

EnDia 2.1: Um Sistema para Construção de uma Base de Dados de Exames de Endoscopia Digestiva Alta

Allan Koch Veiga¹, Hwei Diana Lee¹, Joylan Nunes Maciel¹, Daniel de Faveri Honorato¹, Carlos Andres Ferrero¹, André Gustavo Maletzke¹, Renato Bobsin Machado¹, Luiz Henrique Dutra da Costa¹, Cláudio Saddy Rodrigues Coy^{1,2}, João José Fagundes^{1,2}, Feng Chung Wu^{1,2}

¹ Laboratório de Bioinformática - LABI,
Universidade Estadual do Oeste do Paraná - UNIOESTE, Brasil
² Serviço de Coloproctologia da Faculdade de Ciências Médicas - FCM,
Universidade Estadual de Campinas - UNICAMP, Brasil

Abstract. Technological advances have enabled the storage and process of a huge amount of data. This phenomenon can be observed in many areas, such as medicine, where hospitals and clinics accumulate medical information in digital format arising however a potential problem that is becoming very frequent: the difficulty of analyzing and extracting useful information of databases. In this context, this work introduces the web system EnDia 2.1, which has the objective of building repositories of information of Upper Digestive Endoscopy for further application of knowledge extraction algorithms. It introduces also some optimized aspects of EnDia 2.1 system, like the environment change from desktop to web, a relational entity model which brings more flexibility, and an architecture of scalable systems.

Key words: *Knowledge Discovery from Databases, Gastroenterology, Client-Server Model*

1 Introdução

Com a crescente acessibilidade e amplo uso de novas tecnologias e recursos computacionais, cada vez mais hospitais e clínicas médicas têm armazenado informações sobre pacientes e processos laboratoriais em formatos digitais. Essa tendência representa um avanço na gestão de informações médicas. Nesse sentido, dois pontos podem ser levados em consideração. O primeiro está relacionado à construção de sistemas médicos com número de atributos na casa de centenas ou milhares. Outro ponto está relacionado ao fato de que o acúmulo de informações que são geradas em uma clínica ou hospital dificulta a análise e a extração de informações úteis que poderiam auxiliar os especialistas a realizar análises mais completas com o auxílio de processos apoiados por ferramentas computacionais. Um desses processos é o de Mineração de Dados — MD [1,2], o qual tem por objetivo extrair conhecimento, de modo automático, a partir de bases de dados.

O foco deste trabalho está no primeiro ponto, ou seja, na construção de um sistema médico que lida com um grande número de atributos que possam ser adicionados, retirados ou atualizados dinamicamente de modo interativo e que amplia e facilita o acesso de usuários por meio da *web*. Este trabalho faz parte do projeto de Análise Inteligente de Dados, o qual é desenvolvido em uma parceria entre o Laboratório de Bioinformática — LABI da Universidade Estadual do Oeste do Paraná — UNIOESTE Foz do Iguaçu, o Serviço de Endoscopia Digestiva Alta do Hospital Municipal de Paulínia — HMP e o Serviço de Coloproctologia da Faculdade de Ciências Médicas da Universidade Estadual de Campinas — UNICAMP. Anteriormente, no LABI, foi definido um método para desenvolvimento de repositórios médicos, o qual resultou em uma primeira versão do sistema EnDia. Nos trabalhos de [3,4,5,6,7,8] podem ser encontradas aplicações desse método.

Neste trabalho é apresentado o sistema EnDia 2.1, o qual, como mencionado, tem como objetivo o gerenciamento e a construção de uma Base de Dados — BD estruturada sobre informações de exames de Endoscopia Digestiva Alta — EDA³. Um dos problemas encontrados na primeira versão do EnDia 1.0 foi a dificuldade de lidar com grande quantidade de informações que deveriam ser mapeadas.

O EnDia 2.1 é um sistema, em processo de desenvolvimento, que está sendo implementado na plataforma JBoss Seam [11], a qual permite maior flexibilidade em relação às tecnologias utilizadas no sistema. O sistema foi projetado de maneira genérica e escalável, de modo que possa lidar com um grande número de informações permitindo que o profissional da saúde possa decidir quais informações devem ser armazenadas no sistema, de modo dinâmico e interativo.

Este trabalho está organizado da seguinte maneira: na Seção 2 é realizada uma breve descrição do processo de desenvolvimento do EnDia 1.0. Na Seção 3 é apresentada uma descrição detalhada do EnDia 2.1, com informações sobre a plataforma de desenvolvimento, a arquitetura do novo sistema e o modelo entidade-relacionamento. Na Seção 4 são apresentados os resultados e as discussões do trabalho e na Seção 5 são apresentadas as conclusões e trabalhos futuros.

2 Sistema EnDia 1.0

Durante a realização de exames de EDA, informações a respeito do esôfago, do estômago e do duodeno são coletadas. De modo complementar, também são consideradas informações relacionadas ao histórico do paciente e aos exames de Anatomia Patológica e de Teste da Urease, os quais contribuem para o diagnóstico da bactéria *Helicobacter Pylori*, presente em grande parte das doenças pépticas [9,10,5]. A construção de uma base de dados estruturada sobre essas informações possibilita a aplicação do processo de MD, para auxiliar especialistas na tomada de decisões.

³ Importante exame para o diagnóstico de doenças gastroduodenais, como úlceras e gastrites, as quais representam entidades em gastroenterologia de maior incidência na população mundial [3,9,10]

O EnDia 1.0 foi desenvolvido em quatro fases: (1) análise de requisitos; (2) modelagem do sistema; (3) implementação do sistema; (4) teste do software. Durante a Fase (1), utilizou-se a técnica de prototipação para a coleta de requisitos, o que possibilitou uma visão mais realística do funcionamento do sistema e um aumento da eficácia no levantamento de requisitos e das funcionalidades a serem implementadas. Na Fase (2), foi utilizada a metodologia do processo unificado subsidiada pela linguagem de modelagem *Unified Modeling Language* — UML⁴. Na Fase (3), foi utilizado o ambiente de desenvolvimento *desktop* Delphi 7.0. Após a implementação do sistema, realizou-se o teste de *software* — Fase (4).

O sistema EnDia 1.0 atendeu aos requisitos estipulados na Fase (1). Conforme mencionado, a grande dificuldade no desenvolvimento do EnDia 1.0, foi conseguir organizar e mapear no sistema o grande número de atributos que fizeram parte dos requisitos levantados junto aos especialistas. Desse modo, um dos desafios foi a aplicação de outras ferramentas que pudessem auxiliar na construção, de modo eficiente e escalável, de um sistema do porte do EnDia 1.0.

3 Sistema EnDia 2.1

O EnDia 2.1 manteve os mesmos objetivos do EnDia 1.0, ou seja, o mapeamento de informações de EDA. No entanto, para essa versão estão sendo utilizadas novas tecnologias, uma arquitetura de sistema flexível e reestruturações no modelo de dados, no intuito de atender requisitos de usabilidade, escalabilidade e manutenibilidade. O EnDia 2.1 está sendo implementado no modelo cliente-servidor que agrega aspectos como a centralização dos dados, portabilidade em relação ao sistema operacional, além de prover acesso remoto ao sistema. Desse modo, o sistema pode ser acessado via Internet ou em redes locais, que frequentemente são utilizadas em hospitais e clínicas médicas.

A interface gráfica utilizada no EnDia 2.1 é baseada na *Web 2.0*⁵, a qual inclui animações intuitivas e componentes visuais, como botões e menus, que permitem melhor interatividade entre o usuário e o computador. Ainda, o modelo cliente-servidor associado à plataforma J2EE, possibilita também a criação de interfaces gráficas *desktop* e para equipamentos portáteis, como celulares e *Personal Digital Assistants* — PDA. Para a evolução do sistema EnDia 1.0 para a versão 2.1, de acordo com as características mencionadas, foi necessário um estudo aprofundado de diversas tecnologias que permitissem o desenvolvimento do novo sistema.

3.1 Recursos e Ferramentas de Desenvolvimento

O EnDia 2.1 está sendo desenvolvido utilizando os seguintes recursos e ferramentas:

- Plataforma de desenvolvimento JBoss Seam;

⁴ Linguagem de modelagem de sistemas computacionais orientados a objetos.

⁵ <http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>

- Ambiente de desenvolvimento Red Hat Developer Studio;
- Servidor de aplicação JBoss Application Server;
- Sistema gerenciador de banco de dados PostgreSQL.

O JBoss Seam 2.1 é uma plataforma que integra, em um modelo unificado de desenvolvimento, uma ampla diversidade de tecnologias e *frameworks* consolidados para permitir a construção de aplicações *web* [12]. O JBoss Seam foi baseado no *framework* JavaServer Faces — JSF [13], porém é permitida a utilização de outras tecnologias para desenvolvimento de aplicações *web*, tais como Google Web Toolkit, Flex, Spring, entre outras. Por padrão, o JBoss Seam prevê suporte a Asynchronous Javascript And XML — AJAX e *Rich Internet Application* — RIA por meio do *framework* RichFaces [14,15]. A plataforma utiliza como padrão a linguagem *eXtensible Hypertext Markup Language* — XHTML⁶ para a construção da interface gráfica.

O JBoss Seam permite ainda o uso do Enterprise JavaBeans — EJB 3.0 de modo optativo, provendo escalabilidade e flexibilidade em relação ao uso do mesmo [16]. A plataforma também fornece flexibilidade em relação às tecnologias de manipulação de BD, permitindo o uso de Hibernate, JPA, JDBC entre outras [17].

O *Integrated Development Environment* — IDE que está sendo utilizado é o Red Hat Developer Studio — RHDS⁷. Esse ambiente baseado no IDE Eclipse⁸, é totalmente integrado com o JBoss Seam e o servidor de aplicação JBoss Application Server. O RHDS oferece editor visual de páginas XHTML, paleta de componentes RichFaces e adaptador otimizado para integração com o JBoss Application Server. Esse IDE fornece uma interface no formato de *wizard* para o *framework* Seam-gen, que consiste em um gerador de projetos JBoss Seam, o qual gera toda a estrutura do projeto, arquivos de configurações e classes entidades a partir da BD.

Com o Seam-gen também é possível a geração de aplicações a partir da BD (técnica de engenharia reversa — *botton up*) agilizando o processo de desenvolvimento a partir de um modelo entidade-relacionamento pré-existentes.

O servidor de aplicação utilizado para executar o EnDia 2.1 é o JBoss Server Application, e foi escolhido pelas seguintes características: possui suporte ao EJB, suporte a *webservice*, fácil administração, boa escalabilidade, possui baixo consumo de recursos computacionais, rápida iniciação e integração nativa com RHDS e JBoss Seam [18].

Para persistir as informações manipuladas no EnDia 2.1 está sendo utilizado o sistema gerenciador de banco de dados PostgreSQL 8.2⁹ por ser robusto, *open source* e atender a todos os requisitos do sistema EnDia 2.1.

⁶ <http://www.w3.org/TR/xhtml1/>

⁷ <http://jboss.com/products/devstudio/>

⁸ <http://www.eclipse.org/>

⁹ <http://www.postgresql.org/>

3.2 Arquitetura do Sistema

A plataforma JBoss Seam possui um modelo unificado de componentes, o qual permite que exista indistinção entre componentes de diferentes camadas da arquitetura do sistema [11]. Essa flexibilidade possibilita que a arquitetura seja projetada de acordo com a necessidade do sistema, permitindo definir as camadas da arquitetura e suas interações. A arquitetura projetada para o EnDia 2.1 é apresentada na Figura 1. A arquitetura é baseada no padrão de projeto Modelo-

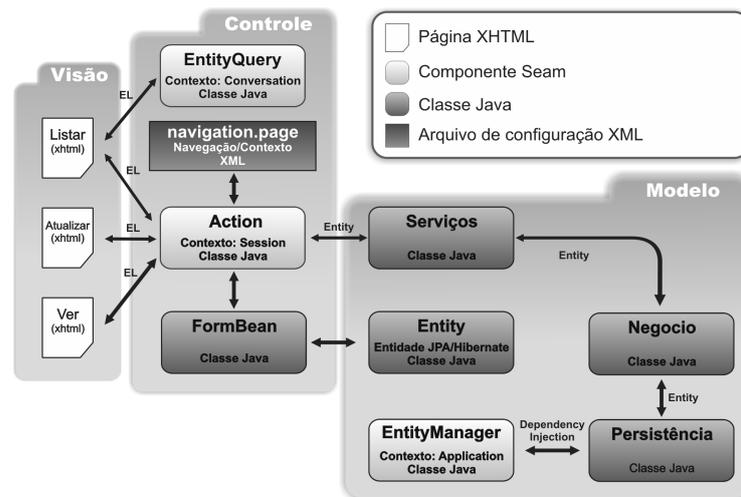


Figura 1. Arquitetura do sistema EnDia 2.1.

Visão-Controle — MVC [19]. Nesse contexto, a arquitetura está organizada da seguinte forma: as páginas XHTML estão na camada Visão; a classe FormBean, o arquivo de configuração navigation.page e os componentes EntityQuery e Action estão na camada Controle; e as demais classes e componentes estão na camada Modelo.

A camada Modelo é subdividida em outras três camadas: Serviços, Negócio e Persistência. Na camada Serviços estão os serviços (métodos) que serão chamados pela camada Controle (componente Action). Toda a lógica de cada serviço é implementada na camada Negócio. Essa camada é independente de tecnologia, para garantir alta escalabilidade, baixo acoplamento, manutenibilidade e independência de tecnologias. Desse modo, caso haja necessidade de trocar qualquer tecnologia de outras camadas, a lógica de negócio do sistema permanecerá inalterada. A camada Persistência está relacionada com interações primárias ao banco de dados, tais como persistências e consultas na BD. As classes Entity, são classes que representam entidades da BD. São as instâncias dessas classes que trafegam pelas camadas em toda a arquitetura.

Como apresentado na Figura 1, os componentes Action e EntityQuery são requisitados pela camada Visão através da *Application Programming Interface* — *API Expression Language* — EL¹⁰. Um Action possui todos os métodos que podem ser utilizados pelas páginas XHTML. O componente EntityQuery é utilizado para controlar tabelas de consultas. Esse componente é gerado automaticamente pelo Seam-gen e permite obter uma lista de entidades ordenadas, filtradas e/ou paginadas. A classe FormBean contém atributos e objetos utilizados na camada Visão.

A camada Visão está sendo implementada utilizando a API ServerFaces, para criação de *templates*, e componentes do *framework* RichFaces. Nesta camada são gerados todos os componentes gráficos, contudo, informações de alguns componentes, como por exemplo, tipo e descrição, poderão ser obtidas a partir da BD. A navegação entre as páginas é configurada por meio do arquivo navigation.page.

Como mencionado, a arquitetura proposta para o EnDia 2.1 tem como objetivo atender às necessidade do sistema, agregando escalabilidade e manutenibilidade, além de permitir o reuso dessa arquitetura em outros sistemas.

3.3 Modelo Entidade-Relacionamento

O Modelo Entidade-Relacionamento — MER¹¹ utilizado para mapear as informações coletadas pelo EnDia 2.1 possui diferenças estruturais em relação ao modelo utilizado no EnDia 1.0. Com o novo modelo, apresentado na Figura 2, é possível mapear de forma otimizada as mesmas informações da primeira versão. Uma das melhorias mais relevantes do EnDia 2.1 em relação ao EnDia 1.0, está na estrutura das entidades que armazenam informações sobre exames de EDA. Durante a Fase (1), análise de requisitos e prototipação, apresentada na Seção 2, foi identificada uma extensa quantidade de atributos correspondentes às características de exames de EDA. Desse modo, na primeira versão, a entidade EDA foi construída com centenas de atributos e caso houvesse a necessidade de remover, alterar ou incluir um atributo seria necessário alterar a estrutura da entidade, afetando assim a implementação do sistema em várias camadas da arquitetura. Essas características podem, ao longo do tempo, apresentar mudanças conforme a necessidade do especialista do domínio de registrar novas informações de interesse no exame de EDA. Outro problema encontrado no EnDia 1.0, está relacionado ao momento de cadastramento de um registro da entidade EDA na BD, pois uma instância da entidade EDA, com centenas de atributos, trafegava da camada Visão até a camada Persistência da arquitetura, consumindo muitos recursos computacionais.

Para contornar esses problemas foi criada uma entidade Atributo, que conterá informações relacionadas a cada características de um exame de EDA. Para organizar estes atributos foram criadas as entidades GrupoAtributo e SubGrupoAtributo. Assim é possível agrupar atributos semelhantes em grupos e subgrupos.

¹⁰ java.sun.com/products/jsp/reference/techart/unifiedEL.html

¹¹ Modelo diagramático que representa o modelo de dados de um sistema.

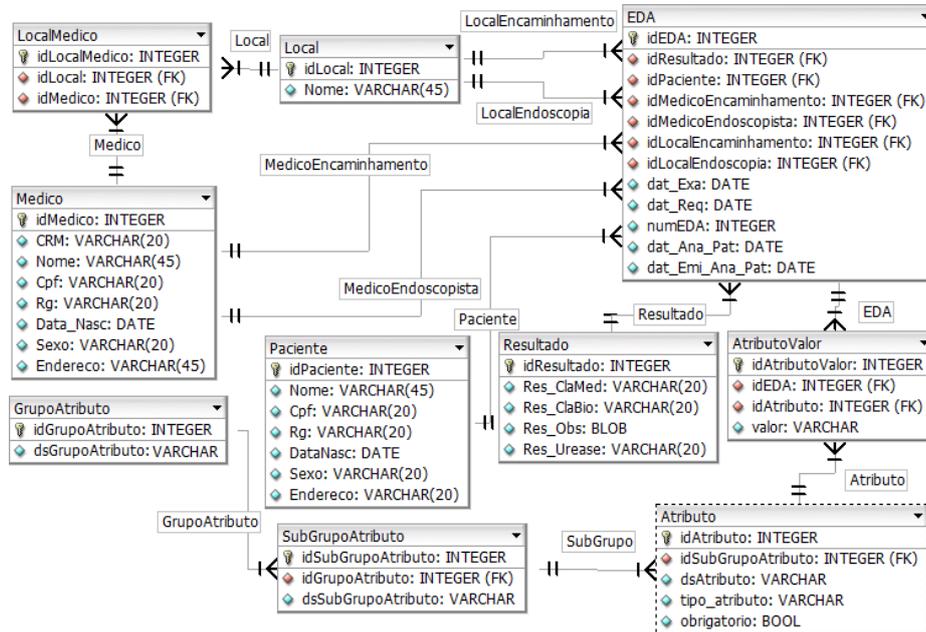


Figura 2. Modelo Entidade-Relacionamento do sistema EnDia 2.1.

Essa estrutura de dados permite que, ao cadastrar uma entidade EDA, apenas as informações referentes aos atributos preenchidos sejam persistidos na entidade AtributoValor com o valor do atributo e o identificador da entidade EDA cadastrada. Outra característica importante é que esse modelo permite a geração de interfaces gráficas dinâmicas por meio de componentes gráficos criados em tempo de execução utilizando as informações dos atributos, e seus grupos, cadastradas na BD.

4 Resultados e Discussão

O EnDia 2.1 é um sistema que fornece uma interface para a construção de BD médicas com informações de exames de EDA. Este sistema está sendo implementado na plataforma cliente-servidor com interface gráfica *web*, de modo a agregar aspectos como a centralização de dados e a portabilidade em relação a sistemas operacionais.

Essa portabilidade está relacionada com o aspecto *web* do EnDia 2.1, pois para utilizar o sistema, é necessário apenas de um navegador, independente do sistema operacional. A centralização de recursos computacionais em um único servidor pode contribuir na redução dos investimentos necessários para a integridade dos dados, a manutenção de *hardware* e a atualização de *software*. Desse modo, computadores terminais (clientes), de baixo custo, podem utilizar serviços disponibilizados e executados pelo servidor, proporcionando uma redução

nos custos econômicos e financeiros relacionados com a Tecnologia da Informação. Além disso, essa estratégia possibilita a aquisição de informações a partir de diversos locais. Contudo um aspecto importante que deve ser observado em sistemas da área de saúde, está relacionado a políticas de privacidade e segurança. A plataforma JBoss Seam provê soluções nesse sentido por meio da API Seam Security¹² que foi construída a partir do Java *Authentication and Authorization Service* — JAAS¹³, que fornece uma API robusta e configurável para autenticação de usuário, além de prover suporte a protocolos de comunicação segura como *Transport Layer Security* — TLS, *Secure Sockets Layer* — SSL, além de prover suporte completo à HTTPS sobre SSL/TLS [20].

A plataforma de desenvolvimento utilizada no EnDia 2.1 permite uma grande flexibilidade em relação às tecnologias empregadas no projeto. Desse modo, a escalabilidade do sistema não fica comprometida, permitindo a inclusão ou alteração de tecnologias, como por exemplo, *frameworks* de persistência ou tecnologias da camada de Visão, sem comprometer a estrutura do sistema. Além disso, o sistema demanda flexibilidade arquitetural, pois posteriormente módulos complementares inerentes ao processo de MD devem ser agregados. O JBoss Seam permite desenvolver uma arquitetura que atende às necessidades do sistema.

A arquitetura utilizada no EnDia 2.1 foi baseada no padrão MVC e está dividida de forma que a camada Modelo fornece Serviços para a camada Controle. Esses Serviços são implementados na camada Negócio, que independe de tecnologias, e assim permite que as tecnologias de outras camadas possam ser alteradas sem alterar a lógica de negócio do sistema.

Em relação ao MER, reestruturações foram implementadas de modo a melhorar alguns aspectos do EnDia 2.1 em relação a versão 1.0. Na primeira versão do EnDia, a entidade EDA possuía um elevado número de atributos, sendo que a interface gráfica era dependente dessa estrutura de informação, dificultando a manutenibilidade ao adicionar, alterar ou remover atributos dessa entidade [3]. Nesta segunda versão, o MER foi projetado de modo a contornar essa problemática, permitindo que o usuário, previamente autorizado pelo especialista, acrescente, altere ou remova novos atributos referentes aos exames de EDA de modo interativo e dinâmico. Desse modo, o modelo proposto foi idealizado considerando as deficiências do modelo anterior. Entretanto, é importante considerar o custo computacional para se manter o modelo proposto a medida que o volume de dados é aumentado. Para tanto, essa fator deverá ser estudado e avaliado em trabalhos futuros.

Como mencionado, a interface gráfica é gerada dinamicamente a partir dos atributos cadastrados. Com a organização dos atributos por meio das entidades GrupoAtributo e SubGrupoAtributo é possível gerar, em tempo de execução, um menu com as informações correspondentes à essas entidades. A Figura 3 apresenta uma tela que utiliza os componentes gráficos que estão sendo empregados no EnDia 2.1 para a geração da interface gráfica automática.

¹² <http://docs.jboss.com/seam/1.1.5.GA/reference/en/html/security.html>

¹³ <http://java.sun.com/javase/technologies/security/>

The screenshot shows the 'Esofago Superior' (Upper Esophagus) section of the EnDia 2.1 system. On the left, a 'Histórico do Paciente' (Patient History) sidebar lists various conditions under three categories: Esôfago (Superior, Médio, Inferior), Estômago (Gastrite, Hipertrofia, Metaplasia, Antrofia, Estenose), and Duodeno (Duodenite, Estenose, Neoplasia). The main area contains the following fields and controls:

- Esofago Superior:** Radio buttons for 'Normal' and 'Anormal' (selected).
- Esofagite:** Radio buttons for 'Sim' (selected), 'Sim com fibrina', and 'Nao'.
- Extensão Longitudinal:** Text input field with '0 cm'.
- Circunferencia:** Text input field with '0 graus'.
- Esofagite Erosiva:** Radio buttons for 'Sim' (selected), 'Sim com fibrina', and 'Nao'.
- Soma das Extensões:** Text input field with '0 cm'.
- Quantidade:** Spin box with '0'.
- Grau:** Spin box with '1'.
- Esofagite Ulcerada:** Radio buttons for 'Sim' (selected), 'Sim com fibrina', and 'Nao'.
- Soma das Extensões:** Text input field with '0 cm'.
- Lesão Esbranquiçada:** Radio buttons for 'Sim' and 'Nao' (selected).
- Extensão Longitudinal:** Text input field with '0 cm'.

At the bottom right, a small text reads: 'Sistema EnDia - Desenvolvido pelo Laboratório de Bioinformática - LABI.'

Figura 3. Componentes gráficos utilizados no EnDia 2.1.

5 Conclusão

Neste trabalho foi apresentado o EnDia 2.1, em fase de desenvolvimento, que tem como objetivo principal fornecer uma interface gráfica *web* para construção de uma BD estruturada com informações de exames de EDA, armazenando-as em um formato adequado para aplicação de processos de MD. De acordo com as tecnologias utilizadas no desenvolvimento do sistema EnDia 2.1, será possível atender os requisitos de escalabilidade, usabilidade, manutenibilidade e segurança, além de possibilitar a criação de uma arquitetura que poderá ser aplicada na construção de sistemas para outros domínios da área médica. O modelo de dados utilizado no sistema EnDia 2.1 possibilita a geração de interface gráfica dinâmica a partir da BD, tornando o sistema mais flexível. O estrutura do MER relacionado a entidade EDA, poderá ser utilizado em trabalhos futuros em que haja a necessidade de se obter informações no formato atributo-valor.

Como trabalhos futuros, após a conclusão da implementação do sistema, é proposta a realização de testes de verificação e aceitação do sistema EnDia 2.1, a implementação e a inserção de módulos para extração de conhecimento utilizando a plataforma, a arquitetura e a base de dados apresentados neste trabalho.

Agradecimentos

Ao Programa de Desenvolvimento Tecnológico Avançado — PDTA da Fundação Parque Tecnológico Itaipu — FPTI-BR e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico — CNPq.

Referências

1. Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P.: From data mining to knowledge discovery in databases. *AI Magazine* vol. 17 (1996) 37–54

2. Rezende, S.O.: Sistemas Inteligentes: Fundamentos e Aplicações. Editora Manole, Barueri, SP (2003)
3. Ferrero, C.A., Maletzke, A.G., Lee, H.D., Chung, W.F., Neto, A.P., Fagundes, J.J., Góes, J.R.N.: EnDia - Uma Ferramenta para Construção de Repositórios de Dados de Endoscopia Digestiva Alta. In: Anais do X Congresso Brasileiro de Informática na Saúde. Florianópolis, SC - Brasil. (2006) 1–6
4. Honorato, D.D.F., Cherman, E.A., Lee, H.D., Monard, M.C., Wu, F.C.: Construção de uma Representação Atributo-valor para Extração de Conhecimento a partir de Informações Semi-estruturadas de Laudos Médicos. In: Anais do Conferência Latinoamericana de Informática - CLEI. San José - Costa Rica. (2007) 1–12
5. Honorato, D.D.F., Lee, H.D., Wu, F.C., Machado, R.B., Neto, A.P., Fagundes, J.J., Góes, J.R.N.: *H.pylori*-mindsys: Um protótipo de sistema baseado em conhecimento para auxílio na predição da bactéria *Helicobacter pylori* em doenças pépticas. In: Anais do V Workshop de Informática Médica, Porto Alegre, RS (2005) 1–1
6. Lee, H.D., Monard, M.C., Wu, F.C.: Seleção de atributos relevantes e não redundantes usando a dimensão fractal do conjunto de dados. In: Anais do V Encontro Nacional de Inteligência Artificial, XXV Congresso da Sociedade Brasileira de Computação, Porto Alegre, RS (2005) 444–453
7. Ferro, M., Lee, H.D., Esteves, S.C.: Intelligent data analysis: A case study of the diagnostic sperm processing. In: Proc. of the International Conference on Computer Science, Software Engineering, Information Technology, e-Business and Applications, Foz do Iguaçu, PR (2002) 116–120
8. Esteves, S.C., Lee, H.D., Monard, M.C.: Inteligência artificial aplicada à andrologia: Um estudo de caso do processamento de sêmen diagnóstico. In: Anais do XXVII Congresso Brasileiro de Urologia, Fortaleza, CE (2001) 1–1
9. Pellicano, R., Fagoonee, S., Palestro, G., Rizzetto, M., Figura, N., Ponzetto, A.: The diagnosis of *Helicobacter pylori* infection: guidelines from the maastricht 2-2000 consensus report. *Minerva Gastroenterol Dietol* **vol. 50(2)** (2004) 125–33
10. Cordeiro, f.: Endoscopia Digestiva. Editora Médica e Científica Ltda. (1994)
11. Allen, D.: Seam in Action. Manning Publications Co., Greenwich, CT, USA (2008)
12. Fayad, M., Schmidt, D.C.: Object-oriented application frameworks. *Commun. ACM* **40** (1997) 32–38
13. Geary, D.M., Horstmann, C.S.: Core JavaServer Faces. Prentice Hall PTR, Upper Saddle River, NJ, USA (2004)
14. Deitel, P., Deitel, H.: Deitel Developer Series AJAX, Rich Internet Applications, and Web Development for Programmers. Prentice Hall Press, Upper Saddle River, NJ, USA (2008)
15. Katz, M.: Practical RichFaces. Apress, Berkely, CA, USA (2008)
16. Cecchet, E., Marguerite, J., Zwaenepoel, W.: Performance and scalability of ejb applications. In: OOPSLA '02: Proceedings of the 17th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications, New York, NY, USA, ACM (2002) 246–261
17. Mckenzie, C.W.: Hibernate Made Easy: Simplified Data Persistence with Hibernate and JPA (Java Persistence API) Annotations. PulpJava (2008)
18. Fleury, M.: The Official JBoss Development and Administration Guide. Pearson Education, Harlow (2002)
19. Pressman, R.S.: Engenharia de Software. McGrawHill, Rio de Janeiro, RJ (2002)
20. Lee, H.K., Malkin, T., Nahum, E.: Cryptographic strength of ssl/tls servers: current and recent practices. In: IMC '07: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement, New York, NY, USA, ACM (2007) 83–92