

Um Modelo Computacional para Cidades Inteligentes Assistivas

Marcelo Josué Telles, Jorge Luis Victória Barbosa e Rodrigo da Rosa Righi

¹Universidade do Vale do Rio dos Sinos UNISINOS
Av. Unisinos, 950, São Leopoldo, Brasil

marcelojtelles@gmail.com, {jbarbosa e rrrighi}@unisinos.br

Abstract. *This article presents MASC, a computational model to support assistive smart cities. The application of ubiquitous computing in the accessibility provides solutions to support people with disabilities (PwD). Unlike related works, MASC uses interactions of PwD to compose trails which will be offered as a service. Moreover, it is generic and supports various disabilities. In addition, MASC was projected for massive applications. The implementation of a prototype allowed MASC evaluation, regarding performance and functionality. The evaluation was conducted with data generated by a context simulator in São Leopoldo - RS. The results indicated that the services offered by the model can be applied in smart cities to collaborate with accessibility, helping PwD, health professionals and public administration.*

Resumo. *Este artigo apresenta o MASC, um modelo computacional para criação de cidades inteligentes assistivas. A aplicação da computação ubíqua na acessibilidade oportuniza soluções para pessoas com deficiências (PcDs). Diferente dos trabalhos relacionados, o MASC utiliza as interações das PcDs para composição de trilhas que são oferecidas como serviços. O modelo é genérico, auxiliando em diferentes tipos de deficiências. Além disso, o MASC foi projetado para viabilizar aplicações massivas. A implementação de um protótipo permitiu a avaliação do MASC, considerando-se desempenho e funcionalidade. A avaliação foi realizada com dados gerados por um simulador de contextos em uma região localizada na cidade São Leopoldo - RS. Os resultados apresentados nos testes indicam que os serviços oferecidos pelo modelo podem ser implantados nas cidades inteligentes para colaborar com acessibilidade, auxiliando PcDs, profissionais da saúde e administração pública.*

1. Introdução

Estudos estatísticos realizados no Brasil em 2010 indicaram um aumento no número de pessoas com deficiências (PcDs). Segundo o mesmo levantamento houve um acréscimo superior a 27% no número de pessoas com limitações para locomoção e aumento de 240% de pessoas com problemas de visão [14]. Diante deste cenário, é importante que sejam desenvolvidas ferramentas para este segmento da população [37]. Além das PcDs, diversas pessoas podem se enquadrar neste grupo de interesse, tais como pessoas transportando algum volume que limite seu deslocamento, pessoas que sofreram acidentes e se deslocam com auxílio de bengalas, idosos ou gestantes que necessitem auxílio para encontrar caminhos mais acessíveis. Além de colaborar com PcDs, idosos, gestantes e obesos o

suporte a acessibilidade auxilia em situações onde qualquer pessoa precise transitar pela cidade.

O idosos constituem um segmento da população beneficiado pela acessibilidade. As definições da Classificação Internacional de Funcionalidade, Incapacidade e Saúde (CIF) descreve as características do indivíduo em diferentes domínios e as características do meio físico e social. Esta classificação tem como propósito o foco na capacidade das pessoas, envolvendo demandas biológicas, psicológicas e sociais e não somente a idade [39].

A infraestrutura disponibilizada pelas cidades inteligentes [47], tais como conectividade, rede de sensores, computação em nuvem, *Big Data*, juntamente com os recursos colaborativos, isto é, informações originadas pelos indivíduos, podem oferecer subsídios para implantação de serviços destinados à acessibilidade [9]. Para proporcionar acesso aos recursos relacionados à acessibilidade é necessária a coleta de informação sobre sua disponibilidade. Informações específicas, tais como largura das calçadas, posicionamento de rampas, piso tátil, locais com rebaixo de meio fio e inclinação das ruas podem ser identificadas por meio de sensores [21] ou descobertas pelas próprias PcDs a medida que estas utilizam tais recursos [19].

O suporte para acessibilidade seja em locais remotos ou em grandes centros urbanos precisa gerenciar informações sobre recursos, PcDs e locais [24]. O gerenciamento inclui disponibilizar informação relevante com base no histórico e contexto da PcD e armazenar as interações dos usuários envolvidos. Com base nas informações geradas pelas PcDs é possível construir uma base de dados históricos para posterior utilização, tanto para colaborar no deslocamento de PcDs, quanto para auxiliar no acompanhamento destes, seja por familiares ou equipe médica. A gestão pública também precisa de informações que colaborem para identificar locais adequados para instalação de equipamentos de acessibilidade [10, 34].

Trabalhos desenvolvidos neste segmento não fazem uso das informações geradas pelas interações das PcDs, gerenciam locais restritos não incluindo recursos dinâmicos e não atendem situações com grande número de PcDs [7, 23, 26, 35, 44]. O MASC (do inglês *a computational Model for Accessibility in Smart Cities*) auxilia PcDs, gerencia contextos [18], recursos, perfis [46] das PcDs, atende aplicações massivas e armazena históricos [33] formando trilhas [6]. As trilhas são compostas por ações e sequências de locais visitados pelas PcDs. Com tais trilhas é possível auxiliar PcDs indicando locais acessíveis, ou seja, locais que outras PcDs com perfil semelhante conseguiram acessar. Os locais visitados pelas PcDs, também podem ser identificados, assim colaborando para instalação de equipamentos de acessibilidade ou dimensionamento de outros recursos de forma otimizada. O fluxo das PcDs também pode colaborar para monitorar os pontos de partida e destino, fazendo com que os caminhos entre tais pontos sejam mais acessíveis para o tipo de PcD que nele se desloca.

Para avaliar o MASC foram realizados dois tipos de testes, um avaliou desempenho e o outro funcionalidade. No teste de desempenho foram realizadas inserções e consultas na base de dados do protótipo para verificar os tempos de resposta. O teste de funcionalidade teve por objetivo atender serviços da administração pública, profissionais da saúde e PcDs.

Este artigo está estruturado em seis seções. A seção 2 resume e compara trabalhos relacionados. A seção 3 descreve o modelo proposto e conceitos sobre ambientes de implantação do mesmo. A seção 4 descreve o protótipo, que contempla um aplicativo para dispositivo móvel, *web services* e *middleware* para sensores. A seção 5 descreve a avaliação do modelo, bem como experimentos realizados e suas limitações. A seção 6 apresenta a conclusão e trabalhos futuros.

2. Trabalhos relacionados

Os trabalhos foram selecionados em pesquisas sobre acessibilidade ubíqua, cidades acessíveis e auxílio para PcDs, estando assim relacionados ao MASC. O comparativo entre os trabalhos incluindo o MASC é apresentado na Tabela 1.

O trabalho mPASS+WhenMyBus [35] propõe uma combinação de sensores, dados abertos e usuários colaboradores para concepção de uma cidade inteligente acessível, identificando rotas e locais para realizar atividades do cotidiano.

O modelo para suporte à acessibilidade ubíqua chamado Hefestos [44] oferece uma arquitetura para auxílio a PcDs. O modelo atende ambientes internos e externos, utilizando uma ontologia para pessoas, recursos e deficiências.

O projeto SHWPWD [26] oferece auxílio para PcDs em seu local de trabalho. O SHWPWD facilita a comunicação e ações das PcDs, utilizando funcionalidades inteligentes do ambiente AmI (do inglês *Ambient Intelligence*), com base em tecnologias assistivas, possibilitando que PcDs utilizem objetos do ambiente.

O sistema de monitoramento e navegação MNDWSN [7] baseia-se em redes de sensores sem fio WSN (do inglês *Wireless Sensor Network*) e *smartphones*. O objetivo do MNDWSN é oferecer auxílio para cegos, surdos e pessoas com deficiência nos membros inferiores e superiores, auxiliando no deslocamento em ambientes internos.

O sistema de navegação LBSBlind [23] adota contexto e oferece auxílio para deficientes visuais proporcionando segurança em seu deslocamento. O sistema LBSBlind atende ambientes internos e externos utilizando RFID e sistema de posicionamento global (GPS do inglês *Global Position System*).

Para realizar a análise comparativa foram estabelecidos os critérios apresentados a seguir:

1. Deficiência física: esse critério identifica se o trabalho oferece suporte a PcDs, nas seguintes situações: deficiência visual (“A”), deficiência auditiva (“B”), deficiência mental (“C”), idosos (“D”), limitações nos membros inferiores (“E”) e limitações nos membros inferiores e superiores (“F”). A classificação “D”, presente nos trabalhos relacionados, é adotada para contemplar pessoas com limitações motoras, entre as quais podem estar os idosos.

2. Perfil: esse critério indica se o trabalho gerencia perfil do usuário, abordando características, preferências e ações dos mesmos. Para este critério os trabalhos receberam a classificação “utiliza” caso gerencie características, preferências e ações das PcDs, “parcial” se apenas gerencie características ou “não” caso não utilize perfis;

3. Trilha: esse critério classifica o trabalho com relação ao gerenciamento de histórico das ações realizadas pelos usuários. Neste critério os trabalhos receberam a

classificação “utiliza” caso ofereça serviços com tais dados de histórico, “parcial” caso apenas armazene histórico ou “não” caso não ofereça serviços nem armazene;

4. ICT: esse critério identifica se o trabalho adota tecnologias consideradas relevantes para esse estudo, recebendo as classificações Computação em Nuvem (“CN”), *Internet of Things* (“IoT”) ou “não” caso não considere nenhuma das tecnologias;

5. Integração: esse critério considera a integração com outros sistemas, recebendo a classificação “integrado” caso utilize dados de outros sistemas e disponibilize serviços, “coleta” se apenas permite o recebimento de informações de outros sistemas ou “não” caso não integre com outros sistemas;

6. Monitoramento: esse critério indica se ocorre monitoramento do usuário e oferta de tais informações para os demais utilizadores do sistema instantaneamente, recebendo a classificação “sim” caso realize tal operação, “parcial” caso monitore, mas não ofereça as informações instantaneamente ou “não” caso não efetue monitoramento;

7. Ontologia: esse critério informa se o trabalho utiliza ferramentas para representação dos domínios cobertos, tais como características dos usuários, locais e recursos. A classificação atribuída para este critério foi “ontologia” caso utilize ontologia, “semântica” caso adote outras ferramentas ou “não” caso não utilize nenhuma forma de representação dos domínios cobertos pelo modelo;

8. Recomendações: esse critério notifica se os usuários recebem suporte com base em seu contexto. A classificação atribuída para este critério foi “adota” ou “não adota”;

9. Ambiente dinâmico: esse critério identifica se há coleta de informações sobre o ambiente utilizando sensores. A classificação atribuída foi “dinâmico”, “estático” caso disponibilize informações sobre o ambiente, mas não ofereça detalhamento por sensores *in loco* ou “ausente” caso não utilize sensores para obter informações do ambiente;

10. Busca de recursos: esse critério indica se ocorre busca de informações sobre recursos em fontes diversas. A classificação foi “sim” caso o trabalho busque em mais de uma fonte ou “parcial” caso realize buscas apenas uma fonte;

11. Aplicado a cidades: esse critério indica se o modelo é escalável e pode ser oferecido nas cidades inteligentes, recebendo classificação “contempla”, “extensível” caso possa ser complementado ou “restrito” caso seja limitado a ambientes específicos.

Com relação ao critério deficiência física, todos os trabalhos oferecem auxílio a PcDs que tenham deficiência visual. Para outras deficiências apenas os trabalhos Hefestos e MNDWSN oferecem suporte para quatro deficiências diferentes. Ainda com relação ao critério 1 apenas o MASC é genérico, pois auxilia no atendimento de todas deficiências indicadas. No critério perfil, apenas os trabalhos mPassWMB e Hefestos fazem uso de mecanismos de gerenciamento de perfil. O trabalho LBSBlind não trata perfil, desta forma os diferentes graus de deficiência visual recebem o mesmo tratamento, já o modelo MASC conta com uma ontologia para representação das deficiências e classificação sobre as particularidades de cada uma.

Quanto a trilhas apenas o trabalho Hefestos faz uso deste recurso. Neste critério o MASC utiliza trilhas para oferecer serviços, tais como permitir profissionais da saúde

Tabela 1: Comparativo entre os trabalhos relacionados

Critério	mPassWMB	Hefestos	SHWPWD	MNDWSN	LBSBlind	MASC
1. Deficiência física	A,B,E	A,B,D,E	A,C,E	A,B,E,F	A	A,B,C,D,E,F
2. Perfil	utiliza	utiliza	parcial	parcial	não	utiliza
3. Trilha	não	utiliza	não	não	não	utiliza
4. ICT	não	não	não	IoT	CN, IoT	CN, IoT
5. Integração	coleta	não	não	coleta	coleta	integrado
6. Monitoramento	não	sim	parcial	parcial	parcial	sim
7. Ontologia	não	ontologia	não	não	semântica	ontologia
8. Recomendações	adota	adota	adota	adota	não	adota
9. Amb. dinâmico	estático	estático	ausente	dinâmico	estático	dinâmico
10. Busca recursos	sim	parcial	parcial	parcial	parcial	sim
11. Apl. cidades	contempla	extensível	restrito	extensível	extensível	contempla

ou familiares acompanhar PcDs em tempo real. Com relação ao critério ICT, todos utilizam funcionalidades de serviços web, mas apenas o LBSBlind faz uso de serviços de computação em nuvem, usa elasticidade de recursos computacionais e atende aplicações massivas. Já as funcionalidades da IoT é utilizada pelo trabalho MNDWSN e LBSBlind.

O recurso de integração não é oferecido por nenhum dos trabalhos, sendo que mPassWMB, MNDWSN e LBSBlind apenas coletam informações de outros sistemas. O MASC é integrado, pois obtém informações de fontes externas de mapas OSM (do inglês *Open Street Maps*)¹ e oferece serviços web para aplicações externas por meio de chamadas *web services*.

No critério monitoramento apenas o trabalho Hefestos realiza tal operação, os trabalhos SHWPWD, MNDWSN e LBSBlind fazem monitoramento, que foi classificado como parcial, conforme o critério indicando que monitoram, mas não oferecem as informações instantaneamente (em tempo real). Este critério é atendido no MASC com o uso de *smartphone* ou *tablet* que deve ser utilizado pela PcD. Para casos onde a PcD não possa utilizar um dispositivo móvel, o monitoramento é feito por meio de tag RFID que será lida por um sensor de tag proposto pelo MASC (MASCnode).

Quanto ao critério ontologia, apenas o trabalho Hefestos adota ontologia para atribuir representação aos elementos do modelo. O trabalho LBSBlind adota semântica apenas para gerenciar contexto. O MASC emprega uma ontologia estendida do modelo Hefestos, que propõe estabelecer representação dos ambientes e trilhas, além das entidades já cobertas pela ontologia original (pessoas, recursos e deficiências).

Com relação ao critério recomendações, apenas o trabalho LBSBlind não contempla este recuso. O MASC opera por meio de mineração de dados para identificar padrões de comportamento a fim de realizar as recomendações. Com relação ao critério ambiente dinâmico apenas o trabalho MNDWSN adota sensores para gerenciamento do estado dos recursos, os trabalhos Hefestos e LBSBlind oferecem informações estáticas dos recursos, ambientes e usuários, já o trabalho mPassWMB oferece informações em tempo real sobre os ônibus da cidade de Bologna. Nesse critério, o MASC utiliza o MASCnode para acompanhar as alterações dos ambientes e disponibilidade dos recursos.

¹Disponível em <https://www.openstreetmap.org/>

O trabalho mPassWMB atende totalmente o critério busca de recursos, os demais trabalhos atendem de forma parcial. Nesse critério o MASC realiza buscas em fontes diversas, tais como usuários colaboradores (*crowdsourcing*), administradores, PcDs, sensores e sistemas externos.

Com relação ao critério de aplicação a cidades, o trabalho mPassWMB contempla cidades inteligentes, ou seja, pode ser implementado neste cenário, pois possui módulos que fazem uso de informações externas originadas nas cidades inteligentes. Os trabalhos Hefestos, MNDWSN e LBSBlind podem ser estendidos para estes cenários, mas atualmente são aplicados em regiões previamente mapeadas e não fazem uso de informações originadas em fontes externas. O trabalho SHWPWD é apenas para ambientes restritos.

O suporte à acessibilidade levando-se em consideração IoT, perfil, contexto e *data mining* (em ações feitas pelo próprio usuário e por demais usuários correlatos), é uma característica que não foi encontrada em nenhum dos trabalhos analisados. Tendo em vista estas questões, é proposto o MASC.

3. Modelo proposto

Antes do desenvolvimento do MASC e de seus componentes foram analisados os cenários de uso e situações do cotidiano das pessoas envolvidas para estabelecer os serviços mais adequados. Diante do exposto, foram analisadas as necessidades presentes no cotidiano das PcDs em serviços básicos, tais como transporte público, mobilidade urbana, compra de alimento, vestuário e uso de recursos. A seguir são descritos cenários ilustrativos onde o mesmo pode ser aplicado.

O primeiro cenário ocorre com a PcD identificada como PcD1. A PcD1 possui limitações nos membros inferiores e utiliza sua cadeira de rodas para se deslocar de sua casa até sua escola. A PcD1, já cadastrada no MASC, utiliza um *smartphone* com o AsP instalado e em execução. Durante o percurso o AsP apresenta os recursos presentes no caminho, assim como armazena os locais que a PcD1 se desloca, gerando uma trilha. Tal trilha é gerada em calçadas específicas.

O segundo cenário ilustrativo ocorre quando uma PcD surda, identificada como PcD2, chega em um *shopping* para uma refeição. A PcD2 também utiliza *smartphone* com o AsP instalado. Ao se deslocar ela visita estabelecimentos na busca por um local com tenha um atendente que entenda LIBRAS. Os locais são registrados pelo AsP e como o tempo que a PcD2 permanece nos locais não é um tempo longo, a trilha continua até que ela permaneça pelo menos 10 minutos em um mesmo estabelecimento. Ao encontrar um restaurante com atendente apto a interpretar LIBRAS, a PcD2 inicia sua refeição. Esta situação acontece mais vezes e todas são registradas no MASC, resultando em um forte indício que o local visitado oferece recursos de intérprete para surdos.

O terceiro cenário ocorre quando uma PcD com limitações visuais, identificada como PcD3 se desloca de sua casa até seu local de trabalho. Esta PcD já foi guiada por um familiar e consegue sair de sua casa e chegar ao local de trabalho com certa facilidade. Uma parte do trajeto é dotada de piso tátil. Durante o deslocamento são armazenados os pontos nos quais a PcD3 se deslocou. Ao chegar em um local com sinal WiFi, o AsP estabelece conexão com o MASCserver e registra os pontos.

Os três cenários permitem a geração de trilhas que podem ser oferecidas para

outras PcDs nas mesmas condições. Uma PcD com limitações nos membros inferiores pode ser auxiliada pela trilha gerada pela PcD1. Os locais pelos quais a PcD1 se deslocou, tais como o lado da rua utilizado para deslocamento e os pontos que atravessou cada rua, constituem informações que podem facilitar a rotina de PcDs com perfis semelhantes. A trilha gerada pela PcD2 fornece informações relevantes e mesmo que o recurso sobre interprete de LIBRAS não seja informado no MASC, o tempo de permanência no local e a reincidência no mesmo local, indica a possibilidade de haver um recurso útil. No terceiro cenário, parte do trajeto percorrido pela PcD3 é equipado com piso tátil, neste a PcD se desloca com uma velocidade ligeiramente maior, fazendo com que os pontos da trilha fiquem mais espaçados. Com a identificação desta alteração na velocidade de deslocamento o MASC pode ajudar outros cegos, indicando o recurso no mapa, por meio de sintetizadores de voz.

É importante considerar o uso de recursos novos visando a otimização de trilhas existentes. Quando novos recursos são instalados, as trilhas já existentes não os contemplam, sendo assim são geradas trilhas que não consideram tais locais, pois não foram acessados por PcDs. Neste sentido a técnica a ser adotada é importante para manutenção do sistema. Tal técnica deve ser realizada por algoritmos de atualização de trilhas. Um exemplo deste algoritmo é quando as informações sobre o recurso são inseridas, fazendo com que sejam identificadas as possíveis trilhas que podem ser alteradas, de acordo com os perfis das PcDs que as realizaram. Após identificar as trilhas que podem ser alteradas e ou beneficiadas pelo novo recurso, devem ser informadas para as PcDs a trilha original assim como os recursos novos.

A implantação em uma situação real, como por exemplo em uma cidade, é algo a ser estudado, principalmente considerando custos de sensores e serviços em nuvem. Uma primeira implantação real do MASC é indicada para um cenário definido, como por exemplo, uma grande empresa ou mesmo uma Universidade.

Na Universidade do Vale do Rio dos Sinos (UNISINOS) os recursos podem ser mapeados facilmente, além disso o uso de sensores pode ser substituído por *smartphones*. Como pré-requisitos existe a necessidade de mapeamento do campus na plataforma OSM, bem como o registro dos veículos do transporte público. Neste local, muitos recursos, tais como rebaixo de meio-fio, elevadores acessíveis, restaurante adaptado, caixas eletrônicos entre outros já foram mapeados pelo modelo de acessibilidade ubíqua Hefestos [44].

3.1. Smart Cities

As definições sobre cidades inteligentes (do inglês *Smart Cities*) [12], [13] e [41] consideram o uso de ICTs para tornar os componentes de infraestrutura e serviços essenciais de uma cidade, mais inteligentes, interligados e eficientes. Este conceito já foi implementado em algumas cidades [8], tais como Brisbane, Malta, Dubai e Kochi. Um dos principais objetivos destas cidades é melhorar a qualidade de vida das pessoas de acordo com diferentes pontos de vista, por exemplo, o nível de acesso às informações, consulta aos recursos relevantes disponíveis, bem como o estado atual de tais recursos.

Tal classificação de cidade apresenta tecnologias que constituem um cenário propício para implantação de diversos serviços, entre eles, acessibilidade e mobilidade urbana [37]. As iniciativas para implantação das *Smart Cities*, definem metodologias que resolvem problemas observados nos grandes centros urbanos [29]. Adicionalmente o ce-

nário das *Smart Cities* é considerado palco de ações planejadas por setores responsáveis e executadas pelos governos locais [40], [15], [31] e [20].

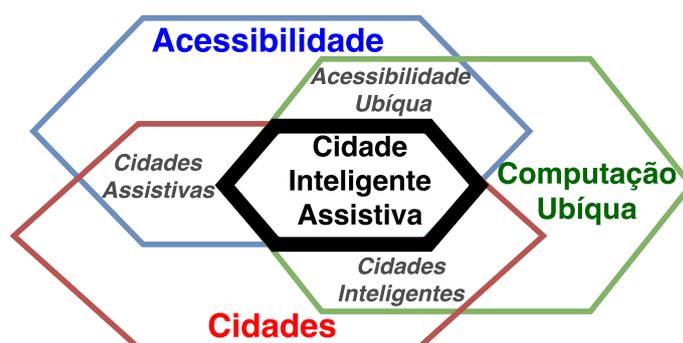
Nas *Smart Cities* muitas ações são possíveis pois certas decisões podem ser tomadas com base em informações que são coletadas em tempo real [50]. Tendo acesso as informações detalhadas e atualizadas, as decisões ainda podem ser comparadas com decisões tomadas anteriormente, assim possibilitando chegar mais certamente na melhor ação a ser tomada.

Para promover tais serviços [49] a infraestrutura de computação deve atender requisitos de elasticidade e balanceamento de carga. Neste sentido a computação em nuvem oferece soluções [48] de Infraestrutura como Serviço (IaaS), Plataforma como Serviço (PaaS) e Software como Serviço (SaaS). No modelo proposto as informações geradas pelas interações das PcDs são armazenadas e gerenciadas com objetivo de oferecer **Trilhas como Serviço (TaaS)** levando em consideração o perfil das PcDs e os ambientes.

A IoT permite obter informações detalhadas do ambiente [43], coletando *in loco* aspectos específicos, tais como presença de objetos em algum local e controle de pessoas em determinado ambiente [11]. O suporte para PcDs utilizando IoT é proposto por [22], que iniciou investigações sobre ferramentas para aumentar a independência e melhorar a participação das PcDs. No MASC são adotados conceitos de IoT para identificar objetos e informações disponíveis sobre os mesmos.

Os conceitos considerados no MASC são apresentados na Figura 1, indicando que a junção das tecnologias da computação ubíqua com as iniciativas de acessibilidade resultam na acessibilidade ubíqua [45]. Uma cidade que adota computação ubíqua torna-se inteligente, já o conceito cidade assistiva é resultante da acessibilidade nas cidades. Ao agregar acessibilidade ubíqua em uma cidade inteligente constitui-se uma **Cidade Inteligente Assistiva**.

Figura 1: Conceitos e tecnologias que compõem as Cidades Inteligentes Assistivas.

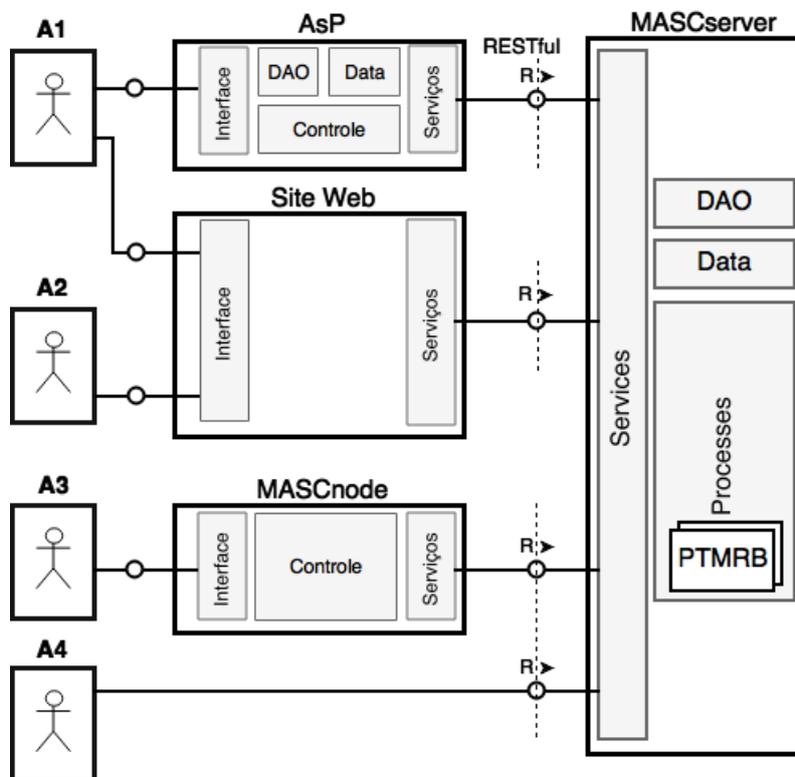


3.2. Arquitetura do modelo

A arquitetura do modelo foi projetada utilizando linguagem unificada de modelagem (UML)². Na Figura 2 é apresentada a arquitetura composta por atores, juntamente com a comunicação que estes estabelecem com Assistente Pessoal (AsP), Site Web, MASCnode e Servidor (MASCserver).

²Disponível em http://www.fmc-modeling.org/download/fmc-and-tam/SAP-TAM_Standard.pdf

Figura 2: Arquitetura do Modelo.



O MASCserver contém bases de dados com informações registradas pelos atores A1, A2, A3 e A4. Atores A1 são PcDs, os quais podem realizar o cadastro e usar serviços por meio do AsP ou Site Web. Atores A2 são familiares das PcDs, profissionais da área da saúde e profissionais da administração pública. Atores A3 são os recursos do ambiente, já atores A4 são os sistemas externos. Os recursos gerenciados pelo componente *Data* podem ser estáticos ou dinâmicos. Estáticos são recursos com posição fixa, já recursos dinâmicos têm posição móvel. Ambos os recursos podem estar ocupados, em manutenção ou desocupados. Os serviços são atendidos pelos *web services* por meio de URIs (do inglês *Uniform Resource Identifier*) com protocolo RESTful [25]. Os *web services* acessam outros dois componentes, *DAO* e *Processes*. O sistema multiagente (SMA) [28] Perfil, Trilha, Monitoramento, Recomendações e Buscas (PTMRB) atua juntamente ao *Processes*. O componente *DAO* é responsável pelo acesso aos dados do modelo que ficam em bancos de dados relacionais [4] e NoSQL [3] [30].

A coleta de dados do ambiente é uma das atividades do MASCnode. Entre elas tem-se a presença de pessoas, registro de PcDs que visitaram o local, conexão e envio de dados para o MASCserver. No Site Web as PcDs informam dados de perfil, consultam informações sobre recursos e obtêm acesso aos serviços oferecidos. Para gerenciamento das trilhas são armazenadas as informações apresentadas na Tabela 2. O AsP permite a coleta de informações das PcDs para realizar monitoramento, além disso oferece mapas com a localização dos recursos e permite buscas por trilhas.

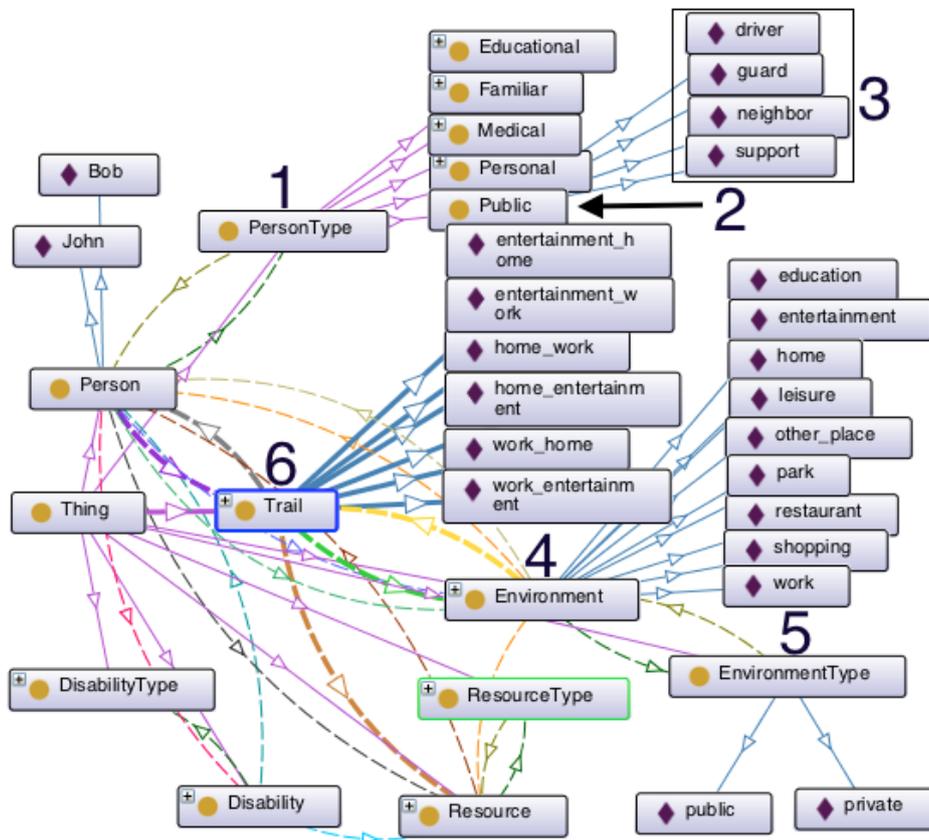
Tabela 2: Propriedades das trilhas

Propriedade	Formato	Descrição
1. evento	<i>TimeStamp</i>	armazena data e hora corrente
2. latitude	Decimal	latitude no formato decimal 17 dígitos no total, 15 decimais
3. longitude	Decimal	longitude no formato decimal 18 dígitos no total, 15 decimais
4. tag	inteiro	identificador da tag registrada
5. recurso	inteiro	identificador do recurso utilizado
6. email	<i>String</i>	identificador da PcD que fez a trilha

3.3. Representação das entidades

A representação dos ambientes, usuários, recursos, deficiências e trilhas é constituída com uso de uma ontologia [38] estendida de Tavares [44], conforme Figura 3. Foram adicionadas as classes *Environment* (4) e *EnvironmentType* (5), além da subclasse *Public* (2) para *PersonType* (1). A classe *Environment* foi adicionada para contemplar os diversos ambientes cobertos pelo MASC. A subclasse *Public* de *PersonType*, foi adicionada para contemplar os tipos de pessoas, tais como motoristas, guardas, vizinhos ou pessoas que prestam suporte (3). Outra modificação foi a classe *Trail* (6) para representação das trilhas e suas respectivas relações com PcDs, recursos e ambientes. A padronização para representação dos ambientes e trilhas segue definições de AASHTO [5].

Figura 3: Ontologia utilizada pelo MASC.



Adaptado de Tavares [44]

A ontologia³ estabelece significado e relações entre as entidades, possibilitando a representação dos atores envolvidos, assim como os tipos de deficiências, recursos para cada deficiência e locais visitados pelas PcDs e monitorados pela equipe médica e administração pública.

O modelo de dados consiste de elementos básicos para atender chamadas *web services*, armazenar dados das aplicações clientes e manter o relacionamento entre as entidades do modelo. A ontologia foi adotada para representação de tais entidades, bem como suas relações em nível de modelagem. Nos testes realizados para avaliação, as fontes de dados da ontologia RDF (do inglês *Resource Description Framework*) não foram empregadas. A avaliação adotou apenas dados gerados na simulação.

Uma das principais contribuições do MASC consiste em oferecer auxílio para PcDs, profissionais da saúde e administração pública no âmbito da acessibilidade. O MASC oferece suporte por meio de gerenciamento de recursos e perfis. O modelo trata as rotas percorridas pelas PcDs, com o objetivo de utilizá-las em situações futuras, onde PcDs com perfil semelhante possam receber tais rotas para realizar suas atividades do cotidiano.

4. Protótipo

O protótipo do modelo contemplou as tecnologias apresentadas na Figura 4. O MASC-server utilizou *web services* e bases de dados. Para desenvolver os *web services* adotou-se o JBoss Developer Studio [2]. O AsP foi desenvolvido com a IDE Android Studio [1] e Java 7. A implementação do MASCnode adotou linguagem *Python* e *hardware* Raspberry Pi. Após o desenvolvimento do protótipo o mesmo foi aplicado na avaliação do MASC.

Para realização da avaliação foi desenvolvida uma aplicação composta por algoritmos em Java com API Swing na IDE Netbeans 7.4⁴. Os *web services* do MASCserver receberam as requisições desta aplicação e os agentes do PTMRB realizaram troca de mensagens, armazenamento de informações e consultas nos dados. O aplicativo para Android, indicado na Figura 4, coleta informações para composição das trilhas. Também é possível visualizar um mapa com recursos de acordo com o perfil da PcD. Na configuração padrão é enviado ao servidor as posições envolvidas no deslocamento da PcD, assim que houver conectividade com internet.

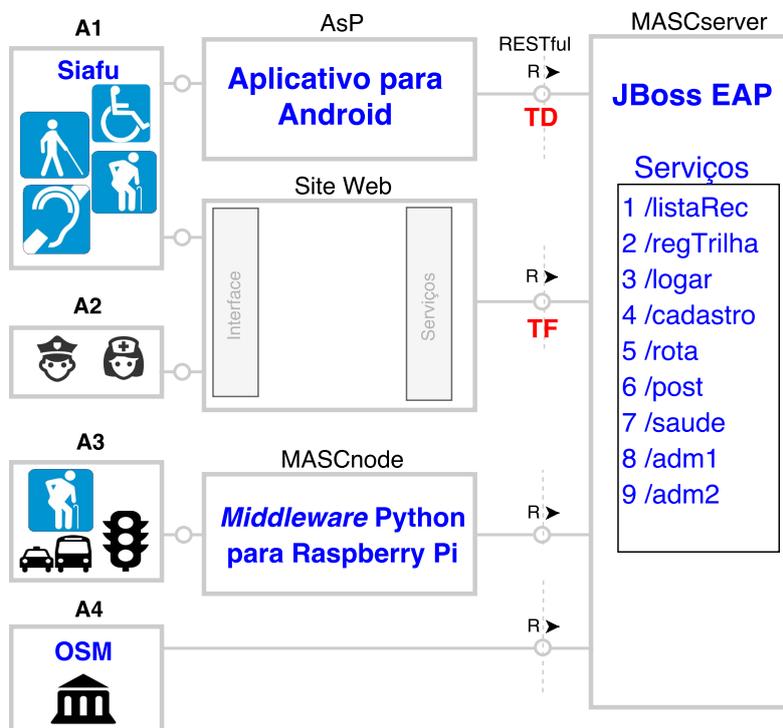
Foi desenvolvido um *middleware* para sensores visando coleta de informações sobre o ambiente e PcDs que não utilizam o AsP. Por meio de leitor de RFID o *middleware* identifica a presença de qualquer PcD e envia tais informações para as bases de dados do MASCserver. Também podem ser coletadas informações contextuais tais como monitoramento de recursos, temperatura, umidade, hora e presença de pessoas. Na Figura 4 também são indicados os testes de desempenho (TD) e funcionalidade (TF), a simulação (Siafu) utilizada para experimentos e o sistema externo (OSM).

Os *web services* utilizam a estrutura JAX-RS que é a implementação de RESTful *web services* em Java, proposta pela especificação JSR-311 [25]. Os *web services* atendem um conjunto de chamadas que se concentram em nove URIs, indicadas no retângulo

³Disponível em <http://webprotege.stanford.edu/#Edit:projectId=5bafb5e9-bdbe-44d9-808d-fab281a332ba>

⁴Disponível em <https://netbeans.org/community/releases/74/>

Figura 4: Tecnologias envolvidas no protótipo.



Legenda da Figura 4

- TD: testes de desempenho;
- TF: testes de funcionalidade;
- Siafu: simulação para geração das trilhas;
- OSM: sistema externo *Open Street Maps*.

Serviços da Figura 4.

O MASCserver foi configurado no OpenShift⁵. Dentre os serviços oferecidos pelo OpenShift foram utilizados *SSH*⁶, *RHC*⁷ e *IDE Integration*⁸. Também foi utilizado o servidor de aplicações JBoss *Enterprise Application Platform 6* (JBoss EAP).

O esquema de dados do MASC é composto por tabelas, relacionamentos e índices. A Figura 5 apresenta o esquema proposto para gerenciamento de recursos, perfis e trilhas das PcDs. Adotou-se o RDBMS MySQL 5.1.73 *Source distribution* [4] que é um sistema gerenciador de banco de dados relacional. Por sua vez, para armazenamento das coleções NoSQL foi utilizado o MongoDB 2.4.9 [3]. As tabelas possuem relacionamentos para garantir consistência das informações. As tabelas *_usuario*, *_trilha* e *_tipo_recurso*, foram as três principais usadas na avaliação. Para gerenciamento do perfil foi desenvolvida a tabela *_tipo_def* que é responsável por armazenar a deficiência de cada PcD. As informações da tabela *_tipo_def* foram previamente cadastradas na base de dados, assim como

⁵OpenShift, disponível em <https://www.openshift.com/> é um serviço de computação em nuvem que oferece PaaS gratuitamente, permitindo a criação de *web services*, bases de dados MySQL, PostgreSQL e MongoDB, com suas respectivas ferramentas de gerenciamento.

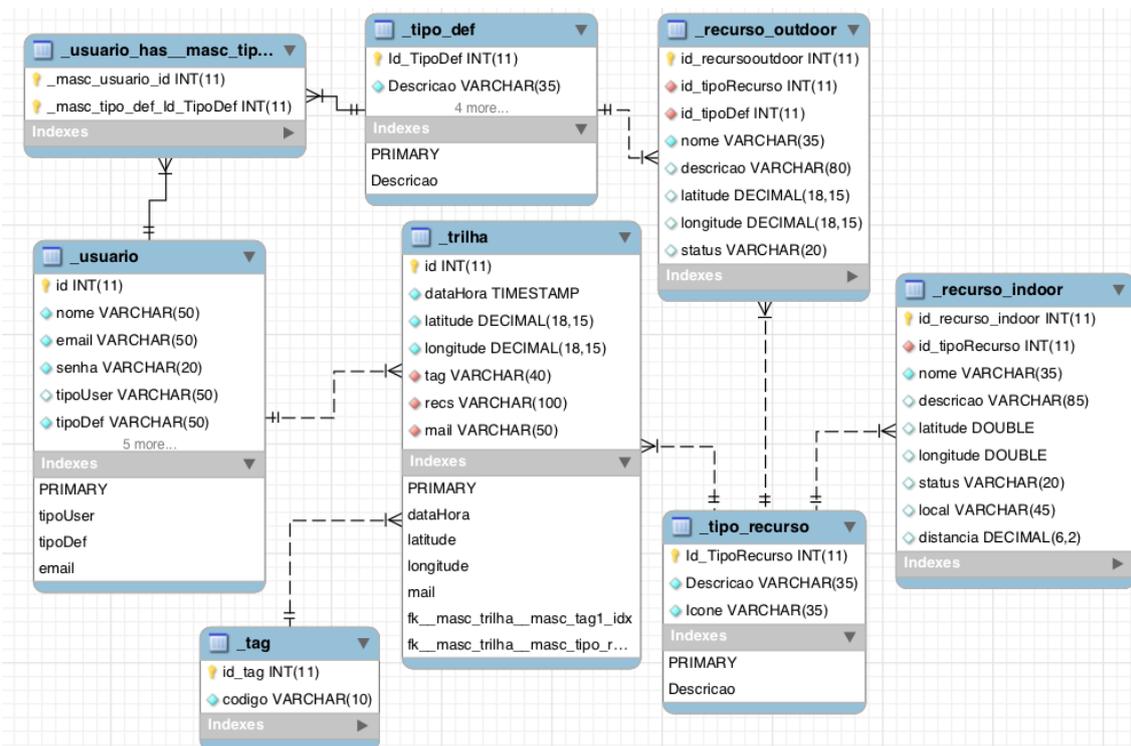
⁶Disponível em <https://developers.openshift.com/en/managing-remote-connection.html>

⁷Disponível em <https://developers.openshift.com/en/managing-client-tools.html>

⁸Disponível em <https://developers.openshift.com/en/getting-started-installing-jboss-studio.html>

as informações das tabelas *_recurso_outdoor*, *_recurso_indoor* e *_tipo_recurso*. A tabela *_trilha* contém as informações dos locais que uma PcD ou tag se deslocaram, conforme descrito na Tabela 2. Com base nestas informações foi realizada a avaliação.

Figura 5: Esquema de dados para Recursos Perfis e Trilhas.



O armazenamento das informações enviadas pelo MASCnode não seguem um esquema, isto é, não são armazenadas em uma tabela com colunas fixas. As informações são armazenadas em coleções no formato *JavaScript Object Notation* (JSON). Um exemplo de formato é apresentado na Figura 6 que mostra o arquivo gerado pelo MASCnode 1, o qual foi desenvolvido para o protótipo. Neste arquivo estão informações de proximidade (linha 1), a qual indica 100 cm. Nesta mesma linha o valor 10 cm aparece, indicando que foi registrada uma aproximação. O endereço *Internet Protocol* (IP) do MASCnode é apresentado na linha 2, o número da tag lida na linha 4, a data e hora do envio das informações na linha 5, já a temperatura e umidade estão na linha 6. Como se trata de um exemplo de MASCnode móvel, as posições são apresentadas entre as linhas 7 e 18.

4.1. Serviços desenvolvidos

A aplicação desenvolvida para avaliar o MASC, acessou os serviços do modelo. Os mesmos estão divididos em serviços básicos que são resumidos na Tabela 3 e serviços para profissionais da saúde e administração pública descritos na Tabela 4. A seguir são apresentados os serviços.

O primeiro serviço oferece para PcDs uma lista de recursos, onde cada item da lista é composto por informações que nomeiam o recurso, indicam sua distância, localização e ícone para sua representação no mapa. O segundo serviço é responsável por armazenar os pontos (latitude e longitude) das trilhas. O terceiro serviço autentica e controla o usuário que está enviando as informações. O quarto serviço realiza o cadastro

Figura 6: Exemplo de estrutura para armazenar dados do MASCnode.

```

1  ▼ {Proximity: [100,10],
2  IP: "189.58.183.121"
3  MASCnode: "1",
4  RFID: [718930480],
5  Date: "2015/07/24 15:49:20",
6  TempHumidity: ["17.0 40.0", "17.0 40.0"],
7  ▼ Local: {status:"Success",
8      lat:-29.687394,
9      lon:-51.107221
10     },
11  ▼ Local: {status:"Success",
12      lat:29.687478,
13      lon:-51.107240
14     },
15  ▼ Local: {status:"Success",
16      lat:-29.687632,
17      lon:-51.107270
18     },
19  }

```

do usuário. O quinto serviço foi utilizado para o teste de funcionalidade, servindo para solicitar uma rota. Os cinco serviços são detalhados na Tabela 3.

Tabela 3: Serviços básicos para protótipo do MASC

URI	Parâmetros	Exemplos de Retorno
/listaRec	PcD; Membros inferiores; -29.792839818491657; -51.15276753902435;	Rampa;9.4914;Membros Inferiores; -29.7929720002;-51.1528106670;amarelo Vaga estacionamento;10.4465;Membros Inferiores; -29.7929097;-51.15277050;azul Elevador;22.7437;Membros Inferiores; -29.7931471667;-51.1526316680;rosa
/regTrilha	-29.791959945956947; -51.15311086177826; 17830284; 0; blanch@server.com.br	trilha ok;
/logar	marcelo@server.com; abc;	ok;PcD;Membros Inferiores;
/cadastro	Marcelo Josué Telles; marcelo@server.com; abc; PcD; Membros inferiores;	E-mail já cadastrado;
/rota	-29.762738198684357; -51.151206493377686; marcelo@server.com; -29.768084050351185; -51.145241260528564;	-29.762738198684357,-51.151206493377686; -29.762589181311057,-51.150412559509280; -29.762421536501150,-51.149532794952390; -29.762309773138735,-51.149082183837890; -29.762272518656890,-51.148653030395510; -29.763539163268756,-51.148352622985840; -29.764936179196734,-51.148009300231934;

O sexto serviço armazena em uma base NoSQL dados coletados pelo MASCnode. Este serviço recebe como parâmetro um número que identifica o MASCnode e dados sobre contexto. Um exemplo de parâmetro foi apresentado na Figura 6. Como retorno o servidor envia um arquivo com a configuração do respectivo MASCnode. Neste arquivo de configuração são definidos os tempos de leitura dos sensores e horário de funcionamento. Na Figura 7 é apresentado um exemplo de um arquivo de configuração de um MASCnode móvel, linha 3 (*dynamic*). Também são parametrizadas as informações sobre seu período de trabalho, que é entre 7 horas e 19 horas, de segunda-feira a sexta-feira (linhas 4 e 5). Os tipos de sensores lidos são temperatura, umidade, proximidade e RFID. A cada 30 segundos o sensor de temperatura e umidade (DHT11) realiza a leitura. O sensor de proximidade faz a leitura a cada 2 segundos, já o leitor de RFID realiza leitura

constantemente. A informação sobre posição é coletada a cada 20 segundos, o endereço IP a cada 120 segundos, todas informações são enviadas para o servidor a cada minuto.

Para alterar o modo de funcionamento do MASCnode basta alterar a respectiva coleção no banco de dados MongoDB. O arquivo apresentado na Figura 7 exemplifica o arquivo recebido como retorno, quando o MASCnode 1 (um) envia dados ao MASCserver.

Figura 7: Arquivo de configuração do MASCnode 1.

```
1  {"_id": ObjectId("55b27533bb03721a8b000001"),
2   "MascNode": "1",
3   "Position": "dynamic",
4   "WorkTime": "7,19",
5   "WorkWeek": "2,6",
6   "Sensors": {"Types": [{"DHT11": {"sleep": "30"}},
7                     {"Proximity": {"sleep": "2"}},
8                     {"RFID": {"sleep": "0"}}
9                   ]
10  },
11  "Data": {"Intervals": [{"Position": {"interval": "20"}},
12                    {"Ip": {"interval": "120"}},
13                    {"SendData": {"interval": "60"}}
14                  ]
15  }
16 }
```

5. Avaliação

Foram realizados testes de desempenho e funcionalidade para avaliação do modelo. Para o desempenho foi feito o envio de dados ao servidor e requisições de consultas. A métrica para testar o desempenho foi o tempo de resposta. Para testar a funcionalidade foram considerados três serviços: serviço para PcDs (/rota), profissionais de saúde (/saude) e administração pública (/adm1 e /adm2).

O teste de desempenho foi implementado com algoritmos utilizando processos paralelos (*Threads*). Foram utilizados dados contextuais gerados pela simulação realizada na ferramenta Siafu [32], com objetivo de simular PcDs na cidade São Leopoldo - RS.

5.1. Simulação e testes

A simulação foi parametrizada com 125 pessoas, representadas por agentes. O último registro do Ministério da Saúde⁹ registrou 254 PcDs na cidade de São Leopoldo - RS, no entanto foi definido que aproximadamente metade da população (125) faria uso do modelo. Este número de agentes foi adotado, também, em razão de limites no processamento da aplicação responsável pela simulação, a qual gerencia até 995 agentes dependendo do tamanho do mapa e tempo de simulação. No mapa adotado, considerando o período de sete dias, o limite de agentes gerenciado foi de 125.

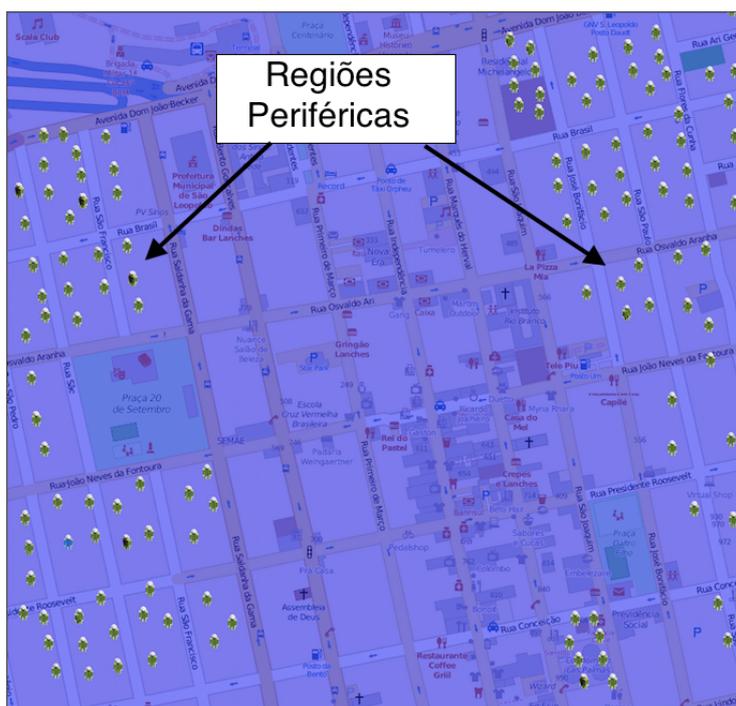
A região da simulação foi indicada pela latitude e longitude, locais das residências, locais de trabalho e lazer para que os agentes se deslocassem. Também é possível configurar o percentual de pessoas que utilizam automóvel como meio de transporte. Nos

⁹Disponível em <http://tabnet.datasus.gov.br/cgi/tabcgi.exe?siab/cnv/SIABFbr.def>

testes este percentual foi configurado com o valor 0 (zero). Outras configurações são possíveis, tais como idade, horário de trabalho, período de sono, zonas de calor, zonas de cobertura de WiFi, índice de assalto entre outras.

A simulação gerou as interações das 125 pessoas no centro da cidade tomando como base latitude e longitude reais. Os agentes foram posicionados nas regiões residenciais, na periferia do mapa. A Figura 8 apresenta os agentes posicionados na região periférica do mapa.

Figura 8: Simulação no início do primeiro dia.

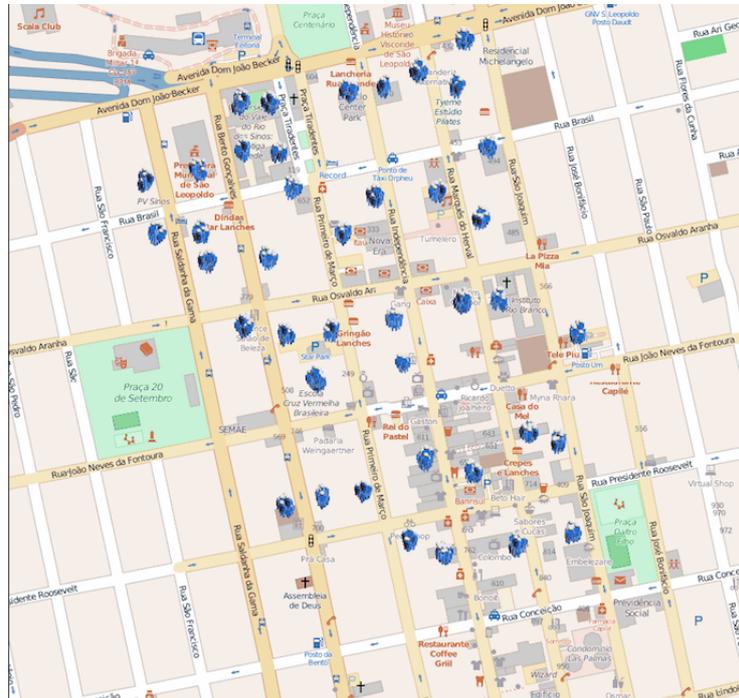


Na Figura 8 é possível estimar os locais a serem percorridos pelos agentes da simulação. Como se trata de uma região central existem diversos caminhos, tais como calçadas, praças e galerias que são abertas ao público e podem ser acessadas no deslocamento diário.

No início do dia os agentes da simulação se deslocaram para os locais de trabalho, criados na região central do mapa. Foram definidos 43 locais de trabalho. Ao final do dia alguns agentes permaneciam no local de trabalho por mais algum tempo, alguns se deslocavam para sua residência, outros se deslocavam para um local de lazer, tendo sido definidos 37 locais de lazer. A Figura 9 apresenta os agentes já posicionados nestes locais.

A simulação seguiu por sete dias tendo sido registrada a posição de cada agente durante esse período. Ao todo foram gerados 270.000 registros de histórico, formando as trilhas. Os dados da simulação foram exportados para um arquivo no formato CSV (do inglês *Comma Separated Value*), para depois serem enviados ao MASCserver. Cada registro da simulação é composto pela identificação do agente, ação que este está realizando e posição indicada pela latitude e longitude. Foram inseridos atributos manualmente, para completar as informações geradas na simulação, por exemplo, a deficiência de cada agente, e-mail, idade e peso.

Figura 9: Simulação com os agentes posicionados nos locais de trabalho.



5.2. Testes de desempenho

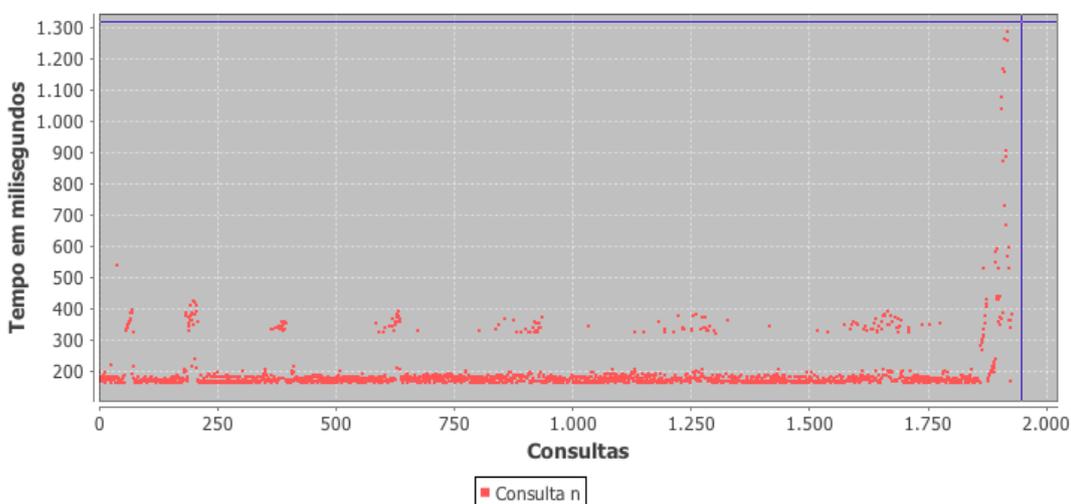
Durante o envio dos dados para o MASCserver, e sua respectiva inserção na base de dados, foi realizado o monitoramento do tempo de resposta. Todos testes foram assíncronos, concorrentes, sendo que o tempo de resposta foi considerado desde o envio da requisição até seu respectivo recebimento [27] pelo cliente. Foram executados 125 processos em paralelo, sendo que cada um ficou responsável por enviar 2160 registros do arquivo CSV.

A média observada para o tempo de resposta foi 286 ms (milissegundos), o menor tempo de resposta foi 164, o maior foi 33006 ms, o desvio padrão foi 858 ms, indicando que a grande maioria dos tempos está a menos de 1000 ms da média. O total de requisições com tempo de resposta maior ou igual a 2000 ms foi de 836 requisições. A grande maioria das requisições foram atendidas em torno de 200 ms e 400 ms. Descartando as requisições que levaram 2000 ms ou mais (836 requisições), pode-se considerar que o serviço atendeu 99,6904 % das requisições em menos de 2 segundos, que é o tempo aceitável para este tipo de aplicação [36]. Este percentual representa a garantia de que as informações para construção das trilhas serão armazenadas, pois a perda de 0,3096 % das requisições não compromete a construção das trilhas. As requisições que levaram 2000 ms ou mais, ficaram dispersas, não concentradas em um momento, indicando que as requisições não atendidas são apenas uma ou duas de cada trilha.

Além do teste para geração de trilhas, foi conduzido um teste de requisições e consultas ao MASCserver. Foi processada a rotina que seria mais frequentemente executada nos testes de funcionalidade. A rotina realiza comparação da distância entre uma posição, geralmente a posição atual da PcD e as posições dos recursos. Além disso a rotina faz junções (*RIGHT JOIN*) das tabelas para obter informações dos tipos de recursos e tipos de deficiências das PcDs. O algoritmo responsável por este teste gerou *Threads*

para simular PcDs. A medida em que eram iniciados os *Threads*, o tempo de resposta das requisições foi armazenado para análise. Foi definido que a cada 100 ms, mais 1 PcD iniciaria requisições ao MASCserver. Cada PcD realizava 1 requisição a cada 5 segundos. O teste tinha por objetivo identificar qual o máximo de PcDs atendidas, desta forma ao identificar a primeira requisição sem retorno, o teste foi finalizado. Ao final do teste, 415 *Threads* haviam sido gerados. O total de requisições atendidas foi 1925 em 42 segundos. Na Figura 10 são apresentados os tempos de resposta das consultas para este teste. Os resultados indicados na figura mostram que o tempo de resposta manteve-se abaixo de 1300 ms. Também é possível visualizar que após 1750 consultas o tempo de resposta perdeu a estabilidade.

Figura 10: Tempo de resposta para consultas dos recursos.



Com os dois testes apresentados concluiu-se que o MASCserver atendeu as requisições para gravação das informações das trilhas e atendeu as requisições de consultas. O servidor suportou o envio de informações sobre trilhas geradas pelos 125 agentes da simulação e atendeu requisições de consultas para 415 agentes.

5.3. Testes de funcionalidade

Os testes de funcionalidade consistiram em atender serviços de acessibilidade para PcDs, profissionais da área da saúde e administração pública.

5.3.1. Serviços para PcDs

O serviço para este teste recebe como parâmetro um ponto de início, e-mail e ponto de fim, conforme exemplo na última linha da Tabela 3 e realiza uma busca. Ao receber esta entrada o *web service* busca pontos próximos do ponto de início, e em seguida é composta a trilha de cada ponto. São consideradas apenas trilhas de PcDs com perfil semelhante à PcD requisitante, comparando idade, peso e tipo de deficiência da pessoa. Começando pela trilha mais próxima, é realizada a verificação se ao longo desta existe um ponto próximo (dez metros) ao ponto de fim, se existe, esta trilha é indicada e o MASCserver retorna a trilha. Na Figura 11 é apresentado um mapa com o resultado desta busca.

Figura 12: Serviço para profissionais da saúde exibindo a última posição das PcDs paraplégicos.

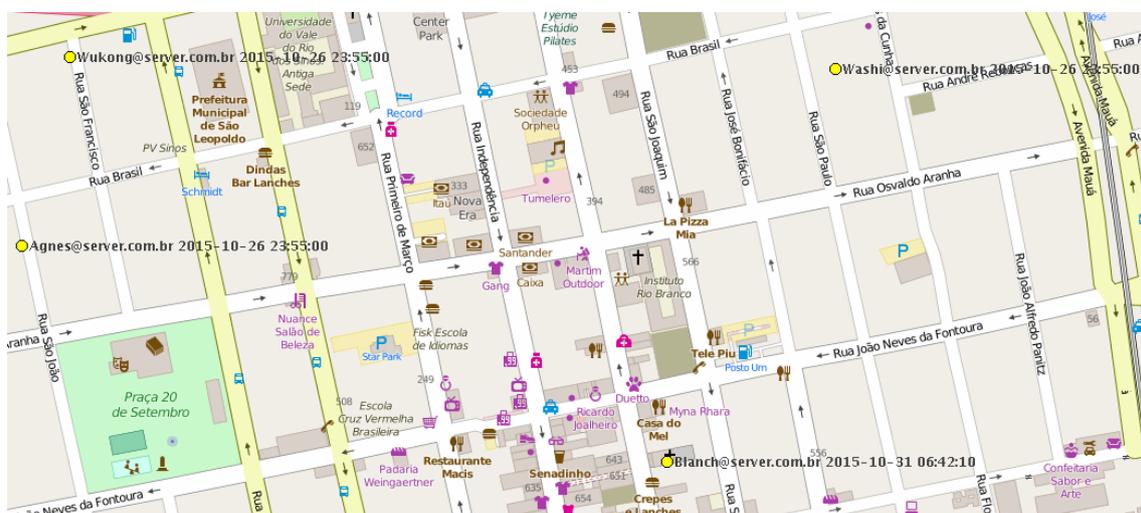


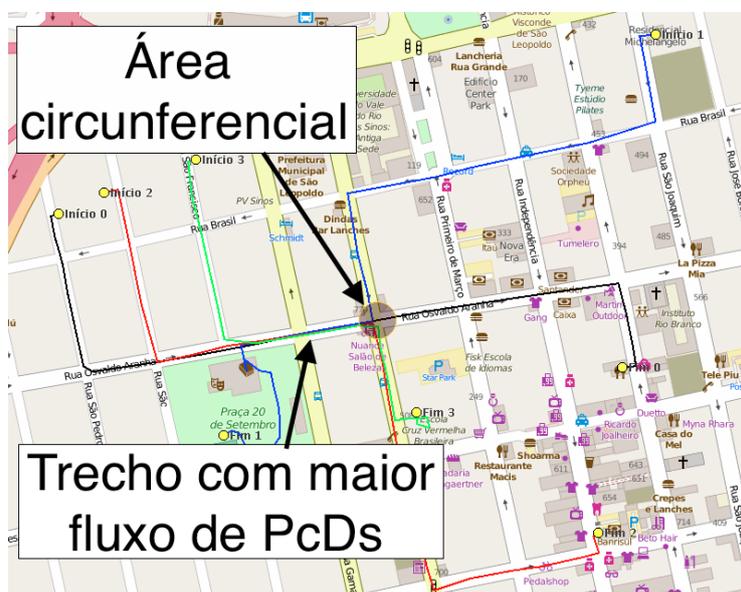
Tabela 4: Serviços para profissionais da saúde e administração pública

URI	Exemplo	Parâmetros	Exemplos de Retorno
/saude	e-mail de uma PcD	0;	marcelo@server.com.br;2015-10-27 22:15:05;-29.767394879916615, -51.1439323425293;
/saude	por deficiência	1; Membros Inferiores;	blanch@server.com.br;2015-10-31 06:42:10;-29.764982746058855;-51.144844293594360; wash@server.com.br;2015-10-26 23:55:00;-29.761611647633180;-51.143186558161354; agnes@server.com.br;2015-10-26 23:55:00;-29.763137962043000;-51.151165614854420; wukong@server.com.br;2015-10-26 23:55:00;-29.761515653016210;-51.150695630256706;
/saude	lista de e-mail	2; rusa@server.com.br; sophie@server.com.br; hikaa@server.com.br; zoraida@server.com.br; tapah@server.com.br;	rusa@server.com.br;2015-10-26 23:55:00;-29.763646733512940;-51.151464695962060; sophie@server.com.br;2015-10-26 23:55:00;-29.764856265686760;-51.142107729880220; hikaa@server.com.br;2015-10-26 23:55:00;-29.767803300427730;-51.144030394143610; zoraida@server.com.br;2015-10-26 23:55:00;-29.764635478067728;-51.151240385131330; tapah@server.com.br;2015-10-26 23:55:00;-29.765029055997303;-51.151197659258810;
/adm1	trilha de PcD	blanch@server.com.br; 2015-10-31;	1;-29.761927914043167;-51.150659322738650; 2;-29.761918600388520;-51.150541305541990; 3;-29.762123500590990;-51.150498390197754; 4;-29.762542613336443;-51.150412559509280; 5;-29.762542613336443;-51.150187253952026; ...
/adm2	área circunferencial	-29.763483282226453; -51.14838480949402; 12;	n conjuntos de pontos, sendo que cada conjunto é formado por uma sequência com identificador, latitude e longitude no mesmo formato do serviço /adm1

este serviço a prefeitura da cidade pode identificar o fluxo de PcDs para instalação de equipamentos de acessibilidade, dimensionar recursos no transporte público, providenciar instalação de equipamentos de segurança em locais onde se concentram mais PcDs, alocar guardas e agentes de segurança em locais estratégicos, identificar locais para instalação de rampas de acesso e previsão de fluxo para tomada de decisão. A Figura 13 apresenta as trilhas que passam pela circunferência.

Uma funcionalidade para administração pública é a identificação automática de

Figura 13: Serviço para prefeitura exibindo trilhas passando pela área circunferencial.



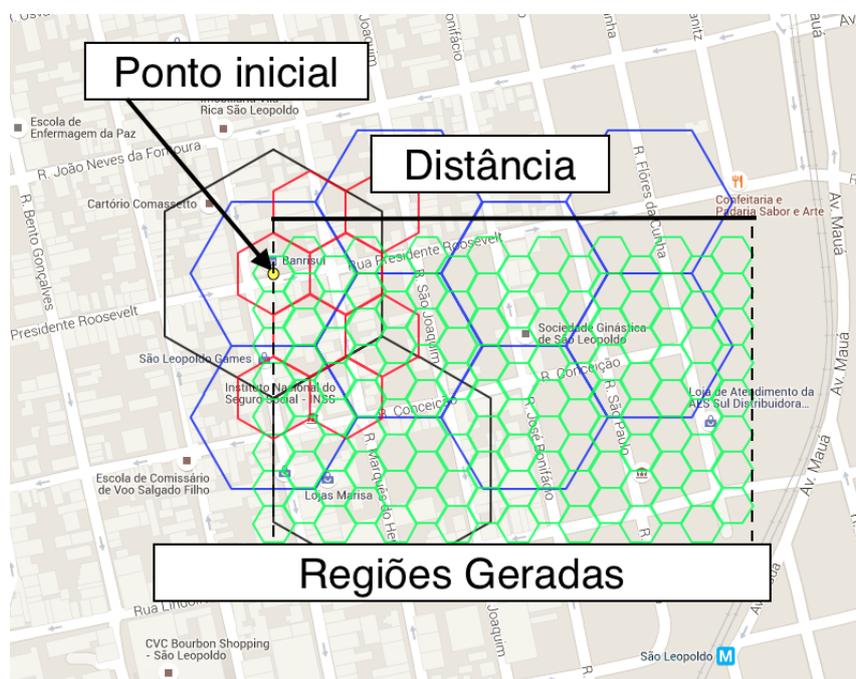
locais com concentração de PcDs em tempo real e ao longo do tempo. Para tanto foi desenvolvido um serviço que recebe como parâmetro um ponto, uma distância e uma ou duas datas.

Ao receber um ponto, uma distância e uma data, o serviço parte do ponto e encontra locais com concentração de trilhas na data informada. A técnica adotada é semelhante a usada no serviço da URI /adm2 pois são geradas regiões de interesse no formato circunferencial. No teste realizado foram geradas regiões de interesse obedecendo a média das dimensões dos quarteirões da região da simulação, que é 110 metros. As áreas de interesse são geradas com 55 metros de raio até atingir a distância informada pelo parâmetro, conforme Figura 14 que apresenta a geração das regiões para a direita do ponto informado, ou seja, a leste no mapa.

Figura 14: Serviço para encontrar regiões com concentração de PcDs expandindo para leste.



Figura 16: Teoria dos lugares centrais aplicada na técnica de tratamento das trilhas.



em consideração tais recursos. Outra limitação do modelo é que a medida que a cidade recebe novas ruas, calçadas ou mesmo galerias estes locais devem ser visitados por PcDs para que sejam recomendados em situações futuras.

Sendo assim trilhas que não foram percorridas pelas PcDs, assim como locais que receberam melhorias ao longo do tempo não são consideradas nas recomendações do modelo. Tais locais não serão descobertos automaticamente, necessitando que alguém responsável pela manutenção do sistema registre-os no modelo.

A composição de trilhas por partes também é uma limitação do modelo. Dentre as alternativas para composição de trilhas pela junção de trechos de diferentes PcDs tem-se algoritmos de caminho mínimo [17] e [42], onde os vértices conhecidos de trilhas já realizadas podem ser utilizados para geração de uma trilha composta por vértices já visitados por diferentes PcDs. Como alternativa também pode ser utilizado o algoritmo do caixeiro viajante, mas não do carteiro chinês que tem a possibilidade de visitar um local já visitado, o que não é interessante no suporte a PcDs. Outras técnicas presentes na área de complexidade de algoritmos também são possíveis, como exemplo o algoritmo de Floyd [17] que utiliza programação dinâmica.

6. Conclusão

Este artigo propôs um modelo computacional para cidades inteligentes assistivas, denominado MASC. A principal contribuição desse trabalho consiste na oferta de trilhas para acessibilidade, permitindo a geração de fontes de dados com perfil, contexto e interações das PcDs. Tais dados contribuem no suporte a PcDs e colaboram para tomada de decisão de administradores das cidades. As interações das PcDs são aproveitadas como fonte de informação para composição das trilhas. O MASC gerencia recursos estáticos e

dinâmicos, utilizando computação em nuvem e sensores. Como o modelo é genérico, o mesmo pode auxiliar pessoas com diferentes tipos de deficiências, tais como: deficiência visual, deficiência auditiva, deficiência mental, idosos, limitações nos membros inferiores e limitações nos membros inferiores e superiores.

Outra contribuição do MASC é a infraestrutura proposta, que pode ser aplicada em larga escala, atendendo sistemas com grande número de PcDs e diferentes tipos de deficiência. Por atender PcDs, profissionais da área de saúde e administração pública, o MASC gerencia recursos de acessibilidade para diversos tipos de usuários.

As tecnologias adotadas no protótipo são de padrão aberto, tanto *software* quanto *hardware*, sendo possível sua integração em qualquer cidade. Com os serviços disponibilizados, podem ser desenvolvidas outras aplicações para áreas diversas, fazendo uso de informações sobre deslocamento de outras pessoas, tais como profissionais de um segmento, estudantes, veículos ou mesmo para controle de cargas.

Como trabalhos futuros destaca-se a necessidade de segmentar o armazenamento dos dados, a fim de manter tempos de resposta abaixo de dois segundos para consultas aos recursos. Outra parte do MASC a ser aperfeiçoada são os serviços, para atender necessidades de outras áreas tais como segurança, transporte e planejamento urbano.

7. Agradecimentos

Os autores agradecem à PROSUP/CAPES (Programa de Suporte à Pós-Graduação de Instituições de Ensino Particulares)¹⁰ pela bolsa concedida. Fica registrado o agradecimento à Universidade do Vale do Rio dos Sinos - UNISINOS¹¹ pelo apoio acadêmico ao trabalho.

Referências

- [1] Android studio. Disponível em: <<http://developer.android.com/sdk/index.html>>. Acesso em: 18 mar. 2014. Site oficial do IDE Android Studio.
- [2] Jboos developer. Disponível em: <<http://www.jboss.org/products/devstudio/get-started/#!project=devstudio>>. Acesso em: 16 mai. 2014, 2014. Site oficial do JBoos Developer Studio.
- [3] MongoDB. Disponível em: <<https://www.mongodb.org/>>. Acesso em: 05 fev. 2014, 2014. Site oficial do MongoDB.
- [4] Mysql. Disponível em: <<https://www.mysql.com/products/community/>>. Acesso em: 20 mar. 2015, 2015. Site oficial do MySQL.
- [5] AASHTO, American Association of State Highway and Transportation Officials. A policy on geometric design of highways and streets. 2001.
- [6] P. C. Albarello. Controle de acesso sensível ao contexto baseado na inferência em trilhas. Master thesis, University of Vale do Rio dos Sinos (Unisinos), Applied Computing Graduate Program (PIPCA), São Leopoldo, Brazil, 2013.
- [7] W. H. F. Aly. MND WSN for helping people with different disabilities. *International Journal of Distributed Sensor Networks*, (489289), 2014.

¹⁰Disponível em <http://www.capes.gov.br/bolsas/bolsas-no-pais/prosup>

¹¹Disponível em <http://www.unisinos.br>

- [8] L. Anthopoulos and P. Fitsilis. From digital to ubiquitous cities: Defining a common architecture for urban development. In *Proceedings of the 2010 Sixth International Conference on Intelligent Environments*, IE '10, pages 301–306, Washington, DC, USA, 2010. IEEE Computer Society.
- [9] R. O. Arbex, B. B. Alves, and M. A. Giannotti. Comparing accessibility in urban slums using smart card and bus gps data. In *Transportation Research Board 95th Annual Meeting*, number 16-5614, 2016.
- [10] D. Belanche, L. V. Casaló, and C. Orús. City attachment and use of urban services: Benefits for smart cities. *Cities*, 50:75 – 81, 2016.
- [11] E. Borgia. The internet of things vision: Key features, applications and open issues, 2014.
- [12] M. N. K. Boulos and N. M. Al-Shorbaji. On the internet of things, smart cities and the who healthy cities. *International journal of health geographics*, 13(1):10, 2014.
- [13] P. Branchi, I. Matías, and C. Fernández. City amp; technology: An analysis matrix to serve citizens. In *New Concepts in Smart Cities: Fostering Public and Private Alliances (SmartMILE), 2013 International Conference on*, pages 1–5, Dec 2013.
- [14] IBGE Instituto Brasileiro de Geografia e Estatística, 2010.
- [15] H. Chourabi, T. Nam, S. Walker, J. Gil-Garcia, S. Mellouli, K. Nahon, T. Pardo, and H. J. Scholl. Understanding smart cities: An integrative framework. In *System Science (HICSS), 2012 45th Hawaii International Conference on*, pages 2289–2297, Wailea, Maui, Hawaii, Jan 2012.
- [16] W. Christaller and C. W. Baskin. *Central Places in Southern Germany*. Prentice-Hall, Englewood Cliffs, N.J., 1966.
- [17] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson. *Introduction to Algorithms*. McGraw-Hill Higher Education, Cambridge, Massachusetts, 2nd edition, 2001.
- [18] C. A. Costa. *Software Infrastructure for Ubiquitous Computing: A Context-aware Service-based Approach*. VDM Verlag, Saarbrücken, Germany, 2009.
- [19] J. S. de Oliveira Neto and S. T. Kofuji. *Inclusive Smart City: An Exploratory Study*, pages 456–465. Springer International Publishing, Cham, 2016.
- [20] M. Deakin. Smart cities: the state-of-the-art and governance challenge. *Triple Helix*, 1(1), 2014.
- [21] A. K. Dey. Understanding and using context. *Personal Ubiquitous Comput.*, 5(1):4–7, Jan. 2001.
- [22] M. C. Domingo. An overview of the internet of things for people with disabilities. *Journal of Network and Computer Applications*, 35(2):584 – 596, 2012. Simulation and Testbeds.
- [23] H. Fernandes, V. Filipe, P. Costa, and J. Barroso. Location based services for the blind supported by rfid technology. *Procedia Computer Science*, 27(0):2 – 8, 2014. 5th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion, DSAI 2013.

- [24] K. J. Fietkiewicz, A. Mainka, and W. G. Stock. e-government in cities of the knowledge society. an empirical investigation of smart cities' governmental websites. *Government Information Quarterly*, 2016.
- [25] M. Hadley and P. Sandoz. JAX-RS: Java API for RESTful Web Services, 2008.
- [26] G. Kbar, S. Aly, I. Elsharawy, A. Bhatia, N. Alhasan, and R. Enriquez. Smart Help at the Workplace for Persons with Disabilities (SHW-PWD). *International Journal of Computer, Control, Quantum and Information Engineering*, 9(1):84 – 90, 2015.
- [27] E. Kim. Oasis advancing open standards for the information society. Disponível em: <http://docs.oasis-open.org/wsqm/WS-Quality-Factors/v1.0/WS-Quality-Factors-v1.0.html>. Acesso em: 15 janeiro de 2016, 2012. Web Services Quality Factors Version 1.0.
- [28] H. K. Kim. Convergence agent model for developing u-healthcare systems. *Future Generation Computer Systems*, 35(0):39 – 48, 2014. Special Section: Integration of Cloud Computing and Body Sensor Networks; Guest Editors: Giancarlo Fortino and Mukaddim Pathan.
- [29] N. Komninos, M. Pallot, and H. Schaffers. Special issue on smart cities and the future internet in europe. *Journal of the Knowledge Economy*, 4(2):119–134, 2013.
- [30] T. D. Le, S. H. Kim, M. H. Nguyen, D. Kim, S. Y. Shin, K. E. Lee, and R. d. R. Righi. Epc information services with No-SQL datastore for the Internet of Things. In *RFID (IEEE RFID), 2014 IEEE International Conference on*, pages 47–54, Orlando, FL, USA, April 2014. IEEE.
- [31] J. H. Lee, R. Phaal, and S.-H. Lee. An integrated service-device-technology roadmap for smart city development. *Technological Forecasting and Social Change*, 80(2):286 – 306, 2013. World problems, Emerging technologies and Creative Enterprise.
- [32] M. Martin and P. Nurmi. A generic large scale simulator for ubiquitous computing. In *Third Annual International Conference on Mobile and Ubiquitous Systems: Networking & Services, 2006 (MobiQuitous 2006)*, San Jose, California, USA, Jul 2006. IEEE Computer Society.
- [33] M. V. L. Martins. Frametrail: um framework para o desenvolvimento de aplicações orientadas a trilhas. Master thesis, University of Vale do Rio dos Sinos (Unisinos), Applied Computing Graduate Program (PIPCA), São Leopoldo, Brazil, 2011.
- [34] A. Meijer. *Smart City Governance: A Local Emergent Perspective*, pages 73–85. Springer International Publishing, Cham, 2016.
- [35] S. Mirri, C. Prandi, P. Salomoni, F. Callegati, and A. Campi. On combining crowdsourcing, sensing and open data for an accessible smart city. In *Next Generation Mobile Apps, Services and Technologies (NGMAST), 2014 Eighth International Conference on*, pages 294–299, Oxford, Sept 2014. IEEE.
- [36] I. Molyneaux. The art of application performance testing. from strategy to tools. *SciTech Book News*, 2014.
- [37] R. Petrolo, V. Loscrí, and N. Mitton. Towards a smart city based on cloud of things. In *Proceedings of the 2014 ACM International Workshop on Wireless and Mobile*

Technologies for Smart Cities, WiMobCity '14, pages 61–66, New York, NY, USA, 2014. ACM.

- [38] W. Qin, Y. Shi, and Y. Suo. Ontology-based context-aware middleware for smart spaces. *Tsinghua Science & Technology*, 12(6):707–713, 2007.
- [39] J. d. M. Quintana, E. Z. Ferreira, S. S. C. Santos, M. T. Pelzer, M. J. Lopes, and E. J. L. Barros. A utilização da Classificação Internacional de Funcionalidade, Incapacidade e Saúde no cuidado aos idosos. *Revista de Enfermagem Referência*, serIV:145 – 152, 03 2014.
- [40] H. Schaffers, N. Komninos, M. Pallot, B. Trousse, M. Nilsson, and A. Oliveira. *Smart Cities and the Future Internet: Towards Cooperation Frameworks for Open Innovation*, volume 6656 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, London New York, 2011.
- [41] D. H. Shin. Ubiquitous computing acceptance model: end user concern about security, privacy and risk. *International Journal of Mobile Communications*, 8(2):169–186, 2010.
- [42] M. Sniedovich. Dijkstra’s algorithm revisited: the dynamic programming connexion. *Control and cybernetics*, 35(3):599, 2006.
- [43] L. T. Sánchez, D. C. Ranasinghe, M. Harrison, and D. McFarlane. Adding sense to the internet of things. *Personal Ubiquitous Comput.*, 16(3):291–308, Mar. 2012.
- [44] J. Tavares, J. L. Barbosa, I. Cardoso, C. A. Costa, A. C. Yamin, and R. Real. Hefestos: an intelligent system applied to ubiquitous accessibility. *Universal Access in the Information Society*, pages 1–19, 2015.
- [45] G. C. Vanderheiden. Ubiquitous accessibility: Building access features directly into the network to allow anyone, anywhere access to ubiquitous computing environments. In C. Stephanidis, editor, *Universal Access in Human-Computer Interaction. Intelligent and Ubiquitous Interaction Environments*, volume 5615 of *Lecture Notes in Computer Science*, pages 432–437, San Diego, CA, USA, 2009. Springer Berlin Heidelberg.
- [46] A. Wagner. Um modelo para gerenciamento de perfis de entidade através de inferência em trilhas. Master thesis, University of Vale do Rio dos Sinos (Unisinos), Applied Computing Graduate Program (PIPCA), São Leopoldo, Brazil, 2013.
- [47] D. Washburn, U. Sindhu, S. Balaouras, R. Dines, N. Hayes, and L. Nelson. Helping cios understand “smart city” initiatives: defining the smart city, its drivers, and the role of the cio. *Forrester Research*, 2010.
- [48] H. Yang and M. Tate. A descriptive literature review and classification of cloud computing research. *Communications of the Association for Information Systems*, 31(2):35–60, 2012.
- [49] M. Zhou, R. Zhang, D. Zeng, and W. Qian. Services in the cloud computing era: A survey. In *Universal Communication Symposium (IUCS), 2010 4th International*, pages 40–46, Beijing, China, Oct 2010.

- [50] S. Zygiaris. Smart city reference model: Assisting planners to conceptualize the building of smart city innovation ecosystems. *Journal of the Knowledge Economy*, 4(2):217–231, 2013.