

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA SUL-RIO-
GRANDENSE - IFSUL, CÂMPUS PASSO FUNDO
CURSO DE TECNOLOGIA EM SISTEMAS PARA INTERNET**

ANDRESSA FERREIRA

**DESENVOLVIMENTO DE UM SISTEMA DE LOCALIZAÇÃO E
AVALIAÇÃO DE RESTAURANTES**

Prof. Me. Adilso Nunes de Souza

PASSO FUNDO, 2015

ANDRESSA FERREIRA

**DESENVOLVIMENTO DE UM SISTEMA DE LOCALIZAÇÃO E
AVALIAÇÃO DE RESTAURANTES**

Monografia apresentada ao Curso de Tecnologia em Sistemas para Internet do Instituto Federal Sul-Rio-Grandense, Câmpus Passo Fundo, como requisito parcial para a obtenção do título de Tecnólogo em Sistemas para Internet.

Orientador (a): Prof. Me. Adilso Nunes de Souza

PASSO FUNDO, 2015

ANDRESSA FERREIRA

DESENVOLVIMENTO DE UM SISTEMA DE LOCALIZAÇÃO E AVALIAÇÃO DE RESTAURANTES

Trabalho de Conclusão de Curso aprovado em ____/____/____ como requisito parcial para a obtenção do título de Tecnólogo em Sistemas para Internet

Banca Examinadora:

Prof. Me. Adilso Nunes de Souza
Orientador

Prof. Dr. Josué Toebe
Avaliador

Prof. Me. Roberto Wiest
Avaliador

Prof. Me. Adilso Nunes de Souza
Coordenador do Curso

PASSO FUNDO, 2015

Dedico primeiramente a Deus, pela força e coragem durante toda esta longa caminhada. À minha família, por acreditar e investir em mim. Aos amigos e colegas pelo incentivo e apoio constantes.

AGRADECIMENTOS

Agradeço a Deus, pela vida e por ter me dado saúde e coragem para realizar essa tarefa.

Ao meu pai Gil Ferreira, pelo amor e dedicação que sempre teve comigo, com seus exemplos de honestidade, perseverança, fé, compreensão, que me deram forças para seguir em frente, a ele meu eterno agradecimento. Á minha mãe Nelsinda Vieira Ferreira (In memoriam) que me ensinou desde criança o real sentido do estudo e dedicação, com seu amor, companheirismo e honestidade. Ao meu irmão e minhas irmãs que sempre me apoiaram e incentivaram em todo momento que precisei.

Ao meu orientador Prof. Me. Adilso Nunes de Souza, pelo suporte, paciência, dedicação, incentivo que foram fundamentais para a construção desse projeto. Aos professores do Curso Superior de Tecnologia em Sistemas para Internet do Instituto Federal Sul-Rio-Grandense Campus Passo Fundo, pelo apoio, dedicação e incentivo durante o decorrer do curso.

Por fim agradeço aos amigos, colegas e demais familiares, pela compreensão durante os momentos que mais necessitei e a todos que de alguma forma participaram direta ou indiretamente para a realização desse trabalho.

“Que os vossos esforços desafiem as impossibilidades,
lembrai-vos de que as grandes coisas do homem foram
conquistadas do que parecia impossível”.

Charles Chaplin

RESUMO

O presente trabalho apresenta um estudo sobre as tecnologias de geolocalização e geração de gráficos do Google e sua aplicação em um Sistema de Localização e Avaliação de restaurantes. A motivação do estudo foi baseado no aumento da procura da população das grandes cidades por realizar refeições fora de casa, e como forma de auxílio, agilizar o contato do cliente com o restaurante e seus colaboradores de forma prática através da avaliação dos estabelecimentos. Durante estudos, constatou-se que a tecnologia da API Google Earth traz recursos interessantes para geolocalização, e em conjunto com a linguagem PHP é possível criar um sistema consistente e simples para a localização dos restaurantes. O mesmo acontece com a tecnologia da API Google Chart para a criação de gráficos que possui vários recursos para implementação e utilização na avaliação dos restaurantes. O objetivo desse trabalho foi aplicar as tecnologias das API Google Earth e Google Chart para o desenvolvimento de um sistema de localização e avaliação de restaurantes para a cidade de Passo Fundo. Por fim constatou-se que estas tecnologias utilizadas possuem praticidade e rapidez na sua aplicação diminuindo a reescrita de código.

Palavras-chave: Geolocalização; Avaliação; API Google Earth; API Google Chart.

ABSTRACT

This paper presents a study on the geolocation technologies and generation of Google charts and their application in a Location System and Evaluation restaurants. The motivation of the study was based on the increasing demand of the population of large cities for carrying meals away from home, and as a form of assistance, streamline customer contact with the restaurant and its employees in a practical manner by evaluating the establishments. During studies, it was found that the API of Google Earth technology brings interesting features for geolocation, and in conjunction with PHP you can create a consistent and simple system for the location of the restaurants. The same goes for the Google Chart API technology to create graphics that has several resources for implementation and use in the evaluation of restaurants. The aim of this study was to apply the technology of Google Earth and Google Chart API to develop a tracking system and evaluation of restaurants to the city of Passo Fundo. Finally, it was found that these technologies have used convenience and speed in applying decreasing code duplication.

Key words: Geolocation; Evaluation; Google Earth API; Google Chart API

LISTA DE TABELAS

Tabela 1 – Caso de Uso Visualizar Mapa	71
Tabela 2 – Caso de Uso Pesquisar por Restaurante.....	71
Tabela 3 – Caso de Uso Avaliar Restaurantes.....	72
Tabela 4 – Caso de Uso Cadastro de Restaurantes.....	74
Tabela 5 – Caso de Uso Cadastro de Proprietários	75
Tabela 6 - Caso de Uso de Cadastrar Especialidades	76
Tabela 7 – Caso de Uso Cadastro de Itens Avaliados	76
Tabela 8 – Caso de Uso Cadastrar Usuários	77
Tabela 9 - Caso de Uso de Cadastrar Promoções	78
Tabela 10 - Caso de Uso de Cadastrar Cardápios Restaurantes	78
Tabela 11 - Caso de Uso Dias da Semana	79
Tabela 12 - Caso de Uso Cadastrar Cardápio do Dia	80

LISTA DE FIGURAS

Figura 1 – População que faz refeições fora de casa.....	13
Figura 2 – Aplicativo Kekanto	17
Figura 3 – Conta do Usuário e Visualização de estabelecimentos	18
Figura 4 – Avaliação.....	18
Figura 5 – Exemplo de marcadores	22
Figura 6 – Exemplo de criação de Marcadores Simples	22
Figura 7 – Carregando Plugin.....	23
Figura 8 – Função para gerar o mapa	24
Figura 9 – Marcadores no Mapa.....	25
Figura 10 – Visualizando Informações no Mapa	25
Figura 11 – Resultado do restaurante por item avaliado	27
Figura 12 – Importando plug-in da API	28
Figura 13 – Exemplo Código Função que Gera Gráficos	28
Figura 14 – Gráfico Pizza 3D. Resultado do restaurante por item avaliado	29
Figura 15 – Gráfico em Colunas. Resultado geral do Restaurante.....	30
Figura 16 – Gráfico Combo.....	30
Figura 17 – Utilização do PHP	36
Figura 18 – Utilização de Banco de Dados por Empresas Brasileiras	37
Figura 19 – Visualização em Smartphones	39
Figura 20 – Fluxograma de Desenvolvimento	40
Figura 21 – Caso de Uso Usuário Cliente	45
Figura 22 – Caso de Uso Ações do Administrador	45
Figura 23 – Caso de Uso Cadastros.....	46
Figura 24 - Caso de Uso Usuário Restaurante.....	46
Figura 25 – Diagrama de Classes	47
Figura 26 – Acesso Usuário Cliente	48
Figura 27 – Acesso Usuário Restaurante.....	49
Figura 28 – Campos Editáveis.....	50
Figura 29 – Campo de CPF	50
Figura 30 – Campos de Senha	51
Figura 31 - Campo de data	51
Figura 32 – Campos de seleção	51
Figura 33 – Campo de Telefone	52

Figura 34 – Validação de CPF.....	52
Figura 35 – Validação de E-mail.....	52
Figura 36 – Listagem de Dados.....	53
Figura 37 – Confirmação de Inclusão.....	53
Figura 38 – Confirmação de Alteração.....	54
Figura 39 – Pergunta Exclusão de Registro.....	54
Figura 40 – Confirmação de Exclusão	54
Figura 41 – Registro já cadastrado	55
Figura 42 – Solicitação não Realizada	55
Figura 43 – Mensagem Para Fazer Login.....	55
Figura 44 – Acesso Negado.....	56
Figura 45 – Consulta de Dados	56
Figura 46 – Botões do Sistema.....	56
Figura 47 – Paginação	57
Figura 48 – Tela Inicial	57
Figura 49 – Pesquisa por Restaurantes	58
Figura 50 – Tela Login	58
Figura 51 – Cadastro de Usuários Cliente.....	59
Figura 52 – Administração dos Proprietários	59
Figura 53 – Formulário de Avaliação.....	60
Figura 54 – Resultado Geral do Restaurante.....	61
Figura 55 – Resultado por Item	61
Figura 56 – Resultado por Item e idade.....	62

LISTA DE ABREVIATURAS E SIGLAS

3D – Três Dimensões

ASP – *Active Server Pages* (Páginas Ativas de Servidor)

API – *Application Programing Interfaces* (Interface de Programação de Aplicativos)

CSS – *Cascading Style Sheets* (Folha de Estilo em Cascata)

FI – *Forms Interpreter*

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

HTTP – *HyperText Transfer Protocol* (Protocolo de Transferência de Hipertexto)

JSP – *Java Server Pages* (Páginas de Servidor Java)

KML – *Keyhole Markup Language* (Linguagem de Marcação Keyhole)

OO – Orientação a Objetos

PHP – *Hypertext Preprocessor* (Pré-Processador de Hipertexto)

SGBDR – Sistema Gerenciador de Base de Dados Relacional

SQL – *Structured Query Language* (Linguagem de Consulta Estruturada)

SVG – *Scalable Vector Graphics* (Gráficos Vetoriais Escaláveis)

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

VML – *Vector Markup Language* (linguagem de marcação vetorial)

XML – *eXtensible Markup Language* (Linguagem de Marcação Extensível)

SUMÁRIO

1	INTRODUÇÃO	12
1.1	MOTIVAÇÃO	13
1.2	OBJETIVOS	14
1.2.1	Objetivo Geral	14
1.2.2	Objetivos Específicos	14
1.3	Estrutura do Trabalho.....	15
2	REFERENCIAL TEÓRICO	16
2.1	SISTEMAS EXISTENTES	16
2.2	UTILIZANDO MAPAS	20
2.2.1	Exibição de dados geográficos	21
2.3	UTILIZANDO GRÁFICOS	26
2.3.1	Exemplo de gráficos	26
2.4	MODELAGEM DO SISTEMA.....	31
2.5	LINGUAGEM DE PROGRAMAÇÃO PHP.....	32
2.6	BANCO DE DADOS.....	36
2.7	LAYOUT RESPONSIVO – BOOTSTRAP	37
3.	METODOLOGIA	39
4.	RESULTADOS OBTIDOS	42
4.1	REQUISITOS FUNCIONAIS.....	43
4.2	REQUISITOS NÃO FUNCIONAIS	44
4.3	DIAGRAMAS DE CASO DE USO.....	44
4.4	Diagrama de Classes.....	47
4.5	Arquitetura do Sistema	48
4.6	PADRÕES DO SISTEMA.....	48
4.7	Detalhes do Sistema	57
5.	CONSIDERAÇÕES FINAIS	63
	REFERÊNCIAS	65
	ANEXOS	69

1 INTRODUÇÃO

A quantidade de atividades rotineiras da vida moderna faz com que as pessoas procurem, cada vez mais, comodidade, e uma forma de agilizar as tarefas a serem enfrentadas. Muitas não encontram tempo para fazer uma refeição adequada, ou até mesmo adiam essa refeição. Porém a maioria da população procura ganhar tempo, dirigindo-se a restaurantes ou casa de lanches.

Com isso, frequentemente as pessoas realizam suas refeições longe de suas residências, por conhecer a comida servida, o ambiente de seu agrado, o bom atendimento, entre outros fatores que fazem a diferença no momento da escolha de um restaurante. Sendo assim, a procura por restaurantes aumenta e por consequência o número de clientes também. Uma pesquisa realizada pela Abia (Associação Brasileira de Indústrias de Alimentação), informou que o segmento de *food service* vem crescendo 14,7% ao ano. Em 2013, 32,9% do consumo alimentício da população brasileira foi feito fora de casa. No ano de 1995 o consumo alimentício era de 19%, o que demonstra um crescimento sólido para o desenvolvimento deste mercado (BRASIL, 2015).

O crescimento da procura por lanchonetes e restaurantes agrada os proprietários desses estabelecimentos, porém eles não contavam com o aumento de clientes exigentes, pois ao realizar um questionário a alguns restaurantes da cidade de Passo Fundo, 100% dos entrevistados afirmaram que acreditam que os clientes estão mais exigentes. Dessa forma, estes clientes procuram não somente por boa comida, mas principalmente por locais que prezam pela qualidade de produtos e serviços oferecidos.

A fim de possibilitar mais rapidez e comodidade para localizar o estabelecimento desejado, foi desenvolvido um sistema localizador de restaurantes. Nesse sistema, o usuário pode encontrar os restaurantes referenciados no mapa, obtendo assim a possibilidade de escolher locais próximos ou distantes de sua casa, trabalho, estudo. Para que o usuário possa encontrar um local de forma mais específica, foi disponibilizado uma busca por especialidade/tipo, de maneira que o cliente possa encontrar o tipo de restaurante que deseja como churrascarias, pizzarias, galeterias, entre outros.

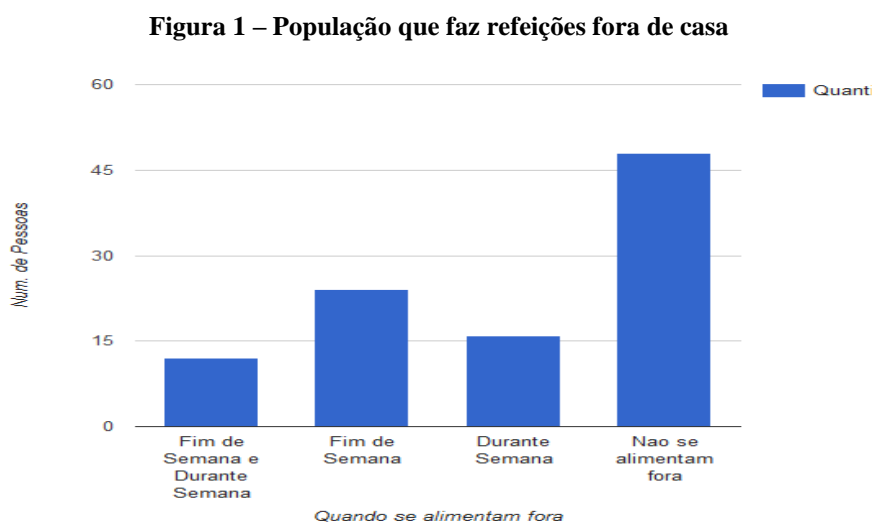
Com o intuito de simplificar o contato entre restaurantes e clientes, o sistema apresenta por meio de gráficos os resultados das avaliações dos frequentadores, facilitando a visualização e o entendimento dos resultados, como também para que os restaurantes

procurem realizar as melhorias necessárias em seus estabelecimentos. Pois o grande diferencial do sistema é o acesso disponibilizado aos proprietários, que estarão mais próximos das opiniões de seus clientes.

Para disponibilizar essas funcionalidades no sistema, decidiu-se pela utilização da tecnologia API (*Application Programming Interface*), essa interface de programação de aplicativos é um conjunto de padrões de programação, que pode ser utilizada em um sistema de forma muito evidente para seus usuários e ao mesmo tempo conectada a diversos outros sistemas e aplicativos sem que o usuário perceba. Sendo assim utilizando a Google Chart API e Google Earth API o desenvolvedor consegue uma comunicação direta com as funções pré-definidas pelo Google.

1.1 MOTIVAÇÃO

Uma das motivações para realização desse trabalho foi que segundo pesquisa realizada no ano de 2010 pela GfK¹, fazer refeições fora de casa já é algo comum para 51% da população brasileira. Conforme levantamento 12% da população fazem refeições fora de casa tanto nos finais de semana quanto nos dias úteis, 24% se alimenta fora apenas no fim de semana e 16% durante os demais dias (OLHAR DIRETO, 2014). Considerando essas estatísticas percebe-se que há uma necessidade de facilitar a localização de restaurantes, casas de lanches e demais estabelecimentos do mesmo ramo. Através desses dados foi possível gerar o gráfico demonstrado na Figura 1.



Fonte: Do Autor

¹ É uma empresa de pesquisa de mercado, que procuram descobrir novos *insights* sobre o modo como as pessoas vivem, pensam e compram. <http://www.gfk.com/br/about-us/Paginas/default.aspx>

Ainda na mesma pesquisa realizada pela GfK, o almoço é a principal refeição realizada fora de casa (71%). Locais como lanchonetes são mais frequentados por jovens entre 18 e 24 anos, enquanto que entre as faixas etárias acima de 35 anos a frequência é mais baixa. Dessa forma percebe-se que a procura por estabelecimentos para fazer uma refeição é comum para metade da população brasileira e que aplicativos/sistemas com funcionalidades como as que foram desenvolvidas nesse projeto, possuem um número comprovadamente bom de possíveis usuários.

A segunda motivação do trabalho desenvolvido foi através da análise de sistemas e aplicativos existentes que possuem objetivos similares, porém muitos desses aplicativos possuem um caráter social, ou seja, são sistemas/aplicativos que procuram entreter o usuário, interagindo com as redes sociais, não possuindo um caráter qualitativo. Diante desta constatação o aplicativo desenvolvido buscou, com as suas funcionalidades, facilitar o contato com o cliente e o restaurante através de um espaço para avaliação do restaurante, disponibilizando os resultados dessas avaliações aos responsáveis dos estabelecimentos e seus clientes, permitindo a análise da opinião dos clientes sobre determinado serviço ou produto oferecido.

1.2 OBJETIVOS

A seguir encontram-se os objetivos do sistema, separados em: Objetivo Geral que demonstra o principal motivo do desenvolvimento do sistema e Objetivos Específicos que apresenta em detalhes os objetivos propostos do sistema desenvolvido.

1.2.1 Objetivo Geral

O objetivo desse sistema é aplicar a tecnologia das APIs Google Chart e Google Earth no desenvolvimento de um sistema de localização e avaliação de restaurantes para a cidade de Passo Fundo.

1.2.2 Objetivos Específicos

- Conhecer a API Google Earth;
- Conhecer a API Google Chart;
- Projetar e desenvolver um protótipo do sistema de localização e avaliação de

restaurantes para a cidade de Passo Fundo;

- Geolocalizar os restaurantes no mapa utilizando a API Google Earth;
- Exibir os resultados das avaliações em gráficos utilizando a API Google Chart;
- Testar e analisar o sistema desenvolvido e apresentar os resultados.

1.3 Estrutura do Trabalho

O presente trabalho está dividido em cinco capítulos, no primeiro capítulo encontra-se a introdução com as motivações e os objetivos que levaram este trabalho a ser realizado. No capítulo 2 tem-se um referencial teórico sobre as tecnologias utilizadas para a construção do sistema, como conceitos, formas de utilizações de desenvolvimento das tecnologias Google Chart API, Google Earth API, PHP (*Hypertext Preprocessor*), Bootstrap e PostgreSQL. No capítulo 3 encontra-se a metodologia que demonstra a forma como o projeto foi desenvolvido. O capítulo 4 contém os resultados obtidos desde a confecção da modelagem do sistema até a sua conclusão. E por fim, o capítulo 5 com as considerações finais.

2 REFERENCIAL TEÓRICO

Nessa seção serão apresentados os conceitos sobre as tecnologias que foram utilizadas para o desenvolvimento do sistema de localização e avaliação de restaurantes, como a API Google Earth no desenvolvimento de mapas e API Google Chart para criação de gráficos, sistemas existentes, linguagem de programação utilizada, banco de dados e layout responsivo.

2.1 SISTEMAS EXISTENTES

Procurando conhecer aplicativos ou sistemas existentes, foram analisados alguns aplicativos que possuem características parecidas com as do sistema, dos quais a maioria para dispositivos móveis. Untappd é uma rede social para apreciadores de cerveja, trata-se de um aplicativo disponível para iPhone, Android, Windows Phone, Black Berry10 e também *web*, que disponibiliza ao usuário marcar quais cervejas já bebeu, além de compartilhar com os amigos as cervejas que mais gostou. Esse sistema também permite as cervejarias gerenciar sua marca com uma interface específica e fazer uma conexão com seus clientes, possibilitando um diálogo à respeito de seus produtos e analisar a tendência de consumo da marca dentro do aplicativo (UNTAPPD, 2015).

O aplicativo Veja Comer & Beber é um guia gastronômico disponível para iPhone e iPod Touch que possibilita encontrar restaurantes mais próximos, adicionar aos favoritos e anotar o que mais gostou de cada estabelecimento e está disponível para download na App Store. Este aplicativo traz informações de 10.000 estabelecimentos em vinte cidades e as respectivas avaliações pelos críticos da VEJA. São restaurantes e bares indicados e premiados pela revista VEJA em 2010. Sua atual versão permite que usuários possam avaliar restaurantes e bares e compartilhar sua resenha nas redes sociais (VEJA, 2015).

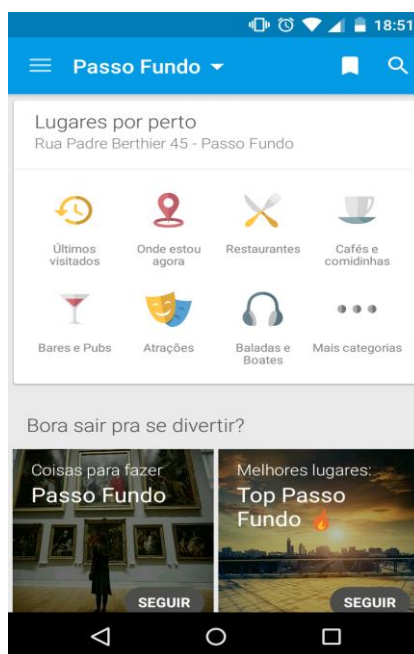
O Foursquare é um aplicativo que possibilita para o usuário realizar *check-ins* automáticos, dar dicas de estabelecimentos a partir dos *check-ins* realizados, avaliações de lugares semelhantes, e locais frequentados por amigos. Também permite que o usuário envie fotos com sua opinião sobre os estabelecimentos. Disponível para iPhone, Android, Windows Phone, *web* e também para iPad, Apple Watch. Em 2009 o Foursquare lançou o *check-in* e compartilhamento de localização em tempo real e cinco anos depois lançou o Swarm um aplicativo próprio para *check-in*, esse aplicativo é uma maneira mais rápida e fácil de acompanhar e encontrar amigos. O Foursquare pode ser encontrado também no Facebook,

Twitter e Medium (FOURSQUARE, 2015).

O Kekanto é um guia de bares, restaurante e “baladas” que dá sugestões conforme a localização do usuário. Possui um diferencial que são as avaliações dos estabelecimentos, onde cada usuário pode publicar sua resenha e enviar fotos. Ainda possui integração com Twitter e Facebook enfatizando um caráter social, possibilitando que utilizadores dessas redes sociais saibam onde seus seguidores e amigos realizaram *check-in*. É um aplicativo gratuito com versões para iOS, Android, Windows Phone e BlackBerry (PREVIDELLI, 2014).

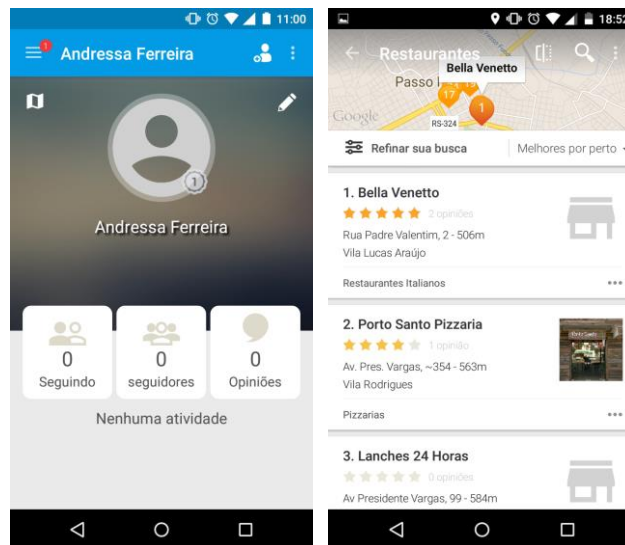
O Kekanto é o guia da cidade, ajuda a encontrar tudo na cidade de forma rápida e fácil, levando todas as informações no bolso. Esse aplicativo que possui um sistema web integrado, possibilita ao usuário encontrar restaurantes, ruas, cafés, hotéis, locais para compras, lugares de uma determinada cidade. Na Figura 2 pode-se visualizar a página inicial do aplicativo que é exatamente igual ao sistema web (KEKANTO, 2015).

Figura 2 – Aplicativo Kekanto



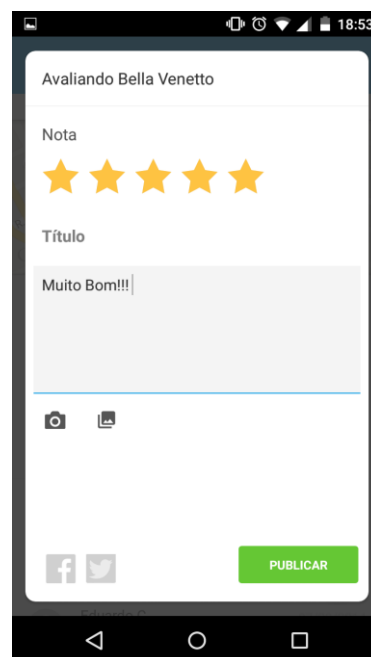
Fonte: SISTEMA, 2015

Ao criar uma conta no sistema/aplicativo, o usuário encontrará uma rede social, pois pode seguir e possuir seguidores, como também visualizar as opiniões que já foram realizadas. Ao escolher uma categoria, como por exemplo restaurantes, o sistema lista os restaurantes cadastrados da cidade em que o usuário está, informando endereço e telefone do restaurante (KEKANTO, 2015). Na Figura 3 encontra-se a tela da conta do usuário, e a listagem dos restaurantes.

Figura 3 – Conta do Usuário e Visualização de estabelecimentos

Fonte: SISTEMA, 2015

As avaliações disponibilizadas no sistema Kekanto são chamadas de opinião, é um espaço onde o sistema oferece ao usuário a possibilidade de criar uma resenha informando sua opinião sobre o estabelecimento. O usuário avalia o local marcando a quantidade que acha válida de estrelas (sendo no máximo 5), essas estrelas possuem um significado para a avaliação: precisa melhorar, regular, bom, muito bom e irretocável (KEKANTO, 2015). Encontra-se na Figura 4 a tela de avaliação de um restaurante utilizando o aplicativo do sistema.































Figura 4 – Avaliação

Fonte: SISTEMA, 2015

Conforme informações disponíveis no site Kekanto (2015), o sistema disponibiliza aos proprietários de estabelecimentos planos para possuírem um acesso diferenciado além de formas de marketing, escolhendo um plano que melhor lhe satisfaça. Todos os planos oferecem: edição de informações, postagem e gerenciamento de fotos e vídeos, responder opiniões e escolher a opinião de destaque de seu estabelecimento, publicar ofertas, receber orçamentos, solicitar matérias promocionais e gratuitas.

Diferente dos aplicativos citados anteriormente, o sistema desenvolvido possui ações que satisfará tanto a clientes como a proprietários dos restaurantes, pois o cliente que acessá-lo encontra os restaurantes, bem como, suas informações com a possibilidade de avaliá-lo e visualizar os resultados dessa avaliação, possibilitando, desta forma a contribuição com a melhoria do estabelecimento. O proprietário do restaurante possui um acesso diferenciado do cliente, pois ele encontrará os resultados através dos gráficos que serão criados com a API do Google Chart. Pode ser visualizada no Quadro 1 uma comparação dos sistemas e suas funções.

Quadro 1 – Comparação de Sistemas

Sistemas Funções	Sistema Desenvolvido	Kekanto	Untappd	Veja Comer & Beber	Foursquare
Localização de Estabelecimentos					
Visualizar Cardápio do dia					
Avaliação do Estabelecimento					
Avaliação por itens					
Resultados de Avaliações em gráficos					
Área administrativa para proprietários					

Fonte: Do Autor

 Contempla essa funcionalidade  Não contempla essa funcionalidade

2.2 UTILIZANDO MAPAS

Segundo Francisco (2014) mapa é uma representação gráfica de um espaço real, em uma superfície plana, como o papel. Nele podem ser representados desde bairro, cidade, estado ou país. Portanto os mapas podem ser criados conforme a abordagem do estudo e do objetivo que se deseja alcançar. No mapa podemos fazer análises de vários locais, sejam eles próximos ou distantes.

Pensando justamente em deixar essa análise e busca por locais mais interessantes a Google disponibilizou o Google Earth, que nada mais é do que um globo virtual que permite visualizar imagens, mapas, terrenos, construções em 3D (três dimensões) via satélite. Isso possibilita ao usuário ter uma visualização muito mais realista do globo terrestre.

Aproveitando a tecnologia Google Earth adquirida, a Google resolveu passar essa comodidade a desenvolvedores de sistemas *web*, foi então que surgiu Google Earth API, uma forma inovadora de disponibilizar um globo 3D e mapas em páginas *web*.

A API JavaScript² com o *plug-in* do Google Earth permite incorporar mapas em um sistema *web*. Utilizando a API é possível desenhar marcadores e linhas, projetar imagens em terrenos, permitindo a criação de um aplicativo de mapas, podendo empregar a tecnologia em apenas uma linha de código (EARTH, 2014).

Dessa forma a API Google Earth destina-se a desenvolvedores que possuem um conhecimento prévio em JavaScript e em conceitos de programação orientada a objetos. Essa API é modelada com base em KML (*Keyhole Markup Language*) para a exibição de dados geográficos em um navegador *web*.

A API Google Earth permite a sua utilização gratuitamente, fazendo com que o desenvolvedor tenha acesso, através de suas licenças, a: características de marca como nomes comerciais, marcas, logotipos, nomes de domínios. Conteúdo incluindo mapa e terreno de dados, imagens. Disponibiliza também a possibilidade de utilizar seu conteúdo próprio na utilização da API. Porém a API possui restrições de uso como: o programador só pode acessar o conteúdo da API que não seja através do serviço disponibilizados, o Google pode exigir um identificador de desenvolvedor para acessar e administrar o serviço e por fim o desenvolvedor não pode esconder a identidade da sua implementação da API. A API do Google Earth pode ser encontrada em: <https://developers.google.com/earth/>

² É uma linguagem de programação interpretada.

2.2.1 Exibição de dados geográficos

KML é um formato de arquivo utilizado para exibição de dados geográficos em um navegador, como Google Earth, Google Maps e Google Maps para celular. O KML utiliza uma estrutura de *tags* com elementos e atributos aninhados e se baseia no padrão XML (*eXtensible Markup Language*). Todas as *tags* diferenciam maiúscula de minúscula e devem aparecer exatamente como estão listadas na referência KML. A referência indica quais *tags* são opcionais. Em determinado elemento, as *tags* devem aparecer na ordem mostrada na referência KML (KML, 2014).

Através da interface de usuários da Google Earth pode-se criar elementos que incluem marcadores, descrições, superposições de solo, caminhos e polígonos. Porém possuem elementos que exigem a criação de KML que pode ser feita com um editor de texto, pois os navegadores sabem como exibir arquivos de texto que são salvos com a extensão .kml ou .kmz (KML, 2014).

Arquivos do tipo KML podem ser criados diretamente do Google Earth, ou seja, não tem a necessidade de criá-los em um editor de texto, ou seja, marcadores, superposições de solo, caminhos e polígonos serão armazenados em um arquivo KML que foi gerado automaticamente. Um exemplo da vantagem dessa funcionalidade pode ser nos marcadores, pois ao gerar o arquivo de forma automática dados como latitude e longitude são adquiridos com mais praticidade.

O marcador é um dos elementos mais usados no Google Earth. Ele marca uma posição na superfície da Terra, sendo que o mais simples inclui apenas um elemento *<Point>*, que especifica o local determinando um nome e um ícone que podem ser personalizados e também adicionar outros elementos de geometria a ele, como pode ser visto na Figura 5 exemplos de ícones de marcadores (KML, 2014).

Figura 5 – Exemplo de marcadores



Fonte: CODE, 2014

A inserção de marcadores simples, que pode ser visto e encontrado na Referência KML³, é muito prática, para iniciar é necessário especificá-lo com a tag `<Placemark>` na qual será definido um *id* para diferenciar os marcadores existentes. Dentro da tag estão as tags de identificação do marcador que são `<name>` para o nome, `<description>` com a descrição do marcador e `<Point>` que possui uma tag `<coordinates>` que com as coordenadas se tem latitude e longitude do ponto a ser marcado no mapa. Na Figura 6, trecho de código que gera os marcadores no mapa (KML, 2014).

Figura 6 – Exemplo de criação de Marcadores Simples

```

16 <Placemark id="mountainpin1">
17   <name>Frigideira</name>
18   <description>
19     <![CDATA[
20       Aspirante Jener, 5433112232
21       Frigideira<br>
22     ]]>
23   </description>
24   <styleUrl>#pushpin_1</styleUrl>
25   <Point>
26     <coordinates>-52.38749519750039,-28.26776233051961</coordinates>
27   </Point>
28 </Placemark>

```

Fonte: Do Autor

Através dos marcadores com seus textos, é possível adicionar *links*, tamanhos, estilos e cores de fontes e especificar o alinhamento de texto e tabelas. Além disso, há também um elemento de marcação automática, no qual converte textos em *hiperlinks* ativos, bastando

³ Disponível em: <https://developers.google.com/kml/documentation/kmlreference>

somente um clique do usuário para habilitar seu funcionamento (KML, 2014).

Diante disso percebe-se que, para a utilização da Google Earth, necessita-se da utilização de arquivos KML, que farão toda a diferença ao escolher a forma de demonstração dos dados no mapa, dessa maneira pode-se personalizar a visualização dos dados no mapa através das *tags* do KML. Todo arquivo KML deve ser gerado a *namespace* padrão como em arquivos XML, porém nesse caso deve-se utilizar `xmlns="http://earth.google.com/kml/2.1"` que é a padrão do KML.

O *plug-in* da API Google Earth é suportado pelos seguintes navegadores: Google Chrome, Internet Explorer, Firefox, Flock e Safari. Para sua utilização é necessário carregar a API Google Earth, criar um elemento *div* para armazenar o *plug-in* e funções para iniciá-lo, chamar as funções de inicialização quando a página estiver carregada. Na Figura 7 é apresentado um exemplo de como carregar o *plug-in* da API, que deve ser utilizado para criação de todos os tipos de mapas. Pode-se notar que é necessário abrir um `<script>` do tipo *text/javascript* e inserir o caminho dentro de *src* (KML, 2014).

Figura 7 – Carregando Plugin

```

12
13      <!--Carregando a API -->
14      <script type="text/javascript" src="https://www.google.com/jsapi"></script>
15
16      <script type="text/javascript">
17          var vceCount = 0;
18          var vcTimeout = null;

```

Fonte: Do Autor

Para a gerar o mapa basta criar a função JavaScript que irá carregá-lo. Como pode ser visto na Figura 8, sendo necessário informar a localização desejada, ou seja, no início da função estão a latitude e longitude que representam a cidade de Passo Fundo em *var pf* e as opções de visualização do mapa em *var mapOptions*. Logo a seguir informa-se o *id* da *div* que será gerado o mapa com suas opções em *var map*, com essas informações já é possível visualizar o mapa. A seguinte informação a ser passada é a chamada do arquivo que gera o KML como pode ser visto em *var ctaLayer*. Por fim chama-se a função criada.

Figura 8 – Função para gerar o mapa

```
<script>

function initialize() {
    var pf = new google.maps.LatLng(-28.260456, -52.409858);
    var mapOptions = {
        zoom: 40,
        center: pf
    }

    var map = new google.maps.Map(document.getElementById('map_canvas'), mapOptions);

    var ctaLayer = new google.maps.KmlLayer({
        url: '<?php echo "http://187.7.88.182:6090/boletimkml.php?" . http_build_query($url_out); ?>'
    });
    ctaLayer.setMap(map);
}

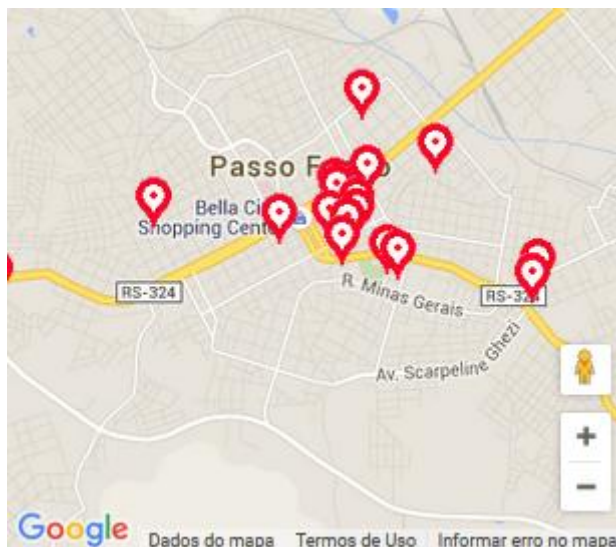
google.maps.event.addDomListener(window, 'load', initialize);

</script>
```

Fonte: Do Autor

Como pode ser visto na figura anterior a função criada chama um arquivo .php que gera automaticamente o KML, para gera-lo é necessário informar as predefinições necessárias para que o KML esteja em perfeita condições resultando na marcação dos restaurantes no mapa. Logo que é gerado o KML é armazenado em *cache*, ou seja, esse arquivo não é salvo no servidor, ele permanece em cache o que diminui a utilização de memória e também permite uma visualização perfeita dos marcadores no mapa.

Esse arquivo que gera os dados KML deve demonstrar as informações que são passadas ao usuário, nesse caso informações dos restaurantes como: nome, endereço, telefone, entre outras informações consideradas importantes para que o usuário tenha o conhecimento de qual restaurante aquele ponto representa. A Figura 9 mostra como os restaurantes cadastrados no sistema que estão apresentados no mapa.

Figura 9 – Marcadores no Mapa

Fonte: Do Autor

Através da Figura 10 é possível visualizar uma forma de personalização das informações, de como elas são mostradas ao usuário, da forma que o ponto no mapa é identificado e traga mais informações importantes para facilitar o acesso do usuário no sistema. Esses ícones no mapa que referenciam a localização do restaurante, são links que ao serem pressionados apresentam essas informações pré-definidas que são buscadas no banco de dados por um arquivo .php que gera o KML correspondente em memória cache

Figura 10 – Visualizando Informações no Mapa

Fonte: Do Autor

2.3 UTILIZANDO GRÁFICOS

A utilização de gráficos é uma forma de representação de dados conquistados de experimentos e pesquisas através de figuras geométricas, como desenhos, imagens, diagramas, que possuem o intuito de facilitar o entendimento dos leitores, para que seja de rápida e objetiva compreensão dos resultados obtidos.

Ao gerar gráficos, é preciso informar necessariamente o que for de grande importância, pois os gráficos são resumos dos dados adquiridos. Portanto deve-se evitar frases longas, para que o leitor de gráficos possa em apenas alguns minutos compreender do que se trata. Para que seja possível alcançar esse propósito, devemos criar o gráfico conforme o tipo de dado coletado, tornando-o mais prático e compreensível (SILVA, 2014).

Como a tecnologia está cada vez mais avançada, é possível utilizar programas de vários segmentos para criação de gráficos, dos quais disponibilizam gráficos mais atraentes e de fácil entendimento, com resultados muito práticos, para que não haja dúvidas quanto ao resultado. Desta forma a Google pensando em auxiliar os desenvolvedores que necessitavam de praticidade e facilidade na geração de gráficos em seus sites e aplicativos, passou a disponibilizar uma API que agiliza esse processo, tornando a manutenção desse código muito mais rápida.

Google Charts API possui uma galeria que disponibiliza dos mais simples aos mais complexos gráficos, fornecendo uma visualização perfeita para páginas web. Esses diversos tipos de gráficos disponíveis fazem com que o desenvolvedor tenha uma comodidade a mais por ter a certeza de que esses dados serão tratados corretamente e de forma simples. Na galeria podemos encontrar tipos de gráficos como: Quadros Geográficos, Gráfico de Dispersão, Gráficos de Colunas, Histograma, Gráfico de Barras, Gráfico Pizza, entre outros (CHARTS, 2014).

2.3.1 Exemplo de gráficos

Os gráficos disponibilizados na Galeria da API Google Charts são processados no navegador por SVG (*Scalable Vector Graphics*) ou VML (*Vector Markup Language*). Através das funções JavaScript que geram os gráficos acabam gerando esse formato automaticamente para serem processados pelo navegador sendo armazenados em cache para a visualização do usuário.

O SVG é um arquivo XML que contém *tags* específicas para criação de imagens

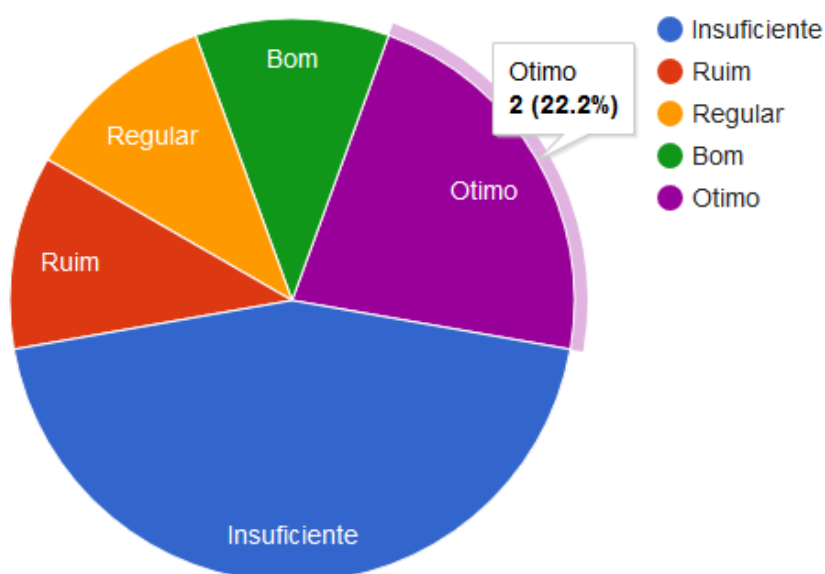
vetorizadas em uma aplicação é suportado por todos os navegadores *web*. Com *tags* muito simples é possível gerar imagens de alta qualidade, que por mais que se altere a resolução não perderá sua qualidade. Essas imagens podem ser tanto fixa como animação, o SVG pode ser trabalhado em conjunto com o JavaScript para manipular eventos nessas imagens. Nesse formato de arquivo encontramos três tipos de objetos gráficos, são eles: imagens, textos ou formas geométricas (DEV MEDIA, 2014).

O VML é uma linguagem baseada em XML, utilizada para criação de elementos gráficos bidimensionais e folhas de estilo em cascata para criar e posicionar elementos como círculos e quadrados, que também podem ser personalizados em vários programas gráficos (OFFICE, 2014), no caso da Google Chart API o VML é utilizado para processar imagens em navegadores com versões antigas, sendo assim o mais utilizado é o formato SVG.

Com a utilização desses formatos de arquivo, tanto SVG como VML, os gráficos gerados são processados pelo navegador e obtém maior qualidade de visualização, independente das proporções de tela, esses gráficos gerados serão visualizados de forma mais eficaz sem problemas de distorção de imagens. Na Figura 11 encontra-se o gráfico de pizza que é demonstrado aos restaurantes com o resultado das avaliações dos clientes por item avaliado.

Figura 11 – Resultado do restaurante por item avaliado

Item: Cardápio



Fonte: Do Autor

Para gerar os gráficos é necessário informar o *plug-in* da API como mostra a Figura 12 bastando apenas inserir conforme o *src* como mostra a figura.

Figura 12 – Importando plug-in da API

```
<!--Importação da API -->
<script type="text/javascript" src="https://www.google.com/jsapi"></script>
<script type="text/javascript">
```

Fonte: Do Autor

Ao criar a função JavaScript como mostra a Figura 13 é possível gerar os gráficos necessários para o sistema. Os gráficos dessa API são produzidos através de vetores, ou seja, as informações que deverão ser visualizadas no gráfico devem ser passadas através de uma matriz ou *array* de duas dimensões. Na figura encontra-se a *var data* que é onde os dados são informados para que sejam visualizados no gráfico. Em *var options* encontra-se as opções como o título do gráfico, se o gráfico é 3D com a opção *is3D: true*. Para carregar o pacote de visualização do gráfico em pizza deve-se inserir *PieChart* como pode ser encontrado em *var chart*. Com as informações passadas na função é possível gerar o gráfico como mostra a imagem a seguir, carregando o modulo de visualização e escolhendo as informações que serão demonstradas.

Figura 13 – Exemplo Código Função que Gera Gráficos

```
function drawChart() {
    var data = google.visualization.arrayToDataTable(<?php echo json_encode($grafico); ?>);

    var options = {
        pieSliceText: 'label',
        title: 'Item: <?php echo exibir_iten($iten); ?>',
        pieStartAngle: 100,
        is3D: true
    };

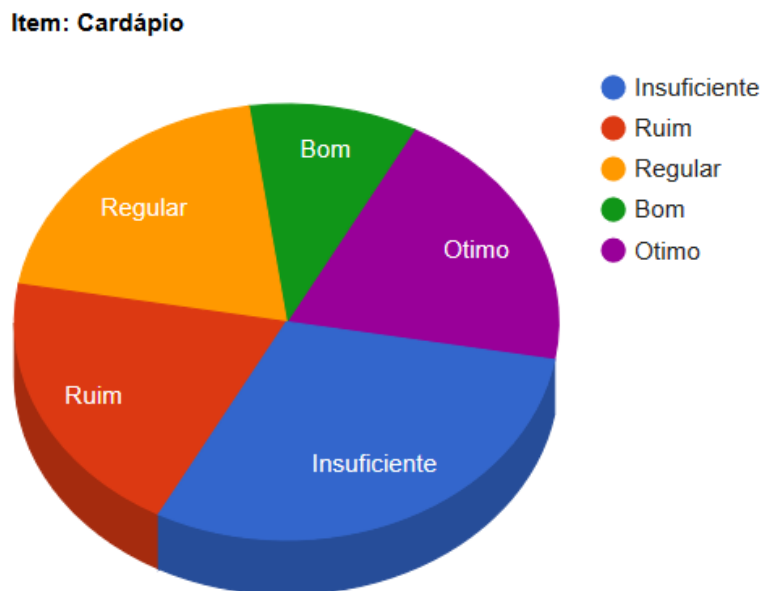
    var chart = new google.visualization.PieChart(document.getElementById('grafico'));
    chart.draw(data, options);
}
```

Fonte: Do Autor

Esse tipo de gráfico também pode ser criado de forma personalizada, um exemplo é o

gráfico em 3D como pode ser visto na Figura 14.

Figura 14 – Gráfico Pizza 3D. Resultado do restaurante por item avaliado

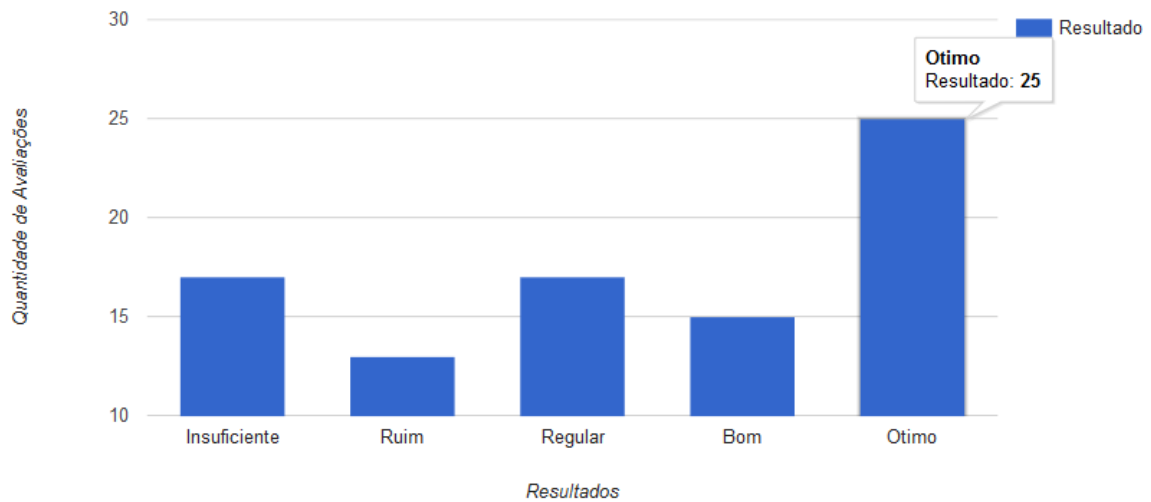


Fonte: Do Autor

A API Google Charts possui uma galeria com diversos tipos de gráficos, disponibilizados para que o desenvolvedor possa escolher qual demonstrará melhor os dados do sistema. No sistema desenvolvido foram utilizados 3 tipos de gráficos: gráfico de pizza demonstrado anteriormente (Figuras 12 e 13), gráfico de colunas e o gráfico combo.

Para visualizar o gráfico em colunas utiliza-se a mesma função do gráfico pizza, basta informar os dados de visualização para o vetor, modificar as opções como título e outras informações que achar necessário. O que não deve ser esquecido é o pacote de visualização ColumnChart, o qual deve ser modificado como a seguir: `var chart = new google.visualization.ColumnChart(document.getElementById('grafico'));` na Figura 15 encontra-se o gráfico em colunas com o resultado geral das avaliações de um restaurante.

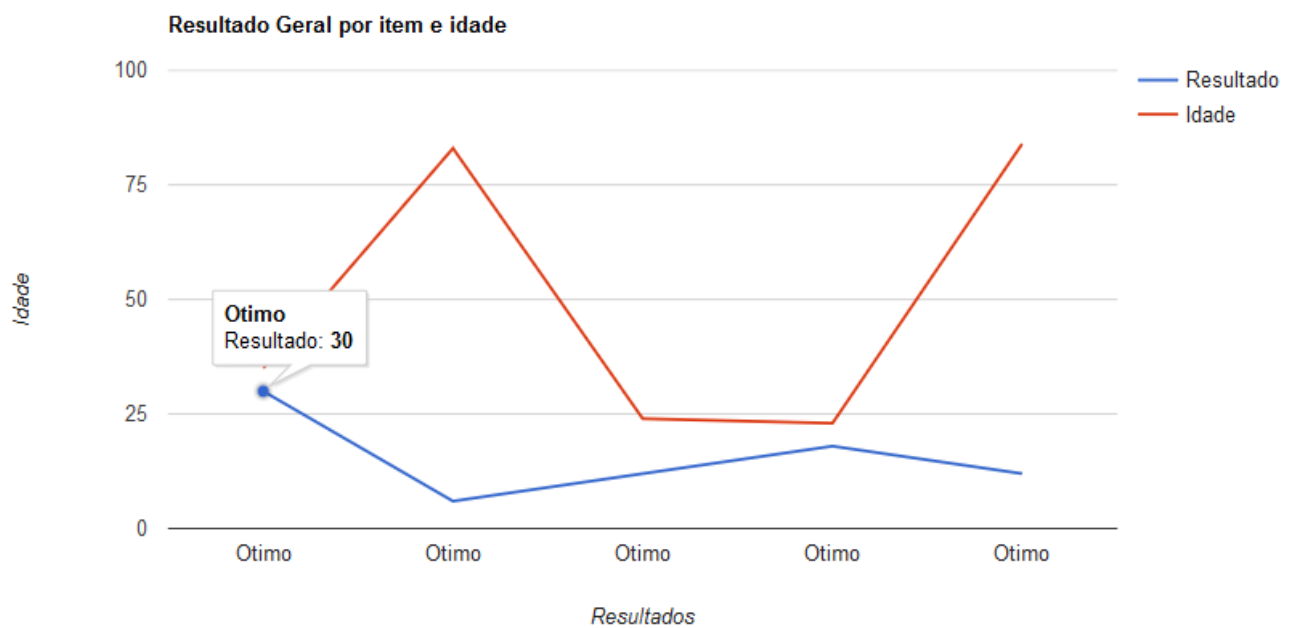
Figura 15 – Gráfico em Colunas. Resultado geral do Restaurante



Fonte: Do Autor

Na Figura 16 um exemplo do gráfico combo. Para gerar o gráfico combo deve-se realizar as mesmas modificações que nos outros tipos de gráficos como os dados a serem visualizados e as opções. Também deve ser modificado o pacote de visualização para o *ComboChart*, como demonstrado a seguir a modificação deve ser em: `var chart = new google.visualization.ComboChart(document.getElementById('grafico'));`

Figura 16 – Gráfico Combo



Fonte: Do Autor

2.4 MODELAGEM DO SISTEMA

Segundo Booch et al (2012, p.4), a modelagem é uma parte central de todas as atividades que levam a implantação de um bom software. Os modelos são construídos para comunicar a estrutura e o comportamento do sistema, visualizar e controlar a arquitetura do sistema, compreender melhor o sistema que está sendo elaborado, expor oportunidades de simplificação e reaproveitamento, e também para gerenciar os riscos.

A modelagem de software é tão importante quanto a modelagem da construção de uma casa, que é uma técnica da engenharia aprovada e bem-aceita. Sendo assim o modelo é uma simplificação da realidade, que podem abranger de planos detalhados até planos mais gerais com uma visão geral do sistema (BOOCH et al, 2012).

A modelagem de software orientada a objetos possui um valor importante na construção de sistemas, pois ela abrange todos os tipos de domínios de problemas e diferentes graus de tamanho e complexidade. Para criar a modelagem do sistema desenvolvido, utilizou-se a linguagem UML (Unified Modeling Language) que pode ser empregada para visualização, especificação, construção e documentação de artefatos de sistemas complexos de software (BOOCH et al, 2012).

De acordo com Booch et al (2012), a UML uma linguagem, portanto, somente faz parte do desenvolvimento de software. Essa linguagem para visualização de modelos que facilitam a comunicação pode ser utilizada em domínios como: sistemas de informações corporativos, serviços bancários e financeiros, telecomunicações, transportes, defesa/espaço aéreo, vendas de varejo, eletrônica médica, científico e serviços distribuídos baseados na *web*.

Através dessa linguagem são disponibilizados vários tipos de diagramas que podem ser utilizados para a modelagem do sistema e que estão divididos em dois tipos: visualização de partes estáticas e visualização de partes dinâmicas (BOOCH et al, 2012).

Os diagramas para visualizar partes estáticas são:

- Diagrama de classes
- Diagrama de componentes
- Diagrama de estrutura composta
- Diagrama de objetos
- Diagrama de implantação

- Diagrama de artefatos

Diagramas de visualização dinâmica:

- Diagrama de caso de uso
- Diagrama de sequências
- Diagrama de comunicação
- Diagrama de atividades
- Diagrama de estados

No projeto desenvolvido foram utilizados dois diagramas o diagrama de classes que são diagramas frequentemente encontrados na modelagem de sistemas orientados a objetos, é um diagrama que demonstra um conjunto de classes, interfaces e colaborações e seus relacionamentos. O diagrama de casos de uso mostra o comportamento de um sistema, de um subsistema ou de uma classe. Cada diagrama informa um conjunto de casos de uso e atores e seus relacionamentos (BOOCH et al, 2012).

2.5 LINGUAGEM DE PROGRAMAÇÃO PHP

Segundo Soares (2011) PHP é acrônimo de *Hypertext Preprocessor* (pré-processador de hipertexto), é uma linguagem de programação *open source*, utilizada mundialmente, sendo principalmente no ambiente *web*. Sua principal característica é a capacidade de se misturar ao HTML (*HyperText Markup Language*), facilitando a geração de páginas dinâmicas e a integração com o usuário.

De acordo com Converse (2003, p. 3):

O PHP é uma linguagem de criação de *scripts* do lado do servidor, que pode ser incorporada em HTML ou utilizada como um binário independente (embora a primeira utilização seja muito mais comum). Outros produtos “proprietários” ou patenteados nesse setor de mercado são o ASP (*Active Server Pages*) da Microsoft, o ColdFusion da Macromedia e o JSP (*Java Server Pages*) da Sun.

Ainda conforme Converse (2003) o *script* do lado do servidor é uma coleção de tags super-HTML ou pequenos programas que executam dentro de páginas web. Um exemplo é a

utilização do PHP para adicionar cabeçalhos e rodapés nas páginas de um site ou armazenar dados de um formulário em um banco de dados. O PHP tem pouco a ver com *layout* ou qualquer coisa relacionado a aparência de uma página *web*. A maior parte do que o PHP realiza é invisível para o usuário final.

O PHP é um módulo oficial de servidor HTTP Apache, o servidor web líder de mercado e gratuito, isso significa que o mecanismo de script do PHP pode ser construído no próprio servidor, o que faz com que a manipulação de dados seja mais rápida. Como o servidor Apache, essa linguagem é multiplataforma, executa em várias versões do UNIX, bem como Windows e Mac OS X. Todos os projetos da Apache Software Foundation que incluem o PHP são de código-fonte aberto (CONVERSE, 2003).

PHP um sucessor de um produto chamado PHP/FI que incluía o Form Interpreter, uma ferramenta de análise sintática de consultas SQL, criado em 1994 por Rasmus Lerdof para a sua utilização pessoal, tratava-se de um wrapper (“envoltório”) que o auxiliava a monitorar as pessoas que acessavam seu site. No ano seguinte ele montou um pacote chamado *Personal Home Page Tools* (PHP Tools, também conhecido como *PHP Construction Kit*) em resposta a demanda de usuários que faziam uso do seu trabalho (CONVERSE, 2003).

No mesmo ano, Rasmus expandiu o PHP, o qual mudou seu nome, utilizando a tecnologia FI (*Forms Interpreter*), uma nova implementação que incluiu funcionalidades básicas do PHP conhecidas hoje, como variáveis no estilo *Perl*, interpretação automática de variáveis de formulários, e sintaxe PHP embutida. Essa sintaxe era muito limitada, um pouco inconsistente, pois para embutir o código em um arquivo HTML (*HyperText Markup Language*), era necessário utilizar comentários HTML, por isso esse método não foi bem recebido pelos desenvolvedores, porém ainda continuou em crescimento, mas não como linguagem (PHP HISTÓRIA, 2014).

As mudanças começaram em Outubro de 1995, Rasmus liberou uma completa reescrita do código. Trazendo de volta o nome PHP (*Personal Home Page Construction Kit*) e foi o primeiro lançamento considerado um avanço de interface *scripting*. A linguagem foi adotada por desenvolvedores habituados com C, Perl e linguagens similares. Tendo sido limitada para sistemas UNIX e sistemas compatíveis com POSIX, com potencial de implementação que estava sendo explorada em um Windows NT (PHP HISTÓRIA, 2014).

Conforme Converse (2003), no ano de 1997 houve uma grande evolução, o PHP era utilizado mundialmente em aproximadamente 50.000 sites, o que acabou tornando-se

complicado para uma pessoa administrar. Sendo assim, uma pequena equipe de desenvolvimento mantinha o projeto através do código-fonte aberto, com contribuições de desenvolvedores e usuários em todo o mundo. A seguir é possível analisar o crescente aumento de funcionalidades do PHP a cada mudança de versão.

A versão 3 do PHP, lançada em 1998, foi a primeira versão que se assemelha com o PHP que existe no momento segundo PHP História (2014). Seu principal ponto forte foram os recursos de extensibilidade, além de fornecer uma interface robusta para múltiplos bancos de dados, protocolos, e APIs o que acabou atraindo dezenas de desenvolvedores, outra atribuição importante foi o suporte a programação orientada a objeto e uma consistente sintaxe de linguagem.

De acordo com PHP História (2014), logo após o lançamento oficial do PHP 3.0 foi iniciada uma reescrita do código base do PHP, os objetivos eram melhorar performance das aplicações complexas, e melhorar a modularização do código base. O PHP 4, lançado oficialmente em 2000, foi baseado em um novo motor chamado ‘Zend Engine’ que alcançou os objetivos com sucesso. Além de altíssima melhoria de performance, foi incluído outros recursos como suporte para maioria dos servidores web, sessões HTTP (*HyperText Transfer Protocol*), saídas de *buffering*, entre outras funções adquiridas.

O PHP 5 segundo PHP História (2014), foi lançado em 2004, após longo período de desenvolvimento e vários pré-lançamentos, como seu core Zend Engine 2.0 possuindo características, novo modelo de objetos e dezenas de outros recursos. O PHP possui dezenas de desenvolvedores trabalhando em seu desenvolvimento ou algo relacionado ao PHP e apoio a projetos como o PEAR⁴, PECL⁵, documentação, infra-estrutura de rede subjacente com mais de uma centena de servidores web em seis dos sete continentes do mundo.

O que diferencia o PHP do JavaScript é a interação com o usuário, pois como o código é executado no servidor *web* ele gera um HTML que é então enviado para o cliente, que recebe os resultados desta execução sem saber como é o seu código fonte, tornando-se então uma forma prática e simples, que o usuário nem percebe como é feito (PHP, 2014). O JavaScript é mais indicado para assuntos e funções que não necessitam de um servidor, sendo assim executados diretamente no navegador sem perda de tempo, como no caso de tratamento

⁴ PEAR é um *framework* e distribuição para componentes reutilizável, disponível em: <http://pear.php.net/>

⁵ PECL é um repositório de extensões PHP que estão disponíveis pelo sistema de pacote PEAR. Disponível em: http://php.net/manual/pt_BR/install.pecl.intro.php

de eventos (*onclick*, por exemplo). Já o PHP é recomendado para tarefas que necessitam de busca de dados em banco e demais tarefas que necessitam de um melhor tratamento utilizando o servidor.

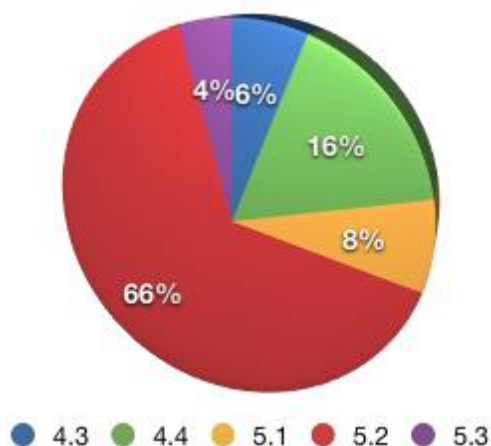
Atualmente o PHP está na versão 5 e conforme Soares (2011), esta versão foi totalmente remodelada para uma melhor aderência ao conceito de OO (Orientação a Objetos). Até a versão 4 o PHP tratava objetos como tipos primitivos (da mesma forma que trata inteiros ou *strings*). A OO é o desenvolvimento de uma estratégia em que os sistemas são construídos baseados em uma coleção de componentes reusáveis conhecidos como objetos, esses objetos possuem duas características: Objetos possuem dados e Objetos manipulam dados e fazem coisas.

Diferentemente da programação estruturada, a orientação a objetos reúne de forma inteligente os dados e seus manipuladores. Os termos básicos da OO são: Objeto que é a entidade que será generalizada. A classe é a abstração de um objeto, ou seja, um modelo a partir do qual os objetos serão criados. O atributo pertencente a classe refere-se a representação de dados que a classe possui. Os métodos que definem o que as classes sabem fazer, são funções que alteram os atributos (SOARES, 2011).

De acordo com Soares (2011) a linguagem de programação PHP é bem dinâmica, pois ela roda em qualquer sistema operacional e acessa diferentes bancos de dados, facilitando assim a sua utilização. Sendo assim essa linguagem agrada diferentes tipos de usuários que para sua utilização não necessitam modificar o ambiente de desenvolvimento do qual estão acostumados.

O PHP é a linguagem mais utilizada na atualidade, o gráfico a seguir apresentado na Figura 17 demonstra a porcentagem de domínios ativos com PHP no ano de 2010. O gráfico representa a utilização da linguagem através de suas versões, percebe-se que quase 80% migraram para a versão 5 do PHP.

Figura 17 – Utilização do PHP



Fonte: IMASTERS, 2015

2.6 BANCO DE DADOS

Para armazenar os dados manipulados no estudo de caso proposto, optou-se pela utilização do banco de dados PostgreSQL que é um SGBDR – Sistema Gerenciador de Base de Dados Relacional que está baseado nos padrões SQL (*Structured Query Language*) ANSI-92, 96 e 99, de alta performance, de fácil administração e utilização em projetos. O PostgreSQL surgiu em 1986, na Universidade Norte-Americana da Califórnia, a Universidade de Berkeley, e desde então vem sendo aprimorado com muitos benefícios e funcionalidades (PEREIRA NETO, 2007).

No ano de 1995 o PostgreSQL obteve um interpretador SQL, sendo chamado de Postgres95. A mudança tratava de um aumento de 25% na linha de código, tornando-se um SGBDR como todos os seus outros concorrentes. No entanto a abertura completa de seu código-fonte se deu em meados de 1996, e neste momento já se tornava conhecido como PostgreSQL. Tratava-se do mesmo código-fonte original, porém com o interpretador SQL construído por Andrew Yu e Jolly Chen. E desde então várias contribuições vêm sendo agregadas ao produto a fim de posicioná-lo de forma extremamente competitiva em nível de funcionalidade com os SGBDRs do mercado, porém com uma principal vantagem: é totalmente livre e de código-aberto (PEREIRA NETO, 2007).

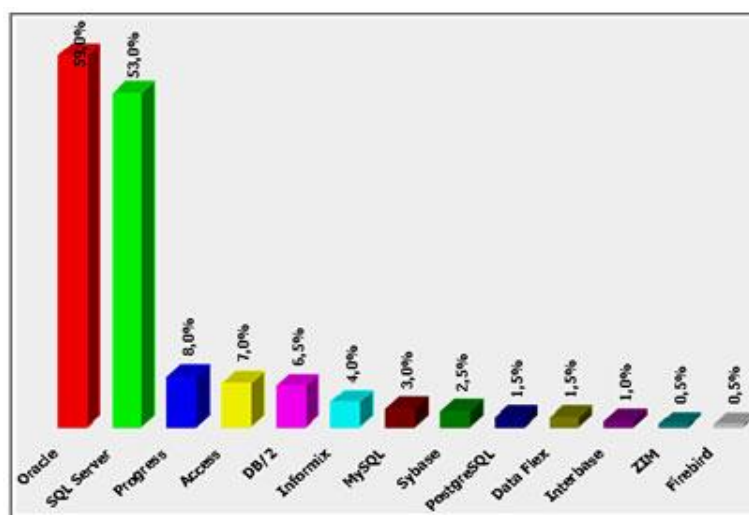
Ainda possui funcionalidades sofisticadas como o controle de ocorrência multirreverso, recuperação em um ponto no tempo, *tablespaces*, replicação assíncronas, transações agrupadas, cópias de segurança, um sofisticado planejador de consultas e

registrator de transações sequencial para tolerância a falhas (POSTGRESQL, 2014).

Facilitando ainda mais a codificação, suporta conjuntos de caracteres internacionais, codificação de caracteres *multibyte*, *Unicode* e sua ordenação por localização, possui sensibilidade a maiúsculas, minúsculas e formatação. É altamente escalável, tanto na quantidade de dados que serão gerenciados quanto no número de usuários concorrentes que pode acomodar.

Durante a escrita deste trabalho a versão atual do PostgreSQL é a 9.4 lançada no final de 2014, essa versão adquiriu várias melhorias nos comandos SQL, armazenamento de dados e demais funções que auxiliam na manutenção dos dados. Com tudo uma pesquisa realizada em 2005 demonstra a utilização de bancos de dados por empresas brasileiras, na Figura 18 é possível visualizar que o PostgreSQL é o terceiro banco de dados mais utilizado no Brasil (PRADO, 2015).

Figura 18 – Utilização de Banco de Dados por Empresas Brasileiras



Fonte: PRADO, 2015

2.7 LAYOUT RESPONSIVO – BOOTSTRAP

O Bootstrap foi criado em 2011 pela equipe de Engenharia do Twitter para resolver um problema interno de inconsistência de desenvolvimento. Os problemas comuns enfrentados por essa equipe se dava na estruturação do código, ou seja, cada engenheiro codificava à sua maneira. Quando uma empresa possui um grande número de engenheiros isso torna-se um problema de inconsistência grave, aumentando o tempo de manutenção do código (UTTERBACK, 2014).

Segundo Utterback (2014), com esses problemas detectados o Bootstrap foi criado inicialmente para incentivar o uso de uma única estrutura, reduzindo as inconsistências no código. A partir de então foi possível desenvolver códigos com mais rapidez e eficiência, com menos inconsistências, permitindo que toda a equipe trabalhasse da mesma forma. Percebendo o grande potencial da tecnologia que foi criada para resolver um problema interno, foi lançado em agosto de 2011 como um projeto de *software* livre, e com o passar do tempo tornou-se cada vez mais conhecido, e em uma das estruturas mais popular em desenvolvimento de projetos móveis com resposta na *web*.

O Bootstrap é uma coleção composta por elementos e funções acessíveis em uma única ferramenta. Esses elementos e funções são uma combinação de HTML (*HyperText Markup Language*), CSS (*Cascading Style Sheets*) e JavaScript. Sendo um software livre, ele é aprimorado constantemente, tendo como principal vantagem 100% de capacidade de resposta móvel e opções jQuery⁶ com muitos recursos (UTTERBACK, 2014).

Durante o desenvolvimento do sistema o Bootstrap disponibiliza a versão 3 e conta com um novo *design*, melhor tratamento de erros, mais e aprimoradas opções de personalização dentre outros benefícios que facilitam e auxiliam na construção de um sistema *web*, pois seu constante aprimoramento atende às necessidades dos desenvolvedores.

Possibilitando a criação de *layout* responsivo, permite organizar o conteúdo em 12 colunas que se adaptam conforme o tipo de resolução, permitindo a resposta em dispositivos móveis, sendo assim foram diagnosticadas vantagens e desvantagens (UTTERBACK, 2014).

Vantagens:

- Possui documentação detalhada e de fácil entendimento;
- É otimizado para o desenvolvimento de *layouts* responsivos;
- Possui componentes suficientes para o desenvolvimento de qualquer site ou sistema *web* com interface simples;
- Facilita a criação e edição de *layouts* por manter padrões;
- Funciona em todos os navegadores atuais (Chrome, Safari, Firefox, IE, Opera).

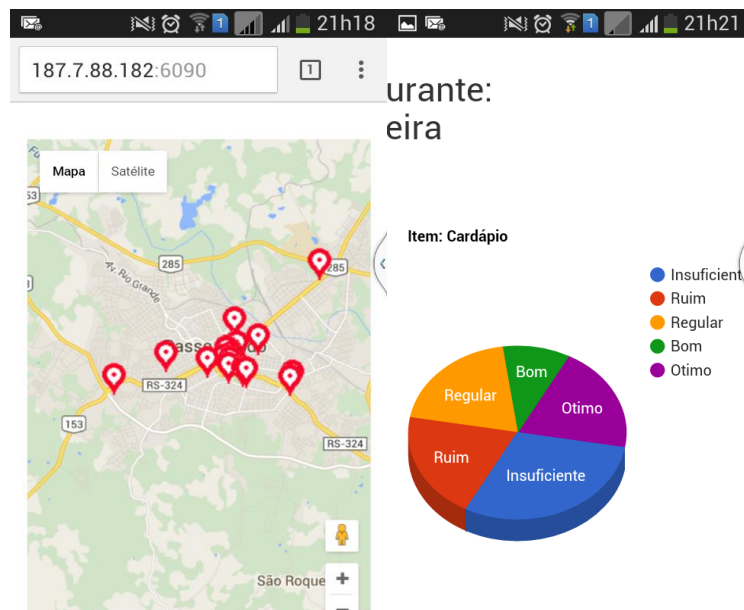
⁶ É uma biblioteca para desenvolvimento rápido de JavaScript que interage com a página HTML.

Desvantagens:

- O código terá de seguir os “padrões de desenvolvimento Bootstrap”;
- Tema padrão e comum do Bootstrap (caso não faça ajustes visuais, o projeto desenvolvido se parecerá com outros que também utilizam o Bootstrap).

Com uma estrutura simples e rápida, ela vem com os códigos já pré-definidos. Então basta adicionar o código que se está desenvolvendo, e em pouco tempo o *layout* está pronto para uso. Com essa grande facilidade o Bootstrap é muito recomendado para desenvolvedores iniciantes que possuem conhecimento básico de HTML e CSS, também para desenvolvedores que buscam o aumento da produtividade. Na Figura 19 foram tirados *print* da tela de um smartphone para uma visualização da utilização do Bootstrap.

Figura 19 – Visualização em Smartphones



Fonte: Do Autor

3. METODOLOGIA

O tipo de pesquisa realizada no presente trabalho é pesquisa exploratória. Conforme Cerro at al (2007):

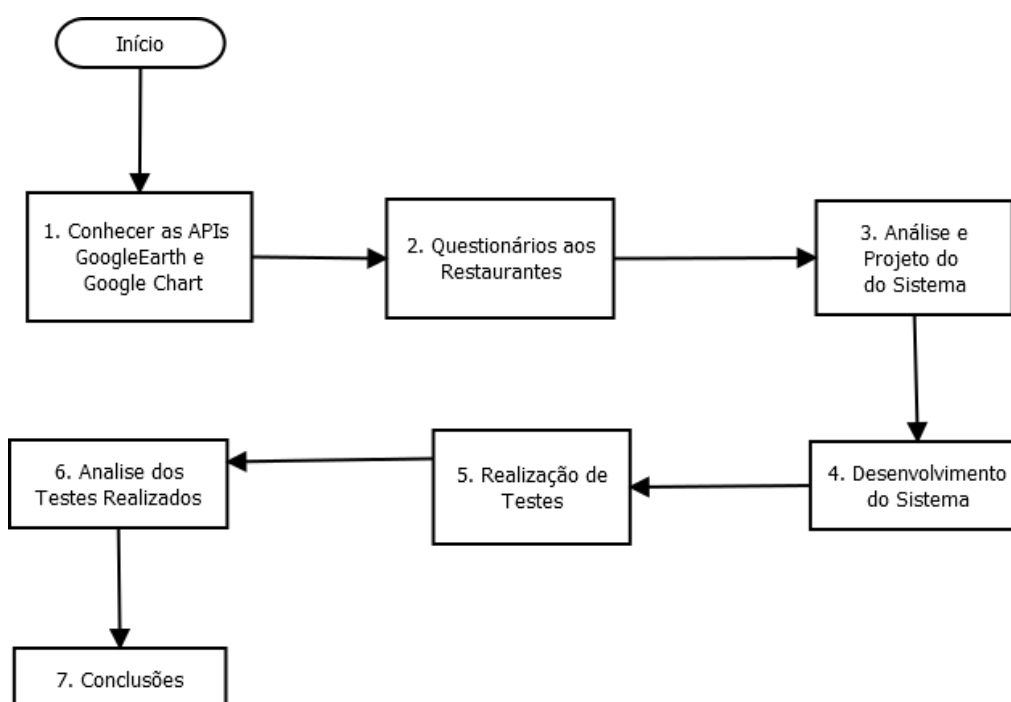
A pesquisa exploratória, designada por alguns autores como pesquisa quase científica ou não científica, é normalmente o passo inicial no processo de pesquisa

pela experiência e um auxílio que traz a formulação de hipóteses significativas para posteriores pesquisas. A pesquisa exploratória e a elaboração de hipóteses a serem testadas no trabalho, restringindo-se a definir objetivos e buscar mais informações sobre determinado assunto de estudo. Tais estudos tem por objetivo familiarizar-se com o fenômeno ou obter uma nova percepção dele e descobrir novas ideias.

Esta pesquisa tem como base o município de Passo Fundo que possui 392 restaurantes e 800 lancherias cadastrados com alvará de funcionamento ativos na prefeitura em específico na Secretaria de Desenvolvimento Econômico. Esses dados são constantemente alterados, pois há muita demanda tanto na solicitação de novos alvarás como no seu cancelamento. Dessa forma percebe-se o grande número de estabelecimentos nessa cidade, como também que muitos clientes talvez não tenham conhecimento sobre a existência de todos esses locais.

Na Figura 20 um fluxograma das etapas de desenvolvimento do sistema.

Figura 20 – Fluxograma de Desenvolvimento



Fonte: Do Autor

A seguir encontram-se as descrições das etapas representadas no fluxograma:

1. Conhecer as APIs Google Earth e Google Charts através de estudo do material bibliográfico disponibilizado pelo Google, para que assim fosse possível entender o funcionamento e suas formas de aplicação. Também foi obtido um aprendizado prático para compreender a forma de geração de gráficos e mapas.

2. Foi realizado um questionário a alguns restaurantes da cidade, para se ter uma análise sobre a aceitação de sistemas com esta característica por possíveis clientes, onde todos os entrevistados responderam que teriam interesse em um aplicativo desse tipo.
3. Nessa etapa foi realizado o levantamento de requisitos e sua modelagem para a construção do sistema, como também a definição dos diagramas de classes e casos de uso.
4. Assim que a análise e projeto do sistema ficaram prontos, a pesquisa passou para a etapa de desenvolvimento do sistema que foi realizado em PHP com as funções JavaScript para gerar o mapa e os gráficos, como também a utilização do layout responsivo e a criação do banco de dados.
5. Nessa etapa o sistema foi disponibilizado para testes em um servidor que permita a busca e carregamento das API pela internet, pois essa busca não é possível em um teste local, pois o *plug-in* da API Google Earth necessita estar conectado diretamente em um servidor com internet para carregar as funções pré-definidas pela API. Foram realizados testes funcionais para testar o comportamento das API com a quantidade de dados no sistema.
6. A partir da realização dos testes foram analisadas todas as funcionalidades pertinentes ao sistema, pois estas tecnologias possuem diversas funções com vários objetivos diferenciados, para que assim fosse possível chegar a conclusão sobre seus funcionamentos.
7. Após o término das etapas realizadas anteriormente, chegou-se as conclusões do projeto.

Ao obter os dados bibliográficos, foram realizados o projeto e a análise, confeccionando a documentação através da modelagem UML (*Unified Modeling Language*), utilizando diagrama de classes e diagramas de caso de uso para demonstrar o comportamento e principais funcionalidade do sistema. O desenvolvimento foi realizado na IDE (*Integrated Development Environment*) NetBeans com a linguagem PHP, utilizando os *plug-ins* das APIs Google Earth e Google Charts.

Através dos *plug-ins* foram gerados os mapas para a localização dos restaurantes com seus marcadores, demonstrando os restaurantes cadastrados no sistema, como também uma forma de busca por tipo de restaurantes que demonstrará somente os restaurantes daquela

especialidade. Para que o cliente possa avaliar o sistema, foi realizado um formulário de avaliação onde o cliente preenche e envia os resultados, sendo registrado no banco de dados, sendo assim, as informações armazenadas no banco serão utilizadas para criação do gráfico com o resultado das avaliações.

E justamente para que o restaurante receba essas avaliações dos seus clientes, foi disponibilizado para os proprietários dos restaurantes um ambiente em que são recebidos os resultados das avaliações. Todas essas funcionalidades foram desenvolvidas para que esse sistema possa auxiliar em um melhor relacionamento entre clientes e restaurantes.

A avaliação dos clientes foi construída com base em um estudo realizado por alunos no restaurante universitário da UFSC – Universidade Federal de Santa Catarina, que analisou os indicadores de qualidade em restaurantes. Nesse estudo foi detectada a exigência dos clientes, para com os serviços prestados, incluindo resultados a partir da satisfação dos frequentadores do restaurante. As avaliações dos clientes foram reavaliadas utilizando o método Kano, cuja abordagem é qualitativa. Esse método procura conhecer a relação entre a avaliação subjetiva do produto, que é expressa pelo nível de satisfação do cliente (satisfeito ou insatisfeito) e avaliação objetiva, que é expressa pelo nível de incorporação física do item de qualidade (suficiente ou insuficiente) (GARCEZ, 2000). Dessa forma serão disponibilizados para o usuário avaliar um item as seguintes opções: Insuficiente, Ruim, Regular, Bom e Ótimo, que medem o nível de satisfação do cliente em relação a um produto ou serviço oferecido.

Segundo Faria (2014), a “Análise de Kano” ou “Diagrama de Kano” é um método para o desenvolvimento ou melhoria de produtos baseado na caracterização das necessidades do cliente, sejam elas verbalizadas ou não. Dessa forma percebe-se que esse diagrama foi utilizado para entender as necessidades do cliente, se um produto ou serviço é insuficiente significa que não supre as necessidades dos clientes, regular atende minimamente, bom parcialmente e ótimo totalmente.

4. RESULTADOS OTIDOS

Neste capítulo estão apresentados os resultados obtidos. A partir do conhecimento adquirido pelas API foi possível desenvolver o sistema de localização e avaliação de restaurantes. Esse conhecimento foi realizado através de materiais disponibilizados pelo Google e aprendizados práticos para visualizar o seu funcionamento. Logo em seguida foi

projetado e desenvolvido o sistema com as visualizações dos gráficos e a geolocalização dos restaurantes no mapa.

Para que fosse possível iniciar a modelagem do sistema, decidiu-se realizar um questionário aos restaurantes para que fosse possível conhecer a opinião de usuários, que de certa forma serão beneficiados com um sistema desse tipo. O questionário foi realizado em 10 estabelecimentos dos quais as respostas foram as seguintes: 100% dos entrevistados responderam que disponibilizando a localização do seu estabelecimento em um sistema de localização aumentaria a procura de clientes. 100% gostariam que seu restaurante fosse indicado em um mapa para a localização, acham importante disponibilizar aos clientes uma forma de avaliação que tenha resultados rápidos e objetivos, acreditam que seria importante obter os resultados de forma rápida e objetiva, aceitariam que o seu estabelecimento fosse cadastrado em um sistema desse tipo.

Também foi questionado quais os itens seriam importantes na avaliação do cliente, esses resultados foram utilizados para a construção dos itens a serem avaliados nesse sistema. Ainda 100% dos entrevistados acham que os clientes estão mais exigentes, e o que motiva na escolha do estabelecimento predominou as respostas: qualidade dos serviços oferecidos e bom atendimento. O questionário pode ser visualizado no Anexo I.

4.1 REQUISITOS FUNCIONAIS

RF-01 Efetuar *login*: O usuário do sistema que for avaliar um restaurante deverá ser identificado através de um *login* e a senha para que ele possa ter acesso às funcionalidades oferecidas. Bem como o usuário restaurante para ter acesso aos resultados das avaliações do sistema.

RF-02 Efetuar *logout*: O Usuário, devidamente logado, finaliza sua sessão no sistema, evitando que outros tenham acesso a suas avaliações e funções realizadas no sistema.

RF-03 Gerenciar Especialidade: O administrador do sistema fará a inclusão, consulta e alteração de dados das especialidades dos restaurantes, cadastrando a descrição da especialidade.

RF-04 Gerenciar Gerentes: O administrador do sistema fará a inclusão, consulta e alteração de dados dos gerentes dos restaurantes cadastrados, os dados que deverão ser cadastrados serão os seguintes: Nome, Email, Telefone e o restaurante que pertence.

RF-05 Gerenciar Cardápio do Dia: O usuário restaurante fará a inclusão, consulta e alteração de dados do cardápio do dia do seu restaurante cadastrado, com os seguintes dados cadastrados: Nome, composição, valor e dia da semana.

RF-06 Gerenciar Restaurantes: O administrador do sistema e o usuário restaurante fará a inclusão, consulta e alteração de dados dos restaurantes cadastrados, com os seguintes dados: Nome, Endereço, Telefone, Especialidade, Gerente e Cardápio do dia.

RF-07 Gerenciar Itens Avaliados: O sistema deverá permitir a inclusão, alteração e consulta dos itens que poderão ser avaliados pelo usuário, informando a descrição do item.

RF-08 Gerenciar Avaliação: O sistema deverá permitir a inclusão, consulta e das avaliações dos restaurantes, com os seguintes dados cadastrados: Restaurante, Item Avaliado, Usuário e Data.

4.2 REQUISITOS NÃO FUNCIONAIS

RNF-01: O sistema deve possuir senhas de acesso e identificação para diferentes tipos de usuários: administrador do sistema, usuário cliente e usuário restaurante.

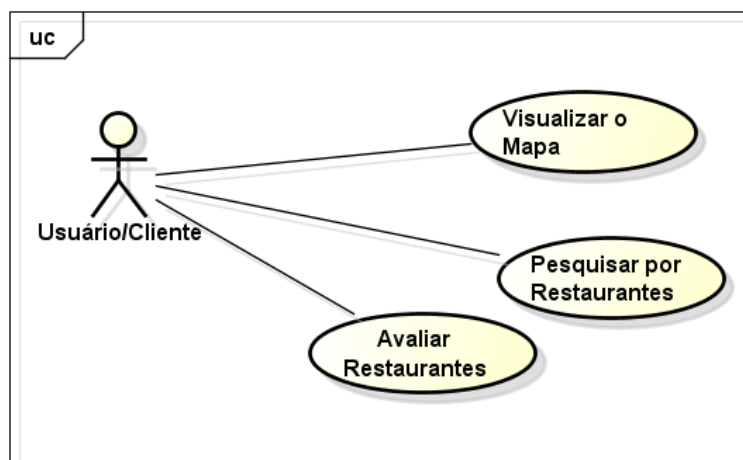
RNF-02: O sistema deve garantir que cada usuário realize apenas uma avaliação por restaurante, durante um tempo determinado, permitindo uma nova avaliação após seu término.

RNF-03: O sistema deve permitir uma interface amigável, ou seja, uma interface que permita o acesso a partir de vários dispositivos, sem distorções de imagens.

4.3 DIAGRAMAS DE CASO DE USO

Nesse diagrama podemos visualizar como serão demonstradas as ações que um usuário cliente terá ao acessar o sistema, como mostra a Figura 21 e sua documentação no Anexo II.

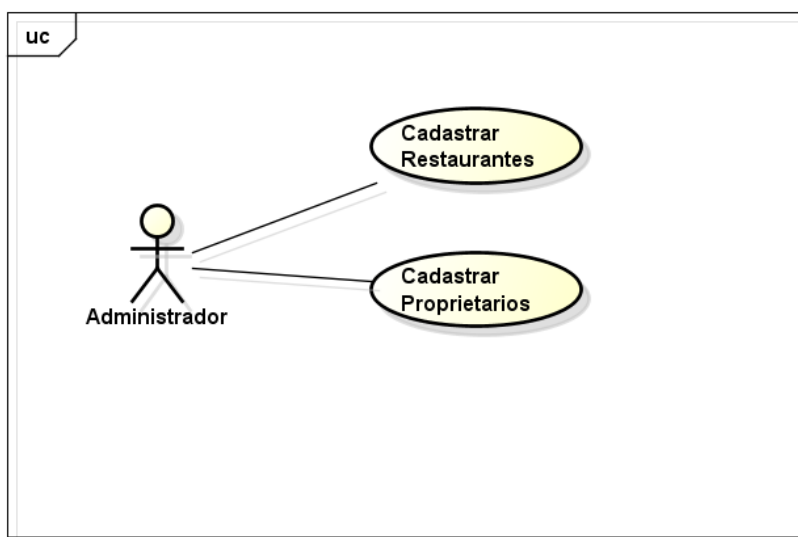
Figura 21 – Caso de Uso Usuário Cliente



Fonte: Do Autor

Na Figura 22 encontra-se o diagrama de caso de uso com as ações do usuário administrador, sua documentação encontra-se no anexo III.

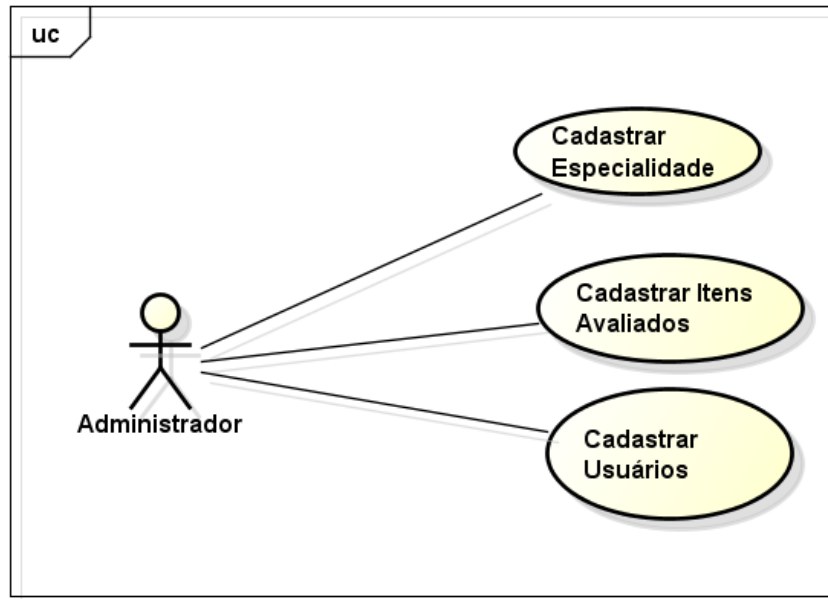
Figura 22 – Caso de Uso Ações do Administrador



Fonte: Do Autor

Na Figura 23, podemos visualizar o diagrama de caso de uso que demonstra os cadastros de Especialidade, Usuários, Itens Avaliados e Cardápio do Dia. Sua documentação pode ser visualizada no Anexo IV.

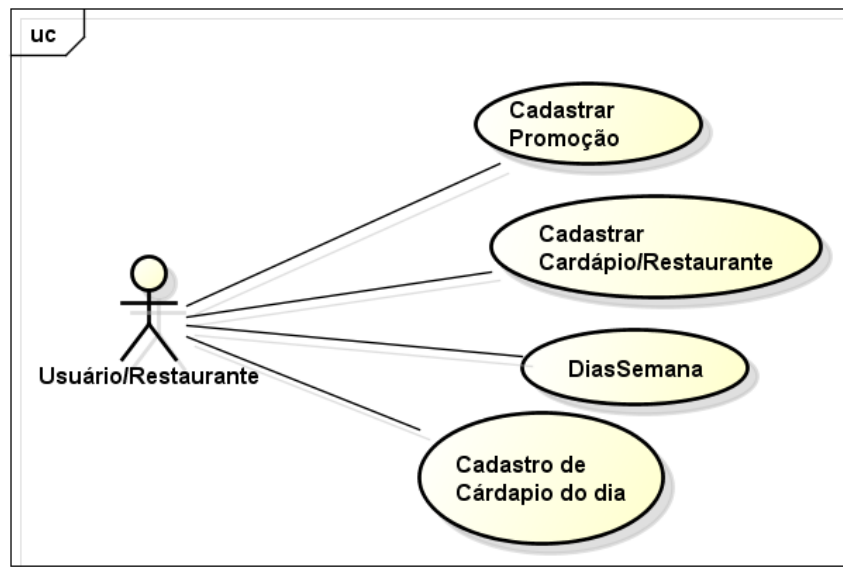
Figura 23 – Caso de Uso Cadastros



Fonte: Autor

Na Figura 24, podemos visualizar o diagrama de caso de uso que demonstra os cadastros de Promoções e Cardápio/Restaurante. Sua documentação pode ser visualizada no Anexo V.

Figura 24 - Caso de Uso Usuário Restaurante

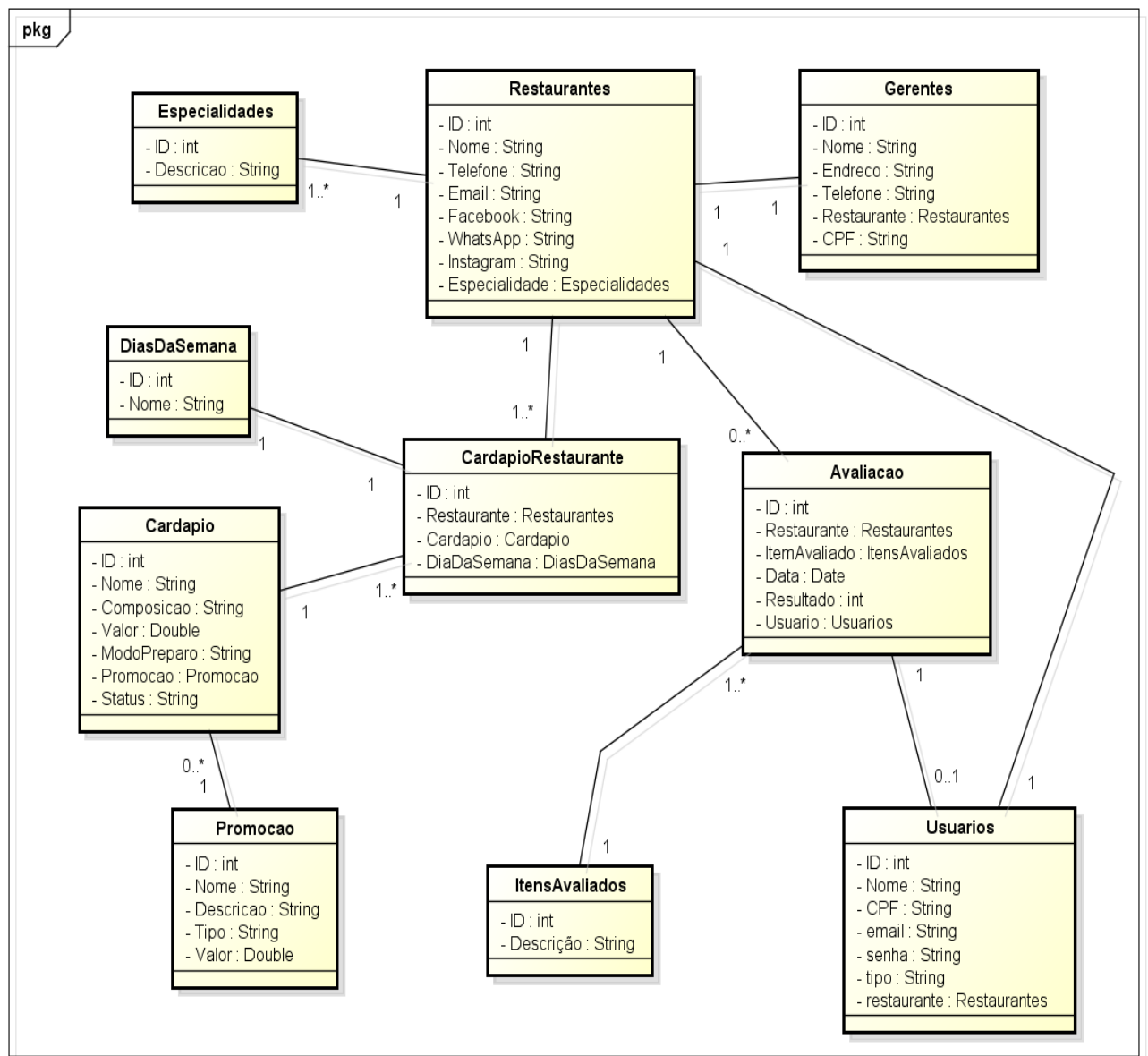


Fonte: Do Autor

4.4 Diagrama de Classes

Na Figura 25 o diagrama de classes informa quais as classes foram criadas no desenvolvimento do sistema. Nele encontra-se a classe Restaurante que armazena os dados dos restaurantes cadastrados com seus atributos e métodos, a classe de Proprietários que possui os dados com informações dos proprietários dos restaurantes. A classe Especialidades que possui as especialidades, na classe usuários será informado dados de acesso para que os usuários possam avaliar o sistema. A classe Avaliação recebe os dados de avaliação dos clientes de acordo com cada restaurante. Com a classe Cardápio do Dia será possível cadastrar o cardápio do dia de cada restaurante para cada dia da semana.

Figura 25 – Diagrama de Classes



Fonte: Do Autor

4.5 Arquitetura do Sistema

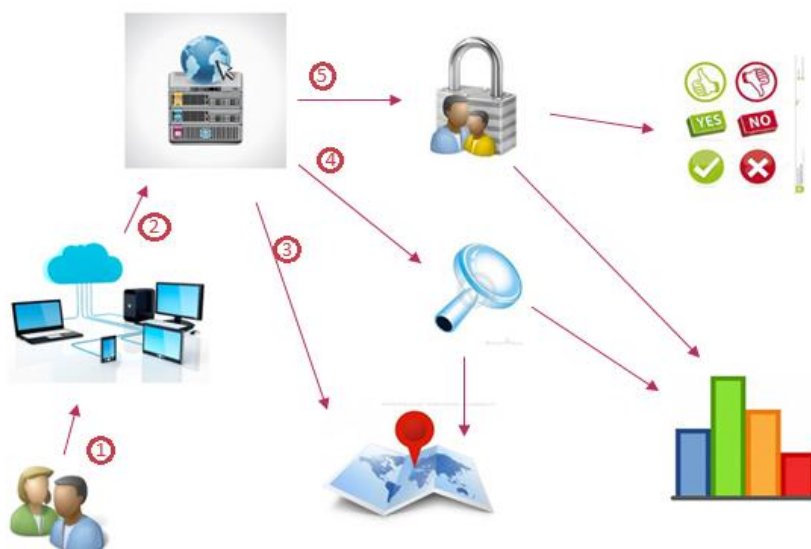
O sistema funcionará da seguinte maneira, o usuário poderá acessá-lo através de qualquer dispositivo, sendo adaptado através da utilização do *framework* Bootstrap a exibição de *layout* responsivo. Logo no acesso do sistema o usuário encontrará o mapa com os restaurantes cadastrados, o que permite fácil localização para os usuários.

Após seu acesso, os usuários poderão encontrar as informações dos estabelecimentos, como o cardápio do dia, avaliação feita por outros usuários. Para que o usuário faça sua avaliação, é necessário fazer um cadastro com *login* e senha, possibilitando um controle de avaliações por usuário. O proprietário possui um acesso diferenciado, com um cadastro no sistema com *login* e senha e o usuário tipo restaurante ele pode visualizar os dados das avaliações do seu restaurante, podendo encontrar os resultados dessas avaliações realizadas por frequentadores do seu estabelecimento, reunindo informações para alcançar a melhoria necessária que agrade seus clientes.

4.6 PADRÕES DO SISTEMA

Nesta sessão serão apresentados os padrões do sistema. Começando com o funcionamento do sistema que possui duas formas de acesso: usuário cliente e usuário restaurante. Na Figura 26, tem-se uma demonstração de acesso do usuário cliente.

Figura 26 – Acesso Usuário Cliente



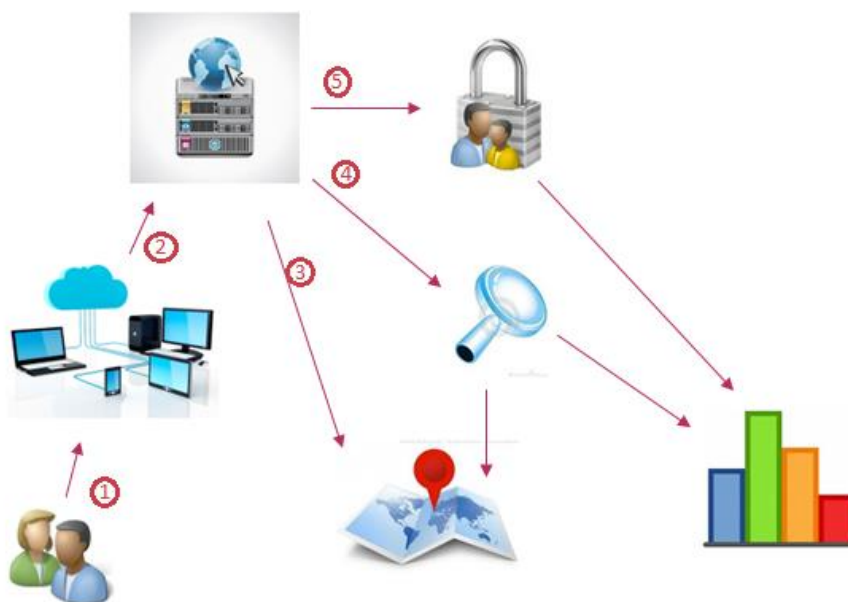
Fonte: Do Autor

A seguir a descrição das etapas do acesso do usuário cliente demonstradas na figura anterior.

1. O usuário cliente que irá acessar o sistema.
2. O usuário poderá acessar a partir de qualquer dispositivo como: computador, smartphone, tablete.
3. Esse acesso passa pelo servidor que encaminha o usuário para a tela inicial onde demonstra os restaurantes cadastrados já referenciados no mapa.
4. Assim que o usuário estiver acessado ele poderá realizar uma busca por restaurantes, onde ele escolhe por tipo como: churrascaria, pizzaria. O usuário também encontra um *link* para visualizar os resultados geral das avaliações de um determinado restaurante.
5. Para que o usuário possa avaliar um restaurante é necessário estar *logado* no sistema, caso não seja cadastrado deve realizar seu cadastro para que assim seja possível avaliar um determinado restaurante.

Na Figura 27 encontra-se o acesso do usuário restaurante.

Figura 27 – Acesso Usuário Restaurante



Fonte: Do Autor

A seguir a descrição das etapas de acesso do usuário restaurante:

1. Usuário restaurante que irá acessar o sistema.
2. Poderá acessar de qualquer dispositivo como: smartphone, tablet, computador.
3. Assim que acessar chegará a página inicial que demonstra o mapa com os restaurantes cadastrados já geolocalizados no mapa.
4. Pode pesquisar por tipo de restaurantes como: churrascaria, pizzaria, tendo os mesmos acessos do usuário cliente, como também visualizar os resultados gerais das avaliações de um determinado restaurante.
5. Para que o usuário tenha um acesso diferenciado dos resultados do seu restaurante é necessário fazer login.

Na

Figura 28 encontra-se o padrão de campos editáveis, esse padrão é utilizado para todos os campos editáveis do sistema. Nesse mesmo padrão estão as validações de campos, não permitindo que o usuário salve dados sem as informações necessárias.

Figura 28 – Campos Editáveis

Nome

Fonte: Do Autor

Campos de CPF também possuem um padrão como o demonstrado na Figura 29. Esse campo possui uma máscara para que o usuário não se preocupe ao digitar os números.

Figura 29 – Campo de CPF

CPF

Fonte: Do Autor

O sistema também possui um padrão para campos de senha no qual todos aparecem da mesma forma ao digitar a senha tanto para criá-la quanto para acessar o sistema. A Figura 30 apresenta um campo de senha.

Figura 30 – Campos de Senha

Senha

Fonte: Do Autor

Outro padrão adotado nesse sistema é o campo de data que possui uma máscara e ao mesmo tempo exibe ao usuário uma caixa de seleção para que o cliente informe a data de forma mais prática. Na Figura 31 um exemplo desse padrão.

Figura 31 - Campo de data

Data de Nascimento

dd/mm/aaaa

novembro de 2015

dom	seg	ter	qua	qui	sex	sáb
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5

Fonte: Do Autor

Os campos de seleção também são padrões do sistema como pode ser visualizado na Figura 32.

Figura 32 – Campos de seleção

Especialidade *

- Buffet (Livre/Kilo)
- Churrascaria
- Galeria
- Lanches
- Pizzaria
- Sushi/Temakeria
- Sorveteria
- Restaurante Mexicano
- Petisqueria/Pizzaria

Fonte: Do Autor

O sistema também possui um padrão para campo de telefones, como pode ser visualizado na Figura 33.

Figura 33 – Campo de Telefone

Telefone 2

Fonte: Do Autor

Para demonstrar a validação de campos, pode-se visualizar a mensagem a seguir na Figura 34 quando o CPF informado é inválido.

Figura 34 – Validação de CPF

CPF Invalido!

OK

Fonte: Do Autor

O sistema também possui um padrão para o campo data como pode ser visualizado na Figura 35.

Figura 35 – Validação de E-mail









Email *

Endereço de email inválido.

Fonte: Do Autor

A forma de listagem dos dados também é a mesma para todos os dados cadastrados, as telas de listagem possibilitam ao usuário inserir um novo registro, alterar e excluir um registro existente. Sendo assim pode-se visualizá-la na Figura 36.

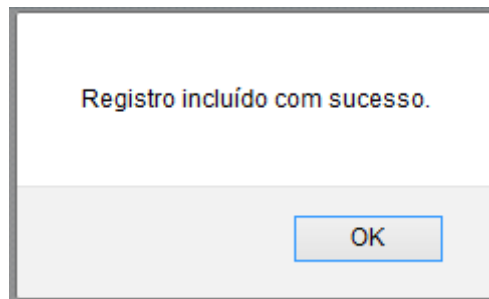
Figura 36 – Listagem de Dados

+ Novo Registro			
Nome	CPF	Email	Operações
Administrador	321.321.432-45	admin@gmail.com	 
Ana	694.658.748-09	ana@gmail.com	 
Ana Maria	86349792297	anama@gmail.com	 
Andressa Ferreira	818.007.251-75	andressa.defe@gmail.com	 

Fonte: Do Autor

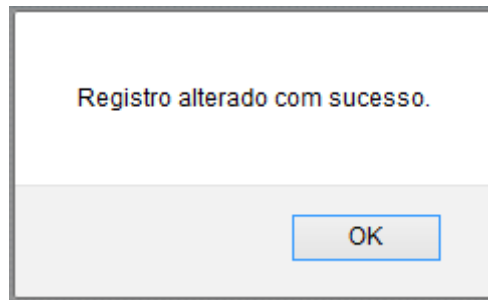
Ao inserir dados no sistema, caso os dados estejam corretos, o mesmo demonstra uma mensagem de confirmação como a demonstrada na Figura 37.

Figura 37 – Confirmação de Inclusão



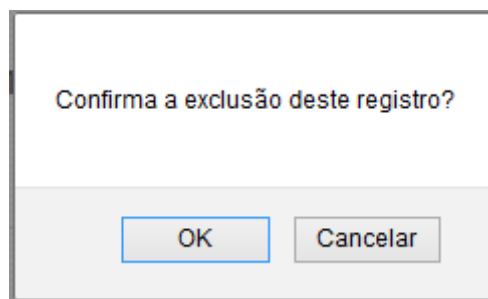
Fonte: Do Autor

Da mesma maneira o sistema informa o usuário quando algum dado foi alterado, informando ao usuário uma tela com a mensagem de confirmação de alteração dos dados, como mostra a Figura 38.

Figura 38 – Confirmação de Alteração

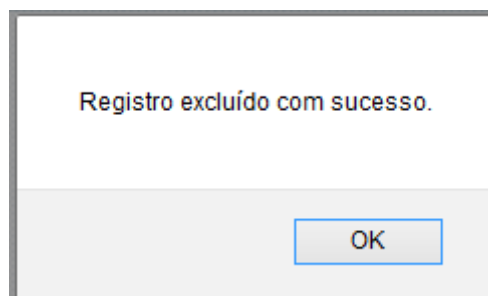
Fonte: Do Autor

Caso o usuário queira excluir um registro, o sistema informa uma mensagem para que ele confirme a exclusão, para que ele tenha certeza de qual registro deve ser excluído. Na Figura 39 encontra-se a mensagem que questiona a confirmação da exclusão.

Figura 39 – Pergunta Exclusão de Registro

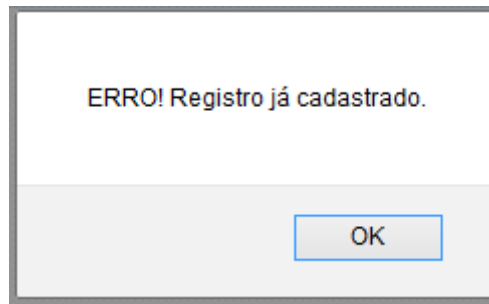
Fonte: Do Autor

Após o usuário confirmar a exclusão o sistema informa a mensagem de conformação da exclusão dos registros como mostra a Figura 40.

Figura 40 – Confirmação de Exclusão

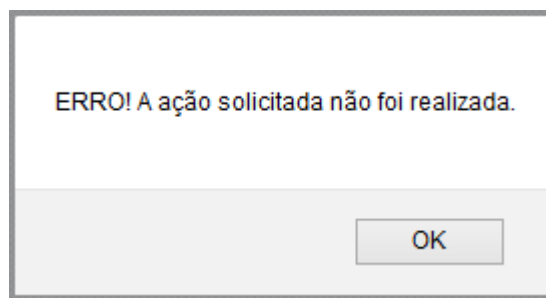
Fonte: Do Autor

Caso o usuário tentar inserir um registro já cadastrado será demonstrada a mensagem que está na Figura 41.

Figura 41 – Registro já cadastrado

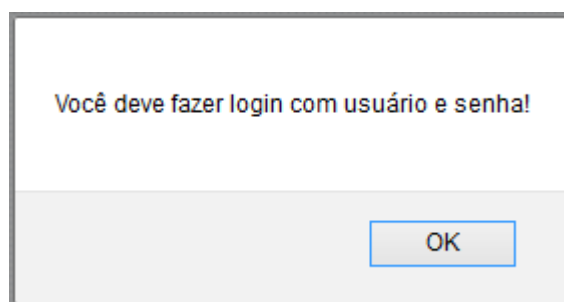
Fonte: Do Autor

Caso não seja realizada alguma ação, o sistema informa o usuário com uma mensagem de erro informando que a ação não foi realizada como pode ser visto na Figura 42.

Figura 42 – Solicitação não Realizada

Fonte: Do Autor

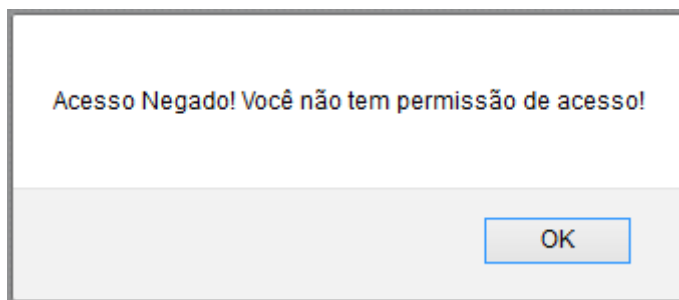
Se o usuário tentar realizar alguma ação da qual necessita estar logado, o sistema informará que o mesmo necessita fazer *login* para que tenha acesso a essa função. Na Figura 43 uma visualização da mensagem de informação ao usuário.

Figura 43 – Mensagem Para Fazer Login

Fonte: Do Autor

Com o usuário já logado no sistema ele tem locais o qual não possui permissão de acesso, sendo assim o sistema informa uma mensagem para que o usuário saiba que não tem essa permissão. Na Figura 44 a mensagem de permissão de acesso.

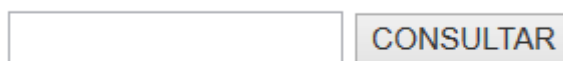
Figura 44 – Acesso Negado



Fonte: Do Autor

O sistema possui um padrão de consulta dos dados, que podem ser pesquisados para encontrar os dados cadastrados no sistema como pode ser visualizado na Figura 45.

Figura 45 – Consulta de Dados



Fonte: Do Autor

Os botões do sistema podem ser visualizados a seguir na Figura 46. Esses botões estão em todos os formulários, o botão salvar irá salvar os dados no banco, limpar limpa os campos do formulário para iniciar novamente sem salvar e o cancelar volta para a página inicial e não salva nenhuma alteração ou inserção.

Figura 46 – Botões do Sistema



Fonte: Do Autor

Para que o usuário possa ter uma melhor visualização de todos os dados em uma listagem, o sistema possui um padrão de paginação e total de registros para que o usuário possa ter uma compreensão melhor sobre as informações cadastradas e possa encontrar os dados mais facilmente. Na Figura 47 uma demonstração da paginação do sistema e o total de registros cadastrados.

Figura 47 – Paginação

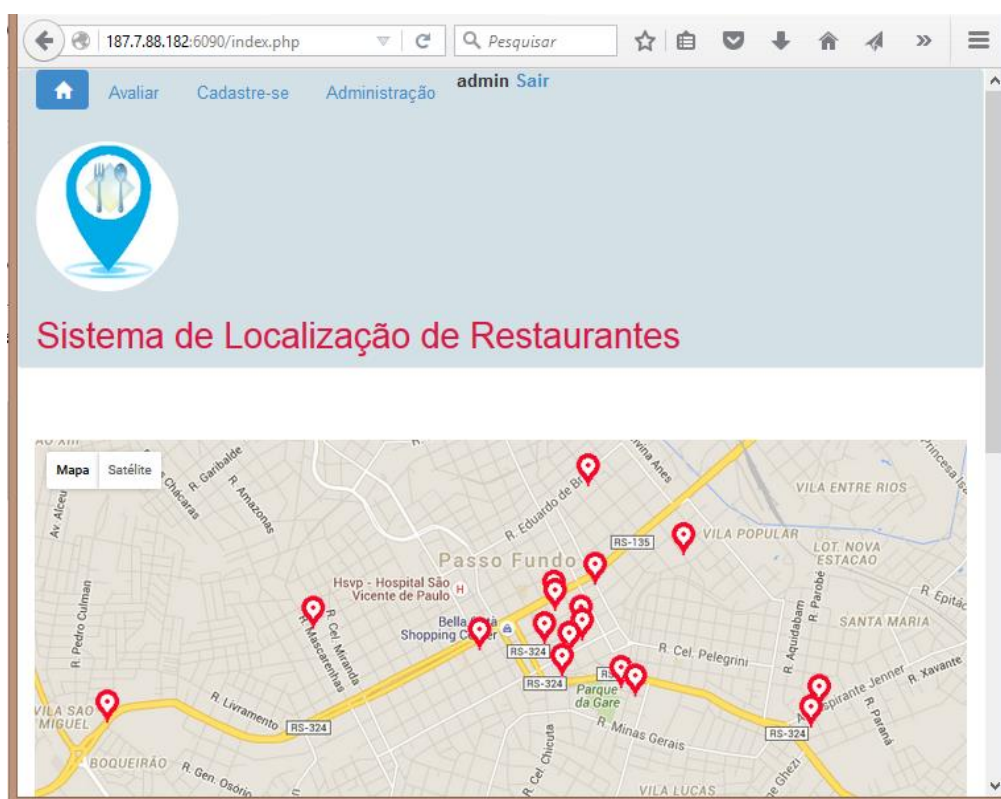
Total de Registros: 13
Páginas: [1] [2] [3] [4]

Fonte: Do Autor

4.7 Detalhes do Sistema

O sistema foi dividido em duas partes área administrativa e área pública. Pode-se visualizar a tela inicial do sistema na Figura 48.

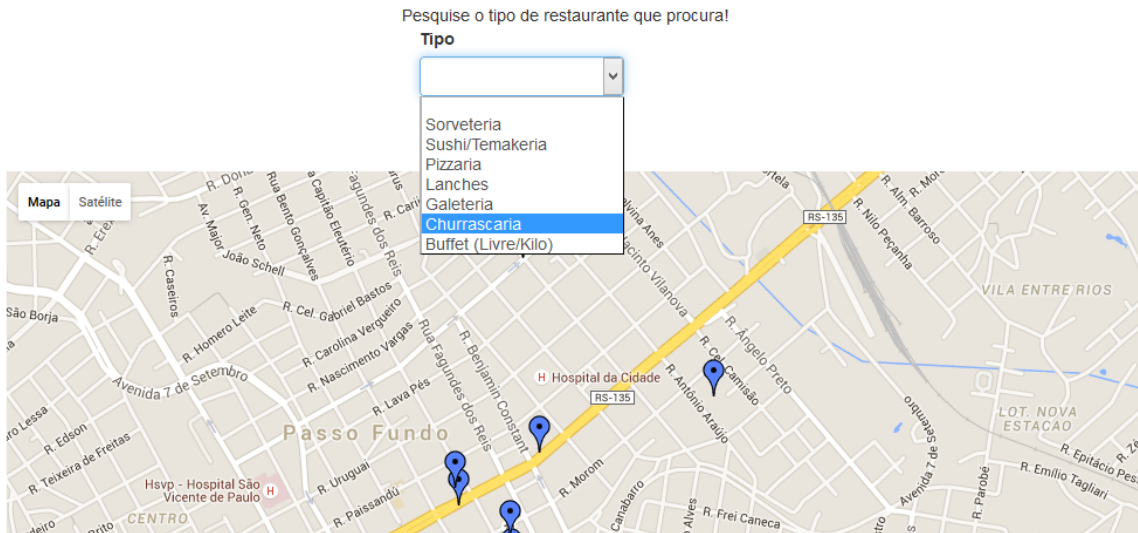
Figura 48 – Tela Inicial



Fonte: Do Autor

O usuário poderá localizar um restaurante no mapa por tipo/especialidade. O usuário seleciona uma especialidade e em resposta a essa procura o sistema carrega o mapa novamente com os restaurantes pertencentes a especialidade escolhida. Essa funcionalidade pode ser visualizada na Figura 49.

Figura 49 – Pesquisa por Restaurantes



Fonte: Do Autor.

Para que seja necessário um acesso ao sistema caso o usuário queira fazer uma alteração em algum cadastro ou funções do sistema que tenham acesso restrito, será necessário realizar um *login* de acesso que pode ser visualizado na Figura 50.

Figura 50 – Tela Login

Login

Insira seu login

Senha

Insira sua senha

Fonte: Do Autor

Um exemplo de formulários disponíveis no sistema pode ser visualizado como mostra a Figura 51, esse é formulário é disponibilizado para usuários clientes realizarem seu cadastro no sistema para que possam avaliar os restaurantes.

Figura 51 – Cadastro de Usuários Cliente

Cadastro de Usuários

* Campos Obrigatórios!

Nome *	Data de Nascimento *
<input type="text" value="Nome Completo"/>	<input type="text"/>
Login *	Senha *
<input type="text" value="Mínimo 3 caracteres"/>	<input type="text" value="mínimo 5 caracteres"/>
CPF *	Email *
<input type="text" value="Apenas números"/>	<input type="text" value="exemplo@gmail.com"/>

Fonte: Do Autor

O acesso restrito dos usuários restaurantes pode ser visualizado na Figura 52. Nesse acesso o usuário encontra o link para os resultados de avaliações e o menu com as funções restritas.

Figura 52 – Administração dos Proprietários

Cardápio Cardápio Restaurante Gerentes Promoção Restaurantes Usuarios ana Sair

Sistema de Localização de Restaurantes

Escolha sua forma de visualização de resultados

- Resultado geral por idade
- Resultado geral das avaliações
- Resultado por item avaliado

Fonte: Do Autor

Um diferencial do sistema é o formulário de avaliação do sistema, esse formulário é o principal para as avaliações onde os usuários irão escolher entre os itens disponibilizados, quais eles pretendem avaliar e como eles avaliam este item. Na Figura 53 o formulário de avaliação.

Figura 53 – Formulário de Avaliação

Usuário	Data
<input type="text" value="maara"/>	<input type="text" value="2015-12-01"/>
Restaurante *	Observação
<input type="text"/>	<input type="text"/>

Itens a serem Avaliados

Cardápio

- Insuficiente
- Ruim
- Regular
- Bom
- Ótimo

Atendimento

- Insuficiente
- Ruim
- Regular
- Bom
- Ótimo

Comida

- Insuficiente
- Ruim
- Regular
- Bom
- Ótimo

Estacionamento

- Insuficiente
- Ruim
- Regular
- Bom
- Ótimo

Preço

- Insuficiente
- Ruim
- Regular
- Bom
- Ótimo

Ambiente Físico

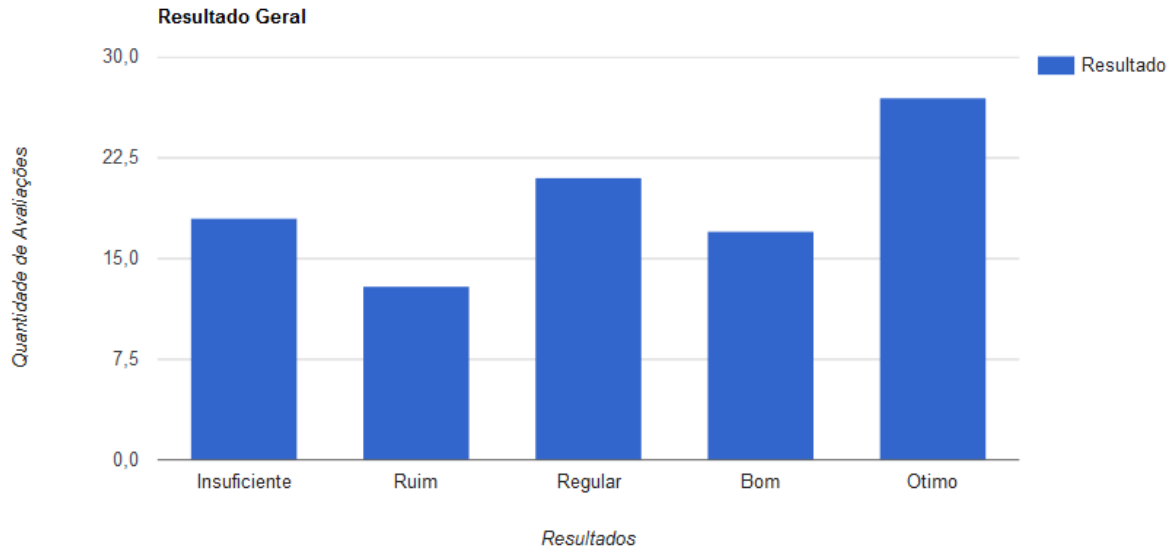
- Insuficiente
- Ruim
- Regular
- Bom
- Ótimo

Fonte: Do Autor

O sistema possui a visualização dos dados em gráficos, são três tipos de gráficos que demonstram diferentes formas de resultados. O gráfico que demonstra o resultado geral de seus restaurantes, é uma contagem de todas as avaliações que este restaurante possui. Esse gráfico pode ser visualizado por usuários clientes e usuários restaurantes. O cliente para visualiza-lo necessita informar qual o restaurante que deseja ver os resultados, e o usuário restaurante não necessita informar seu restaurante, pois essa informação é adquirida no

momento que ele realiza *login* e senha, gravando essa informação da sessão que está carregada. Na Figura 54 o gráfico de resultado geral.

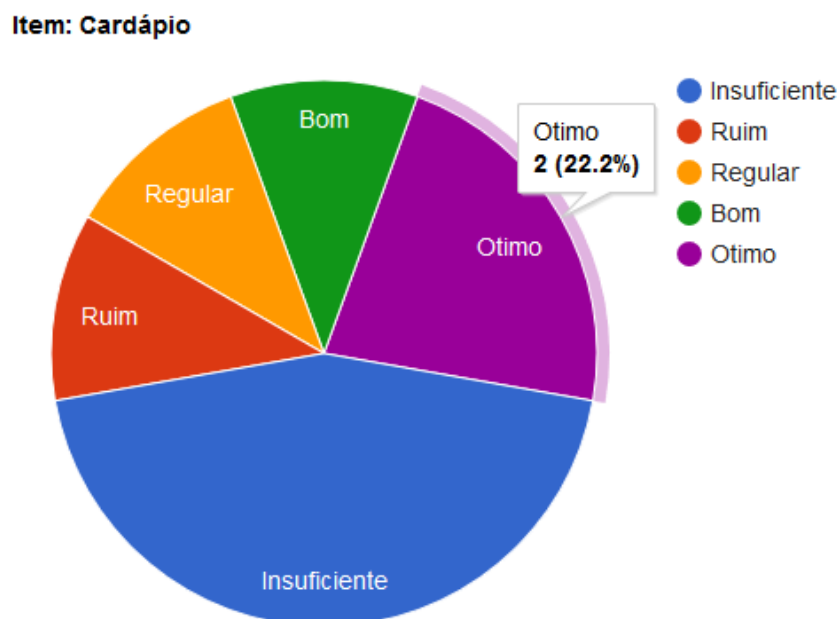
Figura 54 – Resultado Geral do Restaurante



Fonte: Do Autor

O próximo gráfico é o de resultado por item. Esse gráfico só pode ser visualizado por usuários restaurante e somente os resultados de seu estabelecimento em específico não tendo acesso aos resultados de outros restaurantes, para gerar o gráfico é necessário informar o item que deseja conhecer os resultados. Na Figura 55 o gráfico de resultado por item.

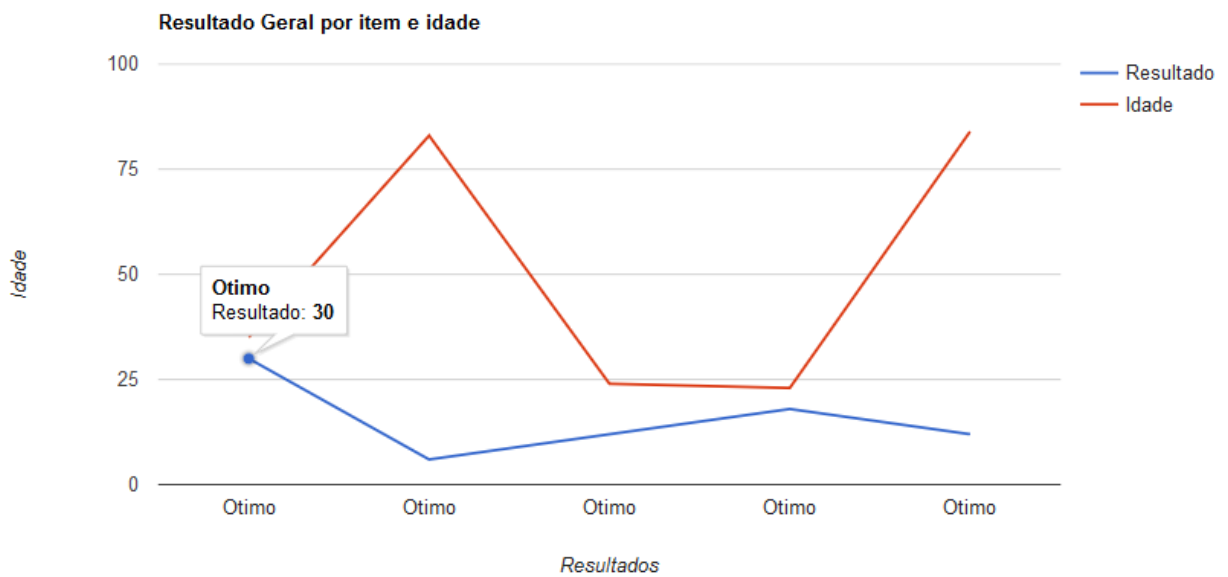
Figura 55 – Resultado por Item



Fonte: Do Autor

E o último gráfico traz a informação da faixa etária dos clientes que avaliaram o seu restaurante. Para gerar o gráfico deve-se informar a avaliação que foi dada pelo item como ótimo, bom, e informar qual o item que deseja ver o resultado como por exemplo cardápio. Esse gráfico também é visualizado somente pelo usuário restaurante. Na Figura 56 o resultado que demonstra a faixa etária de seus clientes.

Figura 56 – Resultado por Item e idade



Fonte: Do Autor

Os testes realizados foram testes funcionais, onde foram analisados o comportamento das API ao gerar o mapa e os gráficos. Desses testes foram possíveis reconhecer a eficácia dessas tecnologias pois conforme são inseridos dados como cadastro de restaurante, uma nova avaliação, o mapa e os gráficos já atualizam suas informações para o usuário final sem um intermédio do desenvolvedor.

5. CONSIDERAÇÕES FINAIS

A partir da construção do projeto foi identificado a importância do aprendizado contínuo, pois essas tecnologias disponibilizadas ainda são pouco utilizadas e poucos conhecem a sua importância e praticidade no desenvolvimento. Com isso, percebe-se que, com a utilização dessas tecnologias se adquire mais comodidade e tempo durante o desenvolvimento de um sistema.

Importante salientar que ao conhecer a API Google Earth, observou-se que esta agiliza o processo de desenvolvimento de geração de mapas e a geolocalização de restaurantes no sistema, pois a API disponibiliza recursos e funcionalidades através de seu *plug-in*. Além disso, um aspecto importante é que seu uso diminui a reescrita de código, facilitando assim a inserção de dados no sistema e a disponibilização dos dados no mapa.

Da mesma forma, é válido destacar que a API Google Chart que permite formar gráficos dinâmicos de forma prática, ao disponibilizar seus recursos e funcionalidades através de seu *plug-in*, permite a desenvolvedores conteúdos leves e rápidos através de sua biblioteca básica JavaScript. Contudo, o aspecto mais importante na sua utilização é a disponibilidade da galeria de gráficos, permitindo que o desenvolvedor possa escolher qual o gráfico que bem representará os dados do seu sistema.

Ao realizar os testes no sistema foi possível analisar o comportamento das API utilizadas no desenvolvimento, ao gerar o mapa com os restaurantes localizados foi possível perceber a praticidade na sua visualização, pois assim tornou-se mais simples a localização dos restaurantes para o usuário. Com isso foram realizados testes funcionais do sistema, com dados fictícios para a análise do comportamento das API.

Da mesma maneira ao gerar os gráficos, percebeu-se que são visualizados sem distorção de imagens por qualquer dispositivo, com fácil monitoramento dos dados para gerá-los, bem como, as opções de visualização, como legendas, tamanho e outras funcionalidades disponibilizadas pela API Google Chart.

Com essas informações adquiridas durante o desenvolvimento e a fase de testes, identificou-se que as tecnologias estudadas auxiliaram para que os objetivos desse projeto fossem alcançados, porém uma dificuldade encontrada no desenvolvimento do sistema foi a renderização dos gráficos para smartphones. Outra dificuldade foi ao criar o arquivo KML em memória cache, pois, a mesma demora para atualizar, isso foi constatado como um problema no

servidor que deve ser analisado, para que a atualização seja instantânea, essa melhoria fica para estudos futuros.

Como projetos futuros, há o aprimoramento do sistema, aumentando as funcionalidades como uma parte mais dedicada a usuários externos que fazem as avaliações, realizar testes diretamente com usuários final do sistema para que possam repassar suas opiniões sobre sistemas desse tipo. Ainda como trabalhos futuros estudar as funcionalidades da API Google Earth com a API Google Maps, como também a utilização dessas tecnologias para sistemas de diferentes segmentos como localização e avaliação de outros tipos de estabelecimentos como: casas noturnas, comércio em geral, supermercados, entre outros.

REFERÊNCIAS

BRASIL, Portal. **Alimentação em bares e restaurantes cresce no Brasil**. Disponível em: <<http://www.brasil.gov.br/turismo/2014/06/alimentacao-em-bares-e-restaurantes-cresce-no-brasil>> Acesso em: nov. 2015

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML Guia do Usuário**. p. 521. Elsevier Editora Ltda. 2 ed. São Paulo 2012.

CERVO, Amado L., BREVIAN, Pedro A., SILVA, Roberto da. **Metodologia científica**. p. 162. Editora Pearson Education. 6 ed. São Paulo 2007.

CHARTS, Google. **Google Charts**. Disponível em: <<https://developers.google.com/chart/interactive/docs/index>> Acesso em: set. 2014

CODE, Google Developers. **HowToAddNewIcons**. Disponível em: <<https://code.google.com/p/google-maps-icons/wiki/HowToAddNewIcons>> Acesso em: nov. 2014

CONVERSE, Tim, PARK, Joyce. **PHP a Bíblia, A história do PHP**. p. 868. Editora Elsevier. 2 ed. Rio de Janeiro 2003

DEVMEDIA. **Entendo e usando o SVG**. Disponível em: <<http://www.devmedia.com.br/entendendo-e-usando-o-svg/19773>> Acesso em: nov. 2014.

EARTH , Google Developers. **Google Earth API**. Disponível em: <<https://developers.google.com/earth/?hl=pt-br>> Acesso em: set. 2014.

FARIA, Caroline. **Diagrama de Kano**. Disponível em: <http://www.infoescola.com/administracao_/diagrama-de-kano/> Acesso em: nov. 2014.

FRANCISCO. Wagner de Cerqueira e. **Mapas**. Disponível em:

<<http://www.mundoeducacao.com/geografia/mapas.htm>> Acesso em: out. 2014.

FOURSQUARE. **Foursquare**. Disponível em: <<https://pt.foursquare.com/about>> Acesso em: set. 2015.

GARCEZ, Eliane M. S.; FACHIN, Gleisy R. B.; JÚNIOR, Pedro P. A. **Indicadores de Qualidade em Restaurantes: Um Estudo de Caso**. Revista de Ciência da Administração UFSC. Universidade Federal de Santa Catarina, Florianópolis. ano 2, n. 3, abr. 2000. Disponível em: <<https://periodicos.ufsc.br/index.php/adm/issue/view/233>> Acesso em: nov. 2014.

GOOGLE. **O que é o Google Earth**. Disponível em: <<https://support.google.com/earth/answer/176145?hl=pt-BR>> Acesso em: out. 2014.

IMASTERS. **Estatísticas de uso do PHP**. Disponível em: <<http://imasters.com.br/artigo/20396/php/estatisticas-de-uso-do-php>> Acesso em: nov. 2015.

KEKANTO. **Kekanto Mobile**. Disponível em: <<https://kekanto.com.br/>> Acesso em: abril de 2015.

KML, Google Developers. **Tutorial do KML**. Disponível em: <https://developers.google.com/kml/documentation/kml_tut?hl=pt-br> Acesso em: nov. 2014.

NASCIMENTO, Thiago. **Desenvolvendo com Bootstrap 3: um framework front-end que vale a pena**. Disponível em: <<http://thiagonasc.com/desenvolvimento-web/desenvolvendo-com-bootstrap-3-um-framework-front-end-que-vale-a-pena>> Acesso em: out. 2014.

PEREIRA NETO, Álvaro. **PostgreSQL Técnicas Avançadas**. p. 286. Editora Érica Ltda. 4 ed.. São Paulo 2007.

OFFICE. **Glossário do FrontPage**. Disponível em:

<<http://office.microsoft.com/pt-br/frontpage-help/glossario-do-frontpage-HP001038137.aspx>> Acesso em: nov. 2014.

OLHAR DIRETO. **Mais da metade da população faz refeições fora de casa segundo pesquisa da GFK**. Disponível em:

<<http://www.olhardireto.com.br/noticias/exibir.asp?id=75788>> Acesso em: set. 2014.

PHP HISTÓRIA. **História do PHP**. Disponível em:

<http://php.net/manual/pt_BR/history.php.php> Acesso em: out. 2014.

PHP. **O que é PHP**. Disponível em: <http://php.net/manual/pt_BR/intro-whatism.php> Acesso em: out. 2014.

POSTGRESQL. **Sobre PostgreSQL**. Disponível em:

<<http://www.postgresql.org.br/old/sobre>> Acesso em: out. 2014.

PRADO, Fabio. **Qual é o Melhor Banco de Dados: Oracle ou SQL Server?** Disponível em:

<<http://www.profissionaisti.com.br/2012/01/qual-e-o-melhor-banco-de-dados-oracle-ou-sql-server/>> 2012. Acesso em: nov. 2015.

PREVIDELLI, Amanda. **5 Apps Para Ajudar na Escolha de um Restaurante**. Disponível em: <<http://info.abril.com.br/noticias/blogs/download-da-hora/iphone/5-apps-para-ajudar-na-escolha-de-um-restaurante/>> Acesso em: nov. 2014.

SISTEMA, Kekanto. **Kekanto Passo Fundo**. Disponível em:

<<https://kekanto.com.br/rs/passos-fundo>> Acesso em: nov. 2015.

SILVA, Marcos Noé Pedro da. **Tipos de Gráficos**. Disponível em:

< <http://www.mundoeducacao.com/matematica/tipos-graficos-1.htm>> Acesso em: out. 2014.

SOARES, Wolace. **PHP 5 Conceitos, Programação e Integração com Banco de Dados**. p. 528. Editora Érica Ltda. 6 ed., São Paulo 2011.

TECMUNDO. **O que é API.** Disponível em:

<<http://www.tecmundo.com.br/programacao/1807-o-que-e-api-.htm>> Acesso em: set. 2014.

UNTAPPD, **Untappd Drink Socially.** Disponível em: <<https://untappd.com/>> Acesso em:

set. 2015.

UTTERBACK, Benjamin. **O que é o Bootstrap? Verdades e Mitos: Parte 1 de 2.**

Disponível em:

<<http://www.prestashop.com/blog/pt/2014/03/06/o-que-e-o-bootstrap-verdades-e-mitos-parte-1-de-2/>> Acesso em: out. 2014.

VEJA. A nova versão do aplicativo VEJA Comer & Beber. Disponível em:

<<http://veja.abril.com.br/noticia/vida-digital/a-nova-versao-do-aplicativo-veja-comer-beber/>>

Acesso em: nov. 2015.

ANEXOS**ANEXO I – QUESTIONÁRIO AOS RESTAURANTES****QUESTIONÁRIO AOS RESTAURANTES**

- Você acha que disponibilizando a localização do seu restaurante, em um sistema de localização, aumentaria a procura de clientes pelo estabelecimento?
 - Sim
 - Não

- Você gostaria que seu restaurante fosse indicado em um mapa por um sistema de localização?
 - Sim
 - Não

- Você acha importante disponibilizar aos seus clientes uma maneira de avaliar seu restaurante?
 - Sim
 - Não

- Quais desses indicadores seriam importantes para serem avaliados pelos seus Clientes?
 - Cardápio
 - Ambiente Físico
 - Atendimento
 - Preço
 - Estacionamento
 - Outros: _____.

- Seria importante obter os resultados dessa avaliação de forma rápida e objetiva?
 - Sim
 - Não

- Você aceitaria que seu restaurante fosse cadastrado em um sistema que permita a localização e avaliação de seu restaurante?
 - Sim
 - Não

- Você acredita que os clientes estão mais exigentes?
 - Sim
 - Não

- Em sua opinião, o que motiva o cliente optar por um restaurante?
 - Qualidade dos serviços oferecidos
 - Preço
 - Bom Atendimento
 - Outros: _____.

ANEXO II – DOCUMENTAÇÃO DIAGRAMA CASO DE USO – UTILIZAÇÃO DO SISTEMA USUÁRIO

Tabela 1 – Caso de Uso Visualizar Mapa

Nome do Caso de Uso	Visualizar Mapa
Ator Principal	Usuário Cliente
Resumo	Este caso de uso refere-se ao acesso do usuário ao mapa, onde serão encontrados os restaurantes cadastrados no sistema, sem a necessidade de uma pesquisa sobre os restaurantes.
Pré-Condições	
Ações do Ator	Ações do Sistema
1. O usuário seleciona a opção para visualizar o mapa (Essa função somente será utilizada caso o usuário estiver em outro local do sistema, pois a função visualizar mapa será <i>default</i> no sistema).	
	2. O sistema carrega o mapa com todos os restaurantes cadastrados, mostrando ao usuário o seu local de origem e os restaurantes próximos.

Tabela 2 – Caso de Uso Pesquisar por Restaurante

Nome do Caso de Uso	Pesquisar por Restaurante
Ator Principal	Usuário Cliente
Resumo	Este caso de uso refere-se à pesquisa que o usuário poderá realizar, encontrando os

	restaurantes de sua escolha.
Pré-Condições	
Ações do Ator	Ações do Sistema
	1. O sistema disponibiliza um local para a busca dos restaurantes.
2. Informa sua pesquisa através da especialidade do restaurante.	
	3. Procura por Restaurantes que pertençam a especialidade informada pelo usuário.
	4. Carrega o mapa com a resposta à pesquisa do usuário.

Tabela 3 – Caso de Uso Avaliar Restaurantes

Nome do Caso de Uso	Avaliar Restaurantes
Ator Principal	Usuário Cliente
Resumo	Este caso de uso refere-se à opção que o usuário possui de avaliar os restaurantes.
Pré-Condições	Estar logado no sistema
Ações do Ator	Ações do Sistema
1. O usuário acessa o <i>link</i> para avaliar.	
	2. O sistema apresenta a tela para avaliação do restaurante com todas as informações necessárias para avaliação.
3. O usuário escolhe o restaurante e os itens que serão avaliados	
4. Realiza sua avaliação	

	5. O sistema salva a avaliação do usuário.
	6. Demonstra as avaliações realizadas por aquele usuário.

ANEXO III – DOCUMENTAÇÃO DE CASO DE USO – AÇÕES DO ADMINISTRADOR

Tabela 4 – Caso de Uso Cadastro de Restaurantes

Nome do Caso de Uso	Cadastro de Restaurantes
Ator Principal	Administrador do Sistema
Resumo	Este caso de uso refere-se aos cadastros de restaurantes que serão visualizados no mapa. Nesse cadastro encontram-se as características necessárias para a identificação do restaurante.
Pré-Condições	Estar logado no sistema
Ações do Ator	Ações do Sistema
1. O administrador acessa o sistema para fazer o cadastro de um novo restaurante.	
	2. O sistema apresenta a tela de cadastro para possibilitar a inserção de um novo restaurante.
3. Realiza o cadastro do Restaurante	
	4. O restaurante é inserido no sistema e salvo no banco de dados.
	5. O administrador é informado se o restaurante foi inserido com sucesso
	6. Lista todos os restaurantes cadastrados

Tabela 5 – Caso de Uso Cadastro de Proprietários

Nome do Caso de Uso	Cadastro de Gerentes
Ator Principal	Administrador do Sistema
Resumo	Esse caso de Uso representa o cadastro de gerentes que será realizado pelo administrador do sistema.
Pré-Condições	Estar logado no sistema
Ações do Ator	Ações do Sistema
1. O administrador acessa o sistema para cadastrar os gerentes no sistema.	
	2. O sistema apresenta uma tela de cadastro.
3. Realiza o cadastro do gerente do restaurante.	
	4. O gerente é inserido e salvo no banco de dados.
	5. Lista todos os gerentes cadastrados.

ANEXO IV – DOCUMENTAÇÃO CASO DE USO – CADASTROS

Tabela 6 - Caso de Uso de Cadastrar Especialidades

Nome do Caso de Uso	Cadastrar Especialidades
Ator Principal	Administrador
Resumo	O administrador do sistema será responsável por cadastrar as especialidades dos restaurantes, como também editá-las.
Pré-Condições	Estar logado no sistema
Ações do Ator	Ações do Sistema
1. O administrador faz o login.	
2. Realiza o cadastro de especialidades	
	3. O sistema grava as informações no banco de dados.
	4. O sistema informa as especialidades cadastradas.

Tabela 7 – Caso de Uso Cadastro de Itens Avaliados

Nome do Caso de Uso	Cadastrar Itens avaliados
Ator Principal	Administrador
Resumo	O administrador do sistema será responsável por cadastrar os itens avaliados, como também editá-los e excluí-los.
Pré-Condições	Estar logado no sistema
Ações do Ator	Ações do Sistema
1. O administrador faz o login.	
2. Realiza o cadastro de itens avaliados	

	3. O sistema carrega as informações que estão gravadas no banco de dados.
	4. O sistema lista todos os itens cadastrados.

Tabela 8 – Caso de Uso Cadastrar Usuários

Nome do Caso de Uso	Cadastrar Usuários
Ator Principal	Administrador
Resumo	O administrador fará o cadastro no sistema, de usuários restaurantes, para que o usuário cadastrado tenha acesso diferenciado referente ao seu restaurante.
Pré-Condições	-
Ações do Ator	Ações do Sistema
1. Solicita ao restaurante informações de um novo usuário restaurante.	
2. Realiza o cadastro do usuário no sistema, possibilitando o acesso ao seu restaurante	
	3. O sistema salva as informações do usuário no banco de dados.
	4. Informa se cadastro foi realizado com sucesso, e demonstra o usuário cadastrado.

ANEXO V – DOCUMENTAÇÃO CASO DE USO – USUÁRIO RESTAURANTE

Tabela 9 - Caso de Uso de Cadastrar Promoções

Nome do Caso de Uso	Cadastrar Promoção
Ator Principal	Usuário Interno (Restaurante)
Resumo	O usuário restaurante será responsável por cadastrar as promoções dos restaurantes, como também editá-las.
Pré-Condições	Estar logado no sistema
Ações do Ator	Ações do Sistema
1. O usuário acessa o sistema para fazer cadastro das promoções.	
	2. Apresenta a tela de cadastros de promoções
3. Realiza o cadastro de promoções	
	4. O sistema grava as informações no banco de dados.
	5. O sistema informa as promoções cadastradas.

Tabela 10 - Caso de Uso de Cadastrar Cardápios Restaurantes

Nome do Caso de Uso	Cadastrar Cardápios Restaurantes
Ator Principal	Usuário Interno (Restaurante)
Resumo	O usuário restaurante será responsável por cadastrar os cardápios com os seus restaurantes, como também editá-los.
Pré-Condições	Estar logado no sistema

Ações do Ator	Ações do Sistema
1. O usuário acessa o sistema para cadastrar cardápios.	
	2. O sistema apresenta a tela de cadastros de cardápios.
3. Realiza o cadastro dos cardápios relacionando com seu restaurante.	
	4. O sistema grava as informações no banco de dados.
	5. O sistema informa os cardápios e restaurantes já relacionados.

Tabela 11 - Caso de Uso Dias da Semana

Nome do Caso de Uso	Dias da Semana
Ator Principal	Usuários Internos (Restaurantes)
Resumo	O sistema disponibiliza um cadastro com os dias da semana para utilização do cadastro de restaurantes e cardápios
Pré-Condições	Estar logado no sistema
Ações do Ator	Ações do Sistema
1. O usuário acessa o sistema para cadastrar os dias da semana no cardápio	
	2. O sistema fornece os dados cadastrados dos dias da semana
3. Utiliza os dados oferecidos pelo sistema no cadastro de cardápios.	

	4. O sistema grava as informações no banco de dados.
	5. Informa os dados cadastrados

Tabela 12 - Caso de Uso Cadastrar Cardápio do Dia

Nome do Caso de Uso	Cadastrar Cardápio do Dia
Ator Principal	Usuários Internos (Restaurantes)
Resumo	O usuário restaurante será responsável por cadastrar o cardápio do dia do seu restaurante, como também editá-los e excluí-los.
Pré-Condições	Estar logado no sistema
Ações do Ator	Ações do Sistema
1. O usuário acessa para cadastrar o cardápio do dia.	
	2. Apresenta a tela de cadastros de cardápios
3. Realiza o cadastro do cardápio	
	4. O sistema grava as informações no banco de dados.
	5. Lista o cardápio inserido.