

A Influência da Tecnologia da Informação em Ambiente Comercial e Desenvolvimento de Protótipo para Empresa de Representação Comercial

Vinícius de Moura da Silveira¹, Regiane Orlovski²

^{1,2} Tecnologia em Análise e Desenvolvimento de Sistemas – Faculdade Guairacá
CEP 85010-000 – Guarapuava – PR – Brasil

¹silveira.vine@gmail.com, ²regianeorlovski@hotmail.com

Abstract: *This article aims to present the participation and influence of information technologies on commerce and development phases of a prototype for a commercial representation company. The programming languages used to create the application were Java, NetBeans IDE and MySQL DBMS. The life cycle was approached to development was the prototyping in order to monitor the prototype development. At the end of the first round of the cycle, it was resulted a prototype that performs the tasks identified in the survey requirements.*

Resumo: *Este artigo tem por objetivo apresentar a participação e influência das Tecnologias de Informação no comércio e as etapas de desenvolvimento de um protótipo para uma empresa de representação comercial. A linguagem de programação utilizada para a criação do aplicativo foi Java, sobre o IDE NetBeans e o SGBD MySQL. O ciclo de vida abordado para o desenvolvimento foi o prototipação com o intuito de acompanhar de forma organizada o desenvolvimento do protótipo. Ao finalizar a primeira volta do ciclo, resultou um protótipo que realiza as atividades identificadas no levantamento de requisitos.*

Introdução

Devido a exigência do comércio na realização das tarefas de forma mais ágil, prática, eficaz e segura, as instituições comerciais buscaram na informática a solução para os seus problemas. Uma das soluções apresentadas foi o desenvolvimento e implantação de sistemas informatizados, onde as tarefas que antes eram realizadas manualmente e corriam um grande risco de falhas passam a ser realizadas sobre um sistema de computador, onde o mesmo tem como um dos objetivos manter a integridade de todas as suas informações e tarefas.

Cada sistema informatizado utilizado pelo comércio é desenvolvido visando às prioridades de cada empresa, fazendo com que alguns benefícios sejam alcançados. Um sistema desenvolvido, além de ser construído baseado nas tarefas que a empresa deve realizar, ainda tem como função organizar todas as informações relevantes e trazê-las para o usuário da forma como ele desejar.

Dessa forma, é relevante analisar que o desenvolvimento e implantação de um sistema, dentro de uma empresa comercial, têm o intuito de aumentar o seu rendimento, pois possibilitará realizar as atividades cotidianas de forma simples e ágil. Além disso, o fato de uma empresa possuir um software a inclui em outro nível comercial, uma vez

que ao buscar adquirir um aplicativo, demonstra que a instituição procura melhorias na qualidade e atualização em relação às novas tecnologias disponibilizadas no mercado.

Este trabalho tem por objetivo apresentar a aplicação das Tecnologias da Informação no ambiente comercial e os seus impactos. Outro assunto a ser abordado é o desenvolvimento de um protótipo para representação comercial, onde a principal função do aplicativo é de auxiliar na confecção de pedidos a partir de produtos, clientes, empresas de representação, transportadoras e formas de pagamento previamente cadastradas pelo usuário. Dentre as ferramentas utilizadas para o desenvolvimento do protótipo estão NetBeans IDE 6.9.1 – ambiente de desenvolvimento, MySQL Workbench 5.2 CE – utilizado para a modelagem lógica do banco de dados e Astah* Community 6.4.1 – utilizado para construir os diagramas da UML. A apresentação das etapas do ciclo de vida aplicado para o desenvolvimento e as metodologias utilizadas também estão presentes neste trabalho.

Desenvolvimento

Fundamentação Teórica

Devido ao intenso crescimento do comércio, conseqüentemente as formas de atendimento aos clientes tiveram que se tornar mais ágeis e práticas, buscando a satisfação dos consumidores. Uma das saídas encontradas para suprir essas necessidades foi recorrer ao setor da TI (Tecnologia da Informação), onde uma das soluções encontradas consistiu-se no desenvolvimento de sistemas informatizados. Segundo Turban, McLean e Wetherbe (2002), os sistemas de informação têm como uma das principais características a capacidade de modificar expressivamente a forma como conduzir um negócio.

Conforme Beraldi e Escrivão Filho (2000), a informatização das tarefas realizadas dentro de uma empresa, onde o sistema realiza suas funções em total eficiência oportunizará grandes vantagens, tanto em relação ao tempo otimizado, à facilidade na obtenção de informações quanto na organização entre outros aspectos que beneficiarão a empresa.

O desenvolvimento de sistemas para o comércio nasceu a partir da necessidade das empresas de utilizarem recursos práticos, precisos, ágeis e seguros na execução de suas tarefas, sendo uma forma de inserir as instituições no mundo informatizado e ainda propiciando a essas empresas uma vantagem competitiva em relação às demais.

Para Turban, McLean e Wetherbe (2002), sistemas informatizados ajudam a obter uma vantagem competitiva mediante a sua contribuição para a consolidação dos objetivos da empresa ou ainda a capacidade de aumentar o desempenho e a produtividade.

A informatização dentro de uma empresa comercial origina benefícios tanto para a instituição quanto para os seus clientes, pois um atendimento mais organizado e ágil resulta na satisfação dos consumidores. Segundo Beraldi e Escrivão Filho (2000), as empresas que possuem um sistema informatizado obtêm vantagens. Levando-se em consideração o desenvolvimento do sistema para o estabelecimento é possível a diminuição de pastas e fichas de arquivos, a simplificação e agilidade em tarefas burocráticas como relatórios de clientes, emissão de nota fiscal, consulta de atividades realizadas por determinado cliente ou funcionário, ou ainda processos referentes à

administração como gerência de marketing, planejamento e controle da produção, análises de custos e previsões orçamentárias por exemplo.

As empresas responsáveis por informatizar essas tarefas e organizar o fluxo das informações por meio de dispositivos de hardware, de aplicativos e de profissionais enquadram-se no ramo da TI. Conforme Campos Filho (1994), as TI são um conjunto de hardware e software que realiza uma ou mais tarefas de processamento das informações do sistema de informação, bem com coletar, transmitir, estocar, recuperar, manipular e exibir dados.

As aplicações das tecnologias de informação causam impacto tanto nos setores de uma empresa onde ela é aplicada quanto nos seus funcionários devido ao alto índice de reformulação das tecnologias e sistemas de computadores. Segundo Oliveira (1996), os impactos a respeito da produtividade e da forma da organização da empresa podem ser de grande importância, pois a TI difere das outras formas de tecnologia, afetando as operações de produção e coordenação.

Conforme Gonçalves (1994), após a aplicação da TI dentro de uma empresa é visível a mudança de patamar dos funcionários responsáveis pelas atividades gerenciais, em termos de habilidades individuais, sendo que para a introdução das tecnologias é necessária uma profunda análise da empresa e uma revisão que compreenda os seus processos e mecanismos de controle.

Normalmente, as empresas buscam os benefícios trazidos pela TI visando aumentar a qualidade do seu negócio e a atualização em relação às tecnologias existentes, tendo como objetivo principal o aumento de lucro. Para Gonçalves (1994), os objetivos desejados pelas empresas ao aceitarem as inovações tecnológicas variam bastante, porém na maioria dos casos trata-se de natureza aquisitiva, destacando-se ainda uma intensa preocupação com a competitividade da empresa, o seu desempenho comparado à concorrência e também pela pressão por parte dos clientes, mas muitas instituições possuem objetivos internos de grande importância, como a atualização, a redução dos custos e a gestão da empresa.

Tabela 1. Motivos para Implantação de TI

Motivos para implantação de TI	%
Necessidade de integração	2
Melhoria de controles organizacionais	24
Competitividade	8
Manter sua participação	5
Aumentar sua participação	22
Melhoria da qualidade de atendimento	8
Aumentar a produtividade	20
Gerar um ambiente criativo	1

Fonte: Prates e Ospina (2004)

Apesar das grandes vantagens resultantes da aplicação das tecnologias de informação, pode-se considerar de grande importância analisar, antes da implementação dessas técnicas, se a instituição se beneficiará com a aplicação delas. Caso existam benefícios, o responsável pela empresa de buscar essas tecnologias deve aprofundar-se no assunto visando aproveitar ao máximo a sua aplicação e os seus benefícios, sendo de grande ajuda basear-se em uma arquitetura da informação. Para Turban, McLean e Wetherbe (2002), uma arquitetura da informação é um mapa ou plano de alto nível das necessidades de informação de uma empresa, visto que ela assegura que a TI da instituição resolva as deficiências estratégicas dos casos da empresa.

De acordo com Beraldi e Escrivão Filho (2000), a primeira etapa é realizar um dimensionamento da instituição na atual situação e também uma dimensão planejada para o futuro, visto que desta forma as necessidades sejam extintas com os equipamentos adquiridos, sendo de grande importância também a indicação de pessoas qualificadas tanto para o dimensionamento quanto para a seleção dos produtos torna-se de grande valor, sendo que, em alguns casos, a escolha de empresas que desenvolvam sistemas personalizados é a melhor opção ao invés de adquirir um conjunto de aplicativos comerciais prontos.

Para implementação das funcionalidades das tecnologias de informação, torna-se necessário conhecer o todo que compreende a estrutura de manipulação dos dados. Para Turban, McLean e Wetherbe (2002), uma infraestrutura de informação baseia-se em todas as instalações físicas, serviços e administração que apoiam todos os recursos informatizados partilhados de uma empresa, destacando-se os cinco principais componentes: (1) hardware; (2) software de desenvolvimento; (3) redes e instalações de comunicação; (4) bancos de dados; (5) pessoal da administração da informação. As infraestruturas incluem esses recursos assim como sua integração, operação, documentação, manutenção e administração.

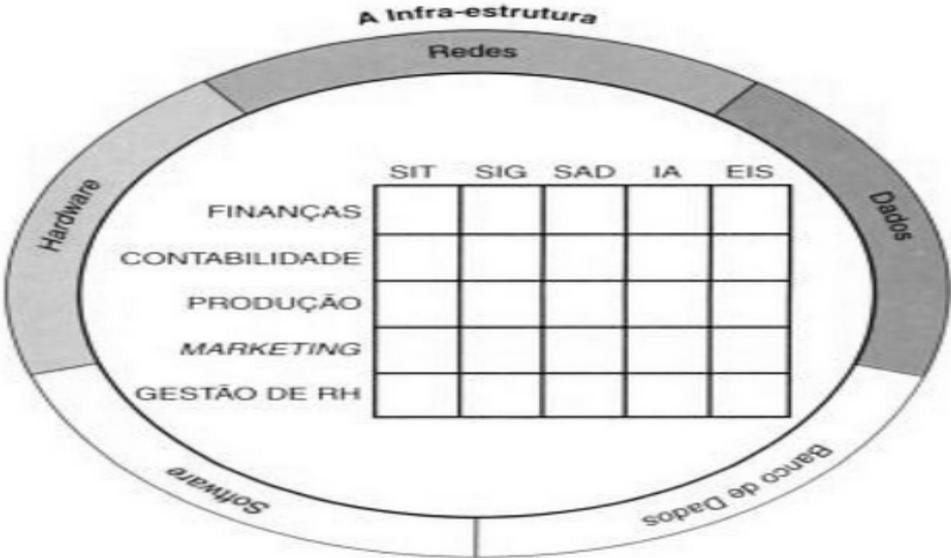


Figura 1. Visão esquemática da informação combinando áreas funcionais, os sistemas de apoio gerais e a infra-estrutura. A matriz interna inclui aplicações específicas em uma empresa. O anel externo inclui os recursos de computação de infra-estrutura.

Fonte: Turban, McLean & Wetherbe (2004)

O termo hardware designa todo e qualquer equipamento físico capaz de interagir com algum software. A utilização de hardware é essencial para a aplicação das tecnologias de informação, uma vez que a TI tem um forte pilar construído a partir da informática. Segundo Campos Filho (1994), o aumento da potência e da velocidade dos hardwares congregados com a diminuição do seu espaço físico é responsável pelo exponencial crescimento das tecnologias dos computadores e dos meios de comunicação.

Antes da escolha das ferramentas a serem utilizadas para o desenvolvimento do protótipo informatizado, torna-se necessário a definição do ciclo de vida a ser aplicado. O ciclo de vida abordado para a construção do aplicativo de representação comercial foi o de prototipação. Conforme Rezende (2005), a prototipagem é a metodologia de construir um software experimental de forma rápida e com baixo custo para demonstração, onde ele é avaliado e modificado conforme as solicitações do cliente. Complementam ainda Dennis e Wixom (2011), a metodologia de protótipo se baseia, nas fases de análise, projeto e implementação de forma coincidente, sendo que as três etapas são realizadas em circuito até a conclusão do sistema.

Por meio deste ciclo de vida é possível criar um modelo do software que será implementado e apresentá-lo ao usuário podendo tanto ser de forma concreta, utilizando papel e caneta, quanto de forma abstrata, por meio de um executável de computador apresentando a possível interface do software. Outra característica deste modelo é a possibilidade do usuário analisar o protótipo inicial verificando se o mesmo atende aos requisitos especificados. O modelo de ciclo de vida prototipação pode ser aplicado de duas formas: prototipação evolutiva, onde o protótipo evolui no decorrer do ciclo se tornando o sistema final; e a prototipação descartável: onde os protótipos iniciais são descartados e não utilizados no sistema final.

O software de desenvolvimento incluso no conceito das tecnologias de informação é o programa utilizado para construir um aplicativo. Para que seja possível desenvolver no aplicativo as operações desejadas, é necessário escolher primeiramente o tipo de programação para desenvolver o aplicativo, seguida da linguagem e por fim o software de desenvolvimento. Na programação existem duas categorias principais: a estruturada e a orientada a objetos. Conforme Mocarzel e Soma (2008), a programação estruturada com as suas regras e suas restrições auxilia o programador a impedir a construção de algoritmos confusos, ilegíveis e de difícil correção; nestes tipos de algoritmos chamados também de “teias de aranha”, a alteração em apenas um comando do programa pode ocasionar a necessidade de outras alterações em partes distintas do algoritmo; a programação orientada a objetos ajuda o programador a organizar os seus algoritmos, facilitando a localização de erros e substituição de trechos ineficazes, sem ser necessário examinar todo o código.

Visando auxiliar o desenvolvimento de softwares orientados a objetos, foram criados padrões de diagramação conhecido com UML (*Unified Modeling Language* – Linguagem de Modelagem Unificada). Segundo Dennis e Wixom (2011), o objetivo da

UML é oferecer uma terminologia comum dos conceitos baseados em objeto e métodos de diagramação que possa ser utilizado para modelar qualquer projeto de desenvolvimento de software, iniciando na fase de análise até a fase de implementação do próprio.

Os padrões criados pela UML totalizam nove técnicas, dentre as quais três utilizadas no desenvolvimento do protótipo estão relacionadas abaixo:

Os diagramas de Caso de Uso tem a função de especificar cada uma das atividades exercidas pelo sistema sendo que as atividades modeladas neste diagrama devem estar compreensíveis por qualquer pessoa que as visualize. Conforme Dennis e Wixom (2011), o caso de uso tem a função de comunicar em um nível alto quais operações o sistema precisa realizar, visto que as diagramações UML seguintes são construídas baseadas nele, exibindo a funcionalidade de maneiras distintas, sendo cada uma com um objetivo diferente.

O diagrama de Classe apresenta o modelo que mais se aproxima de um sistema orientado a objetos. Neste diagrama, como em uma linguagem orientada a objeto, são descritas as classes com seus atributos e métodos, onde uma classe pode possuir relações com outra classe. Segundo Dennis e Wixom (2011), o diagrama de classes é um esquema estático que auxilia na visualização do sistema em desenvolvimento, destacando o fato de ele apresentar as classes e os relacionamentos constantes no programa.

O diagrama de Sequência apresenta as iterações entre os objetos que compõem o sistema. A criação do diagrama auxilia na identificação dos métodos das classes. Para Dennis e Wixom (2011), o diagrama de sequência ilustra os objetos presentes em um caso de uso e as mensagens trocadas por eles por um determinado tempo para um único caso de uso.

Após definir a categoria de programação, é necessário escolher a linguagem de programação para descrição do código-fonte. Por meio da linguagem escolhida as funções e tarefas são construídas pelo programador. Para Mokarzel e Soma (2008), as linguagens tradicionais de programação possuem grande nível de aplicabilidade, sendo que seus propósitos são gerais e inúmeros, tanto para aplicações simples como para complexas. A categoria de programação definida para o desenvolvimento do protótipo foi a orientada a objetos e a linguagem de programação escolhida foi Java. Segundo Deitel (2010), o Java é uma linguagem de programação influente que auxilia na escrita de programas de forma ágil e eficiente, se ajustando ao contexto do desenvolvimento de sistemas que visam apoiar as instituições a construir seus próprios sistemas informatizados.

Dentre as vantagens desta linguagem destaca-se o fato dela ser baseada em orientação a objetos, permitindo de forma dinâmica e fácil a herança e consequente reutilização de códigos. Segundo Horstmann (2005) a herança é um mecanismo onde, se for necessário implementar uma nova classe e uma classe que retrata um conceito mais abrangente já foi criada, a nova pode adquirir as características da classe existente, proporcionando assim a reutilização de código.

O software de desenvolvimento onde o código é descrito e por vezes a interface já é construída, também é conhecido como IDE (*Integrated Development Environment* – Ambiente Integrado de Desenvolvimento). Para Deitel (2010), as organizações a fim de

desenvolverem sistemas de informações, podem buscar recursos em fornecedores de softwares importantes que disponibilizam ambientes de desenvolvimento integrado. Um IDE fornece ferramentas que suportam o método de desenvolvimento de software, compreendendo editores para criação e alteração de programas e ainda os depuradores para localizar os erros de sintaxe.

Complementa ainda Sebesta (2002), que um ambiente de programação é o conjunto de instrumentos utilizados no desenvolvimento de um software. Esse conjunto pode consistir em apenas um sistema de arquivos, um editor de texto, um *linkeditor* e um compilador, ou pode incluir ainda uma grande coleção de ferramentas integradas, onde cada uma delas é acessada por meio de uma interface constante.

O IDE escolhido para o desenvolvimento do protótipo de representação comercial foi o NetBeans IDE 6.9.1. A escolha partiu primeiramente de o software ser gratuito, de código aberto e por ser executável em diversas plataformas, sendo levado em consideração ainda o fato de a ferramenta possuir suporte para criar aplicativos tanto desktop quanto Web. A ferramenta disponibiliza bibliotecas que auxiliam na conexão do aplicativo com o banco de dados e bibliotecas que auxiliam na criação das interfaces da aplicação, facilitando assim o desenvolvimento.

Conforme Deitel (2010), os programas desenvolvidos sobre a linguagem de programação Java realizam a comunicação com o banco de dados por meio do Java *Database Connectivity*, um driver capaz de conectar um aplicativo Java a um banco de dados.

Um componente importante na estrutura das tecnologias de informação e na vida dos sistemas informatizados é o banco de dados. O banco de dados é responsável por armazenar e manter a integridade dos dados que, na maioria das vezes, são inseridos pelo usuário enquanto utiliza o software. Segundo Date (2004), um banco de dados é um conjunto de dados persistentes usados por sistemas de aplicação de uma empresa.

Além do banco de dados em si, faz-se necessário que um aplicativo tenha capacidade de restringir e permitir o acesso ao banco de dados para a manipulação dos dados como, por exemplo, inserir, excluir, modificar e consultar estes dados. O aplicativo em questão é nomeado DBMS (*Database Management System* ou SGBD).

Conforme Elmasri e Navathe (2011), um SGBD (Sistema Gerenciador de Banco de Dados) é um conjunto de programas que propicia aos usuários a criação e a sustentação de um banco de dados. O SGBD é um sistema de software que simplifica o método de definição, construção, manipulação e compartilhamento de banco de dados para diversos usuário e aplicações.

O SGBD escolhido para auxiliar nas funcionalidades relacionadas ao banco de dados por parte do protótipo foi o MySQL 5.1.41 por possuir uma licença de uso livre, no caso a GPL (*General Public License* – Licença Pública Geral), possuir um alto grau de desempenho e ser compatível tanto com a linguagem de programação elegida para o desenvolvimento quanto com o software de modelagem lógica. Outra característica que influenciou na escolha do MySQL foi o de existir um software compatível que possibilitou a visualização dos dados decorrentes das operações durante os testes no desenvolvimento do sistema. O software em questão é o phpMyAdmin 3.2.4; ele possui um ambiente gráfico de fácil operação que deve ser executado sobre um navegador

possibilitando assim a criação, exclusão, alteração e visualização das tabelas, dos dados e de suas características, além de ser possível executar comandos SQL.

Para que fosse possível ter acesso ao ambiente gráfico do phpMyAdmin foi necessário executar um servidor web (neste caso um local) e o escolhido foi o servidor Apache 2.2.14 pelo fato de ser *OpenSource* e compatível com os demais softwares utilizados no desenvolvimento. Para realizar as operações pelo phpMyAdmin, além do servidor web, foi necessário executar o MySQL. Em função disso, foi selecionado o software livre XAMPP 1.7.3; o mesmo possui um pacote de arquivos responsáveis por realizar a conexão com a base de dados MySQL e executar o servidor web Apache.

Etapas do Desenvolvimento do Projeto

Quando se decide criar um software é necessário analisar dentre os ciclos de vida qual o que melhor se encaixa no projeto de desenvolvimento da aplicação. Essa escolha é feita em conjunto pelo cliente e pelo analista de sistemas, pois dependendo das exigências e urgências do cliente o analista apresenta o ciclo de vida mais adequado.

O modelo de ciclo de vida escolhido para construção do aplicativo de representação comercial foi o de prototipação. O modelo de ciclo de vida prototipação pode assumir uma de três formas, onde a forma escolhida foi a de onde há um programa existente que executa uma parte da função desejada, mas que possui outras particularidades e métodos que serão melhorados no decorrer do ciclo.

Iniciando as etapas do ciclo de prototipação, a primeira a ser abordada é a de análise, na qual se enquadra a atividade de obtenção de requisitos. Nesta fase o desenvolvedor e o cliente definem os propósitos gerais do desenvolvimento do software, identificando quais os requisitos existentes e as operações que necessitam ter definições acrescentadas. Para realizar a coleta de requisitos para desenvolvimento do protótipo, o método primeiramente utilizado foi o de entrevista aberta com o proprietário da empresa onde foram relatadas as tarefas atualmente realizadas. Após a definição destas tarefas, foi solicitado ao responsável que realizasse as operações já definidas. Durante a execução dessas operações, foi aplicado o método de observação com o objetivo de especificar e filtrar quais as principais atividades realizadas e as que são executadas com maior frequência.

Finalizando a etapa de obtenção dos requisitos, deu-se início a fase do ciclo de vida chamada de projeto rápido. Durante este período foi realizada a representação dos aspectos do software que são visíveis ao usuário, por exemplo, as regras de negócio, a ordem preferível de apresentação dos dados a serem preenchidos pelo usuário, a delimitação de determinadas informações. Algumas dessas informações se tornaram ainda mais fáceis de serem identificadas devido ao fato de o responsável ter disponibilizado um modelo padrão da estrutura a ser implementada no protótipo, conforme segue Apêndice M. Outros aspectos ainda verificados com o responsável pela empresa foi o de apresentação dos dados já inseridos, as restrições de exclusão de determinados dados, a vinculação de clientes com determinados produtos e empresas representadas.

Um dos tópicos abordados nesta fase foi a de definição da arquitetura do protótipo, bem como a categoria de programação a ser utilizada, a linguagem de programação onde ele será desenvolvido, o IDE de escrita do código e o SGBD para controlar as funcionalidades que envolvam o banco de dados. Antes da definição das

ferramentas a serem utilizadas para o desenvolvimento do protótipo, foi necessário realizar a modelagem lógica do banco de dados, pois a modelagem tem influência total na construção do software, visto que a estrutura do banco é a base para a organização do armazenamento dos dados gerados pelo aplicativo. Uma modelagem bem estruturada e, por consequência, um banco de dados bem estruturado possibilita visualizar e estabelecer as informações de maior prioridade para o usuário, a restrição de determinados dados e a relação entre as diversas tabelas.

O software utilizado para modelagem do banco de dados foi o MySQL Workbench 5.2 CE (*Community Edition*). Esta ferramenta apresenta um ambiente gráfico que possibilita criar de forma prática a estrutura de um banco de dados, desde tabelas, chaves primárias, atributos e suas limitações até a relação entre as diversas tabelas incluindo a opção de chaves estrangeiras e cardinalidade. Após a criação de todas as tabelas, chaves, atributos e relacionamentos do modelo lógico no ambiente gráfico, é possível realizar a conexão com o banco de dados (neste caso deve ser o MySQL) e fazer com que todas as imagens construídas sejam transformadas em código que o SGBD compreenda, ou seja, na linguagem padrão de banco de dados, SQL (*Structured Query Language* – Linguagem de Consulta Estruturada) para criação da base de dados e a consequente inserção nesta base de todas as informações geradas. A modelagem do Banco de Dados utilizando a ferramenta MySQL Workbench encontra-se no Apêndice L.

Seguindo o modelo de prototipação, a fase seguinte abordada foi a de implementação, onde se enquadra a construção do protótipo. Nesta etapa as informações resultantes da obtenção de requisitos e do projeto rápido começam a ser implementadas. Para iniciar a construção do protótipo, primeiramente a categoria de programação escolhida foi a orientada a objetos. Este tipo de programação foi definido pelo fato de uma programação orientada a objetos possibilitar uma organização melhor das funções, métodos e atributos que envolvem as funcionalidades do aplicativo.

Por ser um protótipo, a manutenção e adição de futuras práticas nele se torna mais viável por meio da programação orientada, uma vez que as características e rotinas do objeto se encontram em um único objeto e encapsuladas.

A linguagem de programação orientada a objetos escolhida para o desenvolvimento foi Java. Um dos fatores que influenciou na escolha desta linguagem é pelo fato dela ser compatível com o banco de dados escolhido, MySQL.

Após definir o banco de dados, a categoria de programação e a linguagem, restou apenas selecionar a IDE onde o código é inserido e as interfaces desenvolvidas. Finalizada a seleção de todas as ferramentas, foi iniciada a codificação baseada nos dados resultantes da fase de obtenção de requisitos.

Na sequência da fase de construção do protótipo, a próxima fase a ser abordada é a de avaliação do protótipo. Durante este período o protótipo desenvolvido será utilizado pelo cliente para que seja observada a forma como o usuário realiza as operações, verificar se o aplicativo atende as necessidades relatadas pelo cliente e avaliação do usuário a respeito da interface. O resultado desta fase possui forte impacto no ciclo de vida, pois, dependendo da avaliação realizada pelo cliente, torne-se necessário reiniciar todas as etapas do ciclo.

Concluindo esta etapa, em conjunto com o cliente será realizado um refinamento dos requisitos para corrigir as falhas mencionadas pelo cliente, as funcionalidades que devem ser acrescentadas e as melhorias de interface. Procedendo aos resultados, pode ser necessário conduzir o projeto a sua primeira etapa (obtenção de requisitos) até que o protótipo esteja de acordo e as necessidades relatadas estejam liquidadas.

Enfim, a última etapa do ciclo a ser abordada é a de construção do produto. A nomenclatura desta fase a descreve por si só onde é realizada a confecção final do software. Com base nas etapas seguintes e aprovação do cliente na fase de avaliação, o aplicativo pode ser finalizado persistindo nos requisitos levantados na primeira fase.

Os padrões de diagramas estipulados pela UML facilitam a representação do sistema como um todo, partindo desde a modelagem dos casos de uso, passando pela diagramação das classes e chegando a criação dos diagramas de sequência.

A reprodução do sistema em imagens e formas geométricas facilita para a especificação de todos os elementos que influenciam e fazem parte do protótipo de representação comercial. Para realizar a modelagem dos diagramas de casos de uso, de classes e de sequência foi utilizado o software *astah* Community 6.4.1*.

Os diagramas de Caso de Uso foram utilizados para auxiliar na compreensão dos comportamentos externos do sistema por qualquer pessoa. Ele foi o primeiro a ser realizado pelo fato de mais se aproximar das atividades a serem realizadas pelo sistema e para sua confecção não ser necessária nenhuma formalidade. Por meio deste diagrama procura-se identificar quais serão os tipos de usuário que irão interagir com o sistema, os seus papéis e as suas funções. Os casos de uso representam as tarefas, serviços ou funções que podem ser aproveitados pelos usuários do sistema. Os diagramas referentes encontram-se nos Apêndices A, B e C. O diagrama de Classe apresenta as classes referentes ao protótipo, com seus atributos, métodos e relações. Por meio dele foi mais fácil identificar quais as classes que deveriam fazer parte da programação do protótipo, facilitando também na construção da modelagem do banco de dados. O diagrama de Classe referente ao protótipo se encontram no Apêndice D. O diagrama de Sequência apresenta a ordem da troca de mensagens entre os objetos que compõem o protótipo, detalhando como os objetos contribuem para um cenário de caso de uso. A criação do diagrama auxiliou na identificação dos métodos das classes que fizeram parte da programação do protótipo. Os diagramas de Sequência encontram-se nos Apêndices E, F, G, H, I, J e K.

Resultados

Com o desenvolvimento do protótipo foi possível identificar a importância de cada etapa do ciclo de vida escolhido para o projeto. A etapa de maior autoridade identificada foi a de obtenção de requisitos, pois é nela que a ideia de desenvolvimento de um aplicativo começa a ser posta em prática, sendo a base para a construção do software.

Antes da construção do protótipo, foi realizada uma pesquisa por ferramentas, aplicações de modelagem e desenvolvimento necessárias para o funcionamento do protótipo. Por meio da pesquisa foi possível identificar uma grande quantidade de softwares livre capazes de auxiliar no desenvolvimento do protótipo, dos quais alguns foram selecionados e utilizados. A escolha por softwares livres pode ser vista como um

grande benefício, uma vez que não será necessário adquirir licenças para que o protótipo seja desenvolvido.

No decorrer do desenvolvimento do protótipo, foi necessário realizar um estudo sobre as ferramentas selecionadas buscando aproveitar ao máximo a eficiência de cada uma. Isso proporcionou uma adição ao conhecimento que se deve ter para a criação de qualquer sistema informatizado.

A pesquisa por autores que descrevem a respeito da Tecnologia da Informação no comércio e dos conceitos de desenvolvimento de sistemas resultou em uma compreensão maior das necessidades e das características do mercado que deseja informatizar as suas atividades.

Considerações Finais

As Tecnologias de Informação já se encontram bem disseminadas no comércio nas mais diversas áreas. A aplicação das tecnologias em qualquer ambiente tem como objetivo facilitar e agilizar as tarefas realizadas, porém deve ser analisado o ambiente e as necessidades de cada caso para verificar se realmente é viável e necessário a implementação destas tecnologias.

Com este trabalho foi possível identificar particularmente quais os componentes que compõem a arquitetura das Tecnologias de Informação, destacando-se também a importância da modelagem do banco de dados para a criação de um software. A aplicação de um modelo de ciclo de vida é interessante pelo fato de delimitar as etapas a serem seguidas no desenvolvimento de um software, facilitando assim a identificação das tarefas a serem realizadas. A construção dos diagramas propostos pela UML facilitou na visualização e especificação no momento de começar a descrever o código do protótipo, ainda mais pelo fato de a aplicação ser baseada em linguagem orientada a objetos.

Finalizada a primeira série de desenvolvimento do protótipo e seguindo o ciclo de vida prototipação, na sequência será apresentado o software ao cliente e será realizada a avaliação junto ao mesmo para verificar se as necessidades relatadas na primeira etapa do ciclo foram suprimidas. Será avaliado também em conjunto com o cliente se a interface desenvolvida necessita de melhorias e novas funções a serem implementadas. Com base nas respostas apresentadas pelo cliente, será verificado se o protótipo atende as necessidades do cliente ou se necessita da adição de novas funções.

Referências

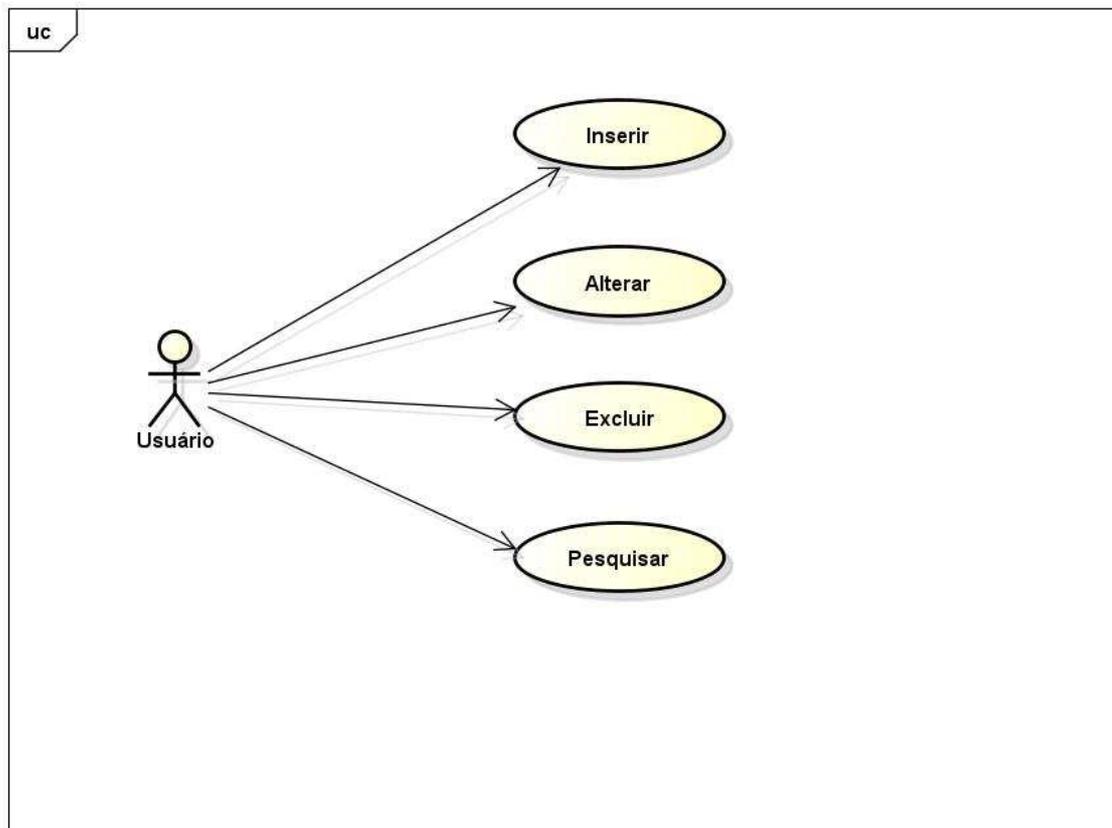
- Beraldi, L. C. & Escrivão Filho, E. (2000) “Impacto da Tecnologia de Informação na Gestão de Pequenas Empresas”. Ci. Inf., Brasília, v.29, n1, p. 46-50, Jan./Abr.
- Campos Filho, M. P. (1994) “Os Sistemas de Informação e as Modernas Tendências da Tecnologia de Negócios”. Revista de Administração de Empresas, São Paulo, v.34, n.6, p. 33-45, Nov/Dez.
- Date, C. J, “Introdução a Sistemas de Bancos de Dados” (2004) [*Introduction to database systems*]. Tradução: Daniel Vieira, 8 ed., Rio de Janeiro - RJ.
- Deitel, P. & Deitel, H. (2010) “Java: como programar”. Tradução Edson Furmankiewicz. 8. ed – São Paulo: Pearson Prentice Hal.

- Dennis, A. & Wixom, B. H. (2011) “Análise e Projeto de Sistemas”. Tradução: Michele Geinhart, 2 ed., Livros Técnicos e Científicos Editora Ltda., Rio de Janeiro – RJ.
- Elmars, R. & Navathe, S. B. (2011) “Sistemas de banco de dados.” [*Fundamentals of database systems*]. Tradução de Daniel Vieira. 6 ed. São Paulo: Pearson Addison-Wesley.
- Gonçalves, J. E. L. (1994) “Os impactos das novas tecnologias nas empresas prestadoras de serviços”. Revista de Administração de Empresas, São Paulo, v.1, n.34, p. 63-68, Jan/Fev.
- Horstmann, C. (2003) “Conceitos de Computação com o Essencial de Java” [*Computing Concepts with Java Essentials*]. Tradução: Werner Loeffler, Artmed Editora S.A., 3 ed., Porto Alegre – RS.
- Mokarzel, F. C. & Soma, N. Y. (2008) “Introdução à Ciência da Computação”. Elsevier Editora Ltda., Rio de Janeiro - RJ.
- Oliveira, A. C. M. da C. (1996) “Tecnologia de Informação: Competitividade e políticas públicas”. Revista de Administração de Empresas, São Paulo, v.36, n.2, p.34-43, Abr/Maio/Jun.
- Prates, G. A. & Ospina, M. T. (2004) “Tecnologia da Informação em Pequenas: Fatores de Êxito, Restrições e Benefícios”. RAC, v. 8, n 2, Abr/Jun 2004: p. 09-26.
- Rezende, D. A. (2005) “Engenharia de Software e Sistemas de Informação”, Brasport Livros e Multimídia Ltda., 3 ed., Rio de Janeiro – RJ.
- Ribeiro Neto, L. G.(1999) “Os impactos da tecnologia de informações nas organizações: uma visão política”. R. Un. Alfenas, Alfenas, 5:95-101.
- Sebesta, R. W. (2002) “Conceitos de Linguagem de programação” [*Concepts of programming language*]. Tradução: Addison Wesley Higher Education, Artmed Editora S.A., 5 ed., Porto Alegre – RS.
- Turban, E., McLean E. & Wetherbe J. (2002) “Tecnologia da Informação para Gestão: Transformando os negócios na economia digital” [*Information technology for management: transforming business in the digital economy*]. Tradução: John Wiley & Sons, Inc., Artmed Editora S.A., 3 ed. Porto Alegre – RS.

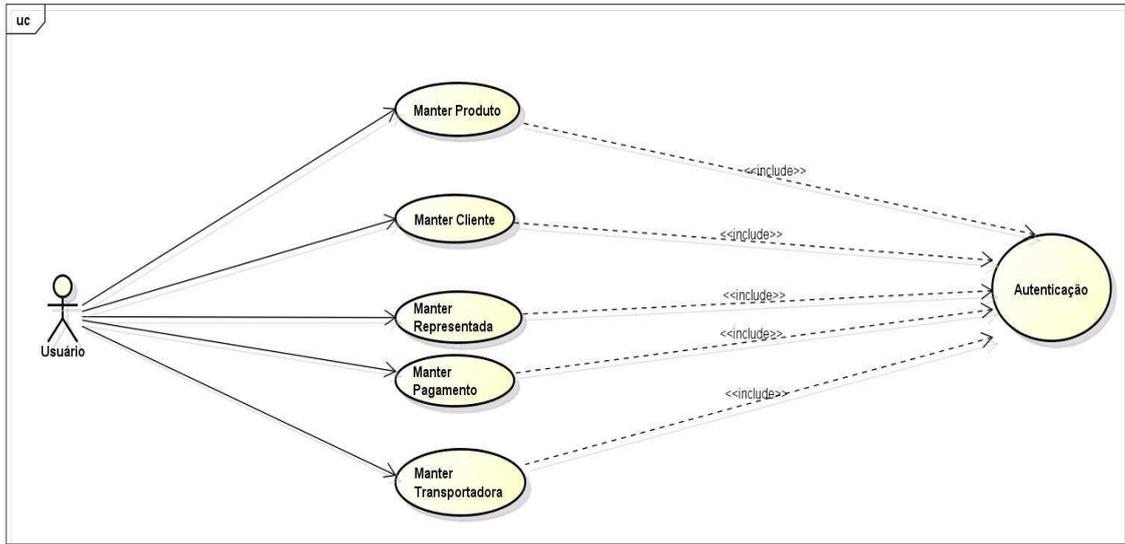
Apêndices

Diagramas UML referentes ao protótipo para empresa de representação comercial

- Diagrama de Casos de Uso
 - Apêndice A: Diagrama Caso de Uso Manter

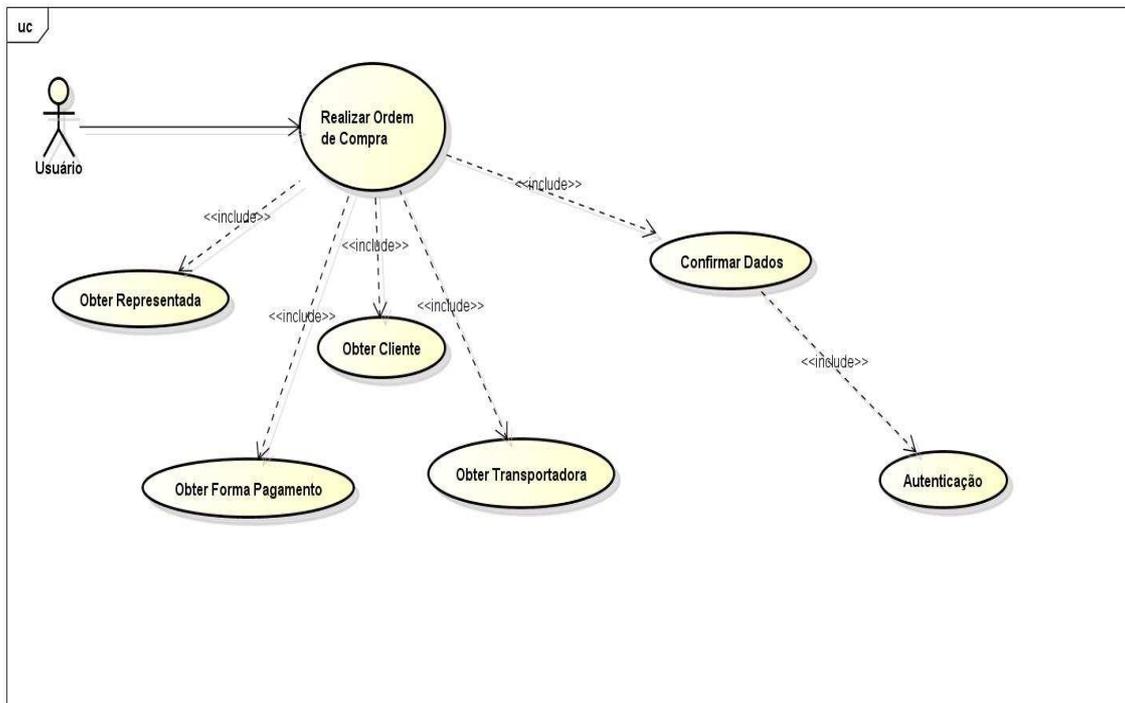


○ Apêndice B: Diagrama Caso de Uso Tarefa



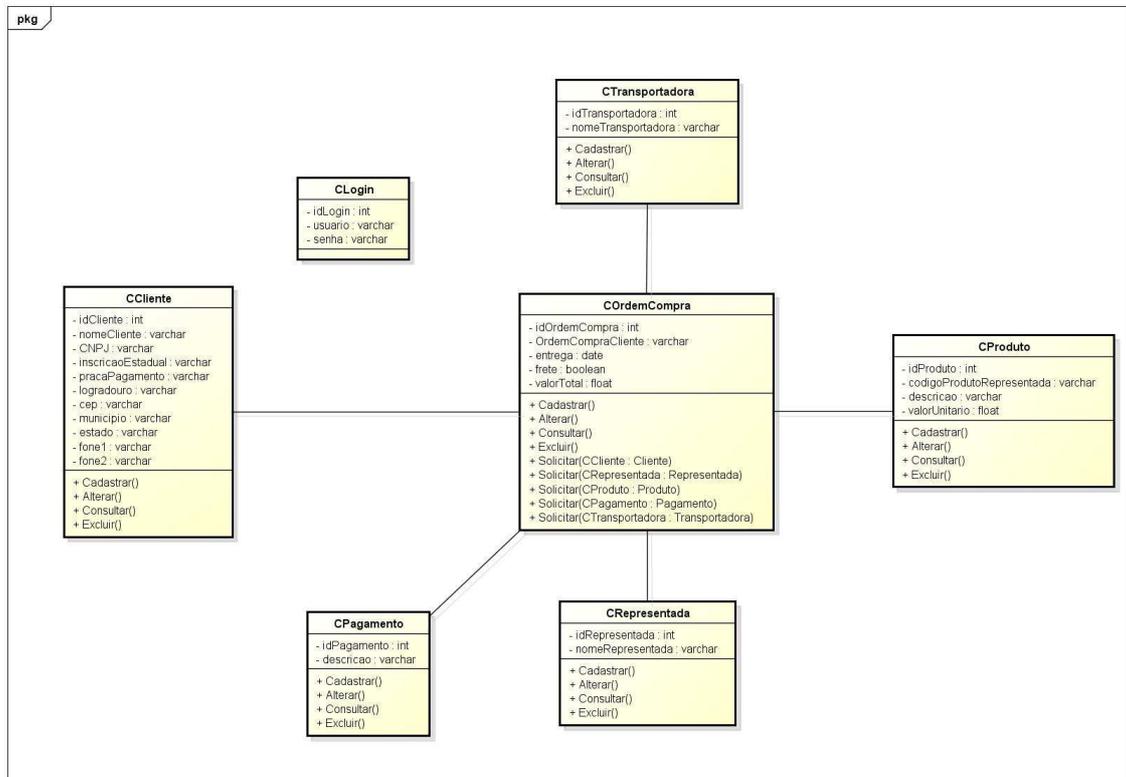
powered by astah

○ Apêndice C: Diagrama Caso de Uso Ordem de Compra



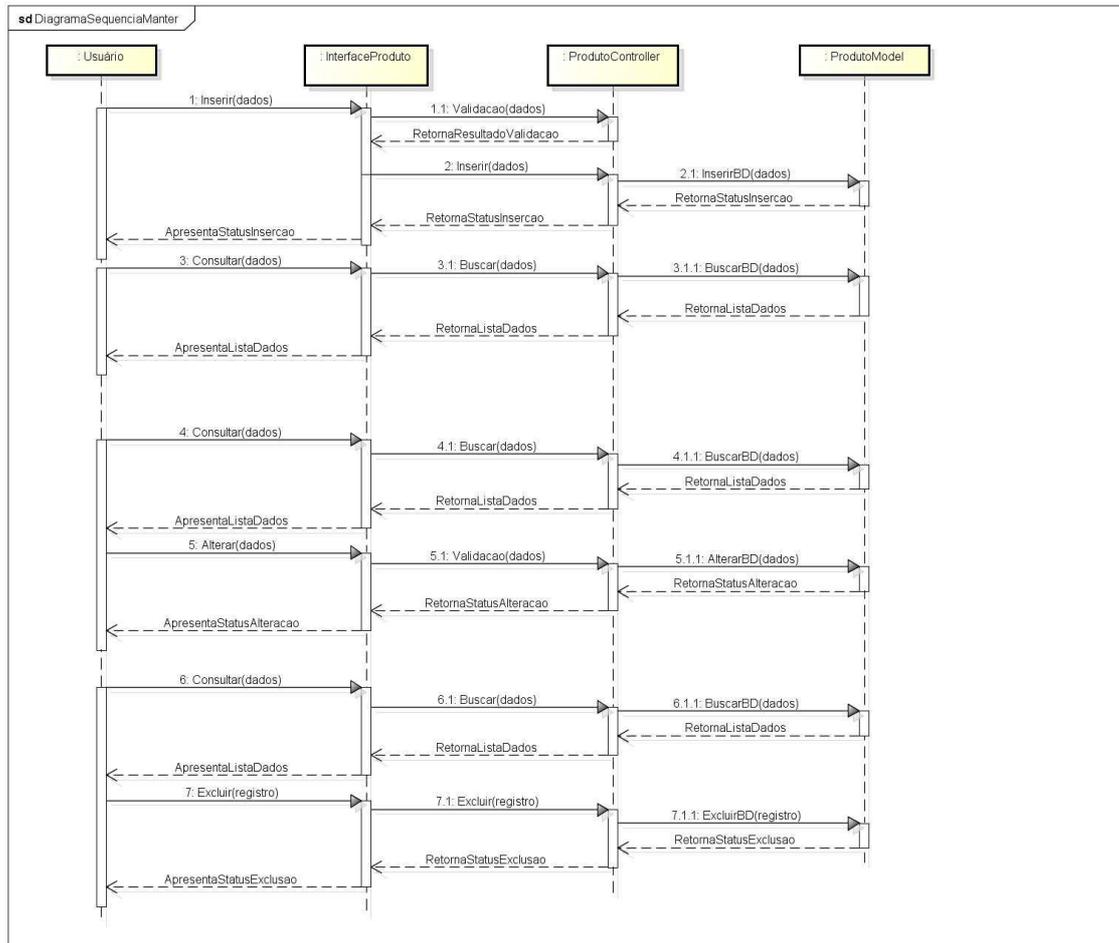
powered by astah

- Apêndice D: Diagrama de Classes

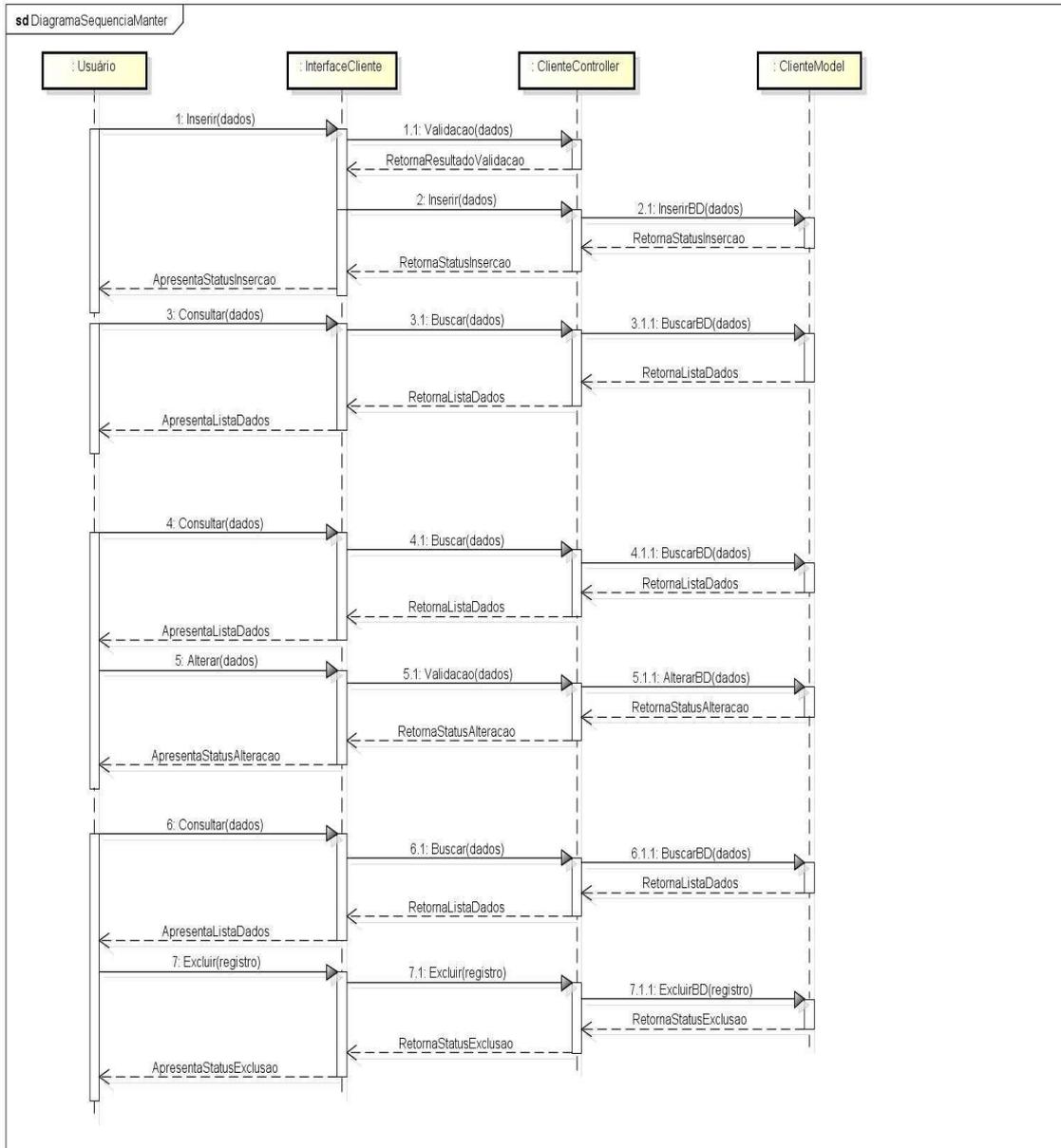


- Diagrama de Sequência

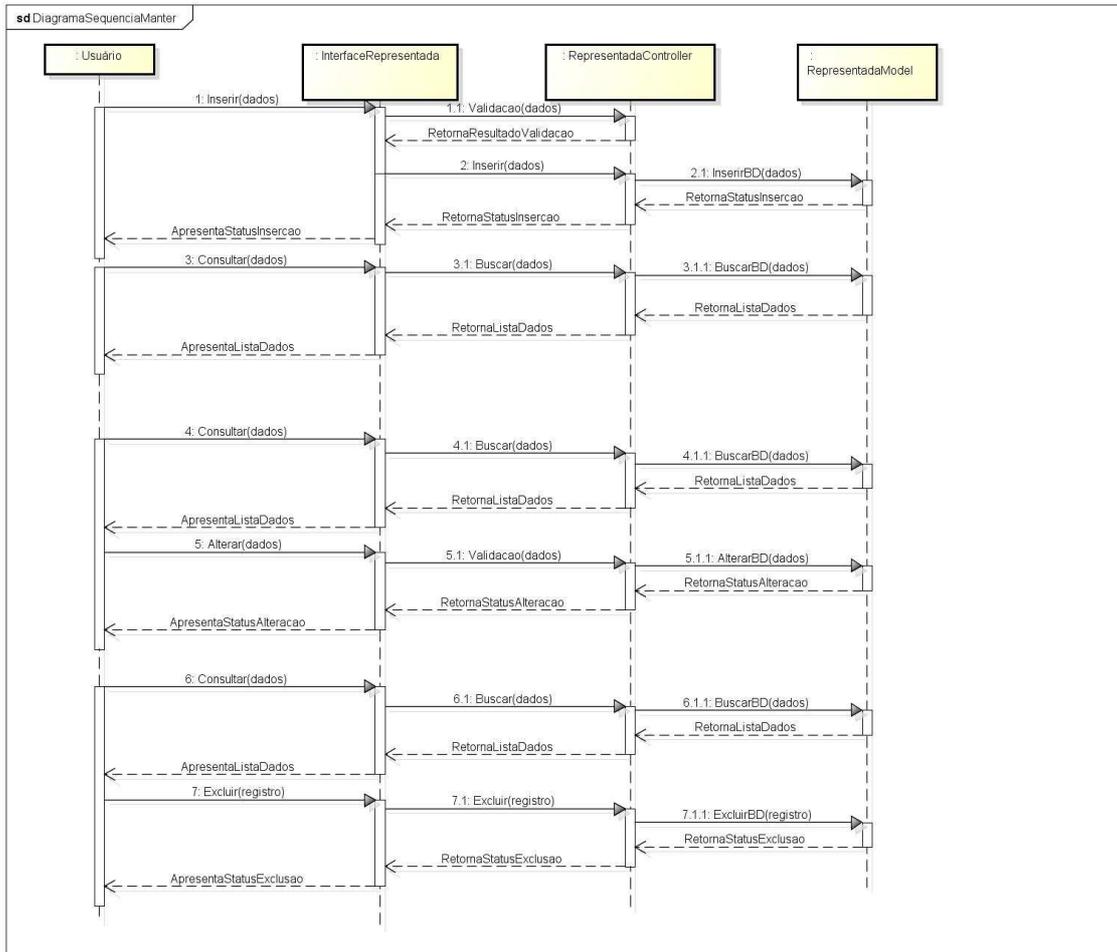
- Apêndice E: Diagrama de Sequência Manter Produto



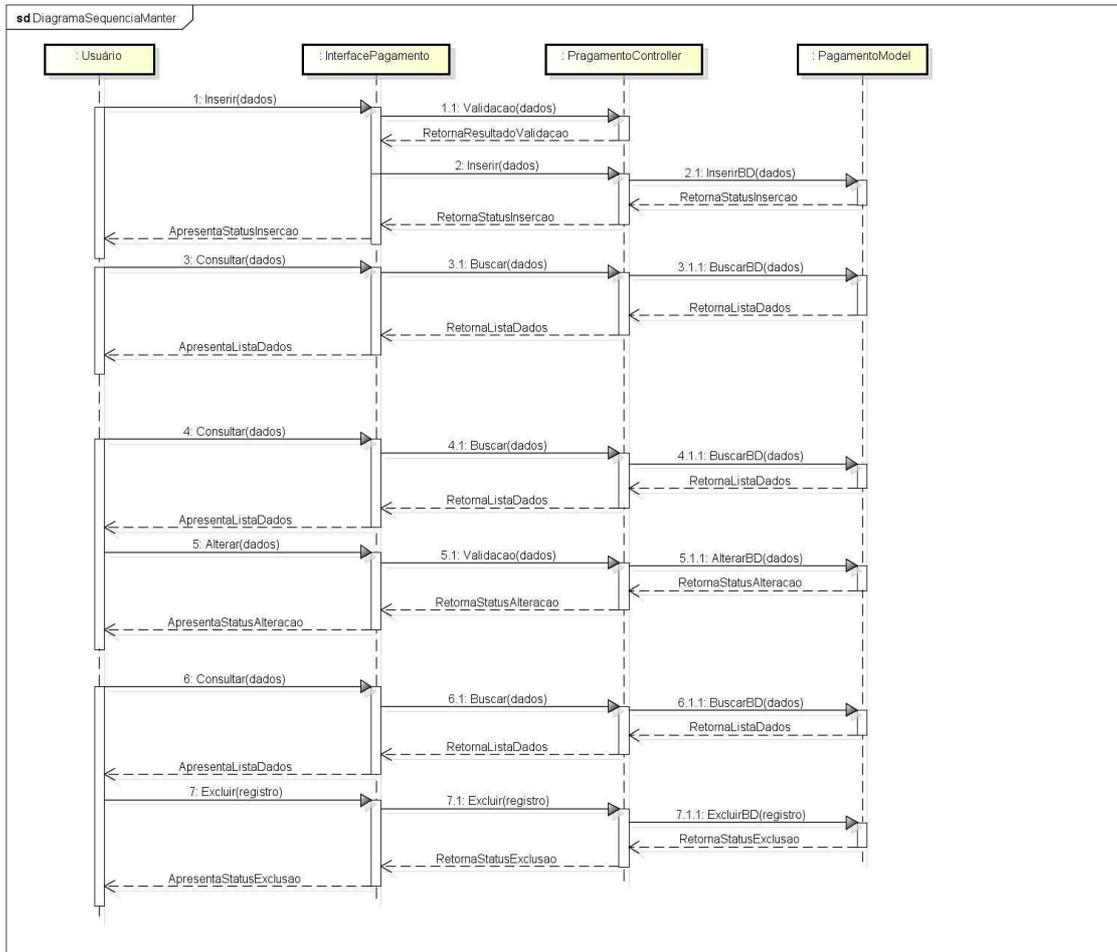
o Apêndice F: Diagrama de Sequência Manter Cliente



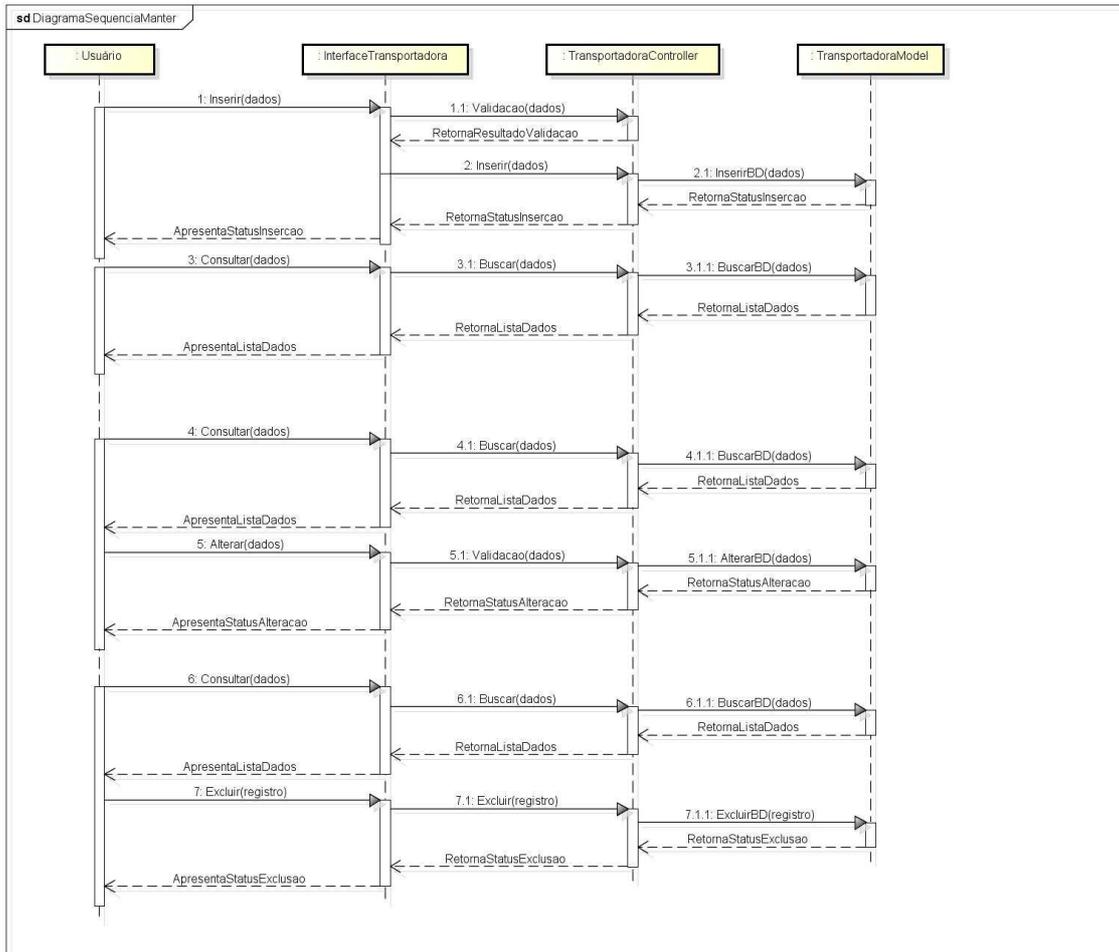
○ Apêndice G: Diagrama de Sequência Manter Representada



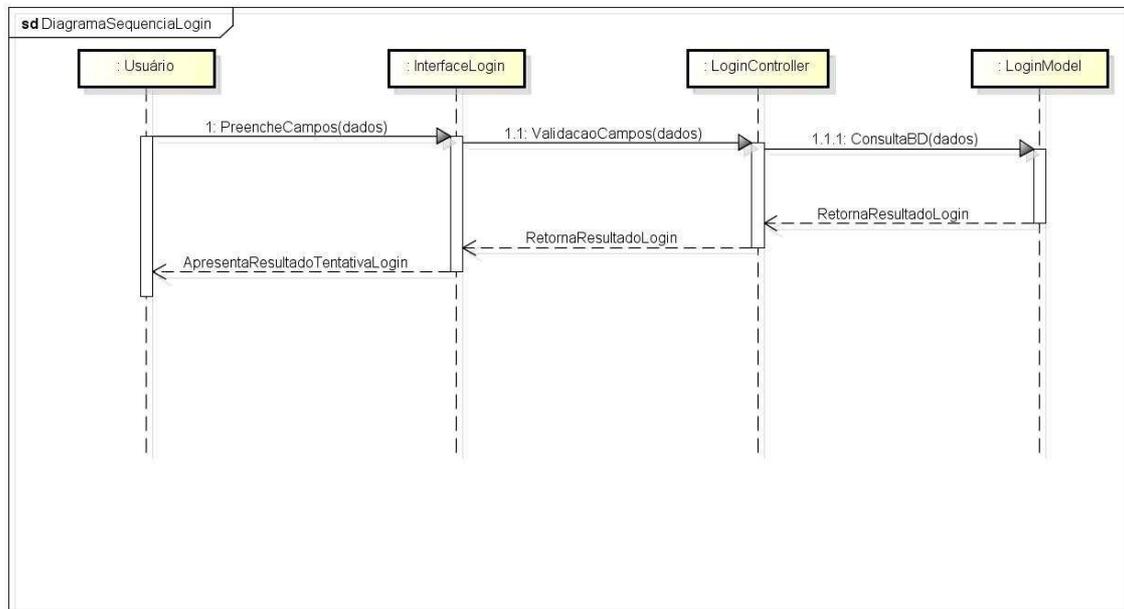
○ Apêndice H: Diagrama de Sequência Manter Pagamento



○ Apêndice I: Diagrama de Sequência Manter Transportadora

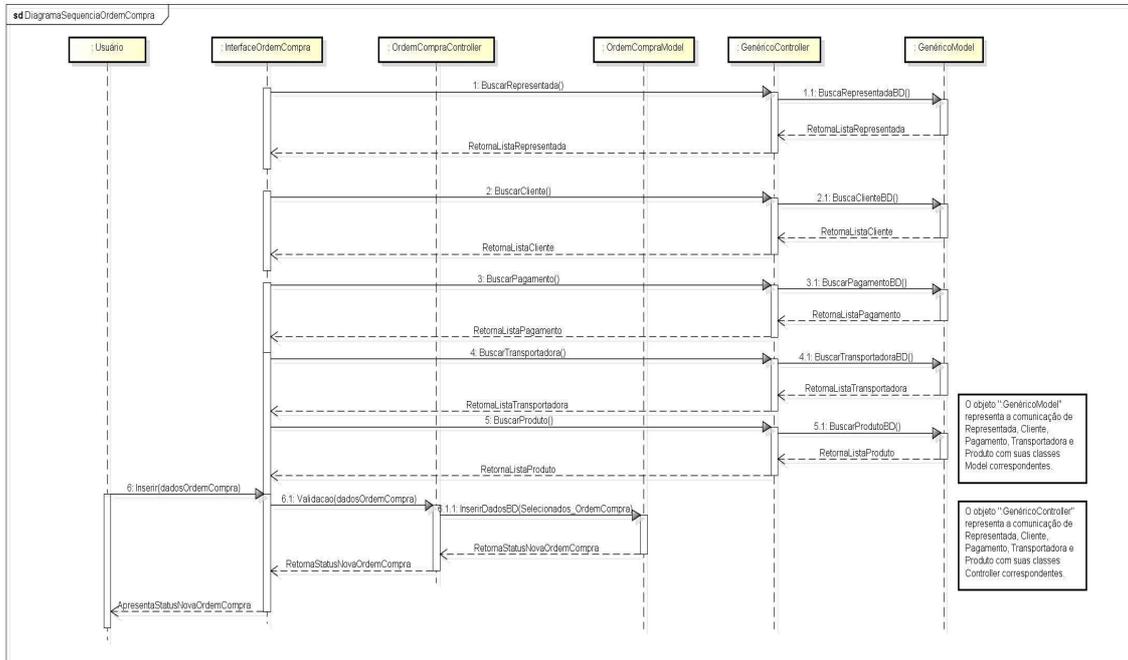


○ Apêndice J: Diagrama de Sequência Login



powered by astah

○ Apêndice K: Diagrama de Sequência Ordem de Compra



powered by astah

o Apêndice L: Modelagem Lógica do Banco de Dados

