

Meu Mercado: Aplicativo Móvel para Publicação de Promoções de Supermercados

Fabricio Baques Bischoff¹, Eduardo Ferreira da Silva¹, Victor Machado Alves¹,
Carla Lisiane de Oliveira Castanho¹

¹Universidade Regional Integrada do Alto Uruguai e das Missões (URI) – Curso de
Ciência da Computação – Campus Santiago
Av. Batista Sobrinho, s/n – CEP 97.700-000 – Santiago – RS – Brasil

{fabriciobischoff27,eduardo.ferreiral983,victor.ccomp}@gmail.com,
carla.castanho@urisantiago.br

Abstract. Supermarkets are one of the places of commerce most visited by people, mainly because they contain basic necessary products for human consumption. However, the search for best prices usually occurs in two ways: make a manual evaluation at the supermarket; or advertising carried out through the media or even physical handout. Although the dissemination of products occurs, it can not be said that this is the best way to reach the specific public. The objective of this work is to develop a mobile application and a web manager on the Android Platform to persist the data. Allowing supermarkets to send notifications about the products on sale to their customers by sending notifications that will be triggered and sent to an application that will allow customers to view the promotions created by supermarkets.

Resumo. Os supermercados são um dos estabelecimentos de comércio mais procurados pelas pessoas, principalmente, por conterem produtos básicos e necessários para o consumo humano. No entanto, a busca por promoções, normalmente se dá de duas maneiras: deslocar-se até o supermercado para fazer uma consulta manual; ou a publicidade realizada através da mídia ou até mesmo panfletos físicos. Embora a divulgação dos produtos ocorra, não se pode afirmar que é a melhor forma de atingir o público específico. O objetivo deste trabalho consiste no desenvolvimento de um aplicativo móvel na plataforma Android e um gerenciador web para persistir os dados permitindo que os supermercados possam disponibilizar informações dos produtos em promoção aos seus clientes através do envio de notificações que serão disparadas e enviadas a um aplicativo que permitirá aos clientes visualizar as promoções criadas pelos supermercados.

1. Introdução

Um dos desafios enfrentados pelas empresas de supermercado, é descobrir diferentes formas de transmitir uma informação aos seus clientes. Durante muito tempo, essas formas eram limitadas ao uso de emissoras de rádio ou televisão e folhetos de papel contendo descrições de seus produtos. Com a evolução tecnológica, o uso da internet e o surgimento dos dispositivos móveis, é possível criar aplicações que tornem esse processo mais ágil. Logo, sem a necessidade de os clientes irem até o supermercado para fazer uma consulta manual sem a certeza de que irão encontrar produtos em promoção.

Esse trabalho propõe o desenvolvimento de um aplicativo móvel na plataforma Android para a divulgação de produtos promocionais de supermercados através do envio

de notificações que serão enviadas a um aplicativo que permitirá aos clientes visualizar as promoções criadas pelos supermercados. Ainda, serão utilizados métodos da engenharia de software e programação para o desenvolvimento do sistema. Espera-se que com o desenvolvimento desse aplicativo, o processo de busca de informação de produtos promocionais torne-se mais ágil e que tanto supermercado quanto cliente se beneficiem com o uso do aplicativo.

No primeiro momento da realização desse trabalho foi utilizado o método de pesquisa bibliográfica em livros e artigos científicos. Em um segundo momento da construção desse trabalho, foram aplicados conceitos e ferramentas da engenharia de software para a modelagem do sistema. Em um terceiro momento se deu início ao desenvolvimento do servidor web para persistir os dados, um Web Service REST para integrar a aplicação ao servidor e codificação inicial do protótipo do aplicativo. Por fim, será a realização dos testes da aplicação que ao final dos mesmos, espera-se que a mesma seja capaz de cumprir com o seu objetivo que é facilitar a disseminação de informação de produtos promocionais de supermercados.

1.1 Trabalhos Relacionados

O trabalho de Dos Santos (2013), teve como objetivo o desenvolvimento de um serviço de integração em Restful aliado a uma aplicação móvel útil tanto para o setor imobiliário, como para pessoas físicas que queiram efetuar o processo de divulgação de seu imóvel sem a necessidade de um intermediário, para venda e locação de imóveis. O Ambiente é formado por uma aplicação Web Service que tem a responsabilidade de atender e processar requisições efetuadas pela aplicação móvel, a API REST foi desenvolvida utilizando a linguagem de programação Java.

No trabalho de Fang (2014), é proposto o desenvolvimento do sistema Mbest Products que visa proporcionar ao usuário a fácil visualização de produtos de uma ou mais lojas que estão em promoção através de um aplicativo em seu dispositivo móvel e também fornecer a localização exata da empresa. O processo é feito com um usuário/funcionário habilitado cadastrando um ou mais produtos no sistema web de uma determinada loja. Após o cadastro, é enviado ao Web Service um arquivo JSON. Em seguida o sistema Android faz o processamento do arquivo e exibe os produtos.

Diante dos trabalhos apresentados observou-se os seguintes pontos em relação ao trabalho que será desenvolvido. Em relação ao desenvolvimento do aplicativo para visualizar imóveis para venda e locação, foi criada uma API REST na linguagem de programação Java, já o trabalho que está sendo proposto, possuirá um Web Service com uma API REST desenvolvida na linguagem de programação PHP. No segundo trabalho, destacamos que este faz uso de um sistema web para que o usuário/funcionário possa cadastrar os produtos no sistema. No trabalho que está sendo proposto, o cadastro e publicação de produtos promocionais é feito usando o próprio aplicativo móvel para que o usuário tenha a facilidade de realizar as operações de qualquer lugar.

2. Desenvolvimento Móvel

A plataforma Android foi criada por um grupo de empresas líderes no mercado de telefonia móvel, liderados pela Google. Esse grupo ficou conhecido como *Open Handset Alliance* (OHA) e tinham como objetivo criar uma plataforma moderna, flexível, de código aberto e livre para celulares. O Android é uma plataforma de desenvolvimento para aplicativos móveis para dispositivos como smartphones e contém um sistema operacional baseado no sistema operacional Linux. Outra característica importante é que

o Android utiliza a linguagem de programação Java para o desenvolvimento das aplicações usufruindo de todos os recursos que a linguagem oferece [Lecheta 2010].

Independentemente se o software for desenvolvido para dispositivos móveis ou computadores desktop, é necessário aplicar a engenharia de software pois, para Sommerville (2011) a engenharia de software pode ser vista como um processo de desenvolvimento que inicia na fase de especificação das necessidades do sistema até as fases finais de implementação e manutenção do mesmo com o objetivo de minimizar erros durante o desenvolvimento permitindo assim, diminuir custos com o processo de reengenharia de software.

3. Web Services

O *Web Service* fornece uma infraestrutura para manter uma forma mais rica e mais estruturada de interoperabilidade entre clientes e servidores. É uma tecnologia que permite que clientes interajam com servidores por meio de requisições e resposta transmitidas através do protocolo HTTP [Coulouris; Dollimore; Kindberg e Blair 2013]. Dornelles (2016), sugere o uso da estratégia Representational State Transfer (REST), uma forma de prover interoperabilidade na internet usando URLs para fornecer operações sobre um dado recurso.

Para Leal (2015), serviços web que usam o padrão REST são identificados por URLs para a descoberta de recursos e serviços e interação através da troca de mensagens do tipo requisição resposta. A interação com este padrão ocorre através de uma interface uniforme, que compreende um conjunto de operações no contexto da web e os métodos *GET*, *PUT*, *DELETE* e *POST* do protocolo HTTP para realizar as operações no servidor.

O padrão REST pode trocar mensagens usando o *JavaScript Object Notation* (JSON) que é um formato para transferência de dados reconhecido pela sua simplicidade e por proporcionar rapidez na troca de dados [Silveira; Oliveira e Oliveira 2016]. De Faria Fernandes e Cordeiro (2016) definem que o formato JSON é dito simples porque geralmente é mais fácil de ser processado do que o próprio XML, pois, consegue transferir altas taxas de informação entre o cliente e o servidor utilizando uma quantidade menor de caracteres.

4. Desenvolvimento do Aplicativo

Nesta seção, apresenta-se o desenvolvimento da modelagem do sistema através da UML dando ênfase aos diagramas de caso de uso e de classes. Em seguida, é apresentada a arquitetura Web Service REST empregada neste trabalho e a implementação do protótipo do aplicativo em desenvolvimento.

4.1 Modelagem do Sistema

O diagrama de caso de uso procura identificar os atores que utilizarão de alguma forma o software, bem como os serviços, ou seja, as funcionalidades que o sistema disponibilizará aos atores. A Figura 1 apresenta o diagrama de caso de uso do ator funcionário, responsável pelo cadastro das informações do supermercado.

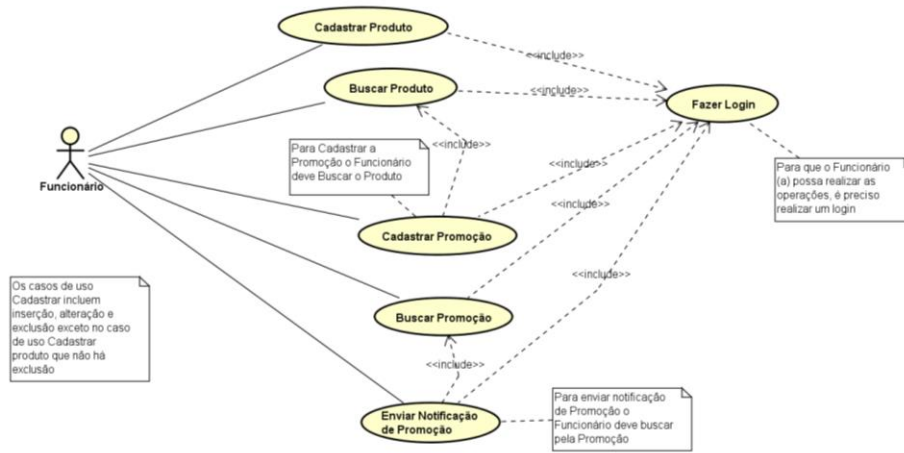


Figura 1 - Diagrama de caso de uso do ator Funcionário

Fonte: Elaborado pelo autor

Já o diagrama de classes tem como objetivo definir as classes utilizadas pelo sistema, assim como seus respectivos atributos, métodos e os relacionamentos entre si. A Figura 2 mostra o modelo de classes do sistema proposto, sendo composto pelas classes Usuário, Produto e Promoção, sendo que a promoção deve conter pelo menos um produto.

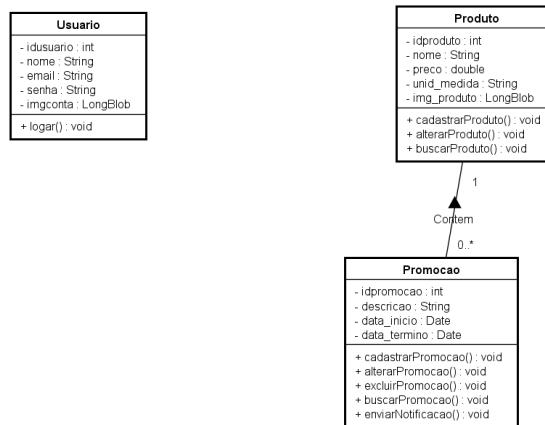


Figura 2 - Diagrama de Classes do Sistema

Fonte: Elaborado pelo autor

4.2 Infraestrutura

Em um segundo momento, iniciou-se o desenvolvimento de um servidor web local junto ao *software* MYSQL utilizado para persistir os dados. O *Web Service* ficará hospedado no servidor Apache e será responsável pela integração do sistema com o servidor *web*. Foi desenvolvida uma API REST na linguagem de programação PHP que utilizará os métodos *POST*, *PUT*, *DELETE* e *GET* do protocolo HTTP para realizar as respectivas requisições de inserção, alteração, exclusão e retorno das informações do servidor. A transferência de dados será realizada através de um arquivo JSON pois é um formato reconhecido pela sua simplicidade e por proporcionar rapidez na troca de dados. A Figura 3 representa a arquitetura de *Web Services Rest* que foi implementada neste trabalho.

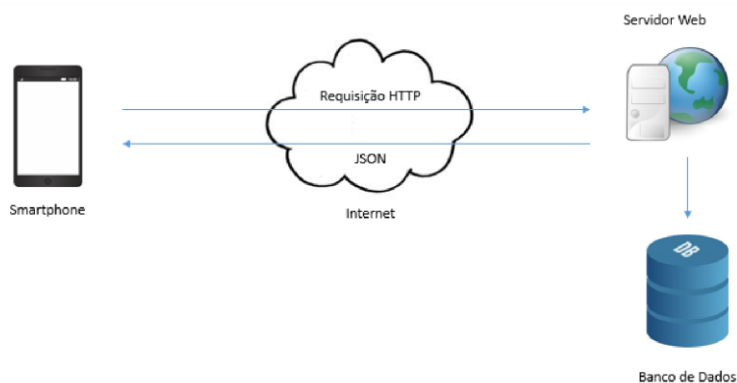


Figura 3 - Arquitetura Web Service REST

Fonte: Elaborado pelo autor

O código representando na Figura 4 é referente a operação *GET* do protocolo HTTP. Na linha 6 cria-se uma conexão com o servidor informando o nome do servidor, nome de usuário e senha para acesso. Na linha 11, espera-se qual método vai ser chamado pela aplicação, neste caso, como foi especificado na linha 13 será realizada uma operação do tipo *GET* para retornar as informações dos produtos e no formato JSON especificado na linha 27.

```

1 <?php
2 $servername = "localhost";
3 $username = "root";
4 $password = "mysql";
5
6 $conn = new mysqli($servername, $username, $password);
7 if ($conn->connect_error) {
8     die("Connection failed: " . $conn->connect_error);
9 }
10
11 $metodoHttp = $_SERVER['REQUEST_METHOD'];
12
13 if ($metodoHttp == 'GET') {
14     $jsonArray = array();
15     $sql = "SELECT idproduto, nome, preco, unid_medida FROM mercado_db.Produto";
16     $result = $conn->query($sql);
17     if ($result && $result->num_rows > 0) {
18         while($row = $result->fetch_assoc()) {
19             $jsonLinha = array(
20                 "idproduto" => $row["idproduto"],
21                 "nome" => $row["nome"],
22                 "preco" => $row["preco"],
23                 "unid_medida" => $row["unid_medida"]);
24             $jsonArray[] = $jsonLinha;
25         }
26     }
27     echo json_encode($jsonArray);
28 }
29 $conn->close();
30 ?>

```

Figura 4 - Operação GET para retorno dos produtos do servidor Web

Fonte: Elaborado pelo autor

Para que a aplicação consuma o serviço REST foi utilizada a biblioteca Retrofit que basicamente recebe um arquivo no formato JSON e retorna uma lista de objetos para a linguagem de programação Java [Square 2017]. Na linha 21 é definido a URL base para acesso ao *Web Service* REST. Já na linha 31 do código é definida uma instância da biblioteca Retrofit passando uma URL na linha 32 e a realização da serialização ou deserialização dos dados na linha 33. Na linha 36 é definida a instância da classe *ProdutoService* que contém a interface definindo o método *GET* com o complemento da URL e na linha 41, no método *onResponse* espera-se uma resposta do servidor se tudo ocorrer de forma correta é criada uma lista de produtos que será mostrada na tela da aplicação, se houver algum problema de comunicação com o servidor o método

onFailure, linha 56, é invocado mostrando o erro ocorrido. Este processo é mostrado na Figura 5.

```
21 private static final String BASE_URL = "http://192.168.40.2/meumercado_service/slim3/";
22 @Override
23 protected void onCreate(Bundle savedInstanceState) {
24     super.onCreate(savedInstanceState);
25     setContentView(R.layout.activity_main);
26
27     final ListView lista = (ListView) findViewById(R.id.listView);
28
29     Gson g = new GsonBuilder().registerTypeAdapter(Produto.class, new ProdutoDec()).create();
30
31     Retrofit retrofit = new Retrofit.Builder()
32         .baseUrl(BASE_URL)
33         .addConverterFactory(GsonConverterFactory.create(g))
34         .build();
35
36     ProdutoService service = retrofit.create(ProdutoService.class);
37     Call<List<Produto>> produtos = service.getProdutos();
38
39     produtos.enqueue(new Callback<List<Produto>>() {
40         @Override
41         public void onResponse(Call<List<Produto>> call, Response<List<Produto>> response) {
42             if (response.isSuccessful()) {
43
44                 final List<Produto> listaProduto = response.body();
45
46                 if (listaProduto != null) {
47                     ProdutoAdapter adapter = new ProdutoAdapter(getApplicationContext(), listaProduto);
48                     lista.setAdapter(adapter);
49                 } else {
50                     Toast.makeText(getApplicationContext(), "Erro: " + response.code(), Toast.LENGTH_LONG).show();
51                 }
52             }
53         }
54     });
55
56     @Override
57     public void onFailure(Call<List<Produto>> call, Throwable t) {
58         Toast.makeText(getApplicationContext(), "Erro: " + t.getMessage(), Toast.LENGTH_LONG).show();
59     }
60 }
```

Figura 5 - Consumo de Web Service com a biblioteca Retrofit
Fonte: Elaborado pelo autor

5. Resultados

Foram realizados testes iniciais com o objetivo de carregar os dados referentes aos produtos armazenados no servidor através do método *GET* do protocolo HTTP, juntamente com a utilização de um *Web Service* REST desenvolvido em PHP e o consumo do *Web Service* pela aplicação através da biblioteca Retrofit. Na Figura 6, apresenta-se o protótipo inicial da aplicação mostrando os dados básicos referente aos produtos carregados do *Web Service*. Salienta-se ainda que o projeto está em fase de desenvolvimento.

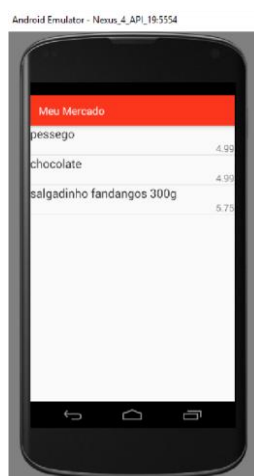


Figura 6 - Consumo de Web Service com a biblioteca Retrofit
Fonte: Elaborado pelo autor

6. Conclusão

Este trabalho apresenta o desenvolvimento de um aplicativo para notificação de promoções em supermercados. Para o desenvolvimento do projeto, explora-se a

implementação de um *Web Service* que padroniza os dados armazenados no servidor utilizando o framework REST e o JSON. Já o aplicativo está sendo desenvolvido utilizando a plataforma Android Studio e o banco de dados Mysql para persistência dos dados. Como trabalhos futuros sugere-se a ampliação do trabalho para que outros tipos de empresas possam usufruir da plataforma.

Referências

- Coulouris, George, et al. (2013) *Sistemas Distribuídos: Conceitos e Projeto*. 5. ed. Porto Alegre: Bookman, 2013. 784 p.
- De Faria Fernandes, Jorge Luiz; Cordeiro, Douglas Farias. (2016) Avaliação de formatos de publicação de dados abertos governamentais através de indicadores de usabilidade. *Tendências da Pesquisa Brasileira em Ciência da Informação*, v. 9, n. 1.
- Dornelles, T. de Azevedo. (2016) *Reconstrução do software AvalWeb usando conceitos de SPA, REST e NoSQL*. 2016. 64 f. Trabalho de Conclusão de Curso (Bacharelado)-Curso de Ciência da Computação, Universidade Federal do Rio Grande do Sul, Porto Alegre.
- Dos Santos, A. F. (2013) *Metodologia de desenvolvimento de uma API de serviço Restful integrado com uma aplicação móvel android para o setor imobiliário*. 61f. Trabalho de Conclusão de Curso (Bacharelado)-Curso de Sistemas de Informação, Centro Universitário Eurípides de Marília, Marília, 2013.
- Fang, M. A. I. (2014) *Mbest Products: Produtos em Promoção apresentados em dispositivos móveis*. 2014. 407 f. Trabalho de Conclusão de Curso (Especialização)Curso de Tecnologia em Análise e Desenvolvimento de Sistemas, Universidade Federal do Paraná, Curitiba, 2015.
- Leal, Nelson G. de V. (2015) *Dominando o android: do básico ao avançado*. 2. ed. São Paulo: Novatec, 952 p.
- Lecheta, Ricardo R. (2010) *Google Android: aprenda a criar aplicações para dispositivos móveis com o Android SDK*. 2. ed. São Paulo: Novatec Editora, 608 p.
- Silveira, P. F. K.; Oliveira, R. R.; Oliveira, R. R. (2016) *Uma tecnologia social baseada em crowdsourcing para busca de pessoas desaparecidas e cadastro de moradores de rua*. XIII SIMPÓSIO BRASILEIRO DE SISTEMAS COLABORATIVOS. 2016.
- Pontifícia Universidade Católica do Rio Grande do Sul. *Anais eletrônicos...* Porto Alegre, 2015. Disponível em: <<http://ebooks.pucrs.br/edipucrs/anais/csbc/assets/2016/sbsc/14.pdf>>. Acesso em: 18 de maio de 2017.
- Sommerville, Ian. (2011) *Engenharia de Software*. 9. ed. São Paulo: Pearson Addison Wesley. xiv, 592 p.
- Square, I. *Retrofit*. 2017. Disponível em: <<http://square.github.io/retrofit/>>. Acesso em: 08 de setembro de 2017.