

DESENVOLVIMENTO DO PET-SOFTWARE: sistema para controle de vacinas e agendas de Pet-Shop

Hewerton Luís P. Santiago ¹

Nicanor Davi Cardoso ²

RESUMO

Nas duas últimas décadas o Brasil vivenciou um grande crescimento do mercado relacionado animais de estimação. Neste sentido houve a expansão de empresas especializadas na comercialização de animais de estimação e também de produtos e serviços voltados para estes animais, conhecidas como Pet Shop. Devido a necessidade de se ter um controle mais confiável sobre os atendimentos realizados e a realizar em uma empresa de Pet Shop surgiu a oportunidade de desenvolvimento de um sistema Web com o objetivo de controlar e manter agendamentos e cadastros de informações. E como uma ferramenta apoio complementar foi desenvolvido um sistema mobile *Android* cujo o objetivo é possibilitar aos clientes o acesso a qualquer tempo e hora aos dados de atendimentos futuros e também daqueles já realizados dando mais comodidade ao cliente e a empresa.

Palavras-chaves: Agenda. Controle. Pet Shop.

ABSTRACT

In the last two decades, Brazil experienced a large growth of pet related market . In this sense there was the expansion of companies specializing in the sale of pets and also products and services for these animals , known as Pet Shop . Because of the need to have a more reliable control of the care provided and to carry on a Pet Shop company emerged from a

¹ Mestre em Administração pela Universidade Metodista de Piracicaba (UNIMEP). Especialista em Administração de Sistemas de Informação pela Universidade Federal de Lavras (UFLA). Coordenador e professor do Curso de Sistemas de Informação – Faculdade Atenas. ² Acadêmico do Curso de Sistemas de Informação – Faculdade Atenas.

Web system development opportunity in order to control and maintain schedules and information registrations. And as an additional support tool was developed a mobile Android system whose goal is to enable customers to access at any time and hour to future calls data and also those made already giving more convenience to the customer and the company .

Keywords: agenda . Control. Pet Shop.

1 INTRODUÇÃO

Gradativamente micro e pequenas empresas estão percebendo o que as grandes corporações já sabem a algum tempo, usar as tecnologias da informação não é apenas um facilitador para realização dos processos internos, mas sim um requisito fundamental para a sobrevivência das empresas num mercado cada vez mais concorrido e onde os cliente são cada vez mais exigentes e os concorrentes cada vez mais inovadores e criativos.

Corroborando a afirmação acima Molinari (2004) afirma que:

A Tecnologia da Informação é considerada um fator de desenvolvimento, principalmente quando é adotada no processo de gestão da organização. Trata-se ao mesmo tempo de insumo e recurso estratégico, podendo, portanto, a tecnologia apresentar-se não como elemento de oportunidade, dado que existe uma grande chance de avanços tecnológicos contínuos, mas também como elemento de risco, caso a organização não lhe dê a devida importância (MOLINARI, 2004, p.53). E quando se fala na "exigência dos clientes" nem sempre isso se refere apenas a

qualidade do produto o serviço final recebido, na era da tecnologia da informação onde tudo e todos estão conectados, busca-se cada vez mais facilidades e comodidade, e em relação a contratação de serviços isto não é diferente. Diante deste contexto, o problema de pesquisa pauta-se em: **de que maneira um sistema desenvolvido na plataforma *Android* e integrado à sistema *WEB* poderá auxiliar uma empresa de *Pet-Shop* na informatização dos cartões de vacinas, assim como na otimização do agendamento de serviços?**

Sendo assim, buscou-se modelar e desenvolver um Sistema *Web* integrado com um aplicativo *Android* para auxiliar no gerenciamento e controle dos serviços e agendas de um *PetShop*. Como objetivos específicos deste artigo, preocupou-se em proporcionar funcionalidades de modo a disponibilizar informações aos clientes e possibilitar a interação do cliente com a empresa por meio de um sistema *mobile*; informatizar cadastros de clientes e animais e estabelecer um vínculo entre ambos; informatizar a agenda de atendimentos; controlar aplicação de vacinas através de uma rotina do sistema web; usar Plataforma *Android* afim de dinamizar a consulta aos agendamentos. O presente estudo parte da hipótese de que, com a utilização de um sistema *Web* para agendamentos e controle de vacinas

integrado a um Aplicativo *Android* para consultas e notificações, será possível melhorar e estreitar o relacionamento com os clientes, uma vez que o cliente terá acesso a agendamento de atendimentos, e histórico de atendimentos anteriores, dentre outros. Como projeto futuro o sistema proporcionará ao cliente a possibilidade de avaliar atendimento recebido, assim acredita-se que a empresa terá um *feedback* para nortear suas decisões no que diz respeito a

ações a serem tomadas para melhoria do atendimento e relacionamento com o cliente.

Uma das relevâncias deste trabalho, consiste no fato de possibilitar o estreitamento de laços entre a empresa e seus clientes tendo como resultado final a fidelização deste clientes. Neste sentido, Cobra (2009) afirma que as empresas para sobreviverem à guerra sem trégua de mercado devem buscar a parceria de seus clientes, procurando fazer do relacionamento com cliente a principal arma para luta de mercado, segundo este o autor a empresa deve criar maneiras para que o cliente se sinta envolvido e até mesmo cúmplice da empresa.

Para o desenvolvimento deste trabalho, inicialmente definiu-se a área de pesquisa e o problema a ser trabalhado, posteriormente, realizou-se pesquisas bibliográficas e também levantamento de dados por meio de entrevistas e questionário, afim de buscar uma maior integração e compreensão do ambiente empresarial. O próximo passo metodológico deu-se na escolha das ferramentas computacionais onde, optou-se pela linguagem de programação Java, com Ambiente de Desenvolvimento Integrado (IDE's) Netbeans para o sistema Web e Eclipse para o sistema Android, utilizou-se as frameworks JSF(Java Server Faces) e Prime Faces para o design de interface com o usuário e Hibernate para persistência de dados, na parte de banco de dados optou-se pela utilização dos SGDB'S MySql e SQLite, sendo o primeiro para o sistema Web e o segundo para o sistema Android, como servidor de aplicação foi utilizado o Apache Tomcat, em seguida, realizou-se o desenvolvimento do software proposto, seguindo o padrão de arquitetura MVC(Model, View, Controller).

2 SISTEMAS DE INFORMAÇÃO E SUAS ABRANGÊNCIAS

O ambiente empresarial no ano de 2015 apresenta um cenário cada vez mais hostil, onde empresas – principalmente as pequenas e médias – estão constantemente sucumbindo ante seus concorrentes, neste contexto sobreviver no mercado não é uma tarefa trivial e requer esforço dedicação, e criatividade, assim cada dia mais as empresas busca desenvolver estratégias não apenas para se manter no mercado mas para se destacarem e buscarem sua afirmação. Dentro do contexto de competitividade e da inovação, a informação é a base para qualquer ação dentro das organizações, pois é ela que dá a base para a tomada de decisão e para o planejamento ações para o presente e futuro. Segundo Oliveira (2007, p.171) “o propósito básico da informação é habilitar a empresa para alcançar seus objetivos pelo uso eficiente dos recursos disponíveis”, neste sentido os sistemas de informação baseados em computador são tão importante para empresas como as pessoas ou processos.

Para Turban (2005) um sistema de informação baseado em computador utiliza a tecnologia de computador para coletar, processar, armazenar analisar e disseminar informações para uma finalidade específica. Nas últimas décadas o Brasil vivenciou um grande crescimento do mercado relacionado ao comércio de animais de estimação e os produtos e serviços voltados para estes animais proporcionando o crescimento das empresas que atual neste seguimento, as chamadas Pet Shops (ELIZEIR, 2013). Em abril de 2015 o Brasil tornou-se o segundo maior mercado Pet do mundo movimentando R\$ 15 bilhões por ano (PORTAL G1, 2015). Diante deste contexto, o mercado de desenvolvimento de software para esta área tornou-se promissor. A área de desenvolvimento do software oferece uma gama enorme de ferramentas, linguagens, técnicas de modelagem e gerenciamento dos processos de desenvolvimento. A escolha das ferramentas adequadas é fundamental para um se chegar ao produto final de forma eficaz, nos próximos tópicos é apresentada as tecnologias computacionais empregadas no estudo e as características básicas destas tecnologias.

2.1 FERRAMENTAS COMPUTACIONAIS

Java é uma linguagem de programação orientada a objetos, extremamente popular, dentre diversos motivos graças a portabilidade que a Máquina Virtual *Java* (JVM) proporciona, uma vez que as aplicações *Java* não são compiladas para um Sistema Operacional específico e sim para serem executados na JVM, desta forma uma aplicação poderá ser executado em um sistema operacional diferente daquele em que foi desenvolvida (DEITEL, 2008). Neste sentido Deitel (2008, pg. 10) afirma que “Se a mesma VM for implementada nas várias plataformas de computador, os aplicativos que ela executa podem ser utilizados em todas estas plataformas”. O indicador da popularidade de linguagens de programação *TIOBE Programming Community*, publicou em maio de 2015 uma pesquisa baseada no número de engenheiros qualificados em todo o mundo, cursos e fornecedores terceirizados e sites de busca populares, como *Google*, *Bing*, entre outros, que afirma a popularidade da linguagem *Java*. Na pesquisa o *Java* aparece em primeiro lugar no *ranking* das linguagens de programação mais populares do mundo com, onde *Java* tinha 16,869% da preferência dos programadores e logo atrás vem a linguagem *C* com 16,847% da preferência (*TIOBE PROGRAMMING COMMUNITY, 2015*).

Dentre os vários benefícios que motivaram a escolha do *Java* com a linguagem e

programação para este projeto, alguns também são apresentados por Deitel (2008) em que o autor destaca que a linguagem fornece aos desenvolvedores níveis mais altos de abstração e uma melhor organização do código obtida através da separação das camadas da aplicação, facilitando a manutenção e extensão do código. Conforme Deitel (2010, p.952) “uma *framework* fornece bibliotecas e às vezes ferramentas de *software* que ajudam a organizar e construir aplicativos”. Para o desenvolvimento de interface com o usuário será utilizado as *frameworks* JSF e *Prime Faces*. “O JSF (*Java Server Faces*) fornece um conjunto de componentes de interface com o usuário que simplificam o design de páginas *Web*”. (DEITEL, 2010, p.952). Já o *PrimeFaces* é apresentado por Luckow e Melo (2010) com sendo uma das mais completas bibliotecas de componentes para JSF, com dezenas de componentes de interface.

O projeto adota o padrão arquitetural de desenvolvimento MVC Modelo-VisãoControle (MVC), que separa o sistema em três camadas, Modelo, Visão e Controle, tornado mais fácil a manutenção do código e seu principal objetivo é definir como as três camadas devem interagir (LOCKOW; MELO, 2010). Cordeiro (2012) apresenta as características do Modelo, Visão e Controle, segundo o autor a Visão (*View*) representa a parte do sistema que interage com o usuário, ou seja, aquilo que está diretamente relacionado a interface gráfica do sistema, o Modelo (*Model*) traz as regras de negócio da aplicação, realiza os cálculos, processamento e etc. O Controle (*Controller*) funciona como um mediador entre a Visão e o Modelo, ele recebe as requisições do usuários (feitas na *View*) motas os objetos correspondentes e os encaminha para o Modelo realizar as operações solicitadas, depois o *Controller* receber o retorno da operação executada e encaminha o resultado para a *View*.

Para persistência de dados foi utilizado o *Hibernate* que é caracterizado por Luckow e Melo (2010) como uma *framework* de Mapeamento Objeto Relacional (ORM), que simplifica e agiliza a persistência dos dados. Como este artigo trata do desenvolvimento de um sistema Web, obrigatoriamente este sistema estará operando online e os dados serão persistidos em banco de dados também online, por tanto faz-se indispensável implementar algumas rotinas voltadas a segurança da informação, para reduzir possíveis vulnerabilidades e exposição aos riscos. A informação é ponto vital para qualquer organização, sendo assim é muito importante que os softwares estejam cada vez mais robustos no que concerne a sua segurança. Com o intuito de reduzir vulnerabilidades e riscos do sistema Web foram utilizadas algumas tecnologias que serão citadas no decorrer deste tópico para aumentar a segurança do software. Para controlar os acessos as partes restritas do sistema de uma forma geral foi utilizado o

framework SpringSecurity, segundo Luckow e Melo (2010) esse *framework* garantirá a autenticação e a autorização dos usuários, permitindo assim que somente quem realmente é cadastrado no sistema tenha acesso aos recursos restritos, com acesso somente ao conteúdo para o qual recebe permissão. Foi implementado também uma rotina para que antes que as senhas dos usuários sejam persistidas no banco de dados elas são submetidas a uma criptografia MD5, assim caso os dados de acesso ao sistema de algum usuário sejam interceptados isto irá dificultar que uma pessoa não autorizada consiga logar no sistema. Um banco de dados de uma forma genérica pode ser entendido como um conjunto organizado de dados armazenados em um determinado local que servirão para um determinado fim.

Dentro do contexto da tecnologia da informação Beighley (2011) define um banco de dados como sendo um contêiner que guarda tabelas de dados e outras estruturas SQL relacionadas a estas tabelas. Sistema de Gerenciamento de Banco de Dados (SGDB) aplica-se apenas ao contexto computacional, para Alves (2009, p.23) um SGDB “é um conjunto de sistema de software para computadores que possui um finalidade genérica, capaz de tornar mais fácil o processo de definição, estruturação, criação, manipulação e manutenção de banco de dados. Com o intuito de se desenvolver um *software*, robusto, seguro e com o mínimo de redundância de dados, será utilizado neste projeto os SGDB’s *MySQL* e o *SQLite*, pois além das vantagens já citadas eles apresentam alto desempenho dentro do contexto a que se aplicam, e a um custo zero. Segundo Alves (2009), o *MySQL* é um sistema de banco de dados Relacional, onde os dados são organizados em tabelas (relações) formadas por linhas e colunas, as tabelas relacionam as informações referentes a um mesmo assunto de modo organizado. Para integrar as aplicações Web e Android e consequentemente realizar o sincronismo de dados entre banco *MySQL* do sistema Web o *SQLite* do Android foi utilizado a tecnologia de Web Service que será explicada no tópico seguinte. Com o intuito de integrar o sistema Web com o Sistema android foi implementado um WEB Service que será consumido pelo sistema Android e cujo o objetivo é acessar o bando de dados *MySQL* online e disponibiliza-las ao sistema Android

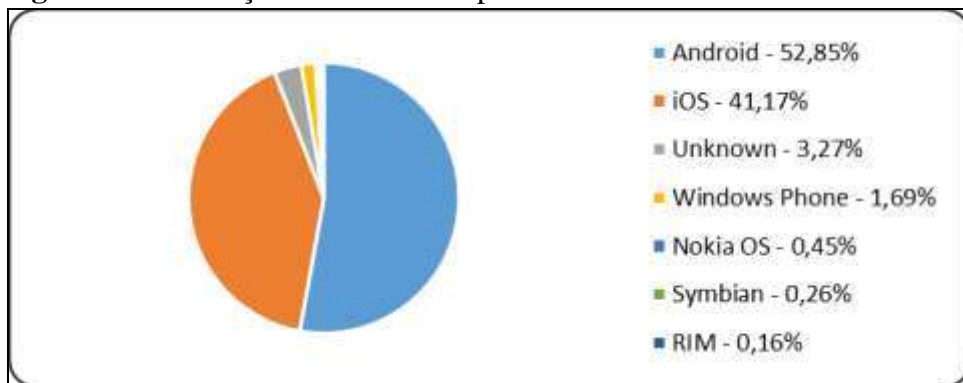
Para implementação do Web Service foi utilizado a tecnologia REST (*Representation State Transfer*) que segundo DEITEL (2008, é um estilo arquitetônico para de implementar Serviços Web, uma arquitetura de rede que utiliza os mecanismos de solicitação/resposta tradicionais da web como solicitações GET e POST, onde cada um de seus métodos é identificado por um URL único.–Para trafegar as informações por um Web Service Rest pode ser utilizado os formatos XML, JSON, HTML, dentre outros. Para o

projeto em questão foi escolhido o formato *JavaScript Object Notation* (JSON). A escolha do JSON se justifica também no fato de o Android possuir classes prontas que facilita os tratamentos para consumo de JSON em aplicações Android, dentre estas classes foi utilizado neste projeto a classe `JSONObject` que facilita o processo de recuperação e manipulação de dados do JSON.

2.1.1 A PLATAFORMA ANDROID

O *Android* é um sistema operacional de código fonte aberto voltado para dispositivos móveis. A organização DeviceAtlas, mostrou em uma pesquisa publicada em março de 2014 o *ranking* do uso de Sistemas Operacionais *Mobile* no mundo e no Brasil conforme ilustra a Figura 01. O ranking é medido através do tráfego de web ocupado pelos sistemas operacionais onde o sistema operacional Android ocupava no Brasil o primeiro lugar do ranking com 52,85%, já uma pesquisa atual (ano 2015) disponibilizada por esta mesma organização em seu web site é mostra um domínio ainda maior da plataforma android com 73,78% (DOTMOBI, 2015). Esse uso massivo da plataforma Android motivou a escolha deste sistema operacional com uma das plataformas para o sistema a ser desenvolvido, pois conforme se percebe pelas pesquisas citadas o Android não é apenas uma plataforma que já se afirmou no mercado, mas é também um plataforma em expansão que apesar de já dominar o mercado continua crescendo.

Figura 01- Utilização de Sistemas Operacionais Mobile no Brasil em 2015.



Fonte: Adaptado de DOTMOBI (2015).

Para desenvolvimento do sistema Pet Software a ser utilizado na plataforma Android, assim como o sistema Web foram seguidos regras e técnicas de engenharia de software, com o intuito de levar o desenvolvimento a um patamar mais profissional pautado

nas melhores práticas de desenvolvimento, para que se pudesse chegar a um produto final de qualidade sem extrapolar prazos e evitar erros de análise projeto e desenvolvimento.

2.1.2 ENGENHARIA DE SOFTWARE

Mesmo com todos os problemas Somerville (2011) afirma que a engenharia de software tem evoluído nos últimos 50 anos aprimorando apresentando um conjunto de métodos e ferramentas que tratam questões práticas de custo, prazo e confiança, assim como a necessidade dos clientes e desenvolvedores. Pressman (1995) acrescenta que a engenharia de software abrange um conjunto de três elementos fundamentais, sendo, métodos, ferramentas e procedimentos, que possibilita ao gerente controlar o processo de desenvolvimento do software, e oferece ao profissional uma base para construção de software de alta qualidade produtivamente. Neste sentido, Somerville (2011) complementa afirmando que a engenharia de software tem por objetivo apoiar o desenvolvimento de software incluindo técnicas que apoiam especificação, projeto e evolução de programas, ela analisa também questões práticas de custo, prazo e confiança, assim como as necessidades dos clientes e produtores de software. Segundo Ahrens *et al.* (2007) um protótipo de um produto ou componente é parte essencial no seu processo de desenvolvimento, possibilitando a análise de sua forma e funcionalidade antes a produção de ferramenta definitiva. Pressman (1995), apresenta as seguintes características da prototipação:

A prototipação é um processo que capacita o desenvolvedor a criar um modelo de software que será implementado. O modelo pode assumir uma das três formas: (1) um protótipo em papel ou modelo baseado em PC que retrata a interação homem-máquina de uma forma que capacita o usuário a entender quanta interação ocorrerá; (2) um protótipo de trabalho que implementa algum subconjunto da função exigida do software desejado; ou (3) um programa existente que executa parte ou toda a função desejada, mas que tem outras características que serão melhoradas em um novo esforço de desenvolvimento (PRESSMAN, 1995, pg.35)

Dentro dos conceitos e características da prototipação Guedes (2011) acrescenta que a prototipação consiste em desenvolver um rascunho do que seria o sistema de informação quando ele estivesse finalizado, o protótipo irá apresentar pouco mais do que a interface do software a ser desenvolvido, ilustrando como as informações seriam inseridas e recuperadas no sistema

2.1.3 ANÁLISE E PROJETO DE SOFTWARE

Segundo Guedes (2011) a fase de análise trabalha com o domínio do problema,

onde é identificado as funcionalidades necessárias ao software e suas restrições. A fase da análise identifica quais as funcionalidades necessários ao software, e para identificar as necessidades do cliente e estabelecer as funcionalidades necessárias ao sistema foi realizado o levantamento de requisitos a partir de onde se pode conhecer melhor o cenário onde o software será empregado e o que se deseja alcançar com a sua utilização. O levantamento de requisitos é uma etapa de extrema importância no desenvolvimento de software, pois segundo Guedes (2011) é nessa etapa que é se busca compreender as necessidades do usuário para que se possa determinar o que o software deverá fazer. MARTINS (2007, pg. 209) acrescenta que “o objetivo desse processo é determinar as características do sistema conforme observado pelo cliente, apontando o desenvolvimento na direção correta”.

De acordo com Kosianski e Soares (2007) as principais causas de falhas em projetos de software estão relacionadas a dificuldade de entender a necessidade do cliente, de forma que o levantamento correto de requisitos é ponto vital para a qualidade do software. Dentre as técnicas de levantamento de requisitos existentes utilizou-se neste trabalho, a técnica de Questionário, sendo esta, essencial na obtenção de maiores informações e de acordo com as respostas, é possível abstrair as reais necessidades do cliente (KOSIANSKI e SOARES 2007). Segundo Martins (2007) após ser definido as características do sistema através dos requisitos levantados, estes requisitos devem ser traduzidos em requisitos detalhados, por meio da UML, de modo que seja possível modelar o sistema. A UML (Linguagem de Modelagem Unificada) como o próprio nome já indica é uma linguagem de modelagem de software. Segundo Guedes (2011), os diagramas UML auxiliam na definição das características do sistema, como seus requisitos, comportamento, estrutura lógica, dinâmica de seus processos e até necessidades físicas em relação a recursos de hardware. Guedes (2011) ainda afirma que os objetivos dos diagramas UML é fornecer múltiplas visões do sistema a ser modelado, analisando-o e modelando-o sob diversos aspectos, com o objetivo de atingir a completitude da modelagem. Considerando a importância desta informação foi utilizado para o desenvolvimento deste projeto os digramas de Caso de Uso, digrama de Classes.

2.1.4 GESTÃO DE PROJETOS

As técnicas e conceitos de gestão de projetos foram aplicadas a este projeto objetivando a elaboração de cronogramas, controle de qualidade, gerência de recursos, de escopo de interfaces, conforme especificação de Molinari. Segundo Molinari (2004, p.29):

Projeto é definido como um empreendimento em que recursos humanos, materiais e financeiros são organizados de maneira distinta, para atingir um único escopo de trabalho de uma dada especificação, dentro de limitações de custo e tempo, para obter uma mudança única e benéfica pela entrega de objetivos quantitativos e qualitativos.

Segundo Menezes (2009, pg.16), “Ter um planejamento orientado na criação, no desenvolvimento e em todas as fases intermediárias de validação facilita o lançamento desses produtos de modo mais controlado. Neste sentido, Molinari (2004) afirma que alguns dos benefícios encontrados ao se aplicar a gerência de projetos são: ajudar as organizações a atenderem às necessidades de seus clientes padronizando tarefas rotineiras e permitir a redução do número de tarefas de forma equilibrada e coerente, sem esquecer de nenhuma tarefa que deveria ser especificada e não foi.

3 DESENVOLVIMENTO DO SOFTWARE

Por meio da especificação dos requisitos, os mesmos foram categorizados em requisitos funcionais e não funcionais. O Quadro 01, ilustra a lista de alguns requisitos funcionais do sistema Pet Shop no ambiente usuário.

QUADRO 01 – Lista de Requisitos Funcionais - USUÁRIO

Prioridade	Identificador	Descrição	RNF
Alta	RF-01	O usuário deverá logar no sistema para poder acessá-lo.	01, 06, 09, 10, 11, 12, 13
Média	RF-02	O usuário poderá realizar cadastros.	01, 02, 03, 04, 05, 09, 10, 13

Fonte: Elaborado pelo autor.

O Quadro 02 apresenta alguns requisitos não-funcionais na categoria de usabilidade, que são alguns princípios onde facilite o uso para o usuário.

QUADRO 02 – Lista de Requisitos Não Funcionais - USABILIDADE

Identificador	Requisito não funcional
RNF-01	O sistema deverá possuir cores que evite o cansaço da visão.

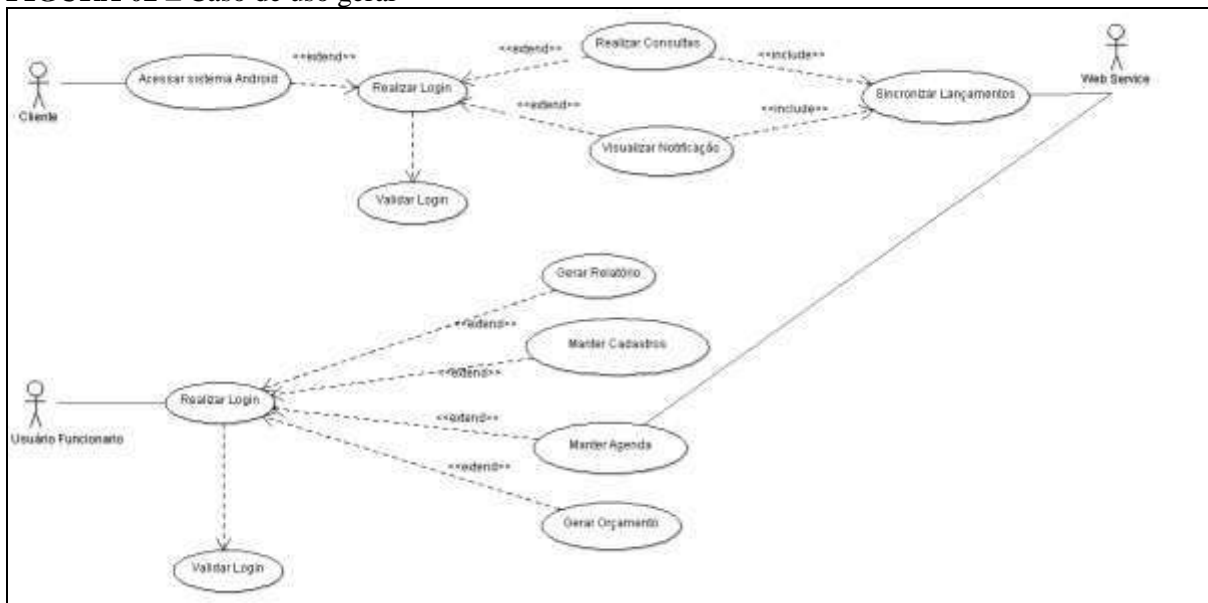
RNF-02

O sistema deverá possuir fontes que facilite a leitura dos textos, incluindo tamanhos que evitem o cansaço da visão.

Fonte: Elaborado pelo autor.

Ao se concluir todo o levantamento de requisitos funcionais e não funcionais do sistema Pet Software deu-se início ao desenvolvimento da modelagem do banco dados. Após ter sido definido as características do sistema através dos requisitos levantados, foi iniciado a modelagem do sistema através do diagrama de Caso de Uso representado na Figura 2.

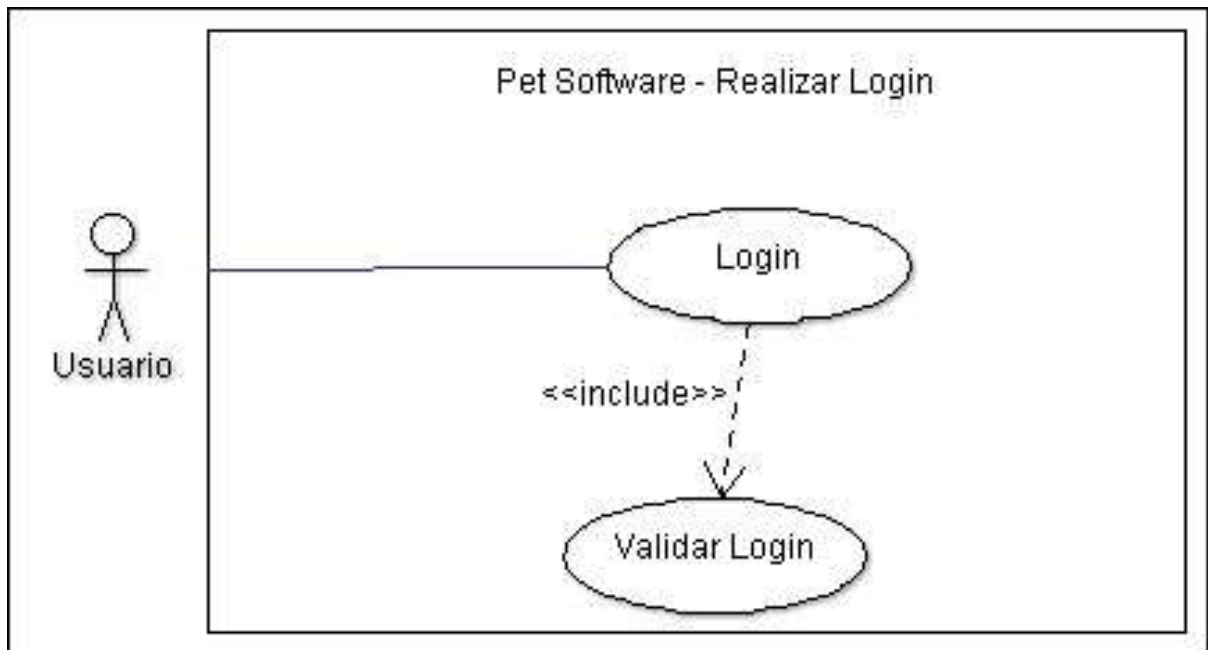
FIGURA 02 – Caso de uso geral



Fonte: Elaborado pelo autor

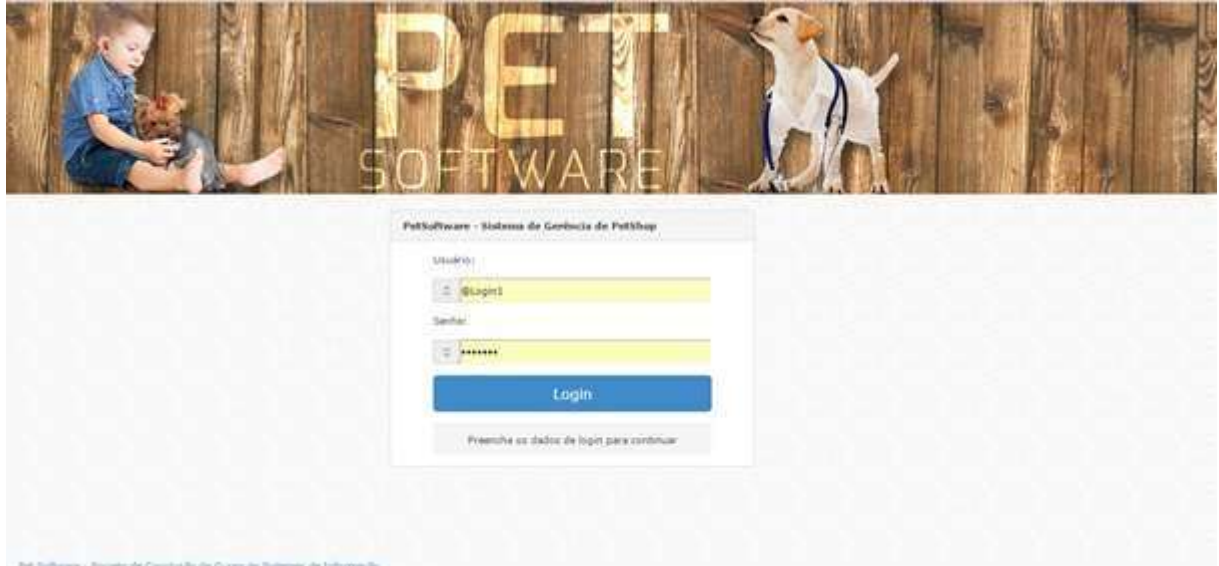
No processo de desenvolvimento do banco de dados, primeiramente elaborou-se o modelo lógico que apresenta um contexto mais lógico mostrando a sua estrutura composta por tabelas, campos e os tipos de cada um, por fim o modelo físico que é todo o código SQL. Após todos os requisitos e modelagens do banco de dados principiou o desenvolvimento do produto de *software* proposto neste trabalho. Elaborou-se o diagrama de caso de uso “Realizar login”, conforme ilustra a Figura 03 mostrando que o usuário deverá se autenticar fornecendo usuário e senha para acessar o sistema.

FIGURA 03 – Diagrama de caso de uso da página de login do usuário.



Com base no diagrama de caso de uso *Login*, desenvolveu-se a tela do software conforme ilustra a Figura 05.

FIGURA 05 – Tela de *login* do usuário.

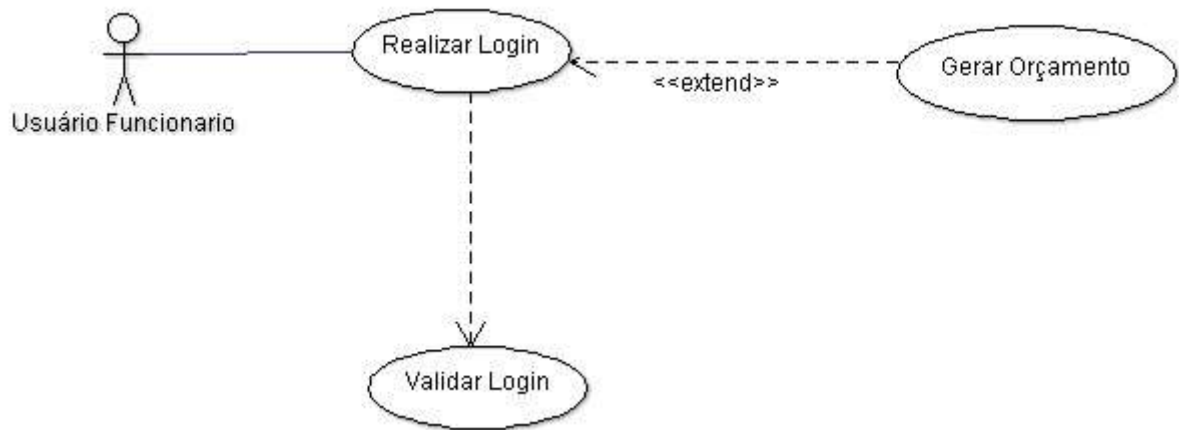


Fonte: Elaborado pelo autor.

É mostrado na imagem acima a tela de *login* do sistema onde o usuário deverá informar seu nome de usuário e senha para ter acesso as áreas restritas do sistema.

Elaborou-se o diagrama de caso de uso “Gerar Orçamento”, conforme ilustra a Figura 06.

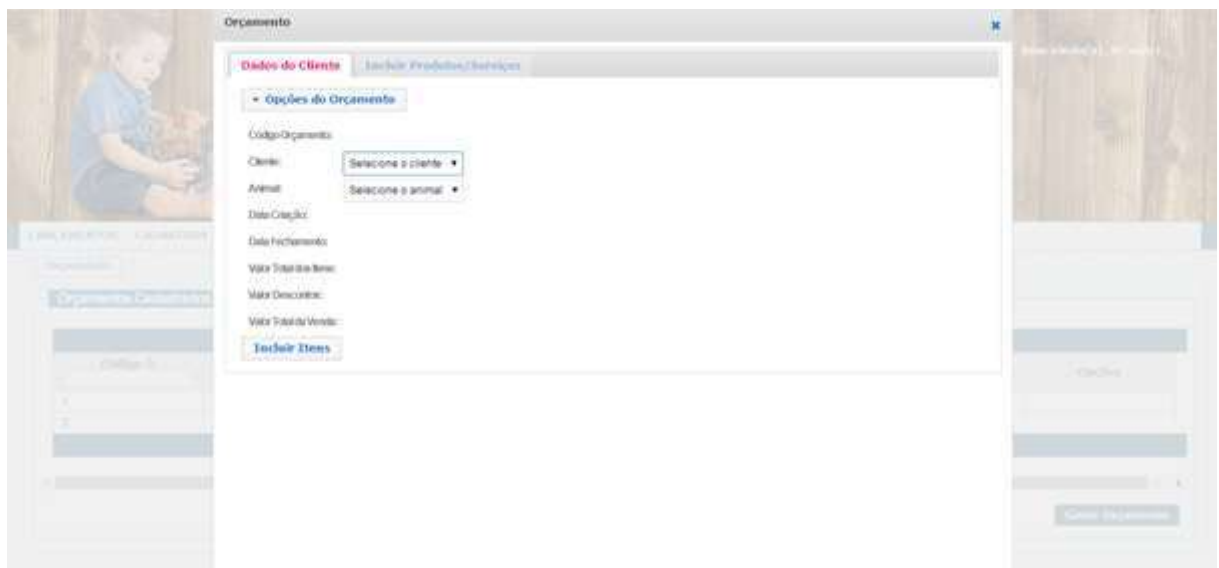
FIGURA 06 – Diagrama de caso de uso da tela de Orçamento.



Fonte: Elaborado pelo autor

Como pode-se observar, após logar no sistema o usuário poderá Gerar Orçamento de serviços. Com base no diagrama de caso de uso anterior, desenvolveu-se a tela de orçamento do sistema conforme ilustra a Figura 07.

FIGURA 06 – Tela de Orçamento – Aba de Cabeçalho



Fonte: Elaborado pelo autor.

3 CONCLUSÕES

Após a finalização do projeto pode ser verificado que o software desenvolvido atendeu ao objetivo especificado inicialmente se mostrando uma ferramenta eficaz para auxiliar a empresa no desenvolvimento de suas atividades, levando mais segurança e transparência aos processo de trabalho. Mas o mercado não é estático e da mesma forma o software para se manter no mercado também não o pode ser, sendo assim tem-se a intenção de crescer futuramente a ao sistema Web rotinas para controle financeiro, envio automático de *email* para alertar o cliente sobre a proximidade de data para revacinação. Para o sistema *Android* pretende-se criar a possibilidade do próprio cliente realizar seus agendamentos, agendamentos estes que serão confirmados por um funcionário da empresa através do sistema Web de forma que o cliente possa receber através do sistema *mobile* a confirmação do agendamento ou a sugestão de novas datas para o atendimento. E para que a empresa possa conhecer melhor os desejos e necessidades do cliente pretende-se desenvolver uma rotina que possibilite ao cliente avaliar atendimentos recebidos e também apresentar críticas, elogios ou sugestões de melhoria.

REFERÊNCIAS

- ABLESON, W.Frank; COLLINS, Charlie; SEN, Robi. **Android em ação**. 3. ed. Rio de Janeiro: Elsevier, 2012.
- ALVES, William Pereira. **Banco de dados: teoria e desenvolvimento**. São Paulo: Érica, 2009.
- COBRA, Marcos. **Administração de marketing no Brasil**. 3. ed. Rio de Janeiro: Elsevier, 2009.
- CORDEIRO, Gilliard. **Aplicações JAVA para web com JSF e JPA**. São Paulo: Casa do Código, 2012
- DEITEL, P.J. **Java: como programar**. Tradução de Edson Furmankiewicz. 8. ed. São Paulo: Pearson Prentice Hall, 2010.
- DEVICEATLAS. Uso da Web por Sistemas Operacionais por país. Disponível em: <https://deviceatlas.com/device-data/explorer/webusage-by-country/traffic/notablet/country/br/type/os_name>. Acesso em 27 Mai. 2014.
- GUEDES, Gilleanes T. A. **UML 2: uma abordagem prática**. 2. ed. São Paulo: Novatec Editora, 2011.
- LUCKOW, Décio Heinzemann; MELO, Alexandre Altair. **Programação Java para a Web**. São Paulo: Novatec, 2010.
- MACHADO, Francis Berenger; MAIA, Luiz Paulo. **Arquitetura de sistemas operacionais**. 4. ed. Rio de Janeiro: LTC, 2011.

MARTINS, José Carlos Cordeiro. **Gerenciamento de Projetos e Desenvolvimento de Software com PMI, RUP e UML**. 4. ed. Rio de Janeiro: Brasport, 2007.

MEZENES, Luís César de Moura. **Gestão de Projetos**. 3. ed. São Paulo: Atlas, 2009.

MOLINARI, Leonardo. **Gestão de Projetos: Técnicas e Práticas com Ênfase em Web**. São Paulo: Érica, 2004.

PORTAL G1 – **Brasil o segundo maior mercado de pet do**. Disponível em <http://g1.globo.com/hora1/noticia/2015/04/brasil-e-o-segundo-maior-mercado-de-petdo-mundo-so-perde-para-os-eua.html>. Acesso em: 06 jun. 2015.

SQLITE DOCUMENTS - **About SQLite**. Disponível em: <>. Acesso em: 18 mai. 2014.

TIOBE Company – **TIOBE Index for May 2015**. Disponível em: <http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>. Acesso em: 02 jun. 2015.

TUCKER, Allen B. **Linguagens de programação: princípios e paradigmas**. 2. ed. São Paulo: McGraw-Hill, 2008.

TURBAN, Efrain; RAINER, R. Kelly; POTTER, Richard E. **Administração de tecnologia da informação: teoria e prática**. Rio de Janeiro: Elsevier, 2005.

KOSIANSKI, André. **Qualidade de software: aprenda as metodologias e técnicas mais modernas para o desenvolvimento de software**. São Paulo: Novatec Editora, 2007.

SILVA, Pedro Tavares. *et al.* **Segurança dos Sistemas de Informação: Gestão estratégica da Segurança Empresarial**. Lisboa, Portugal: Centro Atlântico, 2013