

UNIP – UNIVERSIDADE PAULISTA

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

PIM

PROJETO INTEGRADO MULTIDICIPLINAR

TEMA:

**Análise de Requisitos para um Sistema Geral de Gerenciamento de Frotas**

Semestre: 3ª semestre 2019

Alunos:

Daniel Henrique M. Linhares – D75GBB-1

Luis Gustavo B. Bravin – N27966-4

Matheus Leme dos Santos – N29221-0

Matheus Manabu P. Ikenaga – D6565F-1

Mauro Vieira Pesente – N307FE-5

**São Paulo**

**2019**

Alunos:

Daniel Henrique M. Linhares – D75GBB-1

Luis Gustavo B. Bravin – N27966-4

Matheus Leme dos Santos – N29221-0

Matheus Manabu P. Ikenaga – D6565F-1

Mauro Vieira Pesente – N307FE-5

**Análise de Requisitos para um Sistema Geral de Gerenciamento de Frotas**

Trabalho Semestral para obtenção do título de graduação em Análise e Desenvolvimento de Sistemas apresentado à Universidade Paulista – UNIP

Orientador: Emerson Abraham

**São Paulo**

**2019**

# RESUMO

Uma empresa de controle de frotas de veículos solicitou para uma equipe de desenvolvedores de Software para realizar, no momento, uma análise de requisitos para um sistema que gerencie a frota de carros de uma empresa no setor de telecomunicações. O sistema deve ter portabilidade para desktop, web e mobile, além de abranger todo o território nacional. A empresa de controle de frotas solicita também que ela possa usar esse sistema não só para o cliente de telecomunicações, mas para todos os outros clientes que ela pode vir a ter. O foco do projeto é o gerenciamento de veículos. A empresa deve auxiliar aos seus clientes a migração para veículos com fontes renováveis e limpas de energia. O sistema deverá contemplar algumas partes acessíveis para que eventuais usuários portadores de deficiência possam utilizá-lo, além de alguns dos itens já gerenciados pela empresa, como Veículos, Seguros, Motoristas (CNH, validade da Habilitação, etc.), Viagens (rotas, data de início e retorno etc.), Entrada/Saída (de garagens credenciadas ou estacionamento), Finanças, Estoque de peças, Pneus, Manutenções Corretivas e Preventivas, Abastecimentos, Quilometragem, Multas, Sinistros, Avisos automáticos (pelo celular ao motorista etc.), Relatórios (multas, consumo, revisões etc.), Aluguel de veículos ao cliente (frota interna pequena).

Palavras-Chave: Análise de Requisitos. Gerenciamento de Frotas. Sistema.

# ABSTRACT

A vehicle fleet control company has requested a team of Software developers to perform, at the time, a requirements analysis for a system that manages a company's fleet of cars in the telecommunications sector. The system must have portability for desktop, web and mobile, besides covering the entire national territory. The fleet control company also requests that it can use this system not only for the telecommunications client, but for all other customers it may have. The focus of the project is vehicle management. The company should assist its customers in migrating to vehicles with renewable and clean energy sources. The system should contemplate some accessible parts so that any disabled users can use it, in addition to some of the items already managed by the company, such as vehicles, insurance, drivers (CNH, validity of authorization to Drive etc.), travel (routes, departure/entry date, etc.), entry/exit (from accredited garages or parking), finance, parts inventory, tires, corrective and preventive maintenance, supplies, mileage, fines, claims, automatic notices (by mobile to Driver etc.), reports (traffic tickets, consumption, reviews etc.), vehicle rental to customer (small internal fleet).

Keywords: Requirements analysis. Fleet management. System.

Sumário

[RESUMO 3](#_Toc10757492)

[ABSTRACT 4](#_Toc10757493)

[1. Introdução 6](#_Toc10757494)

[2. Metodologia Utilizada 7](#_Toc10757495)

[3. Requisitos 8](#_Toc10757496)

[3.1.Levantamento de Requisitos 9](#_Toc10757497)

[4. UML 13](#_Toc10757498)

[4.1 Processos do Sistema 14](#_Toc10757499)

[5. Banco de Dados 30](#_Toc10757500)

[6. Manuais 40](#_Toc10757501)

[6.1 Manual de Instalação 40](#_Toc10757502)

[6.2 Manual ao Usuário 41](#_Toc10757503)

[7. Conclusão 45](#_Toc10757504)

[8. Referências 46](#_Toc10757505)

# Introdução

Uma empresa de gerenciamento de frotas recebeu um novo cliente, uma empresa de telecomunicações que possui sua própria frota para gerenciar e eventualmente alugar carros da empresa de gerenciamento. Por ser um cliente grande, a empresa de gerenciamento decidiu atualizar o método ao qual trabalha de uma planilha para um programa completo, decidindo contratar uma equipe de desenvolvimento de Softwares para desenvolver um sistema que gerencie não só a frota do cliente de telecomunicações, mas que seja possível de fazer a gerencia de todo e qualquer tipo de frota de veículos.

O sistema necessita realizar tudo que a empresa já realizava em suas planilhas, como por exemplo controle de veículos disponíveis, controle dos motoristas e sinistros, entre outros, além de o sistema também necessitar possuir portabilidade, ou seja, funcionar em várias plataformas como computadores e celulares, acessibilidade para portadores de deficiências além de avisos automático para os motoristas.

Por ser um sistema bem complexo a equipe de desenvolvimento decidiu começar estruturando todo o projeto antes de começar o desenvolvimento, realizando o levantamento de requisitos, elaboração dos diagramas de UML, estruturação do banco de dados, elaboração dos manuais de instalação e de treinamento dos usuários, além de outros requisitos necessários para o desenvolvimento do sistema.

Este trabalho contempla todos os levantamentos realizados pela equipe de desenvolvimento antes do começar a desenvolver todo o sistema.

# Metodologia Utilizada

Dada a proposta do trabalho que seria apenas o levantamento de requisitos, diagramas de UML e prototipação de telas, ou seja, houve um semestre para a coleta das informações e estruturação de toda a parte lógica do sistema, sem que seja necessário fazer a codificação do mesmo.

Escolhemos o Método Cascata para Desenvolvimento desse projeto dado o tempo disponível de coleta

O método cascata se baseia em:

1. Concepção – A ideia do Projeto.
2. Levantamento de Requisitos – Levantar as necessidades do sistema.
3. Análise de Requisitos – Entender o Procedimento.
4. Homologação de Requisitos – Concluir as informações obtidas.
5. Modelagem – Criar os diagramas UML para melhor entendimento.
6. Projeto – Construção do Software.
7. Testes – Caixa branca/ Caixa Preta.
8. Manutenção – Suporte ao sistema, correções e melhorias.

Neste projeto iremos fazer a construção do passo 1 até o 5

# Requisitos

O grupo pesquisou por softwares já existentes na área e foram analisados os sistemas: Master Frotas, Produsoft e Unifrota.

Antes de iniciar qualquer projeto, é necessário adquirir uma base conhecimento sobre o produto ao qual está trabalhando, levantar o que é necessário para realizar o sistema ideal para o usuário sem fugir do objetivo principal. Existem várias formas de levantar os requisitos, a escolhida para esse projeto foi a utilização de questionários enviados para empresas já estabelecidas na área de gerenciamento de frotas, o questionário foi desenvolvido buscando entender não só a própria função do gerenciamento como também entender a parte técnica do sistema, esse método é muito útil para obter as informações detalhadas porém não garante uma grande quantidade de respostas, e para complementar os dados adquiridos foi realizado analise das documentações disponíveis publicamente nos sites de softwares já existentes de gerenciamento de frotas, as informações resultantes adquiridas foram o suficiente para começar a desenvolver os requisitos para o sistema.

Foi enviado um e-mail com um questionário elaborado pelo grupo, para entender como é feita a gestão e outras informações com relação ao próprio sistema

O questionário foi:

1. Como é feita a Gerencia da Frota de Veículos?
2. Como é feita a Gerência dos Seguros?
3. Como é feito o Cadastro dos Motoristas?
4. A cobrança é feita por Quilometragem ou somente pela Diária?
5. Vocês possuem garagem ou estacionamentos credenciados? Se sim, como é gerenciada a entrada e saída de veículos?
6. Quais as principais receitas e despesas da empresa? (Não é necessário especificar)
7. Vocês possuem um estoque de peças e uma equipe especializada em manutenção ou fazem a manutenção através de uma equipe terceirizada?
8. Como é feito o Controle de Multas e Sinistros?
9. Como é feita a escolha dos veículos da Frota?
10. Vocês fazem algum laudo pericial ao receber o veículo de volta?
11. Vocês utilizam algum ripo de sistema em nuvem?
12. Qual a(s) linguagem(s) de programação utilizada(s) no desenvolvimento do sistema?
13. Qual o banco de Dados utilizado por vocês?
14. Com qual frequência é realizado o backup das aplicações e dos dados?
15. Vocês utilizam alguma metodologia de desenvolvimento? Se sim, qual?
16. O Sistema é multiplataforma?
17. O que você implantaria em seu sistema?
18. Como foram feitos os casos de teste do sistema?
19. Quais são as partes que interagem com o sistema?
20. Vocês possuem certificação ISO/IEC ? Se sim, qual(is) ?
21. Quais as principais funcionalidades do sistema?

Infelizmente, nenhuma respondeu ao questionário ou liberou acesso ao sistema para que fosse analisado

Então o levantamento foi completamente feito utilizando as documentações disponibilizadas pelas próprias empresas nos sites delas.

Os softwares de gestão presentes hoje no mercado, contém muitas informações nas telas e muito pouco intuitivo, com isso levantamos as informações para a construção do SIGEFROTAS que tem a proposta de ser um sistema em nuvem, responsivo, com um design inovador e completamente intuitivo.

## Levantamento de Requisitos

Um levantamento precisa registrar não só os requisitos funcionais, ou seja, o que realmente é necessário e desejado pelo cliente para o programa fique realmente útil para eles, como também os requisitos não funcionais de usabilidade, portabilidade, confiabilidade do sistema, etc. Os Requisitos ficaram da seguinte forma:

Tabela 1 – Requisitos do Sistema

|  |  |
| --- | --- |
| **Funcionais** | **Não Funcionais** |
| Gerencia de Veículos | Usabilidade |
| Controle de Seguros | Confiabilidade |
| Controle de Motoristas e Viagens | Portabilidade |
| Gestão de Finanças | Manutenibilidade |
| Controle de Peças e Estoque | Segurança |
| Manutenção dos Veículos | Disponibilidade |
| Controle de Multas e Sinistros | Performance |
| Gerencia de Aluguel de Veículos | Tolerância a Falha |

Fonte: Autor (2019)

Conforme visto na figura: “Tabela 1 – Requisitos do Sistema”, foram levantados 16 requisitos, sendo 8 Funcionais e 8 Não Funcionais para o SIGEFROTAS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID Requisito** | **Descrição** | **Prioridade** | **Caso de Teste** | **Data Inclusão** | **Data Conclusão** |
| RQF001 | Gerencia de Veículos | Essencial | Manter cadastro de veículos | 15/05/2019 |  |
| RQF002 | Controle de Seguros | Essencial | Manter e atrelar o cadastro do Seguro a um cadastro de veículo | 15/05/2019 |  |
| RQF003 | Controle de Motoristas e Viagens | Essencial | Ter registro de todas as viagens, local e data de saída e chegada | 15/05/2019 |  |
| RQF004 | Gestão de Finanças | Essencial | CRUD de contas a pagar/ receber e registrar suas baixas | 15/05/2019 |  |
| RQF005 | Controle de Peças e Estoque | Essencial | Efetuar o controle de estoque mínimo para funcionamento das manutenções | 15/05/2019 |  |
| RQF006 | Manutenção dos Veículos | Essencial | Avisar quando um veículo chegar na Quilometragem ou Data prevista para manutenção preventiva | 15/05/2019 |  |
| RQF007 | Controle de Multas e Sinistros | Essencial | Manter e atrelar o cadastro de multas e sinistros a um cadastro de veículo | 15/05/2019 |  |
| RQF008 | Gerencia de Aluguel de Veículos | Essencial | Ter registro de todos os alugueis de veículo e seus respectivos condutores | 15/05/2019 |  |
| RQNF001 | Usabilidade | Essencial | Ter padrões em telas, facilitando a capacidade de aprendizado intuitivo | 15/05/2019 |  |
| RQNF002 | Confiabilidade | Essencial | Possibilidade de manter registros ilimitados ao utilizar o sistema | 15/05/2019 |  |
| RQNF003 | Portabilidade | Essencial | Acesso em dispositivo Desktop, Tablet ou Mobile | 15/05/2019 |  |
| RQNF004 | Manutenibilidade | Essencial | Implementar melhoria ou correção no sistema de maneira eficiente | 15/05/2019 |  |
| RQNF005 | Segurança | Essencial | Autenticação de usuário para acesso ao sistema | 15/05/2019 |  |
| RQNF006 | Disponibilidade | Essencial | Utilização e consulta de Informações Cadastradas em qualquer navegador | 15/05/2019 |  |
| RQNF007 | Performance | Essencial | Ter tempo de resposta inferior à 2 minutos com 200 usuários simultâneos | 15/05/2019 |  |
| RQNF008 | Tolerância a Falha | Essencial | Ao ocorrer uma falha o sistema deve voltar a funcionar recarregando a página | 15/05/2019 |  |

Fonte: Autor(2019)

# UML

Unified Modeling Language mais conhecido como UML é uma linguagem única de modelagem para especificação, visualização, construção e documentação de um projeto de software, a modelagem é uma das principais operações que levam a implementação de um software, a UML gera diagramas que servem para seguir durante uma criação, e esses diagramas ajudam a visualizar o software de forma gráfica e entender como ele funciona, como se fosse a planta de um prédio a ser construído, tanto na visão do cliente como na visão da equipe que vai desenvolver a aplicação.

Existem vários diagramas que podem ser usados na linguagem UML, porem os utilizados nesse trabalho foram os diagramas de classe, caso de uso, sequência e atividades.

Descrevendo o que é cada um e para que serve, o diagrama de classe descreve a estrutura de um sistema, apresentando classes, atributos e métodos, esse diagrama é bastante útil pois define todas as classes que o sistema precisa ter, e cada classe do diagrama representa uma tabela do banco de dados.

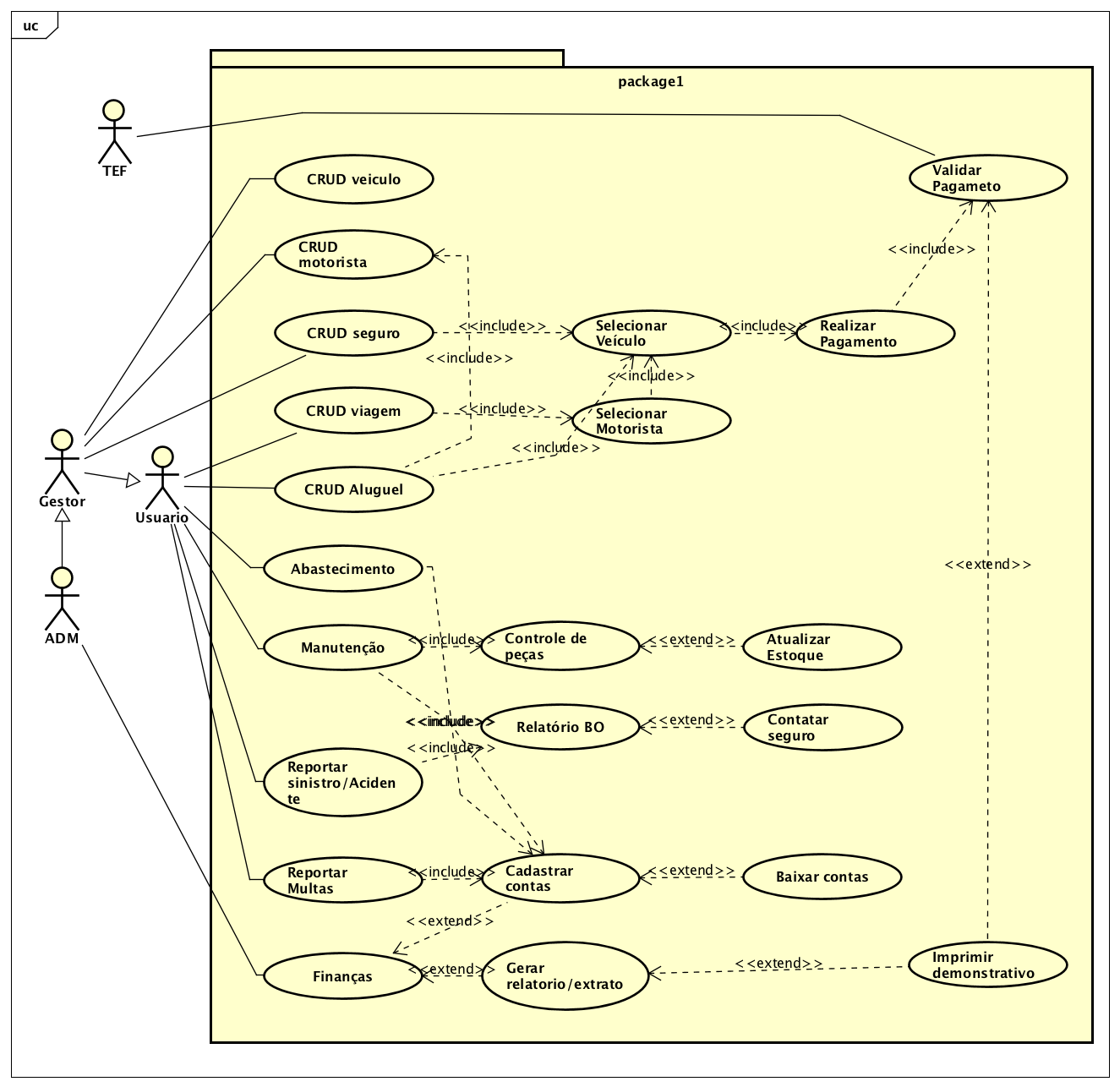
O diagrama de caso de uso descreve as principais funcionalidades do sistema e como os usuários interagem com o mesmo sistema, nele temos uma visão geral do que o sistema faz do ponto de vista do usuário.

Diagrama de sequência determina a sequência de eventos que ocorrem em um processo, descreve o fluxo da operação e tempo em que a operação vai ocorrer.

As funcionalidades do diagrama de atividades são detalhar o passo a passo de uma determinada atividade e mostrar as atividades que compõe um processo do sistema, mostrando onde começa e onde encerra as atividades.

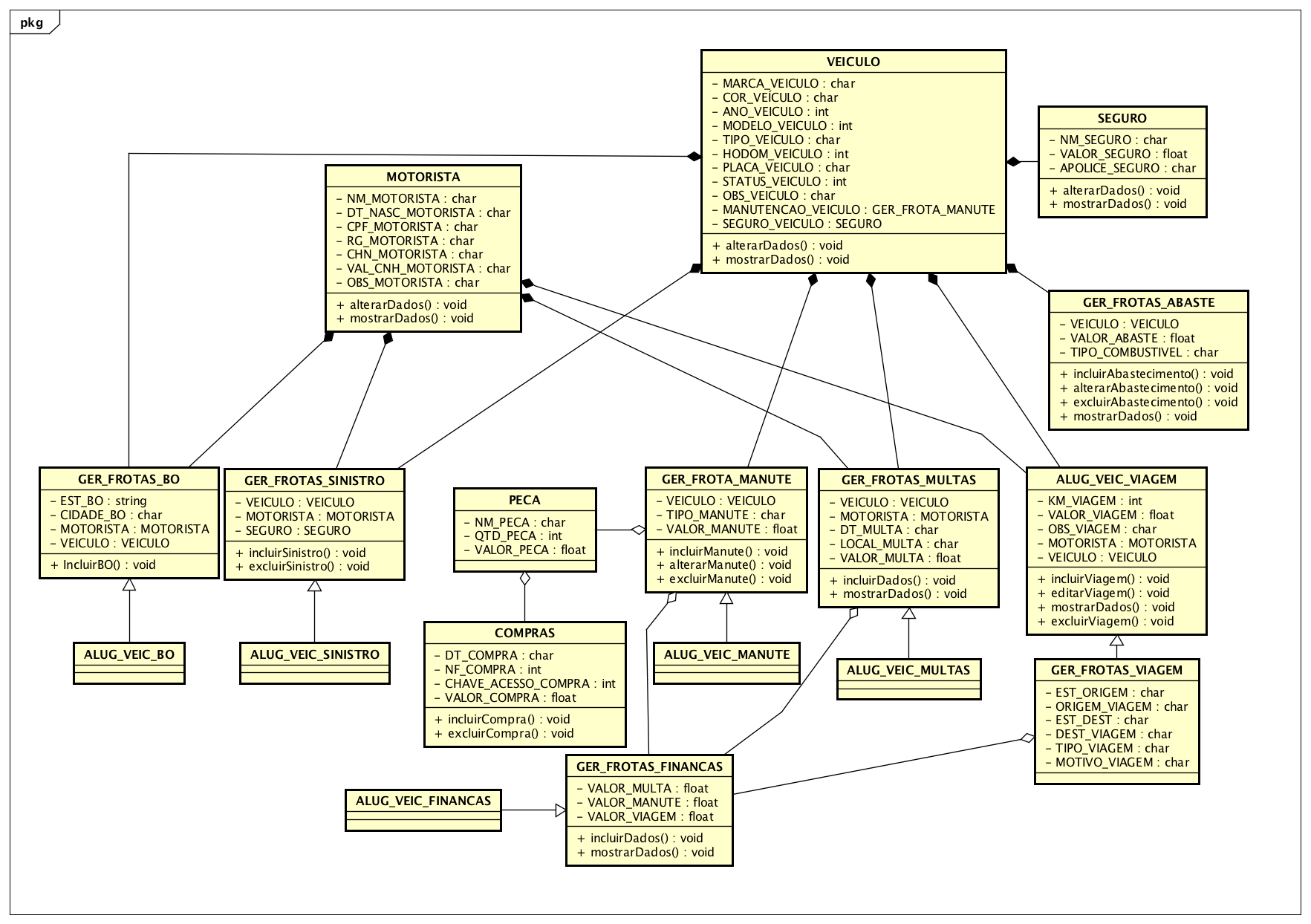
## **4.1 Processos do Sistema**

Diagrama Casos de Uso - SIGEFROTAS



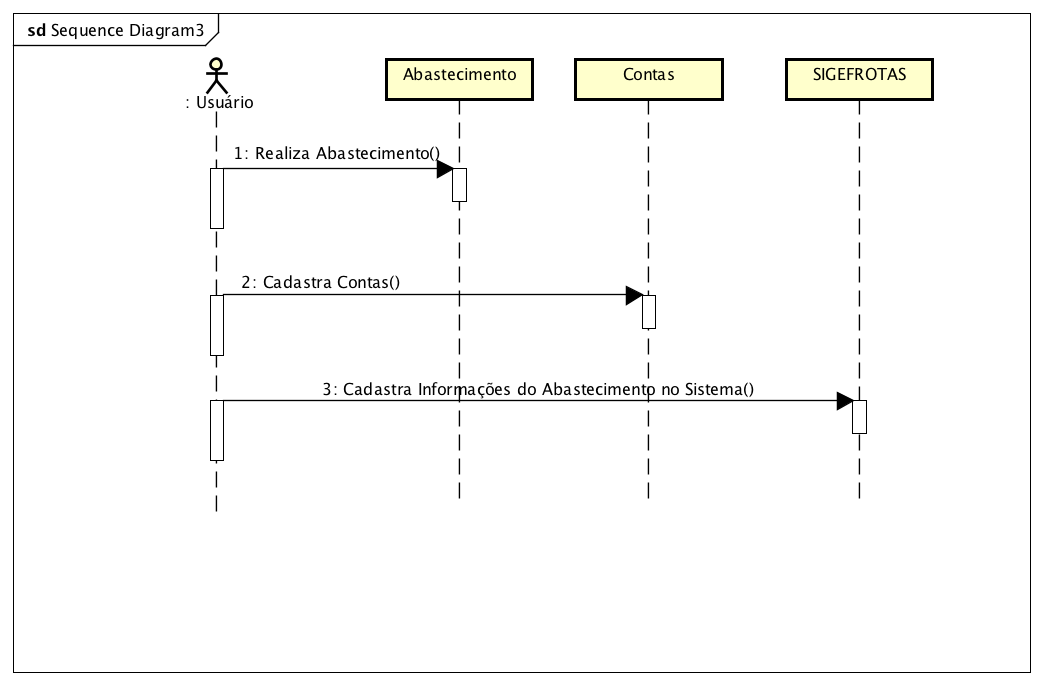
Fonte: Autor (2019)

Diagrama de Classes - GERFROTAS



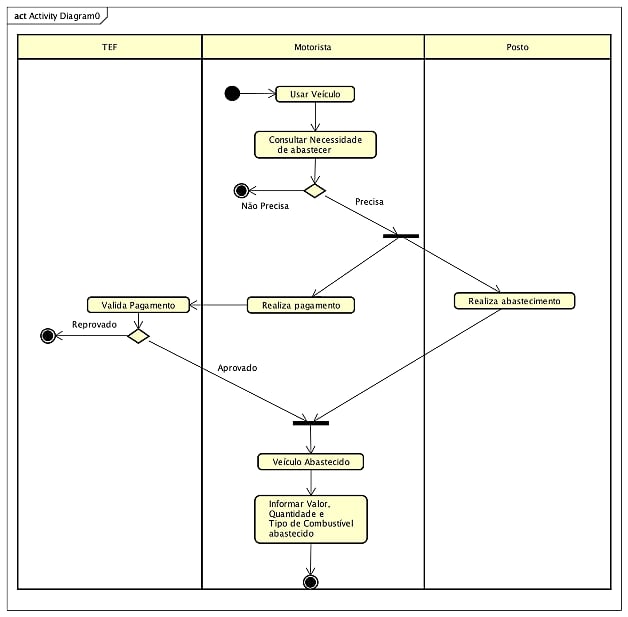
Fonte: Autor(2019)

Diagrama de Sequencia 1 - Abastecimento



Fonte: Autor(2019)

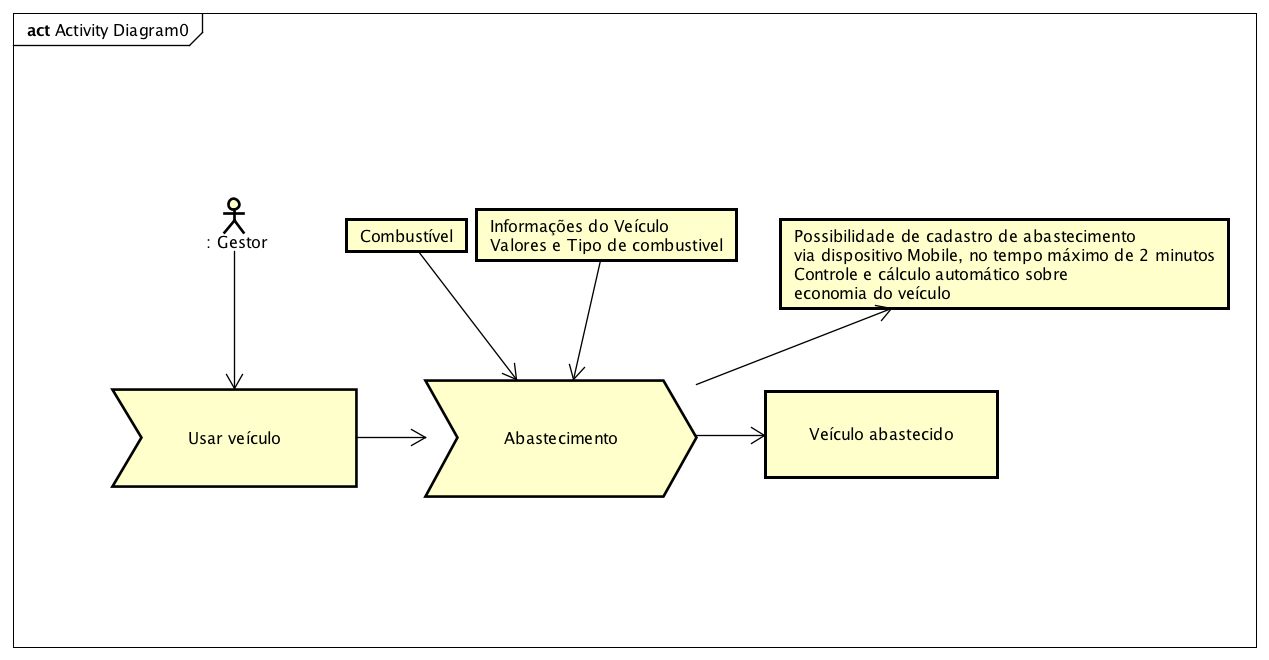
Diagrama de atividades 1 - Abastecimento



Fonte: Autor (2019)

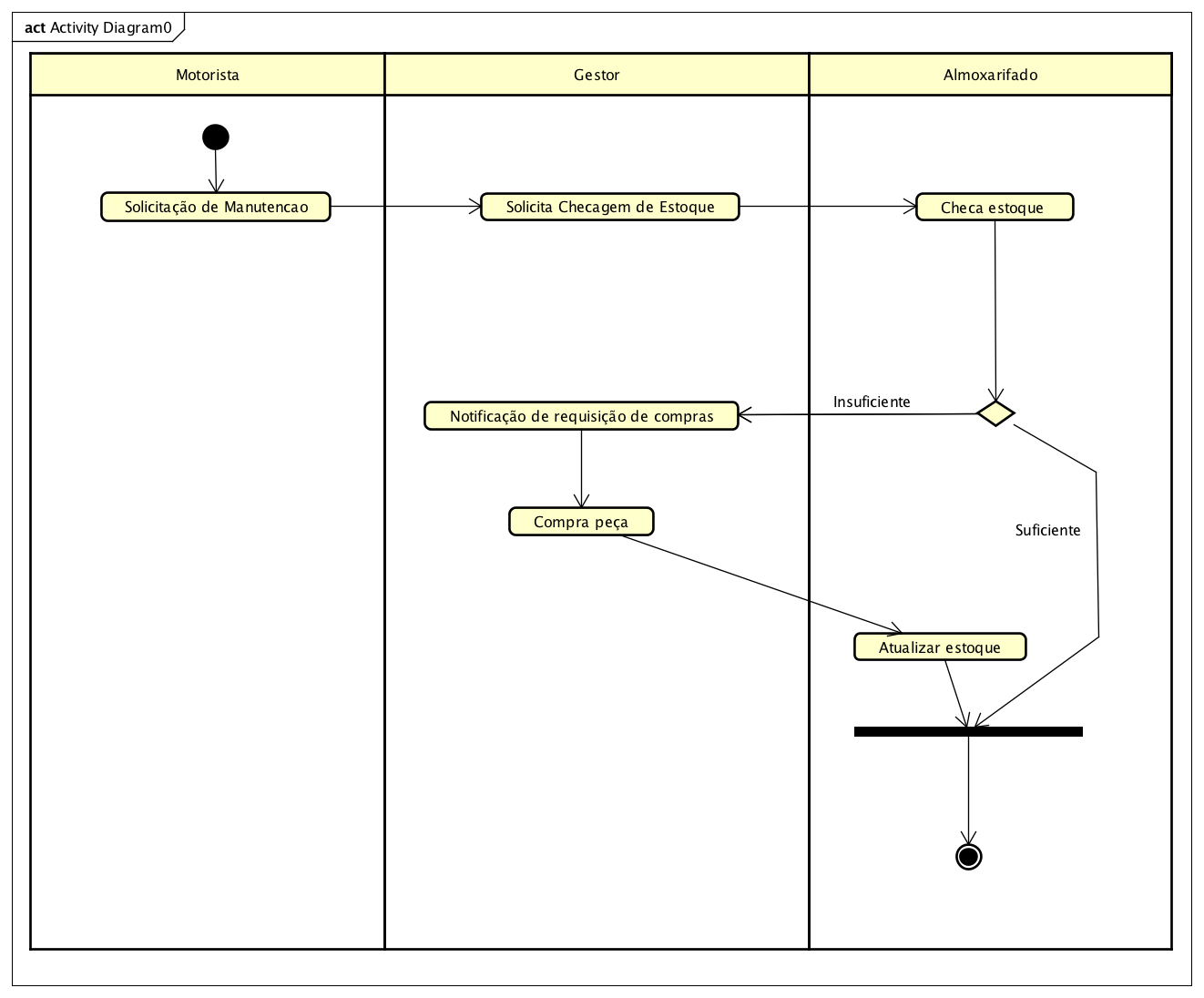
Conforme a figura: “Diagrama de atividades 1 – Abastecimento”, o processo de Abastecimento inicia com o motorista que utiliza o veículo, consulta a necessidade de abastecimento, caso não precise o processo finaliza, caso precisa abastecer, ao mesmo tempo que o posto irá realizar o abastecimento, o motorista irá realizar o pagamento, o TEF irá validar o pagamento, caso reprovado o processo finaliza, caso aprovado o veículo está abastecido e o motorista informará o valor, quantidade e tipo de combustível abastecido, com isso, o processo é finalizado por completo.

Diagrama de Processos 1 - Abastecimento



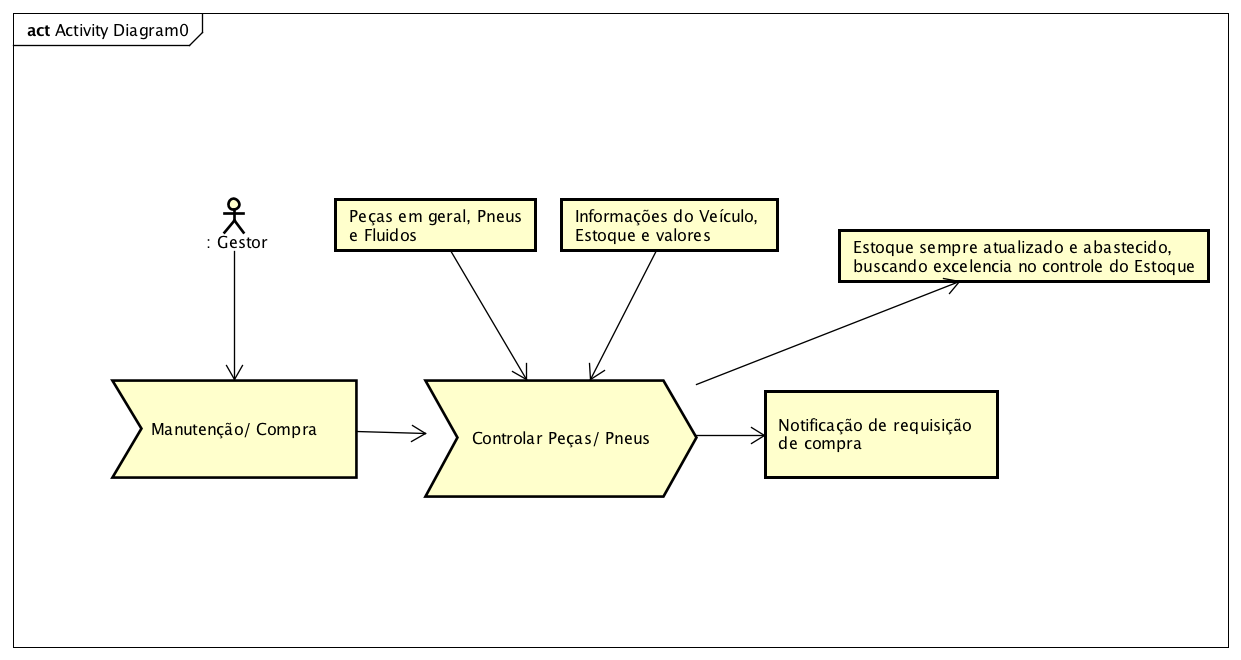
Fonte: Autor (2019)

Conforme a figura: “Diagrama de Processos 1 - Abastecimento”, o processo é iniciado com a utilização de um veículo e para que seja realizado é necessário que possua as informações sobre o veículo e as informações sobre o combustível, sua saída sempre será o veículo abastecido e sua busca é a possibilidade de cadastrar o abastecimento no sistema através do celular onde será realizado um cálculo sobre a economia do veículo.

Diagrama de atividades 2 – Controle de Peças/Pneus

Fonte:Autor (2019)

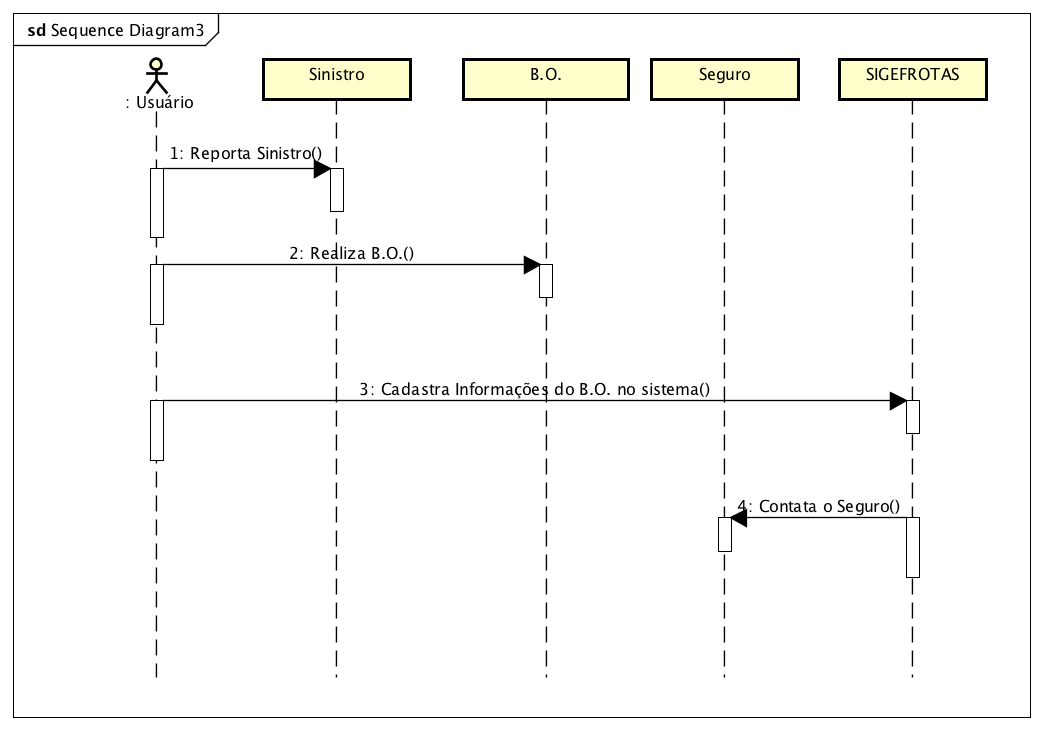
Diagrama de Processos 2 – Controle de Peças/Pneus



Fonte: Autor (2019)

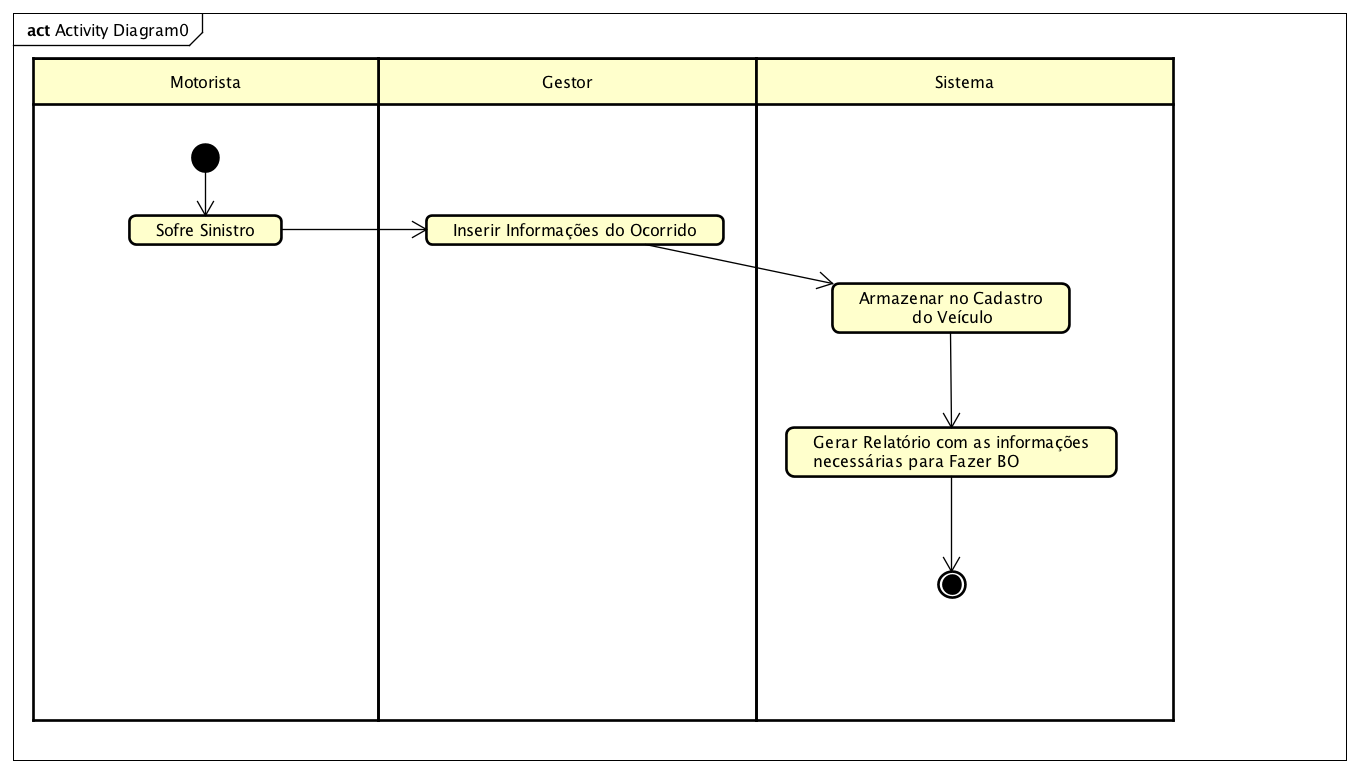
O processo se inicia com a solicitação de compra de uma peça ou a necessidade de manutenção de um veículo, para que o processo ocorra são necessárias informações referentes às peças, pneus, fluido e também as informações sobre o veículo, estoque e valores. Durante o processo sempre é buscada a atualização e abastecimento do estoque, buscando assim a excelência no controle do estoque e tendo como saída a notificação de uma nova requisição de compra.

Diagrama de Sequencia 2 – Gerar relatório BO

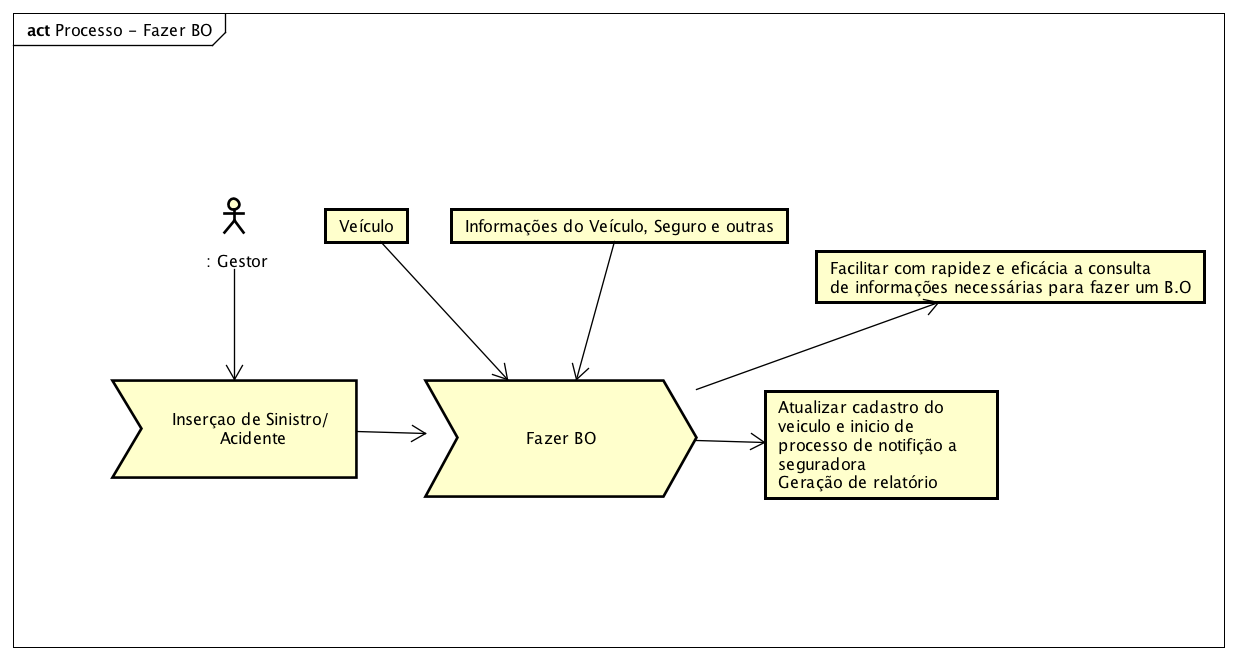


Fonte: Autor (2019)

Diagrama de atividades 3 – Gerar Relatório de Bo



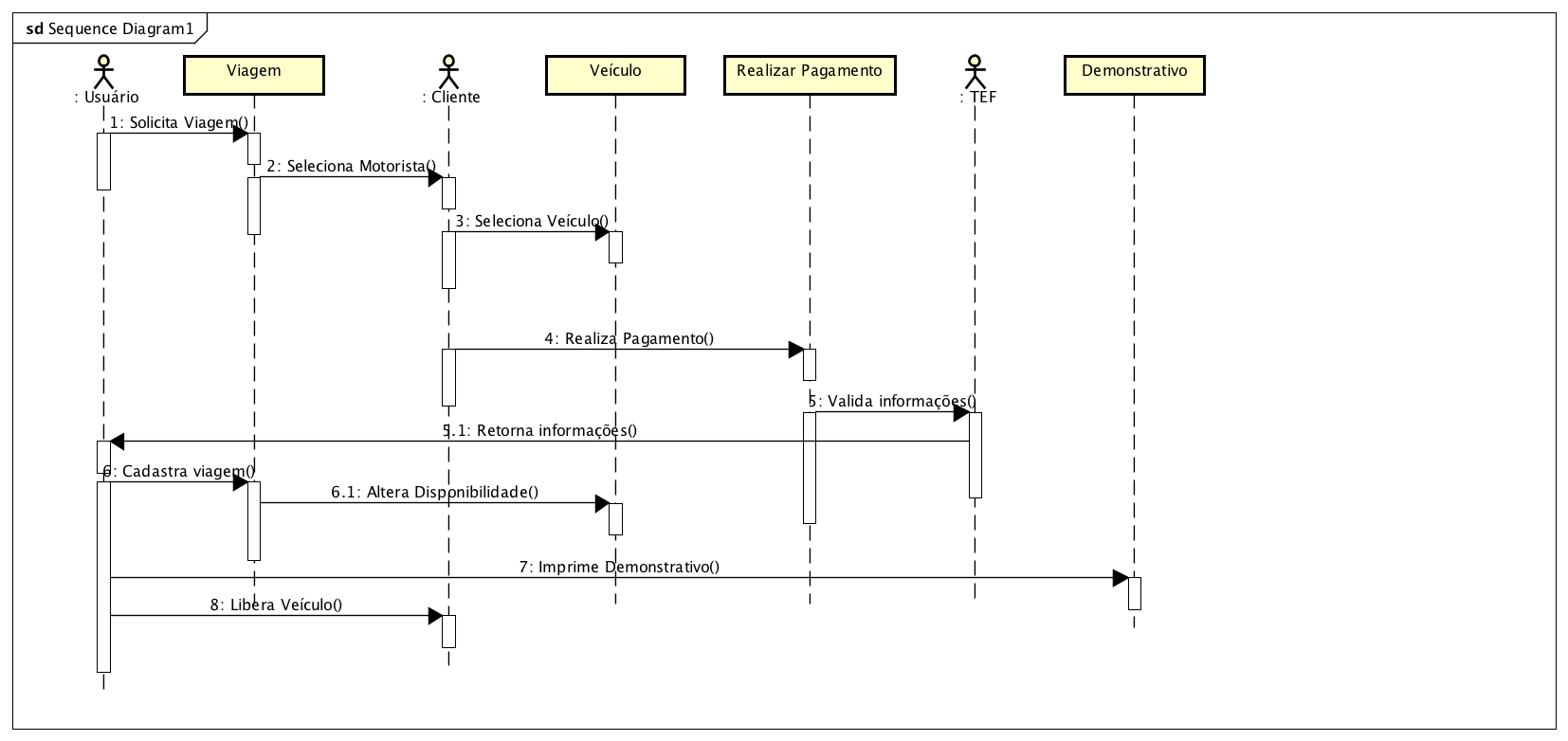
Fonte: Autor (2019)

Diagrama de Processos 3 – Gerar Relatório de Bo

Fonte: Autor (2019)

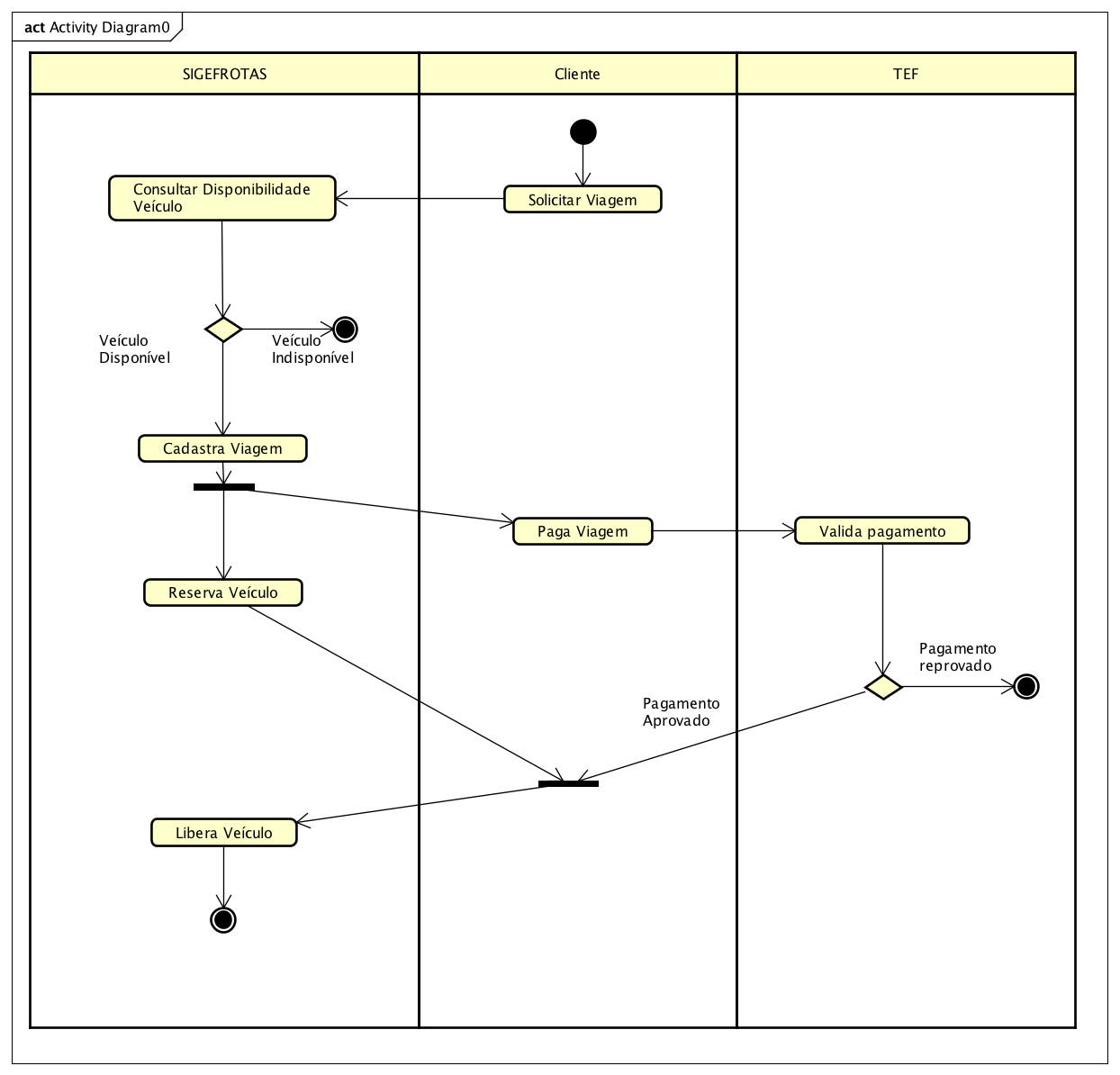
O processo se inicia com a inserção de um sinistro ou um acidente e para que seja possível a realização do B.O. são necessárias as informações do veículo, do seguro e todas as outras informações que possam ser utilizadas, feito o BO as informações do cadastro do veículo são atualizadas e é iniciado o processo de notificação da seguradora. A busca no processo é sempre facilitar e aumentar a velocidade e eficácia da consulta e realização do BO.

Diagrama de Sequencia 3 – Gerencia de Viagem

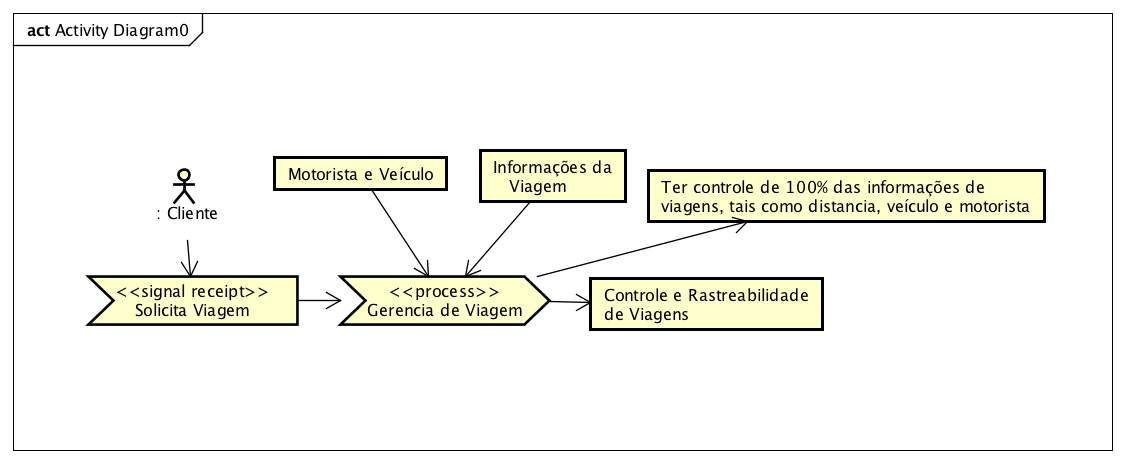


Fonte: Autor (2019)

Diagrama de atividades 4 – Gerencia de viagem

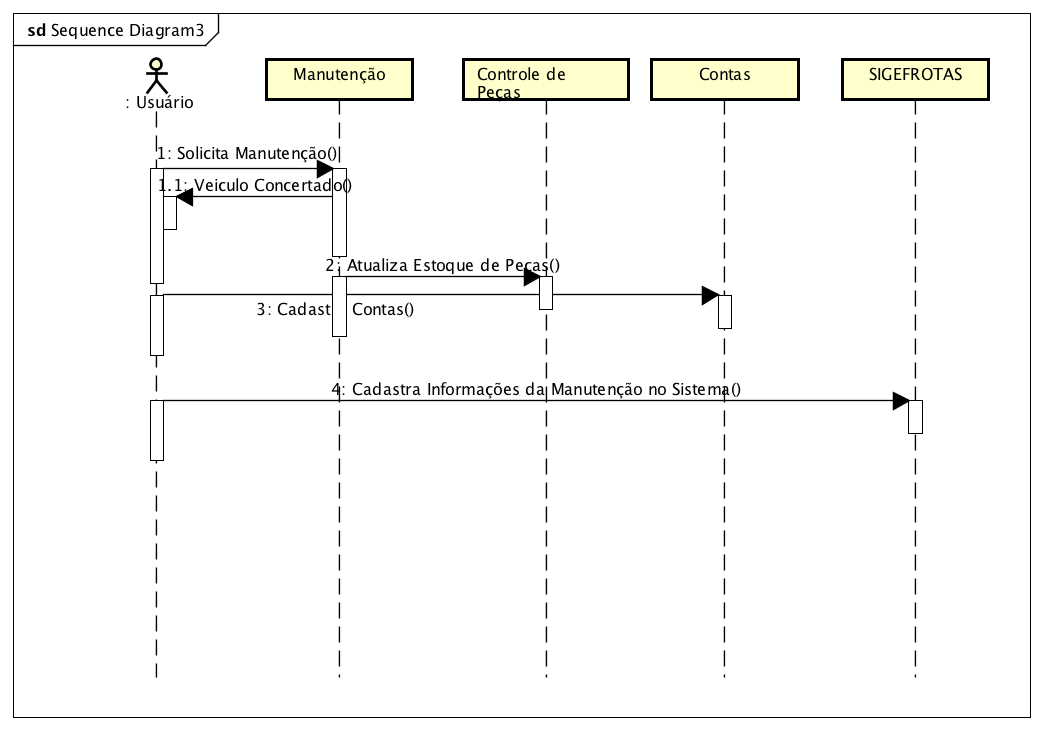


Fonte: Autor (2019)

Diagrama de processos 4 – Gerencia de viagem Fonte: Autor (2019)

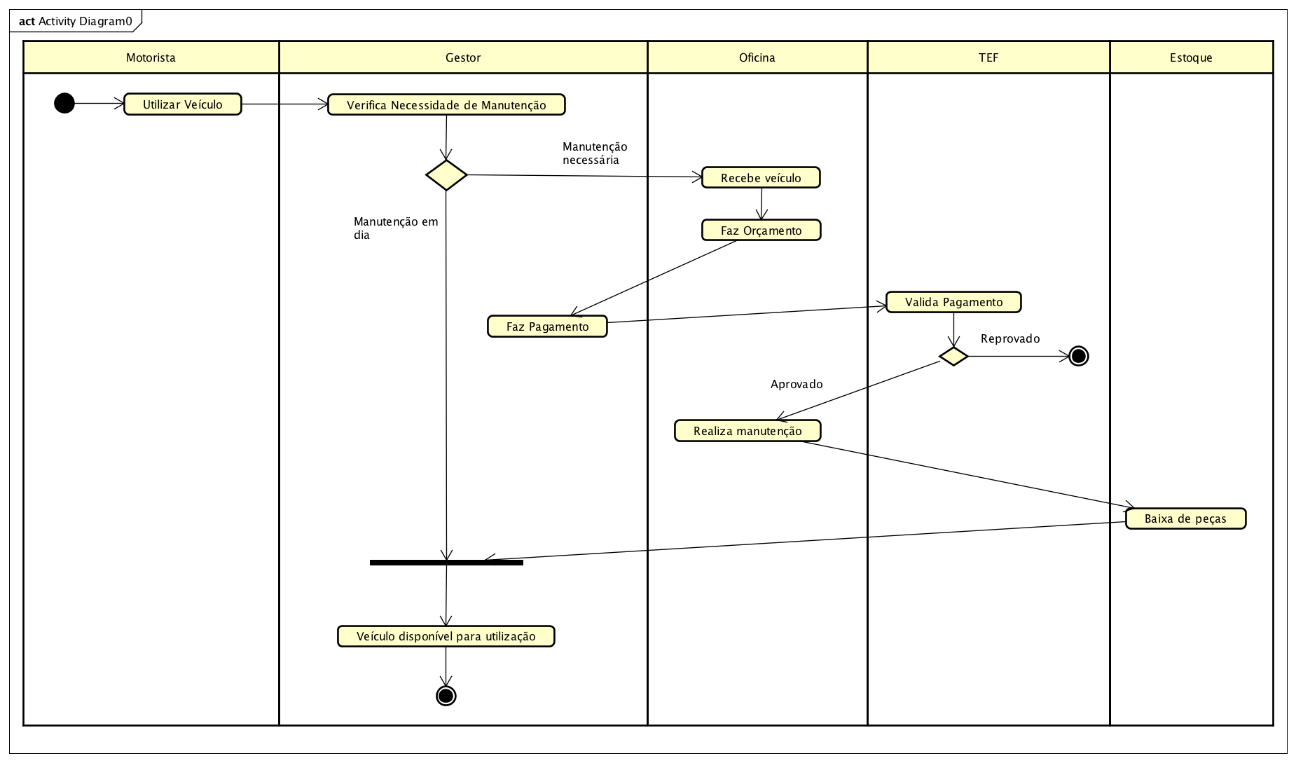
O processo se inicia com o cliente solicitando viagem, devendo ser informados os dados da viagem, como motorista, veículo, local de partida e de destino, distância e o tipo de viagem a ser realizada, como chamado técnico, manutenções internas, etc. Assim que a solicitação de viagem é realizada, é emitido um recibo da viagem com os dados para controle e o sistema e banco de dados são alimentados, gerando dados e relatórios. A meta com a gerência das viagens é aumentar a eficiência, acurácia e rastreabilidade das informações de viagem, aumentando o potencial de crescimento da empresa gerenciada com relatórios detalhados dos serviços realizados.

Diagrama de Sequencia 4 – Manutenção



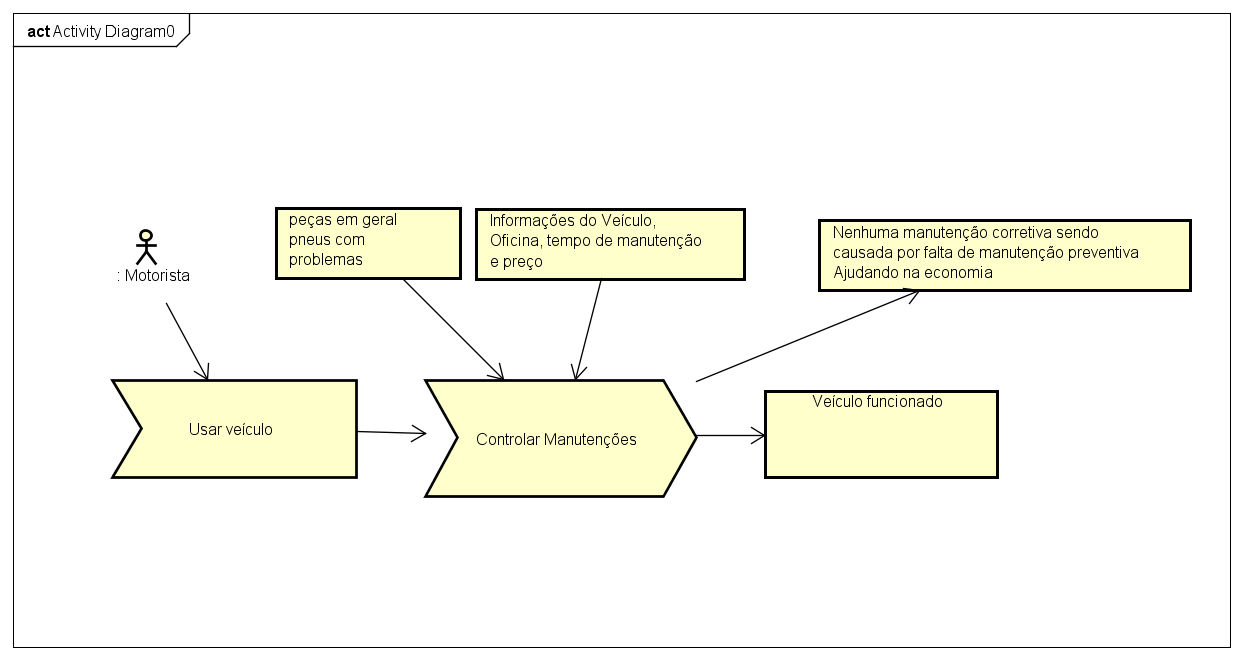
Fonte: Autor (2019)

Diagrama de atividades 5 – Controle de Manutenção



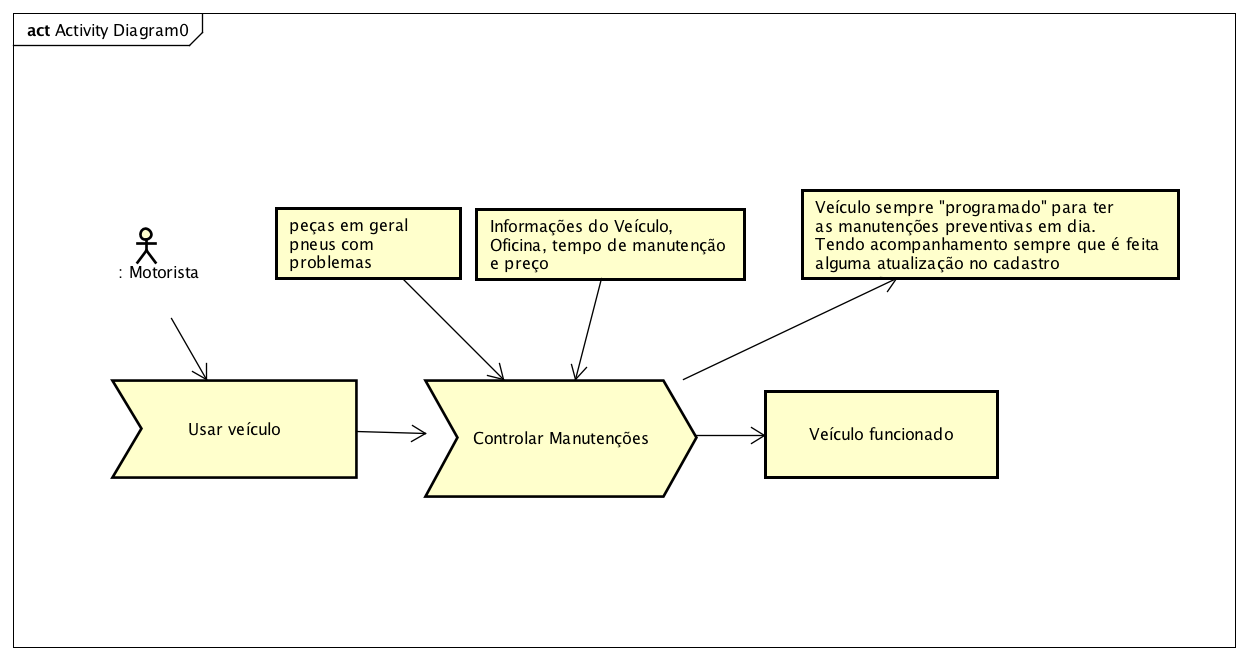
Fonte: Autor (2019)

Diagrama de Processos 5 – Controle de Manutenção Corretiva



Fonte: Autor (2019)

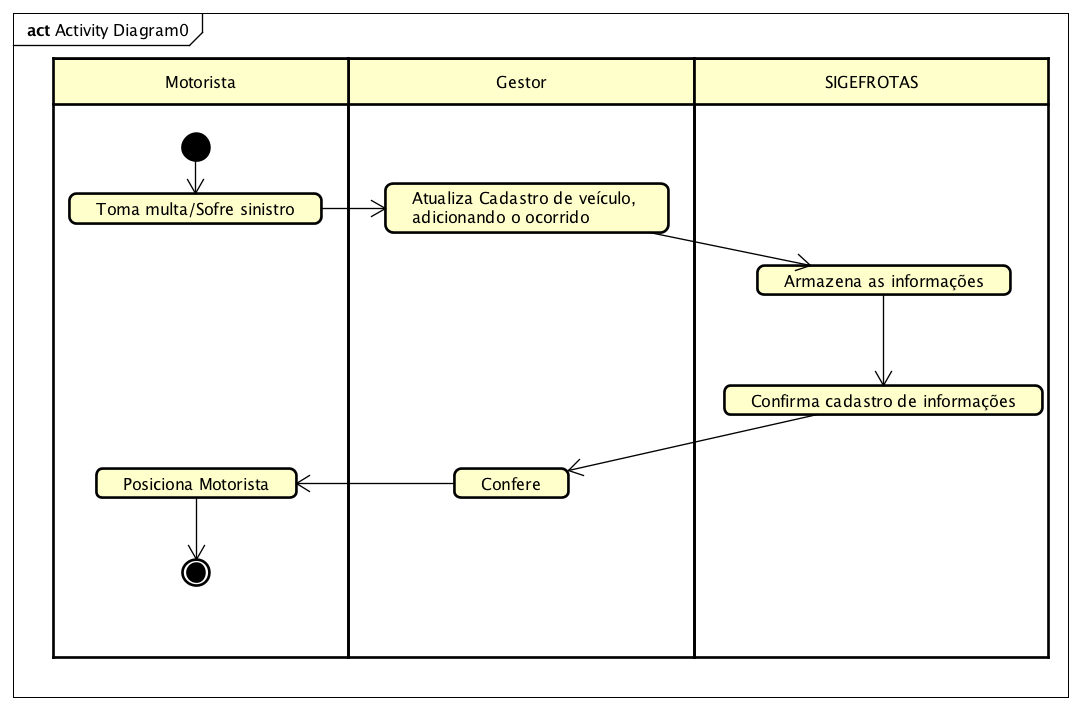
Diagrama de Processos 7 – Controle de Manutenção Preventiva



Fonte: Autor (2019)

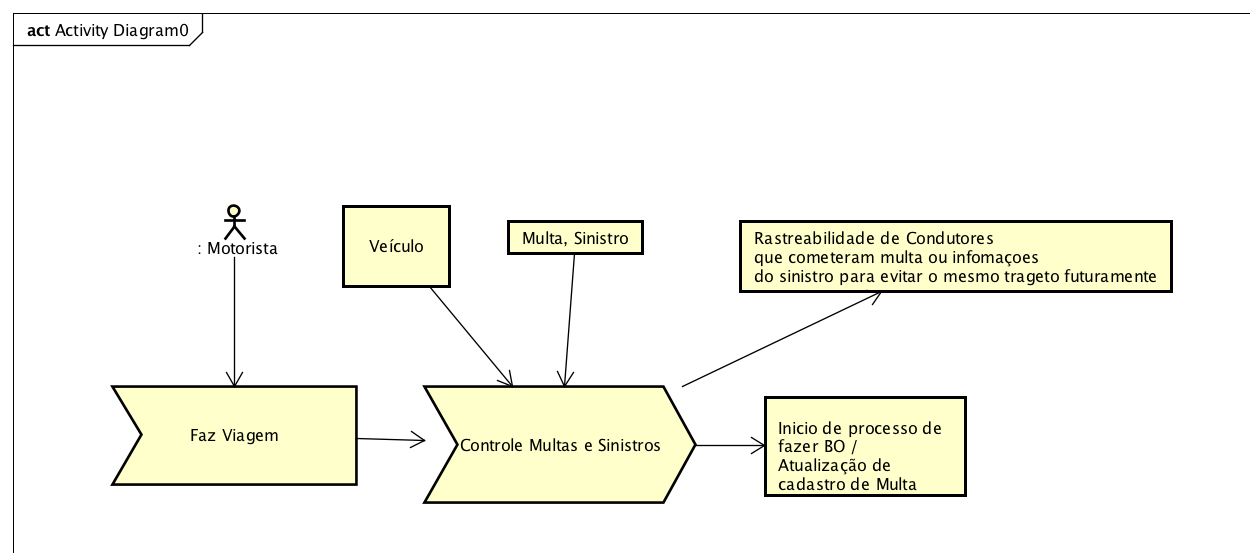
Os diagramas de Manutenção corretiva e Manutenção preventiva iniciam da mesma forma, com o ator “motorista” utilizando o veículo. A manutenção preventiva é essencial para que a corretiva não seja necessária, apenas em casos excepcionais e de emergência. Para ambos, é necessário inserir informações referentes ao veículo, como quilometragem, a oficina a ser utilizada, o tempo previsto de manutenção, as peças necessárias para serem substituídas e os custos gerados. A meta prevista é reduzir a utilização da manutenção corretiva em milésimos, garantindo a segurança do motorista, conservação das frotas e redução nos custos da empresa, aumentando a previsão de lucros.

Diagrama de atividades 6 – Controle de Multas e Sinistros



Fonte: Autor (2019)

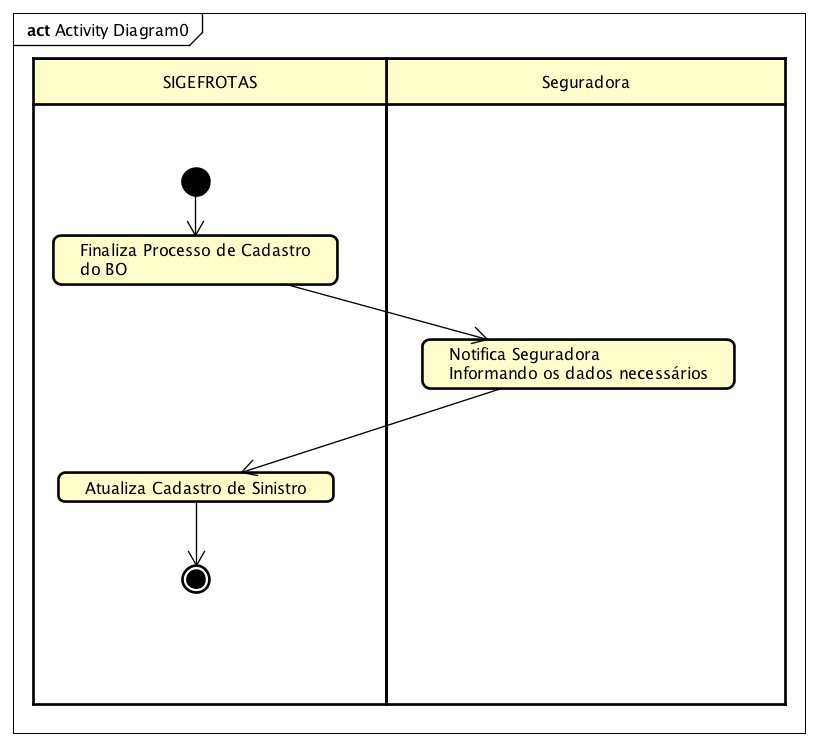
Diagrama de Processos 8 – Controle de Multas e Sinistros



Fonte: Autor (2019)

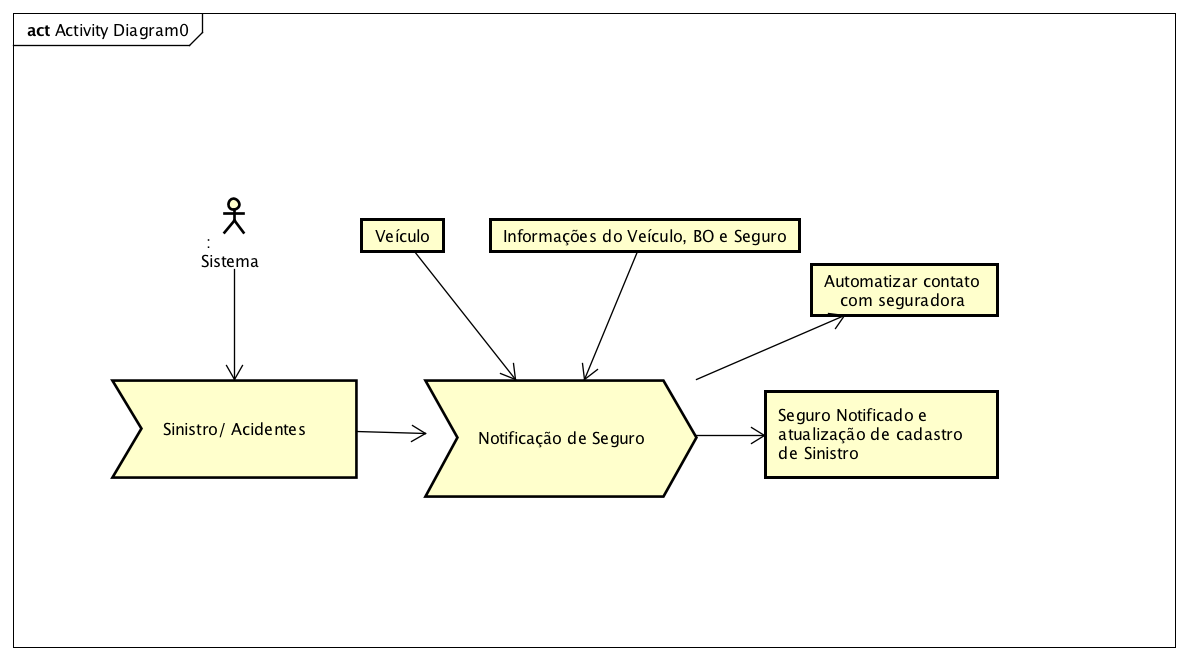
O diagrama de multas e sinistros apresenta a solução para facilitação dos serviços de controle de veículos, aumentando a rastreabilidade sobre as ações dos motoristas com relação a multas e auxílio em gerenciar os sinistros gerados por acidentes ou possivel furto/roubo. O sistema é iniciado ao realizar uma viagem, onde caso haja alguma ocorrência, o motorista deverá relatar em sistema qual o tipo de situação. A saída final é o início do procedimento de realização do Boletim de Ocorrência e a atualização do cadastro das multas e sinistros, gerando modificações financeiras e de relatórios gerenciais.

Diagrama de atividades 7 – Notificação de Seguro



Fonte: Autor (2019)

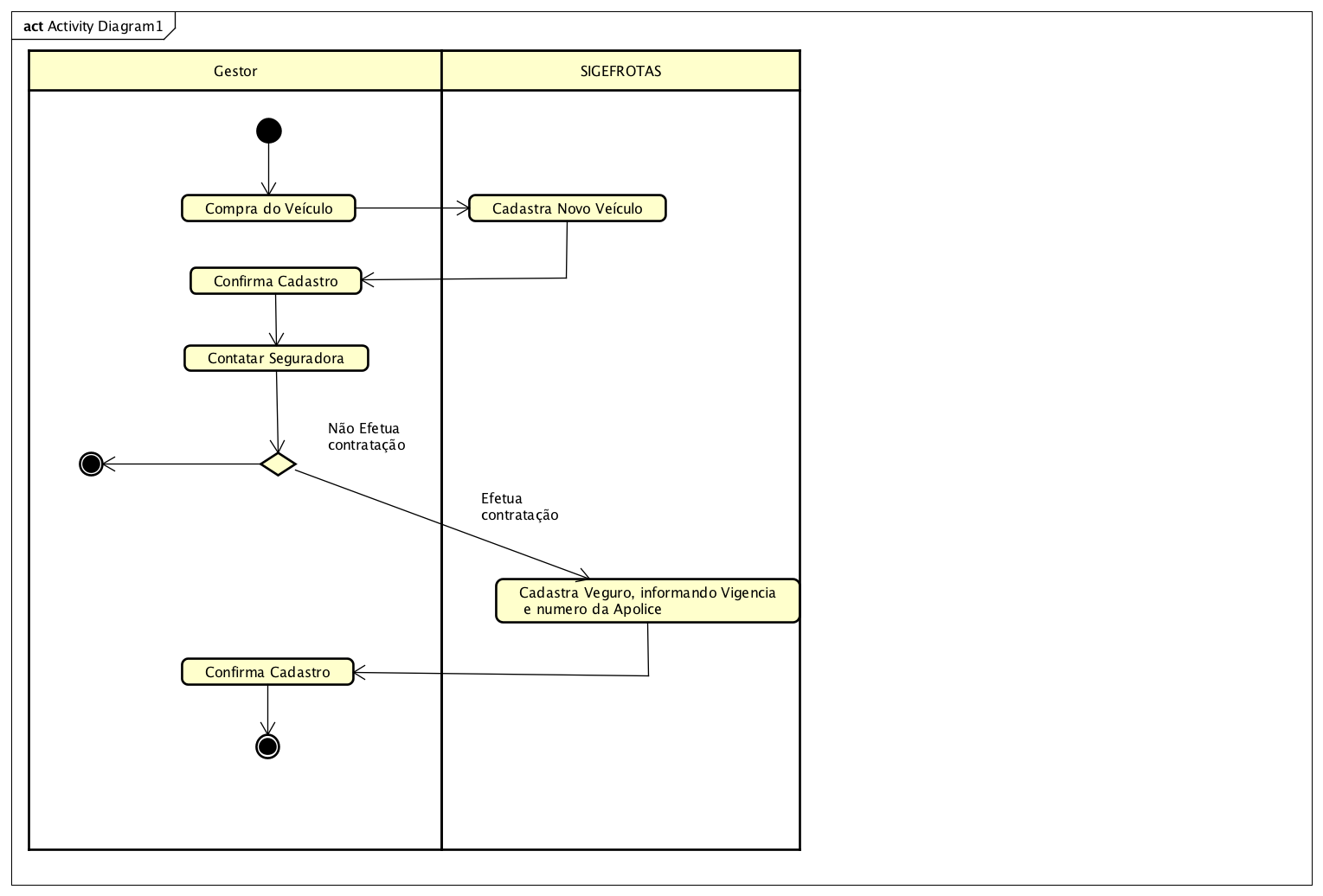
Diagrama de Processos 9 – Notificação de Seguro



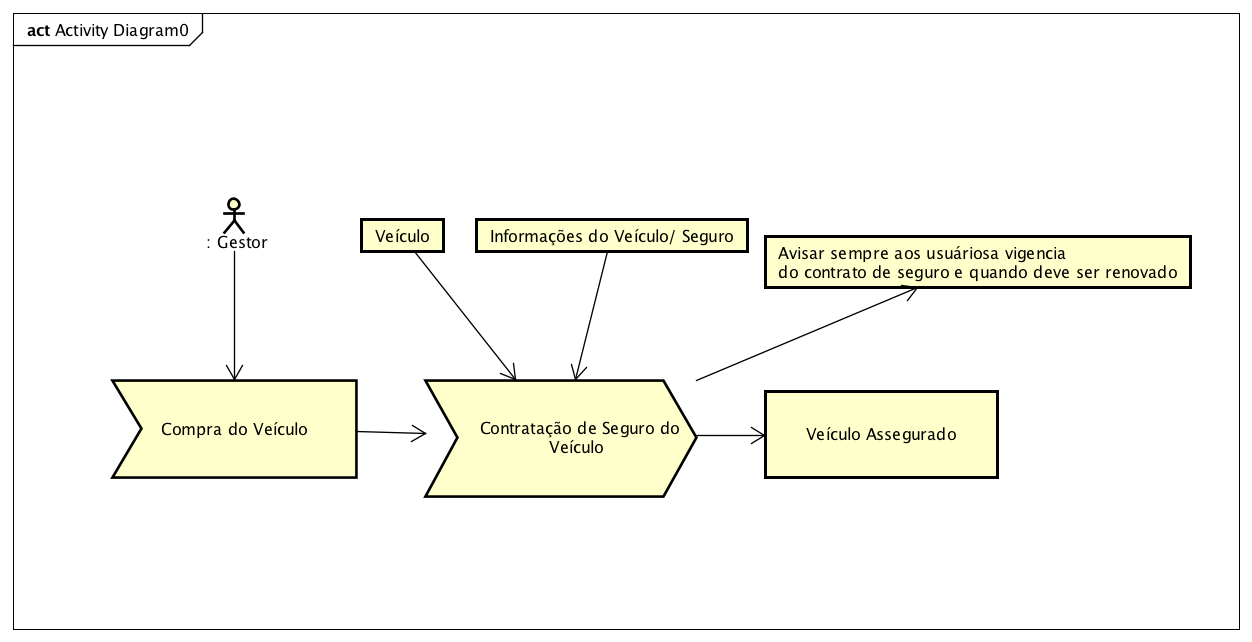
Fonte: Autor (2019)

O diagrama acima representa a notificação de sinistros e acidentes, ativados pela saída do diagrama anterior. Esta atividade se baseia em realizar a notificação à seguradora, adicionando o boletim de ocorrência gerado, as informações do veículo e qual a seguradora cadastrada na apólice. A saída dessa atividade consiste em automatizar a notificação para a seguradora, atualização dos cadastros de veículos, motorista e sinistro, e gerar relatórios administrativos. Nossa meta é automatizar 100% o sistema de gerenciamento de sinistros, retirando a necessidade de burocracias com relação a solicitações às seguradoras.

Diagrama de atividades 8 –Contratação de Seguro



Fonte: Autor (2019)

Diagrama de Processos 10 –Contratação de Seguro

Fonte: Autor (2019)

Este diagrama representa a contratação do seguro dos veículos, sendo ativado o trigger assim que um veículo for adquirido, sendo requisitado a contratação do seguro. As informações necessárias para a contratação são todos os dados do veículo, como Renavam, placa, laudos e documentos. A saída do diagrama é o veículo com o seguro ativo e número de apólice cadastrado no sistema, pronto para ser conduzido por um motorista cadastrado. A meta é sempre manter o gestor notificado quanto a vigência do contrato, reduzindo os riscos e aumentando as chances de renovação antecipada com descontos, gerando aumento do caixa da empresa.

# Banco de Dados

Para a criação do Banco de Dados, utilizamos a seguinte Query:

USE MASTER;

CREATE DATABASE GER\_FROTAS;

GO

USE GER\_FROTAS;

GO

CREATE TABLE MOTORISTA

( CD\_MOTORISTA INT NOT NULL,

NM\_MOTORISTA CHAR (80),

DT\_NASC\_MOTORISTA DATETIME,

CPF\_MOTORISTA CHAR(11),

RG\_MOTORISTA CHAR(10),

CNH\_MOTORISTA CHAR (12) NOT NULL,

VAL\_CNH\_MOTORISTA DATETIME NOT NULL,

OBS\_MOTORISTA varchar(120),

PRIMARY KEY (CD\_MOTORISTA))

GO

CREATE TABLE VEICULO

( CD\_VEICULO INT NOT NULL,

MARCA\_VEICULO varchar (15),

COR\_VEICULO varchar (20),

ANO\_VEICULO int,

MODELO\_VEICULO int,

TIPO\_VEICULO varchar (20),

HODOM\_VEICULO NUMERIC(8),

PLACA\_VEICULO varchar (8),

STATUS\_VEICULO int,

MANUTENCAO\_VEICULO int,

OBS\_VEICULO varchar (120),

SEGURO\_VEICULO int,

PRIMARY KEY (CD\_VEICULO))

GO

CREATE TABLE SEGURO

( CD\_SEGURO INT NOT NULL,

CD\_VEICULO\_SEGURO INT NOT NULL,

VALOR\_SEGURO NUMERIC (7,2),

PRIMARY KEY (CD\_SEGURO),

CONSTRAINT VEICULO\_SEGURO FOREIGN KEY (CD\_VEICULO\_SEGURO) REFERENCES VEICULO (CD\_VEICULO))

GO

CREATE TABLE FORNECEDOR

( CD\_FORNEC int NOT NULL,

END\_FORNEC varchar (120),

OBS\_FORNEC varchar (120),

PRIMARY KEY (CD\_FORNEC))

GO

CREATE TABLE PECAS

( CD\_PECA int NOT NULL,

NOME\_PECA varchar (40),

QTD\_PECA int,

VALOR\_PECA NUMERIC (7,2),

CD\_FORNEC\_PECA INT NOT NULL,

PRIMARY KEY (CD\_PECA),

CONSTRAINT FORNEC\_PECA FOREIGN KEY (CD\_FORNEC\_PECA) REFERENCES FORNECEDOR (CD\_FORNEC))

GO

CREATE TABLE GER\_FROTA\_VIAGEM

( CD\_VIAGEM int NOT NULL,

ORIGEM\_VIAGEM varchar (150),

EST\_ORIGEM\_VIAGEM varchar (50),

DEST\_VIAGEM varchar (150),

EST\_DEST\_VIAGEM varchar (50),

KM\_VIAGEM NUMERIC (5),

TIPO\_VIAGEM varchar (50),

MOTIVO\_VIAGEM varchar (20),

VALOR\_VIAGEM NUMERIC (7,2),

OBS\_VIAGEM varchar (120),

CD\_MOTORISTA\_VIAGEM INT NOT NULL,

CD\_VEICULO\_VIAGEM INT NOT NULL,

PRIMARY KEY (CD\_VIAGEM),

CONSTRAINT MOTORISTA\_VIAGEM FOREIGN KEY (CD\_MOTORISTA\_VIAGEM) REFERENCES MOTORISTA (CD\_MOTORISTA),

CONSTRAINT VEICULO\_VIAGEM FOREIGN KEY (CD\_VEICULO\_VIAGEM) REFERENCES VEICULO (CD\_VEICULO))

GO

CREATE TABLE ALUG\_VEIC\_VIAGEM

( CD\_VIAGEM int NOT NULL,

KM\_VIAGEM NUMERIC (5),

MOTIVO\_VIAGEM varchar (20),

VALOR\_VIAGEM NUMERIC (7,2),

OBS\_VIAGEM varchar (120),

CD\_MOTORISTA\_VIAGEM INT NOT NULL,

CD\_VEICULO\_VIAGEM INT NOT NULL,

PRIMARY KEY (CD\_VIAGEM),

CONSTRAINT MOTORISTA\_VIAGEM\_ALUG FOREIGN KEY (CD\_MOTORISTA\_VIAGEM) REFERENCES MOTORISTA (CD\_MOTORISTA),

CONSTRAINT VEICULO\_VIAGEM\_ALUG FOREIGN KEY (CD\_VEICULO\_VIAGEM) REFERENCES VEICULO (CD\_VEICULO))

GO

CREATE TABLE GER\_FROTA\_ABASTE

( CD\_ABASTE INT NOT NULL,

CD\_VEICULO\_ABASTE INT NOT NULL,

VALOR\_ABASTE NUMERIC (7,2),

TIPO\_COMBUSTIVEL varchar (40),

PRIMARY KEY (CD\_ABASTE),

CONSTRAINT VEICULO\_ABASTE FOREIGN KEY (CD\_VEICULO\_ABASTE) REFERENCES VEICULO (CD\_VEICULO))

GO

CREATE TABLE GER\_FROTA\_BO

( CD\_BO INT NOT NULL,

CD\_VEICULO\_BO INT NOT NULL,

CD\_MOTORISTA\_BO INT NOT NULL,

EST\_BO varchar (50),

CIDADE\_BO varchar (50),

PRIMARY KEY (CD\_BO),

CONSTRAINT VEICULO\_BO FOREIGN KEY (CD\_VEICULO\_BO) REFERENCES VEICULO (CD\_VEICULO),

CONSTRAINT MOTORISTA\_BO FOREIGN KEY (CD\_MOTORISTA\_BO) REFERENCES MOTORISTA (CD\_MOTORISTA))

GO

CREATE TABLE ALUG\_VEIC\_BO

( CD\_BO INT NOT NULL,

CD\_VEICULO\_BO INT NOT NULL,

CD\_MOTORISTA\_BO INT NOT NULL,

EST\_BO varchar (50),

CIDADE\_BO varchar (50),

PRIMARY KEY (CD\_BO),

CONSTRAINT VEICULO\_BO\_ALUG FOREIGN KEY (CD\_VEICULO\_BO) REFERENCES VEICULO (CD\_VEICULO),

CONSTRAINT MOTORISTA\_BO\_ALUG FOREIGN KEY (CD\_MOTORISTA\_BO) REFERENCES MOTORISTA (CD\_MOTORISTA))

GO

CREATE TABLE GER\_FROTA\_MANUTE

( CD\_MANUTE INT NOT NULL,

CD\_VEICULO\_MANUTE INT NOT NULL,

TP\_MANUTE varchar (30),

VALOR\_MANUTE NUMERIC (7,2),

PRIMARY KEY (CD\_MANUTE),

CONSTRAINT VEICULO\_MANUTE FOREIGN KEY (CD\_VEICULO\_MANUTE) REFERENCES VEICULO (CD\_VEICULO))

GO

CREATE TABLE ALUG\_VEIC\_MANUTE

( CD\_MANUTE INT NOT NULL,

CD\_VEICULO\_MANUTE INT NOT NULL,

TP\_MANUTE varchar (30),

VALOR\_MANUTE NUMERIC (7,2),

PRIMARY KEY (CD\_MANUTE),

CONSTRAINT VEICULO\_MANUTE\_ALUG FOREIGN KEY (CD\_VEICULO\_MANUTE) REFERENCES VEICULO (CD\_VEICULO))

GO

CREATE TABLE GER\_FROTA\_MULTAS

( CD\_MULTA INT NOT NULL,

CD\_VEICULO\_MULTA INT NOT NULL,

CD\_MOTORISTA\_MULTA INT NOT NULL,

DT\_MULTA datetime,

LOCAL\_MULTA varchar (120),

VALOR\_MULTA NUMERIC (7,2),

PRIMARY KEY (CD\_MULTA),

CONSTRAINT VEICULO\_MULTA FOREIGN KEY (CD\_VEICULO\_MULTA) REFERENCES VEICULO (CD\_VEICULO),

CONSTRAINT MOTORISTA\_MULTA FOREIGN KEY (CD\_MOTORISTA\_MULTA) REFERENCES MOTORISTA (CD\_MOTORISTA))

GO

CREATE TABLE ALUG\_VEIC\_MULTAS

( CD\_MULTA INT NOT NULL,

CD\_VEICULO\_MULTA INT NOT NULL,

CD\_MOTORISTA\_MULTA INT NOT NULL,

DT\_MULTA datetime,

LOCAL\_MULTA varchar (120),

VALOR\_MULTA NUMERIC (7,2),

PRIMARY KEY (CD\_MULTA),

CONSTRAINT VEICULO\_MULTA\_ALUG FOREIGN KEY (CD\_VEICULO\_MULTA) REFERENCES VEICULO (CD\_VEICULO),

CONSTRAINT MOTORISTA\_MULTA\_ALUG FOREIGN KEY (CD\_MOTORISTA\_MULTA) REFERENCES MOTORISTA (CD\_MOTORISTA))

GO

CREATE TABLE GER\_FROTA\_FINANCAS

( CD\_FINANCA INT NOT NULL,

VALOR\_MULTA NUMERIC (7,2) NOT NULL,

VALOR\_MANUTE NUMERIC (7,2) NOT NULL,

VALOR\_VIAGEM NUMERIC (7,2) NOT NULL,

CD\_SEGURO\_FINANCA INT NOT NULL,

CD\_PECA\_FINANCA INT NOT NULL,

CD\_VIAGEM\_FINANCA INT NOT NULL,

CD\_ABASTE\_FINANCA INT NOT NULL,

CD\_MANUTE\_FINANCA INT NOT NULL,

CD\_MULTA\_FINANCA INT NOT NULL,

PRIMARY KEY (CD\_FINANCA),

CONSTRAINT VLR\_SEG FOREIGN KEY (CD\_SEGURO\_FINANCA) REFERENCES SEGURO (CD\_SEGURO),

CONSTRAINT VLR\_PECA FOREIGN KEY (CD\_PECA\_FINANCA) REFERENCES PECAS (CD\_PECA),

CONSTRAINT VLR\_VIAGEM FOREIGN KEY (CD\_VIAGEM\_FINANCA) REFERENCES GER\_FROTA\_VIAGEM (CD\_VIAGEM),

CONSTRAINT VLR\_MANUT FOREIGN KEY (CD\_MANUTE\_FINANCA) REFERENCES GER\_FROTA\_MANUTE(CD\_MANUTE),

CONSTRAINT VLR\_MULTA FOREIGN KEY (CD\_MULTA\_FINANCA) REFERENCES GER\_FROTA\_MULTAS (CD\_MULTA),

CONSTRAINT VLR\_ABASTE FOREIGN KEY (CD\_ABASTE\_FINANCA) REFERENCES GER\_FROTA\_ABASTE(CD\_ABASTE))

GO

CREATE TABLE ALUG\_VEIC\_FINANCAS

( CD\_FINANCA INT NOT NULL,

VALOR\_MULTA NUMERIC (7,2) NOT NULL,

VALOR\_MANUTE NUMERIC (7,2) NOT NULL,

VALOR\_VIAGEM NUMERIC (7,2) NOT NULL,

CD\_SEGURO\_FINANCA INT NOT NULL,

CD\_PECA\_FINANCA INT NOT NULL,

CD\_VIAGEM\_FINANCA INT NOT NULL,

CD\_MANUTE\_FINANCA INT NOT NULL,

CD\_MULTA\_FINANCA INT NOT NULL,

PRIMARY KEY (CD\_FINANCA),

CONSTRAINT VLR\_SEG\_ALUG FOREIGN KEY (CD\_SEGURO\_FINANCA) REFERENCES SEGURO (CD\_SEGURO),

CONSTRAINT VLR\_PECA\_ALUG FOREIGN KEY (CD\_PECA\_FINANCA) REFERENCES PECAS (CD\_PECA),

CONSTRAINT VLR\_VIAGEM\_ALUG FOREIGN KEY (CD\_VIAGEM\_FINANCA) REFERENCES ALUG\_VEIC\_VIAGEM(CD\_VIAGEM),

CONSTRAINT VLR\_MULTA\_ALUG FOREIGN KEY (CD\_MULTA\_FINANCA) REFERENCES ALUG\_VEIC\_MULTAS (CD\_MULTA),

CONSTRAINT VLR\_MANUT\_ALUG FOREIGN KEY (CD\_MANUTE\_FINANCA) REFERENCES GER\_FROTA\_MANUTE(CD\_MANUTE))

GO

CREATE TABLE GER\_FROTA\_SINISTRO

( CD\_SINISTRO INT NOT NULL,

CD\_VEICULO\_SINISTRO INT NOT NULL,

CD\_MOTORISTA\_SINISTRO INT NOT NULL,

CD\_SEGURO\_SINISTRO INT NOT NULL,

PRIMARY KEY (CD\_SINISTRO),

CONSTRAINT VEICULO\_SINISTRO FOREIGN KEY (CD\_VEICULO\_SINISTRO) REFERENCES VEICULO (CD\_VEICULO),

CONSTRAINT MOTORISTA\_SINISTRO FOREIGN KEY (CD\_MOTORISTA\_SINISTRO) REFERENCES MOTORISTA (CD\_MOTORISTA),

CONSTRAINT SEGURO\_SINISTRO FOREIGN KEY (CD\_SEGURO\_SINISTRO) REFERENCES SEGURO (CD\_SEGURO))

GO

CREATE TABLE ALUG\_VEIC\_SINISTRO

( CD\_SINISTRO INT NOT NULL,

CD\_VEICULO\_SINISTRO INT NOT NULL,

CD\_MOTORISTA\_SINISTRO INT NOT NULL,

CD\_SEGURO\_SINISTRO INT NOT NULL,

PRIMARY KEY (CD\_SINISTRO),

CONSTRAINT VEICULO\_SINISTRO\_ALUG FOREIGN KEY (CD\_VEICULO\_SINISTRO) REFERENCES VEICULO (CD\_VEICULO),

CONSTRAINT MOTORISTA\_SINISTRO\_ALUG FOREIGN KEY (CD\_MOTORISTA\_SINISTRO) REFERENCES MOTORISTA (CD\_MOTORISTA),

CONSTRAINT SEGURO\_SINISTRO\_ALUG FOREIGN KEY (CD\_SEGURO\_SINISTRO) REFERENCES SEGURO (CD\_SEGURO))

GO

CREATE TABLE COMPRAS

( CD\_COMPRA INT NOT NULL,

DT\_COMPRA DATETIME,

NF\_COMPRA NUMERIC (30),

CHAVE\_ACESSO\_COMPRA VARCHAR (120),

VALOR\_COMPRA NUMERIC (7,2),

CD\_PECA\_COMPRA INT NOT NULL,

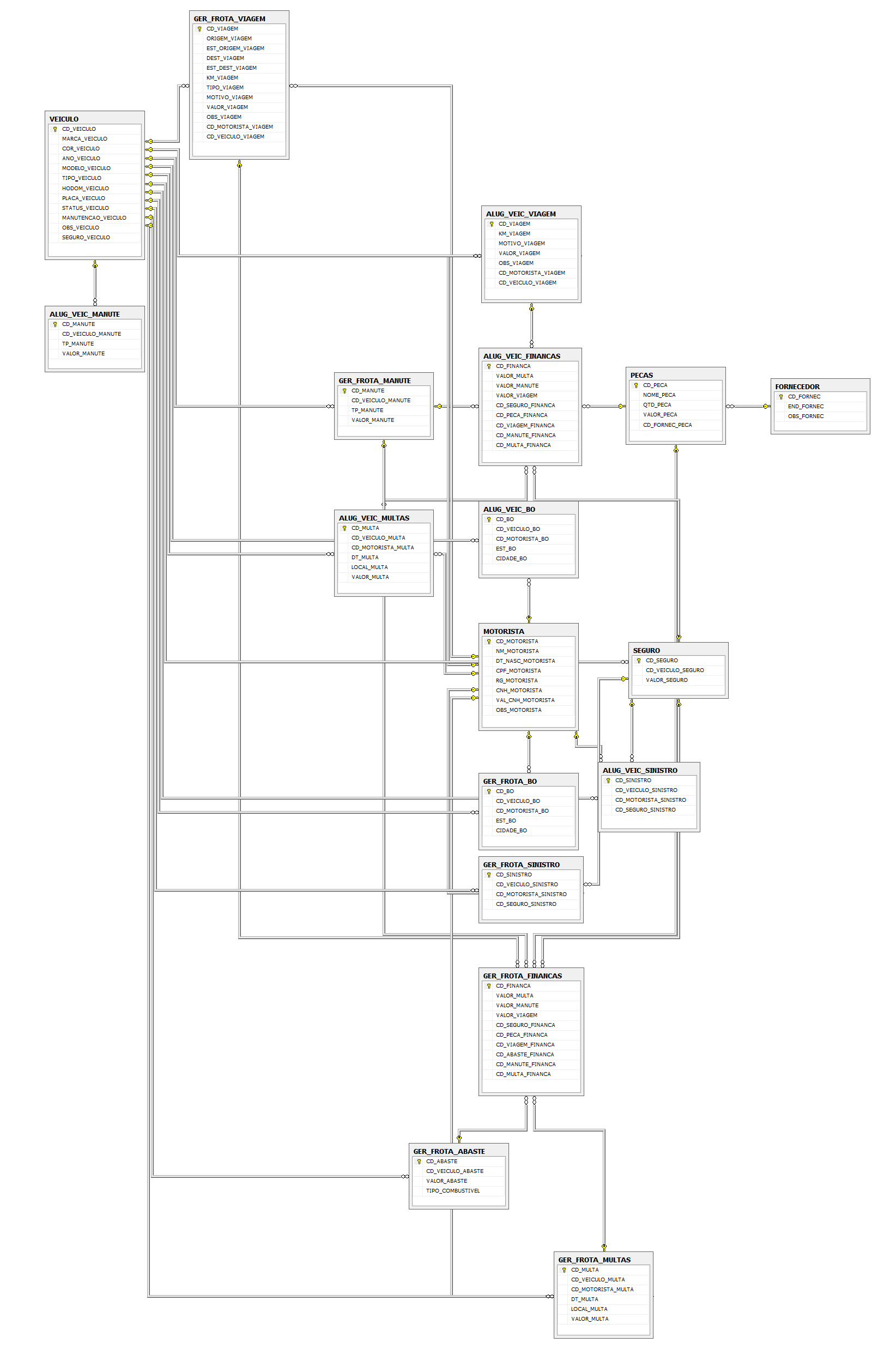
PRIMARY KEY (CD\_COMPRA),

CONSTRAINT COMPRA FOREIGN KEY (CD\_PECA\_COMPRA) REFERENCES PECAS (CD\_PECA))

GO

Com essa Query, foi criado o seguinte Diagrama Relacional:

Diagrama Relacional 1 – Banco de Dados.



Fonte: Autor(2019)

# Manuais

## **6.1 Manual de Instalação**

Os Manuais são muito importantes para os usuários de um sistema, seja qual for, ensina-los desde como instalar o seu produto até como o utilizar é uma boa forma de não só garantir que o usuário utilize de forma correta e tirar maior proveito do sistema, como também garante que haja uma menor quantidade de erros do programas, por isso foram desenvolvidos dois manuais para auxiliar o usuário a ter acesso ao seu programa, o primeiro é o de instalação, ensinando como instalar o produto em seu equipamento caso não queira utilizar em seu navegador, e o segundo manual é o de utilização do equipamento.

**Manual de Instalação do Sistema SIGEFROTAS**

Para realizar a instalar o SIGEFROTAS em seus equipamentos siga os passos a seguir:

1. Após a compra de seu Sistema, você receberá no e-mail informado durante seu cadastro uma mensagem com os links para Download do Sigefrotas.

1. No e-mail clique agora em “Download Setup”, após isso será iniciado o download de um arquivo chamado “Sigefrotas\_Setup.exe”

1. Execute o arquivo e abrirá o Instalador do programa.

1. Logo na primeira tela ele irá solicitar que selecione se deseja instalar em 64 Bits, ou em 32 Bits, marque a opção que seja compatível com a arquitetura do seu sistema operacional.
2. Depois selecione o local onde será instalado os arquivos do programa, de padrão vem a opção “C:\Program Files\Sigefrotas” para 64 Bits e “C:\Program Files (x86)\Sigefrotas” para 32 Bits, mas é possível alterar para algum outro de sua preferência, ao terminar clique em “Avançar”.

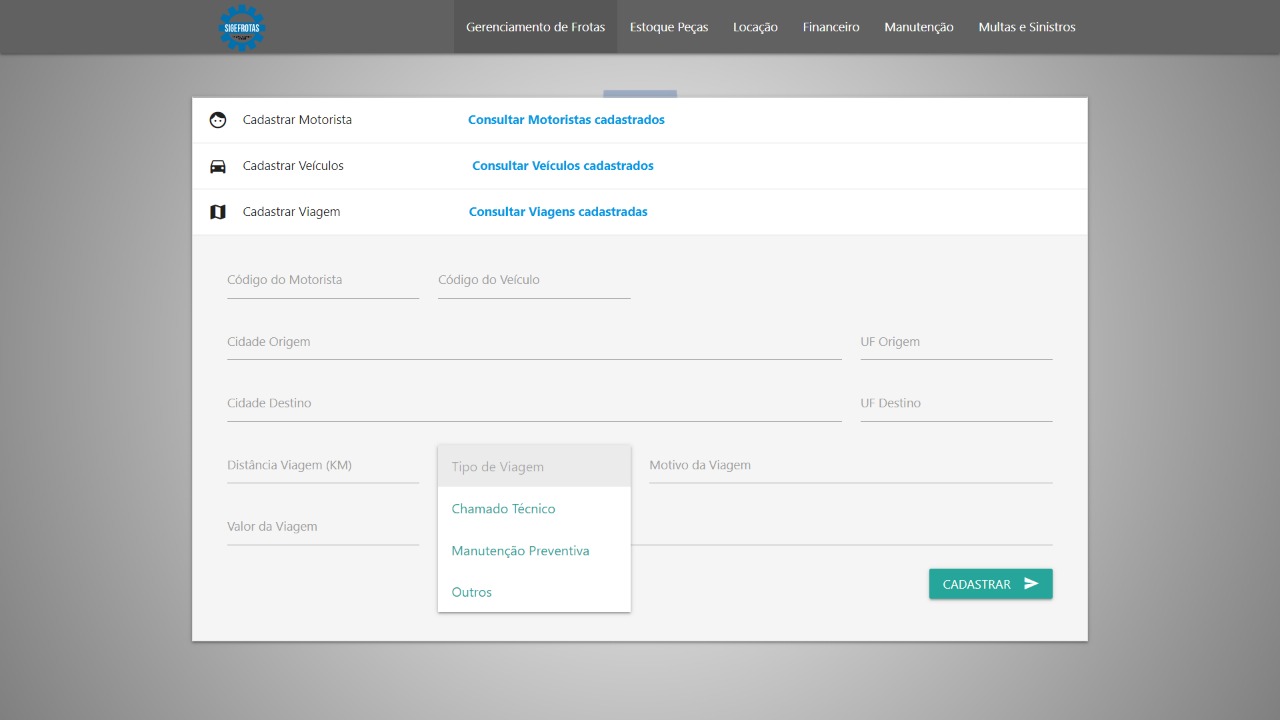
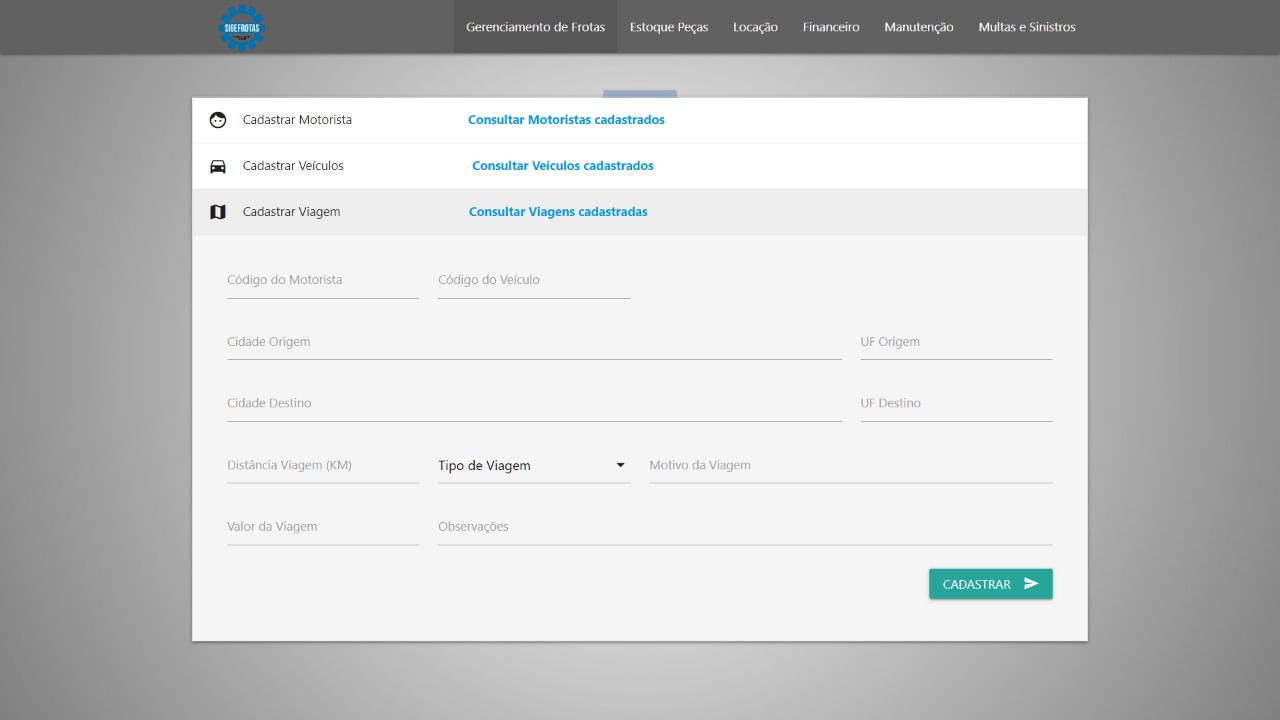
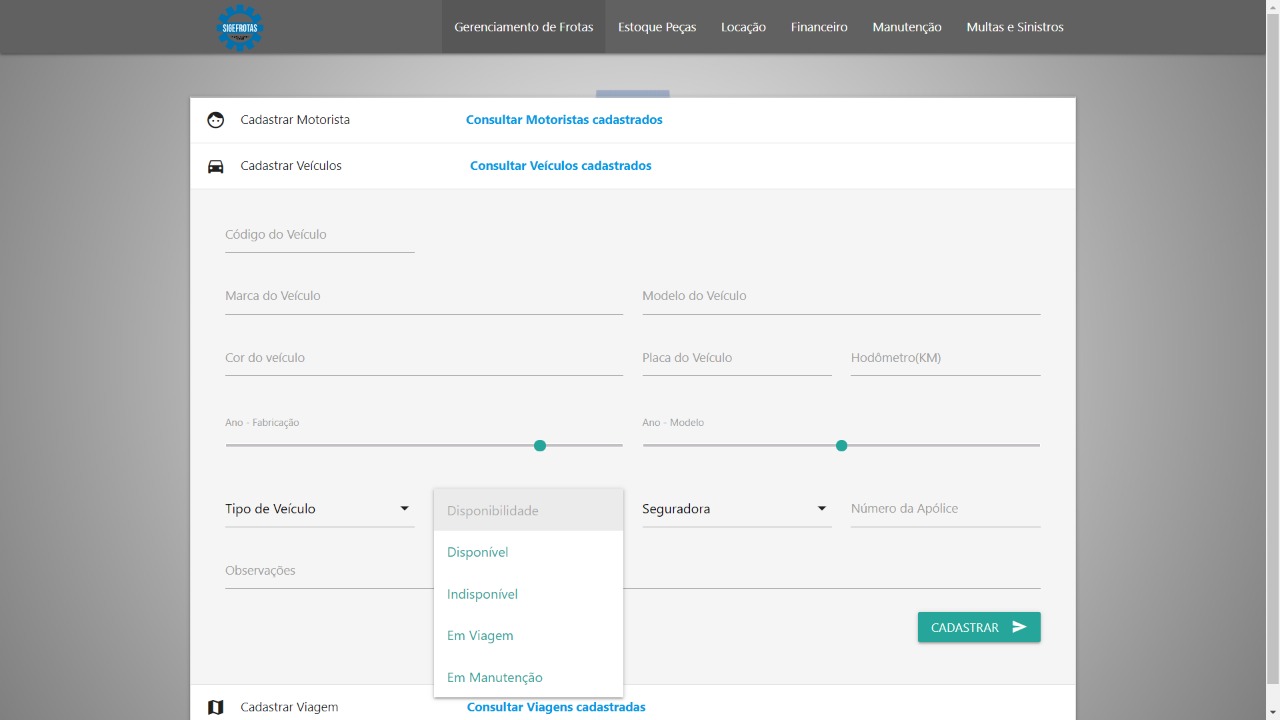
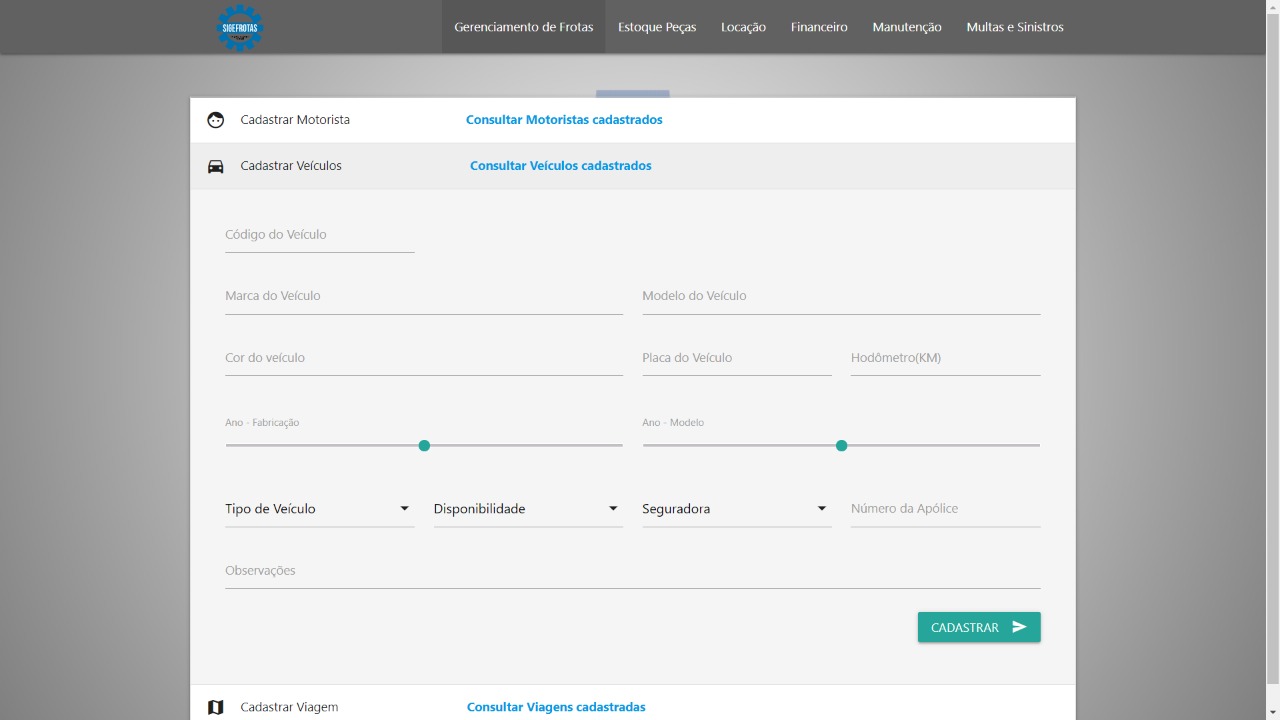
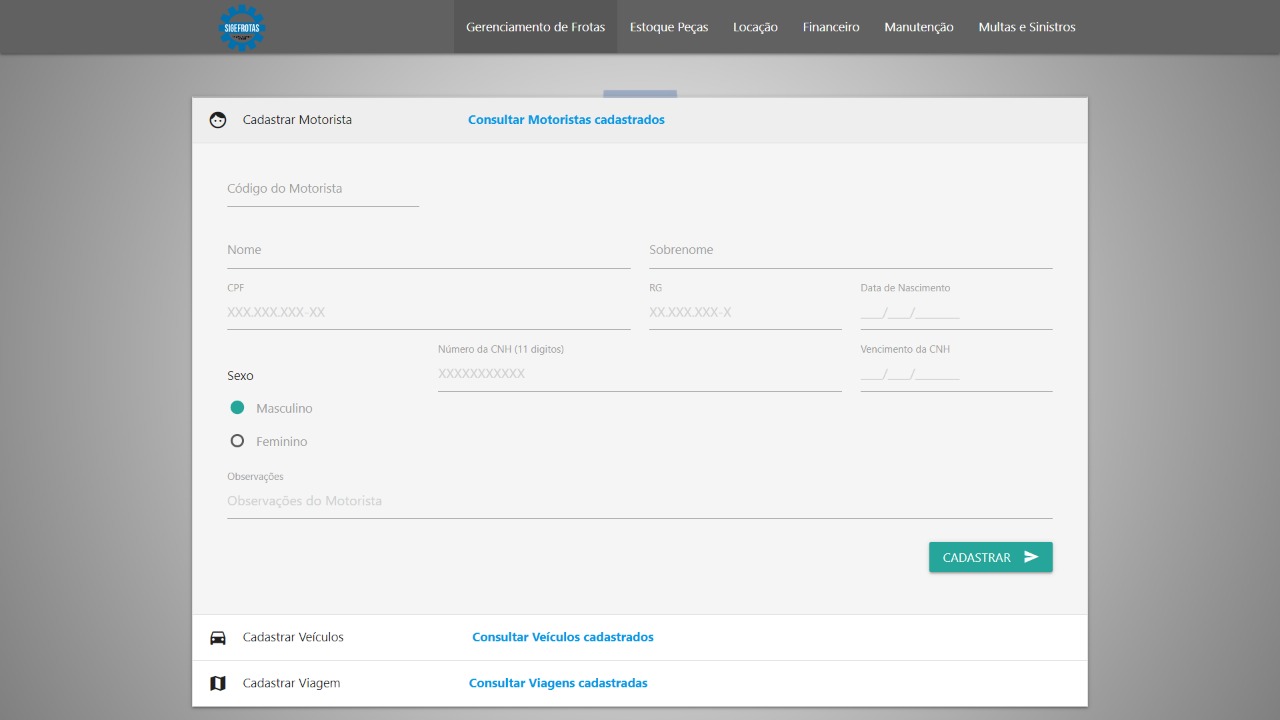
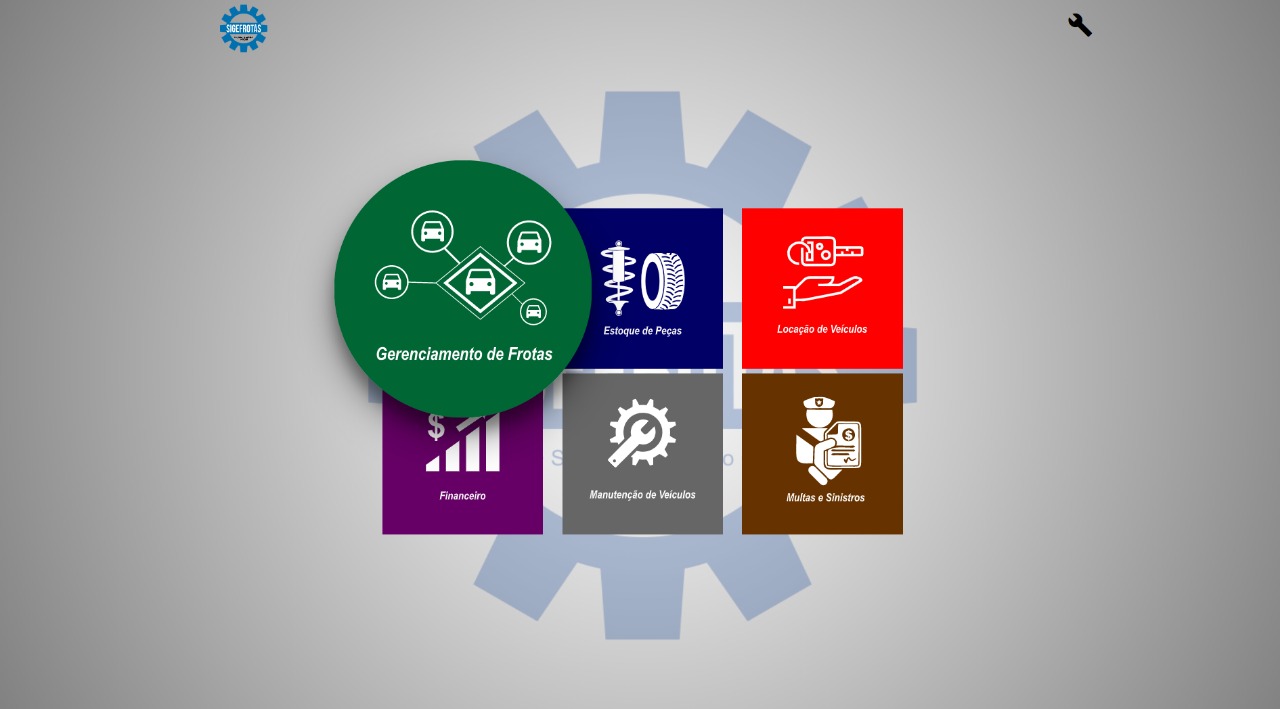
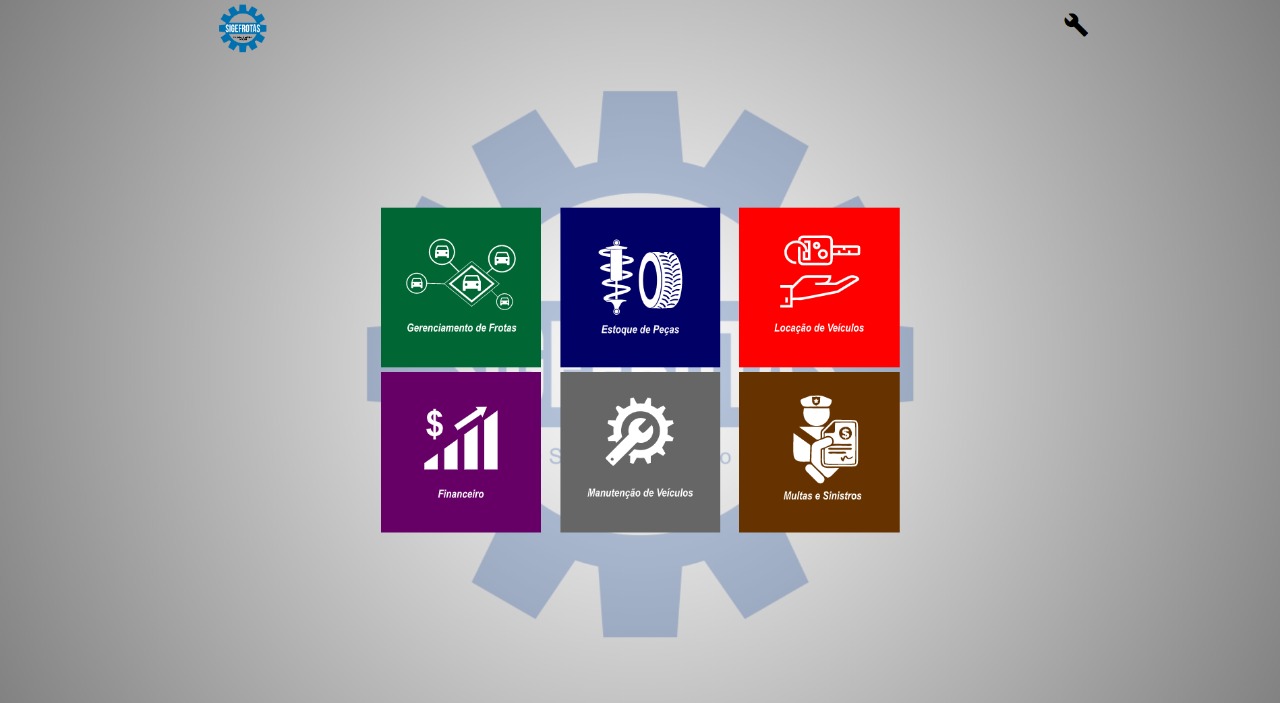
1. Agora na tela de Termos e Uso, recomendamos que leia os Termos antes de aceitá-los, clique em “Concordo” e depois em “Avançar”

1. Após isso é só clicar em “Instalar” e aguardar o fim da instalação do seu Sistema.

1. Depois de terminado clique “Concluir” e o processo de instalação de seu sistema terá sido concluído com sucesso.

## **6.2 Manual ao Usuário**

Imagens : SIGEFROTAS



Fonte: Autor(2019)

As imagens colocadas são referentes a página principal do SIGEFROTAS para ter uma visão melhor à função principal à qual foi criada.

# Conclusão

O desenvolvimento desse estudo possibilitou uma análise de como os levantamentos de requisitos de um software feito sob encomenda pode facilitar em todos os aspectos de desenvolvimento de um programa, começando por realmente entender a real necessidade dos cliente como sistema, e ir destrinchando a partir desse ponto através de diagramas que definem como será realizada uma determinado ação e dando uma ideia geral do sistema.

Através de um questionário enviado a empresas já consolidadas na área de Gestão de Frotas foi possível entender as faces do sistema, e tudo que ele precisa abordar para que faça uma boa gestão da frotas disponíveis, desde quem é o usuário, até entender qual processo precisa seguir para resolver determinada função, mas devido a pouca quantidade de respostas no questionário, houve a necessidade de recolher dados de documentos disponíveis nos próprios sites das empresas, o que por fim deu uma boa ideia de como deve funcionar o sistema como um todo.

Após essa etapa estava na hora de definir todos os módulos e quais processos cada módulo terá para alcançar o objetivo dele, desde o registro de motoristas para necessário para a viagem, até controle de Multas/Sinistros, contratação de seguro e abastecimento dos veículos. Para definir melhor cada um dos processos, foram feitos diagramas detalhando o que era necessário para cada eventualidade.

Com cada uma dessa etapas concluídas conseguimos desenvolver uma ideia base para a aparência do sistema em suas plataformas, assim terminando todo o levantamento dos recursos necessários para que possa ser iniciada a programação do sistema como um todo.

# Referências

# Vazquez, Carlos Eduardo. Engenharia de Requisitos. Software Orientado ao Negócio. Editora Brasport, 2016

# Sites:

# Empresa de Gerenciamento de Frotas Unifrotas.

<http://www.unifrota.com.br/> - Acesso em: 04 de maio de 2019

**Empresa de Gerenciamento de Frotas Produsoft .**

<https://www.produsoft.com.br/informacoes> - Acesso em: 03 de maio de 2019

**Empresa de Gerenciamento de Frotas Masterfrota.**

<https://www.masterfrota.com.br/website/website_home.aspx> - Acesso em: 02 de maio de 2019

**Desenvolvendo um diagrama de Caso de Uso.**

<https://www.devmedia.com.br/o-que-e-uml-e-diagramas-de-caso-de-uso-introducao-pratica-a-uml/23408> - Acesso em: 05 de Junho de 2019

# O que é um Fluxograma de Processos.

<https://certificacaoiso.com.br/o-que-e-fluxograma-de-processos/> - Acesso em: 02 de Junho de 2016