

RUBENS COPCHE BORZANI

**SISTEMA PARA GERENCIAMENTO DE FROTAS
AUTOMOTIVAS**

Assis, SP

2011



Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis - IMESA

RUBENS COPCHE BORZANI

**SISTEMA PARA GERENCIAMENTO DE FROTAS
AUTOMOTIVAS**

Trabalho de Conclusão de Curso
apresentado ao Instituto Municipal de
Ensino Superior de Assis IMESA como
requisito do Curso de Graduação.

ORIENTADOR: Prof. Dr. Luiz Carlos Begosso

Área de Concentração: Desenvolvimento de Sistemas

Assis, SP

2011

FICHA CATALOGRÁFICA

Copche, Rubens

Sistema para Gerenciamento de Frotas Automotivas / Rubens Copche
Borzani, Fundação Educacional do Município de Assis – FEMA –
Assis, 2011.

Número de páginas 65.

Orientador: Prof. Dr. Luiz Carlos Begosso.

Trabalho de Conclusão de Curso – Instituto Municipal de Ensino
Superior de Assis – IMESA.

1. SGFA – Sistema para Gerenciamento de Frotas Automotivas .

CDD: 001.61

Biblioteca da FEMA

SISTEMA PARA GERENCIAMENTO DE FROTAS AUTOMOTIVAS

RUBENS COPCHE BORZANI

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis IMESA como requisito do Curso de Graduação em Processamento de Dados, analisado pela seguinte comissão examinadora:

Orientador: Prof. Dr. Luiz Carlos Begosso

Analisador (1): Profa. Dra. Marisa Nitto

Assis, SP

2011

DEDICATÓRIA

Dedico esse trabalho primeiramente a Deus que me dá amparo em todos os momentos da minha vida. Com muito amor dedico também a meus pais que tudo fizeram pela minha educação e principalmente pela formação do meu caráter e finalmente a minha querida esposa que tanto me apóia e me dá forças para continuar nos meus estudos.

AGRADECIMENTOS

Ao professor Luiz Carlos Begosso, pela orientação, paciência e pelo constante estímulo transmitido desde o primeiro dia dessa jornada.

Agradeço também a todos os professores da FEMA, sem exceções, que direta ou indiretamente contribuíram para o meu enriquecimento intelectual.

Aos meus amigos da faculdade Leonardo, Eduardo que contribuíram muito nos meus trabalhos acadêmicos na troca de conhecimentos e pela amizade que tivemos durante toda essa longa jornada de dificuldades e ALEGRIAS.

Aos meus amigos Carvalho e Marco Antonio, que sempre deram força na minha vida profissional e que foram os grandes incentivadores para que eu começasse esse curso.

A meu irmão Rui pelo apoio em tudo.

E finalmente ao meu supervisor de estágio e amigo Adriano pela grande contribuição que me deu para que eu desenvolvesse esse software dando toda a atenção, ensinamento e crescimento profissional.

“A MENTE QUE SE ABRE A UMA NOVA IDEIA JAMAIS VOLTARÁ AO SEU TAMANHO ORIGINAL.”

ALBERT EINSTEIN.

RESUMO

O Sistema de Gerenciamento de Frotas Automotivas (SGFA) foi desenvolvido para colocar fim nas pranchetas de preenchimento manual, facilitando assim o recolhimento das informações da frota. O SGFA é composto por dois softwares: um para dispositivos móveis e outro para desktop ambos desenvolvidos para plataforma Windows. Os dois softwares se completam sendo que a ideia é buscar as informações contidas no dispositivo móvel (que recolherá os dados em campo) e trazê-las diretamente para o computador com um simples click, assim ganha-se tempo pois não haverá mais digitação por parte do administrador e o mais importante, a empresa contará com maior integridade nos dados recolhidos tendo total controle de sua frota.

ABSTRACT

The Fleet Management System for Automotive (SGFA) was designed to end the drawing board for manual filling, thus facilitating the gathering of information from the fleet. The SGFA has two softwares, mobile and desktop both developed for the Windows platform. Both are complementary and that the idea is to get the information in the mobile device (which collect the data in the field) and bring them directly to your computer with one simple click, so you gain time, because there will be no more typing for the administrator and most importantly, the company will have greater integrity of data collected with total control of your fleet.

LISTA DE FIGURAS

FIGURA 1 - POPULANDO BANCO PRINCIPAL.....	24
FIGURA 2 - CARREGANDO BANCO CE.....	24
FIGURA 3 - POPULANDO BANCO CE.....	25
FIGURA 4 - RECUPERANDO DADOS BANCO CE.....	25
FIGURA 5 – ANALISANDO DADOS CADASTRADOS.....	26
FIGURA 6 - WBS - ESTRUTURA ANALITICA DO PROJETO.....	36
FIGURA 7 - SEQUENCIAMENTO DE ATIVIDADES.....	37
FIGURA 8 - SEQUÊNCIA DE ATIVIDADES (DIAS).....	38
FIGURA 9 - DIAGRAMA DE CASO DE USO DESKTOP.....	44
FIGURA 10 - DIAGRAMA DE CASO DE USO MOBILE.....	45
FIGURA 11 - DIAGRAMA DE CLASSES DESKTOP.....	46
FIGURA 12 - DIAGRAMA DE CASO DE USO DISPOSITIVO MÓVEL.....	47
FIGURA 13 - DIAGRAMA DE ATIVIDADES (ADMINISTRADOR).....	48
FIGURA 14 - DIAGRAMA DE ATIVIDADES (USUÁRIO).....	49
FIGURA 15 - UC1 LOGAR NO SISTEMA.....	50
FIGURA 16 - UC2 CADASTRAR USUÁRIOS.....	51
FIGURA 17 - UC3 CADASTRAR VEÍCULOS.....	52
FIGURA 18 - UC4 CADASTRAR CONTRATOS.....	53
FIGURA 19 - UC5 CADASTRAR ADMINISTRADORES.....	54
FIGURA 20 - UC6 TRANSFERIR DADOS.....	55
FIGURA 21 - UC7 LOGAR NO SISTEMA.....	56
FIGURA 22 - UC1 LOGAR NO SISTEMA.....	57
FIGURA 23 - UC2 CADASTRAR RETIRADA.....	58
FIGURA 24 - UC3 CADASTRAR OBSERVAÇÃO.....	58
FIGURA 25 - UC4 CADASTRAR DEVOLUÇÃO.....	59

SUMÁRIO

1. INTRODUÇÃO.....	18
2. DESENVOLVIMENTO DO TRABALHO.....	19
2.1. Objetivos.....	19
2.2. Justificativa	19
2.3. Motivação.....	19
2.4. Trabalho correlato.....	19
3. LEVANTAMENTO DOS REQUISITOS	21
3.1. Descrição da Organização.....	21
3.2. Descrição do Problema a ser Resolvido.....	21
3.3. Resultados Esperados na Implantação do Software	22
4. ANÁLISE DOS REQUISITOS	23
4.1. Problemas Potenciais.....	23
4.2. Priorização da Implantação dos Requisitos.....	23
4.3. Comportamento do sistema	24
5. METODOLOGIAS DE DESENVOLVIMENTO.....	27
5.1. Descrição da Metodologia.....	27
5.1.1. Principais conceitos da orientação a objeto	27
5.2. Diagramas da UML	28
5.3. DESCRIÇÕES DO AMBIENTE DE DESENVOLVIMENTO	29

5.3.1. Visual Studio 2008 C#	29
5.3.2. SQLEXPRESS	31
5.3.3. SQL Server Compact 3.5 SP2	31
5.3.4. SQL Server Management Studio	32
5.3.5. Crystal Reports.....	32
6. LISTA DE EVENTOS	34
6.1. Desktop.....	34
6.2. Dispositivo móvel	35
7. PLANEJAMENTO DO PROJETO	36
7.1. WBS (Work Breakdown Structure) Estrutura Analítica do Projeto	36
7.2. Sequência de Atividades.....	37
7.3. Cronograma de Atividades.....	39
TABELA 4 – CRONOGRAMA DE ATIVIDADES (1)	39
7.4. ESTIMATIVA DE DURAÇÃO DE ATIVIDADES	40
8. ESPECIFICAÇÃO DOS CUSTOS	41
8.1. Recursos Necessários para o Desenvolvimento do Projeto	41
8.2. ESTIMATIVAS DE CUSTOS.....	42
8.3. Orçamento do Projeto	43
9. DIAGRAMAS	44
9.1. Diagrama de Caso de Uso	44
9.1.1. Desktop	44
9.1.2. Dispositivo Móvel.....	45
9.2. Diagrama de Classes	46
9.2.1. Desktop	46

9.2.2. Dispositivo móvel.....	47
9.3. Diagrama de Atividades	48
9.3.1. Diagrama de Atividades Administrador.....	48
9.3.2. Diagrama de Atividades usuário	49
10. ESPECIFICAÇÃO DE CASO DE USO.....	50
10.1. SGFA Desktop	50
10.1.1.Caso de Uso : Autenticação do Sistema;.....	50
10.1.2.Caso de Uso : Cadastro de usuários;	51
10.1.3.Caso de Uso : Cadastro de veículos;.....	52
10.1.4.Caso de Uso : Cadastro de contratos;	53
10.1.5.Caso de Uso : Cadastro de administradores;	54
10.1.6.Caso de Uso : Transferência de arquivos de banco de dados;.....	55
10.1.7.Caso de Uso : relatórios e Impressões;	56
10.2. SGFA MOBILE	57
10.2.1.Caso de Uso :Autenticação do sistema;	57
10.2.2.Caso de Uso : Movimentação de retirada de veículos	58
10.2.3.Caso de Uso: Cadastro de observações;	58
10.2.4.Caso de Uso : devolução do veículo;.....	59
11. CONCLUSÃO	60
REFERÊNCIAS	61

1. Introdução

A ideia do desenvolvimento desse projeto surgiu no estágio que iniciei em meados de junho de 2010 na empresa TCM. Lá encontrei o projeto do dispositivo móvel iniciado, mas não concluído. Foi iniciado então o desenvolvimento do projeto para desktop e dado continuidade ao projeto do dispositivo móvel.

Hoje as empresas buscam agilidade, rapidez e antecipação de resultados (“Tempo é dinheiro”, Benjamin Franklin (1706-1790)) então por que não desenvolver um software para controlar toda a frota e mostrar seus rendimentos, seus déficits e até uso descontrolado de veículos por parte dos usuários.

Sendo assim encontrei uma grande oportunidade para iniciar nessa área promissora de TI, conseguir experiência profissional, adequar-me ao mercado de trabalho e conseguir uma colocação nele.

Hoje a administração de toda frota da empresa TCM é realizada por planilhas manuais (escritas), dificultando o controle dos eventos ocorridos com os veículos. Com o crescimento da mesma as dificuldades de controle nas filiais aumentaram, fazendo-se necessário a criação de um software para facilitar o manuseio desses dados. O software que inicialmente será testado na matriz tem por finalidade, a longo prazo, controlar toda a sua frota. Esse controle será feito através de um servidor central que receberá todos os dados colhidos pelo dispositivo móvel (POCKET). O Dispositivo móvel estará presente dentro de cada veículo e fará a coleta dos dados em tempo real.

A implantação desse novo sistema trará para o administrador da frota, melhor controle dos veículos e aumentará a integridade dos dados registrados, facilitando a emissão de relatórios e possibilitando melhor análise das ocorrências.

2. Desenvolvimento do Trabalho

2.1. Objetivo

O presente trabalho de conclusão de curso tem por objetivo o desenvolvimento do software denominado de Sistema de Gerenciamento de Frotas Automotivas (SGFA), o sistema proporcionará aos controladores ou administradores de frotas, maior acesso aos dados, integridade nos relatórios, fazer cálculos de quilometragem e gastos com mais facilidade e rapidez para localizar ocorrências como data, hora, quilometragem, usuário, locais, etc.

2.2. Justificativa

O desenvolvimento deste projeto tem grande valia, tendo em vista, que o sistema vem para simplificar e agilizar o processo de recuperação dos dados obtidos em campo, proporcionando aos gerenciadores da frota maior rapidez para diagnosticar gastos feitos por contrato, gastos feitos por veículos, gastos feitos por usuários, quilometragem rodada por veículo, quilometragem rodada por usuário, quilometragem rodada por contrato, etc.

2.3. Motivação

A motivação se deu pelo fato de estar iniciando na programação C# e encontrar um problema dentro da empresa que seria minimizado com essa API (Application Programming Interface). Outro ponto importante para o desenvolvimento do presente trabalho é a oportunidade de aprendizado que terei na área de desenvolvimento de software.

2.4. Trabalho correlato

Conforme mencionado no início desse trabalho, destaca-se que este autor promoveu a continuidade do desenvolvimento do software do dispositivo móvel,

desenvolvido por Moreli (2010), denominado de DESENVOLVIMENTO DE APLICATIVO PARA POCKET PC UTILIZANDO C#.

A partir do trabalho desenvolvido por Moreli (2010) continuou-se a implementação do software do dispositivo móvel incrementando-se mais alguns recursos e quando terminado foi iniciada a implementação de outro software desktop para se fazer a integração entre o dispositivo móvel e o Desktop.

As alterações feitas no software de Moreli (2010) foram as seguintes:

- Escolha de cidades e estados em lista, sendo as cidades filtradas por estado (mais de 9000 cidades cadastradas). Esses estados e cidades são carregados a partir do banco de dados CE (Compact Edition);

- Escolha de contratos em lista. Dados carregados a partir do banco dados CE;

- Tratamento de abertura e fechamento de quilometragem;

- CRUD completo de contratos e empresas;

- Validação dos usuários para quilometragens já abertas;

- Tratamento para não se perder quilometragens abertas caso o dispositivo móvel tenha queda de bateria;

- Carregamento automático da última quilometragem inserida no veículo na próxima abertura de ocorrência.

O termo "integração" refere-se a uma atividade de desenvolvimento de software em que componentes separados são combinados em um todo. A integração é realizada em vários níveis e estágios da implementação.

É importante frisar e entender que a integração ocorre pelo menos uma vez em cada iteração (processo chamado na programação de repetição de uma ou mais ações). Um plano de iteração define os casos de uso que devem ser projetados e as classes que devem ser implementadas. O enfoque da estratégia de integração é determinar a ordem em que as classes são implementadas e combinadas.

3. LEVANTAMENTO DOS REQUISITOS

3.1. Descrição da Organização

Tucunduva & Carvalho Motta, mais conhecida como **TCM Serviços**, surgiu em 20 de novembro de 1992. Através do trabalho sólido e confiável que se traduz em investimentos humanos e materiais, vem conquistando a credibilidade do mercado. Criada a partir da cultura empresarial, da experiência e da legislação, a **TCM Serviços** tem a garantia de bons resultados fundados em preceito céticos e morais.

A **TCM Serviços** é composta por diversas empresas com determinadas modalidades de mão de obra, tornando-se assim, o GRUPO TCM, que tem uma filosofia organizacional firmemente estabelecida, colaborando com os diversos segmentos de serviços terceirizados, fornecendo pessoal altamente especializado. Sempre atento às novas tecnologias em equipamentos e produtos, a empresa oferece o que há de mais moderno aos seus clientes, dando soluções rápidas e eficazes às necessidades apresentadas, buscando gerar satisfação e qualidade em toda sua gama de serviços.

3.2. Descrição do Problema a ser Resolvido

As empresas estão aumentando sua frota de veículos a cada dia, sendo assim o fluxo de informações gerado a partir deste crescimento é gigantesco. E essa situação deixa cada vez mais difícil o controle de quilometragem percorrida, multas, abastecimento, controle de usuários e muitas outras ocorrências. Por isso a necessidade de um sistema ágil e personalizado para atender a demanda e controle de tantas informações.

3.3. Resultados Esperados na Implantação do Software

Espera-se obter maior controle dos dados armazenados, rapidez de acesso a eles e evitar alterações que possam comprometer o orçamento da empresa.

Agilizar o processo de recolhimento dos dados que vem de campo até o administrador, proporcionando assim total controle da frota através de relatórios.

Espera-se também que o sistema desenvolvido para *Windows Mobile* permita aos usuários fácil manuseio e impeça alterações de datas e quilometragem depois de cadastradas.

4. ANÁLISE DOS REQUISITOS

4.1. Problemas Potenciais

- ✓ Login para a segurança do sistema;
- ✓ Cadastro das ocorrências;
- ✓ Cadastrar novos Administradores;
- ✓ Cadastrar novos usuários da frota;
- ✓ Cadastrar novos contratos;
- ✓ Cadastrar novos veículos;
- ✓ Fazer transferência de dados para o servidor ou PC local;
- ✓ Fazer transferência de dados para dispositivo móvel;
- ✓ Emitir relatórios diversos.

4.2. Priorização da Implantação dos Requisitos

- Garantir um sistema realmente seguro;
- Manter cadastro das ocorrências;
- Manter cadastro dos Administradores;
- Manter cadastro dos usuários;
- Manter cadastro dos contratos;
- Manter cadastro dos veículos;
- Garantir transferência íntegra dos dados para o servidor ou PC;
- Garantir transferência íntegra dos dados para o dispositivo móvel;
- Consultas diversas;
- Emitir relatórios das ocorrências e gastos diversos.

4.3. Comportamento do sistema

Administrador loga no sistema desktop faz todos os cadastros necessários e os grava no banco principal que estará no servidor, tem acesso a esses dados depois de cadastrados (Figura 1).



Figura 1 – Populando banco principal

Administrador logado e com permissões faz transferência dos dados do banco principal para o banco do dispositivo móvel (Figura 2).

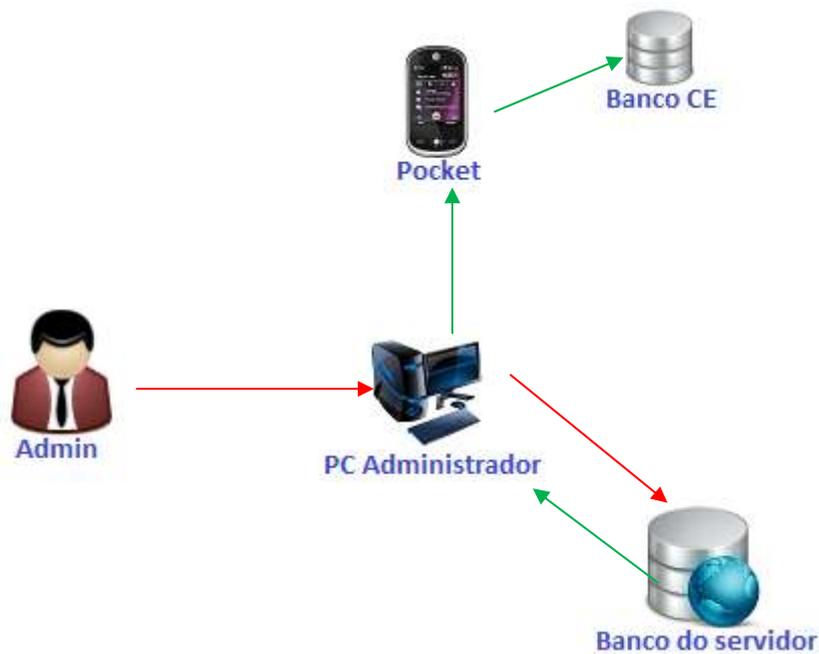


Figura 2 – Carregando banco CE

Usuário loga no sistema do dispositivo móvel e cadastra as ocorrências do veículo (Figura 3).

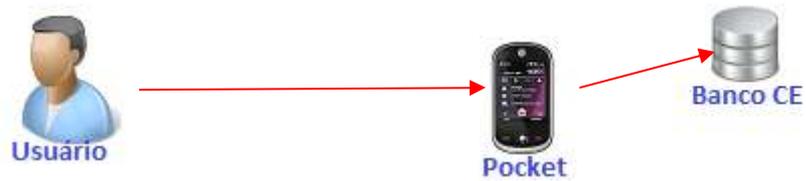


Figura 3 – Populando banco CE

Administrador logado e com permissões recupera os dados do banco do dispositivo móvel e grava no banco principal (Figura 4).

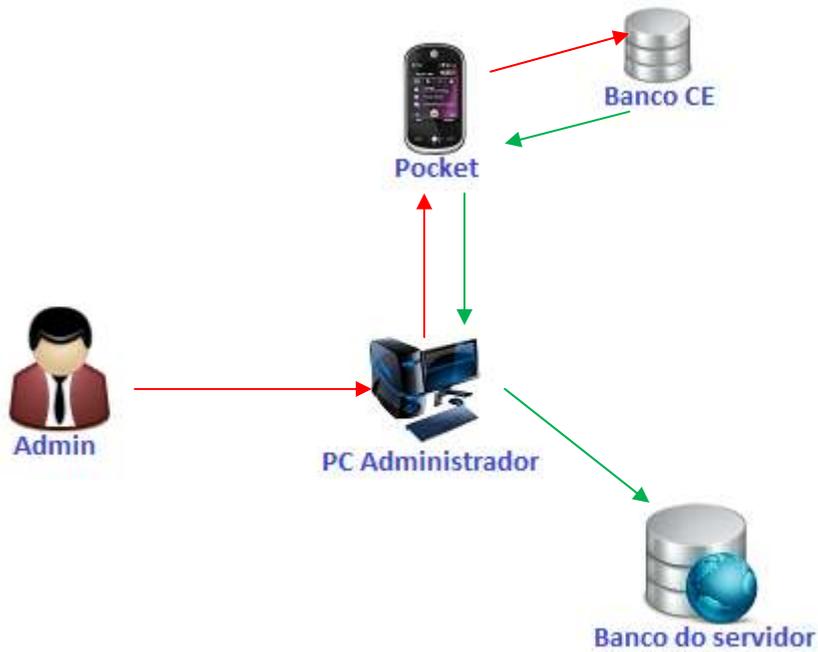


Figura 4 – Recuperando dados do banco CE

Administrador loga no sistema gera relatórios diversos tem acesso a todos os dados cadastrados pelo usuário. (Figura 5).



Figura 5 – Analisando dados cadastrados

5. METODOLOGIAS DE DESENVOLVIMENTO

5.1. Descrição da Metodologia

Desenvolver um software de gerenciamento ou administrativo é conseguir chegar a um sistema ou aplicação que possa suprir as necessidades ou problemas de um grupo, organização ou empresa.

Dentro do contexto de que existem vários métodos para se desenvolver softwares, está o conceito de orientação a objeto que é uma boa prática de programação.

Esse conceito busca abstrair situações ou coisas do mundo real para uma linguagem de software. Isso facilita a decisão do que implementar ou como implementar. É usado também para diminuir as dificuldades de comunicação do profissional modelador com o usuário da área alvo.

“Na programação orientada a objetos, implementa-se um conjunto de classes que definem os objetos presentes no sistema de software. Cada classe determina o comportamento (definido nos métodos) e estados possíveis (atributos) de seus objetos, assim como o relacionamento com outros objetos.” Sica C. (2006).

5.1.1. Principais conceitos da orientação a objeto

Classe: Representa um conjunto de objetos;

Objeto ou instância: Pode armazenar em seus atributos e reagir a mensagens enviadas a ele;

Atributo: São as características de um objeto;

Método: Definem as habilidades dos objetos.

Mensagem: Usada para invocar um de seus métodos;

Herança: Aproveita-se os métodos e atributos da classe pai;

Associação: É o mecanismo pelo qual um objeto utiliza os recursos de outro.

Encapsulamento: Consiste na separação de aspectos internos e externos de um objeto.

Abstração: Concentra-se nos aspectos essenciais de um contexto qualquer, ignorando características menos importantes ou acidentais.

Polimorfismo: Um método pode ter várias formas;

Interface: Deixa público para uso os métodos da classe que a implementa;

Pacotes: São referências para organização lógica de classes e interfaces.

Portando quando usamos a orientação a objetos fazemos a organização do software em grupos de objetos onde esses incorporam uma estrutura e tem comportamentos próprios. Podendo ser usados entre as várias camadas do projeto de software.

5.2. Diagramas da UML

São diagramas que auxiliam desenvolvedores a ter uma visão mais detalhada e padronizada de seus produtos facilitando assim o processo de desenvolvimento. Porém ela não é uma metodologia de desenvolvimento e a sua finalidade não é dizer o que fazer, como fazer e qual sequência, mas sim organizar os detalhes para que o processo de criação seja visualizado de uma forma mais clara.

“Os objetivos da UML são: especificação, documentação, estruturação para sub-visualização e maior visualização lógica do desenvolvimento completo de um sistema de informação. A UML é um modo de padronizar as formas de modelagem. Embora a UML defina uma linguagem precisa, ela não é uma barreira para futuros aperfeiçoamentos nos conceitos de modelagem.”
Lee R. (2001).

5.3. Descrições do Ambiente de Desenvolvimento

5.3.1. Visual Studio 2008 C#

C# é uma linguagem de programação orientada a objetos criada pela Microsoft, faz parte da sua plataforma .NET. É uma linguagem que chama a atenção de desenvolvedores por ter uma sintaxe limpa e de fácil entendimento.

A linguagem C# possui a especificação Common Language Specification (CLS) que é um subconjunto de recursos do CTS (Common Type System) que as linguagens e compiladores devem obedecer, para garantir a interoperabilidade de obedecer, programas escritos em diferentes linguagens. Proporcionando uso e possibilitando estender classes escritas em outras linguagens.

Dentro de suas características, é importante citar, que ele suporta o uso de ponteiros Unsafe Code (código não seguro), possui Garbage Colector (Coletor de lixo), faz herança múltipla (herda duas ou mais classes), sobrecarga, polimorfismo, permite criação de novos tipos de dados a partir dos existentes.

Seus objetivos:

- Orientação a componentes;
- Orientação a objetos;
- Robustez;
- Preservar investimentos.

“O C# se originou da família C de linguagens de programação (C, C++, Objective C, etc.)”. Troelsen (2009).

A linguagem C# também aceita três tipos de passagem de parâmetros:

Valor - as alterações que ocorrem nas variáveis passadas por cópia dentro do método não se refletem fora dele;

Referência (ref) - toda alteração feita na variável passada por referência dentro do método alterará também seu valor fora dele;

Saída (out) - é usada para se inicializar um parâmetro durante a execução do método, retornando o resultado para o método que o chamou.

5.3.1.1. Sobre as Classes

- Campos;
- Métodos;
- Constantes;
- Propriedades;
- Construtores;
- Operadores;
- Eventos;
- Indexadores;
- Destruutores;

5.3.1.2. Tipos Pré-definidos C#

A tabela 1 relaciona tipos de variáveis primitivas e objetos usados na linguagem C#.

Tipos comuns	C# Alias	Descrição
System.Object	<i>object</i>	Base class for all CTS types
System.String	<i>string</i>	String
System.SByte sbyte	<i>Signed</i>	8-bit byte
System.Byte byte	<i>Signed</i>	8-bit byte
System.Int16 short	<i>Unsigned</i>	valor 16-bit
System.Int32 int	<i>Unsigned</i>	valor 16-bit
System.UInt16 ushort	<i>Signed</i>	valor 32-bit
System.UInt32 uint	<i>Unsigned</i>	valor 32-bit
System.Int64 long	<i>Signed</i>	valor 64-bit
System.UInt64 ulong	<i>Unsigned</i>	valor 64-bit
System.Char	<i>char</i>	Caracter 16-bit Unicode
System.Single	<i>float</i>	32-bit float
System.Double	<i>double</i>	64-bit float
System.Boolean	<i>bool</i>	Valor booleano (true/false)
System.Decimal	<i>decimal</i>	128-bit 28 ou 29 casas decimais, usado para aplicações financeiras.

Tabela 1 – Tipos pré-definidos C#

Fonte (MSDN – 2011)

Concluindo, C # é uma linguagem muito poderosa, e são poucas as coisas que você pode querer fazer em C++ que não possa fazer em C #. Faço esse comentário, pois o C# possui recursos avançados que muitas vezes só se encontram em linguagens mais primitivas como C, C++ e algumas outras. Por exemplo: ter acesso direto a memória do sistema e conseguir manipulá-la, para isso basta habilitar código não seguro (*UNSAFE*) e trabalhar da forma que desejar. Devo lembrar que essa é uma técnica de programação avançada e pode ser potencialmente perigosa (daí seu nome *UNSAFE*), porém não vem ao caso detalhá-la, pois não é o foco desse trabalho.

Essas características proporcionam uma ligeira noção do quanto é poderosa a linguagem C# para um programador com conhecimentos avançados de sistema.

5.3.2. SQLEXPRESS

O Microsoft SQL Express é um sistema de gerenciamento de banco de dados relacional cliente/servidor (SGBDR) de alto desempenho. Possui um servidor programável, que pode simplificar substancialmente o desenvolvimento de aplicações através de *procedures*, *triggers*, *function* e *packages*.

Ele é a versão gratuita da plataforma de dados SQL Server também desenvolvida pela Microsoft. Apesar de não contar com todos os recursos que o SQL Server possui também é uma ferramenta muito poderosa que proporciona aos desenvolvedores bons resultados.

5.3.3. SQL Server Compact 3.5 SP2

O SQL Server Compact 3.5 SP2 é uma plataforma de dados livre, fácil de usar e que permite aos desenvolvedores criarem um banco de dados robusto para aplicações móveis que funcionam em todas as plataformas Windows, incluindo

Windows XP, Vista, Windows 7, *Pocket PC* e Smartphone, sendo assim o mais indicado para dispositivos móveis que contenham plataforma Windows.

Em comparação ao SQL Server ele tem algumas limitações: tem capacidade de dados reduzida por ser um RGBDR para dispositivos móveis o número de recursos também é reduzido mas mantém grande integridade nos dados.

5.3.4. SQL Server Management Studio

SQL Server Management Studio é uma ferramenta que compõe o Microsoft SQL Server 2005 e versões posteriores para configurar, gerenciar e administrar todos os componentes do Microsoft SQL Server. A ferramenta inclui editores de script e ferramentas gráficas que trabalham com objetos e recursos do servidor.

Com essa ferramenta foi possível realizar nesse projeto toda a manipulação nas plataformas de dados tanto o SQL Express quanto no SQL Server CE para *mobile*.

Apesar do Visual Studio 2008 também conter ferramentas para configurar, gerenciar e administrar dados no SQL Server, encontrei mais facilidades em trabalhar com ferramentas separadas, pois acredito que dessa forma consiga-se uma organização maior para se gerenciar os dados.

5.3.5. Crystal Reports

O Crystal Reports é uma ferramenta de design de relatório que permite criar relatórios capazes de recuperar e formatar um conjunto de resultados a partir de um banco de dados ou outra fonte de dados por exemplo um classe encapsulada.

Ele surgiu em uma pequena empresa em Vancouver chamada Crystal Services que em 1988 desenvolveu um relatório baseado em DOS para a ACCPAC contabilidade. Alguns anos mais tarde, lançou relatórios divulgando-os como o

"primeiro escritor do mundo de relatórios Windows", pouco depois a Microsoft normalizaria o Crystal Reports, como o mecanismo de relatório para o Visual Basic.

"The flexibility of the report designer means that it can be used for many different types of output, depending on your needs." McAmis (2004).

6. LISTA DE EVENTOS

6.1. Desktop

A Tabela 2 mostra detalhadamente as ações que os administradores do sistema executam para fazer uso do mesmo.

Nº	DESCRIÇÃO	CASO DE USO
01	Administrador loga no sistema	Efetuar login
02	Administrador gerencia cadastro de Administradores	Cadastrar administradores e permissões
03	Administrador gerencia cadastro de usuários	Cadastrar usuários
04	Administrador gerencia cadastro de veículos	Cadastrar veículo
03	Administrador gerencia cadastro de contratos	Cadastrar contratos
	Administrador gerencia cadastro de empresas	Cadastrar empresas
05	Administrador tem permissões de transferência de dados do Dispositivo móvel para o PC.	Transferir dados para Desktop
06	Administrador tem permissões de transferência de dados do PC para Dispositivo móvel.	Transferir dados para dispositivo móvel
07	Administrador emite relatórios	Gerar relatórios
08	Administrador pode imprimir relatórios	Imprimir relatórios
09	Administrador pode gerar arquivos em vários formatos	Gerar arquivo

Tabela 2 – Lista de Eventos Desktop

6.2. Dispositivo móvel

A Tabela 3 mostra detalhadamente as ações que os usuários do sistema executam para fazer uso do mesmo.

Nº	DESCRIÇÃO	CASO DE USO
01	Usuário loga no sistema	Efetuar login
02	Usuário escolhe veículo	Escolher veículo
03	Usuário faz o cadastro da retirada do veículo	Cadastrar retirada
04	Usuário pode fazer cadastros ocorrências com o veículo	Cadastrar ocorrência
05	Usuário faz o cadastro da devolução do veículo	Cadastrar devolução

Tabela 3 – Lista de Eventos Dispositivo móvel

7. PLANEJAMENTO DO PROJETO

7.1. WBS (Work Breakdown Structure) Estrutura Analítica do Projeto

A Figura 6 ilustra a estrutura de todo o planejamento do projeto, começando pelo plano de desenvolvimento até o momento de implantação.

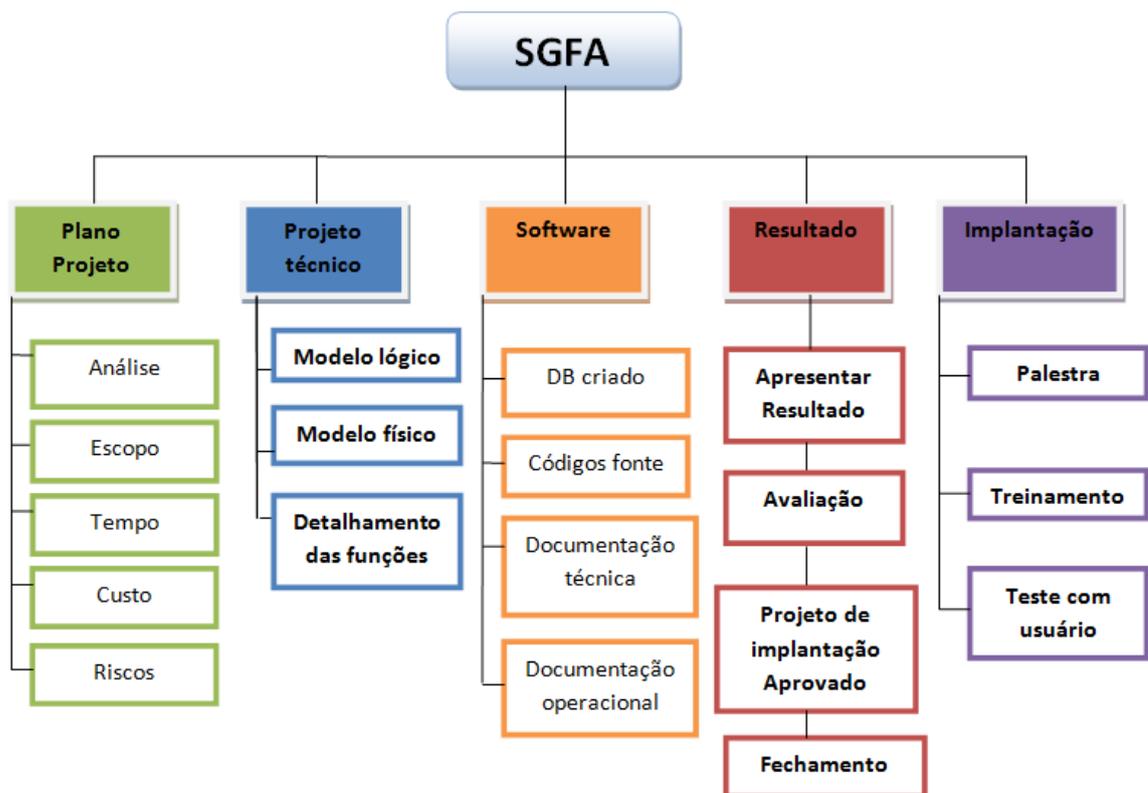


Figura 6 - WBS - Estrutura Analítica do Projeto

7.2. Sequência de Atividades

O gráfico de sequência de atividades ilustra o andamento do projeto com algumas de suas fases e trabalhos realizados paralelamente (Figura 7).

Já na Figura 8, a sequência de atividades é mostrada em dias para a realização do projeto.

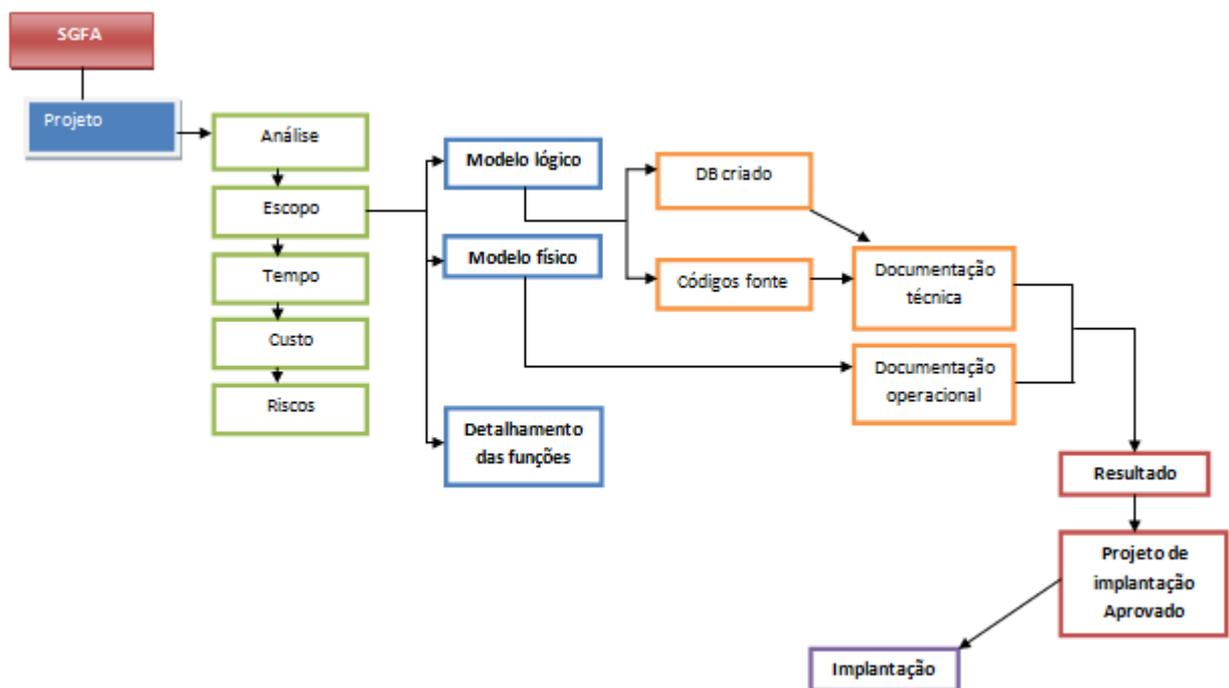


Figura 7 - Sequenciamento de atividades

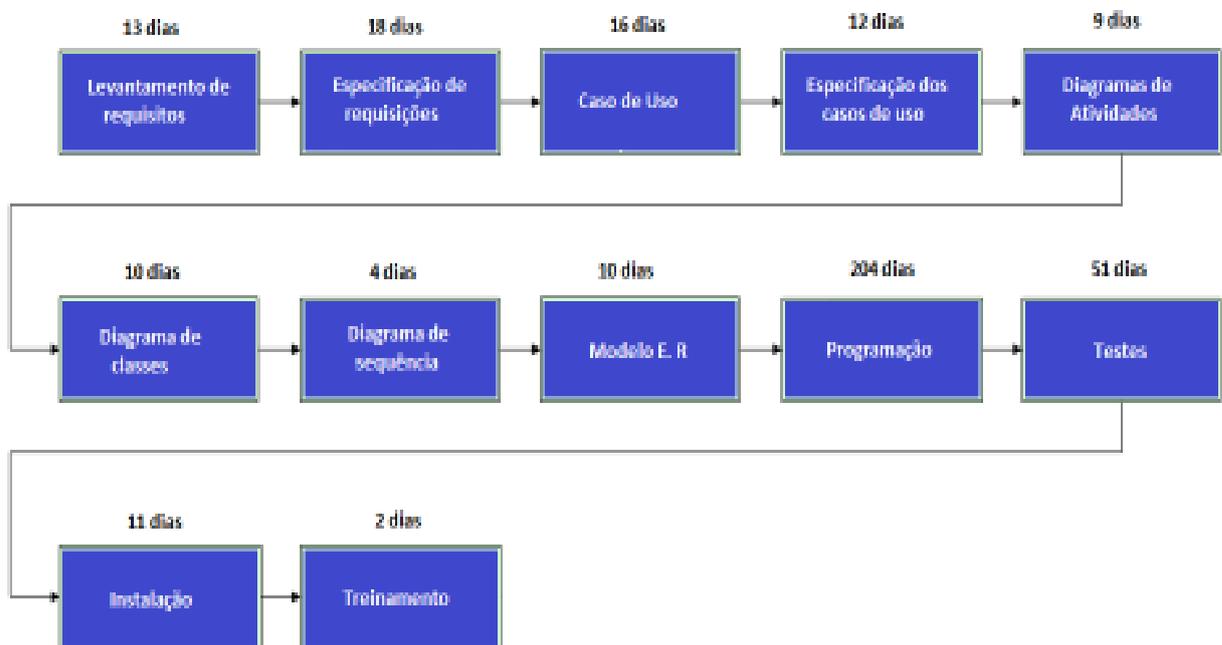


Figura 8 - Sequência de Atividades (Dias)

7.3. Cronograma de Atividades

A Tabela 4 de cronograma de atividades apresenta as atividades realizadas no segundo semestre de 2010, já a Tabela 5 representa as atividades do primeiro semestre de 2011.

2010	Jul				Ago				Set				Out				Nov				Dez			
Atividade/Tempo	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Levantamento de Requisitos																								
Especificação de Requisitos																								
Caso de Uso																								
Especificação dos UC																								
Diagrama Atividades																								
Diagrama Classes																								
Diagrama de Seqüência																								
Modelo E-R																								
Programação																								
Teste																								
Instalação																								

Tabela 4 – Cronograma de atividades (1)

2011	Jan				Fev				Mar				Abr				Mai				Jun			
Atividade/Tempo	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Levantamento de Requisitos																								
Especificação de Requisitos																								
Caso de Uso																								
Especificação dos UC																								
Diagrama Atividades																								
Diagrama Classes																								
Diagrama de Seqüência																								
Modelo E-R																								
Programação																								
Teste																								
Instalação																								

Tabela 5 – Cronograma de atividades (2)

7.4. ESTIMATIVA DE DURAÇÃO DE ATIVIDADES

Na estimativa de duração de atividades encontra-se os cálculos de dias usados para realização de cada atividade em forma de tabela (tabela 6).

ATIVIDADE	DATA INICIAL	DATA FINAL	TOTAL DIAS
Levantamentos Requisitos	11/07/2010	24/07/2010	13
Especificação Requisitos	25/07/2010	12/08/2010	18
Definição dos Casos de Uso	01/08/2010	17/08/2010	16
Diagrama de Caso de Uso	19/08/2010	22/08/2010	3
Especificação Caso de Uso	29/08/2010	10/09/2010	12
Diagrama de Classes	06/09/2010	16/09/2010	10
Diagrama de Atividades	17/09/2010	26/09/2010	9
Diagrama Seqüência	26/09/2010	30/09/2010	4
Modelo ER	29/09/2010	09/10/2010	10
Programação	05/10/2010	27/04/2011	204
Teste	05/03/2011	25/04/2011	51
Instalação	18/04/2011	29/04/2011	11
Total entre a data de inicio e a data do fim.	11/07/2010	29/04/2011	292

Tabela 6 – Estimativa de duração das atividades

8. ESPECIFICAÇÃO DOS CUSTOS

8.1. Recursos Necessários para o Desenvolvimento do Projeto

Os recursos necessários para o desenvolvimento do projeto SGFA para a empresa TCM estão relacionados a baixo, bem como as quantidades de cada recurso necessárias para se desenvolver o projeto proposto.

Sendo que os custos do projeto do analista, programador estão vinculados as horas de estágio, cedidas pelo Grupo TCM. Este que se propõe a capacitar o aluno Rubens Copche Borzani e aumentar seus conhecimentos na área de desenvolvimento de software.

Humano

- 01 Analista / Programador;

Equipamento

- 01 Notebook Core Duo de 2.10 Ghz, 4 Gb de memória RAM e 320Gb de HD;
- 01 Motorola A3100;
- 01 impressora multifuncional HP;

Softwares

- Argo UML;
- Visual Studio 2008;
- SGDBR Microsoft SQL Express 2008;
- SGDBR Microsoft SQL Server CE 3.5 SP2;
- Microsoft SQL Server Management Studio 2008;
- Crystal Reports 2008;

8.2. ESTIMATIVAS DE CUSTOS

Custo dos dispositivos para a Desenvolvimento:

Equipamento:

01 Notebook:

- Valor unitário = R\$3.000,00 para desenvolvimento.
- Dias (de uso) = 212 dias (considerar: 1060hs / 5 hs de trabalho/dia = 212)
- Depreciação = R\$3.000,00 / 24 meses (24 meses Tempo de depreciação) = R\$125,00mês.
- Custo dia = R\$125,00 / 212 (dias) = R\$ 0,65 (ao dia)
- Custo de cada computador = R\$0,59 * 1060 (total de horas do projeto) = R\$ 625,40.

01 Motorola A3100:

- Valor unitário = R\$1.500,00 para teste do software.
- Dias (de uso) = 80 dias (considerar: 400hs / 5 hs de trabalho/dia = 80)
- Depreciação = R\$1.500,00 / 24 meses (24 meses Tempo de depreciação) = R\$62,50mês.
- Custo dia = R\$62,50/80 (dias) = R\$0.78 (ao dia)
- Custo de cada Dispositivo = R\$0.78 * 400 (total de horas do projeto) = R\$312,00.

01 Impressora multifuncional HP:

- Valor unitário = R\$499,00 para teste do software.
- Dias (de uso) = 60 dias (considerar: 300hs / 5 hs de trabalho/dia = 60)
- Depreciação = R\$499,00 / 24 meses (24 meses Tempo de depreciação) = R\$20,79mês.
- Custo dia = R\$20,79/60 (dias) = R\$0.34 (ao dia)
- Custo de cada impressora = R\$0.34 * 300 (total de horas do projeto) = R\$104,00.

8.3. *Orçamento do Projeto*

A Tabela 7 mostra o gasto com os recursos para o desenvolvimento do projeto.

ITENS	VALOR
Analista/Programador	R\$3.200,00
Notebook	R\$ 625,40
Impressora	R\$ 104,00
01 Motorola A3100	R\$ 312,00
Banco de Dados	R\$ 0,00
Visual Studio Express	R\$ 0,00
Crystal Reports	R\$ 0,00
Custo Total	R\$ 4.241,40

Tabela 7 – Orçamento do projeto

9. DIAGRAMAS

9.1. Diagrama de Caso de Uso

O diagrama de caso de uso descreve a funcionalidade proposta para um novo sistema, que será projetado.

“Diagrama de Caso de Uso é um documento narrativo que descreve a sequência de eventos de um ator que usa um sistema para completar um processo” (Ivar Jacobson, 2000).

9.1.1. Diagrama de Caso de Uso Desktop

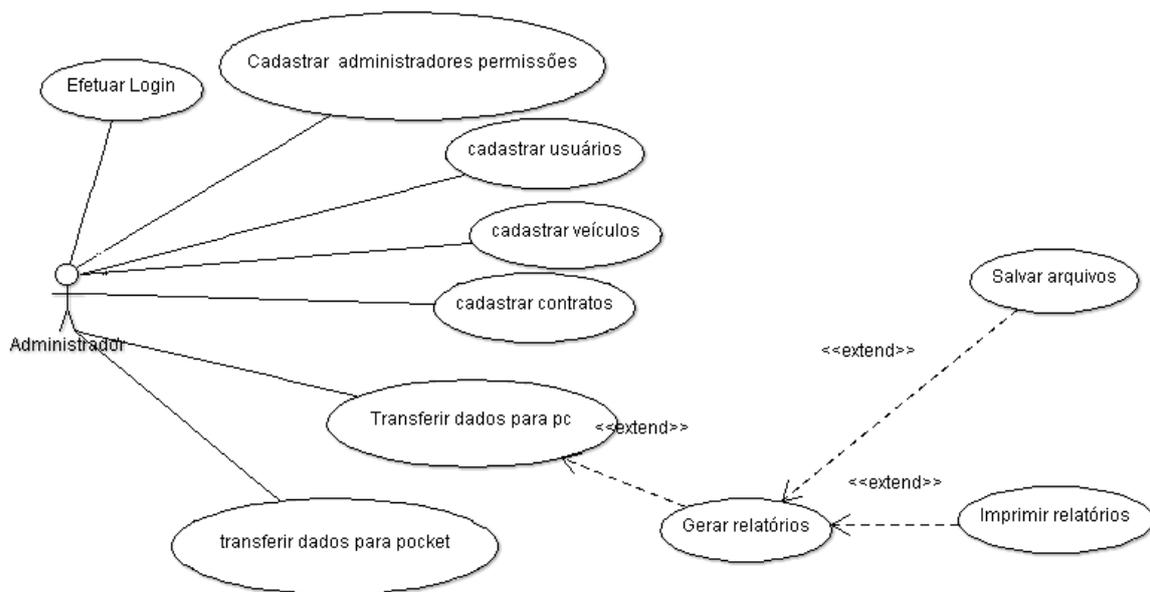


Figura 9 - Diagrama de Caso de Uso Desktop

9.1.2. Diagrama de Caso de uso Dispositivo Móvel

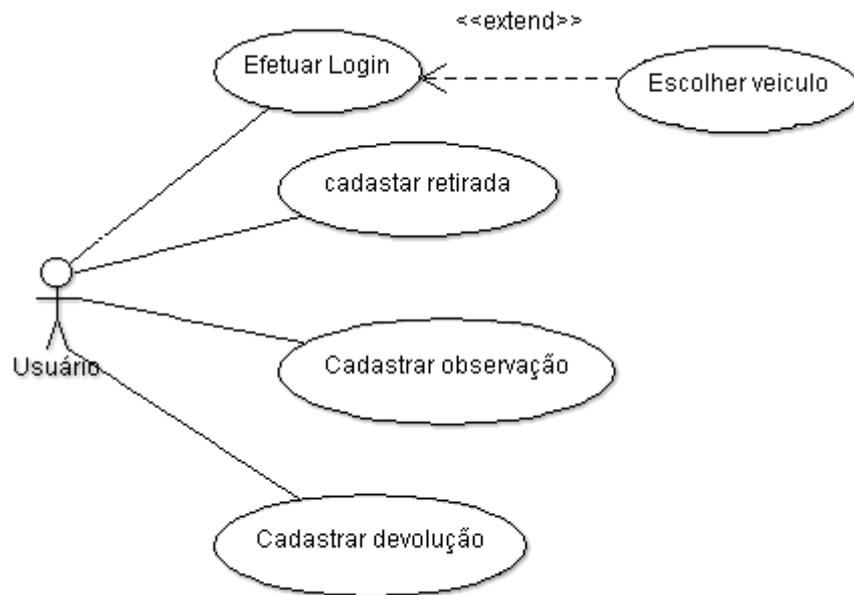


Figura 10 - Diagrama de Caso de Uso mobile

9.2. Diagrama de Classes

Para programadores o Diagrama de Classes tem a função de representação da estrutura e relação das classes que servem de modelo para objetos (Figuras 11 e 12).

9.2.1. Diagrama de Classes Desktop

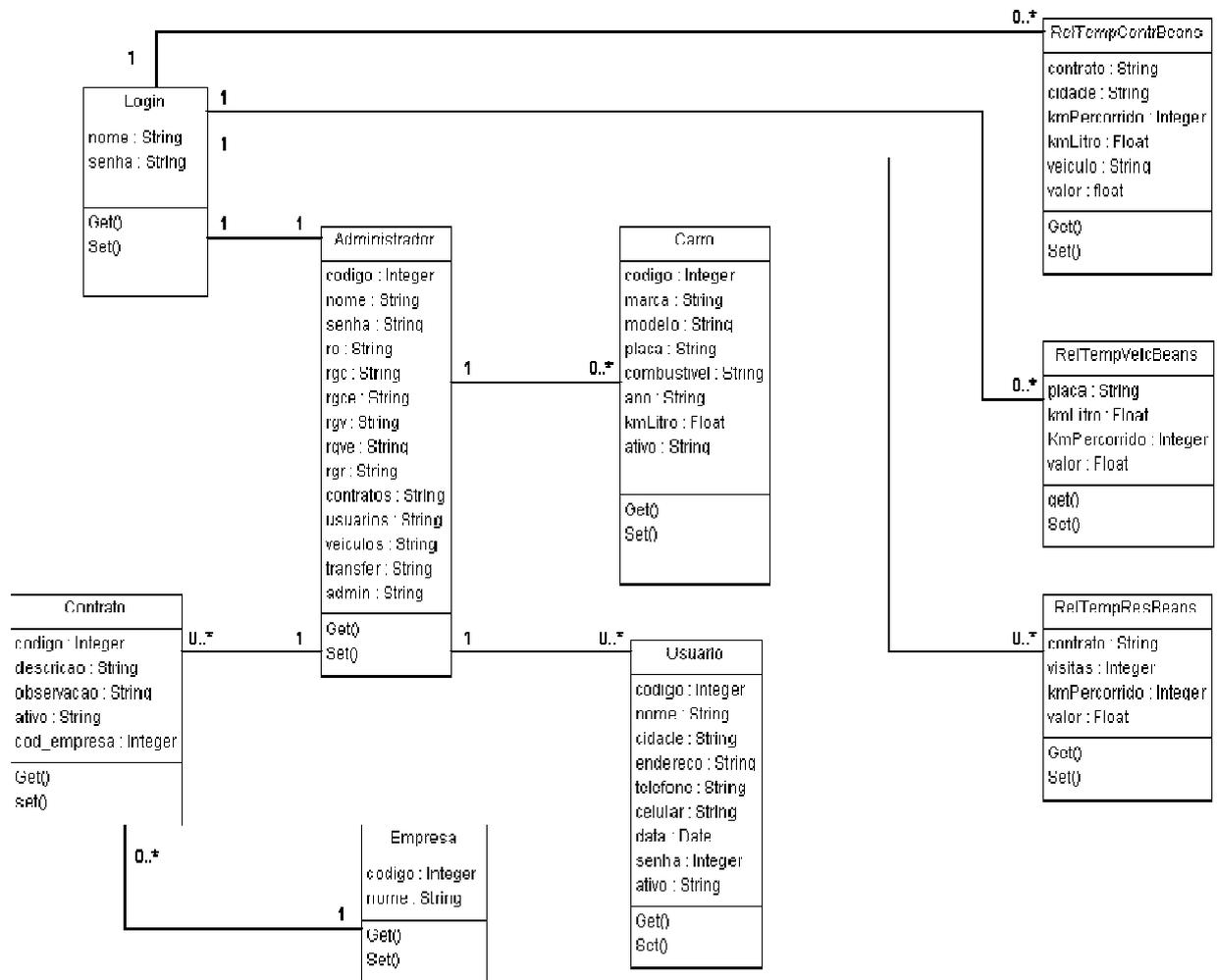


Figura 11 - Diagrama de classes Desktop

9.2.2. Diagrama de Classes Dispositivo móvel

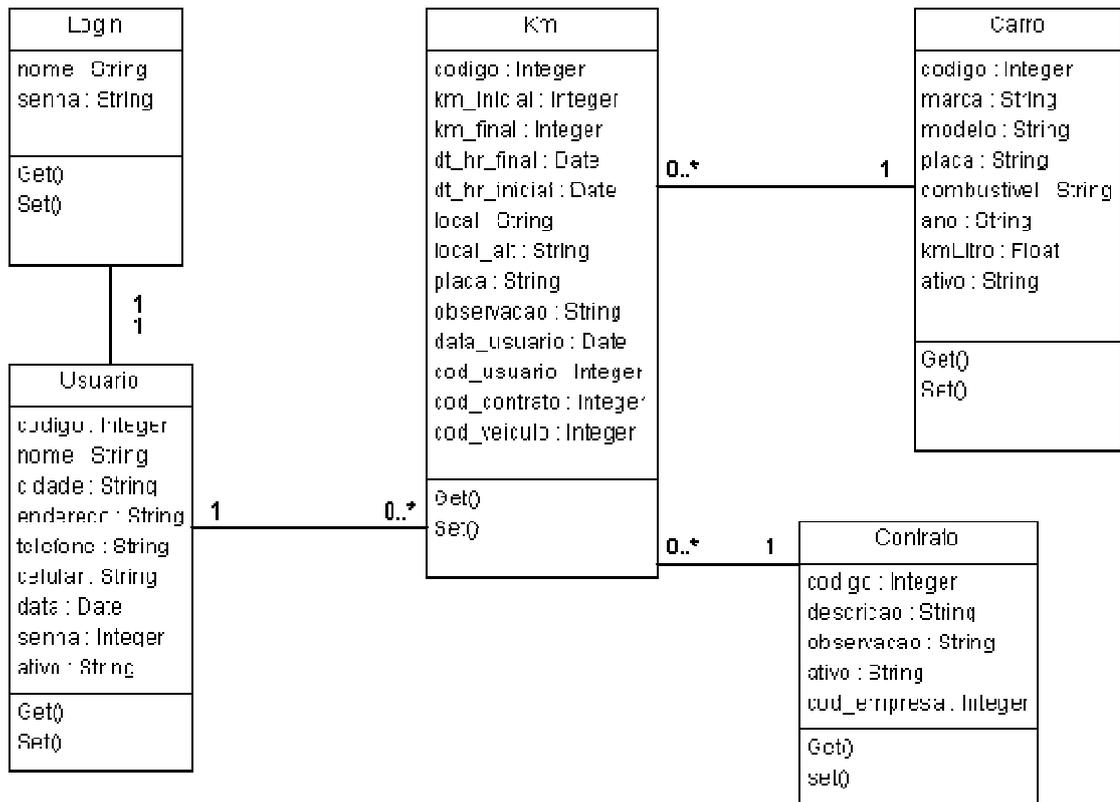


Figura 12 - Diagrama de Caso de Uso Dispositivo Móvel

9.3. Diagrama de Atividades

Os diagramas de atividade representam os fluxos conduzidos por processamentos, ou seja ilustram o uso do software e suas ações perante algum pedido ou requisição do usuário ou administrador (Figuras 13 e 14).

9.3.1. Diagrama de Atividades Administrador

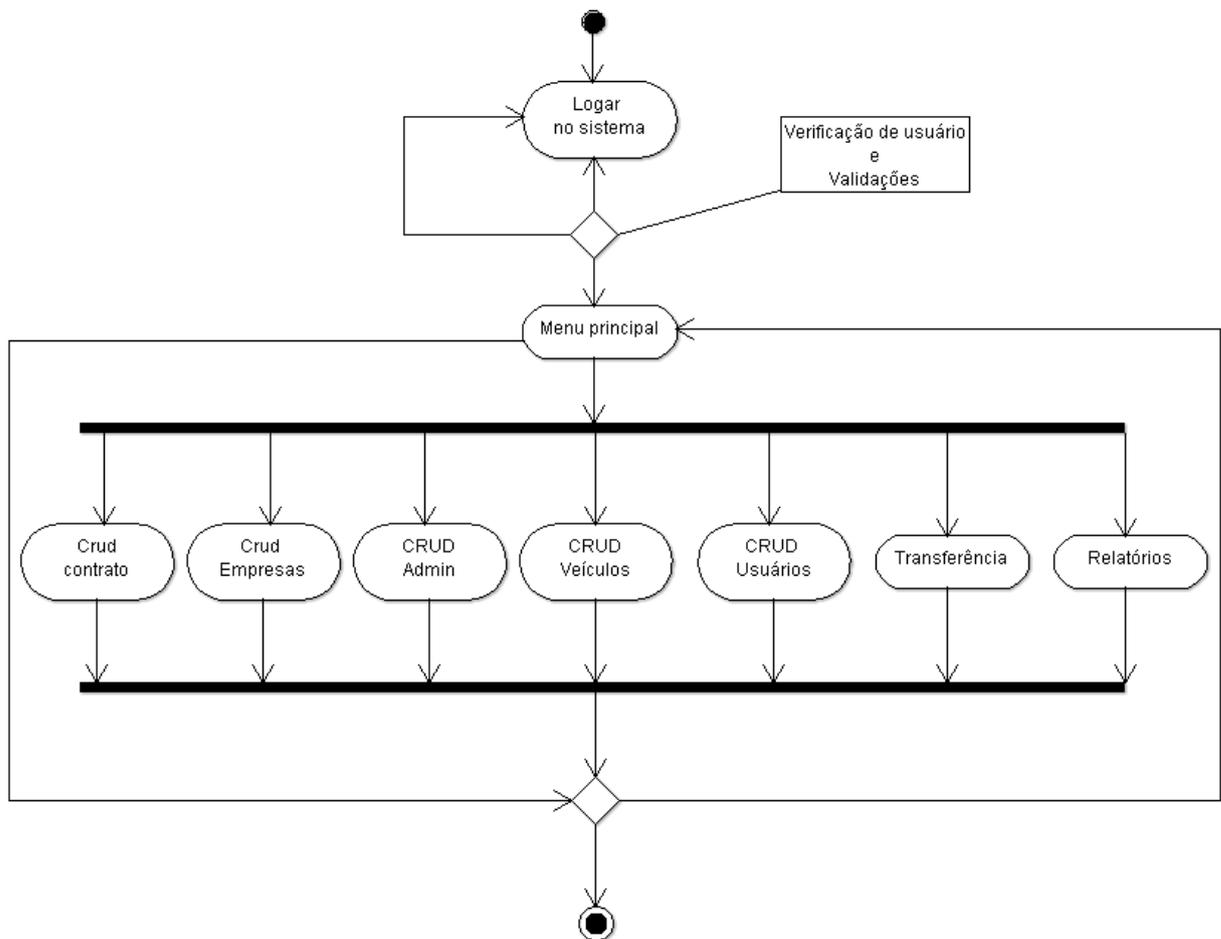


Figura 13 - Diagrama de Atividades (Administrador)

9.3.2. Diagrama de Atividades usuário

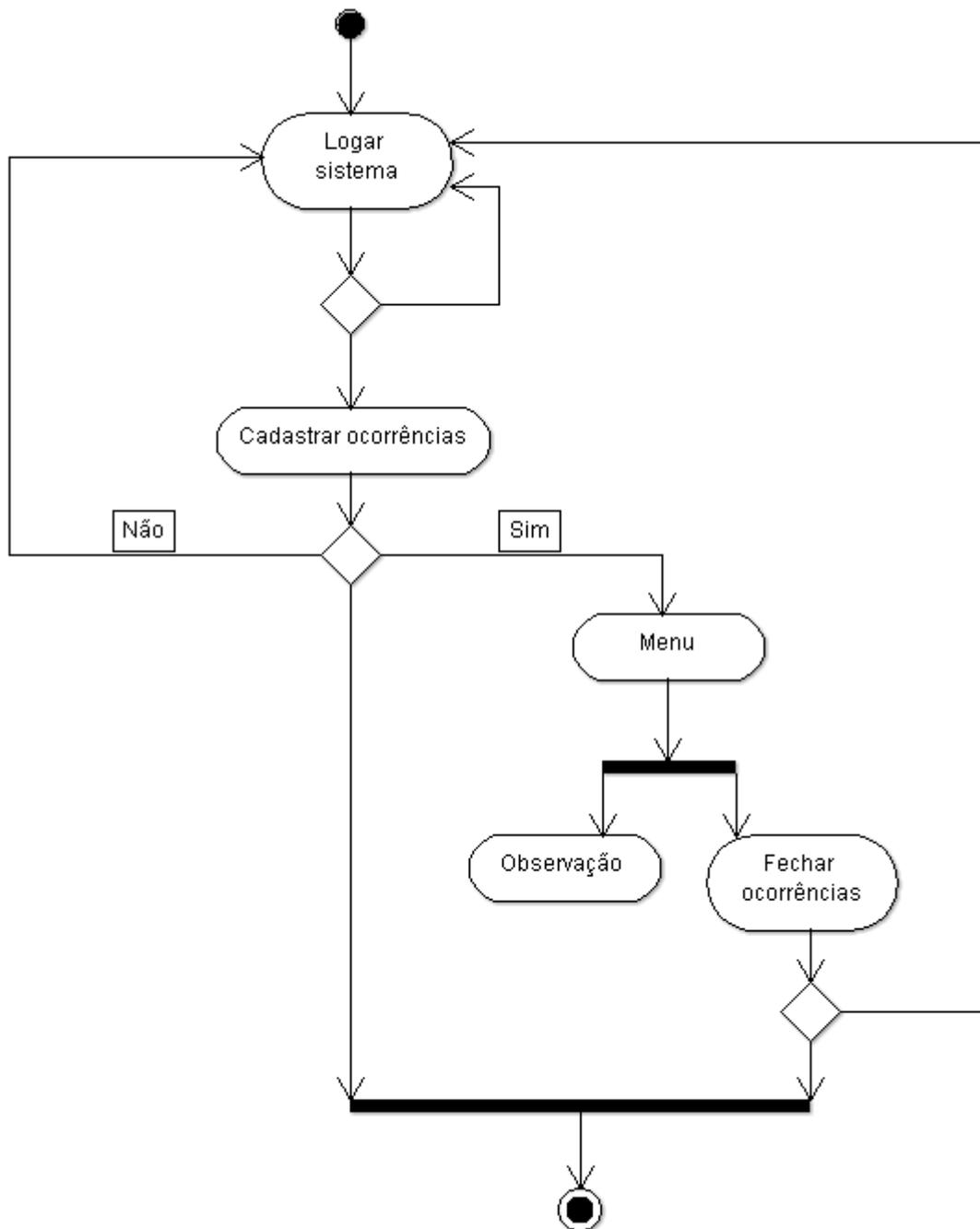


Figura 14 - Diagrama de Atividades (usuário)

10. ESPECIFICAÇÃO DE CASO DE USO

Aqui aparece nas figuras seguintes detalhadamente cada caso de uso do SGFA para desktop e Dispositivo móvel.

10.1. SGFA Desktop

10.1.1. Caso de Uso : Autenticação do Sistema;

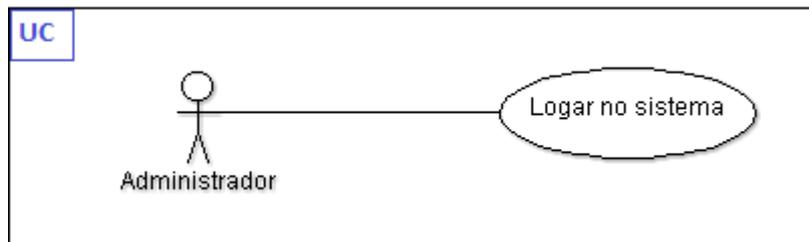


Figura 15 - UC1 Logar no sistema

Ator: Administrador;

- O sistema oferece a interface para o administrador efetuar o login;
- O ator efetua o login; (A1)
- O sistema direciona o ator para a tela de menu de acordo com o seu nível e permissões.

Fluxo Alternativo [A1]: Erro de Login;

- O sistema exibe a mensagem de que o usuário ou a senha está incorreto.
- O sistema retorna ao login;

10.1.2. Caso de Uso : Cadastro de usuários;

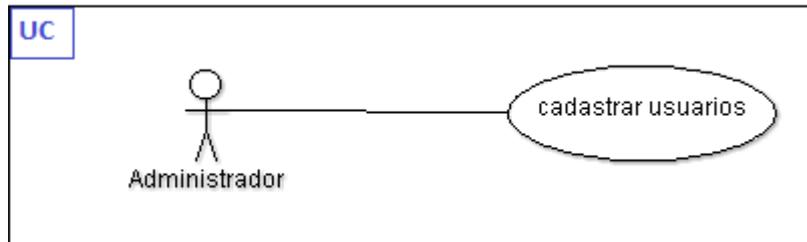


Figura 16 - UC2 Cadastrar Usuários

Ator: Administrador;

- O ator seleciona cadastro de usuários;
- O sistema exibe as contas já existentes; (A1, A2)
- O sistema oferece a Interface para um novo cadastro;
- O ator preenche os dados e grava as informações;
- O sistema informa que o cadastro foi realizado com sucesso;
- O sistema atualiza os usuários cadastrados existentes.

Fluxo Alternativo [A1]: Excluir;

- O Ator Seleciona um Registro para exclusão;
- O Sistema pergunta se o ator tem certeza;
- O Sistema exclui o registro, e exibe a mensagem de concretização;

Fluxo alternativo [A2]: Alterar Registro;

- O Ator seleciona o registro a ser alterado;
- O Sistema exibe as informações do Registro;
- O Ator altera os dados do registro e clica em salvar;
- O Sistema altera os dados e exibe a mensagem de que a operação foi realizada com sucesso;

10.1.3. Caso de Uso : Cadastro de veículos;

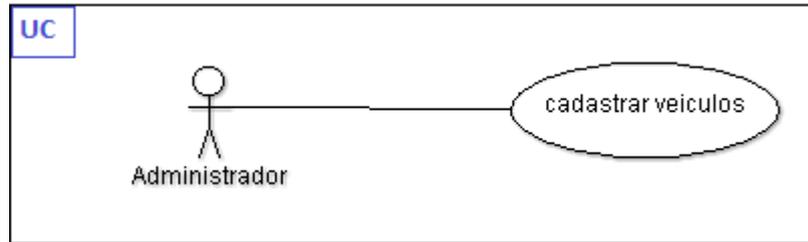


Figura 17 - UC3 Cadastrar Veículos

Ator: Administrador;

- O ator seleciona cadastro de Veículos;
- O sistema exibe os veículos já existentes; (A1, A2)
- O sistema oferece a Interface para um novo cadastro;
- O ator preenche os dados e salva as informações;
- O sistema informa que o cadastro foi realizado com sucesso;
- O sistema atualiza a lista de veículos cadastrados existentes.

Fluxo Alternativo [A1]: Excluir;

- O Ator seleciona um registro para exclusão;
- O Sistema pergunta se o ator tem certeza;
- O Sistema exclui o registro, e exibe a mensagem de concretização;

Fluxo alternativo [A2]: Alterar Registro;

- O Ator seleciona o registro a ser alterado;
- O Sistema exibe as informações do Registro;
- O Ator altera os dados do registro e clica em salvar;
- O Sistema altera os dados e exibe a mensagem de que a operação foi realizada com sucesso;

10.1.4. Caso de Uso : Cadastro de contratos;

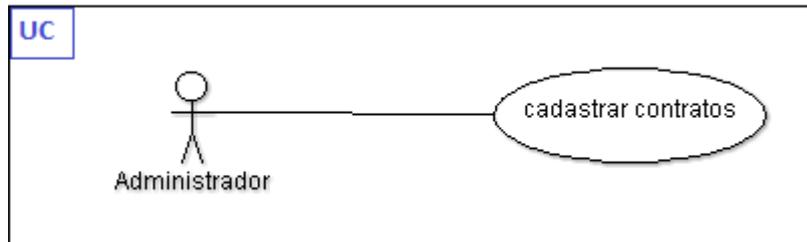


Figura 18 - UC4 Cadastrar Contratos

Ator: Administrador;

- O ator seleciona cadastro de contratos;
- O sistema exibe os contratos já existentes; (A1, A2)
- O sistema oferece a Interface para um novo cadastro;
- O ator preenche os dados e grava as informações;
- O sistema informa que o cadastro foi realizado com sucesso;
- O sistema atualiza a lista de contratos cadastrados existentes.

Fluxo Alternativo [A1]: Excluir;

- O Ator seleciona um registro para exclusão;
- O Sistema pergunta se o ator tem certeza;
- O Sistema exclui o registro, e exibe a mensagem de concretização;

Fluxo alternativo [A2]: Alterar Registro;

- O Ator seleciona o registro a ser alterado;
- O Sistema exibe as informações do Registro;
- O Ator altera os dados do registro e clica em salvar;
- O Sistema altera os dados e exibe a mensagem de que a operação foi realizada com sucesso;

10.1.5. Caso de Uso : Cadastro de administradores;

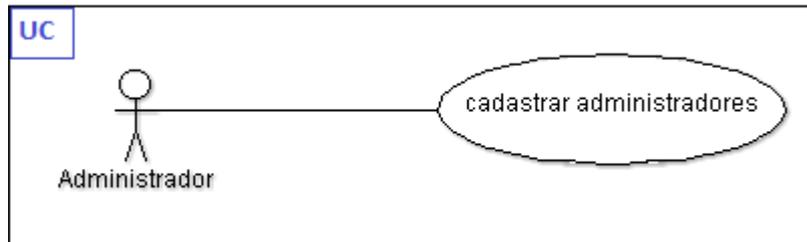


Figura 19 - UC5 Cadastrar Administradores

Ator: Administrador;

- O ator seleciona cadastro de administradores;
- O sistema exibe os administradores já existentes; (A1, A2, A3)
- O sistema oferece a Interface para um novo cadastro;
- O ator preenche os dados e salva as informações;
- O sistema informa que o cadastro foi realizado com sucesso;
- O sistema atualiza a lista de veículos cadastrados existentes.

Fluxo Alternativo [A1]: Excluir;

- O Ator seleciona um registro para exclusão;
- O Sistema pergunta se o ator tem certeza;
- O Sistema exclui o registro, e exibe a mensagem de concretização;

Fluxo alternativo [A2]: Excluir Registro Admin (root);

- O Ator seleciona um registro para exclusão;
- O Sistema pergunta se o ator tem certeza;
- O Sistema exibe mensagem de Erro ao excluir Root;

Fluxo alternativo [A3]: Alterar Registro;

- O Ator seleciona o registro a ser alterado;
- O Sistema exibe as informações do Registro;
- O Ator altera os dados do registro e clica em salvar;
- O Sistema altera os dados e exibe a mensagem de que a operação foi realizada com sucesso;

10.1.6. Caso de Uso : Transferência de arquivos de banco de dados;

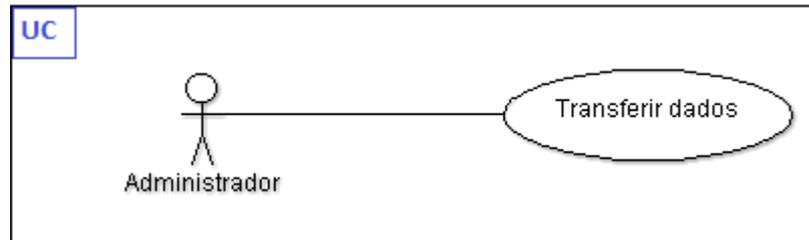


Figura 20 - UC6 Transferir Dados

Ator: Administrador;

- O ator seleciona transferência (A1, A2);
- Após transferência dos dados do dispositivo móvel para o PC o ator pode manipular os dados para relatórios.

Fluxo Alternativo [A1]: Dispositivo móvel para PC;

- O Sistema faz a transferência direta do banco do dispositivo móvel para a pasta backup no pc;
- O Sistema faz leitura do banco trazido do dispositivo móvel e faz a inserção dos dados lidos no banco principal;
- O Sistema exibe a mensagem de concretização.

Fluxo alternativo [A2]: PC para Dispositivo móvel;

- O sistema faz a leitura das atualizações no banco principal;
- O sistema faz a inserção das atualizações dos dados lidos no banco SQL Server CE que está localizado na pasta Transfer;
- O Sistema executa a transferência do banco para o dispositivo móvel;
- Sistema exibe mensagem de concretização.

10.1.7. Caso de Uso : relatórios e Impressões;

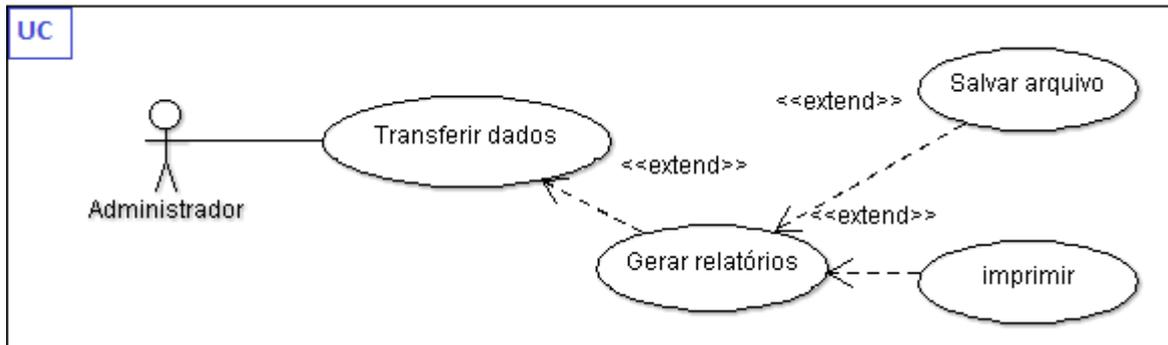


Figura 21 - UC7 Logar no Sistema

Ator: Administrador;

- O ator seleciona relatórios;
- O sistema da opções de vários relatórios;
- Relatórios manipuláveis;
- Relatórios fixos;
- O ator escolhe o tipo de relatório que deseja (A1, A2, A3);
- O sistema cria o relatório com Crystal Reports;

Fluxo Alternativo [A1]: Manipulação ;

- O ator monta o relatório;
- O ator escolhe gerar relatório;

Fluxo Alternativo [A2]: Gravar ;

- O Ator tem a opção de salvar o relatório;
- O Sistema da a opção de varias extensões de arquivo para que o ator escolha;
- O Sistema gera o tipo de arquivo escolhido e envia a mensagem de sucesso.

Fluxo Alternativo [A3]: imprimir relatório;

- O Ator tem a opção de imprimir o relatório;
- O sistema abre a interface de impressão;
- O ator escolhe a impressora e numero de cópias;
- O sistema faz a impressão.

10.2. SGFA MOBILE

10.2.1. Caso de Uso :Autenticação do sistema;

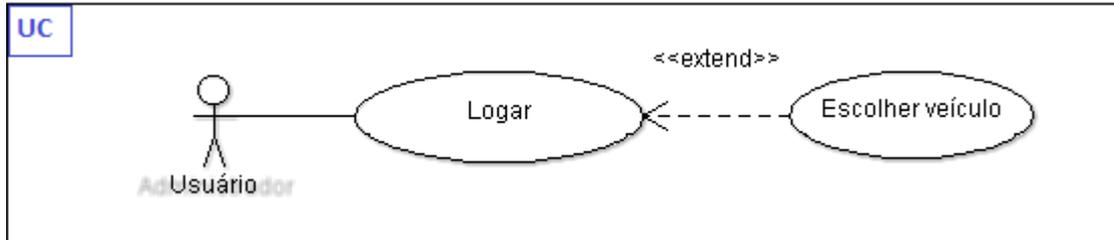


Figura 22 - UC1 Logar no Sistema

Ator: Usuário;

- O sistema oferece a interface para o usuário efetuar o login;
- O sistema permite a escolha do veículo;
- O ator faz a escolha do veículo;
- O ator efetua o login; (A1)
- O sistema direciona o ator para a tela abertura de quilometragem.

Fluxo Alternativo [A1]: Erro de Login;

- O sistema exibe a mensagem de que o usuário ou a senha está incorreto.
- O sistema retorna ao login;

10.2.2. Caso de Uso : Movimentação de retirada de veículos;

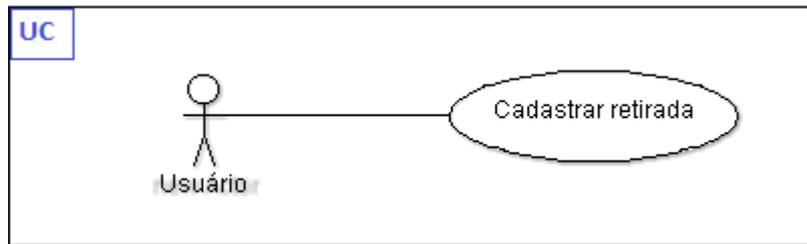


Figura 23 - UC2 Cadastrar Retirada

Ator: Usuário;

- O sistema oferece a Interface para preenchimento da situação do veículo;
- O ator preenche os dados para as informações serem enviadas ao banco;
- O ator envia a informações; (A1)
- O sistema devolve mensagem de sucesso;
- O sistema direciona o ator para o menu;

Fluxo Alternativo [A1]: Erro de envio;

- O sistema devolve mensagem de erro;

10.2.3. Caso de Uso: Cadastro de observações;

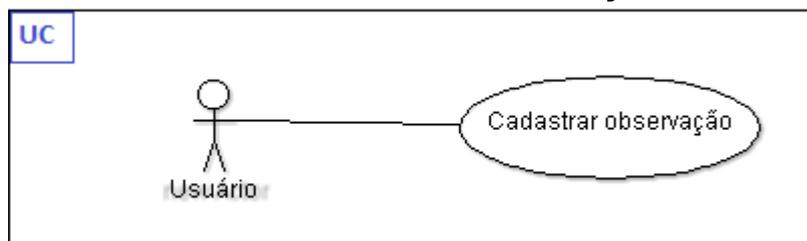


Figura 24 - UC3 - Cadastrar Observação

Ator: Usuário;

- O ator escolhe o item observação;
- O sistema oferece a Interface para preenchimento de observações;
- O ator preenche com as observações;
- O ator salva essas observações;
- O sistema armazena as observações em uma variável estática;

- O sistema devolve mensagem de sucesso;
- O sistema direciona o ator novamente para tela de menu;

10.2.4. Caso de Uso : devolução do veículo;

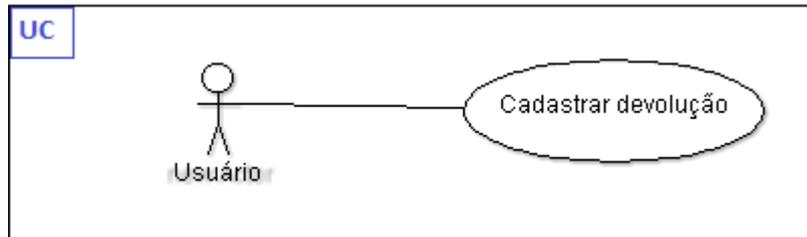


Figura 25 - UC4 Cadastrar Devolução

Ator: Usuário;

- O ator escolhe fecha a quilometragem;
- O sistema oferece a Interface para preenchimento da situação do veículo novamente (observação);
- O sistema traz o valor da variável estática;
- O ator preenche os dados para as informações serem enviadas ao banco;
- O autor envia a informações; (A1)
- O sistema devolve mensagem de sucesso;
- O sistema direciona o ator novamente para a tela de login;

Fluxo Alternativo [A1]: Erro de envio;

- O sistema devolve mensagem de erro;

11. CONCLUSÃO

Além dos conhecimentos adquiridos no curso de Processamento de Dados que foram fundamentais para a base de todo este trabalho, um ponto importante que deve ser citado foi o estágio realizado no Grupo TCM. Este que aumentou muito meus conhecimentos em desenvolvimento C# dando base para iniciar a vida profissional como programador.

Grandes foram os esforços para conseguir vencer essa etapa de estudos e principalmente a insegurança de se criar o primeiro Sistema para uma empresa. Mas todas as expectativas e metas foram atingidas trazendo uma grande felicidade e confiança profissional. Todas as funcionalidades do sistema desenvolvidas até o presente momento estão terminadas e testadas.

Espero que esse software seja de grande valia para a empresa e possa ajudar na tomada de decisões.

E que esse Trabalho de Conclusão de Curso possa trazer benefícios àqueles que o buscarem para leitura.

REFERÊNCIAS

BOOCH, G.; RUMBAUCGH J. e JACOBSON I.; **UML Guia do Usuário**. Tradução de Fábio Freitas da Silva. Rio de Janeiro: Editora Campus, 2000.

Carlos Sica. **PHP Orientado a Objetos - Fale a Linguagem da Internet**. 1 ed. Rio de Janeiro: Ciência Moderna, 2006.

John S.; Microsoft Visual C# 2008 - Passo a Passo; 1º edição; Ed. Bookman; 2006.

LEE, R. C; TEPFENHART, W. M; **UML e C++ Guia Prático de Desenvolvimento Orientado a Objeto**. Tradução de Celso Roberto Paschoa. São Paulo: Editora Makron Books, 2001.

McAmis D.; **Professional Crystal Reports for Visual Studio.NET**; Second Edition; by Wiley Publishing; 2004.

MSDN, tipos pré-definidos C#, [http://msdn.microsoft.com/en-us/library/aa664635\(v=vs.71\).aspx](http://msdn.microsoft.com/en-us/library/aa664635(v=vs.71).aspx) Acesso em 07 jun. 2011.

MORELI, Vitor. **DESENVOLVIMENTO DE APLICATIVO PARA POCKET PC UTILIZANDO C#**. Trabalho de Conclusão de Curso, Tecnologia em Processamento de dados (FEMA – Fundação Educacional do Município de Assis), São Paulo, Assis, 2010.

Toelsen A.; **Profissional C# e a Plataforma .NET 3.5 Curso Completo**; 1º edição, 2009.

Wikipedia, orientação a objetos, http://pt.wikipedia.org/wiki/Orientação_a_objetos Acesso em 03 jun. 2011.

Wikipedia, UML, <http://pt.wikipedia.org/wiki/UML> Acesso em 31 mai. 2011.

Anexos

INTERFACES DO SISTEMA



Figura 26 – login do sistema

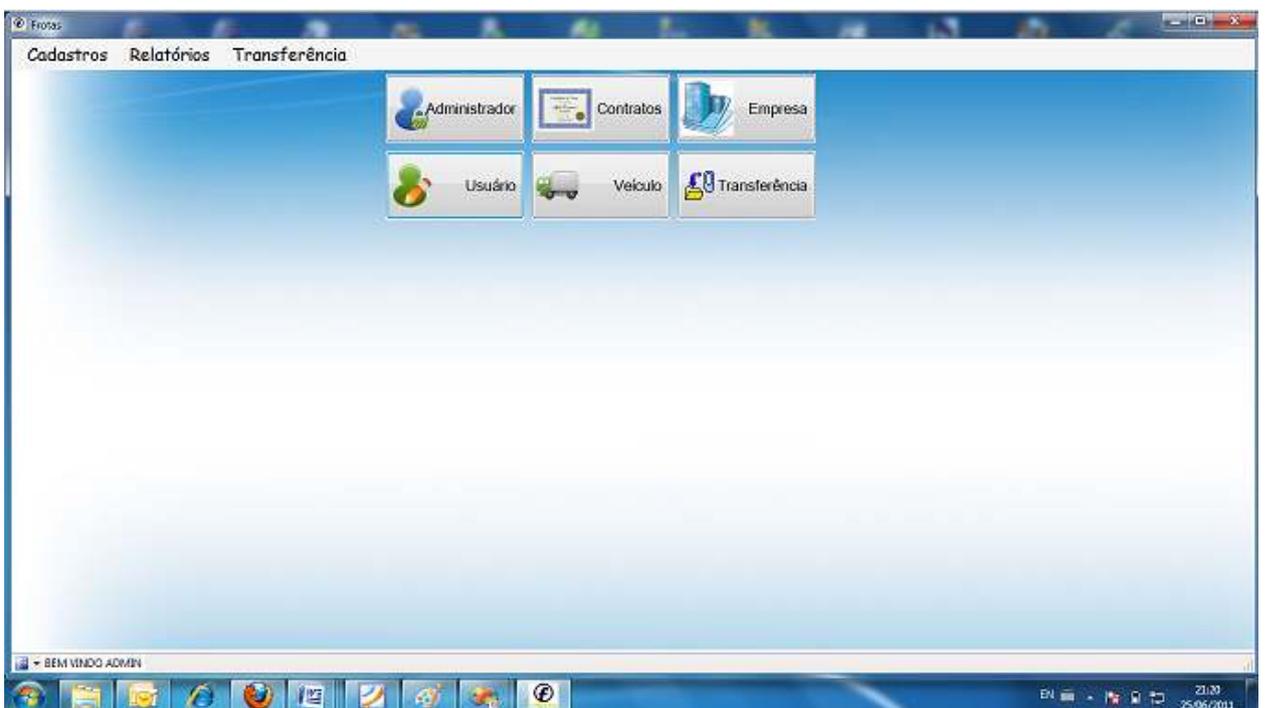


Figura 27 – Tela principal do sistema

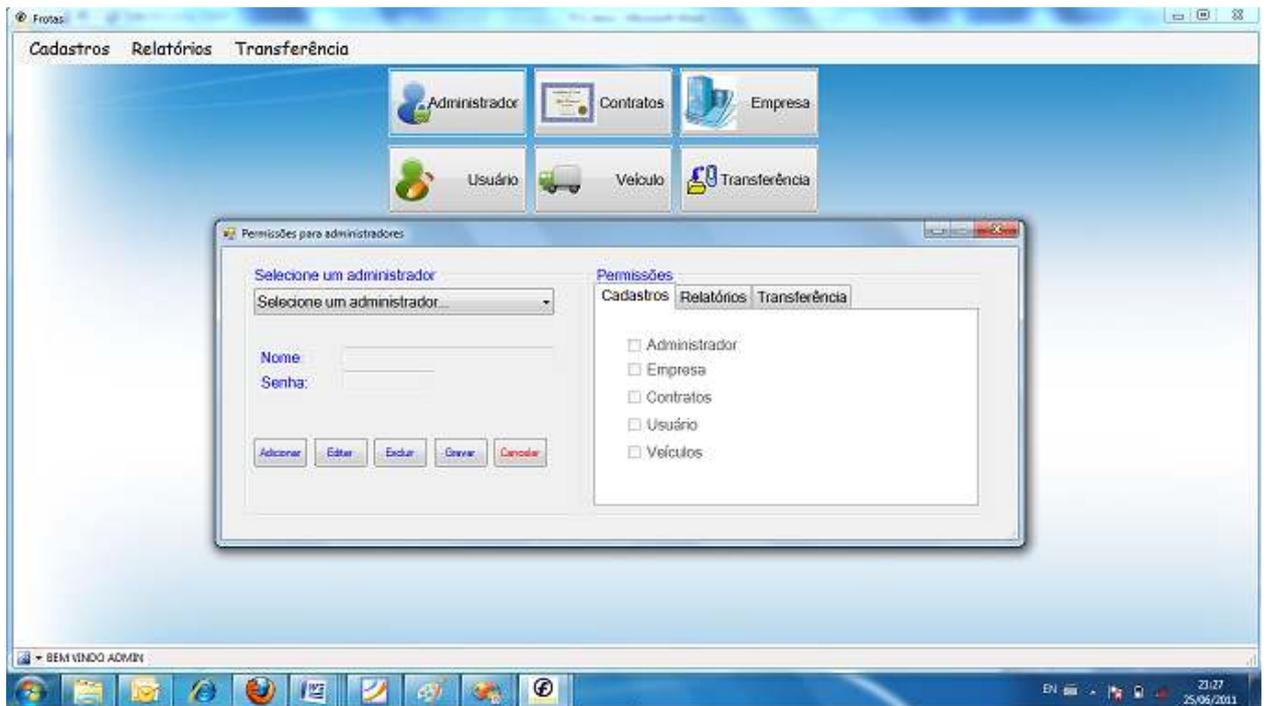


Figura 28 – Cadastro de administradores

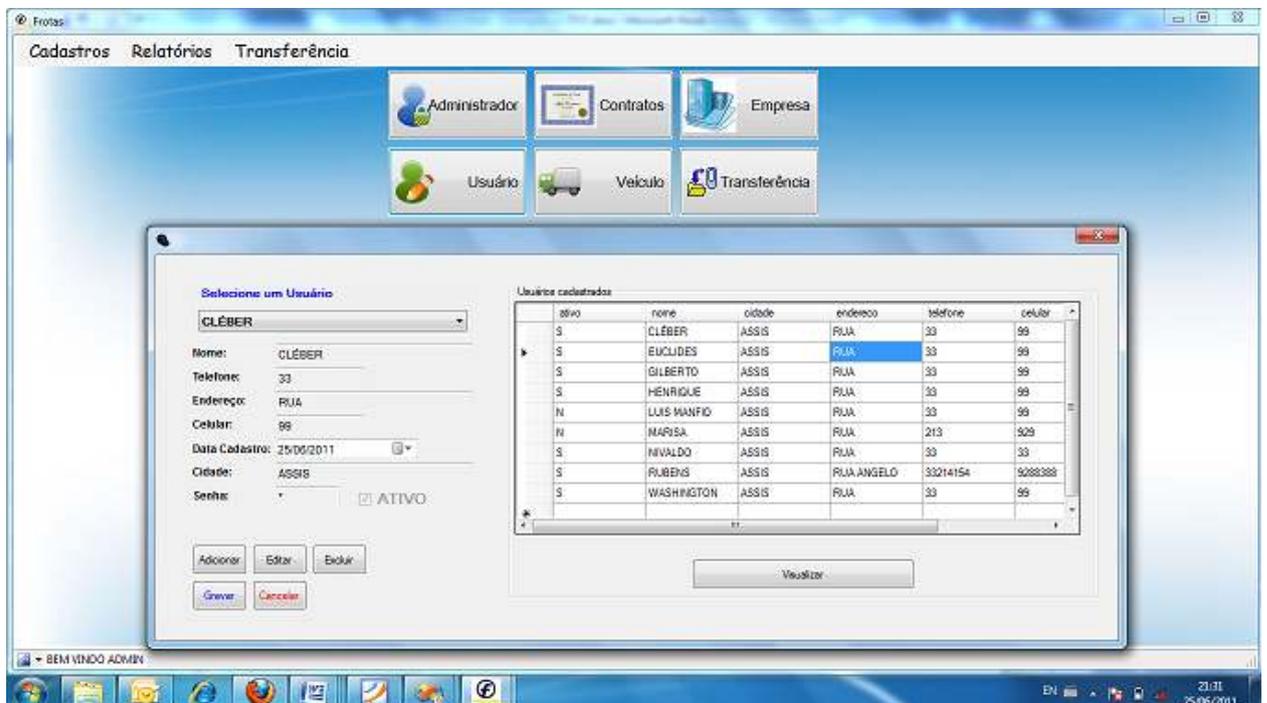


Figura 28 – Cadastro de usuários



Figura 29 – Login Mobile



Figura 30 – Abertura de ocorrência