



**UNIVERSIDADE DO ESTADO DO RIO GRANDE DO NORTE
UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO**



SUELLEM STEPHANNE FERNANDES QUEIROZ

**SOS MÓVEL: SISTEMA PARA AUXILIAR PESSOAS NA
SOLICITAÇÃO DE SOCORRO**

**MOSSORÓ - RN
2016**

SUELLEM STEPHANNE FERNANDES QUEIROZ

**SOS MÓVEL: SISTEMA PARA AUXILIAR PESSOAS NA
SOLICITAÇÃO DE SOCORRO**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação - associação ampla entre a Universidade do Estado do Rio Grande do Norte e a Universidade Federal Rural do Semi-Árido, para a obtenção do título de Mestre em Ciência da Computação.

Orientador: Prof^ª D.Sc. Cícilia Raquel Maia Leite

Coorientador: Prof^º D.Sc. Pedro Fernandes Ribeiro Neto

**MOSSORÓ - RN
2016**

**Catálogo da Publicação na Fonte.
Universidade do Estado do Rio Grande do Norte.**

Queiroz, Suellem Stephanie Fernandes

SOS Móvel: Sistema Para Auxiliar Pessoas Na Solicitação De Socorro / Suellem Stephanie Fernandes Queiroz – Mossoró, RN, 2016.

100 f.

Orientador (a): Prof. D.Sc. Cíclia Raquel Maia Leite

Dissertação (Mestrado) Universidade do Estado do Rio Grande do Norte. Campus Central. Programa de Pós-Graduação em Ciência da Computação - associação ampla entre a Universidade do Estado do Rio Grande do Norte e a Universidade Federal Rural do Semi-Árido,

1. Solicitação de socorro – Sistema. 2. Detecção de risco - M-Health. 3. Agentes inteligentes. I. Leite, Cíclia Raquel Maia. II. Universidade do Estado do Rio Grande do Norte. III. UFERSA. IV. Título.

UERN/ BC

CDD 004.11


SUELLEM STEPHANNE FERNANDES QUEIROZ

SOS MÓVEL: SISTEMA PARA AUXILIAR PESSOAS NA SOLICITAÇÃO DE
SOCORRO

Dissertação apresentada ao
Programa de Pós-Graduação em
Ciência da Computação para a
obtenção do título de Mestre em
Ciência da Computação.

APROVADA EM: ____/____/____.


BANCA EXAMINADORA



Prof. Dra. Cicilia Raquel Maia Leite (Presidente e Orientadora)



Prof. Dr. Pedro Fernandes Ribeiro Neto (Coorientador/UERN)



Prof. Dr. Francisco Milton Mendes Neto (UERN)



Prof. Dra. Suélia de Siqueira Rodrigues Fleury Rosa (UNB)

Dedico este trabalho a Deus, por ter me concebido sabedoria para concluí-lo; e a minha família, que esteve presente em todos os momentos dessa etapa.

AGRADECIMENTOS

Todo o mérito para conclusão deste trabalho dedico a Deus, pois mais uma vez mostrou seu amor irrefutável a mim. Agradeço porque Ele me deu a força precisa quando o sentimento maior era de desistência e incapacidade. Como diz em Salmos 126:3: “Grandes coisas fez o Senhor por nós, pelas quais estamos alegres”.

Meus sinceros agradecimentos a todos aqueles que de alguma forma doaram um pouco de si para que a conclusão deste trabalho se tornasse possível: à minha mãe, principalmente, que durante toda a minha vida se esforçou para me proporcionar uma boa educação e me repassar valores que vou levar pelo resto da minha vida. Do mesmo modo meu pai, minha irmã e minha família materna, que sempre estiveram ao meu lado, torcendo pela minha conquista e me incluindo em suas orações. Amo a todos, de todo o coração e com todo meu carinho.

Agradeço também a minhas fiéis amigas Sâmua, Willyane e Mariza, que estão sempre acompanhando de perto minhas preocupações e angústias, me dando forças para continuar e com apoio constante, seja por ligações, mensagens, orações ou palavras confortantes. Também agradeço a meu querido amigo Kayo que esteve me acompanhando e dividindo comigo os momentos bons e os difíceis do mestrado e graduação. A Érico, por companherismo e paciência ao longo da construção do projeto, e a Davi, que desprende de tanto tempo para tirar dúvidas e foi um dos responsáveis por esse trabalho ter chegado até aqui. Não teria conseguido sem a ajuda de todos vocês. A todos, meu sincero agradecimento por tudo que fizeram por mim, meu sentimento por vocês é sincero.

Agradeço ainda à orientadora Cicília Maia e meu coorientador Pedro Fernandes, que ajudaram a direcionar as ideias para o alcance de meus objetivos do trabalho. Agradeço também seu auxílio, disponibilidade e pelo fornecimento do material para pesquisa do tema. Meus agradecimentos também a todos os professores do Programa de Pós-Graduação em Ciência da Computação pela transmissão segura dos conhecimentos.

É difícil agradecer a todos que de algum modo, nos momentos serenos ou apreensivos, fizeram ou fazem parte da minha vida, por isso agradeço à vocês, mesmo não estando citados aqui, contribuíram para a conclusão desta etapa tão importante.

"Mesmo com todos os espinhos, nunca deixou de ser flor."

Isabella Martins.

RESUMO

O *M-Health* pode desempenhar um papel essencial no serviço de emergência médica. Devido aos índices elevados de acidentes, incidentes e ocorrências, qualquer um está vulnerável e pode necessitar de informações sobre centros de saúde mais próximos para o socorro. Na busca de reduzir o alto índice de ocorrências fatais em acidentes e incidentes quaisquer, medidas tecnológicas passaram a ser empregadas para contatar rapidamente socorro. O SOS Móvel é um sistema para auxiliar pessoas na solicitação de socorro. O sistema faz reconhecimento de atividades, utiliza sensores móveis com sensibilidade ao contexto e agentes inteligentes para disparo de solicitações de socorro e detecção de situações de risco. O sistema é composto por uma parte web e uma móvel. O aplicativo móvel é utilizado ativamente pelo usuário, enquanto que a interface web é utilizada pelos familiares ou gestores da saúde que podem monitorar o usuário e visualizar seu histórico de alertas e localização. Os experimentos realizados com o SOS Móvel em um ambiente de simulações permitiram concluir que o sistema apresentou desempenho satisfatório para detectar riscos e enviar solicitações de socorro, possuindo também grande potencial para aplicação em diversas situações.

Palavras-chave: Solicitação de socorro, Detecção de risco, M-Health, Sensores

ABSTRACT

The M-Health can play a key role in emergency medical service. Due to the high rates of accidents, incidents and occurrences, anyone is vulnerable and may need information on health centers closer to the rescue. In seeking to reduce the high rate of fatalities in accidents and any incidents, technological measures started to be used to quickly contact help. The Mobile SOS is a system to help people in distress request. The system recognition activities, using mobile sensors with sensitivity to context and intelligent agents to trigger relief requests and detection of risk situations. The system comprises a web portion and a mobile. The mobile application is actively used by the user, while the web interface is used by family or health managers can monitor the user and view its alert history and location. The experiments performed with the Mobile SOS in a simulated environment showed that the system showed satisfactory performance to identify risks and send rescue requests, but also has great potential for application in various situations.

Key-words: Request for help, Risk detection, M-Health, Sensors

LISTA DE ALGORITMOS

1	Algoritmo que monitora os sensores do <i>Smartphone</i>	59
2	Algoritmo de checagem do <i>status</i>	61

LISTA DE FIGURAS

Figura 1 – Relação entre computação ubíqua, móvel e pervasiva	17
Figura 2 – Diferença entre os tipos de quadro de coordenadas	21
Figura 3 – Orientações <i>alfa</i> , <i>beta</i> e <i>gama</i>	21
Figura 4 – Diferença entre o acelerômetro e o giroscópio	23
Figura 5 – Arquitetura do sistema	26
Figura 6 – Aplicativo Virtual Check up	27
Figura 7 – Equipamento MedMinder	28
Figura 8 – Telas da aplicação EmergentHelper V2	30
Figura 9 – Computação móvel, ubíqua e pervasiva relacionada	31
Figura 10 – Exemplos de tecnologias assistivas para diferentes deficiências.	37
Figura 11 – Visão conceitual de um agente	41
Figura 12 – Interface da ferramenta <i>TeleOrthoPaedics</i>	43
Figura 13 – Visão geral do SCIADS	44
Figura 14 – Interface da ferramenta <i>CrowdHelp</i>	45
Figura 15 – Interface da ferramenta desenvolvida	46
Figura 16 – Exemplo de caída simulada	47
Figura 17 – Página inicial do aplicativo	48
Figura 18 – Aperfeiçoamento da diferenciação entre carros e objetos	49
Figura 19 – Telas da aplicação	50
Figura 20 – Arquitetura da aplicação	51
Figura 21 – Telas principais	52
Figura 22 – Visão geral do SOS Móvel	57
Figura 23 – Modelo de agentes visual	64
Figura 24 – Modelo de tarefas	64
Figura 25 – Modelo de papéis	65
Figura 26 – Fluxograma do processo de realização de cadastro	66
Figura 27 – Telas principais do SOS Móvel	67
Figura 28 – Interface SOS <i>Web</i>	68
Figura 29 – Notificação de socorro	68
Figura 30 – Representação dos movimentos do acelerômetro nos eixos	72
Figura 31 – Gráfico de rotação para o teste 1 da atividade Parado	74
Figura 32 – Gráfico de rotação para o teste 20 da atividade Parado	74
Figura 33 – Gráfico com rotação positiva para todos os testes da atividade Caminhando	76
Figura 34 – Gráfico representando o teste 3 e 4 de rotação para Correndo	78
Figura 35 – Gráfico representando o teste 18 sem rotação da atividade Bicicleta	80

Figura 36 – Gráfico representando o teste 7 para atividade Carro	81
Figura 37 – Gráfico representando o teste 1 para o caso de capotagem	83
Figura 38 – Gráfico representando o teste 3 para o caso de capotagem	83
Figura 39 – Gráficos de porcentagem de erros e acertos para cada atividade . . .	85

LISTA DE TABELAS

Tabela 1 – Dimensões semânticas do Contexto	35
Tabela 2 – Tabela de comparação entre os trabalhos apresentados	53
Tabela 3 – Modelo conceitual do agente	63
Tabela 4 – Dados obtidos estando sem movimento	73
Tabela 5 – Dados obtidos estando caminhando	76
Tabela 6 – Dados obtidos estando correndo	77
Tabela 7 – Dados obtidos da bicicleta	79
Tabela 8 – Dados obtidos estando no caso para o caso de colisão	82
Tabela 9 – Dados obtidos estando no caso para o caso de capotamento	84

LISTA DE SIGLAS

ADA *American with Disabilities Act*

AML *Agent Modeling Language*

API *Application Programming Interface*

APK *Android Application Package File*

AVD *Android Virtual Device Manager*

BD Banco de dados

COOPI *Cooperação Internacional*

CSM *Central de Supervisão Médica*

CSR *Central de Saúde Residencial*

EMR *Electronic Medical Record*

FPS *Frames Por Segundo*

GMM *Gaussian Mixture Models*

GPS *Global Positioning System*

HTML *HyperText Markup Language*

IA *Inteligência Artificial*

IBGE *Instituto Brasileiro de Geografia e Estatística*

IDE *Integrated Development Environment*

IP *Internet Protocol*

JRE *Java Runtime Environment*

JSP *JavaServer Pages*

MARG *Magnetic, Angular Rate and Gravity*

MEMS *Micro-Electro-Mechanical Systems*

NFC *Near-Field Communication*

OBD-II *On-Board Diagnostic System II*

OMS Organização Mundial de Saúde

SAMU Serviço de Atendimento Móvel de Urgência

SCIADS Sistema Computacional Inteligente de Assistência Domiciliar à Saúde

SDK *Software Development Kit*

SGBD Sistema Gerenciador do Banco de Dados

SO Sistema Operacional

SMS *Short Message Service*

SPG Solucionador de Problemas Gerais

SPOT *Small Programmable Object Technology*

SQL *Structured Query Language*

TA Tecnologias Assistivas

TAMA *Treatment Advice from Mobile Alerts*

TEA Transtorno do Espectro Autista

TIC Tecnologias da Informação e Comunicação

UNICEF Fundo das Nações Unidas para a Infância

VANET *Veicular Ad Hoc Networks*

XML *EXtensible Markup Language*

SUMÁRIO

1	Introdução	13
1.1	Motivação	14
1.2	Objetivo	15
1.3	Organização do Documento	15
2	Fundamentação Teórica	16
2.1	Computação Ubíqua	16
2.1.1	Computação Móvel	18
2.1.1.1	Sensores	19
2.1.1.1.1	Acelerômetro	22
2.1.1.1.2	Giroscópio	23
2.1.1.2	M-Health	24
2.1.2	Computação Pervasiva	29
2.1.3	Sensibilidade ao Contexto	31
2.2	Tecnologias Assistivas	36
2.3	Agentes Inteligentes	38
3	Trabalhos Relacionados	43
4	SOS Móvel: Sistema para Auxiliar no Pedido de Socorro e Detecção Automática de Situações de Risco	55
4.1	Visão Geral	55
4.2	Etapas	56
4.2.1	Configuração do ambiente	56
4.2.2	Aquisição de dados	57
4.2.3	Pré-processamento	57
4.2.4	Processamento e Pós-Processamento	58
4.2.4.1	Modelagem do Agente	62
4.3	Implementação do SOS Móvel	65
4.3.1	Funcionalidades do SOS Móvel	65
4.3.2	Ferramentas e tecnologias	69
5	Estudo de caso: Aplicação do SOS Móvel em ambiente de testes	71
5.1	Descrição do experimento	71
5.2	Avaliação do experimento	73
5.2.1	Caso 1: Parado	73
5.2.2	Caso 2: Caminhando	75
5.2.3	Caso 3: Correndo	76
5.2.4	Caso 4: Bicicleta	78
5.2.5	Caso 5: Carro	80

5.2.5.1	Caso: Colisão	80
5.2.5.2	Caso: Capotamento	81
5.3	Discussões e Resultados	85
6	Considerações Finais	87
REFERÊNCIAS		89
APÊNDICES		98
APÊNDICE A Modelo de objetos e recursos do agente		99

1 INTRODUÇÃO

A crescente evolução de tecnologias na área médica tem propiciado benefícios e facilidades para aqueles que necessitam de atendimento especializado. A telemedicina, serviço importante para a área da saúde, é utilizada amplamente para representar o uso das modernas Tecnologias da Informação e Comunicação (TIC), que dispõem de serviços, de treinamento e de conhecimento em saúde para fornecimento de informação e atenção médica a pacientes e especialistas na área da saúde que se encontram distantes fisicamente (RIFTIN, 2013).

As grandes aliadas, E-Health e M-Health, têm apresentado grande potencial para auxiliar o sistema de saúde a atingir o objetivo de ampliar a qualidade do atendimento, aprimorar a saúde da população e diminuir o custo para o cuidado médico-pessoal (Faculdade de Medicina da Universidade do Porto, 2014).

O M-Health ou Saúde Móvel é um termo utilizado para a prática da Medicina e da saúde pública em dispositivos móveis. O termo é comumente utilizado em razão do uso de dispositivos de comunicação móvel para exercer serviços de saúde onipresentes. Com o advento dos telefones com tecnologia avançada e inteligente (Smartphones), nos últimos anos, o M-Health tem se tornado mais abrangente devido à utilização de dispositivos móveis para a execução de serviços da área médica (MEDEIROS, 2015).

A propagação da utilização de Smartphones possibilitou a agregação de tecnologias como a computação ubíqua, que dispõe de dispositivos com a capacidade de imersão nos elementos cotidianos, provendo serviços de forma transparente aos usuários (LOUREIRO et al., 2009).

Com a evolução das pesquisas e do desenvolvimento da tecnologia, surgiram conceitos importantes que passaram a compor a computação ubíqua, como a computação móvel e a computação pervasiva (ARAUJO, 2003). Enquanto a computação móvel foca na ampliação da mobilidade de serviços entre os ambientes, a computação pervasiva centra-se na capacidade que os dispositivos computacionais têm de serem vinculados e embutidos no ambiente de forma onipresente (SANTOS et al., 2012).

Dentre os diferentes domínios da computação ubíqua, destaca-se o de adaptação de conteúdo. A sensibilidade ao contexto é um meio de comunicação entre os sistemas e seus usuários, pois é a partir da compreensão de um contexto que um sistema pode, em meio a variadas circunstâncias, modificar suas ações, fornecer diferentes informações e adaptar-se às necessidades de um contexto. Sistemas sensíveis ao contexto têm a capacidade de modificar seus comportamentos baseados nas informações do estado que o cercam (GONZALEZ; BRÉZILLON, 2008).

Conforme Santos et al. (2012), para os sistemas de computação ubíqua se tornarem invisíveis, têm de ser sensíveis ao contexto. Um requisito importante tanto

para a sensibilidade ao contexto quanto para a computação ubíqua é a detecção contínua de estados para a classificação e o processamento multitarefa. Tais ações são realizadas por sensores (ORTIZ, 2015).

Os sensores têm sido utilizados amplamente em dispositivos móveis desde que os mesmos foram criados. Gradativamente, novas tecnologias são empregadas aos Smartphones, tornando-os cada vez mais autônomos. Os sensores móveis fazem a detecção do ambiente ao seu redor e são parte do sensoriamento onipresente que integra outras áreas, como a de agentes (PEREIRA et al., 2013). Além disso, podem estar autocontidos em dispositivos, integrar-se ao hardware ou podem ser interconectados como nós, formando uma rede de sensores (ORTIZ, 2015).

De acordo com Frigo et al. (2004), a tecnologia de agentes tende a ser uma das principais tecnologias utilizadas na computação ubíqua para reconhecimento e análise de informação de contexto de ambientes distintos.

1.1 MOTIVAÇÃO

Em escala mundial, acidentes e incidentes são causas de dizimações diárias em milhares de pessoas ou são os responsáveis por causar mutilações e sequelas que aprisionam os indivíduos às suas enfermidades Silva (2015). As lesões decorrentes de acidentes, desde os mais corriqueiros até os mais graves, e violências são definidas ou classificadas como causas externas de morbidade e mortalidade.

Conforme dados coletados da Secretaria de Vigilância em Saúde (2013) sobre a distribuição de atendimentos devido a violências e acidentes de urgências e emergências, constatou-se que o tipo de ocorrência mais frequente de acidente no Brasil diz respeito a quedas, com 30,9% da porcentagem; seguido por acidentes de transporte, com 26,2%; 8,4% de agressões; 1,9% de queimaduras; 0,9% de lesões e 0,1% intervenções. Demais acidentes e incidentes, como afogamento, ferimento por arma de fogo, choques, dentre outros, contabilizaram 31,6% das ocorrências totais.

Segundo a Programa Nacional de Prevenção de Acidentes (2010), conforme dados colhidos das regiões que constituem a Organização Mundial de Saúde (OMS), (193 estados-membros), estima-se que morrem, por ano, 5 milhões de pessoas devido a acidentes. Os dados afirmam ainda que a maior parte dessas mortes ocorrem em populações com idades entre 15 e 44 anos e suas principais causas são acidentes de viação (em vias de trânsito), quedas, afogamentos, queimaduras e intoxicações.

Os acidentes intencionais ou não-intencionais, que resultam em fatalidades, podem ocorrer devido a um conjunto de circunstâncias que são decisivas para a sobrevivência de um indivíduo. Um dos principais fatores, dentre os demais, é a espera pelo socorro. O tempo gasto desde contatar um Serviço de Atendimento Móvel de Urgência (SAMU) até a prestação de socorro a uma pessoa pode ser um fator decisivo entre a vida e a morte.

Cunha (2008) afirma que, para pessoas que sofreram parada cardíaca, por exemplo, cada minuto de atraso para resgate reduz a taxa de sobrevivência de 7% a 10%. Se o procedimento de ressuscitação cardiopulmonar for realizado dentro de 5 minutos, a porcentagem de sobrevivência sobe para 30%. Caso contrário, diminui para 7%, com redução gradual entre 3 e 4% por minuto. As ações de socorro durante os minutos iniciais de atendimento a uma emergência são fatores determinantes em relação à sobrevivência de uma vítima.

Com o maior envolvimento dos sistemas de comunicação atuais, é possível utilizar recursos tecnológicos e assistenciais para permitir a promoção de cuidados médicos, como também agilizar os processos de socorro. Na busca de reduzir o alto índice de ocorrências fatais em acidentes e incidentes quaisquer, medidas tecnológicas passaram a ser empregadas para contatar rapidamente socorro. Devido a sua rápida capacidade de comunicação entre demais dispositivos, acesso à rede e possibilidade de agregação a outras tecnologias, os Smartphones tornaram-se facilitadores que atuam de modo autônomo e inteligente no âmbito de urgências e emergências.

Algumas medidas tecnológicas já vêm sendo aplicadas à área médica a fim de amenizar os elevados índices de fatalidades em acidentes, porém, a maioria delas não se encontram disponíveis totalmente gratuitas. É notória a necessidade do desenvolvimento de um meio que venha ajudar na solicitação de socorro a pessoas envolvidas em acidentes.

1.2 OBJETIVO

Diante do contexto abordado, o objetivo deste trabalho é desenvolver um sistema para auxiliar no pedido de socorro e detecção automática de situações de risco. O sistema funcionará de modo similar aos botões de pânico padrões, entretanto, com a agregação de reconhecimento de atividades, sensores móveis com sensibilidade ao contexto, e agentes para disparo de solicitações de socorro e reconhecimento de possíveis acidentes. O sistema é desenvolvido para utilização em Sistema Operacional (SO) Android e possui também uma interface de monitoramento e recebimento de alertas web.

1.3 ORGANIZAÇÃO DO DOCUMENTO

O trabalho encontra-se organizado da seguinte forma: o capítulo 2 apresenta os fundamentos teóricos sobre M-Health, computação ubíqua, seguida da computação móvel e pervasiva, a sensibilidade ao contexto, tecnologias assistivas, sensores e agentes inteligentes. Tais áreas foram utilizadas para construção da base cognitiva da dissertação. No capítulo 3 são apresentados os trabalhos relacionados e as inovações nas áreas similares. Já no capítulo 4, é abordada toda a especificação deste trabalho. No capítulo 5 é exposto o SOS Móvel em ambiente de teste e exibidos seus resultados. No capítulo 6, são apresentadas as discussões finais, conclusões e perspectivas futuras.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo é destinado à apresentação de toda a teoria na qual o trabalho está embasado, abordando conceitos teóricos necessários ao desenvolvimento do mesmo, suporte para estudos e coleta de informações qualitativas. A fundamentação teórica apresentada foi a base para a elaboração, a análise e a interpretação dos dados obtidos, além da justificativa do problema em estudo.

- **Computação Ubíqua:** discute as definições e os conceitos importantes sobre esta área da computação e aborda mais detalhadamente a computação móvel, destacando os sensores e a M-Health, a computação pervasiva e a sensibilidade ao contexto;
- **Tecnologias Assistivas:** aborda as definições e subcategorias que compõem as tecnologias assistivas;
- **Agentes inteligentes:** apresenta as definições, caracterização, conceitos importantes, tipos e utilização no trabalho em questão.

2.1 COMPUTAÇÃO UBÍQUA

A utilização de componentes eletrônicos e digitais teve um grande crescimento à medida que o preço dos mesmos teve um declínio significativo. Gradativamente, os grandes desenvolvedores estão integrando uma quantidade maior de dispositivos específicos aos seus produtos finais. Conforme Abowd e Mynatt (2000), computadores trabalhando em conjunto proveem serviços mais significativos do que isoladamente.

De acordo com Endler (2001), o termo ubíquo é utilizado para expressar que a computação e os computadores estão embutidos na vida das pessoas e estão presentes em qualquer lugar, ao contrário da realidade virtual, onde as pessoas são introduzidas em esferas virtuais geradas pelos próprios computadores. Desse modo, a computação ubíqua tem como objetivo retirar os computadores do foco central dos usuários e movê-los para uma esfera invisível, onde os mesmos são utilizados de forma subconsciente.

Idealizado por Weiser (1991)), o termo computação ubíqua é introduzido pela primeira vez em *The Computer for the 21st Century* com a frase “as mais profundas tecnologias são aquelas que desaparecem”, na qual define-a como um meio para tornar o uso do computador disponível em qualquer ambiente físico de forma invisível ao usuário, ou seja, o computador se integra à vida das pessoas de modo que não seja percebido, mas seja utilizado plenamente.

Na Computação Ubíqua, o objeto-computador é uma extensão das habilidades humanas, auxiliando os usuários a realizarem tarefas do cotidiano ou até mesmo

aquelas que não são possíveis de executar. Nesse sentido, serviços e recursos estão sempre disponíveis ao usuário, em qualquer lugar e a qualquer tempo, independente de dispositivo (MONTEIRO; GOMES, 2015).

Mark Weiser idealizou ambientes físicos com dispositivos computacionais integrados que auxiliariam pessoas na execução de suas tarefas diárias ao fornecer serviços e informações continuamente e de forma transparente. A ideia inicial previa que a interação entre os computadores e os usuários se distanciaria dos dispositivos tradicionais propriamente ditos, como hardwares, teclado, mouse e monitor, e se aproximaria de uma interação, na qual as pessoas falam, gesticulam e escrevem para interagirem entre si (LOPES et al., 2008).

Em Araujo (2003), é apresentado um conjunto de características que descrevem um sistema como sendo ubíquo. São elas: a informação tem seu acesso através de múltiplos dispositivos heterogêneos; os dispositivos se autorrelacionam; determinadas tarefas são executadas autonomamente; constantemente a aplicação acompanha o usuário e responde a mudanças no ambiente; o ambiente troca informações com os dispositivos e vice-versa; e os dispositivos distintos apresentam visões diferentes da mesma aplicação. Todavia, algumas das características destacadas são também observadas em dois outros modelos computacionais: a computação móvel e a computação pervasiva.

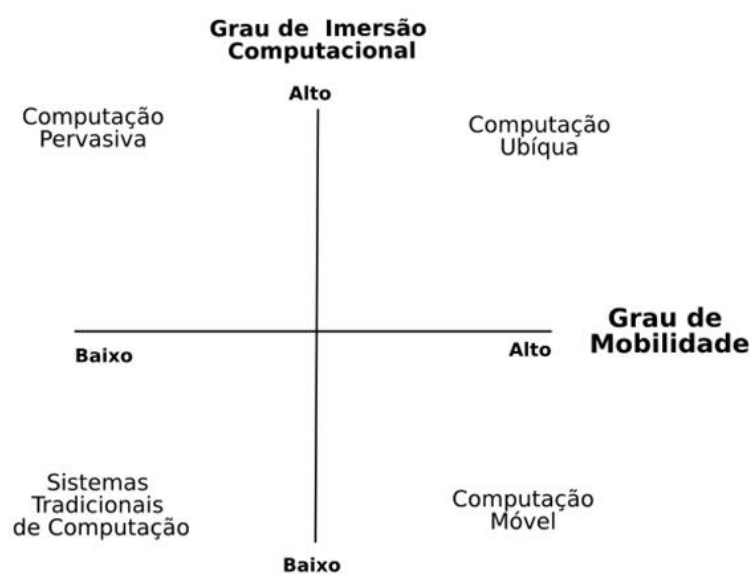


Figura 1 – Relação entre computação ubíqua, móvel e pervasiva

Fonte: Adaptado de Lyytinen e Yoo (2002)

Na Figura 1, é demonstrada a relação entre as duas vertentes da computação ubíqua. Pode-se notar que, quanto maior o grau de mobilidade existente, mais a aplicação se aproxima da computação móvel. Entretanto, para que seja ubíqua, além de possuir alta mobilidade, também é preciso ter um grau elevado de imersão computacional, ou seja, ser invisível ao usuário. De forma contrária, se possuir baixo nível de

mobilidade, é caracterizada como pervasiva. Os sistemas tradicionais de computação são caracterizados como aqueles que possuem baixo nível de imersão computacional e de mobilidade.

O impacto social dos computadores a qualquer hora e em qualquer local é análogo às duas tecnologias destacadas anteriormente, que se tornaram ubíquas. A computação móvel se distingue das demais devido a sua capacidade de acompanhar o usuário mesmo em constante movimento, e continuar provendo serviços computacionais. Contudo, o modelo computacional não se modifica quando há movimentação do usuário, o que faz concluir que a computação móvel não é capaz de perceber informações do contexto atual e adequar-se. Em contrapartida, a computação pervasiva busca obter e oferecer serviços computacionais no ambiente, de modo que não seja perceptível ao usuário.

O modelo computacional da mesma se adequa dinamicamente e age de forma proativa. Dessa forma, a computação ubíqua integra características da computação móvel e da computação pervasiva, integrando mobilidade com sensibilidade ao contexto (LYYTINEN; YOO, 2002).

2.1.1 Computação Móvel

Atualmente, a informação pode ser acessada de praticamente todo lugar e a todo momento. Nota-se uma grande evolução e popularização de dispositivos móveis, além de ampla difusão das redes sem fio. Os dispositivos com capacidade de comunicação com redes fixas tradicionais e com demais dispositivos móveis induzem ao conceito de computação móvel, que é uma área da tecnologia que amplia o domínio da computação distribuída devido fazer uso da comunicação sem fio para eliminar a limitação da mobilidade (TONIN; GOLDMAN, 2012).

Foi através dos dispositivos móveis que, segundo Voss et al. (2015), surgiu uma quarta revolução computacional, ampliando o conceito tradicional da computação distribuída e eliminando a necessidade de se permanecer conectado a uma infraestrutura estática ou fixa. A computação móvel pode ser representada como um novo paradigma computacional que concede aos usuários processamento, mobilidade e comunicação sem fio em seus dispositivos móveis (MATEUS; LOUREIRO, 1998).

Para um dispositivo ser dito como móvel, é preciso que o mesmo tenha capacidade de realizar processamento, trocar informações via uma rede, possuir fácil manipulação e ser capaz de ser transportado facilmente pelo utilizador. Para esta última característica, é importante ressaltar que o dispositivo deve possuir tamanho reduzido e não necessitar de cabos para conectá-lo a uma fonte de energia elétrica ou uma rede de internet (VOSS et al., 2015).

A evolução da computação móvel contempla diversos fatos marcantes, desde a sua criação até atualmente. As duas grandes guerras mundiais foram motivos relevantes para o rápido avanço da comunicação. A tecnologia digital veio acelerar ainda mais

esse processo. Os computadores surgiram como uma terceira geração dos sistemas de comunicação. As demais evoluções significativas, como a criação de centros de processamento de dados em meados de 1960, o surgimento dos terminais na década de 70 e as redes de computadores nos anos 80, foram alicerces para a computação móvel (MATEUS; LOUREIRO, 1998).

Ito et al. (2003) afirma que um dos principais objetivos da computação móvel é a mobilidade dos computadores portáteis, pois o ambiente de operação do usuário é constantemente dinâmico. Todavia, a mobilidade também introduz problemas e desafios que são inexistentes em ambientes fixos. Diversos problemas que são praticamente resolvidos na computação tradicional tornam-se irresolúveis em ambientes móveis.

Os principais problemas que a mobilidade apresenta se encontram desde a comunicação com outros dispositivos, até a duração da bateria dessa unidade. São alguns exemplos destacados por Voss et al. (2015):

- Interface com demais dispositivos móveis: as formas de interação com alguns dispositivos são diferentes, variando de um dispositivo ao outro. Podem-se encontrar limitações relacionadas a telas menores, saídas sonoras e visuais ou componentes inexistentes, como mouse e teclados;
- Características do meio: redes sem fio têm limitada largura de banda com elevadas taxas de erro de transmissão devido a desconexões frequentes ou interferências;
- Capacidade dos dispositivos móveis: possuem recursos restritos de memória e processamento;
- Adaptação: um dispositivo móvel pode utilizar serviços e realizar comunicação independente de sua localização física ou do fato de estar em movimento. No entanto, os dispositivos de computação móvel não possuem a capacidade de obter informações sobre o ambiente ou o contexto, no qual estão inseridos, e modificá-los;
- Segurança: em virtude da ausência de uma rede física, as redes sem fio são mais sujeitas a ataques maliciosos devido a sua propagação wireless, o que possibilita a interceptação de suas informações;
- Energia: devido a sua mobilidade, os dispositivos móveis têm suas próprias fontes de energia, porém, estas ainda possuem curta duração.

2.1.1.1 Sensores

Os Smartphones têm se apresentado como plataformas multifuncionais com sensores integrados que os tornam capazes de gerar informações de elevada precisão. Tais sensores, como microfones, câmeras, acelerômetro, magnetômetro, detector de

luminosidade, entre outros, são capazes de desempenhar diversas funções fundidas em um só dispositivo. Devido à multifuncionalidade dos sensores, os Smartphones estão além do simples objetivo de proporcionar a comunicação entre as pessoas (PINTO; CENTENO, 2012).

Medeiros (2015) cita que os sensores podem prover a percepção de contextos específicos, como a orientação da tela do dispositivo, a temperatura local, a luminosidade, som local, aceleração, dentre outros. A quantidade de sensores presentes em Smartphones e Tablets está constantemente aumentando, o que possibilita a percepção de novos tipos de contexto.

Os sensores, em termos físicos, são componentes de hardware conectados aos dispositivos de forma que fornecem dados do mundo real para as aplicações. De acordo com Möller (2006), sensores são dispositivos tecnológicos que detectam sinais ou condições físicas, em que a maioria são elétricos ou eletrônicos e são amplamente utilizados em diversos dispositivos de uso pessoal. O autor destaca ainda que uma das principais características dos sensores se encontra na pervasividade, o que implica afirmar que os mesmos podem ser aplicados nos mais diversos setores da cadeia produtiva.

De acordo com Shepherd et al. (2014), os sensores nos Smartphones seguem um quadro de coordenadas para detecção e extração de algumas de suas características físicas. Um quadro de coordenadas é um sistema no qual a orientação dos três eixos é definida em referência a um objeto. Consoante o autor, existem dois tipos de quadro de coordenadas quando são utilizados eventos de orientação e de movimento: o quadro de coordenadas geográficas e o quadro de coordenadas do dispositivo.

O quadro de coordenadas geográficas é o que se encontra fixo no centro da Terra, ou seja, os eixos são alinhados com base na força da gravidade e da orientação padrão do norte magnético. São utilizadas letras maiúsculas (“X”, “Y” e “Z”) para referenciar os eixos geográficos. O quadro de coordenadas do dispositivo diz respeito às coordenadas fixadas no centro do dispositivo. São usadas letras minúsculas (“x”, “y” e “z”) para descrever os eixos de coordenadas do dispositivo. Na Figura 2 é apresentada a diferença entre os tipos citados.

Os movimentos captados pelos sensores do Smartphone, com relação a orientação e movimentação do dispositivo, podem ser visualizados na Figura 3. O *deviceorientation* é um evento que devolve um objeto com três propriedades: *alfa*, *beta* e *gama* (Figura 3), que corresponde à rotação do dispositivo em graus em torno dos eixos z, x e y, respectivamente.

A propriedade *beta* tem um intervalo de -180° a $+180^\circ$, já o *gama*, entre -90° a 90° . Um dispositivo deitado deve reportar *beta* e *gama* com valores zero. A inclinação total da tela do dispositivo para a direita aumenta a *gama* positivamente a 90° , e o oposto, inclinação total a esquerda, diminui *gama* para -90° . Da mesma forma, inclinando

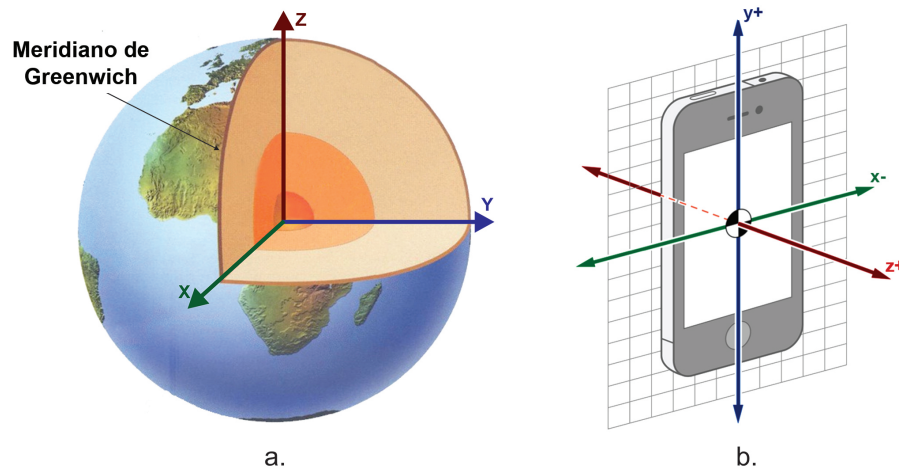


Figura 2 – Diferença entre os tipos de quadro de coordenadas
 a. Quadro de coordenadas geográficas b. Quadro de coordenadas do dispositivo
 Fonte: Autoria própria

o dispositivo para a frente, elevando uma das extremidades em direção ao usuário, aumenta-se o *beta* no sentido de 90° . E inclinando-o para baixo, diminui o *beta* no sentido de -90° . O ângulo *alfa* é de 0° quando parte superior do dispositivo é apontada diretamente para o polo norte do planeta, e aumenta à medida que o dispositivo é rodado para a esquerda. O *alfa* é relatado pela bússola, por exemplo 90° como oeste (SHEPHERD et al., 2014).

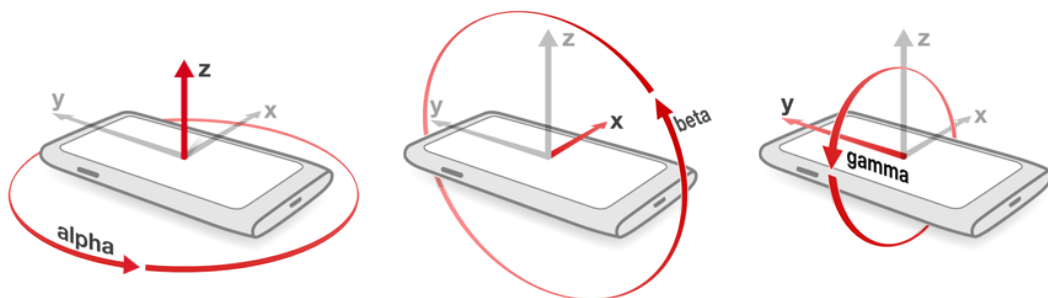


Figura 3 – Orientações *alfa*, *beta* e *gama*
 Fonte: Shepherd et al. (2014)

De acordo com (DUARTE, 2013), a plataforma Android é constituída por três categorias de sensores, são eles:

- Sensores de movimento: estes sensores medem as forças de aceleração e de rotação (linear e angular) ao longo de três eixos *x*, *y* e *z*. Nessa categoria estão incluídos o acelerômetro, giroscópio, sensor de gravidade e sensor de rotação vetorial (SO Android 2.3 ou superior).

- Sensores de posicionamento: dispositivos eletrônicos que possibilitam a obtenção da posição física do dispositivo móvel com alta precisão. Bússola, sensores de geoposicionamento *Global Positioning System* (GPS), proximidade e de campos magnéticos são exemplos.
- Sensores ambientais: existem vários parâmetros ambientais que os sensores deste tipo medem, tais como temperatura ambiente, pressão atmosférica, iluminação, umidade relativa, dentre outros. Esta categoria inclui barômetros, fotômetros e termômetros.

A maior parte dos Smartphones tem pelo menos dois sensores considerados base, que são os chamados sensores de movimento, compreendendo os sensores: acelerômetro, que detecta a variação de movimento ou aceleração nos eixos x, y, z; e o giroscópio, que mede e obtém a orientação do aparelho (DUARTE, 2013).

2.1.1.1.1 Acelerômetro

Um sensor de aceleração mede a aceleração aplicada ao dispositivo, incluindo a força da gravidade. O acelerômetro é um componente eletrônico que mede as forças de aceleração que são exercidas em um determinado objeto. Tais forças podem ser estáticas ou dinâmicas. A estática corresponde à força da aceleração gravitacional (constante), e a dinâmica é obtida por meio do movimento ou vibração provocada no acelerômetro (GOODRICH, 2015).

De acordo com Duarte (2013), existem vários tipos de acelerômetros e diversas características, desvantagens e vantagens para cada um. Entretanto, podem ser divididos em dois grandes grupos: os mecânicos e os eletromecânicos. No primeiro grupo, os acelerômetros mais comuns são os capacitivos, piezoelétricos (geram tensão elétrica por resposta a uma pressão mecânica) e os piezoresistivos (usa a variação de resistência de um cristal quando tensionado). Já no segundo grupo, de eletromecânicos, podem-se destacar os sistemas microeletromecânicos, do inglês *Micro-Electro-Mechanical Systems* (MEMS).

O princípio de funcionamento dos acelerômetros se faz a partir da movimentação do segmento, ou local onde o sensor foi fixado, que com sua inércia, em relação a uma base fixa no seu interior, é detectada e transformada em um sinal elétrico. Quando um objeto encontra-se em repouso, os sinais do acelerômetro indicam a sua aceleração estática, que é medida por meio da projeção da aceleração da gravidade sobre os eixos do acelerômetro para determinar o ângulo de inclinação do dado objeto (Kionix, 2015).

Um acelerômetro de um *Tablet* ou *Smartphone* é capaz de medir acelerações no intervalo $\pm 2G$ (G corresponde a aceleração da gravidade) em relação a um referencial inercial. O sensor mede as componentes da aceleração em três eixos perpendiculares x,

y, z em m/s^2 . O eixo x corresponde à aceleração frontal, o y à aceleração vertical e o z à aceleração lateral (VIEIRA, 2013).

2.1.1.1.2 Giroscópio

Inventado por *Léon Foucault* em 1852, o giroscópio é constituído por um rotor suspenso com um eixo em rotação, composto por vários anéis que giram em qualquer direção. Essa roda mantém uma posição fixa mesmo após ser exposta a direções não paralelas, ou seja, o acelerômetro possui a propriedade exclusiva de não modificar sua direção de origem, mantendo sua posição inicial de rotação independente do movimento do conjunto (CUARELLI, 2013).

Assim como os acelerômetros medem variáveis do movimento translacional, os giroscópios medem variáveis do movimento rotacional ou orientação dos acelerômetros. O giroscópio é um dispositivo que utiliza a gravidade da Terra para determinar orientações e mede a variação da rotação em rad/s em torno dos eixos x, y e z (GOODRICH, 2015).

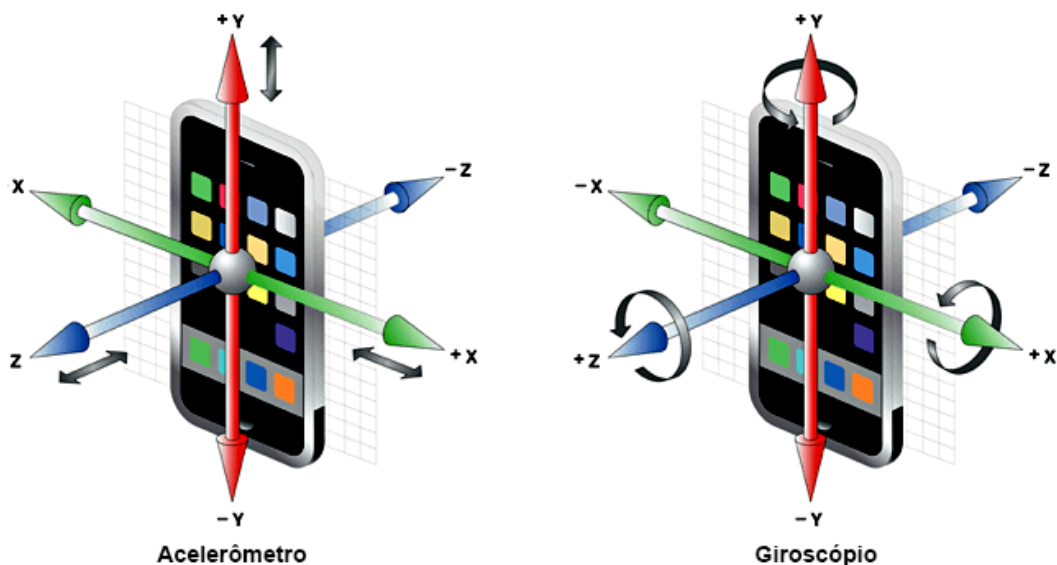


Figura 4 – Diferença entre o acelerômetro e o giroscópio

Fonte: Apple Developer (2015)

A principal diferença entre o acelerômetro e o giroscópio diz-se ao fato de um poder medir a rotação, enquanto o outro, a aceleração (Figura 4). Ao acelerar um dispositivo em uma determinada direção, o acelerômetro é incapaz de distinguir entre essa aceleração e a fornecida por meio da força gravitacional da Terra. O giroscópio mantém o seu nível de eficácia por ser capaz de medir a velocidade de rotação em torno de um determinado eixo.

As aplicações para cada sensor podem variar, apesar de suas finalidades semelhantes. Um giroscópio, por exemplo, é usado na navegação em veículos aéreos não tripulados, bússolas e barcos, para ajudar com a estabilidade na navegação, entre outros. Já os acelerômetros são igualmente difundidos e podem ser encontrados na engenharia, em máquinas, monitoramento de hardware, construção e monitoramento estrutural, navegação, transporte, produtos eletrônicos pessoais, e outros.

2.1.1.2 M-Health

Devido à grande quantidade de informações e às operações a serem realizadas nos processos de gestão hospitalar, o gerenciamento e o controle dos dados que auxiliavam no apoio à tomada de decisão tornaram-se bastante complexos. Era necessário um meio que viesse a suprir e solucionar as dificuldades encontradas, bem como facilitar o uso dos serviços disponíveis. Nesse âmbito, surge a telemedicina, que, conforme definida pela OMS, caracteriza-se como um conjunto de recursos, tecnologias e aplicações que possibilitam a realização de assistência médica a distância e disponibilização de serviços ligados ao cuidado com a saúde. Tais serviços são desempenhados por profissionais da área, os quais se encontram em estabelecimentos de saúde e, por meio das TICs, fornecem informações válidas para diagnósticos, prevenções e tratamentos de doenças.

A utilização da telemedicina visa facilitar e ampliar o atendimento por parte dos profissionais da saúde através do uso das tecnologias de comunicação, porém, conforme (BAPTISTA, 2010), a mesma teve seu início no século XX, quando o médico neerlandês Willem Einthoven, descobridor do mecanismo do eletrocardiograma, iniciou experiências de consultas remotas por meio da rede telefônica. Há também registros da utilização da telemedicina durante a Primeira Guerra Mundial, com o uso do rádio para permitir a comunicação entre médicos, em meados de 1916. A partir de 1965, a telemedicina foi alcançando outros horizontes com o emprego da tecnologia em seus recursos, inaugurando essa nova era com um sistema de videoconferência entre os centros médicos de Boston e Massachusetts e realização de triagens por meio deste. A partir da década de 90, as aplicações médicas a distância se multiplicaram e os projetos de telemedicina desenvolveram-se rapidamente.

A telemedicina é uma realidade presente em diversas áreas de assistência à saúde em vários países. Anualmente, calcula-se um investimento de aproximadamente 20 bilhões de dólares no mundo em tecnologias voltadas à mesma. A principal finalidade da telemedicina, segundo Gomes et al. (2011), é de fornecer suporte clínico remoto e superar barreiras geográficas a fim de conectar usuários que não se encontram no mesmo local físico. Para que tal objetivo se torne viável, a telemedicina tem contado com o apoio da tecnologia, uma vez que esta tem se tornado imprescindível na prática médica e no desenvolvimento da aplicação dos cuidados da saúde.

A utilização do potencial oferecido pelas atuais tecnologias de telecomunicação

na prestação de cuidados médicos tem trazido muitos benefícios para a população que necessita de assistência médica, tornando-se um elo entre aqueles que não têm condições de acessar centros médicos e os profissionais da área. O termo M-Health, do acrônimo Mobile Health, tem sido utilizado atualmente como uma das áreas provenientes da telemedicina. Segundo Blaya et al. (2010), M-Health é o conceito de apoio a práticas médicas e de saúde pública por meio de dispositivos móveis e sem fio, a fim de facilitar e melhorar os resultados de serviços prestados na área médica.

Um dos principais objetivos do M-Health é reduzir o custo dos cuidados médicos, maximizando a eficiência no sistema de saúde e promovendo a prevenção. Além disso, também possui o benefício do acompanhamento diário obrigatório por parte de alguns pacientes com determinadas doenças que exigem assistência frequente, o que traz alívio àqueles que necessitam de cadeira de rodas para se locomover ou pacientes acamados, impossibilitados de ir a centros médicos para a realização de diagnósticos e testes.

Com o advento de telefones com tecnologia avançada e inteligente (Smartphones) nos últimos anos, o M-Health tem se tornado mais abrangente devido à utilização de um dispositivo móvel para a execução de vários serviços de maneira potencializada. De certa forma, o M-Health transforma o tradicional modo de cuidado de saúde, permitindo, assim, que estes continuem sendo executados, porém, de forma otimizada e ágil a qualquer momento e em qualquer localidade. É possível, por meio do M-Health, promover cuidados preventivos, gerenciar o cuidado de doenças crônicas e demais enfermidades, e, principalmente, obter resultados para melhorar o bem-estar pessoal da população em geral (MEDEIROS, 2015).

Um número progressivo de países em desenvolvimento está adotando a utilização de tecnologias móveis como meios de praticidade para atender às necessidades de saúde daqueles que precisam de atenção médica. Vários dispositivos móveis são contemplados pelo M-Health, variando desde aparelhos simples de celular a modernos dispositivos habilitados a realizar tarefas específicas de Medicina. As áreas de aplicações do M-Health são bastante dinâmicas e possuem vasta quantidade de aplicações e recursos em desenvolvimento. São alguns exemplos:

- Diagnóstico e monitoramento remoto

Essa área tem como finalidade auxiliar pacientes na obtenção de diagnósticos prévios e monitoramento de condições de saúde por meio de dispositivos que estabelecem a conexão com bases médicas ou com profissionais da área, anulando, dessa forma, a necessidade de estar presencialmente em um centro médico. Além de garantir a aproximação do paciente com o médico e garantir o regime de medicação, o monitoramento remoto possibilita novos meios para o cuidado com pacientes em regime ambulatorial.

O funcionamento dessas aplicações geralmente se dá por meio da existência de bases de dados remotas que atuam como apoio a decisões e que possibilitam a conexão direta com o especialista da área médica através da transferência de pacotes de voz, imagens ou vídeos. A Figura 5 ilustra um sistema de monitoramento onde as informações vitais do paciente são obtidas pelo relógio do mesmo e salvas em um chip. Essas informações são transmitidas via internet sem fio ou móvel para o servidor e este faz a comunicação com o dispositivo móvel ou computador em que o profissional de saúde está conectado. Assim, o especialista tem informações em tempo real da saúde do paciente e pode enviar mensagens de alertas para o mesmo, vice-versa. Outros exemplos: Cell-Life Project, da África do Sul; Mobile Care Support and Treatment Manager, desenvolvido pela Índia; Phoned Pill Reminders for TB Treatment, criado Tailândia; entre outros (BARROS, 2015).

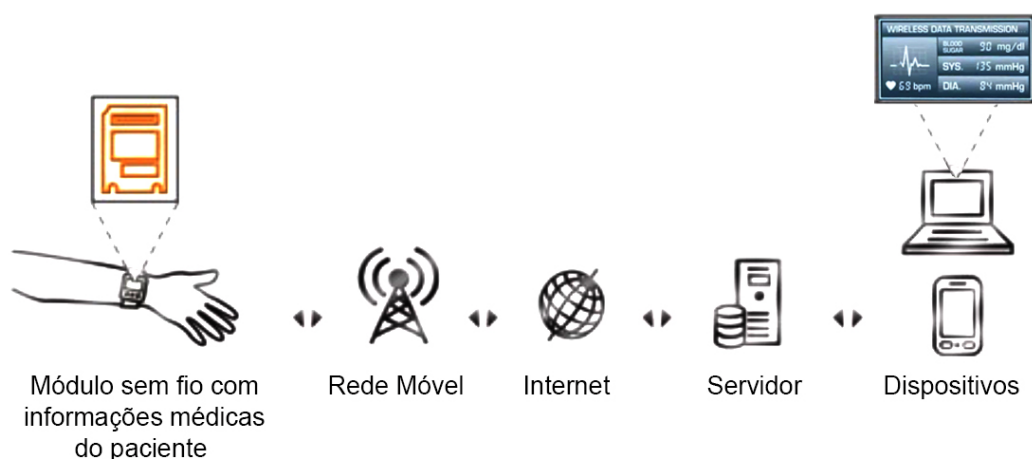


Figura 5 – Arquitetura do sistema

Fonte: Adaptado de (GEMALTO, 2012)

- Prevenção e conscientização

É de grande importância a prevenção de doenças para evitar a contração de outras enfermidades ou agravamentos contagiosos. Além disso, a conscientização e o conhecimento sobre meios de prevenção abrem possibilidades para procura de tratamento e cura prévios. Doenças como catapora, caxumba, sarampo, tuberculose, rubéola, gripe H1N1 e hepatite são comuns principalmente em crianças e adolescentes entre 12 meses e 12 anos de idade, porém, podem se estender às demais faixas etárias.

Conforme Portal da Saúde (2014), vacinas para o combate de enfermidades contagiosas resultam em 97% de sucesso, no entanto, as informações de existência de vacinação, muitas vezes, são inacessíveis para a maioria da população. Para aplicações com propósito informativo e preventivo, geralmente, são utilizados serviços de envio de *Short Message Service* (SMS) para alertas sobre a saúde no geral. O baixo custo e a

onipresença das mensagens de texto têm o potencial de mudar a educação da saúde por ser acessível e proporcionar à população chances de tomar decisões bem informadas.

Na Figura 6 é apresentado o “Virtual Check up”, que é uma ferramenta desenvolvida como aplicativo de Smartphones, que tem como objetivo fornecer informações relacionadas à prevenção, ao aconselhamento e ao monitoramento médico. De acordo com dados fornecidos pelo usuário, uma lista de recomendações em prevenção é gerada e também uma listagem dos exames que devem ser realizados pelo usuário, além de vacinas.

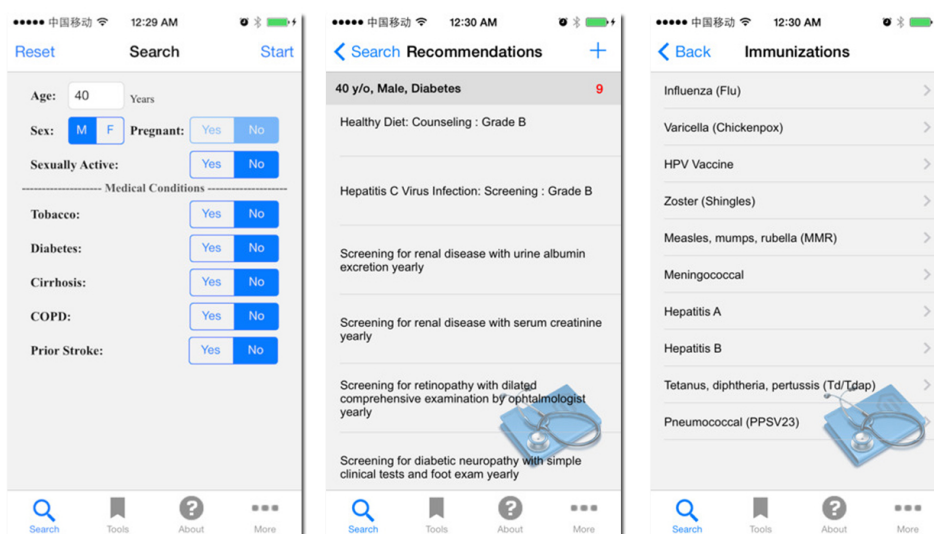


Figura 6 – Aplicativo Virtual Check up

Fonte: Adaptado de (iTunes, 2014)

Há milhares de aplicações desenvolvidas para Smartphones, como o mDiabetes, para envio de SMS com alertas de hidratação para equilibrar o consumo de açúcar para pacientes muçulmanos durante o Ramadã; TERA, que fornece mensagens sobre higiene e cuidados especiais para a doença Ebola; Projeto Masiluleke, que faz uso das mensagens de texto para auxiliar na prevenção do HIV/AIDS na África do Sul; dentre outros.

- Sistemas para consultas e reservas

Em muitas partes do mundo, a informação sobre reservas de medicamentos ou consultas ainda é realizada por meio de formulários manuais, o que causa grande risco de atraso no envio de informações para centros hospitalares ou, em pior caso, perda de parte das informações. Pacientes que moram em áreas remotas dependem da disponibilidade de medicamentos dos centros de saúde para tratamento de doenças, porém, muitas vezes, o paciente se desloca até a clínica e não há remédios para atendê-lo.

Sistemas para reserva de medicamentos são essenciais para reduzir as dificuldades daqueles que moram distantes de centros médicos. Tais sistemas são amplamente aplicáveis e propiciam redução de custos para seus usuários, além de facilidade ao solicitar uma consulta e da disponibilidade do medicamento preciso.

A exemplo prático tem-se a COOPI M-Health, projeto desenvolvido pela Cooperação Internacional (COOPI) e apoiado o Fundo das Nações Unidas para a Infância (UNICEF) que tem por objetivo o monitoramento móvel da gestão de reservas de medicamentos a nível de clínicas de aldeia e de surtos de doenças na Somália. Os pequenos centros médicos são administrados por funcionários governamentais, que realizam visitas às pessoas necessitadas para colher informações e enviar as bases médicas e assim realizar a reserva para fornecimento dos medicamentos necessários ou consultas (FAVA, 2014).

- Tratamento e bem-estar

A onipresença da internet em telefones móveis apresenta uma oportunidade para avaliar e tratar comportamentos de saúde em grandes segmentos da população. Com a rápida expansão de redes celulares e Smartphones, agora é possível transmitir digitalmente dados de paciente de áreas remotas para especialistas em áreas urbanas, receber feedback em tempo real e salvar essa consulta em um Banco de Dados (BD). Tratamentos a distância têm sido soluções de grande valia para área médica.

Existem no mercado diversos projetos M-Health para promoção de tratamento e bem-estar do paciente. Uma delas é o MedMinder (Figura 7), um dispensador de pílula inteligente com conectividade sem fio.



Figura 7 – Equipamento MedMinder

Fonte: (MEDMINDER, 2014)

O dispositivo avançado M-Health controla a ingestão de medicamentos, além de enviar alertas médicos, lembretes via SMS, e-mail ou telefonema; pedidos de recargas

dos medicamentos e também dispara um alarme sonoro para alertar o paciente quando a medicação deve ser tomada. Caso as doses regulares não sejam ingeridas no tempo determinado, os cuidadores recebem alertas em tempo real para que possam tomar os cuidados médicos devidos (MEDMINDER, 2014).

A tecnologia M-Health agregada a demais dispositivos oferece grande flexibilidade e custo-benefício para melhoria da saúde. Outros projetos voltados ao tratamento e bem-estar dos pacientes podem ser destacados, como o *Treatment Advice from Mobile Alerts* (TAMA), que fornece dicas e lembretes para pacientes com HIV/AIDS, lembra-os de tomar a medicação na hora certa e de participar de consultas médicas; o *Timed e Targeted Counseling*, que utiliza a tecnologia móvel para melhorar as interações entre os profissionais de saúde e os pacientes; dentre outros (Grameen Foundation, 2014).

- Urgência e emergência

Urgência é uma situação que exige assistência rápida no menor período de tempo, a fim de evitar complicações ou sofrimento. Já emergência é todo caso em que existe a ameaça iminente à vida, havendo necessidade de tratamento médico imediato para evitar casos extremos como o óbito. O M-Health também contempla essa área tão delicada que expõe todos a situações de risco.

A fim de amenizar e reduzir o número de vítimas fatais e também agilizar o atendimento de socorro, a aplicação móvel *EmergentHelper V2* (Figura 8), desenvolvida para Smartphones com sistema operacional Android, propõe uma solução para acelerar o resgate de vidas em situações perigosas. O usuário poderá previamente salvar informações pessoais e agravantes na saúde (se assim possuir) para informação médica sobre o paciente. Além disso, também poderá salvar contatos de emergência para notificação familiar. Outras funcionalidades adicionais como luzes de resgate, posicionamento via GPS e *widget* com botão de emergência para acesso mais rápido, também são presentes no aplicativo.

2.1.2 Computação Pervasiva

As aplicações têm se tornado gradativamente presentes nos ambientes dinâmicos, seja devido a mudanças nas preferências do usuário ou pela variação da disponibilidade de recursos. Tais aplicações são amplamente distribuídas, móveis e hospedadas em ambientes inerentemente abertos e dinâmicos. Nesse contexto, de acordo com Möller (2006), a pervasividade atua na elaboração de um ambiente físico, onde o foco é o ser humano ou a tarefa a ser realizada, possibilitando aos usuários acesso imediato a serviços, de forma transparente.

O termo pervasivo é de origem inglesa, do termo *pervasive*, cuja definição é estar difundido inteiramente por toda parte, ou seja, aquilo que existe em cada parte de uma coisa e se espalha para todas as partes de um todo. Entretanto, tal termo é inexistente

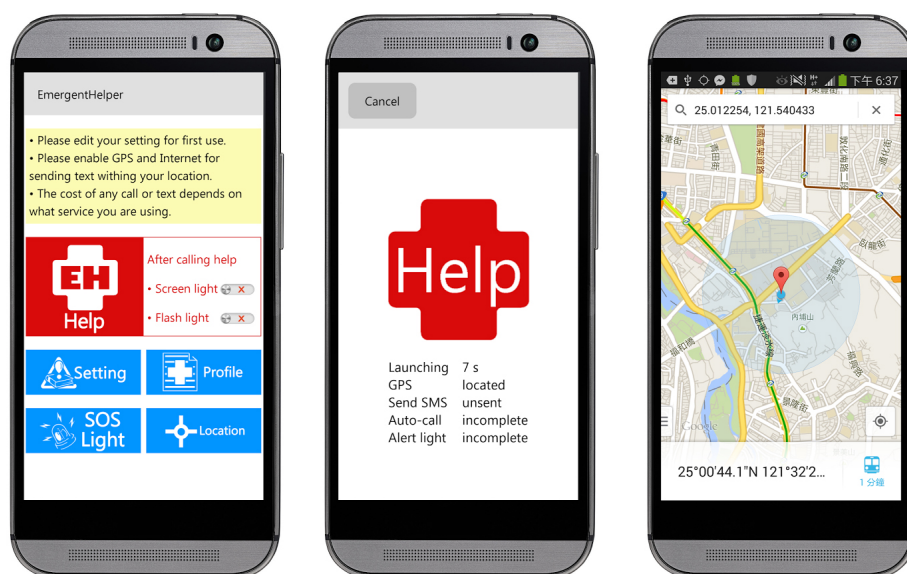


Figura 8 – Telas da aplicação EmergentHelper V2

Fonte: (ReBall Software, 2014)

no vocabulário da Língua Portuguesa (SOUZA, 2009). A pervasividade descrita por Saccol e Reinhard (2007) contemplam o acesso móvel a dados, bem como os recursos necessários para dar suporte a determinado conjunto de usuários móveis com enfoque em espaços inteligentes ou ativos, ambos com ciência de contexto.

A computação pervasiva é um paradigma computacional baseado na ideia de prover acesso às aplicações por meio de interações naturais com os usuários, de modo que seu uso possa ser o mais transparente possível. Vale salientar também que a computação pervasiva obtém informações sobre o ambiente ao seu redor para adaptar e configurar a aplicação para que aja de forma eficiente ou personalizada em um dado ambiente (OBAIDAT et al., 2011).

A ideia da computação pervasiva é disponibilizar uma variedade de dispositivos inteligentes nos ambientes. Estes dispositivos se estruturam para cada um fornecer acesso universal das informações aos usuários e dar suporte para realizarem suas tarefas (SACCOL; REINHARD, 2007). A busca por maior integração entre a computação e o ambiente físico envolvido é uma das características da computação pervasiva. Dentre as demais, Hansmann et al. (2013) destacam:

- **Descentralização:** ocorre a distribuição de tarefas e serviços entre diversos dispositivos heterogêneos, que colaboram entre si de forma dinâmica em redes que estão em constante mudança.
- **Diversificação:** com o uso de dispositivos móveis personalizados e outros dispositivos embarcados de uso específico, um grande desafio da computação pervasiva

é permitir a colaboração entre esses dispositivos para que um determinado serviço funcione adequadamente em todos eles.

- **Conectividade:** diversos dispositivos estão conectados uns aos outros, utilizando as mais diversas tecnologias. Por exemplo, um Smartphone pode adquirir dados por meio de uma conexão de dados de Terceira Geração, transmitir dados através de Bluetooth para um notebook que, por sua vez, pode transmitir dados através de redes WiFi ou Ethernet. A adoção de padrões abertos de comunicação facilita a implantação da conectividade.
- **Simplicidade:** os dispositivos pervasivos são ferramentas especializadas e, portanto, eles devem desempenhar muito bem as tarefas para as quais foram escolhidos. Com isso, a simplicidade de uso é uma característica fundamental da computação pervasiva.

Cabe destacar que, em geral, os termos computação pervasiva e computação móvel referem-se de certa forma ao conceito de computação ubíqua. Todavia, a característica embutido refere-se exclusivamente a computação pervasiva. A Figura 9 demonstra a relação entre os termos citados.

Conforme exposto na gravura, pode-se perceber que a computação ubíqua beneficia-se dos avanços tecnológicos de ambos os ramos de pesquisa relatados, sendo uma junção de características desses conceitos. Por exemplo, a computação ubíqua se favorece do avanço da tecnologia móvel com os dispositivos mais portáteis de fácil mobilidade e, conseqüentemente, de maior usabilidade. A computação pervasiva, por sua vez, beneficia-se da habilidade que os dispositivos computacionais têm de operar de forma inteligente e estar cada vez mais presentes, assim por diante.

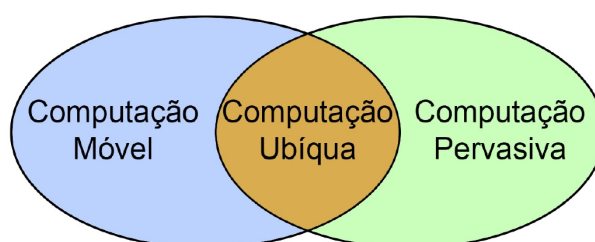


Figura 9 – Computação móvel, ubíqua e pervasiva relacionada

Fonte: Magalhães (2015)

2.1.3 Sensibilidade ao Contexto

Para proporcionar cada vez mais pervasividade às aplicações, a sensibilidade ao contexto tem sido utilizada, pois permite utilizar informações relevantes sobre entidades para facilitar a interação entre usuários e aplicações. Tais informações podem ser unidas

para associações. O agrupamento dessas informações associadas são utilizadas nas diversas situações do cotidiano das pessoas, fazendo, assim, uso do conhecimento do contexto em que se encontram para delimitar e direcionar ações e comportamentos a serem executados (VENECIAN, 2010).

A sensibilidade ao contexto, do inglês *Context-Aware*, descreve um paradigma no qual o contexto de um usuário é relevante para definição de seu perfil atual. Em ambientes em que há sensibilidade ao contexto, as informações fornecidas pelo usuário são obtidas dinamicamente a partir da interação com os dispositivos computacionais, levando em consideração o ambiente no qual estão inseridos (SILVA et al., 2013).

Raz et al. (2006) definem a Computação Sensível ao Contexto como um paradigma computacional que utiliza informações contextuais obtidas por meios diversos, de modo a agregar valor aos seus serviços, possibilitando a adaptação, a expansão, a restrição ou a alteração de suas funcionalidades conforme as mudanças percebidas nos contextos observados. Todos os tipos de atividades, incluindo a comunicação entre humanos, são rodeadas e influenciadas por um ou mais contextos. Do mesmo modo que um gesto ou palavra tem diferentes significados dependendo da situação na qual é utilizado, um usuário de um sistema também está rodeado pelo seu contexto enquanto interage com um serviço ou tecnologia.

Desde 1992, com o trabalho apresentado por Want et al. (1992), o termo sensibilidade ao contexto vem sendo utilizado. É apresentada pelo autor, neste trabalho, a aplicação nomeada como *Active Badge Location System*, onde era possível determinar a localização atual do usuário utilizando-a para encaminhar as ligações telefônicas para os telefones mais próximos, baseado na tecnologia infravermelho. Demais trabalhos foram desenvolvidos ao decorrer dos anos 90 e novos conceitos sobre sensibilidade ao contexto foram formalizados, como pode ser visualizado abaixo:

- Schilit e Theimer (1994) introduziram o termo Sensibilidade ao Contexto em trabalhos acadêmicos sobre computação móvel e distribuída. A definição dada pelos autores engloba a localização e identidade dos usuários, assim como o estado dos objetos em seu ambiente;
- Ward et al. (1997) definiram o contexto primariamente como a localização de um objeto em um dado ambiente;
- Pascoe et al. (1999) foram os pioneiros a propor que um contexto se referisse a algo além de uma localização, fazendo extensão do termo para adicionar atributos do ambiente;
- Schmidt et al. (1999) expandiram mais ainda a ideia de contexto como sendo algo além de localização, englobando informações sobre o clima, infraestrutura, dispositivos, usuários e atividades;

- Chen et al. (2000) introduziram o conceito de contexto de tempo, definindo-o como um conjunto de estados do ambiente e configurações que determinam o comportamento da aplicação ou na qual um evento da aplicação ocorre em duas categorias: contexto ativo (influencia o comportamento da aplicação) e contexto passivo (não é fundamental, mas importante à aplicação);
- Yamin et al. (2003) trabalharam o contexto como toda informação relevante para aplicação que pode ser obtida da infraestrutura computacional, em que a alteração em quaisquer de seus estados dispara uma condição de adaptação àquela aplicação;
- Dey (2001) definiu contexto como qualquer informação que possa ser usada para caracterizar a situação de uma entidade. Uma entidade pode ser uma pessoa, objeto ou lugar que é considerado relevante para a interação entre um usuário e uma aplicação, podendo incluir os próprios;
- Chen et al. (2005) formalizaram o conceito histórico de contexto e a necessidade de armazenamento de informações contextuais como fonte de tomada de decisões e construção de aplicações sensíveis ao contexto;
- Adomavicius e Tuzhilin (2011) apresentaram um novo conceito de classificação de contexto em sistemas de recomendação, utilizando fatores contextuais como atributos de um ambiente. Essa classificação baseia-se em dois aspectos: o que são fatores contextuais em um ambiente e como estes mudam com o tempo.

De acordo com Dey (2010), um contexto, na área de sistemas computacionais, pode ser definido como qualquer informação que pode ser utilizada para fazer caracterização de uma situação de determinada entidade considerada relevante para a interação entre o usuário e uma aplicação. Silva et al. (2013) afirmam que um contexto é definido de acordo com informações referentes a propriedades que se aliam para definir e caracterizar uma entidade e seu papel de uma forma entendível pelo nó final.

Como referência clássica na área, Schilit (1995) e Schilit e Theimer (1994) abordam ainda contextos, dividindo-os em três categorias básicas. São elas o contexto computacional, no qual diz respeito à conectividade da rede, aos custos de comunicação, à largura de banda e aos recursos disponíveis; o contexto do usuário, dispondo do perfil do usuário, localização, pessoas por proximidade, estado astral, entre outros; e o contexto físico, compondo-se de luminosidade, níveis de ruído, condições ambientais, temperatura, clima e outros.

Chen et al. (2005) defendem ainda a inclusão do Tempo com seus atributos, como hora do dia, dia da semana, mês, ano, etc., como uma quarta categoria. Raz et al. (2006) definem algumas características das informações de contextos:

- Meio: diz respeito à característica material do contexto, podendo ser especificado como físico, quando é possível mensurá-lo, ou intangível, quando trata de informações imensuráveis ou pessoais;
- faz menção a que se referem as informações de um determinado contexto. Por exemplo, dados sobre o perfil do usuário ou sua localização são características do contexto do usuário, já dados do endereço de *Internet Protocol* (IP) ou quantidade de memória disponível dizem respeito a características de um contexto de um dispositivo;
- Persistência: um contexto pode ser permanente (a informação obtida não precisa ser atualizada, pois não muda no decorrer do tempo) ou temporário (a informação ou parte dela precisa de atualização em certos períodos de tempo);
- Evolução: existem contextos que sofrem mudanças com mais frequência do que os demais. Aqueles que mudam rapidamente chamam-se dinâmicos, como, por exemplo, as informações obtidas pelo GPS. Já os contextos que não mudam com tanta frequência são chamados de estáticos;
- Situação temporal: define a temporalidade do contexto, que pode conter informações sobre o passado, presente ou futuro;
- Relevância: o contexto pode ser opcional ou necessário para a aplicação ou o serviço. Ele é obrigatório se o serviço desempenhado não puder ser executado corretamente sem sua existência, e é opcional se o contexto for útil para funções adicionais ou para melhorar o serviço.

Uma das definições mais utilizadas na literatura está em Dey e Abowd (2000), no qual definem as dimensões semânticas do contexto, apresentadas na Tabela 1. Zimmermann et al. (2007) expandiram a definição de Dey (2001) ao sugerir que o contexto também pode conter um aspecto social, que representa o relacionamento entre entidades diferentes; e um aspecto temporal, no qual a classificação do contexto sofre alterações ao longo do tempo.

A sensibilidade ao contexto para as aplicações apresenta inúmeros desafios, dentre eles: a caracterização dos elementos de contexto para uso na aplicação; a aquisição do contexto a partir de fontes heterogêneas, tais como sensores físicos, base de dados, agentes e aplicações; a representação de um modelo semântico formal de contexto; o processamento e a interpretação das informações de contexto adquiridas; a disseminação do contexto às entidades interessadas de forma distribuída e no momento oportuno; e o tratamento da qualidade da informação contextual (VENECIAN, 2010).

Para que seja possível utilizar contextos em aplicações, é preciso haver mecanismos que permitam perceber o contexto atual e o descrevam para a aplicação. A obtenção

Tabela 1 – Dimensões semânticas do Contexto

Dimensão	Descrição
Quem	Quem realiza uma determinada atividade, quem pode alterar o contexto ou quem pode ser notificado caso o contexto seja alterado.
Onde	Onde o contexto está. Esta é uma das dimensões mais usadas devido ao grande interesse de sistemas baseados em localização.
Quando	Informação temporal, de duração ou de intervalo de tempo para determinar quanto tempo uma entidade está dentro de um contexto.
Por onde	Trajetos que uma entidade percorreu durante determinado tempo, permitindo rastrear os caminhos que uma entidade tomou durante um período.
O quê	O que o usuário está fazendo no momento. Geralmente necessita de sensores para determinar qual é a atividade.
Porquê	Por que razão o usuário está realizando determinada atividade.

da localização do usuário e sua movimentação ao longo do espaço são informações contextuais de grande relevância para a sensibilidade ao contexto. A captação dos registros dos contextos de localização vindos dos dispositivos faz com que seja perceptível a proximidade entre as pessoas e os aparelhos tecnológicos.

Efetivamente, a maioria das informações disponíveis em um dado momento da interação podem ser identificadas como informações contextuais, dentre as quais, podem-se destacar algumas: as diversas tarefas exigidas pelos usuários; a grande quantidade de dispositivos que se combinam para a criação de sistemas móveis com infraestrutura de serviços associada; a disponibilidade de recursos, como por exemplo, a condição da bateria, o tamanho de tela do dispositivo, dentre outros; a situação física, a exemplo o nível de ruído, a temperatura, nível de luminosidade, entre outros; a informação espacial, como sua localização, velocidade, orientação, etc.; e sua informação temporal, como por exemplo a hora do dia, data e outros.. (MONTEIRO; GOMES, 2015).

Em Smartphones, Tablets e dispositivos pessoais móveis, tornou-se comum a agregação de tecnologias, como Bluetooth, *Near Field Communication* (NFC) e GPS. Esta última possibilita localizar o dispositivo e obter dados de sua localização em diversos formatos. Existem inúmeras tecnologias utilizadas para este fim, usadas de acordo com a conveniência e disponibilidade de tecnologia, como uso do NFC em etiquetas para informar a um dispositivo que há um objeto de seu interesse em sua proximidade.

A informação contextual de tempo, geralmente, é adquirida pelo relógio interno dos próprios dispositivos. Em dispositivos que dispõem de conexão móvel com a internet, como redes 3G, a utilização da largura de banda para obter dados para contextos específicos pode ser um fator importante para algumas aplicações. Sensores também são meios que podem prover a percepção de contextos específicos, como a orientação e o posicionamento do dispositivo, a temperatura local, a luminosidade, a aceleração, etc. (MAGALHÃES, 2015).

2.2 TECNOLOGIAS ASSISTIVAS

A evolução tecnológica caminha na direção de tornar a vida mais fácil. Ferramentas e recursos são pensados e desenvolvidos especialmente para beneficiar e facilitar as atividades desenvolvidas na rotina cotidiana, melhorando o desempenho em realizar funções pretendidas (BERSCH, 2013). É evidente que a tecnologia é uma grande aliada das pessoas com alguma deficiência, pois são elas que possibilitam substituir ou compensar sentidos que foram perdidos ao longo da vida ou congênitos.

A acessibilidade e a inclusão são pontos fundamentais para a inserção dessas pessoas na sociedade. Para permitir autonomia e tornar mais fácil a vida das pessoas que têm necessidades especiais, foram desenvolvidas as chamadas TAs, que buscam soluções voltadas à inclusão social com intervenções do contexto de vida do paciente, assegurando oportunidades iguais para cada indivíduo e atendendo particularmente suas diferenças.

Oficialmente, a tecnologia assistiva foi criada como um elemento da legislação norte-americana em 1988, intitulada como *Public Law 100-4007*, na qual foi composta também a *American with Disabilities Act (ADA)* 1. Com essas leis, os cidadãos norte-americanos com deficiência passaram a dispor legalmente dos capitais públicos para compras dos recursos de que os mesmos necessitam e que favorecem uma vida mais incluída no contexto social; além de regularizar seus demais direitos como cidadãos portadores de deficiência (ROBITAILLE, 2010).

Tais recursos envolvem, em sua descrição, produtos, sistemas fabricados genericamente ou sob medida, equipamentos, aparelhos assistivos, itens ou auxiliares utilizados nesse contexto para dispor, facilitar ou melhorar as capacidades funcionais das pessoas com deficiência (BERSCH, 2013).

King (1999) afirma que as tecnologias assistivas concentram-se nas necessidades de pessoas especiais de todas as idades, em que estes podem possuir várias deficiências, limitações ou desafios que limitam a sua participação na vida diária, na qual a assistência especial, seja ela motora, cognitiva ou linguística, torna-se necessária. Conforme cita Manzini e J. (2005), a TA permite compensar limitações motoras, sensoriais ou mentais, possibilitando uma plena inclusão social. O termo utilizado na legislação brasileira abrange produtos, instrumentos e equipamentos, ou tecnologias adaptadas ou especialmente projetadas para melhorar a funcionalidade da pessoa com deficiência, favorecendo a autonomia pessoal total.

As tecnologias assistivas podem ser aplicadas em uma vasta categoria de necessidades, abrangendo aspectos sociais, políticos e educacionais, com objetivo de amenizar dificuldades, reduzindo as barreiras no cotidiano. Na maioria dos casos, as pessoas utilizam combinações de várias aplicações de TA em cada uma dessas áreas que a compõem.

Segundo Greve (1999), a tecnologia pode ser considerada assistiva a partir do

momento em que é utilizada para auxiliar no desenvolvimento do desempenho funcional de atividades de pessoas portadoras de deficiência, reduzindo as incapacidades de realização de atividades no cotidiano. O termo Tecnologia Assistiva é utilizado para identificar todo o arsenal de recursos e serviços que contribuem para proporcionar ou ampliar habilidades funcionais de pessoas com deficiência e, conseqüentemente, promover vida independente e inclusão, como cita Manzini e J. (2005) e Bersch (2013). Em outras palavras, a tecnologia assistiva pode ser definida como processos, métodos, ou invenções que dão suporte a pessoas com deficiência (BRYANT et al., 2003). Na Figura 10 são exemplificadas algumas aplicações práticas para modalidades diferentes de deficiência, dentre elas, visual, física, auditiva e cognitiva.



Figura 10 – Exemplos de tecnologias assistivas para diferentes deficiências.

Fonte: Adaptado de Bersch (2013).

Nos últimos 30 anos, a tecnologia assistiva vem obtendo um grau de desenvolvimento bastante elevado, porém, vale salientar que, desde os primórdios, tal alternativa de acessibilidade vem sendo aplicada. A acessibilidade é um tema relativamente novo, se comparado à existência da deficiência em si. As invenções mais antigas descobertas ainda são aplicadas e melhoradas com o decorrer das décadas até os dias de hoje.

Em Cook e Polgar (2008), é mencionada a atuação da acessibilidade ainda na Idade da Pedra, onde um dos primeiros dispositivos de tecnologia assistiva seria concebido, descoberto e colocado em uso: uma vara, a qual seria utilizada para auxiliar e sustentar, como uma bengala, um dos integrantes do bando que teria quebrado a perna durante uma expedição de caça. Outro destaque foi a utilização de um chifre vazio de um animal, atuando como um aparelho auditivo, no qual compensaria a audição desgastada daqueles que teriam idades mais avançadas, fazendo com que os mesmos ouvissem melhor.

As aplicações descobertas há décadas passaram por muitas alterações. Apesar de algumas manterem a mesma estrutura, foram alterados os materiais utilizados para

fabricá-las. Contudo, algumas aplicações só puderam ser criadas após o avanço da tecnologia. Conforme citado por Cook e Polgar (2008), durante a Guerra Civil dos Estados Unidos, começou-se a desenvolver próteses para membros inferiores, visto que muitos soldados, após a Guerra, acabavam se tornando inválidos por perderem a locomoção.

Pode-se citar também a invenção do braile em 1824, por Louis Braille; e a invenção de Alexander Graham Bell durante década 1870, que, na tentativa desenvolver um dispositivo para fazer com que sua esposa surda pudesse ouvir, acabou inventando o telefone (ROBITAILLE, 2010). Tais inventos foram inovadores para época e foram essenciais para o promissor desenvolvimento das tecnologias assistivas do atual século. A evolução da eletrônica é a razão para a maior parte desses ganhos favoráveis e os computadores são os veículos através do qual os avanços têm sido concretizados (COOK; POLGAR, 2008).

2.3 AGENTES INTELIGENTES

A capacidade de compreender, perceber e aprender possibilita ao ser humano prever algumas situações, tomando como base o conhecimento já adquirido anteriormente. Assim, baseado nisso, a área da Inteligência Artificial (IA) busca compreender e construir máquinas inteligentes para que a inteligência similar à humana seja exibida por meio de mecanismos ou softwares. A IA é uma área dedicada a buscar métodos ou dispositivos computacionais com a capacidade racional semelhante a do ser humano de pensar, de realizar escolhas e de ser inteligente (NORVIG; RUSSELL, 2014).

Surgida em meados de 1950, a IA foi alvorecida após a Guerra Mundial, quando diversos avanços na Lógica Matemática e na Psicologia Cognitiva davam seus primeiros passos na representação do cérebro humano. O sucesso inicial se deu com o desenvolvimento do programa Solucionador de Problemas Gerais (SPG), construído por Newell e Simon. Projetado para copiar e reproduzir protocolos humanos, o SPG solucionava problemas dentro de uma classe limitada de quebra-cabeças e foi considerado o primeiro programa a incorporar a abordagem do pensamento humano (COPPIN, 2004).

As técnicas inteligentes são ferramentas computacionais que buscam otimizar seus resultados, aplicando características inerentes ao funcionamento dos mecanismos biológicos às resoluções inteligentes criadas. Com o progresso da IA, as técnicas inteligentes tornaram possível o desenvolvimento de sistemas capazes de obter informações a partir de bases de dados (REZENDE, 2005).

Conforme Louro et al. (2004), a crescente relevância e utilização do paradigma de agentes no domínio informático tem suscitado discussões sobre as características em comum e os aspectos que os distinguem dos demais programas. O paradigma trouxe às entidades computacionais comportamentos próprios quando inseridos em determinados

ambientes coordenados por suas próprias ações, com base no conhecimento sobre as ações possíveis e suas consequências sobre os outros agentes, e o ambiente que o rodeia.

Existem diversas definições para agentes, porém, pode-se entender que os agentes derivam do emprego das técnicas da IA para auxílio a usuários na execução de tarefas específicas. Na Literatura, podem-se encontrar diversos conceitos formalmente definidos pelos pioneiros na área:

- Brustoloni (1991) afirma que agentes são sistemas capazes de ações autônomas, com objetivos próprios no mundo real.
- Coen (1994) conceitua agentes de software como programas que dialogam entre si, coordenam e negociam transferências de informação.
- Smith et al. (1994) citam que um agente é uma entidade persistente de software dedicada a um determinado objetivo. Além disso, tem suas próprias ideias sobre como completar suas tarefas. Ter um objetivo específico distingue-os das aplicações com múltiplas funções.
- Hayes-Roth (1995) relata que os agentes inteligentes executam continuamente três funções: fazem a percepção das condições do ambiente, atuam de acordo com essas condições e raciocinam para resolver problemas e determinar suas ações.
- Russell et al. (1995) atestam que um agente é qualquer objeto que percepção o seu ambiente através de sensores e pode atuar sobre esse ambiente através de atuadores.
- Wooldridge e Jennings (1995) declaram que um agente é um sistema computadorizado em hardware ou software, que possui as propriedades de autonomia, capacidade social, reatividade e proatividade.
- Maes (1996) cita que agentes autônomos são sistemas computacionais que habitam em ambientes dinâmicos e complexos, e atuam autonomamente realizando um conjunto de objetivos ou tarefas para as quais foram designados.
- Franklin et al. (1996): definem que agentes são sistemas que estão situados em ambientes e são parte deles, realizando percepções e atuando sobre o mesmo no decorrer do tempo. Os agentes seguem suas próprias programações e desta forma podem afetar o que será percebido posteriormente.
- Ferber e Gutknecht (1998) consideram que um agente pode ser físico ou virtual. Além disso, tem capacidade de atuar em um ambiente, de se comunicar com outros agentes, de ter objetivos individuais de satisfação, possuir recursos próprios, ser capaz de perceber o ambiente, possuir habilidades e oferecer serviços.

- Rudowsky (2004) afirma que agentes inteligentes são entidades em software que executam um conjunto de operações em nome de um utilizador ou outro programa com algum grau de independência ou autonomia, e, ao realizá-lo, utilizam conhecimentos dos objetivos do utilizador.

Conceitos mais recentes sobre agentes também podem ser encontrados na literatura, como:

- Artero (2008) define agentes como programas que realizam diálogos para negociação e coordenação de transferências de informação. Além disso, executam várias operações para um usuário com base na representação do conhecimento contido nos objetivos deste utilizador.
- Wooldridge (2009) cita que um agente é um sistema computacional capaz de ações autônomas em um ambiente com o propósito de atingir os objetivos que são delegados.
- Norvig e Russell (2014) descrevem os agentes como entidades de software ou não, que faz caracterização dos mesmos em concordância com as ideias citadas anteriormente, abrangendo autonomia, objetivos e proatividade.
- Calado (2015) explana que o termo agente é usado para fazer representação de dois conceitos ortogonais, em que o primeiro exprime a capacidade que um agente tem de executar ações autonomamente, e o segundo a capacidade do mesmo raciocinar diante de um domínio específico.

Em maior parte das definições, é destacado como ponto principal o fato de um agente ser uma entidade computacional com capacidade de raciocínio que faz sua percepção através dos sensores e realiza ações por meio de atuadores. Conforme cita Henderson-Sellers (2005), os agentes devem possuir duas características básicas, sendo uma delas a capacidade de agir de forma autônoma, e a outra a capacidade de interação social. A característica de independência diz respeito ao agente possuir conhecimento e métodos para buscar a base do problema e tomar uma decisão para alcançar seu propósito. A outra característica citada faz alusão à ação humana de relacionar-se. Os agentes utilizam protocolos de interação social inspirados nos humanos para fazer referência às funcionalidades, como a de cooperação, competição, coordenação e negociação (ZIDAN; EL-SAADANY, 2012).

Além de suas características básicas, os agentes também devem possuir a propriedade de reatividade e proatividade. Agentes inteligentes não atuam necessariamente sobre ambientes somente em resposta a estímulos. Os mesmos podem apresentar comportamentos orientados a objetivos, bem como ter iniciativa em certas ocasiões, o

que caracteriza a proatividade. A reatividade diz respeito à sensibilidade do agente ao ambiente inserido e sua reação às mudanças ocorridas nele (SILVA et al., 2013).

Conforme (SARAIVA, 2012), basicamente, o elemento agente é composto por duas partes que possibilitam a integração dele ao ambiente inserido. As ferramentas que o agente deve possuir são os sensores, para obter as informações do ambiente, e os atuadores, responsáveis por executar as ações que são tomadas de acordo com o resultado da análise implementada no agente. A Figura 11 apresenta a representação do agente.

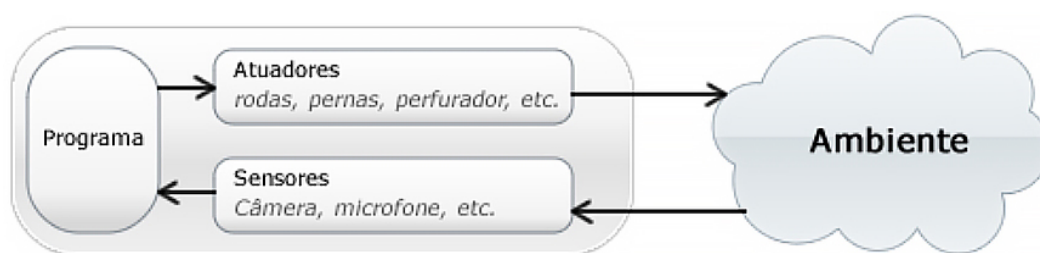


Figura 11 – Visão conceitual de um agente

Fonte: Adaptado de Artero (2008)

De acordo com Artero (2008) e Norvig e Russell (2014), os agentes podem se classificar nos seguintes tipos: agente tabela, reativo simples, reativo baseado em modelos, baseado em objetivos, baseado em utilidade e agente com aprendizagem. Agente tabela é o tipo de agente mais simples, onde o nível de complexidade é mais baixo, tendo em vista sua logística. Esse tipo de agente mantém uma tabela interna com todas as possíveis percepções e as ações que podem ser executadas. A desvantagem dessa abordagem se encontra na necessidade da adição de todas as percepções e ações manualmente, o que requer tempo.

Diferentemente do agente tabela, o agente reativo simples seleciona suas ações baseado na necessidade percebida atualmente e ignora o histórico de percepções, fazendo com que escolha a mais adequada para o momento. Esse tipo de agente utiliza um conjunto de regras de condição-ação em vez de tabelas, porém, por não possuírem memória, são incapazes de planejar ações futuras.

O agente reativo baseado em modelos controla o estado atual do mundo utilizando um modelo interno do ambiente e mantém o histórico de suas ações e percepções, combinando essas informações para gerar a descrição atualizada do estado atual. Com essas informações, o agente então escolhe a ação mais adequada, da mesma forma que o reativo simples.

Também nomeados como agentes cognitivos, os agentes baseados em objetivos são aqueles que avaliam constantemente suas ações e verificam se tais ações os aproximam de seus objetivos, ou seja, esses agentes ponderam suas ações de acordo com a descrição do estado atual e dos objetivos, que se deseja atingir. É possível observar que

a tomada de decisão desse tipo de agente é distinta das regras de condição-ação devido envolver considerações futuras.

O agente baseado em utilidade assemelha-se ao baseado em objetivo. O mesmo faz a escolha de suas ações também visando aproximar-se de um objetivo, contudo, com o diferencial de buscar maximizar o valor obtido pela função de utilidade. Tal função mensura o grau de felicidade do agente ao realizar determinada ação, ou seja, faz um mapeamento de um estado ou sequência de estados e guarda em uma variável com número real. Assim, quanto maior o valor obtido pela função, maior será a eficiência de execução daquela ação.

Correspondente à ideia de máquina inteligente caracterizada por Turing em 1950, os agentes com aprendizagem são aqueles capazes de atuar plenamente em um ambiente inicialmente desconhecido, adaptando-se às mudanças ocorridas no mesmo. Para isso, são utilizados algoritmos de aprendizagem que usam elementos de geração de problemas (sugestão de ações que podem trazer informação útil), aprendizagem (feedback que avalia a atuação do agente de acordo com o desempenho assistido) e desempenho (responsável por selecionar as ações do agente) (ALONSO et al., 2001).

3 TRABALHOS RELACIONADOS

Na Literatura, foram encontrados alguns trabalhos que realizam o monitoramento e o acompanhamento de pacientes através de tecnologias móveis. Dentre os estudos relacionados à prestação de socorro, foram encontrados botões de pânico e alguns aplicativos veiculares que realizam o procedimento. Dispositivos e aplicativos com sensoriamento remoto são os meios mais promissores e eficazes de fornecer cuidados de saúde móvel.

O M-Health pode desempenhar um papel essencial no serviço de emergência médica. Devido aos índices elevados de acidentes, incidentes e ocorrências, qualquer um está vulnerável e pode necessitar de informações sobre centros de saúde mais próximos para o socorro. O desenvolvimento de projetos em prol do auxílio à saúde, segurança e pronto atendimento se tornou cada vez mais comum.

Um sistema sem fio e móvel de emergência é apresentado por Hadjinicolaou et al. (2009) como uma solução que permite aos ortopedistas, em uma área remota, obterem opiniões de outros médicos em grandes centros de análise ou hospitais. Os especialistas têm a possibilidade de trocar imagens, vídeos e dados importantes com outros médicos, e, assim, realizar consultas remotas com rapidez e precisão (Figura 12). O sistema desenvolvido possui uma interface amigável e utiliza dos meios da telemedicina e do M-Health para propor uma solução confiável, segura e de baixo custo. O TeleOrthoPaedics, como assim é chamado, foi testado em dois principais hospitais do Chipre e obteve resultados de desempenho elevado.



Figura 12 – Interface da ferramenta *TeleOrthoPaedics*

Fonte: Hadjinicolaou et al. (2009)

O trabalho proposto por Sethia et al. (2014) é o desenvolvimento de um sistema de saúde móvel baseado na tecnologia NFC. É apresentada uma arquitetura de Registros Médicos Eletrônicos, do inglês *Electronic Medical Record* (EMR), para melhoria do sistema de saúde através da utilização do NFC e de um serviço em nuvem para gerenciar as informações em tempo real. O usuário tem a possibilidade de solicitar socorro a qualquer momento por meio da etiqueta NFC e também de armazenar suas informações pessoais referentes à saúde, como tipo sanguíneo, presença de enfermidades, alergias, entre outros.

No sistema de assistência médica desenvolvido por Das e Alam (2014), é demonstrada a utilização do projeto de mapeamento colaborativo de fornecimento de informações sobre os hospitais próximos, tomando por base a localização do usuário. O sistema compreende um banco de dados, onde as informações detalhadas sobre os centros emergenciais se encontram. O usuário pode fazer marcações no mapa dos centros hospitalares mais próximos de sua posição ou pode visualizar informações dos mesmos. A eficácia do sistema é determinada de acordo com o mapeamento do local.

Carvalho et al. (2011) desenvolveram uma proposta nomeada como Sistema Computacional Inteligente de Assistência Domiciliar à Saúde (SCIADS). O objetivo é interligar pacientes em seus ambientes domiciliares aos profissionais de saúde por meio do monitoramento remoto da saúde do paciente. Através da Central de Saúde Residencial (CSR), são recebidos os dados do paciente e do ambiente. Tais dados são obtidos por meio do *WristClinic*, medidor de pressão arterial e frequência cardíaca; e o *Small Programmable Object Technology* (SPOT), sensor para reconhecer a atividade do paciente. A Figura 13 apresenta uma visão geral da proposta.

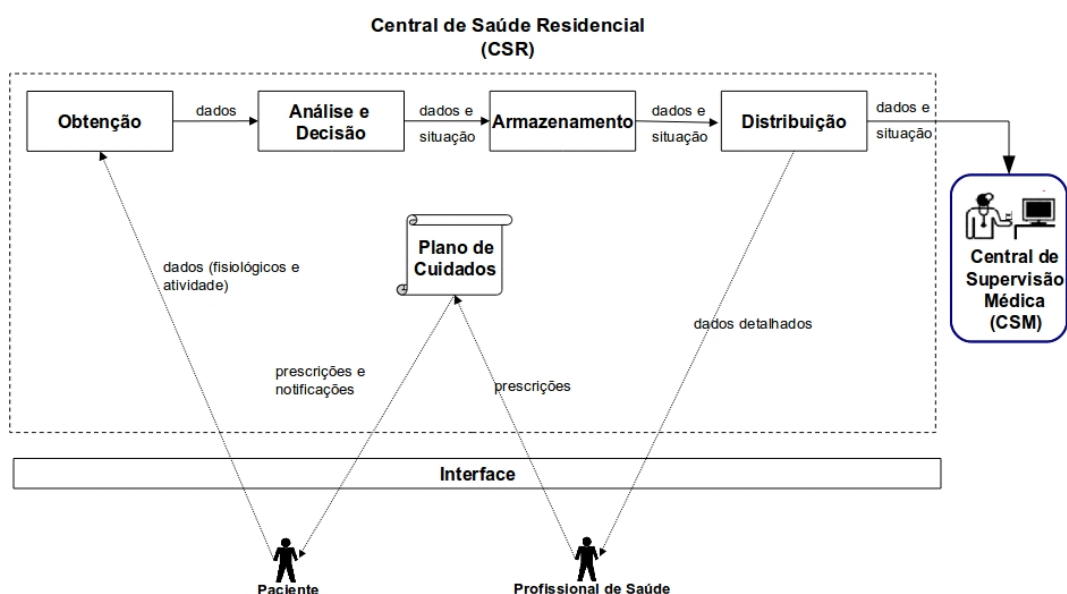


Figura 13 – Visão geral do SCIADS

Fonte: Carvalho et al. (2011)

O sistema é iniciado de acordo com a programação do plano de cuidados, composto por prescrições médicas. A identificação das situações de risco é obtida pelo módulo de análise e decisão, que identifica a situação de saúde do paciente (por meio dos sensores) e a categoriza entre normal, alerta ou emergência. Caso seja uma situação de emergência, um alarme é enviado imediatamente para a Central de Supervisão Médica (CSM) com os dados do paciente.

Em Besaleva e Weaver (2014), é apresentada uma ferramenta de coleta de solicitações emergenciais utilizando o modelo *crowdsourcing* (obtenção de conteúdo mediante a contribuição voluntária de um grupo de pessoas) para fornecer aos gestores da saúde dados sobre acidentes ocorridos.

A ferramenta *CrowdHelp* (Figura 14) também possibilita enviar textos, mídias e locais do corpo atingidos para fornecer informações mais precisas. Com os dados recebidos, é gerado um relatório que informa os registros das urgências constatadas e o local da eventualidade. A localização é exibida no mapa para que o gestor possa instruir a vítima no que for possível ser realizado a distância, como informar qual hospital mais próximo ou procedimentos rápidos ao acidentado.

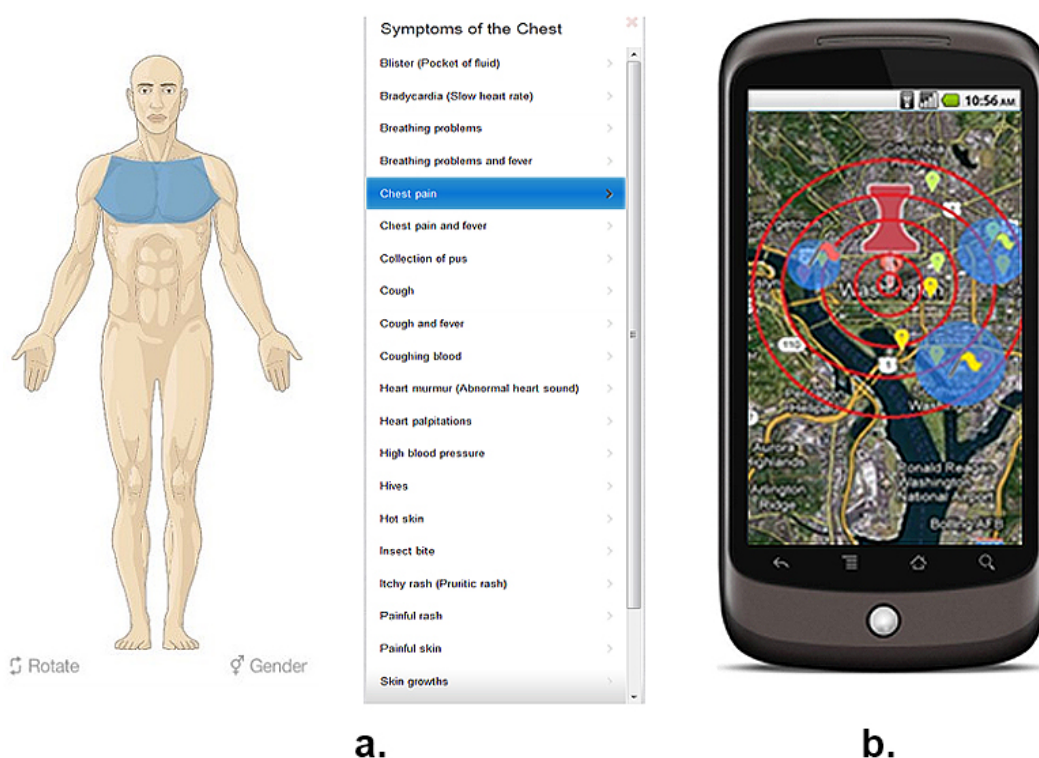


Figura 14 – Interface da ferramenta *CrowdHelp*

- a. Seleção de parte do corpo, indicando área lesada e os sintomas prováveis.
 b. Exemplo solicitação de socorro, onde a cor das marcações denota níveis de gravidade

Fonte: Besaleva e Weaver (2014)

Mohammedali et al. (2011) desenvolveram um aplicativo com sensibilidade ao

contexto para ajudar pessoas com autismo a lidar com os ataques de ansiedade e o Transtorno do Espectro Autista (TEA). A aplicação permite marcar o início de um ataque de pânico ou ansiedade. Ativa-se o nível de intervenção, que exibe estímulos, buscando passar instruções e acalmar o autista enquanto seus responsáveis são contatados. No modo ativo, são exibidos na tela do Smartphone três botões, que são correspondentes aos dois níveis de pânico e um terceiro indicando que está tudo bem. Ao tocar em um dos botões, é criado um registro da ação e, a partir daí, iniciado o processo de resposta configurado de acordo com a sensibilidade ao contexto. As respostas podem ser criadas pelos responsáveis do autista, assim como as ações que devem ser tomadas (Figura 15).

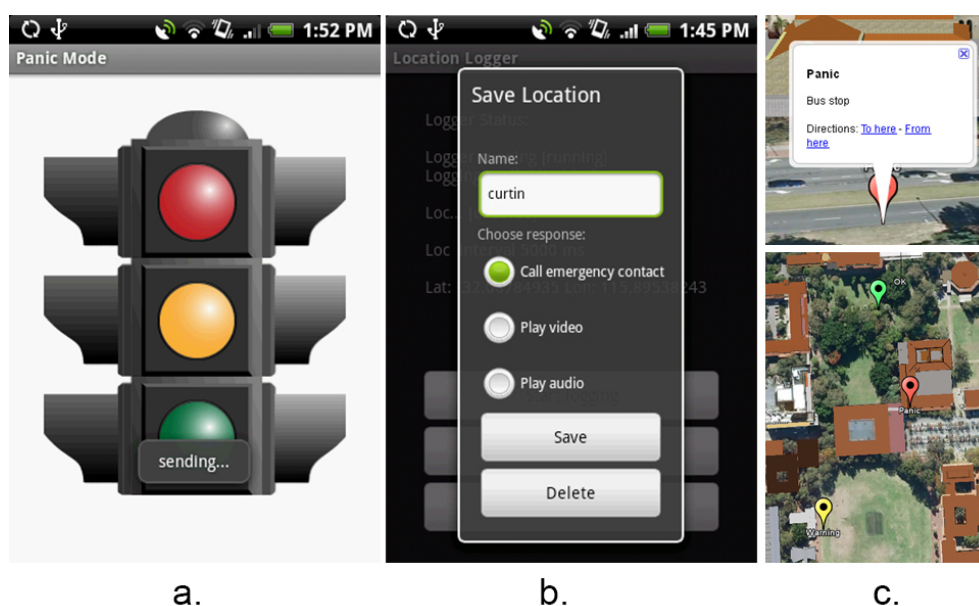


Figura 15 – Interface da ferramenta desenvolvida

- a. Tela com botão para indicar pânico, aviso ou se está tudo certo.
- b. Configuração para resposta de pânico sensível à localização.
- c. Entradas de alertas por localização incluindo o tipo de pânico

Fonte: Adaptado de Mohammedali et al. (2011)

A aplicação pode realizar uma ou ambas ações disponíveis (chamar um contato de emergência ou reproduzir mídia). Para os níveis de pânico elevado e do “ok”, um alerta via SMS é enviado para o responsável com as informações de localização e o grau do pânico registrado.

Em relação a identificação de situações de risco, são encontrados na literatura diversos trabalhos voltados à detecção de quedas. O trabalho desenvolvido por Rougier et al. (2011) é um deles, no qual retrata a utilização da visão computacional para analisar o comportamento das pessoas e detectar eventos incomuns, como quedas (Figura 16). A técnica é empregada rastreando a silhueta do indivíduo ao longo de uma sequência de vídeos capturados (5 Frames Por Segundo (FPS)) e logo após alisadas as anomalias encontradas nessa sequência. As quedas são detectadas utilizando o modelo de mistura

de Gauss, do inglês *Gaussian Mixture Models* (GMM). Além disso, o sistema é instruído a enviar alertas para meios externos ao detectar eventos anormais.

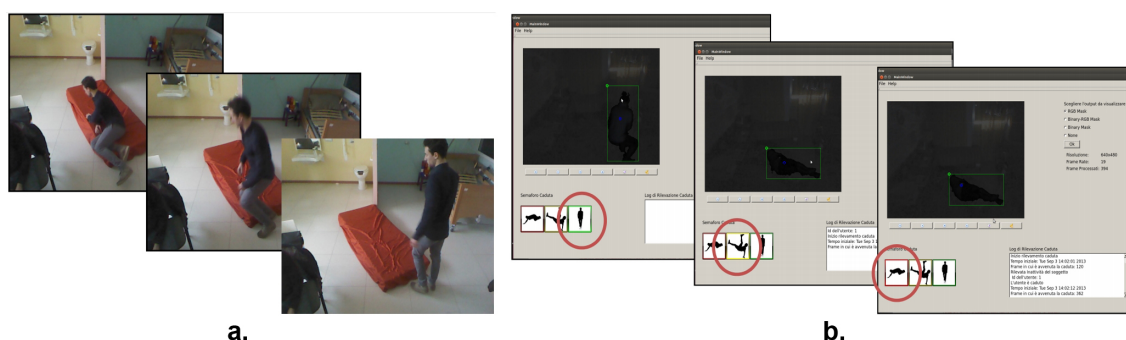


Figura 16 – Exemplo de caída simulada

a. Simulação de caída b. Detecção de caídas em ambiente com pouca luz

Fonte: Adaptado de Bevilacqua et al. (2014)

Em Bevilacqua et al. (2014) também é desenvolvido o trabalho de detecção de quedas de pessoas com deficiência física utilizando visão computacional. É utilizada uma câmera RGB-D e o sensor *Kinect* para reconhecer a silhueta do corpo. O sistema proposto é capaz de detectar vários tipos de queda e não requer conhecimento prévio do ambiente, como o reconhecimento do chão no ambiente virtual. O subsistema para detectar a queda é um módulo de software baseado nas bibliotecas *OpenNI* e *OpenCV*. Também é utilizado o filtro de *Kalman* para reduzir a captação de ruídos. A abordagem baseia-se na avaliação da redução da largura, altura e profundidade e a velocidade em que isto ocorre, bem como a posição do indivíduo no espaço.

O trabalho de Shuo (2015) é desenvolvido para atender a necessidades emergenciais de idosos utilizando o Arduino FIO. O objetivo é projetar um sistema de detecção de queda precisa com melhoria acentuada na precisão e especificidade da ocorrência, baseado no acelerômetro existente na placa. É colocado no utilizador um cinto elástico para obter dados da aceleração ao longo do tempo. A decisão para comprovar a queda considera apenas a detecção de picos da aceleração, ignorando o comportamento do vector de aceleração. Para notificar uma queda, um e-mail é enviado como alerta aos cuidadores.

Devido à ampla presença no cotidiano e custo acessível, os Smartphones têm sido utilizados como dispositivos de detecção devido minimizarem custos, como de instalação operacional, construção e distribuição. Como consequência, muitos sistemas de detecções de queda baseados em Smartphones têm sido propostos pela Literatura durante os últimos cinco anos, enquanto que há uma diminuição do número de novos protótipos que são implementados em hardware para estes fins especiais.

Ainda sobre o contexto em destaque, em Yavuz et al. (2010), é proposto um detector de quedas que utiliza o acelerômetro do Smartphone para detecção e incorpora

diferentes algoritmos, como o *thresholding* e a transformada de *wavelet* para extração de características a fim de reconhecer falsas ocorrências. A aplicação também fornece informações de localização e ativação de alertas para enviar via SMS, e-mails e até mensagens em redes sociais aos cuidadores.

Uma das características mais importantes do aplicativo é a sua interface simples com quatro botões grandes na tela principal (Figura 17). São eles o *Start*, botão correspondente inicialização do serviço de detecção em segundo plano; o *Stop*, que interrompe o serviço; O *Panic Button*, usado para o envio manual de alertas emergenciais aos cuidadores; e o botão *Settings*, onde as configurações e dados de contato emergenciais e informações do usuário são guardadas.



Figura 17 – Página inicial do aplicativo

Fonte: Yavuz et al. (2010)

Trabalhos sobre botões de pânico e seus similares, baseados em *Smartphones*, podem ser encontrados em Woei (2002), Ulanday et al. (2015), Shinde et al. (2012), Piva et al. (2014), Jadhav et al. (2014), Kumar e Kumar (2014), Gogoi e Sharma (2013), Rocha et al. (2015) e Vijayanti et al. (2014). A ideia geral dos trabalhos destacados segue a mesma linha de raciocínio de utilizar o dispositivo móvel como meio de solicitação de socorro, porém aplicada a diferentes áreas com contextos específicos e estáticos.

Em relação a detecções de ocorrências em ambientes automotivos, no trabalho apresentado por Kim e Song (2013), é desenvolvido um sistema de reconhecimento de veículos utilizando sensores para ativação automática da frenagem. Com base no movimento dos automóveis, é possível reconhecer a forma do veículo e seu estado, se o mesmo é dinâmico (está em movimento) ou estacionário (está parado). O objetivo do trabalho é prevenir acidentes, detectando aproximação de automóveis rapidamente e acionando o sistema de frenagem do veículo para redução da velocidade. O algoritmo utilizado na aplicação é aprimorado para não detecção de falsos eventos (Figura 18).

Adibi (2014) apresenta um sistema M-Health automotivo para aumentar a



Figura 18 – Aperfeiçoamento da diferenciação entre carros e objetos

Fonte: Kim e Song (2013)

segurança dos ocupantes de um veículo. A proposta exposta pelos autores dá-se da utilização de sensores biomédicos para captura e emissão de sinais vitais emitidos pelo corpo, a fim de avaliar e monitorar as condições de saúde do indivíduo em tempo real e prever incidentes. O sistema utiliza a tecnologia de telefonia móvel 4G (quarta geração), que sua utilização possibilita o controle da localização do veículo, bem como a obtenção das informações vitais relacionadas à saúde dos ocupantes do automóvel.

No trabalho de Amarante et al. (2015), foi desenvolvida uma aplicação para prover segurança no trânsito, realizando a detecção automática de acidentes em tempo real. A ideia é detectar e enviar alertas a veículos próximos ao acidente e para dispositivos instalados nas vias de trânsito, para retransmiti-los a servidores web. A comunicação entre os veículos se dá por meio da rede de comunicação intraveicular, do inglês *Veicular Ad Hoc Networks* (VANET). Também foi utilizado o módulo OBD-II, com os parâmetros de velocidade e informações sobre o airbag (caso exista). A aplicação foi avaliada em um dispositivo real de comunicação e obteve resultados positivos, com transmissão de alertas com latência abaixo de 100 milissegundos e de alcance superior a 150 metros.

Através da combinação de Smartphones com veículos com Sistema de Diagnóstico a Bordo, do inglês *On-Board Diagnostic System II* (OBD-II), Zaldivar et al. (2011) apresentam uma aplicação, baseada na plataforma Android, para monitoramento do veículo por meio da interface OBD-II e detecção de acidentes. A conexão entre o OBD-II e o dispositivo móvel é ativada por meio do Bluetooth. Através da força G empregada pelo passageiro, no caso da ocorrência de uma colisão frontal, o alerta é detectado e disparado. Em 3 segundos, são enviados detalhes sobre o acidente por meio de e-mail ou SMS para serviços de emergência, juntamente com a localização do veículo.

A detecção de acidentes se dá por meio da obtenção e da análise dos parâmetros obtidos pelo OBD-II, tais como ativação de airbag, desaceleração rapidamente, etc. O complemento dessas informações são obtidas pelo próprio Smartphone, como localização e envio de solicitação de socorro. O aplicativo também oferece informações de uso geral para o motorista, incluindo os níveis de gás, detecção de falhas em elementos mecânicos, dados do motor, entre outros (Figura 19).

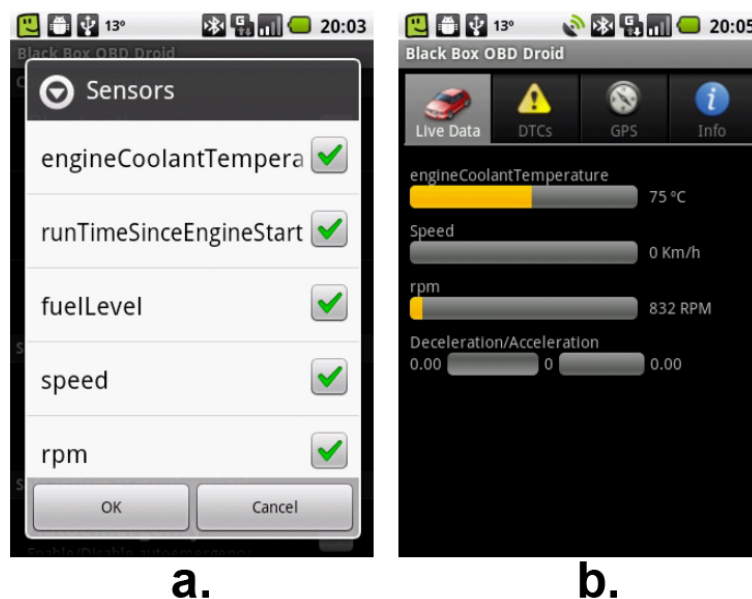
**a.****b.**

Figura 19 – Telas da aplicação

a. Menu de seleção de parâmetros b. Visualização de parâmetros em tempo real

Fonte: Adaptado de Zaldivar et al. (2011)

Thompson et al. (2010) criaram um aplicativo chamado WreckWatch, para detecção de acidentes baseado no acelerômetro do Smartphone e dados acústicos. Quando detectado um acidente, o dispositivo móvel utiliza a 3G para enviar um alerta ao servidor e este processa a informação e a exibe na aplicação web. Além disso, fornece informações em tempo real do acidente através de fotografias, coordenadas do GPS, chamadas de voz e gravação de dados. Essas informações são enviadas ao socorrista previamente. Essa solução exige que o Smartphone esteja fixo dentro do carro para evitar detecção de um falso acidente.

Em relação a reconhecimento de atividades, em Duarte (2013), é desenvolvida uma aplicação para Android que utiliza os sensores embarcados dos Smartphones em conjunto com tecnologias web para realizar classificação de atividades. A solução é feita com base na arquitetura cliente-servidor, em que a aplicação cliente realiza a aquisição de dados dos sensores e gravação do sinal, e a aplicação servidor recebe a informação adquirida pelo cliente, processa-a e classifica de acordo com um algoritmo previamente treinado. O trabalho foi testado posicionando o Smartphone junto à cintura, dentro do bolso frontal direito e obteve precisão de 95%.

Rocha (2012) desenvolve o MonitorMe, um sistema de monitorização online que permite o reconhecimento de atividades e a gravação de um vídeo do ambiente no qual o indivíduo esteja inserido. O sistema inclui um Smartphone Android, mantido na parte superior, e um módulo *Magnetic, Angular Rate and Gravity* (MARG) colocado na parte inferior. A aplicação obtém dados dos sensores integrados em ambos os dispositivos para a realização do reconhecimento online de 6 atividades diferentes: de

pé, sentado, deitado, andando, correndo e queda (Figura 20). O reconhecimento é obtido utilizando um algoritmo de baixo custo computacional, cujo desenvolvimento levou em consideração as restrições relativas à capacidade de processamento e à duração da bateria dos dispositivos móveis.

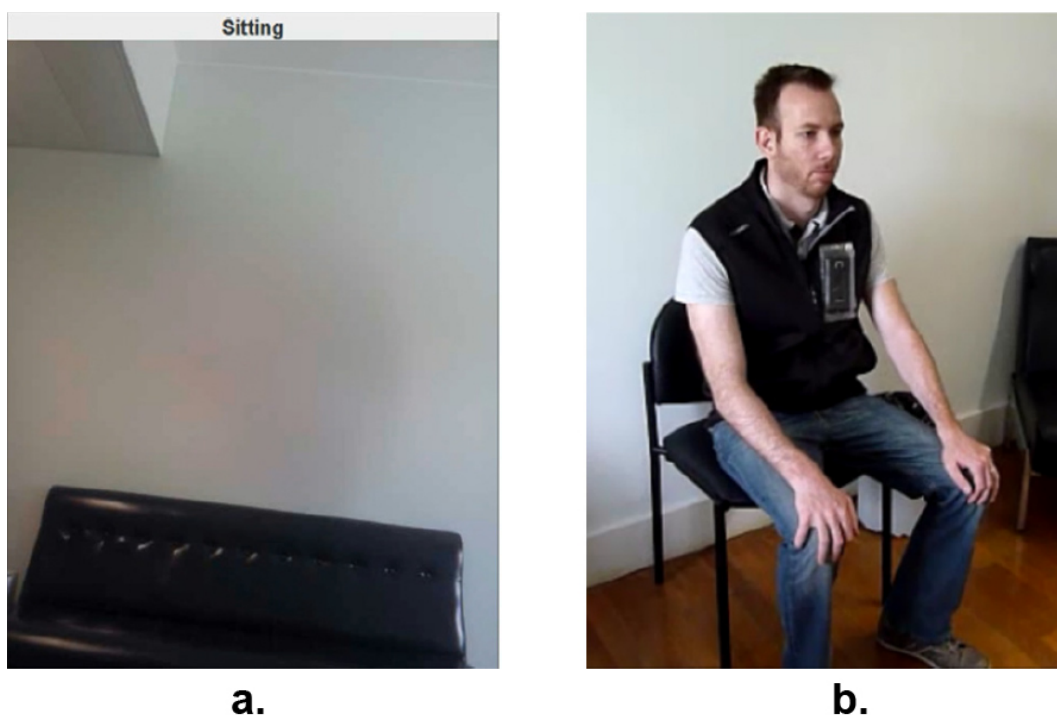


Figura 20 – Arquitetura da aplicação

a. Captura de tela do dispositivo móvel b. Identificação da atividade a partir do vídeo
Fonte: Rocha (2012)

Também foi realizada uma pesquisa de mercado, na qual foram analisados aplicativos para Smartphones com o sistema operacional Android na *Google play*. Na loja foi possível encontrar aproximadamente 40 aplicativos voltados à solicitação de socorro imediato e apenas alguns que atuam na detecção de acidentes automobilísticos. Dos aplicativos analisados, podem-se destacar 5 que têm semelhança com este trabalho.

O primeiro deles, o Monban, desenvolvido pela empresa Akash (2013), é um aplicativo de segurança pessoal baseado na redução do tempo de resposta a situações de risco. Com ele é possível enviar alertas para contatos de confiança via e-mail, contendo histórico de localização e áudios gravados. Há também um botão de alerta com acesso rápido e um mecanismo preventivo de agendamento de alertas, em que o Smartphone fica enviando check-ins para ter conhecimento do estado do usuário. Caso estes alertas não sejam respondidos, notificações são enviadas. Além disso, também é possível marcar locais considerados perigosos para que seja ativado o modo de segurança nesses locais caso o celular seja desligado.

Gemin (2015) desenvolveu um botão de pânico que utiliza como ativador um dispositivo externo sem fio Bluetooth, geralmente comercializado para tirar fotos *selfies*.

Quando o usuário pressiona o botão em uma situação de emergência, o mesmo se conecta ao Smartphone e envia pedidos de socorro via SMS e e-mail configuradas previamente. Uma vez registrada a ativação do botão, é enviada a localização GPS do celular a cada 10 minutos.

O aplicativo desenvolvido pela empresa SOSmart (2016) tem o mesmo funcionamento de um botão de pânico, acrescido da detecção de acidente veicular com base na velocidade, taxa de deslocamento e sensores do Smartphone. Ao detectar o impacto, a aplicação envia uma notificação com o local do acidente para contatos de emergência e informa a melhor rota até o local. O aplicativo não possui identificação de falsas ocorrências e não entra em contato diretamente com centrais de emergência.

A empresa iXtencia (2015) também desenvolveu um aplicativo para emergências que permite enviar alertas SOS para amigos e familiares rapidamente através de gatilhos disparados por meio de agitos no dispositivo ou gestos predefinidos. O alerta inclui o envio da localização do usuário, o nível de bateria do dispositivo e um clipe de áudio da situação. O gesto de trepidação permite solicitar ajuda sem esforço e discretamente. A aplicação fica executando como serviço e a calibração da intensidade da vibração pode ser personalizada, bem como o gesto de ativação (Figura 21). A solicitação de socorro só é ativada mediante a detecção do disparo do gatilho.

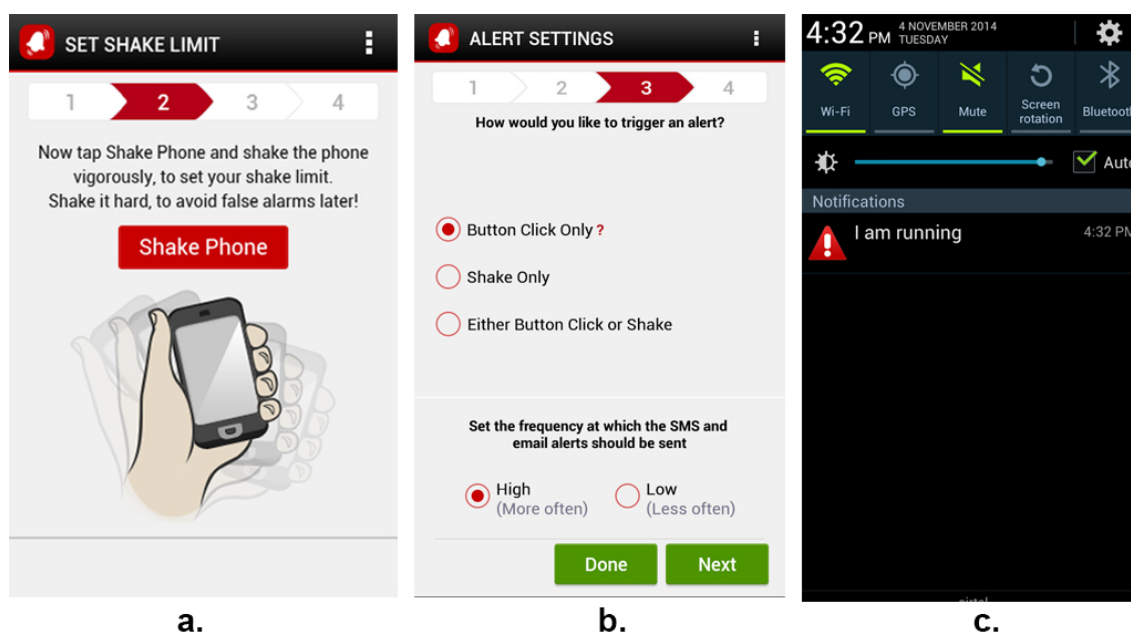


Figura 21 – Telas principais

- a. Ativação do alerta b. Configuração do gatilho c. Aplicação rodando em serviço
Fonte: iXtencia (2015)

Para uma melhor compreensão entre as áreas de atuação dos trabalhos citados, a Tabela 2 apresenta um comparativo entre estes trabalhos e o SOS Móvel, usando como comparativo algumas nomenclaturas: M-Health (MH); Sensibilidade ao Contexto

(SC); Reconhecimento de Atividades (RA); Monitoramento (MO); Detecção de Situações de Risco (DSR); Mobilidade (MB); Sensores (SE); Solicitação de Socorro (SS); Agentes Inteligentes (AI); Tecnologia Assistiva (TA); Hardware Específico (HE).

Tabela 2 – Tabela de comparação entre os trabalhos apresentados

Autores	MH	SC	RA	MO	DSR	MB	SE	SS	AI	TA	HE
Hadjinicolaou et al. (2009)	★			★		★					
Sethia et al. (2014)	★					★		★		★	
Das e Alam (2014)	★	★				★	★				
Carvalho et al. (2011)		★	★	★	★		★	★	★	★	★
Besaleva e Weaver (2014)	★	★		★		★		★			
Mohammedali et al. (2011)	★	★				★		★		★	
Rougier et al. (2011)			★	★	★		★	★	★		★
Bevilacqua et al. (2014)					★		★				★
Shuo (2015)					★		★	★	★		★
Yavuz et al. (2010)	★	★			★	★	★	★	★		
Kim e Song (2013)				★	★		★		★		★
Adibi (2014)	★	★		★	★	★	★	★	★		★
Amarante et al. (2015)				★	★		★	★	★		★
Zaldivar et al. (2011)	★	★		★	★	★	★	★	★		★
Thompson et al. (2010)	★	★			★	★	★	★	★		
Duarte (2013)			★	★		★	★				
Rocha (2012)			★	★		★	★				★
Akash (2013)	★	★		★		★	★	★			
Gemin (2015)	★	★				★	★	★	★		★
Médéric (2015)	★	★				★	★	★			
SOSmart (2016)	★	★		★	★	★	★	★	★		
iXtentia (2015)	★	★			★	★	★	★	★		
SOS Móvel	★	★	★	★	★	★	★	★	★	★	

Apesar de existirem diversos trabalhos que apresentam ferramentas de detecção de riscos, reconhecimento de atividades, sensibilidade ao contexto, sensores e aplicação da M-Health, estas ferramentas possuem condições muito particulares, e muitas delas

somente podem ser aplicadas a públicos específicos.

Este trabalho se concentra no desenvolvimento de um sistema que possa atender pessoas em diferentes contextos e agilizar o atendimento médico através da detecção de situações de risco. Neste cenário, o SOS Móvel se diferencia das demais ferramentas presentes na Literatura por possuir em conjunto todas as características inexistentes nas demais e não necessitar de um hardware específico para a finalidade descrita.

4 SOS MÓVEL: SISTEMA PARA AUXILIAR NO PEDIDO DE SOCORRO E DETECÇÃO AUTOMÁTICA DE SITUAÇÕES DE RISCO

Este capítulo é destinado à apresentação do SOS Móvel, objeto de desenvolvimento desta dissertação de mestrado. Neste capítulo serão apresentadas a contextualização e descrição do sistema, suas funcionalidades, a metodologia e o desenvolvimento do sistema, organizados em:

- Visão geral: apresentação das circunstâncias que levaram a criação do SOS Móvel e denotação da ideia conceitual e apresentação do SOS Móvel;
- Etapas: exposição das etapas utilizadas para desenvolvimento do SOS Móvel. São elas o pré-processamento, processamento, pós-processamento, como também a modelagem do agente;
- Implementação: apresentação da implementação do sistema, ferramentas utilizadas, métodos e funcionalidades do SOS Móvel.

4.1 VISÃO GERAL

O SOS móvel é um sistema de detecção e solicitação de socorro formado por duas partes que interagem entre si: um aplicativo para Smartphones e dispositivos móveis com SO Android, e uma interface de gerenciamento na internet. O aplicativo é o componente, que será utilizado ativamente pelo usuário, enquanto a interface web permitirá que familiares e/ou gestores da saúde possam monitorar e acompanhar remotamente o usuário, como também visualizar seu histórico de alertas e localização.

A aplicação móvel leva em consideração dois níveis de solicitação de socorro, são eles o manual e o automático. O primeiro, o manual, possibilita ao usuário escolher como realizar seu pedido de socorro, podendo optar entre avisar familiares ou enviar um alerta de emergência diretamente aos serviços de atendimento de urgência. Nesse modo, o usuário precisa interagir diretamente com o dispositivo móvel para realizar a solicitação.

No segundo nível, o automático, o SOS Móvel permanece constantemente detectando o que o usuário está fazendo e define situações anormais para cada atividade identificada, assumindo que o mesmo esteja em perigo. Para cada atividade detectada, há um conjunto de condições para que essa ação seja reconhecida como atípica. Ao constatar a anormalidade, se o usuário não confirmar uma detecção de um “alarme falso”, automaticamente é disparado um alerta aos cuidadores e familiares contendo sua geolocalização, horário e data. Nesse nível, o usuário não necessita interagir com o Smartphone diretamente.

O SOS Móvel foi desenvolvido para ser sensível ao contexto, utilizando 5 das 4 dimensões que descritas por Dey e Abowd (2000) na Seção 2.1.3, do Capítulo 2 deste trabalho. As dimensões usadas possibilitam perceber a atividade do usuário, sua localização, momento e data, enviar notificações e adaptar-se aos contextos do usuário. A adaptação ao contexto do usuário diz respeito à possibilidade de personalização das mensagens de socorro de acordo com a necessidade e situação do usuário, seja ele deficiente, patológico ou não. Outro ponto importante a ser destacado é a utilização de agentes inteligentes para atuar na solicitação de socorro em conjunto com o reconhecimento de atividades por meio da API Activity Recognition. Com o uso de agentes no SOS Móvel, foi possível executar ações com mais precisão. Foi utilizado o agente do tipo baseado em objetivos, o qual fica constantemente avaliando as ações e verificando se estas se aproximam do objetivo que se deseja alcançar. Com isso, o reconhecimento de atividades obteve uma redução de falsas ocorrências. O reconhecimento das atividades se dá por meio da Interface de Programação de Aplicações, do inglês Application Programming Interface (API), do Google Activity Recognition. São reconhecidas seis atividades, são elas: parado, caminhando, correndo, bicicleta, colisão frontal e capotamento. As situações de risco são identificadas por meio do acelerômetro do Smartphone, no qual é fixado um valor entendido como anormal para cada atividade, separadamente. O valor foi obtido por meio de testes e experiências realizadas, que serão descritas no capítulo 5.

4.2 ETAPAS

O desenvolvimento do SOS Móvel foi realizado em cinco etapas, detalhadas nas subseções seguintes. São elas: a configuração do ambiente, a aquisição de dados, o pré-processamento, o processamento e o pós-processamento. A Figura 22 apresenta a arquitetura do SOS Móvel, bem como sua visão geral e etapas seguidas desde a coleta de dados até a obtenção dos resultados.

4.2.1 Configuração do ambiente

A configuração do SOS Móvel inicia-se pela interface móvel, onde o usuário é submetido à realização de um cadastro, sendo obrigatório, pois nele é possível o usuário informar doenças existentes e contatos familiares de emergência. A personalização das mensagens de socorro também faz parte da configuração do ambiente, pois estas são salvas e enviadas ao servidor como alertas com sensibilidade ao contexto.

Por meio da interface de gerenciamento web, é possível receber os alertas enviados pela interface móvel e localizar o usuário. A conexão via web se dá por meio do login associado ao número de telefone dos contatos do usuário cadastrado.

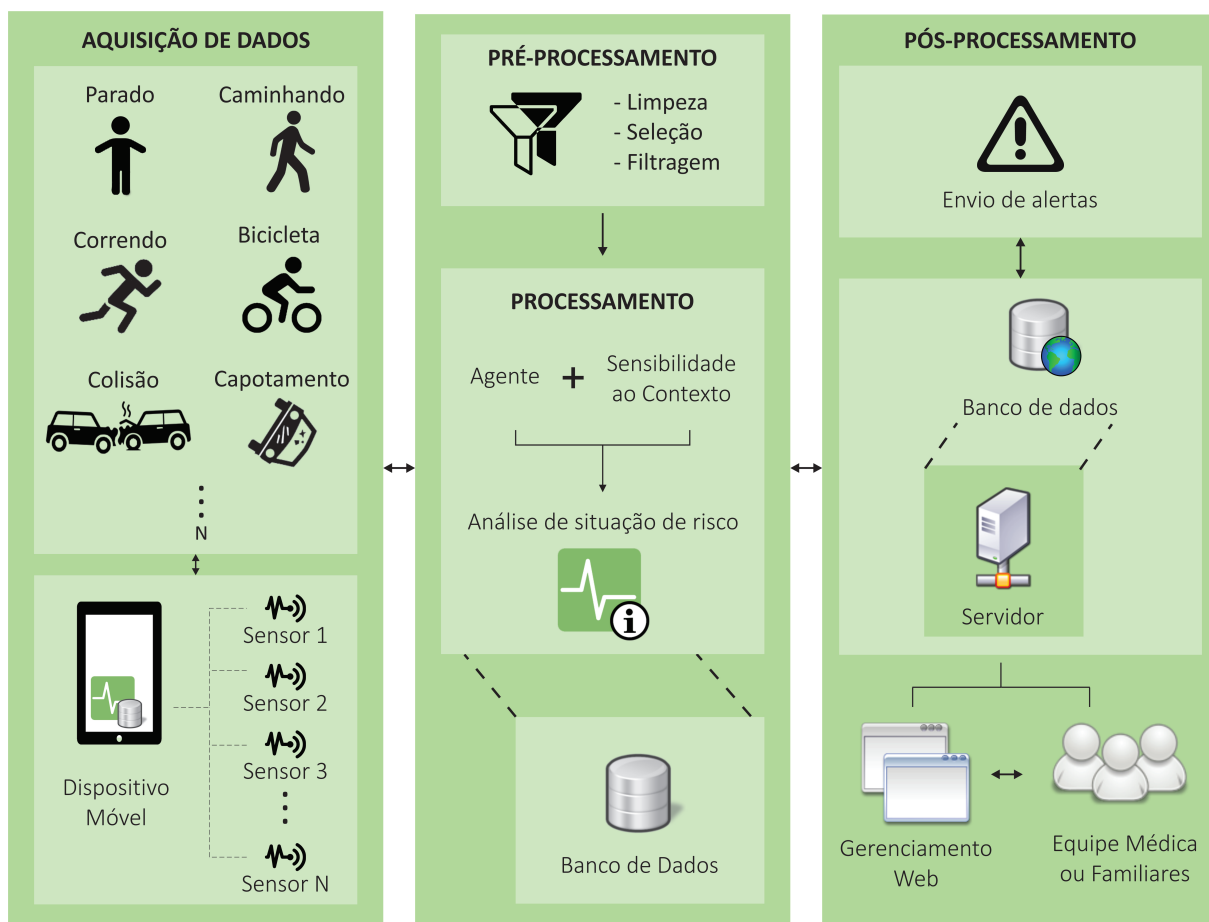


Figura 22 – Visão geral do SOS Móvel

Fonte: Autoria Própria

4.2.2 Aquisição de dados

A aquisição de dados foi realizada através da utilização do acelerômetro, giroscópio e magnetômetro presentes no próprio Smartphone, em conjunto com a API para executar constantemente a detecção do que o usuário está fazendo. Os dados coletados foram inicialmente dos sensores de aceleração (normal e linear), de rotação e intensidade de campo magnético, de localização, além das 6 atividades reconhecidas pela API usada (parado, caminhando, correndo, bicicleta, colisão frontal e capotamento).

4.2.3 Pré-processamento

Essa etapa foi responsável pela realização da filtragem e da seleção dos principais sensores a serem utilizados para este trabalho, que foram o sensor de aceleração linear, o giroscópio, o magnetômetro e o GPS. Foi constatado que, ao utilizar apenas o acelerômetro, o eixo z permanecia com seu valor constante de +9,81 mesmo em seu estado de repouso total. Esse valor corresponde à aceleração do dispositivo, em m/s^2 , levando em consideração a força gravitacional da terra.

Dessa forma, foi utilizado o sensor de aceleração linear, pois o mesmo considera os valores do eixo z já com os descontos gravitacionais, justificando o uso do sensor. Entretanto, foi também utilizado indiretamente o acelerômetro, em razão de que a API utiliza o acelerômetro em seu estado interno para o cálculo e o reconhecimento das atividades, o que resulta no uso total de 4 sensores base para o desenvolvimento deste trabalho. São eles: o giroscópio, o acelerômetro, o magnetômetro e a aceleração linear.

Foi também realizada a restrição das atividades reconhecidas pela API, reduzindo-as a 6. São elas: parado, caminhando, correndo, bicicleta, colisão frontal e capotamento. Nessa etapa também foram definidas quais seriam as situações de emergência para cada atividade citada. Para as atividades parado, caminhando, correndo e andando de bicicleta, as possíveis situações de emergência estabelecidas para detecção foram queda e desmaio. Para a atividade de dirigir o carro, é possível identificar uma batida (colisão frontal) e um capotamento.

4.2.4 Processamento e Pós-Processamento

Nessa etapa, encontra-se a parte principal do desenvolvimento do trabalho, abordando como foi desenvolvido o agente e incorporada a sensibilidade ao contexto, para fornecer o resultado da análise e a detecção das emergências. Como mencionado anteriormente, o SOS Móvel possui um agente que permanece constantemente observando a atividade que o usuário está fazendo, por meio da *API Activity Recognition*.

Basicamente, a tarefa do agente se divide em três partes. A primeira delas é receber a dedução da atividade por meio da API e armazenamento de seu resultado. Depois disso, receber os dados dos sensores de aceleração linear, giroscópio e magnetômetro para calcular a rotação do celular. Na terceira e mais importante parte, o agente permanece verificando constantemente a aceleração linear e, se detectar variações maiores que 5.0 m/s^2 nos três eixos, o mesmo analisa os dados das outras duas partes descritas para, assim, determinar se houve um acidente ou não.

O Algoritmo 1 descreve o processo que o agente realiza para obtenção dos dados dos sensores e cálculo da rotação do celular. O método descrito nesse Algoritmo é chamado sempre que algum sensor sofre alguma mudança e é invocado pelo próprio Android na aplicação, como forma de avisar que tem novos dados identificados. Dependendo da capacidade do dispositivo e da taxa de atualização estabelecida, esse método é chamado aproximadamente de 200 a 500 vezes por segundo para cada sensor separadamente.

Pode-se observar, nas linhas de 2 a 12, o armazenamento dos dados de acordo com o tipo de sensor utilizado. Nesse caso, três variáveis correspondentes ao acelerômetro, ao magnetômetro e à aceleração linear, respectivamente. Se o agente já possuir esses dados, então é obtida a matriz de rotação por meio de acelerômetro, magnetômetro e dois vetores.

Algoritmo 1 Algoritmo que monitora os sensores do *Smartphone*

```

1: função ALTERACAOSENSOR(mudancaSensor)
2:   se mudancaSensor.tipo = acelerometro então
3:     acc ← mudancaSensor.values
4:   senão
5:     se mudancaSensor.tipo = magnetometro então
6:       mag ← mudancaSensor.values
7:     senão
8:       se mudancaSensor.tipo = aceleracaoLinear então
9:         acclin ← mudancaSensor.values
10:      fim se
11:    fim se
12:  fim se
13:
14:  se acc <> vazio e mag <> vazio e acclin <> vazio então
15:    R[] ← real[9]
16:    I[] ← real[9]
17:    sucesso ← monitoraSensor.matrizRotacao(R, I, acc, mag)
18:    se sucesso então
19:      orientacao[] ← real[3]
20:      monitoraSensor.pegarOrientacao(R, orientacao)
21:      eixoZ ← orientacao[0]
22:      eixoX ← orientacao[1]
23:      eixoY ← orientacao[2]
24:      adiciona a janelaGiroscopio(Ponto3D(eixoX, eixoY, eixoZ))
25:      se janelaGiroscopio.tamanho > LIMITE_DA_JANELA então
26:        Remove valor mais antigo da janela
27:      fim se
28:    fim se
29:  fim se
30:  Executa função de checagem do status para verificar a ocorrência de acidentes
31: fim função

```

Conforme referido no Subseção 2.1.1.1 do Capítulo 2, existem processos de transformação de coordenadas para obtenção de dados dos sensores. A plataforma Android auxilia neste aspecto com classes e interfaces que contêm métodos específicos na obtenção de matrizes rotação, obtenção da orientação, dentre outros fatores relevantes. O método nativo da API Android, o *getRotationMatrix*, nomeado no Algoritmo 1 como *matrizRotacao* (Linha 17), gera a matriz de rotação responsável por mapear as coordenadas do dispositivo em coordenadas geográficas.

Nas Linhas de 19 a 23, é obtida a matriz de rotação da posição do celular nos eixos Z, X, e Y e adicionada essa posição à variável *janelaGiroscopio* (Linha 24), para representação de um ponto de três eixos. Esse procedimento foi necessário apenas ao giroscópio, devido à necessidade de o agente analisar o fluxo observando o janelamento de dados, como um conjunto de valores de uma fila de 200 posições. A análise de apenas

um valor pontual para o caso do giroscópio não é relevante, visto que o que se busca é saber se houve rotação. E, para identificar a rotação, é preciso comparar o valor do dado atual do sensor com os outros valores anteriores dentro de uma janela de tempo.

O *Ponto3D* é um vetor dinâmico (*arraylist*) dentro da variável *janelaGiroscopio*. Após adicionada a posição do dispositivo na janela de dados *janelaGiroscopio*, é verificado se o tamanho do janelamento é superior ao limite estabelecido (200), para que, se for, seja removido o valor excedente de posição mais antiga, para recebimento de um novo no final da fila (Linhas 25 a 27).

Dentro do janelamento, é analisado apenas o eixo Y. É percorrido todo o vetor da janela em busca de mudança de sinal. Primeiramente, é obtido o valor mais recente e verificado se o mesmo é menor do que zero. Se for, ou seja, se for negativo, então é percorrido todo o vetor do eixo Y em busca de algum número que seja positivo, pois isso indicaria uma mudança de sinal. Inversamente, se o primeiro valor não for negativo, também é percorrido o vetor para realizar a mesma ação em busca de um número oposto. Como resposta, é retornado *true* caso achar e, se não, retorna *false*, indicando que não houve rotação.

O Algoritmo 2 determina se houve acidentes e qual, de acordo com a atividade realizada pelo usuário. Tal atividade, como citado anteriormente, é detectada a partir da API *Activity Recognition* em conjunto com o agente, que permanece constantemente recebendo as atualizações da mesma. Conforme as linhas de 15 a 21, para a atividade “VEÍCULO” detectada, havendo existência de rotação, é deduzida uma situação de capotagem. Em contrapartida, não havendo, é inferido que ocorreu apenas uma colisão.

Em relação a sensibilidade ao contexto, no trabalho foram contempladas 4 dimensões abordadas por Zimmermann et al. (2007) no Capítulo 2. São elas as dimensões “quem”, “onde”, “quando” e “o quê”. Em outras palavras, o trabalho é sensível a quem está utilizando, ao local, ao momento do acontecimento e ao que o usuário está fazendo, respectivamente.

No que diz respeito à sensibilidade ao contexto do utilizador, o trabalho possui duas vertentes principais de uso: a do usuário e a do cuidador ou familiar responsável. A aplicação móvel é usada majoritariamente pelo usuário final solicitante, o qual poderá realizar o cadastro ou requerer emergência. Já a aplicação de monitoramento web pode ser utilizada por uma equipe de atendimento médico ou familiares do solicitante. Assim, é possível diferenciar ou visualizar quem realiza determinada atividade ou quem pode ser notificado caso o contexto seja alterado.

Para identificar onde o contexto se encontra, foi utilizado o GPS para geolocalização do usuário pelo Smartphone, em conjunto com a API do Google Maps para representação de sua localização em um mapa (web), quanto à obtenção e ao envio da posição por meio das coordenadas (Smartphone).

Uma das informações mais importantes, além da localização do usuário, são

Algoritmo 2 Algoritmo de checagem do *status*

```

1: função CHECARSTATUS
2:   se acclin[0] > ACELERACAO_LIMITE ou acclin[1] > ACELERACAO_LIMITE
   ou acclin[2] > ACELERACAO_LIMITE então
3:     se atividadeAtual = AtividadeDetectada.PARADO então
4:       Informar provável desmaio
5:     senão
6:       se atividadeAtual = AtividadeDetectada.ANDANDO então
7:         Informar provável queda
8:       senão
9:         se atividadeAtual = AtividadeDetectada.CORRENDO então
10:          Informar provável queda
11:        senão
12:          se atividadeAtual = AtividadeDetectada.BICICLETA então
13:            Informar provável queda
14:          senão
15:            se atividadeAtual = AtividadeDetectada.VEICULO então
16:              se houveRotacao então
17:                Informar Provável Capotagem
18:              senão
19:                Informar Provável Colisão
20:            fim se
21:          fim se
22:        fim se
23:      fim se
24:    fim se
25:  fim se
26:  senão
27:    Informar impacto não identificado
28:  fim se
29: fim função

```

seus dados temporais. Estes dizem respeito ao momento em que foi gerada a solicitação ou o momento em que foi detectada a situação de risco. O intervalo de tempo determina a quantos instantes uma entidade está em determinado contexto. Assim, juntamente com as coordenadas e a localização do usuário, também constam-se suas informações temporais.

A identificação do que o usuário está fazendo no momento é realizada pela API *Activity Recognition*, que dispõe das informações dos sensores do Smartphone para diferenciar as ações do usuário dentre 6 atividades de uso no trabalho. Dessa maneira, a sensibilidade ao contexto é usada nessas 4 principais vertentes descritas acima.

No SOS Móvel, foi utilizado o *SQLite* para criar um banco de dados da aplicação usando o modelo de dados relacional. O MySQL foi o Sistema de Gerenciamento de Banco de Dados (SGBD) utilizado para gerir as informações recebidas da aplicação SOS Móvel pelo servidor. A aplicação móvel possui um banco de dados interno, no qual foi

utilizado o `sqlite` para modelagem; possui um outro banco de dados externo que faz conexão com o servidor via JSP.

Ao realizar o cadastro, algumas informações do usuário são salvas no BD da aplicação e, logo em seguida, todas são sincronizadas para o BD do servidor. No gerenciamento web, é possível visualizar todas as ocorrências de emergência presentes no banco de dados para o usuário logado, bem como receber alertas de novas solicitações e suas respectivas informações emergenciais, como localização, SMS com prévia do motivo do pedido, entre outros.

4.2.4.1 Modelagem do Agente

Seguindo a arquitetura apresentada na Figura 22, foi modelado o agente utilizando a metodologia *MAS-CommonKADS+* (II, 2010), que é uma extensão do *CommonKADS* (SCHREIBER, 2000), uma metodologia da engenharia do conhecimento que utiliza vários modelos para representar o conhecimento empregado por um agente para realizar uma tarefa.

A *MAS-CommonKADS+* foi selecionada para o trabalho principalmente por permitir a modelagem detalhada do comportamento dos agentes; ser baseada em UML; e apresentar uma ferramenta de suporte à Linguagem de Modelagem de Agentes, do inglês *Agent Modeling Language* (AML). Foram utilizados 4 modelos da metodologia citada para representação do agente: modelo de objetos e recursos, de agentes, de tarefas e de papéis.

O modelo de objetos e recursos tem o objetivo de fornecer uma melhor definição do sistema, sem focar apenas nos agentes. Levando do pressuposto que em um sistema podem existir tanto entidades ativas como passivas, esse modelo é capaz de modelar essas entidades passivas na metodologia. Um recurso é uma aplicação específica que fornece informação e/ou serviço ao agente. Os objetos são aqueles identificados a partir dos requisitos do sistema (CHAGAS, 2013). No *link* é apresentado o modelo citado para o SOS Móvel: <https://goo.gl/4cljd7>, bem como anexo no Apêndice A.

O modelo de agentes tem o objetivo de definir o agente em termos de metas, papéis e tarefas, como também suas características comportamentais. É nesse modelo onde há uma descrição detalhada das características do agente, descrevendo seus recursos, as percepções, os atuadores, as condições de ativação e de parada e a arquitetura do agente (II, 2010). A Tabela 3 apresenta o modelo de agentes conceitual e a Figura 23 o visual.

No caso do SOS Móvel, conforme a Figura 23, o agente *DetectorRisco* possui como seu sensor o método *onSensorChanged()*, que é chamado no código sempre que há mudança nos valores dos sensores utilizados (acelerômetro, giroscópio e magnetômetro), ou seja esse método é responsável por receber constantemente os novos valores emitidos pelos sensores, como explicado na Subseção 4.2.4. O atuador do agente se dá por meio da análise da ação do usuário e seu envolvimento dessa ação em uma possível situação

Tabela 3 – Modelo conceitual do agente

Agente	DetectorRisco
Descrição	O agente executa a verificação constante da atividade que está sendo realizada, pelo usuário por meio da API e a armazena numa variável. Concomitantemente o agente guarda também os dados dos sensores de aceleração linear, o acelerômetro, giroscópio e o magnetômetro para identificar a rotação do smartphone e verifica a aceleração linear do dispositivo para cada ação do usuário identificada.
Objetivo	Identificar a ação que o usuário está executando e a partir disso identificar situações de risco de acordo com a atividade detectada e solicitar socorro.
Parâmetros de entrada	Dados dos sensores e da API de reconhecimento de ação. float[] mGravity; float[] mGeomagnetic; float[] orientation; float[] linearAcceleration; float azimuth, pitch, roll; String Atividade; onSensorChanged(SensorEvent sensorEvent)
Parâmetros de saída	Detectar acidentes ou situações de risco houveRotacao() checarStatus()
Condição de ativação	Executar a aplicação pelo menos uma vez para iniciar o serviço de monitoramento constante. linearAcceleration != null magnometro != null mGeomagnetic != null linearAcceleration[0] > ACELERAÇÃO_LIMITE linearAcceleration[1] > ACELERAÇÃO_LIMITE linearAcceleration[2] > ACELERAÇÃO_LIMITE
Condição de finalização	Forçar parada da aplicação
Condição de fracasso	Sensores do smartphone não detectados ou ausentes, falsa identificação de acidente. mSensorManager.getDefaultSensor (Sensor.TYPE_ACCELEROMETER) == null mSensorManager.getDefaultSensor (Sensor.TYPE_MAGNETIC_FIELD) == null mSensorManager.getDefaultSensor (Sensor.TYPE_LINEAR_ACCELERATION) == null

de risco, através do método *checarStatus()*. O método *houveRotacao()* foi utilizado como adicional para certificação dos resultados obtidos pelo método citado anteriormente.

O modelo de tarefas é responsável por descrever as funcionalidades que o sistema deve contemplar. Todas as atividades do sistema são apresentadas no diagrama com suas respectivas subtarefas, que são realizadas para completar o objetivo do agente. O

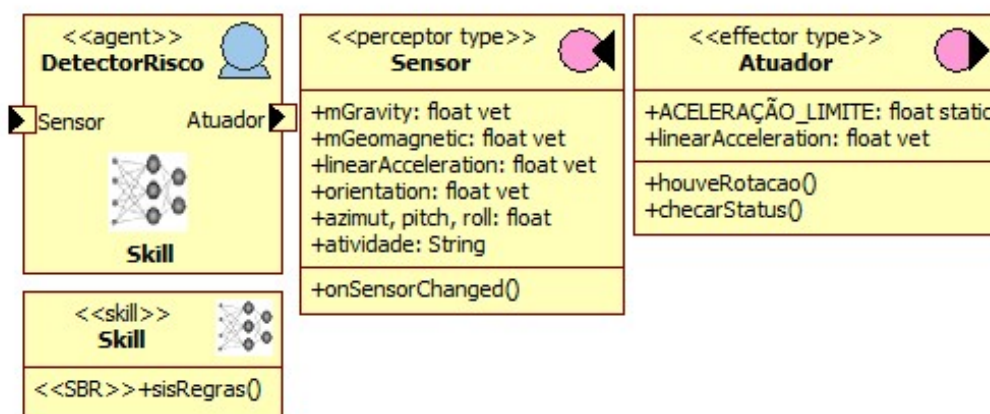


Figura 23 – Modelo de agentes visual

Fonte: Autoria Própria

modelo de tarefas descreve as atividades necessárias para atingir determinada meta (II, 2010). E, no caso específico do SOS Móvel, a tarefa principal do agente é solicitar socorro por meio da detecção de risco. Porém, para que essa tarefa seja executada, subtarefas necessitam ser realizadas a priori. O modelo de tarefas para o agente do SOS Móvel pode ser visualizado na Figura 24.

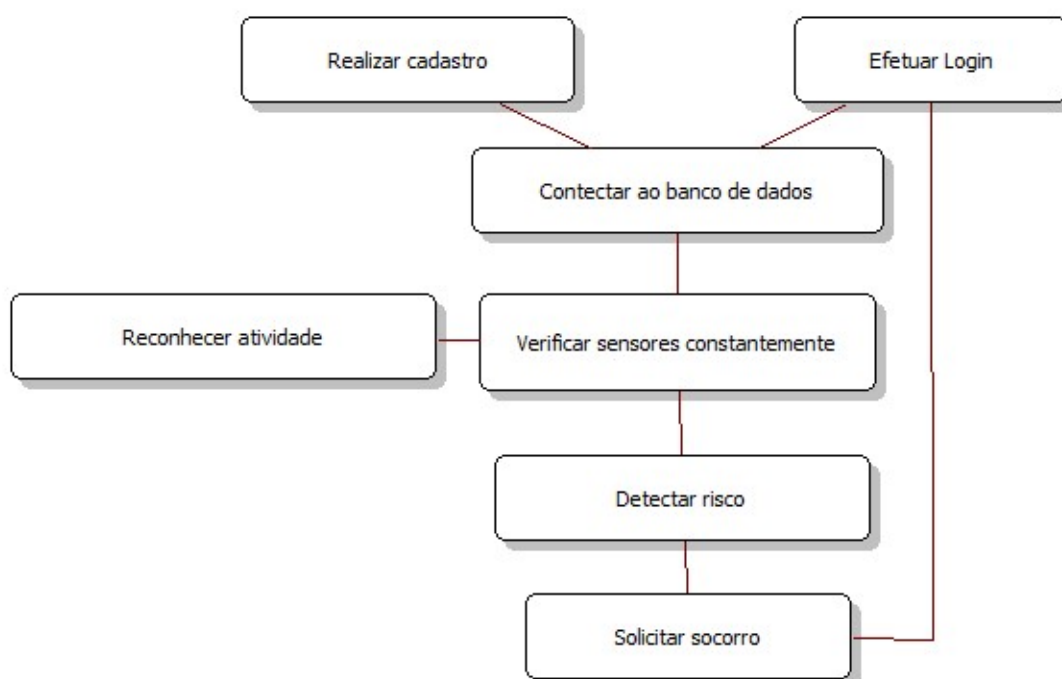


Figura 24 – Modelo de tarefas

Fonte: Autoria Própria

O modelo de papéis descreve as funções da organização e quais tarefas são

da responsabilidade de determinada entidade. Conforme II (2010), um papel é uma abstração que define qual tarefa um agente deve realizar no contexto o qual está operando. E este pode ser responsável por um ou mais papéis em um sistema. A metodologia *MAS-CommonKADS+* ajuda a descrever os papéis do agente, sem limitar suas inúmeras funções ou papéis que pode ter em uma organização. Com isso, o modelo de papéis da Figura 25 permite a modelagem de forma correta desses papéis.

No SOS Móvel, para o ambiente do dispositivo móvel, o agente *DetectorRisco* possui o papel de checar o *status* ou estado do usuário, além de também realizar a solicitação de socorro, realizando o procedimento descrito na subseção 4.2.4.

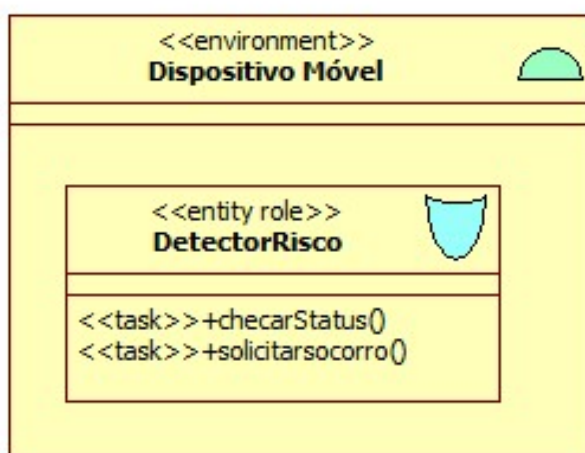


Figura 25 – Modelo de papéis

Fonte: Autoria Própria

O tipo do agente utilizado no trabalho foi o baseado em objetivos, o qual identifica a ação do usuário por meio da API e mantém a verificação constantemente para que a atividade detectada venha a se aproximar mais do objetivo que se deseja alcançar, que é a detecção correta de uma situação de risco.

4.3 IMPLEMENTAÇÃO DO SOS MÓVEL

4.3.1 Funcionalidades do SOS Móvel

O desenvolvimento do SOS Móvel tem como principal característica a concepção de um meio que possibilita acesso rápido as suas funcionalidades emergenciais. Nesta seção serão descritas todas as funcionalidades do SOS Móvel, tanto do aplicativo, quanto da interface de monitoramento web.

O cadastro do usuário é uma função obrigatória para utilização do SOS Móvel, pois é nele onde são informados dados básicos do usuário, como nome, telefone, doenças portadas e o número de três contatos emergenciais. As informações são salvas no banco

de dados da aplicação e, posteriormente, sincronizadas para o banco de dados do servidor. Os cadastros e demais informações são mantidos sincronizados de forma automática. A Figura 26 apresenta o fluxograma do cadastro do usuário.

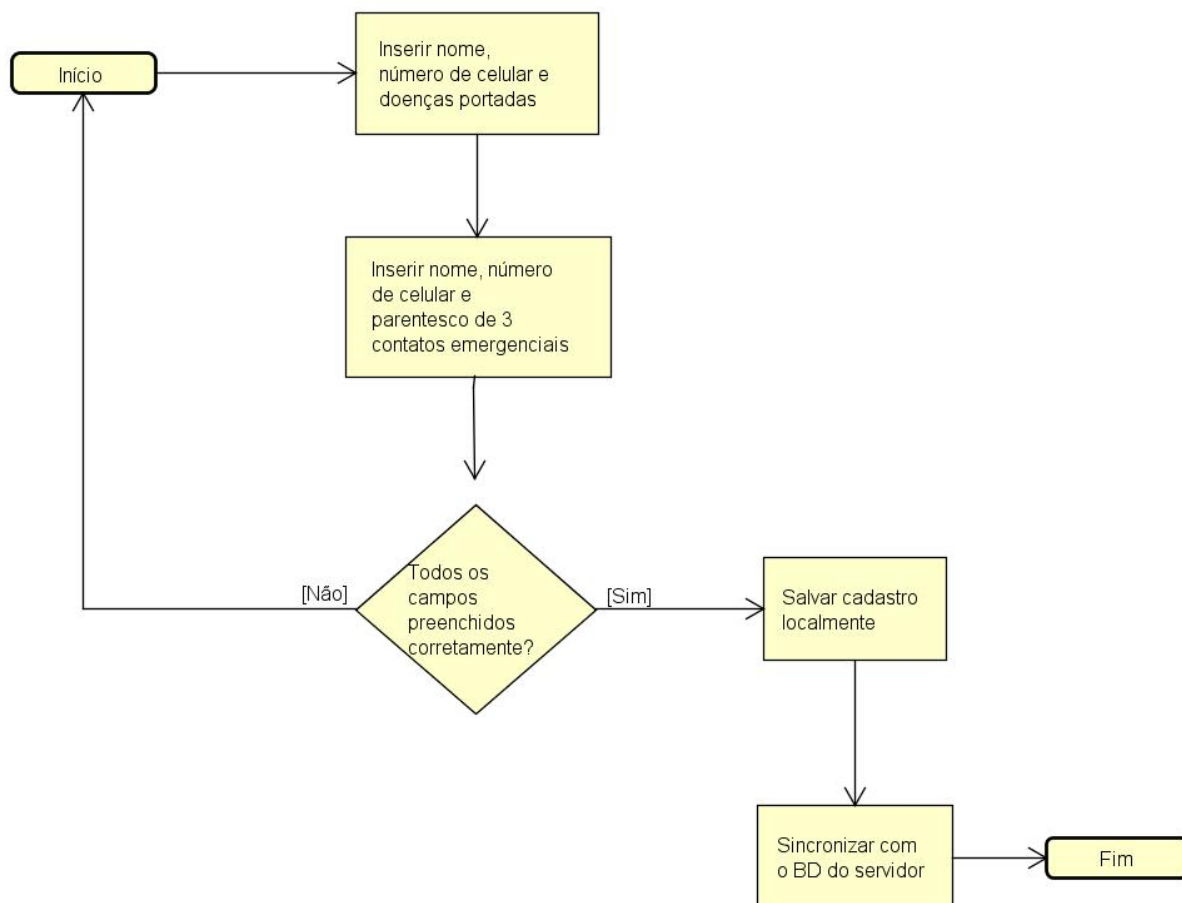


Figura 26 – Fluxograma do processo de realização de cadastro

Fonte: Autoria Própria

O usuário realiza login através apenas do número do celular informado. Este faz parte da tabela Usuários, que é a chave primária e serve como identificador único do usuário. Uma vez efetuado o cadastro, a aplicação não exige mais a realização do mesmo e permanece logado àquele número, a menos que seja apagada a aplicação ou seu *cache*. Esta funcionalidade estará presente apenas no aplicativo utilizado pelo usuário.

No cadastro, o usuário informa três números emergenciais para envio de mensagens rápidas, caso solicite ajuda manualmente. Na tela inicial, são exibidos três botões principais de texto, onde são contidas as mensagens predefinidas pelo usuário. Ao pressionar um dos três botões emergenciais, se confirmada, a informação descrita nele é enviada para os três números emergenciais do cadastro do usuário.

Essa funcionalidade tem o intuito de agilizar a comunicação emergencial, como também informar previamente a familiares ou a cuidadores de que o usuário está

necessitando ou sentindo. Dessa forma, o agrupamento dessas informações associadas fazem uso da sensibilidade ao contexto, tornando-se relevante para definição do estado atual do usuário. A Figura 27 exhibe as telas para as funcionalidades descritas.

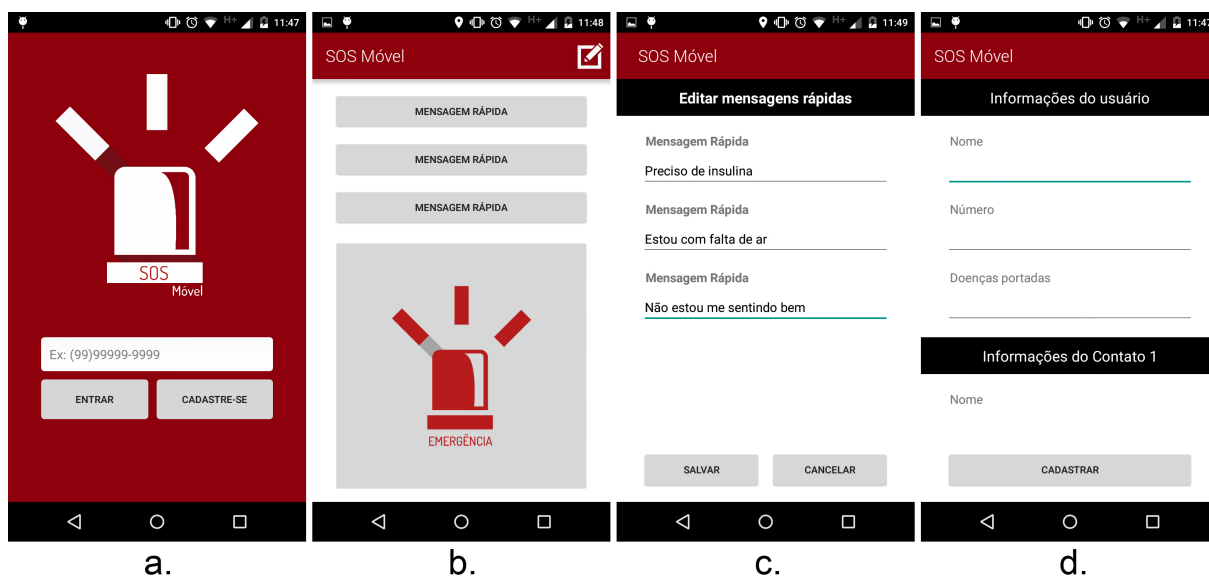


Figura 27 – Telas principais do SOS Móvel

a. Tela de login

b. Tela inicial para solicitar socorro manual

c. Edição das mensagens rápidas

d. Criação de cadastro

O monitoramento do usuário se dá pela interface web (Figura 28) onde é possível ter informações sensíveis ao contexto da localidade do usuário em tempo real, seu histórico de solicitações de socorro e também receber novos alertas. Para utilizar o SOS Móvel web, é preciso que o familiar ou responsável use o número de telefone do usuário como seu login e seu próprio telefone como a senha. Desse modo, apenas os números salvos no cadastro do usuário podem utilizar o sistema.

Ao iniciar a aplicação, após a realização do cadastro e do login, inicia-se a identificação da atividade e a aplicação permanece executando em modo de serviço. Mesmo que o usuário feche a aplicação, o serviço continuará ativo, realizando a detecção de situações de risco. Para identificação da atividade, é usada a API do Google de reconhecimento de atividade (*Activity Recognition*).

A API é uma interface que permite detectar em qual atividade o usuário está envolvido sem o incômodo de obter dados brutos de sensores individuais e, em seguida, ter de executar uma análise complexa para chegar a uma conclusão. A API retorna a atividade detectada em conjunto com a confiança de seus resultados. Como resultado, tem-se uma lista de atividades que um usuário pode estar executando em um determinado momento. As atividades são classificadas segundo a atividade, a mais

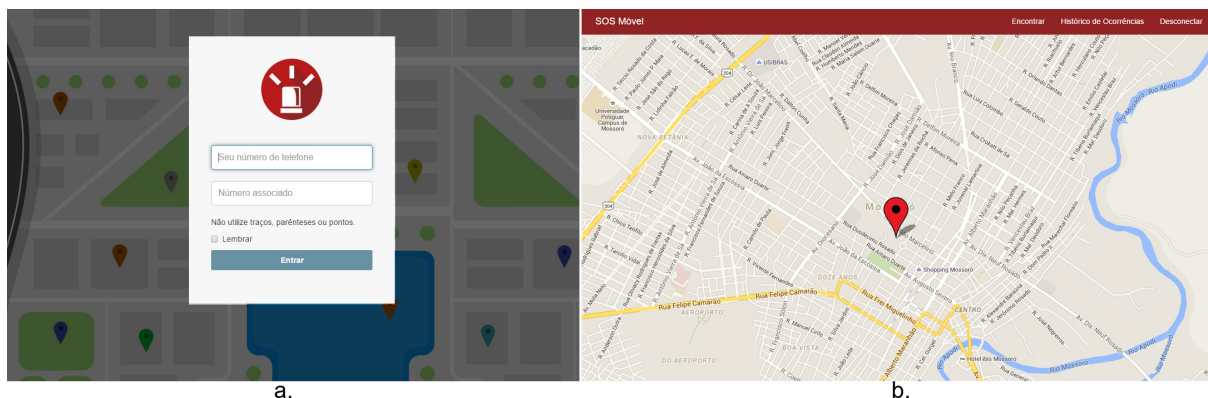


Figura 28 – Interface SOS Web
 a. Tela de login web
 b. Identificação do usuário web

provável em primeiro lugar. A confiança está associada a cada atividade, que indica a probabilidade de qual delas está sendo realizada.

Como mencionado anteriormente, a solicitação de socorro pode ser realizada manualmente e de forma automática. A primeira delas, a manual, é realizada ao pressionar o botão de emergência que se encontra na tela principal. Foi acrescentada uma confirmação de envio para diminuir possíveis erros. A solicitação é enviada para o servidor e exibida como alerta na interface web, junto com a localização, o horário e a data.

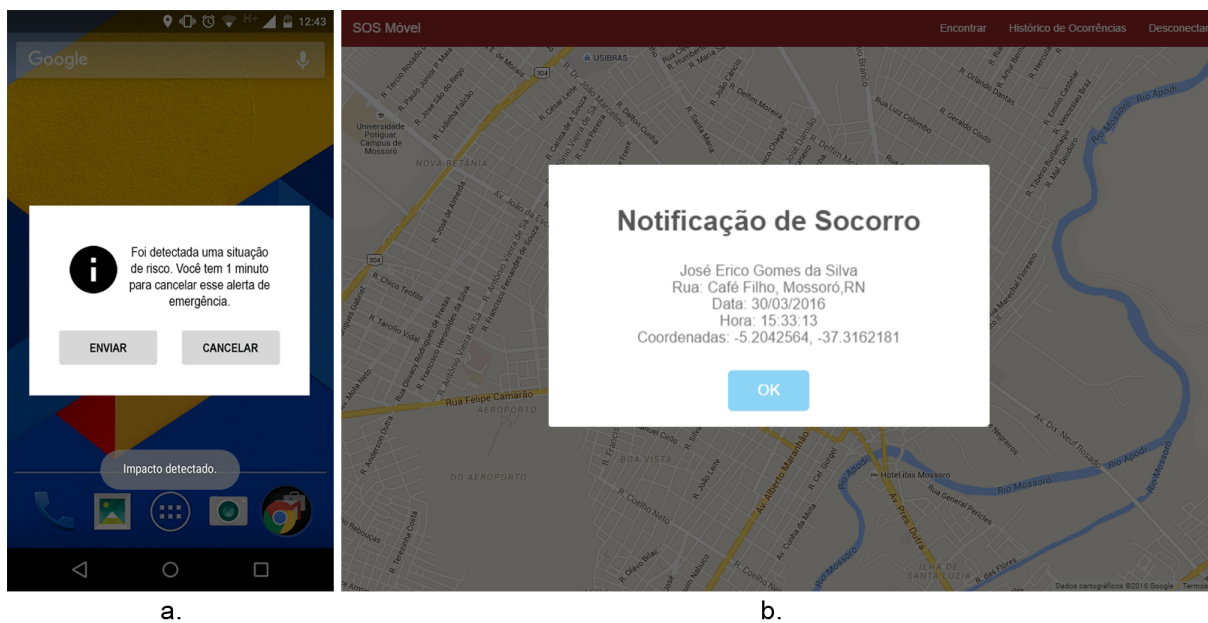


Figura 29 – Notificação de socorro
 a. Identificação de situação de risco b. Notificação recebida na interface web

Na forma automática, a aplicação móvel permanece em execução e, ao serem diagnosticados valores excedentes ao limite de uma atividade para os eixos do acelerômetro, também é enviado para o servidor o alerta para ser exibida na interface web com os dados adicionais. Nesse modo, não é necessária a interação do usuário para realizar a solicitação. Também foi adicionado um alerta com duração de 1 minuto ao modo automático, para o caso de anulação de solicitação se for detectada uma situação de risco equivocada (Figura 29).

4.3.2 Ferramentas e tecnologias

Java é uma linguagem de programação orientada a objeto que foi desenvolvida por James Gosling na década de 90. A mesma é compilada para um *bytecode* executado por uma máquina virtual. Nativamente, sistemas na linguagem Java possuem um grande conjunto de APIs que estão contidas no *Java Runtime Environment* (JRE) (FERNANDES, 2012). Baseado na plataforma Java com o Kernel Linux, o *Android* é um SO móvel gratuito e aberto, com maior parte dos códigos liberada sob a licença *Apache* (LEE, 2012).

O SOS Móvel foi elaborado para executar no sistema operacional *Android*, versão 4.0.3 (*Ice Cream Sandwich*) ou versões superiores, por compreender atualmente 87,9% dos usuários que utilizam o sistema operacional (Android, 2016). De acordo com Cardoso e Barros (2014), o Android possui mais de 1 bilhão de aparelhos ativos, estando presente em 78% das vendas de dispositivos móveis. No Brasil, o sistema operacional representa 91% dos aparelhos.

No desenvolvimento do SOS Móvel foi utilizado a IDE *Android Studio*, devido prover um ambiente de desenvolvimento, *debug*, testes e perfil multiplataforma para o sistema operacional *Android*. O *Android Studio* também possui a funcionalidade de exportar arquivos para *Android Application Package File* (APK), que é o arquivo compilado para distribuição e instalação da aplicação. Em conjunto com *Android Studio* é disponibilizado o SDK do *Android* (Android, 2016).

Para armazenamento de dados foi utilizado o *SQLite*, que é uma biblioteca em linguagem C que implementa um banco de dados com Linguagem de Consulta Estruturada, do inglês *Structured Query Language* (SQL) (*SQLite*, 2014); e o *MySQL*, que é um sistema de gestão de bases de dados relacionais que utiliza a linguagem SQL como interface. (SUEHRING, 2002).

Na interface web foi utilizado o *JavaServer Pages* (JSP), uma linguagem utilizada para criação de páginas *web* geradas dinamicamente baseadas em na Linguagem de Marcação de Hipertexto, do inglês *HyperText Markup Language* (HTML), XML entre outros. O JSP permite construir aplicações que acessam o banco de dados, manipulam arquivos no formato texto, capturam informações a partir de formulários, captam informações sobre o servidor, etc (Caelum, 2016). Para o desenvolvimento dos JSPs foi

utilizado a IDE NetBeans versão 8.0.2, que é um ambiente de desenvolvimento integrado de código aberto que possuem suporte às principais linguagens de programação.

5 ESTUDO DE CASO: APLICAÇÃO DO SOS MÓVEL EM AMBIENTE DE TESTES

Este capítulo é destinado à apresentação do estudo de caso. Nele está exposta a pesquisa, a análise e a interpretação dos dados coletados, além dos resultados quantitativos obtidos. As sessões detalham o uso do SOS Móvel em um ambiente de simulação. Dessa forma, para melhor entendimento, este capítulo está subdividido em:

- Descrição do experimento: Expostos os detalhes do objeto de estudo desta dissertação de mestrado, destacando o modo de simulação, o ambiente e como foi realizada a aquisição de dados;
- Avaliação do experimento: Apresentada os teste para validação do SOS Móvel em um ambiente de simulação;
- Discussões e Resultados: Demonstrada a análise e desempenho dos resultados obtidos sob dados capturados nos testes simulados, mediante validação do estudo de caso.

5.1 DESCRIÇÃO DO EXPERIMENTO

Os testes do SOS Móvel desenvolvidos neste trabalho foram realizados em um ambiente aberto e em uma sala do laboratório, com simulação de 6 atividades, que são elas: parado, caminhando, correndo, bicicleta, colisão e capotamento. Foram realizados 20 testes para cada atividade citada, resultando em 120 resultados coletados.

Os critérios destacados nos testes para obtenção de resultados significativos foram as rotações, os eixos do acelerômetro, a atividade que foi detectada pela API *Activity Recognition*, a conclusão de ocorrência sob os dados recebidos, o tempo de envio de alerta e se houve sucesso ou não no resultado coletado. Cada critério descrito corresponde a uma coluna da tabela gerada.

No que diz respeito à rotação, para identificar sua ocorrência, é utilizado o giroscópio com análise de seus dados à parte dos demais sensores. Isso foi necessário, pois se torna irrelevante obter apenas um resultado em um dado momento sem analisar os dados anteriores. Então, para identificar o movimento de rotação, são armazenados, aproximadamente, 200 milissegundos de dados do giroscópio em um vetor e comparado o valor atual com n outros valores anteriores dentro de uma janela de tempo (descrito no algoritmo 1, da subseção 4.2.4). Dessa forma, quando o valor obtido do eixo Y tiver mudança de sinal em comparação com os valores anteriores, é indicada rotação. Vale salientar que o magnetômetro também está sendo utilizado para calcular a posição do celular em conjunto com o giroscópio.

Os eixos do acelerômetro são obtidos de forma diferente do giroscópio devido o mesmo não necessitar de um histórico de variações, visto que o acelerômetro está sendo usado para verificar a aceleração, e, portanto, só é relevante obter o dado maior do que o padrão de normalidade encontrado de 5m/s. Para obter esse padrão, foram realizados alguns pré-testes para obtenção de padrões de regularidade para algumas ações, como também para observar o comportamento dos sensores em um ambiente veicular. Isso se fez necessário para reduzir a taxa de erros que poderia ser gerada pelo SOS Móvel ao detectar uma ação errada. A Figura 30 ilustra os movimentos realizados nos eixos.

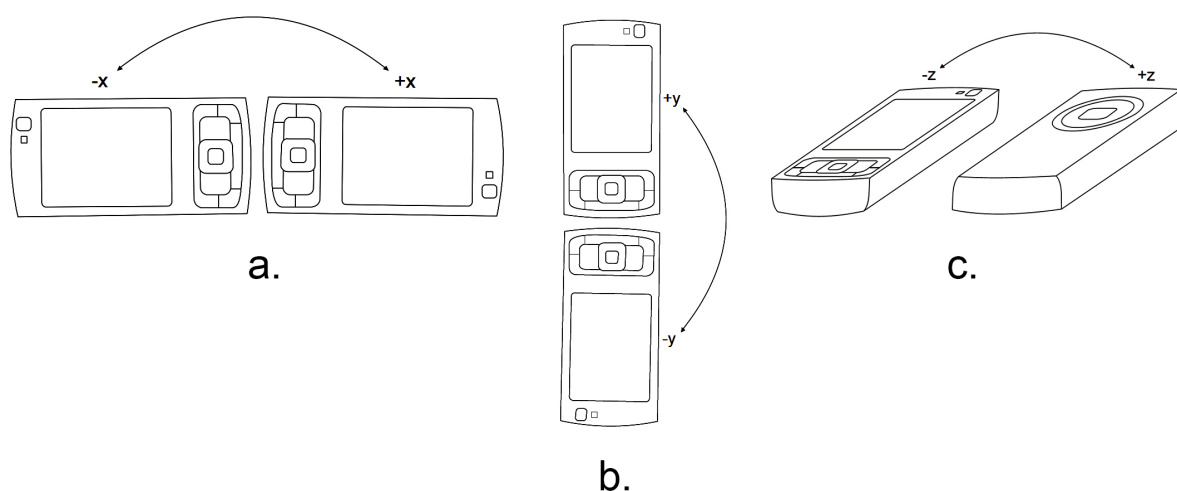


Figura 30 – Representação dos movimentos do acelerômetro nos eixos

a. Eixo x b. Eixo y c. Eixo z

Fonte: Adaptado de Silva e Silva (2011)

O critério de atividade detectada é de responsabilidade da API *Activity Recognition*, utilizada para identificação da ação do usuário. Já o critério de conclusão obtida leva em consideração a análise da rotação, dos eixos do acelerômetro e da atividade detectada pelo usuário. Havendo rotação e valores excedentes para o acelerômetro, é identificada uma situação de risco, que se encaixa à atividade do usuário. Por exemplo, se ocorrerem essas variações e a API detectar a atividade “parado”, então a conclusão deduzida é de queda ou desmaio. Ou seja, a conclusão vai variar de acordo com a atividade realizada pelo usuário.

O tempo de envio de alerta é dado em segundos e diz respeito ao intervalo entre o momento de detecção da situação de risco e a de envio de alerta de confirmação para o Smartphone. Como forma de redução de erros dos alertas gerados, é exibida uma mensagem de confirmação da ocorrência. Se o usuário não anular o alerta em 1 minuto, é enviado o alerta aos familiares ou cuidadores e ao monitoramento web, informando possível acidente envolvendo o usuário.

5.2 AVALIAÇÃO DO EXPERIMENTO

Nesta seção serão apresentados os resultados obtidos pelos testes simulados para as 6 atividades identificadas com o SOS Móvel.

5.2.1 Caso 1: Parado

Para a atividade “Parado”, as simulações foram realizadas com o *Smartphone* na extremidade do braço humano a partir do pulso e simulando uma queda sob uma camada de ar obtida por compressão, sobre a qual foi apoiado o corpo em desaceleração. A Tabela 4 abaixo expõe os resultados dos testes realizados para a atividade em questão.

Tabela 4 – Dados obtidos estando sem movimento

Nº	Rotação	Eixo X	Eixo Y	Eixo Z	Atividade detectada	Conclusão obtida	Tempo de envio de alerta	Resultado coletado
1	Sim	0,00	10,93	0,01	Parado	Queda/Desmaio	0,79	Acerto
2	Não	6,87	0,06	0,03	Parado	Queda/Desmaio	0,89	Acerto
3	Sim	6,73	0,00	0,03	Parado	Queda/Desmaio	1,17	Acerto
4	Não	6,66	0,01	0,08	Parado	Queda/Desmaio	1,07	Acerto
5	Sim	0,06	11,25	0,06	Parado	Queda/Desmaio	0,61	Acerto
6	Não	11,61	0,07	0,02	Parado	Queda/Desmaio	0,77	Acerto
7	Não	0,04	0,01	11,20	Parado	Queda/Desmaio	0,56	Acerto
8	Não	8,84	0,07	0,08	Parado	Queda/Desmaio	0,63	Acerto
9	Sim	9,31	0,03	0,00	Parado	Queda/Desmaio	0,99	Acerto
10	Não	0,02	6,75	0,08	Parado	Queda/Desmaio	1,11	Acerto
11	Não	6,90	0,08	0,00	Parado	Queda/Desmaio	0,58	Acerto
12	Não	0,09	8,70	0,00	Parado	Queda/Desmaio	0,99	Acerto
13	Não	7,70	0,06	0,06	Parado	Queda/Desmaio	1,07	Acerto
14	Sim	0,06	0,03	6,73	Parado	Queda/Desmaio	0,47	Acerto
15	Não	0,10	7,34	0,07	Parado	Queda/Desmaio	1,18	Acerto
16	Não	8,65	0,02	0,03	Parado	Queda/Desmaio	1,16	Acerto
17	Sim	8,09	0,03	0,00	Parado	Queda/Desmaio	1,29	Acerto
18	Sim	7,82	0,08	0,07	Parado	Queda/Desmaio	0,48	Acerto
19	Não	11,39	0,02	0,04	Parado	Queda/Desmaio	0,84	Acerto
20	Não	10,91	0,08	0,09	Parado	Queda/Desmaio	0,62	Acerto

Durante a simulação para essa categoria, dos 7 testes com rotação positiva, 2 obtiveram rotação no acelerômetro no eixo y, ou seja, captada variação acima do limite do movimento na vertical do Smartphone. 4 testes com variação no eixo x e apenas 1 no eixo z. Os 14 demais testes, aqueles com rotação negativa, 10 tiveram rotação no eixo x (movimento na horizontal), 3 no eixo y e 1 no eixo z (movimento para cima e para baixo). Tais movimentos foram demonstrados na Figura 31.

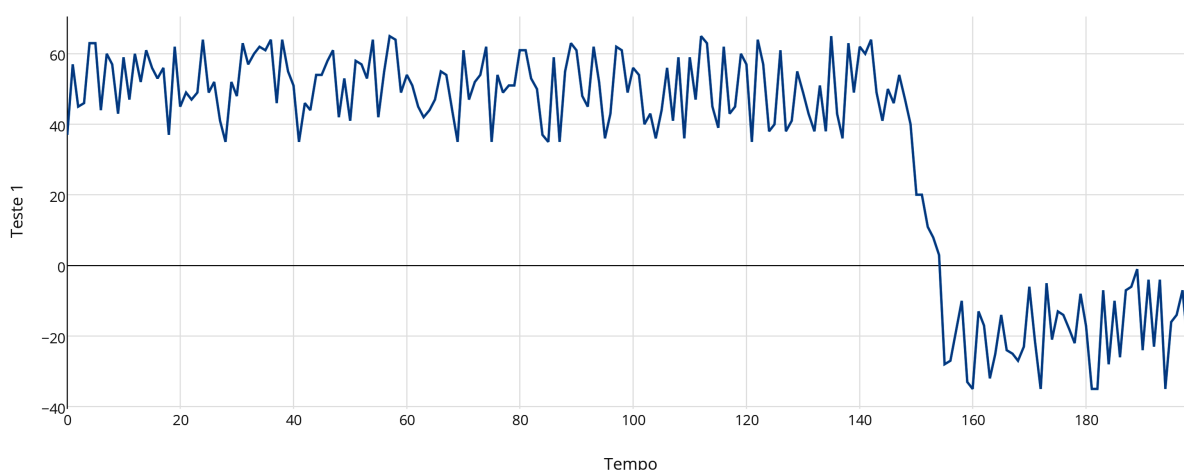


Figura 31 – Gráfico de rotação para o teste 1 da atividade Parado

Fonte: Autoria Própria

Para cada teste realizado, foi gerado separadamente um gráfico que exibe os dados do giroscópio para o critério de rotação, como explicado anteriormente na Seção 5.1. Na Figura 31 são exibidos os dados referentes aos 200 valores armazenados para o Teste 1 da Tabela 4. Pode-se perceber que houve mudança de sinal do eixo do giroscópio, caracterizando assim a presença de rotação no Smartphone. Vale salientar que todos os resultados onde foram identificados movimentos giratórios obtiveram dados com valores semelhantes de comportamento.

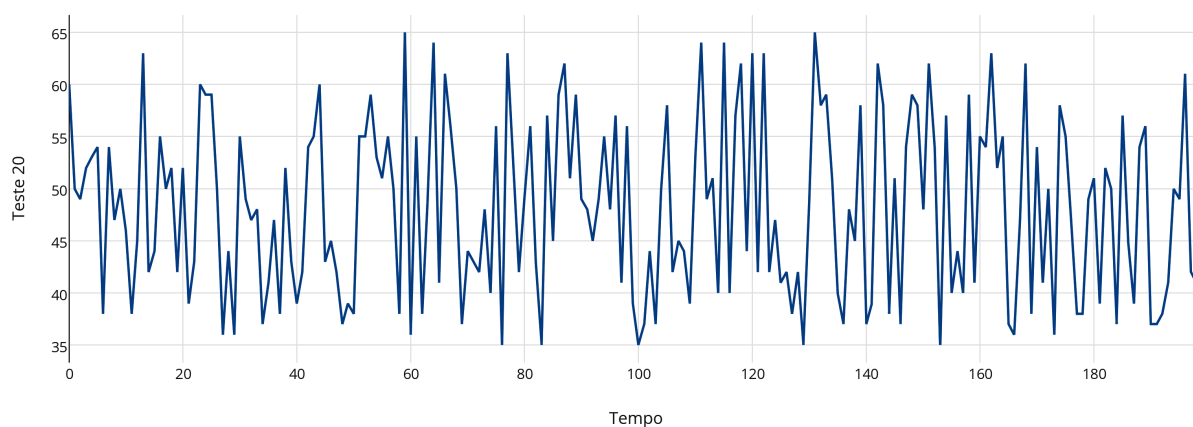


Figura 32 – Gráfico de rotação para o teste 20 da atividade Parado

Fonte: Autoria Própria

Para os casos onde não ocorreu rotação, os gráficos permanecem com os dados em um mesmo quadrante, isto é, sem presença de variação de sinal. Nos testes coletados, há valores aproximadamente entre -40 e 70 para o giroscópio nessa atividade, porém,

não são valores fixos ou absolutos. A Figura 32 exibe as alterações do sensor para o Teste 20.

Fazendo uma comparação entre as duas Figuras apresentadas, pode-se perceber que, para os dois casos, tanto o que houve rotação, quanto este sem rotação, ambos não precisaram ter um valor fixo limite para identificação do movimento giratório, e sim uma análise do comportamento do sensor ao longo do tempo de uso.

Para todos os testes, a API *Activity Recognition* identificou a atividade “Parado”, sendo eficiente e obtendo resultado satisfatório para os testes nessa categoria. A média de tempo para o envio dos alertas para esta atividade foi de 0,865 segundos.

5.2.2 Caso 2: Caminhando

Assim como a atividade “Parado”, os testes para a atividade “Caminhando” foram também realizados com o *Smartphone* na extremidade do braço humano a partir do pulso e simulando uma caminhada simples seguida de uma diminuição da intensidade declivando sob a superfície flexível usada para apoio do corpo em desaceleração. A Tabela 4 expõe os resultados dos testes realizados para a atividade em questão.

Em 3 testes a API *Activity Recognition* detectou equivocadamente atividades como “Correndo” e “Parado”, porém estes casos em específico não afetaram a conclusão obtida, visto que mesmo não estando realizando as ações identificadas erroneamente, com os dados de rotação e do acelerômetro o usuário pode ter sofrido uma queda ou um desmaio e precisa de ajuda. Entretanto, para um caso em específico não foi reconhecida nenhuma ação pela API, o que neste caso caracterizou um erro no teste devido a não detecção da atividade e conseqüentemente o impacto não identificado. Nas três atividades detectadas erroneamente havia presença de movimento rotacionário.

O resultado coletado obteve taxa de acerto satisfatória, salvo o teste onde não foi identificada a atividade. A média de tempo para o envio dos alertas para esta atividade foi de 0,775 segundos. Os Testes realizados para a atividade “Caminhando” estão representados na Tabela 5.

A maioria dos testes obteve variação no eixo y do acelerômetro e grande parte com rotação. O comportamento dos sensores em relação aos valores obtidos foi similar aos dos testes realizados na atividade “Parado”.

Todos os testes envolvendo a rotação do *Smartphone* obtiveram os dados com similaridade e com momento de detecção semelhantes também, aproximadamente um segundo e meio. A diferença entre eles são em alguns valores entre o máximo e mínimo dos testes. Na Figura 33 é apresentado o comportamento do giroscópio em conjunto com o magnetômetro exibindo a representação de todos os testes com rotação para a atividade “Caminhando”.

Tabela 5 – Dados obtidos estando caminhando

Nº	Rotação	Eixo X	Eixo Y	Eixo Z	Atividade detectada	Conclusão obtida	Tempo de envio de alerta	Resultado coletado
1	Sim	0,17	0,09	10,00	Caminhando	Queda	1,1	Acerto
2	Não	0,13	10,92	0,10	Caminhando	Queda	0,59	Acerto
3	Não	0,08	9,05	0,31	Caminhando	Queda	0,68	Acerto
4	Sim	0,12	0,35	8,53	Correndo	Queda	0,56	Acerto
5	Sim	0,50	0,16	7,33	Caminhando	Queda	0,46	Acerto
6	Sim	0,15	9,48	0,09	Caminhando	Queda	0,73	Acerto
7	Não	11,09	0,06	0,13	Caminhando	Queda	1,36	Acerto
8	Sim	0,07	11,51	0,08	Caminhando	Queda	0,75	Acerto
9	Não	0,15	8,23	0,08	Caminhando	Queda	0,9	Acerto
10	Sim	0,11	0,10	7,42	Parado	Queda	0,92	Acerto
11	Não	0,12	9,13	0,12	Caminhando	Queda	0,8	Acerto
12	Sim	0,49	0,07	9,63	Caminhando	Queda	0,51	Acerto
13	Sim	0,15	11,08	0,15	Caminhando	Queda	0,56	Acerto
14	Sim	8,92	0,19	0,09	Atividade não identificada	Impacto não identificado	0,98	Erro
15	Sim	0,10	0,06	10,75	Caminhando	Queda	1,4	Acerto
16	Não	0,05	7,98	0,14	Caminhando	Queda	1,42	Acerto
17	Não	0,16	8,50	0,05	Parado	Queda	0,44	Acerto
18	Sim	0,18	7,47	0,14	Caminhando	Queda	0,43	Acerto
19	Sim	0,13	10,62	0,04	Caminhando	Queda	0,94	Acerto
20	Sim	0,15	7,35	0,15	Caminhando	Queda	1,44	Acerto

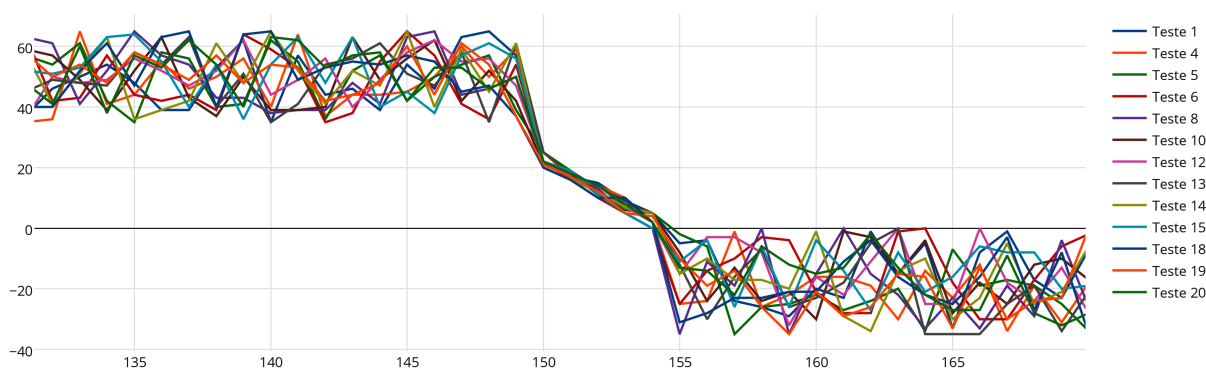


Figura 33 – Gráfico com rotação positiva para todos os testes da atividade Caminhando

Fonte: Autoria Própria

5.2.3 Caso 3: Correndo

As simulações realizadas para a atividade “Correndo” assemelham-se as demais citadas acima, uma vez que não é necessário ter meios de deslocamento além da própria movimentação humana para realização dos testes. A Tabela 4 abaixo expõe os resultados

dos testes realizados para a atividade em questão.

Tabela 6 – Dados obtidos estando correndo

Nº	Rotação	Eixo X	Eixo Y	Eixo Z	Atividade detectada	Conclusão obtida	Tempo de envio de alerta	Resultado coletado
1	Sim	0,36	11,29	0,17	Correndo	Queda	0,5	Acerto
2	Sim	0,30	0,16	10,81	Correndo	Queda	0,42	Acerto
3	Sim	10,25	0,27	7,51	Caminhando	Queda	1,43	Acerto
4	Não	0,24	0,24	9,30	Correndo	Queda	1,01	Acerto
5	Não	0,45	9,48	0,11	Correndo	Queda	0,63	Acerto
6	Sim	0,27	0,19	11,57	Correndo	Queda	0,53	Acerto
7	Sim	7,46	0,14	0,19	Correndo	Queda	1,37	Acerto
8	Sim	0,12	0,32	9,49	Correndo	Queda	1,1	Acerto
9	Sim	7,98	0,13	9,69	Correndo	Queda	0,72	Acerto
10	Sim	0,19	11,52	0,28	Caminhando	Queda	1,13	Acerto
11	Sim	11,45	14,22	10,40	Parado	Queda	1,12	Acerto
12	Sim	0,47	7,97	0,25	Correndo	Queda	1,33	Acerto
13	Não	0,17	0,29	10,67	Correndo	Queda	1,37	Acerto
14	Sim	10,54	0,23	0,28	Correndo	Queda	0,58	Acerto
15	Sim	8,49	11,52	0,22	Caminhando	Queda	1,03	Acerto
16	Sim	0,16	0,17	7,69	Correndo	Queda	1,04	Acerto
17	Sim	0,24	10,34	8,97	Correndo	Queda	0,68	Acerto
18	Não	7,44	0,10	9,87	Correndo	Queda	1,31	Acerto
19	Sim	0,26	0,25	7,45	Correndo	Queda	1,36	Acerto
20	Sim	11,63	0,27	8,59	Correndo	Queda	0,42	Acerto

Para esta atividade em questão, não houveram erros detectados, uma vez que a API conseguiu identificar as atividades corretamente na maioria dos casos, exceto em 4 testes, alternando os resultados entre “Caminhando” e “Parado”. Dos 20 testes, 16 foram realizados com rotação no *Smartphone* e 4 sem movimento giratório.

O tempo médio de envio de alerta para esta atividade foi de 1,035 segundos e o teste com maior espera foi um dos que não foram detectados corretamente pela API. Não houve nenhum erro no resultado coletado e na conclusão obtida pelo SOS Móvel, apesar da API não ter identificado todas as ações corretamente. O desempenho testado para essa atividade foi considerado satisfatório, sem evidência de erros.

A Figura 34 apresenta a diferença entre os dados obtidos com o *Smartphone* em movimento giratório e sem rotação. A principal diferença entre eles se encontra na ocorrência de troca de sinal indicando a rotação; e a permanência no mesmo quadrante, indicando sem circulação do *Smartphone*.

Para o caso sem rotação (teste 4), como não houve mudança de sinal, foi preciso analisar o eixo do acelerômetro em que teve maior variação, no caso o eixo y. A oscilação caracterizou movimento flexível para cima e para baixo, similar a de uma corrida. O valor

mais significativo do eixo y foi de 9,30m/s. Esse valor foi superior ao de normalidade encontrado, o que levou a conclusão da ocorrência de uma queda ou desmaio.

Diferentemente para o caso com rotação (teste 3), com a mudança de sinal pode-se comprovar com mais evidência uma alteração na normalidade da atividade realizada pelo usuário. Além disso, dois eixos do acelerômetro excederam o valor limite de habitualidade, o eixo x e o z, podendo caracterizar um choque com o chão.

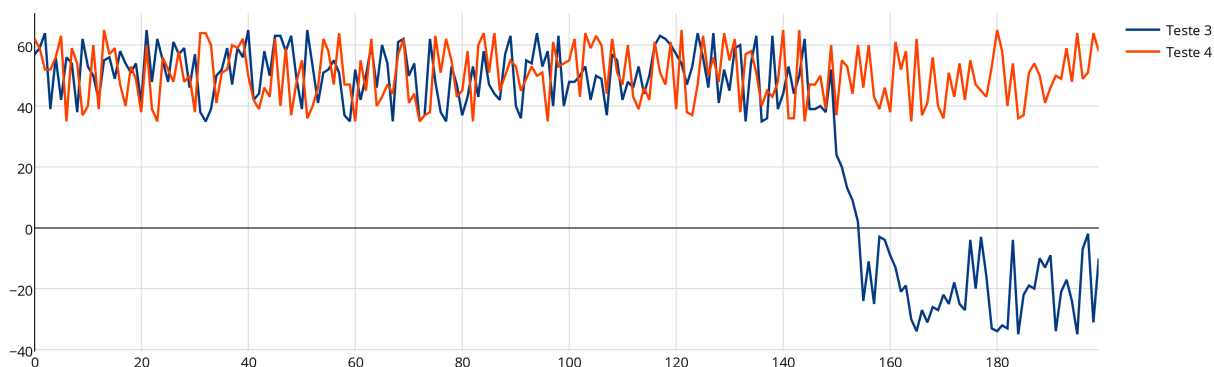


Figura 34 – Gráfico representando o teste 3 e 4 de rotação para Correndo

Fonte: Autoria Própria

5.2.4 Caso 4: Bicicleta

Para a atividade “Bicicleta”, os testes foram realizados utilizando o meio de transporte em questão e em baixa velocidade, simulando um impacto seguido de apoio do dispositivo móvel sob uma região do corpo. O resultado dos testes pode ser visualizado na Tabela 7.

Como foi utilizado um meio externo para os testes, os valores para o acelerômetro tiveram números mais elevados, e, conseqüentemente, acima dos limites estabelecidos para normalidade. Entretanto, foram captados apenas 4 erros referentes a não identificação da atividade e o reconhecimento errado da ação do usuário, ambos por parte da interpretação e funcionamento da API.

Para os testes 7 e 12 no qual foi detectada a atividade “Carro” invés de “Bicicleta”, a variação no acelerômetro foi apresentada em apenas um dos eixos e pouca atividade nos demais eixos. O teste 7 foi simulado com rotação e o valor significativo para o acelerômetro foi de 8,23m/s obtido apenas no eixo x. Já o teste 12, também com movimento rotatório, obteve 6,75m/s de variação máxima apenas no eixo y. A identificação da ocorrência fora da normalidade, nesse caso, foi correta. Porém, a conclusão para situação envolvida foi errada.

A Figura 35 apresenta os testes citados anteriormente no momento em que é detectada a rotação. Como se pode perceber, antes da mudança de sinal, o teste 12

Tabela 7 – Dados obtidos da bicicleta

Nº	Rotação	Eixo X	Eixo Y	Eixo Z	Atividade detectada	Conclusão obtida	Tempo de envio de alerta	Resultado coletado
1	Sim	0,75	7,59	0,25	Bicicleta	Queda de bicicleta	1,05	Acerto
2	Sim	7,45	0,22	0,22	Bicicleta	Queda de bicicleta	0,79	Acerto
3	Sim	0,27	0,52	11,87	Bicicleta	Queda de bicicleta	1,09	Acerto
4	Sim	11,36	0,31	8,23	Atividade não identificada	Impacto não identificado	0,44	Erro
5	Não	0,42	7,41	0,58	Bicicleta	Queda de bicicleta	0,54	Acerto
6	Sim	0,35	0,65	7,35	Bicicleta	Queda de bicicleta	0,52	Acerto
7	Sim	8,23	0,20	0,42	Carro	Capotamento do carro	0,47	Erro
8	Sim	0,22	0,42	8,33	Bicicleta	Queda de bicicleta	0,53	Acerto
9	Sim	13,86	0,28	11,94	Correndo	Queda de bicicleta	1,19	Acerto
10	Sim	0,21	10,43	0,26	Bicicleta	Queda de bicicleta	0,8	Acerto
11	Sim	13,41	0,27	10,51	Bicicleta	Queda de bicicleta	0,73	Acerto
12	Sim	0,67	6,75	0,70	Carro	Capotamento do carro	0,88	Erro
13	Sim	0,18	0,70	7,01	Bicicleta	Queda de bicicleta	0,83	Acerto
14	Sim	8,86	0,68	0,23	Bicicleta	Queda de bicicleta	0,83	Acerto
15	Sim	0,17	7,42	0,68	Bicicleta	Queda de bicicleta	1,47	Acerto
16	Sim	0,66	0,27	11,62	Bicicleta	Queda de bicicleta	1,43	Acerto
17	Não	0,60	0,77	10,50	Bicicleta	Queda de bicicleta	0,93	Acerto
18	Não	0,63	0,55	7,57	Carro	Batida de carro	1,49	Erro
19	Sim	12,29	0,15	0,22	Bicicleta	Queda de bicicleta	1,07	Acerto
20	Sim	0,75	0,45	7,35	Bicicleta	Queda de bicicleta	1,2	Acerto

apresentava maior atividade no sensor do que o teste 7. Aproximando-se do momento da rotação, o teste 7 apresentou maior variação na atividade rotacionário e permaneceu também após a rotação.

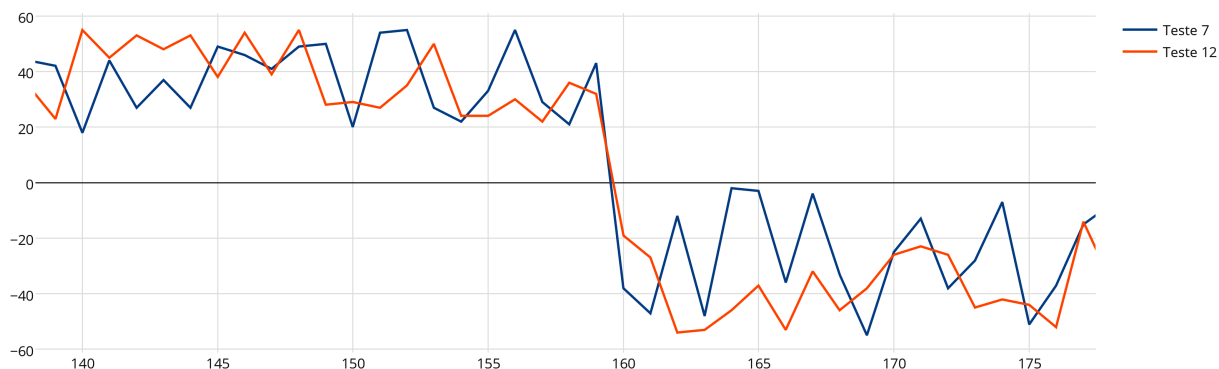


Figura 35 – Gráfico representando o teste 18 sem rotação da atividade Bicicleta

Fonte: Autoria Própria

O tempo médio de envio de alertas para esta atividade foi de 0,855 segundos. E o desempenho testado para essa atividade foi considerado satisfatório, porém com a presença de erros pertinentes de troca de atividade e não identificação correta da mesma.

5.2.5 Caso 5: Carro

Como destacado na problemática deste trabalho, os acidentes de transporte resultam em 26,2% de acidentes mais frequentes no país. Desta forma, decidiu-se utilizar a identificação de situações de risco para duas vertentes principais ligadas ao ambiente veicular. São elas a colisão e o capotamento. A primeira delas diz respeito ao impacto entre dois automóveis frontalmente, enquanto que o capotamento trata-se de dar a volta sobre seu próprio eixo uma ou mais vezes.

Para ambos os casos, tanto a colisão quanto o capotamento, os testes foram realizados em um ambiente automotivo com simulações que viessem a se aproximar da ação proposta.

5.2.5.1 Caso: Colisão

Para captação dos testes, o veículo encontrava-se em movimento e era impulsionado o *Smartphone* para frente na horizontal, similar a um impacto sem rotação de uma colisão frontal real. A frenagem foi simulada amortecendo o *Smartphone* nas mãos para desaceleração rápida.

Vale salientar que para essa atividade específica houve certa cautela para realização das simulações, pois para sucesso dos testes não deveria haver rotação no *Smartphone*, visto que é simulada uma colisão frontal sem movimentos giratórios.

A Tabela 8 mostra os resultados obtidos para a atividade “Carro” com a ocorrência de uma colisão. Como pode-se observar, foram obtidos 16 resultados acertivos e apenas 4 erros. Nos testes 7 e 10 a API identificou corretamente a atividade do usuário, porém a aplicação falhou ao não distinguir corretamente a colisão do capotamento. O erro se deu ao fato do giroscópio ter identificado movimento rotatório quando na verdade teria sido apenas uma frenagem. O gráfico da Figura 36 mostra o momento da troca de sinais captada pelo giroscópio.

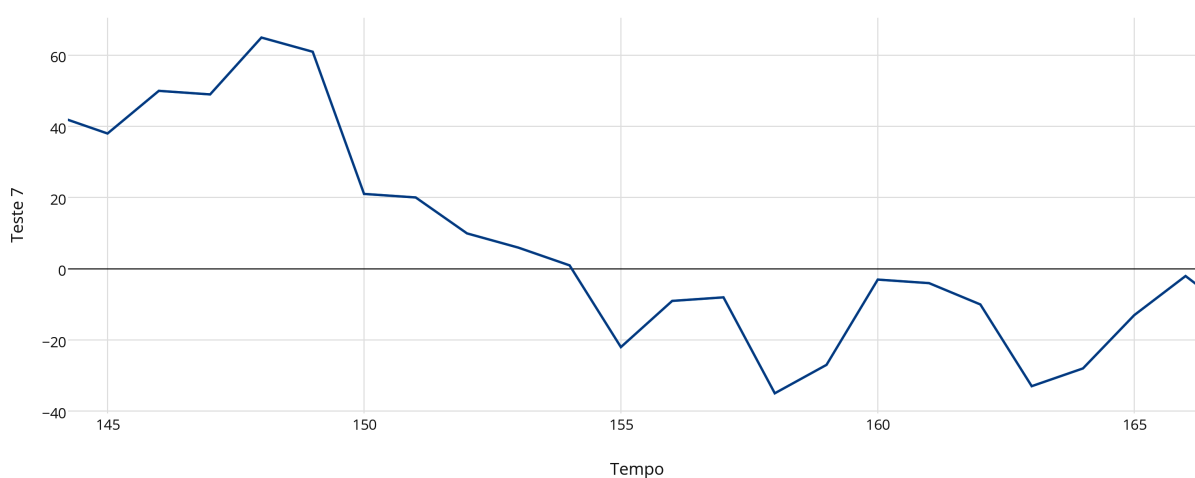


Figura 36 – Gráfico representando o teste 7 para atividade Carro

Fonte: Autoria Própria

As simulações para esta atividade apresentaram equívocos da API em dois casos onde foi captada uma atividade diferente da executada nos testes. Tal imprecisão resultou em dois erros fatais nos testes 7 e 10, onde foi detectado rotação e conseqüentemente uma atividade errada por conseguinte. Os demais erros foram decorrentes da identificação atividade do usuário.

O tempo médio de envio de alertas para esta atividade foi de 0,84 segundos. O desempenho para essa atividade foi considerado satisfatório mediante a quantidade de acertos obtidos em virtude dos erros existentes.

5.2.5.2 Caso: Capotamento

A captação dos testes para capotamento também se deu com o veículo em movimento e com o dispositivo móvel posicionado nas mãos. A simulação se deu com o impulsionamento do *Smartphone* para frente com rotação. A frenagem também foi simulada fazendo o dispositivo móvel perder a intensidade da ação nas mãos.

Tabela 8 – Dados obtidos estando no caso para o caso de colisão

Nº	Rotação	Eixo X	Eixo Y	Eixo Z	Atividade detectada	Conclusão obtida	Tempo de envio de alerta	Resultado coletado
1	Não	15,22	12,65	0,21	Carro	Batida de carro	1,36	Acerto
2	Não	1,58	14,08	14,04	Carro	Batida de carro	1,06	Acerto
3	Não	14,47	1,28	11,99	Carro	Batida de carro	0,53	Acerto
4	Não	1,19	0,78	14,89	Carro	Batida de carro	0,78	Acerto
5	Não	0,67	11,67	0,63	Carro	Batida de carro	0,49	Acerto
6	Não	0,61	1,46	13,11	Carro	Batida de carro	0,76	Acerto
7	Sim	6,57	14,31	0,82	Carro	Capotamento do carro	1,41	Erro
8	Não	1,73	13,95	10,41	Carro	Batida de carro	1,33	Acerto
9	Não	1,42	1,06	10,49	Carro	Batida de carro	0,51	Acerto
10	Sim	0,29	11,25	11,14	Carro	Capotamento do carro	0,98	Erro
11	Não	1,15	1,50	7,70	Carro	Batida de carro	1,42	Acerto
12	Não	0,68	12,83	9,36	Correndo	Queda	0,58	Erro
13	Não	6,56	1,24	6,60	Carro	Batida de carro	0,9	Acerto
14	Não	12,67	0,60	0,17	Carro	Batida de carro	1,27	Acerto
15	Não	1,50	12,52	1,08	Carro	Batida de carro	0,58	Acerto
16	Sim	14,25	0,81	14,93	Correndo	Queda	0,67	Erro
17	Não	1,09	1,03	10,42	Carro	Batida de carro	0,54	Acerto
18	Não	11,29	1,71	13,05	Carro	Batida de carro	0,95	Acerto
19	Não	1,02	12,01	14,68	Carro	Batida de carro	0,56	Acerto
20	Não	12,94	1,01	10,15	Carro	Batida de carro	1,42	Acerto

O resultado dos testes pode ser visualizado na Tabela 9. Para a atividade de identificação de capotamento os resultados constataram uma ocorrência maior de erros em relação as demais atividades, resultando em 5 erros e 15 acertos. Com a simulação

pode-se notar que em apenas 1 caso houve identificação equivocada da rotação do dispositivo móvel, o que ocasionou o erro do resultado coletado. Outro erro identificado foi na detecção da atividade do usuário, onde a API falhou 4 vezes por classificar erroneamente estas ações.

Nos testes 1 e 19 o acelerômetro obteve valores acima do normal estabelecido nos três eixos simultaneamente, caracterizando um movimento de volta sobre seu próprio eixo. Os demais testes obtiveram na maioria variação em dois eixos ou em apenas um. A média de tempo para envio de alertas para esta atividade foi de 0,72 segundos. A Figura 37 exibe o gráfico de rotação para o teste 1 e a Figura 38 para o teste 3.

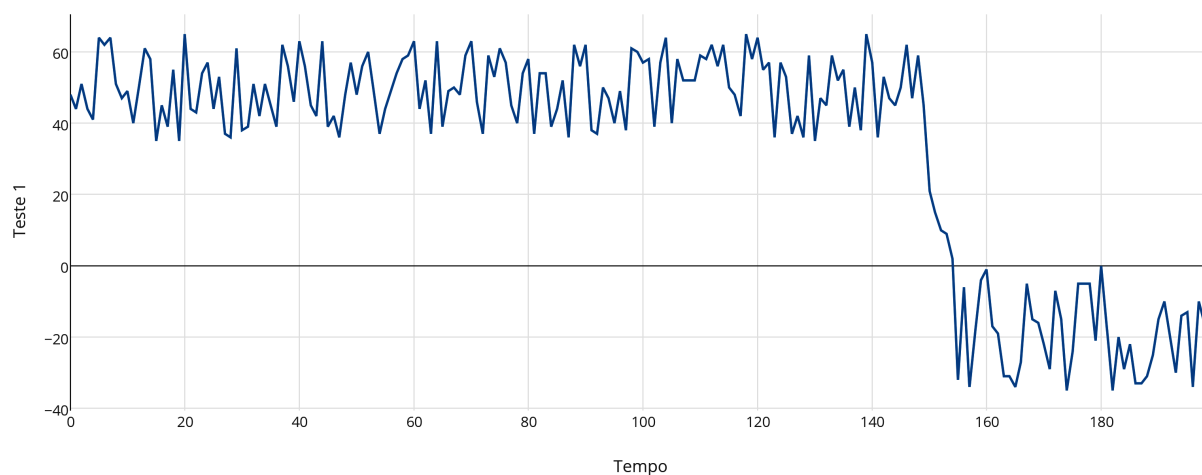


Figura 37 – Gráfico representando o teste 1 para o caso de capotagem

Fonte: Autoria Própria

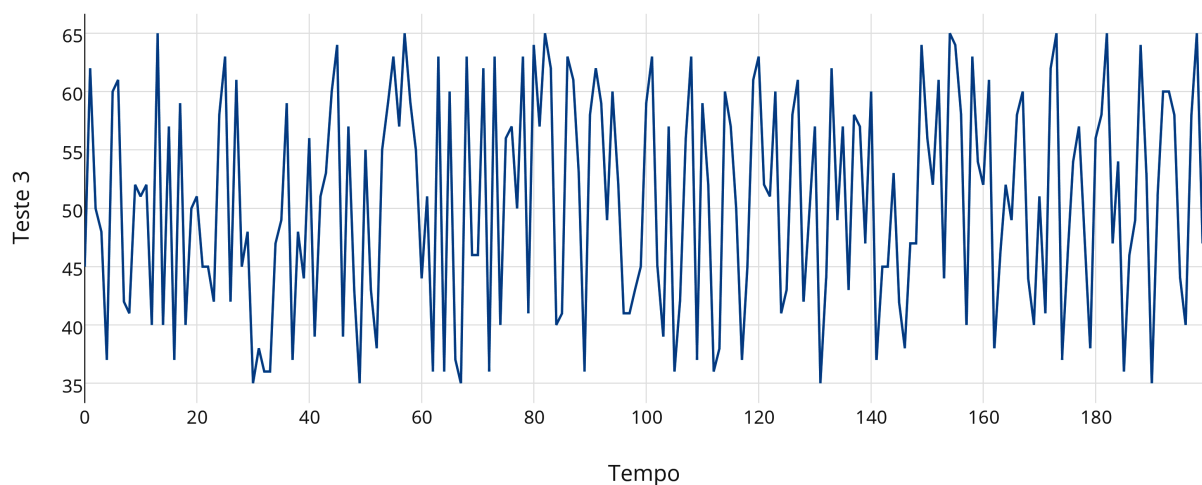


Figura 38 – Gráfico representando o teste 3 para o caso de capotagem

Fonte: Autoria Própria

Tabela 9 – Dados obtidos estando no caso para o caso de capotamento

Nº	Rotação	Eixo X	Eixo Y	Eixo Z	Atividade detectada	Conclusão obtida	Tempo de envio de alerta	Resultado coletado
1	Sim	16,33	12,66	10,29	Carro	Capotamento do carro	0,83	Acerto
2	Sim	0,76	8,24	14,55	Carro	Capotamento do carro	1,02	Acerto
3	Não	14,04	0,65	8,28	Carro	Batida do carro	0,61	Acerto
4	Sim	1,21	1,19	14,44	Correndo	Queda	1,16	Erro
5	Sim	1,20	7,41	1,34	Carro	Capotamento do carro	0,78	Acerto
6	Sim	0,74	1,17	12,19	Carro	Capotamento do carro	0,71	Acerto
7	Sim	9,61	13,61	1,46	Carro	Capotamento do carro	0,53	Acerto
8	Sim	1,55	12,94	13,19	Carro	Capotamento do carro	1,36	Acerto
9	Sim	15,27	1,21	2,65	Carro	Capotamento do carro	0,73	Acerto
10	Sim	0,22	13,86	0,30	Atividade não identificada	Impacto não identificado	0,55	Erro
11	Sim	16,29	0,68	0,26	Carro	Capotamento do carro	0,78	Acerto
12	Não	1,09	11,42	0,71	Carro	Batida do carro	0,5	Erro
13	Sim	6,56	0,75	14,61	Carro	Capotamento do carro	0,51	Acerto
14	Sim	14,55	1,58	0,20	Carro	Capotamento do carro	0,93	Acerto
15	Sim	1,46	8,70	0,59	Bicicleta	Queda de bicicleta	0,46	Erro
16	Sim	1,77	1,17	9,53	Carro	Capotamento do carro	1,11	Acerto
17	Sim	10,50	1,17	7,12	Bicicleta	Queda de bicicleta	0,56	Erro
18	Sim	0,77	10,88	8,99	Carro	Capotamento do carro	0,58	Acerto
19	Sim	10,85	11,30	13,07	Carro	Capotamento do carro	1,36	Acerto
20	Sim	11,35	0,56	13,83	Carro	Capotamento do carro	0,41	Acerto

5.3 DISCUSSÕES E RESULTADOS

A partir dos testes simulados, foi possível concluir que o SOS Móvel obteve bom desempenho às funções a este encarregadas, em que foram simulados 6 casos de utilização do sistema para identificação de situações de risco. Para cada caso isolado, foram realizados 20 testes, resultando no total de 120 testes para a detecção e a tomada de decisão automática.

A taxa de acertos para a atividade a “Parado” foi de 100%, já a atividade “Caminhando” atingiu 95% de sucesso, “Correndo” resultou também em 100% de acerto, “Bicicleta” e “Colisão” obtiveram 80% e “Capotamento” resultou 75% de acertos, como apresentado na Figura 39.

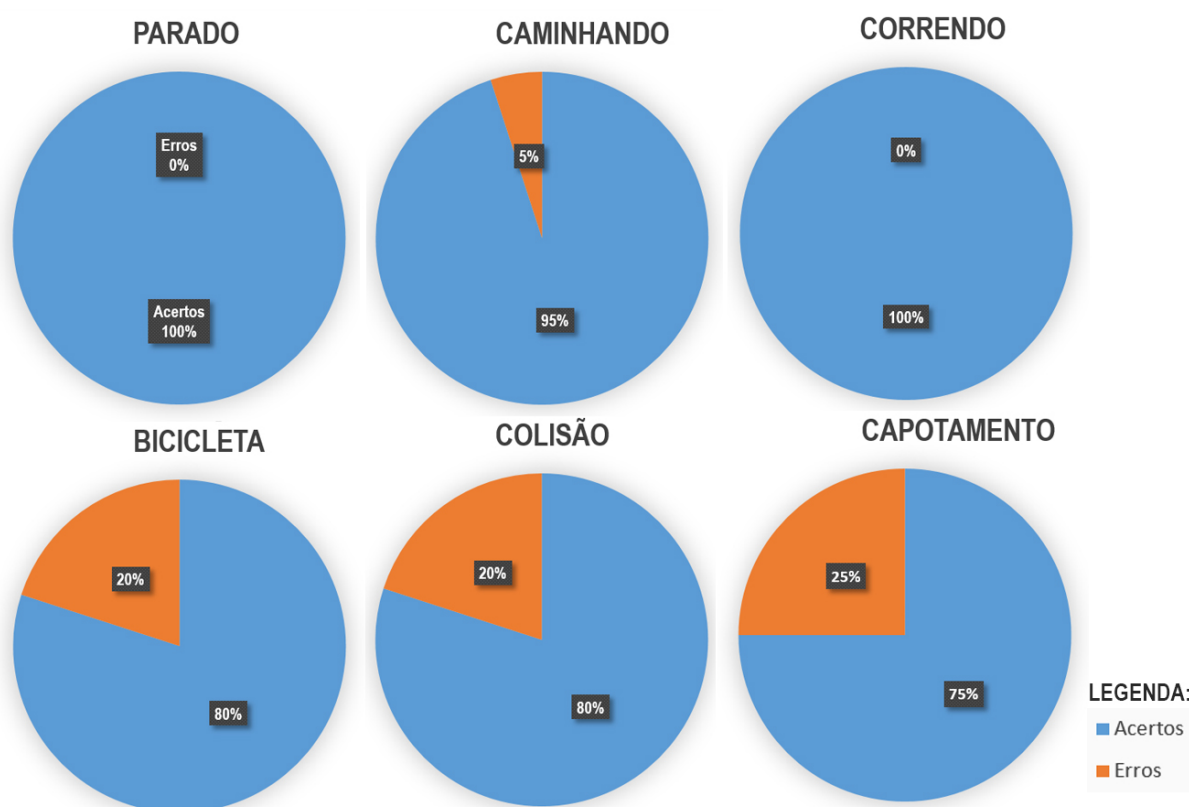


Figura 39 – Gráficos de porcentagem de erros e acertos para cada atividade

Fonte: Autoria Própria

Das 6 atividades captadas e simuladas, as que obtiveram 100% de aproveitamento e acerto nos testes foram a de “Parado” e de “Correndo”. As demais obtiveram pelo menos 1 erro em um dos testes. A atividade “Caminhando” possuiu apenas 1 erro e as atividades “Bicicleta” e “Colisão” tiveram 4 erros cada. As simulações que obtiveram maior erros foi a de “Capotagem”, com 5 erros.

Esses dados comprovam que a sensibilidade ao contexto provida pelo SOS Móvel auxiliou na captação dos dados temporais do usuário e a identificação do que está sendo

feito por ele.

6 CONSIDERAÇÕES FINAIS

Vista a necessidade de implementação de um método para reduzir e agilizar o atendimento médico de urgência, o SOS móvel foi desenvolvido com intuito de agir sob situações em que envolve o usuário em acidentes e incidentes. O objetivo principal do SOS Móvel é o desenvolvimento de um sistema para auxiliar no pedido de socorro e detecção automática de situações de risco.

O sistema proposto foi desenvolvido em um trabalho contínuo de pesquisa, planejamento e implementação. Ao fim do desenvolvimento e dos testes por meio de simulações, o sistema obteve bom desempenho na identificação e na solicitação de socorro. Os testes foram realizados com simulação de atividades corriqueiras do usuário, com acréscimo de possíveis incidentes. Vale salientar que o SOS Móvel possui todas as funcionalidades descritas no Capítulo 4 ativas e em perfeito funcionamento.

Os testes do SOS Móvel foram realizados em um ambiente aberto e em uma sala do laboratório, com utilização do Smartphone Moto X, de modelo XT1058 e com sistema operacional Android versão 5.1 Lollipop. Além disso, foi utilizada uma camada de ar obtida por compressão para amortecimento de impactos e desaceleração do Smartphone para as atividades de “parado”, “caminhando” e “correndo”.

As simulações para a atividade “bicicleta” foram realizados com o meio de transporte em movimento acrescido do colchão para neutralizar a concussão. Para colisão e capotagem foi utilizado um automóvel também em movimento e simuladas as atividades com o movimento do *smartphone* dentro do carro.

As situações de risco estabelecidas para as atividades foram basicamente quatro. São elas queda, desmaio, colisão e capotamento, onde as duas primeiras situações são resultados de risco estabelecidos para as atividades “parado”, “caminhando”, “correndo” e “bicicleta”; e as outras duas situações são para as atividades de “colisão” e “capotamento”.

Foram realizados 20 testes para cada atividade citada anteriormente, resultando em 120 resultados coletados. Destes, 14 resultaram em erros decorrentes da detecção errada da atividade do usuário por meio da API, enquanto 106 decorreram em acertos. Dessa forma, pode-se concluir que, com os testes realizados, o sistema apresentou bom desempenho e é adequado para o monitoramento remoto de atividades de autocuidados em usuários com alguma doença crítica, com potencial para aplicação em diversos casos.

Vale salientar também que, apesar de o modelo proposto atender às necessidades primárias de envio e de recebimento de alertas emergenciais, o mesmo pode ser utilizado por uma unidade médica de atendimento para auxílio à população. Visto que possui uma interface web de monitoramento e recebimento de alertas.

O SOS Móvel foi desenvolvido para Smartphones com o sistema operacional

Android, que possuam a versão 4.0.3 (*Ice Cream Sandwich*) ou versões superiores.

Fundamentando-se nessa premissa, como limitações para o uso do SOS Móvel, tem-se que a instalação é possível apenas em dispositivos Android, excluindo, assim, dispositivos com sistemas operacionais que não funcionem baseados nesse sistema. Além disso, é necessária a conexão com a internet para envio e recebimento de alertas.

O SOS Móvel para Web foi desenvolvido baseado em plataformas web, funcionando, dessa maneira, em qualquer navegador de qualquer dispositivo com acesso à internet, excluindo, portanto, outras limitações. Como não exige um hardware ou sistema operacional específico para usar o monitoramento, o SOS Móvel para Web foi pensado para uso dos profissionais da saúde, cuidadores ou familiares.

A utilização do SOS Móvel e o web possibilita rapidez e facilidade para informar acidentes ou situações de risco a familiares ou cuidadores por meio de seu menu de mensagens rápidas, além da solicitação de socorro manual e também automática. Também auxilia no processo de identificação da situação em que o usuário se encontra por meio da sensibilidade ao contexto e o seu monitoramento.

A partir deste trabalho surgem possibilidades de trabalhos futuros, como:

- A adição de identificação de novas atividades, e, conseqüentemente situações de risco aliadas a estas;
- Implantação do SOS Móvel e *web* a um sistema de monitoramento de serviço de urgência e atendimento médico de ambiente real;
- Agregar a utilização de mais sensores ao sistema para maior precisão dos dados obtidos;
- Validação das funcionalidades do sistema com usuários portadores de alguma enfermidade crônica ou doença específica;
- Desenvolver técnicas inteligentes para identificação e adaptação da rotina do usuário sem necessidade de intervenção do mesmo.

REFERÊNCIAS

- ABOWD, G. D.; MYNATT, E. D. Charting past, present, and future research in ubiquitous computing. *ACM Transactions on Computer-Human Interaction (TOCHI)*, ACM, v. 7, n. 1, p. 29–58, 2000.
- ADIBI, S. Mobile health (mhealth) emergency-based vehicular system. In: *Electrical and Computer Engineering (CCECE), 2014 IEEE 27th Canadian Conference on*. [S.l.: s.n.], 2014. p. 1–7. ISSN 0840-7789.
- ADOMAVICIUS, G.; TUZHILIN, A. Context-aware recommender systems. In: *Recommender systems handbook*. [S.l.]: Springer, 2011. p. 217–253.
- AKASH, S. I. *Segurança Familiar - MONBAN*. 2013. Disponível em: <<https://goo.gl/o1NB5i>>.
- ALONSO, E.; D'INVERNO, M.; KUDENKO, D.; LUCK, M.; NOBLE, J. Learning in multi-agent systems. *The Knowledge Engineering Review*, Cambridge Univ Press, v. 16, n. 03, p. 277–284, 2001.
- AMARANTE, T. C. et al. Detecção automática e alerta de acidentes de trânsito em redes veiculares reais. Programa de Pós-Graduação em Ciência da Computação, 2015.
- Android. *Site Oficial Android*. 2016. Disponível em: <https://www.android.com/intl/pt-BR_br/history/>.
- Apple Developer. *Features for Expanded Ad Units*. 2015. Disponível em: <<https://goo.gl/kaRvpT>>.
- ARAÚJO, R. B. de. Computação ubíqua: Princípios, tecnologias e desafios. In: *XXI Simpósio Brasileiro de Redes de Computadores*. [S.l.: s.n.], 2003. v. 8, p. 11–13.
- ARTERO, A. *Inteligência artificial teórica e prática*. Editora Libreria da Física, 2008. ISBN 9788578610296. Disponível em: <<https://books.google.com.br/books?id=NOGuQQAACAAJ>>.
- BAPTISTA, F. J. *Telemedicina em Catástrofe*. Dissertação (Mestrado) — Universidade do Porto, 2010. Disponível em: <<http://goo.gl/e4lkbj>>.
- BARROS, N. M. *Um sistema de auxílio ao diagnóstico de perda auditiva em neonatos*. Dissertação (Mestrado) — Universidade do Estado do Rio Grande do Norte / Universidade Federal Rural do Semi-Árido, 2015.
- BERSCH, R. Tecnologia assistiva. 2013. Disponível em: <<http://goo.gl/WqYFqE>>.
- BESALEVA, L. I.; WEAVER, A. C. Crowdhelpt: M-health application for emergency response improvement through crowdsourced and sensor-detected information. In: *Wireless Telecommunications Symposium (WTS)*. [S.l.: s.n.], 2014. p. 1–5.
- BEVILACQUA, V.; NUZZOLESE, N.; BARONE, D.; PANTALEO, M.; SUMA, M.; D'AMBRUOSO, D.; VOLPE, A.; LOCONSOLE, C.; STROPPIA, F. Fall detection in indoor environment with kinect sensor. In: IEEE. *Innovations in Intelligent Systems and*

- Applications (INISTA) Proceedings, 2014 IEEE International Symposium on*. [S.l.], 2014. p. 319–324.
- BLAYA, A.; JOAQUIN FRASER, S.; FERNANDES HOLT, B.; BRIAN. *E-health technologies show promise in developing countries*. [S.l.]: Health Affairs, 2010.
- BRUSTOLONI, J. C. *Autonomous agents: characterization and requirements*. [S.l.]: Citeseer, 1991.
- BRYANT, D.; P. BRYANT, B.; R. *Assistive technology for people with disabilities*. [S.l.]: Allyn and Bacon, 2003.
- Caelum. *Java para desenvolvimento web*. 2016. Disponível em: <<https://www.caelum.com.br/apostila-java-web/>>.
- CALADO, M. P. *Serviço de emergência médica angolano: otimização utilizando sistemas multi-agente*. Tese (Doutorado) — UNIVERSIDADE DE LISBOA, 2015.
- CARDOSO, D. S.; BARROS, K. A. A. de. Usabilidade dos sistemas operacionais móveis android, ios e windows phone. *Anais dos Simpósios de Informática do IFNMG-Câmpus Januária*, 2014.
- CARVALHO, S. T. de; COPETTI, A.; FILHO, O. G. L. Sistema de computação ubíqua na assistência domiciliar à saúde. *Journal Of Health Informatics*, v. 3, n. 2, 2011.
- CHAGAS, J. F. S. *Sistema Multiagente De Apoio À Gestão De Conhecimento Em Projetos De Software Que Implementam Cmmi-Nível 2*. Dissertação (Mestrado) — Universidade do Estado do Rio Grande do Norte / Universidade Federal Rural do Semi-Árido, 2013.
- CHEN, G.; KOTZ, D. et al. *A survey of context-aware mobile computing research*. [S.l.], 2000.
- CHEN, H.; FININ, T.; JOSHI, A. *Umbc ebiquity project: Context broker architecture*. 2005.
- COEN, M. H. *Sodabot: A software agent environment and construction system*. 1994.
- COOK, A.; POLGAR, J. *Cook & Hussey's Assistive Technologies: Principles and Practice*. Mosby Elsevier, 2008. ISBN 9780323039079. Disponível em: <<http://goo.gl/ABRKe7>>.
- COPPIN, B. *Artificial Intelligence Illuminated*. Jones and Bartlett Publishers, 2004. (Jones and Bartlett illuminated series). ISBN 9780763732301. Disponível em: <<http://goo.gl/xEQ9Cz>>.
- CUARELLI, G. *Estimação de parâmetros da marcha humana a partir da medições feitas por acelerômetros e giroscópios e uso de mecanismo de referência*. Dissertação (Mestrado) — Instituto Federação de Educação, Ciência e Tecnologia de São Paulo - IFSP, 2013.
- CUNHA, A. *Cavalo Do Século: Por Que Os Acidentes de Transito Acontecem?* [S.l.]: Arte e Ciência, 2008.
- DAS, R.; ALAM, T. Location based emergency medical assistance system using openstreetmap. In: *Informatics, Electronics Vision (ICIEV), 2014 International Conference on*. [S.l.: s.n.], 2014. p. 1–5.

- DEY, A. K. Understanding and using context. *Personal and ubiquitous computing*, Springer-Verlag, v. 5, n. 1, p. 4–7, 2001.
- DEY, A. K. Context-aware computing. *Ubiquitous Computing Fundamentals*, Chapman and Hall/CRC Boca Raton, FL, USA, p. 321–352, 2010.
- DEY, A. K.; ABOWD, G. D. Towards a better understanding of context and context-awareness. *Workshop on the What, Who, Where, When, and How of Context-Awareness*, 2000.
- DUARTE, F. J. A. Classificação de atividades físicas através do uso do acelerômetro do smartphone. 2013.
- ENDLER, M. Aplicações para computação ubíqua. 2001.
- Faculdade de Medicina da Universidade do Porto. *Informática Médica*. 2014. Disponível em: <<http://goo.gl/AxU4aB>>.
- FAVA, P. *Tecnologia Móvel Para a Saúde: Práticas Fundamentais para Implementadores de RRC*. 2014. Disponível em: <<http://goo.gl/gI2VFw>>.
- FERBER, J.; GUTKNECHT, O. A meta-model for the analysis and design of organizations in multi-agent systems. In: IEEE. *Multi Agent Systems, 1998. Proceedings. International Conference on*. [S.l.], 1998. p. 128–135.
- FERNANDES, M. P. *Simplicidade e qualidade utilizando APIs Java*. 2012. Disponível em: <<http://goo.gl/Mbw1K5>>.
- FRANKLIN, S.; GRAESSER, A.; OLDE, B.; SONG, H.; NEGATU, A. Virtual mattie-an intelligent clerical agent. In: *AAAI Symposium on Embodied Cognition and Action*, Cambridge MA. [S.l.: s.n.], 1996.
- FRIGO, L. B.; POZZEBON, E.; BITTENCOURT, G. O papel dos agentes inteligentes nos sistemas tutores inteligentes. In: *World Congress on Engineering and Technology Education*. [S.l.: s.n.], 2004. p. 86.
- GEMALTO. *Watch the mHealth case study 'Assisted Living'*. 2012. Disponível em: <<http://goo.gl/qsUIMl>>.
- GEMIN, W. A. *Panic Button Selfie*. 2015. Disponível em: <<https://play.google.com/store/apps/details?id=panic.ButtonSelfie.demo>>.
- GOGOI, D.; SHARMA, R. K. Android based emergency alert button. *International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN*, p. 2278–3075, 2013.
- GOMES, A.; GUANAES MORAES, J. B. M.; XAVIER TIMERMAN, S.; SERGIO. *Telemedicina e sua aplicação em emergências*. 2011. Disponível em: <<http://goo.gl/1HBVaR>>.
- GONZALEZ, A.; BRÉZILLON, P. Integrating two context-based formalisms for improved representation of human tactical behavior. *The Knowledge Engineering Review*, v. 23, n. 2, p. 1–21, 2008.

GOODRICH, R. Accelerometer vs. gyroscope: What's the difference? *LiveScience Contributor*, 2015.

Grameen Foundation. *Health*. 2014. Disponível em: <<http://goo.gl/gSyTZy>>.

GREVE, J. *Medicina de reabilitação aplicada à ortopedia e traumatologia*. [S.l.]: Roca, 1999.

HADJINICOLAOU, M.; NILAVALAN, R.; ITAGAKI, T.; VOSKARIDES, S.; PATTICHIS, C.; SCHIZAS, A. Emergency teleorthopaedics m-health system for wireless communication links. *Communications, IET*, v. 3, n. 8, p. 1284–1296, August 2009. ISSN 1751-8628.

HANSMANN, U.; MERK, L.; NICKLOUS, M. S.; STOBER, T. *Pervasive computing handbook*. [S.l.]: Springer Science & Business Media, 2013.

HAYES-ROTH, B. An architecture for adaptive intelligent systems. *Artificial Intelligence*, Elsevier, v. 72, n. 1, p. 329–365, 1995.

HENDERSON-SELLERS, B. *Agent-oriented methodologies*. [S.l.]: IGI Global, 2005.

II, M. M. *MAS-Commonkads+: Uma Extensão à Metodologia MAS-Commonkads para Suporte ao Projeto Detalhado de Sistemas Multiagentes Racionais*. Tese (Doutorado) — Master Thesis, Universidade Estadual do Ceará, Fortaleza, Brazil, 2010.

ITO, G. C.; FERREIRA, M.; SANT'ANA, N. *Computação móvel: Aspectos de gerenciamento de dados*. INPE, 2003.

iTunes. *Virtual Check Up*. 2014. Disponível em: <<http://goo.gl/ph066D>>.

iXtentia. *SOS - Stay Safe!* 2015. Disponível em: <<https://play.google.com/store/apps/details?id=com.extentia.sos>>.

JADHAV, R.; PATEL, J.; JAIN, D.; PHADHTARE, S. Emergency management system using android application. Citeseer, 2014.

KIM, H. tae; SONG, B. Vehicle recognition based on radar and vision sensor fusion for automatic emergency braking. In: *Control, Automation and Systems (ICCAS), 2013 13th International Conference on*. [S.l.: s.n.], 2013. p. 1342–1346. ISSN 2093-7121.

KING, T. *Assistive technology: essential human factors*. [S.l.]: Allyn e Bacon, Incorporated, 1999.

Kionix. *Accelerometers*. 2015. Disponível em: <<http://goo.gl/v8Sz5p>>.

KUMAR, M.; KUMAR, R. Iprob - emergency application for women. *International Journal of Scientific and Research Publications (IJSRP)*, v. 4, 2014.

LEE, W.-M. *Beginning android 4 application Development*. [S.l.]: John Wiley & Sons, 2012.

LOPES, J.; AFONSO, F.; PERNAS, A.; PALAZZO, L.; YAMIN, A. Exehda-on: Uma abordagem baseada em ontologias para sensibilidade ao contexto na computação pervasiva. 2008.

- LOUREIRO, A. A. F.; OLIVEIRA, R. A. R.; SILVA, T.; JÚNIOR, W. R. P.; OLIVEIRA, L. d.; MOREIRA, R.; SIQUEIRA, R.; ROCHA, B.; RUIZ, L. Computação ubíqua ciente de contexto: Desafios e tendências. *Simpósio Brasileiro De Redes De Computadores E Sistemas Distribuídos, Recife*, v. 27, n. 99, p. 149, 2009.
- LOURO, S. F. G.; OLIVEIRA, E. da C.; REIS, L. P.; PORTO, U. do. *MAICC–Sistema Multi-Agente para Controlo de Câmaras Inteligentes*. [S.l.: s.n.], 2004.
- LYYTINEN, K.; YOO, Y. Ubiquitous computing. *Communications of the ACM*, v. 45, n. 12, p. 63–96, 2002.
- MAES, P. Intelligent software: Easing the burdens that computers put on people. *IEEE Expert, IEEE*, v. 11, n. 6, p. 62–63, 1996.
- MAGALHÃES, D. A. *Vocalizador Digital Baseado na Computação Sensível ao Contexto*. Dissertação (Mestrado) — Universidade do Estado do Rio Grande do Norte / Universidade Federal Rural do Semi-Árido, 2015.
- MANZINI, E.; J. Tecnologia assistiva para educação: recursos pedagógicos adaptados. ensaios pedagógicos: construindo escolas inclusivas. SEESP/MEC, p. 82–86, 2005. Disponível em: <<http://goo.gl/N37wAF>>.
- MATEUS, G. R.; LOUREIRO, A. A. F. *Introdução à computação móvel*. [S.l.]: DCC/IM, COPPE/UFRJ, 1998.
- MEDEIROS, R. A. d. *Desenvolvimento de um sistema inteligente para monitoramento contínuo de pessoas diabéticas para melhoria da qualidade de vida*. Dissertação (Mestrado) — Universidade do Estado do Rio Grande do Norte / Universidade Federal Rural do Semi-Árido, 2015.
- MEDMINDER. *Automatic Pill Dispensers So Your Loved Ones Can Stay Independent*. 2014. Disponível em: <<http://www.medminder.com>>.
- MÖLLER, E. d. S. *Pervasividade no Contexto dos Dispositivos de Borda*. Dissertação (Mestrado) — Universade Católica de Pelotas, 2006.
- MOHAMMEDALI, M.; PHUNG, D.; ADAMS, B.; VENKATESH, S. A context-sensitive device to help people with autism cope with anxiety. In: *ACM. CHI'11 Extended Abstracts on Human Factors in Computing Systems*. [S.l.], 2011. p. 1201–1206.
- MONTEIRO, B. d. S.; GOMES, A. S. O. *Ambiente de aprendizado ubíquo youubi: design e avaliação*. Tese (Doutorado), 2015.
- NORVIG, P.; RUSSELL, S. *Inteligência Artificial, 3ª Edição*. Elsevier Brasil, 2014. ISBN 9788535251418. Disponível em: <<https://books.google.com.br/books?id=BsNeAwAAQBAJ>>.
- OBAIDAT, M. S.; DENKO, M.; WOUNGANG, I. *Pervasive computing and networking*. [S.l.]: John Wiley & Sons, 2011.
- ORTIZ, J. *Smartphone-Based Human Activity Recognition*. Springer International Publishing, 2015. (Springer Theses). ISBN 9783319142746. Disponível em: <<https://books.google.com.br/books?id=Uq0qBgAAQBAJ>>.

- PASCOE, J.; RYAN, N.; MORSE, D. Issues in developing context-aware computing. In: SPRINGER. *Handheld and ubiquitous computing*. [S.l.], 1999. p. 208–221.
- PEREIRA, M. R.; AMORIM, C. L. de; CASTRO, M. C. S. de. Tutorial sobre redes de sensores. *Cadernos do IME-Série Informática*, v. 14, p. 39–53, 2013.
- PINTO, F. S.; CENTENO, J. A. S. A realidade aumentada em smartphones na exploração de informações estatísticas e cartográficas. *Bol. Ciênc. Geod., sec. Artigos*, SciELO Brasil, v. 18, n. 2, p. 282–301, 2012.
- PIVA, L. S.; BRAGA, R. B.; FERREIRA, A. B.; ANDRADE, R. M. de C. falert: Um sistema android para monitoramento de quedas em pessoas com cuidados especiais. 2014. Portal da Saúde. *Pesquisa em Saúde*. 2014. Disponível em: <<http://goo.gl/VfwQX0>>.
- Programa Nacional de Prevenção de Acidentes. Promoção nacional de prevenção de acidentes: Direção de serviços de promoção e proteção da saúde. Ministério da Saúde, 2010. Disponível em: <<http://goo.gl/ZFCO3L>>.
- RAZ, D.; JUHOLA, A. T.; SERRAT-FERNANDEZ, J.; GALIS, A. *Fast and efficient context-aware services*. [S.l.]: John Wiley & Sons, 2006.
- ReBall Software. *EmergentHelper V2*. 2014. Disponível em: <<http://goo.gl/Kvc3iE>>.
- REZENDE, S. *Sistemas inteligentes: fundamentos e aplicações*. Manole, 2005. ISBN 9788520416839. Disponível em: <<http://goo.gl/0V0SRb>>.
- RIFTIN, N. ehealth e mhealth: os caminhos da medicina em um país que está envelhecendo. eWave Medical do Brasil, 2013. Disponível em: <<http://goo.gl/om27rj>>.
- ROBITAILLE, S. *The Illustrated Guide to Assistive Technology and Devices: Tools and Gadgets for Living Independently: Easyread Super Large 20pt Edition*. CreateSpace, 2010. ISBN 9781458764898. Disponível em: <<http://goo.gl/xhxlPD>>.
- ROCHA, A. P. O. F. d. Monitorme: Android-based online activity recognition system. Universidade de Aveiro, 2012.
- ROCHA, C. C. L. da; COSTA, C. A. da; RIGHI, R. da R. Um modelo para monitoramento de sinais vitais do coração baseado em ciência da situação e computação ubíqua. 2015.
- ROUGIER, C.; MEUNIER, J.; ST-ARNAUD, A.; ROUSSEAU, J. Robust video surveillance for fall detection based on human shape deformation. *Circuits and Systems for Video Technology, IEEE Transactions on, IEEE*, v. 21, n. 5, p. 611–622, 2011.
- RUDOWSKY, I. Intelligent agents. *The Communications of the Association for Information Systems*, v. 14, n. 1, p. 48, 2004.
- RUSSELL, S.; NORVIG, P.; INTELLIGENCE, A. A modern approach. *Artificial Intelligence*. Prentice-Hall, Egnlewood Cliffs, Citeseer, v. 25, 1995.
- SACCOL, A. Z.; REINHARD, N. Tecnologias de informação móveis, sem fio e ubíquas: definições, estado-da-arte e oportunidades de pesquisa. *Revista de administração contemporânea*, SciELO Brasil, v. 11, n. 4, p. 175–198, 2007.
- SANTOS, F. C. d. et al. Computação ubíqua para aplicações em saúde. 2012.

- SARAIVA, F. d. O. *Aplicação de sistemas multiagentes para gerenciamento de sistemas de distribuição tipo Smart Grids*. Tese (Doutorado) — Universidade de São Paulo, 2012.
- SCHILIT, B. N. A context-aware system architecture for mobile distributed computing. *Unpublished PhD, Columbia University*, 1995.
- SCHILIT, B. N.; THEIMER, M. M. Disseminating active map information to mobile hosts. *Network, IEEE, IEEE*, v. 8, n. 5, p. 22–32, 1994.
- SCHMIDT, A.; BEIGL, M.; GELLERSEN, H.-W. There is more to context than location. *Computers & Graphics, Elsevier*, v. 23, n. 6, p. 893–901, 1999.
- SCHREIBER, G. *Knowledge engineering and management: the CommonKADS methodology*. [S.l.]: MIT press, 2000.
- Secretaria de Vigilância em Saúde. Viva: Vigilância de violências e acidentes: Sistema de vigilância de violências e acidentes. Ministério da Saúde, 2013. Disponível em: <<http://goo.gl/yL7RGU>>.
- SETHIA, D.; GUPTA, D.; MITTAL, T.; ARORA, U.; SARAN, H. Nfc based secure mobile healthcare system. In: *Communication Systems and Networks (COMSNETS), 2014 Sixth International Conference on*. [S.l.: s.n.], 2014. p. 1–6.
- SHEPHERD, E.; PATONNIER, J.; YOSHINO, K. *Orientation and motion data explained*. 2014. Disponível em: <<https://goo.gl/jEv0ut>>.
- SHINDE, P.; TAWARE, P.; THORAT, S. et al. Emergency panic button. 2012.
- SHUO, C. Fall detection system using arduino fio. 2015.
- SILVA, L. C.; NETO, F. M. M.; JÚNIOR, L. J. Mobile: Um ambiente multiagente de aprendizagem móvel baseado em algoritmo genético para apoiar a aprendizagem ubíqua. *Revista Brasileira de Informática na Educação*, v. 21, n. 01, p. 62, 2013.
- SILVA, R. M. d. A percepção de risco como fator para o evento acidente de trabalho. 2015.
- SILVA, R. V.; SILVA, F. A. Utilização de dispositivos móveis com acelerômetro para controle de aplicações. In: *Colloquium Exactarum*. [S.l.: s.n.], 2011. v. 2, n. 1, p. 12–20.
- SMITH, D. C.; CYPHER, A.; SPOHRER, J. Kidsim: programming agents without a programming language. *Communications of the ACM, ACM*, v. 37, n. 7, p. 54–67, 1994.
- SOSmart. *SOSmart car crash notification*. 2016. Disponível em: <<https://play.google.com/store/apps/details?id=com.sosmart.app>>.
- SOUZA, R. S. *Uma Contribuição à Coordenação na Computação Pervasiva com Aplicações na Área Médica*. Dissertação (Mestrado), 2009.
- SQLite. *About SQLite*. 2014. Disponível em: <<http://goo.gl/l0xi1Y>>.
- SUEHRING, S. *Mysql: a bíblia*. In: *MySQL: a bíblia*. [S.l.]: Campus, 2002.

- THOMPSON, C.; WHITE, J.; DOUGHERTY, B.; ALBRIGHT, A.; SCHMIDT, D. C. Using smartphones to detect car accidents and provide situational awareness to emergency responders. In: *Mobile Wireless Middleware, Operating Systems, and Applications*. [S.l.]: Springer, 2010. p. 29–42.
- TONIN, G. S.; GOLDMAN, A. Tendências em computação móvel. *Departamento de Ciências da Computação do Instituto de Matemática e Estatística da Universidade de São Paulo*, 2012.
- ULANDAY, C. C. A.; MAGPANTAY, E. P.; IGNO, D. A. C.; BARSOLASO, A. J. I.; DISTOR, E. S. Helpshake: Android based helping system for wearable devices. 2015.
- VAIJAYANTI, P.; WANKHADE, N.; DIPIKA, N.; KANCHAN, J.; NEHA, P. Sciwars android app for women safety. *Engineering Research and Applications*, v. 4, 2014.
- VENECIAN, L. R. Um mecanismo de sensibilidade ao contexto com suporte sem^antico para computaç ao ubiqua. 2010.
- VIEIRA, L. P. *Experimentos com o acelerômetro de Tablets e Smartphones*. Tese (Doutorado) — Universidade Federal do Rio de Janeiro. Rio de Janeiro, 2013.
- VOSS, G. B.; NUNES, F. B.; HERPICH, F.; MEDINA, R. D. Ambientes virtuais de aprendizagem e ambientes imersivos: um estudo de caso utilizando tecnologias de computação móvel e web viewers. *Tecnologias, Sociedade e Conhecimento*, v. 2, n. 1, p. 24–42, 2015.
- WANT, R.; HOPPER, A.; FALCAO, V.; GIBBONS, J. The active badge location system. *ACM Transactions on Information Systems (TOIS)*, ACM, v. 10, n. 1, p. 91–102, 1992.
- WARD, A.; JONES, A.; HOPPER, A. A new location technique for the active office. *Personal Communications, IEEE*, IEEE, v. 4, n. 5, p. 42–47, 1997.
- WEISER, M. The computer for the 21st century. *Scientific american*, Nature Publishing Group, v. 265, n. 3, p. 94–104, 1991.
- WOEI, C. S. Emergency beacon for elderly. Universiti Malaysia Pahang, 2002.
- WOOLDRIDGE, M. *An introduction to multiagent systems*. [S.l.]: John Wiley & Sons, 2009.
- WOOLDRIDGE, M.; JENNINGS, N. R. Intelligent agents: Theory and practice. *The knowledge engineering review*, Cambridge Univ Press, v. 10, n. 02, p. 115–152, 1995.
- YAMIN, A. C.; BARBOSA, J. V.; AUGUSTIN, I.; SILVA, L. C. D.; REAL, R.; GEYER, C.; CAVALHEIRO, G. Towards merging context-aware, mobile and grid computing. *International Journal of High Performance Computing Applications*, SAGE Publications, v. 17, n. 2, p. 191–203, 2003.
- YAVUZ, G.; KOCAK, M.; ERGUN, G.; ALEMDAR, H. O.; YALCIN, H.; INCEL, O. D.; ERSOY, C. A smartphone based fall detector with online location support. In: *International Workshop on Sensing for App Phones; Zurich, Switzerland*. [S.l.: s.n.], 2010. p. 31–35.

ZALDIVAR, J.; CALAFATE, C.; CANO, J.; MANZONI, P. Providing accident detection in vehicular networks through obd-ii devices and android-based smartphones. In: *Local Computer Networks (LCN), 2011 IEEE 36th Conference on*. [S.l.: s.n.], 2011. p. 813–819. ISSN 0742-1303.

ZIDAN, A.; EL-SAADANY, E. F. A cooperative multiagent framework for self-healing mechanisms in distribution systems. *Smart Grid, IEEE Transactions on, IEEE*, v. 3, n. 3, p. 1525–1539, 2012.

ZIMMERMANN, A.; LORENZ, A.; OPPERMANN, R. An operational definition of context. In: *Modeling and using context*. [S.l.]: Springer, 2007. p. 558–571.

Apêndices

APÊNDICE A – MODELO DE OBJETOS E RECURSOS DO AGENTE

