KIU - KARIRI INFORMAÇÃO UTILITÁRIA

Documento de Arquitetura

Projeto KIU

**Versão 1.0** 09 de dezembro de 2015

Responsáveis:

Dagno da Silva Souza

Alba Maria Barros Freires

**Histórico de revisões**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versão | Descrição | Autor |
| 27/11/2015 | 0.1 | Versão Inicial do documento de Arquitetura | Dagno da Silva Souza |
| 02/12/2015 | 0.2 | Versão de desenvolvimento do documento de Arquitetura | Dagno da Silva Souza |
| 07/12/2015 | 0.3 | Revisão do documento de arquitetura | Dagno da Silva Souza |
| 08/12/2015 | 0.4 | Modificação no documento de arquitetura | Alba Maria Barros Freires |
| 09/12/2015 | 1.0 | Revisão final do documento | Alba Maria Barros Freires |

1. Introdução

Neste documento iremos tratar os padrões de arquitetura proposto pelo projeto KIU Mobile. Mostrando as camadas de negócio, visões de projeto, tipos de cliente e sua conectividade com o aplicativo. Será utilizado a arquitetura Cliente-Servidor, onde um ou mais dispositivos clientes (DC) solicitam dados do servidor.

1. Objetivos

O documento de arquitetura objetiva uma visão geral do escopo da aplicação, com intuito de facilitar o entendimento do projeto e a facilitar a comunicação entre o arquiteto e o restante da equipe. Também será utilizado caso seja introduzido novos integrantes a equipe, pois esse documento terá de formas gerais as informações necessárias para o entendimento arquitetural do aplicativo.

1. Responsabilidades

O arquiteto tem como responsabilidade elaborar o documento de arquitetura e manter a integridade de seus dados, assim como fazer as modificações nesse documento caso necessário. Para isso ele deve documentar todas as mudanças arquiteturais e participar das reuniões que podem arcar com mudanças na arquitetura do projeto, assim como resolver problemas relacionados a arquitetura.

1. Referencias

Lee, V., Schneider, H. e Schell, R. em Aplicações Moveis: arquitetura, projeto e desenvolvimento. Traduzido. Amaury Bentes & Deborah Rüdiger. Pearson, 2005

SOMMERVILLE, Ian. Engenharia de software, 8° edição. Pearson,2007

1. Arquitetura
2. Cliente Servidor

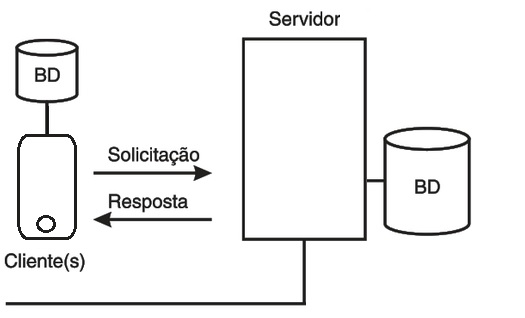
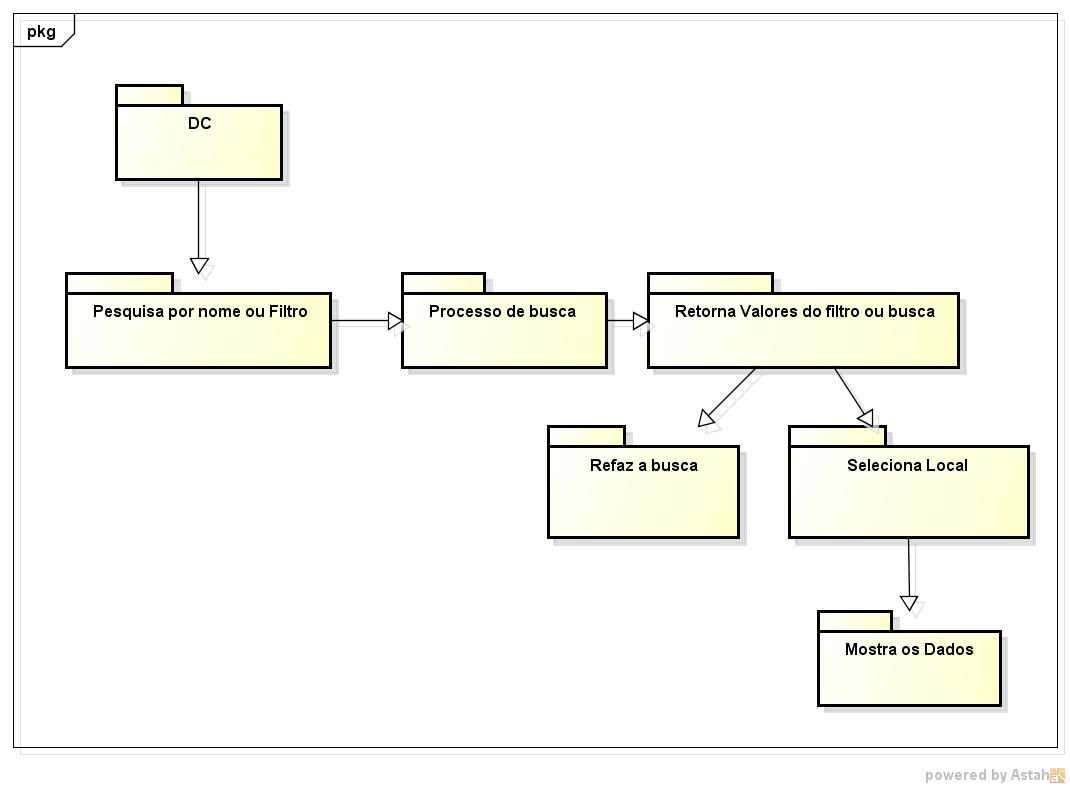


Diagrama 1.

No Diagrama 1, vemos como o programa se portará quanto as requisições feitas pelos DC’s que terão um banco de dados salvo em seu dispositivo para informações mais básicas e corriqueiras. Essas informações serão compostas por categorias como por exemplo delegacias, hospitais, prefeituras, bancos, pontos de utilidade pública ou supermercados. Quando conectado com a internet esses dados presentes no DC será atualizado caso haja mudança de endereços, telefones e etc. assim como cadastros de locais feitos com DC off-line. O serviço online será mais abrangente, pois estarão dispostos todos os locais cadastrados em nossa base de dados e também será possível avaliar o local, cadastrar e editar informações de locais em tempo real (ambas ações passarão por um teste de veracidade de dados.

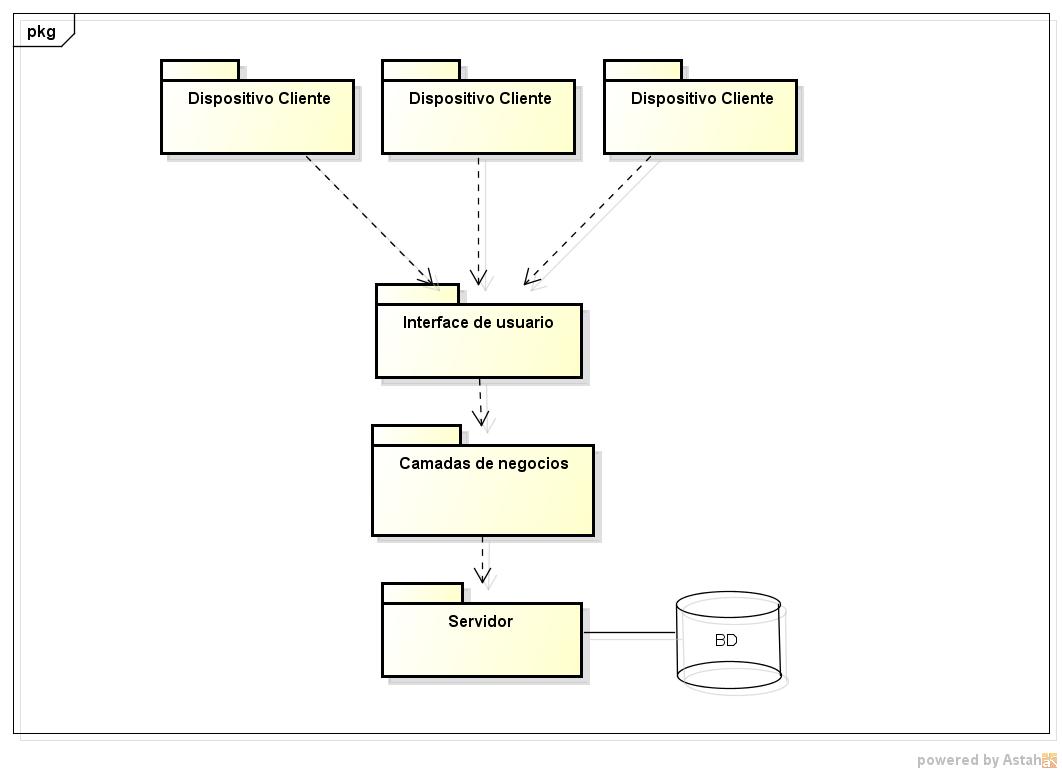
1. Processamento de Dados

Os sistemas de processamentos de dados são sistemas de processamentos em lotes nos quais os dados são entradas e saídas em lotes de um arquivo ou banco de dados em vez de entradas ou saídas ou saídas para um terminal de usuário. (SOMMERVILLE, Ian. Engenharia de software, 8° edição)

Nossa aplicação gira em torno dos dados, de acordo com os dados de entrada os dados de saída vão se alterar, mas sempre vai partir dessa premissa de troca de informações, o DC irá inserir dados para sua busca (como nome do local ou uma pesquisa por filtro por exemplo) e em seguida será retornado do banco de dados a lista de locais compatíveis com a pesquisa ou filtro sugerido pelo DC, vejamos o diagrama a seguir. 

1. Divisão de Camadas

As camadas são a divisórias de códigos dentro da DC ou servidor, a quantidade de camadas irá definir o tipo de cliente que utilizará nosso sistema. No lado do cliente podem existir de zero a três camadas, no lado do servidor pode existir de uma a três camadas, em nosso caso, ambos os lados vão apresentar três camadas como mostra o diagrama a seguir.



1. Ferramentas arquiteturais

* Phonegap e Jquery Mobile
* MySQL
* Java Enterprise Edition – JEE
* Tomcat

1. Phonegap e Jquery Mobile

Para fazer o front-end do projeto KUI, a equipe escolheu o Phonegap com as bibliotecas de Jquery Mobile, devido ao software se mostrar de maneira simples. Por ser um aplicativo simples de se mostrar o phonegap é a ferramenta mais indicada, pois é toda implementada de forma web, além do fato de gerar os arquivos executáveis para as plataformas necessárias para nosso projeto que inicialmente serão Android e IOS.

1. MySQL

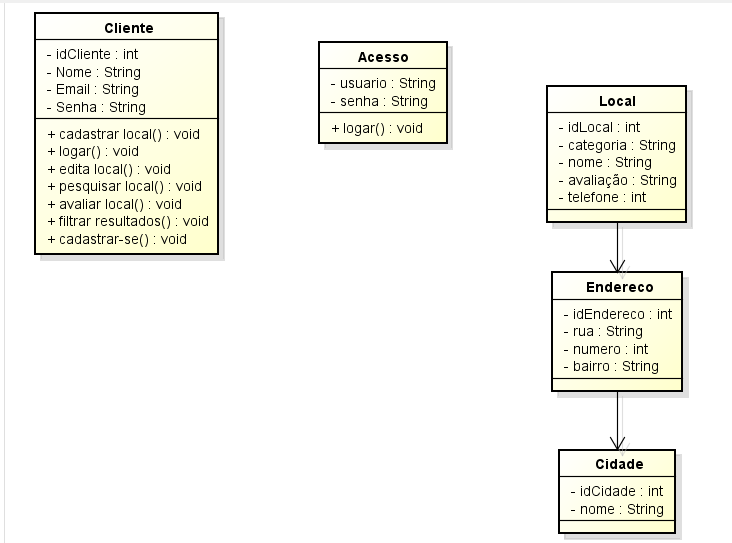
Para banco de dados será usado o MySLQ, porque supre as necessidades do aplicativo, ele atende projetos pequenos ou robustos, então apesar de nosso aplicativo começar com um fluxo pequeno de dados, futuramente caso os dados aumentem o MySQL tem um banco de dados expansível. Além do fato de ser um banco totalmente gratuito e código aberto.

1. Java Enterprise Edition e Tomcat

Para desenvolver o back-end vamos utilizar o JEE juntamente com o tomcat para servidor. A ferramenta JEE é uma ferramenta da Oracle que pode-se trabalhar com servlets e páginas JSP, e o servidor tomcat é capaz de atuar em página HTML/HTTP, como o front-end do aplicativo vai ser desenvolvido de forma WEB, optamos por essa ferramenta.

1. Diagrama de classes

A seguir o diagrama de classes descrevendo o modelo geral do aplicativo com os objetos e seus respectivos atributos e métodos:



1. Diagrama Entidade-Relacionamento

O diagrama a seguir representa as tabelas no banco de dados, contém as entidades e seus respectivos relacionamentos.

