

## ODONTOSYSTEM - SOFTWARE PARA CONSULTÓRIO ODONTOLÓGICO

**Bárbara Batista Alves<sup>1</sup>; Gisele Pereira Borges<sup>1</sup>; João Paulo Alves<sup>1</sup>; Kéllita Martins Costa<sup>1</sup>; Marcos Evaristo Oliveira Costa<sup>1</sup>; Roger Amandio Luz<sup>2</sup>; Hulgo Leonardo Jacinto Andrade<sup>3</sup>**

<sup>1</sup>Graduando em Sistemas de Informação, pelo Instituto Luterano de Ensino Superior de Itumbiara ILES – ULBRA; Av. Beira Rio, 1.001 – Bairro Nova Aurora – CEP 75.523-200; e-mail: barbara0304941517@hotmail.com; giselepereiraborges1991@gmail.com; joaopauloalves1@outlook.com; kelliit@hotmail.com; marcosovaristololiveira@gmail.com. <sup>2</sup>Mestre em Processamento da Informação, pela Universidade Federal de Uberlândia, Av. João Naves de Ávila, 2121 - Santa Mônica, Uberlândia – MG, CEP 38408-100; e-mail: luzroger@gmail.com <sup>3</sup>Especialista em Inteligência Empresarial, pelo Instituto Luterano de Ensino Superior de Itumbiara ILES – ULBRA; Av. Beira Rio, 1.001 – Bairro Nova Aurora – CEP 75.523-200; e-mail: hulgoleo@gmail.com

**RESUMO** – O software OdontoSystem visa manipular informações de dados de seus pacientes de forma que facilite e agilize os serviços a eles prestados. O consultório odontológico entrevistado não possui um software que supra suas necessidades. Dentro desta problemática e fazendo uma análise de requisitos foram identificados alguns problemas como: falta de espaço físico para armazenar a documentação dos pacientes, agendamento de horários mal organizados, falta de controle financeiro, dificuldade para elaborar um controle de pacientes de forma eficiente, dentre outros. Portanto, o objetivo deste projeto se baseou em métodos, ferramentas e autores da engenharia de software, para análise, levantamento e coleta de requisitos e utilizando também à interdisciplinaridade que ajudaram no desenvolvimento deste software. Utilizou-se a linguagem JAVA para desenvolvimento do OdontoSystem na plataforma NetBeans IDE 7.0.1, para a criação dos relatórios foi usada a ferramenta iReport versão 3.0.1 e para o armazenamento dos dados dos cadastros foi elaborado um banco de dados em SQLServer.

**Palavras-chave:** Engenharia de software, sistema, UML.

### INTRODUÇÃO

O desenvolvimento de um software odontológico foi uma proposta determinada no 2º período do curso de Sistemas de Informação, tema este escolhido por nossos professores e orientadores. Apanhado o tema e toda a proposta deste software, deu-se início ao planejamento, documentação, pesquisa e desenvolvimento do projeto.

O objetivo principal é a manipulação de informações dos pacientes para facilitar e agilizar os processos e atividades de um consultório odontológico, auxiliando no controle de agendamentos de consultas e orçamentos, contando

com cadastro de pacientes, funcionários e fornecedores. O sistema possui cadastros de paciente, funcionário e fornecedor, controle financeiro, relatórios e consultas.

Para a fundamentação da documentação deste sistema foram utilizados conceitos de engenharia de software, engenharia de requisitos funcionais e não-funcionais – hardware, software e segurança -, modelagem de sistemas, diagramas de caso de uso, caso de uso expandido, diagrama de classe.

A construção do sistema iniciou com a elaboração do banco de dados, depois a construção das telas do sistema e relatórios.

Este software dispõe de diversas funcionalidades, foi especialmente projetado para proporcionar, de forma integrada, maior eficiência e controle das atividades clínicas, administrativas e financeiras de um consultório.

### METODOLOGIA

O projeto foi desenvolvido no Instituto Luterano de Ensino Superior de Itumbiara – GO. Para fundamentação teórica foram referenciados autores como BEZERRA (2007), SOMMERVILLE (2011) e PRESSMAN (2011). O software OdontoSystem foi desenvolvido em linguagem JAVA utilizando a plataforma NetBeans IDE 7.0.1 e armazenado em um banco de dados SQLServer. Para a criação dos relatórios foi usada a ferramenta iReport versão 3.0.1

De acordo com Pressman (2011), os métodos de engenharia de software proporcionam os detalhes de “como fazer” para construir o software, os métodos envolvidos incluem: planejamento e estimativa do projeto, análise de requisitos de software e de sistemas, projeto da estrutura de dados, arquitetura de programa e algoritmo de processamento, codificação, teste e manutenção.

Segundo Sommerville (2011), requisitos de um sistema são descrições dos serviços que devem ser fornecidos por esse sistema e as suas restrições operacionais. Neste projeto foram realizadas levantamento de requisitos funcionais, não funcionais, de software, hardware e segurança exemplificado na figura 1.

A primeira atividade da engenharia de requisitos corresponde à atividade de aquisição, também conhecida por levantamento e/ou elicitação de requisitos. De acordo com (Bezerra, 2007) a etapa de levantamento de requisitos corresponde a:

Compreensão do problema aplicada ao desenvolvimento de software. O principal objetivo do levantamento de requisitos é que usuários e desenvolvedores tenham a mesma visão do problema a ser resolvido. Nessa etapa, os desenvolvedores, juntamente com os clientes, tentam levantar e definir as necessidades dos futuros usuários do sistema a ser desenvolvido. Essas necessidades são geralmente chamadas de requisitos. (Bezerra, 2007).

Durante o levantamento de requisitos, a equipe de desenvolvimento visita empresas do ramo odontológico e tenta levantar as necessidades dos futuros usuários do sistema, bem como realizar estudos em sistemas existentes e documentos a fim de compreender o problema de deverá ser automatizado.

Requisitos funcionais são diretamente ligados à funcionalidade do software, o que o sistema deve prover, portanto neste projeto os requisitos principais foram: cadastro de pacientes, funcionários e fornecedores; controle financeiro (contas a pagar e contas a receber); consulta do histórico do paciente; exibir diagnóstico dos pacientes e ocorrências de atendimentos; agendamento de consultas; ficha de anamnese (entrevista realizada ao seu paciente, que tem a intenção de ser um ponto inicial no diagnóstico de uma doença); sistema de orçamentos, ou seja, uma cotação para o paciente.

Requisitos não funcionais são requisitos que expressam restrições que o software deve atender ou qualidades específicas que o software deve ter, dividido em três partes principais, os quais são: requisitos de software com gerenciador de banco de dados SQL Server, linguagem de programação JAVA, S.O. Windows XP ou Windows 7; requisitos de hardware, processador Dual Core, memória RAM 2GB, HD 320GB;

requisitos de segurança, controle de acesso por login e senha.

O modelo de processo de software utilizado foi RUP (Rational Unified Process) conforme mostrado na figura 2, é um exemplo de modelo de processo que foi derivado do trabalho sobre UML e do Processo Unificado de Desenvolvimento de Software associado (RUMBAUGH, 2006), por isso é um processo genérico que possui a meta de alcançar a qualidade, tempo e custos previstos. Para a modelagem dos diagramas de Use case, Use case expandido, diagramas de classe e outros, utilizamos as regras definidas pela UML (Unified Modeling Language), uma linguagem padrão para modelagem orientada a objetos e permite aos desenvolvedores visualizarem os produtos de seu trabalho em diagramas padronizados, e é muito usada para criar modelos de sistemas de software.

## RESULTADOS E DISCUSSÃO

Definido os métodos e ferramentas a serem utilizados, foram determinadas algumas métricas a serem seguidas neste projeto. A proposta de desenvolvimento do software visa manipular informações de dados de seus pacientes de forma que possa facilitar e agilizar os serviços a eles prestados. Através do sistema é possível agendar horários para a consulta, emitir relatórios com o diagnóstico dos pacientes e das consultas realizadas, orçamentos, bem como um cadastro para cada paciente, fornecedores, cirurgiões dentistas e demais funcionários. O software possui um controle financeiro atualizações de entradas (contas a receber) e saídas (contas a pagar).

Além disso, os procedimentos feitos pela endodontista além dos requisitos citados de cadastros possuem um histórico onde constarão os procedimentos e medidas feitas em cada dente e a data do atendimento, caso o paciente apresente alguma lesão é feito o agendamento do retorno para continuação do tratamento após a data determinada pela mesma, o cadastro de cada paciente possui um campo para o preenchimento da ficha de anamnese (um questionário sobre a saúde do paciente).

Depois de escolhidos os materiais e métodos que seriam utilizados o sistema começou a ser elaborado pela parte de desenvolvimento do banco de dados, depois foram criadas as telas de cadastros, consultas, menu principal, login, e relatórios.

Através de pesquisas e entrevistas feitas em consultórios odontológicos, foi observada a falta de

um sistema que permitisse a organização da documentação dos pacientes, cadastros, agendamentos de consultas que não são feitos de forma satisfatória e com a implantação do software OdontoSystem esses problemas seriam supridos.

### CONCLUSÕES

Este projeto de análise e desenvolvimento terá alcançado seu objetivo quando for implantado em uma clínica especializada, e os testes necessários para a constatação do pleno funcionamento do sistema e o treinamento dos usuários forem realizados.

Na conclusão do OdontoSystem, fundamentados por excelentes autores, com apoio de nossos orientadores e professores através da interdisciplinaridade pudemos conhecer, aprender mais para desenvolver este software e esperamos obter resultados satisfatórios quanto as expectativas e objetivos estabelecidos.

### REFERÊNCIAS

BLAHA, M., RUMBAUGH, J. **Modelagem e Projetos Baseados em Objetos com UML 2**, Elsevier, 2006.

BEZE  
RRA,  
Eduard

o. **Princípios de Análise e Projeto de Sistemas com UML**. Rio de Janeiro: Campus, 2007.

PRESSMAN, Roger S. **Engenharia de Software**. São Paulo: Makron Books, 2011.

SIERRA, Kathy. **Use a cabeça - JAVA**, 2ªed. Rio de Janeiro: Alta Books, 2005.

SOMMERVILLE, Ian. **Engenharia de Software**, 9ª ed. São Paulo: Pearson Education - BR, 2011.

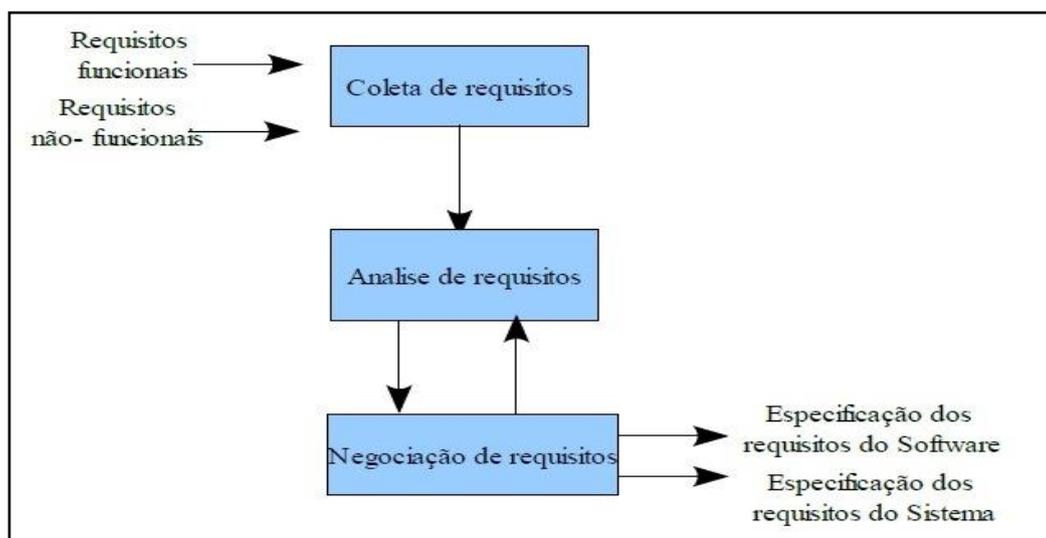


Figura 1. Processo de definição de requisitos do software

Fonte: [http://www.sidneyvieira.kinghost.net/abas/disciplinas/download/ESI/ES\\_RequisitosSoftware.pdf](http://www.sidneyvieira.kinghost.net/abas/disciplinas/download/ESI/ES_RequisitosSoftware.pdf)

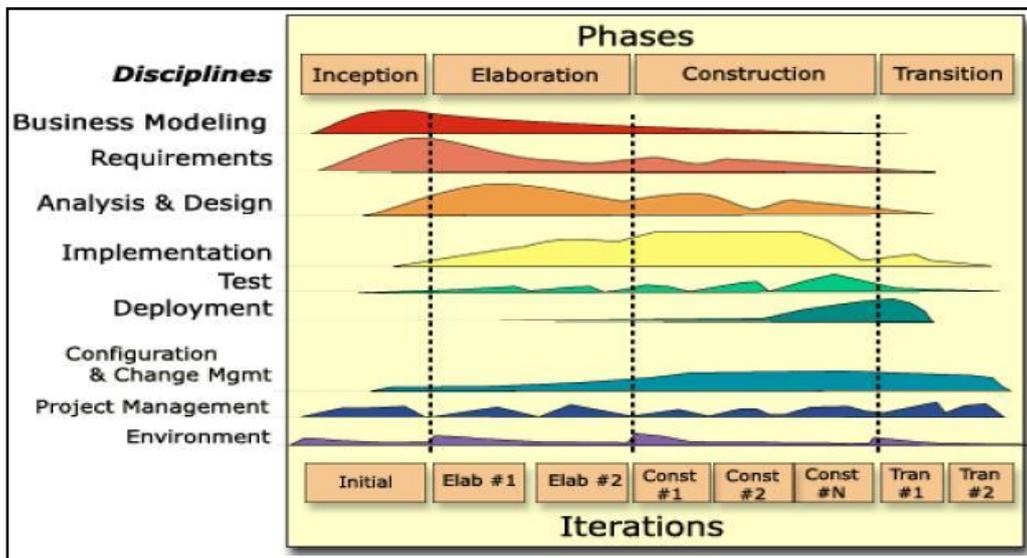


Figura 2. Modelo RUP (RationalUnifiedProcess)

Fonte: <http://qualidade-de-software.blogspot.com.br/2010/03/obtendo-qualidade-de-software-com-o-rup.html>