

SOCIEDADE EDUCACIONAL DE SANTA CATARIA - SOCIESC
INSTITUTO SUPERIOR TUPY - IST
TECNOLOGIA EM SISTEMAS PARA INTERNET - TSI

GASFINDER: PROPOSTA DE UM PORTAL DE PESQUISA DE PREÇOS DE
COMBUSTÍVEIS WEB ON-LINE

ALEXANDRE DA SILVA MATEUS
WILLIAM DOS SANTOS DE OLIVEIRA

JOINVILLE
2012/01

ALEXANDRE DA SILVA MATEUS
WILLIAM DOS SANTOS DE OLIVEIRA

**GASFINDER: PROPOSTA DE UM PORTAL DE PESQUISA DE PREÇOS DE
COMBUSTÍVEIS WEB ON-LINE**

Trabalho de Conclusão de Curso submetido ao Instituto Superior Tupy como parte dos requisitos para obtenção do grau de Tecnólogo em Sistemas para Internet, sob orientação do Prof. MSc. Luiz Carlos Camargo.

PROF. MSc. LUIZ CARLOS CAMARGO
Professor Orientador

JOINVILLE
2012/01

MATEUS, Alexandre da Silva; OLIVEIRA, William dos Santos. **GasFinder: Proposta de um Portal de pesquisa de preços de combustíveis WEB online.** Joinville: SOCIESC, 2012/1.

**ALEXANDRE DA SILVA MATEUS
WILLIAM DOS SANTOS DE OLIVEIRA**

**GASFINDER: PROPOSTA DE UM PORTAL DE PESQUISA DE PREÇOS DE
COMBUSTÍVEIS WEB ON-LINE**

Este trabalho foi julgado e aprovado em sua forma final, sendo assinado pelos professores da Banca Examinadora.

Joinville, __ de _____ de 2012.

Prof. MSc. Luiz Carlos Camargo - IST (Orientador)

Prof^a. MSc. Edicarsia Barbiero Pillon - IST

Prof. Esp. Paulo Rogério Pires Manseira - IST

Dedicamos esse trabalho a todos que se aventuram
em busca de novos conhecimentos.

AGRADECIMENTOS

Aos nossos pais, esposas, professores, colegas de trabalho, amigos.

Os grandes navegadores devem
sua reputação aos temporais
e às tempestades.
(EPICURO)

RESUMO

Este trabalho trata do desenvolvimento de uma aplicação WEB com foco na criação de uma proposta, para um Portal de Pesquisa de Preços de Combustíveis WEB *on-line*, fazendo a divulgação dos postos e sua localidade para os usuários, a fim de proporcionar uma solução atraente para consulta e comparação de preços dos combustíveis. Serão utilizadas Metodologias Ágeis para a definição do projeto, Scrum e XP, em conjunto com linguagens de programação livres para o desenvolvimento da aplicação. Utilizando conceitos do modelo Scrum, este trabalho resultará em um *website* prestador de serviço, com o objetivo de atender uma necessidade que faz parte do cotidiano dos proprietários de veículos automotores: busca e comparação de preços de combustíveis. Terá como objetivo secundário proporcionar economia para os consumidores e empresas que necessitam abastecer seu veículo diariamente, independente da localidade onde estiver situado. O desenvolvimento do aplicativo contempla várias etapas, incluindo engenharia de software, levantamento de requisitos até a concepção do sistema.

Palavras-chave: Portal. Combustíveis. Posto. Scrum. Java. Python. ANP.

ABSTRACT

This work deals with the development of a web application with a focus on creating a proposal for a Research Portal Fuel Prices WEB online, making the dissemination of the posts and their location for users, in order to provide an attractive solution to query and comparison of fuel prices. Agile methodologies are used to define the project, Scrum and XP, along with free programming languages for application development. Using Scrum concepts of the model, this work will result in a website service provider, in order to meet a need that is part of everyday life of the owners of motor vehicles: search and comparison of fuel prices. Secondary objective will provide savings for consumers and businesses that need to fuel your vehicle each day, regardless of the location where you are located. The development application includes several steps, including software engineering, requirements gathering to system design.

Keywords: Portal. Fuel. Gas Station. Scrum. Java. Python. ANP.

LISTA DE ILUSTRAÇÕES

Figura 1 - Finalidades da Agência Nacional do Petróleo.....	23
Figura 2 - Sistema de Levantamento de Preços - SLP	24
Figura 3 - Tela Inicial do <i>website</i> da ANP	28
Figura 4 - Tela Inicial do SLP	28
Figura 5 - Exemplo de página com busca avançada do Yahoo	30
Figura 6 - Engenharia de Software em Camadas	33
Figura 7 - Modelo em Cascata	35
Figura 8 - Modelo em Espiral	36
Figura 9 - Modelo em Prototipagem	38
Figura 10 - Metodologia de Desenvolvimento Scrum.....	41
Figura 11 - Metodologia de Desenvolvimento XP	43
Figura 12 - Fluxo do processo com ações de WebE.....	45
Figura 13 - Processo da Engenharia de Requisitos	48
Figura 14 - Cronograma de Atividades do Projeto	51
Figura 15 - Horas de Projeto Previstas / Realizada.....	52
Figura 16 - Funcionalidades disponíveis por usuário	53
Figura 17 - Percentual de completude das funcionalidades do GasFinder	54
Figura 18 - Diagrama de Caso de Uso Geral	61
Figura 19 - Diagrama de Entidade-Relacionamento	62
Figura 20 - Diagrama de Classe.....	63
Figura 21 - Diagrama de Atividade – Pesquisar Postos	64
Figura 22 - Diagrama de Seqüência – Detalhar Postos	66
Figura 23 - Wireframe – Home	67
Figura 24 - Frameworks	69
Figura 25 - Amostra do código <i>Scraper</i> desenvolvido em Python.....	70
Figura 26 - Fluxo de informações do GasFinder	71
Figura 27 - GasFinder - Tela Inicial.....	72
Figura 28 - GasFinder - Tela de Resultado de Pesquisa	73
Figura 29 - GasFinder - Tela de Busca no Mapa	74
Figura 30 - GasFinder - Tela de Login	75

Figura 31 - GasFinder - Tela de Detalhamento do Posto	76
Figura 32 - GasFinder - Tela de Contato.....	77
Figura 33 - Diagrama de Atividade - Compartilhar em Redes Sociais	90
Figura 34 - Diagrama de Atividade - Publicar Comentários	91
Figura 35 - Diagrama de Atividade - Cadastrar Postos	92
Figura 36 - Wireframe - Home.....	93
Figura 37 - Wireframe - Contato.....	94
Figura 38 - Wireframe - Quem somos	95
Figura 39 - Wireframe - Login.....	96
Figura 40 - Wireframe - Pesquisa na lista de Postos	97
Figura 41 - Wireframe - Sem resultados para pesquisa	98
Figura 42 - Wireframe - Pesquisa diretamente no mapa	99
Figura 43 - Wireframe - Manutenção de Preços.....	100
Figura 44 - Wireframe - Informações dos Postos	101
Figura 45 - Wireframe - Erro genérico	102
Figura 46 - Código Fonte Java - Classe de Busca	103
Figura 47 - Código Fonte Java - Classe de Conexão com Banco de Dados	104
Figura 48 - Código Fonte Java - Classe de Persistência	105

LISTA DE TABELAS

Tabela 1 - Matriz de Comunicação do Projeto	51
Tabela 2 - Relação de Localidades	122

LISTA DE SIGLAS

AI Information Architecture

AJAX Asynchronous Javascript and XML

ANP Agência Nacional do Petróleo, Gás Natural e Biocombustíveis

CASE Computer-Aided Software Engineering

CSS Cascading Style Sheets

EEP Exploração e Produção de Petróleo

GNV Gás Natural Veicular

HTML HyperText Markup Language

IDE Integrated Development Environment

JSF Java Server Faces

MVC Model View Controller

OOHDM Object Oriented Hypermedia Design Method

RF Requisito Funcional

RNF Requisito Não Funcional

RSS Really Simple Syndication

SGDB Sistema de Gerenciamento de Banco de Dados

SLP Sistema de Levantamento de Preços

UI User Interface

WEBAPP Web Based Application

WEBE Web Engineering

XP Extreme Programming

SUMÁRIO

1	INTRODUÇÃO.....	17
2	AGÊNCIA NACIONAL DO PETRÓLEO (ANP).....	22
2.1	SISTEMA DE LEVANTAMENTO DE PREÇOS (SLP).....	23
2.2	USABILIDADE DA ANP E SLP.....	25
2.2.1	Navegação.....	26
2.2.2	Busca.....	29
3	ENGENHARIA DE SOFTWARE	32
3.1	MODELO EM CASCATA	34
3.1	MODELO EM ESPIRAL.....	35
3.2	MODELO DE PROTOTIPAGEM.....	37
3.3	DESENVOLVIMENTO ÁGIL.....	39
3.3.1	Scrum	40
3.3.2	Extreme Programming.....	42
3.1	ENGENHARIA WEB	44
3.1.1	Atividades da Engenharia WEB	45
3.1.2	Padrões de Projeto de Hipermídia	46
3.1.3	OOHDM.....	46
3.1.4	Protótipo da Interface	Erro! Indicador não definido.
3.2	ENGENHARIA DE REQUISITOS	48
3.3	UML	49
4	DESENVOLVIMENTO DO GASFINDER	50
4.1	GESTÃO E METODOLOGIA DE DESENVOLVIMENTO	50
4.1.1	Horas Prevista / Realiza	52
4.1.2	Atividades previstas e não realizadas	53
4.2	ESCOPO DO APLICATIVO GASFINDER	55
4.2.1	Descrição Geral.....	55
4.2.2	Funções do Aplicativo	55
4.2.3	Características dos Usuários.....	56

4.2.4	Restrições Gerais	56
4.2.5	Suposições e Dependências	56
4.2.6	Requisitos	57
4.3	MODELAGEM.....	60
4.3.1	Diagrama de Casos de Uso	60
4.3.2	Diagrama de Entidade-Relacionamento	62
4.3.3	Diagrama de Classe	63
4.3.4	Diagrama de Atividade	64
4.3.5	Diagrama de Seqüência	65
4.3.6	Wireframe	67
4.4	FERRAMENTAS DE DESENVOLVIMENTO	68
4.4.1	Extração de Dados	70
4.5	AMOSTRA DO APLICATIVO GASFINDER	71
4.5.1	Tela Inicial.....	71
4.5.2	Tela de Resultado de Pesquisa.....	73
4.5.3	Tela de Busca no Mapa.....	74
4.5.4	Tela de Login	75
4.5.5	Tela de Detalhamento do Posto	76
4.5.6	Tela de Contato	77
4.6	CONSIDERAÇÕES FINAIS DO TRABALHO	78
5	CONCLUSÃO.....	80
	REFERÊNCIAS.....	82
	APÊNDICE.....	89
	APÊNDICE 1 - DIAGRAMA DE ATIVIDADE - COMPARTILHAR EM REDES SOCIAIS.....	90
	APÊNDICE 2 - DIAGRAMA DE ATIVIDADE - PUBLICAR COMENTÁRIOS	91
	APÊNDICE 3 - DIAGRAMA DE ATIVIDADE - CADASTRAR POSTOS.....	92
	APÊNDICE 4 - WIREFRAME - HOME	93
	APÊNDICE 5 - WIREFRAME - CONTATO	94

APÊNDICE 6 - WIREFRAME - QUEM SOMOS.....	95
APÊNDICE 7 - WIREFRAME - LOGIN.....	96
APÊNDICE 8 - WIREFRAME - PESQUISA NA LISTA DE PRODUTOS	97
APÊNDICE 9 - WIREFRAME - SEM RESULTADOS PARA PESQUISA.....	98
APÊNDICE 10 - WIREFRAME - PESQUISA DIRETAMENTE NO MAPA	99
APÊNDICE 11 - WIREFRAME - MANUTENÇÃO DE PREÇOS	100
APÊNDICE 12 - WIREFRAME - INFORMAÇÕES DOS POSTOS.....	101
APÊNDICE 13 - WIREFRAME - ERRO GENÉRICO	102
APÊNDICE 14 - CÓDIGO FONTE JAVA - CLASSE DE BUSCA.....	103
APÊNDICE 15 - CÓDIGO FONTE JAVA - CLASSE DE CONEXÃO COM BANCO DE DADOS	104
APÊNDICE 16 - CÓDIGO FONTE JAVA - CLASSE DE PERSISTÊNCIA	105
APÊNDICE 17 – RELAÇÃO DE LOCALIDADES.....	106

1 INTRODUÇÃO

Geralmente os usuários, sejam eles empresas ou pessoa física, são atraídos por produtos e serviços que apresentem a melhor relação custo-benefício. Para metade dos brasileiros, a relação custo-benefício é considerada o fator mais importante no processo de compra, enquanto a maioria dos europeus, 81% ponderam o custo-benefício como fator decisivo. (BRIDGE RESEARCH; VOLTAGE, 2010).

Os usuários tendem a selecionar espontaneamente, aquelas opções que oferecem “mais por menos” (leia-se: mais vantagens com menor custo). Para tanto, entram em ação diversos mecanismos para proporcionar o melhor investimento: consulta e comparação de preços, avaliação da qualidade, realização de orçamento, barganha, entre outros.

O *buzzmarketing*, popularmente chamado de “comunicação boca a boca”, é uma prática freqüentemente utilizada no cotidiano dos usuários, onde a opinião sobre produtos e serviços é transmitida entre um grupo de usuários, possibilitando que compradores em potencial analisem antecipadamente o custo-benefício antes de novas aquisições. (CHAVES, 2007, p. 10 - 11).

A Internet tornou-se um grande influenciador no processo de compra, onde com pouco esforço, o usuário pode fazer *buzzmarketing*, pesquisar sobre a qualidade de produtos e principalmente, comparar preços. Através de mecanismos de busca, como o Google, facilitou-se a comparação de preço de produtos em geral. 80% dos usuários que compram via Internet, costumam comparar preços antes de adquirir produtos *online*. Cerca de 43% destes consumidores do comércio eletrônico, também recorrem a Internet antes de comprar produtos *offline*, ou seja, diretamente no varejo. (IBOPE, 2010).

Paradoxalmente, quando o assunto é custo fixo (aluguel, seguro e impostos, por exemplo), a busca pelo melhor custo-benefício é deixada de lado. O combustível utilizado diariamente nos automóveis tornou-se mais um dentre os inúmeros custos fixo que uma pessoa ou empresa possa ter. Acredita-se que o consumidor final¹, raramente busque o melhor preço ou economia no momento de abastecer.

¹ Neste trabalho será convencionado o uso do termo “consumidor final” a fim de indicar empresas ou pessoa física, que participam como clientes no comércio de combustível (Gasolina, Etanol, Diesel ou GNV) revendidos em postos de combustível.

Entende-se que isso ocorra pela falta de tempo, somado pela ausência de ferramentas adequadas, ou ainda pelo pleno desconhecimento ou desinteresse sobre a economia que é possível realizar nesta atividade cotidiana. O Procom (Órgão de Proteção e Defesa do Consumidor) orienta que o consumidor final compare os preços dos combustíveis antes de abastecer, pois segundo o órgão, muitos postos realizam promoções relâmpago, e reduzem o preço do combustível, o que pode ser oportuno para quem busca economizar. (PROCOM, 2011).

Também existe atualmente na WEB, inúmeros *websites*, que oferecem ferramentas onde o consumidor final pode consultar os preços dos combustíveis praticados pelos postos em determinadas cidades. Há inclusive, softwares móveis (para celular e *tablet*) para esta finalidade. Entretanto, há somente uma única fonte oficial dos preços praticados pelos postos, que é coletada pela Agência Nacional do Petróleo (ANP), órgão responsável pela regulamentação e fiscalização do comércio de combustível no Brasil. A ANP também disponibiliza os preços dos combustíveis em seu *website*, www.anp.gov.br/preco, através de um serviço chamado de Sistema de Levantamento de Preços (SLP), sendo este serviço, um dos objetos de estudo deste trabalho.

O objetivo geral deste trabalho, é o desenvolvimento de um aplicativo² chamado GasFinder, que ofereça, como serviço, a comparação dos preços dos combustíveis *online*, tornando-se uma opção de consulta alternativa para o consumidor final, além do serviço oferecido no *website* da ANP. Para realizar este trabalho, será adotado ferramentas para desenvolvimento de software para WEB, incluindo Java, Python, MySQL e Eclipse; ScrapperWiki, Spring, Hibernate, JSF e Primefaces como *frameworks* auxiliares; tecnologias AJAX, HTML, CSS e JavaScript para camada de interface de usuário; conceitos ágeis para o gestão de projeto, Scrum e XP. Será usado como fonte oficial dos preços dos combustíveis, os dados fornecidos pela ANP.

² O uso do termo “aplicativo” neste trabalho, está direcionado para representar, primeiramente, o GasFinder - Portal de Pesquisa e Comparação de Preços de Combustível WEB *Online*. Este termo também será apresentado por autores em citações diretas, como sinônimo de “software”, “sistema” ou “sistema WEB”.

Os objetivos específicos deste trabalho incluem a abordagem da Agência Nacional do Petróleo (ANP), onde será abordado a história e atribuições do órgão regulador da indústria do petróleo, que fiscaliza os postos de combustíveis e mantém uma base de dados com os preços dos combustíveis. Será analisada a usabilidade e limitações do Sistema de Levantamento de Preços (SLP), através do qual a ANP disponibiliza as informações de preços de combustíveis ao consumidor final.

Também é objetivo específico deste trabalho, contextualizar a Engenharia de Software e seus elementos considerados fundamentais para obtenção de softwares confiáveis. São explorados três modelos de desenvolvimento de software: Modelo em Cascata, Modelo em Espiral e Modelo de Prototipagem. Estes modelos oferecem diferentes alternativas para garantir o progresso no desenvolvimento de software, onde cada modelo propõe um conjunto de atividades seqüenciais específicas.

Não somente, é objetivo específico deste trabalho, abranger a filosofia acerca do Desenvolvimento Ágil, através de duas metodologias de desenvolvimento: Scrum e XP. Cada metodologia oferece um conjunto de diretrizes que visam, entre outros, minimizar as falhas no desenvolvimento de software através de entregas incrementais do software ao cliente.

Os objetivos específicos deste trabalho também compreendem uma abordagem da Engenharia WEB, uma especialização da Engenharia de Software convencional que oferece um conjunto de atividades diferenciadas a fim de tratar melhor, as particularidades existentes no desenvolvimento de aplicações para WEB. Abrange inclusive o método de projeto WEB mais discutido segundo as referências deste trabalho, chamado OOHDM (*Object Oriented Hypermedia Design Method*), que propõe atividades de projeto específicas para aplicações WEB. Também é feita uma abordagem acerca de Protótipo de Interface, utilizado neste trabalho, para criação de leiaute de interface do usuário (*wireframes*), visando obter uma visão antecipada dos requisitos do aplicativo.

A Engenharia de Requisitos também está contida nos objetivos específicos deste trabalho, a fim de expor as etapas relacionadas a obtenção de requisitos, especificação, detalhamento e validação dos requisitos da aplicação. Por sua vez, a UML (*Unified Modeling Language*) se fará necessário explorar neste trabalho, como modelo de linguagem visual para representação das perspectivas do aplicativo GasFinder.

A etapa de Desenvolvimento do GasFinder, completa os objetivos específicos deste trabalho, que irá demonstrar desde a fase de especificação até a fase do desenvolvimento funcional da aplicação. Será aplicado na gestão e metodologia de desenvolvimento, algumas boas práticas do Scrum e XP, conforme se faz necessário, ao decorrer do desenvolvimento da aplicação.

Dando continuidade a fase de especificação, haverá a definição do escopo e definição das características da aplicação. Tem-se a descrição geral do GasFinder, como sendo uma aplicação que tem como objetivo, permitir a busca e comparação de preços de combustíveis. A função do GasFinder será, a capacidade de efetuar a busca dos postos e preços dos combustíveis em uma determinada localidade³. Por sua vez, tem-se a definição dos tipos de usuários da aplicação, que terá três perfis distintos: Visitante, Usuário Cadastrado e Administrador.

Após o levantamento dos Requisitos Funcionais (RF) e Requisitos Não-Funcionais (RNF), tem-se a fase de modelagem através de UML, onde será usado o Diagrama de Casos de Uso, Diagrama de Entidade-Relacionamento (DER), Diagrama de Classe, Diagrama de Seqüência e Diagrama de Atividade. Outra sessão específica terá como objetivo expor as ferramentas de desenvolvimento utilizadas neste trabalho. Em seguida, será abordada a fase de extração de dados do *website* da ANP, onde será acessado o Sistema de Levantamento de Preços (SLP) a fim de obter informações sobre os postos e os preços dos combustíveis, para posteriormente, ser disponibilizado ao consumidor final na interface do GasFinder.

³ O termo “localidade” foi reservado neste trabalho, segundo o termo empregado pela ANP (2012, p. 1), para representar a área geográfica onde estão localizados os postos de combustíveis, em sua maior parte, delimitada por municípios.

No fim da etapa de Desenvolvimento, tem-se a amostra de telas da aplicação, com objetivo de demonstrar o resultado do desenvolvimento do GasFinder, onde será descrito características como, as funcionalidades, a navegação e a interação da Tela Inicial, Tela de Resultado de Pesquisa, Tela de Busca no Mapa, Tela de Login, Tela de Detalhamento do Posto e Tela de Contato.

Após a etapa de Desenvolvimento, será exposto as considerações finais do trabalho, a fim de situar o nível de completude dos objetivos propostos neste trabalho, observando-se as principais atividades que contribuíram para o desenvolvimento do GasFinder. Haverá também, sugestão de novas funcionalidades para quem interessar basear-se neste trabalho, para desenvolver novos softwares.

A conclusão deste trabalho irá expor as considerações e resultados obtidos acerca do desenvolvimento do GasFinder, a importância do uso de Metodologias Ágeis, Scrum e XP, que criam condições para um processo de desenvolvimento mais dinâmico e flexível, dificuldades encontradas no desenvolvimento da aplicação e no uso das tecnologias e, sugestão de novas funcionalidades para aplicação.

A sessão Apêndices terá como objetivo agrupar informações complementares, que é considerada relevante para melhor compreensão deste trabalho, com destaque especial para expor diagramas, amostra de código fonte do aplicativo e *wireframe* que será utilizado na etapa de desenvolvimento do GasFinder.

Este trabalho está organizado na apresentação seqüencial dos seguintes assuntos:

- ANP: Abordagem da Agência Nacional do Petróleo (ANP) e do Sistema de Levantamento de Preços (SLP);
- Engenharia de Software: Compreende o planejamento do software;
- Desenvolvimento do aplicativo GasFinder: Definição de requisitos, modelagem e amostra de telas do aplicativo;
- Considerações Finais: Aborda o nível de completude dos objetivos atingidos neste trabalho.
- Conclusão: Lições aprendidas neste trabalho, dificuldades encontradas e sugestão de novas funcionalidades para aplicação.
- Apêndice: Agrupa informações complementares deste trabalho.

2 AGÊNCIA NACIONAL DO PETRÓLEO (ANP)

A Agência Nacional do Petróleo (ANP) é o órgão regulador das atividades que integram a indústria do petróleo no Brasil, cuja finalidade segundo o Planalto (1997), é “promover a regulação, a contratação e a fiscalização das atividades econômicas integrantes da indústria do petróleo, do gás natural e dos biocombustíveis”.

ANP foi criada em 1997, quando o presidente Fernando Henrique Cardoso aprovou a Lei do Petróleo, o que anulou o monopólio da Petrobras, que por quase meia década (desde 1953 até 1997), foi a única empresa que detinha autorização para explorar, produzir e refinar petróleo no Brasil. Desde então, a Petrobras concorre com outras empresas nacionais do mesmo ramo e inclusive, com empresas estrangeiras.

O processo de flexibilização do monopólio das atividades da indústria petrolífera, introduzido através da Emenda Constitucional nº 9/95, colocou fim a mais de 40 anos do monopólio estatal exercido pela Petrobras. A Lei nº 9.478/97 (Lei do Petróleo), que complementou a Emenda Constitucional nº 9/95, tinha por objetivo o estímulo a concorrência, a atração de investimentos e a regulamentação das participações governamentais sobre a exploração e produção de petróleo e gás natural. No âmbito desta lei, foi criada a Agência Nacional do Petróleo (ANP), responsável pelo exercício da regulação das atividades do setor. (FERNANDES, 2007, p. 9).

Com o mercado aberto e regulado pela ANP, o Brasil soma desde 2007 até os dias de hoje, a atuação de 74 grandes empresas de EeP (Exploração e Produção de Petróleo), sendo 38 brasileiras, 36 estrangeiras. Também contabiliza o início das atividades de cerca de 30 pequenas e médias empresas de EeP. (LIMA, 2011, p. 10)

A criação da ANP permitiu estabelecer novas regras que viabilizaram a criação de um mercado mais competitivo e que trouxeram mais vantagens para o país e para o consumidor final. A fiscalização da ANP é responsável por realizar uma constante auditoria da qualidade dos combustíveis comercializados pelos postos e também pela coleta dos preços dos combustíveis praticados no país, que posteriormente são disponibilizados aos consumidores para consulta *online*, através do Sistema de Levantamento de Preços (SLP), disponível no *website* da ANP.

Para o país, os benefícios da atuação da APN podem ser resumidos em uma maior arrecadação fiscal e também a redução da importação de petróleo, segundo Júnior (2008, p. 97) e Biazotto (2004, p. 13). Na Figura 1 é demonstrado as três principais finalidades da ANP.

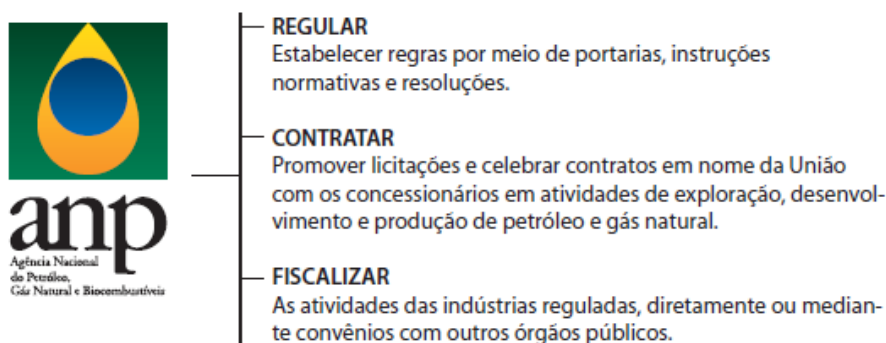


Figura 1 - Finalidades da Agência Nacional do Petróleo

Fonte: (ANP, 2011, p. 2)

A seguir será abordado em maior profundidade, o Sistema de Levantamento de Preços (SLP), um serviço utilidade pública da ANP, porém, com limitações que podem prejudicar o interesse de uso pelos consumidores finais, como uma opção de pesquisa de preços de combustíveis *online*.

2.1 SISTEMA DE LEVANTAMENTO DE PREÇOS (SLP)

Especialmente para o consumidor final, a ANP oferece em seu *website*, acesso ao Sistema de Levantamento de Preços (SLP), através do endereço www.anp.gov.br/preco. Trata-se de um serviço de consulta de preços de combustíveis *online*, atualizado uma vez por semana, onde é possível verificar os valores praticados pelos postos em 555 localidades⁴, especificados no Apêndices deste trabalho, correspondendo a 10% do território nacional. (ANP, 2010, p. 1 - 15).

O SLP permite que o consumidor final realize pesquisas por estados e por cidades, ou escolha entre três tipos de resumos: Semanal, Mensal ou o resumo das Quatro últimas semanas. Como resultado da pesquisa ou seleção de resumos, será listado os postos de combustíveis para a cidade selecionada, e os preços dos combustíveis praticados por cada respectivo posto.

⁴ Apenas 538 localidades será importada pelo GasFinder. Mais detalhes na sessão Apêndices.

Também será exibido outras informações complementares sobre os postos, como: endereço, bairro, preço de compra do combustível, modalidade de compra, fornecedor e data em que foi realizada a coleta do preço do combustível pela ANP. Também é exibido a média do preço do combustível, onde este valor é previamente calculado, baseando-se nos resultados de pesquisa, e exibido em conjunto com outros cálculos, como valor mínimo e o valor máximo do combustível praticado pelos postos que compõem os resultados da pesquisa.

Ainda na tela de resultados da pesquisa, o SLP irá exibir uma lista dos postos de combustíveis que não apresentaram nota fiscal de compra no ato da coleta dos preços pela ANP, o que indica cautela para consumidor final ao abastecer nestes postos, pois, segundo o Planalto (1990), a sonegação fiscal é caracterizada como crime contra ordem tributária, sob pena de reclusão de dois a cinco anos, e multa. No estado de Santa Catarina, em Julho de 2009, dos cerca de 1830 postos depurados pelos servidores da Secretaria do Estado da Fazenda, cerca de 200 apresentam irregularidade fiscal, segundo Sefaz (2009).

Na figura 2, é demonstrado os resultados de uma pesquisa realizada no SLP, havendo-se selecionado previamente a cidade de Joinville - Santa Catarina.



anp
Agência Nacional
do Petróleo,
Gás Natural e Biocombustíveis

CSA | SLP
SISTEMA DE LEVANTAMENTO DE PREÇOS

Você está em » Home » Por Município » Posto por Município »

Síntese dos Preços Praticados - JOINVILLE
Resumo I - Gasolina R\$/l
Período : De 22/01/2012 a 28/01/2012

Relação de Postos pesquisados Postos sem apresentar Nota Fiscal de Compra

DADOS MUNICÍPIO								
RAZÃO SOCIAL	ENDEREÇO	BAIRRO	BANDEIRA	PREÇO VENDA	PREÇO COMPRA	MODALIDADE DE COMPRA	FORNECEDOR (B. BRANCA)	DATA COLETA
Auto Posto Jc Ltda.	Avenida Santa Catarina, 3896	Santa Catarina	BRANCA	2,610	2,350	CIF	REJAILE	26/01/2012
Auto Posto Jc Ltda.	Rua Anita Garibaldi, 2001	Anita Garibaldi	BRANCA	2,610	2,207	CIF	ROYAL PETRO	26/01/2012
Posto Z10 Ltda.	Rua Minas Gerais, 1330	Nova Brasília	PETROBRAS DISTRIBUIDORA S.A	2,707	-	-	-	26/01/2012
Auto Posto Bucarein Ltda	Rua Anita Garibaldi, 1099	Anita Garibaldi	PETROBRAS DISTRIBUIDORA S.A	2,710	-	-	-	26/01/2012

Figura 2 - Sistema de Levantamento de Preços - SLP

Fonte: (http://www.anp.gov.br/preco/prc/Resumo_Por_Municipio_Posto.asp)

2.2 USABILIDADE DA ANP E SLP

Segundo Shackel (1993 apud MEMÓRIA, 2005, p. 6), “usabilidade é a capacidade, em termos funcionais humanos, de um sistema ser usado facilmente e com eficiência pelo usuário”. Para Krug (2006, p. 3), o princípio da usabilidade é manifestado pelo usuário na seguinte expressão: “Não me faça pensar”. Isso significa que, tanto quanto for humanamente possível, um software deve ser por si só auto-explicativo.

A usabilidade está tradicionalmente associada a importantes atributos, que não devem-se considerar como propriedades unicamente exclusivas de uma interface de usuário (UI, *User Interface*). Segundo Nielsen (1993 apud MEMÓRIA, 2005, p. 6 - 7) e Nielsen (2000, 2002, 2006, apud Ferreira, 2008, p. 20), apontam-se estes cinco atributos em uma interface de usuário:

- Ser fácil de aprender: o sistema deve ser intuitivo de tal modo, que um novo usuário consiga trabalhar rapidamente, sem muito esforço no aprendizado;
- Ser eficiente na utilização: o sistema deve ser eficiente e funcionar rapidamente, para acompanhar a velocidade de interação e uso de recursos simultâneos utilizados por usuários mais experientes;
- Ser fácil de ser recordado: o sistema deve ser simples, de tal modo, que o usuário possa parar de utilizar o sistema por um período, e voltar a utilizá-lo sem ser necessário reaprender tudo novamente.
- Ter poucos erros: a ocorrência de erros deve ser a menor possível, sendo que erros graves devem ser tratados, devem-se evitar aqueles erros em que usuário perde todo o processamento que estava realizando e, quando ocorrer, o sistema deve recuperar os dados para o usuário.
- Ser subjetivamente agradável: o sistema deve ser agradável de ser utilizado, deve proporcionar que os usuários fiquem satisfeitos com sua utilização.

Durante o uso do SLP pela equipe deste trabalho (Alexandre e William), constataram-se limitações acerca da usabilidade, envolvendo a navegação, e a ausência de recursos de busca, conforme será descrito nos sub-tópicos seguintes.

2.2.1 Navegação

Reconhece-se um bom projeto de navegação, somente quando o usuário consegue responder a três principais questões, segundo Nielsen (2000, p. 188 - 197), Pressman (2010, p. 335) e Memória (2005, p. 52):

- Onde estou?
- Onde estive?
- Onde posso ir?

Com estas questões em mente, a equipe deste trabalho (Alexandre e William) fará um experimento sobre a navegação do SLP: Colocou-se na perspectiva simbólica de um consumidor final, que inicialmente desconhece a existência do Sistema de Levantamento de Preços (SLP) e, esta em busca de informações sobre os preços de combustíveis praticados em uma cidade qualquer.

Ao se acessar a página inicial do *website* da ANP (www.anp.gov.br), a equipe deste trabalho deparou-se com a ausência de qualquer informação que sinalizasse que a ANP desempenha o serviço de coleta de preços de combustível e, tão pouco, que estas informações estariam sendo disponibilizadas pela ANP aos consumidores finais.

Para continuar o experimento, investiu-se no garimpo e leitura detalhada de todas as informações na tela, incluindo texto, imagens, menus. Selecionaram-se vários links aleatoriamente que levaram até outras telas, porém sem sucesso na busca por informações sobre os preços dos combustíveis. Segundo Krug (2006, p. 11), ao estar na condição de usuário, “assim que encontramos um link que pareça poder levar ao que estamos procurando, há uma boa chance de que cliquemos no mesmo.”

Em certo momento, retornou-se para tela inicial da ANP e, após rolar a tela para baixo e acessar uma caixa de seleção no canto inferior direito, verificou-se a existência de opção intitulada “Levantamento de Preço”, conforme destacado na figura 3, onde se supôs, tendo-se sido esgotado as possibilidades mais óbvias, que aquela poderia ser a opção correta que levaria até as informações sobre os preços dos combustíveis.

Finalmente, depois de certo esforço, concluiu-se com sucesso o experimento, ao se ter obtido acesso a Tela Inicial do Sistema de Levantamento de Preços (SLP), por sua vez, demonstrado na figura 4.

Porém, constatou-se ao carregar o SLP em tela, o uso inadequado de janelas adicionais (*popups*). Segundo Ferreira (2008, p. 48), não é indicado o uso de janelas adicionais, pois “esta pratica, além de cobrir a janela em que o usuário efetivamente está trabalhando, faz também com que o botão ‘Voltar’ seja desativado, uma vez que novas janelas não herdam o histórico de navegação.” Pode-se destacar como agravante, os recursos de bloqueio automático da abertura de janelas adicionais disponível em softwares que permitem a navegação na Internet. Estes recursos atuam sem intervenção do usuário, ou seja, não se emite qualquer aviso e, quando se ocorre uma situação de bloqueio da abertura de uma janela, os usuários menos experientes podem concluir que houve um erro no *website*, pois o mesmo não será capaz de exibir o conteúdo. O bloqueador de janelas está disponível em softwares que permitem a navegação na Internet, como o Firefox. MONZILLA (2012).

Como resultado deste experimento, pode-se afirmar que houve dificuldade em encontrar rapidamente a opção desejada, sendo que a opção de acesso ao SLP estava posicionada em um local pouco intuitivo no *website* da ANP, o que prejudicou o reconhecimento desta opção. Segundo Krug (2006, p. 7), “se você frustrar os usuários, eles irão para outro lugar”. Certamente, se não houvesse um grande interesse na busca da informação, objetivo do experimento, certamente um usuário real, na condição de consumidor final, teria desistido e abandonado sua busca e tentado outras fontes alternativas, acessando outros *websites* ou ferramentas de busca *online*.

Deve-se haver sempre um destaque maior para as informações relevantes, inclusive sobre o SLP neste contexto de uso, pois segundo Pressman (2010, p. 340) e Ferreira (2008, p. 46), o usuário não deve ter que ficar explorando as opções na tela, até encontrar aquilo que precisa.

Não somente, as telas precisam concentrar-se na fronteira entre exibir informações relevantes, ou excesso de informações que podem invariavelmente deixar o usuário ainda mais confuso. “Os usuários tem tolerância variadas a complexidade e distrações. Algumas pessoas não tem problemas com páginas cheias e confusas, mas outras sim”, segundo Krug (2006, p. 19).

ANP
Agência Nacional do Petróleo, Gás Natural e Biocombustíveis

Página Principal

- Conheça a ANP
- Legislação
- Petróleo e Derivados
- Gás Natural
- Biocombustíveis
- Qualidade
- Fiscalização
- Defesa da Concorrência e Preços
- Rodadas de Licitações
- Participações Governamentais e de Terceiros

Notícias

- Notas à Imprensa - Fevereiro 2012**
Produção de petróleo e gás em 2011 foi a maior da história do Brasil
- Notas à Imprensa - Fevereiro 2012**
Nota conjunta à imprensa - Grupo de acompanhamento e avaliação
Nota sobre o acompanhamento do vazamento de óleo na Bacia de Santos....
- Mais Notícias**
 - Técnicos da ANP farão vistoria em plataforma na Bacia de Santos a partir de amanhã (1/2)
 - ANP investiga vazamento de óleo no Rio Grande do Sul
 - ANP autua dois postos da grande SP por desrespeito a determinação de vender diesel com menor teor de enxofre (S50)
 - ANP submete a consulta pública proposta de aprimoramento da qualidade do biodiesel
 - Produção de petróleo em novembro de 2011 foi recorde

Destaques

- Estudos de Geologia e Geofísica**
Nova regulamentação para aquisição de dados sísmicos
Resoluções aperfeiçoam procedimentos operacionais
- Óleo diesel S-50 e S-10**
Conheça as regras de implantação do Diesel S-50
Óleo diesel S-50 já está disponível para o consumidor em todo território nacional...

Avisos

- Etanol**
Sistema para envio de dados semanais
Comunicado a respeito da adequação à Resolução ANP 66/2011, por parte dos fornecedores de etanol e distribuidores....

Busca Rápida

- Escolha uma opção
- Escolha uma opção
- Cartilha do Posto Revendedor de Combustíveis
- Informe Conjuntura & Informação
- Revenda de GLP
- Notas Técnicas ANP
- Piano Plurianual de Geologia & Geofísica
- Registro de produtos
- Relatório de Gestão
- Reuniões de Diretoria
- Revenda de GLP
- Revenda de combustíveis automotivos
- Sistema de consulta de postos revendedores

Transparência Pública

CSA | CENTRAL DE SERVIÇOS ANP

Conheça a cartilha **GÁS DE BOTIJÃO (GLP)**

10 ORIENTAÇÕES PARA GARANTIR SEUS DIREITOS E SUA SEGURANÇA

Agência Nacional do Petróleo, Gás Natural e Biocombustíveis - DF / RJ / SP / BA / AM
Fale Conosco de 2ª a 6ª, de 8h às 20h - Ligação Gratuita: 0800-970-0267 ou pela Internet

Figura 3 - Tela Inicial do website da ANP

Fonte: (<http://www.anp.gov.br>)

anp
Agência Nacional do Petróleo, Gás Natural e Biocombustíveis

CSA | SLP
SISTEMA DE LEVANTAMENTO DE PREÇOS

Você está em » Home »

Resumos

SEMANAL - RESUMO I

MENSAL - RESUMO II

4 ÚLTIMAS SEMANAS - RESUMO

III

Pesquisas

POR ESTADO

POR MUNICÍPIO

ÚLTIMAS COLETAS

Figura 4 - Tela Inicial do SLP

Fonte: (<http://www.anp.gov.br/preco/>)

2.2.2 Busca

O recurso de busca é reconhecidamente, um dos mais importantes elementos de interface de um *website*. Assim como o logotipo do *website*, o recurso de busca situa-se entre os primeiros elementos que os usuários tentam identificar em um *website*. (MEMÓRIA, 2005, p. 60 - 62), Isto ocorre, pois pouco mais da metade dos usuários preferem navegar por busca, segundo Nielsen (2000, p. 224 – 225).

Pode-se classificar a navegação dos usuários, da seguinte forma, segundo Krug (2006, p. 32):

- Usuários dominados pelas pesquisas: geralmente procuram as caixas de busca assim que acessam em um *website*;
- Usuários dominados por links: geralmente, utilizam a navegação por links primeiro, deixando a busca, para ser usada somente quando se esgotar as opções de links mais óbvias levariam ao conteúdo desejado, ou quando já estiverem frustrados por não localizar o que desejam.

Ainda segundo Krug (2006, p. 3), “nada importante deve estar a mais de dois cliques de distancia”. Neste aspecto, o recurso de busca torna-se uma opção eficaz para oferecer ao usuário, acesso rápido e direto ao conteúdo desejado, sobretudo para *websites* considerados grandes, “aquelas com mais de 200 páginas”, segundo Memória (2005, p. 60).

Nota-se a ausência de uma ferramenta de busca na Tela Inicial do SLP (ver figura 4), que ofereça resultados relevantes e acessíveis em poucos cliques. Caso houve-se, poderia ser utilizada para permitir, por exemplo, que o usuário informa-se uma cidade para, obter como resultado, uma lista de postos de combustíveis da respectiva cidade, junto com os preços dos combustíveis. Também é importante haver uma opção de busca avançada, onde pode-se oferecer diferentes filtros e formas de busca, segundo Memória (2005. p. 62), conforme exemplo demonstrado na figura 5.

O SLP oferece somente a navegação por links, com resumos de informações previamente estabelecidos, questionáveis, pois podem não refletir substancialmente uma condição de pesquisa mais freqüentemente utilizada por consumidores finais, com resumo Semanal, Mensal ou resumo das Quatro últimas semanas, que filtram os resultados dos preços dos combustíveis baseados nestas seleções.

Por fim, conclui-se neste trabalho, que a ANP oferece um serviço *online* parcialmente adaptado para o público-alvo relativo ao consumidor final. O SLP possui insuficiência de importantes funcionalidades, como recursos de busca, uma navegação mais flexível, além de outros recursos de interações comuns atualmente na WEB, úteis para distribuição eficaz de conteúdo *online*, como integração com redes sociais para compartilhamento de informações, ou mesmo RSS (*Really Simple Syndication*) para distribuição de conteúdo.

Todas as limitações citadas criam uma experiência de navegação pouco intuitiva para o consumidor final. Prejudicado por estas restrições, o consumidor final pode invariavelmente sentir-se desmotivado a continuar utilizando o Sistema de Levantamento de Preços da ANP.

YAHOO! BRASIL

Vá lá? **BUSCAR NA WEB**

[Yahoo!](#) - [Yahoo! Search](#) - [Ajuda](#)

Busca avançada na web

Você pode usar as opções desta página para tornar sua busca mais específica. Preencha apenas os campos necessários para sua pesquisa. **Busca avançada**

Mostrar resultados com

todas estas palavras em qualquer lugar da página ▼

a expressão exata em qualquer lugar da página ▼

qualquer uma destas palavras em qualquer lugar da página ▼

nenhuma destas palavras em qualquer lugar da página ▼

Dica: Utilize estas opções para procurar uma expressão exata ou excluir páginas que contenham palavras específicas. Você pode também limitar sua busca a partes específicas da página.

Site/Domínio

☒ Qualquer domínio

☐ Apenas domínios .com.br ☐ Apenas domínios .edu.br

☐ Apenas domínios .gov.br ☐ Apenas domínios .org.br

☐ Procure apenas neste site/domínio:

Dica: Você pode procurar resultados em um site específico (ex.: yahoo.com.br) ou em domínios de alto nível - TLDs (ex.: .com, .org, .gov).

Formato de arquivo

Mostrar apenas resultados em:

Figura 5 - Exemplo de página com busca avançada do Yahoo

Fonte: Adaptado (MEMÓRIA, 2005, p. 62; YAHOO, 2012)

Contudo, indiferente das limitações abordadas, entende-se que Sistema de Levantamento de Preços permanece com sua plena utilidade para consulta por outros órgãos de auditoria como o PROCOM ou para outras finalidades não citadas. Inclusive, há possibilidade de uso pelos proprietários de postos de combustíveis em busca dos preços praticados pelos concorrentes. Porém, entende-se que estas possibilidades não refletem as principais finalidades do SLP.

Segundo a ANP (2012), é permitida a reprodução total ou parcial das informações publicadas no SLP, incluindo os preços dos combustíveis. Haja visto esta possibilidade, neste trabalho, as informações do SLP, será processada pelo GasFinder de forma automatizada via *data scraping*, uma tecnologia que será abordada posteriormente neste trabalho.

O GasFinder será capaz de importar os preços dos postos dos combustíveis do SLP e irá exibi-los em uma interface WEB, tornando-se um novo canal de consulta para o usuário, alternativo, minimizando as limitações de usabilidade oferecidas pela interface do SLP e oferecendo uma experiência de navegação moderna, com ferramenta de busca, localização geográfica por ip, navegação por mapa, e integração com redes sociais para o compartilhamento de informações.

Torna-se necessário ressaltar, novamente, que a proposta do GasFinder é preencher uma lacuna existente única e exclusivamente na tecnologia de distribuição de conteúdo *online* do SLP e suas limitações de usabilidade. Portanto, se reconhece neste trabalho, a total importância do órgão ANP e sua contribuição para o país e os consumidores.

A ausência do monopólio da indústria do petróleo, através de uma concessão moderna viabilizada pela ANP, tornou-se primordial para garantia de um mercado mais competitivo, gerando vantagens para o país e para o consumidor. Observa-se neste capítulo, uma importância indispensável da ANP, contudo, entende-se que há espaço para melhorias na tecnologia de divulgação dos preços dos combustíveis via WEB, através do SLP, principalmente para o consumidor final, sendo o GasFinder, uma proposta que visa suprimir as restrições citadas acerca do Sistema de Levantamento de Preços (SLP). Para tanto, a seguir neste trabalho, será realizado a abordagem da Engenharia de Software, que visa constituir o embasamento teórico necessário para o desenvolvimento do aplicativo GasFinder.

3 ENGENHARIA DE SOFTWARE

Este capítulo visa reunir o embasamento teórico deste trabalho, acerca da Engenharia de Software e Engenharia WEB, abordando Modelos de Desenvolvimento de Software, Metodologias de Desenvolvimento Ágeis, Engenharia de Requisitos e UML.

Segundo Sommerville (2007, p. 5), “a Engenharia de Software é uma disciplina de engenharia relacionada com todos os aspectos da produção de software, desde os estágios iniciais de especificação do sistema até sua manutenção”. Nesta definição, notam-se duas frases importantes:

1. Disciplina de engenharia: os engenheiros fazem as coisas funcionarem. Eles aplicam teorias, métodos e ferramentas onde for apropriado, mas eles usam de forma seletiva e sempre procuram descobrir soluções para os problemas, mesmo quando não existem teorias e métodos aplicáveis. Os engenheiros reconhecem também que devem trabalhar sob restrições organizacionais e financeiras, e procuram soluções sem perder de vista estas restrições.
2. Todos os aspectos da produção de software: a engenharia de software não está relacionada apenas com os processos técnicos de desenvolvimento de software, mas também com atividades como o gerenciamento de projeto de software e o desenvolvimento de ferramentas, métodos e teorias que apoiem a produção de software. (SOMMERVILLE, 2007, p. 5)

Toda Engenharia é baseada em processos, que são “maneiras pelas quais se realiza uma operação, segundo determinadas normas”. (PAULA FILHO, 2009, p. 4). Deste modo, entende-se que a Engenharia de Software, permite que o desenvolvedor ou engenheiro de software obtenha um maior controle dos processos de forma geral. Pressman (2006, p. 17), define a Engenharia de Software, como sendo “a criação e utilização de sólidos princípios de engenharia a fim de obter softwares econômicos que sejam confiáveis e que trabalhem eficientemente em máquinas reais”.

De acordo com a figura 6, durante toda abordagem de Engenharia de Software, deve-se possuir um compromisso constante com a qualidade de software, quase

como uma filosofia que deve ser seguida, para consolidar uma cultura que ofereça como resultado, o aperfeiçoamento contínuo do processo, que por sua vez, reflita na melhoria da qualidade de software.

As quatro camadas fundamentais da Engenharia de Software segundo Pressman (2006, p. 17 - 18), descreve-se conforme abaixo:

- **Ferramentas:** Oferecem apoio automatizado aos métodos e os processos. Quando uma ferramenta precisa compartilhar informação com outra ferramenta, é estabelecido um sistema de suporte ao desenvolvimento de software que se chama Engenharia de Software Auxiliada por Computador (CASE - *Computer-Aided Software Engineering*).
- **Métodos:** Fornecem os detalhes de como fazer para construir um software. Abrangem um vasto conjunto de tarefas, que são: comunicação, análise de requisitos, modelagem, codificação, testes e manutenção.
- **Processo:** Elo entre métodos e ferramentas, o que permite o desenvolvimento do software. Também forma a base necessária para o controle gerencial do projeto de software, fornecendo documentos para gestão, como modelos, formulários, etc.
- **Foco na qualidade:** É a base em que se apóiam todas as outras camadas da Engenharia de Software. É usada como referencia, que deve ser monitorada constantemente pelas outras camadas de engenharia (ferramentas, métodos e processos), auxiliando na tomada de decisão, tendo sempre em vista, a busca constante pela qualidade.



Figura 6 - Engenharia de Software em Camadas

Fonte: Adaptado (PRESSMAN, 2006, p. 17)

A seguir, será abordado modelos de processo desenvolvimento de software, Cascata, Espiral e Prototipagem, previstos segundo a Engenharia de Software.

3.1 MODELO EM CASCATA

O Modelo em Cascata, algumas vezes chamado de ciclo de vida clássico, sugere uma abordagem sistemática e seqüencial para o desenvolvimento de software que começa com a especificação dos requisitos pelo cliente e progride ao longo do planejamento, modelagem, construção e implantação, culminando na manutenção progressiva do software acabado. (PRESSMAN, 2006, p. 39).

Segundo Sommerville (2007, p. 45), as vantagens do Modelo em Cascata são “a documentação produzida em cada fase e sua aderência a outros modelos de processo de engenharia”. Paula Filho (2009, p. 92 - 93), afirma que os principais processos “são executados em estrita seqüência, o que permite demarcá-los em pontos de controle bem definidos. Estes pontos facilitam a gestão de projetos, o que faz com que esse processo seja, em princípio, confiável para projetos de qualquer escala”.

O maior problema do Modelo em Cascata, é a inflexibilidade de mudanças de requisitos que este modelo fatalmente oferece, pois as principais decisões do projeto são tomadas no estagio inicial do processo. Segundo Paula Filho (2009, p. 93), “as atividades de requisitos, análise e desenho tem que ser muito bem dominadas, pois, teoricamente, o processo não prevê a correção posterior de problemas nas fases anteriores”. Por este motivo, deve-se utilizar o Modelo em Cascata “apenas quando os requisitos forem bem compreendidos e houver pouca probabilidade de mudanças radicais durante o desenvolvimento do sistema”, segundo Sommerville (2007, p. 45).

De acordo com a figura 7, o Modelo em Cascata possui cinco fases principais, Comunicação, Planejamento, Modelagem, Construção e Implantação. O objetivo de cada fase, é descrito conforme abaixo. (PRESSMAN, 2006, p. 39):

- Comunicação: iniciação do projeto e levantamento de requisitos;
- Planejamento: estimativas, geração de cronogramas e monitoramento do projeto;
- Modelagem: etapa de análise;
- Construção: fase de codificação e testes;
- Implantação: etapa final, onde é realizada a entrega, manutenção e análise dos comentários sobre o projeto.

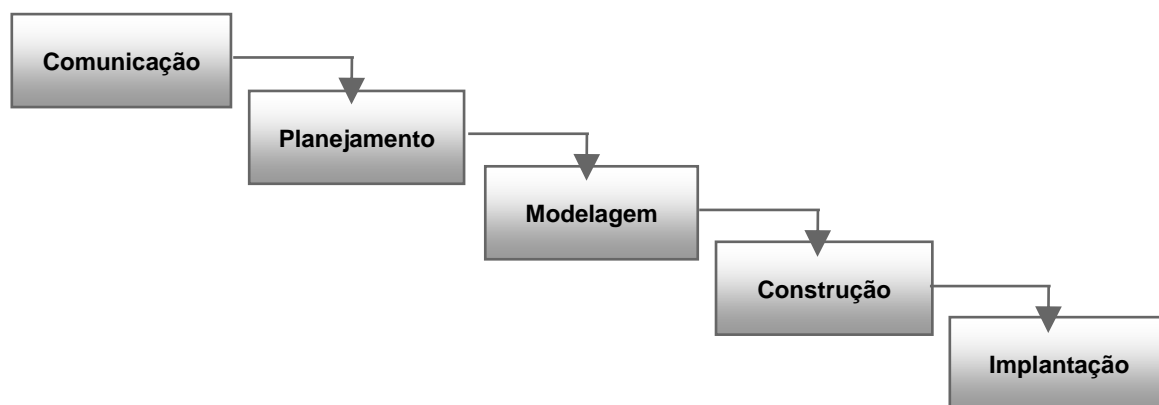


Figura 7 - Modelo em Cascata

Fonte: Adaptado (PRESSMAN, 2006, p. 39)

3.1 MODELO EM ESPIRAL

Este modelo de desenvolvimento segundo Pressman (2006, p. 44), é “modelo de processo guiado por risco e usado para guiar a engenharia de sistemas intensivos em software com vários interessados concorrentes”. Diferentemente do Modelo em Cascata, no Modelo em Espiral, o software é desenvolvido através de uma série de interações, onde cada interação equivale a uma volta na espiral e, ao final de cada ciclo, libera-se uma parte funcional do software para avaliação do cliente. A representação gráfica do Modelo em Espiral é feita conforme é demonstrado na figura 8.

A principal diferença entre o Modelo em Espiral e os outros modelos de processo de software é o reconhecimento explícito do risco no Modelo em Espiral. Informalmente, risco significa simplesmente algo que pode dar errado. Por exemplo, se a intenção for usar uma nova linguagem de programação, um risco é que os compiladores disponíveis não sejam confiáveis ou não produzam código-objeto suficientemente eficaz. Os riscos podem causar problemas no projeto, tal como ultrapassar o cronograma e os custos. (SOMMERVILLE, 2007, p. 49).

Entre as vantagens do Modelo em Espiral, pode-se citar, segundo Paula Filho (2009, p. 94), a possibilidade de “construir produtos em prazos curtos, com novas características e recursos que são agregados a medida que a experiência descobre sua necessidade.”

Segundo Pressman (2006, p. 45), “o Modelo em Espiral não é uma panacéia⁵. Pode ser difícil convencer os clientes que a abordagem evolucionária é controlável. Ela exige competência considerável para se obter sucesso.” Deste modo, além das vantagens citadas anteriormente, este modelo também apresenta possíveis barreiras quando este ciclo de vida é adotado.

O Modelo em Espiral contém as cinco atividades principais descritas abaixo. Além destas atividades, há uma sub-fase oculta na atividade de planejamento, notadamente importante que deve-se citar, a prototipação, usada para a redução de riscos, a fim de elucidar os requisitos ao usuário. As cinco macro-atividades do Modelo em Espiral são demonstradas conforme a seguir (PRESSMAN, 2006, p. 45):

- Comunicação: início do projeto e objetivos;
- Planejamento: estimativa de projeto, cronograma e análise de riscos;
- Modelagem: análise de projeto;
- Construção: codificação e testes;
- Implantação: entrega e análise de resultados e comentários.

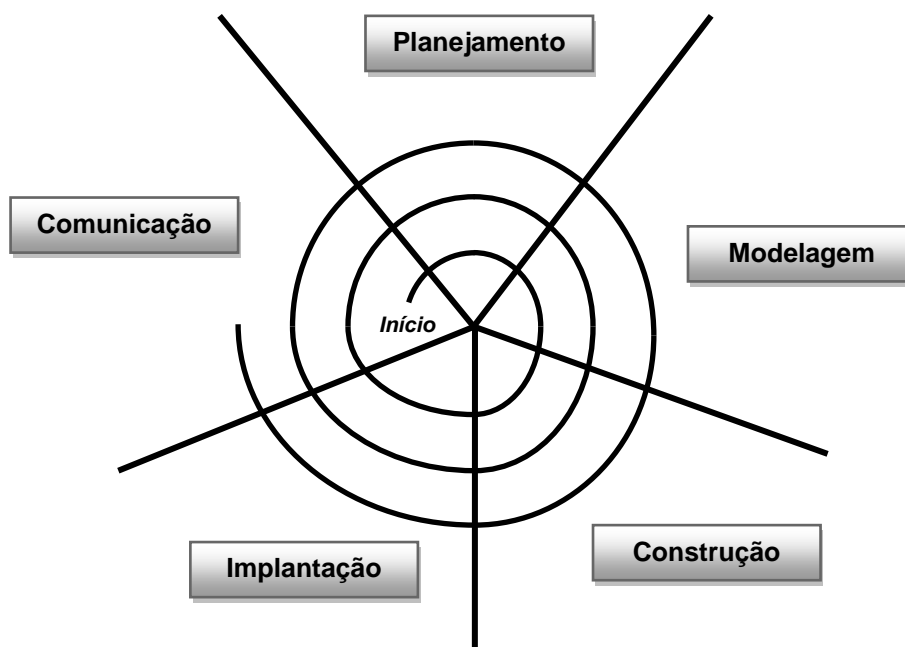


Figura 8 - Modelo em Espiral

Fonte: Adaptado (PRESSMAN, 2006, p. 45)

⁵ Panacéia: Remédio para todos os males. (FERREIRA, 2009, 1476). Pode-se afirmar que esta expressão refere-se a algo, como uma solução universal para todos os problemas, no contexto onde é aplicado neste trabalho.

3.2 MODELO DE PROTOTIPAGEM

Sommerville (2007, p. 271) afirma que “um protótipo é a versão inicial de um sistema de software usado para demonstrar conceitos, experimentar opções de projeto e, geralmente, conhecer mais sobre o problema e suas possíveis soluções.”

A partir deste ponto de vista, um processo de desenvolvimento de software, pode utilizar-se da criação de protótipos de forma rápida, visando obter uma avaliação antecipada dos requisitos pelo cliente. Após essa avaliação, é provável que o cliente sinalize a necessidade de ajustes, para que o software criado atenda os requisitos do projeto. Esta possibilidade é sinalizada por Gustafson (2003, p. 13), ao afirmar que “o esforço despendido no protótipo geralmente é compensado pelo não-desenvolvimento de características desnecessárias.”

Um protótipo pode ser utilizado em um processo de desenvolvimento de software de três maneiras, segundo Sommerville (2007, p. 271):

1. No processo de engenharia de requisitos, um protótipo pode ajudar na descoberta e validação dos requisitos do sistema;
2. No processo de projeto do sistema, um protótipo pode ser usado para explorar soluções específicas de software e apoiar o projeto de interface com o usuário;
3. No processo de teste, um protótipo pode ser usado para realizar testes complexos com o sistema que será entregue para o cliente.

Um estudo de 39 projetos de prototipação, identificou que a prototipação geralmente aumenta os custos iniciais do processo de software, mas reduz esses custos nos estados mais avançados de desenvolvimento. Este estudo também apontou cinco benefícios da prototipação, segundo Sommerville (2007, p. 272):

- Maior usabilidade do software;
- Maior adequação do software com as necessidades do usuário;
- Maior qualidade do projeto;
- Manutenção mais fácil;
- Menor esforço no desenvolvimento.

Segundo Pressman (2006, p. 42), o Modelo de Prototipagem, por sua vez, “auxilia o engenheiro de software e o cliente a entenderem melhor o que deve ser construído quando os requisitos estão confusos.”

O Modelo de Prototipagem faz o uso da prototipação no processo de desenvolvimento, e é indicado quando o cliente não define detalhadamente os requisitos de entrada, processamento e saída do software desejado, e simplesmente aborda um conjunto de objetivos gerais ou ainda, quando o desenvolvedor não possui segurança sobre a eficiência, adaptabilidade, forma de interação homem-máquina. (PRESSMAN, 2006, p. 43).

O Modelo de Prototipagem possui alguns riscos clássicos: O cliente visualiza o protótipo e tem sensação de urgência para colocar em produção, não considerando a qualidade geral do software. Sommerville (2007, p. 272), lembra que “um mesmo protótipo não pode atender a todos objetivos”. Gustafson (2003, p. 13), também afirma que o Modelo de Prototipagem possui limitações, pois “este modelo de ciclo de vida constrói uma versão descartável do sistema”. Em muitos casos a equipe de desenvolvimento deseja colocar o protótipo em execução muito rápido, com isso, é possível que um sistema operacional ou linguagem de programação inadequada seja empregado.

Na figura 9 pode-se observar o Modelo de Prototipagem em uma interação contínua, que visa o constante refinamento do protótipo até a concepção de uma solução adequada. A seguir, será abordado duas metodologias de Desenvolvimento Ágil, Scrum e XP, a fim de oferecer uma estrutura conceitual para administrar projetos de Engenharia de Software.

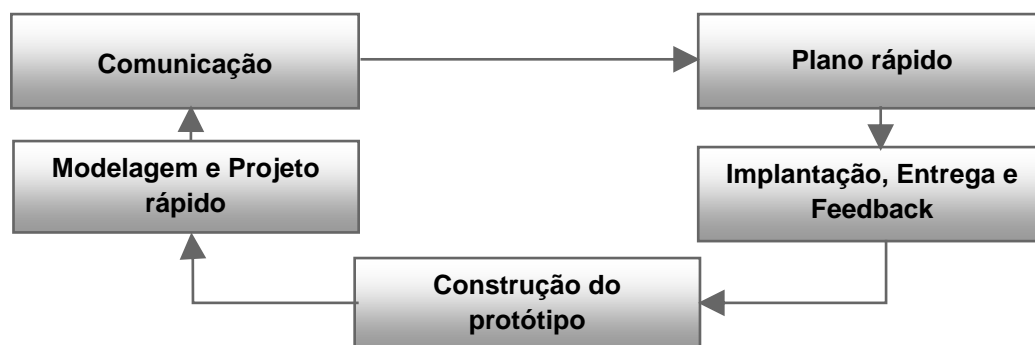


Figura 9 - Modelo em Prototipagem

Fonte: Adaptado (PRESSMAN, 2006, p. 43)

3.3 DESENVOLVIMENTO ÁGIL

Segundo Schwaber (2004 apud NETO, 2008, p. 72), o Desenvolvimento Ágil abrange uma série de “processos para desenvolvimento de software, utilizando técnicas interativas e incrementais calcadas em equipes auto organizadas, auto gerenciáveis e multifuncionais. Focado em pessoas e o que elas podem fazer.”

Os princípios do Desenvolvimento Ágil visam minimizar as falhas no desenvolvimento do projeto, para isso utiliza-se o desenvolvimento de forma incremental, em curtos períodos, chamados de iteração⁶, ou seja, mini-projetos de software que incluirão as tarefas necessárias para alcançar a funcionalidade do produto.

A engenharia de software ágil combina uma filosofia e um conjunto de diretrizes de desenvolvimento. A filosofia encoraja a satisfação do cliente e a entrega incremental do software logo de início, equipes de projeto pequenas, altamente motivadas, métodos informais, produtos de trabalho de engenharia de software mínimos e simplicidade global de desenvolvimento. As diretrizes de desenvolvimento enfatizam a entrega em contraposição a análise e ao projeto (apesar dessas atividades não serem desencorajadas) e a comunicação ativa entre desenvolvedores e clientes. (PRESSMAN, 2006, p. 58).

Segundo Paula Filho (2009, p. 102), o Desenvolvimento Ágil é caracterizado por princípios, que se situam “mais baseados no trabalho cooperativo do que no formalismo e na documentação escrita”. Deste modo, considera-se princípio do Desenvolvimento Ágil: priorizar as pessoas, interações com processos e ferramentas e, software executável, no lugar de uma documentação ampla e, principalmente a colaboração do cliente com resposta rápidas, visando alterações no escopo durante o andamento do projeto.

Estes princípios traduzem algumas atitudes, como a busca pela satisfação do cliente, a colaboração próxima entre gerentes e desenvolvedores, facilidade na comunicação e ênfase na confiança entre as pessoas. A motivação, sobretudo, onde as equipes de negócio e desenvolvimento devem trabalhar juntas diariamente, onde deve-se haver um ambiente e apoio necessário para realização do trabalho.

⁶ Iteração: Ato de iterar; repetição. (FERREIRA, 2009, p. 1141).

Ter o software funcionando é sempre a melhor medida de progresso. A equipe deve receber bem as alterações nos requisitos, mesmo estando em um nível adiantado de desenvolvimento, pois o direcionamento dessas mudanças constitui em um diferencial competitivo para o cliente. (PRESSMAN, 2006, p. 61 - 62).

3.3.1 Scrum

O Scrum é definido por Pressman (2006, p. 69), como um processo que fornece “a habilidade de declarar um produto ‘pronto’ sempre que necessário”. Trata-se de uma metodologia para gerenciamento de projetos, também chamado de *framework* para desenvolvimento de software, onde seu objetivo principal é a agilidade do processo. E para isto, tem como premissa, não fazer novas negociações e entregar sempre no prazo proposto.

Segundo Adami et al. (2007, p. 42) o Scrum é uma metodologia que “aceita o desenvolvimento de software imprevisível e formaliza a abstração, sendo aplicável a ambientes voláteis. Ele se destaca das demais Metodologias Ágeis pela maior ênfase no gerenciamento de projeto”.

Segundo Larman (2007, p. 67), o Scrum é uma “abordagem interativa popular que aplica interações de tempo limitado de 30 dias e com uma reunião diária, em pé, com três questões especiais respondidas por cada membro da equipe”. Se porventura forem notificados imprevistos pela equipe, o prazo poderá ser revisto dentro do projeto, contudo isto não impedirá que o software seja entregue no prazo.

A seguir apresenta-se uma descrição das atividades a serem realizadas em cada fase do Scrum, segundo Schwaber (2004 apud SEVERINO, 2009, p. 44 - 45):

1. Planejamento: Desenvolvimento claro e objetivo de uma lista de atividades do produto; Definição da data de *release* e funcionalidades de um ou mais *sprints*; Definição do *release* apropriado para começar o ciclo de desenvolvimento; Mapeamento e estimativa das atividades; Definição do time do projeto; Avaliação e controle de riscos; Avaliação das ferramentas de desenvolvimento e infra-estrutura; Estimativa de custos.

2. Arquitetura: Revisão e possíveis ajustes na lista de atividades do produto; Identificação das mudanças necessárias para implementar as atividades do produto; Realizar a análise de domínio; Refinar a arquitetura do sistema; Identificação de possíveis problemas ou impedimentos na implementação dos requisitos; Reunião de revisão do *design*, onde são apresentadas cada proposta de implementação de cada item do *backlog*.
3. Desenvolvimento (*Sprint*): Reunião de planejamento do *sprint*, a ser realizada sempre no primeiro dia de cada *sprint*. Essa reunião deve definir as atividades a serem incluídas na iteração corrente. Reuniões diárias com os membros da equipe para revisar o andamento do projeto; Revisão e ajustes nos requisitos do projeto; *Sprints* iterativos até que o produto seja considerado pronto para a entrega.

10. A metodologia Scrum segue o processo conforme é demonstrado na figura

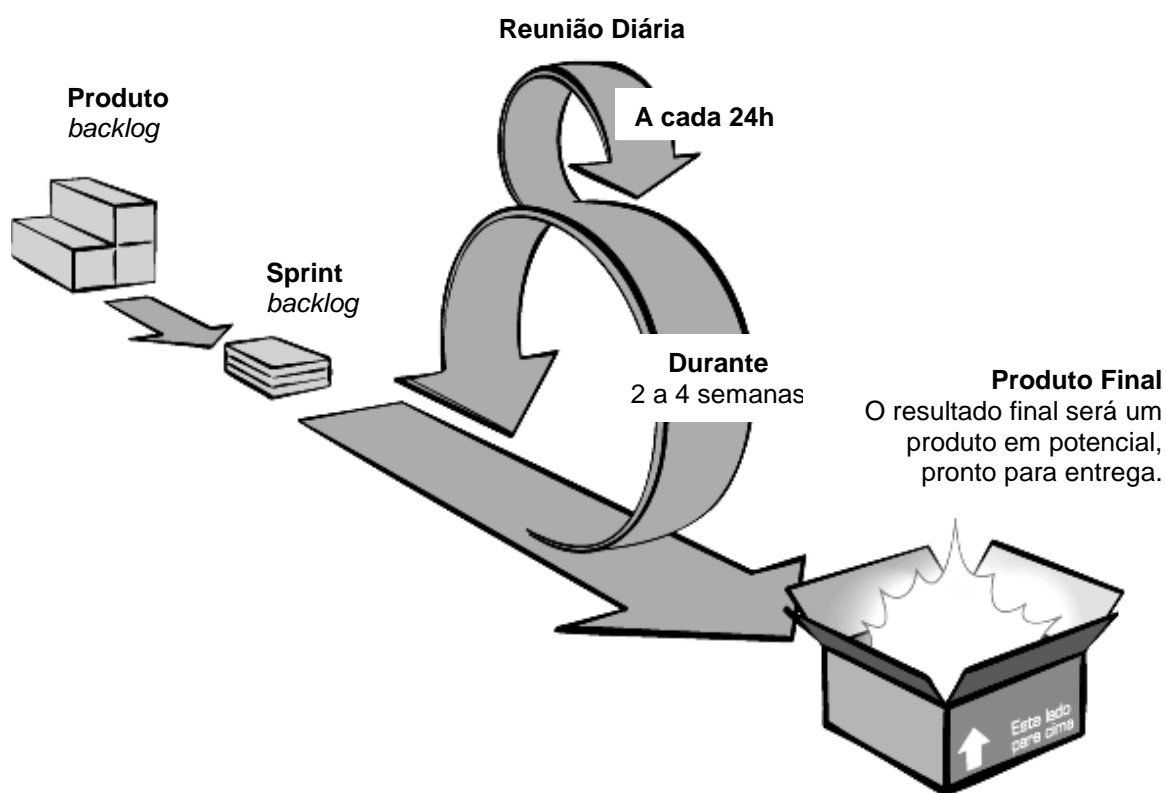


Figura 10 - Metodologia de Desenvolvimento Scrum

Fonte: Adaptado segundo Schwaner (2004 apud EBERT, 2009, p. 28)

3.3.2 Extreme Programming

Programação Extrema (do inglês *Extreme Programming*), ou simplesmente XP, é uma metodologia ágil para equipes pequenas ou médias que buscam desenvolver software com requisitos imprecisos ou em constante transformação. Segundo Sommerville (2007, p. 50), na metodologia XP, “os requisitos são desenvolvidos de forma incremental, de acordo com as prioridades do usuário, e a elicitação de requisitos provém de usuários que fazem parte da equipe de desenvolvimento”.

Os quatro valores fundamentais da metodologia XP são: Comunicação, Simplicidade, *Feedback* e Coragem, descritos a seguir, segundo Beck (1999 apud BALLE, 2011, p. 30 - 31):

1. Comunicação: A maioria dos problemas que ocorrem nos projetos invariavelmente tem sua causa associada ao fato de alguém não ter informado alguma coisa muito importante para uma pessoa que precisava saber. Dessa forma, a comunicação é o valor de maior importância no XP.
2. Simplicidade: A simplicidade não é fácil. Uma das coisas mais difíceis de se fazer é não olhar para as coisas que serão necessárias implementar no dia seguinte, na semana seguinte e no mês seguinte. Deve ser implementado apenas aquilo que é necessário e realmente importa ser construído. Isto significa dizer que as pessoas só devem se preocupar em solucionar hoje os problemas de hoje. A complexidade custa muito caro e tentar prever o futuro é muito difícil. É necessário aguardar o futuro para ver se está certo ou não.
3. *Feedback*: A veracidade dos relatórios do estado atual das atividades é extremamente importante em XP. *Feedback* significa perguntar e aprender com as respostas. A única forma de saber a necessidade do cliente é perguntando a ele. O único modo de saber se um código faz o que ele se destina fazer é testando-o. Quanto mais cedo se obter o *feedback*, mais tempo se terá disponível para reagir. A metodologia XP fornece um *feedback* rápido e freqüente por parte dos seus seguidores.
4. Coragem: Depois de se falar nos três valores, comunicação, simplicidade e *feedback*, é hora de se esforçar como nunca antes. Se o trabalho não for desempenhado na sua velocidade mais rápida possível, alguém mais o irá fazer, e ele vai lucrar no lugar. A coragem significa tomar as decisões na hora em que elas precisam ser tomadas. Se uma funcionalidade não

está funcionando, ela deve ser consertada. Se algum código não parece estar bem escrito, ele deve ser refatorado. Se não será possível entregar tudo o que se havia prometido dentro do prazo acordado, o cliente deve ser informado. A coragem é uma virtude difícil de se aplicar. Ninguém gosta de estar errado ou quebrar um acordo. O único modo de consertar um engano é admitir que houve um engano e consertá-lo.

Para colocar em pratica os valores fundamentais ao decorrer de um projeto de software utilizando XP, deve-se existir uma confiança significativa quanto a sinergia dos desenvolvedores e clientes, onde os possíveis pontos fracos do projeto tendem a ser superados pelos pontos fortes.

Aconselha-se a priorização do desenvolvimento de funções que agregue mais valor para o negócio do cliente. Desta forma, quando ocorrer a necessidade de reduzir o escopo, as funcionalidades com menos importância terão o desenvolvimento postergado ou até mesmo cancelado. (PRESSMAN, 2006, p. 63 - 64).

Percebe-se, que a XP estimula o domínio sobre qualidade do projeto, pois qualquer ganho razoável na produtividade, se reduzir a qualidade, então não poderá compensar as futuras perdas (ou até limitações) a médio e longo prazo. Na figura 11 é demonstrado o processo da metodologia de desenvolvimento XP.

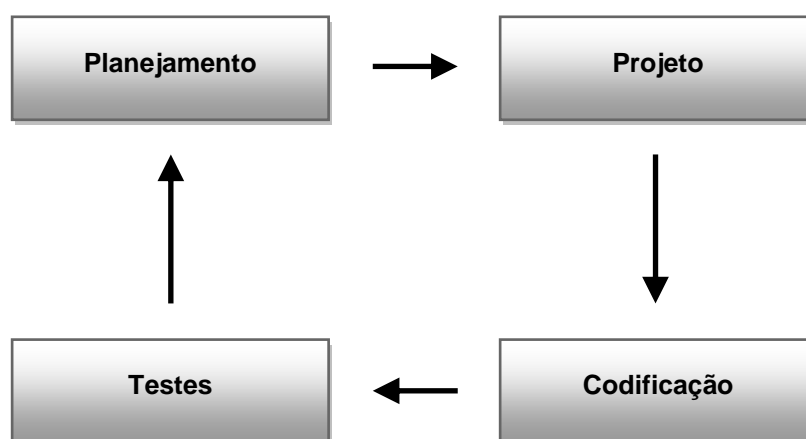


Figura 11 - Metodologia de Desenvolvimento XP

Fonte: Adaptado (PRESSMAN, 2006, p. 64)

3.1 ENGENHARIA WEB

A Engenharia WEB (WebE) surge como uma solução específica para o crescente avanço do desenvolvimento de aplicações para WEB - também conhecida como WebApp (*Web Based Application*) - visto sua complexidade, similar e senão superior a projetos *off-line*.

As WebApps evoluíram consideravelmente e atualmente oferecem soluções integradas com ERPs, sistemas legados, bancos de dados diversos em praticamente todos softwares empresariais. Contudo, segundo Ginige (2002 apud OBERDORFER, 2005, p. 19), o desenvolvimento de WebApps apresentam diversas deficiências de projeto, “84% em aplicações não atendem as necessidades do mercado, 53% não cobrem suas funcionalidades requeridas, 63% excedem seus orçamentos, sendo que 79% destes contratempos resultam de problemas no projeto”. Isto ocorre, segundo Ginige e Murugesan (2001 apud BRITO, 2003, p. 15), porque a Engenharia de Software convencional, não oferece abstração suficientemente capaz de especificar as WebApps, que modificam constantemente os requisitos durante o ciclo de vida.

Segundo Ginige e Muragesan (2001 apud GONÇALVES, 2010, p. 33), a engenharia WEB é definida como a “aplicação de princípios científicos, de engenharia, gestão e abordagem sistemática para o desenvolvimento, implantação e manutenção de sistemas e aplicações WEB de alta qualidade”. Para Pressman e Lowe (2009, p. 11), a engenharia WEB “propõe um arcabouço⁷ ágil, porém disciplinado para montagem de WebApps de qualidade industrial”.

Um arcabouço estabelece o alicerce para um processo completo de engenharia WEB, identificando um pequeno número de atividades de arcabouço que se aplicam a todos os projetos de WebApp, independentemente de seu tamanho e complexidade. Além disso, um arcabouço abrange um conjunto de atividades guarda-chuva, que se aplicam ao processo WebE inteiro. (PRESSMAN; LOWE, 2009, p. 12)

⁷ Arcabouço: O esqueleto; traços gerais. (FERREIRA, 2009, p. 180). Pode-se afirmar no contexto de uso deste trabalho, como uma expressão que refere-se a uma estrutura principal de algo.

As aplicações WebApps possuem um grupo de características únicas que a diferenciam de um software convencional (PRESSMAN; LOWE, 2009, p. 5 - 6):

- Intensidade da rede: acesso local ou Internet;
- Simultaneidade: capacidade de aceitar acesso simultâneo;
- Carga imprevisível: em função de ataques de hackers, por exemplo;
- Desempenho: tempo de processamento e resposta da aplicação;
- Disponibilidade: período em que os serviços estão acessíveis;
- Orientada a objetos: suporte a banco de dados e linguagem de programação moderna;
- Sensível ao conteúdo: oferecer o conteúdo adequado aos usuários;
- Urgência: necessidade de lançar o produto rapidamente no mercado;
- Segurança: com objetivo de abrigar todo conteúdo privilegiado;
- Estética: característica estética e de interação com o usuário.

3.1.1 Atividades da Engenharia WEB

A WebE (Engenharia WEB) apresenta atividades genéricas que se aplicam a grande maioria de projetos de WebApp, conforme é demonstrado na figura 12 (PRESSMAN; LOWE, 2009, p. 12 - 13):

- Comunicação: abrange interação com o cliente e interessados;
- Planejamento: define um plano de entrega incremental;
- Modelagem: criação de modelos para compreensão do projeto;
- Construção: codificação com emprego de HTML, CSS e etc;
- Implantação: entrega um incremento de WebApp ao cliente.

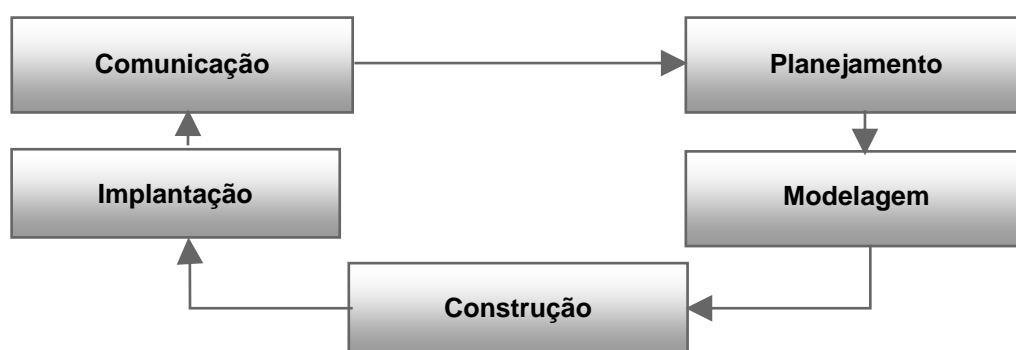


Figura 12 - Fluxo do processo com ações de WebE

3.1.2 Padrões de Projeto de Hipermídia

As WebApps continuam evoluindo e aumentando sua complexidade. Contudo, os Padrões de Projeto clássicos da Engenharia de Software ainda não são totalmente apropriados para o desenvolvimento WEB, que possui características próprias.

Segundo Pressman (2006, p. 446): “Padrões de projeto usado na engenharia da WEB englobam duas classes principais: (1) padrões de projeto genérico aplicáveis a todos os tipos de software e (2) padrões de projeto de hipermídia específicos de WebApps.”

No contexto de software baseado em WEB, existem as seguintes categorias de padrões. (PRESSMAN, 2006, p. 446 - 447):

- Padrões arquiteturais: apóiam o projeto de conteúdo e de arquitetura da WebApp;
- Padrões de construção de componentes: recomendam métodos para combinar componentes da WebApp;
- Padrões de navegação: correspondem aos fluxos de navegação e *links*;
- Padrão de apresentação: trata da usabilidade e hierarquia de conteúdo;
- Padrões de comportamento/usuário-interação: define como a interface deve sinalizar e responder as interações do usuário.

3.1.3 OOHDM

Segundo Pressman (2006, p. 447), “um certo número de métodos de projeto para aplicações WEB foi proposto na década passada. Até agora, nenhum método simples conseguiu sobressair-se.” Nessa sessão, será abordado um dos métodos mais discutidos segundo Pressman (2006, p. 447).

OOHDM ou Método de Projeto de Hipermídia Orientado a Objetos (*Object-Oriented Hypermedia Designs Method*), “vem se apresentando muito satisfatório na comunidade hipermídia, sendo bem aceito em diversos países”, segundo Oberderfer (2005, p. 19).

Ainda segundo Costagliola (2002, apud OBERDERFER, 2005, p. 25):

The OOHDM methodology improves maintainability and reusability, thanks to both the separation of the design phases and the abstraction capabilities which are characteristics of the object-oriented design, isto é, a metodologia OOHDM melhora a manutenibilidade e a reusabilidade, permitidos pela separação das fases do projeto e pelas potencialidades de abstração, que são características dos projeto orientados a objeto.

OOHDM é composto de quatro atividades de projeto segundo Pressman (2006, p. 448 - 449):

- Projeto conceitual: define a semântica e o domínio da aplicação;
- Projeto navegacional: considera o perfil do usuário e a tarefa alvo, enfatizando aspectos cognitivos;
- Projeto e implementação de interface abstrata: especifica os objetos visíveis pelo usuário e, por sua vez, a implementação viabiliza a criação de uma solução compatível para executar onde o WebApp irá opera.

3.1.4 Modelos Wireframe

Segundo Memória (2005, p. 30), *wireframes* “são como uma planta baixa do site, prevendo cada detalha e funcionalidade que será utilizada. O nível de complexidade do desenho depende do projeto, que normalmente começa simples e vai ganhando detalhes com o tempo”. Segundo Pressman e Lowe (2009, p. 235 - 236), os *wireframes* têm como objetivo, “a descrição de como uma página individual (ou páginas) dentro de uma WebApp poderia se parecer conceitualmente.”

Através da criação de *wireframes* (leiaute da interface do usuário), é possível visualizar a distribuição dos elementos na tela, também permite analisar a interação e realizar testes e avaliações com o usuário antes da codificação do sistema, evitando retrabalho com ajustes imprevistos ao decorrer do desenvolvimento do sistema. Os *wireframes* desenvolvidos neste trabalho podem ser observados na sessão Apêndices.

3.2 ENGENHARIA DE REQUISITOS

Segundo Sommerville (2007, p. 95), o objetivo da engenharia de requisitos é “criar e manter um documento de requisitos do sistema”. O processo compreende quatro etapas principais, que estão relacionadas a validar se o software é útil para empresa, incluindo a obtenção dos requisitos, especificação destes requisitos e finalmente a validação destes requisitos, para verificar se realmente o software atende as necessidades do cliente.

A engenharia de requisitos ajuda os engenheiros de software a compreender melhor o problema que eles vão trabalhar para resolver. Ela ainda inclui o conjunto de tarefas que levam a um entendimento de qual será o impacto do software sobre o negócio, do que o cliente quer e de como os usuários finais vão interagir com o software. (PRESSMAN, 2006, p. 116).

As atividades demonstradas na figura 13 se referem a obtenção, documentação e verificação dos requisitos. Além do relacionamento entre estas atividades também é demonstrado os documentos resultantes de cada etapa.

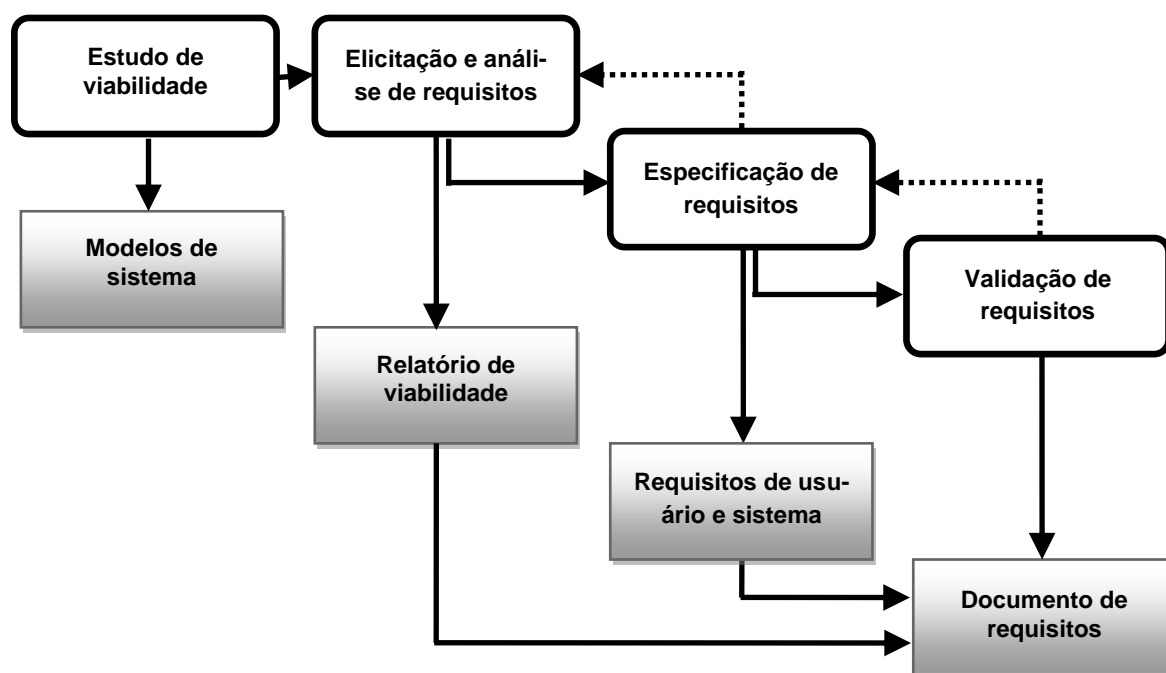


Figura 13 - Processo da Engenharia de Requisitos
Fonte: Adaptado (SOMMERVILLE, 2007, p. 96)

3.3 UML

Segundo Larman (2007, p. 39), a “Linguagem de Modelagem Unificada (UML) é uma linguagem visual para especificar, construir e documentar os artefatos dos sistemas”. Para Booch, Rumbaugh e Jacobson (2006, p. 13), a UML é uma linguagem padrão para a “elaboração da estrutura de projetos de software que poderá ser empregada para a visualização, a especificação, a construção e a documentação de artefatos que façam uso de sistemas complexos de software.”

Através da UML, é possível que um desenvolvedor, elabore um modelo para representar algum artefato do software, de tal modo, que qualquer outro desenvolvedor seja capaz de interpretá-lo.

A UML auxilia na etapa de especificação, que significa construir modelos precisos e sem ambigüidade. Por sua vez, também é possível utilizar a UML na construção do software, onde os modelos gerados podem ser conectados a diferentes linguagens de programação, inclusive é possível mapear até tabelas de bancos de dados. Finalmente, a UML abrange a documentação da arquitetura do software e de todos os seus detalhes.

Segundo Booch, Rumbaugh e Jacobson (2006, p. 16), “a UML é capaz de representar tudo que possa ser melhor expresso em termos gráficos”. Para isso, a UML utiliza a notação diagramática. É através de diagramas que os artefatos de um software são representados pelo desenvolvedor ou pelo engenheiro de software.

Para o desenvolvimento do GasFinder, será abordado neste trabalho, Diagramas de Caso de Uso, Diagrama de Seqüência, Diagrama de Classe, Diagrama de Entidade-Relacionamento e Diagrama de Atividade.

Neste capítulo, constituiu-se o embasamento teórico, a fim de reunir as informações necessárias, para o desenvolvimento deste trabalho. Foi abordado a Linguagem de Modelagem Unificada (UML), que fornece a notação diagramática para representação dos aspectos de um software; Protótipos de Interface, que permite evidenciar antecipadamente os requisitos de um software; Metodologia Ágil Scrum e XP que será de grande importância para a etapa seguinte deste trabalho, para gestão de projeto e desenvolvimento do GasFinder.

4 DESENVOLVIMENTO DO GASFINDER

Neste capítulo, visa conceber o documento de requisitos do GasFinder, contendo tópicos que foram adaptados para este trabalho, de acordo com a estrutura de documentos de requisitos, padronizado pelo IEEE/ANSI 830-1998, segundo Sommerville (2007, p. 92).

Serão abordados os tópicos oriundos gestão de projeto, análise, desenvolvimento, até a amostra de telas funcional do GasFinder, desenvolvido aplicando conceitos e definições adquiridos nos capítulos anteriores, alguns superficialmente e outros em maior profundidade, conforme foram sendo oportunos e necessários para a evolução do trabalho.

4.1 GESTÃO E METODOLOGIA DE DESENVOLVIMENTO

No auxílio a gestão e metodologia de desenvolvimento, utilizou-se algumas boas práticas do Scrum e XP (*Extreme Programming*), por oferecer conceitos dinâmicos de gestão e desenvolvimento, úteis para o trabalho.

De acordo com o cronograma das atividades de projeto demonstrado na figura 14, o desenvolvimento do Gasfinder foi dividido em três fases principais, onde cada fase contém um conjunto de as atividades e seus respectivos recursos, responsáveis pela execução de cada atividade. Os recursos nesta ocasião, tratam-se da equipe deste trabalho (Alexandre e William):

1. Fase de Planejamento: Foi definida uma lista de atividades do produto, as ferramentas e infra-estrutura de desenvolvimento, estimativa das horas de desenvolvimento para cada atividade e a quantidade e o tamanho de cada *Sprint*, que ficou definido em 40 horas, estimando 4 horas diárias de trabalho em um total de 4 *Sprints*.
2. Fase de Arquitetura: Foi revisada a lista de atividade, onde nada foi alterado. Foi feita análise de domínio, utilizando boas práticas do XP, onde os requisitos foram escritos em forma de história e a revisão por pares, em cima de cada história.

3. Fase de Desenvolvimento: Pela dificuldade em fazer reuniões diárias, foi defino reuniões semanais, através do Skype (ferramenta de mensagem instantânea e voz sobre ip), para discutir o que cada membro da equipe tinha concluído, quais a dificuldades encontradas e o que seria feito na próxima semana, conforme a matriz de comunicação, demonstrado na tabela 1.

	Task Name	Duration	Start	Finish	Realized	Resource Names
1	<input type="checkbox"/> Planejamento	20 hrs	Wed 08/06/11	Fri 10/06/11	21 hrs	
2	Criar lista de atividade do produto	4 hrs	Wed 08/06/11	Wed 08/06/11	4 hrs	Todos
3	Definir ferramentas	2 hrs	Wed 08/06/11	Wed 08/06/11	2 hrs	Todos
4	Definir infra-estrutura de desenvolvimento	4 hrs	Wed 08/06/11	Thu 09/06/11	10 hrs	Todos
5	Estimativa de horas de desenvolvimento	8 hrs	Thu 09/06/11	Fri 10/06/11	10 hrs	Todos
6	Definir Sprints(Tamanho/Quantidade)	2 hrs	Fri 10/06/11	Fri 10/06/11	1 hr	Todos
7	<input type="checkbox"/> Arquitetura	56 hrs	Fri 10/06/11	Tue 21/06/11	61 hrs	
8	Revisão da lista de Atividade	2 hrs	Fri 10/06/11	Fri 10/06/11	4 hrs	Todos
9	Levantamento de Requisitos	8 hrs	Fri 10/06/11	Mon 13/06/11	12 hrs	Todos
10	Análise de Domínio	8 hrs	Mon 13/06/11	Tue 14/06/11	10 hrs	Todos
11	Escrever as Histórias	8 hrs	Tue 14/06/11	Wed 15/06/11	16 hrs	Todos
12	Revisão por Pares	8 hrs	Wed 15/06/11	Thu 16/06/11	6 hrs	Todos
13	Definir MER	4 hrs	Thu 16/06/11	Fri 17/06/11	4 hrs	Todos
14	Definir Classes	4 hrs	Fri 17/06/11	Fri 17/06/11	3 hrs	Todos
15	Criar Caso de Uso	4 hrs	Fri 17/06/11	Mon 20/06/11	4 hrs	Todos
16	Diagrama de Atividade	4 hrs	Mon 20/06/11	Mon 20/06/11	4 hrs	Todos
17	Diagrama de Sequência	6 hrs	Mon 20/06/11	Tue 21/06/11	4 hrs	Todos
18	<input type="checkbox"/> Desenvolvimento	161 hrs	Tue 21/06/11	Tue 19/07/11	155 hrs	
19	<input type="checkbox"/> Sprint 1	40 hrs	Tue 21/06/11	Tue 28/06/11	39 hrs	Todos
20	Criação do Banco de Dados	1 hr	Tue 21/06/11	Tue 21/06/11	1 hr	Alexandre
21	Criação do Script de Scraper	8 hrs	Tue 21/06/11	Wed 22/06/11	7 hrs	William
22	Carga no banco com os dados da ANP	30 hrs	Wed 22/06/11	Tue 28/06/11	30 hrs	Todos
23	Reunião Semanal	1 hr	Tue 28/06/11	Tue 28/06/11	1 hr	Todos
24	<input type="checkbox"/> Sprint 2	41 hrs	Tue 28/06/11	Tue 05/07/11	45 hrs	Todos
25	Cadastro de Usuário	5 hrs	Tue 28/06/11	Wed 29/06/11	10 hrs	William
26	Consulta de Usuários	5 hrs	Wed 29/06/11	Wed 29/06/11	4 hrs	Alexandre
27	Cadastro de Preços	5 hrs	Wed 29/06/11	Thu 30/06/11	4 hrs	William
28	Cadastro de Estados	5 hrs	Thu 30/06/11	Thu 30/06/11	4 hrs	Alexandre
29	Cadastro de Postos	5 hrs	Fri 01/07/11	Fri 01/07/11	7 hrs	Alexandre
30	Cadastro de Combustíveis	5 hrs	Fri 01/07/11	Mon 04/07/11	5 hrs	Alexandre

Figura 14 - Cronograma de Atividades do Projeto

Tipo de Comunicação	Objetivo	Meio	Participantes	Dono	Frequência
Reunião Semanal	Analisar o que foi feito e o que será feito	Skype	equipe	William	Semanal
Alteração de Escopo	Analisar necessidade e viabilidade de alteração de escopo	Skype	equipe	William	-

Tabela 1 - Matriz de Comunicação do Projeto

4.1.1 Horas Prevista / Realiza

De acordo com o gráfico das horas previstas e horas realizadas no projeto, demonstrado na figura 15, pode-se observar que a Fase de Planejamento ultrapassou as horas previstas, uma variação de 4,76% acima do planejado. Este estouro se deu, pela dificuldade em se definir a infra-estrutura de desenvolvimento. A Fase de Arquitetura ultrapassou em 8,95% as horas previstas, haja vista a dificuldade em realizar-se o levantamento de todos os requisitos da aplicação. A Fase de Desenvolvimento foi realizada dentro do tempo planejado, sendo concluída em apenas 96,27% das horas estimadas, ou seja, um ganho de 3,73% em horas no cronograma do projeto. Isto se pelo fato de a equipe deste trabalho (Alexandre e William) ter optado por implementar apenas parcialmente 4 requisitos funcionais, e não implementar 2 requisitos, visando concluir o projeto sem estourar o cronograma. (A ação deste replanejamento e suas implicações, será abordada no próximo tópico).

Embora se tenha computado variações, para mais e para menos, durante a execução das atividades, a diferença entre as horas previstas e realizadas foi equilibrada ao decorrer do projeto. As horas adicionais somadas pelas dificuldades encontradas durante a Fase de Planejamento e a Fase de Arquitetura, foram compensadas pelo replanejamento ocorrido na Fase de Desenvolvimento. Por fim, ambas estimativas, horas previstas e realizadas, totalizaram-se em 237 horas, não havendo portanto, atraso ou produtividade significativamente importante para ser considerada acerca do cronograma do projeto.

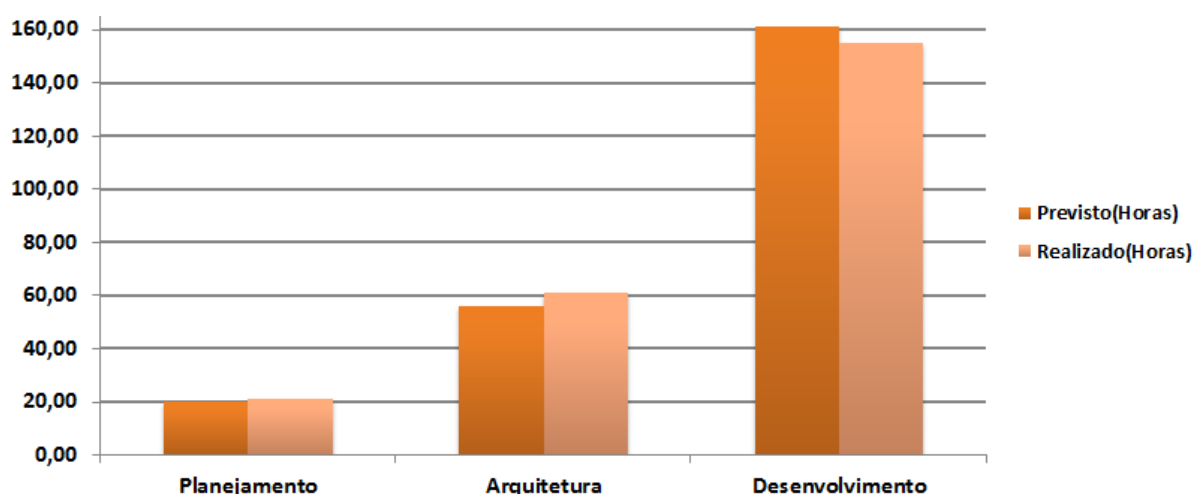


Figura 15 - Horas de Projeto Previstas / Realizada

4.1.2 Atividades previstas e não realizadas

Um total de seis Requisitos Funcionais (RF) teve a implementação desviada, em função de um replanejamento ocorrido na Fase de Desenvolvimento. Esta manobra compromete somente as funcionalidades fornecidas ao Administrador do GasFinder, conforme é demonstrado na figura 16.

RF parcialmente concluído (Administrador)

Para os requisitos abaixo, foi implementado somente a função que lista as informações em tela. As funções inserir, alterar e excluir não foi implementada.

- RF 9 - Cadastrar Combustíveis
- RF 12 - Cadastrar Bandeiras de Postos
- RF 13 - Cadastrar Estados
- RF 14 - Cadastrar Cidades

RF não realizado (Administrador):

Para os requisitos abaixo, a implementação foi cancelada por tempo indeterminado.

- RF 11 - Aprovar Comentários
- RF 15 - Aprovar Preços

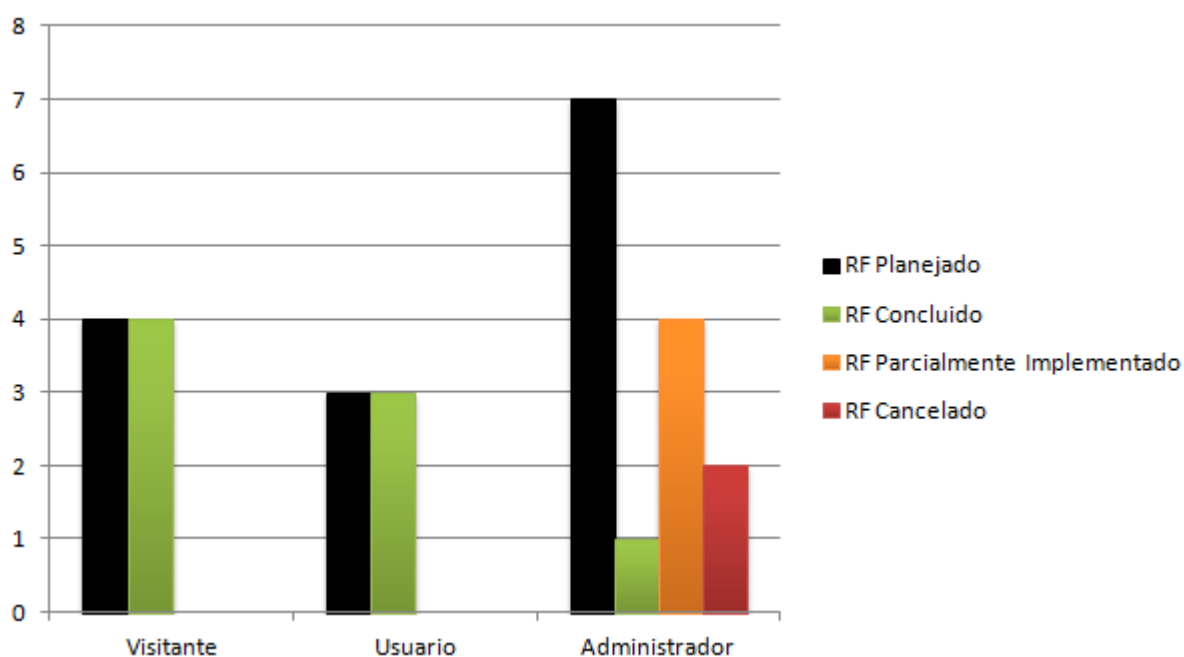


Figura 16 - Funcionalidades disponíveis por usuário

O replanejamento foi necessário, pois percebeu-se durante a Fase de Desenvolvimento, que as horas previstas para as atividades de desenvolvimento foram subestimadas no cronograma do projeto. Encontrou-se dificuldade em agregar mais horas ao cronograma do projeto, devido ao conflito com outras atividades de pesquisa necessária para elaboração do material escrito deste trabalho.

O critério principal para escolha dos requisitos que será comprometido pelo replanejamento, foi desviar apenas os requisitos que causem o menor impacto no uso do aplicativo pelo usuário final. Como critério secundário, foi definido que parte dos requisitos devem ser no mínimo, parcialmente implementados.

De acordo com a figura 17, a soma dos Requisitos Funcionais integralmente implementado e parcialmente implementado, correspondem a 86% das funcionalidades que será contemplada ao fim do desenvolvimento do aplicativo. Estará ausente na aplicação, apenas 14% das funcionalidades planejadas, sendo que estas funcionalidades não afetam diretamente o usuário final, público alvo da aplicação.

Por fim, conclui-se que este replanejamento, causa um pequeno impacto para o projeto, tolerável, pois foi mantêm as principais funcionalidades da aplicação, que caracterizam a essência do aplicativo. As ações realizadas durante o replanejamento, são transparente para o usuário final do GasFinder, Visitante e Usuário Cadastrado, limitando somente os recursos oferecidos para o Administrador.

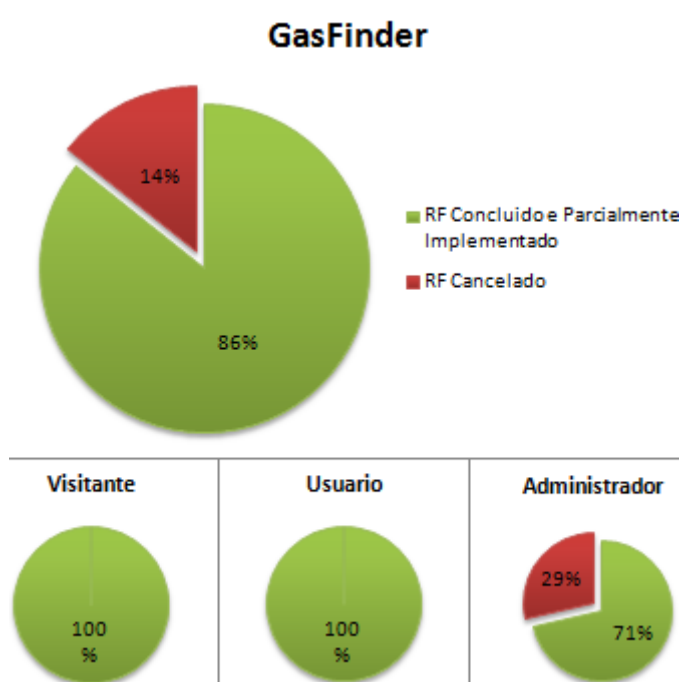


Figura 17 - Percentual de completude das funcionalidades do GasFinder

4.2 ESCOPO DO APLICATIVO GASFINDER

O escopo deste trabalho compreende o desenvolvimento de um aplicativo WEB, com as seguintes características:

- Apresentar a média dos preços dos combustíveis da cidade de acesso, na tela inicial do aplicativo, onde a identificação da cidade será feita através de localização geográfica por IP;
- Buscar e comparar os preços dos combustíveis;
- Marcar os postos de combustíveis em um mapa utilizando o Google Maps;
- Atualizar os preços dos combustíveis fornecidos pelo usuário;
- Adicionar comentários, sobre os postos ou preços dos combustíveis;
- Permitir que usuários e visitantes compartilhem os preços dos combustíveis no Facebook e no Twitter;
- Listar nas buscas somente os postos disponibilizados na ANP.

4.2.1 Descrição Geral

O aplicativo tem por objetivo disponibilizar um serviço de busca, marcação de informações em um mapa, compartilhamento nas redes sociais e comparação dos preços dos combustíveis dos postos de uma localidade.

Irá suportar três tipos de usuários distintos: o visitante que corresponde ao maior número de usuários do GasFinder, que irá realizar a consulta dos preços dos combustíveis; o usuário cadastrado, que é um visitante com privilégio para realizar comentários e atualizar informações; o administrador, responsável por gerenciar todo o aplicativo, adicionar novos postos, cidades, estados e etc.

4.2.2 Funções do Aplicativo

As funções deste aplicativo são efetuar uma busca dos postos e preços dos combustíveis de uma determinada localidade. A aplicação irá carregar automaticamente na tela inicial do aplicativo, informações baseadas na cidade de origem do acesso do usuário, gerando a média dos preços dos combustíveis daquela localidade.

A busca precisará inicialmente ser feita através do campo de busca e posteriormente os postos serão listados e também marcados em um mapa visando identificar sua localização.

A aplicação irá permitir que o usuário visitante efetue o cadastro para obter privilégios adicionais que incluem a permissão para adicionar comentários sobre os postos de combustíveis e alterar o valor dos combustíveis. Também será possível compartilhar os preços dos combustíveis através do Twitter e Facebook.

4.2.3 Características dos Usuários

Existirão três tipos de usuário:

- **Visitante:** serão os usuários que não terão contas no aplicativo e, utilizarão o GasFinder somente para pesquisa.
- **Usuário Cadastrado:** serão os usuários que terão contas no aplicativo e, poderão alterar os valores dos combustíveis e adicionar comentários.
- **Administrador:** será o usuário administrador, que poderá controlar todo o aplicativo.

4.2.4 Restrições Gerais

- Os postos exibidos pelo aplicativo deverão estar cadastrados na ANP.

4.2.5 Suposições e Dependências

- Este trabalho, fará uma amostra, de um aplicativo inovador proposto para suprir uma necessidade sistemática.
- Os dados apresentados pelo GasFinder são de responsabilidade da ANP.
- A periodicidade de atualização dos preços é responsabilidade da ANP.

4.2.6 Requisitos

Será listado a seguir os Requisitos Funcionais (RF) e posteriormente, os Requisitos Não-Funcionais (RNF) do GasFinder, para que a aplicação atenda os objetivos deste trabalho.

Segundo Ferreira (2008, p. 20), os RF “descrevem o que o sistema faz, isto é, as funções necessárias para cumprir os objetivos do sistema”. Para Sommerville (2007, p. 80), “requisitos funcionais são declarações de serviços que o sistema deve fornecer, como o sistema deve reagir a entradas específicas e como o sistema deve se comportar em determinadas situações.”

Baseado na metodologia XP de Desenvolvimento Ágil, será criado um conjunto de histórias de usuário (*user stories*), que segundo Paula Filho (2009, p. 104), é uma “espécie variante simplificada dos casos de uso”. Para Pressman (2006, p. 63) as histórias de usuário “descrevem as características e funcionalidades requeridas para o software a ser construído”.

Para tanto, cada história de usuário deve responder a três questões específicas, criando uma frase afirmativa que irá identificar o requisito, como por exemplo:

- *“Como usuário do tipo X, quero realizar a seguinte atividade Y, para obter o respectivo resultado Z.”*

Os Requisitos Funcionais do GasFinder são:

- **RF 1 - Pesquisar de Postos**
 - Como: Visitante/Usuário/Administrador.
 - Quero: Pesquisar postos de determinada cidade.
 - Para: Obter informações do posto e dos preços dos combustíveis.
- **RF 2 - Cadastrar de Usuário**
 - Como: Visitante/Administrador.
 - Quero: Realizar meu cadastro ou cadastrar outro usuário.
 - Para: Ter acesso a funcionalidades que solicitem autenticação.
- **RF 3 - Marcar Pesquisa no Mapa**
 - Como: Visitante/Usuário/Administrador.
 - Quero: Visualizar no mapa a localização dos postos.
 - Para: Melhor interpretação da distância.

- **RF 4 - Detalhar Postos**
 - Como: Visitante/Usuário/Administrador.
 - Quero: Consultar mais informações sobre o posto.
 - Para: Obter informações relevantes sobre o posto.
- **RF 5 - Cadastrar Preços**
 - Como: Usuário/Administrador.
 - Quero: Cadastrar o valor dos combustíveis.
 - Para: Manter os valores atualizados.
- **RF 6 - Publicar Comentários**
 - Como: Usuário/Administrador.
 - Quero: Publicar comentários sobre o posto e preços.
 - Para: Gerar informação relevante sobre o posto e preços.
- **RF 7 – Compartilhar em Redes Sociais**
 - Como: Usuário/Administrador.
 - Quero: Compartilhar informações nas redes sociais.
 - Para: Que meus seguidores e amigos, sejam informados.
- **RF 8 - Cadastrar Postos**
 - Como: Administrador.
 - Quero: Cadastrar novos postos.
 - Para: Tornar disponível para visitantes e usuários.
- **RF 9 - Cadastro de Combustíveis**
 - Como: Administrador.
 - Quero: Cadastrar novos combustíveis que venham a surgir.
 - Para: Disponibilizar seu preço nos postos.
- **RF 10 - Consultar Usuários**
 - Como: Administrador.
 - Quero: Consultar os usuários cadastrados.
 - Para: Gerar interação com os usuários ou enviar notificações se necessário.
- **RF 11 - Aprovar Comentários**
 - Como: Administrador.
 - Quero: Aprovar comentários que tenha sido feitos aos postos.
 - Para: Manter a qualidade das informações geradas.

- **RF 12 - Cadastrar Bandeiras de Postos**
 - Como: Administrador.
 - Quero: Cadastrar novas bandeiras de postos, que venha a surgir.
 - Para: Melhorar a identificação e classificação dos postos.
- **RF 13 - Cadastrar Estados**
 - Como: Administrador.
 - Quero: Cadastrar estados para informar a localização dos postos.
 - Para: Permitir a localização detalhada do posto.
- **RF 14 - Cadastrar Cidades**
 - Como: Administrador.
 - Quero: Cadastrar cidades para informar a localização dos postos.
 - Para: Permitir a localização detalhada do posto.
- **RF 15 - Aprovar Preços**
 - Como: Administrador.
 - Quero: Aprovar preços, que tenham sido atualizados nos combustíveis.
 - Para: Garantir a qualidade das informações.

Os Requisitos Não-Funcionais (RNF), por sua vez, diz respeito às qualidades do aplicativo que visam proporcionar uma experiência de uso agradável. Segundo Paula Filho (2009, p. 190), os RNF “incluem requisitos de desempenho e outros atributos de qualidade do produto.” Para Ferreira (2008, p. 20), “desconsiderar esses fatores na definição de requisitos constitui uma das principais razões da insatisfação do usuário com o produto”.

Os Requisitos Não-Funcionais do GasFinder são:

- **RNF 1** - Ser um aplicativo WEB.
- **RNF 2** - Ter uma interface amigável.
- **RNF 3** - Os campos obrigatórios para preenchimento devem ser identificados com um asterisco no lado direito.
- **RNF 4** - Todos os campos devem possuir uma legenda.
- **RNF 5** - Ser uma fonte alternativa de consulta dos dados da ANP.

4.3 MODELAGEM

A modelagem do aplicativo foi feita através da UML, para isso foi utilizado a ferramenta CASE JUDE, especializada para projetar softwares através de diagramas UML. VISION (2012).

Na modelagem do banco de dados foi utilizado a ferramenta MER MySQL Workbench, ferramenta que permite a modelagem de dados, desenvolvimento de SQL e também integra recursos de administração de banco de dados. (WORKBENCH, 2012).

4.3.1 Diagrama de Casos de Uso

Segundo Sommerville (2007, p. 102), “os casos de uso constituem uma técnica baseada em cenários para elicitación de requisitos” e, permitem demonstrar o tipo de interação entre os objetos. Gustafson (2003, p. 22), afirmar que um caso de uso “mostra a importância do ator e as funcionalidades do sistema. Atores são representados por bonecos e as funções, por elipse. Atores estão associados as funções que eles podem executar.”

Na figura 18 é demonstrado o caso de uso geral do GasFinder que descreve as funcionalidades do aplicativo e seus relacionamentos com os atores. Define-se três atores para utilização do aplicativo: o primeiro é o visitante, com acesso apenas a consulta e comparação de preços.

O segundo é o Usuário Cadastrado, que poderá cadastrar comentários e novos preços dos combustíveis. O Visitante passa a ser um usuário do aplicativo, somente após realizar um cadastro no aplicativo, onde por sua vez, irá obter novos privilégios.

O terceiro e último ator é o Administrador, que possui acesso de administração no aplicativo. O administrador tem permissões de cadastrar novos postos, incluir cidades e estados, aprovar alteração nos preços dos combustíveis e liberar comentários publicados pelos usuários, entre outros.

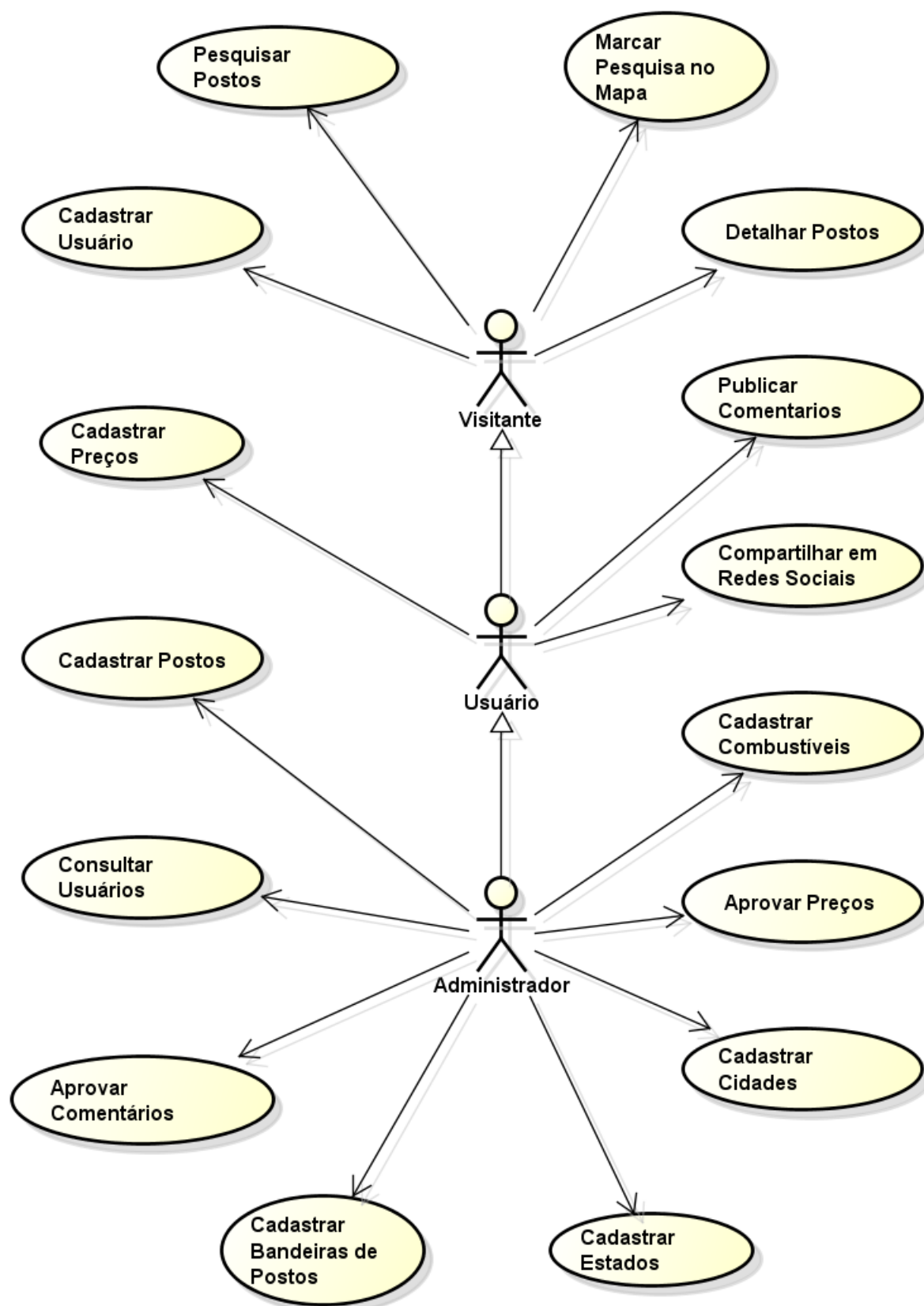


Figura 18 - Diagrama de Caso de Uso Geral

4.3.2 Diagrama de Entidade-Relacionamento

O Diagrama Entidade-Relacionamento (DER) é uma notação diagramática associada ao Modelo Entidade-Relacionamento (MER). O MER é um “modelo de dados conceitual de alto nível”, segundo Elmasri, Navathe (2005, p. 35), utilizado no projeto conceitual de aplicações de banco de dados.

DER é uma técnica de modelagem de dados, que mostra as entidades de dados, seus atributos associados e a relação entre as entidades. Para Silberschatz, Korth e Sundarshan (2006, p. 142), utiliza-se DER quando é preciso “expressar graficamente a estrutura lógica geral de um banco de dados”. Segundo Sommerville (2007, p. 118), são úteis para representar um modelo semântico de dados de um software.

Na figura 19, é demonstrado a estrutura de banco de dados do GasFinder. O MySQL foi adotado como SGDB (Sistema de Gerenciamento de Banco de Dados) neste projeto, por se tratar do banco de dados de código aberto mais popular, que permite relacionamentos entre as tabelas e garante a integridade dos dados. (MYSQL, 2012).

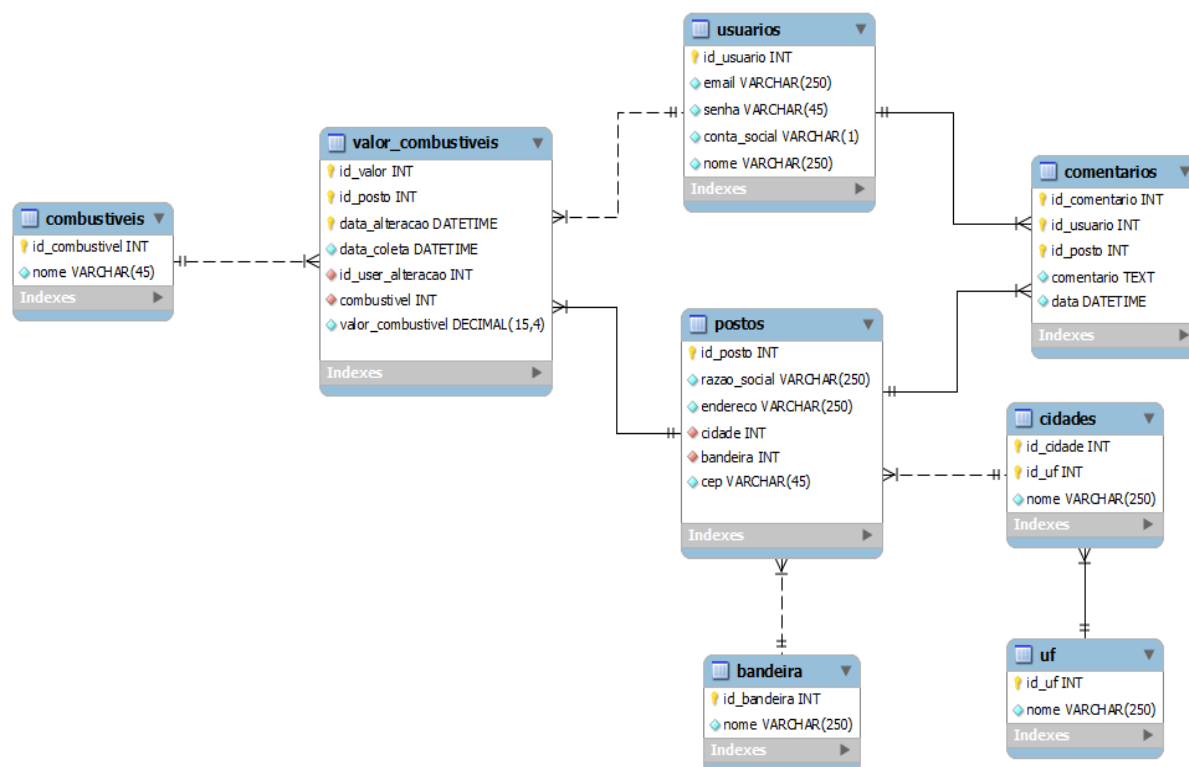


Figura 19 - Diagrama de Entidade-Relacionamento

4.3.3 Diagrama de Classe

Segundo Larman (2007, p. 266), o diagrama de classe é utilizado para “ilustrar classes, interfaces e suas associações.” Para Booch, Rumbaugh e Jacobson (2006, p. 26), “um diagrama de classe exibe um conjunto de classes, interfaces e colaborações, bem como seus relacionamentos.” Este tipo de diagrama é utilizado para modelagem dos objetos, proporcionando uma perspectiva conceitual do aplicativo e ajudando a obter uma visão panorâmica e estática da dimensão do aplicativo e do relacionamento entre os objetos.

O diagrama de classe demonstrado na figura 20 exibe a arquitetura do GasFinder, seguindo o MVC (*Model-View-Controller*).

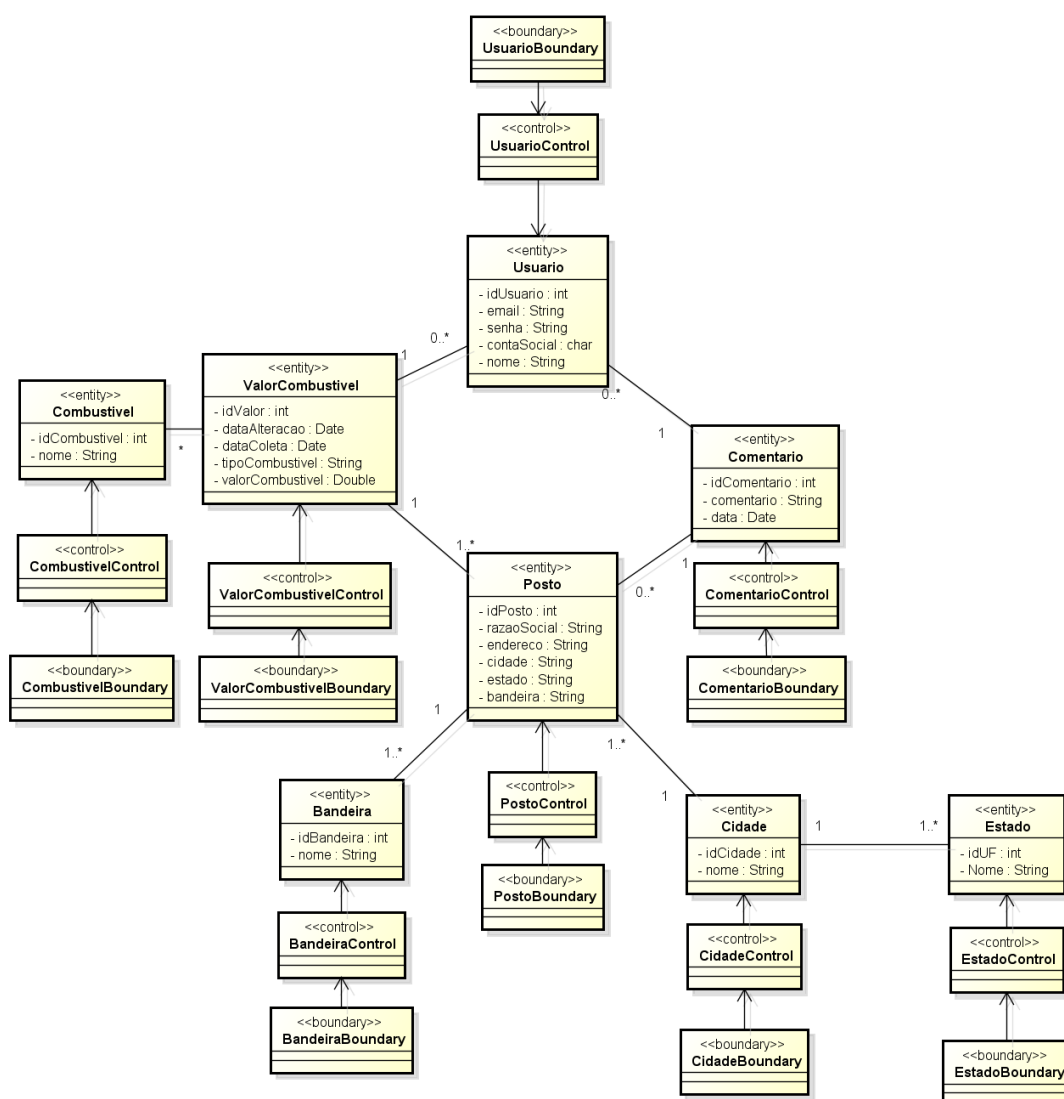


Figura 20 - Diagrama de Classe

4.3.4 Diagrama de Atividade

Segundo Larman (2007, p. 483), “um diagrama de atividade UML mostra atividades seqüenciais e paralelas em um processo.” Para Booch, Rumbaugh e Jacobson (2006, p. 268), “um diagrama de atividade é essencialmente um gráfico de fluxo, mostrando o fluxo de controle de uma atividade para outra”. É útil para mostrar o fluxo de trabalho de um software e o fluxo dos dados, visando descrever as atividades em cada operação.

Destaca-se neste trabalho, o Diagrama de Atividade: Pesquisar Postos. Considera-se a principal funcionalidade do aplicativo GasFinder, antevistas também no Diagrama Caso de Uso (figura a 18), onde pode-se notar o relacionamento desta atividade com os diferentes atores da aplicação (Visitante, Usuário Cadastrado e Administrador).

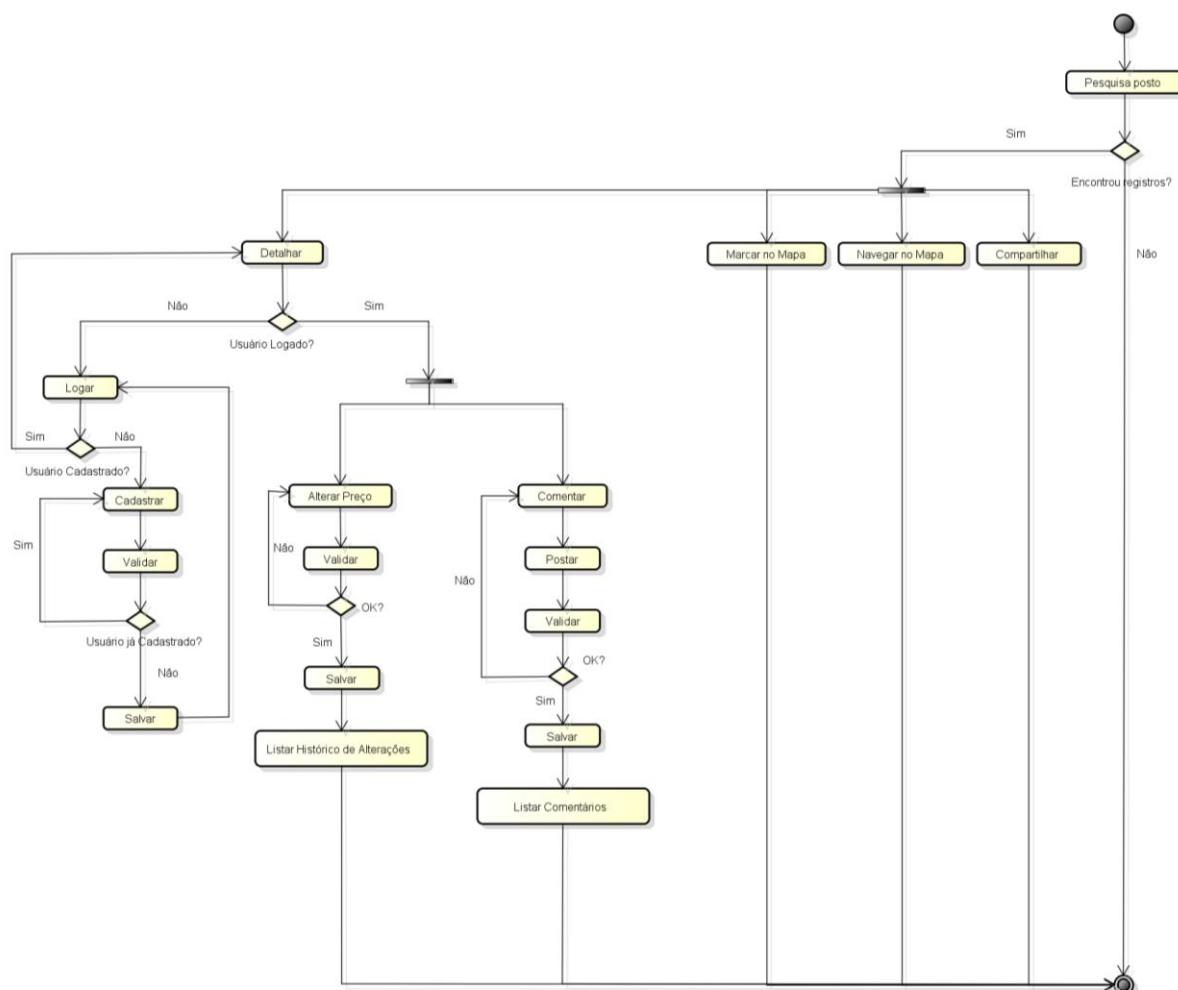


Figura 21 - Diagrama de Atividade – Pesquisar Postos

De acordo com o Diagrama de Atividade demonstrado na figura 21, quando é feita uma operação de pesquisa por postos de combustíveis, o GasFinder realiza uma busca na base de dados, pelos dados informados pelo usuário na interface do aplicativo. Ao encontrar resultados, o aplicativo exibe-os em formato de lista, em uma tela de resultados, onde é dado ao usuário, quatro opções disponíveis: Detalhar posto; Marcar posto no Mapa; Navegar no Mapa; Compartilhar informações no Twitter e Facebook.

Ao ser escolhido a opção Detalhar Posto, antes de continuar, o GasFinder verifica se o usuário havia informado seu usuário e senha previamente em outro momento no aplicativo.

Em caso positivo, será acrescentado duas opções na tela de Detalhamento do Posto: Alterar Preço do Combustível; Adicionar Comentário. Estas opções são permissões específicas para o Usuário Cadastrado e o Administrador do GasFinder. Ao selecionar-se a opção Alterar Preço do Combustível, é possível atualizar o preço do combustível do posto, onde o aplicativo irá validar se o dado informado pelo usuário está em formato numérico válido, para somente então, registrar a alteração e atualizar a lista com o histórico de preços na tela. A opção Adicionar Comentário, possibilita adicionar opiniões do usuário acerca do posto. O aplicativo valida a adição de um comentário, verificando se o campo não está vazio, registra o comentário e atualiza a lista de comentários na tela.

Em caso negativo, caso o usuário não tenha identificado-se previamente no aplicativo, o GasFinder conclui que trata-se de um usuário do tipo Visitante, e exibe a tela de Detalhamento do Posto apenas para consulta. Haverá uma opção de cadastro, que será dada para o Visitante.

4.3.5 Diagrama de Seqüência

Segundo Larman (2007, p. 198), um diagrama de seqüência é “uma notação que pode ilustrar as interações de atores e as operações iniciadas por eles.” Através deste diagrama é possível observar a troca de mensagens entre objetos em um software, suas interações e colaborações, focando na ordem em que as mensagens são enviadas e recebidas.

Os diagramas de sequência e os diagramas de comunicação - chamados de diagramas de interação - são dois dos diagramas utilizados pela UML para a modelagem dos aspectos dinâmicos de sistemas. Um diagrama de interação mostra uma interação, formada por um conjunto de objetos e seus relacionamentos, incluindo as mensagens que poderão ser enviadas entre eles. Um diagrama de sequência é um diagrama de interação que dá ênfase a ordenação temporal das mensagens; o diagrama de comunicação é um diagrama de interação que dá ênfase a organização estrutural dos objetos que enviam e recebem as mensagens. (BOOCH, RUMBAUGH E JACOBSON, 2006, p. 251).

A figura 22 é a representação gráfica do Diagrama de Seqüência: Detalhar Postos. Esta é um das operações abordadas no Caso de Uso (figura 17), e também é uma das opções, oferecidas aos usuários (Visitante, Usuário Cadastrado e Administrador) durante uma atividade de pesquisa de postos, conforme demonstrado anteriormente no Diagrama de Atividade na figura 21.

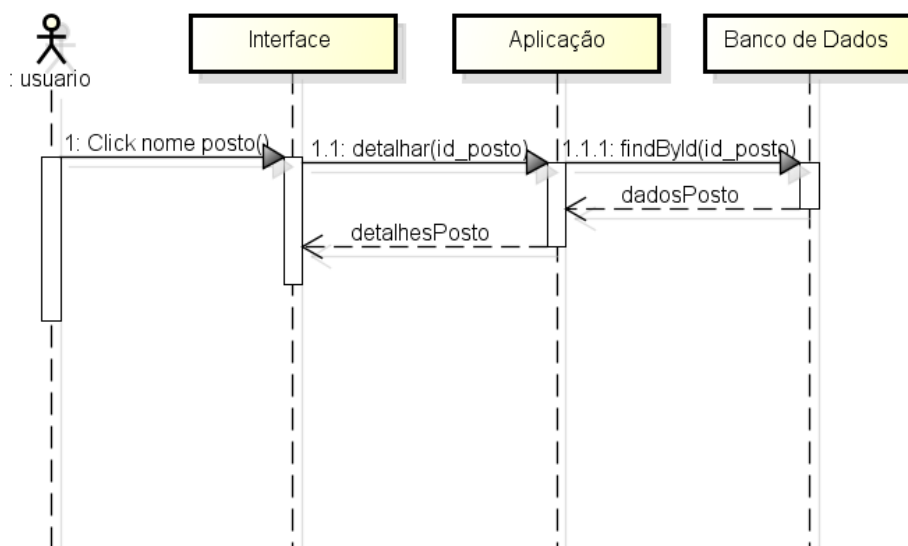


Figura 22 - Diagrama de Seqüência – Detalhar Postos

De acordo com o Diagrama de Seqüência (Detalhar Postos) demonstrado na figura 23, têm-se três objetos que interagem com o ator principal, o Usuário da aplicação, a camada de Interface de Usuário, a camada de Aplicação e a camada de Banco de Dados.

Inicialmente, o Usuário interage com a camada de Interface de Usuário (tela do aplicativo), informando o termo de pesquisa, que por sua vez aciona a camada de Aplicação, responsável pela regra de negócio e persistência, onde finalmente é acessado o Banco de Dados, que retorna os resultados encontrados para camada de Aplicação, e por fim, é exibido ao usuário na camada de Interface.

Na sessão Apêndices deste trabalho, será possível consultar outros Diagramas de Seqüência representando outras operações do Caso de Uso: Cadastrar Postos; Compartilhar em Redes Sociais; Publicar Comentários.

4.3.6 Wireframe

Segundo Memória (2005, p. 36), “*wireframes* são um rascunho de uma tela específica que posiciona a informação e a navegação, incluindo o agrupamento, ordem e hierarquia de conteúdo.” Os *wireframes* são úteis para criação de leiaute de interface de usuário, onde é possível representar as principais telas que estarão disponíveis no aplicativo e as informações que elas exibirão.

O *wireframe* da tela inicial do GasFinder (index.xhtml) é definida em três áreas principais conforme é demonstrado na figura 23. A área superior exibe o logo do GasFinder, seguido de um campo de busca e um *link* para *login* (login.xhtml), tela que concede acesso aos usuários cadastrados no aplicativo.

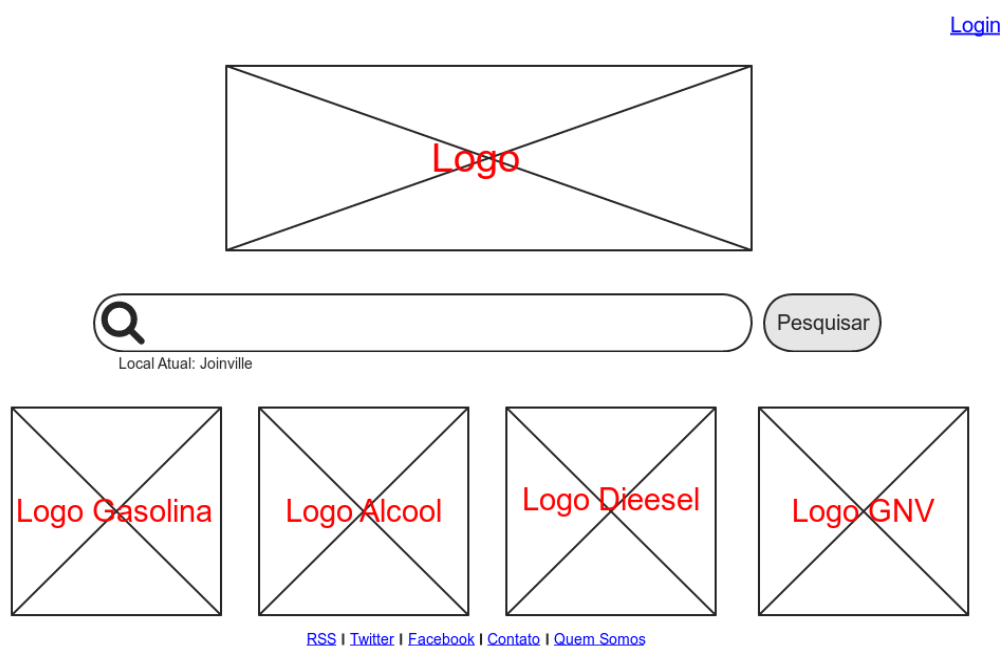


Figura 23 - Wireframe – Home

Logo abaixo, na área central, estão localizados os blocos onde serão gerados os conteúdos dinamicamente, exibindo os preços dos combustíveis, baseados na localidade de acesso do usuário, através de recurso de localização geográfica por IP. E por fim, é exibido o rodapé, que contem os links de Contato, Quem Somos, Facebook, Twitter e RSS. Os demais *wireframes* do GasFinder podem ser consultados no Apêndices deste trabalho.

4.4 FERRAMENTAS DE DESENVOLVIMENTO

As ferramentas de desenvolvimento foram adotadas baseando-se na familiaridade da equipe deste trabalho (Alexandre e William), pela solidez das respectivas ferramentas no mercado, considerando sua vasta documentação e comunidade ativa.

Como IDE (*Integrated Development Environment* ou Ambiente Integrado de Desenvolvimento) foi utilizado o Eclipse para codificação do aplicativo, que faz toda a organização dos recursos do projeto e também ajuda na produtividade de desenvolvimento, com funções de busca, recurso que auto-completa, realça código e realiza a depuração do código. “Escrita completamente em Java, a plataforma pode ser utilizada em estações de desenvolvimento que incluam os sistemas operacionais Linux, OSx e Windows”, segundo Macenas, (2006, p. 15).

Foi utilizado Java como linguagem de programação, uma tecnologia independente de plataforma, permite o desenvolvimento de software orientado a objetos e é fácil de aprender. (JAVA, 2012). Segundo Horstmann e Cornell (2001, p. 3), uma das vantagens da linguagem Java é oferecer “uma biblioteca em tempo de execução que visa proporcionar independência de plataforma: pressupõe-se que seja capaz de usar o mesmo código nos sistemas Windows 95/98/NT, Solaris, Unic e Machintosh” Java também foi adotado, tendo em vista a compatibilidade com *frameworks* utilizados neste trabalho.

Conforme é demonstrado nas camadas da figura 24, utilizou-se o Spring Framework, como responsável pela injeção de dependência em conjunto com o padrão de projetos MVC (*Model-View-Controller*). (SPRING, 2012).

Na camada de *model*, camada responsável pela persistência dos dados e interação da aplicação com o banco de dados, foi utilizado o Hibernate, um *framework* de persistência de dados desenvolvido em Java e utilizado para o mapeamento de objetos. (HIBERNATE, 2012).

Na camada de interface de usuário, camada responsável pela parte visual e de interação, foi utilizada JSF (*Java Server Faces*) em conjunto com o Primefaces, uma suíte de componentes que conta com mais de 100 componentes de interface de usuário, prontos para uso. JSF é uma tecnologia que simplifica a criação de interfaces de usuário (ORACLE, 2012). Quando somada JSF ao uso do Primefaces, é possível, por exemplo, gerar componentes de tela que compartilham objetos diretamente do JAVA com o AJAX (*Asynchronous Javascript and XML*), o que não seria possível obter facilmente, sem o uso destes *frameworks*. (PRIMEFACES, 2012).

Também foi utilizado HTML, CSS e JavaScript, para customizações visuais. Por sua vez, na camada de controle, camada responsável pela regra de negócio da aplicação e ligações entre as demais camadas, também foi utilizado a Spring Framework, já citado anteriormente.

O MySQL foi utilizado como SGDB (Sistema de Gerenciamento de Banco de Dados), para armazenar e garantir a integridade dos dados. Por ser um trabalho que engloba Java e WEB, também foi necessário utilizar um servidor de aplicação Java. Para isso foi adotado o TOMCAT, servidor WEB leve e de fácil administração, que permite disponibilizar na WEB, softwares desenvolvidos em Java. (TOMCAT, 2012).

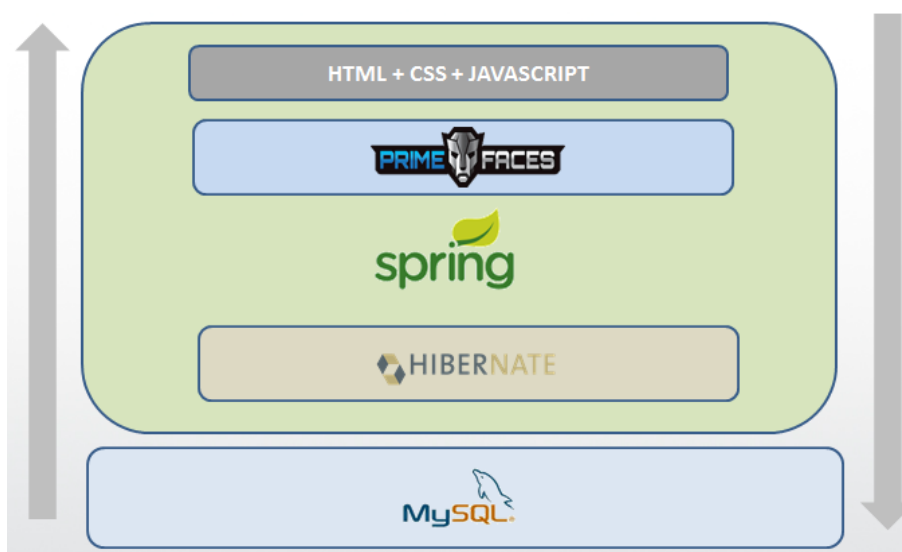


Figura 24 - Frameworks

4.4.1 Extração de Dados

Como a ANP, não exporta os dados dos preços dos combustíveis previamente formatadas em um arquivo XML, JSON ou similar, foi necessário utilizar uma técnica conhecida como *data scraping*, que segundo Carvalho (2009, p. 5), consiste na raspagem dos dados de um *website*.

Foi analisado sintaticamente o HTML do *website* da ANP e obtido todas as informações necessárias, incluindo os preços dos postos de combustíveis. Para isso, foi utilizado o serviço *online* do *website* ScrapperWiki (www.scrapewiki.com), onde desenvolvedores podem criar *scripts* para raspagem de dados com finalidades diversas, utilizando o Python, uma linguagem de programação interpretada, orientada a objetos. Os *scripts* desenvolvidos em Python, denominado *Scraper*, são executados diretamente no servidor do *website* ScrapperWiki, e são capazes de raspar dados de páginas HTML e arquivos PDF que estão acessíveis via WEB. (SCRAPER-WIKI, 2012).

Na figura 25, é exibida uma amostra do código do *Scraper*, que tem a finalidade de extrair dados de uma página HTML e adicionar em uma base de dados do serviço WEB da ScrapperWiki.

```

87 #adiciona os postos que estão no soup
88 def getPostos(soup):
89     cod_municipio = soup.find("input", { 'name' : 'municipio' })['value']
90     cod_combustivel = soup.find("input", { 'name' : 'Cod_Combustivel' })['value']
91     cod_semana = soup.find("input", { 'name' : 'cod_semana' })['value']
92     cidade = soup.find("input", { 'name' : 'desc_municipio' })['value']
93     id = 0
94     rows = soup.table.findAll("tr")
95     header_rows = rows.pop(0)
96     try:
97         for row in rows:
98             cell = row.findAll("td")
99             adicionarPosto(id, cod_municipio, cod_combustivel, cell, cidade)
100            id += 1
101     except:
102         scrape_error = str(sys.exc_info()[1])
103         print scrape_error
104         registerFail(cod_semana, cod_combustivel, cod_municipio, scrape_error)
105         return 0
106

```

Figura 25 - Amostra do código *Scraper* desenvolvido em Python

Segundo a figura 26, o *Scraper* irá extrair os dados do *website* da ANP, enviar os dados para armazenamento no banco de dados do GasFinder onde finalmente os dados serão disponibilizados ao usuário final do GasFinder.

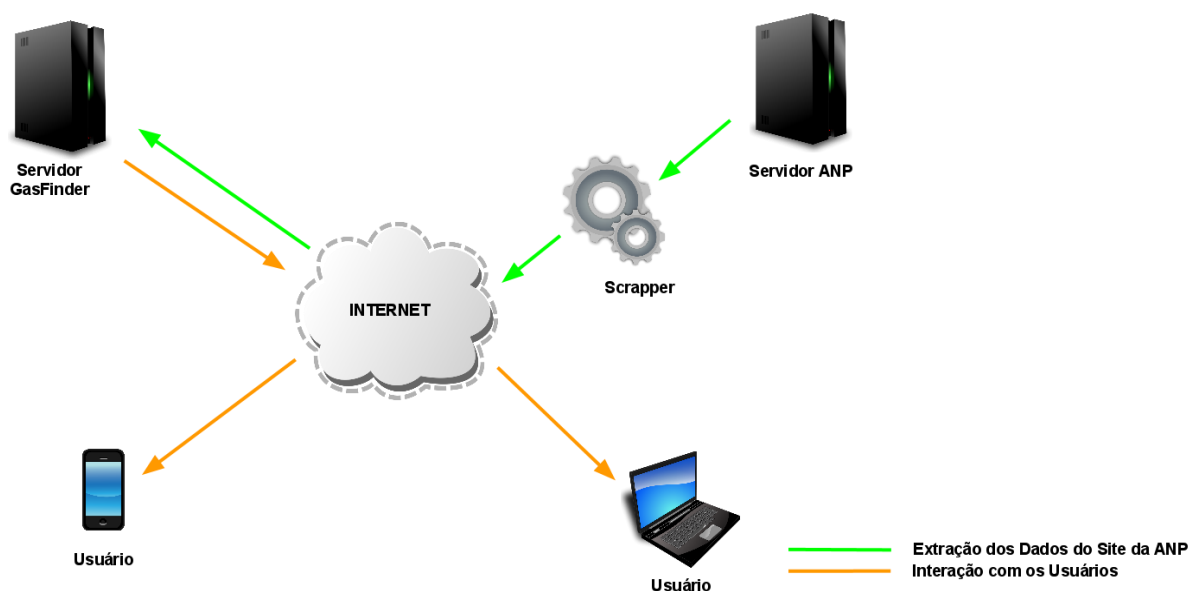


Figura 26 - Fluxo de informações do GasFinder

4.5 AMOSTRA DO APLICATIVO GASFINDER

O resultado do desenvolvimento do GasFinder, o detalhamento das telas e explicação das funcionalidades será descrito nos sub-tópicos seguintes.

4.5.1 Tela Inicial

O aplicativo GasFinder foi construído em Java, para navegação do mesmo, baseia-se na tela inicial `index.xhtml`, esta tela será carregada durante o primeiro acesso ao aplicativo. A tela inicial disponibiliza no topo, um *link* para *login* (`login.xhtml`) destinado aos usuários cadastrados e administradores da aplicação. Quando um usuário administrador estiver identificado, a tela inicial (`index.xhtml`) mostrará no topo dois *links*, Administração e Admin.

No corpo da tela inicial (index.xhtml), é exibido quatro blocos em destaque, posicionados na horizontal e alinhados no centro da tela. Cada bloco exibe o preço de um tipo combustível, Gasolina, Etanol, diesel e GNV, da esquerda para direita. Os valores exibidos são calculados conforme o preço médio do respectivo combustível, baseado na localidade de acesso do usuário, que é consultado via recurso de localização geográfica por ip.

No campo de busca, o usuário pode preencher a localidade onde deseja consultar e comparar os preços dos combustíveis. O GasFinder consulta 4 tipos de informações sobre os Postos, que podem ser utilizados durante a pesquisa:

1. Cidade;
2. Estado;
3. Endereço (Rua);
4. Nome do Posto.

Ao selecionar o botão “buscar”, o aplicativo realizará uma busca do valor informado na tela, retornando os registros encontrados para uma tela de resultados da pesquisa (pesquisa.xhtml).

Na figura 27 é demonstrada a interface da tela inicial do GasFinder, contendo os preços calculados automaticamente para a cidade de Joinville.

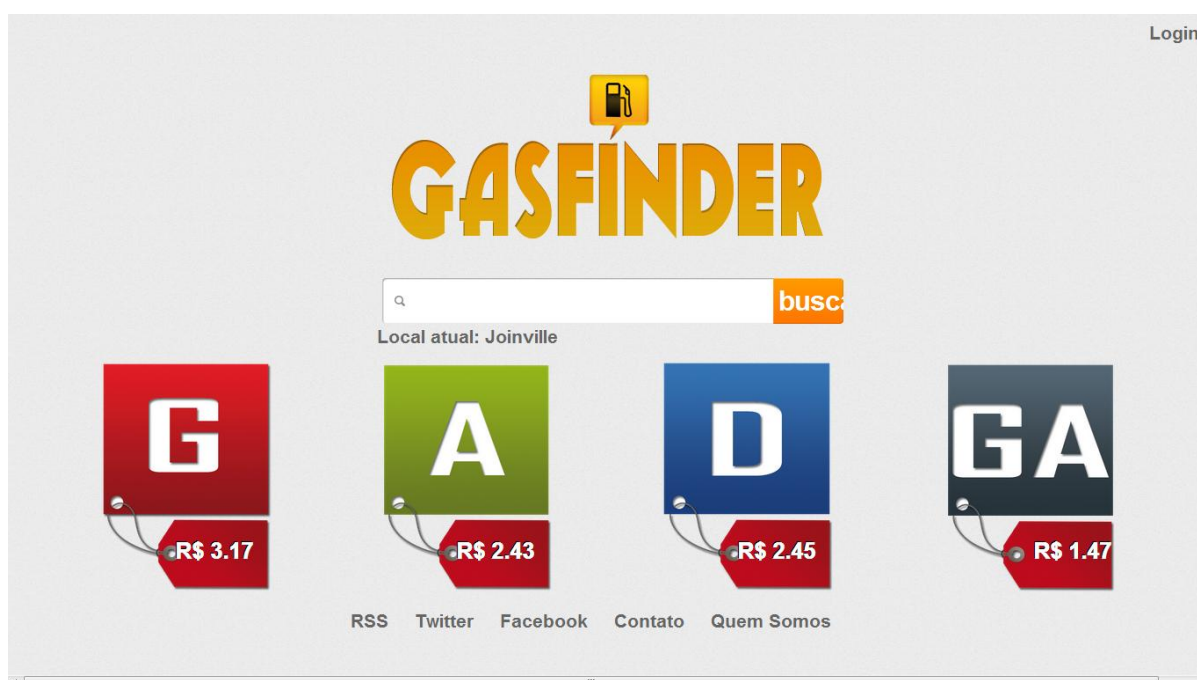


Figura 27 - GasFinder - Tela Inicial

4.5.2 Tela de Resultado de Pesquisa

A tela de resultado de pesquisa (pesquisa.xhtml) é uma tela onde é exibido o resultado da busca. O resultado inicial, é exibido em forma de lista, onde em negrito e em azul é exibido o nome do posto e em seguida, os valores dos combustíveis disponibilizados pelo respectivo posto. Em segundo plano, na linha posterior em verde é exibido o endereço do posto. Logo abaixo é organizado as opções de compartilhamento (Facebook e Twitter) e marcação no mapa, onde o usuário poderá marcar e visualizar o respectivo posto em um mapa.

Além do resultado da busca inicial (em formato de lista), é possível exibir todos os postos marcados em um mapa, proporcionando uma nova experiência para o usuário e auxiliando na localização geográfica dos postos. Para isso, basta selecionar a aba “Mapa”.

Na figura 28 é demonstrada a interface da tela de resultado de pesquisa, contendo os resultados em lista para uma busca por postos de combustíveis realizados meramente como exemplo, na cidade de Rio Branco.



Figura 28 - GasFinder - Tela de Resultado de Pesquisa

4.5.3 Tela de Busca no Mapa

Na tela de busca no mapa (pesquisa.xhtml) são exibidos os resultados de uma pesquisa através de uma representação visual da localidade onde os postos estão localizados. Os postos são marcados através de uma bandeira, sinalizando ao usuário a localização dos postos de combustíveis.

Quando o usuário seleciona a opção “Mapa” disponível logo abaixo de cada resultado de pesquisa em formato de lista, a tela de busca no mapa destaca o posto selecionado, aplicando a cor laranja no ícone do respectivo posto, facilitando a identificação para o usuário. Ao manter o ponteiro do *mouse* sobre uma bandeira de um posto, é exibida uma legenda como o nome do respectivo posto.

O recurso de mapa adotado para esta funcionalidade é disponibilizado gratuitamente pelo Google Maps. O mapa deste fornecedor possui algumas funcionalidades de interação, permitindo deslocar o mapa para os lados, para cima e para baixo, aproximar ou afastar a visualização permitindo maior detalhamento dos gráficos e das ruas.

Na figura 29 é demonstrada a interface da tela de busca no mapa.

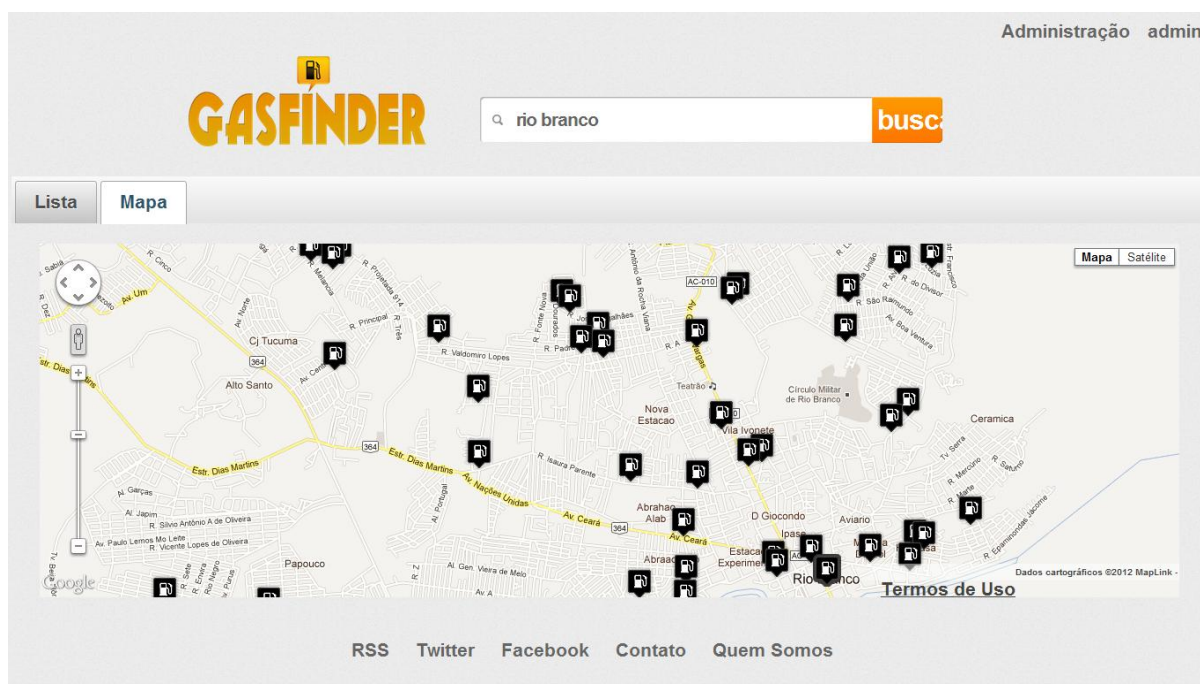


Figura 29 - GasFinder - Tela de Busca no Mapa

4.5.4 Tela de Login

A tela de *login* (login.xhtml) é destinada a realizar a identificação de usuários cadastrados. O aplicativo permite acesso de dois perfis de usuário, Usuário Cadastrado e administrador. Ao informar o nome de usuário e a senha no formulário de *login*, o aplicativo verifica se as informações preenchidas em tela constam na base de dados e analisa se são verdadeiras.

O Administrador possui acesso total ao aplicativo, onde é possível cadastrar novas cidades e estados, alterar os valores dos preços dos combustíveis, cadastrar novos combustíveis, aprovar comentários, consultar usuários, entre outros. Por sua vez, o Usuário Cadastrado possui privilégios adicionais comparados ao visitante. Quando um visitante utiliza a opção “cadastre-se” na tela de cadastro (cadastroUsuario.xhtml), o mesmo poderá adicionar comentários sobre os postos dos combustíveis, e inclusive, alterar os preços dos combustíveis.

A tela de login (login.xhtml) mantém um *link* chamado “Home”, que direciona para a tela inicial do GasFinder (index.xhtml). Na figura 30 é demonstrada a interface da tela de login.

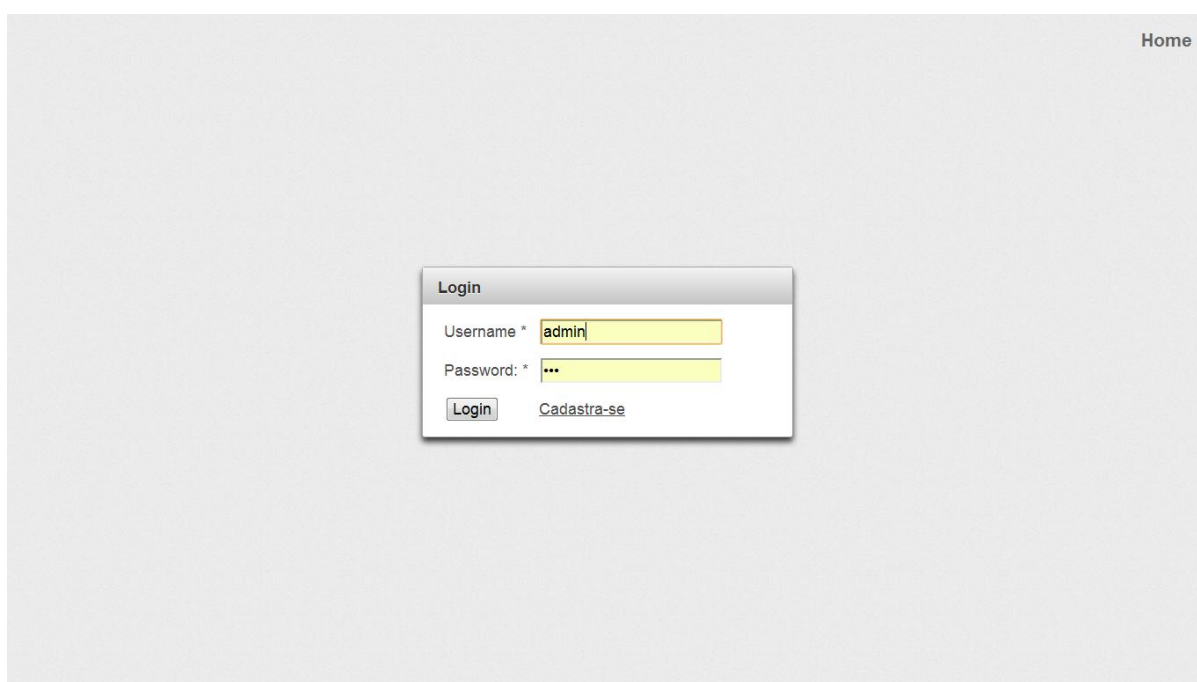


Figura 30 - GasFinder - Tela de Login

4.5.5 Tela de Detalhamento do Posto

Na tela de detalhamento do posto (posto.xhtml) é possível realizar a atualização dos preços dos combustíveis de cada posto. Esta tela mantém um histórico dos preços praticados pelo respectivo posto de combustível, assim como as alterações realizadas manualmente por Usuários Cadastrados e pelo Administrador. Para alterar as informações de preços de um combustível, basta o Usuário Cadastrado ou Administrador selecionar o texto do combustível desejado, e imediatamente o aplicativo irá atualizar a tela para o modo de edição, permitindo que usuário informe os novos valores. Para finalizar, basta o usuário selecionar a opção “Salvar”. Nesta tela também é possível consultar e adicionar comentários sobre o posto.

Somente um Usuário Cadastrado ou um Administrador pode atualizar os preços dos combustíveis e adicionar comentários. O Visitante terá acesso a tela de detalhamento do posto somente para consulta. Caso o visitante tenha interesse em editar as informações de um posto, deverá realizar o cadastro prévio através da tela de login (login.xhtml), utilizando a opção “Cadastre-se”.

Na figura 31 é demonstrada a tela de detalhamento do posto.

BR Auto Posto Rio Branco Derivados de Petróleo Ltda

Home Administração admin

G R\$ 2.85 **A** R\$ 2.00 **GA** R\$ 1.69 **D** R\$ 2.05

Histórico de Alterações

Data Alteração	Gasolina	Etanol	GNV	Diesel
2011-12-10 00:00:00.0	2.8490	1.9990	1.6900	2.0480
1969-12-31 21:00:00.0	2.8600	2.0000	1.6900	2.0500
2011-12-12 19:51:43.0	2.8700	2.0000	1.6900	2.0500
2011-12-16 18:06:52.0	2.8500	2.2300	1.6900	2.0500
2011-12-18 20:56:16.0	2.8500	2.0000	1.6900	2.0900
2011-12-20 14:52:47.0	2.8500	2.0400	1.6900	2.0500
2011-12-20 16:33:12.0	2.8500	2.0700	1.6900	2.0500

Comentários

2011-12-10 17:37:15.0 Lorem Ipsum é simplesmente uma simulação de texto da indústria tipográfica e de

Figura 31 - GasFinder - Tela de Detalhamento do Posto

4.5.6 Tela de Contato

Na tela de contato (contato.xhtml) é oferecido aos visitantes e usuários um espaço para enviar mensagens para a equipe de administração do GasFinder (sendo a equipe deste trabalho, Alexandre e William). Oportunamente, poderá ser utilizado para enviar sugestão de melhoria, aviso de erros ou reclamações sobre o serviço do GasFinder.

Para enviar uma mensagem, o visitante ou usuário deverá informar no formulário seu e-mail e em seguida, sua mensagem. Finalmente bastará selecionar a opção “Enviar” para que a mensagem seja encaminhada para a equipe de administração do GasFinder. Na figura 32 é demonstrada a tela de contato.

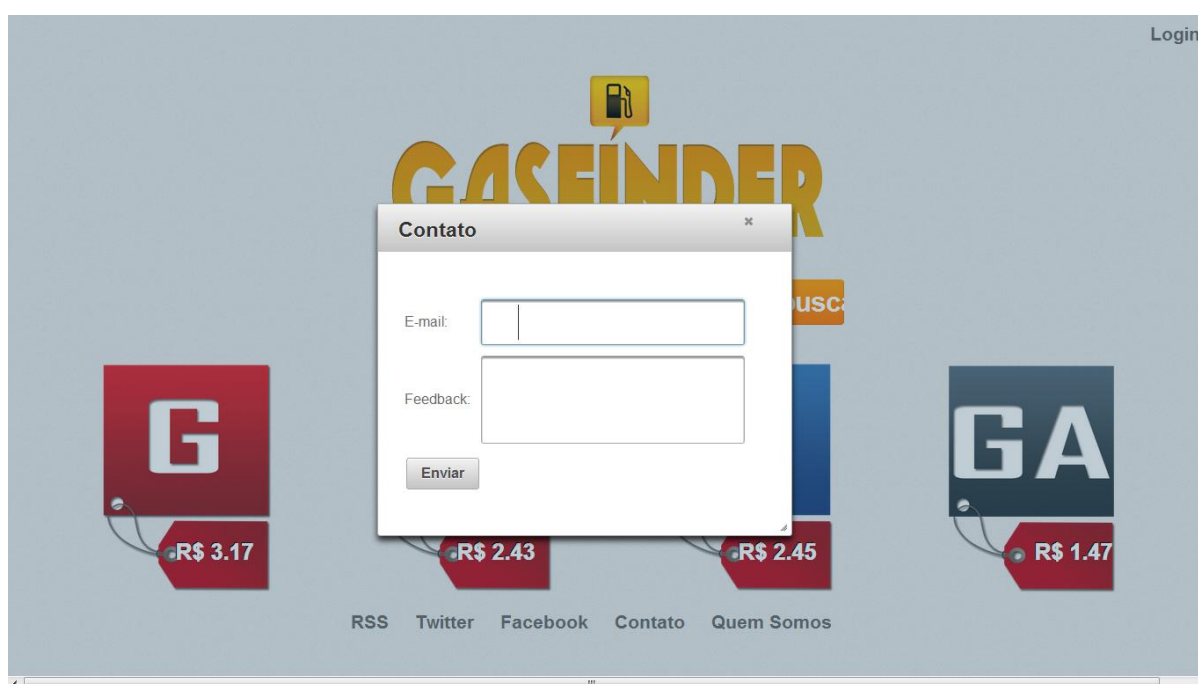


Figura 32 - GasFinder - Tela de Contato

Neste capítulo foi possível expor as etapas de desenvolvimento, incluindo a análise dos requisitos até a amostra de telas do aplicativo, onde foi abordado os aspectos de interação e funcionalidades de cada tela. A seguir, será abordado as considerações finais do trabalho, a fim de situar o nível de completude dos objetivos propostos neste trabalho.

4.6 CONSIDERAÇÕES FINAIS DO TRABALHO

Durante a realização deste trabalho, assumiu-se como desafio, a amostra de um aplicativo proposto para consulta de preços de combustíveis *online*. Pode-se considerar, que foi atingido os objetivos gerais esperados na proposta deste trabalho, concretizado no desenvolvimento do GasFinder.

Entretanto, acerca dos objetivos específicos, exclusivamente sobre a Fase de Desenvolvimento do GasFinder, pode-se considerar que foi parcialmente atingida. Durante o desenvolvimento do aplicativo, foi necessário realizar um replanejamento, devido a indisponibilidade de horas para o projeto, que limitou a implementação de todas as funcionalidades previamente planejadas. Ao fim do desenvolvimento, as funcionalidades disponíveis no GasFinder ficou apenas sensivelmente abaixo da totalidade dos requisitos estimados para a aplicação, sendo que estas limitações não afetam o uso pelos usuários público-alvo desta aplicação, restringindo-se apenas ao Administrador.

Através da análise das qualidades de usabilidade do Sistema de Levantamento de Preços da ANP, permitiu-se identificar restrições, que foram potencializadas em diferenciais aplicados ao GasFinder. Deste modo, o GasFinder situa-se plenamente capaz, de ser uma fonte alternativa para consulta e comparação de preços de combustíveis *online*, para o consumidor final.

Pode-se afirmar, que o uso da Engenharia WEB, foi indispensável para o desenvolvimento assertivo do aplicativo. Com o uso de protótipos de *wireframes*, permitiu-se testar os requisitos antecipadamente, evitando retrabalho.

Considera-se, de maneira geral, que um grande facilitador no desenvolvimento deste trabalho, esteve no uso de tecnologias de desenvolvimento WEB, com as quais a equipe (Alexandre e William) já estava familiarizada. Outro ponto importante foi o uso de *frameworks* de desenvolvimento, que oferecem componentes de interface e persistência, o que possibilitou a equipe focar na regra de negócio da aplicação, acelerando o desenvolvimento do GasFinder.

O uso de algumas boas práticas das Metodologias Ágeis, Scrum e XP, conceberam o cenário ideal para o desenvolvimento da aplicação, flexível e formal na medida ideal.

Finalmente, cabe ressaltar duas sugestões de implementação, para quem interessar basear-se no GasFinder para desenvolvimento de aplicações com objetivos similares. Sugere-se primeiramente, a criação de novas funcionalidades para integração com mídias sociais, Facebook e Twitter. Frequentemente estes serviços adicionam novos recursos de integração e possivelmente no futuro, poderá haver novas funcionalidades oportunas; Por fim, a segunda sugestão é a disponibilização de novas informações sobre os postos de combustíveis, como fotos e outros serviços de conveniência, que podem ser úteis para consulta pelo usuário final.

5 CONCLUSÃO

O desenvolvimento de software tem evoluindo constantemente e está se tornando um processo cada dia mais dinâmico, oneroso, mas também avançado, com objetivo de se adequar as necessidades do mercado, rapidamente. Contudo, apesar de uma grande evolução na criação e disponibilidade de ferramentas e metodologias a disposição dos engenheiros de software, quando se trata de desenvolvimento de software para WEB, ainda existem limitações e falta de exatidão no processo de desenvolvimento.

A Engenharia WEB, sobretudo, foi uma resposta na tentativa de oferecer novos parâmetros e ferramentas, que possibilitem aos engenheiros, criar software para WEB, com a mesma qualidade, confiabilidade e nível de modelagem que a Engenharia de Software convencional oferece.

Por sua vez, as metodologias classificadas como “ágeis”, como o XP e o Scrum, também vem ganhando espaço no âmbito de planejamento de software, possibilitando que o cliente final esteja muito próximo do processo e do desenvolvimento, permitindo portando, o acompanhamento de toda a evolução do projeto. Através de entregas planejadas (*sprints*), o cliente pode determinar a qualquer momento se a aplicação está alinhada com os requisitos, evitando retrabalho e ganhando mais precisão nas horas de projeto planejadas e executadas.

Na ausência de qualidades de usabilidade abordadas acerca da ANP e do Sistema de Levantamento de Preços (SLP), pode-se verificar que a usabilidade é um fator decisivo para se obter usuários adeptos de um serviço. É uma boa medida considerar a usabilidade no cronograma de desenvolvimento de um software.

Durante o desenvolvimento do trabalho, foi constatada uma dificuldade em manter cronograma de acordo com a mensuração de esforço das atividades de desenvolvimento, sendo necessário um replanejamento durante a Fase de Desenvolvimento. Se fez necessário reduzir de escopo do projeto, limitando as funcionalidades que seriam implementadas no aplicativo. Embora o aplicativo final, tenha ficado apenas sensivelmente aquém das expectativas, nota-se como lição aprendida, que mesmo as equipes mais seguras, capacitadas e competentes, está a mercê de imprevistos, que podem desviar o planejamento dos objetivos iniciais.

A favor deste trabalho, esteve a facilidade na implementação do aplicativo, pois foi adotado tecnologias com a qual a equipe (Alexandre e William) já possuía conhecimento, principalmente a linguagem de programação Java, Python e o banco de dados MySQL. O uso de *frameworks* de desenvolvimento, contribuíram significativamente para produtividade no desenvolvimento da aplicação, dispensando esforço adicional, na construção de módulos que não iriam agregar vantagens diretas para o usuário final. Como lição aprendida, pode-se citar que as equipes de desenvolvimento, devem sinalizar sempre que possível para os gestores do projeto, a necessidade de considerar horas no cronograma do projeto, para atividades de pesquisa, onde é possível encontrar tecnologias que permitam o desenvolvedor focar na regra de negócio durante o desenvolvimento do software.

Por fim, o GasFinder atingiu a proposta inicial deste trabalho, concretizando-se em um serviço capaz de importar os preços dos combustíveis disponíveis no Sistema de Levantamento de Preços da ANP, e disponibilizar ao consumidor final, em uma interface contendo ferramentas de busca, localização geográfica por ip e compartilhamento de informações nas mídias sociais, Twitter e Facebook. O GasFinder, tornou-se efetivamente, uma fonte alternativa de consulta de preços de combustíveis para o consumidor final, utilizando todo o conteúdo fornecido com autorização da ANP, por sua vez, apresentado em uma interface de usuário que oferece uma experiência de navegação moderna.

Não será dada continuidade no desenvolvimento do GasFinder após encerramento deste trabalho. A fim de contribuir com possíveis interessados na implementação de novas aplicações baseadas no GasFinder, sugere-se a inclusão de informações adicionais sobre os postos de combustíveis, como fotos, descrição de produtos e outros serviços oferecidos nos postos, como serviços de conveniência. O uso de mídias sociais, como Twitter e Facebook para divulgação dos preços dos combustíveis também é um item que pode ser explorado ainda mais, pois já é uma aplicação comum na rotina dos usuários de internet e pode auxiliar na divulgação de um novo aplicativo que venha a ser criado por terceiros.

Espera-se que este trabalho, torne-se um referencial útil para contribuir com pesquisas acerca de Metodologias Ágeis, Usabilidade, Engenharia de Software e Engenharia Web, *Frameworks* e tecnologias de desenvolvimento WEB.

REFERÊNCIAS

ADAMI, Virgílio et al. **Analogia entre Scrum e PMBOK**. Santa Catarina: FGV, 2007. 87 p. Trabalho de Conclusão de Curso - MBA em Gerência de Projetos, Fundação Getúlio Vargas, Santa Catarina, 2007.

ANP, Agência Nacional do Petróleo, Gás Natural e Biocombustíveis. **Folder Institucional da ANP - Português**. 2011. Disponível em: <<http://www.anp.gov.br/SITE/acao/download/?id=2436&cachebust=1328732208773>>. Acesso em: 18 de Jan. 2012.

_____. **Metodologia utilizada para realização da pesquisa de preços no âmbito do Levantamento de Preços e de Margens de Comercialização de Combustíveis da ANP: Pesquisa de Preços de Combustíveis em Postos de Revenda**. 2010. Disponível em: <<http://www.anp.gov.br/SITE/acao/download/?id=41567>>. Acesso em: 18 de Jan. 2012.

_____. **Sistema de Levantamento de Preço (SLP)**. 2012. Disponível em: <www.anp.gov.br/preco>. Acesso em: 18 de Jan. 2012.

BALLE, Andrea Raymundo. **Análise de Metodologias Ágeis: Conceitos, Aplicações e Relatos sobre XP e Scrum**. Porto Alegre: UFRGS, 2011. 79 p. Trabalho de Conclusão de Curso (Graduação) – Ciência da Computação, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2011. Disponível em: <<http://www.lume.ufrgs.br/bitstream/handle/10183/31028/000782065.pdf?sequence=1>>. Acesso em: 25 de Jan. 2012.

BIAZOTTO, Ana Lidia. **Uma avaliação dos impactos potenciais do auxílio-gás**. Ribeirão Preto: USP, 2004. 70 p. Trabalho de Conclusão de Curso (Graduação) - Curso de Economia, Administração e Contabilidade, Universidade de São Paulo, Ribeirão Preto, 2004. Disponível em: <http://www.anp.gov.br/CapitalHumano/Arquivos/PRH04/Ana-Lidia-Biazotto_PRH04_USP_G.pdf>. Acesso em: 13 de Jan. 2012.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML - Guia do Usuário: O mais avançado tutorial sobre Unified Modeling Language (UML), elaborado pelos próprios criadores da linguagem.** Rio de Janeiro: Elsevier, 2006.

BRIDGE RESEARCH; VOLTAGE. **Percepção do cliente.** 2010. Disponível em: <<http://www.empreendedor.com.br/reportagens/consumidor-contempor%C3%A2neo-busca-valores-humanos-nos-produtos>>. Acesso em: 15 de Jan. 2012.

BRITO. Lício Sérgio Gerraz de. **WEBSCHARTS: Uma Ferramenta de Desenvolvimento de Aplicações WEB Baseada no HMBS/M.** 96 p. Campo Grande: UFMG, 2003. Dissertação (Mestrado) - Pós-Graduação Ciência da Computação, Universidade Federal do Mato Grosso do Sul, Campo Grande, 2003. Disponível em: <<http://www.dct.ufms.br/mestrado/dissertacoes/2003/licio.pdf>>. Acesso em: 13 de Jan. 2012.

CARVALHO, Alberto. **Scraped Data and Sticky Prices: Frequency, Hazards, and Synchronization.** EUA: Harvard University, p. 5, 2009. Disponível em: <http://isites.harvard.edu/fs/docs/icb.topic643207.files/09_0915_Cavallo.pdf>. Acesso em: 13 de Jan. 2012.

CHAVES, Leandro Costa. **Espalhe: Buzzmarketing como Ferramenta de Comunicação.** Fortaleza: FA7, 2007. 56 p. Monografia (Bacharelado) – Comunicação Social, Publicidade e Propaganda - Faculdade 7 de Setembro, Fortaleza, 2007. Disponível em: <http://www.fa7.edu.br/recursos/imagens/File/publicidade/monografia/2007/mon_o-chaves.pdf>. Acesso em: 18 de Jan. 2012.

EBERT, Cassiano. **Uma proposta para o gerenciamento ágil de projetos baseada em sua complexidade.** Recife: UFPE, 2009. 133 p. Dissertação (Mestrado) – Pós-Graduação em Ciência da Computação - Universidade Federal de Pernambuco, Recife, 2009. Disponível em: <<http://www.projetodiario.com.br/download/category/26-uma-proposta-para-o-gerenciamento-agil-de-projetos-baseada-em-sua-complexidade/190-proposta-para-o-gerenciamento-agil-de-projetos-baseada-em-sua-complexidade>>. Acesso em: 17 de Jan. 2012.

ELMASRI, Ramez; NAVATHE, Shamkant B. **Sistemas de banco de dados**. 4ª ed. São Paulo: Pearson Education do Brasil, 2005.

FERNANDES, Camila Formozo. **A Evolução da Arrecadação de Royalties do Petróleo no Brasil e seu Impacto sobre o Desenvolvimento Econômico do Estado do Rio de Janeiro**. Rio de Janeiro: UFRJ, 2007. 72 p. Monografia (Bacharelado) - Economia, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2007. Disponível em: <http://www.dnc.gov.br/CapitalHumano/Arquivos/PRH21/Camila-Formozo-Fernandes_PRH21_UFRJ_G.pdf>. Acesso em: 11 de Jan. 2012.

FERREIRA, Aurélio Buarque de Holanda. **Novo dicionário Aurélio da língua portuguesa**. 4ª ed. Curitiba: Positivo, 2009.

FERREIRA, Simone Bacellar Leal. **e-Usabilidade**. Rio de Janeiro: LTC, 2008.

GONÇALVES, Rodrigo Franco. **Uma abordagem sistemática para o processo de produção em Engenharia WEB, na fase de concepção**. São Paulo: USP, 2010. p 140. Tese (Doutorado) – Engenharia da Produção, Escola Politécnica da Universidade de São Paulo, São Paulo, 2010. Disponível em: <http://www.teses.usp.br/teses/disponiveis/3/3136/tde-08122011-100840/publico/Tese_Rodrigo_Franco_Goncalves.pdf>. Acesso em: 12 de Jan. 2012.

GUSTAFSON, David A. **Teoria e problemas de engenharia de software**. Porto Alegre: Bookman, 2003.

HIBERNATE. **Hibernate - JBoss Community**. 2012. Disponível em: <<http://docs.jboss.org/hibernate/core/4.0/quickstart/en-US/html/hibernate-gsg-preface.html>>. Acesso em: 13 de Jan. 2012.

HORSTMANN, Cay S.; CORNELL, Gary. **Core Java 2: Fundamentos. Volume 1**. São Paulo: Alta Books, 2001.

IBOPE, Instituto Brasileiro de Opinião Pública e Estatística. **IBOPE Mídia traça o perfil do e-commerce brasileiro.** 2010. Disponível em: <<http://www.ibope.com.br/calandraWeb/servlet/CalandraRedirect?temp=5&proj=PortalIBOPE&pub=T&db=caldb&comp=Noticias&docid=A6E4E55AB142CF2E832577F30061438B>>. Acesso em: 13 de Jan. 2012.

JAVA. **New to Java Programming Center.** 2012. Disponível em: <<http://www.oracle.com/technetwork/topics/newtojava/overview/index.html>>. Acesso em: 12 de Jan. 2012.

JÚNIOR, Jorge Antônio Pedroso. **A internacionalização das national oil companies e o direito internacional.** Rio de Janeiro: UERJ, 2008. 191 p. Dissertação (Mestrado) – Pós-Graduação em Direito, Universidade do Estado do Rio de Janeiro, Rio de Janeiro, 2008. Disponível em: <http://www.anp.gov.br/site/extras/prh/docs/ANP_10anos/PRH_33.pdf>. Acesso em: 17 de Jan. 2012.

KRUG, Steve. **Não me faça pensar.** São Paulo: Alta Books, 2006.

LARMAN, Craig. **Utilizando UML e Padrões.** 3ª ed. Porto Alegre: Bookman, 2007.

LIMA, Haroldo. **Pré Sal, Desenvolvimento e Soberania.** São Paulo, 2011. Disponível em: <<http://www.anp.gov.br/SITE/acao/download/?id=57407>>. Acesso em: 13 de Jan. 2012.

MACENAS, Ivan. **Eclipse 3.1 Programando com Visual Editor.** 2ª ed. Rio de Janeiro: Alta Books, 2006.

MEMÓRIA, Felipe. **Design para a Internet: Projetando a experiência perfeita.** Rio de Janeiro: Elsevier, 2005.

MONZILLA, Firefox. **Bloqueador de janelas popup.** 2012. Disponível em: <<http://br.mozdev.org/firefox/popup>>. Acesso em: 15 de Jan. 2012.

MYSQL. **MySQL: The world's most popular open source database.** 2012. Disponível em: <<http://dev.mysql.com/doc/refman/5.6/en/introduction.html>>. Acesso em: 14 de Jan. 2012.

NETO, Erasmo Isotton. **Ferramenta Educacional para Ensino de Práticas do SCRUM.** Porto Alegre: PUCRS, 2008. 80 p. Trabalho de Conclusão de Curso (Bacharelado) - Sistemas de Informação - Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, 2008. Disponível em: <<http://www.inf.pucrs.br/~rafael/Scrumming/Scrumming.pdf>>. Acesso em: 13 de Jan. 2012.

NIELSEN, Jakob. **Projetando Websites.** Rio de Janeiro: Elsevier, 2000.

OBERDERFER, Saulo Bazzi. **Engenharia de WebAPP para um grupo de escoteiro utilizando o método OOHDM.** Chapecó: UNOCHAPECÓ, 2005. 176 p. Trabalho de Conclusão de Curso - Ciência da Computação, Fundação Universitária do Desenvolvimento do Oeste, Chapecó, 2005. Disponível em: <http://www.unochapeco.edu.br/saa/tese/2400/Saulo_Bazzi_Oberderfer_TCCII.pdf>. Acesso em: 12 de Jan. 2012.

ORACLE. **JavaServer Faces Technology.** 2012. Disponível em: <<http://www.oracle.com/technetwork/java/javaee/javaserverfaces-139869.html>>. Acesso em: 18 de Jan. 2012.

PAULA FILHO, Wilson de Pádua. **Engenharia de Software: Fundamentos, Métodos e Padrões.** 3ª ed. Rio de Janeiro: LTC, 2009.

PLANALTO - Congresso Nacional. **Art. 1º, Lei nº 8.137, de 27 de dezembro de 1990.** 1990. Disponível em: <http://www.planalto.gov.br/ccivil_03/leis/L8137.htm>. Acesso em: 13 de Jan. 2012.

PLANALTO - Congresso Nacional. **Art. 8º, Lei nº 9.478, de 6 de agosto de 1997.** 1997. Disponível em: <http://www.planalto.gov.br/ccivil_03/leis/L9478.htm>. Acesso em: 15 de Jan. 2012.

PRESSMAN, Roger S. **Engenharia de Software. 6ª ed.** São Paulo: McGraw-Hill, 2006.

_____. **Software engineering: a practitioner's approach. 7ª ed.** Nova York: McGraw-Hill, 2010.

PRESSMAN, Roger S.; LOWE, David. **Engenharia WEB.** Rio de Janeiro: LTC, 2009.

PRIMEFACES. **Primefaces.** 2012. Disponível em: <<http://www.primefaces.org/showcase-labs/ui/home.jsf>>. Acesso em: 13 de Jan. 2012.

PROCOM, Órgão de Proteção e Defesa do Consumidor. **Consumidor deve ficar atento aos seus direitos na hora de abastecer.** 2011. Disponível em: <<http://www.procon.pr.gov.br/modules/noticias/article.php?storyid=179&tit=Consumidor-deve-ficar-atento-aos-seus-direitos-na-hora-de-abastecer>>. Acesso em: 15 de Jan. 2012.

SCRAPERWIKI. **Refine, reuse and request data | ScraperWiki.** 2012. Disponível em: <<https://scraperwiki.com/about>>. Acesso em: 11 de Jan. 2012.

SEFAZ, Secretaria da Fazenda do Estado de Santa Catarina. **Fazenda fecha o cerco sobre a sonegação no setor de combustíveis.** Santa Catarina: Sefaz, 2009. Disponível em: <http://www.sef.sc.gov.br/index.php?option=com_content&task=view&id=621&Itemid=82>. Acesso em: 13 de Jan. 2012.

SEVERINO, Lucas José. **O Poder do PMBOK no Gerenciamento de um Projeto SCRUM**. Londrina: UEL, 2009. 61 p. Trabalho de Conclusão de Curso (Graduação) - Ciência da Computação - Universidade Estadual de Londrina, Londrina, 2009. Disponível em: <<http://www2.dc.uel.br/nourau/document/?view=878>>. Acesso em: 14 de Jan. 2012.

SILBERSCHATZ, Abraham; KORTH, Henry F.; SUDARSHAN, S. **Sistema de banco de dados**. Rio de Janeiro: Elsevier, 2006.

SOMMERVILLE, Ian. **Engenharia de Software**. 8ª ed. São Paulo: Pearson Addison Wesley, 2007.

SPRING. **Documentation | SpringSource.org - Spring Framework**. 2012. Disponível em: <<http://www.springsource.org/documentation>>. Acesso em: 13 de Jan. 2012.

TOMCAT. **The Apache Software Foundation**. 2012. Disponível em: <<http://tomcat.apache.org>>. Acesso em: 13 de Jan. 2012

VISION, I. C. **JUDE: UML, ER, CRUD, DFD, Flowchart and Mind Map: Design and Modeling Tool**. 2012. Disponível em: <<http://jude.change-vision.com/jude-web/support/faq.html>>. Acesso em: 18 de Jan. de 2012.

WORKBENCH, MySQL. **MySQL: MySQL Workbench 5.2**. 2012. Disponível em: <<http://www.mysql.com/products/workbench>>. Acesso em: 19 de Jan. 2012.

YAHOO. **Ferramenta de busca avançada na WEB**. 2012. Disponível em: <<http://br.search.yahoo.com/web/advanced?ei=UTF-8&fr=yfp-t-707&p=>>>. Acessado em: 13 de Jan. 2012.

APÊNDICE

APÊNDICE 1 – DIAGRAMA DE ATIVIDADE – COMPARTILHAR EM REDES SOCIAIS

APÊNDICE 2 – DIAGRAMA DE ATIVIDADE – PUBLICAR COMENTÁRIOS

APÊNDICE 3 – DIAGRAMA DE ATIVIDADE – CADASTRAR POSTOS

APÊNDICE 4 – WIREFRAME – HOME

APÊNDICE 5 – WIREFRAME – CONTATO

APÊNDICE 6 – WIREFRAME – QUEM SOMOS

APÊNDICE 7 – WIREFRAME – LOGIN

APÊNDICE 8 – WIREFRAME – PESQUISA NA LISTA DE PRODUTOS

APÊNDICE 9 – WIREFRAME – SEM RESULTADOS PARA PESQUISA

APÊNDICE 10 – WIREFRAME – PESQUISA DIRETAMENTE NO MAPA

APÊNDICE 11 – WIREFRAME – MANUTENÇÃO DE PREÇOS

APÊNDICE 12 – WIREFRAME – INFORMAÇÕES DOS POSTOS

APÊNDICE 13 – WIREFRAME – ERRO GENÉRICO

APÊNDICE 14 – CÓDIGO FONTE JAVA – CLASSE DE BUSCA

APÊNDICE 15 – CÓDIGO FONTE JAVA – CLASSE DE CONEXÃO COM BANCO DE DADOS

APÊNDICE 16 – CÓDIGO FONTE JAVA – CLASSE DE PERSISTÊNCIA

APÊNDICE 17 – RELAÇÃO DE LOCALIDADES

APÊNDICE 1 - DIAGRAMA DE ATIVIDADE - COMPARTILHAR EM REDES SOCIAIS

Neste diagrama de atividade demonstra-se a operação de compartilhamento em redes sociais, conforme a figura 33.

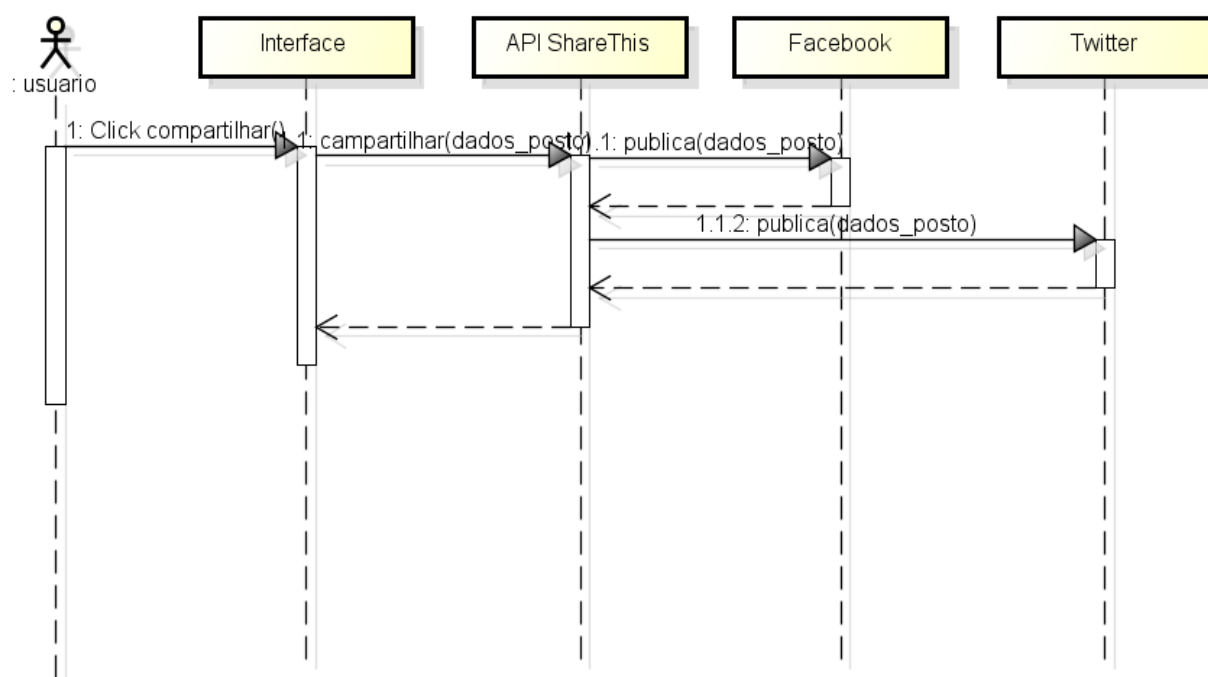


Figura 33 - Diagrama de Atividade - Compartilhar em Redes Sociais

APÊNDICE 2 - DIAGRAMA DE ATIVIDADE - PUBLICAR COMENTÁRIOS

Neste diagrama de atividade demonstra-se a operação publicar comentários, conforme a figura 34.

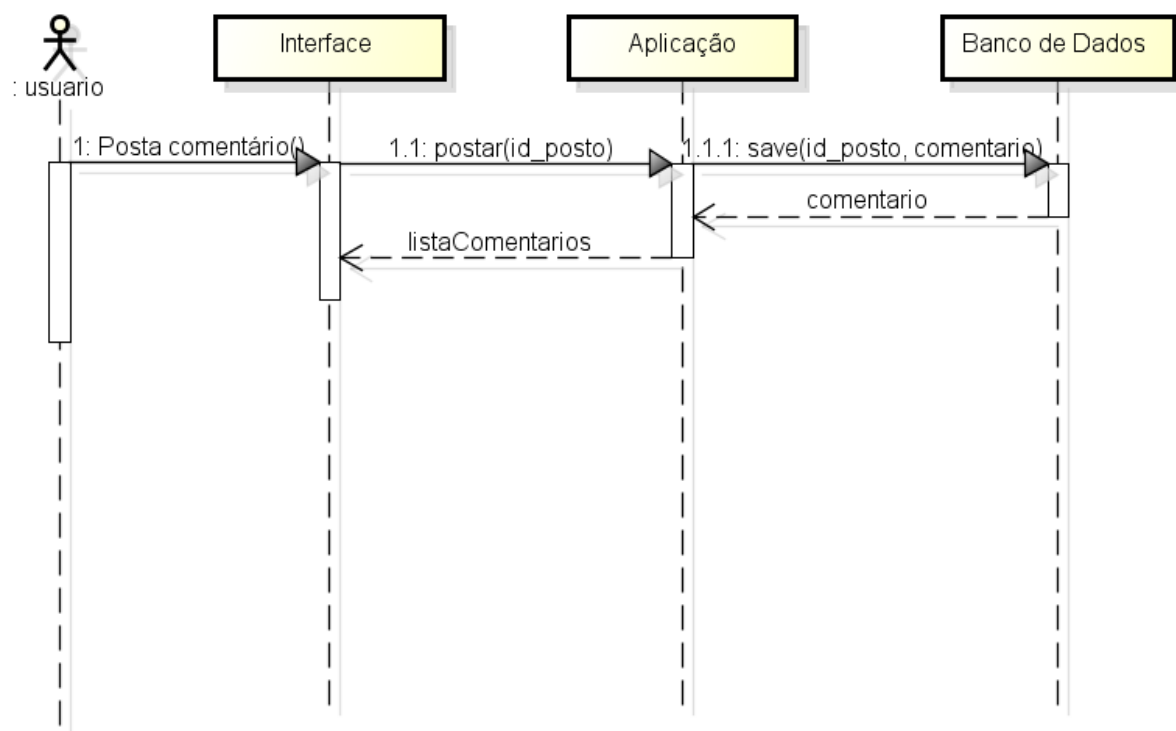


Figura 34 - Diagrama de Atividade - Publicar Comentários

APÊNDICE 3 - DIAGRAMA DE ATIVIDADE - CADASTRAR POSTOS

Neste diagrama de atividade demonstra-se a operação cadastrar postos, conforme a figura 35.

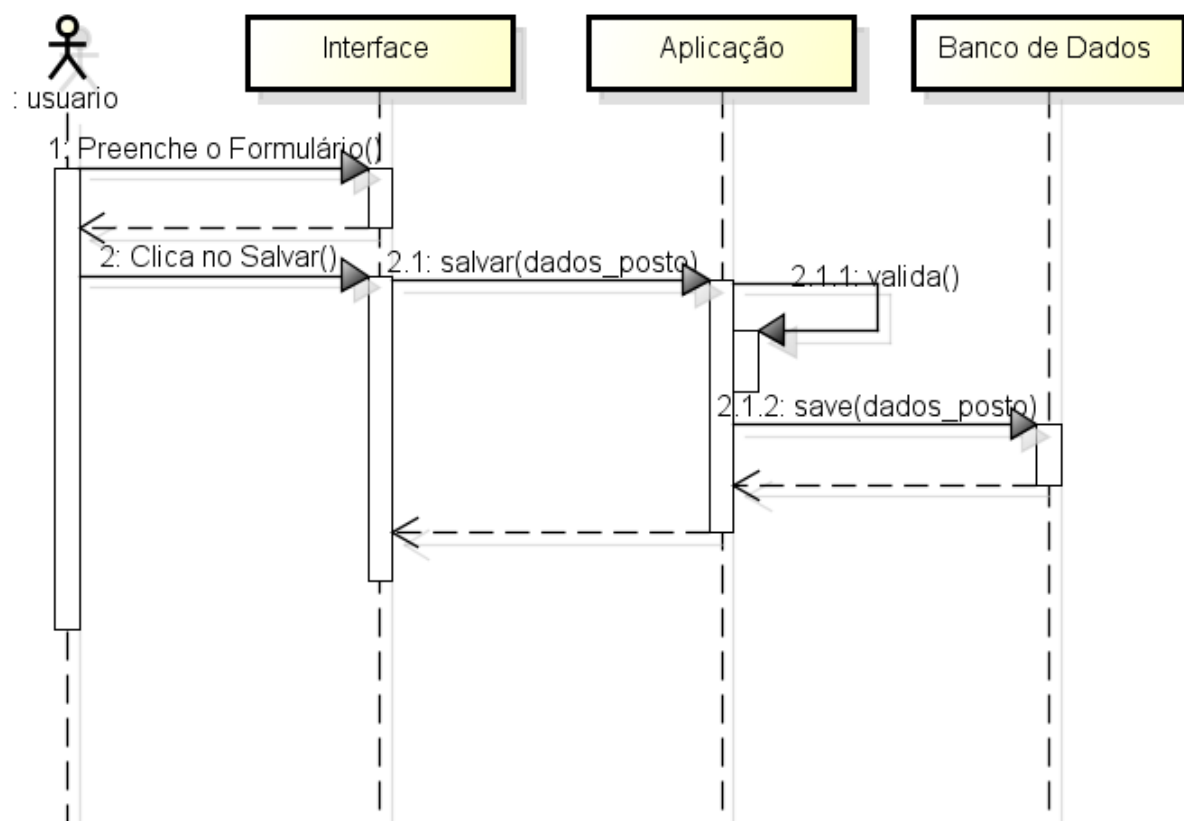


Figura 35 - Diagrama de Atividade - Cadastrar Postos

APÊNDICE 4 - WIREFRAME - HOME

A tela principal do GasFinder irá exibir a média preços dos combustíveis, sugerindo automaticamente os Postos baseado na localidade de origem do acesso do usuário. A tela Home destaca o preço da Gasolina, Alcool, Diesel e GNV, conforme demonstrado na figura 36.

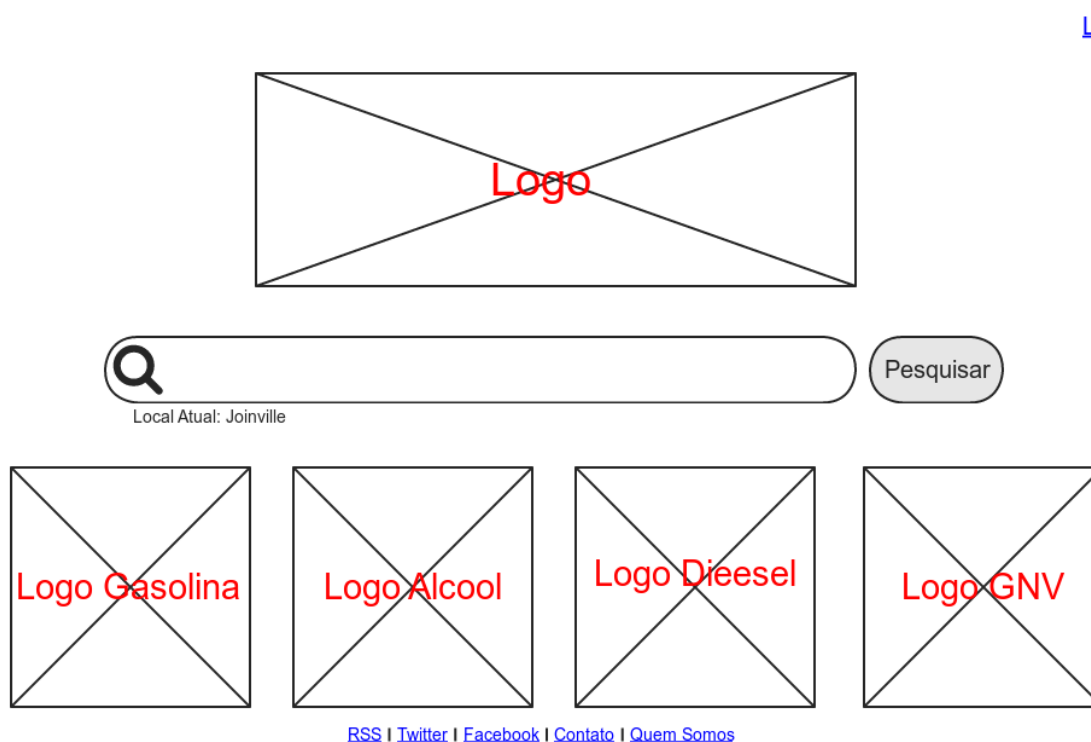


Figura 36 - Wireframe - Home

APÊNDICE 5 - WIREFRAME - CONTATO

A tela de Contato fornece um formulário onde o usuário poderá postar suas dúvidas e comentários acerca dos serviços do GasFinder. A interface é proposta conforme a figura 37.

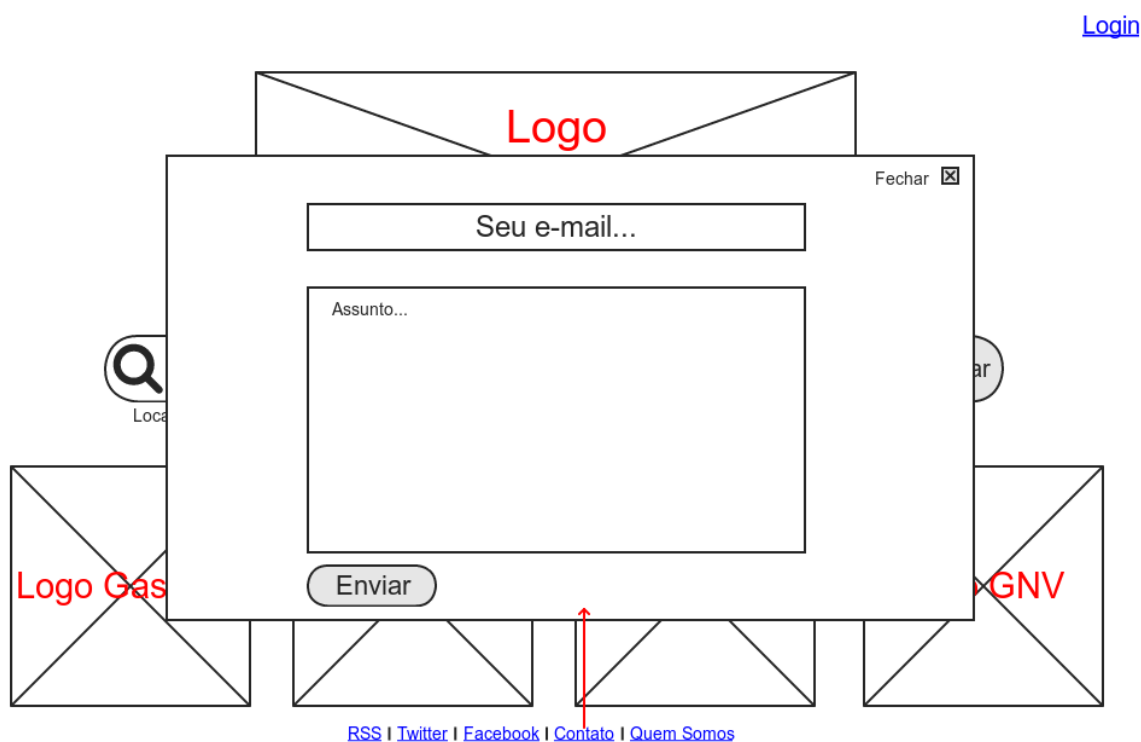


Figura 37 - Wireframe - Contato

APÊNDICE 6 - WIREFRAME - QUEM SOMOS

A tela de Quem Somos visa anunciar as principais informações acerca dos serviços do GasFinder, o objetivo do serviço e outras informações dos colaboradores. A interface é proposta conforme a figura 38.

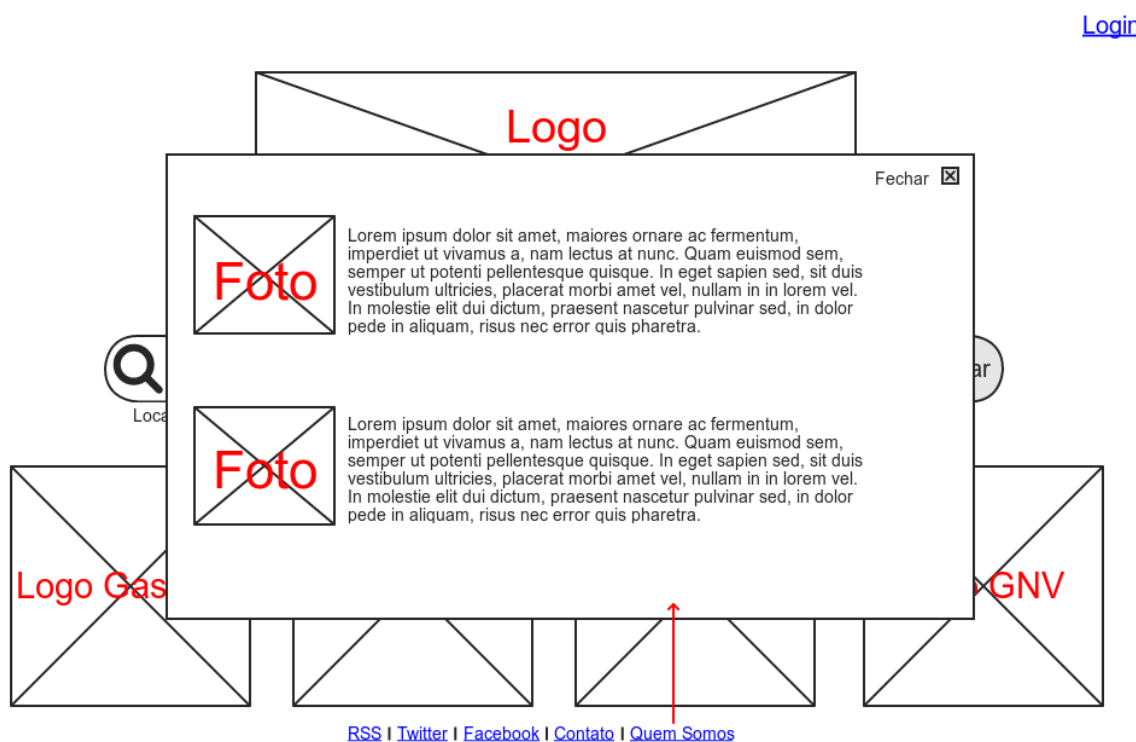


Figura 38 - Wireframe - Quem somos

APÊNDICE 7 - WIREFRAME - LOGIN

A tela de Login permite que o Administrador acesse a interface de gerenciamento do aplicativo, ou o Usuário Cadastrado acesse sua conta no GasFinder, conforme é demonstrado na figura 39.

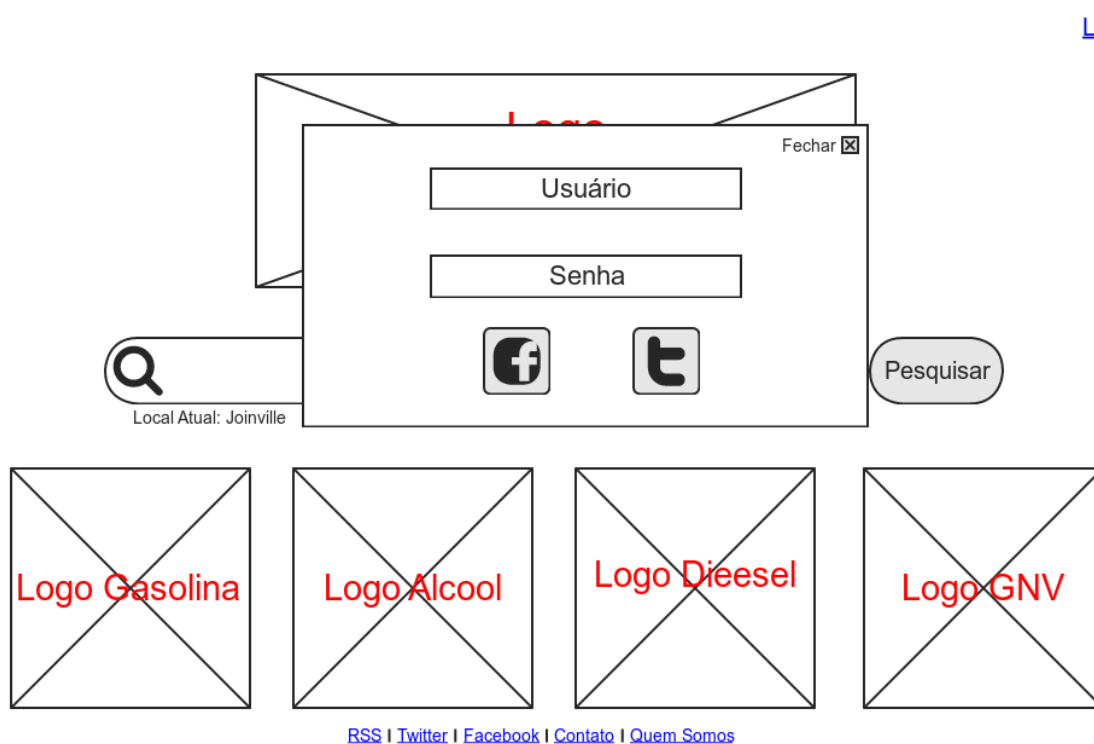


Figura 39 - Wireframe - Login

APÊNDICE 8 - WIREFRAME - PESQUISA NA LISTA DE PRODUTOS

A tela de Pesquisa permite consultar a lista de postos e comparar o preço dos combustíveis, conforme proposto na figura 40.

[Login](#)

Logo

Pesquisar

Visualizar como ☒ Lista ☐ Mapa

Posto XYZ LTDA

R\$ 9,99

R\$ 9,99

R\$ 9,99

R\$ 9,99

Rua Do Biro Biro, 999

Posto XYZ LTDA

R\$ 9,99

R\$ 9,99

R\$ 9,99

R\$ 9,99

Rua Do Biro Biro, 999

Posto XYZ LTDA

R\$ 9,99

R\$ 9,99

R\$ 9,99

R\$ 9,99

Rua Do Biro Biro, 999

Posto XYZ LTDA

R\$ 9,99

R\$ 9,99

R\$ 9,99

R\$ 9,99

Rua Do Biro Biro, 999

Posto XYZ LTDA

R\$ 9,99

R\$ 9,99

R\$ 9,99

R\$ 9,99

Rua Do Biro Biro, 999

Posto XYZ LTDA

R\$ 9,99

R\$ 9,99

R\$ 9,99

R\$ 9,99

Rua Do Biro Biro, 999

Posto XYZ LTDA

R\$ 9,99

R\$ 9,99

R\$ 9,99

R\$ 9,99

Rua Do Biro Biro, 999

1

2

3

4

5

Mais

RSS

Twitter

Facebook

Contato

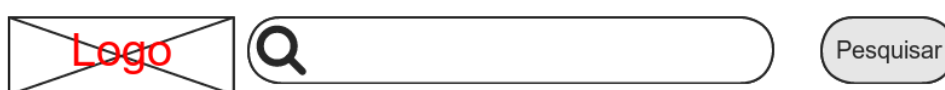
Quem Somos

Figura 40 - Wireframe - Pesquisa na lista de Postos

APÊNDICE 9 - WIREFRAME - SEM RESULTADOS PARA PESQUISA

Esta tela de resultados pode ser exibida quando a pesquisa não retorna nenhum posto na localidade desejada. Quando esta condição ocorrer, uma mensagem poderá ser exibida conforme o exemplo demonstrado na figura 41.

[Login](#)



Não foi possível encontrar a pesquisa XYZ

[RSS](#) | [Twitter](#) | [Facebook](#) | [Contato](#) | [Quem Somos](#)

Figura 41 - Wireframe - Sem resultados para pesquisa

APÊNDICE 10 - WIREFRAME - PESQUISA DIRETAMENTE NO MAPA

Esta tela permite que o usuário navegue diretamente no mapa, onde é possível deslizar o mapa para uma área desejada e visualizar graficamente os postos disponíveis em uma determinada localidade. O modelo proposto é exibido na figura 42.

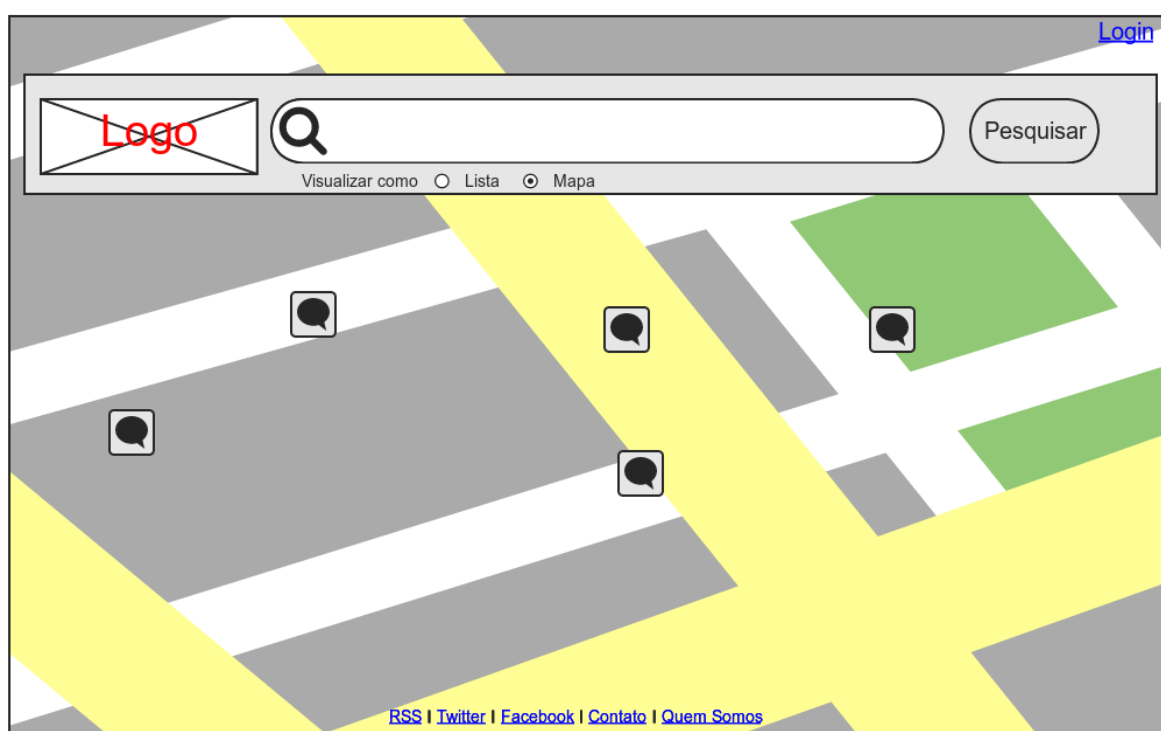


Figura 42 - Wireframe - Pesquisa diretamente no mapa

APÊNDICE 11 - WIREFRAME - MANUTENÇÃO DE PREÇOS

Esta é a interface de administração de preços dos combustíveis onde o administrador pode alterar os valores dos combustíveis. Esta tela foi projetada conforme é demonstrado na figura 43.

[Login](#)

Logo

Visualizar como ☒ Lista ☐ Mapa

Posto XYZ LTDA ☒ R\$ 9,99 ☒ R\$ 9,99 ☒ R\$ 9,99 ☒ R\$ 9,99

Rua Do Biro Biro, 999

Fechar


<input checked="" type="checkbox"/>	Gasolina	9,99
<input checked="" type="checkbox"/>	Alcool	9,99
<input checked="" type="checkbox"/>	Diesel	9,99
<input checked="" type="checkbox"/>	GNV	9,99

Atualizar

Figura 43 - Wireframe - Manutenção de Preços


APÊNDICE 12 - WIREFRAME - INFORMAÇÕES DOS POSTOS

Nesta tela é possível realizar um detalhamento das informações de cada Posto, visualizando comentário de outros usuários e também consultar o histórico dos preços praticados por um determinado Posto, conforme proposto na figura 44.




Posto do Zé Bob LTDA


[William S. de Oliveira](#)


Gasolina


9,99


Alcool

9,99


Diseel

9,99



GNV

9,99

Histórico de Alterações

Data Alterção	Gasolina	Álcool	Diseel	GNV
10/01/2011	R\$ 9.99	R\$ 9.99	R\$ 9.99	R\$ 9.99
10/01/2011	R\$ 9.99	R\$ 9.99	R\$ 9.99	R\$ 9.99
10/01/2011	R\$ 9.99	R\$ 9.99	R\$ 9.99	R\$ 9.99
10/01/2011	R\$ 9.99	R\$ 9.99	R\$ 9.99	R\$ 9.99
10/01/2011	R\$ 9.99	R\$ 9.99	R\$ 9.99	R\$ 9.99
10/01/2011	R\$ 9.99	R\$ 9.99	R\$ 9.99	R\$ 9.99

Comentários



Fulado de tal

Lorem ipsum dolor sit amet, maiores ornare ac fermentum, imperdiet ut vivamus a, nam lectus at nunc. Quam euismod sem, semper ut potenti pellentesque quisque. In eget sapien sed, sit duis vestibulum ultricies, placerat morbi

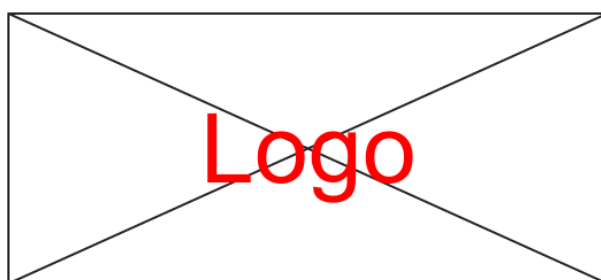
Postar

[RSS](#) | [Twitter](#) | [Facebook](#) | [Contato](#) | [Quem Somos](#)

Figura 44 - Wireframe - Informações dos Postos

APÊNDICE 13 - WIREFRAME - ERRO GENÉRICO

Esta tela de erro tem por objetivo alertar o usuário na ocorrência de erros genéricos. O modelo proposto é demonstrado na figura 45.



Ooopss!!! O pneu furou. Já alocamos 12 macacos treinados para trocar!!!

[Clique aqui para retornar ao Home!!!](#)

Figura 45 - Wireframe - Erro genérico

APÊNDICE 14 - CÓDIGO FONTE JAVA - CLASSE DE BUSCA

A ferramenta de busca compreende um dos recursos principais do aplicativo e possui uma classe desenvolvida em Java para esta função. Uma amostra do código fonte em Java da classe da ferramenta de busca do GasFinder é demonstrado na figura 46.

```

60  @SuppressWarnings({ "unchecked", "unused" })
61  public List<PostoBusca> buscar() throws ParseException{
62      Session session = BeanFactory.getSessionFactory().openSession();
63
64      FullTextSession fullTextSession = Search.getFullTextSession(session);
65      Transaction tx = fullTextSession.beginTransaction();
66
67      String[] fields = new String[]{"nomePosto", "endereco", "bandeira", "cidade", "uf", "combustivel", "valor"};
68      MultiFieldQueryParser parser = new MultiFieldQueryParser(fields, new StandardAnalyzer());
69      Query query = parser.parse( this.palavras );
70
71      org.hibernate.Query hibQuery = fullTextSession.createFullTextQuery(query, PostoBusca.class);
72
73      List<PostoBusca> result = hibQuery.list();
74      return result;
75  }
76
77  public void marcar(ActionEvent actionEvent) throws ParseException{
78      this.id = FacesContext.getCurrentInstance().getExternalContext().getRequestParameterMap().get("id");
79      this.idxTab = 1;
80      simpleModel = new DefaultMapModel();
81      List<PostoBusca> posto = this.postoBuscaDAO.buscarPorId(new Long(this.id));
82      this.marker(posto.get(0));
83  }
84
85  public void marker(PostoBusca posto) throws ParseException {
86      try {
87          LatLng coord1 = new LatLng(posto.getLatitude(), posto.getLongitude());
88          simpleModel.addOverlay(new Marker(coord1, posto.getNomePosto(),posto,"images/gazstation.png"));
89      } catch (Exception e) {
90          System.out.println("Erro: "+posto.getNomePosto());
91          e.printStackTrace();
92      }
93  }
94

```

Figura 46 - Código Fonte Java - Classe de Busca

APÊNDICE 15 - CÓDIGO FONTE JAVA - CLASSE DE CONEXÃO COM BANCO DE DADOS

A classe de conexão fornece o retorno da conexão com o banco de dados, utilizando-se esta classe, em diversas partes do aplicativo. Uma amostra do código fonte em Java da classe de conexão com banco de dados é demonstrado na figura 47.

```

6 public class Conexao {
7
8     //Variável para carregar o driver
9     private static final String STR_DRIVER = "com.mysql.jdbc.Driver";
10    //Variável para passar o database
11    private static final String DATABASE = "tcc2";
12    //Variável para carregar o IP (no meu caso localhost)
13    private static final String IP = "localhost";
14    //Variável para passar o caminho da conexão juntando com IP e o database
15    private static final String STR_CON = "jdbc:mysql://" + IP + ":3306/" + DATABASE;
16    //Variável para passar o login
17    private static final String USER = "root";
18    //Variável para passar a senha
19    private static final String PASSWORD = "";
20
21    public static Connection getConexao() throws ClassNotFoundException {
22        //Variável de conexão
23        Connection conn = null;
24        try {
25            //Carregando o driver
26            Class.forName(STR_DRIVER);
27            //Estabelecendo conexão
28            conn = DriverManager.getConnection(STR_CON, USER, PASSWORD);
29            System.out.println("Obtendo conexao");
30            //Retornando a conexão
31            return conn;
32        } catch (ClassNotFoundException e) {
33            //Tratando caso não carregue o driver
34            String errorMsg = "Driver nao encontrado";
35            throw new ClassNotFoundException(errorMsg, e);
36        } catch (SQLException e) {
37            //Tratando caso não consiga obter a conexão
38            String errorMsg = "Erro ao obter a conexao";
39            throw new ClassNotFoundException(errorMsg, e);
40        }
41    }
42 }

```

Figura 47 - Código Fonte Java - Classe de Conexão com Banco de Dados

APÊNDICE 16 - CÓDIGO FONTE JAVA - CLASSE DE PERSISTÊNCIA

As classes de persistência criam a camada de persistência do aplicativo e implementam as regras de negócio, como por exemplo: Buscar todos os Postos; Gravar valor do combustível; Excluir valor do combustível; e etc. Uma amostra do código fonte em Java da classe de persistência é demonstrado na figura 48.

```
1 package br.com.gasfinder.dao;
2
3 import java.util.List;
4
5 import org.springframework.orm.hibernate3.support.HibernateDaoSupport;
6
7 import br.com.gasfinder.entidade.ValorCombustivel;
8
9
10 public class ValorCombustivelDAO extends HibernateDaoSupport {
11
12     public List<ValorCombustivel> buscarTodos(){
13         return getHibernateTemplate().loadAll(ValorCombustivel.class);
14     }
15
16     public ValorCombustivel buscarPorId(Long id){
17         return getHibernateTemplate().get(ValorCombustivel.class, id);
18     }
19
20     public void gravar(ValorCombustivel valor){
21         getHibernateTemplate().saveOrUpdate(valor);
22     }
23
24     public void excluir(ValorCombustivel valor){
25         getHibernateTemplate().delete(valor);
26     }
27
28 }
29
```

Figura 48 - Código Fonte Java - Classe de Persistência

APÊNDICE 17 – RELAÇÃO DE LOCALIDADES

A relação de localidades, demonstrada na Tabela 2, lista os municípios onde há coleta de preços de combustível, conforme divulgado pela ANP (2010, p. 5 – 15), totalizando 15.122 postos de combustíveis importados pelo GasFinder.

Cabe-se observar, que na data de importação dos dados do SLP (Serviços de Levantamento de Preços), realizado pelo GasFinder em 08/02/2012, não foi retornado resultado para 17 localidades (nº 95 à 112), do estado de Goiás (GO). Foi constatado que estas localidades também estavam indisponíveis quando consultado manualmente o SLP, através do endereço www.anp.gov.br/preco. Portanto, descartou-se a possibilidade de erro durante a importação.

Deste modo, a exibição dos resultados de pesquisa para as localidades nº 95 à 112, será omitida no aplicativo GasFinder.

Nº	UF	Localidade*	Nº de Postos Revendedores**
1	AC	CRUZEIRO DO SUL	13
2	AC	RIO BRANCO	83
3	AC	SENA MADUREIRA	10
4	AC	SENADOR GUIOMARD	5
5	AC	TARAUACA	7
6	AC	XAPURI	3
7	AL	ARAPIRACA	19
8	AL	DELMIRO GOUVEIA	7
9	AL	MACEIO	62
10	AL	MARECHAL DEODORO	6
11	AL	PALMEIRA DOS INDIOS	12
12	AL	RIO LARGO	11
13	AL	SANTANA DO IPANEMA	14
14	AM	HUMAITA	7
15	AM	ITACOATIARA	18
16	AM	MANACAPURU	19
17	AM	MANAUS	270
18	AM	PARINTINS	7
19	AM	PRESIDENTE FIGUEIREDO	4
20	AM	TEFE	11
21	AP	CALCOENE	2
22	AP	LARANJAL DO JARI	3
23	AP	MACAPA	32
24	AP	OIAPOQUE	4
25	AP	PORTO GRANDE	3
26	AP	SANTANA	12
27	BA	ALAGOINHAS	15
28	BA	BARRA	9
29	BA	BARREIRAS	21
30	BA	BRUMADO	11

Nº	UF	Localidade*	Nº de Postos Revendedores**
31	BA	CAETITE	12
32	BA	CAMACARI	24
33	BA	CAMPO FORMOSO	8
34	BA	CANDEIAS	13
35	BA	CONCEICAO DO JACUIPE	9
36	BA	EUNAPOLIS	14
37	BA	FEIRA DE SANTANA	30
38	BA	GUANAMBI	25
39	BA	ILHEUS	15
40	BA	IPIRA	11
41	BA	IRECE	17
42	BA	ITABUNA	31
43	BA	ITAMARAJU	11
44	BA	JACOBINA	14
45	BA	JAGUAQUARA	18
46	BA	JEQUIE	19
47	BA	JUAZEIRO	19
48	BA	LAURO DE FREITAS	20
49	BA	LIVRAMENTO DE NOSSA SENHORA	9
50	BA	PAULO AFONSO	14
51	BA	POCOES	14
52	BA	PORTO SEGURO	18
53	BA	RUY BARBOSA	8
54	BA	SALVADOR	171
55	BA	SANTO ANTONIO DE JESUS	17
56	BA	SENHOR DO BONFIM	11
57	BA	SERRINHA	10
58	BA	SIMOES FILHO	14
59	BA	TEIXEIRA DE FREITAS	13
60	BA	VALENCA	13

Nº	UF	Localidade*	Nº de Postos Revendedores**
61	BA	VITORIA DA CONQUISTA	22
62	CE	BEBERIBE	4
63	CE	CANINDE	11
64	CE	CAUCAIA	38
65	CE	CRATEUS	11
66	CE	CRATO	23
67	CE	FORTALEZA	214
68	CE	ICO	10
69	CE	IGUATU	13
70	CE	IPU	10
71	CE	ITAPIOCA	15
72	CE	JUAZEIRO DO NORTE	24
73	CE	LIMOEIRO DO NORTE	12
74	CE	MARACANAU	23
75	CE	MORADA NOVA	13
76	CE	PEDRA BRANCA	9
77	CE	QUIXADA	17
78	CE	SOBRAL	21
79	DF	BRASILIA	193
80	ES	ARACRUZ	15
81	ES	BARRA DE SAO FRANCISCO	15
82	ES	CACHOEIRO DE ITAPEMIRIM	27
83	ES	CARIACICA	41
84	ES	CASTELO	4
85	ES	COLATINA	16
86	ES	GUARAPARI	19
87	ES	ITAPEMIRIM	7
88	ES	JAGUARE	9
89	ES	LINHARES	27
90	ES	NOVA VENECIA	6
91	ES	SAO MATEUS	22
92	ES	SERRA	51
93	ES	VILA VELHA	31
94	ES	VITORIA	29
95	GO	AGUAS LINDAS DE GOIAS	Não há ***
96	GO	ANAPOLIS	Não há ***
97	GO	APARECIDA DE GOIANIA	Não há ***
98	GO	CALDAS NOVAS	Não há ***
99	GO	CATALAO	Não há ***
100	GO	FORMOSA	Não há ***

Nº	UF	Localidade*	Nº de Postos Revendedores**
101	GO	GOIANIA	Não há ***
102	GO	GOIATUBA	Não há ***
103	GO	ITUMBIARA	Não há ***
104	GO	JATAI	Não há ***
105	GO	LUZIANIA	Não há ***
106	GO	MINEIROS	Não há ***
107	GO	MORRINHOS	Não há ***
108	GO	PLANALTINA	Não há ***
109	GO	PORANGATU	Não há ***
110	GO	RIO VERDE	Não há ***
111	GO	TRINDADE	Não há ***
112	GO	VALPARAISO DE GOIAS	Não há ***
113	MA	ACAILANDIA	10
114	MA	BACABAL	9
115	MA	BALSAS	11
116	MA	BARRA DO CORDA	15
117	MA	CAROLINA	7
118	MA	CAXIAS	10
119	MA	CODO	7
120	MA	IMPERATRIZ	51
121	MA	PINHEIRO	7
122	MA	PRESIDENTE DUTRA	9
123	MA	SANTA INES	9
124	MA	SAO DOMINGOS DO MARANHAO	5
125	MA	SAO JOAO DOS PATOS	5
126	MA	SAO JOSE DE RIBAMAR	14
127	MA	SAO LUIS	55
128	MA	TIMON	10
129	MG	ALFENAS	26
130	MG	ARAGUARI	35

Nº	UF	Localidade*	Nº de Postos Revendedores**
131	MG	ARAXA	16
132	MG	BARBACENA	27
133	MG	BELO HORIZONTE	219
134	MG	BETIM	45
135	MG	BOM DESPACHO	11
136	MG	CAMPO BELO	17
137	MG	CARATINGA	10
138	MG	CATAGUASES	9
139	MG	CONGONHAS	14
140	MG	CONSELHEIRO LAFAIETE	20
141	MG	CONTAGEM	99
142	MG	CORONEL FABRICIANO	18
143	MG	CURVELO	17
144	MG	DIAMANTINA	11
145	MG	DIVINOPOLIS	32
146	MG	FORMIGA	14
147	MG	FRUTAL	17
148	MG	GOVERNADOR VALADARES	35
149	MG	GUAXUPE	17
150	MG	IBIRITE	15
151	MG	IPATINGA	29
152	MG	ITABIRA	18
153	MG	ITAJUBA	24
154	MG	ITAUNA	24
155	MG	ITUIUTABA	33
156	MG	JANAUBA	8
157	MG	JANUARIA	12
158	MG	JOAO MONLEVADE	16
159	MG	JOAO PINHEIRO	17
160	MG	JUIZ DE FORA	78

Nº	UF	Localidade*	Nº de Postos Revendedores**
161	MG	LAVRAS	23
162	MG	LEOPOLDINA	16
163	MG	MANHUACU	22
164	MG	MARIANA	6
165	MG	MONTE CARMELO	25
166	MG	MONTES CLAROS	34
167	MG	MURIAE	15
168	MG	NOVA LIMA	20
169	MG	OLIVEIRA	16
170	MG	OURO PRETO	19
171	MG	PARA DE MINAS	17
172	MG	PARACATU	17
173	MG	PASSOS	32
174	MG	PATOS DE MINAS	38
175	MG	PATROCINIO	24
176	MG	POCOS DE CALDAS	40
177	MG	POUSO ALEGRE	35
178	MG	RIBEIRAO DAS NEVES	22
179	MG	SABARA	22
180	MG	SANTA LUZIA	23
181	MG	SANTOS DUMONT	8
182	MG	SAO JOAO DEL REI	25
183	MG	SAO LOURENCO	9
184	MG	SAO SEBASTIAO DO PARAISO	22
185	MG	SETE LAGOAS	37
186	MG	TEOFILO OTONI	29
187	MG	TIMOTEO	18
188	MG	TRES CORACOES	18
189	MG	UBA	20
190	MG	UBERABA	60
191	MG	UBERLANDIA	115
192	MG	UNAI	23
193	MG	VARGINHA	33
194	MG	VESPASIANO	17
195	MG	VICOSA	9
196	MS	CAMPO GRANDE	169
197	MS	CORUMBA	17
198	MS	COXIM	20
199	MS	DOURADOS	49
200	MS	NOVA ANDRADINA	18

Nº	UF	Localidade*	Nº de Postos Revendedores**
201	MS	PARANAIBA	22
202	MS	PONTA PORA	12
203	MS	TRES LAGOAS	29
204	MT	ALTA FLORESTA	14
205	MT	CACERES	18
206	MT	CUIABA	104
207	MT	RONDONOPOLIS	40
208	MT	SANTO ANTONIO DO LEVERGER	3
209	MT	SINOP	26
210	MT	SORRISO	17
211	MT	VARZEA GRANDE	33
212	PA	ABAETETUBA	8
213	PA	ALENQUER	6
214	PA	ALTAMIRA	6
215	PA	ANANINDEUA	15
216	PA	BELEM	40
217	PA	BRAGANCA	7
218	PA	BREVES	7
219	PA	CAMETA	8
220	PA	CASTANHAL	9
221	PA	CONCEICAO DO ARAGUAIA	6
222	PA	ITAITUBA	8
223	PA	MARABA	14
224	PA	PARAGOMINAS	4
225	PA	PARAUPEBAS	9
226	PA	REDENCAO	10
227	PA	SANTANA DO ARAGUAIA	5
228	PA	SANTAREM	18
229	PA	TUCURUI	7
230	PA	XINGUARA	8

Nº	UF	Localidade*	Nº de Postos Revendedores**
231	PB	BAYEUX	12
232	PB	CABELO	15
233	PB	CAMPINA GRANDE	51
234	PB	JOAO PESSOA	95
235	PB	MAMANGUAPE	8
236	PB	PATOS	16
237	PB	SANTA RITA	10
238	PB	SAPE	7
239	PB	SOUSA	15
240	PE	ABREU E LIMA	11
241	PE	AFOGADOS DA INGAZEIRA	5
242	PE	ARARIPINA	11
243	PE	ARCOVERDE	24
244	PE	BELO JARDIM	14
245	PE	BOM CONSELHO	9
246	PE	CABO DE SANTO AGOSTINHO	17
247	PE	CAMARAGIBE	19
248	PE	CARUARU	30
249	PE	GARANHUNS	21
250	PE	GOIANA	10
251	PE	GRAVATA	11
252	PE	IGARASSU	23
253	PE	JABOATAO DOS GUARARAPES	51
254	PE	LAJEDO	12
255	PE	OLINDA	41
256	PE	PAULISTA	26
257	PE	PETROLINA	30
258	PE	RECIFE	180
259	PE	SALGUEIRO	14
260	PE	SANTA CRUZ DO CAPIBARIBE	14

Nº	UF	Localidade*	Nº de Postos Revendedores**
261	PE	SAO BENTO DO UNA	11
262	PE	SAO LOURENCO DA MATA	15
263	PE	SERRA TALHADA	13
264	PE	SERTANIA	4
265	PE	VITORIA DE SANTO ANTAO	19
266	PI	BARRAS	7
267	PI	CAMPO MAIOR	11
268	PI	OEIRAS	9
269	PI	PARNAIBA	19
270	PI	PICOS	11
271	PI	PIRIPIRI	8
272	PI	TERESINA	120
273	PR	ALMIRANTE TAMANDARE	13
274	PR	APUCARANA	30
275	PR	ARAPONGAS	28
276	PR	ARAUCARIA	22
277	PR	ASSIS CHATEAUBRIAND	17
278	PR	CAMBE	40
279	PR	CAMPO LARGO	27
280	PR	CAMPO MOURAO	27
281	PR	CASCADEL	59
282	PR	CASTRO	18
283	PR	CIANORTE	28
284	PR	COLOMBO	42
285	PR	CORNELIO PROCOPIO	14
286	PR	CURITIBA	230
287	PR	FOZ DO IGUACU	51
288	PR	FRANCISCO BELTRAO	21
289	PR	GUARAPUAVA	27
290	PR	LARANJEIRAS DO SUL	13
291	PR	LONDRINA	145
292	PR	MARECHAL CANDIDO RONDON	14
293	PR	MARINGA	59
294	PR	PARANAGUA	43
295	PR	PARANAVAI	23
296	PR	PATO BRANCO	23
297	PR	PINHAIS	24
298	PR	PONTA GROSSA	45
299	PR	SANTO ANTONIO DA PLATINA	23
300	PR	SAO JOSE DOS PINHAIS	31

Nº	UF	Localidade*	Nº de Postos Revendedores**
301	PR	TOLEDO	39
302	PR	UMUARAMA	34
303	PR	UNIAO DA VITORIA	12
304	RJ	ANGRA DOS REIS	16
305	RJ	ARARUAMA	20
306	RJ	ARMACAO DOS BUZIOS	6
307	RJ	BARRA DO PIRAI	13
308	RJ	BARRA MANSA	35
309	RJ	BELFORD ROXO	14
310	RJ	CABO FRIO	12
311	RJ	CAMPOS DOS GOYTACAZES	56
312	RJ	DUQUE DE CAXIAS	63
313	RJ	ITABORAI	21
314	RJ	ITAGUAI	11
315	RJ	ITAPERUNA	18
316	RJ	MACAE	16
317	RJ	MAGE	12
318	RJ	MANGARATIBA	9
319	RJ	MARICA	12
320	RJ	NILOPOLIS	10
321	RJ	NITEROI	35
322	RJ	NOVA FRIBURGO	25
323	RJ	NOVA IGUACU	41
324	RJ	PARAIBA DO SUL	9
325	RJ	PARATI	6
326	RJ	PETROPOLIS	22
327	RJ	QUEIMADOS	5
328	RJ	RESENDE	18
329	RJ	RIO BONITO	14
330	RJ	RIO DE JANEIRO	268

Nº	UF	Localidade*	Nº de Postos Revendedores**
331	RJ	SANTO ANTONIO DE PADUA	10
332	RJ	SAO FRANCISCO DE ITABAPOANA	9
333	RJ	SAO GONCALO	47
334	RJ	SAO JOAO DE MERITI	26
335	RJ	SAPUCAIA	10
336	RJ	SAQUAREMA	13
337	RJ	TERESOPOLIS	23
338	RJ	TRES RIOS	10
339	RJ	VALENCA	13
340	RJ	VASSOURAS	6
341	RJ	VOLTA REDONDA	26
342	RN	CAICO	13
343	RN	CURRAIS NOVOS	10
344	RN	MOSSORO	27
345	RN	NATAL	94
346	RN	PARNAMIRIM	36
347	RN	SAO JOSE DE MIPIBU	11
348	RO	ARIQUEMES	15
349	RO	CACOAL	13
350	RO	GUAJARA-MIRIM	9
351	RO	JI-PARANA	28
352	RO	PIMENTA BUENO	12
353	RO	PORTO VELHO	87
354	RO	SAO MIGUEL DO GUAPORE	9
355	RO	VILHENA	15
356	RR	BOA VISTA	42
357	RR	CANTA	6
358	RR	CARACARAI	7
359	RR	MUCAJAI	3
360	RR	RORAINOPOLIS	5

Nº	UF	Localidade*	Nº de Postos Revendedores**
361	RS	ALEGRETE	19
362	RS	ALVORADA	21
363	RS	BAGE	34
364	RS	BENTO GONCALVES	25
365	RS	CACAPAVA DO SUL	15
366	RS	CACHOEIRA DO SUL	12
367	RS	CACHOEIRINHA	19
368	RS	CANOAS	41
369	RS	CAXIAS DO SUL	66
370	RS	CHUI	4
371	RS	CRUZ ALTA	21
372	RS	ERECHIM	34
373	RS	ESTEIO	20
374	RS	GRAMADO	24
375	RS	GRAVATAI	43
376	RS	GUAIBA	10
377	RS	IJUI	15
378	RS	JAGUARAÓ	12
379	RS	LAJEADO	18
380	RS	NOVO HAMBURGO	34
381	RS	OSÓRIO	21
382	RS	PALMEIRA DAS MISSOES	13
383	RS	PASSO FUNDO	50
384	RS	PELOTAS	61
385	RS	PORTO ALEGRE	152
386	RS	RIO GRANDE	51
387	RS	SANTA CRUZ DO SUL	30
388	RS	SANTA MARIA	39
389	RS	SANTA ROSA	21
390	RS	SANTA VITÓRIA DO PALMAR	8
391	RS	SANTANA DO LIVRAMENTO	26
392	RS	SANTO ÂNGELO	18
393	RS	SAO BORJA	10
394	RS	SAO GABRIEL	20
395	RS	SAO LEOPOLDO	31
396	RS	SAO LUIZ GONZAGA	12
397	RS	SAPIRANGA	19
398	RS	SAPUCAIA DO SUL	29
399	RS	TORRES	13
400	RS	TRAMANDAI	16

Nº	UF	Localidade*	Nº de Postos Revendedores**
401	RS	URUGUAIANA	16
402	RS	VACARIA	19
403	RS	VIAMAO	27
404	SC	ARARANGUA	14
405	SC	BALNEARIO CAMBORIU	15
406	SC	BIGUACU	9
407	SC	BLUMENAU	43
408	SC	BRUSQUE	24
409	SC	CACADOR	11
410	SC	CHAPECO	37
411	SC	CONCORDIA	15
412	SC	CRICIUMA	22
413	SC	FLORIANOPOLIS	103
414	SC	ITAJAI	37
415	SC	JARAGUA DO SUL	23
416	SC	JOINVILLE	57
417	SC	LAGES	28
418	SC	LAGUNA	11
419	SC	MAFRA	13
420	SC	PALHOCA	25
421	SC	SAO JOSE	29
422	SC	SAO MIGUEL D'OESTE	11
423	SC	TUBARAO	26
424	SC	VIDEIRA	14
425	SC	XANXERE	13
426	SE	ARACAJU	33
427	SE	ITABAIANA	9
428	SE	ITABAIANINHA	5
429	SE	LAGARTO	11
430	SE	NOSSA SENHORA DO SOCORRO	15

Nº	UF	Localidade*	Nº de Postos Revendedores**
431	SE	SIMAO DIAS	4
432	SE	TOBIAS BARRETO	6
433	SP	ADAMANTINA	19
434	SP	AMERICANA	37
435	SP	AMPARO	16
436	SP	APIAI	6
437	SP	ARACATUBA	54
438	SP	ARARAQUARA	39
439	SP	ARARAS	23
440	SP	ASSIS	36
441	SP	ATIBAIA	41
442	SP	AVARE	17
443	SP	BARRETOS	30
444	SP	BARUERI	35
445	SP	BAURU	62
446	SP	BEBEDOURO	23
447	SP	BIRIGUI	38
448	SP	BOTUCATU	28
449	SP	BRAGANCA PAULISTA	35
450	SP	CACAPAVA	29
451	SP	CAMPINAS	105
452	SP	CAMPOS DO JORDAO	11
453	SP	CAPAO BONITO	12
454	SP	CARAGUATATUBA	19
455	SP	CARAPICUIBA	37
456	SP	CATANDUVA	35
457	SP	COSMOPOLIS	15
458	SP	COTIA	30
459	SP	CRUZEIRO	18
460	SP	CUBATAO	10

Nº	UF	Localidade*	Nº de Postos Revendedores**
461	SP	DIADEMA	36
462	SP	DRACENA	18
463	SP	EMBU	20
464	SP	FERRAZ DE VASCONCELOS	16
465	SP	FRANCA	37
466	SP	FRANCISCO MORATO	19
467	SP	FRANCO DA ROCHA	12
468	SP	GARÇA	19
469	SP	GUARATINGUETÁ	25
470	SP	GUARUJÁ	36
471	SP	GUARULHOS	76
472	SP	HORTOLÂNDIA	37
473	SP	IBITINGA	18
474	SP	INDAIATUBA	47
475	SP	ITANHÉM	17
476	SP	ITAPECERICA DA SERRA	15
477	SP	ITAPETININGA	37
478	SP	ITAPEVA	17
479	SP	ITAPEVI	13
480	SP	ITAPIRÁ	13
481	SP	ITAPOLIS	25
482	SP	ITAQUAQUECETUBA	21
483	SP	ITATIBA	30
484	SP	ITU	34
485	SP	JABOTICABAL	24
486	SP	JACAREÍ	26
487	SP	JALES	16
488	SP	JANDIRA	16
489	SP	JAÚ	34
490	SP	JOSE BONIFÁCIO	23
491	SP	JUNDIAÍ	44
492	SP	LEME	34
493	SP	LIMEIRA	36
494	SP	LINS	35
495	SP	LORENA	21
496	SP	MARILIA	30
497	SP	MATÃO	26
498	SP	MAUA	32
499	SP	MIRASSOL	20
500	SP	MOCOCA	28

Nº	UF	Localidade*	Nº de Postos Revendedores**
501	SP	MOGI DAS CRUZES	46
502	SP	MOGI-GUACU	30
503	SP	MOGI-MIRIM	21
504	SP	MONTE ALTO	17
505	SP	OLIMPIA	17
506	SP	OSASCO	71
507	SP	OURINHOS	32
508	SP	PARAGUACU PAULISTA	18
509	SP	PAULINIA	31
510	SP	PINDAMONHANGABA	33
511	SP	PIRACICABA	54
512	SP	PIRASSUNUNGA	19
513	SP	POA	16
514	SP	PORTO FERREIRA	15
515	SP	PRAIA GRANDE	36
516	SP	PRESIDENTE PRUDENTE	49
517	SP	PRESIDENTE VENCESLAU	17
518	SP	RIBEIRAO PIRES	22
519	SP	RIBEIRAO PRETO	122
520	SP	RIO CLARO	21
521	SP	SALTO	26
522	SP	SANTA BARBARA D'OESTE	38
523	SP	SANTA CRUZ DO RIO PARDO	20
524	SP	SANTO ANDRE	71
525	SP	SANTOS	63
526	SP	SAO BERNARDO DO CAMPO	66
527	SP	SAO CAETANO DO SUL	25
528	SP	SAO CARLOS	45
529	SP	SAO JOAO DA BOA VISTA	27
530	SP	SAO JOAQUIM DA BARRA	19

Nº	UF	Localidade*	Nº de Postos Revendedores**
531	SP	SAO JOSE DO RIO PRETO	65
532	SP	SAO JOSE DOS CAMPOS	103
533	SP	SAO PAULO	562
534	SP	SAO ROQUE	16
535	SP	SAO VICENTE	50
536	SP	SERTAOZINHO	22
537	SP	SOROCABA	54
538	SP	SUMARE	37
539	SP	SUZANO	33
540	SP	TABOAO DA SERRA	23
541	SP	TATUI	25
542	SP	TAUBATE	54
543	SP	TUPA	16
544	SP	UBATUBA	25
545	SP	VALINHOS	17
546	SP	VARZEA PAULISTA	18
547	SP	VINHEDO	14
548	SP	VOTORANTIM	26
549	SP	VOTUPORANGA	25
550	TO	ARAGUAINA	22
551	TO	DIANOPOLIS	7
552	TO	GURUPI	39
553	TO	PALMAS	28
554	TO	PARAISO DO TOCANTINS	16
555	TO	PORTO NACIONAL	14

TOTAL 15122

* Localidades obtidas conforme ANP (2010, p. 5 – 15).

** Soma dos postos de combustíveis importados pelo GasFinder para cada respectiva localidade. Importação realizada em 08/02/2012.

*** A importação do GasFinder não retornou resultados para esta localidade na data de importação.

Tabela 2 - Relação de Localidades
Fonte: Adaptado (ANP, 2010, p. 5 - 15)