



Programa Interdisciplinar de Pós-Graduação em

Computação Aplicada

Mestrado Acadêmico

Paulo Henrique Santini

eGroup: Um Modelo para Gerenciamento de Grupos
Dinâmicos de Entidades

São Leopoldo, 2017

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

PAULO HENRIQUE SANTINI

eGroup:UM MODELO PARA GERENCIAMENTO DE GRUPOS DINÂMICOS DE
ENTIDADES

SÃO LEOPOLDO
2017

Paulo Henrique Santini

eGroup:Um Modelo para Gerenciamento de Grupos Dinâmicos de Entidades

Dissertação apresentada à Universidade do Vale do Rio dos Sinos – Unisinos, como requisito parcial para obtenção do título de Mestre em Computação Aplicada.

Aprovado em 30 de agosto de 2016

BANCA EXAMINADORA

Prof. Dr. Jorge Luis Victória Barbosa – Universidade do Vale do Rio dos Sinos (UNISINOS)

Prof. Dr. Leandro Krug Wives – Universidade Federal do Rio Grande do Sul (UFRGS)

Prof. Dr. Rodrigo da Rosa Righi – Universidade do Vale do Rio dos Sinos (UNISINOS)

Prof. Dr. Jorge Luis Victória Barbosa

Visto e permitida a impressão
São Leopoldo,

Prof. Dr. Sandro José Rigo
Coordenador PPG em Computação Aplicada

Ficha catalográfica

SS235e Santini, Paulo Henrique
eGroup : um modelo para gerenciamento de grupos dinâmicos
de entidades / por Paulo Henrique Santini– 2017.
87 f. : il. ; 30 cm.

Dissertação (mestrado) — Universidade do Vale do Rio dos
Sinos, Programa de Pós-graduação em Computação Aplicada,
São Leopoldo, RS, 2017.

“Orientador: Prof. Dr. Jorge Luis Victória Barbosa.”

1. Formação de grupos dinâmicos de entidades. 2. Gerenciamento de
grupos dinâmicos. 3. Aplicações ubíquas para gerenciamento de grupos.
4. Computação ubíqua. I. Título.

CDU: 004.415.2

Catálogo na Publicação:
Bibliotecário Alessandro Dietrich - CRB 10/2338

Aos meus pais, Gilmar e Neiva, e minha Irmã, Nicolý, que sempre estiveram ao meu lado durante a minha vida. Pelo amor, incentivo e investimento que sempre me propuseram desde a graduação até a realização desta dissertação, mas principalmente pela minha formação como pessoa.

A minha Madrinha, pelo suporte e aconchego durante os dois anos de mestrado.

A Professora Dra. Eliane Regina de Almeida Valiati por ter me impulsionado ao mestrado.

Ao Orientador Dr. Jorge Luis Victória Barbosa pelas oportunidades durante o mestrado.

Ao Professor Me. Márcio Garcia Martins pela ajuda durante o mestrado.

AGRADECIMENTOS

A Deus, pela vida e oportunidade que me destes.

Durante o mestrado, muitas pessoas me apoiaram e estiveram ao meu lado durante estes dois anos. Gostaria de maneira geral, agradecer a todos pela contribuição e incentivo que cada um me cedeu.

Agradeço primeiramente ao Santander Universidades, pelo investimento financeiro através de uma bolsa de mestrado para elaboração deste trabalho.

Especialmente aos meus pais, Gilmar e Neiva, e minha Irmã, Nicolý, que sempre estiveram ao meu lado durante a minha vida. Pelo amor, incentivo e investimento que sempre me propuseram desde a graduação até a realização deste trabalho, mas principalmente pela minha formação como pessoa.

À minha madrinha Arenilse, pelo suporte e aconchego durante os dois anos em que estive residindo em seu apartamento. A minha tia Leni, por ter me possibilitado estudar inglês em sua escola. A Clair Quevedo por ter me levado na Unisinos para o processo seletivo. E aos avós e demais familiares, que de uma forma ou de outra, estiveram comigo nessa trajetória.

À Professora Dra. Eliane Regina de Almeida Valiati, pelo apoio, pelas trocas de e-mails e por ter me impulsionado à tentar o mestrado. Ao Professor Dr. João Francisco Valiati pelo apoio e recepção na Unisinos. Aos meus professores de graduação do Senac, Professor Me. Eder Pazinato, Leandro Bianchi e Marlon Vanz Silva.

Ao meu orientador, Professor Dr. Jorge Luis Victória Barbosa, pela oportunidade que me destes, pelo incentivo, pelas correções, pelas revisões e orientação desde o início desta etapa da minha vida. Agradecer pelas tantas conversas e discussões sobre o trabalho, pelos artigos em parceria e pela atuação constante no MobiLab.

Ao meu Co-orientador Professor Dr. Cristiano André da Costa, pelas aulas, reuniões e incentivo. Aos Professores Drs. Sandro José Rigo e Rodrigo da Rosa Righi, por participarem da minha qualificação, e pelas ideias e sugestões apresentadas. Ao Pipca (Programa Interdisciplinar de Pós-graduação em Computação Aplicada) pelos esclarecimentos as várias dúvidas em relação ao mestrado, especialmente a Luciana Grimaldi Aquino e Bruna Severo.

Ao Professor Me. Márcio Garcia Martins pela parceria, ajuda, motivação, conversas, ideias e discussões sobre o trabalho. A Me. Tana Cassia Malacarne Martins pela amizade, conversas e vários almoços na Unisinos. Aos bolsistas atuante no MobiLab Wagner Manganelli e Renan Santos.

Agradeço ao Me. André Wagner pelas trocas de e-mails, pela disponibilização da sua dissertação e do seu trabalho que serviu de inspiração e base para o desenvolvimento desta dissertação.

Aos meus colegas de mestrado, pelo apoio, pelos dias de estudos e aprendizado em grupo. Agradecer especialmente ao colega Marcelo Josué Telles pela parceria, pelos vários dias de Unisinos e MobiLab, pelos artigos publicados e pelos não aceitos também. Ao colega Rodrigo Silva Pereira pelas várias conversas e apoio, pelos trabalhos realizados em grupo. Aos colegas Natália Navarro, Deividi Schumacher Velho, William Hart, Vinícius Facco Rodrigues, Felipe Plets, Milene Martini e Adalto Selau Sparremberger, pela parceria, discussões e estudos.

Aos meus amigos que contribuíram de alguma forma na evolução do trabalho.

"O único homem que nunca comete erros é aquele que nunca faz coisa alguma. Não tenha medo de errar, pois você aprenderá a não cometer duas vezes o mesmo erro. Se fracassar, ao menos que fracasse ousando grandes feitos, de modo que a sua postura não seja nunca a dessas almas frias e tímidas que não conhecem nem a vitória nem a derrota."

(Theodore Roosevelt)

RESUMO

Na área da computação nas últimas décadas o principal fenômeno a ser destacado é a difusão dos dispositivos móveis e os avanços da computação ubíqua, que propõe um cenário onde os diferentes tipos de serviços computacionais encontram-se disponíveis aos usuários a qualquer momento e em qualquer lugar. Esses serviços, também devem interagir no cotidiano das pessoas de forma onipresente, invisível e proativa. Com base nisso, diversas aplicações têm surgido aplicadas em diferentes áreas, como educação, saúde, transporte, acessibilidade. Nestas áreas pode ser constatado que a formação e gerenciamento de grupos é um desafio comum e possui grande importância como pode ser notado pelos esforços recentes e pesquisas relevantes sobre o assunto. Nessas pesquisas tem-se apontado que é importante considerar aspectos de contexto das entidades, mas também o histórico de contexto de determinada entidade. Neste cenário, esta dissertação propõe o eGroup, um modelo para o gerenciamento de grupos dinâmicos de entidades, que visa auxiliar aplicações ubíquas que almejem agrupar e gerenciar entidades levando em consideração características de perfis das entidades, o contexto em que elas estão inseridas, bem como histórico destes. A estratégia adotada neste trabalho se diferencia dos trabalhos relacionados por não trabalhar com um domínio específico, considerar todos os aspectos de contexto e o histórico de contexto, gerando de forma dinâmica um perfil e trilha do grupo. Com os resultados obtidos através de uma validação por cenários, foi possível verificar a viabilidade do modelo, além de propor uma solução para o problema de pesquisa.

Palavras-chave: Formação de Grupos Dinâmicos de Entidades. Gerenciamento de Grupos Dinâmicos. Aplicações Ubíquas para Gerenciamento de Grupos.

ABSTRACT

In recent decades in the area of computing the main phenomenon to be highlighted is the diffusion of mobile devices and advances in ubiquitous computing, which proposes a scenario where the different types of computer services are available to users anytime and anywhere. These services should also interact ubiquitously, invisible and proactive in daily life. Based on this, several applications have emerged applied in different areas such as education, health, transportation, accessibility. These can be seen that the formation and management groups is a common challenge and has great importance as can be seen by recent efforts and relevant research on the subject. In these studies we have pointed out that it is important to consider the context of aspects of the entities but also the determined entity context history. In this scenario, this dissertation proposes the eGroup, a model for managing dynamic groups of bodies which aims to assist ubiquitous applications that aim group and manage entities taking into account characteristics of the entities profiles, the context in which they operate, as well as historical thereof. The strategy adopted in this study differs from related work by not working with a particular domain, consider all aspects of the context and the context of history, generating dynamically a profile and group track. The results obtained through a validation scenarios, it was possible to verify the viability of the model, and to propose a solution to the problem of research.

Keywords: Forming Dynamic Groups Entities. Management Dynamic Groups. Applications Ubiquitous for Group Management.

LISTA DE FIGURAS

Figura 1	Taxonomia dos problemas para a Computação Ubíqua	22
Figura 2	Modelo de tarefas de grupos	26
Figura 3	Grupos através do perfil das entidades	27
Figura 4	Conceitos ubíquos do eGroup	27
Figura 5	Conceitos ubíquos do eGroup	28
Figura 6	Termos de busca para trabalhos relacionados	30
Figura 7	Critérios para seleção dos trabalhos relacionados	32
Figura 8	Visão geral	33
Figura 9	<i>Group Allocation Architecture</i>	36
Figura 10	Visão geral	38
Figura 11	Visão geral do <i>Didja</i>	39
Figura 12	<i>GCF High Level Architecture</i>	40
Figura 13	Arquitetura para contexto compartilhado	41
Figura 14	Arquitetura <i>Smart Chat Group</i>	42
Figura 15	Visão geral do sistema de grupos dinâmicos centrado no usuário . . .	43
Figura 16	Visão geral do eGroup	47
Figura 17	eGroup conectado a mais de um gerenciador de trilhas e várias apli- cações	48
Figura 18	Arquitetura do eGroup	49
Figura 19	Protocolo de Comunicação	52
Figura 20	eGroupID	53
Figura 21	Modelo de perfil das entidades	56
Figura 22	Exemplo de modelo de perfil das entidades utilizando <i>JSON</i>	57
Figura 23	Fluxo de dados para formar os grupos através de uma regra	58
Figura 24	Modelo de Regras Simples do eGroup	59
Figura 25	Modelo de Regras do Completo eGroup	60
Figura 26	Modelo de contexto das entidades	61
Figura 27	Modelo de contexto das entidades utilizando <i>JSON</i>	62
Figura 28	Modelo de trilha das entidades	63
Figura 29	Modelo de trilha das entidades utilizando <i>JSON</i>	64
Figura 30	Modelo de perfis dos grupos <i>JSON</i>	64
Figura 31	Modelo de trilha dos grupos <i>JSON</i>	65
Figura 32	Diagrama de Classes do eGroup	67
Figura 33	Diagrama de Sequência do eGroup	68
Figura 34	Telas da <i>aplicação web</i> para Efetivos do Exército sendo executadas no navegador	72
Figura 35	Regra para primeiro e segundo grupo	75
Figura 36	Telas da base de dados com id das entidades agrupadas	75
Figura 37	Tempo de resposta para geração dos grupos com aumento de entidades	78

LISTA DE TABELAS

Tabela 1	Comparação entre os trabalhos apresentados	45
----------	--	----

LISTA DE LISTAGENS

Código 1:JSON de entrada (START)	70
Código 2:JSON com os modelo de perfis das entidades	71
Código 3:JSON com os modelo de contexto das entidades	71
Código 4:JSON com os perfis das entidades	72
Código 5:JSON com o contexto das entidades	74
Código 6:JSON com o perfil do primeiro grupo	75
Código 7:JSON com o perfil do primeiro grupo	76
Código 8:JSON com a trilha do grupo gerado com base no primeiro grupo	76
Código 9:JSON com a trilha do grupo gerado com base no segundo grupo	77

LISTA DE SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
PIPCA	Programa Interdisciplinar de Pós-graduação em Computação Aplicada
CSCL	Computer Supported Collaborative Learning
AG	Algoritmo Genético
GCF	Group Context Framework
IHC	Interação Humano-Computador
UBICOMP	Computação Ubíqua
UML	Unified Modeling Language
UNISINOS	Universidade do Vale do Rio dos Sinos
JSON	JavaScript Object Notation
XML	eXtensible Markup Language
HTTP	Hypertext Transfer Protocol
URL	Uniform Resource Locator
RS	Rio Grande do Sul
MS	Milisegundos

SUMÁRIO

1: INTRODUÇÃO	16
1.1: Motivação	17
1.2: Questão de Pesquisa	19
1.3: Objetivos	19
1.4: Metodologia	20
1.5: Organização da Dissertação	20
2: FUNDAMENTAÇÃO TEÓRICA	21
2.1: Computação Ubíqua	21
2.2: Sensibilidade a Contexto	23
2.3: Perfis	24
2.4: Trilhas	24
2.5: Gerenciamento de Grupos	25
2.6: Considerações sobre o Capítulo	29
3: TRABALHOS RELACIONADOS	30
3.1: Metodologia para a escolha dos trabalhos relacionados	30
3.2: Bais et al. (2013)	32
3.3: Medina et al. (2013)	33
3.4: Boratto, Carta e Satta (2010)	34
3.5: Ani et al. (2010)	34
3.6: Webber e Lima (2012)	34
3.7: Kyprianidou et al. (2012)	35
3.8: Stalljohann e Hackeloer (2011)	35
3.9: Lowe et al. (2011)	35
3.10 Sancho-Asensio et al. (2014)	36
3.11 Mujkanovic et al. (2012)	37
3.12 Rana, Kristiansson e Synnes (2012)	37
3.13 Freitas e Dey (2015a)	38
3.14 Freitas e Dey (2015b)- <i>GCF Framework</i>	39
3.15 Julien (2011)	41
3.16 Felix e Tedesco (2008)	41
3.17 Hallberg et al. (2007)	42
3.18 Muehlenbrock (2005)	43
3.19 Coutand et al. (2005)	44
3.20 Considerações sobre os trabalhos relacionados	44
3.21 Considerações sobre o capítulo	46
4: MODELO PROPOSTO	47
4.1: Visão geral do eGroup	47
4.2: Arquitetura	49
4.3: Comunicação do eGroup	51
4.4: Perfis das Entidades	56
4.5: Modelo de Regras	56
4.6: Modelo de Contexto	59
4.7: Modelo de Trilha	61
4.8: Perfis e Trilhas dos grupos	61

4.9: Considerações sobre o capítulo	63
5: ASPECTOS DE IMPLEMENTAÇÃO E AVALIAÇÃO	66
5.1: Aspectos de Implementação	66
5.2: Aspectos de Avaliação	69
5.2.1: Cenário - Sistema para Gerenciamento de Efetivo de Exércitos	70
5.2.2: Avaliação do desempenho da criação de grupos	78
6: CONSIDERAÇÕES FINAIS	79
6.1: Conclusões	79
6.2: Trabalhos Futuros	80
REFERÊNCIAS	81
ANEXO A: PRODUÇÃO DE ARTIGOS	87

1 INTRODUÇÃO

Ao longo dos últimos anos, têm-se observado a popularização do acesso à internet, bem como a massificação dos dispositivos móveis que se encontram no mercado, principalmente, *notebooks*, *netbooks*, *smartphones* e *tablets*. Atualmente, estes dois últimos têm se tornado mais acessíveis financeiramente, devido a diversificação de modelos desenvolvidos por fabricantes, além de apresentar uma gama de recursos disponíveis, entre eles: câmera digital, sensores, *GPS*, acelerômetro, conexões Wi-Fi, *Bluetooth*, telas de alta definição, dentre outros.

Cabe ressaltar que além dessa massificação dos dispositivos móveis, os usuários interagem com grande variedade de sistemas computacionais, os quais integram-se de forma transparente às suas vidas e que de uma forma ou de outra auxiliam na resolução de suas mais diversas atividades diárias.

Esse cenário é propício para a viabilização do paradigma da Computação Ubíqua, que segundo Hansmann et al. (2003) é o acesso a informação na forma conveniente e fácil, através de uma nova classe de aplicações inteligentes, que acessam informações relevantes com a capacidade de tomar determinadas ações quando e onde for preciso, tudo isso sendo executado de forma transparente à vida das pessoas. A *Internet* impulsionou o avanço apresentado por este paradigma, pois possibilitou a conectividade e a integração destes usuários.

Segundo Weiser (1991), a Computação Ubíqua vislumbra toda essa expansão no uso de dispositivos móveis, para transformar a computação em uma tecnologia onipresente, transparente e invisível, ou seja, o principal intuito é fazer com que o usuário usufrua destas tecnologias presentes no seu cotidiano, sem que perceba que está utilizando-a. Já para Satyanarayanan (2001), o principal objetivo da Computação Ubíqua é oferecer aos usuários o acesso às informações de forma uniforme, considerando que grande parte dos dispositivos estarão conectados em qualquer lugar, assim como a informação estará presente em qualquer lugar e a qualquer hora.

Para que esses cenários funcionem plenamente, é preciso que essas aplicações considerem determinadas informações de contexto. Segundo Dey (2001), contexto é qualquer informação que pode ser usada para caracterizar a situação de uma determinada entidade, onde uma entidade pode ser uma pessoa, local ou objeto que seja considerado relevante para a interação entre usuário e aplicação.

Alguns trabalhos demonstram interesse em coletar e armazenar informações a respeito de entidades, tendo como objetivo conhecer suas preferências e quem realmente essa entidade representa, o que possibilita à aplicação personalizar seus serviços oferecidos. Ainda segundo Viviani, Bennani e Egyed-Zsigmond (2010), essas aplicações alocam tais informações a respeito das entidades em um perfil de entidade, os quais são armazenados em um formato de modelo de perfil específico.

As pesquisas que circundam esta área têm mostrado que além da aplicação conhecer o contexto atual de uma determinada entidade, é importante que a mesma armazene o histórico

de contextos visitados pela entidade. Os autores Silva et al. (2010) dão à esse histórico de contextos o nome de Trilha.

Com base na evolução destes cenários que envolvem à computação, diversas oportunidades têm surgido em diferentes áreas, como educação (BARBOSA et al., 2011), saúde (DAMASCENO VIANNA; BARBOSA, 2014), comércio (FRANCO et al., 2011), jogos (BUZETO et al., 2013), transportes (OH; OH, 2011), acessibilidade (TAVARES et al., 2012), e principalmente a formação, gestão e acompanhamento de grupos (LIN; HUANG; CHENG, 2010), (FERREIRA; BARBOSA; GLUZ, 2013) e (LAWRIE; MATTHEWS; GAHAN, 2010), entre tantas outras áreas que estes paradigmas Ubíquos se tornam aplicáveis.

1.1 Motivação

Atualmente, os dispositivos móveis acabaram tomando o cotidiano das pessoas, onde a proporção é justamente o contrário de décadas anteriores, abraçando um horizonte de vários computadores ou dispositivos para uma única pessoa. Estes aspectos expressam exatamente a visão proposta por Weiser (1991), que define a Computação Ubíqua como serviços computacionais que encontram-se disponíveis à qualquer momento e em qualquer lugar.

Por outro lado, os autores Weiser e Brown (1996) discutem os impactos futuros que o modelo atual de interação humano-computador possa vir a sofrer, principalmente no cotidiano dos usuários, uma vez que estes estão diariamente saturados de informações. Com isso, os autores apresentam o conceito da Tecnologia Calma, onde a interação humano-computador ocorre na periferia do usuário, ao invés de estar constantemente no centro de suas atenções, ou seja, as informações irão se deslocar ao usuário apenas no momento em que for necessário. Por fim, os autores Weiser e Brown (1996) definem a Tecnologia Calma como: “O que informa, mas não exige o nosso foco ou atenção”.

Já Satyanarayanan (2001) descreveu que para um sistema se tornar invisível, o mesmo precisa ser proativo, ou seja, é preciso saber qual a intenção do usuário para com o sistema. Para isso, é preciso conhecer informações como, quem é este usuário, o que ele busca, o que ele precisa realizar e através disso, definir como o sistema pode se adaptar à ele.

Para conhecer essas informações, é preciso descobrir o contexto em que determinada entidade se encontra em um dado momento e quais foram suas ações tomadas anteriormente. Essas informações sobre as entidades, podem ser buscadas através do perfil da entidade (VIVIANI; BENNANI; EGYED-ZSIGMOND, 2010), já as informações de contexto são encontradas na trilha da entidade (WAGNER; BARBOSA; BARBOSA, 2014).

Este trabalho, considera o conceito de entidade no lugar do clássico conceito de usuário. Enquanto o usuário normalmente é uma pessoa que utiliza a aplicação, uma entidade refere-se a qualquer objeto físico ou não (como um carro), ou ainda um local (como uma sala).

Embora existam trabalhos que considerem pelo menos uma informação que pode ser considerada o contexto de um usuário (localização, por exemplo), muitos não levam em consideração

características de perfil do usuário para exercer a formação de grupos. Ainda, é possível afirmar que a maior parte dos trabalhos não operam com o conceito de entidade, ou seja, os grupos são formados apenas pelo conceito clássico de usuário (pessoa). Um exemplo disso, é um grupo de estudantes que realiza determinada tarefa educacional, por outro lado, um exemplo de grupo de entidade é a formação de grupos de cavalos.

O trabalho proposto pelos autores Christodouloupoulos e Papanikolaou (2007), visa uma solução para contemplar novas formas de educação, treinamento e ensino. Esses ensinamentos deveriam ser efetivos, mas ocupar pouco tempo e extinguir o deslocamento por parte, tanto dos alunos quanto dos educadores. Nesse contexto está o Ensino Colaborativo Auxiliado por Computador (do inglês CSCL - *Computer Supported Collaborative Learning*) que é um paradigma onde pressupõe aos alunos o trabalho em grupo, possibilitando uma maior interação entre ambos e criando um ambiente onde uns ajudam os outros na resolução de tarefas, dúvidas e problemas.

Por outro lado, a maior parte dos trabalhos está direcionado para um único domínio de aplicação, entre os principais: educação (CHRISTODOULOPOULOS; PAPANIKOLAOU, 2007), (KYPRIANIDOU et al., 2012) e (WEBBER; LIMA, 2012), recomendação de objetos aprendizagem (FERREIRA; BARBOSA; GLUZ, 2013), (BORATTO; CARTA; SATTA, 2010) e saúde (HALLBERG et al., 2007).

No trabalho apresentado pelos autores Hallberg et al. (2007), é proposto um modelo de comunicação para formação e gestão de grupos dinâmicos, que baseado em informações de contexto consegue atribuir a comunicação entre os usuários. O domínio de aplicação deste trabalho é saúde, e a formação de grupos se dá entre usuários. Um dos cenários apresentados no artigo, é de uma família que tem a necessidade de exercer cuidados sobre uma senhora de idade e então, através do grupo formado, é possível que haja interação entre quem está com a idosa no atual momento e os demais que não estão, como os médicos.

Ainda dentro deste cenário, a formação de grupos se mostra como um mecanismo que pode contribuir e auxiliar também no desenvolvimento em ambientes de aprendizagem colaborativa. Os autores De Faria, Adán-Coello e Yamanaka (2006) descrevem e avaliam um abordagem que adota a formação de grupos de alunos, com base no estilo de programação de cada um. Eles destacam os resultados apresentados pelos experimentos, onde comprovam que a formação de grupos foi eficaz para a melhora dos estilos de programação para cada aluno.

Considerando estes cenários, muitos dos trabalhos envolvem apenas a formação de grupos (KYPRIANIDOU et al., 2012), (LOWE et al., 2011), ou então se preocupam com formação e gestão dos grupos (FREITAS; DEY, 2015a), (RANA; KRISTIANSOON; SYNNESE, 2012), ou exercem apenas a formação e acompanhamento dos mesmos (FELIX; TEDESCO, 2008). Desta forma, o problema identificado é como poderia ser um modelo que através dos conceitos Ubíquos efetue a formação, gestão e acompanhamento do grupo durante o seu tempo de vida.

A maioria dos trabalhos não leva em consideração informações de contexto segundo Dey, Abowd e Salber (2001) para a formação dos grupos, apenas consideram localização e/ou situ-

ação em que tal entidade se encontra em um dado momento para efetuar o agrupamento. Com base no que foi apresentado, não foram encontrados em pesquisas, trabalhos que armazenem um histórico de contexto do grupo, desde sua formação até o término. Para essas informações de contexto é dado o nome de trilha do grupo, considerando a definição de Silva et al. (2010). Também não foram encontrados trabalhos que mantenham armazenados um perfil dinâmico com as características do grupo, conforme entrada e saída de entidades.

Considerando a importância de se gerenciar grupos, este trabalho propõe o eGroup, que é um modelo para o gerenciamento de grupos dinâmicos de entidades.

1.2 Questão de Pesquisa

Como seria um modelo computacional para aplicação da Computação Ubíqua no gerenciamento de grupos de entidades ?

1.3 Objetivos

Esta dissertação tem como objetivo geral especificar, implementar e validar um modelo para o gerenciamento de grupos de entidades, denominado eGroup. A formação dos grupos é realizada através do recebimento de regras, com base em informações genéricas de perfil das entidades, podendo ainda utilizar informações de contexto e trilha das entidades.

O eGroup é um modelo genérico, ou seja, um modelo que permite como entrada qualquer informação de regra, perfil, contexto e trilha das entidades, desde que estejam no formato suportado pelo mesmo. Sendo assim, são objetivos específicos deste trabalho:

- Identificar os principais conceitos relacionados ao tema proposto;
- Especificar o modelo e arquitetura do eGroup;
- Desenvolver um protótipo para o modelo;
- Validar o modelo aplicando o protótipo através de um cenário.

Após a análise dos trabalhos relacionados, apresentados no capítulo 3, viabilizou-se algumas características importantes para implementação do eGroup. Entre elas:

- Geração de um histórico dos grupos (chamado de trilha dos grupos) de forma dinâmica;
- Geração de um perfil dos grupos de forma dinâmica;
- Uso de informações de perfil, contexto e trilha das entidades para formar os grupos;
- Domínio de aplicação genérico, podendo ser utilizado por diferentes aplicações;

1.4 Metodologia

Inicialmente buscou-se identificar quais conceitos seriam estudados, utilizados e descritos nesta dissertação, a fim de obter o embasamento teórico necessário para compreensão da visão geral do eGroup. Após ter identificado as tecnologias necessárias, efetuado a leitura e compreendido o que realmente seria necessário para a descrição desta dissertação, iniciou-se a escrita.

Na sequência, foram realizadas pesquisas com a finalidade de identificar possíveis trabalhos relacionados com o eGroup, e os modelos atuais para o gerenciamento de grupos, bem como, estudar as abordagens utilizadas e verificar quais resultados foram obtidos através delas. Logo após, foi realizado um comparativo entre os trabalhos relacionados, que foram adotados através do critério de seleção. Em seguida, foram apresentadas as características de cada trabalho em uma tabela comparativa.

Conhecendo aspectos importantes sobre cada trabalho, foi especificado a visão geral do eGroup, bem como sua arquitetura de *software*. O objetivo é apresentar os componentes necessários para o modelo e como eles interagem internamente e com as aplicações externas.

A próxima etapa do trabalho foi a implementação de um protótipo, a implementação teve como objetivo auxiliar etapa final do trabalho que é a validação do modelo.

Por fim, foi realizada a validação do modelo através de um cenário para aplicação do mesmo, estes cenários foram elaborados com o objetivo de explorar os ganhos trazidos pelo modelo, bem como sua eficácia frente ao problema de pesquisa apresentado.

1.5 Organização da Dissertação

O trabalho está organizado da seguinte forma. O Capítulo 2 descreve os principais conceitos referentes às áreas relacionadas a este trabalho. O Capítulo 3 descreve trabalhos relacionados na área de gerenciamento de grupos, apresentando os principais desafios da área. No Capítulo 4 é proposto o modelo do eGroup, bem como sua forma de operação. O Capítulo 5 apresenta os aspectos de implementação e um cenário para avaliação do modelo proposto. Por fim, o Capítulo 6 apresenta as considerações finais.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta uma breve descrição sobre os conceitos e tecnologias que são considerados relevantes para essa dissertação. Na primeira seção é apresentada uma visão geral sobre os avanços computacionais, principalmente da Computação Ubíqua. A seção seguinte aborda aspectos da sensibilidade ao contexto. Na seção 2.3 é realizado um apanhado geral sobre os perfis de entidades. Já a seção 2.4 detalha o conceito de trilhas. Por fim, na última seção é apresentado a junção dos conceitos da Computação Ubíqua com os de grupos, e descrito o termo gerenciamento de grupos.

2.1 Computação Ubíqua

O conceito de Computação Ubíqua (do inglês - *Ubiquitous Computing*), também chamado de *ubicomp*, foi introduzido em 1991, quando uma equipe do Laboratório de Ciência Computacional da *Xerox Palo Alto Research Center* (WEISER, 1991), liderado pelo cientista Mark Weiser, este considerado por muitos o pai da Computação Ubíqua (COSTA, 2009), responsável pelos avanços significativos dentro da área, identificaram em conjunto, uma série de tendências relacionadas à computação:

- O tamanho dos dispositivos passava por um período de transformação, onde eram cada vez menor;
- O custo do *hardware* também tornava menor, tendência que se expande até atualidade;
- A proporção de computadores e dispositivos por pessoas, estava se tornando cada vez maior.

Segundo Hansmann et al. (2003), os dispositivos móveis deveriam lidar com a escassez de energia e que suas aplicações deveriam ser capazes de retomar após um eventual desligamento. Ainda segundo Hansmann et al. (2003), todos esses dispositivos móveis em breve iriam superar os computadores pessoais, e estariam ligados em uma rede com um largo processamento de informações sobre cada entidade.

Já Araujo (2003) defende que esses dispositivos apresentam-se de forma invisível para o usuário, com a capacidade de adequar um determinado ambiente conforme informações obtidas à respeito do mesmo. A partir dessa interação, surge a possibilidade de computadores atuarem e agirem de forma inteligente.

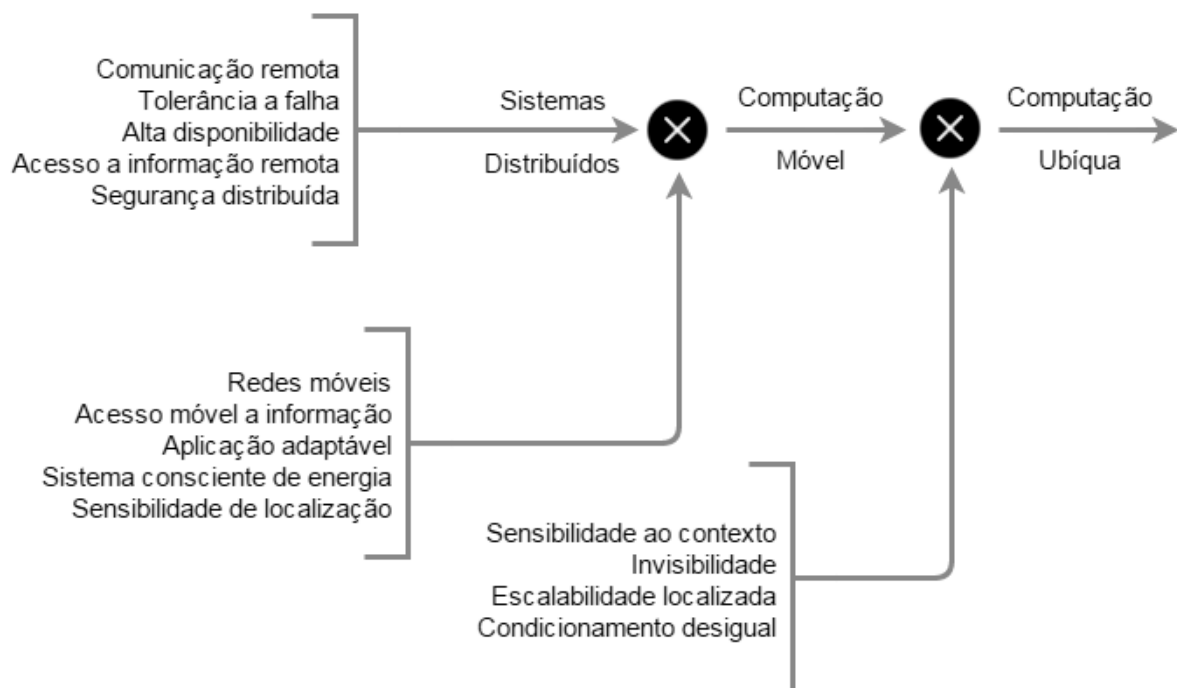
Em 1996, os autores Weiser e Brown (1996) concluíram que essas tendências realmente ocasionariam impacto no cotidiano das pessoas e que o aumento no número de dispositivos, e conseqüentemente a sobrecarga de informações gerada sobre os usuários, recairiam sobre a qualidade da interação entre humanos e computadores (IHC), alterando assim a qualidade de vida das pessoas.

A fim de concretizar e solidificar o conceito proposto, em 1991, Mark Weiser escreveu o artigo: “*O Computador para o século 21*” (WEISER, 1991). Cabe destacar, que quando Weiser escreveu esse pensamento visionário, os dispositivos computacionais ainda eram limitados, ou seja, prever que pouco tempo depois, os recursos tecnológicos iriam ser superados e fariam parte do cotidiano pessoal de cada um, foi sem dúvidas, uma enorme contribuição.

Naquele artigo, na primeira frase Weiser afirma: “as tecnologias mais profundas são aquelas que desaparecem. Elas se entrelaçam no tecido do dia-a-dia até que se tornem indistinguíveis dele”. É nesse contexto que se enquadra a *ubicomp*, onde vislumbra toda a expansão no uso de dispositivos móveis, a fim de transformar a computação, em uma tecnologia onipresente, transparente e invisível. O principal intuito é fazer com que o usuário usufrua destas tecnologias no seu cotidiano, sem que perceba que está utilizando-a.

Os autores Weiser e Brown (1996) apresentam um novo conceito, definido como Tecnologia Calma, essa tecnologia seria a solução para a sobrecarga de informações sobre o usuário. Esse conceito está vinculado à *ubicomp*, onde os autores concluem a Tecnologia Calma como sendo: “O que informa, mas não exige o nosso foco ou atenção”. Já os autores Costa, Yamin e Geyer (2008), apresentam uma arquitetura de *software* para suportar os principais desafios impostos pela *ubicomp*, a fim de contribuir com a comunidade desenvolvedora e para avaliar o *middleware* desenvolvido.

Figura 1: Taxonomia dos problemas para a Computação Ubíqua



Fonte: Adaptado de Satyanarayanan (2001)

Em 2001, Satyanarayanan (2001) publicou um artigo apresentando possíveis cenários para aplicação prática do conceito, além dos problemas que ainda precisavam ser resolvidos den-

tro da *ubicomp*. Para Satyanarayanan (2001), a computação móvel pode ser considerada uma extensão da área de sistemas distribuídos com a junção de características como redes móveis, aplicação adaptável e localização. Da mesma forma a Computação Ubíqua pode ser considerada uma extensão da computação móvel, uma vez que herdou algumas características da mesma, integrando com suas próprias características, como sensibilidade à contexto e invisibilidade. A Figura 1 mostra como se deu essa evolução.

Satyanarayanan (2001) afirma que para um sistema se tornar totalmente invisível, o mesmo precisa ser proativo, ou seja, é preciso saber qual a intenção do usuário para com o sistema. Para isso, é preciso conhecer informações do mesmo, como: quem é este usuário, o que ele busca, o que ele precisa realizar e através dessas coleta, definir como o sistema pode adaptar-se ao usuário.

2.2 Sensibilidade a Contexto

Uma das principais áreas de pesquisa integrada a *ubicomp*, é computação ciente de contexto (do inglês - *Context-Awareness*), conhecida também como: computação sensível ao contexto (do inglês - *Context-Aware Computing*). Essa é uma área de pesquisa recente dentro da *ubicomp*, onde é possível se ter várias aplicações, em diferentes contextos (LOKE, 2006).

Satyanarayanan (2001) descreveu que para um sistemas ser intrusivo, é necessário que seja capaz de reconhecer determinada situação do usuário, e através disso, adaptar-se à ele. Nesse processo, a informação é chamada de contexto e sistemas que são capazes de realizar essa adaptação, são chamados de sensíveis a contexto.

Já Schilit, Adams e Want (1994) definem contexto, como: quem você é, com quem você está e que recursos estão próximos de você. Desta forma, o contexto é mais do que apenas a localização do usuário, estão inclusos nele: conectividade de rede, qualidade da conexão e até mesmo a situação social, como as pessoas que estão ao redor.

Os autores Abowd et al. (1999) apresentam uma categorização sobre o que é contexto, onde definem que a identidade, localização, data e atividade são os principais tipos de contexto para caracterizar a situação de uma entidade. Estes tipos de contexto não só respondem às perguntas de quem, o que, quando e onde, mas também são importantes para outras informações de contexto. Por exemplo, dada a identidade de uma pessoa, podemos adquirir várias informações relacionadas a ela, tais como telefone, endereços, e-mail, data de nascimento, entre outras. Obtendo a localização de uma entidade é possível ainda determinar que outros objetos ou pessoas estão próximos da entidade e qual atividade ou situação da entidade.

Para o modelo eGroup, o contexto é uma informação relevante para agrupar entidades, pois através dele é possível se obter informações como: quem é a entidade, sua localização, situação e tempo. Entre tantas definições de contexto, a que se tornou referência é a realizada por Dey, Abowd e Salber (2001), onde o contexto pode ser:

“Qualquer informação que possa ser utilizada para caracterizar a situação de entidades (pessoa, lugar ou objeto) que sejam consideradas relevantes para interação entre um usuário e uma aplicação (incluindo o usuário e a aplicação)”.

Por extensão, Dey, Abowd e Salber (2001) definem que “um sistema é sensível a contexto se ele utiliza o contexto para oferecer informações relevantes e/ou serviços ao usuário, onde a relevância depende da tarefa do usuário”.

2.3 Perfis

Os autores Mulvenna, Anand e Büchner (2000) apresentam uma visão sobre personalização, onde a definem como a disponibilização de produtos, serviços e informações que são adaptadas individualmente. Ainda segundo os autores, dentro da personalização os programas aprendem normas, hábitos diários, bem como as preferências de um determinado usuário.

Já Silva et al. (2010) definem entidade como uma extensão do conceito de usuário, uma vez que, a definição atribuída a entidade, é de que pode ser qualquer entidade, uma pessoa, local ou objeto. Por exemplo, pode ser um carro, um dispositivo, um container, exatamente o contrário da definição de usuário, onde se refere a uma pessoa que está utilizando um aplicativo.

Nesse sentido, um perfil para o eGroup significa: um conjunto de informações relevantes a respeito de uma determinada entidade, e que são definidas a partir das decisões tomadas por aquela entidade analisada. Os perfis dessas entidades chegarão ao modelo através de qualquer aplicação, desde que contenha as informações necessárias, já os perfis do grupo, serão gerados automaticamente, armazenando características específicas de cada grupo formado.

2.4 Trilhas

O termo trilha tem sido utilizado por muitos trabalhos da área da *ubicomp*, uma vez que descreve e registra diversas funcionalidades. Os autores Driver e Clarke (2004) afirmam que para uma aplicação se adaptar ao usuário, é preciso que ela vá além do simples desejo de conhecer o contexto atual de determinado usuário, e torne-se sensível a contexto, ou seja, identifique também, o histórico de contextos visitados pelo mesmo. Para esse histórico de contexto, é atribuído o nome de trilha (DRIVER; CLARKE, 2008).

Alguns autores defendem que o conceito de trilha geralmente está vinculado à localização de um usuário ou de uma entidade (HIGHTOWER; BORRIELLO, 2001), mas ele é bem mais amplo, e inclui as atividades que foram realizadas por uma determinada entidade, os conteúdos que foram acessados, aplicações que foram utilizadas, tudo isso dentro de um determinado contexto, localização, situação e um certo período de tempo (SILVA et al., 2010).

Nesse contexto, o conceito de trilha foi adotado pelo eGroup como sendo um conjunto de informações sobre o histórico de contexto atual e também visitados por entidades, levando-se em consideração os recursos das entidades, como: dispositivos utilizados, localização, tempo,

situação e a própria entidade. Essas informações chegarão ao modelo através de qualquer aplicação, desde que contenha todas ou pelo menos uma destas informações descritas, já as trilhas dos grupos, serão geradas automaticamente, armazenando informações de histórico de contexto atuais e visitados dos grupos.

2.5 Gerenciamento de Grupos

Considerando as situações apresentadas nas seções anteriores, onde é possível utilizar as atuais Tecnologias de Informação e Comunicação (TIC) para a formação, gestão e acompanhamento de grupos, a fim de propiciar uma melhor interação, cuidado, segurança, desempenho e monitoramento destes grupos.

O autor Forsyth (2006) define grupo como sendo a união de duas ou mais pessoas que são ligados uns aos outros através de suas relações sociais, uma vez que estes membros, estão ligados entre si, através de processos interpessoais, como: comunicação, influência, identificação, semelhança, relacionamento interpessoal, entre outros. O autor ainda cita que os grupos na sua grande maioria, são formados em busca de um objetivo específico, definido através de uma série de tarefas internas.

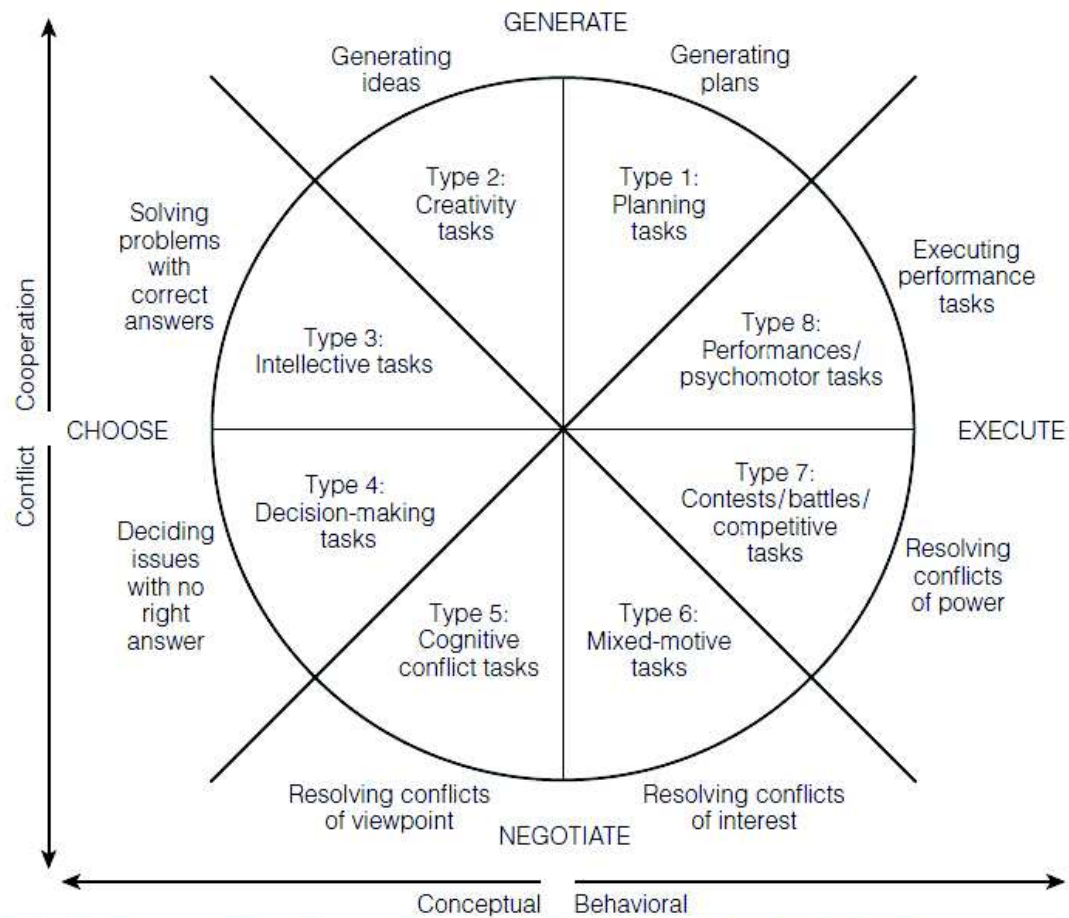
Ainda segundo Forsyth (2006), geralmente existe uma razão para a formação de grupos, pode-se considerar, a união de pensamentos, problemas, objetivos, entre outros aspectos para que membros se unam. Conforme a Figura 2, o autor apresenta um modelo de tarefas para grupos, onde são ilustrados os quatro objetivos básicos de grupos: gerar, escolher, executar e negociar. Na Figura 2 pode também ser visualizado uma subdivisão das categorias citadas, demonstrando uma possível extensão do modelo.

A interatividade dentro dos grupos, é extremamente necessária para o bom funcionamento dos mesmos, que podem ser formados e aplicados para uma área específica, como por exemplo: educação (OUNNAS; DAVIS; MILLARD, 2009). Alguns pesquisadores afirmam que o trabalho e o relacionamento em grupo é extremamente importante para o desenvolvimento da aprendizagem (SILVEIRA; BARONE, 2006).

Os autores Cunha, Fuks e Lucena (2002) destacam que o uso de tecnologias da informação valoriza e potencializa o trabalho em grupo, e também disponibilizam um sistema para auxílio à formação de grupos em um ambiente de aprendizagem *web*.

Por outro lado, os autores Backstrom et al. (2006) vão além de realizar apenas à formação de grupos, e se comprometem também em acompanhá-los, onde monitoram frequentemente: a participação de cada membro para com o grupo, a evolução do grupo e o crescimento do mesmo. Já os autores Cudak e Taib (2007), apresentam uma aplicação para a formação e gestão de grupos. Essa aplicação, tem como principal intuito, proporcionar ao administrador do grupo, a gerência sobre o mesmo.

Figura 2: Modelo de tarefas de grupos



Fonte: (FORSYTH, 2006)

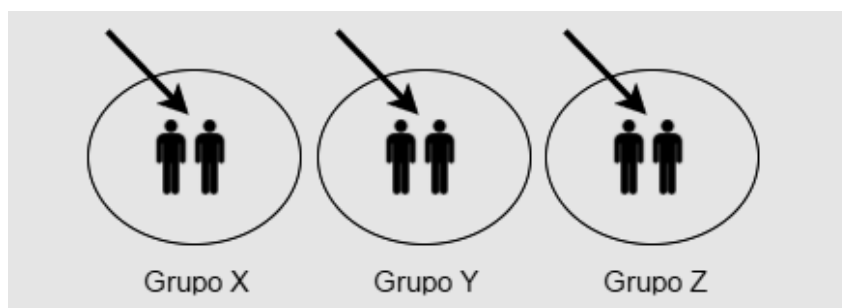
Finalmente, neste trabalho o termo gerenciamento de grupos é a integração destes três fatores, descritos e exemplificados acima, são eles:

- Formação de grupos;
- Acompanhamento de grupos;
- Gestão dos grupos.

Portanto para o eGroup, o gerenciamento de grupos envolve o agrupamento de duas ou mais entidades, onde entidade pode ser caracterizada como sendo: uma pessoa, local ou objeto que seja considerado relevante para aplicação, já que o eGroup pode considerar informações de perfil, contexto e histórico dos contextos visitados pelas entidades.

As Figuras apresentadas a seguir, esboçam uma visão geral sobre a junção dos conceitos da Computação Ubíqua com o gerenciamento de grupos. Na Figura 3, é possível destacar que existem três círculos e cada círculo é considerado um grupo, portanto temos três grupos formados: X, Y e Z.

Figura 3: Grupos através do perfil das entidades

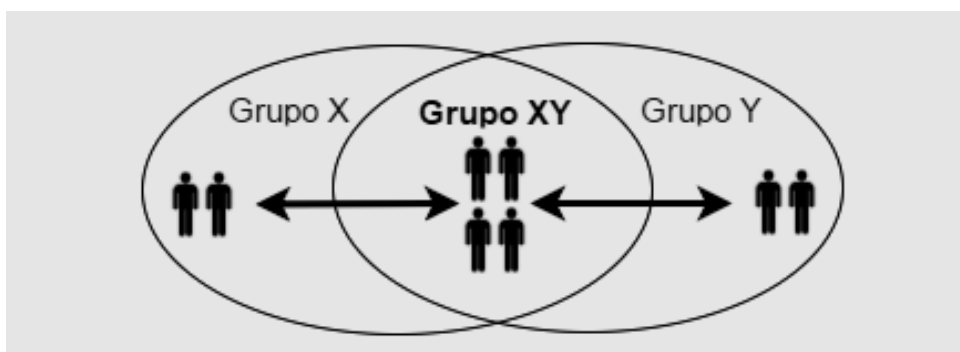


Fonte: Elaborado pelo autor

Cada um dos grupos possui duas entidades agrupadas, já que o eGroup considera o termo gerenciamento de grupos como sendo o agrupamento de duas ou mais entidades, mas poderia ser várias entidades. A entidade neste caso é uma pessoa. As entidades podem ser agrupadas através de regras que são enviadas pelas aplicações para o eGroup, e essas regras podem ser formadas com base em informações de perfil, contexto e/ou trilha das entidades.

A Figura 4, complementa o exemplo atribuído nesta seção, onde temos o grupo X e Y formados através de duas regras, uma para o grupo X e outra para o Y, com base em informações de perfil das entidades. Desta forma, é possível que uma nova regra agrupe entidades que já estão agrupadas em outros grupos, como é o caso da Figura 4.

Figura 4: Conceitos ubíquos do eGroup



Fonte: Elaborado pelo autor

Na Figura 4, é possível visualizar dois grupos formados através de duas regras distintas, com uma terceira regra é possível formar o grupo XY, que é a junção do grupo X com Y. Essa junção, nos permite agrupar entidades de forma mais específica, através de regras específicas.

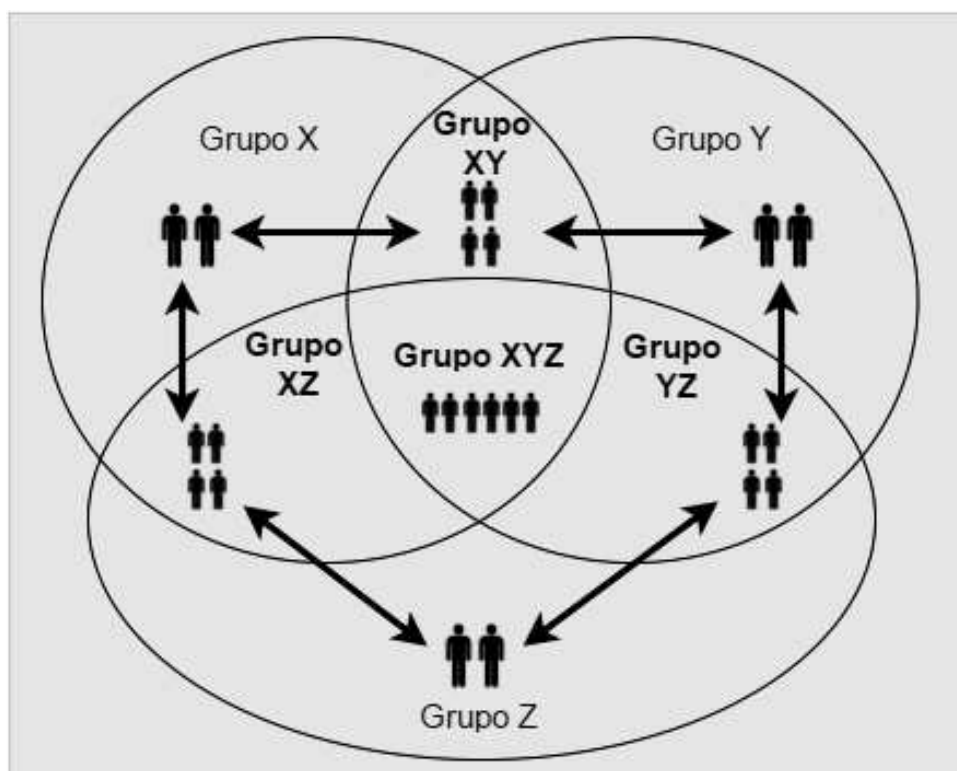
Um exemplo clássico para atribuir essa formação, é a utilização dos conceitos da Computação Ubíqua (DEY; ABOWD; SALBER, 2001). A disponibilização de informações de contexto das entidades permitem que o eGroup agrupe tais entidades através de sua localização (por

exemplo), ou seja, uma regra poderia ser: todas as entidades que se encontram na Unisinos, na sala do MobiLab¹, que possuem no seu perfil idade entre 20 a 30 anos.

Cada entidade possui um perfil com suas características e pode ainda ter uma trilha que representa o histórico de contexto visitado por tal entidade, isso possibilita que uma regra agrupe entidades com base em informações passadas, através deste histórico.

A Figura 5 ilustra o exemplo completo, onde é possível destacar que caso as entidades alterem suas informações de contexto, por exemplo: as entidades que se encontram na Unisinos, mas que saíram da sala do MobiLab. A localização *indoor* neste caso foi alterada, consequentemente subentende-se que estas entidades alteraram seus contextos, mas que ainda apresentam alguma característica de perfil em comum, portanto podem fazer parte ou não do grupo dependendo da regra aplicada para tal agrupamento.

Figura 5: Conceitos ubíquos do eGroup



Fonte: Elaborado pelo autor

Na Figura 5, é possível visualizar inicialmente três grupos formados através de regras distintas, no caso X, Y e Z. A partir disso, podemos perceber que com há uma intersecção destes, que somado a novas regras possibilitam a formação de novos grupos que contenham entidades que já estão agrupadas em outros, como é o caso do XY, XZ e YZ. Dando a ideia de que temos um grupo dentro de outro grupo. Por fim, ao centro da Figura 5 com base em uma regra bem mais específica, é possível a junção dos três grupos com seis entidades.

¹MobiLab - Laboratório de Computação Móvel e Ubíqua da Unisinos

2.6 Considerações sobre o Capítulo

Neste capítulo foram apresentados os principais conceitos envolvidos com a Computação Ubíqua, bem como: sensibilidade a contexto, perfis, trilhas e o termo gerenciamento de grupos. Com a definição destes conceitos foi possível identificar que a Computação Ubíqua nos permite acompanhar os passos e ações de uma determinada entidade, o eGroup propõe uma evolução, pois apresenta um modelo que viabiliza à aplicação solicitante o acompanhamento do perfil, contexto e trilha dos grupos formados por suas entidades. Através da sensibilidade a contexto será possível compreender o ambiente das entidades, bem como do grupo em geral. Através dos perfis será possível armazenar e disponibilizar de forma dinâmica, diversas informações em um perfil do grupo. Através das trilhas será possível analisar e armazenar os históricos de contextos dos grupos. Por fim, foi descrito o termo gerenciamento de grupos, definido nesta dissertação como a união de três fatores, que sejam, formação, acompanhamento e gestão de grupos.

O próximo capítulo apresenta os trabalhos relacionados ao gerenciamento de grupos.

3 TRABALHOS RELACIONADOS

Neste capítulo são apresentados os trabalhos mais relevantes que foram encontrados nas áreas de pesquisa relacionadas a esta proposta. Este capítulo está dividido em 20 seções, a primeira seção do capítulo apresenta a metodologia definida para as pesquisas, bem como os critérios adotados para a seleção dos trabalhos relacionados. As seções 3.2 até 3.19 apresentam uma descrição resumida dos trabalhos selecionados. Finalmente, na seção 3.20 é apresentado uma análise comparativa entre os trabalhos.

3.1 Metodologia para a escolha dos trabalhos relacionados

A pesquisa foi iniciada através da definição das palavras-chaves, utilizadas para realizar a busca de artigos, são elas: *Groups* (Grupos), *Groups Formation* (Formação de Grupos), *Management Groups* (Gerenciamento de Grupos), *Groups Formation Context* (Formação de Grupos Contexto). Uma vez definidos estes 4 termos de busca, foi gerado uma *string* para pesquisa em qualquer parte do artigo de algum destes termos ou possíveis outras palavras similares. O termo de busca pode ser visto na Figura 6.

Figura 6: Termos de busca para trabalhos relacionados

Termos de Busca: groups AND ("groups dynamics" OR formation OR forming OR create OR creating OR "management groups") AND groups formation ("groups dynamics" OR "groups dynamics" OR formation OR forming OR create OR creating OR "management groups") AND Management Groups ("management groups" OR "groups dynamics" OR formation OR forming OR create OR creating) AND Groups Formation Context ("formation groups" OR "groups dynamics" OR formation OR forming OR create OR creating OR "management groups").

Fonte: Elaborado pelo autor

Para realizar a busca dos trabalhos relacionados, foram utilizados oito repositórios digitais, são eles, *ACM Digital Library*¹, *IEEE*², *Springer*³, *Google Acadêmico*⁴, *CiteSeer*⁵, *Web Of Science*⁶, *Science Direct*⁷ e *DBLP*⁸. A opção de buscar artigos através desta lista de repositórios digitais, justifica-se pela confiabilidade em que estes repositórios apresentam e também porque

¹<http://dl.acm.org/dl.cfm>

²<http://ieeexplore.ieee.org/Xplore/home.jsp?reload=true&tag=1>

³<http://link.springer.com/>

⁴<https://scholar.google.com.br/>

⁵<http://citeseer.ist.psu.edu/index;jsessionid=A5EA12DC639F83C29FBA1FE54D3C11E1>

⁶http://apps.webofknowledge.com/UA_GeneralSearch_input.do?product=UA&search_mode=GeneralSearch&SID=1BvEmidARdYxFHgchvR&preferencesSaved=

⁷<http://www.sciencedirect.com/>

⁸<http://www.dblp.org/search/index.php>

os artigos publicados em revistas reconhecidas cientificamente podem ser facilmente encontrados por estes. Além de verificar a existência ou não de trabalhos que abordassem este modelo proposto.

Para verificar a existência de trabalhos que circundam o tema proposto, foram realizadas buscas entre o período de Agosto de 2014 até Setembro de 2015, utilizando todos os termos definidos, ou então, envolvendo apenas partes dos termos definidos, obtendo assim uma busca mais ampla e abrangente. Para essa busca foram utilizados os oito repositórios listados acima, foram analisadas as 14 primeiras páginas dos resultados do Google Acadêmico, e assim, exaustivamente com o restante dos repositórios, tendo como principal base de dados o *ACM Digital Library*, uma vez que, a partir da décima quarta página os artigos deixaram de possuir uma forte relação com o tema de pesquisa. Em uma primeira etapa, foram encontrados 70 artigos por meio da leitura dos títulos e *abstracts*. Entre este montante de artigos, foram adotados critérios para seleção dos trabalhos relacionados, os quais são apresentados na sequência.

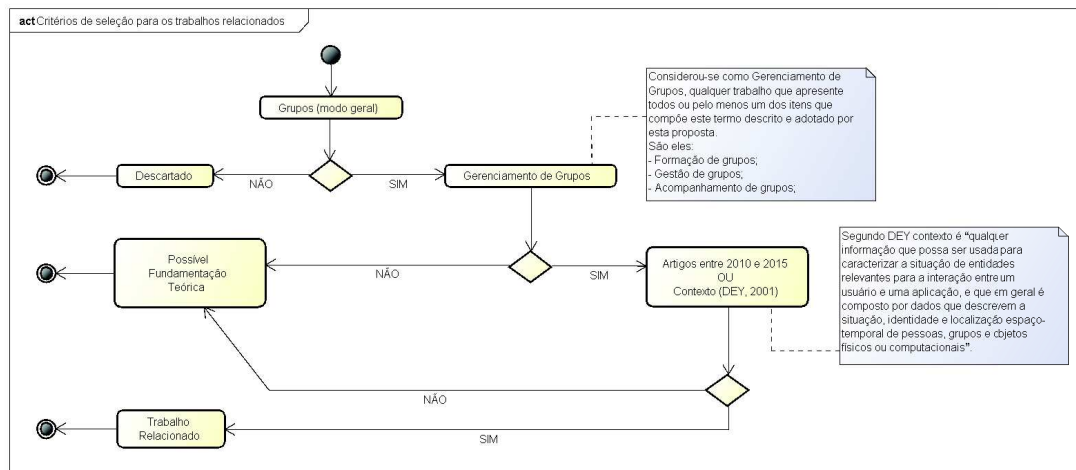
Para iniciar o processo de aplicação dos critérios de seleção adotados, levou-se em consideração a abordagem definida no UbiGroup Ferreira, Barbosa e Gluz (2013), em que foi proposto um modelo para recomendação ubíqua de conteúdo educacional para grupos de aprendizes, auxiliando o professor durante todo o processo de busca e seleção de materiais educacionais considerando os perfis dos alunos e o histórico de contextos em que estes alunos estão inseridos. A partir deste trabalho e dos termos de busca, foram pesquisados artigos que abordassem grupos de aprendizes ou qualquer trabalho que envolvesse grupos, nas suas mais diversas utilidades. Esta pesquisa retornou o maior número de possíveis artigos relacionados, dos 70 artigos encontrados, foram considerados pelo primeiro critério de seleção (Grupos) 53 trabalhos que realmente envolvem grupos. Os outros 17 artigos, foram descartados.

Ainda baseando-se na proposta de Ferreira, Barbosa e Gluz (2013), imaginou-se a criação de um processo que envolvesse a formação, gestão e acompanhamento destes grupos, tema este que foi definido e caracterizado no capítulo 2, seção 2.5, como gerenciamento de grupos. Nesta etapa de seleção, foi considerado como trabalho relacionado, qualquer artigo que apresentasse todos ou pelo menos um destes termos inclusos na definição de gerenciamento de grupos para esta dissertação. Nesta seleção, foram analisados 53 artigos e destes, apenas 31 artigos apresentaram pelo menos um dos conceitos inclusos na definição de gerenciamento de grupos. Os outros 22 artigos, foram caracterizados como material de apoio a fundamentação teórica, uma vez que servem como referência de estudo para algum assunto específico desta proposta.

Foi considerado como critério de seleção, artigos publicados entre 2010 e 2015 ou então, que envolvessem o termo gerenciamento de grupos com contexto, segundo a definição de Dey, Abowd e Salber (2001), de que contexto é “qualquer informação que possa ser usada para caracterizar a situação de entidades relevantes para a interação entre um usuário e uma aplicação, e que em geral é composto por dados que descrevem a situação, identidade e localização espaço-temporal de pessoas, grupos e objetos físicos ou computacionais”. Nesse critério de seleção, foram analisados 31 artigos e destes, foram considerados como trabalhos relacionados 10 arti-

gos sem contexto e 8 artigos com contexto, totalizando 18 artigos. Os outros 13 artigos, que se encontraram fora deste cenário, foram considerados como possível material de apoio. Na Figura 7 é apresentado os critérios que foram adotados para selecionar os trabalhos relacionados com o modelo proposto.

Figura 7: Critérios para seleção dos trabalhos relacionados



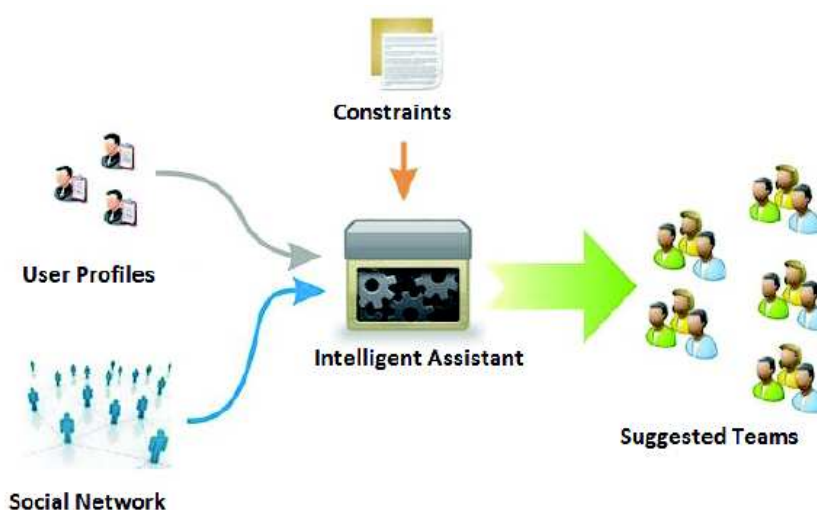
Fonte: Elaborado pelo autor

3.2 Bais et al. (2013)

No artigo apresentado por Bais et al. (2013), os autores defendem que a formação de grupos é um aspecto fundamental na aprendizagem colaborativa assistida por computador, uma vez que as diferentes características dos estudantes pode influenciar o desempenho do grupo. O assistente proposto e desenvolvido neste trabalho auxilia na formação de grupos, com base em satisfações e restrições ponderadas, o modelo apresentado considera três características individuais dos alunos para formar os grupos: estilos psicológicos, regras ou funções da equipe e redes sociais.

Na Figura 8 é apresentada a visão geral do modelo proposto por Bais et al. (2013), que é capaz de combinar restrições e preferências para indivíduos e grupos, é possível destacar que esse assistente pode ajudar os professores a formar grupos de alunos ou afins, sempre combinando as três características individuais. Um perfil (estudante) nesse caso, consiste em estilos psicológicos, funções e regras de tais grupos. Mas além disso, o modelo também obtém informações sobre os usuários ou sobre determinado perfil, através das redes sociais, que também é usada para considerar a relação que há entre o perfil analisado, o grupo e/ou então com os outros perfis (estudantes).

Figura 8: Visão geral



Fonte: (BAIS et al., 2013)

Com isso, os autores propuseram uma solução baseada em problemas de restrição de satisfação (*CSP*), em particular sobre problemas de satisfação restrição ponderadas (*WCSP*). Um exemplo de restrição é que todas as funções de um grupo devem estar presentes em um grupo. Os perfis de usuário, as redes sociais e as restrições são as entradas do assistente inteligente. As saídas são os grupos sugeridos para essas entradas.

3.3 Medina et al. (2013)

Medina et al. (2013) propõem um método que permite a formação de grupos de alunos através de indicadores que analisam as tarefas realizadas anteriormente pelos alunos. Nos dias atuais, a grande maioria dos estudantes participam e realizam atividades de Aprendizagem Colaborativa, que consiste na realização de tarefas acadêmicas em grupos. Dentro deste contexto, existem ferramentas que podem auxiliar a realização destas atividades, como por exemplo a Aprendizagem Colaborativa Assistida por Computador (*CSCL*) que oferece apoio para estas atividades coletivas através do uso computador.

Esse método para a formação dos grupos de alunos apresentado pelos autores, torna-se um elemento fundamental para alcançar uma colaboração e interação entre as entidades que serão agrupadas. Nele é possível que o professor por exemplo, possa definir se os alunos do mesmo grupo devem ter valores semelhantes ou diferentes em determinados indicadores. A formação dos grupos pode ser tratada através de duas abordagens, a primeira é heterogênea, onde os alunos de um grupo podem possuir uma heterogeneidade em algumas habilidades ou atitudes. O segundo critério é a homogênea, que pode ser aplicado simultaneamente, onde os alunos registram valores semelhantes em outros indicadores. Com isso, como estudo de caso e

possibilidade aplicação do método, os autores Medina et al. (2013) desenvolveram e validaram um software para formar grupos de alunos que devem resolver algumas tarefas de programação.

3.4 Boratto, Carta e Satta (2010)

Os autores Boratto, Carta e Satta (2010) apresentam um novo algoritmo capaz de agrupar usuários com recomendações de conteúdos individuais. Com isso, possibilita a identificação de grupos de usuários que apresentam alguma semelhança, através da comparação do perfil de um com o outro e conseqüentemente melhorar o desempenho e qualidade de tais recomendações dentro do grupo através daqueles sistemas que conseguem identificar e agrupar automaticamente os grupos. Este algoritmo utiliza recomendações individuais e também um algoritmo de agrupamento clássico para identificar e agrupar indivíduos.

3.5 Ani et al. (2010)

Os autores Ani et al. (2010) apresentam um método para a formação de grupos usando o algoritmo genético (AG), que são algoritmos muito utilizados para a solução de problemas de otimização e busca. Uma das justificativas apresentadas pelos autores, é que o trabalho em grupo pode auxiliar e garantir que as entregas dos produtos (*softwares*) sejam realizada no prazo correto e com qualidade.

No entanto, devido à falta de habilidades de programação especial em linguagem de programação Java, a maioria dos projetos de software dos alunos não pode ser entregue com êxito. Para resolver este problema, a formação de grupos sistemática é um dos fatores iniciais que devem ser considerados para garantir que cada grupo é constituído por indivíduos que são bons na programação.

3.6 Webber e Lima (2012)

O trabalho apresentado por Webber e Lima (2012) destaca a importância que os trabalhos em grupos têm para os professores, uma vez que com o agrupamento torna-se possível criar situações de aprendizagem colaborativa. A contribuição atribuída com esse trabalho consistiu no estudo e aplicação de experimentos, a fim de comprovar que a formação de grupos automáticos poderiam ser uma solução na aplicação de atividades de aprendizagem.

Em um dos cenários aplicados, foi preparado uma experiência com alunos do curso de graduação em programação, onde estes alunos foram em um primeiro momento, convidados a realizar atividades individualmente, para que em um momento futuro fossem formadas duplas automaticamente, conforme o desempenho de cada aluno.

Foram observados 20 alunos durante um certo período de tempo, os autores utilizaram uma escala para identificar qual era o nível de conhecimento de cada aluno: 1) muito fraco; 2)

fraco; 3) intermediário e 4) forte. O experimento foi submetido ao algoritmo que formou quatro grupos, representados por quatro categorias, criadas através da semelhança de perfil, conforme a análise do perfil de cada estudante, ou seja, foi comparado o desempenho que determinado aluno teve na resolução de tarefas individuais e depois colaborativas.

3.7 Kyprianidou et al. (2012)

Os autores Kyprianidou et al. (2012) apresentam um trabalho que explora a formação de grupos de alunos com base no estilo de aprendizagem que cada aluno possui, estes alunos são liderados respectivamente por seus professores. Para validação do trabalho, em um primeiro momento foi desenvolvido um sistema web, chamado *Pegasus*, este sistema é quem identifica os estilos de aprendizagem que cada perfil de aluno possui, para depois distribuí-los aos grupos. Após a coleta dos dados, foi possível perceber que os alunos estavam dispostos a colaborar uns com os outros, dentro dos seus respectivos grupos.

3.8 Stalljohann e Hackeloer (2011)

No trabalho apresentado por Stalljohann e Hackeloer (2011) é proposto uma ferramenta para o gerenciamento de grupos baseado em um conjunto de regras, essa ferramenta serve de apoio para os processos organizacionais que são voltados para as atividades de aprendizagem colaborativa no ensino superior, com isso, a formação de grupos torna-se um requisito básico para tais atividades colaborativas.

Os autores desenvolveram uma ferramenta para apoiar o agrupamento de estudantes baseado em regras definidas por um módulo que encapsula os grupos de estudantes como um serviço para diferentes ferramentas de atividades colaborativas, ou seja, pode ser utilizado por diferentes ferramentas de colaboração, mas dentro de uma única plataforma.

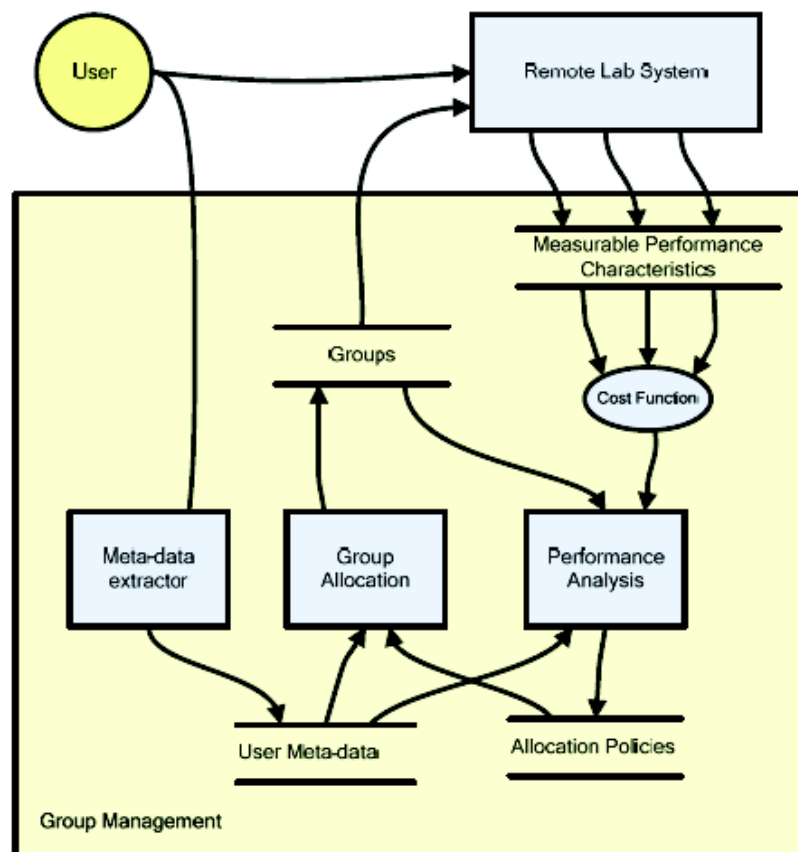
3.9 Lowe et al. (2011)

Os autores Lowe et al. (2011) propõem uma arquitetura para a formação de grupos dentro de laboratórios remotos, utilizando uma abordagem para a formação de grupos de forma automática, analisando frequentemente o desempenho do grupo, possibilitando a utilização de regras para validar qual a melhor composição para tal grupo. Essas regras foram utilizadas posteriormente para alocar estudantes a grupos que tem um melhor desempenho, no que diz respeito a aprendizagem.

Na Figura 9 é ilustrada a arquitetura proposta pelos autores, que destacam como principal mudança a inclusão de um mecanismo para análise de desempenho do estudante, uma vez que com esta arquitetura é possível medir o desempenho dos estudantes de forma constante a partir do sistema do laboratório remoto. Cabe destacar também, que essas medidas podem limitar

o funcionamento do grupo em geral, por exemplo: nível em que determinado estudante se encontra, diálogo dele para com o grupo, tempo que determinado estudante leva para concluir uma determinada tarefa, entre outras tantas medidas que podem aqui ser aplicadas, validando e comprovando a total aplicação do modelo proposto.

Figura 9: *Group Allocation Architecture*



Fonte: (LOWE et al., 2011)

3.10 Sancho-Asensio et al. (2014)

Os autores Sancho-Asensio et al. (2014) apresentam uma ferramenta para apoiar a formação de grupos de trabalho em ambientes de Aprendizagem Colaborativa, tendo em vista o processo de agrupamento de alunos por parte do corpo docente, uma vez que os professores analisam o perfil individual de cada aluno, com base em tais informações presentes no perfil, cabe ao professor distribuí-lo para um determinado grupo. É justamente neste aspecto que esse trabalho auxilia, uma vez que permite a geração destes grupos de forma automática.

3.11 Mujkanovic et al. (2012)

Os autores Mujkanovic et al. (2012) apresentam um algoritmo para efetuar a formação de grupos de indivíduos, que se adaptam a várias regras para essa formação, podendo uma métrica de desempenho específica ser otimizada, por exemplo, as habilidades, conhecimentos, raciocínio de um usuário pode ser afetado pelas características que determinado grupo possui.

A aplicação desenvolvida, utiliza a regressão linear múltipla, isso possibilita atualizar de forma adaptativa as regras que são definidas e utilizadas para a formação destes grupos. Uma implementação desta abordagem, comprovou a validação e aplicação do algoritmo proposto, onde vários estudantes realizaram experiências em um laboratório, foi possível perceber que o comportamento de vários estudantes foram alterados, agregando uma maior interatividade deste com o grupo e na própria análise de resultados das experiência realizada.

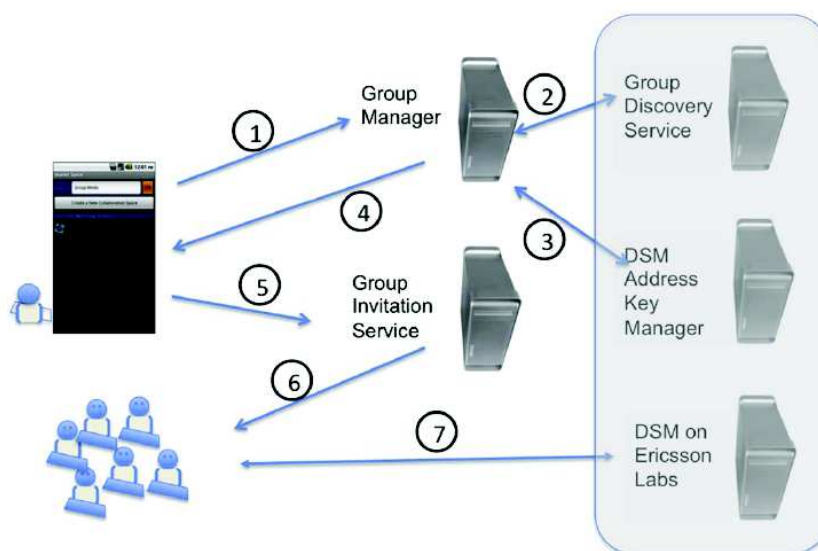
3.12 Rana, Kristiansson e Synnes (2012)

Os autores Rana, Kristiansson e Synnes (2012) apresentam um modelo que oferece a gestão e formação de grupos que se comunicam eletronicamente, desde que estes perfis estejam em um ambiente compartilhado, uma vez que, é levado em consideração aspectos como o contexto social em que tal usuário/perfil se encontra e também as suas relações sociais. Para formação, gestão e interação dentro do grupo, os autores levam em consideração a força relacional que um perfil têm para com os demais integrantes, esse cálculo é feito através de várias informações sobre a forma como esses perfis se comunicam, principalmente através de redes móveis.

A fim de reconhecer e identificar os parâmetros contextuais para gestão do grupo através de contexto, os autores levam em consideração itens de extrema relevância, tais como tags, locais e objetos são usados para gerar um gráfico social agregado, a fim de formar automaticamente um grupo. Na Figura 10 os autores detalham o serviço de integração modelo/aplicação.

A partir da interface do dispositivo do usuário, o mesmo fornece ao *Group Manager* (Gerenciador de Grupos) (1) pedidos para formar grupos, através de palavras-chave. O *Group Manager* acessa serviços em nuvem e processa as solicitações do usuário. O *Group Discovery Service* fornece listas de contatos recomendadas (2) e também recebe a chave de acesso do grupo de *DSM Key Manager* (3) para o grupo requerido. Logo em seguida, o *Group Manager*, acessa registros de chamadas, localização e listas de contato para refinar e priorizar a lista de participantes. Depois que o gerente do grupo envia a lista de participantes recomendada para a interface do usuário (4). O usuário pode então rever e modificar essa lista antes de enviar a lista final de participantes no *Group Invitation Service* (5). Se o usuário não quiser rever a lista de contatos recomendados, a lista poderá ser enviada automaticamente e diretamente ao *Group Invitation Service*. Este então distribui o convite a cada participante utilizando diferentes ferramentas de comunicação como e-mail, SMS, contato pessoal ou *tweets* (6). A mensagem de convite contém o ponto de acesso e espaço compartilhado dos recursos para este ambiente

Figura 10: Visão geral



Fonte: (RANA; KRISTIANSSON; SYNNESE, 2012)

colaborativo recém-formado. Finalmente, os participantes são capazes de acessar o espaço compartilhado, por meio do serviço *DSM on Ericsson Labs*.

3.13 Freitas e Dey (2015a)

Os autores Freitas e Dey (2015a) apresentam uma técnica que permite dispositivos móveis possam se comunicar um com o outro, a fim de formar grupos, para essa formação é levado em consideração o contexto em que tal dispositivo se encontra, principalmente a sua posição, localização e situação. A unificação destes dispositivos, possibilita aos usuários a troca de informações de forma espontânea, além da interatividade entre ambos. Um aspecto relevante deste trabalho é a utilização de múltiplos contextos para formar grupos, com isso é possível melhorar a detecção e formação dos grupos.

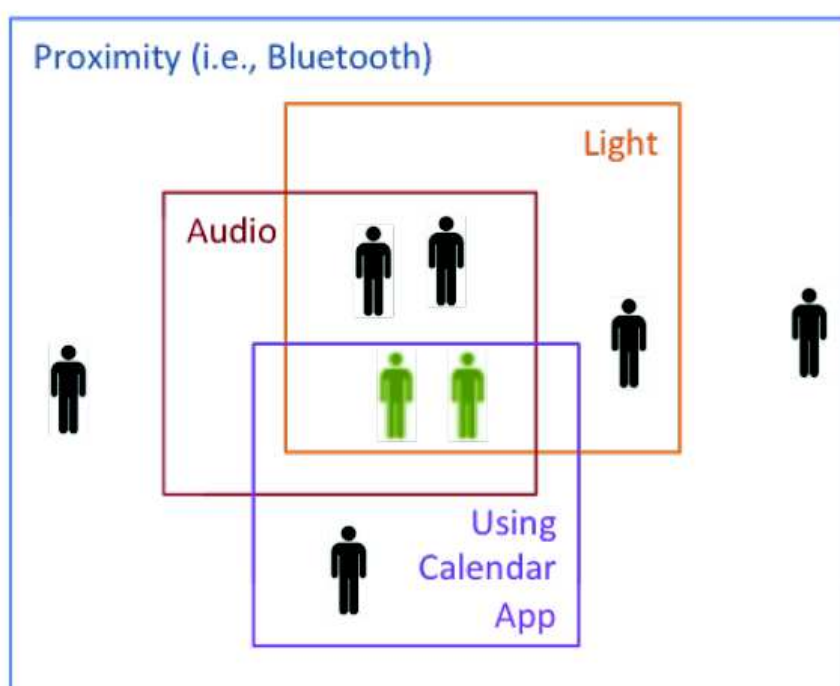
Os autores apresentaram o *Didja*, que é um kit de ferramentas de software desenvolvido especificamente para coleta e análise de múltiplas informações contextuais com a finalidade de encontrar dispositivos e usuários a fim de formar grupos de forma automática.

O *Didja* foi aplicado em duas soluções com a finalidade de validar o modelo e o kit proposto, o ferramental permite que aos dispositivos móveis reunir múltiplos fluxos de contextos e analisá-los praticamente em tempo real e quando estes contextos são semelhantes, são executadas duas funções: Primeiro, a ferramenta detecta e compartilha informações de contexto entre os dispositivos através de uma rede local, cada dispositivo manda uma mensagem com o contexto em que deseja se enquadrar. Em seguida, dispositivos próximos detectam esse pedido e entregam os dados. Segundo, o *Didja* analisa o contexto de acordo com uma série de regras

que são configuradas na aplicação, a partir do momento em que chega o novo contexto, ele é imediatamente adicionado e avaliado por um comparador de contextos, usando uma medida de correlação estatística e análise proporcional. Por fim, estes resultados são utilizados para identificar e determinar se os dois dispositivos estão em um mesmo contexto ou não.

Na Figura 11 os autores apresentam uma visão geral do modelo proposto, o *Didja* examina o contexto de todos os dispositivos que encontram-se próximos, um com o outro. Logo em seguida, analisa a semelhanças, e a partir disto forma grupos com usuários em múltiplos contextos (ao centro).

Figura 11: Visão geral do *Didja*



Fonte: (FREITAS; DEY, 2015a)

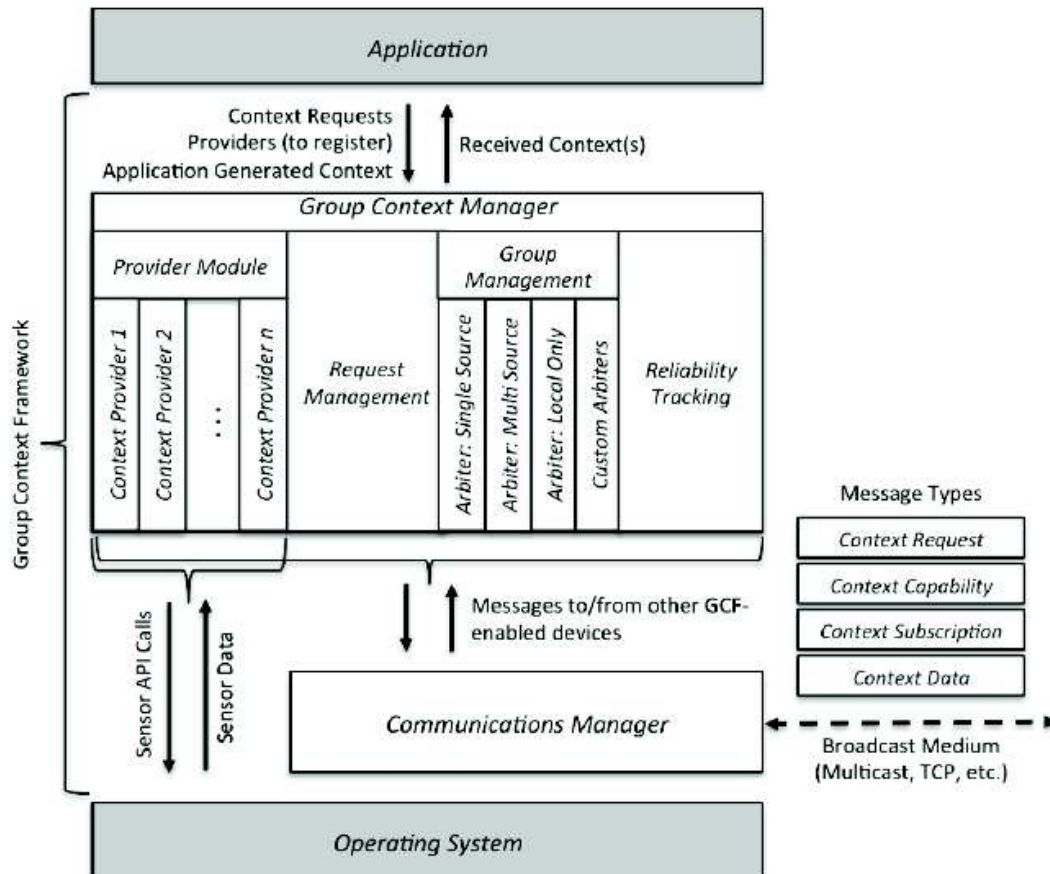
3.14 Freitas e Dey (2015b)-GCF Framework

Os autores Freitas e Dey (2015b) apresentam o *GCF*, um *Framework*⁹ extensível que permite aos dispositivos móveis compartilhar informações contextuais, pode-se dizer que este trabalho é uma melhoria do já citado acima (FREITAS; DEY, 2015a), uma vez que o *GCF* fornece aos desenvolvedores uma padronização para solicitação de dados contextuais, a fim de utilizar em outras aplicações.

⁹Segundo (CRESPO, 2000) um *Framework* pode ser definido como uma estrutura semiacabada de software utilizada para o desenvolvimento de aplicações. Para (DA FONTOURA, 1999), o *Framework* deve ser preenchido através da instanciação de seus componentes, possibilitando assim a criação de novas aplicações que possam interagir com o seu domínio-alvo.

O *Framework* permite que os dispositivos e usuários possam colaborar uns com os outros, após o agrupamento dos mesmos, através da realização de amplas atividades colaborativas possíveis. Os autores apresentam uma arquitetura bem definida, a qual está ilustrada na Figura 11, além de também validarem e comprovarem a utilização de tal arquitetura, implementando duas aplicações: *groupMap* e *GroupPhike*.

Figura 12: *GCF High Level Architecture*



Fonte: (FREITAS; DEY, 2015a)

O principal objetivo com *GCF* foi criar um conjunto de ferramentas que permitisse que dispositivos móveis pudessem formar grupos em um determinado contexto. Para conseguir isso, os autores apresentaram quatro princípios fundamentais do projeto:

- **Facilidade de Uso:** um *Framework* extensivo possibilita e facilita que programadores utilizem os ferramentais suportados.
- **Extensibilidade:** pode ser usado para formar uma gama de grupos, ou ainda, ser estendido, implementado conforme necessidade.
- **Agrupamento automático:** abstrai o processo de descoberta dos dispositivos e colaboração, os desenvolvedores precisam somente definir qual contexto sua aplicação requer.

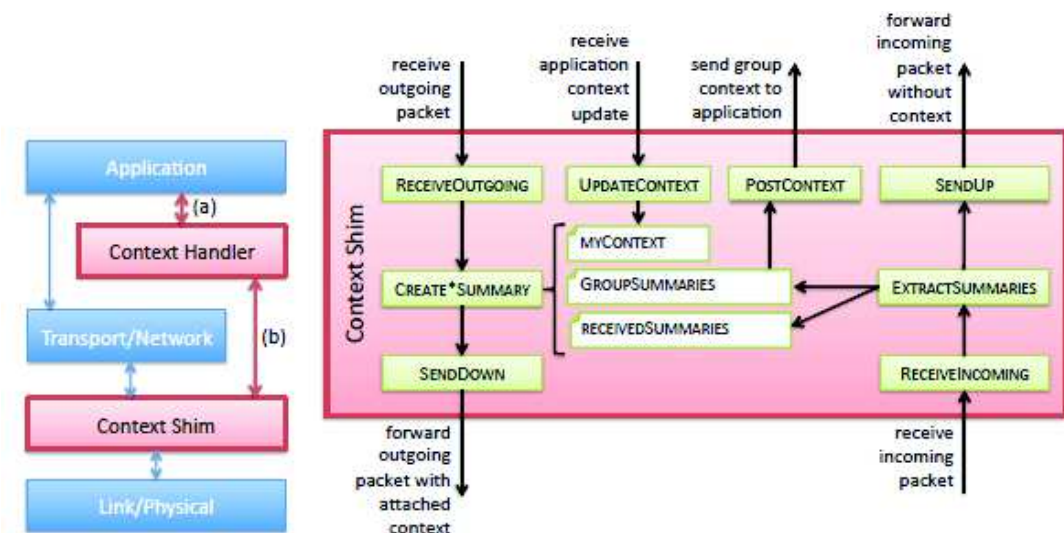
- Gestão de grupo dinâmico: monitoramento constante do estado do ambiente dentro do grupo e modificando conforme necessidade.

3.15 Julien (2011)

Segundo Julien (2011) o contexto em ambientes dinâmicos é imprevisível e entidades nestes ambientes não estão isoladas, uma vez que essas entidades dependem de alguma noção de contexto e agir através destas informações coletadas. Este artigo apresenta a necessidade de se compartilhar informações de contexto para a computação móvel, o autor defende que o contexto de um grupo de entidades dentro de uma rede dinâmica pode ser tão importante, quanto o comportamento global que tal sistema impõe.

Na Figura 13 o autor apresenta a arquitetura do *Framework* para um contexto compartilhado, é possível destacar que uma aplicação externa envia dados sobre contextos para o manipulador de contextos, que por sua vez recupera as informações sobre cada entidade e também do grupo. O módulo de *Context Shim* analisa examina a entrada de pacotes e atribui resumo a saída dos mesmos. O restante da arquitetura (destacado em vermelho e a direita), representa a parte interna do *Context Shim*, que armazena o *myContext*, que seriam as informações de contexto de tal entidade, em um dado momento.

Figura 13: Arquitetura para contexto compartilhado



Fonte: (JULIEN, 2011)

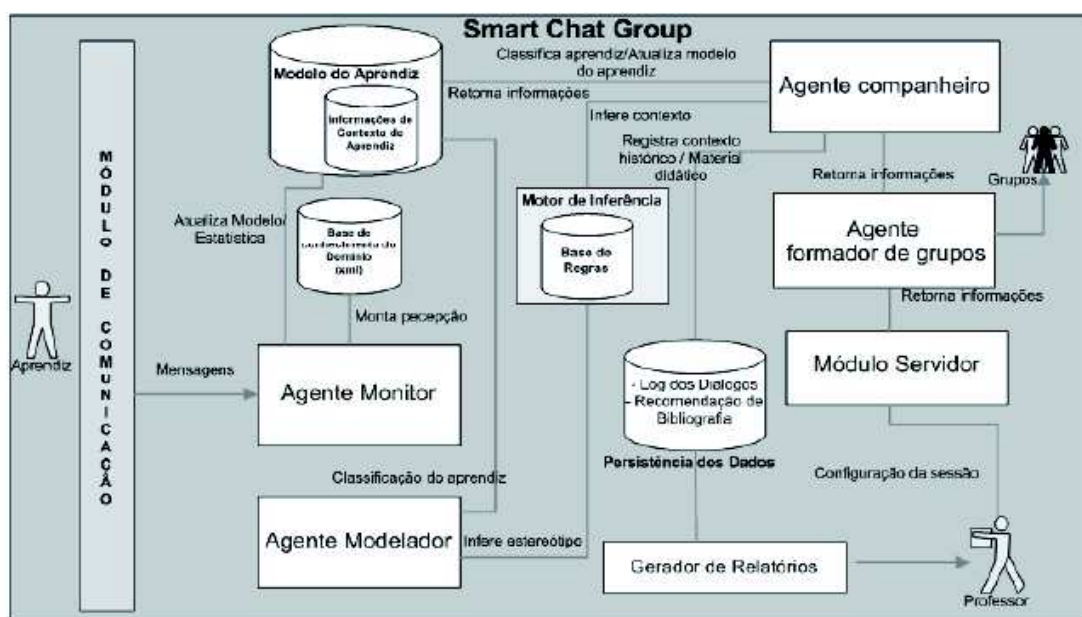
3.16 Felix e Tedesco (2008)

No artigo apresentado por Felix e Tedesco (2008), os autores apresentam a *Smart Chat Group*, uma ferramenta que usufrui de agentes inteligentes para possibilitar o acompanhamento,

sugestão e a formação automática de pequenos grupos de aprendizagem, estes grupos são baseados em informações de contexto de tais aprendizes. Portanto, é uma ferramenta colaborativa e ciente de contexto auxilia em atividades voltadas aos aprendizes, uma vez que acompanha e monitora o desempenho de cada um deles, possibilitando ao professor a recomendação de material didático aos alunos. Ainda disponibiliza ao professor um ambiente onde ele pode testar a formação de pequenos grupos em diferentes cenários de aprendizagem.

Na Figura 14 é ilustrado o modelo proposto pelos autores, onde utilizou-se dois agentes de softwares embarcados: um agente acompanhador e outro formador de grupos. O primeiro, é responsável pela interação com o aprendiz, recomendar material, atividades, troca mensagens e principalmente pelo auxílio na captação do contexto primário do aprendiz, e a inferência do contexto secundário do aprendiz, como: suas habilidades, deficiências e reputação. Já o outro, é responsável por formar e recomendar grupos, a formação destes grupos ocorre basicamente a partir das informações de contexto do aprendiz. Os grupos de aprendizes são separados pelo agente, através dos seus interesses pessoais e das informações de contexto dos mesmos.

Figura 14: Arquitetura *Smart Chat Group*



Fonte: (FELIX; TEDESCO, 2008)

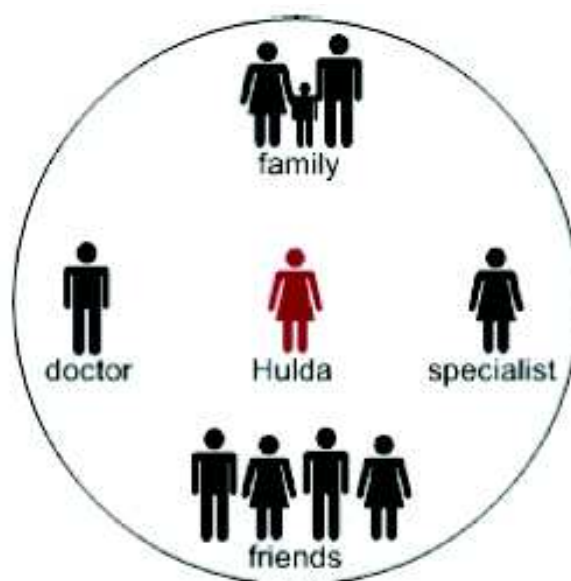
3.17 Hallberg et al. (2007)

Os autores Hallberg et al. (2007) apresentam um modelo de comunicação para formação e gestão de grupos dinâmicos, que utiliza informações baseadas em contexto para atribuir essa comunicação entre os membros. O foco principal deste artigo foi definir um modelo para grupos dinâmicos, levando em consideração aspectos importantes que também são ressaltados por essa

proposta, no que diz respeito a grupos, por exemplo: 1) Como encontrar usuários; 2) Como formar e manter os grupos; 3) Como formar, manter e apoiar os grupos, através de informações de contexto compartilhado.

Na Figura 15 é apresentado uma visão geral do sistema de grupos dinâmicos centrado no usuário, esse cenário é de uma perspectiva de saúde, onde o usuário que está ao centro do modelo tem total prioridade sobre o ambiente que está exposto ao grupo. *Hulda* (usuário ao centro) é uma senhora idosa, que vive em casa e utiliza deste sistema para comunicação entre o grupo, com a finalidade de manter relações sociais, uma vez que com a ferramenta ela possui fácil interação com os mesmos que compõe o grupo. Além dessa comunicação, o modelo utiliza informações sensíveis ao contexto, modulado justamente por razões médicas neste caso (em específico), dentre outras funções, como por exemplo: calendário para lembrá-la de tal evento em redes sociais, hora para ingerir tais remédios.

Figura 15: Visão geral do sistema de grupos dinâmicos centrado no usuário



Fonte: (HALLBERG et al., 2007)

3.18 Muehlenbrock (2005)

Muehlenbrock (2005) desenvolveram um módulo que coleta informações de contexto e demais dados que os perfis dos alunos possuam em tal situação, a fim de coincidir informações semelhantes entre os mesmos, para que possa ocorrer a formação de grupos de aprendizagem inteligente. A principal finalidade da análise dos perfis e agrupamento dos mesmos, é a colaboração, interação e crescimento que determinado perfil do aluno venha a conquistar dentro do grupo.

3.19 Coutand et al. (2005)

Os autores Coutand et al. (2005) propuseram um projeto que promove a gestão de grupos móveis sensíveis a contexto, esse projeto recebeu o nome de *MobiLife* e também apresenta uma arquitetura do sistema que foi desenvolvido, que aborda principalmente conceitos da Computação Ubíqua, para aplicações que almejem formação para grupos de forma onipresente.

O *MobiLife* dispõe a administração do grupo em diversos ambientes móveis, o que permite que grupos sejam criados a qualquer lugar e qualquer hora e que possam ser administrados, também suportando informações de contexto em que o grupo se encontra e mantendo o controle de informações relevantes do grupo.

3.20 Considerações sobre os trabalhos relacionados

A Tabela 1 apresenta um comparativo entre os trabalhos que foram apresentados durante este capítulo e o eGroup, a primeira coluna faz referência aos trabalhos analisados, enquanto as outras seis colunas detalham as características utilizadas pelos trabalhos analisados. Cabe ressaltar, que para realizar o comparativo, foram levados em consideração os seguintes aspectos:

- **Entidades Agrupadas:** É identificado o tipo de agrupamento entre as entidades, considerando que a entidade pode ser uma pessoa, local ou objeto que seja considerada relevante para a interação entre um usuário e uma aplicação, incluindo o próprio usuário e aplicação (WAGNER; BARBOSA; BARBOSA, 2014). Na Tabela 1 é indicado qual o tipo de entidade que no domínio da aplicação o modelo suporta, ou seja, quem é que naquele determinado domínio de aplicação é agrupado ou gerenciado, por exemplo: a entidade pode ser genérica (sendo utilizada para os mais diversos tipos de aplicação) ou também como no caso da Educação, pode ser uma entidade professor, aprendiz, aluno, ou ainda, aluno e professor.
- **Domínio de Aplicação:** Identifica se o modelo analisado tem uma aplicação específica ou se o modelo é genérico. Na Tabela 1 é especificado o domínio de aplicação de cada trabalho, por exemplo: Educação, Recomendação, Saúde, Economia, Jogos, entre outros ou então genérico.
- **Tratamento dos grupos:** É verificado entre os trabalhos relacionados, como eles adotam os grupos e utilizam as definições do termo gerenciamento de grupos, adotados no eGroup. Na Tabela 1, são usados os 3 termos para definir se o trabalho envolve todos, nenhum ou pelo menos um destes termos: **Formação de Grupos**, **Gestão do Grupo** e **Acompanhamento do Grupo** ou então o gerenciamento de grupos que é proposto e adotado nesta dissertação, conforme descrição na seção 2, subseção 2.4.

- **Contexto:** Identifica se o artigo adota informações de contexto do usuário para o gerenciamento de grupos, para isso, é levada em consideração a definição de Dey, Abowd e Salber (2001), onde “qualquer informação que possa ser usada para caracterizar a situação de entidades relevantes para a interação entre um usuário e uma aplicação, e que em geral é composto por dados que descrevem a situação, identidade e localização espaço-temporal de pessoas, grupos e objetos físicos ou computacionais”. Portanto, é adotado na Tabela 1 **Sim** para trabalhos com contexto e **Não** para trabalhos sem contexto.
- **Trilhas dos Grupos:** Considera se o trabalho armazena as trilhas dos grupos. A Tabela 1 a definição **Sim** significa que o modelo gerencia algum tipo de histórico dos dados do grupo, do contrário é atribuído **Não**.
- **Perfis dos Grupos:** Define que o perfil do grupo é tratado, por exemplo, um determinado campo que informa quantas pessoas tem em determinado grupo, qual a predominância de perfis destes grupos e o interesse do grupo. Na Tabela 1 é adotado **Sim** para trabalhos que abordam esses aspectos, do contrário é atribuído **Não**.

Tabela 1: Comparação entre os trabalhos apresentados

Trabalhos Relacionados	Entidades Agrupadas	Domínio Aplicação	Características			
			Tratamento dos Grupos	Contexto Dey, Abowd e Salber (2001)	Trilhas dos Grupos	Perfis dos Grupos
Bais et al. (2013)	Aluno e Professor	Educação	Formação	Não	Não	Não
Medina et al. (2013)	Aluno e Professor	Educação	Formação	Não	Não	Não
Boratto, Carta e Satta (2010)	Usuários	Recomendação	Formação e Recomendação de conteúdos	Não	Não	Não
Ani et al. (2010)	Aluno	Educação	Formação	Não	Não	Não
Webber e Lima (2012)	Aluno e Professor	Educação	Formação	Não	Não	Não
Kyprianidou et al. (2012)	Aluno e Professor	Educação	Formação	Não	Não	Não
Stalljohann e Hackeloer (2011)	Aluno	Educação	Formação	Não	Não	Não
Lowe et al. (2011)	Aluno	Educação	Formação	Não	Não	Não
Sancho-Asensio et al. (2014)	Aluno	Educação	Formação	Não	Não	Não
Mujkanovic et al. (2012)	Usuários	Educação	Formação	Não	Não	Não
Rana, Kristiansson e Synnes (2012)	Usuários	Educação	Formação e Gestão	Sim	Não	Não
Freitas e Dey (2015a)	Usuários	Genérico	Formação e Gestão	Sim	Não	Não
Freitas e Dey (2015b)- <i>GCF Framework</i>	Usuários	Genérico	Formação e Gestão	Sim	Não	Não
Julien (2011)	Usuários	Genérico	Formação e Gestão	Sim	Não	Não
Felix e Tedesco (2008)	Aluno	Educação	Formação e Acompanhamento	Sim	Não	Não
Hallberg et al. (2007)	Usuários	Saúde	Formação e Gestão	Sim	Não	Não
Muehlenbrock (2005)	Aluno	Educação	Formação	Sim	Não	Não
Coutand et al. (2005)	Usuários	Genérico	Formação e Gestão	Sim	Não	Não
eGroup	Genérico	Genérico	Gerenciamento de Grupos	Sim	Sim	Sim

Fonte: Elaborado pelo autor

Conforme a Tabela 1, é possível identificar que a maioria dos trabalhos apresentam características que são úteis para o modelo do eGroup. Uma das características importantes é que todos os trabalhos relacionados que foram adotados nesta pesquisa, trabalham com grupos. Também é possível destacar, que metade dos trabalhos relacionados realizam o agrupamento de alunos e aluno/professor, o restante agrupa qualquer tipo de entidade. O eGroup agrupará qualquer tipo de entidade, por isso foi denotado como genérico.

A maior parte dos trabalhos estão concentrados em um mesmo domínio de aplicação, que é a educação, outros trabalhos permitem o uso para a saúde e para recomendação de conteúdo, por fim, o restante dos trabalhos permitem a utilização em qualquer domínio de aplicação.

Apresentar um domínio específico torna o modelo mais simples, enquanto o domínio genérico possibilita a utilização do modelo para a formação de grupos em qualquer domínio de aplicação, seja na educação, saúde, transporte, entre outros.

Todos os trabalhos analisados efetuam a formação de grupos como um forma de tratamento para os mesmos. Alguns trabalhos vão além do simples agrupamento de entidades, efetuando também a gestão dos grupos, acompanhamento e recomendação de conteúdos. Não foram identificados trabalhos que envolvessem a integração destes 3 termos: **Formação de Grupos**, **Gestão do Grupo** e **Acompanhamento do Grupo**. A integração destes 3 termos é um dos diferenciais proposto pelo eGroup, criando o termo **Gerenciamento de Grupos**, conforme descrição no capítulo 2, seção 2.5.

A maioria dos trabalhos não leva em consideração informações de contexto segundo Dey, Abowd e Salber (2001), para a formação, gestão e/ou acompanhamento de grupos. Ainda que alguns destes trabalhos considerem a localização e/ou situação em que tal entidade se encontra em um dado momento para efetuar o agrupamento.

Nenhum dos trabalhos analisados armazena um histórico de dados do grupo, após a sua formação. Essa é uma das contribuições relevantes do eGroup, onde o modelo gera de forma instantânea uma trilha para cada grupo formado, atualizando-a conforme entrada e saída de entidades. Também não foram encontrados trabalhos que gerem um perfil do grupo, após a formação do mesmo, uma vez que esse perfil irá conter informações relevantes de cada grupo formado e será atualizado dinamicamente.

3.21 Considerações sobre o capítulo

Este capítulo apresentou os passos adotados para realização da pesquisa e os critérios de busca por trabalhos relacionados, após foram listadas as base de dados nas quais resultaram o montante de trabalhos da área. A partir da seção 3.2 até 3.19 foram descritos os trabalhos que foram selecionados conforme os critérios adotados. Por fim, a seção 3.20 apresenta um comparativo entre os trabalhos apresentados neste capítulo e o modelo que está sendo proposto.

O próximo capítulo apresenta o modelo proposto, o eGroup.

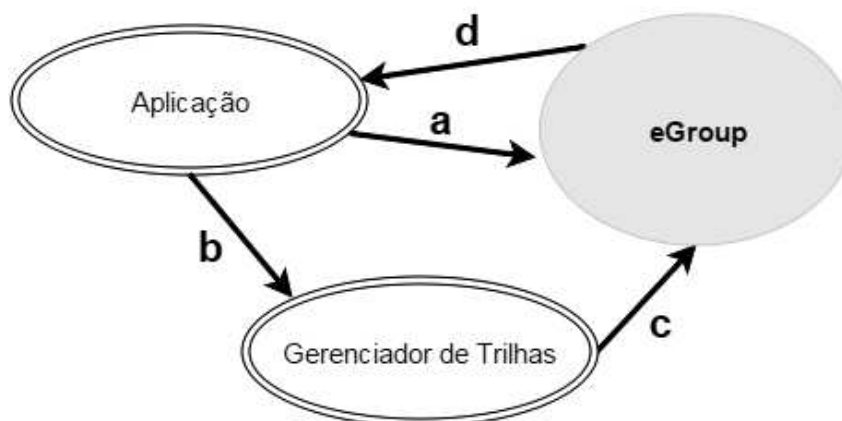
4 MODELO PROPOSTO

Neste capítulo é apresentado o modelo eGroup. Para o desenvolvimento do mesmo seguiu-se a linha de raciocínio desenvolvido pelo eProfile (WAGNER, 2013), o qual tornou-se uma referência para aplicabilidade de um modelo genérico para elaboração desta dissertação. A primeira seção apresenta uma visão geral e o funcionamento do modelo proposto. A seção 4.2 detalha a arquitetura e a comunicação dos módulos internos. A seção 4.3 detalha o protocolo de comunicação no modelo. A seção 4.4 apresenta o modelo de perfis das entidades. A seção 4.5 mostra como devem ser implementadas as regras para o gerenciamento dos grupos. A seção 4.6 apresenta o modelo de contexto das entidades para as aplicações. A seção 4.7 detalha o modelo de trilha das entidades para as aplicações. A seção 4.8 detalha como é o modelo de perfis e trilhas dos grupos gerados pelo eGroup. Por fim, a seção 4.9 apresenta as considerações sobre o Capítulo.

4.1 Visão geral do eGroup

Na Figura 16 é apresentada uma visão geral da utilização do eGroup, onde cada círculo representa um componente, e as setas indicam o fluxo dos dados.

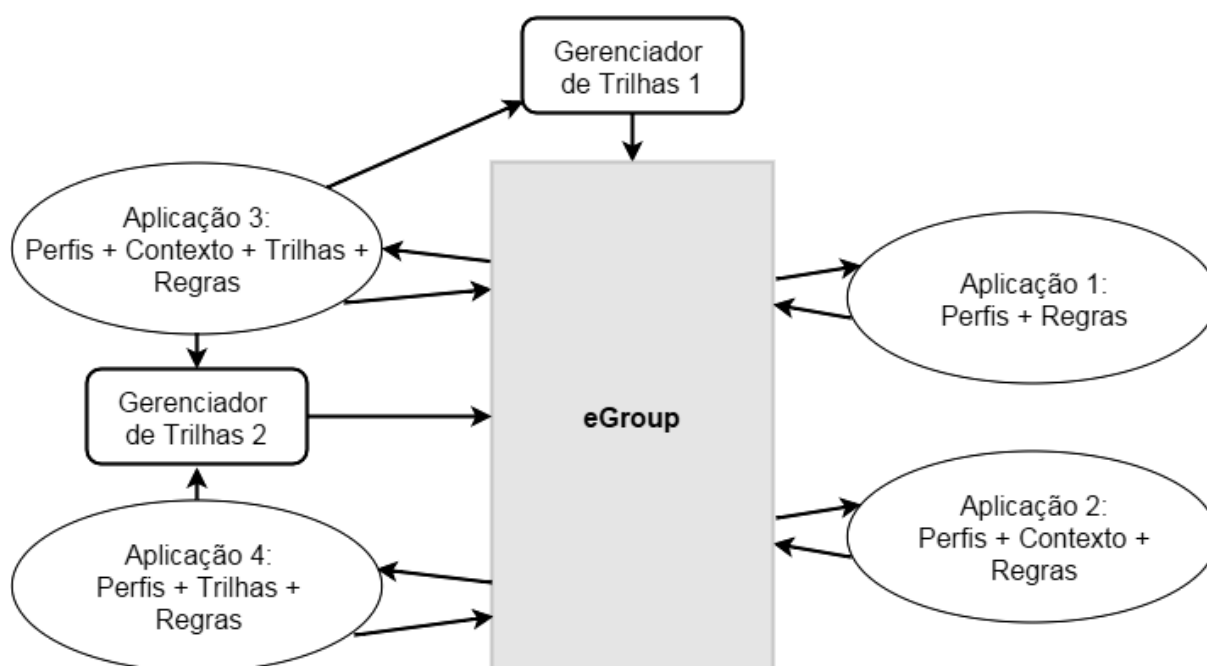
Figura 16: Visão geral do eGroup



Fonte: Elaborado pelo autor

O primeiro componente representado na Figura 16 é a **aplicação**. A aplicação para o modelo do eGroup, é qualquer aplicação que deseje utilizar o modelo, e que almeje o gerenciamento de grupos de entidades genéricas. Uma aplicação pode utilizar as funcionalidades oferecidas pelo eGroup de duas maneiras, a primeira delas é conectar-se diretamente com o eGroup (de modo que não possua a trilha das entidades armazenadas) ou então, se possuir trilhas, deve-se obrigatoriamente registrar suas informações em um gerenciador de trilhas suportado pelo eGroup, e conseqüentemente deverá seguir o protocolo de comunicação do eGroup a fim de se

Figura 17: eGroup conectado a mais de um gerenciador de trilhas e várias aplicações



Fonte: Elaborado pelo autor

comunicar com o mesmo. Uma vez seguidos estes pré-requisitos, a aplicação pode ser genérica, ou seja, de qualquer tipo e operar em qualquer dispositivo desde que possua comunicação com a internet.

O segundo componente representado é um **gerenciador de trilhas**. Um gerenciador de trilhas é capaz de registrar eventos que são realizados por uma determinada entidade, registrando informações relevantes do evento em questão (como o contexto em que o evento foi realizado). O eGroup é projetado para oferecer suporte a qualquer gerenciador de trilhas, desde que este seja desenvolvido externamente e siga o protocolo de comunicação do eGroup, especificado na seção 4.3.

A Figura 16 apresenta ainda um terceiro componente, o **eGroup**. Na seção 4.2 é detalhado a arquitetura do mesmo.

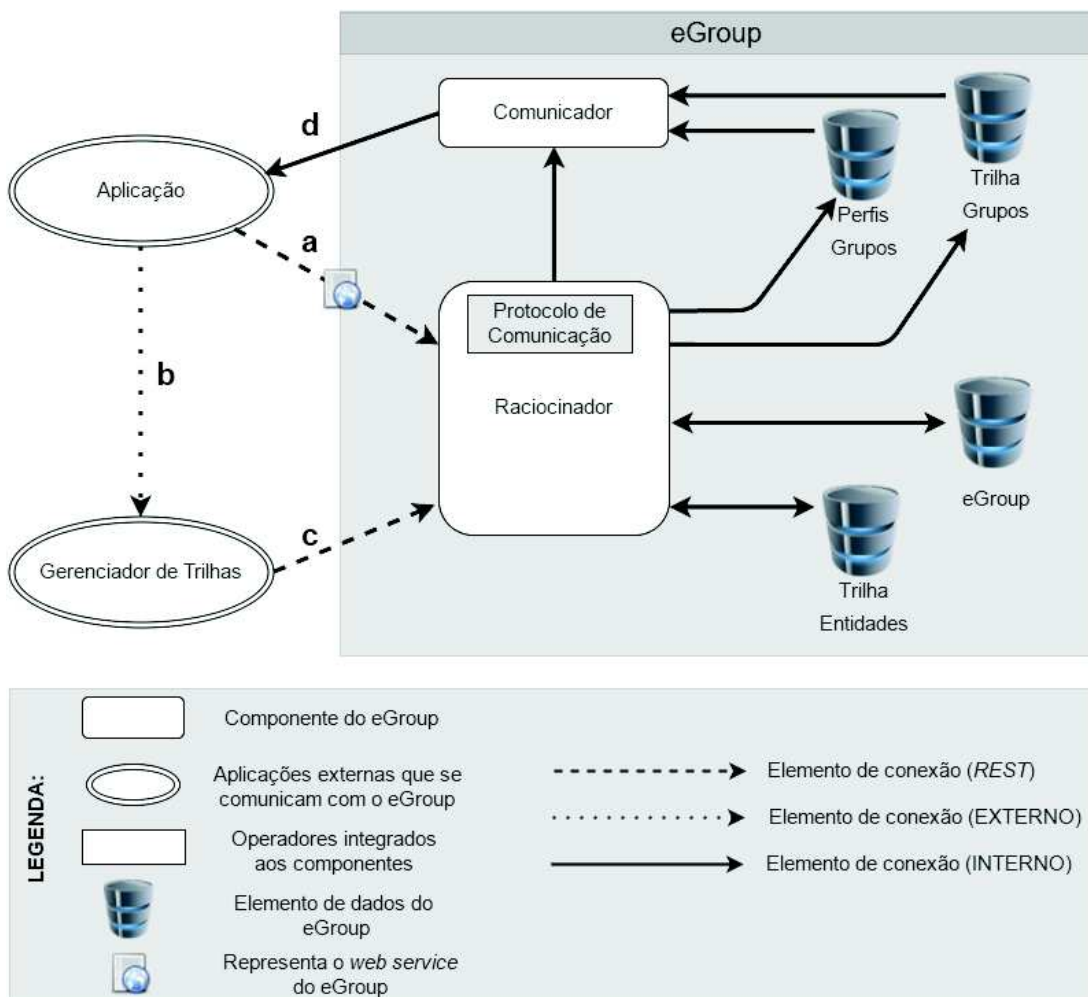
As setas da Figura 16 indicam o fluxo dos dados. A comunicação é iniciada quando a aplicação solicita um serviço ao eGroup, obedecendo o protocolo de comunicação (ver seção 4.3). Com a comunicação estabelecida, a aplicação já pode enviar as regras para formação dos grupos, os perfis das entidades e ainda o contexto das entidades (seta “a”). A partir deste momento, a aplicação pode registrar os eventos no gerenciador de trilhas (seta “b”) e conseqüentemente enviá-los ao eGroup (seta “c”). Desta forma, a cada nova informação recebida o eGroup analisa e reconstrói os grupos sempre com base nas regras. À medida que a aplicação tem necessidade de formar um grupo, ou então, informações atualizadas sobre os mesmos, ela poder efetuar consulta e buscar qual o perfil dos grupos ou então, a trilha dos mesmos (seta “d”).

A Figura 17 apresenta o eGroup conectado a várias aplicações, é possível visualizar que as aplicações podem conectar-se ao eGroup possuindo apenas os perfis das entidades, até possuir contexto e/ou trilha das entidades, sempre obedecendo o protocolo de comunicação (ver seção 4.3). Também é possível que as diferentes aplicações que se comunicam com o eGroup, utilizem diferentes gerenciadores de trilhas, desde que entreguem as trilhas no formato aceito pelo eGroup (ver seção 4.7).

4.2 Arquitetura

Na Figura 18 é apresentada a arquitetura de software do modelo eGroup, que é baseada em uma arquitetura de componentes, onde os componentes **aplicação** e **gerenciador de trilhas** são componentes externos.

Figura 18: Arquitetura do eGroup



Fonte: Elaborado pelo autor

As setas da Figura 18 são as mesmas representadas na Figura 16, porém na Figura 18, é expandido o funcionamento interno dos componentes do eGroup e cada seta representa o caminho dos dados. É possível verificar que existe dois pontos de entrada de dados (**setas a e c**) e apenas um ponto para saída de dados (**seta d**).

Cada aplicação que desejar utilizar o eGroup deve obrigatoriamente obedecer o protocolo de comunicação (ver seção 4.3). As primeiras dez trocas de mensagens do protocolo de comunicação, que ocorre entre a aplicação e o eGroup, permitem estabelecer comunicação e entrega de informações necessárias para formar os grupos. Na Figura 18, é possível visualizar o componente **raciocinador**, e dentro dele está integrado o **protocolo de comunicação**, o qual é inicializado quando ocorrer qualquer contato de uma aplicação externa, representada pela **seta a**. Essa comunicação é realizada via *web service*, utilizando um tipo de serviço *REST*.

Cabe ressaltar, que dessas dez mensagens, a **aplicação** realiza apenas quatro mensagens com o envio de informações para o eGroup. Para a sequência da comunicação, é necessário que ocorra a comunicação externa do eGroup, entre a **aplicação** e o **gerenciador de trilhas** indicados pela **seta b**. Desta forma, o **gerenciador de trilhas** comunica-se com o componente **raciocinador** do eGroup, enviando a trilha das entidades. O **raciocinador** assim que receber a trilha irá armazená-las na base de dados trilha das entidades.

Para os próximos passos, a aplicação precisa enviar ao eGroup em formato *JSON*, quatro informações: **(i)** que a aplicação solicite comunicação, envie o *IDMensagem* e uma descrição, **(ii)** modelo de perfil (ver seção 4.4), contexto (ver seção 4.6) e/ou trilha das entidades (ver seção 4.7) para obedecer o formato aceito pelo eGroup, **(iii)** regras para formar os grupos, conforme o modelo (ver seção 4.5), **(iv)** o perfil das entidades, conforme o modelo (ver seção 4.4).

Essas informações recebidas pelo eGroup, são interpretadas pelo **raciocinador**, e consequentemente armazenadas em tabelas separadas dentro da base de dados **eGroup**, são elas: *app*, *regras*, *perfis* e *contexto*. Em seguida, essas informações podem ser consultadas pelo **raciocinador**, que é o componente lógico do eGroup. Caso a **aplicação** envie a trilha das entidades, através do **gerenciador de trilhas**, o **raciocinador** irá armazená-las na base de dados **trilha das entidades**.

Tendo armazenado as regras e os perfis das entidades em suas base de dados, o eGroup consegue formar os grupos, através do seu componente **raciocinador**, nele está contido a lógica e o algoritmo responsável pelo gerenciamento dos grupos. O **raciocinador** agrupa as entidades sempre com base na regra que foi enviado pela **aplicação**, e registra as informações das entidades que formam determinado grupo na base de dados **trilha dos grupos**, sendo assim, é possível sempre ter informações atualizadas sobre o histórico de determinado grupo. Um grupo será destruído automaticamente pelo **raciocinador** quando nenhuma entidade possuir características conforme a regra de formação, ou então, quando o **raciocinador** receber da **aplicação**, através do protocolo de comunicação, uma regra para destruição do grupo *x*.

Semelhante ao caso das **trilha dos grupos**, aplica-se o funcionamento dos **perfis dos grupos**. Onde o **raciocinador** armazena na base de dados **perfis dos grupos**, a regra pelo qual o

grupo foi formado, por exemplo: as características que formam o grupo, o contexto (localização) que formam os grupos.

Através do componente **comunicador**, o eGroup envia para a **aplicação** um arquivo em formato *JSON*, com a **trilha dos grupos** e o **perfis dos grupos** (ver seção 4.8). Desta forma, sempre que preciso, a **aplicação** pode solicitar ao eGroup (através de regras), informações sobre os **perfis dos grupos** formados, e/ou o histórico de **trilha dos grupos** dos mesmos. O **raciocinador** interpreta a regra e comunica-se diretamente com **comunicador**, que recebe as informações da base de dados e notifica a aplicações sobre as informações solicitadas.

4.3 Comunicação do eGroup

A maior parte das comunicações do eGroup são síncronas no início das mesmas, mas no decorrer elas podem tornar-se assíncronas, uma vez que, o eGroup pode receber somente os contextos das entidades atualizados, ou então, somente os perfis das entidades atualizados, ou seja, uma vez recebido, não há uma obrigatoriedade à ser seguida.

Toda a comunicação do eGroup é realizada via *web service*, e um dos tipos de *web service* que tem se consolidado como padrão quando se trata em disponibilizar serviços na *web*, é o *REST* (MONTEIRO, 2013). Esse serviço apresenta algumas premissas, como por exemplo:

- Deve-se utilizar os métodos do protocolo *HTTP*;
- Apresentar informações através de *URLs*;
- O serviço não deve armazenar estado entre requisições;
- Transmitir dados em formato *XML* e/ou *JSON*.

Uma das vantagens do serviço *REST* é a compatibilidade com qualquer tipo de plataforma, desde a *web* até *mobile*. Como resultado deste serviço, está a obtenção de uma *URL* que, quando acessada por alguma aplicação, utilizando o método *HTTP* correto e os parâmetros necessários, retorna dados em formato de texto.

Esses dados, refere-se ao ato de pedir informações para um outro serviço qualquer. No caso do eGroup por exemplo, quando uma aplicação solicitar o perfil dos grupos, o eGroup devolverá as informações em um arquivo de texto, esse arquivo tem um formato conhecido como *JSON*, que é um formato utilizado para representar dados, principalmente quando é preciso converter dados estruturados para a forma textual (ver seção 4.8).

Para estabelecer uma comunicação com o eGroup, as aplicações devem seguir o protocolo de comunicação, detalhado na Figura 19. A comunicação é definida e tratada como sendo de aplicação para aplicação.

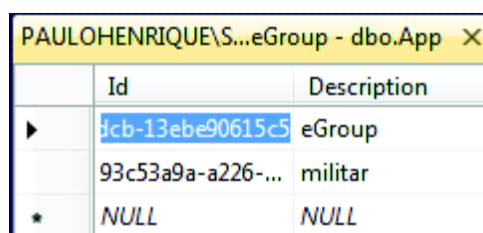
Cabe ressaltar, que inicialmente a **aplicação** realiza apenas quatro mensagens com o envio de informações para o eGroup, detalhadas nas mensagens 01, 04, 07 e 09 do protocolo de comunicação. Caso a **aplicação** já tenha se conectado ao eGroup, não é necessário realizar

trilhas (ver 4.2).

A seguir é detalhar cada mensagem do protocolo de comunicação ilustrado na Figura 19.

- **Mensagem 01:** Na primeira troca de mensagens entre o eGroup e a aplicação, pode ser enviado uma solicitação para conexão com o eGroup, a aplicação por sua vez deve enviar o *IDMensagem* que serve para o eGroup identificar qual tipo de serviço está sendo solicitado, como essa é a primeira troca de mensagens a aplicação deve enviar o *IDMensagem: 1*, juntamente com uma **Descrição**, que pode ser o nome dela ou algo que lhe identifique, essa descrição deve conter no máximo 50 caracteres.
- **Mensagem 02:** Após receber a solicitação e a descrição, o eGroup sinaliza o aceite de solicitação para a aplicação, e conseqüentemente envia um **eGroupID**, o qual é gerado internamente pelo eGroup, quando a aplicação conecta-se pela primeira vez. Esse **eGroupID**, é um identificador único da aplicação, que será acoplado junto a descrição enviada pela aplicação. Isso tudo é armazenado dentro de uma tabela chamada **App**, que está na base de dados chamada eGroup, ambas são criadas por padrão pelo eGroup, toda vez que uma nova aplicação é aceita. A Figura 20 mostra a parte interna do *SQL*.

Figura 20: eGroupID



	Id	Description
▶	1cb-13ebe90615c5	eGroup
	93c53a9a-a226-...	militar
*	NULL	NULL

Fonte: Elaborado pelo autor

- **Mensagem 03:** O eGroup notifica a aplicação que as informações recebidas não estão de acordo com o funcionamento do eGroup, anunciando falha na comunicação. Os temporizadores destacados na Figura 19, emitem um alerta para aplicação a cada minuto.
- **Mensagem 04:** Em casos onde a aplicação já tenha se conectado ao eGroup, ou seja, já possua o *eGroupID*, a mesma não precisa passar pelos passos anteriores, desde que, envie o *IDMensagem: 2*, juntamente com o seu *eGroupID* e o modelo de perfis, contexto e trilha das entidades.
- **Mensagem 05:** Após receber o modelo de perfis, contexto e/ou trilhas das entidades, o eGroup sinaliza o aceite destes modelos para a aplicação, e conseqüentemente armazena as informações em tabelas separadas dentro do banco de dados, essas tabelas são geradas automaticamente pelo eGroup com o nome da descrição enviada pela aplicação, mais o nome de cada modelo. Por exemplo: militarPerfis, militarContexto e militarTrilhas.

- **Mensagem 06:** O eGroup notifica a aplicação que as informações recebidas não estão de acordo com o funcionamento do eGroup, anunciando falha no recebimento do modelo de perfil, contexto e/ou trilha das entidades. Os temporizadores destacados na Figura 19, emitem um alerta para aplicação a cada minuto.
- **Mensagem 07:** Recebendo o aceite do modelo de perfil, contexto e/ou trilha, a aplicação deve enviar o *IDMensagem: 3*, juntamente com o seu *eGroupID* e as regras para formação dos grupos, o eGroup interpreta e armazena essas regras em uma tabela chamada regras que está dentro da base de dados chamada eGroup, ambas são criadas por padrão pelo eGroup. Os identificadores da Figura 20, indicam que a aplicação pode enviar novas regras, e conseqüentemente formar novos grupos a cada instante.
- **Mensagem 08:** O eGroup notifica a aplicação que as informações recebidas não estão de acordo com o funcionamento do eGroup, anunciando falha no recebimento das entidades. Os temporizadores da Figura 19, emitem um alerta para aplicação a cada minuto.
- **Mensagem 09:** Nesta mensagem a aplicação deve enviar o *IDMensagem: 4*, juntamente com o seu *eGroupID* e enviar as entidades para o eGroup, este por sua vez, armazena em uma tabela que está dentro da base de dados chamada eGroup, ambas são criadas por padrão pelo eGroup. O nome da tabela é atribuído pela junção da descrição enviada pela aplicação + o nome trilha, ficando: *militarTrilha*. Os identificadores da Figura 20, indicam que quando a aplicação necessitar enviar suas entidades atualizadas (devido alguma alteração), o eGroup sobrescreve todas as informações.
- **Mensagem 10:** O eGroup notifica a aplicação que as informações recebidas não estão de acordo com o funcionamento do eGroup, anunciando falha no recebimento das regras ou no formato delas. Neste caso, o eGroup envia para a aplicação um modelo de regras, que deve ser seguido, para que assim o eGroup consiga interpretá-lo. Os temporizadores da Figura 19, emitem um alerta para aplicação a cada minuto.

Essas 10 primeiras trocas de mensagens são necessárias para que o eGroup consiga formar os grupos, com base nos perfis das entidades e regras genéricas, ambos definidos e enviados pela aplicação. As próximas mensagens são para o envio dos contextos e trilhas das entidades, e também para solicitações de grupos formados e destruição dos mesmos.

- **Mensagem 11:** Caso a aplicação armazene o contexto de suas entidades, a mesma pode enviá-los para o eGroup, desde que sigam o modelo estabelecido pelo eGroup. A aplicação deve obedecer o protocolo enviando o *IDMensagem: 5*, juntamente com o seu *eGroupID*.
- **Mensagem 12:** O eGroup notifica a aplicação que as informações recebidas não estão de acordo com o funcionamento do eGroup, anunciando falha no recebimento do contexto

ou no formato dele. Neste caso, o eGroup envia para a aplicação o modelo de contexto que deve ser seguido, para que assim o eGroup consiga interpretá-lo. Os temporizadores da Figura 19, emitem um alerta para aplicação a cada minuto.

- **Mensagem 13:** Caso a aplicação armazene as trilhas das entidades, a mesma deve se comunicar com um gerenciador de trilhas, esse gerenciador é o responsável pela comunicação com o eGroup e ele deve ser desenvolvido externamente do eGroup, além de obedecer o modelo de trilha estabelecido pelo eGroup. A aplicação deve obedecer o protocolo enviando o *IDMensagem: 6*, juntamente com o seu *eGroupID*.
- **Mensagem 14:** O eGroup notifica a aplicação que as informações recebidas não estão de acordo com o funcionamento do eGroup, anunciando falha no recebimento da trilha ou no formato dela. Neste caso, o eGroup envia para a aplicação o modelo de trilha que deve ser seguido, para que assim o eGroup consiga interpretá-lo. Os temporizadores da Figura 19, emitem um alerta para aplicação a cada minuto.
- **Mensagem 15:** Neste caso a aplicação solicita ao eGroup os grupos que estão ativos, enviando o *IDMensagem: 7*.
- **Mensagem 16:** O eGroup envia para a aplicação os grupos que estão ativos, essas informações são enviadas em um arquivo no formato *JSON* (ver seção 4.8).
- **Mensagem 17:** Neste caso a aplicação solicita ao eGroup os perfis dos grupos que estão ativos, enviando o *IDMensagem: 8*.
- **Mensagem 18:** O eGroup envia para a aplicação os perfis dos grupos que estão ativos, essas informações são enviadas em um arquivo no formato *JSON* (ver seção 4.8).
- **Mensagem 19:** Neste caso a aplicação solicita ao eGroup a trilha dos grupos que estão ativos e/ou que já foram formados, enviando o *IDMensagem: 9*.
- **Mensagem 20:** O eGroup envia para a aplicação a trilha dos grupos que estão ativos e também dos que já foram destruídos, essas informações são enviadas em um arquivo no formato *JSON* (ver seção 4.8).
- **Mensagem 21:** A aplicação pode enviar ao eGroup regras para destruir os grupos, enviando o *IDMensagem: 10*.
- **Mensagem 22:** O eGroup envia para a aplicação que o grupo foi destruído conforme tal regra enviada, essas informações são enviadas em um arquivo no formato *JSON* (ver seção 4.8).

4.4 Perfis das Entidades

Para enviar o perfil das entidades ao eGroup, a **aplicação** deve seguir o modelo de perfil aceito pelo eGroup, conforme indicado da Figura 21. Observando a Figura, é possível perceber que existem atributos que são obrigatórios para a comunicação entre a **aplicação** e o eGroup, mas também existem outros atributos que são opcionais, já que os perfis das entidades podem conter diferentes e diversas informações, e essas informações podem ser alteradas e enviadas em um intervalo de tempo especificado.

A Figura 21, além de exemplificar os tipos primitivos aceitos pelo eGroup para os atributos obrigatórios, esboça outros tipos aceitos para os atributos opcionais.

Figura 21: Modelo de perfil das entidades

Atributos Obrigatórios	
IDMensagem	4
eGroupID	string
descricao	string
Atributos Opcionais	
atributo1	tipo primitivo
atributo2	tipo primitivo
atributo3	tipo primitivo
atributo4	tipo primitivo

Fonte: Elaborado pelo autor

O primeiro atributo obrigatório é o *IDMensagem: 4* que aparece na linha 2 da Figura 22 e a **descrição** que aparece na linha 4 da Figura 22, cujo qual imaginou um exemplo com informações relacionadas a **cavalo**, mas poderia ser qualquer *string* que identifique a **aplicação**. O outro atributo obrigatório é o id gerado pelo eGroup, que aparece como **eGroupID** na linha 3 da Figura 22, o qual é gerado assim que houver a comunicação entre a **aplicação** e o eGroup, que por sua vez, envia o **eGroupID** para a **aplicação** através da mensagem 02 do protocolo de comunicação (ver seção 4.3).

Outra obrigatoriedade do eGroup, é o formato do arquivo que será enviado pela **aplicação**. Na Figura 22 as informações encontram-se em um arquivo no formato *JSON*, que é o formato aceito pelo eGroup (ver seção 4.3).

4.5 Modelo de Regras

O envio de regras da **aplicação** para o eGroup permitem o gerenciamento dos grupos, ou seja, é através de uma regra executada pelo **raciocinador**, que serão formados os grupos, com

Figura 22: Exemplo de modelo de perfil das entidades utilizando *JSON*

```

1  [{"IDMensagem" : "4",
2     "eGroupId" : "1003",
3     "description" : "cavalo"
4  },
5  {
6     "id" : "0001",
7     "nome" : "vendaval",
8     "raca" : "criolo",
9     "idade" : "23",
10    "peso" : "103.4"
11  },
12  {
13     "id" : "0002",
14     "nome" : "pangare",
15     "raca" : "shile",
16     "idade" : "29",
17     "peso" : "110.6"
18  },
19  {
20     "id" : "0003",
21     "nome" : "catatau",
22     "raca" : "shile",
23     "idade" : "21",
24     "peso" : "86.9"
25  },
26  {
27     "id" : "0004",
28     "nome" : "pangare",
29     "raca" : "arabe",
30     "idade" : "20",
31     "peso" : "86.4"
32  }
33 ]

```

Fonte: Elaborado pelo autor

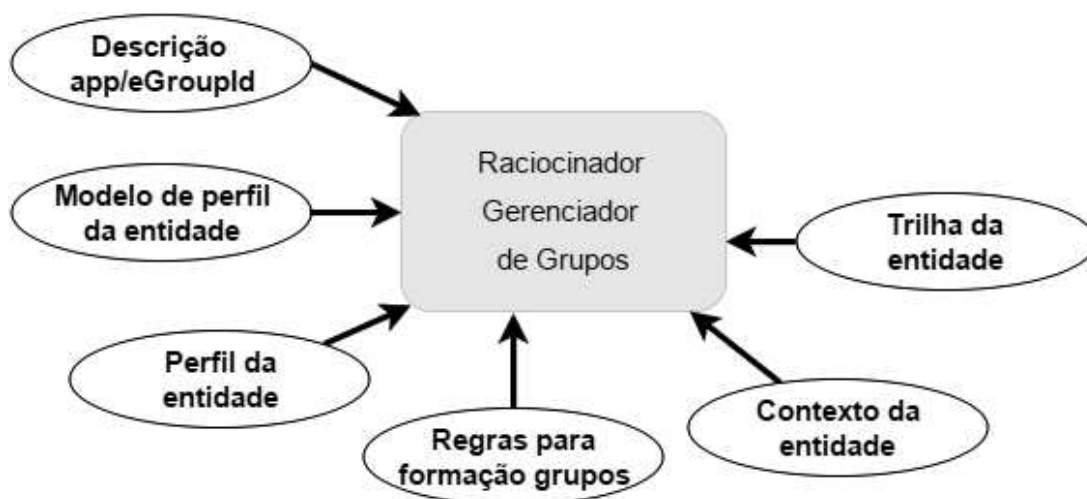
base nas informações que já estão armazenadas no eGroup. Desta forma, para que uma regra seja aplicada pelo eGroup, é necessário que as seguintes informações já estejam armazenadas no eGroup:

- A aplicação que irá utilizar o eGroup, com uma descrição (*string*) e *eGroupId*;
- Um modelo de perfil das entidades;
- As informações dos perfis das entidades, seguindo o formato do modelo estabelecido;
- É opcional o envio de informações de contexto e/ou trilhas.

A partir destas três e/ou cinco informações (nos casos onde tenha contexto e trilhas) o **ra-ciocinador** tem condições para formar os grupos, conforme a Figura 23. Cabe ressaltar que

as regras sempre são definidas e enviadas pela **aplicação**, e só serão aceitas pelo eGroup se estiverem no formato definido. Uma regra não pode ser editada, caso a **aplicação** deseje formar grupos com outras informações ou outro cenário, terá de enviar um novo conjunto de regras. Uma regra é composta por um conjunto de informações, especificadas abaixo:

Figura 23: Fluxo de dados para formar os grupos através de uma regra



Fonte: Elaborado pelo autor

- **Quantidade de filtros:** É o campo onde a **aplicação** deve definir a quantidade de filtros que ela deseja utilizar para formar os grupos. No exemplo da Figura 24, existe um único filtro, que é buscar e agrupar todas as entidades com base no atributo localização, que é *RS*.
- **Início da condição:** É um campo importante para o funcionamento do eGroup, é através dele que o **raciocinador** interpreta o início de um novo filtro, que está dentro de uma mesma regra, ou seja, é possível ter inúmeros filtros, dentro de uma única regra. O início da condição é sempre representado pela **sustenido (#)**.
- **Atributo:** É a informação que a **aplicação** deseja atribuir como regra para agrupar as entidades, geralmente é o campo cadastrado no modelo de perfil da entidade (ver seção 4.4).
- **Condição:** Refere-se a condição aplicada a uma determinada regra. Abaixo é exemplificado uma estrutura condicional aceita pelo **raciocinador**:
 - < representa o sinal de menor;
 - > representa o sinal de maior;
 - >= representa o sinal de maior e igual;

- <= representa o sinal de menor e igual.
- **Valor:** É o valor que a **aplicação** deseja encontrar dentro do atributo, no exemplo da Figura 24 é *RS*.
- **Operador lógico:** Entram em ação quando a **aplicação** envia mais de um filtro, abaixo encontram-se os dois operadores lógicos utilizados e aceitos pelo **raciocinador**:
 - | representa o ou;
 - & representa a junção de duas ou mais condições.

Na Figura 24, é possível visualizar um exemplo de regra simples, onde existe apenas um filtro, onde o atributo especificado é a localização das entidades que estejam no RS. Já na Figura 25, cabe destacar que a quantidade de filtros é 3, neste exemplo ocorre a junção de filtros, e sempre que ocorrer a junção destes, o **raciocinador** irá validar e agrupar as 2 primeiras colunas, caso exista uma terceira coluna, o **raciocinador** irá validar e comparar com as 2 primeiras. Ainda sobre a Figura 25, tem-se 3 atributos, o primeiro deles é raça que deve ser árabe, o segundo é idade que deve ser maior ou igual a 20 e por fim, idade maior ou igual a 23.

Figura 24: Modelo de Regras Simples do eGroup

Quantidade de Filtros	Início da Condição	Atributo	Condição	Valor
1	#	localizacao	=	RS

Regra simples no formato

1#localizacao=RS

Fonte: Elaborado pelo autor

4.6 Modelo de Contexto

Para enviar o contexto das entidades ao eGroup, a **aplicação** deve seguir o modelo de contexto aceito pelo eGroup, conforme exemplo da Figura 26. Observando a Figura, é possível perceber que existem atributos que são obrigatórios para a comunicação entre a **aplicação** e o eGroup, mas também existem outros atributos que são opcionais, já que o contexto das entidades podem ser alteradas e enviadas constantemente.

Na Figura 27 o primeiro atributo obrigatório é o *IDMensagem: 5* que aparece na linha 2, o outro atributo obrigatório é o id gerado pelo eGroup, que aparece como *eGroupId* na linha 3

Figura 25: Modelo de Regras do Completo eGroup

Quantidade de Filtros	Início da Condição	Atributo	Condição	Valor
3	#	raca	=	arabe

Início da Condição	Operador Lógico	Atributo	Condição	Valor
#	&	idade	>=	20

Início da Condição	Operador Lógico	Atributo	Condição	Valor
#	&	idade	<=	23

Regra com operadores lógicos no formato

3#raca=arabe#&idade>=20#&idade<=23

Fonte: Elaborado pelo autor

da Figura 27, o qual é gerado assim que houver a comunicação entre a **aplicação** e o eGroup, que por sua vez, envia o *eGroupId* para a **aplicação** através da mensagem 02 do protocolo de comunicação (ver seção 4.3).

Outra obrigatoriedade do eGroup, é o formato do arquivo que será enviado pela **aplicação**. Na Figura 27 as informações encontram-se em um arquivo no formato *JSON*, que é o formato aceito pelo eGroup.

Ainda sobre a Figura 27, as outras informações são um exemplo do que a **aplicação** pode enviar para o eGroup como sendo contexto, caso a **aplicação** não armazene o contexto das entidades esse arquivo *JSON* virá vazio. Abaixo são listada as informações básicas considerando contexto:

- Identidade: id;
- Localização: latitude e longitude;
- Situação: apto e não apto;
- Tempo: data/hora.

Figura 26: Modelo de contexto das entidades

Atributos Obrigatórios	
IDMensagem	5
eGroupID	string
Atributos Opcionais	
atributo1	valor
atributo2	valor
atributo3	valor
atributo4	valor

Fonte: Elaborado pelo autor

4.7 Modelo de Trilha

Para enviar a trilha das entidades ao eGroup, a **aplicação** deve obrigatoriamente comunicar-se com um gerenciador de trilha e deve seguir o modelo de trilha, conforme exemplo da Figura 28. Observando a Figura, é possível perceber que existem atributos que são obrigatórios para a comunicação entre a **aplicação** e o eGroup, mas também existem outros atributos que são opcionais e que podem variar dependendo as informações que **aplicação** registra na trilha das entidades.

Na Figura 29 o primeiro atributo obrigatório é o *IDMensagem: 6* que aparece na linha 2, o outro atributo obrigatório é o id gerado pelo eGroup, que aparece como *eGroupId* na linha 3 da Figura 27, o qual é gerado assim que houver a comunicação entre a **aplicação** e o eGroup, que por sua vez, envia o *eGroupId* para a **aplicação** através da mensagem 02 do protocolo de comunicação (ver seção 4.3).

Outra obrigatoriedade do eGroup, é o formato do arquivo que será enviado pela aplicação. Na Figura 29 as informações encontram-se em um arquivo no formato *JSON*, que é o formato aceito pelo eGroup (ver seção 4.3).

4.8 Perfis e Trilhas dos grupos

Um grupo para o modelo proposto neste trabalho é o agrupamento de duas ou mais entidades que possuam características de perfil e/ou contexto e/ou informações relacionadas a um período de tempo específico através das trilhas e que sejam agrupadas baseado em regras específicas.

Os perfis dos grupos são basicamente a replicação de uma regra, ou seja, é através da regra enviada pela aplicação que é gerado o perfil de um grupo. Como exemplo podemos citar, uma regra que define que as entidades devem estar na sala do MobiLab e que sejam graduados em Ciência da Computação ou Sistema de Informação. O grupo será formado por entidades

Figura 27: Modelo de contexto das entidades utilizando *JSON*

```

1  [[
2      "IDMensagem" : "5",
3      "eGroupId" : "1003"
4  ],
5  {
6      "id" : "0001",
7      "latitude" : "-29.7612660",
8      "longitudo" : "-51.1514820",
9      "situacao" : "apto",
10     "data" : "19-10-2015",
11     "hora" : "06:05:10"
12  },
13  {
14     "id" : "0002",
15     "latitude" : "-29.7610164",
16     "longitudo" : "-51.1514820",
17     "situacao" : "apto",
18     "data" : "19-10-2015",
19     "hora" : "06:05:10"
20  },
21  {
22     "id" : "0003",
23     "latitude" : "-29.7610164",
24     "longitudo" : "-51.1514820",
25     "situacao" : "nao-apto",
26     "data" : "15-10-2015",
27     "hora" : "06:07:10"
28  },
29  {
30     "id" : "0004",
31     "latitude" : "-29.7610164",
32     "longitudo" : "-51.1514820",
33     "situacao" : "nao-apto",
34     "data" : "15-10-2015",
35     "hora" : "06:07:10"
36  }}

```

Fonte: Elaborado pelo autor

que apresentam esse perfil e informações de contexto, no momento da formação, se nenhuma entidade apresentar tais características, o grupo estará vazio. No caso de três entidades possuírem o perfil e informações de contexto com as características definidas pela regra o grupo será formado por essas três entidades.

Na Figura 30 na linha 1 pode-se visualizar o identificador da regra pelo qual o grupo foi formado, na linha 3 é possível visualizar as entidades que pertencem aquele grupo naquele momento conforme a regra. As informações lógicas que compõem a regra são separadas por

Figura 28: Modelo de trilha das entidades

Atributos Obrigatórios	
IDMensagem	6
eGroupID	string
Atributos Opcionais	
id	0001
latitude	-29.7610164
longitude	-51.1514820
situacao	apto
data	15-10-2015
hora	06:07:10

Fonte: Elaborado pelo autor

um atributo lógica conforme linha 6 Figura 30.

Já a trilha do grupo é a replicação das informações de contexto do grupo, uma vez que a trilha é o histórico de contexto visitados por uma determinada entidade, para o eGroup essas informações são a data e hora em que o grupo foi formado, o número de entidades que fazem parte do grupo em um determinado momento, informação com a qual é possível caracterizar se houve variação no tamanho do grupo e os identificadores das entidades que compõem aquele grupo.

Na Figura 31 é possível visualizar que o primeiro objeto do arquivo *JSON* é basicamente a replicação do perfil do grupo, armazenando na linha 3 qual o identificador do grupo e na linha 4 o identificador da regra pela qual o grupo foi formado. No próximo objeto da Figura 31 é possível visualizar o retorno das informações que caracterizam a trilha do grupo.

4.9 Considerações sobre o capítulo

Neste capítulo foi apresentado o modelo do eGroup, inicialmente foi efetuado uma abordagem a visão geral do modelo, seguido de uma apresentação do modelo conectado a várias aplicações e como funciona o fluxo de dados do mesmo. A seção 4.2 detalhou a arquitetura baseada em componentes, bem como a comunicação externa e de seus módulos internos. A seção 4.3 especificou a comunicação do eGroup, detalhando as mensagens que são exercidas no protocolo de comunicação. As seções seguintes apresentam o modelos de perfis, regras, contexto e trilhas, em ambas elas é especificado os formatos aceitos pelo modelo. A seção 4.8 apresentou uma visão geral a respeito da geração do perfil e trilha dos grupos.

O próximo Capítulo define os aspectos de implementação do protótipo e os cenários de avaliação do modelo.

Figura 29: Modelo de trilha das entidades utilizando *JSON*

```

1  [{"IDMensagem" : "6",
2     "eGroupId" : "1003"
3  },
4  {
5     "id" : "0001",
6     "latitude" : "-29.7612660",
7     "longitude" : "-51.1514820",
8     "situacao" : "apto",
9     "data" : "19-10-2015",
10    "hora" : "06:05:10"
11  },
12  {
13     "id" : "0002",
14     "latitude" : "-29.7610164",
15     "longitude" : "-51.1514820",
16     "situacao" : "apto",
17     "data" : "19-10-2015",
18     "hora" : "06:05:10"
19  },
20  {
21     "id" : "0003",
22     "latitude" : "-29.7610164",
23     "longitude" : "-51.1514820",
24     "situacao" : "nao-apto",
25     "data" : "15-10-2015",
26     "hora" : "06:07:10"
27  },
28  {
29     "id" : "0004",
30     "latitude" : "-29.7610164",
31     "longitude" : "-51.1514820",
32     "situacao" : "nao-apto",
33     "data" : "15-10-2015",
34     "hora" : "06:07:10"
35  }
36  ]

```

Fonte: Elaborado pelo autor

Figura 30: Modelo de perfis dos grupos *JSON*

```

1  [
2     {
3         "idRegra" : "0111",
4         "idEntidades" : "0001, 0003, 0005",
5         "localizacao" : "MobiLab",
6         "graduado" : "Ciência Computação",
7         "logica" : "ou",
8         "graduado" : "Sistema Informação"
9     }
10 ]

```

Fonte: Elaborado pelo autor

Figura 31: Modelo de trilha dos grupos *JSON*

```
1  [
2    {
3      "idGrupo" : "0001"
4      "idRegra" : "0111",
5      "localizacao" : "MobiLab",
6      "graduado" : "Ciência Computação",
7      "logica" : "ou",
8      "graduado" : "Sistema Informação"
9    }
10 ]
11 [
12   {
13     "idEntidades" : "0001, 0003, 0005",
14     "data" : "01/08/2016",
15     "hora" : "12:00",
16     "tamanho" : "3"
17   },
18
19   {
20     "idEntidades" : "0001, 0003, 0005",
21     "data" : "05/08/2016",
22     "hora" : "12:05",
23     "tamanho" : "3"
24   },
25   {
26
27     "idEntidades" : "0003, 0005",
28     "data" : "10/08/2016",
29     "hora" : "12:10",
30     "tamanho" : "2"
31   },
32   {
33     "idEntidades" : "0001, 0003, 004, 005",
34     "data" : "15/08/2016",
35     "hora" : "12:15",
36     "tamanho" : "4"
37   }
38 ]
```

Fonte: Elaborado pelo autor

5 ASPECTOS DE IMPLEMENTAÇÃO E AVALIAÇÃO

Este Capítulo apresenta os aspectos de implementação e avaliação do modelo proposto. Para que o eGroup seja avaliado, foi necessário implementar um protótipo do modelo, para que fosse utilizado em experimentos com aplicações. A seção 5.1 apresenta os aspectos de implementação para o desenvolvimento do protótipo do eGroup. A seção 5.2 descreve a metodologia utilizada para a avaliação do eGroup. Por fim, a seção 5.2.1 apresenta o cenário desenvolvido para a aplicação e avaliação do eGroup.

5.1 Aspectos de Implementação

O processo de implementação do eGroup foi dividido em duas etapas. A primeira etapa, denominada Planejamento/Análise, consistiu em gerar toda a documentação técnica necessária para a segunda etapa, denominada Codificação, onde foi desenvolvido um protótipo do eGroup.

Na etapa de Planejamento/Análise foi utilizado o UML (*Unified Modeling Language*) (UML, 2016) para modelagem do sistema. O UML é um padrão internacionalmente reconhecido tanto no meio acadêmico quanto no mercado de desenvolvimento, conhecido como uma linguagem visual que tem como objetivo especificar, construir e documentar os artefatos para o desenvolvimento de um sistema de informação (LARMAN, 2002).

O Diagrama de Classes do eGroup é apresentado na Figura 32, onde podem ser visualizadas as classes que foram implementadas durante o desenvolvimento. Para a elaboração deste Diagrama foi utilizada a ferramenta *Astah Community*¹.

A classe *Raciocinador* apresentada na Figura 32, é a principal classe do modelo eGroup, nela estão contidos métodos que processam as informações que são recebidas pelo modelo. Essa classe troca mensagens com a classe *ConectorWS* e *ConectorDB*. É considerada a classe inteligente do modelo, é ela quem consulta as base de dados e processa as informações através dos métodos conforme solicitada pelas aplicações.

A classe *ConectorWS* é responsável por conectar o eGroup com às aplicações externas e também aos gerenciadores de trilhas, onde a comunicação ocorre através de *web services* (FIELDING, 2000). Essa classe inicializa um servidor HTTP² que recebe as requisições das aplicações e dos gerenciadores de trilhas. A classe também é responsável por enviar notificações via *HTTP* para as aplicações e gerenciadores de trilhas.

A classe *ConectorDB* possibilita ao *Raciocinador* através de seus métodos efetuar uma conexão ao banco de dados. O perfil, o contexto e a trilha das entidades são recebidos e armazenados no banco de dados *SQL Server 2014* (MICROSOFT, 2015a). O *SQLServer* é um sistema gerenciador de Banco de dados relacional que foi desenvolvido pela *Microsoft*, e foi escolhido por apresentar bom desempenho com as linguagens de programação que foram adotadas para o

¹<http://astah.net/editions/community/>

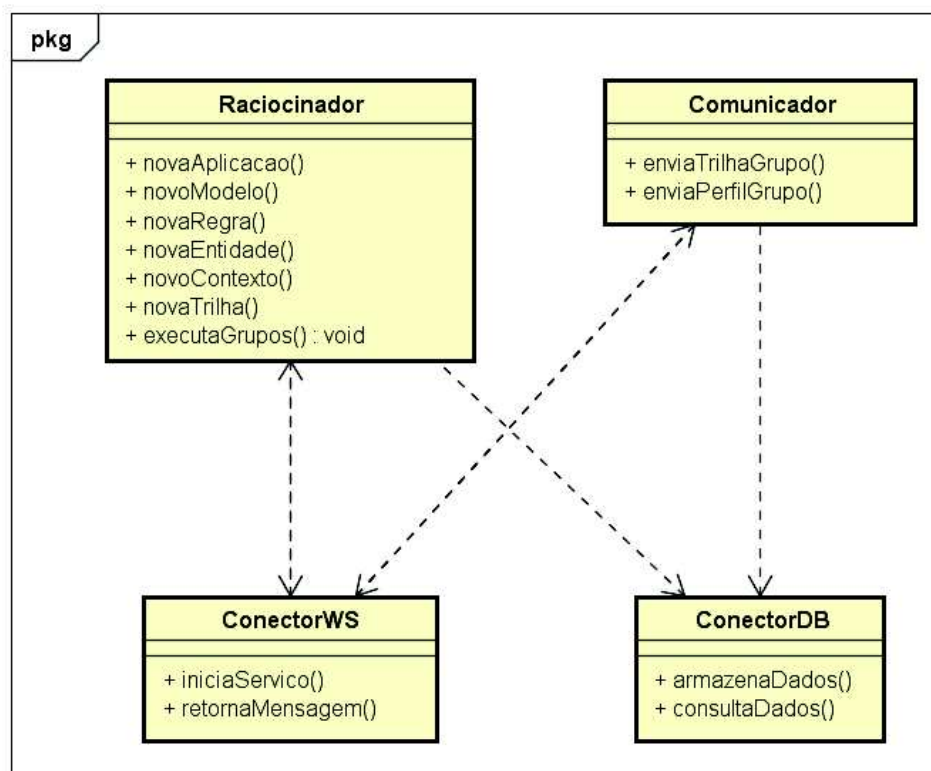
²*HyperText Transfer Protocol* - Protocolo de Transferência de Hipertexto

desenvolvimento do protótipo do eGroup.

A classe *Comunicador* entrega as aplicações os perfis e trilhas dos grupos formados pelo *Raciocinador* e armazenados nas respectivas bases de dados dentro do eGroup.

Para a etapa de codificação das classes apresentadas na Figura 32 foi utilizada a linguagem de programação *CSharp* (MICROSOFT, 2010a) e (MICROSOFT, 2010b) juntamente com um conjunto de tecnologias e ferramentas necessárias para o desenvolvimento do eGroup, como: *VisualStudio 2015* (MICROSOFT, 2015b), *ASP.NET Entity Framework*, *JSON*, *IIS (Internet Information Services)* (MICROSOFT, 2015c), *REST* (FIELDING, 2000), *SQL Server 2014* (MICROSOFT, 2015a).

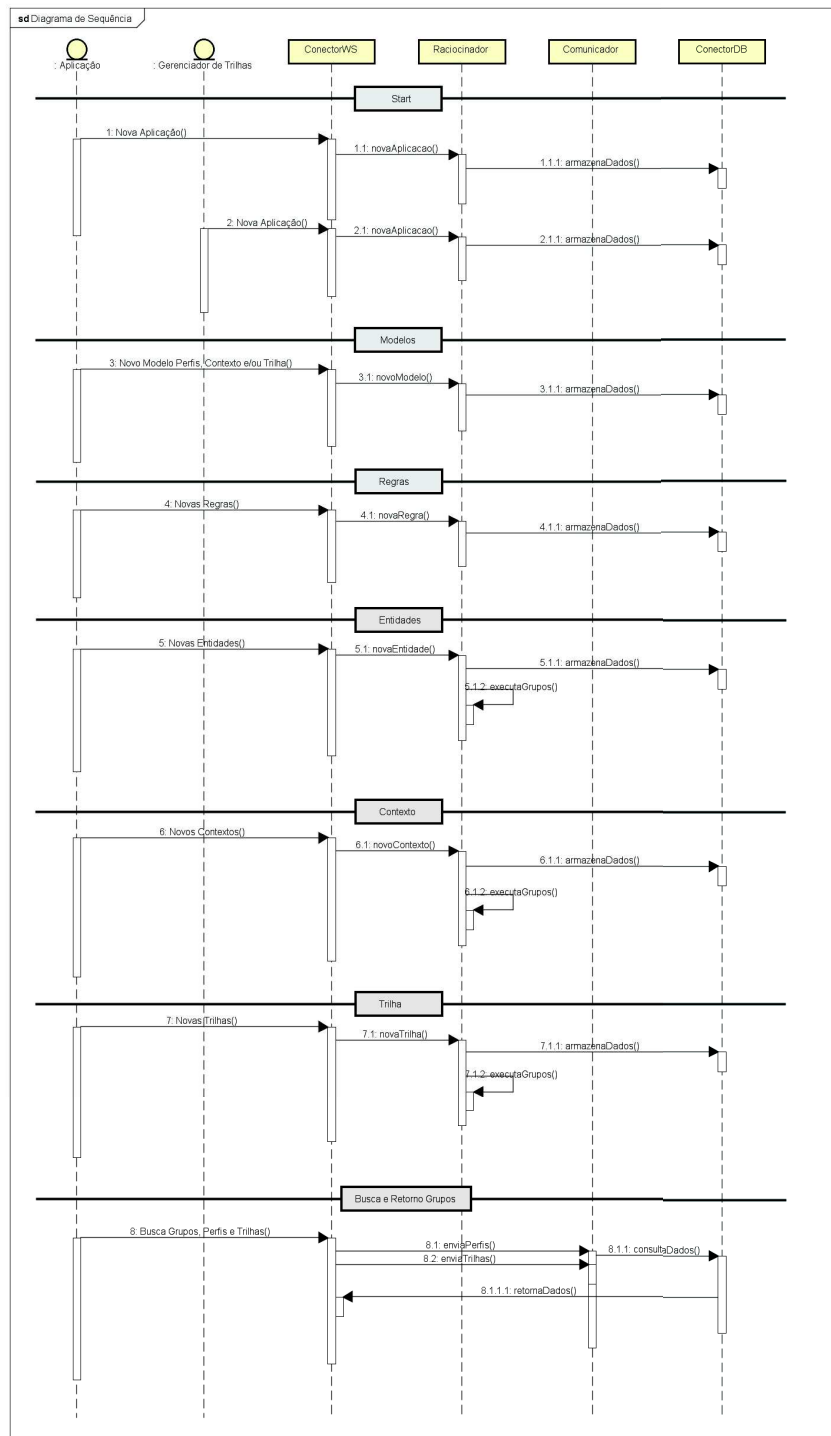
Figura 32: Diagrama de Classes do eGroup



Fonte: Elaborado pelo autor

A Figura 33 apresenta o Diagrama de Sequência do eGroup. Os textos representados pelos números inteiros indicam uma comunicação através de *web services*, enquanto o restante representam chamadas de métodos internos do eGroup. As classes que estão representadas neste diagrama são as mesmas da Figura 32. No Diagrama de Sequência estão representadas as seis principais operações do eGroup.

Figura 33: Diagrama de Sequência do eGroup



Fonte: Elaborado pelo autor

1. **Start:** Ocorre quando uma nova aplicação ou um novo gerenciador de trilhas se conecta com o eGroup. É armazenado uma descrição da aplicação ou gerenciador.
2. **Modelos:** É quando a aplicação define qual é o seu modelo perfil, contexto e/ou trilha. Essas definições podem ocorrer a qualquer momento, mesmo que a aplicação já tenha definido outros modelos anteriormente. O envio de novos modelos podem alterar os já cadastrados ou então excluí-los.
3. **Regras:** A aplicação define quando enviar uma nova regra para a formação dos grupos, esse envio pode ocorrer a qualquer momento após a comunicação inicial, a aplicação não tem a opção de alterar uma regra, caso deseje, deve enviar uma nova regra.
4. **Entidades:** Após os passos anteriores a aplicação pode enviar as entidades, após o recebimento das entidades, é possível que o eGroup execute os métodos para formação dos grupos com base apenas nos perfis das entidades.
5. **Contexto:** A aplicação pode também enviar o contexto de suas entidades conforme a necessidade.
6. **Trilha:** O recebimento das trilhas por parte do eGroup ocorre periodicamente conforme o envio por parte da aplicação.
7. **Buscas e Retorno Grupos:** Nesse passo a aplicação poderá buscar o perfil dos grupos e trilhas atualizado, através do envio de regras para buscar os grupos.

5.2 Aspectos de Avaliação

Esta seção apresenta a metodologia de avaliação aplicada no eGroup. Para avaliar as funcionalidades do sistema, foi definido um cenário para aplicação do eGroup como uma ferramenta para gerenciamento de grupos, principalmente para o agrupamento de entidades. Para auxiliar neste processo foi necessário o desenvolvimento de um protótipo do eGroup, o qual foi descrito na seção 5.1.

A escolha do tipo de avaliação aplicada no eGroup, ocorreu através de uma estratégia de validação por cenários, a qual vem sendo utilizada pela comunidade científica para realização e validação de experimentos em ambientes sensíveis a contexto (DEY, 2001) e ambientes que envolvem aplicações ubíquas (SATYANARAYANAN, 2001).

A avaliação teve como objetivo realizar experimentos da utilização do eGroup em um cenário para gerenciar atividades realizadas por efetivos de exércitos, onde foi desenvolvido uma aplicação web a fim de validar o modelo proposto. Para esse cenário foi considerado o utilização de diferentes perfis de entidades e diferentes contextos das mesmas a fim de analisar o processo de gerenciamento dos grupos pelo eGroup.

Para executar o cenário, foi utilizado um computador pessoal e um com sistema operacional *Windows 10*. Para executar a aplicação, foi utilizando o navegador de internet *Chrome*.

5.2.1 Cenário - Sistema para Gerenciamento de Efetivo de Exércitos

A Figura 34 apresenta quatro telas da aplicação que foi desenvolvida para a avaliação do eGroup, a aplicação foi executada no *browser* para elaboração do cenário. Na Figura 34 (a) é apresentado a tela inicial da aplicação para Efetivos do Exército. Na Figura 34 (b) estão as funções para inserção de entidades (soldados) no caso para população da base de dados da aplicação. Na Figura 34 (c) é apresentado a possibilidade de edição e exclusão de entidades que já estão armazenadas na base de dados da aplicação. Na Figura 34 (d) é ilustrado a comunicação entre a aplicação e o eGroup.

Para o desenvolvimento da aplicação foi utilizado a linguagem *php*, a aplicação foi desenvolvida para testar a comunicação com o eGroup e com isso foi possível popularizar um banco de dados *SQL Server* com inúmeros cadastros de entidades, nesse caso as entidades são soldados. As informações para caracterização de perfil dos soldados foram preenchidas com base em informações retiradas do site do Exército Brasileiro (POSTOS BRASILEIROS, 2016a) e (POSTOS BRASILEIROS, 2016b).

As informações de contexto foram geradas através da utilização de um gerador de contexto implementado em *C++*, que simulou soldados em diferentes regiões do Rio Grande do Sul especificamente para este cenário proposto.

A aplicação desenvolvida não possui um gerenciador de trilhas, portanto não envia ao eGroup os históricos de contextos visitados pelas entidades, mas o eGroup gera os perfis e trilha dos grupos mesmo sem as informações de trilha das entidades.

O botão *start* destacado na Figura 34 (d) é o responsável por iniciar a comunicação entre a aplicação (AppExército) e o eGroup *web services*, nesse caso em específico o botão foi pré-configurado para se comunicar com o *web services IIS*³ rodando em máquina local⁴ e enviar para o eGroup um arquivo no formato *JSON* contendo o *IDMensagem: 1*, a descrição da aplicação que está se comunicando, que neste caso específico é: AppMilitar. O eGroup irá retornar para a aplicação um eGroupID, que é um identificador da aplicação.

Mathematica Código 1: JSON de entrada (START)

```

1 {
2   "IDMensagem" : "1",
3   "descricao" : "AppMilitar"
4 }
```

³*Internet Information Services (IIS)* : é um servidor web criado pela Microsoft para seus sistemas operacionais. <http://www.iis.net/>

⁴URL : <http://192.168.1.100:1951/api/eGroup/>

O botão Modelos de perfis, contexto e trilhas destacado na Figura 34 (d) é o responsável por enviar os modelos com as informações que a aplicação considera como perfis das entidades (Código Código 2), as informações de contexto (Código Código 3) e caso a aplicação armazene a trilha das entidades.

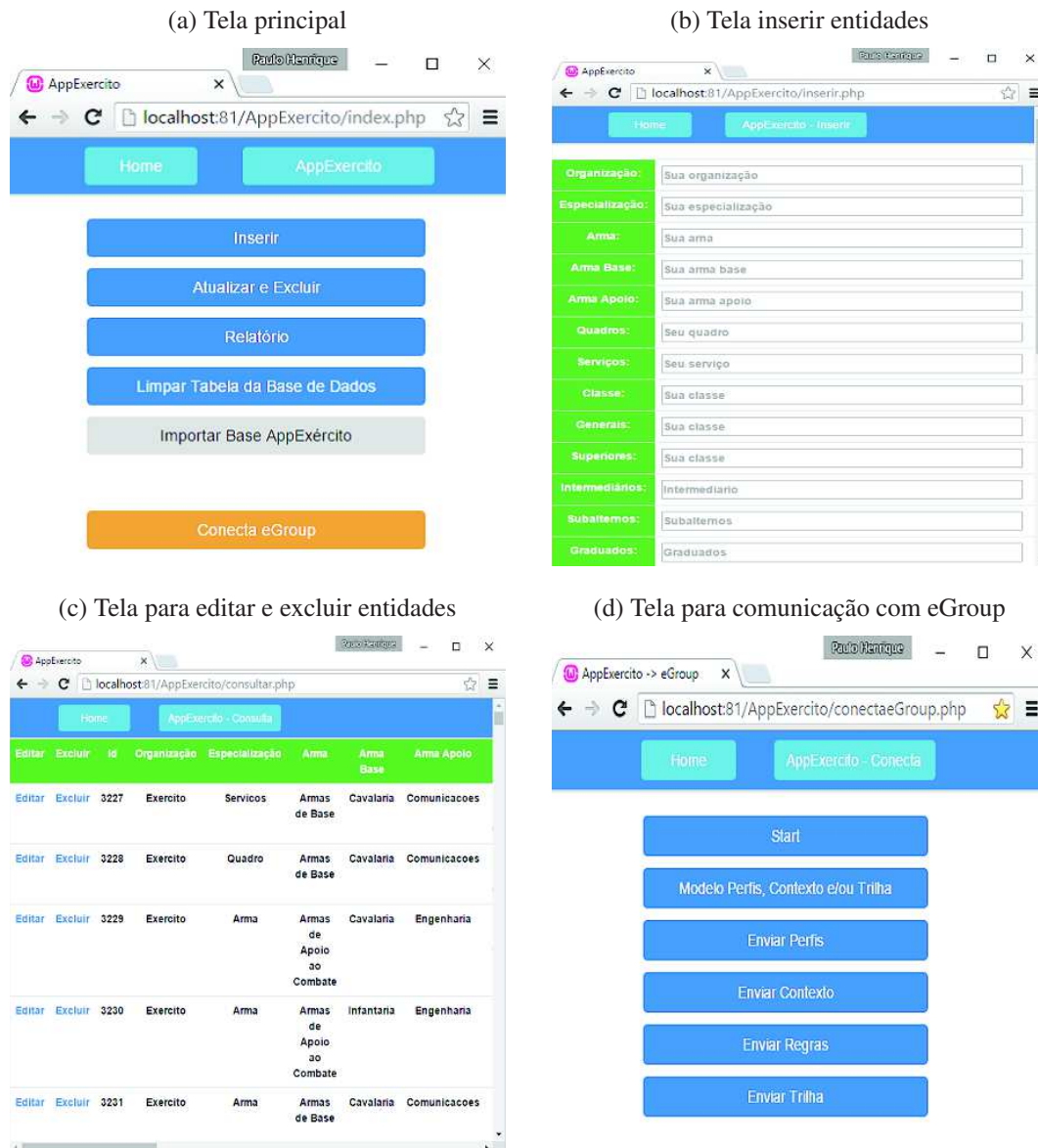
Mathematica Código 2: JSON com os modelo de perfis das entidades

```
1 {
2   "IDMensagem" : "2",
3     "eGroupId" : "1003",
4     "organizacao" : "string",
5     "especializacao" : "string",
6     "arma" : "string",
7     "arma_base" : "string",
8     "arma_apoio" : "string",
9     "quadros" : "string",
10    "servicos" : "string",
11    "classe" : "string",
12    "gerais" : "string",
13    "superiores" : "string",
14    "intermediarios" : "string",
15    "subalternos" : "string",
16    "graduados" : "string",
17    "nome" : "string",
18    "idade" : "int",
19    "peso" : "string",
20    "altura" : "string",
21    "sangue" : "string",
22    "filhos" : "string",
23    "naturalidade" : "string",
24    "sexo" : "string",
25    "estado_civil" : "string",
26    "pele" : "string"
27 }
```

Mathematica Código 3: JSON com os modelo de contexto das entidades

```
1 {
2   "IDMensagem" : "2",
3     "eGroupId" : "1003",
4     "Localizacao" : "string"
5 }
```

Na sequência a aplicação envia os perfis das entidades, esses perfis estão armazenados em

Figura 34: Telas da *aplicação web* para Efetivos do Exército sendo executadas no navegador

um banco de dados *sql* e foram convertidos para formato *JSON* e enviados para o eGroup. No Código Código 4 é possível identificar o *IDMensagem: 4* e o mesmo eGroupID dos códigos anteriores, que define qual é a aplicação que está comunicando com o eGroup. Abaixo disso, a partir da linha 6, é possível visualizar as informações que caracterizam os perfis das entidades neste cenário.

Mathematica Código 4: JSON com os perfis das entidades

```

1 {
2   "IDMensagem" : "4",
3   "eGroupId" : "1003",
4 }
5 {

```



```
6      "id" : "0001",
7      "organizacao" : "Exercito",
8      "especializacao" : "Arma",
9      "arma" : "Armas de Apoio ao Combate",
10     "arma_base" : "Cavalaria",
11     "arma_apoio" : "Engenharia",
12     "quadros" : "Quadro Auxiliar Oficiais (QAO)",
13     "servicos" : "Assistencia Religiosa",
14     "classe" : "Oficiais Subalternos",
15     "generais" : "Marechal",
16     "superiores" : "Major",
17     "intermediarios" : "Capitao",
18     "subalternos" : "Segundo Tenente",
19     "graduados" : "Taifeiro-Mor",
20     "nome" : "Barbosa",
21     "idade" : "30",
22     "peso" : "100",
23     "altura" : "2.22",
24     "sangue" : "o-",
25     "filhos" : "Sim",
26     "naturalidade" : "Viamao",
27     "sexo" : "Feminino",
28     "estado_civil" : "Casado",
29     "pele" : "Clara"
30 },
31 {
32     "id" : "0002",
33     "organizacao" : "Exercito",
34     "especializacao" : "Quadro",
35     "arma" : "Armas de Base",
36     "arma_base" : "Cavalaria",
37     "arma_apoio" : "Comunicacoes",
38     "quadros" : "Quadro Auxiliar Oficiais (QAO)",
39     "servicos" : "Saude",
40     "classe" : "Oficiais Graduados",
41     "generais" : "General Exercito",
42     "superiores" : "Tenente Coronel",
43     "intermediarios" : "Capitao",
44     "subalternos" : "Segundo Tenente",
45     "graduados" : "Taifeiro-Mor",
46     "nome" : "Geraldo",
```

```

47     "idade" : "29",
48     "peso" : "107.8",
49     "altura" : "2.29",
50     "sangue" : "o-",
51     "filhos" : "Sim",
52     "naturalidade" : "Porto Alegre",
53     "sexo" : "Feminino",
54     "estado_civil" : "Companheiro",
55     "pele" : "Negra"
56 }

```

Após o envio dessas informações para o eGroup através da comunicação o mesmo já suporta o recebimento de regras para a formação de grupos com base nos perfis das entidades. Caso a aplicação envolva um cenário ubíquo, onde armazene o contexto das entidades, pode enviar estas informações caracterizadas. É justamente esse o próximo passo executado por este cenário, onde a aplicação envia ao eGroup as informações de contexto que ela considera e armazena, que nesse caso é apenas a localização de forma fixa, conforme apresenta o Código Código 5.

Mathematica Código 5: JSON com o contexto das entidades

```

1 {
2     "IDMensagem" : "5",
3     "eGroupId" : "1003",
4 }
5 {
6     "id" : "0001",
7     "localizacao" : "Santa Maria",
8 },
9 {
10     "id" : "0002",
11     "localizacao" : "Porto Alegre",
12 }

```

Dado o envio destas informações, o cenário proposto é o seguinte: “*Foram enviadas 200 entidades com as respectivas informações exemplificada no Código Código 4, com as informações de contexto exemplificada no Código Código 5. Com base nestas informações elaborou-se um cenário voltado ao efetivo de exércitos, onde era preciso a formação de dois grupos com regras específicas em cada um dos grupos. O primeiro grupo (conforme Figura 36a) deveria ser formado por soldados que fossem da infantaria ou cavalaria, com idade entre 20 e 40 anos, estado civil solteiro pois a missão é de alto risco e que estivessem na região de Santa Maria-RS. O segundo grupo (conforme Figura 36b) deveria ser formado por soldados que fossem da artilharia ou comunicações, com idade entre 20 e 45 anos, que pesassem mais que 100kg e que estivessem na região de Porto Alegre-RS. Para embasamento deste cenário será preciso a*

aplicação enviar ao eGroup as regras (conforme Figura 35) para formação destes grupos.

Figura 35: Regra para primeiro e segundo grupo

Regra 1
 5#arma_base=infantaria#|arma_base=cavalaria#&idade>=20#&idade<=40#&localizacao=Santa Maria

Regra 2
 5#arma_base=artilharia#|arma_base=comunicacoes#&idade>=20#&idade<=45#&localizacao=Porto Alegre

Fonte: Elaborado pelo autor

Figura 36: Telas da base de dados com id das entidades agrupadas

(a) Primeiro Grupo		(b) Segundo Grupo	
PAULO\SQLEXPRESS...eracaoSantaMaria	Id	PAULO\SQLEXPRESS...eracaoPortoAlegre X	Id
	0008		0011
	0020		0031
	0026		0051
	0053		0059
	0063		0069
	0103		0079
	0110		0109
	0137		0118
	0138		0122
	0146		0125
	0153		0126
	0154		0141
	0159		0143
	0161		0151
	0167		0168
	0176		0194
	0189		
	0192		
	0197		

O Código Código 6 apresenta o perfil do grupo gerado com base na formação do primeiro grupo e da Regra 1. O Código Código 7 apresenta o perfil do grupo gerado com base na formação do segundo grupo e da Regra 2.

Mathematica Código 6: JSON com o perfil do primeiro grupo

```

1 {
2   "idRegra" : "0001",
3   "idEntidades" : "0008, 0020, 0026, 0053, 0063, 0103, 0110, 01
4   37, 0138, 0146, 0153, 0154, 0159, 0161, 0167, 0176, 0189,
   0192, 0197",
   "arma base" : "infantaria",

```

```

5     "logica" : "ou",
6     "arma base" : "cavalaria",
7     "logica" : "e",
8     "idade" : "20",
9     "logica" : "entre",
10    "idade" : "40",
11    "logica" : "e",
12    "localizacao" : "Santa Maria"
13 }

```

Mathematica Código 7: JSON com o perfil do primeiro grupo

```

1 {
2     "idRegra" : "0002",
3     "idEntidades" : "0011, 0031, 0051, 0059, 0069, 0079, 0109, 01
4         18, 0122, 0125, 0126, 0141, 0143, 0151, 0168, 0194",
5     "arma base" : "artilharia",
6     "logica" : "ou",
7     "arma base" : "comunicacoes",
8     "logica" : "e",
9     "idade" : "20",
10    "logica" : "entre",
11    "idade" : "45",
12    "logica" : "e",
13    "localizacao" : "Porto Alegre"

```

O Código Código 8 apresenta a trilha do grupo que foi gerada com base na formação do primeiro grupo e da Regra 1. O Código Código 9 apresenta a trilha do grupo que foi gerada com base na formação do segundo grupo e da Regra 2.

Mathematica Código 8: JSON com a trilha do grupo gerado com base no primeiro grupo

```

1 [{
2     "idGrupo" : "0001",
3     "idRegra" : "0001",
4     "arma base" : "infantaria",
5     "logica" : "ou",
6     "arma base" : "cavalaria",
7     "logica" : "e",
8     "idade" : "20",
9     "logica" : "entre",
10    "idade" : "40",
11    "logica" : "e",

```

```

12     "localizacao" : "Santa Maria"
13  }}
14  [{
15     "idEntidades" : "0008, 0020, 0026, 0053, 0063, 0103, 0110, 01
        37, 0138, 0146, 0153, 0154, 0159, 0161, 0167, 0176, 0189,
        0192, 0197",
16     "data" : "10/08/2016",
17     "hora" : "16:00",
18     "tamanho" : "19"
19  }}

```

Mathematica Código 9: JSON com a trilha do grupo gerado com base no segundo grupo

```

1  [{
2     "idGrupo" : "0002",
3     "idRegra" : "0002",
4     "arma base" : "artilharia",
5     "logica" : "ou",
6     "arma base" : "comunicacoes",
7     "logica" : "e",
8     "idade" : "20",
9     "logica" : "entre",
10    "idade" : "45",
11    "logica" : "e",
12    "localizacao" : "Porto Alegre"
13  }}
14  [{
15     "idEntidades" : "0011, 0031, 0051, 0059, 0069, 0079, 0109, 01
        18, 0122, 0125, 0126, 0141, 0143, 0151, 0168, 0194",
16     "data" : "10/08/2016",
17     "hora" : "17:00",
18     "tamanho" : "16"
19  },
20  {
21     "idEntidades" : "0011, 0031, 0051, 0059",
22     "data" : "10/08/2016",
23     "hora" : "18:00",
24     "tamanho" : "4"
25  }
26  ]

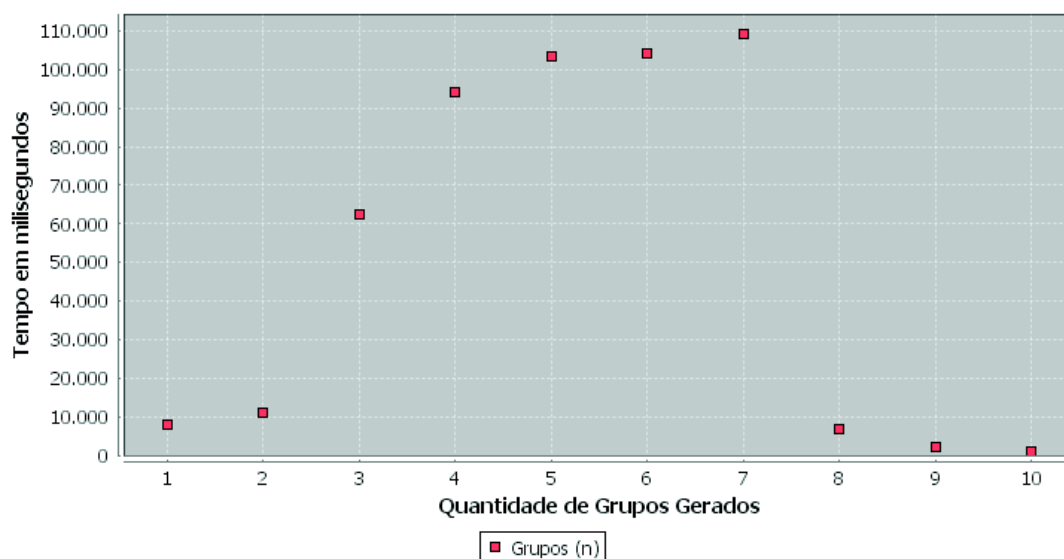
```

5.2.2 Avaliação do desempenho da criação de grupos

A execução desse cenário mostrou a viabilidade de um modelo genérico para o gerenciamento de grupos, envolvendo a formação de grupos com base em características de perfis e informações de contexto das entidades, através de regras genéricas. Além de considerar o uso de histórico de contextos (trilhas). Embora o eGroup tenha sido integrado apenas com a AppE-xército, sua proposta é genérica e pode ser utilizada por qualquer aplicação ou gerenciador de trilha.

Com a finalidade de verificar o tempo de execução e resposta do eGroup para a geração dos grupos com o aumento de entidades, foram coletados dados através de sua execução. Com os dados coletados, foi gerado um gráfico (Figura 37).

Figura 37: Tempo de resposta para geração dos grupos com aumento de entidades



Fonte: Elaborado pelo autor

No gráfico mostrado na Figura 37, é possível perceber que foram gerados 10 grupos, sendo cada grupo com quantidades de entidades diferentes: Grupo 1 (100 entidades), Grupo 2 (200 entidades), Grupo 3 (1000 entidades), Grupo 4 (2000 entidades), Grupo 5 (3000 entidades), Grupo 6 (4000 entidades), Grupo 7 (5000 entidades), Grupo 8 (800 entidades), Grupo 9 (50 entidades) e Grupo 10 (10 entidades).

O maior tempo para geração dos grupos foi de 109042ms, já o menor tempo obtido foi 990ms, com um tempo médio de 50244ms.

6 CONSIDERAÇÕES FINAIS

Este capítulo apresenta as considerações finais da dissertação. A seção 6.1 apresenta as principais conclusões e contribuições do eGroup. Por fim, é apresentado os possíveis trabalhos futuros.

6.1 Conclusões

Nos capítulos iniciais foram apresentados os conceitos da Computação Ubíqua, a sensibilidade a contexto, a importância dos perfis das entidades, o armazenamento de históricos de contextos visitados pelas entidades, conhecido como trilha e por fim foram apresentados os aspectos que envolvem o gerenciamento de grupos em suas diversas aplicabilidades.

No capítulo 3 foram apresentados primeiramente os critérios para seleção dos trabalhos relacionados, após essa etapa foram descritos e apresentadas as principais arquiteturas e modelos encontrados dentro da área de pesquisa, por fim foi apresentado uma tabela comparativa entre os trabalhos analisados e o eGroup.

No capítulo 4 foi apresentado o modelo proposto, onde foi apresentado inicialmente uma visão geral da utilização do eGroup e logo depois foi detalhado a arquitetura do mesmo. Nas seções seguintes, foi detalhado como ocorre a comunicação entre uma aplicação e o eGroup, além de especificar modelos de perfis, contexto e trilha.

No capítulo 5 foi apresentado os aspectos de implementação e avaliação do eGroup, na primeira parte do capítulo foi apresentado o diagrama de classe e de sequência do eGroup, e na sequência foi apresentado um cenário para avaliação do modelo proposto.

Com a realização do cenário apresentado foram alcançadas as seguintes conclusões nessa dissertação:

- O desenvolvimento do eGroup mostrou a viabilidade da criação de um modelo genérico para o gerenciamento de grupos;
- A utilização de contexto segundo Dey (2001) permitiu uma maior abrangência de informações com relação a entidade, além de possibilitar o agrupamento de entidades com base em informações relevantes, como a localização por exemplo. Além de possibilitar a integração do modelo com aplicações ubíquas;
- O recebimento e uso de histórico de contexto visitados pelas entidades para o gerenciamento de grupos com base em informações passadas tornou-se um dos diferenciais para esta dissertação.
- A integração do eGroup com qualquer aplicação permitiu que fosse constatado que o modelo suporta diferentes aplicações relacionadas ao gerenciamento de grupos, validando a proposta de ser um modelo genérico;

- Com o primeiro cenário cenário foi possível validar que o uso de perfis e contexto das entidades podem ser relevantes para o gerenciamento de grupos;
- O termo gestão para grupos genéricos, conclui-se que o eGroup não consegue validar sem possuir um agente ou uma ontologia para grupos.

6.2 Trabalhos Futuros

Descrever e validar um segundo cenário com o eGroup, este cenário é a utilização do eGroup em um ambiente de pessoas com deficiência (PcDs), onde será gerado a integração do eGroup com o *MASC (Model for Assistive Smart Cities)* (TELLES, 2016), o qual possui um gerenciador de trilhas que armazena históricos de diferentes PcDs, o modelo *MASC* também é genérico pois suporta diferentes tipos de deficiências.

A criação de uma ontologia para grupos para a padronização das entidades e representação do conhecimento através da mesma, possibilitaria uma maior organização quanto a aspectos de documentação do modelo eGroup.

A evolução dos grupos poderia ser uma questão específica para se trabalhar a partir deste trabalho, como evoluir um grupo genérico. O Raciocinador precisa de uma regra para acompanhar os grupos devido ao trabalho ser genérico.

REFERÊNCIAS

- ABOWD, G. D.; DEY, A. K.; BROWN, P. J.; DAVIES, N.; SMITH, M.; STEGGLES, P. Towards a Better Understanding of Context and Context-Awareness. In: INTERNATIONAL SYMPOSIUM ON HANDHELD AND UBIQUITOUS COMPUTING, 1., 1999, London, UK, UK. **Proceedings...** Springer-Verlag, 1999. p. 304–307.
- ANI, Z. C.; YASIN, A.; HUSIN, M. Z.; HAMID, Z. A. A method for group formation using genetic algorithm. **International Journal of Computer Science Engineering (IJCSE)**, Uum Sintok, Kedah, Malaysia, v. 2, n. 09, p. 3060–3064, 2010.
- ARAÚJO, R. B. de. **Computação ubíqua: princípios, tecnologias e desafios**. Natal, CE, Brasil: Sociedade Brasileira de Computação, 2003. 11–13 p. v. 8.
- BACKSTROM, L.; HUTTENLOCHER, D.; KLEINBERG, J.; LAN, X. Group Formation in Large Social Networks: membership, growth, and evolution. In: ACM SIGKDD INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING, 12., 2006, Philadelphia, PA, USA. **Proceedings...** ACM, 2006. p. 44–54. (KDD '06).
- BAIS, J.; GILETTO, J.; BALMACEDA, J.; SCHIAFFINO, S.; DÍAZ, J. A. An assistant for group formation in CSCL based on constraint satisfaction. **Proceedings of ASAI**, Argentina, p. 12–23, 2013.
- 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, n. 1, p. 62–83, May 2011.
- BORATTO, L.; CARTA, S.; SATTÀ, M. Groups identification and individual recommendations in group recommendation algorithms. In: WORKSHOP ON THE PRACTICAL USE OF RECOMMENDER SYSTEMS, ALGORITHMS AND TECHNOLOGIES (PRSAT), HELD IN CONJUNCTION WITH RECSYS, 2010, Barcelona, Spain. **Anais...** Workshop on the Practical Use of Recommender Systems: Algorithms and Technologies (PRSAT): Held in Conjunction with RecSys, 2010. p. 27–34.
- BUZETO, F.; CAPRETZ, M.; CASTANHO, C.; JACOBI, R. uOS: a resource rerouting middleware for ubiquitous games. In: UBIQUITOUS INTELLIGENCE AND COMPUTING, 2013 IEEE 10TH INTERNATIONAL CONFERENCE ON AND 10TH INTERNATIONAL CONFERENCE ON AUTONOMIC AND TRUSTED COMPUTING (UIC/ATC), 2013, Vietri sul Mare. **Anais...** IEEE, 2013. p. 88–95.
- CHRISTODOULOPOULOS, C.; PAPANIKOLAOU, K. A Group Formation Tool in an E-Learning Context. In: TOOLS WITH ARTIFICIAL INTELLIGENCE, 2007. ICTAI 2007. 19TH IEEE INTERNATIONAL CONFERENCE ON, 2007. **Anais...** [S.l.: s.n.], 2007. v. 2, p. 117–123.
- COSTA, C. A. da. **Software Infrastructure for Ubiquitous Computing: a context-aware service-based approach**. Saarbrücken, Germany, Germany: VDM Verlag, 2009.
- COSTA, C. da; YAMIN, A.; GEYER, C. Toward a General Software Infrastructure for Ubiquitous Computing. **Pervasive Computing, IEEE**, [S.l.], v. 7, n. 1, p. 64–73, Jan 2008.

COUTAND, O.; DROEGEHORN, O.; DAVID, K.; NURMI, P.; FLORÉEN, P.; KERNCHEN, R.; HOLTMANNS, S.; CAMPADELLO, S.; KANTER, T.; MARTIN, M. et al. Context-aware group management in mobile environments. **Dresden, Germany: IST Mobile and Wireless Communications Summit**, [S.l.], 2005.

CRESPO, S. "**Composição em WebFrameworks.**" **M.S. thesis**. 2000. Tese (Doutorado em Ciência da Computação) — Pontifícia Universidade Católica do Rio de Janeiro Rio de Janeiro, 2000.

CUDAK, M.; TAIB, R. B. F. **Group application for group formation and management**. US Patent 7,231,206.

CUNHA, L. M.; FUKS, H.; LUCENA, C. Formação de Grupos no Ambiente AulaNet Utilizando Agentes de Software. **XIII Simpósio Brasileiro de Informática na Educação-SBIE**, [S.l.], v. 12, 2002.

DA FONTOURA, M. F. M. C. **A systematic approach to framework development**. 1999. Tese (Doutorado em Ciência da Computação) — Pontifical Catholic University of Rio de Janeiro Rio de Janeiro, 1999.

DAMASCENO VIANNA, H.; BARBOSA, J. A Model for Ubiquitous Care of Noncommunicable Diseases. **Biomedical and Health Informatics, IEEE Journal of**, [S.l.], v. 18, n. 5, p. 1597–1606, Sept 2014.

DE FARIA, E. S. J.; ADÁN-COELLO, J. M.; YAMANAKA, K. Forming groups for collaborative learning in introductory computer programming courses based on students' programming styles: an empirical study. In: FRONTIERS IN EDUCATION CONFERENCE, 36TH ANNUAL, 2006. **Anais...** [S.l.: s.n.], 2006. p. 6–11.

DEY, A. K. Understanding and Using Context. **Personal Ubiquitous Comput.**, London, UK, UK, v. 5, n. 1, p. 4–7, Jan. 2001.

DEY, A. K.; ABOWD, G. D.; SALBER, D. A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-aware Applications. **Hum.-Comput. Interact.**, Hillsdale, NJ, USA, v. 16, n. 2, p. 97–166, Dec. 2001.

DRIVER, C.; CLARKE, S. Hermes: a software framework for mobile, context-aware trails. In: WORKSHOP ON COMPUTER SUPPORT FOR HUMAN TASKS AND ACTIVITIES AT PERVASIVE, VIENNA, AUSTRIA, 2004. **Anais...** [S.l.: s.n.], 2004.

DRIVER, C.; CLARKE, S. An application framework for mobile, context-aware trails. **Pervasive and Mobile Computing**, [S.l.], v. 4, n. 5, p. 719–736, 2008.

FELIX, Z. C.; TEDESCO, P. A. Smart Chat Group: ferramenta ciente de contexto para formaçãp de grupos-versão final. **SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO-SBIE**, [S.l.], v. 19, 2008.

FERREIRA, L. G. A.; BARBOSA, J. L. V.; GLUZ, J. C. Um Modelo de Recomendação Ubíqua de Conteúdo para Grupos de Aprendizes. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 2013. **Anais...** [S.l.: s.n.], 2013. v. 24, n. 1.

FIELDING, R. T. **Architectural Styles and the Design of Network-based Software Architectures**. 2000. Tese (Doutorado em Ciência da Computação) — University of California, Irvine, 2000. AAI9980887.

- FORSYTH, D. **Group dynamics**. [S.l.]: Thomson/Wadsworth, 2006.
- 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.
- FREITAS, A. A. de; DEY, A. K. Using Multiple Contexts to Detect and Form Opportunistic Groups. In: ACM CONFERENCE ON COMPUTER SUPPORTED COOPERATIVE WORK SOCIAL COMPUTING, 18., 2015, New York, NY, USA. **Proceedings...** ACM, 2015. p. 1612–1621. (CSCW '15).
- FREITAS, A. A. de; DEY, A. K. The Group Context Framework: an extensible toolkit for opportunistic grouping and collaboration. In: ACM CONFERENCE ON COMPUTER SUPPORTED COOPERATIVE WORK & SOCIAL COMPUTING, 18., 2015, New York, NY, USA. **Proceedings...** ACM, 2015. p. 1602–1611. (CSCW '15).
- HALLBERG, J.; NORBERG, M. B.; KRISTIANSSON, J.; SYNNESE, K.; NUGENT, C. Creating Dynamic Groups Using Context-awareness. In: INTERNATIONAL CONFERENCE ON MOBILE AND UBIQUITOUS MULTIMEDIA, 6., 2007, New York, NY, USA. **Proceedings...** ACM, 2007. p. 42–49. (MUM '07).
- HANSMANN, U.; MERK, L.; NICKLOUS, M.; STOBER, T. **Pervasive Computing**. [S.l.]: Springer, 2003. (Springer Professional Computing).
- HIGHTOWER, J.; BORRIELLO, G. Location systems for ubiquitous computing. **Computer**, [S.l.], v. 34, n. 8, p. 57–66, 2001.
- JULIEN, C. The Context of Coordinating Groups in Dynamic Mobile Networks. In: INTERNATIONAL CONFERENCE ON COORDINATION MODELS AND LANGUAGES, 13., 2011, Berlin, Heidelberg. **Proceedings...** Springer-Verlag, 2011. p. 49–64. (COORDINATION'11).
- KYPRIANIDOU, M.; DEMETRIADIS, S.; TSIATSOS, T.; POMBORTSIS, A. Group formation based on learning styles: can it improve students' teamwork? **Educational Technology Research and Development**, [S.l.], v. 60, n. 1, p. 83–110, 2012.
- LARMAN, C. **Utilizando UML e padrões**. [S.l.]: Bookman Editora, 2002.
- LAWRIE, G. A.; MATTHEWS, K. E.; GAHAN, L. R. Forming groups to foster collaborative learning in large enrolment courses. In: THE AUSTRALIAN CONFERENCE ON SCIENCE AND MATHEMATICS EDUCATION (FORMERLY UNISERVE SCIENCE CONFERENCE), 2010. **Proceedings...** [S.l.: s.n.], 2010. v. 16.
- LIN, Y.-T.; HUANG, Y.-M.; CHENG, S.-C. An automatic group composition system for composing collaborative learning groups using enhanced particle swarm optimization. **Computers Education**, [S.l.], v. 55, n. 4, p. 1483 – 1493, 2010.
- LOKE, S. **Context-Aware Pervasive Systems: architectures for a new breed of applications**. [S.l.]: CRC Press, 2006.
- LOWE, D.; MUJKANOVIC, A.; GUETL, C.; KOSTULSKI, T. **An architecture for automated group formation within remote laboratories**. [S.l.]: International Association of Online Engineering, 2011.

MEDINA, R. D.; GÓMEZ-PÉREZ, D.; NIETO-REYES, A.; SANTOS, C. B. A Method to Form Learners Groups in Computer-supported Collaborative Learning Systems. In: FIRST INTERNATIONAL CONFERENCE ON TECHNOLOGICAL ECOSYSTEM FOR ENHANCING MULTICULTURALITY, 2013, Salamanca, Spain. **Proceedings...** ACM, 2013. p. 261–266. (TEEM '13).

MICROSOFT. **Programming in C with microsoft visual studio 2010 Volume 1.** [S.l.]: Microsoft Official Course, 2010.

MICROSOFT. **Programming in C with microsoft visual studio 2010 Volume 2.** [S.l.]: Microsoft Official Course, 2010.

MICROSOFT. Microsoft SQL Server 2014 Express, Disponível em: <<https://www.microsoft.com/pt-br/download/details.aspx?id=42299/>>. Acesso em: 15 novembro 2015.

MICROSOFT. Visual Studio, Disponível em: <<https://www.visualstudio.com/pt-br/downloads/download-visual-studio-vs.aspx>>. Acesso em: 15 novembro 2015.

MICROSOFT. Internet Information Services (IIS), Disponível em: <<http://www.iis.net/>>. Acesso em: 15 novembro 2015.

MONTEIRO, J. **Google e Android - Crie aplicações para celulares e tablets.** [S.l.]: Casa do Código, 2013.

MUEHLENBROCK, M. Formation of Learning Groups by using Learner Profiles and Context Information. In: AIED, 2005. **Anais...** IOS Press, 2005. p. 507–514. (Frontiers in Artificial Intelligence and Applications, v. 125).

MUJKANOVIC, A.; LOWE, D.; WILLEY, K.; GUETL, C. Unsupervised learning algorithm for adaptive group formation: collaborative learning support in remotely accessible laboratories. In: INFORMATION SOCIETY (I-SOCIETY), 2012 INTERNATIONAL CONFERENCE ON, 2012. **Anais...** [S.l.: s.n.], 2012. p. 50–57.

MULVENNA, M. D.; ANAND, S. S.; BüCHNER, A. G. Personalization on the Net using Web mining: introduction. **Communications of the ACM**, [S.l.], v. 43, n. 8, p. 122–125, 2000.

OH, J.; OH, S. Some aspects of the ubiquitous services on the u-City implementation. In: MOBILE IT CONVERGENCE (ICMIC), 2011 INTERNATIONAL CONFERENCE ON, 2011. **Anais...** [S.l.: s.n.], 2011. p. 78–81.

OUNNAS, A.; DAVIS, H. C.; MILLARD, D. E. A framework for semantic group formation in education. **Journal of Educational Technology & Society**, [S.l.], v. 12, n. 4, p. 43–55, 2009.

POSTOS Brasileiros. Postos e Graduações do Exército Brasileiro, Disponível em: <http://www.eb.mil.br/postos-e-graduacoes/-/asset_publisher/DQlwhsMH8YR7/content/exercito?redirect=http%3A%2F%2Fwww.eb.mil.br%2Fpostos-e-graduacoes%3Fp_p_id%3D101_INSTANCE_DQlwhsMH8YR7%26p_p_lifecycle%3D0%26p_p_state%3Dnormal%26p_p_mode%3Dview%26p_p_col_id%3Dcolumn-1%26p_p_col_count%3D1>. Acesso em: 04 abril 2016.

- POSTOS Brasileiros. Postos e Graduações do Exército Brasileiro, Disponível em: <<http://www.eb.mil.br/armas-quadros-e-servicos>>. Acesso em: 04 abril 2016.
- RANA, J.; KRISTIANSSEN, J.; SYNNE, K. Supporting Ubiquitous Interaction in Dynamic Shared Spaces through Automatic Group Formation Based on Social Context. In: SOCIAL INFORMATICS (SOCIALINFORMATICS), 2012 INTERNATIONAL CONFERENCE ON, 2012. **Anais...** [S.l.: s.n.], 2012. p. 121–130.
- SANCHO-ASENSIO, A.; SOLE, X.; MONTERO, J.; NAVARRO, J.; CANALETA, X.; VERNET, D. Support tool for the formation of working groups in collaborative learning environments. In: INFORMATION SYSTEMS AND TECHNOLOGIES (CISTI), 2014 9TH IBERIAN CONFERENCE ON, 2014. **Anais...** [S.l.: s.n.], 2014. p. 1–6.
- SATYANARAYANAN, M. Pervasive computing: vision and challenges. **Personal Communications, IEEE**, [S.l.], v. 8, n. 4, p. 10–17, 2001.
- SCHILIT, B.; ADAMS, N.; WANT, R. Context-Aware Computing Applications. In: FIRST WORKSHOP ON MOBILE COMPUTING SYSTEMS AND APPLICATIONS, 1994., 1994, Washington, DC, USA. **Proceedings...** IEEE Computer Society, 1994. p. 85–90. (WMCSA '94).
- SILVA, J. M.; ROSA, J. H.; BARBOSA, J. L.; BARBOSA, D. N.; PALAZZO, L. A. Content distribution in trail-aware environments. **Journal of the Brazilian Computer Society**, [S.l.], v. 16, n. 3, p. 163–176, 2010.
- SILVEIRA, S. R.; BARONE, D. A. C. Formação de grupos colaborativos em cursos a distância via web: um estudo de caso utilizando técnicas de inteligência artificial. **Revista brasileira de informática na educação. Florianópolis. Vol. 14, n. 2 (maio-ago. 2006), p. 29-40**, [S.l.], 2006.
- STALLJOHANN, P.; HACKELOER, A. A rule-based group management tool - supporting organizational processes for collaborative learning activities in higher education. In: INTERACTIVE COLLABORATIVE LEARNING (ICL), 2011 14TH INTERNATIONAL CONFERENCE ON, 2011. **Anais...** [S.l.: s.n.], 2011. p. 149–153.
- TAVARES, J.; BARBOSA, J.; COSTA, C.; YAMIN, A.; REAL, R. Hefestos: a model for ubiquitous accessibility support. In: INTERNATIONAL CONFERENCE ON PERVASIVE TECHNOLOGIES RELATED TO ASSISTIVE ENVIRONMENTS, 5., 2012, Heraklion, Crete, Greece. **Proceedings...** ACM, 2012. p. 27:1–27:8. (PETRA '12).
- TELLES, M. J. **Masc**: um modelo computacional para cidades inteligentes assistivas. 2016. Dissertação (Mestrado em Ciência da Computação) — Universidade do Vale do Rio dos Sinos, São Leopoldo, RS, 2016.
- UML. Unified Modeling Language, Disponível em: <www.uml.org>. Acesso em: 04 abril 2016.
- VIVIANI, M.; BENNANI, N.; EGYED-ZSIGMOND, E. A Survey on User Modeling in Multi-application Environments. In: THIRD INTERNATIONAL CONFERENCE ON ADVANCES IN HUMAN-ORIENTED AND PERSONALIZED MECHANISMS, TECHNOLOGIES AND SERVICES, 2010., 2010, Washington, DC, USA. **Proceedings...** IEEE Computer Society, 2010. p. 111–116. (Centric '10).

WAGNER, A. **Um Modelo Para o Gerenciamento de Perfis de Entidades Através de Inferência em Trilhas**. 2013. Dissertação (Mestrado em Ciência da Computação) — Universidade do Vale do Rio dos Sinos, São Leopoldo, RS, 2013.

WAGNER, A.; BARBOSA, J. L. V.; BARBOSA, D. N. F. A model for profile management applied to ubiquitous learning environments. **Expert Systems with Applications**, [S.l.], v. 41, n. 4, p. 2023–2034, 2014.

WEBBER, C. G.; LIMA, M. d. F. W. d. P. Evaluating automatic group formation mechanisms to promote collaborative learning—a case study. **International Journal of Learning Technology**, [S.l.], v. 7, n. 3, p. 261–276, 2012.

WEISER, M. The Computer for the 21st Century. **SIGMOBILE Mob. Comput. Commun. Rev.**, New York, NY, USA, v. 3, n. 3, p. 3–11, July 1991.

WEISER, M.; BROWN, J. S. Designing calm technology. **PowerGrid Journal**, [S.l.], v. 1, n. 1, p. 75–85, 1996.

ANEXO A PRODUÇÃO DE ARTIGOS

Artigos desenvolvidos durante o mestrado:

Durante o mestrado foram produzidos dois artigos científicos, um para a Revista SQLMagazine (Qualis C), com foco em banco de dados. Este artigo foi produzido em parceria com o colega de mestrado Marcelo Josué Telles. Também foi desenvolvido um artigo para a Revista de Sistemas de Informação da Faculdade Salesiana Maria Auxiliadora RSIFSMA (Qualis B5). O artigo para RSIFSMA contou com a colaboração do colega Marcelo Josué Telles, dos professores Jorge Luis Victória Barbosa e José Vicente Canto dos Santos. A seguir são listados os artigos:

1. TELLES, M.J., SANTINI, P.H. Weka Software: Mineração de dados com Weka – SQLMagazine 128, fevereiro de 2015. ISBN 16779185. Disponível em: <http://www.devmedia.com.br/weka-software-mineracao-de-dados-com-weka/32096>. Acesso em: 24 junho de 2015.
2. TELLES, M.J., SANTINI, P.H., dos SANTOS, J.V.C., BARBOSA, J.L.V. Prevaba: Um Modelo Bayesiano para Predição da Existência de Vítimas em Acidentes de Trânsito – RSIFSMA, 2015. ISSN 1983-5604. Disponível em: http://www.fsma.edu.br/si/edicao16/FSMA_SI_2015_2_Estudantil_3.pdf. Acesso em: 12 janeiro de 2016.

Também foram desenvolvidos trabalhos que ainda estão em processo de desenvolvimento e revisão, listados abaixo:

1. *A Collaborative Model For Ubiquitous Learning Environments – International Journal on E-Learning (IJEL)* – (submetido, em revisão).
2. Um estudo sobre otimização em algoritmos de ordenação – (em desenvolvimento).
3. Algoritmos para análises de coordenadas geográficas – (em desenvolvimento).
4. eGroup: Um Modelo para Gerenciamento de Grupos Dinâmicos de Entidades – (em desenvolvimento).