

Sistema Móvel de Apoio a Decisão Médica Aplicado ao Diagnóstico de Asma – InteliMED

Dyego Carlos Sales de Moraes¹, Bruno Carlos Sales de Moraes¹, Júlio Venâncio de Menezes Júnior¹, Cristine Martins Gomes de Gusmão^{1,2}

¹Núcleo de Telessaúde – Universidade Federal de Pernambuco (UFPE)
Recife – PE – Brazil

²Departamento de Cirurgia – Centro de Ciências da Saúde – Universidade Federal de Pernambuco (UFPE) Recife – PE – Brazil

{dyego.morais,bruno.morais,julio.menezes,
cristine.gusmao}@nutes.ufpe.br

Abstract. *The public health structure in Brazil has essentially decentralized services, which enables the use of mobile devices. Yet it is clear that there is a gap in the use of these means in primary health care, in addition to making necessary tools to support decision-making based on evidence collected. In this context, this paper presents the experience of building a mobile decision support system to medical diagnosis using computational intelligence. The discussion approaches the simultaneous application of agile methodologies, applicability of mobile devices in health and use of artificial intelligence systems to support medical decision.*

Resumo. *A estrutura de saúde pública possui essencialmente serviços descentralizados, o que viabiliza a utilização de dispositivos móveis. Ainda assim percebe-se que há uma lacuna quanto ao uso desses meios em saúde básica. Além de se fazerem necessárias ferramentas de apoio à tomada de decisão baseada em evidências coletadas. Neste contexto, este artigo apresenta a experiência da construção de um sistema móvel de suporte à decisão para apoio ao diagnóstico médico utilizando inteligência computacional. A discussão aborda aplicação simultânea de diferentes metodologias ágeis, aplicabilidade de dispositivos móveis em saúde e utilização de inteligência artificial em sistemas de apoio a decisão médica.*

1. Introdução

Tecnologias móveis produzem benefícios quando utilizadas para auxílio da atenção básica à saúde. Alguns mais visíveis são: (i) simplificação do acesso e da atualização das informações; (ii) fortalecimento da medicina baseada em evidências através de sistemas de apoio à decisão; e (iii) diminuição de erros e inconsistências [Menezes Jr. et al. 2011].

Os serviços de atenção à saúde primária e básica são essencialmente descentralizados e realizados por profissionais que possuem conhecimento generalizado. Em saúde comunitária, ou mais especificamente na Estratégia de Saúde da Família (ESF), há uma estrutura organizacional que viabiliza a utilização de dispositivos

móveis, pois opera com unidades de saúde básicas distribuídas, nas quais as equipes de profissionais respondem por uma pequena parcela da população e, além disso, realizam visitas domiciliares [Menezes Jr. et al. 2011].

O uso de técnicas de inteligência artificial em sistemas que visam apoiar a decisão médica já é bastante comum, como é possível constatar-se nos trabalhos relacionados [Kulikowski, Weiss 1982; Patil, Szolovits, Schwartz 1982; Shortliffe, Scott, Bischoff 1981; Razzouk, Shirakawa, Mari 2000; Seixas et al. 2011]. Esses sistemas de suporte à decisão podem apoiar a atenção básica à saúde ao contribuir para a diminuição da ocorrência de casos graves e internações hospitalares com elevado custo.

Nesse contexto, este artigo apresenta a experiência de desenvolvimento de um sistema de apoio à decisão descentralizado por meio do uso de dispositivos móveis – InteliMED [Gusmão, 2012; Menezes Jr. et al. 2011]. Deseja-se apresentar a construção de um sistema móvel de suporte remoto sobre diagnósticos médicos iniciais com foco em atenção básica à saúde. Para avaliação dos conceitos envolvidos, foi desenvolvida uma solução para apoio ao diagnóstico clínico de asma devido à importância da doença para o Sistema Único de Saúde e ao fato de requerer diversos atributos e evidências, muitas vezes despercebidos pelo profissional não especialista.

Após esta seção introdutória, o artigo está organizado da seguinte forma: a seção 2 traz a fundamentação teórica necessária ao entendimento da pesquisa; a seção 3 demonstra a solução proposta e suas peculiaridades; a seção 4 apresenta os resultados obtidos; a seção 5, por fim, descreve considerações finais e propostas futuras.

2. Fundamentação Teórica

Esta seção apresenta os principais conceitos utilizados: *Mobile Health*, trabalhos relacionados e novas tecnologias móveis; Asma e sua relevância; uso de Inteligência Artificial para suporte ao diagnóstico médico; e uso simultâneo de diferentes metodologias ágeis de desenvolvimento de software.

2.1. Mobile Health

Mobile Health (ou *mHealth*) é o termo utilizado para a prática de medicina e saúde pública suportada por dispositivos móveis - como *smartphones* e *tablets*. A área de *mHealth* emergiu como um subsegmento de *eHealth*, que consiste no uso de Tecnologia da Informação e Comunicação para auxiliar serviços de saúde [Vital Wave Consulting 2009]. Aplicações *mHealth* podem incluir: o uso de dispositivos móveis na coleta de dados clínicos e de saúde da comunidade; o fornecimento de informações de saúde para os profissionais, pesquisadores e pacientes; e a prestação direta de cuidados via telemedicina móvel [Germanakos, Mourlas, Samaras 2005].

Conforme apresentado por Menezes Jr. e demais autores (2011), em uma análise comparativa entre sistemas móveis para saúde, observou-se que boa parte destes sistemas são disponibilizados para PDA (Personal Digital Assistant) e utilizam tecnologia J2ME (Plataforma Java 2 Micro Edition), a exemplo do HandMED [Castro et al. 2004] e do PDAEmbu [Moraes, Pisa, Lopes 2004].

Contudo, sistemas operacionais para dispositivos móveis vem avançando bastante, a exemplo de Android (sistema operacional para dispositivos portáteis

desenvolvido por Google e empresas parceiras), que, em se tratando apenas de *smartphones*, já é o sistema móvel mais utilizado no mundo [InfoGraphic Labs 2012]. Essa nova plataforma possui como dispositivos principais, *smartphones* e *tablets*. Esses avanços tecnológicos provavelmente serão responsáveis por mudanças no cenário de dispositivos e tecnologias utilizadas para *mHealth*.

2.2. Asma

A asma é uma das doenças crônicas mais comuns em todo o mundo. Segundo a Sociedade Brasileira de Pneumologia e Tisiologia [Sociedade Brasileira de Pneumologia e Tisiologia 2006], no Brasil a asma atinge cerca de 20% de crianças e adolescentes. Outro dado importante é que 2,3% de todas as internações pelo SUS (Sistema Único de Saúde) são por asma. Essas internações custam aproximadamente 76 milhões de reais, o que chega a ser o terceiro maior valor gasto com uma única doença no país.

A atenção básica à saúde desempenha uma importante função no diagnóstico e tratamento corretos de doenças, uma vez que é a principal porta de entrada dos pacientes no SUS através da ESF. O diagnóstico clínico da asma na atenção básica à saúde é importante para iniciar corretamente o tratamento, e conseqüentemente diminuir a ocorrência de casos graves e internações hospitalares com elevado custo.

2.3. Inteligência Computacional

A mineração de dados – principal atividade do processo de descoberta de conhecimento (Knowledge Discovery in Databases - KDD) – consiste em aplicar algoritmos com a finalidade de extrair conhecimento de bases de dados [Fayyad, Piatetsky-Shapiro, Smyth 1996]. Esta descoberta de conhecimento pode ser realizada de diversas formas: agrupamentos, hipóteses, regras de associação, árvores de decisão, redes neurais, dentre outras.

De acordo com Fayyad, Piatetsky-Shapiro e Smyth (1996), pode-se considerar o desafio do InteliMED em sua essência um problema de classificação, uma vez que o objetivo é classificar a partir das entradas (neste caso, perguntas extraídas de *guidelines* relacionados ao diagnóstico de asma) se o paciente possui ou não asma.

Conforme um estudo realizado por Menezes Jr. et al. (2011), a técnica de mineração mais adequada para solução do problema é árvore de decisão, uma vez que esta técnica possui grande aplicabilidade ao domínio do diagnóstico em saúde, devido a sua natureza de apoio a decisão baseada na inferência em dados coletados em paralelo com histórico de casos passados. Além disso, outra vantagem da árvore de decisão é que ela pode ser facilmente validada junto aos usuários do sistema.

As árvores de decisão são modelos estatísticos, cujo principal objetivo é aprender como classificar objetos. A partir de um conjunto de instâncias, cujas classes são conhecidas, a ideia do algoritmo é que após a fase de treinamento ele seja capaz de classificar novas instâncias [Quinlan 1996]. Na Figura 1, pode-se observar um exemplo de árvore de decisão, onde estão ilustrados seus elementos [Ken et al. 2010]:

De acordo com a Figura 1 tem-se que:

1. Nós (elipses) - são os atributos que definem a classificação do objeto (previsão, umidade e vento), e no presente trabalho eles representam as perguntas de um formulário sobre asma;
2. Arestas (setas/testes) - definem os caminhos da árvore (Ex.: se previsão=chuvoso, então seguir para vento), e neste projeto as arestas representam as respostas das perguntas do formulário;
3. Folhas (retângulos) - indica a classificação do objeto, que neste trabalho são restritas a: asma ou não asma.

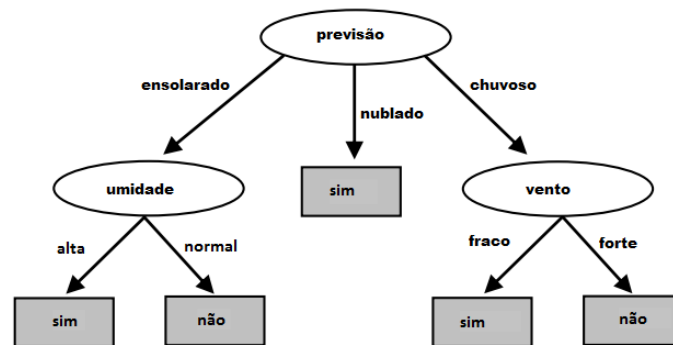


Figura 1. Exemplo de árvore de decisão adaptada [Ken et al. 2010].

2.4. Metodologias Ágeis de Desenvolvimento de Software

A aplicação de metodologias de desenvolvimento de software pesadas ou tradicionais produzem software muitas vezes de qualidade suspeita ou que não cumprem exigências de prazo, custos e escopo [Soares 2004]. Metodologias ágeis se diferem das tradicionais por que possuem o enfoque em pessoas e em software funcionando, além de serem consideradas adaptativas [Beck et al. 2001].

As metodologias ágeis mais conhecidas são o XP (Extreme Programming) [Beck 1999] e o *Scrum* [Schwaber, Beedle 2002]. Elas possuem algumas semelhanças e diferenças que permitem que sejam utilizadas em conjunto, pois se complementam. Enquanto o *Scrum* foca principalmente em planejamento e gerência de projetos, o XP possui enfoque em práticas de desenvolvimento de software.

O *Scrum* atribui papéis aos interessados no projeto, como *stakeholders* (todos os interessados), *Scrum Master* (responsável por garantir o fluxo do processo), *Product Owner* (representante do cliente) e *Scrum Team* (composto pela equipe de desenvolvimento multifuncional e autogerenciável, pelo representante do cliente e pelo *Scrum Master*). O *Scrum* possui alguns artefatos, dentre os quais se destacam: *Burn Down Chart* é um gráfico que representa a quantidade de esforço que falta para a conclusão das tarefas estipuladas para um determinado tempo; *Product Backlog* é uma lista priorizada de requisitos de alto nível; e *Sprint Backlog* é uma lista de funcionalidades a serem feitas na iteração.

O XP sugere a aplicação da técnica de programação em pares, o que acelera o desenvolvimento e diminui a probabilidade de erros, visto que mais de um programador

trabalha em um código ao mesmo tempo. Esta técnica também é útil em casos de evasão, já que o conhecimento acerca dos códigos do sistema está distribuído entre os programadores – propriedade coletiva. Além disso, elaboração de histórias do usuário, refatoração de código, integração contínua, testes e prototipação são práticas comuns em projetos que utilizam essa metodologia.

Apesar das peculiaridades de cada processo, existem também várias práticas que são comuns entre *Scrum* e XP. O *Scrum* divide o desenvolvimento em *Sprints* (de 2 a 4 semanas), o que pode ser comparado a cada iteração do XP. Reuniões de planejamento a cada iteração é comum a ambos os processos, bem como uma reunião curta (com os nomes de *Daily Scrum*, no *Scrum* e *Daily Meeting*, no XP) feita diariamente. Nessas reuniões diárias são discutidos: o que falta ser feito até o fim da iteração; as dificuldades encontradas; e os avanços desde a última reunião de iteração. Além do mais, ambas as metodologias prezam pelo cliente inserido no processo de desenvolvimento.

3. InteliMED

Nesta seção são apresentados: a metodologia, incluindo técnicas e ferramentas, empregada durante o planejamento e construção do InteliMED; o funcionamento do sistema, bem como tecnologias e arquitetura utilizadas; e o processo de descoberta de conhecimento em base de dados.

3.1. Processo de Desenvolvimento

Devido a multidisciplinaridade do sistema, o InteliMED possuiu equipes envolvendo profissionais e estudantes de saúde e de computação. A equipe de saúde auxiliou na coleta e validação de dados, bem como na elaboração do formulário utilizado como entrada para o algoritmo inteligente. Os formulários foram aplicados a pacientes atendidos em serviços ambulatoriais de alergia e de pediatria e adolescência. A equipe de computação planejou e desenvolveu o sistema computacional.

Durante o processo de desenvolvimento do InteliMED a metodologia aplicada foi considerada híbrida, possuindo características adaptadas principalmente de dois métodos ágeis – o XP e o *Scrum*. O processo resultante foi iterativo e incremental, já que diversas partes do sistema eram desenvolvidas ao mesmo tempo e a cada iteração era concluída uma parte do sistema funcionando [Sommerville 2004].

Dispositivos móveis possuem diversos problemas quanto a interação com o usuário, dentre outros motivos por possuírem tamanhos de telas pequenos e diferentes dos que o usuário está habituado. Assim, foi feita prototipação de telas baseada na apresentação *Excellence in the Android User Experience* [Nurik 2010] do *Google Developer Day 2010* aprovada junto ao *Product Owner* – um estudante de medicina vinculado ao projeto.

Além de prototipação, aplicou-se programação em pares, refatoração de código e integração contínua provenientes do XP. Histórias de Usuários e um *Product Backlog* foram produzidos pelo *Scrum Team*, principalmente pelo *Scrum Master* e pelo *Product Owner*. *Sprint Backlogs* foram estimados utilizando técnica de *Planning Poker*, que permite que os próprios programadores estimem quanto tempo levam para completar uma determinada tarefa. O gerenciamento do cronograma e produtividade se dava através do *Burn Down Chart*.

A implementação do servidor está integrada ao Jenkins, sistema de gerenciamento com integração contínua que possui diversas funcionalidades para auxiliar o monitoramento e a execução automática de testes [Smart 2011], e ao Sonar, plataforma de gestão da qualidade dedicada a analisar e medir continuamente a qualidade do código fonte [SonarSource 2011]. Além disso, o sistema móvel também foi integrado a um projeto responsável por todos os testes unitários, através do framework de testes JUnit para Android [Google Inc. 2010].

Com o objetivo de intensificar a comunicação entre as equipes foram utilizadas algumas ferramentas como Yammer [Yammer 2012], Manymoon [Manymoon 2011], Google Groups [Google Inc. 2012], Skype [Skype 2012] e Team Viewer [Team Viewer 2012]. O Yammer, que é uma rede social corporativa auxiliou o grupo quanto a troca de informações que exigiam comunicação rápida. No Manymoon, que auxiliou no gerenciamento de atividades, eram cadastradas tarefas com seus respectivos prazos. Cada pessoa do grupo escolhia uma tarefa a fazer, sendo possível, por exemplo, saber qual programador implementou qual funcionalidade e em quanto tempo. Informações de interesse de todas as equipes eram compartilhadas no grupo de discussão do Google. Para reuniões que não podiam ser presenciais, utilizou-se o Skype, para compartilhamento de mensagens de áudio entre todos os integrantes, em conjunto com o Team Viewer, no qual um membro compartilhava a tela com todos os demais.

Devido a equipe de desenvolvimento ser constituída por mais de um programador, houve a necessidade de gerenciar o acesso a arquivos compartilhados. Para tal, foi utilizado o sistema de controle de versão SVN [CollabNet 2009], por ser do modo *Copy-Modify-Merge*, o que permite que mais de uma pessoa trabalhe em um arquivo ao mesmo tempo. O código de todo o projeto está integrado com um repositório no Google Code [Google Inc. 2011].

3.2. O Sistema

O sistema é composto pelos seguintes módulos: aplicação móvel e servidor. Duas equipes trabalharam nesses módulos, ambas de computação. A arquitetura do InteliMED pode ser vista na Figura 2. A visão global da arquitetura mostra que a aplicação está distribuída entre o equipamento móvel e o servidor.

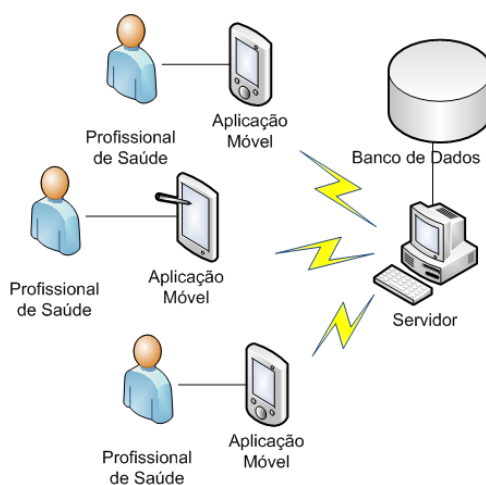


Figura 2. Arquitetura do InteliMED.

Na aplicação móvel utilizou-se a tecnologia Java e a plataforma Android. A principal funcionalidade do InteliMED é prover o diagnóstico de asma, e essa funcionalidade é executada mesmo sem conexão com o servidor, aumentando a aplicabilidade prática do sistema inclusive em áreas de população carente onde a rede é limitada de recursos, ou até mesmo inexistente.

A funcionalidade da aplicação móvel que determina o diagnóstico ocorre quando o médico responde a um questionário exibido na tela, coletando evidências, as quais serão utilizadas pela aplicação móvel para determinar o diagnóstico. A árvore de decisão é percorrida de acordo com as respostas do questionário e, concluído o processamento, o resultado é informado ao usuário. Após exibir o diagnóstico, o aplicativo questiona se o médico concorda com o resultado, ou não, e possibilita a inserção de comentários.

Outras funcionalidades do sistema móvel são:

- “Enviar Evidência” – tem o objetivo de enviar as informações do diagnóstico (incluindo opinião médica), que estão armazenadas localmente, para o servidor.
- “Atualizar Árvore” – possibilita a atualização da árvore de decisão do dispositivo móvel, a antiga é removida e usa-se a gerada pelo servidor.

No servidor foi armazenada a árvore para diagnóstico de asma resultante do algoritmo de mineração de dados C4.5, que é uma implementação da técnica de árvore de decisão, conforme detalhado na seção 3.3 [Quinlan 1992]. A tecnologia utilizada para implementação do servidor foi Grails, um *framework* para construção de aplicações *web* através da linguagem de programação Groovy (uma linguagem dinâmica para plataforma Java) com foco em alta produtividade e desenvolvimento rápido [SpringSource 2011]. No servidor são armazenadas as evidências enviadas pela aplicação móvel, com o objetivo de melhorar a árvore de decisão, por meio da utilização dos dados coletados.

3.2.1. Fluxo de Informação

Após ser instalado, acessa-se o InteliMED através do menu do aparelho móvel, sendo a tela inicial uma validação do usuário, conforme visualizado na Figura 3(A).



Figura 3. Telas do sistema móvel.

Ao acessar o sistema o usuário pode diagnosticar, enviar evidências e atualizar o *script* da árvore utilizada nos diagnósticos, como ilustrado na Figura 3(B). A Figura 3(C) mostra a tela de preenchimento do formulário, exibida ao clicar em “Diagnóstico”. Após o preenchimento do questionário é exibida a tela com o resultado processado pelo sistema, que pode ser vista na Figura 3(D). Já ao selecionar “Enviar evidência” ou “Atualizar árvore”, é exibida uma mensagem enquanto os dados são sincronizados.

3.3. Mineração de Dados

A mineração de dados – principal atividade do processo de descoberta de conhecimento (Knowledge Discovery in Databases - KDD) – consiste em aplicar algoritmos com a finalidade de extrair conhecimento de bases de dados [Fayyad, Piatetsky-Shapiro, Smyth 1996]. Neste projeto, conforme justificado na seção 2, o algoritmo escolhido foi árvore de decisão. Para auxiliar o processo de KDD foi utilizada uma ferramenta de código aberto chamada WEKA (Waikato Environment for Knowledge Analysis), que possui implementações de vários algoritmos de mineração de dados [Hall et al. 2009].

Como entradas para o algoritmo foram utilizados dados obtidos a partir de um formulário com 40 questões pré-selecionadas pela equipe de saúde. Para que o resultado obtido tivesse maior relevância foram realizados 34 experimentos na ferramenta WEKA. O algoritmo C4.5 foi utilizado em todos os casos, variando-se apenas alguns atributos importantes do algoritmo, tais como: o tipo de poda; o tipo de pré-processamento e o tipo de validação utilizada nos testes [Quinlan 1992].

Devido ao número reduzido de dados coletados (apenas 93 pessoas responderam o formulário), houve a necessidade da criação de mais instâncias para o formulário. Desta forma, com o auxílio da equipe de saúde foram criados mais 50 cenários. No total 143 instâncias foram utilizadas, um número ainda pequeno, porém suficiente para resultar numa árvore com taxa de acerto de 91,61%. Esta taxa de acerto pode ser considerada aceitável, devido a natureza de apoio a decisão do sistema, ou seja, o sistema visa auxiliar o médico no diagnóstico e não substituí-lo nesta tarefa.

4. Contribuições

Além do próprio sistema, diversos artefatos foram produzidos, como documentos de arquitetura e requisitos, e artefatos gerados a partir de testes e avaliação junto ao time de saúde. Esta seção apresenta a árvore de decisão e os testes realizados.

4.1. Árvore de Decisão

A construção da árvore de decisão gerada a partir do processo de mineração de dados aplicado aos dados obtidos pela equipe de saúde foi outro relevante artefato gerado. A Figura 4 apresenta a árvore de decisão utilizada para construção do sistema.

A árvore foi gerada utilizando inicialmente *guidelines* de saúde e o algoritmo C4.5, de modo a torná-la sucinta, com poucas perguntas, mas com a capacidade de classificação de Asma e Não Asma aceitável. Desta forma, como observado na figura 4, o objetivo foi alcançado, pois inicialmente o questionário utilizado possuía 40 perguntas e ao final da execução do algoritmo, este ficou reduzido a 8 perguntas, que apóiam satisfatoriamente o diagnóstico da doença de Asma.

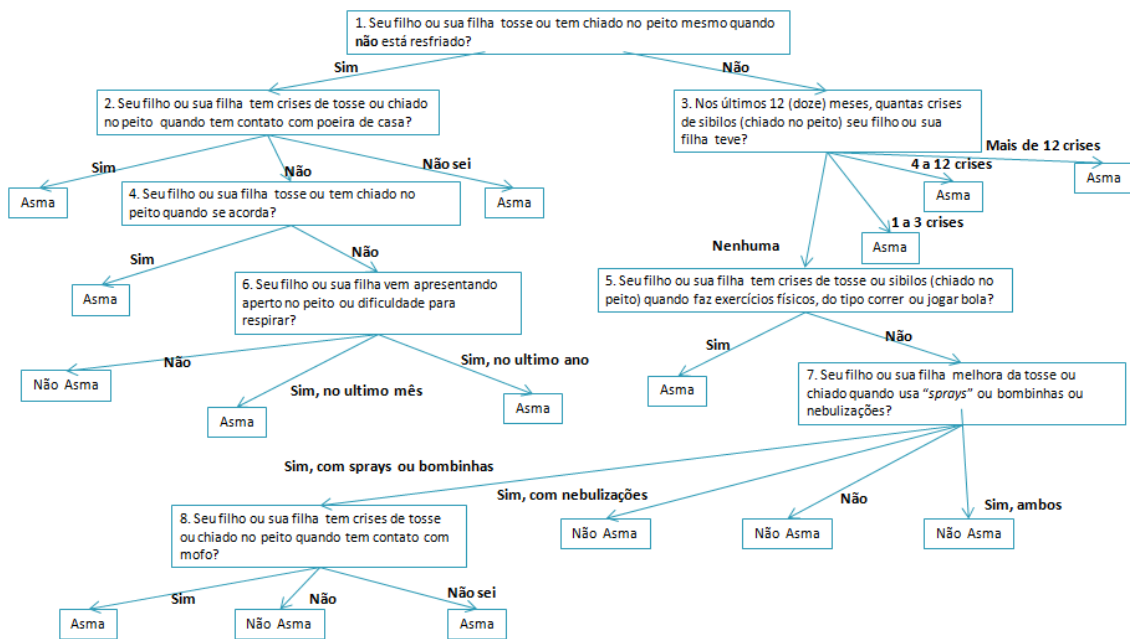


Figura 4. Árvore de decisão.

4.2. Testes

Foram elaborados planos de testes através de casos de testes, que especificavam descrição, pré-condições e os resultados esperados do processamento. Foram feitos ao todo 44 casos de testes funcionais, sendo 7 realizados por desenvolvedores e 37 por usuários com domínio da área médica. O sistema foi também testado por médicos sem vínculos com o projeto, em unidades de saúde básica.

Os testes de sistema, realizados por desenvolvedores, avaliaram o funcionamento do sistema completamente integrado no ambiente de produção, e as execuções dos planos de testes duraram em média 60 minutos. Nesses testes, além de utilizar o equipamento móvel foi necessário acessar o servidor para checar a integridade de todo o sistema.

Já os testes executados por usuários com domínio da área médica focaram nos requisitos específicos do servidor e da aplicação móvel. Em média cada execução de plano de teste teve duração de 20 minutos. Os principais casos de testes funcionais executados foram: 1) Efetuar *login* - com dados corretos, incorretos e campos vazios; 2) Realizar Diagnóstico - um teste para cada um dos 16 possíveis caminhos da árvore; 3) Armazenar Evidência - testes nos quais o médico concorda ou não com o resultado, além de justificar sua opinião; 4) Enviar evidência para o servidor – testes que possuíam bases de dados contendo evidências em que o médico concordou com o resultado e contendo evidências em que o médico não concordou com o resultado e justificou; 5) Atualizar árvore de decisão através do servidor para classificação de asma - testes com a base de dados local contendo uma árvore pré-cadastrada para sugerir a uma pessoa ir ou não jogar tênis e contendo uma árvore pré-cadastrada para classificação de asma. 6) Inserção de novo usuário; 7) Editar informações de usuário cadastrado; e 8) Exclusão de usuário.

Os testes de sistema realizados foram criados pela permutação dos itens 2, 4 e 5 da lista de casos de testes executados por usuários, e, adicionalmente, foi realizado um teste extra devido a sua importância: atualizou-se a árvore de decisão e depois testou-se todos os 16 possíveis caminhos da árvore de decisão.

Adicionalmente, foi realizada avaliação de usabilidade através de teste com utilizadores e avaliação heurística, pela qual foram avaliadas 24 heurísticas de cores, conteúdo, imagens, instruções, navegação e usabilidade, das quais 16 atendem completamente as recomendações, 7 atendem parcialmente e apenas uma, que exige que todos os botões e textos habilitados tenham textos de ajuda, não atende completamente. Nos testes com utilizadores, foram passadas algumas tarefas, tais como: obter diagnóstico de um paciente. Observou-se que usuários de *smartphones*, mesmo que sem sistema Android, executavam mais rapidamente as tarefas. Os utilizadores sugeriram algumas melhorias, como: (i) quando o sistema entrar em “descanso” seria interessante armazenar o estado do sistema no qual o usuário se encontra, para que ao retornar do “descanso” ele possa ir para a tela de autenticação e posteriormente possa voltar ao estado salvo; (ii) no menu iniciar, ao clicar sobre o botão “voltar” do dispositivo, seria bom que o sistema, não apenas fizesse *logoff*, mas também fechasse a aplicação.

5. Considerações Finais e Trabalhos Futuros

Este trabalho apresentou conhecimentos aplicados na experiência de construção de um sistema de apoio a decisão médica desenvolvido como prova de conceito. Técnicas de mineração de dados foram aplicadas, produzindo uma árvore de decisão capaz de sugerir um diagnóstico (com taxa de acerto de 91,61%) para asma.

Dentre as dificuldades encontradas durante o desenvolvimento deste trabalho algumas devem ser destacadas: a dificuldade de obtenção de equipamentos como *smartphones* e *tablets* para os testes; a integração de equipes distribuídas fisicamente, parcialmente resolvida com uso de ferramentas de apoio; evasão de desenvolvedor, problema pequeno devido à aplicação do conceito de propriedade coletiva.

Como trabalhos futuros, planeja-se que o InteliMED seja implantado em um servidor usando o conceito de computação nas nuvens, de modo a prover maior facilidade na implantação prática do sistema. Pretende-se também a aplicação do InteliMED em assentamentos rurais com o intuito de capacitar e promover a atenção à saúde básica por meio de protocolos educativos.

6. Agradecimentos

Agradecemos à FINEP (Financiadora de Estudos e Projetos) e ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) pelo apoio financeiro recebido para execução deste projeto.

Referências

- Beck, K. (1999). “Programação Extrema Explicada”. Bookman.
- Beck, K. et al. (2001). “Agile Manifesto”. Disponível em: <<http://agilemanifesto.org/>>. Acessado em Dezembro de 2011.

- Castro, L.S.S. et al. (2004). “HandMed – Um Sistema Móvel Integrado para Captura Automática de Sintomas”. In: IX Congresso Brasileiro de Informática na Saúde.
- CollabNet. (2009). “Tigris Subversion”. Disponível em: <<http://subversion.tigris.org/>> Acessado em Dezembro de 2011.
- Fayyad, U., Piatetsky-Shapiro, G., Smyth, P. (1996). “From data mining to knowledge discovery in databases”. AI Magazine, v. 17, p. 37-54.
- Germanakos, P., Mourlas, C., & Samaras, G. (2005). "A Mobile Agent Approach for Ubiquitous and Personalized eHealth Information Systems". Proceedings of the Workshop on 'Personalization for e-Health' of the 10th International.
- Google Inc. (2010) “Android Developers Guide – Testing Fundamentals - junit”. Disponível em: <http://developer.android.com/guide/topics/testing/testing_android.html#JUnit>. Acessado em Dezembro de 2011.
- Google Inc. (2011). “Google Code”. Disponível em: <<http://code.google.com/intl/>> Acessado em Dezembro de 2011.
- Google Inc. (2012). “Google Groups”. Disponível em: <<https://groups.google.com>>. Acessado em Janeiro de 2012.
- Gusmão, C. M. G. (2012). “Relatório Técnico FINEP – Suporte Remoto a Diagnóstico Médico utilizando Tecnologias Inteligentes”. Relatório Técnico. Janeiro de 2012.
- Hall, M. et al. (2009). “The weka data mining software: An update”. SIGKDD Explorations, v. 11.
- InfoGraphic Labs. (2012). “Android vs iPhone”. Disponível em: <<http://infographiclabs.com/infographic/best-of-the-smartphones-android-vs-iphone-4/>>. Acessado em Janeiro de 2012.
- Ken, F. et al. A Tree-Based Decision Model to Support Prediction of the Severity of Asthma Exacerbations in Children. Journal of Medical Systems, v.34, p.551-562, 2010.
- Kulikowski, C. A., Weiss, S. (1982). “Representation of expert knowledge for consultation. The CASNET and EXPERT projects”. In: Szolovitz, P. (Ed.) Artificial Intelligence in Medicine. Westview Press, Boulder, CO, pp. 21-55.
- Manymoon. (2011) “Manymoon”. Disponível em: <<https://manymoon.com/>>. Acessado em Dezembro de 2011.
- Menezes Jr., J. et al. (2011). “InteliMED: uma experiência de desenvolvimento de sistema móvel de suporte ao diagnóstico médico”. Revista Brasileira de Computação Aplicada. Passo Fundo, v. 3, n. 1, p. 30-42.
- Moraes, D. A., Pisa, I. T., Lopes, P. R. L. (2004). “Protótipo para Coleta de Informações em Saúde Utilizando Dispositivos Móveis”. In: XI Congresso Brasileiro de Informática na Saúde.
- Nurik, R. (2010). “Excellence in the Android User Experience”. Disponível em: <<http://www.slideshare.net/mobilegui/excellence-in-the-android-user-experience>>. Acessado em Setembro de 2011.

- Patil, R. S., Szolovits, P., Schwartz, W. B. (1982). "Modeling knowledge of the patient in acid-base and electrolyte disorders". In: Szolovitz, P. (Ed.) Artificial Intelligence in Medicine. Westview Press, Boulder, CO, pp. 191-226.
- Quinlan, J. (1996). Learning decision tree classifiers. ACM Computing Surveys (CSUR), 28(1):71-72.
- Quinlan, J. (1992). C4.5: Programs for Machine Learning. Morgan Kaufmann.
- Razzouk, D., Shirakawa, I., Mari, J. de J. (2000). "Sistemas inteligentes no diagnóstico da esquizofrenia". Revista Brasileira de Psiquiatria [online]. Vol.22, suppl.1, pp. 35-37. Disponível em: <<http://www.scielo.br/pdf/rbp/v22s1/a12v22s1.pdf>> Acessado em Dezembro de 2011.
- Schwaber, K., Beedle, M. (2002). "Agile Software Development with SCRUM". Prentice-Hall.
- Seixas, F. L. et al. (2011). "Sistema de Apoio à Decisão Aplicado ao Diagnóstico da Doença de Alzheimer". XI Workshop de Informática Médica. XXXI Congresso da Sociedade Brasileira de Computação. Disponível em: <http://www.dimap.ufrn.br/csbc2011/anais/eventos/contents/WIM/WIM_Sessao_3_Artigo_7_Seixas.pdf>. Acessado em Dezembro de 2011.
- Shortliffe, E.H., Scott, A., Bischoff, M. (1981). "ONCOCIN: an expert system for oncology protocol management". Proc. Internat. Joint Conf. Artif. Intellig., p. 876-881.
- Skype. (2012). "Skype". Disponível em: <http://www.skype.com/>>. Acessado em Janeiro de 2012.
- Smart, J. F. (2011). "Jenkins: The Definitive Guide". O'Reilly.
- Soares, M. S. (2004). "Metodologias Ágeis Extreme Programming e Scrum para o Desenvolvimento de Software". Revista Eletrônica de Sistemas de Informação v. 3, n. 1.
- Sociedade Brasileira de Pneumologia e Tisiologia. (2006). "IV Consenso Brasileiro no Manejo da Asma". Disponibilizado no Congresso Brasileiro de Pneumologia.
- Sommerville, I. (2004). "Engenharia de Software". São Paulo: Addison Wesley.
- SonarSource. (2011) "Sonar". Disponível em: <<http://www.sonarsource.com>>. Acessado em Dezembro de 2011.
- SpringSource. (2011). "Grails". Disponível em: <<http://grails.org/>>. Acessado em Dezembro de 2011.
- Team Viewer. (2012). "Team Viewer". Disponível em: <<http://www.teamviewer.com/pt/index.aspx>>. Acessado em Janeiro de 2012.
- Vital Wave Consulting (2009). "mHealth for Development: The Opportunity of Mobile Technology for Healthcare in the Developing World". United Nations Foundation, Vodafone Foundation. p. 9.
- Yammer. (2012). "Yammer – The Enterprise Network". Disponível em: <<http://www.yammer.com>>. Acessado em Janeiro de 2012.