

**SOCIEDADE EDUCACIONAL DE SANTA CATARIA – SOCIESC
INSTITUTO SUPERIOR TUPY – IST**

**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**

É objetivo deste trabalho, apresentar uma proposta de um Portal de Pesquisa de Preços de Combustível WEB On-line com base em conceitos de escalabilidade tecnológica e de acordo com as especificações da ANP - Agencia Nacional de Petróleo.

PROF. MSc. LUIZ CARLOS CAMARGO
Professor Orientador

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**

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

Joinville, __ de _____ de 2012.

Profª. MSc. Luiz Carlos Camargo – IST

Profª. MSc. Edicarsia Barbiero Pillon – IST



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

AGRADECIMENTOS

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

Jovem, “lembra-te, pois, do teu
grandioso Criador, nos dias da tua idade viril,
antes que passem a vir os dias calamitosos ou
cheguem os anos em que dirás: Não tenho agrado neles.”
(REI SALOMÃO)

RESUMO

Este trabalho trata do desenvolvimento de uma aplicação web com foco na criação de um Portal de Pesquisa de Preços de Combustíveis, 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 solução. 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 região onde estiver situado. O desenvolvimento de software 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 Portal Search Fuel Prices, making the dissemination of the posts and their location for users, in order to provide a solution atraen you for consultation and price comparison of fuel. Will be used, the methodology for agile project definition, Scrum and XP, along with programming languages for the free development of the solution. Using Scrum model concepts, this work will result in a website service provider, in order to meet a need that is part of the lives of owners of motor vehicles: search and comparison of fuel prices. Secondary objective will provide savings for consumers and companies that need to fuel your vehicle every day, whether the region is situated. Software development 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 - Engenharia de Software em Camadas | 15 |
| Figura 2 - Modelo em Cascata | 16 |
| Figura 3 - Modelo em Espiral | 17 |
| Figura 4 - Modelo de Prototipagem | 18 |
| Figura 5 - Metodologia de Desenvolvimento Scrum | 20 |
| Figura 7 - Fluxo do processo com ações de WebE | 23 |
| Figura 8 - Processo Engenharia de Requisitos | 26 |
| Figura 11 - Diagrama de Entidade Relacionamento | 35 |
| Figura 13 - Diagrama de Seqüência – Pesquisar, Detalhar, | 37 |
| Figura 14 - Diagrama de Atividade - Pesquisar | 38 |
| Figura 15 - Wireframe - Home | 39 |
| Figura 16 - Frameworks | 40 |
| Figura 17 - Amostra do código Scraper desenvolvido em Python | 41 |
| Figura 18 - Fluxo de informações do GasFinder | 42 |
| Figura 19 - GasFinder - Tela Inicial | 43 |
| Figura 20 - GasFinder - Tela de Resultado de Pesquisa | 44 |
| Figura 21 - GasFinder - Tela de Busca no Mapa | 45 |
| Figura 22 - GasFinder - Tela de Login | 46 |
| Figura 23 - GasFinder - Tela de Detalhamento do Posto | 47 |
| Figura 24 - GasFinder - Tela de Contato | 48 |
| Figura 25 - Wireframe - Home | 54 |
| Figura 26 - Wireframe - Contato | 55 |
| Figura 27 - Wireframe - Quem somos | 56 |
| Figura 28 - Wireframe - Login | 57 |
| Figura 29 - Wireframe - Pesquisa na lista de Postos | 58 |
| Figura 30 - Wireframe - Sem resultados para pesquisa | 59 |
| Figura 31 - Wireframe - Pesquisa diretamente no mapa | 60 |
| Figura 32 - Wireframe - Manutenção de Preços | 61 |
| Figura 33 - Wireframe - Informações dos Postos | 62 |
| Figura 34 - Wireframe - Erro genérico | 63 |

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

GNV Gás Natural Veicular

HTML HyperText Markup Language

OOHDM Object Oriented Hypermedia Design Method

RSS Really Simple Syndication

SAAS Software as a service

SGDB Sistema de Gerenciamento de Banco de Dados

WEB World Wide Web

WEBAPP Web Based Application

WEBE Web Engineering

XP Extreme Programming

SUMÁRIO

| | |
|---|-----------|
| 1 INTRODUÇÃO..... | 13 |
| 2 AGÊNCIA NACIONAL DO PETRÓLEO (ANP)..... | 14 |
| 3 ENGENHARIA DE SOFTWARE..... | 15 |
| 3.1 MODELO EM CASCATA | 15 |
| 3.2 MODELO EM ESPIRAL | 17 |
| 3.3 MODELO DE PROTOTIPAGEM..... | 17 |
| 3.4 DESENVOLVIMENTO ÁGIL | 19 |
| 3.4.1 Scrum | 19 |
| 3.4.2 Extreme Programming..... | 21 |
| 3.5 ENGENHARIA WEB | 22 |
| 3.5.1 Atividades da Engenharia WEB..... | 23 |
| 3.5.2 Padrões de Projeto de Hipermedia | 23 |
| 3.5.3 OOHDM | 24 |
| 3.5.4 Protótipo da Interface..... | 25 |
| 3.6 ENGENHARIA DE REQUISITOS..... | 25 |
| 3.7 UML | 26 |
| 3.8 SAAS – SOFTWARE COMO SERVIÇOS | 27 |
| 4 DESENVOLVIMENTO DO GASFINDER..... | 28 |
| 4.1 GESTÃO E METODOLOGIA DE DESENVOLVIMENTO | 28 |
| 4.2 ESCOPO DO SISTEMA..... | 28 |
| 4.2.1 Descrição Geral | 29 |
| 4.2.2 Funções do Produto | 29 |
| 4.2.3 Características dos Usuários | 30 |
| 4.2.4 Restrições Gerais | 30 |
| 4.2.5 Assertivas e Dependências | 30 |
| 4.2.6 Requisitos | 30 |
| 4.3 MODELAGEM..... | 33 |
| 4.3.1 Casos de Uso | 33 |
| 4.3.2 Diagrama de Entidade Relacionamento | 35 |
| 4.3.3 Diagrama de Classe | 36 |

| | | |
|-------|---|----|
| 4.3.4 | Diagrama de Seqüência | 37 |
| 4.3.5 | Diagrama de Atividade..... | 38 |
| 4.3.6 | Wireframe | 38 |
| 4.4 | FERRAMENTAS DE DESENVOLVIMENTO..... | 39 |
| 4.4.1 | Extração de Dados | 41 |
| 4.5 | APRESENTAÇÃO DO SISTEMA..... | 42 |
| 4.5.1 | Tela Inicial..... | 42 |
| 4.5.2 | Tela de Resultado de Pesquisa | 44 |
| 4.5.3 | Tela de Busca no Mapa | 45 |
| 4.5.4 | Tela de Login | 46 |
| 4.5.5 | Tela de Detalhamento do Posto | 47 |
| 4.5.6 | Tela de Contato | 48 |
| 5 | CONCLUSÃO | 49 |
| | REFERÊNCIAS..... | 50 |
| | APÊNDICE..... | 53 |
| | APÊNDICE 1 – WIREFRAME – HOME | 54 |
| | APÊNDICE 2 – WIREFRAME – CONTATO | 55 |
| | APÊNDICE 3 – WIREFRAME – QUEM SOMOS..... | 56 |
| | APÊNDICE 4 – WIREFRAME – LOGIN..... | 57 |
| | APÊNDICE 5 – WIREFRAME – PESQUISA NA LISTA DE PRODUTOS..... | 58 |
| | APÊNDICE 6 – WIREFRAME – SEM RESULTADOS PARA PESQUISA | 59 |
| | APÊNDICE 7 – WIREFRAME – PESQUISA DIRETAMENTE NO MAPA..... | 60 |
| | APÊNDICE 8 – WIREFRAME – MANUTENÇÃO DE PREÇOS | 61 |
| | APÊNDICE 9 – WIREFRAME – INFORMAÇÕES DOS POSTOS | 62 |
| | APÊNDICE 10 – WIREFRAME – ERRO GENÉRICO | 63 |

1 INTRODUÇÃO

Geralmente os usuários, sejam eles empresas ou pessoa física, estão em busca de produtos e serviços que apresentem a melhor relação custo-benefício. Naturalmente selecionamos aquelas soluções 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, pesquisa sobre a opinião do produto/serviço em um grupo de adquirentes, realização de orçamento, barganha, entre outros.

Estas ações são praticas constantemente na aquisição de novos produtos, principalmente na Internet através de ferramentas e serviços que facilitam a comparação de preço de produtos em geral. Paradoxalmente, quando o assunto é custo fixo, 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 poucas pessoas realizam uma busca pelo melhor preço ou economia no momento de abastecer. 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 objetivo deste trabalho é o desenvolvimento um Portal chamado GasFinder, que ofereça, como serviço, a comparação dos preços dos combustíveis *online*, viabilizando a busca pela economia, indiferente da localização do usuário. Para realizar este trabalho, será adotado ferramentas WEB, incluindo Java, Python, MySQL e Eclipse para o desenvolvimento; Spring, Hibernate e Primefaces como frameworks auxiliares; tecnologias AJAX, HTML, CSS e JavaScript e, também conceitos ágeis para o gerenciamento do projeto, Scrum e XP. Serão utilizados aos dados dos postos de combustíveis fornecidos pela Agencia Nacional do Petróleo (ANP).

Para atingir os objetivos deste trabalho, os tópicos apresentados estão situados da seguinte forma:

- ANP: Apresentação da Agencia Nacional do Petróleo
- Engenharia de Software: Compreende o planejamento do software
- Desenvolvimento do aplicativo: Definição de requisitos e projeto

2 AGÊNCIA NACIONAL DO PETRÓLEO (ANP)

ANP ou Agência Nacional do Petróleo, Gás Natural e Biocombustíveis é o órgão regulador das atividades que integram a indústria do petróleo e gás natural e a dos biocombustíveis no Brasil. ANP possui o objetivo de estabelecer regras que viabilizem a criação de um mercado mais competitivo e que tragam vantagens para o país, especialmente, para os consumidores. Para o país, estes benefícios podem ser resumidos em uma maior arrecadação fiscal e também na diminuição das importações de petróleo. (ANP, 2012)

Através do Portal ANP, o consumidor final pode consultar os preços dos combustíveis *online* e verificar os valores praticados nas cidades onde há fiscalização. É uma funcionalidade útil, pois pode permitir que o consumidor final compare os preços dos combustíveis antes de abastecer, proporcionando economia.

Contudo, as informações disponibilizadas para o consumidor estão em formato pouco amigável, possui uma navegação ligeiramente confusa, e também carece de ferramentas de busca e recursos de interações comuns atualmente na WEB, como integração com redes sociais para compartilhamento de informações, ou mesmo RSS (*Really Simple Syndication*) para distribuição de conteúdo. Estas limitações criam uma experiência de navegação sofrível para o usuário, que invariavelmente pode se sentir desmotivado a continuar utilizando este serviço de consulta de preços.

Deste modo, entendemos que o consumidor final raramente acessa o Portal ANP para consultar os preços dos combustíveis, embora este serviço permaneça com sua plena utilidade para consulta pelos órgãos de auditoria como o PROCOM ou para outras finalidades não citadas.

Neste trabalho, as informações do Portal ANP serão processadas pelo GasFinder de forma automatizada via *data scraping*, uma tecnologia que será abordada posteriormente neste trabalho. Em suma, o GasFinder será capaz de importar os preços dos postos dos combustíveis do Portal ANP e apresentar em uma interface WEB, tornando-se um novo canal de consulta para o usuário, alternativo, minimizando as limitações oferecidas pela interface do Portal ANP e oferecendo uma experiência de navegação moderna, com ferramenta de busca, localização geográfica por ip e integração com redes sociais para o compartilhamento de informações.

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. A definição de Engenharia de Software segundo Pressman (2006, p. 17) é “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”.

A Engenharia de Software é uma tecnologia que permite ao desenvolvedor ou engenheiro de software um maior controle do processo de forma geral, focando sempre na qualidade de software, conforme é possível observar na figura 1. Ainda segundo Pressman (2006, p. 17), os três elementos fundamentais da Engenharia de Software são descritos conforme abaixo:

- **Processo:** Elo entre métodos e ferramentas, o que permite o desenvolvimento do sistema.
- **Métodos:** Fornecem os detalhes de como fazer para construir um sistema. Abrangem um vasto conjunto de tarefas, que são: comunicação, análise de requisitos, modelagem, codificação, testes e manutenção.
- **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*).



Figura 1 - Engenharia de Software em Camadas

Fonte: (PRESSMAN, 2006, p. 17)

3.1 MODELO EM CASCATA

O Modelo em Cascata é apresentado por Pressman (2006, p. 39) como 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”, conforme demonstrado na figura 2.

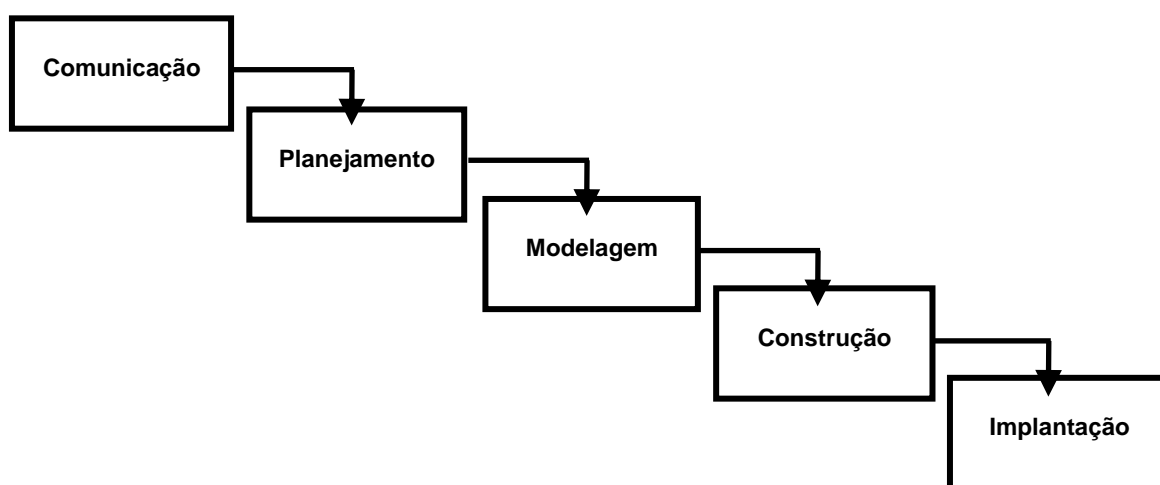


Figura 2 - Modelo em Cascata

Fonte: (PRESSMAN, 2006, p. 39)

Este modelo, considerado clássico e antigo de desenvolvimento, possui os seguintes objetivos em cada fase (PRESSMAN, 2006):

- 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.

3.2 MODELO EM ESPIRAL

Este modelo de desenvolvimento segundo Pressman (2006, p. 44), é “um gerador de modelo de processo guiado por risco e usado para guiar a engenharia de sistemas intensivos em software com vários interessados concorrentes”.

O Modelo em Espiral é definido por quatro importantes atividades, utiliza-se da prototipação para a redução de riscos e a abordagem sistêmica sugerida pelo ciclo de vida. (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.

A representação gráfica do modelo espiral é feita conforme é demonstrado na figura 3.

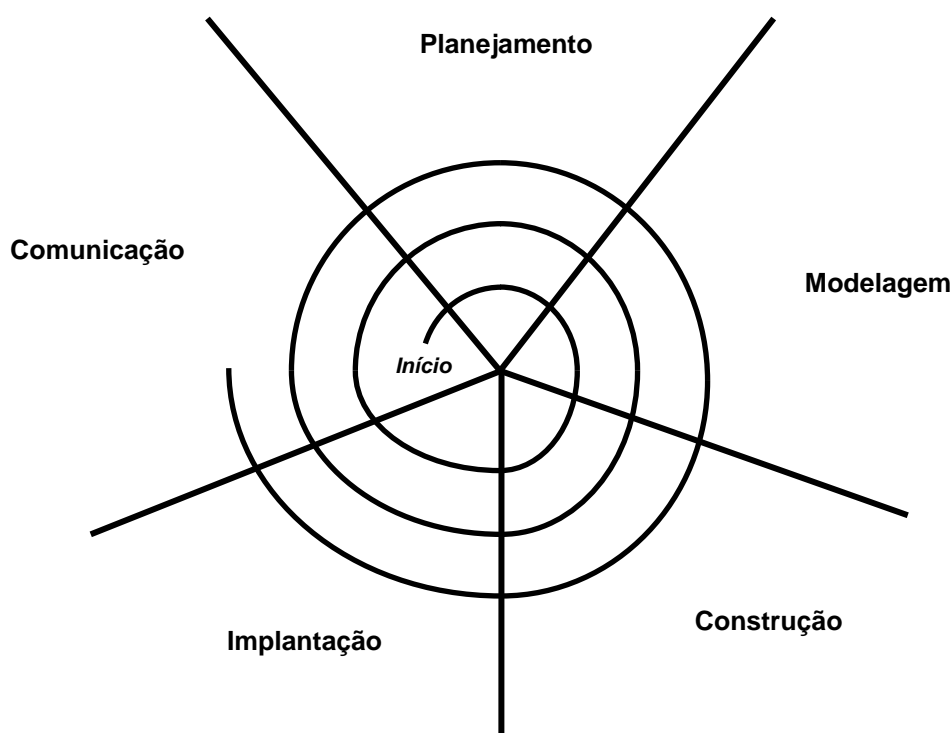


Figura 3 - Modelo em Espiral
Fonte: (PRESSMAN, 2006, p. 45)

3.3 MODELO DE PROTOTIPAGEM

Segundo Pressman (2006, p. 42), modelo de prototipagem “auxilia o engenheiro de software e o cliente a entenderem melhor o que deve ser construído quando os requisitos estão confusos.” Neste caso, um protótipo é criado de forma rápida para ser avaliado pelo cliente. Após essa avaliação, é provável que o cliente sinalize a necessidade de ajustes, para que o software criado atenda os requisitos.

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 sistema. Em muitos casos a equipe de desenvolvimento também deseja colocar o protótipo em execução muito rápido, com isso, é possível que um sistema operacional ou linguagem de programação inadequada seja empregado, meramente porque é a solução mais próxima a disposição do projeto.

Não somente, 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, entre outros fatores, este método pode ser o mais indicado. (PRESSMAN, 2006, p. 43).

Na figura 4 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.

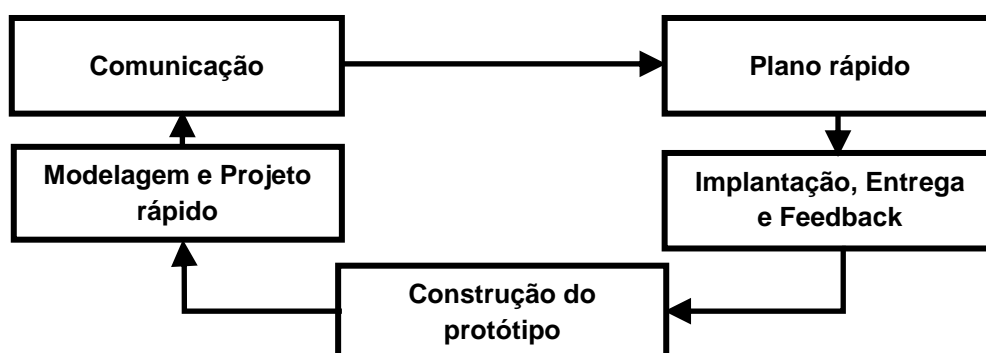


Figura 4 - Modelo de Prototipagem

Fonte: (PRESSMAN, 2006, p. 43)

3.4 DESENVOLVIMENTO ÁGIL

Os princípios do desenvolvimento ágil visam minimizar as falhas no desenvolvimento do projeto de forma incremental, em curtos períodos, chamados de interação, ou seja, mini-projetos de software que incluirão as tarefas necessárias para alcançar a funcionalidade do produto: planejamento, análise de requisitos, projeto, codificação, teste e documentação.

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)

O Desenvolvimento Ágil prioriza 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.

A motivação é um dos focos de tais modelos, as equipes de negócio e desenvolvimento trabalham juntas diariamente, deve ser dado o ambiente e o apoio necessário aos indivíduos para que eles confiem na realização do trabalho. Ter o software funcionando é a principal 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)

3.4.1 Scrum

Segundo Pressman (2006, p. 69), Scrum é 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. E se por ventura surgirem imprevistos o prazo poderá ser revisto dentro do projeto, mas isto não impedirá que o produto/serviço seja entregue no prazo. A metodologia Scrum segue o processo conforme é demonstrado na figura 5.

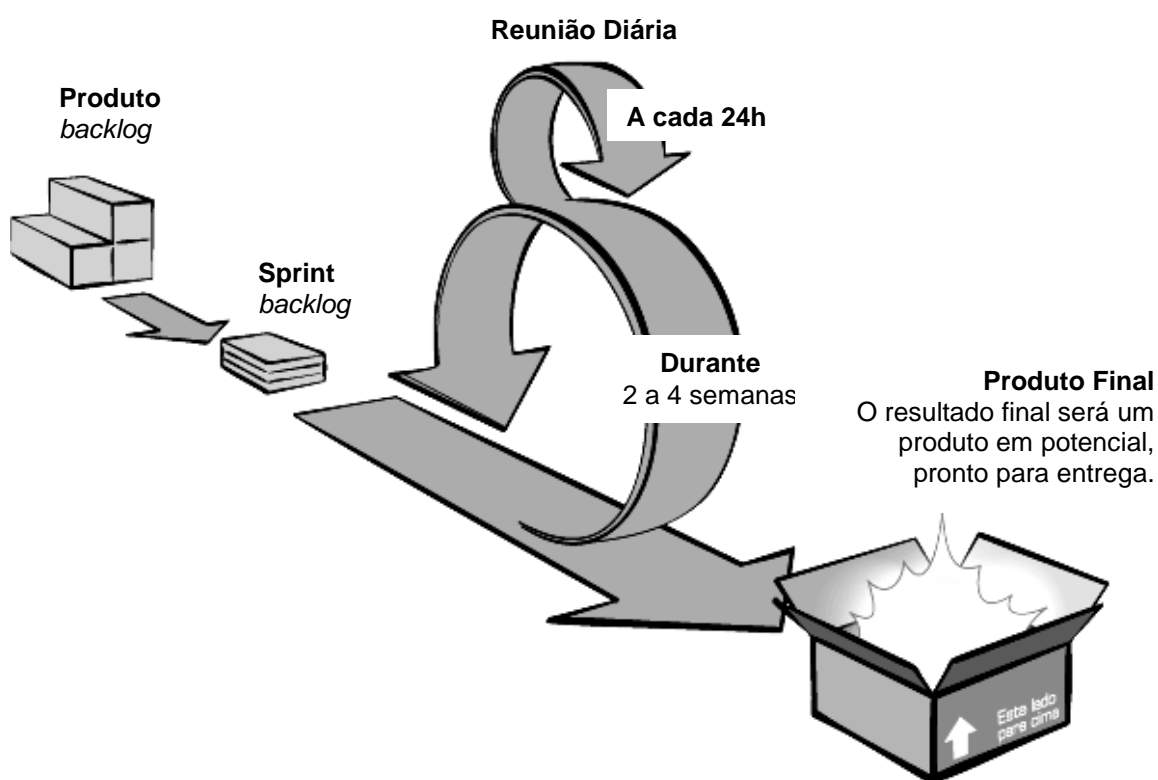


Figura 5 - Metodologia de Desenvolvimento Scrum

Fonte (http://improveit.com.br/images/br/scrum/ciclo_scrum.gif)

O Scrum é um nome oriundo do esporte rúgbi, teve suas origens em companhias nipônicas no início de 1986 por Takeuchi e Nonaka, mas começou a ser usado mais recentemente apenas em 1996. Segundo Larman (2008, 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”.

3.4.2 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. Para isso, segue a estratégia de acompanhamento permanente e cumprimento de vários ajustes pontuais ao decorrer do desenvolvimento de software.

Os cinco valores fundamentais da metodologia XP são: simplicidade, *feedback*, comunicação, respeito e coragem. A partir desses valores, possui como princípios básicos: *feedbacks* pontuais, ponderar simplicidade, ajustes incrementais, incentivar mudanças e trabalhar com qualidade.

Entre os itens que compõem a gestão de projetos (escopo, tempo, qualidade e custo), existe um foco constante no escopo. Para isso, aconselha-se a priorização de funções que agregue mais valor para o negócio. Desta forma, quando ocorrer a necessidade de reduzir o escopo, as funcionalidades com menos importância terão o desenvolvimento postergado ou até mesmo cancelado. Na figura 6 é demonstrado o processo da metodologia de desenvolvimento XP. (PRESSMAN, 2006).

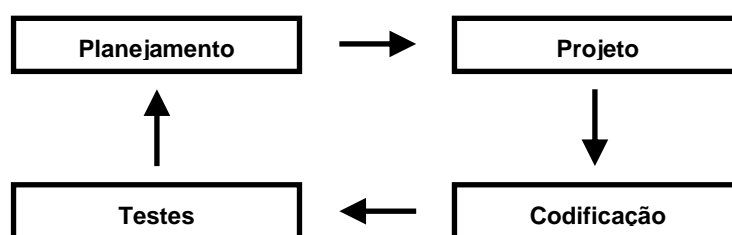


Figura 6 - Metodologia de Desenvolvimento XP

Fonte: (PRESSMAN, 2006, p. 64)

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.

Para colocar em pratica os princípios e valores ao decorrer de um projeto de software, existe uma confiança significativa quanto a sinergia, onde os possíveis pontos fracos do projeto tendem a ser superados pelos pontos fortes.

3.5 ENGENHARIA WEB

A engenharia WEB surge com uma solução específica para o crescente avanço do desenvolvimento de aplicações para WEB - também conhecida como WebApp - 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 sistemas empresariais.

Segundo Pressman e Lowe (2009, p. 11), a engenharia WEB “propõe um arcabouço 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)

Em geral, as aplicações WebApps possuem um grupo de características únicas que a diferenciam de um software convencional:

- 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.

3.5.1 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 7 (PRESSMAN; LOWE, 2009, p. 12):

- 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.

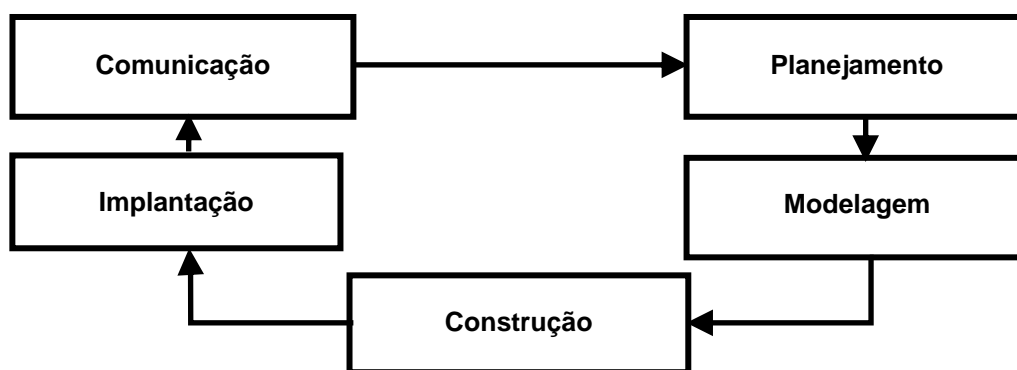


Figura 7 - Fluxo do processo com ações de WebE

Fonte: (PRESSMAN; LOWE, 2009, p. 25)

3.5.2 Padrões de Projeto de Hipermídia

As WebApps continuam evoluindo e aumentando sua complexidade. Contudo, os padrões de projetos 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 sistemas baseados em WEB, temos as seguintes categorias de padrões. Pressman (2006):

- 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.

3.5.3 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, apresentamos 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*) é composto de quatro atividades de projeto segundo Pressman (2006, p. 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.

3.5.4 Protótipo da Interface

Foram utilizados neste trabalho, alguns dos princípios abordados pela engenharia WEB, entre outros, em maior profundidade a AI - Arquitetura da Informação ou *Information Architecture* - para criação de protótipos de leiaute da interface do usuário.

Segundo Pressman e Lowe (2009, p. 12), “o leiaute da interface do usuário, o conteúdo que ela apresenta, os mecanismos de interação que ela implementa e a estética geral das conexões usuário-WebApp tem muito a ver com a satisfação do usuário e a aceitabilidade geral da WebApp.”

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 Apêndice.

3.6 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 sistema é útil para empresa, incluindo a obtenção dos requisitos, especificação destes requisitos e finalmente a validação destes requisitos, para verificar se realmente o sistema atende as necessidades.

As atividades demonstradas na figura 8 se referem a obtenção, documentação e verificação dos requisitos. Além do relacionamento entre estas atividades também é demonstrado os documentos resultantes de cada etapa.

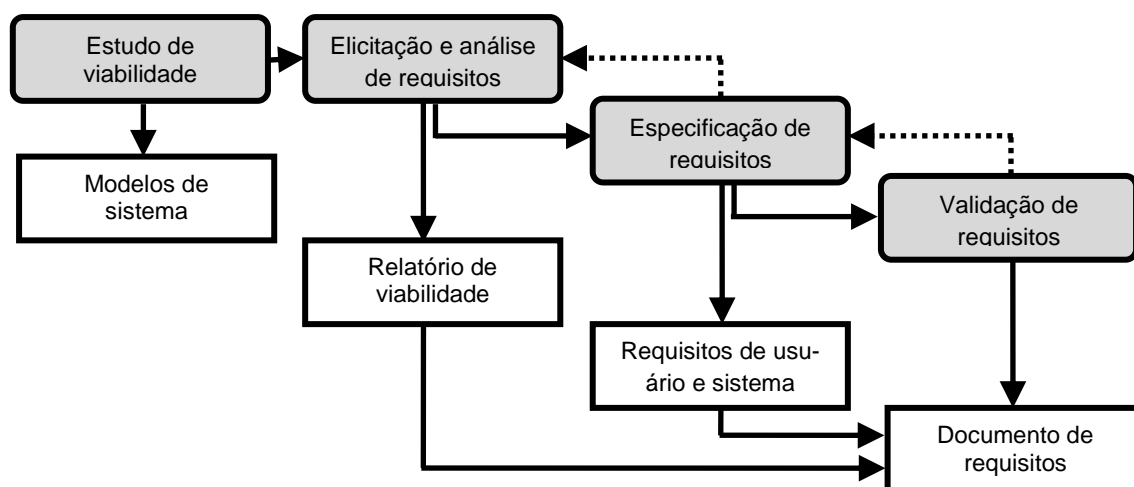


Figura 8 - Processo Engenharia de Requisitos

Fonte: (SOMMERVILLE, 2007, p. 96)

3.7 UML

Com sua logo identificada na figura 9, a UML trata-se de uma metodologia para gerenciamento de projetos, também chamado de *framework* para desenvolvimento, que segundo Bezerra (2002) é definida como uma linguagem de modelagem visual que possui um conjunto de notações e correspondente semântica, visando alcançar a representação visual de uma ou mais perspectivas do sistema. Larman (2008) caracteriza a UML como uma notação para modelagem de sistemas, fazendo uso dos conceitos de orientação a objeto.



Figura 9 – UML

Fonte: (http://www.ibm.com/developerworks/rational/library/content/RationalEdge/nov02/booch_uml_logo.gif)

A UML é composta por vários elementos de modelo que tem no seu objetivo representar as diferentes etapas de um sistema. Segundo Furlan (1998), a UML é uma linguagem de modelagem, não uma metodologia. Assim, na construção de um software, a UML deve ser usada em conjunto com uma metodologia de Engenharia de Software Orientada a Objetos.

Para o desenvolvimento do sistema proposto, serão apresentados os diagramas de caso de uso, diagrama de sequência, diagrama de classe, diagrama de entidade relacionamento e diagrama de atividade.

3.8 SAAS – SOFTWARE COMO SERVIÇOS

Software como serviço, do inglês *Software as a Service*, é um recurso de mercado que permite fornecer um software como um serviço, segundo Solari (2008). O software em questão é executado em um servidor, portando, não é preciso configurar o sistema na estação dos clientes, bastando somente acessá-lo via WEB. Como por exemplo, o Google Docs que fornece à seus usuários um pacote completo de programas de escritório.

Segundo Solari (2008 apud GOTH, 2008, p. 53), “acredita-se que SaaS terá um grande impacto na indústria de software, pois Software como Serviço irá transformar a maneira como as pessoas implementam, vendem, compra e utilizam software”.

No modelo de venda de software clássico, geralmente o produto é entregue após o pagamento ou é cobrado uma determinada quantia pelo desenvolvimento. Por sua vez, no modelo de SaaS, é desenvolvido uma solução sem cobrar qualquer quantia do cliente, ao final do desenvolvimento o serviço será disponibilizado na WEB para vários clientes e, neste caso, o cliente interessado irá investir em uma locação que concede o direito de uso do serviço por tempo determinado.

Este trabalho utiliza o modelo SaaS para disponibilizar os serviços ao usuário. Não somente, foi adotado parcialmente alguns dos conceitos abordados neste capítulo, como a criação de diagramas UML, protótipos de interface, recursos de metodologia ágil Scrum e XP que foram de suma importância para conclusão do trabalho, assim como o estudo feito no embasamento teórico sobre Engenharia de Software.

4 DESENVOLVIMENTO DO GASFINDER

Neste capítulo serão apresentados os tópicos oriundos de análise, desenvolvimento e apresentação funcional do GasFinder, 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. Na 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*.

Na 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.

Na fase de desenvolvimento, pela dificuldade em fazer reuniões diárias, foi definido reuniões semanais, através do *Skype* (ferramenta de mensagem instantânea e voz sobre ip), para discutir o que cada membro da equipe tinha concluído, quais as dificuldades encontradas e o que seria feito na próxima semana.

4.2 ESCOPO DO SISTEMA

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 dos 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 sistema 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 região.

Irá suportar três tipos de usuários distintos: o visitante que corresponde ao maior número de usuários do sistema, 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 sistema, adicionar novos postos, cidades, estados e etc.

4.2.2 Funções do Produto

As funções deste aplicativo são efetuar uma busca dos postos e preços dos combustíveis de uma determinada região. A solução 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 região.

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.

4.2.3 Características dos Usuários

Existirão três tipos de usuário:

- Usuários do tipo 1: serão os usuários que não terão contas no sistema e, utilizarão o aplicativo somente para pesquisa. (ex.: usuário final do sistema, ou consumidor);
- Usuários do tipo 2: serão os usuários que terão contas no sistema e, poderão alterar os valores dos combustíveis e adicionar comentários. (ex.: usuário do sistema);
- Usuários do tipo 3: será o usuário administrador, que poderá controlar todo o sistema. (ex.: administrador).

4.2.4 Restrições Gerais

Os postos exibidos pelo aplicativo deverão estar cadastrados na ANP.

4.2.5 Assertivas e Dependências

Este trabalho apresenta um software inovador para suprir uma necessidade sistemática.

4.2.6 Requisitos

Após o levantamento dos requisitos, temos listado abaixo os requisitos que atendem este trabalho.

- Pesquisa de Postos
 - Como: Visitante/Usuário/Administrador.
 - Quero: Pesquisar postos de determinada cidade.
 - Para: Obter informações do posto e dos preços dos combustíveis.

- Cadastro de Usuário
 - Como: Visitante/Administrador.
 - Quero: Realizar meu cadastro ou cadastrar outro usuário.
 - Para: Ter acesso a funcionalidades que solicitem autenticação.
- Marcação de Pesquisa no Mapa
 - Como: Visitante/Usuário/Administrador.
 - Quero: Visualizar no mapa a localização dos postos.
 - Para: Melhor interpretação da distância.
- Detalhes dos Postos
 - Como: Visitante/Usuário/Administrador.
 - Quero: Consultar mais informações sobre o posto.
 - Para: Obter informações relevantes sobre o posto.
- Cadastro de Preços
 - Como: Usuário/Administrador.
 - Quero: Cadastrar o valor dos combustíveis.
 - Para: Manter os valores atualizados.
- Publicação de Comentários
 - Como: Usuário/Administrador.
 - Quero: Publicar comentários sobre o posto e preços.
 - Para: Gerar informação relevante sobre o posto e preços.
- Integração com Redes Sociais
 - Como: Usuário/Administrador.
 - Quero: Compartilhar informações nas redes sociais.
 - Para: Que meus seguidores e amigos, sejam informados.
- Cadastro de Postos
 - Como: Administrador.
 - Quero: Cadastrar novos postos.
 - Para: Tornar disponível para visitantes e usuários.
- Cadastro de Combustíveis
 - Como: Administrador.
 - Quero: Cadastrar novos combustíveis que venham a surgir.
 - Para: Disponibilizar seu preço nos postos.

- Consulta de Usuários
 - Como: Administrador.
 - Quero: Consultar os usuários cadastrados.
 - Para: Gerar interação com os usuários ou enviar notificações se necessário.
- Aprovação de Comentários
 - Como: Administrador.
 - Quero: Aprovar comentários que tenha sido feitos aos postos.
 - Para: Manter a qualidade das informações geradas.
- Cadastro de Bandeiras de Postos
 - Como: Administrador.
 - Quero: Cadastrar novas bandeiras de postos, que venha a surgir.
 - Para: Melhorar a identificação e classificação dos postos.
- Cadastro de Estados
 - Como: Administrador.
 - Quero: Cadastrar estados para informar a localização dos postos.
 - Para: Permitir a localização detalhada do posto.
- Cadastro de Cidades
 - Como: Administrador.
 - Quero: Cadastrar cidades para informar a localização dos postos.
 - Para: Permitir a localização detalhada do posto.
- Aprovação Preços
 - Como: Administrador.
 - Quero: Aprovar preços, que tenham sido atualizados nos combustíveis.
 - Para: Garantir a qualidade das informações.
- Média dos Preços da Cidade
 - Como: Sistema.
 - Quero: A média dos preços da cidade, na tela inicial.
 - Para: Informação inicial de preços.

Os requisitos não funcionais visam proporcionar uma experiência agradável de navegação.

- Ser um aplicativo WEB.
- Ter uma interface amigável.
- Os campos obrigatórios para preenchimento devem ser identificados com um asterisco no lado direito.
- Todos os campos devem possuir uma legenda.
- Ser capaz de coletar, armazenar, manipular e exibir os dados da ANP para usuário.

4.3 MODELAGEM

Na modelagem do sistema foi feita através da UML, para isso foi utilizado a ferramenta case JUDE. Na modelagem do banco de dados foi utilizado a ferramenta MER MySQL Workbench.

4.3.1 Casos de Uso

Segundo Sommerville (2007, p. 102), “os casos de uso constituem uma técnica baseada em cenários para eliciação de requisitos”. Em suma, os casos de uso demonstram o tipo de interação entre os objetos.

Na figura 10 é apresentado o caso de uso geral do sistema que descreve as funcionalidades possíveis do sistema e seus relacionamentos com os atores. Serão três atores na utilização do sistema: o primeiro é o visitante, com acesso apenas a consulta e comparação de preços.

O segundo é o usuário, que poderá cadastrar comentários e novos preços dos combustíveis. O visitante passa a ser um usuário efetivamente, somente após realizar um cadastro no Portal, onde por sua vez, irá obter novos privilégios. Finalmente, o terceiro e último ator é o administrador que possui acesso de administração do sistema. O administrador tem habilidades de cadastrar novos postos, incluir cidades e estados, aprovar alteração nos preços dos combustíveis e liberar comentários publicados pelos usuários, entre outros.

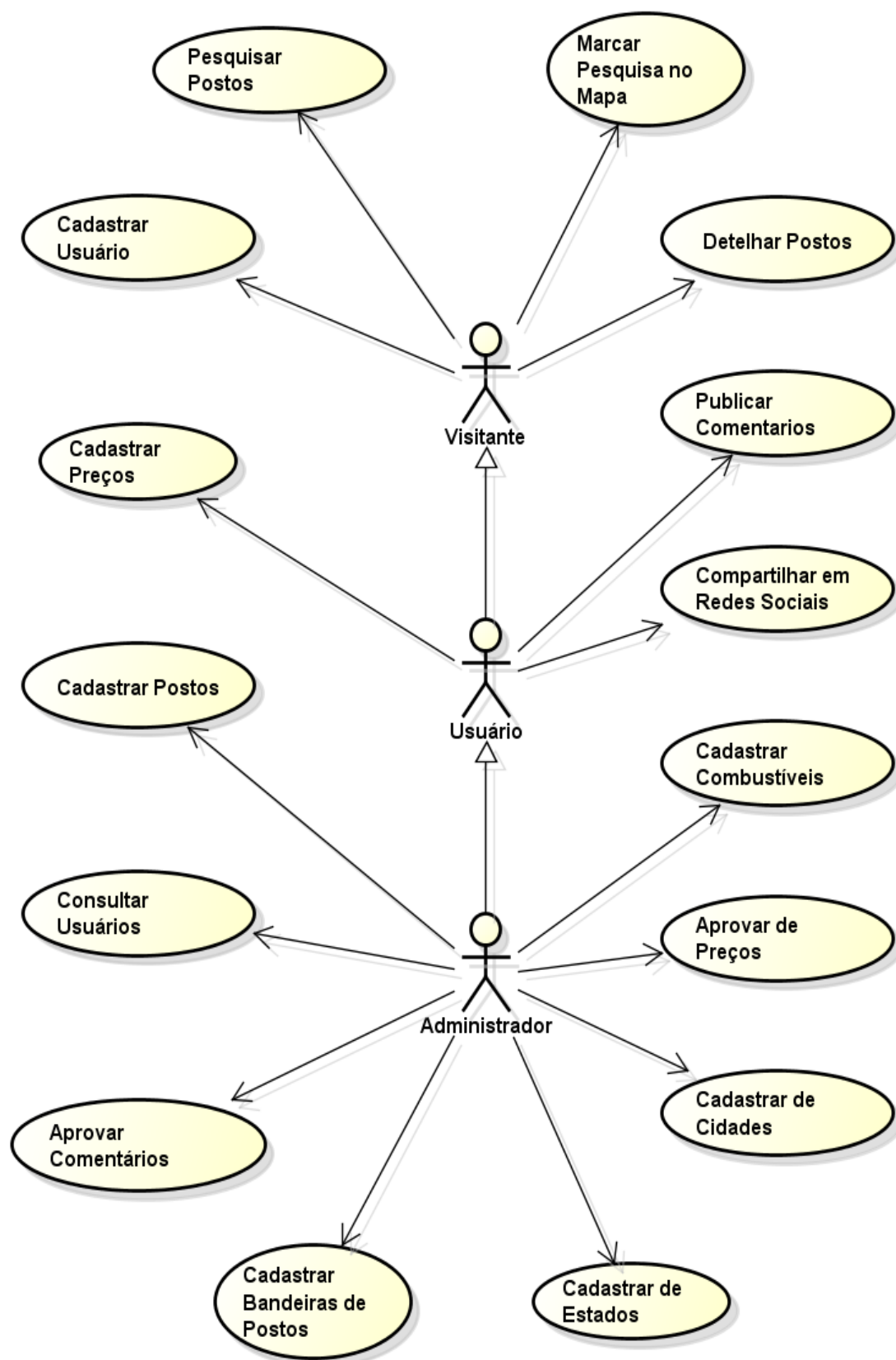


Figura 10 - Caso de Uso Geral

Fonte: Os Autores (2012)

4.3.2 Diagrama de Entidade Relacionamento

O Diagrama Entidade Relacionamento ou DER é uma técnica de modelagem de dados, que mostra as entidades de dados, seus atributos associados e a relação entre as entidades e, segundo Sommerville (2007, p. 118 - 119), são úteis para representar um modelo semântico de dados do sistema.

Na figura 11, é demonstrado a estrutura de banco de dados do GasFinder. O banco de dados MySQL foi adotado neste projeto, pois permite relacionamentos entre as tabelas e garante a integridade dos dados. Também foi utilizado a *engine* InnoDB.

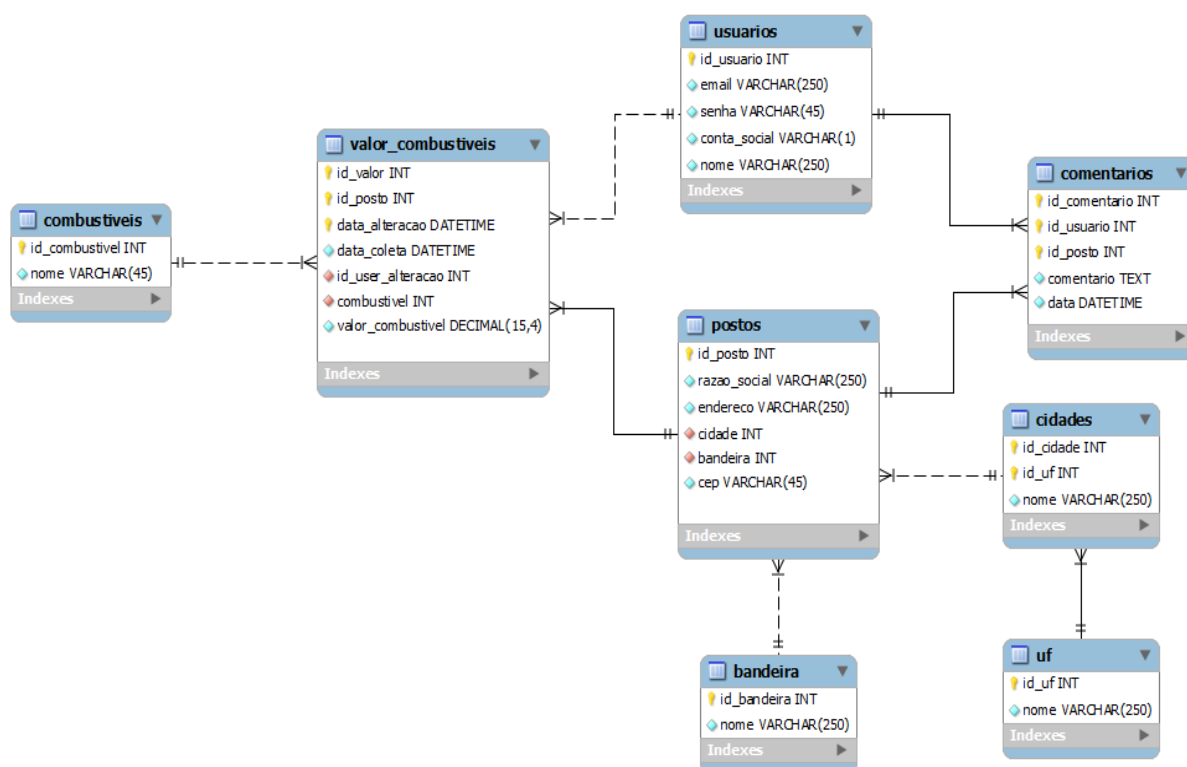


Figura 11 - Diagrama de Entidade Relacionamento

Fonte: Os Autores (2012)

4.3.3 Diagrama de Classe

Segundo Larman (2008, p.198) o diagrama de classe é utilizado para “ilustrar classes, interfaces e suas associações.” Eles são utilizados para modelagem dos objetos, proporcionando uma perspectiva conceitual do sistema e ajudando a obter uma visão panorâmica da dimensão do sistema e a interação entre os objetos.

O diagrama de classe demonstrado na figura 12 apresenta a arquitetura Gas-Finder, seguindo o MVC (*Model-View-Controller*), padrão de projeto popular em sistemas web.

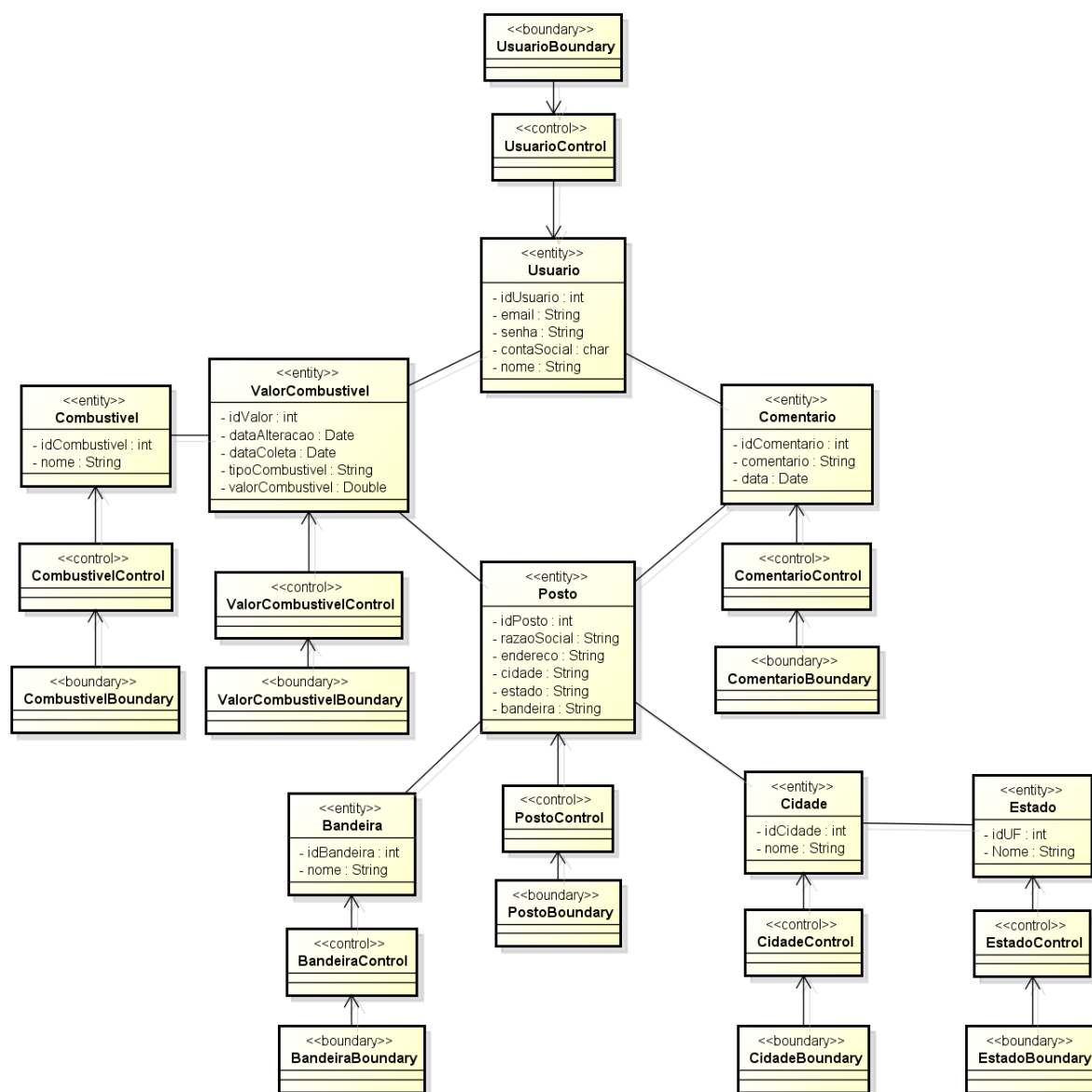


Figura 12 - Diagrama de Classe

Fonte: Os Autores (2012)

4.3.4 Diagrama de Seqüência

Segundo Larman (2008, 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 o comportamento dos objetos em um sistema, suas interações e colaborações, focando na ordem em que as mensagens são enviadas e recebidas.

A figura 13 é a representação gráfica do diagrama de seqüência Pesquisar / Detalhar / Alterar Valor do Combustível / Enviar Comentário.

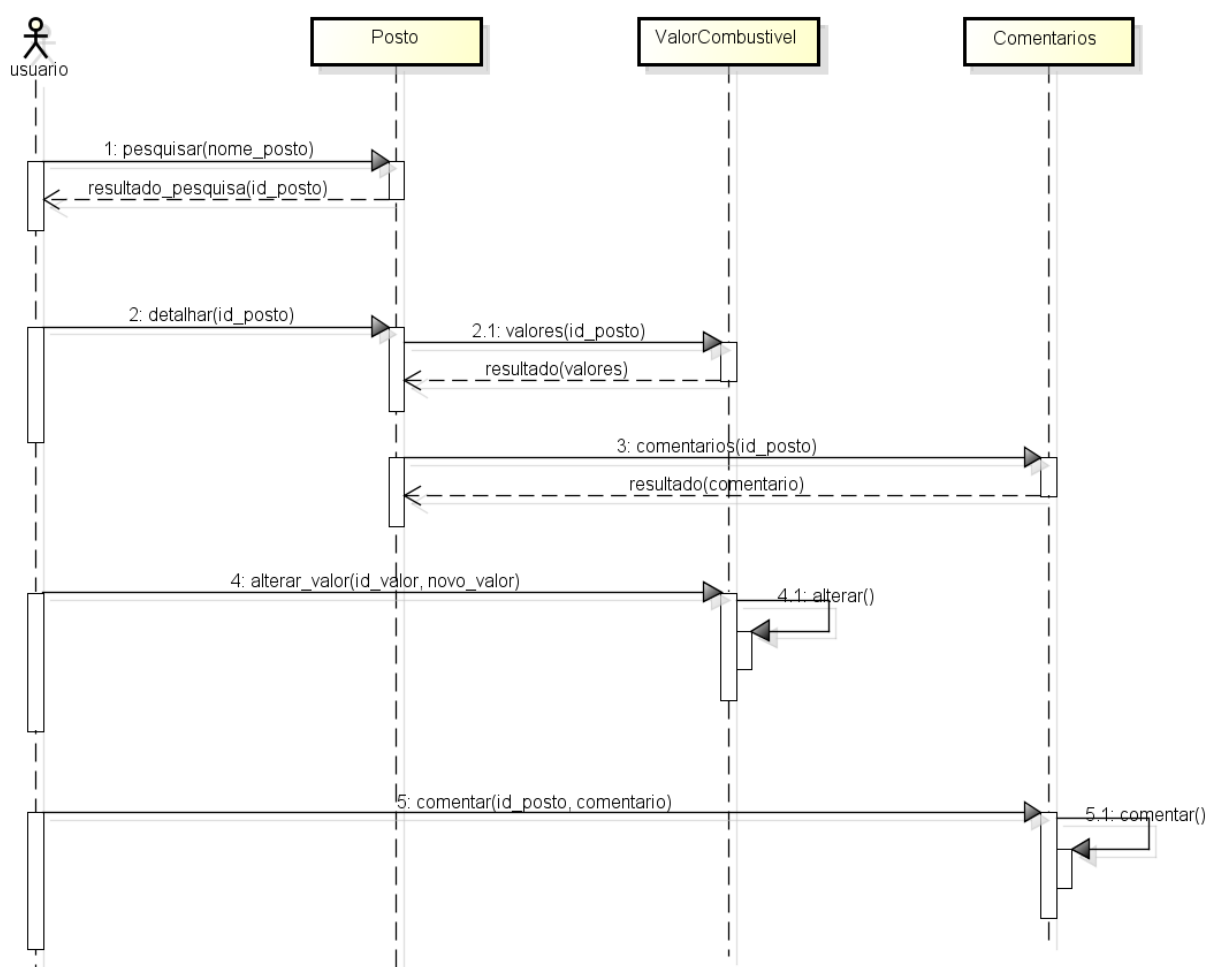


Figura 13 - Diagrama de Seqüência – Pesquisar, Detalhar, Alterar Valor do Combustível e Enviar Comentário

Fonte: Os Autores (2012)

4.3.5 Diagrama de Atividade

Segundo Larman (2008, p.482) “um diagrama de atividade UML mostra atividades seqüenciais e paralelas em um processo.” É útil para mostrar o fluxo de trabalho de um sistema e o fluxo dos dados, visando descrever as atividades em cada operação.

A figura 14 é a representação gráfica do diagrama de atividade de pesquisa, e demonstra as condições de interações disponíveis após uma pesquisa.

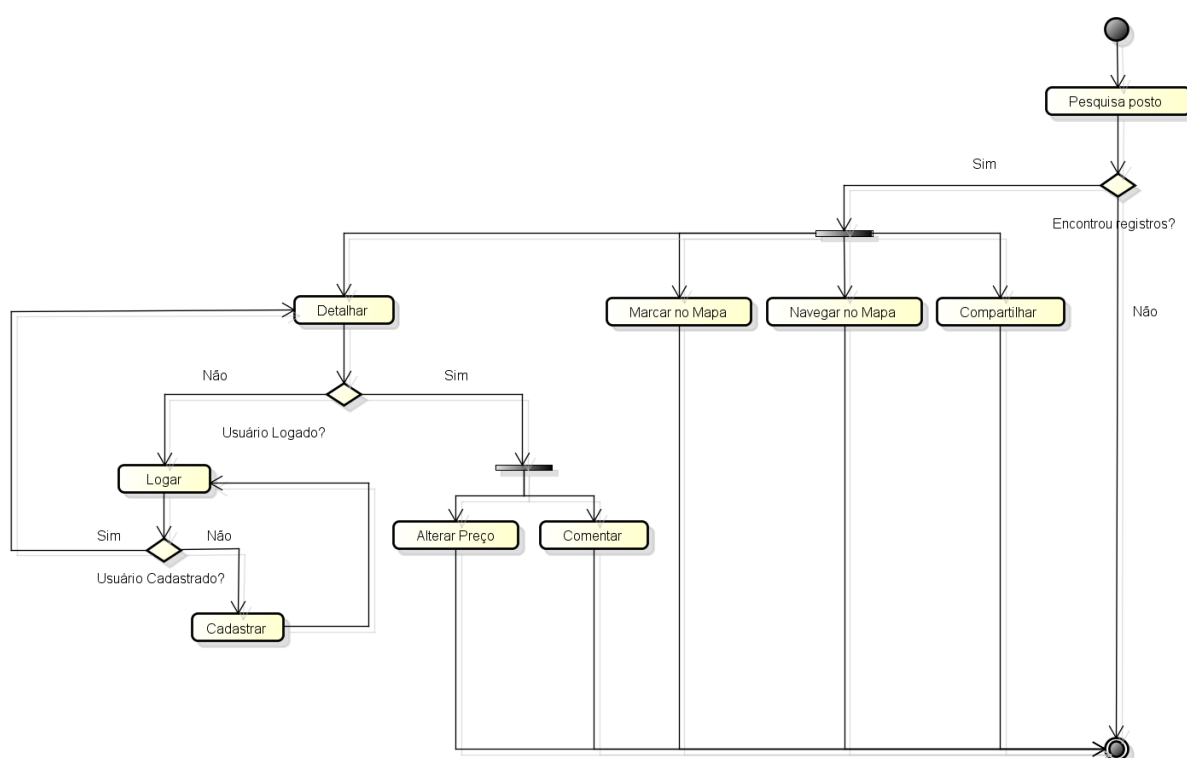


Figura 14 - Diagrama de Atividade - Pesquisar

Fonte: Os Autores (2012)

4.3.6 Wireframe

Os *wireframes* definem as telas que estarão disponíveis no sistema e as informações que elas apresentarão. A *wireframe* da tela inicial do GasFinder (index.xhtml) é definida em três áreas principais conforme é demonstrado na figura 15.

A área superior exibe o logo do sistema, seguido de um campo de busca e um *link* para *login* (login.xhtml), tela que concede acesso aos usuários cadastrados no Portal. Logo abaixo, na área central, estão localizados os blocos onde serão gerados os conteúdos dinamicamente, exibindo os preços dos combustíveis. E finalmente 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 apêndice deste trabalho.

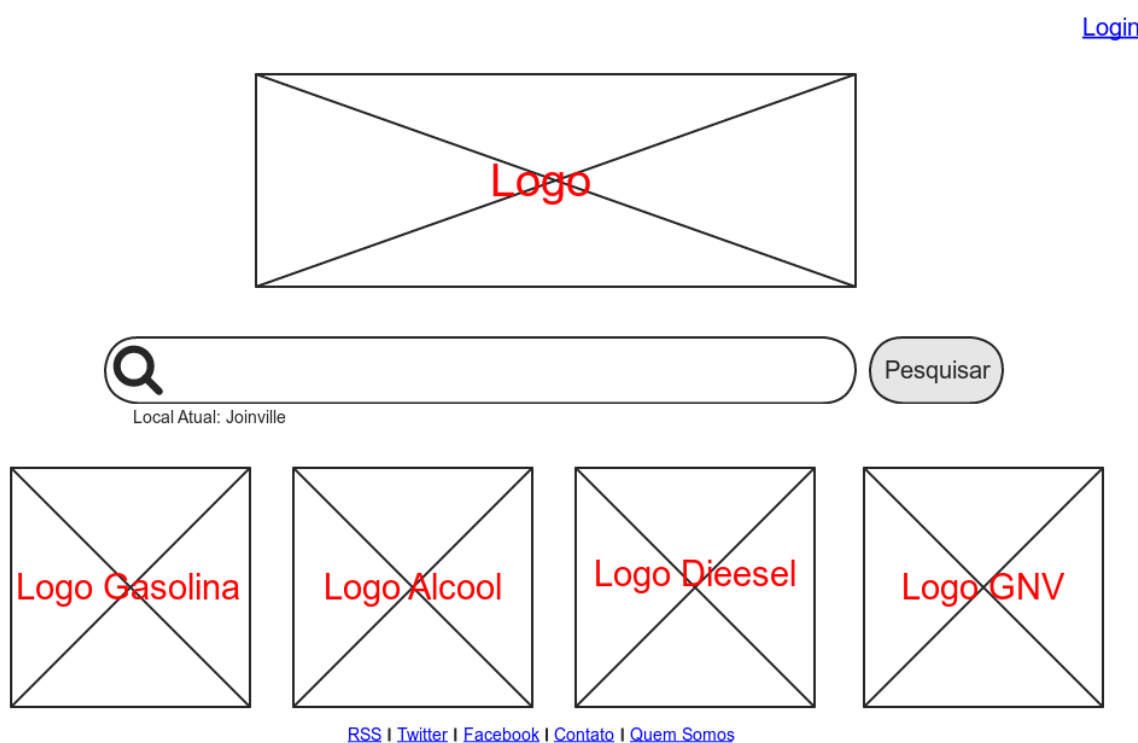


Figura 15 - Wireframe - Home

Fonte: Os Autores (2012)

4.4 FERRAMENTAS DE DESENVOLVIMENTO

Todas as ferramentas de desenvolvimento foram adotadas baseando-se na familiaridade da equipe, pela solidez das respectivas ferramentas no mercado, considerando sua vasta documentação e comunidade ativa.

Como IDE de codificação foi utilizado o Eclipse, que faz toda a organização dos recursos do projeto. Além de ajudar na produtividade, com funções de busca, auto-completar o código, realçar código e realizar a depuração do código. Foi utilizado Java como linguagem de programação, para codificação do sistema. Para auxiliar, facilitar e garantir qualidade na codificação da aplicação, também foram adotados alguns *frameworks*.

Conforme é demonstrado na figura 16, com auxílio do Spring Framework, foi utilizado o padrão de projetos MVC (*Model-View-Controller*). Na camada de *model*, camada responsável pela persistência dos dados e interação do sistema com o banco de dados, foi utilizado o Hibernate. Na camada de interface, camada responsável pela parte visual e de interação com o usuário, foi utilizada JSF através do Primefaces para gerar os componentes de tela, permitindo abstrair o uso do AJAX com o Java. Também foi utilizado HTML, CSS e JavaScript, para customizações visuais. Por sua vez, na camada de controle, camada responsável pela regra de negócio do sistema e ligações entre as demais camadas, também foi utilizado a Spring Framework.

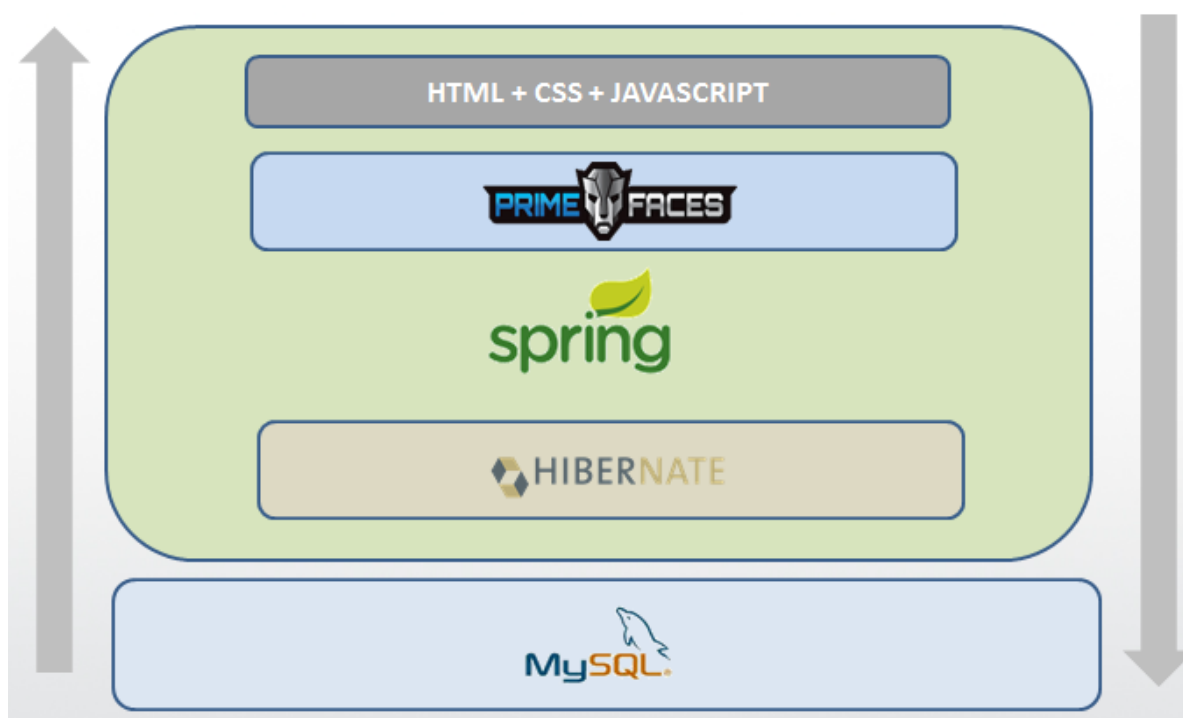


Figura 16 - Frameworks

Fonte: Os Autores (2012)

O MySQL foi utilizado como ferramenta de SGDB, 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, ferramenta leve e de fácil administração.

4.4.1 Extração de Dados

Como a ANP, não exporta os dados dos preços dos combustíveis e demais informações formatadas em um arquivo XML, JSON ou similar, foi necessário utilizar uma técnica conhecida como *data scraping*, que consiste na raspagem dos dados do *website*. Foi analisado sintaticamente o HTML do Portal ANP e obtido todas as informações necessárias, incluindo os preços dos postos de combustíveis. Para isso, foi utilizado o serviço *online* do site ScrapperWiki, um site onde as usuários podem desenvolver *scripts* para diversas finalidades utilizando a linguagem de programação Python.

Na figura 17, é exibido uma amostra do código do Scraper, que tem a finalidade de extrair dados de uma página HTML, através de um laço de programação, que executa uma chamada de função que adiciona os dados extraídos numa 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 17 - Amostra do código Scraper desenvolvido em Python

Fonte: Os Autores (2012)

Dessa forma, o Scraper extraí os dados do site da ANP, envia os dados para armazenamento no banco de dados do GasFinder onde finalmente os dados são disponibilizados ao usuário final do Portal. Esta dinâmica pode ser observada na figura 18.

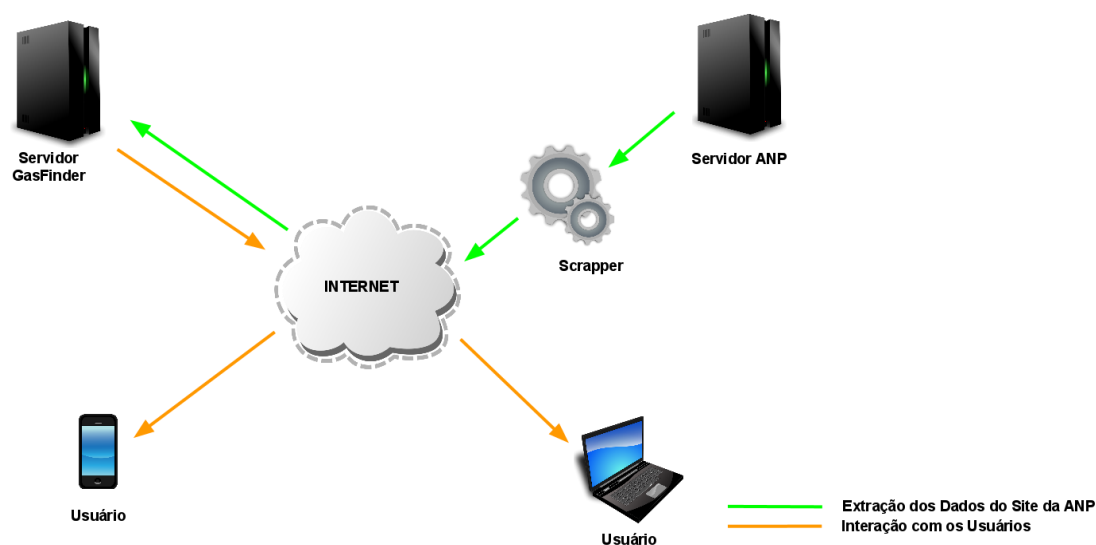


Figura 18 - Fluxo de informações do GasFinder

Fonte: Os Autores (2012)

4.5 APRESENTAÇÃO DO SISTEMA

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

4.5.1 Tela Inicial

O Portal 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 Portal. A tela inicial disponibiliza no topo, um *link* para *login* (`login.xhtml`) destinado aos usuários cadastrados e administradores do Portal. 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 visitante pode informar uma cidade onde deseja consultar e comparar os preços dos combustíveis. Ao selecionar o botão “buscar”, o sistema realizará uma busca do valor informado na tela, retornando todos os registros encontrados para uma tela de resultados da pesquisa (pesquisa.xhtml).

Na opção *login*, são validados dois tipos de usuários, o usuário cadastrado, e o administrador do sistema. Na figura 19 é demonstrado a tela inicial do GasFinder.

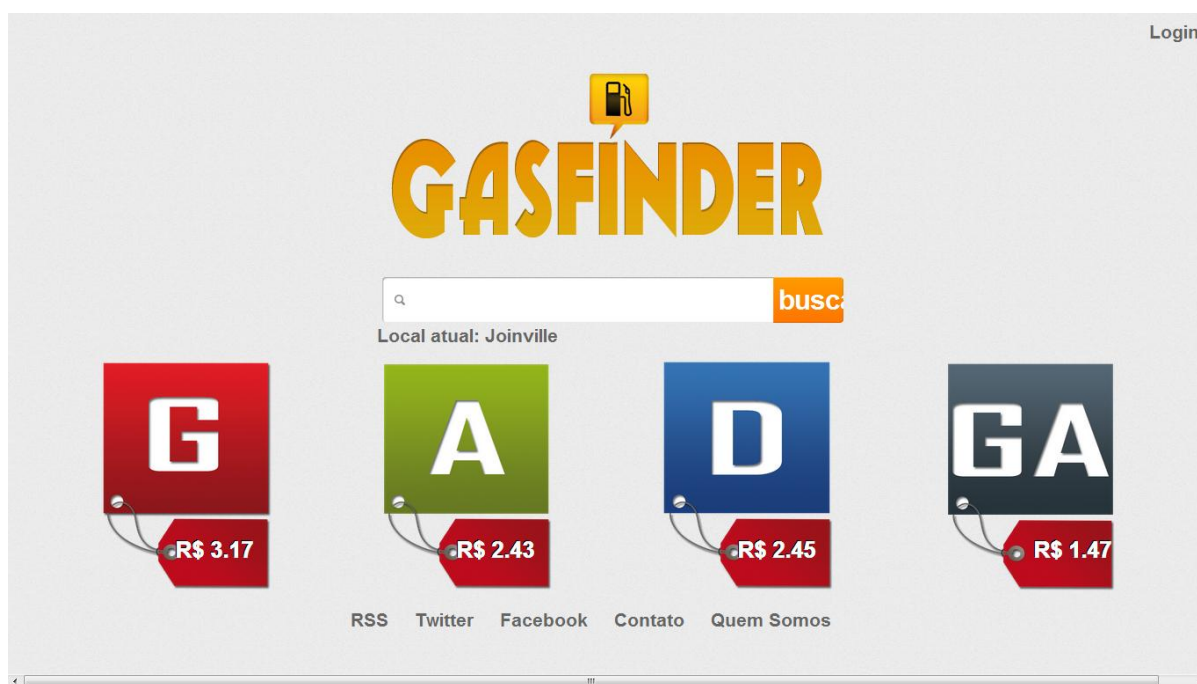


Figura 19 - GasFinder - Tela Inicial

Fonte: Os Autores (2012)

4.5.2 Tela de Resultado de Pesquisa

A tela de resultado de pesquisa (pesquisa.xhtml) é uma tela do resultado da busca inicial. 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” para ser redirecionado para a tela pesquisa.xhtml.

Na figura 20 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 20 - GasFinder - Tela de Resultado de Pesquisa

Fonte: Os Autores (2012)

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 região 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 a respectiva bandeira posto na cor laranja, 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 21 é demonstrada a interface da tela de busca no mapa.

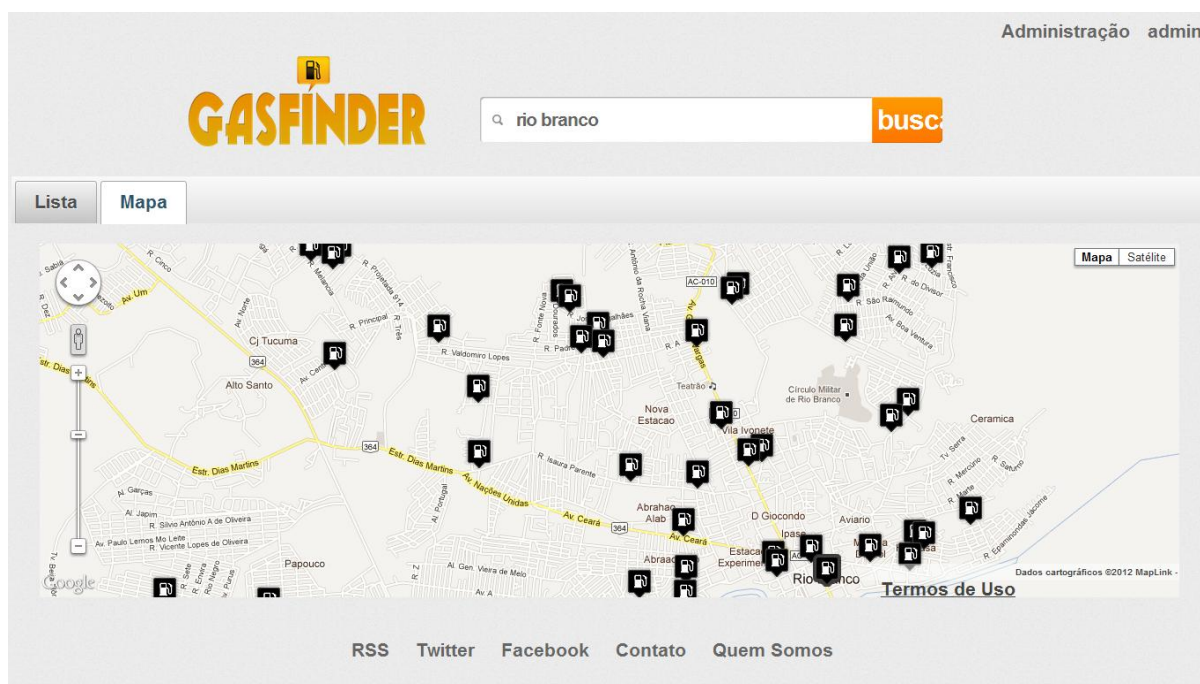


Figura 21 - GasFinder - Tela de Busca no Mapa

Fonte: Os Autores (2012)

4.5.4 Tela de Login

A tela de *login* (login.xhtml) é destinada a realizar a identificação de usuários cadastrados. O sistema permite acesso de dois perfis de usuário, usuário cadastrado e administrador. Ao informar o nome de usuário e a senha, o sistema verifica se as informações preenchidas em tela constam na base de dados e analisa se são verdadeiras.

O administrador possui acesso total ao sistema, 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, os usuários cadastrados possuem alguns 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* para a tela inicial do Portal (index.xhtml). Na figura 22 é demonstrada a interface da tela de login.

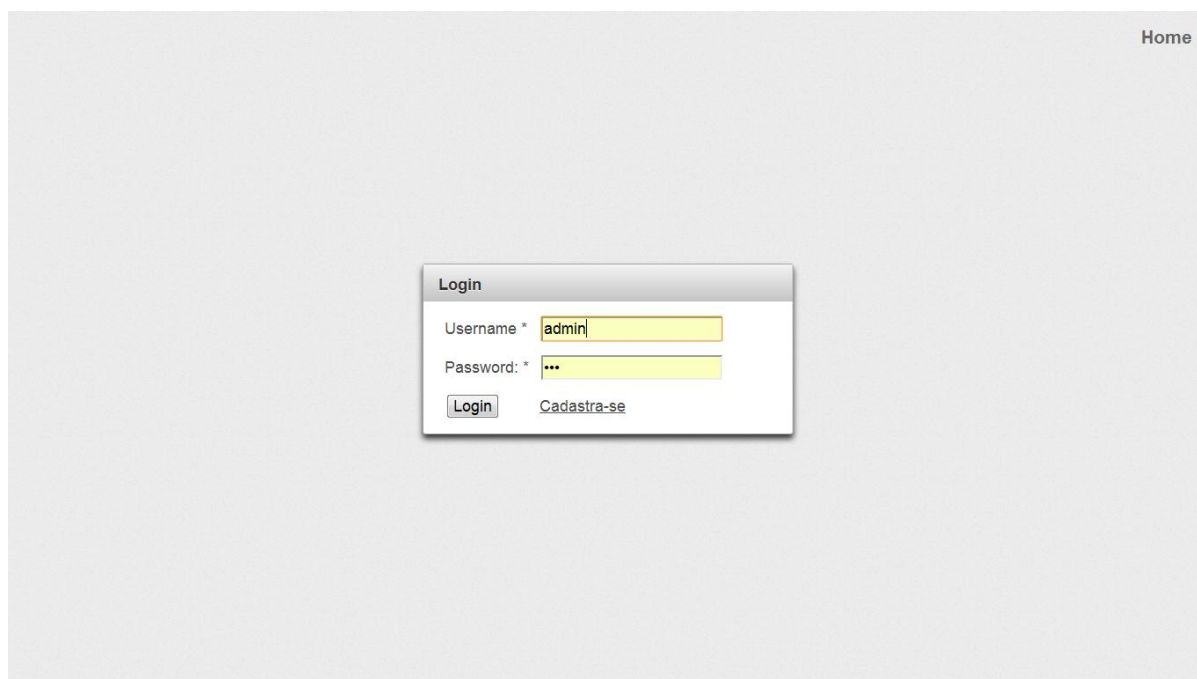


Figura 22 - GasFinder - Tela de Login

Fonte: Os Autores (2012)

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. Para alterar as informações de preços de um combustível, basta o usuário cadastrado ou administrador selecionar o texto do combustível desejado, e imediatamente o sistema 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 os usuários cadastrados ou um administrador poderão atualizar os preços dos combustíveis e adicionar comentários. O visitante terá acesso à 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).

Na figura 23 é demonstrada a tela de detalhamento do posto.

Home Administração admin

BR Auto Posto Rio Branco Derivados de Petróleo Ltda

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

| 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 23 - GasFinder - Tela de Detalhamento do Posto

Fonte: Os Autores (2012)

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. Oportunamente, poderá ser utilizado para enviar sugestão de melhoria, aviso de erros ou reclamações sobre o serviço do Portal.

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 24 é demonstrada a tela de contato.



Figura 24 - GasFinder - Tela de Contato

Fonte: Os Autores (2012)

Neste capítulo foi possível expor as etapas de desenvolvimento, incluindo a análise dos requisitos até apresentação do sistema, onde foram abordados os aspectos de interação e funcionalidades. Através da versão funcional do sistema, foi possível verificar na prática o resultado do desenvolvimento do GasFinder.

5 CONCLUSÃO

O desenvolvimento de software tem evoluindo constantemente e está se tornando um processo ainda mais dinâmico, com objetivo de se adequar as necessidades do mercado, rapidamente. Contudo, apesar de uma grande evolução e disponibilidade de ferramentas e metodologias de software, ainda existem limitações e falta de exatidão no processo de desenvolvimento.

Por este motivo, as metodologias classificadas como “ágeis”, como o XP e o Scrum, vem ganhando mais adeptos 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 solução está alinhada com os requisitos, evitando retrabalho e ganhando mais precisão nas horas de projeto planejadas/executadas.

Durante o desenvolvimento do trabalho foi constatada uma facilidade maior na mensuração de esforço em atividades planejadas para serem executadas com uma tecnologia com a qual a equipe já possuía conhecimento, principalmente a linguagem de programação Java e o banco de dados MySQL. A equipe também teve facilidade no uso dos recursos em Python em virtude da facilidade de uso e proposta da própria linguagem.

As telas desenvolvidas em HTML e CSS visaram facilitar a indexação através dos mecanismos de busca, como Google, o que entendemos que é vital para uma solução que possui grande volume de informações.

Como futura evolução do trabalho proposto, serão incluídas mais informações dos postos de combustíveis, como fotos e descrição de produtos e outros serviços oferecidos nos postos, como serviços de conveniência. O uso de mídias sociais, como Twitter para divulgação dos preços dos combustíveis também é um item que pode ser explorado ainda mais, pois já é uma solução comum na rotina dos usuários de internet e pode auxiliar na divulgação do Portal.

REFERÊNCIAS

ANP. **Agência Nacional do Petróleo, Gás Natural e Biocombustíveis**. Disponível em < <http://www.anp.gov.br> > Acesso em: 13 de Jan. 2012.

BEZERRA, E. **Princípios de análise e projeto de sistemas com UML**. Rio de Janeiro: Elsevier, 2002.

CADENHEAD, Rogers; LEMAY, Laura. **Aprenda em 21 dias Java 2**. Rio de Janeiro: Elsevier, 2005.

FURLAN, José Davi. **Modelagem de objetos através da UML - the Unified Modeling Language**. São Paulo: Makron-Books, 1998.

GOODMAN, Danny. **JavaScript a Bíblia**. Rio de Janeiro: Elsevier, 2001.

GRAEF, G.; GAEDKE, G. **Development and Evolution of Web-Applications using the WebComposition Process Model**. Amsterdam: The Netherlands, 2000.

HADOOP. **Apache Hadoop Core**. Disponível em < <http://hadoop.apache.org/> >. Acesso em: 13 de Jan. 2012.

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

LARMAN, Craig. **Utilizando UML e Padrões**. 3ª ed. Porto Alegre: Bookman, 2008.

LEMAY, Laura. **Aprenda a Criar Páginas com HTML e XHTML em 21 Dias**. São Paulo: Pearson Education do Brasil, 2002.

Luckow, Melo, A. **Programando Java para WEB**. São Paulo: Novatec, 2010.

LUTZ, Mark. ASCHER, David. **Aprendendo Python. 2ª ed.** Porto Alegre: Bookman, 2007.

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

MORAES, Lucas. **A vida com Scrum.** <<http://alemdati.wordpress.com/2010/03/26/a-vida-com-scrum/>> Acesso em: 13 de Jan. 2012.

PILGRIM, Mark. **Mergulhando no Python - O guia rápido e prático para dominar o Python.** Rio de Janeiro: Alta Books, 2005.

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

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

SOLARI, Marcelo Lins Baia. **Análise da Disponibilização de Arquitetura de Software como Serviço (SAAS - Software as a Service) Através de Aliança Estratégica: Um Estudo de Caso.** Brasília: Universidade de Brasília, 2008. Dissertação – Universidade de Brasília, Brasília, 2008.

SILVA, M. S. **Construindo sites com CSS e (X) HTML. 1ª ed.** São Paulo: Novatec, 2007.

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

Tomcat, Apache. Version 7.0. **The Apache Software Foundation.** Disponível em: < <http://tomcat.apache.org/> >. Acesso em: 13 de Jan. 2012.

ULLMAN, L. **MySQL**. 2^a ed. Canadá: Peachpit Press, 2006.

VISION, I. C. **JUDE: UML, ER, CRUD, DFD, Flowchart and Mind Map: Design and Modeling Tool**. 2011. Disponível em: <<http://jude.change-vision.com/jude-web/index.html>>. Acesso em: 13 de Jan. de 2012.

APÊNDICE

APÊNDICE 1 – WIREFRAME – HOME

APÊNDICE 2 – WIREFRAME – CONTATO

APÊNDICE 3 – WIREFRAME – QUEM SOMOS

APÊNDICE 4 – WIREFRAME – LOGIN

APÊNDICE 5 – WIREFRAME – PESQUISA NA LISTA DE PRODUTOS

APÊNDICE 6 – WIREFRAME – SEM RESULTADOS PARA PESQUISA

APÊNDICE 7 – WIREFRAME – PESQUISA DIRETAMENTE NO MAPA

APÊNDICE 8 – WIREFRAME – MANUTENÇÃO DE PREÇOS

APÊNDICE 9 – WIREFRAME – INFORMAÇÕES DOS POSTOS

APÊNDICE 10 – WIREFRAME – ERRO GENÉRICO

APÊNDICE 1 – WIREFRAME – HOME

A tela principal do Portal 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, Dieesel e GNV, conforme demonstrado na figura 25.

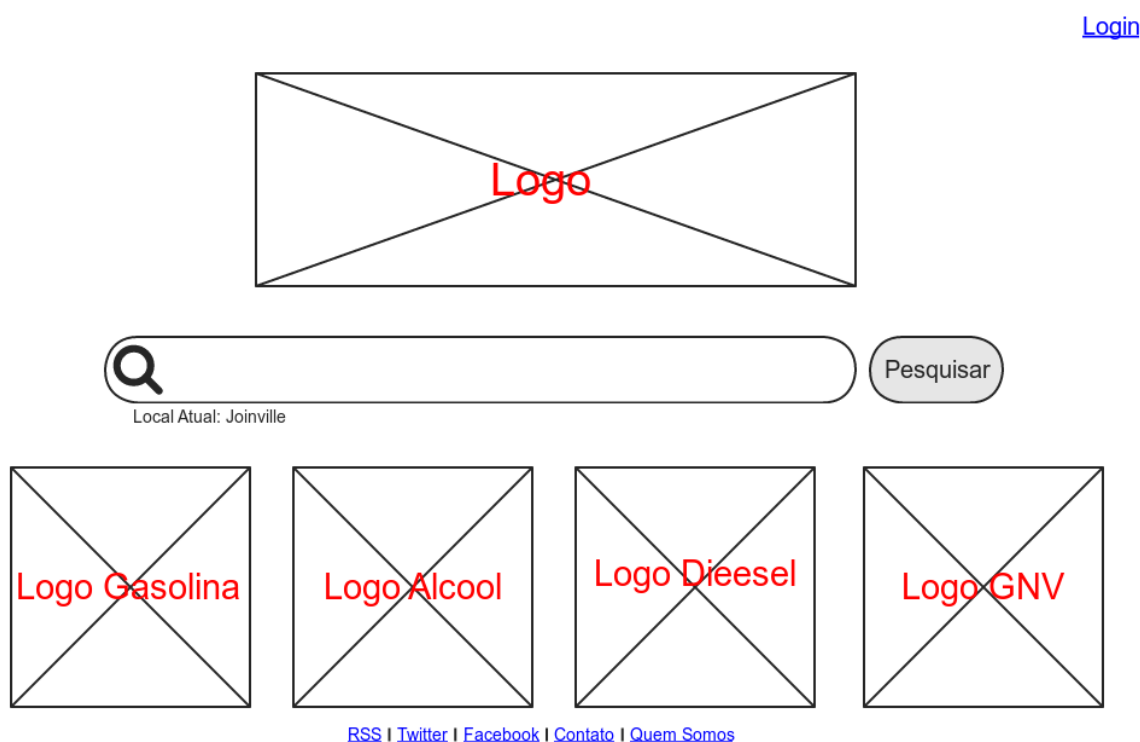


Figura 25 - Wireframe - Home

Fonte: Os Autores (2012)

APÊNDICE 2 – WIREFRAME – CONTATO

A tela de Contato fornece um formulário onde o usuário poderá postar suas dúvidas e comentários acerca dos serviços do Portal. A interface é proposta conforme a figura 26.

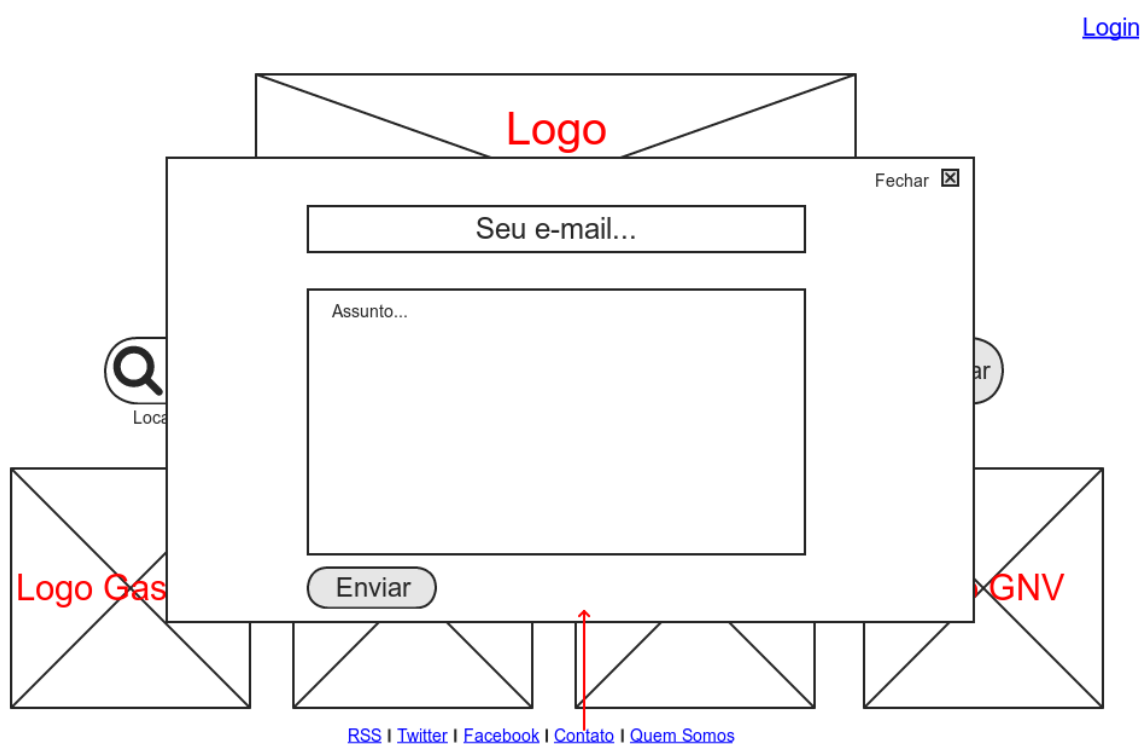


Figura 26 - Wireframe - Contato

Fonte: Os Autores (2012)

APÊNDICE 3 – WIREFRAME – QUEM SOMOS

A tela de Quem Somos visa anunciar as principais informações acerca dos serviços do Portal, o objetivo do serviço e outras informações dos colaboradores. A interface é proposta conforme a figura 27.

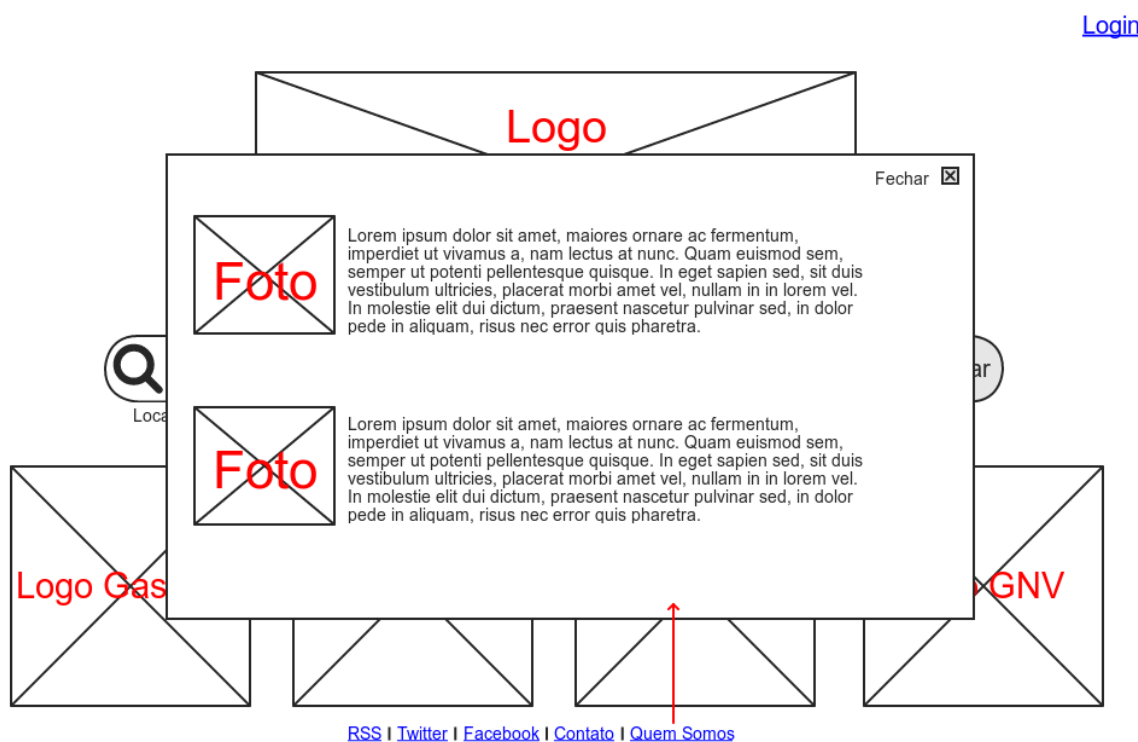


Figura 27 - Wireframe - Quem somos

Fonte: Os Autores (2012)

APÊNDICE 4 – WIREFRAME – LOGIN

A tela de Login permite que o administrador acesse a interface de gerenciamento do sistema, ou o usuário cadastrado acesse sua conta no Portal, conforme é demonstrado na figura 28.

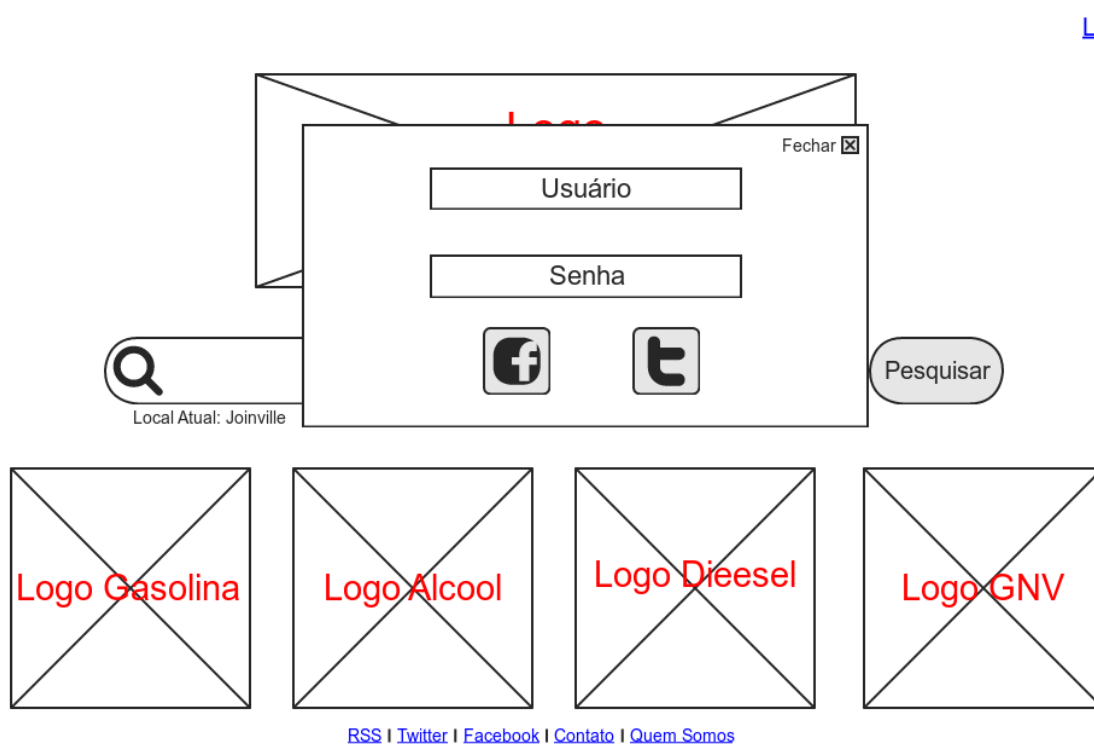


Figura 28 - Wireframe - Login

Fonte: Os Autores (2012)

APÊNDICE 5 – 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 29.

[Login](#)

Visualizar como ☒ Lista ☐ Mapa

| | | | | | | | | |
|-----------------------|-------------------------------------|----------|-------------------------------------|----------|-------------------------------------|----------|-------------------------------------|----------|
| Posto XYZ LTDA | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 |
| Rua Do Biro Biro, 999 | | | | | | | | |
| Posto XYZ LTDA | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 |
| Rua Do Biro Biro, 999 | | | | | | | | |
| Posto XYZ LTDA | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 |
| Rua Do Biro Biro, 999 | | | | | | | | |
| Posto XYZ LTDA | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 |
| Rua Do Biro Biro, 999 | | | | | | | | |
| Posto XYZ LTDA | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 |
| Rua Do Biro Biro, 999 | | | | | | | | |
| Posto XYZ LTDA | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 | <input checked="" type="checkbox"/> | R\$ 9,99 |
| Rua Do Biro Biro, 999 | | | | | | | | |

1 | 2 | 3 | 4 | 5 | [Mais](#)

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

Figura 29 - Wireframe - Pesquisa na lista de Postos

Fonte: Os Autores (2012)

APÊNDICE 6 – 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 30.

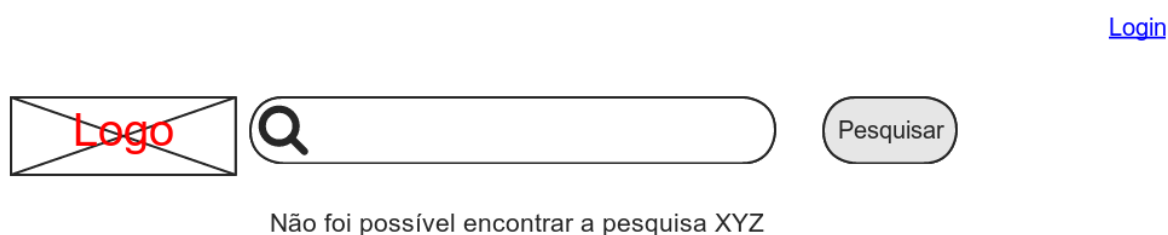


Figura 30 - Wireframe - Sem resultados para pesquisa

Fonte: Os Autores (2012)

APÊNDICE 7 – WIREFRAME – PESQUISA DIRETAMENTE NO MAPA

Esta tela permite que o usuário navegue diretamente no mapa, deslizando a para uma área desejada e visualizando graficamente os postos disponíveis em uma determinada região. O modelo proposto é exibido na figura 31.

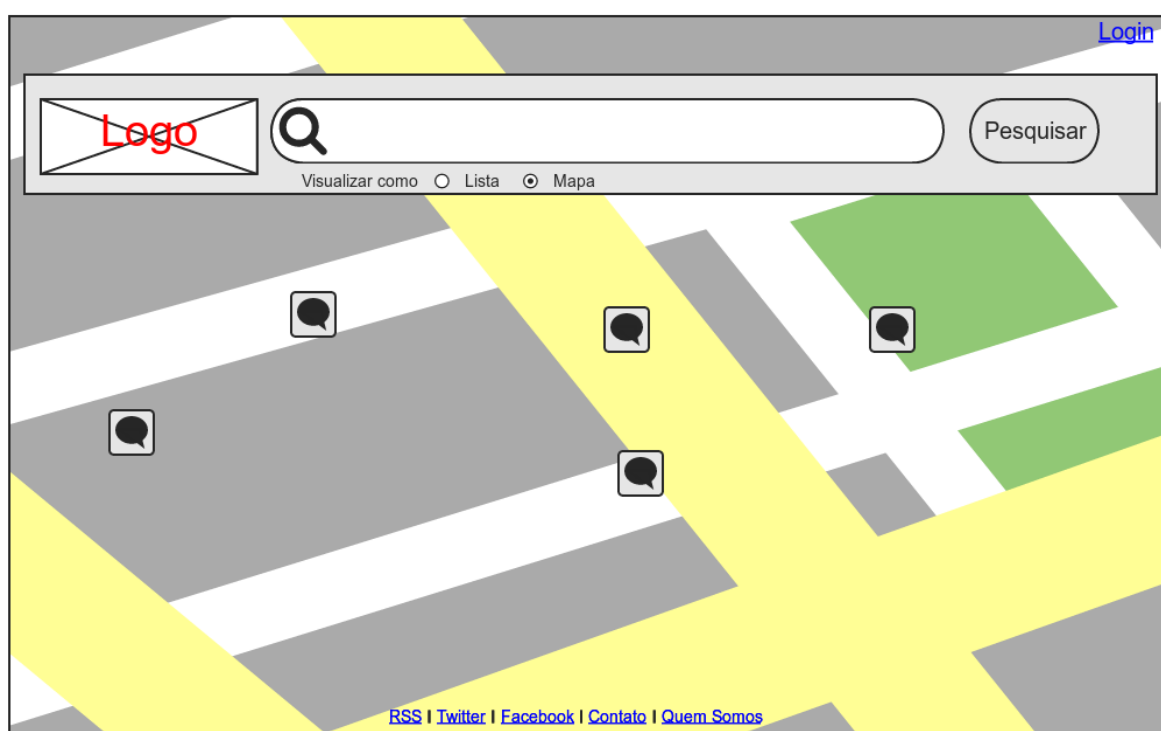


Figura 31 - Wireframe - Pesquisa diretamente no mapa

Fonte: Os Autores (2012)

APÊNDICE 8 – 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 32.

[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

Fechar ☐

| | | |
|--------------------------|----------|-----------------------------------|
| <input type="checkbox"/> | Gasolina | <input type="text" value="9,99"/> |
| <input type="checkbox"/> | Alcool | <input type="text" value="9,99"/> |
| <input type="checkbox"/> | Dieseel | <input type="text" value="9,99"/> |
| <input type="checkbox"/> | GNV | <input type="text" value="9,99"/> |


Atualizar

Figura 32 - Wireframe - Manutenção de Preços

Fonte: Os Autores (2012)


APÊNDICE 9 – WIREFRAME – INFORMAÇÕES DOS POSTOS

Nesta tela é possível realizar um detalhamento das informações de cada Posto, visualizando comentário de outros usuários e também consultando o histórico dos preços praticados por um determinado Posto, conforme proposto na figura 33.




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

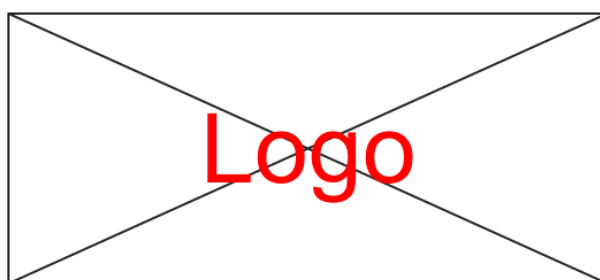
[RSS](#) | [Twitter](#) | [Facebook](#) | [Contato](#) | [Quem Somos](#)

Figura 33 - Wireframe - Informações dos Postos

Fonte: Os Autores (2012)

APÊNDICE 10 – 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 34.



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

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

Figura 34 - Wireframe - Erro genérico

Fonte: Os Autores (2012)