

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

DEPARTAMENTO ACADÊMICO DE INFORMÁTICA
CURSO DE ESPECIALIZAÇÃO EM TECNOLOGIA JAVA

GLÁUCIA MARIA PASTRO MENDES

**SISTEMA MOBILE WEB
PARA BUSCA GEORREFERENCIADA DE IMÓVEIS**

MONOGRAFIA DE ESPECIALIZAÇÃO

CURITIBA - PR

2011

GLÁUCIA MARIA PASTRO MENDES

**SISTEMA MOBILE WEB
PARA BUSCA GEORREFERENCIADA DE IMÓVEIS**

Monografia de Especialização apresentada ao Departamento Acadêmico de Informática, da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção do título de “Especialista em Tecnologia Java”.
Orientador: Prof. Robson Ribeiro Linhares
Co-orientador: Prof. Paulo Bordin

CURITIBA - PR

2011

AGRADECIMENTOS

Pela oportunidade de crescimento pessoal e profissional, agradeço a todas as pessoas que direta ou indiretamente auxiliaram na realização deste trabalho.

RESUMO

Este trabalho apresenta um estudo sobre ferramentas de desenvolvimento *mobile web*, com destaque para o framework jQuery Mobile e a API Google Maps. Aqui são apresentadas as ferramentas e tecnologias e sua aplicação para a construção de um sistema de busca de imóveis através da Geolocalização do usuário, intitulado *Cadê Meu Apê*. Este sistema apresenta como principal característica a utilização de uma interface que se ajusta tanto para a visualização em computadores desktops como em dispositivos móveis.

Palavras-chave: Mobile, Web, Geolocalização, jQuery, Google Maps API.

ABSTRACT

This work presents a study on *mobile* web development tools, with emphasis on the jQuery framework API and Google Maps Mobile. Here are given the tools and technologies and their application for the construction of a system of real estate search by user Geolocation, entitled *Cadê Meu Apê*. This system has as main characteristic the use of an interface that fits well for viewing on desktops and *mobile* devices.

Keywords: *Mobile*, Web, Geolocation, jQuery, Google Maps API.

LISTA DE FIGURAS

Figura 1 – Matriz de compatibilidade de navegadores móveis	17
Figura 2 – Componentes de interface jQuery Mobile	20
Figura 3 – Diagrama de Caso de Uso	28
Figura 4 – Diagrama de Classes	29
Figura 5 – Diagrama de Sequencia – Caso de Uso “Entrar em Contato”	29
Figura 6 – Diagrama de Sequencia – Caso de Uso “Pesquisar Imóvel”	30
Figura 7 – Utilização do JQM em uma página HTML	32
Figura 8 – Trecho de código HTML da página inicial	33
Figura 9 – Trecho de código HTML da página de busca	33
Figura 10 – Tela inicial do sistema (navegador desktop Mozilla Firefox 6.x)	34
Figura 11 – Tela inicial do sistema (navegador <i>mobile</i>)	35
Figura 12 – Tela de contato (navegador desktop Mozilla Firefox 6.x)	35
Figura 13 – Tela de contato (navegador <i>mobile</i>)	36
Figura 14 – Tela de pesquisa (navegador desktop Mozilla Firefox 6.x)	36
Figura 15 – Tela de pesquisa (navegador <i>mobile</i>)	37
Figura 16 – Tela de resultado (navegador desktop Mozilla Firefox 6.x)	37
Figura 17 – Tela de resultado (navegador <i>mobile</i>)	38

LISTA DE TABELAS

Tabela 1 – Características da Web <i>Mobile</i> e da Web Desktop.....	13
Tabela 2 – Tabela de compatibilidade entre navegadores e o HTML5.....	24
Tabela 3 – Descrição do caso de uso “Pesquisar Imóvel”.....	31
Tabela 4 – Descrição do caso de uso “Entrar em Contato”.....	32

LISTA DE ABREVIATURAS E SIGLAS

LCD – *Liquid Crystal Display* (Display de cristal líquido)

CSS – *Cascading Style Sheets* (Folhas de Estilo Encadeadas)

IDE – *Integrated Development Environment* (Ambiente Integrado para Desenvolvimento)

HTML – *HyperText Markup Language* (Linguagem de Marcação de Hipertexto)

UML – *Unified Modeling Language* (Linguagem de Modelagem Unificada)

SUMÁRIO

1. INTRODUÇÃO	11
1.1 Objetivo	11
1.2 Escopo	12
1.3 Organização	12
2. ESTUDO DAS TECNOLOGIAS	13
2.1 Mobile Web versus Desktop Web	13
2.1.1 Características de aplicações Web Desktop.....	14
2.1.2 Características de aplicações Web <i>Mobile</i>	14
2.1.3 Usuários	15
2.1.4 Mercado de Trabalho	15
2.1.5 Ferramentas.....	15
2.2 jQuery Mobile	16
2.2.1 Compatibilidade.....	16
2.2.2 Características	18
2.2.3 Estrutura de uma página jQuery Mobile.....	18
2.2.4 Transição de Páginas	19
2.2.5 Temas.....	19
2.2.6 Formulários	19
2.3 Google Maps API.....	21
2.3.1 Google Maps JavaScript API.....	21
2.3.2 Tipos de Mapas	22
2.4 HTML5.....	23
2.4.1 Modularização e Compatibilidade	24
2.4.2 Elementos Novos, Modificados ou Ausentes	24
2.4.3 Novos Tipos de Campos	25
2.4.4 Geolocalização com HTML5.....	25
3. DESENVOLVIMENTO	27
3.1 Especificação de Requisitos	27
3.1.1 Requisitos Funcionais	27
3.1.2 Requisitos Não Funcionais	27
3.1.3 Restrições.....	27
3.2 Modelagem do Sistema	28
3.2.1 Diagrama de Casos de Uso	28
3.2.2 Diagrama de Classes	28
3.2.3 Diagrama de Sequência	29
3.2.4 Descrição de Casos de Uso	30
3.3 Implementação.....	32
3.3.1 Instalação do jQuery Mobile.....	32
3.3.2 Design das telas com HTML5	33
3.3.3 Obtendo a localização do usuário	34
3.4 Apresentação do Sistema Desenvolvido	34
3.4.1 Tela Inicial	34
3.4.2 Tela de Contato	35
3.4.3 Tela de Pesquisa de Imóveis	36

3.4.4	Tela de Resultado da Pesquisa.....	37
4.	CONCLUSÃO	39
4.1	Projetos Futuros.....	39
	REFERÊNCIAS	41

1. INTRODUÇÃO

O crescimento do crédito imobiliário no Brasil tem movimentado o mercado de imóveis. Com o aumento da procura, a internet se tornou o principal meio de divulgação de imóveis e um importante canal de relacionamento entre imobiliárias, clientes e corretores.

Segundo Sandra Turchi, Superintendente de Marketing da ACSP - Associação Comercial de São Paulo e SCPC – Serviço Central de Proteção ao Crédito (Revista Eletrônica PEGN, 2010) “cerca de 88% das pessoas que adquirem quaisquer produtos pesquisam na web antes de realizar suas compras. Dos que adquirem imóveis, são mais de 90%, pois isso facilita a seleção prévia daqueles que serão visitados, de acordo com faixa de preço, localização, tamanho e todas as especificações que o interessado buscar.”

Paralelo ao crescimento expressivo do mercado imobiliário, a tecnologia móvel se expande, os *smartphones* conquistam os consumidores e a demanda por aplicativos *mobile* cresce à medida que os usuários se familiarizam com a possibilidade do acesso à internet em qualquer lugar, a qualquer momento. Os dispositivos móveis tem se apresentado como peça fundamental em nossas vidas, facilitando a comunicação em vários aspectos, capaz de nos divertir, informar, facilitar serviços e orientar nossos caminhos.

Com a popularidade dos dispositivos móveis, surgirão muitas tecnologias para criar aplicações e sites para as minúsculas telas de LCD. Com isso, a capacidade de desenvolver aplicações que funcionem tanto no desktop/web como em dispositivos *mobile* será um grande diferencial no mercado.

1.1 Objetivo

O presente trabalho propõe a construção de uma ferramenta de busca de imóveis baseada em localização geográfica, que possa ser acessada a partir de qualquer dispositivo com acesso à internet, permitindo assim uma busca com resultados mais precisos e que atenda os mais diversos perfis de clientes.

O principal objetivo deste projeto é estudar o potencial de algumas ferramentas de desenvolvimento e a aplicação prática de uma solução para um mercado em expansão. O produto deste estudo disponibilizará ao usuário de dispositivos móveis, uma maneira diferenciada para buscar imóveis, isto é, o usuário poderá filtrar a busca por pontos

específicos no mapa, evitando a perda de tempo com informações de imóveis que não se encaixam no seu perfil de interesses.

Outro ponto importante que será explorado é a construção de uma aplicação *web/mobile* independente de plataforma, ou seja, ao invés de desenvolver um sistema específico para cada tipo de dispositivo móvel ou sistema operacional, será utilizado um framework que permitirá desenvolver uma aplicação customizada que funcionará nas mais populares plataformas de *smartphones* e *tablets*.

1.2 Escopo

O escopo deste trabalho concentra-se no estudo e utilização de ferramentas de desenvolvimento para a construção de um protótipo que permitirá a consulta de imóveis a partir de uma localização geográfica.

O sistema não tratará da manutenção das tabelas de apoio, como a inserção dos imóveis na base de dados. Também não faz parte do escopo a autenticação do usuário.

1.3 Organização

O presente documento está organizado da seguinte maneira: No Capítulo 2 são apresentadas e exploradas as ferramentas utilizadas para o desenvolvimento do projeto. O Capítulo 3 apresenta os detalhes do desenvolvimento do protótipo com exemplos conceituais da utilização das ferramentas escolhidas. Finalmente, o Capítulo 4 descreve as conclusões do trabalho realizado.

2. ESTUDO DAS TECNOLOGIAS

2.1 Mobile Web versus Desktop Web

Aplicações que tem a web como base, utilizam um browser (navegador) como porta de entrada (*client-side*) para o acesso a servidores que processam as requisições da aplicação (*server-side*). Portanto, os conceitos para o desenvolvimento *mobile web* partem dos mesmos princípios do desenvolvimento web, porém com algumas limitações de hardware e software que geram diferentes problemas, demandando novas soluções.

	Web Mobile	Web Desktop
Duração média	2 - 3 minutos	10 - 15 minutos da sessão
Tamanho mínimo da tela	90 x 60	800 x 600
Tamanho máximo	240 x 400 para dispositivos populares	Sem limite da tela
Revendedores do navegador	+ 12 e crescendo	Dois com fatia de mercado acima de 5%
Erros do navegador	Frequente. Permanente, exceto para smartphone com SO atualizável.	Raros e corrigíveis.
Padrões W3C	Irregulares. Algumas vezes ignorados ou desafiados pela indústria de dispositivo móvel.	Maduros e aceitos.
Linguagens de marcação	WML CHTML XHTML HTML	XHTML e HTML
JavaScript e Ajax	Não em 90% dos dispositivos móveis. Disponível como ECMAScript-MP e JavaScript. O Modelo de Objetos do Documento (DOM) e os eventos suportados variam. As APIs patenteadas são comuns.	Geralmente disponível
Cientes endereçáveis	Três bilhões de assinantes em todo o mundo	Um bilhão de notebooks, computadores de mesa e servidores no total.

Tabela 1 – Características da Web *Mobile* e da Web Desktop.

Fonte: [FREDERICK, 2011]

2.1.1 Características de aplicações Web Desktop

- **Resolução:** Apesar de existir diferentes resoluções de tela para diferentes áreas de trabalho, essa quantidade é bem administrada pelos desenvolvedores. O padrão atual de resolução utilizada ou suportada pela maioria dos computadores é de 1024 x 768 pixels.
- **CSS e Flash:** Os browsers desktop mais utilizados atualmente, suportam CSS e Flash.
- **Navegação:** Recursos como a navegação por abas e menus *drop-down* são utilizados para tornar a navegação mais intuitiva e enriquecer a experiência dos usuários no site.
- **Conteúdo:** Aplicações desktop podem suportar grandes volumes de conteúdo, como imagens e gráficos.

2.1.2 Características de aplicações Web Mobile

- **Resolução:** O tamanho da tela de dispositivos móveis varia muito de acordo com o modelo. Desta forma, uma aplicação web será renderizada de diferentes formas, de acordo com as configurações de resolução do aparelho.
- **CSS e Flash:** Alguns navegadores de celular não são capazes de renderizar elementos CSS e Flash, portanto, web sites que utilizam estas tecnologias não poderão ser visualizados corretamente nestes dispositivos.
- **Navegação:** A navegação em dispositivos móveis deve ser mais simples e descomplicada possível, por causa das limitações de recursos de navegação e do tamanho da tela.
- **Conteúdo:** Também pelo motivo de limitação do tamanho da tela, as aplicações para dispositivos móveis devem apresentar somente o conteúdo mais importante.

2.1.3 Usuários

Diferente do usuário desktop, que além de acessar uma aplicação, ouve música, verifica e-mail, Orkut (www.orkut.com), Twitter (www.twitter.com), tudo ao mesmo tempo, o usuário *mobile* ainda não é multitarefa, quando ele busca uma informação não quer encontrar barreiras para chegar lá. Usuários *mobile* sempre procuram algo específico e podem desistir de acessar um sistema rapidamente.

Com o tempo os usuários *mobile* tendem a evoluir o seu comportamento para alguns aspectos que os usuários desktop já praticam como fazer compras online por exemplo. Atualmente ainda é aceitável um site não ter uma versão *mobile*, mas com o avanço rápido da tecnologia, isso vai durar pouco tempo. A partir do momento que o concorrente tiver um site *mobile*, os clientes questionarão por que aquele site ainda não tem.

2.1.4 Mercado de Trabalho

Quando uma nova tendência tecnológica começa a se popularizar, o mercado de trabalho abre as portas para absorver as novas demandas.

Serviços *Mobile* Marketing, Sites *Mobile*, Serviços baseados em localização dentre muitos outros estão criando uma nova demanda de profissionais no mercado, que tenham o conhecimento de todos os recursos que esses dispositivos móveis podem oferecer.

2.1.5 Ferramentas

As ferramentas para desenvolvimento *Mobile* Web são as mesmas utilizadas para desenvolvimento desktop: IDE, Banco de dados e Framework basicamente. Para o desenvolvimento *mobile*, basta acrescentar um ambiente de teste, que envolve vários aparelhos de celular ou um simulador.

Com o passar do tempo estão surgindo frameworks e ferramentas específicas para desenvolvimento *mobile*, como o jQuery Mobile, tema de estudo deste trabalho.

2.2 jQuery Mobile

Baseado no jQuery, framework JavaScript já consagrado no mercado, o jQuery Mobile (JQM) surgiu em agosto de 2010.

A principal abordagem do JQM é atingir a maior quantidade de plataformas móveis possíveis. Ele trata os navegadores nos dispositivos móveis exatamente da mesma forma que trata os navegadores no desktop. O framework atua sobre as páginas HTML já existentes e simula uma interface *mobile*. Isso garante a compatibilidade com praticamente qualquer dispositivo habilitado para a web. A proposta do jQuery Mobile é a criação de sites otimizados para navegação com toque.

A plataforma JQM fornece uma série de eventos personalizados, permitindo que aplicações web, quando visualizadas em um navegador *mobile*, realizem ações específicas, como toque e mudança de orientação da tela.

2.2.1 Compatibilidade

Ao invés de escrever aplicações específicas para cada tipo dispositivo móvel ou sistema operacional, o framework jQuery Mobile permite desenvolver aplicações web customizadas que funcionam nas mais populares plataformas de *smartphones* e *tablets*.

Disponível na versão 1.0 Beta 1, o jQuery Mobile já é compatível com uma grande quantidade de plataformas móveis. A documentação disponível apresenta uma relação das plataformas já testadas:

- Apple iOS: iPhone, iPod Touch, iPad (todas as versões)
- Android: todos os dispositivos (todas as versões)
- Blackberry Torch™ (versão 6)
- Palm WebOS Pre, Pixi
- Nokia N900 (em fase de teste)

Versões mais antigas do Blackberry, Nokia / Symbian e Windows Mobile também são compatíveis, mas ainda apresentam erros de JavaScript e bugs de renderização que estão sendo corrigidos.

A Figura 1 apresenta uma matriz de compatibilidade entre as versões do jQuery Mobile, as plataformas e os browsers testados.

MOBILE GRADED BROWSER SUPPORT													
Platform	Version	Native	Opera Mobile				Opera Mini		Fennec		Ozone	Netfront	Phonegap
			8.5	8.65	9.5	10.0	4.0	5.0	1.0	1.1	0.9	4.0	0.9
iOS	v2.2.1	B										A	
	v3.1.3, v3.2	A					A					A	
	v4.0	A					A					A	
Symbian S60	v3.1, v3.2	C	C	C		B	C	B		C	C		
	v5.0	A	C	C		A	C	A				A	
Symbian UIQ	v3.0, v3.1			C						C			
	v3.2			C						C			
Symbian Platform	3.0	A											
BlackBerry OS	v4.5	C					C	C					
	v4.6, v4.7	C					C	B				C	
	v5.0	B					C	A				A	
	v6.0	A						A				A	
Android	v1.5, v1.6	A										A	
	v2.1	A										A	
	v2.2	A				A		C		A		A	
Windows Mobile	v6.1	C	C	C	C	B	C	B			C		
	v6.5.1	C	C	C	A	A	C	A					
	v7.0	A				A	C	A					
webOS	1.4.1	A										A	
bada	1.0	A											
Maemo	5.0	B				B			C	B			
MeeGo	1.1	A				A				A			

Figura 1 – Matriz de compatibilidade de navegadores móveis

Fonte: [jQuery Mobile, 2011]

A - Possuem uma alta compatibilidade com JQM.

B - São dispositivos que permitem o funcionamento do JQM, mas com alguns bugs.

C - Não são compatíveis com JQM. Neste caso, os elementos serão mostrados na tela como um HTML simples.

Os pontos sem marcação não foram testados.

2.2.2 Características

As características básicas de jQuery Mobile incluem:

- **Compatibilidade:** É compatível com todas as principais plataformas móveis: IOS, Android, Blackberry, Palm WebOS, Nokia / Symbian, Windows Mobile e qualquer outro dispositivo com navegador que interprete HTML;
- **Tamanho leve:** Cerca de 12Kb para a biblioteca JavaScript, 6Kb para o CSS, além de alguns ícones.
- **Integração:** Fácil integração com outras linguagens de programação(PHP, plataforma .NET), pois é feito em JavaScript, atuando somente do lado do cliente.
- **Design Flexível:** A estrutura fornece um sistema de temas, que permite que o desenvolvedor personalize o estilo de layout do aplicativo.
- **Simplicidade:** A estrutura é simples de usar. jQuery Mobile permite desenvolver páginas utilizando principalmente linguagem de marcação com o mínimo ou nenhum conhecimento de JavaScript.
- **Acessibilidade:** jQuery Mobile é projetado baseado em acessibilidade. Ele tem suporte a *WAI-ARIA - Accessible Rich Internet Applications* (Aplicações Internet Ricas e Acessíveis) para ajudar a tornar as páginas acessíveis para os visitantes com necessidades especiais.

2.2.3 Estrutura de uma página jQuery Mobile

De maneira geral a estrutura de uma página deve conter as seguintes seções:

- **Barra do cabeçalho:** Contém o título da página e o botão Voltar;
- **Conteúdo:** Apresenta o conteúdo da aplicação;
- **Barra de rodapé:** Contém elementos de navegação ou qualquer informação que possa ser apresentada no rodapé de uma página.

2.2.4 Transição de Páginas

O JQM oferece suporte para transições de página baseado em CSS, que são aplicados quando se navega para uma nova página e retorna para a página anterior. As opções de transições incluem:

- **Slide:** Transição horizontal na tela;
- **SlideUp e SlideDown:** Transições para cima e para baixo na tela;
- **Pop:** Transição do tipo “explosão”;
- **Desvanecer:** Transição em que a página vai se dissipando, sumindo;
- **Flip:** Transição do tipo “virar de página”.

2.2.5 Temas

O jQuery Mobile oferece suporte a temas que podem ser facilmente combinados. Quase todos os componentes do JQM suportam temas que podem ser facilmente modificados através do atributo *data-theme*. Este sistema tira proveito das propriedades CSS3 para adicionar cantos arredondados, caixas de texto, sombras e gradientes evitando o uso de imagens no layout, o que torna a aplicação mais leve.

Os temas incluem várias cores pré-definidas que podem ser livremente combinadas para criar efeitos visuais mais ricos. Cada tema pode conter até 26 amostras de cores diferentes (rotulado de 'a' a 'z'), que controlam a cor de primeiro plano, cor de fundo e gradiente. O sistema de temas fornece também algumas bibliotecas que permitem a personalização de ícones, fontes e sombras.

2.2.6 Formulários

O JQM personalizou os elementos de formulário padrão, para torná-los visualmente mais atraentes e fáceis de manipular em dispositivos touchscreem. JQM suporta diferentes tipos de elementos de interface com o usuário como barras de ferramentas, botões, listas, tabelas, menus pop-up, caixas de diálogo, transições e elementos de formulário. A Figura 2 apresenta alguns dos componentes suportados.

Smartphone Mobile UI elements

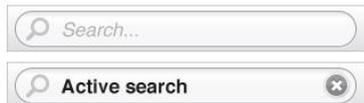
Buttons



Toolbar



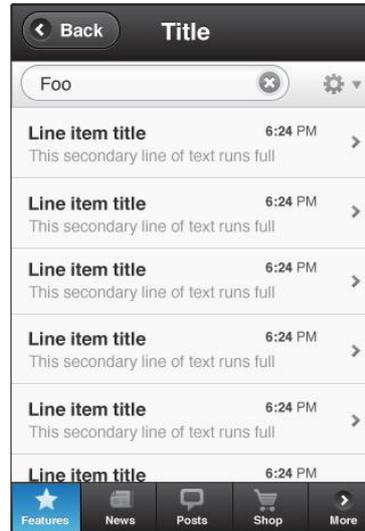
Search bar



Dialog



List view



Popup menu

(4-6 items max, drill page for more)



Figura 2 – Componentes de interface jQuery Mobile

Fonte: [jQuery Mobile, 2011]

2.3 Google Maps API

O Google Maps é um serviço Web, gratuito, de pesquisa e visualização de mapas e imagens de satélite. O serviço oferece mapas de ruas, planejamento de rotas e diversos outros serviços. Pela gratuidade e disponibilidade na internet, torna-se uma ótima ferramenta para a publicação de dados de interesse público.

O Google Maps API é um serviço gratuito, para uso não comercial oferecido pelo Google Maps. A API (*Application Programming Interface*) foi desenvolvida para ser usada em conjunto com um mapa e permite o acesso as funcionalidades do Google Maps através de qualquer página web. Desta forma, os mapas podem ser manipulados através de vários utilitários disponibilizados pela API. Esta API foi escolhida pela grande popularidade e ótima documentação disponível.

Os requisitos para a utilização da API são mínimos. O primeiro passo é obter uma chave (*key*), fornecida ao fazermos a inscrição no serviço, que está disponível na internet: <http://code.google.com/intl/pt-BR/apis/maps/signup.html>. A chave gerada é única para cada usuário de Google Maps e possui alguns termos e condições de uso, como por exemplo, a gratuidade do serviço para os usuários finais e a não permissão de alterar ou ocultar os logotipos do Google no mapa.

Com a chave gerada, o próximo passo é codificar a página que irá receber o mapa. Um requisito para o funcionamento do serviço Google Maps é que o código seja executado em um servidor público.

2.3.1 Google Maps JavaScript API

A API do Google Maps foi desenvolvida inicialmente para visualização de mapas em navegadores da web para desktop, onde a diversidade de tamanhos de monitor e recursos de navegador não se compara a variedade de dispositivos e recursos de aparelhos móveis.

A partir do momento em que houver uma padronização dos dispositivos móveis, com navegadores compatíveis com *JavaScript* baseado em HTML 5, o desenvolvimento de aplicações será facilitado.

A Versão 3 da Google Maps API, oferece uma grande variedade de recursos:

- Qualquer aparelho móvel que suporta *HTML5* e *JavaScript* pode carregar e visualizar uma aplicação com recursos do Google Maps API;
- Por ser uma aplicação web, não é necessário ter conhecimento do código nativo específico de cada dispositivo;
- Quando a aplicação é atualizada, não é necessário que os usuários atualizem seus aplicativos, basta recarregar a página no navegador;
- Quando o Google lança correções de bugs ou adiciona novos recursos, eles são automaticamente adicionados às aplicações que utilizam a API.

2.3.2 Tipos de Mapas

Ao desenvolver aplicações que utilizam a API do Google Maps é possível escolher o tipo de mapa que será exibido aos usuários. Dentre as opções disponíveis pela API, destacam-se:

- **G_NORMAL_MAP**: É para um mapa normal, político, com os nomes das localizações. Quando o tipo de mapa não é especificado na aplicação, este é o padrão definido.
- **G_SATELLITE_MAP**: Este mapa permite ver o mundo com imagens tomadas por satélite.
- **G_HYBRID_MAP**: Este mapa combina a vista satélite e a vista normal.

2.4 HTML5

O HTML5 surgiu da necessidade de encapsulamento dos recursos oferecidos pela internet, centralizando-os em uma única tecnologia, evitando assim a inserção desmedida de plug-ins nos navegadores. Esta é a primeira versão do HTML que fornece ferramentas para o CSS e o *JavaScript* funcionarem da melhor forma possível.

Um dos principais objetivos do HTML5 é facilitar a manipulação dos elementos modificando os objetos de maneira transparente para o usuário final.

Dentre as inovações oferecidas pelo HTML5 destacam-se:

- Redução da necessidade de plug-ins externos (como o Flash);
- Melhor manipulação de erros;
- Independente de dispositivos;
- A inclusão de vídeo e áudio de forma nativa e tratados como objetos HTML, possibilitando uma série de manipulações através da linguagem JavaScript;
- Tags que permitem ao desenvolvedor implementar recursos gráficos sofisticados, sem a necessidade da utilização de plug-ins;
- Recursos de geolocalização, permitindo que os aplicativos determinem sua localização geográfica;
- Inclusão de tags específicas para cada tipo de conteúdo em particular, como áudio e vídeo;
- Utilização de banco de dados local, o que possibilitará a criação de aplicações web que funcionem sem necessidade de estar conectado à internet.
- Compatibilidade com os browsers atuais possibilitando a utilização imediata do novo padrão.

Antes do HTML5 não existia nenhum padrão de construção dos elementos para indicar o que é o cabeçalho e o que é o rodapé de uma página, por exemplo. No HTML5, foi criado um padrão de tags que permite marcar de forma padronizada as principais estruturas de uma página.

O HTML5 modifica a forma como o código é escrito e como a informação é organizada em uma página. Um ponto importante é que nenhum site já existente precisará ser refeito totalmente para se adaptar às novas regras.

2.4.1 Modularização e Compatibilidade

Nas versões antigas do HTML, era necessário aguardar que a plataforma fosse desenvolvida e testada por completo para que pudesse ser utilizada pelos desenvolvedores. O HTML5 foi dividido em módulos, que podem ser utilizados assim que são lançados, sem a necessidade de esperar que o padrão seja completamente disponibilizado. Cada um desses módulos são independentes e podem implementar novas versões a qualquer momento.

A Tabela 2 apresenta uma tabela de compatibilidade entre os navegadores atuais e os principais módulos do HTML5:

	Safari	Chrome	Opera	Firefox	IE 8	IE 9
Local Storage	s	s	s	s	s	s
Histórico de Sessão	s	s	s	s	s	s
Aplicações Offline	s	s	n	s	n	n
Novos tipos de campos	s	s	s	n	n	n
Form: Autofocus	s	s	s	n	n	n
Form: Autocomplete	n	n	s	n	n	n
Form: Required	s	s	s	n	n	n
Video, Audio e Canvas Text	s	s	s	s	n	s

Tabela 2 – Tabela de compatibilidade entre navegadores e o HTML5

Fonte: [HTML5 - Curso W3C Escritório Brasil]

2.4.2 Elementos Novos, Modificados ou Ausentes

O HTML5 incorporou diversos elementos que definem as principais seções de uma página HTML, como o cabeçalho e rodapé. Com isso os buscadores de conteúdo conseguem trabalhar de maneira mais eficaz. Segue a relação de alguns dos principais elementos incluídos no HTML5:

- **article**: Este elemento marca uma seção de código que poderá ser distribuída e reutilizada em FEEDs por exemplo.
- **hgroup**: Agrupa títulos de H1 até H6 quando estes possuem diversos níveis como títulos com subtítulos.
- **header**: É a introdução de um documento.
- **footer**: Representa o rodapé da página. É o último elemento antes de fechar a tag HTML. Pode incluir autor, data, contato, etc.
- **time**: Define um tempo ou data precisa no calendário gregoriano.

Alguns elementos que tinham apenas características visuais foram descontinuados por atrapalhar o código ou porque sua função foi suprida pelo CSS, como o *big*, *center* e *font*, outros elementos tiveram seus significados modificados, como é o caso do elemento *hr* que agora é utilizado para quebrar linhas e fazer separações de parágrafo.

2.4.3 Novos Tipos de Campos

O elemento *input* agora aceita novos tipos de campos para o atributo *type*, dentre eles:

- **search**: Campo do tipo busca.
- **email**: Campo para e-mail, com validação e formatação.
- **url**: Apresenta um endereço da web, com validação e formatação.
- **color**: Exibe um controle de seleção de cor. O valor será uma cor no formato hexadecimal.

2.4.4 Geolocalização com HTML5

Geolocalização é o nome dado à identificação do local geográfico de um usuário ou dispositivo, realizada através da coleta de dados. Este serviço está disponível apenas para dispositivos e navegadores que oferecem suporte a Geolocalização.

Obter a localização de um usuário é bastante útil para aplicações web, principalmente aplicações que irão rodar em dispositivos móveis. A especificação do HTML5 fornece os

dados para geolocalização de uma maneira simples. Uma característica importante desta especificação, é que ela não restringe a origem da informação. Com ela é possível obter a localização de um usuário tanto em um dispositivo móvel, como um GPS ou de uma base dados de endereço IP.

A localização de um usuário é representada por uma coordenada, com latitude e longitude. O resultado da pesquisa é um objeto do tipo `Position` que possui um atributo do tipo `Coordinates` com diversas informações:

- ***latitude***: Posicionamento horizontal no mapa
- ***longitude***: Posicionamento vertical no mapa
- ***accuracy***: precisão em metros das coordenadas de localização
- ***altitude***: altitude em metros em relação ao nível do mar
- ***altitudeAccuracy***: precisão em metros da altitude
- ***heading***: direção da viagem, definida em graus (0°, 360°), contando no sentido horário em relação ao norte geográfico
- ***speed***: velocidade de terra do usuário em metros/segundo.

3. DESENVOLVIMENTO

Neste capítulo será apresentado o desenvolvimento de um Sistema Mobile/Web para Busca Georreferenciada de Imóveis.

A aplicação apresenta como principais características:

- Facilidade e simplicidade de utilização;
- Interface que se ajusta tanto para a visualização em computadores desktops como em dispositivos móveis;
- Busca de imóveis localizados próximos ao endereço informado pelo usuário;

3.1 Especificação de Requisitos

3.1.1 Requisitos Funcionais

- Executar em qualquer computador ou dispositivo móvel que possua navegador compatível;
- Realizar a localização geográfica do usuário;
- Realizar a localização geográfica do bairro selecionado;
- Exibir todos os imóveis que atendam aos filtros informados;
- Exibir informações detalhadas do imóvel selecionado.

3.1.2 Requisitos Não Funcionais

- Por se tratar de uma aplicação que poderá ser acessada via dispositivos móveis, a usabilidade deverá ser simples e eficiente, com uma interface limpa e acessível e com desempenho aceitável.

3.1.3 Restrições

- Ambiente: Ter acesso à internet e um navegador compatível com HTML5 (Tabela 2)
- Segurança: O usuário deverá permitir através de uma caixa de diálogo, que o sistema acesse as informações da sua localização.

3.2 Modelagem do Sistema

As funcionalidades oferecidas pela aplicação serão apresentadas através da linguagem de modelagem UML (*Unified Modeling Language*). A UML é uma linguagem que fornece diagramas padronizados que ajudam a descrever e projetar sistemas de software, especialmente aplicações orientadas a objetos. O objetivo da UML é especificar, documentar e estruturar o desenvolvimento completo de um sistema de informação.

3.2.1 Diagrama de Casos de Uso

O diagrama de caso de uso será usado para demonstrar graficamente, em alto nível, como os usuários interagem como o sistema. A Figura 3 apresenta o diagrama de casos de uso da aplicação.

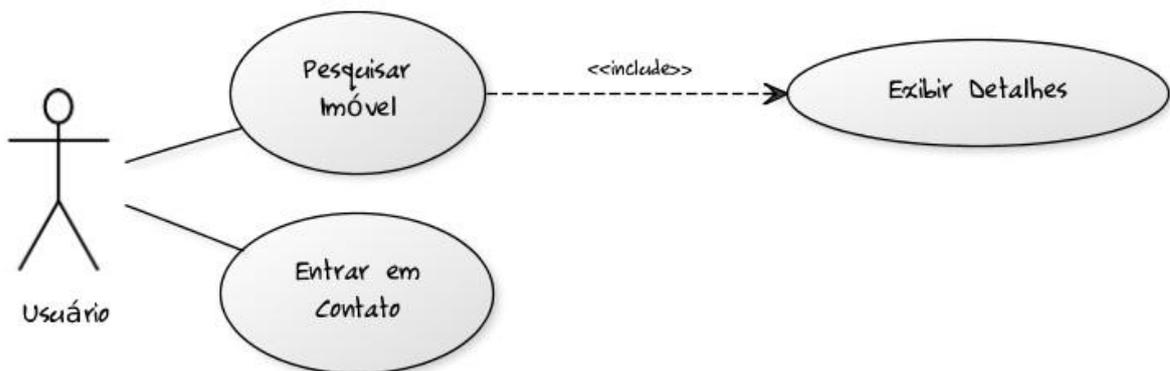


Figura 3 – Diagrama de Caso de Uso

3.2.2 Diagrama de Classes

O diagrama de classes representa a estrutura e relações das classes que servem de modelo para os objetos. É ele que define todas as classes que o sistema necessita. A Figura 4 apresenta o diagrama de classes da aplicação.

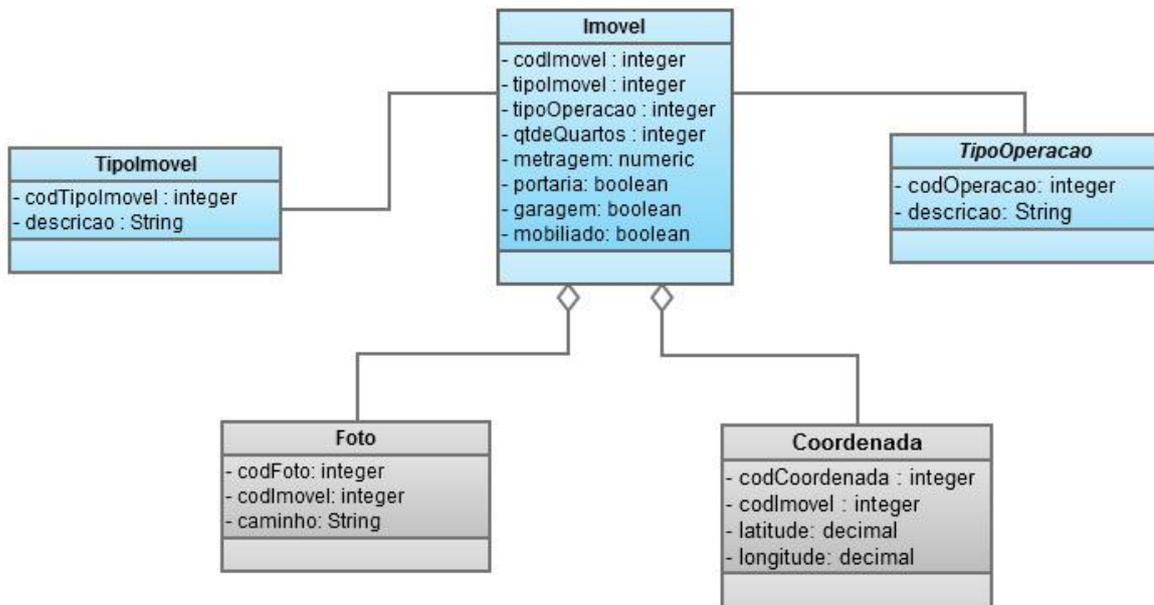


Figura 4 – Diagrama de Classes

3.2.3 Diagrama de Sequência

O diagrama de sequência representa a sequência de processos na aplicação. Ele descreve a maneira como os grupos de objetos colaboram em algum comportamento ao longo do tempo.

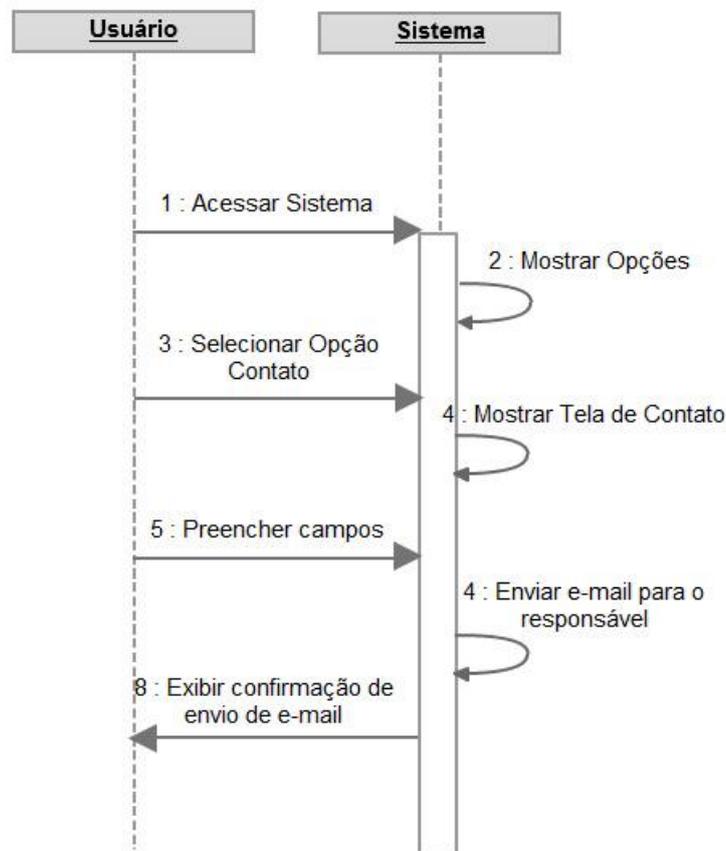


Figura 5 – Diagrama de Sequência – Caso de Uso “Entrar em Contato”

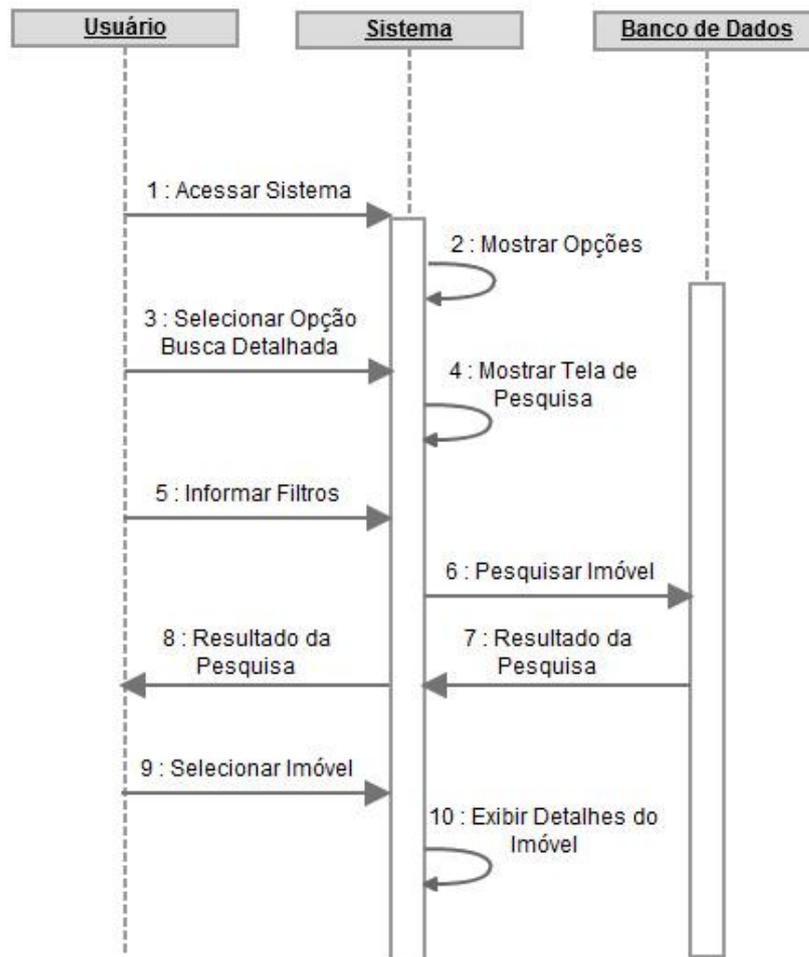


Figura 6 – Diagrama de Sequencia – Caso de Uso “Pesquisar Imóvel”

3.2.4 Descrição de Casos de Uso

O diagrama de caso de uso apresentado na Figura 3 é eficaz para caracterizar de forma gráfica as funcionalidades oferecidas pela aplicação. Porém não ficam claros quais os passos necessários para que a interação do usuário com o sistema seja eficiente. Neste caso serão apresentadas a seguir as descrições dos casos de uso, que explicam passo a passo as interações entre o usuário e os casos de uso da aplicação.

Caso de Uso 01 – Pesquisar Imóvel	
Atores:	Usuário
Pré-condição:	Nenhuma
Pós-condição:	Imóveis que atendem aos filtros de pesquisa são

	exibidos no mapa.
Descrição:	Este caso de uso permite que o usuário selecione filtros de pesquisa para o imóvel desejado. Em seguida o sistema realiza a localização geográfica do usuário e exibe os imóveis disponíveis naquele local.
Sequência de Eventos	
Ator	Sistema
1. Este caso de uso inicia quando o ator seleciona na tela inicial a opção “Busca Detalhada”.	
	2. Apresenta uma tela com diversas opções de filtro.
3. Preenche os filtros de pesquisa, indicando suas preferências para o imóvel desejado. Após definidas as preferências, o usuário clica em “Buscar”.	
	4. Faz a localização geográfica do ator e exibe um mapa com os imóveis encontrados naquela região, baseado nos filtros informados pelo usuário.
Fluxo Alternativo 01 – Exibir Imóvel	
1. Seleciona um dos imóveis indicados no mapa.	
	2. Exibe um box com as informações detalhadas sobre o imóvel selecionado.

Tabela 3 – Descrição do caso de uso “Pesquisar Imóvel”

Caso de Uso 02 – Entrar em Contato	
Atores:	Usuário
Pré-condição:	Nenhuma
Pós-condição:	Um e-mail é enviado ao responsável pelo sistema.
Descrição:	Este caso de uso permite que o usuário entre em contato com o responsável pelo sistema, através de um formulário.
Sequência de Eventos	
Ator	Sistema
1. Este caso de uso inicia quando o ator seleciona na tela inicial a opção “Contato”.	

	2. Apresenta uma tela com um formulário de contato.
3. Preenche os campos apresentados na tela e clica em “Enviar”.	
	4. Envia um e-mail para o responsável pelo sistema. O conteúdo do e-mail é formatado de acordo com os dados informados pelo usuário.

Tabela 4 – Descrição do caso de uso “Entrar em Contato”

3.3 Implementação

Para a implementação do sistema foi utilizada a IDE Eclipse 3.6, o servidor web Apache Tomcat 6.0, as linguagens de programação Java e *JavaScript* e o gerenciador de banco de dados PostgreSQL. O Struts foi utilizado como framework de apoio ao desenvolvimento e o Hibernate para realizar a persistência de dados.

3.3.1 Instalação do jQuery Mobile

Assim como outros frameworks JavaScript para desenvolvimento web, para utilizar as funcionalidades do jQuery Mobile, basta indicar na página HTML os arquivos necessários para o framework funcionar. É possível apontar para os arquivos css/js diretamente no repositório jQuery ou baixar a última versão e incluir os arquivos dentro do projeto. Os arquivos para download são encontrados no site oficial do framework: <http://jquerymobile.com/>

```

<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
    <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0b1/jquery.mobile-1.0b1.min.css" />
    <script src="http://code.jquery.com/jquery-1.6.1.min.js"></script>
    <script src="http://code.jquery.com/mobile/1.0b1/jquery.mobile-1.0b1.min.js"></script>
  </head>
  <body>

    [...]

  </body>
</html>

```

Figura 7 – Utilização do JQM em uma página HTML

3.3.2 Design das telas com HTML5

Para a criação das telas, foi utilizado o padrão de tags e elementos HTML5. As figuras a seguir apresentam trechos de código HTML das telas da aplicação.

```
<div data-role="page" data-theme="b">
  <div data-role="header">
    <h1>Cadê Meu Apê?</h1>
  </div>

  <div data-role="content">
    <ul data-role="listview" data-inset="true">
      <li data-role="list-divider">MENU</li>
      <li><a href="#buscaDetalhada">Busca Detalhada</a></li>
      <li><a href="#contato">Contato</a></li>
    </ul>
  </div>
</div>
```

Figura 8 – Trecho de código HTML da página inicial

```
<div data-role="fieldcontain">
  <label for="mobiado">Mobiado:</label>
  <select name="mobiado" id="mobiado" data-role="slider">
    <option value="off">Off</option>
    <option value="on">On</option>
  </select>
</div>

<div data-role="fieldcontain">
  <label for="slider">Metragem Mínima:</label>
  <input type="range" name="slider" id="slider" value="0" min="50" max="150"/>
</div>

<div data-role="fieldcontain">
  <label for="slider">Quartos:</label>
  <input type="range" name="slider" id="slider" value="0" min="1" max="5"/>
</div>

<div data-role="fieldcontain">
  <label for="foto">Com Foto:</label>
  <select name="foto" id="foto" data-role="slider">
    <option value="off">Off</option>
    <option value="on">On</option>
  </select>
</div>

<div data-role="fieldcontain">
  <input type="submit" value="Buscar" data-inline="true"/>
</div>
```

Figura 9 – Trecho de código HTML da página de busca

3.3.3 Obtendo a localização do usuário

Para que a aplicação realize a localização do usuário, é necessário incluir na página JSP uma chamada a biblioteca da API Google Maps para que ela realize a geolocalização. Este código funcionou com sucesso nos navegadores Opera, Firefox, Chrome e Safari. O navegador Internet Explorer não teve sucesso no teste.

Quando o código é executado, o navegador solicita que o usuário permita que sua localização seja compartilhada. A chamada a biblioteca da API Google Maps é feita através da URL <http://maps.google.com/maps/api/js?sensor=true>

3.4 Apresentação do Sistema Desenvolvido

Nas telas do sistema desenvolvido, apresentadas a seguir, é possível ver como a aplicação se comporta quando carregada em um navegador desktop e na sequência carregada em um navegador de dispositivo móvel.

3.4.1 Tela Inicial

A tela inicial do sistema apresenta as opções “Busca Detalhada” e “Contato”.

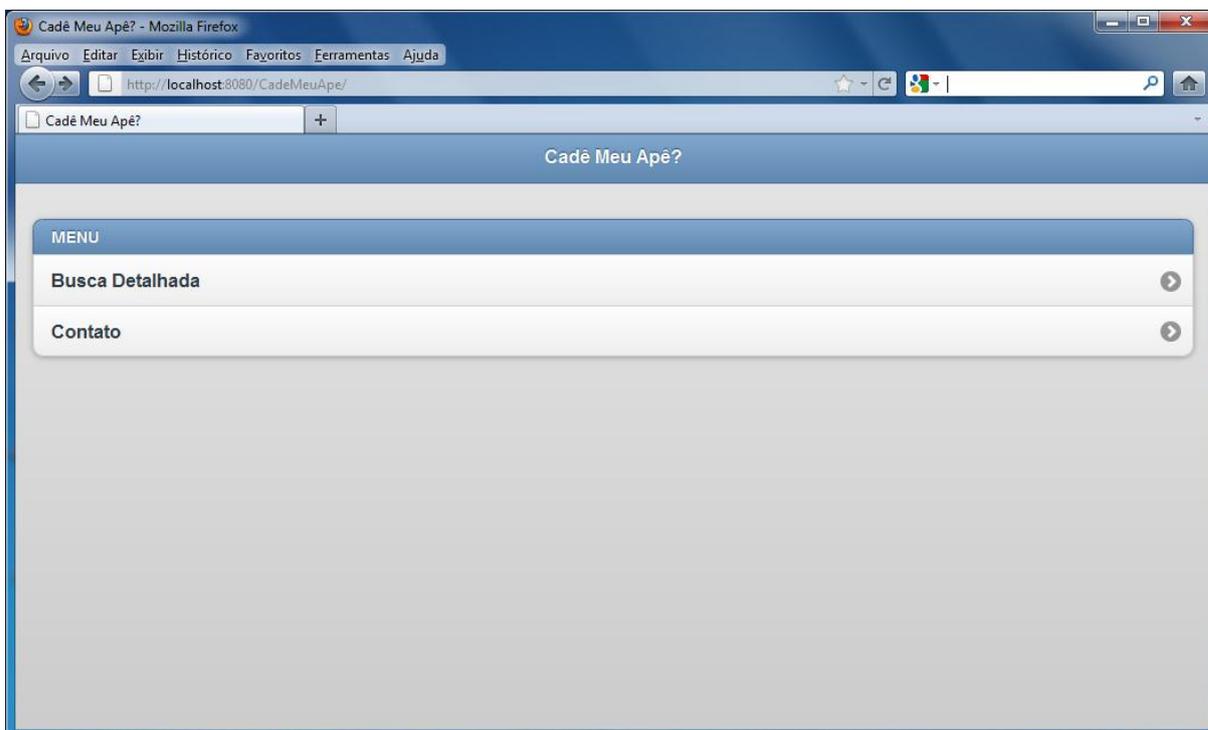


Figura 10 – Tela inicial do sistema (navegador desktop Mozilla Firefox 6.x).



Figura 11 – Tela inicial do sistema (navegador *mobile*).

3.4.2 Tela de Contato

Na tela de contato o usuário deve preencher os campos e acionar o botão “Enviar”. O sistema envia um e-mail para o responsável pelo sistema com os dados preenchidos pelo usuário.

A desktop contact form titled "Contato". It features four input fields: "Nome:", "E-mail:", "Assunto:", and "Mensagem:". The "Mensagem:" field is a larger text area with a small grid icon in the bottom right corner. A rounded "Enviar" button is located at the bottom left of the form.

Figura 12 – Tela de contato (navegador desktop Mozilla Firefox 6.x).



Figura 13 – Tela de contato (navegador *mobile*).

3.4.3 Tela de Pesquisa de Imóveis

A tela de busca detalhada exibe os filtros que devem ser preenchidos para a realização da pesquisa. Dentre as opções o usuário pode escolher se deseja utilizar sua localização atual ou outra região da cidade.

Cadê Meu Apê?

Operação: Comprar Alugar

Tipo do Imóvel: Apartamento

Garagem: Não

Mobiliado: Não

Metragem Mínima: 50

Quartos: 1

Com Foto: Não

Utilizar localização atual: Não

Região: Centro

Buscar

© 2011 :: Gláucia M. Pasto Mendes :: Especialização em Tecnologia JAVA

Figura 14 – Tela de pesquisa (navegador desktop Mozilla Firefox 6.x).



Figura 15 – Tela de pesquisa (navegador *mobile*).

3.4.4 Tela de Resultado da Pesquisa

Ao submeter os filtros de pesquisa o sistema exibe o resultado, apresentando um mapa indicando os imóveis disponíveis. Quando o usuário seleciona um ponto no mapa, um *box* é exibido contendo detalhes sobre o imóvel.

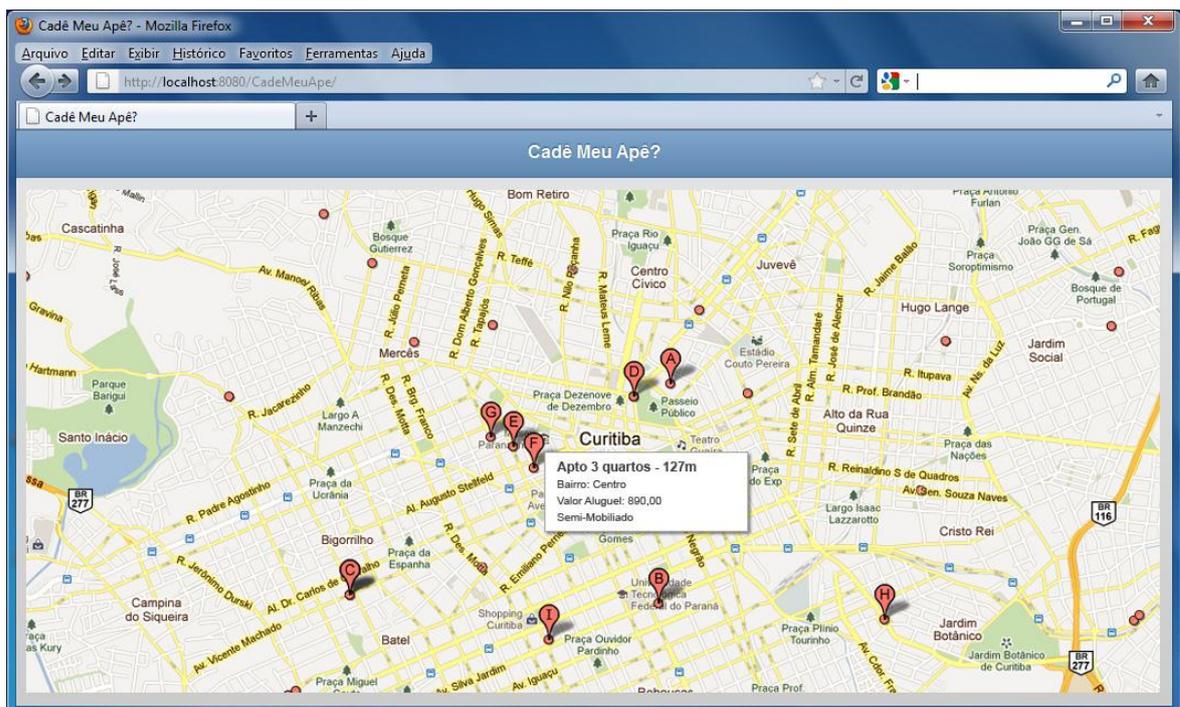


Figura 16 – Tela de resultado (navegador desktop Mozilla Firefox 6.x).



Figura 17 – Tela de resultado (navegador *mobile*).

4. CONCLUSÃO

O foco deste trabalho abordou um estudo sobre as tecnologias envolvidas na construção de aplicações *mobile web*, ou seja, uma mesma aplicação que pode ser utilizadas tanto em navegadores desktop quanto em navegadores *mobile*, sem a necessidade de adaptações de código.

A utilização do framework jQuery Mobile, aliada a API do Google Maps permitiu a criação de uma aplicação simples e funcional, de maneira que o seu emprego pode ser realizado na construção de aplicações cuja característica principal é a mobilidade e facilidade de uso.

O protótipo desenvolvido cumpriu as exigências dos testes realizados e comprovou a viabilidade da utilização das ferramentas e tecnologias propostas. Dentre as questões observadas é importante ressaltar que as ferramentas utilizadas são licenciadas como software livre, o que pode significar uma opção atrativa de custo/benefício para as empresas interessadas neste tipo de solução.

A realização deste trabalho permitiu a consolidação do conhecimento adquirido durante o curso de Especialização em Tecnologia Java e permitiu a aplicação e exploração de várias disciplinas apresentadas no curso.

4.1 Projetos Futuros

Como complemento deste trabalho pode-se considerar a realização de melhorias na interface com o usuário:

- Cadastro e autenticação do usuário para que o sistema possa tratar o usuário de maneira personalizada;
- Opção para salvar os imóveis em uma seção de “Favoritos”, para que seja visualizado com facilidade no futuro;
- Opção para que o sistema realize uma busca automática de imóveis em locais pré-definidos pelo usuário em intervalos de tempo também pré-definidos;
- Envio de alerta por e-mail quando um novo imóvel que atenda às preferências do usuário for inserido no sistema.

- Possibilidade de busca diretamente sobre o mapa. O usuário clica em um ponto no mapa e o sistema exibe os imóveis disponíveis em um raio pré-determinado.
- Cadastro de “locais preferenciais” do usuário, para que a busca seja realizada em outros locais além do local atual do usuário.

REFERÊNCIAS

BATISTA, Diogo C. T. **O Impacto do HTML5 no Desenvolvimento para a Internet.**

Disponível em: http://www.diogocezar.com/files/html5/artigo_html5.pdf

Acesso em: 05/06/2011, 22h05min.

BOOCH, Grady; RUMBAUGH, James e JACOBSON, Ivar, 2005: **UML, Guia do Usuário.** Campus 2ª Ed.

DEITEL, Harvey M.; DEITEL, Paul J., 2005: **Java: Como Programar.** Prentice-Hall 6ª Ed.

Desenvolvimento com o API de Google Maps. Disponível em:

<http://www.criarweb.com/desenvolvimento-google-maps/>

Acesso em: 29/05/2011, 11h26min.

Desenvolvimento Mobile Web. Disponível em:

<http://www.jera.com.br/artigos/o-que-e-desenvolvimento-mobile-web/>

Acesso em: 29/05/2011, 15h12min.

Divisão de Processamento de Imagens (DPI). Disponível em:

<http://www.dpi.inpe.br>

Acesso em: 22/05/2011, 23h01min.

FREDERICK Gail R.; LAL, Rajesh, 2011: **Dominando o Desenvolvimento Web para Smartphone.** Alta Books 1ª Ed.

Google Maps JavaScript API V3. Disponível em:

<http://code.google.com/intl/ptBR/apis/maps/documentation/javascript/>

Acesso em: 10/05/2011, 17h07min.

HTML5 - Curso W3C Escritório Brasil. Disponível em:

<http://www.w3c.br/pub/Cursos/CursoHTML5/html5-web.pdf>

Acesso em: 22/08/2011, 22h10min.

HTML5 - Um guia de referência para os desenvolvedores web. Disponível em:

<http://tableless.com.br/html5/>

Acesso em: 05/06/2011, 22h05min.

jQuery Mobile, 2011. Disponível em:

<http://jquerymobile.com/>

Acesso em: 04/09/2011, 21h05min.

LEE, Valentino; SCHNEIDER, Heather e Robbie, 2005: **Aplicações Móveis – Arquitetura, Projeto e Desenvolvimento**. Pearson 1ª Ed.

Revista Eletrônica PEGN, 2010. Disponível em:

<http://revistapegn.globo.com/Revista/Common/0,,EMII61092-17141,00.html>

Acesso em: 10/07/2011, 15h05min.

SILVA, Maurício Samy, 2011: **Html5 - A Linguagem da Marcação Que Revolucionou a Web**. Novatec 1ª Ed.

W3C Geolocation API. Disponível em:

<http://www.w3.org/TR/geolocation-API/>

Acesso em: 03/09/2011, 22h15min.