GliCHECK

Projeto Arquitetural

There is guidance within this template that appears in a style named InfoBlue. This style has a hidden font attribute that allows you to toggle whether it is visible or hidden in this template. Use the Microsoft® Word® menu **Tools > Options > View > Hidden Text** check box to toggle this setting. There is also an option for printing: **Tools > Options > Print**.

# Objetivo

Este documento tem o objetivo de descrever filosofia, decisões, restrições, justificativas, premissas e quaisquer outros aspectos que dão forma ao projeto e sua implementação.

# Metas Arquiteturais e Filosofia

O principal direcionador da escolha foi a necessidade de monitoração de uma doença que afeta cerca de 250 milhões de pessoas no mundo. O aplicativo não precisa de acesso à internet, uma vez que o banco de dados será interno no Android. O usuário irá inserir os dados de forma verdadeira as medições, para que o aplicativo cumpra com sua premissa de monitoração da Diabetes, assim como os medicamentos do paciente.

# Premissas e Dependências

# Requisitos Críticos da Arquitetura

<http://developer.android.com/index.html>

# Decisões, Restrições e Justificativas

# Mecanismos Arquiteturais

# 

Na camada **Linux Kernel** (Baseado no Kernel 2.6 do Linux), é onde todos os serviços de baixo nível é executado, como por exemplo, controle de processos, gerenciamento de memória, threads, protocolo de redes, modelo de drivers e a segurança de arquivos.

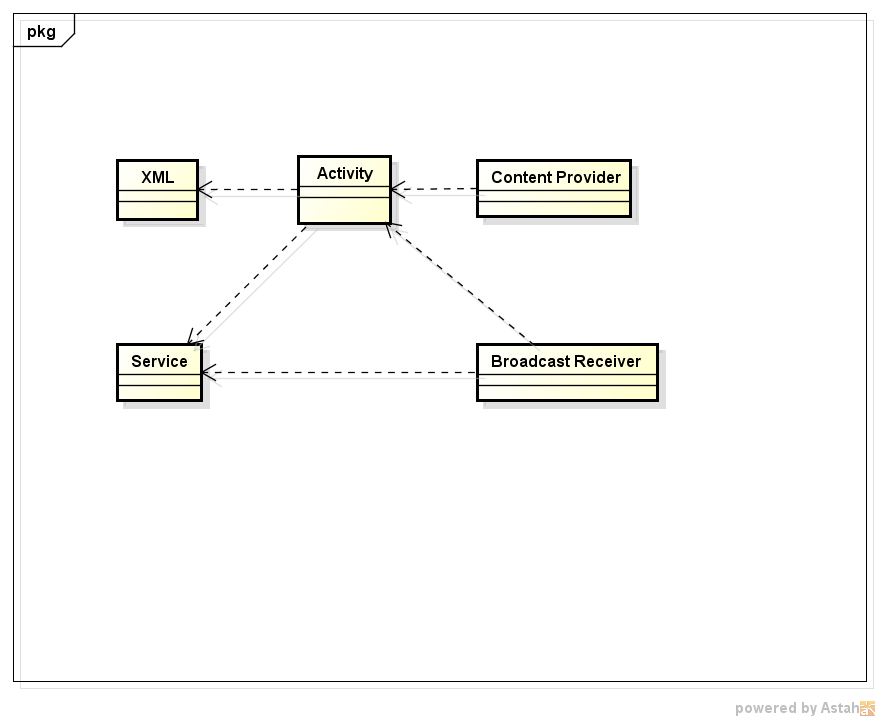
Na camada **Libraries** é onde estão todas as bibliotecas do Android, que estão escritas em C/C++, que nada mais é que um conjunto de instruções que dizem ao dispositivo como lidar com diferentes tipos de dados. Nelas estão inseridas também as bibliotecas de multimídia, visualização de camadas 2D e 3D, funções para navegadores web, funções de aceleradores de Hardware, onde engloba a Renderização 3D para gráficos. E muito importante, os acessos ao Banco de Dados SQLite.

Na camada **Android Runtime,** ondefica localizado no mesmo nível da camada de Libraries, é onde se instancia a máquina virtual do Android, chamada de Dalvik. Ela é criada para cada aplicação executada no Android, isso é importante por algumas razoes, por exemplo, cada programa independe de outro, se um programa para, ela não afeta nenhum outro rodando no momento, e na simplificação do gerenciamento de memória.

Na camada **Application Framework** é onde os programas gerenciam funções básicas do telefone, como alocação de recursos, aplicações do telefone, mudança de processos ou programas, além de ficar de olho na localização física do aparelho. Os desenvolvedores têm acesso total a essa camada, possibilitando que eles tirem vantagens dos recursos que o Android oferece quando estão construindo uma aplicação na plataforma.

E na camada **Applications,** é ondese localiza todos os aplicativos que estão sendo executados sobre o sistema operacional, por exemplo, cliente de SMS, e-mail, navegador, mapas, calculadora, entre outros.

# Principais Abstrações



# Camadas do Framework da Arquitetura

**Activity**: Em curtas palavras, é a tela que é exibida ao usuário. Possui a interface composta por Views, componentes gráficos, eventos, entre vários outros. Para cada Activity, existe um arquivo XML associado a ela, onde esse arquivo ira ser usado para configurar como os componentes de visualização irão ser mostrados na tela.

**Broadcast Receiver:** Trata uma reação de evento externo, por exemplo, atender uma ligação, receber um SMS. Esses eventos são avisados, e para aqueles programas que desejam responder essas ações, podem faze-las utilizando a classe BroadcastReceiver.

**Service**: é um programa que roda por si mesmo sem necessidade de uma interface de usuário. Um exemplo de uso desse serviço é quando você está usando algum navegador para acessar a internet e de repente recebe uma mensagem, você minimiza o navegador para responder a essa mensagem, de vez de seu navegador ser fechado, ele ficara em segundo plano.

**Content Provider**: Serve para compartilhar dados entre aplicativos Android. Ele consegue armazenar e recuperar dados através de um repositório. Lembrando que um Banco de Dados local não é acessível para outro programa. Isso somente acontece usando os recursos do Provedor de Conteúdo.

# Visões Arquiteturais

• Lógica:

• Operacional:

• Caso de uso:

Android: