



Programa Interdisciplinar de Pós-Graduação em
Computação Aplicada
Mestrado Acadêmico

Henrique Damasceno Vianna

U'DUCTOR: Um Modelo para Cuidado Ubíquo de Doenças
Crônicas não Transmissíveis

São Leopoldo, 2013

UNIVERSIDADE DO VALE DO RIO DOS SINOS — UNISINOS
UNIDADE ACADÊMICA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA
NÍVEL MESTRADO

HENRIQUE DAMASCENO VIANNA

U'DUCTOR: UM MODELO PARA CUIDADO UBÍQUO DE DOENÇAS CRÔNICAS NÃO
TRANSMISSÍVEIS

SÃO LEOPOLDO
2013

Henrique Damasceno Vianna

U'DUCTOR: UM MODELO PARA CUIDADO UBÍQUO DE DOENÇAS CRÔNICAS NÃO
TRANSMISSÍVEIS

Dissertação apresentada como requisito parcial
para a obtenção do título de Mestre pelo
Programa de Pós-Graduação em Computação
Aplicada da Universidade do Vale do Rio dos
Sinos — UNISINOS

Orientador:
Prof. Dr. Jorge Luis Victória Barbosa

São Leopoldo
2013

Ficha catalográfica

V617u Vianna, Henrique Damasceno
U'Ductor: um modelo para cuidado ubíquo de doenças
crônicas não transmissíveis / por Henrique Damasceno
Vianna. – 2013.
101 f. : il., 30cm.

Dissertação (mestrado) — Universidade do Vale do Rio
dos Sinos, Programa de Pós-Graduação em Computação
Aplicada, 2013.
Orientação: Prof. Dr. Jorge Luis Victória Barbosa.

1. Computação ubíqua. 2. Gerenciamento de doenças
crônicas. I. Título.

CDU 004.75.057.5

Catálogo na Fonte:
Bibliotecária Vanessa Borges Nunes - CRB 10/1556

(Esta folha serve somente para guardar o lugar da verdadeira folha de aprovação, que é obtida após a defesa do trabalho. Este item é obrigatório, exceto no caso de TCCs.)

À Cristiane e aos meus pais.

AGRADECIMENTOS

Obrigado Cristiane pela calma e paciência, e por lembrar que em certos momentos era necessário mudar o foco. Pais, obrigado pelo suporte e carinho. Dona Cristina e Seu João, obrigado por toda ajuda dada no período do mestrado. Um agradecimento especial ao Professor Jorge Barbosa, por toda confiança e atenção despendida nesses dois anos. Agradeço o colega Rodrigo Remor pela parceria durante o curso. Gostaria também de agradecer aos sujeitos de pesquisa, os quais concederam seu tempo e valiosas sugestões, e a CAPES pelo apoio financeiro através do Programa de Suporte à Pós-Graduação de Instituições de Ensino Particulares (PROSUP).

“A tecnologia só é tecnologia para quem nasceu antes dela ter sido inventada.”.
(Alan Kay)

RESUMO

De acordo com a Organização Mundial de Saúde, doenças cardíacas, doenças respiratórias crônicas, câncer e diabetes são as doenças crônicas não transmissíveis mais comuns, assim como uma das principais causas da mortalidade no mundo. Nesses casos é necessário mais do que o engajamento do paciente no auxílio ao cuidado destas doenças. O apoio da comunidade e das organizações de saúde também é desejável. Estes devem apoiar os pacientes em suas atividades de autogerenciamento, fazendo-os sentirem-se confiantes e motivados. A computação ubíqua dá condições para ajudar os pacientes de doenças crônicas na gestão de suas atividades, oferecendo-lhes apoio a qualquer hora, em qualquer lugar. O presente trabalho apresenta o U'Ductor, um modelo para cuidado ubíquo de doenças crônicas não transmissíveis, cujo objetivo é facilitar a integração entre pacientes e recursos da comunidade e organizações de saúde. O U'Ductor dá um passo a frente em relação aos trabalhos estudados ao integrar pacientes, membros da comunidade e organizações de saúde, e recursos da comunidade e organizações de saúde. Tais características não foram exploradas nos trabalhos estudados da maneira como é feita no U'Ductor. Um protótipo do modelo foi avaliado por pacientes crônicos que deram pareceres positivos em relação a sua aplicabilidade nas atividades de cuidado de doenças crônicas não transmissíveis.

Palavras-chave: Computação Ubíqua. Gerenciamento de Doenças Crônicas.

ABSTRACT

Accordingly with the World Health Organization, heart disease, chronic respiratory diseases, cancer and diabetes are the most common non-communicable diseases and one of the leading causes of the mortality in the world. In such cases we need more than patient engagement to help to control of the disease, community and health organizations support is also desirable. These roles must support the patients self-management activities, making them feel confident and motivated. This support can be accomplished by ubiquitous computing. The ubiquitous computing gives conditions to help chronic diseases patients in the management of their activities, offering support to them anytime, anywhere. This work presents U'Ductor, a model for supporting ubiquitous non-communicable disease care, whose goal is to help the integration between patient and community resources. The U'Ductor gives a step forward in relation to the studied related works by integrating patients, community resources and community members. Those features were not completely explored in the studied works in the way it's employed in U'Ductor. An implementation of the model was evaluated by chronic patients, which had given a positive feedback about it.

Keywords: Ubiquitous Computing. Chronic Disease Management.

LISTA DE FIGURAS

Figura 1:	The Chronic Care Model	30
Figura 2:	Cuidados Inovadores para Condições Crônicas	31
Figura 3:	Componentes móveis do MAHI	37
Figura 4:	Interface de uso do WD	39
Figura 5:	Principais componentes do Chronius	40
Figura 6:	Framework conceitual de gerenciamento de tratamento medicamentoso	41
Figura 7:	Arquitetura geral do sistema	42
Figura 8:	Plataforma ERMHAN	43
Figura 9:	Exemplo do modelo de entidades do Continuum	50
Figura 10:	Exemplo de organização de elementos do U'Ductor	51
Figura 11:	Componentes do middleware ConnDuctor	52
Figura 12:	Arquitetura do componente de Módulos Executáveis	53
Figura 13:	Arquitetura do componente de Proxy	53
Figura 14:	Arquitetura do componente de Recursos Compartilhados	54
Figura 15:	Arquitetura do componente de Contexto	55
Figura 16:	Linha de tempo de histórico de contexto	56
Figura 17:	Arquitetura do componente Mensageiro	58
Figura 18:	Arquitetura do componente Controle de Acesso	58
Figura 19:	Arquitetura do componente Local	59
Figura 20:	Arquitetura do componente Nodos	60
Figura 21:	Arquitetura da Interface REST	61
Figura 22:	Estrutura hierárquica dos serviços REST do ConnDuctor	62
Figura 23:	Requisição de um novo NL pelo ConnDuctor	64
Figura 24:	Registro de um NP em um NL	64
Figura 25:	Requisitos principais do ChronicDuctor	65
Figura 26:	Legenda de componentes gráficos da metodologia Prometheus	66
Figura 27:	Diagrama da Visão Geral de Análise	67
Figura 28:	Diagrama da Visão Geral de Sistema	68
Figura 29:	Modelo Ontológico ConnDuctor	70
Figura 30:	Modelo Ontológico de Recursos - ChronicDuctor	71
Figura 31:	Modelo Ontológico de Planos - ChronicDuctor	73
Figura 32:	Planta baixa do cenário de simulação	78
Figura 33:	Formulário de construção de plano de cuidado	78
Figura 34:	Telas do aplicativo ChronicDuctor	80
Figura 35:	Resultado geral de utilidade percebida e facilidade de uso percebida	82
Figura 36:	Localização dos Recursos	84

LISTA DE TABELAS

Tabela 1:	Relação fatores de riscos x prevenção/controle	32
Tabela 2:	Comparação de suportes oferecidos pelos trabalhos relacionados	48
Tabela 3:	Contextos armazenados	56
Tabela 4:	Trilha inferida	57
Tabela 5:	Mapeamento das URIs de Serviços do ConnDuctor	63
Tabela 6:	Condição Crônica x Sujeito	77
Tabela 7:	Tipos de recursos encontrados no ambiente de simulação	77
Tabela 8:	Questionário de avaliação	81
Tabela 9:	Plano de cuidado do Usuário	83
Tabela 10:	Objetivos de cuidado do Usuário	83
Tabela 11:	Mapeamento dos Recursos Oferecidos pelos Locais	84
Tabela 12:	Total de contextos coletados	86
Tabela 13:	Roteiro do teste de bateria	88
Tabela 14:	Dados coletados sobre consumo de bateria	89
Tabela 15:	Suportes Oferecidos pelo U'Ductor	94

LISTA DE SIGLAS

ANATEL	Agência Nacional de Telecomunicações
BAN	Body Area Network
CCM	Chronic Care Model
CPU	Central Processing Unit
DCNT	Doenças Crônicas Não Transmissíveis
GPS	Global Position System
HIV	Human Immunodeficiency Virus
HTTP	Hypertext Transfer Protocol
IBGE	Instituto Brasileiro de Geografia e Estatística
JRE	Java Runtime Environment
MBU	Mobile Base Unit
MIME	Multipurpose Internet Mail Extensions
NL	Nodo Local
NP	Nodo Pessoal
OMS	Organização Mundial de Saúde
OWL	Web Ontology Language
PC	Personal Computer
PDA	Personal Digital Assistant
PDF	Portable Document Format
P2P	Peer-to-Peer
REST	Representational State Transfer
SMS	Short Message Service
SMTP	Simple Mail Transfer Protocol
SOAP	Simple Object Access Protocol
SRP-NL	Serviço de Resolução Posição <-> Nodo Local
SSL	Secure Socket Layer
UML	Unified Modeling Language
URI	Uniform Resource Identifier
WBAN	Wireless Body Area Network
WD	Wellness Diary
WSDL	Web Services Description Language

SUMÁRIO

1 INTRODUÇÃO	23
1.1 Motivação	23
1.2 Definição do Problema	24
1.3 Objetivos	25
1.4 Metodologia	25
1.5 Organização do Texto	26
2 SAÚDE UBÍQUA E CUIDADO DE DOENÇAS CRÔNICAS	27
2.1 Computação Ubíqua	27
2.2 Modelos de Cuidados para Condições Crônicas	28
2.2.1 Modelo CCM	29
2.2.2 Modelo CICC	30
2.2.3 Considerações sobre os modelos CCM e CICC	31
2.3 Fatores de Risco de DCNT	31
2.4 Saúde Ubíqua	32
2.4.1 Aplicações Hospitalares	33
2.4.2 Monitoramento de Pacientes	33
2.4.3 Suporte ao Bem-Estar	34
2.5 Considerações sobre o Capítulo	34
3 TRABALHOS RELACIONADOS	37
3.1 MAHI	37
3.2 Wellness Diary	38
3.3 CHRONIOUS	39
3.4 Framework para gerenciamento de medicação de Koutkias et-al.	40
3.5 Sistema de Gerenciamento de DCNT de Paganelli e Giuli	42
3.6 Comparativo entre os trabalhos estudados	45
4 MODELO U'DUCTOR	49
4.1 Modelo Organizacional do U'Ductor	49
4.2 Middleware ConnDuctor	52
4.2.1 Componentes do Middleware ConnDuctor	52
4.3 ReDuctor	62
4.4 ChronicDuctor	63
4.4.1 Requisitos do ChronicDuctor	64
4.4.2 Modelagem do ChronicDuctor	66
4.5 Modelo Ontológico do U'Ductor	69
4.5.1 Ontologia do ConnDuctor	69
4.5.2 Ontologia de Recursos do ChronicDuctor	71
4.5.3 Ontologia de Plano do ChronicDuctor	72
4.6 Considerações sobre o capítulo	74
5 AVALIAÇÃO DO MODELO	75
5.1 Aspectos de Implementação	75
5.2 Teste de aceitação do ChronicDuctor	76
5.3 Teste de uso estendido do ChronicDuctor	82
5.4 Avaliação de Consumo de Bateria	88

6	CONSIDERAÇÕES FINAIS	91
6.1	Conclusões	91
6.2	Contribuições	92
6.3	Trabalhos Futuros	93
	REFERÊNCIAS	97

1 INTRODUÇÃO

1.1 Motivação

Segundo dados da Organização Mundial de Saúde (OMS), no ano de 2008 morreram em torno de 36 milhões de pessoas devido a doenças crônicas não transmissíveis (DCNT) em todo o mundo (OMS, 2011), e segundo a própria organização, 80% das mortes prematuras causadas por algumas destas doenças podem ser prevenidas (OMS, 2005). A OMS considera doenças cardiovasculares, cânceres, diabetes e doenças crônicas pulmonares como sendo as quatro principais doenças crônicas existentes, e de acordo com a organização, a carga destas doenças vem crescendo em países de baixa renda (OMS, 2005). No Brasil, o relatório de 2010 do sistema Vigitel¹ apontou que 23,3% dos entrevistados haviam sido diagnosticados com hipertensão e 6,3% dos entrevistados relataram ter diagnóstico de diabetes (SAÚDE, 2011).

Em sua maioria, as doenças crônicas são causadas por hábitos, tais como, sedentarismo, tabagismo, entre outros, que resultam em “alterações metabólicas/fisiológicas”, como por exemplo, hipertensão arterial, sobrepeso, obesidade. Tanto os hábitos, quanto os resultados destes hábitos compõem os fatores de riscos que devem ser controlados como forma de prevenir casos destas doenças (OMS, 2005).

O tratamento de DCNT deve ser feito de forma contínuo já que a maioria dessas doenças não possui cura. Dessa forma, é preciso que o paciente tenha consciência de suas condições, siga o tratamento determinado pelo responsável médico e saiba agir quando necessário. Em determinadas ocasiões o paciente pode não ter a confiança necessária para realizar determinadas atividades e necessite da colaboração de alguém com a experiência na atividade em questão (WAGNER et al., 2001; WAGNER; GROVE, 2002; BODENHEIMER; WAGNER; GRUMBACH, 2002). Nesse sentido, é possível observar que smartphones mesclam duas características úteis no auxílio ao controle de tais doenças: estão sempre próximos ao seu proprietário, propiciando um gerenciamento contínuo, e possuem acesso a internet, viabilizando o contato com pessoas experientes em certas atividades de cuidado de DCNT.

De acordo com o relatório sobre internet móvel, da companhia de assessoria financeira Morgan Stanley² (MEEKER et al., 2009), o uso de internet através de dispositivos móveis superará ao dos computadores pessoais no contexto global até o ano de 2014. O mesmo relatório aponta que dispositivos móveis de grande capacidade como, por exemplo, smartphones, tablets, serão utilizados para o acesso de funcionalidades da internet que anteriormente eram feitas através dos computadores pessoais (PC).

Pouco a pouco a descrição do futuro da computação feita por Mark Weiser em 1991 vai se concretizando. Nesse cenário, ao invés da interação ser feita de forma direta com o com-

¹Sistema de Vigilância de doenças crônicas mantido pelo Ministério da Saúde do, que realiza entrevista por meio de inquérito telefônico em 27 cidades.

²<http://www.morganstanley.com/>

putador, ela passaria a ser feita através de inúmeros dispositivos computacionais de forma tão transparente e integrada ao ambiente que não seria possível perceber a existência do computador. Na visão de Weiser, os computadores tornam-se, móveis, pequenos e espalhados no ambiente em grande quantidade (WEISER, 1991). Essa abordagem da computação é chamada de computação ubíqua ou pervasiva, e é possível aplicá-la em diferentes áreas, tais como educação (BARBOSA et al., 2011), comércio (FRANCO et al., 2011) e saúde, onde também é conhecida como u-Health ou Pervasive Health.

Em um contexto de u-Health para cuidado de DCNT, os smartphones poderiam ser utilizados no suporte à prevenção dos fatores de riscos inerentes a estas doenças. Tais dispositivos podem ser utilizados para dar recomendações de locais adequados para uma alimentação saudável, na assistência a prática de atividades físicas, no alerta de zonas de riscos à saúde e na gestão de práticas para prevenção e controle, tais como, leitura de pressão arterial, glicemia, colesterol e ingestão de medicamentos.

1.2 Definição do Problema

A incidência das DCNT tem crescido em todo mundo. De acordo com relatórios da OMS, em 2020 elas representarão 43% da carga global entre todas as doenças (OMS, 2003). O cuidado eficiente das DCNT vai além do envolvimento do paciente. Ele também requer o envolvimento das organizações de saúde, família e membros da comunidade dentro das atividades de assistência ao gerenciamento de doença. Tais atividades transpõem a simples execução de procedimentos clínicos por médicos, enfermeiros e voluntários, elas também devem englobar o suporte à educação e treinamento do paciente, que deve estar preparado para tomar decisões em circunstâncias relativas a sua saúde (WAGNER et al., 2001).

Em sua maioria, os sistemas ubíquos de suporte ao cuidado de DCNT enfatizam o monitoramento do paciente e a geração de alertas. Pouca ênfase é dada ao suporte a treinamento do paciente e na integração entre agentes que possam auxiliar no cuidado dessas doenças. Outro ponto é falta de foco no controle de fatores de risco dessas doenças. Segundo a OMS, o controle desses fatores pode representar ganhos significativos na prevenção de doenças cardíacas, enfartes, diabetes tipo 2 e cânceres (OMS, 2005).

Com base nisso, abre-se a seguinte questão de pesquisa: quais características e elementos seriam necessários num modelo computacional para cuidado ubíquo de DCNT?

1.3 Objetivos

O objetivo central desse trabalho é a criação do U'Ductor³, um modelo para cuidado ubíquo de doenças crônicas não transmissíveis. Sistemas de computação ubíquos podem auxiliar pacientes a manter a sua concentração no controle dessas doenças, ao oferecer ferramentas que possibilitem ao paciente gerenciar e organizar aspectos como alimentação adequada, execução de atividades físicas, riscos comportamentais que possam levar ao agravamento do estado de saúde, medições de indicadores biológicos - como pressão arterial, peso e índices glicêmicos. A oferta de conteúdos educacionais e indicações de contatos sociais, também são pontos a serem observados. Em certas circunstâncias o paciente necessita lidar com entendimento de procedimentos, uso de medicamentos e dispositivos de forma não trivial (LAKE; STAIGER, 2010; WAGNER et al., 2001).

Assim, o presente trabalho possui os seguintes objetivos:

- propor o U'Ductor, um modelo para cuidado ubíquo de DCNT;
- desenvolver um protótipo que implemente os elementos definidos pelo modelo proposto;
- elaborar cenários para avaliação do modelo;
- avaliar o modelo através dos cenários propostos.

1.4 Metodologia

O primeiro passo executado para atingir os objetivos descritos foi efetuar uma revisão bibliográfica nos temas relacionados a u-Health e cuidado de doenças crônicas. Este estudo teve como objetivo determinar a forma como os trabalhos relacionados oferecem suporte ubíquo à saúde - que tecnologias são utilizadas, como é feita a identificação de contexto, como é realizada a interação entre o sistema e seus usuários e qual papel desempenha dentro do contexto de cuidado de DCNT.

A segunda etapa foi descrever o U'Ductor. Nesta etapa foram descritos os elementos que compõe o modelo, como eles se relacionam e de que forma contribuem para que o objetivo do modelo possa ser atingido.

Após especificado o modelo, foi construído um protótipo que possuía as funcionalidades necessárias para execução dos experimentos utilizados para avaliar o modelo. O primeiro experimento foi utilizado para que fosse possível avaliar a viabilidade de uso do U'Ductor em possíveis situações de cuidado e prevenção de DCNT. Na execução do experimento, indivíduos com diagnóstico de alguma DCNT utilizaram durante aproximadamente 15 minutos um

³O título U'Ductor é inspirado no nome científico do peixe piloto, que tem por característica sua relação de mutualismo com outras espécies marinhas, como tubarões e tartarugas. Cientificamente esta espécie chama-se *Naucrates Ductor* ou *N'Ductor*. Assim, o título do modelo é uma composição da palavra ubíquo mais o nome *N'Ductor*. Fonte: http://en.wikipedia.org/wiki/Pilot_fish

dispositivo móvel que possuía o protótipo instalado. Após a realização do experimento, os indivíduos foram convidados a efetuar um questionário que tinha o objetivo de avaliar a utilidade e facilidade de uso do protótipo no cuidado de DCNT.

O segundo experimento consistiu de uma avaliação funcional estendida do protótipo. Esta avaliação foi feita por um usuário com diagnóstico de DCNT, tendo a duração de um mês e meio, e tinha como objetivos coletar dados de contexto do usuário, calcular o tamanho necessário para o armazenamento dos dados coletados, e identificar questões de uso relacionadas às funcionalidades do protótipo.

Por fim, um terceiro teste foi efetuado para avaliar o consumo de bateria pelo protótipo do modelo, com o objetivo de produzir indicadores de referências para trabalhos futuros.

1.5 Organização do Texto

Esta dissertação foi organizada da seguinte maneira: o capítulo 2 descreve os conceitos chave relacionados ao modelo. Estes conceitos abrangem os temas de computação ubíqua, modelos de cuidados crônicos e saúde ubíqua. No capítulo 3 são apresentados e comparados cinco trabalhos relacionados aos temas de saúde ubíqua e cuidado de doenças crônicas. O capítulo 4 apresenta o modelo U'Ductor e seus componentes. Aspectos de implementação do protótipo são abordados no capítulo 5, onde também é apresentado como o modelo foi avaliado e os resultados obtidos. A dissertação é encerrada no capítulo 6, onde apresenta-se as conclusões acerca deste trabalho, contribuições do modelo proposto em relação aos trabalhos apresentados no capítulo 3, e sugestões para trabalhos futuros.

2 SAÚDE UBÍQUA E CUIDADO DE DOENÇAS CRÔNICAS

A computação ubíqua descreve um ambiente computacional que aperfeiçoa a experiência de uso de espaços físicos aos seus usuários, ao utilizar um ambiente saturado por computadores integrados de forma imperceptível como infraestrutura. A computação ubíqua pode ser aplicada em diferentes áreas tais como educação, comércio e saúde.

O uso da infraestrutura de computação ubíqua na área de saúde é chamado de u-Health ou Pervasive Health, podendo ser aplicada no gerenciamento de rotinas hospitalares, monitoramento de pacientes e suporte ao bem-estar. O cuidado de DCNT também pode ser beneficiado por tecnologias de computação ubíqua. No entanto, é importante que esse suporte seja baseado nos modelos e guias de cuidado de condições crônicas já estabelecidos. Dessa forma, esse capítulo abordará o tema computação ubíqua e sua aplicação à saúde, como também os modelos e práticas de cuidado de DCNT.

2.1 Computação Ubíqua

O termo computação ubíqua, ou ubicomp, foi criado por Mark Weiser, e seus colegas da Xerox, no início da década de 1990 (WEISER, 1991). Essa visão descreve um ambiente de computação altamente distribuído e integrado, saturado de dispositivos computacionais, onde a noção de computação pessoal como conhecemos desaparece. O computador torna-se parte da cultura, é algo comum, e está em praticamente toda parte - assistentes pessoais, roupas, utensílios domésticos ou carros.

Em 2001 Satyanarayanan enquadrava a visão de Weiser dentro do estado tecnológico existente. Para Satyanarayanan (SATYANARAYANAN, 2001) a computação ubíqua - que também pode ser chamada de computação pervasiva - representa uma evolução dentro da computação, tendo sido precedida pelos campos de estudos de sistemas distribuídos e computação móvel. Tecnicamente, a computação móvel é o ponto de partida da computação ubíqua que herda os conceitos dos seus predecessores, e adiciona mais alguns, que segundo Satyanarayanan seriam: espaços inteligentes, invisibilidade, escalabilidade localizada e mascaramento de condições desiguais.

Entende-se por espaços inteligentes, a utilização da infraestrutura computacional existente em um local, para aperfeiçoar a qualidade de uso desse ambiente por alguém. Em uma palestra internacional, a seleção automática da linguagem de tradução simultânea, de acordo com a cultura do ouvinte, seria um exemplo.

Invisibilidade é a capacidade que um sistema ubíquo tem de não ser percebido enquanto é utilizado. A interação entre sistema e usuário deve ser feita “praticamente em um nível subconsciente” (SATYANARAYANAN, 2001). Por exemplo, um sistema de navegação que guia o motorista de forma adequada, e sem surpresas, até o seu destino.

Escalabilidade localizada é determinada pelo fato que, em um ambiente ubíquo, a de-

manda por recursos fisicamente próximos é maior do que a demanda por recursos distantes. Desta maneira, sistemas ubíquos devem ser projetados para atender requisições de entidades localizadas fisicamente mais próximas. Para ilustrar, pode-se imaginar uma impressora pertencente a um setor de uma empresa, que dá prioridade de impressão para documentos de usuários localizados nesse setor.

O mascaramento de condições desiguais é motivado pela possibilidade de existirem espaços inteligentes com menos capacidade e riqueza de recursos que outros. Nesse caso, fica a cargo do ambiente pessoal do usuário compensar esta carência. Por exemplo, em uma reunião, na ausência de um projetor, as telas dos dispositivos dos participantes poderiam estar sincronizadas com a tela do dispositivo do interlocutor.

Kindberg e Fox (KINDBERG; FOX, 2002) completam a definição da área, indicando que sistemas ubíquos são orientados por duas características principais: integração física e interoperação espontânea. A característica de integração física é similar ao conceito de espaço inteligente, onde objetos integrados a dispositivos computacionais potencializam a experiência de uso do espaço físico. Essa característica determina o princípio de fronteira, esse princípio define que sistemas ubíquos devem indicar de forma clara os limites de início e fim entre esses espaços.

A interoperação espontânea é a capacidade de um dispositivo se comunicar com outros dispositivos em um ambiente dinâmico, isto é, em um ambiente onde os dispositivos participantes mantêm associações passageiras. Essa característica determina o princípio da volatilidade, onde é definido que, uma vez que os dispositivos participantes de sistemas ubíquos são dinâmicos e imprevisíveis (i.e. não é possível determinar o seu comportamento) deve haver um tipo de regra que dirija a execução do sistema.

Em linhas gerais, a computação ubíqua pode ser descrita como um ambiente saturado de dispositivos computacionais com capacidade de comunicação, onde dispositivos móveis alternam de ambientes dinamicamente, integrando-se a estes espaços de forma transparente, para aperfeiçoar a experiência de uso de alguma aplicação aos seus usuários.

2.2 Modelos de Cuidados para Condições Crônicas

Doenças circulatórias e respiratórias, cânceres e diabetes, são consideradas as DCNT de maior incidência em um nível global (OMS, 2005, 2008; CASTRO et al., 2008). As DCNT pertencem ao conjunto de condições crônicas que envolvem um espectro maior de problemas de saúde, tais como, distúrbios mentais de longo prazo, condições transmissíveis permanentes e deficiências físicas contínuas (OMS, 2003). De modo geral, estas condições possuem algumas características comuns, pois “exigem mudança no estilo de vida”, necessitam de “gerenciamento de saúde por um longo período de tempo”, como também são causadas por estilos de vida e comportamentos de risco, como dietas inadequadas, tabagismo, sedentarismo, alcoolismo, práticas sexuais de alto-risco e estresse social. Além dos fatores comportamentais,

a idade também é um fator que pode influenciar a incidência de algumas condições crônicas (OMS, 2003; WAGNER et al., 2001; CASTRO et al., 2008).

O rápido crescimento da incidência dessas condições em um nível global (OMS, 2003), bem como a mudança de paradigma determinada pelo tratamento de condições crônicas, uma vez que ela não pode ser encarada da mesma forma que quadros agudos, onde não há a necessidade de cuidados de longo prazo e existe pouco envolvimento por parte do paciente (WAGNER et al., 2001), forçaram a criação de modelos que propiciem o gerenciamento de condições crônicas (OMS, 2003; WAGNER et al., 2001). “The Chronic Care Model” (CCM) (WAGNER et al., 2001; ICIC, 2011) e “Cuidados Inovadores para Condições Crônicas” (CICC) (OMS, 2003) são alguns exemplos dessas iniciativas sendo descritos nas próximas seções.

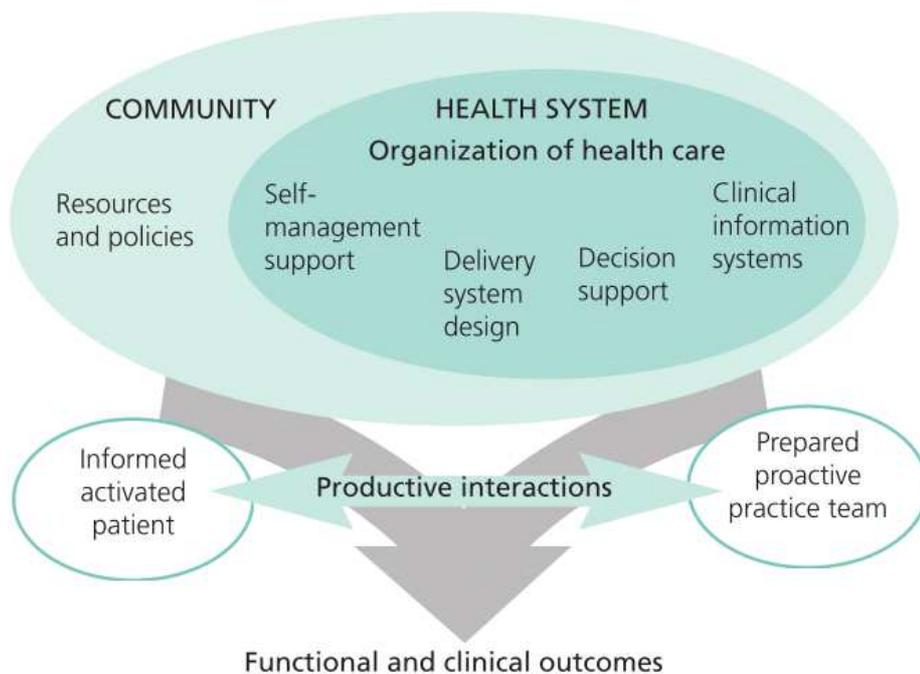
2.2.1 Modelo CCM

O objetivo do CCM é “guiar o aperfeiçoamento qualitativo e atividades de gerenciamento de doenças”, descrevendo as práticas que contribuem para a interação entre os provedores de saúde e pacientes. Assim, um CCM deve garantir que os pacientes tenham confiança para gerenciar suas condições; assegurar o controle e prevenção de complicações; definir planos de cuidado entendíveis para pacientes e times médicos; e propiciar o sequenciamento regular. O CCM é composto por cinco elementos principais que são: a comunidade (Community), o sistema de saúde (Health System), suporte ao autogerenciamento (Self-management support), projeto de sistema de entrega (Delivery system design), suporte a decisão (Decision support), e sistemas de informação clínicos (Clinical information systems). Estes cinco itens são descritos a seguir:

- Comunidade: engloba os pontos de ligação entre pacientes e sistema de saúde, mantendo parcerias para serviços e recursos que não existem localmente, isto é, que não existem na comunidade em que o paciente vive. Esse elemento também tem o papel de intervir em políticas de saúde que beneficiem a comunidade;
- Sistema de Saúde: é responsável por assegurar que as equipes médicas tenham condições de oferecer suporte aos pacientes. Esse elemento incorpora os outros três elementos principais do modelo;
 - Suporte ao autogerenciamento: é o provimento das condições necessárias para que o paciente entenda o seu papel no tratamento de sua condição, e se engaje nesse tratamento;
 - Projeto de Sistema de Entrega: assegura a delegação de atividades entre membros da equipe de cuidado ao definir papéis e responsabilidade entre esses membros;
 - Suporte a Decisão: são os métodos utilizados para orientar o paciente, de forma que o seu autogerenciamento obtenha sucesso;

- Sistemas de Informação Clínicos: os sistemas informatizados podem ser utilizados para auxiliar os paciente em seu autogerenciamento, como também as equipes médicas ao manter informações sobre os pacientes possibilitando o seu monitoramento.

Figura 1: The Chronic Care Model



Fonte: (ICIC, 2011)

A integração destes elementos possibilita que o paciente mantenha-se informado e ativo dentro das suas responsabilidades de gerenciamento, e os provedores de cuidado cientes das necessidades existentes na comunidade (Figura 1).

2.2.2 Modelo CICC

O CICC é uma iniciativa da OMS para criar um modelo abrangente, aplicável ao gerenciamento de diversas condições crônicas, tais como HIV, tuberculose, doenças cardiovasculares, diabetes e distúrbios mentais de longo prazo. O CICC divide seus componentes em três níveis de abrangência, micro, meso e macro, partindo do pressuposto que para obter sucesso o modelo de cuidado crônico deve abranger todos os níveis que envolvem desde o paciente até o nível governamental, responsável pela criação de políticas de saúde (Figura 2).

O nível micro é composto pela tríade pacientes e familiares, grupo de apoio da comunidade e equipes de atenção à saúde, que devem trabalhar em sincronia para que todos elementos estejam preparados, informados e motivados para auxiliar no tratamento das condições crônicas.

A comunidade e as organizações de saúde são os componentes de nível meso que dão apoio a tríade. As organizações de saúde devem coordenar o tratamento do paciente, fornecer recursos

Figura 2: Cuidados Inovadores para Condições Crônicas



Fonte: (OMS, 2003)

de trabalho para o pessoal de área de saúde e dar apoio ao autogerenciamento e prevenção. A comunidade deve auxiliar seus líderes para que enfatizem a necessidade de cuidados de doenças crônicas, promover campanhas de saúde, capacitar voluntários locais e prover recursos básicos de tratamento das condições crônicas.

O nível macro é representado pelo nível governamental, sendo responsável pelo planejamento e execução de políticas necessárias para reduzir a carga de condições crônicas.

2.2.3 Considerações sobre os modelos CCM e CICC

É possível observar que em ambos modelos existe uma forte integração entre comunidade e organizações de saúde, como forma de garantir o sucesso do controle da condição crônica pelo paciente. Esses três elementos devem estar atualizados a respeito das práticas de tratamento de doenças crônicas, bem como informados sobre políticas e projetos de auxílio ao tratamento e controle dessas doenças. Nesse sentido, a educação e comunicação são fatores importantes para que todos os membros do modelo obtenham sucesso em suas funções.

2.3 Fatores de Risco de DCNT

O gerenciamento, a prevenção e o controle das DCNT podem ser feitos através do combate aos fatores de riscos comportamentais e biológicos relacionados a essas enfermidades. Dietas inadequadas, sedentarismo e tabagismo são os principais riscos comportamentais considerados pela OMS, enquanto que hipertensão, alto colesterol e sobrepeso são considerados como os

principais fatores de riscos biológicos a serem controlados (OMS, 2010). No relatório Preventing CHRONIC DISEASES a vital investment é dito que “se os fatores de riscos forem eliminados, pelo menos 80% de todas doenças cardíacas, enfartes e diabetes tipo 2 seriam evitadas; mais 40% dos cânceres poderiam ser prevenidos” (OMS, 2005).

A Tabela 1 organiza um comparativo entre os principais fatores de riscos considerados pela OMS e a forma de como devem ser prevenidos ou controlados. Entende-se por controle alimentar o hábito de consumir adequadamente frutas e vegetais, e a ingestão controlada de sal e gorduras. Atividade física compreende a execução de pelo menos 30 minutos de algum tipo de atividade física moderada em pelo menos cinco dias da semana. Medicamentos entende-se pela medicação prescrita pelo médico como forma de controlar aspectos da doença. Sensoriamento compreende os mecanismos utilizados para medir sinais vitais necessários para acompanhar a execução do tratamento da doença ou também como forma de prevenir possíveis casos de DCNT.

Tabela 1: Relação fatores de riscos x prevenção/controlado

Fator de Risco	Controle Alimentar	Atividade Física	Medicamento	Sensoriamento
Hipertensão	Sim	Sim	Sim	Sim
Colesterol	Sim	Sim	Sim	Sim
Sedentarismo	Não	Sim	Não	Não
Tabagismo	Não	Não	Sim	Não
Obesidade/ Sobrepeso	Sim	Sim	Sim	Sim
Dietas inadequadas	Sim	Não	Não	Não

Fonte: Elaborado pelo autor.

2.4 Saúde Ubíqua

Em termos gerais, o tema saúde ubíqua, ou u-Health, pode ser entendido como a oferta de assistência médica em qualquer lugar, a qualquer momento, por meio de tecnologias de comunicação de banda larga e sem-fio móveis (CACERES et al., 2006; JEONG; JUNG; PARK, 2009; RODRIGUES et al., 2011). De uma maneira mais resumida, u-Health também pode ser entendido como a utilização de infraestrutura de computação ubíqua para assistência à saúde (LIM et al., 2009; DEY; ESTRIN, 2011).

De um modo geral, as aplicações de u-Health concentram-se em uma das seguintes atividades: gerenciamento de rotina hospitalar, monitoramento de pacientes e suporte ao bem-estar. As próximas subseções descrevem estas atividades ao apresentar alguns trabalhos relacionados a elas. Primeiramente optou-se em selecionar os trabalhos apresentados na seção Pervasive

Health, da revista *Pervasive Computing*¹, pelo fato desta revista ser uma referência no tema de computação ubíqua. Entretanto, alguns textos de outras publicações também foram utilizados por caracterizarem melhor a atividade de u-Health a ser descrita.

2.4.1 Aplicações Hospitalares

As aplicações hospitalares tentam aperfeiçoar as atividades de rotinas de médicos e enfermeiros, tais como a administração e prescrição de medicamentos, colaboração entre médicos, conferências, cirurgias e emergências. É preciso ter em mente que a dinâmica de trabalho de enfermeiros e médicos em um ambiente hospitalar é diferente da convencional. Geralmente esses profissionais não têm um ponto fixo de trabalho, precisam memorizar atividades paralelas, e trocam de atividade de maneira rápida (BARDRAM; CHRISTENSEN, 2007).

Bardram et. al. realizaram uma pesquisa de uso de computação baseada em atividades (BARDRAM, 2005) no suporte ao trabalho hospitalar. Uma vez que o trabalho dos profissionais de saúde em ambientes hospitalares não é convencional, a simples utilização de softwares convencionais não é adequada, já que a busca por dados do paciente e outras informações relevantes, além da execução de aplicações, demandaria tempo e sobrecarregaria o profissional. A ideia central da pesquisa é utilizar a computação baseada em atividades para facilitar a recriação do contexto da atividade que o médico encontra-se, de forma transparente (BARDRAM; CHRISTENSEN, 2007).

O Stroke Angel (HOLTMANN et al., 2007; ORWAT et al., 2010) é outro trabalho relevante no segmento hospitalar. O objetivo do Stroke Angel é otimizar o tempo de atendimento dentro de situações de emergência de enfartes. De acordo com os autores, as primeiras três horas são cruciais para o sucesso da terapia e a qualidade de vida do paciente. O sistema conta com o uso de um PDA, que é utilizado conjuntamente com outros equipamentos, para efetuar o diagnóstico de enfarte. Uma vez diagnosticado o enfarte, é iniciado o processo de admissão em um hospital especializado. Os dados do paciente são enviados para hospital através de tecnologias de rede sem-fio, possibilitando a preparação do atendimento ao paciente, enquanto ele está se deslocando ao hospital.

2.4.2 Monitoramento de Pacientes

O caso de uso MS Nurses (RASHID et al., 2007; ORWAT et al., 2010) utilizou um cinto equipado com um acelerômetro de três eixos, para monitorar a atividade de pacientes, e assim auxiliar no diagnóstico e terapia de esclerose múltipla.

Em U-Health Smart Home (AGOULMINE et al., 2011), é descrito o projeto de casa inteligente, da Universidade de Ciência e Tecnologia de Pohang, para auxílio a idosos ou pessoas portadoras de doenças crônicas. A casa é projetada para coletar dados fisiológicos dos pacientes,

¹<http://www.computer.org/portal/web/computingnow/pervasivecomputing>

por meio de uma rede de sensores corporais, e também coletar dados do ambiente como, por exemplo, umidade, temperatura e luminosidade. Estas informações são utilizadas para que seja possível tomar ações de forma automática, quando necessário. Essas ações podem ser a liberação de medicamento, através de um atuador disponível no corpo do paciente, ou até mesmo apagar a luz de um cômodo que o paciente esqueceu acesa. Além disso, a casa também é conectada a uma central de serviços médicos que provê assistência em casos de emergências ou situações de risco.

2.4.3 Suporte ao Bem-Estar

A categoria de bem-estar compreende aquelas aplicações orientadas ao gerenciamento pessoal de saúde, tais como gerenciamento de dietas alimentares, assistentes de treinamento de atividades físicas e jogos orientados a prática de atividades físicas.

Monster & Gold é um jogo cujo objetivo é auxiliar seus usuários a alcançar benefícios cardiovasculares, através da prática de exercícios físicos. O jogo é executado em um PDA com GPS embutido (BUTTUSI; CHITTARO, 2010). Um oxímetro de pulso, anexado à orelha do usuário, é utilizado para leitura da frequência cardíaca. Conforme o usuário corre em determinado percurso, e de acordo com o seu contexto físico, o jogo apresenta recompensas ou desafios, representados por ouro, poções, escudos e monstros. Ouro, poções e escudos são oferecidos quando o usuário precisa aumentar o ritmo de sua atividade física. Os monstros são apresentados em momentos onde o usuário deve diminuir o ritmo de atividade, para que ela não se torne prejudicial.

O Intelligent Fitness Guide (IFG) é um sistema de otimização de programas de exercício físico (LIM et al., 2009). A proposta do sistema é oferecer uma programação de exercícios físicos adaptada ao contexto do usuário. Dessa forma, o sistema leva em consideração os sinais vitais do paciente (pressão arterial e ritmo cardíaco) e objetivos estipulados por ele (perder peso ou diminuir a pressão arterial), a fim de oferecer uma programação de exercícios físicos que melhor se adapte aos objetivos selecionados.

2.5 Considerações sobre o Capítulo

O presente capítulo apresentou uma introdução aos temas de computação ubíqua, sua aplicação na área de saúde, e modelos e práticas de cuidado de DCNT existentes. A computação ubíqua é o estado onde a computação é altamente integrada e distribuída, e sua aplicação na assistência à saúde é chamada de u-health. Esta aplicação em geral é feita no apoio às rotinas hospitalares, no monitoramento de pacientes e no suporte ao bem-estar.

O gerenciamento de fatores de risco e os modelos de cuidados crônicos são ferramentas para suporte ao cuidado de DCNT. Parte do gerenciamento de fatores de risco é feito pelo paciente através do controle de alimentação, prática de atividades e auto monitoramento. Os modelos de

cuidados crônicos abordam a integração entre pessoas e organizações como um meio para obter sucesso no cuidado de condições crônicas.

U-Health pode ser utilizado para auxiliar o cuidado de DCNT, tanto no gerenciamento de fatores de risco, quanto no suporte a implementação dos modelos de cuidados crônicos. O próximo capítulo apresentará trabalhos de u-Health que de alguma maneira concentram-se no cuidado de DCNT.

3 TRABALHOS RELACIONADOS

A utilização de u-Health no suporte ao cuidado de DCNT pode ser realizado na assistência às atividades de autogerenciamento (MAMYKINA et al., 2008; MATTILA et al., 2008), na oferta de meios para comunicação entre paciente e cuidadores (MAMYKINA et al., 2008), no monitoramento de condições de saúde em tempo real (ROSSO et al., 2010; KOUTKIAS et al., 2010; PAGANELLI; GIULI, 2011) ou também através do envio de mensagens aos responsáveis médicos em casos onde o paciente encontra-se em situação de risco. Esta seção apresenta cinco trabalhos que concentram-se no uso de u-Health para suporte às atividades de cuidado de DCNT.

3.1 MAHI

MAHI, ou Mobile Access to Health Application, é uma aplicação cujo objetivo é auxiliar pacientes com diagnóstico de diabetes a tomarem decisões em questões de sua vida cotidiana, como por exemplo, qual o tipo de refeição mais adequada para que o paciente faça em seu almoço (MAMYKINA et al., 2008). De acordo com os autores, o paciente deve ser o controlador de sua rotina de cuidados e a aplicação deve auxiliá-lo a atingir esse hábito, que pode ser alcançado através da educação a respeito da doença, e de como ela deve ser tratada. Segundo os autores, isto fará com que o indivíduo possa aprender a conviver com a doença em sua rotina e tomar decisões mais adequadas, que favoreçam não só ao controle da doença, mas também a sua qualidade de vida.

A arquitetura MAHI é dividida em dois componentes: uma aplicação móvel e um web site. A aplicação móvel é composta por glicosímetro, adaptador bluetooth e um telefone celular com suporte a Java, e tem por finalidade registrar informações como os níveis de glicose sanguíneo dos pacientes, hábitos alimentares, e dúvidas relacionadas ao tratamento (Figura 3). Para estas duas últimas questões a aplicação pode utilizar os recursos de captura de áudio e imagem do telefone.

Figura 3: Componentes móveis do MAHI



Fonte: (MAMYKINA et al., 2008)

O website funciona como um canal de comunicação entre o paciente e um educador de diabetes. O educador visualiza as informações compartilhadas pelo paciente - índices glicêmicos, perguntas e rotina alimentar - com o objetivo de orientar e treinar o paciente no seu cuidado diário. O website também serve para que o paciente possa consultar o histórico de suas decisões, possibilitando que ele tenha atitudes de melhor qualidade.

Os autores fizeram um experimento de uso da ferramenta com pacientes do Centro de Educação de Diabetes do Hospital St. Claire em Dover, Nova Jersey. Foram recrutados 49 pacientes que foram divididos em dois grupos, um grupo teve acesso ao uso da ferramenta MAHI, enquanto o outro grupo apenas teve o treinamento semanal sobre diabetes. O objetivo do experimento era verificar as diferenças entre os dois grupos para questões como: estado emocional com relação a capacidade de controle da doença por parte do paciente; capacidade de compreensão da diabetes; e a capacidade dos pacientes em gerenciar suas metas.

O resultado da primeira questão mostrou um incremento positivo na consciência de responsabilidade de gerenciamento da doença, por parte dos pacientes que utilizaram a aplicação. Para as outras duas questões os resultados foram semelhantes para ambos os grupos.

3.2 Wellness Diary

Wellness Diary (WD) é um aplicativo de gerenciamento de bem-estar móvel e pessoal baseado no conceito de Terapia Cognitivo-Comportamental (TCC) (MATTILA et al., 2008). TCC é um tipo de terapia em que o paciente utiliza auto monitoramento para identificar e aprender quais comportamentos e ações têm influência na manutenção de sua condição de saúde. Segundo os autores essa abordagem seria adequada para o controle de comportamentos de riscos, tais como alimentação inadequada e sedentarismo, que contribuem para o aumento das ocorrências de doenças. Outro apontamento dos autores, é que o sistema de saúde não é capaz de gerenciar esses comportamentos, dessa forma é necessário o comprometimento dos indivíduos para que a prevenção das doenças relacionadas a estes comportamentos obtenha sucesso.

O WD é dividido em três componentes: entrada de observações (Figura 4(a)); visualização do decurso das observações (Figura 4(b)); e agenda de atividades (Figura 4(c)). Na entrada de observações os usuários inserem informações de consumo de alimentos e bebidas, peso, exercícios, pressão sanguínea, qualidade de sono, estresse. A visualização do decurso das observações é feita através de gráficos, que permitem ao usuário interpretar facilmente o resultado de suas decisões. A agenda de atividades é integrada com a aplicação de calendário do dispositivo móvel. Essa integração foi feita para diminuir a curva de aprendizado da ferramenta, já que parte dos usuários deve estar familiarizado com ela, e integrar-se às atividades dos usuários.

Dois estudos foram efetuados para avaliação da usabilidade e aceitação da aplicação. O primeiro concentrou-se no gerenciamento de peso de pessoas com sobrepeso. Esse estudo foi efetuado com 29 pessoas e durou 3 meses. Após o período de testes os usuários participaram de entrevistas semi-estruturadas e questionários, para identificar suas impressões com relação à

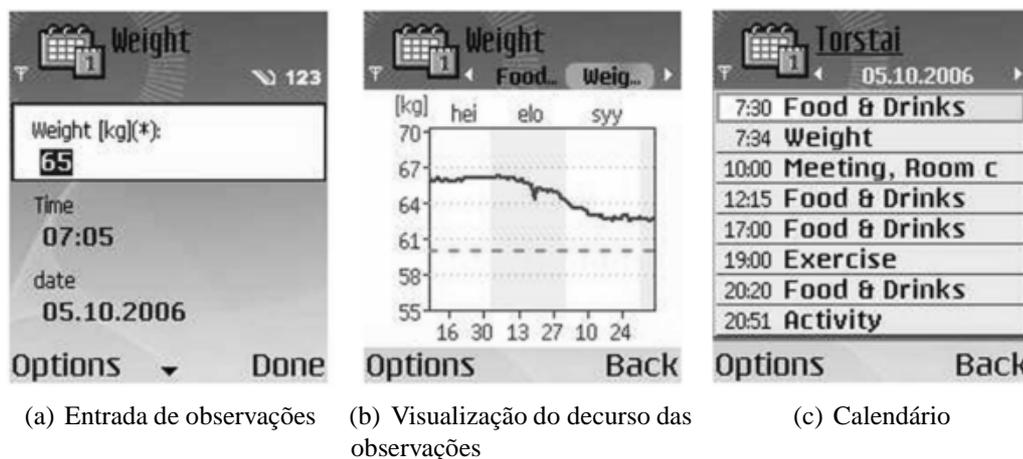


Figura 4: Interface de uso do WD

aplicação. Posteriormente, os usuários foram divididos em dois grupos: usuários que perderam peso e usuários que não perderam peso. Foi notado que a maioria dos usuários que perdeu peso avaliou a aplicação como útil, o que não ocorreu com o outro grupo. Do grupo que perdeu peso, 90% dos integrantes avaliaram a aplicação como útil, enquanto que 79% dos integrantes do outro grupo avaliaram dessa mesma maneira. Outro resultado observado foi que os usuários que perderam peso fizeram mais registros de atividades que os usuários do outro grupo.

O segundo estudo foi feito para avaliação da aplicação com relação ao gerenciamento de bem-estar de um modo geral, não sendo abordada uma aplicação específica. Este estudo esteve concentrado na avaliação de usabilidade e aceitação. Pressão sanguínea, peso e exercícios foram considerados os dados de controle mais importantes.

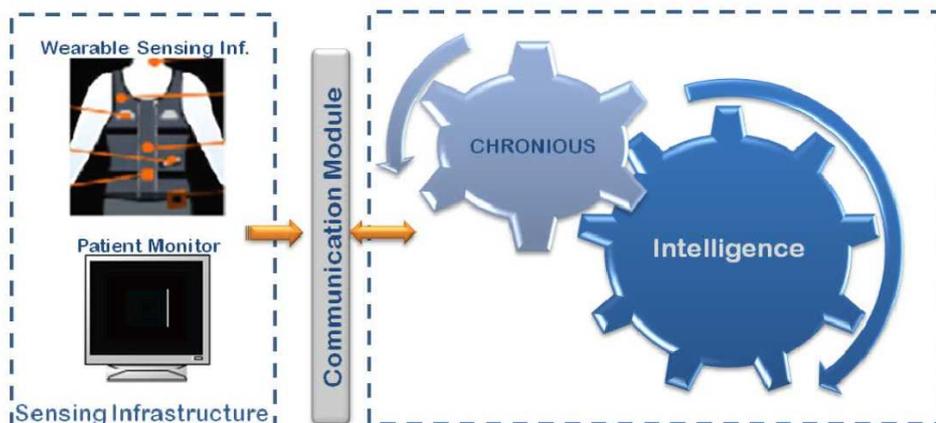
3.3 CHRONIOUS

CHRONIOUS (ROSSO et al., 2010) é uma plataforma para gerenciamento de doenças crônicas formada por três componentes principais: infraestrutura de sensores, módulo de comunicação e inteligência do CHRONIOUS (Figura 5).

A infraestrutura de sensores é responsável pelo monitoramento dos sinais vitais do paciente. Como era necessário um modo de leitura “contínuo, discreto e não invasivo”, foi projetada uma vestimenta no formato de uma camiseta com capacidade de extrair dados de frequência cardíaca, frequência respiratória, temperatura e umidade do ambiente, movimento e detecção de tosse e ronco.

A unidade de processamento de dados da vestimenta também é capaz de analisar os dados coletados em tempo real e disparar eventos de alerta. A infraestrutura de sensores também é responsável por coletar informações de hábitos de vida do paciente tais como, dieta e ingestão de medicamentos. Estes dados são coletados por meio de questionários respondidos pelo paciente através de uma unidade de monitoramento de paciente, que também pode coletar dados

Figura 5: Principais componentes do Chronius



Fonte: (ROSSO et al., 2010)

ambientais, tais como luz ambiente ou qualidade do ar.

O módulo de comunicação é responsável por transferir os dados adquiridos através da infraestrutura de sensores e enviá-los ao componente de inteligência. O componente de inteligência deve auxiliar profissionais de saúde no gerenciamento dos pacientes monitorados. Este gerenciamento é feito em dois níveis. O primeiro nível é responsável pela identificação da situação de saúde do paciente e é localizado no assistente pessoal do paciente.

O segundo nível é chamado de Sistema de Suporte a Decisão Clínica, ou Clinical Decision Support System (CDSS). O CDSS tem por objetivo fornecer interpretações e sugestões clínicas a partir dos dados fornecidos pelo paciente, de forma que os profissionais clínicos responsáveis não sejam sobrecarregados pelo excesso de informação coletada. Este sistema é acionado nas situações de leitura de novos dados ou quando os alertas são enviados pelo primeiro nível.

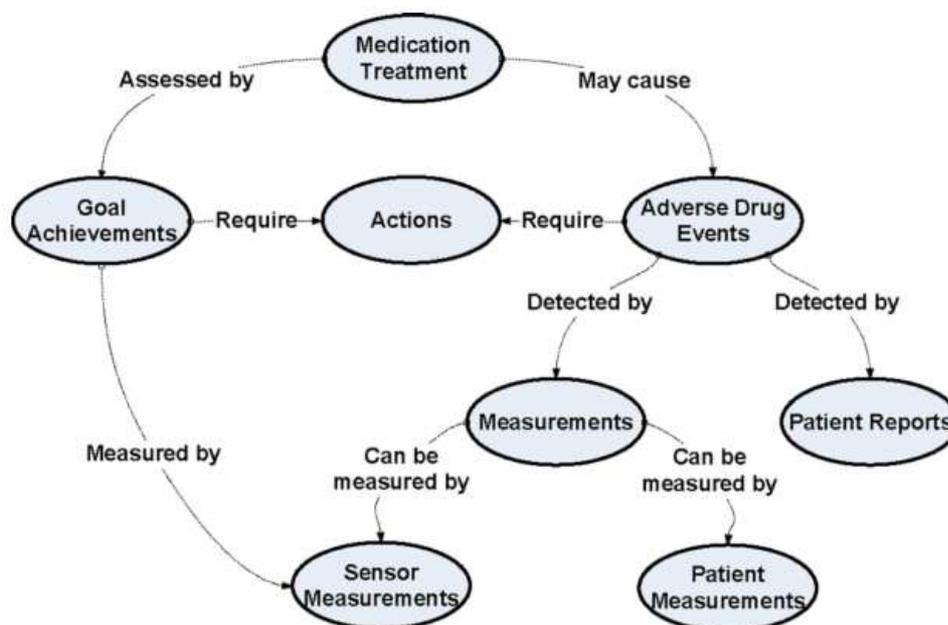
3.4 Framework para gerenciamento de medicação de Koutkias et-al.

Koutkias et-al. (KOUTKIAS et al., 2010), propõem um framework cujo objetivo é utilizar os dados coletados através de sensores de monitoramento, para identificar como o paciente reage aos medicamentos prescritos e efeitos adversos que podem ser causados por eles. De acordo com os autores, durante os últimos anos várias soluções para o monitoramento de pacientes fora do ambiente hospitalar foram propostas, mas não é viável que o ambiente hospitalar tente ser replicado no ambiente em que o paciente vive, uma vez que isso aumentaria a quantidade de recursos, humanos e tecnológicos, necessários para o cuidado médico.

Nesse contexto, um framework conceitual foi criado para descrever os aspectos relacionados ao domínio de monitoramento de efeitos de uso de medicamentos (Figura 6). O framework conceitual é uma ontologia que descreve os objetivos a serem alcançados com o uso do medicamento, por exemplo, reduzir as medidas de pressão arterial; efeitos adversos que podem ser causados pelo uso do medicamento, por exemplo, aumento da taxa de batimentos cardíacos

do paciente; e de que forma estas informações devem ser adquiridas, por exemplo, através de sensores que coletam os sinais automaticamente, ou por meio de questões enviadas ao paciente.

Figura 6: Framework conceitual de gerenciamento de tratamento medicamentoso



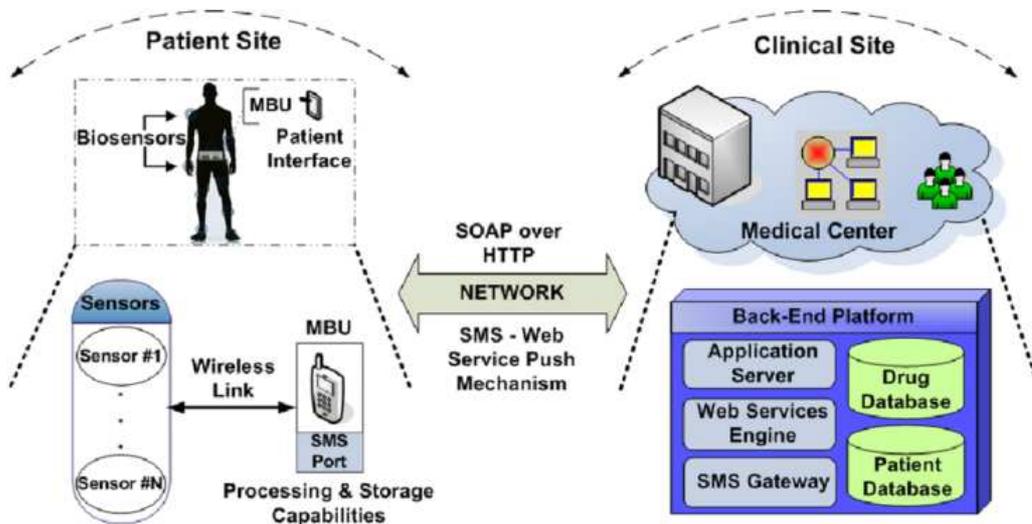
Fonte: (KOUTKIAS et al., 2010)

O framework conceitual é implementado dentro de um sistema composto por dois elementos principais, ou subsistemas, que se comunicam através de uma arquitetura orientada a serviços (Figura 7). No subsistema clínico (lado direito da Figura 7) residem o banco de dados de informações de drogas e o banco de dados de informações do paciente. O banco de dados de informações de drogas armazena as informações sobre substâncias, indicações e contraindicações de uso, e efeitos adversos. O banco de dados do paciente é responsável por manter informações de dados clínicos dos pacientes como relatórios, alertas e dados monitorados.

O subsistema do paciente (lado esquerdo da Figura 7) é composto por uma rede de área corporal (BAN) controlada pela unidade base móvel (MBU). O subsistema clínico comunica-se com o subsistema do paciente por meio da MBU, a partir de mensagens de texto (SMS). Dessa forma os SMSs funcionam como meio de transporte assíncrono para as requisições de serviço entre os subsistemas.

Para ilustrar o funcionamento do sistema os autores modelaram um cenário onde um paciente com diagnóstico de hipertensão recebe a prescrição de um novo medicamento. O médico então descreve a dosagem, frequência de uso do medicamento, objetivos a serem alcançados com o uso do medicamento, possíveis efeitos adversos, e um cronograma de leitura de informações referentes aos efeitos do uso do medicamento. Essas informações são enviadas à MBU, que então inicia o monitoramento. No período de monitoramento, a MBU recebe as informações de pressão sanguínea e batimentos cardíacos a partir de sensores ligados ao paciente. Informações sobre efeitos adversos, como falta de ar e náusea, são respondidas pelo paciente

Figura 7: Arquitetura geral do sistema



Fonte: (KOUTKIAS et al., 2010)

diretamente na MBU através de uma interface gráfica. Depois de finalizado o monitoramento, a MBU automaticamente envia um relatório contendo os resultados ao subsistema clínico.

3.5 Sistema de Gerenciamento de DCNT de Paganelli e Giuli

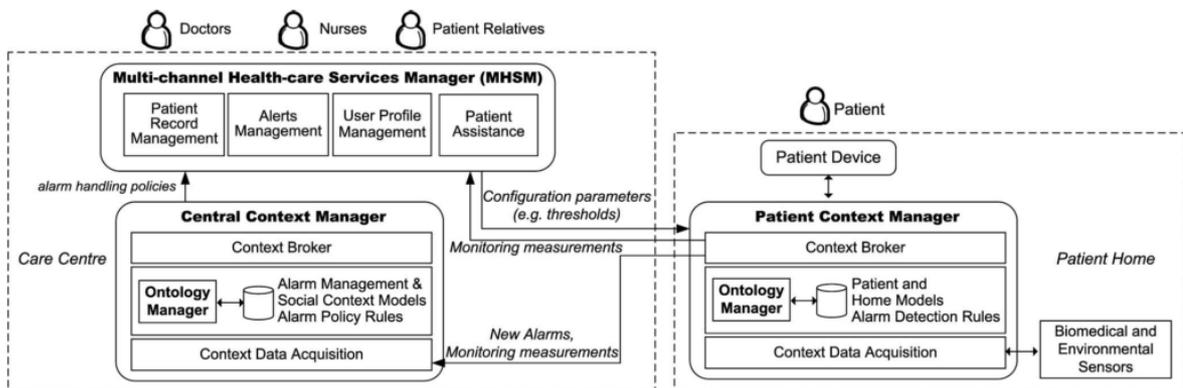
Paganelli e Giuli apresentam um sistema de gerenciamento e suporte ao tratamento de DCNT sensível ao contexto. Esse sistema é executado por uma plataforma orientada a serviços de suporte ao cuidado de condições crônicas chamada EHRMAN (PAGANELLI; SPINICCI; GIULI, 2008), que utiliza um modelo para raciocínio de contexto baseado em ontologia (PAGANELLI; GIULI, 2011). A motivação dos autores é dada pelo fato que o número de casos doenças crônicas têm aumentado e são particularmente mais expressivos nas populações idosas, que também têm crescido. Diferentemente dos tratamentos de doenças agudas, as doenças crônicas envolvem tratamentos mais lentos e prolongados. Modelos de gerenciamento de doenças crônicas começaram a ser promovidos recentemente, baseados em modelos de cuidado continuado. Nesses modelos pacientes, familiares, e outros membros da comunidade, partilham responsabilidades nas atividades de cuidado da doença.

Em modelos de cuidado continuado, a plataforma deve possuir duas capacidades para que seja possível atingir os requisitos de cuidado a longo prazo: reconfigurabilidade, devido a mudanças sociais, culturais e políticas que podem ocorrer em diferentes casos; e adaptação sensível ao contexto, isto é, o sistema deve agir de acordo com a situação na qual os pacientes e cuidadores estão inseridos. Segundo os autores, sistemas ubíquos têm condições de oferecer auxílio a pacientes portadores de doenças crônicas, já que estes se concentram em oferecer auxílio a qualquer momento, em qualquer lugar e de forma personalizada a situação do paciente. Esses sistemas podem monitorar a situação clínica do paciente, como também gerar alertas de

riscos e lembretes de atividades. Nesse ambiente, ontologias são utilizadas para expressar o conhecimento de um domínio de forma inequívoca, auxiliando o raciocínio do contexto.

A plataforma de serviços ERMHAN concentra-se no auxílio das atividades dos atores da rede de cuidado, provendo um ambiente de monitoramento doméstico para o paciente, e geração de alertas para a rede de cuidado, de acordo com regras definidas no sistema. A arquitetura da plataforma é dividida em três componentes principais: gerente de serviços de cuidado de saúde multicanal, gerente de contexto central e gerente de contexto do paciente. O primeiro componente é responsável por oferecer serviços de alerta, prontuário médico, perfil e assistência a pacientes. O gerente de contexto central é responsável por entregar a informação de contexto relativa ao paciente aos membros da rede de cuidado, de acordo com as regras de disponibilidade e nível de alerta. O último componente é responsável pelo monitoramento do paciente por meio dos parâmetros de saúde e ambiente configurados no sistema. São nesses dois últimos componentes que estão inseridos os modelos ontológicos apresentados pelos autores. A Figura 8 apresenta a organização dessa plataforma.

Figura 8: Plataforma ERMHAN



Fonte: (PAGANELLI; GIULI, 2011)

Os atores da rede de cuidado são divididos entre, pacientes, familiares, médicos, enfermeiras, assistentes sociais, centro de cuidado e membros da comunidade. Pacientes são monitorados e recebem atenção dos familiares e profissionais de saúde, de acordo com a especificação do seu plano de cuidado. Familiares representam membros da família que fazem mudanças de rotina para se envolver no auxílio ao paciente. Eles têm um papel importante no tratamento do paciente, portanto necessitam estar atualizados sobre as condições do paciente. Enfermeiras e médicos são profissionais de saúde mais especializados e devem ser requisitados em situações mais críticas. Médicos também são responsáveis pela elaboração do plano de cuidado do paciente, de acordo com o seu histórico e perfil clínico. Os operadores de assistência social provêm suporte as atividades diárias do paciente. O centro de cuidados manipula o plano de cuidado do paciente, gerencia alertas e chamadas, e planeja intervenções. Os membros da comunidade estendem os serviços de cuidado. Esse último papel pode envolver pessoas de diferentes atividades como voluntários, farmacêuticos, vendedores, etc.

Dessa forma, o modelo criado pelos autores visa representar o conhecimento relacionado aos seguintes casos:

- capturar dados de monitoramento do paciente a partir de fontes diferentes;
- identificar a situação do paciente;
- demonstrar a situação do paciente;
- especificar planos de cuidado personalizados.

A ontologia foi dividida em quatro sub ontologias: ontologia pessoal do paciente; ontologia de domínio da casa do paciente; ontologia de gerenciamento de alarmes; e ontologia de contexto social. Estas ontologias representam os conceitos relacionados às funcionalidades definidas para o sistema. A ontologia Pessoal do Paciente representa os elementos responsáveis pelo monitoramento da condição do paciente. A situação do paciente é inferida através da aplicação de regras sobre os dados coletados de sensores distribuídos na casa ou ligados ao paciente. Intervalos de valores limite são utilizados para determinar situações de alarme, para os itens monitorados. O modelo define quatro níveis de alarme: muito baixo, baixo, médio e alto. A ontologia de domínio da casa do paciente é análoga à ontologia pessoal, só que nesse caso ela é responsável pela representação do conhecimento do ambiente, como temperatura e umidade.

A ontologia de gerenciamento de alarmes descreve uma política de alerta que determina os atores que serão notificados, conforme a gravidade da situação que o paciente está inserido. Dependendo do nível de alerta é necessário uma confirmação de recebimento da notificação. Caso o contato não tenha feito a confirmação dentro da janela de tempo adequada, uma nova notificação é enviada para outro contato. Caso nenhum contato tenha confirmado o recebimento, o sistema de emergência local será notificado.

A ontologia de contexto social representa os membros da rede de cuidado, conforme o papel que eles exercem no modelo, por exemplo, paciente, familiares, operadores de saúde e membros da comunidade. A ontologia também representa conceitos de disponibilidade e localização destes papéis, que são utilizadas pelas regras de inferência do sistema.

Os dados de contexto do modelo são inferidos por meio de duas abordagens: raciocínio baseado na ontologia e raciocínio baseado em regras. O raciocínio baseado na ontologia utiliza os dados existentes na base de conhecimento para classificar o conhecimento de acordo com os conceitos que a ontologia descreve. O raciocínio através de regras é utilizado para descrever regras de inferência de contexto como, por exemplo, nível de alerta ou atores que deverão ser alertados conforme a situação. Segundo os autores, essa abordagem permite flexibilidade na reconfiguração do sistema, uma vez que os valores limite dos sinais monitorados podem mudar de acordo com o paciente, assim como o número de aparelhos utilizados para monitorá-lo, que podem aumentar ou diminuir.

A avaliação da aceitação do modelo foi efetuada através de um experimento que abordou os seguintes casos: avaliação da efetividade das características do sistema na prática de

atividades diárias, análise das desvantagens detectadas e qualificação delas, e coleta da satisfação do usuário. O experimento foi conduzido em uma clínica de repouso com três grupos de usuários: clínico-gerais, enfermeiros e assistentes pessoais. Após o experimento foram obtidos os seguintes resultados: 60% dos usuários ficaram satisfeitos com as funcionalidades gerais do sistema; 91% consideraram úteis ou muito úteis as funcionalidades de descrever o perfil e disponibilidade para casos de intervenção; 91% dos usuários apreciaram a capacidade de modificação dos valores limites para alarme por paciente; 95% consideraram útil a funcionalidade de notificação de acordo com o nível de alerta.

3.6 Comparativo entre os trabalhos estudados

Para que o cuidado de DCNT seja eficiente não basta apenas o envolvimento do paciente. Também é necessário suporte e engajamento de membros da comunidade e organização de saúde, que devem dar suporte nas atividades de autogerenciamento do paciente, fazendo com que ele se sinta motivado e engajado nessas atividades. Dessa forma, um sistema de cuidado ubíquo de DCNT deve integrar esses diferentes papéis, e oferecer oportunidades de colaboração entre eles (WAGNER et al., 2001; ICIC, 2011; OMS, 2003). Outra característica desejável é que ele seja orientado ao gerenciamento e controle de fatores de riscos, que possuem relação direta com o surgimento de outras DCNT (OMS, 2005).

Assim, os trabalhos estudados serão comparados com relação ao suporte aos seguintes itens:

- propiciar o autogerenciamento do paciente;
- suportar o gerenciamento dos fatores de risco;
- oferecer suporte para integração entre paciente e recursos da comunidade;
- oferecer suporte para integração entre paciente e recursos das organizações de saúde;
- oferecer suporte para integração entre paciente e membros da comunidade;
- oferecer suporte para integração entre paciente e membros das organizações de saúde;
- oferecer suporte à educação dos envolvidos no cuidado de doenças crônicas.

A Tabela 2 apresenta um comparativo dos suportes ao cuidado de DCNT oferecidos pelos trabalhos estudados. As comparações foram organizadas em três categorias: autogerenciamento, gerenciamento dos fatores de risco e integrações com o paciente. Estas categorias correspondem a primeira coluna da Tabela 2 e são explicadas a seguir.

O suporte ao autogerenciamento é determinado através de objetivos e atividades estipulados por uma equipe de cuidados e que devem ser executados pelo paciente. O suporte para essas atividades pode ser feito através do auxílio na ingestão de alimentos saudáveis, práticas de exercícios físicos ou lembretes de cuidados que o paciente deve fazer continuamente. Com

exceção do CHRONIUS, todos os trabalhos estudados dão suporte ao autogerenciamento para alguma atividade.

O suporte ao gerenciamento dos fatores de risco apresenta um comparativo entre os trabalhos estudados, no que diz respeito às atividades de controle alimentar, execução de atividades físicas, medicamento e sensoriamento. Para as atividades de controle alimentar, apenas o MAHI e o WD apresentam algum tipo de suporte. O MAHI permite ao usuário registrar através de áudio ou fotografias as suas refeições. O WD oferece um aplicativo com uma lista de opções de alimentação, na qual o usuário pode selecionar para efetuar o registro das suas refeições. Essas funcionalidades permitem aos usuários lembrarem as opções feitas anteriormente, dando subsídio para efetuar escolhas mais adequadas nas próximas refeições.

Apenas o WD possui algum tipo de auxílio no gerenciamento de atividades físicas. O sistema permite a recepção de contagem de passos através de um pedômetro. No suporte a ingestão de medicamentos, os trabalhos apresentados por Koutkias et-al (KOUTKIAS et al., 2010) e Paganelli e Giuli (PAGANELLI; GIULI, 2011) geram notificações lembrando que o paciente deve tomar alguma medicação. O modelo apresentado por Koutkias et-al. também faz o monitoramento de reações adversas de medicamentos, através do monitoramento de variações de sinais vitais do paciente.

Todos os trabalhos estudados realizam algum tipo de sensoriamento com o intuito de monitorar o progresso do tratamento do paciente, em algum tipo de atividade de prevenção e controle de fator de risco. MAHI, WD e CHRONIUS concentram-se no monitoramento de tipos específicos de sinais vitais como, por exemplo, frequência cardíaca ou temperatura. Por sua vez, os trabalhos de Koutkias et-al. e Paganelli e Giuli apresentam modelos que abstraem o mecanismo de monitoramento, permitindo a configuração das informações que serão coletadas, como também a configuração de níveis limite superiores e inferiores para os sinais lidos, que são utilizados para a geração de alertas.

O suporte à integração com o paciente abrange os meios pelos quais os trabalhos estudados utilizam os recursos oferecidos pela comunidade e organizações de saúde para auxiliar o paciente na sua rotina de cuidados. Nessa categoria são comparados os seguintes itens: “Integração Paciente x Recursos Comunidade/Organização Saúde”; “Integração Paciente x Membros Comunidade”; “Integração Paciente x Membros Organização Saúde”; e “suporte à educação”. Entende-se por “Integração Paciente x Recursos Comunidade/Organização Saúde” a forma como os trabalhos relacionados auxiliam o paciente a localizar serviços e recursos próximos, que podem ser úteis no seu tratamento ou bem-estar. Estes recursos podem ser uma farmácia que oferece um medicamento com um preço mais acessível, um parque público com aparelhos para exercício, ou aulas públicas de algum tipo de atividade física. Nenhum dos trabalhos possui um mecanismo que possibilite a busca por estes tipos de recursos. A ontologia de contexto social exposta por Paganelli e Giuli (PAGANELLI; GIULI, 2011) oferece uma descrição para este tipo de suporte, mas esse suporte não é explorado no seu protótipo.

O item “Integração Paciente x Membros Comunidade” indica se o trabalho possui algum

mecanismo que possibilite o contato com membros da comunidade, como forma de obter suporte em algum tipo de atividade de cuidado de DCNT. Por exemplo, para um paciente que tenha diabetes, esse mecanismo poderia recomendar algum indivíduo próximo, que possa auxiliá-lo na aplicação de insulina. Nesse sentido, o WD possibilita ao seu usuário enviar dados que ele registrou, para algum contato escolhido por ele.

A “Integração Paciente x Membros Organização Saúde” aponta a existência de mecanismos de comunicação entre paciente e organizações de saúde, e mecanismos que permitam o envio de alertas para os membros destas organizações. Estes alertas podem ser gerados para informar a necessidade de algum tipo de intervenção, ou para comunicar que o plano de tratamento estipulado não está surtindo efeito. De alguma maneira, todos os trabalhos estudados oferecem suporte a essa funcionalidade, ou através da seleção manual dos destinatários pelo usuário, ou pela geração automática de alertas para o corpo de cuidado.

O “suporte à educação” pode ser feito de diferentes formas. Ele pode ser feito através do fornecimento de material educativo eletrônico para instrução dos envolvidos nos processos de cuidado; na orientação fornecida por membros da organização de saúde, ou da comunidade, para os envolvidos no processo de cuidado; ou até mesmo através do próprio processo de auto-gerenciamento - casos do MAHI e WD.

De acordo com a Tabela 2, percebe-se que a característica Integração Paciente x Recursos Comunidade/Organização de saúde não é explorada pelos trabalhos estudados, e a característica Integração Paciente x Membros Comunidade é explorada por apenas um dos trabalhos. Os modelos de gerenciamento de condições crônicas CCM (WAGNER et al., 2001; ICIC, 2011) e CICC (OMS, 2003) citam a necessidade de suporte a essas características. Dessa forma, entende-se que sistemas ubíquos para cuidado de DCNT devam oferecer suporte a elas. Além disso, nenhum dos trabalhos ofereceu suporte a todas as características relacionadas ao Gerenciamento dos fatores de risco. De maneira geral, os trabalhos concentram-se no suporte ao sensoriamento e mais uma das atividades de gerenciamento de fatores de risco, por exemplo, controle alimentar, atividade física ou ingestão de medicamento.

O próximo capítulo apresenta um modelo para cuidado ubíquo de DCNT, que tem por objetivo integrar os conceitos comparados nesta seção.

Tabela 2: Comparação de suportes oferecidos pelos trabalhos relacionados

Tipo de Suporte	Característica	MAHI	WD	CHRONIUS	Koutkias et-al.	Paganelli e Giuli
Auto-gerenciamento	Suporte ao autogerenciamento	Sim	Sim	Não se aplica	Sim	Sim
Gerenciamento dos Fatores de Risco	Controle Alimentar	Imagens e áudio. Site de consulta	Aplicativo de registro	Não	Não se aplica	Não
	Atividade Física	Não se aplica	Carga de dados de pedômetro	Não se aplica	Não se aplica	Não se aplica
	Medicamento	Não	Não se aplica	Não	Controle de efeitos adversos e lembretes	Lembretes
	Sensoriamento	Glicosímetro	Pedômetro e registro manual de peso e consumo de alimentos	Frequência cardíaca e respiratória, Volume respiratório, Temperatura e umidade ambiente, atividade motora e detecção de tosse e ronco	Sensoriamento genérico de sinais vitais	Sensoriamento genérico de sinais vitais e ambientais
Integrações com Paciente	Integração Paciente x Recursos Comunidade/ Organização Saúde	Não	Não	Não se aplica	Não	Não
	Integração Paciente x Membros Comunidade	Não	Envio de mensagens manuais	Não	Não	Não
	Integração Paciente x Membros Organização Saúde	Contato com educador	Envio de mensagens manuais	Envio de alertas automáticos	Envio de alertas automáticos	Envio de alertas automáticos ou manuais
	Suporte à educação	Através do autogerenciamento e contato com educador	Através do autogerenciamento	Não	Não	Não

4 MODELO U'DUCTOR

Com base nas informações previamente descritas, será apresentado nesse capítulo o U'Ductor, um modelo para cuidado ubíquo de DCNT.

4.1 Modelo Organizacional do U'Ductor

O modelo organizacional dos elementos participantes do U'Ductor é inspirado no modelo de abstração de entidades utilizado pelo Continuum (COSTA, 2008). Esse modelo é composto por sete entidades, CoDimension, CoCell, CoNode, CoBase, CoMobi, CoGadget e CoPerson, que relacionam-se por meio de associações, composições ou agregações, podendo assim representar, de maneira abstrata, um modelo do mundo físico.

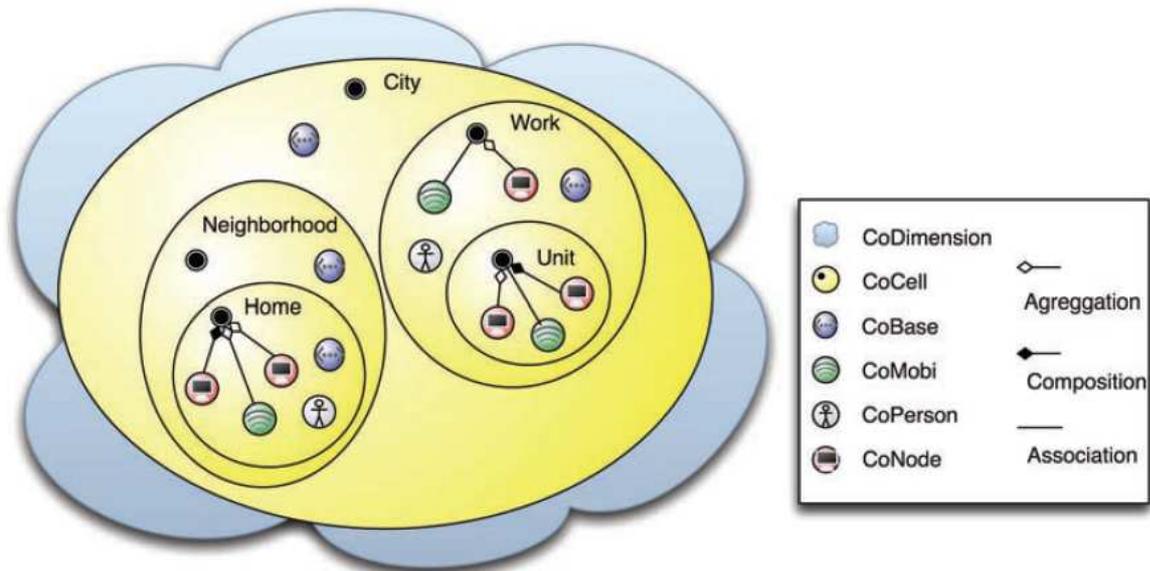
A Figura 9 mostra um exemplo de modelo criado a partir das entidades e relacionamentos utilizados pelo Continuum. A CoDimension é uma super-entidade, que engloba todas as outras entidades existentes no modelo. A CoDimension é composta por uma entidade CoCell, que representa uma localização física, no caso do exemplo uma cidade (identificada na figura como City). Esta cidade é composta por uma entidade CoBase e outras duas CoCells: Neighborhood (Vizinhança) e Work (Trabalho). A entidade CoBase é um dispositivo encarregado pelo gerenciamento de uma CoCell, ou um grupo de CoCells dependentes a esta CoCell, e pela interação entre as CoCells. A CoCell Vizinhança é composta pela CoCell Casa (identificada na figura como Home). A Casa é composta por uma CoBase e um CoNode. Agregadas à Casa estão uma entidade CoMobi e outro CoNode. Além disso, associada a esta Casa, há uma entidade do tipo CoPerson. As entidades CoNode e CoMobi representam dispositivos que executam os serviços do Continuum, eles são distintos pois CoNodes representam entidades fixas e CoMobis entidades móveis. CoPerson representa um usuário do sistema Continuum presente em alguma CoCell. Além das entidades descritas, existe ainda a CoGadget, que representa um dispositivo pessoal, tal como um PDA ou smartphone.

No modelo do U'Ductor foram criados seis elementos: Mundo, Nodo Local (NL), Nodo Pessoal (NP), Serviço de Resolução Posição <-> Nodo Local (SRP-NL), Contexto e Recursos. O elemento Mundo é a infraestrutura de rede utilizada para comunicação entre o SRP-NL, NLs e NPs.

Ambos os tipos de nodos (NP e NL) compartilham recursos e informações de contexto. Segundo Fielding (FIELDING; TAYLOR, 2000), Recurso é “qualquer informação que possa ser nomeada”, sendo que esta informação, ou recurso, pode possuir mais de uma representação. Por exemplo, uma instrução de aplicação de medicamento poderia estar disponível em áudio ou vídeo. Contexto é toda informação que representa a situação de uma entidade (DEY; ABOWD; SALBER, 2001), estas informações podem ser o tipo de atividade que um NP efetua (por exemplo, caminhando, corrento, parado), ou a temperatura ambiente de um NL.

Um NL representa uma região espacial disposta em algum local geográfico como, por exem-

Figura 9: Exemplo do modelo de entidades do Continuum



Fonte: (COSTA, 2008)

plô, uma casa. No caso do U'Ductor o NL é análogo a junção das entidades CoCell e CoBase do Continuum. O NL mantém referências para os recursos, pessoas e informações de contexto do local, como por exemplo, temperatura, umidade do ar, ou qualquer outra informação relevante. Assim, seria possível auxiliar habitantes de uma casa a encontrarem sensores de ambiente ou recursos de auxílio, como uma balança ou um medicamento.

Um NP representa uma pessoa de posse de um PDA ou smartphone, que movimenta-se entre espaços físicos, representados pelos NL. Dessa forma, um NP é análogo as entidades CoPerson e CoGadget do Continuum. Quando é identificada uma troca de NL, o NP conecta-se a esse novo nodo, e passa a fazer parte dos NPs associados àquele local. O NP utiliza os NLs para encontrar elementos que possam auxiliar o seu usuário em algum tópico de interesse. Esses elementos podem ser uma outra pessoa, um recurso de auxílio ao tratamento, ou alguma informação de contexto. Além disso, da mesma forma que os NLs, um NP poderá compartilhar recursos que possam ser úteis a outros NPs, ou disponibilizar informações de contexto. Dessa maneira, seria possível a um médico ou familiar que utilize um NP, monitorar o paciente em tempo real, ou que um paciente possa encontrar um usuário de NP que possua conhecimento em um tópico de tratamento no qual ele tem dúvida.

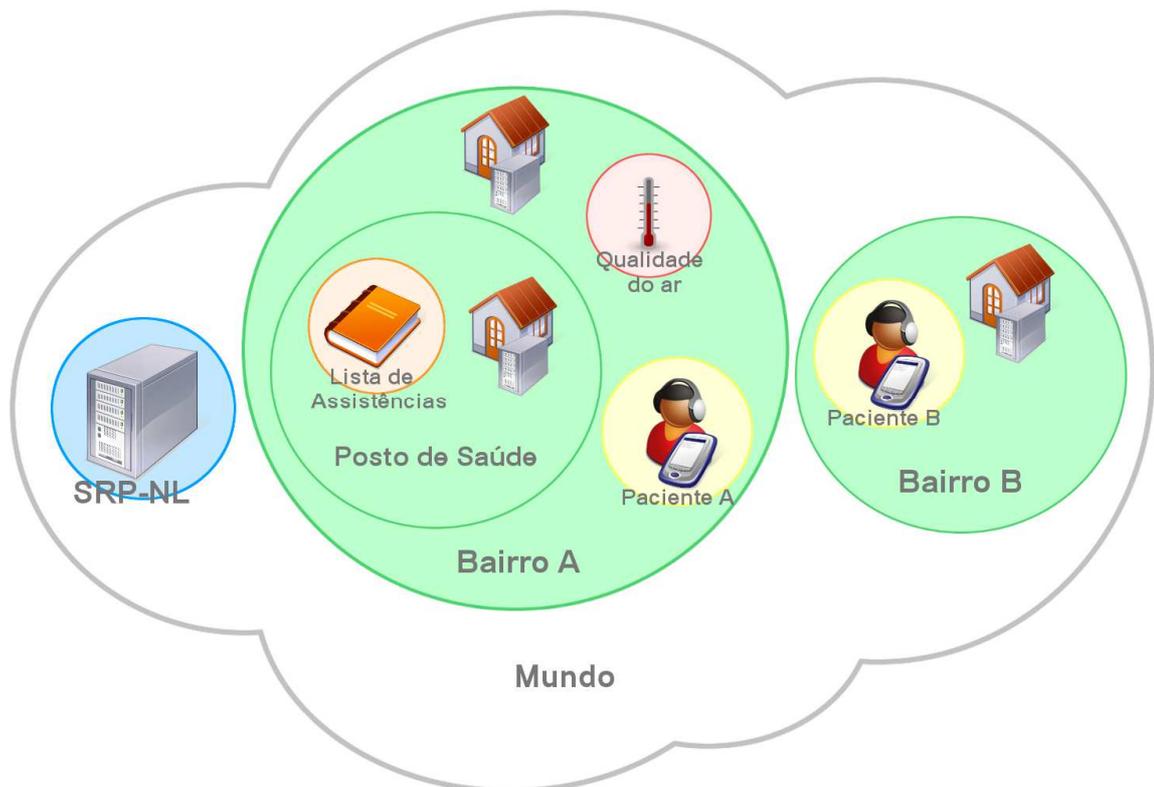
O SRP-NL é responsável por identificar o local onde um NP se encontra e associá-lo ao NL que representa esse local. Essa associação possibilita ao NP encontrar recursos, pessoas e outros lugares disponíveis nesse local, como também receber informações de contexto desse local, e.g. a temperatura do local.

O modelo descrito forma uma rede P2P onde os nodos compartilham algum tipo de conteúdo que é acessível diretamente a outros nodos sem a necessidade de um intermediário (SCHOLLMEIER, 2001). Esses nodos ficam então organizados através de uma estrutura de hierarquia de locais

(BISCHOFFS et al., 2004), que facilita a busca de recursos de interesses, pois estes devem estar localizados em nodos próximos.

A Figura 10 apresenta um exemplo de como os elementos do U'Ductor interagem entre si. Nesse exemplo, um SRP-NL está disponível em algum local do Mundo. Os NLs (Posto de Saúde, Bairro A e Bairro B) utilizaram esse SRP-NL para registrarem-se como um local, e os NPs (Paciente A e Paciente B) utilizaram esse serviço para identificarem o NL que representa o local onde eles se encontram, e registrarem-se como estando nesse local. O NL Bairro A disponibiliza informações de qualidade do ar. Esta informação é relevante ao Paciente B que possui problemas respiratórios e dirige-se a este bairro. O Paciente A possui diagnóstico de hipertensão e encontra-se em deslocamento pelo Bairro A. Ele pode utilizar seu PDA que possui os serviços do U'Ductor instalados, para localizar algum local que faça medição de pressão arterial. O Posto de Saúde compartilha uma lista de assistências disponíveis, dentro dessa lista está incluída o suporte a necessidade do Paciente A. Dessa forma, aplicações executadas no PDA deste paciente podem utilizar os serviços do U'Ductor para notificar o Paciente A que existe suporte a sua necessidade no Posto de Saúde do Bairro A.

Figura 10: Exemplo de organização de elementos do U'Ductor



Fonte: Elaborado pelo autor.

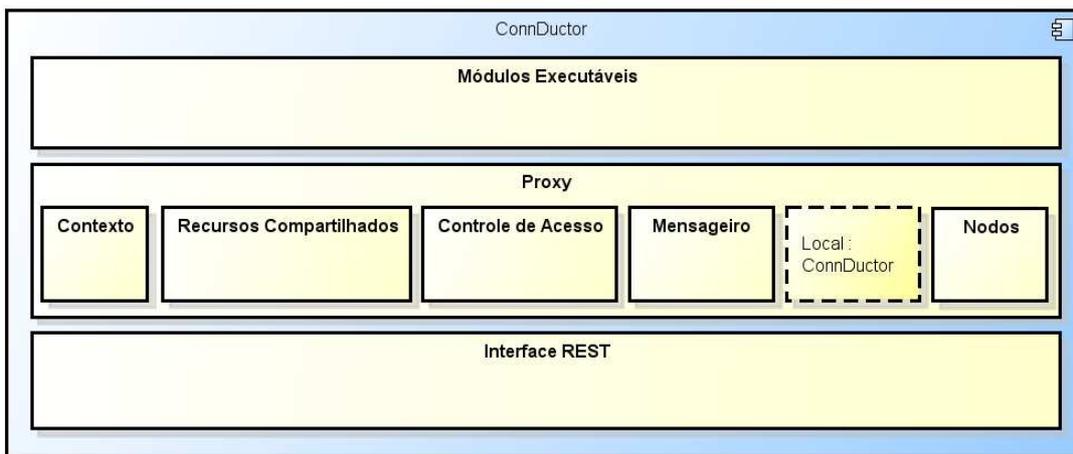
4.2 Middleware ConnDuctor

O middleware ConnDuctor é responsável por oferecer funcionalidades de comunicação, localização de recursos, notificação de eventos de mudança de contexto, e mecanismos de controle de acesso a recursos. As próximas seções explicam os componentes que compõem o middleware, e como as suas funcionalidades são oferecidas aos nodos do modelo.

4.2.1 Componentes do Middleware ConnDuctor

O ConnDuctor é formado por nove componentes: Módulos Executáveis, Proxy, Contexto, Recursos Compartilhados, Controle de Acesso, Mensageiro, Local, Nodos e Interface REST. A funcionalidade de cada componente será explicada nesta seção. A Figura 11 ilustra como esses componentes estão organizados dentro do ConnDuctor.

Figura 11: Componentes do middleware ConnDuctor



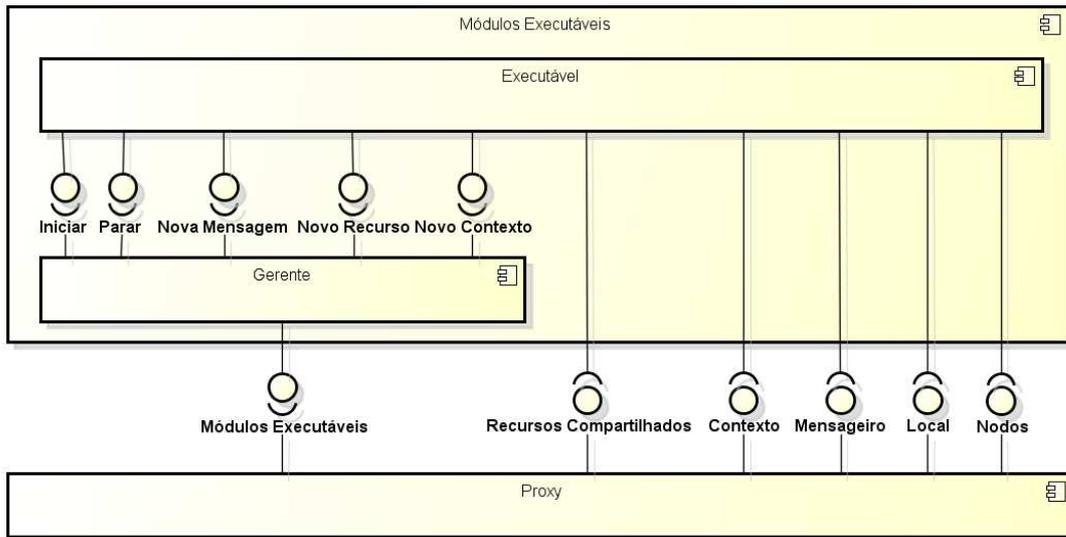
Fonte: Elaborado pelo autor.

4.2.1.1 Módulos Executáveis

Os módulos executáveis permitem a utilização dos recursos do middleware por meio de módulos criados por desenvolvedores e registrados no ConnDuctor. Estes módulos possuem a capacidade de receber as seguintes notificações do middleware: mensagens recebidas pelo nodo; criação, alteração e exclusão de recursos compartilhados; e alterações de contextos. Além disso, os módulos executáveis podem acessar as funcionalidades dos componentes de Contexto, Recursos Compartilhados, Controle de Acesso, Mensageiro, Local e Nodos.

A Figura 12 é um diagrama de componentes na notação UML que mostra a arquitetura do componente de Módulos Executáveis. Este componente é formado por um componente Gerente e por zero ou mais instâncias do componente Executável. O componente Gerente é responsável por iniciar os Executáveis e notificá-los da ocorrência de novas mensagens, recursos,

Figura 12: Arquitetura do componente de Módulos Executáveis



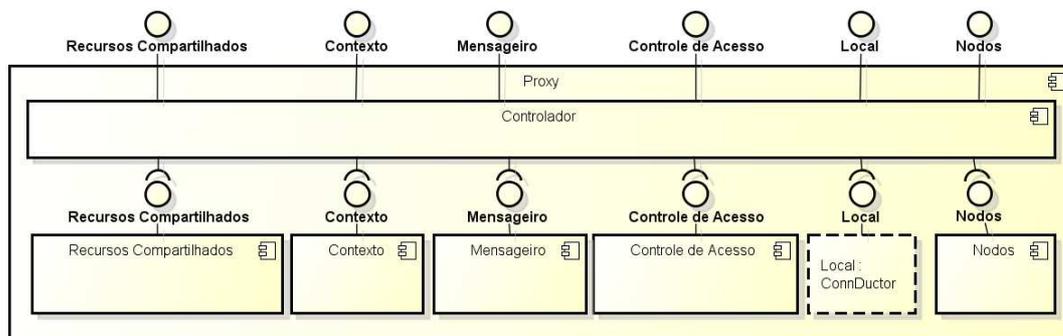
Fonte: Elaborado pelo autor.

ou contextos recebidos pelo middleware. Os Executáveis poderão acessar as funcionalidades dos componentes de Recursos Compartilhados, Contexto, Mensageiro, Local e Nodos do middleware através do componente Proxy.

4.2.1.2 Proxy

O componente Proxy é encarregado de verificar se um cliente do middleware, isto é, um nodo externo, ou Executável, possui acesso a uma funcionalidade requisitada. O acesso à funcionalidade será negado caso o cliente não possua permissão. As autorizações de acesso são gerenciadas pelo componente de Controle de Acesso.

Figura 13: Arquitetura do componente de Proxy



Fonte: Elaborado pelo autor.

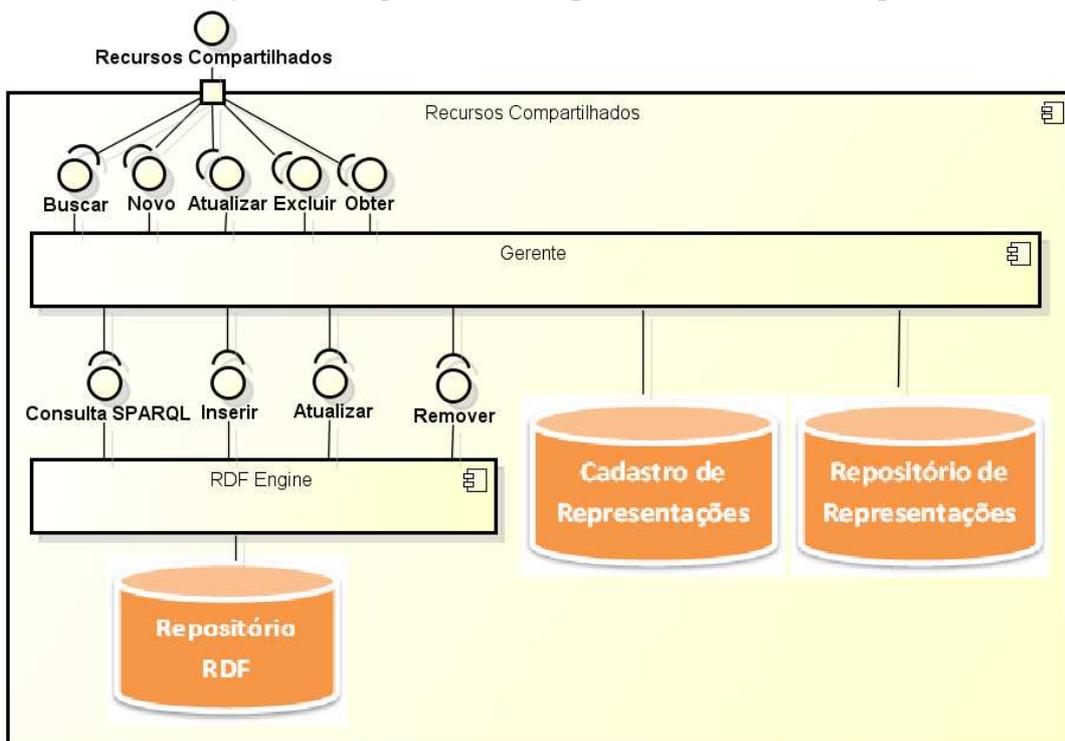
Na Figura 13 é mostrada a arquitetura do componente Proxy. O componente Controlador é utilizado como intermediário no acesso às funcionalidade dos componentes de Recursos Compartilhados, Contexto, Controle de Acesso, Local e Nodos pelos clientes. Sempre que um

cliente solicitar o acesso a uma funcionalidade do middleware, o Controlador utilizará o componente de Controle de Acesso para verificar se esse cliente possui autorização de acesso a funcionalidade requisitada.

4.2.1.3 Recursos Compartilhados

Recurso Compartilhado é qualquer tipo de conteúdo compartilhado por um nodo que possua capacidade de auxiliar os usuários do sistema na execução de alguma de suas atividades. Um nodo deve oferecer no mínimo a descrição semântica do recurso de forma que elementos de software, que executam em outros nodos da rede, possam localizar conteúdos, conhecimentos, dispositivos, pessoas, locais ou objetos necessários ao cumprimento de suas atividades. A descrição semântica é feita com o uso de uma ontologia que descreve o significado do recurso compartilhado para as aplicações executadas no modelo U'Ductor;

Figura 14: Arquitetura do componente de Recursos Compartilhados



Fonte: Elaborado pelo autor.

A arquitetura do componente de Recursos Compartilhados é apresentada na Figura 14. Internamente ao componente de Recursos Compartilhados estão os componentes Gerente e RDF Engine. O componente Gerente é encarregado por resolver a solicitação feita por algum cliente. O Gerente utiliza o Cadastro de Representações para armazenar a localização dos recursos compartilhados pelo nodo no sistema de arquivos, e para armazenar as informações de apresentação desses recursos. A informação de apresentação determina os formatos em que o recurso pode ser apresentado como, por exemplo, texto, imagem, ou vídeo. O Repositório de

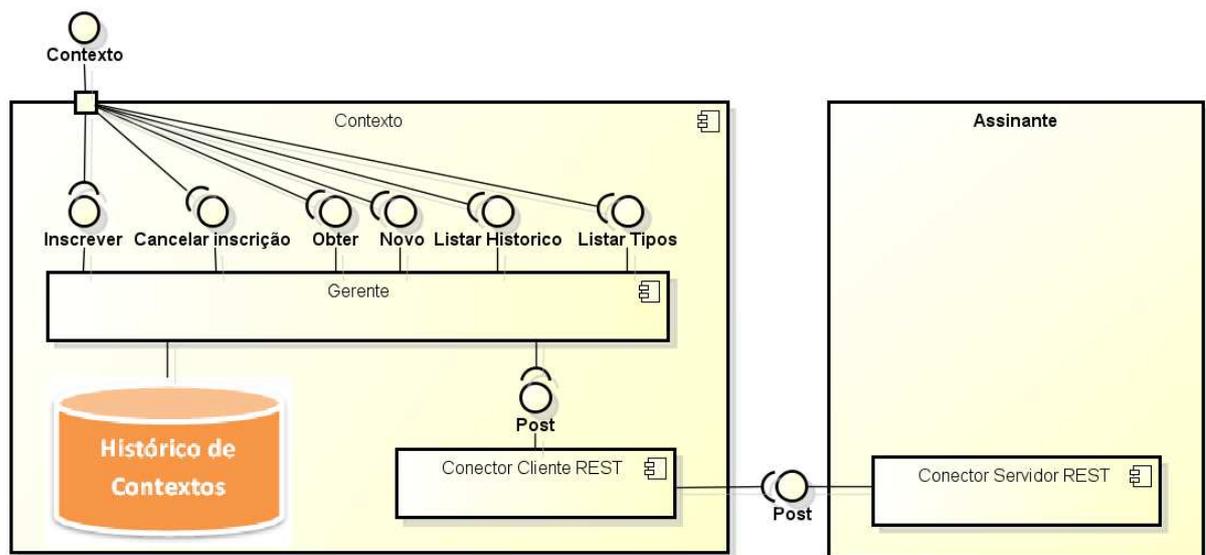
Representações é a área do sistema de arquivos onde estão armazenados os recursos em seus diferentes formatos de apresentação.

O componente RDF Engine é utilizado para gerenciar as descrições semânticas dos recursos compartilhados. Estas descrições são feitas no formato RDF e armazenadas no Repositório RDF. Outra característica do RDF Engine é permitir a execução de consultas no formato SPARQL (PRUD'HOMMEAUX; SEABORNE, 2008). SPARQL é uma linguagem de consulta para RDF, e é utilizada pelo Gerente para encontrar recursos compartilhados conforme os critérios informados pelos clientes.

4.2.1.4 Contexto

O componente de Contexto permite aos clientes e módulos do middleware registrarem o seu interesse nos atributos que identificam a situação de um nodo em particular, e receberem notificações das alterações desses atributos. Caso o nodo represente uma pessoa esses atributos poderiam ser a sua temperatura, pressão arterial, índice glicêmico, ou qualquer outro atributo relevante que a pessoa compartilhe. No caso de um local, esses atributos podem ser o índice de qualidade do ar, umidade relativa do ar, índice de raios ultravioleta, ou outra informação pertinente compartilhada pelo local. Outra funcionalidade deste módulo é o armazenamento de informações de contexto, geradas ou recebidas pelo nodo. Dessa maneira o nodo mantém um histórico de contextos, ou trilhas (SILVA et al., 2010), permitindo às aplicações executadas pelo middleware anteciparem situações que um usuário possa enfrentar através da inferência desse histórico (DEY; ABOWD; SALBER, 2001). Por exemplo, identificar a ocorrência de um enfarte.

Figura 15: Arquitetura do componente de Contexto



Fonte: Elaborado pelo autor.

O funcionamento do componente de Contexto é baseado no framework conceitual de ge-

renciamento de contexto definido por Dey (DEY; ABOWD; SALBER, 2001). No caso do U'Ductor o componente de Contexto funciona com um agregador (aggregator) de contexto, encarregado de receber novas informações de contexto, que chegam através dos dispositivos de entrada de contexto (widgets). No modelo U'Ductor os widgets são implementados pelos Executáveis e pelos nodos. Os componentes Gerente e Conector Cliente REST integram o componente de Contexto (Figura 15). As principais atribuições do Gerente são: gerenciar as inscrições de contexto, isto é, inscrever e cancelar a assinatura de clientes; armazenar as informações de contexto recebidas dos widgets no repositório de Histórico de Contextos; e enviar notificações de troca de contexto aos assinantes. O envio de notificações é feito através do Conector Cliente REST. No estilo arquitetural REST os conectores disponibilizam uma interface genérica que é utilizada para comunicações entre clientes e servidores, sendo o método Post utilizado para criar uma novo contexto no servidor assinante.

Figura 16: Linha de tempo de histórico de contexto

Tipos de Contexto	Localização	Trabalho	Arr	staurant			
	Tipo de Movimento	Parado	Andando			Parado	
	Estado do Telefone	Livr		Ocupado	Livr		

Fonte: Elaborado pelo autor.

Tabela 3: Contextos armazenados

Tipo de Contexto	Valor	Data - Hora
Localização	Trabalho	03/01/2013 - 12h00m
Tipo de Movimento	Parado	03/01/2013 - 12h00m
Estado do Telefone	Livre	03/01/2013 - 12h00m
Tipo de Movimento	Andando	03/01/2013 - 12h05m
Localização	Arredores do Trabalho	03/01/2013 - 12h10m
Estado do Telefone	Ocupado	03/01/2013 - 12h15m
Localização	Restaurante	03/01/2013 - 12h20m
Estado do Telefone	Livre	03/01/2013 - 12h20m
Tipo de Movimento	Parado	03/01/2013 - 12h25m

Fonte: Elaborado pelo autor.

A Figura 16 exibe uma linha de tempo dos contextos de Localização, Tipo de Movimento, e Estado do Telefone, de um nodo. Inicialmente o nodo está no local identificado como “Trabalho”, com tipo de movimento igual a “Parado”, e o estado do telefone igual a “Livre”. Às 12h05m ocorre uma troca de contexto, no qual o tipo de movimento é alterado de “Parado” para “Andando”, o que indica que a entidade que representa o nodo encontra-se em deslocamento.

Tabela 4: Trilha inferida

Elemento da Trilha	Localização	Tipo de Movimento	Estado do Telefone	Data - Hora
1	Trabalho	Parado	Livre	03/01/2013 - 12h00m
2	Trabalho	Andando	Livre	03/01/2013 - 12h05m
3	Arredores do Trabalho	Andando	Livre	03/01/2013 - 12h10m
4	Arredores do Trabalho	Andando	Ocupado	03/01/2013 - 12h15m
5	Restaurante	Andando	Livre	03/01/2013 - 12h20m
6	Restaurante	Parado	Livre	03/01/2013 - 12h25m

Fonte: Elaborado pelo autor.

Às 12h10m ocorre uma troca de contexto, na qual a localização é alterada de “Trabalho” para “Arredores do Trabalho”. Às 12h15m o estado do telefone é alterado de “Livre” para “Ocupado”, o que indica que a entidade que representa o nodo está atendendo uma ligação. Às 12h20m a localização é alterada de “Arredores do Trabalho” para “Restaurante” e o estado do telefone é alterado de “Livre” para “Ocupado”. Às 12h25m o tipo de movimento é alterado de “Andando” para “Parado”, o que indica que a entidade que representa o nodo chegou a seu destino.

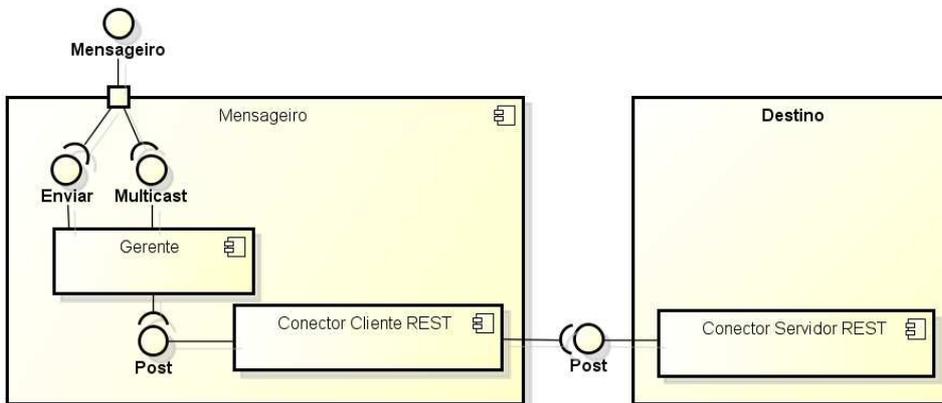
Sempre que ocorrer uma de troca de contexto será criado um registro no repositório de Histórico de Contexto contendo a data, hora, tipo de contexto e o valor deste novo contexto. A Tabela 3 exibe os registros que serão armazenados no repositório de Histórico de Contexto para o exemplo da Figura 16. A Tabela 4 apresenta a trilha que pode ser inferida a partir do histórico de contextos apresentado na Tabela 3. Cada registro do histórico é a base para um elemento da trilha. O elemento da trilha é composto pelo valor, data e hora desse registro, e pelo último valor de cada tipo de contexto diferente desse registro, e que tenha sido armazenado antes dele.

4.2.1.5 Mensageiro

O componente Mensageiro é utilizado para enviar mensagens entre os nodos da rede. O ConnDuctor não define um formato específico para estas mensagens, permitindo as aplicações que utilizem o middleware criarem protocolos de mensagem específicos aos seus propósitos.

De acordo com a solicitação feita ao Mensageiro, o Gerente deverá executar o envio da mensagem para um único nodo, no caso do envio ser do tipo unicast, ou para todos os nodos que estão associados, caso o envio ser do tipo multicast. O Conector REST é responsável pela entrega da mensagem aos nodos de destino. Os componentes do Mensageiro são exibidos na Figura 17.

Figura 17: Arquitetura do componente Mensageiro

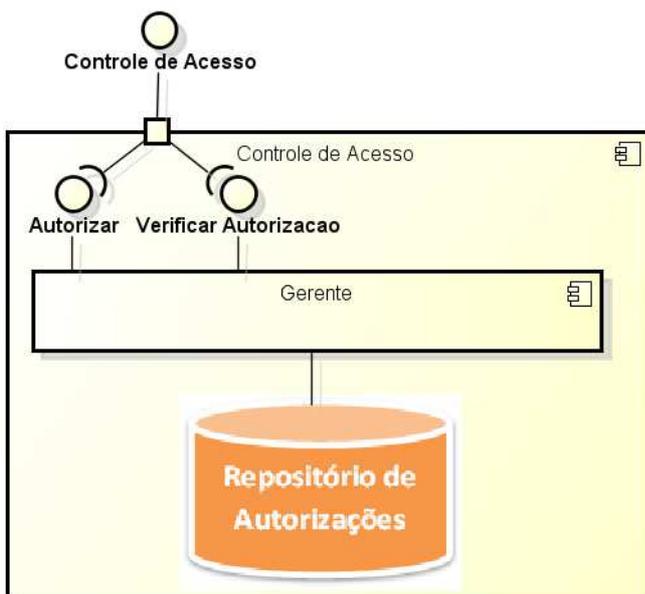


Fonte: Elaborado pelo autor.

4.2.1.6 Controle de Acesso

O componente Controle de Acesso é encarregado por gerenciar as autorizações de acesso dos clientes às funcionalidades dos componentes de Contexto, Recursos Compartilhados, Controle de Acesso, Mensageiro, Local e Nodos. Por exemplo, a informação que indica que um NP está autorizado a receber atualizações de temperatura de um NL é armazenada por este componente. Da mesma maneira, a informação indicando que o NP de uma nutricionista está autorizado a acessar a imagem de uma refeição armazenada no NP de seu paciente.

Figura 18: Arquitetura do componente Controle de Acesso



Fonte: Elaborado pelo autor.

A Figura 18 apresenta a arquitetura do componente de Controle de Acesso. O Gerente é responsável por registrar e validar as autorizações de acesso dos clientes às funcionalidades

do nodo. As autorizações às funcionalidades do nodo são armazenadas no Repositório de Autorizações, e são compostas por três informações: identificador do nodo, URI e método.

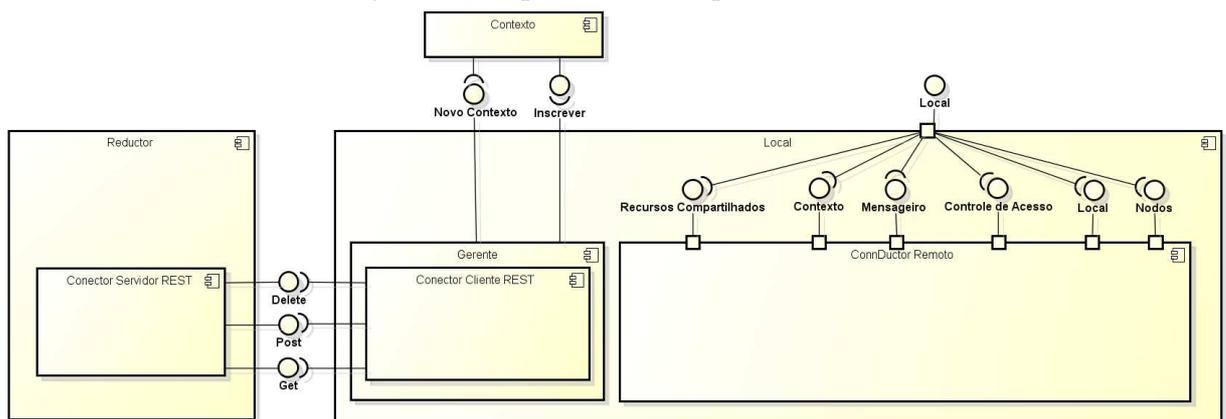
O identificador do nodo determina qual é o nodo proprietário da autorização. O URI é utilizado para identificar a funcionalidade ou recurso do middleware que o nodo possui autorização. Por exemplo, “/contexto/inscricao” identifica a funcionalidade de assinatura de contexto, enquanto que “/recursos-compartilhados/manual-instrucao-monitor-pressao”, identifica um recurso compartilhado que encontra-se armazenado no nodo. O método descreve o tipo de ação que um nodo está autorizado a executar como, por exemplo, seleção, criação, atualização ou exclusão.

Os URIs são montados pelo Controlador do componente Proxy. O Controlador também é encarregado por efetuar as requisições ao Controle de Acesso, quando são enviados o identificador do nodo, o URI e o tipo de método.

4.2.1.7 Local

O componente Local representa uma referência ao NL onde encontra-se um nodo NP ou NL. Essa referência é utilizada para que o nodo possa acessar as funcionalidades compartilhadas pelo NL, podendo assim encontrar oportunidades de interesse. Para acessar os recursos de um nodo externo o componente Local possui um componente chamado ConnDuctor Remoto, que é responsável por executar as solicitações de acesso às funcionalidades do middleware no nodo remoto (Figura 19).

Figura 19: Arquitetura do componente Local



Fonte: Elaborado pelo autor.

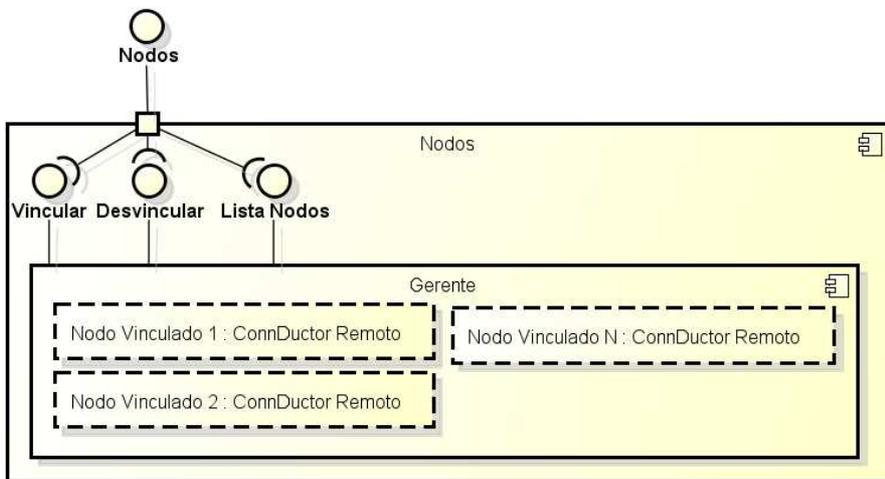
O componente Gerente é encarregado por receber as atualizações de localização GPS do componente de Contexto, e enviar estas informações para o Reductor, a fim de identificar se houve uma troca de NL. Caso haja uma troca de NL, o Gerente removerá o vínculo no NL onde se encontrava, se registrará no NL onde está, e atualizará a referência de nodo externo do componente ConnDuctor Remoto.

4.2.1.8 Nodos

O componente Nodos permite que nodos remotos se vinculem ao nodo. O componente interno Gerente armazena as associações aos nodos vinculados como referências do tipo ConnDuctor Remoto (Figura 20).

O componente Nodos e o componente Local se complementam, possibilitando que os nodos naveguem entre si, promovendo a localização de recursos necessários ao cumprimento de suas atividades.

Figura 20: Arquitetura do componente Nodos



Fonte: Elaborado pelo autor.

4.2.1.9 Interface REST

A interface REST garante o acesso remoto às funcionalidades dos NPs e NLs. REST é a abreviação de Representational State Transfer, ou Transferência de Estado Representacional (FIELDING; TAYLOR, 2000), sendo uma abstração do modelo da arquitetura Web, onde componentes comunicam-se por meio de uma interface uniforme para requisitar e transferir a representação do estado de algum recurso. O estilo arquitetural REST é uma alternativa para construção de aplicações orientadas a serviços, e é utilizado pelo middleware ConnDuctor.

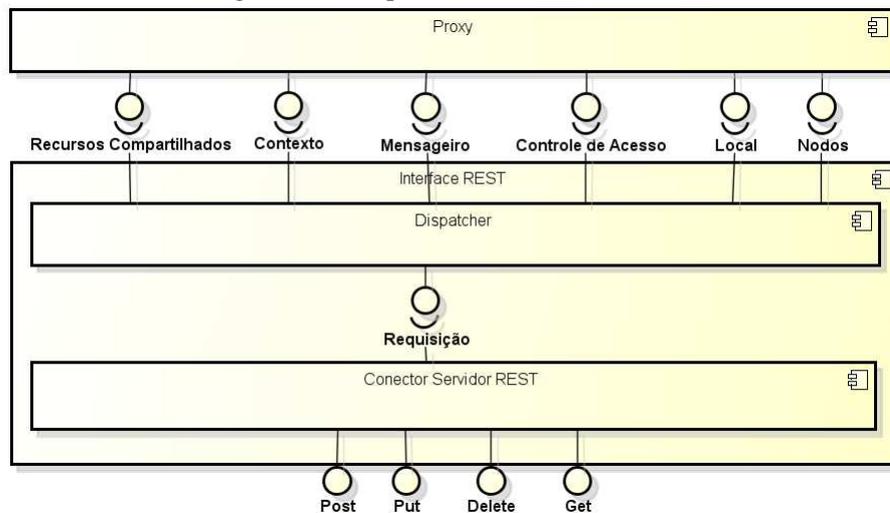
Em REST, um componente é uma abstração de alguma entidade cliente ou servidor. Clientes e servidores comunicam-se entre si através de conectores que abstraem o entendimento do protocolo de requisição e resposta utilizado entre eles¹, isto é, abstraem a sua interface uniforme. O conector de servidor é encarregado por interpretar a requisição feita pelo cliente para determinado recurso e transferir uma representação desse recurso ao cliente. Desta forma, conectores cliente e servidor podem ser encapsulados para oferecer diferentes funcionalidades, tais como

¹ Podem existir conectores para protocolos distintos, por exemplo, conectores cliente e servidor para protocolo SMTP, como conectores cliente e servidor para o protocolo HTTP, etc.

segurança, cache ou tunelamento de informações. A representação determina a forma como um recurso pode ser oferecido. Por exemplo, um livro pode ser oferecido em um formato de áudio ou texto. Por fim, um recurso é definido como “qualquer informação que possa ser nomeada” (FIELDING; TAYLOR, 2000).

O estilo arquitetural REST foi projetado para garantir alta escalabilidade, assim um dos requisitos existentes nesse estilo é não manter sessões de clientes armazenadas nos servidores. Essa característica garante algumas vantagens como a possibilidade de substituição de um componente, ou a paralelização de requisições através de vários servidores. A portabilidade é outra característica desse estilo. Em uma aplicação REST um mesmo recurso pode ser apresentado por formatos diferentes, dessa forma, caso o cliente não possa exibir o formato padrão ele pode utilizar um formato alternativo disponibilizado pelo servidor (FIELDING; TAYLOR, 2000). A facilidade de implementação de serviços é outro ponto positivo desse estilo, já que o número de decisões arquiteturais é menor em relação às implementações de serviços baseados em WSDL/SOAP (PAUTASSO; ZIMMERMANN; LEYMAN, 2008). Estas características foram levadas em consideração para que os serviços do ConnDuctor tivessem sido implementados utilizando o estilo arquitetural REST.

Figura 21: Arquitetura da Interface REST



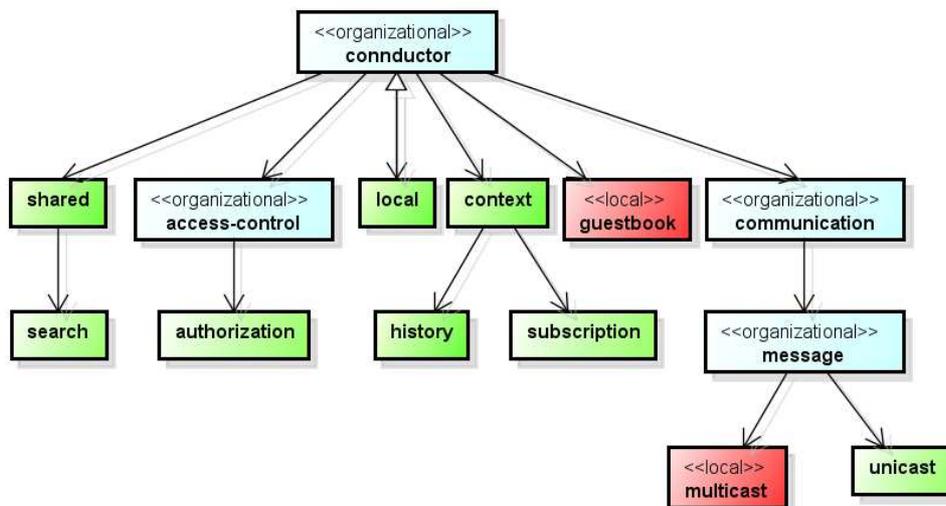
Fonte: Elaborado pelo autor.

A Figura 21 apresenta a arquitetura de como é estruturado o componente de Interface REST proposto pelo U'Ductor. As requisições chegam ao nodo através do Conector Servidor REST que permite aos clientes externos acessarem as funcionalidades do middleware. As requisições são encaminhadas ao componente Dispatcher que é encarregado de interpretá-las e acionar o componente responsável pela execução da funcionalidade requisitada.

A Figura 22 apresenta a hierarquia de serviços que são disponibilizados pela interface REST do ConnDuctor. Cada retângulo representa um elemento de endereçamento para o serviço, e as arestas representam a ligação de um elemento superior com seus dependentes. A estrutura tem início com o elemento raiz connductor, os URIs para os serviços são construídos a

partir da concatenação dos elementos superiores com seus subordinados, por exemplo, /conductor/shared, /conductor/guestbook, /conductor/communication/message/unicast, etc. Os retângulos que possuem o estereótipo “organizational” não disponibilizam funcionalidades, tendo um papel apenas organizacional. Os retângulos com o estereótipo “local” são serviços oferecidos apenas por NLs. Os retângulos sem estereótipo representam os serviços oferecidos por NPs e NLs. A Tabela 5 descreve esses serviços.

Figura 22: Estrutura hierárquica dos serviços REST do ConnDuctor



Fonte: Elaborado pelo autor.

4.3 ReDuctor

O SRP-NL, chamado de ReDuctor, é o serviço responsável por transformar as informações de geoposicionamento de um nodo em uma referência de NL. A Figura 23 mostra, em alto nível, como ocorre a negociação de um novo local. Na etapa 1, o módulo de Contexto informa ao ConnDuctor, executado a partir de um NP, que ocorreu uma alteração no seu posicionamento. O ConnDuctor solicita ao ReDuctor uma nova referência de NL (etapa 2). Baseado nos dados do novo posicionamento, o ReDuctor envia a referência do novo NL em que se encontra o NP (etapa 3).

A Figura 24 mostra, o processo de registro em um NL. Após receber a referência do novo NL, o NP irá desregistrar-se do NL no qual se encontrava (etapa 1), e registrar-se no NL em que encontra-se atualmente (etapa 2).

Para NLs o processo de requisição e registro em NLs é o mesmo que o dos NPs. Porém, este processo deverá ocorrer apenas uma vez, pois um NL não altera o seu posicionamento.

Tabela 5: Mapeamento das URIs de Serviços do ConnDuctor

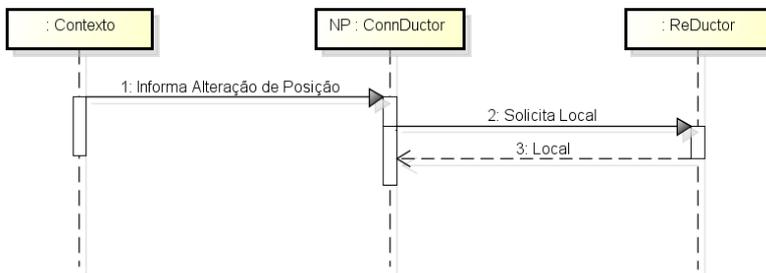
URI do Serviço	Método	Descrição
/connductor/shared	GET	Retorna a lista de recursos compartilhados pelo nodo. É possível acrescentar um identificador ao endereço para obter a representação de um recurso compartilhado específico, por exemplo, /connductor/shared/1.
/connductor/shared/search	POST	Faz a requisição de uma busca para o nodo e retorna o resultado desta requisição. É possível enviar critérios que indiquem o tipo de recurso compartilhado que está sendo pretendido.
/connductor/access-control/authorization/	POST	Utilizado para solicitar uma autorização a algum recurso do nodo.
	DELETE	Exclui uma autorização a um recurso do nodo.
/connductor/local	GET	Retorna a referência ao local onde se encontra o nodo, funcionando com um atalho para o local. Dessa forma, é possível construir endereços que apontem para recursos de outros locais, como por exemplo, /connductor/local/local/communication/message/multicast.
/connductor/guestbook	GET	Retorna uma lista de nodos conectados ao local.
	POST	Solicita o registro de um nodo no NL.
	DELETE	Solicita a retirada de um nodo no NL.
/connductor/context/	GET	Quando utilizado sem identificador, retorna a lista de contextos percebidos pelo nodo. Caso seja enviado um identificador, recupera o último registro do contexto solicitado.
	PUT	Faz o registro de dados de algum tipo de contexto percebido pelo nodo. Essa funcionalidade pode ser utilizada em caso de delegação de atividade, por exemplo, o índice glicêmico do paciente pode ser registrado remotamente pelo laboratório responsável pelo seu exame de sangue.
/connductor/context/subscription	POST	Inscribe o nodo cliente como assinante de um contexto do nodo servidor.
	DELETE	Remove a inscrição de um contexto do nodo cliente.
/connductor/context/history	GET	Retorna o histórico de um tipo de contexto. O tipo de contexto deve ser enviado junto da requisição.
/connductor/communication/message/multicast	POST	Envia uma mensagem a todos os nodos registrados no local.
/connductor/communication/message/unicast	POST	Envia uma mensagem ao nodo servidor deste recurso.

Fonte: Elaborado pelo autor.

4.4 ChronicDuctor

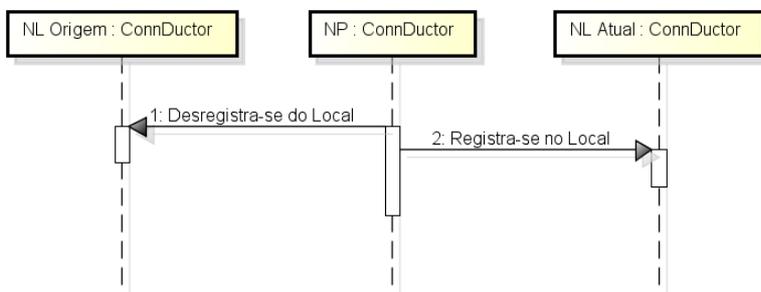
O ChronicDuctor é um assistente de saúde pessoal com foco no gerenciamento e prevenção de DCNT, executado a partir de um NP como um módulo executável do middleware ConnDuc-

Figura 23: Requisição de um novo NL pelo ConnDuctor



Fonte: Elaborado pelo autor.

Figura 24: Registro de um NP em um NL



Fonte: Elaborado pelo autor.

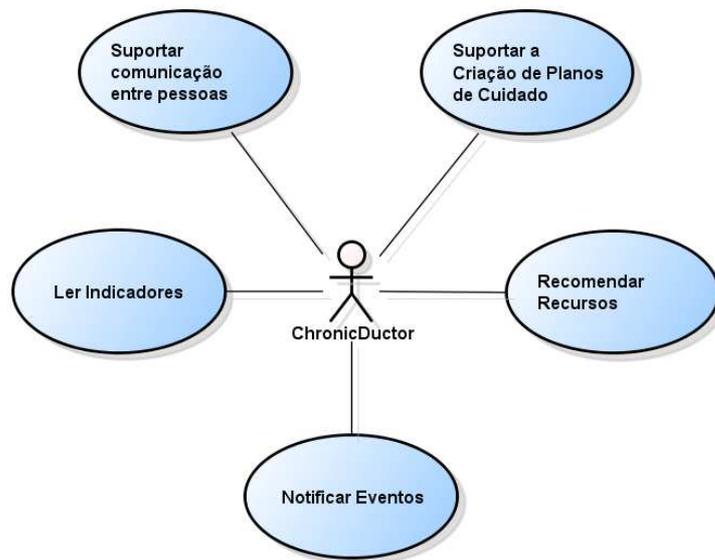
tor.

4.4.1 Requisitos do ChronicDuctor

O ChronicDuctor deve atingir cinco requisitos principais: suportar a criação de planos de cuidado, recomendar recursos de interesse, notificar eventos, ler indicadores de saúde, e suportar a comunicação entre pessoas, conforme apresentado no diagrama de caso de uso da Figura 25 e descritos a seguir:

- **Suportar a Criação de Planos de Cuidado:** deve garantir que médicos, nutricionistas e educadores corporais possam criar planos que devem ser seguidos pelos pacientes, com o objetivo de controlar e prevenir as DCNT. Os planos podem envolver recomendações de dietas, prescrições de medicamentos, plano de atividades físicas, pedidos de exames ou orientações médicas para controle de algum sinal vital que deva ser medido. Os planos podem ser finitos, no caso de um controle ser feito por tempo determinado; ou indeterminados, não há prazo para que o plano termine. Ambos os planos podem ser progressivos, caso haja a necessidade do aumento gradativo de determinada ação;
- **Recomendar Recursos:** consiste em apresentar oportunidades aos pacientes, de acordo com o contexto ao qual eles estão inseridos. Por exemplo, apresentar locais onde o pa-

Figura 25: Requisitos principais do ChronicDuctor



Fonte: Elaborado pelo autor.

ciente possa fazer uma refeição adequada, algum local em que seja possível medir algum dado corporal - como, por exemplo, a pressão arterial - apresentar pessoas com os quais o paciente possa tirar dúvidas e oferecer orientações sobre o cuidado da doença, ou até mesmo, exibir locais onde ele pode fazer a aquisição de algum medicamento;

- **Notificar Eventos:** serve como um guia para que o paciente possa seguir o controle e tratamento de sua doença, utilizando o planejamento criado pelo médicos, nutricionistas e educadores corporais. As notificações devem alertar o paciente caso ele esteja perdendo o foco do seu cuidado, e em certos casos alertar o responsável médico da falta de engajamento do paciente; lembrá-lo da necessidade de tomar ou executar certa ação, como por exemplo tomar algum medicamento, fazer a leitura de algum sinal vital, ou informar sobre algum exame ou consulta médica que deve ser realizado; e notificar que alguma meta foi atingida pelo paciente como forma de motivar a continuidade do seu engajamento no cuidado da doença;
- **Ler Indicadores:** permite a observação daqueles dados do paciente que são lidos através de instrumentos, como um glicosímetro, balanças, etc. Essa coleta poderá ser feita tanto de forma automática, caso a arquitetura do componente permita, ou de forma assistida, isto é, inserida pelo paciente manualmente por meio de uma interface gráfica;
- **Suportar a Comunicação entre Pessoas:** deve permitir a comunicação entre pacientes, médicos e cuidadores, como forma de ajudá-los na execução de suas atividades;

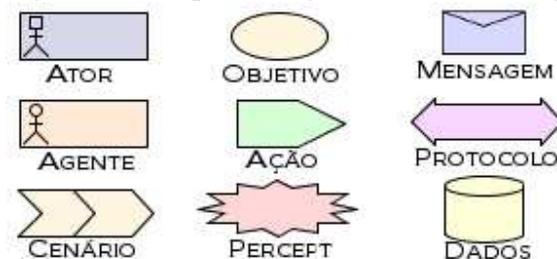
4.4.2 Modelagem do ChronicDuctor

De acordo com Padgham e Winikoff (PADGHAM; WINIKOFF, 2004), os componentes de um sistema devem ser projetados como agentes caso a resposta para a maioria das seguintes questões sejam “sim”:

- É autônomo?
- Tem objetivos?
- Visto como um objeto, é ativo?
- Faz várias atividades simultaneamente? Caso sim, necessita raciocinar entre as diferentes atividades?
- Necessita mudar a maneira como age de acordo com as mudanças do ambiente?

No caso do ChronicDuctor, a resposta é “sim” para todas as questões. Desta maneira, esse módulo foi projetado com o uso de agentes, para tanto, foi utilizada a metodologia Prometheus, já que ela foi desenvolvida tendo em vista os processos de desenvolvimento de sistemas orientados a agentes (PADGHAM; WINIKOFF, 2004; PADGHAM; THANGARAJAH; WINIKOFF, 2008; PADGHAM; THANGARAJAH; PAUL, 2012). A Figura 26 apresenta os componentes gráficos utilizados nos diagramas da metodologia Prometheus.

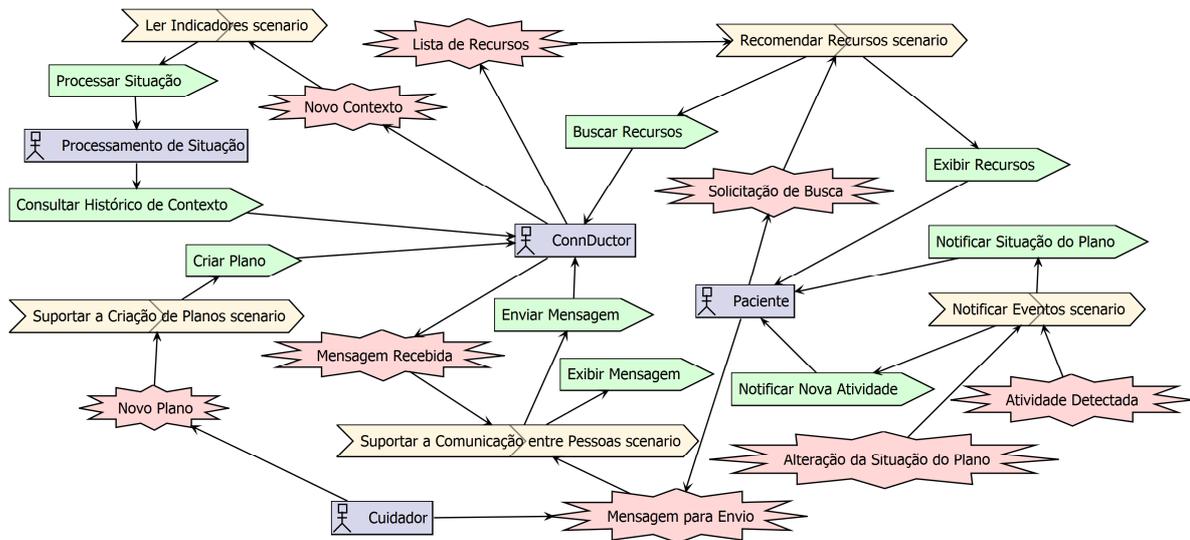
Figura 26: Legenda de componentes gráficos da metodologia Prometheus



Fonte: (PADGHAM; THANGARAJAH; WINIKOFF, 2008)

Na metodologia Prometheus, o diagrama da Visão Geral de Análise é utilizado para demonstrar o funcionamento geral do sistema. Nesse diagrama é possível especificar os cenários de aplicação e as ligações com os componentes envolvidos nesses cenários, propiciando uma visão de alto nível dos requisitos (PADGHAM; THANGARAJAH; WINIKOFF, 2008; PADGHAM; THANGARAJAH; PAUL, 2012). A Figura 27 apresenta o diagrama da Visão Geral de Análise do ChronicDuctor. Nesse diagrama os cinco requisitos principais tornaram-se cenários, cujo relacionamento com outros componentes do sistema demonstra como os requisitos especificados podem ser atingidos.

Figura 27: Diagrama da Visão Geral de Análise



Fonte: Elaborado pelo autor.

O cenário Suportar a Criação de Planos é atingido quando o ator Cuidador gera o percept Novo Plano. Este percept é recebido pelo cenário Suportar a Criação de Planos, que então aciona o ConnDuctor para que seja executada a ação Criar Plano.

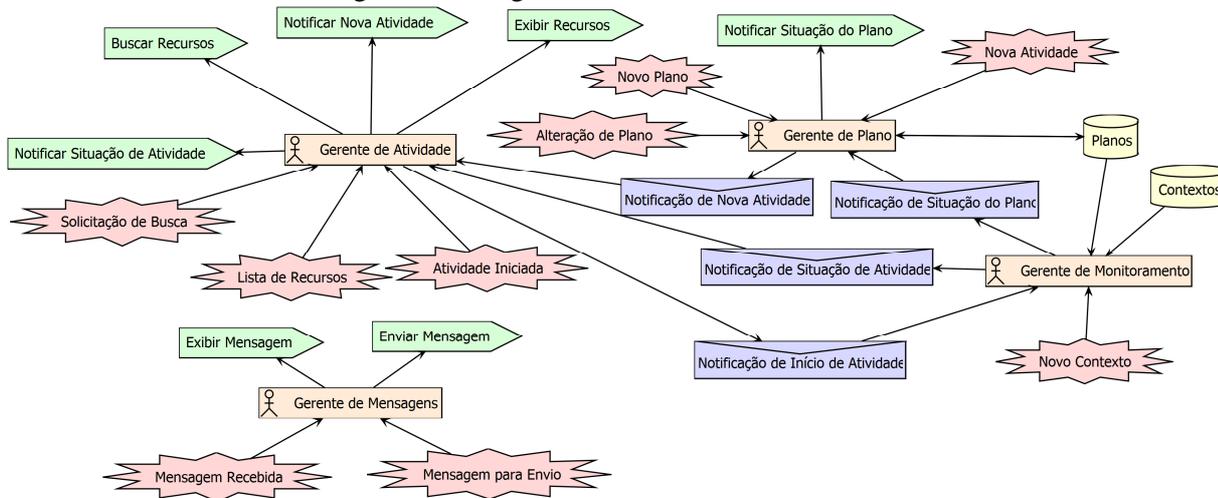
O cenário Ler Indicadores é efetuado quando o ConnDuctor gera um percept Novo Contexto, que ocasiona em uma chamada a ação Processar Situação, executada pelo ator Processamento de Situação. Este ator é responsável por identificar se os objetivos definidos para o plano do paciente estão sendo alcançados, para tanto ele necessitará consultar o histórico de contexto armazenado pelo ConnDuctor.

O cenário Notificar Eventos é responsável pela execução de duas ações: Notificar Situação do Plano e Notificar Nova Atividade. Ambas as ações são notificações enviadas para o Paciente. A primeira é executada quando o percept Alteração da Situação do Plano é recebida, enquanto que a segunda é efetuada quando é recebido o percept Atividade Detectada.

A recomendação de recursos é iniciada após o Paciente receber a notificação de uma nova atividade. Após receber esta notificação, o paciente poderá selecionar a atividade e efetuar uma Solicitação de Busca de recursos referentes a ela. A solicitação de busca gera um novo percept que é recebido pelo cenário Recomendar Recursos. O cenário aciona a ação de Busca de Recursos através do ConnDuctor, que localiza os recursos próximos ao paciente e os retorna para o cenário (percept Lista de Recursos). Após receber os recursos encontrados, o cenário aciona a ação Exibir Recursos que apresenta os recursos encontrados ao Paciente.

O cenário Suportar Comunicação entre Pessoas é encarregado pelas ações Enviar Mensagem e Exibir Mensagem. A ação Enviar Mensagem será acionada quando o Paciente ou Cuidador produzirem o percept Mensagem para Envio. Quando o cenário receber um percept Mensagem Recebida, a ação Exibir Mensagem deve ser executada para que o Paciente ou Cuidador visualizem a mensagem.

Figura 28: Diagrama da Visão Geral de Sistema



Fonte: Elaborado pelo autor.

A Figura 28 apresenta o diagrama da Visão Geral do Sistema, o qual demonstra como as ações e percepts, que relacionam cenários e atores, interagem com os agentes do sistema. Nesse diagrama os cenários foram substituídos por agentes que são encarregados por atingir os requisitos definidos para cada cenário. O Gerente de Plano é encarregado pelos requisitos do cenário Suportar a Criação de Planos e divide, com o agente Gerente de Atividade, a responsabilidade em atingir os requisitos do cenário Notificar Eventos. O Gerente de Atividade também é responsável por atingir os requisitos do cenário Recomendar Recursos, e o cenário Ler Indicadores é realizado pelo Gerente de Monitoramento. O Gerente de Mensagens é responsável por implementar o cenário Suportar a Comunicação entre Pessoas.

O cenário Suportar a Criação de Planos tem seus objetivos atingidos quando um Cuidador cria ou altera um plano utilizando a interface REST do ConnDuctor. Este notifica o Gerente de Plano através dos percepts Novo Plano ou Alteração do Plano, os quais armazenam os planos no repositório de Planos.

O objetivo do cenário Ler Indicadores é atingido quando o Gerente de Monitoramento recebe do ConnDuctor um percept identificando a chegada de um Novo Contexto. Nesse momento o Gerente de Monitoramento inicia o processamento para identificar se houve alguma alteração no quadro do paciente. O repositório de contexto é consultado, caso haja alguma mudança ele enviará uma mensagem de Notificação de Situação do Plano para o Gerente de Plano, colaborando para que ele possa notificar o paciente da situação de sua condição crônica, i.e. se houve melhora ou agravamento da condição.

O cenário Recomendar Recursos atinge seus objetivos quando o Gerente de Plano detecta que uma Nova Atividade deve ser executada. Ao fazer esta detecção, o Gerente de Plano notifica o Gerente de Atividade por meio de uma mensagem de Notificação de Nova Atividade. O Gerente de Atividade deve então notificar o paciente a respeito da nova atividade. Ao receber a notificação, o paciente poderá selecionar a atividade e efetuar uma Solicitação de Busca.

Esta solicitação fará com que o ConnDuctor localize os recursos que podem ser utilizados como suporte para a execução da atividade, conforme descrito pelo cenário Recomendar Recursos.

Para certos tipos atividades o usuário deverá informar que iniciou a sua execução. Quando isto ocorrer será gerado um percept Atividade Iniciada pelo Gerente de Atividade, e consequentemente o envio da mensagem de Notificação de Início de Atividade para o Gerente de Monitoramento. Ao receber esta mensagem o Gerente de Monitoramento iniciará o controle dos objetivos programados para atividade, por exemplo, caminhar por trinta minutos. Ao atingir os objetivos da atividade, ou caso o paciente encontre-se em uma situação de risco no momento da atividade, o Gerente de Monitoramento enviará uma mensagem de Notificação de Situação de Atividade para o Gerente de Atividade, que deve informar o paciente ao executar a ação Notificar Situação de Atividade.

4.5 Modelo Ontológico do U'Ductor

Para representar o conhecimento relacionado ao domínio do modelo proposto foi criada uma ontologia. Como forma de organizar as responsabilidades do modelo, a ontologia criada foi dividida em três subontologias: ontologia do ConnDuctor, ontologia de recursos e ontologia de planos, as duas últimas do ChronicDuctor.

Uma das vantagens do uso da ontologia, é a possibilidade de reaproveitar conhecimentos previamente descritos em outras bases ontológicas. A ontologia é um componente básico para o modelo, uma vez que ela representa o conhecimento a respeito do domínio de cuidado de DCNT, de forma que os componentes de software compartilhem o mesmo entendimento sobre os conceitos expressos nessa representação de conhecimento. A ontologia do U'Ductor foi inspirada pelos modelos de conhecimento apresentados por Koutkias et-al (KOUTKIAS et al., 2010) e Paganelli e Giuli (PAGANELLI; GIULI, 2011), tendo sido utilizada principalmente como modelo de mensagens para comunicação entre nodos, middleware e módulos executáveis.

4.5.1 Ontologia do ConnDuctor

A Figura 29 apresenta a ontologia que representa os conceitos do ConnDuctor. Entidade é a classe central dessa ontologia, de acordo com Dey (DEY; ABOWD; SALBER, 2001), as entidades mais relevantes são Locais, Pessoas e Coisas (things). Uma entidade é identificada por um URI. Um URI é um conjunto de caracteres gerados por meio de um esquema de nomeação extensível, que permite a identificação uniforme de diferentes tipos de recursos (BERNERS-LEE; FIELDING; MASINTER, 2005). Assim, todas as classes que possuem o relacionamento “identificado por URI” equivalem a classe Recurso.

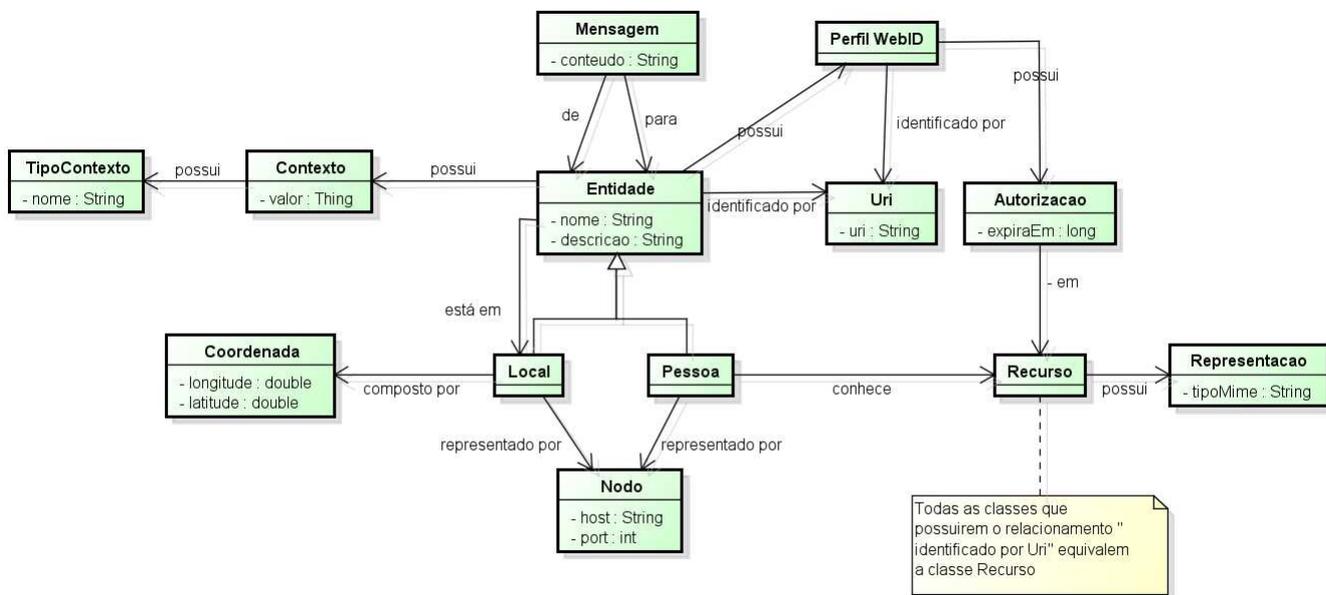
Um Recurso possuirá uma ou mais representações, identificadas através de tipos MIME que determinam o formato de apresentação desse recurso (FREED; BORENSTEIN, 1996). Este último conceito é descrito na ontologia através do relacionamento “possui Representacao”.

Um Local é utilizado para representar uma região espacial disponível em algum local geográfico, e.g. uma casa, uma rua, um parque. O relacionamento “está em” denota que uma entidade está presente em algum local. A relação “composto por Coordenada” descreve os pontos que delimitam a área do local. Os atributos latitude e longitude mantêm as coordenadas geográficas que identificam cada ponto que compõe esse local.

A classe Pessoa representa um usuário do ambiente. A relação conhece permite descrever que uma pessoa conhece algum Recurso específico, por exemplo, um lugar, uma pessoa, um livro.

A classe Nodo descreve um dispositivo que executa o middleware ConnDuctor, mantendo em seus atributos as informações necessárias para comunicação com ele. O Nodo pode representar uma Pessoa ou um Local.

Figura 29: Modelo Ontológico ConnDuctor



Fonte: Elaborado pelo autor.

A classe Perfil WebId descreve a relação entre a entidade e o documento utilizado para certificar a autenticidade de sua identidade (INKSTER et al., 2011), e descrever as ligações sociais existentes com outras entidades, de modo que seja possível inferir as autorizações a recursos através de confiança (GAMBLE; GOBLE, 2010). Autorizações a recursos que não puderam ser inferidas automaticamente serão representadas por meio da classe Autorização, que possui um prazo para expiração, descrito pelo atributo “expiraEm”. Essa organização de autorização possibilita a anonimização da identidade de uma pessoa já que a classe Perfil WebId é um recurso, e esse requer autorização para que seja acessado.

Tipo Contexto e Contexto são utilizados para representar genericamente as informações que caracterizam a situação de uma entidade (DEY; ABOWD; SALBER, 2001). Um Contexto deve possuir um valor que o representa. Uma vez que esse valor pode possuir muitos

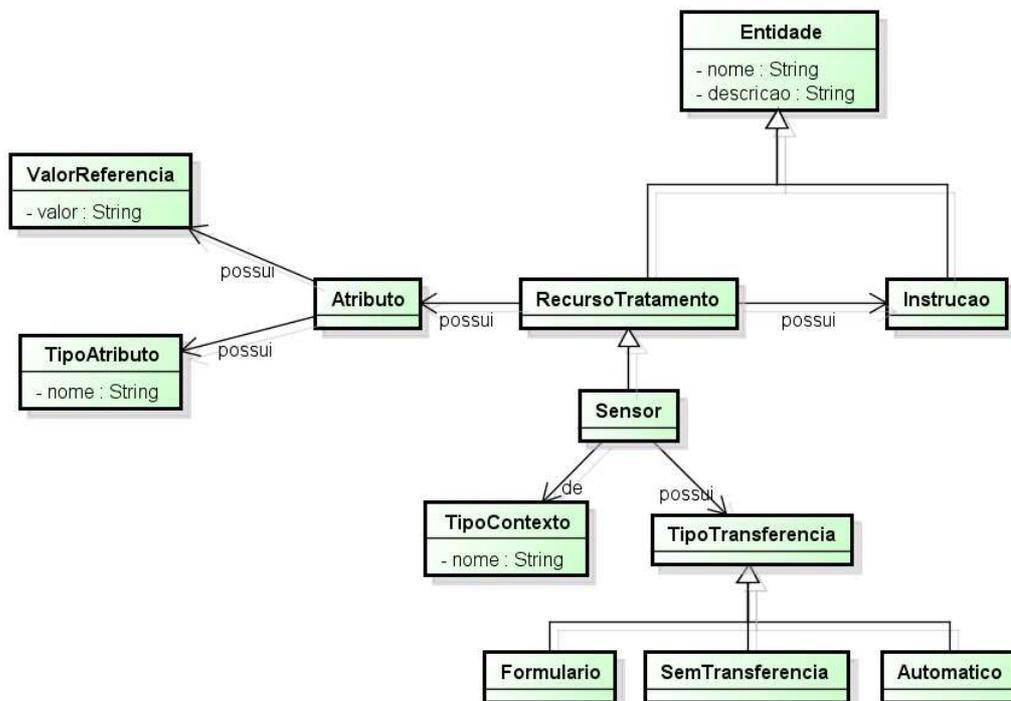
formatos ele é descrito como sendo do tipo Thing, que é a classe raiz do modelo ontológico. Dessa maneira, todas as classes definidas no modelo também são subclasses de Thing. A classe TipoContexto representa a classificação de determinada informação de contexto, e.g., local, clima, quantidade de tráfego de dados.

A classe Mensagem é utilizada para caracterizar a comunicação entre duas ou mais entidades do modelo e possui os relacionamentos de, que representa a entidade de origem da comunicação, para, que representa os destinatários da comunicação, e o atributo conteudo, representando a mensagem propriamente dita, que foi enviada pelo remetente aos destinatários.

4.5.2 Ontologia de Recursos do ChronicDuctor

A Ontologia de recursos do ChronicDuctor é apresentada na Figura 30. Esta ontologia envolve os conceitos por trás dos recursos utilizados para o cuidado de DCNT. A classe RecursoTratamento deve ser utilizada para representar qualquer recurso que possa ser utilizado para o cuidado de DCNT. Um recurso de tratamento pode possuir instruções de como ele pode ser utilizado, esse conceito é representado pela relação possui Instrucao. A bula de um medicamento, um vídeo de operação de um monitor de pressão arterial ou qualquer outro material que possa auxiliar o paciente, ou responsável, nas atividades de cuidado pode ser considerado como uma instrução.

Figura 30: Modelo Ontológico de Recursos - ChronicDuctor



Fonte: Elaborado pelo autor.

Um RecursoTratamento possuirá atributos que descrevem suas características. Este con-

ceito é descrito pelo relacionamento `RecursoTratamento` possui `Atributo`. Cada `Atributo` é descrito por um par de `TipoAtributo` e `ValorReferencia`. Dessa forma é possível descrever diferentes tipos de recursos de tratamento como, por exemplo, medicamentos e alimentos.

`Sensor` é um tipo especial de recurso de tratamento utilizado para efetuar medições de algum tipo de indicador, necessário para acompanhamento da situação de saúde do paciente. Um glicosímetro, uma balança ou um monitor de pressão arterial são exemplos de sensores. O relacionamento “de `TipoContexto`” descreve o tipo de contexto que aquele sensor irá gerar, e.g. índice glicêmico, peso, pressão arterial sistólica e diastólica. O relacionamento “possui `TipoTransferencia`” descreve de que forma os dados são transmitidos pelo sensor. A ontologia de recursos do `ChronicDuctor` prevê três tipos de entrada: `Formulario`, a transferência de dados entre o sensor e o smartphone do paciente é feita por meio de um formulário eletrônico; `SemTransferencia`, o sensor não provê um meio para transferência da leitura efetuada; e `Automatica`, a transferência é feita sem que o usuário necessite digitar os dados lidos, embora possa haver a necessidade de pareamento com o sensor.

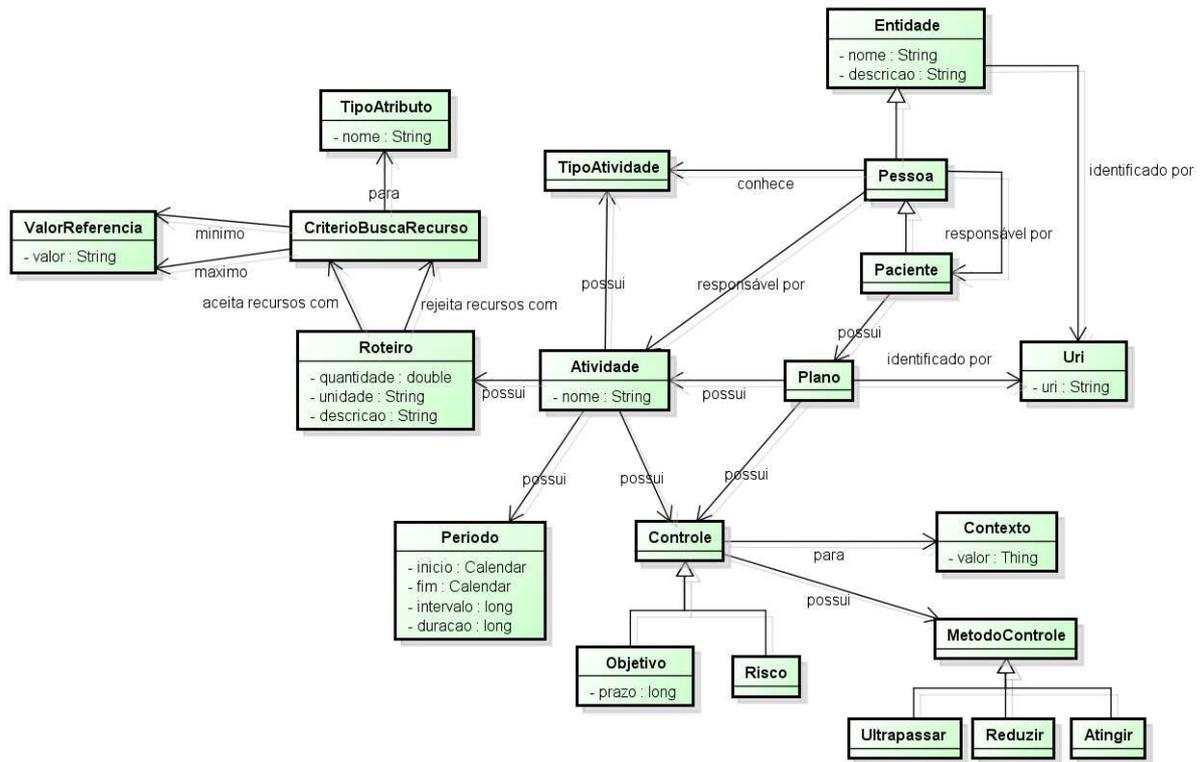
4.5.3 Ontologia de Plano do `ChronicDuctor`

A ontologia de plano do `ChronicDuctor` descreve o plano criado por médicos e outros profissionais de saúde com o objetivo de auxiliar o paciente a manter a concentração no cuidado de sua doença, ajudando a alcançar uma melhora no seu estado de saúde através do autogerenciamento e colaboração com membros da comunidade que o paciente está inserido (WAGNER et al., 2001; BODENHEIMER; WAGNER; GRUMBACH, 2002; OMS, 2003; ICIC, 2011).

Nessa ontologia a classe `Pessoa` adquire três novos relacionamentos, “responsável por `Paciente`”, “responsável por `Atividade`” e “conhece `TipoAtividade`”. O relacionamento “responsável por `Paciente`” indica que a pessoa é responsável, em algum nível, pelo andamento do cuidado do paciente, sendo autorizado a receber notificações alterações da situação de saúde do paciente. Médicos, enfermeiros e familiares são exemplos de responsáveis pelo paciente. O relacionamento “responsável por `Atividade`” descreve as pessoas que são responsáveis por alguma atividade específica do paciente, por exemplo, um nutricionista é responsável pelas atividades alimentares do paciente, enquanto que um educador corporal é responsável pelas atividades físicas executadas pelo paciente. Os responsáveis por atividades são autorizados a acompanhar o andamento dos indicadores relacionados a atividade. O relacionamento “conhece `TipoAtividade`” descreve que uma pessoa conhece, ou possui experiência, e algum um tipo de atividade de cuidado, assim é possível localizar pessoas que possam auxiliar o paciente em suas atividades de cuidado.

A classe `Paciente` representa uma pessoa que possui algum problema médico tratado através de um `Plano` de tratamento (“possui `Plano`”). A classe `Plano` define o plano de tratamento criado em parceria com o responsável médico do paciente, utilizado para auxiliar o paciente no cuidado de sua saúde. Um plano é composto por atividades (“possui `Atividade`”) e controles

Figura 31: Modelo Ontológico de Planos - ChronicDuctor



Fonte: Elaborado pelo autor.

(“possui Controle”).

A classe Controle faz parte da descrição do plano de cuidado do paciente, servindo para auxiliar o paciente a manter a concentração no tratamento de sua doença. O relacionamento “para Contexto” indica um parâmetro utilizado como referência para controlar algum tipo de indicador corporal, enquanto que o relacionamento “possui MetodoControle” indica a forma como é feito o controle. Existem três métodos de controle: Ultrapassar, que indica que o novo contexto gerado pelo paciente ultrapassou o valor do parâmetro de controle; Reduzir que indica que o novo contexto gerado pelo paciente é menor que o valor do parâmetro de controle; e Atingir, que indica que o contexto gerado pelo paciente é igual ao valor do parâmetro de controle. Essa forma de controle permite avaliar se o paciente alcançou os objetivos planejados ou, ao contrário, se atingiu uma situação de risco.

A Atividade é uma tarefa que deve ser executada pelo paciente em um momento determinado, por exemplo, ingerir um medicamento, fazer a medição de algum indicador corporal ou efetuar uma atividade física. A relação “possui Período” define o momento de início e fim dessa atividade, o intervalo que essa atividade deverá se repetir, e a sua duração. Por exemplo, o paciente deverá efetuar nebulização todo dia durante seis meses por dez minutos, nesse caso, o atributo início indica a data e hora em que o paciente deve iniciar a nebulização, o atributo fim define o momento em que a nebulização será interrompida, o atributo intervalo descreve de quanto em quanto tempo essa atividade se repetirá (no caso do exemplo, uma vez por dia) e

a duração define por quanto tempo essa atividade será efetuada. A relação Atividade “possui Controle” possibilita a descrição de parâmetros que auxiliem no monitoramento da atividade que está sendo executada, por exemplo, a frequência cardíaca durante uma atividade aeróbica não pode exceder o resultado da seguinte equação, $220 - \{\text{idade do paciente}\}$ (AVERBUJ; TIJMAN, 2011).

A relação Atividade “possui Roteiro” fornece instruções daquilo que o paciente deve executar na atividade. Ela indica a quantidade de vezes que um RecursoTratamento será utilizado dentro da atividade, possibilitando que sejam descritas dietas, planos de atividade física, lembretes de uso de medicamento ou medições de algum tipo de indicador corporal. Nesse caso, a classe CriterioBuscaRecurso é utilizada para descrever as características dos recursos que podem ser utilizados pelo roteiro, através da relação aceita recursos com CriterioBuscaRecurso, ou recursos que devem ser ignorados, através da relação rejeita recursos com CriterioBuscaRecurso. A relação CriterioBuscaRecurso para TipoAtributo indica uma característica que o recurso deve possuir, enquanto que as relações CriterioBuscaRecurso minimo ValorReferencia e CriterioBuscaRecurso maximo ValorReferencia, descrevem os valores válidos que as características devem possuir para que os recursos sejam aceitos.

4.6 Considerações sobre o capítulo

O presente capítulo apresentou o U’Ductor, um modelo para cuidado ubíquo de DCNT. No U’Ductor NPs e NLs compartilham recursos e informações de contexto com o intuito de cooperar nas atividades de cuidado de DCNT. O middleware ConnDuctor oferece a infraestrutura para que os nodos possam compartilhar e localizar recursos e contextos necessários ao cumprimento de alguma atividade de cuidado. O ReDuctor é responsável por identificar a localização dos nodos e converter essa localização em uma referência de NL. O ChronicDuctor é um módulo de auxílio ao autogerenciamento de DCNT, executado pelo middleware ConnDuctor nos NPs. As ontologias criadas mapeiam os conceitos relacionados ao modelo, como compartilhamento e localização de recursos, e cuidado de DCNT.

5 AVALIAÇÃO DO MODELO

A avaliação do U'Ductor foi dividida em duas etapas. A primeira consistiu da implementação do protótipo do modelo. Na segunda etapa foi utilizado o protótipo para execução de três tipos de avaliação. O primeiro tipo consistiu de um teste de aceitação de tecnologia que teve a participação de pacientes portadores de doenças crônicas. O segundo tipo consistiu no uso estendido do protótipo que teve a duração de um mês e meio. Na terceira avaliação foi feito um estudo de consumo de bateria e CPU pelo middleware do NP.

O projeto necessitou de apreciação do Comitê de Ética em Pesquisa da Unisinos, uma vez a que pesquisa envolvia seres humanos e sua área temática era orientada ao desenvolvimento de insumos para saúde. O Comitê de Ética em Pesquisa é um colegiado com objetivo de garantir o cumprimento dos princípios éticos em pesquisas que envolvem seres humanos. A pesquisa foi cadastrada na Plataforma Brasil¹ com o título “ChronicDuctor: Assistente para cuidado ubíquo de doenças crônicas não transmissíveis”, onde foram anexados o texto descritivo do projeto, termo de consentimento livre e esclarecido, e o documento de autorização do coordenador do Programa de Pós-Graduação em Computação Aplicada. Finalizado o cadastro da pesquisa, a mesma foi submetida para apreciação do comitê de ética, tendo sido aprovada no dia 19 de novembro de 2012, com o N° 12/109.

5.1 Aspectos de Implementação

A construção do protótipo envolveu a implementação dos seguintes itens: middleware ConnDuctor, ontologia, ChronicDuctor e ReDuctor.

O middleware para NPs foi desenvolvido para plataforma móvel Android², enquanto que o middleware para NLs foi desenvolvido utilizando a plataforma J2SE³. Esta distinção deve-se a restrições de arquitetura, pois a plataforma Android possui uma implementação específica do ambiente de execução Java (JRE) (DEVELOPERS, 2012), o que faz com que algumas bibliotecas utilizadas tenham versões diferentes para cada plataforma. Apenas foram desenvolvidos os componentes do middleware que eram necessários para avaliar a funcionalidade e usabilidade do modelo, com relação aos requisitos especificados para o ChronicDuctor. Desta forma, os componentes de Controle de Acesso e Proxy foram excluídos da implementação.

Para o desenvolvimento da interface REST na plataforma J2SE foi utilizado o pacote Jersey⁴, que é uma implementação da especificação JAX-RS⁵ para Web Services REST na linguagem Java. Na versão Android a interface REST foi implementada com o uso da API Rest-

¹<http://aplicacao.saude.gov.br/plataformabrasil/>

²<http://www.android.com/>

³<http://www.oracle.com/technetwork/java/javase/index.html>

⁴<http://jersey.java.net/>

⁵<http://jax-rs-spec.java.net/>

let⁶, o que permitiu a criação de um servidor HTTP nessa plataforma.

Os recursos compartilhados foram descritos através da linguagem TURTLE⁷ e armazenados em um repositório RDF para permitir a busca semântica de recursos. A linguagem SPARQL⁸ foi utilizada como linguagem de consulta ao repositório RDF. As apis Jena-ARQ⁹, na plataforma J2SE, e Androjena-ARQoid¹⁰, para a plataforma Android, foram utilizadas para armazenar e consultar dados no repositório RDF. Devido ao alto tempo de compilação do ARQoid, foi necessário que o componente de recursos compartilhados da plataforma Android fosse colocado em um processo separado, comunicando-se com o middleware por meio de uma interface AIDL¹¹.

Os componentes de contexto e mensageiro foram desenvolvidos utilizando praticamente a mesma programação nos NPs e NLs. A única diferença entre as implementações dos NLs e NPs se deu na implementação dos NPs, onde o componente de contexto recebeu dados de localização oriundos do GPS dos smartphones utilizados nos testes. O componente de Nodos retorna a lista de nodos vinculados à instância do middleware no formato JSON¹². Da mesma forma, a referência ao local no qual o nodo está vinculado também é apresentada no formato JSON. Isto foi possível porque as classes da ontologia foram portadas para classes Java. Mesmo assim, a ontologia permaneceu sendo utilizada pelo repositório RDF.

O ChronicDuctor foi desenvolvido para a plataforma Android, tendo sido implementado como um serviço. A comunicação com o middleware é feita através de interface AIDL dessa plataforma. Cada agente foi implementado como uma classe, e os percepts e mensagens descritos na metologia Prometheus foram transformados em métodos destas classes. O processo do ChronicDuctor é responsável por acionar os métodos que representam percepts. Os métodos que representam mensagens são invocados pelos próprios agentes.

O ReDuctor foi desenvolvido na plataforma J2SE. Este serviço utilizou um servidor HTTP para receber requisições contendo coordenadas de localização. O servidor desse serviço foi implementado utilizando o pacote Jersey. O serviço respondia a cada requisição de localização com uma representação do nodo no formato JSON. Como as avaliações trataram de localizações fixas, a funcionalidade de registro de locais não foi efetuada. Um arquivo XML foi utilizado para armazenar a hierarquia de locais e localização de cada local.

5.2 Teste de aceitação do ChronicDuctor

Um teste de aceitação de tecnologia foi efetuado para avaliar o ChronicDuctor. Neste teste um experimento foi executado em um cenário construído para demonstrar as características do

⁶<http://www.restlet.org>

⁷<http://www.w3.org/TeamSubmission/turtle/>

⁸<http://www.w3.org/TR/rdf-sparql-query/>

⁹<http://jena.apache.org/documentation/query/index.html>

¹⁰<http://code.google.com/p/androjena/>

¹¹<http://developer.android.com/guide/components/aidl.html>

¹²<http://www.json.org/>

ChronicDuctor para os sujeitos participantes. Para estar apto a participar da pesquisa o sujeito deveria possuir uma das seguintes condições crônicas, ou fatores de risco: hipertensão, diabetes tipo 2, colesterol alto, obesidade, ou sobrepeso. Foi perguntado aos indivíduos encontrados que possuíam o perfil desejado, se eles tinham interesse em participar do experimento. Uma data foi agendada para apresentar e avaliar o ChronicDuctor por aqueles indivíduos que demonstraram interesse. Dez sujeitos participaram da pesquisa, a Tabela 6 lista as condições crônicas dos indivíduos que realizaram a pesquisa.

Tabela 6: Condição Crônica x Sujeito

Sujeito	Colesterol Alto	Diabetes	Hipertensão	Obesidade	Sobrepeso
Sujeito #1			X	X	
Sujeito #2				X	
Sujeito #3	X		X		X
Sujeito #4				X	
Sujeito #5	X				
Sujeito #6			X		X
Sujeito #7					X
Sujeito #8				X	
Sujeito #9			X		
Sujeito #10			X		X

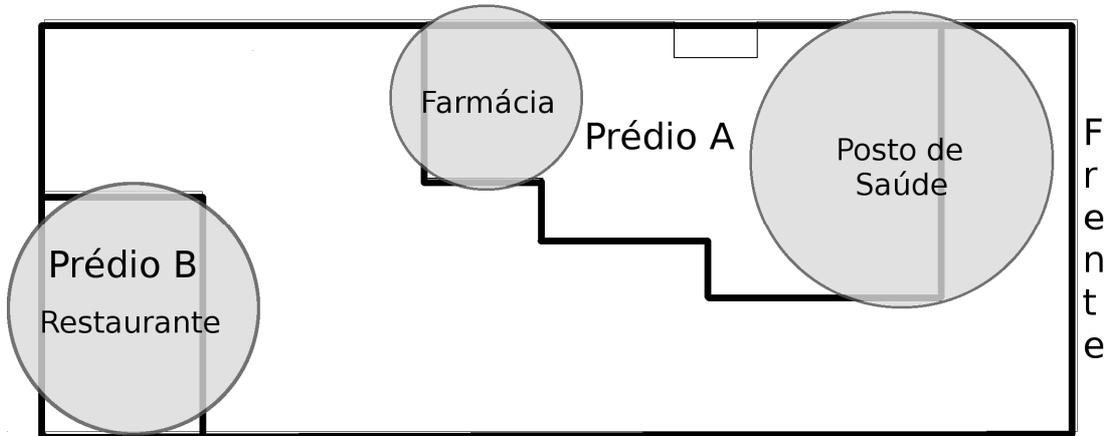
O experimento foi executado em uma área localizada no município de Nova Santa Rita, Rio Grande do Sul, Brasil. Um cenário que reproduzia um bairro hipotético foi montado no local. Os dois prédios existentes na área foram utilizados para distribuir os lugares pertencentes ao bairro hipotético. Cada círculo na Figura 32 representa um local pertencente a este bairro, sendo que cada um destes locais possuía um NL que o representava virtualmente. Os recursos de cuidado foram distribuídos nos NLs de acordo com a função do recurso, por exemplo, medicamentos ficavam na farmácia e o restaurante oferecia refeições. A Tabela 7 descreve os tipos de recursos que eram encontrados em cada local.

Tabela 7: Tipos de recursos encontrados no ambiente de simulação

Local	Tipo de Recurso
Restaurante	Alimentos com baixo teor de gordura, sal ou calorias
Farmácia	Balança, medicamentos para hipertensão, colesterol alto, e insulina
Posto de Saúde	Monitor de pressão sanguínea, glicosímetro

Cada sujeito efetuou o experimento individualmente. Para cada sujeito foi criado um plano de cuidado simples com uso do ChronicDuctor, para que fosse possível demonstrar o aplicativo no cuidado de DCNT. Na construção do plano de cuidado foi possível definir atividades nas quais o sujeito deveria cumprir em um determinado tempo, recursos que a atividade necessitava, e objetivos que o sujeito deveria alcançar (Figura 33). Por exemplo em um caso de cuidado

Figura 32: Planta baixa do cenário de simulação



de hipertensão, uma das atividades seria “Aferir pressão arterial”, esta atividade deveria ser configurada para buscar por monitores de pressão arterial, e o objetivo definido para o usuário seria alcançar uma medida de pressão arterial sistólica de 119 mmHg¹³ e uma pressão diastólica de 79 mmHg.

Figura 33: Formulário de construção de plano de cuidado

Plano

Atividades

Nome: Tipo:

Início: (dd/mm/aaaa hh:mm) Repetir a cada: Por:

Responsável:

Critério para Busca de Recursos

Tipo Recurso:

Controles

Indicador: Objetivo: Valor:

Depois da criação do plano de cuidado, cada sujeito caminhou no cenário por aproximadamente 15 minutos utilizando o aplicativo em um smartphone, um Samsung GT-S5570B contendo o sistema operacional Android 2.2. O aplicativo foi utilizado para que o sujeito recebesse notificações de atividade (Figura 34(a)); entrar com dados de sinais vitais (Figura 34(b)); buscar

¹³mmHg: Milímetro de mercúrio

recursos de cuidado (Figura 34(c)); localizar pessoas que pudessem dar suporte nas atividades definidas no plano de cuidado (Figura 34(d)); conversar com pessoas (Figura 34(e)); receber alertas de objetivos alcançados (Figura 34(f)); e receber alertas sobre falta de entrada de sinais vitais (Figura 34(g)). O autor da dissertação utilizou o aplicativo instalado em outro smart-phone, um aparelho Motorola Spice XT300 com Android 2.1, para que fosse possível simular as funcionalidades de localização de pessoas e conversação.

Depois do uso do aplicativo os sujeitos foram convidados a responder um questionário de avaliação. O questionário foi preparado seguindo o modelo de aceitação de tecnologia, ou technology acceptance model (TAM), proposto por Fred D. Davis (DAVIS, 1989) e expandido por Yoon e Kim para viabilizar a sua aplicação em ambientes de computação ubíqua (YOON; KIM, 2007). No modelo TAM a satisfação dos usuários é medida através da utilidade percebida e da facilidade de uso percebida. A utilidade percebida é utilizada para determinar se a tecnologia proposta consegue ajudar seus usuários em realizar um trabalho melhor, enquanto que a facilidade de uso avalia se a tecnologia tem condições de ser utilizada com um mínimo de esforço. Assim, o questionário de avaliação foi dividido em duas categorias: afirmações de utilidade percebida e afirmações de facilidade de uso. O questionário foi composto por 17 afirmações de itens Likert de cinco níveis (LIKERT, 1932), que foram divididos em discordo totalmente, discordo parcialmente, indiferente, concordo parcialmente e concordo totalmente .

As afirmações do questionário estão listadas na Tabela 8. A Figura 35 exhibe o resultado geral para cada tipo de satisfação do usuário. Os resultados obtidos demonstram uma boa aceitação dos sujeitos de pesquisa em relação ao foco dado ao autogerenciamento e a integração entre recursos oriundos da comunidade e das organizações de saúde.

A aceitação com relação às funcionalidades de autogerenciamento são expressas pelas afirmações 1, 2, 3, 4 e 10 do questionário de utilidade percebida. Para essas afirmações 96% dos sujeitos declararam “concordar totalmente”. A aceitação das funcionalidades referentes a integração entre pacientes e recursos da comunidade, e das organizações de saúde, é expressa através das afirmações 5 e 6 do questionário de utilidade percebida. Para essas afirmações 90% dos sujeitos declararam “concordar totalmente”. A aceitação das funcionalidades de integração entre pacientes e membros da comunidade, e das organizações de saúde, é expressa através das questões 8 e 9 do questionário de utilidade percebida. Para essas questões 100% dos sujeitos declararam “concordar totalmente”. Uma visão geral da aceitação de utilidade da aplicação é expressa pela afirmação 7 do questionário de utilidade percebida, onde 90% dos sujeitos declararam “concordar totalmente” com este item.

Além da avaliação feita através do questionário, os sujeitos #6, #8 e #10 documentaram textualmente suas impressões com relação ao aplicativo. Os sujeitos #6 e #8 abordaram sugestões para melhoria da usabilidade do aplicativo. Já o sujeito #10 comentou as suas impressões quanto ao benefício do uso do aplicativo em seu dia-a-dia. Os comentários efetuados pelos sujeitos encontram-se transcritos abaixo.

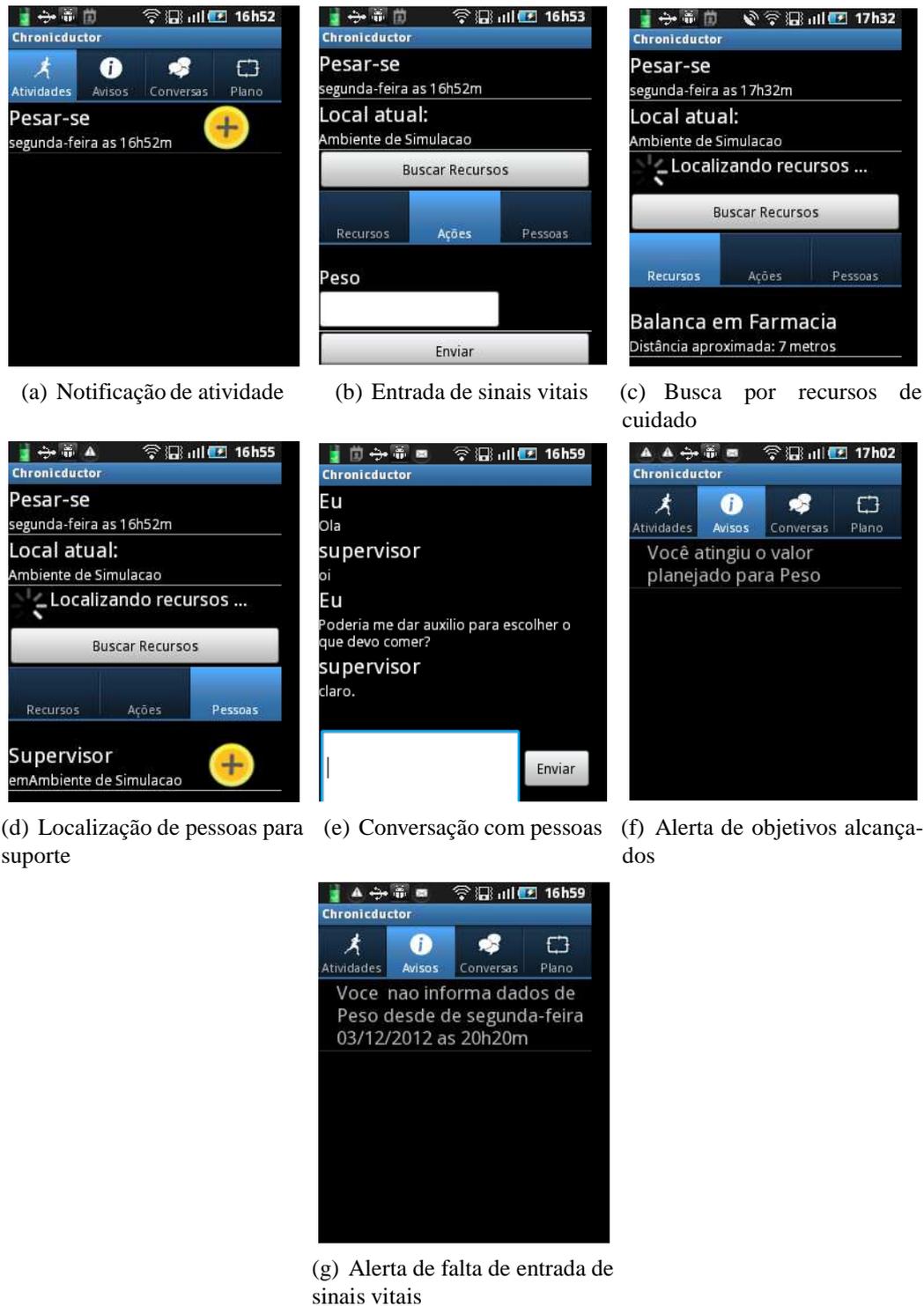


Figura 34: Telas do aplicativo ChronicDuctor

- “1. Números de ordem e total das conversas realizadas;
2. Possibilidade de criar pastas para arquivar conversas importantes;
3. Localizar recursos através de busca pela internet;
4. Importar da internet informações de recursos disponíveis na região onde localiza-se o usuário, também uma forma de cadastrar recursos locais;

Tabela 8: Questionário de avaliação

Tipo de aceitação	Afirmação
Utilidade percebida	1. É possível descrever as atividades de cuidado com um nível suficiente de detalhamento
	2. A notificação de atividades é útil à rotina de cuidado
	3. Os alertas de objetivos atingidos são úteis à rotina de cuidado
	4. Os alertas de progresso de objetivos são úteis à rotina de cuidado
	5. A possibilidade de localizar recursos próximos de acordo com a atividade é útil à rotina de cuidados
	6. O aplicativo oferece recursos necessários às atividades de cuidado
	7. O aplicativo auxilia nas atividades de cuidado
	8. A possibilidade de localizar pessoas próximas de acordo com o conhecimento que ela possui em relação à atividade é útil à rotina de cuidados
	9. A possibilidade de conversar com pessoas próximas é útil à rotina de cuidados
	10. A possibilidade de inserir dados de indicadores corporais (por exemplo, peso, pressão arterial, glicose), é útil à rotina de cuidados
Facilidade de uso percebida	1. O formulário para preenchimento do plano de cuidado é de fácil entendimento
	2. A notificação de atividades é de fácil entendimento
	3. Os alertas de objetivos atingidos são de fácil compreensão
	4. Os alertas de progresso de objetivos são de fácil compreensão
	5. É fácil localizar recursos e pessoas utilizando o aplicativo
	6. É fácil conversar com outras pessoas utilizando o aplicativo
	7. É fácil inserir dados de indicadores corporais (por exemplo, peso, pressão arterial, glicose)

5. Exibir imagem do contato de apoio no momento do chat.” (Sujeito #6)

- “1. Cadastro de contatos para facilitar o acesso aos profissionais de apoio no momento da execução da(s) atividade(s) que foi(ram) definida(s) inicialmente no plano;
2. Em caso de emergência, emissão de alerta de acordo com o contexto, enviando dados a unidade de pronto atendimento mais próxima;
3. Alertas de voz motivacionais: - antes, no início, durante, no final e depois da execução da atividade definida. Por exemplo: ao executar a atividade definida, o usuário receberá mensagens incentivadoras, informando o quanto falta para atingir o seu objetivo;
4. Personalização dos avisos de objetivos, deixando-os mais amigáveis.” (Sujeito #8)

“Achei o aplicativo de fácil uso, muito útil no dia-a-dia, ajudando a encontrar locais rapidamente. O lembrete de que não medimos a pressão, por exemplo, é muito inte-

ressante porque na correria do dia-a-dia acabamos esquecendo, ou não temos tempo. Mas com ele dá sempre para lembrar e manter o seu objetivo. Com certeza o aplicativo será de muito bom uso, no meu caso.” (Sujeito #10)

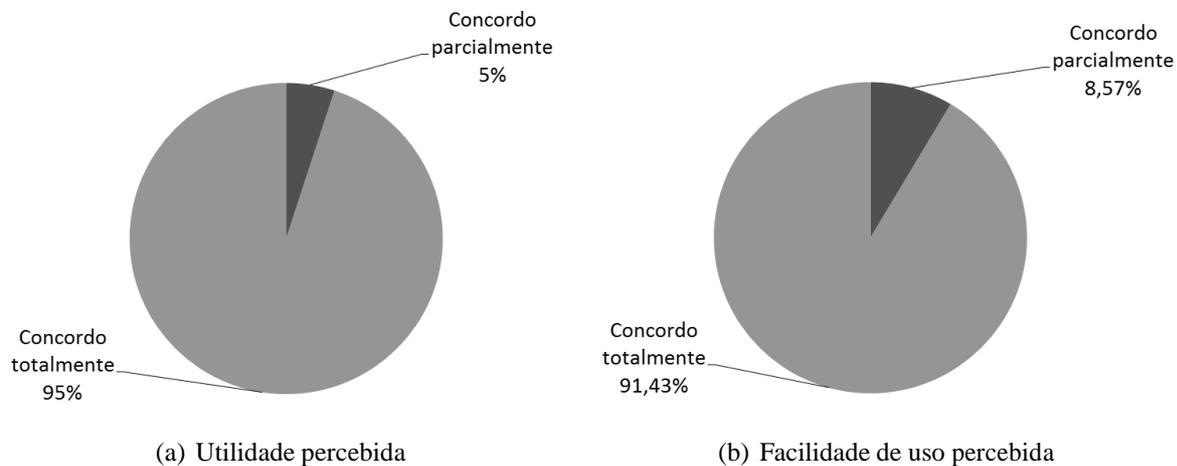


Figura 35: Resultado geral de utilidade percebida e facilidade de uso percebida

5.3 Teste de uso estendido do ChronicDuctor

A segunda avaliação efetuada consistiu de um teste de uso do ChronicDuctor efetuado por um paciente portador de hipertensão e sobrepeso. Este teste foi efetuado por 1 mês e meio, tendo abrangido o período de 15/12/2012 à 31/01/2013, com objetivo de coletar dados de contexto do usuário, calcular o tamanho necessário para o armazenamento dos dados coletados, e para identificar questões de uso relacionadas às funcionalidades de busca de recurso, entrada de dados de sinal vital, visualização de atividade e visualização de progresso de objetivo.

O ChronicDuctor foi utilizado para auxiliar o usuário em suas atividades de cuidado por meio de alertas de atividades que deveriam ser executadas, inserção de dados de controle, e localização de recursos de cuidado disponíveis próximo de onde o usuário se localizava. O ChronicDuctor foi instalado no smartphone do usuário, um Samsung modelo GT-I81160L, com sistema operacional Android versão 2.3.

As orientações definidas pelo cardiologista e endocrinologista do usuário foram transformadas em um plano de cuidado no modelo do U'Ductor, que continha uma lista de atividades e uma lista de controles. A Tabela 9 apresenta a lista de atividades configuradas no plano, a frequência de ocorrência destas atividades, o tipo de atividade, e os tipos de recursos que deveriam ser localizados pelo ChronicDuctor para auxiliar o usuário no cumprimento da atividade. Na Tabela 10 são listados os tipos de contextos que foram controlados pelo ChronicDuctor, o método de controle e o percentual alcançado pelo usuário ao final dos testes.

Os recursos necessários para o cumprimento das atividades do usuário foram distribuídos

Tabela 9: Plano de cuidado do Usuário

Atividade	Frequência	Tipo	Recursos Utilizados
Café da Manhã	Diária	Refeição	Carboidratos; frutas
Colação	Diária	Refeição	Frutas
Almoço	Diária	Refeição	Vegetais folhosos; legumes; carnes brancas; carboidratos
Lanche da Tarde	Diária	Refeição	Frutas
Jantar	Diária	Refeição	Vegetais folhosos; legumes; carnes brancas; carboidratos
Controle de Peso	Diária	Entrada de sinal vital	Balança
Aferir pressão	Diária	Entrada de sinal vital	Monitor de pressão arterial
Caminhada	Diária	Atividade Física	-

Tabela 10: Objetivos de cuidado do Usuário

Tipo de Contexto	Método de Controle	% Cumprido ao Final dos Testes
Peso	Alcançar	-26,7%
Pressão diastólica	Alcançar	100%
Pressão sistólica	Alcançar	73,34%

através dos NLs. Estes NLs foram distribuídos em duas áreas de maior presença do usuário: área dos arredores da residência, localizada no centro do município de Nova Santa Rita (Figura 36(a)); e área dos arredores do local de trabalho, localizada no bairro Mathias Velho do município de Canoas (Figura 36(b)). A Tabela 11 apresenta os locais existentes nessas áreas e os recursos oferecidos por cada local.

Para que houvesse disponibilidade das funcionalidades de localização e busca de recursos do modelo, os NLs e o serviço ReDuctor foram hospedados no OpenShift¹⁴, que é uma plataforma de computação na nuvem oferecida como serviço. Todas as instâncias de NLs foram hospedadas em <http://connductor-uductor.rhcloud.com>. O acesso a cada instância se dava através do acréscimo de um identificador do NL ao final do endereço do servidor, por exemplo, o endereço <http://connductor-uductor.rhcloud.com/posto-saude-mv/connductor> representava o posto de saúde dos arredores do local de trabalho, e o endereço <http://connductor-uductor.rhcloud.com/farmacia-1-ns/connductor> representava a farmácia #1 dos arredores da residência. O ReDuctor ficou hospedado no endereço <http://reductor-uductor.rhcloud.com>.

Ao término dos testes o usuário preencheu um formulário de opinião, que foi utilizado para identificar as dificuldades de uso e aspectos positivos e negativos do aplicativo ChronicDuctor. O usuário avaliou as funcionalidades do ChronicDuctor em relação a sua frequência de utilização, dificuldades de uso encontradas, pontos positivos, e sugestões de melhoria. Para a avaliação da frequência de utilização foi utilizada uma escala contendo as seguintes opções: Nunca utilizada; baixa (utilizada 1 a 4 vezes); média (utilizada 5 a 8 vezes); Alta (utilizada mais de oito vezes). As opiniões do usuário são descritas a seguir.

¹⁴<https://openshift.redhat.com>



(a) Arredores da Residência



(b) Arredores do Local de Trabalho

Figura 36: Localização dos Recursos

Tabela 11: Mapeamento dos Recursos Oferecidos pelos Locais

Região	Local	Recursos
Arredores da Residência	1. Residência	Balança; monitor de pressão sanguínea
	2. Mercado	Frutas; carboidratos; bebidas
	3. Farmácia#1	Balança; medicamentos para controle de pressão arterial e colesterol
	4. Farmácia#2	Balança
	5. Posto de Saúde	Monitor de pressão sanguínea
	6. Farmácia#3	Balança; medicamentos para controle de pressão arterial e colesterol
	7. Restaurante#1	Bebidas; carboidratos; carnes brancas; carnes vermelhas; vegetais folhosos
	8. Restaurante#2	Bebidas; carboidratos; carnes brancas; carnes vermelhas; vegetais folhosos
Arredores do Local de Trabalho	1. Local de Trabalho	-
	2. Restaurante	Bebidas; carboidratos; carnes brancas; carnes vermelhas; vegetais folhosos
	3. Posto de Saúde	Monitor de pressão sanguínea;
	4. Farmácia	Balança; medicamentos para controle de pressão arterial e colesterol
	5. Mercado	Frutas; carboidratos; bebidas

1. Opiniões referentes à funcionalidade de Busca de recursos baseada na localização:

(a) Frequência de uso: Baixa;

(b) Justificativa de frequência de uso: “Utilizei a busca de recursos baseada na localização, mas como as opções que foram demonstradas pelo aplicativo tornaram-se repetitivas, a frequência de uso foi baixa.”;

- (c) Dificuldades encontradas: “O número de opções que o aplicativo demonstrou foram poucas.”;
- (d) Pontos positivos: “Os aspectos positivos da localização de recursos é a facilidade de utilização para o dia-a-dia, para manter a rotina de um controle alimentar e da pressão arterial.”.

2. Opiniões referentes à funcionalidade de Entrada de dados de sinal vital:

- (a) Frequência de uso: Alta;
- (b) Justificativa de frequência de uso: “Utilizei diariamente para conseguir manter o controle da pressão arterial e do peso corporal.”;
- (c) Dificuldades encontradas: “Para inserir os dados com melhor precisão seria melhor listas de opções. Em alguns momentos de uso o sistema demonstrou o agrupamento de dados a serem inseridos. Por exemplo, quando inseria os dados de pressão aparecia juntamente as partes para inserir alimentos.”;
- (d) Pontos positivos: “De extrema importância para o controle de pressão arterial. Ajuda no controle do peso, pois é necessário inserir os dados de peso, proporcionando um cuidado maior para conseguir atingir o objetivo.”.

3. Opiniões referentes à funcionalidade de Visualização de atividade:

- (a) Frequência de uso: Alta;
- (b) Justificativa de frequência de uso: “No tratamento e controle da pressão arterial e do colesterol são necessários cuidados variados. Com os alertas foi possível administrar melhor as tarefas.”;
- (c) Dificuldades encontradas: “Não encontrei dificuldades nessa função”;
- (d) Pontos positivos: “Os alertas foram de grande importância para conseguir administrar os alimentos ingeridos, sendo que inicialmente consegui manter uma dieta na utilização do aplicativo. De extremamente importância para o controle da pressão arterial e dos medicamentos, sendo que contribuiu para uma melhora significativa no controle de minha pressão arterial.”.

4. Opiniões referentes à funcionalidade de Visualização de progresso de objetivo:

- (a) Frequência de uso: Média;
- (b) Justificativa de frequência de uso: “O progresso de objetivos foi utilizado inicialmente. Como no meio do processo de uso do aplicativo ocorreram festas de final de ano e minha rotina alimentar mudou bastante, não obtive muitos progressos.”
- (c) Dificuldades encontradas: “Seria interessante se ele demonstrasse não somente o último dado inserido, e sim uma amostra do progresso geral.”;

- (d) Pontos positivos: “Incentiva positivamente no controle alimentar e também da pressão arterial, sendo que cada dado inserido demonstra o progresso obtido no tratamento e controle.”.

Além dos tipos de contexto utilizados no controle dos objetivos do usuário, o ChronicDuctor possibilitou também o armazenamento de dados de localização e registros de refeição. Os contextos de localização foram registrados automaticamente pelo middleware, que foi configurado para receber atualizações de localização a cada 30 minutos, caso houvesse um deslocamento superior a 30 metros. Esta configuração foi utilizada pois as atividades mais significativas do usuário demandavam mais de 30 minutos, por exemplo, trabalho, almoço, e deslocamento entre residência e trabalho. Os dados de peso, tipo e porção de alimentos, e pressão arterial sistólica e diastólica, foram inseridos manualmente pelo usuário. A Tabela 12 apresenta a quantidade de contextos coletados e o tamanho gasto com o armazenamento destes dados. A maior quantidade de dados manuais inseridos é relacionado ao registro de refeições devido a frequência de ocorrência destas atividades.

Tabela 12: Total de contextos coletados

Tipo de Contexto	Descrição	Total Coletado	Tamanho Ocupado (bytes)
Localização GPS	Coordenadas de longitude e latitude oriundas do GPS	766	344.928
Localização ReDuctor	Contexto que representa o lugar onde o usuário estava localizado. Além disso, esse tipo de contexto fornece uma descrição semântica da localização do usuário, por exemplo, casa ou trabalho	477	222.134
Localização baseada em rede	Localização de menor precisão que utiliza informações de força de sinal de rede celular ou rede wi-fi. Este tipo de localização geralmente era utilizada quando o GPS não estava ativo, ou estava em inicialização	158	73.429
Peso	Medida de peso inserida pelo usuário	41	10.658
Porção	Porção ingerida de determinado alimento	191	52.138
Pressão Arterial Diastólica	Medida da pressão arterial diastólica inserida pelo usuário	49	13.859
Pressão Arterial Sistólica	Medida da pressão arterial sistólica inserida pelo usuário	49	13.780
Tipo de Alimento	Tipo de alimento ingerido em alguma refeição	191	52.776
	Total:	1.922	783.702

Através do teste estendido foi possível observar aspectos e situações que não haviam ocorrido nos outros testes efetuados com o aplicativo. Um destes aspectos foi o baixo uso da funcionalidade de localização de recursos, embora esta funcionalidade tenha sido entendida como útil

tanto pelos usuários que efetuaram o teste de aceitação do ChronicDuctor, quanto pelo usuário que efetuou o teste estendido. Pelas afirmações dadas pelo usuário, o baixo índice de uso deveu-se ao fato de não haver renovação dos recursos apresentados. Desta forma, é necessário o acréscimo de funcionalidades para registro de novos recursos e locais.

De acordo com o relato do usuário houve uma inconsistência na apresentação dos formulários de entrada de sinais vitais. Esta falha deve-se à biblioteca ARQoid retornar resultados inválidos para algumas consultas. Uma solução para o problema não foi efetuada, já que o mesmo ocorria de forma aleatória, sendo difícil reproduzi-lo. Além disso, o problema não afetou o uso do aplicativo, de acordo com informações dadas pelo usuário.

A visualização de atividades e a entrada de sinais vitais foi bastante utilizada. A frequência de uso da visualização de atividades deve-se ao fato do aplicativo estar centralizado no gerenciamento destas atividades. A entrada de sinais vitais teve seu uso elevado pois a mesma é o meio de fazer a inserção de dados de controle. No entanto, esta funcionalidade precisa de aperfeiçoamento nos aspectos de usabilidade, pois o usuário necessitou inserir dados de refeição que em algumas ocasiões se repetem. Funcionalidades como lista de opções ou auto completar podem auxiliar o usuário nessa tarefa.

A visualização do progresso foi utilizada de forma moderada pelo usuário. Pelos relatos do usuário, é necessário um aperfeiçoamento no modo de visualização desse progresso. O ChronicDuctor exibe os alertas de progresso junto com outros avisos do aplicativo. Este agrupamento pode ter dificultado o usuário a acompanhar o progresso de seu cuidado. A utilização de gráficos para acompanhamento de progresso foi utilizada no WD (MATTILA et al., 2008) e é uma alternativa que pode ser utilizada no ChronicDuctor.

Os resultados relacionados ao controle de objetivos do usuário não foram conclusivos. Dos três controles efetuados, peso, pressão arterial diastólica e pressão arterial sistólica, o usuário obteve sucesso apenas no controle da pressão arterial diastólica. No controle da pressão arterial diastólica o usuário obteve progresso, mas não chegou a atingir o objetivo. Já no controle de peso o usuário teve um retrocesso. É possível que algumas das dificuldades relatadas pelo usuário tenham contribuído para uma possível perda de foco no controle de seus objetivos. Assim, seria adequado que um teste de efetividade do aplicativo fosse efetuado após a eliminação destas dificuldades.

No período do teste estendido foram coletados 1922 contextos. Destes, 521 foram inseridos pelo usuário. O tamanho médio de cada registro de contexto ficou em torno de 408 bytes. Este tamanho deve-se ao fato do valor do contexto ter sido armazenado como uma instância da classe Contexto, serializada no formato JSON. O valor de contexto foi armazenado desta maneira por permitir maior facilidade na restauração do estado do objeto, e na comunicação entre processos. Além disso, a maioria das linguagens de programação dispõe de bibliotecas para manipulação de objetos em JSON, o que torna prático o uso da informação nesse formato. Como contrapartida, não é possível utilizar funções de agregação como, por exemplo, somar, média, contar, maior, ou menor, existentes em bases de dados relacionais para manipular esses

dados.

5.4 Avaliação de Consumo de Bateria

Uma avaliação de consumo de bateria foi efetuada utilizando a versão do middleware de plataforma móvel. O objetivo desta avaliação foi produzir dados de referência do percentual de consumo de bateria para trabalhos futuros. A avaliação foi efetuada em um smartphone Samsung GT-S5570B, onde foi instalado o aplicativo Battery Stats Plus¹⁵, utilizado para coletar os dados de consumo de recursos desejados.

Foram executados um total de oito testes, que foram divididos em dois tipos: com bloqueio de suspensão, e sem bloqueio de suspensão. As etapas efetuadas em cada teste são listadas na Tabela 13. A divisão dos testes em dois tipos deveu-se a um problema que ocorria quando o sistema operacional entrava em modo de suspensão. Ao entrar em modo de suspensão o sistema operacional interrompe o uso de CPU, fazendo com que o conector REST de servidor, executado nos NPs, tivesse sua execução finalizada. Desta maneira, funcionalidades como compartilhamento de recursos, recepção de eventos de mudança de contexto e recepção de mensagens eram cessadas. Para corrigir este problema foi utilizado um mecanismo de bloqueio de suspensão da API do Android¹⁶, que evita que o sistema entre em modo de suspensão. Nos experimentos com bloqueio de suspensão utilizou-se o tipo de bloqueio SCREEN_DIM_WAKE_LOCK, que mantém a tela do aparelho ligada com brilho diminuído (POWER MANAGER | ANDROID DEVELOPERS, 2012).

Tabela 13: Roteiro do teste de bateria

Etapa	Descrição
1.	Smartphone é iniciado com carga total de bateria;
2.	Middleware é iniciado após o boot do sistema operacional;
3.	Sistema permanece em execução até que a carga da bateria seja completamente consumida, desligando o smartphone;
4.	Smartphone é ligado conectado a rede elétrica. As estatísticas da última execução são registradas;
5.	A bateria do smartphone é recarregada;
6.	Sistema é reiniciado para execução de um novo teste.

O middleware foi configurado para receber atualizações de GPS a cada 30 minutos caso o smartphone tivesse um deslocamento superior a 30 metros em todos os testes. Dessa forma, os componentes de Contexto e Local foram utilizados para registro e resolução de localização. As outras funcionalidades do middleware não foram utilizadas.

A Tabela 14 apresenta os resultados obtidos. O indicador Média de tempo de vida descreve o tempo médio que o dispositivo permaneceu ligado para cada tipo teste. Já o indicador Média

¹⁵<http://battery.rootinstaller.com/start>

¹⁶<http://developer.android.com/reference/android/os/PowerManager.WakeLock.html>

de uso de bateria fornece a média do percentual de bateria utilizado pelo middleware em cada tipo de teste.

Tabela 14: Dados coletados sobre consumo de bateria

Indicador	Com bloqueio de suspensão	Sem bloqueio de suspensão
Média de tempo de vida	9h37m	16h28m
Média de uso de bateria (%)	4,84%	4,10%

De acordo com os resultados, é possível verificar que houve uma redução média de 15,15% no consumo de bateria pelo middleware e um ganho médio de 71,08% de tempo de vida da bateria do dispositivo, quando não há bloqueio de suspensão. A diferença significativa entre a média de tempo de vida de bateria deve-se ao bloqueio de suspensão manter a tela do dispositivo ligada, acarretando em um maior consumo de bateria pelo sistema operacional, e consequentemente numa redução do tempo de vida de bateria.

Para que os NPs permaneçam funcionais sem a necessidade de utilização de bloqueio de suspensão será necessário que uma adaptação no modelo seja efetuada. Nesta solução o NP passaria a ser executado em um computador sem restrição de consumo de bateria, e o dispositivo móvel se tornaria um cliente associado ao NP, requisitando novos contextos e mensagens quando saísse do modo de suspensão.

6 CONSIDERAÇÕES FINAIS

O U'Ductor é um modelo para cuidado ubíquo de DCNT que concentra-se em preencher a lacuna apresentada em outros trabalhos com relação a integração entre pacientes e recursos da comunidade e organizações de saúde, sendo este tipo de integração uma característica relevante e citada nas documentações de modelos de cuidados crônicos (ICIC, 2011; OMS, 2003).

O modelo define uma arquitetura distribuída, baseada no estilo arquitetural REST (FIELDING; TAYLOR, 2000), e que propicia a escalabilidade localizada no acesso a recursos e informações de contexto (SATYANARAYANAN, 2001). Isto é possível já que os nodos ficam ligados a nodos locais que representam o local onde estão fisicamente presentes. A ontologia proposta possibilita a descrição semântica de seus elementos, facilitando a sua localização. A aplicação ChronicDuctor foi criada para explorar as funcionalidades da arquitetura proposta no cuidado de DCNT, oferecendo suporte ao autogerenciamento e integração com recursos e membros da comunidade e organizações de saúde.

Para avaliar a viabilidade do modelo efetuaram-se dois experimentos. O primeiro experimento foi um teste de aceitação de tecnologia por dez sujeitos que possuíam algum tipo de DCNT ou fator de risco. O segundo experimento consistiu de um teste de funcionalidade estendido do protótipo, efetuado por usuário portador de DCNT, e que teve a duração de um mês e meio. Além destes dois experimentos, foi efetuado uma avaliação de consumo de bateria pelo middleware, com o objetivo de produzir indicadores de referências para trabalhos futuros.

As seções a seguir descrevem as conclusões alcançadas com o desenvolvimento deste trabalho, as contribuições que este deixa, e sugestões para sua continuação.

6.1 Conclusões

Os resultados obtidos no teste de aceitação demonstraram que o modelo proposto foi aceito pelos sujeitos que participaram de sua avaliação. Se os índices de concordância total e concordância parcial forem somados, é possível dizer que houve uma aceitação de 100% dos usuários tanto em relação a utilidade percebida, quanto em relação a facilidade de uso percebida.

No que diz respeito à utilidade percebida, houve uma concordância total por parte de 95% dos usuários. Uma vez que parte destas questões dão enfoque às integrações entre pacientes e membros da comunidade ou organizações de saúde, e entre pacientes e recursos da comunidade ou organizações de saúde, entende-se que sistemas ubíquos que deem suporte ao cuidado de DCNT devem também oferecer mecanismos que suportem estes tipos de integração.

Com relação a facilidade de uso percebida, houve uma concordância total de 91,3% dos indivíduos, o que mostra que é necessário ajustes de aspectos relacionados à interação entre usuário e aplicativo. Esse resultado vai ao encontro dos comentários feitos pelos sujeitos #6 e #8, que sugerem a organizações de conversas em pastas, possibilidade para criação de recursos e locais, alertas de voz motivacionais, e personalização das mensagens de aviso. O sujeito

#8 ainda sugere a emissão de alertas a unidades de atendimentos próximas de acordo com o contexto do usuário. Embora este tipo de funcionalidade não tenha sido explorada no protótipo, é possível implementá-la com o uso do U'Ductor.

O teste estendido também evidenciou a necessidade de aperfeiçoamentos nos aspectos relacionados à facilidade de uso. Através do teste estendido foi possível identificar a necessidade de criar mecanismos para facilitar a entrada de dados de sinais vitais e o acompanhamento do progresso de cuidado.

6.2 Contribuições

O U'Ductor complementa os trabalhos estudados ao dar suporte à característica Integração Paciente x Recursos Comunidade/Organização de Saúde que não foi explorada nos trabalhos estudados. A Tabela 15 complementa a Tabela 2, apresentada na seção 3.6, ao descrever como o U'Ductor provê suporte ao autogerenciamento, gerenciamento de fatores de risco e integrações com o paciente.

Devido a arquitetura utilizada pelo modelo, é possível encontrar recursos de auxílio de acordo com a localização do paciente. Nesse modelo, o paciente tem liberdade para buscar recursos de auxílio a qualquer momento e qualquer local, não ficando restrito ao suporte oferecido pela organização de saúde (MAMYKINA et al., 2008; ROSSO et al., 2010; KOUTKIAS et al., 2010; PAGANELLI; GIULI, 2011), ou ao auxílio de contatos conhecidos (MATTILA et al., 2008).

Complementarmente, o modelo possibilita localizar pessoas e material de apoio que deem assistência na execução das atividades de Controle Alimentar e Atividade Física. Em tarefas de uso de Medicamento, é possível buscar locais que disponibilizem o medicamento prescrito. Nas integrações Paciente x Membros Comunidade e Integração Paciente x Membros Organização Saúde é possível conversar com o membro da comunidade, ou organização, através do dispositivo móvel. E com relação ao Suporte à Educação, é possível o compartilhamento de material de apoio e instruções de uso de recursos.

O U'Ductor foi projetado para suportar o gerenciamento de fatores de risco, como também suportar o trabalho colaborativo entre os principais membros participantes da cadeia de cuidado de DCNT. Para tanto, foi criado um modelo que se divide em quatro componentes: ChronicDuctor, ConnDuctor, ReDuctor e modelo ontológico. O ChronicDuctor funciona como o assistente de saúde do paciente crônico, lembrando-o e auxiliando-o nas tarefas diárias de tratamento. Já o ConnDuctor e o ReDuctor geram uma estrutura descentralizada, que possibilita a colaboração direta entre os participantes das atividades de cuidado de DCNT. O modelo ontológico é um meio para descrever informações e atividades que dizem respeito a pacientes, comunidades e organizações médicas.

Um artigo apresentando o modelo U'Ductor foi publicado nos anais do XIII Simpósio em Sistemas Computacionais (WSCAD-SSC 2012), com o título U'Ductor: a Model for Support-

ing Ubiquitous Chronic Disease Care Management (VIANNA et al., 2012). Neste artigo foram apresentadas a arquitetura do U'Ductor e uma versão do ChronicDuctor em estágio inicial de desenvolvimento. Naquela fase, o ChronicDuctor permitia a configuração de plano de cuidado e a recomendação de recursos de cuidado de acordo com a atividade e localização do usuário. O ChronicDuctor foi utilizado por uma paciente com diagnóstico de sobrepeso e hipertensão durante três dias para avaliar o seu uso como ferramenta de suporte ao cuidado de DCNT. Durante esses dias, a paciente recebeu recomendações de recursos disponíveis nos arredores do seu local de trabalho para cada atividade agendada. Após o período de uso, a paciente foi convidada a responder um questionário de avaliação da aplicação. As respostas dadas pela paciente indicaram a viabilidade do uso do ChronicDuctor como ferramenta de suporte ao cuidado de DCNT.

6.3 Trabalhos Futuros

De acordo com os resultados obtidos nos testes de avaliação do modelo propõe-se, para dar continuidade a este trabalho, a adição de um novo elemento ao U'Ductor, a inserção de elementos de gamification no ChronicDuctor, e um teste para avaliação de sua efetividade.

Durante o teste estendido foi identificado que o conector REST de servidor, executado nos NPs, tinha sua execução finalizada quando o sistema operacional Android entrava em estado de suspensão. Com isso, algumas das funcionalidades do middleware como, por exemplo, compartilhamento de recursos, recepção de eventos de mudança de contexto e recepção de mensagens eram interrompidas. A maneira utilizada para resolver este problema no Android foi utilizar o bloqueio de suspensão oferecido pela API do Android, que faz com que o sistema não entre em estado de suspensão. No entanto, esta solução acarreta em redução do tempo de vida da bateria, pois o CPU continua em uso¹.

Uma solução para este problema seria o readequamento do modelo, onde o NP não executaria no smartphone do usuário, mas em um computador sem restrição de consumo de bateria. Desta forma, os recursos oferecidos pelo NP ficariam disponíveis independente do estado do smartphone. Nessa solução o smartphone se tornaria um nodo associado ao NP, e seria responsável por:

- notificar o NP a respeito de trocas de contexto que são gerenciadas por ele, como por exemplo, mudança na posição GPS;
- enviar recursos compartilhados ao NP;
- solicitar novas mensagens e novos contextos sempre que o smartphone estiver ativo; e
- gerenciar os módulos executáveis, notificando-os a respeito de novas mensagens e novos contextos.

¹<http://developer.android.com/reference/android/os/PowerManager.html>

Tabela 15: Suportes Oferecidos pelo U'Ductor

Tipo de Suporte	Característica	U'Ductor
Autogerenciamento	Suporte ao autogerenciamento	Sim
Gerenciamento dos Fatores de Risco	Controle Alimentar	Lembrete de atividade; localização de assistência para atividade através de pessoas e material de apoio; localização de locais que ofereçam alimentação conforme plano.
	Atividade Física	Lembrete de atividade; localização de assistência para atividade através de pessoas e material de apoio; programação de objetivos para atividade.
	Medicamento	Lembrete de atividade; busca de locais com disponibilidade de medicamento; localização de assistência para atividade através de pessoas e material de apoio.
	Sensoriamento	Modelo de sensoriamento genérico.
Integrações com Paciente	Integração Paciente x Recursos Comunidade/ Organização Saúde	Localização de recursos compartilhados.
	Integração Paciente x Membros Comunidade	Localização de pessoas com o perfil necessário para auxílio; conversação com o membro através do dispositivo móvel.
	Integração Paciente x Membros Organização Saúde	Localização de pessoas com o perfil necessário para auxílio; conversação com o membro através do dispositivo móvel.
	Suporte à Educação	Compartilhamento de material de apoio e instruções de uso de recursos.

Fonte: Elaborado pelo autor.

Esta modificação traria pequeno impacto, pois a arquitetura do middleware se manteria igual, sendo necessário o acréscimo no modelo de um tipo de nodo que represente um dispositivo móvel. Assim, a interrupção de comunicação entre um NP e seu smartphone não prejudicaria a colaboração entre os NPs, pois estes executariam continuamente.

A inserção de elementos de gamification no ChronicDuctor é outro estudo relevante. Gamification pode ser entendido como o uso de elementos de jogos em contextos não relacionados a

jogos (DETERDING et al., 2011). Elementos de jogos, tais como avatares, regras, pontuação, reputação e equipes, são utilizados para aumentar o engajamento do usuário nas suas rotinas. O uso de Gamification pode ser uma forma de suprir a carência de elementos motivadores, citada pelo sujeito #8 em seus comentários. Assim, entende-se que a aplicação destes aspectos no ChronicDuctor pode aperfeiçoar o desempenho de seus usuários em alcançar ganhos de saúde.

Estudos deste tipo de aplicação podem ser encontrados em trabalhos como Monster & Gold (BUTTUSSI; CHITTARO, 2010) e American Horsepower Challenge (XU et al., 2012). Monster & Gold é um jogo executado em um smartphone que possui comunicação com um oxímetro de pulso, sendo utilizado para auxiliar seus usuários a adquirirem benefícios cardíacos. O American Horsepower Challenge foi uma competição entre escolas americanas onde estudantes utilizaram pedômetros para medir a quantidade de passos efetuados durante as atividades físicas. Os estudantes enviavam os dados de suas atividades através do website da competição, que classificava as escolas de acordo com o total de passos de seus alunos. Quanto maior a quantidade de passos melhor classificada a escola ficava.

Por fim, a avaliação de efetividade é necessária, uma vez que os resultados produzidos pelo teste de uso estendido não são conclusivos. Dos três controles efetuados, peso, pressão arterial diastólica e pressão arterial sistólica, o usuário obteve sucesso apenas no controle da pressão arterial diastólica. No controle da pressão arterial diastólica o usuário obteve progresso, mas não chegou a atingir o objetivo. Já no controle de peso o usuário teve um retrocesso. Além disso, é necessário que o teste seja efetuado por um maior número de indivíduos para obtenção de resultados mais conclusivos.

Imagina-se que o readequamento do U'Ductor e a inserção de elementos de gamification devem melhorar a eficiência do uso do modelo no cuidado de DCNT. Já a avaliação de efetividade deve ser feita para averiguar o potencial do ChronicDuctor no cuidado de DCNT.

REFERÊNCIAS

- AGOULMINE, N.; DEEN, M.; LEE, J.-s.; MEYYAPPAN, M. U-Health Smart Home. *IEEE Nanotechnology Magazine*, [S.l.], v. 5, n. 3, p. 6–11, Sept. 2011.
- AVERBUJ, D.; TIJMAN, G. (Ed.). *Guias essenciais para a saúde: obesidade*. Porto Alegre: L&PM Editores, 2011. 64 p.
- BARBOSA, J. L. V.; HAHN, R. M.; BARBOSA, D. N. F.; SACCOL, A. I. d. C. Z. A ubiquitous learning model focused on learner interaction. *Int. J. Learn. Technol.*, Inderscience Publishers, Geneva, SWITZERLAND, v. 6, p. 62–83, May 2011.
- BARDRAM, E. Activity-based computing: support for mobility and collaboration in ubiquitous computing. *Personal Ubiquitous Comput.*, London, UK, UK, v. 9, n. 5, p. 312–322, Sept. 2005.
- BARDRAM, J. E.; CHRISTENSEN, H. B. Pervasive Computing Support for Hospitals: an overview of the activity-based computing project. *Pervasive Computing*, IEEE, [S.l.], v. 6, n. 1, p. 44 –51, jan.-march 2007.
- BERNERS-LEE, T.; FIELDING, R. T.; MASINTER, L. Uniform Resource Identifier (URI): generic syntax. Fremont, CA, USA: RFC Editor, 2005. RFC. (3986).
- BISCHOFFS, L.; HASSELBRING, W.; SCHLEGELMILCH, J.; STEFFENS, U. A Hierarchical Super Peer Network for Distributed Artifacts. In: *PRE-PROCEEDINGS OF THE SIXTH THEMATIC WORKSHOP OF THE EU NETWORK OF EXCELLENCE DELOS*. S. MARGHERITA DI PULA, 2004. Anais. . . [S.l.: s.n.], 2004. p. 105–114.
- BODENHEIMER, T.; WAGNER, E. H.; GRUMBACH, K. Improving Primary Care for Patients With Chronic Illness. *JAMA, Family and Community Medicine*, University of California, San Francisco, USA. tbodie@earthlink.net, v. 288, n. 14, p. 1775–1779, Oct. 2002.
- BUTTUSSI, F.; CHITTARO, L. Smarter Phones for Healthier Lifestyles: an adaptive fitness game. *Pervasive Computing*, IEEE, [S.l.], v. 9, n. 4, p. 51 –57, october-december 2010.
- CACERES, C.; FERNANDEZ, A.; OSSOWSKI, S.; VASIRANI, M. Agent-Based Semantic Service Discovery for Healthcare: an organizational approach. *Intelligent Systems*, IEEE, [S.l.], v. 21, n. 6, p. 11 –20, nov.-dec. 2006.
- CASTRO, A. M. de; SIMONI, C. L. de; GONÇALVES, C. C. M.; GOSCH, C. S.; MALTA, D. d. C.; SARDINHA, L. M. V. *Diretrizes e Recomendações para o Cuidado Integral de Doenças Crônicas Não-Transmissíveis*. Brasília: Ministério da Saúde. Secretaria de Vigilância à Saúde. Secretaria de Atenção à Saúde., 2008.
- COSTA, C. A. da. *A Context-aware Service-based Software Infrastructure for Ubiquitous Computing*. 2008. Tese (Doutorado em Ciência da Computação) — Universidade Federal do Rio Grande do Sul., 2008.
- DAVIS, F. D. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Q.*, Minneapolis, MN, USA, v. 13, n. 3, p. 319–340, Sept. 1989.

DETERDING, S.; DIXON, D.; KHALED, R.; NACKE, L. From game design elements to gamefulness: defining "gamification". In: INTERNATIONAL ACADEMIC MINDTREK CONFERENCE: ENVISIONING FUTURE MEDIA ENVIRONMENTS, 15., 2011, New York, NY, USA. Proceedings. . . ACM, 2011. p. 9–15. (MindTrek '11).

DEVELOPERS, A. What is Android? [Online; acessado em 8 de abril de 2012], <http://developer.android.com/guide/basics/what-is-android.html>.

DEY, A.; ABOWD, G.; SALBER, D. A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications. *Human-Computer Interaction*, [S.l.], v. 16, n. 2, p. 97–166, Dec. 2001.

DEY, A. K.; ESTRIN, D. Perspectives on Pervasive Health from Some of the Field's Leading Researchers. *Pervasive Computing*, IEEE, [S.l.], v. 10, n. 2, p. 4–7, feb. 2011.

FIELDING, R. T.; TAYLOR, R. N. Principled design of the modern Web architecture. In: SOFTWARE ENGINEERING, 22., 2000, New York, NY, USA. Proceedings. . . ACM, 2000. p. 407–416. (ICSE '00).

FRANCO, L. K.; ROSA, J. H.; BARBOSA, J. L.; COSTA, C. A.; YAMIN, A. C. MUCS: a model for ubiquitous commerce support. *Electronic Commerce Research and Applications*, [S.l.], v. 10, n. 2, p. 237 – 246, 2011. Special Issue on Electronic Auctions: Strategies and Methods.

FREED, E. E.; BORENSTEIN, N. S. Multipurpose Internet Mail Extensions (MIME). Part Two: media types. [S.l.]: Internet Engineering Task Force, 1996. (RFC 2046).

GAMBLE, M.; GOBLE, C. Standing on the shoulders of the trusted web: trust, scholarship and linked data. In: WEBSCI10: EXTENDING THE FRONTIERS OF SOCIETY ON-LINE, 2010. Anais. . . [S.l.: s.n.], 2010.

HOLTMANN, C.; MÜLLER-GORCHS, M.; RASHID, A.; KLAUSWEIDENHAUPT; ZIEGLER, V.; GRIEWING, B.; WEINHARDT, C. Medical Opportunities by Mobile IT Usage – A Case Study in the Stroke Chain of Survival. In: EUROPEAN CONFERENCE ON EHEALTH 2007, 2007. Anais. . . Gesellschaft für Informatik: Bonn, 2007. p. 115–130.

ICIC. The Chronic Care Model. [Online; acessado em 28 de janeiro de 2012], http://www.improvingchroniccare.org/index.php?p=The_Chronic_Care_Model&s=2.

INKSTER, T.; STORY, H.; HARBULOT, B.; BACHMAN-GMÜR, R. WebID 1.0. 2011. 1–17 p. n. December.

JEONG, K.; JUNG, E. young; PARK, D. K. Trend of wireless u-Health. In: COMMUNICATIONS AND INFORMATION TECHNOLOGY, 2009. ISCIT 2009. 9TH INTERNATIONAL SYMPOSIUM ON, 2009. Anais. . . [S.l.: s.n.], 2009. p. 829 –833.

KINDBERG, T.; FOX, A. System software for ubiquitous computing. *Pervasive Computing*, IEEE, [S.l.], v. 1, n. 1, p. 70 – 81, jan-mar 2002.

KOUTKIAS, V. G.; CHOUVARDA, I.; TRIANTAFYLLIDIS, A.; MALOUSI, A.; GIAGLIS, G. D.; MAGLAVERAS, N. A personalized framework for medication treatment management in chronic care. *IEEE transactions on information technology in biomedicine* : a publication of the IEEE Engineering in Medicine and Biology Society, [S.l.], v. 14, n. 2, p. 464–72, Mar. 2010.

- LAKE, A. J.; STAIGER, P. K. Seeking the views of health professionals on translating chronic disease self-management models into practice. *Patient Education and Counseling*, [S.l.], v. 79, n. 1, p. 62 – 68, 2010.
- LIKERT, R. A technique for the measurement of attitudes. *Archives of Psychology*, [S.l.], v. 22, n. 140, p. 1–55, 1932.
- LIM, J.-E.; CHOI, O.-H.; NA, H.-S.; BAIK, D.-K. A context-aware fitness guide system for exercise optimization in U-health. *IEEE transactions on information technology in biomedicine : a publication of the IEEE Engineering in Medicine and Biology Society*, [S.l.], v. 13, n. 3, p. 370–9, May 2009.
- MAMYKINA, L.; MYNATT, E.; DAVIDSON, P.; GREENBLATT, D. MAHI: investigation of social scaffolding for reflective thinking in diabetes management. In: *PROCEEDING OF THE TWENTY-SIXTH ANNUAL SIGCHI CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS, 2008*, New York, NY, USA. Anais. . . ACM, 2008. p. 477–486. (CHI '08).
- MATILTA, E.; PÄRKKÄ, J.; HERMERSDORF, M.; KAASINEN, J.; VAINIO, J.; SAMPOSALO, K.; MERILAHTI, J.; KOLARI, J.; KULJU, M.; LAPPALAINEN, R.; KORHONEN, I. Mobile diary for wellness management—results on usage and usability in two user studies. *IEEE transactions on information technology in biomedicine : a publication of the IEEE Engineering in Medicine and Biology Society*, [S.l.], v. 12, n. 4, p. 501–12, July 2008.
- MEEKER, M.; DAWSON, J.; LU, J.; LU, B.; JI, R.; DEVITT, S.; FLANNERY, S.; DELFAS, N.; SCHNEIDER, M.; MEEKER, M.; FLANNERY, S.; SCHNEIDER, M.; DAWSON, J.; JI, R. The Mobile Internet Report - Ramping Faster than Desktop Internet, the Mobile Internet Will Be Bigger than Most Think. [S.l.]: MORGAN STANLEY RESEARCH, 2009.
- OMS. Cuidados inovadores para condições crônicas: componentes estruturais de ação: relatório mundial. Brasília: World Health Organization, 2003.
- OMS. Preventing CHRONIC DISEASES a vital investment. Geneva: World Health Organization, 2005.
- OMS. WHO global strategy on diet, physical activity and health: a framework to monitor and evaluate implementation. 2008.
- OMS. Global status report on noncommunicable diseases 2010. 20 Avenue Appia, 1211 Geneva 27, Switzerland: World Health Organization, 2010.
- OMS. World Health Statistics 2011. 20 Avenue Appia, 1211 Geneva 27, Switzerland: World Health Organization, 2011.
- ORWAT, C.; RASHID, A.; HOLTSMANN, C.; WOLK, M.; SCHEERMESSER, M.; KOSOW, H.; GRAEFE, A. Adopting Pervasive Computing for Routine Use in Healthcare. *Pervasive Computing*, IEEE, [S.l.], v. 9, n. 2, p. 64 –71, april-june 2010.
- PADGHAM, L.; THANGARAJAH, J.; PAUL, P. Prometheus Design Tool Version 2.5 User Manual. [Online; acessado em 24 de março de 2012].

PADGHAM, L.; THANGARAJAH, J.; WINIKOFF, M. The Prometheus Design Tool – A Conference Management System Case Study. In: LUCK, M.; PADGHAM, L. (Ed.). Agent-Oriented Software Engineering VIII. [S.l.]: Springer Berlin / Heidelberg, 2008. p. 197–211. (Lecture Notes in Computer Science, v. 4951).

PADGHAM, L.; WINIKOFF, M. Developing Intelligent Agent Systems: a practical guide. [S.l.]: John Wiley and Sons, 2004.

PAGANELLI, F.; GIULI, D. An ontology-based system for context-aware and configurable services to support home-based continuous care. IEEE transactions on information technology in biomedicine : a publication of the IEEE Engineering in Medicine and Biology Society, [S.l.], v. 15, n. 2, p. 324–33, Mar. 2011.

PAGANELLI, F.; SPINICCI, E.; GIULI, D. ERMHAN: a context-aware service platform to support continuous care networks for home-based assistance. Int. J. Telemedicine Appl., New York, NY, United States, v. 2008, p. 4:1–4:13, Jan. 2008.

PAUTASSO, C.; ZIMMERMANN, O.; LEYMANN, F. Restful web services vs. "big" web services: making the right architectural decision. In: WORLD WIDE WEB, 17., 2008, New York, NY, USA. Proceedings. . . ACM, 2008. p. 805–814. (WWW '08).

POWER Manager | Android Developers. [Online; acessado em 10 de outubro de 2012], <http://developer.android.com/reference/android/os/PowerManager.html>.

SPARQL Query Language for RDF. [S.l.]: World Wide Web Consortium, 2008. RASHID,

A.; SCHLÜFTER, F.; HOLTSMANN, C.; KUNZE, C.; THALER, K.; DLÄUMER, M.; SCHLESINGER, S.; GRIEWING, B. Usage of Accelerometers in Home Care for multiple sclerosis patients. In: EUROPEAN CONFERENCE ON EHEALTH 2007, 2007. Anais. . . Gesellschaft für Informatik: Bonn, 2007. p. 177–192.

RODRIGUES, S.; DILLI, R.; VENECIAN, L.; YAMIN, A.; COSTA, C.; BARBOSA, J.; LOPES, J.; SOUZA, R.; GEYER, C. A Semantic-Based Software Architecture for Processing Context Information in Ubiquitous Medicine. Simpósio em Sistemas Computacionais, Los Alamitos, CA, USA, v. 0, p. 7, 2011.

ROSSO, R.; MUNARO, G.; SALVETTI, O.; COLANTONIO, S.; CIANCITTO, F. CHRONIOUS: an open, ubiquitous and adaptive chronic disease management platform for chronic obstructive pulmonary disease (copd), chronic kidney disease (ckd) and renal insufficiency. Conference proceedings: Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference, [S.l.], v. 2010, n. 2, p. 6850–3, Jan. 2010.

SATYANARAYANAN, M. Pervasive Computing: vision and challenges. IEEE Personal Communications, [S.l.], v. 8, p. 10–17, 2001.

SAÚDE, M. D. VIGITEL BRASIL 2010 - VIGILÂNCIA DE FATORES DE RISCO E PROTEÇÃO PARA DOENÇAS CRÔNICAS POR INQUÉRITO TELEFÔNICO. Brasília: MINISTÉRIO DA SAÚDE, 2011.

SCHOLLMEIER, R. A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications. In: PEER-TO-PEER COMPUTING, 2001.

- PROCEEDINGS. FIRST INTERNATIONAL CONFERENCE ON, 2001. Anais. . . [S.l.: s.n.], 2001. p. 101 –102.
- SILVA, J. M.; ROSA, J. a. H.; BARBOSA, J. L. V.; BARBOSA, D. N. F.; PALAZZO, L. a. M. Content distribution in trail-aware environments. *Journal of the Brazilian Computer Society*, [S.l.], v. 16, n. 3, p. 163–176, July 2010.
- VIANNA, H.; BARBOSA, J.; COSTA, C.; YAMIN, A. C. U’Ductor: a model for supporting ubiquitous chronic disease care management. In: *COMPUTER SYSTEMS (WSCAD-SSC), 2012 13TH SYMPOSIUM ON*, 2012. Anais. . . [S.l.: s.n.], 2012. p. 156 –162.
- WAGNER, E. H.; AUSTIN, B. T.; DAVIS, C.; HINDMARSH, M.; SCHAEFER, J.; BONOMI, A. Improving Chronic Illness Care: translating evidence into action. *Health Aff, W.A. MacColl Institute for Healthcare Innovation at the Center for Health Studies, Group Health Cooperative of Puget Sound, Seattle, USA.*, v. 20, n. 6, p. 64–78, Nov. 2001.
- WAGNER, E. H.; GROVE, T. Care for chronic diseases - The efficacy of coordinated and patient centred care is established, but now is the time to test its effectiveness. *British Medical Journal*, [S.l.], v. 325, p. 913–914, 2002.
- WEISER, M. The Computer for the 21st Century. *Scientific American*, [S.l.], p. 91–95, sept. 1991.
- XU, Y.; POOLE, E. S.; MILLER, A. D.; EIRIKSDOTTIR, E.; CATRAMBONE, R.; MYNATT, E. D. Designing pervasive health games for sustainability, adaptability and sociability. In: *INTERNATIONAL CONFERENCE ON THE FOUNDATIONS OF DIGITAL GAMES, 2012*, New York, NY, USA. *Proceedings. . . ACM*, 2012. p. 49–56. (FDG ’12).
- YOON, C.; KIM, S. Convenience and TAM in a ubiquitous computing environment: the case of wireless lan. *Electronic Commerce Research and Applications*, [S.l.], v. 6, n. 1, p. 102 – 112, 2007.