colloquium exactarum

DESENVOLVIMENTO DE UM SISTEMA WEB DE GERENCIAMENTO DE CONTEÚDO COMERCIAL AUTÔNOMO

DEVELOPMENT OF A WEB CONTENT MANAGEMENT SYSTEM AS COMMERCIAL

Marcel Aimar dos Santos Estácio[1]; Francisco Assis da Silva[2]; Francisco Virginio Maracci[2]; Mário Augusto Pazoti[2]

Discente da Faculdade de Informática da UNOESTE[1]; Docente da Faculdade de Informática da UNOESTE[2]

RESUMO - A tecnologia é uma das muitas ferramentas que as organizações usam para ajudar a resolver problemas de gestão empresarial. Entretanto, um sistema computacional corporativo moldado às regras de negócio para um microempreendedor autônomo pode não ser acessível, devido ao alto custo e tempo para desenvolvimento. A partir destas dificuldades, surgiu a idealização deste trabalho, que propõe o desenvolvimento de um sistema de gerenciamento de conteúdo (CMS) destinado ao gerenciamento de dados, a fim de possibilitar que um microempreendedor possa ter uma gestão eficiente de seu negócio e tomar decisões sob o foco de seu negócio. Um estudo de caso real foi utilizado como método para avaliar o comportamento do CMS desenvolvido, e com isso poder verificar e validar o gerenciamento de processos e manipulações de dados disponíveis no sistema desenvolvido.

Palavras-chave: CMS; Microempreendedor; Tecnologia; Gestão.

ABSTRACT – The technology is one among a lot of tools that organizations use to help them solve business management problems. However, a corporate computer system molded to the business rules of an independent micro entrepreneur cannot be accessible, due to the high cost and amount of time to the development. From these difficulties, the idealization from this research came up, proposing the development of a management content system tool (CMS) destinated to data manipulation, to allow a micro entrepreneur have an efficient management of his business and take decisions in focus of his business. A real case study was used as a method to measure the behavior of CMS developed, and thus be able to check and validate the process management and data manipulation available on the system developed.

Keywords: CMS; Micro Entrepreneur; Technology; Management.

Recebido em:20/05/2014
Revisado em: 25/07/2014
Aprovado em: 10/09/2014

1 INTRODUÇÃO

As empresas estão sempre tentando melhorar a eficiência de suas operações a fim de conseguir maior lucratividade. Segundo Rezende (2005), sistemas de informações são destinados às organizações que necessitam tomar decisões sob o foco de seu negócio empresarial ou de sua atividade pública. Laudon e Laudon (2007) certifica que o software está entre as ferramentas mais importantes para atingir altos níveis de eficiência, produtividade nas operações e apoio às decisões, já que de acordo com Hirama (2012), a falta de um software pode ocasionar a falha do controle financeiro. Entretanto, os custos, manutenções e prazos relacionados ao desenvolvimento de software podem não ser compatíveis com a real situação econômica de um microempreendedor autônomo. Rezende (2005) alega que o custo de manutenção do software abrange na maioria dos casos, o que representa uma enorme parcela do custo de desenvolvimento. Diante disso, para os microempreendores autônomos, surge a necessidade de um recurso de baixo custo, de fácil utilização e que os auxilie nas tarefas de organização e tomadas de decisão nos seus negócios. Uma alternativa, que já está sendo usada para diversos tipos de sites Web, são os CMS (*Content Management System*), entretanto, não possuem recursos para ambientes de auxílio ao gerenciamento de negócios. Foi com base nessa lacuna que surgiu a idealização deste trabalho, sendo proposto o desenvolvimento de um sistema de gerenciamento de conteúdo voltado para um microempreendor.

O sistema de gerenciamento de conteúdo (CMS) é uma ferramenta de software com o propósito de reunir todo o conteúdo criado para um site e gerenciá-lo mais eficientemente, eliminando a necessidade de usuários especializados em programação (HOLZSCHLAG, 2004). Como o CMS é uma ferramenta que deve ser instalada em um servidor, Holzschlag (2004) menciona que existem provedores de serviços de instalação de CMS disponíveis para todos os tipos de usuários. Esses, por sua vez, oferecem toda a tecnologia necessária para o serviço provido pelo CMS. Este tipo de serviço pode ser uma escolha razoável para muitas situações, devido os baixos custos e na maioria das vezes, um suporte que acompanha o serviço. Além da redução dos custos de infraestrutura e manutenção, os CMS possibilitam a criação, gerenciamento, distribuição, publicação e recuperação de informações corporativas, por meio de uma interface baseada em navegador, de fácil utilização (BOIKO, 2001).

Com o intuito de auxiliar microempreendedores autônomos a gerenciar seus negócios, foi desenvolvido um

CMS nomeado SIMA (Sistema para Microempreendedor Autônomo), que possibilite o gerenciamento do conteúdo comercial de forma dinâmica, a fim de fazer com que as funcionalidades do sistema, adaptam-se à realidade de cada organização. Conteúdo comercial se refere a um conjunto de dados que representam informações para uma determinada organização. Este conteúdo por sua vez é coletado, recuperado, processado, armazenado e distribuído por um sistema de informação, que tem a finalidade de transformar a informação em uma forma utilizável para a coordenação de fluxo de trabalho de uma empresa, ajudando empregados ou gerentes a tomar decisões, analisar e visualizar assuntos. Os sistemas de informação fazem isso através de um ciclo de três atividades básicas: entrada, processamento e saída (LAUDON; LAUDON, 1999).

O desenvolvimento de um software requer uma grande quantidade de tempo para se coletar todos os dados, preparar os documentos de especificação, proceder às avaliações formais e executar a completa implantação, com o risco de inibição do roteiro, fazendo com que etapas sejam repetidas, documentações refeitas, aumentando o tempo e custo do desenvolvimento (REZENDE, 2005). Com o gerenciador de conteúdo é possível criar todo o controle de entrada, processamento e saída, feito pelo sistema de informação, de forma prática e acessível. Com as informações necessárias de entrada, é possível elaborar formulários, relatórios, e consultas de forma dinâmica.

As operações realizadas por um sistema projetado especificamente para cada tipo de organização ainda depende de um profissional especializado. Desenvolver as tarefas básicas de um sistema que consiste em gerenciar um conjunto de informações (cadastros, módulos de controles, consultas e relatórios), demasiadamente grande. Com a utilização do CMS, desenvolvido desse trabalho, é possível atingir organizações de diversos ramos de atividades e gerenciar suas tarefas sem a necessidade de um profissional de computação. Utilizando uma linguagem de quarta geração, o CMS consegue acessar, na base de dados, os atributos de entrada que são informados pelo usuário durante a instalação do sistema e gerar as interfaces necessárias para que o empreendedor tenha controle de suas informações e consiga gerenciar seu negócio de forma mais prática e eficiente.

As demais seções deste trabalho estão organizadas da seguinte maneira: na Seção 2 são apresentados os trabalhos relacionados com o projeto desenvolvido; na Seção 3 é apresentada a arquitetura básica do funcionamento do SIMA, a estrutura e os métodos utilizados na construção de um

gerenciador de conteúdos e algumas particularidades do SIMA; a Seção 4 é composta por um experimento baseado em um estudo de caso; a Seção 5 apresenta os resultados obtidos e utilização do SIMA no estudo de caso; e por fim, na Seção 6 encontra-se a conclusão e os trabalhos futuros deste trabalho.

2 TRABALHOS RELACIONADOS

Esta Seção tem por finalidade apresentar alguns dos trabalhos relacionados a este trabalho proposto. São destacadas algumas das principais contribuições relacionadas ao contexto deste trabalho.

Chagas, Carvalho e Silva (2008) apresentam uma análise denominada "Um estudo sobre os sistemas de gerenciamento de conteúdo de código aberto", que serve para classificar os CMS mais populares do mercado, dentre eles: CMS Made Simple; Drupal; Ez Publish; Joomla; Liferay Journal; Mambo; Plone e Typo3. Os autores abordam desde as funcionalidades dos CMS, processo de instalação, até seus pontos negativos e positivos. Utilizando como critérios de avaliação a usabilidade, disponibilidade, segurança, suporte, dentre outros, os autores apresentaram que a escolha de um CMS que atenda os objetivos organizacionais é um processo trabalhoso e que a escolha depende dos aspectos definidos pelas organizações. Caso a organização quisesse desenvolver o seu próprio CMS, a solução seria custosa e de processo demorado, e ainda correria o risco do produto não possuir a qualidade desejada.

No trabalho de Mrack, Pimenta e Moreira (2004) o principal objetivo foi automatizar as interfaces de usuário. Para exemplos de iniciativas que utilizam esse paradigma, podem ser citadas as ferramentas: JANUS; TRIDENT; TADEUS; e GENIUS (SCHLUNGBAUM; ELWERT, 1996). Com o foco principal nas interfaces CRUD, a ferramenta denominada Merlin se propôs a gerar interfaces em modo de execução, facilitar as ligações para regras de negócio, reuso de padrões de mercado e ambiente multiplataforma.

3 DESCRIÇÃO DO CMS

Para o gerenciamento das informações e iteração com o usuário através de um sistema de software, é necessário além dos levantamentos de requisitos, criar a modelagem de dados, elaborar algoritmos para realizar as operações e a comunicação com o gerenciador de banco de dados. O SIMA é um gerenciador de conteúdos com a finalidade de gerar todas essas tarefas de forma automatizada. O SIMA faz o controle de ações por meio de grupos de trabalho e níveis de usuários. Um usuário comum está restrito às permissões que a ele são

concedidas. O usuário administrador tem permissão para gerenciar grupos e usuários. Somente um usuário do nível Super Administrador tem total controle sobre o sistema e tem permissões de criar, editar e excluir um gerenciamento.

Um novo gerenciamento é criado a partir das informações disponibilizadas pelo usuário. Com essas informações, a aplicação persiste os dados no banco de dados e cria uma tabela referente ao gerenciamento. Para obter alguma informação após o gerenciamento criado, o usuário realiza uma consulta utilizando a aplicação, e esta faz uma pesquisa ao banco de dados que retorna o conteúdo referente ao seu nível de usuário. O esquema mostrado na Figura 1 mostra o funcionamento básico da ferramenta.

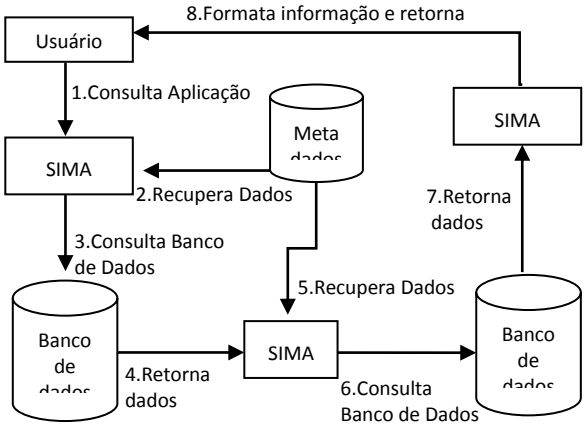

Figura 1. Esquema básico do funcionamento da ferramenta.

Conforme ilustrado na Figura 1, o usuário faz uma consulta na aplicação (1), e esta por meio das informações dos metadados[1] (2) faz uma consulta ao banco de dados solicitando os dados responsáveis pela estrutura do gerenciamento (3). Com os dados do gerenciamento retornados (4), com as informações dos metadados (5) a aplicação faz uma nova consulta ao banco de dados (6), agora solicitando o conteúdo da tabela referente ao gerenciamento. Os dados são retornados para a aplicação (7), que formata a informação corretamente e apresenta ao usuário (8).

3.1 GERENCIADOR DE CONTEÚDOS

Um CMS é um sistema complexo constituído por muitas partes que interagem entre si, formando um macro processo. Segundo Pressman (2011) e Boiko (2001), um CMS gerencia o conteúdo para o usuário por meio das invocações de três subsistemas integrados, que são: subsistema de coleção, subsistema de gestão e um subsistema de publicação.

- Subsistema de coleção: Os conteúdos são extraídos dos dados e de informações que devem ser criados ou adquiridos por um gerador de conteúdo. O subsistema de coleção abrange todas as ações necessárias para criar e/ou adquirir conteúdo e as funções técnicas que serão necessárias para converter conteúdo de

[1] Os metadados são dados que descrevem dados ou um conjunto de dados, geralmente, definido de forma clássica como “dados sobre dados” (NISO, 2004).

maneira que possa ser representado por uma linguagem de marcação, e organiza os conteúdos em pacotes que podem ser efetivamente mostrados no lado do cliente (PRESMAN, 2011). No caso deste trabalho, o SIMA oferece um ambiente para a produção da informação, os serviços de publicação produzem os formulários em HTML para inserir o conteúdo, que deverão ser convertidos para corresponder ao teor das normas aceites pelo sistema.

• Subsistema de gestão: Uma vez existindo o conteúdo, ele deve ser armazenado em um repositório apropriado do objeto de conteúdo, e objetos de conteúdo relacionados. Portanto, o subsistema de gestão implementa um repositório para abranger elementos de banco de dados de conteúdo, recursos de banco de dados e funções de gestão de configuração. Além desses elementos, o subsistema de gestão implementa uma função de administração que abrange os metadados e regras que controlam a estrutura global do conteúdo e a maneira pela qual é suportado (PRESMAN, 2011). Deste modo, este subsistema influencia todas as partes que constituem um CMS: Administração (Responsável pela configuração dos parâmetros e estrutura do CMS), repositório (Conjunto de bases de dados, diretórios de ficheiros, e outras estruturas que armazenam o conteúdo do sistema, assim como, quaisquer outros dados associados aos CMS) e fluxo de trabalho (Responsável por coordenar, programar, e fazer cumprir horários e tarefas). No CMS desenvolvido neste trabalho, os dados de configurações de acesso ao banco de dados e demais parâmetros, são armazenados em arquivos XML (*eXtensible Markup Language*), arquivados em diretórios do servidor. A estrutura do CMS e dados de configuração das páginas e conteúdo são armazenados no sistema de gerenciamento de banco de dados, MySQL, baseado no modo de normalização da terceira forma normal (3FN), afim de reprimir campos multivalorados e redundância de dados detectadas pelas demais formas normais (HEUSER, 2009). O CMS foi desenvolvido na linguagem de programação PHP (*Personal Home Page*), baseando-se no recurso de orientação a objetos (*Object-Oriented Programming* - OPP).

• Subsistema de publicação: O sistema de publicação é responsável pela saída dos conteúdos do repositório e automaticamente criar as próprias publicações. O conteúdo deve ser extraído de um repositório, convertido para uma forma que seja conveniente para a publicação e formatado de maneira que possa ser transmitido aos navegadores do lado do cliente. O subsistema de publicação executa essas tarefas usando uma série de *templates*,

que fazem a ponte da diferença entre conteúdos neutros no repositório e as necessidades de uma determinada publicação (BOIKO, 2001). Os *templates* guiam a criação e publicação a partir do conteúdo armazenado no repositório. Usam uma linguagem proprietária ou aberta para especificar a lógica da construção da publicação. No caso deste trabalho, as páginas Web criadas são construídas de forma dinâmica a partir dos conteúdos dos repositórios. Neste caso, o utilizador faz a solicitação para o servidor, os serviços de publicação carregam o *template* e exibem a página ao utilizador, empregando a linguagem de marcação HTML, com os padrões de organização do W3C.

Como citado anteriormente, o CMS é responsável pela coleção, gestão, e publicação de fragmentos de informação. De uma forma sistemática, esse processo pode ser representado como a ilustração da Figura 2.

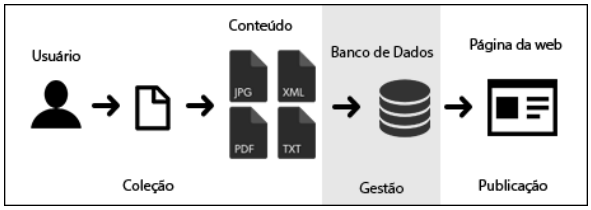

Figura 2. CMS visto como um processo.

A Figura 2 mostra como a informação é recolhida através do subsistema de coleção que a transforma em conteúdos. Após isso, os conteúdos passam por um sistema de gestão, sendo uma espécie de base de dados ou repositórios, que armazena estes componentes. O subsistema de publicação desenha os conteúdos para "fora" do subsistema de gestão e transforma-os em publicações para a Web.

3.2 ESTRUTURA DO SIMA

O SIMA trabalha sob informações disponibilizadas pelo usuário com o uso de formulários. Com base nessas informações, entidades, atributos e relacionamentos são gerados no banco de dados de forma dinâmica, para posteriormente serem utilizadas em operações CRUD[2].

O acesso ao gerenciador de banco de dados também é disponibilizado pelo usuário durante a instalação do SIMA. Com as informações disponibilizadas, a aplicação persiste os dados em um XML, ilustrado na Figura 3, que servirá como base para os acessos ao gerenciador do banco de dados durante a utilização do sistema.

[2]CRUD (*Create, Read, Update e Delete*): criar, consultar, atualizar e remover representam as quatro operações básicas sobre unidades de informação (WAZLAWICK, 2011).

```
1. <?xml version="1.0" encoding="ISO-8859-1"?>
2. <BancoDados>
3.   <Servidor>Nome do servidor</Servidor>
4.   <Nome>Nome do banco de dados</Nome>
5.   <Usuario>Usuário do banco de dados</Usuario>
6.   <Senha>Senha do banco de dados</Senha>
7.   <PrefixoTabela>Prefixo das tabelas</PrefixoTabela>
8.   <PrefixoAtributo>Prefixo dos atributos</PrefixoAtributo>
9. </BancoDados>
```

Figura 3. Estrutura do XML com dados de acesso a base.

O XML apresentado na Figura 3 segue os padrões da W3C. A primeira linha do documento é uma declaração XML e deve sempre ser incluída, pois define a versão XML do documento e a codificação utilizada. Neste caso está sendo especificada a versão 1.0 e codificação ISO-8859-1 da XML. Na linha 2 o bloco do elemento “BancoDados” é iniciado e seu conteúdo é constituído pelo nome do servidor do banco de dados (Linha 3), nome do banco de dados (Linha 4), nome do usuário do banco de dados (Linha 5), senha para o acesso ao banco de dados (Linha 6), prefixo que será inserido no nome das tabelas que o SIMA gerar automaticamente (Linha 7) e prefixo dos atributos da tabela gerada (Linha 8).

Os prefixos nas tabelas e atributos são utilizados a fim de evitar que os nomes das tabelas geradas automaticamente pelo SIMA não coincida com as tabelas já utilizadas na modelagem. Os prefixos nos atributos da tabela também são necessários pelo mesmo propósito, pois atributos de controle são adicionados à tabela. As operações CRUD são automatizadas através de classes nativas da linguagem PHP e classes implementadas especificamente para o SIMA.

As classes implementadas são organizadas em três tipos: Classe de entidade, classe de negócio e classe de dados. Para cada classe de entidade, existe uma classe de negócio e uma classe de dados que a representa. A classe de entidade é composta pelos métodos “GET” e “SET” e variáveis privadas que identificam os atributos do objeto em questão. A classe de negócio é responsável pela validação das entradas e saídas de dados realizadas pela classe de dados, que por sua vez, é a classe que realiza as conexões com a base de dados e executa as operações CRUD a partir das informações atribuídas pelo método “SET” na classe de entidade.

Dentre as classes implementadas, a classe “Gerenciamento” destaca-se como a mais importante para a gestão da estrutura do SIMA. Como representada na Figura 4, essa classe é constituída de atributos que representam a estrutura das tabelas e colunas que foram geradas pelo usuário fazendo o uso do SIMA. A Figura 5 mostra as tabelas da base de dados do SIMA manipuladas pela classe “Gerenciamento”.

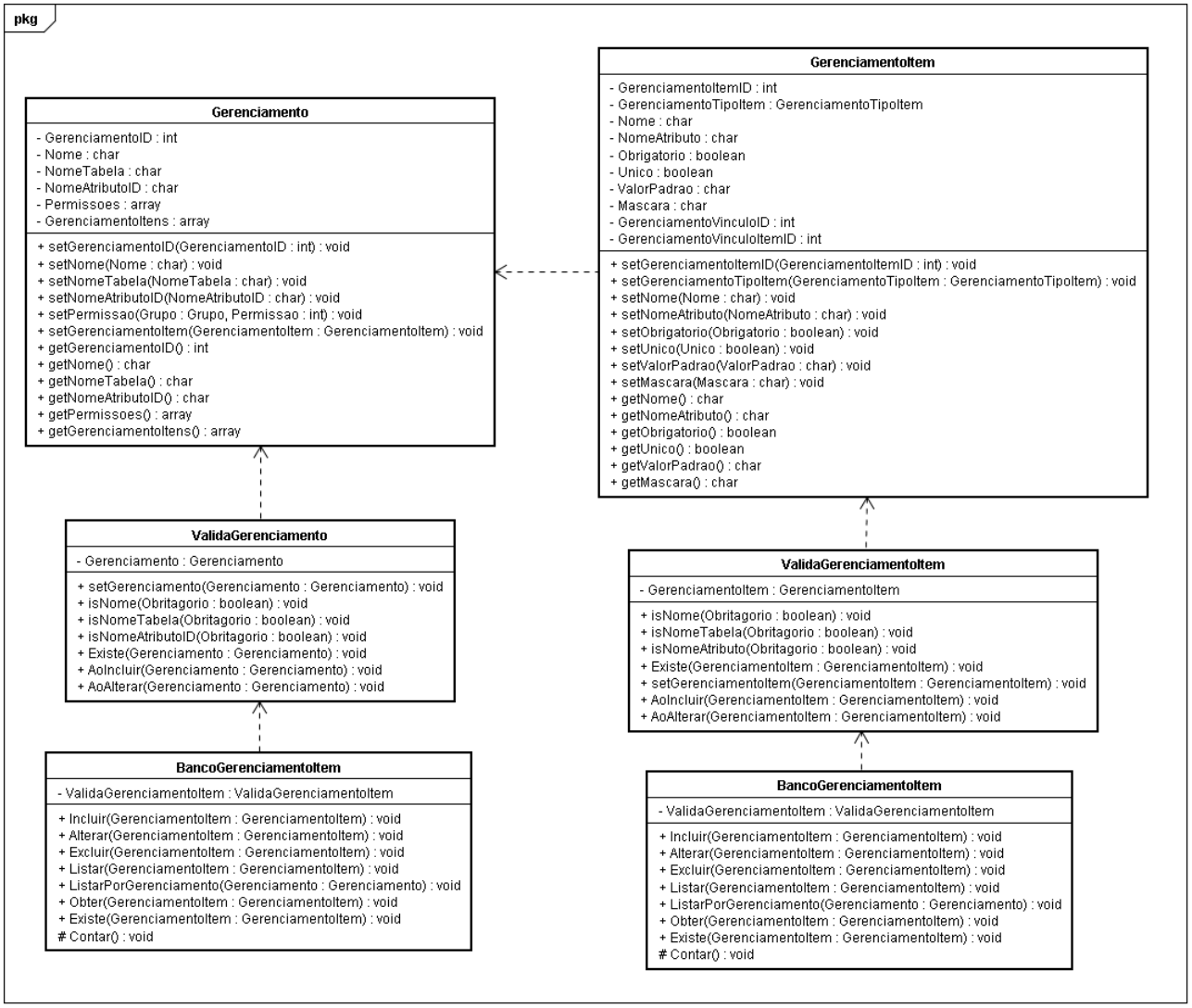

Figura 4. Diagrama de classes das classes “Gerenciamento” e “GerenciamentoItem”.

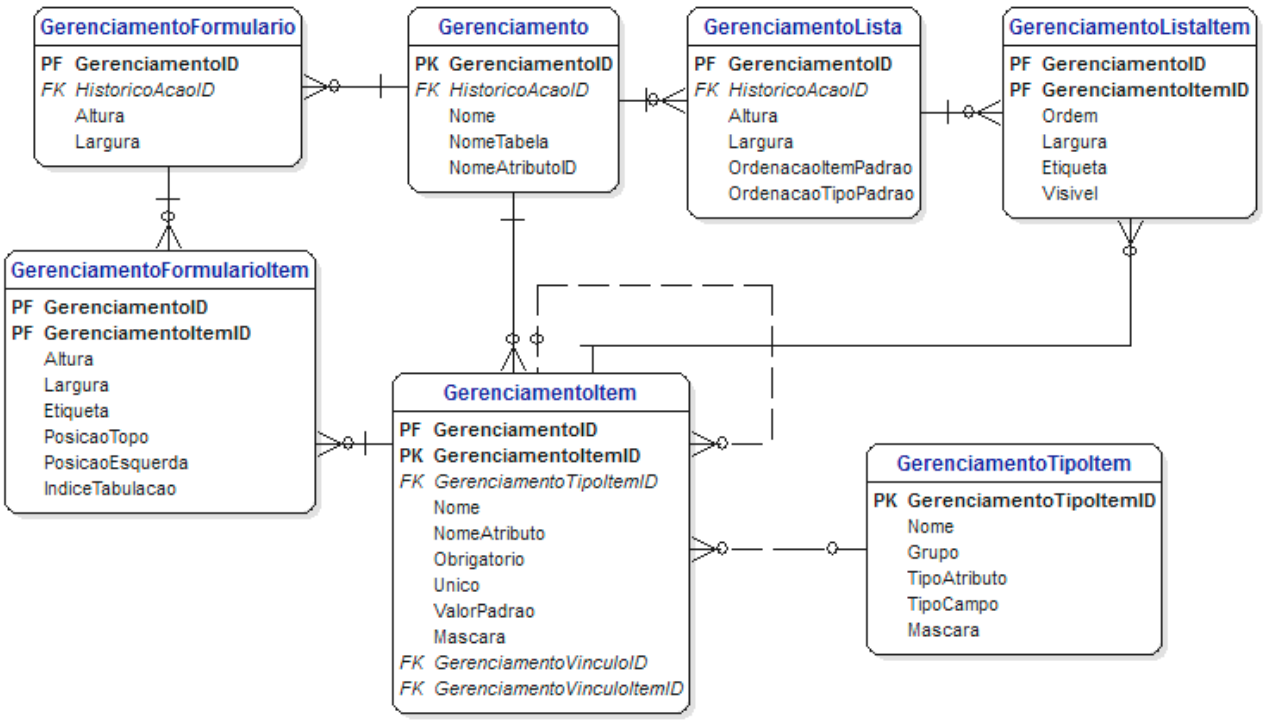

Figura 5. Tabelas da base de dados do SIMA.

A modelagem representada na Figura 5 mostra o conjunto de tabelas responsáveis por armazenar os dados necessários para gerar SQL[3] dinamicamente e realizar as operações CRUD ou retornar os dados necessários para o subsistema de publicação confeccionar as páginas destinadas ao navegador do cliente.

Em um exemplo onde o usuário gerencia "Clientes", a distribuição dos dados na modelagem da Figura 5 seria da seguinte maneira: Na tabela "Gerenciamento", seriam incluídos dados referente à tabela "Clientes", como o nome de exibição, nome da tabela que será representada e atributo de chave primária. Na tabela de "GerenciamentoItem" os campos referente aos atributos dos produtos que foram informados pelo usuário (Nome, Data de Aniversário, CPF, etc..) e seus tipos de dados. As tabelas "GerenciamentoFormulario" e "GerenciamentoFormularioItem" são responsáveis de armazenar as especificações necessárias para a camada de apresentação produzir os formulários para inserção e alteração dos dados do produto. Por fim, as tabelas "GerenciamentoLista" e "GerenciamentoListaItem" armazenam os dados necessários para camada de apresentação produzir a grade com os produtos cadastrados.

[3]SQL (*Structured Query Language*): é uma linguagem de consulta estruturada, declarativa e padrão para banco de dados relacional (COSTA, 2006).

Toda operação que modifique os dados das tabelas “Gerenciamento” e “GerenciamentoItem” resulta na alteração da estrutura da tabela que elas representam. No momento em que um novo item é adicionado na tabela “Gerenciamento”, uma tabela é criada para representar o conteúdo, e os campos da tabela são adicionados na mesma proporção que são inseridos na tabela “GerenciamentoItem”. Um exemplo dessa estrutura está definido na Figura 6.

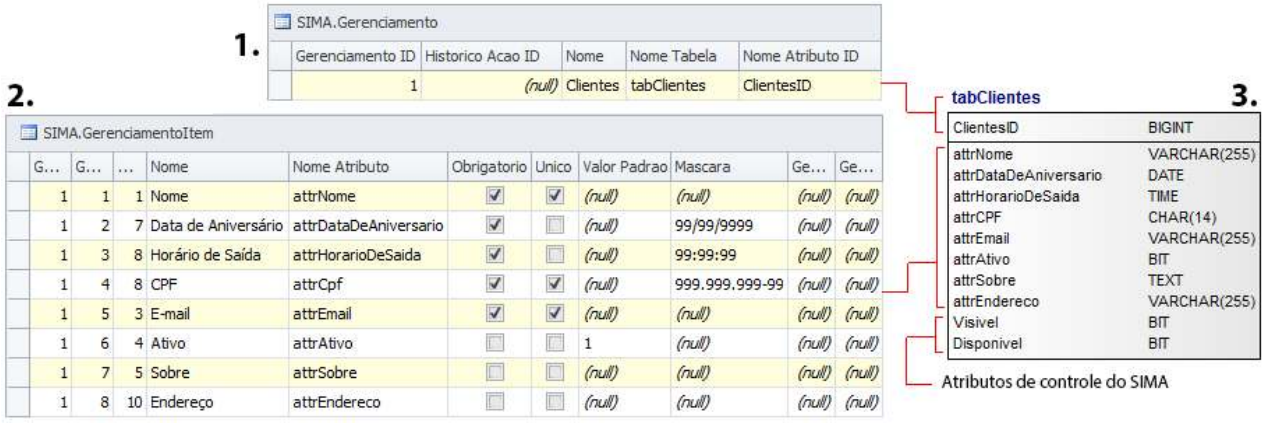

Figura 6. Exemplo de um gerenciamento criado pelo SIMA em que uma tabela foi criada com os dados do Gerenciamento.

A Figura 6 representa o resultado do exemplo citado anteriormente em que o usuário apresenta a necessidade de gerenciar “Clientes”. Com os dados informados pelo usuário, a aplicação persiste as informações nas tabelas 1 (“Gerenciamento”) e 2 (“GerenciamentoItem”) da figura, e cria a tabela 3 (“tabClientes”) que será responsável por armazenar o conteúdo referente aos dados do “Cliente”. Na tabela 1 da figura, o nome da tabela 3 e chave primária são armazenados, assim como o nome do objeto que ela representa. Na tabela 2 da figura, são armazenados as informações dos atributos da tabela 3.

A confecção das telas geradas dinamicamente pela aplicação tem como base a biblioteca jQuery[4], que utilizando os seus métodos de acesso ao servidor, quando solicitado pelo usuário, faz uma requisição AJAX (*Asynchronous Javascript And XML*) para a aplicação, que por si, recupera os dados do gerenciador de banco de dados e retorna ao jQuery via JSON[5] os valores necessários para a confecção das telas no formato HTML (*HyperText Markup Language*) e CSS (*Cascading Style Sheets*).

[4]jQuery é uma biblioteca JavaScript cross-browser desenvolvida para simplificar os scripts client side que interagem com o HTML. (jQuery, 2014)

[5]JSON (*JavaScript Object Notation*) é um subconjunto da notação de objeto de JavaScript com um formato leve para intercâmbio de dados computacionais. (JSON, 2014)

O design do SIMA é baseado no conceito de interfaces de usuário que incluem os aspectos interativos dos sistemas operacionais, com a utilização de símbolos gráficos (conhecidos como ícones) dinamicamente organizados em uma área de trabalho, que se trata de um ambiente gráfico adequado ao usuário, em que ele tem total liberdade em abrir janelas de aplicativos e efetuar operações básicas sobre elas e sobre o ambiente em si. A janela é o método utilizado para exibir a saída ou permitir entrada de dados no SIMA. Apresentadas como objetos bidimensionais e retangulares, organizados na área de trabalho, as janelas possuem certo conjunto de propriedades. Uma delas é a posicionamento, que pode ser alterado horizontalmente e verticalmente, outra o tamanho, que pode ser redimensionado horizontalmente, verticalmente ou para ambos. A Figura 7 mostra o *design* da aplicação SIMA baseado em ícones e janelas, organizados na área de trabalho.



Figura 7. Tela da aplicação SIMA.

4 EXPERIMENTO

Nesta seção é apresentado um experimento baseado em um estudo de caso real de um micro empreendedor individual, que opera no ramo de malharia e confecções e tem como principal atividade a fabricação de uniformes e camisetas personalizadas. Com sua sede em Presidente Venceslau, estado de São Paulo, a empresa conta com um faturamento anual de R$ 55.000. É uma empresa considerada familiar, pois nela trabalham duas pessoas, o empresário e seu filho. O empreendimento não consta de um sistema para automatizar os cálculos para elaboração do custo, preço final do produto e gerenciamentos de entrada e saída. Sem um controle adequado sobre os custos industriais, a empresa corria um sério risco de formar preços inadequados, causando o comprometimento de suas finanças. Com o propósito de auxiliar esta organização nas

tarefas de gerenciamento e tomadas de decisão, o SIMA foi implantado e configurado pelo usuário em um plano de hospedagem Web.

Neste caso, por se tratar de empreendimento familiar, o módulo de segurança disponível pelo SIMA através de permissões para grupos de usuário não foi utilizado. Para armazenar as informações dos fornecedores de estampas e camisetas foi criado um gerenciamento de "Fornecedores" na plataforma SIMA. Para evitar redundância de dados, o campo CNPJ foi imposto como único. Neste gerenciamento foram acrescentados os atributos necessários para reconhecer o fornecedor, saber sua real localização e dados de contato. Como cada fornecedor é responsável por distribuir os produtos necessários para a confecção das camisetas, um gerenciamento de "Matéria-Prima" foi elaborado. Para solucionar o fato de que um fornecedor disponibiliza uma ou mais matérias-primas, e uma matéria-prima pode ser cedida por um ou mais fornecedores, um gerenciamento de "Origem de Matéria-Prima" foi criado e relacionado com o gerenciamento de "Fornecedores" e "Matéria-Prima", acrescido dos campos referente a valores e prazos de entrega. A Figura 8 ilustra as tabelas geradas no sistema depois do relacionamento entre "Fornecedores" e "Matéria-Prima".

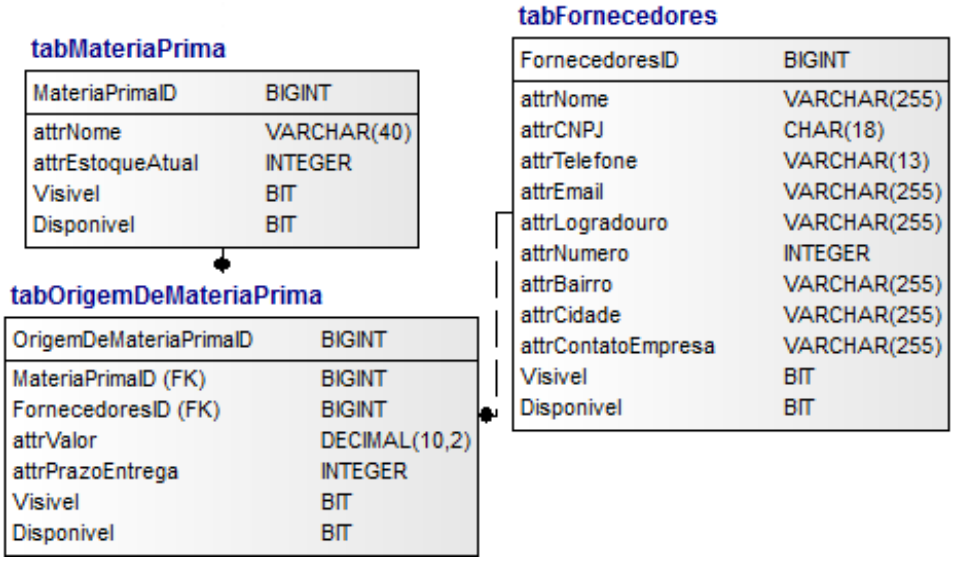

Figura 8. Tabelas criadas pelo SIMA para representar "Fornecedores", "Matéria-Prima" e "Origem de Matéria Prima".

O armazenamento dos resultados obtidos com o as matérias-primas foi adicionado no gerenciamento criado com o nome de "Produtos". Como cada produto pode receber várias matérias-primas, foi criado um gerenciamento "Confecção" que recebe como ligação o gerenciamento de "Matéria-Prima" e "Produtos", acrescido do campo para informar a quantidade utilizada. A fim de organizar as informações e dados de

contato, um gerenciamento de “Clientes” foi criado, constituído de vários atributos que possibilitam o armazenamento de todas as informações necessárias para manter o contato com o cliente. Para evitar redundância de dados, o campo CPF foi imposto como único. Por fim, para controlar as saídas e manter um histórico de movimentações, foi feito um gerenciamento denominado “Vendas”, que recebeu como relacionamento o gerenciamento de “Clientes” e “Produtos”. A Figura 9 ilustra a estrutura gerada pelo SIMA com base nos gerenciamentos criados pelo microempreendedor.

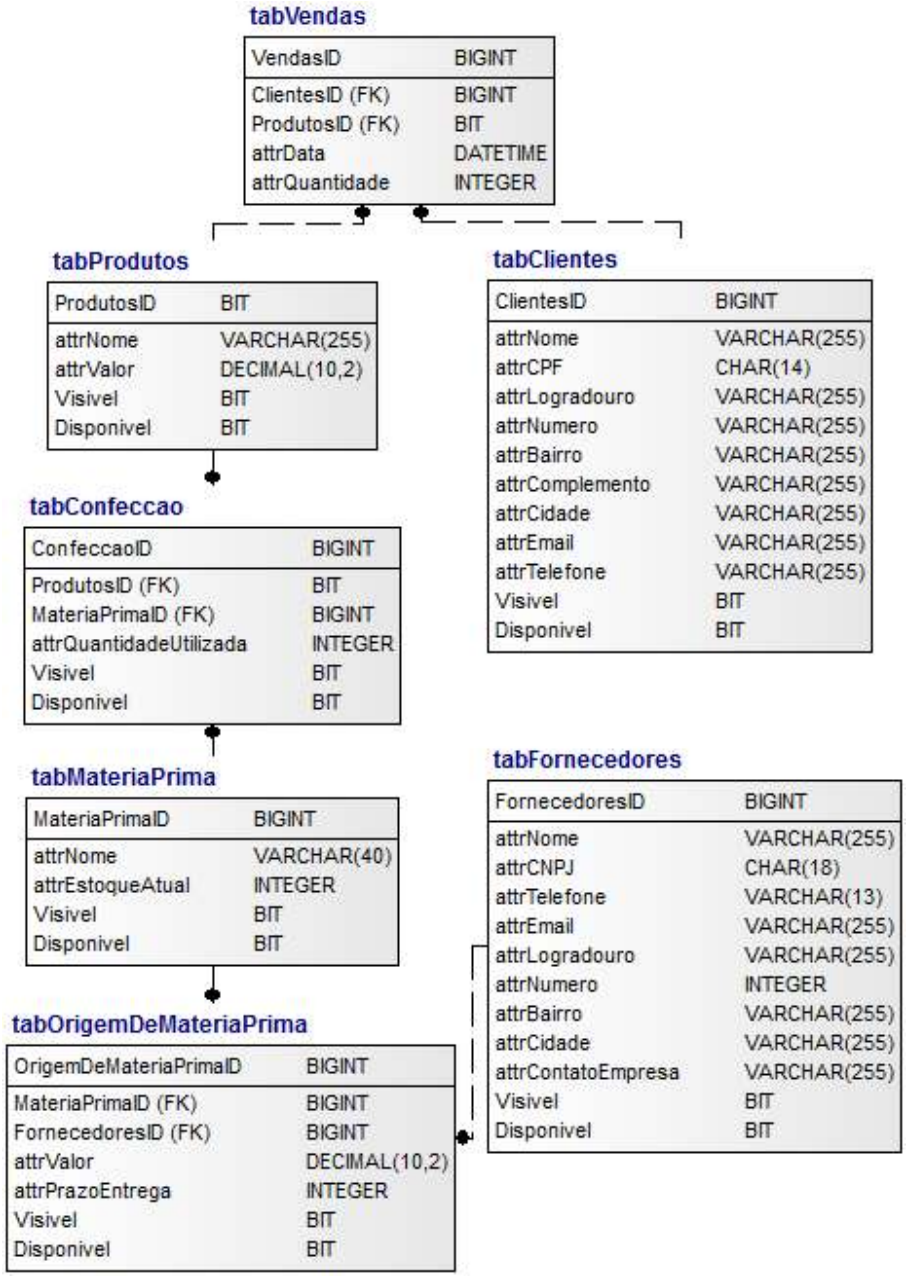

Figura 9. Estrutura final da base de dados criada pelo SIMA para a microempresa de confecções.

Com os gerenciamentos criados e estruturados, teve inicio o processo de cadastros. De inicio os gerenciamentos de “Clientes”, “Fornecedores” e “Matéria-Prima” foram os primordiais para o início do processo, pois é a partir deles que os outros gerenciamentos são estruturados. Em seguida os demais gerenciamentos foram alimentados, dando início ao processo de gestão do sistema.

5 RESULTADOS

Com a facilidade de utilização e o controle das informações automatizado, o SIMA proporcionou ao microempreendedor um aumento da qualidade da decisão e elevou seu poder identificar melhor seus clientes e saber quais as suas preferências, com base nas informações de vendas que são armazenadas no sistema. Com o melhor conhecimento da matéria-prima e demais gastos, a empresa pode analisar melhor as informações cruciais para se chegar a um preço de venda justo e lucrativo.

Antes da implantação do SIMA, o microempreendedor tinha dificuldades em manter um histórico de vendas e apontar as preferências e hábitos dos seus clientes. Atrasos de pedidos e inconsistência dos valores eram repentinos, em razão das entradas de matéria-prima e controle do estoque não usufruírem de uma gerencia, fazendo com que a quantidade de pedidos exceda a quantidade atual de matéria-prima.

Contudo, o SIMA proporcionou melhoras nos procedimentos da empresa, uma vez que o trabalho tornou-se mais formalizado e profissional e as tarefas diárias ficaram melhor organizadas.

6 CONCLUSÕES E TRABALHOS FUTUROS

Com esse trabalho foi possível demonstrar que diante da situação econômica de um micro empreendedor autônomo, um gerenciador de conteúdos pode ser uma forma de contribuir no crescimento e na administração da organização, com baixos custos de manutenção e prazos inferiores.

Por meio do resultado obtido pelo estudo de caso, pode-se avaliar como positivo o uso do SIMA, pois todas as atividades dificultosas foram organizadamente desenvolvidas, permitindo obter grande vantagem competitiva em relação aos concorrentes, pois o processo decisório estará mais focado e embasado em informações corretas e disponíveis em tempo hábil.

O SIMA tem como objetivos futuros o desenvolvimento de templates, para disponibilidade de alterações de formato de tela com melhores adaptações para dispositivos móveis e um painel para

soluções de ajuda às principais dúvidas obtidas pelos usuários.

REFERÊNCIAS

BOIKO, B**. Content Management Bible**. New York: John Wiley & Sons, 2001.

CHAGAS, F.; CARVALHO, C.L.; SILVA, J.C.. **Um estudo sobre os sistemas de gerenciamento de conteúdo de código aberto.** 2008. Disponível em: <http://www.inf.ufg.br/sites/default/files/uploads/relatorios-tecnicos/RT-INF_002-08.pdf>. Acesso em: 03 mai. 2014.

COSTA, R.L.C. **SQL:** guia prático. 2. ed. Rio de Janeiro: Brasport, 2006.

JQUERY. **jQuery write less, do more**. Disponível em: <http://jquery.com>. Acesso em: 03 mai. 2014.

JSON. **Introdução ao JSON.** Disponível em: <http://json.org/>. Acesso em: 03 mai. 2014.

HEUSER, C. **Projeto de Banco de Dados**. São Paulo. Bookman, 2009.

HIRAMA, K. **Engenharia de Software, qualidade e produtividade com tecnologia**. Rio de Janeiro: Elsevier, 2012.

HOLZSCHLAG, M.E. **250 Segredos para Web Designers**. Rio de Janeiro: Elsevier, 2004.

LAUDON, K. C.; LAUDON, J. P. **Sistema de Informação: com Internet**. 4ª edição. Rio de Janeiro: LTC – Livros Técnicos e Científicos Editora S.A., 1999.

LAUDON, K. C.; LAUDON, J. P. **Sistemas de Informação Gerencial**. 7ª edição. São Paulo: Pearson Prentice Hall, 2007.

MRACK, M.; PIMENTA, M.; MOREIRA, A.F. **Merlin:** interfaces CRUD em tempo de execução. 2004. Disponível em: <http://3layer.com.br/confluence/download/attachments/3571742/merlinFerramentasSbc2006-revisado-em6paginas.pdf>. Acesso em: 03 mai. 2014.

NISO. **Understanding metadata.** 2004. Disponível em: <http://www.niso.org/publications/press/UnderstandingMetadata.pdf>. Acesso em: 03 mai. 2014.

PRESSMAN, R.S. **Engenharia de software.** Uma abordagem profissional. 7. ed. Porto Alegre: Bookman, 2011.

REZENDE, D.A. **Engenharia de software e sistema de informação**. 3. ed. Rio de Janeiro: Braspost, 2005.

SCHLUNGBAUM, E.; ELWERT, T. Automatic user interface generation from declarative models. In CADUI'96, **Procedings…** 1996.

WAZLAWICK, R.S. **Análise e projeto de sistemas de informações orientados a objetos.** 2. ed. Rio de Janeiro : Elsevier, 2011.