

Diretrizes para criação de um padrão de desenvolvimento de sistemas de informação baseados em cots

Roosewelt Sanie Da Silva¹

¹Ciência da Computação – Universidade Presidente Antônio Carlos (UNIPAC)
Rodovia MG 368, KM 12 – Colônia Rodrigo Silva - Barbacena – MG
CEP: 36200–000

{rsanie@hotmail.com}

Resumo. Atualmente exige-se cada vez mais a implementação de sistema com maior qualidade e, com menor tempo, devido ao grande mercado de software existente, desta forma, esse artigo propõe diretrizes a serem seguidas, com intuito de ser padronizada a implementação de cots, e dessa forma torna-lo mais seguros e confiáveis e com maior qualidade, para que se possa ter um mercado com maior qualidade em sistemas para reuso.

1. Introdução

Com o passar dos anos, pode-se notar que a informatização está cada vez mais presente na vida do ser humano. E é quase impossível ter que fazer diversos trabalhos manualmente, como por exemplo, a logística de uma empresa.

A partir deste fato este artigo tem como objetivo a elaboração de diretrizes padrão de como elaborar um sistema (comercial) baseados em COTS: forma de desenvolvimento utilizando softwares prontos. COTS comercial off-the-shelf system (produto de prateleira produzido por terceiros), sistemas cots são aplicações completas que oferecem uma API (Application Programming Interface) [Boaventura,2004].

A construção de sistemas pela integração de sistemas cots, é uma estratégia de desenvolvimento viável para alguns tipos de sistemas, tais como sistemas de comércio, eletrônicos, telecomunicação, governo. [Boaventura,2004].

Atualmente muito se têm interessado por este método de desenvolvimento, como governo, indústria e comércio. O desenvolvimento de software de prateleira traz muitos benefícios, entre os quais redução de tempo, reduzir gastos e facilidade em implementar funcionalidades e maior satisfação.

A implementação consiste principalmente em produção de componentes competitivos, fáceis de interagir.

O desenvolvimento é de grande importância em resoluções de problemas complexos e extensos, pois, através deles se tem um trabalho final. Uma vez que parte da implementação já está pronta, restando apenas junta-las e tratar possíveis erros.

É possível encontrar vários sistemas prontos no mercado que podem ser integrados e tornar um sistema mais complexo, reduzindo o tempo de desenvolvimento e os gastos.

Uma questão muito importante é a confiabilidade destes cots encontrados no mercado, pois se foram implementados com sua devida atenção, não saberemos se terá garantia, confiabilidade, segurança e qualidade.

Outra questão importante é a responsabilidade do programador, pois uma vez não sabendo quem desenvolveu, se preocupou em tratar de modo correto as partes do desenvolvimento ou estabelecer uma regra para o sucesso da implementação.

Esta preocupação se deve ao fato de que uma vez um componente com “bugs” (defeito) ao integrar com o sistema, pela regra de herança, o sistema irá herdar estes defeitos.

É pensando nestes problemas, que será proposto regras para implementação de cots, e também será desenvolvido um programa seguindo estas regras passo a passo para mostrar sua usabilidade, para que se tenha um ótimo resultado.

Este artigo está organizado da seguinte forma: A seção 1 apresenta uma Introdução sobre o problema de um mal desenvolvimento de um sistema baseado em cots. Na seção 2 são apresentados as “diretrizes” a serem seguidas para padronização da implementação. Na seção 3 são apresentadas as ferramentas utilizadas no desenvolvimento e como eles se interagem. Na seção 4 é apresentado o Estudo de Caso, modelagem criação da base de dados, e a implementação. Na seção 5 são apresentadas algumas conclusões e também algumas considerações sobre futuro desenvolvimento de um sistema de informações baseado em cots.

2. Diretrizes

Para que tudo saia nos seus devidos lugares, a modernidade exige o uso obrigatório de um roteiro, uma diretriz a ser seguida, pois as mesmas auxiliam muito no desenvolver do trabalho, e foi pensando nisto que foram propostos as diretrizes para implementação de sistemas baseado em cots.

As seguintes diretrizes são propostas para que possam ser seguidas e assim ter uma melhor qualidade nos softwares desenvolvidos baseados em cots, em cada passo é abordado um estudo diferente, são eles:

1º- Estudo e entendimento do problema, no caso o software a ser implementado, capturar todas as informações possíveis para uma boa elaboração do projeto.

Consistem em ter o maior número de informações possíveis do software a ser implementado, para que se possa ter um domínio maior da aplicação e quais recursos estarão disponíveis na aplicação.

Neste passo é que identifica para que o software vai ser desenvolvido, quais as operações que deve conter o software, quais os tipos de usuários, que tipo de ambiente o software vai ser utilizado.

2º- Estudo e compreensão da linguagem padrão do domínio em questão.

Consiste em um estudo de linguagem padrão para que se possa escolher o melhor modo de se implementar e tratar o problema a ser resolvido, em outras palavras, são técnicas para trabalhar o problema, caso ela exista, pois pode acontecer de não existir uma linguagem padrão para determinado problema.

3º- Agrupar as características em comum com o sistema que será desenvolvido.

Consiste em separar tarefas em grupos, a serem trabalhadas no desenvolvimento para que se possa ter um domínio maior do que está sendo feito no momento.

Neste passo é que identificamos as tarefas que ficarão em determinado local no software, como por exemplo, cadastro. Seleciona o que vai ser cadastrado, para que seja trabalhado de uma só vez.

4º- Modelar o sistema.

Consiste em criar toda a documentação para a implementação do software, aqui se cria diagrama de classe entre outros documentos.

Neste passo, aplica se o software de modelagem para criar a relação entre as classe para que se possa ter domínio do funcionamento do programa.

5º- Avaliar a utilização de software livre para ser utilizado no desenvolvimento do sistema.

Consiste em selecionar as ferramentas de modo a reduzir gastos, tentando ao máximo possível optar pelo software livre por não ter que pagar licença de uso.

Neste passo, faz se uma pesquisa sobre as ferramentas disponíveis no mercado, para que seja escolhido a melhor para o desenvolvimento, desde que tenha conhecimento da ferramenta selecionada.

6º- Implementar o sistema, e colocá-lo em teste para possíveis alterações, caso algum passo anterior não tenha sido seguido corretamente.

Consiste em colocar o sistema em teste para que possam ser avaliados suas funções e seu desempenho para possíveis alterações.

Neste passo, o sistema é colocado em um estabelecimento para ser analisado seu comportamento em andamento de trabalho, para verificar possíveis erros tanto de operação quando de aparencia, onde serão tratado e corrigido em uma próxima versão.

3. Ferramentas a serem utilizadas em cots.

De acordo com a 5ª diretriz, anteriormente citada, parte dos componentes utilizados segue a idéia de softwares livres, ou seja, aplica-se ao máximo possível de softwares livre para implementação do sistema, de modo que, reduza os custos.

Entre muitos, foram selecionados alguns componentes, que são considerados cots, para integração do sistema.

1) SOFTWARE DE DADOS. Software gratuito para base de dados, onde, será criado toda estrutura do banco de dados do software de exemplo, MYSQL é uma linguagem para geração de base de dados, totalmente gratuita, ela fora desenvolvido para plataforma linux (*sistema operacional Livre*), porém hoje já se encontra versões para Windows, neste caso utiliza-se a versão para Windows, que se trata de versão, MYSQL 4.1.

2) SOFTWARE PARA CONTROLE DOS DADOS. Mysql administration, software gratuito com interface interativa para controle de usuários e manipulação da base de dados em MYSQL. Através dele, é possível detectar erros ou problemas na base de dados, este software funciona como uma solução preventiva de problemas e controle de acesso restrito a base de dados, tornando assim o sistema mais seguro.

3) FRONT_MYSQL. Software gratuito com interface interativa para criação de tabela e campos na base de dados de modo interativo. Objetivo deste software: agilidade em criação e alterações na base de dados de modo a ganhar tempo.

4) SOFTWARE DE MODELAGEM. Argo UML, Software gratuito, com interface interativa para geração de diagramas de classe. Entre outros diagramas, para auxílio no desenvolvimento do sistema.

5) LINGUAGEM VISUAL. Recomenda se o uso de uma linguagem visual por se de simples compreensão e facilidade de manuseio, existe muita linguagens visuais gratuitas, entre as quais, KYLIX.

No nosso caso foi utilizado delphi 7, uma linguagem visual não gratuita, pelo fato de se ter maior domínio e costume de trabalho.

6) COMPONENTES NATIVOS. Recomenda o uso de componentes nativos da própria linguagem, uma vez que já estão disponíveis. Por uma questão de habilidade e costume de trabalho, será utilizado neste projeto o componente ZEOS.

O uso das ferramentas citadas anteriormente é de grande importância para que se possa ter um bom resultado na implementação do sistema, pois elas interagem entre si, cobrindo os seguintes requisitos:

Segurança, manutenção, prevenção, padronização, alteração e qualidade.

A figura 1 mostra como os componentes interagem entre si, para o desenvolvimento do sistema, mostra cada processo ao qual o sistema passará até a sua finalização.

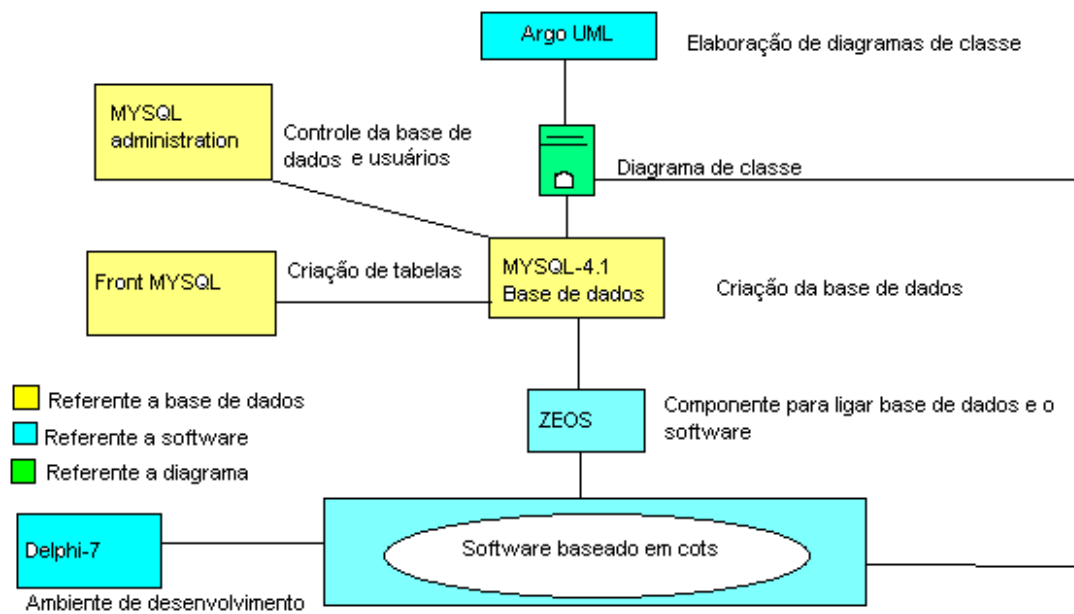


Figura 1- interação entre si das ferramentas

O Argo UML será utilizado para geração do diagrama de classe, depois de gerado este documento (diagrama de classes) cria-se a base de dados em MYSQL_4.1. O diagrama de classes é um documento ao qual será utilizado para o desenvolvimento do software e também para a base de dados .

No MYSQL fica toda a base de dados criada, através do MYSQL ADMINISTRATION controla-se se esta base de dados, com as seguintes operações: manutenção, verificação se está tudo certo e controle de usuários de acesso.

Através do FRONT MYSQL cria-se, altera-se, exclui-se tabelas, campos, no modo interativo (visual), facilitando e reduzindo o tempo gasto de trabalho.

Através do DELPHI 7, desenvolve-se a maior parte do programa (software) que será utilizado, e através do ZEOS faz-se a ligação (comunicação) entre o software e a base de dados ele é responsável pelo reconhecimento do banco de dados e a comunicação com o programa.

4. Estudo de caso

Para comprovar que as regras funcionam, foi implementado um sistema seguindo passo a passo todas diretrizes, O sistema desenvolvido foi um software de controle de materiais de construção .

E para isto foram estudados dois projetos de modelagem de materiais de construção: Um estudo de casos de sistema de informação [Filho, 2005] e uma abordagem do desenvolvimento de um sistema de controle de vendas de materiais de construção com utilização de RUP (Rational Unified Process) e UML (Unified Modeling Language) [Abreu, 2005].

Aplicando a primeira diretriz, foi selecionado o que há de comum entre os dois, para implementação de um novo software genérico, toda a modelagem foi feita orientado a objeto sendo que a implementação foi feita estruturada.

Depois de feito este processo, aplica-se a segunda diretriz, faz-se um estudo sobre linguagem padrões [Braga,2002] para que se possa escolher um modelo melhor para construção do diagrama de classes e depois implementar o sistema.

Aplicando a terceira diretriz, agrupou-se as informações a serem implementadas do sistema, para que se possa ter um maior controle e eficácia do sistema.

Aplicando a quarta diretriz, faz-se a modelagem, começa a partir das informações do sistema a ser implementado, como por exemplo, o que será cadastrado, o que será controlado, como o software deverá se comportar, o que será restrito, entre outras informações.

Para modelagem foi utilizada a ferramenta ARGO UML para geração do diagrama de classe ao qual servirá de base na construção do programa, na base de dados e a comunicação do usuário e o banco de dados.

A ferramenta ARGO UML possui um visual interativo de fácil compreensão, desenvolvida em Java e gratuita.

Argo UML é responsável pela elaboração de diagramas ao qual foi usado para implementação do sistema, um diagrama de classe depois de pronto tem a seguinte aparência, figura-2.

Diagrama de classes do sistema genérico de controle de materiais de construção.

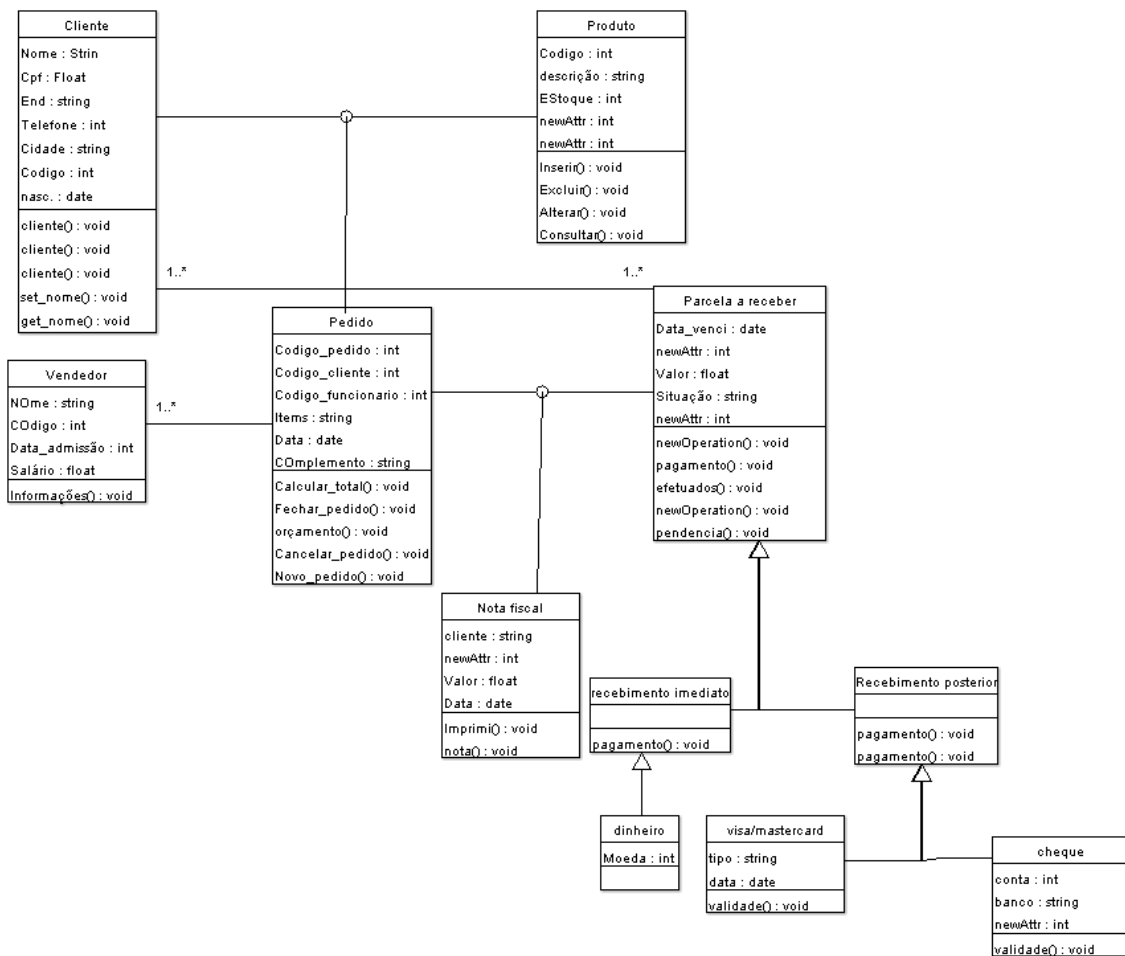


Figura-2 diagrama de classes

Após feito o diagrama de classes, cria-se o diagrama de interface, para que se possa ter melhor visão das operações que vai compõem o software, unindo os dois diagramas em um só, diagrama de classes com suas respectivas interfaces.

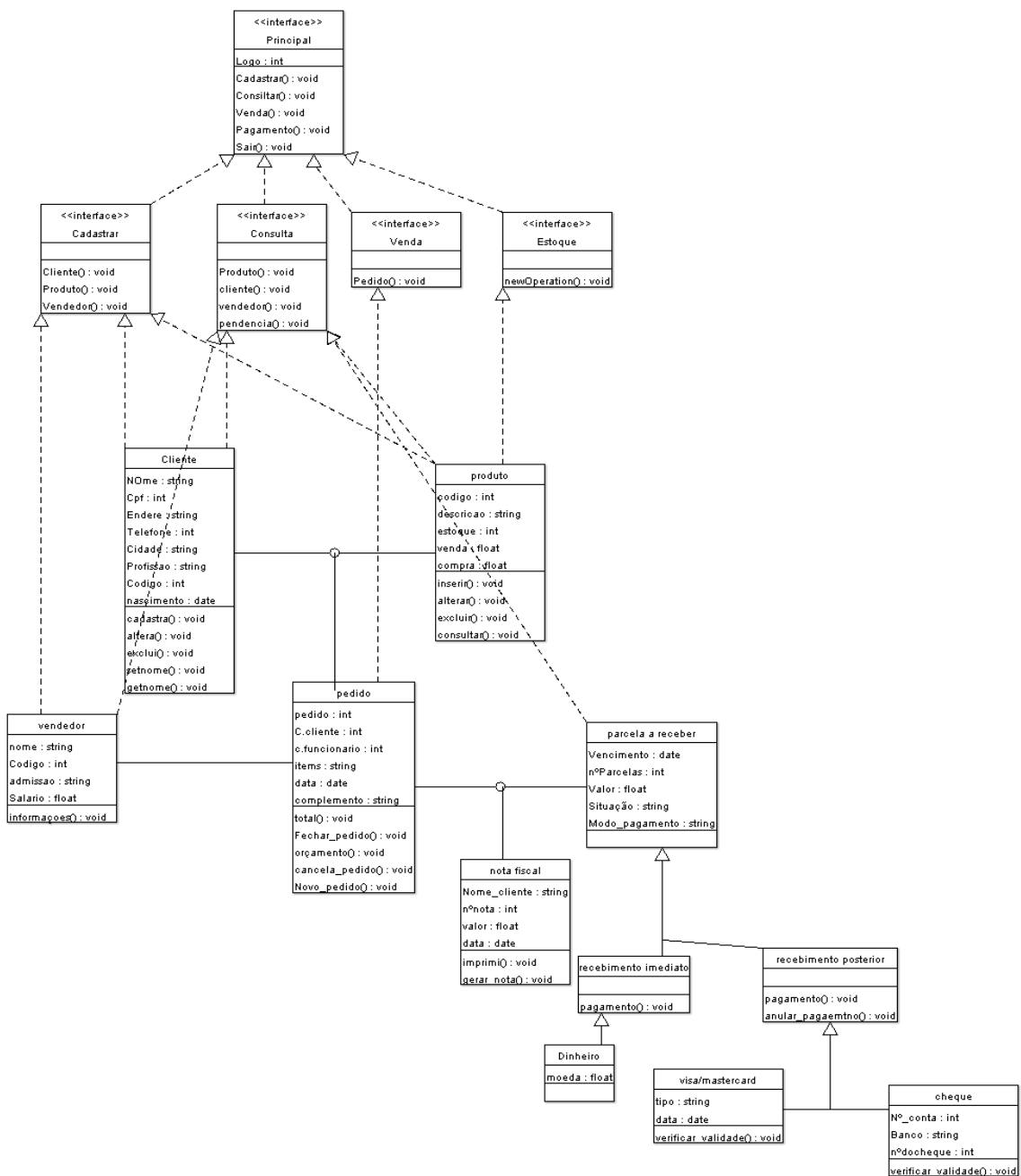


Figura-3 diagrama de interfaces

O diagrama apresenta as seguintes classes:

1º Cliente: responsável por cadastro e controle de todos os clientes presentes na loja.

2º Vendedor: responsável por cadastro e controle dos vendedores da loja.

3º Produto: responsável pelo cadastro e controle de mercadorias e preços dos produtos contidos na loja.

4º Pedido: onde é feita a venda para o cliente, controla preço orçamento total de compra e produtos comprados.

5º Parcelas a vencer: controle de pendências financeiras de compras passadas, contendo um histórico do cliente sobre o que falta para pagar.

6º Nota fiscal: contendo todos os dados dos produtos comprados e taxas para fim de fiscalização.

Depois de feito o diagrama de classe, se tem em mãos um documento ao qual será utilizado no desenvolvimento do software e de todo funcionamento do mesmo.

Aplicando a quinta diretriz, faz-se a avaliação de software livre, para redução de custos na implementação do sistema.

Aplicação da quinta diretriz consiste em seleção dos softwares utilizados para implementação.

O diagrama de classe, foi utilizado para criar toda estrutura da base de dados, no caso, MYSQL.4.1, apesar de que o diagrama não é especialmente desenvolvido para criação da base de dados, mas sim para o software.

Para criação de modo interativo, usa-se a ferramenta FRONT-MYSQL.

Ferramenta para manutenção e controle dos dados usa-se MYSQL administration.

Depois de criado a base de dados, resta agora, a construção do programa. E para isto, utiliza-se a ferramenta DELPHI 7, ferramenta visual de desenvolvimento modo estruturado.

A utilização do delphi 7 foi devido por se tratar de desenvolvimento para o sistema windows, sendo assim, optou-se pelo software não gratuito por uma questão de plataforma, porém recomenda-se a utilização do KYLIX por ser um software gratuito.

Para conexão do software com a base de dados, foi utilizado o componente que é adicionado ao delphi (não vem com ele) chamado ZEOS ACCESS.

Porém o delphi já tem um componente nativo chamado DBEXPRESS, componente ZEOS foi utilizado somente por questão de opção e também pelo costume de trabalho com o mesmo.

Após todos estes passos, obtém-se o software baseado em cots, no exemplo deste artigo, trata-se de um software de controle de materiais de construção , figura 4.

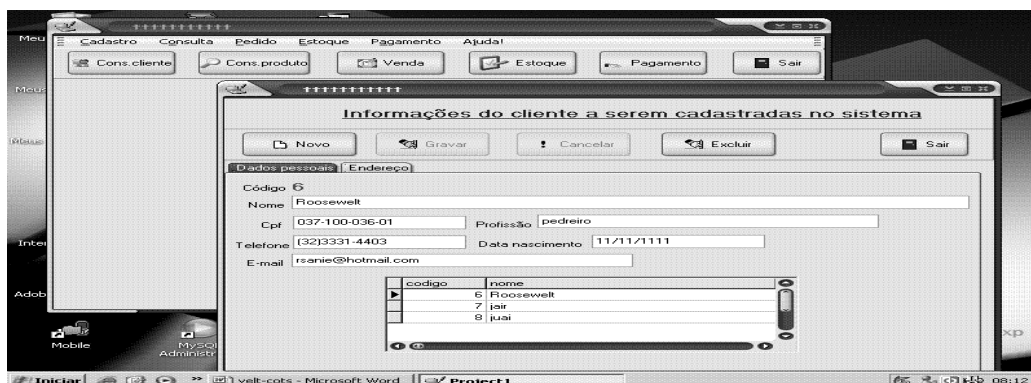


Figura-4 Aplicativo Demonstrativo

Depois de terminado a implementação do software, aplica se a sexta diretriz que consiste em fase de teste do software para possíveis alterações.

Assim concluído os passos das regras, obtém-se um software com maior qualidade, baixo custo, alterável e com maior confiabilidade devido a vários tratamentos diferenciados a cada processo de trabalho.

5- conclusão

O que foi proposto neste artigo, foi diretrizes para desenvolvimento de software baseado em cots, mostrando a importância de se ter regras para melhor tratar o desenvolvimento. Explicando passo a passo de modo a padronizar o desenvolvimento, pois é de grande importância para uma boa elaboração com mais qualidade e eficácia, para que possa ser utilizado em futuras implementações de sistemas mais complexos.

Utilizando estas diretrizes, temos em uma boa elaboração do projeto a ser desenvolvido, controle sobre o trabalho a ser realizado, e um método de implementação comum a todos, pelo simples fato de trabalhar cada parte do desenvolvimento de modo a obter maior aproveitamento e domínio da resolução.

Trabalhos futuros:

Como podemos observar com estes dados explicitados, podemos concluir que este tema não se esvazia em si, mas sim pode-se continuar os estudos sobre sistemas baseados em cots, na parte de implementação para que se possa melhorar o modo de desenvolvimento e documentação do software.

Isto ocorre pois, é um mercado que tende a crescer devido as exigências de hoje, nem sempre os desenvolvedores tem tempo suficiente para implementação de um sistema, tendo que recorrer para um sistema já parcialmente pronto, o que requer conhecimento e técnicas específicas para que possa ser modificado com melhor qualidade e agilidade.

6- Referências bibliográficas.

Filho, E. P. A. (2005) “Um estudo de casos de sistema de informação: desenvolvimento de um software de controle de matérias em uma empresa”, Unipac, Universidade Presidente Antônio Carlos.

Abreu, A.V. L. (2005) “Uma abordagem do desenvolvimento de um sistema de controle de vendas de materiais de construção com utilização de RUP e UML”, Unipac, Universidade Presidente Antônio Carlos.

Braga, R. T. V. , Germano, F. S.R., Masiero, P.C. (2002) “GRN Uma linguagem de padrões para gestão de recursos de negócios”, ICMC, Universidade de São Paulo.

Souza, F.M., Alencar, F.M.R., Castro, J.F.B. (1999) “O Impacto dos cots no processo de engenharia de requisitos”, UFPE Universidade Federal de Pernambuco.

Boaventura, A. P. (2004) “ Projeto de Reuso”

http://beast.dcce.ibilce.unesp.br/~ines/cursos/eng_soft/aula10, Dezembro.