

S-Gestão

Aplicativo Móvel de Gestão

Sidnei Oliveira Bandeira¹, Edécio Fernando Iepesen¹

¹Análise e Desenvolvimento de Sistemas
Faculdade de Tecnologia SENAC Pelotas (FATEC)
Rua Gonçalves Chaves 602 – 96015560 – Pelotas – RS – Brazil

`sid-bandeira@hotmail.com, edeciofernando@gmail.com`

Abstract. *This article aims to present the application S-Management (Mobile Application Management), an application that aims to assist business managers in decision-making, control inventory of goods and monitor the daily financial movement of the company. With the growing need for real-time information and the widespread popularization of smartphones, a mobile application increasingly helps the dissemination of information. This article will detail the process for mobile application development such as database modeling, use cases, file model for data transfer and application features.*

Keywords: *Android, Web service, MySQL, control, management.*

Resumo. *Este artigo tem por objetivo a apresentação do aplicativo S-Gestão (Aplicativo Móvel de Gestão), aplicativo que visa auxiliar gestor de empresas nas tomadas de decisões, controlar estoque de mercadorias e acompanhar a movimentação financeira diária da empresa. Com a crescente necessidade de informações em tempo real e a ampla popularização de Smartphones um aplicativo móvel auxilia cada vez mais a disseminação das informações. Neste artigo será detalhado o processo para desenvolvimento do aplicativo móvel como modelagem do banco de dados, casos de uso, modelo de arquivos para transferência de dados e funcionalidades do aplicativo.*

Palavras-Chave: *Android, Web Service, MySQL, controle, gestão.*

1. Introdução

No Brasil a utilização de *Smartphones* não para de crescer, a cada ano mais pessoas aderem ao uso e com isso temos a necessidade de aplicativos que auxiliem nossas vidas, seja para uma tarefa mais simples como chamar um transporte ou buscar informações para tomada de decisões.

O desenvolvimento de aplicativos corporativo possibilita uma gama infinita de aplicações para auxiliar no dia a dia das empresas, desta forma independente da área de atuação, sempre será possível desenvolver um aplicativo que auxilie em uma tarefa. Segundo pesquisa da SOCIOMANTIC [SocioMant 2016] de 2013 a 2015 tivemos um crescimento de 40% na utilização de *Smartphones* sendo 78% com o sistema operacional Android.

Visando este mercado foi desenvolvido o aplicativo S-Gestão (aplicativo móvel de Gestão) que tem a finalidade de auxiliar o gestores de empresas nas tomadas de decisões, controlar estoque de mercadorias e acompanhar a movimentação financeira diária de suas empresas.

2. Referencial Teórico

O desenvolvimento deste projeto surgiu da necessidade dos gestores terem em mãos as informações de suas empresas em qualquer lugar e a qualquer hora. Foi observado que esta área de mercado esta em crescente expansão nos dias atuais, desta forma o projeto foi estruturado para permitir a integração com diferentes tipos de sistemas utilizando arquivo JSON(*JavaScript Object Notation*) facilitando o processo de integração independente das linguagens de programação utilizadas.

Para chegar a este modelo foi utilizado como referência o software de desenvolvimento Umov.me [Umov.me 2017] uma plataforma de desenvolvimento de aplicativos corporativos, onde cada empresa pode desenvolver o aplicativo conforme sua necessidade, mas limitando-se ao layout e regras de distribuição e licenças da empresa Umov.me.

O diferencial do aplicativo S-Gestão é que todas as alterações, customização, controle de desenvolvimento de novos recursos e distribuição ficam a cargo do desenvolvedor. A integração será feita através de um *Web Service* que será responsável por alimentar uma base de dados disponibilizada na internet, esta integração será feita através de arquivos com layout específicos(ver anexo) e no formato JSON. Para o envio do arquivo será necessário que o cliente possua uma chave criptografada no formato MD5(*Message-Digest algorithm 5*) que será validada junto ao seu nome de usuário e senha no momento do envio. Da mesma forma o usuário do aplicativo deverá se conectar ao *Web Service* através de um usuário e senha de forma que tenha acesso somente as informações disponibilizadas para seu usuário.

2.1. Web service

Fábio Perez [Marzullo 2009] define *Web Service* como a disponibilização de um serviço que pode ser acessado por meio da Internet. Representa uma lógica de negócio bem definida permitindo a interação com clientes, os quais enviam requisições bem definidas e recebem respostas síncronas, na qual o cliente fica bloqueado esperando o serviço completar sua execução, e assíncronas não havendo necessidade de esperar o término da execução. Serviços expostos como *Web Service* tem lógicas de negócio que devem encapsular regras ou funcionalidades que representem regras de negócio. Devem ser publicados e acessados por intermédio de uma linguagem padrão de publicação e um protocolo específico de comunicação.

Web Service é a tecnologia ideal para comunicação entre sistemas, sendo muito usado em aplicações B2B. A comunicação entre os serviços é padronizada possibilitando a independência de plataforma e de linguagem de programação[JavaFree 2017].

2.2. Criptografia

Criptografia pode ser entendida como um conjunto de métodos e técnicas para cifrar ou codificar informações legíveis através de um algoritmo, convertendo um texto original em um texto ilegível, sendo possível através do processo inverso recuperar as informações originais. [Singh 2000]

Segundo [Kahn 1996], o primeiro exemplo documento da escrita cifrada aconteceu aproximadamente no ano de 1900 a.C, quando o escriba de Khnumhotep II teve a ideia de substituir algumas palavras ou trechos de texto. Caso o documento fosse rou-

bado, o ladrão não encontraria o caminho que o levaria ao tesouro e morreria de fome perdido nas catacumbas da pirâmide.

Segundo [Vale 2007], o Message-Digest Algorithm5 (MD5) é um algoritmo de hash de 128 bits que foi desenvolvido pela RSA Data Security, com a finalidade de substituir o algoritmo MD4 que apresentou falhas de segurança. O MD5 é utilizado basicamente para verificação de integridade e logins.

Ainda de acordo com [Vale 2007], o MD5 é um algoritmo unidirecional no qual através de uma saída não é possível identificar o conteúdo de entrada. Portanto, o método de verificação de hash é dado pela comparação entre dois hash, um armazenado na base de dados e outro gerado a partir da tentativa de login por exemplo.

[Vale 2007] ressalta ainda que, independente da mensagem de entrada, o MD5 gera um valor hash de tamanho fixo de 128 bits.

3. Desenvolvimento do sistema

Nesta seção serão apresentados as etapas do desenvolvimento do sistema, tecnologias utilizadas e diagramas do sistema.

3.1. Tecnologias Utilizadas

Para o desenvolvimento do projeto foram utilizadas as tecnologias Php para o desenvolvimento do *Web Service*, Mysql como banco de dados e Android Studio plataforma de desenvolvimento do aplicativo.

3.1.1. PHP

Segundo Dall'Obligio [Dall'Oglio 2015], a linguagem de programação PHP (*Hypertext Preprocessor*), que no início significava *Personal Home Page Tools*, foi criada em 1994 por Rasmus Lerdorf. Essa linguagem era formada por um conjunto de scripts escritos em linguagem C, voltados à criação de páginas dinâmicas. Em 1995, o código-fonte do PHP foi liberado, e com isso mais desenvolvedores puderam se juntar ao projeto. Ao longo de mais de uma década, o PHP vem adicionando mais e mais recursos e se consolida ano após ano como uma das linguagens de programação orientadas a objetos que mais crescem no mundo.

3.1.2. MySQL

O MySQL é um servidor e gerenciador de banco de dados (SGBD) relacional, de licença dupla (sendo uma delas de software livre), projetado inicialmente para trabalhar com aplicações de pequeno a médio portes, mas hoje também atendendo a aplicações de grande porte. Possui todas as características que um banco de dados de grande porte precisa, sendo reconhecido por algumas entidades como o banco de dados *open source* com maior capacidade para concorrer com programas similares de código fechado, tais como SQL Server e Oracle. O MySQL teve origem quando os desenvolvedores David Axmark, Allan Larsson e Michael "Monty" Widenius, na década de 90, precisaram de uma interface

SQL compatível com as rotinas ISAM(*Indexed Sequential Access Method*) que utilizavam em suas aplicações e tabelas. Utilizando a API do mSQL, escreveram em C e C++ uma nova API que deu origem ao MySQL. A partir desta fase, o MySQL tornou-se mais conhecido por suas características de rápido acesso e cada vez mais utilizado. Atualmente encontra-se na versão 5.1 [Milani 2007].

3.1.3. Android Studio

O Android Studio é o ambiente de desenvolvimento integrado (IDE) oficial para o desenvolvimento de aplicativos Android e é baseado no IntelliJ IDEA (ambiente de desenvolvimento integrado java). Além do editor de código e das ferramentas de desenvolvedor avançados do IntelliJ, o Android Studio oferece ainda mais recursos para aumentar sua produtividade na criação de aplicativos Android. O Android Studio usa o Gradle como o sistema de compilação de base, com outros recursos específicos do Android sendo disponibilizados pelo *Android Plugin for Gradle*. Esse sistema de compilação é executado como uma ferramenta integrada no menu do Android Studio e de forma independente na linha de comando conforme [Android 2017].

3.1.4. Android 4.4 KitKat

Lançado em outubro de 2013, o KitKat trouxe o Android para todos, pois conseguiu executar o sistema operacional mesmo em dispositivos com menos de 512MB de RAM, devido às diversas melhorias de desempenho e otimizações feitas no sistema operacional. Conforme o artigo de Felipe Gugelmin[Gugelmin 2016] em agosto de 2016, foi divulgado pelo google um relatório dos sistemas operacionais Android mais utilizados conforme Figura 1, são eles Lollipop(35.5%), KitKat(29.2%), Jelly Bean(16.7%) e Marshmallow(15.2%).

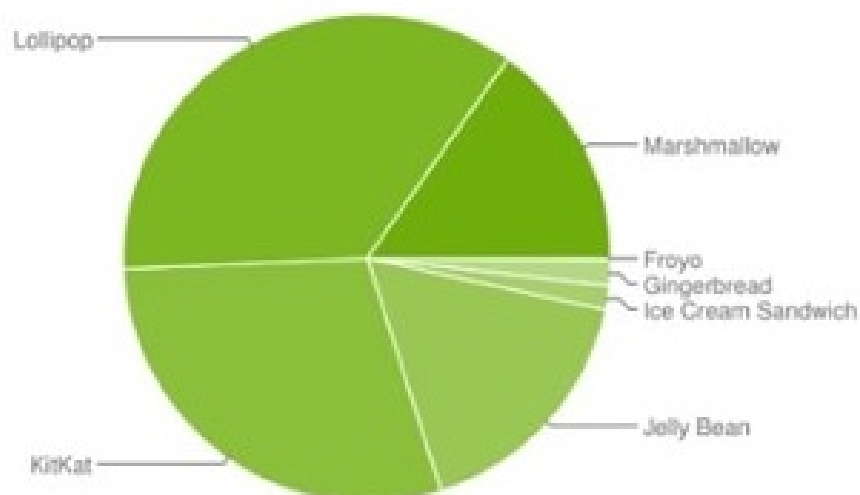


Figura 1. Gráfico de versões Android mais utilizadas

3.2. Modelagem do Banco de Dados

A Figura 2 representa o modelo de entidade e relacionamento do banco de dados disponibilizado na *web* para o armazenamento dos dados as serem exibidos no aplicativo móvel. Nesta representação temos 11 tabelas que serão responsáveis por armazenar as informações necessárias para a utilização do aplicativo móvel S-Gestão. Destas tabelas a principal é a tabela "empresa", esta tabela é responsável por armazenar as informações das empresas, sendo o campo "token" o mais importante pois somente com este token será possível tanto o envio dos dados ao banco de dados quanto o consumo deste banco pelo aplicativo móvel. Outra tabela que será alimentada pelo desenvolvedor é a tabela "Menu" visto que esta tabela é responsável por liberar os menus do aplicativo conforme o ramo de atividade do cliente, pois no aplicativo temos recursos específicos a postos de combustíveis. Após a empresa cliente estar cadastrada na tabela "empresa" e seus menus configurados será disponibilizado ao cliente um token e com isso já estará liberado para enviar os dados para alimentar as outras tabelas do banco de dados. A tabela "unidade-negocio" armazenara os dados da empresa matriz e filiais, desta forma será possível que o usuário ao se logar no aplicativo possa escolher qual de suas empresas ele deseja visualizar as informações, a partir desta tabela é que será disponibilizado as informações de venda, geração de pedidos de compra e em caso de cliente do ramo de postos de combustíveis as informações do saldo dos tanques de combustíveis que estará armazenado na tabela "tanque". Também fazem parte deste modelo as tabelas "fornecedor", "produto" e "produtofornecedor" que armazenaram os dados referente ao cadastro de produto e fornecedor do cliente que será utilizada na geração do pedido de compra.

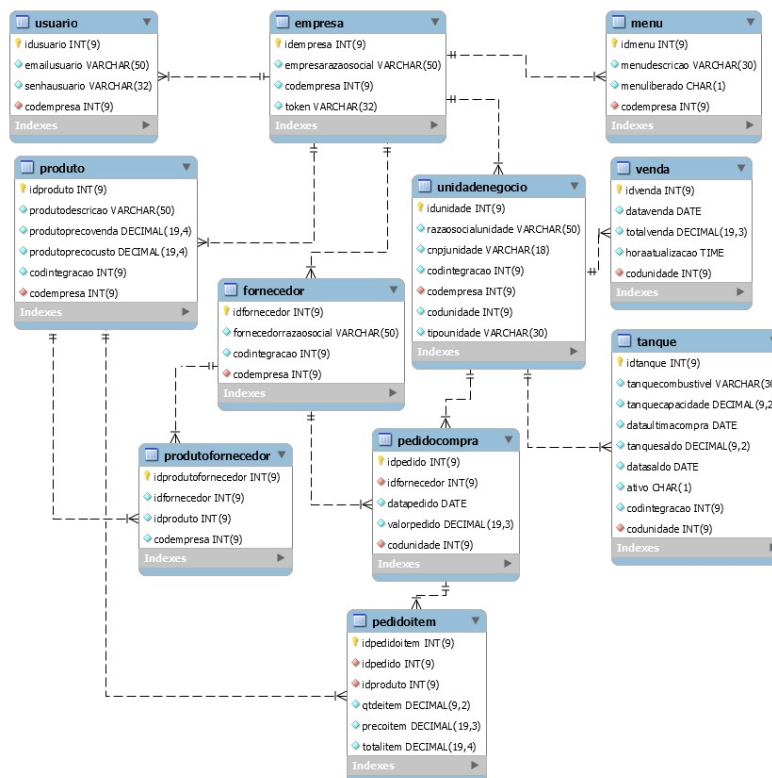


Figura 2. Modelo de Entidade e Relacionamento

3.3. Diagrama de Casos de Uso

A Figura 3 demonstra o cenário das funcionalidades do aplicativo, temos um ator que é responsável por consumir o aplicativo móvel. O ator "usuário aplicativo" terá as funcionalidades de logar no sistema e após executar o *login* poderá gerar relatórios em tela do estoque de produtos, venda diária, visualizar graficamente o saldo de combustíveis e também será capaz de gerar pedidos de compra de mercadorias.

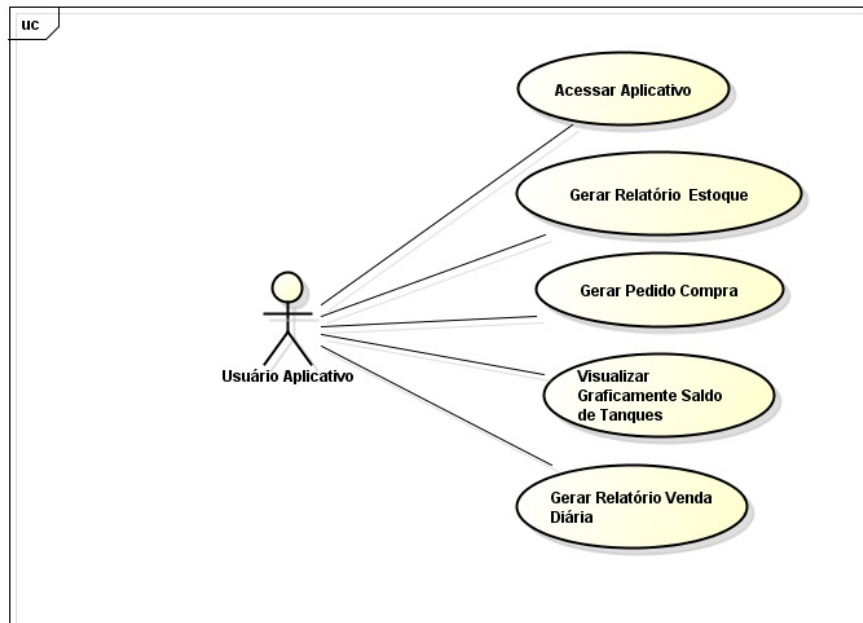


Figura 3. Diagrama de Casos de Uso

4. Descrição do sistema

(ESCREVER)

5. Conclusões

(ESCREVER)

5.1. Dificuldades encontradas

(ESCREVER)

5.2. Trabalhos Futuros

desenvolver portal para acesso do webservice.

5.3. Considerações Finais

(ESCREVER)

Referências

- Android (2017). <https://developer.android.com/studio/intro/index.html?hl=pt-br>. Acesso em: 2017-04-18.
- Dall'Oglio, P. (2015). *PHP Programando com Orientação a Objetos*. Novatec.
- Gugelmin, F. (2016). Versões mais usadas do android. <https://goo.gl/UdG8wP>. Acesso em: 2017-04-16.
- JavaFree (2017). <https://goo.gl/BPZB0f>. Acesso em: 2017-04-14.
- Kahn, D. A. (1996). *Scribner Book Company*. Scribner Book Company.
- Marzullo, F. P. (2009). *SOA na Prática*. Novatec.
- Milani, A. (2007). *MySQL - Guia do Programador*. Novatec.
- Singh, S. (2000). *The Code Book: The Secret History of Codes and Code-breaking*. HarperCollins Publishers.
- SocioMant (2016). Sociomant from dunnhumby. <https://www.sociomantic.com.br/infografico-mobile-br/>. Acesso em: 2017-04-05.
- Umov.me (2017). Umov.me - plataforma de mobilidade completa. <http://www.umov.me/>. Acesso em: 2017-04-01.
- Vale, C. R. (2007). Criptografia md5. <http://www.devmedia.com.br/criptografia-md5/2944>. Acesso em: 2017-04-12.