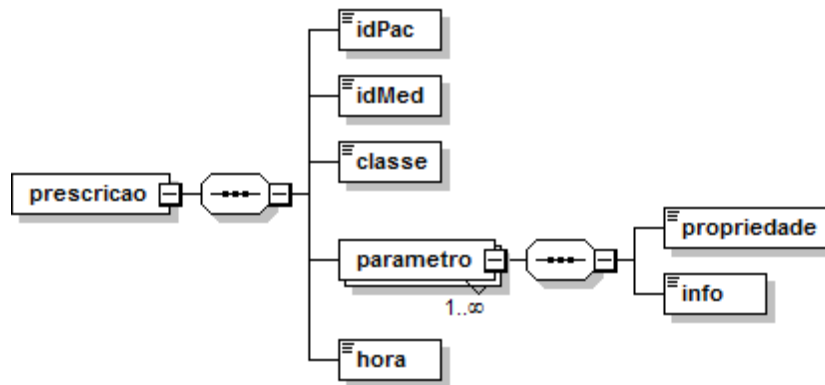


Figura 7.6: Esquema do arquivo XML utilizado para cadastrar prescrições médicas (prescricao.xml).

O documento de uma prescrição contém os seguintes elementos: *idPac*, indicando o paciente para qual a prescrição é destinada; *idMed*, o qual informa o médico, conectado ao sistema, que realizou a prescrição; *classe*, o qual representa a classe na ontologia (e.g. Medicamento); *parametro*, responsável por representar propriedades de dados da ontologia. Este elemento é composto por outros dois elementos, os quais indicam o nome da propriedade (elemento *propriedade*) e o valor dessa propriedade (elemento *valor*). Por exemplo, esses elementos poderiam ter como conteúdo *posologia*, para o elemento *propriedade* e *5mg* para o elemento *valor*.

Cada nova prescrição inserida no sistema pervasivo é associada a um evento, o qual é controlado por um temporizador. A partir do horário de uma prescrição, o sistema verifica na ontologia de contexto atual se o paciente está ou não seguindo o tratamento recomendado. E da mesma forma que o sistema verifica se o paciente está atendendo as prescrições médicas, ele verifica se, ao realizar um atividade, essa está mapeada no plano de cuidados ou está sendo

realizada por conta própria e, assim, um cuidador poderá ficar ciente do que o paciente está fazendo e minimizar riscos à sua saúde, por exemplo quedas.

Com o intuito de tornar o cuidado do paciente portador de demência mais efetivo, a arquitetura proposta neste trabalho inclui um repositório de exercícios cognitivos. Segundo Camara *et al.* (2009), objetiva-se com a estimulação cognitiva o resgate das funções comprometidas, o estímulo das áreas bloqueadas e o desenvolvimento de potenciais remanescentes do paciente, proporcionando a este uma melhor qualidade de vida.

Os exercícios são baseados nos sintomas característicos de demência (Capítulo 3) e focam em funções cognitivas como raciocínio, orientação e memória. A partir da prescrição no plano de cuidados, o sistema busca a atividade indicada na base de atividades e então envia-a a um dispositivo computacional do paciente, como um *tablet*. Pelos motivos já citados, essas atividades também são armazenadas no sistema utilizando arquivos XML, como representado na Figura 7.7.

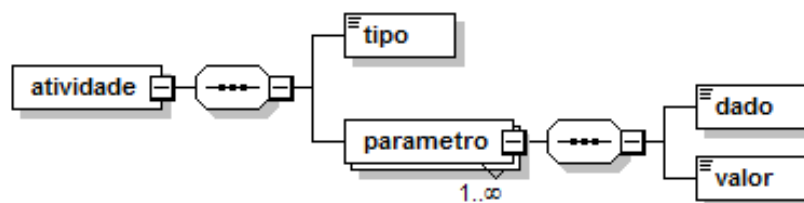


Figura 7.7: Esquema do arquivo XML que representa uma atividade cognitiva (atividades.xml).

Este documento é composto pelos seguintes elementos: *tipo*, o qual indica o tipo de exercício (memória, raciocínio, etc); e *parametro*. O último é composto pelos elementos *dado* e *valor*, os quais são responsáveis pelos parâmetros da atividade. Por exemplo, se for um exercício de memorização de uma sequência de cores, o documento poderia conter diversos elementos *parametro* que representariam as cores (elemento *dado*) e a ordem em que elas devam aparecer (elemento *valor*).

Com o documento XML acima é possível que o aplicativo que está executando no dispositivo computacional monte a interface e apresente a atividade ao paciente para que ele realize-a em qualquer lugar da casa.

7.6 Módulo de processamento da nuvem computacional

Este módulo é responsável pelo processamento das informações na nuvem computacional. Através desse módulo é criada a ontologia que contém apenas classes específicas referentes ao

contexto atual do ambiente *homecare*, conforme descrito na Seção 7.4.

Entretanto, antes da criação de uma nova ontologia, é necessário ter a base de dados do sistema PEP atualizada. Assim, juntamente com o arquivo *nuvem.xml*, que servirá para informar quais entidades devem ser mapeadas, o *módulo de processamento da nuvem* recebe também um arquivo XML (arquivo *pep.xml* enviado pelo *módulo OntoHC*) com informações referentes às entidades presentes na ontologia de contexto atual. Esse arquivo pode ser visualizado na Figura 7.8.

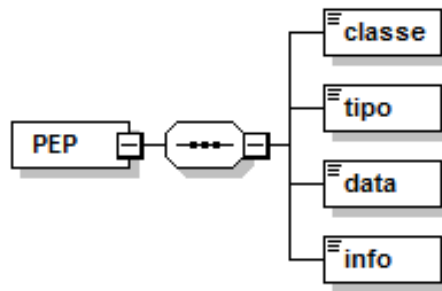


Figura 7.8: Esquema do arquivo XML que descreve os campos da base de dados de saúde do sistema PEP (*pep.xml*).

Este arquivo contém *tags* com informações referentes à cada entidade presente na ontologia. Ou seja, o *módulo OntoHC* deve enviar ao *módulo de processamento da nuvem* um arquivo para cada uma das entidades que estavam mapeadas na ontologia de contexto atual. A primeira *tag* refere-se à classe a qual pertence as informações. A segunda, armazena informações sobre o tipo de informação que está sendo enviada. Pode-se dizer que esta *tag* representa uma propriedade de dado da classe. Por sua vez, a terceira *tag* informa a data em que ocorreu a modificação naquela instância da ontologia. A necessidade dessa *tag* é devido ao fato de que o PEP só irá atualizar o seu conteúdo, caso a data seja diferente. A última *tag* guarda a instância da entidade, ou seja, a qual indivíduo da ontologia pertencem as informações presentes no documento XML pertencem.

Com a ajuda de um *parser*, *módulo de processamento da nuvem* consegue manipular essas informações e enviá-las para o sistema PEP, o qual atualiza sua base de dados (a). A atualização do sistema PEP corresponde à primeira tarefa do processo de criação de uma nova ontologia de contexto atual, o qual pode ser visualizado na Figura 7.9.

Depois de atualizada a base de saúde do paciente no sistema PEP, o *módulo de processamento da nuvem* cria uma nova estrutura ontológica (*OntoHc.owl*) (passo *b* do fluxo) com informações necessárias ao contexto atual, ou seja, ele busca na ontologia de *homecare* ape-

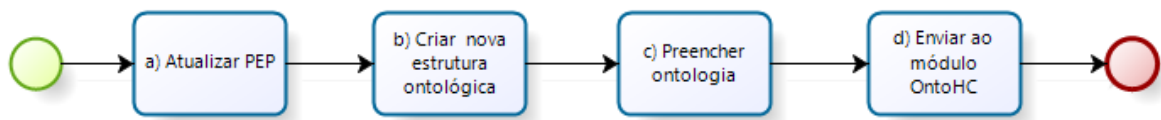


Figura 7.9: Fluxo do processo de criação de uma nova ontologia de contexto atual

nas a estrutura das classes, atributos, relações e restrições das entidades necessárias para um determinado momento.

Após criada a estrutura, o módulo faz consultas à base de dados de saúde do paciente a procura de informações para instanciar a nova ontologia (c). Em seguida, ela é enviada para o *módulo OntoHC* responsável pela manipulação da mesma (d). Nesse ponto, o sistema tem uma ontologia que representa o contexto atual em um módulo localizado dentro do ambiente *homecare* e assim não é necessário que o sistema se comunique mais com a nuvem computacional para executar uma aplicação, a menos que uma nova entidade seja detectada no ambiente. Logo, há uma melhoria de desempenho do sistema, uma vez que são utilizados recursos locais para seu funcionamento.

7.7 Módulo de notificação

Em contrapartida ao *módulo de monitoramento e entrada de dados*, o *módulo de notificação* tem o papel de entregar as informações do sistema para os dispositivos pervasivos do ambiente *homecare*, sejam eles dispositivos móveis ou fixos, além de permitir o envio de informações para usuários que não estão na casa, como um médico. Para notificar as pessoas dentro do ambiente, esse módulo necessita ser orientado à localização. Uma vez que o sistema saiba onde o paciente se encontra, ele deve exibir a devida notificação no dispositivo mais próximo. A Figura 7.10 apresenta o fluxo do processo de notificação de um usuário.

Quando há necessidade de enviar uma notificação para um usuário, o *módulo de notificação* é notificado pelo *módulo OntoHC* (a) e verifica junto a este as informações de contexto necessárias (b). No caso do usuário ser um paciente, o *módulo de notificação* deve ser considerar também as suas necessidades, visto que pessoas idosas tendem a ter alguma deficiência, seja ela motora, visual, auditiva, ou cognitiva.

O trabalho desenvolvido por Martini (2012) traz uma abordagem para personalização de interfaces de usuário em ambientes *homecare* pervasivos que pode ser utilizada nesse módulo. Além de uma ontologia que descreve o perfil dos usuários, o trabalho conta com um

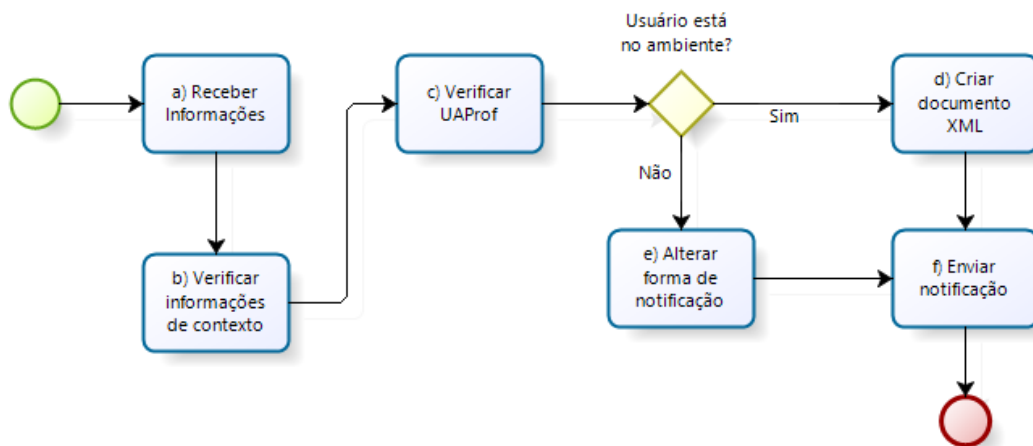


Figura 7.10: Fluxo do processo de notificação.

repositório o qual contém arquivos baseados na especificação *User Agent Profiling Specification* (UAProf). Cada documento UAProf fornece as características de *hardware* e *software* de um dispositivo em específico, além de informações sobre a rede a qual o dispositivo se encontra (MARTINI, 2012).

Ciente das informações de contexto, o *módulo de notificação* verifica junto ao UAProf (c) do dispositivo quais as adaptações podem ser aplicadas para personalizar a interface do usuário, conforme lhe seja mais oportuno, por exemplo, aumentar o tamanho da fonte.

A interação entre o *módulo de notificação* e os dispositivos pervasivos presentes no ambiente de *homecare* também se dá através de arquivos XML. Assim, o *módulo de notificação* cria um documento XML com as informações que devem ser apresentadas ao usuário (d) e, no momento em que se comunica com algum dispositivo computacional, envia esse documento (e), cuja estrutura é apresentada na Figura 7.11.

O arquivo enviado ao dispositivo computacional é composto por oito elementos. O primeiro deles refere-se ao identificador do dispositivo e serve para o sistema saber para qual dispositivo está enviando as informações. O segundo elemento refere-se ao usuário que irá interagir com o sistema pervasivo, onde é armazenado o código identificador do usuário. O terceiro elemento guarda a classe da ontologia a qual pertence o usuário. O elemento *idNotifi* faz referência à notificação que é gerada na ontologia de contexto atual. Esse elemento é utilizado quando, por exemplo, o paciente visualiza uma notificação e então o sistema atualiza a instância da notificação na ontologia de contexto atual, cujo *identificador* é igual ao valor armazenado nesse elemento. O elemento seguinte refere-se à tarefa que o usuário irá realizar através do dispositivo

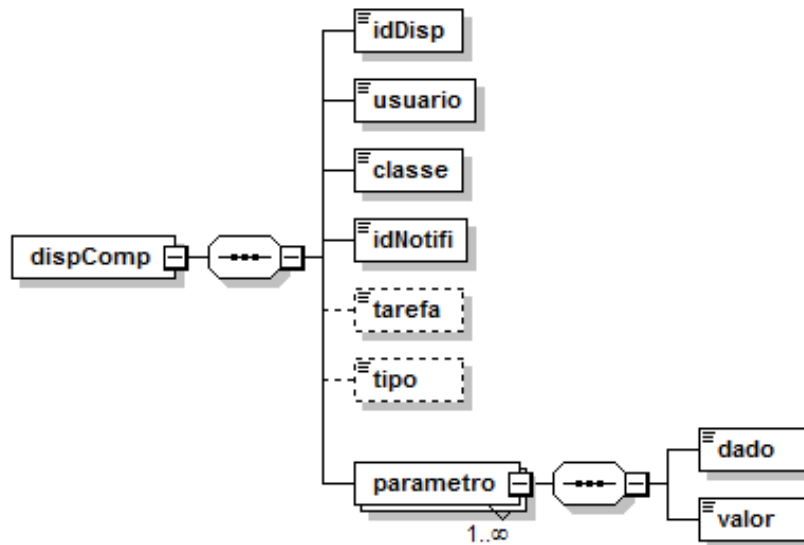


Figura 7.11: Esquema do arquivo XML para interação com os dispositivos computacionais (dispComp.xml).

computacional. A *tag tarefa* foi definida como opcional, visto que nem sempre o arquivo pode ser utilizado para enviar um simples alerta ao usuário, não exigindo o cumprimento de uma tarefa. O conteúdo dessa *tag* armazena uma propriedade de objeto da ontologia, por exemplo *tomar*, indicando que o paciente deva tomar um medicamento.

Por sua vez, o elemento *tipo* representa o tipo de tarefa que será realizada. Da mesma forma que o elemento anterior, este foi definido como opcional. Por fim, o elemento *parametro* é composto por outras duas *tags*, as quais armazenam informações necessárias para a execução da tarefa. A primeira refere-se ao *dado* que está sendo passado, enquanto *valor* armazena o valor específico de *dado*. O elemento *dado* pode conter uma propriedade de dado da ontologia, como nome de um exame ou um sinal vital.

Com esse arquivo, os aplicativos móveis que estarão executando nos dispositivos computacionais serão capazes de apresentar as informações ao usuário, seja um alerta ou um lembrete de tarefa.

Porém, caso o usuário esteja fora do ambiente de *homecare*, é possível adaptar o serviço de notificação (f) e enviar uma mensagem SMS, por exemplo.

7.8 Dispositivos computacionais

Muitas vezes os usuários precisarão entrar com informações no sistema de forma ativa, ao invés de apenas serem passivos do monitoramento. Um paciente que deve realizar uma exercício cognitivo, por exemplo, pode fazê-lo através de um dispositivo computacional que permita a

interação com o sistema de *homecare* pervasivo. Entre estes dispositivos estão *smartphones*, *smart TVs*, entre outros. Para isso, é necessário ter aplicações executando em tais dispositivos.

Nesta parte da abordagem proposta se encaixam os desenvolvedores de aplicativos para dispositivos móveis. Os desenvolvedores implementam a lógica de sua aplicação e criam uma interface padrão para seu aplicativo.

Nesta dissertação não serão discutidas questões referentes à lógica de implementação e interface dos aplicativos. No entanto, os aplicativos devem atender alguns requisitos básicos. Primeiramente, o aplicativo móvel deve ser capaz de interpretar os documentos XML recebidos pelo dispositivo computacional e apresentar as informações ao usuário, bem como adaptar a interface quando for o caso (e.g. aumentar o tamanho de um botão). É necessário também que o aplicativo consiga identificar e atribuir as permissões aos diferentes usuários do sistema, fornecendo as interfaces para que estes executem suas tarefas. Por exemplo, enquanto o paciente pode apenas realizar exercícios cognitivos e receber notificações, o médico tem permissão para cadastrar novos exercícios e novas diretrizes.

Para o desenvolvimento dos aplicativos móveis sugere-se a utilização de qualquer linguagem que possua a lógica separada do projeto de interface. O *Android* por exemplo, possui a lógica descrita na linguagem Java e a interface descrita em XML (FRIESEN, 2010).

7.9 Considerações do capítulo

Esse capítulo apresentou uma arquitetura para sistemas pervasivos, voltado ao cuidado de pacientes com demência em ambientes *homecare*, a qual teve como referência a arquitetura proposta em (FREITAS et al., 2012). Foram descritas as contribuições da dissertação em relação a arquitetura de referência, além de cada uma das partes que a compõe, bem como os dados de entrada e saída de cada módulo.

A implementação desta arquitetura, utilizando a ontologia apresentada no Capítulo 6, possibilita o desenvolvimento de aplicações que: (a) auxiliem o cuidado de pacientes acometidos de doenças em suas casas; (b) diminuam a sobrecarga do(s) cuidador(es); (c) aumentem a qualidade de vida do paciente e retardem o avanço da doença.

A fim de evidenciar a abordagem proposta, a seguir será apresentado um estudo de caso o qual representa uma situação de funcionamento de um sistema pervasivo desenvolvido com base nesta abordagem.

8 ESTUDOS DE CASOS E RESULTADOS

Para evidenciar a funcionalidade de sistemas desenvolvidos a partir da abordagem proposta deste trabalho, bem como exemplificar o funcionamento de um sistema que utiliza a mesma, foram realizados dois estudos de casos. O primeiro caso tem o objetivo de demonstrar como um sistema poderia auxiliar o paciente a gerir suas tarefas rotineiras de forma autônoma. Ou seja, além diminuir a necessidade de um cuidador, o paciente se mantém ativo, auto gerenciando seu tratamento.

O segundo caso, por sua vez, ilustra como um sistema pervasivo conseguiria estimular as funções cognitivas de um paciente com demência, de forma a reduzir a deterioração cognitiva através de exercícios.

8.1 Estudo de caso 1: Notificação de prescrição médica

O primeiro estudo de caso foi adaptado de Zarghami et al. (2011). No cenário, John é um paciente que apresenta doença obstrutiva pulmonar crônica de baixa escala. Apesar de não ser um caso específico de demência, John apresenta problemas moderados de memória e necessita ser lembrado de suas atividades, como tomar um medicamento, tal como pacientes portadores de demência. Além disso, John possui deficiência auditiva leve.

Sua qualidade de vida é melhorada quando o paciente se mantém ativo. No entanto, John necessita que seus sinais vitais sejam monitorados em função de sua doença, visto que uma queda brusca no nível de saturação de oxigênio, por exemplo, pode levar à hospitalização onde o cuidado será mais caro e terá de ficar um período longo.

O paciente convive com sua esposa, Marie, e é auxiliado por Nancy, cuidadora de idosos responsável por criar e adaptar os serviços de *homecare* instalados na casa do paciente, com base no estado de saúde do paciente. Dentre esses serviços existem lembretes para que John tome os medicamentos em horários determinados e lembre de fixar os aparelhos medidores de sinais vitais.

Para isso, a casa é equipada com *Tablets*, *PDA*s, *dispensers* de medicamentos inteligentes e medidores de sinais vitais, além dos sensores de ambiente. Nesse sentido, o sistema adapta o serviço de notificação para direcionar o lembrete para dispositivo computacional de John, o qual possui um aplicativo móvel executando, considerando sua deficiência auditiva. John, quando acompanhado, prefere receber o lembrete sem volume, apenas com seu dispositivo vibrando.

Caso o *tablet* não esteja funcionando, o sistema automaticamente direciona notificação para o PDA da Marie.

Em caso de frequentes lembretes ignorados (não visualizados) pelo paciente, o sistema re-adapta o serviço (aumentando o volume, por exemplo) e notifica Nancy. Caso ocorra alguma crise de saúde, por exemplo quando um sinal vital atinge um limite pré-definido, um alerta é enviado à central médica, responsável pelos serviços de *homecare* do paciente.

Para o estudo de caso é considerado o cenário onde o paciente necessita tomar um medicamento prescrito por um médico. Nesse caso, o sistema deve informar ao paciente onde está localizado o medicamento. Caso o paciente não tome o medicamento, um alerta é enviado à Nancy para que ela auxilie John.

Primeiramente, é importante ressaltar que cada prescrição é cadastrada por um médico a partir do seu dispositivo móvel conectado ao sistema, o qual deve estar executando o aplicativo móvel. Então, as informações da prescrição são enviadas ao *módulo de monitoramento e entrada de dados*. Esse módulo encaminha as informações da prescrição para o *módulo de plano de cuidados*, o qual cria um arquivo XML para cada prescrição (*prescricao.xml*) e a armazena em um repositório. Para esse estudo de caso, o arquivo *prescricao.xml* contém as seguintes informações:

```
<?xml version="1.0" encoding="UTF-8"?>
  <prescricao xmlns:xsi="prescricao.xsd">
    <idPac>PAC001</idPac>
    <idMed>MED001</idMed>
    <classe>Medicamento</classe>
    <parametro>
      <propriedade>nome</propriedade>
      <info>Donepezilo</info>
    </parametro>
    <parametro>
      <propriedade>posologia</propriedade>
      <info>1 comprimido</info>
    </parametro>
    <parametro>
      <propriedade>necessitaAuxilio</propriedade>
      <info>>false</info>
    </parametro>
    <hora>12:00</hora>
  </prescricao>
```

Com base no horário da prescrição (*tag hora*) o sistema pervasivo deve notificar o paciente para que este atenda o tratamento. Percebe-se também que a tarefa não necessita a ajuda do cuidador (*tag necessitaAuxilio*). Esse dado é informado pelo médico ao cadastrar a prescrição, com base no estágio da doença e nas capacidades do paciente.

Quando uma prescrição é cadastrada, ela é associada a um evento, controlado por um temporizador, o qual é responsável por iniciar o processo de notificação do paciente. O fluxo do processo de notificação é apresentado na Figura 8.1 através de um diagrama de processos.

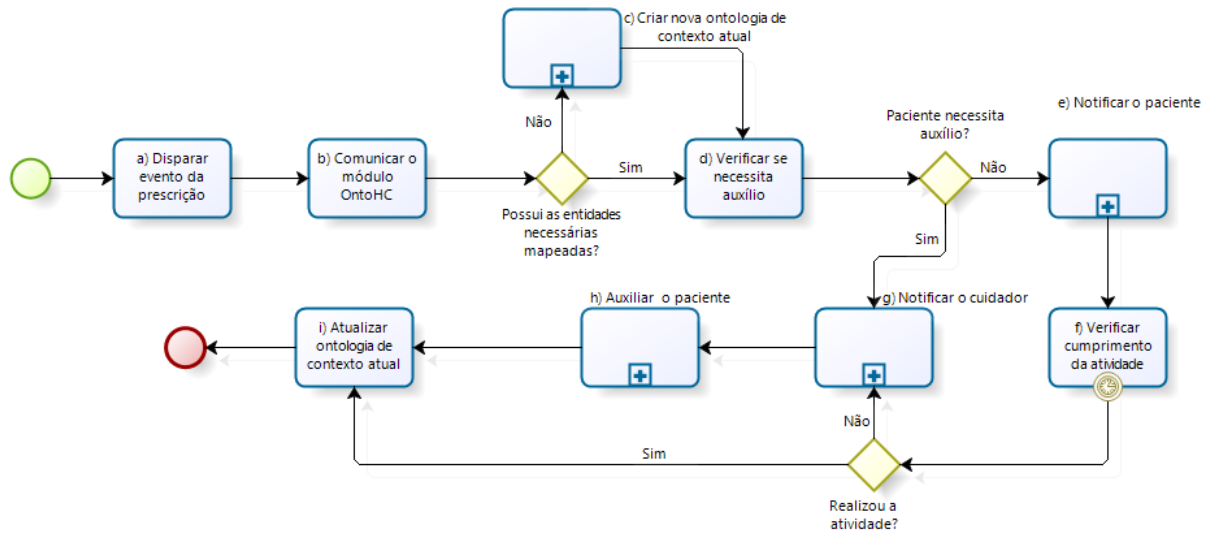


Figura 8.1: Fluxo de um notificação de prescrição médica

A partir do momento em que o evento é disparado (a), o *módulo do plano de cuidados* notifica o *módulo OntoHC* informando sobre a prescrição em questão (b). Ao estabelecer essa comunicação também é enviado o arquivo *prescrição.xml*.

Através de consultas, disponíveis em sua base de regras, o *módulo ontoHC* verifica junto a ontologia de contexto atual se o paciente está realizando a atividade prescrita. Para que seja possível essa verificação, primeiramente as entidades envolvidas devem estar mapeadas na ontologia de contexto atual. Nesse ponto é importante ressaltar que a ontologia de contexto atual representa o conhecimento encontrado no ambiente naquele momento. Porém, algumas entidades devem estar sempre presentes na ontologia, visto que são importantes para o funcionamento do sistema, por exemplo, *Médico*. Porém, se as entidades não estiverem mapeadas, uma nova ontologia deverá ser criada para representar o contexto atual do ambiente pervasivo.

O processo de criação de uma nova ontologia (c) é responsabilidade do *módulo de processamento da nuvem*. Primeiramente, é necessário atualizar o sistema PEP, visto que o *módulo de processamento da nuvem* irá utilizar informações armazenadas nesse sistema para preencher a nova ontologia. Para isso, o *módulo OntoHC* cria um arquivo XML (*pep.xml*) e insere as informações referentes às entidades presentes na ontologia até então. Após, esse arquivo é enviado para o *módulo de processamento da nuvem*. O arquivo *pep.xml* pode conter, por exemplo,

informações sobre os sinais vitais do paciente, como mostra o código a seguir:

```
<?xml version="1.0" encoding="UTF-8"?>
  <pep xmlns:xsi="pep.xsd">
    <classe>SinaisVitais</classe>
    <tipo>frequenciaCardiaca</tipo>
    <data>20121225</data>
    <info>82</info>
  </pep>
```

Após enviado o arquivo que servirá para atualizar o sistema PEP, a ontologia de contexto atual é armazenada no repositório de ontologias e, então, o *módulo OntoHC* cria um novo arquivo, chamado *nuvem.xml*, que irá conter as informações sobre as entidades necessárias para representar o contexto atual do ambiente. Por exemplo, se a classe *Medicamento* não estivesse mapeada na ontologia, mas necessária ou presente no contexto atual, o arquivo XML iria conter as seguintes informações:

```
<?xml version="1.0" encoding="UTF-8"?>
  <nuvem xmlns:xsi="nuvem.xsd">
    <dado>
      <id>Pac001</id>
      <classe>Paciente</classe>
    </dado>
    <dado>
      <id>Med001</id>
      <classe>Medicamento</classe>
    </dado>
  </nuvem>
```

Assim como o anterior, esse arquivo será enviado para a nuvem computacional para ser criada a nova ontologia. Na nuvem, o *módulo de processamento da nuvem* cria uma nova estrutura ontológica a partir da ontologia de *homecare*, a qual também está presente na nuvem computacional. Essa estrutura irá conter as classes, relações, atributos e restrições referentes às entidades necessárias, e será preenchida com informações atualizadas oriundas do sistema PEP.

Após criada, o módulo de processamento da nuvem envia a nova ontologia de contexto atual (*ontohc.owl*) para o *módulo OntoHC*, o qual é responsável por manipula-la de maneira adequada. Nesse momento, o *módulo OntoHC* consulta sua base de regras e verifica a relação entre as classes *Medicamento* e *Paciente* através da propriedade *toma*, indicando que o paciente deve tomar o medicamento.

A partir desse ponto, o sistema continua o fluxo do processo de notificação, verificando se na prescrição médica está definido se o paciente necessita ou não auxílio para a realização da tarefa (d). Essa verificação é importante para que o sistema saiba para qual dispositivo irá enviar o lembrete. No cenário adotado é considerado que o paciente possa tomar o medicamento sozinho. Logo, o sistema deve notificar John para que ele tome seu medicamento (e).

Nesse momento, o *módulo OntoHC* realiza consultas de contexto atual para buscar informações sobre o dispositivo e localização do *dispenser* que contém o medicamento. Um exemplo é a consulta abaixo, que verifica se o dispositivo de John está conectado ao sistema:

```
Paciente(?pac) ^ utiliza(?pac, ?disp) ^ status(?disp, ?sta)
-> sqwrl:select(?sta)
```

Após constatar que o dispositivo computacional de John está funcionando (*status true*), é gerado um arquivo XML contendo as informações necessárias para que o aplicativo móvel apresente o lembrete ao paciente. Parte do arquivo XML é apresentado a seguir:

```
<?xml version="1.0" encoding="UTF-8"?>
  <dispComp xmlns:xsi="dispComp.xsd">
    <idDisp>smartPhone01</idDisp>
    <usuario>PAC001</usuario>
    <classe>Paciente</classe>
    <idNotifi>lemb001</idNotifi>
    <tarefa>toma</tarefa>
    <tipo>Medicamento</tipo>
    <parametro>
      <dado>nome</dado>
      <valor>Donepezilo</valor>
    </parametro>
    <parametro>
      <dado>posologia</dado>
      <valor>1 comprimido</valor>
    </parametro>
    <parametro>
      <dado>estaEM</dado>
      <valor>dispenser001</valor>
    </parametro>
    <parametro>
      <dado>status</dado>
      <valor>>false</valor>
    </parametro>
    ....
  </dispComp>
```

Com o arquivo XML criado, o *módulo OntoHC* infere na ontologia que a notificação deve ser enviada, utilizando uma regra de inferência. Após, o arquivo é encaminhado ao *módulo de notificação*, o qual é responsável por enviá-lo para o dispositivo computacional do paciente. A partir do recebimento do arquivo XML, a aplicação que está executando no dispositivo computacional é capaz de apresentar o lembrete ao usuário.

A Figura 8.2 apresenta uma simulação de como a notificação poderia ser visualizada pelo paciente no seu smartphone.

A partir da visualização da notificação, o sistema pervasivo verifica se o paciente cumpriu a atividade, ou não, dentro de um intervalo de tempo pré-definido (f). Essa informação pode



Figura 8.2: Ilustração do aplicativo móvel apresentando um lembrete ao paciente

ser capturada pela rede de sensores do ambiente de *homecare* pervasivo. Nesse estudo de caso, considera-se que o paciente cumpre a orientação no momento em que ele retira o medicamento *dispenser*. Caso fosse uma prescrição para realizar alguma atividade física, os sensores poderiam verificar a aderência à prescrição através de movimentos, por exemplo.

Caso o paciente não tenha realizado a tarefa, o sistema pervasivo redireciona a notificação para o dispositivo computacional do cuidador (g). Esse, por sua vez, auxilia o paciente na realização da atividade (h). Por fim, a ontologia de contexto atual é atualizada, indicando que a tarefa foi realizada (i). Para isso, a *tag status* do arquivo *dispComp.xml* é alterada para *true* e o arquivo enviado ao *módulo de monitoramento e entrada de dados*, o qual comunica o *módulo OntoHC* para que a ontologia seja atualizada. Quando o cuidador auxilia o paciente, a aplicação permite que o cuidador informe o cumprimento da tarefa de forma manual, através do seu dispositivo computacional.

Com este estudo de caso é possível perceber que o sistema diminui a sobrecarga do cuidador. Embora este tenha que auxiliar o paciente, caso ele não cumpra uma prescrição médica, o tratamento não fica vulnerável à algum tipo de falha humana, como esquecimento de medicação, causado principalmente pela exaustão emocional do cuidador. Logo, o cuidador só irá intervir quando o paciente necessitar ajuda para realizar determinada tarefa.

Ainda, o paciente é capaz de manter sua autonomia, se mantendo ativo no tratamento, de

forma que também diminua a necessidade de intervenção do cuidador. É importante considerar que para o estudo de caso assume-se que o paciente possui condições de operar seu *smartphone*, porém, o sistema poderia enviar as informações por um sistema de som ou televisão, se fosse necessário.

8.2 Estudo de caso 2: Exercício de estimulação cognitiva

O segundo estudo de caso foi gerado a fim de demonstrar como o sistema pode auxiliar os pacientes com demência a estimularem suas funções cognitivas remanescentes, retardando a deterioração. Para isso, o sistema pervasivo contém uma base de exercícios cognitivos, os quais podem ser acessados pelo dispositivo computacional do paciente.

Os exercícios cognitivos são inseridos em arquivos XML e armazenados na *base de aplicações cognitivas*. Dessa forma, um exercício pode ser facilmente modificado pelo cuidador ou médico, bem como utilizado por qualquer aplicação que leia arquivos XML. Um exemplo é mostrado a seguir:

```
<?xml version="1.0" encoding="UTF-8"?>
  <atividade xmlns:xsi="atividadescog.xsd">
    <tipo>memoria</tipo>
    <descricao>texto</descricao>
    <parametro>
      <dado>1</dado>
      <valor>1</valor>
    </parametro>
    <parametro>
      <dado>4</dado>
      <valor>2</valor>
    </parametro>
    <parametro>
      <dado>2</dado>
      <valor>3</valor>
    </parametro>
    <parametro>
      <dado>0</dado>
      <valor>4</valor>
    </parametro>
    ...
  </atividade>
```

O arquivo acima refere-se a um exercício de memória (*tag tipo*), onde será apresentada uma sequência de números (*tag dado*) e a ordem em que esses números irão aparecer (*tag valor*). Assim como no caso anterior, o paciente deve realizar alguma atividade a partir de uma prescrição médica. Logo, a prescrição serve tanto para medicamentos, quanto para atividades físicas e cognitivas.

O fluxo desse estudo de caso se assemelha ao anterior. A partir do evento de uma prescrição

médica o *módulo do plano de cuidados* irá se comunicar com *módulo OntoHC* informando sobre a tarefa que deve ser realizada naquele determinado momento. Para isso é necessário ter as entidades necessárias mapeadas na ontologia de contexto atual.

Então, o *módulo OntoHC* comunica-se como o *módulo de notificação* para que a notificação seja entregue ao paciente. No entanto, o estudo de caso considera que o paciente não está com o dispositivo em mãos, logo, deve-se enviar uma notificação para um dispositivo computacional próximo. O *módulo de notificação* busca estas informações junto ao *módulo OntoHC* através do componente de *análise de contexto*. Para isso o *módulo OntoHC* executa a seguinte consulta:

```
Paciente(?p) ^ estaEm(?p, ?comodo) ^ Equipamento(?disp) ^
estaEm(?disp, ?comodo) ^ status(?disp, ?sts) ^
identificador(?disp, ?id) -> sqwrl:select(?id, ?sts)
```

A consulta traz como resultado que no cômodo que o paciente está existe uma TV Digital (identificador tv001) ligada (*status true*). Logo, a TV será utilizada para notificar o paciente para que ele porte o dispositivo móvel pelo qual irá realizar o exercício cognitivo.

Junto com a notificação, o sistema envia um arquivo XML ao *tablet* do paciente contendo as informações sobre a tarefa que deve ser executada. Após pegar o *tablet* e visualizar a notificação no dispositivo, o aplicativo móvel envia o arquivo XML para o *módulo de monitoramento e entrada de dados*. Nesse momento, o arquivo contém as seguintes informações:

```
<?xml version="1.0" encoding="UTF-8"?>
  <dispComp xmlns:xsi="dispComp.xsd">
    <idDisp>tablet001</idDisp>
    <usuario>paciente001</usuario>
    <classe>Paciente</classe>
    <idNotifi>note001</idNotifi>
    <tarefa>realiza</tarefa>
    <tipo>AtividadeCognitiva</tipo>
    <parametro>
      <dado>tipoAtividadeCognitiva</dado>
      <valor>memoria</valor>
    </parametro>
    <parametro>
      <dado>status</dado>
      <valor>>false</valor>
    </parametro>
    <parametro>
      <dado>hora</dado>
      <valor>15:00</valor>
    </parametro>
  </dispComp>
```

O *módulo de monitoramento e entrada de dados* recebe o arquivo e notifica *módulo OntoHC* sobre a atividade que o paciente irá realizar. Então, o *módulo OntoHC* busca junto ao *módulo do*

plano de cuidados o exercício correspondente, a qual é enviado ao *tablet* do paciente. A partir do recebimento desse arquivo, o aplicativo móvel estrutura a interface e apresenta o exercício ao paciente. A Figura 8.3 mostra como o paciente pode interagir com o aplicativo móvel, nesse caso.



Figura 8.3: Ilustração do aplicativo móvel apresentando um exercício cognitivo ao paciente

Após realizar a atividade, o *módulo OntoHC* é notificado e atualiza a ontologia de contexto atual informando que a atividade foi realizada, bem como o horário de início e fim, para posterior análise de um médico. Caso o paciente não realize a atividade em um tempo pré-definido, o cuidador pode receber uma notificação no seu dispositivo computacional.

Dessa forma, esse caso de estudo mostra como o sistema pervasivo possibilita o retardo na deterioração das funções cognitivas do paciente, uma das metas do tratamento da demência. Nesse caso, o paciente pode realizar um exercício cognitivo de memória a partir do seu *tablet*, sem que seja necessário se deslocar até um cômodo específico ou ser auxiliado por um cuidador. O exercício cognitivo é inserido em um arquivo XML de maneira clara e objetiva. Assim, é possível que uma aplicação utilize um *parser XML* e monte a interface gráfica para a tarefa.

8.3 Considerações do capítulo

A partir da análise dos casos de estudo apresentados percebe-se que a abordagem proposta neste trabalho pode ser aplicada para auxiliar o cuidado de pessoas portadoras de demência em suas próprias casas, diminuindo a sobrecarga do cuidador e mantendo a independência funcional do paciente sem que este receba um tratamento precário. Um sistema pervasivo que utilize a abordagem proposta é não só capaz de monitorar a saúde do paciente, mas também fornecer aplicações a fim de estimular e preservar suas funções cognitivas e motoras.

Entretanto, é necessário ressaltar que para o funcionamento correto desta proposta, a infraestrutura do ambiente domiciliar em questão necessita de dispositivos computacionais que o caracterizem como um ambiente pervasivo, bem como dito no início do capítulo. A arquitetura proposta nesta dissertação pode ser comparada com trabalhos relacionados à este encontrados na literatura, os quais são abordados no próximo capítulo.

9 TRABALHOS RELACIONADOS

Este capítulo apresenta trabalhos relacionados a proposta desta dissertação, ou seja, trabalhos que utilizam conceitos de computação pervasiva que possam ser utilizados para auxiliar no tratamento de pacientes com demência em ambientes domiciliares. Segundo, Engelhardt *et al.*(2005), o tratamento do paciente portador de demência deve atender três pilares: diminuição da dependência funcional do paciente; redução da sobrecarga do cuidador; e retardo na deterioração das funções cognitivas e motoras do paciente.

Em (RIAÑO *et al.*, 2009) é descrito o *K4Care*, um projeto voltado a desenvolver uma plataforma web que auxilie serviços para profissionais da saúde no cuidado de pacientes idosos em *homecare*. O projeto, no entanto, considera pacientes portadores de doenças crônicas que estejam de alguma forma incapazes. Tal projeto é baseado em um plano de intervenções individual e duas ontologias. A primeira contém os atores presentes em um ambiente *homecare* e as ações que cada atuador pode realizar, enquanto a segunda ontologia é responsável pelas informações médicas sobre sintomas, síndromes e doenças. Ambas as ontologias são representadas utilizando a linguagem OWL, porém, diferentemente desta dissertação, elas não fazem usos de inferências sobre as mesmas, utilizando-as apenas para consultas. Ainda, o trabalho apresentado por Riaño *et al.* (2009) foca em um plano de intervenção de pacientes em caso de emergência, não disponibilizando aplicações para esses usuários. Dessa maneira, o paciente não consegue gerenciar seu tratamento, e, conseqüentemente, retardar o avanço da doença.

O trabalho de Tapia e Corchado (2009) apresenta um sistema (*ALZ-MAS*) baseado em multi-agentes cujo objetivo é melhorar os cuidados e assistência de pacientes com *Alzheimer* em residências geriátricas. O sistema mantém o foco no monitoramento dos pacientes, como localização, tarefas diárias e situações anormais de saúde. Para isso, ele utiliza tecnologias conscientes do contexto para obter informações dos usuários dentro do ambiente, através de modelos BDI (CORCHADO *et al.*, 2004). Além disso, o sistema fornece um plano de cuidados para gerenciar as tarefas que devem ser realizadas pelos os pacientes da residência. Esse plano pode ser alterado a partir de qualquer dispositivo móvel que um médico possua e disponibilizado para todos os enfermeiros. Com isso, a preocupação do cuidador em lembrar das atividades que o paciente deva realizar é diminuída, visto que ele será avisado. Porém, como o sistema é direcionado apenas aos profissionais envolvidos no tratamento do paciente, e este fica passivo de monitoramento, o que não o diminui a sua dependência. Já na proposta desta dissertação, o

paciente pode interagir com o sistema pervasivo. Ainda, o trabalho de Tapia e Corchado não oferece exercícios de estimulação cognitiva ao paciente.

Por sua vez, o mPCA *Mobile Patient Care-Giving Assistant* (HELAL et al., 2003) é um sistema pervasivo cujo objetivo é auxiliar pacientes idosos com demência em suas casas, utilizando informações do contexto fornecidas pelo ambiente inteligente. De acordo com os autores, o sistema foca-se em enviar lembretes ao paciente para que realize suas tarefas através de um *smartphone*. Entre essas tarefas estão tomar medicamento, fazer uma refeição e visitar o médico. Ainda, o sistema fornece um passo-a-passo para o paciente executar as tarefas, de maneira proativa. É possível também, pelo sistema, que o paciente localize objetos importantes para ele dentro do ambiente, como um animal de estimação ou frascos de remédio. Por último, o sistema registra as atividades realizadas pelo paciente em um banco de dados localizado em uma nuvem computacional. Através dessa proposta é possível que o paciente tenha um autogerenciamento sobre seu tratamento, aumentando a qualidade de vida do paciente. Nesse sentido, a sobrecarga do cuidador é diminuída. Porém, o sistema não permite que dados sobre sinais vitais sejam capturados, forçando o cuidador a realizar essa tarefa. Ainda, os autores não informam como o contexto é gerenciado.

Por fim, o trabalho apresentado por Zarghami et al. (2011) traz uma plataforma que fornece serviços para ambientes de *homecare*, considerando a dinamicidade e obstáculos encontrados neste tipo de ambiente. Essa plataforma utiliza dispositivos móveis para interação com o sistema e é capaz de monitorar os sinais vitais do paciente e oferecer lembretes sobre determinada tarefa. Entretanto, o trabalho não apresenta uma maneira de estimular cognitivamente o paciente através de exercícios cognitivos, e assim, não oferece mecanismos para retardar o avanço da doença.

A partir dos trabalhos encontrados na literatura percebe-se a ausência de uma ferramenta que suporte os três pilares do tratamento, como é a proposta desta dissertação. A Tabela 9.1 apresenta um comparativo entre os trabalhos relacionados e a proposta desta dissertação.

Tabela 9.1: Comparativo entre trabalhos relacionados

Projeto	Diminuição de dependência funcional do paciente	Redução de sobrecarga do cuidador	Estimulação cognitiva	Monitoramento de sinais vitais
(RIANO et al., 2009)	X	–	–	X

(TAPIA; CORCHADO, 2009)	–	X (parcialmente)	–	X
(HELAL et al., 2003)	X	X	X (parcialmente)	–
(ZARGHAMI et al., 2011)	X	X	–	X
Proposta apresentada	X	X	X	X

A partir da tabela acima, pode-se compreender que a abrangência da proposta dessa dissertação é maior quando comparada aos trabalhos relacionados, no que diz respeito ao tratamento de pacientes com demência em ambientes *homecare*, uma vez que une as características das demais, além de contemplar as deficiências apresentadas pelas mesmas.

10 CONCLUSÃO

Com o envelhecimento populacional e o conseqüente aumento da incidência de doenças crônico-degenerativas, como a demência, o modelo de saúde tradicional é cada vez mais insuficiente, impulsionando, assim, o cuidado em *homecare*. Através desse tipo de cuidado, um paciente pode receber tratamento em sua casa da mesma maneira que seria feito em um hospital, seja por necessidade ou por vontade do paciente, uma vez que estará em um ambiente familiar.

O cuidado em *homecare*, no entanto, apresenta uma série de desafios. Por se tratar de um ambiente domiciliar, o monitoramento da saúde do paciente tende a tornar-se complexo, uma vez que podem existir outras pessoas circulando no ambiente e pacientes com demência, dependendo do estágio da doença, não conseguem expressar uma situação crítica de saúde e solicitar ajuda. Outro problema diz respeito ao cuidador, o qual pode estar desatendo devido a sobrecarga causada pelo intenso cuidado, ou não possuir a técnica adequada de monitoramento.

A computação pervasiva representa uma ideia interessante para solucionar problemas como estes. Através de sistemas pervasivos dados fisiológicos e atividades do paciente podem ser coletados proativamente, de forma que o sistema se antecipe a alguma situação crítica. Nesse contexto diversos trabalhos são encontrados na literatura, os quais visam monitoramento do paciente. Porém, há uma escassez de ferramentas voltadas à pacientes portadores de demência, especificamente. Esse perfil de paciente difere dos demais no seu tratamento, visto que é necessário estimular as funções físicas e cognitivas do paciente.

Desta forma, este trabalho apresentou a proposta de uma arquitetura para sistemas pervasivos voltados ao cuidado de pacientes portadores de demência em ambientes *homecare*. Um sistema pervasivo desenvolvido com base na arquitetura proposta é capaz de monitorar o paciente de forma proativa e enviar aplicações e lembretes para que o mesmo consiga autogerenciar seu tratamento, dependendo do estado clínico, através de um plano de cuidados e uma base de aplicações cognitivas.

Além da arquitetura, utilizando conceitos de computação pervasiva, foi modelada uma ontologia, a qual representa o conhecimento encontrado neste tipo de ambiente. Para a manipulação da ontologia foram desenvolvidas regras de inferência através da linguagem SWRL e consultas através de SQWRL. A partir das regras o sistema pode inferir novos conhecimentos na ontologia a partir de um contexto em específico. As consultas, por sua vez, são utilizadas para buscar informações na ontologia para que aplicações possam ser enviadas para os usuários.

Para validar a metodologia proposta e esclarecer o funcionamento do sistema, foram apresentados dois casos de estudo simulados, baseados em situações em que sistema deva auxiliar o paciente portador de demência no seu tratamento. O primeiro caso de estudo apresenta a situação onde o paciente deva tomar um medicamento prescrito pelo médico. No entanto, ao invés de um cuidador lembrá-lo, o sistema realiza essa tarefa e assim o paciente consegue autogerenciar seu tratamento. Dessa maneira, o tratamento não fica suscetível a um esquecimento do cuidador, causado pela sobrecarga. Já o segundo caso, trata de mostrar como o sistema pode estimular cognitivamente o paciente, através de uma aplicação executada em seu dispositivo computacional. Foi prescrita um exercício pelo médico, o qual o paciente pode realizar sem depender da ajuda do cuidador, mantendo-se autônomo. Nesse caso, também há uma diminuição das tarefas, e conseqüentemente, sobrecarga do cuidador ao passo em que as funções cognitivas remanescentes são exercitadas em qualquer lugar e a qualquer momento pelo paciente.

Assim, com a abordagem proposta, acredita-se conseguir uma melhoria significativa no tratamento de pacientes com demência em suas casas. Além do auxílio cognitivo, a abordagem prevê que os sistemas pervasivos desenvolvidos a partir dela consigam monitorar o paciente e informar situações críticas de saúde ao cuidador ou médico responsável. É considerado também que o sistema consiga adaptar o ambiente do paciente conforme o contexto e suas necessidades, por exemplo, ligar um ar condicionado quando estiver calor, para não aumentar a pressão arterial do paciente.

No entanto, ainda é preciso aprimorar a abordagem futuramente. Um trabalho futuro pode ser adicionar um módulo de aprendizado à arquitetura. Logo, não será preciso explicitar regras de inferência para cada paciente, mas sim um sistema reagirá de acordo com ações tomadas durante um período de tempo, individualizando cada usuário.

Outra contribuição futura interessante é o desenvolvimento de um escalonador de alertas e tarefas de acordo com o conhecimento médico. Até o momento é possível definir os níveis de alerta para as notificações.

Por fim, a realização de testes através de um protótipo em um ambiente real também seria de grande valia para a proposta. Desta forma, poderia-se analisar o funcionamento do sistema de forma mais precisa e, então, disponibilizá-los para usuários reais.

REFERÊNCIAS

- ALTOVA, G. *Altova XMLSpy 2013 Tutorial*. , [S.l.], 2012. Disponível em <http://www.altova.com/documents/XMLSpyTutorial.pdf>. Acesso em 23 de Abril de 2013.
- ANTONIOU, G.; HARMELEN, F. **Web Ontology Language: owl**. Disponível em <http://www.cs.vu.nl/frankh/postscript/OntoHandbook03OWL.pdf>. Acesso em 10/04/2012.
- BARDAM, J. E. Applications of context-aware computing in hospital work: examples and design principles. **ACM SYMPOSIUM ON APPLIED COMPUTING**, New York, NY, USA, p.1574 – 1579, 2004.
- BASTIANI, E.; SOARES, K.; LIBRELOTTO, G. R. Uma abordagem para monitoramento de pacientes com Alzheimer em ambientes homecare pervasivos. **XII Workshop de Informática Médica**, Brasil, 2012.
- BORST, W. **Construction of engineering ontologies for knowledge sharing and reuse**. 1997. Tese (Doutorado) — University of Twente.
- BRAY, T.; PAOLI, J.; SPERBERG-MCQUEEN, C. M.; MALER, E.; YERGEAU, F. *Extensible Markup Language*. , [S.l.], 2008.
- CAMARA, V. D.; GOMES, S. S.; RAMOS, F.; MOURA, S.; DUARTE, R.; COSTA, S. A.; RAMOS, P. A. F.; LIMA, J. A. M. C.; CAMARA, I. C. G.; SILVA, L. E.; SILVA, A. B.; RIBEIRO, M. C.; FONSECA, A. A. Reabilitação Cognitiva das Demências. **Revista Brasileira de Neurologia**, São Paulo, SP, Brasil, v.45, n.1, p.025–033, 2009.
- CARAMELLI, P.; BARBOSA, M. T. Como diagnosticar as quatro causas mais frequentes de demência? **Revista Brasileira de Psiquiatria**, Brasil, v.24, p.7–10, 2002.
- CHEN, G.; KORTZ, D. A. *A Survey of Context-Aware Mobile Computing Research*. , [S.l.], 2000.
- COPETTI, A.; LOQUES, O.; LEITA, J. C. B. Intelligent Context-Aware Monitoring in Home Care. **2nd International Conference: E-Medical Systems**, „Tunisia, 2008.

CORCHADO, J. M.; PAVÓN, J.; CORCHADO, E.; CASTILHO, L. F. Development of CBR-BDI agents: a tourist guide application. **7th European Conference on Case-based Reasoning**, [S.l.], 2004.

DEY, A.; ABOND, G. The context toolkit: aiding the development of context-aware applications. **Proceedings of Human Factors in Computing Systems**, [S.l.], 2006.

ENGELHARDT, E.; BRUCKI, S. M.; CAVALCANTI, J. L. S.; FORLENZA, O. V.; LAKS, J.; VALE, F. A. C. Tratamento da doença de Alzheimer. **Arquivos de Neuro-Psiquiatria**, São Paulo, SP, Brasil, v.63, n.4, 2005.

FLORIAN, C. A.; SCHRAMM, F. R. Atendimento domiciliar ao idoso: problema ou solução? **Caderndo de Saúde Pública**, Rio de Janeiro, RJ, Brasil, v.20, n.4, p.986–994, 2004.

FOLSTEIN, M.; FOLSTEIN, S.; MCHUGH, P. Mini-Mental State. A practical method for grading the cognitive state of patients for the clinician. **Journal of Psychiatric Research**, [S.l.], v.12, p.189–198, 1975.

FORNARI, L. H. T.; GARCIA, L. P.; HILBIG, A.; FERNANDEZ, L. As diversas faces da síndrome demencial: como diagnosticar clinicamente? **Journal Scientia Medica**, Porto Alegre, RS, Brasil, v.20(2), p.185–193, 2010.

FREITAS, L. O. Uma metodologia para assistir pacientes em ambientes homecare pervasivos. **Dissertação de Mestrado**, Santa Maria, RS, Brasil, 2011.

FREITAS, L. O.; LIBRELOTTO, G. R.; PEREIRA, H. G. G.; KASPER, J.; MARTINI, R. G.; MOZZAQUATRO, B.; PEREIRA, R. T. Applying pervasive computing in an architecture for homecare environments. **Ubiquitous Intelligence & Computing and 9th International Conference on Autonomic & Trusted Computing (UIC/ATC)**, New York, New York, USA, 2012.

FRIESEN, J. Lean Java for Android Development. , New York, NY, USA, 2010. Disponível em http://www.centarzaeducaciju.com/katalog/Learn_Java_for_Android_Development.pdf. Acesso em 02 de Dezembro de 2012.

GASSEN, J. B. Uma metodologia para o uso de ontologias aplicadas à descrição de contexto em ambientes hospitalares pervasivos. **Dissertação de Mestrado**, Santa Maria, RS, Brasil, 2010.

GRAU, B. C.; HORROCKS, I.; MOTIK, B.; PARSIA, B.; PATEL-SCHNEIDER, P.; SATTLER, U. **OWL 2, the next step for OWL**. DOI: <http://dx.doi.org/10.1016/j.bbr.2011.03.031>.

GRUBER, T. R. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. **Formal Ontology in Conceptual Analysis and Knowledge Representation**, Padova, Itália, v.43, n.5-6, p.907–928, 1993.

GUARINO, N. Formal Ontology and Information Systems. In: 1998. **Anais...** IOS Press, 1998. p.3–15.

GUIMARÃES, E. M. P.; SPAGNOL, C. A.; FERREIRA, E.; SALVIANO, M. E. Utilização do plano de cuidados como estratégia de sistematização da assistência de enfermagem. **Ciencia y enfermería**, [S.l.], v.8, n.2, 2002.

GUIMARÃES, F. J. Z. Utilização de ontologias no domínio B2C. **Dissertação de Mestrado**, Rio de Janeiro, RJ, Brasil, 2002.

GUIMARÃES, J.; FONSECA, R.; GARRETT, C. Demência Frontotemporal - Que Entidade? **Acta Médica Portuguesa**, Portugal, v.19, p.319–324, 2006.

HELAL, S.; GIRALDO, C.; KADDOURA, Y.; LEE, C.; ZABADANI, H. E.; MANN, E. Smart phone based cognitive assistant. **UbiHealth 2003: The 2nd International Workshop on Ubiquitous Computing for Pervasive Healthcare Applications**, [S.l.], 2003.

HERRERA, E.; CARAMELLI, P.; NITRINI, R. Estudo epidemiológico populacional de demência na cidade de Catanduva, Estado de São Paulo, Brasil. **Revista Brasileira de Psiquiatria**, Brasil, v.25, p.70–3, 1998.

HODGINS, D.; BERTSCH, A.; POST, N.; FRISCHHOLZ, M.; VOLCKAERTS, B.; SPENSLEY, J.; WASIKIEWICZ, J. M.; HIGGINS, H.; STETTEN, F.; KENNEY, L. Healthy Aims:developing new medical implants and diagnostic equipment. **IEEE Pervasive Computing**, [S.l.], 2008.

HORRIDGE, M.; KNUBLAUCH, H.; RECTOR, A.; STEVENS, R.; WROE, C. A Practical Guide To Building OWL Ontologies Using The Protégé-OWL Plugin-OWL Plugin and CO-ODE tools. , [S.l.], 2004. Disponível em http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/resources/ProtegeOWLTutorialP3_v1_0.pdf. Acesso em 20 de Janeiro de 2013.

HORROCKS, I.; HARMELEN, F. v.; PATEL-SCHNEIDER, P.; BERNERS-LEE, T.; BRICKLEY, D.; CONNOLLY, D.; DEAN, M.; DECKER, S.; FENSEL, D.; FIKES, R.; HAYES, P.; HEFLIN, J.; LASSILA, O.; MCGUINNESS, D.; STEIN, L. A. DAML+OIL. , [S.l.], 2001. Disponível em <http://www.daml.org/2001/03/daml+oil-index.html>. Acesso em 22 de Dezembro de 2012.

HORROCKS, I.; PATEL-SCHNEIDER, P. F.; BOLEYM, H.; TABET, S.; GROSOFF, B.; DEAN, M. **SWRL: a semantic web rule language combining owl and ruleml**. Disponível em <http://www.w3.org/Submission/SWRL/>. Acesso em 28/11/2012.

JUPP, S.; MOULTON, G.; RECTOR, A.; STEVENS, R.; WROE, C. **A Practical Guide To Building OWL Ontologies Using Prétége 4 and CO-ODE Tools - Edition 1.1**. Disponível em <http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/>. Acesso em 16/08/2012.

LASSILA, O.; SWICK, R. Resource Description Framework (RDF) Model and Syntax Specification. , [S.l.], 1999. Disponível em <http://www.w3.org/TR/1999/REC-rdf-syntax-19990222>. Acesso em 14/04/2012.

LIBRELOTTO, G. R.; GASSEN, J. B.; FREITAS, L. O.; SILVEIRA, M. C.; SILVA, F. L.; AUGUSTIN, I.; HENRIQUES, P. R. Uma Ontologia aplicada a um Ambiente Pervasivo Hospitalar. **8a Conferência da associação portuguesa de sistemas de informação**, Setubal, Portugal, 2008.

LIMA, J. C.; CARVALHO, C. L. Extensible Markup Language (XML). **Relatório Técnico**, [S.l.], 2005.

LIMA, J. C.; CARVALHO, C. L. Ontologias - OWL (Web Ontology Language). **Relatório Técnico**, [S.l.], 2005.

LOPES, J. L. B. Sensibilidade ao Contexto na Computação Pervasiva: avaliando o uso de ontologias. , [S.l.], 2006.

LOUREIRO, A. A. F.; OLIVEIRA, R. A. R.; SILVA, T. R. B.; JÚNIOR, W. R. P.; OLIVEIRA, L. B. R.; MOREIRA, R. A.; SIQUEIRA, R. G.; ROCHA, B. P. S.; RUIZ, L. B. Computação Ubíqua Ciente de Contexto: desafios e tendências. **27º Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, SBRC**, [S.l.], 2009.

MARTINI, R. G. Uma abordagem para a personalização automática de interfaces de usuário para dispositivos móveis em ambientes pervasivos. **Dissertação de Mestrado**, Santa Maria, RS, Brasil, 2012.

MCGEE-LENNON, M. R. Requirements Engineering for Home Care Technology. **Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems**, [S.l.], p.1439–1442, 2008.

MCGEE-LENNON, M. R.; GRAY, P. D. Including Stakeholders in the Design of Home Care Systems: identification and categorisation of complex user requirements. **INCLUDE Conference**, [S.l.], 2007.

NETO, J. G.; TAMELINI, M. G.; FORLENZA, O. V. Diagnóstico Diferencial das Demências. **Revista de Psiquiatria Clínica**, São Paulo, SP, Brasil, v.32, n.3, p.119–130, 2005.

NOY, F. N.; MCGUINNESS, D. L. Ontology development 101: a guide to create your first ontology. **Stanford Knowledge Systems Laboratory Technical Report**, [S.l.], 2001.

O'CONNOR, M. J.; DAS, A. SQWRL: a query language for owl. **OWLED**, [S.l.], 2008.

RAMOS, A.; STEIN, A.; CASTRO, E.; CHAVES, M. L. F.; OKAMOTO, I.; NITRINI, R. **Demência no Idoso: diagnóstico na atenção primária à saúde**. 2009.

RIAÑO, D.; REAL REAL, F.; CAMPANA, F.; ERCOLANI, S.; ANNICCHIARICO, R. An Ontology for the Care of the Elder at Home. , [S.l.], 2009.

SAHA, D.; MUKHERJEE, A. Pervasive Computing: a paradigm for the 21st century. **Computer**, [S.l.], p.25–31, 2003.

SCHLINDWEIN-ZANINI, R. Demência no idoso: aspectos neropsicológicos. **Revista Neurociências**, São Paulo, SP, Brasil, p.220–226, 2005.

SMID, J. **Demências**. Disponível em <http://www.medicinanet.com.br/conteudos/revisoes/1341/demencias.htm> Acesso em 06/06/2012.

SOWA, J. F. **Building, sharing and merging onologies**. Disponível em <http://users.bestweb.net/sowa/ontology/ontosar.htm>. Acesso em 15/05/2012.

- SOYLU, A.; CAUSMAECKER, P. d.; DESMET, P. Context and Adaptivity in Pervasive Computing Environments: links with software engineering and ontological engineering. **Journal of Software**, [S.l.], v.4, n.9, p.992–1013, 2009.
- SPERBERG-MCQUEEN, C. M.; THOMPSON, H. XML Schema. , [S.l.], 2004. Disponível em <http://www.w3.org/XML/Schema>. Acesso em 23 de Abril de 2013.
- STUDER R.; BENJAMINS, R. F. D. Knowledge Engineering: principles and methods. **IEEE Transactions on Data and Knowledge Engineering**, [S.l.], 1998.
- TAPIA, D. I.; CORCHADO, J. M. An Ambient Intelligence Based Multi-Agent System for Alzheimer Health Care. **International Journal of Ambient Computing and Intelligence**, [S.l.], 2009.
- TATSCH, M. F.; NITRINI, R.; NETO, M. R. L. Demência com corpúsculos de Lewy: uma entidade distinta com tratamento específico? **Revista Brasileira de Psiquiatria**, Brasil, v.24(3), p.156–156, 2002.
- TEIXEIRA-JR, A. L.; CARDOSO, F. Demência com corpos de Lewy: abordagem clínica e terapêutica. **Revista Neurociências**, São Paulo, SP, Brasil, v.13(1), p.028–033, 2005.
- VAQUERO, L. M.; RODERO-MERINO, L.; CACERES, J.; LINDNER, M. A break in the clouds: towards a cloud definition. **SIGCOMM Comput. Commun. Rev.**, New York, NY, USA, v.39, p.50–55, 2008.
- WANG, F.; TURNER, K. J. Towards personalised home care systems. **1st international conference on Pervasive Technologies Related to Assistive Environments**, New York, NY, USA, 2008.
- WEISER, M. The Computer of the 21st Century. **Scientific American**, [S.l.], 1991.
- WHO. Dementia: a public health priority. , [S.l.], 2012. Disponível em http://whqlibdoc.who.int/publications/2012/9789241564458_eng.pdf. Acesso em 20/10/2012.
- WOOD, A.; STANKOVIC, J.; VIRONE, G.; SELAVO, L.; HE, Z.; CAO, Q.; DOAN, T.; WU, Y.; FANG, L.; STOLERU, R. Context-Aware Wireless Sensor Networks for Assisted-Living and Residential Monitoring. **Ieee Network**, [S.l.], v.22, 2008.

YAMIN, A. **Arquitetura para um Ambiente de Grade Computacional Direcionado às Aplicações Distribuídas, Móveis e Conscientes do Contexto da Computação Pervasiva**. 2004.

Tese (Doutorado) — Universidade Federal do Rio Grande do Sul.

YAMIN, A.; BARBOSA, J. L.; AUGUSTIN, I.; SILVA, L. C. d.; REAL, R. A.; GEYER, C. F. R.; CAVALHEIRO, G. G. H. Towards Merging Context-Aware, Mobile and Grid Computing.

IJHPCA, [S.l.], 2003.

ZARGHAMI, A.; ESLAMI, M. Z.; SAPKOTA, M.; SINDEREN, M. Toward Dynamic Service Provisioning in the Homecare Domain. **Pervasive computing technologies for healthcare**,

[S.l.], 2011.

APÊNDICE A PROPRIEDADES DE DADOS DA ONTOLOGIA

Nessa seção serão apresentadas as propriedades de dados mapeadas na ontologia de *home-care*. No entanto, as propriedades *identificador*, *descricao* e *necessario* não estão descritas na Tabela A.1, visto que estão presentes em todas as classes.

Tabela A.1: Tabela de propriedade de dados da ontologia

Classe	Propriedade	Tipo de dado	Valores
Cômodo	luminosidade	float	0-100
	temperatura	float	–
	umidade	float	0-100
Móvel	tipoMovel	string	cadeira, sofá, balcão, cama, mesa ou outro
Objeto	tipoObjeto	string	quadro, porta-retrato, vaso ou outro
AtividadePaciente	inicioAtividade	dateTime	–
	status	boolean	true ou false
	terminoAtividade	dateTime	–
AtividadeCognitiva	tipoAtividadeCognitiva	string	memória, raciocínio, atenção ou linguagem
AtividadeFisica	intensidade	string	baixo, moderado ou alto
	necessitaAuxilio	boolean	true ou false
	tipoAtividadeFisica	string	caminhar, alongar, correr ou outro
AtividadeVidaDiaria	tipoAVD	string	beber, comer, dormir, acordar, banheiro, medicamento, vestir, sentar, levantar, banho, acordar, beber ou outro
DeficienciaVisual	grauDefVisual	float	leve, moderada, grave, profunda ou total
DeficienciaCognitiva	grauDefCognitiva	int	baixo, moderado, alto
DeficienciaFisica	grauDefFisica	string	baixo, moderado ou alto
DeficienciaAuditiva	grauDefAuditiva	string	leve, moderada, grave, profunda ou total
Doença	icd	string	–
Demência	curso	string	estável, progressivo ou reversível

	estágio	string	inicial, intermediário ou avançado
Equipamento	horaAtivacao	time	–
	horaDesligamento	time	–
	status	boolean	true ou false
	data	date	–
	diagnóstico	string	–
	observação	string	–
	visualizado	boolean	true ou false
	horario	time	–
Medicamento	quantidadeDisponivel	int	–
	posologia	string	–
Notificacao	horaNotificacao	dateTime	–
	mensagem	string	–
	visualizada	boolean	true ou false
Alerta	prioridade	string	normal ou alta
Pessoa	nome	string	–
	dataNascimento	date	–
Medico	CRM	string	–
	especialidade	string	geriatria, geral, gastroenterologia, neurologia, cardiologia ou outro
Enfermeiro	COREN	string	–
TecnicoEnfermagem	COREN	string	–
Visitante	grauAfinidade	string	baixo, médio ou alto
SinaisVitais	frequenciaCardiaca	float	–
	frequenciaRespiratoria	float	–
	satOxigenio	float	–
	pressaoDiastolica	float	–
	pressaoSistolica	float	–
	temperatura	float	–
Sintoma	icd	string	–
AplicativoMovel	executando	boolean	true ou false
	versão	string	–
OntologiaCAtual	dataCriacao	date	–
	dataArmazenamento	date	–
PEP	versão	string	–
	dataAtualizacao	date	–
	executando	date	–

APÊNDICE B PROPRIEDADES DE OBJETOS DA ONTOLOGIA

Este apêndice apresenta as propriedades de objetos (relacionamentos) presentes na ontologia, visualizados na B.1.

Tabela B.1: Tabela de propriedades de objetos da ontologia

Classe	Relacionamento	Classe	Relação Inversa
AplicativoMovel	comunica	ModEntrada	–
	recebe	Notificacao	–
	apresenta	Notificacao	–
	instaladoEm	Dispositivo Movel	possui
AtividadeFisica	podeInfluenciar	SinaisVitais	influenciadoPor
Comodo	contem	Equipamento, Movel, Objeto e Pessoa	estaEm
Doenca	temSintoma	Sintomas	eSintomaDe
	podeSerCausaDe	Sindrome	podeSerExpressaoDe
	podeCausar	Deficiencia	causadaPor
Objeto	estaEm	Movel	contem
Enfermeiro	assiste	Paciente	assistidoPor
	auxilia	Paciente	auxiliadoPor
	administra	Medicamento	administradoPor
Exame	pertenceA	Paciente	–
Medicamento	estaEm	Dispenser	contem
	indicadoPara	Paciente	–
	combate	Sintoma	–
Medico	prescreve	Medicamento	prescritoPor
	recomenda	AtividadePaciente	–
	requisita	Exame	requisitadoPor
	visualiza	Exame	visualizadoPor
	responsávelPor	Paciente	sobResponsabilidade
	indica	Médico	–
ModOntoHC	comunica	OntologiaCAtual, ModNuvem, ModNotific e ModPICuidados	–
ModEntrada	comunica	ModOntoHC e ModPICuidados	–
	comunica	ModOntoHC e ModPICuidados	–
ModPICuidados	comunica	ModOntoHC	–
ModNotific	comunica	Equipamento	–
	envia	Notificacao	–
ModNuvem	comunica	OntologiaHC, PEP	–
Notificacao	dispDestino	Equipamento	–
Paciente	realiza	Exame	realizadoPor
	realiza	AtividadePaciente	–

	possui	Doença, Deficiência, e Sinais Vitais	–
	apresenta	Sintoma	–
Pessoa	possui	Dispositivo Computacional	pertenceA
	utiliza	Equipamento	utilizadoPor
Responsável	auxilia	Paciente	auxiliadoPor
	assiste	Paciente	acompanhadoPor
Sensor	monitora	Paciente e Comodo	monitoradoPor
Síndrome	temSintoma	Sintoma	eSintomaDe
Técnico Enfermagem	auxilia	Paciente	auxiliadoPor
	assiste	Paciente	assistidoPor
Visitante	assiste	Paciente	assistidoPor

APÊNDICE C RESTRIÇÕES DE PROPRIEDADES NA ONTOLOGIA

Tabela C.1: Tabela de restrições da ontologia

Domínio	Propriedade	Tipo de restrição	Range
Comodo	contem	someValuesFrom	Móvel, Objeto, Pessoa e Equipamento
Móvel	estaEm	allValuesFrom	Comodo
Objeto	estaEm	someValuesFrom	Móvel, Comodo
Equipamento	estaEm	allValuesFrom	Comodo
Dispositivo Movei	pertenceA	allValuesFrom	Medico, Enfermeiro, Responsavel e Paciente
	pertenceA	cardinality	1
Dispenser	contem	allValuesFrom	Medicamento
Exame	pertenceA	allValuesFrom	Paciente
Exame	pertenceA	allValuesFrom	Paciente
Pessoa	estaEm	someValuesFrom	Comodo
Paciente	possui	someValuesFrom	Doenca, SinaisVitais
	apresenta	allValuesFrom	Sintoma
SinaisVitais	pertenceA	allValuesFrom	Paciente
AplicativoMovei	apresenta	allValuesFrom	Notificacao
	comunica	allValuesFrom	ModEntrada
ModOntoHC	comunica	allValuesFrom	OntologiaCAAtual, ModNuvem, ModPICuidados e ModNotific
	gera	allValuesFrom	Notificacao
ModNotific	comunica	allValuesFrom	Equipamento Conforto, Dispositivo Computacional e ModOntoHC
ModEntrada	comunica	allValuesFrom	ModOntoHC e ModPICuidados
ModPICuidados	comunica	allValuesFrom	ModOntoHC
ModNuvem	comunica	allValuesFrom	ModOntoHC, OntologiaHC e PEP
	gera	allValuesFrom	OntologiaCA

APÊNDICE D CONSULTAS E INFERÊNCIAS

Nesta seção serão apresentadas as consultas e regras de inferências desenvolvidas para a manipulação da ontologia. As regras de inferência são baseadas tanto na lógica do sistema, quanto nas diretrizes médicas obtidas durante as reuniões com profissionais da saúde, como explicado na Seção 6.1. Por sua vez, as consultas são utilizadas para buscar alguma informação relevante ao contexto atual.

É importante ressaltar que as regras podem ser modificadas pelo médico responsável, através do aplicativo móvel que está executando em seu dispositivo, conforme o estado de saúde do paciente em questão. Também deve ser considerado que as regras se aplicam aos pacientes em geral, ou seja, devem ser modificadas para cada caso em específico. A seguir são listadas as regras desenvolvidas obedecendo a semântica e sintaxe da linguagem SWRL.

- Regra para envio de alerta em caso de hipertensão arterial do paciente:

```
Paciente(?p) ^ SinaisVitais(?sv) ^ possui(?p, ?sv) ^ pressaoDialostica(?sv, ?pd) ^
pressaoSistolica(?sv, ?ps) ^ swrlb:greaterThanOrEqual(?pd, 100.0) ^
swrlb:greaterThanOrEqual(?ps, 140.0) ^ acompanha(?c, ?p)^ DispositivoMovel(?dm) ^
pertenceA(?dm, ?c) ^ Alerta(?ale) -> mensagem(?ale, "Situação de risco: Hipertensão")
^ prioridade (?ale,"Alta") ^ dispDest(?ale, ?dm)
```

- Regra para envio de alerta em caso de hipotensão arterial do paciente:

```
Paciente(?p) ^ SinaisVitais(?sv) ^ possui(?p, ?sv) ^ pressaoDialostica(?sv, ?pd) ^
pressaoSistolica(?sv, ?ps) ^ swrlb:lessThanOrEqual(?pd, 100.0) ^
swrlb:lessThanOrEqual(?ps, 140.0) ^ acompanha(?c, ?p)^ DispositivoMovel(?dm) ^
pertenceA(?dm, ?c) ^ Alerta(?ale) -> mensagem(?ale, "Situação de risco: Hipotensão")
^ prioridade (?ale,"Alta") ^ dispDest(?ale, ?dm)
```

- Regra para envio de alerta em caso de alta temperatura corporal (febre)

```
Paciente(?p) ^ SinaisVitais(?sv) ^ possui(?p, ?sv) ^ temperatura(?sv, ?t) ^
swrlb:greaterThan(?t, 37.2) ^ acompanha(?c, ?p)^ DispositivoMovel(?dm) ^
pertenceA(?dm, ?c) ^ Alerta(?ale) -> mensagem(?ale, "Situação de risco: Febre") ^
prioridade (?ale,"Alta") ^ dispDest(?ale, ?dm)
```

- Regra para envio de alerta em caso de baixa temperatura corporal (hipotermia)

```
Paciente(?p) ^ SinaisVitais(?sv) ^ possui(?p, ?sv) ^ temperatura(?sv, ?t) ^
swrlb:lessThan(?t, 34.0) ^ acompanha(?c, ?p)^ DispositivoMovel(?dm) ^
pertenceA(?dm, ?c) ^ Alerta(?ale) -> mensagem(?ale, "Situação de risco: Hipotermia")
^ prioridade (?ale,"Alta") ^ dispDest(?ale, ?dm)
```

- Regra para envio de alerta em caso de baixa frequência de batimentos cardíacos (bradicardia)

```
Paciente(?p) ^ SinaisVitais(?sv) ^ possui(?p, ?sv) ^ frequenciaCardiaca(?sv, ?fc) ^
swrlb:lessThan(?fc, 60) ^ acompanha(?c, ?p)^ DispositivoMovel(?dm) ^
pertenceA(?dm, ?c) ^ Alerta(?ale) -> mensagem(?ale, "Situação de risco: Bradicardia") ^
prioridade (?ale,"Normal") ^ dispDest(?ale, ?dm)
```

- Regra para envio de alerta em caso de alta frequência de batimentos cardíacos (taquicardia):

```
Paciente(?p) ^ SinaisVitais(?sv) ^ possui(?p, ?sv) ^ frequenciaCardiaca(?sv, ?fc) ^
swrlb:greaterThan(?fc, 100) ^ acompanha(?c, ?p)^ DispositivoMovel(?dm) ^
pertenceA(?dm, ?c) ^ Alerta(?ale) -> mensagem(?ale, "Situação de risco: Taquicardia") ^
prioridade (?ale,"Normal") ^ dispDest(?ale, ?dm)
```

- Regra para envio de alerta em caso de baixa saturação de oxigênio:

```
Paciente(?p) ^ SinaisVitais(?sv) ^ possui(?p, ?sv) ^ satOxigenio(?sv, ?oxig) ^
swrlb:lessThan(?oxig, 92) ^ acompanha(?c, ?p)^ DispositivoMovel(?dm) ^
pertenceA(?dm, ?c) ^ Alerta(?ale) ->
mensagem(?ale, "Situação de risco: Saturação de Oxigênio Baixa") ^
prioridade (?ale,"Alta") ^ dispDest(?ale, ?dm)
```

- Regra para ligar aparelho de ar condicionado quando a temperatura for superior ou igual a 28°:

```
Comodo(?com) ^ temperatura(?com, ?temp) ^ swrlb:greaterThanOrEqual(?temp, 28) ^
ArCondicionado(?ar)^ ? estaEm(?ar, ?com) ^ temperatura(?ar, 25.5) -> status(?ar, true)
```

- Regra para ligar aparelho de ar condicionado quando a temperatura for inferior ou igual a 21°:

```
Comodo(?com) ^ temperatura(?com, ?temp) ^ swrlb:lessThanOrEqual(?temp, 21) ^
ArCondicionado(?ar)^ ? estaEm(?ar, ?com) ^ temperatura(?ar, 25.5) -> status(?ar, true)
```

- Regra para ligar aparelho de conforto (som) a partir de um sintoma de agitação:

```
Paciente(?pac) ^ apresenta(?pac, ?sin) ^ icd(?sin, ?ic) ^ swrlb:equal(?ic, "R45.1")
^ Som(?som) ^ estaEm(?som, ?como) ^ estaEm(?pac, ?como) -> status(?som, true)
```

- Consulta que retorna o *dispenser* e cômodo, respectivamente, onde está um determinado medicamento:

```
Medicamento(?m) ^ nome(?m, ?n) ^ swrlb:contains(?n, "Nome") ^ estaEm(?m, ?disp)
^ Dispenser(?disp) ^ estaEm(?disp, ?comodo) -> sqwrl:select(?disp, ?comodo)
```

- Consulta que retorna se o paciente está realizando alguma atividade (cognitiva, física ou atividade de vida diária):

```
Paciente(?pac) ^ realiza (?pac, ?atv) ^ AtividadePaciente(?atv) -> sqwrl:select(?atv)
```

- Consulta que retorna um sinal vital em específico do paciente (e.g. temperatura):

```
Paciente(?pac) ^ possui (?pac, ?sv) ^ SinaisVitais(?sv) ^ temperatura(?sv, ?t) ->
sqwrl:select(?t)
```

- Consulta que retorna um sinal vital em específico do paciente (e.g. temperatura):

```
Paciente(?pac) ^ possui (?pac, ?sv) ^ SinaisVitais(?sv) ^ temperatura(?sv, ?t) ->
sqwrl:select(?t)
```