**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA**

**ETEC PHILADELPHO GOUVÊA NETTO**

**ENSINO TÉCNICO INTEGRADO AO MÉDIO EM INFORMÁTICA**

**GIOVANA CACIOLI**

**HELOIZA DOS SANTOS**

**HENRIQUE DORETO**

**IGOR GOMES**

**HIGH PET**

**SÃO JOSÉ DO RIO PRETO – SP**

**2016**

**GIOVANA CACIOLI**

**HELOIZA DOS SANTOS**

**HENRIQUE DORETO**

**IGOR GOMES**

**HIGH PET**

Trabalho apresentado ao centro estadual de educação tecnológica Paula Souza ETEC Philadelpho Gouvêa Netto, como pré-requisito na obtenção do título de técnico em informática. Orientadores: Msc. Humberto Cecconi e Msc. Camila Brandão Fantozzi.

**SÃO JOSÉ DO RIO PRETO – SP**

**2016**

GIOVANA CACIOLI

HELOIZA DOS SANTOS

HENRIQUE DORETO

IGOR GOMES

HIGH PET

ENSINO TÉCNICO INTEGRADO AO MÉDIO EM INFORMÁTICA

BANCA EXAMINADORA

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Orientador: Msc. Humberto Cecconi

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Orientador: Msc. Camila Brandão Fantozzi

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Professor: Esp. Alexei Bueno

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Professora: Esp. Rosana Régia

Resultado: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

SÃO JOSÉ DO RIO PRETO 21/11/2016

Dedicamos este trabalho aos nossos professores, que nos ensinaram o melhor, e que tiveram paciência de nos ensinar mesmo com todas as dificuldades. E também dedicamos a nossa escola que tem nos acolhido como exemplos de alunos.

**AGRADECIMENTOS**

Agradecemos primeiramente a Deus que nos deu a oportunidade de estar em uma escola ótima, agradecemos aos nossos pais por terem paciência e amor e de quererem o melhor para nós e pelo apoio ao longo do curso, enfim aos professores pelos conhecimentos transmitidos, pelos conselhos e por todo amor e carinho em querer nos ensinar.

“Julgue seu sucesso pelas coisas que você teve que renunciar para conseguir. ”

Dalai Lama

**RESUMO**

O trabalho tem a função de possibilitar ao público alvo uma forma informatizada de gerenciar a fidelidade de seus clientes, e poder ter uma forma interativa de lidar com os seus clientes fiéis e lhes retribuírem com bônus.

Esses bônus de fidelidade serão feitos por meio de pontuação que o cliente do *Pet Shop* irá adquirir pelos serviços feitos ao seu animal de estimação, cada tipo de serviço terá uma pontuação diferente de acordo com o seu preço.

Palavras Chaves: *Pet Shop*; Bônus de Fidelidade; Forma informatizada; Gerenciar; Pontuação; Animal; Serviços.

**ABSTRACT**

Work will have the function to possibility the target audience a computerized form of managing the fidelity of its customers, and be able to have an interactive way to deal with their loyal customers and give back to them with bonuses.

These loyalty bonuses will be made through the costumer score of *Pet Shop* will acquire the services made to your *pet,* each type will have a different score according to its price.

Key words: *Pet Shop*; Loyalty bonuses; Computerized form; Managing; Score; Pet; Services.

**LISTA DE ILUSTRAÇÕES**

[Figura 1- Logotipo do Visual Studio 15](#_Toc467665538)

[Figura 2- Logotipo da Microsoft Sql Sever 2012 16](#_Toc467665539)

[Figura 3- Logotipo do CorelDraw x6 17](#_Toc467665540)

[Figura 4-Icone do brModelo 18](#_Toc467665541)

[Figura 5- Diagrama Entidade-Relacionamento 21](#_Toc467665542)

[Figura 6- Tela de Login do software HIGH PET 23](#_Toc467665543)

[Figura 7- Tela para Cadastrar funcionário do software HIGH PET 23](#_Toc467665544)

[Figura 8- Tela para fazer alterações no cadastro do funcionário 24](#_Toc467665545)

[Figura 9- Tela de menu do software HIGH PET 24](#_Toc467665546)

[Figura 10- Tela de cadastro de Clientes no software HIGH PET 25](#_Toc467665547)

[Figura 11- Tela de consulta de clientes 26](#_Toc467665548)

[Figura 12-Tela de cadastro de serviços 26](#_Toc467665549)

[Figura 13- Telas de consulta de serviços 27](#_Toc467665550)

[Figura 14-Tela de bônus fidelidade 28](#_Toc467665551)

**LISTA DE TABELAS**

[Tabela 1 - Executável do *Visual Studio* 15](#_Toc467067466)

**LISTA DE ABREVIATURAS E SIGLAS**

APIs Interface de Programação de Aplicativos

C# CSharp;

DER Diagrama Entidade-Relacionamento

INT Inteiro

LINQ Consulta integrada à linguagem;

MER Modelo Entidade-Relacionamento

NET Network;

SQL Linguagem Estruturada de Consultas

W3C World Wide Web Consortium

XML Linguagem Extensiva de Marcação Genérica

Frm Formulários

SUMÁRIO

[1. INTRODUÇÃO 13](#_Toc468096179)

[2. OBJETIVOS 14](#_Toc468096180)

[2.1. Objetivos Gerais 14](#_Toc468096181)

[2.2. Objetivo específico 14](#_Toc468096182)

[3. FUNDAMENTAÇÃO TEÓRICA 15](#_Toc468096183)

[3.1. Ferramentas 16](#_Toc468096184)

[3.1.1. *Visual Studio* 16](#_Toc468096185)

[3.1.2. MS SQL Server 17](#_Toc468096186)

[3.1.3. *CorelDraw* 18](#_Toc468096187)

[3.1.4. brModelo 19](#_Toc468096188)

[3.2. Linguagens 20](#_Toc468096189)

[3.2.1. C#: 20](#_Toc468096190)

[3.2.2. SQL: 20](#_Toc468096191)

[4. DESENVOLVIMENTO 21](#_Toc468096192)

[4.1. Banco de Dados 21](#_Toc468096193)

[4.1.1. Tipos de dados de um banco 21](#_Toc468096194)

[4.1.2. Diagrama de Entidade e Relacionamento 22](#_Toc468096195)

[4.2. Interface Gráfica do *software* 24](#_Toc468096196)

[4.2.1. Tela inicial de Login (frm\_login) 24](#_Toc468096197)

[4.2.2. Tela de Cadastro de funcionário (frm\_cadastrofunc) 24](#_Toc468096198)

[4.2.3. Tela de Alteração de Dados de Funcionário (frm\_Alterfunc) 25](#_Toc468096199)

[4.2.4. Tela Menu Principal (Frm\_Menu) 25](#_Toc468096200)

[4.2.5. Tela de Cadastro/Alterar/Excluir do Cliente (Frm\_Cadastro) 26](#_Toc468096201)

[4.2.6. Tela para Consultar Pets (Frm\_ConsultaPets) 27](#_Toc468096202)

[4.2.7. Tela de Cadastro de Serviço (Frm\_Serviço) 27](#_Toc468096203)

[4.2.8. Tela de Consulta de Serviços (Frm\_ConsultaServ) 28](#_Toc468096204)

[4.2.9. Tela de Bônus Fidelidade (Frm\_BonusFidelidade) 29](#_Toc468096205)

[5. CONCLUSÃO 30](#_Toc468096206)

[6. REFERÊNCIAS 32](#_Toc468096207)

[7. GLOSSÁRIO 35](#_Toc468096208)

[8. APÊNDICES 36](#_Toc468096209)

[8.1. Script do Banco 36](#_Toc468096210)

# INTRODUÇÃO

Desde o início quando escolhemos a melhor ideia para o grupo, o principal obstáculo foi tentar fazer inovações para o tema. Assim, fazendo pesquisas em algumas pequenas empresas de *Pet Shop* percebeu-se que: havia a necessidade de um sistema (*software* ou programa) com um recurso para facilitar o atendimento aos clientes “fieis”, registrando-os no banco de dados e de uma forma que também se possa organizar os seus clientes por meio de interfaces gráficas. Então foi pensado em fazer o gerenciamento dos pontos de fidelidade do cliente que serão descontados de acordo com os benefícios que lhes serão concedidos pelo *Pet Shop*, por meio da frequência de visitas com uso de serviços (banho, tosa, hidratação) nas ‘empresas que adquirirem o programa.

Outro ponto que também teve problemas foi na hora de nomear o recurso principal do sistema para tais empresas, a princípio colocou-se o nome popular da forma manual de fazer um gerenciamento desse ponto, o nome de “cartão fidelidade”, esse nome não era adequado ao tipo de trabalho digitalizado, pois teria que fazer funcionar essa ideia em um computador comum, sem uma máquina para ler um cartão. A ideia foi encontrar um nome chamativo para apresentar esse recurso do sistema aos usuários do mesmo, que foi “bônus fidelidade”. O *software* foi nomeado como “HIGH PET”, com decisão de todos integrantes do grupo.

Ao final de tudo, a proposta foi criar um *software* com um novo recurso para que se passe também averiguar a fidelidade do cliente com a empresa, podendo assim gerenciar bônus por meio de pontos, aumentando assim o vínculo da empresa com os seus clientes e vice-versa.

# OBJETIVOS

## Objetivos Gerais

O objetivo em relação ao mercado do *Pet Shop* é criar novas ideias e simplificar ou facilitar o gerenciamento da fidelidade dos clientes por meio dos funcionários da empresa, assim estreitando as relações comerciais com os clientes.

Tem no *software* HIGH PET: Cadastro do cliente, ficha para o animal e gerenciamento de pontos para os bônus de fidelidade dos clientes da empresa. O público alvo é os proprietários de *Pet Shop*. O *software* administra a ficha do animal e do seu dono e também as dos pontos para os bônus de fidelidade.

Em muitos *Pet*s esse trabalho é manual, ou seja, uso de papeis. O que dificulta na hora de procurar algum arquivo, pode chegar a perder algum documento, ou registros, muitos chegam se atrapalhar quando querem usar o computador.

A função do HIGH PET é informatizar esse processo, fazer com que o trabalho seja simples para todos os usos dentro do *Pet Shop*. A rapidez e eficácia com o uso do sistema que o *Software* está propondo. Fazendo com que o funcionário trabalhe melhor e com facilidade e os seus clientes fiquem satisfeito.

## Objetivo específico

O objetivo é desenvolver um *software* direcionado para um estabelecimento de *Pet Shop*. Será um *software* que irá ajudar nos registros, e que terá benefício de fidelidade, o cliente irá acumular pontos que serão descontados na troca de serviços gratuitos com o auxílio do programa HIGH PET.

O estabelecimento deverá ficar mais prático, tanto para a própria empresa, quanto para os clientes. O que se quer alcançar com esse novo programa é: mais rapidez, praticidade e eficácia para ambos os lados.

Enfim a visão seria um *software* que possa resolver problemas de vários *Pets Shops*, passando a adquirir essa nova funcionalidade oferecida. E a missão seria ser reconhecido pelos *Pets Shops* em geral por ter desenvolvido um *software* que possa suprir as suas necessidades.

# FUNDAMENTAÇÃO TEÓRICA

O mercado de *Pet Shop* no Brasil tem se mostrado um fragmento promissor para os empreendedores. Afinal, com tantos animais de estimação nos lares brasileiros, há cada vez mais gente investindo em produtos e serviços voltados a cães e gatos.

“De acordo com a Pesquisa Nacional de Saúde (PNS) com dados de 2013, realizada pelo Instituto Brasileiro de Geografia e Estatística (IBGE) e divulgada em junho de 2015, 44,3% dos domicílios brasileiros o que equivale a 28,9 milhões tenham pelo menos um cachorro. Também 17,7% dos domicílios contavam com ao menos um gato: eram 11,5 milhões de lares com felinos. ” – Destino e Negócio.

Os cachorros, por exemplo, necessitam de banho, tosa, e obviamente, ração. Essa totalidade de animais presentes nos lares brasileiros não poderia ter outro efeito, senão o aumento dos investimentos em *Pet Shop*s, produtos e serviços.

Geralmente, um *Pet Shop* no Brasil integra diversos setores relacionados aos animais de estimação. A lista dos investimentos é grande, pois é necessário oferecer diversos serviços, como banho, tosa e vacinas; e também é essencial vender produtos para animais, como: alimentos, medicamentos, acessórios e roupas.

Segundo a Associação Brasileira da Indústria de Produtos para Animais de Estimação (ABINPET) “o Brasil tem a segunda maior renda do mundo no segmento *pet* e veterinário, ficando atrás apenas dos Estados Unidos”.

## Ferramentas

### *Visual Studio*

****

Figura 1- Logotipo do Visual Studio

Fonte: *The Register*

*Visual Studio* é um conjunto completo de ferramentas para construir aplicativos *desktop* e aplicativos *Web* empresariais desenvolvidos por equipes. Além de criar aplicativos individuais de alto desempenho, pode usar as poderosas ferramentas de desenvolvimento baseadas em componentes do *Visual Studio* e outras tecnologias para simplificar o trabalho em equipe no projeto, desenvolvimento e implantação de soluções empresariais.

|  |  |
| --- | --- |
| Edição | Aplicativo de instalação |
| *Visual Studio* *ultimate* | vs\_*ultimate*.exe |
| *Visual Studio* *Professional* | vs\_*profissional*.exe |
| *Visual Studio* *Premium* | vs\_*premium*.exe |

Tabela 1 - Executável do Visual Studio

FONTE: (msdn.microsoft.com/pt-br/library/52f3sw5c(v=vs.90).aspx)

### MS SQL Server

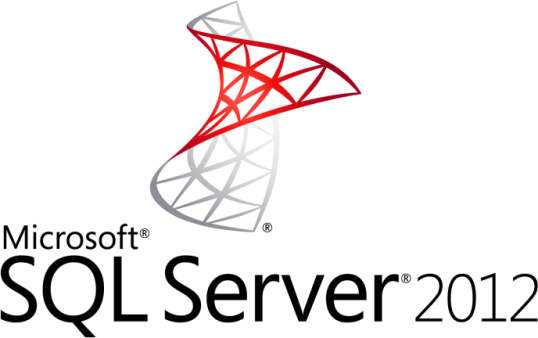


Figura 2- Logotipo da Microsoft Sql Sever 2012

Fonte:GetInto

O **MS SQL *Server***  é um sistema gerenciador de Banco de Dados (SGBD) desenvolvido pela Microsoft. Criado em parceria com a SYBASE em 1988 inicialmente para a plataforma OS/2. Parceria que durou até 1994, com o lançamento da versão para Windows NT e desde então a Microsoft mantém a manutenção do produto como um Banco de Dados. Sua principal função é a de armazenar e recuperar dados solicitados por outras aplicações de *software* seja aqueles no mesmo computador ou aqueles em execução em outro computador através de uma rede (incluindo a Internet).Resumindo: Linguagem SQL se trata de uma linguagem usada nos bancos de dados relacionais, enquanto que SQL Server e MySQL são sistemas para gerenciamento de banco de dados e que fazem o uso da linguagem SQL.

### *CorelDraw*



Figura 3- Logotipo do CorelDraw x6

Fonte:techtudo

No trabalho, em algumas partes gráficas irá usar o *CorelDraw*. Para logotipos ou para vetorizar imagens.

O *CorelDraw* é um programa para criação de desenhos com vetores, ou seja, retas com direção, sentido e comprimento. É muito utilizado por designers para fazer ilustrações e montar *layouts* diversos, pois disponibiliza uma ampla gama de recursos. (Luciana Vieira2012).

O *CorelDraw* surgiu em 1989, apenas em inglês. Em 1995, surgiu a primeira versão em 32 *bits* (*CorelDraw* 6). Dois anos depois surgiu a primeira versão para computadores *Macintosh*. No ano seguinte, foi lançada a primeira versão para Linux. Em 2003, surgiu a versão 12 que está preparada para o *Windows* XP. Sendo que a última versão criada no início de 2006 se denomina *Corel* X 3 (“X” em algarismos romanos = 10 + 3 = versão 13).

### brModelo

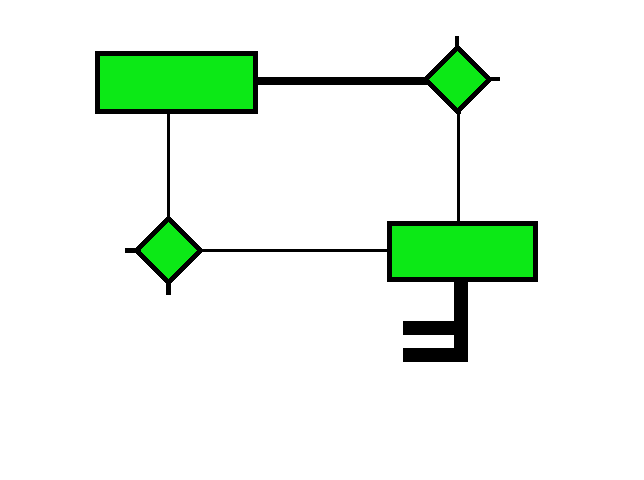


Figura 4-Icone do brModelo

Fonte: *Paint*

Para se fazer uma demonstração de um banco de dados uma das opções para se criar tal demonstração é o *software* brModelo. Ela permite que possa criar em duas formas práticas a representação de um banco de dados, o Modelo Entidade Relacionamento (MER) e Diagrama Entidade-Relacionamento (DER).

Para entender melhor essas duas formas de representação de um banco de dados. Vamos começar pela a mais simples das duas:

Modelo Entidade Relacionamento (MER): é a representação mais simples, bastante utilizada na engenharia de *software*, ele representa cada tabela (entidade) do BD na figura geométrica de um retângulo na grande maioria das vezes, e cada coluna (campo ou atributos) onde ficam as características de cada tabela abaixo, identificadas por meio de traços que saem direto do retângulo (tabela) que eles pertencem.

Diagrama Entidade-Relacionamento (DER): já essa segunda, é uma representação mais real (semelhante) a forma física do BD, cada entidade é literalmente é uma tabela e com isso possui colunas (campos) onde estão contidas as características de cada tabela.

.

## Linguagens

### C#:

Está sendo utilizada a Linguagem C# no *software*.

C# é uma linguagem elegante e de tipos protegidos, orientada a objeto e que permite aos desenvolvedores construírem uma diversidade maior de aplicações seguras e robustas, compatíveis com o 6*.NET Framework* (O .*NET Framework* é um conjunto de APIs desenvolvido pela Microsoft que serve de base para vários aplicativos). Segundo o site da Microsoft

A linguagem C# pode ser usada para criar vários aplicativos (serviços *Web* XML, aplicativos de banco de dados, etc.). Nela pode se encontrar ferramentas para designers de interface gráfica, editor de código entre outras ferramentas.

O C# é familiar com o C++ e com Java. Sendo uma linguagem mais expressiva e complexa, porém, fornecendo recursos melhores.

### SQL:

Fundada pela IBM e uniformizada por institutos como a ANSI (*American National Standards Institute*) e também pela ISO (*International Standard Organization*), com o alvo de unir um molde de linguagem para operações em banco de dados. Hoje se tornou a linguagem SQL(*Structure Querry Language*).

A linguagem Estruturada de Consultas foi desenvolvida para ser uma linguagem padrão para operações em B.D. (Banco de Dados).

A linguagem SQL é autônoma de *hardware* ou de *software*, ou seja, ao usar SQL no modelo padrão não precisa ter entendimento do s*oftware* de banco de dados envolvido ou sobre o funcionamento do *hardware* que vai abrir a instrução. Tudo que será necessário saber é a instrução SQL para solicitaras suas informações. A linguagem SQL possui comandos para gerar bancos de dados e tabelas, inserir, excluir, selecionar e alterar dados.

# DESENVOLVIMENTO

## Banco de Dados

### Tipos de dados de um banco

* **Chave Primaria:** É um campo ou conjunto de campos de uma tabela que identifica de uma forma exclusiva, cada registro armazenado na tabela;
* **Chave Estrangeira:** É o campo que estabelece o relacionamento entre duas tabelas. Então, uma coluna corresponde à mesma coluna que é a Chave primaria de outra tabela;
* **Ligação de (1,1):** Quando a ligação de uma tabela com a outra e única. Exemplo: (1) BONUS\_FIDELIDADE para (1) FICHA\_ANIMAL, ou seja, um bônus por *pet*;
* **Ligação de (1,n):** Quando a ligação de uma tabela pode se relacionar com vários elementos da outra tabela. Exemplo: (n) Vários TB\_ANIMAL para (1) TB\_DONO;
* **INT:** Apenas entrada de números Inteiros;
* **VARCHAR(N):** A letra “N” dentro dos parênteses representa a quantidade de caracteres permitida de caracteres que são tanto letras quanto números;
* **BOOLEAN():** Tipo de dado com os vetores “verdadeiro” e “falso”, que no caso no banco de dado do *software*, os vetores funciona como “sim” e “não”;
* **CHAR(N):** A quantidade de caracteres numéricos permitido;
* **NUMERIC (0,0):** A quantidade de números antes e depois da vírgula. Exemplo: (6,3) o máximo de caractere permitido seria 999,999;

### Diagrama de Entidade e Relacionamento

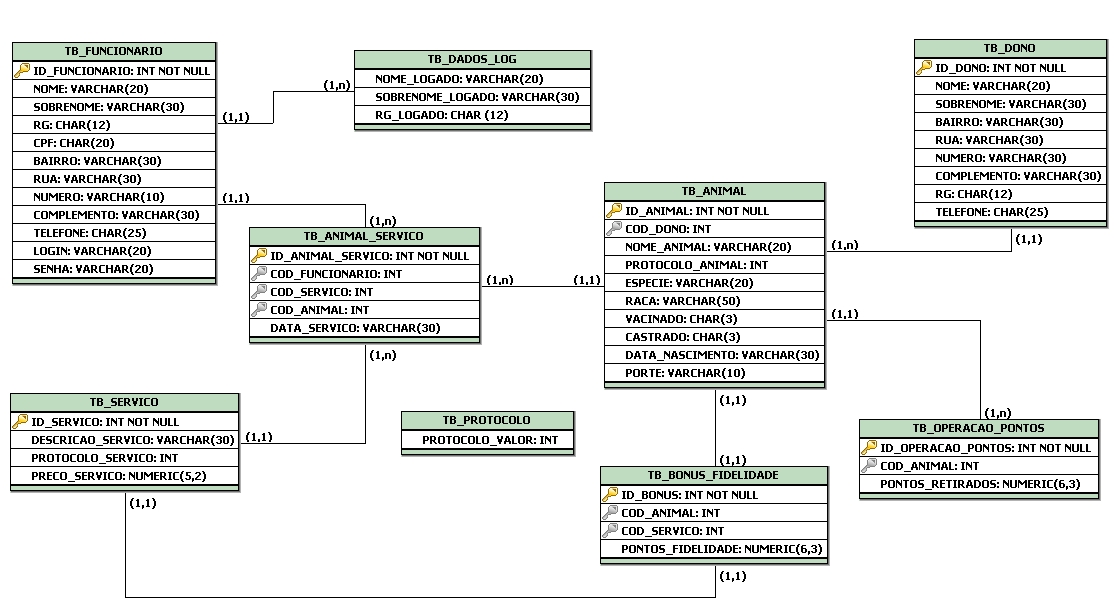


Figura 5- Diagrama Entidade-Relacionamento

Fonte: brModelo, Giovana Cacioli, Heloiza dos Santos, Henrique Doreto e Igor A. Gomes.

* **TB\_DONO:** Terá as informações do dono do animal, para que se possa saber a que cliente será concedido o bônus fidelidade pelos serviços ao animal.
* **TB\_FUNCIONÁRIO:** É o local onde ficam os registros dos funcionários que trabalham no *Pet Shop,* como o *Login* e a Senha para que somente pessoas relacionadas ao comércio tenha acesso ao programa.
* **TB\_SERVICO:** É a tabela que irá armazenar as informações do serviço que foi prestado ao animal, como descrição do serviço (se foi um banho, por exemplo), e o protocolo do serviço (PROTOCOLO\_SERVICO), que vai diferenciar um serviço do outro para os usuários. Dependendo do serviço que foi realizado terá uma pontuação que influenciara nos pontos de fidelidade.
* **TB\_ANIMAL:** A tabela responsável de ficar com as informações do animal, sendo relacionado à tabela referente ao seu dono. Os dados que serão armazenados aqui é como o nome do animal (NOME\_ANIMAL), o protocolo do animal (PROTOCOLO\_ANIMAL) entre outras referências necessárias para se administrar os animais que receberão os pontos de fidelidade.
* **TB\_OPERACAO\_PONTOS:** Irá armazenar os pontos que já foram removidos dos animais que receberão serviços da empresa, o que vai diferenciar cada animal para que esses registros sejam feitos corretamente é a chave-estrangeira (coluna horizontal com a abreviatura COD\_ANIMAL) dos pets, que é um valor inteiro, que nunca se repete no banco de dados.
* **TB\_BONUS\_FIDELIDADE:** Aqui ficara a pontuação total de fidelidade feita pelos clientes, onde serão todas somadas por meio de um comando *select* da linguagem SQL que é utilizada para manipular o banco de dados.
* **TB\_ANIMAL\_SERVICO:** É a parte (tabela) responsável no banco de dados em: armazenar o registro completo do serviço prestado, que terá a informação do serviço que foi feito, o funcionário que fez e o animal que recebeu o serviço.
* **TB\_PROTOCOLO:** Possui uma função simples no banco de dados do HIGH PET, mas fundamental para o funcionamento de todo o programa, essa tabela será responsável em guardar o ultimo protocolo que foi gerado no *software*, independente se foi um protocolo para o serviço ou para o animal que recebeu o mesmo.
* **TB\_DADOS\_LOG:** A função dessa tabela e simplesmente guardar o usuário que fez o *login* e quando *deslogar* e fizer o *login* em outra conta esse dados serão alterados.

## Interface Gráfica do *software*

### **Tela inicial de Login (frm\_login)**

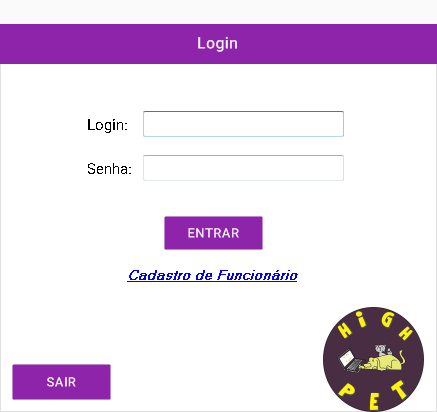


Figura 6- Tela de Login do software HIGH PET

Fonte: *Visual Studio*

Esta tela será utilizada pelo funcionário para entrar no menu principal do HIGH PET, ou realizando seu cadastro.

### Tela de Cadastro de funcionário (frm\_cadastrofunc)



Figura 7- Tela para Cadastrar funcionário do software HIGH PET

Fonte: *Visual Studio*

Este formulário realizará o cadastramento do funcionário, colocando os seus dados nos campos. Assim concluindo o seu cadastro.

### **Tela de Alteração de Dados de Funcionário (frm\_Alterfunc)**

****

Figura 8- Tela para fazer alterações no cadastro do funcionário

Fonte: *Visual Studio*

Este formulário trará opções em relação ao cadastro do funcionário, sendo elas a alteração e exclusão dos dados, porém, para fazer a alteração ou a exclusão é preciso pesquisar o cadastro do funcionário pelo seu RG.

### **Tela Menu Principal (Frm\_Menu)**



Figura 9- Tela de menu do software HIGH PET

Fonte: *Visual Studio*

Este formulário tem como funcionalidade abrir as ferramentas do *Software* HIGH PET sendo eles alterar funcionário, cadastro de animais, cadastro de serviços, lista de animais, lista de serviço e o Bônus Fidelidade.

### Tela de Cadastro/Alterar/Excluir do Cliente (Frm\_Cadastro)

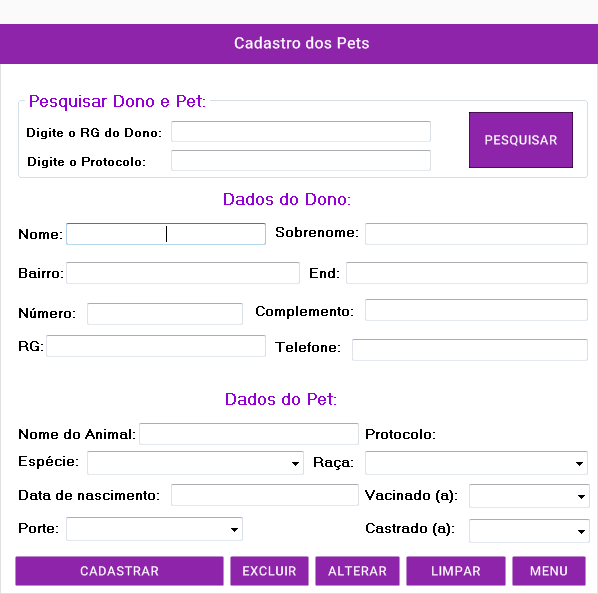


Figura 10- Tela de cadastro de Clientes no software HIGH PET

Fonte: *Visual Studio*

O formulário cadastro é aonde o funcionário irá registrar, alterar ou até mesmo excluir algum cliente, o cadastro de um cliente funciona assim: O funcionário registra o dono e o animal juntos, se o cliente tiver mais de um animal o funcionário terá que cadastrar o primeiro e pesquisar o RG do cliente para habilitar o botão de novo animal assim, cadastrando mais de um animal por cliente. O excluir e o alterar só poderão ser habilitados caso a pesquisa seja realizada. O protocolo situado e um “identificador” do animal para que possa realizar a pesquisa e assim podendo alterar o registro do animal

### Tela para Consultar Pets (Frm\_ConsultaPets)

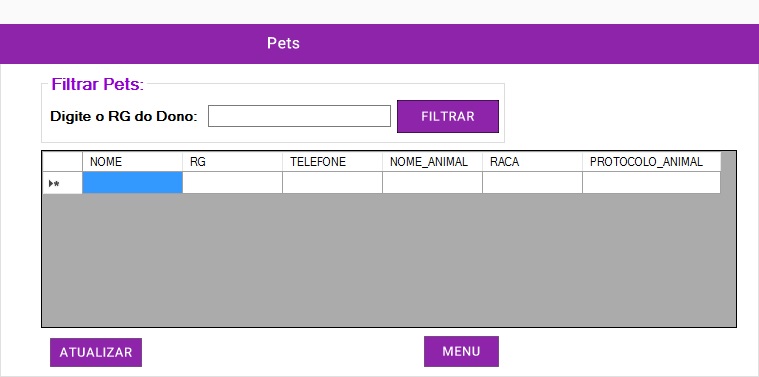


Figura 11- Tela de consulta de clientes

Fonte: *Visual Studio*

A função desse formulário e basicamente listar os dados mais importantes em uma tabela e mostrar para o funcionário.

### Tela de Cadastro de Serviço (Frm\_Serviço)

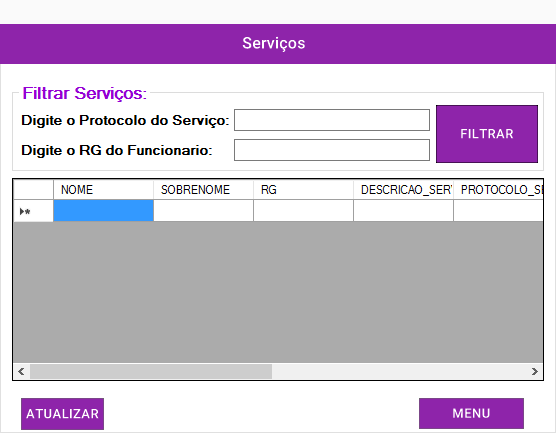


Figura 12-Tela de cadastro de serviços

Fonte: *Visual Studio*

O funcionário ira utilizar esse formulário para registrar os serviços prestados aos animais e já cadastrar o Bônus fidelidade clicando no botão “Gerar Pontos” e depois no “Cadastrar”. O protocolo irá funcionar do mesmo jeito que do frm\_cadastro.

### Tela de Consulta de Serviços (Frm\_ConsultaServ)



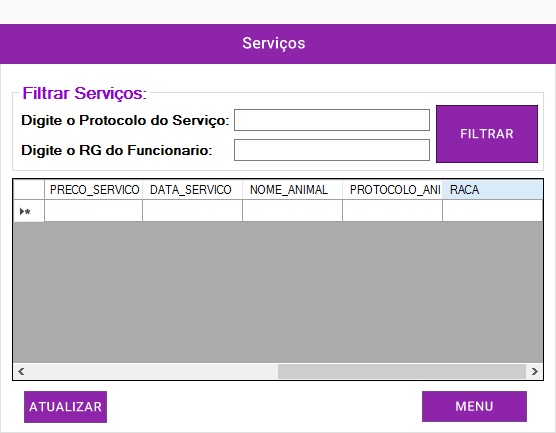


Figura 13- Telas de consulta de serviços

Fonte: *Visual Studio*

Este formulário de consulta contem tabelas de dados importantes de serviços prestados pelos funcionários, também possível à filtragem de serviços pelo RG do funcionário ou pelo protocolo de serviço.

### Tela de Bônus Fidelidade (Frm\_BonusFidelidade)

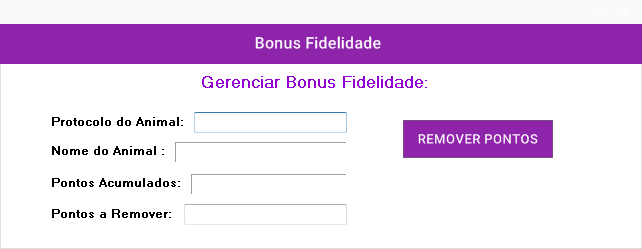


Figura 14-Tela de bônus fidelidade

Fonte: *Visual Studio*

O formulário de Bônus fidelidade é o formulário aonde vão os pontos dos clientes do *software* HIGH PET que se o *Pet Shop* quiser utilizar essa ferramenta poderá registra os pontos de acordo com o preço do serviço prestados.

# CONCLUSÃO

Desde o começo do desenvolvimento do *software*, teve-se várias escolhas que foram feitas sempre buscando as alternativas mais simples e práticas para solucionar todos os problemas existentes durante o desenvolvimento, primeiro buscou uma escolha fundamental para fazer-se a programa, que foi a forma de programação.

É muito comum nos dias de hoje programadores optarem por trabalharem em uma estrutura de classes, que é uma forma de programação bem dividida e de certa forma bastante complexa se tratando de um *software* do nível HIGH PET, por conta disso logo se percebeu que se precisava optar inicialmente por uma forma diferente de programação, mais simples e rápida, escolhendo assim desenvolver cada formulário individualmente com cada um tendo sua seus próprios códigos para acessarem o banco de dados e manipular os registros das tabelas, sendo que cada formulário iria fazer manipulações em tabelas diferentes no banco de dados. Mas ao terminar a maior da programação, foi criado uma classe *DataBase* para facilitar a conexão do *software* HIGH PET com o banco de dados.

Outro ponto em que ocorreram controvérsias foi em questão de como fazer uma diferenciação dos animais, e dos serviços que serão armazenados no programa, tendo a assim necessidade de criar (programar) uma forma de o sistema gerar protocolos para diferenciá-los, pois somente assim os usuários poderiam identificar e pesquisar animais e serviços feitos nas empresas de *Pet Shop*. Conseguimos solucionar esse problema, e nosso *software* cria os protocolos no momento de cadastrar os animais e os serviços e no banco de dados, possibilitando assim a distinção dos mesmos e uma forma de encontrá-los (pesquisá-los no sistema).

Mas o grande desafio foi em inventar uma maneira de se fazer o gerenciamento dos pontos de fidelidade que os animais dos clientes irão receber de acordo com o valor dos serviços prestados. E logo após acumular uma determinada quantidade de pontos, eles seriam descontados devido à troca com outros serviços que os animais dos clientes receberão gratuitamente por conta de sua fidelidade indicada pela sua pontuação no programa.

Pontuação essa que fica sobre critério do dono da empresa a respeito da quantidade necessária para que ocorra a troca por serviços gratuitos.

Concluiu-se então um software capaz de auxiliar funcionários de Pets Shops, no registro dos serviços prestados aos seus clientes e também no gerenciamento de pontos de fidelidade agradando os mesmos, não deixando de enfatizar o aumento do vínculo da empresa com sua clientela. Com interfaces de fácil entendido pelo usuário, para que tenha-se assim uma boa manipulação do programa.

Enfim toda base de estudo passada foi utilizada e aprimorada ao decorrer de todo o trabalho, para que pudesse assim complementar o software HIGH PET.

# REFERÊNCIAS

BIJORA, helito.O’que é NET framework; Como instalar o Microsoft .NET Framework no Windows 8; Disponível em:

<http://www.techtudo.com.br/dicas-e-tutoriais/noticia/2014/12/como-instalar-o-microsoft-net-framework-no-windows-8.html>; Acessado em: 01/05/2016às 15h:06m.

Blogdigicad.O’que é sqlserver; MS SQL Server, o que é, como funciona e para quem é direcionado; Disponível em:

<https://blogdigicad.wordpress.com/2014/05/05/ms-sql-server-o-que-e-como-funciona-e-para-quem-e-direcionado/>; Acessado em: 02/05/2016 às 08h:43m.

InfoEscola.O’que é sqlserver; MS SQL Server,SQL Server; Disponível em:

<http://www.infoescola.com/informatica/sql-server/>;Acessado em: 17/10/2016 as 11h:49m.

CIRIACO, douglas.O’que é API;O que é API?;Disponível em: <http://www.tecmundo.com.br/programacao/1807-o-que-e-api-.htm>; Acessado em: 01/05/2016 às 15h:39m.

Destino Negócio. Mercado de *Pet Shop*; mercado de *Pet Shop*; Disponível em: <http://destinonegocio.com/br/empreendedorismo/investir-no-mercado-pet-shop-no-brasil-pode-ser-promissor/>; Acessado em: 25/04/2016 às 10h:50m.

Intellectuale.O'que é SQL; O’ que é SQL?;Disponível em: <http://cursosdeprogramacao.com.br/blog/o-que-e-sql/>; Acessado em: 02/05/2016 às 09h:48m.

Luis.Qual a diferença entre Linguagem SQL, SQL Server e MySQL; Qual a diferença entre Linguagem SQL, SQL Server e MySQL?;Disponível em: <http://www.luis.blog.br/qual-a-diferenca-entre-linguagem-sql-sql-server-e-mysql.aspx>; Acessado em: 02/05/2016 às 09h:44m.

MARACOTTI, José; oque e LINQ; O que é LINQ; Disponível em: <http://www.macoratti.net/07/12/net\_linq.htm>; Acessadoem: 07/06/2016 às14h:34m.

Microsoft.O'que é Linguagem c#; Introdução à linguagem C# e ao .NET Framework; Disponivel em:

<https://msdn.microsoft.com/pt-br/library/z1zx9t92.aspx>; Acessado em: 01/05/2016 às 14h:09m.

Microsoft.O'que é *Visual Studio*; VisualStudio; Disponivel em: <https://msdn.microsoft.com/pt-br/library/52f3sw5c(v=vs.90).aspx>; Acessado em: 01/05/2016 às 13h:12m.

PEREIRA, Ana; O’que é XML; O que é XML?;Disponível em: <http://www.tecmundo.com.br/programacao/1762-o-que-e-xml-.htm>; Acessado em: 07/06/2016 às14h:40m.

Pet&Dicas. Preços de serviços no *Pet Shop*; Disponível em:

<http://www.petsedicas.com/2013/03/banho-e-tosa-preco.html>; Acessado em: 18/08/16 às 14h:08m.

Pet Link. Preços de serviços no *Pet Shop*; Disponível em:

<http://www.petlink.com.br/mp/precobt.html>; Acessado em: 18/08/16 às 14h:11m.

PetMais Rações e Suplementos. Preços de serviços no *Pet Shop*; Disponível em:

<http://*Pet Shop*emnatal.com.br/precos-de-banho-e-tosa/>; Acessado em: 18/08/16às14h:01m.

Siglas.O'que significa a sigla NET; NET; Disponível em: <http://www.siglaseabreviaturas.com/net/>; Acessado em: 01/05/2016 às 15h:42m.

Significados.O'que significa XML; Significado de XML; Disponível em: <http://www.significados.com.br/xml/>; Acessado em: 01/05/2016 às15h:44m.

VIEIRA. Luciana; O’que e *CorelDraw*; Como usar o *CorelDraw*; Disponível em: <http://www.techtudo.com.br/dicas-e-tutoriais/noticia/2012/03/como-usar-o-*CorelDraw*.html>; Acessado em: 03/05/2016 às15h:16m.

*Visual Studio*. Instalando o *Visual Studio*; Downloads do *Visual Studio*; Disponível em: <https://www.visualstudio.com/pt-br/downloads>; Acessado em: 01/05/2016às 14h:09m.

W3C; w3c; SOBRE O W3C; Disponível em:

<http://www.w3c.br/Sobre/>. Acessado em: 07/06/2016 às 14h:36m.

# GLOSSÁRIO

C *Sharp* (C#): é uma linguagem de “programação orientada a objetos”, desenvolvida pela empresa Microsoft na plataforma .*NET*, tendo como base a linguagens C++ e semelhança a outras linguagens Pascal e Java.

*Network* (NET) ou *Networking*: esse termo indica a capacidade de estabelecer uma rede de contatos ou uma conexão com algo ou com alguém, essa rede de contatos é um sistema de suporte onde existe a partilha de serviços e informação entre indivíduos.

APIs: é um conjunto de rotinas e padrões de programação para acesso a um aplicativo de *software* ou plataforma baseado na *Web*. A sigla API refere-se ao termo em inglês "*Application Programming Interface*" que significa em tradução para o português "Interface de Programação de Aplicativos".

XML: Do português, linguagem Extensiva de Marcação é recomendada pela empresa W3C para o desenvolvimento de documentos com dados organizados hierarquicamente, tais arquivos como textos, banco de dados ou desenhos vetoriais.

W3C: É um consórcio denominado *‘* é um consórcio internacional no qual organizações filiadas, uma equipe em tempo integral e o público trabalham juntos para desenvolver padrões para a Web.

SQL: É uma linguagem para consulta e alteração de um bando de dados

LINQ:É um conjunto de recursos introduzidos no .*NET* *Framework* 3.5 que permitem a realização de consultas diretamente em base de dados, documentos XML, estrutura de dados, coleção de objetos usando uma sintaxe parecida com a linguagem SQL.

# APÊNDICES

## Script do Banco

--CRIANDO BANCO DE DADOS

CREATE DATABASE BD\_PETSHOP;

--ABRINDO BANCO DE DAODOS

USE BD\_PETSHOP;

--TABELA DONO

CREATE TABLE TB\_DONO

(

ID\_DONO INT NOT NULL PRIMARY KEY IDENTITY(1,1),

NOME VARCHAR(20),

SOBRENOME VARCHAR(30),

BAIRRO VARCHAR(30),

RUA VARCHAR(30),

NUMERO VARCHAR(30),

COMPLEMENTO VARCHAR(30),

RG CHAR(12),

TELEFONE CHAR(25)

);

--TABELA FUNCIONARIO

CREATE TABLE TB\_FUNCIONARIO

(

ID\_FUNCIONARIO INT NOT NULL PRIMARY KEY IDENTITY(1,1),

NOME VARCHAR(20),

SOBRENOME VARCHAR(30),

RG CHAR(12),

CPF CHAR(20),

BAIRRO VARCHAR(30),

RUA VARCHAR(30),

NUMERO VARCHAR(10),

COMPLEMENTO VARCHAR(30),

TELEFONE CHAR(25),

LOGIN VARCHAR(20),

SENHA VARCHAR (20)

);

--TABELA SERVIÇO

CREATE TABLE TB\_SERVICO

(

ID\_SERVICO INT NOT NULL PRIMARY KEY IDENTITY(1,1),

DESCRICAO\_SERVICO VARCHAR(30),

PROTOCOLO\_SERVICO INT,

PRECO\_SERVICO NUMERIC(5,2)

);

--TABELA ANIMAL

CREATE TABLE TB\_ANIMAL

(

ID\_ANIMAL INT NOT NULL PRIMARY KEY IDENTITY(1,1),

COD\_DONO INT,

NOME\_ANIMAL VARCHAR(20),

PROTOCOLO\_ANIMAL INT,

ESPECIE VARCHAR(20),

RACA VARCHAR(50),

VACINADO CHAR(3),

CASTRADO CHAR(3),

DATA\_NASCIMENTO VARCHAR(30),

PORTE VARCHAR(10),

CONSTRAINT FK\_COD\_DONO

FOREIGN KEY (COD\_DONO)

REFERENCES TB\_DONO (ID\_DONO)

);

--TABELA OPERAÇÃO\_PONTOS

CREATE TABLE TB\_OPERACAO\_PONTOS

(

ID\_OPERACAO\_PONTOS INT NOT NULL PRIMARY KEY IDENTITY(1,1),

COD\_ANIMAL INT,

PONTOS\_RETIRADOS NUMERIC(6,3),

CONSTRAINT FK\_COD\_ANIMAL\_TB\_OPERACAO\_PONTOS

FOREIGN KEY (COD\_ANIMAL)

REFERENCES TB\_ANIMAL (ID\_ANIMAL)

);

--TABELA BONUS FIDELIDADE

CREATE TABLE TB\_BONUS\_FIDELIDADE

(

ID\_BONUS INT NOT NULL PRIMARY KEY IDENTITY(1,1),

COD\_ANIMAL INT,

COD\_SERVICO INT,

CONSTRAINT FK\_COD\_ANIMAL

FOREIGN KEY (COD\_ANIMAL)

REFERENCES TB\_ANIMAL (ID\_ANIMAL),

CONSTRAINT FK\_COD\_SERVICO

FOREIGN KEY (COD\_SERVICO)

REFERENCES TB\_SERVICO (ID\_SERVICO),

PONTOS\_FIDELIDADE NUMERIC(6,3)

);

--TABELA ANIMAL\_SERVIÇO

CREATE TABLE TB\_ANIMAL\_SERVICO

(

ID\_ANIMAL\_SERVICO INT NOT NULL PRIMARY KEY IDENTITY(1,1),

COD\_FUNCIONARIO INT,

COD\_SERVICO INT,

COD\_ANIMAL INT,

CONSTRAINT FK\_COD\_FUNCIONARIO

FOREIGN KEY (COD\_FUNCIONARIO)

REFERENCES TB\_FUNCIONARIO (ID\_FUNCIONARIO),

CONSTRAINT FK\_COD\_SERVICO\_TB\_ANIMAL\_SERVICO

FOREIGN KEY (COD\_SERVICO)

REFERENCES TB\_SERVICO (ID\_SERVICO),

CONSTRAINT FK\_COD\_ANIMAL\_TB\_ANIMAL\_SERVICO

FOREIGN KEY (COD\_ANIMAL)

REFERENCES TB\_ANIMAL (ID\_ANIMAL),

DATA\_SERVICO VARCHAR(30)

);

--TABELA PROTOCOLO

CREATE TABLE TB\_PROTOCOLO

(

PROTOCOLO\_VALOR INT

);

--INSERINDO O PRIMEIRO PROCOLO

INSERT INTO TB\_PROTOCOLO VALUES ('100001')

--TABELA DADOS DA PESSOA LOGADA

CREATE TABLE TB\_DADOS\_LOG

(

NOME\_LOGADO VARCHAR(20),

SOBRENOME\_LOGADO VARCHAR(30),

RG\_LOGADO VARCHAR(12)

);

--INSERINDO DADOS INICIAIS NA TB\_DADOS\_LOG

INSERT INTO TB\_DADOS\_LOG (NOME\_LOGADO, SOBRENOME\_LOGADO, RG\_LOGADO) VALUES ('0000000', '00000000', '00000000')