

Sistema de Recomendação Mobile: Um Estudo de Caso para Delivery

Marcelo Garbin¹

Sidnei Renato Silveira²

Resumo. O presente artigo apresenta o desenvolvimento de um sistema de recomendação mobile para delivery. O sistema implementado possibilita que o usuário possa fazer, acompanhar e consultar o histórico de seus pedidos, acompanhamento do pedido, além de receber recomendações para futuras compras de acordo com seu perfil. As tecnologias utilizadas para o desenvolvimento deste sistema foram PHP, JavaScript, HTML5 com CSS3, utilizando os recursos do framework Phonegap com a integração dos bancos de dados MySQL e SQLite.

Palavras-Chave: Sistemas de Recomendação, Delivery, Aplicativo Móvel

Abstract. This paper presents the development of a recommender system for mobile delivery solution. The implemented system allows the user to make requests, monitoring of application, consult history and have recommendations for future purchases based on your profile. The technologies used for the development of this system will be PHP, JavaScript, HTML5 with CSS3, using the resources of the Phonegap framework with integration of MySQL and SQLite.

Keywords: Recommender Systems, Delivery, Mobile Application

1. Introdução

Esse trabalho tem seu objetivo motivado na modernização e utilização de Sistemas de Informação no comércio local (Frederico Westphalen - RS e região), bem como, no emprego de novas tecnologias, especialmente as que compreendem a computação móvel. Dentre as tecnologias que estão em ascensão, tanto para uso pessoal quanto para uso empresarial, estão às voltadas para dispositivos móveis, os quais se destacam por sua comodidade e versatilidade, impactando assim em uma nova oportunidade de aproximação entre clientes, empresa e serviços, tornando-se um dos principais fatores para o acréscimo da tecnologia nos dias atuais.

^{1,2} Departamento de Tecnologia da Informação Universidade Federal de Santa Maria
UFSM/Frederico Westphalen

A solução apresentada neste artigo envolve o desenvolvimento de um Sistema de Recomendação, utilizando recursos de computação móvel (plataforma *mobile*). Baseando-se nas preferências do usuário, os Sistemas de Recomendação surgem como aliados na busca de novos itens de consumo de acordo com o interesse indicado pelo usuário, fazendo com que estes sistemas sejam amplamente utilizados como uma estratégia na potencialização de novos negócios (SOUZA, 2012).

Outro fator importante para o desenvolvimento deste trabalho é o uso de *softwares* livres, tais como o sistema operacional *Android*, os *frameworks Phonegap*, *CodeIgniter 2* e *Jquery*, além da linguagem de programação PHP e os bancos de dados *MySQL* e *SQLite* juntamente com a linguagem de marcação *Hypertext Markup Language* (HTML), *Cascading Style Sheets* (CSS) e *JavaScript*. O uso do sistema operacional *Android*, neste trabalho, baseia-se na sua popularidade de utilização no mercado atual de *smartphones* e *tablets*, bem como devido a sua flexibilidade de programação (IDC, 2017).

Portanto, ao agregar as tecnologias propostas neste trabalho, desenvolvemos uma aplicação capaz de agregar um diferencial na utilização do serviço de entrega na área de alimentação – que é a área compreendida no estudo de caso, principalmente por estabelecimentos como pizzarias e lanchonetes, sobretudo os situados em nosso mercado regional. Sendo assim, acreditamos que esse diferencial possibilitará ao estabelecimento, além da organização dos pedidos, fidelização dos clientes atuais e também a busca por novos clientes em potencial, ter seu atendimento e relacionamento, junto ao cliente, melhorado, contribuindo assim para um possível acréscimo em suas vendas.

Neste sentido, o texto deste artigo está dividido em 5 seções. A seção 2 apresenta um breve referencial teórico das áreas envolvidas neste trabalho, apresentando conceitos de sistemas de recomendação, bem como alguns conceitos sobre Computação Móvel e *Web Services*. A seção 3 apresenta os trabalhos relacionados. Devido ao uso do serviço de *delivery mobile* ser relativamente novo na área, não foi possível obter dados precisos sobre tais implementações, porém foram feitas algumas comparações básicas entre os sistemas estudados e a solução implementada. A seção 4 apresenta a solução implementada, juntamente com as tecnologias, métodos e ferramentas que fizeram parte do desenvolvimento da aplicação. Para finalizar este artigo, a seção 5 apresenta as considerações finais sobre o trabalho.

2. Referencial Teórico

Esta seção apresenta alguns conceitos envolvidos no desenvolvimento deste trabalho, em especial a computação móvel e os sistemas de recomendação.

2.1 Computação Móvel

Segundo Cirilo (2007), a computação móvel consiste em sistemas computacionais distribuídos em diferentes dispositivos que se comunicam entre si por meio de uma rede de comunicação sem fio, o que permite a mobilidade desses aparelhos. O usuário, então,

é capaz de utilizar os serviços que um computador oferece independente de sua localização física.

Em relação ao conceito de computação móvel, é cabível defini-lo como sendo composto por dispositivos ou aplicativos, os quais possuem a capacidade móvel de transportar informações reais ao mesmo tempo em que são conectados à internet ou rede (MATEUS, 2017).

No momento em que nos movemos e mudamos de ambiente, entramos em um novo espaço em que a computação móvel passa a pesquisar uma nova forma de conexão. Neste cenário, a computação móvel tem a capacidade de obter novas informações do ambiente e utilizar de forma a atribuir a melhor conexão e se ajustar as necessidades do usuário (MATEUS, 2017).

O surgimento de novos sistemas e aplicações móveis vem se tornando crescente em nosso cotidiano, junto à mobilidade na utilização de dispositivos. O compartilhamento e processamento de informações são requisitos importantes no uso desta tecnologia.

2.1.1 Aplicativos para Computação Móvel

Um dos fatores relevantes para o crescimento na utilização de dispositivos móveis está ligado à utilização da internet, bem como *games* e redes sociais, como, por exemplo, o *Facebook*.

As aplicações que se utilizam da tecnologia móvel propiciam a melhoria de atividades comuns. Por exemplo, em um sistema para *delivery*, onde em qualquer cenário, casa, escritório, ou até mesmo no automóvel, com a utilização de um dispositivo móvel é possível fazer pedidos *online* na pizzaria mais próxima, em tempo real na comodidade de seu ambiente e com a possibilidade de pagamento remoto e ainda receber sua pizza em sua casa ou qualquer outro local de possível entrega, sem se quer fazer algum esforço a mais por isso.

Alguns pontos devem ser analisados no desenvolvimento de tais aplicativos. É importante observar que, para projetar um sistema que tenha uma identificação de conteúdo com uma navegação intuitiva, é necessário que o sistema possua uma boa usabilidade e adaptabilidade entre usuário e aplicação, fazendo com que o usuário final não fique confuso na utilização da mesma (KAHL, 2011).

2.1.2 Sistemas Operacionais para Plataformas Móveis

O mercado atual de dispositivos móveis nos proporciona uma gama de diversos modelos de aparelhos, junto deles, cada fabricante opta por um determinado sistema operacional, o qual é responsável pela integração entre aplicação e *hardware*. Diante disto iremos abordar algumas noções dos sistemas operacionais mais utilizados atualmente, que são o *Android* da *Google* e o *iOS* da *Apple*.

2.1.2.1 Android

O sistema operacional *Android* da *Google* é baseado no núcleo do sistema operacional Linux. Em 2008 seu código foi publicado como *Open Source*, neste mesmo ano o

primeiro aparelho móvel com o sistema *Android* foi lançado comercialmente (AMADEO, 2017).

Diferentemente do *iOS*, uma das principais características do *Android*, é a versatilidade com que o sistema é executado em dispositivos de fabricantes distintos. Este sistema operacional, não possui restrição de *hardware* sendo assim um atrativo para sua utilização por diversas empresas de dispositivos móveis (AMADEO, 2017).

O desenvolvimento de aplicativos para o *Android* é amplamente incentivado pela *Google*. O kit de desenvolvimento de software (SDK *Software Development Kit*) do *Android* é compatível com diversos sistemas operacionais. A linguagem de programação utilizada na criação de novos aplicativos é o JAVA. Para testar a aplicação pode ser feito o uso de simuladores que fazem parte do SDK, porém para poder comercializá-lo é necessário se cadastrar na loja de aplicativos *Google Play* e pagar uma taxa de US\$ 25,00. Após o cadastro de desenvolvedor no *Google Play*, é possível comercializar aplicativos para os dispositivos móveis que possuem o sistema *Android* (QUERINO FILHO, 2013).

2.1.2.2 iOS

Antes chamado *iPhone OS*, o sistema operacional móvel da *Apple* está presente nos dispositivos como *iPhone*, *iPod Touch*, *iPad* e também na *Apple TV*. A utilização do sistema não é permitida em *hardware* de terceiros e seu lançamento ocorreu no ano de 2007 juntamente com o *iPhone*. Suas primeiras versões não permitiam aplicações desenvolvidas por terceiros. O sistema *iOS* é composto por quatro camadas de abstração sendo, a primeira camada *Core OS*, seguida da camada *Core Services*, a camada mídia e por fim a camada *Cocoa Touch* (APPLE, 2013).

No desenvolvimento de aplicações para o *iOS*, os principais *frameworks* são encontrados na camada *Cocoa Touch*, nesta primeira camada alguns serviços de alto nível estão disponíveis, como serviço de notificação *Apple push*, proteção de dados e multitarefa. A camada mídia possui tecnologias que envolvem gráficos, áudio e vídeo, esta camada foi projetada para oferecer aos desenvolvedores uma forma mais fácil para o desenvolvimento de aplicativos multimídia. Na camada *Core Services*, alguns serviços como, *SQLite*, *XML support*, *in-app purchase* e *grand central dispatch*, são fundamentais. Estes serviços podem não ser utilizados por desenvolvedores, mas muitas partes do sistema são construídas sobre eles. A camada *Core OS* é o núcleo do sistema operacional, nesta camada são encontrados o OS Kernel, segurança, gerenciamento de energia, certificado e sistema de arquivos. Esta camada é utilizada por desenvolvedores quando se faz necessária a utilização de recursos na parte de segurança ou comunicação com acessórios de hardware externo (ROCHA et al., 2011).

Para poder iniciar a criação de aplicativos nativos para o *iOS*, primeiramente é necessário possuir um computador MAC, pois as ferramentas de desenvolvimento da *Apple* estão disponíveis apenas para seu sistema. O cadastro de desenvolvedor *Apple* é feito para que se possa testar os aplicativos desenvolvidos em um aparelho real e também para publicar o mesmo na *iTunes App Store*, que é a loja da *Apple* para aplicativos móveis. Para manter este cadastro ativo, faz-se necessário o pagamento de

US\$ 99,00 anuais. Após este cadastro, o desenvolvedor terá o direito de efetuar o *download* gratuitamente do SDK do *iOS* (SCHMITT, 2010).

2.1.2 Web Services

Em termos gerais um *web service* é uma solução que facilita a integração entre sistemas, plataformas ou linguagens. Booth (2017) explica que um *web service* é uma tecnologia que surgiu para flexibilizar o processo de interação de diferentes aplicações. De forma simples podemos definir *web service* como um *software* que permite uma interação com mais de uma interface; com ele é possível comunicar-se de forma padronizada em diferentes aplicações.

Para que isso ocorra basicamente é utilizada uma linguagem padrão, por meio da qual ocorre a interação através da troca de mensagens que, geralmente, são documentos que utilizam o *Extensible Markup Language* (XML). Ou seja, a linguagem do *web service* é importante e necessária para a comunicação entre diferentes aplicações, diferentes sistemas operacionais ou programação. Dentre essa linguagem padrão usada pelo *web service* podemos ainda incluir o protocolo *HyperText Transfer Protocol* (HTTP). No que se refere à troca de informações, podemos utilizar outros protocolos, entre os mais importantes estão os protocolos *Simple Object Access Protocol* (SOAP), *Universal Description, Discovery and Integration* (UDDI) e *Web Services Description Language* (WSDL), responsáveis pela entrega das mensagens a aplicação.

Os *web services* tornaram-se o padrão de comunicação entre diferentes linguagens e aplicações principalmente no que se refere ao acesso pela internet. O *web service* tem como finalidade tornar esse processo de comunicação mais leve e dinâmico, além de ser mais fácil de ser aceito pelos diversos desenvolvedores de software (ERL, 2017).

Uma das grandes vantagens dos *web services* envolve a capacidade da interligação de diferentes componentes. Muito utilizados por empresas, essa solução trouxe agilidade no processo de interação e comunicação, disponibilizando informações através da internet. A Figura 1 apresenta um exemplo de como pode ser feita a comunicação entre cliente-servidor, por meio de um *web service* que utiliza as linguagens de programação PHP e *JavaScript*. A aplicação do lado do cliente faz a requisição de algum determinado dado enviando esta requisição para o servidor, sendo interpretado em uma página PHP. Posteriormente, após o tratamento da entrada dos dados, a página PHP faz a consulta com o banco de dados; o banco de dados verifica qual a situação da requisição e retorna novamente para a página PHP, a página PHP faz o processamento e com isso faz o retorno ao cliente através de um arquivo em formato *json*.

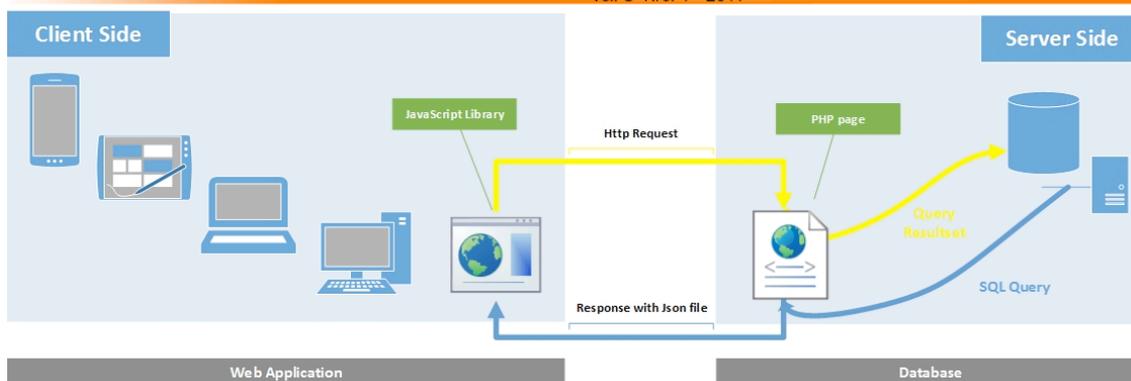


Figura 1. Funcionamento de um *web service* utilizando PHP e JavaScript
(Fonte: dos autores, 2017)

O sistema operacional *Android* conta com um banco de dados integrado; o mesmo é gerado em um arquivo que é armazenado internamente no dispositivo. Sendo assim, este banco de dados tem seu acesso limitado apenas para tal dispositivo. Para que se possam efetuar cadastros, consultas, e ter acesso a um banco de dados remoto, a utilização de *web services* se faz necessária para que se possa ter uma integração entre tais bancos de dados com segurança (RANCH, 2017).

2.2 Sistemas de Recomendação

A propagação e utilização da Internet fez com que a utilização de sistemas de *e-commerce* se tornasse uma alternativa bastante atrativa para empresas físicas entrarem no meio virtual. Um dos principais benefícios que o *e-commerce* trouxe foi o aumento das vendas e fidelização de clientes. Diante deste fato, inúmeros itens são ofertados dentro de um site de *e-commerce* (LINDEN, 2003).

A grande variedade e quantidade de itens ofertados em sites de *e-commerce* alavancaram a necessidade de um sistema de recomendação, o qual faz de seu papel principal, o auxílio ao usuário nas escolhas de itens de acordo com seu interesse e necessidade, também podendo haver a possibilidade de uma ou mais recomendações serem vindas de outros usuários (LINDEN, 2003).

Sistemas de Recomendação (SR) são ferramentas de *software* e técnicas que fornecem sugestões de itens para serem de utilidade de um usuário. As sugestões referem-se a vários processos de tomada de decisão, tais como quais itens compatíveis com o interesse estão disponíveis para se comprar, sugestão de música para ouvir, ou quais notícias *on-line* são destaques e que são de interesse para ler (SOUZA, 2012).

“Item” é o termo geral usado para designar o que o sistema recomenda aos usuários. A recomendação normalmente se concentra em um tipo específico de item (por exemplo, CDs ou notícias) e, conseqüentemente, seu design, sua interface gráfica do usuário, e a técnica de recomendação central usada para gerar as recomendações são todas personalizadas para fornecer sugestões úteis e eficazes para esse específico tipo de item (RICCI, 2011).

O processo de recomendação consiste nos seguintes passos (TORRES, 2004):

1. O usuário fornece ao sistema a sua opinião sobre os assuntos em questão;
2. O sistema compara essa opinião com as dos demais usuários e seleciona os usuários com os perfis mais parecidos;
3. Baseando-se nos gostos dos usuários selecionados, o sistema recomenda os itens que têm maior probabilidade de semelhança;
4. Se o usuário conhece alguns dos itens recomendados, pode classificá-los e voltar-se ao ponto 2 permitindo obter sugestões mais confiáveis.

Para coletar a opinião dos usuários sobre os assuntos do SR podemos coletar os dados e forma implícita ou explícita, como apresentam as próximas seções.

2.2.1 Extração Implícita

A extração implícita de dados do usuário ocorre de uma maneira imperceptível ao mesmo. Com a utilização do sistema o usuário tem as informações coletadas de forma transparente; essas informações podem ser relativas ao histórico de compras, itens procurados, cliques em anúncios, acesso a menus e categorias dentro do sistema, entre outros. Após serem coletadas tais informações sobre o usuário, as mesmas são armazenadas em um banco de dados, para que possam ser utilizadas posteriormente (TORRES, 2004).

2.2.2 Extração Explícita

No método de extração explícita, diferentemente do método de extração implícita, o usuário é quem define e informa quais os conteúdos que são de seu interesse. Esses dados podem ser extraídos de diversas maneiras, tais como questionários no momento do cadastro em determinado *web site*, avaliações de diversos tipos de itens entre outras formas de coleta de dados. Com os dados coletados e armazenados, é criado um perfil para o usuário que receberá indicações para possíveis itens de seu interesse. Este perfil pode ser alterado através de outras perguntas que o sistema poderá fazer em tempos diferentes, como por exemplo, diariamente, semanalmente, etc. (TORRES, 2004).

2.2.3 Técnicas de Recomendação

Para que um sistema de recomendação possa recomendar os itens mais adequados aos usuários, podem-se utilizar diferentes técnicas de recomendação, dentre as quais destacamos a filtragem baseada em conteúdo e a filtragem colaborativa (TORRES, 2004):

- Na filtragem baseada em conteúdo, o sistema recomenda itens que são semelhantes aos que o usuário gostou no passado. A semelhança de itens é calculada com base nas características associadas com os itens comparados. Por exemplo, se um usuário tem avaliado positivamente um filme que pertence ao gênero de comédia, então o sistema pode recomendar outros filmes do mesmo gênero. Uma das vantagens deste tipo de sistema é que os mesmos são simples para dados textuais e não necessitam de muitas informações sobre um usuário para sugerir itens. Todavia, além de serem difíceis de aplicar em contextos

multimídia, podem oferecer recomendações repetitivas, recomendando sempre assuntos, produtos ou serviços que o usuário já conhece (SOUZA, 2012);

- A filtragem colaborativa consiste na recomendação de itens que pessoas, com gosto semelhante, preferiram no passado. Analisa-se a vizinhança do usuário a partir da regra: "Se um usuário gostou de A e de B, um outro usuário que gostou de A também pode gostar de B". Esse tipo de recomendação apresenta resultados positivos na prática, e evita o problema de recomendações repetitivas. Uma desvantagem é que requer grande número de informações sobre o usuário e sua vizinhança para funcionar precisamente (SOUZA, 2012).

Podemos utilizar os dois tipos de filtragem de forma combinada, em um sistema híbrido. Este tipo de filtragem consiste em combinar as duas abordagens mencionadas, tentando fortificá-las e superar suas desvantagens (SOUZA, 2012).

2.3 Aplicativos Mobile Multiplataforma

O mercado de dispositivos móveis impõe uma vasta diversidade de modelos, plataformas e linguagens para serem utilizadas no desenvolvimento de aplicativos móveis. Com isso o tempo para desenvolvimento aumenta significativamente juntamente com o custo final do produto (MINETTO, 2014).

Diante da necessidade de diminuir a curva de aprendizado e, conseqüentemente, o tempo no desenvolvimento de aplicações móveis juntamente com sua manutenção podemos empregar *frameworks* ou *kits* de desenvolvimento. A utilização de *kits* de desenvolvimento distintos para cada sistema pode ser trocada por *frameworks* que fazem do uso do HTML5, CSS3 e *Javascript* na criação de aplicativos nativos multiplataformas de diversos dispositivos (STARK, 2010).

A utilização de padrões *web* no desenvolvimento de aplicativos *mobile* não impede que o sistema desenvolvido possua suporte a alguns recursos presentes em distintos sistemas operacionais, alguns recursos como *Global Positioning System* (GPS), notificações, acelerômetro, arquivos e entre outros podem ser utilizados a partir do uso de um *framework* adequado que englobe tais recursos nativamente (STARK, 2010).

3. Estado da arte

Nesta seção serão apresentados alguns sistemas relacionados com a área de *Delivery* que são utilizados em dispositivos móveis. Ao final da seção será traçado um comparativo entre os sistemas e o trabalho aqui apresentado.

3.1 iFood

O *iFood* é um aplicativo que possui o cadastramento de estabelecimentos e usuários interessados pelo serviço de entrega de refeições. A Figura 2 demonstra algumas telas do sistema *iFood* para *Android*. São três telas que demonstram os restaurantes

cadastrados, os pratos de determinado restaurante e as formas de pagamento que são aceitas pelo mesmo (MENDONÇA, 2011).

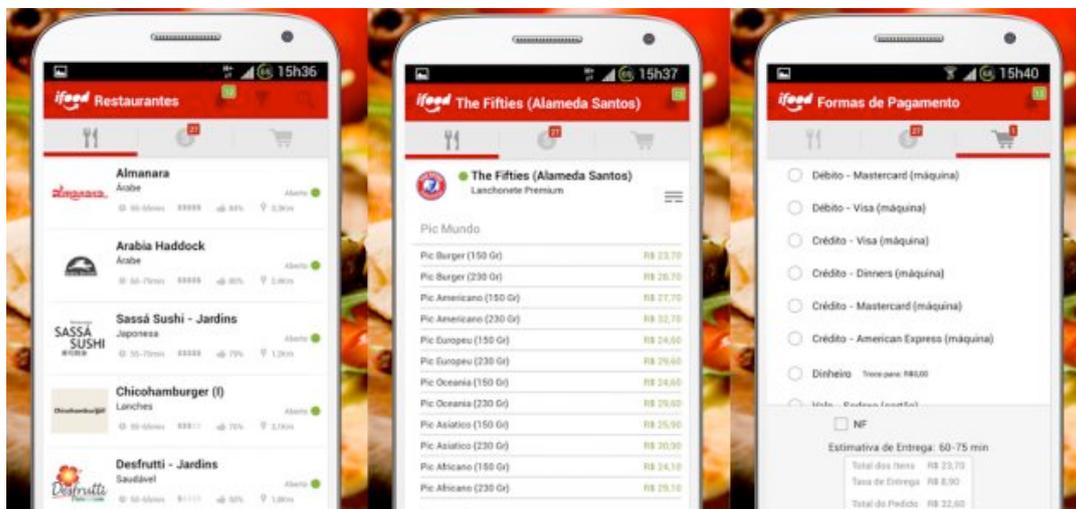


Figura 2. Ilustração das telas do iFood para o sistema Android (Fonte: Google Play, 2016)

O lançamento do *iFood* ocorreu em maio de 2011 através do desenvolvimento de um *site*. A ideia principal era que a utilização do telefone para a realização de pedidos de lanches pudesse ser substituída pela utilização da internet. Com a crescente procura pelo serviço *online* de *delivery*, a empresa *iFood* anunciou em 2012 o lançamento de seu aplicativo para os sistemas *iOS* e *Android* (IFOOD, 2017).

O aplicativo permite que o usuário possa utilizar da geolocalização dos dispositivos móveis para que seja possível obter uma lista de estabelecimentos localizados na área de entrega desejada. O usuário também possui acesso a cardápios, preços, formas de pagamentos e também é possível efetuar o *login* no sistema por uma conta do *Facebook*. Atualmente o *iFood* atende grandes centros. Inicialmente a região atendida compreendia São Paulo, partindo para Rio de Janeiro e sendo expandido para as regiões de Recife e Fortaleza (IFOOD, 2017).

3.2 Delivery Much Online

A *Delivery Much* é uma empresa especializada em *marketing* e pedidos *online* para empresas do ramo alimentício. A Figura 3 mostra a tela inicial do sistema, na qual podemos ver que são demonstrados os estabelecimentos cadastrados e os campos para acessar o sistema e também os pedidos já feitos (DELIVERY MUCH ONLINE, 2017).

Figura 4. Formas de entrega e pagamento do pedido (Fonte: Web site Delivery Much, 2017)

O sistema *Delivery Much* é um sistema inovador na região de Santa Maria, Pelotas e Passo Fundo, porém ainda não conta com uma aplicação nativa *mobile*. Seu acesso por meio de dispositivos móveis se dá pelo próprio *browser* do sistema operacional, sendo assim o sistema é acessado como um *web site* (GELATTI, 2014).

Por se tratar de um novo sistema, o mesmo deve passar por algumas reformulações. Funcionalidades como um sistema de recomendação de serviços e recomendação de itens, ainda precisam ser implementadas.

3.3 Comparação entre sistemas

Após o estudo sobre sistemas semelhantes ao implementado, elaboramos um quadro com os comparativos entre os mesmos. O resultado da comparação pode ser visualizado no Quadro 1.

Quadro 1 – Comparativo entre sistemas

Características	<i>iFood</i>	<i>Delivery Much</i>	Sistema Implementado
Plataforma Desenvolvida	<i>Android/iOS</i>	<i>Web</i>	<i>Web/Android</i>
Linguagem de Programação	<i>Java/Objective-C</i>	PHP	PHP/ <i>JavaScript</i>
Possui recomendação de itens/produtos	Não	Não	Sim
Possui edição de recomendação de itens/produtos pelo usuário	Não	Não	Sim
Suporte nativo para acesso a dispositivos móveis	Sim	Não	Sim

O Quadro 1 apresenta o comparativo entre os sistemas estudados e o sistema implementado. Foram encontradas algumas semelhanças entre os mesmos. Entretanto, os diferenciais implementados neste trabalho são:

- A linguagem de programação utilizada envolve o PHP para a plataforma *Web* e *JavaScript* para a plataforma *mobile*;
- O sistema de recomendação é o principal diferencial neste trabalho, sendo possível, por parte do usuário, efetuar a alteração com novas preferências de recomendação.

- **Implementação do Banco de Dados:** foi utilizado o banco de dados SQLite para o aplicativo *mobile*, sendo que o banco de dados principal da aplicação é o MySQL. O banco de dados principal servirá para gerenciar os cadastros, consultas e relatórios de acordo com a proposta do trabalho;
- **Administração do Sistema:** foi desenvolvido um módulo de administração do sistema. Neste módulo é possível cadastrar e gerenciar informações dos produtos, gerenciar pedidos, gerenciar usuários, visualizar relatórios através de interface *web* para *desktop*;
- **Aplicativo Mobile:** foi realizada a implementação de um aplicativo *mobile* utilizando o *framework Phonegap*. Neste aplicativo pode ser feito o cadastro de um novo usuário, configuração de perfil, escolha de produtos para efetuar um pedido, recomendações de produtos e acompanhamento do pedido;
- **Web Site:** criação de um *web site* utilizando *frameworks* para o auxílio no desenvolvimento da aplicação. Este *web site* visa o atendimento de usuários que não possuem um dispositivo móvel.

4.2 Modelagem da aplicação

Na modelagem da aplicação, são demonstrados alguns dos principais elementos que fazem parte do sistema. A Figura 6 mostra um diagrama de caso de uso com as funcionalidades do sistema implementado.

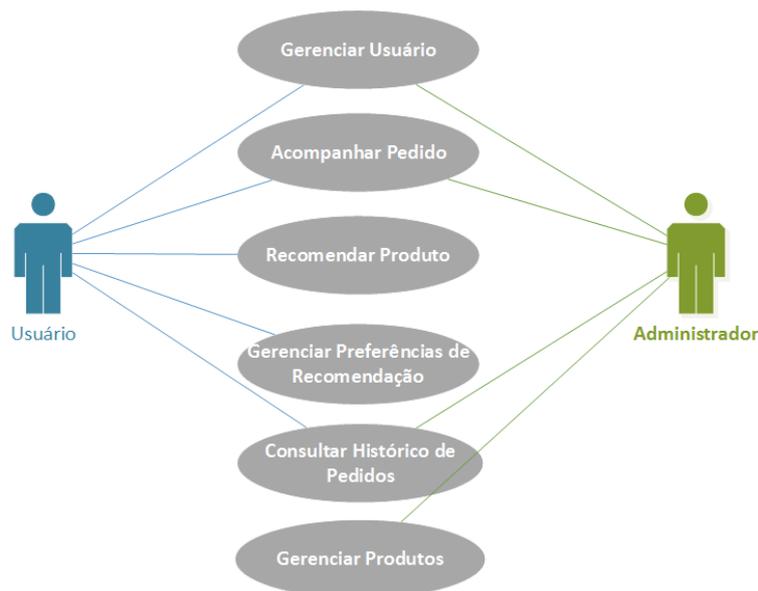


Figura 6. Diagrama de Caso de Uso. (Fonte: dos autores, 2017)

Após definir as funcionalidades previstas para o sistema, foi elaborado um diagrama de entidade e relacionamento (DER), o qual demonstra as tabelas que são necessárias para guardar os dados da aplicação e também as informações necessárias para a recomendação de produtos feitas para os usuários do sistema. O DER proposto pode ser visualizado na Figura 7.

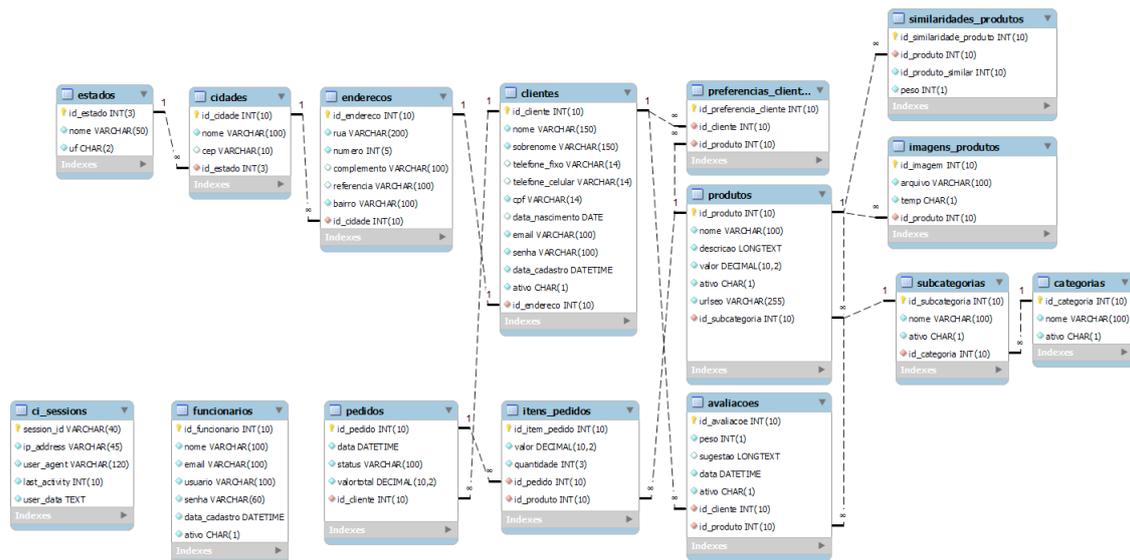


Figura 7. Diagrama Modelo E-R. (Fonte: dos autores, 2017)

Com base na Figura 7, apresentaremos uma breve explicação sobre as tabelas que compõem o banco de dados do sistema implementado:

- Tabela *ci_sessions*: é responsável por armazenar dados vindos do *framework CodeIgniter*, os principais campos dela são *ip_address*, *user_agent*, *last_activity* e *user_data*. Esses campos são utilizados para guardar dados das sessões de acessos dos usuários ao sistema, tendo assim um maior controle de segurança para usuários logados no sistema desenvolvido por meio deste *framework*;
- Tabela *funcionarios*: é responsável pelo armazenamento dos dados dos administradores do sistema que terão acesso à área restrita e administrativa do mesmo;
- Tabelas *clientes* e *enderecos*: nestas tabelas são armazenados os dados que são requisitados para o cadastro de um novo usuário no sistema. A mesma possui um relacionamento com a tabela *enderecos*, a qual possui os campos necessários para o armazenamento de dados sobre o endereço do usuário;
- Tabelas *categoria*, *subcategoria* e *produtos*: as tabelas *categoria* e *subcategoria* possuem campos em comum, como o campo nome e ativo, porém a tabela *subcategoria* possui uma chave estrangeira que cria o relacionamento com a tabela *categoria*. A tabela *produto* possui a chave estrangeira da tabela *subcategoria*, sendo assim, para o cadastramento de novos produtos no sistema, as tabelas *categoria* e *subcategoria* devem possuir dados para haver a associação dos campos;
- Tabelas *pedidos* e *itens_pedidos*: essas tabelas são as responsáveis pelo armazenamento dos itens selecionados pelos clientes na hora da compra. Ao adicionar itens no carrinho de compra e finalizar a compra, a tabela

itens_pedidos, grava a chave estrangeira da tabela *produtos* e também da tabela *pedidos*, como também os campos de valor e quantidade de cada produto comprado. A tabela *pedidos* possui relacionamento com a tabela *clientes*, para assim identificar o cliente responsável por cada pedido concluído, juntamente com o *status* de entrega de cada um;

- As tabelas *preferencias_clientes*, *avaliacoes* e *similaridades_produtos* são tabelas que fazem parte do funcionamento do sistema de recomendação. A tabela *preferencias_clientes* salva os dados de quais produtos o cliente mais prefere. Esses dados são coletados no cadastro de um novo usuário como, também podem ser alterados posteriormente pelo usuário. A tabela *similaridades_produtos* é utilizada para que os administradores do sistema possam gerenciar a similaridade entre produtos, para que as recomendações sejam realizadas de acordo com as preferências dos usuários. Na tabela *avaliacoes* são armazenadas informações que os clientes cadastram após a conclusão e entrega do produto, campos como peso (nota), sugestão e data da avaliação fazem parte da mesma. Esses dados são utilizados para geração de recomendações com base na nota de cada item.

Todas as tabelas citadas anteriormente possuem chaves primárias. As funções para adicionar, excluir e alterar dados se dá por meio de formulários via interface *web*.

4.3 Tecnologias Empregadas

Algumas das tecnologias empregadas para o desenvolvimento para dispositivos *mobile* foram estudadas no decorrer deste trabalho. A linguagem de programação atualmente mais utilizada para desenvolver aplicativos para sistema *Android* é o Java. O sistema operacional *Android* é um dos mais utilizados em dispositivos móveis (IDC, 2017).

Caso optássemos pelo desenvolvimento de um aplicativo nativo em diversos sistemas diferentes, no desenvolvimento da aplicação, as ferramentas que cada plataforma proporciona para o desenvolvedor teriam que ser utilizadas separadamente. Porém o tempo gasto e a curva de aprendizagem não compensariam no desenvolvimento de um único aplicativo para várias plataformas sem saber quais plataformas seu público alvo estaria utilizando.

Uma das tecnologias empregadas neste trabalho envolve o *framework Phonegap*. Esta tecnologia é *Open Source*² que tem como principal característica desenvolver aplicativos *webapp*, que são *web sites* similares a aplicativos nativos e que utilizam de funcionalidades das plataformas móveis. O desenvolvimento com o *Phonegap* pode gerar aplicativos funcionais para ambientes tanto *online* como *off-line* (AMBROS, 2017).

² O termo "*open source*" foi cunhado em 1998. Ele se refere ao software cujo código-fonte pode ser lido (visto), escrito (modificado) e adquirido gratuitamente ou por uma taxa simbólica. Trata-se de um conjunto de critérios empírico e mensurável para o código propriamente dito. Contudo, o OSS também possui dimensões sociais e comerciais. Assim, há uma série de variações no conceito de código aberto e disponível, cada um com mais ou menos ênfase em diversas prioridades de desenvolvedores ou usuários (MICROSOFT, 2017).

O *Phonegap* é mantido pela *Adobe Systems* e utiliza padrões *web*, tais padrões incluem o HTML atualmente na versão 5, *Cascading Style Sheets* (CSS) atualmente na versão 3 e *JavaScript*, a qual é uma linguagem de *script* interpretada com base em objetos. A tecnologia utilizada para o uso de banco de dados é o *SQLite*. De acordo com as configurações feitas durante a utilização do *framework*, é possível disponibilizar o aplicativo para mais de 7 sistemas operacionais diferentes, tais como *Android*, *iOS* e *Windows Phone*.

O banco de dados utilizado na aplicação mobile é o *SQLite*, o mesmo já possui suporte junto ao *Phonegap*. O *SQLite* é uma biblioteca desenvolvida em C e sua sintaxe padrão é SQL (SQLITE, 2017), para que seja possível a comunicação entre o banco de dados local da aplicação mobile e o banco de dados remoto que será do *web site*, será utilizado um *web service*, dessa forma o MySQL é utilizado como banco de dados remoto e também como banco principal da aplicação. O MySQL é um banco de dados gratuito, de código fonte aberto, seguro e prático (SANTOS, 2017). Para a programação *mobile* será utilizada a linguagem de *script JavaScript*. Esta linguagem foi desenvolvida para ser executada no lado do cliente, sendo assim, suas funcionalidades dependem das funcionalidades hospedadas no navegador do usuário (SAMMY, 2010).

Na parte do sistema *web*, foi utilizado o *framework CodeIgniter 2*, uma das principais características é a utilização das camadas *Model*, *View* e *Controller*, estas camadas fazem a separação da lógica do negócio, a qual envolve banco de dados, controladores e visões da aplicação (SOFTERIZE, 2017).

A implementação da interface e criação do layout *web* foi realizada por meio da utilização do *framework front-end Bootstrap 3*, como pode ser visto na Figura 8, o qual é um *framework open-source* mantido e criado pelo Twitter, com ele é possível a criação de layout responsável, que segundo Silva (2014), seu conceito deve ser entendido como o design capaz de “responder” às características do dispositivo ao qual é servido. Responder, nesse contexto, tem o sentido de movimentar-se expandindo e contraindo.

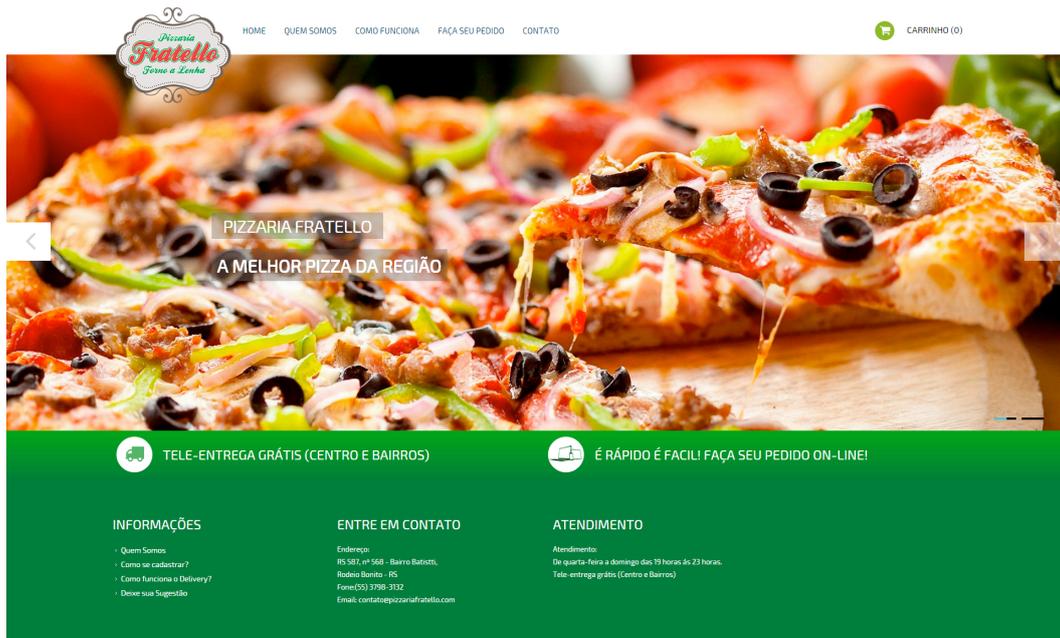


Figura 8. Página Inicial do *website* (Fonte: dos autores, 2017)

4.4 Geração das Recomendações

Para a geração das recomendações foram utilizados dados já cadastrados pelo administrador do sistema através do módulo de administração, como mostra a Figura 9. Esses dados são os itens que indicam a categoria, subcategoria e o produto, que neste caso são as *pizzas*.

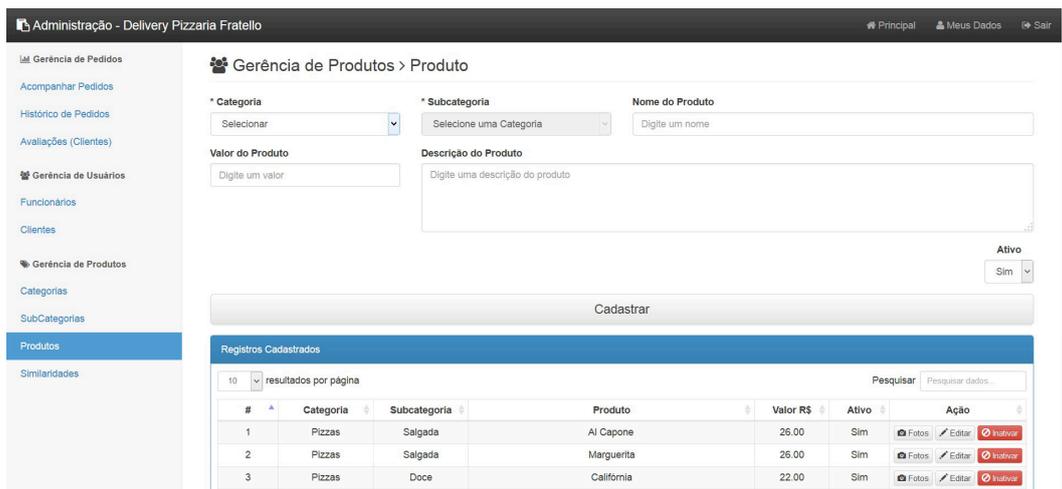


Figura 9. Gerenciamento de Produtos. (Fonte: dos autores, 2017)

Com estes dados cadastrados na base de dados é possível efetuar a coleta de informações do usuário de forma explícita na hora de seu cadastro, dessa forma será possível traçar o perfil inicial e gerar recomendações para o mesmo.

Quando um novo usuário efetua seu cadastro no sistema, é solicitado que o mesmo entre com suas preferências de recomendação, conforme mostra a Figura 10.

HOME QUEM SOMOS COMO FUNCIONA FAÇA SEU PEDIDO CONTATO

CARRINHO (0)

Cadastro de Usuário > Novo

Nome: Digite um Nome

Sobrenome: Digite o sobrenome

Telefone Fixo: Digite o telefone fixo

Telefone Celular: Digite o telefone celular

CPF: Digite o CPF

Data de Nasc.: Data de Nascimento

E-mail: Digite um E-mail

Senha: Digite uma Senha

★ Preferências de Recomendação

Pizzas - Salgadas

Bacon Brócolis Calabreza Carpaccio

Coração Frango Catupiri Marguerita Mussarela

Pizzas - Doces

Banana Brigadeiro Califórnia Prestigio

Chocolate Branco Romeu e Julieta Tropical

Cancelar Cadastrar Usuário

TELE-ENTREGA GRÁTIS (CENTRO E BAIRROS)

É RÁPIDO É FÁCIL! FAÇA SEU PEDIDO ON-LINE!

INFORMAÇÕES

- Quem Somos
- Como se cadastrar?
- Como funciona o Delivery?
- Deixe sua Suggestion

ENTRE EM CONTATO

Endereço:
RS 567, nº 558 - Bairro Babatti,
Bairro Boato - RS
Fone: (51) 3390-3832
Email: contato@pizzafratello.com

ATENDIMENTO

Atendimento:
De quarta-feira a domingo das 19 horas às 23 horas.
Tele-entrega grátis (Centro e Bairros)

Figura 10. Cadastro de Usuário (Fonte: dos autores, 2017)

Para ativar e visualizar as recomendações para o usuário é necessário clicar em “Desejo Recomendações”. Após clicar neste botão, o usuário será redirecionado para outra página onde é possível visualizar uma lista de itens que foram recomendados e que possuem relação com as preferências do usuário logado no sistema, como mostra a Figura 11. Também é possível clicar no botão “Mais Recomendações” (no lado direito da tela, na parte inferior) para visualização de outros itens.

Olá Usuário, Seja Bem-vindo | Meus Dados | Acompanhar Pedidos | Preferências de Recomendação | Sair

HOME QUEM SOMOS FAÇA SEU PEDIDO DESEJO RECOMENDAÇÕES CONTATO

CARRINHO (0)

★ Lista de Recomendações

Olá Usuário, encontramos alguns produtos que talvez você possa estar interessado. Caso houver interesse em algum desses itens, clique no botão "Adicionar" para enviar o mesmo para o seu carrinho de compras.

A pontuação média é a soma de todas as avaliações dos clientes, a pontuação mínima é 1 e máxima 5.

	Marguerita Categoria: Pizzas Salgadas Descrição: mussarela, salame italiano, tomate seco, requeijão	Pontuação Média 4,51	R\$ 26,00 Adicionar
	Brigadeiro Categoria: Pizzas Doces Descrição: brigadeiro e granulado preto (opção-morango)	Pontuação Média 4,36	R\$ 26,00 Adicionar
	Calabresa Categoria: Pizzas Salgadas Descrição: mussarela, calabresa	Pontuação Média 3,98	R\$ 26,00 Adicionar

[Mais Recomendações](#)

TELE-ENTREGA GRÁTIS (CENTRO E BAIRROS) É RÁPIDO É FÁCIL! FAÇA SEU PEDIDO ON-LINE!

INFORMAÇÕES

- Quem Somos
- Como se cadastrar?
- Como funciona o Delivery?
- Dê sua Sugestão

ENTRE EM CONTATO

Endereço:
RS 501, nº 568 - Bairro Batista,
Rodeio Bonito - RS
Fone:(51) 3798-3132
Email: contato@pizzariafratello.com

ATENDIMENTO

Atendimento:
De quarta-feira a domingo das 19 horas às 23 horas.
Tele-entrega grátis (Centro e Bairros)

Figura 11. Lista de Recomendações (Fonte: dos autores, 2017)

Caso as recomendações oferecidas pelo sistema não sejam mais compatíveis com as preferências do usuário, o mesmo poderá editá-las em seu perfil clicando no menu “Preferências de Recomendação” localizado na barra superior na cor verde da página.

Os tipos de filtragem que foram empregados neste sistema são a baseada em conteúdo e a colaborativa. Na filtragem baseada em conteúdo, são utilizadas as avaliações dos produtos que o usuário fez anteriormente como base. Na filtragem colaborativa, se um usuário possuir as preferências semelhantes a outros, será demonstrada uma lista das pizzas que foram bem avaliadas e também das piores avaliadas pelos demais usuários. A avaliação de itens proposta neste trabalho foi feita da seguinte forma: após o usuário efetuar o pedido e a entrega ser concluída no sistema, o usuário fornece uma nota de 1 a 5, para o serviço de entrega como também para os itens/produtos comprados. Sendo a nota 1 ruim, 2 regular, 3 bom, 4 muito bom e nota 5 excelente. Assim é possível gerar uma lista de melhores e piores itens avaliados.

4.4 Validação

Após o desenvolvimento do sistema, foi feita a validação do mesmo junto ao gerente da Pizzaria Fratello, empresa localizada na cidade de Rodeio Bonito - RS. O gerente fez o acompanhamento das funcionalidades e do funcionamento do sistema.

Após o acompanhamento da implementação do sistema, o gerente fez alguns elogios e sugestões. O mesmo acredita que este tipo de sistema é um diferencial para sua empresa e também para a comodidade dos clientes, mantendo assim um laço entre

clientes e a empresa. Algumas das sugestões feitas pelo gerente são apresentadas nas considerações finais deste trabalho como trabalhos futuros.

No primeiro momento foi feita a demonstração do sistema localmente. Após a aprovação do mesmo foi feito a requisição para registro de domínio em nome da empresa na internet como também da hospedagem *web*. Para o aplicativo *mobile* foi realizada a requisição de cadastro no *Google Play* para a publicação do mesmo.

5. Considerações Finais

Acredita-se que os principais objetivos propostos neste trabalho foram alcançados. O presente trabalho apresentou técnicas de sistemas de recomendação, as quais foram utilizadas na implementação de um aplicativo *mobile*, bem como, no desenvolvimento de um *website*.

Estes tipos de sistemas, *mobile* e *web*, que envolvem técnicas de recomendação, são conhecidos de grandes empresas, porém para a realidade local de muitos empresários da região de Frederico Westphalen - RS, é uma inovação no mercado regional. Devido à escassez de sistemas como esses em nosso meio, foi um desafio e uma oportunidade para desenvolver tal sistema, com isso muito conhecimento foi agregado neste trabalho. Dessa forma a implementação do sistema de recomendação, além de ser um diferencial para a empresa, também proporcionou novos conhecimentos.

Entretanto, devido ao desenvolvimento de aplicativos *mobile* voltado para sistemas de *delivery* ser um assunto relativamente novo, alguns estudos na área precisariam ser mais aprofundados, tendo em vista a continuidade do projeto. Pretendendo aplicá-lo efetivamente para fins comerciais, alguns trabalhos futuros deverão ser elaborados, tais como a elaboração e adaptação da interface para portadores de necessidades especiais; um módulo para Reservas e Eventos na sede da empresa; um módulo para controle de estoque de produtos; um módulo para controle financeiro sobre as vendas e integração com sistemas de pagamentos *online*.

Referências

- AMADEO, R. **The history of Android**. Disponível em: <<http://arstechnica.com/gadgets/2014/06/building-android-a-40000-word-history-of-googles-mobile-os>>. Acesso em junho, 2017.
- AMBROS, L. **Diferença entre Aplicativos Nativos, Híbridos e Mobile Web Apps**. Disponível em: <<http://luisaambros.com/blog/diferenca-entre-aplicativos-nativos-hibridos-e-mobile-web-apps>>. Acesso em junho, 2017.
- APPLE, Inc. **iOS Technology Overview**. Disponível em: <<https://developer.apple.com/library/ios/documentation/miscellaneous/conceptual/iPhoneTechOverview/Introduction/Introduction.html>>. Acesso em junho, 2014.
- BOOTH, D. et al. **Web Services Architecture**, W3C Working Group. Disponível em: <<http://www.w3.org/TR/ws-arch/>>. Acesso em julho, 2017.
- CIRILO, C. E. **Computação Ubíqua: definição, princípios e tecnologias**. 2007. Disponível em:

- <https://www.academia.edu/1733697/Computacao_Ubiqua_definicao_principios_e_tecnologias>. Acesso em abril, 2017.
- DELIVERY MUCH ONLINE, **Delivery Much - Imprensa**. Disponível em: <<https://www.deliverymuch.com.br/sobre/imprensa>>. Acesso em junho, 2017.
- ERL, T. **Introdução às tecnologias Web Services: SOA, SOAP, WSDL e UDDI**. Disponível em: <<http://www.devmedia.com.br/introducao-as-tecnologias-web-services-soa-soap-wsdl-e-uddi-parte1/2873>> Acesso em maio, 2017.
- GELATTI, J. **Delivery Much é pioneira em oferecer em um só site diferentes serviços de tele-entrega de comida**. 2014. Disponível em: <<http://diariodesantamaria.clicrbs.com.br/rs/economia/noticia/2014/02/delivery-much-e-pioneira-em-oferecer-em-um-so-site-diferentes-servicos-de-tele-entrega-de-comida-4418768.html>>. Acesso em junho, 2017.
- IDC (2017). **Smartphones OS Market Share**. Disponível em: <<http://www.idc.com/promo/smartphone-market-share/os>>. Acesso em Março de 2017.
- I FOOD. **Qual a história do iFood? Como Surgiu?**. Disponível em: <<https://ifood.zendesk.com/hc/pt-br/articles/200740154-Qual-a-hist%C3%B3ria-do-iFood-Como-surgiu->>. Acesso em junho, 2017.
- KAHL, M; FLORIANO, D. **Computação Ubíqua, Tecnologia Sem Limites**. Disponível em: <http://www.ceavi.udesc.br/arquivos/id_submenu/387/diogo_floriano_marcelo_kahl_computacao_ubiqua.pdf>. Acesso em maio, 2017.
- LINDEN, G.; SMITH, B.; YORK, J. **Amazon.com Recommendations Item-to-Item Collaborative Filtering**. Disponível em: <<http://www.cs.umd.edu/~samir/498/Amazon-Recommendations.pdf>>. Acesso em junho, 2017.
- MATEUS, G. R.; LOUREIRO, A. A. F. **Introdução à Computação Móvel**. Disponível em: <http://homepages.dcc.ufmg.br/~loureiro/cm/docs/cm_livro_1e.pdf>. Acesso em maio, 2017.
- MENDONÇA, Marcelo. **O Inovador Serviço iFood**, Bahia Delivery. Disponível em: <http://www.bahiadelivery.com.br/entrevista.php?entrevista_id=13>. Acesso em: 12 jun. 2014.
- MICROSOFT. **"Participação em um Mundo de Escolhas" - Perspectivas sobre Open Source e Microsoft**. Disponível em: <<http://www.microsoft.com/brasil/opensource/resource-center.aspx>>. Acesso em junho, 2017.
- MINETTO, E.. **Quanto custa criar um aplicativo**. 2014. Disponível em: <<http://startupi.com.br/2014/quanto-custa-criar-um-aplicativo>>. Acesso em abril, 2017.
- QUERINO FILHO, L. C. **Desenvolvendo seu primeiro aplicativo Android**. São Paulo: Novatec, 2013.
- RANCH, H. **Sqlite for beginners**. Disponível em: <<http://henryranch.net/software/sqlite-for-beginners>>. Acesso em junho, 2017.
- RICCI, F.; ROKACH, L.; SHAPIRA, B.; KANTOR, P. B. **Recommender Systems Handbook**. New York: Springer Science + Business Media, 2011.

- ROCHA, A. M.; NETO, R. M. F. **Introdução a Arquitetura Apple iOS**. 2011 Disponível em: <http://www.enacomp.com.br/2011/anais/trabalhos-aprovados/pdf/enacomp2011_submission_58.pdf>. Acesso em junho, 2017.
- SANTOS, A. **MySQL: Quem é você?**. Disponível em: <<http://www.devmedia.com.br/mysql-quem-e-voce/1752>>. Acesso em junho, 2017.
- SILVA, M. S. **Web Design Responsivo**. São Paulo: Novatec Editora Ltda, 2014.
- SCHMITT, R. **Os primeiros passos para o desenvolver para iOS**. Disponível em: <<http://www.devmac.com.br/2010/08/os-primeiros-passos-para-desenvolver-para-ios>>. Acesso em junho, 2017..
- SOFTERIZE, **CodeIgniter**. Disponível em: <<http://www.softerize.com.br/blog/o-que-e-codeigniter>>. Acesso em abril, 2017.
- SOUZA, R. G. D. **Sistemas de Recomendação: Aplicando Sistemas de Recomendação em Situações Práticas**. 2014. <http://www.ibm.com/developerworks/br/local/data/sistemas_recomendacao>. Acesso em abril, 2017.
- SQLite, **About SQLite**. Disponível em: <<https://sqlite.org/about.html>>. Acesso em abril, 2017.
- STARK, J. **Building Adroid Apps with HTML, CSS and JavaScript**. Sebastopol: O'Reilly Media, 2010.
- TORRES, R. **Personalização na Internet: como descobrir os hábitos de consumo de seus clientes, fidelizá-los e aumentar o lucro de seu negócio**. São Paulo: Novatec, 2004.